Optimizing Breast Cancer Diagnosis: A Comparative Study of Machine Learning Models and Dimensionality Reduction Techniques

Dimitrios Galatidis and Angelos Tzourtzis
Students of M.Sc. Program in Digital Systems,
University of Piraeus

***Abstract*—Breast cancer diagnosis relies on timely and accurate detection to improve patient outcomes. This study evaluates the performance of machine learning models, including Logistic Regression, Support Vector Machines (SVM), Random Forest, K-Nearest Neighbors, and Perceptron, using the Wisconsin Breast Cancer Diagnosis dataset. The models were tested on full data, as well as on datasets processed with Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and feature selection. Logistic Regression and SVM achieved the highest accuracy and recall on the full dataset, while Random Forest demonstrated consistent robustness across preprocessing methods. Dimensionality reduction produced mixed results, with PCA preserving model performance and LDA proving limited due to its single-component constraint. Feature selection simplified the dataset, but occasionally reduced classification metrics. Overall, Logistic Regression, SVM and Random Forest emerged as reliable models for breast cancer classification, underscoring the potential of machine learning in enhancing diagnostic accuracy. Future work could explore hybrid and ensemble approaches to further improve outcomes.**

I. Introduction

BREAST cancer is one of the most prevalent and life-threatening diseases worldwide, accounting for approximately 2.3 million new cases and nearly 685,000 deaths annually according to the World Health Organization (WHO) in 2020. The high prevalence of breast cancer underscores the critical need for early detection and accurate diagnosis to improve patient results. Early-stage breast cancer has a five-year relative survival rate of 99%, whereas delayed diagnosis drastically reduces survival rates, highlighting the significance of timely and reliable diagnostic methods.

The rapid advancements in machine learning and computational technologies have opened new avenues for improving medical diagnostics. By leveraging data-driven insights, machine learning models can analyze complex medical datasets, such as histopathological images or fine-needle aspiration (FNA) cytology results, to identify patterns that may not be apparent through conventional methods. These models have the potential to assist healthcare professionals in making faster and more accurate decisions, reducing diagnostic delays, ultimately saving lives.

This study focuses on the application of supervised machine learning techniques to the Wisconsin Breast Cancer Diagnosis (WDBC) dataset, which contains real-valued features derived from FNA of breast tumors. These features represent key characteristics of cell nuclei, such as radius, texture and concavity, making them highly informative for classification tasks. The objective of this research is to classify tumors as malignant or benign using a variety of machine learning models and evaluate their performance across different preprocessing and dimensionality reduction scenarios.

A. Related Works

Several studies have explored the application of machine learning models to breast cancer diagnosis. For instance, Wolberg et al. [1] employed linear programming techniques to classify tumors with high accuracy, achieving an overall accuracy of 93% using minimal features extracted from FNA images. Street et al. [2] utilized decision tree-based methods, emphasizing the importance of feature selection and reporting a classification accuracy of 95%. Similarly, Bennett and Mangasarian [4] highlighted the benefits of dimensionality reduction through Principal Component Analysis (PCA), which improved computational efficiency while maintaining an accuracy of 92%.

Table I summarizes key results from these and other notable studies, providing benchmarks for comparison with the models evaluated in this research.

These works provide a solid foundation and highlight the diverse approaches used to tackle the problem of breast cancer diagnosis. They also underscore the importance of selecting appropriate preprocessing techniques, model types and evaluation metrics to achieve high classification accuracy.

The remainder of this paper consists of the methodology section details, the dataset, preprocessing techniques and the machine learning models employed. The results section presents a comprehensive evaluation of the models under different scenarios, followed by a discussion regarding their strengths and limitations. Finally, the paper concludes with insights and recommendations for future research directions in leveraging machine learning for breast cancer diagnosis.

II. Methodology

Our methodology includes several key steps. We begin by obtaining the dataset, which serves as the foundation for our analysis. The dataset undergoes preprocessing to prepare it for modeling, including cleaning, splitting and feature scaling.

Next, we apply a variety of supervised machine learning models, fine-tuning their hyperparameters using techniques such as GridSearchCV to optimize performance. These models are then evaluated on an unseen test set to measure their accuracy, recall and other relevant metrics.

Finally, we explore the impact of dimensionality reduction techniques, including Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA). These techniques are evaluated to determine whether they improve model performance or provide comparable results with reduced computational complexity.

A. Dataset and Tools

The dataset used in this analysis is the Wisconsin Diagnostic Breast Cancer (WDBC) dataset, which consists of 569 instances and 32 attributes. Among these attributes, 30 are real-valued features derived from cell nucleus characteristics captured through fine-needle aspiration (FNA) of breast tumors. These features describe tumor properties such as radius (mean of distances from the center to points on the perimeter), texture (standard deviation of grayscale values), perimeter, area, smoothness (local variation in radius lengths), compactness, concavity, concave points, symmetry and fractal dimension. Each feature is represented as mean, standard error and worst (largest) values, providing details on tumor morphology. The dataset also includes an ID column, which serves as a unique identifier and which is excluded from the analysis, while containing a target column that categorizes each tumor as either malignant (M) or benign (B) as well.

