Project Report Group 9

Introduction of the data set

The source of the data is from the Global Health Observatory (GHO) data repository under the World Health Organization (WHO). It keeps track of the health status as well as many other related factors for all countries. The dataset 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 the 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 the 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.

Objective

Explore the trends in global life expectancy from 2000 to 2015.Identify countries with significant improvements or declines in life expectancy.Analyze the factors contributing to variations in life expectancy among different countries.Investigate the impact of socio-economic factors and healthcare policies on life expectancy trends.The project attempts to survey life expectancy and mortality rates across multiple countries, gauge a range of indicators that could potentially influence life expectancy, attain an insight into the factors controlling these values, majorly using Regression concepts to draw our inferences.Purpose of the dataset is tracking health status and related factors for different countries.

Dataset Overview

Description of the dataset: Life expectancy and health factors for 193 countries.
Selection criteria: Critical factors chosen for analysis.
Timeframe: Data from 2000 to 2015 selected for analysis.
Total rows and columns in the dataset: 2938 rows and 22 columns.

Target variable

From the dataset, there are various target variables such as life expectancy, adult mortality, infant deaths, etc. The target variable represents what we are attempting to predict. It is a dependent variable that can be deduced based on other variables.

The target variable might vary based on the context or the objective of the analysis. If we're studying factors influencing infant mortality rates, then "infant deaths" or "under-five deaths" would be our target variable. For this analysis, we are interested in predicting life expectancy. So "life expectancy" is the target variable for this study.

Categorical and continuous variables

The dataset has 2,938 observations (rows) and 22 features (columns) that can be categorized as below.

Column	Description	Categories
Country	Represents different countries	
Year	Represents different years. It could be treated as ordinal or nominal.	
Adult Mortality	Rates of death among adults	**Mortality**
infant deaths	Number of infant deaths.	
HIV/AIDS	Rates of HIV/AIDS prevalence.	
thinness 1-19 years	Prevalence of thinness among 10-19-year-olds.	
thinness 5-9 years	Prevalence of thinness among 5-9-year-olds.	
under-five deaths	Number of deaths under the age of five.	
Hepatitis B	Hepatitis B immunization coverage.	**Immunization related**
Measles	Number of reported measles cases.	
Polio	Polio immunization coverage.	
Diphtheria	Diphtheria immunization coverage.	
percentage expenditure	Health expenditure as a percentage of GDP.	**Economical**
Total expenditure	Total health expenditure as a percentage of GDP.	
GDP	A measure of a country's economic performance.	
Population	Total population count.	
Income composition of resources	A composite index representing income composition in a country.	
Alcohol	Alcohol consumption, possibly measured in liters per capita.	**Social**
BMI	Body Mass Index, a measure of body fat based on height and weight.	
Schooling	Average number of years of schooling.	
Status	The economic status of a country (Developed, Developing).	
Life expectancy	Represents the average lifespan	

Summary statistics of numerical columns

These statistics provide an overview of the central tendency, variability, and range of values for each numerical column in the dataset.

Univariate analysis

Missing values and Imputation

```
Country                            0.000
Year                               0.000
Status                             0.000
Life expectancy                    0.340
Adult Mortality                    0.340
infant deaths                      0.000
Alcohol                            6.603
percentage expenditure             0.000
Hepatitis B                       18.822
Measles                            0.000
 BMI                               1.157
under-five deaths                  0.000
Polio                              0.647
Total expenditure                  7.692
Diphtheria                         0.647
 HIV/AIDS                          0.000
GDP                               15.248
Population                        22.192
 thinness  1-19 years              1.157
 thinness 5-9 years                1.157
Income composition of resources    5.684
Schooling                          5.548
dtype: float64
```

Imputation choices based on categories :

- **Mean:** Imputing missing values with the mean is a common strategy for continuous variables - all numerical columns without much skew in this case. It maintains the overall average of the dataset, which might help in preserving the original distribution.
- **Median:** When data is skewed or contains outliers, the median can be a more robust measure of central tendency than the mean. It's less affected by extreme values.
- **Mode:** For Country, Year, Status that are categorical / discrete, imputing with the mode (most frequent value) is reasonable. It preserves the most common category, which can be representative of the missing values.

The code identifies potential outliers in each numerical column of the dataset using the Interquartile Range (IQR) method. Here's a summary of the identified potential outliers in each column:

- 'Adult Mortality': 15 potential outliers ranging from 599 to 717.
- 'Infant Deaths': 102 potential outliers with values ranging across the set.
- 'Percentage Expenditure': 181 potential outliers.

- 'Hepatitis B': 116 potential outliers.
- 'Measles': 183 potential outliers.
- 'Under-five Deaths': 94 potential outliers.
- 'Polio': 88 potential outliers.
- 'Total Expenditure': 6 potential outliers.
- 'Diphtheria': 98 potential outliers.
- 'HIV/AIDS': 153 potential outliers.
- 'GDP': 144 potential outliers.
- 'Population': 135 potential outliers.
- 'Thinness 1-19 Years': 45 potential outliers.
- 'Thinness 5-9 Years': 54 potential outliers.
- 'Schooling': 7 potential outliers.

It's important to assess these outliers carefully as they might represent extreme or influential data points that could affect the analysis or modeling. Depending on the context and domain knowledge, further investigation or treatment of these outliers might be necessary to ensure the integrity of the dataset and the validity of subsequent analyses.

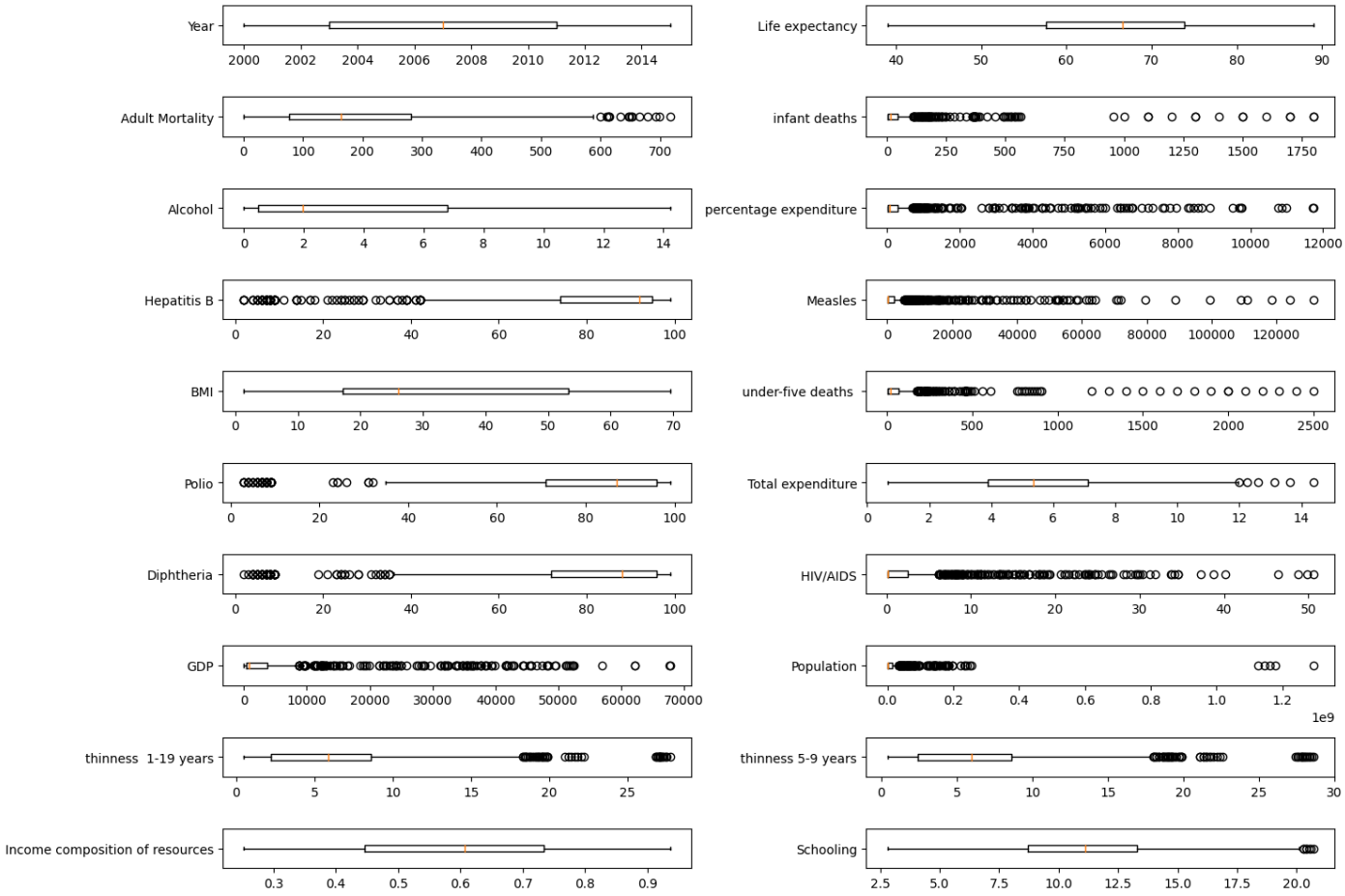
Identification of outliers

The code identifies potential outliers in each numerical column of the dataset using the Interquartile Range (IQR) method. Here's a summary of the identified potential outliers in each column:

- 'Adult Mortality': 15 potential outliers ranging from 599 to 717.
- 'Infant Deaths': 102 potential outliers with values ranging across the set.
- 'Percentage Expenditure': 181 potential outliers.
- 'Hepatitis B': 116 potential outliers.
- 'Measles': 183 potential outliers.
- 'Under-five Deaths': 94 potential outliers.
- 'Polio': 88 potential outliers.
- 'Total Expenditure': 6 potential outliers.
- 'Diphtheria': 98 potential outliers.
- 'HIV/AIDS': 153 potential outliers.
- 'GDP': 144 potential outliers.
- 'Population': 135 potential outliers.
- 'Thinness 1-19 Years': 45 potential outliers.
- 'Thinness 5-9 Years': 54 potential outliers.
- 'Schooling': 7 potential outliers.

