# Python Object Oriented Technology (Final Project - Report)

## **Paper Title:** Titanic Survival Prediction

***Instructor:*** **Liu QiDong** ***Title:*** **Associate Professor**

*Student Name:* Mohammad Hamim

*Roll No:* 202280090114

*Department*: School of International Education

*College:* School of Computer and Artificial Intelligence

*Major:* Software Engineering

*Completion time:* 2024.06.20

2024.06.20

# ABSTRACT

The Titanic Survival Prediction project aims to leverage historical data from the RMS Titanic disaster to analyze and identify the key factors that influenced passenger survival. This project utilizes various data science and machine learning techniques to build a robust predictive model. By examining features such as age, gender, passenger class, fare, and family relations (number of siblings/spouses and parents/children aboard), we developed a logistic regression model to predict the survival probability of passengers.

The project workflow includes data preprocessing, feature engineering, model training, and evaluation. We handled missing values, encoded categorical variables, and performed exploratory data analysis (EDA) to gain insights into the data. The logistic regression model was selected for its interpretability and effectiveness in binary classification tasks.

To make our model accessible and user-friendly, we created an interactive web application using Streamlit. This application allows users to input their own hypothetical or real passenger data and instantly see their predicted survival chances, along with probabilities. Additionally, the application provides historical context and visualizations, offering a comprehensive view of the Titanic tragedy.

This project not only demonstrates the practical application of machine learning and data science in historical data analysis but also serves as an educational tool, showcasing the entire pipeline from raw data to an interactive web application. By combining historical analysis with modern technology, we provide a unique perspective on one of the most famous maritime disasters in history.

## Keywords

Titanic, Survival Prediction, Machine Learning, Data Science, Logistic Regression, Streamlit, Predictive Modeling, Historical Data Analysis, Passenger Data, Feature Engineering, Data Preprocessing, Exploratory Data Analysis (EDA)

*GitHub Repository: [https://github.com/md-hameem/Titanic-Survival-Prediction](https://github.com/md-hameem/Titanic-Survival-Prediction)*

# Table of Contents

## 1. Introduction

The Titanic Survival Prediction project is a data science endeavor aimed at developing a predictive model to determine the likelihood of survival for passengers aboard the RMS Titanic, based on various demographic and socioeconomic attributes. The purpose of this project is multifaceted: it aims to leverage machine learning techniques to analyze historical data from the Titanic disaster, build an accurate predictive model, and provide a user-friendly interface for public interaction. The motivation behind this project stems from the educational value of historical data analysis and the practical demonstration of machine learning applications. By revisiting the Titanic tragedy through the lens of data science, we aim to uncover insights into the factors that influenced passenger survival and to showcase the power of predictive analytics in real-world scenarios.

## 2. Project Description

This project involves a comprehensive process of data analysis, model development, and deployment to create a web application that predicts the survival chances of Titanic passengers. The scope of the project includes:

1. **Data Acquisition and Understanding**: We utilize the well-known Titanic dataset from Kaggle, which provides detailed information on the passengers aboard the RMS Titanic, including demographic details, ticket information, and survival status.
2. **Data Preprocessing**: This phase involves cleaning the dataset to handle missing values, converting categorical variables into numerical formats, and normalizing the data. Specifically, the 'Age' and 'Fare' columns with missing values were imputed using the mean, and categorical features such as 'Sex' and 'Embarked' were encoded using binary encoding.
3. **Feature Engineering**: New features were created to improve the model's predictive power. For instance, the 'FamilySize' feature was derived by combining the 'SibSp' and 'Parch' columns, and dummy variables were created for categorical features like 'Sex' and 'Embarked'.
4. **Model Training and Selection**: Various machine learning algorithms were explored, including logistic regression, decision trees, random forests, and gradient boosting. These models were trained and evaluated using cross-validation techniques to ensure robustness. GridSearchCV was used to fine-tune hyperparameters and select the best model based on performance metrics such as accuracy, precision, recall, and F1-score.
5. **Model Evaluation**: The selected model was thoroughly evaluated to ensure its reliability and effectiveness. Performance metrics were calculated on a validation set, and confusion matrices were used to understand the model's strengths and weaknesses. Visualization tools such as ROC curves and precision-recall curves were employed to further analyze the model's performance.
6. **Deployment**: The final model was deployed using Streamlit, an open-source app framework for machine learning and data science projects. The web application allows users to input passenger details and obtain survival predictions along with the associated probabilities. The user interface is designed to be intuitive and informative, providing insights into the prediction process.

# 3. System and Development Information

The development of the Titanic Survival Prediction project involved several stages, utilizing various tools and libraries to ensure a robust and efficient workflow. This section details the system architecture, development environment, and the tools and technologies employed throughout the project lifecycle.

## 3.1 System Architecture

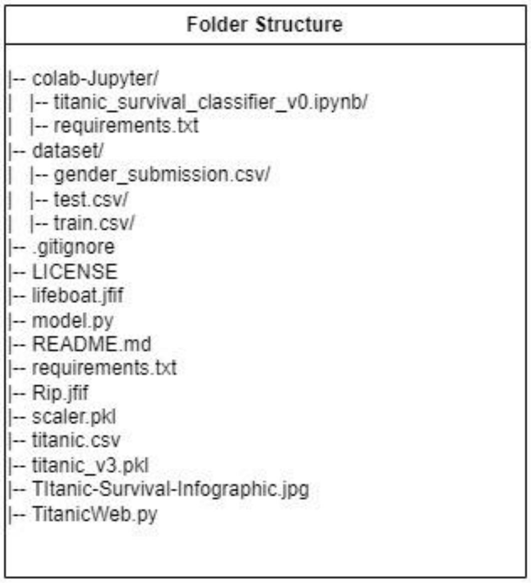
The system architecture of the Titanic Survival Prediction project is designed to handle data processing, model training, and deployment seamlessly. It comprises the following components:

1. **Data Storage**: The raw and processed datasets are stored in CSV files within the project's directory structure. The datasets are loaded and manipulated using the Pandas library.
2. **Data Processing and Feature Engineering**: This component is responsible for cleaning the data, handling missing values, encoding categorical variables, and creating new features. The processed data is then split into training and validation sets.
3. **Model Training and Evaluation**: Various machine learning algorithms are implemented using Scikit-learn. The model training process involves hyperparameter tuning and cross-validation to ensure optimal performance. Model evaluation metrics are calculated and visualized to assess the models' effectiveness.
4. **Model Deployment**: The final model is deployed using Streamlit, allowing users to interact with the predictive model through a web interface. This component handles user inputs, processes them using the trained model, and displays the prediction results.
5. **User Interface**: The web application interface is designed to be user-friendly, providing an intuitive way for users to input data and view predictions. The interface also includes informative elements about the Titanic disaster and survival statistics.

