Titanic Survival Prediction: Model Development Report

Student: Yunzhen Wu

Course: MSDS 422 Practical Machine Learning

Instructor: Dr. Irene Tsapara

Date: Feb. 8, 2025

Research Question: How can we accurately predict whether a passenger survived the Titanic disaster based on available demographic and ticket-related information, and what are the important factors affecting survival?

1. Introduction

The objective of this analysis is to develop a predictive model to determine whether a passenger survived the sinking of the Titanic based on various features such as age, gender, ticket class, and family relations. The dataset consists of a training set with survival outcomes and a test set for evaluation.

In this report, we explore ensemble learning methods, specifically **Random Forest**, **Extra Trees**, and **Gradient Boosting**, to improve prediction accuracy compared to simpler models like Logistic Regression or KNN.

2. Data Loading and Exploration

2.1 Data Import

The training and test datasets were loaded using "pandas.read_csv()". An initial inspection using "info()" and "describe()" provided an overview of missing values and data distributions.

2.2 Data Overview

- The dataset includes features like "Pclass" (ticket class), "Sex", "Age", "SibSp" (siblings/spouses aboard), "Parch" (parents/children aboard), "Ticket", "Fare", "Cabin", and "Embarked" (port of embarkation).

- "Survived" is the target variable (0 = No, 1 = Yes).

- Visualizations:

- Age Distribution: The age distribution is right-skewed, with a peak around 25-30 years old. A noticeable number of passengers were children below 10, and fewer passengers were older than 60. This visualization helps in understanding potential missing age values and its impact on survival.

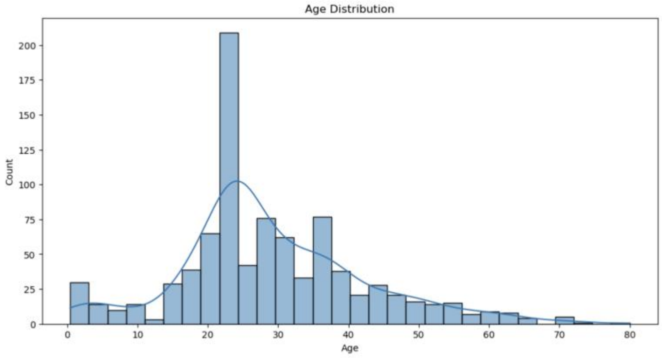


Figure 1. Age Distribution

- Fare Distribution: The fare distribution is **highly skewed** with most passengers paying low fares, while a few paid significantly higher fares exceeding 500. The skewness indicates that fare transformation might be necessary to improve model performance.

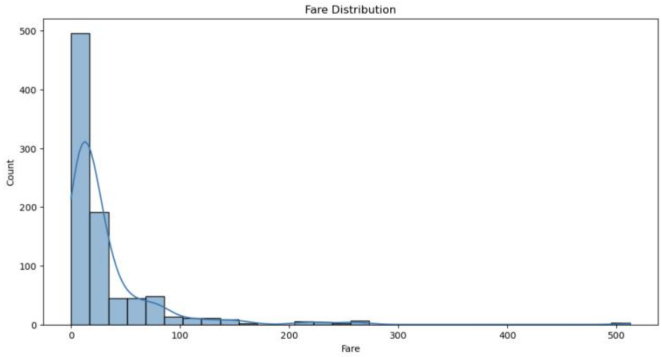
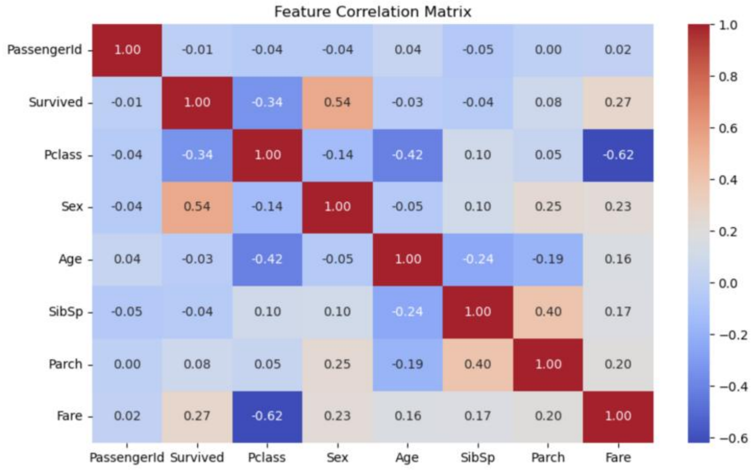


