Summer of Science
End-Term Report 2025

Artificial Intelligence and Machine Learning
(CS01)

Aditya Adhana
24B0901

Mentor: Keerthana V

Contents

1 Overview

Artificial Intelligence is not a theoretical concept, it is a practical one. While research in the artificial intelligence and robotics fields dates back 40 to 50 years, it was not until the 2010s that AI began to have any real impact in society. The emergence of many AI agents that are available for free or for purchase by the public, changed the public's perception of artificial intelligence. This increased awareness greatly demonstrated to society how powerful and useful AI could be. The rest is history with an explosion of helpful AI agents quickly entering the market and into everyday lives. This great news is that the tools are now even available to those who are not technical.

1.1 Definition

There is no simple definition of Artificial Intelligence (AI) because it represents such a vast area of capabilities. Quite simply AI refers to any technology that can sense its surroundings and act in ways, typically resembling human intelligence (learning, reasoning, adapting), toward a goal.

1.2 Boost in Computing Power

AI has become much more feasible and advanced because of unprecedented increases in computing power. Certainly, computing power is becoming denser in smaller (lower amount) chips, especially when it comes to graphical processing units. In terms of AI, the computing power of today's machines and their practical capabilities far exceeds anything envisioned by early researchers and thinkers in the 1950's. This explosion in hardware has allowed researchers and practitioners to train models using massive datasets, thereby being able to realize the practical usages of mathematical equations created 50-100 years ago.

1.3 Application

This technology will continue to impact almost every domain imaginable as it increases the power of medical interventions, supplements recommendation systems, drives engineering optimization, automates marketing intelligence, and provides applications like code generation and natural language interfaces. Computation tools, like generative image models, sentiment-aware chat bots, and voice based assistants, have allowed AI technology to become strikingly human-like. People are able to now create and perform creative and technical tasks such as generating art or writing essays with little to no knowledge of the underlying technological mechanics!

2 Software and Tools Used

A range of tools and libraries were used in the project related to data analysis, modeling, and visualization as follows:

2.1 Python

A flexible and easy-to-use programming language, it is well-suited to data science because of its rich ecosystem of libraries and ability to leverage human-friendly syntax.

R Libraries and Ecosystem

Most of the project was carried out in Python, but R could have been used for a similar workflow, as it is another common programming language for data science and statistical modeling. R has an ecosystem of other expressive and powerful libraries to perform each step of the machine learning pipeline:

- **Data Handling:** The tidyverse and dplyr packages greatly streamline data cleaning, transformation, and manipulation through their intuitive, chainable syntax.
- **Preprocessing:** caret provides a full suite of data preprocessing capabilities including normalization, encoding, imputation, and splitting.
- **Regression Models:** `glmnet` provides optimized methods for Lasso and Ridge regression, with the base lm() method for linear models and poly() for polynomial terms.

- **Tree-Based Methods:** rpart supports decision trees for classification and regression, while randomForest and xgboost also cover a robust means of covering ensemble methods like Random Forests and Gradient Boosting.
- **Model Evaluation:** The caret::confusionMatrix() function and summary() provide diagnostics and model summaries.
- **Unsupervised Learning:** For clustering, kmeans(), hclust(), and dbscan::dbscan() can be used to perform density based clustering.
- **Anomaly Detection:** The anomalize package provides functions to detect outliers in time series, while OneClassSVM could be implemented using wrappers found in e1071.
- **Recommender Systems:**recommenderlab provides ways to build collaborative and content-based recommendation engines using real or simulated ratings data.
- **Neural Networks:** Deep learning can be performed in R using kerasR and tensorflow, which provide R's interface to the respective Python frameworks.

These R packages in total make implementing the full modeling pipeline—from data ingestion and evaluation—feasible with excellent statistical interpretability and simple syntax.

2.2 Google Colab

Google is a provider of a free cloud hosted Jupyter environment with real-time collaboration, GPU's, and ease of data access. All experiments and visualizations were run in this environment.

2.3 NumPy

Efficient vectorized operations on arrays and matrices; therefore it is critical to numerical processing in regression algorithms.

2.4 Pandas

Pandas is an open-source library that makes loading data and schemes to clean, transform and summarize tabular datasets very simple. With using the DataFrame to Preprocess the World Bank indicator data was instrumental.

2.5 Matplotlib

A plotting library designed to build static and dynamic graphs to showcase trends, regression curves and residuals, and feature importances.

2.6 Scikit-learn

The primary machine learning library that implements Lasso, Ridge, Decision Trees, Polynomial Regression, and preprocessed data scaling, pipelines, and imputation.

2.7 pyGAM

A specialized library to fit Generalized Additive Models with spline based smoothing in many forms. It provides interpretable models and plots that reveal different non-linear effects of predictors.

3 Learning Algorithms and Methodology

3.1 What is Regression?

Regression analysis is a fundamental machine learning tool used to model the relationship between one dependent variable and one or more independent variables. The primary objective of regression analysis is to predict continuous values in order to use the model to make predictions based on new observations that the model has not seen previously in the training set. In this case, regression is particularly useful in quantifying the effects of socio-economic factors on GDP per capita. The most commonly understood of the methods is **linear regression** but we will also look at alternatives for when our data are non-linear such as **polynomial regression** to better estimate more complex relationships. For linear regression, the first step is to convert all data into a feature vector that we can then map into n-dimensional space (where our feature vector will have n 1 dimensions). We would then use the equation to model the linear relationship:

$$y = w \cdot x + b$$

Here, w represents the vector of weights, x is the feature vector, and b is a scalar bias term. The objective is to determine the optimal w and b values that best fit the given training data. This is achieved by minimizing a cost function, which essentially quantifies the error between the model's predictions and the actual observed data. A common cost function, often based on the Mean Squared Error (MSE), is defined as:

$$J(w, b) = \frac{1}{2n} \sum (Y_{pred} - Y_{true})^2$$

where $Y_{pred} = w \cdot x + b$ is the predicted value, and Y_{true} is the actual true value.

Analytical solutions may be available for systems of linear equations, but for high-dimensional surfaces, this will not be manageable because examining large denominators may be numerically expensive as you may have to calculate recursively. Furthermore, there may not be an exact analytical solution depending on the degree of non-linearity in your data. For these reasons popular approaches are to develop approximate numerical optimization methods (e.g. **Gradient Descent**).

