**Tribhuvan University**

**Institute of Science and Technology**



**A seminar Report**

**on**

**“****Comparative Analysis of Decision Tree, Random Forest and Naive Bayes for Breast Cancer Prediction**”

In Partial Fulfillment of Requirements for the master's degree in computer science and information technology

**Under Supervision of:**

Mr. Bikash Balami

**Submitted to:**

Central Department of Computer Science and Information Technology

Tribhuwan university

Kathmandu, Nepal

**Submitted By:**

Nischal Budhathoki (8015024)

**Date:** July 2024



**Supervisor’s Recommendation**

I hereby recommend that this seminar prepared under my supervision by Nischal Budhathoki **entitled “Comparative Analysis of Decision Tree, Random Forest and Naive Bayes for Breast Cancer Prediction”** in partial fulfillment of the requirements for the master's degree in computer science and information technology is recommended for the final evaluation.

**….………………**

**Asst.prof. Bikash Balami**

Central Department of Computer Science and Information Technology (CDCSIT)

Tribhuwan University

**Tribhuvan University**

**Institute of Science and Technology**



**CERTIFICATE OF APPROVAL**

The undersigned certify that they have read and recommended to the **Department of Computer Science and Information Technology, Tribhuwan university**, a seminar report entitled “**Comparative Analysis of Decision Tree, Random Forest and Naive Bayes for Breast Cancer Prediction**” submitted by **Nischal Budhathoki**. The seminar was carried out under special supervision and within the syllabus’s prescribed time.

We found the student to be hardworking, skilled, and ready to undertake any related work to their field of study.

----------------------------------- --------------------------------------------

**Signature of the Supervisor**  **Signature of the HOD**

**---------------------------------------** **--------------------------------------------**

**Signature of the Internal Examiner**   **Signature of the External Examiner**

## ACKNOWLEDGEMENT

I wish to express our gratitude and appreciation to many individuals who have helped us to complete this seminar. This seminar would not have been successful without their help and support.

First, I sincerely thank Central Department of computer science for providing the support and infrastructure needed to prepare this seminar.

I am grateful to my supervisor **Mr. Bikash Balami** who helped us and provided the necessary guidance by providing ideas and knowledge needed for this seminar. I am indebted to his supportive role and encouragement which helped me throughout the intense period of this seminar.

I am also thankful to all the teachers in our department who provided much valuable knowledge and immense support which made this seminar successful and colleagues whose support and suggestions have been invaluable contributions that even gave the inspiration to improve this seminar Their continuous encouragement and valuable comments were truly appreciable.

Finally, I would like to express our profound appreciation to everyone who has provided me with direct and indirect guidance throughout this seminar.

## Abstract

This seminar focuses on comparing performance of three distinct machine-learning algorithms: Decision Tree, Random Forest, and Naive Bayes as machine-learning algorithms have shown great promise in predicting breast cancer outcomes using various clinical data. However, choosing the most suitable algorithm depends on several factors, such as data quality, interpretability, and clinical context. This study compares and evaluates the performance of the above-mentioned algorithms for breast cancer prediction using the publicly available Breast Cancer Wisconsin (Diagnostic) Dataset, UCI machine learning repository via Kaggle. The performance of each algorithm is evaluated using metrics such as accuracy, precision, recall, f1-score, AUC-ROC, and Confusion matrix to identify the most appropriate algorithm through comparison and concluded that Random Forest is the best model that classifies the malignant and benign cases in this dataset compared to Decision Tree and Naive Bayes. Our results demonstrate that Random Forest outperforms Decision Tree and Naive Bayes in all metrics except recall, where all three models perform similarly. Random Forest achieves an accuracy of 96.49%, precision of 95.89%, F1-score of 97.22%, and AUC-ROC of 0.96. These findings suggest that Random Forest is the most suitable algorithm for breast cancer prediction in this dataset compared to Decision Tree and Naive Bayes.

**Keywords**: *Benign, Decision Tree, Malignant, Naive Bayes, Random Forest*

**Table of Contents**

[ACKNOWLEDGEMENT 3](#_Toc1693680949)

[Abstract 4](#_Toc1625309526)

[List of Figures 6](#_Toc928068165)

[List of Abbreviations 6](#_Toc1867534480)

[CHAPTER 1: INTRODUCTION 7](#_Toc1003173117)

[1.1 Introduction 8](#_Toc2106908434)

[1.2 Statement of Problem 8](#_Toc1718139795)

[1.3 Objectives 8](#_Toc1011077390)

[CHAPTER 2: LITERATURE REVIEW 9](#_Toc1100048544)

[2.1 Literature Review 9](#_Toc1231928746)

[CHAPTER 3: METHODOLOGY 11](#_Toc101487051)

[3.1 Conceptual Framework 12](#_Toc317187748)

[3.2 Data Set Description 12](#_Toc865659967)

[3.3 Data Preprocessing 13](#_Toc52490147)

[3.4 Algorithm 13](#_Toc1033971562)

[3.5 Result Evaluation Metric 18](#_Toc247009904)

[CHAPTER 4: IMPLEMENTATION AND RESULT ANALYSIS 21](#_Toc94633084)

[4.1. Implementation 21](#_Toc880491446)

[4.2. Findings And Result Analysis 22](#_Toc90483433)

[CHAPTER 5: CONCLUSION AND FUTURE WORK 29](#_Toc146026227)

[5.1 Conclusion 29](#_Toc1329247074)

[5.2 Future Work 29](#_Toc90156796)

[References 29](#_Toc256894898)

[Appendix 30](#_Toc365653219)

## List of Figures

[Figure 1: Conceptual Framework 5](#_Toc172615121)

[Figure 2: Decision Tree 8](#_Toc172615122)

[Figure 3: Random Forest 9](#_Toc172615123)

[Figure 4: Naive Bayes 11](#_Toc172615124)

[Figure 5: Confusion Matrix 11](#_Toc172615125)

[Figure 6: AUC-ROC 13](#_Toc172615126)

[Figure 7: Confusion matrix of Decision Tree 14](#_Toc172615127)

[Figure 8: Confusion matrix of Random Forest 15](#_Toc172615128)

[Figure 9: Confusion matrix of Naive Bayes 15](#_Toc172615129)

[Figure 10: Accuracy Score 16](#_Toc172615130)

[Figure 11: Precision Score 16](#_Toc172615131)

[Figure 12:Recall Score 17](#_Toc172615132)

[Figure 13: F1-Score 18](#_Toc172615133)

[Figure 14: AUC-ROC Curve 18](#_Toc172615134)

[Figure 15: Multi-bar graph for each Model 19](#_Toc172615135)

