|  |  |
| --- | --- |
| Life expectancy calculation | Abstract  Life Expectancy Calculation using machine learning has emerged as a promising application in the field of healthcare and data science. In this project, we developed Machine Learning Models to calculate Life Expectancy. Through this exploration, we aim to demonstrate the potential of machine learning in improving our understanding of longevity and its implications across various domains.  By Steshi  Btech Computer Science |

**INTRODUCTION**

Life Expectancy Calculation using machine learning has emerged as promising application in the field of healthcare and data science. The GHO keeps track of the health status as well as many other related factors for all countries. The datasets are made available to public for the purpose of health data analysis. In this project, we will train our models using this dataset. We will be using four machine learning models including Linear Regression, Multiple Linear Regression, Polynomial Regression and Multiple Polynomial Regression Model. Through this exploration, we aim to demonstrate the potential of machine learning in improving our understanding of longevity and its implications across various domains.

Such productive models not only aid in understanding the impact of different factors on life expectancy but can also inform policy decisions, healthcare resource allocation and intervention strategies to improve overall population health.

The social benefits of a Life Expectancy Calculation using machine learning are numerous and impactful.

* It can bring improvements in healthcare, economic planning, social services and overall quality of life.
* By analysing patterns in historical data, the model can identify risk factors and trends associated with certain diseases or conditions. This can lead to advancements in medicine, genetics, lifestyle interventions and public health strategies.
* Government and policymakers can utilize life expectancy predictions to make informed decisions about social welfare programs, retirement age policies and workforce planning contributing to a stable and sustainable economy.
* A better understanding of life expectancy factors can lead to public awareness about healthy lifestyles and behaviours that can positively impact longevity. This promotes healthier living and reduces the burden on healthcare systems.
* Its potential to empower decision-makers with accurate insights into longevity has the capacity to positively influence multiple aspects of society.

**DATA CONTENT**

The project relies on accuracy of data. The Global Health Observatory (GHO) data repository under World Health Organization (WHO) keeps track of the health status as well as many other related factors for all countries. The data-sets are made available to public for the purpose of health data analysis. The data-set related to life expectancy, health factors for 193 countries has been collected from the same WHO data repository website and its corresponding economic data was collected from United Nation website. Among all categories of health-related factors only those critical factors were chosen which are more representative.

It has been observed that in the past 15 years, there has been a huge development in health sector resulting in improvement of human mortality rates especially in the developing nations in comparison to the past 30 years. Therefore, in this project we have considered data from year 2000-2015 for 193 countries for further analysis. The individual data files have been merged together into a single data-set.

On initial visual inspection of the data showed some missing values. As the data-sets were from WHO, we found no evident errors. Missing data was handled in R software by using Missmap command. The result indicated that most of the missing data was for population, Hepatitis B and GDP. The missing data were from less known countries like Vanuatu, Tonga, Togo, Cabo Verde etc. Finding all data for these countries was difficult and hence, it was decided that we exclude these countries from the final model data-set.

The final merged file (final dataset) consists of 22 Columns and 2938 rows which meant 20 predicting variables. All predicting variables was then divided into several broad categories:​Immunization related factors, Mortality factors, Economical factors and Social factors.

**EDA AND PREPROCESSING**

The dataset is not always ready to use. Exploratory Data Analysis (EDA) and Preprocessing are crucial steps in the Machine Learning process. We began by loading the data and then we described its size. After that we observed that some of our column names contained trailing spaces. Thus, we renamed them to remove the same. Next, we checked the data types of the features. It was correct for all the features. Following that, we checked for the null values and found missing values in some features. For some of them, we filled the null values with the mean values and for the remaining we dropped the null values. Subsequently, after cleaning the data we were left with 2896 rows and 22 columns including one target variable and 21 features.