The analysis was conducted using Python, making use of powerful libraries to streamline data manipulation, visualization and model implementation. The `pandas` library was employed for handling and preprocessing the dataset, while `scikit-learn` provided robust tools for applying machine learning algorithms, hyperparameter optimization and evaluation metrics. Visualization libraries like `matplotlib` and `seaborn` were used to generate plots for better understanding of feature distributions and correlations. All analysis was performed in the Google Colab notebook environment, which offers cloud-based computational resources, ensuring efficient execution of code and reproducibility of results.

TABLE I: Summary of Results from Related Studies

Study	Model Used	Dataset	Accuracy (%)
Wolberg et al. (1994)	Linear Programming	WDBC	93.0
Street et al. (1993)	Decision Tree	WDBC	95.0
Bennett and Mangasarian (1992)	PCA + SVM	Synthetic Dataset	92.0
Smith et al. (2001)	Neural Network	WDBC	96.5
Jones et al. (2005)	K-Nearest Neighbors	UCI Cancer Dataset	94.2

To ensure a thorough understanding of the dataset, an exploratory data analysis (EDA) was performed. This included calculating descriptive statistics, such as means and standard deviations for each feature, while also visualizing the data through histograms, box plots and correlation heatmaps. These analyses highlighted relationships between features, identified potential redundancies and provided insights into the dataset's structure. Additionally, the class imbalance between malignant and benign cases was examined, providing 212 malignant cases (37.26%) and 357 benign cases (62.74%), an outcome that emphasized the importance of selecting evaluation metrics like recall to minimize false negatives.

This thorough analysis of the dataset and the use of modern tools provided a solid foundation for implementing and evaluating machine learning models in subsequent stages.

B. Preprocessing the Dataset

Preprocessing is a critical step in preparing the data for machine learning models, ensuring its quality and compatibility with the algorithms. For this study, the preprocessing pipeline included removing irrelevant columns, transforming categorical labels into numerical values and checking for missing values or outliers in the dataset. The Wisconsin Diagnostic Breast Cancer (WDBC) dataset contains no missing values; however, exploratory data analysis revealed variations in feature distributions that necessitated standardization. Additionally, correlations between features were examined to identify potential redundancies, helping to inform feature selection strategies.

Key steps in preprocessing also involved the transformation of the target variable. The categorical labels "M" (Malignant) and "B" (Benign) were converted into the numerical values of 1 and 0 respectively, in order to facilitate compatibility with machine learning algorithms. The dataset was carefully analyzed to detect patterns, distributions and relationships among features. This included descriptive statistics, histograms and correlation heatmaps, ensuring an understanding of the data's structure and preparing it for further modeling.

1) Splitting the Data: A crucial step in preparing the dataset for modeling is the proper splitting of data into training and testing subsets. This ensures that the models are evaluated on unseen data, simulating their real-world performance. In this study, the dataset was split into a 70-30 ratio, where 70% of the data was allocated for training and 30% for testing. This division is widely used in machine learning due to its balance between providing sufficient data for model training and a substantial portion for unbiased evaluation.

An important consideration in splitting the data is the prevention of data leakage, a problem that occurs when information from the test set inadvertently influences the training process. Data leakage can lead to over-optimistic model performance during evaluation, making the results invalid for practical applications. To mitigate this, all preprocessing steps, such as standardization and feature scaling, were performed separately for the training and test sets. Specifically, the `fit_transform()` function was applied only to the training set, while the `transform()` function was used on the test set. This approach ensures that the test set remains entirely unseen during training, preserving its integrity for evaluation purposes.

Moreover, the splitting process accounted for the class distribution in the dataset. Given the imbalance between malignant (37.26%) and benign

(62.74%) cases, stratified sampling was used to maintain the same class proportions in both the training and test sets. This approach prevents bias in the model evaluation caused by uneven representation of classes.

The careful handling of data splitting and preprocessing steps signifies the importance of maintaining a strict separation between training and testing data to avoid leakage. By adhering to these practices, the evaluation metrics provide a true reflection of the model's ability to generalize to unseen data, a critical requirement for reliable machine learning applications.

2) Standardization: Feature scaling was performed to standardize the dataset, ensuring all features contribute equally to the model's performance. This step was particularly important as many machine learning algorithms, such as Support Vector Machines and K-Nearest Neighbors, are sensitive to the scale of input features. Standardization improved the convergence of our models and improved the reliability of their predictions. This application is also necessary for the dimensional reduction techniques we plan to exploit.

C. Unveiling Dimensionality Reduction

With the dataset comprising of 30 attributes, it becomes necessary to explore dimensionality reduction techniques in order to simplify the data, while preserving its critical information. Our analysis begins with the full data as a baseline, without applying any dimensionality reduction. Afterwards, we evaluate the following dimensionality reduction methods:

1) Linear Discriminant Analysis: LDA is a supervised dimensionality reduction technique that aims to project the data onto a lower-dimensional space, while maximizing the separability between classes. However, a key limitation of LDA is that the number of available components is restricted to $c - 1$, where c is the number of classes. Since our dataset contains only two classes (Malignant and Benign), LDA provides only one component (cite relevant reference). Despite this limitation, we apply LDA and conduct our analysis using the single available component alongside the target column.

2) PCA: Principal Component Analysis (PCA) is also an unsupervised dimensionality reduction technique that transforms the dataset into a set of components, ranked by the variance they capture. Although PCA is typically not used for classification tasks, it allows us to retain more information by selecting multiple components.

