**Detailed Report on Titanic Dataset Analysis and Classification Model**

**Introduction**

This report describes a comprehensive data processing and machine learning pipeline applied to the Titanic dataset to predict passenger survival. This pipeline includes data cleaning, outlier handling, normalization, feature engineering, encoding, model training, and evaluation using a Random Forest classifier. The goal of this analysis is to improve the prediction accuracy of survival on the Titanic based on several features.

**Step 1: Data Cleaning (clean\_data Function)**

The clean\_data function is crucial for ensuring the dataset is clean and consistent. This step includes:

1. **Conversion of Non-numeric Data**:
   * The dataset contains numeric columns that may have non-numeric values due to errors or outliers.



Figure 11.Conversion of Non-numeric Data

* + Here, pd.to\_numeric(..., errors='coerce') converts non-numeric values to NaN for columns Age, Fare, Parch, and SibSp.

1. **Handling Missing Values**:
   * Missing values in the dataset are imputed based on the column’s type and distribution.

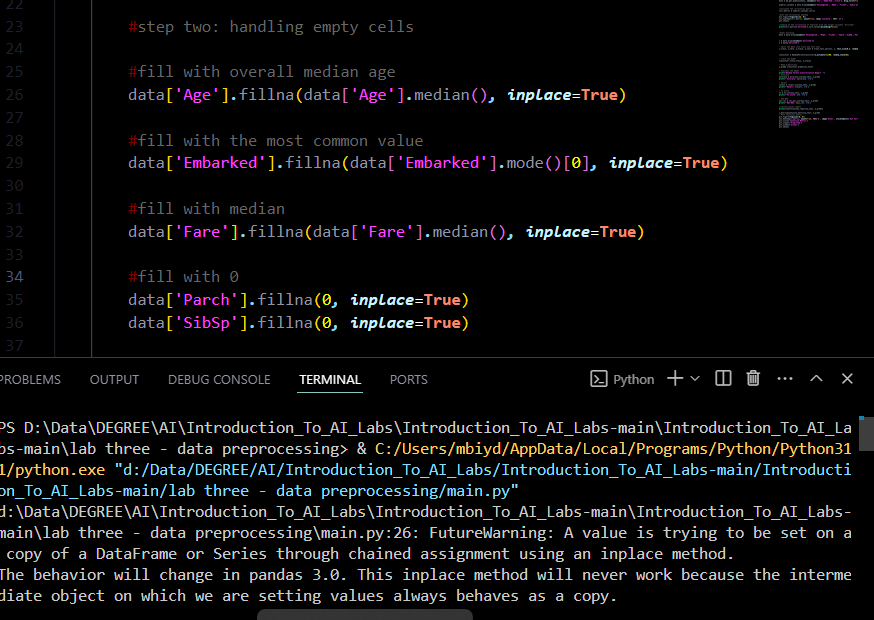


Figure 2 2.Handling Missing Values

* + Missing Age values are filled with the median, while Embarked is filled with the mode. The Fare column is also filled with its median, and missing values in Parch and SibSp are set to 0, assuming that no value means no parents, children, siblings, or spouses.

**Step 2: Handling Outliers**

Outliers are handled using **box plots** and **capping** methods for Fare and Age.

1. **Visualizing Outliers**:
   * Box plots for Fare and Age help visualize the spread and identify outliers.

