ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR845

Visualizing Transformers for Breast Histopathology

A. Anu Priya[1]
Supervisor, Assistant Professor,
M.E. Computer Science and Engineering,
KIT – Kalaignar Karunanidhi Institute of Technology (Autonomous),
Coimbatore, TN, India

T. Pramoth Krishnan[2]
Student, M.E. Computer Science and Engineering,
KIT – Kalaignar Karunanidhi Institute of Technology
(Autonomous), Coimbatore, TN, India

Dr. C. Suresh[3]
Assistant Professor /Computer science and Engineering,
KIT – Kalaignar Karunanidhi Institute of Technology
(Autonomous), Coimbatore, TN, India

Abstract:- Detecting breast cancer early is crucial for improving patient survival rates. Using machine learning models to predict breast cancer holds promise for enhancing early detection methods. However, evaluating the effectiveness of these models remains challenging. Therefore, achieving high accuracy in cancer prediction is essential for improving treatment strategies and patient outcomes. By applying various machine learning algorithms to the Breast Cancer Wisconsin Diagnostic dataset, researchers aim to identify the most efficient approach for breast cancer diagnosis. They evaluate the performance of classifiers such as Random Forest, Naïve Bayes, Decision Tree (C4.5), KNN, SVM, and Logistic Regression, considering metrics like confusion matrix, accuracy, and precision.

The assessment reveals that Random Forest outperforms other classifiers, achieving the highest accuracy rate of 97%. This study is conducted using the Anaconda environment, Python programming language, and Sci-Kit Learn library, ensuring replicability and accessibility of the findings. In summary, this study demonstrates the potential of machine learning algorithms for breast cancer prediction and highlights Random Forest as the most effective approach. Its findings contribute valuable insights to the field of breast cancer diagnosis and treatment.

Keywords:- *Machine Learning Models, Data Exploratory Techniques, Breast Cancer Diagnosis, Tumors Classification.*

I. INTRODUCTION

Breast cancer, a widespread issue impacting women globally, continues to present a significant danger to female health on a global scale. In 2020, breast cancer saw over 2.2 million new cases and nearly 685,000 fatalities, ranking second only to lung cancer in terms of female mortality. In the United States alone, there were 281,550 new cases and 43,600 female deaths attributed to breast cancer in 2021. Emerging from breast tissue, breast cancer develops from cells found within milk ducts or lobules responsible for milk production. These cancerous cells arise from alterations or mutations in DNA and RNA, which can happen spontaneously or due to factors like radiation, chemicals, aging, and cellular damage. Breast cancer is divided into benign and malignant tumors, with the latter being more severe and life-threatening. Studies suggest that about 20% of women with malignant tumors do not survive, highlighting the critical importance of early detection and swift treatment. Over the past few decades, there has been a rise in breast cancer cases globally, although advancements in screening and treatment have led to a decrease in mortality rates. Specifically, mammography screenings have played a role in reducing mortality by 20%, while improvements in cancer treatments have further boosted survival rates by 60%. The field of machine learning has emerged as a valuable tool in predicting and diagnosing various diseases, including breast cancer. By utilizing demographic, lifestyle, laboratory data, mammographic patterns, patient biopsy information, and even genetic data, researchers have made significant progress in improving early detection and prognosis. To enhance the accuracy and efficiency of breast cancer diagnosis, several machine learning algorithms such as Random Forest, Naïve Bayes, Decision Tree, KNN, Support Vector Machine, and Logistic Regression have been utilized. These algorithms, along with data exploration techniques, aim to classify tumors as benign or malignant more precisely and with reduced computational time. In conclusion, this research contributes to the progression of breast cancer diagnosis through the utilization of sophisticated machine learning models. It establishes a structure for making precise predictions and conducting efficient analyses of tumor characteristics. With a focus on enhancing both time effectiveness and diagnostic precision, the study seeks to elevate the standards and trustworthiness of breast cancer diagnosis methodologies.

II. LITERATURE REVIEW

Breast cancer is responsible for a significant number of deaths globally. Despite traditional methods for detecting cancer, recent technological advancements provide experts with numerous adaptive approaches to identifying breast cancer in women. These modern technologies, combined with various techniques in data science (DS), aid in the gathering and assessment of cancer-related data, facilitating the prediction of this deadly disease. Among these DS methods, machine learning algorithms have shown promise in analyzing cancer-related data. For instance, a study [21] demonstrated that these algorithms can substantially enhance diagnostic accuracy. While an expert physician achieved a diagnostic accuracy of 79.97%, machine learning algorithms achieved an impressive 91.1% accuracy in predicting breast cancer.

In recent decades, the utilization of machine learning in medical applications has steadily grown. However, the collection of patient data and the expertise of medical practitioners remain pivotal for diagnosis. Machine learning classifiers have been instrumental in minimizing human errors and providing thorough analysis of medical data in a timely manner [22]. There exists a range of machine learning classifiers for modeling and predicting data. In our research, we employed Random Forest, Naïve Bayes, Decision Tree, KNN, Support Vector Machine, and Logistic Regression for the prediction of breast cancer.

