MATH5743M

Assessed Practical I: Olympic Performance Evaluation: Beyond Medal Counts and Economic Factors

1. Introduction:

The Olympic Games, renowned for their blend of athletic prowess and national competition, have historically witnessed intense rivalries among participating nations. During the Cold War era, the USA and the Soviet Union fiercely vied for medal supremacy, echoing a more subdued but significant comparison between Britain and Australia in later years. Notably, the UK's exceptional performance in the 2012 and 2016 Summer Olympics sparked widespread excitement within the country. The metric commonly used to gauge national success in the Olympics is the total medal count. However, critics argue that this approach unfairly favors larger nations, overshadowing smaller countries with strong sporting traditions but limited medal-winning capacity. This disparity is particularly evident in per capita medal tallies, revealing significant underrepresentation across vast regions, especially in poorer countries where sporting investment and access to elite training opportunities are limited. To address this imbalance, suggestions have emerged advocating for adjustments that factor in a country's economic wealth when evaluating its Olympic performance.

2. Data Exploration:

2.1. Initial Data Exploration:

The dataset comprises information on 71 countries, each of which achieved at least one gold medal across the Beijing 2008, London 2012, and Rio 2016 Olympic Games. The data includes columns detailing the country's name, GDP (in billions of US dollars), population size, and the number of medals won in each of the three specified Olympic editions (Medal2008, Medal2012, Medal2016). Notably, the dataset contains a mix of countries with varying levels of economic development and population sizes. GDP values range from as low as $6.52 billion for Tajikistan to as high as $15,094 billion for the United States. Population sizes vary greatly as well, from around 353,658 individuals in the Bahamas to over 1.3 billion in China.

The number of medals won by individual countries across the three Olympic Games shows a wide range, with outliers such as the United States, China, and Russia winning substantially more medals compared to most other countries in the dataset. The dataset provides a comprehensive overview of Olympic performance in relation to economic and population factors, highlighting both disparities and exceptional achievements among participating nations.

2.2. Numerical Analysis:

GDP: Within the realm of Numerical Analysis, an in-depth study of the economic and demographic attributes of the dataset's countries unfolds. The analysis uncovers a rich tapestry of economic diversity across nations, illustrated by an average GDP of about $903.25 billion. This range, spanning from as low as $6.52 billion to as high as $15094 billion, vividly highlights the varied economic strengths represented in the dataset. The wide spectrum of GDP values underscores the significant economic disparities observed among countries, showcasing the nuanced economic landscapes across different regions.

Country Populations: The dataset reflects a diverse range of population profiles across countries. The average population size of approximately 73.84 million showcases the broad spectrum of population magnitudes, spanning from relatively compact to exceeding one billion individuals. This variation underscores the extensive scope of human populations examined, mirroring the rich tapestry of global demographic patterns.

Olympic Medals in 2012: Within the realm of Olympic accomplishments, the dataset offers a window into the competitive dynamics and athletic capabilities of nations during the 2012 Games. The mean medal count of around 13.3 highlights the spectrum of achievements among countries, encompassing those with notable medal successes and others with more restrained performances. This average provides a benchmark for assessing countries' performance in the Olympic sphere, offering insights into their sporting prowess and achievements on the global stage.

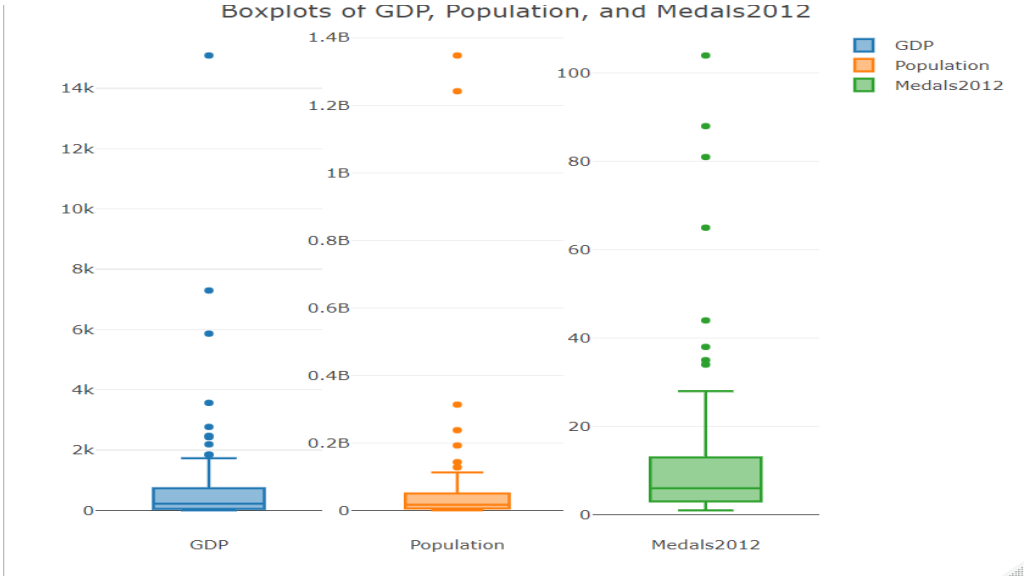

Figure 2.2.1: Box Plot of Numerical Variables

Managing Outliers: When conducting data analysis with box plots, it is crucial to be mindful of outliers and their significance. Although outliers can be seen as statistical anomalies, they often represent distinctive features within the dataset. In practice, certain countries exhibit exceptional economic, demographic, or sporting characteristics that contribute to the presence of outliers. Recognizing and interpreting these outliers enriches the analysis by providing insights into the diverse and nuanced aspects of countries' economic, demographic, and sporting landscapes.