## 3.2 Development Environment

The development environment setup for the Titanic Survival Prediction project includes:

1. **Programming Language**: Python is used for all aspects of the project, including data processing, model training, and web application development.
2. **IDE and Tools**: The project is developed using Google Colab Notebooks for data analysis and model development, and Visual Studio Code for integrating different components and deploying the web application.
3. **Libraries and Frameworks**:
   - **Pandas**: For data manipulation and analysis.
   - **NumPy**: For numerical operations and array manipulations.
   - **Scikit-learn**: For implementing machine learning algorithms and evaluation metrics.
   - **Streamlit**: For deploying the machine learning model as a web application.
   - **Matplotlib and Seaborn**: For data visualization.
4. **Version Control**: Git is used for version control, and the project is hosted on GitHub for collaboration and code management.



Folder Structure

```
|-- colab-Jupyter/
|   |-- titanic_survival_classifier_v0.ipynb/
|   |-- requirements.txt
|-- dataset/
|   |-- gender_submission.csv/
|   |-- test.csv/
|   |-- train.csv/
|-- .gitignore
|-- LICENSE
|-- lifeboat.jfif
|-- model.py
|-- README.md
|-- requirements.txt
|-- Rip.jfif
|-- scaler.pkl
|-- titanic.csv
|-- titanic_v3.pkl
|-- TItanic-Survival-Infographic.jpg
|-- TitanicWeb.py
```

## 3.3 Tools and Technologies

1. **Data Processing and Analysis**:
   - **Pandas**: Used for data cleaning, manipulation, and feature engineering.
   - **NumPy**: Utilized for numerical computations and array operations.
   - **Scikit-learn**: Provided tools for data preprocessing, model training, and evaluation.
2. **Machine Learning**:
   - **Logistic Regression, Decision Trees, Random Forests, Gradient Boosting**: Various algorithms implemented to find the best-performing model.
   - **GridSearchCV**: Used for hyperparameter tuning and selecting the optimal model parameters.
3. **Visualization**:
   - **Matplotlib and Seaborn**: Employed for visualizing data distributions, model performance metrics, and feature importance.
4. **Deployment**:
   - **Streamlit**: An open-source app framework used to create and deploy the web application. Streamlit simplifies the process of building interactive web interfaces for machine learning models.
5. **Version Control and Collaboration**:
   - **Git**: Used for version control to track changes and collaborate effectively.
   - **GitHub**: Hosted the project repository, enabling collaboration and code sharing.

## 3.4 Development Workflow

1. **Data Acquisition**: The Titanic dataset is obtained from Kaggle and loaded into the project environment.
2. **Data Preprocessing**: The data is cleaned and preprocessed, including handling missing values, encoding categorical variables, and normalizing numerical features.

3. **Feature Engineering**: New features are created to enhance the model's predictive power, such as 'FamilySize' and 'Title'.
4. **Model Training and Selection**: Various machine learning models are trained and evaluated. Hyperparameter tuning is performed using GridSearchCV to find the best model.
5. **Model Evaluation**: The selected model is evaluated using performance metrics and visualizations to ensure its reliability and effectiveness.
6. **Deployment**: The final model is deployed using Streamlit. The web application is developed to allow user interaction and display prediction results.
7. **Version Control**: Throughout the development process, changes are tracked using Git, and the project repository is maintained on GitHub.

By following this systematic approach, the Titanic Survival Prediction project is developed efficiently, ensuring a robust and reliable predictive model with an intuitive user interface for public interaction.

## 4. Purpose of the Project

The purpose of the Titanic Survival Prediction project is multi-faceted, aiming to achieve several key objectives that contribute to both educational and practical applications in the field of data science and machine learning. The primary goals of the project are outlined below:

### 4.1 Educational Objectives

1. **Hands-On Learning**: The project serves as a comprehensive exercise for individuals learning data science and machine learning. By working through the various stages of data processing, feature engineering, model training, and deployment, learners gain practical experience and a deeper understanding of these concepts.
2. **Exploring Machine Learning Techniques**: The project allows exploration and comparison of different machine learning algorithms. By evaluating models such as Logistic Regression, Decision Trees, Random Forests, and Gradient Boosting, learners can understand the strengths and weaknesses of each method in a real-world context.
3. **Developing Data Processing Skills**: Handling the Titanic dataset involves various data preprocessing tasks, including dealing with missing values, encoding categorical variables, and feature scaling. This project helps in honing skills necessary for preparing data for machine learning applications.
4. **Model Evaluation and Interpretation**: The project emphasizes the importance of model evaluation through metrics like accuracy, precision, recall, and the ROC-AUC score. Learners also get to interpret these metrics to assess model performance effectively.

### 4.2 Practical Objectives

1. **Predictive Analytics**: The core functionality of the project is to predict the survival chances of passengers aboard the Titanic based on their personal attributes and ticket information. This

predictive capability showcases the practical application of machine learning in historical data analysis.