In recent investigations, harmonic imaging and real-time compounding have proven effective in enhancing the clarity of images and the characterization of lesions. Additionally, ultrasound elastography has emerged as a promising technique. Initial findings suggest that it can enhance the accuracy of ultrasound in distinguishing breast masses, improving both specificity and positive predictive value. Lesions become visible on mammography or ultrasound due to variations in density and acoustic resistance compared to the surrounding breast tissue. In their work titled "Breast Cancer Prediction Using Genetic Algorithm Based Ensemble Approach," Pragya Chauhan and Amit Swami propose a system indicating that predicting breast cancer remains an area ripe for exploration in research.

This research investigates the application of diverse machine learning algorithms for Breast Cancer Prediction. Decision tree, random forest, support vector machine, neural network, linear model, adaboost, and naive bayes methods are utilized for prediction purposes. To enhance prediction accuracy, an ensemble method is employed, introducing a novel technique known as the GA-based weighted average ensemble method. This method addresses the limitations observed in classical weighted average approaches. By utilizing genetic algorithms, the GA-based weighted average method is applied for predicting multiple models. A comparative analysis among Particle Swarm Optimization (PSO), Differential Evolution (DE), and Genetic Algorithm (GA) reveals that GA performs exceptionally well for weighted average methods. Furthermore, a comparison between the classical ensemble method and the GA-based weighted average method concludes that the latter surpasses the former. This study, titled "On Breast Cancer Detection: An Application of Machine Learning Algorithms on the Wisconsin Diagnostic Dataset," is authored by Abien Fred M. Agarap.

In this study, six different machine learning algorithms are applied to detect cancer, focusing specifically on breast cancer diagnosis. The GRUSVM model, along with Linear Regression, Multilayer Perceptron (MLP), Nearest Neighbor (NN) search, Softmax Regression, and Support Vector Machine (SVM), are evaluated using the Wisconsin Diagnostic Breast Cancer (WDBC) dataset. The evaluation includes measuring their accuracy in classification testing, as well as their sensitivity and specificity values. This dataset contains features derived from digitized images of Fine Needle Aspiration (FNA) tests conducted on breast masses. The dataset is split into 70% for training and 30% for testing during the implementation of the machine learning algorithms. The findings indicate that all the machine learning algorithms performed strongly in classifying carcinoma, distinguishing between benign and malignant tumors effectively. Consequently, the statistical measures employed for classification tasks are considered satisfactory. To validate these results further, the study suggests employing cross-validation techniques like k-fold cross-validation. Using such methods not only provides a more precise evaluation of model prediction performance but also aids in determining the most optimal hyperparameters for the machine learning algorithms.

The paper "Breast Cancer Diagnosis by Various Machine Learning Techniques Using Blood Analysis Data" authored by Muhammet Fatih Aslan, Yunus Celik, Kadir Sabanci, and Akif Durdu focuses on early detection of carcinoma. It employs four distinct machine learning algorithms – Artificial Neural Network (ANN), Extreme Learning Machine (ELM), Support Vector Machine (SVM), and Nearest Neighbor (k-NN) – to analyze routine blood analysis results. Utilizing a dataset sourced from the UCI library, containing attributes like age, BMI, glucose, insulin, HOMA, leptin, adiponectin, resisting, and MCP1, the study explores the significance of these attributes in breast cancer detection. The study also utilizes hyperparameter optimization for k-NN and SVM, with ELM exhibiting the highest accuracy of 80% and a training time of 0.42 seconds. In "Performance Assessment of Machine Learning Techniques for Breast Cancer Prediction" by Yixuan Li and Zixuan Chen, two datasets are analyzed: the BCCD dataset with 116 volunteers and the WBCD dataset with 699 volunteers. After preprocessing, the WBCD dataset comprises 683 volunteers and a tumor indication index. Random Forest (RF) emerges as the top-performing classification model due to its superior accuracy, F-measure metric, and ROC curve performance. However, the study recognizes limitations related to data quantity and suggests exploring combinations of RF with other data mining technologies for enhanced results. In "Breast Cancer Prediction and Detection Using Data Mining Classification Algorithms: A Comparative Study" by Mumine Kaya Keles, various data mining classification algorithms are compared using the Weka tool to predict and detect breast cancer early. Random Forest emerges as the best-performing algorithm during 10-fold cross-validation, achieving an average accuracy of 92.2%. The study underscores the importance of further research to address data quantity limitations and the potential for combining RF with other data mining technologies. Lastly, in "Breast Cancer Prediction Using Data Mining Methods" by Haifeng Wang and Sang Won Yoon, the impact of feature space reduction on breast cancer prediction is explored. A hybrid approach combining principal component analysis (PCA) with data mining models is proposed. Performance evaluation using two test datasets – the Wisconsin Breast Cancer Database (1991) and the Wisconsin Diagnostic Breast Cancer (1995) – shows the effectiveness of PCA pre-processing, with PCs-SVM achieving 97.47% accuracy for WBC data and PCi-ANN achieving 99.63% accuracy for WDBC data. This is attributed to noise reduction and enriched feature space.

