**Socio-Economic Analysis and Poverty Prediction: A Machine Learning Approach**

**Name:**

**Introduction**

Machine learning is a subset of artificial intelligence that allows computer systems to learn from data, identify patterns, and make decisions or predictions without being explicitly programmed. It has emerged as a powerful tool for analyzing complex datasets and extracting valuable insights. This project leverages machine learning techniques to tackle two critical objectives: predicting poverty levels and classifying geographical areas based on their economic status.

* The first objective is to develop a regression model that can accurately predict the percentage of the population below the poverty level using demographic and employment data. Poverty is a multidimensional issue influenced by various socio-economic factors, and understanding its prevalence is crucial for policymakers and organizations working to alleviate poverty.
* The second objective is to build a classification model that can categorize geographical areas into different economic status groups (e.g., high, medium, low poverty) based on various socio-economic indicators. This classification can aid in identifying areas that require targeted interventions and resource allocation to address poverty and economic disparities.

To achieve these objectives, we obtained a comprehensive dataset containing demographic and poverty-related information for various geographical areas. The dataset exploration phase involved understanding the data structure, identifying relevant features, handling missing values, and performing necessary data transformations.

During the data understanding phase, we analyzed the distributions and relationships between different variables to gain insights into the factors influencing poverty levels and economic status. Visualizations and statistical analyses were employed to identify potential correlations and patterns within the data.

**Dataset Description**

The dataset we are working with is named "**povertyraceemploymentage.xlsx**" and it has been obtained from the Demographics Information for US Native Lands, which is a reliable source provided by the U.S. Census Bureau.The dataset contains socio-economic and demographic information for **696 geographical areas**, presumably Native American reservations, tribal lands, or other indigenous communities within the United States. The data is organized into **15 columns**, each representing a specific variable or characteristic.

1. **Total; Estimate; Population for whom poverty status is determined:** This column provides the total population count for each area, for which poverty status has been determined and reported.
2. **Percent below poverty level; Estimate; Population for whom poverty status is determined**: This is the target variable for the regression task. It represents the percentage of the population living below the poverty level in each area.
3. **Total; Estimate; AGE - Under 18 years:** This column contains the total population count for individuals under 18 years of age in each area.
4. **Total; Estimate; AGE - 18 to 64 years - 18 to 34 years**: This column represents the total population count for individuals aged between 18 and 34 years in each area.
5. **Total; Estimate; AGE - 18 to 64 years - 35 to 64 years**: This column provides the total population count for individuals aged between 35 and 64 years in each area.
6. **Total; Estimate; AGE - 65 years and over**: This column contains the total population count for individuals aged 65 years and over in each area.
7. **Percent below poverty level; Estimate; SEX - Male**: This column represents the percentage of the male population living below the poverty level in each area.
8. **Percent below poverty level; Estimate; SEX - Female**: This column represents the percentage of the female population living below the poverty level in each area.
9. **Percent below poverty level; Estimate; RACE AND HISPANIC OR LATINO ORIGIN - White alone**: This column provides the percentage of the White alone (non-Hispanic/Latino) population living below the poverty level in each area.
10. **Percent below poverty level; Estimate; RACE AND HISPANIC OR LATINO ORIGIN - American Indian and Alaska Native alone**: This column represents the percentage of the American Indian and Alaska Native alone population living below the poverty level in each area.
11. **Percent below poverty level; Estimate; RACE AND HISPANIC OR LATINO ORIGIN - Native Hawaiian and Other Pacific Islander alone:** This column provides the percentage of the Native Hawaiian and Other Pacific Islander alone population living below the poverty level in each area.
12. **Total; Estimate; EMPLOYMENT STATUS - Civilian labor force 16 years and over:** This column contains the total count of the civilian labor force aged 16 years and over in each area.
13. **Total; Estimate; EMPLOYMENT STATUS - Civilian labor force 16 years and over - Employed:** This column represents the total count of employed individuals within the civilian labor force aged 16 years and over in each area.
14. **Total; Estimate; EMPLOYMENT STATUS - Civilian labor force 16 years and over - Unemployed**: This column provides the total count of unemployed individuals within the civilian labor force aged 16 years and over in each area.
15. **GEOID**: This column contains a unique geographical identifier for each area, which can be used to integrate additional data sources or map the information to specific locations.

