IISPS_INT_2062_LIFE EXPECTANCY PREDICTION

USING MACHINE LEARNING

PROJECT REPORT

SUBMITTED BY:

CHAITANYA GOEL goelchaitanya1998@gmail.com

Academic email ID: SI05202000905@smartinternz.com

ACKNOWLEDGEMENT

I WOULD LIKE TO EXPRESS MY GRATITUDE TO **SMARTBRIDGE** FOR ALLOWING ME TO LEARN NEW A TECHNOLOGY. MY MENTORS, **MR PRASHANTH**, **MR MCHARANREDDY** AND **MS.LALITHA GAYATRI LOLLA,** WERE VERY SUPPORTIVE AND WERE ALWAYS AVAILABLE TO HELP WITH FULL ENTHUSIASM. THEY MADE MY JOURNEY EASIER IN COMPLETING MY PROJECT ON THE TOPIC **"PREDICTING LIFE EXPECTANCY USING MACHINE LEARNING "**.

THE INTERNSHIP WAS INQUISITIVE AT EVERY STEP AND VERY INFORMATIVE. I WOULD ALSO LIKE TO THANK MY PARENTS WHO WERE ALWAYS STANDING BESIDE ME IN THE NEED OF THE HOUR.

CONTENTS

INTRODUCTION

OVERVIEW

Life expectancy is a statistical measure of the average time a human being expects to live. Life expectancy depends on various factors, such as:

1. Regional variations
2. Economic circumstances
3. Sex differences
4. Mental illness
5. Physical illness
6. Year of birth

There are other demographic factors that affect the life expectancy of a person, but they have a varying impact on the health of a person. It mainly depends on an individual basis.

This project gives us an insight into the future of the life expectancy of the people based on the present data in hand. The dataset used in this project has been provided by the WHO, listing the various features that affected the life expectancy of people from the year 2000 – 2015.

The project is in python 3.8, with the collaboration of IBM Cloud services. The model created was deployed with the help of Node-Red application that is based on the Node.js framework.

The IBM cloud services assisted in launching the model to the internet in the form of a webpage by which anyone could predict the life expectancy just by entering the details demanded by the model working in the backend of the application.

PURPOSE

The prediction of Life expectancy is crucial as it gives a reality check to the people as well as helps the government in realising their shortcomings that resulted in such a major catastrophe.

Some significant and broad applications of this in the world includes:

- **Child Birth Prediction:** The prediction of the age of a child can be of great help to the parents as it would help them in planning the nutritional needs of their child and other things they may have to take care of at the time of upbringing of their child.
- **Improving Health Conditions:** The government of various countries have to take care of the health needs of their people. The life expectancy prediction would help in giving the government an idea of the major and the minor factors affecting the health of their people, where the government itself is lacking behind. Thus, they can do the planning and take the optimum steps towards that.
- **Research purposes:** Life expectancy predictions may help actuarial sciences and other areas of research where the life expectancy is studied to form schemes and other policies.

  Also, various scientists may study this data and predict whether a particular area is capable of handling an epidemic and if it spreads which areas may be immune to prevent its further flow.

- **Formation of Policies and schemes:** Government, as well as companies, may follow this prediction data to form their policies that may be beneficial for their customers and people.

  The most evident and significant application would be to form retirement and pension schemes and policies related to it that may help the people depending upon the results predicted.

India is to become a superpower in the next 20-30 years. It is the result of the prediction of the demographics that are currently available. The primary factor affecting the demographics is the prediction of life expectancy of the people living in India which shows that the population of the young people would be a lot more as compared to the rest of the world.

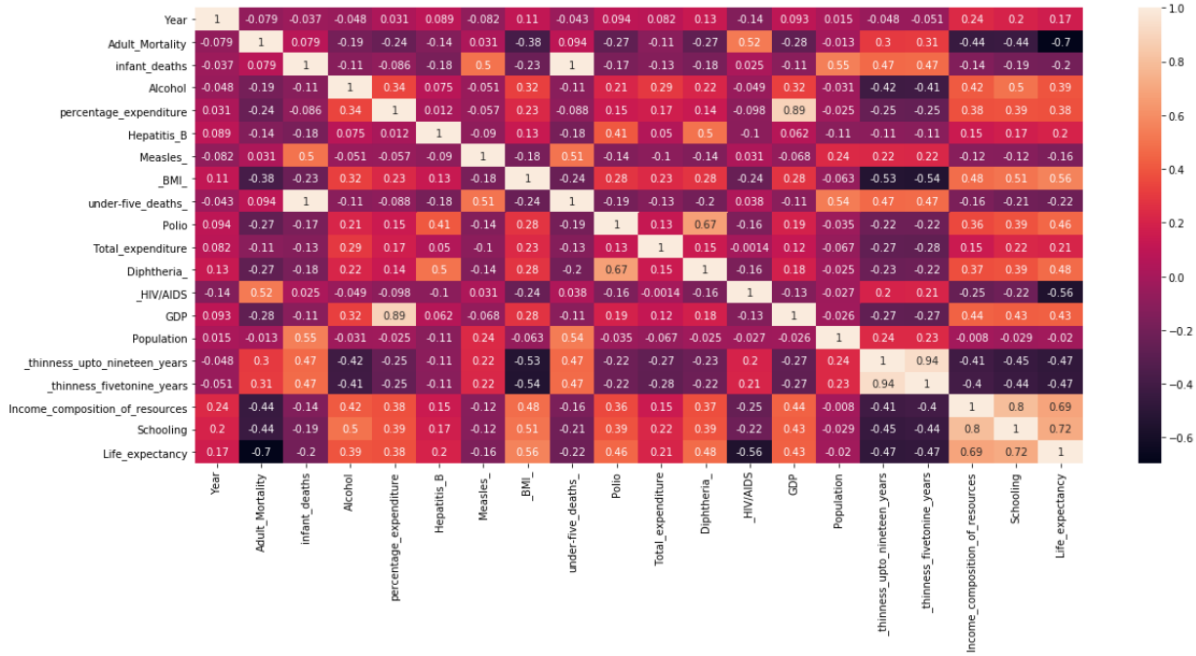
At the same time, superpower like China when launched its One-child policy didn't know the result that would have on the future generation ratio of the people. Due to that, even though China has the largest population in the world, there are more older adults as compared to the young ones.

LITERATURE SURVEY

This survey has been done to have a better insight to the project and to do a profound study of the problems

EXISTING PROBLEM

We can analyse the data in a better way with the help of a heatmap.



Here, from the heatmap, we can derive that the attributes (in decreasing order of effect on life expectancy) such as:

- Schooling
- Income composition of resources
- BMI
- Percentage of expenditure
- Alcohol
- GDP

All of these attributes represent a higher standard of living and better financial conditions that one can afford whether on an individual basis or country basis. Also, these attributes show that the higher are their values; the higher is the life expectancy of the people.

Whereas attributes such as:

- Adult Mortality
- HIV/AIDS
- Thinness up to 19 years
- Thinness up to 5-9 years

Attributes like this and other disease-related attributes cover the problems like:

- Malnutrition
- Epidemic/other widespread diseases
- Poverty
- Poor lifestyle
- Poor education

The conclusion we can draw from this is, better the standard of living in a country, better is the life expectancy. In the broad term, developed countries have a better life expectancy.