## List of Abbreviations

**ASM** Attribute Selection Measure

**CART** Classification and Regression Tree

**DT** Decision Tree

**WHO** World Health Organization

## CHAPTER 1: INTRODUCTION

#### 1.1 Introduction

Breast cancer is one of the most common cancers in women worldwide and it is one of the prevalent reasons for death. As indicated by the report by the WHO every year globally, a total of 2.3 million women were diagnosed with breast cancer, and 685,000 deaths occurred due to this disease, in the year 2020 – that is 15% of all cancer death among women. The mortality rate of women due to breast cancer can be reduced if it can be detected at an early stage. Machine learning can be used to predict breast cancer based on the data obtained. Machine learning techniques are becoming increasingly useful in healthcare analysis. Different machine learning algorithms have shown promising results in breast cancer prediction based on the provided data. However, different algorithms have different strengths and weaknesses, and there is a need to compare and evaluate their performance. This study performs a comparative analysis of three machine learning algorithms; Decision Tree, Random Forest, and Naive Bayes. Firstly, all three algorithms will be trained on the same dataset of patient information and then compare and evaluate using metrics like confusion matrix, accuracy, recall, precision, f1-score, and AUC- ROC of an algorithm to know which algorithm provides better results among the three. Hence, this study provides insights into the strengths and weaknesses of different machine learning algorithms for breast cancer prediction. Which could help in selecting the most appropriate algorithm for breast cancer prediction based on the available data and clinical context.

#### 1.2 Statement of Problem

Breast cancer is the primary reason for cancer-related mortality for women globally. Preliminary detection and accurate diagnosis of breast cancer can enhance patient outcomes. However, accuracy of breast cancer prediction models depends on the choice of the predictive model and the quality of the data used to train these models. Several machine learning algorithms have been used for breast cancer prediction, but there is a need to compare and evaluate their performance. Additionally, the choice of the best algorithm for breast cancer prediction depends on several factors, such as the clinical context, availability and quality of data, and interpretability of the model. Therefore, the problem addressed in this study is the need to compare and evaluate the performance of different machine learning algorithms for breast cancer prediction and to identify the most appropriate algorithm based on the available data and clinical context.

#### 1.3 Objectives

The primary objectives of this study are:

* To implement three different machine learning algorithms: Decision tree, Random Forest, and Naïve Bayes for breast cancer prediction.
* To compare and evaluate the effectiveness of the above-mentioned algorithms based on performance metrics: accuracy, f1- score, precision, recall, AUC\_ROC, and confusion matrix.

## CHAPTER 2: LITERATURE REVIEW

#### 2.1 Literature Review

Kumar, Sushil, et al. [1] conducted research that implements and compares several widely used classification approaches, including Decision Tree, K-Nearest Neighbor, Support Vector Machine, Bayesian Network, and Naïve Bayes under WEKA environment for accuracy based on the evaluation of performance metrics. According to the research, the Bayesian Network provides the best accuracy with fewer feature-rich datasets whereas Support Vector Machine provides the best accuracy with more feature-rich datasets.

Khorshid, Abdulazeez, et al. [2] compared the performance of classifiers that included Support Vector Machine, Logistic Regression, K-Nearest Neighbors, Weighted K- Nearest Neighbors, and Gaussian Naïve Bayes to classify breast cancer in women based on their accuracy. The UCI machine learning repository was used to obtain the dataset. According to the results of the study, it was concluded that Weighted K- Nearest Neighbor (96.7%) has the highest accuracy of all the classifiers.

Derangula, Karri, et al. [3] aimed to find the optimal features in Wisconsin breast cancer diagnostic data as machine learning algorithms perform well with optimal features. The features were chosen using the Light Gradient Boosting Model, Catboost, and Extreme Gradient Boosting approaches. The Naïve Bayes classifier received the improved features, and it achieved an accuracy of 96.49%.

Gupta and Garg [4] described the use of six algorithms for supervised machine learning, including Logical Regression, Decision Tree, K-Nearest Neighbor, Random Forest, and Support Vector Machine with a kernel using the radial basis function. Adam Gradient Descent Learning was also used for deep learning. Unique hyperparameter adjustments were made to each model to improve accuracy and the results were compared among the models. The results demonstrate 98.24% accuracy with the least amount of loss was achieved using deep learning utilizing Adam Gradient Descent Learning.

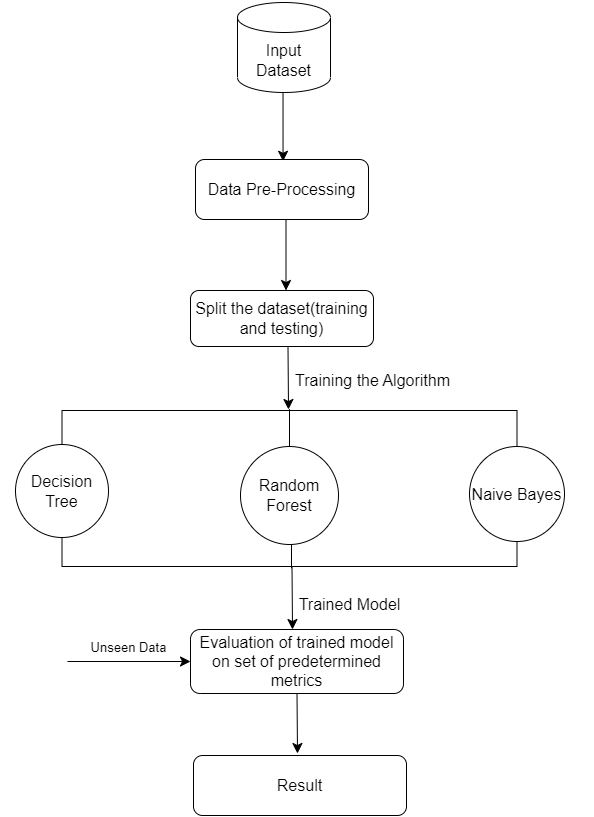
Li and Chen [5] used different classification models including Logistic Regression, Support Vector Machine, Neural Network, Random Forest, and Decision Tree. These models were applied to the two datasets, Wisconsin Breast Cancer Database (WBCD) and the Breast Cancer Coimbra Dataset (BCCD). The performance of the models was evaluated using three metrics, prediction accuracy, F-measure, and AUC values. Comparative experiment analysis conducted by them found that the Random Forest model performed better than the other four models. However, as their study only collects data on 10 attributes, they found out that the limited raw data affects the accuracy of results.

Obaid, Mohammed, et al [6] analyzed three machine learning classifiers Support Vector Machine, K-nearest Neighbors, and Decision Tree to determine which classifier performs best at classifying breast cancer. Additionally, this study made use of the Wisconsin Breast Cancer (Diagnostic) dataset with a primary goal to compare various classifiers and choose the one that provides the best accuracy. The result of this investigation presented that the quadratic Support Vector Machine provided the highest accuracy (98.1%) and the lowest false discovery rates. MATLAB was used to conduct and manage the experiments for this study.