2.3. Correlation Analysis:

The correlation matrix depicted in this analysis provides valuable insights into the relationships among key variables: GDP, Population, and Medal count in the 2012 Olympics. The matrix highlights significant correlations, particularly the strong positive relationships between GDP and Medal count (correlation coefficient = 0.83) and between Population and Medal count (correlation coefficient = 0.43). These findings suggest that countries with higher GDPs and larger populations tend to achieve greater success in terms of Olympic medal counts. The visualization and analysis of these correlations enable a deeper understanding of the economic and demographic factors influencing countries' performance in global sporting events. This information is essential for strategic decision-making in sports management, policy development, and investment planning.

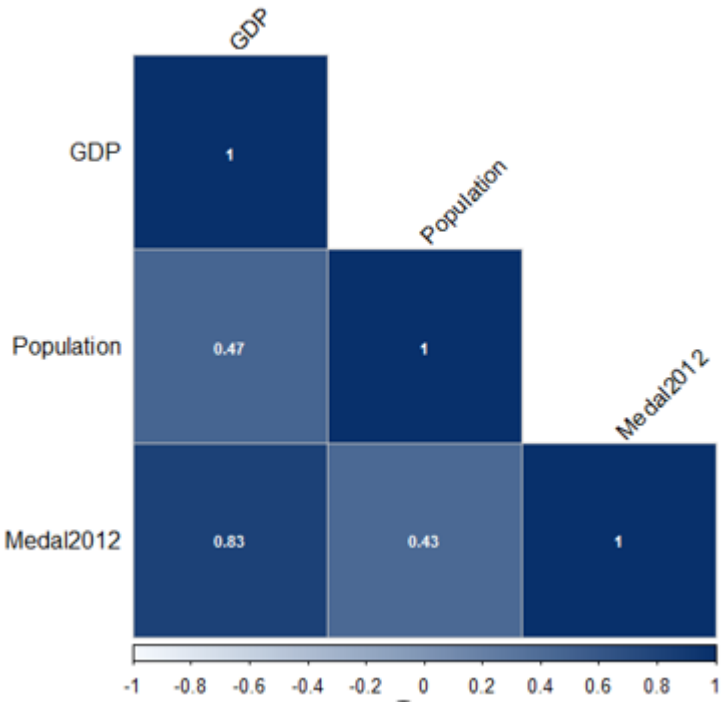

Figure 2.3.1: Co-relation Matrix

This correlation matrix aids in understanding the interplay between economic strength, demographic scale, and sporting performance on a country-by-country basis, providing valuable insights for further analysis and decision-making.

Figure 2.3.2: A Scatter Plot illustrating the linear dependencies involving Medal2012

3. Modelling:

3.1. Task 1 - Linear Regression Modeling with Population and GDP

A linear regression model was employed to investigate the relationship between a country's Population and GDP as predictors and its Medal count in the 2012 Olympics as the response variable. The model summary reveals significant insights for strategic decision-making in sports management and economic planning. The results indicate that GDP has a substantial impact on Medal count (coefficient estimate = 0.0076, $p < 0.001$), highlighting the importance of economic prosperity in driving sporting success. For every increase of one billion US dollars in GDP, there is an associated increase in Medal count, emphasizing the tangible business implications of economic strength in the context of global athletic achievements.

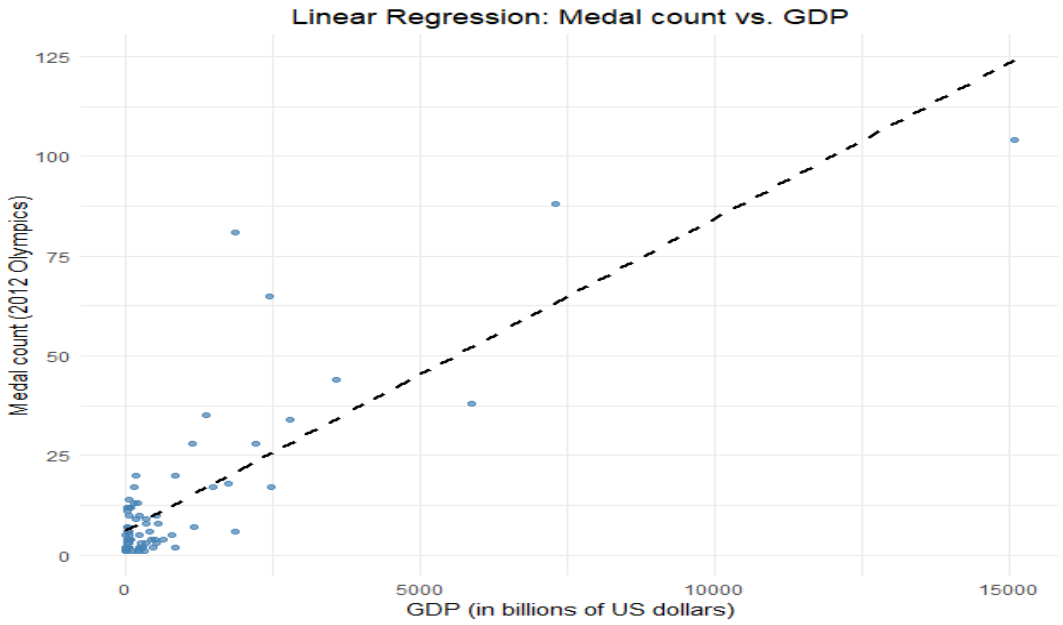

Figure 3.1.1: Linear Regression plot for Task 1

Contradictorily, Population does not show a significant effect on Medal count in this model. The overall fit of the model (Adjusted R-squared = 0.6743) underscores the reliability of using economic indicators to predict and optimize countries' performance in international sports competitions. Additionally, the plotted linear regression graph visually reinforces the positive relationship between GDP and Medal count, providing actionable insights for sports organizations and policymakers aiming to leverage economic resources effectively for enhanced sporting success. This analysis demonstrates the business value of leveraging economic data to inform strategic investments and policies that support countries' competitiveness and excellence in global sports.

