Using Genetic Algorithm for Breast Cancer Feature Selection

Abstract:

Breast cancer has been one of the most widespread cancer types in women worldwide. Breast cancer can be treated when detected early; otherwise, it has one of the highest mortality rates among cancer types. Many tools can be used for detection, but computer-based diagnosis systems have become popular as they are cheaper and quicker. This brings incorrect detections as well. Hence, feature selection is an important factor that can enhance the accuracy of computer-based programs. This study uses genetic algorithms for feature selection within a wrapper methodology for breast cancer diagnosis. The proposed model has been tested with 17 different classifiers in order to evaluate its effectiveness. There has been an increase in training accuracy after feature selection was employed with genetic algorithms. The highest training accuracy was reported in Extra Trees, MLP, Random Forest, and Logistic at 100%, and the lowest was reported in GaussianNB at 0.925. Furthermore, feature selection improved validation accuracy, sensitivity, specificity, F1-score, Matthews Correlation Coefficient, specificity, and sensitivity.

Keywords: Genetic Algorithm, Feature Selection, Breast Cancer, Machine Learning Classifiers, Random Forest

1. Introduction:

The rapid advancements in technology-enabled datasets to grow and become publicly available. Datasets have thousands of features; as a result, researchers use advanced pattern detection methods to decipher these rich samples. Datasets are created for many fields, from biology to astrophysics, encompassing thousands of features. On the other hand, features can be either redundant or irrelevant, so some do not contribute to machine learning and deep learning models. Thus, as datasets have many features, it is vital to reduce the dimensionality of datasets and extract important features [1, 2, 3]. By feature reduction, the efficiency of the classification model can greatly increase. The selection of features is crucial, but the characteristics of the dataset should not be lost. Hence, it is important to classify the features as weak and strong [4]. This process is also known as data mining. Data mining has been used in the medical domain in order to find a relationship as it may be challenging for medical experts to diagnose with a welter of data [5]. Automated diagnostic systems are one of the fields where database analysis is applied the most.

Breast cancer is the most prevalent invasive cancer among women [6]. Around 2.3 million women worldwide are diagnosed with breast cancer yearly [7]. Most patients with brain cancer are over fifty years old [7]. Early detection of breast cancer has been proven to be very effective in reducing the mortality rate of patients [7]. However, the survival rate depends on many factors, especially stage and molecular subtypes [7].

Classical methods exist in the detection of breast cancer, such as biopsy, physical testing, breast ultrasound, diagnostic mammogram, and Breast magnetic resonance imaging (MRI) [8]. Once breast cancer is detected, additional tests are conducted to determine if the cancer cells have spread within the breast or to areas of the body. This process is known as staging. The stage of breast cancer is determined by whether it is confined to the breast, has reached the lymph nodes in the armpit, or has extended beyond the breast. Based on the type and stage of breast cancer, doctors can determine the treatment required for you [8].

Mammography is one of the most common methods for breast cancer detection, which is utilized by radiologists. However, radiologists may interpret the results differently or inaccurately; thus, the accuracy rate of mammography fluctuates between 68% and 79% [9, 10]. Another way is a biopsy, which can be expensive, risky, invasive, but accurate. These detection techniques can categorize patients into a 'benign' group without breast cancer or a 'malignant' group showing substantial signs of the disease [10]. Also, it is essential to note that benign tumors are safer than malignant tumors in many cases. Computer-aided systems may help doctors to understand the differences between these two categories. As mentioned before, feature selection, a preprocessing technique, may help doctors in breast cancer detection/classification.

Three approaches to feature selection are Filter, Wrapper, and Embedded [11, 12]. The filter approach scores the selected subset based on the intrinsic properties of the data without considering the classifier algorithm [13]. The wrapper method finds the best set of features for a specific algorithm and area [14]. In other words, the chosen set of features is determined by training and assessing a classifier using only the variables within the suggested group. The optimal feature subset of features is selected during the model-building process in the embedded approach [11].

Numerous studies in the literature have utilized a range of feature selection techniques on breast cancer datasets. These techniques encompass the ant colony algorithm, discrete particle swarm optimization, the wrapper strategy combined with a genetic algorithm, feature selection rooted in support vectors, incorporating fisher's linear discriminate and support vector machine, the rapid correlation-based feature selection (FCBF), its multi-threaded version, the decision-dependent and -independent correlation (DDC- DIC), the Rough set K-Means Clustering method, and the adjusted correlation rough set feature selection approach (MCRSFS) [11, 15-21].

1.1.Short Literature Review:

Large databases created by improvements in facilities can be gathered by the medical industry, which needs to find hidden links in data. For these objectives, data mining techniques are heavily utilized in the medical field [22a]. Automated diagnostic systems are one of the applications of database analysis. These programs can aid doctors in making choices. Finding strategies to enhance clinical research, lower costs, and improve patient outcomes is another application. Additionally, there has never been a greater need for automated diagnosis than in the case of fatal diseases like cancer, when early discovery can significantly increase patients' prospects of long-term survival and lower expenditures. The most prevalent invasive malignancy in women is thought to be breast cancer. It is ranked as the second leading cause of death for women in the USA and the leading cause of death for women between the ages of 40 and 55. Early detection has been shown to significantly lower mortality rates among breast cancer patients [23a]. Physical examination, mammography, and biopsy techniques such fine needle aspiration biopsy (FNAB or FNAC), core needle biopsy, surgical biopsy, and lymph node biopsy are the three traditional approaches for finding breast cancer.

One of the most used procedures for finding breast cancer is mammography. Radiologists' interpretations of mammograms vary widely in the literature. Mammography accuracy ranges from 68% to 79%. When a tumor is found by mammography, a biopsy is necessary to determine its malignancy. Although surgical biopsy is accurate almost 100% of the time, it is expensive, invasive, time-consuming, and uncomfortable. FNAC is frequently used to diagnose breast

cancer. Depending on the doctor's experience, the accuracy of FNAC with visual interpretation ranges from 35% to 95% . Therefore, it is essential to create improved detection techniques to detect breast cancer. With the aid of these diagnostic techniques, patients can be categorized into two groups: those who are "benign," meaning they do not have breast cancer, and those who are "malignant," meaning they clearly have. Generally speaking, malignant tumors are more severe than benign tumors. As already indicated, the odds of successfully treating breast cancer are significantly increased by early identification. To do this, it is essential to have high levels of precision and dependability in diagnostic technologies that enable medical professionals to differentiate between benign and malignant breast tumors [22a].

The abundance of characteristics in diagnostic systems is one issue. These features' irrelevance and redundancy worsen the classification algorithm's perplexity and reduce learning precision. One approach that can address this issue and is crucial to classification is feature selection. One of the pre-processing methods used in data mining, feature selection is widely employed in the domains of statistics, pattern recognition, and the medical industry [22a].

Wrapper, Filter, and Embedded are the three methods for choosing features [23a]. By training and analyzing a classifier using only the variables included in the suggested subset, the wrapper approach determines the goodness of a chosen subset of features. The filter strategy ignores the classification algorithm and employs some strategies to score the chosen subset. In other words, the data's intrinsic qualities alone were used to determine the goodness of a chosen subset of features. In the embedded technique, the best subset of features is chosen while building the model. The literature contains a significant amount of research on breast cancer datasets using feature selection methods, including the ant colony algorithm, discrete particle swarm optimization method, wrapper approach with genetic algorithm , support vector-based feature selection using fisher's linear discriminate and support vector machine, fast correlation based feature selection (FCBF), multi thread based FCBF feature selection, and decision dependent-decision independent [24a].

In the study, we adopted a wrapper feature selection method derived from a genetic algorithm. By employing 17 different classifiers, the impact of the genetic algorithm on the accuracy of these classifiers using the breast cancer dataset was examined.

2. Dataset

The dataset was taken from the UC Irvine machine learning repository [23]. Characteristics of cell nuclei are derived from a digitized image taken from a fine needle aspirate (FNA) of a breast mass. Dr. William H. Wolberg has collected the data between 1989 and 1991 [22]. This dataset used 569 patients' data along with 31 different features. There are 357 benign and 212 malignant patients.

Table 1: Possible Features for Feature Selection

Features		
Diagnosis	Fractional Dimension Mean	Radius Worst
Radius Mean	Radius Standard Error	Texture Worst
Texture Mean	Texture Standard Error	Perimeter Worst

Perimeter Mean	Perimeter Standard Error	Area Worst
Area Mean	Area Standard Error	Smoothness Worst
Smoothness Mean	Smoothness Standard Error	Compactness Worst
Compactness Mean	Compactness Standard Error	Concavity Worst
Concavity Mean	Concavity Standard Error	Concave Points Worst
Concave Points Mean	Concave Points Standard Error	Symmetry Worst
Symmetry Mean	Fractional Dimension Standard Error	Fractional Dimension Worst

Ten real-valued features are computed for each cell nucleus as it is seen in table 1.

3. Classifiers

In this study, 21 different machine learning classifiers have been used in this study to compare the effectiveness of genetic algorithms.

3.1 Extra Trees Classifier

There are many tree-based algorithms and models that exist; however, it is different as instead of using a bootstrap replica, it grows the trees using the entire learning sample and selects cut points for nodes entirely at random [23]. A random subset of attributes from the dataset is chosen for each decision tree. The dataset is then divided based on random divisions within those attributes, with the optimal split being selected [24]. The Extra-Trees classifier makes a set of decision trees using the usual top-down way [25]. One of the strengths of this algorithm is computational efficiency [23].

3.2 Adaboost Classifier

Adaboost, as its name suggests, is a boosting machine learning algorithm. It combines multiple weak learning models and a weighted linear combination [25]. AdaBoost applies a step-by-step learning method to adjust versions of the initial training data [26]. Adaboost operates iteratively, and when there are misclassified instances, more weight is given to other iterations. Weights of miscorrectly classified instances are raised/increased, but correctly classified instances are diminished [25]. The algorithm consistently uses the base classifier on the training data, altering weights in every cycle [25]. The final model is a linear combination of the models obtained from various cycles.