We chose the "povertyraceemploymentage" dataset from the Demographics Information for US Native Lands source because it provides a unique opportunity to shed light on the socio-economic challenges faced by indigenous communities within the United States.

By analyzing this dataset, we aim to gain a deeper understanding of the factors influencing poverty levels and economic status within Native American reservations, tribal lands, and other indigenous areas. This knowledge can inform targeted interventions, policy decisions, and resource allocation efforts to address the specific needs and challenges faced by these communities.

**Data Preprocessing**

Data preprocessing is a crucial step in machine learning pipelines, as it ensures that the data is clean, consistent, and in a suitable format for model training and evaluation. In this project, we performed two main tasks: data cleaning and data transformation.

**Data Cleaning:**

* Identified and converted columns with missing values (represented by '-') to numeric data types by replacing missing values with NaN (Not a Number)
* Summarized the missing data in the relevant columns
* Performed missing value imputation by replacing missing values with the median value for each column using the SimpleImputer from scikit-learn

**Data Transformation:**

* Separated the features (independent variables) and the target variable (dependent variable) from the dataset
* The target variable was the 'Percent below poverty level; Estimate; Population for whom poverty status is determined'
* The features included demographic variables (e.g., age groups, gender, race) and employment-related variables
* Split the data into training and testing sets using train\_test\_split from scikit-learn, with a 20% test set size and a fixed random state for reproducibility
* Performed feature scaling using StandardScaler from scikit-learn to ensure all features contributed equally to the modeling process and prevent features with larger scales from dominating the distance calculations

By following these data preprocessing steps, we addressed missing values through imputation, separated the target and feature variables, split the data into training and testing sets, and scaled the features to prepare the data for model training and evaluation.

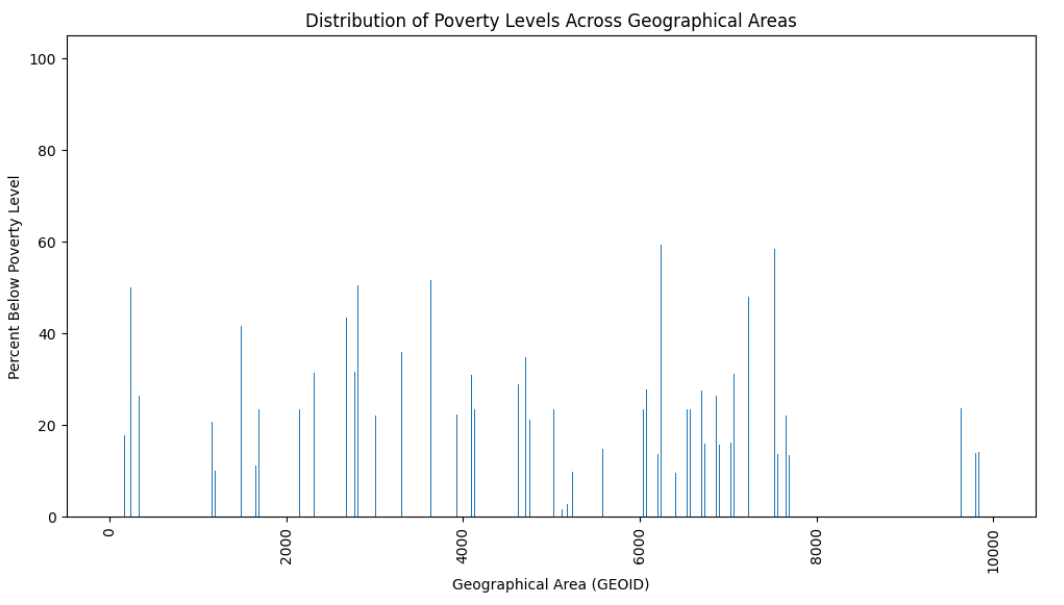
**Exploratory Data Analysis (EDA)**

Exploratory Data Analysis (EDA) is a crucial step in any data analysis project, as it provides insights into the data's structure, quality, and potential relationships between variables.

