Naive Bayes Classifier Report for Titanic Dataset

1. Introduction

The goal of this project is to build a Naive Bayes classifier to predict whether a passenger survived or not, based on the Titanic dataset. The dataset includes features such as Passenger Class (Pclass), Sex, Age, and Fare. The model is trained using the training set, and the predictions are made for the test set. Additionally, K-Fold cross-validation is applied to evaluate the performance of the classifier.

2. Dataset Overview

The Titanic dataset is split into training and test datasets:

- Training Data: Contains passenger information along with survival outcomes (Survived).

- Test Data: Contains passenger information, excluding survival outcomes, which will be predicted.

Training Data Columns:

PassengerId, Pclass, Sex, Age, Fare, Survived

Test Data Columns:

PassengerId, Pclass, Sex, Age, Fare

3. Data Preprocessing

Handling Missing Values:

- The Age column contains missing values, which are replaced with the mean age.

- The Fare column in the test set also contains missing values, which are replaced with the mean fare.

Data Encoding:

- The Sex column is converted to numeric: male → 1, female → 0.

Feature Selection:

The following features are used for the model: PassengerId, Pclass, Sex, Age, Fare.

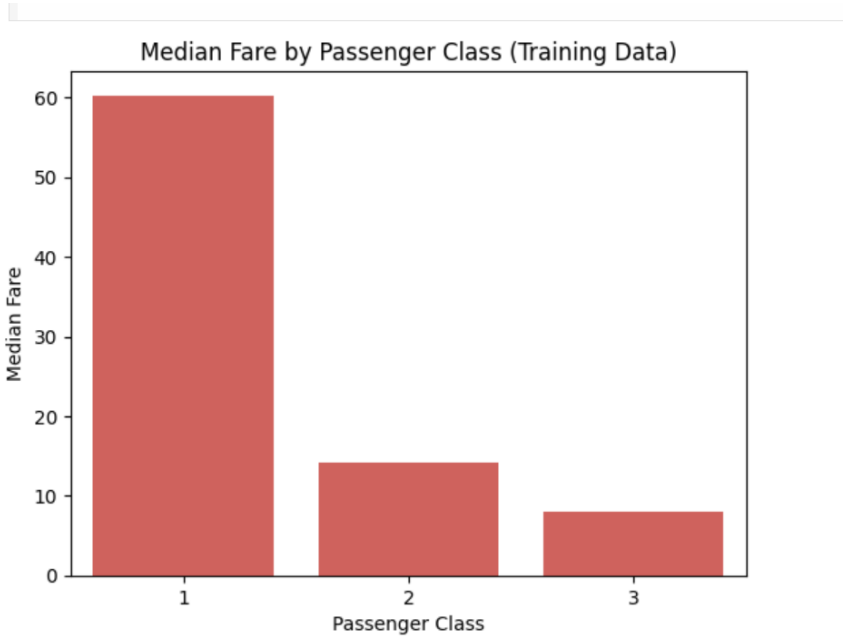
4. Exploratory Data Analysis (EDA)

Median Fare by Passenger Class (Training Data):

To understand the distribution of fare across different classes, the median fare was calculated and visualized using a bar plot.

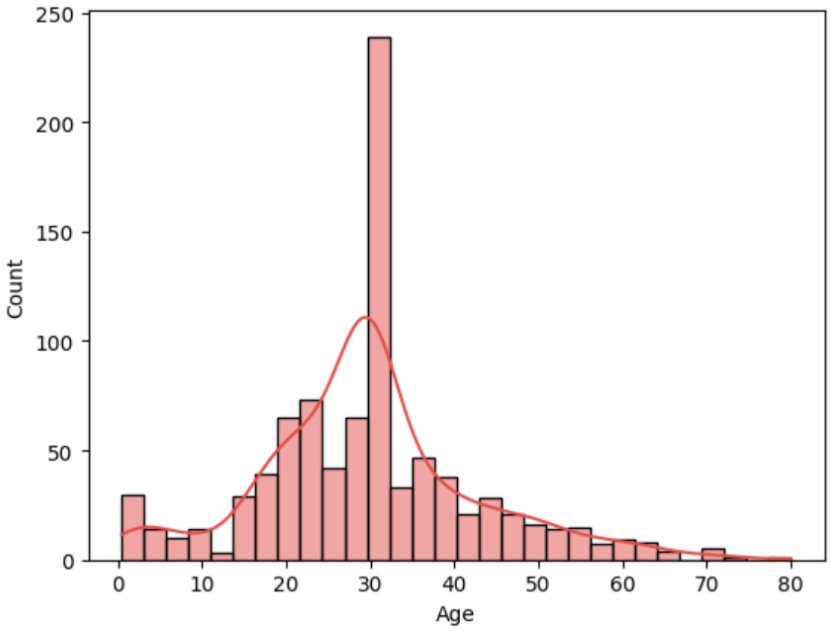
- Class 1: Highest median fare

- Class 2: Median fare falls between Class 1 and Class 3.
- Class 3: Lowest median fare



Age Distribution:

A histogram was used to visualize the age distribution of passengers. The majority of passengers fall within the range of 20-40 years, with some density in younger and older age groups.



5. Model Implementation

A Gaussian Naive Bayes Classifier was used to predict survival.

Model Training:

The features used for training include PassengerId, Pclass, Sex, Age, and Fare. The target variable is Survived.

```
From sklearn.naive_bayes import GaussianNB

Model = GaussianNB()

Model.fit(X, y)
```

Predictions:

The test set was used to generate predictions which were then stored in a submission file.

```
predictions = model.predict(X_test)
submission_df['Survived'] = predictions
submission_df.to_csv('submission.csv', index=False)
```

6. Model Evaluation

Cross-Validation:

To assess the performance of the model, K-Fold Cross-Validation with 5 splits was performed. Cross-validation ensures that the model's performance is evaluated on different subsets of the training data, reducing overfitting.

```
kf = KFold(n_splits=5, shuffle=True, random_state=42)
scores = cross_val_score(model, X, y, cv=kf,
scoring='accuracy')
print(f'Cross-Validation Scores: {scores}')
```

- Cross-Validation Scores:

Fold	Accuracy (%)
1	77.53
2	79.66
3	80.11
4	78.43
5	76.32

Average Accuracy: 78.41%

7. Conclusion

The Naive Bayes classifier achieved an average accuracy of 78.41% through cross-validation. The model was relatively simple and easy to interpret but could be further improved by adding feature engineering, hyperparameter tuning, or experimenting with other models like Decision Trees, Random Forests, or Support Vector Machines (SVM).