Final Project Report: SML - Classification in Titanic dataset

Author: Matteo Ciccarese

1. Introduction

This report presents the final project for the Coursera course "Supervised Machine Learning: Classification." The objective is to build a predictive model using machine learning techniques to classify survival outcomes in the Titanic dataset. Various classification algorithms were implemented and evaluated to determine the best-performing model.

2. Dataset Overview

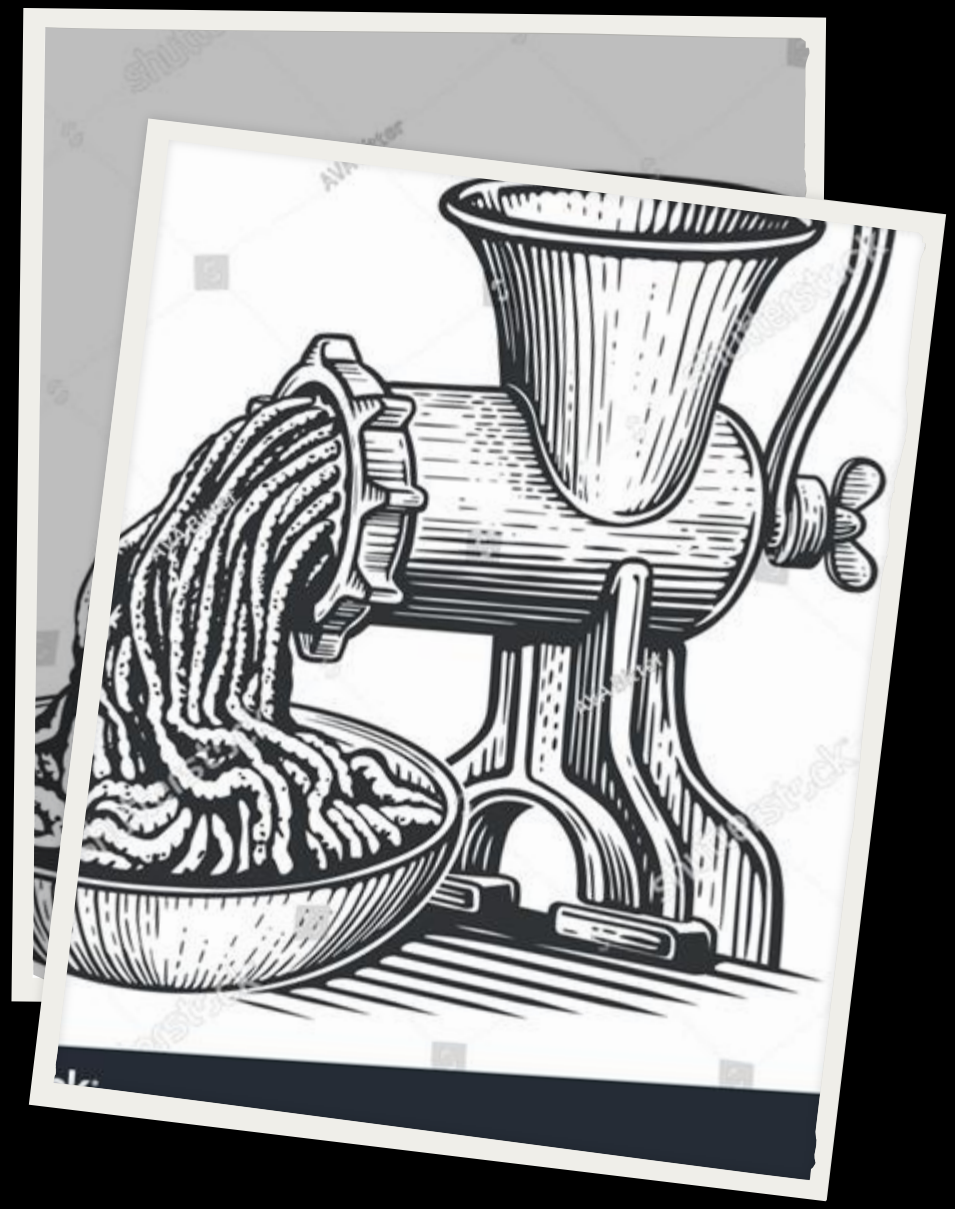
The dataset used for this project comes from the Titanic passenger list. It consists of demographic and ticket-related information, with the target variable being Survived (1 for survived, 0 for not survived). The dataset includes the following key features:

- Pclass: Ticket class (1st, 2nd, 3rd)
- Sex: Gender of the passenger
- Age: Age in years
- SibSp: Number of siblings/spouses aboard
- Parch: Number of parents/children aboard
- Fare: Ticket fare
- Embarked: Port of embarkation (C, Q, S)

3. Data Preprocessing

To ensure data quality and improve model performance, the following preprocessing steps were applied:

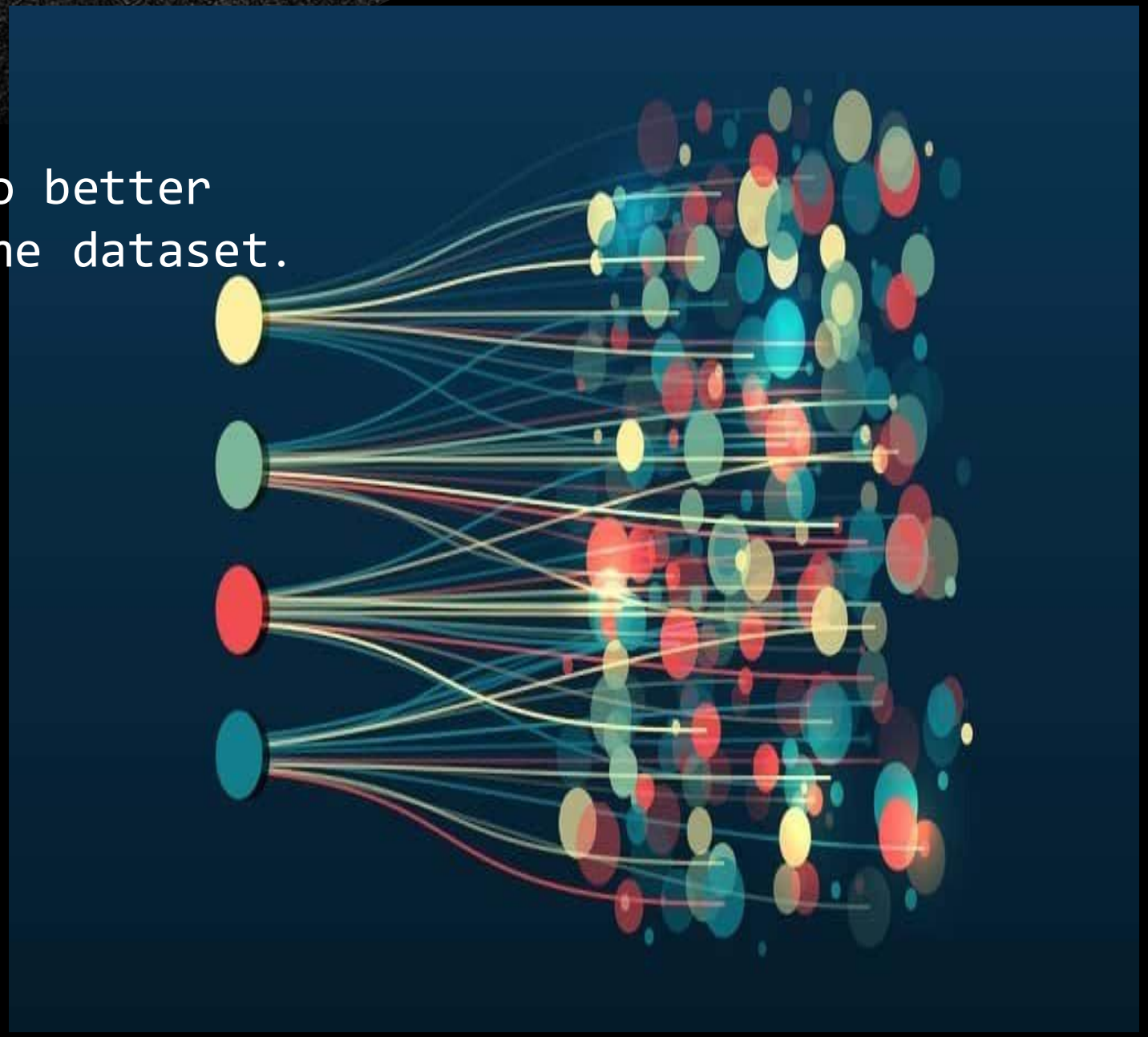
- Handling Missing Values:
  - Imputed missing Age values using the median.
  - Filled missing Embarked values with the most frequent category.
  - Replaced missing Fare values with the median.
  - Dropped the Cabin column due to excessive missing values.
- Feature Engineering:
  - Created a new feature FamilySize by combining SibSp and Parch.
- Encoding Categorical Variables:
  - Converted Sex and Embarked into numerical values using Label Encoding.
- Feature Scaling:
  - Standardized numerical features using StandardScaler to normalize feature distributions.



