Life Expectancy Prediction Project

Python Analysis & Modeling Report

December 18, 2025

1 Introduction

Project Objective

The objective of this project is to analyze the factors that affect life expectancy across different countries and years, and to build machine learning models capable of predicting life expectancy based on health, economic, and social indicators.

The project follows a complete data science pipeline including:

- Data understanding
- Data cleaning
- Feature engineering
- Preprocessing
- Modeling and evaluation

2 Dataset Overview

Data Source

The dataset is provided by the World Health Organization (WHO) and the United Nations, and was obtained from Kaggle.

- **Number of countries:** 193
- **Time period:** 2000 – 2015
- **Total rows:** 2938
- **Total features:** 22

Target Variable

- Life expectancy (in years)

Feature Categories

- **Health factors:** Mortality rates, immunization, HIV/AIDS
- **Economic factors:** GDP, expenditure

- **Social factors:** Schooling, income composition
- **Demographic factors:** Population, status

3 Data Understanding

Initial Observations

- The dataset contains both numerical and categorical features.
- Some column names contained extra spaces.
- Several features contained missing values.
- Many numerical features were highly skewed.
- The data represents real-world measurements with natural variability.

No modifications were applied at this stage; the goal was only to understand the data and identify issues.

4 Data Cleaning

4.1 Column Name Cleaning

Problem: Some column names contained leading or trailing spaces, which could cause inconsistencies during analysis.
Solution: All column names were stripped of extra spaces to ensure consistent naming.

4.2 Handling Missing Values

Problem: Several important features contained missing values, such as Population, GDP, Hepatitis B, Alcohol, and Schooling. Missing value analysis showed that missingness often depended on other variables (e.g., country status).

Solution: Different strategies were applied depending on the feature:

- Group-wise median imputation (based on Status) for features where missingness depended on development level.
- Global median imputation for features with moderate missing values.
- Dropping rows with missing target-related values (Life expectancy, Adult Mortality).

This approach preserved data integrity while minimizing bias.

4.3 Handling Skewed Features

Problem: Several numerical features showed strong right skewness, such as GDP, Population, Measles, Infant deaths, and HIV/AIDS. Skewed distributions can negatively affect model performance.

Solution: A logarithmic transformation ($\log(1 + x)$) was applied to highly skewed features.

- Original columns were overwritten.
- No additional columns were added.

- The transformation reduced skewness while preserving relative differences.

5 Feature Engineering

Objective

Feature engineering was applied to create meaningful variables that better represent real-world conditions affecting life expectancy. Only a small number of features were engineered to avoid overcomplication.

Engineered Features

Child Mortality Rate Combines infant deaths and under-five deaths to represent overall child mortality burden.

Immunization Average Average coverage of Hepatitis B, Polio, and Diphtheria.

Economic Strength Combines GDP and income composition to reflect economic stability.

Education Index Combines schooling and income composition to represent human development.

Mortality Pressure Combines adult mortality and HIV/AIDS prevalence.

These features are intuitive, interpretable, and aligned with domain knowledge.

6 Preprocessing

6.1 Saving Dataset for Analysis

After cleaning and feature engineering, a clean and interpretable dataset was saved for analytical purposes. This version:

- Contains country and year information.
- Contains engineered features.
- Is not encoded or scaled.

6.2 Encoding Categorical Variables

Problem: Machine learning models require numerical inputs.
Solution: The categorical feature *Status* was encoded using binary encoding:

- Developed → 1
- Developing → 0

The *Country* column was retained at this stage and removed only before modeling.

6.3 Train/Test Split

The dataset was split into:

- 80% training data
- 20% testing data

This step was performed before scaling and feature selection to prevent data leakage.

6.4 Scaling

Numerical features were scaled using `StandardScaler`.

- The scaler was fitted on training data only.
- The same scaler was applied to test data.

Scaling ensured that all features contributed equally to the learning process.

6.5 Saving Preprocessed Arrays

The final preprocessed datasets were saved as NumPy arrays: `X_train`, `X_test`, `y_train`, and `y_test`. This ensures reproducibility and efficient experimentation with multiple models.

7 Modeling

7.1 Baseline Model – Linear Regression

A linear regression model was used as a baseline.

- **MAE:** ≈ 2.7 years
- **RMSE:** ≈ 3.5 years
- **R^2:** ≈ 0.85

The close performance between training and testing sets indicated good generalization.

7.2 Advanced Model – Random Forest Regressor

A Random Forest model was trained to capture non-linear relationships.

- **MAE:** ≈ 1.1 years
- **RMSE:** ≈ 1.7 years
- **R^2:** ≈ 0.97

The advanced model significantly outperformed the baseline, showing that the relationship between features and life expectancy is non-linear.

8 Model Comparison Summary

The Random Forest model provided a substantial improvement in predictive accuracy.

Table 1: Performance comparison between Baseline and Advanced models

Model	**MAE**	**RMSE**	**R^2**
Linear Regression	≈ 2.7	≈ 3.5	≈ 0.85
Random Forest	≈ 1.1	≈ 1.7	≈ 0.97

9 Conclusion

A complete end-to-end machine learning pipeline was successfully implemented.

- Data quality issues were systematically identified and resolved.
- Feature engineering enhanced model performance without overcomplicating the dataset.
- The baseline model achieved strong performance, while the advanced model captured complex relationships more effectively.
- The final solution is accurate, stable, and suitable for real-world analysis.