Figure 2.Fare Distribution

- Fare Transformation: To reduce skewness and normalize the distribution of Fare, we applied a **log transformation** (log1p). This transformation helps mitigate the impact of extreme values and allows models to learn patterns more effectively.
- Feature Correlation Matrix: A heatmap was generated to examine feature correlations. **Sex (being female)** has a strong positive correlation with **survival**, while **Pclass** and **Fare** also exhibit significant relationships. **Higher-class** passengers had better survival rates, while **lower-class** passengers faced higher risks.

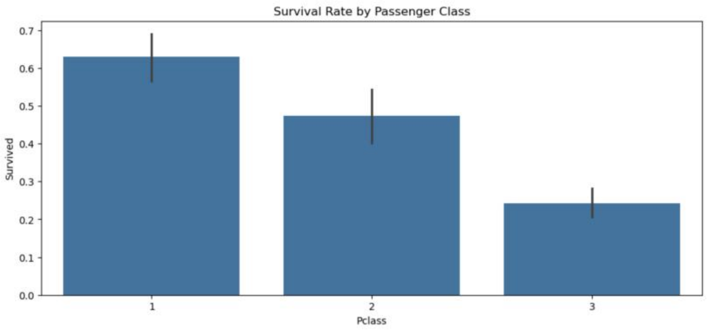


Feature Correlation Matrix

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare
PassengerId	1.00	-0.01	-0.04	-0.04	0.04	-0.05	0.00	0.02
Survived	-0.01	1.00	-0.34	0.54	-0.03	-0.04	0.08	0.27
Pclass	-0.04	-0.34	1.00	-0.14	-0.42	0.10	0.05	-0.62
Sex	-0.04	0.54	-0.14	1.00	-0.05	0.10	0.25	0.23
Age	0.04	-0.03	-0.42	-0.05	1.00	-0.24	-0.19	0.16
SibSp	-0.05	-0.04	0.10	0.10	-0.24	1.00	0.40	0.17
Parch	0.00	0.08	0.05	0.25	-0.19	0.40	1.00	0.20
Fare	0.02	0.27	-0.62	0.23	0.16	0.17	0.20	1.00

Figure 3.Feature Correlation Matrix

- Survival Rate by Passenger Class & Gender: Higher survival rates in **first-class** passengers, and **women** had significantly higher survival rates.



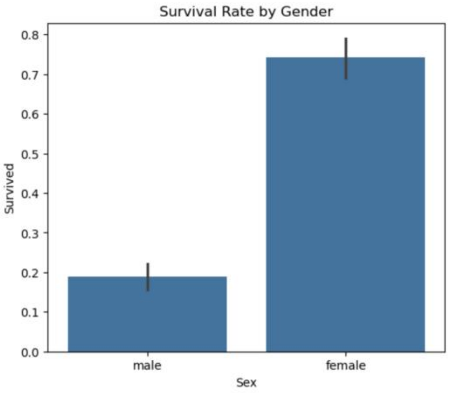


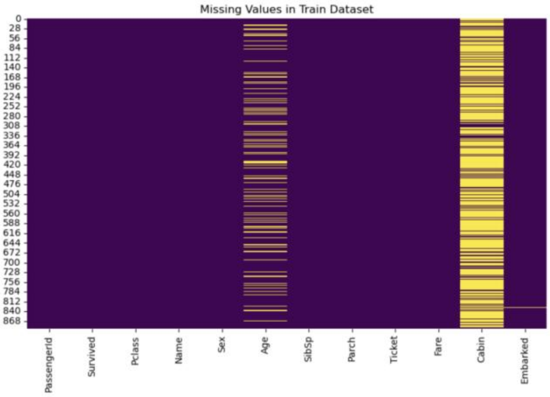
Figure 4.Survival Rate by Passenger Class

Figure 5.Survival Rate by Gender

3. Data Preprocessing and Feature Engineering

3.1 Handling Missing Values

Missing Values Heatmap: Showed significant missing values in "Age", "Fare", "Cabin", and "Embarked".