The problems listed above may arise due to the economic as well as political reasons, i.e. a country's status (developing or developed) has a significant impact on the life expectancy of its people. Also, the intractable factor of the people plays a vital role, i.e. the typical mentality, social taboos that people may derive from the previous generations builds their challenging future. People should have an open mindset towards change and should have the sense to differentiate between what's right and what's wrong for them.

PROPOSED SOLUTION

The solutions to this problem would be as follows:

- The government should make suitable policies and should intelligently organise financial schemes using the help of current technology.

- Special priority should be given to the health sector, and a lot of work should be done in monitoring and creation of the policies that will benefit the people.

- Countries should promote entrepreneurship and start-ups should be given a distinguished place among the society and the industry to encourage the manufacture and selling of own's goods in one individual's country, hence increasing the GDP.

- Particular medicines and vaccines should be made to make the public immune to various diseases.

- Other than policies and schemes, citizens should be encouraged to speak aloud about mental health, and people facing such difficulties should be given equal importance as compared to a physical inability faced by the people.

- People should be taught the ways to become emotionally intelligent to handle the stressful situations they are bound to have in life to increase the happiness factor in the people.

- To eradicate poverty, the best tool is education. So, to promote that, specialised NGOs or campaigns should be started as well as free schools should be opened that will give the students an advantage of being educated and stand equally with the rest of the world.

- Educated people should be encouraged to teach in schools, give free tuitions whenever they have the time to do so. Also, older people who have retired should teach poor children either in their home or nearby empty areas, as they can devote themselves entirely to teaching them, which would greatly benefit the children.

- For improving the conditions of the self-employed, business sector, the citizens should promote the use of homeland's products by which the money circulates in the country only which could improve the financial conditions gradually.

Solutions to this problem are endless as this problem deals both with physical as well as mental stability on an individual base. I have thus listed the answers from a broader perspective.

Another significant area we should look at is the three steps for any scheme or policy:

1. Creation
2. Execution
3. Monitoring

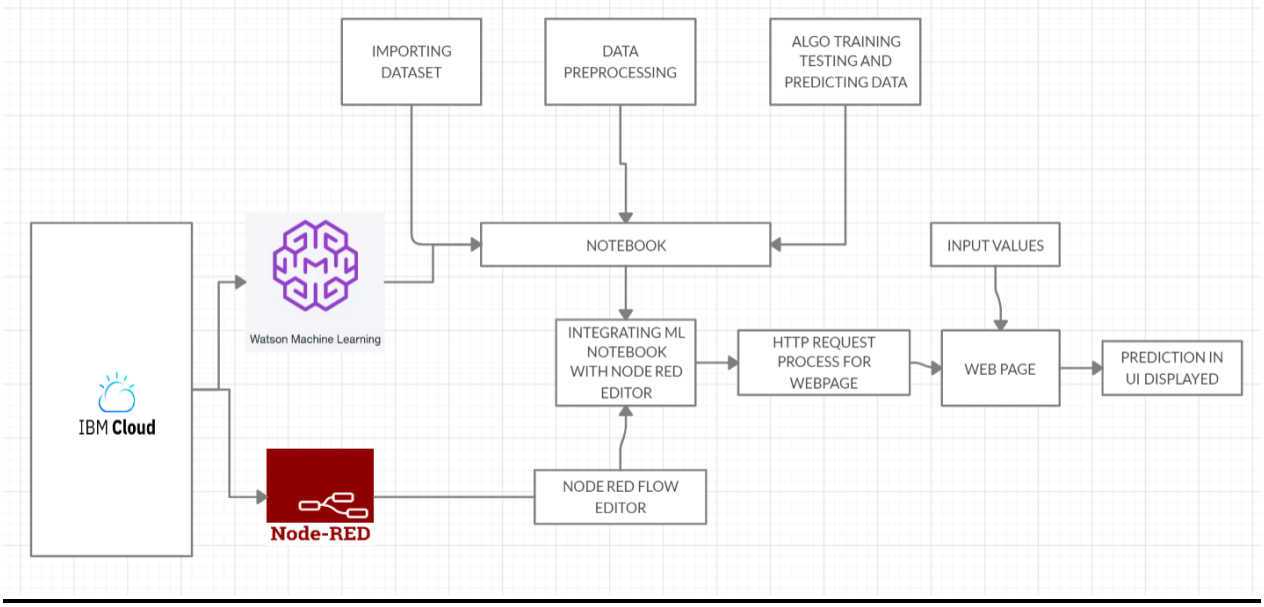
In this, monitoring of the schemes and the policies should be done regularly to ensure the proper functioning because that's the part where there is a lack in service. Thus, the plan fails to benefit the citizens.

This is a significant mistake that is being made by the government that, they always miss the point where they have to look after their policies and have to ensure their smooth functioning.

THEORETICAL ANALYSIS

This analysis explains the process of this project, and various tools used in it.

BLOCK DIAGRAM



HARDWARE/SOFTWARE DESIGNING

For the software design, the components used are:

1. IBM CLOUD PLATFORM
2. IBM WATSON STUDIO ML SERVICE
3. IBM NODE RED SERVICE

1. IBM CLOUD PLATFORM

The IBM cloud platform combines a platform as a service (PaaS) with infrastructure as a service (IaaS) to provide an integrated experience. The platform scales and supports both small development teams and organisations and large enterprise businesses. Globally deployed across data centres around the world, the solution we build on IBM Cloud™ spins up fast and performs reliably in a tested and supported environment we can trust.

The platform is built to support our needs, whether it is working only in the public cloud or taking advantage of a multi-cloud deployment model. With our open-source technologies, such as Kubernetes, Red Hat OpenShift, and a full range of compute options, including virtual

machines, containers, bare metal, and serverless, we have as much control and flexibility needed to support workloads in your hybrid environment. We can deploy cloud-native apps while also ensuring workload portability.

Whether we need to migrate apps to the cloud, modernise our existing apps by using cloud services, ensure data resiliency against regional failure, or leverage new paradigms and deployment topologies to innovate and build our cloud-native apps, the platform's open architecture is built to accommodate the use case.

What's built into the platform?

As the following diagram illustrates, the IBM Cloud platform is composed of multiple components that work together to provide a consistent and dependable cloud experience.

- A robust console that serves as the front end for creating, viewing, managing our cloud resources
- An Identity and access management component that securely authenticates users for both platform services and controls access to resources consistently across IBM Cloud
- A catalog that consists of hundreds of IBM Cloud offerings
- A search and tagging mechanism for filtering and identifying the resources
- An account and billing management system that provides exact usage for pricing plans and secure credit card fraud protection