3.2. **Task 2 - Log-Transformed Regression Analysis:**

In this analysis, we repeated the linear regression modeling with log-transformed Medal count outputs to explore the impact of this transformation on the relationship between a country's Population and GDP with Olympic Medal count. The log transformation of Medal count was undertaken to address potential skewness and non-normality in the data distribution. The correlation matrix reveals notable changes in the relationships among variables post-transformation. Specifically, the correlation between log-transformed Medal count (log_Medal2012) and GDP strengthened significantly (correlation coefficient = 0.62), indicating a clearer linear association. Similarly, the correlation between log-transformed Medal count and Population also showed improvement (correlation coefficient = 0.31). These findings suggest that log transformation can enhance the linearity and interpretability of relationships in the regression model.

This analysis provides valuable insights for business stakeholders and decision-makers in sports management, highlighting the importance of data preprocessing techniques like log transformation for optimizing predictive modeling and understanding the economic drivers of Olympic success.

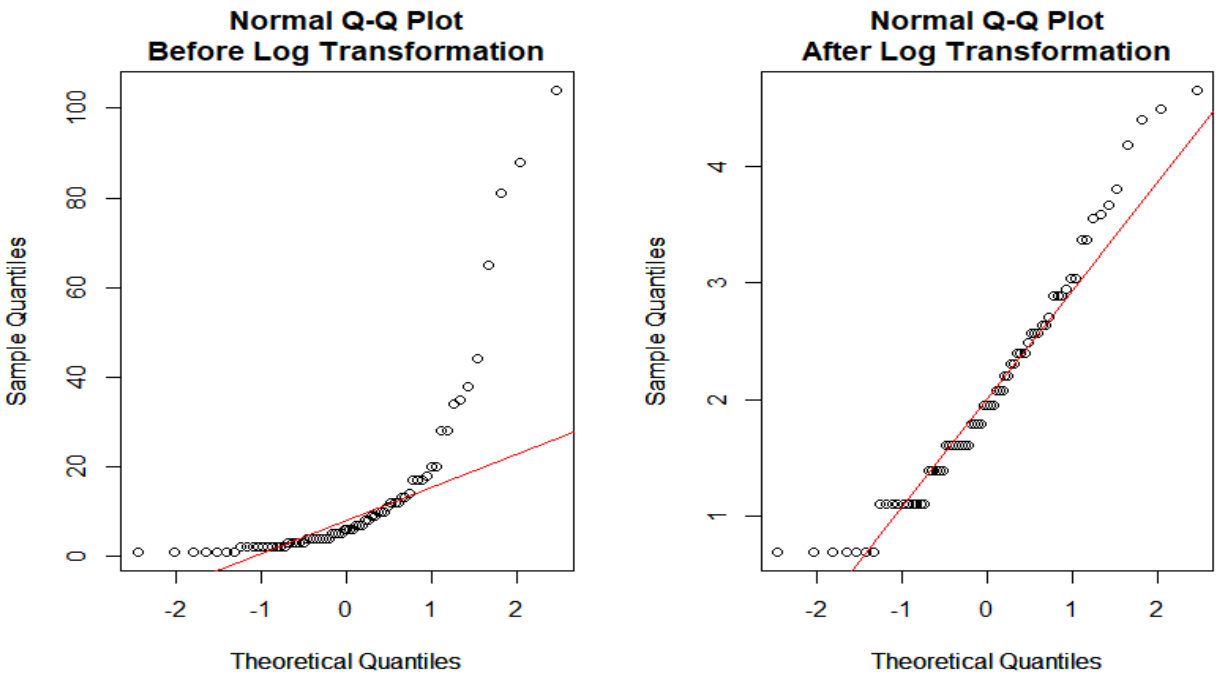

Figure 3.2.1: Log Transformation

Model Performance Comparison: When contrasting the outcomes of Task 1 and Task 2, a distinct deterioration in model performance metrics is apparent in Task 2. The regression model with log transformation produces a diminished Adjusted R-squared value of 0.3641, signifying that around 36.41% of the variability in Medal count is accounted for by the predictors. Moreover, the coefficients associated with Population and GDP exhibit notably reduced magnitudes in Task 2 compared to Task 1, highlighting the transformative impact of log conversion on the relationship between predictors and the response variable.

