Collage of computing

Data Analysis 2: DS3114

Project report

Task 1: Naïve bayes

Instructor: Omaima Fallatah

Student Name	ID
Mayar Turki Al-Owaydhi	443003550
Fajr faisal Al-Zahrani	44410657

Introduction

Heart disease continues to be a major global health concern, making early prediction and risk assessment essential for effective prevention and treatment. In this study, we analyze the **Heart Disease Dataset** from Kaggle, which comprises 1025 patient records and 14 health-related variables, including age, cholesterol levels, and exercise-induced angina. The primary goals of this analysis are to predict the likelihood of heart disease using machine learning models and to identify key risk factors that contribute to its development.

Through the use of Exploratory Data Analysis (EDA) and the application of machine learning models like Gaussian Naive Bayes and Logistic Regression, this study aims to uncover significant insights about heart disease predictors and evaluate the effectiveness of these models in classification tasks.

Dataset

Heart Disease Dataset:
https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset

Objective

- Predict the likelihood of heart disease based on various patient health metrics
- identify the most significant risk factors contributing to heart disease.

Exploration the dataset (EDA)

The df.head() function is commonly used in data analysis to quickly inspect the contents of a DataFrame and display first 5 rows, The output of the df.head() will be:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	0
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
1020	59	1	1	140	221	0	1	164	1	0.0	2	0	2	1
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3	0
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2	0
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2	1
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3	0

The df.tail() function is commonly used in data analysis to quickly inspect the contents of a DataFrame and display last 5 rows, The output of the df.tail() will be:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1025 entries, 0 to 1024
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1025 non-null   int64
 1   sex       1025 non-null   int64
 2   cp        1025 non-null   int64
 3   trestbps  1025 non-null   int64
 4   chol      1025 non-null   int64
 5   fbs       1025 non-null   int64
 6   restecg   1025 non-null   int64
 7   thalach   1025 non-null   int64
 8   exang     1025 non-null   int64
 9   oldpeak   1025 non-null   float64
 10  slope     1025 non-null   int64
 11  ca        1025 non-null   int64
 12  thal      1025 non-null   int64
 13  target    1025 non-null   int64
dtypes: float64(1), int64(13)
memory usage: 112.2 KB
```

The output of df.info() function provides a concise summary of a DataFrame's structure, including the number of entries, column names, non-null counts, data types, and memory usage, we have 14 columns, 1025 rows and 2 data types: int, float.

df.isna() function to find the massing value, and we don’t have a massing value in our dataset

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1020	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1021	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1022	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1023	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1024	False	False	False	False	False	False	False	False	False	False	False	False	False	False

1025 rows × 14 columns

