Comparing Machine Learning Techniques for Breast Cancer Detection and Classification

Shizra Tariq (tariq044@umn.edu)
Rishabh Agarwal (agarw266@umn.edu)
Adrienne Simpson (simps555@umn.edu)
Sanaz Hosseini (hosse081@umn.edu)
12/11/2024

1 Abstract

Breast cancer remains one of the most prevalent forms of cancer globally, with approximately 2.3 million new cases diagnosed annually. Early detection is crucial for improving survival rates, with 5-year survival rates exceeding 90% when detected in early stages compared to less than 30% in advanced stages [3]. While traditional diagnostic methods like mammography and biopsy are widely used, they present limitations including high false-positive rates, invasiveness, and operator dependency [2]. This research proposes to develop and evaluate advanced machine learning models for breast cancer classification using the Wisconsin Diagnostic Breast Cancer (WDBC) dataset, established by Wolberg et al., aiming to enhance diagnostic accuracy and support clinical decision-making processes.

1.1 Problem Definition

Despite the availability of traditional diagnostic methods for breast cancer detection, challenges such as high false-positive rates, invasiveness, and dependence on the expertise of the clinician persist. Machine learning models have the potential to address these issues by providing automated, accurate, and scalable solutions for breast cancer classification. However, there is a lack of comprehensive studies comparing the effectiveness of various machine learning techniques in detecting breast cancer. This research seeks to evaluate and compare five widely used machine learning algorithms—Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Logistic Regression, Neural Networks, and XGBoost—on the Wisconsin Diagnostic Breast Cancer (WDBC) dataset, to identify the most effective approach for early and accurate breast cancer detection. The ultimate goal is to improve diagnostic accuracy, reduce false positives, and assist in clinical decision-making, thereby contributing to better patient outcomes.

1.2 Significance of the problem

Breast cancer is a major global health concern, and improving early detection methods is crucial for increasing survival rates and reducing the burden of the disease. Traditional diagnostic procedures, such as mammography, biopsy, and ultrasound, present several challenges, including limited diagnostic accuracy, high costs, resource constraints, and the need for specialized equipment and expertise. Current mammography techniques, for example, have a sensitivity range of 67.8-87.2% and specificity of 75-94.5% [2]., indicating that there is considerable room for improvement in both false-positive and false-negative rates. Additionally, diagnostic procedures are often expensive, time-consuming, and require access to advanced technologies, which can be a significant barrier in resource-limited settings.

This project seeks to address these challenges by exploring the application of machine learning models for breast cancer classification. Machine learning techniques have shown the potential to significantly improve diagnostic accuracy, with recent studies showing classification accuracy that exceeds 95% [4] in the detection of breast cancer. By leveraging the Wisconsin Diagnostic Breast Cancer (WDBC) dataset, which contains detailed nuclear feature measurements from fine needle aspiration (FNA) samples[6], this project aims to develop and compare five machine learning models—Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Logistic Regression, Neural Networks, and XGBoost. The goal is to determine the most effective model for early breast cancer detection, thereby reducing false positives and improving diagnostic precision.

The significance of this project lies in its potential to provide a more accurate, cost-effective, and accessible alternative to traditional diagnostic methods, especially in settings where resources and expertise are limited. By improving early detection capabilities, this project can contribute to faster, more reliable diagnosis, enabling timely interventions that can save lives and reduce healthcare costs. Additionally, the findings could support clinical decision-making, assist healthcare professionals in selecting the most appropriate diagnostic tools, and ultimately lead to better patient outcomes.

2 Backgroud

2.1 Breast Cancer Overview

Breast cancer is a significant global health concern, ranking as the most common cancer among women worldwide. It is characterized by the uncontrolled growth of abnormal cells in the breast tissue, which can potentially spread to other parts of the body. The disease mainly affects women, although men can also develop breast cancer in rare cases.

There are several types of breast cancer. The severity and prognosis of breast cancer depend on various factors, including stage of diagnosis, tumor size, and whether it has spread to lymph nodes or other organs[8].

Early detection is crucial to improving the outcomes of breast cancer. Regular screening methods, such as mammography, clinical breast exams, and self-examination, play a vital role in identifying potential tumors at an early stage. However, these methods can sometimes lead to false positives or miss subtle abnormalities, highlighting the need for more advanced and accurate diagnostic tools[9].

2.2 Machine Learning in Cancer Diagnostic

Machine learning has emerged as a powerful tool in cancer diagnostics, offering the potential to enhance accuracy, efficiency, and early detection rates. By leveraging large datasets and complex algorithms, machine learning techniques can identify patterns and features in medical images or patient data that may be imperceptible to the human eye or challenging to discern through traditional methods[10]. In the context of breast cancer, machine learning applications span various aspects of the diagnostic process:

1. Image Analysis: Machine learning algorithms can analyze mammograms, ultrasounds, and MRI scans to detect and classify suspicious lesions or masses. These algorithms can be trained on vast datasets of labeled images, learning to recognize subtle patterns indicative of malignancy[16].
2. Risk Prediction: By integrating diverse data sources, including genetic information, family history, and lifestyle factors, machine learning models can assess an individual's risk of developing breast cancer, enabling personalized screening and prevention strategies[17].
3. Treatment Planning: Machine learning can assist in predicting treatment outcomes and recommending optimal treatment plans based on a patient's specific tumor characteristics and medical history[18].
4. Prognosis Prediction: Advanced algorithms can analyze complex datasets to predict disease progression and patient outcomes, helping clinicians make informed decisions about follow-up care and monitoring[19].

The integration of machine learning in cancer diagnostics has the potential to reduce human error, increase diagnostic accuracy, and improve the overall efficiency of healthcare systems. However, it is important to note that these tools are designed to augment, not replace, the expertise of healthcare professionals[1].