3.3 Random Forest Classifier

The Random Forest classifier is very similar to the extra trees classifier. The algorithm forms a group of decision trees to improve the decision trees' accuracy. Also, this classifier uses a random selection of features and a bagging sample method [25, 27-30]. Using bagging, every decision tree in the ensemble is formed from a resampled version of the training data. Each tree in the ensemble serves as a base estimator to establish the class label for an unlabeled sample, with the final decision made based on the majority of votes/average [25].

3.4 Bagging Classifier

The bagging classifier is one of the meta-estimators, creating models by fitting each base classifier on an arbitrary subsample of the dataset [25]. Afterward, it gathers the outcomes from all the models to make the final decision. The bagging classifiers use two different methods: the

highest average likelihood from the base classifiers and majority voting, which rules the suspicious nodes in the network to establish the predicted label [25].

3.5 Gaussian Naive Bayes Classifier

The naive Bayes technique is largely applied in machine learning models as it has a computational efficiency. This algorithm has a low variance value and a high cost of bias. It uses incremental learning, which means estimations can be updated. It operates on the premise that each individual parameter independently influences the outcome variable. It uses probabilities so thoughtless to noise [31].

3.6 Category Boosting Classifier

Category Boosting is a machine learning algorithm designed to handle categorical features. It is one of the gradient-boosting frameworks that build an ensemble of decision trees in a sequential manner. Moreover, for reducing overfitting, it has an algorithm to encode categorical features and uses L2 regularization.

3.7 LightGBM Classifier

Light Gradient Boosting (LightGBM) is a gradient-boosting framework using tree-based learning algorithms [32, 33]. It is effective in training large datasets and with high-dimensional features. Gradient-based One-Side Sampling helps to retain data instances with large gradients. Gradient-based one-sided sampling helps to maintain data instances with large angles. It randomly samples a small portion of data instances with small gradients, reducing the data used in each iteration with minimal loss in accuracy [32, 33]. Moreover, it can achieve faster training times compared to other gradient-boosting algorithms [32, 33].

3.8 Quadratic Discriminant Analysis(GDA)

Quadratic Discriminant Analysis (QDA) is a classification method used in statistics, probabilistic, and machine learning. QDA is a generalization of Linear Discriminant Analysis in order to handle where each class has its own covariance matrix rather than assuming a common covariance matrix for all classes [34, 35]. GDA is used for classifying data into multiple classes based on the maximization of the posterior class probability. For each class, it computes the likelihood of a data point belonging to that class based on its Gaussian distribution [34, 35]. Afterward, it multiplies this likelihood by the prior probability of that class. The class with the highest posterior probability is the predicted class for the data point. The decision boundary is derived by setting the posterior probabilities of two classes to be equal.

3.9 Support Vector Machine Classifier

The Support Vector Machine (SVM) is a small-sample learning method. Also, it is a supervised machine learning algorithm used for classification tasks and regression [36]. It finds the hyperplane, decision boundaries, that best divides a dataset into classes. Thus, it is mostly accurate in separable and non-separable problems [37]. However, when the number of input features is more than 3, it has more than two-dimensional planes.

3.10 Linear Regression

Linear regression is a type of supervised machine-learning algorithm. When there is only one independent feature, it is termed Univariate Linear Regression. However, if there are multiple features, it is referred to as Multivariate Linear Regression. The algorithm aims to

identify the optimal linear equation to estimate the dependent variable's value using the independent variables.

3.11 Gradient Boosting Classifier

The Gradient Boosting Classifier is a popular machine learning algorithm used for classification and regression. It is an ensemble learning method. It converts weak learners into strong learners, and it was built stage-wise. It uses the decision trees as base learners. The algorithm is effective for classifying complex datasets. It is based on probability and approximately correct learning. The objective is to minimize the loss between the actual and predicted class values.

3.12 K-nearest neighbors Classifier

K-nearest neighbors are one of the most common classifiers used in machine learning. However, it is a very simple an effective algorithm for classification and regression. However, it just memorizes and uses the training set directly during the test set. Also, it is non-parametric, which means it does not make any assumptions about the underlying data distribution. There is a distance metric that measures the distance between data points using Euclidean distance. After this phase, the number of nearest neighbors to consider when making a classification decision. It classifies an unknown point based on the majority class among its k nearest neighbors.

3.13 Logistic Classifier

The logistic classifier is an algorithm that is used in machine learning and statistics. It is used for predicting a categorical outcome variable, and the outcome comes in the form of a binary outcome variable. It uses the sigmoid function in order to squeeze a linear equation between 0 and 1. Logistic regression creates a linear decision boundary in the feature space.

3.14 Passive Aggressive

The Passive-Aggressive Classifier is used for large-scale learning, text classification tasks, and multiple classes. It updates the model incrementally. It can quickly adapt to new data and does not need to store the dataset; in other words, it is memory-efficient. It uses a weight vector, which gives an idea of the importance of different features and creates a linear model.

3.15 Multinomial Naive Bayes

The Multinomial Naive Bayes algorithm is built on top of Naive Bayes and is useful for classification. It counts each occurrence of each class in the training data and calculates the probability of each class and the possible state of each given class. For additional data points, the algorithm computes each class, calculates the posterior probability, and assigns the class the highest probability. Furthermore, it does not require high memory and is still unsuitable for continuous features.

3.16 Decision Tree

Decision Tree is a hierarchical decision-support framework that uses a tree-structured representation of choices and their potential outcomes. It is a supervised machine-learning algorithm that can be used for regression and classification problems. It is one of the most interpretable machine learning algorithms, but it has a possibility to overfit, especially when the tree is deep. It uses a tree-like model for decision-making and splits the dataset into two or more homogeneous sets based on the most significant attribute at each level.

3.17 Multilayer perceptron

A multilayer perceptron is a multilayer neural network that has three hidden layers/neurons. Neurons utilize a nonlinear activation function and are uniquely capable of classifying data that is not linearly separable.

3.18 Stochastic Gradient Descent

Stochastic Gradient Descent is an optimization technique frequently employed in machine learning to identify the model parameters that yield the closest match between predicted and observed outcomes. For classification tasks, the algorithm utilizes a straightforward Stochastic Gradient Descent (SGD) learning process, which accommodates multiple loss functions and penalties.

4. Genetic Algorithm

Originating in the 1960s and 1970s by John Holland and his team, the genetic algorithm (GA) is an abstraction of biological evolution rooted in Charles Darwin's theory of natural selection [38]. It is likely that Holland pioneered the application of crossover and recombination, mutation, and selection in the exploration of adaptive and artificial systems, which will be discussed in this research paper [38]. As a problem-solving strategy, these genetic operators are pivotal to the genetic algorithm [38]. In the time since, numerous genetic algorithm variants have emerged, addressing a broad spectrum of optimization challenges, ranging from graph coloring and pattern recognition to both discrete and continuous systems [38].

In the evolutionary algorithms, genetic algorithms stand out due to the vast range of their applications. Using an iterative process, a Genetic Algorithm is employed for Search and Optimization to determine the best solution among multiple options. A Genetic Algorithm is essential in identifying the optimal hyperparameters and their values to enhance a deep learning model's performance and can be used to find the most suitable number of features when constructing machine learning models.

Some of the important terminology for genetic algorithms are population, phenotype, chromosome, and fitness score. Figure 1 shows the general structure of the genetic algorithms.

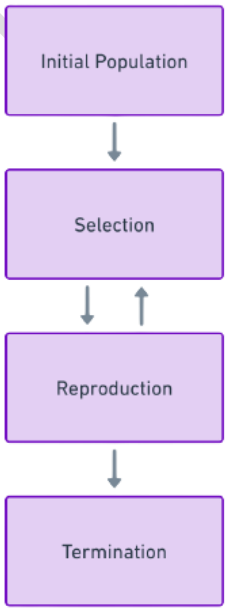


Figure 1: General structure of genetic algorithms

4.1 Initial Population

The Genetic Algorithm Process begins with Population Initialization. Within the current generation, the population represents a subset of solutions. Each individual possesses a gene sequence, often referred to as DNA. An individual's DNA signifies a potential solution to the targeted problem, and it must be structured appropriately. Hence, it is essential to initialize each individual to guarantee they possess some form of DNA. In the context of genetic algorithms, it is crucial to preserve the population's diversity to avoid an issue called premature convergence. This term in evolutionary algorithms refers to the algorithm settling before achieving the best possible solution. There are two ways of population initialization. The first one is random initialization, which initializes the population with completely random gene values. When gene values are randomly assigned, possible genetic diversity increases within the population. The second method is heuristic initialization, which uses heuristics to solve a complex issue.

4.2 Selection

The selection process is essential in genetic algorithms. Each individual has the fitness value of their corresponding DNA. The fitness value of an individual indicates its optimality, showing how close it is to the best solution compared to others. When the fitness function doesn't produce superior fitness values, the genetic algorithm might struggle to generate top-notch solutions. Once a proper fitness function is established, each individual's fitness is determined. The population is then organized based on these fitness levels, and a portion of those with the lowest fitness is removed. However, a few with lower fitness remain to maintain genetic variety within the group [39].

4.3 Reproduction: Crossover and Mutation

After the selection process, reproduction takes place. Reproduction happens through crossover and mutation. Crossover, which is simply the mating. Crossing over occurs when genes from the two most fit parents are mixed randomly to create a new solution or genotype. Depending on the segments of genes swapped from the parents, this can be a one-point or multi-point crossover. The primary goal of crossover is to produce new descendants from individuals with high fitness, thereby enhancing the population's overall fitness. Once a new population emerges from selection and crossover, it undergoes random alterations via mutation. Mutation serves as a random method to modify a genotype, fostering diversity within the population and aiding in discovering enhanced and more efficient solutions. The algorithm's search space broadens when enough new genes are introduced by randomly modifying the genes of the next generation's offspring.