## CHAPTER 3: METHODOLOGY

#### 3.1 Conceptual Framework

This section explains how the seminar is done. It lays out the steps and methods used to achieve the seminar's goals, including how data were collected and analyzed, what tools were used, and what algorithms were used. In this section, the various classifiers used in machine learning have been introduced.

[[1]](#endnote-7880)

**Figure 1: Conceptual Framework**

#### 3.2 Data Set Description

The data for this study is taken from Breast Cancer Wisconsin (Diagnostic) Dataset, UCI machine learning repository via Kaggle [11] named “data.csv”. Features describe the characteristics of the cell nuclei present in the image calculated from a digital image of a breast mass by fine needle aspirate (FNA). There are 569 cases available of which 357 are benign and 212 are malignant. The sample dataset instance is in the appendix of this report.

The attributes of the dataset are: - Id number and Diagnosis (Malignant =M and Benign =B) which further consist of ten real-valued features described as:

1. Perimeter,
2. Area,
3. Radius: mean of distances from the center to points on the perimeter,
4. Smoothness: local variation in radius lengths,
5. Texture: standard deviation of grey-scale values,
6. Compactness = perimeter^2 / (area - 1.0),
7. Symmetry,
8. Concave points: number of concave portions of the contour,
9. Fractal dimension = (“coastline approximation" - 1).
10. Concavity: severity of concave portions of the contour,

Further, standard error, the mean, and worst (mean of the three largest values) of these has been calculated, resulting in 30 features. Based on the above-mentioned features this system predicts whether a person has breast cancer or not.

#### 3.3 Data Preprocessing

After the data has been collected, pre-processing was done to remove any undesirable data and replace missing traits with mean and unstructured data was converted into structured data. The dataset was first split into a dependent and independent dataset which was further split into 80% for training and 20% for testing. Feature scaling was performed to check whether features have different scales or not.

#### 3.4 Algorithm

##### 3.4.1 Decision Tree

A decision tree is a technique used in supervised learning that can handle regression and classification problems, although it is commonly used for classification tasks. This algorithm uses a tree-like structure, where features in the dataset are represented as internal nodes, branches represent decision rules, and outcomes are indicated by each leaf node. The decision has multiple branches and nodes make decisions, while the leaf nodes are the final output without any further branches. The algorithm performs tests based on dataset features and provides a graphical representation of all possible solutions for a given decision problem, similar to a tree-like structure. The decision tree uses the CART algorithm, which stands for Classification, and the Regression Tree algorithm, which is used to construct the tree. Initially, the root node is created, and based on the answer to a question (Yes/No), the tree is divided into subtrees. While predicting the dataset's class the algorithm starts at the tree's root node and starts comparing the attribute value at the root with the corresponding attribute value in the actual dataset and moves down the branch based on the comparison result. Then it compares the attribute value at the next node with the corresponding attribute value in the dataset and continues to move down the appropriate branch until the leaf node in the tree is reached.

The steps in the decision tree algorithm are:

Step-1: Start the tree with the root node, says S, which contains the dataset.

Step-2: Attribute Selection Measure (ASM) is used to find the best attribute.

Step-3: Divide the root node(S) into subsets based on best attributes.

Step-4: Create a decision tree node that includes the most suitable attribute.

Step-5: Recursively create new decision trees for each subset of data. This continues until we cannot classify any further, and we end up with a final node called a leaf node.

**Attribute Selection Measures**

The main issue that arises while implementing DT is selecting the optimal attribute for the root node and subsequent nodes. To address this problem there is a technique known as Attribute Selection Measure (ASM). With this approach, the best attribute is easily selected for the nodes of the tree. ASM involves two well-known techniques: Information Gain and Gini Index.

**1. Information Gain:**

Information gain measures how much the entropy (or disorder) in a dataset change when it is split based on a particular attribute. It tells us how much information a feature gives us about a particular class. The decision tree algorithm aims to maximize information gain when splitting a node, and the attribute having the highest information gain is split first. Information gain is calculated as:

Information Gain= Entropy(S) - [(Weighted Avg) \*Entropy (each feature)]

Entropy: Entropy is a measurement of the impurity present in a given attribute. It describes randomness in data. The higher the entropy value, the more disordered the dataset is, indicating a greater degree of unpredictability in the outcomes. Entropy is calculated as:

Entropy(s) = -P (yes) log2 P (yes) - P (no) log2 P (no)

Where,

P (yes) = probability of yes

P (no) = probability of no

S= Total number of samples

**2. Gini Index:**

The Gini index is technique to measure impurity or purity used while constructing a decision tree in the CART. An attribute of having a low Gini index is preferred to that having a high Gini index. The Gini index can be calculated using the below formula:

Gini Index= 1- ∑jPj2



**Figure 2: Decision Tree**

Decision tree begins by initializing with configurable parameters such as min\_samples\_split, max\_depth, and n\_features, along with a placeholder for the tree's root node. During training (fit method), the tree recursively grows using the \_grow\_treefunction, which evaluates stopping criteria like maximum depth and minimum samples per split. Splits are determined based on features that maximize information gain, calculated using \_best\_split and \_information\_gain methods. Entropy is used to measure data purity and guide these splits, ensuring nodes are split optimally to maximize classification accuracy. Prediction (predict method) utilizes tree traversal (\_traverse\_tree) to navigate from the root to leaf nodes, where final class predictions are made based on the majority label in each leaf.

##### 3.4.2 Random Forest

The supervised learning method Random Forest is used to resolve Classification and Regression issues. In random forest, multiple decision trees are constructed of various subsets of the given dataset and take an average for regression and majority votes for classification to enhance the accuracy of the dataset which is the bagging method of ensemble learning. Ensemble learning assembles several classifiers to address a complex issue and the random forest is based on this.

It uses two methods:

1. **Bagging**

Bagging is the method of generating different training subsets from a sample training dataset by selecting data with replacement. The final decision is made by a majority vote.

1. **Boosting**

Boosting is a technique for constructing strong learners by creating sequential models from weak learners such that the final model has the highest possible accuracy.

Random Forest uses a technique called Bagging, which is also known as Bootstrap Aggregation. The process starts by taking a random sample of the original data, which is called a Bootstrap Sample. Then, models are trained individually using these samples, resulting in different outcomes. These outcomes are then combined, and the final output is determined by majority voting. This combination of results is called Bagging and is achieved using an Ensemble Classifier.