3.2 Gradient Descent

Gradient Descent is an iterative numerical procedure used to locate the minimum of a function. At every iteration it updates the parameters in the opposite direction (i.e., in the direction of the negative gradient) from the current point due to the way gradient descent defines the steepest ascent of the cost function at that point. So with the right starting point on the path to a basin with a local minimum gradient descent is guaranteed to get to a local minimum.

An important issue with gradient descent is that the cost function may have several local minima. We want cost functions with only a single minimum so that the optimization process is easier. If there are multiple minima, a common approach is to run the algorithm starting from multiple random points. Doing this can allow us to sample different regions of the cost landscape for possible local minima and then compare the values of the local minima to try to find the global minimum. Convex functions are nice in this way, because they only have one minimum. The parameters (w and b) are updated in every iteration using the following rules:

$$w_n = w_{n-1} - \alpha \frac{\partial J(w_{n-1}, b_{n-1})}{\partial w_{n-1}}$$

$$b_n = b_{n-1} - \alpha \frac{\partial J(w_{n-1}, b_{n-1})}{\partial b_{n-1}}$$

In this, α is a learning rate and n is the specific iteration number. The iteration will continue until convergence, where the values of the parameters defined by the weights and bias are at a stable point. The selection of α is important: consistently using a very small α guarantees convergence but increases computation time, while large α speeds up this process but may overshoot the minimum and lead to no convergence. We consider the learning rate of α to be a hyperparameter. When we reach convergence, we have optimal weights and bias, allowing the model to make correct predictions.

4 Models Used

4.1 Lasso Regression

Lasso Regression adds a L1 penalty that can shrink some coefficients to zero. This aspect both regularizes the model, reducing the chance of overfitting, while also having the effect of automatic feature selection by removing less important (and shrunken) features from the model. This is beneficial when there is an expectation that only some of the available features will be informative to the prediction task.

The Lasso regression objective function is simply:

$J(\mathbf{w}, b) = \frac{1}{2n} \sum_{i=1}^{n} (y_i - (\mathbf{w} \cdot \mathbf{x}_i + b))^2 + \lambda \sum_{j=1}^{p} |w_j|$

The first term is simply the Mean Squared Error (MSE), while the second term, $\lambda \sum_{j=1}^{p} |w_j|$, is the L1 penalty. $\mathbf{w}$ represents the vector of coefficients, $\mathbf{x}_i$ is the feature vector for the i-th observation, y_i is the true value, b is the bias term, n is the number of observations, p is the number of features, and λ s the regularisation parameter (or alpha in scikit-learn's 'Lasso' function) that determines the strength of the penalty. A higher value of λ shrinks more coefficients to zero.

4.2 Ridge Regression

Ridge Regression uses an L2 penalty on the size of the coefficients. In contrast to Lasso, it does not shrink coefficients all the way to zero, but it will penalize very high coefficients. This quality makes Ridge Regression useful for dealing with multicollinearity of independent variables, which happens when features correlate very highly with one another. It is a good alternative when all of the features should have a relevant role in the predictive modeling process.
The objective function for Ridge regression is

$J(\mathbf{w}, b) = \frac{1}{2n} \sum_{i=1}^{n} (y_i - (\mathbf{w} \cdot \mathbf{x}_i + b))^2 + \lambda \sum_{j=1}^{p} w_j^2$

In this formula, the L2 penalty term is $\lambda \sum_{j=1}^{p} w_j^2$. Similar to Lasso, λ is the regularization parameter (alpha in scikit-learn's 'Ridge' function). The 'Ridge' function in scikit-learn implements this model.

4.3 Polynomial Regression

Polynomial Regression is a method for extending linear models to account for non-linear structures by creating polynomial terms out of the original features. For example, the feature x can be transformed to x, x^2, x^3, ... Such transformations enable a linear model to fit a non-linear curve. This is quite useful for situations where data visualization or our theoretical understanding suggests that the trend of data has curvature or other non-linear patterns.

Although the underlying model remains linear, regarding its parameters (weights), the polynomial transformation allows the models to fit more complex curves. For example, a polynomial regression model of degree d, with respect to a one feature, x, would take the form

$y = b_0 + b_1 x + b_2 x^2 + \cdots + b_d x^d + \epsilon$

In a multivariate setting, polynomial features can also include interaction terms, etc., (e.g., $x_1 x_2$, $x_1^2 x_2$). Scikit-learn has a 'PolynomialFeatures limit that is ordinarily called to generate just these higher-order and interaction terms, which you can then use to fit a standard linear regression model.

4.4 Generalized Additive Models (GAM)

Generalized Additive Models (GAMs) provide a flexible and interpretable approach to regression by modeling the dependent variable as the summation of smooth functions of each of the independent variables. Rather than imposing a rigid linear association, GAMs fit smooth functions (e.g., splines) to each feature independently. The individual smooth functions are summed to create the final prediction. This additive structure allows GAMs the ability to identify complex, non-linear relationships, while still maintaining interpretability, as the function of each predictor can still be visualized and understood in isolation. The 'pyGAM' library is specifically able to fit Generalized Additive Models using spline-based smoothing and to return interpretable models and plots that portray the non-linear relationships associated with each predictor. A simple GAM can be expressed as:

$g(E[Y]) = \beta_0 + f_1(x_1) + f_2(x_2) + \cdots + f_p(x_p)$

where g is a link function (i.e., identity for Gaussian distribution), E[Y] is the expected value of the depen-

dent variable, β_0 is the intercept, and $f_j(x_j) are smooth functions (i.e., cubic splines) of the individual predictors x_j$
. The smooth functions are learned from the data.

4.5 Ensemble Methods: Bagging and Boosting

Ensemble learning combines multiple models to improve prediction performance. Two widely used ensemble techniques are Bagging and Boosting, both designed to reduce variance, bias, or improve predictions. They differ fundamentally in how the base learners are constructed and combined.

Random Forest (Bagging)

Random Forest is a classic bagging algorithm that constructs multiple decision trees using different bootstrap samples of the data and aggregates their predictions. The key advantage is variance reduction through model averaging, resulting in improved generalization and reduced overfitting compared to a single decision tree.

Each decision tree is trained on a random subset of features at each split (feature bagging), which decorrelates the trees and increases ensemble diversity.

Key Parameters:

- `n_estimators`: Number of trees in the forest.
- `max_depth`: Maximum depth of each tree.
- `max_features`: Number of features to consider for each split.