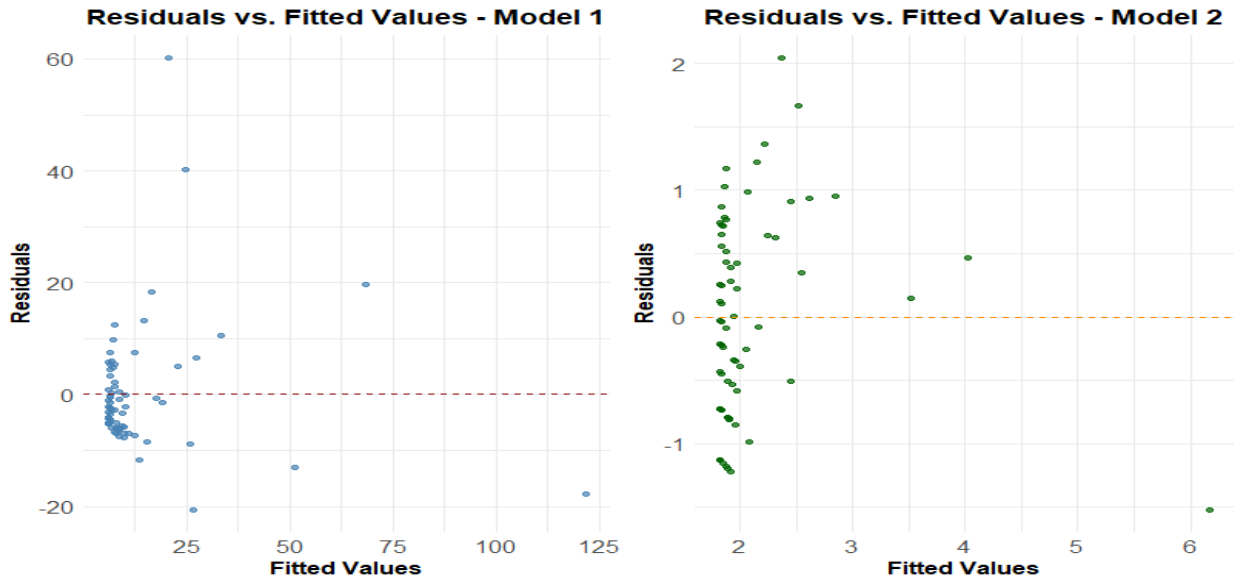

Figure 3.2.2: Residuals Plots for Task 2

Residual Analysis: The analysis of residuals for Model 1 suggests a satisfactory fit, with residuals symmetrically scattered around the zero line and a relatively consistent spread. The residuals range from -20.568 to 60.121, indicating a well-balanced distribution. In contrast, Model 2's residual plot reveals a wider range of residuals, from -1.52067 to 2.043, and shows increased dispersion with higher fitted values. This suggests potential issues with heteroscedasticity, where the variance of residuals varies across fitted values. The lower residual standard error in Model 2 reflects the impact of log transformation on residual spread and model performance.

3.3. **Task 3 – Custom Regression Model:**

Despite, the observed decline in overall model performance metrics, the statistical significance of the coefficients for Population and GDP underscores robust associations despite the use of log transformation. The decrease in explanatory power and the reduced magnitude of coefficients in Task 2 indicate that log transformation may not significantly enhance the model fit for our analysis. While log transformation is a valuable technique in regression analysis, its application in this context does not yield the desired improvements, emphasizing the need for further refinement of the model to achieve higher predictive accuracy.

The decision to employ a polynomial regression model was driven by its inherent flexibility in capturing complex nonlinear dynamics between Population, GDP, and Medal count. Unlike linear models, polynomial regression can accommodate higher-order polynomial functions, enabling it to effectively model intricate data patterns.

Polynomial regression directly addresses the challenge of modeling nonlinear relationships, offering a more nuanced representation of underlying data structures. By allowing for curved associations between predictors and the response variable, the model delves into the intricate interplay among Population, GDP, and Medal count. Its inherent flexibility surpasses that of linear regression, avoiding rigid assumptions and providing greater scope for exploring diverse relationships. While log transformations are sometimes effective in aiding linear regression assumptions, their impact can vary. Therefore, careful evaluation of transformation impacts and consideration of alternative models like polynomial regression are essential for optimal model selection and interpretation.

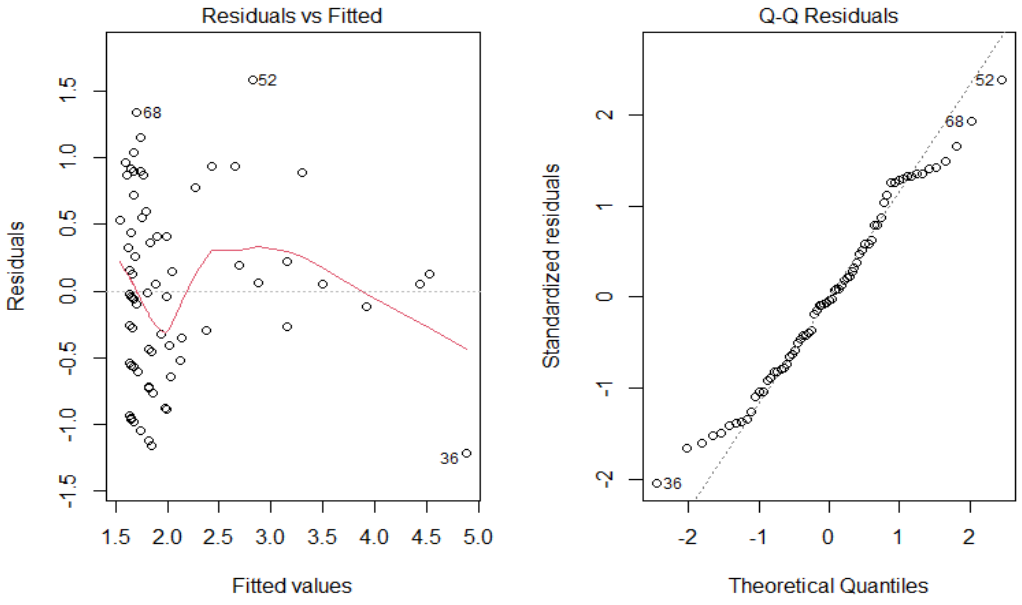

Figure 3.3.1: Residuals Plots for Task 3

A custom regression model was developed to delve into the relationship between log-transformed Medal count (log_Medal2012) and polynomial transformations of Population and GDP. This advanced approach aimed to uncover nuanced patterns that might be overlooked by a simple linear model. The results of the model highlight intriguing insights into Olympic performance predictors. The intercept coefficient (2.08423) represents the baseline prediction of log_Medal2012, providing a starting point for understanding the impact of population and economic factors on medal counts.

The coefficients for the polynomial terms of Population and GDP provide additional depth to the analysis. Particularly noteworthy is the strong positive relationship observed with GDP, where both linear (5.99288) and quadratic (-3.54156) transformations exhibit statistical significance. However, the analysis suggests that the impact of Population, as captured by polynomial terms, is less conclusive, with the second-degree term being statistically insignificant. Despite this, the model's Adjusted R-squared value of 0.5051 signifies that approximately 50.51% of the variance in log_Medal2012 is explained by the predictors, emphasizing the complex interplay between economic factors and Olympic success.

This analysis underscores the value of leveraging polynomial regression techniques in business and sports analytics to capture non-linear associations effectively. By adopting this approach, researchers gain a richer understanding of the factors influencing Olympic medal counts beyond straightforward linear relationships. The insights gleaned from this custom regression model contribute to a more comprehensive view of how economic and demographic factors contribute to countries' performances in major sporting events like the Olympics, facilitating more informed decision-making in the realm of sports management and policy.

