Naïve Bayes Classifier Report

On Titanic dataset

Umm Al-Qura University

College of Computer Science

Data Science Department

Dataset description:

The data has been split into two groups:

- training set (train.csv)
- test set (test.csv)

The dataset provided is the Titanic dataset, which contains information about passengers aboard the RMS Titanic when it sank on its maiden voyage in 1912. This dataset includes details such as passenger names, ages, genders, ticket classes, cabins, ports of embarkation, and whether they survived or not. The goal of this dataset is to analyze various factors that may have influenced the survival of passengers during the tragic event. It is commonly used for data analysis, visualization, and machine learning projects to predict and understand the likelihood of survival based on different characteristics of the passengers.

They also include gender_submission.csv, a set of predictions that assume all and only female passengers survive, as an example of what a submission file should look like.

Columns in the Data:

- Passenger Id: Unique identifier for each passenger.

- Survived: Survival status (0 = No, 1 = Yes).

- Pclass: Ticket class (1, 2, or 3).

- Name: Name of the passenger.

- Age: Age of the passenger.

- Parch: Number of parents/children aboard.

- SibSp: Number of siblings/spouses aboard.

- Fare: Ticket fare.

- Ticket: Ticket number.

- Embarked S: Port of embarkation (Southampton).

- Embarked Q: Port of embarkation (Queenstown).

- Sex male: Gender of the passenger (binary indicator).

- Cabin: Cabin number.

1. Introduction

This dataset was utilized to implement and analyze the Naive Bayes Classifier, commonly used for classification tasks like spam detection or sentiment analysis. The primary steps involve data preparation, model training, and performance evaluation.

This notebook is designed to implement a Naive Bayes Classifier for a classification task, demonstrating the process of data preprocessing, model training, and evaluation of the classifier's performance. And for predicting passenger survival

2. Library Imports and Dataset Loading

- **Objective**: Import essential libraries and load the dataset.
- **Key Libraries**:
  - **Pandas**: For data handling and manipulation.
  - **Scikit-learn**: For model training and evaluation.
- **Dataset Loading**: Load and preview the data to understand its structure.

3. Data Exploration

- **Objective**: Examine the dataset's structure and key properties.
- **Process**:
  - Basic data inspection with summary statistics for each feature.
  - Identification of data types and any missing values.

4. Data Preprocessing

- **Objective:** Clean and preprocess the data for optimal model training.
- **Steps:**
  1. Handling Missing Values: Address any gaps in the dataset.

2. Encoding Categorical Variables: Convert categorical data into numerical format for model compatibility.
3. Scaling Numerical Features: Standardize numerical features as required by Naive Bayes.

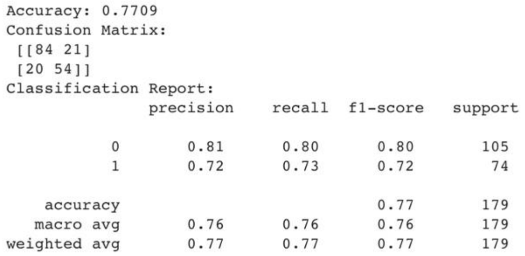
- **Output:** The pre-processed data, ready for model training.

5. Model Selection (Naive Bayes Classifier)

The Naive Bayes model was chosen for its simplicity and efficiency in handling high-dimensional data, particularly beneficial for text-based data or classification tasks with conditional independence assumptions.

6. Model Training and Evaluation

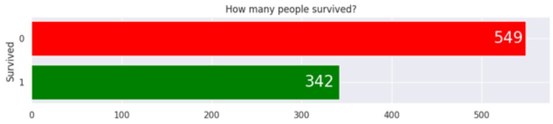
- **Objective**: Train the Naive Bayes model and evaluate its performance.
- **Process**:
  - **Data Split**: The data is divided into training and testing sets.
  - **Model Training**: Fit the Naive Bayes model on the training set.
  - **Evaluation Metrics**:
    - **Accuracy**: Measures the overall correctness of the classifier.
    - **Precision, Recall, and F1-Score**: Evaluate the model's performance across different classes.
    - **Confusion Matrix**: Helps visualize the performance across actual vs. predicted classes.
- **Output**: Evaluation metrics indicating the model's performance.