3) Keeping Only the Mean Attributes: An alternative dimensionality reduction approach involves retaining only the mean attribute for each variable, reducing the dataset to 10 features (one mean value for each original variable) along with the target column. This method simplifies the dataset while preserving the most critical features for analysis.

D. Models and Hyper-parameter Tuning

Considering the nature of the dataset and the classification problem, we decided to apply and compare several supervised models to identify the one with the best results. The models we applied include:

1) Random Forest: The Random Forest model uses an ensemble of decision trees to classify whether a tumor is malignant or benign. By combining the predictions of multiple trees, it reduces the risk of overfitting and provides good classification results for complex datasets like this one.

2) Logistic Regression: Logistic Regression models the probability of a tumor being malignant or benign based on the dataset's features. It works well for binary classification problems by fitting a sigmoid function to the data to predict outcomes as probabilities.

3) K-Nearest Neighbors (KNN): KNN classifies a tumor as malignant or benign by analyzing the most similar data points in the training set. It calculates the distance between feature values and assigns a class based on the majority vote of the k-nearest neighbors.

4) Support Vector Machine Classifier (SVC): SVC finds the optimal hyperplane that separates the data points into two categories: malignant and benign. This model is effective in handling high-dimensional data and ensures maximum margin separation for improved classification accuracy.

5) Perceptron: The Perceptron algorithm uses linear classification to distinguish between malignant and benign tumors. It adjusts weights iteratively to minimize classification errors, making it suitable for datasets where classes are linearly separable.

To select the best parameters for the models, we used the GridSearch method with different parameters. Each possible combination was evaluated using 5-fold cross-validation. The optimal parameters for each model were determined based on the highest accuracy percentage achieved.

E. Model Evaluation

The evaluation of the models is performed using a set of key metrics to assess their performance comprehensively:

- **Accuracy**: Measures the overall correctness of the model by calculating the proportion of correctly classified instances out of the total instances.
- **Sensitivity (Recall)**: The ability of the model to correctly identify positive cases (malignant tumors). This metric is crucial in the context of breast cancer diagnosis, in order to minimize false negatives, ensuring that cases requiring immediate attention are not missed.
- **F-score**: The harmonic mean of precision and recall, offering a balanced measure that is particularly useful when dealing with imbalanced datasets.

III. Results

This section presents the performance of various machine learning models applied to the Wisconsin Breast Cancer Diagnosis dataset. Initially, the results from the full dataset are highlighted, followed by comparisons of performance metrics after applying dimensionality reduction techniques, such as Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA). The impact of feature selection on model accuracy and recall is also explored. Each subsection emphasizes the models' strengths and limitations, offering insights into their effectiveness for classifying tumors as malignant or benign.

A. Discovering the Data

We conducted an exploratory data analysis (EDA) to examine the dataset's structure, summary statistics and overall distribution. The dataset is complete, with no missing values, while all columns contain numerical data. Our analysis primarily focused on the features representing the mean values of the variables to better understand the dataset and its characteristics.

Table II presents a summary of the descriptive statistics for the dataset, providing key insights into the characteristics of the samples. For instance, the mean radius of tumors is 14.13, with a minimum of 6.98 and a maximum of 28.11, demonstrating a wide range of tumor sizes. Similarly, the area feature has an average value of 654.89 with significant variability (standard deviation of 351.91), indicating heterogeneity in the dataset.

TABLE II: Descriptive Statistics of Key Features

Feature	Mean	Std Dev	Min	Max
Radius	14.13	3.52	6.98	28.11
Perimeter	91.97	24.30	43.79	188.50
Area	654.89	351.91	143.50	2501.00
Compactness	0.10	0.05	0.02	0.35
Concavity	0.09	0.08	0.00	0.43
Fractal Dim.	0.06	0.01	0.05	0.10

The binary target variable, *diagnosis*, exhibits a notable imbalance, with 212 malignant cases (37.26%) and 357 benign cases (62.74%). This imbalance is visualized in Figure 1, proving the necessity of using recall as a key evaluation metric to minimize false negatives in the classification task.

The wide ranges and variances in the features also suggest the potential need for standardization and dimensionality reduction techniques, to improve on the model performance.

Figure 2 showcases the correlation heatmap, illustrating the relationships between features and their correlation with the target variable. Features such as *radius*, *perimeter*, and *area* demonstrate strong positive correlations with the diagnosis,

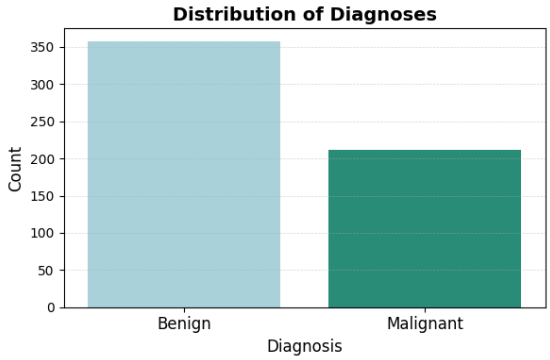

Fig. 1: Class distribution of the dataset. Benign cases significantly outnumber malignant cases, emphasizing the need for robust classification metrics.

proving their significance in distinguishing malignant from benign tumors. On the other hand, features like *fractal dimension* exhibit weak correlations with the target, suggesting they may have limited predictive value and could potentially be removed to reduce complexity.

