Date of Submission July 2024

Digital Object Identifier Not Applicable

A Study on Implementation of Naive Bayes Classifiers

NIMESH GOPAL PRADHAN[1], RAMAN BHATTARAI[1]
[1]Department of Electronics and Computer Engineering, Thapathali Campus, Tribhuvan University, Kathmandu, Nepal

ABSTRACT Naive Bayes classifiers are widely employed in ML for their simplicity and effectiveness in classification tasks. Based on Bayes' theorem, these models assume independence among features given the class label, making computations manageable even with large datasets. Categorical Naive Bayes extends this framework to handle categorical features by calculating probabilities using frequency counts within each class. Conversely, Gaussian Naive Bayes assumes continuous features follow a Gaussian distribution, using mean and variance estimates to compute probabilities. This paper also introduces a Hybrid classifier which combines both Gaussian and Categorical Naive Bayes and compare their performance in classification task. Naive Bayes classifiers also have the advantage of computational efficiency and easy implementation, making them suitable for real-time applications and large-scale data processing tasks.

INDEX TERMS Categorical naive bayes, classification, gaussian naive bayes, variance

I. INTRODUCTION

Naive Bayes is a foundational probabilistic classifier rooted in Bayes' theorem, renowned for its simplicity, efficiency, and versatility across a wide range of machine learning applications. The algorithm leverages the assumption of feature independence conditioned on the class label, allowing it to compute probabilities swiftly, even with large and high-dimensional datasets. This characteristic makes Naive Bayes particularly well-suited for tasks where rapid classification and scalable performance are paramount. Despite its "naive" assumption of independence, the classifier often delivers competitive results, particularly when this assumption aligns closely with the data structure or when the dataset is sufficiently large to mitigate the impact of violations. Moreover, Naive Bayes' straightforward implementation and low computational overhead make it an attractive choice for real-time applications and settings where interpretability and speed are crucial. Its robust performance across diverse domains and its ability to handle mixed data types, including both categorical and continuous features, underscore its enduring relevance and widespread adoption in both academic research and industrial contexts alike.

There are several types of Naive Bayes classifiers, each of which are tailored to handle different kinds of data. The most common types of Naive Bayes classifiers include Gaussian Naive Bayes, Multinomial Naive Bayes, and Bernoulli Naive Bayes. Gaussian Naive Bayes is used when the features are continuous and are assumed to follow a normal distribution. This type is particularly effective in scenarios where the data exhibits normal distributional property. Multinomial Naive Bayes, on the other hand, is designed for discrete data and is frequently employed in text classification tasks, where word counts or frequencies serve as important features. Bernoulli Naive Bayes is another variant that works with binary or Boolean features, making it suitable for tasks like email spam detection where the presence or absence of certain words is used as a feature. Despite these differences, each of these variants retains the core simplicity and efficiency of the Naive Bayes algorithm while being optimized for different data characteristics and application areas.

The flexibility of Naive Bayes extends to hybrid models that combine different variants to leverage their respective strengths. For example, a hybrid model approach may integrate Gaussian and Multi-

nomial Naive Bayes to handle datasets containing both continuous and categorical features. Such hybrid models are particularly useful in complex real-world applications where data cannot be neatly categorized into a single type. By combining the probabilistic reasoning of Naive Bayes with the ability to process diverse data types, these hybrid models enhance both classification accuracy and robustness. The continuous development and refinement of Naive Bayes methodologies highlight its dynamic nature and potential for innovation, ensuring its continued utility in both emerging and established fields of machine learning.

This study utilizes the Naive Bayes classifier to perform a classification task on the Heart Disease dataset, aiming to determine whether an individual has heart disease. Here, Gaussian Naive Bayes, Categorical Naive Bayes and hybrid model integrating both Gaussian Naive Bayes and Categorical Naive Bayes are utilized to assess their effectiveness in classification. The results using these variants of Naive Bayes are observed and evaluated using various performance evaluation metrices like accuracy, precision, recall and F1 score to identify the optimal Naive Bayes classifier for this Heart disease dataset.

II. RELATED WORKS

The need for classification is essential in real-life applications. The Naive Bayes classifier, as a mathematical classification method, performs a series of probabilistic computations to determine the most suitable classification for a given data set within a specific problem domain. This paper [1] details the implementation of a Naive Bayes classifier, which serves as a versatile toolkit applicable to various classification domains. To validate the accuracy of the probabilistic computations, a sample data set is used to test this classifier.

Naive Bayes is a classification algorithm based on Bayes' theorem, assuming feature independence given the class. This paper [2] reviews Naive Bayes, covering its concepts, hidden Naive Bayes, text classification, traditional Naive Bayes, and applications in machine learning. It also discusses augmented Naive Bayes with examples and examines its applications, advantages, and disadvantages.

Among classification algorithms designed to predict the class or label of a categorical target variable, the Naïve Bayes classifier is noted for its simplicity and is commonly used with large text datasets. The article [3] describes the functioning of the Naive Bayes algorithm, including Bayes' theorem, binary classification problems, prior and class conditional probability computations, and posterior probability prediction. It also addresses handling continuous data, incomplete datasets, and mixed features with Scikit Learn, and highlights the primary limitation of the Naïve Bayes classifier: the assumption of feature independence.

In the study [4] by the authors, NPCm, a novel Naïve Bayesian-like Possibilistic Classifier for mixed categorical and numerical data, is introduced. This classifier integrates a bi-module belief estimation framework with the Generalized Minimum-based (G-Min) algorithm, originally designed for categorical data classification. Notably, the design incorporates a probability-to-possibility transform-based possibilistic approach for both categorical and numerical belief estimation modules, offering a robust alternative to traditional probabilistic methods in uncertain decision-making scenarios. Experimental validation using 12 datasets, encompassing mixed data types, demonstrates the superior performance of NPCm over conventional Bayesian-like classifiers, highlighting the efficacy of the bi-module possibilistic estimation strategy alongside the G-Min algorithm for mixed data classification.

The Naive Bayes induction algorithm is widely used in classification, but discretizing numeric data into symbols can impact performance. Recent attempts using normal distribution for numeric data often rely on single-value estimates, which can lead to inaccurate population estimates. This paper [5] proposes Extended Naive Bayes (ENB) to address these issues by combining the original Naive Bayes approach for categorical data with statistical methods that consider both mean and variance for continuous data. Experimental results demonstrate ENB's efficiency compared to classifiers like CART, DT, and MLP.

III. METHODOLOGY

A. DATASET INFORMATION

The Heart Disease dataset used in this study involves various factors affecting cancer and contains multiple columns that indicate symptoms of heart disease. It has a total of 270 instances and 14 columns which are as follows:

1) **age**: This column shows the age of patients ranging from 0 to 100.
2) **sex**: This column indicates the gender of patients (0 for Female, 1 for Male).
3) **chest**: This column shows the chest pain readings of patients.
4) **resting_blood_pressure**: Blood pressure readings where values below 120/80 mm Hg are considered normal. Elevated blood pressure is when readings consistently range from

120 to 129 systolic and less than 80 mm Hg diastolic.