It's important to assess these outliers carefully as they might represent extreme or influential data points that could affect the analysis or modeling. Depending on the context and domain knowledge, further investigation or treatment of these outliers might be necessary to ensure the integrity of the dataset and the validity of subsequent analyses.

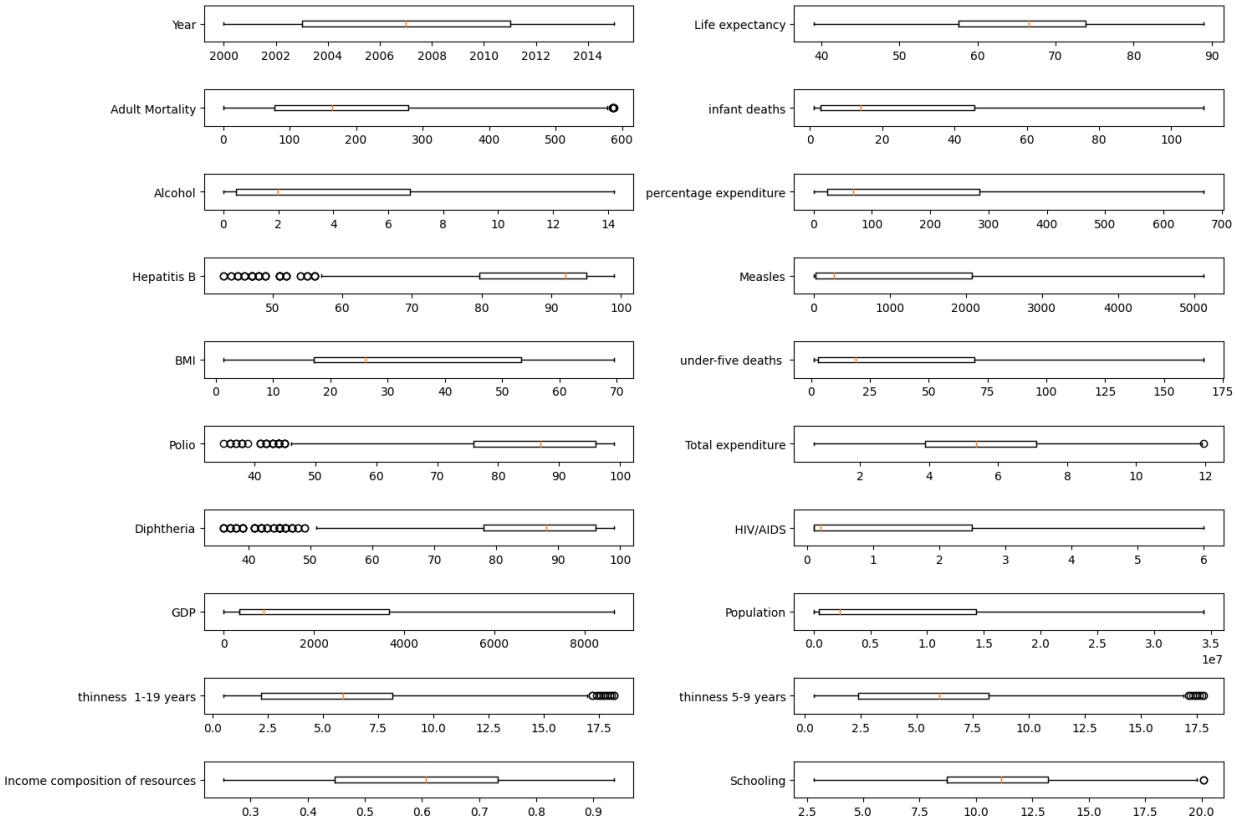
Outliers of each feature are represented visually by the following chart.



Imputation of outliers

The code detects outliers in various columns of the dataset and then replaces those outliers with the mean (for numerical columns) or the mode (for categorical columns).

Outlier representation of the features after mean imputation -



Distribution

Distribution of numerical data :

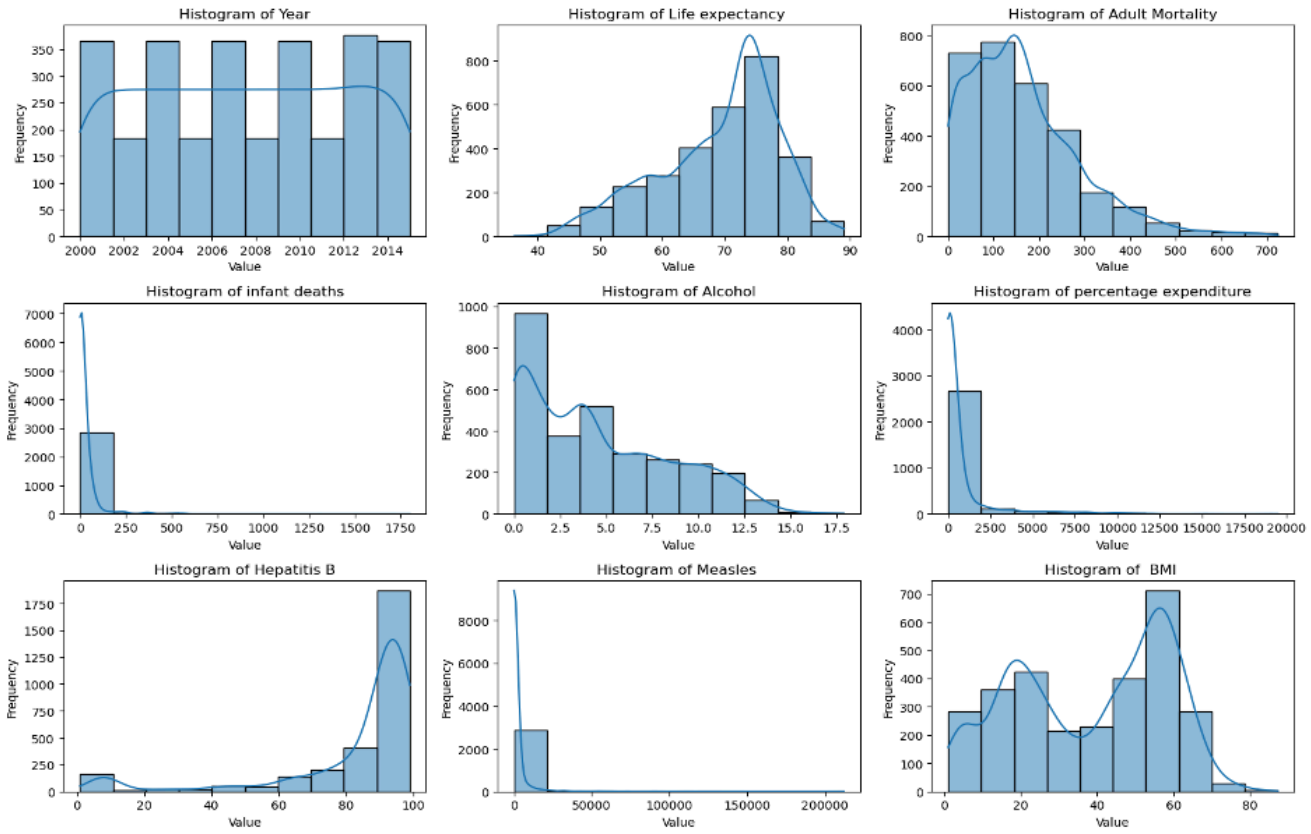
The distribution of the numerical data represents the spread and central tendency of the values.A histogram was used to visualize the distribution of numerical values. The histogram illustrates the frequency of the numerical values within different intervals or bins.There are various types of distributions in statistics, each with its own characteristics and applications. Here are some common ones:

1. Normal Distribution (Gaussian Distribution): A bell-shaped symmetrical distribution characterized by its mean and standard deviation. Many natural phenomena follow this distribution, like heights or IQ scores.
2. Binomial Distribution: Used to model the number of successes in a fixed number of independent Bernoulli trials, where each trial has the same probability of success.

3. Poisson Distribution: Models the number of events occurring within a fixed interval of time or space under certain conditions, such as rare events happening independently of each other.
4. Exponential Distribution: Describes the time between events in a Poisson process, where events occur continuously and independently at a constant average rate.
5. Uniform Distribution: All outcomes are equally likely within a certain range, with a constant probability density function.