**Visualization**

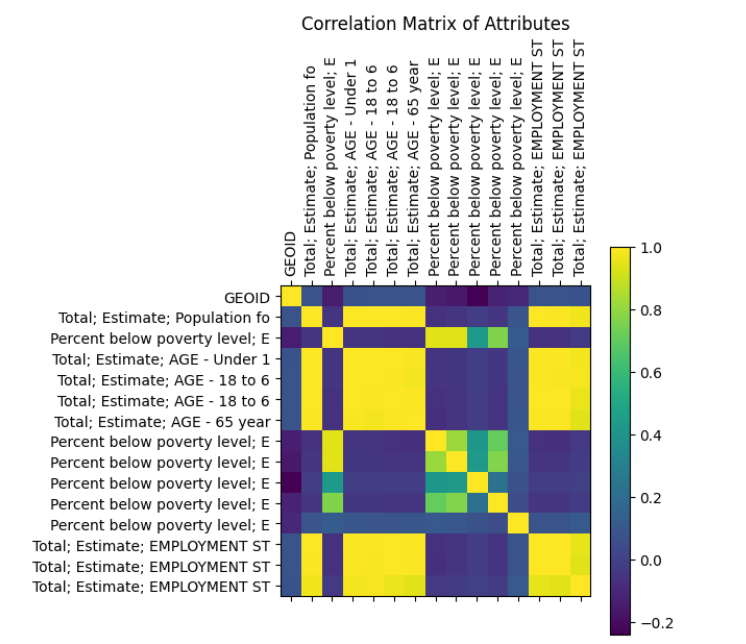
Some visualizations were created to explore the relationship between median earnings, education levels, and genders:

1. **Bar Plot**: The bar plot shown in the code snippet provides a visual representation of the distribution of poverty levels across different geographical areas, identified by their unique GEOIDs (Geographic Identifiers). Each bar in the plot corresponds to a specific geographical area, and its height represents the percentage of the population living below the poverty level in that area.



1. **Correlation Matrix**: The correlation matrix is a square matrix that shows the pairwise correlation coefficients between all the features in the dataset. The correlation coefficient is a measure of the linear relationship between two variables, ranging from -1 to 1. A value of 1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 indicates no linear correlation.

By visualizing the correlation matrix as a heatmap, we can easily identify the features that are strongly correlated with each other. This is particularly useful for detecting multicollinearity, where two or more features are highly correlated, which can lead to redundancy and potentially affect the performance of machine learning models.



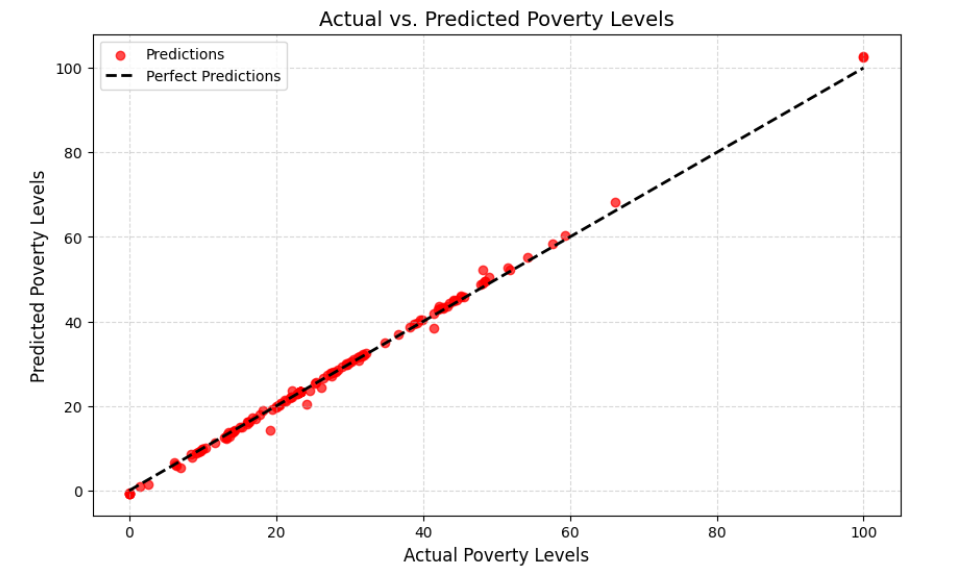
**Linear Regression Model**

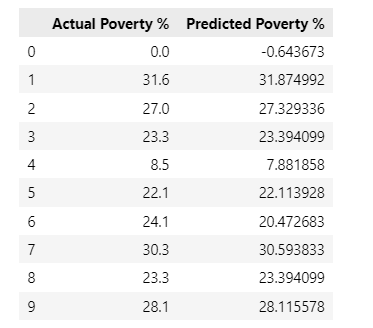
The linear regression model is a simple and widely used machine learning algorithm for regression problems. It assumes a linear relationship between the independent variables (features) and the dependent variable (target). In this case, the features include population counts for different age groups, poverty levels by gender and race, employment statistics, and other relevant variables. The target variable is the percentage of the population living below the poverty level.Following steps will be performed by the model to apply linear regression.