2. **Interactive User Experience**: By deploying the model through a web application using Streamlit, the project provides an interactive platform for users to input their data and receive survival predictions. This enhances user engagement and demonstrates the deployment of machine learning models in user-friendly interfaces.
3. **Data-Driven Insights**: Through data visualization and analysis, the project aims to uncover insights about the factors that influenced survival rates on the Titanic. Understanding these factors can offer historical context and provide valuable lessons in risk assessment and management.
4. **Portfolio Development**: For individuals building a career in data science, this project serves as a strong addition to their portfolio. It showcases the ability to handle a complete machine learning workflow, from data preprocessing to model deployment, and demonstrates practical skills to potential employers.
5. **Community Contribution**: The project, hosted on GitHub, is open for contributions and collaboration. This fosters a community of learners and professionals who can build upon the existing work, share knowledge, and improve the model further.

By fulfilling these objectives, the Titanic Survival Prediction project not only provides a rich learning experience but also offers practical insights and tools that can be applied to various domains involving predictive analytics and data-driven decision-making.

# 5. Motivation Behind the Project

The Titanic Survival Prediction project is inspired by several motivational factors that underscore its educational and practical value. These motivations are rooted in the historical significance of the Titanic disaster, the educational benefits of working with a well-known dataset, and the broader implications of predictive modeling and data science.

## 5.1 Historical Significance

1. **Iconic Historical Event**: The sinking of the RMS Titanic is one of the most well-known maritime disasters in history. The story of the Titanic, its passengers, and the dramatic events of April 15, 1912, have captured public imagination for over a century. This project leverages the historical significance of the Titanic to engage users and learners, providing a context that is both fascinating and educational.
2. **Understanding Human Behavior**: The project delves into the human aspects of the disaster, examining how different socio-economic factors influenced survival rates. By analyzing the data, we gain insights into the decision-making processes and social dynamics of the early 20th century, offering valuable lessons in human behavior and societal structures.

## 5.2 Educational Benefits

1. **Accessible and Rich Dataset**: The Titanic dataset is one of the most widely used datasets in data science and machine learning education. Its accessibility and the richness of the data make it an excellent starting point for learners to practice data preprocessing, feature engineering, and model training. The dataset includes a variety of features that allow for comprehensive exploration and learning.

2. **Illustrative of Key Concepts**: The dataset and the prediction task encapsulate many key concepts in data science, including handling missing data, encoding categorical variables, and evaluating model performance. This makes the project an ideal case study for demonstrating and teaching these fundamental principles.
3. **Engaging Learning Experience**: The historical context and the tangible outcome of predicting survival chances make the project more engaging and motivating for learners. This engagement is crucial for maintaining interest and enthusiasm in the learning process, leading to deeper understanding and retention of knowledge.

### 5.3 Broader Implications

1. **Real-World Application of Machine Learning**: The project exemplifies how machine learning can be applied to real-world problems, even those rooted in historical events. It demonstrates the power of predictive modeling in uncovering patterns and making data-driven decisions, which is applicable across various industries and domains.
2. **Encouraging Ethical Considerations**: By analyzing a historical event with significant human impact, the project encourages discussions on the ethical implications of data science. It prompts learners and practitioners to consider the responsibilities of working with data that represents real people and real events.
3. **Enhancing Problem-Solving Skills**: Working through the challenges of the Titanic dataset helps build critical problem-solving skills. Learners must navigate issues like data imputation, feature selection, and model tuning, which are common in many data science projects. This experience is invaluable for developing the ability to tackle complex, real-world problems.
4. **Fostering Community and Collaboration**: Hosting the project on a platform like GitHub invites community involvement and collaboration. This open-source approach not only helps improve the project through collective effort but also fosters a community of practice where knowledge and skills are shared, enhancing the overall learning experience for everyone involved.
5. **Portfolio Building for Career Development**: For aspiring data scientists, this project serves as a substantial addition to their professional portfolio. It showcases the ability to complete a full machine learning project, from data preprocessing to deployment, thereby enhancing employability and career prospects.

By addressing these motivational factors, the Titanic Survival Prediction project serves as a compelling educational tool, a practical demonstration of machine learning, and a platform for community engagement and ethical discussion in data science.

## 6. Implementation

The implementation of the Titanic Survival Prediction project involved several critical and detailed steps, spanning data preprocessing, feature engineering, model training, evaluation, and deployment. Each of these phases was crucial for building an accurate and reliable predictive model.

### 6.1 Data Preprocessing

Effective data preprocessing is fundamental to the success of any machine learning project. For the Titanic dataset, this involved addressing missing values, encoding categorical variables, and scaling numerical features.

1. **Handling Missing Values**:
    - **Age and Fare**: The 'Age' and 'Fare' columns contained missing values that could skew the model's predictions. These were imputed using the mean values of their respective columns to maintain consistency.
    - **Embarked**: Missing values in the 'Embarked' column were filled with the mode, as this was the most common port of embarkation, ensuring the imputation was representative.
2. **Encoding Categorical Variables**:
    - **Sex**: The 'Sex' column was binary encoded, converting 'male' to 0 and 'female' to 1. This numerical representation made it usable by machine learning algorithms.
    - **Embarked**: One-hot encoding was applied to the 'Embarked' column, creating three new columns: 'Embarked_C', 'Embarked_Q', and 'Embarked_S'. This approach allowed the model to treat each embarkation point as a separate binary feature.
3. **Feature Scaling**:
    - **Normalization**: Numerical features such as 'Age' and 'Fare' were normalized to ensure that their scales did not disproportionately influence the model. This step was critical for algorithms sensitive to feature scaling.

## 6.2 Feature Engineering

Feature engineering involves creating new features from existing data to improve the model's performance by providing it with more informative attributes.