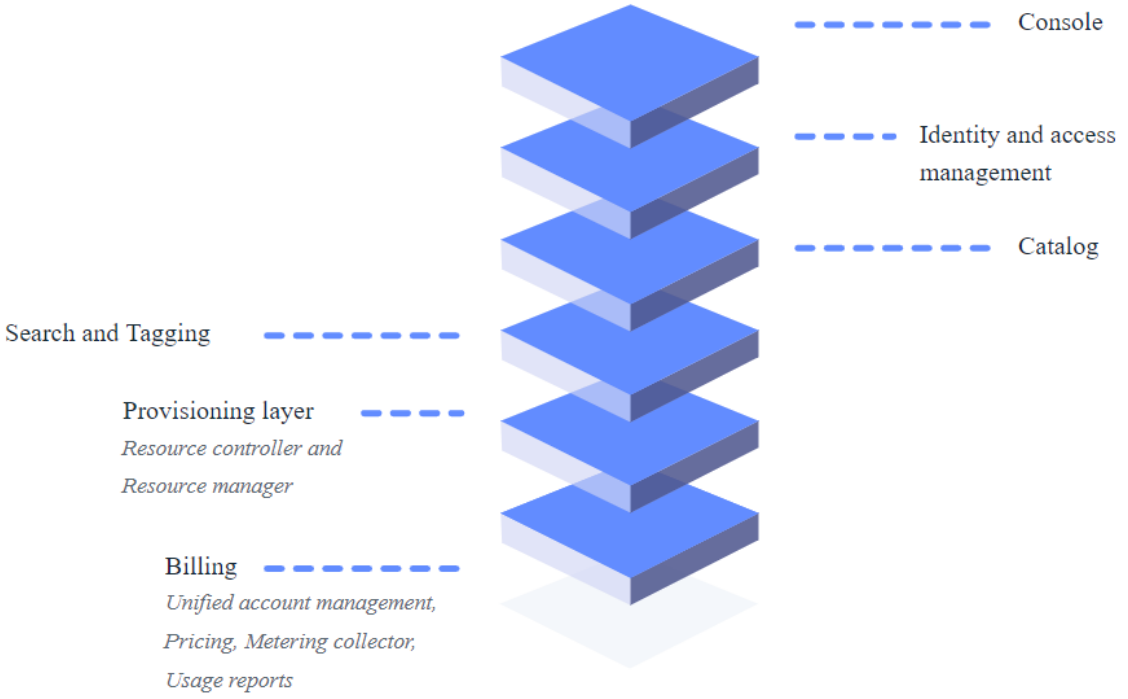


Figure 1. Components of the IBM Cloud platform

Whether there is an existing code that has to modernise and bring to the cloud or we're developing a brand-new application, our developers can tap into the rapidly growing ecosystem of available services and runtime frameworks in IBM Cloud.

2. IBM WATSON STUDIO

IBM Watson Studio is an integrated environment designed to make it easy to develop, train, manage models, and deploy AI-powered applications. It is a SaaS solution delivered on the IBM Cloud. It is an evolving Data Science Experience on IBM Cloud with a lot of new features to build AI applications

Watson Studio provides advanced new capabilities

With Watson Studio, we are doing the following:

- Extending the skills, it gives around deep learning, including TensorFlow scoring
- Allowing us to access pre-trained models from the Watson Services, such as Watson Visual Recognition
- Enabling us to bring in non-structured data
- Further automating and providing insight into model management
- Continuing to provide us with a choice of best-in-breed data science/ML tools
- Strengthening our drag-and-drop interface to build analytics models using SPSS Modeler
- Enabling us to visualise the insights with dynamic dashboards

Deploy and run AI models with Watson Machine Learning

IBM Watson Machine Learning helps data scientists and developers accelerate AI and machine-learning deployment. With its open, extensible model operation, Watson Machine Learning helps businesses simplify and harness AI at scale across any cloud. Watson Machine Learning provides capabilities to support:

- Deploy models built with IBM Watson Studio and open source tools.
- Dynamically retrain models
- Automatically generate APIs to build AI-powered applications
- Manage models through integration with IBM Watson Openscale
- Streamline model management and deployment end-to-end with an easy-to-use interface

Hardware:

A laptop with at least 4GB RAM

Software:

- **Jupyter** – Inbuilt in Watson ML service

- **Scientific Computation Library** – Pandas, NumPy

- **Visualisation Libraries** – Matplotlib, Seaborn

- **Algorithmic Libraries** – Scikit-Learn

- **Dependencies** – Data from internet

3. WATSON NODE RED SERVICE

Node-RED is a flow-based development tool for visual programming developed initially been by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things.

Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions. Elements of applications can be saved or shared for re-use. The runtime is built on Node.js. The flows generated in Node-RED are stored using JSON.

Flow-based Programming

Invented by J. Paul Morrison in the 1970s, flow-based programming is a way of describing an application's behaviour as a network of black-boxes, or "nodes" as they are called in Node-RED. Each Node has a well-defined purpose; it is given some data; it does something with that data, and then it passes that data on. The network is responsible for the flow of data between the nodes.

The light-weight runtime is built on Node.js, taking full advantage of its event-driven, non-blocking model. This makes it ideal to run at the edge of the network on low-cost hardware such as the Raspberry Pi as well as in the cloud. With over 225,000 modules in Node's package repository, it is easy to extend the range of palette nodes to add new capabilities.

It is a model that lends itself very well to a visual representation and makes it more accessible to a broader range of users. If someone can break down a problem into discrete steps, they can look at a flow and get a sense of what it is doing; without having to understand the individual lines of code within each Node.

Runtime/Editor

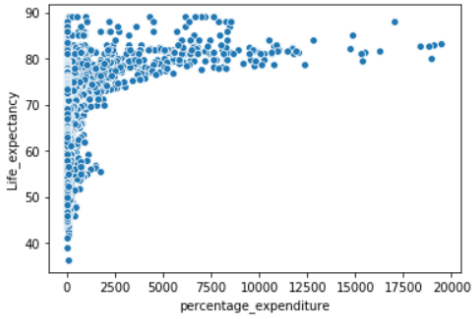
Node-RED consists of a Node.js based runtime that we point a web browser at to access the flow editor. Within the browser, we create your application by dragging nodes from the palette into a workspace and start to wire them together. With a single click, the app is deployed back to the runtime where it is run.

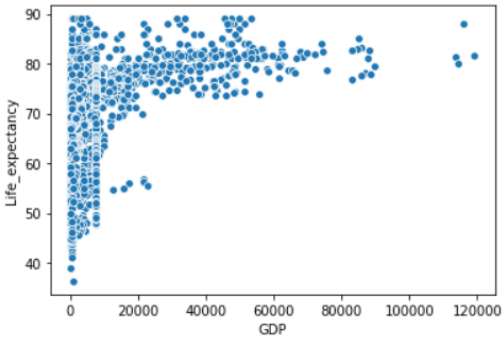
The palette of nodes can be easily extended by installing new nodes created by the community, and the flows you create can be easily shared as JSON files.

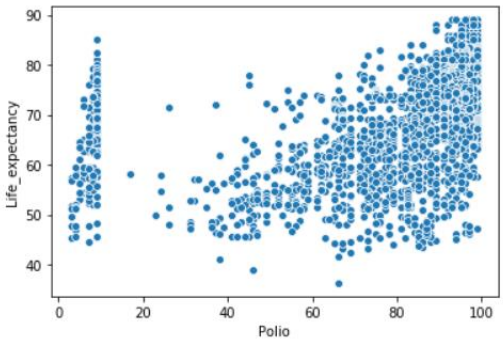
Social Development

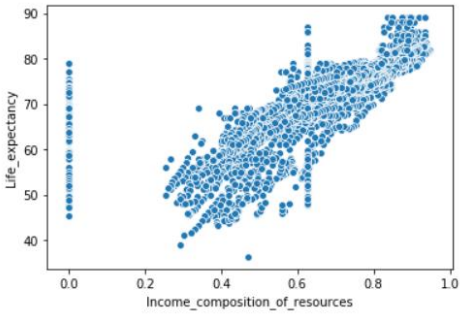
The flows created in Node-RED are stored using JSON, which can be easily imported and exported for sharing with others. An online flow library allows us to share our best flows with the world