4.4 Termination

This part is the last step of the genetic algorithm. When the genes of the next generation's offspring are randomly modified, the algorithm's search space expands due to the introduction of sufficient new genes. If the termination conditions are met, the evolutionary algorithm can be terminated, and output can be seen.

5. Methodology

The hardware used for this experiment has a 2,4 GHz Quad-Core Intel Core i5, 8 GB RAM 2133 MHz LPDDR3, and Tesla P100-PCIE GPU. Experiment has been performed using Python code. The breast cancer dataset was independently fed into 17 different classifiers. The proposed model (Figure 2) has been applied to the Wisconsin breast cancer dataset.

Flowchart of the working genetic algorithm and proposed model that is used in this study can be seen below:

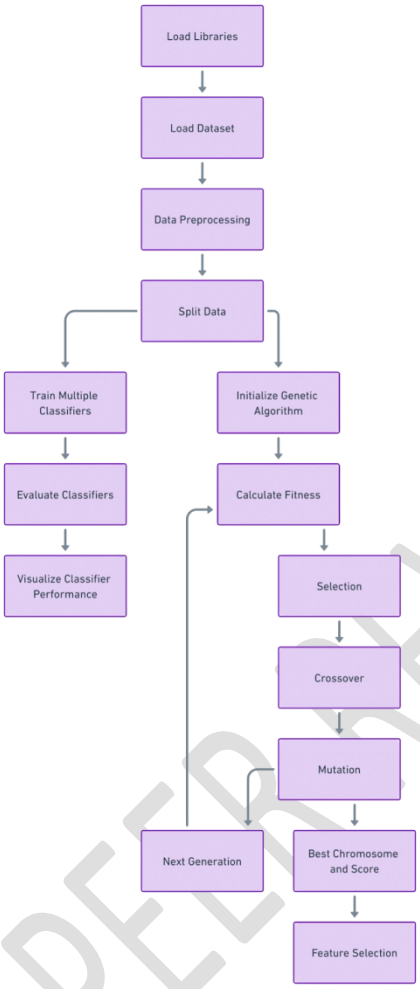


Figure 2: Flowchart and proposed model for this study

In the proposed model, there are 80 chromosomes in each population. The mutation rate has been set to 0.15. There is a single-point crossover at the midpoint of the chromosome. The number of generations has been developed to 10.

Each feature selection is different for different classifiers and algorithms. For instance, for the AdaBoost Classifier, selected features are radius mean, area mean, smoothness mean, concavity mean, texture standard error, area standard error, smoothness standard error, compactness standard error, fractal dimension standard error, texture worst, perimeter worst, area worst, smoothness worst, compactness worst, concavity worst, concave points worst. However, for the decision tree classifier, the selected features are radius mean, texture mean, perimeter mean, smoothness mean, compactness mean, concave points mean, fractal dimension mean, radius standard error, texture standard error, smoothness standard error, concave points standard error, symmetry standard error, fractal dimension standard error, radius worst, texture worst, smoothness worst, compactness worst, concavity worst, symmetry worst, fractal dimension worst. Thus, genetic algorithms have helped to make predictions by reducing the features in the dataset.

The dataset has been splitted into 50/50, 60/40, 70/30, 80/20, 90/10 for the training and test set, but the ratio of 80/20 has been kept because it has led to the highest accuracy among other ratios.

6. Results

"Performance metrics of the model were calculated to ascertain the reliability of the study [40]." Some of the metrics that have been used in this study can be seen below:Sensitivity (Sens), Specificity (Spec), F1-score (F1), Matthews Correlation Coefficient (MCC)

Accuracy=TP+TNTP+TN+FP+FN

Sensitivity=TPTP+FN

Precision=TPTP+FP

F1-score=2TP(2TP+FP+FN)

Specificity=TNFP+TN

Matthews Correlation Coefficient =(TPxTN)-(FPxFN)(TP+FP)(TP+FN)(TN+FP)(TN+FN)

Before employing a genetic algorithm for feature selection, the accuracy of various classifiers and algorithms can be observed below.

Table 2: Accuracies of Different Classifiers

Classifier	Training Accuracy
Extra Trees	0.973684
AdaBoost	0.973684
GaussianNB	0.973684
MLP	0.973684
LGBM	0.973684
CatBoost	0.964912
Random Forest	0.964912
SGD	0.964912
Bagging	0.964912
QDA	0.956140
Gradient Boosting	0.956140

KNeighbors	0.956140
Logistic	0.956140
RadialSVM	0.947368
Passive Aggressive	0.947368
PolySVM	0.947368
MultinomialNB	0.938596
Decision Tree	0.938596

After employing a genetic algorithm for feature selection, there was an increase in accuracy for most of the classifiers, although some of the classifiers did not have any improvement. Table 3 shows the extracted features using a genetic algorithm.

Table 3: Classifiers and Extracted Features

Classifier	Extracted Features
ExtraTrees	radius mean, area mean, smoothness mean, concavity mean, texture standard error, area standard error, smoothness standard error, compactness standard error, fractal dimension standard error, texture worst, perimeter worst, area worst, smoothness worst, compactness worst, concavity worst, concave points worst
AdaBoost	radius mean, texture mean, perimeter mean, area mean, smoothness mean, concavity mean, texture standard error, area standard error, compactness standard error, concave points standard error, symmetry standard error, fractal dimension standard error, radius worst, texture

	worst, smoothness worst, compactness worst, symmetry worst
GaussianNB	radius mean, area mean, compactness mean, concavity mean, concave points mean, symmetry mean, perimeter standard error, concavity standard error, concave points standard error, texture worst, perimeter worst, concave points worst
MLP	radius mean, perimeter mean, compactness mean, concave points mean, symmetry mean, radius standard error, area standard error, compactness standard error, symmetry standard error, fractal dimension standard error, radius worst, texture worst, concavity worst
LGBM	texture mean, area mean, smoothness mean, concave points mean, symmetry mean, fractal dimension mean, area standard error, smoothness standard error, compactness standard error, concave points standard error, fractal dimension standard error, texture worst, smoothness worst, concavity worst
CatBoost	area mean, smoothness mean, compactness mean, concavity mean, radius standard error, texture standard error, area standard error, smoothness standard error, compactness standard error, fractal dimension standard error, perimeter worst, area worst, smoothness worst
RandomForest	area mean, smoothness mean, compactness mean, concavity mean, radius standard error, texture standard error, area standard error, smoothness standard error, compactness standard error, fractal dimension standard error, perimeter worst, area worst, smoothness worst
SGD	texture mean, area mean, smoothness mean, symmetry mean, texture standard error, perimeter standard error, compactness standard error, concavity standard error, fractal dimension standard error, texture worst, perimeter worst, area worst, concavity worst, fractal dimension worst

Bagging	area mean, smoothness mean, compactness mean, concavity mean, radius standard error, texture standard error, area standard error, smoothness standard error, compactness standard error, fractal dimension standard error, perimeter worst, area worst, smoothness worst
QDA	texture mean, area mean, smoothness mean, symmetry mean, texture standard error, perimeter standard error, compactness standard error, concavity standard error, fractal dimension standard error, texture worst, perimeter worst, area worst, concavity worst, fractal dimension worst
GradientBoosting	radius mean, texture mean, smoothness mean, concavity mean, symmetry mean, fractal dimension mean, perimeter standard error, smoothness standard error, concavity standard error, fractal dimension standard error, perimeter worst, compactness worst, concavity worst, symmetry worst, fractal dimension worst
KNeighbors	texture mean, perimeter mean, smoothness mean, concavity mean, concave points mean, symmetry mean, area standard error, smoothness standard error, concave points standard error, symmetry standard error, fractal dimension standard error, perimeter worst, smoothness worst, concavity worst, concave points worst, fractal dimension worst
Logistic	texture mean, perimeter mean, smoothness mean, concavity mean, concave points mean, symmetry mean, area standard error, smoothness standard error, concave points standard error, symmetry standard error, fractal dimension standard error, perimeter worst, smoothness worst, concavity worst, concave points worst, fractal dimension worst
RadialSVM	radius mean, texture mean, smoothness mean, concavity mean, area standard error, smoothness standard error, compactness standard error, concavity standard error, radius worst, texture worst, smoothness worst, concavity worst, concave points worst, symmetry worst, fractal dimension worst
PolySVM	radius mean, perimeter mean, area mean, concavity mean, concave points mean, fractal dimension mean, perimeter standard error, area

	standard error, concavity standard error, concave points standard error, symmetry standard error, texture worst, perimeter worst, smoothness worst, concavity worst, fractal dimension worst
MultinomialNB	perimeter mean, area mean, smoothness mean, compactness mean, concavity mean, concave points mean, fractal dimension mean, perimeter standard error, area standard error, fractal dimension standard error, radius worst, texture worst, perimeter worst, smoothness worst, concavity worst, symmetry worst
DecisionTree	radius mean, texture mean, perimeter mean, smoothness mean, compactness mean, concave points mean, fractal dimension mean, radius standard error, texture standard error, smoothness standard error, concave points standard error, symmetry standard error, fractal dimension standard error, radius worst, texture worst, smoothness worst, compactness worst, concavity worst, symmetry worst, fractal dimension worst

Table 4 shows the true positive (TP), True Negative (TN), False Positive (FP), Training Accuracy (TA), Validation Accuracy (VA), Sensitivity (Sens), Specificity (Spec), F1-score (F1), Matthews Correlation Coefficient (MCC), and Time of each classifier; more information can be found in the appendix section.