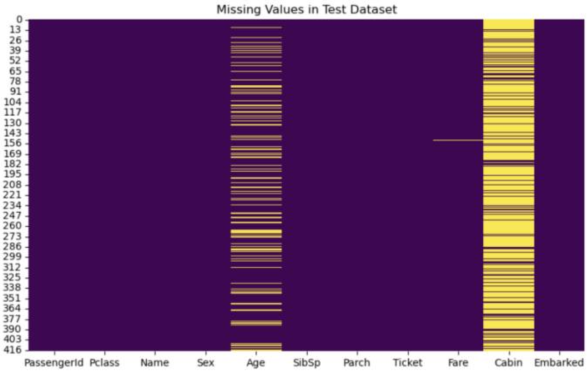
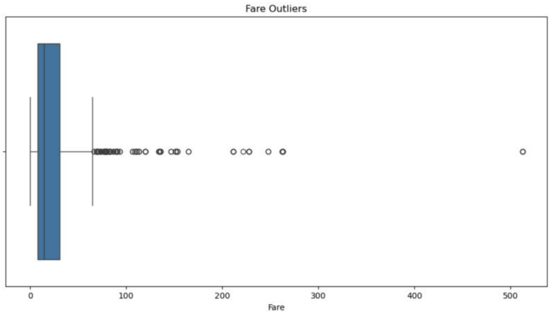


Figure 6.Missing Values in Train Dataset

Figure 7.Missing Values in Test Dataset

- "Age": Missing values were imputed with the **median** age within each "Pclass".
- "Embarked": Missing values were filled with the **most frequent** embarkation point ("S" for Southampton).
- "Fare": The single missing value was imputed with the **median** fare.
- "Cabin": Because of high missing percentage, the whole column is **removed**.

3.2 Handling Outliers



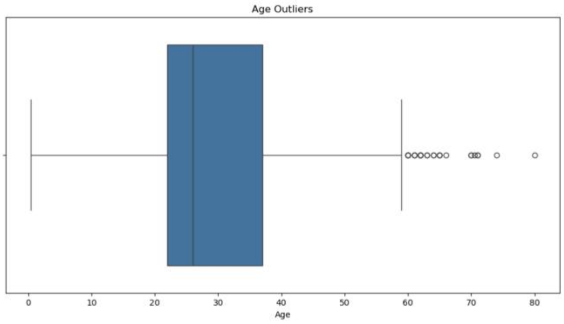


Figure 8. Fare Outliers

Figure 9.Age Outliers

- Boxplots revealed extreme values in "Fare" and "Age".

- Outliers were removed using the **99th percentile** for both "Fare" and "Age" to ensure robust model training.

3.3 Feature Engineering

- Encoding Categorical Variables:

- "Sex" was converted to numeric (0 = Male, 1 = Female).
- "Embarked" and "Pclass" were one-hot encoded.

- Family Features:

- "**FamilySize**" was created as "SibSp + Parch + 1".
- "**IsAlone**" was introduced, indicating whether the passenger traveled alone.

- Title Extraction:

- **Titles** (e.g., Mr, Mrs, Miss) were extracted from names and grouped into meaningful categories.
- Unique titles in the dataset included 'Mr', 'Mrs', 'Miss', 'Master', 'Noble', 'Officer', and 'Other'.

3.4 Feature Scaling

- **For this assignment, feature scaling was intentionally removed** as it is not necessary for tree-based models like Random Forest, Extra Trees, and Gradient Boosting. These models split data based on feature values rather than distances or gradients, making scaling redundant.

4. Model Training and Evaluation

4.1 Model Selection

The following models were trained and evaluated:

- **Random Forest Classifier:** Uses multiple decision trees to reduce variance and improve generalization.

- **Extra Trees Classifier:** Similar to Random Forest but introduces more randomness in feature selection and splits.

- **Gradient Boosting Classifier:** Boosts weak learners sequentially, optimizing performance over multiple iterations.

