



Summer Internship Report

Predicting Life Expectancy Using Machine Learning

19th May 2020 – 17th June 2020

By:

Robin Rodrigues

From:

SmartBridge Educational
Services Pvt Ltd. Plot No 132,
Above DCB bank, 2nd floor,
Bapuji Nagar, Habsiguda,
Nacharam Main Road, Hyderabad – 500076

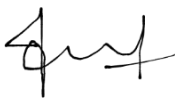
Date: 21/05/2020.

Dear **Robin Rodrigues**

SmartBridge Educational Services Pvt Ltd, is pleased to offer a training cum internship opportunity. During this period you would be associated with our mentors and The Smart Practice School Platform.

For further details you can contact us on +91 8499004200.

Thanks and Regards,



Ch. Jaya Prakash
Program Manager
– SIP2020, Date:
16/05/2020.

Index

1	INTRODUCTION
	1.1 Overview
	1.2 Purpose
2	LITERATURE SURVEY
	2.1 Existing problem
	2.2 Proposed solution
3	THEORITICAL ANALYSIS
	3.1 Block diagram
	3.2 Hardware / Software designing
4	EXPERIMENTAL INVESTIGATIONS
5	FLOWCHART
6	RESULT
7	ADVANTAGES & DISADVANTAGES
8	APPLICATIONS
9	CONCLUSION
10	FUTURE SCOPE
11	BIBILOGRAPHY
	APPENDIX
	A. Source code

1. Introduction

1.1 Overview

Problem Statement: Predicting Life Expectancy Using Machine Learning

Problem Description:

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.

Life expectancy refers to the number of years a person is expected to live based on the statistical average. Life expectancy varies by geographical area and by era. Since 1900 the global average life expectancy has more than doubled and is now above 70 years. The inequality of life expectancy is still very large across and within countries. In 2019 the country with the lowest life expectancy is the Central African Republic with 53 years, in Japan life expectancy is 30 years longer. Since, life expectancy differs country-wise by such a huge margin, it's necessary to prepare some prediction model which will help the countries with low life expectancy to improve the factors which affect their life expectancy rate.

1.2 Purpose

To construct a typical regression model that 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. The purpose of the project is to help a particular country know its average life expectancy, which negative factors affect life expectancy by a huge margin and hence take appropriate steps against those factors to improve life expectancy.

2. Literature Survey

The dataset for this project is taken from Kaggle. Dataset name is Life Expectancy (WHO) (<https://www.kaggle.com/kumarajarshi/life-expectancy-who>).

2.1 Existing Problem

Each country should have a prediction model so that they can understand which factors affects more in decreasing life expectancy of their country and hence take appropriate decision to increase life expectancy of human being in their country.

2.2 Proposed Solution

This project takes following aspects (features) as input:

1. Country
2. Year
3. Status
4. Life Expectancy
5. Adult Mortality
6. Alcohol
7. Percentage Expenditure
8. Hepatitis B
9. Measles
10. BMI
11. Under-five deaths
12. Polio
13. Total Expenditure
14. Diphtheria
15. HIV/AIDS
16. GDP
17. Population
18. Thinness 1-19 years
19. Thinness 5-9 years
20. Income composition of resources
21. Schooling

Target is Life Expectancy, measured in number of years.

The assumptions are:

1. 1. These are country level average
2. 2. There is no distinction between male and female.