Table 4: Classifiers and Performance Evaluation

Classifiers	TP	FP	TN	FP	TA	VA	Sens	Spec	Prec	NPV	FPR	FDR	FNR	F1	MCC	Time
Extra Trees	65	2	44	3	1.0000	0.9561	0.9559	0.9565	0.9701	0.9362	0.0435	0.0299	0.0441	0.9630	0.9094	14.597
AdaBoost	69	2	41	2	0.9912	0.9649	0.9718	0.9535	0.9718	0.9535	0.0465	0.0282	0.0282	0.9718	0.9253	197.39
GaussianNB	71	0	39	4	0.925	0.9649	0.9467	1.0000	1.0000	0.9070	0	0	0.0533	0.9726	0.9266	7.82
MLP	70	1	4	39	1.0000	0.9561	0.9459	0.9750	0.9859	0.9070	0.0250	0.0141	0.0541	0.9655	0.9068	1298.99
LGBM	69	2	41	2	0.9912	0.9649	0.9718	95.35	97.18	95.35	0.0465	0.0282	0.0282	0.9718	0.9253	224.51
CatBoost	71	0	41	2	0.9912	0.9825	0.9726	1.0000	1.0000	0.9535	0	0	0.0274	0.9861	0.9630	864.42
Random Forest	71	0	42	1	1.0000	0.9912	0.9861	1.0000	1.0000	0.9767	0	0	0.0139	99.30	98.14	608.55
SGD	71	0	35	8	0.9737	92.98	0.8987	1.0000	1.0000	0.8140	0	0	0.1013	0.9467	0.8553	12.73
Decision	70	1	39	4	0.9737	0.9649	0.9467	1.0000	1.0000	0.9070	0	0	0.0533	0.9726	0.9266	23.06

Tree																
QDA	67	4	41	2	0.9825	0.9391	0.9571	0.9111	0.9437	0.9318	0.0889	0.0563	0.0429	0.9504	0.8719	10.75
Gradient Boosting	71	0	41	2	0.9912	0.9825	0.9726	1.0000	1.0000	0.9535	0	0	0.0274	0.9861	0.9630	452.80
KNeighbors	71	0	40	3	0.9825	0.9737	95.95	1.0000	1.0000	0.9302	0	0	0.0405	0.9793	0.9447	28.50
Logistic	70	1	41	2	1.0000	0.9737	97.22	0.9762	0.9859	0.9535	0.0238	0.0141	0.0278	0.9790	0.9439	277.44
Radial SVM	71	0	39	4	0.9649	0.9649	0.9467	1.0000	1.0000	0.9070	0	0	0.0533	0.9726	0.9266	48.33
Poly SVM	71	0	39	4	0.9649	0.9649	0.9467	1.0000	1.0000	0.9070	0	0	0.0533	0.9726	0.9266	31.30
MultinomialNB	71	0	37	6	0.9561	0.9474	0.9221	1.0000	1.0000	0.8605	0	0	0.0779	0.9595	0.8907	9.40
Bagging	70	1	40	3	0.9825	0.9649	0.9589	0.9756	0.9859	0.9302	0.0244	0.0141	0.0411	0.9722	0.9253	97.31

After feature selection, there has been an increase in training accuracy and performance metrics. All the models had a training accuracy higher than 0.9500; however, without feature selection, nine of the classifiers fell below this threshold. The time required for training varied widely, ranging from 7.82 seconds for GaussianNB to 1298.99 seconds for MLP. Below, you may find the summary of the table and maximum/minimum values for each performance metric.

Table 5: Maximum/Minimum values for each performance metrics.

Metric	Maximum Values	Minimum Values
TP (True Positive)	71 (Multiple Classifiers)	65 (Extra Trees)
FP (False Positive)	4 (QDA)	0 (Multiple Classifiers)
TN (True Negative)	42 (Random Forest)	4 (MLP)

TA (Training Accuracy)	1.0000 (Extra Trees, MLP, Random Forest, Logistic)	0.925 (GaussianNB)
VA (Validation Accuracy)	0.9912 (Random Forest)	0.9298 (SGD)
Sens (Sensitivity)	1.0000 (Multiple Classifiers)	0.8987 (SGD)
Spec (Specificity)	1.0000 (Multiple Classifiers)	0.9070 (Multiple Classifiers)
Prec (Precision)	1.0000 (Multiple Classifiers)	0.9070 (Multiple Classifiers)
NPV (Negative Predictive Value)	0.9767 (Random Forest)	0.8140 (SGD)
FPR (False Positive Rate)	0.0889 (QDA)	0 (Multiple Classifiers)
FDR (False Discovery Rate)	0.0563 (QDA)	0 (Multiple Classifiers)
FNR (False Negative Rate)	0.1013 (SGD)	0.0139 (Random

		Forest)
F1 (F1 Score)	99.30 (Random Forest)	0.9467 (SGD)
MCC (Matthews Correlation Coefficient)	98.14 (Random Forest)	0.8553 (SGD)
Time	1298.99 (MLP)	7.82 (GaussianNB)

7. Discussion

In various studies, the advantages of using Genetic Algorithms for feature selection have been well-documented [41, 42]. This study designs a feature selection model that employs Genetic Algorithms to pinpoint relevant features, which is particularly useful when dealing with problems with many features.

When compared with 17 classifiers without any feature selection, the results indicate that feature selection has improved performance metrics. Table 6 below presents a comparative analysis of classification accuracies from other studies that employed different feature selection methods for the same dataset.

Table 6: Comparison of other proposed methods [11, 43, 44]

Classifier	This Study (Random Forest)	ANN	SVM	Graph-Based	PS-Classifier
Test Accuracy	99.12%	96.70%	96.50%	96.40%	96.90%

This table indicates that Random Forest is the most effective classifier for feature selection using genetic algorithms. Additionally, this study has outperformed many other methods in the literature as seen in Table 6. The use of genetic algorithms has yielded a significant improvement in accuracy compared to traditional methods like Threshold Variance and Pearson Correlation for feature selection

8. Conclusion

In the proposed model, feature selection has been conducted using a genetic algorithm for breast cancer detection. The model has been evaluated with 17 different classifiers, and the

highest test accuracy was achieved with the Random Forest Classifier (99.12%). The results show that selecting appropriate features can improve classification performance. This research demonstrates that a genetic algorithm is effective for feature selection and suggests that future studies could explore its applicability to other types of cancer, such as brain, prostate, and kidney.

9. References:

1. Robbins, K. R., et al. "The ant colony algorithm for feature selection in high-dimension gene expression data for disease classification." *Mathematical Medicine and Biology: a Journal of the IMA* 24.4 (2007): 413-426.

2. Moradi, Parham, and Mehrdad Rostami. "A graph theoretic approach for unsupervised feature selection." *Engineering Applications of Artificial Intelligence* 44 (2015): 33-45.

3. https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020-00398-3

4. Kohavi, R., John, G.H., "Wrappers for feature subset selection", Artificial Intelligence, 97(1-2): 273-324, (1997).

5 Aalaei, Shokoufeh et al. "Feature selection using genetic algorithm for breast cancer diagnosis: experiment on three different datasets." *Iranian journal of basic medical sciences* vol. 19,5 (2016): 476-82.

6 Łukasiewicz, Sergiusz et al. "Breast Cancer-Epidemiology, Risk Factors, Classification, Prognostic Markers, and Current Treatment Strategies-An Updated Review." *Cancers* vol. 13,17 4287. 25 Aug. 2021, doi:10.3390/cancers13174287

7 Basha, S. Saheb, and K. Satya Prasad. "AUTOMATIC DETECTION OF BREAST CANCER MASS IN MAMMOGRAMS USING MORPHOLOGICAL OPERATORS AND FUZZY C--MEANS CLUSTERING." *Journal of Theoretical & Applied Information Technology* 5.6 (2009).

8 Kuhl, Christiane K et al. "Mammography, breast ultrasound, and magnetic resonance imaging for surveillance of women at high familial risk for breast cancer." *Journal of clinical oncology : official journal of the American Society of Clinical Oncology* vol. 23,33 (2005): 8469-76. doi:10.1200/JCO.2004.00.4960

9 Elmore, Joann G., et al. "Variability in radiologists' interpretations of mammograms." *New England Journal of Medicine* 331.22 (1994): 1493-1499.

10 Fletcher, Suzanne W., et al. "Report of the international workshop on screening for breast cancer." *JNCI: Journal of the National Cancer Institute* 85.20 (1993): 1644-1656.

11 Aalaei, Shokoufeh et al. "Feature selection using genetic algorithm for breast cancer diagnosis: experiment on three different datasets." *Iranian journal of basic medical sciences* vol. 19,5 (2016): 476-82.

12 Guyon, Isabelle, and André Elisseeff. "An introduction to variable and feature selection." *Journal of machine learning research*3.Mar (2003): 1157-1182.

13 Bermejo, Pablo, Jose A. Gámez, and Jose M. Puerta. "A GRASP algorithm for fast hybrid (filter-wrapper) feature subset selection in high-dimensional datasets." *Pattern Recognition Letters* 32.5 (2011): 701-711.

14https://link.springer.com/chapter/10.1007/978-1-4615-5725-8_3#:~:text=The%20wrapper%20method%20searches%20for,approach%20to%20feature%20subset%20selection.

15 Aghdam, Mehdi Hosseinzadeh, Nasser Ghasem-Aghaee, and Mohammad Ehsan Basiri. "Application of ant colony optimization for feature selection in text categorization." *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*. IEEE, 2008.