5) **serum_cholestoral**: This measures the amount of high- and low-density lipoprotein cholesterol (HDL and LDL) in a person's blood.
6) **fasting_blood_sugar**: Normal fasting blood glucose concentration values are between 70 mg/dL (3.9 mmol/L) and 100 mg/dL (5.6 mmol/L). Values between 100 to 125 mg/dL (5.6 to 6.9 mmol/L) suggest lifestyle changes and monitoring glycemia.
7) **resting_electrocardiographic_results**: This column shows the results of a test that measures the electrical activity of the heart. The test usually takes 5 to 10 minutes.
8) **maximum_heart_rate_achieved**: The maximum heart rate is calculated by subtracting the patient's age from 220. For example, for a 45-year-old, the maximum heart rate is 175 beats per minute.
9) **exercise_induced_angina**: This column indicates whether stable angina was triggered by physical activity, where narrowed arteries slow down blood flow during increased heart demand.
10) **oldpeak**: This measures the ST segment shift relative to exercise-induced increments in heart rate.
11) **slope**: This represents the ST/heart rate slope, proposed as an accurate ECG criterion for diagnosing significant coronary artery disease (CAD).
12) **number_of_major_vessels**: This column shows the major blood vessels connected to the heart, including the aorta, superior vena cava, and inferior vena cava.
13) **thal**: Thalassemia is an inherited blood disorder causing less hemoglobin than normal.
14) **result**: This column indicates the result of the heart disease test (1 for Positive, 0 for Negative).

B. DATASET PREPROCESSING

Preprocessing the dataset is crucial to ensure effective model performance. In this study, several preprocessing steps were performed.

- **Handling Missing Values**: No missing values were found in the dataset, eliminating the need for imputation techniques.
- **Handling Categorical Features**: The categorical features were already encoded numerically, with each category represented by integers.
- **Handling Continuous Features**: Continuous variables were discretized into bins using `KBinsDiscretizer` with:
  - **Number of Bins**: 5
  - **Encode Method**: Ordinal (bins are encoded as integers)
  - **Strategy**: Uniform (bins are of equal width)
- **Dataset Splitting**: The dataset was split into training and testing sets with an 80:20 ratio. This split ensures that a majority of the data was used for training the model and a portion of the data is used to evaluate the model. The `stratify` parameter was used to ensure that the proportion of negative (result 0) and positive (result 1) outcomes remains balanced between the training and testing sets, with 150 instances of result 0 and 120 instances of result 1.

C. TERMINOLOGIES

Understanding Bayes' theorem and the Naive Bayes classifier involves knowing about several fundamental concepts in probability theory and machine learning. Below are key terminologies required for understanding these concepts:

1) **Probability**: The measure of the likelihood that an event will occur. It ranges from 0 (impossible event) to 1 (certain event).
2) **Conditional Probability**: The probability of one event occurring given that another event has already occurred. Mathematically, $P(A|B)$ represents the probability of event A given that event B has occurred.
3) **Joint Probability**: The probability of two (or more) events occurring together. For events A and B, it is denoted as $P(A \cap B)$.
4) **Prior Probability**: The initial belief about the probability of an event before evidence is taken into account. In Bayes' theorem, $P(A)$ is the prior probability of A.
5) **Posterior Probability**: The revised probability of an event after taking into consideration new evidence. In Bayes' theorem, $P(A|B)$ is the posterior probability of A given evidence B.

6) **Bayesian Inference**: The process of updating beliefs or predictions about a hypothesis as new evidence becomes available, using Bayes' theorem.
7) **Likelihood**: The probability of the evidence given the hypothesis, often denoted as $P(B|A)$ in Bayes' theorem.
8) **Evidence**: Evidence (or data) refers to the observed information or measurements that are used to update beliefs about hypotheses or model parameters. It is denoted as $P(\text{data})$ or $P(\text{evidence})$.
9) **Prior Probability**: Prior probability is the initial belief about the probability of a hypothesis (or an event) before observing any evidence. It is denoted as $P(\text{hypothesis})$ or $P(\text{prior})$.
10) **Posterior Probability**: Posterior probability is the updated probability of a hypothesis (or an event) after taking into account observed evidence or data. It is denoted as $P(\text{hypothesis}|\text{data})$ or $P(\text{posterior})$.
11) **Marginal Probability**: The probability of an event without considering any other events, often denoted as $P(B)$ in Bayes' theorem.

D. BAYES' THEOREM

Bayes' theorem is a fundamental theorem in probability theory named after the Reverend Thomas Bayes. It describes the probability of an event based on prior knowledge of conditions that might be related to the event. Mathematically it is expressed in Equation 1

Bayes' theorem allows us to update the probability of a hypothesis (event A) given new evidence (event B), by considering both the prior probability of A and the likelihood of B given A. This theorem is useful in various fields, like statistics, machine learning, and Bayesian inference

E. NAIVE BAYES CLASSIFIER

The Naive Bayes classifier is a probabilistic machine learning model based on applying Bayes' theorem with strong (naive) independence assumptions between the features. It predicts the probability of a class label C_k given a set of features $x_1, x_2, \ldots, x_n$.

1) Bayes' Theorem Application

The Naive Bayes classifier predicts the probability of a class label C_k given a set of features $x_1, x_2, \ldots, x_n$ using Bayes' theorem given in Equation 2

2) Naive Independence Assumption

The "naive" part of Naive Bayes comes from assuming that features are conditionally independent given the class label C_k. This simplifies the calculation of $P(x_1, x_2, \ldots, x_n|C_k)$ to the product of individual conditional probabilities as shown in Equation 3

This assumption reduces the computational complexity and makes the model computationally efficient even with large datasets. Using this assumption the Equation 2 changes to Equation 4. The denominator $P(x_1, x_2, \ldots, x_n)$ can be ignored during classification as it acts as a normalizing constant across all classes and does not affect the decision-making process. Therefore, the final form of the Naive Bayes equation simplifies to the form shown in Equation 5

3) Types of Naive Bayes Classifiers

Naive Bayes classifiers can be categorized into several types based on the distributional assumptions of the features:

1) **Gaussian Naive Bayes**:
Gaussian Naive Bayes assumes that continuous features follow a Gaussian (normal) distribution. It is suitable when the features are real-valued and can be modeled using a Gaussian distribution.
Working:
   - Estimate the mean $\mu_{k,i}$ and variance $\sigma^2_{k,i}$ of each feature x_i for each class C_k.
   - Use the Gaussian probability density function $P(x_i \mid C_k)$ to compute the likelihood shown in Equation 6

2) **Categorical Naive Bayes**:
Categorical Naive Bayes assumes that features have categorical distributions. It is suitable for discrete features, such as word counts or presence/absence of features.
Working:
   - Estimate the probability $P(x_i = v_j \mid C_k)$ for each category v_j of feature x_i and each class C_k.
   - The probability of category v_j in feature x_i given class C_k is calulated by using Equation 7

3) **Hybrid Naive Bayes**:
Hybrid Naive Bayes combines predictions from models trained on features with different distributions (e.g., Gaussian for continuous features, categorical for discrete features) within the same model. It is suitable when

the dataset contains a mix of continuous and discrete features.

Working:

- Train Gaussian Naive Bayes (GNB) on continuous features and Categorical Naive Bayes (CNB) on categorical features.
- Obtain probability estimates $\hat{P}(C_k \mid \mathbf{x})$ from both GNB and CNB.
- Combine the predictions by taking the arithmetic mean shown in Equation 8

F. SHORTCOMINGS OF NAIVE BAYES CLASSIFIER

While Naive Bayes classifiers are popular for their simplicity and efficiency, they have certain limitations that can affect their performance, especially in practical applications.