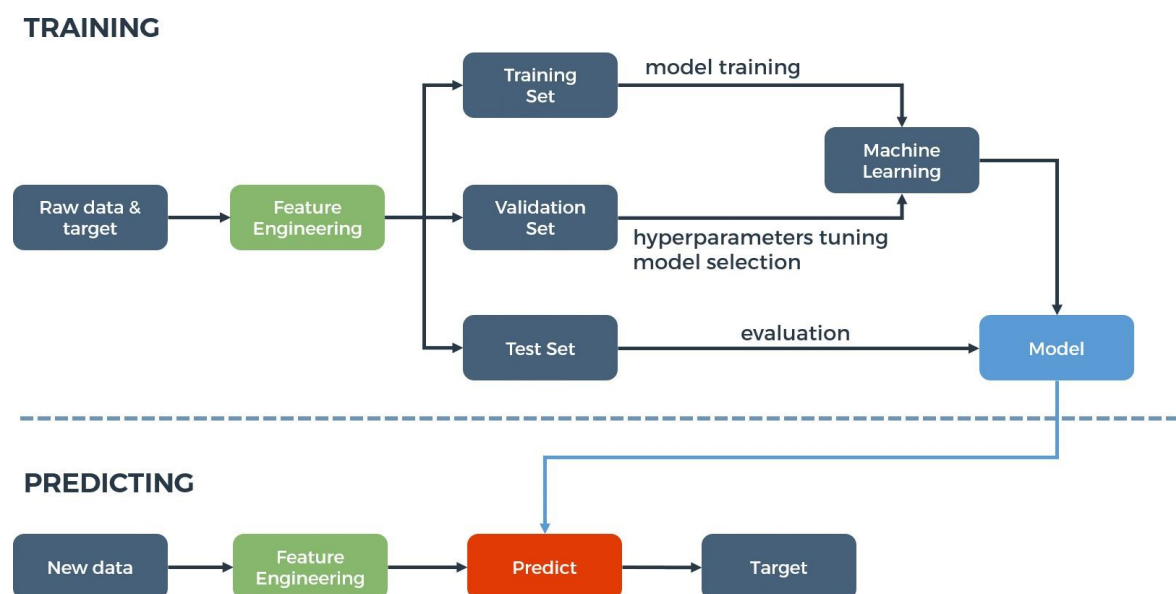
Among these input features I have converted BMI from numerical to ordinal for better accuracy. Being numerical, the model may sometimes fail to capture the relationship of values which are on the border between any two BMI categories. Converting this feature into Ordinal works better as it will be easy for the model to distinguish the numbers into categories.

So, among all the input features Country, Status and BMI are categorical and remaining all features are numeric. The NULL values are being replaced by their modes for the categorical columns and by their means for the numeric columns. There are a few outliers for some columns but they have not been removed or modified since it doesn't affect the accuracy of our model that much. I have experimented on Linear Regression model, Decision Tree Regression model and Random Forest Regression model and the best accuracy was found with Random Forest Regression model. I have done this on IBM Watson Studio.

Also IBM cloud has provided a great feature of Auto AI which optimizes the model to the best possible accuracy by using Hyper Parameter optimization and Extra Tree Regressor.

3. Theoretical Analysis

3.1 Block Diagram



3.2 Hardware / Software designing

Hardware :

- Processor : i3 7th gen or more
- Speed : 2 GHz or more
- Hard Disk Space : 10 GB or more
- Ram Memory : 4 GB or more

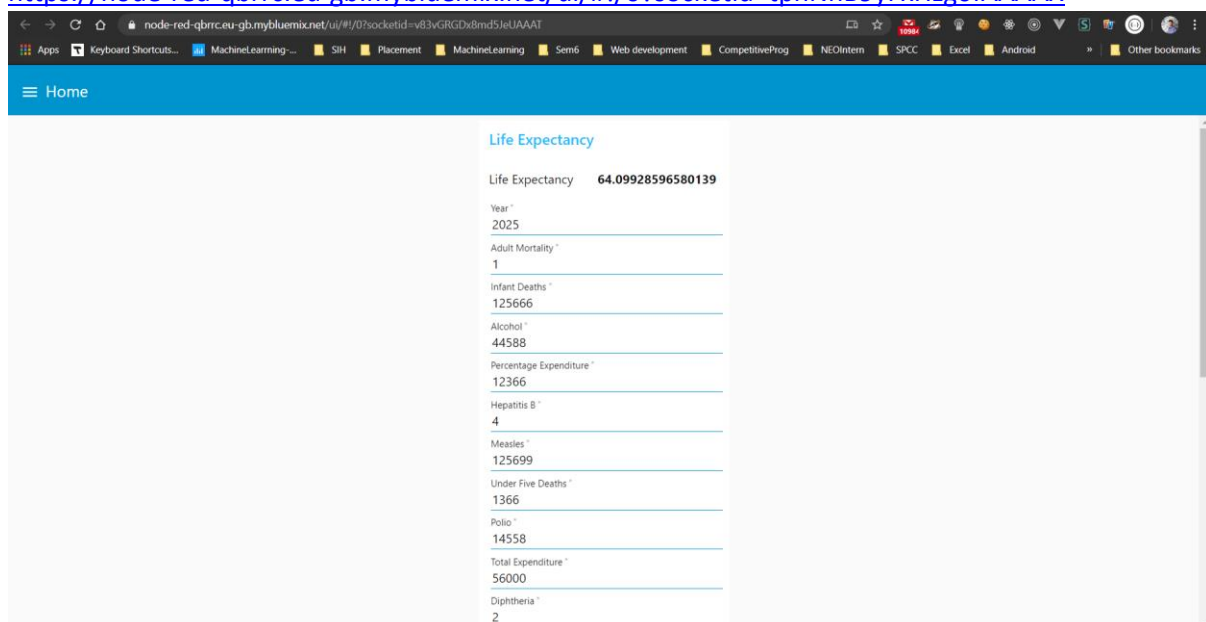
Software:

- Browser : Google Chrome, Mozilla Firefox,etc
- IBM cloud Software
- Node-Red App[Webpage].
- IBM Watson Studio

Designing:

A node red flow is made for the prediction of life expectancy, which is given below.

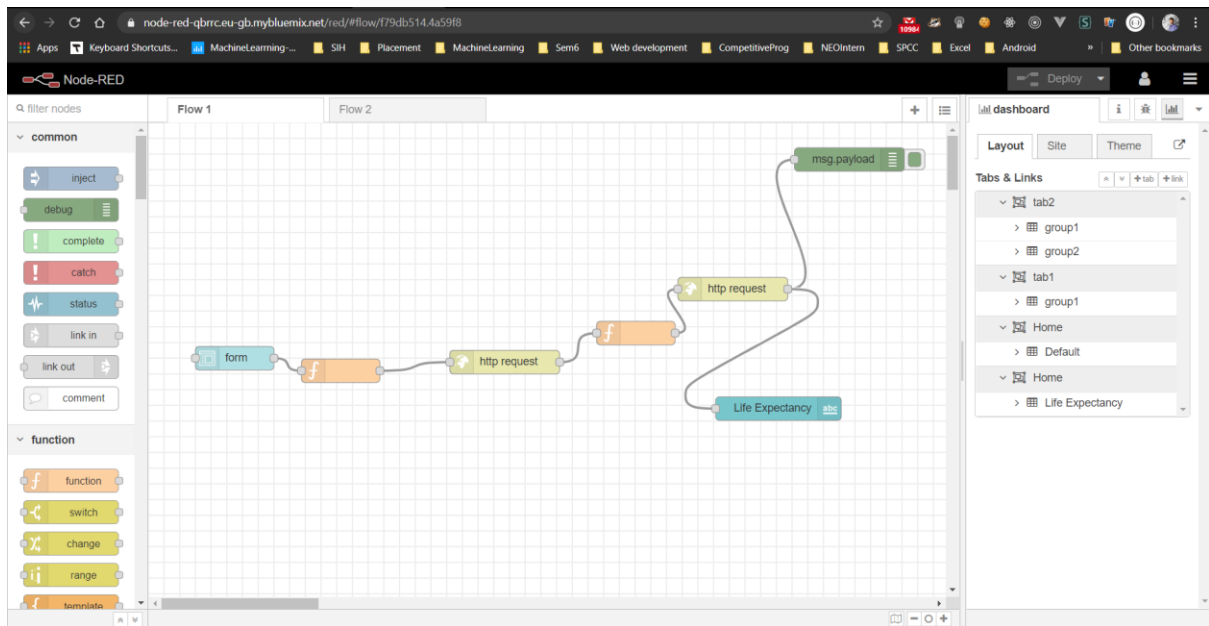
<https://node-red-qbrcc.eu-gb.mybluemix.net/ui/#!/0?socketid=tphNnB9yFxFzgOfAAAAAX>



The screenshot shows a web application interface with a blue header bar containing a 'Home' link. The main content area is titled 'Life Expectancy' and displays a list of input fields with their corresponding values. The values are as follows:

Field	Value
Life Expectancy	64.09928596580139
Year *	2025
Adult Mortality *	1
Infant Deaths *	125666
Alcohol *	44588
Percentage Expenditure *	12366
Hepatitis B *	4
Measles *	125699
Under Five Deaths *	1366
Polio *	14558
Total Expenditure *	56000
Diphtheria *	2

