Team Details

Team Name: Team Synergy
Team Members:

1. Rishitha Rani Pakam – Team Lead
   Email ID: rpaka1@unh.newhaven.edu
2. Narasimha Reddy Padire
   Email ID: npadi1@unh.newhaven.edu
3. Lakshmi Reddy Bhavanam
   Email ID: lbhav2@unh.newhaven.edu

Dataset Title: Life Expectancy (World Health Organization) 2024

Source: https://www.kaggle.com/datasets/sonialikhan/life-expectancy-who-2024

Selected dataset contains 22 attributes and 2938 records. It contains a wide range of variables across 193 countries, including health, economic, and social indicators. Attributes in the dataset are Country, Year, Status, Life expectancy, Adult Mortality, infant deaths, Alcohol percentage, expenditure, Hepatitis B, Measles, BMI, under-five deaths, Polio, Total expenditure, Diphtheria, HIV/AIDS, GDP, Population, thinness 1-19 years, thinness 5-9 years, Income composition of resources, Schooling. The dataset contains both numerical and categorical data. There are 2 attributes of object type, 11 decimal and 9 integer types.

Research Question

What are the most significant factors predicting life expectancy, and how do health, economic, and social variables interact to influence life expectancy in the top two developed countries and the top two developing countries?

Data Mining Techniques:

1. K-Means Clustering.
2. Multiple Linear Regression.
3. Random Forest Regression.

Parameters:

These are values learned from the data during the training process. They define the model's behavior based on the input data.

Data Mining Technique	Data Modelling	Parameters
K-Means Clustering	Clustering	• n_clusters • random_state • cluster labels • features
Multiple Linear Regression	Regression	• Root Mean Square Error • Mean Absolute Error • R^2 Score • CV R^2 Score

Random Forest Regression	Regression	• Root Mean Square Error • Mean Absolute Error • R^2 Score • CV R^2 Score

Hyperparameters:

These are configurations set before the training process begins. They control the training process and the structure of the model.

Data Mining Techniques	Hyperparameters	Values Used
K-Means Clustering	➢ n_clusters(param_grid) ➢ init ➢ n_init ➢ max_iter ➢ tol	• different k values • k-means++, random • [10,20] • [300, 500] • [1e-4, 1e-3]
Multiple Linear Regression	➢ n_estimators ➢ max_depth ➢ min_samples_split ➢ min_samples_leaf	• [100, 200, 300] • [5, 10, 15] • [2, 5, 10] • [1, 2, 4]
Random Forest Regression	➢ n_estimators ➢ max_depth ➢ min_samples_split ➢ min_samples_leaf	• [100, 200, 300] • [5, 10, 15] • [2, 5, 10] • [1, 2, 4]

1. n_estimators: The number of trees in the forest.
2. max_depth: The maximum depth of each tree.
3. min_samples_split: The minimum number of samples required to split an internal node.
4. min_samples_leaf: The minimum number of samples required to be at a leaf node.
5. n_clusters(param_grid): Number of clusters to form in **KMeans**.
6. init: Method for initializing centroids.
7. n_init: Number of times the algorithm runs with different initializations.
8. max_iter: Maximum number of iterations for the KMeans algorithm.
9. tol: Tolerance to declare convergence based on the change in WCSS.

Hardware Configuration:

- Programming Language: Python
- Tools Used: Kaggle
- Processor: i9
- Hard Disk: 1 TB
- Memory: 32 GB
- Operating System: Windows 13th Gen 64-bit

Outcomes of data mining techniques:

1) K-Means Clustering:

In this project, we applied K-Means clustering to categorize countries into two groups: developed and developing, based on various health, economic, and social features. The dataset was filtered to focus on top two developing countries – Japan, the United States of America and top two developing countries-- India, and Nigeria—and labeled as 'Developed' or 'Developing' based on their economic status. We then used K-Means clustering to identify hidden patterns in these countries' health and economic data, with a focus on variables such as GDP, infant mortality, alcohol consumption, and education levels.

