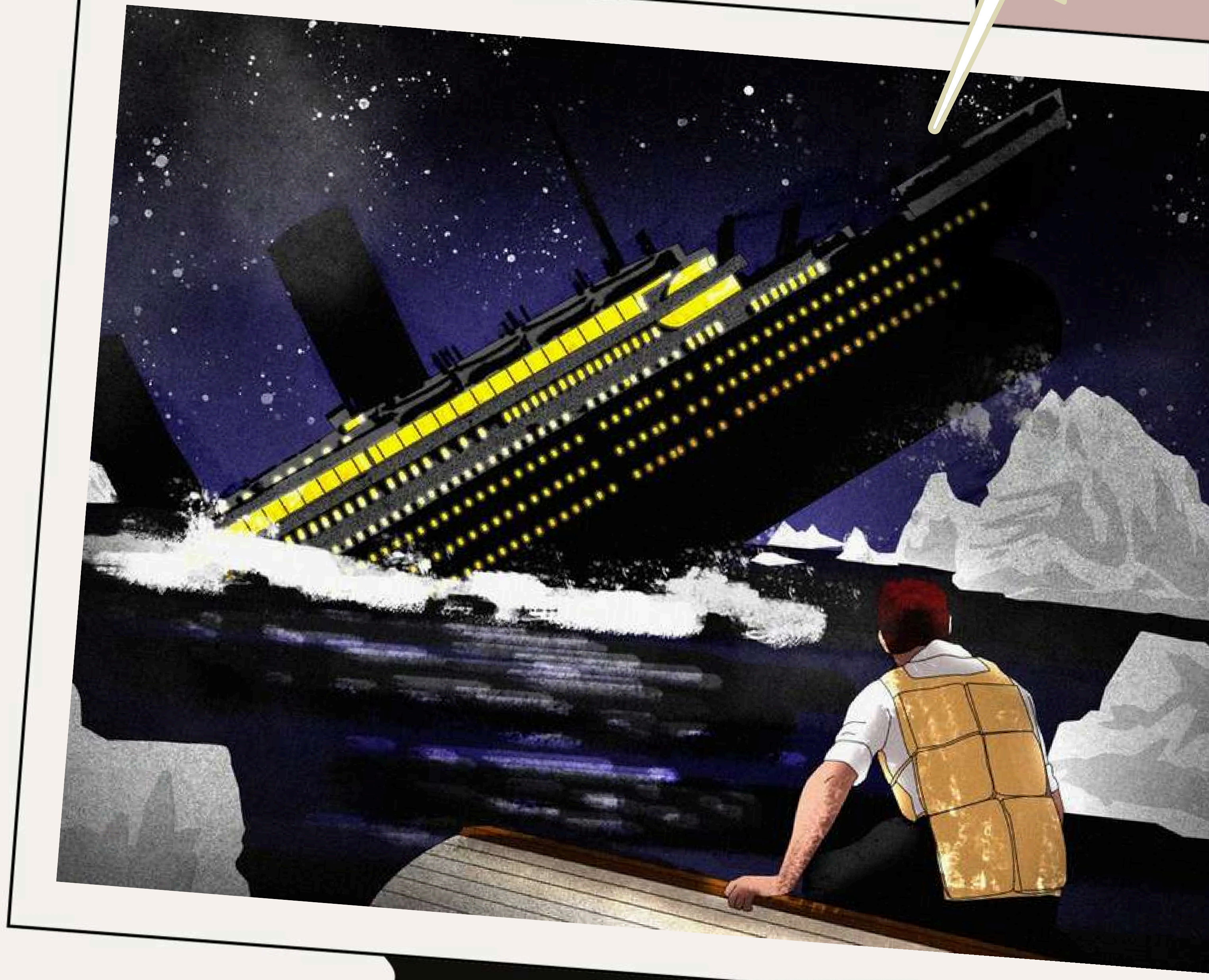
INDIAN INSTITUTE OF TECHNOLOGY PATNA

Titanic Survival Predictior

DATA SCIENCE PROJECT-3

NAME-Varun Gupta
ROLL NO.- 2312res727



Problem Understanding & Project Objective

ProblemStatement: ***The Titanic disaster of 1912 led to over 1,500 deaths, with survival chances influenced by factors like age, gender, and passenger class. This project aims to analyze these factors and predict survival outcomes using machine learning.***

OBJECTIVES

The main objective of this project is to develop a machine learning model that can accurately predict whether a passenger would survive the Titanic disaster, based solely on their personal and ticket details.

GOALS

- Prepare and clean Titanic passenger data.
- Explore key factors affecting survival.
- Train a machine learning model to predict survival.
- Evaluate model performance.

Key Components of Titanic Survival Predictior

Dataset Overview

The project uses the Titanic passenger dataset, which includes details like age, gender, ticket class, fare, family members, and survival status. This dataset provides a real-world example to analyze and predict passenger outcomes using various features.

Model Selection

For this project, a Random Forest Classifier was selected because it effectively manages tabular data, handles missing values well, and provides good accuracy for classification problems like survival prediction.

Training Process

The cleaned data was split into training and validation sets. The model was then fitted on the training data, and performance was assessed using accuracy and other metrics on the validation set to ensure reliable predictions.