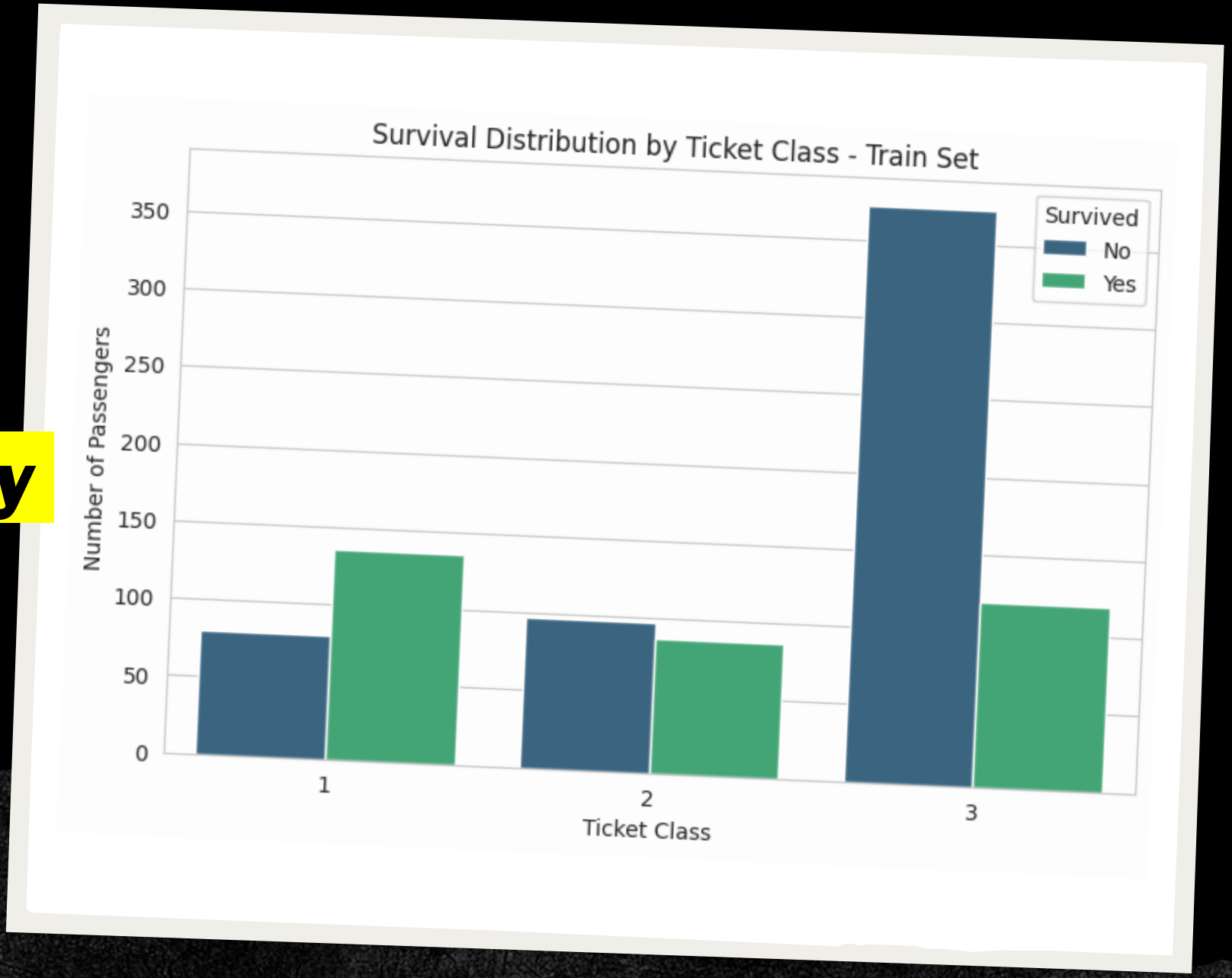
Data Visualization

We ran some visualizations to better evaluate some variables in the dataset.

- **Bar Charts:** Distribution of survival by ticket class, gender, and embarkation point
- **Correlation Heatmap:** Relationships between numerical variables
- **Boxplot:** Age distribution based on survival



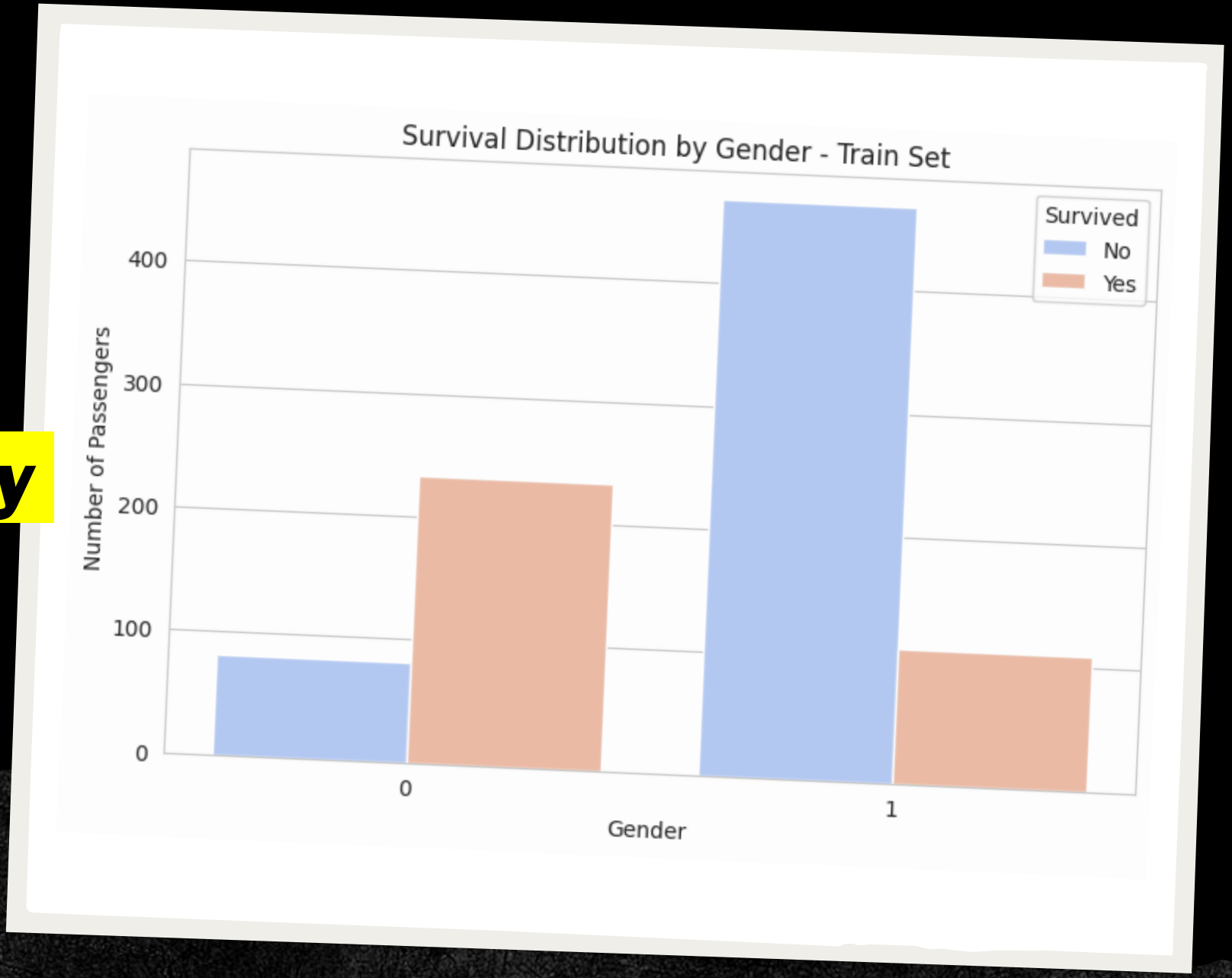
Bar Chart Survival Distribution by Ticket Class



The graph shows us that the highest number of victims occurred among third class passengers.