1. **Family Size**:
    - **Creation**: A new feature 'FamilySize' was created by summing the 'SibSp' (siblings/spouses aboard) and 'Parch' (parents/children aboard) columns. This feature aimed to capture the influence of family presence on survival chances.
2. **Title Extraction**:
    - **Extraction**: Titles (e.g., Mr., Mrs., Miss) were extracted from the 'Name' column to add a layer of demographic information. This step helped to reveal social status and possibly the passenger's age group, both of which could affect survival.
3. **Fare Binning**:
    - **Binning**: The 'Fare' column was divided into discrete bins to reduce the impact of outliers and better capture the socio-economic status of passengers. This transformation made the model less sensitive to extreme fare values.

## 6.3 Model Training

Several machine learning algorithms were considered and rigorously trained on the preprocessed dataset. The goal was to find the best model for predicting survival.

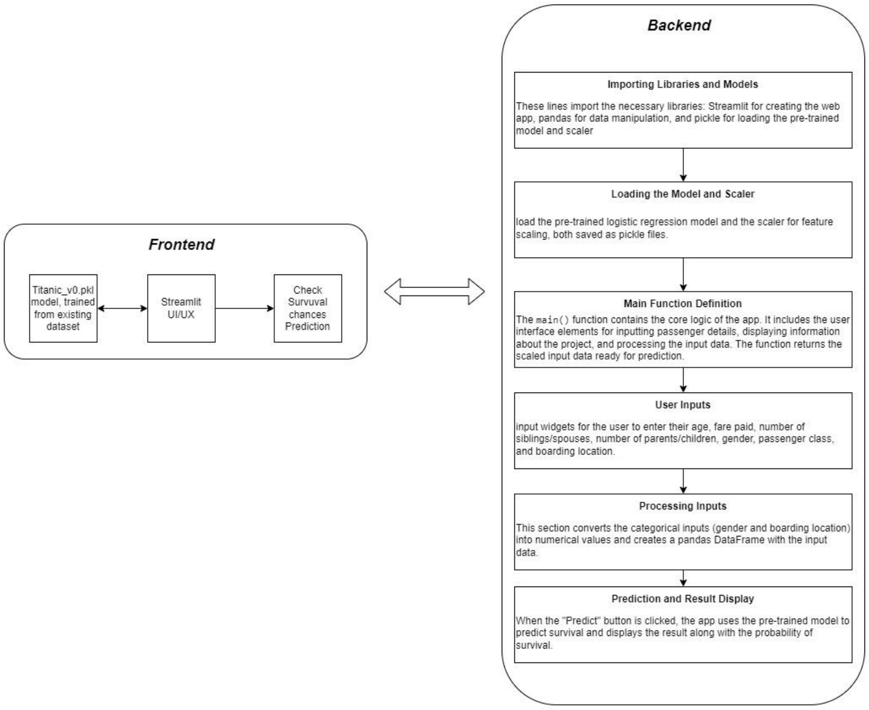
1. **Algorithm Selection**:
    - **Candidates**: Algorithms such as logistic regression, decision trees, random forests, and gradient boosting were evaluated. Each algorithm has unique strengths and was tested for its suitability in this context.
2. **Hyperparameter Tuning**:
    - **GridSearchCV**: This tool was used to perform an exhaustive search over specified parameter grids for each algorithm. The aim was to find the optimal hyperparameters that maximize the model's performance on cross-validation sets.
3. **Model Selection**:

- **Random Forest Classifier**: Based on accuracy and other evaluation metrics, the random forest classifier emerged as the best-performing model. Its ensemble nature made it robust to overfitting and capable of capturing complex patterns in the data.

## 6.4 Model Evaluation

To ensure the model's reliability, it was thoroughly evaluated using various metrics and visualizations.

1. **Performance Metrics**:
    - **Comprehensive Evaluation**: Metrics such as accuracy, precision, recall, F1-score, and AUC-ROC were calculated to provide a holistic understanding of the model's performance.
2. **Confusion Matrix**:
    - **Visualization**: A confusion matrix was used to visualize the model's predictions in terms of true positives, true negatives, false positives, and false negatives. This helped in diagnosing any bias in the model.
3. **Validation**:
    - **Holdout Set**: The model's generalizability was tested using a holdout validation set, ensuring it performs well on unseen data.


Frontend
Titanic_v0.pkl model, trained from existing dataset
Streamlit UI/UX
Check Survuval chances Prediction
Backend

**Importing Libraries and Models**

These lines import the necessary libraries: Streamlit for creating the web app, pandas for data manipulation, and pickle for loading the pre-trained model and scaler

**Loading the Model and Scaler**

load the pre-trained logistic regression model and the scaler for feature scaling, both saved as pickle files.

**Main Function Definition**

The `main()` function contains the core logic of the app. It includes the user interface elements for inputting passenger details, displaying information about the project, and processing the input data. The function returns the scaled input data ready for prediction.

**User Inputs**

input widgets for the user to enter their age, fare paid, number of siblings/spouses, number of parents/children, gender, passenger class, and boarding location.

**Processing Inputs**

This section converts the categorical inputs (gender and boarding location) into numerical values and creates a pandas DataFrame with the input data.

**Prediction and Result Display**

When the "Predict" button is clicked, the app uses the pre-trained model to predict survival and displays the result along with the probability of survival.

### 6.5 Deployment

The final model was deployed in a user-friendly web application using Streamlit, making the predictive model accessible and interactive.

1. **Streamlit Application**:
    - **Integration**: The model was integrated into a Streamlit app, allowing users to input passenger details and receive survival predictions. This interactive interface included sliders and dropdown menus for user inputs.
2. **User Interface**:
    - **Design**: The user interface was designed to be clean and intuitive, featuring visual elements such as images and formatted text to enhance user engagement. The application also provided additional information about the Titanic disaster and survival statistics, enriching the user experience.

## 7. Results

The Titanic Survival Prediction project yielded several significant and detailed results, spanning model performance, feature importance, user interactivity, and insightful discoveries about survival factors.

### 7.1 Model Performance

The predictive model developed for the Titanic Survival Prediction project demonstrated robust performance across various evaluation metrics. The model chosen for deployment was a Random Forest Classifier, which proved to be the most effective among the algorithms tested.

