🚢 Titanic Survival Prediction

Model Development & Deployment Documentation

---

1. Project Overview

This project focuses on predicting whether a passenger survived the Titanic disaster using machine learning techniques. It demonstrates an **end-to-end data science workflow**, covering data understanding, preprocessing, feature engineering, model training, evaluation, and deployment through a Streamlit web application.

The project is designed to showcase practical data science skills and real-world deployment considerations.

---

2. Dataset Description

The dataset used in this project is the classic **Titanic Dataset**, which contains demographic and travel-related information about passengers aboard the Titanic.

Target Variable:

- **Survived**
  - 0 → Did Not Survive
  - 1 → Survived

Key Features:

- **Pclass** – Passenger class (1st, 2nd, 3rd)
- **Sex** – Gender of the passenger
- **Age** – Passenger age
- **Fare** – Ticket fare
- **Embarked** – Port of embarkation

---

3. Exploratory Data Analysis (EDA)

Exploratory Data Analysis was performed to understand the data distribution, missing values, and relationships between features and survival.

EDA Highlights:

- Analyzed overall survival distribution and class imbalance

- Examined survival patterns across:
  - Passenger class
  - Sex
  - Age
  - Fare
  - Embarkation port
- Identified skewed distributions in Fare
- Investigated relationships between categorical variables and survival outcomes

---

4. Data Cleaning & Feature Engineering

Data preprocessing and feature engineering were applied to improve model performance and interpretability.

Steps Performed:

- Handled missing values:
  - Age filled using median-based strategy
  - Embarked filled using mode
- Removed high-cardinality and identifier columns:
  - PassengerId
  - Name
  - Ticket
  - Cabin
- Created engineered features:
  - **AgeGroup**: Child, Teen, Adult, Senior
  - **FareBin**: Very Low, Low, High, Very High (using quantile-based binning)
  - **IsAlone**: Binary feature indicating whether the passenger traveled alone
- Converted categorical features into model-ready format

---

5. Model Training

Multiple machine learning models were trained and evaluated.

Models Used:

- Logistic Regression (baseline model)
- Random Forest Classifier (final model)

Training Strategy:

- Stratified train-test split to preserve survival ratio
- Hyperparameter tuning using GridSearchCV
- Stratified cross-validation to ensure stable evaluation across folds

6. Model Evaluation

Model performance was evaluated using **ROC-AUC**, which is suitable for classification problems with class imbalance.

Performance Summary:

- Logistic Regression ROC-AUC ≈ 0.85
- Random Forest ROC-AUC ≈ 0.86

The **Random Forest model** demonstrated better generalization and was selected as the final model.

---

7. Streamlit Application Integration

The trained Random Forest model was integrated into an interactive **Streamlit web application**.

Application Features:

- User-friendly input controls (dropdowns and checkboxes)
- Inputs collected:
  - Passenger Class
  - Sex
  - Port of Embarkation
  - Age Group
  - Fare Category
  - Traveling Alone indicator
- Manual dummy-column alignment to match training features
- Display of:
  - Survival prediction
  - Probability score

---

8. Deployment

The Streamlit application was deployed using **Streamlit Cloud**, enabling real-time predictions via a web interface.

Live Application Link:
https://titanicsurvivalprediction-prasadshetty.streamlit.app/

---

9. Complete Project Structure

```
Titanic-Survival-Prediction/
│
├── 📊 TITANIC SURVIVAL PREDICTION.ipynb
│   └── Jupyter Notebook containing:
│       - Exploratory Data Analysis (EDA)
│       - Data Cleaning & Feature Engineering
│       - Model Training & Evaluation
│       - Hyperparameter Tuning using GridSearchCV
│
├── 📄 Titanic-Dataset.csv
│   └── Original Titanic dataset used for analysis and modeling
│
├── 🧠 titanic_model.pkl
│   └── Trained Random Forest model saved using joblib
│
├── 🌐 app.py
│   └── Streamlit application for real-time survival prediction
│
├── 📦 requirements.txt
│   └── Python dependencies required to run the project
│
├── 📝 README.md
│   └── Project overview, instructions, and usage details
│
└── 📘 Titanic_Survival_Prediction_Model_Development_and_Deployment.pdf
    └── Detailed project documentation
```

---

10. Key Learnings

- End-to-end machine learning workflow implementation
- Importance of feature engineering over model complexity
- Handling categorical variables consistently during deployment
- Using stratified sampling and evaluation metrics appropriately
- Debugging real-world production issues
- Deploying machine learning models using Streamlit Cloud

---

11. Conclusion

This project demonstrates a complete machine learning pipeline from data understanding to deployment. It highlights practical data science skills including EDA, feature engineering, model tuning, and production-level deployment. The deployed application allows users to interactively explore survival predictions, making the project both educational and practical.

👤 Author

Durga Prasad
GitHub: https://github.com/shettyprasad-git