1) Zero Probability Issue

One of the fundamental assumptions of the Naive Bayes classifier is the independence of features given the class label. When a particular feature value does not appear in the training dataset for a given class, the conditional probability $P(x_i \mid C_k)$ becomes zero. This situation leads to the overall probability $P(x_1, x_2, \ldots, x_n \mid C_k)$ being zero as well.

2) Overcoming Zero Probability with Smoothing Techniques

To mitigate the issue of zero probabilities, smoothing techniques are employed, the most common of which are Laplace smoothing and the M-estimate.

- **Laplace Smoothing (Additive Smoothing)**: Laplace smoothing is a simple technique where a small positive value (typically 1) is added to all counts to ensure that no probability is ever zero. This method is effective in smoothing out extreme probabilities and its equation is shown in Equation 9
- **M-Estimate**: The M-estimate is another approach to smoothing that adjusts the probability estimate by considering a prior pseudo-count of all possible outcomes. It is formulated as shown in Equation 10

These smoothing techniques help in handling sparse data and prevent the Naive Bayes classifier from assigning zero probabilities to unseen feature values, thereby improving its robustness and generalization capability.

G. PERFORMANCE METRICS

When evaluating the effectiveness of a Naive Bayes Classifier, metrics such as Accuracy (Equation 11), Precision (Equation 12), Recall (Equation 13), and F1 Score (Equation 14) provide insights into its performance. Accuracy measures overall correctness but can be misleading with imbalanced datasets. Precision quantifies the proportion of correctly predicted positive instances among all predicted positives, while Recall assesses the proportion of correctly predicted positive instances out of all actual positives. The F1 Score balances Precision and Recall, providing a single metric that accounts for both. Additionally, ROC and PR curves with their respective AUC values offer insights into classifier performance across different threshold settings.

H. BLOCK DIAGRAM

The system block diagram for Naive Bayes classification algorithm is as illustrated in Figure 2. It outlines the process for designing and utilizing Naive Bayes classifier model to predict class of an instance. The first major step involves cleaning the dataset to remove any irrelevant or noisy data and correct any errors, ensuring the dataset is accurate and reliable. Then it undergoes preprocessing which includes encoding features, and selecting relevant features in the dataset. After preprocessing, the dataset is split into two subsets: a training dataset and a testing dataset. The training dataset is used to train the model, while the testing dataset is reserved for evaluating the model's performance on unseen data. At this point, a decision is made regarding which type of Naive Bayes model to use: CNB, GNB or HNB. The Naive Bayes model is then trained using the training dataset. Once the training is complete, the model's performance is tested using the testing dataset to evaluate how well it classifies new, unseen data.

Once the model has been trained and tested, various performance evaluation metrics are calculated to measure its effectiveness. Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) are also computed to gain insights into the model's performance.The results of these performance metrics are then outputted, providing a comprehensive evaluation of the model's performance.

I. FLOWCHART

The flowchart depicted in Figure 1, illustrates the sequential steps involved in constructing and analyzing a dataset using a Naive Bayes classifier. The implementation of a Naive Bayes classifier starts with importing the dataset, ensuring that all necessary data is loaded into the system. The next

step is to clean the dataset, which involves removing any irrelevant or noisy data, correcting errors, and ensuring the dataset is accurate and consistent. After cleaning, features in the dataset are encoded to convert categorical data into numerical values that the model can understand. Once the dataset is preprocessed, it is split into two parts: a training set and a testing set.

If the CNB model is selected, all continuous values in the dataset are discretized into categorical bins. The model is then trained using the training dataset and subsequently tested using the testing dataset to evaluate its performance. If the GNB model is chosen, the process skips the discretization step. The GNB model is directly trained on the training dataset, assuming the data follows a normal distribution, and then tested on the testing dataset. For a HNB approach, the dataset is split into categorical and continuous data. The GNB model is trained on the continuous data, while the CNB model is trained on the categorical data. Both models are then tested using the testing dataset, and their outputs are averaged to make the final predictions. If the average output is greater than or equal to 0.5, the prediction is set as true (1).

After the model has been trained and tested, various performance evaluation metrics such as accuracy, precision, recall, and the F1 score are calculated to assess its effectiveness. Additionally, the Receiver Operating Characteristic (ROC) curve and the Area Under the Curve (AUC) are calculated to provide further insights into the model's performance. The results of the testing and the calculated performance metrics are outputted, providing a comprehensive evaluation of the model's performance.

dThe

IV. IMPLEMENTATION DETAILS AND RESULTS

The Heart Disease dataset was analyzed, which revealing an imbalanced class distribution with more instances labeled as "No" (150) compared to "Yes." (120) The dataset was free of empty or null values and consisted of both continuous and categorical features. Categorical data were given in numerical form so encoding was not needed when training the Gaussian Naive Bayes Classifier, The continuous data were discretized into 5 bins of equal width when training the Categorical Naive Bayes Classifier also the dataset was split into training and testing sets.

The various distribution of continuous data can be visualized starting from Figure 3 up to Figure 7. For this visualization purpose the data was discritezed into 30 bins and Kernel Density Estimate of the data is also shown which unlike the jagged histograms is a continuous and smooth estimate of the data distribution.

Before training the Naive Bayes Classifier a correlation matrix between each of the features was calculated which is shown in Figure 17. The diagonal values are all 1 since the correlation of any variable with itself is 1. From this matrix we can observe the highest correlation between 2 variables is between columns age and maximum heart rate achieved with a value of -0.402. This value indicates a moderate negative linear relationship between the variables. Naive Bayes assumes features are conditionally independent given the class label. While this correlation level suggests some interdependence between features, Naive Bayes can still be trained with this level of correlation without significantly violating its assumptions.

A Gaussian Naive Bayes classifier was initialized and trained using the training set. Its performance can be observed through the various metrics shown in the classification report in Table 1. The confusion matrix is also shown in Figure 8. This classifier obtained an accuracy score of 0.83. The ROC curve (Figure 9) and the PR curve (Figure 10) was also plotted by varying the threshold value and the area under curve was calculated as 0.93 and 0.92 respectively.

Then, a Categorical Naive Bayes classifier was initialized and trained using the training set. But before the training the continuous data values were discretized to 5 bins having equal widths. Its performance can be observed through the various metrics shown in the classification report in Table 2. The confusion matrix is also shown in Figure 11. This classifier obtained an accuracy score of 0.85. The ROC curve (Figure 12) and the PR curve (Figure 13) was also plotted by varying the threshold value and the area under curve was calculated as 0.92 and 0.88 respectively. The accuracy of the categorical classifier was higher than the accuracy of the gaussian classifier but the area under curve of ROC curve and PR curve is lower for categorical classifier

Finally, a Hybrid Naive Bayes Classifer was trained. To train this hybrid model firstly a gaussian classifier was trained using just the continuous data and a categorical classifier was trained using just the categorical data. Finally the prediction probabilities from both the model were combined taking their mean to get the final Hybrid Naive Bayes Classifier. Its performance can be observed through the various metrics shown in the classification report in Table 3. The confusion matrix is also shown in Figure 14. This classifier obtained an accuracy score of 0.89. The ROC curve (Figure 15) and the PR curve (Figure 16) was also plotted by varying

the threshold value and the area under curve was calculated as 0.95 for both curves. The hybrid classifier performed the best out of the 3 classifiers achieving the highest values in F1-score, accuracy and area under curve values for ROC curve and PR curve.