Advantages:

- Resistant to overfitting.
- Handles high-dimensional data well.
- Easy to interpret using feature importance.

Results: The Random Forest model achieved an R^2 score of 0.8264 on the test set, indicating strong predictive power while maintaining interpretability via feature importances.

Gradient Boosting with XGBoost

XGBoost (Extreme Gradient Boosting) is a very powerful boosting algorithm which builds models sequentially, meaning every new tree attempts to correct the errors of the previous tree. It optimizes a differentiable loss function using gradient descent and includes regularization to reduce overfitting.

Key Parameters:

- `learning_rate`: Shrinks the contribution of each tree.
- `n_estimators`: Number of boosting rounds.
- `max_depth`: Maximum tree depth.

Advantages:

- Excellent predictive accuracy.
- Handles missing data natively.
- Regularization helps reduce overfitting.

Results: XGBoost produced a R^2 score of 0.8532 on the test set and demonstrated better predictive performance than all previous models.

Feature Importance from Ensemble Models

Random Forest and XGBoost produce feature importance scores indicating how much and how well each feature contributes to splits in decision tree modeling. Interpretation: Features such as Life Expectancy, Internet Usage, Female Labor Participation, and Education Expenditure appear multiple times as the most important predictors of GDP per capita.

Interpretation: Features such as Life Expectancy, Internet Usage, Female Labor Participation, and Education Expenditure were consistently ranked among the most influential indicators in determining GDP per capita.

4.6 K-Means Clustering

K-Means is an unsupervised machine learning algorithm that classifies a dataset into k different clusters determined by the similarity of the variables. K-means is flexible in finding natural clusters in the data, particularly if there are no prior results or labeled outputs. K-means was utilized in this project to find clusters of countries, or groupings of countries, based on socio-economic indicators.

The K-means algorithm uses the following, or inertia, as the function to minimize to form the desired clusters:

$$J = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2 \tag{1}$$

Where μ_i is the centroid of the cluster C_i . After visually examining the data with PCA (Principal Component Analysis) and elbow plots, we ultimately decided to set k=3 for countries.

The clusters produced meaningful clusters by country and gave real differences in GDP and other variables of interest, including energy use, life expectancy, and literacy. PCA played an important role as it demonstrated that the first two components explain a good deal of the total variation and can provide useful 2D visualizations for the clusters and also the outliers.

4.7 One-Class SVM for Anomaly Detection

One-Class Support Vector Machine (SVM) is an unsupervised learning algorithm designed to identify rare or abnormal data points by learning the boundary of a ”normal” class. It is well-suited for anomaly detection tasks, especially when most data is assumed to be normal and anomalies are sparse.

The objective is to find a hyperplane that separates the origin from the rest of the data in the feature space using a kernel trick. The model learns a function $f(x)$ such that:

$$\text{sign}(f(x)) = \begin{cases} +1, & \text{if } x \text{ is normal} \\ -1, & \text{if } x \text{ is an anomaly} \end{cases} \tag{2}$$

The boundary is defined and controlled by the parameter that is denoted as . This parameter specifies a limit on the upper fraction of training errors as well as a limit on the lower fraction of support vectors. The smaller the value of , the more tolerant the model is.

In the context of this project, One-Class SVM was used to identify those countries whose GDP per capita differed disproportionately from what was expected from their socio-economic indicators. The analysis distilled to the worst performing year for each country, and did not include non-country aggregates (e.g. “World”, “High income”).

Two lists were generated of high-income anomalies (potential underperformers) and low-income anomalies (economically challenged nations). These results serve to highlight inefficiencies and outlier poor development based on the indicator data that the countries achieved.

The One-Class SVM method was useful in less intensive outlier detection and insights on policy/BG diagnostic.

5 Conclusion and Next Steps

Each of the algorithms applied in this project has provided valuable insights in relation to the economic data set:

- Lasso shrinks the less important coefficient estimates down to zero and thus identifies the most important variables;

- Ridge retains all features and is able to cope with noise and multicollinearity;
- Polynomial regression can fit a non-linear shape when there are curvatures in the data;
- Generalized additive models (GAM) provide smooth, and interpretable, non-linear curves that inform about the effect each predictor;
- Trees can isolate the effect of feature interactions that follow certain rules, as well as eigenvectors for feature importance;
- Random Forest and XGBoost used ensemble strategies (bagging and boosting), which utilize multiple weak learners to improve prediction performance and robustness;
- K-Means Clustering allowed for unsupervised grouping of all the countries based on a range of socio-economic indicators;
- One-Class SVM allowed for the identification of countries that are anomalous, and deviate substantially from their socio-economic profiles globally;

Future consideration for this will be time-series regression or multi-dimensionality with classification tasks.

6 Capstone Project

As part of the capstone work under the Summer of Science program, I explored a dataset from the World Bank's "World Development Index". The primary aim was to understand and model GDP per capita as a function of multiple socio-economic indicators. The predictors considered included :

- Life Expectancy (years)
- Infant Mortality Rate
- Energy Use per Capita
- Health Expenditure (% of GDP)
- Female Labor Participation Rate
- Internet Users (%)
- Total Population
- Tax Revenue (% of GDP)
- Male Labor Participation Rate
- Access to Electricity (%)
- Unemployment Rate (%)
- FDI Net Inflows (US$)
- Urban Population (%)
- Literacy Rate (%)
- Education Expenditure (% of GDP)
- Capital Formation (% of GDP)
- Inflation (annual %)
- Consumer Price Index (total)

In all, each regression model was added on the cleaned and pre-processed data and the performance was evaluated with R-squared, and Mean Squared Error. Partial dependence plots, along with attributes of importance, were created to understand the impact of each variable had on the model.