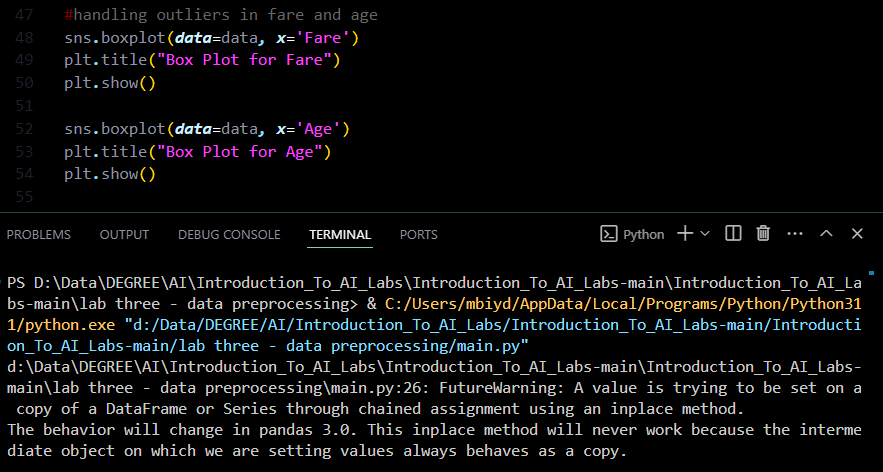
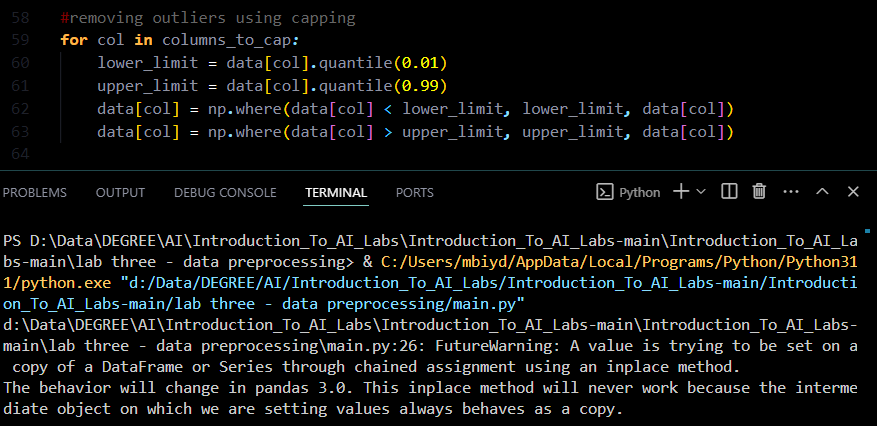


Figure 3 Handling Outliers

1. **Outlier Removal Using Capping**:
   * Outliers are capped to the 1st and 99th percentiles to avoid extreme values that may skew model performance.



**Step 3: Data Normalization**

The Fare and Age columns are normalized using **Z-score normalization** to ensure values have a mean of 0 and a standard deviation of 1. Normalization helps standardize different scales, benefiting model training.

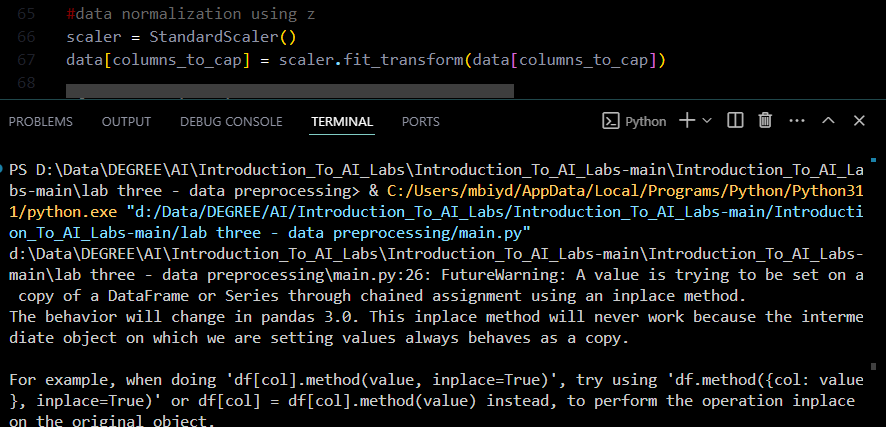


Figure 4 Data Normalization

**Step 4: Feature Engineering**

Feature engineering enhances model performance by creating new variables that capture important information:

1. **Family Size**:
   * Calculated as the sum of SibSp and Parch, representing the number of family members aboard.

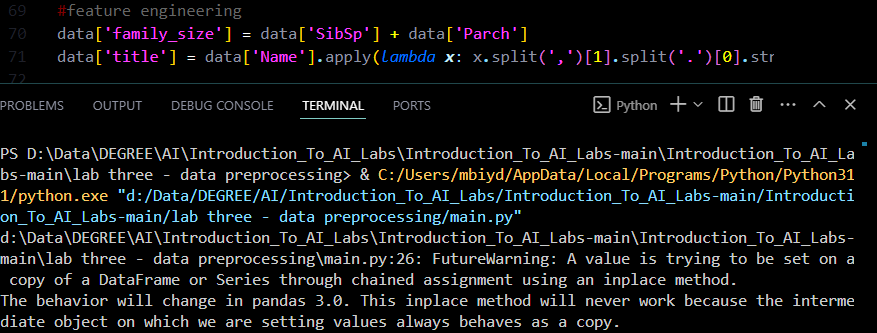


Figure 5 Feature Engineering

1. **Title Extraction**:
   * Titles (Mr., Mrs., etc.) are extracted from the Name column. Titles often correlate with age, gender, and social class, which are relevant to survival.