From the classification report shown in Table 1, Table 2, Table 3 we can observe different metrics like accuracy, precision, recall and f1-score. There are 2 values of precision, recall, f1-score each value for each of the classes in the dataset. We can also see the number of occurrence of each class through Support and various averages(Weighted and Macro) of precision, recall and F1-score. The Macro Average is calculated for each label and calculates their unweighted mean. This does not take label imbalance into account. The Weighted Average calculates the average weighted by the support. This accounts for label imbalance.

The confusion matrix shown in Figure 8, Figure 11, Figure 14 shows the number of accurate prediction and number of false prediction. It shows the number of True Negatives and False Positives in the first row and the number of False Negative and True Positive in the second row. True positive is the case where the model predicted the class correctly as positive. True Negative is the case where the model predicted the class correctly as negative. False positive also known as Type 1 erorr, are cases where the model predicted the class as positive, but the actual class is negative and False Negative also known as Type 2 error, are cases where the model predicted the class as negative but the actual class is positive.

The area under curve value of ROC curve quantifies the overall performance of the classifier with high value indicating better performance. The area under curve value of PR curve summarizes the PR curve in a single value with high value indicating better performance. In both cases the AUC value ranges from 0 to 1. In the ROC Curve (Figure 9, Figure 12, Figure 15) a dotted diagonal line is also plotted which represents the performance of a random classifier which makes classification based on random guesses. If the ROC curve lies close to this diagonal curve it is not performing better than random chance. So, this line can also be used as an indication of the classifier's performance. The further the ROC curve is from the diagonal line towards the top-left corner the better is the classifier's performance. In our ROC curve for all 3 classifiers, it is further than the diagonal line towards the top-left corner so we the classifier is performing nicely.

V. DISCUSSION AND ANALYSIS

Figure 3-Figure 7 show the distributions of continuous variables employed for training the Naive Bayes models, it is observed that most distributions exhibit a nearly Gaussian shape with slight skewness. This observation shows that while the variables are comparable to a Gaussian distribution, there are deviations that could impact the assumptions of perfect normality in Gaussian Naive Bayes. Naive Bayes models, however, are known for their robustness to such deviations, often performing adequately even when the underlying assumptions are not perfectly met which can be proved by the high scores of various evaluation metrics. The variable 'old peak' displays a distinctly right-skewed distribution. This skewness indicates that the majority of values cluster towards lower values, with a long tail extending towards higher values. Such skewness can pose challenges in accurately modeling extreme values or outliers within this variable.

Figure 9, Figure 12, Figure 15 show the ROC curve it is shaped like a ladder due to its contruction based on varying thresholds for classifying the binary outcome. The ROC curve starts at the bottom-left corner where both the True Positive and False Positive Rate is low. As the threshold decreases the model classifies more instances as positive which increases both the true positive, and false positive rate.

The Precision-Recall (PR) curve (Figure 10, Figure 13, Figure 16) show how precision and recall vary with the classification threshold. At the beginning of the curve, precision tends to be high while recall is relatively low. This occurs because a high threshold means the classifier predicts positive instances only when very confident, resulting in fewer false positives but missing many actual positives. As the threshold decreases along the curve, recall increases as more true positives are identified, but this inclusion of more positives also increases false positives, thereby decreasing precision. The PR curve demonstrates the trade-off between precision and recall, where higher precision requires a higher threshold to maintain accuracy, whereas higher recall requires a lower threshold to capture more positives but risks more false positives. The Area Under the PR Curve (AUCPR) summarizes the overall performance, indicating how well the classifier balances precision and recall across all thresholds.

A. COMPARING NAIVE BAYES MODELS

In our comparative study, we trained Gaussian Naive Bayes (GNB), Categorical Naive Bayes (CNB), and Hybrid Naive Bayes (HNB) using a dataset comprising both categorical and continuous

features, employing distinct preprocessing strategies.

1) Model Training Approach

GNB was trained using all features, where categorical variables were not encoded as they were already in integer format. Conversely, CNB utilized the same feature set, but continuous variables underwent discretization into ordinal categories using `KBinsDiscretizer` while categorical ones remained unchanged. The Hybrid Naive Bayes model utilized both GNB and CNB independently on continuous and categorical features, respectively, and combined their predictions by averaging the prediction probabilities for final classification.

2) Performance Metrics Comparison

CNB demonstrated better performance metrics with an accuracy of 0.85 compared to GNB's 0.83, and a slightly higher F1-score of 0.85 against GNB's 0.84. These results suggest CNB's better adaptability to the dataset's features and underlying distributions. However, despite GNB's lower accuracy, it achieved higher AUC values for both ROC (0.93) and PR (0.92) curves compared to CNB's AUC values of 0.92 and 0.88, respectively.

The Hybrid Naive Bayes model outperformed both GNB and CNB with an accuracy and F1-score of 0.89. Additionally, it achieved higher AUC values for both ROC (0.95) and PR (0.95) curves compared to GNB and CNB. This performance enhancement can be attributed to leveraging the strengths of both models: GNB's flexibility in modeling continuous data and CNB's effectiveness with categorical features.

CNB likely outperformed GNB due to its specific design for categorical data, effectively handling discrete features and their dependencies. Also, since the dataset included more categorical features and the continuous features deviated from perfect Gaussian distributions, these factors could have contributed to CNB achieving higher classification accuracy compared to GNB.

The higher AUC values for GNB despite its lower accuracy indicate its robustness in distinguishing between classes near decision boundaries. CNB may have lower AUC values due to its reliance on discrete categorical features, which can lead to less discriminative power near decision boundaries. This categorical approach might struggle to generalize across varying data distributions, especially in scenarios where features exhibit complex dependencies that are not effectively captured by the model's assumptions. Additionally, the discretization of continuous values in CNB could contribute to information loss, as fine-grained differences between values within each category are smoothed out, potentially reducing the model's ability to distinguish subtle variations in the data.

VI. CONCLUSION

Based on the results of the study, it is evident that the Naive Bayes classifier performed well on the Heart Disease dataset, which consisted of both continuous and categorical features and had no missing values. The application of different Naive Bayes classification algorithm on this dataset has provided valuable insights into the behavior and performance of the models. The dataset exhibited an imbalanced class distribution with more instances labeled as "No" (150) compared to "Yes" (120). The data preprocessing steps involved encoding categorical data, which were already in numerical form, and discretizing continuous data into 5 bins of equal width for the Categorical Naive Bayes Classifier. The correlation matrix revealed some interdependence between features, particularly a moderate negative linear relationship between age and maximum heart rate achieved. Despite this, the Naive Bayes model, which assumes feature independence given the class label, was successfully trained and evaluated. By implementing the various Naive Bayes classifier, we successfully constructed different model that accurately classified instances based on the provided features.