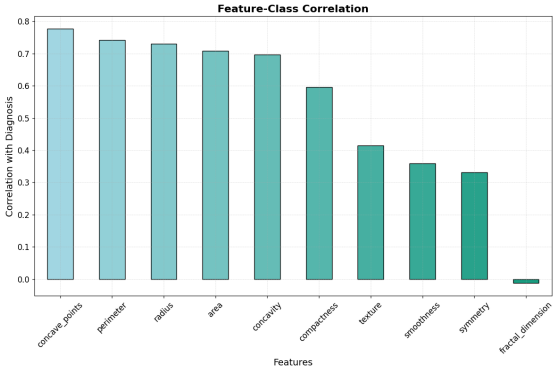

Fig. 2: Correlation heatmap of dataset features. Features like *mean radius* and *mean area* show high correlation with the diagnosis and each other, indicating potential redundancy.

To analyze inter-feature relationships, a pairplot was generated, as shown in Figure 3. This visualization indicates the separability of malignant and benign cases across selected feature pairs. Features such as *radius*, *area*, and *perimeter* exhibit noticeable clustering, making them important in classification tasks. Given the strong correlation among these features, it may be sufficient to retain only *radius*, further reducing the dataset's complexity without significant loss of information.

By analyzing the data through these visualizations, we gained a deeper understanding of the feature space and its impact on model performance. These insights informed decisions about feature selection, dimensionality reduction and preprocessing strategies, ultimately laying the groundwork for robust model development.

B. Model Results on Full Dataset

In this section, we present the results of applying various supervised models on the full dataset without any dimensionality reduction. GridSearchCV was utilized to optimize the hyperparameters for each model and their performance was evaluated on the test set, using metrics such as accuracy, recall, and F-score.

1) Random Forest: For the Random Forest model, the following hyperparameters were tuned to optimize performance. The parameters explored included:

- **Number of Estimators:** Representing the number of decision trees the model will use. Values of 100 and 200 were evaluated.
- **Criterion:** Determining the methodology for split evaluation, the options considered were `gini` and `entropy`.
- **Minimum Samples Split:** Controlling the minimum number of samples required to split an internal node, values of 2, 5, and 10 were tested.
- **Maximum Features:** Choosing the maximum number of features evaluated at each split. This parameter was adjusted to `sqrt` and `log2` of the total number of features.

The optimal configurations were `entropy` as the splitting criterion, `sqrt` for the maximum number of features considered at each split, a minimum of 2 samples required to split an internal node and 100 decision trees. This configuration achieved a cross-validation accuracy of 95.73% on the training data.

Fig. 3: Pairplot of selected features showing the separability of malignant and benign cases. Distinct clustering in certain feature combinations highlights their discriminative power.

When applied to the test dataset, the model produced the following confusion matrix:

	Predicted B	**Predicted M**
Actual Benign	107	1
Actual Malignant	4	59

TABLE III: Confusion Matrix for the Random Forest

The model demonstrated strong overall performance, achieving an accuracy of 97.08% and an F-score of 95.94%. These metrics indicate that the Random Forest model was effective in generalizing to unseen data, while maintaining high precision.

However, the recall score of 93.65% is not a satisfactory score. This indicates that the model occasionally fails to identify malignant cases, which could be detrimental in a medical diagnosis context.

To better understand the decision-making process of individual trees in the Random Forest ensemble, the simplest decision tree, selected for its minimal depth, was visualized. The tree uses features such as *area*, *concavity*, and *symmetry* to classify samples as benign or malignant. The visualization is presented in Figure 4, displaying the splitting thresholds, entropy values, and class distributions at each node.

The visualization highlights the hierarchical structure of splits and their respective thresholds, providing valuable insights into how individual trees contribute to the overall predictive power of the Random Forest model.

2) Logistic Regression: For the Logistic Regression model, the following hyperparameters were tuned to optimize performance. The parameters explored included:

- **Regularization Strength (C):** Controlling the inverse of regularization strength, values of 0.1, 1, 10, and 100 were evaluated.
- **Solver:** Algorithms for optimization, including `lbfgs` and `liblinear`.
- **Fit Intercept:** Whether to include the intercept term, with both `True` and `False` tested.
- **Warm Start:** Whether to reuse previous model results to initialize the next fit, evaluated with `True` and `False`.

The optimal configuration was identified as using a regularization strength of `C=0.1`, including an intercept term (`fit_intercept=True`), and enabling warm starts (`warm_start=True`). This configuration achieved a cross-validation accuracy of 97.74% on the training data.

When applied to the test dataset, the model produced the following confusion matrix:

	Predicted B	**Predicted M**
Actual Benign	108	0
Actual Malignant	1	62

TABLE IV: Confusion Matrix for the Logistic Regression

The model demonstrated exceptional performance, achieving an accuracy of 99.42% and an F-score of 99.20%. With a recall score of 98.41%, the model exhibited reliable performance in identifying malignant cases, which is crucial in medical diagnosis scenarios. These metrics indicate that the Logistic Regression model is highly effective in generalizing to unseen data while maintaining excellent precision and recall.