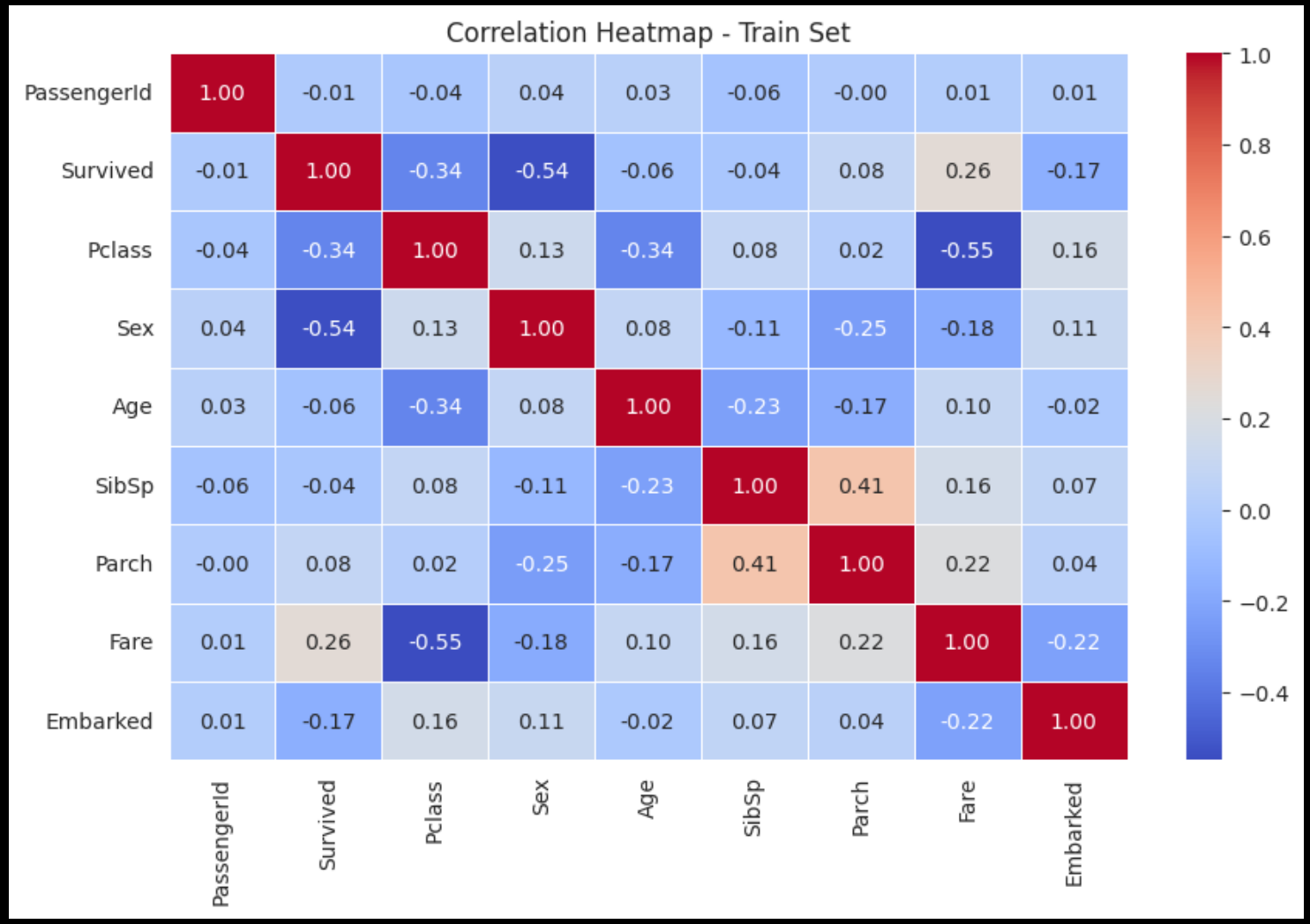
Bar Chart Survival Distribution by Gender

The graph divides the passengers into Females (0) and Males (1) showing how the victims were mainly males.