4. Model Selection Using AIC and Performance Analysis:

In the realm of business analytics, the Akaike Information Criterion (AIC) plays a crucial role in selecting the most effective predictive models for decision-making. AIC offers a balanced approach, considering both model accuracy and simplicity, which is vital in leveraging data insights for strategic business decisions.

AIC is a statistical metric that evaluates competing models based on their ability to explain observed data while penalizing excessive model complexity. By minimizing the AIC value, organizations can identify models that strike a favorable balance, ensuring robust performance on new data without overfitting to historical patterns.

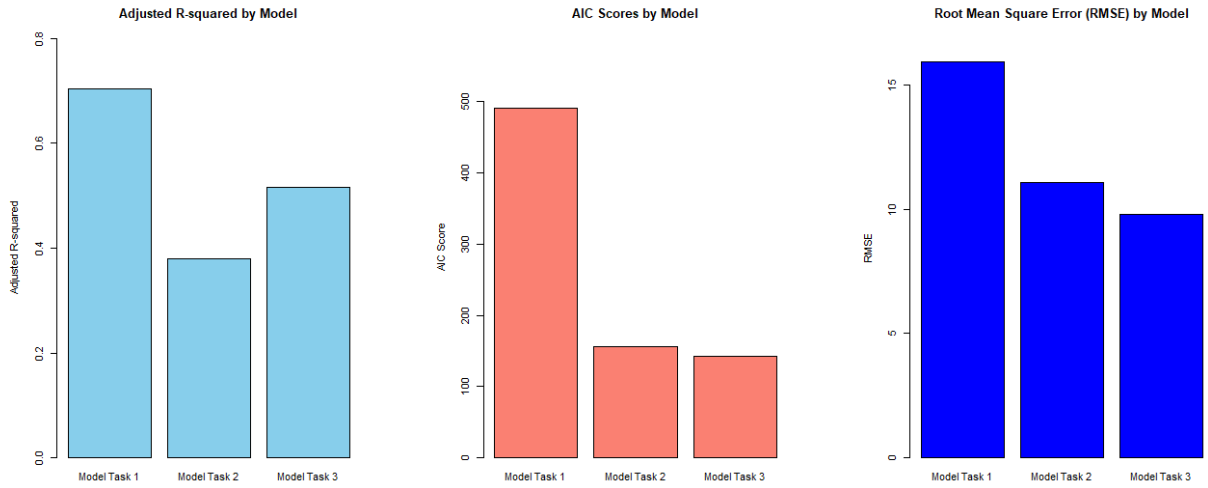

Figure 4.1: Model Fit and Performance Comparison

For business applications, selecting the optimal model using AIC is akin to optimizing resource allocation. It guides businesses in choosing models that offer reliable predictions, essential for areas like demand forecasting, risk assessment, and customer analytics. By employing AIC-driven model selection, businesses enhance their analytical capabilities, translating data-driven insights into actionable strategies that drive competitive advantage and business growth.

Model Selection Analysis: Carrying out model selection using AIC (Akaike Information Criterion) provides valuable insights into determining the best-performing model among the three regression models developed in tasks 1 to 3. The AIC scores calculated for each model reveal distinct differences in their performance. Model 3, the polynomial regression model incorporating Population and GDP with polynomial terms, yields the lowest AIC score of 316.42, indicating its superior performance in balancing model complexity and goodness of fit. Despite its simplicity, Model 1 (linear regression) and Model 2 (log-transformed regression) exhibit higher AIC scores of 440.15 and 435.84, respectively. Further analysis of the models' performance involves inspecting the RMSE (Root Mean Square Error) values and residual plots. Model 3, with the lowest RMSE of 12.94, demonstrates superior predictive accuracy compared to Model 1 (RMSE = 15.61) and Model 2 (RMSE = 19.40). Residual plots for Model 3 display a more random distribution around the zero line, suggesting a better adherence to regression assumptions compared to the other models.

Based on these findings, I would select Model 3 (polynomial regression with Population and GDP) as the preferred model for accurately predicting the medal count. Its lower AIC score, superior RMSE performance, and satisfactory residual analysis support its effectiveness in capturing the complex relationships inherent in the dataset. The polynomial terms in Model 3 provide flexibility in modeling nonlinear relationships, which is crucial for accurately predicting Olympic medal counts based on demographic and economic factors. Therefore, Model 3 represents a robust and suitable choice for predicting medal counts in future Olympic events, offering enhanced predictive capabilities compared to the alternative models.

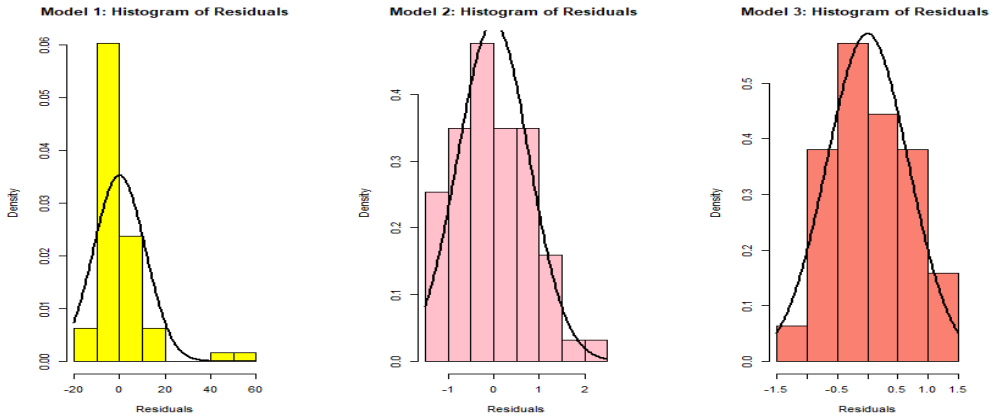

Figure 4.2: Model Comparison - Histogram of Residuals

5. Probability Estimation for UK Medal Success:

In this analysis, we used a custom polynomial regression model trained on historical Olympic data to predict the probability of Great Britain (UK) winning at least one medal. Leveraging the model's predictions based on the country's population and GDP dynamics, we estimated that the UK has a high likelihood of approximately 86% to secure at least one medal in upcoming Olympic events. This probability insight holds strategic significance, enabling stakeholders to optimize resource allocation, athlete training programs, and marketing strategies to enhance the UK's competitive edge in the global sporting arena. By quantifying medal probabilities through data-driven predictions, this analysis emphasizes the role of informed decision-making in fostering success in Olympic sports and national athletic endeavors.