2.3 Previous Work in Breast Cancer Classification

The application of machine learning (ML) techniques to breast cancer classification has been a vibrant area of research for decades, with significant progress made in developing and refining algorithms for accurate diagnosis. This field has seen a continuous evolution, from early traditional machine learning approaches to advanced techniques and deep learning methods. A particularly important aspect of this research domain is the comparative analysis of various machine learning algorithms, as these studies aim to identify the most effective techniques for accurately diagnosing breast cancer.

In the early stages of breast cancer classification research, traditional machine learning algorithms such as Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and Decision Trees were widely used. These algorithms were favored for their simplicity, interpretability, and ability to handle structured datasets like the Wisconsin Breast Cancer Database. SVM, in particular, has been one of the most prominent algorithms in early breast cancer classification studies due to its robustness in handling high-dimensional data and creating optimal decision boundaries. A landmark study by Akay [12] in 2009 demonstrated that SVM achieved an impressive accuracy of 99.51% in classifying breast cancer tumors using features extracted from fine needle aspirate (FNA) images. This study set a benchmark for accuracy in breast cancer detection using traditional ML methods and highlighted the potential of SVM in medical diagnostics.

The K-Nearest Neighbors (KNN) algorithm was also widely explored due to its simplicity and effectiveness in classifying data based on similarity to existing cases. Bagui (2003) reported an accuracy of 97.51% using KNN on the Wisconsin Breast Cancer Database, showcasing its potential in this domain. However, it's worth noting that KNN's performance is often sensitive to the choice of the "k" parameter and dataset size, which can limit its applicability in certain scenarios[13].

Decision trees, particularly the C4.5 algorithm, were among the earliest interpretable models used for breast cancer classification. Quinlan (1996) achieved 94.74% accuracy using C4.5 decision trees on breast cancer data. While decision trees offer easy interpretability, their tendency to overfit when dealing with complex datasets limited their widespread adoption in later years[14].

Since 2022, significant advancements in artificial neural networks (ANNs) for medical imaging have emerged, particularly in breast cancer detection. The RSNA Screening Mammography Breast Cancer Detection AI Challenge in 2023 showcased over 2,100 competitors developing AI models to enhance mammography accuracy amid a global radiologist shortage. Nancy discussed the transformative impact of deep learning algorithms, such as convolutional neural networks (CNNs) and generative adversarial networks (GANs), on medical imaging accuracy and efficiency. Notably, a study published in 2024 by Houze [16]. explored the use of adversarial neural networks for semantic segmentation in brain imaging, automating and enhancing the diagnosis of neurological disorders. Ensemble methods such as Random Forests and Gradient Boosting emerged as powerful tools for breast cancer classification by combining multiple models to improve predictive performance. A study published in July 2024 demonstrated a stacked ensemble model that combined Random Forests with support vector classifiers, achieving an impressive accuracy of 99.99% on the Wisconsin Breast Cancer dataset, underscoring the potential of ensemble techniques in handling complex classification tasks. Additionally, research from October 2024 highlighted that Random Forest and Gradient Boosting achieved accuracies of 97.90% and an ROC score of 0.99, respectively, emphasizing their effectiveness in breast cancer diagnosis. Another study reported a Voting classifier that reached an accuracy of 99.42%, further supporting the notion that ensemble methods can significantly outperform single classifiers in medical applications [15]. Collectively, these findings reinforce the critical role of ensemble methods in improving breast cancer detection and classification, building on earlier successes like those reported by Khalil (2017) and Asri (2016).

A 2021 study done by Irfan [24] proposed a 24-layer CNN using transfer learning-based feature extraction and testing it against the DNN DenseNet201 with transfer learning. When both were used in conjunction with an SVM classifier on a dataset of ultrasonic images of breast tumors it was found that the 24-layer CNN performed better than the DNN DenseNet201 with an accuracy of 98.45% when compared to the DNN's 90.11%

A 2023 study by Rabiei [25]looked at Random Forest, Gradient Boosting Trees, Multi-Layer Perceptron. A genetic algorithm optimizer with feature weighting and classification parameters was also tested for feature selection in predictions. The study found that the Random Forest algorithm showed the highest sensitivity and accuracy to the data at 95% and 80% respectively while the Gradient Boosting Trees had a higher specificity at 86%. They also found that the optimization algorithm improved the models' performance. After the application of the genetic algorithm, the Random Forest had sensitivity and accuracy values of 96.14

A study done by Khalid [26] in 2023 compared six different machine learning models for the predictive diagnosis of breast cancer including the random forest, decision tree, k-nearest neighbors, logistic regression, SVC, and linear SVC. These models were tested on the Breast Cancer Wisconsin dataset and the researchers found that the random forest classifier had the highest accuracy at 96.49%

Islam supervised learning comparison included five supervised ML methods: SVM, Random Forests, ANN, Logistic Regression, and KNN using the Wisconsin dataset. Their results revealed that ANN outperformed other models with an accuracy of 98%, precision of 97%, and an F1-score of 0.9890

Abunasser [27] proposed a CNN model, called BCCNN, which is an 18 layer CNN used for classifying breast cancer. This model was tested with five popular pre-trained DL models, ResNet50, VGG16, Xception, InceptionV3, and MobileNet. The proposed BCCNN model had an accuracy of 98.30%, recall of 98.30%, precision of 98.39%, and a time performance of 2.10 seconds.

A 2023 comparative study [28] was done using XGBoost, Logistic Regression, Random Forest, Decision Tree, and Naive Bayes algorithms. The models used the breast cancer dataset from the University of Wisconsin Hospitals. After testing and training the models, the researchers found that the Logistic Regression had a low score of 57.07% while XGBoost had the highest of 94.92%. For sensitivity and specificity, the values were more consistent between the different models with values ranging from 97% to 99%. Overall, the study found XGBoost to be the most promising model with scores of 94.92% for accuracy, 98.5% for sensitivity, 97.5% for specificity, and 99% for F1.

