AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

FACULTY OF SCIENCE & TECHNOLOGY

COURSE: INTRODUCTION TO DATA SCIENCE

FALL 2025-2026

Section-E , Group-E

Supervised By

KAMRUN NAHER KOLI

Lecturer, Faculty
DEPARTMENT OF COMPUTER SCIENCE

FINAL TERM PROJECT

Title: Predicting Human Development Index (HDI) Category Using GDP and Population Data Scraped from Wikipedia.

SUBMITTED BY:

NAME	ID
FARDIN-AL-SEZAN	22-46868-1
MD. JANNATUL ADON	22-46887-1
TASMIA JAHAN MILA	22-46880-1

Date of submission: 6th January, 2026

Abstract

This project implements a complete data science workflow to predict the Human Development Index (HDI) category of countries using economic and demographic indicators collected through web scraping. Data were scraped from real Wikipedia tables containing nominal GDP, population, and HDI values using R and the rvest package. The scraped data were cleaned and integrated into a unified dataset consisting of 186 countries. Feature engineering was performed by computing GDP per capita and applying logarithmic transformations to reduce skewness. Exploratory Data Analysis (EDA) demonstrated a strong positive relationship between GDP per capita and HDI. Two classification models—Multinomial Logistic Regression and Random Forest—were trained and evaluated. The Multinomial Logistic Regression model achieved the highest performance, obtaining 83.33% accuracy on the test dataset. The findings indicate that GDP per capita is the most influential predictor of HDI category, while population alone has limited predictive impact. This report satisfies all IDS project requirements, including web scraping, preprocessing, EDA, modeling, evaluation, and result interpretation.

1. Research Objective

1.1 Objective Statement

The main objective of this project is: **To predict a country's HDI category (Low, Medium, High) using nominal GDP and population data scraped from Wikipedia.**

This is formulated as a **multi-class classification problem**, where HDI category is the target variable and GDP/population features serve as predictors.

1.2 Scope of the Study

- Use real data scraped from Wikipedia tables.
- Clean and preprocess the scraped data.
- Perform exploratory data analysis and visualization.
- Apply feature engineering to strengthen predictive power.
- Train and evaluate machine learning classification models.
- Interpret results and draw conclusions.

2. Data Sources and Web Scraping

Web scraping is a mandatory requirement for this project.

2.1 Data Sources

The dataset was collected from the following Wikipedia pages:

1. **List of countries by GDP (nominal)** – GDP values

2. **List of countries and dependencies by population** – population values
3. **List of countries by Human Development Index** – HDI scores

2.2 Web Scraping Tools

The following R packages were used:

- rvest to download and parse HTML pages (read_html())
- html_table(fill=TRUE) to extract tables
- dplyr, tidyr, stringr for cleaning and transformation
- janitor for standardized column naming (clean_names())
- ggplot2 and caret are also used

2.3 Rationale for Choosing Wikipedia

Wikipedia was chosen because it provides open-access, structured tables with minimal scraping restrictions. Additionally, it contains comprehensive country-level data with enough samples to perform meaningful analysis and model training.

3. Methodology

This project followed a complete IDS workflow consisting of: **Data Acquisition → Data Cleaning → Data Integration → Feature Engineering → EDA → Modeling → Evaluation → Interpretation.**

3.1 Data Acquisition (Scraping GDP, Population, HDI)

Each Wikipedia page was accessed using read_html(), and tables were extracted using html_table(fill=TRUE). Because each page contained multiple tables, the correct table index was identified by previewing the tables using head().

```
# STEP 2: Scrape GDP Table from Wikipedia

gdp_url <- "https://en.wikipedia.org/wiki/List_of_countries_by_GDP_(nominal)"
gdp_page <- read_html(gdp_url) # Read the HTML
gdp_tables <- gdp_page %>% html_table(fill = TRUE) # Extract ALL tables from the page
length(gdp_tables) # Check how many tables were found

# View the few values from table to identify the correct one
head(gdp_tables[[3]])
```

```
> head(gdp_tables[[3]])
# A tibble: 6 × 4
  `Country/Territory` `IMF(2025)[6]` `World Bank(2024)[7]` `United Nations(2023)[8]`
  <chr>               <chr>          <chr>                 <chr>
1 World               117,165,394    111,326,370           100,834,796
2 United States       30,615,743     28,750,956            27,720,700
3 China[n 1]          19,398,577     18,743,803            17,794,782
4 Germany             5,013,574      4,685,593             4,525,704
5 Japan               4,279,828      4,027,598             4,204,495
6 India               4,125,213      3,909,892             3,575,778
> |
```

```
# STEP 4: Scrape Population Table from Wikipedia

pop_url <- "https://en.wikipedia.org/wiki/List_of_countries_and_dependencies_by_population"
pop_page <- read_html(pop_url) # Read HTML
pop_tables <- pop_page %>% html_table(fill = TRUE) # Extract all tables
length(pop_tables) # Check how many tables found

# Preview tables to find the correct population table
head(pop_tables[[3]])
```

```
> head(pop_tables[[1]])
# A tibble: 6 × 6
  Location      Population    `% ofworld` Date        Source (official or fromthe United Nati…¹ Notes
  <chr>         <chr>         <chr>       <chr>       <chr>                                      <chr>
1 World         8,232,000,000 100%        13 Jun 2025 UN projection[1][3]                        ""
2 India         1,417,492,000 17.2%       1 Jul 2025  Official projection[4]                     "[b]"
3 China         1,408,280,000 17.1%       31 Dec 2024 Official estimate[5]                       "[c]"
4 United States 340,110,988   4.1%        1 Jul 2024  Official estimate[6]                       "[d]"
5 Indonesia     284,438,782   3.5%        30 Jun 2025 National annual projection[7]              ""
6 Pakistan      241,499,431   2.9%        1 Mar 2023  2023 census result[8]                      "[e]"
```

```
# STEP 8A: Scrape HDI Table from Wikipedia
hdi_url <- "https://en.wikipedia.org/wiki/List_of_countries_by_Human_Development_Index"
hdi_page <- read_html(hdi_url) # Read page
hdi_tables <- hdi_page %>% html_table(fill = TRUE) # Extract all tables
length(hdi_tables) # Check how many tables found

# Preview first table
head(hdi_tables[[2]])
```

```
> head(hdi_tables[[2]])
# A tibble: 6 × 5
   Rank Changesince2015 `Country or territory` `HDI value` `%annual growth(2010-2023)`
  <int> <chr>           <chr>                        <dbl> <chr>
1     1 "(2)"           Iceland                      0.972 0.28%
2     2 "(1)"           Norway                       0.97  0.25%
3     2 ""              Switzerland                  0.97  0.24%
4     4 "(2)"           Denmark                      0.962 0.35%
5     5 "(1)"           Germany                      0.959 0.19%
6     5 ""              Sweden                       0.959 0.38%
```