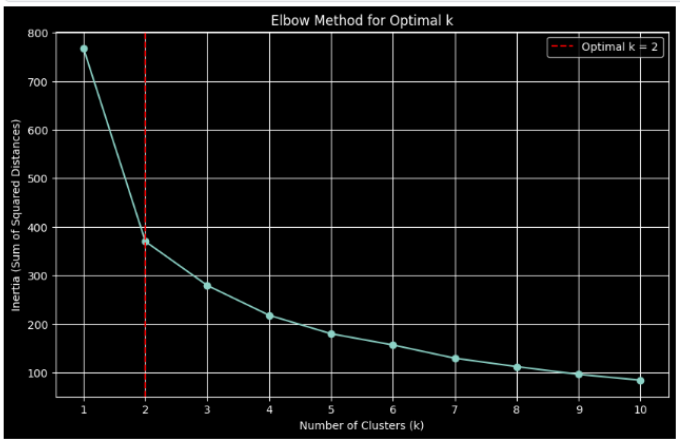
Countries in Cluster 0:

	Country	Development Status
1186	India	Developing
1187	India	Developing
1188	India	Developing
1189	India	Developing
1190	India	Developing
1191	India	Developing
1192	India	Developing
1193	India	Developing
1194	India	Developing
1195	India	Developing
1196	India	Developing
1197	India	Developing
1198	India	Developing
1199	India	Developing
1200	India	Developing
1201	India	Developing
1893	Nigeria	Developing
1894	Nigeria	Developing
1895	Nigeria	Developing
1896	Nigeria	Developing
1897	Nigeria	Developing
1898	Nigeria	Developing
1899	Nigeria	Developing
1900	Nigeria	Developing
1901	Nigeria	Developing
1902	Nigeria	Developing
1903	Nigeria	Developing

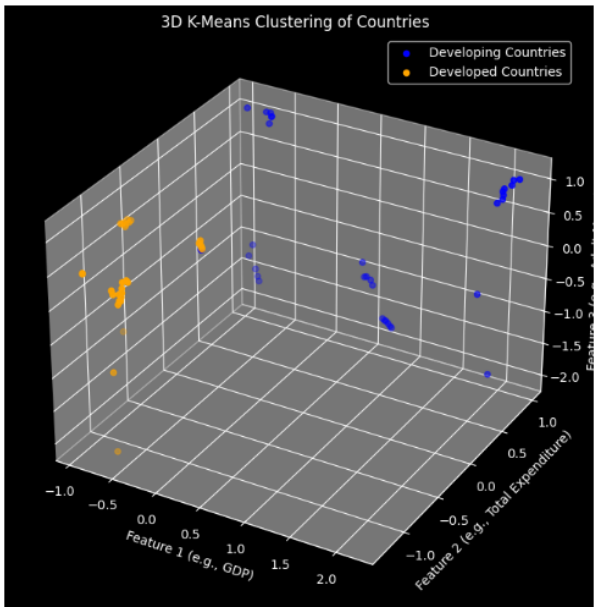
Countries in Cluster 1:

	Country	Development Status
1314	Japan	Developed
1315	Japan	Developed
1316	Japan	Developed
1317	Japan	Developed
1318	Japan	Developed
1319	Japan	Developed
1320	Japan	Developed
1321	Japan	Developed
1322	Japan	Developed
1323	Japan	Developed
1324	Japan	Developed
1325	Japan	Developed
1326	Japan	Developed
1327	Japan	Developed
1328	Japan	Developed
1329	Japan	Developed
2794	United States of America	Developed
2795	United States of America	Developed
2796	United States of America	Developed
2797	United States of America	Developed
2798	United States of America	Developed
2799	United States of America	Developed
2800	United States of America	Developed
2801	United States of America	Developed
2802	United States of America	Developed
2803	United States of America	Developed

We standardized the data using StandardScaler and applied K-Means with an initial assumption of 2 clusters. The Elbow Method confirmed that 2 clusters were optimal, clearly separating developed and developing nations.