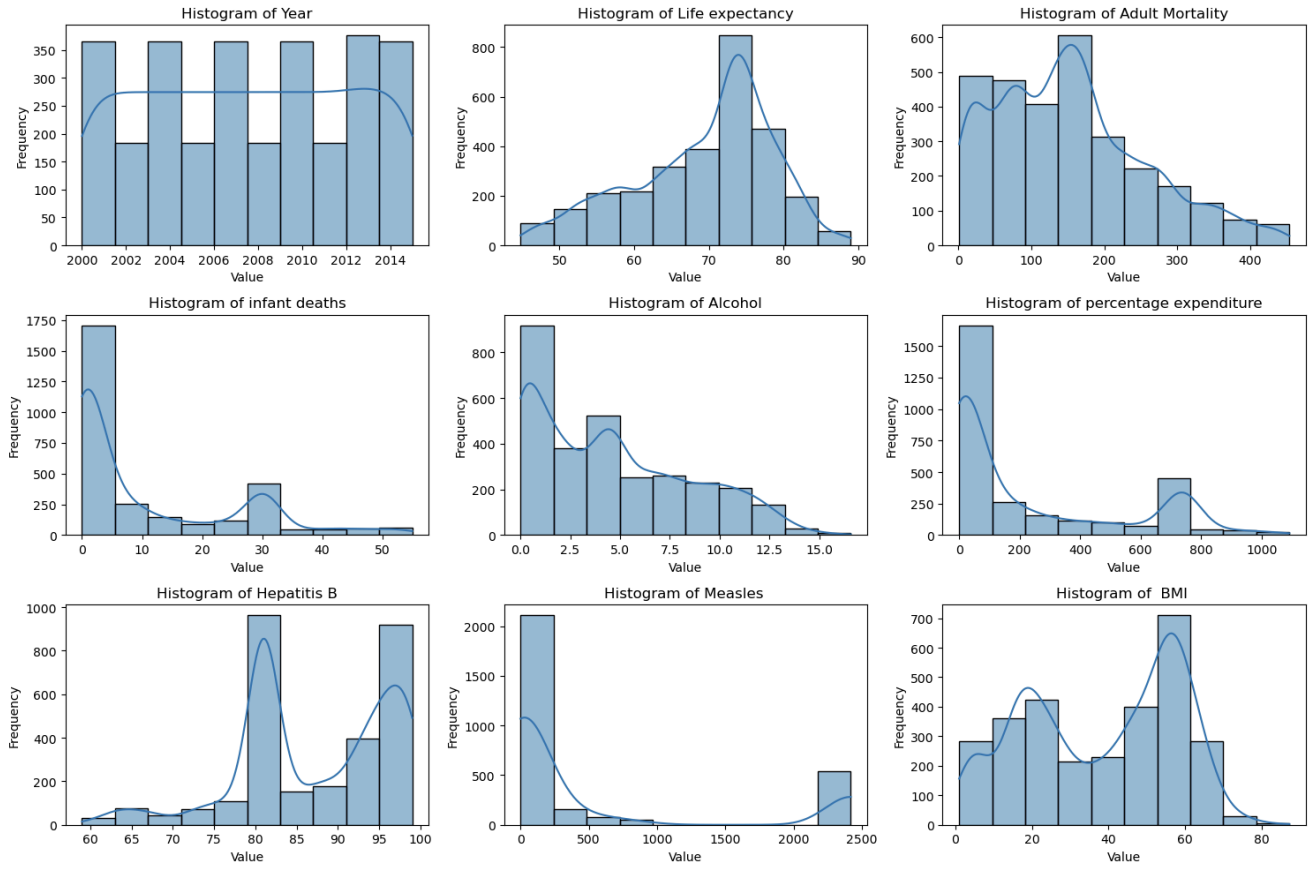
Distribution before imputation:

The distribution of the data appears to be skewed with longer tails and peaks.



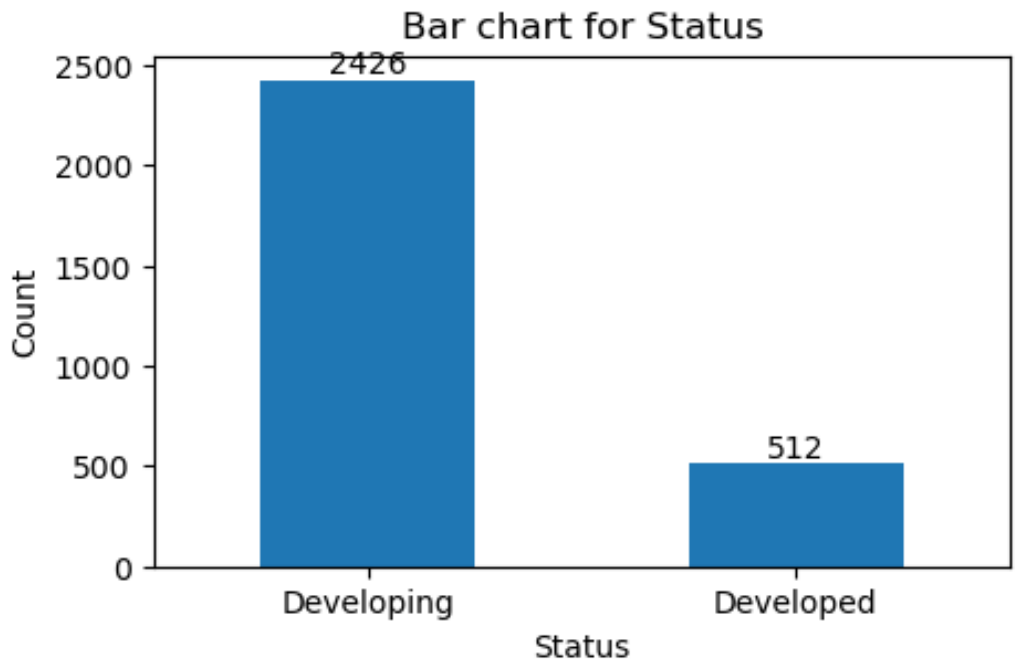
Distribution after imputation:

The skewness has decreed after performing mean imputation.



Distribution of Categorical data:

The distribution of categorical data refers to how the different categories or levels of a variable are distributed within a dataset. Unlike numerical data, which can take on a range of values, categorical data consists of distinct categories or groups.We can represent the distribution of categorical data using frequency tables, bar charts, pie charts, or histograms. These visualizations allow you to see the frequency or proportion of observations within each category. Analyzing the distribution of categorical data can provide insights into patterns, preferences, or trends within a dataset. It's often a crucial step in exploratory data analysis and can help guide further analysis or decision-making processes.

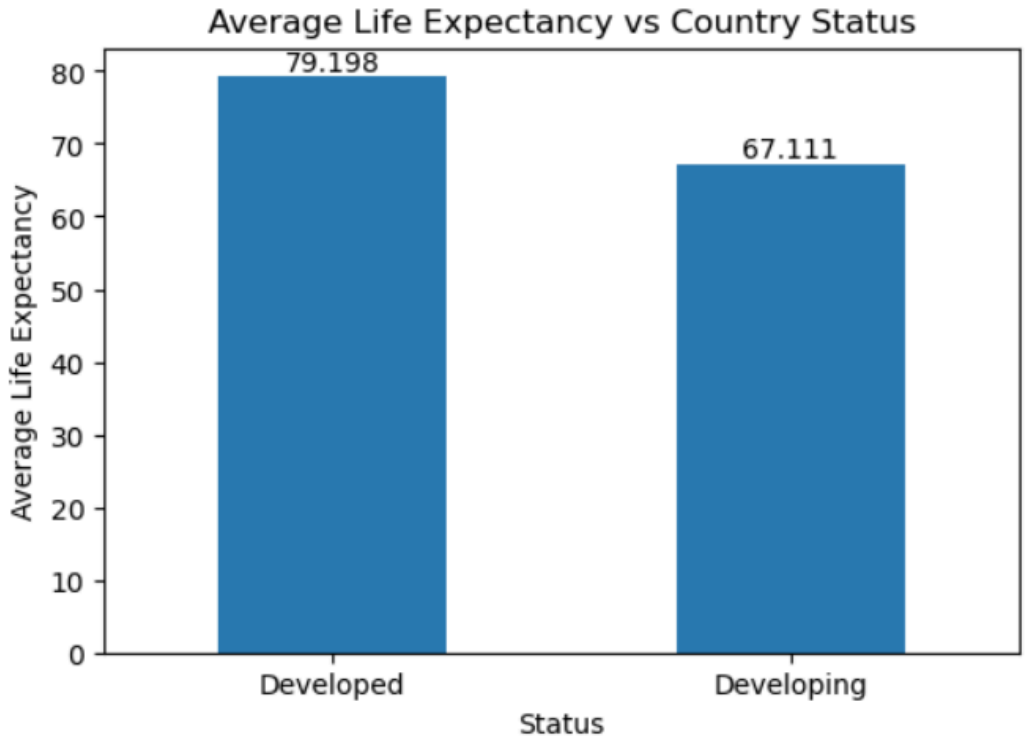


Bivariate Analysis

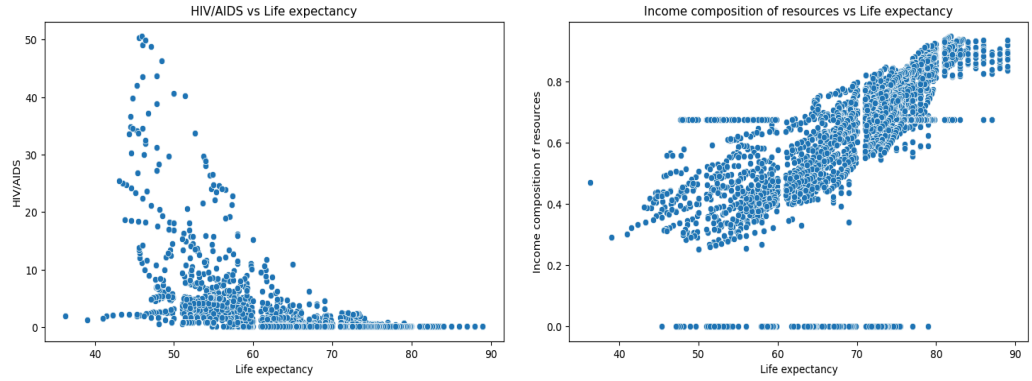
Bivariate analysis involves the analysis of two variables simultaneously to determine if there is a relationship between them. This analysis helps to understand the patterns, correlations, or associations between two variables.