3.2 Data Cleaning

After scraping, the extracted tables contained several formatting issues such as commas inside numeric values, missing values, and footnotes attached to country names (e.g., [n 1], [b], etc.). Therefore, separate cleaning procedures were applied for each dataset (GDP, Population, and HDI) to ensure consistency and usability for analysis and modeling.

3.2.1 GDP Data Cleaning

The GDP table contained country names with footnotes and GDP values with commas. Some countries also had missing GDP values (shown as "—N/a") which caused NA after conversion.

The following cleaning steps were applied:

1. **Country name cleaning:** Footnotes (e.g., [n 1]) were removed using regular expressions: str_replace_all(country, "\\[.*?\\]", "")
2. **GDP numeric conversion:**
   - Commas were removed from GDP values.
   - Only numeric digits were extracted using str_extract("\\d+") to avoid non-numeric symbols.
   - GDP was converted to numeric using as.numeric().
3. **Handling missing GDP values:** Rows with missing or non-available GDP values were filtered out using: filter(!is.na(gdp_imf))

This produced a cleaned GDP dataset (gdp_final) with **195 countries**.

```r
# STEP 3: Clean GDP Table (IMF column)

gdp_raw <- gdp_tables[[3]]
gdp_raw <- gdp_raw %>% clean_names() # Clean column names
names(gdp_raw) # Check column names
# Rename important columns for clarity
gdp_clean <- gdp_raw %>%
  rename(
    country = country_territory,
    gdp_imf = imf_2025_6
  ) %>%
  # Remove footnote text like [n 1], [6], etc. from country names
  mutate(
    country = str_replace_all(country, "\\[.*?\\]", ""),
    country = str_trim(country)
  ) %>%
  # Remove commas and convert GDP to numeric
  mutate(
    gdp_imf = str_replace_all(gdp_imf, ",", ""),
    gdp_imf = as.numeric(gdp_imf)
  )
head(gdp_clean, 10) # Preview results
sum(is.na(gdp_clean$gdp_imf)) # Check missing values
summary(gdp_clean$gdp_imf)

# STEP 3.1: Improved GDP Cleaning (Fix NA issue)
gdp_clean <- gdp_raw %>%
  rename(
    country = country_territory,
    gdp_imf = imf_2025_6
  ) %>%
  mutate(
    # clean country names
    country = str_replace_all(country, "\\[.*?\\]", ""),
    country = str_trim(country),

    # remove commas and keep only digits in GDP
    gdp_imf = str_replace_all(gdp_imf, ",", ""),
    gdp_imf = str_extract(gdp_imf, "\\d+"),
    gdp_imf = as.numeric(gdp_imf)
  )
sum(is.na(gdp_clean$gdp_imf)) # Check how many NAs remain
gdp_clean %>% filter(is.na(gdp_imf)) %>% head(10) # See which countries have missing GDP

# STEP 3.2: Final GDP Dataset (Remove missing GDP)
gdp_final <- gdp_clean %>%
  filter(!is.na(gdp_imf)) %>%
  select(country, gdp_imf)
# Preview
head(gdp_final, 10)
nrow(gdp_final)
```

```
> head(gdp_final, 10)
# A tibble: 10 × 2
   country          gdp_imf
   <chr>              <dbl>
 1 World          117165394
 2 United States   30615743
 3 China           19398577
 4 Germany          5013574
 5 Japan            4279828
 6 India            4125213
 7 United Kingdom   3958780
 8 France           3361557
 9 Italy            2543677
10 Russia           2540656
> nrow(gdp_final)
[1] 195
```

3.2.2 Population Data Cleaning

The population table also contained:

- commas in population values
- footnotes in country names
- some rows that were not required for modeling

The following cleaning steps were applied:

1. **Country name cleaning:**Footnotes such as [b], [c], etc. were removed using: str_replace_all(country, "\\[.*?\\]", "")
2. **Population numeric conversion:**
    - Commas were removed: str_replace_all(population, ",", "")
    - Only digits were extracted: str_extract(population, "\\d+")
    - Converted to numeric with as.numeric()
3. **Missing value handling:**Unlike GDP, the population table produced **no missing values**, verified using:sum(is.na(pop_clean$population)) .This returned **0**, meaning all extracted population values were valid.