4.2 Hyperparameter Tuning

GridSearchCV was used to optimize the following parameters:

- **n_estimators** (number of trees): [50, 100, 500, 1000]

- **max_features** (number of features per split): [‘auto’, ‘sqrt’, ‘log2’]

- **max_depth** (depth of trees): [4, 6, 8, 10]

- **criterion** (splitting rule, only for Random Forest & Extra Trees): [‘gini’, ‘entropy’]

After optimization, the best hyperparameters were selected for each model:

Random Forest - criterion: 'entropy', max_depth: 6, max_features: 'sqrt', n_estimators: 50

Gradient Boosted Trees - max_depth: 4, max_features: 'log2', n_estimators: 100

Extra Trees - criterion: 'gini', max_depth: 6, max_features: 'sqrt','n_estimators: 50

4.3 Model Performance Metrics

- Accuracy Scores (Cross-Validation Results):

Model	Accuracy	Precision	Recall	F1 Score	AUC
Random Forest	0.822857	0.823983	0.822857	0.818369	0.850539
Gradient Boosting	0.811429	0.813318	0.811429	0.805666	0.870094
Extra Trees	0.817143	0.818640	0.817143	0.812041	0.842800

Table 1. Accuracy Scores

- **Random Forest achieved the highest accuracy** (82.29%), slightly outperforming Extra Trees (81.71%) and Gradient Boosting (81.14%).
- **Gradient Boosting had the highest AUC** (0.87), meaning it has the best ability to distinguish between classes. However, its recall (0.8114) is slightly lower than Random Forest.
- **Extra Trees performed similarly to Random Forest**, but its AUC (0.8428) is the lowest among the three models.
- Precision, Recall, and F1 Score are **well-balanced** for all models, with Random Forest and Extra Trees showing slightly better recall and overall F1 performance.

- Confusion Matrix Analysis:

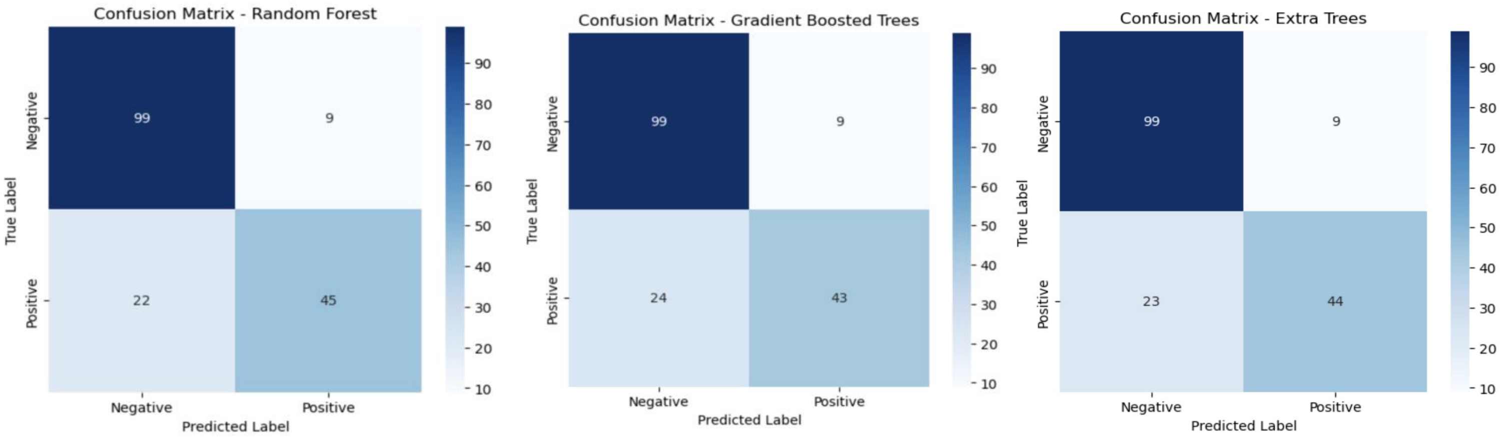


Figure 10-12. Confusion Matrix of Three Models

- Random Forest had **the lowest false negatives** (22 FN), meaning it captured the most survivors correctly.
- Gradient Boosting had **the highest false negatives** (24 FN), meaning it missed more survivors.
- Extra Trees had **slightly better FN** than Gradient Boosting but worse than Random Forest.