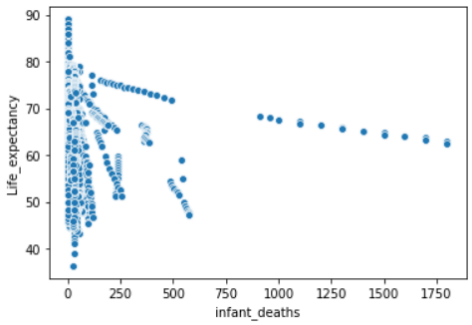
EXPERIMENTAL INVESTIGATIONS

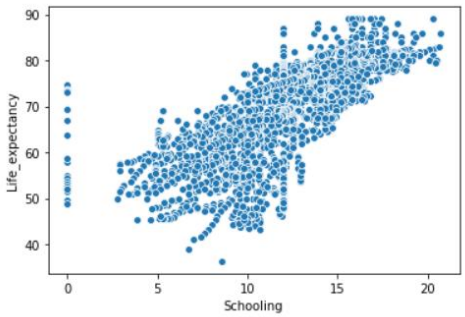


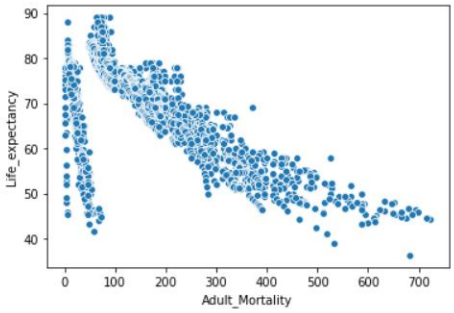


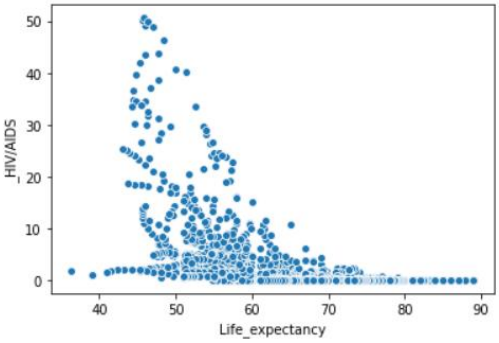


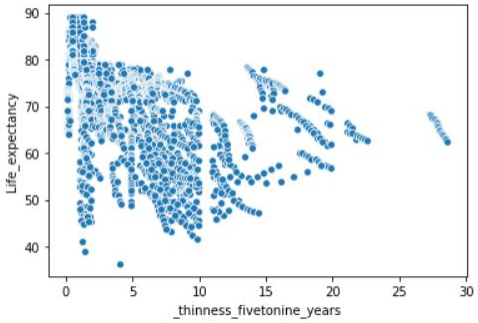


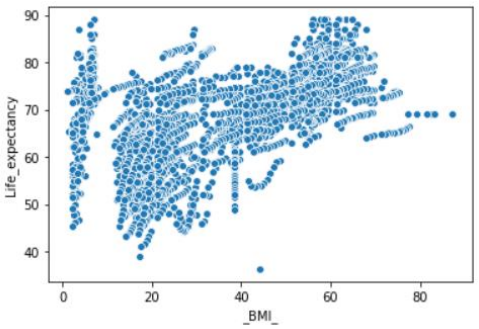












From the above graphs, we can derive that,

1. The health of the people is directly proportional to the life expectancy of them.
2. The epidemic and other chronic diseases that are given above like HIV/AIDS, Polio affect the health of the people crucially and can reduce the life span of the person drastically.
3. All the attributes, the benefits of which only the financially sound people can afford also tells us a lot. The characteristics are GDP, BMI, Schooling, expenditure, Income composition of resources etc. The life expectancy of such people is higher than the ones who live in poverty.
4. It is probably because wealthy people can take better care of themselves as compared to the poor ones.
5. In the context of the above statement, the location where the poor people live are not equipped with facilities; the wealthier have access to.
6. The rich countries, which have higher GDP, have a high life expectancy. If calculated, the happiness factor would also stand out of the impoverished country's.
7. It also shows that Emotional Quotient is just as important as the financial coefficient is.

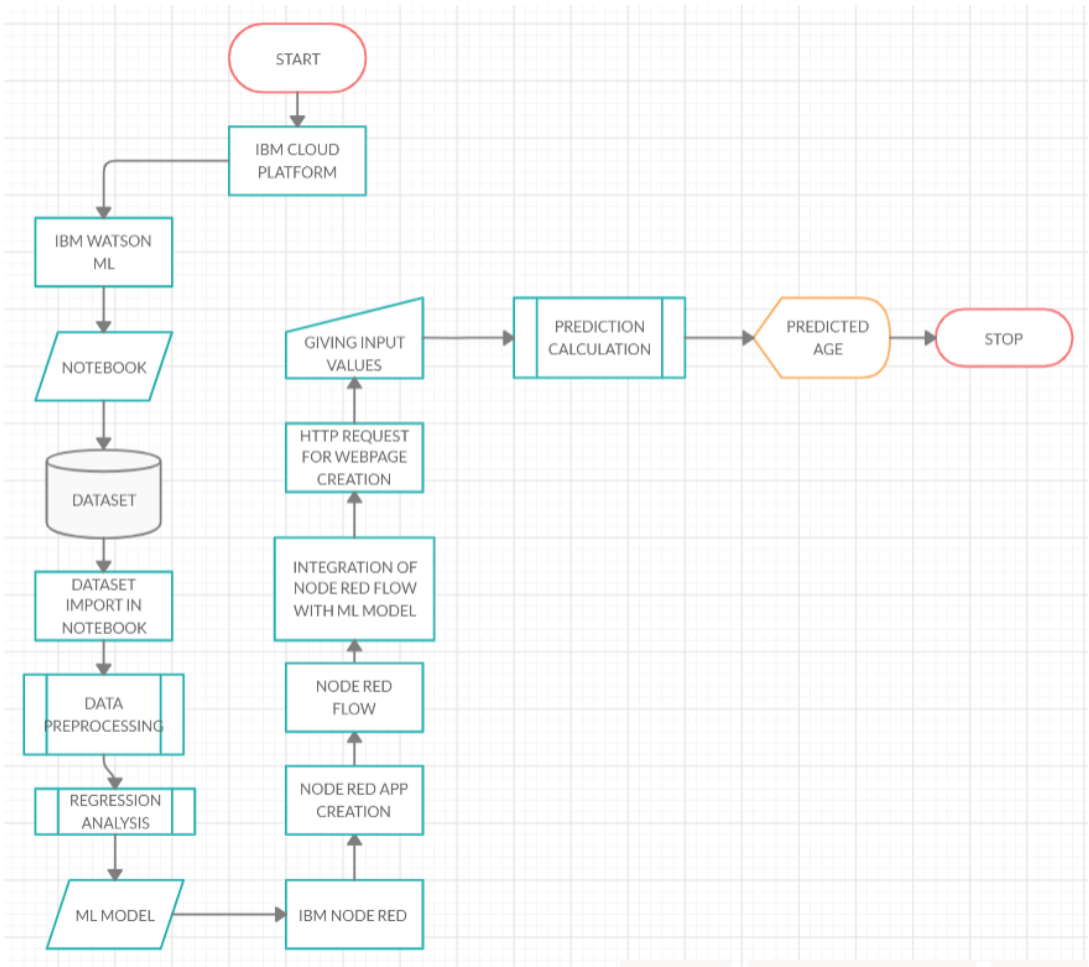
Regression Analysis

During regression analysis, it was found that linear regression produced the results with 81% accuracy. This could be improved by either scaling the data or by selecting another algorithm. Scaling of the data further was providing results with predictions having 83% accuracy.