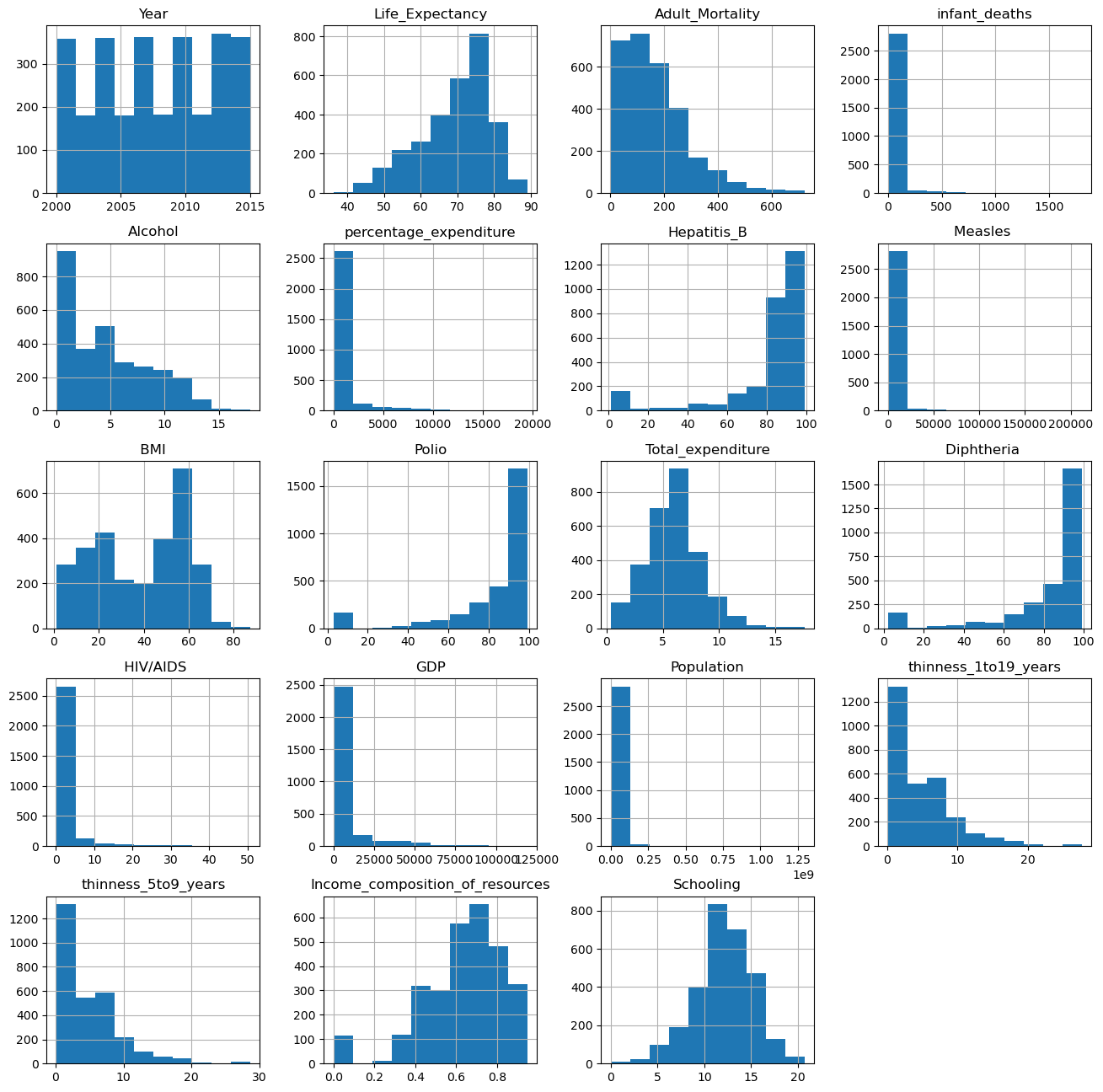
After cleaning the dataset, we proceeded to visualize the dataset and draw more results from it.

In the first place, we began by visualizing our target variable using swarmplot from the seaborn library. The results from the visualization are shown in the figure:



It seems from the plot that the life expectancies are between 55 to 85 years of age.