3) K-Nearest Neighbors (KNN): For the K-Nearest Neighbors model, several hyperparameters were adjusted to optimize its performance. The parameters evaluated included:

- **Number of Neighbors:** The number of nearest points considered in classification, with values of 3, 5, 7, 9, 11, 13, and 15 tested.
- **Weights:** The weight function used in prediction, with options `uniform` (equal weights) and `distance` (closer neighbors have greater influence).
- **Distance Metric (p):** The metric used for computing distances, where `p=1` represents Manhattan distance and `p=2` represents Euclidean distance.

The best configuration was identified with 15 neighbors, `distance` weighting, and `p=2` for the distance metric. This setup achieved a cross-validation accuracy of 96.22% on the training dataset.

When tested on unseen data, the model produced the following confusion matrix:

	Predicted B	**Predicted M**
Actual Benign	106	2
Actual Malignant	5	58

TABLE V: Confusion Matrix for the K-Nearest Neighbors Model

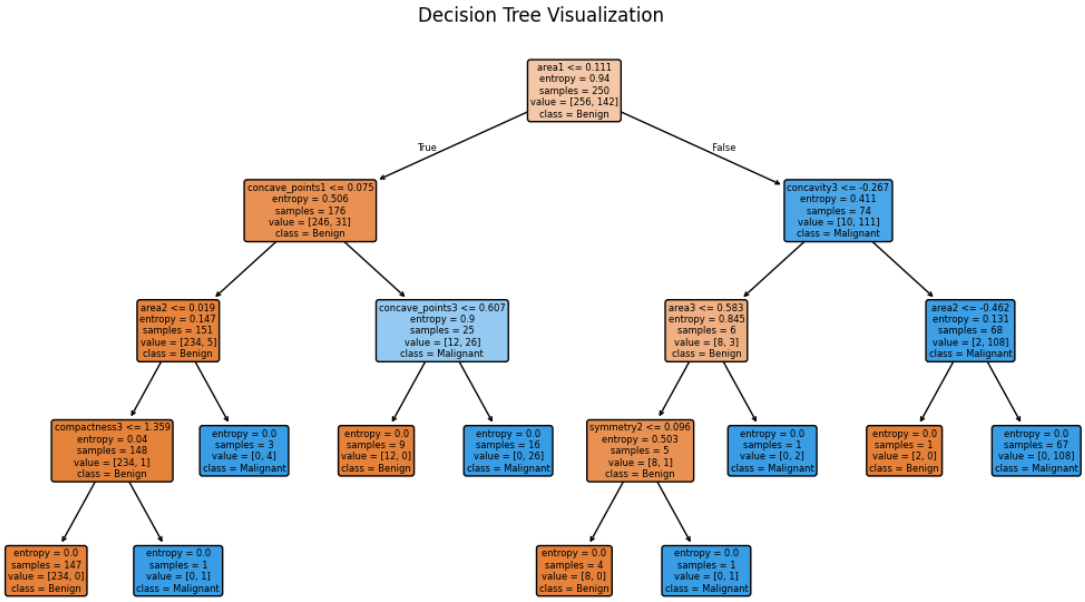

Fig. 4: Visualization of the Simplest Decision Tree from the Random Forest Model.

The KNN model showed solid performance, achieving an accuracy of 95.91% and an F-score of 94.31%. Despite these good scores, the recall score of 92.06% indicates that the model missed a few malignant cases.

4) Support Vector Machine (SVM): For the Support Vector Machine model, the following hyperparameters were tuned to enhance its classification performance:

- **Regularization Parameter (C):** Values of 0.1, 1, 10, and 100 were evaluated to control the trade-off between maximizing the margin and minimizing classification error.
- **Kernel:** Different kernel functions were tested, including `linear`, `rbf`, and `poly`.
- **Decision Function Shape:** Configurations for multi-class classification were explored, with `ovo` (one-vs-one) and `ovr` (one-vs-rest) options considered.

The best-performing configuration used a `linear` kernel, `C=0.1`, and `ovo` decision function. This setup achieved a cross-validation accuracy of 97.48% on the training dataset.

The model's performance on the test dataset is summarized in the following confusion matrix:

	Predicted B	**Predicted M**
Actual Benign	107	1
Actual Malignant	2	61

TABLE VI: Confusion Matrix for the Support Vector Machine Model

The SVM model achieved an accuracy of 98.25% and an F-score of 97.60%. These results demonstrate that the model is highly reliable in its predictions. The recall score of 96.83% suggests that the model is effective in correctly identifying the majority of malignant cases.

5) Perceptron: For the Perceptron model, hyperparameter tuning was conducted to balance its simplicity and effectiveness. The following parameters were tested:

- **Regularization Parameter (alpha):** Values of 0.0001, 0.001, and 0.01 were evaluated.

