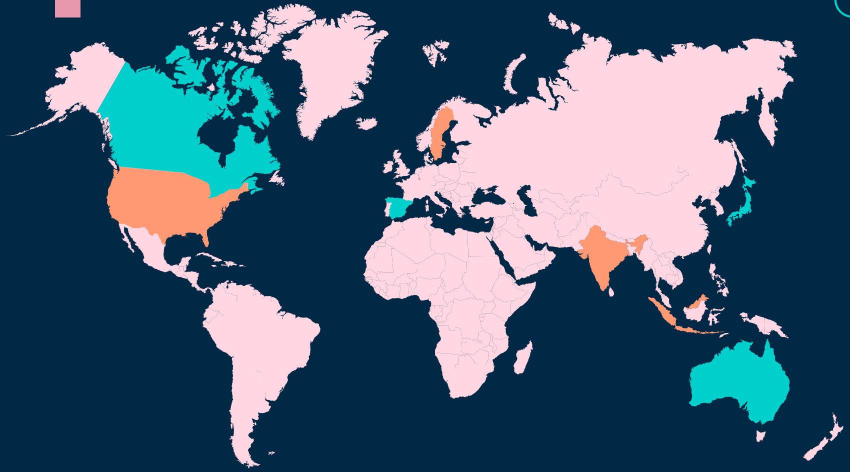


Country Life Expectancy

under supervision of:

Prof/Amira Yassin
Eng/Fatma Kamal

Understanding Life Expectancy

- ## Problem statement

  Life expectancy is a vital measure of a nation's healthcare effectiveness. However, it varies widely due to complex **economic** and **environmental factors Countries. So, understanding the key contributors to live expectancy can help policymakers develop focused strategies to enhance public health outcomes.**

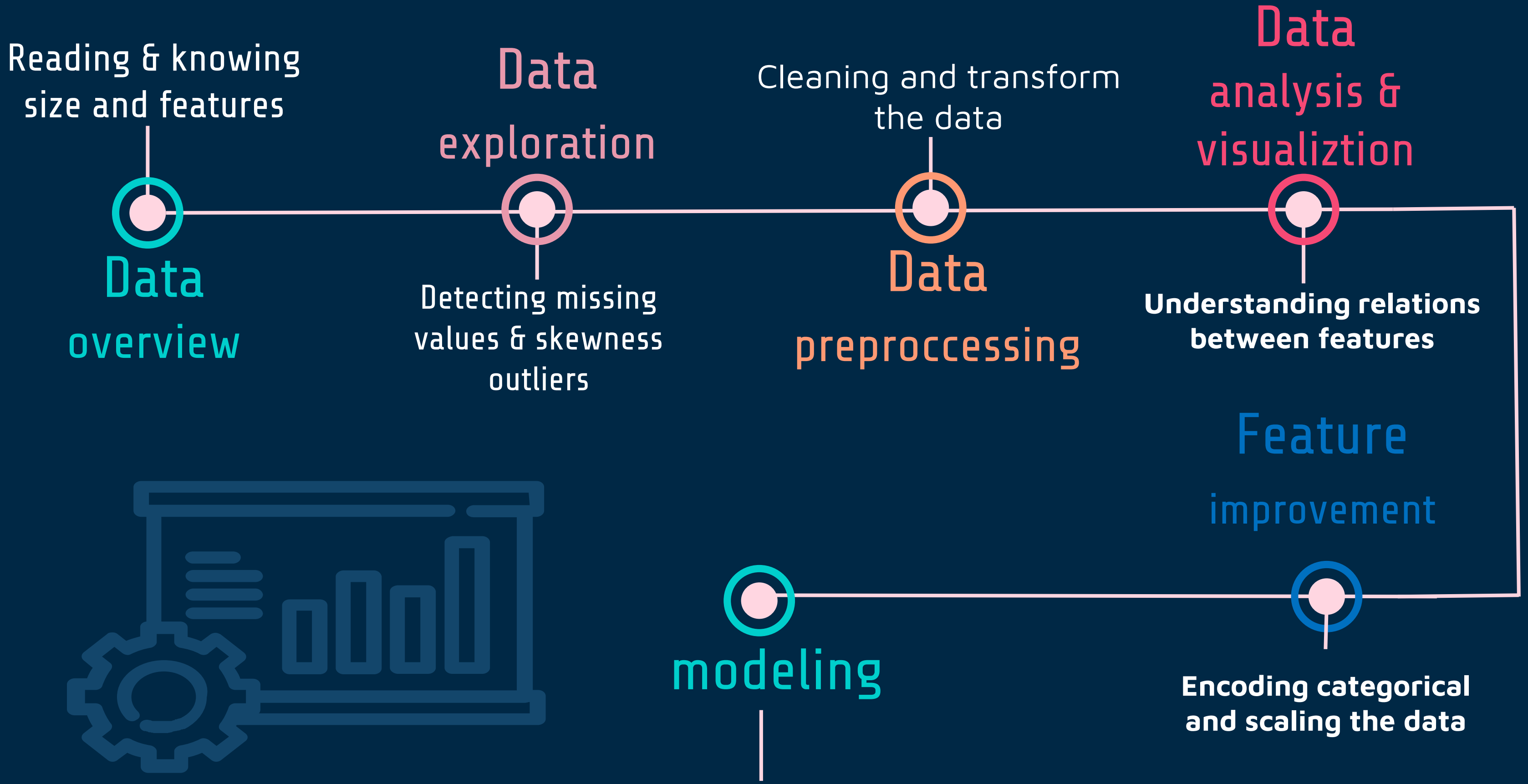
- ## Objective

  Predict **Life Expectancy (Years)** for nations using machine learning to uncover complex relationships between health, economic, demographic, and behavioral factors.

- ## Key Features

  Health indicators (e.g., vaccination coverage, mortality rates, HIV/AIDS prevalence),
  Economic metrics (e.g., GDP, health expenditure),
  Demographic data (e.g., total population, thinness percentage),
  Behavioral factors (e.g., alcohol consumption rate).

OUR PROCESS



Data overview

Reading & knowing size and features

Data exploration

Detecting missing values & skewness outliers

Data preproccessing

Cleaning and transform the data

Data analysis & visualiztion

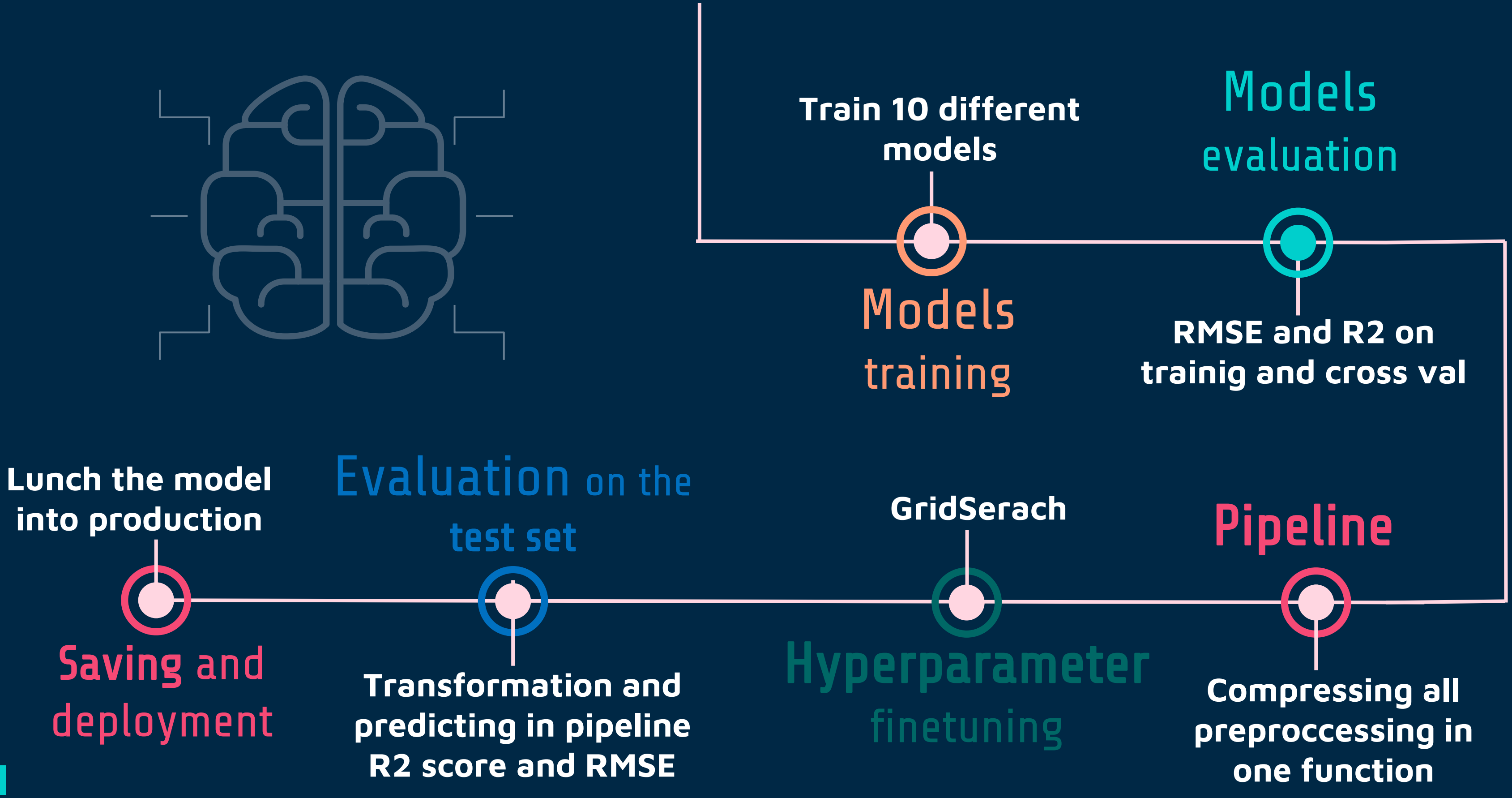
Understanding relations between features