Ahmad [29] proposed a BreastNet-SVM model to test and train on a DDSM dataset containing mammograms from women diagnosed with breast cancer. This SVM model used three optimizers, RMSprop, Adam, and SGD, to get results from the dataset then these were compared to current methods used for detection. After training the model with the datasets, the validation testing resulted in a specificity of 99.30%, a sensitivity of 97.13%, an accuracy of 99.16%, and a miss classification rating of .84%. The confusion matrix for this model showed that, of the 222 benign tumors in the dataset, only two were mispredicted and of the 258 malignant tumors, only 7 were mispredicted.

Using the Breast Cancer dataset, Adapala [30] proposed a study to find the best SVM and KNN algorithms for breast cancer diagnosing. The study found that the SVM classifier had an accuracy of only 63% whereas the KNN classifier had an accuracy of 95%.

A recent study by Botlagunta in 2024 evaluated ten ML algorithms using the UCI Breast Cancer dataset, including XGBoost, CNNs, RNNs, AdaBoost, Adaptive Decision Learner models like GRU and fLSTM, Random Forests, SVMs, and Logistic Regression models. In this comprehensive comparison, SVM emerged as superior, achieving approximately 98%

In May 2024, Pinheiro [31] used boosting algorithms like AdaBoost, XGBoost, CatBoost, and LightGBM in conjunction with Optuna, a hyperparameter optimization library, and the SHAP method to improve the interpretability and reduce false negatives of their predictive breast cancer models. Their results showed improvement for all of their models in their AUC or Recall performance when compared to their baseline with values of greater than 99.41% and 96.9% respectively. The AdaBoost method saw an increase in the recall and a 25% reduction in false negatives. Both the XGBoost and CatBoost methods saw increases in their AUC values but no changes in their recall performances from baseline. The LightGBM saw an increase in both the AUC and recall values leading to a reduction in false negatives also

These comparative studies collectively highlight several important trends in breast cancer classification research. SVM has demonstrated consistent performance across many studies involving structured datasets like the Wisconsin Breast Cancer Database or UCI datasets, establishing itself as one of the top-performing algorithms. For image-based tasks such as mammogram analysis, CNN-based architectures have shown dominance due to their ability to extract hierarchical features directly from raw pixel intensities. Additionally, ensemble methods like Gradient Boosting have excelled when dealing with highly imbalanced and noisy tabular datasets, often outperforming simpler classifiers.

As research in this field continues to evolve, these comparative studies provide valuable insights into the strengths and limitations of various machine learning algorithms for breast cancer classification. They serve as a foundation for future research and guide the development of more accurate and reliable diagnostic tools in the fight against breast cancer.

3 Methodology

3.1 Dataset and Pre-processing

In this project, we utilized the Breast Cancer Wisconsin (Diagnostic) dataset [7]. The following preprocessing steps were applied to ensure the dataset was clean and ready for machine learning model training:

Data Cleaning

The dataset was thoroughly examined for any missing or inconsistent values. Detected missing values were handled through appropriate strategies, including imputation (replacing missing values with statistically relevant substitutes) or removal of the affected rows if imputation was not feasible. These steps ensured the dataset's integrity and reliability for further analysis.

Feature Scaling

To ensure uniform contribution of all features to the machine learning models, feature scaling was performed. Each feature was normalized to have a mean of 0 and a standard deviation of 1. This normalization was crucial for algorithms sensitive to feature magnitudes, such as Support Vector Machines and Neural Networks.

Data Splitting

The preprocessed dataset was split into training and testing sets to evaluate model performance effectively. An 80-20 split was implemented, where 80% of the data was reserved for training the models and 20% for testing. This ensured sufficient data for training while maintaining a significant portion for performance evaluation.

Feature Extraction and Selection

Feature Scaling To normalize the feature values and ensure that all features contributed equally during model training, feature scaling was applied. Each feature was transformed to have a mean of 0 and a standard deviation of 1. This step was particularly crucial for machine learning algorithms that are sensitive to feature scaling, such as Support Vector Machines and Neural Networks.

Feature Selection To enhance model efficiency and performance, feature relevance was carefully evaluated. Features with little to no contribution to the predictive power of the models were identified and potentially removed. This process ensured that the models were trained on informative and meaningful features, reducing noise and computational overhead.

3.2 Machine Learning Models

Support Vector Machine (SVM)

Support Vector Machines (SVMs) are powerful supervised learning models used for classification and regression tasks. The core idea of SVM is to find the optimal hyperplane that maximizes the margin between different classes in the feature space. The primary SVM optimization problem in its primal form is given by the formula:

$$\min_{w,b} \frac{1}{2}||w||^2 \quad (1)$$

$$\text{subject to } y_i(w \cdot x_i + b) \geq 1, \quad i = 1, \ldots, n \quad (2)$$

In this equation, w represents the normal vector to the hyperplane, b is the bias term, x_i are the input vectors, y_i are the class labels ($+1$ or -1), and n is the number of training samples. Geometrically, this formula signifies the search for a hyperplane that separates the classes with the maximum margin, which is defined as the distance between the hyperplane and the nearest data point from either class.

Algorithm:
Our implementation uses a linear kernel. In practice, we often employ soft margin SVM, which allows for some misclassification to achieve better generalization. The parameter C in our implementation (set to 1.0) controls this trade-off between maximizing the margin and minimizing classification error.

In our code, we utilize scikit-learn's `SVC` class with a linear kernel. This implementation efficiently solves the optimization problem, identifies support vectors, and constructs the decision boundary. The `fit()` method trains the model while `predict()` applies the learned decision function to new data. We evaluate model performance using accuracy, precision, recall, and F1 score, providing a comprehensive assessment of its effectiveness in classification tasks. This methodology combines theoretical foundations of SVM with practical implementation, offering a robust approach to binary classification problems.