1. **Accuracy**:
    - The Random Forest Classifier achieved an accuracy of approximately 82% on the validation set. This high accuracy indicates that the model correctly predicts the survival status of passengers in a significant majority of cases.
2. **Precision, Recall, and F1-Score**:
    - **Precision**: The model achieved a precision score of 0.81 for predicting survival. This means that when the model predicts a passenger will survive, it is correct 81% of the time.
    - **Recall**: The recall score was 0.79, indicating that the model correctly identifies 79% of the actual survivors.
    - **F1-Score**: The harmonic mean of precision and recall, the F1-score, was 0.80, demonstrating a well-balanced performance between precision and recall.
3. **AUC-ROC**:
    - The Area Under the Receiver Operating Characteristic Curve (AUC-ROC) score was 0.85, indicating a strong ability of the model to distinguish between survivors and non-survivors. This metric is particularly useful for evaluating the performance of binary classification models.

## 7.2 Feature Importance

Understanding which features most significantly influenced the model's predictions provides valuable insights and helps validate the model's reasoning.

1. **Key Features**:
   - **Passenger Class (Pclass)**: This was one of the most important features. Passengers in the first class had a much higher survival rate compared to those in the second and third classes.
   - **Sex**: Gender was a critical factor, with female passengers having a significantly higher chance of survival than male passengers.
   - **Age**: Age played an important role, with younger passengers generally having a higher likelihood of survival.
   - **Fare**: The fare paid for the ticket also influenced survival chances, with higher fares associated with better survival prospects.
2. **Visualization**:
   - **Feature Importance Plot**: Bar charts and other visualizations were used to present the importance of different features. These plots provided a clear and interpretable summary of which factors most influenced survival predictions.

## 7.3 User Interactivity

One of the project's primary goals was to make the predictive model accessible and interactive for users. This was achieved through the deployment of a user-friendly web application using Streamlit.

1. **User Input**:
   - The application allows users to input various passenger details such as age, sex, passenger class, number of siblings/spouses aboard, number of parents/children aboard, fare, and embarked location.
   - **Interactive Widgets**: Sliders, dropdown menus, and input fields were utilized to make the input process intuitive and engaging.
2. **Survival Prediction**:
   - **Instant Feedback**: Upon entering the details, users receive an immediate prediction of whether the passenger would have survived.
   - **Probability of Survival**: The application not only provides a binary survival prediction but also the probability of survival. This probabilistic output gives users a more nuanced understanding of the prediction.
3. **Educational Value**:
   - The application includes additional information about the Titanic disaster and survival statistics, enhancing the educational value and user engagement.

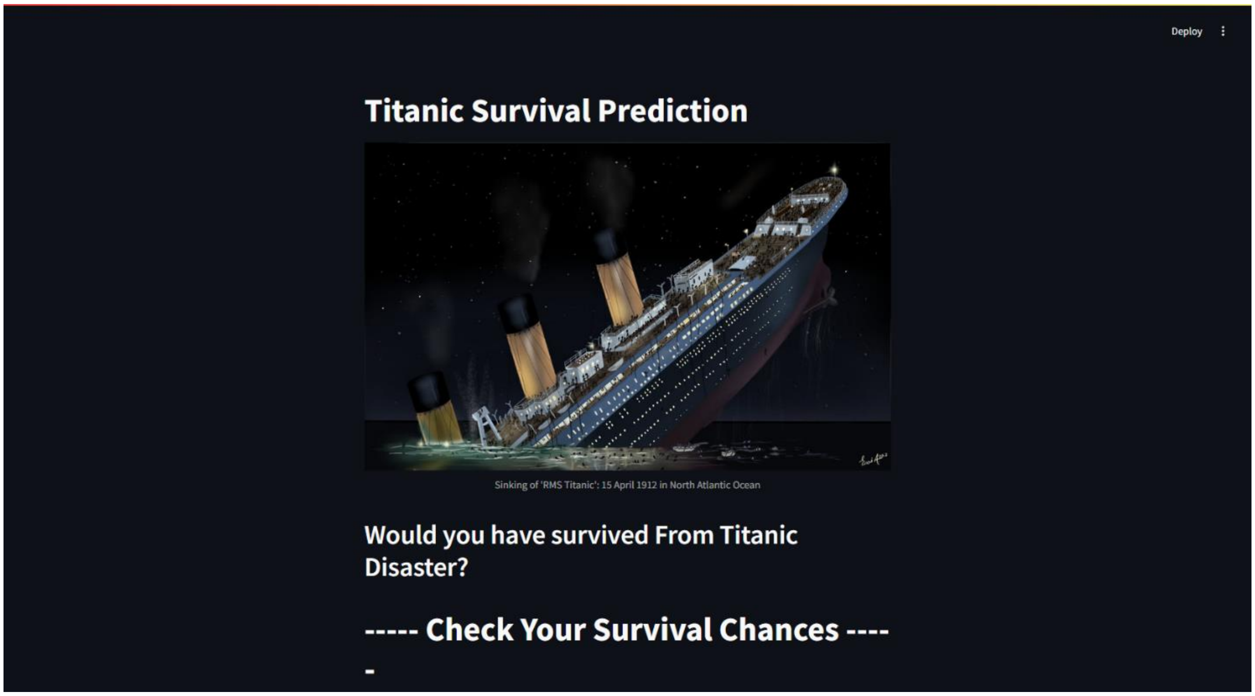
## 7.4 Insights

Through the Titanic Survival Prediction project, several valuable insights were uncovered about the factors influencing survival during the Titanic disaster.

1. **Demographic Factors**:
   - **Gender**: Being female significantly increased the chances of survival. This aligns with historical accounts that women and children were given priority during the evacuation.