Feature improvement

Encoding categorical and scaling the data

modeling

OUR PROCESS



Models training

Train 10 different models

Models evaluation

RMSE and R2 on trainig and cross val

Pipeline

Compressing all preproccessing in one function

Hyperparameter finetuning

GridSerach

Evaluation on the test set

Transformation and predicting in pipeline R2 score and RMSE

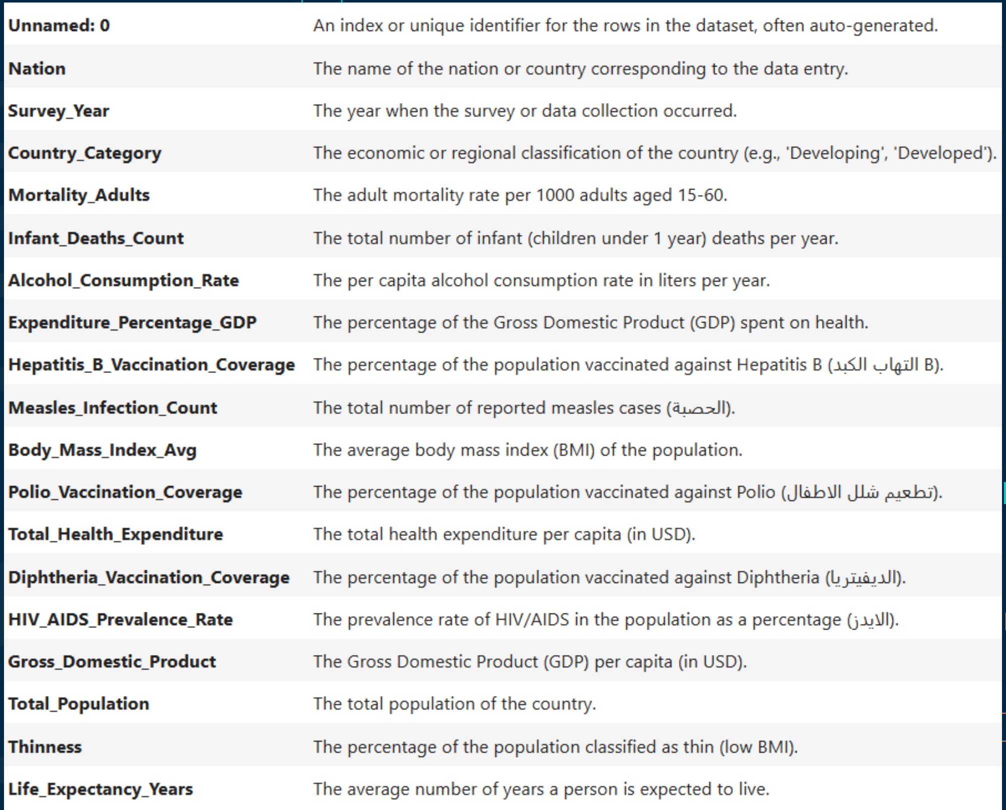
Saving and deployment

Lunch the model into production

Data overview

Reading & knowing size and features

Schema To understand the data

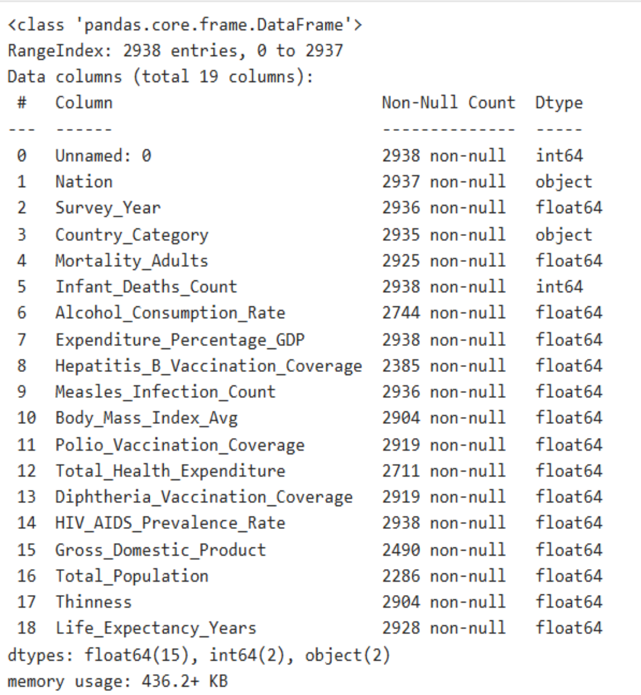


Unnamed: 0	An index or unique identifier for the rows in the dataset, often auto-generated.
Nation	The name of the nation or country corresponding to the data entry.
Survey_Year	The year when the survey or data collection occurred.
Country_Category	The economic or regional classification of the country (e.g., 'Developing', 'Developed').
Mortality_Adults	The adult mortality rate per 1000 adults aged 15-60.
Infant_Deaths_Count	The total number of infant (children under 1 year) deaths per year.
Alcohol_Consumption_Rate	The per capita alcohol consumption rate in liters per year.
Expenditure_Percentage_GDP	The percentage of the Gross Domestic Product (GDP) spent on health.
Hepatitis_B_Vaccination_Coverage	The percentage of the population vaccinated against Hepatitis B (التهاب الكبد B).
Measles_Infection_Count	The total number of reported measles cases (الحصبة).
Body_Mass_Index_Avg	The average body mass index (BMI) of the population.
Polio_Vaccination_Coverage	The percentage of the population vaccinated against Polio (تطعيم شلل الاطفال).
Total_Health_Expenditure	The total health expenditure per capita (in USD).
Diphtheria_Vaccination_Coverage	The percentage of the population vaccinated against Diphtheria (الديفيتريا).
HIV_AIDS_Prevalence_Rate	The prevalence rate of HIV/AIDS in the population as a percentage (الايدز).
Gross_Domestic_Product	The Gross Domestic Product (GDP) per capita (in USD).
Total_Population	The total population of the country.
Thinness	The percentage of the population classified as thin (low BMI).
Life_Expectancy_Years	The average number of years a person is expected to live.

01

Data Overview

Using info to look through the data

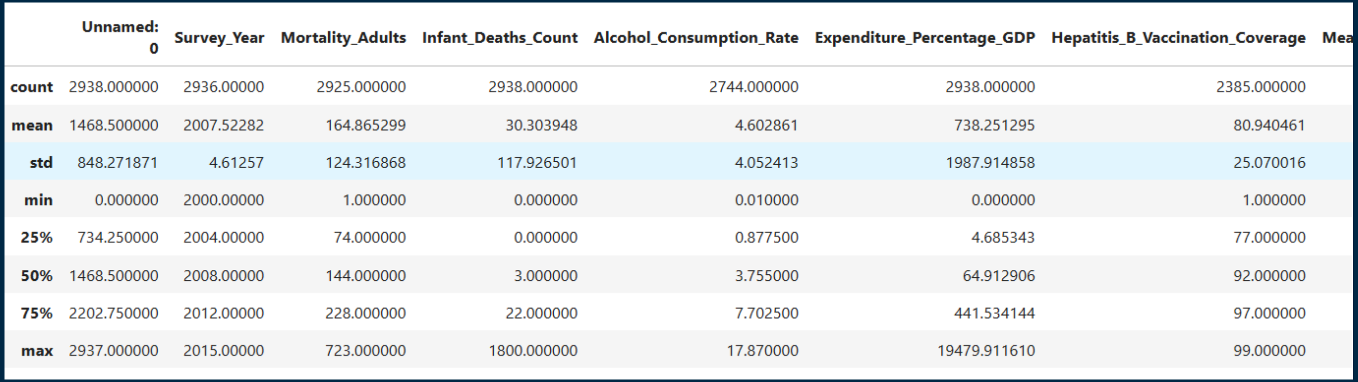


```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2938 entries, 0 to 2937
Data columns (total 19 columns):
 #   Column                           Non-Null Count  Dtype  
---  ------                           --------------  -----  
 0   Unnamed: 0                       2938 non-null   int64  
 1   Nation                           2937 non-null   object 
 2   Survey_Year                      2936 non-null   float64
 3   Country_Category                 2935 non-null   object 
 4   Mortality_Adults                 2925 non-null   float64
 5   Infant_Deaths_Count              2938 non-null   int64  
 6   Alcohol_Consumption_Rate         2744 non-null   float64
 7   Expenditure_Percentage_GDP       2938 non-null   float64
 8   Hepatitis_B_Vaccination_Coverage 2385 non-null   float64
 9   Measles_Infection_Count          2936 non-null   float64
 10  Body_Mass_Index_Avg              2904 non-null   float64
 11  Polio_Vaccination_Coverage       2919 non-null   float64
 12  Total_Health_Expenditure         2711 non-null   float64
 13  Diphtheria_Vaccination_Coverage  2919 non-null   float64
 14  HIV_AIDS_Prevalence_Rate         2938 non-null   float64
 15  Gross_Domestic_Product           2490 non-null   float64
 16  Total_Population                 2286 non-null   float64
 17  Thinness                         2904 non-null   float64
 18  Life_Expectancy_Years            2928 non-null   float64
dtypes: float64(15), int64(2), object(2)
memory usage: 436.2+ KB
```

01

Data Overview

Describe to get statistics about the data



	Unnamed: 0	Survey_Year	Mortality_Adults	Infant_Deaths_Count	Alcohol_Consumption_Rate	Expenditure_Percentage_GDP	Hepatitis_B_Vaccination_Coverage	Mea
count	2938.000000	2936.00000	2925.000000	2938.000000	2744.000000	2938.000000	2385.000000	
mean	1468.500000	2007.52282	164.865299	30.303948	4.602861	738.251295	80.940461	
std	848.271871	4.61257	124.316868	117.926501	4.052413	1987.914858	25.070016	
min	0.000000	2000.00000	1.000000	0.000000	0.010000	0.000000	1.000000	
25%	734.250000	2004.00000	74.000000	0.000000	0.877500	4.685343	77.000000	
50%	1468.500000	2008.00000	144.000000	3.000000	3.755000	64.912906	92.000000	
75%	2202.750000	2012.00000	228.000000	22.000000	7.702500	441.534144	97.000000	
max	2937.000000	2015.00000	723.000000	1800.000000	17.870000	19479.911610	99.000000	

01

Data Overview

Shape (2938,19)

Some of the data are Categorical
Nation and Country Category
and other are numerical
so make two lists to define them