This produced a cleaned population dataset (pop_final) with **240 rows**.

```
# STEP 5: Clean Population Table

pop_raw <- pop_tables[[1]]
pop_raw <- pop_raw %>% clean_names() # Clean column names
names(pop_raw) # Check column names
# Create clean population dataset
pop_clean <- pop_raw %>%
  rename(
    country = location,
    population = population
  ) %>%
  mutate(
    # Remove footnotes like [b], [c]
    country = str_replace_all(country, "\\[.*?\\]", ""),
    country = str_trim(country),

    # Remove commas, keep digits only, convert to numeric
    population = str_replace_all(population, ",", ""),
    population = str_extract(population, "\\d+"),
    population = as.numeric(population)
  )
sum(is.na(pop_clean$population)) # Check missing values
# Keep final usable rows
pop_final <- pop_clean %>%
  filter(!is.na(population)) %>%
  select(country, population)
# Preview
head(pop_final, 10)
nrow(pop_final)
```

```
> head(pop_final, 10)
# A tibble: 10 × 2
   country        population
   <chr>               <dbl>
 1 World          8232000000
 2 India          1417492000
 3 China          1408280000
 4 United States   340110988
 5 Indonesia       284438782
 6 Pakistan        241499431
 7 Nigeria         223800000
 8 Brazil          213421037
 9 Bangladesh      169828911
10 Russia          146028325
> nrow(pop_final)
[1] 240
>
```

3.2.3 HDI Data Cleaning

The HDI table contained country names and HDI values, but some values can contain formatting and footnotes.

Cleaning steps included:

1. **Country name cleaning:**Footnotes were removed and names were trimmed using: str_replace_all(country, "\\[.*?\\]", "") , str_trim(country)
2. **HDI numeric conversion:**HDI values were converted directly to numeric using: hdi = as.numeric(hdi)
3. **Missing value handling:**Missing HDI rows were removed using: filter(!is.na(hdi)) No missing HDI remained after cleaning.

This produced a clean HDI dataset (hdi_final) with **193 rows**.

```
# STEP 8B: Clean HDI Table
hdi_raw <- hdi_tables[[2]]
hdi_raw <- hdi_raw %>% clean_names()
names(hdi_raw) # Check columns
hdi_clean <- hdi_raw %>%
  rename(
    country = country_or_territory,
    hdi = hdi_value
  ) %>%
  mutate(
    # Remove footnotes if any
    country = str_replace_all(country, "\\[.*?\\]", ""),
    country = str_trim(country),

    # Ensure numeric
    hdi = as.numeric(hdi)
  )
# Remove missing HDI values if any
hdi_final <- hdi_clean %>%
  filter(!is.na(hdi)) %>%
  select(country, hdi)
# Preview
head(hdi_final, 10)
nrow(hdi_final)
# Check any missing HDI
sum(is.na(hdi_final$hdi))
summary(hdi_final$hdi)
```

```
> head(hdi_final, 10)
# A tibble: 10 × 2
   country       hdi
   <chr>       <dbl>
 1 Iceland     0.972
 2 Norway      0.97 
 3 Switzerland 0.97 
 4 Denmark     0.962
 5 Germany     0.959
 6 Sweden      0.959
 7 Australia   0.958
 8 Netherlands 0.955
 9 Hong Kong   0.955
10 Belgium     0.951
> nrow(hdi_final)
[1] 193
```

3.3 Data Integration (Merging Datasets)

The cleaned GDP and population datasets were merged by country name using inner_join(). Initial merging resulted in 188 countries due to naming mismatches across the tables. To resolve this issue, country names were standardized using a mapping approach (e.g., “Democratic Republic of the Congo” → “DR Congo”). After correction, the merged dataset increased to 193 countries.