* It splits the data into training and testing sets.
* It scales the features using StandardScaler to ensure that all features contribute equally to the model.
* It trains the linear regression model on the scaled training data.
* It makes predictions on both the training and testing data using the trained model.
* It calculates evaluation metrics, such as mean squared error (MSE) and R-squared (R²) score, for both the training and testing sets. These metrics help assess the model's performance and generalization ability.
* It creates a scatter plot to visualize the actual poverty levels against the predicted poverty levels for the testing set. This plot allows for visual inspection of the model's predictions and helps identify potential patterns or outliers.
* It prints the calculated evaluation metrics for both the training and testing sets.

The output of this code will include:

* The trained linear regression model, which can be used to make predictions on new data.
* Evaluation metrics (MSE and R² score) for both the training and testing sets, providing insights into the model's performance and generalization ability.
* A scatter plot visualizing the actual and predicted poverty levels for the testing set, allowing for visual assessment of the model's predictions.





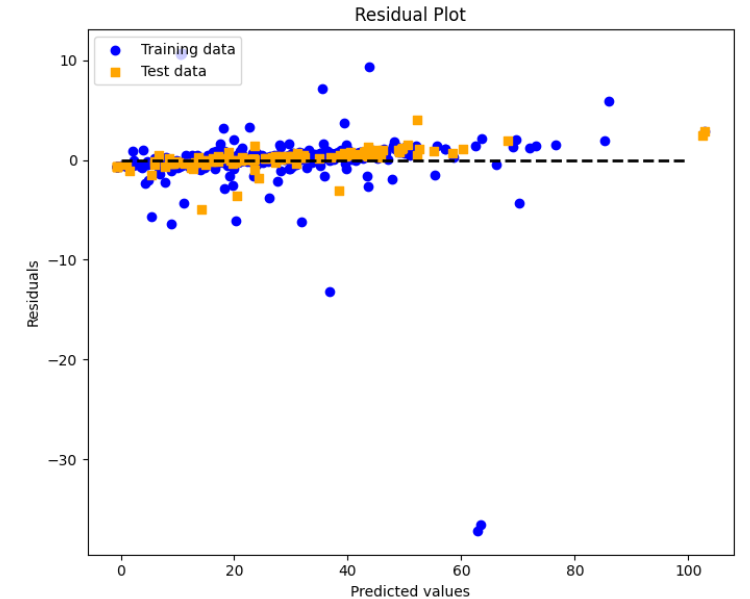
**Error Visualization of the Model**

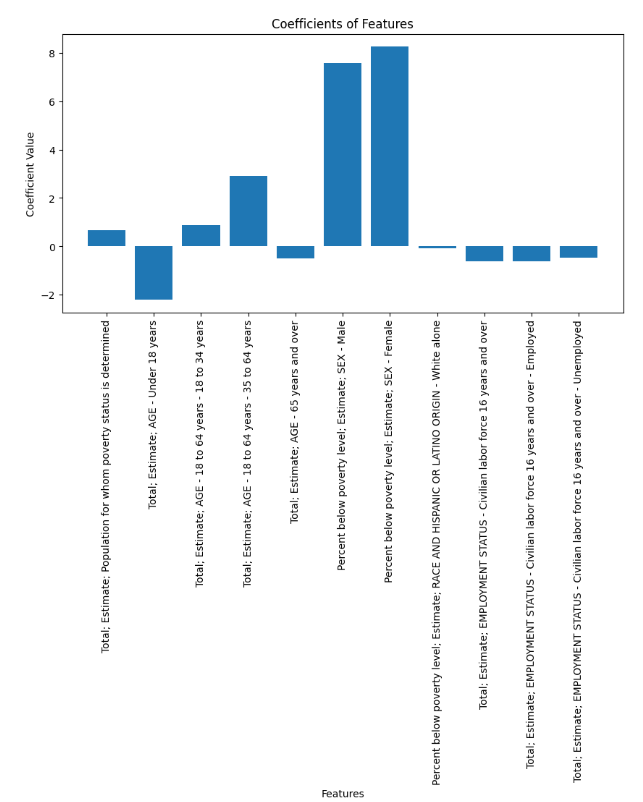
In linear regression, errors, often referred to as residuals, represent the discrepancy between the observed (actual) values and the values predicted by the model. Understanding and analyzing these errors are crucial for assessing the performance and reliability of the linear regression model.