Finally we spitted the data into train and test set

Data exploration

Detecting missing values & skewness outliers

02

Data Exploration

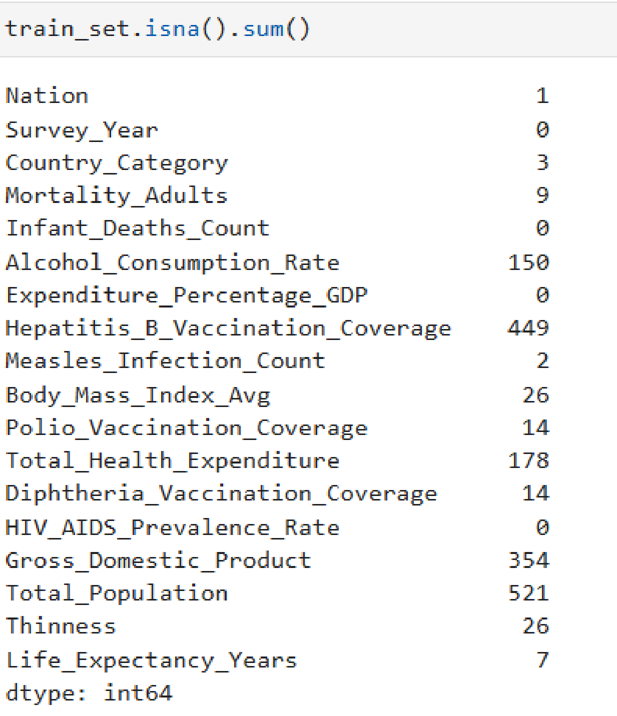
Inspect the data for :

- missing values
- Outliers
- Duplicates
- Skewness

02 Data Exploration

Missing values

There are missing values With different ranges In the dataset



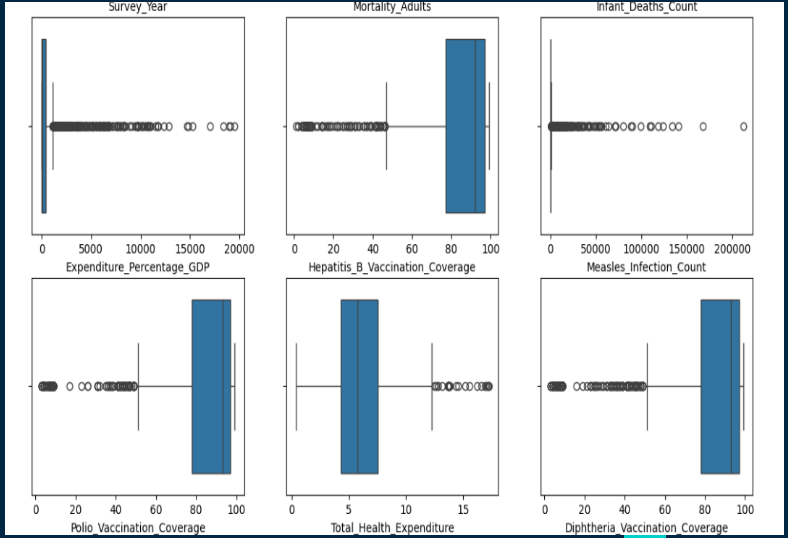
```
train_set.isna().sum()
```

```
Nation                              1
Survey_Year                         0
Country_Category                    3
Mortality_Adults                    9
Infant_Deaths_Count                 0
Alcohol_Consumption_Rate          150
Expenditure_Percentage_GDP          0
Hepatitis_B_Vaccination_Coverage  449
Measles_Infection_Count             2
Body_Mass_Index_Avg                26
Polio_Vaccination_Coverage         14
Total_Health_Expenditure          178
Diphtheria_Vaccination_Coverage    14
HIV_AIDS_Prevalence_Rate            0
Gross_Domestic_Product            354
Total_Population                  521
Thinness                           26
Life_Expectancy_Years               7
dtype: int64
```

02 Data Exploration

Outliers

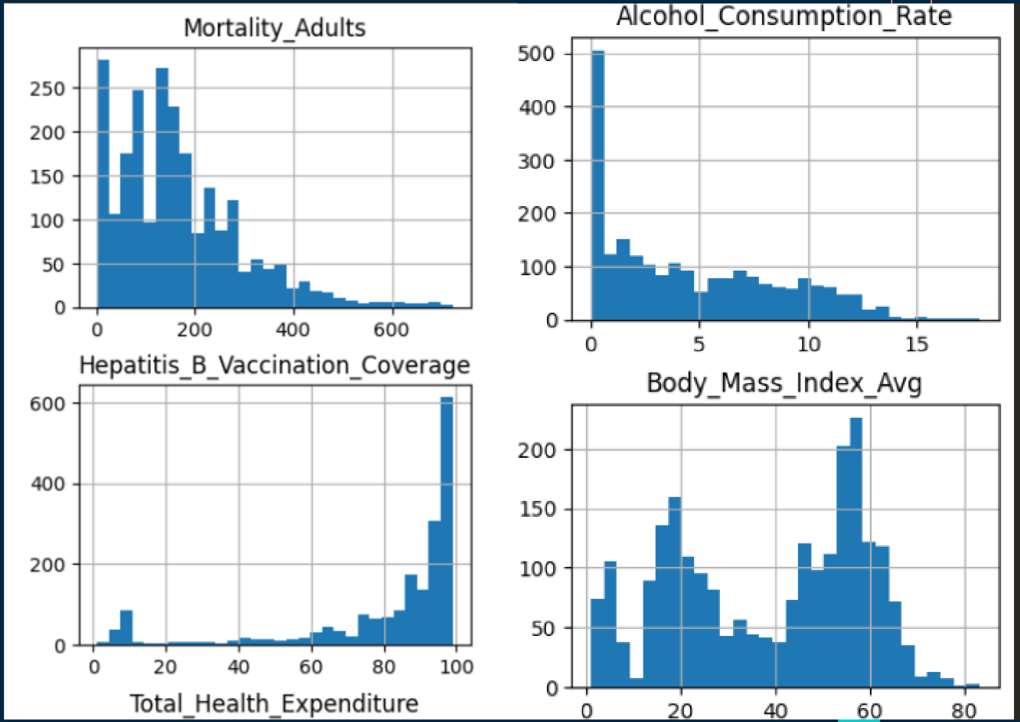
Using Boxplot there are a lot of outliers appear in each column



02 Data Exploration

Skewness

Using Histogram, there skewness in the data some to the left other to the right



02

Data Exploration conclusion

- **Unnamed Column**: It is useless, so we dropped it.
- **Data Types**: Defined the train_set into categorical and numerical types.
- **Duplicates**: There are no duplicate entries in the train_set.
- **Missing Values**: Most columns have a small number of missing values, but:
  - **Gross Domestic Product:** 15.25% missing
  - **Total Population:** 22.19% missing
  - **Hepatitis B Vaccination Coverage:** 18.82% missing
  - **Total Health Expenditure:** 7.73% missing
  - **Alcohol Consumption Rate:** 6.60% missing
- **Skewness**: Many columns exhibit skewness, which should be handled appropriately.
- **Outliers**: Detected many outliers through boxplots. These should be removed or handled.
- **Scaling**: The train_set has varying ranges, so it needs to be scaled.

Data preprocessing

Cleaning and transform the data

03 Data preprocessing

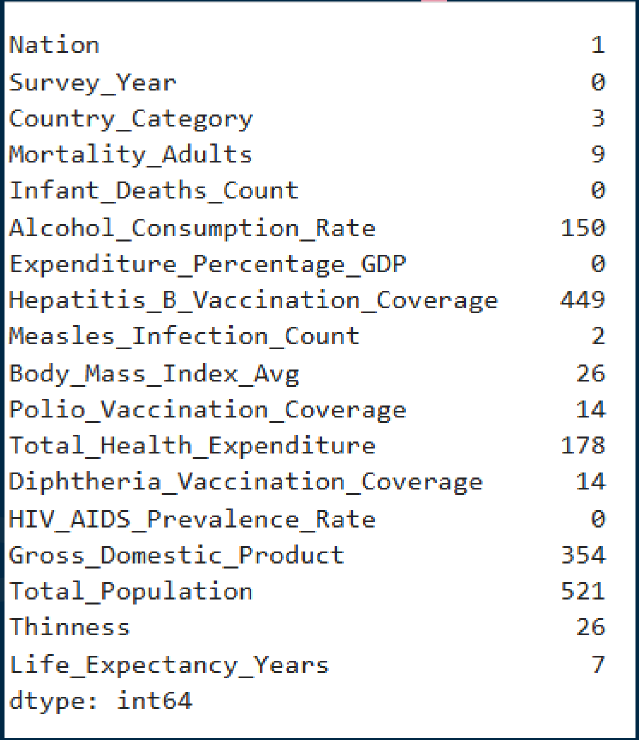
preprocessing data through :

- Handling wrong values
- Replacing missing values
- Handling outliers
- Handling skewness

03 Data preprocessing

Handling wrong values

- Some percentage column have wrong values that are beyond 100 so we replaced it by Nan

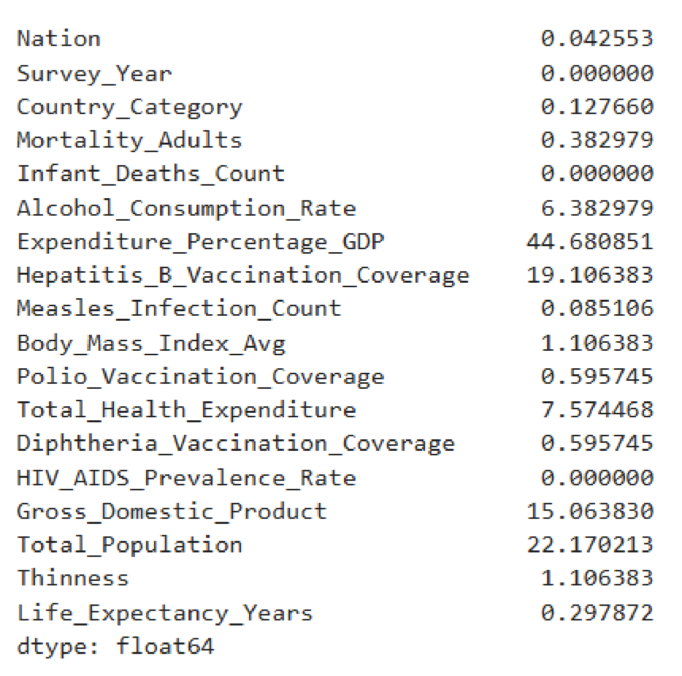


```
Nation                                1
Survey_Year                           0
Country_Category                      3
Mortality_Adults                      9
Infant_Deaths_Count                   0
Alcohol_Consumption_Rate            150
Expenditure_Percentage_GDP            0
Hepatitis_B_Vaccination_Coverage    449
Measles_Infection_Count               2
Body_Mass_Index_Avg                  26
Polio_Vaccination_Coverage           14
Total_Health_Expenditure            178
Diphtheria_Vaccination_Coverage      14
HIV_AIDS_Prevalence_Rate              0
Gross_Domestic_Product              354
Total_Population                    521
Thinness                             26
Life_Expectancy_Years                 7
dtype: int64
```