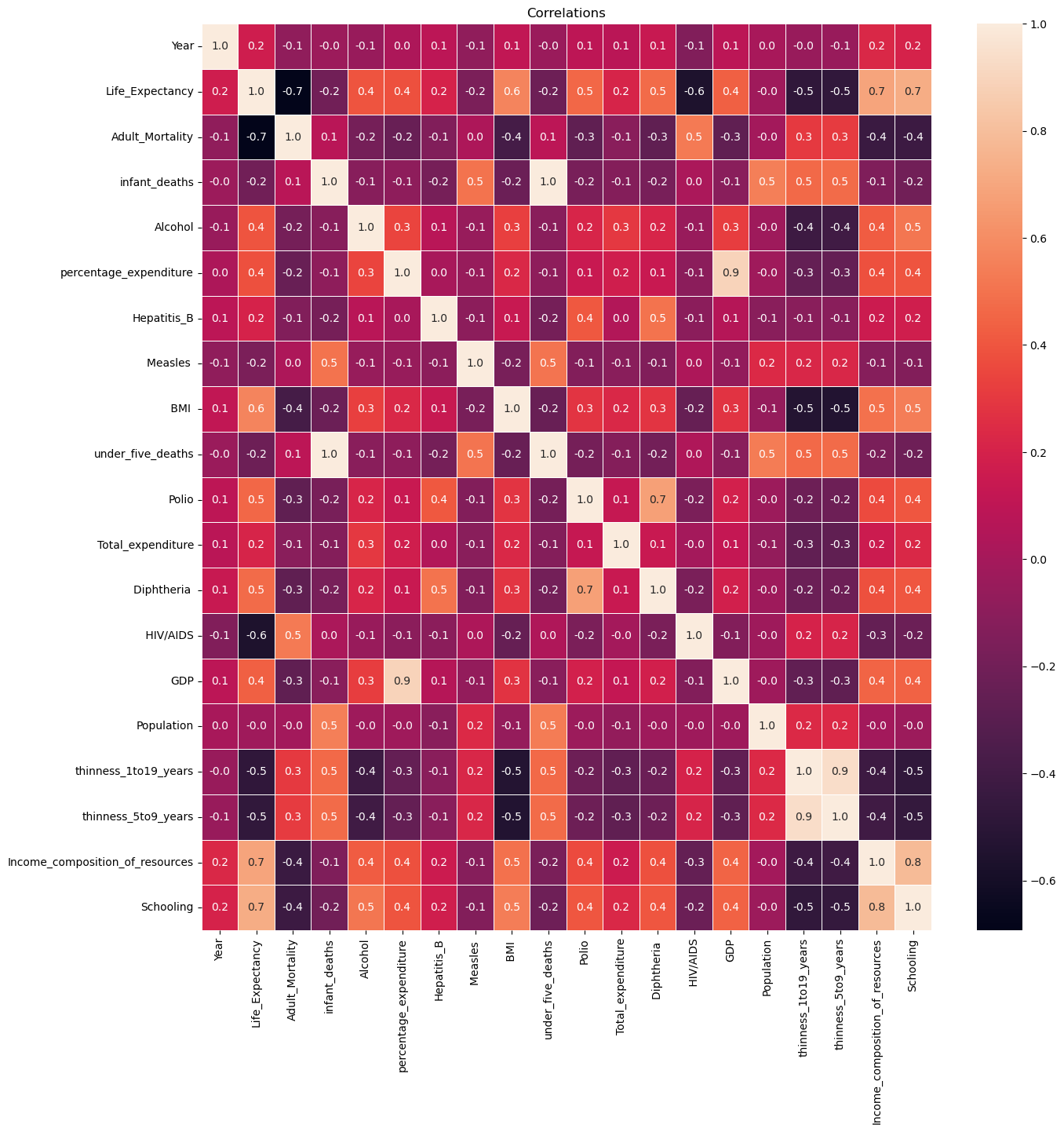
Following that, we plotted the histogram for all the features to show the distribution of their values. It is shown as below:



The above histograms show the distribution of the values of the

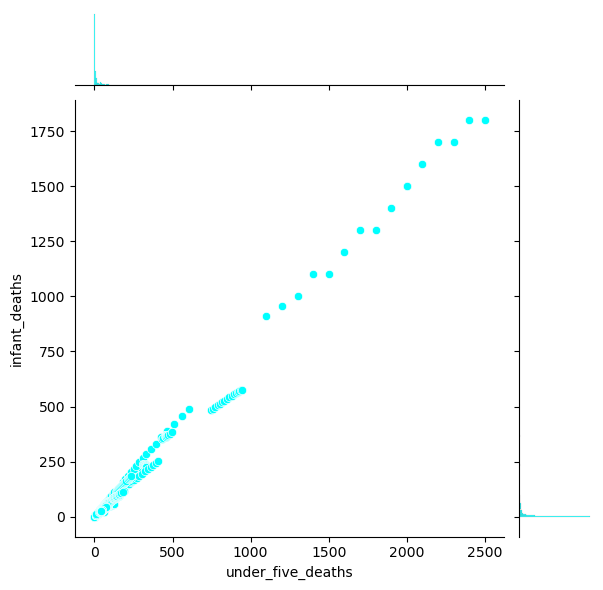
features by dividing them into intervals(or bins) and showing the frequency of data points falling within each bin.

