Assessment Report

on

"Titanic Survival Prediction"

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE(AI)

By

NAME	ROLL NUMBER	Section
Shreya Mittal	202401100300240	D
Yash Kalra	202401100300284	D
Uday Banduni	202401100300267	D
Tanishka Tyagi	202401100300240	D
Yashika Tyagi	2024001100300286	D

Under the supervision of

Mr. Abhishek Shukla

KIET Group of Institutions, Ghaziabad

May, 2025

1. Introduction

The sinking of the RMS Titanic is one of the most infamous shipwrecks in history. With the increasing accessibility of historical data, data science offers powerful tools to analyze such events. This project focuses on predicting passenger survival using the Titanic dataset, a classic supervised learning problem. By analyzing features such as age, gender, passenger class, and fare, we aim to build a classification model that can effectively predict survival outcomes. The model will be based on the Naive Bayes algorithm, and data preprocessing steps such as cleaning and handling missing values will be integral to improving model performance.

---

2. Problem Statement

To develop a classification model using the Titanic dataset to predict whether a passenger survived the disaster or not. The objective is to use relevant passenger features and apply data cleaning techniques alongside the Naive Bayes algorithm to generate accurate survival predictions. This will demonstrate how machine learning can be applied to real-world historical data for classification tasks.

---

3. Objectives

- Preprocess the Titanic dataset for training a machine learning model.
- Train a **Naive Bayes** classifier to predict passenger survival.
- Evaluate model performance using standard classification metrics.
- Visualize the confusion matrix using a heatmap for better interpretability.

---

4. Methodology

- **Data Collection**: The dataset is extracted from a ZIP archive containing `train.csv`, representing information about passengers aboard the Titanic.
- **Data Preprocessing**:
    - Missing values in numerical columns like `Age` and `Fare` are handled using **median imputation**.
    - Missing values in categorical columns like `Embarked` are filled using the **mode**.
    - Categorical variables (`Sex` and `Embarked`) are encoded using **Label Encoding**.
- **Model Building**:
    - The dataset is split into **80% training** and **20% testing** subsets.
    - A **Gaussian Naive Bayes** classifier is trained using the cleaned dataset.
- **Model Evaluation**:

- Performance is evaluated using **accuracy, precision, recall**, and **F1-score**.
- A **confusion matrix** is generated and visualized using Seaborn heatmap.

---

5. Data Preprocessing

The Titanic dataset is cleaned and prepared using the following steps:

- Numerical columns (`Age`, `Fare`) are converted to numeric types and missing values are filled with the **median**.
- Categorical values in `Embarked` are filled with the **mode**.
- `Sex` and `Embarked` are encoded using **Label Encoding** for compatibility with the model.
- The final dataset is split into training and testing subsets using an 80-20 split ratio.

---

6. Model Implementation

A **Gaussian Naive Bayes** classifier is used due to its efficiency with categorical and continuous features. It assumes independence among features and is particularly useful for binary classification tasks like survival prediction. The model is trained using the processed training data and predictions are made on the test set.

---

7. Evaluation Metrics

The following evaluation metrics are used:

- **Accuracy**: Measures the overall correctness of predictions.
- **Precision**: Measures the proportion of predicted survivors who actually survived.
- **Recall**: Measures the proportion of actual survivors correctly predicted.
- **F1 Score**: Harmonic mean of precision and recall, providing a balanced measure.
- **Confusion Matrix**: Visualized using a heatmap to show the distribution of true positives, true negatives, false positives, and false negatives.

---

8. Results and Analysis

- The Naive Bayes classifier achieved a reasonable **accuracy** on the test data.
- The **classification report** displayed balanced precision and recall for both survival classes.

- The **confusion matrix heatmap** offered insight into the types of classification errors, helping identify whether the model leaned toward false positives or false negatives.