III. PROBLEM STATEMENT

The aim is to anticipate, using past data, whether a specific lump is benign (non-cancerous) or malignant (cancerous). This forecast is pivotal for early identification, allowing for timely intervention if there's a possibility of the lump progressing to cancer. Detecting the condition early enables swift treatment, improving the likelihood of recovery and reducing the potential for fatalities.

IV. MODULE DESCRIPTION

The central goal of our study is to differentiate between benign and malignant lumps, and to determine the most accurate classification model among the three employed - Decision Tree, Naïve Bayes, and Random Forest classifier. The methodology adopted to tackle this issue is outlined as follows:

A. Dataset Used
B. Data Pre-Processing
C. Feature selection and scaling
D. Training the models

A. Dataset Used

The research relies on the Breast Cancer Wisconsin (Diagnostic) Dataset sourced from Kaggle, comprising cases classified as either benign or malignant. It consists of 569 entries and 32 attributes, totaling 33 unique features. Both benign and malignant tumor instances are present in the dataset for examination.

- **Wisconsin Diagnostic Breast Cancer (WDBC):** The WDBC dataset, curated by Dr. William H. Wolberg and sourced from the General Surgery Department at the University of Wisconsin-Madison, USA, encompasses 10 attributes associated with breast tumors, derived from 569 patient samples. These data, accessible via FTP, were generated from fluid samples extracted from solid breast masses and analyzed using the Xcyt software for cytological feature assessment. Xcyt employs a curve-fitting algorithm to compute ten features, including mean, worst, and standard error values for each feature, resulting in 30 values per sample. An additional ID column is included for sample differentiation, and the diagnosis outcome (malignant or benign) is appended to each sample. In total, the dataset comprises 32 attributes, including ID, diagnosis, and 30 input features, across 569 instances. These attributes represent diverse tumor traits, including dimensions like radius, perimeter, and area, as well as texture, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension.

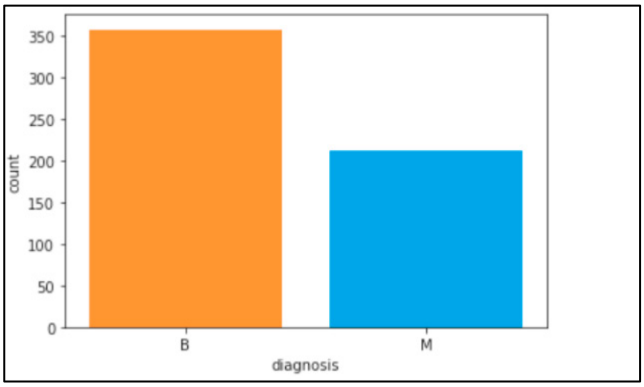


Fig 1: Wisconsin Breast Cancer Diagnostic Datasets

B. Data Preprocessing

The first and foremost stage in the procedure is data processing. It's necessary to process the data to filter out unnecessary information that could disrupt the prediction process. Data preprocessing involves identifying and removing outliers, addressing null values, and refining the dataset to ensure its suitability for model training.

- **Removal of Outliers:** An outlier denotes a data point that significantly deviates from the norm within a random sample from a population. To put it plainly, outliers are data points that stand far apart from the majority of the dataset. Eliminating these outliers is crucial as their extreme values can disrupt both the training and prediction processes, potentially leading to inaccurate outcomes. Various techniques exist for outlier detection and removal, with the interquartile range method being a widely accepted approach. In this study, I opted for the interquartile range method to identify outliers. This method involves scrutinizing each feature of the dataset to pinpoint outliers and subsequently replacing the minimum values with the minimum quartile and the maximum values with the maximum quartile.
- **Removal of Null Values:** As there were no values present in the dataset, there was no need to perform null value removal.

C. Features Selection and Scaling

The next step in the prediction process involves feature selection, where we identify the attributes that will be used for classification. In this case, our classification class is determined by a single attribute, 'diagnosis'. Datasets often include features with diverse magnitudes, units, and ranges. However, since many machine learning algorithms compute distances between data points using the Euclidean method, it's crucial to normalize the features to the same magnitude level. This normalization process, known as scaling, ensures that all features are comparable and aids in accurate predictions.

D. Training the Models

In the realm of machine learning methodologies, the learning process can be bifurcated into two primary categories: supervised and unsupervised learning. In supervised learning, a dataset containing input-output pairs is utilized to train the model, where each input is associated with a corresponding output label. Conversely, unsupervised learning involves working with unlabeled data, where there are no predetermined input-output pairs, rendering the learning task more complex.