Deployed Website demo



Designing of Node RED flow

The screenshot shows the IBM Watson Studio interface. The top navigation bar includes the URL 'datapatform.cloud.ibm.com/ml/deployments/1b30b726-8680-42d7-b2ba-9cea2340c588/test?projectId=a7018e83-c8b6-4699-8d16-0141692800...' and the user 'Robinson Rodrigues's Account'. The main content area is titled 'Life_Expectancy_AutoAI' and has three tabs: 'Overview', 'Implementation', and 'Test'. The 'Test' tab is active, showing a form for 'Enter input data' with fields for 'Country' (India), 'Year' (2014), 'Status' (Developing), and 'Adult Mortality' (184). A 'Predict' button is at the bottom left. On the right, a JSON response is displayed:

```
{  "predictions": [    {      "fields": [        "prediction"      ],      "values": [        67.63999938964844      ]    }  ]}
```

Auto AI Experiment

4. Experimental Investigations

- **Dataset:**

Collection of data set from Kaggle.

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

- **Method 1 (Python with IBM Watson Studio):**

- Created a Empty Project in IBM Watson Studio.
- Created a Python notebook in the project.
- Imported the dataset.
- Data cleaning:
 - Changed the BMI input feature from numerical to ordinal.
 - Replaced the NULL values by appropriate data imputation methods for categorical and numerical values.
- Train test split:

```
#Splitting into training and testing datasets
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=1)
```

- Pipeline steps for further data transformation and training the model:

- One Hot Encoding for Categorical columns

```
cat_transform = Pipeline(steps = [('onehot', OneHotEncoder(handle_unknown = 'ignore'))],)
```

- Preprocessing Pipeline created

```
pre_process = ColumnTransformer(transformers=[
    ('cat', cat_transform, Life_Exp_df.columns[17:]),
])
```

- Linear Regression model

```
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor
```

```
lreg = LinearRegression()
dtreg = DecisionTreeRegressor()
```

```
pip_lreg = Pipeline([('preprocessor', pre_process), ('LRegressor', lreg)])
```

```
pip_lreg.fit(X_train, y_train)
```

```
Pipeline(memory=None,
 steps=[('preprocessor', ColumnTransformer(n_jobs=None, remainder='drop', sparse_threshold=0.3,
 transformer_weights=None,
 transformers=[('cat', Pipeline(memory=None,
 steps=[('onehot', OneHotEncoder(categorical_features=None, categories=None,
 dtype=<class 'numpy.float64'>...gressor', LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
 normalize=False)))]))])])
```

- Decision Tree Regression model

```
pip_dtreg = Pipeline([('preprocessor',pre_process),('DTRegressor',dtreg)])
pip_dtreg.fit(X_train,y_train)
```

- Random Forest Regression model (Best accuracy model)

```
from sklearn.ensemble import RandomForestRegressor

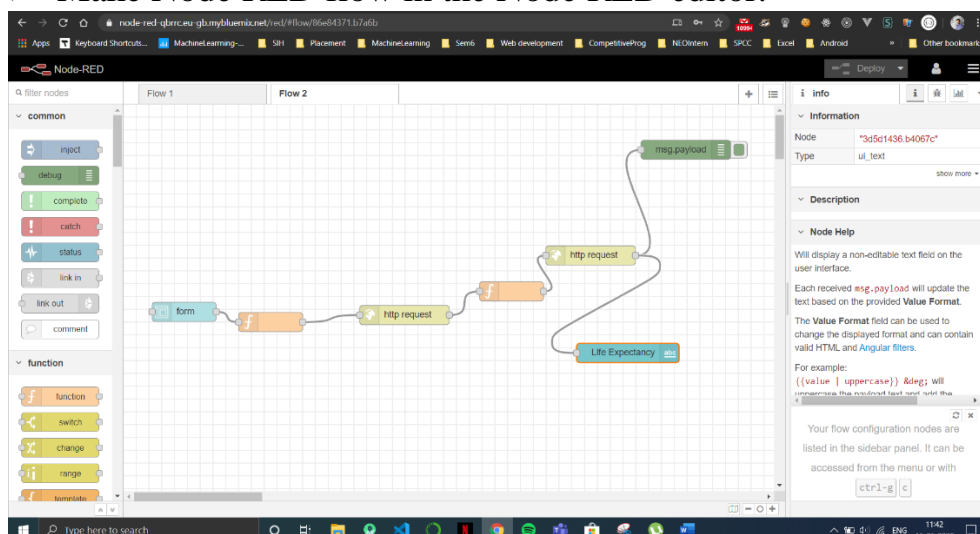
rfreg = RandomForestRegressor(max_depth = 100, n_estimators = 1200, random_state = 0)

pip_rfreg = Pipeline([('preprocessor',pre_process),('RFRegressor',rfreg)])
pip_rfreg.fit(X_train,y_train)
```