The obtained results highlight the predictive capabilities and effectiveness of the these Naive Bayes model in classifying if a patients has heart disease. The performance of the classifiers was assessed using various metrics, including accuracy, precision, recall, F1 score, and the area under the ROC and PR curves. The Gaussian Naive Bayes classifier achieved an accuracy of 0.83, with AUC values of 0.93 and 0.92 for the ROC and PR curves, respectively. The Categorical Naive Bayes classifier outperformed the Gaussian model in terms of accuracy (0.85) but had slightly lower AUC values (0.92 and 0.88). The Hybrid Naive Bayes classifier, which combined predictions from both Gaussian and Categorical models, demonstrated the best overall performance with an accuracy of 0.89 and AUC values of 0.95 for both curves. The hybrid model achieved the highest values in F1-score, accuracy, and AUC, indicating its superior ability to classify heart disease. The confusion matrices provided insights into the classifiers' prediction capabilities, showing the number of true positives, true negatives, false positives, and false negatives. This study showcases the application of Naive Bayes in classification tasks, affirming their viability and effectiveness in predicting heart disease. The study concluded that the Hybrid Naive Bayes classifier is

the most effective approach for this dataset, providing robust and reliable predictions for heart disease classification.

APPENDIX

A. EQUATIONS

1) Bayes Theorem

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{1}$$

where:

- $P(A|B)$ is the probability of event A occurring given that B is true.
- $P(B|A)$ is the probability of event B occurring given that A is true.
- $P(A)$ and $P(B)$ are the probabilities of events A and B occurring independently.

2) Naive Bayes

$$P(C_k|x_1, x_2, \ldots, x_n) = \frac{P(x_1, x_2, \ldots, x_n|C_k) \cdot P(C_k)}{P(x_1, x_2, \ldots, x_n)} \tag{2}$$

where:

- $P(C_k|x_1, x_2, \ldots, x_n)$: Posterior probability of class C_k given features $x_1, x_2, \ldots, x_n$.
- $P(x_1, x_2, \ldots, x_n|C_k)$: Likelihood of observing features $x_1, x_2, \ldots, x_n$ given class C_k.
- $P(C_k)$: Prior probability of class C_k.
- $P(x_1, x_2, \ldots, x_n)$: Probability of observing features $x_1, x_2, \ldots, x_n$.

$$P(x_1, x_2, \ldots, x_n|C_k) = \prod_{i=1}^{n} P(x_i|C_k) \tag{3}$$

$$P(C_k|x_1, x_2, \ldots, x_n) = \frac{P(C_k) \prod_{i=1}^{n} P(x_i|C_k)}{P(x_1, x_2, \ldots, x_n)} \tag{4}$$

$$P(C_k|x_1, x_2, \ldots, x_n) \propto P(C_k) \prod_{i=1}^{n} P(x_i|C_k) \tag{5}$$

3) Gaussian Naive Bayes

$$P(x_i \mid C_k) = \frac{1}{\sqrt{2\pi\sigma_{k,i}^2}} \exp\left(-\frac{(x_i - \mu_{k,i})^2}{2\sigma_{k,i}^2}\right) \tag{6}$$

4) Categorical Naive Bayes

$$P(x_i = v_j \mid C_k) = \frac{N_{ic} + \alpha}{N_c + \alpha \cdot |V_i|} \tag{7}$$

where N_{ic} is the number of times category v_j appears in the samples belonging to class C_k, N_c is the number of samples with class C_k, $|V_i|$ is the number of available categories of feature x_i, and α is a smoothing parameter.

5) Hybrid Naive Bayes

$$P(C_k \mid \mathbf{x}) = \frac{1}{2}\left(\hat{P}_{\text{GNB}}(C_k \mid \mathbf{x}) + \hat{P}_{\text{CNB}}(C_k \mid \mathbf{x})\right) \tag{8}$$

where $\hat{P}_{\text{GNB}}(C_k \mid \mathbf{x})$ and $\hat{P}_{\text{CNB}}(C_k \mid \mathbf{x})$ are the probability estimates from GNB and CNB respectively.

6) Laplace Smoothing

$$P(x_i = v_j \mid C_k) = \frac{N_{ic} + \alpha}{N_c + \alpha \cdot |V_i|} \tag{9}$$

where N_{ic} is the count of category v_j in feature x_i for class C_k, N_c is the total count of samples in class C_k, $|V_i|$ is the number of possible categories of feature x_i, and α is the smoothing parameter (usually set to 1).

7) M-Estimate Smoothing

$$P(x_i = v_j \mid C_k) = \frac{N_{ic} + \alpha \cdot P_{\text{prior}}(x_i = v_j)}{N_c + \alpha} \tag{10}$$

where $P_{\text{prior}}(x_i = v_j)$ is the prior probability of category v_j under class C_k, and α is a smoothing parameter.

8) Accuracy

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

where:

- TP (True Positives) are the correctly predicted positive instances.
- TN (True Negatives) are the correctly predicted negative instances.
- FP (False Positives) are the incorrectly predicted positive instances.
- FN (False Negatives) are the incorrectly predicted negative instances.