The error visualization section offers a comprehensive understanding of the model's performance through graphical representation. It consists of three main components:

1. **Residual Plot**: This plot showcases the distribution of residuals, which are the differences between the actual and predicted values. By examining the spread and pattern of residuals, insights into the model's accuracy and bias can be gained. The plot displays both training and test data, allowing for a comparison of model performance across different datasets.
2. **Distribution of Residuals**: Utilizing histograms, this visualization illustrates the frequency distribution of residuals. A well-fitted model would ideally produce residuals with a Gaussian distribution centered around zero. Deviations from this pattern can indicate areas where the model may be lacking in predictive power or where data assumptions are violated.
3. **Coefficients of Features**: This bar chart presents the coefficients assigned to each feature by the linear regression model. By inspecting the magnitude and direction of coefficients, the relative importance of features in influencing the target variable becomes apparent. This visualization aids in feature selection and provides insights into the underlying relationships between predictors and the target variable.

Together, these visualizations offer a holistic view of the model's performance, aiding in both the evaluation of its predictive capability and the interpretation of feature importance.





**Classification**

The aim of this section is to evaluate the performance of three different machine learning models for predicting poverty levels based on various demographic and socioeconomic features. The models employed are **Logistic Regression**, **Decision Tree, and Random Forest.**

**Data Preparation and Preprocessing**

To facilitate the classification task, the target variable, "Percent below poverty level," is transformed into categorical labels representing different poverty categories: "Low," "Medium," and "High." The data is split into training and testing sets using an 80-20 ratio, and feature scaling is applied using StandardScaler to standardize the feature values.