Introduction to Titanic Survival Predictior

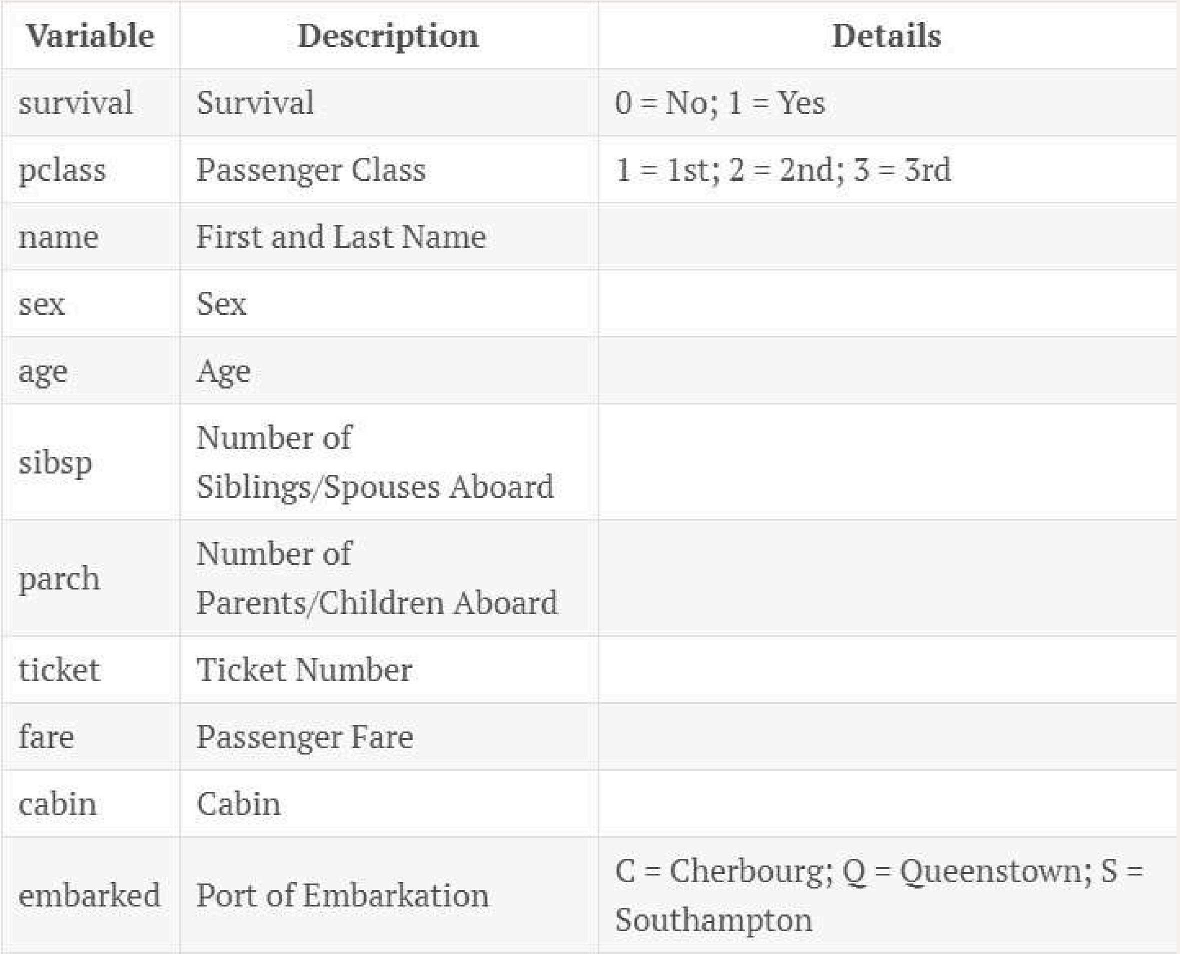
The Titanic Survival Predictor is a machine learning project designed to analyze passenger data from the historic Titanic voyage and predict whether a passenger would have survived the disaster. By processing real-world data and applying classification techniques, the project demonstrates how data science can uncover key survival factors and make accurate predictions.

Understanding how machines learn to recognize patterns and predict outcomes from data.