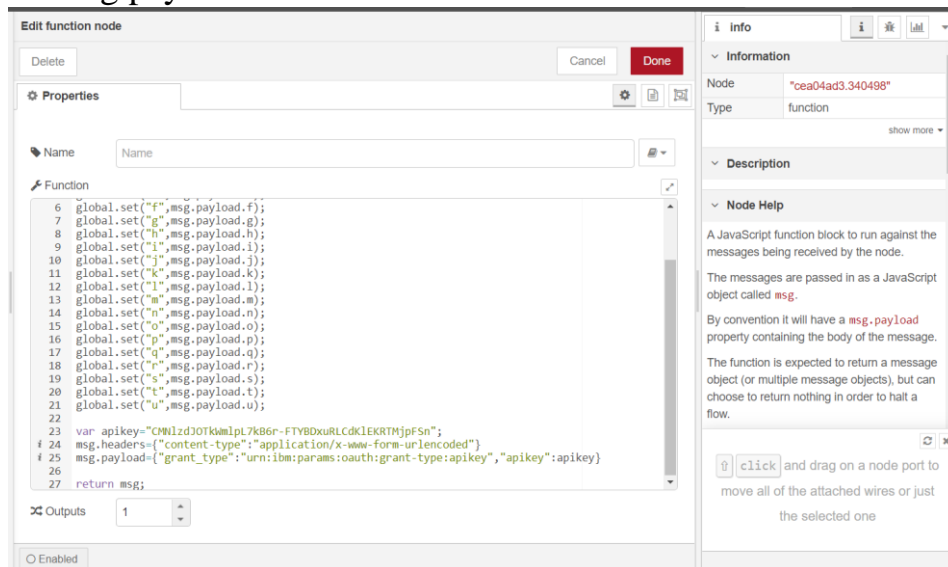
- **Method 2 (Auto AI):**

On IBM Watson Studio machine learning using Auto AI build a model to predict life expectancy.

- To do so first create account on IBM Watson studio.
- Using Add to project choose auto ai.
- Then upload data set
- Choose best way to predict.
- Save as a model which is on the top
- Deploy the model.
- Test the model.
- Create service credential
- Create cloud foundry app
<https://node-red-qbrcc.eu-gb.mybluemix.net/red/#flow/f79db514.4a59f8>
- Make Node RED flow in the Node RED editor.



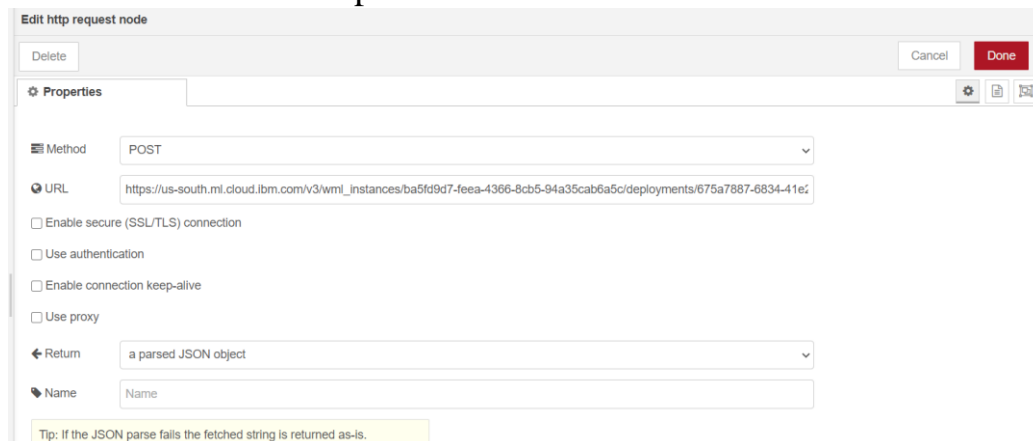
- Then add the API key in the function which accepts the form's msg.payload.