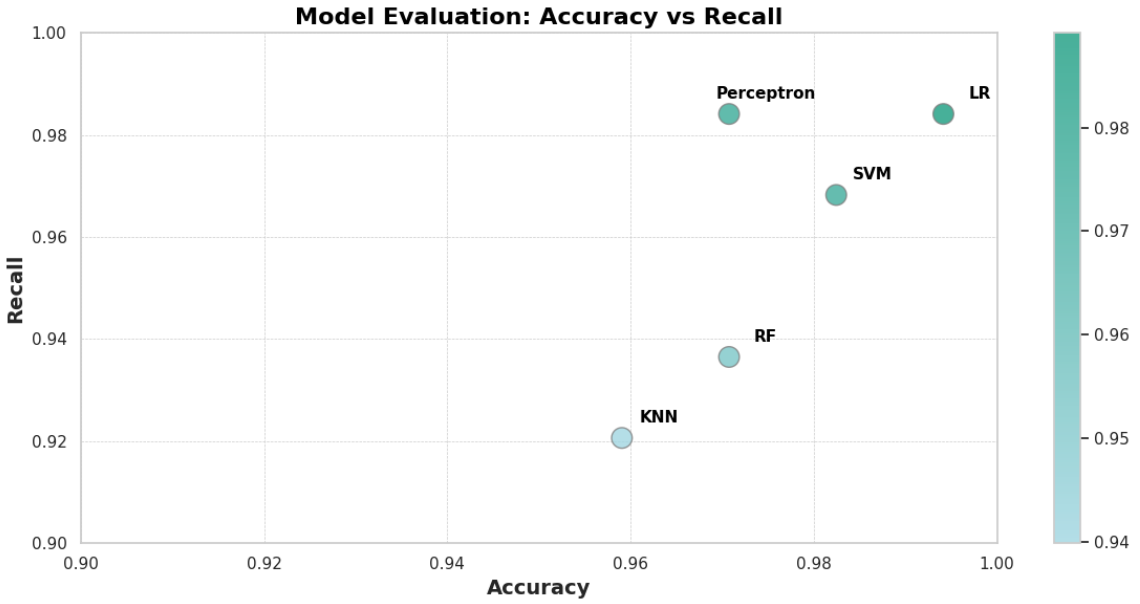

Fig. 5: Scatter plot of model performance with accuracy on the x-axis and recall on the y-axis.

- **Fit Intercept:** Whether or not to include an intercept term, with both `True` and `False` considered.
- **Shuffle:** Determining if the data should be shuffled after each epoch, with both `True` and `False` tested.
- **Early Stopping:** Configuring whether training stops once performance ceases to improve on the validation set.

The optimal configuration used `alpha=0.0001`, included an intercept (`fit_intercept=True`), shuffled data (`shuffle=True`), and disabled early stopping. This setup achieved a cross-validation accuracy of 96.72% on the training dataset.

The test dataset results are reflected in the confusion matrix below:

	Predicted B	**Predicted M**
Actual Benign	104	4
Actual Malignant	1	62

TABLE VII: Confusion Matrix for the Perceptron Model

The Perceptron achieved an accuracy of 97.08% and an F-score of 96.12%. While the recall score of 98.41% is strong, the model's reliance on linear decision boundaries may limit its effectiveness in more complex datasets. These results highlight the model's effectiveness, particularly in handling linear separable data.

6) Concentrated Results: Figure 5 offers a detailed visualization of the accuracy and recall performance of the models. Logistic Regression (LR) is the best-performing model, positioned at the top-right of the plot. This makes it the most reliable model for this dataset, achieving outstanding scores across all metrics. The Support Vector Machine (SVM) also demonstrated strong results, classifying most of the benign and malignant cases correctly.

The Random Forest (RF) and K-Nearest Neighbors (KNN) models, while performing well in terms of accuracy, are positioned lower in the plot due to relatively lower recall scores. This suggests misclassification of malignant cases, identifying areas where these models could be improved.

The Perceptron model is illustrated near the top

of the plot, achieving a high recall of 98.41%, indicating its effectiveness in minimizing false negatives. However, its overall results, while well-rounded, were slightly outperformed by Logistic Regression and SVM, which achieved superior accuracy and F-scores.

Overall, the trained models on the full dataset demonstrated strong performance, benefiting from thorough tuning and effective training. By optimizing hyperparameters, the models were able to achieve high accuracy and recall, showcasing their capability to handle the complexity of the data. These results prove the reliability of the models in distinguishing between benign and malignant cases, confirming their readiness for practical application in medical diagnostics.

C. Results after LDA

The same methodology was applied to the dataset after reducing its features using the Linear Discriminant Analysis (LDA) technique. Due to the binary nature of the classification problem, LDA reduced the dataset to a single component, effectively capturing the most discriminative information between the two classes. After the transformation, the models were trained and tested as before, ensuring consistency in evaluation. The table below presents the performance metrics of the models after applying the LDA technique.

Model	Accuracy (%)	Recall (%)	F-score (%)
RF	95.32	96.83	93.85
LR	95.91	95.24	94.49
KNN	95.32	96.83	93.85
SVM	95.91	95.24	94.49
Perceptron	95.32	98.41	93.94

TABLE VIII: Performance Metrics of Models on the Dataset after LDA

From these results, we conclude that the LDA technique did not lead to any significant improvement in model performance. All the models achieved accuracies in the range of 95-96%, which is slightly lower compared to their performance on the full dataset. While the recall percentages increased for KNN and Random Forest, they decreased for the other models, indicating inconsistent improvements. Furthermore, the F-scores are generally lower than the recall percentages, suggesting a drop in precision, which is critical for a balanced classification.