Subsequently, the HDI dataset was merged, producing a final dataset of 186 countries with no missing values.

```
> head(data_merged1, 10)
# A tibble: 10 × 3
   country         gdp_imf population
   <chr>             <dbl>      <dbl>
 1 United States  30615743  340110988
 2 China          19398577 1408280000
 3 Germany         5013574   83497147
 4 Japan           4279828  123190000
 5 India           4125213 1417492000
 6 United Kingdom  3958780   69487000
 7 France          3361557   68736000
 8 Italy           2543677   58925596
 9 Russia          2540656  146028325
10 Canada          2283599   41575585
> nrow(data_merged1)
[1] 188
```

```
# Fix population country names to match GDP country names
pop_final_fixed <- pop_final2 %>%
  mutate(country = case_when(
    country == "Democratic Republic of the Congo" ~ "DR Congo",
    country == "Republic of the Congo" ~ "Congo",
    country == "Hong Kong (China)" ~ "Hong Kong",
    country == "Puerto Rico (US)" ~ "Puerto Rico",
    country == "Macau (China)" ~ "Macau",
    TRUE ~ country
  ))

# Merge again after fixing names
data_merged2 <- inner_join(gdp_final2, pop_final_fixed, by = "country")
nrow(data_merged2)
head(data_merged2, 10)
```

```
> head(data_merged2, 10)
# A tibble: 10 × 3
   country        gdp_imf population
   <chr>            <dbl>      <dbl>
 1 United States 30615743  340110988
 2 China         19398577 1408280000
 3 Germany        5013574   83497147
 4 Japan          4279828  123190000
 5 India          4125213 1417492000
 6 United Kingdom 3958780   69487000
 7 France         3361557   68736000
 8 Italy          2543677   58925596
 9 Russia         2540656  146028325
10 Canada         2283599   41575585
> nrow(data_merged2)
[1] 193
```

```
# STEP 9: Merge HDI with GDP + Population

# Merge all into one dataset
final_data <- inner_join(data_merged2, hdi_final, by = "country")
nrow(final_data)
head(final_data, 10)

# Check missing values
colSums(is.na(final_data))

summary(final_data)
```

```
> nrow(final_data)
[1] 186
> head(final_data, 10)
# A tibble: 10 × 4
   country        gdp_imf population   hdi
   <chr>            <dbl>      <dbl> <dbl>
 1 United States 30615743  340110988 0.938
 2 China         19398577 1408280000 0.797
 3 Germany        5013574   83497147 0.959
 4 Japan          4279828  123190000 0.925
 5 India          4125213 1417492000 0.685
 6 United Kingdom 3958780   69487000 0.946
 7 France         3361557   68736000 0.92
 8 Italy          2543677   58925596 0.915
 9 Russia         2540656  146028325 0.832
10 Canada         2283599   41575585 0.939
```

4. Dataset Description

After merging all sources, the final dataset included **186 countries** with the following main variables:

Variable	**Description**	**Type**
country	Country name	Categorical
gdp_imf	Nominal GDP (IMF 2025)	Numeric
population	Population estimate	Numeric
hdi	HDI score	Numeric

```
> summary(final_data)
   country             gdp_imf          population       
 Length:186         Min.   :      58   Min.   :1.064e+04  
 Class :character   1st Qu.:   14707   1st Qu.:1.824e+06  
 Mode  :character   Median :   47911   Median :9.091e+06  
                    Mean   :  621499   Mean   :4.171e+07  
                    3rd Qu.:  311056   3rd Qu.:3.173e+07  
                    Max.   :30615743   Max.   :1.417e+09  
      hdi        
 Min.   :0.3880  
 1st Qu.:0.6295  
 Median :0.7650  
 Mean   :0.7451  
 3rd Qu.:0.8620  
 Max.   :0.9720  
>
```

5. Feature Engineering and Target Definition

Feature engineering was performed to improve model performance and to satisfy preprocessing requirements.

5.1 Engineered Features

1. **GDP per capita** was calculated to represent economic output per person:

$$GDP\_per\_capita = \frac{GDP \times 10^6}{Population}$$

2. **Log transformations** were applied to reduce skewness:

- log GDP (log_gdp)
- log population (log_population)
- log GDP per capita (log_gdp_per_capita)

5.2 Target Variable: HDI Category

HDI values were converted into three categories:

- **Low:** HDI < 0.55
- **Medium:** 0.55 ≤ HDI < 0.70
- **High:** HDI ≥ 0.70

Class distribution:

- High = 122
- Medium = 41
- Low = 23

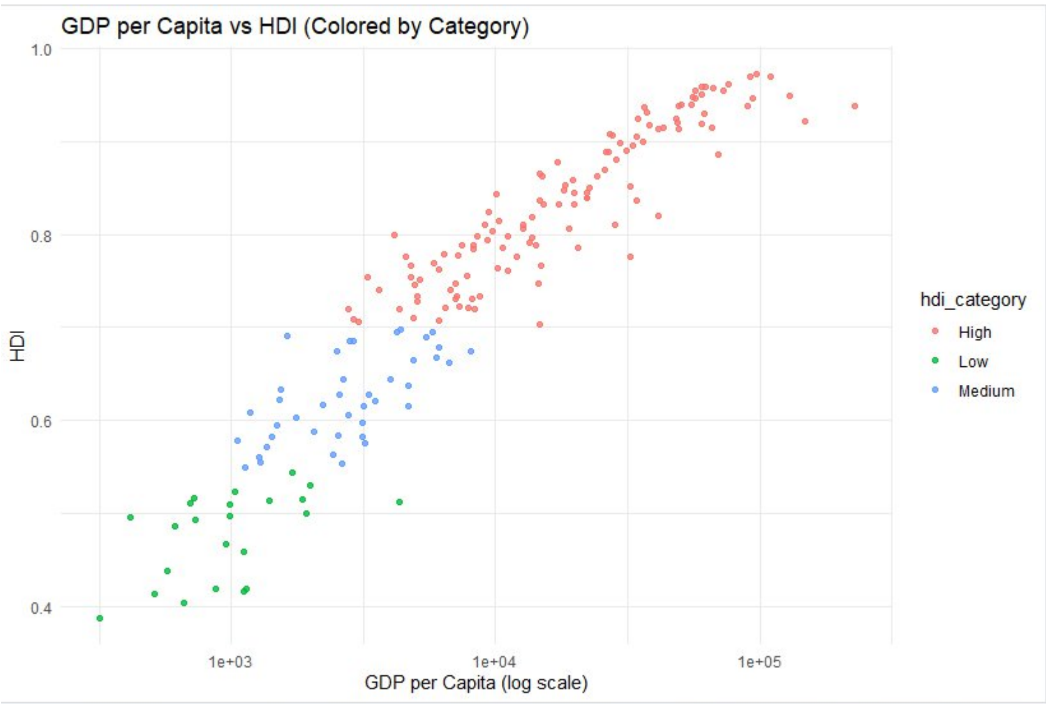
```
> head(final_data2, 10)
# A tibble: 10 × 9
   country         gdp_imf population   hdi gdp_per_capita log_gdp log_population log_gdp_per_capita hdi_category
   <chr>             <dbl>      <dbl> <dbl>          <dbl>   <dbl>          <dbl>              <dbl> <chr>
 1 United States  30615743  340110988 0.938         90017.    17.2           19.6              11.4  High
 2 China          19398577 1408280000 0.797         13775.    16.8           21.1               9.53 High
 3 Germany         5013574   83497147 0.959         60045.    15.4           18.2              11.0  High
 4 Japan           4279828  123190000 0.925         34742.    15.3           18.6              10.5  High
 5 India           4125213 1417492000 0.685          2910.    15.2           21.1               7.98 Medium
 6 United Kingdom  3958780   69487000 0.946         56972.    15.2           18.1              11.0  High
 7 France          3361557   68736000 0.92          48905.    15.0           18.0              10.8  High
 8 Italy           2543677   58925596 0.915         43168.    14.7           17.9              10.7  High
 9 Russia          2540656  146028325 0.832         17398.    14.7           18.8               9.76 High
10 Canada          2283599   41575585 0.939         54926.    14.6           17.5              10.9  High
> table(final_data2$hdi_category)

  High    Low Medium 
   122     23     41 
```

6. Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) was conducted to understand the distribution of key variables, detect patterns, and examine relationships between predictors and the target variable (HDI). Both visualizations and statistical summaries were used to interpret how GDP, population, and GDP per capita relate to human development. The findings from this section guided feature engineering and model selection.

6.1 Relationship Between GDP per Capita and HDI

To examine the direct association between economic wellbeing and human development, a scatter plot of **GDP per capita vs HDI** was created.



Interpretation:
The scatter plot shows a clear positive trend: countries with higher GDP per capita tend to have higher HDI values. This indicates that national income per person is strongly associated with development outcomes such as health, education, and living standards. A log scale is used because GDP per capita is highly skewed, and the transformation improves visualization and interpretability.

Key observation: As GDP per capita increases, HDI rises steadily, suggesting GDP per capita is a strong predictor of HDI.

6.2 GDP per Capita Across HDI Categories

To compare economic differences among development groups, a boxplot was created showing **GDP per capita distribution across HDI categories (Low, Medium, High)**.

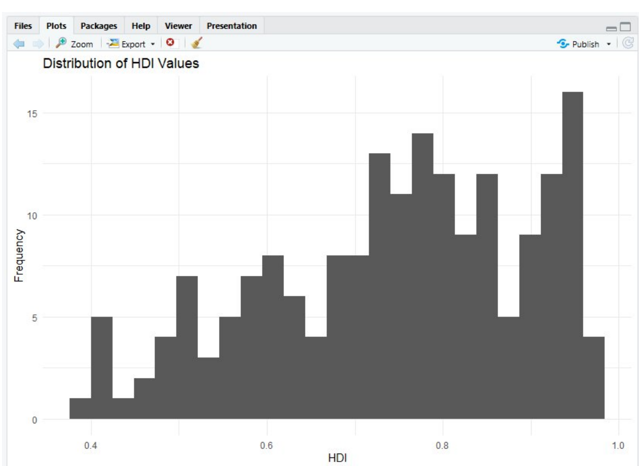
Interpretation:
The boxplot clearly shows that **High HDI countries** have much higher GDP per capita compared to Medium and Low HDI countries. The median and interquartile range for High HDI countries are significantly larger, indicating stronger economic conditions. Low HDI countries have the lowest GDP per capita distribution, reflecting limited economic productivity per person.