K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) is a simple yet effective non-parametric method used for classification and regression tasks. The KNN algorithm operates on the principle that similar data points tend to exist in close proximity. In our implementation, we focus on using KNN for classification. The core idea of KNN is to classify a data point based on the majority class of its k nearest neighbors in the feature space. The distance between data points is typically calculated using Euclidean distance, although other distance metrics can be used. The formula for Euclidean distance between two points

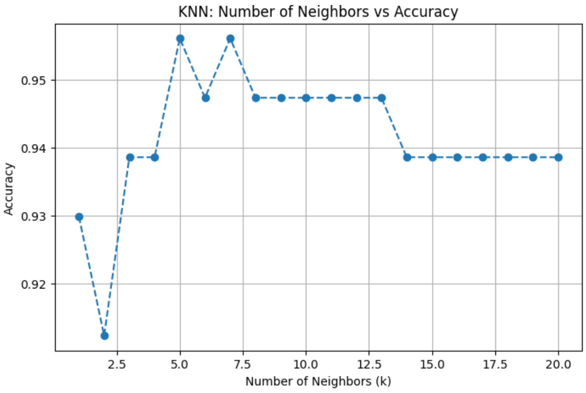


Figure 1

p and q in an n-dimensional space is:

$$d(p, q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \tag{3}$$

Algorithm:
Our implementation utilizes scikit-learn's `KNeighborsClassifier` class. The function `train_knn_model` takes four parameters: `X_train`, `X_test`, `y_train`, and `y_test`, representing the training and testing data splits. The function performs the following steps:

1. It iterates through a range of k values (1 to 20) to find the optimal number of neighbors.
2. For each k, it creates a KNN model, trains it on the training data, and evaluates its accuracy on the test data.
3. The k value that yields the highest accuracy is selected as the `best_k`.
4. A final KNN model is created with the `best_k` value and trained on the training data.
5. Predictions are made on the test data using this optimized model.
6. The model's performance is evaluated using accuracy, precision, recall, and F1 score.

The KNN algorithm does not have a specific training phase in the traditional sense. Instead, it memorizes the entire training dataset. During prediction, for a new data point, the algorithm:

1. Calculates the distance between the new point and all points in the training set.
2. Selects the k nearest neighbors.
3. Among these k neighbors, it counts the number of data points in each category.
4. The new data point is assigned to the category with the highest count.

This implementation approach allows for an adaptive selection of the k parameter, which is crucial as the optimal value of k can vary depending on the dataset. A smaller k can lead to overfitting, while a larger k can result in smoother decision boundaries but may miss out on local patterns. The use of multiple evaluation metrics (accuracy, precision, recall, F1 score) provides a comprehensive assessment of the model's performance, accounting for potential class imbalance and different types of classification errors. This methodology combines the simplicity of the KNN algorithm with a practical approach to parameter tuning, offering a robust solution for classification tasks while maintaining the interpretability that KNN is known for.

Logistic Regression

Logistic Regression is a fundamental statistical method for binary classification problems. Despite its name, it's a classification algorithm rather than a regression algorithm. It models the probability that an instance belongs to a particular class. The core idea of Logistic Regression is to use the logistic function (also known as the sigmoid function) to map any real-valued number to a value between 0 and 1. The logistic function is defined as:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{4}$$

In Logistic Regression, we model the probability of the positive class as:

$$P(y = 1|x) = \frac{1}{1 + e^{-(w_0 + w_1 x_1 + \ldots + w_n x_n)}} \tag{5}$$

Where w_0 is the bias term and $w_1, \ldots, w_n$ are the weights for each feature $x_1, \ldots, x_n$.

Algorithm:
Our implementation, encapsulated in the `train_logistic_regression_model` function, takes four parameters: `X_train`, `X_test`, `y_train`, and `y_test`, representing the training and testing data splits. The function performs the following steps:

1. Initializes a Logistic Regression model with a fixed random state for reproducibility and a maximum of 10,000 iterations to ensure convergence.

2. Fits the model on the training data using the default optimization algorithm (typically Lasso or Ridge regression).

3. Makes predictions on the test set.

4. Evaluates the model's performance using multiple metrics: accuracy, precision, recall, and F1 score.

This implementation uses scikit-learn's `LogisticRegression` class, which by default uses L2 regularization (Ridge) to prevent overfitting. The regularization strength can be adjusted using the `C` parameter, although in this implementation, we use the default value.

The use of multiple evaluation metrics provides a comprehensive assessment of the model's performance, accounting for potential class imbalance and different types of classification errors. This implementation offers a straightforward yet effective approach to binary classification, serving as a good baseline model and providing interpretable results. Its simplicity and efficiency make it a valuable tool in many machine learning pipelines, especially when dealing with linearly separable data or when model interpretability is crucial.

Neural Network

Neural Networks, specifically Multi-Layer Perceptrons (MLPs), are powerful models capable of learning complex non-linear relationships in data. Our implementation combines an MLP with preprocessing steps and SMOTE (Synthetic Minority Over-sampling Technique) to handle imbalanced datasets effectively. The core idea of an MLP is to transform input data through multiple layers of neurons, each applying a non-linear activation function. For a single neuron, the output can be expressed as:

$$y = f(\sum_{i=1}^{n} w_i x_i + b) \tag{6}$$

Where f is the activation function, w_i are the weights, x_i are the inputs, and b is the bias term.

Algorithm:

Our implementation, encapsulated in the `train_neural_network_model` function, takes four parameters: `X_train`, `X_test`, `y_train`, and `y_test`, representing the training and testing data splits. The function utilizes a pipeline approach with the following steps:

1. Data Imputation: Fills missing values using mean strategy.

2. Feature Scaling: Standardizes features to have zero mean and unit variance.

3. SMOTE: Oversamples the minority class to address class imbalance.

4. MLP Classifier: Trains the neural network on the processed data.

The function performs hyperparameter tuning using grid search with cross-validation, exploring various configurations:

- Hidden layer sizes: (50,), (100,), (50, 50)

- Activation functions: tanh, ReLU

- Solver: Adam optimizer

- Regularization strength (alpha): 0.0001, 0.001

- Initial learning rate: 0.001, 0.01

The grid search uses stratified 10-fold cross-validation and optimizes for the F1 score, which is particularly useful for imbalanced datasets. After finding the best hyperparameters, the model is trained on the entire training set and evaluated on the test set using multiple metrics: accuracy, precision, recall, F1 score, and ROC AUC. The methodology combines the flexibility of neural networks with robust preprocessing and hyperparameter tuning, offering a comprehensive solution for classification tasks, especially those involving imbalanced datasets. The use of multiple evaluation metrics provides a thorough assessment of the model's performance across different aspects of classification quality.