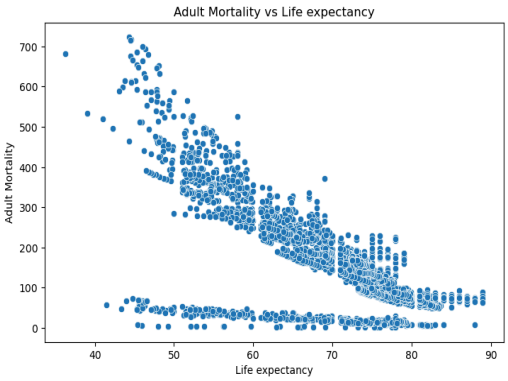
Scatter Plot: A scatter plot is a graphical representation that shows the relationship between two variables by plotting data points on a Cartesian plane. It's useful for visualizing the pattern or trend between two continuous variables.

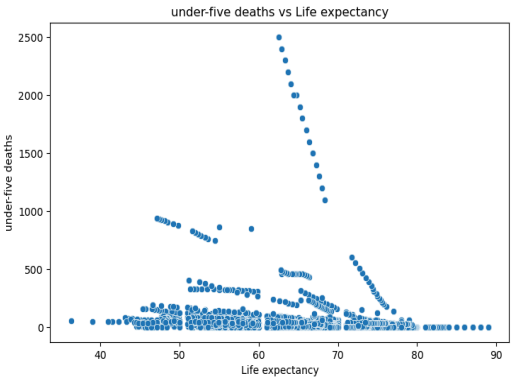
Bar chart representing the relationship between Life expectancy and the country status.Scatterplot

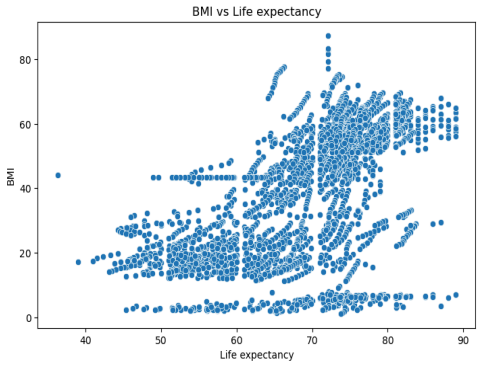


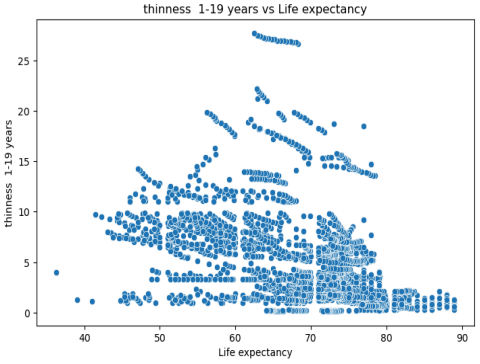
Scatterplot representing the relationship between life expectancy and other important features.

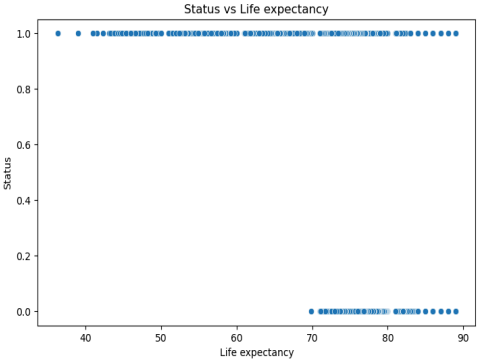












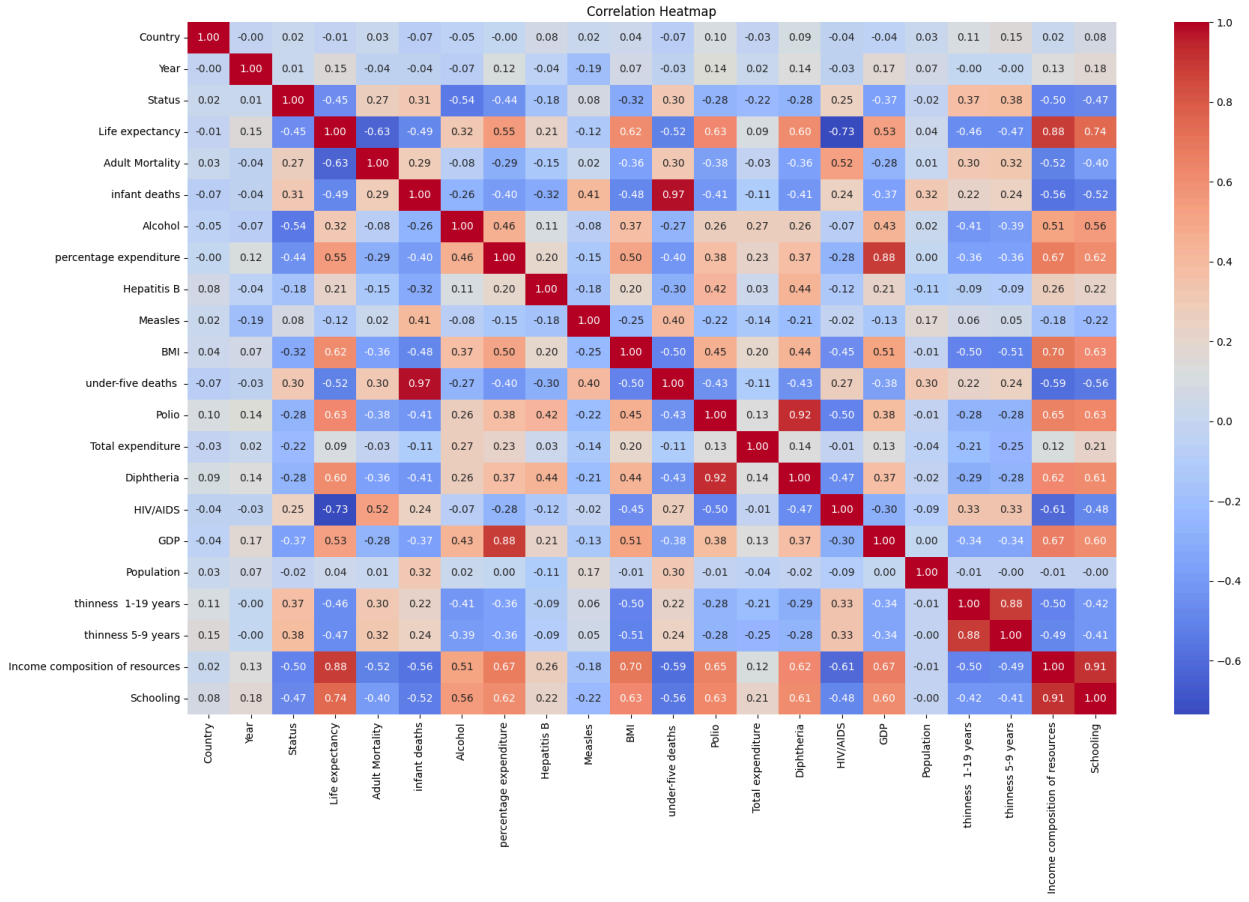
Multivariate Analysis

Correlation Matrix

The correlation matrix and heatmap provide insights into the relationships between different features in the dataset. From the correlation matrix obtained, here are some insights :

(a) The Features which are Most Positively Correlated with the target variable ('Life expectancy') are: ***Income composition of resources*** and ***Schooling***. These features show the highest positive correlation coefficients of approximately 0.88 and 0.74, respectively, with 'Life expectancy'. This indicates that as 'Income composition of resources' and 'Schooling' increase, 'Life expectancy' tends to increase as well.

(b) The Feature which is Most Negatively Correlated with the target variable ('Life expectancy') is: ***HIV/AIDS***. This feature exhibits the highest negative correlation coefficient of approximately -0.73 with 'Life expectancy'. As the 'HIV/AIDS' rate increases, 'Life expectancy' tends to decrease.



The heatmap provides a visual representation of these correlations, where darker shades indicate stronger correlations (either positive or negative), and lighter shades indicate weaker correlations or no correlation.

Visualization

https://public.tableau.com/views/DATA_230_Project/Story1?:language=en-US&publish=yes&:sid=&:display_count=n&:origin=viz_share_link

Transformation

Data transformation refers to the process of converting or altering the original features in a dataset to make them more suitable for analysis or modeling. This process can involve various techniques such as scaling, normalization, logarithmic transformations, and polynomial transformations.