We created a 3D visualization of the clustered data to clearly depict how the countries grouped based on key features. This visualization allowed us to observe the distinct differences between the clusters. To evaluate the effectiveness of the clustering, we used performance metrics such as the Silhouette Score and Davies-Bouldin Index. These metrics provided valuable insights into the quality and reliability of the clusters, helping us assess the appropriateness of the chosen model.



Silhouette Score: 0.4654048687948913

Davies-Bouldin Index: 0.9363167031369237

Inertia (Within-cluster sum of squares): 84.40503842751517

To improve the clustering, we used GridSearchCV to adjust key hyperparameters like the number of clusters, initialization method, and iterations. The optimized model performed better, confirming the clustering results were more reliable.

```
Fitting 3 folds for each of 80 candidates, totalling 240 fits
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=10, tol=0.0001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=10, tol=0.0001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=10, tol=0.0001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=10, tol=0.001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=10, tol=0.001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=10, tol=0.001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=20, tol=0.0001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=20, tol=0.0001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=20, tol=0.0001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=20, tol=0.001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=20, tol=0.001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=2, n_init=20, tol=0.001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=3, n_init=10, tol=0.0001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=3, n_init=10, tol=0.0001; total time=   0.0s
[CV] END init=k-means++, max_iter=300, n_clusters=3, n_init=10, tol=0.0001; total time=   0.0s
```

Best parameters found: {'init': 'random', 'max_iter': 300, 'n_clusters': 6, 'n_init': 10, 'tol': 0.0001}

Silhouette Score (Tuned): 0.5728174088913952

Davies-Bouldin Index (Tuned): 0.7189071511449656

Inertia (Tuned): 69.25043322255956

2) Multiple Linear Regression:

For all the techniques which we have used below, we utilized the two datasets i.e., Test and Train. From these two we have used the 80% of the Test data and 20% of the Train data throughout all the techniques.

Data Preparation:

The dataset was filtered to focus on top two developing countries – Japan, the United States of America and top two developing countries-- India, and Nigeria. Relevant features, such as GDP, alcohol consumption, adult mortality, and schooling, were selected for predicting life expectancy. The data was then split into training (80%) and testing (20%) sets to ensure proper model evaluation.

```
# Split the data into training and testing sets (80% training, 20% testing)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Standardize the features (scaling)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

Model Training:

The Multiple Linear Regression model was trained using the scaled dataset to predict life expectancy. This model was evaluated using the key performance metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), R-squared (R^2) and CV R^2 Scores.

```
Multiple Linear Regression Metrics:
Mean Squared Error: 11.841541510677867
Mean Absolute Error: 2.901816590157667
R-squared Score: 0.9228665759884956
Multiple Linear Regression - CV R² Scores: [0.81894904 0.87944779 0.77515756 0.81871457 0.95907993]
Multiple Linear Regression - Mean CV R² Score: 0.8502697791788725
```

+ Code + Markdown

3) Random Forest Regression:

Data Preparation:

The dataset was filtered to focus on top two developing countries – Japan, the United States of America and top two developing countries-- India, and Nigeria. Relevant features, such as GDP, alcohol consumption, adult mortality, and schooling, were selected for predicting life expectancy. The data was then split into training (80%) and testing (20%) sets to ensure proper model evaluation.