- Add an instance id in another function.

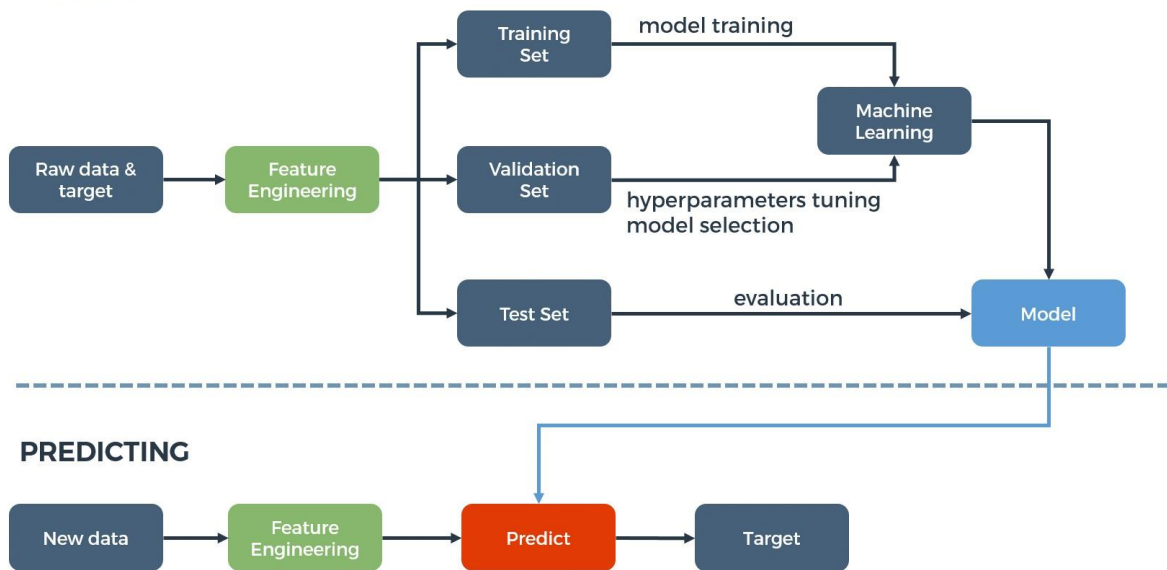


- Then add the response_scoring url from Auto AI deployment model to the http request in the Node RED editor so that it can access the Auto AI model for prediction.



5. Flowchart

TRAINING



6. Result

Accuracy:

- Linear Regression Model

```
from sklearn.metrics import r2_score

print("Simple Linear Regressor R2 Score: ", r2_score(pip_lreg.predict(X_test), y_test))
print("Simple Linear Regressor Train Score: ", pip_lreg.score(X_train, y_train))
print("Simple Linear Regression Test Score: ", pip_lreg.score(X_test, y_test))
```

Simple Linear Regressor R2 Score: 0.8289942487844177
Simple Linear Regressor Train Score: 0.9729682761514374
Simple Linear Regression Test Score: 0.8535551240248549

- Decision Tree Regression Model

```
print("Decision Tree Regressor R2 Score: ", r2_score(pip_dtreg.predict(X_test), y_test))
print("Decision Tree Train Score: ", pip_dtreg.score(X_train, y_train))
print("Decision Tree Test Score: ", pip_dtreg.score(X_test, y_test))
```

Decision Tree Regressor R2 Score: 0.9038698420485863
Decision Tree Train Score: 0.9962195921216345
Decision Tree Test Score: 0.898264765630321

- Random Forest Regression Model

```
print("Random Forest Regressor R2 Score: ", r2_score(pip_rfreg.predict(X_test), y_test))
print("Random Forest Train Score: ", pip_rfreg.score(X_train, y_train))
print("Random Forest Test Score: ", pip_rfreg.score(X_test, y_test))
```

Random Forest Regressor R2 Score: 0.8866294769913609
Random Forest Train Score: 0.9849440582074698
Random Forest Test Score: 0.9055888770522058

- Auto AI Model

My projects / Life Expectancy / life_expectancy_autoai

Experiment summary

Pipeline comparison

Rank by: Root mean squared err...