Normalization: Normalization rescales the feature values to have a mean of 0 and a standard deviation of 1. This transformation is particularly useful when the distribution of feature values is skewed or non-normal, as it helps improve the numerical stability of algorithms and makes them less sensitive to outliers.

Label Encoding: Label encoding is the process of converting categorical data into a numerical format so that it can be used as input for machine learning algorithms. Categorical data represents qualitative variables with discrete categories or levels. However, regression algorithms require numerical input, which means that categorical data must be transformed into a numerical label representation before it can be used for modeling. In label encoding, each unique category in a categorical variable is assigned a unique numerical label.

Winsorization: Winsorizing is a data preprocessing technique used to mitigate the influence of outliers in a dataset. It involves replacing extreme values (outliers) in the dataset with less extreme values, typically the nearest non-outlier values. Winsorizing is performed by setting a threshold value (also known as a "trimming" value) beyond which data points are considered outliers.

Proposed Solution

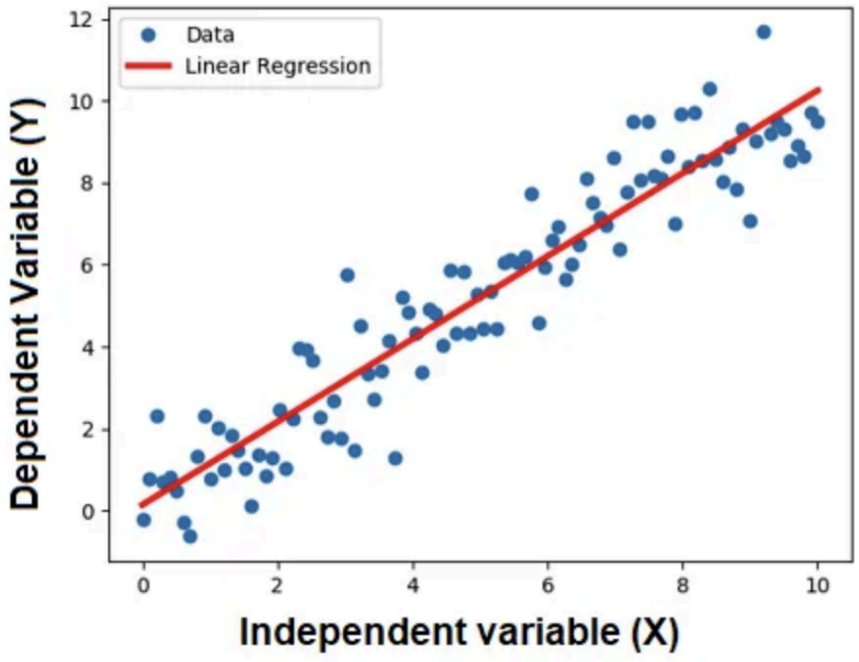
The Life expectancy dataset contains data related to immunization factors, mortality factors, economic factors, and social factors from 193 countries. Machine learning algorithms can capture complex relationships between these factors and life expectancy, which may not be easily discernible using traditional statistical methods. There are various machine learning algorithms like supervised, unsupervised, and reinforcement learning. In this project, we want to understand the relationships between features and output variables that we need to predict or estimate and all these variables are continuously varying numerical data. Hence, using Regression models will be a good approach for Life Expectancy prediction.

ML Regression Techniques

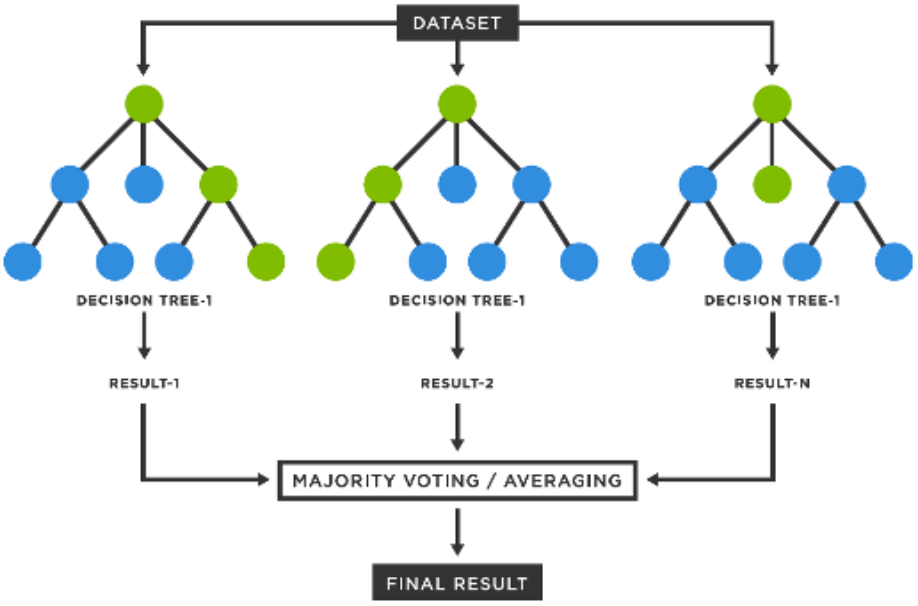
ML Regression techniques are statistical methods used to model the relationship between a dependent variable and one or more independent variables. Regression models can provide valuable insights when analyzing numerical datasets. By fitting a regression model to the data, we can identify patterns, trends, and correlations that exist within the dataset. These insights can help in making predictions, identifying influential factors, and making data-driven decisions. There are many popular regression models like linear regression, decision tree regression, random forest regression, gradient boot regression, support vector regression, etc. Factors such as the nature of the data, the assumptions of the model, and the desired balance between interpretability and predictive accuracy must be considered to choose the most appropriate regression technique.

Model Selection:

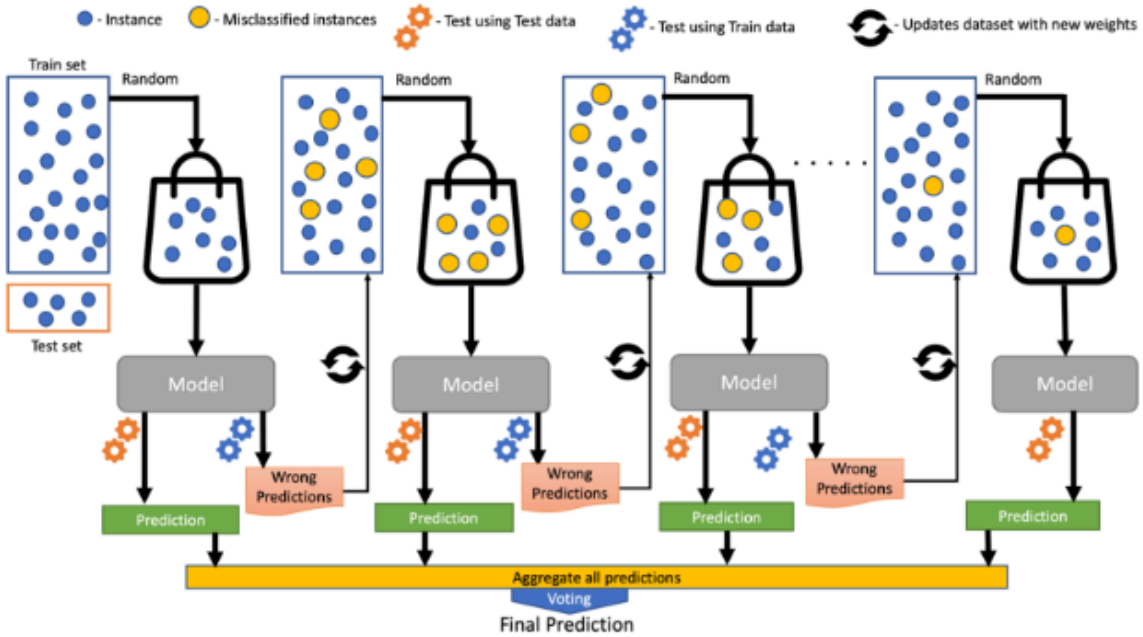
In this project, we will be using the following popular regression techniques:
Linear Regression: It is a simple and widely used regression technique that models the relationship between the dependent variable and independent variables by fitting a linear equation to the observed data. It provides a simple and good solution for a dataset containing continuous target variables with a linear relationship between the features and the target. However, if the relationships are non-linear it leads to high bias.