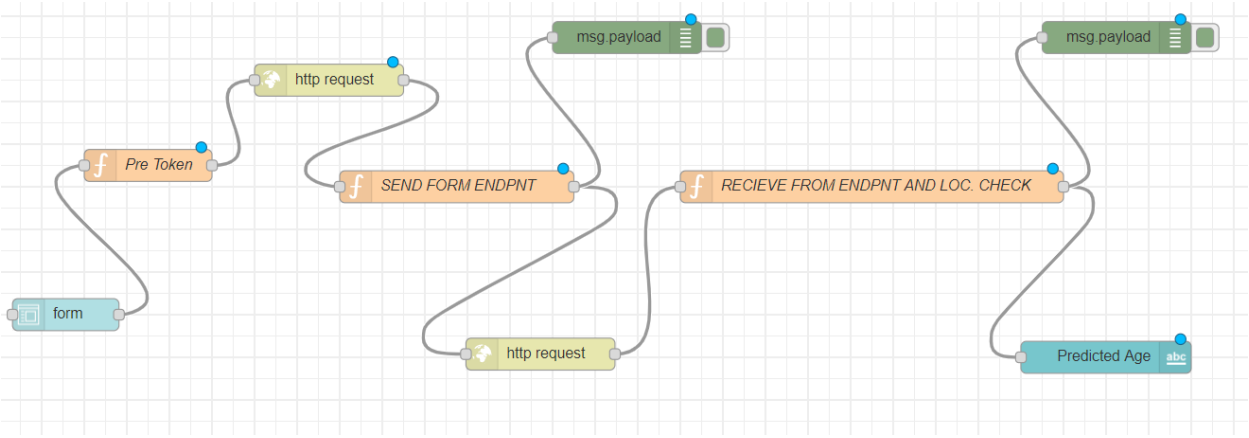
Then random forest regression was applied as it was better than the decision tree algorithm and overcame its shortcomings.

The Random Forest algorithm provided promising results with 95% accuracy. The algorithm could've been overfitting with the data, but the testing data prediction proved the thought to be wrong. Hence Random Forrest was used.

FLOWCHART

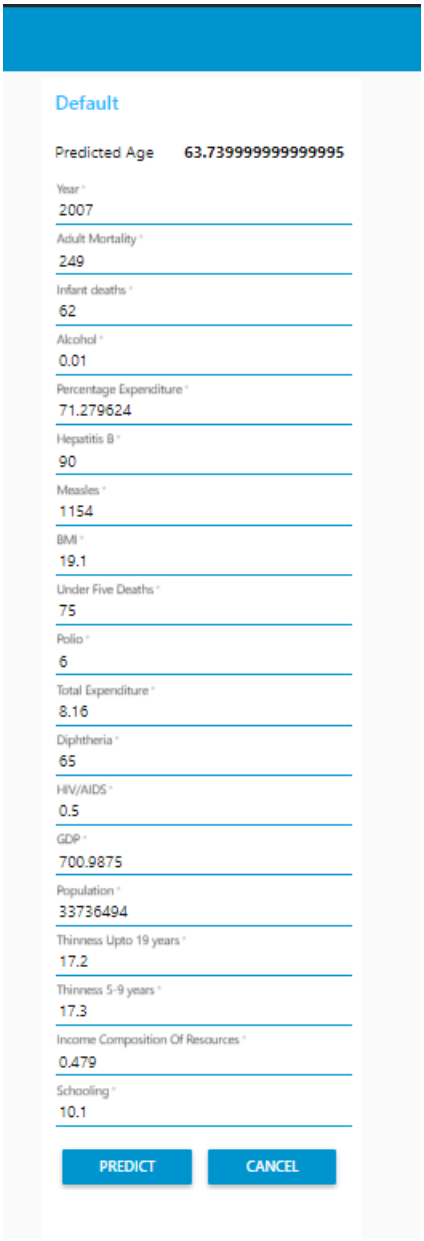


NODE RED FLOW



RESULT

The values had to be input in the demanded attributes by the user, and the predicted age was displayed then.



ADVANTAGES AND DISADVANTAGES

ADVANTAGES

Some advantages of this project are:

1. The prediction of life expectancy of a country or an individual gives them a reality check about their health and where it would end up with the current lifestyle.

2. Life expectancy prediction allows the government to look upon their people in a better way.

3. International organisations like WHO can help the government accordingly and can suggest better suggestions to the government

4. Researchers can work on this data and analyse the health trend of the people from a particular period to another. This would be reported to media, and they would make people realise what they have been doing wrong

DISADVANTAGES

1. It is a prediction, and even if it is somewhat accurate, it cannot tell about the mishaps that may happen In the future like an accident or of any fatal conditions like coronavirus, tumour etc.

2. It does not cover the local areas of a region or a country where there may be variations. Eg, the people living in slums may not have a higher life expectancy as compared to those living in cities.

3. It doesn't cover a patient's genetic disorder history in the prediction. It thus cannot predict if the patient is going to have any genetic disorders in the future or not.

APPLICATIONS

1. Predicting life expectancy helps the patients in **Advance Care Planning (ACP**). ACP is a process during which patients make decisions about the health care they wish to receive in the future, in case the patient loses the capacity of making decisions or communicating about them. The doctors can thus present the option of having a surgery that wouldn't be possible in the calculated future, or doctors could use this to prevent excessive treatment.

2. **For government**, they can analyse the prediction to **plan their policies and schemes** etc. which could be beneficial soon in the future, given the projections. Suppose a policy may be helpful for the people after ten years given the health conditions of the people, so the government won't launch the policy now and would wait for the time when it would benefit the most.

3. **The insurance companies** could use life expectancy predictions in choosing a better premium plan or Mediclaim for them using this.

4. This could give the people a **reality check about their health,** and they could act upon it. By this, the mortality rate would go down, increasing the life expectancy.

5. Life expectancy prediction can be used by the **actuarial sciences** to do some research, and they can include this in their research topics or case studies etc.

CONCLUSION

By the application of random forest regression, I conclude that according to the given dataset, the features that affect the life expectancy the most are Adult mortality rate, Percentage expenditure, Polio, Measles, GDP, HIV/AIDS, Schooling, Income composition, BMI, and Alcohol consumption rate. Application of random forest regression to the dataset has produced successful results in predicting the life expectancy of the population in the world. The accuracy score has been good, which indicates that the predicted outcome is almost accurate. Further, the application of the linear regression generates results which were less reliable than the results produced by the random forest regression model. Thus, random forest regression is a better and more accurate model that produces better results and almost precise prediction.
It may be possible than on the world basis, random forest regression exceptionally well, but on a country basis, linear regression may produce better results. For that, further research would be required to examine a particular country. For that, the dataset may also be according to that country. But, I'm not going that far as my project doesn't require that.

FUTURE SCOPE

The life expectancy prediction surely provides promising results. With more technologies like Data Science, AI, better machine learning algorithms, this model could give better results.

Furthermore, the technologies in the future may provide an exact result which could be a prediction to set in stone.

For that our dataset should be reliable, i.e. more attributes, detailed report of the country would be required. I identified that the collection of data would be a massive challenge due to the privacy and government policy considerations, which will require the collaboration of various bodies in the health industry.

As the proposed solution requires processing and transmitting health information of users, information security is a crucial aspect to consider, such as privacy as well as ethical requirements recommenced by regulation bodies.