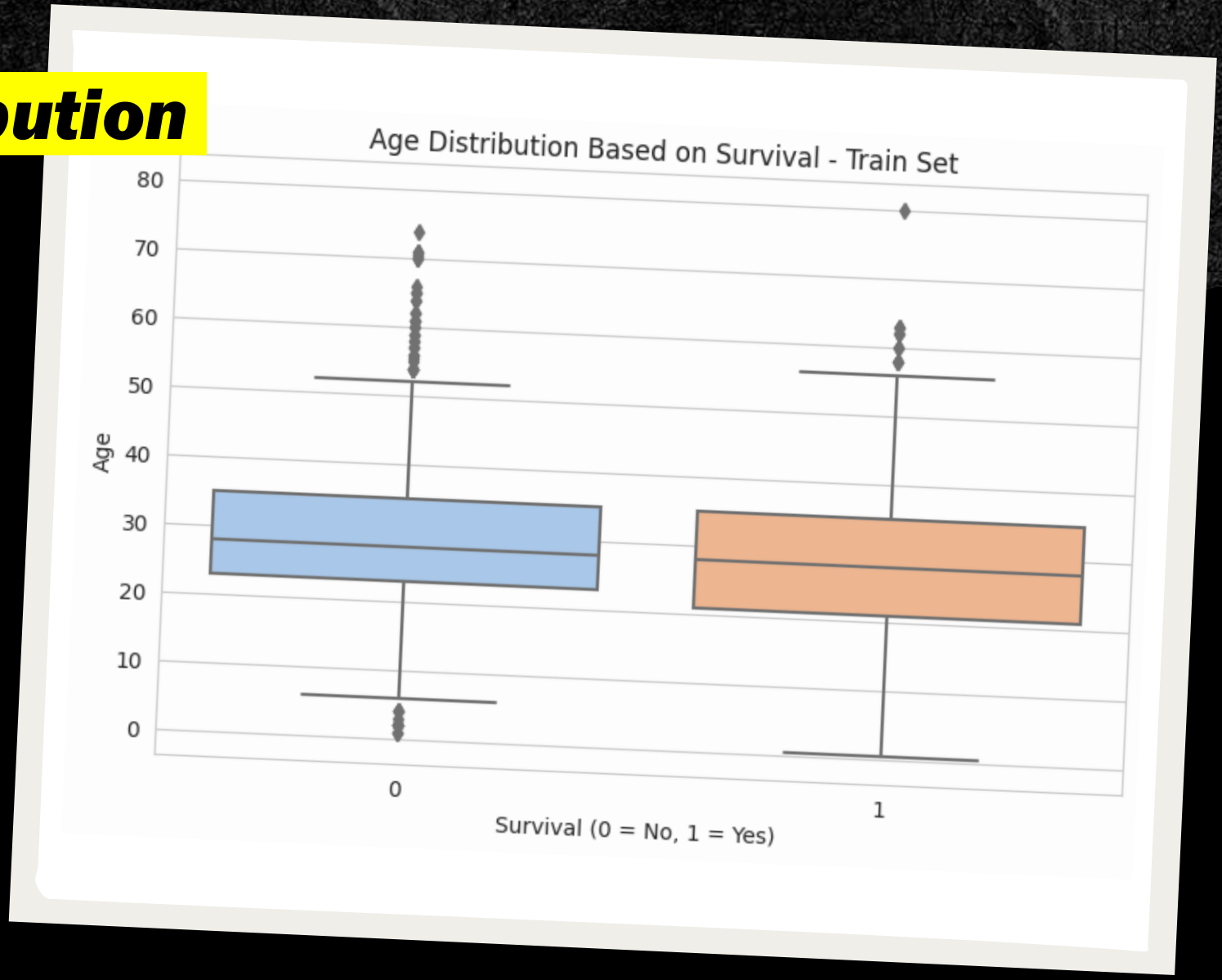
Correlation Heatmap: Relationships between numerical variables

Correlation Heatmap shows us the different correlations between variables. The highest correlations are between sex vs survival (-0.55) and class vs survival (-0.34)



Boxplot: Age distribution based on survival

This boxplot shows us that age is not a significant factor, in fact among survivors and non-survivors there is roughly the same mean and standard deviation.

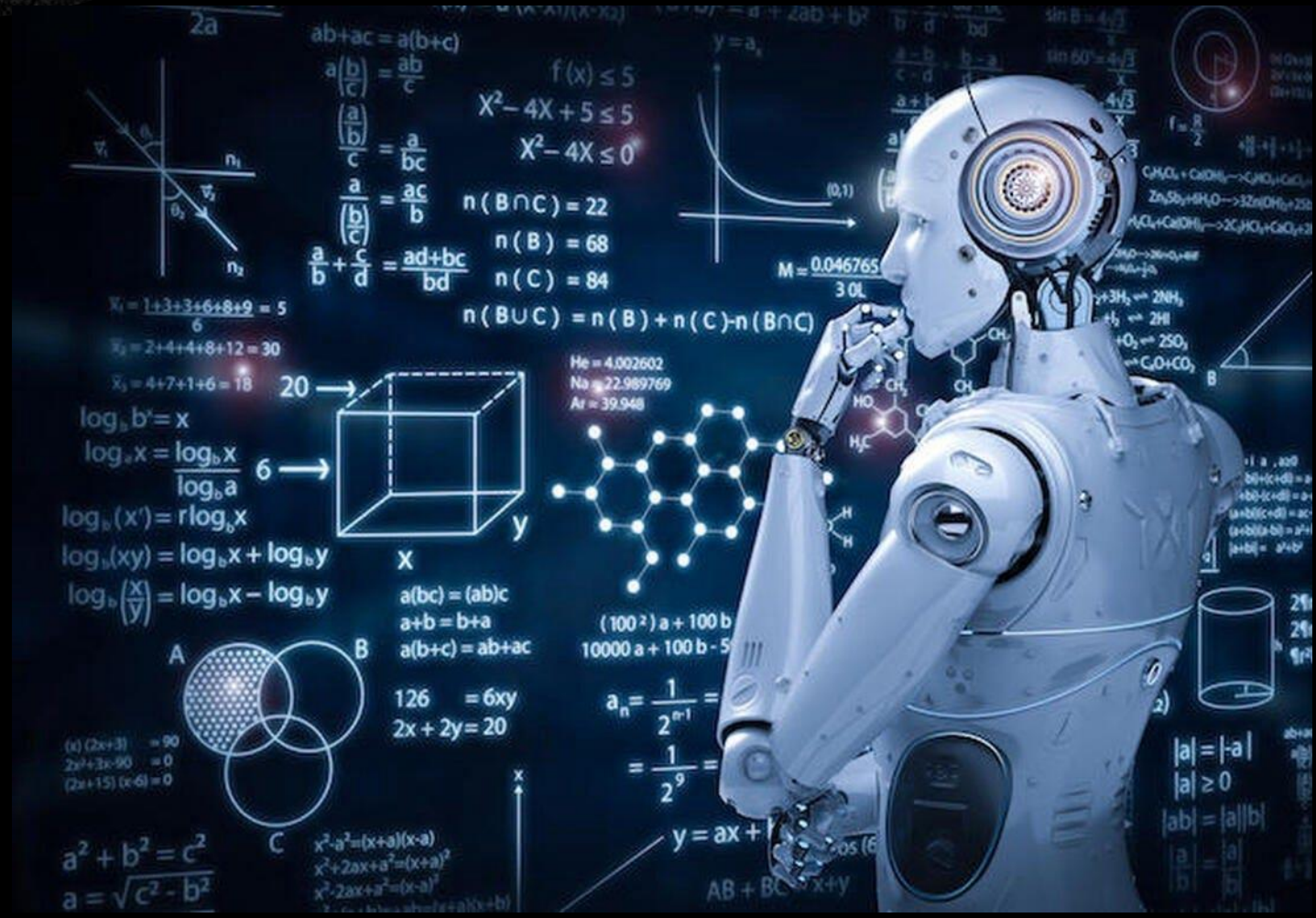


4. Model Selection and Training

Several machine learning models were trained and evaluated to determine the most effective classifier:

1. Logistic Regression
2. Random Forest Classifier
3. Support Vector Machine (SVM)
4. Gradient Boosting Classifier
5. K-Nearest Neighbors (KNN)

Each model was trained using a train-test split (80%-20%), and performance was assessed using accuracy, precision, recall, and F1-score.



Logistic Regression

- **Logistic Regression** is a statistical model used for binary classification tasks. It predicts the probability that a given input belongs to one of two classes. The model applies the **sigmoid function** to a linear combination of input features, producing an output between 0 and 1, which can be interpreted as a probability. A threshold (typically 0.5) is then used to classify the input. Logistic Regression is simple, efficient, and widely used in various domains, including medical diagnosis, spam detection, and customer churn prediction.

Logistic Regression Accuracy: 0.8045

	precision	recall	f1-score	support
0	0.82	0.86	0.84	105
1	0.78	0.73	0.76	74
accuracy			0.80	179
macro avg	0.80	0.79	0.80	179
weighted avg	0.80	0.80	0.80	179

```
[[90 15]
 [20 54]]
```

Random Forest Classifier

- **Random Forest Classifier** is an ensemble learning method that combines multiple decision trees to improve accuracy and reduce overfitting. Each tree is trained on a random subset of the data and makes an independent prediction. The final classification is determined by majority voting across all trees. This model is robust, handles large datasets well, and is less sensitive to noise, making it useful for applications like fraud detection, medical diagnosis, and recommendation systems.

Random Forest Accuracy: 0.8212

	precision	recall	f1-score	support
0	0.84	0.86	0.85	105
1	0.79	0.77	0.78	74
accuracy			0.82	179
macro avg	0.82	0.81	0.81	179
weighted avg	0.82	0.82	0.82	179

```
[[90 15]
 [17 57]]
```

Support Vector Machine (SVM)

- **Support Vector Machine (SVM)** is a powerful supervised learning algorithm used for classification tasks. It works by finding the optimal **hyperplane** that best separates data points of different classes in a high-dimensional space. SVM maximizes the **margin** between classes, making it effective in handling complex, non-linearly separable data by using **kernel functions**. It is widely used in applications like image recognition, text classification, and bioinformatics due to its high accuracy and ability to handle high-dimensional data.

Support Vector Machine Accuracy: 0.8101

	precision	recall	f1-score	support
0	0.81	0.89	0.85	105
1	0.81	0.70	0.75	74
accuracy			0.81	179
macro avg	0.81	0.79	0.80	179
weighted avg	0.81	0.81	0.81	179

```
[[93 12]
 [22 52]]
```

Gradient Boosting Classifier

- **Gradient Boosting Classifier** is an ensemble learning technique that builds a strong predictive model by combining multiple weak learners, typically decision trees. It works by training models sequentially, where each new model corrects the errors made by the previous one. The predictions of all models are then combined, with more weight given to those that performed better. This method is highly effective for both classification and regression tasks and is known for its high accuracy, especially in complex datasets. Gradient Boosting is widely used in areas like finance, marketing, and healthcare.

Gradient Boosting Accuracy: 0.8045

	precision	recall	f1-score	support
0	0.81	0.88	0.84	105
1	0.80	0.70	0.75	74
accuracy			0.80	179
macro avg	0.80	0.79	0.79	179
weighted avg	0.80	0.80	0.80	179

```
[[92 13]
 [22 52]]
```