- ROC Curve & AUC Comparison:

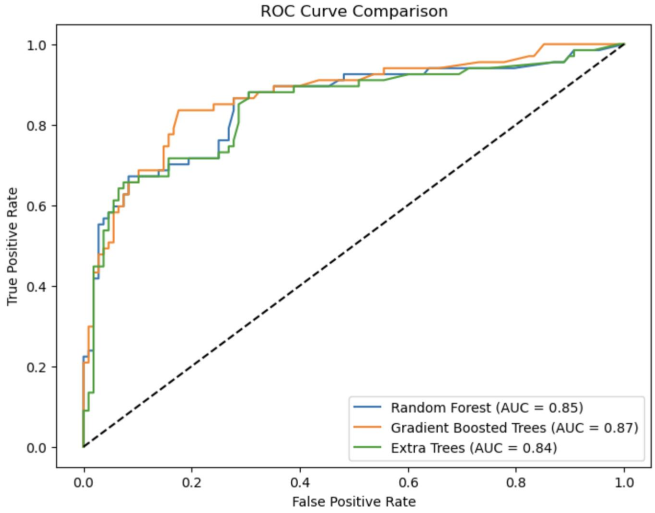


Figure 13. ROC Curve Comparison

- Gradient Boosting achieved the **highest AUC** (0.87), meaning it is the best at distinguishing survivors from non-survivors.
- Random Forest and Extra Trees had slightly **lower AUC** (0.85 and 0.84, respectively), but their performance is still strong.
- **All three models performed well**, with clear separation from the diagonal baseline (random guessing).

- Precision-Recall Curve:

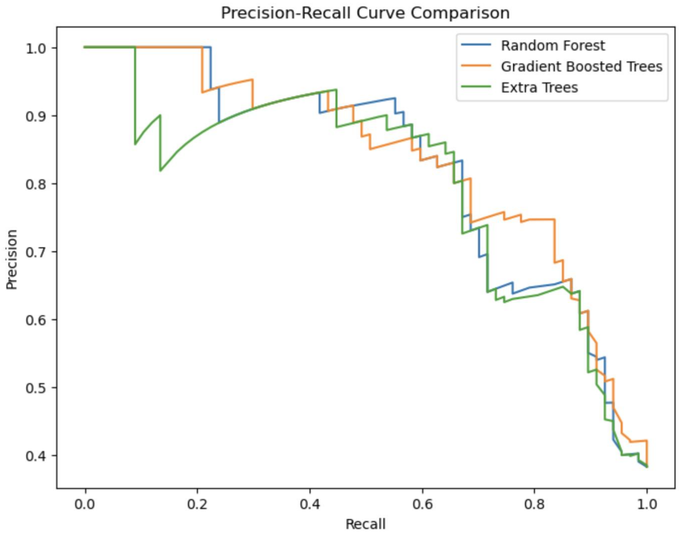


Figure 14. Precision-Recall Curve Comparison

- Gradient Boosting starts with the **highest precision** but **drops more quickly** as recall increases.
- Random Forest and Extra Trees maintain **a steadier balance** between precision and recall.
- At high recall levels, all models show some trade-offs in precision.

4.4 Overall Results of Three Models

This analysis evaluated Random Forest, Extra Trees, and Gradient Boosting for predicting Titanic survival. The models were compared based on Accuracy, Precision, Recall, F1 Score, AUC, and Confusion Matrices:

- **Best Overall Choice → Random Forest** (most balanced in accuracy, recall, and precision).
- **Best for Classification Confidence** (AUC Optimization) → **Gradient Boosting** (highest AUC).

- **Best Alternative for Balanced Performance → Extra Trees** (comparable to Random Forest).

4.5 Feature Importance Analysis

The feature importance analysis reveals which factors had the most significant impact on predicting survival in the Titanic dataset across Random Forest, Extra Trees, and Gradient Boosting models.