Regression and classification represent the two fundamental approaches within supervised learning. Regression seeks to predict a continuous target variable, whereas classification predicts a discrete target variable.

This study explores six distinct classification algorithms, each serving its unique purpose in the predictive modeling process.

- *Decision Tree*
- *Naïve Bayes*
- *Random Forest Classifier*
- *k-Nearest Neighbors (K-NN)*
- *Support Vector Machine (SVM)*
- *Logistic Regression*

- *Decision Tree*

The Decision Tree C4.5 stands as a versatile tool for predictive modeling, applicable across various domains. It's crafted using an algorithmic approach that splits datasets in multiple ways based on various conditions. In regression modeling, these trees predict future outcomes or classify data based on input. Visually resembling flowcharts, decision trees begin at the root node with a specific data query, branching out to potential answers, and progressing to decision nodes that pose further questions, ultimately ending in terminal or "leaf" nodes. In machine learning, decision trees offer an effective decision-making method by laying out the problem and its potential outcomes. As the algorithm accesses more data, it can forecast future outcomes. Entropy values for each variable are calculated, with subtracting these values from one yielding information values. A higher information gain indicates a superior attribute, positioning it higher in the tree. The Gini index gauges how frequently a randomly chosen element would be incorrectly identified, with lower values signifying better attributes. While decision trees are straightforward to interpret, they may encounter issues such as overfitting when handling datasets with numerous features. Hence, it's pivotal to discern when to halt tree growth. Two common methods for preventing overfitting are pre-pruning, which halts growth early but requires selecting a stopping point, and post-pruning, which involves cross-validation to determine if expanding the tree improves results or leads to overfitting. The structure of a decision tree encompasses a root node, splitting, decision nodes, terminal nodes, sub-trees, and parent nodes. The induction process comprises two primary phases: the growth phase, where training data is recursively partitioned to form the tree, and the pruning phase, which assesses whether expanding the tree enhances results or leads to overfitting. Decision trees possess a natural "if", "then", "else" construction, facilitating their integration into programmatic structures. Gini index can be found with the given formula:

$$G=\sum pi*(1-pi)$$

$$\text{for } i=1\ldots n \tag{1}$$

A decision tree provides a simple approach to analysis. However, when confronted with datasets containing multiple features, there's a danger of overfitting, where the model becomes overly customized to the training data and performs poorly on new data. Thus, it's crucial to identify the appropriate moment to halt the tree's expansion. Two common techniques for averting overfitting are pre-pruning, which involves stopping the tree's growth prematurely, although determining the optimal stopping point can be challenging, and post-pruning, which entails cross-validation to ascertain whether expanding the tree enhances its performance or exacerbates overfitting. The architecture of a decision tree encompasses various elements such as root nodes, decision nodes, terminal nodes, sub-trees, and parent nodes. The construction of a decision tree typically entails two phases: the growth phase, during which the training data is partitioned recursively, resulting in a tree structure, and the pruning phase, where the tree is refined to mitigate overfitting. Decision trees exhibit an inherent "if-then-else" format, rendering them easily adaptable into programming paradigms.

- *Naïve Bayes*

Naive Bayes models are classifiers based on probability theory, specifically the Bayes Theorem. They typically require less training data compared to other classifiers like neural networks and support vector machines, and they have fewer parameters. Additionally, Naive Bayes models are adept at filtering out irrelevant inputs and noise. However, they make the simplifying assumption that input variables are independent, which is often not the case in real-world classification tasks. Despite this limitation, Naive Bayes models have been successful in various applications. They work by calculating the probability of a given instance belonging to a certain class and selecting the class with the highest probability. Although they assume independence among variables, they can still yield good results in many scenarios. Recent research has focused on enhancing Naive Bayes classifiers, as they offer a straightforward and efficient approach to classification based on Bayes theorem. It is represented below:

$$P(X|Y)=P(Y|X)P(X)P(Y) \tag{2}$$

The foundational concept of this algorithm operates on the assumption that each variable independently and uniformly influences the outcome. As a consequence, each feature is perceived as unrelated to others and carries an equal weight in determining the output. Consequently, applying the naive Bayes theorem directly to real-world problems may yield suboptimal results, potentially resulting in reduced accuracy. Gaussian Naive Bayes represents a specific instantiation of the naive Bayes approach, presuming that features adhere to a normal distribution. It posits that features follow a Gaussian distribution and assigns them conditional probabilities accordingly. The theorem for Gaussian Naive Bayes is articulated as follows:

P(x_i|y)=12πσ2ye^(−(xi−μy)22σ2y) (3)