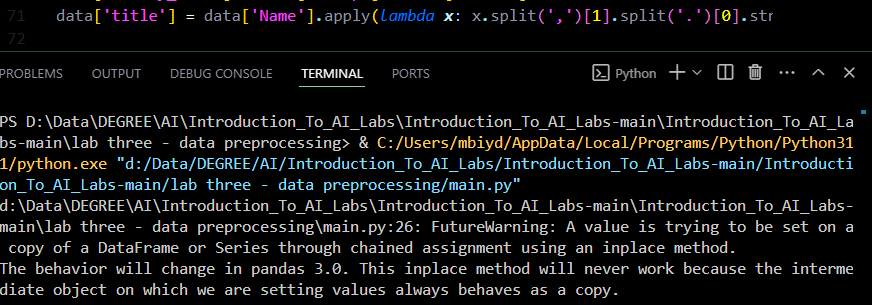


Figure 6 Title Extraction

**Step 5: Encoding Categorical Variables**

Categorical variables (Sex, Embarked, and title) are converted to binary columns using one-hot encoding.

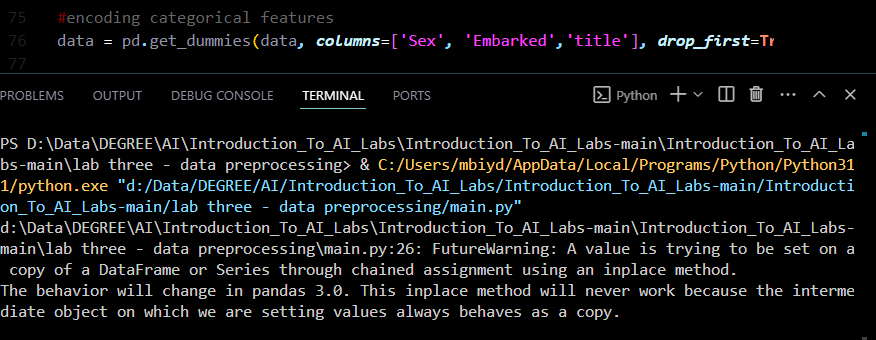


Figure 7 Encoding Categorical Variables

Using drop\_first=True prevents multicollinearity by dropping one category from each categorical variable.

**Step 6: Feature Selection Using Correlation Analysis**

To understand which features correlate with the target variable (Survived), correlation analysis is performed:

1. **Correlation Matrix**:
   * A correlation matrix is calculated to assess linear relationships between features.

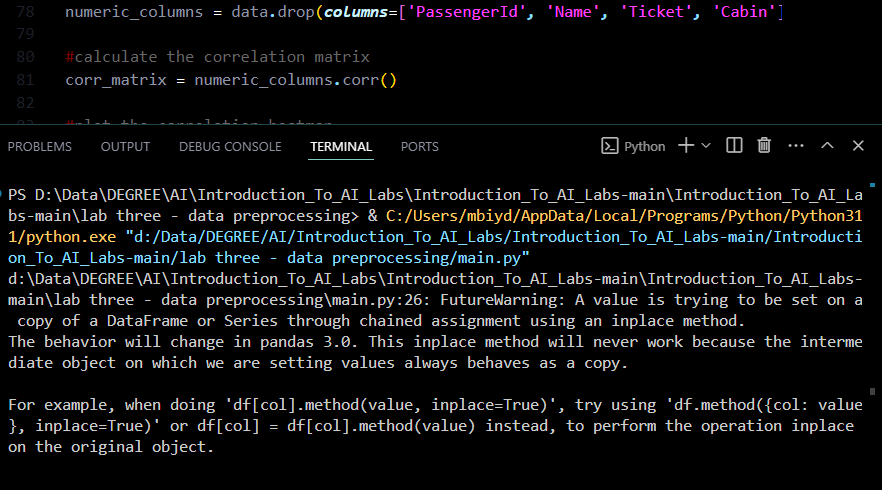


Figure 8 Correlation Matrix:

1. **Correlation Heatmap**:
   * The heatmap visually represents these correlations. Features with higher correlation values are more predictive of survival.