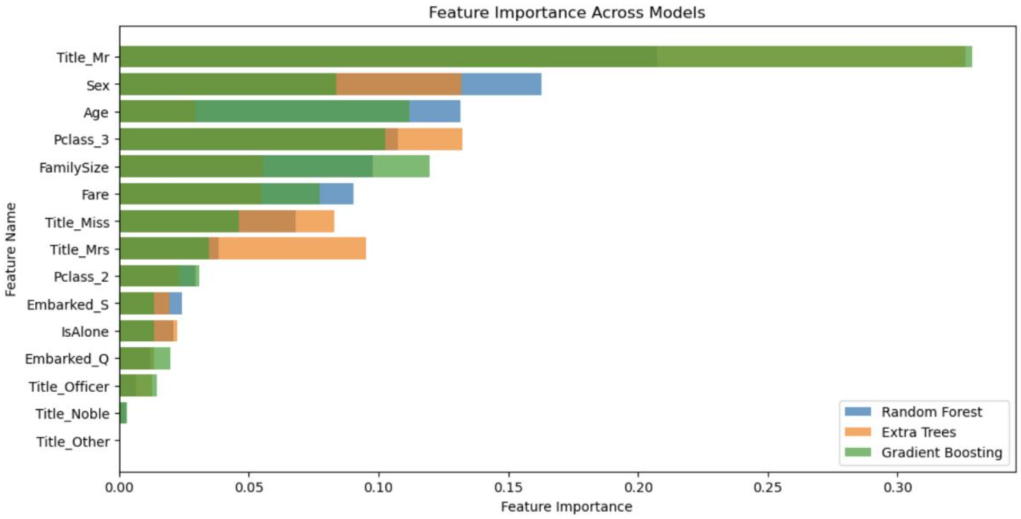


Figure 15. Feature Importance Across Models Bar Chart

- **Title_Mr is the most important** feature across all models, indicating that male passengers had a significantly lower survival rate.
- **Sex is a crucial determinant**, with females having a much higher survival probability.
- **Age and Passenger Class** (Pclass_3) are **highly influential**, suggesting that younger passengers and those in lower-class cabins had different survival probabilities.
- **Family-related features** (FamilySize, IsAlone) play a **moderate** role, showing that traveling with family members affected survival chances.

- **Fare is more important in Gradient Boosting**, which suggests a stronger focus on financial status in survival prediction.
- **Embarked location** (Embarked_S, Embarked_Q) and **specific titles** (Title_Miss, Title_Mrs) also contribute but are less influential.

5. Conclusion and Future

This analysis successfully developed Titanic survival prediction models using Random Forest, Extra Trees, and Gradient Boosting. The models were evaluated using accuracy, precision, recall, F1 score, and AUC, as well as confusion matrices and feature importance analysis. The results showed that **Random Forest was the best general model**, and Gradient Boosting was able to distinguish survivors from non-survivors well. Extra Trees performed comparable to Random Forest, making it a strong alternative model.

Feature importance analysis revealed that social and demographic factors played the most critical role in survival prediction. **Title was the most influential predictor** across all models, with passengers labeled "Mr." having the lowest survival rate, while passengers labeled "Miss" and "Mrs." had significantly higher odds of survival. Gender was another major factor, with female passengers having a greater likelihood of survival, which confirms the historical record of lifeboats prioritizing women and children. This effect was particularly pronounced in Random Forest and Extra Trees, confirming that gender itself was a strong predictor of survival.

In addition to social status, **passenger class and economic status** were also key determinants. Third class passengers have a lower chance of survival, while first class passengers have a higher chance of survival, likely due to better class accommodation and easier access to lifeboats. Fare is particularly significant in gradient boosting, indicating that economic status plays a role in the probability of survival. Family-related features have a moderate effect, indicating that passengers traveling with family have a slightly higher chance of survival.

In later studies, to improve model performance, further feature selection and engineering can refine key variables such as Title, Fare, and Pclass while removing redundant features. Hyperparameter tuning using **Bayesian optimization** or **XGBoost** can improve prediction accuracy. Ensemble methods such as stacking can combine the strengths of different models to improve robustness. These improvements will enhance prediction accuracy and strengthen historical insights.