```
(1025, 14)
```

df.shape() function to display the number of columns and rows, we have 1025 rows and 14 columns.

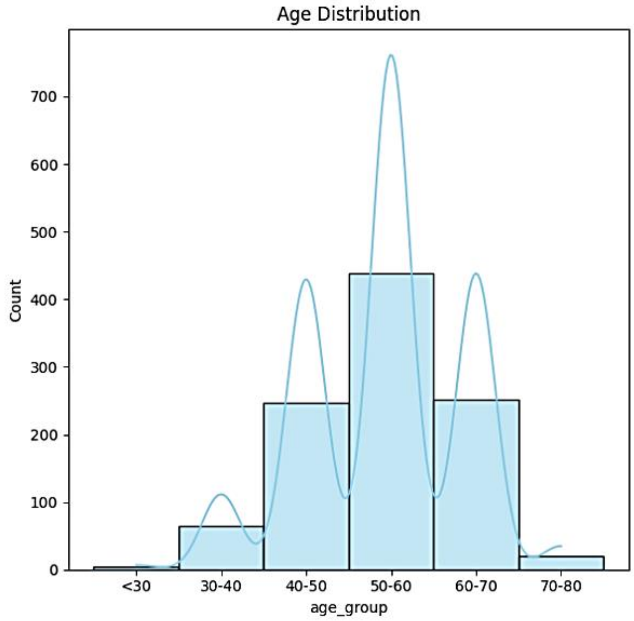
	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.00000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000
mean	54.434146	0.695610	0.942439	131.611707	246.00000	0.149268	0.529756	149.114146	0.336585	1.071512	1.385366	0.754146	2.323902	0.513171
std	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.527878	23.005724	0.472772	1.175053	0.617755	1.030798	0.620660	0.500070
min	29.000000	0.000000	0.000000	94.000000	126.00000	0.000000	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	48.000000	0.000000	0.000000	120.000000	211.00000	0.000000	0.000000	132.000000	0.000000	0.000000	1.000000	0.000000	2.000000	0.000000
50%	56.000000	1.000000	1.000000	130.000000	240.00000	0.000000	1.000000	152.000000	0.000000	0.800000	1.000000	0.000000	2.000000	1.000000
75%	61.000000	1.000000	2.000000	140.000000	275.00000	0.000000	1.000000	166.000000	1.000000	1.800000	2.000000	1.000000	3.000000	1.000000
max	77.000000	1.000000	3.000000	200.000000	564.00000	1.000000	2.000000	202.000000	1.000000	6.200000	2.000000	4.000000	3.000000	1.000000

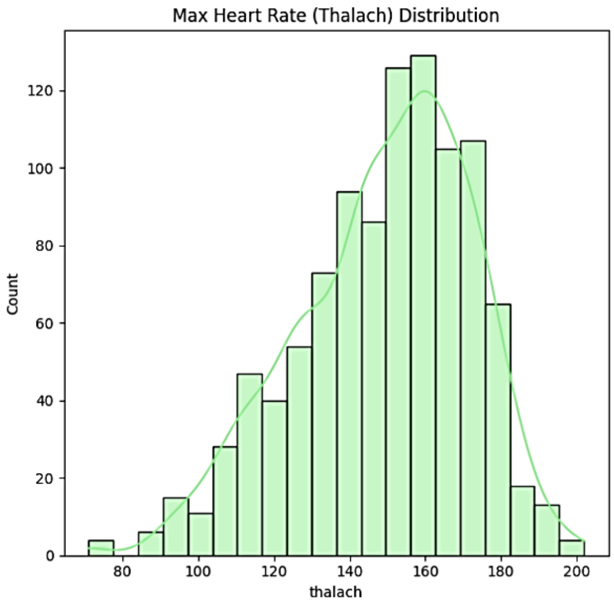
The df.describe() function used for summary statistics and insights about a dataset, helping to identify feature distributions, data types, missing values, and informing preprocessing steps for Naive Bayes classification.

```
0       50-60
1       50-60
2       60-70
3       60-70
4       60-70
        ...
1020    50-60
1021    50-60
1022    40-50
1023    40-50
1024    50-60
Name: age_group, Length: 1025, dtype: category
Categories (7, object): ['<30' < '30-40' < '40-50' < '50-60' < '60-70' < '70-80' < '80+']
```

There effectively segments the continuous age data in the **age** column of the DataFrame into discrete age groups, The new **age_group** column provides categorical data.

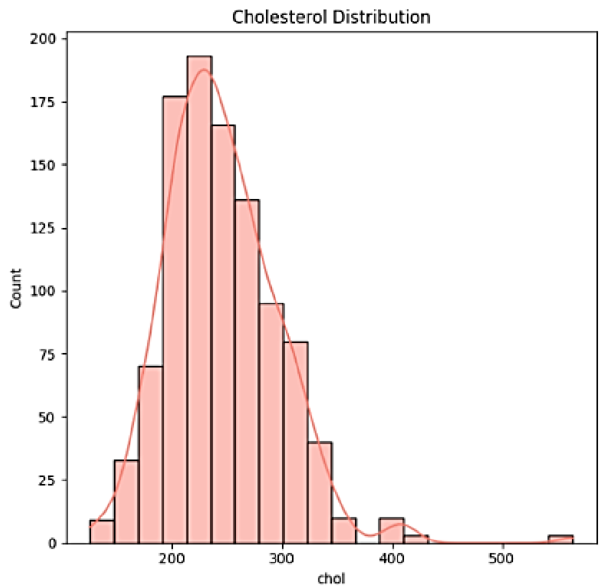
Age Distribution: The age group of the patients, with a majority falling in the 50-60 age range.





Max Heart Rate Distribution (Thalach): Most patients have a max heart rate between 140-160.

Cholesterol Distribution (Chol): Cholesterol levels show a distribution centered around 200-300, with a small tail of higher values.