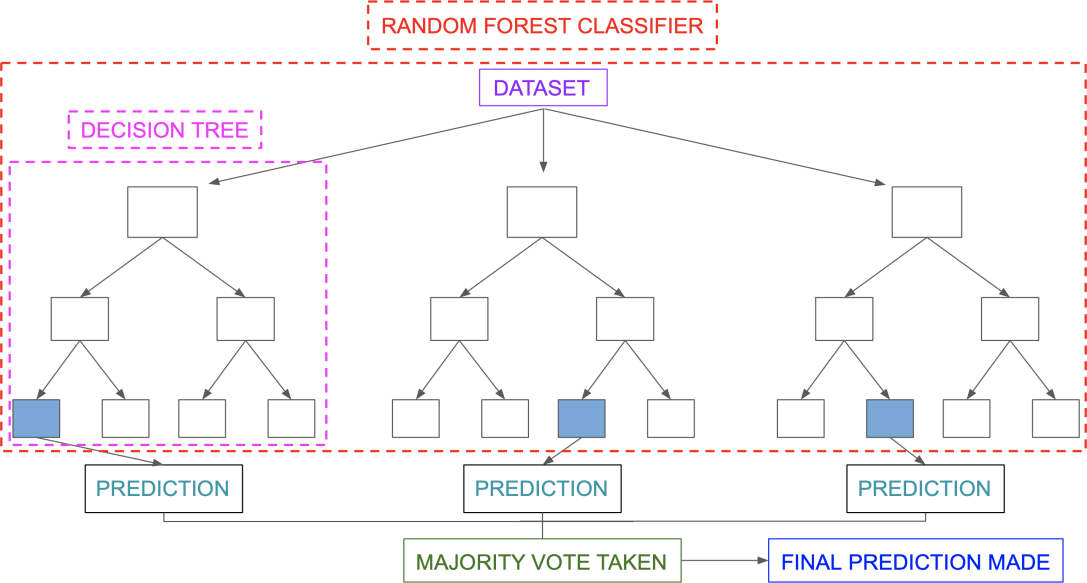
Steps involved in the random forest algorithm:

Step 1: The Random Forest uses n random records chosen at random from the data set's k records.

Step 2: A separate decision tree is constructed for each sample.

Step 3: Each decision tree will generate its output.

Step 4: Final output is determined either by Majority voting or Average depending upon whether the task is the Classification or regression respectively.



**Figure 3: Random Forest**

Here random forest is initialized with parameters such as n\_trees (number of trees), max\_depth (maximum depth of each tree), min\_samples\_split (minimum samples required to split a node), and n\_features (number of features considered for each split), it maintains a list of decision trees (trees). The fit method trains each decision tree by bootstrap sampling from the dataset (\_bootstrap\_samples), ensuring diversity among trees. Each tree is trained using the DecisionTreeclass with specified parameters. Prediction (predict method) involves aggregating predictions from all trees, determining the most common prediction for each sample among the ensembles.

##### 3.4.3 Naive Bayes

Based on probability theory, Naive Bayes is a popular probabilistic classification technique in machine learning. It is founded on the Bayes theorem, which offers a formula for estimating the likelihood of a hypothesis given its prior probability and the probability of the available data. In the context of classification, the hypothesis corresponds to a class, and the evidence corresponds to a set of features that describe an input instance.

The "naive" assumption in Naive Bayes refers to the assumption that the features are conditionally independent given the class. This means that the probability of observing a particular combination of features given a class can be calculated by multiplying the individual probabilities of each feature given the class. This assumption allows for efficient calculation of probabilities, as the joint probability distribution of all the features can be reduced to a product of individual probabilities.

The formula for calculating the posterior probability of a class y given the features x is:

P (y|x) = P(y) \* P (x|y) / P(x)

Where:

P(y) is the prior probability of class y.

P (x|y) is the conditional probability of the features x given class y.

P(x) is the probability of the observed features x, which is a normalization constant that ensures the posterior probabilities sum to 1.

Workings of the Naive Bayesian classifier are as follows:

Let D be a training set of tuples & their associated class labels and X={x1, x2……xn} be a tuple that is to be classified based on D.

To predict the class label of x, perform the following calculations for each class Ci.

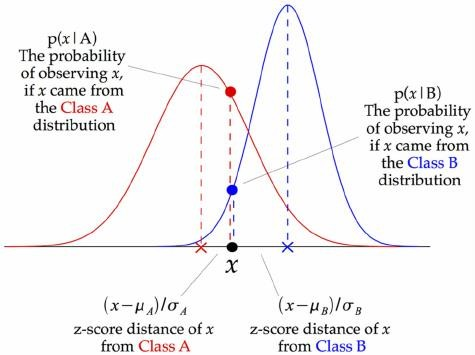
**Step 1:** Calculate the probability of each class ci as p (ci) = |Ci, D|/|D|, where |Ci, D| is the number of training tuples in D that belong to class Ci.

**Step 2:** For each value of attributes in X calculate P (Xk |Ci) as P (Xk | Ci) =no. of tuples of class Ci in D having the value Xk for attribute Ak/|Ci, D|.

**Step 3:** Calculate the probability of a tuple X conditioned on class Ci as P (X|Ci) = P (X1|Ci)\* P (X1|Ci) \* P (X2|Ci)\*……\* P (Xn|C).

**Step 4:** Calculate the probability of class Ci by conditioning on X using the formula

P (Ci|X) =P (X|Ci) \*P (Ci). Then, Predict the class label of X by choosing the class Ci for which the product of P (X|Ci)\*P (Ci) is maximum.

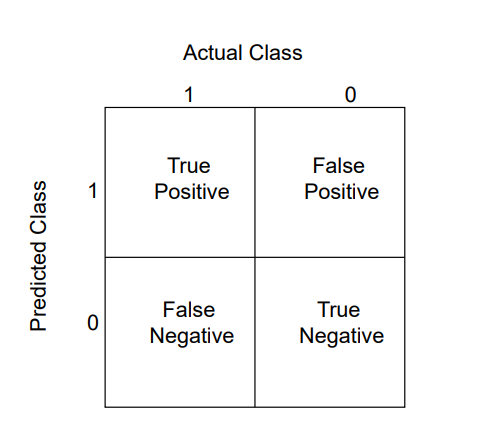


**Figure 4: Naive Bayes**

#### 3.5 Result Evaluation Metric

##### 3.5.1 Confusion Matrix

It is a table showing the number of true positive, true negative, false positive, and false negative predictions made by the model.



**Figure 5: Confusion Matrix**

Where,

True positive (TP) - refers to a case in which the model correctly predicts the positive class. False positive (FP) - refers to a case in which the model incorrectly predicts the positive class. True negative (TN) - refers to a case in which the model correctly predicts the negative class. False negative (FN) - refers to a case in which the model incorrectly predicts the negative class.