- *Random Forest*

Random Forest (RF) stands as a non-parametric method utilizing classification techniques. It rapidly categorizes data by employing multiple decision trees. Each tree selects a random subset of input variables, and their collective output improves inference from the data. Random forests, also known as random decision forests, serve as ensemble methods for classification, regression, and similar tasks. They create numerous decision trees during training and determine the most frequent class (for classification) or average prediction (for regression) from the individual trees. This approach mitigates the risk of decision trees overfitting their training data. The rotation forest algorithm aims to construct classifiers by extracting attributes from the dataset. It randomly divides the attribute set into K subsets to build accurate and robust classifiers. Feature selection poses a primary challenge in decision tree models, with various strategies available. Random forest addresses this challenge by searching for the best feature among a random subset, rather than consistently choosing the most significant one when splitting nodes. Furthermore, it can introduce additional randomness by employing random thresholds for each feature instead of seeking the optimal one. Tweaking specific parameters within the model can enhance its performance. Parameters such as max features, n estimators, and min sample leaf contribute to improving prediction accuracy. Conversely, parameters like n jobs and random state aid in accelerating the model's execution. In this study, adjustments were made to the n estimators parameter, determining the number of trees to grow, and setting the random state parameter to improve both the accuracy and speed of the model.

- Randomly sample K data points from the training dataset.
- Utilize these K data points to construct decision trees.
- Determine the desired number of trees, denoted as N, and iterate through steps (i) and (ii).
- Aggregate the predictions of the N trees to classify a new data point. Assign the new data point to the category with the highest predicted probability.

- *K-Nearest Neighbors (K-Nn)*

K-Nearest Neighbors (KNN) is an instance-based, non-parametric machine learning algorithm utilized for classification and regression tasks. Unlike conventional methods that construct explicit models from training data, KNN retains all instances and classifies new ones based on similarity. When presented with a new data point, KNN identifies the K nearest neighbors from the training set using a chosen distance measure like Euclidean or Manhattan distance. These neighbors are determined by comparing the distance between the new data point and each instance in the training set. Once the K nearest neighbors are determined, KNN predicts the class of the new data point in classification tasks by majority voting among its neighbors' classes. In regression tasks, KNN predicts the target value of the new data point by averaging its neighbors' target values. KNN is valued for its simplicity and intuitive nature, making it straightforward to understand and implement. However, the selection of the parameter K, representing the number of neighbors to consider, and the choice of an appropriate distance metric are pivotal decisions that significantly influence the algorithm's performance. Despite its simplicity, KNN has demonstrated effectiveness in various domains, including pattern recognition, medical diagnosis, and recommendation systems. Its ability to adapt to complex data distributions without making strong assumptions makes it a versatile tool in machine learning applications. If the number of neighbors is denoted by N in K-NNs, then N samples are evaluated using the specified distance metric value Minkowski

Distance: Dist(x, y) = $\boxed{\left(\sum_{i=1}^{n}|x_i - y_i|^p\right)^{\frac{1}{p}}}$ (4)

When p=1, it represents Manhattan distance; when p=2, it signifies Euclidean distance; and when p=∞, it denotes Chebyshev distance. Despite the array of choices available, Euclidean distance remains the commonly adopted metric. Within the collection of K neighbors, the process evaluates the distribution of information across each class. Following this assessment, the algorithm assigns the new data point to the class with the highest occurrence.Top of Form

- *Logistic Regression*

Logistic regression is a statistical method primarily utilized in binary classification tasks, aiming to predict the probability of an outcome belonging to one of two classes. Unlike linear regression, which forecasts continuous values, logistic regression estimates the likelihood of the binary outcome using a logistic or sigmoid function. Throughout the training process, logistic regression learns the optimal coefficients by maximizing the likelihood of observed data or minimizing an appropriate loss function, such as binary cross-entropy loss. Following training, logistic regression applies the learned coefficients to new input data, passing the result through the logistic function to generate predictions. If the predicted probability exceeds a specified threshold, the input is classified as belonging to the positive class; otherwise, it is categorized as belonging to the negative class. Logistic regression is extensively applied across diverse domains, including healthcare, finance, and marketing, due to its simplicity, interpretability, and ability to provide probabilistic forecasts for binary classification tasks.

- *Support Vector Machine (SVM)*

The Support Vector Machine (SVM) is a flexible supervised machine learning method primarily utilized for classification tasks, although it can also be adapted for regression purposes. Its operation revolves around identifying the optimal hyperplane that effectively separates data points into different classes within the feature space. The core principle of SVM is to maximize the margin between this hyperplane and the nearest data points from each class, known as support vectors, thereby enhancing its ability to generalize and withstand noise.

During the training phase, SVM determines the optimal hyperplane by identifying support vectors and defining coefficients that govern the hyperplane's orientation. This is achieved through an optimization process aimed at maximizing the margin while simultaneously minimizing classification errors. In cases where the data lacks linear separability, SVM employs kernel functions to transform the input space into a higher-dimensional feature space, potentially facilitating separability. Common kernel functions include linear, polynomial, radial basis function (RBF), and sigmoid kernels.

A notable advantage of SVM lies in its effectiveness at handling high-dimensional data, making it suitable for tasks with numerous features. Additionally, SVM demonstrates robust generalization to unseen data and is less prone to overfitting, particularly when the regularization parameter is appropriately tuned.