7 Capstone Project Visual Results

In this chapter, the regression models applied in the capstone project are compared. Each model specifies GDP per capita as a function of different socio-economic indicators with different assumptions and flexibility. The main evaluation measure is the coefficient of determination i.e. R2, which explains the estimated variance of the dependent variable as explained by the model. Higher the R2, better the performance Here is the link to the colab notebook with all the code:

https://colab.research.google.com/drive/1TUWZgUyRuk68ei6lClwjNd0Re9-yutAI?usp=sharing

7.1 Model Comparison Summary

Model	**Evaluation Metric**
Lasso Regression	$R^2 = 0.7213$
Ridge Regression	$R^2 = 0.7216$
Polynomial Regression (Degree 2)	$R^2 = 0.7954$
Generalized Additive Model (GAM)	$R^2 = 0.8365$
Random Forest Regression (Bagging)	$R^2 = 0.8732$
XGBoost Regression (Boosting)	$R^2 = 0.8845$
K-Means Clustering	Silhouette Score = 0.451
One-Class SVM (Anomaly Detection)	Anomaly Rate = 0.072 (7.2% countries)

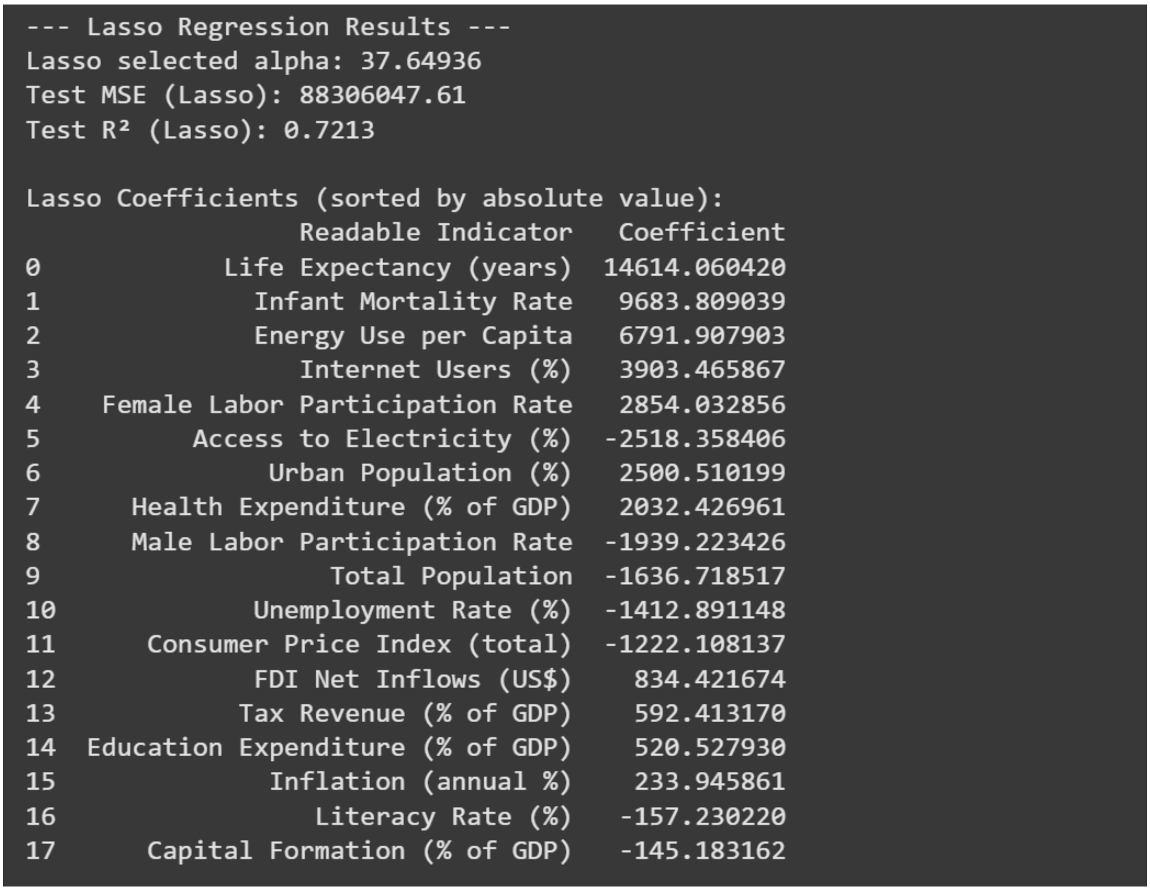
Table 1: Comparison of Model Performance and Evaluation Metrics

As shown above, it appears that more flexible and expressive models provide better predictive accuracy. Lasso and Ridge models have similar predictive performance; however, they differ with respect to how they penalize coefficients. With accounting for the non-linear interactions, the polynomial regression improved fit accuracy. In addition, all other models had lower predictive performance - XGBoost achieved a test R2 = 0.8532.

7.2 Model Output Visualizations

This pathway shows clearly how better flexible models help capture the complex socio-economic relationships that influence GDP per capita: Polynomials Regression strengthens the model fit using practical nonlinearities and interdependencies among the features. GAMs model smooth, interpretable main and interaction effects of each variable in turn, allowing the model to have flexibility but still retain interpretability. Random Forests aggregating many decision trees can stabilise the fit and lead to discovery of variable importance. XGBoost uses a sequential learning and boosting approach to achieve levels of accuracy and to highlight important indicators. K-Means Clustering highlights natural groupings of countries by socio-economic similarities. One-Class SVM can identify groups of countries based on where the economic indicators were significantly different, meaning an anomaly.