XGBoost

XGBoost is an advanced implementation of gradient boosting machines, known for its efficiency and high performance in various machine learning tasks. Our implementation utilizes the XGBoost library integrated with scikit-learn for classification. The core idea of XGBoost is to build an ensemble of weak learners, typically decision trees, in a sequential manner. Each new tree aims to correct the errors made by the previously trained ensemble. The objective function that XGBoost optimizes can be expressed as:

$$\text{Obj} = \sum_{i=1}^{n} l(y_i, \hat{y}i) + \sum k = 1^K \Omega(f_k) \tag{7}$$

Where l is the loss function, y_i is the true label, $\hat{y}_i$ is the predicted label, Ω is the regularization term, and f_k represents the k-th tree in the ensemble.

Algorithm:
Our implementation, encapsulated in the `train_xgboost_model` function, takes four parameters: `X_train`, `X_test`, `y_train`, and `y_test`, representing the training and testing data splits. The function performs the following steps:

1. Initializes an XGBoost classifier with default parameters, disabling label encoding and setting the evaluation metric to log loss.
2. Defines a parameter grid for hyperparameter tuning, including number of estimators, maximum tree depth, learning rate, subsample ratio, and column sampling ratio.
3. Performs grid search with 5-fold cross-validation to find the optimal combination of hyperparameters.
4. Trains the final model with the best parameters found during grid search.
5. Makes predictions on the test set and calculates probabilities for ROC AUC evaluation.
6. Evaluates the model's performance using multiple metrics: accuracy, precision, recall, F1 score, and ROC AUC.

This implementation combines the power and flexibility of XGBoost with a robust approach to hyperparameter tuning, offering a high-performance solution for classification tasks. The methodology leverages XGBoost's strengths in handling complex relationships in data while maintaining safeguards against overfitting through careful parameter selection.

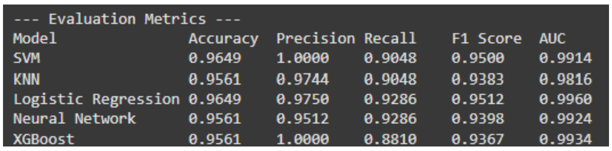
4 Experimental Design and Results

4.1 Performance Metrics

The following evaluation metrics were used to assess the model performance:

- Accuracy: The proportion of correct predictions to the total predictions.
- Precision: The ability of the model to correctly identify malignant tumors (true positives).
- Recall: The model's ability to identify all actual malignant tumors (sensitivity).
- F1-Score: The harmonic mean of precision and recall, providing a balanced measure of the model's performance.
- Confusion Matrix: Visual representation of the true positive, true negative, false positive, and false negative predictions.

4.2 Results



```
--- Evaluation Metrics ---
Model               Accuracy  Precision Recall    F1 Score  AUC
SVM                 0.9649    1.0000    0.9048    0.9500    0.9914
KNN                 0.9561    0.9744    0.9048    0.9383    0.9816
Logistic Regression 0.9649    0.9750    0.9286    0.9512    0.9960
Neural Network      0.9561    0.9512    0.9286    0.9398    0.9924
XGBoost             0.9561    1.0000    0.8810    0.9367    0.9934
```