16 Unler, Alper, and Alper Murat. "A discrete particle swarm optimization method for feature selection in binary classification problems." *European Journal of Operational Research* 206.3 (2010): 528-539.
17 Karegowda, Asha Gowda, M. A. Jayaram, and A. S. Manjunath. "Feature subset selection problem using wrapper approach in supervised learning." *International journal of Computer applications* 1.7 (2010): 13-17.
18 Youn, Eunseog, et al. "Support vector-based feature selection using Fisher's linear discriminant and Support Vector Machine." *Expert Systems with Applications* 37.9 (2010): 6148-6156.
19 Deisy, C., et al. "Efficient dimensionality reduction approaches for feature selection." *International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007)*. Vol. 2. IEEE, 2007.
20 Sridevi, T., and A. Murugan. "An intelligent classifier for breast cancer diagnosis based on K-Means clustering and rough set." *International Journal of Computer Applications* 85.11 (2014).
21 Sridevi, T., and A. Murugan. "A novel feature selection method for effective breast cancer diagnosis and prognosis." *International Journal of Computer Applications* 88.11 (2014).
22 10.24432/C5DW2B
https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic
22a Karegowda AG, Jayaram M, Manjunath A. Feature subset selection problem using wrapper approach in supervised learning. Int J Comput Appl. 2010;1:13–17.
23a oun E, Koenig L, Jeong MK, Baek SH. Support vector-based feature selection using Fisher's linear discriminant and Support Vector Machine. Exp Syst Appl. 2010;37:6148–6156.
23 DOI 10.1007/s10994-006-6226-1
24a Deisy C, Subbulakshmi B, Baskar S, Ramaraj N. Efficient dimensionality reduction approaches for feature selection. Conference on Computational Intelligence and Multimedia Applications, 2007 International Conference on; 2007: IEEE
24https://www.tacoma.uw.edu/sites/default/files/2021-08/melanson_david_senior_thesis_2020.pdf
25 https://www.mdpi.com/2078-2489/11/6/332
26 Freund, Yoav, and Robert E. Schapire. "Experiments with a new boosting algorithm." *icml*. Vol. 96. 1996.
27 Breiman, Leo. "Bagging predictors." *Machine learning* 24 (1996): 123-140.
28 Amit, Yali, and Donald Geman. "Shape quantization and recognition with randomized trees." *Neural computation* 9.7 (1997): 1545-1588.
29 Ho, Tin Kam. "The random subspace method for constructing decision forests." *IEEE transactions on pattern analysis and machine intelligence* 20.8 (1998): 832-844.
30 Ho, Tin Kam. "Random decision forests." *Proceedings of 3rd international conference on document analysis and recognition*. Vol. 1. IEEE, 1995.
31https://www.researchgate.net/publication/361392986_Gaussian_Naive_Bayes_Algorithm_A_Reliable_Technique_Involved_in_the_Assortment_of_the_Segregation_in_Cancer
32 https://www.mdpi.com/2077-1312/9/5/496
33https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf
34 https://arxiv.org/pdf/1906.02590.pdf
35 https://academicworks.cuny.edu/cgi/viewcontent.cgi?article=1535&context=gc_etds
36 https://link.springer.com/chapter/10.1007/978-3-642-34041-3_27

37 https://eudl.eu/pdf/10.4108/eai.13-7-2017.2270596
38 https://doi.org/10.1016/B978-0-12-416743-8.00005-1
39 https://inis.iaea.org/collection/NCLCollectionStore/_Public/38/027/38027911.pdf
40 Eroltu, Kaan. "Comparing different Convolutional Neural Networks for the classification of Alzheimer's Disease." *Journal of High School Science* 7.3 (2023).
41 Oh, Il-Seok, Jin-Seon Lee, and Byung-Ro Moon. "Hybrid genetic algorithms for feature selection." *IEEE Transactions on pattern analysis and machine intelligence* 26.11 (2004): 1424-1437.
42 Hadizadeh, Farzin, Saadat Vahdani, and Mehrnaz Jafarpour. "Quantitative structure-activity relationship studies of 4-imidazolyl-1, 4-dihydropyridines as calcium channel blockers." Iranian journal of basic medical sciences 16.8 (2013): 910.
43. Senturk ZK, Kara R. Breast Cancer Diagnosis via Data Mining: Performance Analysis of Seven different algorithms. *Computer Science & Engineering.* 2014;4:35.
44. Noruzi A, Sahebi H. A graph-based feature selection method for improving medical diagnosis. *Adv Comput Sci.* 2015;4:36–40.

Appendix

Figure 1: Confusion Matrix of Extra Trees Classifier

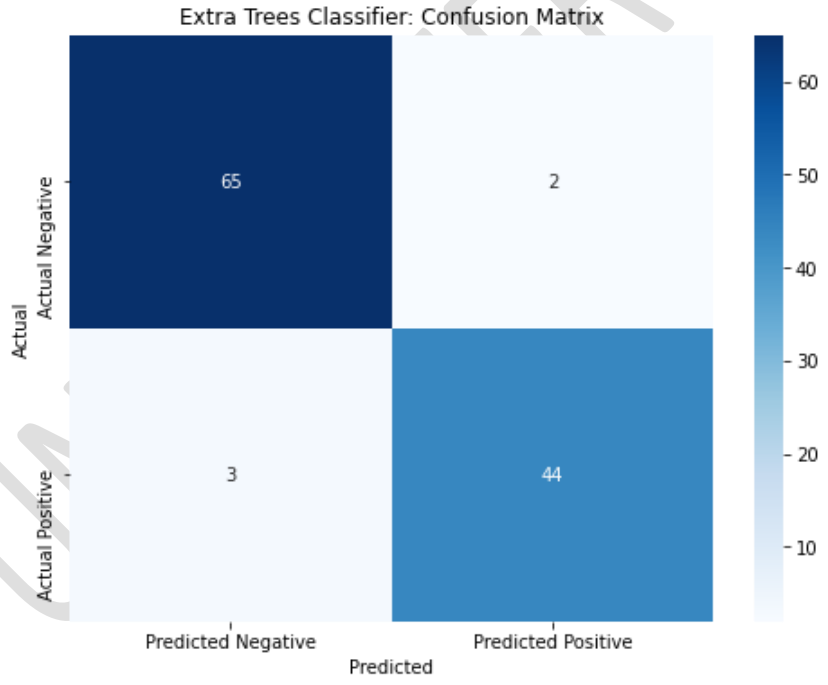


Figure 2: Confusion Matrix of AdaBoost Classifier

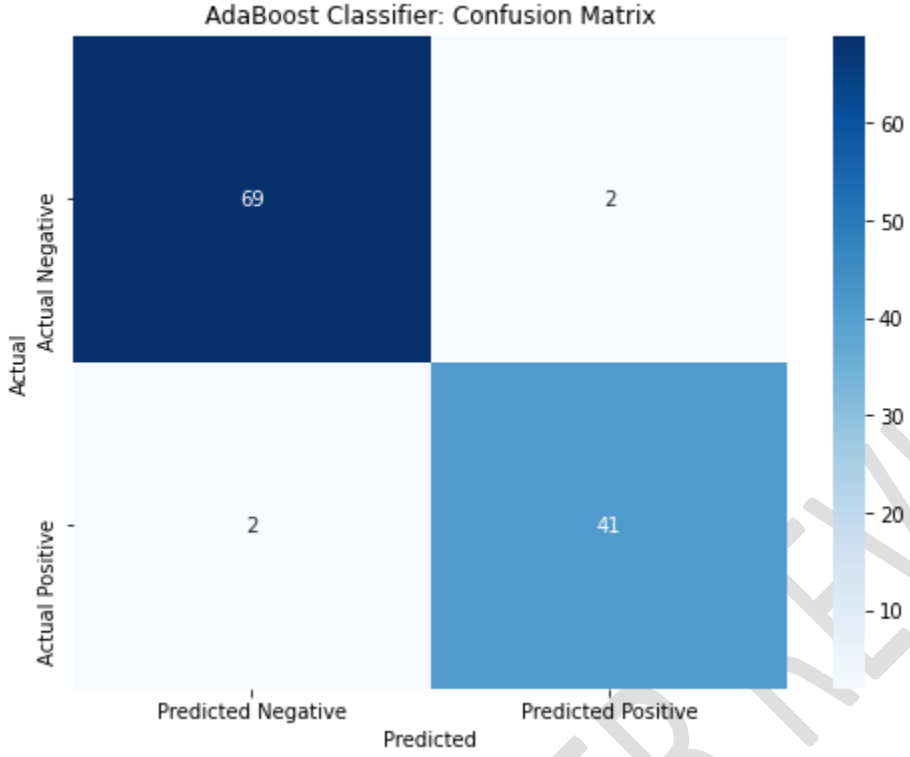
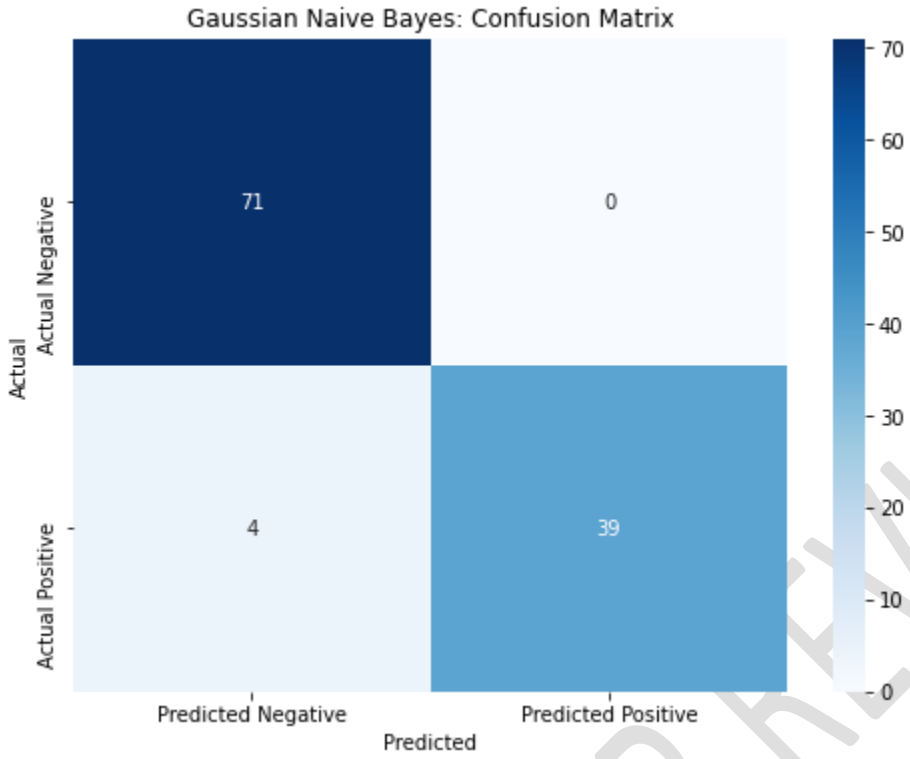