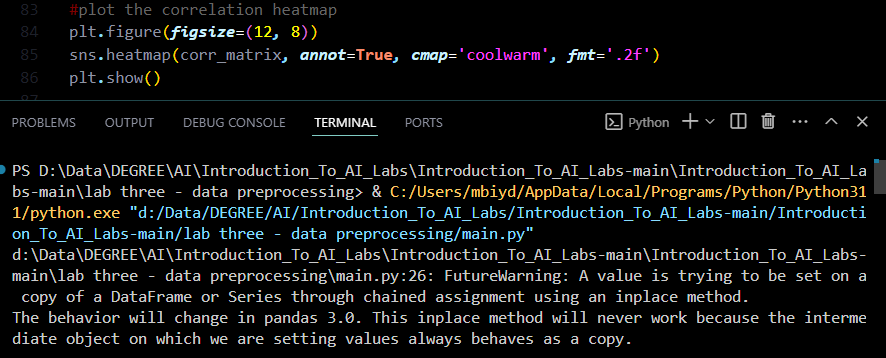


Figure 9 Correlation Heatmap

1. **High-correlation Features**:
   * Features with the strongest correlation to Survived are displayed to help identify the most significant predictors.

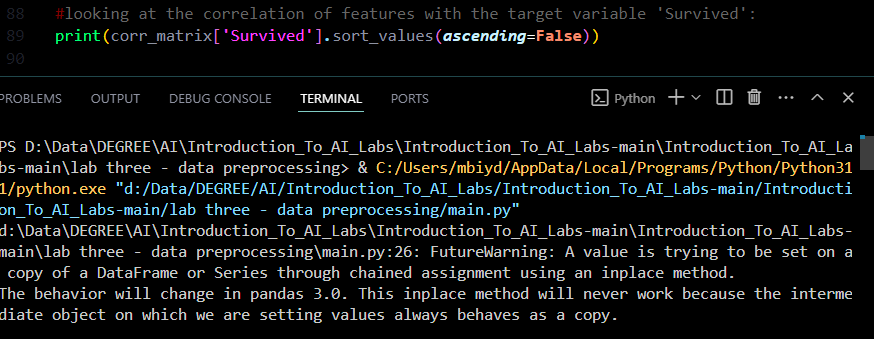


Figure 10 High-correlation Features

**Step 7: Model Building with Random Forest Classifier**

With the data prepared, a Random Forest model is built for survival prediction.

1. **Data Splitting**:
   * The dataset is split into training and testing sets (80% train, 20% test).

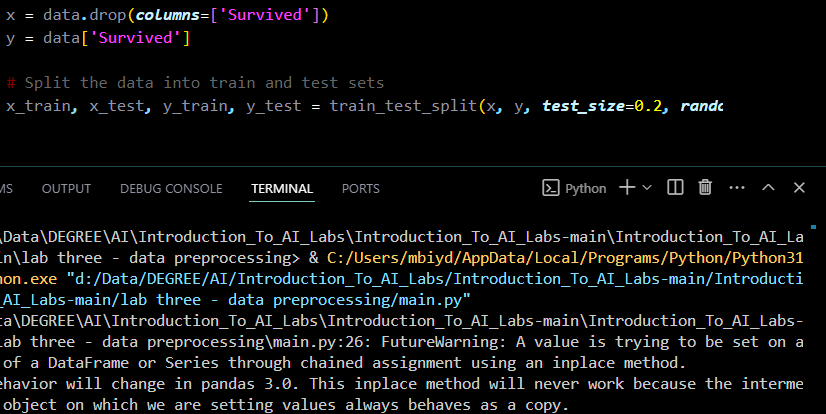


Figure 11 Data Splitting

1. **Random Forest Classifier**:
   * A Random Forest with 100 estimators is created. Random Forests work well with a combination of categorical and numeric features and offer robust performance on tabular data.

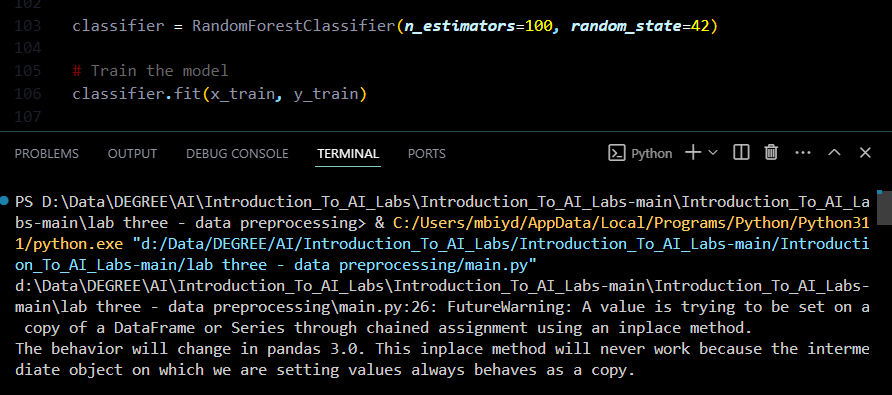


Figure 12 Random Forest Classifier

1. **Predictions**:
   * Predictions are made on the test set.