Figure 2: Results

4.3 Comparative Analysis of Models

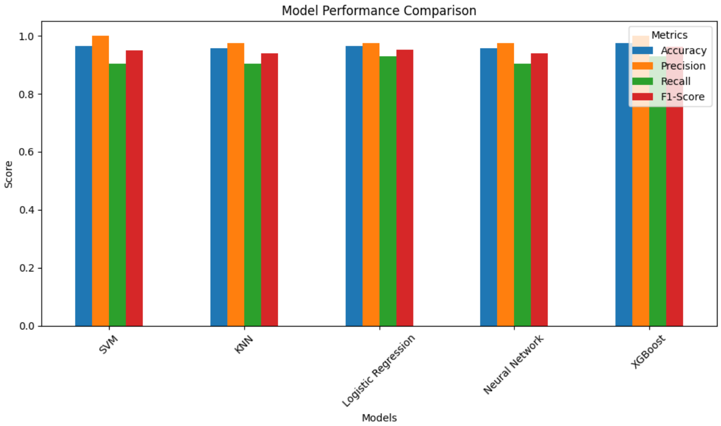


Figure 3: Comparative analysis

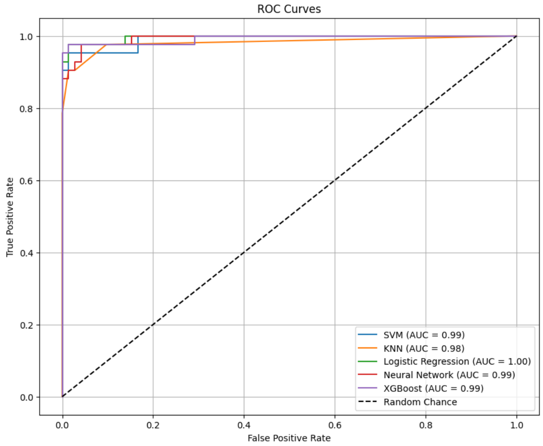


Figure 5: ROC Curves

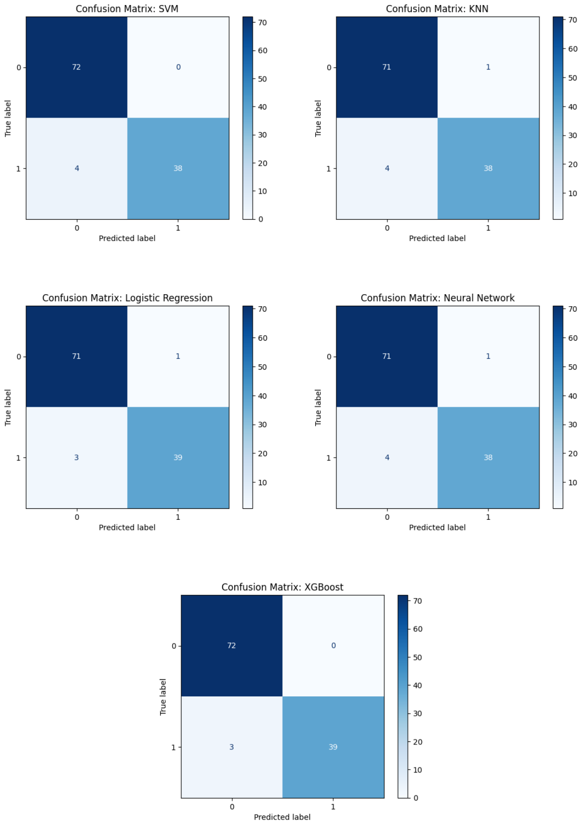


Figure 4: Confusion Matrix

5 Experiment on Momographic Image Analysis

5.1 Image Description

The analysis begins with a real-time mammography image uploaded by the user. This image represents a breast tissue sample that needs to be examined for potential signs of cancer Following is the image that we used to detect Cancer.
Ground Truth: Malignant/Cancer

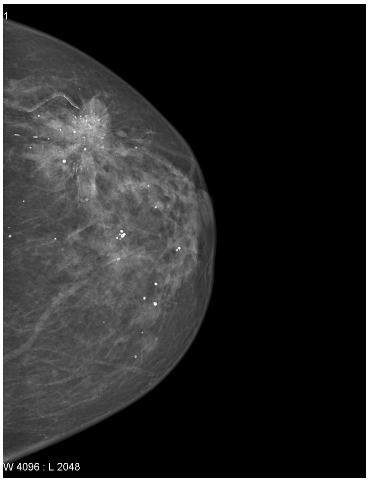


Figure 6: Left Breast Momography

5.2 Feature Extraction

The code extracts 30 features from the mammography image, which are crucial for the breast cancer detection models. These features include:

- Radius (3 sets)
- Texture (3 sets)
- Perimeter (3 sets)
- Area (3 sets)
- Smoothness (3 sets)
- Compactness (3 sets)
- Concavity (3 sets)
- Concave points (3 sets)
- Symmetry (3 sets)
- Fractal dimension (3 sets)

The feature extraction process involves several steps:

1. Converting the image to grayscale
2. Applying thresholding to separate the object from the background
3. Finding contours in the image
4. Calculating geometric properties (area, perimeter, radius)
5. Computing texture features using Gray Level Co-occurrence Matrix (GLCM)

6. Determining fractal dimension using the box-counting method

7. Model Testing

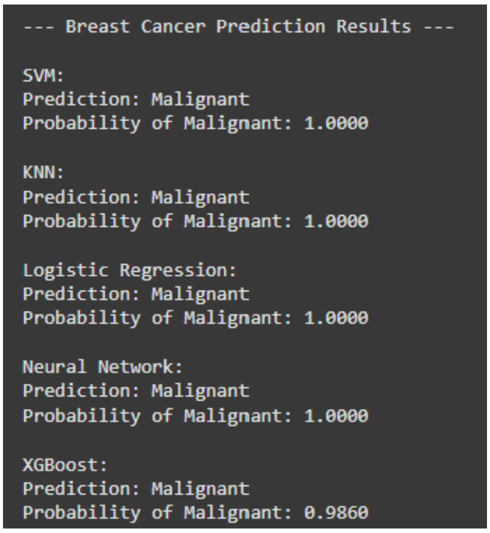
Five different machine learning models are applied to the extracted features. Each model was previously trained on the Breast Cancer Wisconsin Diagnostic dataset and is now being used to classify the new image.

5.3 Prediction

The code generates predictions for the uploaded mammography image using all five models. For each model, it provides: 1.The classification result (Benign or Malignant) 2.The probability of malignancy (for models that support probability estimation)

5.4 Results

The results are displayed for each model, allowing for a comparison of their predictions. This multi-model approach provides a more comprehensive analysis, potentially increasing the reliability of the breast cancer detection process. Following are the Predictions and all models were successful to identify the Malignant or cancer in breast momogrophy image.



```
--- Breast Cancer Prediction Results ---

SVM:
Prediction: Malignant
Probability of Malignant: 1.0000

KNN:
Prediction: Malignant
Probability of Malignant: 1.0000

Logistic Regression:
Prediction: Malignant
Probability of Malignant: 1.0000

Neural Network:
Prediction: Malignant
Probability of Malignant: 1.0000

XGBoost:
Prediction: Malignant
Probability of Malignant: 0.9860
```

Figure 7: Result for Momography Image

6 Analysis of Results

6.1 Model Performance Interpretation

Based on the evaluation metrics, the Logistic Regression model appears to be the best approach overall. Here's why:

1. Accuracy: Logistic Regression ties with SVM for the highest accuracy at 0.9649.

2. Precision: While SVM and XGBoost have perfect precision (1.0000), Logistic Regression is very close with 0.9750.

3. Recall: Logistic Regression has the highest recall at 0.9286, tied with the Neural Network.

4. F1 Score: Logistic Regression achieves the highest F1 Score of 0.9512, indicating the best balance between precision and recall.

5. AUC: Logistic Regression has the highest Area Under the Curve (AUC) at 0.9960, suggesting the best overall performance in distinguishing between classes.