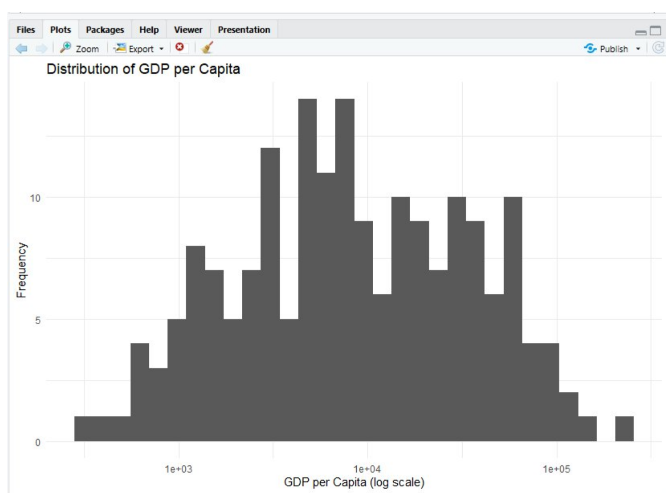
Key observations:

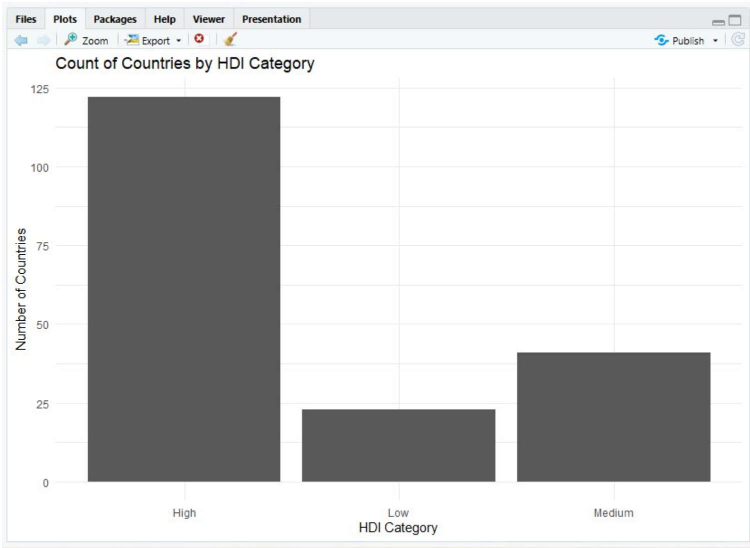
- High HDI → highest GDP per capita
- Low HDI → lowest GDP per capita
- The separation between categories supports the classification approach in this project.

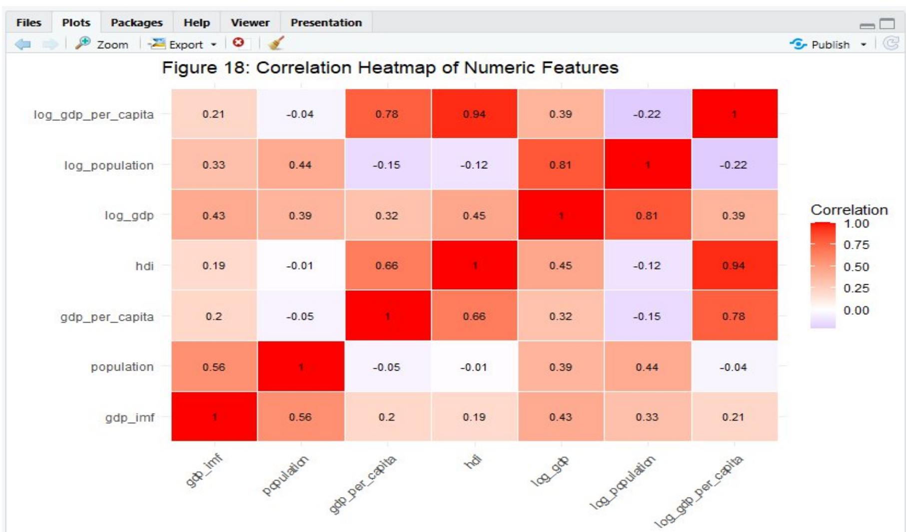
6.3 Correlation Analysis Among Numerical Features

A correlation matrix was computed to quantify relationships among numeric variables including GDP, population, GDP per capita, HDI, and log-transformed features.









Interpretation:

The correlation results indicate that the strongest association exists between **HDI and log GDP per capita (r = 0.945)**, showing that this feature is the most informative predictor in the dataset. GDP per capita also shows a moderate positive correlation with HDI (r = 0.657). In contrast, population demonstrates almost no correlation with HDI (r ≈ -0.01), suggesting that development level is not determined by country size alone.

Key correlation findings:

- **HDI vs log GDP per capita: 0.945 (very strong)**
- **HDI vs GDP per capita: 0.657 (moderate)**
- **HDI vs population: -0.011 (negligible)**

6.4 Summary of Key EDA Findings

Based on the EDA and correlation analysis, the following insights were obtained:

1. GDP per capita has a strong positive relationship with HDI.

2. High HDI countries have significantly higher GDP per capita than Medium and Low groups.
3. Log transformation improves interpretability and strengthens the relationship with HDI.
4. Population is not a strong predictor of HDI, meaning country size alone does not determine development outcomes.

