Naive Bayes Classification

Introduction

Heart disease remains one of the leading causes of mortality worldwide, making early detection and prevention vital for effective treatment and management. In this project, we leverage machine learning techniques to predict the presence of heart disease based on various patient features. Using the “Predicting Heart Disease” dataset, we aim to develop classification models that can accurately identify whether a patient is likely to have heart disease.

The dataset comprises multiple attributes related to patient demographics, clinical measurements, and other risk factors. By utilizing algorithms such as Decision Tree and Gaussian Naive Bayes, we explore the relationship between these features and the likelihood of heart disease, ultimately striving to enhance diagnostic accuracy and support clinical decision-making.

Data Preprocessing and Exploration

2.1 Loading the Data

The dataset was loaded from a CSV file using the Pandas library. After loading, the structure and basic information about the dataset were displayed to understand its contents, including the number of records and available features.

2.2 Dropping Unnecessary Columns

The dataset contained columns that were not useful for the analysis, such as “Unnamed: 32” and “id.” These columns were removed to clean the dataset and improve its accuracy for modeling.

2.3 Visualization of Class Distribution

We used Seaborn to visualize the class distribution of the target variable (the presence of heart disease), which contains two categories:

. **presence of heart disease (1)**

. **pabsence of heart disease (0)**.

This plot shows the number of instances for each class, helping us understand the balance of the dataset between the different classes.

2.4 Mapping Diagnosis to Numerical Values

To facilitate the use of the target variable for training the model, categorical values were mapped to numerical labels:

- **Presence of heart disease** was assigned a value of **1** to represent the positive class.
- **Absence of heart disease** was assigned a value of **0** to represent the negative class.

Feature Selection

3.1 Correlation Analysis

We calculated the correlation between each feature and the target variable (the presence of heart disease). This analysis helped us understand which features are most strongly associated with the likelihood of heart disease, allowing us to focus on the most relevant attributes for our classification models.

3.2 Dropping Less Important Features

Based on the correlation analysis, some features that exhibited very low correlation with the target variable were dropped. These features included:

- **symmetry_se**
- **fractal_dimension_mean**
- **texture_se**
- **compactness_se**, among others.

This step reduced the dimensionality of the dataset and simplified the models without sacrificing performance, making the training process more efficient and potentially enhancing model interpretability.

Model Building and Evaluation

4.1 Data Splitting

The dataset was split into training and testing sets, with 80% of the data allocated for training and 20% for testing. This split ensures that the model can be evaluated on unseen data, providing a robust assessment of its performance.

4.2 Training the Decision Tree and Naive Bayes Models

We initialized both the Decision Tree and Gaussian Naive Bayes models. Each model was trained using the training data (X_train, y_train). This allows us to compare the performance of the two algorithms on the same dataset.

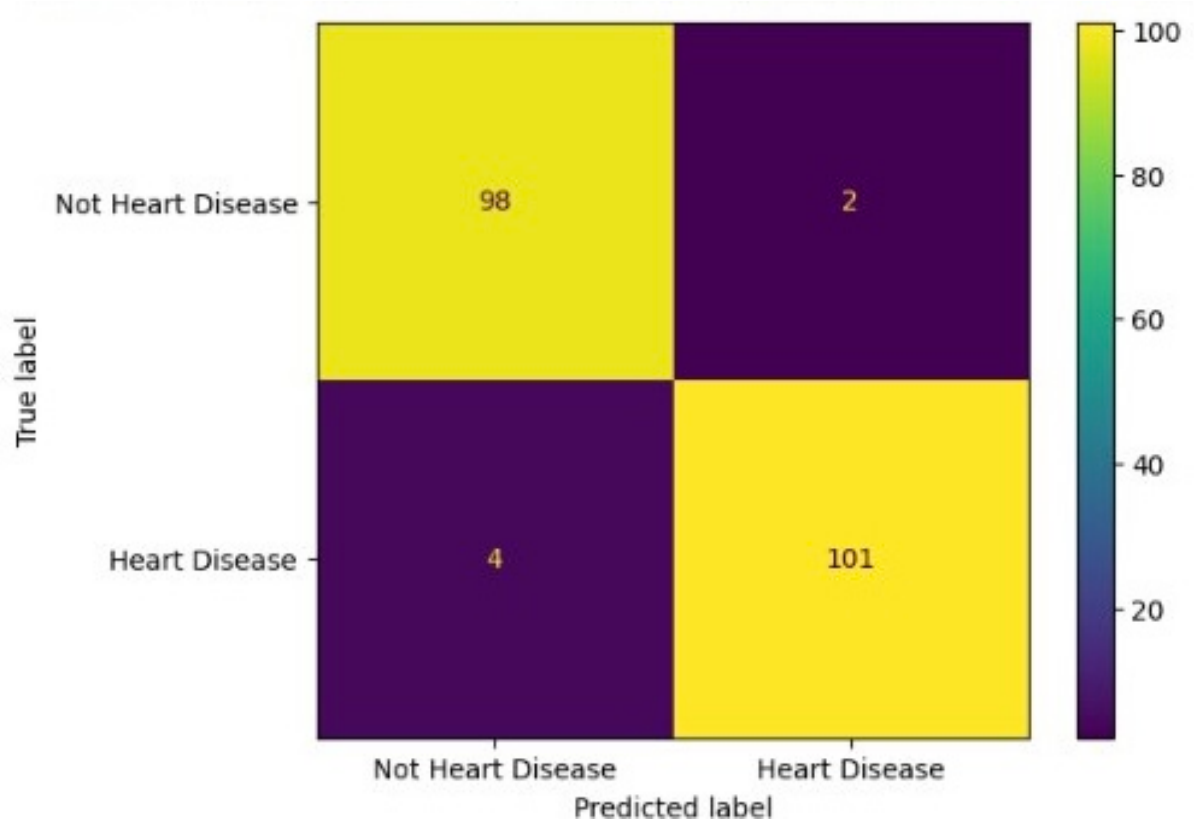
4.3 Making Predictions

Both models were used to make predictions on the test set (X_test). The predictions from each model were compared to the actual values of the target variable to assess their accuracy.

4.4 Model Evaluation

The models were evaluated using several metrics, including:

- **Accuracy**: The Decision Tree model achieved an accuracy of approximately **97**%, while the Gaussian Naive Bayes model achieved an accuracy of **63**%.



. **Confusion Matrix**: Displays the counts of true positives, true negatives, false positives, and false negatives.

4.5 Confusion Matrix Visualization

The confusion matrix was visualized to provide a clearer understanding of the model's performance in terms of true positives, true negatives, false positives, and false negatives. This visualization helps to identify how well the model classifies instances of heart disease and allows for a quick assessment of its strengths and weaknesses in predictions.

Conclusion

In this specific project, the Decision Tree classifier outperformed the Naive Bayes classifier, achieving an accuracy of **97%** compared to **68%** for Naive Bayes. This indicates that Decision Trees may be better suited for the heart disease dataset as they can capture more complex patterns and interactions between features.

However, the choice between Naive Bayes and Decision Tree classifiers depends on the specific context and requirements of the problem at hand. Naive Bayes may be preferable for larger datasets with more categorical features, while Decision Trees provide better interpretability and performance on complex datasets. In practice, it may be beneficial to experiment with both algorithms and possibly ensemble methods to achieve the best results.