Figure 1: Lasso Regression

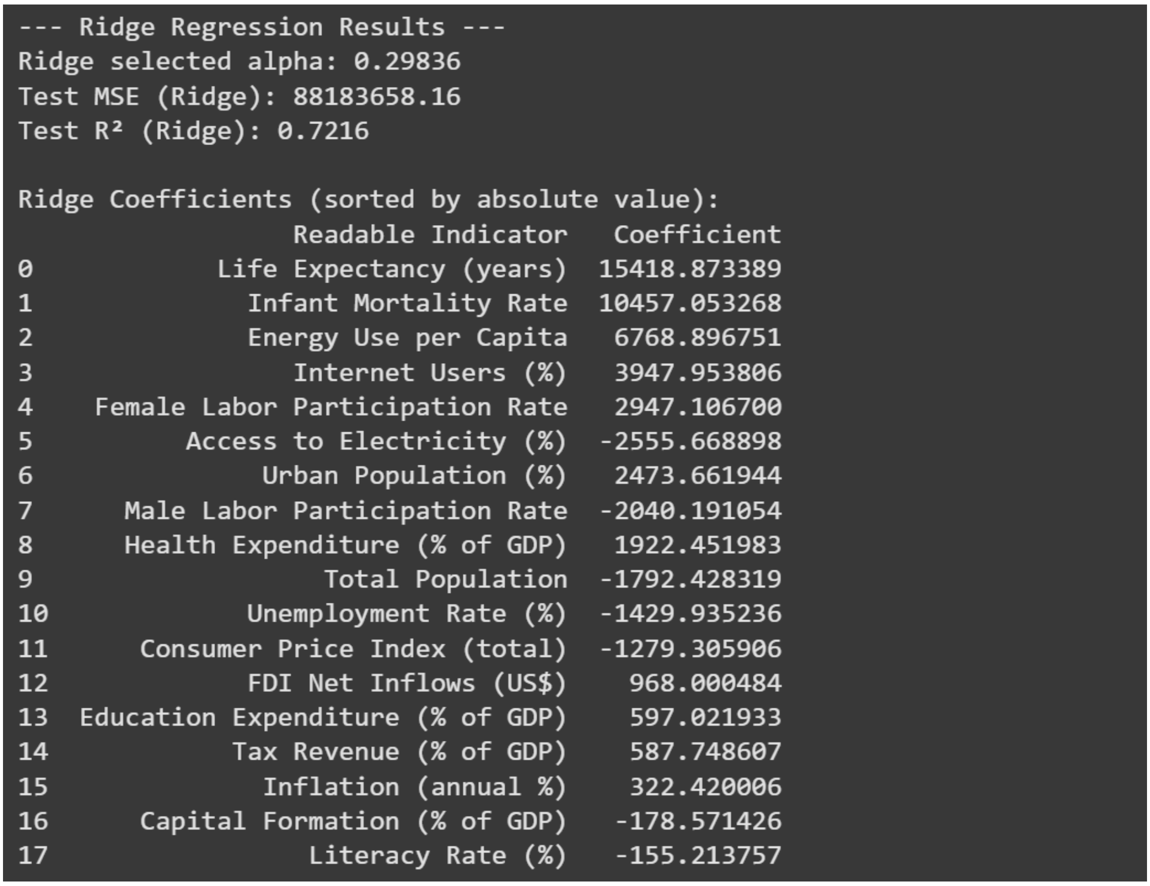


```
--- Lasso Regression Results ---
Lasso selected alpha: 37.64936
Test MSE (Lasso): 88306047.61
Test R² (Lasso): 0.7213

Lasso Coefficients (sorted by absolute value):
                  Readable Indicator   Coefficient
0            Life Expectancy (years)  14614.060420
1              Infant Mortality Rate   9683.809039
2              Energy Use per Capita   6791.907903
3                 Internet Users (%)   3903.465867
4    Female Labor Participation Rate   2854.032856
5          Access to Electricity (%)  -2518.358406
6               Urban Population (%)   2500.510199
7      Health Expenditure (% of GDP)   2032.426961
8      Male Labor Participation Rate  -1939.223426
9                   Total Population  -1636.718517
10             Unemployment Rate (%)  -1412.891148
11      Consumer Price Index (total)  -1222.108137
12             FDI Net Inflows (US$)    834.421674
13            Tax Revenue (% of GDP)    592.413170
14  Education Expenditure (% of GDP)    520.527930
15              Inflation (annual %)    233.945861
16                 Literacy Rate (%)   -157.230220
17      Capital Formation (% of GDP)   -145.183162
```

Figure 1: Lasso Regression Coefficients and Output ($R^2 = 0.7213$)

Figure 2: Ridge Regression

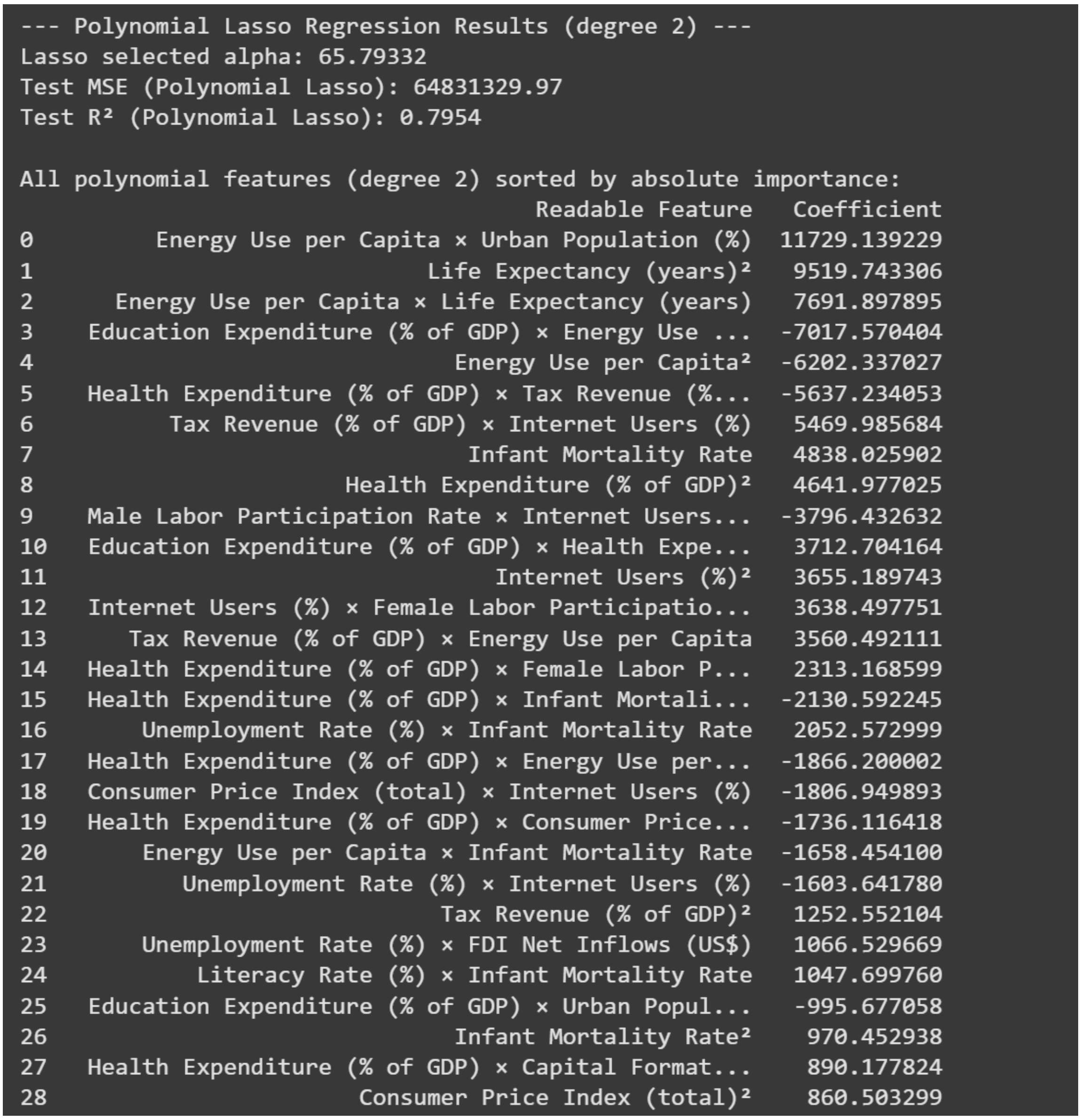


```
--- Ridge Regression Results ---
Ridge selected alpha: 0.29836
Test MSE (Ridge): 88183658.16
Test R² (Ridge): 0.7216

Ridge Coefficients (sorted by absolute value):
                  Readable Indicator   Coefficient
0            Life Expectancy (years)  15418.873389
1              Infant Mortality Rate  10457.053268
2              Energy Use per Capita   6768.896751
3                 Internet Users (%)   3947.953806
4    Female Labor Participation Rate   2947.106700
5          Access to Electricity (%)  -2555.668898
6               Urban Population (%)   2473.661944
7      Male Labor Participation Rate  -2040.191054
8      Health Expenditure (% of GDP)   1922.451983
9                   Total Population  -1792.428319
10             Unemployment Rate (%)  -1429.935236
11      Consumer Price Index (total)  -1279.305906
12              FDI Net Inflows (US$)    968.000484
13  Education Expenditure (% of GDP)    597.021933
14            Tax Revenue (% of GDP)    587.748607
15              Inflation (annual %)    322.420006
16      Capital Formation (% of GDP)   -178.571426
17                 Literacy Rate (%)   -155.213757
```