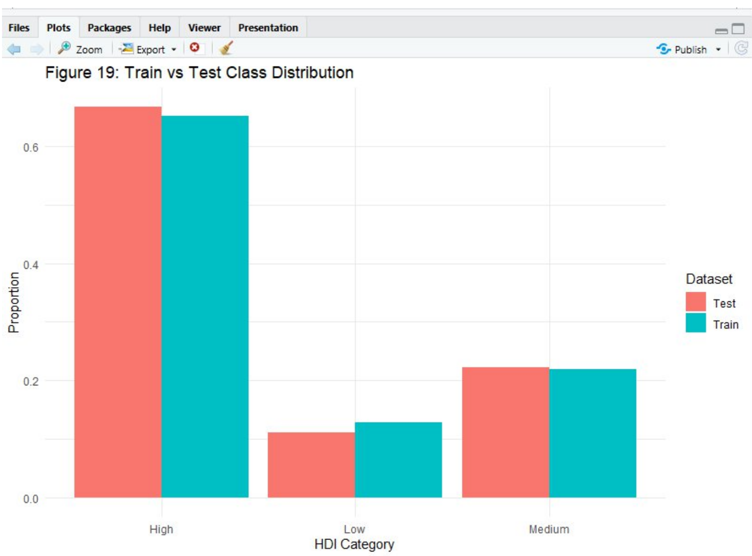
These results justify using **log GDP per capita** as a primary predictor in the machine learning models and support the classification objective of the project.

7. Modeling and Evaluation

This section presents the machine learning models used to predict HDI category and evaluates their performance. In accordance with IDS project requirements, the dataset was divided into training and testing subsets, classification models were trained, and model performance was assessed using accuracy, kappa statistics, and confusion matrices.

7.1 Train–Test Split

To ensure reliable evaluation, the dataset was split into **70% training** and **30% testing** samples. A **stratified sampling approach** was applied so that the proportions of HDI categories (Low, Medium, High) remained similar in both training and testing sets. This prevents biased performance evaluation caused by imbalanced class distribution.



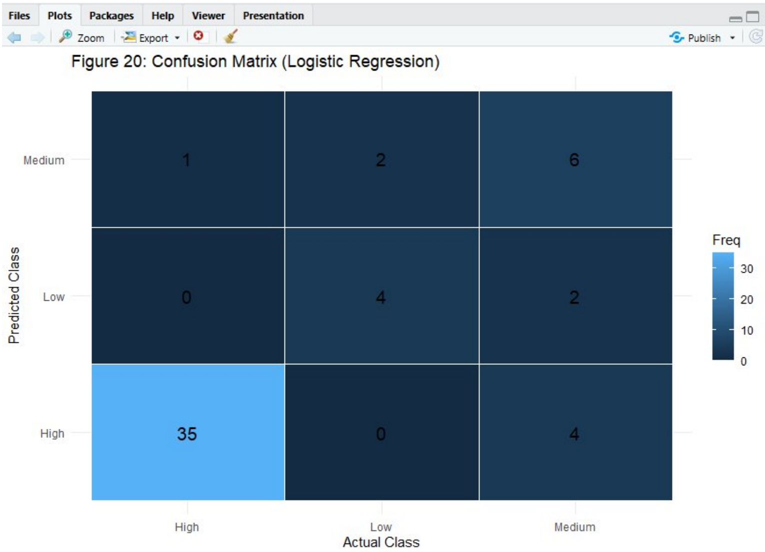
Key Point: The distribution of classes in the training and test sets remained very similar, ensuring fairness in model evaluation.

7.2 Model 1: Multinomial Logistic Regression

Multinomial Logistic Regression was selected as the baseline model because it is interpretable, efficient, and suitable for multi-class classification. The model was trained using cross-validation to improve generalization and avoid overfitting. The predictors used were log-transformed GDP per capita, GDP, and population.

Results :Test Accuracy: 83.33%

The confusion matrix indicates that the model performed strongly overall, especially for predicting the **High** HDI category. Some misclassification occurred between **Medium** and **Low/High**, which is expected due to overlap in economic conditions among countries near category boundaries.



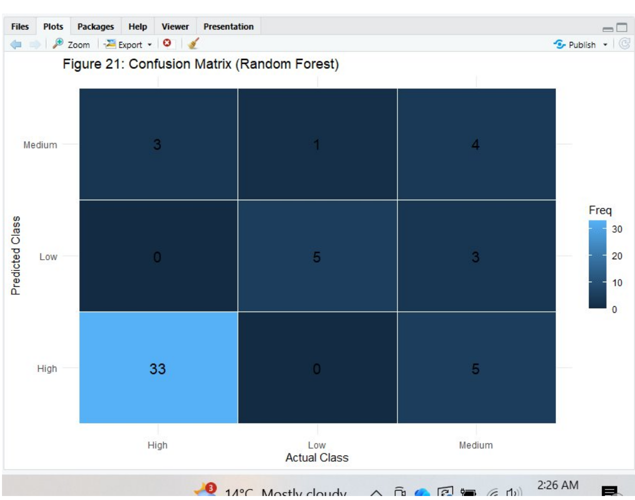
Key Observation: This model achieved the highest test accuracy and was therefore considered the best-performing model in this study.

