Project Report

Life Expectancy Prediction

Index

1. Introduction

1.1 Overview

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 the 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 the healthcare system and some specific disease-related deaths that happened in the country are given.

1.2 Purpose

The purpose of this project is to develop a machine learning regression model to use the existing data such that the model helps in predicting the life expectancy of the people living in different parts of the world having different geographic, demographic, economic and social background. The second purpose of this project is to identify the features/reasons which are responsible for the variation of life expectancy among people.

2. Literature Survey

2.1 Existing Problem

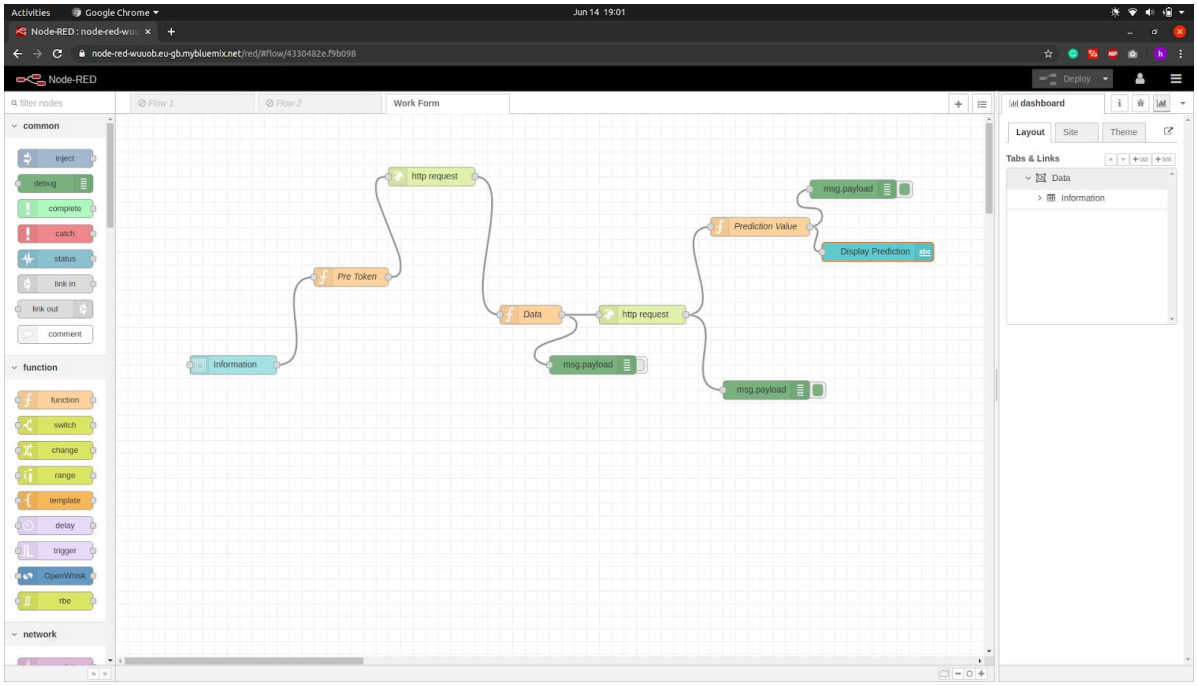
At present, we have the data available for life expectancy. With this raw data, we are not able to predict the life expectancy of people as well as we are not able to identify the major reasons/features that are responsible for given the value of the life expectancy.

2.2 Purposed Solution

To develop a machine learning regression model to use this existing raw data which predicts the life expectancy of the people living in different parts of the world having different geographic, demographic, economic and social background as well as help to identify the major factor.

3. Theoretical Analysis

3.1 Block Diagram of node-red flow



This image shows the node-red flow diagram.

3.2 Software/Hardware designing

For resolving the problem I have created software on IBM Watson because linking the backend (ML code) to the frontend is relatively simple in it. For the frontend, I have used node-red as it is simple and creates a webpage by easily linking different nodes each adding a different property to the webpage. The above figure is the node-red flow which I have created for the frontend. On the website, the UI seen is the default UI of the node-red

flow. We can also edit it according to our needs.

The regression model was written in on the jupyter notebook, which was available on the Watson Studio Service. Watson Studio Service was used as it was easy to link the file present in it to the node-red. As for the linear regression model I have used the Random Forest Regressor as it was giving the least RSME value (I have also tried some other models and this was giving the best accuracy). Random Forest Regressor is an ensemble of several decision trees, where every tree has a weight associated with it for the output. As a standard practice, the train-to-test ratio of data was 75%-25%. The standard scalar normalisation technique was also applied to the selected data columns in order to convert all data to the same scale of -1 to 1. This was done because the value in GDP was exponentially larger than the some the other parameters like alcohol, HIV/AIDS etc. so this was decreasing the model accuracy to some extent. Some entries in the data were blank giving the NAN value. This was first handled by filling up these using the mean of the column. After this, blank values were still left, for these, I removed them from the data as filling these values were consuming a lot of time.

For linking the frontend and backend WatsonMachineLearningAPIClient library was added to jupyter notebook. This helped in creating an API which directly calculates the output using the model, based on the input provided that the format of the input given and the variables accepted by the model are in the same format.

4. Experimental Investigations

While working with the data I came observed many things which were inconsistent with a real-life scenario. Some of these were outliers while others seem to be the human error while compiling the data. The following points summarise these observations-

- The minimum infant deaths are 0 which is hard to believe
- The minimum adult mortality is 1.

- Similarly, the under-five death can't be 0.
- BMI value of 1 seems too low and the value of 87.3 seems too high for any person to have.
- Many of the columns were blank giving NAN value of the same.
- The population of 34 also seems unrealistic.

5. Result

According to the different metrics of accuracy, the values were -

- Root Mean Square Error (RSME) - 1.7977954833759113
- Mean Absolute Error (MAE): 1.1169747747189136
- Mean Square Error (MSE): 3.232068600046827

The low value of the RSME values shows the prediction values by the model were close to the actual value of the given the in data.

6. Applications

The main application of this project would to changes some of the values of the input features to identify the ways by which life expectancy of people can improve in every part of the world.

7. Conclusion

Having such a low RSME makes this model very good at predicting the life expectancy of the people provided the data given is accurate.

8 Future Scope

The model created above can further be improved if the data given has more accurate values as well as has fewer outliers as well as if the human error(blank data, questionable data entries etc.) in compiling the data is decreased.

9. Bibliography

The project was done using the following links as the references for learning IBM Watson as well as about Machine Learning

- Webinars by SmartInternz Mentor's
- Youtube playlist of IBM Watson - https://www.youtube.com/watch?v=W3iPbFTAAds&feature=youtu.be
- IBM Documentation on Red Node - https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application

Appendix

The can be found in the GitHub repository - https://github.com/SmartPracticeschool/llSPS-INT-2067-Predicting-Life-Expectancy-using-Machine-Learning