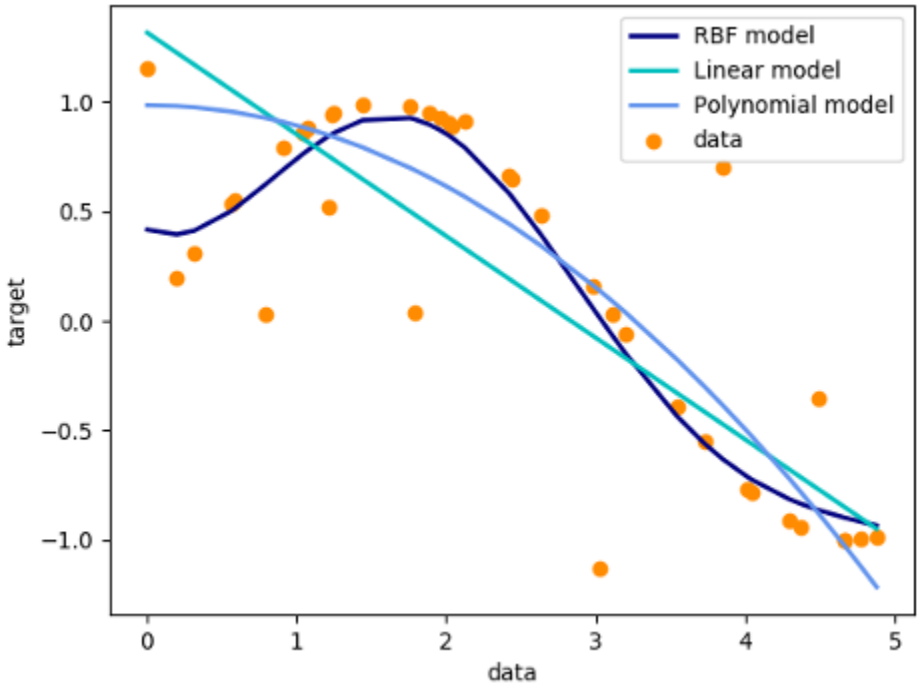
Random Forest Regression: It is an ensemble learning technique that combines multiple decision trees to improve predictive performance. Decision trees are tree-like structures used to make predictions based on decision rules applied to the input features. Each internal node of the tree represents a decision based on a feature, and each leaf node represents the predicted value of the target variable. Decision trees can capture non-linear relationships between features and the target. In this technique, multiple decision trees are built using random subsets of the training data, and the predictions are averaged to make the final prediction. Random forests are robust to overfitting and handle non-linear relationships well.



Gradient Boosting Regression: Gradient boosting regression is another ensemble learning technique that builds an ensemble of weak learners (typically decision trees) sequentially, where each new learner corrects the errors made by the previous ones. Gradient boosting can achieve high predictive accuracy and is robust to outliers and noisy data.



Support Vector Regression (SVR): Support vector regression is a regression technique based on support vector machines (SVMs). It works by finding the hyperplane that best fits the data while minimizing the error between the predicted and actual values. SVR is effective in sparse high-dimensional datasets with complex non-linear relationships and is robust to overfitting and noise. SVR uses Kernel functions which are mathematical functions used to map input features into higher dimensional space where the data points may be more separable. There are various kernel functions like Linear, Polynomial, and Radial Basis Functions (RBF). However, this technique is computationally intensive and memory-intensive.

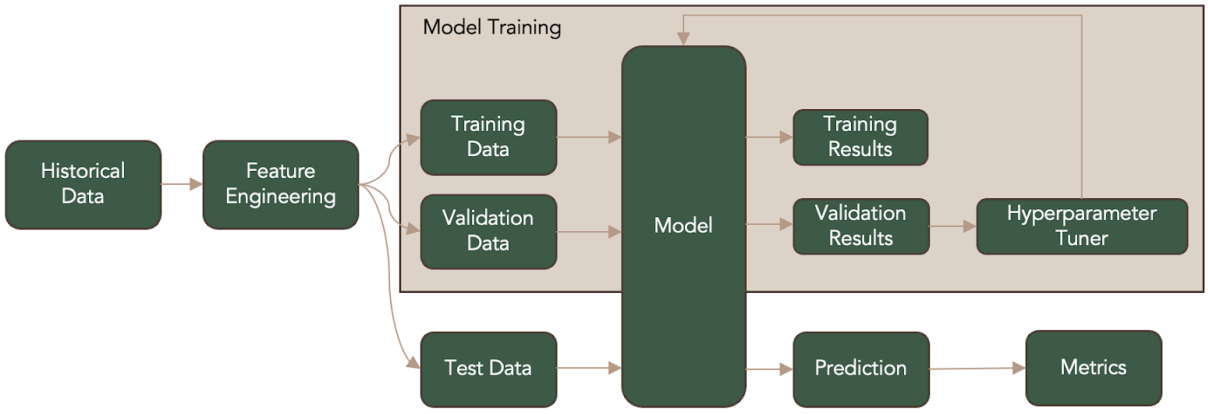


Ensemble Prediction: In this method, the predictions of multiple individual models are combined to produce a single, more robust prediction. Each regression model may capture different aspects of the underlying relationship between the features and the target variable. By combining multiple models, we can leverage the diversity of predictions to obtain a more comprehensive understanding of the data and potentially improve predictive accuracy. Ensembling also helps to reduce the risk of overfitting by aggregating the predictions of multiple models. If one model performs poorly on certain data points or in certain regions of the feature space, the contributions of other models can compensate for this, leading to more robust predictions. Furthermore, different regression models have different biases and variances. Combining multiple models with different biases and variances through weighted averaging can help strike a better balance between bias and variance, leading to improved generalization performance.

One popular ensemble technique is weighted averaging, where the predictions of each model are combined using weighted averaging to produce the final prediction. The weights assigned to each model are typically determined through optimization techniques, the nature of the dataset, or domain knowledge.

Training and Evaluation Process

The machine learning model training and evaluation process involves several key steps to ensure accurate and reliable performance. Feature engineering plays a crucial role in selecting, transforming, or extracting useful features from raw data to improve model accuracy. This includes feature transformation for numerical variables, feature encoding for categorical variables, and feature extraction/data cleaning. Once the features are prepared, cross-validation techniques such as k-fold validation are utilized to further validate the model and obtain more reliable performance estimates. Additionally, hyperparameter tuning is essential for optimizing model performance. For instance, in Support Vector Regression (SVR), parameters like kernel type and regularization parameter are adjusted. Similarly, Random Forest and Gradient Boosting models require tuning of parameters such as the number of trees, maximum depth of trees, and minimum number of samples per leaf. By systematically tuning hyperparameters and validating the model using cross-validation, machine learning practitioners can build robust and accurate models for a wide range of applications.



Evaluation Metrics:
Regression model evaluation metrics provide insights into how well a regression model performs in predicting the target variable. Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) are commonly used metrics to quantify the deviation between predicted and actual values. MAE represents the average absolute difference between predicted and actual values, providing a straightforward measure of prediction accuracy. MSE computes the average squared difference between predicted and actual values, giving more weight to large errors and penalizing outliers. RMSE, the square root of MSE, provides an interpretable measure

in the same units as the target variable, making it easier to understand the scale of prediction errors. Additionally, the coefficient of determination (R2) measures the proportion of variance in the dependent variable explained by the regression model. R2 values range from 0 to 1, with higher values indicating a better fit of the model to the data. Together, these evaluation metrics offer a comprehensive assessment of the performance and predictive accuracy of regression models.

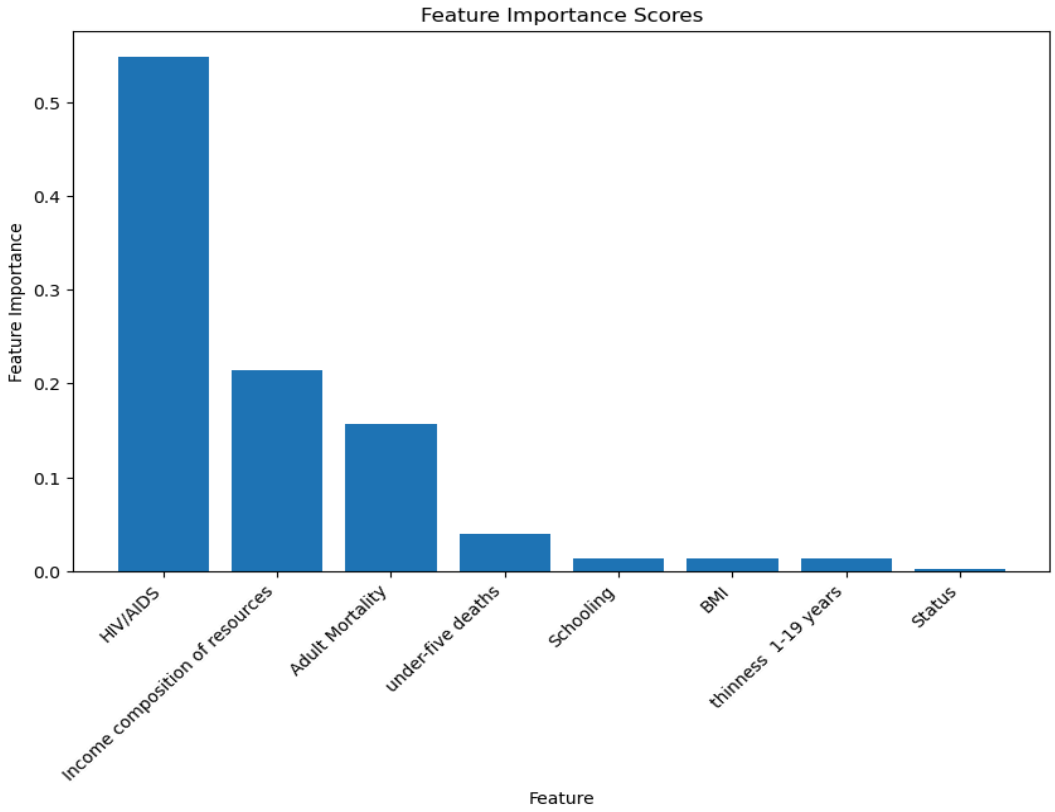
Results

Model was trained and evaluated initially to establish a baseline performance level.