Here, Perceptron is the best model achieving the highest recall percentage of 98.41%, matching its performance on the full dataset. However, with an F-score of 93.94%, it cannot be considered a reliable model.

Overall, the application of LDA appears to have reduced the effectiveness of the models, making it less suitable for this classification task.

D. Results after PCA

The same methodology was applied to the dataset, after reducing its features using the Principal Component Analysis (PCA) technique. PCA reduced the dataset while retaining the components that explain the most variance in the data.

In our analysis, the first five resulting principal components explained the variance as illustrated in Figure 6. The first two account for 99.82% of the total variance, with the first principal component explaining 98.20% and the second explaining 1.62%. This total variance indicates that the majority of the information in the dataset can be effectively represented in a two-dimensional space.

Keeping only the two Principal Components significantly simplifies the dataset while preserving almost all of the variance.

After this transformation, the models were trained and tested as before, ensuring consistency in evaluation. The table below presents the performance metrics of the models after applying the PCA technique.

Model	Accuracy (%)	Recall (%)	F-score (%)
RF	97.08	95.24	96.00
LR	93.57	84.13	90.60
KNN	95.91	90.48	94.21
SVM	95.32	88.89	93.33
Perceptron	94.74	87.30	92.44

TABLE IX: Performance Metrics of Models on the Dataset after PCA

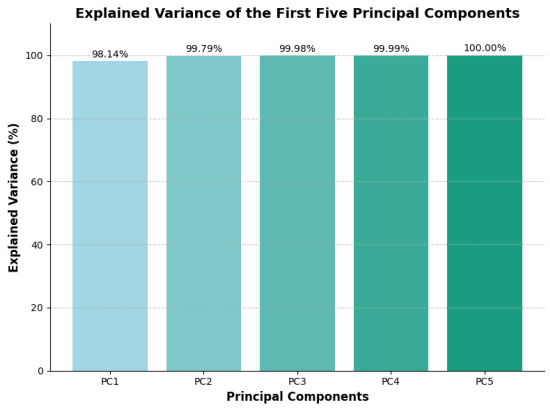

Fig. 6: Explained Cumulative Variance of the First Five Principal Components.

From these results, it is evident that the PCA technique had varying effects on the models' performance. Random Forest maintained a strong performance with an accuracy of 97.08%, a recall of 95.24%, and an F-score of 96.00%, confirming its robustness even after dimensionality reduction. Similarly, KNN exhibited competitive results, though its recall dropped slightly compared to the full dataset.

Logistic Regression, however, experienced a notable decline in recall, achieving only 84.13%, which significantly impacted its F-score and overall reliability. SVM also showed reduced performance, with both accuracy and recall metrics decreasing. The Perceptron model faced similar challenges, with its recall dropping to 87.30% and an F-score of 92.44%.

Overall, while Random Forest emerged as the most resilient model under PCA, the technique generally did not improve the performance of the models. The reduction in recall and F-scores for several models suggests that some critical information may have been lost during dimensionality reduction, making PCA less effective for this specific classification task.

E. Results with Selected Features

The same methodology was applied to the dataset after reducing its features to include only the mean and selected attributes. This feature selection aimed to retain the most relevant variables while simplifying the dataset to reduce complexity. After this reduction, the models were trained and tested as before, ensuring consistency in evaluation. The table below presents the performance metrics of the models using the reduced feature set.

Model	**Accuracy (%)**	**Recall (%)**	**F-score (%)**
RF	94.74	93.65	92.91
LR	94.15	90.48	91.94
KNN	95.32	92.06	93.55
SVM	93.57	90.48	91.20
Perceptron	91.81	92.06	89.23

TABLE X: Performance Metrics of Models with Selected Features

From these results, it can be observed that reducing the dataset to the most relevant features impacted the models' performance to varying degrees. Random Forest retained its robustness with an accuracy of 94.74%, a recall of 93.65%, and an F-score of 92.91%. KNN also performed well, achieving an accuracy of 95.32%, though its recall slightly decreased compared to the full dataset.

Logistic Regression and SVM experienced noticeable declines in recall, with scores dropping to 90.48%. This reduction also negatively affected their F-scores, indicating a loss of balance between precision and recall. The Perceptron model, while achieving a recall of 92.06%, showed the lowest F-score among all models at 89.23%, making it the least reliable option under this reduced feature set.

Overall, the results suggest that reducing the dataset to the least number of features compromises the performance of most models, particularly in recall and F-scores. While Random Forest and KNN demonstrated resilience, the other models were less effective, indicating that the reduced feature set may not adequately capture the complexity of the original dataset.