Moving forward, we plotted the heatmap for all the features to show the correlation among them. It is a graphical representation of data using colours to visualize the value of the matrix. It is shown as below:



As it can be seen from the heatmap, under\_five\_deaths and infant\_deaths

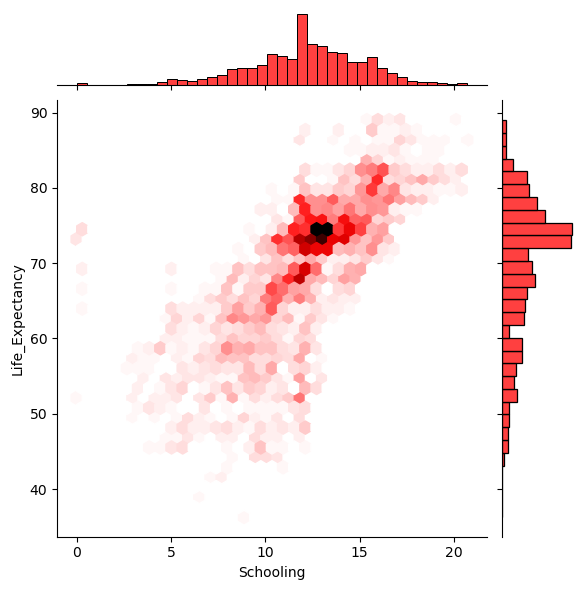
are correlated. We verified this by using the jointplot as shown:

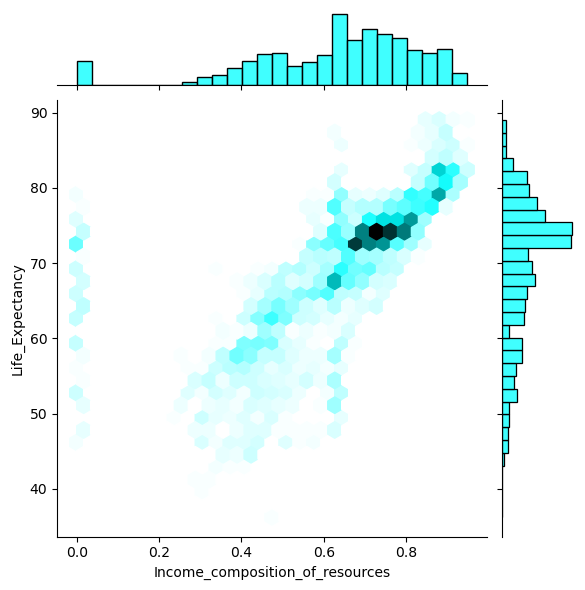


From the above plot we concluded that these features are highly correlated with each other. Thus, we removed one of them(under\_five\_deaths) before further processing of data.

Further on, we will see the impact of other features on Life Expectancy using hexplots from seaborn, beginning with the most correlated feature with Life Expectancy:

* ‘Schooling’ and ‘Income Composition of Resources’ features (with 0.7 correlation):