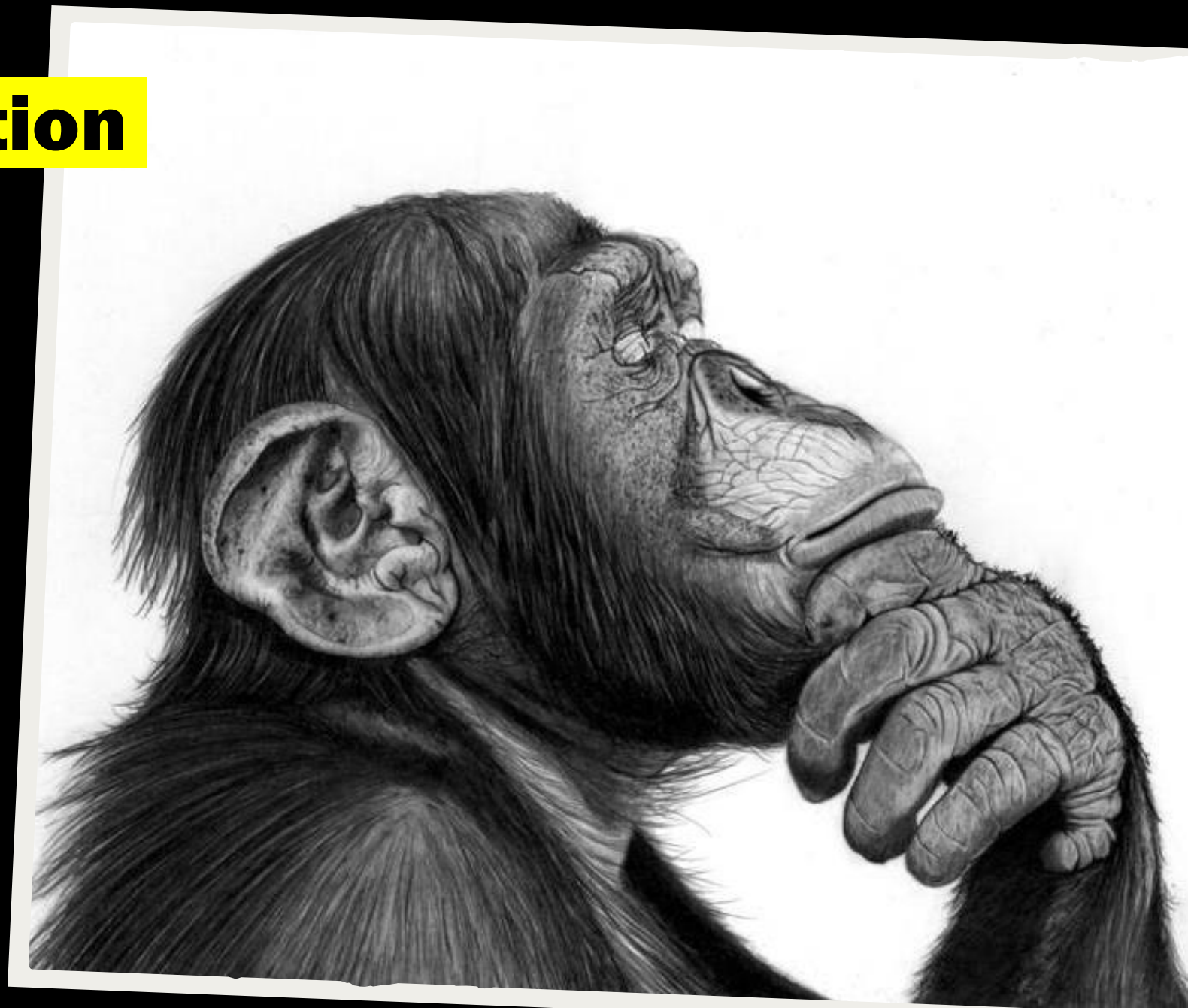
K-Nearest Neighbors (KNN)

- **K-Nearest Neighbors (KNN)** is a simple, non-parametric algorithm used for classification and regression tasks. It works by finding the 'K' nearest data points to a given input based on a distance metric (such as Euclidean distance) and assigning the most common class (for classification) or average value (for regression) among those neighbors. KNN is easy to understand, effective for small datasets, and can handle complex decision boundaries. However, it can become computationally expensive as the dataset grows and is sensitive to irrelevant features and noisy data.

```
K-Nearest Neighbors Accuracy: 0.7877
              precision    recall  f1-score   support

           0       0.81      0.83      0.82       105
           1       0.75      0.73      0.74        74

    accuracy                           0.79       179
   macro avg       0.78      0.78      0.78       179
weighted avg       0.79      0.79      0.79       179

[[87 18]
 [20 54]]
```

5. Model Evaluation and Results

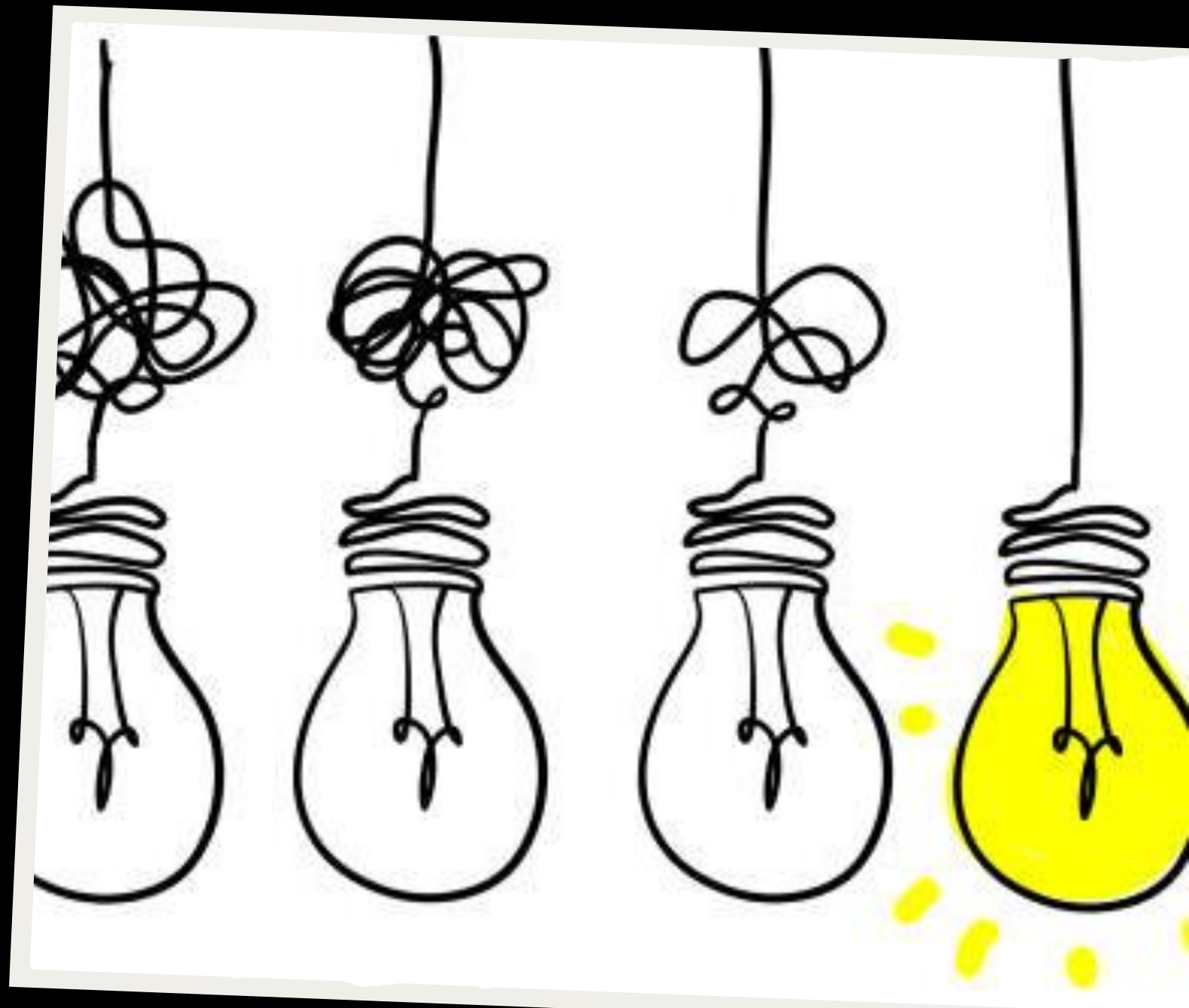
- The performance of each model is summarized below: Model Accuracy
- Logistic Regression 80.45%
- Random Forest 82.12%
- Support Vector Machine 81.01%
- Gradient Boosting 80.45%
- K-Nearest Neighbors 78.77%