9) Precision

$$\text{Precision} = \frac{TP}{TP + FP} \tag{12}$$

10) Recall

$$\text{Recall} = \frac{TP}{TP + FN} \tag{13}$$

11) F1 Score

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{14}$$

B. FIGURES

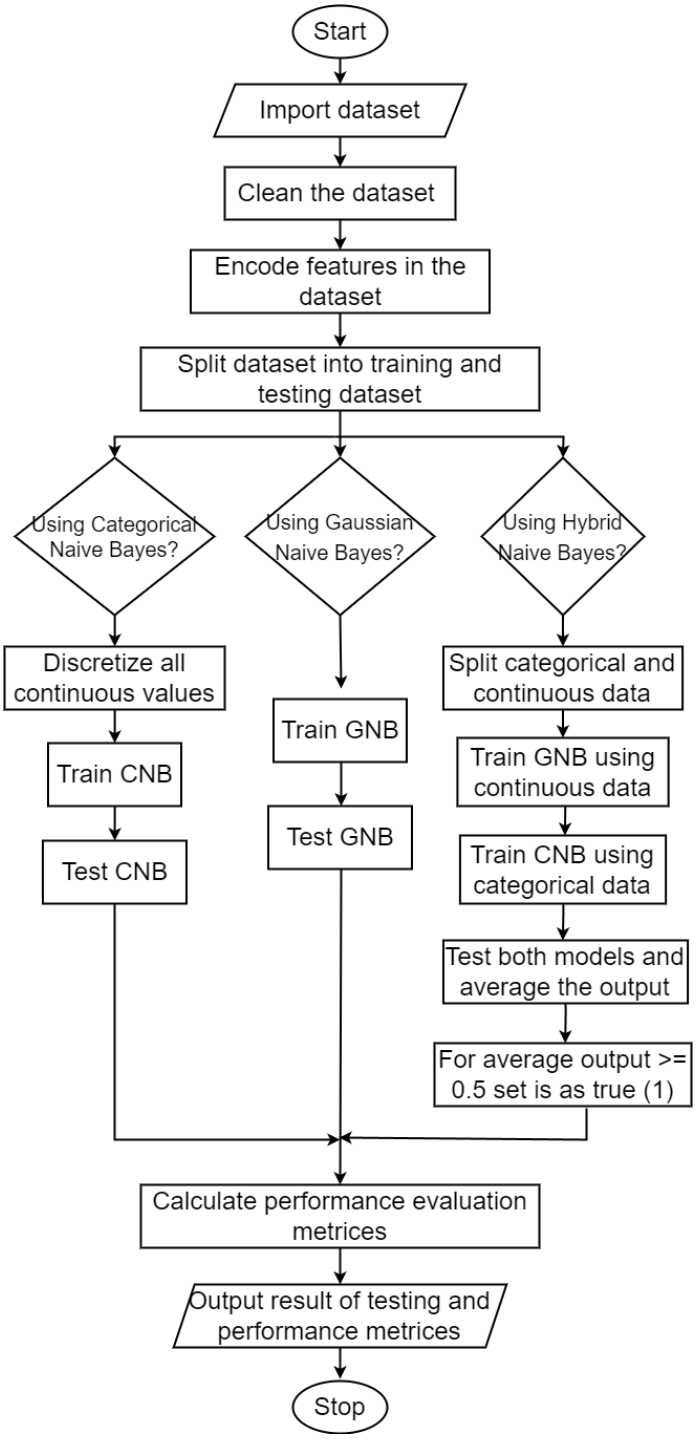

FIGURE 1. Flowchart

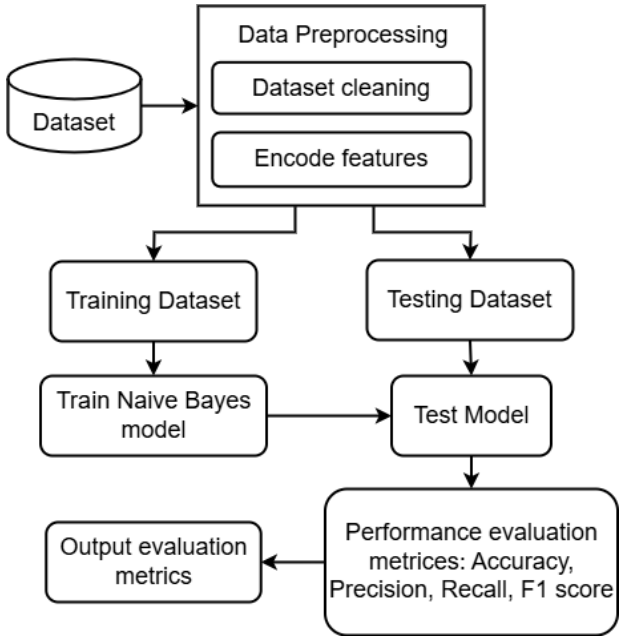

FIGURE 2. System Block Diagram

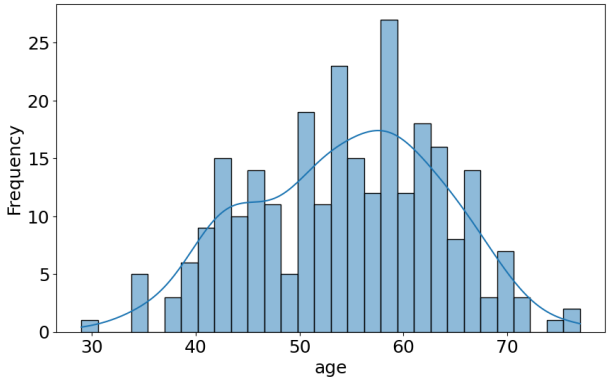

FIGURE 3. Distribution of Age

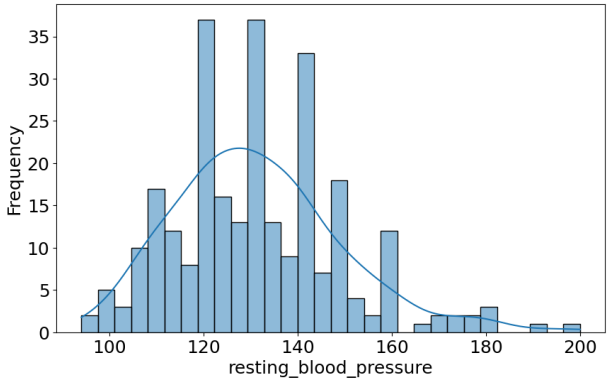

FIGURE 4. Distribution of Resting Blood Pressure

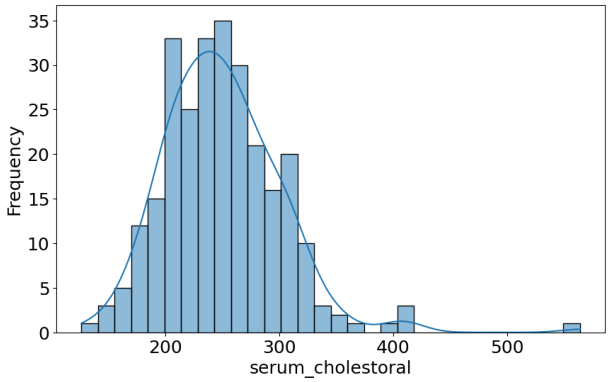

FIGURE 5. **Distribution of Serum Cholestoral**

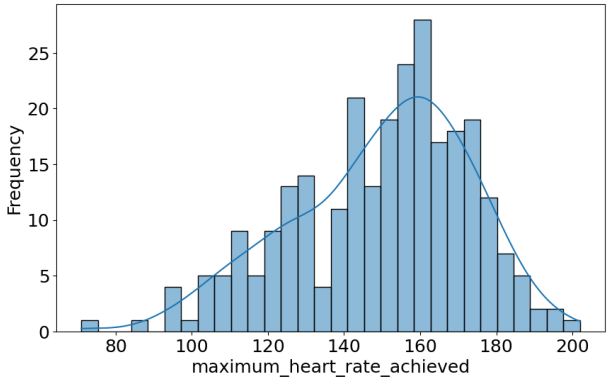

FIGURE 6. **Distribution of Maximum Heart Rate Achieved**

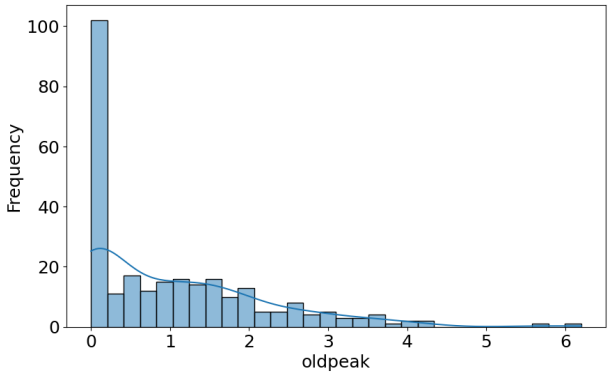

FIGURE 7. **Distribution of Old Peak**

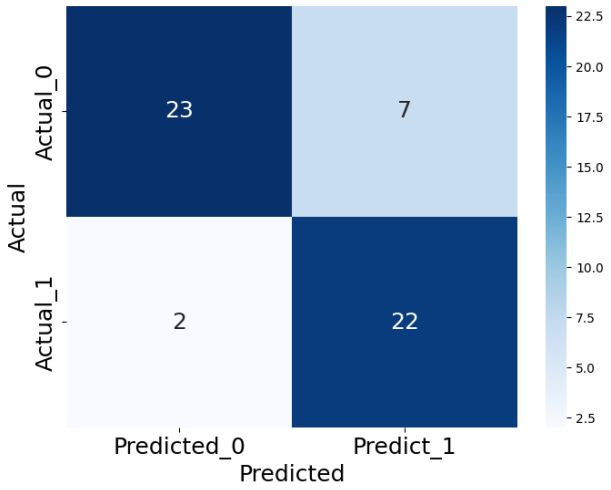

FIGURE 8. **Confusion Matrix of Gaussian Naive Bayes Classifier**

Class	Precision	Recall	F1-Score	Support
0	0.92	0.77	0.84	30
1	0.76	0.92	0.83	24
Accuracy			0.83	54
Macro Avg	0.84	0.84	0.83	54
Weighted Avg	0.85	0.83	0.83	54