03 Data preprocessing

Handling wrong values

The Expenditure_percentage_GDP has Almost half of the values Non values So we dropped it



```
Nation                             0.042553
Survey_Year                        0.000000
Country_Category                   0.127660
Mortality_Adults                   0.382979
Infant_Deaths_Count                0.000000
Alcohol_Consumption_Rate           6.382979
Expenditure_Percentage_GDP        44.680851
Hepatitis_B_Vaccination_Coverage  19.106383
Measles_Infection_Count            0.085106
Body_Mass_Index_Avg                1.106383
Polio_Vaccination_Coverage         0.595745
Total_Health_Expenditure           7.574468
Diphtheria_Vaccination_Coverage    0.595745
HIV_AIDS_Prevalence_Rate           0.000000
Gross_Domestic_Product            15.063830
Total_Population                  22.170213
Thinness                           1.106383
Life_Expectancy_Years              0.297872
dtype: float64
```

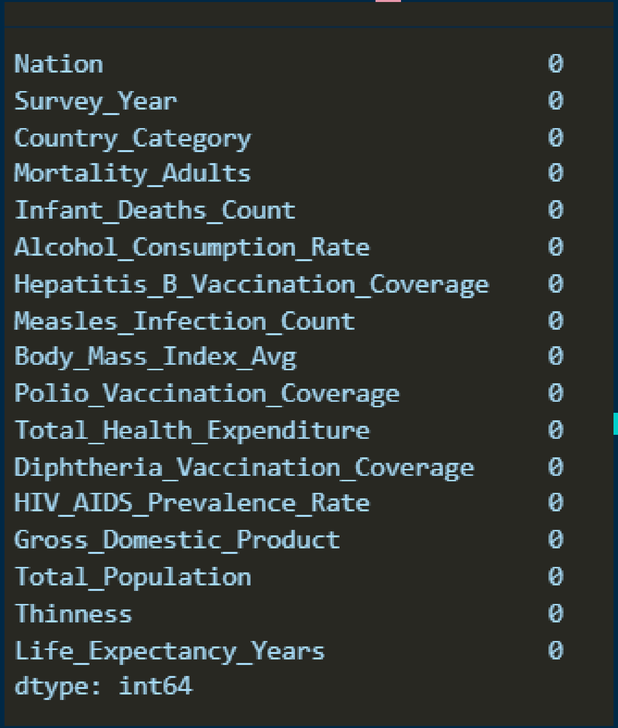
03 Data preprocessing

Replacing missing values

- Filled the missing values using pandas fillna with median

- We dropped the row with missing value in nation

- In country category there were some countries that hasn't been provided searched in the internet and filled it

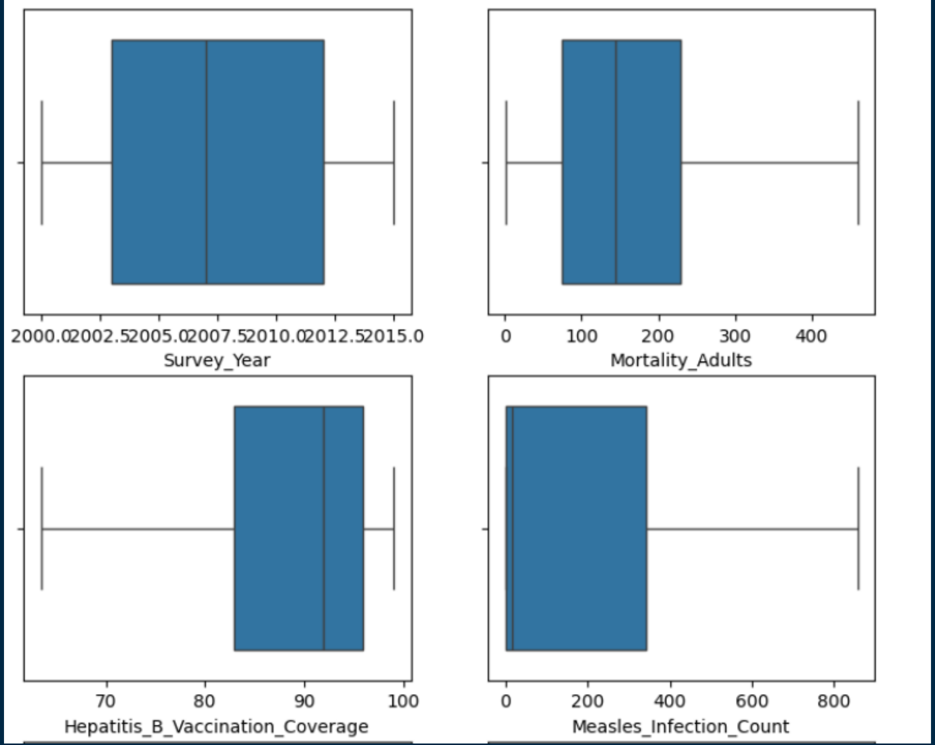


```
Nation                             0
Survey_Year                        0
Country_Category                   0
Mortality_Adults                   0
Infant_Deaths_Count                0
Alcohol_Consumption_Rate           0
Hepatitis_B_Vaccination_Coverage   0
Measles_Infection_Count            0
Body_Mass_Index_Avg                0
Polio_Vaccination_Coverage         0
Total_Health_Expenditure           0
Diphtheria_Vaccination_Coverage    0
HIV_AIDS_Prevalence_Rate           0
Gross_Domestic_Product             0
Total_Population                   0
Thinness                           0
Life_Expectancy_Years              0
dtype: int64
```

03 Data preprocessing

Handling outliers

- We used clip() func to remove data that is over first and third quartile
- Then we use sns.boxplot() for visualization



03

Data preprocessing

Handling skewness

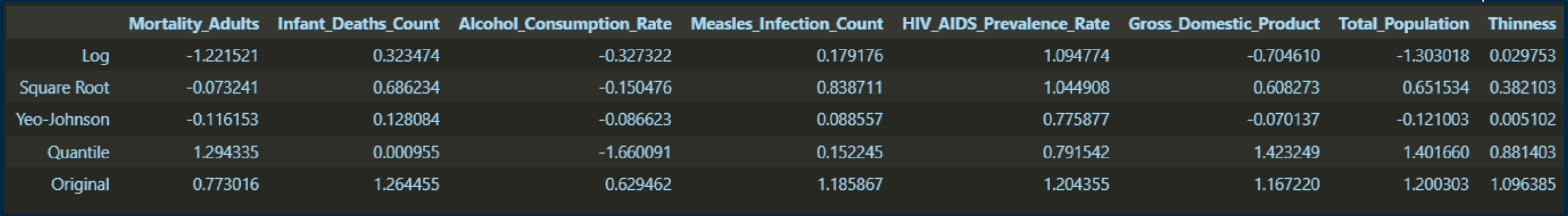
When we handle skewing, we must take care of positive skewness and negative skewness

03 Data preprocessing

Positive skewness

We have tested 4 transformers to decide which will have best result to solve positive skewness which is :

Yeo-Johnson



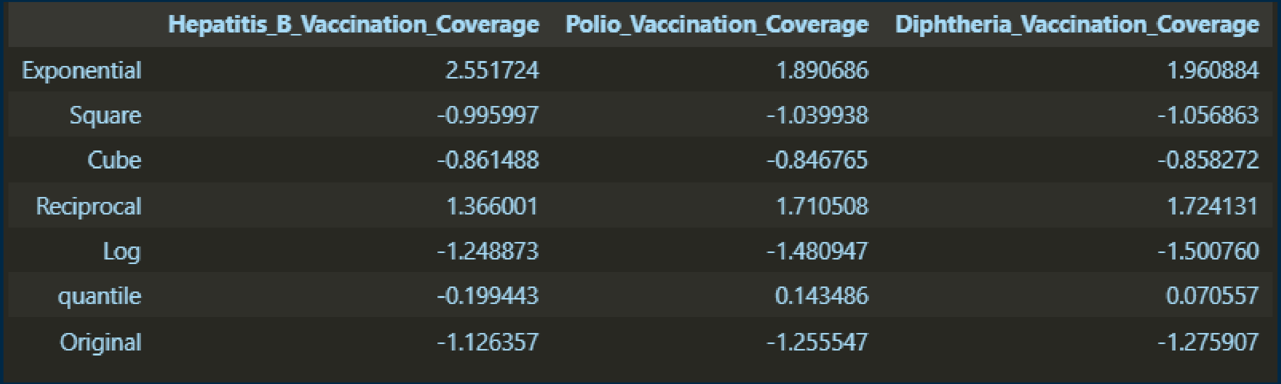
	Mortality_Adults	Infant_Deaths_Count	Alcohol_Consumption_Rate	Measles_Infection_Count	HIV_AIDS_Prevalence_Rate	Gross_Domestic_Product	Total_Population	Thinness
Log	-1.221521	0.323474	-0.327322	0.179176	1.094774	-0.704610	-1.303018	0.029753
Square Root	-0.073241	0.686234	-0.150476	0.838711	1.044908	0.608273	0.651534	0.382103
Yeo-Johnson	-0.116153	0.128084	-0.086623	0.088557	0.775877	-0.070137	-0.121003	0.005102
Quantile	1.294335	0.000955	-1.660091	0.152245	0.791542	1.423249	1.401660	0.881403
Original	0.773016	1.264455	0.629462	1.185867	1.204355	1.167220	1.200303	1.096385