```
              age       sex        cp  trestbps      chol       fbs  \
age       1.000000 -0.103240 -0.071966  0.271121  0.219823  0.121243
sex      -0.103240  1.000000 -0.041119 -0.078974 -0.198258  0.027200
cp       -0.071966 -0.041119  1.000000  0.038177 -0.081641  0.079294
trestbps  0.271121 -0.078974  0.038177  1.000000  0.127977  0.181767
chol      0.219823 -0.198258 -0.081641  0.127977  1.000000  0.026917
fbs       0.121243  0.027200  0.079294  0.181767  0.026917  1.000000
restecg  -0.132696 -0.055117  0.043581 -0.123794 -0.147410 -0.104051
thalach  -0.390227 -0.049365  0.306839 -0.039264 -0.021772 -0.008866
exang     0.088163  0.139157 -0.401513  0.061197  0.067382  0.049261
oldpeak   0.208137  0.084687 -0.174733  0.187434  0.064880  0.010859
slope    -0.169105 -0.026666  0.131633 -0.120445 -0.014248 -0.061902
ca        0.271551  0.111729 -0.176206  0.104554  0.074259  0.137156
thal      0.072297  0.198424 -0.163341  0.059276  0.100244 -0.042177
target   -0.229324 -0.279501  0.434854 -0.138772 -0.099966 -0.041164

           restecg   thalach     exang   oldpeak     slope        ca  \
age      -0.132696 -0.390227  0.088163  0.208137 -0.169105  0.271551
sex      -0.055117 -0.049365  0.139157  0.084687 -0.026666  0.111729
cp        0.043581  0.306839 -0.401513 -0.174733  0.131633 -0.176206
trestbps -0.123794 -0.039264  0.061197  0.187434 -0.120445  0.104554
chol     -0.147410 -0.021772  0.067382  0.064880 -0.014248  0.074259
fbs      -0.104051 -0.008866  0.049261  0.010859 -0.061902  0.137156
restecg   1.000000  0.048411 -0.065606 -0.050114  0.086086 -0.078072
thalach   0.048411  1.000000 -0.380281 -0.349796  0.395308 -0.207888
exang    -0.065606 -0.380281  1.000000  0.310844 -0.267335  0.107849
oldpeak  -0.050114 -0.349796  0.310844  1.000000 -0.575189  0.221816
slope     0.086086  0.395308 -0.267335 -0.575189  1.000000 -0.073440
ca       -0.078072 -0.207888  0.107849  0.221816 -0.073440  1.000000
thal     -0.020504 -0.098068  0.197201  0.202672 -0.094090  0.149014
target    0.134468  0.422895 -0.438029 -0.438441  0.345512 -0.382085