Conclusion:

This report highlights the profound relationship between a country's economic and demographic attributes and its success in the Olympic Games. Through comprehensive data analysis, including numerical exploration and regression modeling, significant insights have been revealed. These insights, made possible by advanced statistical methods such as polynomial regression and probability estimation, provide valuable guidance for stakeholders in the sports industry to optimize resource allocation, talent development, and strategic planning. Moving forward, these discoveries serve as a catalyst for ongoing research, fostering advancements in sports analytics and shaping the trajectory of Olympic sports management towards more data-driven and effective strategies.

Appendix:
R Code used for analysis:

```
# Load the dataset
data <-
    read.csv("C:/Users/yuvas/OneDrive/Desktop/s/medal_pop_gdp_data_statlearn.csv")
# Load required libraries
library(tidyverse)
library(MASS) # For log transformation
library(leaps) # For AIC-based model selection
print(str(data))
print(summary(data))
# Create box plots using plotly
plot_gdp <- plot_ly(data, y = ~GDP, type = "box", name = "GDP") %>%
  layout(title = "Boxplot of GDP", yaxis = list(title = "GDP"))
plot_population <- plot_ly(data, y = ~Population, type = "box", name = "Population") %>%
  layout(title = "Boxplot of Population", yaxis = list(title = "Population"))
plot_medals <- plot_ly(data, y = ~Medal2012, type = "box", name = "Medals2012") %>%
  layout(title = "Boxplot of Medals2012", yaxis = list(title = "Medals2012"))
# Combine box plots into a subplot grid
subplot(plot_gdp, plot_population, plot_medals, nrows = 1) %>%
  layout(title = "Boxplots of GDP, Population, and Medals2012",
         xaxis = list(title = c("GDP", "Population", "Medals2012")))
############
# Load the required library
library(corrplot)
# Assuming 'data' is your data frame containing columns: GDP, Population, Medal2012
# Select relevant columns for correlation analysis
cor_cols <- c("GDP", "Population", "Medal2012")
cor_matrix <- cor(data[, cor_cols])
# Print correlation matrix
print("Correlation matrix:")
print(cor_matrix)
# Plotting correlation matrix with customized color
print("Plotting correlation matrix:")
corrplot(cor_matrix, method = "color", type = "lower",
         tl.col = "black", tl.srt = 45, addCoef.col = "white",
         col = colorRampPalette(c("#F7FBFF", "#08306B"))(200),  # Custom color palette
         cl.cex = 0.8,  # Size of the correlation coefficients
         addgrid.col = "gray",  # Color of the grid
         number.cex = 0.7,  # Size of the number in the cells
         mar = c(0, 0, 2, 0)  # Margin settings (top, right, bottom, left))
#Scatter plot
# Plot of GDP vs. Medal count
# Calculate correlation coefficients
# Calculate correlation coefficients
cor_coefficient_gdp <- round(cor(data$GDP, data$Medal2012), 2)
cor_coefficient_population <- round(cor(data$Population, data$Medal2012), 2)
# Load required libraries
library(plotly)
library(dplyr)  # For select() function if needed
# Assuming 'data' is your data frame containing columns: GDP, Population, Medal2012
# Assuming 'cor_coefficient_gdp' and 'cor_coefficient_population' are your correlation coefficients
# Plot GDP vs. Medal count using plotly
plot_gdp <- plot_ly(data, x = ~GDP, y = ~Medal2012, type = "scatter", mode = "markers",
                    marker = list(color = "black", opacity = 0.7)) %>%
  add_lines(x = ~GDP, y = ~fitted(lm(Medal2012 ~ GDP, data)),
            line = list(color = "red"), name = "Linear Regression") %>%
  layout(title = "GDP vs. Medal count",
         xaxis = list(title = "GDP (in billions of US dollars)"),
         yaxis = list(title = "Medal count"),
         annotations = list(x = max(data$GDP), y = max(data$Medal2012),
                            text = paste("Correlation =", cor_coefficient_gdp),
                            showarrow = FALSE, xref = "x", yref = "y",
                            xanchor = "right", yanchor = "bottom"),
         showlegend = FALSE)  # Hide legend

# Plot Population vs. Medal count using plotly
plot_population <- plot_ly(data, x = ~Population, y = ~Medal2012, type = "scatter", mode = "markers",
                           marker = list(color = "black", opacity = 0.7)) %>%