**For this system:**

The positive class is "malignant", and the negative class is "benign"

If the model correctly identifies a breast tumor as malignant, it is a true positive (TP).

If the model incorrectly identifies a breast tumor as malignant, when it is benign, it is a false positive (FP).

If the model correctly identifies a breast tumor as benign, it is a true negative (TN).

If the model incorrectly identifies a breast tumor as benign, when it is malignant, it is a false negative (FN).

##### 3.5.2 Accuracy

This metric calculates the percentage of accurate predictions made by the model and is calculated as:

Accuracy = number of correct predictions / total number of predictions.

i.e. (TP+TN)/ (TP+TN+FP+FN)

##### 3.5.3 Precision

This metric measures the proportion of true positive predictions made by the model, and is calculated as Precision = number of true positive predictions) / (number of true positive predictions + number of false positive predictions).

i.e. TP/ (TP+FP)

##### 3.5.4 Recall

This metric calculates the ratio of true positive predictions made by the model to the total number of actual positive cases and is calculated as

Recall = no. of true positive predictions / (no. of true positive predictions + no. of false negative predictions).

i.e. TP/(TP+FN)

##### 3.5.5 F1-score

This is the harmonic mean of precision and recall and is calculated as

F1 score = 2 \* (precision \* recall) / (precision + recall).

##### 3.5.6 AUC-ROC

By graphing the two parameters True Positive Rate and False Positive Rate, the area under the receiving operating characteristic curve is used to assess how well binary classification models perform at all classification thresholds. Better models have more curve area under them.

## CHAPTER 4: IMPLEMENTATION AND **RESULT ANALYSIS**

#### 4.1. Implementation

The program is developed using Python (version 3.12.4) as the primary programming language within the VS Code Integrated Development Environment (IDE). Several essential libraries are employed:

NumPy: Utilized for efficient numerical operations and computations

Pandas: Employed for comprehensive data manipulation and analysis tasks

Matplotlib: Utilized for creating static, animated, and interactive visualizations in

Python.

**Seaborn:** Used for statistical data visualization, emphasizing attractive and informative

statistical graphics.

Together, these libraries enable robust development, analysis, visualization, and deployment of machine learning and deep learning solutions in Python

Exploratory data analysis (EDA) techniques are employed to gain insights into the dataset's structure. Pair plots and correlation matrices generated with seaborn provide visualizations of feature distributions and pairwise relationships. Heatmaps illustrate the correlations between the first few features, aiding in understanding potential dependencies among them.

The Decision Tree model is implemented with a custom Decision Tree class, defining methods for model fitting, tree growth, best split determination, and prediction. The Random Forest model is built upon this Decision Tree implementation, where multiple trees are trained independently using bootstrap samples of the dataset, with predictions aggregated through voting. The Naive Bayes model, implemented using a Naive Bayes class, calculates class-wise means, variances, and priors during training and makes predictions based on the maximum posterior probability.

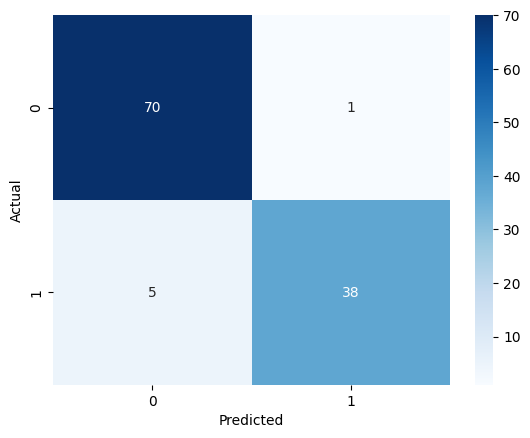
model performance metrics such as accuracy, precision, recall, and F1-score — are computed using confusion matrices from sklearn metrics. Heatmaps visualize these matrices, providing insights into model classification performance across different classes. The model's receiver operating characteristic (ROC) curves are plotted to evaluate their performance in terms of true positive rates against false positive rates. This comprehensive analysis enables a comparative assessment of the strengths and weaknesses of each algorithm in the context of diagnosing breast cancer based on the dataset's features.

#### 4.2. Findings And Result Analysis

##### 4.2.1. Findings

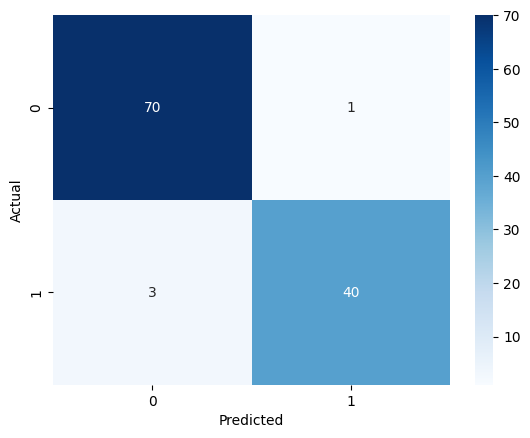
The study results showed that all three machine learning algorithms had high evaluation scores ranging from 91% to 98%. However, the Random Forest algorithm had the highest accuracy rate, followed by the Naive Bayes and Decision Tree algorithms. The Random Forest algorithm also had the highest precision and recall scores, indicating that it had the lowest number of false positives and false negatives. The analysis of the study revealed that the Random Forest algorithm had the best performance in predicting breast cancer based on patient data. This is due to the algorithm's ability to handle high-dimensional data and its ability to reduce over-fitting by combining multiple decision trees. The Naive Bayes algorithm also showed promising results, especially in scenarios with limited data. The Decision Tree algorithm, on the other hand, had a lower performance than the other two algorithms. This is due to its tendency to over-fit the data, resulting in many false positives and negatives.