The scope of security and ethical requirements need to be clearly defined and specified for future work as challenges are expected to build a centralised database with the incorporation of health networks.

It would take a high level of understanding between countries to work hand in hand and move towards technological advancements.

Other than that, the future scope of this tech is unfathomable, and there may be new technology in the field of AI which would require the life expectancy of a person to its accurate value so as to help the individual lead a healthy life.

BIBLIOGRAPHY

https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-019-0775-2

https://www.ijitee.org/wp-content/uploads/papers/v8i10/J91560881019.pdf

https://developer.ibm.com/blogs/top-5-reasons-to-use-node-red-right-now/

https://www.ibm.com/in-en/cloud/machine-learning

https://developer.ibm.com/components/node-red/gettingstarted/

https://www.ibm.com/support/knowledgecenter/SS3PWM_1.0.0/wsj/wmls/overview.html

https://app.creately.com/manage/project/home

https://www.thinkadvisor.com/2016/05/27/9-factors-that-affect-longevity/

https://www.ons.gov.uk/peoplepopulationandcommunity/healthandsocialcare/healthandlifeexpectancies/articles/whatismylifeexpectancyandhowmightitchange/2017-12-01

https://medium.com/swlh/predicting-life-expectancy-w-regression-b794ca457cd4

APPENDIX

SOURCE CODE

//importing jupyter notebook in Watson ml

```
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 noteb
ook.
client_29e2eabbac3440ddb072699d667dbf66 = ibm_boto3.client(service_name
='s3',
    ibm_api_key_id='o9s7J6wEFN_PKpqN0WhaRY6a_Eey7tqi8eav49WWjpBu',
    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_29e2eabbac3440ddb072699d667dbf66.get_object(Bucket='lifee
xpectancy-donotdelete-pr-7qdlgyd7j70d8j',Key='Life_Expectancy_Data.csv'
)['Body']
# add missing __iter__ method, so pandas accepts body as file-like obje
ct
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __i
ter__, body )

data = pd.read_csv(body)
data.head()
```

//importing libraries

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
# replacing the column spaces with string acc to jupyter config
data.columns = data.columns.str.replace(' ','_')

shifting target to end of the table
life_ex = data['Life_expectancy_']
del data['Life_expectancy_']
data['Life_expectancy'] = life_ex
```

// data cleaning

```
data.describe()
```

Out[12]:

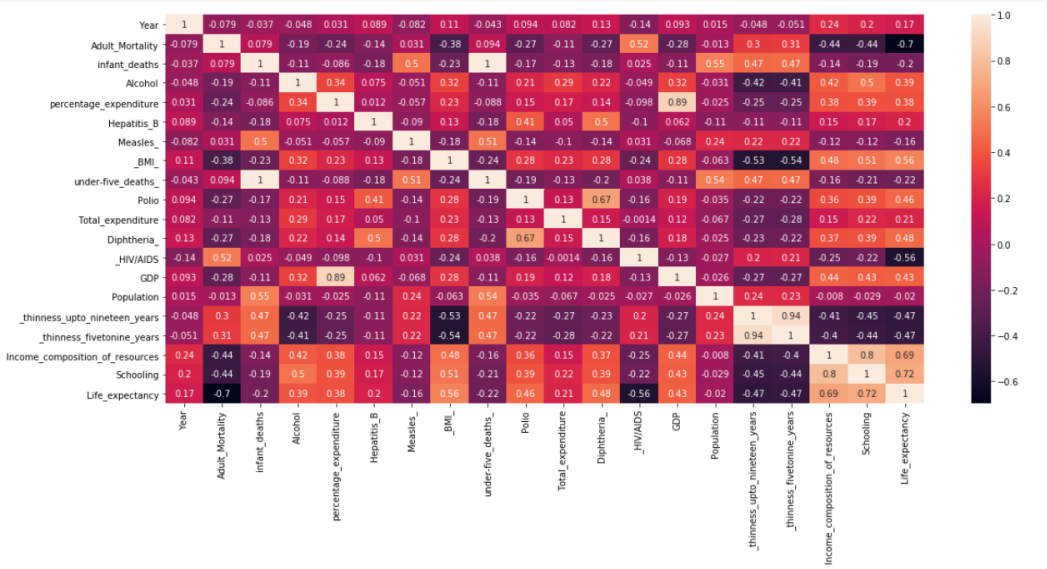
	Year	Adult_Mortality	infant_deaths	Alcohol	percentage_expenditure	Hepatitis_B	Measles_	_BMI_	under-five_deaths_
count	2938.000000	2928.000000	2938.000000	2744.000000	2938.000000	2385.000000	2938.000000	2904.000000	2938.000000
mean	2007.518720	164.796448	30.303948	4.602861	738.251295	80.940461	2419.592240	38.321247	42.035739
std	4.613841	124.292079	117.926501	4.052413	1987.914858	25.070016	11467.272489	20.044034	160.445548
min	2000.000000	1.000000	0.000000	0.010000	0.000000	1.000000	0.000000	1.000000	0.000000
25%	2004.000000	74.000000	0.000000	0.877500	4.685343	77.000000	0.000000	19.300000	0.000000
50%	2008.000000	144.000000	3.000000	3.755000	64.912906	92.000000	17.000000	43.500000	4.000000
75%	2012.000000	228.000000	22.000000	7.702500	441.534144	97.000000	360.250000	56.200000	28.000000
max	2015.000000	723.000000	1800.000000	17.870000	19479.911610	99.000000	212183.000000	87.300000	2500.000000

```
data.isnull().sum()
```

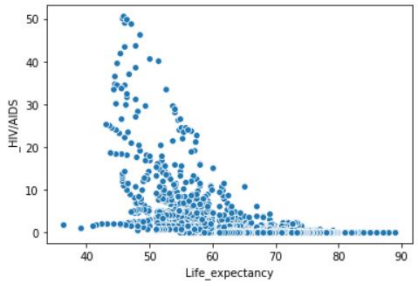
Out[13]:
```
Country                          0
Year                             0
Status                           0
Adult_Mortality                 10
infant_deaths                    0
Alcohol                        194
percentage_expenditure           0
Hepatitis_B                    553
Measles_                         0
_BMI_                           34
under-five_deaths_               0
Polio                           19
Total_expenditure              226
Diphtheria_                     19
_HIV/AIDS                        0
GDP                            448
Population                     652
_thinness__1-19_years           34
_thinness_5-9_years             34
Income_composition_of_resources    167
Schooling                      163
Life_expectancy                 10
dtype: int64
```