```python
# Split the data into training and testing sets (80% training, 20% testing)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Standardize the features (scaling)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

Model Training:

The Multiple Linear Regression model was trained using the scaled dataset to predict life expectancy. This model was evaluated using the key performance metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), R-squared (R^2) and CV R^2 Scores.

```
Random Forest Regressor Metrics:
Mean Squared Error: 155.0250477086624
Mean Absolute Error: 11.457675716440423
R-squared Score: -0.009802036882902154
Random Forest Regressor - CV R² Scores: [-0.07596395 -0.09803333 -0.01421127 -0.00228045 -0.02669795]
Random Forest Regressor - Mean CV R² Score: -0.043437390575500646
```

+ Code + Markdown

To optimize the performance of the Random Forest Regressor, hyperparameter tuning was performed using GridSearchCV. A range of parameters such as the number of estimators, maximum depth, minimum samples split, and minimum samples leaf were tested to find the best combination. The grid search process improved the model's performance, resulting in better prediction accuracy. The best parameters were selected based on the highest cross-validation R^2 score, which confirmed the reliability of the optimized model for predicting life expectancy.

```python
from sklearn.model_selection import GridSearchCV

# Define the hyperparameters to tune
param_grid = {
    'n_estimators': [100, 200, 300],
    'max_depth': [5, 10, 15],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4],
    #'max_features': ['auto', 'sqrt', 'log2']
}
```

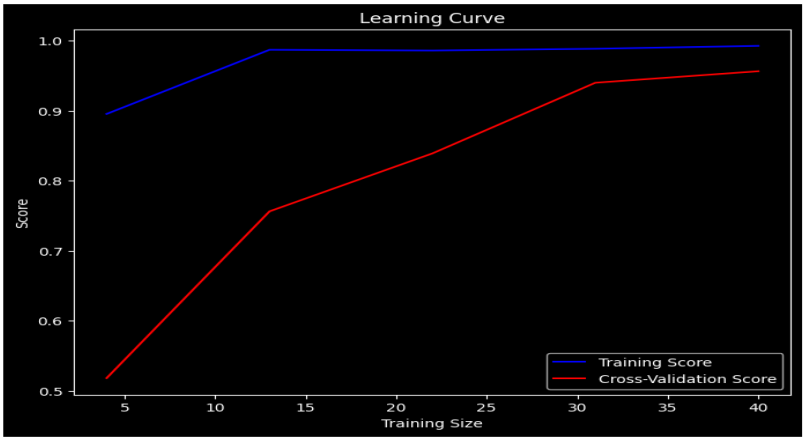
```
Best Parameters: {'max_depth': 5, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100}
Best CV R² Score: 0.9564305715687891

Best Random Forest Regressor Metrics:
Mean Squared Error: 0.2432994548435542
Mean Absolute Error: 0.34766401098901195
R-squared Score: 0.9984151961976155
```

+ Code + Markdown

Learning Curve:

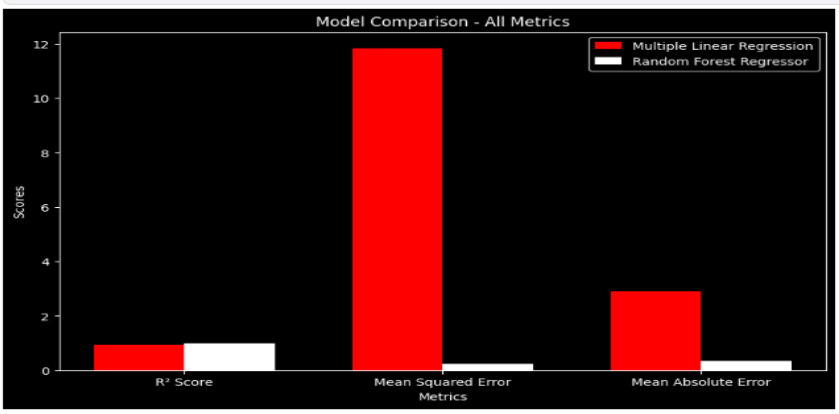
We plotted the learning curve for the Random Forest Regressor to observe how its performance improved as the training set size increased. This allowed us to identify potential issues with overfitting or underfitting. The curve showed that the model's performance stabilized as more data was used. This confirmed that the Random Forest Regressor was effectively predicting life expectancy based on the available data.



The learning curve shows a model's performance improving with larger training data. The small gap between the blue (training) and red (validation) lines suggests good generalization, indicating the model is not overfitting.