Dataset Overview

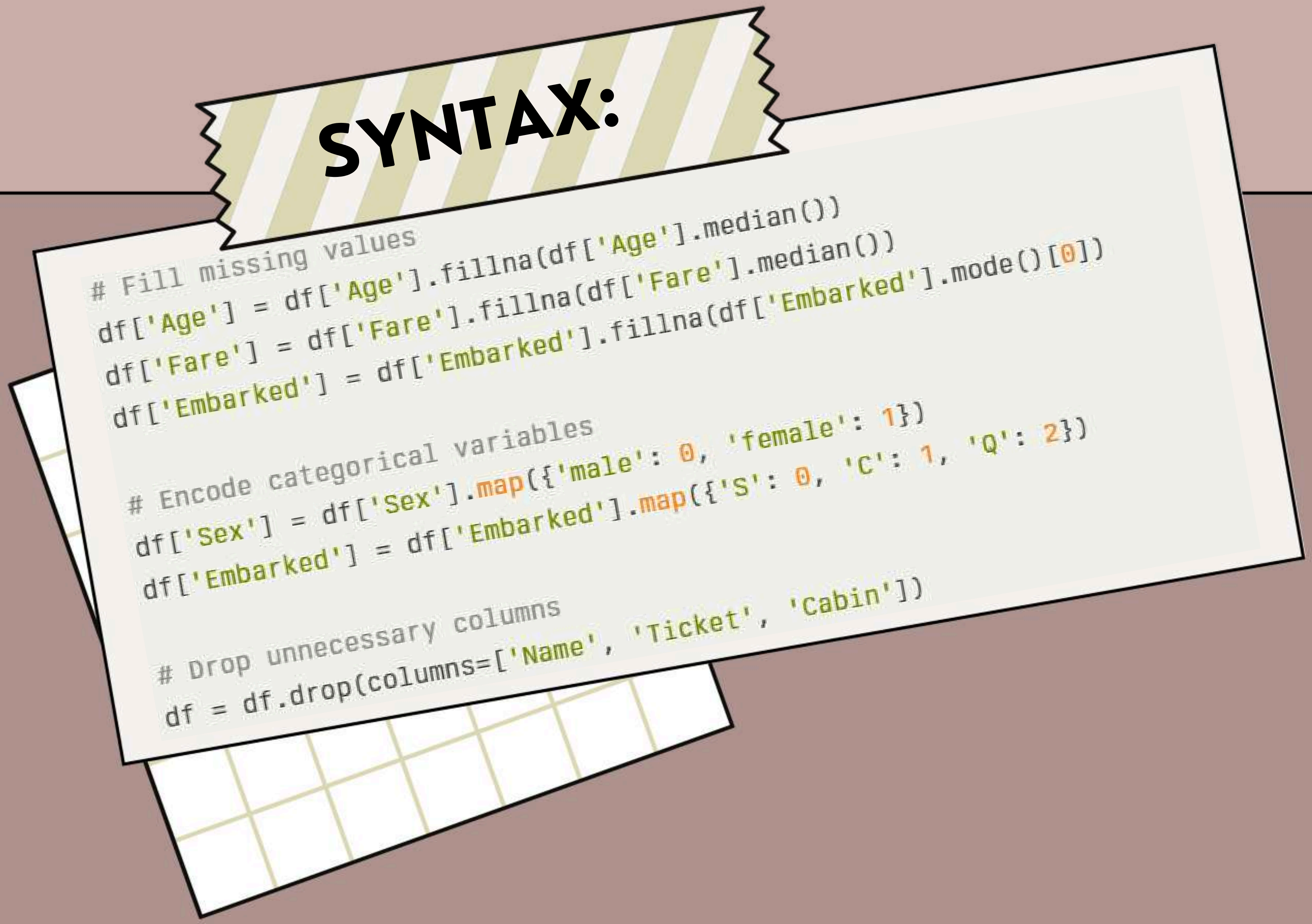
- The Titanic dataset provides passenger-level data from the 1912 shipwreck.
- Each entry represents one passenger on the Titanic.
- The main target variable is “Survived,” indicating if the passenger lived (1) or did not (0).
- The dataset mixes both numerical and categorical variables.
- Contains missing values in some fields, which requires data cleaning.
- Widely used for classification tasks and as a benchmark in beginner machine learning projects.
- Offers a practical example to study the impact of different passenger characteristics on survival rates.

DATASET



Variable	Description	Details
survival	Survival	0 = No; 1 = Yes
pclass	Passenger Class	1 = 1st; 2 = 2nd; 3 = 3rd
name	First and Last Name	
sex	Sex	
age	Age	
sibsp	Number of Siblings/Spouses Aboard	
parch	Number of Parents/Children Aboard	
ticket	Ticket Number	
fare	Passenger Fare	
cabin	Cabin	
embarked	Port of Embarkation	C = Cherbourg; Q = Queenstown; S = Southampton

Data Preprocessing



SYNTAX:

```
# Fill missing values
df['Age'] = df['Age'].fillna(df['Age'].median())
df['Fare'] = df['Fare'].fillna(df['Fare'].median())
df['Embarked'] = df['Embarked'].fillna(df['Embarked'].mode()[0])

# Encode categorical variables
df['Sex'] = df['Sex'].map({'male': 0, 'female': 1})
df['Embarked'] = df['Embarked'].map({'S': 0, 'C': 1, 'Q': 2})

# Drop unnecessary columns
df = df.drop(columns=['Name', 'Ticket', 'Cabin'])
```

- **Handled missing values:**

Missing values in the Age and Fare columns were filled with median values, while missing Embarked entries were filled with the most frequent port.

- **Encoded categorical features:**

Categorical features like Sex and Embarked were converted into numerical codes to make them suitable for machine learning algorithms.

- **Normalized data formats:**

Ensured all numerical columns are in proper type for modeling.