03 Data preprocessing

Negative skewness

We have tested 6 transformers to decide which will have best result to solve negative skewness which is :

Square

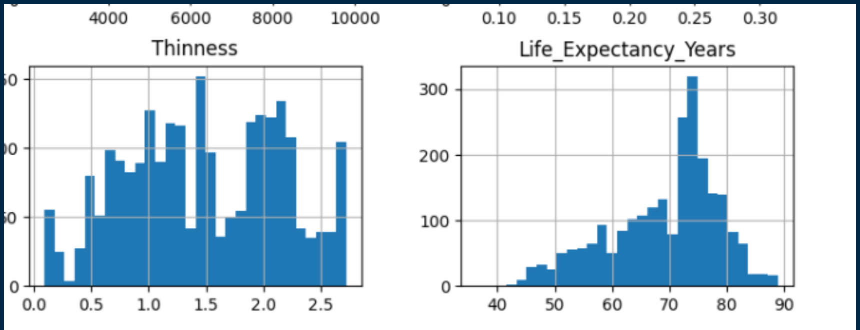


	Hepatitis_B_Vaccination_Coverage	Polio_Vaccination_Coverage	Diphtheria_Vaccination_Coverage
Exponential	2.551724	1.890686	1.960884
Square	-0.995997	-1.039938	-1.056863
Cube	-0.861488	-0.846765	-0.858272
Reciprocal	1.366001	1.710508	1.724131
Log	-1.248873	-1.480947	-1.500760
quantile	-0.199443	0.143486	0.070557
Original	-1.126357	-1.255547	-1.275907

03 Data preprocessing

Handling skewness Result

- After positive and negative skewness



Data analysis & visualization

Understanding relations between features

04 Data visulaization

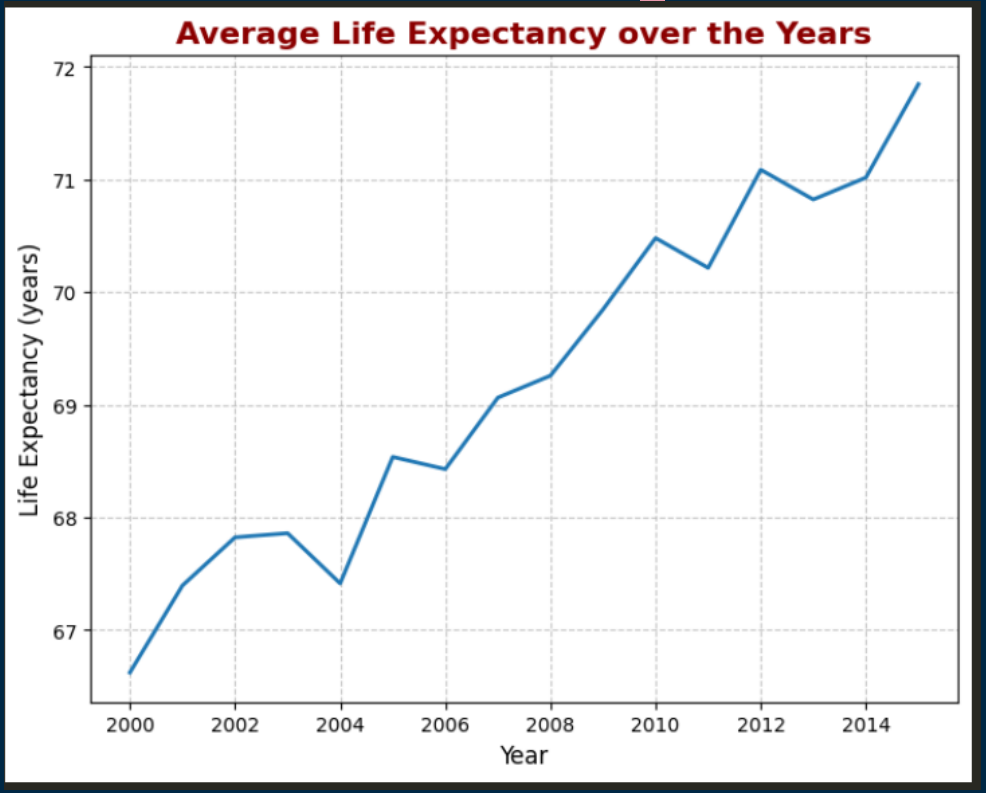
Analysis and Visualization:

- Data over years
- Sum Mortality Adults over the Years
- Developed vs Developing
- Correlation and multivariable analysis

04

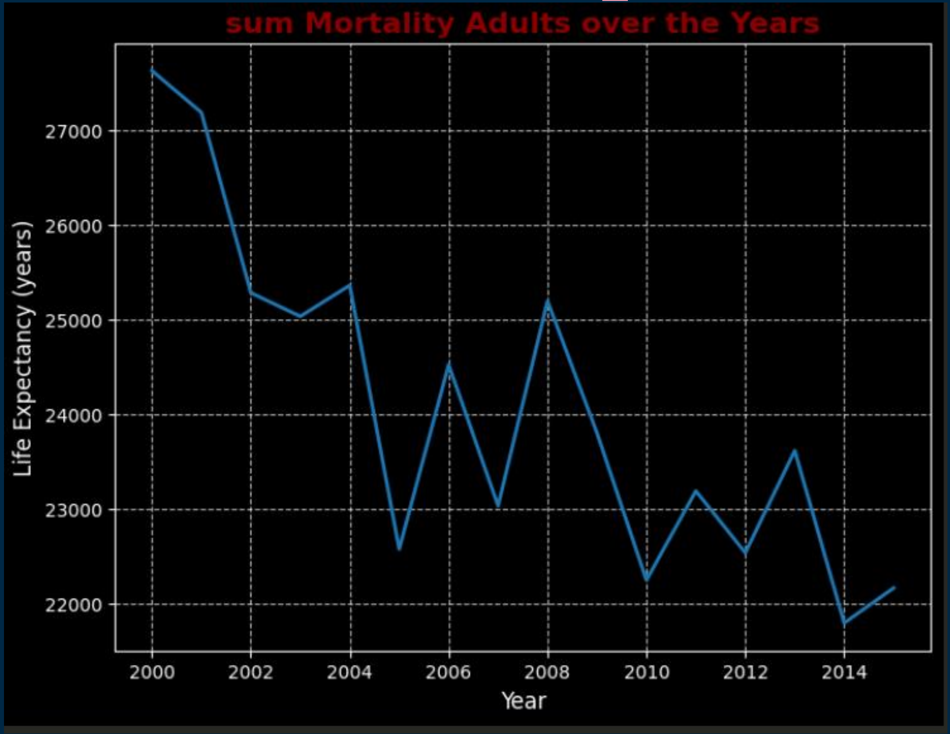
Data visulaization

Data average over years



04 Data visulaization

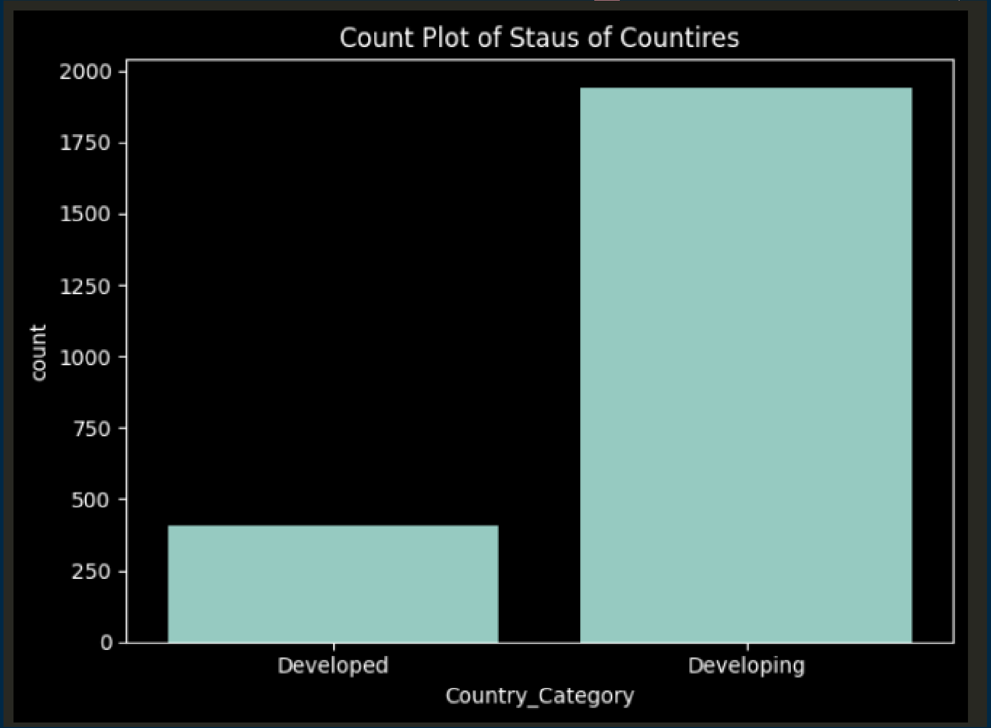
Sum Mortality Adults over the Years



04 Data visulaization

Developed vs Developing

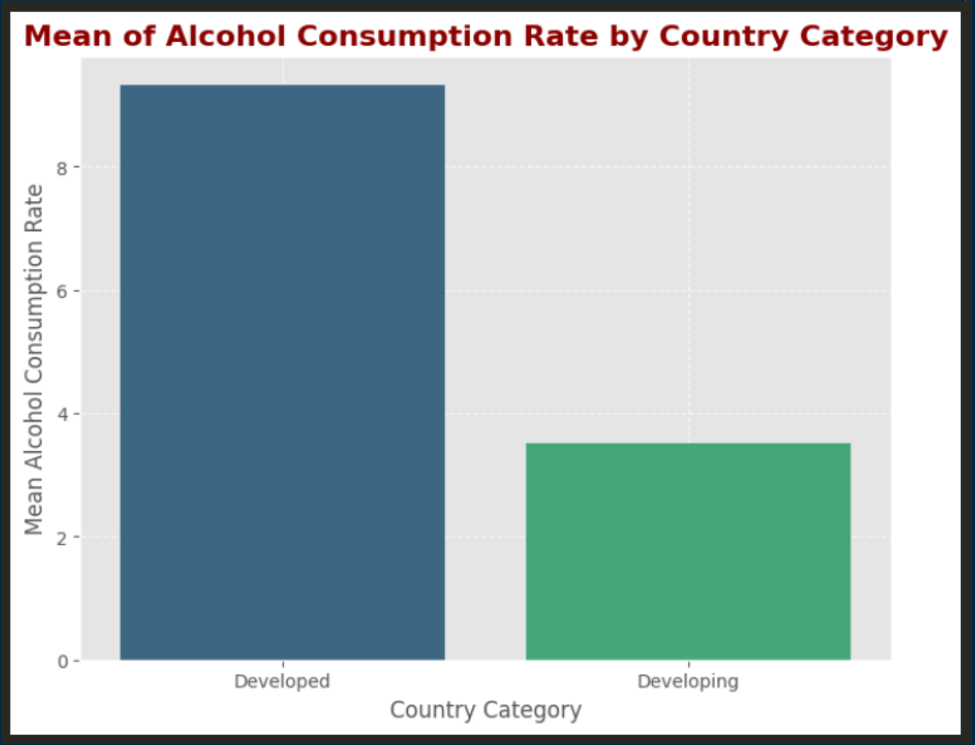
- Comparing by count of country_category