Figure 3: Confusion Matrix of Gaussian Naive Bayes Classifier



\

Figure 4: Confusion Matrix of Categorical Boosting Classifier

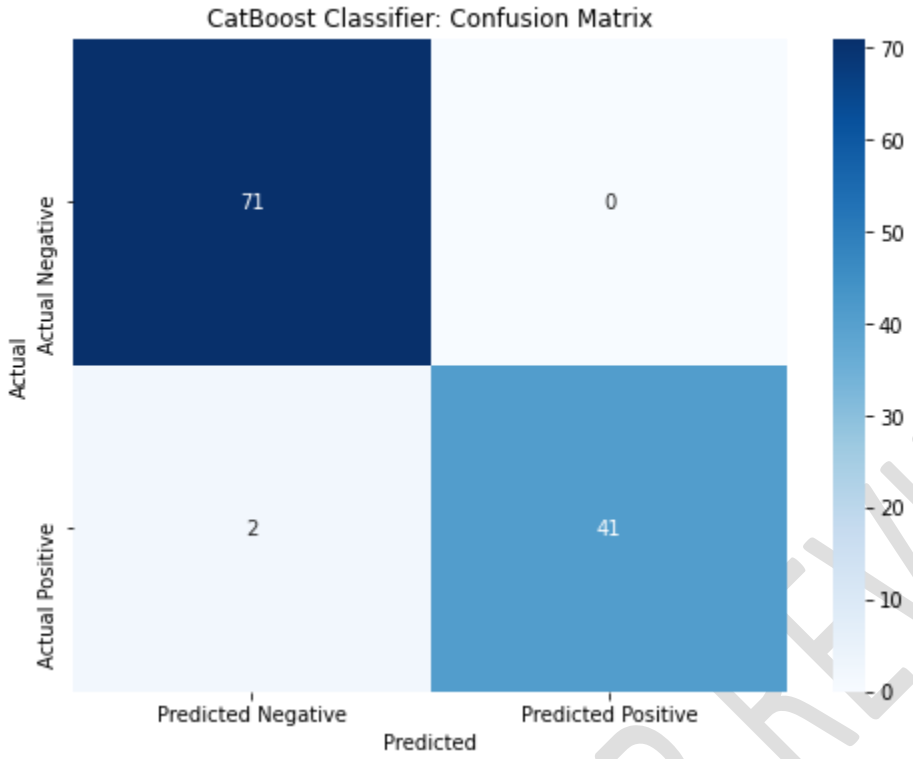


Figure 5: Confusion Matrix of Random Forest Classifier

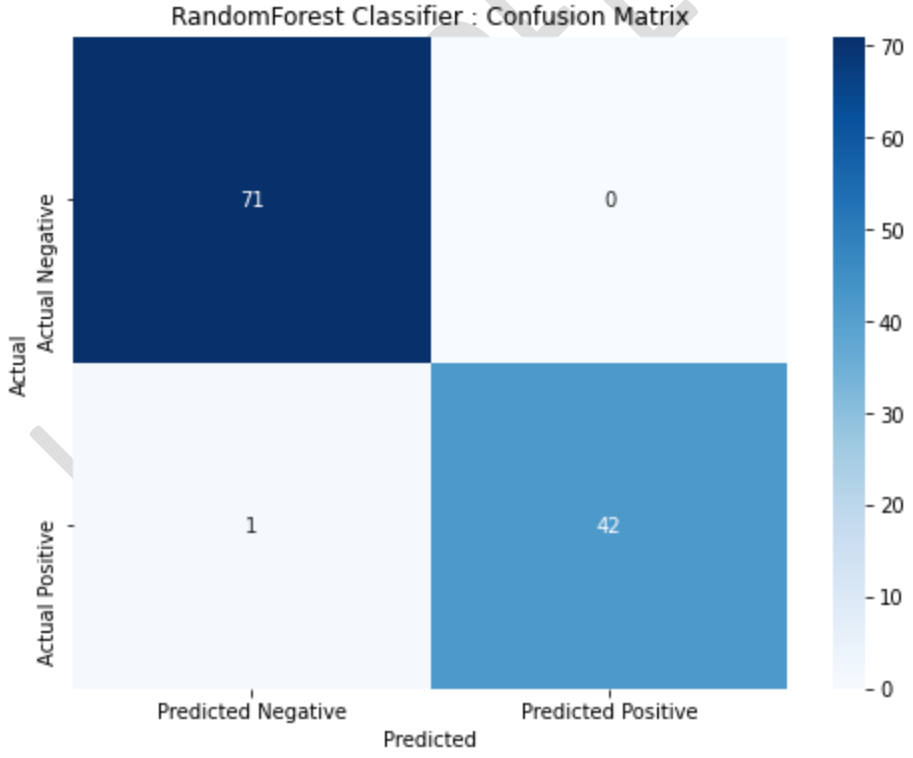


Figure 6: Confusion Matrix of Bagging Classifier

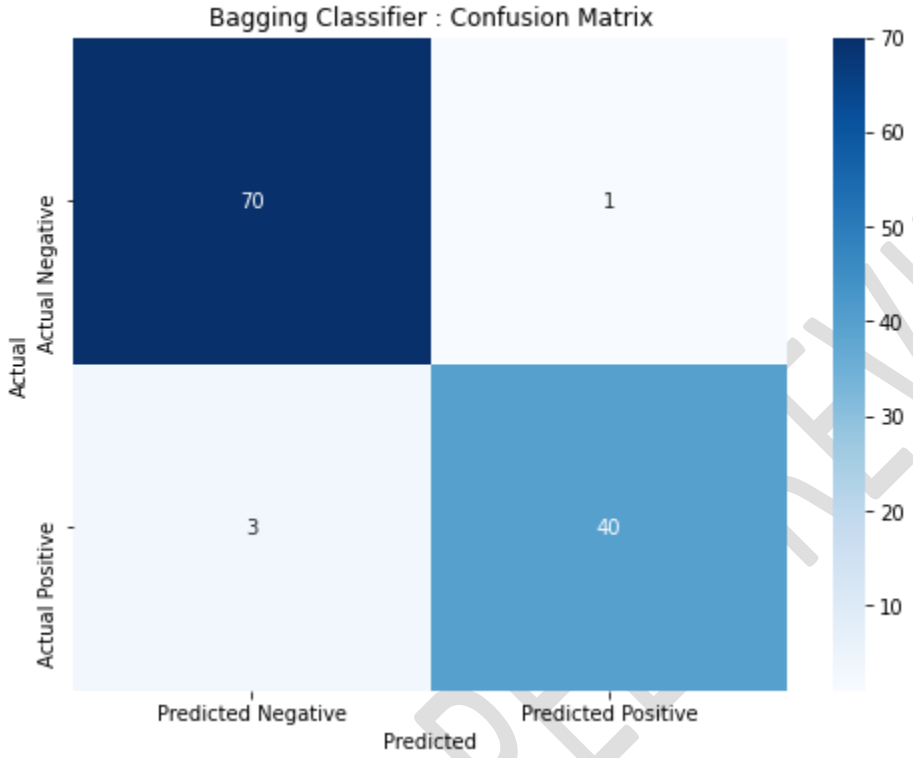


Figure 7: Confusion Matrix of Light Gradient Boosting Machine Classifier

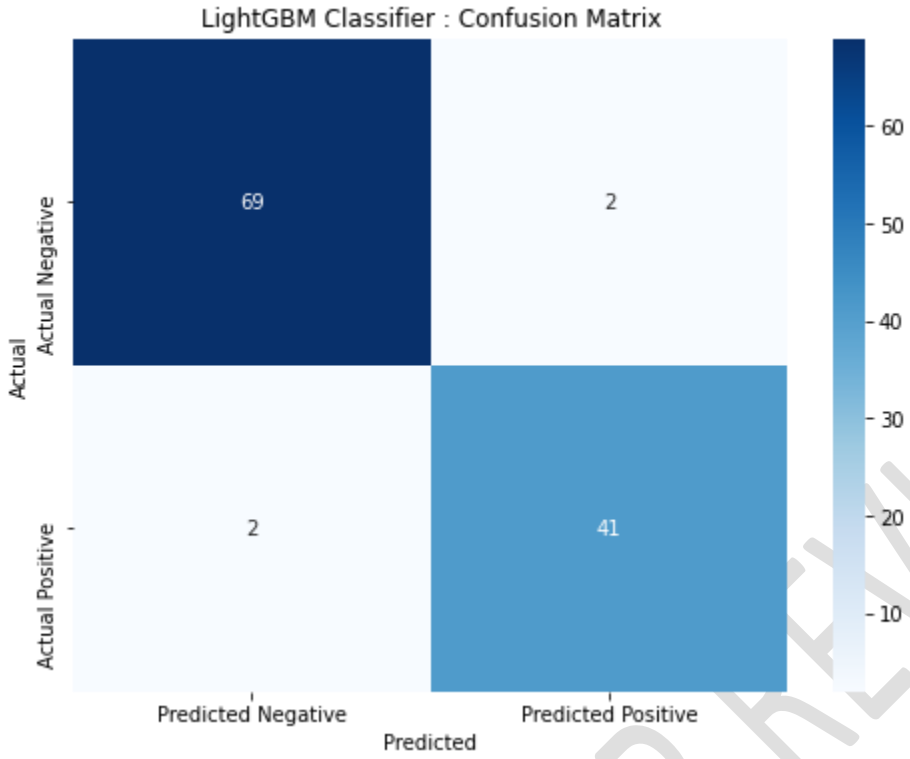