Exploratory Data Analysis (EDA)

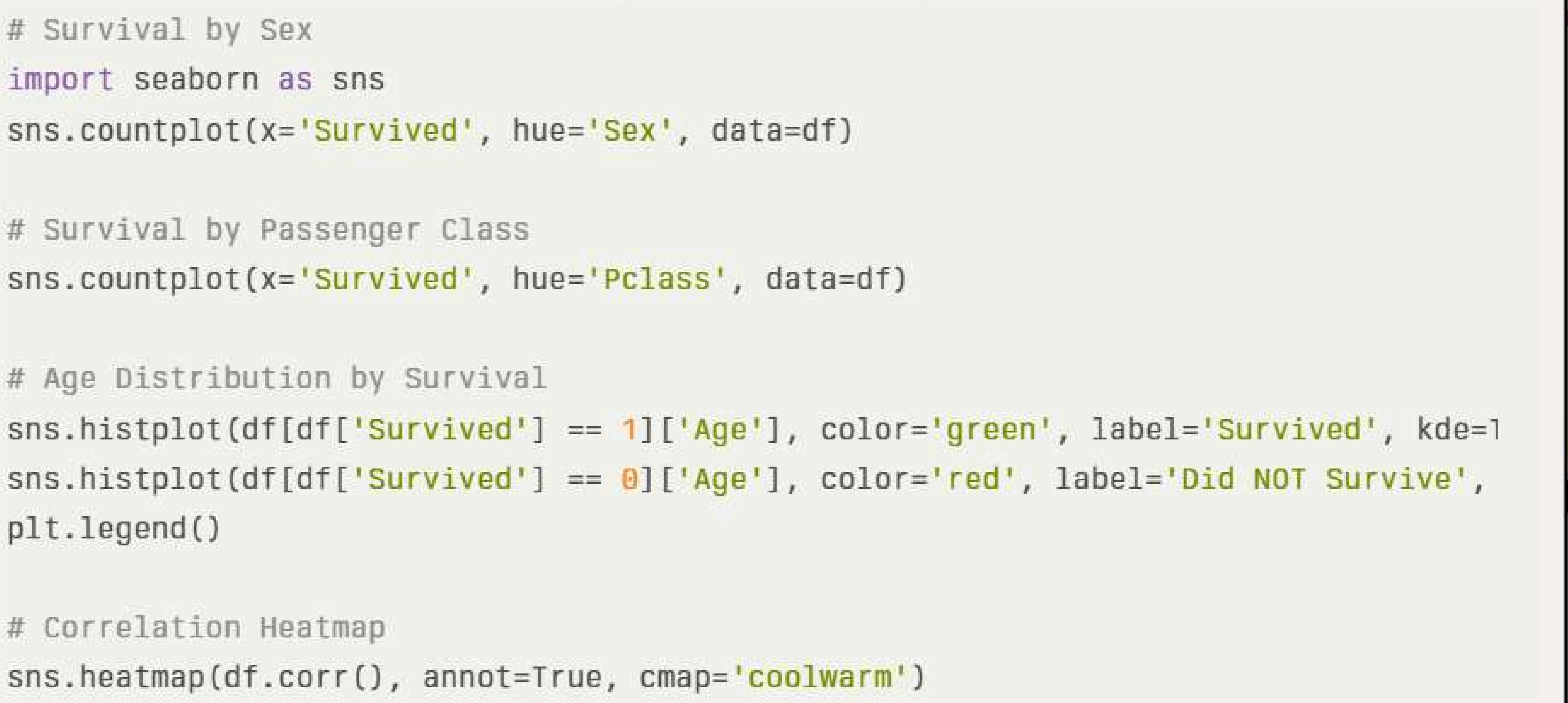
EDA was performed to uncover trends and relationships between features and survival rates.

Key Findings:

- Survival chances were higher for females and 1st class passengers.
- Younger passengers had a slightly better chance of survival.
- The class of the ticket (Pclass) and fare paid showed strong influence on survival.

Visualization:

- Count plots for survival by gender and passenger class.
- Age distribution histograms for survived vs. non-survived.
- Correlation heatmap to identify important feature relationships.