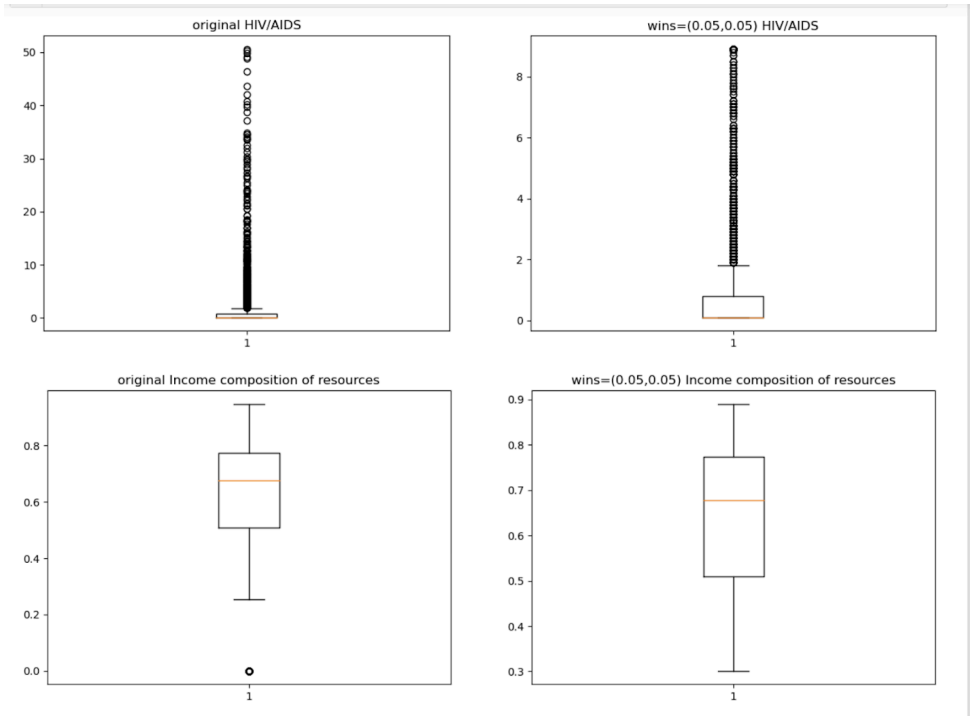
```
8
9  def train_and_evaluate_model(model, x_train, y_train, x_test, y_test, results_df):
10     # initialize your model object
11     model = model
12     model_name = model.__class__.__name__
13
14     # Fit the model on the training data
15     model.fit(x_train, y_train)
16
17     # Calculate the score of the model on the training data
18     train_score = model.score(x_train, y_train)
19     #print(f"Score of the {model_name} model on the training data is: {train_score}")
20
21     # Make predictions on the test data
22     predictions = np.round(model.predict(x_test), decimals = 2)
23
24     # See R2 score on the test data
25     test_r2_score = r2_score(y_test, predictions)
26     #print(f"R2 score of the {model_name} model on the test data is: {test_r2_score}")
27
28     mae_score = np.mean(np.abs(y_test - predictions))
29     #print(f"MAE score of the {model_name} model on the test data is: {mae_score}")
30
31     rmse_score = np.sqrt(((predictions - y_test) ** 2).mean())
32
33     # Append the model scores to the resutls DataFrame
34     model_scores = pd.DataFrame({'Model': [model_name], 'Training Score': [train_score], 'R2 Score': [test_r2_sc
35     results_df = pd.concat([results_df, model_scores], ignore_index=True)
36     return results_df
37
38 results_df = pd.DataFrame(columns=['Model', 'Training Score', 'R2 Score', 'MAE', 'RMSE'])
39 results_df = train_and_evaluate_model(LinearRegression(), x_train, y_train, x_test, y_test, results_df)
40 results_df = train_and_evaluate_model(SVR(C = 9.0, epsilon = 0.9, kernel = 'rbf'), x_train, y_train, x_test, y_t
41 results_df = train_and_evaluate_model(RandomForestRegressor(n_estimators = 100, max_depth=7, min_samples_split=5
42 results_df = train_and_evaluate_model(GradientBoostingRegressor(n_estimators = 100, max_depth = 6, min_samples_s
43
44 print(results_df)
45
```

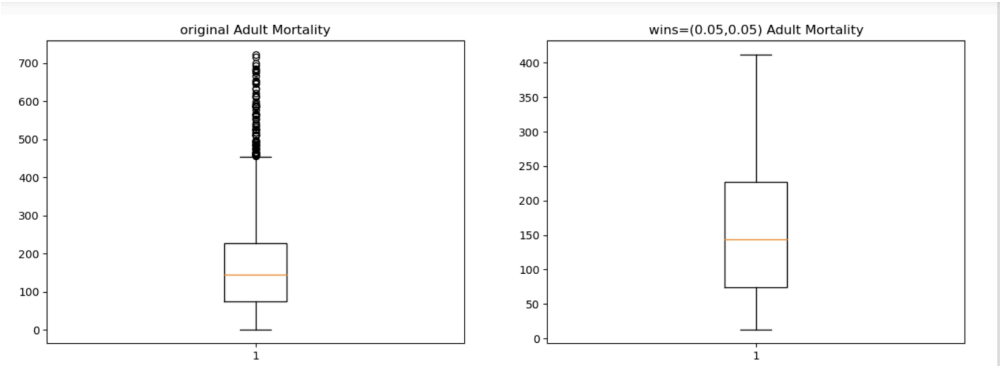
```
                   Model  Training Score  R2 Score       MAE      RMSE
0           LinearRegression        0.785050  0.780977  3.168027  4.356851
1                        SVR        0.898691  0.907321  1.929218  2.834125
2      RandomForestRegressor        0.965495  0.953154  1.430034  2.014947
3  GradientBoostingRegressor        0.990315  0.967968  1.108010  1.666174
```

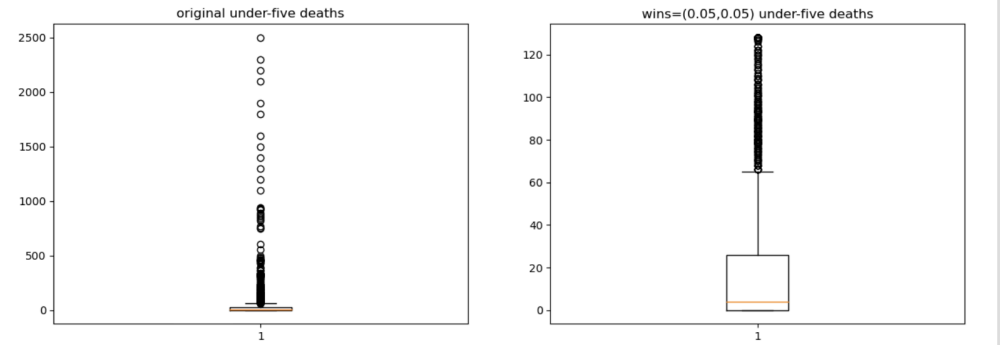
Feature importance scores were then computed to ascertain the significance of features. Feature importance scores quantify the relevance of different input variables in influencing the model's predictions. They help identify which features have the most impact on the output.

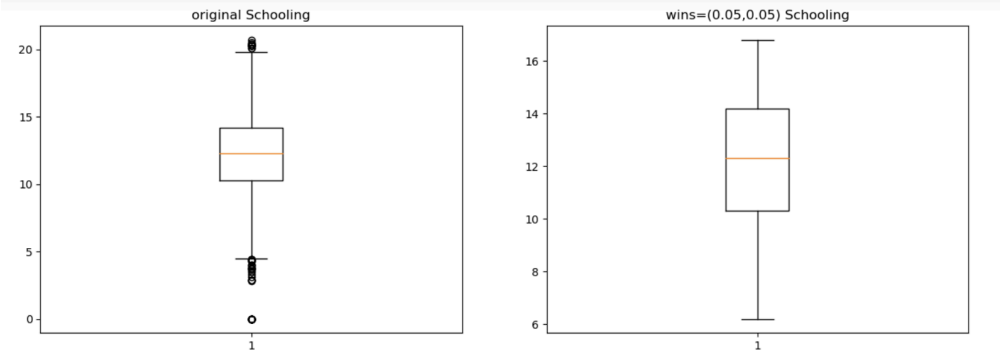


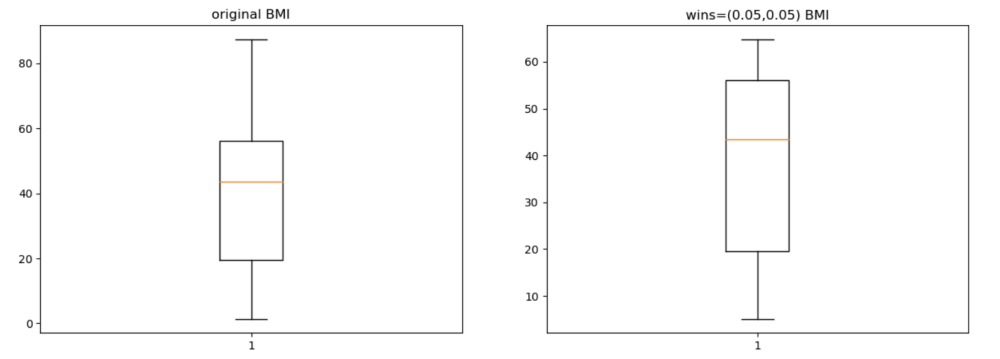
Winsorization was applied to mitigate outliers, followed by correlation matrix analysis to understand feature relationships post-winsorization.

