

# 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**.

## 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.ensemble` 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 pallette.
- Now create the flow with the help of following node.
  - a. Inject
  - b. Debug
  - c. Function
  - d. Ui\_Form
  - e. Ui\_Text
- Deploy and run Node Red app.

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