```
data.Adult_Mortality.fillna(data.Adult_Mortality.mean(),inplace = True)
data.Alcohol.fillna(data.Alcohol.mean(),inplace = True)
data.Hepatitis_B.fillna(data.Hepatitis_B.mean(),inplace = True)
data._BMI_.fillna(data._BMI_.mean(),inplace = True)
data.Polio.fillna(data.Polio.mean(),inplace = True)
data.Total_expenditure.fillna(data.Total_expenditure.mean(),inplace = T
rue)
data.Diphtheria_.fillna(data.Diphtheria_.mean(),inplace = True)
data.GDP.fillna(data.GDP.mean(),inplace = True)
data.Population.fillna(data.Population.mean(),inplace = True)
data.rename(columns = {'_thinness__1-19_years':'_thinness_upto_nineteen
_years','_thinness_5-9_years':'_thinness_fivetonine_years'},
inplace = True)
data._thinness_upto_nineteen_years.fillna(data._thinness_upto_nineteen_
years.mean(),inplace = True)
data._thinness_fivetonine_years.fillna(data._thinness_fivetonine_years.
mean(),inplace = True)
data.Income_composition_of_resources.fillna(data.Income_composition_of_
resources.mean(),inplace = True)
data.Schooling.fillna(data.Schooling.mean(),inplace = True)
data.Life_expectancy.fillna(data.Life_expectancy.mean(),inplace = True)
```

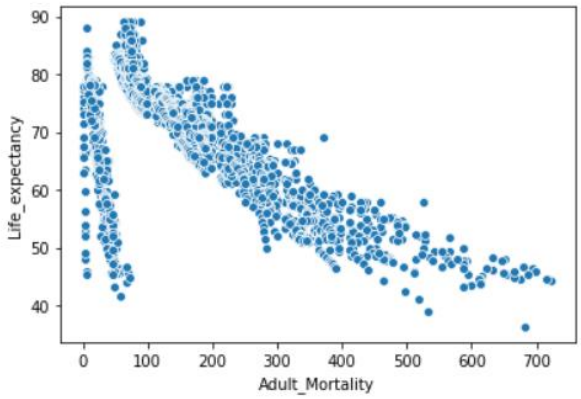
```
data.shape


//starting to analyse the data manually using heatmap
plt.figure(figsize = (19, 8))
sns.heatmap(data.corr(),annot = True)
```



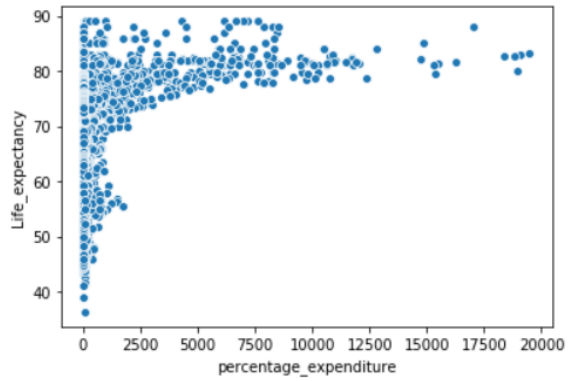

```
sns.scatterplot(y = data["_HIV/AIDS"],x = data["Life_expectancy"])
```



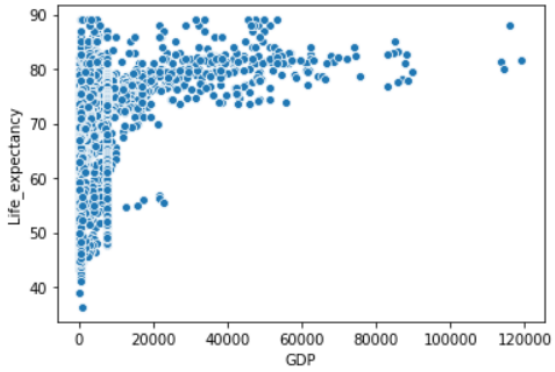

```
sns.scatterplot(x = d                                    ata["Adult_Mortality"]
, y = data["Life_expectancy"])
```



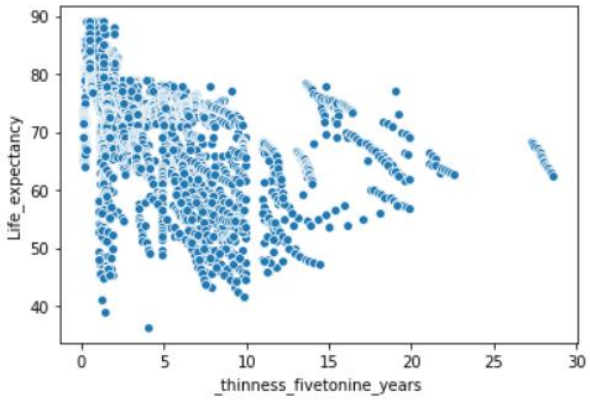

```
sns.scatterplot(x=data["percentage_expenditure"],y=data["Life_expectanc
y"])
```



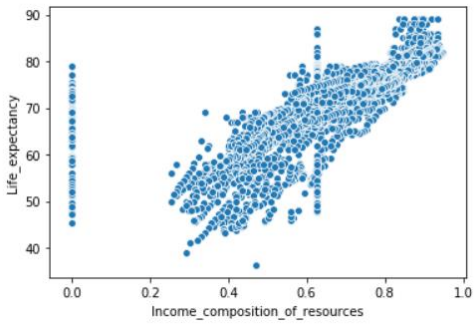

```
sns.scatterplot(x=data["GDP"],y=data["Life_expectancy"])
```



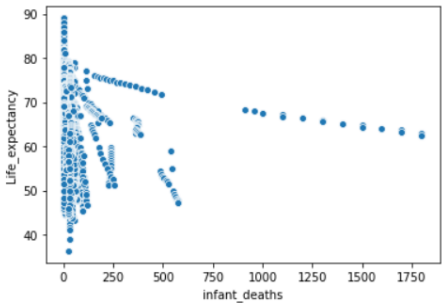

```
sns.scatterplot(x=data["_thinness_fivetonine_years"],y=data["Life_expectancy"])
```



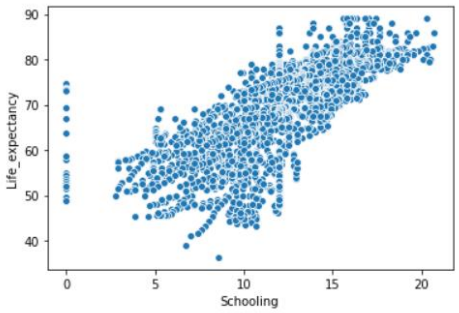

```
sns.scatterplot(x=data["Income_composition_of_resources"],y=data["Life_expectancy"])
```



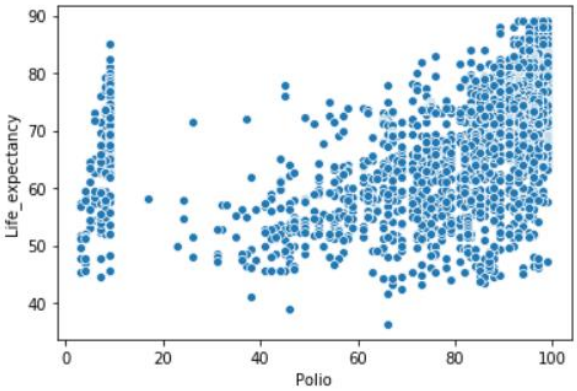

```
sns.scatterplot(x=data["infant_deaths"],y=data["Life_expectancy"])
```



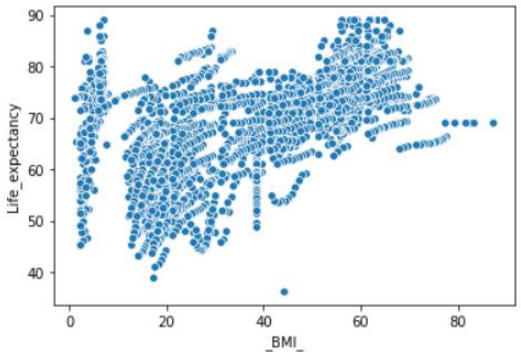

```
sns.scatterplot(x=data["Schooling"],y=data["Life_expectancy"])
```




```
sns.scatterplot(x=data["Polio"],y=data["Life_expectancy"])
```




```
sns.scatterplot(x=data["_BMI_"],y=data["Life_expectancy"])
```




```
X = data.iloc[:,0:21]
del X['Country']
del X['Status']
Y = data.iloc[:,21:22]