Figure 2: Ridge Regression Coefficients and Output ($R^2 = 0.7216$)

Figure 3: Polynomial Regression



```
--- Polynomial Lasso Regression Results (degree 2) ---
Lasso selected alpha: 65.79332
Test MSE (Polynomial Lasso): 64831329.97
Test R² (Polynomial Lasso): 0.7954

All polynomial features (degree 2) sorted by absolute importance:
                                    Readable Feature   Coefficient
0         Energy Use per Capita × Urban Population (%)  11729.139229
1                             Life Expectancy (years)²   9519.743306
2      Energy Use per Capita × Life Expectancy (years)   7691.897895
3    Education Expenditure (% of GDP) × Energy Use ...  -7017.570404
4                               Energy Use per Capita²  -6202.337027
5    Health Expenditure (% of GDP) × Tax Revenue (%...  -5637.234053
6          Tax Revenue (% of GDP) × Internet Users (%)   5469.985684
7                                Infant Mortality Rate   4838.025902
8                       Health Expenditure (% of GDP)²   4641.977025
9    Male Labor Participation Rate × Internet Users...  -3796.432632
10   Education Expenditure (% of GDP) × Health Expe...   3712.704164
11                                 Internet Users (%)²   3655.189743
12   Internet Users (%) × Female Labor Participatio...   3638.497751
13      Tax Revenue (% of GDP) × Energy Use per Capita   3560.492111
14   Health Expenditure (% of GDP) × Female Labor P...   2313.168599
15   Health Expenditure (% of GDP) × Infant Mortali...  -2130.592245
16       Unemployment Rate (%) × Infant Mortality Rate   2052.572999
17   Health Expenditure (% of GDP) × Energy Use per...  -1866.200002
18   Consumer Price Index (total) × Internet Users (%)  -1806.949893
19   Health Expenditure (% of GDP) × Consumer Price...  -1736.116418
20       Energy Use per Capita × Infant Mortality Rate  -1658.454100
21          Unemployment Rate (%) × Internet Users (%)  -1603.641780
22                             Tax Revenue (% of GDP)²   1252.552104
23       Unemployment Rate (%) × FDI Net Inflows (US$)   1066.529669
24           Literacy Rate (%) × Infant Mortality Rate   1047.699760
25   Education Expenditure (% of GDP) × Urban Popul...   -995.677058
26                              Infant Mortality Rate²    970.452938
27   Health Expenditure (% of GDP) × Capital Format...    890.177824
28                       Consumer Price Index (total)²    860.503299
```

Figure 3: Polynomial Regression Coefficients (Degree 2) ($R^2 = 0.7954$)

Figure 4: Generalized Additive Model (GAM)