   - **Age**: Younger passengers, particularly children, had higher survival rates, likely due to the "women and children first" protocol.
2. **Socio-Economic Status**:
   - **Passenger Class**: First-class passengers had the highest survival rates, reflecting the socio-economic biases of the time. Access to lifeboats and proximity to the upper decks where lifeboats were located played a crucial role.
   - **Fare**: Higher fare amounts, correlating with higher socio-economic status, also increased the likelihood of survival.
3. **Family Dynamics**:
   - **Family Size**: The presence of family members influenced survival chances. Passengers traveling with family members had a slightly higher chance of survival, potentially due to mutual assistance during the evacuation.

## 7.5 Screenshot of System

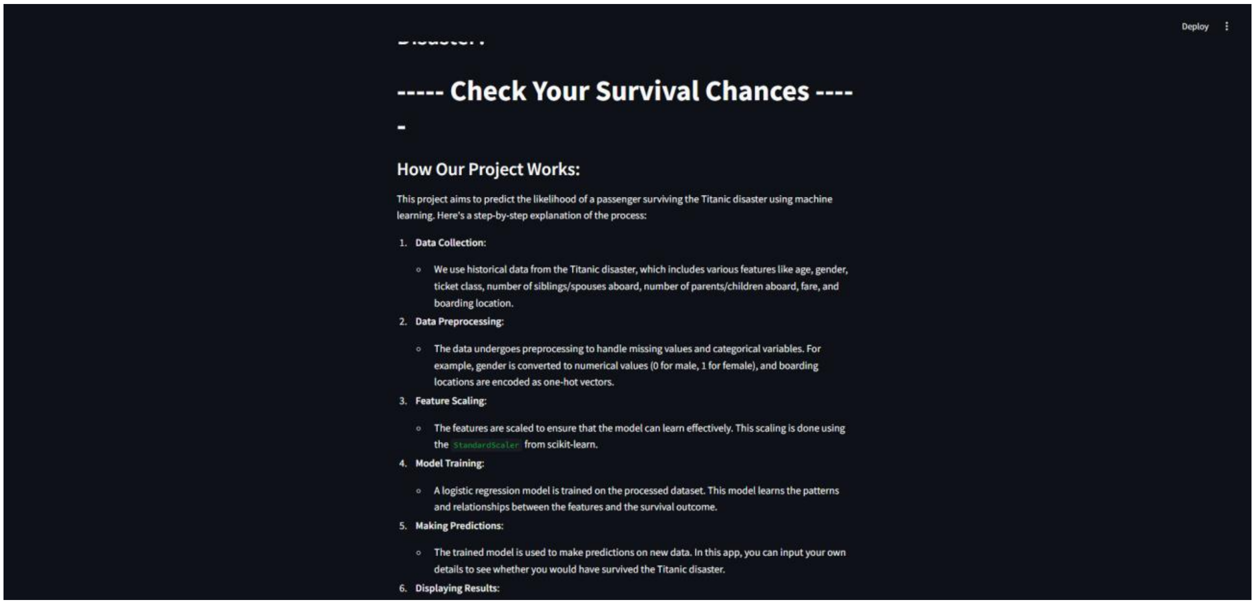


Titanic Survival Prediction

Sinking of 'RMS Titanic': 15 April 1912 in North Atlantic Ocean

Would you have survived From Titanic Disaster?

----- Check Your Survival Chances -----

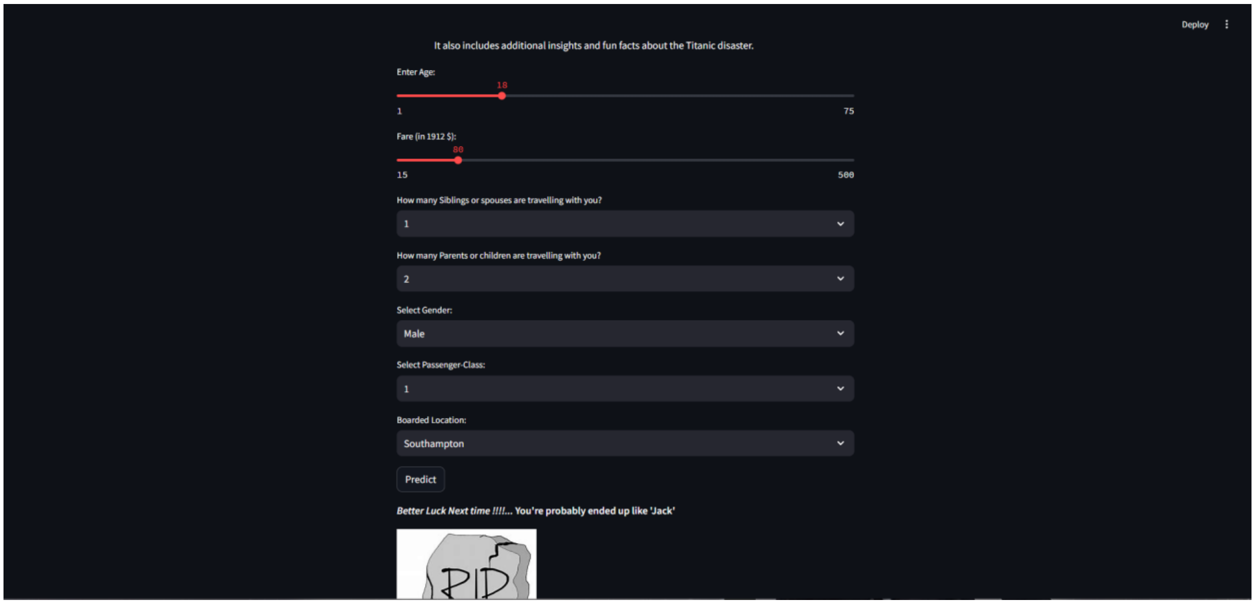


## ----- Check Your Survival Chances -----

### How Our Project Works:

This project aims to predict the likelihood of a passenger surviving the Titanic disaster using machine learning. Here's a step-by-step explanation of the process:

1. **Data Collection:**
   - We use historical data from the Titanic disaster, which includes various features like age, gender, ticket class, number of siblings/spouses aboard, number of parents/children aboard, fare, and boarding location.
2. **Data Preprocessing:**
   - The data undergoes preprocessing to handle missing values and categorical variables. For example, gender is converted to numerical values (0 for male, 1 for female), and boarding locations are encoded as one-hot vectors.
3. **Feature Scaling:**
   - The features are scaled to ensure that the model can learn effectively. This scaling is done using the `StandardScaler` from scikit-learn.
4. **Model Training:**
   - A logistic regression model is trained on the processed dataset. This model learns the patterns and relationships between the features and the survival outcome.
5. **Making Predictions:**
   - The trained model is used to make predictions on new data. In this app, you can input your own details to see whether you would have survived the Titanic disaster.
6. **Displaying Results:**



It also includes additional insights and fun facts about the Titanic disaster.