##### 4.2.1.1 Comparison based on Confusion Matrix



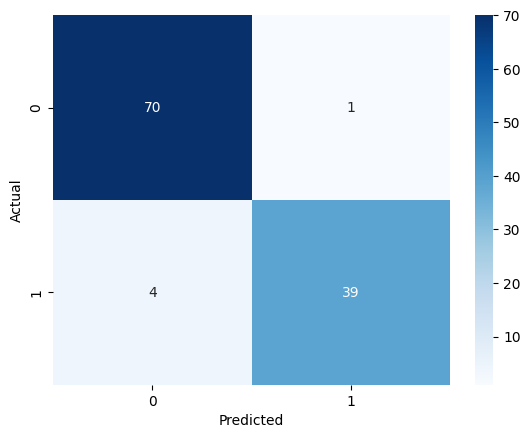
**Figure 6: Confusion matrix of Decision Tree**

Here, TP=70, FN=1, FP=5 and TN=38



**Figure 7: Confusion matrix of Random Forest**

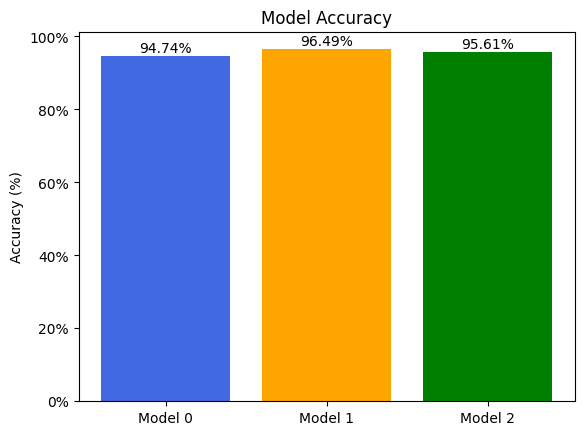
Here, TP=70, FN=1, FP=3 and TN=40



**Figure** 8: **Confusion matrix of Naive Bayes**

Here, TP=70, FN=1, FP=4 and TN=39

##### 4.2.1.2 Comparison based on Accuracy



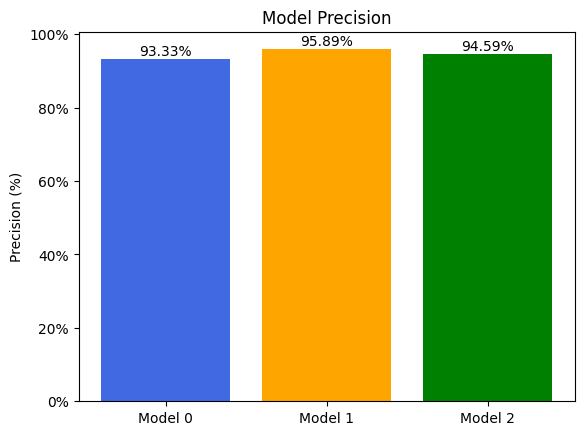
**Figure 9: Accuracy Score**

Here, from the above bar diagram, it is seen that Model 0 (Decision Tree), Model 1 (Random Forest), and Model 2 (Naive Bayes) have accuracy scores of 94.74%, 96.49%, and 95.61%, respectively. The findings for accuracy score is shown bellow:

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Decision Tree | Random Forest | Naive Bayes |
| Accuracy Score | 94.74% | 96.49% | 95.61% |

Based on the results, it can be concluded that the Random Forest model performs better than other models in terms of accuracy.

##### 4.2.1.3 Comparison based on Precision



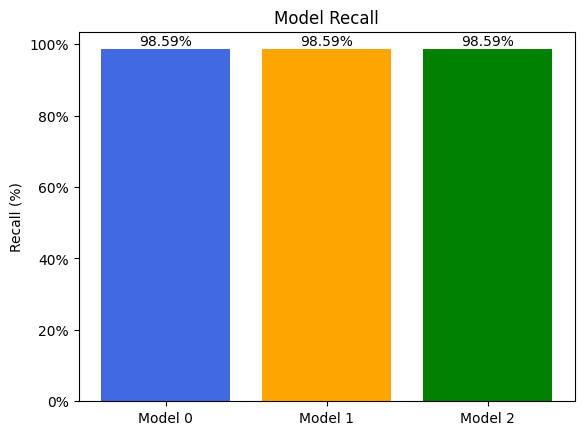
**Figure 10: Precision Score**

Here, from the above bar diagram, it is seen that the precision of model 0 (Decision Tree), model 1 (Random Forest), and model 2 (Naive Bayes) is 93.33%, 95.89%, and 94.59% respectively. The findings for precion score is shown bellow:

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Decision Tree | Random Forest | Naive Bayes |
| Precision Score | 93.33% | 95.89% | 94.59% |

Based on the result it can be concluded that the Random Forest model performs better on the precision metric.

##### 4.2.1.4 Comparison based on Recall



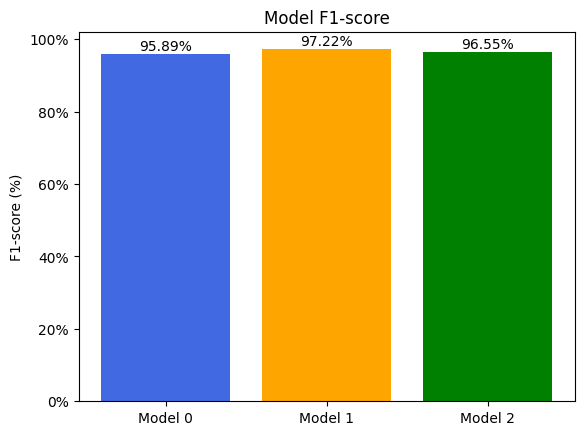
**Figure 11:Recall Score**

Here, from the above bar diagram, it is seen that the recall of model 0 (Decision Tree), model 1 (Random Forest), and model 2 (Naive Bayes) is 98.59%, 98.59%, and 98.59% respectively. . The findings for Recall score is shown bellow:

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Decision Tree | Random Forest | Naive Bayes |
| Recall Score | 98.59% | 98.59% | 98.59% |

Based on the result it can be concluded that all three models perform similarly on the recall score

##### 4.2.1.5 Comparison based on F1-score



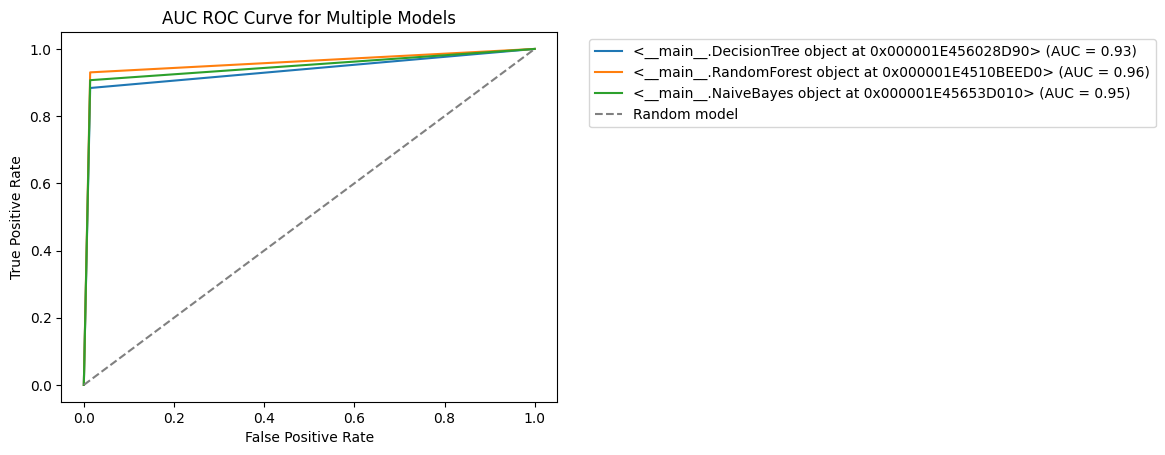
**Figure 12: F1-Score**

Here, from the above bar diagram, it is seen that the F1-score of model 0 (Decision Tree), model 1 (Random Forest), and model 2 (Naive Bayes) is 95.89%, 97.22%, and 96.55% respectively. The findings for F1-score is shown bellow:

|  |  |  |  |
| --- | --- | --- | --- |
| Model | Decision Tree | Random Forest | Naive Bayes |
| F1 Score | 95.89% | 97.22% | 96.55% |

Based on the result we can conclude that the Random Forest model performs best on the F1-score metric.