TABLE 1. **Classification Report of Gaussian Naive Bayes Classifier**

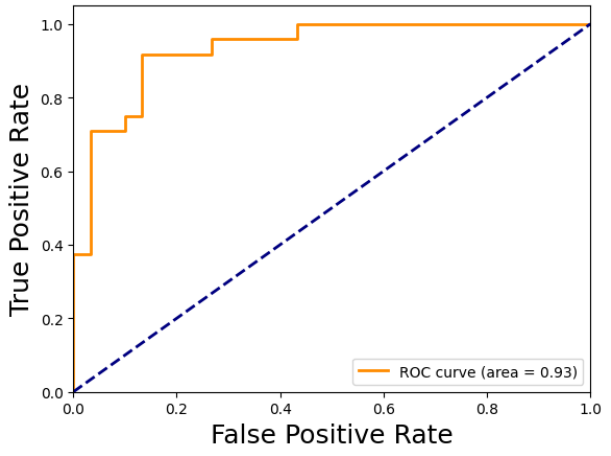

FIGURE 9. **ROC Curve of Gaussian Naive Bayes Classifier**

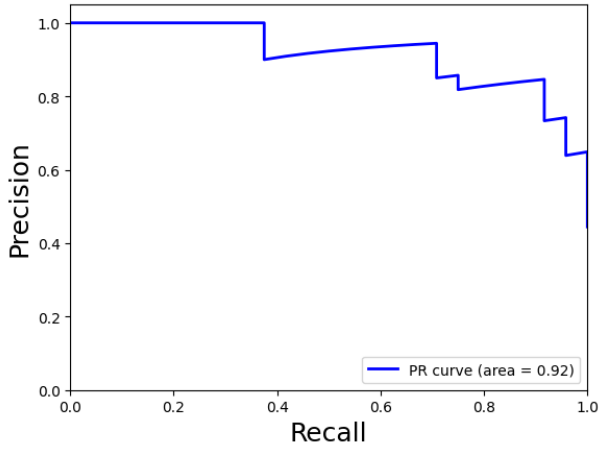

FIGURE 10. PR Curve of Gaussian Naive Bayes Classifier

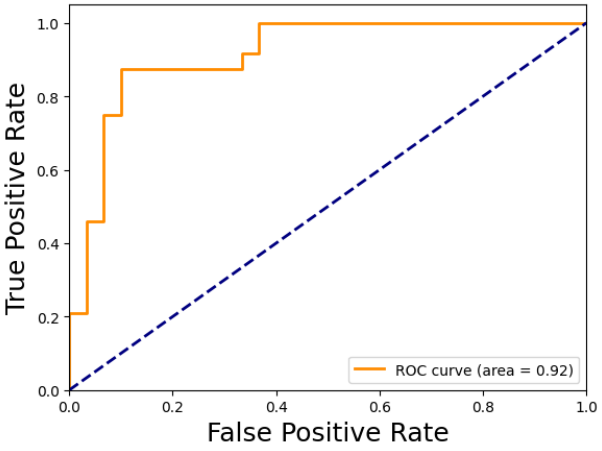

FIGURE 12. ROC Curve of Categorical Naive Bayes Classifier

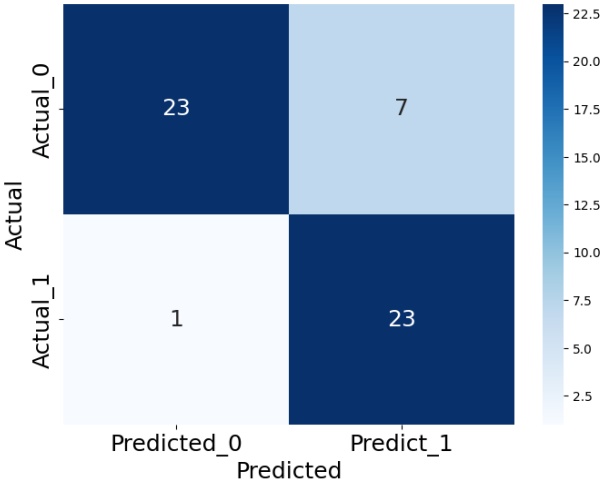

FIGURE 11. Confusion Matrix of Categorical Naive Bayes Classifier

Class	Precision	Recall	F1-Score	Support
0	0.96	0.77	0.85	30
1	0.77	0.96	0.85	24
Accuracy			0.85	54
Macro Avg	0.86	0.86	0.85	54
Weighted Avg	0.87	0.85	0.85	54

TABLE 2. Classification Report of Categorical Naive Bayes Classifier

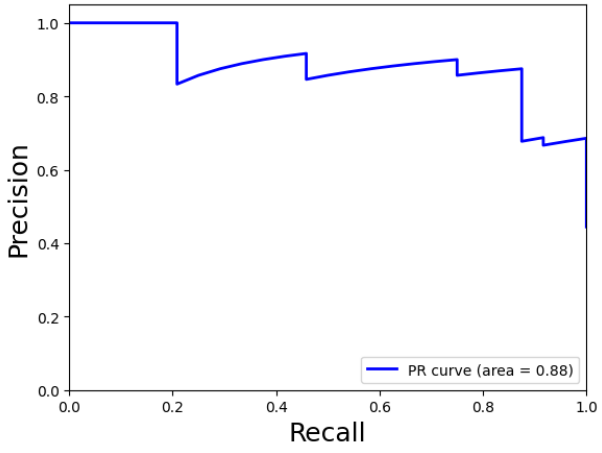

FIGURE 13. PR Curve of Categorical Naive Bayes Classifier

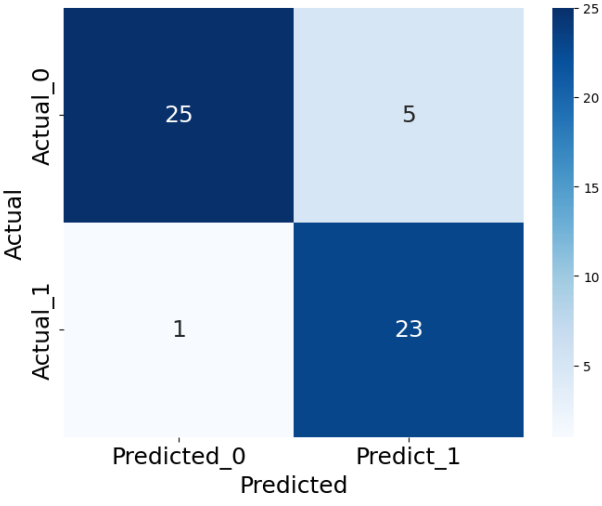

FIGURE 14. Confusion Matrix of Hybrid Naive Bayes Classifier

Class	Precision	Recall	F1-Score	Support
0	0.96	0.83	0.89	30
1	0.82	0.96	0.88	24
Accuracy			0.89	54
Macro Avg	0.89	0.90	0.89	54
Weighted Avg	0.90	0.89	0.89	54