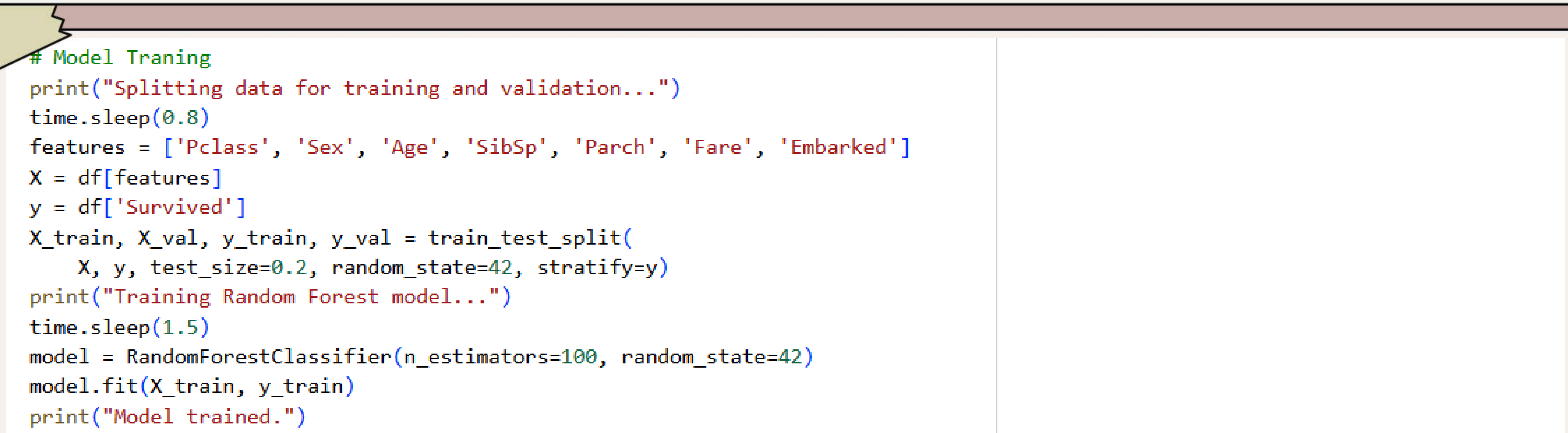
```
# Survival by Sex
import seaborn as sns
sns.countplot(x='Survived', hue='Sex', data=df)

# Survival by Passenger Class
sns.countplot(x='Survived', hue='Pclass', data=df)

# Age Distribution by Survival
sns.histplot(df[df['Survived'] == 1]['Age'], color='green', label='Survived', kde=
sns.histplot(df[df['Survived'] == 0]['Age'], color='red', label='Did NOT Survive',
plt.legend()

# Correlation Heatmap
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
```

EDA SYNTAX

Training the Model

- Chose a Random Forest Classifier for its reliability in handling classification problems and mixed data types.
- Selected important features like Pclass, Sex, Age, SibSp, Parch, Fare, and Embarked to train the model.
- Divided the dataset into training and validation sets (typically 80% training, 20% validation) to accurately assess performance.
- Trained the model on the training data so it could learn the patterns linked to passenger survival.
- Readied the model for evaluation, using the unseen validation set to test how well it predicts survival outcomes.

SYNTAX

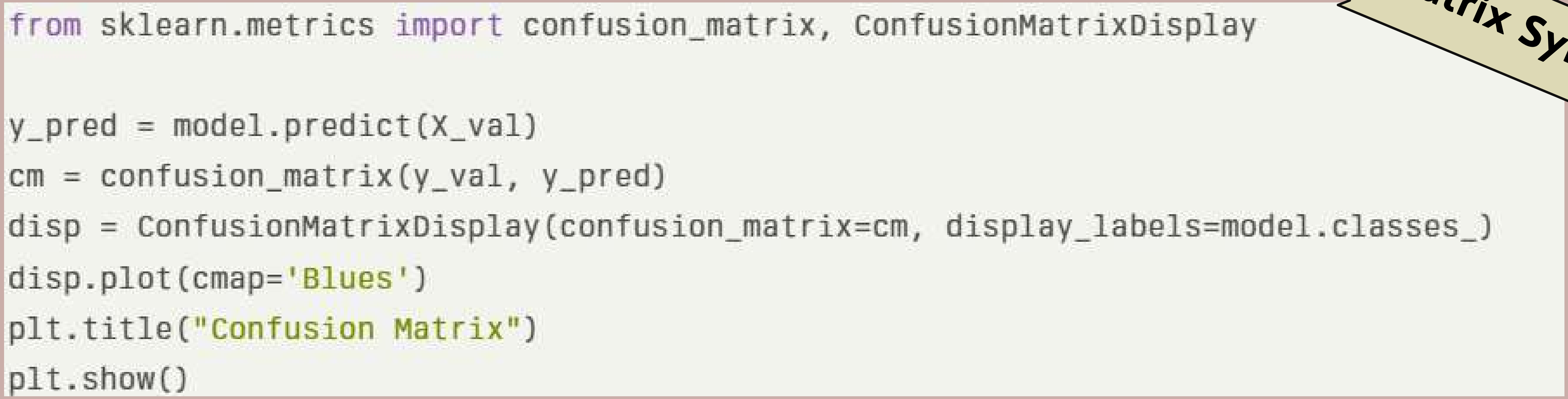


```
# Model Traning
print("Splitting data for training and validation...")
time.sleep(0.8)
features = ['Pclass', 'Sex', 'Age', 'SibSp', 'Parch', 'Fare', 'Embarked']
X = df[features]
y = df['Survived']
X_train, X_val, y_train, y_val = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y)
print("Training Random Forest model...")
time.sleep(1.5)
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
print("Model trained.")
```

Model Evaluation

- The model’s accuracy was assessed on the validation set to estimate real-world performance.
- A confusion matrix was used to visualize correct and incorrect predictions for survivor and non-survivor classes.
- A classification report provided metrics such as precision, recall, F1-score, and overall accuracy.
- Feature importance analysis highlighted which factors most strongly influenced model decisions.