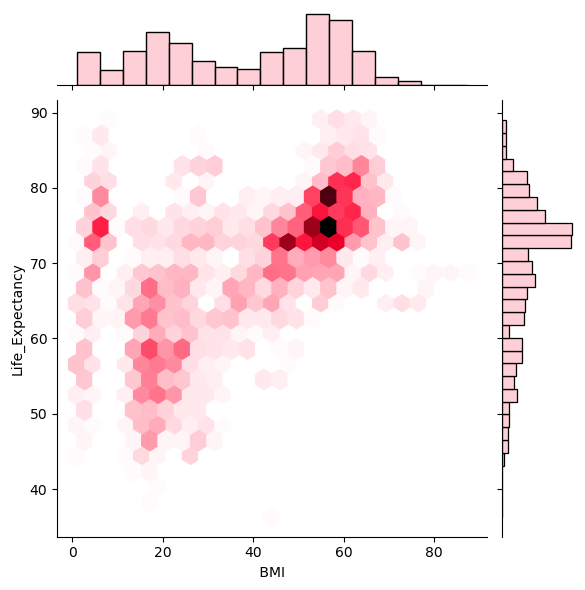




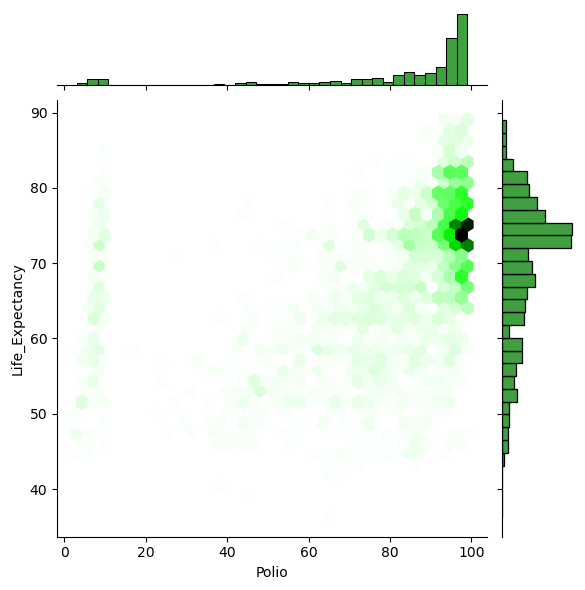
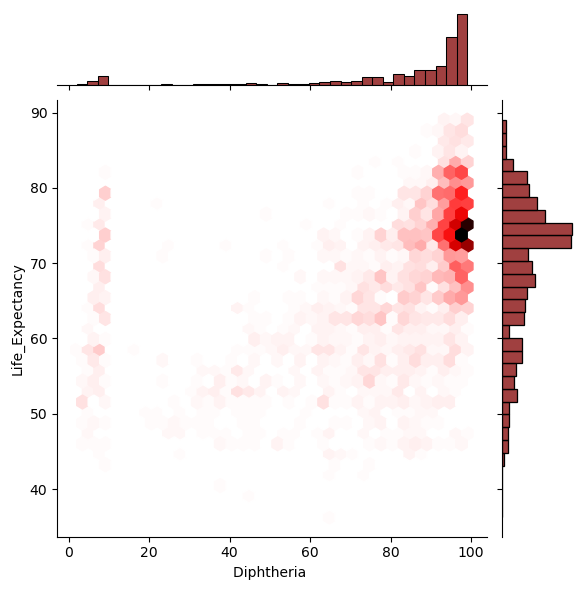
These hexplots split the plotting window into various hexbins and the number of observations which fall into each bin corresponds with a colour to indicate density. A darker colour hexbin means that there are more observations or more density within that region. The observations frequency bar can also be seen along the spines as an additional reference for information.

After that, we used hexplots for plotting the remaining features that are correlated with Life Expectancy.

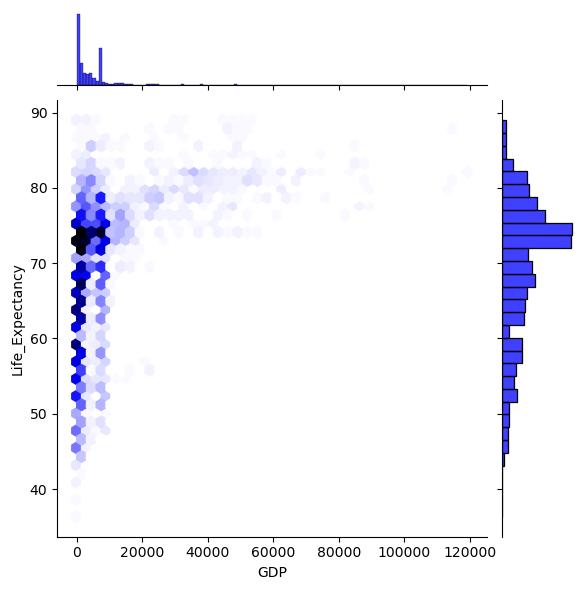
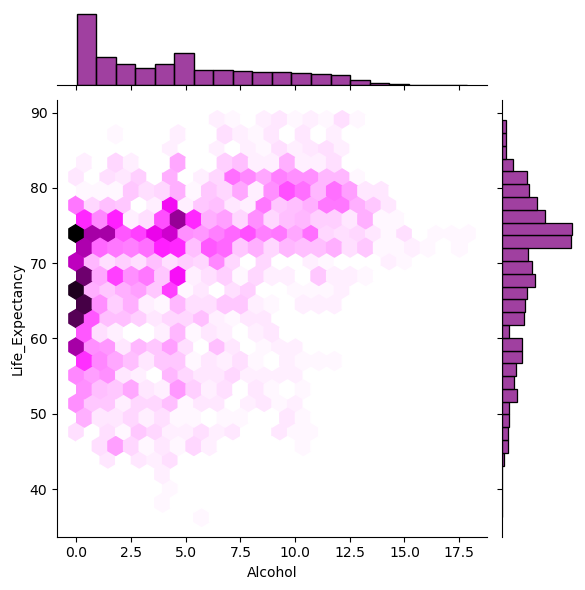
* BMI(with 0.6 correlation):