Nonetheless, SVMs can be influenced by the choice of hyperparameters such as the regularization parameter and the selection of the kernel function, necessitating careful calibration and validation. Despite this sensitivity, SVM finds widespread application across diverse domains such as image classification, text classification, bioinformatics, and finance, due to its effectiveness, adaptability, and ability to handle complex classification tasks involving high-dimensional data.

The concept of a hyperplane is central to SVM, representing a boundary within an n-dimensional space. This hyperplane, which exists in (n - 1) dimensions, defines a level subspace that may not intersect the origin. Visualizing a hyperplane in higher dimensions presents challenges, hence the utilization of a (n - 1) dimensional level subspace remains pertinent. Constructing an SVM classifier is straightforward when a separating hyperplane is discernible. However, if the dataset's categories cannot be adequately delineated by a hyperplane, expanding the feature space using Gaussian radial basis function (RBF), sigmoid function, cubic, quadratic, or even higher order polynomial functions becomes necessary. The formulation of the hyperplane in p-dimensions can be expressed as follows:

$$\beta 0 + \beta 1X1 + \beta 2X2 + \ldots\ldots\ldots + \beta pXp = 0 \qquad (5)$$

Where X1, X2,…, and Xp are the data points in the sample space of p-dimension and β0, β1, β2,…, and βp are the hypothetical values.

V. CONFUSION MATRIX

The criteria for measuring the efficiency of the models are as follows:

- Accuracy: Accuracy determines how accurate the model has predicted both positive and negative results and to know whether the model is over fitted or not or is there any biasedness in the model towards a particular class or not.

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+FN+FP+TN)}$$

- Sensitivity: Sensitivity, also referred to as Recall, indicates the ratio of actual positive instances correctly classified as positive (true positives) among all positive cases.

$$\text{Sensitivity}= \frac{(TP)}{(TP+FN)}$$

- Specificity: Specificity gives the measure of the proportion of those values that got predicted as negatives or true negatives.

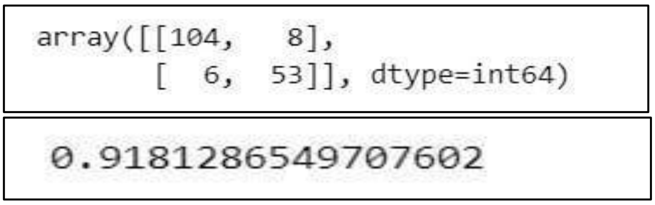
$$\text{Specificity}=\frac{(TN)}{(FP+TN)}$$

- Precision: Precision is one indicator of a machine learning model's performance – the quality of a positive prediction made by the model. Precision refers to the number of true positives divided by the total number of positive predictions (i.e., the number of true positives plus the number of false positives).

$$\text{Precision}=\frac{(TP)}{(FP+TP)}$$

VI. RESULT AND ANALYSIS

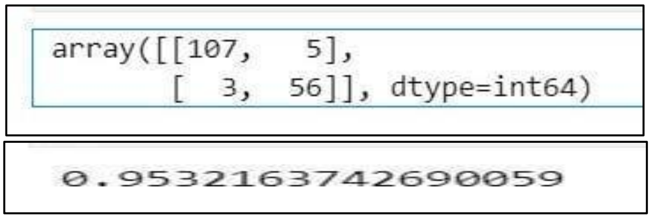
Next, we'll gather the results from different machine learning algorithms and evaluate them against our performance criteria to identify the model that best meets our needs. We'll compare the confusion matrices and accuracy scores of the various models to make our assessment.



```
array([[104,   8],
       [  6,  53]], dtype=int64)
```

```
0.9181286549707602
```

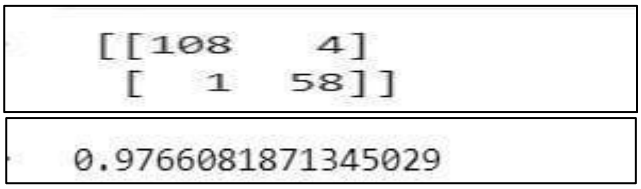
Fig 2: Decision Tree Confusion Matrix and its Accuracy



```
array([[107,   5],
       [  3,  56]], dtype=int64)
```

```
0.9532163742690059
```

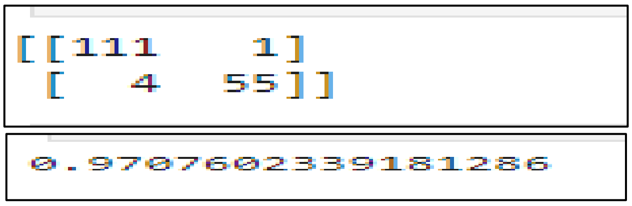
Fig 3: Naïve Bayes Confusion Matrix and its Accuracy



```
[[108   4]
 [  1  58]]
```

```
0.9766081871345029
```