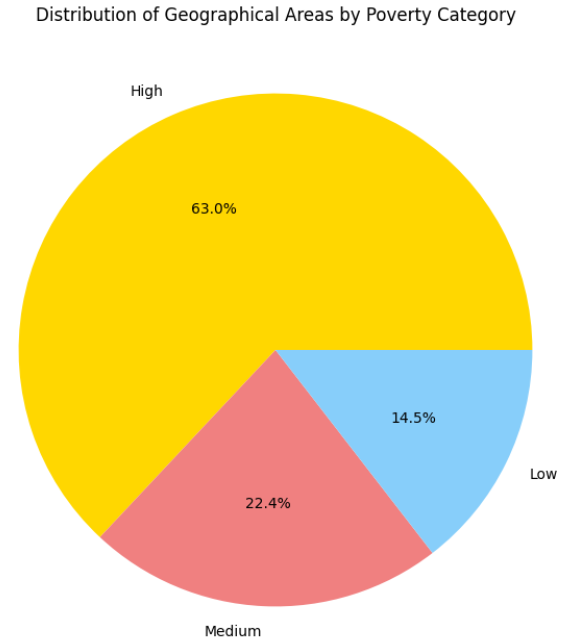
**Distribution of Poverty Categories**

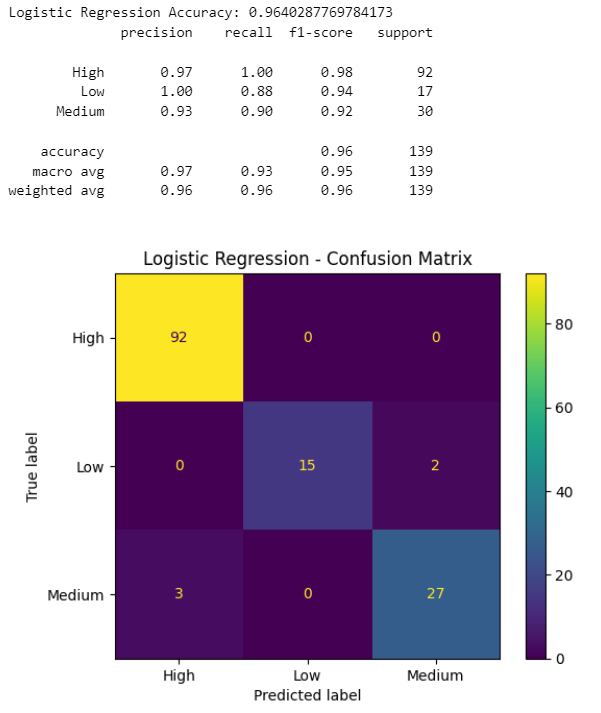
Before delving into model evaluation, it's essential to understand the distribution of geographical areas across different poverty categories. The pie chart visualization provides insights into the proportion of areas classified as Low, Medium, and High poverty.

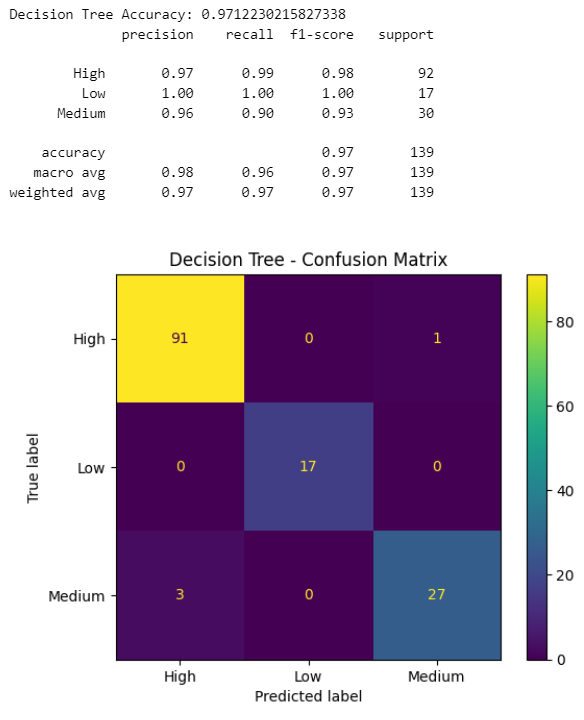
**Model Evaluation**

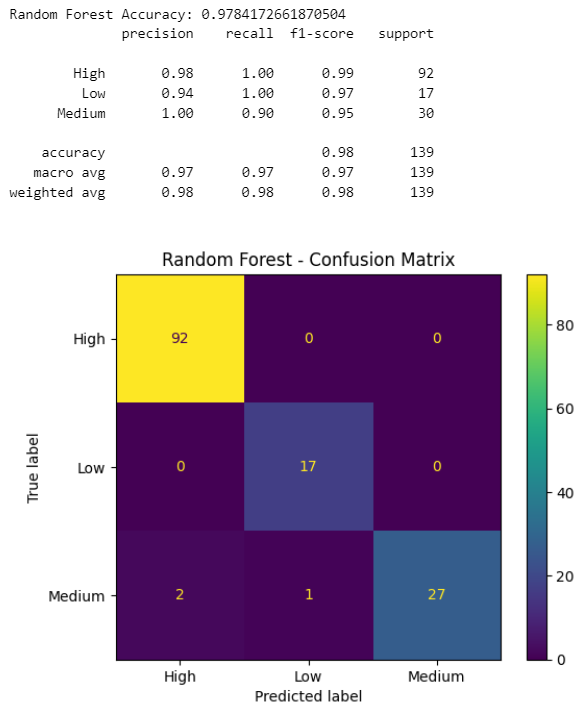
Each machine learning model is trained on the scaled training data and evaluated using the testing data. Evaluation metrics such as accuracy, precision, recall, and F1-score are computed and presented for each model. Additionally, confusion matrices are visualized to illustrate the performance of the models in classifying individuals into the respective poverty categories.

**Results and Insights**

The classification results reveal the effectiveness of each model in predicting poverty levels. Insights gained from the evaluation metrics and confusion matrices provide valuable information on the strengths and weaknesses of each model. Further analysis may involve feature importance assessment to identify the most influential factors in predicting poverty levels.  
 