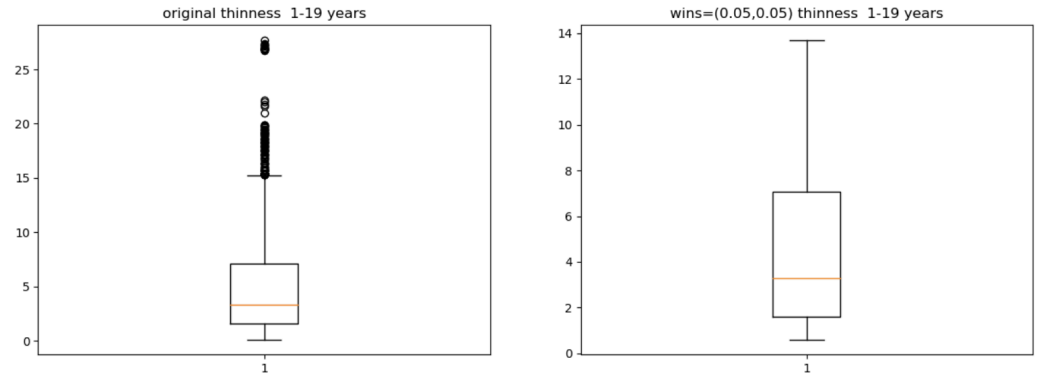


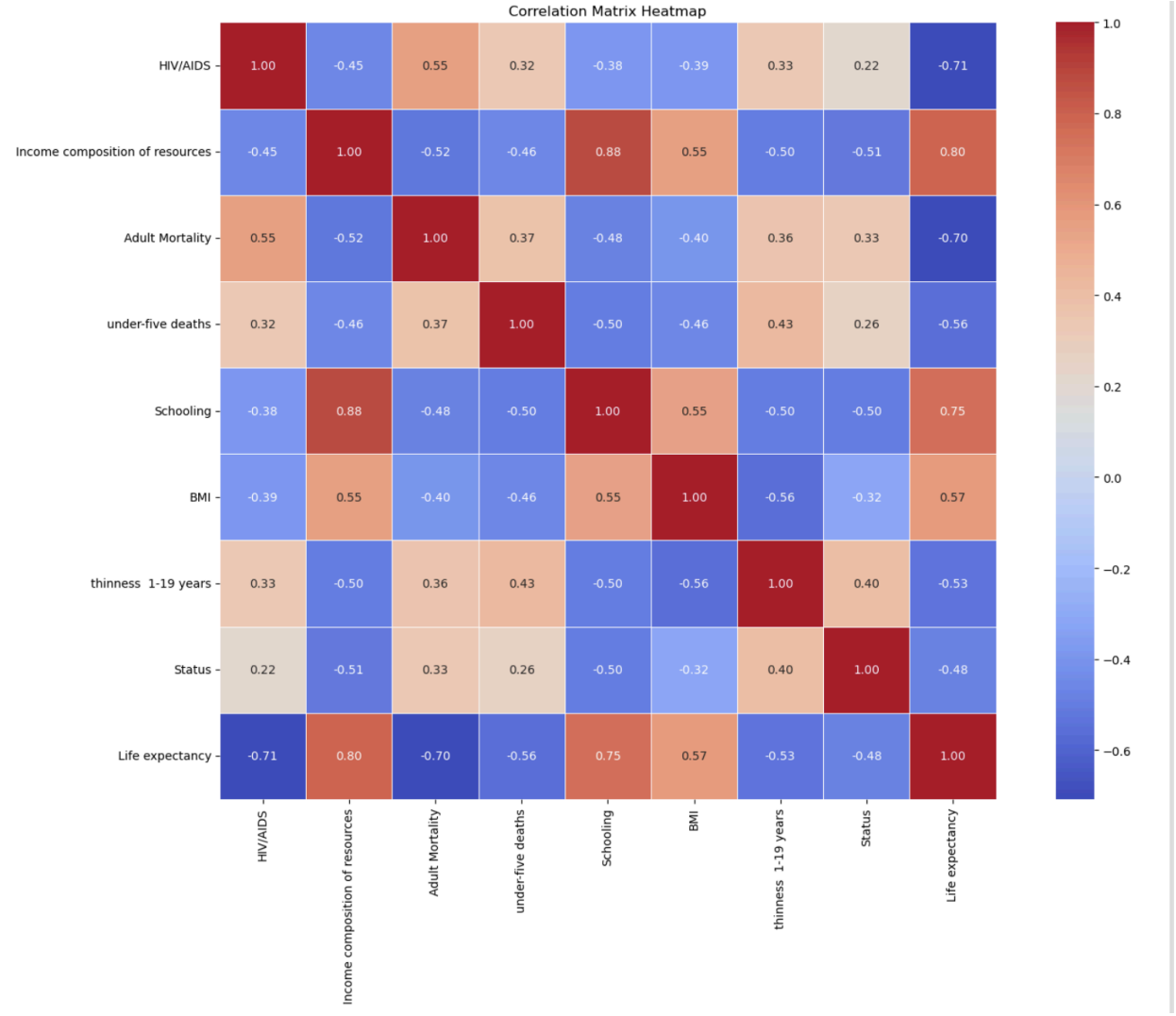

original Adult Mortality
wins=(0.05,0.05) Adult Mortality


original under-five deaths
wins=(0.05,0.05) under-five deaths


original Schooling
wins=(0.05,0.05) Schooling


original BMI
wins=(0.05,0.05) BMI


original thinness 1-19 years
wins=(0.05,0.05) thinness 1-19 years


Correlation Matrix Heatmap
HIV/AIDS
Income composition of resources
Adult Mortality
under-five deaths
Schooling
BMI
thinness 1-19 years
Status
Life expectancy

Hyperparameter tuning, conducted using cross-validation techniques, optimized the model's performance. Hyperparameter tuning involves adjusting the settings of a machine learning algorithm to optimize its performance. Cross-validation techniques are used during this process to assess how well the model generalizes to unseen data by splitting the dataset into multiple subsets for training and validation. This helps prevent overfitting and ensures the chosen hyperparameters work well across different data samples.

Post-optimization training and evaluation further refined the model.
Additionally, Voting Regressor, an ensemble technique, which combines predictions from multiple regression models through a weighted average was trained and evaluated, demonstrating improvement compared to individual models

```
results_df = pd.DataFrame(columns=['Model', 'Training Score', 'R2 Score', 'MAE', 'RMSE' ])
results_df = train_and_evaluate_model(LinearRegression(), x_train, y_train, x_test, y_test, results_df)
results_df = train_and_evaluate_model(SVR(C = 10.0, epsilon = 0.9, kernel = 'rbf'), x_train, y_train, x_test, y_
results_df = train_and_evaluate_model(RandomForestRegressor(n_estimators = 289, max_depth=30, min_samples_split=
results_df = train_and_evaluate_model(GradientBoostingRegressor(n_estimators = 100, max_depth = 6, min_samples_s


# Create individual models
linear_reg = LinearRegression()
svr = SVR(C = 10.0, epsilon = 0.9, kernel = 'rbf')
random_forest = RandomForestRegressor(n_estimators = 289, max_depth=30, min_samples_split=2)
gradient_boost = GradientBoostingRegressor(n_estimators = 100, max_depth = 6, min_samples_split = 5)

# Create ensemble model using VotingRegressor
ensemble_model = VotingRegressor([
    ('linear_reg', linear_reg),
    ('svr', svr),
    ('random_forest', random_forest),
    ('gradient_boost', gradient_boost)
], weights=[0.5, 3, 38, 23])


results_df = train_and_evaluate_model(ensemble_model, x_train, y_train, x_test, y_test, results_df)

print(results_df)
```

	Model	Training Score	R2 Score	MAE	RMSE
0	LinearRegression	0.853757	0.814226	2.713435	3.792404
1	SVR	0.938671	0.944342	1.399456	2.075810
2	RandomForestRegressor	0.995503	0.967906	0.988980	1.576276
3	GradientBoostingRegressor	0.993785	0.965539	1.030561	1.633379
4	VotingRegressor	0.994369	0.969046	0.976327	1.548032

Comparison of Metrics:

Model	R2 Score Before	R2 Score After	MAE Before	MAE After	RMSE Before	RMSE After
Linear Regression Model	0.7809	0.8142	3.1680	2.7134	4.3568	3.7924
SVR	0.9073	0.9443	1.9292	1.3994	2.8341	2.0758
Random Forest Regressor	0.9531	0.9679	1.4300	0.9889	2.0149	1.5762
Gradient Boosting Regressor	0.9679	0.9655	1.1080	1.0305	1.6661	1.6333

Conclusion

The project findings indicate that features such as HIV prevalence, income levels, and adult mortality rates exert a significant influence on life expectancy prediction. Through preprocessing the data and fine-tuning hyperparameters, notable enhancements in model performance were achieved. Additionally, the ensemble prediction outperformed individual models in terms of accuracy, suggesting the efficacy of combining multiple models for more reliable predictions. These insights underscore the importance of feature selection, data preprocessing, and model optimization techniques in improving the accuracy and robustness of life expectancy prediction models.