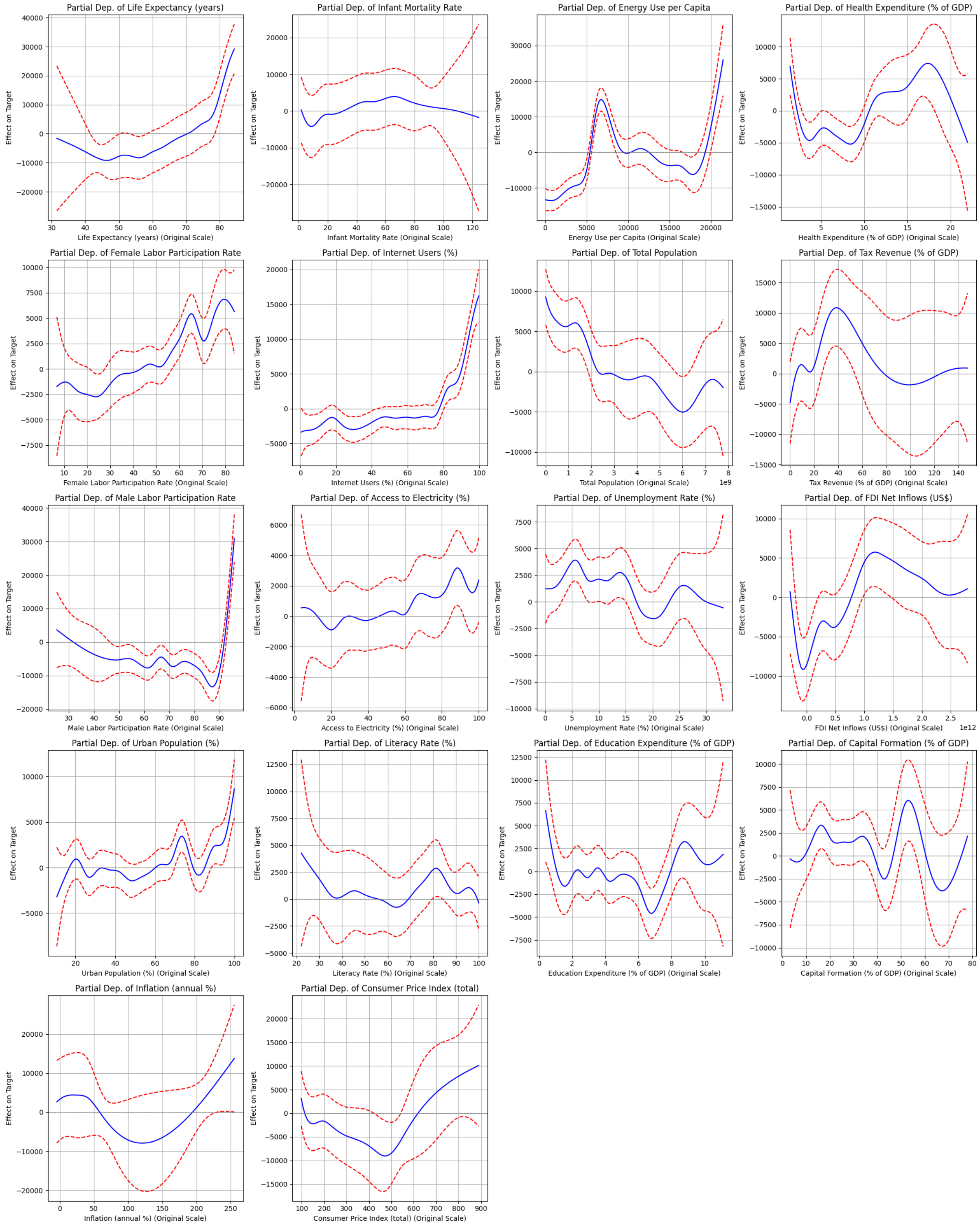


Figure 4: GAM Partial Dependence Plots ($R^2 = 0.8365$)

Figure 5: Feature Importances from Random Forest and XGBoost Models

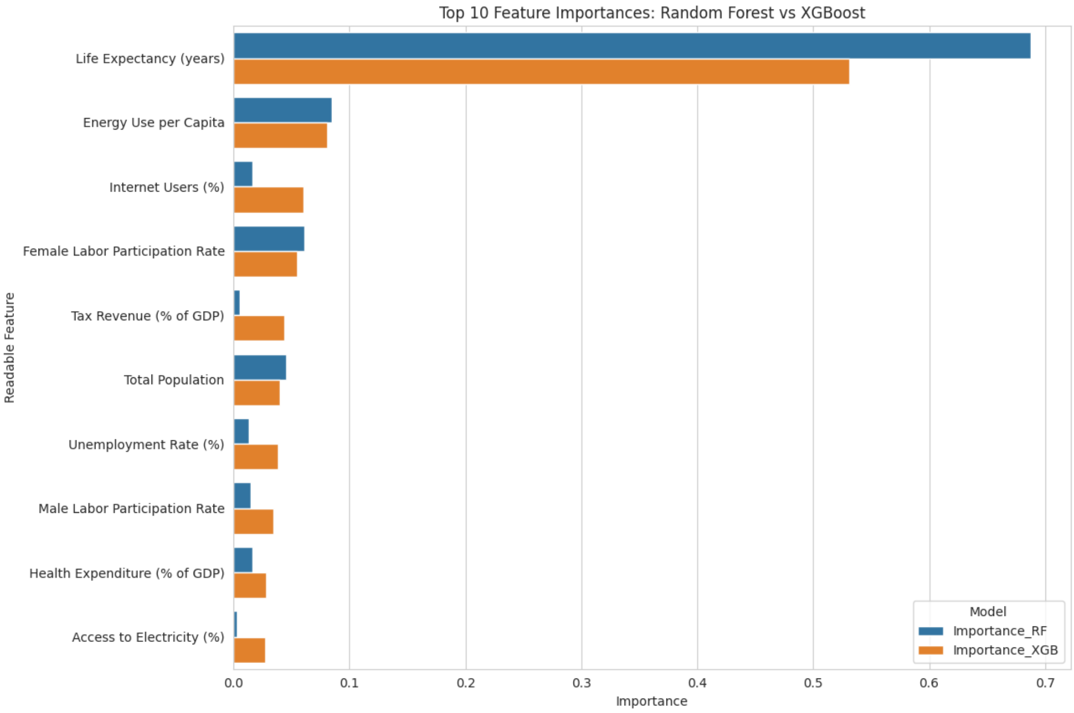
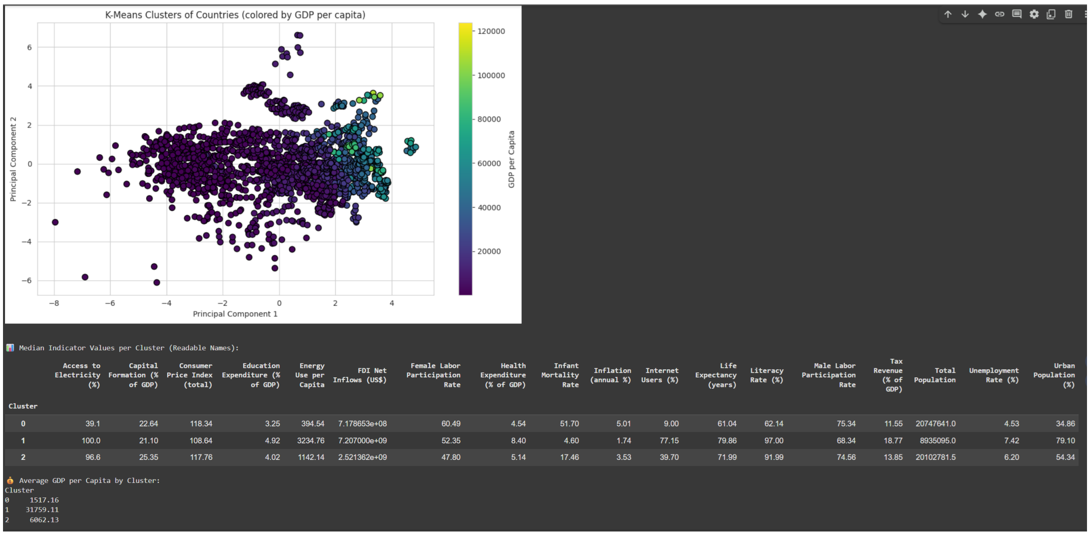