Visualizations:

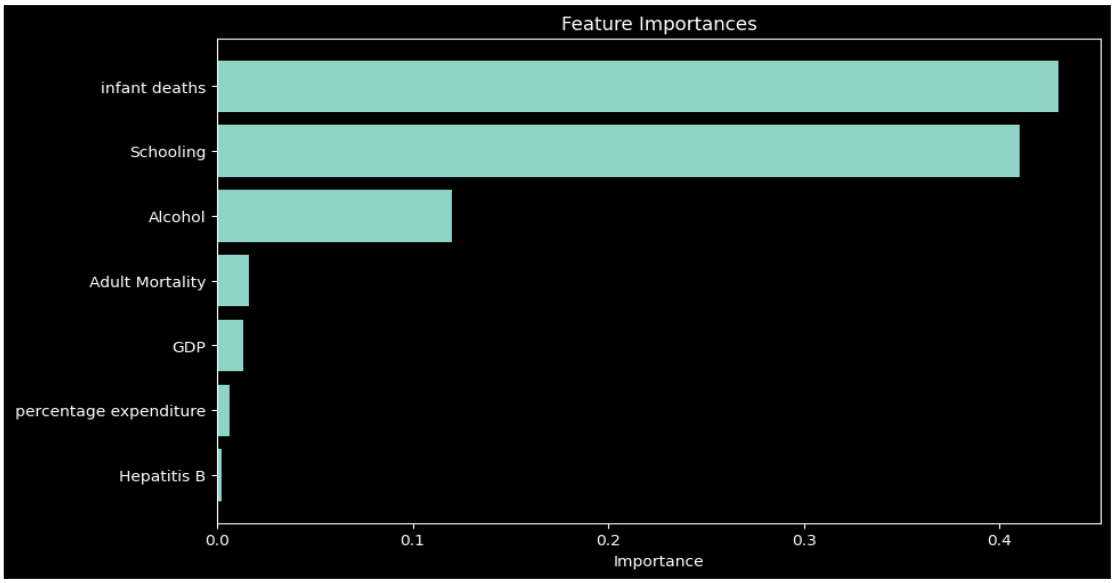
We visualized the model performance using a grouped bar chart. This chart displayed the comparison of key metrics—R^2 Score, Mean Squared Error (MSE), and Mean Absolute Error (MAE)—for both the Multiple Linear Regression and Random Forest Regressor models. The bars for each model were placed side by side to allow for easy comparison.



We visualized the feature importances in the Random Forest Regressor model to identify the most influential factors in predicting life expectancy. By plotting the features against their important

scores, we were able to clearly see which variables, such as GDP and Alcohol, played a key role. The horizontal bar chart provided a concise representation, with longer bars indicating higher importance, helping us prioritize features for further analysis or model refinement.

```
Feature Importances:
Hepatitis B: 0.0024
percentage expenditure: 0.0067
GDP: 0.0137
Adult Mortality: 0.0166
Alcohol: 0.1199
Schooling: 0.4105
infant deaths: 0.4302
```



Conclusion:

In conclusion, our data modeling effectively addressed the research question by identifying the most significant factors predicting life expectancy and examining how health, economic, and social variables interact in top 2 developed and developing countries. The optimized Random Forest Regressor, with an R-squared of 0.998 after hyperparameter tuning, accurately predicts life expectancy, highlighting the role of economic factors like GDP and social variables such as education. High life expectancy countries like Japan and the US show strong correlations with these factors, while lower life expectancy countries like India and Nigeria are more influenced by infant mortality and healthcare access. While Multiple Linear Regression also provides strong predictions with an R-squared of 0.923, it's slightly less accurate but stable, making it a solid alternative. K-Means clustering shows moderate cluster separation, which could be improved. Overall, optimized Random Forest model proved to be a strong predictor of life expectancy based on health, economic, and social variables, with Multiple Linear Regression as a reliable backup.

Github Repository Address: https://github.com/rishitharani24/Data-Mining---Team-Synergy