//splitting data
from sklearn import model_selection
X_train, X_test, Y_train, Y_test = model_selection.train_test_split(X,
Y)

//calling on random forest regressor
from sklearn.ensemble import RandomForestRegressor
regressor = RandomForestRegressor(n_estimators = 10, random_state = 0)
regressor.fit(X_train,Y_train)
```

```
Out[124]: RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=None,
                      max_features='auto', max_leaf_nodes=None,
                      min_impurity_decrease=0.0, min_impurity_split=None,
                      min_samples_leaf=1, min_samples_split=2,
                      min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=None,
                      oob_score=False, random_state=0, verbose=0, warm_start=False)
```

```
ytrain = np.array(Y_train)
training_y = ytrain.ravel()
Y_pred = regressor.predict(X_test)
test_y = np.array(Y_test)
ytest = test_y.ravel()
def score(y_truth, y_pred):
    u = ((y_truth - y_pred)**2).sum()
    v = ((y_truth - y_truth.mean())**2).sum()
    return 1 - u/v
print(score(ytest,Y_pred))
```

```
0.9453240021754661
```

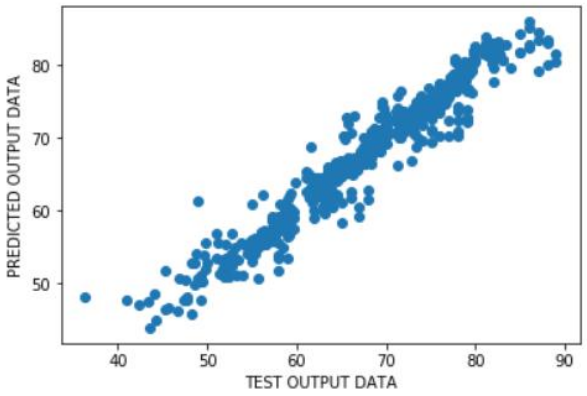
```
from sklearn import metrics
metrics.mean_squared_error(ytest,Y_pred)
```

```
Out[135]: 4.683501351728435
```

```
np.sqrt(metrics.mean_squared_error(ytest,Y_pred))
```

```
Out[136]: 2.1641398641789387
```

```
import matplotlib.pyplot as plt
plt.xlabel("TEST OUTPUT DATA")
plt.ylabel("PREDICTED OUTPUT DATA")
plt.scatter(ytest,Y_pred)
plt.show()
```



// create an endpoint for the model to be used in NODE RED

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

```
In [139]: !pip install watson-machine-learning-client

Requirement already satisfied: watson-machine-learning-client in /opt/conda/envs/Python36/lib/python3.6/site-packages (1.0.376)
Requirement already satisfied: tqdm in /opt/conda/envs/Python36/lib/python3.6/site-packages (from watson-machine-learning-clien
t) (4.31.1)
Requirement already satisfied: requests in /opt/conda/envs/Python36/lib/python3.6/site-packages (from watson-machine-learning-c
lient) (2.21.0)
Requirement already satisfied: lomond in /opt/conda/envs/Python36/lib/python3.6/site-packages (from watson-machine-learning-cli
ent) (0.3.3)
Requirement already satisfied: pandas in /opt/conda/envs/Python36/lib/python3.6/site-packages (from watson-machine-learning-cli
ent) (0.24.1)
Requirement already satisfied: ibm-cos-sdk in /opt/conda/envs/Python36/lib/python3.6/site-packages (from watson-machine-learnin
g-client) (2.4.3)
Requirement already satisfied: certifi in /opt/conda/envs/Python36/lib/python3.6/site-packages (from watson-machine-learning-cl
ient) (2020.4.5.1)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python36/lib/python3.6/site-packages (from watson-machine-learning-cl
ient) (1.24.1)
Requirement already satisfied: tabulate in /opt/conda/envs/Python36/lib/python3.6/site-packages (from watson-machine-learning-c
lient) (0.8.2)
Requirement already satisfied: idna<2.9,>=2.5 in /opt/conda/envs/Python36/lib/python3.6/site-packages (from requests->watson-ma
chine-learning-client) (2.8)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /opt/conda/envs/Python36/lib/python3.6/site-packages (from requests->wa
tson-machine-learning-client) (3.0.4)
Requirement already satisfied: six>=1.10.0 in /opt/conda/envs/Python36/lib/python3.6/site-packages (from lomond->watson-machine
-learning-client) (1.12.0)
Requirement already satisfied: pytz>=2011k in /opt/conda/envs/Python36/lib/python3.6/site-packages (from pandas->watson-machine
-learning-client) (2018.9)
Requirement already satisfied: numpy>=1.12.0 in /opt/conda/envs/Python36/lib/python3.6/site-packages (from pandas->watson-machi
ne-learning-client) (1.15.4)
Requirement already satisfied: python-dateutil>=2.5.0 in /opt/conda/envs/Python36/lib/python3.6/site-packages (from pandas->wat
son-machine-learning-client) (2.7.5)
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.*,>=2.0.0 in /opt/conda/envs/Python36/lib/python3.6/site-packages (fro
m ibm-cos-sdk->watson-machine-learning-client) (2.4.3)
Requirement already satisfied: ibm-cos-sdk-core==2.*,>=2.0.0 in /opt/conda/envs/Python36/lib/python3.6/site-packages (from ibm-
cos-sdk->watson-machine-learning-client) (2.4.3)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python36/lib/python3.6/site-packages (from ibm-cos-sdk
-core==2.*,>=2.0.0->ibm-cos-sdk->watson-machine-learning-client) (0.9.3)
Requirement already satisfied: docutils>=0.10 in /opt/conda/envs/Python36/lib/python3.6/site-packages (from ibm-cos-sdk-core==
```

```
from watson_machine_learning_client import WatsonMachineLearningAPIClie
nt
wml_credentials ={
  "apikey": "wshdFPikiwX3YjQ2K0kFAJ_qzSl6NVm3y4gWYv9Qrh_D",
  "instance_id": "a3c08f0a-15aa-462f-9411-825f00d713d0",
  "url": "https://eu-gb.ml.cloud.ibm.com"
}

client = WatsonMachineLearningAPIClient( wml_credentials )

model_props = {client.repository.ModelMetaNames.AUTHOR_NAME: "Chaitanya
", client.repository.ModelMetaNames.AUTHOR_EMAIL: "SI05202000905@smarti
nternz.com" ,
               client.repository.ModelMetaNames.NAME: "LIFE_EXPECTANCY_P
REDICTION"}

model_artifact = client.repository.store_model(regressor,meta_props = m
odel_props)

published_model_uid = client.repository.get_model_uid(model_artifact)

deployment = client.deployments.create(published_model_uid ,name = "LIF
E_EXPECTANCY_PREDICTION")
```

```
#######################################################################################

Synchronous deployment creation for uid: 'abca4cf4-7dc6-4f2a-8f49-79907cffbfac' started

#######################################################################################


INITIALIZING
DEPLOY_SUCCESS


------------------------------------------------------------------------------------------------
Successfully finished deployment creation, deployment_uid='3b04508f-aefb-4c22-9f5c-26ee0edff552'
------------------------------------------------------------------------------------------------
```

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

Out[150]: 'https://eu-gb.ml.cloud.ibm.com/v3/wml_instances/a3c08f0a-15aa-462f-9411-825f00d713d0/deployments/3b04508f-aefb-4c22-9f5c-26ee0edff552/online'