TABLE 3. Classification Report of Hybrid Naive Bayes Classifier

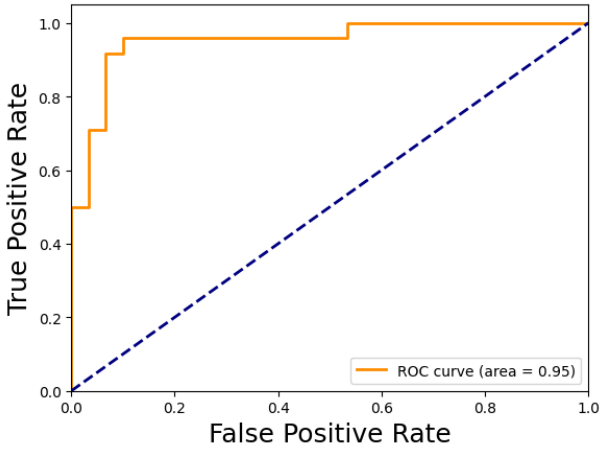

FIGURE 15. ROC Curve of Hybrid Naive Bayes Classifier

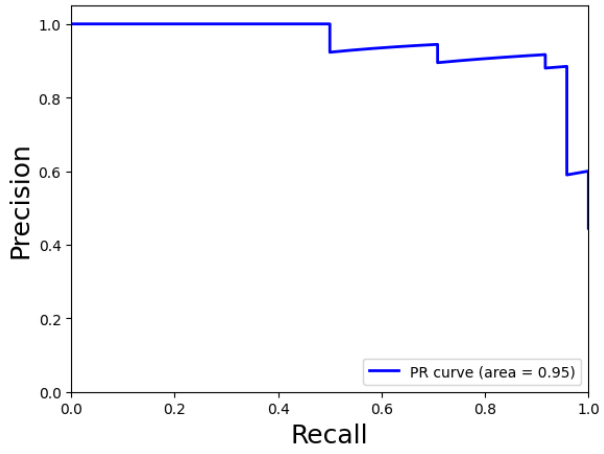

FIGURE 16. PR Curve of Hybrid Naive Bayes Classifier

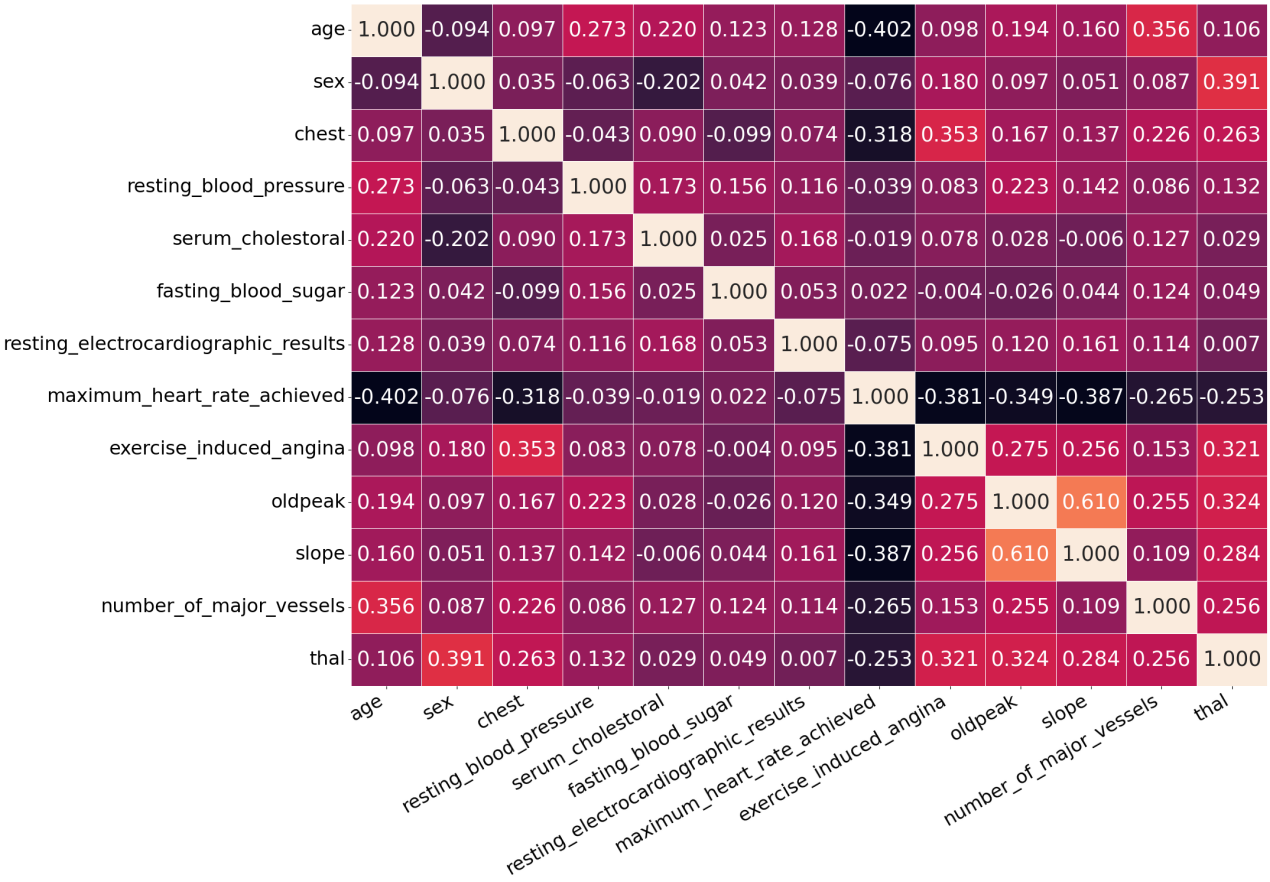

FIGURE 17. Correlation Matrix

REFERENCES

[1] F. -J. Yang, "An Implementation of Naive Bayes Classifier," *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA, 2018, pp. 301-306.

[2] P. Kaviani and S. Dhotre, "Short Survey on Naive Bayes Algorithm," *International Journal of Advance Research in Computer Science and Management*, vol. 04, Nov. 2017.

[3] R. Roy, "The Naive Bayes classifier," *Towards Data Science*, Dec. 29, 2021. [Online]. Available: https://towardsdatascience.com/the-naive-bayes-classifier-how-it-works-e229e7970b84. [Accessed: June 30, 2024].

[4] K. Baati, T. M. Hamdani, A. M. Alimi, and A. Abraham, "A New Possibilistic Classifier for Mixed Categorical and Numerical Data Based on a Bi-module Possibilistic Estimation and the Generalized Minimum-based Algorithm," *Journal of Intelligent & Fuzzy Systems*, vol. 36, no. 4, pp. 3513-3523, Apr. 2019.

[5] C.-C. Hsu, Y.-P. Huang, and K.-W. Chang, "Extended Naive Bayes classifier for mixed data," *Expert Systems with Applications*, vol. 35, no. 3, pp. 1080-1083, 2008.

NIMESH G. PRADHAN is currently pursuing a Bachelor's degree in Electronics, Communication, and Information Engineering at Thapathali Campus. He is currently in the final year of his degree. His interests lie in the fields of Data Mining, Machine Learning, and Deep Learning.

RAMAN BHATTARAI is currently pursuing a Bachelor's degree in Electronics, Communication, and Information Engineering at Thapathali Campus. He is currently in the final year of his degree. His interests lie in the fields of Data Mining, Computer Vision, and Deep Learning.