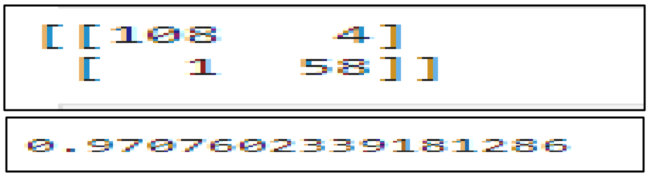
Fig 4: Random Forest Confusion Matrix and its Accuracy



```
[[111    1]
 [  4   55]]
```

```
0.9707602339181286
```

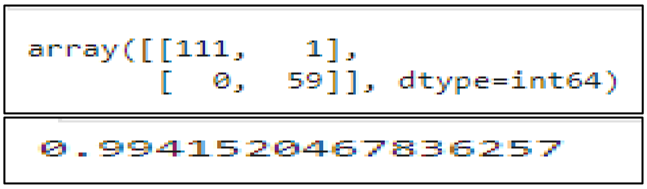
Fig 5: KNN Confusion Matrix and its Accuracy



```
[[108     4]
 [   1   58]]
```

```
0.9707602339181286
```

Fig 6: Logistic Regression Confusion Matrix and its Accuracy



```
array([[111,   1],
       [  0,  59]], dtype=int64)
```

```
0.9941520467836257
```

Fig 7: Support Vector Machine Confusion matrix and its accuracy

VII. CONCLUSION

To uncover the outcomes, we utilized all 529 data entries from the dataset. These were divided into two segments: 70% were allocated for training the model, while the remaining 30% were reserved for testing purposes. Across six distinct machine learning models, the Support Vector Machine classifier emerged with the highest count of true negatives, numbering 59, followed by KNN, Logistic Regression, Random Forest, Naïve Bayes, and Decision Tree, in that order. Conversely, the Decision Tree model exhibited the highest count of false negatives, totaling 5, followed by Naïve Bayes. Additionally, the Decision Tree produced the greatest number of false positives, with 9, followed by Naïve Bayes. In terms of accuracy, Random Forest achieved 97%, Decision Tree 91%, Naïve Bayes 95%, KNN 97%, Logistic Regression 97%, and SVM 99%.

REFERENCES

[1]. sjoc Park SH, Han K. Methodologic guide for evaluating clinical performance and effect of artificial intelligence technology for medical diagnosis and prediction. Radiol Soc N Am. 2018;286(3):800–9.

[2]. Breast Cancer: Statistics, Approved by the Cancer.Net Editorial Board, 04/2017. [Online]. Available: http://www.cancer.net/cance r-types/breast-cancer/statistics. Accessed 26 Aug 2018.

[3]. Mori M, Akashi-Tanaka S, Suzuki S, Daniels MI, Watanabe C, Hirose M, Nakamura S. Diagnostic accuracy of contrast-enhanced spectral mammography in comparison to conventional full-feld digital mammography in a population of women with dense breasts. Springer. 2016;24(1):104–10.

[4]. Kurihara H, Shimizu C, Miyakita Y, Yoshida M, Hamada A, Kanayama Y, Tamura K. Molecular imaging using PET for breast cancer. Springer. 2015;23(1):24–32.

[5]. Azar AT, El-Said SA. Probabilistic neural network for breast cancer classification. Neural Comput Appl. 2013;23

[6]. Nagashima T, Suzuki M, Yagata H, Hashimoto H, Shishikura T, Imanaka N, Miyazaki M. Dynamic-enhanced MRI predicts metastatic potential of invasive ductal breast cancer. Springer. 2002;9(3):226–30.

[7]. Park CS, Kim SH, Jung NY, Choi JJ, Kang BJ, Jung HS. Interobserver variability of ultrasound elastography and the ultrasound BI-RADS lexicon of breast lesions. Springer. 2013;22(2):153–60.

[8]. Ayon SI, Islam MM, Hossain MR. Coronary artery heart disease prediction: a comparative study of computational intelligence techniques. IETE J Res. 2020; https://doi.org/10.1080/03772063.2020.1713916.

[9]. Muhammad LJ, Islam MM, Usman SS, Ayon SI. Predictive data mining models for novel coronavirus (COVID-19) infected patients’ recovery. SN Comput Sci. 2020;1(4):206.

[10]. Islam MM, Iqbal H, Haque MR, Hasan MK. Prediction of breast cancer using support vector machine and K-Nearest neighbours. In: Proc. IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Dhaka, 2017, pp. 226–229.

[11]. Haque MR, Islam MM, Iqbal H, Reza MS, Hasan MK. Performance evaluation of random forests and artificial neural networks for the classification of liver disorder. In: Proc. International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2), Rajshahi, 2018, pp. 1–5.