---

9. Code

```
from google.colab import files
uploaded = files.upload()
```

```
Choose Files titanic.zip
• titanic.zip(application/x-zip-compressed) - 34877 bytes, last modified: 5/27/2025 - 100% done
Saving titanic.zip to titanic.zip
```

```
import pandas as pd
import zipfile  # ✅ This is the missing import
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt
```

```
# Open the ZIP file and load train.csv
with zipfile.ZipFile("titanic.zip") as z:
    with z.open("train.csv") as f:
        df = pd.read_csv(f)
```

```
# Data preprocessing
df.replace("\\N", pd.NA, inplace=True)
df["Age"] = pd.to_numeric(df["Age"], errors="coerce")
df["Age"] = df["Age"].fillna(df["Age"].median())
df["Fare"] = pd.to_numeric(df["Fare"], errors="coerce")
df["Fare"] = df["Fare"].fillna(df["Fare"].median())
df["Embarked"] = df["Embarked"].fillna(df["Embarked"].mode()[0])
```

```
# Select features and target
features = ["Pclass", "Sex", "Age", "SibSp", "Parch", "Fare", "Embarked"]
X = df[features].copy()
y = df["Survived"]
```

```
# Encode categorical variables
X["Sex"] = LabelEncoder().fit_transform(X["Sex"])
X["Embarked"] = LabelEncoder().fit_transform(X["Embarked"])
```

```
# Split data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train Naive Bayes model
model = GaussianNB()
model.fit(X_train, y_train)
```

```
GaussianNB
GaussianNB()
```

```
# Predict and evaluate
y_pred = model.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))
```

```
# Confusion matrix visualization
conf_matrix = confusion_matrix(y_test, y_pred)
sns.heatmap(conf_matrix, annot=True, fmt="d", cmap="Blues",
            xticklabels=["Not Survived", "Survived"],
            yticklabels=["Not Survived", "Survived"])
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.title("Confusion Matrix")
plt.show()
```

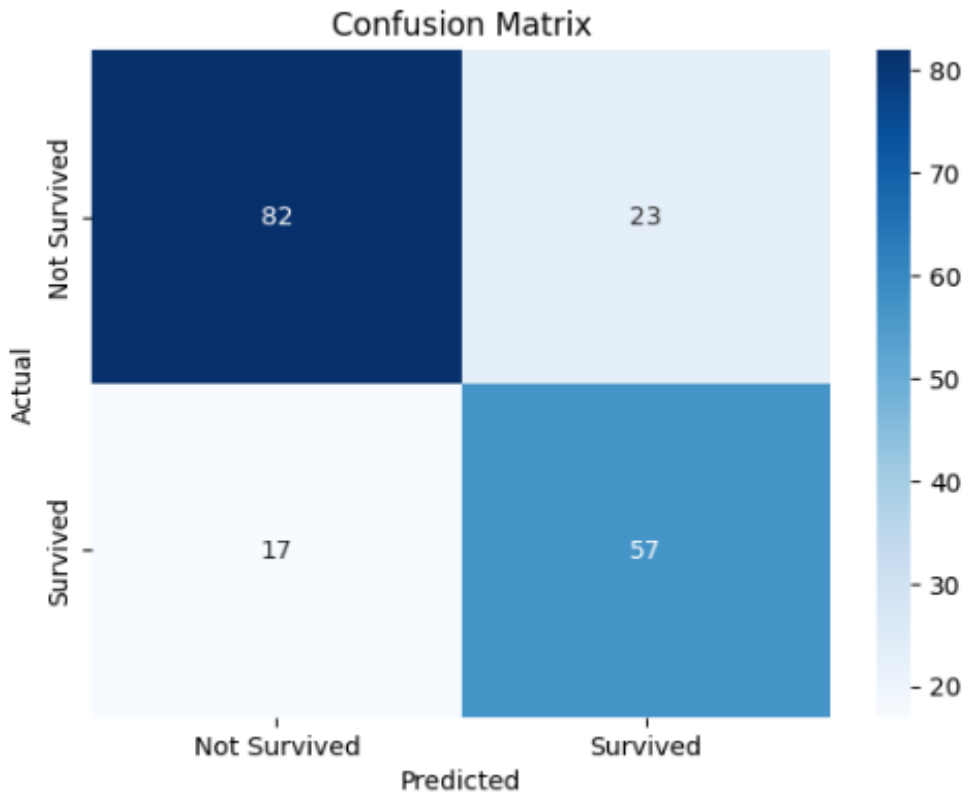
10. Output

```
Accuracy: 0.776536312849162

Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.78      0.80       105
           1       0.71      0.77      0.74        74

    accuracy                           0.78       179
   macro avg       0.77      0.78      0.77       179
weighted avg       0.78      0.78      0.78       179
```



11. Conclusion

The Naive Bayes model successfully classified passenger survival with satisfactory performance. This project illustrates how simple models can effectively solve binary classification problems with proper data preprocessing. For improved performance, future work may involve:

- Trying advanced models like **Logistic Regression, Random Forests**, or **XGBoost**
- Handling **class imbalance**
- Engineering new features (e.g., family size, title extraction from names)

---

12. References

- scikit-learn documentation
- pandas documentation
- Seaborn visualization library
- Research on machine learning applications in Titanic survival prediction

---