Figure 5: Feature Importances from Random Forest and XGBoost Models

Figure 6: PCA Plot with Clusters Colored by GDP



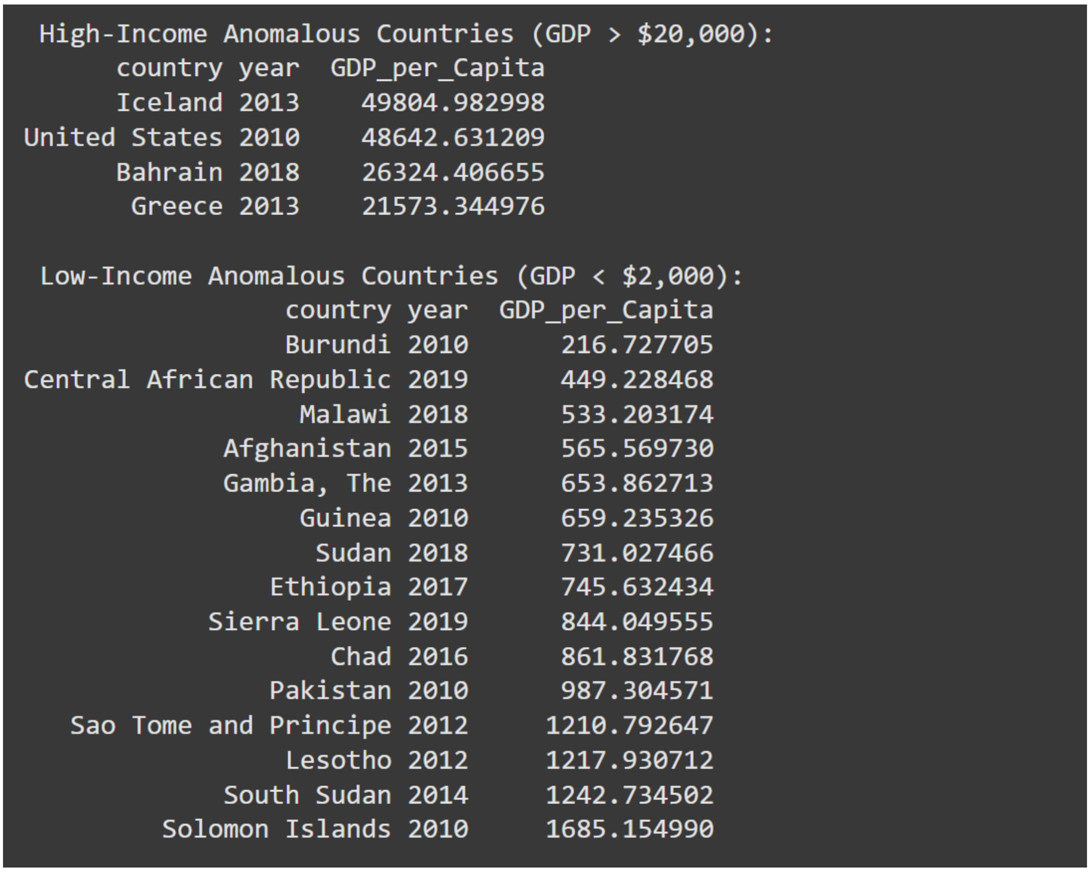
📊 Median Indicator Values per Cluster (Readable Names):

Cluster	Access to Electricity (%)	Capital Formation (% of GDP)	Consumer Price Index (total)	Education Expenditure (% of GDP)	Energy Use per Capita	FDI Net Inflows (US$)	Female Labor Participation Rate	Health Expenditure (% of GDP)	Infant Mortality Rate	Inflation (annual %)	Internet Users (%)	Life Expectancy (years)	Literacy Rate (%)	Male Labor Participation Rate	Tax Revenue (% of GDP)	Total Population	Unemployment Rate (%)	Urban Population (%)
0	39.1	22.64	118.34	3.25	394.54	7.178653e+08	60.49	4.54	51.70	5.01	9.00	61.04	62.14	75.34	11.55	20747641.0	4.53	34.86
1	100.0	21.10	108.64	4.92	3234.76	7.207000e+09	52.35	8.40	4.60	1.74	77.15	79.86	97.00	68.34	18.77	8935095.0	7.42	79.10
2	96.6	25.35	117.76	4.02	1142.14	2.521362e+09	47.80	5.14	17.46	3.53	39.70	71.99	91.99	74.56	13.85	20102781.5	6.20	54.34

```
💰 Average GDP per Capita by Cluster:
Cluster
0     1517.16
1    31759.11
2     6062.13
```

Figure 6: K-Means Clustering Results Visualized via PCA (colored by GDP per capita)

Figure 7: Anomalous Countries Identified by One-Class SVM



```
 High-Income Anomalous Countries (GDP > $20,000):
      country year  GDP_per_Capita
      Iceland 2013    49804.982998
United States 2010    48642.631209
      Bahrain 2018    26324.406655
       Greece 2013    21573.344976

 Low-Income Anomalous Countries (GDP < $2,000):
                 country year  GDP_per_Capita
                 Burundi 2010      216.727705
Central African Republic 2019      449.228468
                  Malawi 2018      533.203174
             Afghanistan 2015      565.569730
             Gambia, The 2013      653.862713
                  Guinea 2010      659.235326
                   Sudan 2018      731.027466
                Ethiopia 2017      745.632434
            Sierra Leone 2019      844.049555
                    Chad 2016      861.831768
                Pakistan 2010      987.304571
   Sao Tome and Principe 2012     1210.792647
                 Lesotho 2012     1217.930712
             South Sudan 2014     1242.734502
         Solomon Islands 2010     1685.154990
```

Figure 7: Countries Detected as Anomalous Based on Socio-Economic Performance

7.3 References

References

[1] ML Course by Andrew Ng on Coursera

[2] W3Schools.com

[3] Numpy Documentation

[4] Data Wrangling and Visualization with R part of Introduction to Data Science online book

[5] Deep Learning AI youtube playlist on regression