              thal    target
age       0.072297 -0.229324
sex       0.198424 -0.279501
cp       -0.163341  0.434854
trestbps  0.059276 -0.138772
chol      0.100244 -0.099966
fbs      -0.042177 -0.041164
restecg  -0.020504  0.134468
thalach  -0.098068  0.422895
exang     0.197201 -0.438029
oldpeak   0.202672 -0.438441
slope    -0.094090  0.345512
ca        0.149014 -0.382085
thal      1.000000 -0.337838
target   -0.337838  1.000000
```

Negative Correlations:

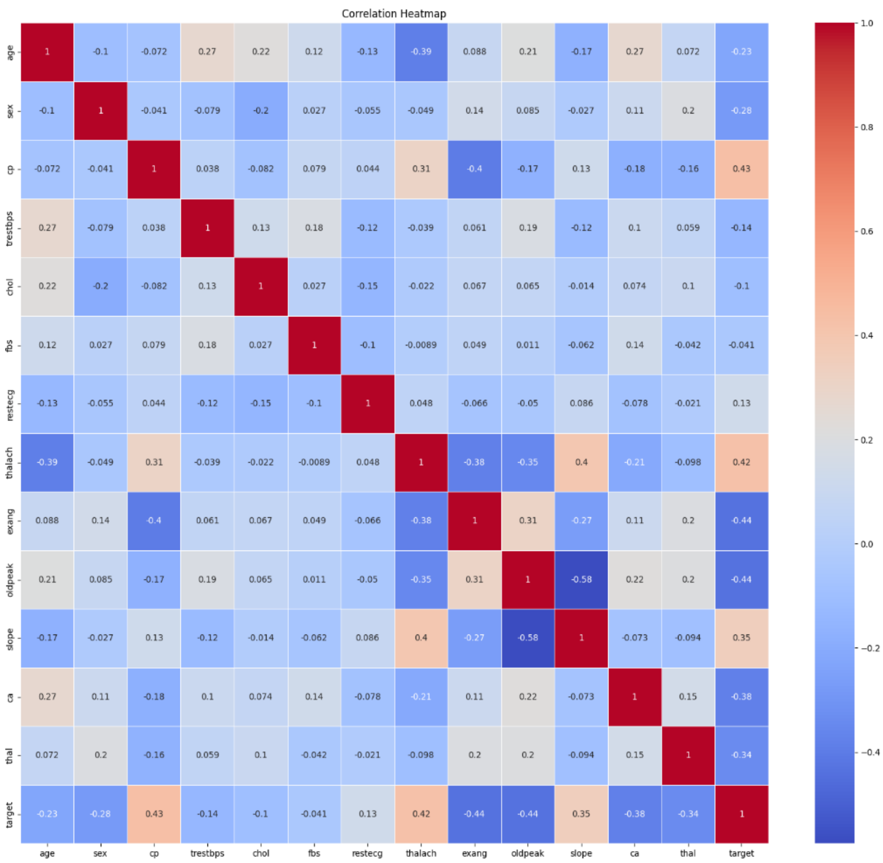
- **Age and Max Heart Rate (Thalach):** Older age is associated with lower maximum heart rates (correlation of -0.39).
- **Exercise-Induced Angina (Exang) and Heart Disease (Target):** Higher levels of angina correlate with a lower likelihood of heart disease (correlation of -0.44).
- **Oldpeak and Heart Disease (Target):** Higher oldpeak values (indicating more ischemia) are associated with a lower likelihood of heart disease (correlation of -0.44).

Positive Correlations:

- **Chest Pain Type (cp) and Heart Disease (Target):** Certain types of chest pain are positively associated with a higher likelihood of heart disease (correlation of 0.43).
- **Number of Major Vessels (ca) and Heart Disease (Target):** A positive correlation (0.15) suggest

- Red cells indicate a strong positive correlation between two variables.
- Blue cells represent a negative correlation.

The diagonal values are all 1 since a variable is perfectly correlated with itself.



Gaussian naïve bayes (GNB)

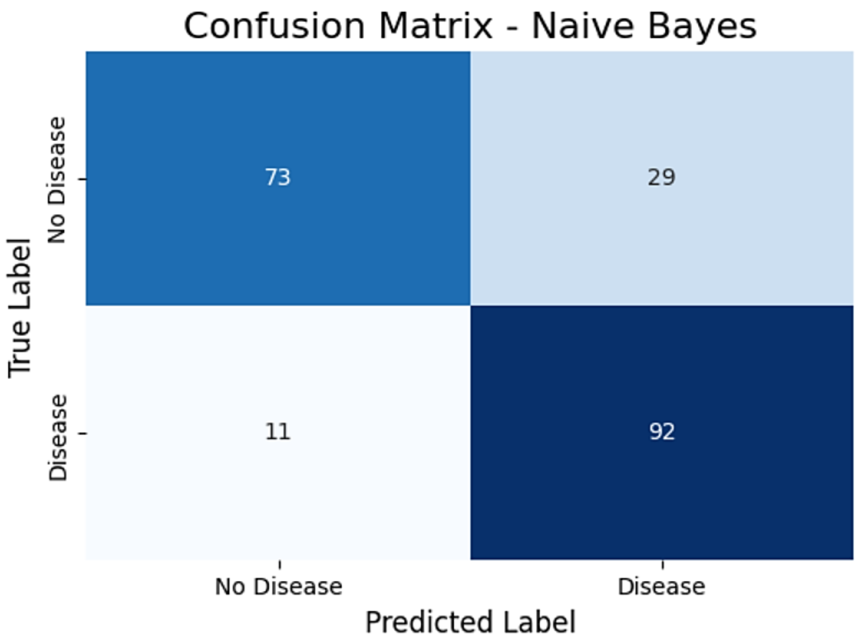
```
0.8048780487804879
              precision    recall  f1-score   support

           0       0.87      0.72      0.78       102
           1       0.76      0.89      0.82       103

    accuracy                           0.80       205
   macro avg       0.81      0.80      0.80       205
weighted avg       0.81      0.80      0.80       205
```

The Gaussian Naive Bayes model achieved approximately 80.5% accuracy, performing well with high precision (0.87) for Class 0 but slightly lower recall (0.72). For Class 1, it showed strong recall (0.89) and a good F1-score (0.82). Overall, the macro and weighted averages (both around 0.80) indicate balanced performance across both classes, suggesting the model is effective for the classification task.