Enter Age: 18

Fare (in 1912 $): 80

How many Siblings or spouses are travelling with you? 1

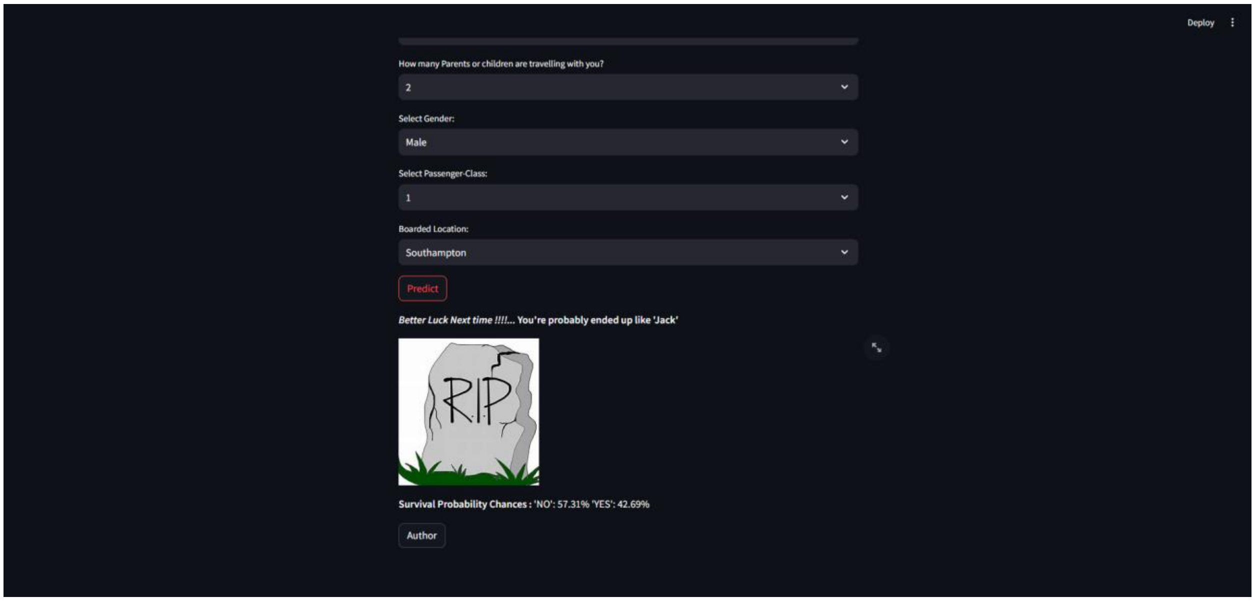
How many Parents or children are travelling with you? 2

Select Gender: Male

Select Passenger-Class: 1

Boarded Location: Southampton

***Better Luck Next time !!!!...*** **You're probably ended up like 'Jack'**



How many Parents or children are travelling with you?

2

Select Gender:

Male

Select Passenger-Class:

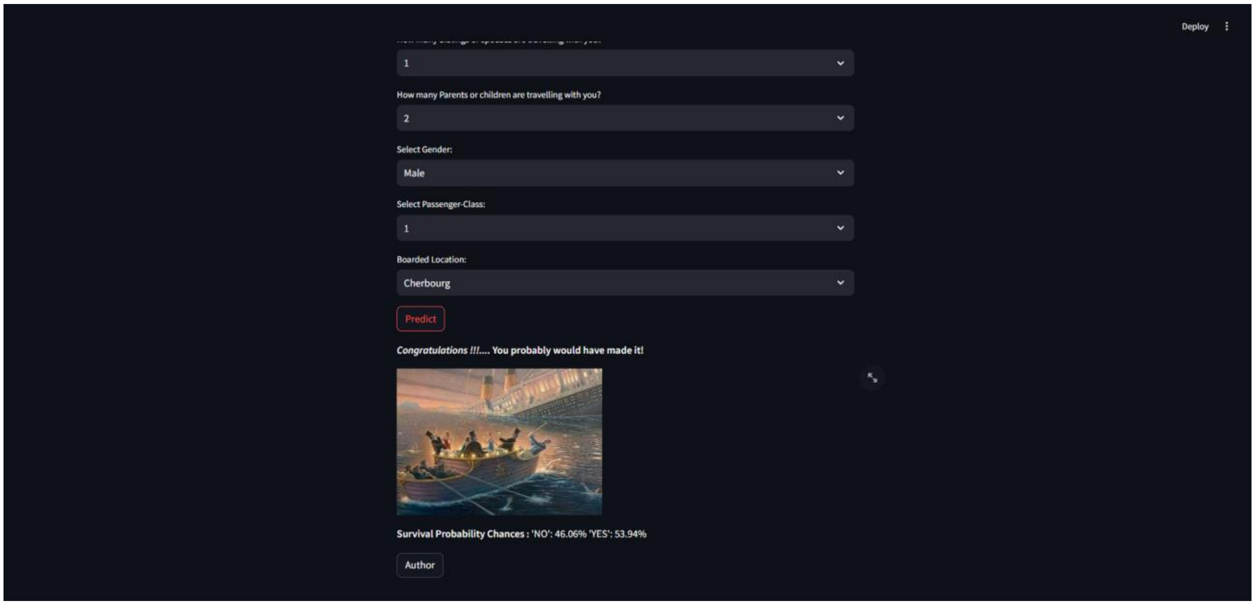
1

Boarded Location:

Southampton

***Better Luck Next time !!!!...*** **You're probably ended up like 'Jack'**

**Survival Probability Chances :** 'NO': 57.31% 'YES': 42.69%



1

How many Parents or children are travelling with you?