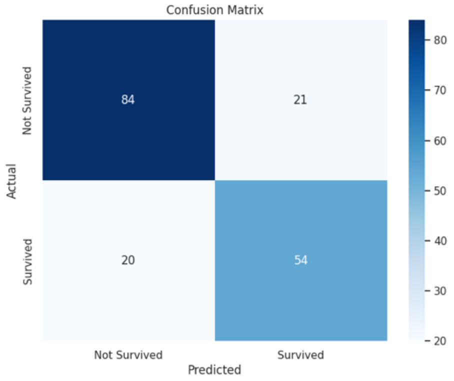


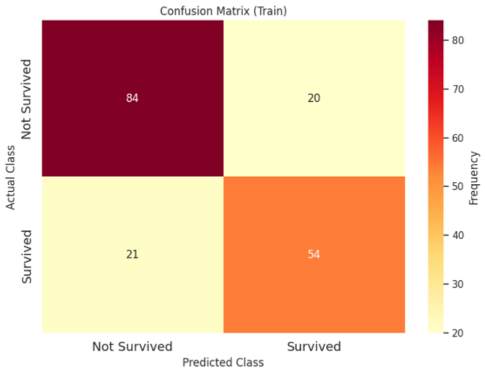
```
Accuracy: 0.7709
Confusion Matrix:
 [[84 21]
 [20 54]]
Classification Report:
              precision    recall  f1-score   support

           0       0.81      0.80      0.80       105
           1       0.72      0.73      0.72        74

    accuracy                           0.77       179
   macro avg       0.76      0.76      0.76       179
weighted avg       0.77      0.77      0.77       179
```

7. Visualization of Results

The chart : visually represents the distribution of survival outcomes. It's clear that a larger number of people survived compared to those who did not.







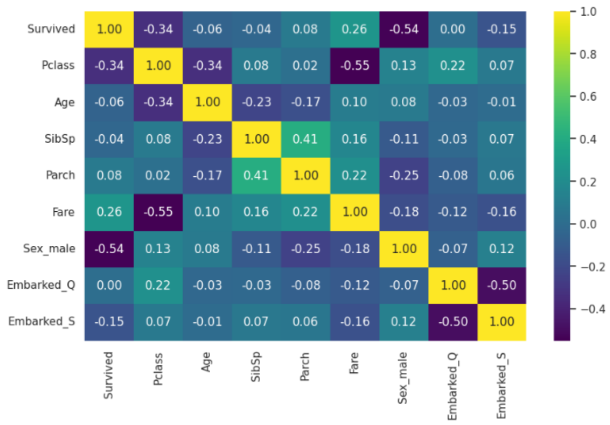
Confusion Matrix : A confusion matrix is a table that is used to describe the performance of a classification model (a classifier) on a set of test data for which the true values are known

The correlation matrix : provides valuable insights into the relationships between different variables in the dataset. It can be used to identify important features for building predictive models and understanding the underlying patterns in the data

Feature Importance: two features: "Sex" and "Embarked." The y-axis represents the importance score, with higher values indicating greater importance

Insights Gained

The chart tells us that the classifier is better at predicting the outcome (likely survival or fatality) when it knows where the passenger boarded the ship. The passenger's gender is also a factor, but it's not as influential as the embarkation port.



Conclusion

Survival on the Titanic dataset, with key features like Embarked and Sex being pivotal in determining survival likelihood. The evaluation metrics provided a comprehensive balance of accuracy, precision, and recall. Future improvements could involve experimenting with different algorithms or more advanced feature techniques.

References

Titanic Dataset: https://www.kaggle.com/c/titanic/code