04 Data visulaization

Developed vs Developing

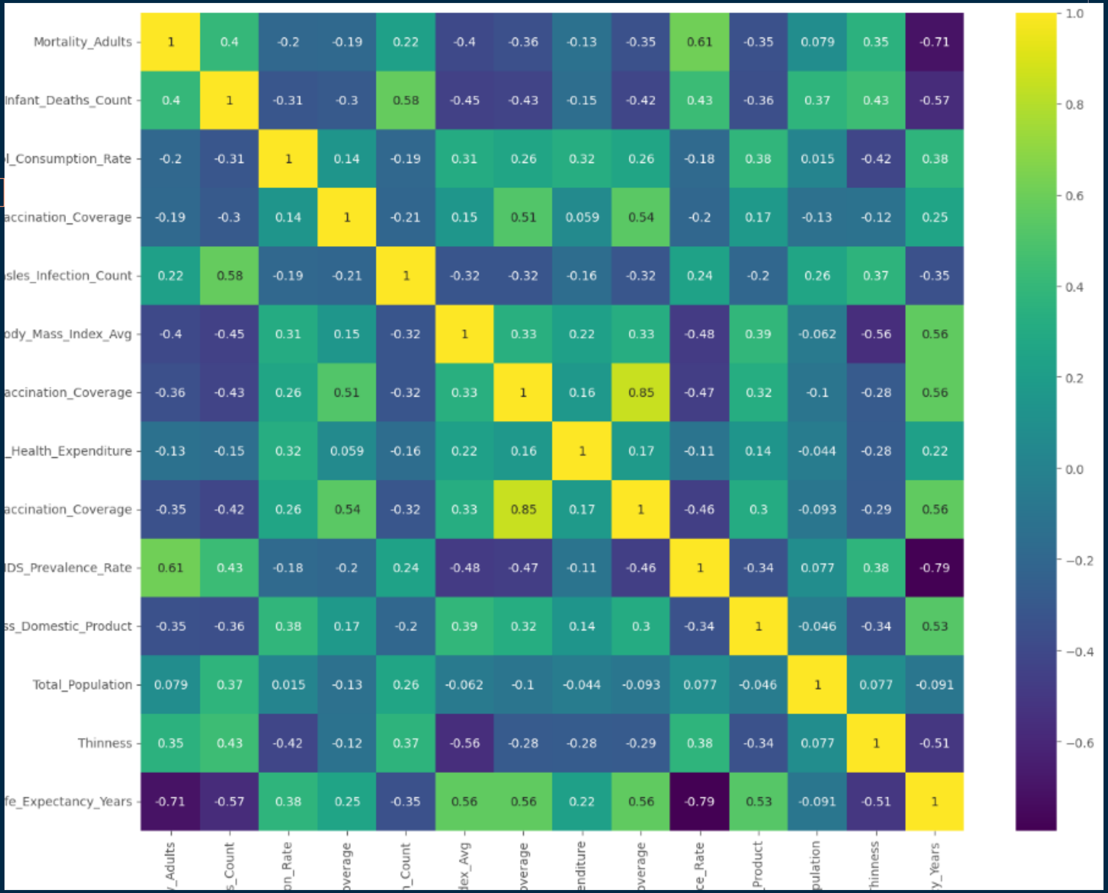
- **Mean of Alcohol Consumption Rate by Country Category**



04 Data visulaization

Correlation and multivariable analysis

- Correlation matrix to visualize correlation between columns and each other



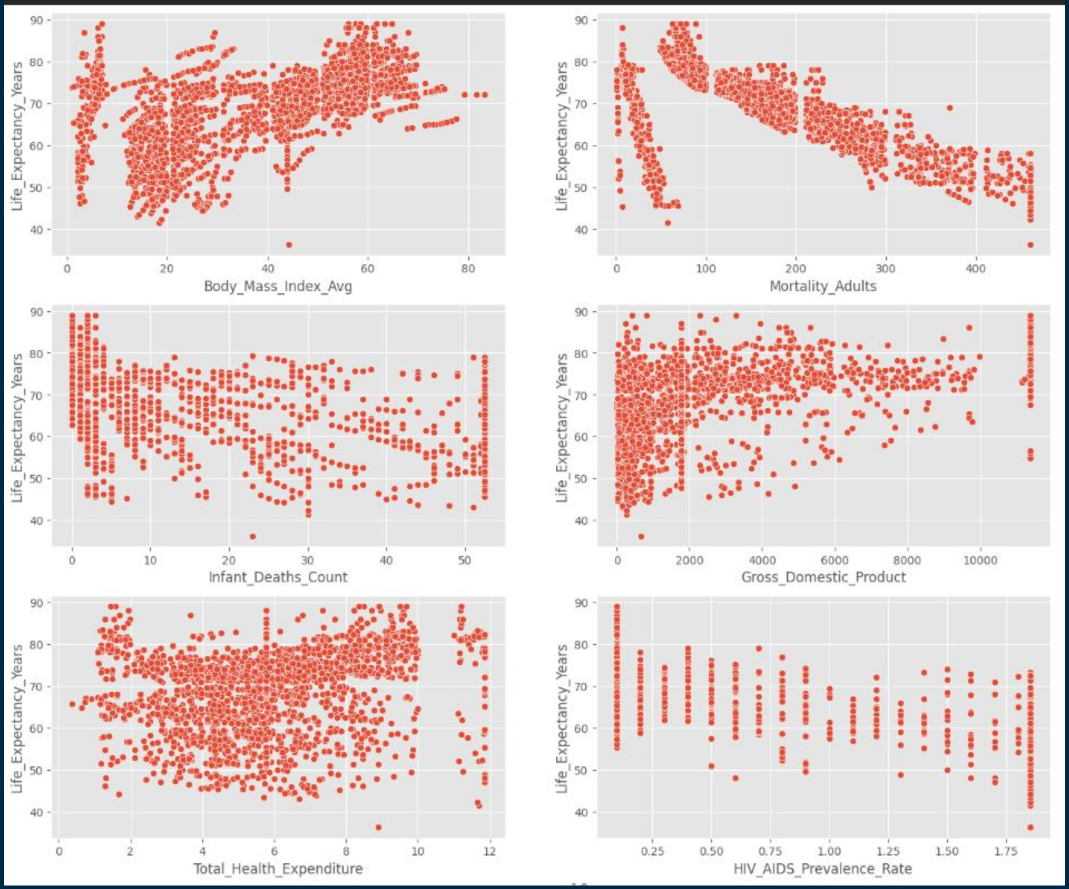
	y_Adults	s_Count	on_Rate	overage	n_Count	dex_Avg	overage	enditure	overage	ce_Rate	Product	pulation	hinness	y_Years
Mortality_Adults	1	0.4	-0.2	-0.19	0.22	-0.4	-0.36	-0.13	-0.35	0.61	-0.35	0.079	0.35	-0.71
Infant_Deaths_Count	0.4	1	-0.31	-0.3	0.58	-0.45	-0.43	-0.15	-0.42	0.43	-0.36	0.37	0.43	-0.57
l_Consumption_Rate	-0.2	-0.31	1	0.14	-0.19	0.31	0.26	0.32	0.26	-0.18	0.38	0.015	-0.42	0.38
accination_Coverage	-0.19	-0.3	0.14	1	-0.21	0.15	0.51	0.059	0.54	-0.2	0.17	-0.13	-0.12	0.25
sles_Infection_Count	0.22	0.58	-0.19	-0.21	1	-0.32	-0.32	-0.16	-0.32	0.24	-0.2	0.26	0.37	-0.35
ody_Mass_Index_Avg	-0.4	-0.45	0.31	0.15	-0.32	1	0.33	0.22	0.33	-0.48	0.39	-0.062	-0.56	0.56
accination_Coverage	-0.36	-0.43	0.26	0.51	-0.32	0.33	1	0.16	0.85	-0.47	0.32	-0.1	-0.28	0.56
_Health_Expenditure	-0.13	-0.15	0.32	0.059	-0.16	0.22	0.16	1	0.17	-0.11	0.14	-0.044	-0.28	0.22
accination_Coverage	-0.35	-0.42	0.26	0.54	-0.32	0.33	0.85	0.17	1	-0.46	0.3	-0.093	-0.29	0.56
IDS_Prevalence_Rate	0.61	0.43	-0.18	-0.2	0.24	-0.48	-0.47	-0.11	-0.46	1	-0.34	0.077	0.38	-0.79
ss_Domestic_Product	-0.35	-0.36	0.38	0.17	-0.2	0.39	0.32	0.14	0.3	-0.34	1	-0.046	-0.34	0.53
Total_Population	0.079	0.37	0.015	-0.13	0.26	-0.062	-0.1	-0.044	-0.093	0.077	-0.046	1	0.077	-0.091
Thinness	0.35	0.43	-0.42	-0.12	0.37	-0.56	-0.28	-0.28	-0.29	0.38	-0.34	0.077	1	-0.51
fe_Expectancy_Years	-0.71	-0.57	0.38	0.25	-0.35	0.56	0.56	0.22	0.56	-0.79	0.53	-0.091	-0.51	1