Matrix Syntax

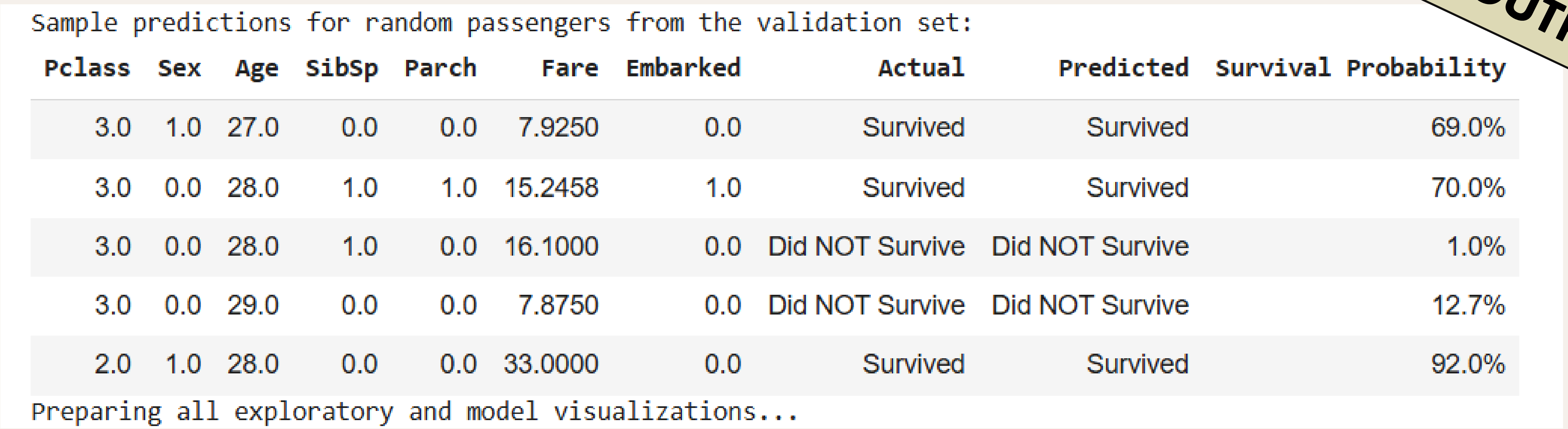


```
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay

y_pred = model.predict(X_val)
cm = confusion_matrix(y_val, y_pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=model.classes_)
disp.plot(cmap='Blues')
plt.title("Confusion Matrix")
plt.show()
```

Sample Predictions

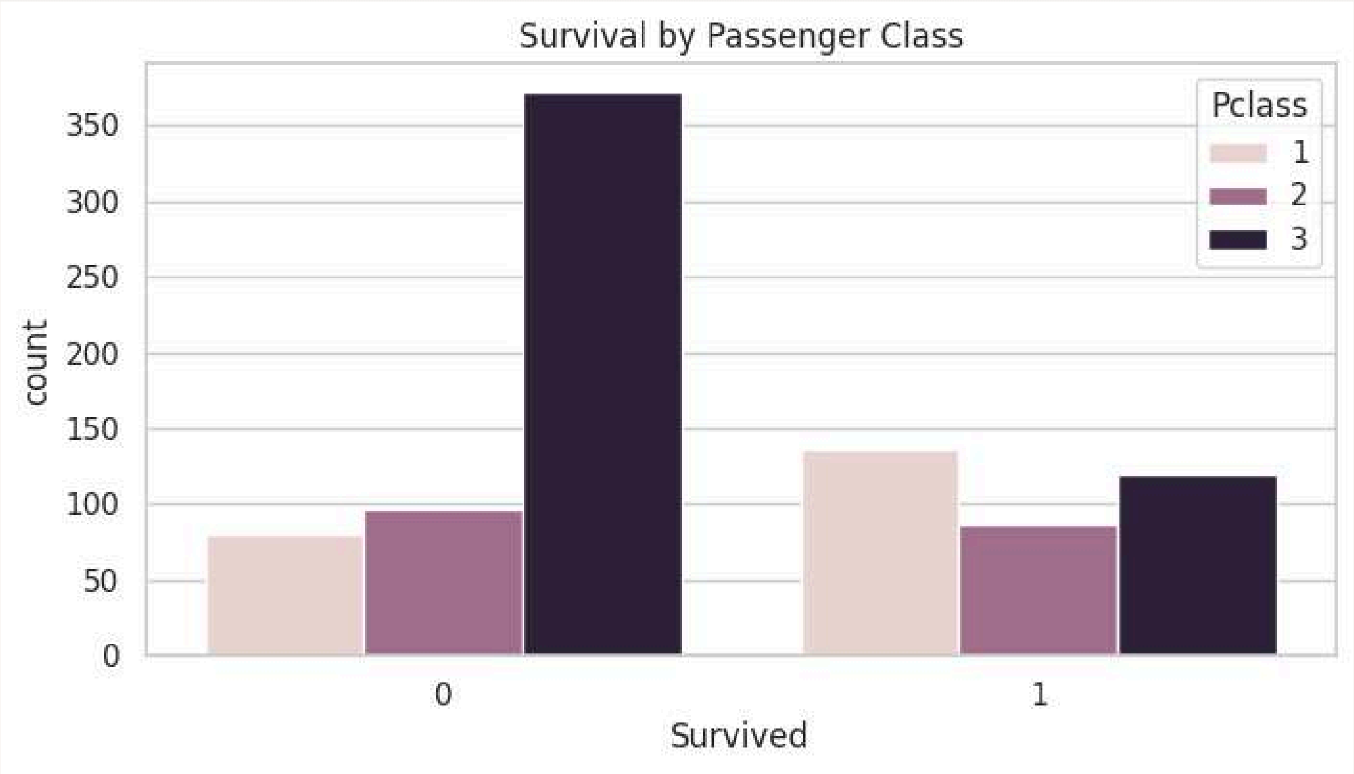
- Showcases examples of the model's predicted vs. actual survival outcomes for selected passengers.
- Helps illustrate how well the model performs on individual cases from the validation set.
- Highlights both correct and incorrect predictions, providing insight into model strengths and weaknesses.
- Makes the results more relatable by connecting predictions to real passenger data and features.