* Diphtheria and Polio(with 0.5 correlation):

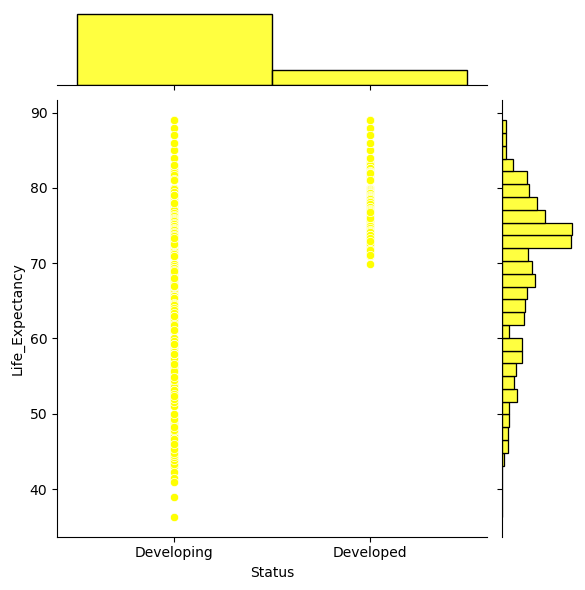


* Alcohol and GDP(with 0.4 correlation):



Thus, the hexplots shows the variation of observations and their corresponding densities.

After this, we plotted the variation of ‘Status’ feature with the Life Expectancy using jointplot as shown in figure :



It is clear from the above plot that the Life Expectancies in case of Developed countries is more as compared to that of Developing countries.

Now we have visualized and cleaned our data. Before creating the machine learning models, we first label encode our features which are in the form of strings into numerical form. It is done with the help of LabelEncoder, it is used to transform non-numerical labels to numerical labels.

In this project we would be doing the Input-Output Split(Splitting of the input features and the target variable) and Test-Train Split(Splitting the data for testing and training purpose, with 70% being used for training and 30% for testing) according to the regression model itself.

**REGRESSION MODELS**

Since we have to predict a range of values in this model, it is a regression task. In this project, I would be using four different Machine Learning models: Linear Regression, Multiple Linear Regression, Polynomial Regression and Multiple Polynomial Regression. Finally, we compared the performance of all the models applied and chose the one with the highest performance.

**LINEAR REGRESSION**

Linear regression is a type of supervised machine learning algorithm that computes the linear relationship between a dependent variable and an independent features. The goal of the algorithm is to find the best linear equation that can predict the value of the dependent variable based on the independent variables. The equation provides a straight line that represents the relationship between the dependent and independent variables. The slope of the line indicates how much the dependent variable changes for a unit change in the independent variable. Linear regression is used in many different fields, including finance, economics, and psychology, to understand and predict the behavior of a particular variable.

We have assumed earlier that our independent feature is X and Y is the dependent variable. Let’s assume there is a linear relationship between x and Y that can be predicted using:

Y= w0 + w1x

The model gets the best regression fit line by finding the best θ1 and θ2 values.

* **w0:** intercept
* **w1:** coefficient of x

Once we find the best θ1 and θ2 values, we get the best-fit line.

For this model, we got the following results:

* Mean Absolute error: 4.89
* Mean Squared error: 44.87
* R2\_score: 0.52

**MULTIVARIATE** **LINEAR REGRESSION**

Multiple Linear Regression is one of the important regression algorithms which models the linear relationship between a single dependent continuous variable and more than one independent variable. In Multiple Linear Regression, the target variable(Y) is a linear combination of multiple predictor variables x1, x2, x3, ...,xn. Since it is an enhancement of Simple Linear Regression, so the same is applied for the multiple linear regression equation, the equation becomes:

Y= w0 + w1x1 + w2x2 +w3x3 + ...... +wnxn.