04

Data visulaization

Correlation and multivariable analysis

- Using scatterplot to show if there is any pattern in features with target column



Feature improvement

Encoding categorical and scaling the data

05 Feature improvment

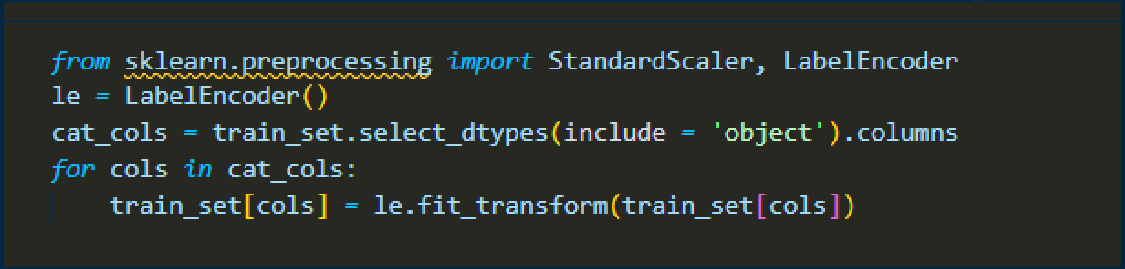
Improving features:

- Converting categorial into numerical
- Scaling data

05 Feature improvment

Converting categorial into numerical

- **Using label encoder to replace categorial data with numerical**

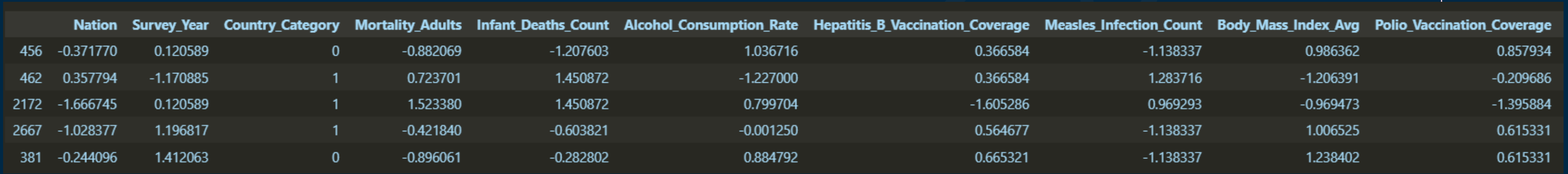


```
from sklearn.preprocessing import StandardScaler, LabelEncoder
le = LabelEncoder()
cat_cols = train_set.select_dtypes(include = 'object').columns
for cols in cat_cols:
    train_set[cols] = le.fit_transform(train_set[cols])
```

05 Feature improvment

Scaling data

- Using StandardScalar() to scale data



	Nation	Survey_Year	Country_Category	Mortality_Adults	Infant_Deaths_Count	Alcohol_Consumption_Rate	Hepatitis_B_Vaccination_Coverage	Measles_Infection_Count	Body_Mass_Index_Avg	Polio_Vaccination_Coverage
456	-0.371770	0.120589	0	-0.882069	-1.207603	1.036716	0.366584	-1.138337	0.986362	0.857934
462	0.357794	-1.170885	1	0.723701	1.450872	-1.227000	0.366584	1.283716	-1.206391	-0.209686
2172	-1.666745	0.120589	1	1.523380	1.450872	0.799704	-1.605286	0.969293	-0.969473	-1.395884
2667	-1.028377	1.196817	1	-0.421840	-0.603821	-0.001250	0.564677	-1.138337	1.006525	0.615331
381	-0.244096	1.412063	0	-0.896061	-0.282802	0.884792	0.665321	-1.138337	1.238402	0.615331

Models training

Train 10 different models

05 Models training

We start with training 10 models which are:

- **Random Forest regressor**
- **Extra Tree regressor**
- **Gradientboost regressor**
- **XGB regressor**
- **Linear regression**
- **Polynomial regression (1 to 5) degree**

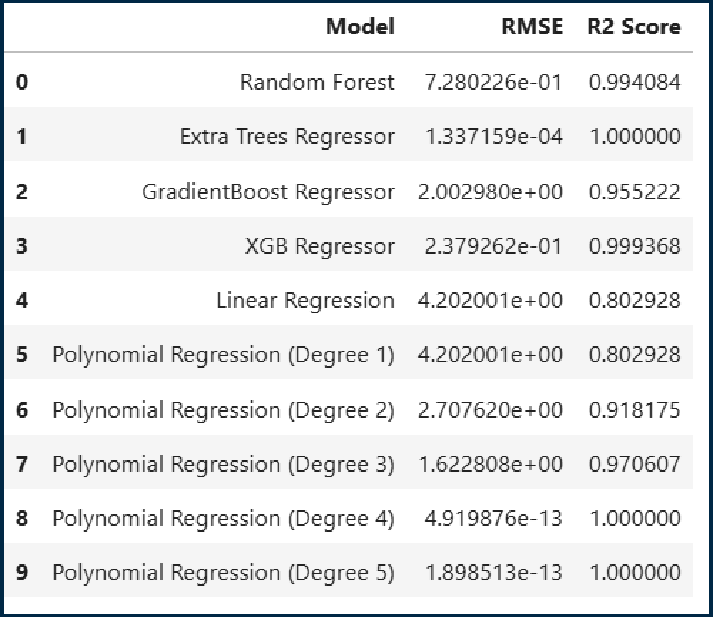
Models Evaluation

Choosing the best model

06

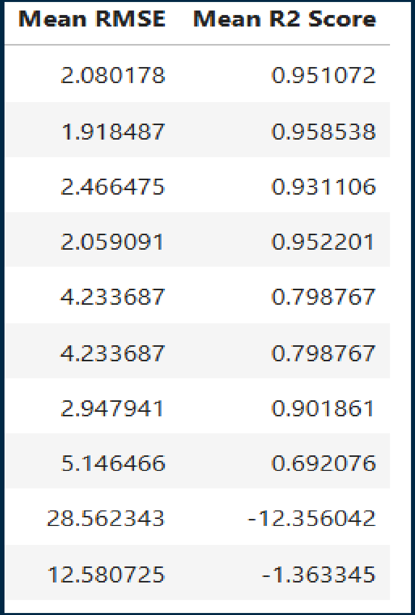
Compare between models Performance on the training And cross Validation

Training



	Model	RMSE	R2 Score
0	Random Forest	7.280226e-01	0.994084
1	Extra Trees Regressor	1.337159e-04	1.000000
2	GradientBoost Regressor	2.002980e+00	0.955222
3	XGB Regressor	2.379262e-01	0.999368
4	Linear Regression	4.202001e+00	0.802928
5	Polynomial Regression (Degree 1)	4.202001e+00	0.802928
6	Polynomial Regression (Degree 2)	2.707620e+00	0.918175
7	Polynomial Regression (Degree 3)	1.622808e+00	0.970607
8	Polynomial Regression (Degree 4)	4.919876e-13	1.000000
9	Polynomial Regression (Degree 5)	1.898513e-13	1.000000

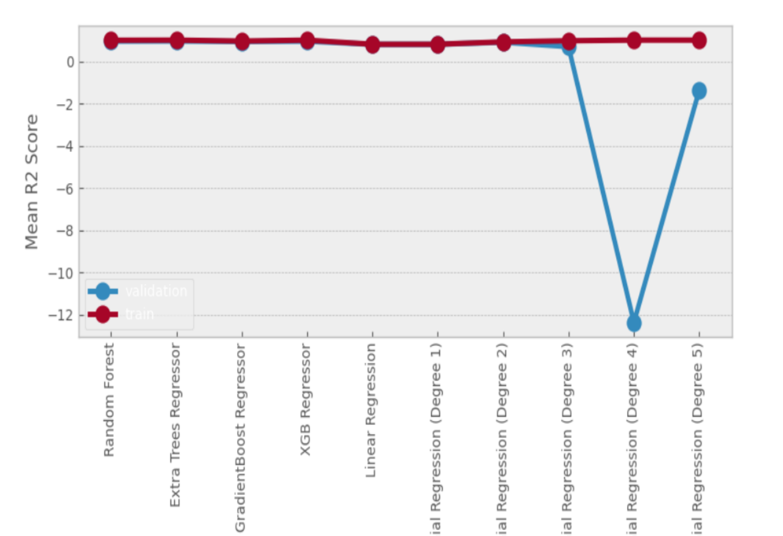
Cross Val

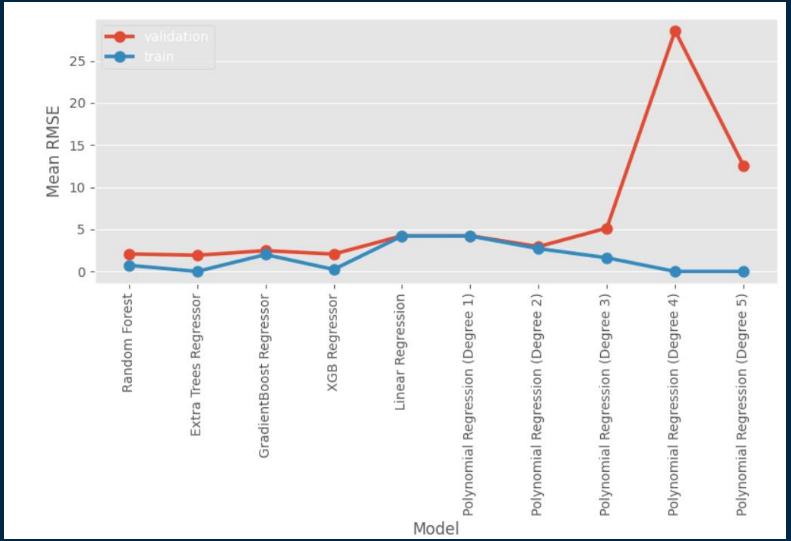


Mean RMSE	Mean R2 Score
2.080178	0.951072
1.918487	0.958538
2.466475	0.931106
2.059091	0.952201
4.233687	0.798767
4.233687	0.798767
2.947941	0.901861
5.146466	0.692076
28.562343	-12.356042
12.580725	-1.363345

06 Models Evaluation

Validation 🟥
Training 🟦





06 Models Evaluation

From the evaluation :

some models show a high performance on both Training and Validation

Random Forest
XGBoost

Which we can choose between them

06 Models Evaluation

From the evaluation :