[12]. Cancer Prediction”, by Yixuan Li, Zixuan Chen October 18, 2018

[13]. “Breast Cancer Prediction and Detection Using Data Mining Classification Algorithms: A Comparative Study” by Mumine Kaya Keles, Feb 2019

[14]. “Breast Cancer Prediction Using Data Mining Method” by Haifeng Wang and Sang Won Yoon, Department of Systems Science and Industrial Engineering State University of New York at Binghamton Binghamton, May 2015.

[15]. “Machine Learning with Applications in Breast Cancer Diagnosis and Prognosis” by Wenbin Yue, Zidong Wang, 9 May 2018.

[16]. WHO | Breast cancer’, WHO. http://www.who.int/cancer/prevention/diagnosis-screening/breast-cancer/en/ (accessed Feb. 18, 2020).

[17]. Datafloq - Top 10 Data Mining Algorithms, Demystified. https://datafloq.com/read/top-10-data-mining-algorithmsdemystified/1144. Accessed December 29, 2015.

[18]. S. Nayak and D. Gope, "Comparison of supervised learning algorithms for RF-based breast cancer detection," 2017 Computing and Electromagnetics International Workshop (CEM), Barcelona, 2017, pp.

[19]. B.M. Gayathri and C. P. Sumathi, "Comparative study of relevance vector machine with various machine learning techniques used for detecting breast cancer," 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Chennai, 2016, pp. 1-5.

[20]. H. Asri, H. Mousannif, H. A. Moatassime, and T. Noel, ‘Using Machine Learning Algorithms for Breast Cancer Risk Prediction and Diagnosis’, Procedia Computer Science, vol. 83, pp. 1064–1069, 2016, doi: 10.1016/j.procs.2016.04.224.

[21]. Y. khoudfi and M. Bahaj, Applying Best Machine Learning Algorithms for Breast Cancer Prediction and Classification, 978-1-5386- 4225- 2/18/$31.00 ©2018 IEEE.

[22]. L. Latchoumi, T. P., & Parthiban, “Abnormality detection using weighed particle swarm optimization and smooth support vector machine,” Biomed. Res., vol. 28, no. 11, pp. 4749–4751, 2017.

[23]. H. Osman, “An Enhanced Breast Cancer Diagnosis Scheme based on Two-Step-SVM Technique,” Int. J. Adv. Comput. Sci. Appl., vol. 8, no. 4, pp. 158–165, 2017.

[24]. Noble WS. What is a support vector machine? Nat Biotechnol. 2006;24(12):1565-1567. doi:10.1038/nbt1206-1565.

[25]. Larose DT. Discovering Knowledge in Data. Hoboken, NJ, USA: John Wiley & Sons, Inc.; 2004.

[26]. Hastie T, Tibshirani R, Friedman J. The elements of statistical learning. New York, NY: Springer-Verlag;2001.

[27]. Quinlan JR. C4.5: Programs for Machine Learning.; 2014:302. https://books.google.com/books?hl=fr&lr=&id=b3ujBQAAQBAJ&pgis=1.

[28]. “UCI Machine Learning Repository: Breast Cancer Wisconsin (Diagnostic) Data Set.”

[29]. Fabian Pedregosa and all (2011). "Scikit-learn: Machine Learning in Python". Journal of Machine Learning Research. 12: 2825–2830.

[30]. Arefan D, Mohamed AA, Berg WA, Zuley ML, Sumkin JH, Wu S. Deep learning modeling using normal mammograms for predicting breast cancer risk. *Med Phys*. 2020;47(1):110–8. doi: 10.1002/mp.13886. [PMC Free Article] [PMC free article] [PubMed] [Cross Ref] [Google Scholar]

[31]. Yanes T, Young MA, Meiser B, James PA. Clinical applications of polygenic breast cancer risk: a critical review and perspectives of an emerging field. *Breast Cancer Res*. 2020;22(1):21. doi: 10.1186/s13058-020-01260-3. [PMC Free Article] [PMC free article] [PubMed] [Cross Ref] [Google Scholar]

[32]. Feld SI, Woo KM, Alexandridis R, Wu Y, Liu J, et al. Improving breast cancer risk prediction by using demographic risk factors, abnormality features on mammograms and genetic variants. *AMIA Annu Symp Proc*. 2018; 2018:1253–62. [PMC Free Article] [PMC free article] [PubMed] [Google Scholar]

[33]. Behravan H, Hartikainen JM, Tengström M, Kosma VM, Mannermaa A. Predicting breast cancer risk using interacting genetic and demographic factors and machine learning. *Sci Rep*. 2020;10(1):11044. doi: 10.1038/s41598-020-66907-9. [PMC Free Article] [PMC free article] [PubMed] [Cross Ref] [Google Scholar]

[34]. Dai B, Chen RC, Zhu SZ, Zhang WW. Using random forest algorithm for breast cancer diagnosis. 2018 International Symposium on Computer, Consumer and Control (IS3C); Taichung, Taiwan: IEEE; 2018. p. 449-52. doi: 10.1109/IS3C.2018.00119. [Cross Ref] [Google Scholar].