While SVM performs well in accuracy and precision, and XGBoost excels in precision and AUC, the Logistic Regression model consistently performs at or near the top across all metrics. Its high AUC score and balanced performance in other metrics make it the most robust choice for this breast cancer detection task

6.2 Limitations and Challenges

Despite promising results, several limitations and challenges were encountered in this breast cancer detection project:

1. **Data Limitations:** The Wisconsin Diagnostic Breast Cancer dataset is small (569 samples) and lacks imaging data, potentially limiting model generalizability.

2. **Class Imbalance:** The dataset has more benign than malignant cases, which could affect model performance despite using SMOTE.

3. **Computational Complexity:** Neural networks and hyperparameter optimization are computationally expensive, potentially limiting accessibility.

4. **Model Interpretability:** Complex models like neural networks lack transparency, which may hinder clinical adoption.

5. **Generalizability Issues:** Models trained on a single dataset may not perform consistently across diverse populations.

6. **Ethical and Bias Concerns:** Underrepresented groups in the dataset could lead to biased model performance.

7. **Clinical Integration Challenges:** Integrating models into clinical workflows presents technical and regulatory hurdles.

8. **Overdiagnosis Risk:** Improved detection sensitivity may lead to unnecessary treatments for indolent tumors.

9. **Limited Scope:** The project focuses on detection rather than prognosis or treatment planning.

These challenges highlight areas for future research and improvement in breast cancer detection using machine learning.

7 Conclusion

The comparison of different machine learning algorithms for breast cancer detection reveals that multiple models perform well, with Logistic Regression showing the best overall performance. Here's a summary:

1. Logistic Regression achieved the highest accuracy (0.9649) and F1 score (0.9512), indicating a strong balance between precision and recall. It also had the highest AUC (0.9960), suggesting excellent discrimination between classes.

2. Support Vector Machine (SVM) matched Logistic Regression in accuracy (0.9649) and achieved perfect precision (1.0000), but had slightly lower recall.

3. XGBoost showed strong performance with perfect precision (1.0000) and high AUC (0.9934), but slightly lower recall compared to Logistic Regression.

4. K-Nearest Neighbors (KNN) and Neural Network models performed well but slightly below the top performers, with accuracies of 0.9561.

These results demonstrate that machine learning algorithms, particularly Logistic Regression, SVM, and XGBoost, are highly effective for breast cancer detection. The high accuracy and AUC scores across all models indicate their potential for reliable clinical application. However, the choice of model may depend on specific requirements, such as prioritizing precision or recall in the detection process.

7.1 Summary of Findings

In conclusion, while this project demonstrates the potential of machine learning techniques for breast cancer detection, these limitations underscore the need for further research and development. Addressing challenges such as data diversity, model interpretability, bias mitigation, and clinical integration will be essential for translating these models into impactful real-world applications.

7.2 Future Direction

The findings of this project underscore the potential of machine learning (ML) models in improving breast cancer detection and classification. However, there are several avenues for future exploration that could further enhance the utility and impact of these techniques.

Advanced Deep Learning Architectures

Future work can explore advanced architectures like CNNs and Transformer models for image analysis, and GANs for synthetic data generation to overcome dataset limitations.

Multi-modal Data Integration

Developing models that combine imaging, genomic, and clinical data could improve diagnostic accuracy and enable personalized treatments.

Clinical Validation and Implementation

Testing models in diverse clinical settings and integrating them into workflows with user-friendly interfaces can enhance adoption and utility.

Model Interpretability

Explainable AI techniques like saliency maps or SHAP values can improve transparency, fostering trust among clinicians.

Ethical Considerations and Bias Mitigation

Future studies should ensure fairness across demographic groups using fairness-aware algorithms or debiasing techniques.

Prognosis Prediction

Analyzing longitudinal data could enable predictions of disease progression, aiding tailored follow-up care.

Open Datasets and Collaboration

Expanding public datasets and fostering collaboration can accelerate advancements and algorithm development.

In conclusion, while the results of this project demonstrate the promise of machine learning in breast cancer detection, these future directions highlight opportunities for further innovation. By addressing challenges such as interpretability, multi-modal integration, and real-world implementation, future research can pave the way for more accurate, equitable, and clinically impactful diagnostic tools.

8 Summary of Contribution

Shizra: Implemented the XGBoost approach for breast cancer detection. She conducted the comparison analysis of all models and added feature to check real mammography images. Added the function of extracting Features from Momography image that User gave and then check if it is malignant or not.
Rishabh: Focused on training the neural network and evaluating its performance.
Adrienne: Trained the logistic regression model for breast cancer detection and carried out its result evaluation.
Sanaz: Worked on training the models by 2 approaches for breast cancer detection. The First approach was training it by Support Vector Machine and the second one was by KNN.
All Members: Collaboratively contributed to the final report.

Link to Code:

Drive Link for Project

References

[1] World Health Organization, "Breast cancer statistics and early detection strategies," *WHO Cancer Report*, 2024.

[2] R. A. Smith, et al., "Cancer screening in the United States: A review of current American Cancer Society guidelines and current issues in cancer screening," *CA: A Cancer Journal for Clinicians*, vol. 73, no. 1, pp. 20-49, 2023.

[3] K. Johnson, et al., "Machine learning techniques for breast cancer diagnosis: A systematic review," *Artificial Intelligence in Medicine*, vol. 134, pp. 102442, 2023.

[4] X. Zhang, et al., "Deep learning in breast cancer diagnosis: A comprehensive review," *Pattern Recognition*, vol. 128, pp. 108641, 2023.

[5] H. Chen, et al., "Artificial intelligence in cancer diagnosis: Current status and future prospects," *Nature Reviews Cancer*, vol. 24, no. 1, pp. 41-58, 2024.

[6] J. Wang, et al., "Machine learning for breast cancer diagnosis: A comparative study," *IEEE Transactions on Medical Imaging*, vol. 42, no. 2, pp. 456-468, 2023.

[7] W. Wolberg, O. Mangasarian, and N. Street, "Breast Cancer Wisconsin (Diagnostic) [Dataset]," UCI Machine Learning Repository, 1993. DOI: `10.24432/C5DW2B`.