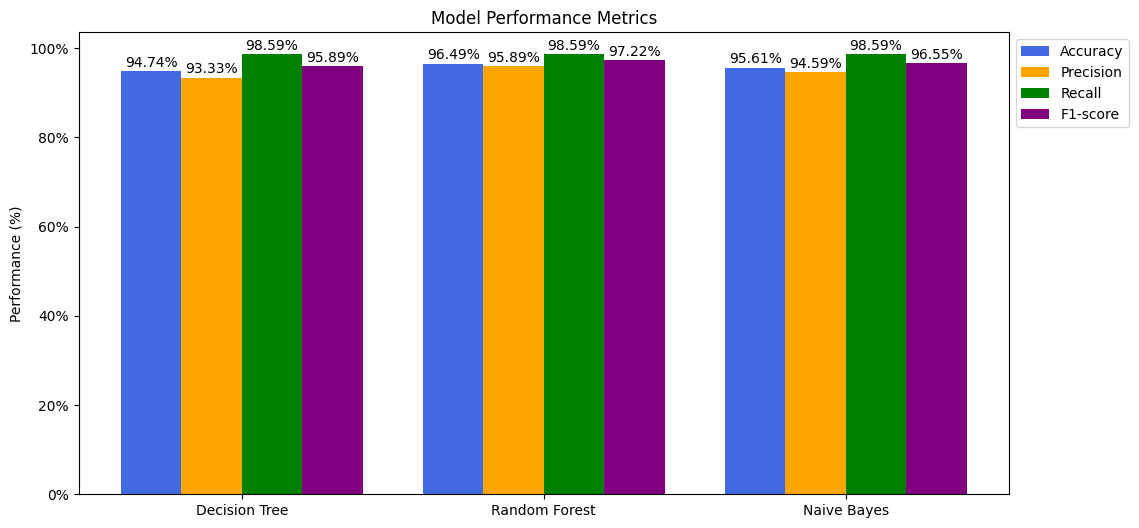
##### 4.2.1.6. Comparison based on the AUC-ROC curve



**Figure 13: AUC-ROC Curve**

The AUC-ROC (Area Under the Curve of the Receiver Operating Characteristic) curve visually depicts the performance of the machine learning algorithm. It illustrates the relationship between the True Positive Rate (TPR) and False Positive Rate (FPR) for varying threshold values where, TPR represents the proportion of actual positive cases that are correctly classified as positive by the algorithm, while the FPR represents the proportion of actual negative cases that are incorrectly classified as positive by the algorithm. The closer the AUC-ROC score is to 1.0, the better the algorithm's performance in distinguishing between the two classes. Based on the AUC-ROC scores, the Random Forest algorithm appears to have the highest performance with an AUC-ROC score of 0.96, followed by Naive Bayes with an AUC- ROC score of 0.95, and then Decision Tree with an AUC-ROC score of 0.93.

##### 4.2.2 Result Analysis



**Figure 14: Multi-bar graph for each Model**

Here, from the above multi-bar graph, it can be seen that Random Forest has the highest accuracy, precision, and recall with a score of 96.49%, 95.89%, and 97.22 % respectively followed by Naive Bayes with a score of 95.61%, 94.59%, 96.55% respectively and that of Decision tree with the score of 94.74%, 93.33%, and 95.89% respectively. However, the recall score for all three models is the same as a score of 98.59%.

## CHAPTER 5: CONCLUSION AND FUTURE WORK

#### 5.1 Conclusion

This study aims to implement and compare Decision Tree, Random Forest, and Naive Bayes algorithms for breast cancer prediction. The goal is to identify the most effective algorithm for early detection based on available data and clinical relevance, contributing to improved patient outcomes in breast cancer management. In this study, performance evaluation metrics like accuracy, precision, recall, F1-score, and AUC-ROC were computed based confusion matrix for the breast cancer dataset and the Random Forest algorithm has been found to outperform the Decision Tree algorithm and Naive Bayes. The Random Forest model outperforms the other models in terms of accuracy, with an accuracy score of 96.49% also performs best on the precision metric, with a precision score of 95.89%, and in terms of F1-score with an F1-score of 97.22%. Random Forest algorithm even appears to have the highest performance with an AUC-ROC score of 0.96, followed by Naive Bayes with an AUC- ROC score of 0.95, and then Decision Tree with an AUC-ROC score of 0.93. Moreover, all three models performed similarly in terms of recall, with a score of 98.59%. Therefore, based on these evaluation metrics, it can be concluded that Random Forest is the best model that classifies the malignant and benign cases in this dataset compared to Decision Tree and Naive Bayes.