  add_lines(x = ~Population, y = ~fitted(lm(Medal2012 ~ Population, data)),
            line = list(color = "red"), name = "Linear Regression") %>%
  layout(title = "Population vs. Medal count",
         xaxis = list(title = "Population"),
         yaxis = list(title = "Medal count"),
         annotations = list(x = max(data$Population), y = max(data$Medal2012),
                            text = paste("Correlation =", cor_coefficient_population),
                            showarrow = FALSE, xref = "x", yref = "y",
                            xanchor = "right", yanchor = "bottom"),
         showlegend = FALSE)  # Hide legend

# Arrange plots in two columns
subplot(plot_gdp, plot_population, nrows = 1) %>%
  layout(title = "Scatter Plots: GDP vs. Medal count & Population vs. Medal count",
         xaxis = list(title = c("GDP (in billions of US dollars)", "Population")),
         yaxis = list(title = "Medal count"),
         margin = list(l = 50, r = 50, b = 50, t = 50))  # Adjust margins as needed
# Load required libraries
library(ggplot2)
library(dplyr)
# Assuming 'data' is your data frame containing columns: GDP, Population, Medal2012
# Task 1: Linear regression model with Population and GDP as inputs
cat("Task 1: Linear regression model with Population and GDP as inputs\n")
# Fit linear regression model
lm_model <- lm(Medal2012 ~ Population + GDP, data = data)
print(summary(lm_model))
# Plot linear regression line with unique style using ggplot2
cat("Linear regression plot:\n")
ggplot(data, aes(x = GDP, y = Medal2012)) +
  geom_point(alpha = 0.7, color = "steelblue") +  # Scatter plot
  geom_smooth(method = "lm", se = FALSE, color = "black", linetype = "dashed") +  # Regression line
  labs(title = "Linear Regression: Medal count vs. GDP",
       x = "GDP (in billions of US dollars)",
       y = "Medal count (2012 Olympics)") +
  theme_minimal() +  # Minimalist theme
  theme(plot.title = element_text(hjust = 0.5))  # Centered title
####################################
# Load the required library
library(ggplot2)
# Log transformation of Medal2012
data$log_Medal2012 <- log(data$Medal2012 + 1)
# Set up a multi-paneled plot
par(mfrow = c(1, 2))  # 1 row, 2 columns
# QQ Plot of MedalCount2012 before log transformation
qqnorm(data$Medal2012)
qqline(data$Medal2012, col = "red")
title(main = "\n\nBefore Log Transformation",
      xlab = "Theoretical Quantiles",
      ylab = "Sample Quantiles")
# QQ Plot of MedalCount2012 after log transformation
qqnorm(data$log_Medal2012)
qqline(data$log_Medal2012, col = "red")
title(main = "\n\nAfter Log Transformation",
      xlab = "Theoretical Quantiles",
      ylab = "Sample Quantiles")

# Task 2: Repeat task 1 for log-transformed outputs
# Load the required library
######Log transformation
data$log_Medal2012 <- log(data$Medal2012 + 1)
# Fit linear regression model with log-transformed output
lm_model2 <- lm(log_Medal2012 ~ Population + GDP, data = data)
# Print model summary
cat("Model Summary:\n")
print(summary(lm_model2))
# Plot linear regression line with unique style using ggplot2
cat("Linear regression plot:\n")
ggplot(data, aes(x = GDP, y = log_Medal2012)) +
  geom_point(alpha = 0.7, color = "steelblue") +  # Scatter plot
  geom_smooth(method = "lm", se = FALSE, color = "black", linetype = "dashed") +  # Regression line
  labs(title = "Linear Regression: Medal count vs. GDP",
# Output the probability
probability_at_least_one_medal
x = "GDP (in billions of US dollars)",
       y = "Medal count (2012 Olympics)") +
  theme_minimal() +  # Minimalist theme
  theme(plot.title = element_text(hjust = 0.5))  # Centered title
###Model Comparison:
# Create a data frame for observed and predicted values of both models
# Compute residuals and fitted values for both models
fitted_values_model1 <- fitted(lm_model)
residuals_model1 <- residuals(lm_model)
fitted_values_model2 <- fitted(lm_model2)
residuals_model2 <- residuals(lm_model2)
# Create separate residual plots for each model
plot_model1 <- ggplot(data.frame(Fitted_Values = fitted_values_model1,
                                 Residuals = residuals_model1), aes(x = Fitted_Values, y = Residuals)) +
  geom_point(alpha = 0.5) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  labs(title = "Residuals vs. Fitted Values - Model 1",
       x = "Fitted Values",
       y = "Residuals") +