```
Confusion Matrix:
[[73 29]
 [11 92]]
```



- True Positives (Bottom-right, 92): Correctly predicted "Disease."
- True Negatives (Top-left, 73): Correctly predicted "No Disease."
- False Positives (Top-right, 29): Incorrectly predicted "Disease."
- False Negatives (Bottom-left, 11): Incorrectly predicted "No Disease."

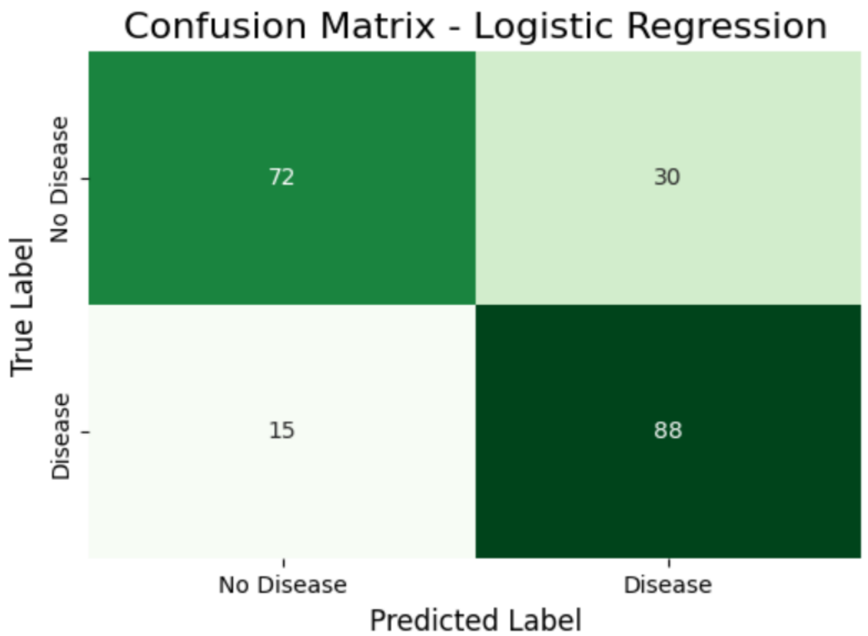
Logistic regression (LR)

```
0.7804878048780488
              precision    recall  f1-score   support

           0       0.83      0.71      0.76       102
           1       0.75      0.85      0.80       103

    accuracy                           0.78       205
   macro avg       0.79      0.78      0.78       205
weighted avg       0.79      0.78      0.78       205
```

The Logistic Regression model achieved approximately 78.1% accuracy. It showed high precision for Class 0 (0.83) but lower recall (0.71), while Class 1 had strong recall (0.85) and a good F1-score (0.80). The macro and weighted averages around 0.78 indicate balanced performance across both classes, suggesting the model is effective for the classification task.



- True Positives (Bottom-right, 88): correctly predicted "Disease" when the actual label was "Disease."
- True Negatives (Top-left, 72): correctly predicted "No Disease" when the actual label was "No Disease."
- False Positives (Top-right, 30): incorrectly predicted "Disease" when the actual label was "No Disease."
- False Negatives (Bottom-left, 15): incorrectly predicted "No Disease" when the actual label was "Disease."

Conclusion

In conclusion, this study successfully employed both Gaussian Naive Bayes and Logistic Regression models to predict the likelihood of heart disease. The Gaussian Naive Bayes model achieved a slightly higher accuracy (80.5%) compared to Logistic Regression (78.1%). Each model displayed strengths in different areas, with Gaussian Naive Bayes excelling in precision for "No Disease" predictions and Logistic Regression performing well in recall for "Disease" predictions. The analysis also highlighted important correlations, such as the negative association between age and maximum heart rate, and positive correlations between certain chest pain types and heart disease. These findings contribute to a better understanding of heart disease risk factors and support the use of machine learning in healthcare for early diagnosis and prevention strategies.