Figure 8: Confusion Matrix of Quadratic Discriminant Analysis Machine Classifier

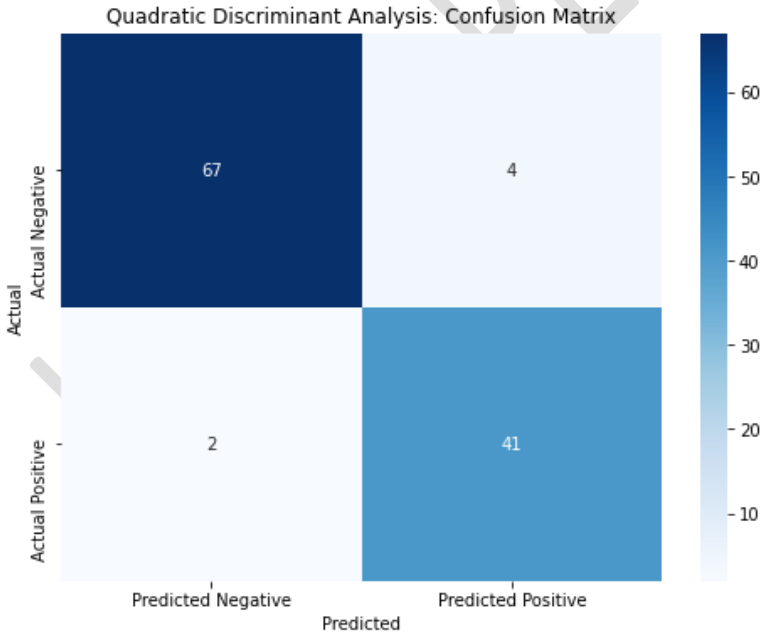


Figure 9: Confusion Matrix of Stochastic Gradient Descent Classifier

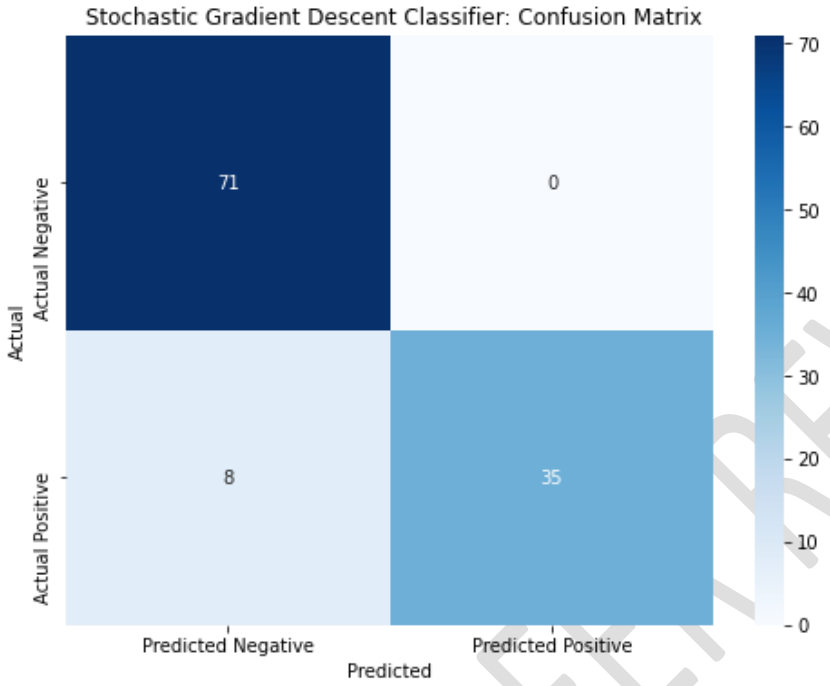


Figure 10: Confusion Matrix of Gradient Boosting Classifier

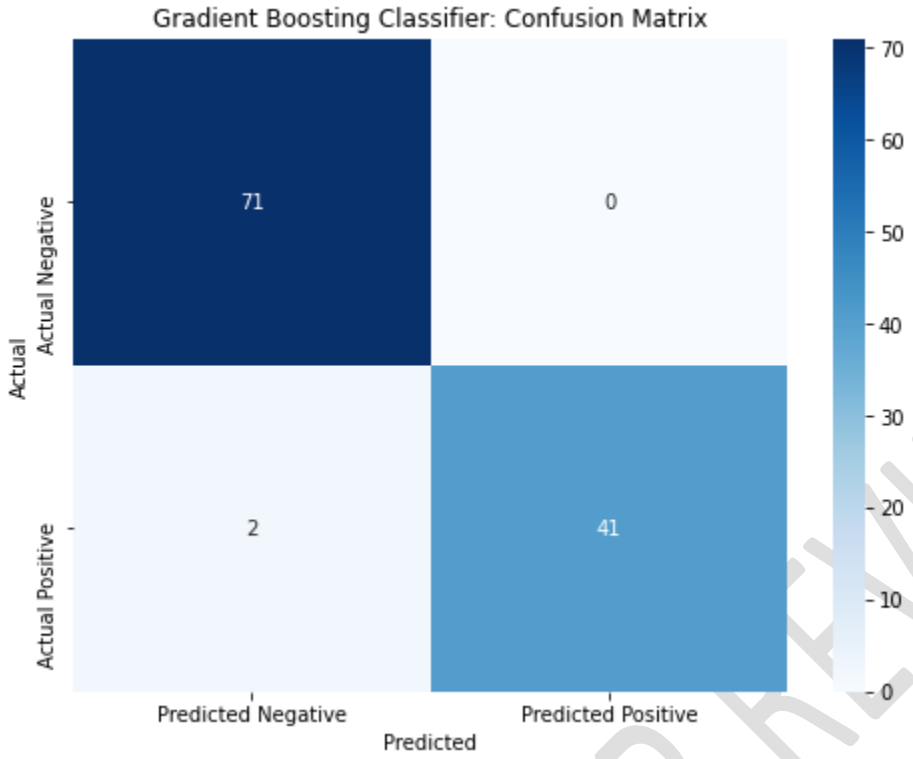


Figure 11: Confusion Matrix of K-nearest Neighbors Classifier

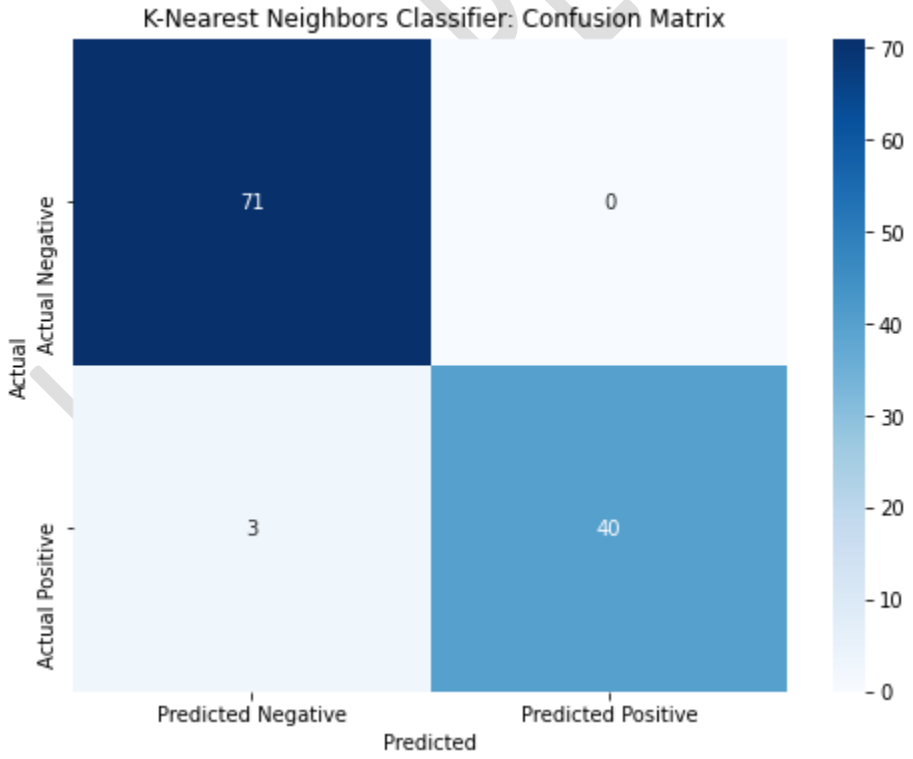


Figure 12: Confusion Matrix of Logistic Regression Classifier

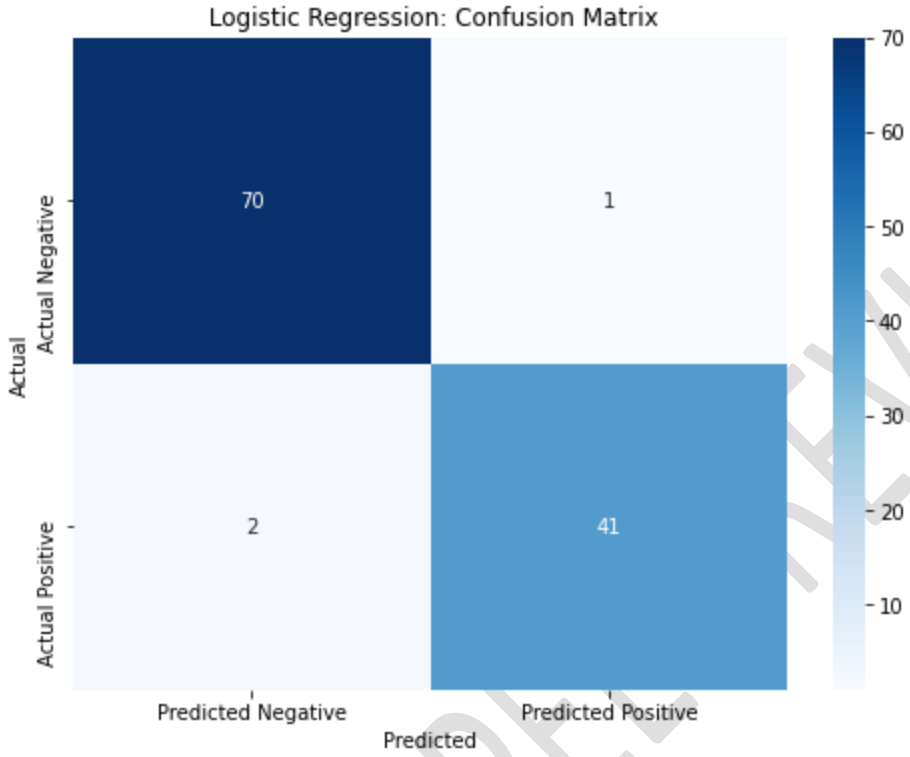