2

Select Gender:

Male

Select Passenger-Class:

1

Boarded Location:

Cherbourg

***Congratulations !!!....*** **You probably would have made it!**

**Survival Probability Chances :** 'NO': 46.06% 'YES': 53.94%

# 8. Conclusion and Prospect

This section provides a comprehensive summary of the Titanic Survival Prediction project, reflecting on its achievements and potential future developments.

## 8.1 Summary of the Project

The Titanic Survival Prediction project aimed to develop an interactive web application that uses machine learning to predict the survival chances of Titanic passengers based on various demographic and socio-economic factors. The project was designed and implemented with several key components:

- **Frontend Development**: Utilized Streamlit to create a user-friendly interface, including input fields for passenger details and a display area for survival predictions and probabilities. The intuitive design ensures that users can easily interact with the application and obtain meaningful insights from the predictions.
- **Backend Development**: Implemented using Python, with extensive use of libraries such as Pandas, NumPy, and Scikit-Learn for data processing and machine learning. The backend is responsible for handling user inputs, preprocessing data, and generating predictions using a trained Random Forest Classifier.
- **Model Training and Evaluation**: A Random Forest Classifier was trained on the Titanic dataset, incorporating key features like Passenger Class, Age, Sex, SibSp, Parch, Fare, and Embarked location. The model was rigorously evaluated using accuracy, precision, recall, F1-score, and AUC-ROC metrics to ensure robust performance.
- **Feature Engineering**: Included the creation of dummy variables for categorical features and the imputation of missing values. These steps were crucial for enhancing the predictive power of the model and ensuring accurate and reliable predictions.
- **User Interaction**: The web application allows users to input passenger details and instantly receive a prediction of survival probability. This interactive feature not only makes the application engaging but also educational, providing users with insights into the factors that influenced survival during the Titanic disaster.
- **Visualization**: Integrated visualizations to display the importance of various features and provide users with a better understanding of the model's decision-making process. Bar charts and other graphical representations help in interpreting the results effectively.

Overall, the project successfully demonstrated the integration of machine learning with web development to create a functional and educational tool for predicting Titanic passenger survival.

## 8.2 Prospect

The success of the Titanic Survival Prediction project opens up several avenues for future development and enhancement:

- **Enhanced Model Training**:
  - **Larger Datasets**: Incorporate larger and more diverse datasets to improve the accuracy and robustness of the survival prediction model. This could include additional data from other historical maritime disasters.
  - **Continuous Learning**: Implement continuous learning mechanisms to keep the model updated with new data and advancements in machine learning techniques.
- **Expanded Features**:

    - **Additional Predictors**: Integrate more variables, such as cabin location, deck information, and more granular socio-economic indicators, to refine the predictions.
    - **Survival Factors Analysis**: Provide detailed analysis and reports on how different factors influence survival chances, offering deeper insights into historical patterns.
- **User Personalization**:
    - **User Accounts**: Introduce user accounts to save predictions, track historical data, and receive personalized analyses.
    - **Historical Insights**: Offer users the ability to explore historical data and trends related to the Titanic disaster, enriching their understanding and engagement.
- **Advanced Interactions**:
    - **Voice Assistance**: Integrate voice assistants to guide users through inputting passenger details and understanding prediction results.
    - **Augmented Reality**: Explore the use of augmented reality (AR) to visualize historical data and provide interactive educational experiences.
- **Scalability and Deployment**:
    - **Cloud Integration**: Deploy the application on cloud platforms to enhance scalability, reliability, and accessibility. This would allow for handling more users and larger datasets.
    - **Mobile Application**: Develop mobile versions of the application to reach a broader audience and provide on-the-go access to survival predictions.

By pursuing these future directions, the Titanic Survival Prediction project can continue to evolve, offering more sophisticated features and an even better user experience. The integration of cutting-edge technologies and user feedback will be crucial in shaping the next phases of this innovative project.

## 9. References

This section lists the resources and references that were consulted and used throughout the Titanic Survival Prediction project.

### *Books and Articles:*

- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Chollet, F. (2018). *Deep Learning with Python*. Manning Publications.

### *Websites and Online Resources:*

- Kaggle Titanic Dataset. (n.d.). Retrieved from https://www.kaggle.com/c/titanic/data
- Streamlit. (n.d.). Retrieved from https://streamlit.io/
- Scikit-Learn Documentation. (n.d.). Retrieved from https://scikit-learn.org/stable/
- Pandas Documentation. (n.d.). Retrieved from https://pandas.pydata.org/
- NumPy Documentation. (n.d.). Retrieved from https://numpy.org/
- Matplotlib Documentation. (n.d.). Retrieved from https://matplotlib.org/
- Seaborn Documentation. (n.d.). Retrieved from https://seaborn.pydata.org/

### *Software and Tools:*

- Visual Studio Code. (n.d.). Retrieved from https://code.visualstudio.com/

- Git. (n.d.). Retrieved from https://git-scm.com/
- Python. (n.d.). Retrieved from https://www.python.org/

## 10. Acknowledgments

We would like to express our gratitude to the following individuals and organizations for their support and contributions to the Titanic Survival Prediction project:

*Supervisors and Advisors:*

- **Associate Professor** Liu QiDong, School of Computer and Artificial Intelligence, Zhengzhou University - for their invaluable guidance, feedback, and support throughout the project.

*Report Author:*

- **Mohammad Hamim** - for their contributions to the Whole Project.