Score:

Cross validation

Holdout

Pipeline leaderboard

Rank	↑	Name	Algorithm	RMSE (Optimized)	Enhancements	Build time
>	★ 1	Pipeline 3	Extra Trees Regressor	2.010	HPO-1 FE	00:00:54
>	2	Pipeline 4	Extra Trees Regressor	2.010	HPO-1 FE HPO-2	00:00:36
>	3	Pipeline 1	Extra Trees Regressor	2.070	None	00:00:01
>	4	Pipeline 2	Extra Trees Regressor	2.070	HPO-1	00:00:12
>	5	Pipeline 7	Decision Tree Regressor	2.742	HPO-1 FE	00:00:43
>	6	Pipeline 8	Decision Tree Regressor	2.742	HPO-1 FE HPO-2	00:00:08
>	7	Pipeline 5	Decision Tree Regressor	2.807	None	00:00:01
>	8	Pipeline 6	Decision Tree Regressor	2.807	HPO-1	00:00:02

▼

★ 1

Pipeline 3

Extra Trees Regressor

2.010

HPO-1 FE

Model evaluation measures

	Cross validation score	Holdout score
Explained variance	0.956	0.961
MAE	1.282	1.182
MSE	4.057	3.347
MSLE	0.001	0.001
MedAE	0.747	0.740
RMSE	2.010	1.830
RMSLE	0.031	0.028
R ²	0.956	0.961

7. Advantages and Disadvantages

Advantages:

- Since we can predict the life span, we can know what factors are influencing the expectancy on life span in what ways.
- So, therefore by trying to change those factors in the real world we can increase the life span.
- Random Forest Regression model gives a good accuracy score for this dataset.

Disadvantages:

- Predictions are not always true, but it works in maximum number of cases, except for some exceptional ones.
- Testing accuracy could be increased if we can have more amount of data.

8. Applications

Correlations:

- There is a strong positive correlation between 'Schooling' and 'Life Expectancy'. This may be because education is more established and prevalent in wealthier countries. This means countries with less corruption, infrastructure, healthcare, welfare, and so forth.
 - Similarly, there is a moderate positive correlation between 'GDP' and 'Life Expectancy', most likely due to the same reason.
 - There's a moderate positive correlation between 'Alcohol' and 'Life Expectancy'. This is due to the fact that the consumption of alcohol is more prevalent among wealthier populations.
-
- Predicting life expectancy will help to country to know their average rate of life expectancy.
 - Country can analyse what facors affects more to increase life expectancy.
 - Country can also analyse which factors affects more to decrease life expectancy so that they can take appropriate decision to increase life expectancy of human being in their country.

9. Conclusion

Predicting Life Expectancy using Machine Learning project will help country to know their life expectancy.

10. Future Scope

The problem of processing datasets such as electronic medical records(EMR) and their integration with genomics, environmental factors, socioeconomic factor and patient behavior variations have posed a problem for researchers the health industry. Due to rapid innovations in machine learning field such as big data, analytics, visualization, deep learning, health workers now have improved way of processing, and developing meaningful information from huge datasets that have been accumulated over many years.

Big data and machine learning can benefit public health researchers with analysing thousands of variables to obtain data regarding life expectancy. We can use demographics of selected regional areas and multiple behavioural health disorders across regions to find correlation between individual behaviour indicators and behavioural health outcomes.

11. Bibilography

Dataset:

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

How to create Project Planning and Kickoff:

<https://www.allbusinesstemplates.com/download/?filecode=2KBA4&lang=en&iuid=9f9faa69-9fab-%2040ee-8457-ea0e5df8c8de>

IBM Academic Initiative account:

<https://my15.digitalexperience.ibm.com/b73a5759-c6a6-4033-ab6b-d9d4f9a6d65b/dxsites/151914d1-%2003d2-48fe-97d9-d21166848e65/academic/home>

Creating a Node-RED application:

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

APPENDIX:

A. Source Code:

https://github.com/SmartPracticeschool/IISPS-INT-2004-Predicting-Life-Expectancy-using-Machine-Learning/blob/master/life_expectancy.ipynb

B. Github Repository:

<https://github.com/SmartPracticeschool/IISPS-INT-2004-Predicting-Life-Expectancy-using-Machine-Learning>

C. Node RED flow:

<https://node-red-qbrcc.eu-gb.mybluemix.net/ui/#/0?socketid=tphNnB9yFxFHzg0fAAAAAX>