Figure 13: Confusion Matrix of Radial Kernel Support Vector Machine Classifier

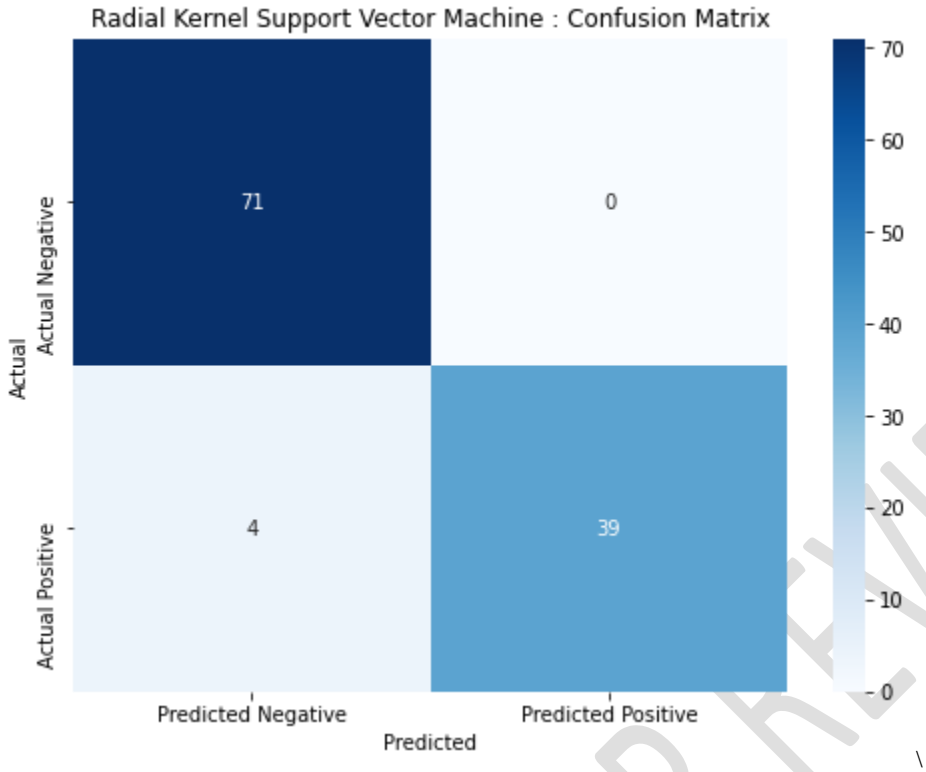


Figure 14: Confusion Matrix of Polynomial Kernel Support Vector Machine Classifier

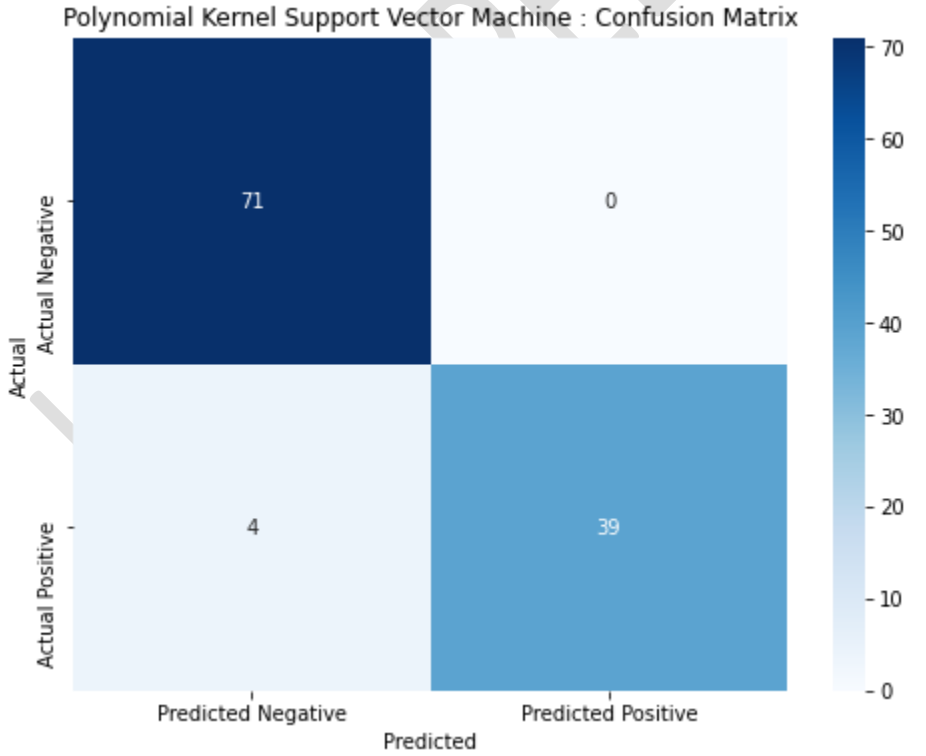


Figure 15: Confusion Matrix of Multilayer Perceptron Classifier

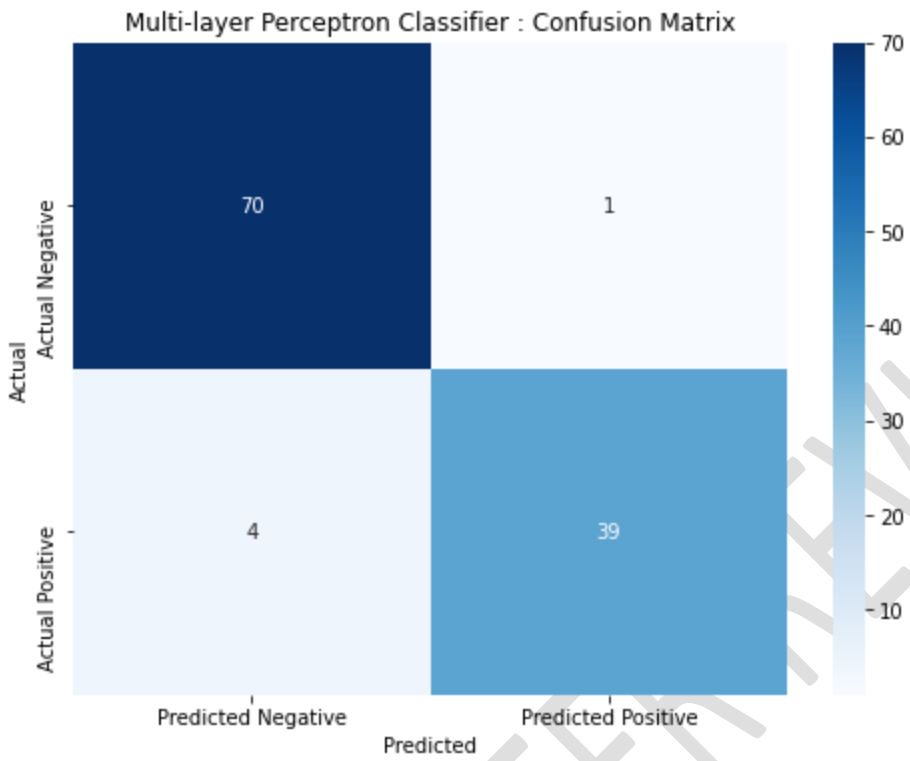


Figure 16: Confusion Matrix of Multinomial Naive Bayes Classifier

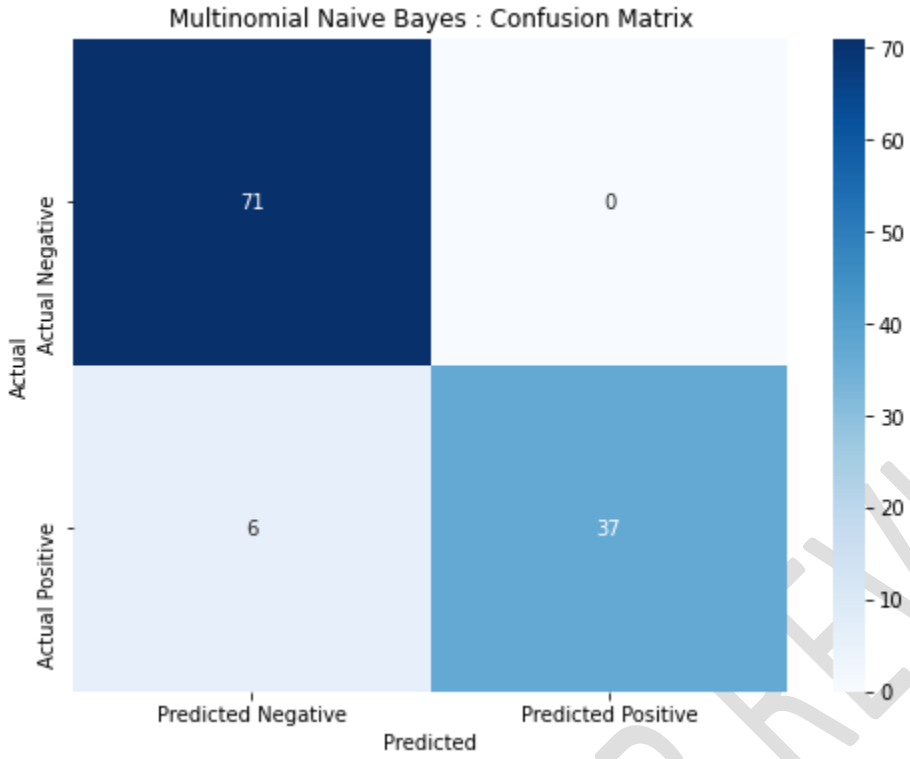


Figure 17: Confusion Matrix of Multi-layer Perceptron Classifier