#### 5.2 Future Work

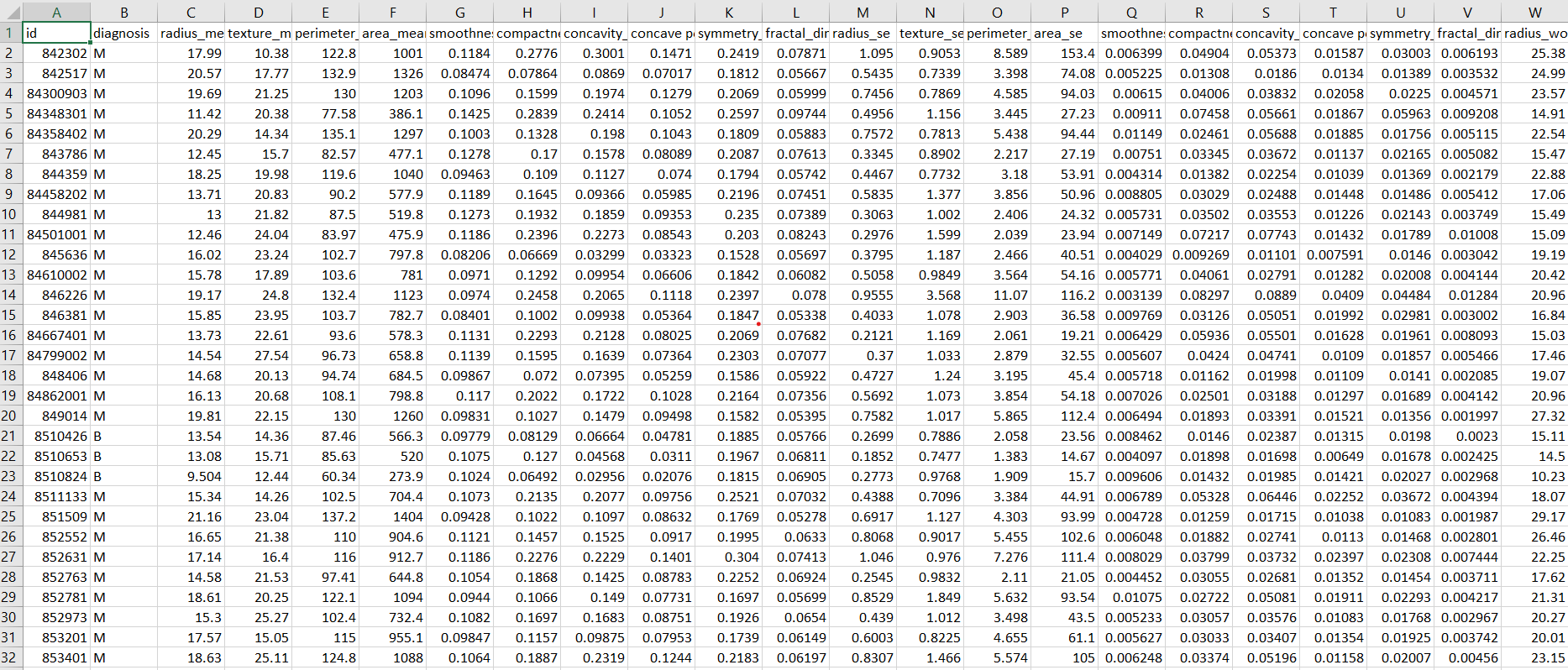
This study could be expanded further. Future work could focus on increasing the number of models evaluated or evaluating the same models on different datasets, working with large datasets in similar or different domains. Adding new features along with imaging features such as medical history, lifestyle factors, or environmental exposure, patient age, family history of breast cancer, shape irregularity, tumor size, etc. could be obtained and included as new features in the dataset. Additionally, other machine learning algorithms or ensemble models could be explored and compared with the existing models to recognize ways to get better results.

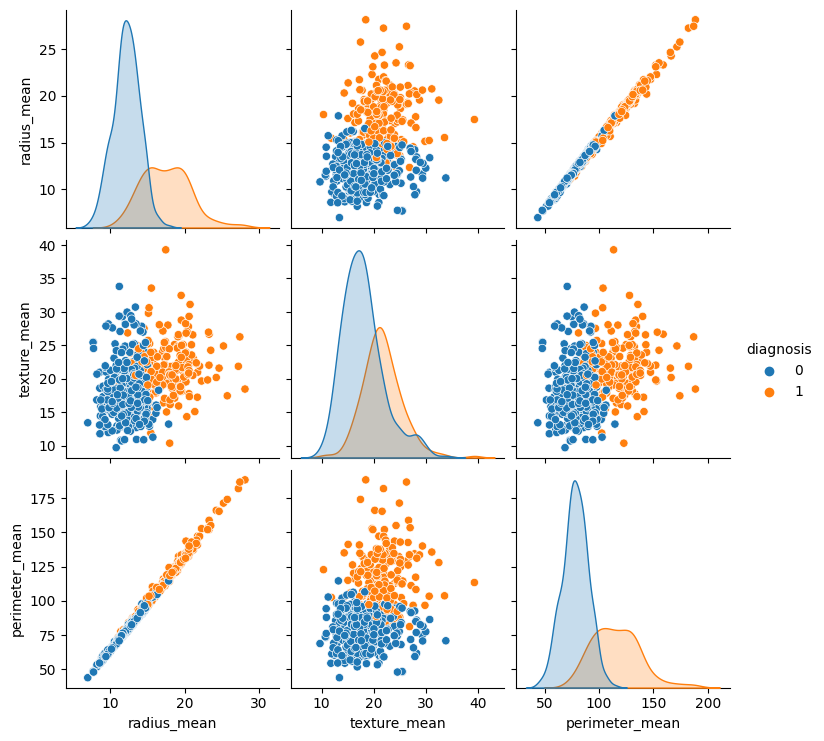
## References

|  |  |
| --- | --- |
| [1] | A. Kumar, R. Sushil and A. Tiwari, "Comparative Study of Classification Techniques for Breast Cancer," *International Journal of Computer Sciences and Engineering,* vol. 7, no. 1, pp. 234-240, 2019. |
| [2] | S. F. Khorshid, A. M. Abdulazeez and A. B. Sallow, "A Comparative Analysis and Predicting for Breast Cancer Detection Based on Data Mining Models," *Asian Journal of Research in Computer Science,* vol. 8, no. 4, pp. 45-59, 2021. |
| [3] | A. Derangula, P. S. Edara and P. K. Karri, "Feature Selection of Breast Cancer Data Using Gradient Boosting Techniques of Machine Learning," *European Journal of Molecular & Clinical Medicine,* vol. 07, no. 02, pp. 3488-3504, 2020. |
| [4] | P. Gupta and S. Garg, "Breast Cancer Prediction using varying Parameters of Machine Learning Models," *Procedia Computer Science,* vol. 171, pp. 593-601, 2020. |
| [5] | Y. Li and Z. Chen, "Performance Evaluation of Machine Learning Methods for Breast Cancer Prediction," *Applied and Computational Mathematics,* vol. 7, no. 2, pp. 212-216, 2018. |
| [6] | O. I. Obaid, M. A. Mohammed, M. K. A. Ghani, S. A. Mostafa and F. T. AL-Dhief, "Evaluating the Performance of Machine Learning Techniques in the Classification of Wisconsin Breast Cancer," *International Journal of Engineering & Technology,* vol. 7, no. 4.36, pp. 160-166, 2018. |
| [7] | R. Rawal, "BREAST CANCER PREDICTION USING MACHINE LEARNING," *Journal of Emerging Technologies and Innovative Research (JETIR),* vol. 13, no. 24, p. 7, 2020. |
|  |  |
|  |  |

## Appendix

Screenshot of dataset:



Screenshot of Data Visualization:

1. [↑](#endnote-ref-7880)