Other models perform bad on both the sets

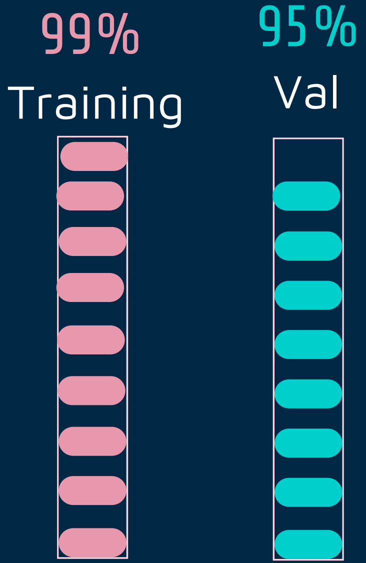
- Linear Regression
- indication for Underfitting

Others perform very well on the training but performance Decrees on the validation

Higher order polynomials

Random Forest

Would be the Chosen model as it is the less shown overfitting



Pipeline

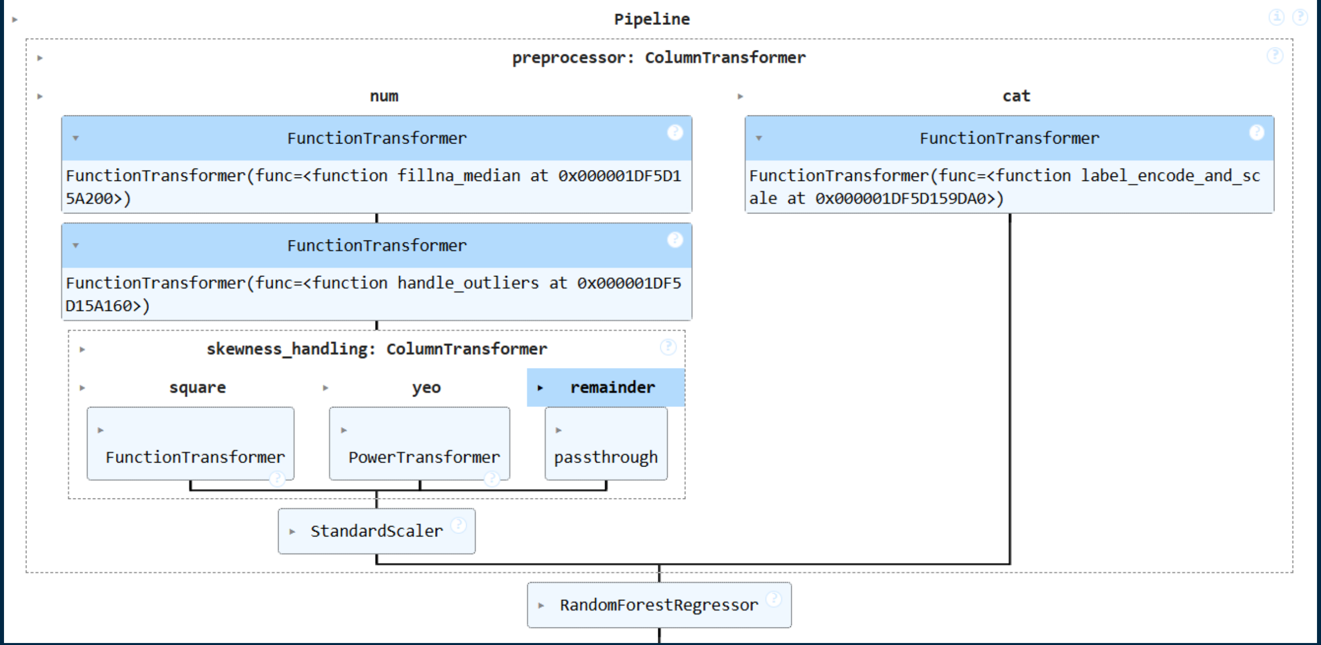
ALL IN ONE

07 Pipeline

Compress all the preprocessing , Cleaning , Transformation, Scaling, encoding, modeling and Predicting in one function

Takes row data as input and return prediction in one shot

07 Pipeline



Pipeline

preprocessor: ColumnTransformer

num

FunctionTransformer

```
FunctionTransformer(func=<function fillna_median at 0x000001DF5D15A200>)
```

FunctionTransformer

```
FunctionTransformer(func=<function handle_outliers at 0x000001DF5D15A160>)
```

skewness_handling: ColumnTransformer

square

FunctionTransformer

yeo

PowerTransformer

remainder

passthrough

StandardScaler

cat

FunctionTransformer

```
FunctionTransformer(func=<function label_encode_and_scale at 0x000001DF5D159DA0>)
```

RandomForestRegressor

Hyperparameter Finetuning

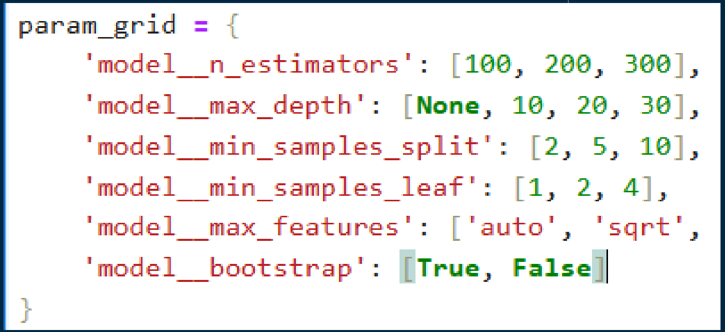
Trying to finetune the performance

08 Hyperparameter finetuning

Using grid search we tried manually to set a combination of parameters

We fitted 5 folds for each 1920 Candidates which is overall number of Fits equal to 9400 fit

Unfortunately, the RMSE still the same



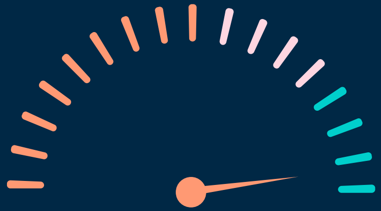
```
param_grid = {
    'model__n_estimators': [100, 200, 300],
    'model__max_depth': [None, 10, 20, 30],
    'model__min_samples_split': [2, 5, 10],
    'model__min_samples_leaf': [1, 2, 4],
    'model__max_features': ['auto', 'sqrt',
    'model__bootstrap': [True, False]
}
```

Test Set

The Final Evaluation

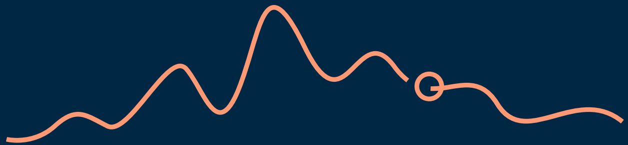
09 Test Set

As we have a **pipeline**, we passed the test set to it
And it gives us the **predictions directly**



R2 Score

0.95



RMSE

2.14

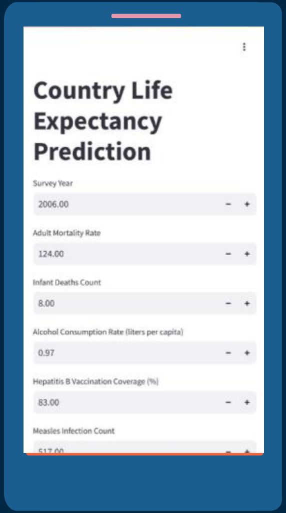
Model Deployment

Lunch the model into production

08 Saving & Deployment

We saved the model using pickle module
And then deployed it using Streamlit

We created a website for the naïve user to enter the raw data and get the prediction



Country Life Expectancy Prediction

Survey Year: 2006.00

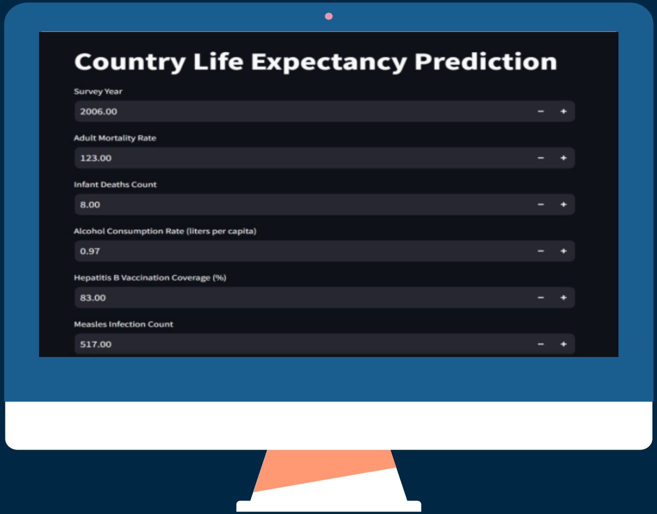
Adult Mortality Rate: 124.00

Infant Deaths Count: 8.00

Alcohol Consumption Rate (liters per capita): 0.97

Hepatitis B Vaccination Coverage (%): 83.00

Measles Infection Count: 517.00



Country Life Expectancy Prediction

Survey Year: 2006.00

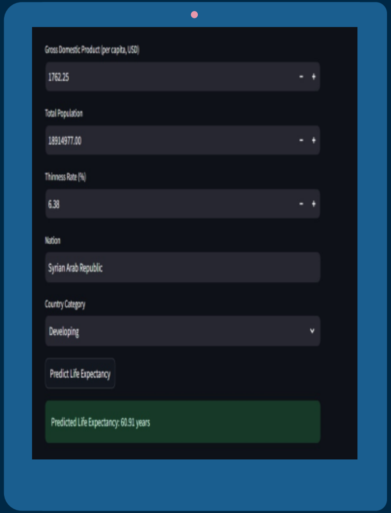
Adult Mortality Rate: 123.00

Infant Deaths Count: 8.00

Alcohol Consumption Rate (liters per capita): 0.97

Hepatitis B Vaccination Coverage (%): 83.00

Measles Infection Count: 517.00



Gross Domestic Product (per capita, USD): 1762.25

Total Population: 18914977.00

Thinness Rate (%): 6.38

Nation: Syrian Arab Republic

Country Category: Developing

Predicted Life Expectancy: 60.91 years

THANKS

Ahmed Mostafa Gamal Eldein