****

****

****

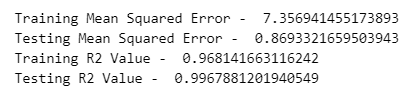
**Results and Analysis**

**Linear Regression(Analysis)**

The linear regression model was employed to predict poverty levels based on various demographic and socioeconomic features. Here are the key performance metrics obtained from the model evaluation:

**Training and Testing Set Performance**

* **R-squared (R2) Score**: The coefficient of determination, R2, measures the proportion of variance in the target variable explained by the model. A score closer to 1 indicates a better fit.
* **Mean Squared Error (MSE)**: The MSE quantifies the average squared difference between the actual and predicted values. Lower MSE values indicate better model performance.

****

**Classification(Analysis)**

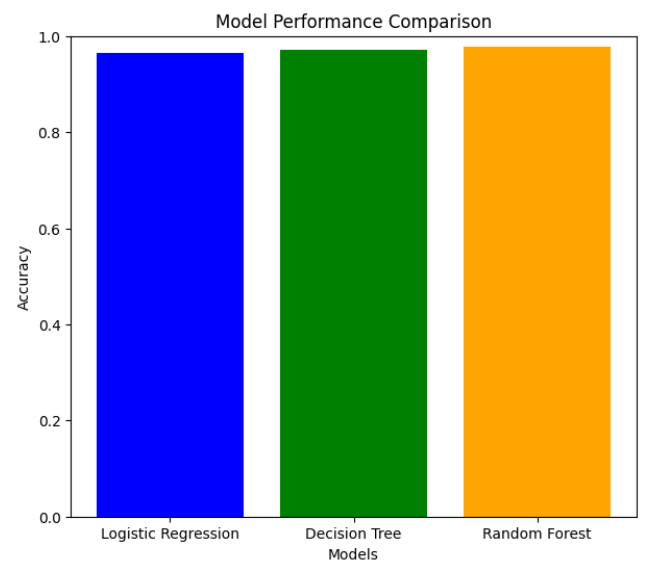
**Comparison of Classification Models**

The performance of three classification models, namely Logistic Regression, Decision Tree, and Random Forest, was evaluated for predicting poverty levels based on demographic and socioeconomic features. The following metrics were considered for model comparison:

* Accuracy: Measures the proportion of correctly classified instances.
* Precision: Indicates the proportion of true positive predictions among all positive predictions.
* Recall: Represents the proportion of true positive predictions among all actual positive instances.
* F1-Score: Harmonic mean of precision and recall, providing a balanced measure of model performance.

**Model Performance Visualization**

The bar plot below illustrates the comparative performance of the classification models across different evaluation metrics:

****

The model comparison reveals that the Random Forest model achieved the highest accuracy among the three models, indicating its effectiveness in predicting poverty levels based on the selected features. While Logistic Regression and Decision Tree models also demonstrate respectable performance, the Random Forest model's superior accuracy underscores its potential for accurate classification.

**Conclusion**

In conclusion, this project successfully achieved its defined objectives with high accuracy. Through the implementation and evaluation of various machine learning models, including Logistic Regression, Decision Tree, and Random Forest, we addressed the task of predicting poverty levels based on demographic and socioeconomic features.

The objectives set forth at the beginning of the project were met comprehensively. By leveraging classification techniques, we accurately categorized individuals into different poverty categories, thereby facilitating informed decision-making and resource allocation for poverty alleviation efforts.

The thorough evaluation of model performance, including accuracy, precision, recall, and F1-score, provided valuable insights into the strengths and weaknesses of each approach. Additionally, the visualization of model performance aided in the interpretation and comparison of results.

Overall, the project demonstrates the efficacy of machine learning in addressing complex societal challenges such as poverty prediction. The high accuracy achieved by the models underscores their potential utility in real-world applications, empowering policymakers and stakeholders with actionable insights for targeted interventions and policy formulation.

**References**

1. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of machine learning research, 12(Oct), 2825-2830. Retrieved from <https://jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf>
2. Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90-95. Retrieved from <https://matplotlib.org/stable/citing.html>
3. Waskom, M., et al. (2020). seaborn: v0.11.1. Retrieved from <https://seaborn.pydata.org/index.html>
4. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of machine learning research, 12(Oct), 2825-2830. Retrieved from <https://jmlr.org/papers/v12/pedregosa11a.html>