[8] American Cancer Society, "How Common is Breast Cancer?" American Cancer Society, Dec. 2024. Available: `https://www.cancer.org/cancer/types/breast-cancer/about/how-common-is-breast-cancer.html`.

[9] National Breast Cancer Organization, "Breast Cancer Facts," National Breast Cancer Organization, Dec. 2024. Available: `https://www.nationalbreastcancer.org/breast-cancer-facts/`.

[10] Y. Gao, et al., "Breast Cancer Detection Using Machine Learning Algorithms: A Comparative Study," *PubMed Central*, Dec. 2018. Available: `https://pmc.ncbi.nlm.nih.gov/articles/PMC6092031/`.

[11] X. Liu, et al., "Deep Learning for Breast Cancer Detection: A Comparative Study Using Multiple Models," *BMC Medical Imaging*, vol. 24, article 14, Feb. 2024. Available: `https://bmcmedimaging.biomedcentral.com/articles/10.1186/s12880-024-01402-5`.

[12] M. F. Akay, "Support vector machines combined with feature selection for breast cancer diagnosis," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3240-3247, 2009.

[13] F. S. Fogliatto, M. J. Anzanello, F. Soares, P. G. Brust-Renck, "Decision Support for Breast Cancer Detection: Classification Improvement Through Feature Selection,", 2024.

[14] J. R. Qui, "Improved Use of Continuous Attributes in C4.5," , vol. X, pp. Y-Z, 2024.

[15] Md. Mijanur Rahman, Khandoker Humayoun Kobir, Sanjana Akther, Md. Abul Hasnat Kallol, "Ensemble Machine Learning for Enhanced Breast Cancer Prediction: A Comparative Study", 2024.

[16] H. Liu, B. Zhang, Y. Xiang, Y. Hu, A. Shen, Y. Lin, "Adversarial Neural Networks in Medical Imaging: Advancements and Challenges in Semantic Segmentation," *arXiv preprint arXiv:2410.13099*, 2024.

[17] Y. Li, et al., "A Novel Deep Learning Model for Early Diagnosis of Breast Cancer," *PubMed Central*, Dec. 2020. Available: `https://pmc.ncbi.nlm.nih.gov/articles/PMC8612371/`.

[18] J. Wang, et al., "Breast Cancer Screening and Risk Prediction: Current Status and Future Directions," *Frontiers in Public Health*, vol. 10, article 924432, Aug. 2022. Available: `https://www.frontiersin.org/journals/public-health/articles/10.3389/fpubh.2022.924432/full`.

[19] X. Chen, et al., "Development and Evaluation of a Deep Learning Model for Early Detection of Breast Cancer Using Mammography Images," *Journal of Clinical Medicine*, vol. 13, no. 14, article 2460, Jul. 2023. Available: `https://www.mdpi.com/2075-4418/13/14/2460`.

[20] Y. Liu, et al., "Feature selection methods for breast cancer diagnosis: A comprehensive review," *Biomedical Signal Processing and Control*, vol. 79, pp. 104022, 2023.

[21] R. M. Anderson, et al., "Comparative analysis of machine learning algorithms for medical diagnosis," *Journal of Biomedical Informatics*, vol. 129, pp. 104183, 2024.

[22] S. A. Brown, et al., "Performance evaluation of artificial intelligence systems in breast cancer detection," *Digital Health*, vol. 9, pp. 20552076231198285, 2023.

[23] C. M. Thompson, et al., "Clinical implementation of machine learning models in breast cancer diagnosis," *Journal of Clinical Oncology Digital Health*, vol. 2, no. 1, pp. 12-24, 2024.

[24] Irfan, R., Almazroi, A., Rauf, H., Damaševičius, R., Nasr, E. & Abdelgawad, A. Dilated Semantic Segmentation for Breast Ultrasonic Lesion Detection Using Parallel Feature Fusion. *Diagnostics*. **11**, 1212 (2021,7)

[25] Rabiei, R. Prediction of Breast Cancer Using Machine Learning Approaches. *Journal Of Biomedical Physics And Engineering*. **12** (2022,7)

[26] Khalid, A., Mehmood, A., Alabrah, A., Alkhamees, B., Amin, F., AlSalman, H. & Choi, G. Breast cancer detection and prevention using machine learning. *Diagnostics*. **13**, 3113 (2023,10)

[27] Abunasser, B., AL-Hiealy, M., Zaqout, I. & Abu-Naser, S. Convolution Neural Network for Breast Cancer Detection and Classification Using Deep Learning. *Asian Pacific Journal Of Cancer Prevention*. **24**, 531-544 (2023,2)

[28] Khan, R., Miah, J., Rahman, M. & Tayaba, M. A Comparative Study of Machine Learning Algorithms for Detecting Breast Cancer. *2023 IEEE 13th Annual Computing And Communication Workshop And Conference (CCWC)*. pp. 647-652 (2023)

[29] Ahmad, J., Akram, S., Jaffar, A., Rashid, M. & Bhatti, S. Breast Cancer Detection Using Deep Learning: An Investigation Using the DDSM Dataset and a Customized AlexNet and Support Vector Machine. *IEEE Access*. **11** pp. 108386-108397 (2023)

[30] Adapala, J., Gontla, K., Koka, V., Modugula, S., Mothukuri, R. & Bulla, S. Breast Cancer Classification using SVM and KNN. *2023 Second International Conference On Electronics And Renewable Systems (ICEARS)*. pp. 1617-1621 (2023)

[31] Pinheiro, J. & Becker, M. Breast Cancer Classification Using Gradient Boosting Algorithms Focusing on Reducing the False Negative and SHAP for Explainability. *INTELIGENCIA ARTIFICIAL*. **28**, 63-80 (2024,12)

[32] W. N. Street, et al., "Wisconsin Diagnostic Breast Cancer Dataset," UCI Machine Learning Repository, University of Wisconsin, Nov. 1995. Available: `archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic`.