**Y= Output/Response variable**

**w0, w1, w2, w3,… wn= Coefficients of the model.**

**x1, x2, x3, x4,...xn= Various Independent/feature variable.**

For this model, we got the following results:

* Mean Absolute error: 3.13
* Mean Squared error: 17.82
* R2\_score: 0.809

These results are clearly far better than the Linear Regression with single input only.

**POLYNOMIAL REGRESSION**

Polynomial Regression is a regression algorithm that models the relationship between a dependent(Y) and independent variable(x) as nth degree polynomial. If we apply a linear model on a linear dataset, then it provides us a good result as we have seen in Simple Linear Regression, but if we apply the same model without any modification on a non-linear dataset, then it will produce a drastic output. Due to which loss function will increase, the error rate will be high, and accuracy will be decreased. So for such cases, where data points are arranged in a non-linear fashion, we need the Polynomial Regression model. We can understand it in a better way using the below comparison diagram of the linear dataset and non-linear dataset. The Polynomial Regression equation is given below:

Y=w0+w1x1+ w2x12+ w2x13+...... wnx1n

It makes use of a linear regression model to fit the complicated and non-linear functions and datasets. Hence, in Polynomial regression, the original features are converted into Polynomial features of required degree (2,3,..,n) and then modelled using a linear model.

For this model, we got the following results:

* Mean Absolute error: 4.17
* Mean Squared error: 32.65
* R2\_score: 0.63

The results obtained from this model are better than the Linear Regression model but not than the Multiple Linear Regression.

**MULTIVARIATE POLYNOMIAL REGRESSION**

**Multivariate polynomial regression is used to model complex relationships with multiple variables. These complex relationships are usually non-linear and high in dimensions.** In this technique, the relationship is modelled as a polynomial function of the independent variables, allowing for more complex and nonlinear relationships to be captured. **Once an accurate equation (model) is created or found, this equation can be used for future accurate predictions.**

A Second Order Multiple Polynomial Regression can be expressed as:

Y=w1x1 + w2x12 + w11x2 + w22x22 + w12

Here, w1, w2 are called as linear effect parameters.

w11, w22 are called as quadratic effect parameters.

w12 is called as interaction effect parameter.

In this form, you can include various polynomial terms up to a certain degree for each independent variable. For example, you can include linear terms (xi​), quadratic terms (xi2​), interaction terms (xixj​), and so on. The choice of the degree of the polynomial and the interaction terms depends on the complexity of the underlying relationship between the variables and the data you're working with.

For this model, we got the following results:

* Mean Absolute error: 2.22
* Mean Squared error: 9.59
* R2\_score: 0.89

The results obtained from this model are better than the all of the other models implemented.

**RESULTS**

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Mean Absolute Error | Mean Squared Error | r2\_Score |
| Linear Regression | 4.89 | 44.87 | 0.52 |
| Multivariate Linear Regression | 3.13 | 17.82 | 0.809 |
| Polynomial Regression | 4.17 | 32.65 | 0.63 |
| Multivariate Polynomial Regression | 2.22 | 9.59 | 0.89 |

The above figure shows the Mean Absolute Error, Mean Squared Error and r2\_Score for all the models.

The best possible score for r2\_Score is 1. The more close is the score to 1, the more good is the model.

Clearly, the r2\_score is lower for the models which take single input (Linear Regression and Polynomial Regression) as compared to the models with multiple inputs (Multivariate Linear Regression and Multivariate Polynomial Regression).

However, the best r2\_score (0.89) is obtained from the Multivariate Polynomial Regression with 2.22 Mean Absolute Error and 9.59 Mean Squared Error.

**CONCLUSION**

It can be concluded that employing ML for Life Expectancy Calculation enhances prediction accuracy, provides deeper insights into influencing factors and enables more informed and targeted decision-making in healthcare and public policy. The analysis of life expectancy using all the four models shows that the Multivariate Polynomial Regression model achieved the highest R-squared(r2) score among the models. This indicates that the multivariate polynomial regression model provided the best fit to the data and explained the most variance in the life expectancy predictions. This implies that considering multiple variables and using a polynomial relationship improved the accuracy of the predictions and results in a more comprehensive representation of the underlying relationships affecting Life Expectancy as compared to the other models.