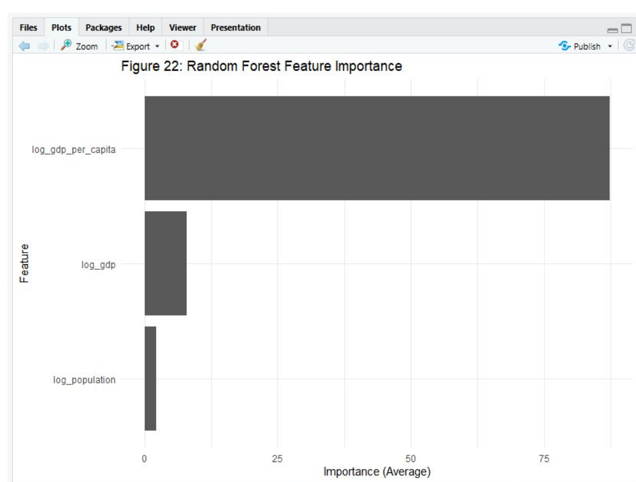
7.3 Model 2: Random Forest

Random Forest was trained as a non-linear ensemble model and was also used to identify feature importance. It can capture complex relationships among predictors and often performs well on structured tabular datasets.s

Results :Test Accuracy: 77.78%

Random Forest performed lower than Logistic Regression for this dataset. A possible reason is that the dataset contains a relatively small number of countries (186) and only three predictors, which limits the benefit of complex ensemble learning.



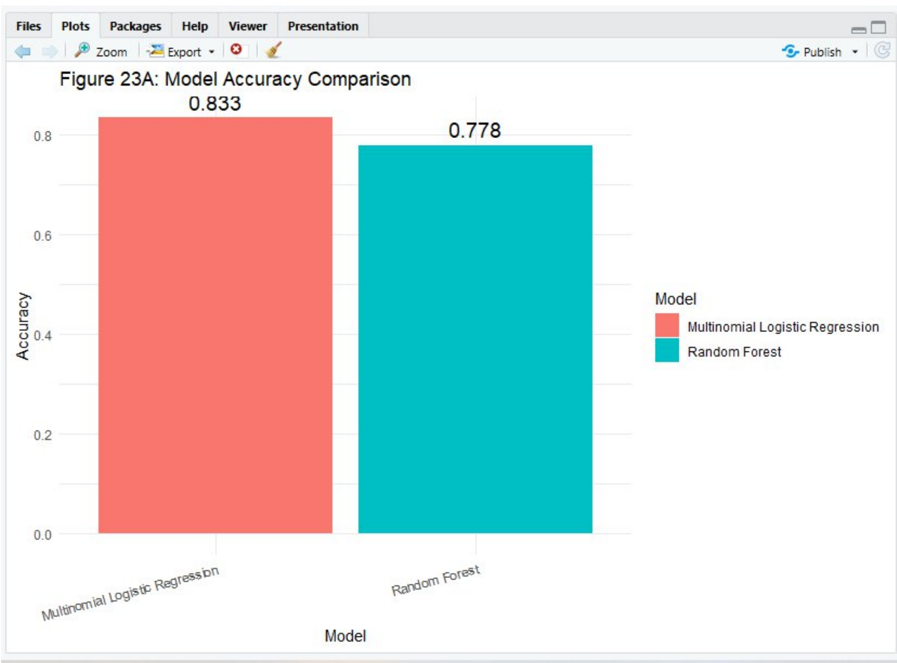


Key Observation: Random Forest feature importance highlighted that **log GDP per capita** is the most influential predictor for all categories.

7.4 Model Comparison

To select the best model, both models were compared using test accuracy and kappa statistics.

The Multinomial Logistic Regression model achieved higher accuracy and kappa, and therefore it was selected as the final model for predicting HDI category.



8. Results and Discussion

The results confirm that economic wellbeing measured by GDP per capita is a strong indicator of HDI category. EDA and correlation analysis demonstrated a consistent relationship between log GDP per capita and HDI. Modeling results reinforced these findings, as the logistic regression model achieved 83.33% accuracy. The medium category was harder to predict due to overlapping values between low and high development groups. Random Forest feature importance also highlighted log GDP per capita as the dominant predictor, supporting the analytical findings of this study.

9. Conclusion

This project successfully implemented a full IDS workflow using real-world data collected through web scraping. Data from Wikipedia were scraped, cleaned, merged, and transformed into a complete dataset containing 186 countries. Feature engineering improved interpretability and model performance by introducing GDP per capita and log-based transformations. EDA demonstrated strong relationships between GDP per capita and HDI. Among the two models evaluated, Multinomial Logistic Regression achieved the best classification performance with 83.33% accuracy. The project concludes that GDP per capita is the most influential factor among the considered features for predicting HDI category.

10. Limitations and Future Work

10.1 Limitations

- Wikipedia data may change over time, potentially affecting reproducibility.
- Only GDP and population were used; HDI depends on education and health indicators.
- The dataset contains class imbalance (High category dominates), which may affect classification of minority classes.

10.2 Future Work

- Include additional predictors such as life expectancy, education index, and GNI per capita.
- Analyze multiple years of data to study trends in HDI over time.

11. References

1. https://en.wikipedia.org/wiki/List_of_countries_by_GDP_(nominal)
2. https://en.wikipedia.org/wiki/List_of_countries_and_dependencies_by_population
3. https://en.wikipedia.org/wiki/List_of_countries_by_Human_Development_Index
4. R packages: rvest, ggplot2, caret, randomForest, dplyr