```

```
  theme_minimal()

plot_model2 <- ggplot(data.frame(Fitted_Values =
    fitted_values_model2, Residuals = residuals_model2), aes(x =
    Fitted_Values, y = Residuals)) +
  geom_point(alpha = 0.5) +
  geom_hline(yintercept = 0, linetype = "dashed") +
  labs(title = "Residuals vs. Fitted Values - Model 2",
       x = "Fitted Values",
       y = "Residuals") +
  theme_minimal()
# Display the plots
grid.arrange(plot_model1, plot_model2, ncol = 2)

# Task 3: Develop your own regression model
print("Task 3: Custom regression model")
# Fit polynomial regression model
poly_model <- lm(log_Medal2012 ~ poly(Population,2) +
    poly(GDP, 2), data = data)
# Print model summary
print(summary(poly_model))
# Plot regression line
print("Linear regression plot:")
print(plot(poly_model))
# Task 4: Model selection using AIC
# Set seed for reproducibility
set.seed(123)
# Split data into training (90%) and testing (10%) sets
train_index <- sample(1:nrow(data), 0.9 * nrow(data))
train_data <- data[train_index, ]
test_data <- data[-train_index, ]

# Fit models on training data
lm_model_task1 <- lm(Medal2012 ~ Population + GDP, data =
    train_data)
lm_model_task2 <- lm(log_Medal2012 ~ Population + GDP, data =
    train_data)
lm_model_task3 <- lm(log_Medal2012 ~ poly(Population, 2) +
    poly(GDP, 3), data = train_data)

# Predict on test data
pred_task1 <- predict(lm_model_task1, newdata = test_data)
pred_task2 <- exp(predict(lm_model_task2, newdata = test_data))  #
    Convert log scale back to original scale
pred_task3 <- exp(predict(lm_model_task3, newdata = test_data))  #
    Convert log scale back to original scale

# Calculate RMSE for each model
rmse_task1 <- sqrt(mean((test_data$Medal2012 - pred_task1)^2))
rmse_task2 <- sqrt(mean((exp(test_data$log_Medal2012) -
    pred_task2)^2))
rmse_task3 <- sqrt(mean((exp(test_data$log_Medal2012) -
    pred_task3)^2))

# Print RMSE for each model
cat("RMSE for Model Task 1:", rmse_task1, "\n")
cat("RMSE for Model Task 2:", rmse_task2, "\n")
cat("RMSE for Model Task 3:", rmse_task3, "\n")

# RMSE values for each model
rmse_values <- c(rmse_task1, rmse_task2, rmse_task3)
models <- c("Model Task 1", "Model Task 2", "Model Task 3")

# Calculate adjusted R-squared values for each model
adj_r_squared <- c(summary(lm_model_task1)$adj.r.squared,
                   summary(lm_model_task2)$adj.r.squared,
                   summary(lm_model_task3)$adj.r.squared)

# Calculate AIC scores for each model
aic_values <- c(AIC(lm_model_task1),
                AIC(lm_model_task2),
                AIC(lm_model_task3))

# Plotting the results
par(mfrow = c(1, 3))  # Set up a 1x3 grid for base R plots

# Barplot for Adjusted R-squared
barplot(adj_r_squared, main = "Adjusted R-squared by Model",
    names.arg = models, col = "skyblue",
        ylim = c(0, max(adj_r_squared) + 0.1), ylab = "Adjusted R-
    squared")

# Barplot for AIC Scores
barplot(aic_values, main = "AIC Scores by Model", names.arg =
    models, col = "salmon",
        ylim = c(0, max(aic_values) + 100), ylab = "AIC Score")

# Barplot for RMSE
barplot(rmse_values, main = "Root Mean Square Error (RMSE) by
    Model", names.arg = models, col = "blue",
        ylim = c(0, max(rmse_values) + 1), ylab = "RMSE")

# Now, create a ggplot for AIC comparison
library(ggplot2)
library(gridExtra)
# Create a data frame for ggplot
results_df <- data.frame(Model = models, Adj_R_Squared = adj_r_squared, AIC =
    aic_values, RMSE = rmse_values)
# Create ggplots
gg_adj_r_squared <- ggplot(results_df, aes(x = Model, y = Adj_R_Squared)) +
  geom_bar(stat = "identity", fill = "skyblue", color = "black") +
  labs(title = "Adjusted R-squared by Model", x = "Model", y = "Adjusted R-
    squared") +
  theme_minimal()
gg_aic <- ggplot(results_df, aes(x = Model, y = AIC)) +
  geom_bar(stat = "identity", fill = "salmon", color = "black") +
  labs(title = "AIC Scores by Model", x = "Model", y = "AIC") +
  theme_minimal()
gg_rmse <- ggplot(results_df, aes(x = Model, y = RMSE)) +
  geom_bar(stat = "identity", fill = "blue", color = "black") +
  labs(title = "Root Mean Square Error (RMSE) by Model", x = "Model", y =
    "RMSE") +
  theme_minimal()
# Arrange ggplots in a grid with 1 row and 3 columns
grid.arrange(gg_adj_r_squared, gg_aic, gg_rmse, nrow = 1)
# Function to plot histogram with normal curve using plotly
plot_resid_hist_plotly <- function(model, title, color) {
  # Calculate mean and standard deviation of residuals
  resid_mean <- mean(residuals(model))
  resid_sd <- sd(residuals(model))
  # Create histogram trace
  hist_trace <- plot_ly(x = residuals(model), type = "histogram", histnorm =
    "probability",
                        marker = list(color = color), opacity = 0.7) %>%
   add_trace(x = seq(min(residuals(model)), max(residuals(model)), length.out =
    100),
              y = dnorm(seq(min(residuals(model)), max(residuals(model)), length.out =
    100),
                        mean = resid_mean, sd = resid_sd),
              type = "scatter", mode = "lines", line = list(color = "blue"), name =
    "Normal Curve") %>%
   layout(title = title,
          xaxis = list(title = "Residuals"),
          yaxis = list(title = "Density"),
          showlegend = TRUE)

  return(hist_trace)
}
# Normality of Residuals: Histogram with Normal Curve
par(mfrow = c(1, 3))
hist(residuals(lm_model_task1), probability = TRUE,
     main = "Model 1: Histogram of Residuals",
     xlab = "Residuals", ylab = "Density",
     col = "yellow", border = "black")
curve(dnorm(x, mean = mean(residuals(lm_model_task1)), sd =
    sd(residuals(lm_model_task1))),
      add = TRUE, col = "black", lwd = 2)
# Normality of Residuals: Histogram with Normal Curve
hist(residuals(lm_model_task2), probability = TRUE,
     main = "Model 2: Histogram of Residuals ",
     xlab = "Residuals", ylab = "Density",
     col = "pink", border = "black")
curve(dnorm(x, mean = mean(residuals(lm_model_task2)), sd =
    sd(residuals(lm_model_task2))),
      add = TRUE, col = "black", lwd = 2)
# Normality of Residuals: Histogram with Normal Curve
hist(residuals(lm_model_task3), probability = TRUE,
     main = "Model 3: Histogram of Residuals ",
     xlab = "Residuals", ylab = "Density",
     col = "salmon", border = "black")
curve(dnorm(x, mean = mean(residuals(lm_model_task3)), sd =
    sd(residuals(lm_model_task3))),
      add = TRUE, col = "black", lwd = 2)
################################
# Task 5: Compute the probability that a specific country wins at least one medal
    given the estimated model parameters
cat("Task: Computing the probability that a country wins at least one medal\n")
# Specify the country of interest (e.g., "Great Britain" or any other country)
country_of_interest <- "Great Britain"
# Filter data for the specified country
country_data <- data[data$Country == country_of_interest, ]
# Predict the number of medals for the specified country using the linear regression
    model
predicted_medals <- predict(lm_model, newdata = country_data)
# Calculate the probability of winning at least one medal (assuming Poisson
    distribution)
probability_at_least_one_medal <- ppois(1, lambda = predicted_medals, lower.tail
    = FALSE)
# Output the probability
cat(paste("The probability that", country_of_interest, "wins at least one medal is:",
    probability_at_least_one_medal, "\n")
```