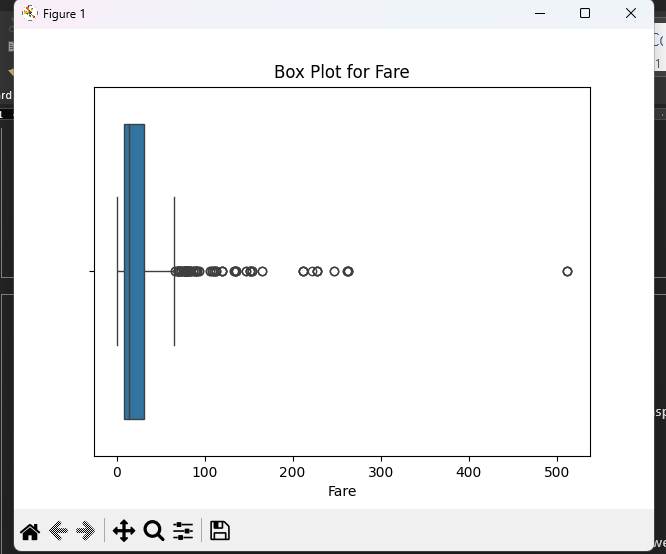
Figure 13 Predictions

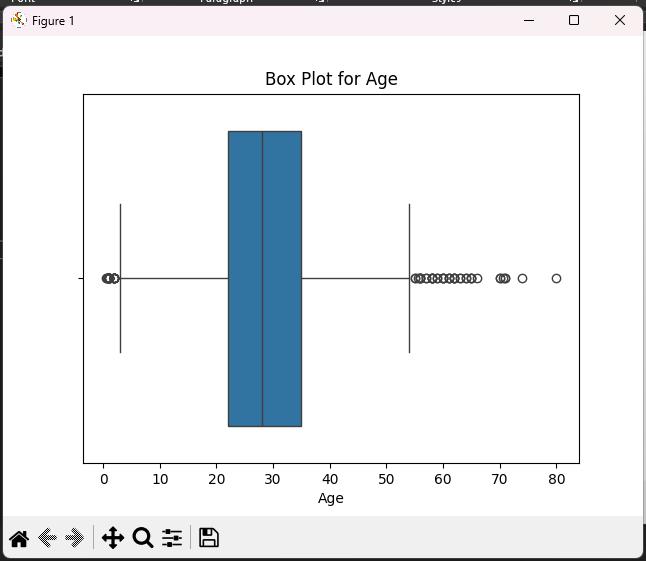
**Step 8: Model Evaluation**

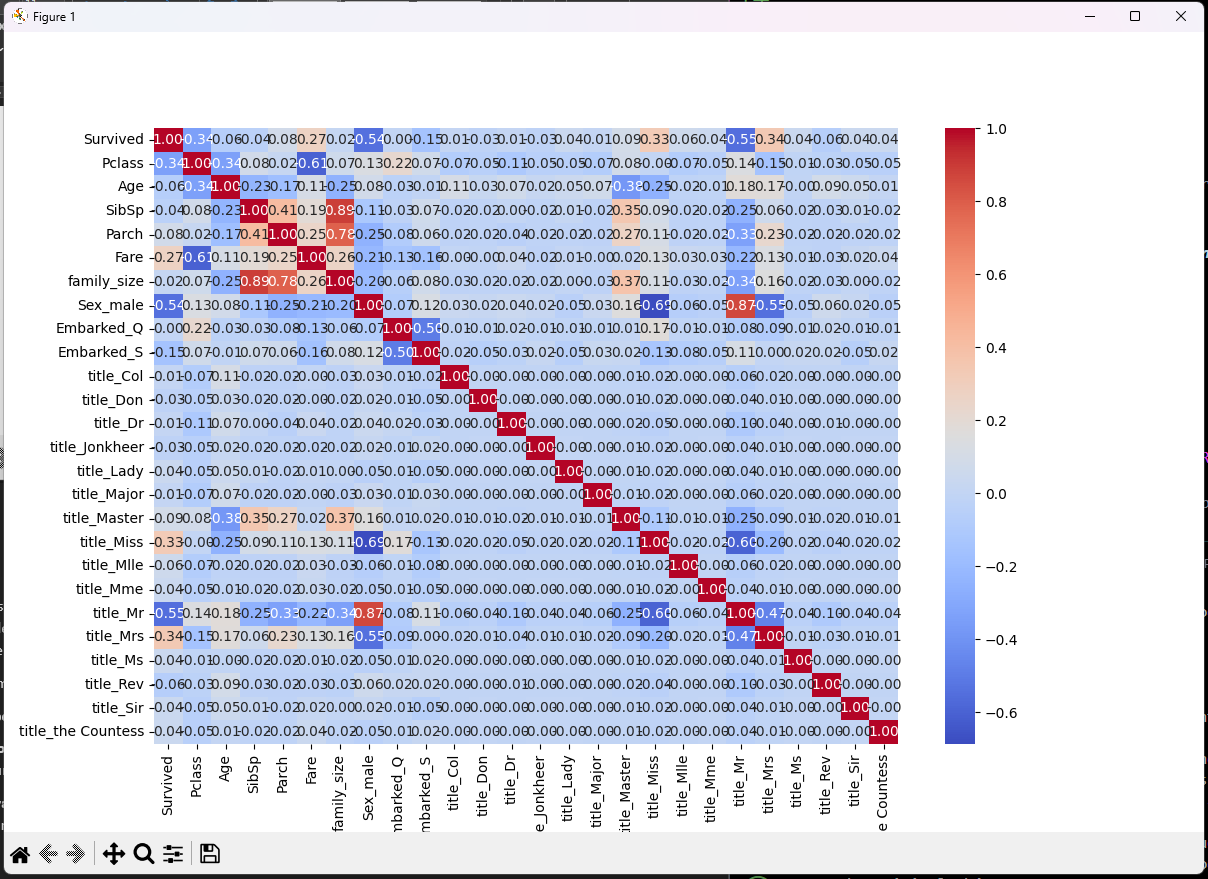
Model performance is evaluated using several metrics: accuracy, recall, F1 score, AUC-ROC, and a classification report.

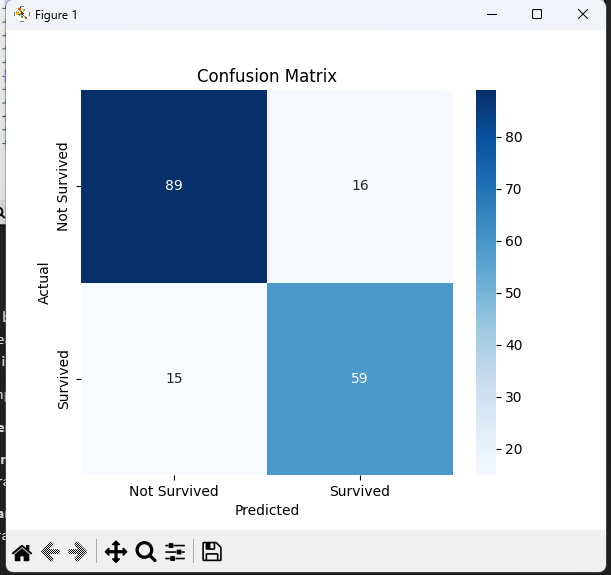
1. **Accuracy, Recall, F1 Score, and AUC-ROC**:
   * These metrics provide insights into model performance in various aspects.
2. **Classification Report**:
   * Displays precision, recall, F1 score, and support for each class.
3. **Confusion Matrix**:
   * The confusion matrix reveals how well the model distinguishes between survivors and non-survivors.

**Results**

****

****

****

****

**Conclusion**

This analysis builds a well-optimized Random Forest model to predict survival on the Titanic. Each stage, from data cleaning to model evaluation, was handled systematically to maximize model performance. The pipeline is adaptable to other structured datasets where binary classification is needed.

For future improvements:

1. **Hyperparameter Tuning** can be performed to optimize Random Forest parameters.
2. **Algorithm Comparison** with other classifiers such as Logistic Regression, Gradient Boosting, and Neural Networks.
3. **Advanced Feature Engineering** could explore more sophisticated transformations or interactions among existing features.