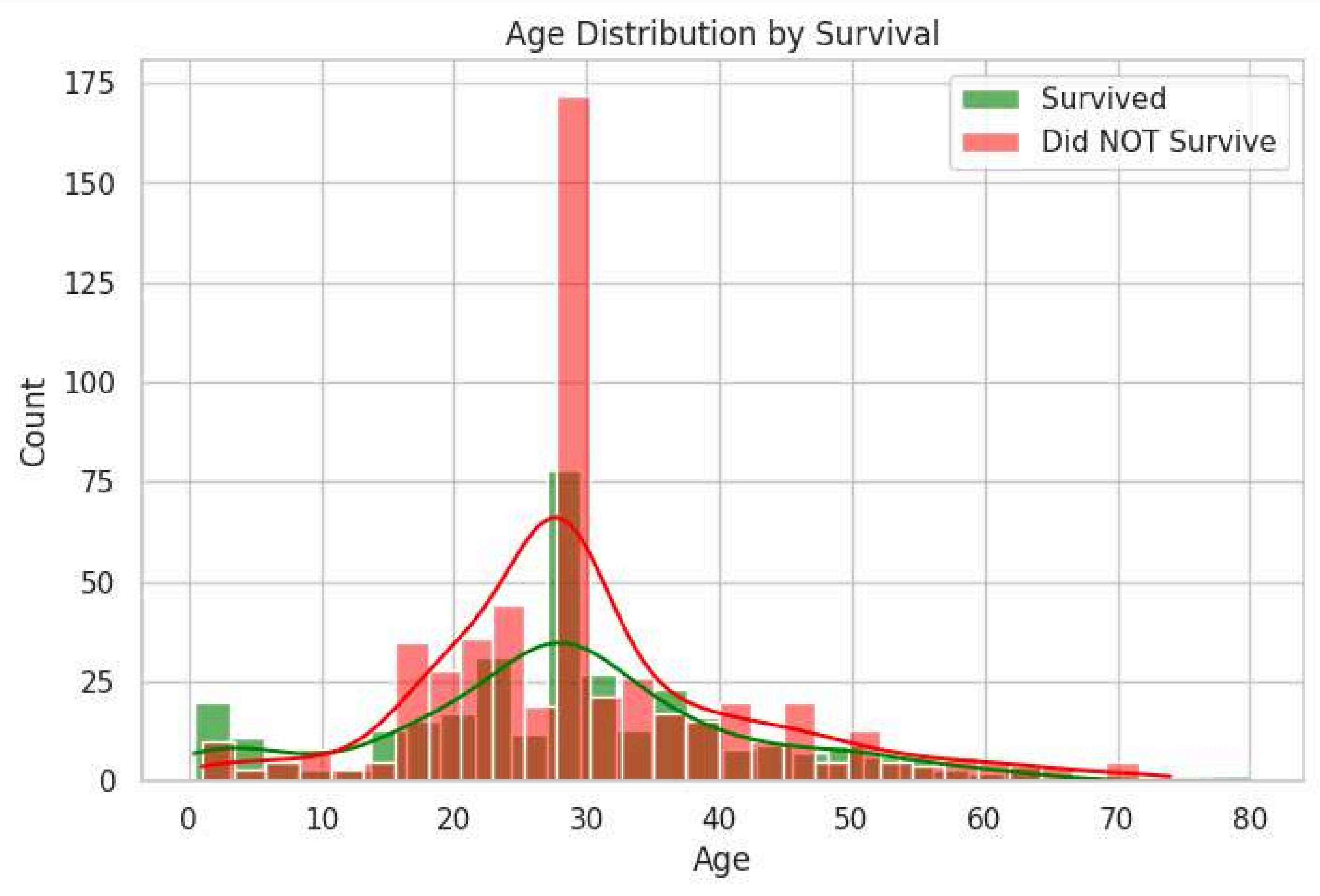
OUTPUT

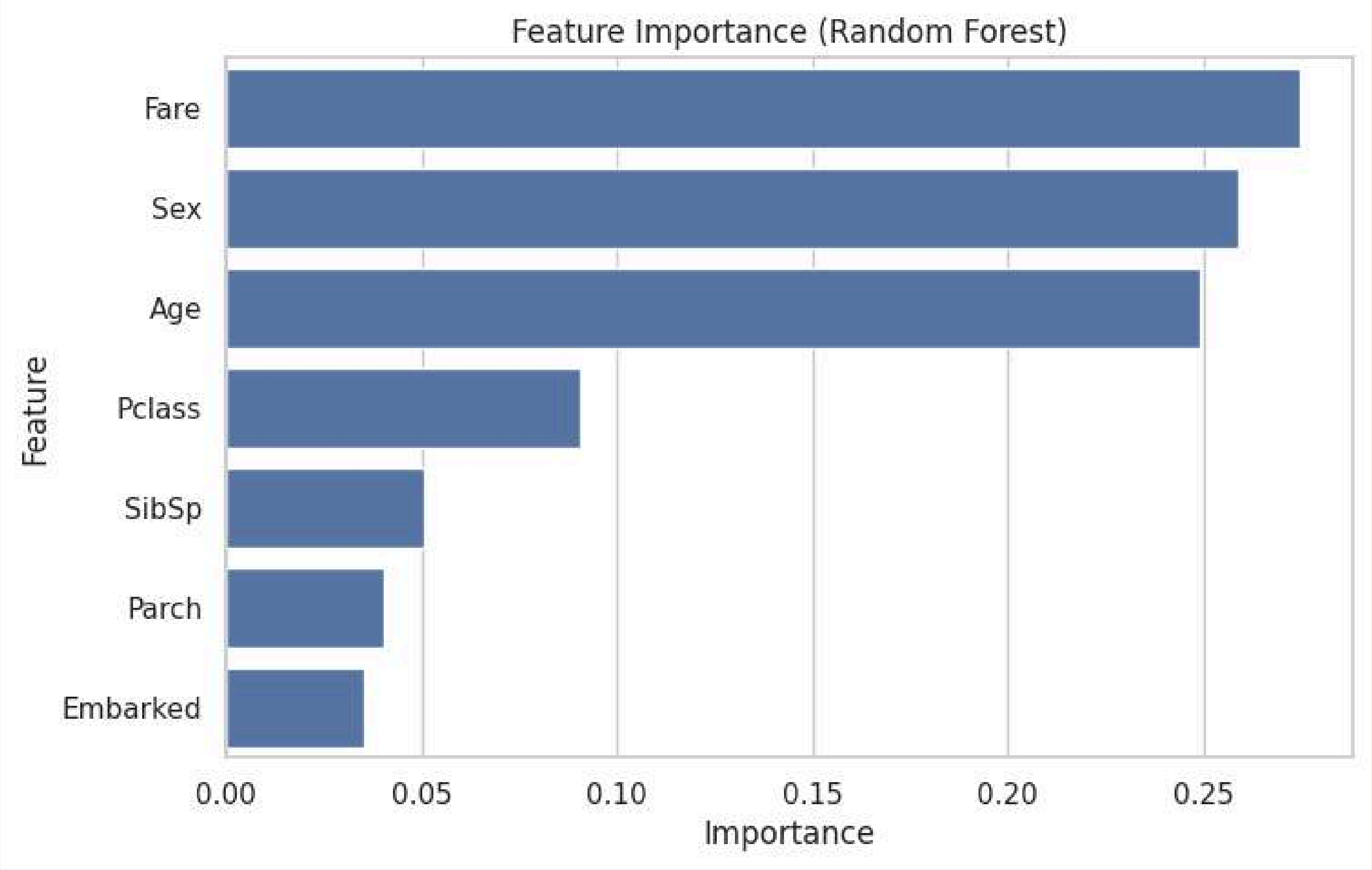


```
Sample predictions for random passengers from the validation set:
 Pclass  Sex   Age  SibSp  Parch     Fare  Embarked           Actual        Predicted  Survival Probability
    3.0  1.0  27.0    0.0    0.0   7.9250       0.0         Survived         Survived                 69.0%
    3.0  0.0  28.0    1.0    1.0  15.2458       1.0         Survived         Survived                 70.0%
    3.0  0.0  28.0    1.0    0.0  16.1000       0.0  Did NOT Survive  Did NOT Survive                  1.0%
    3.0  0.0  29.0    0.0    0.0   7.8750       0.0  Did NOT Survive  Did NOT Survive                 12.7%
    2.0  1.0  28.0    0.0    0.0  33.0000       0.0         Survived         Survived                 92.0%
Preparing all exploratory and model visualizations...
```

Visualizations:







Conclusion & Refrences

Conclusion:

- Machine learning can effectively predict survival outcomes using Titanic passenger data.
- Key factors like gender, passenger class, and age strongly influenced survival chances.
- The Random Forest model achieved good accuracy and revealed important data patterns.
- This project shows the power of data analysis for getting actionable insights from historical events.

References:

- Titanic dataset sourced from a popular online data science platform.
- Python libraries such as pandas, NumPy, matplotlib, and seaborn were used for analysis and visualization.
- Machine learning modeling and evaluation performed with the scikit-learn package.
- Data preprocessing, exploration, and results interpretation were guided by standard data science practices and documentation.

COLAB LINK : CLICK HERE 👆

Thank you!

Project BY: Varun Gupta
ROLL NO.-2312res727