6. Conclusion

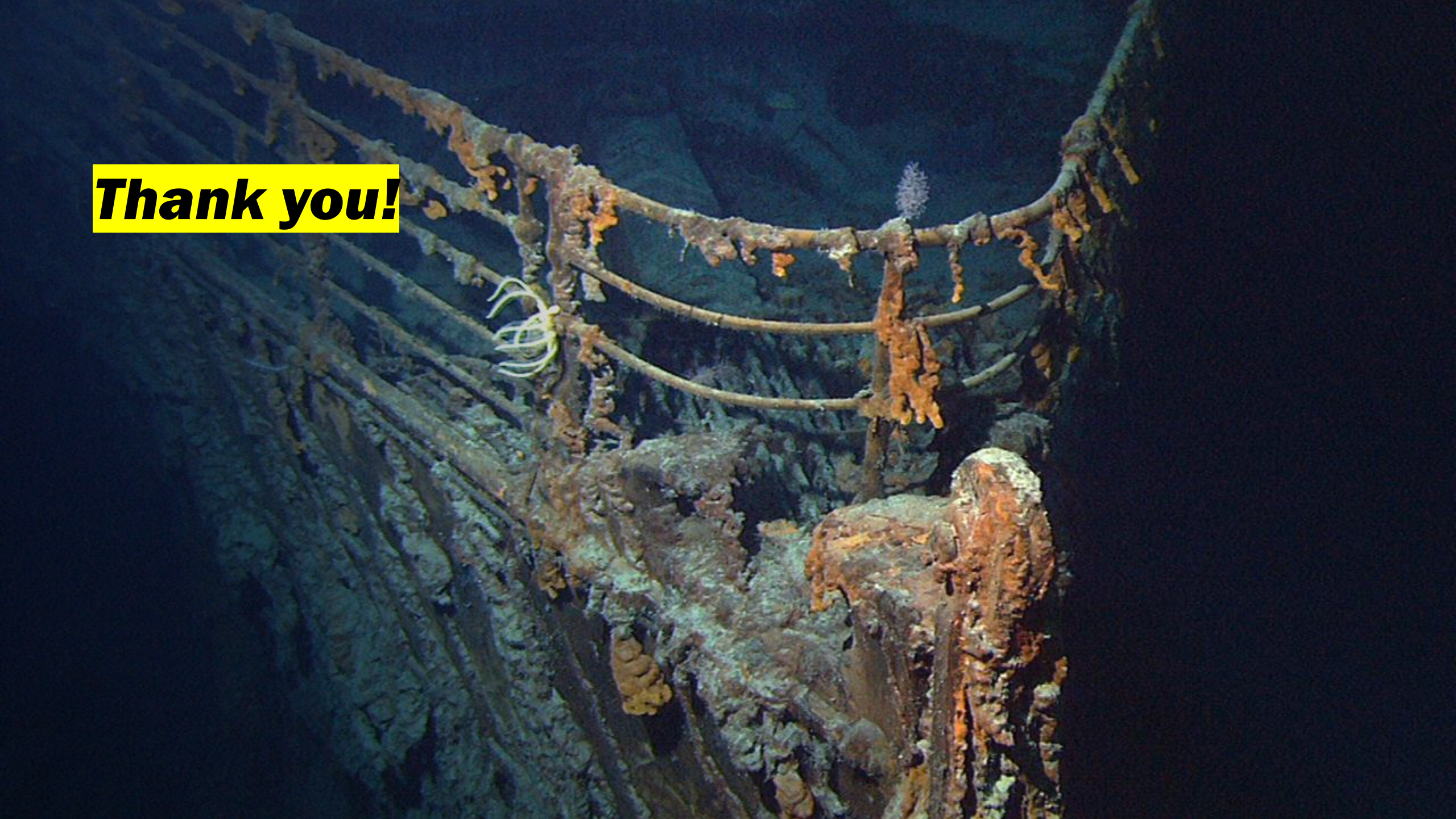
Based on the results, Random Forest was selected as the final model due to its superior performance. Future improvements may include:

- Hyperparameter tuning to optimize model performance.
- Feature selection to reduce dimensionality and improve efficiency.
- Ensemble techniques combining multiple models for better generalization.



7. Future Work

- This project demonstrated the application of supervised machine learning techniques in a classification problem. The knowledge gained can be extended to real-world applications such as medical diagnosis, fraud detection, and customer segmentation.


Thank you!