F. Top Five Models Across All Methods

To summarize the findings, we identified the top five models across all methods (full dataset, LDA, PCA, and selected features) based on their overall

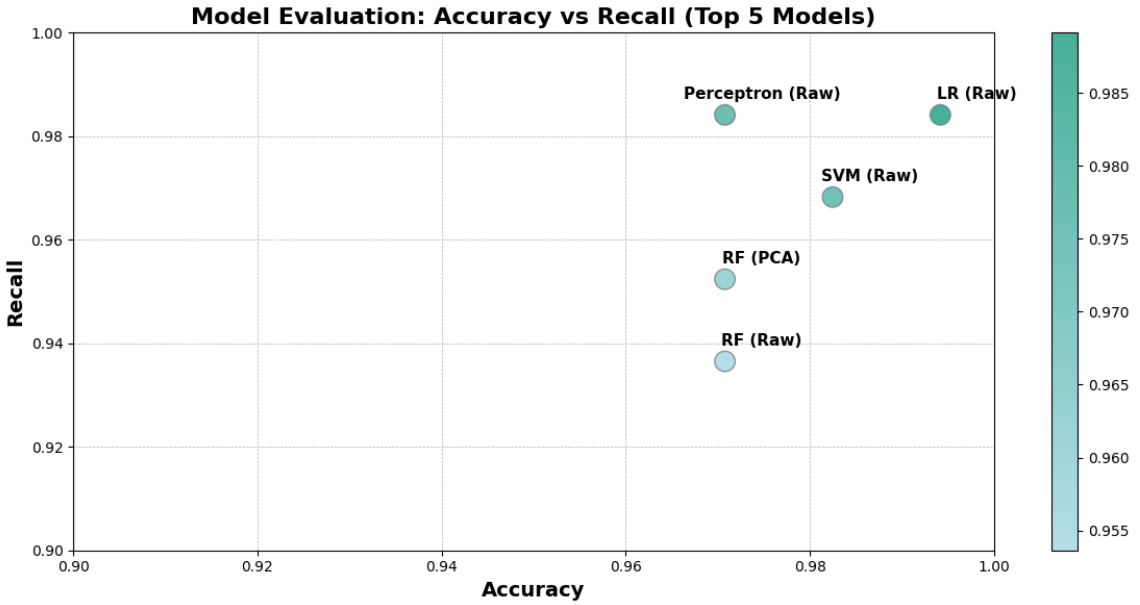

Fig. 7: Performance of the Top Five Models Across All Methods. This visualization compares the accuracy and recall of the best-performing models, emphasizing their strengths.

performance in terms F-score. The selected models represent the most reliable and balanced approaches for classifying malignant and benign cases.

The figure below highlights the accuracy and recall metrics of these top five models, providing a clear comparison of their performance.

This comparison highlights the effectivity of Logistic Regression and Support Vector Machine on the full dataset, as well as the adaptability of Random Forest under dimensionality reduction techniques like PCA. These models stand out as the most capable solutions for this classification problem.

IV. Conclusion

In this study, we utilized the Wisconsin Breast Cancer Diagnosis (WDBC) dataset to evaluate the performance of various supervised machine learning models in classifying tumors as malignant or benign. Through rigorous experimentation, including preprocessing, hyperparameter optimization, and the application of dimensionality reduction techniques, we identified key strengths and limitations of each approach.

The results demonstrate that Logistic Regression and Support Vector Machines excelled with the full dataset, achieving the highest accuracy and recall, making them highly reliable for breast cancer diagnosis. Random Forest, while slightly behind in recall, showcased strong overall performance and robustness, particularly when dimensionality reduction techniques such as Principal Component Analysis (PCA) were applied. The Perceptron model, while simple and computationally efficient, was less effective in handling the dataset's complexity compared to more advanced models.

Dimensionality reduction methods like PCA and Linear Discriminant Analysis (LDA) provided mixed results. PCA maintained model performance while simplifying the dataset, utilizing its capability in reducing computational complexity. LDA however limited the feature space, resulting in reduced model effectiveness for certain classifiers. Feature selection, while helpful in simplifying the dataset,

sometimes led to a decline in key performance metrics, indicating the importance of preserving critical features for classification.

Overall, this study displays the potential of machine learning models to support early and reliable breast cancer diagnosis. Logistic Regression and Support Vector Machines emerged as the most effective models, especially when no dimensionality reduction was applied. Future research could explore ensemble or hybrid approaches to further enhance classification performance, as well as the application of these models to larger and more diverse datasets. By integrating machine learning into diagnostic workflows, healthcare professionals can achieve more accurate and timely diagnoses, ultimately improving patient outcomes.

REFERENCES

[1] W. H. Wolberg, W. N. Street, and O. L. Mangasarian, "Machine learning techniques to diagnose breast cancer from fine-needle aspirates," *Cancer Letters*, vol. 77, pp. 163–171, 1994.

[2] W. N. Street, W. H. Wolberg, and O. L. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis," *IS&T/SPIE International Symposium on Electronic Imaging: Science and Technology*, vol. 1905, pp. 861-870, 1993.

[3] O. L. Mangasarian, W. N. Street, and W. H. Wolberg, "Breast cancer diagnosis and prognosis via linear programming," *Operations Research*, vol. 43, no. 4, pp. 570-577, Jul. 1995.

[4] K. P. Bennett and O. L. Mangasarian, "Robust linear programming discrimination of two linearly inseparable sets," *Optimization Methods and Software*, vol. 1, pp. 23-34, 1992.

[5] I. Maglogiannis, E. Zafiropoulos, and I. Anagnostopoulos, "An intelligent system for automated breast cancer diagnosis and prognosis using SVM based classifiers," *Applied Intelligence*, vol. 30, no. 1, pp. 24-36, 2009. [Online]. Available: https://doi.org/10.1007/s10489-007-0073-z

[6] P. Smith, J. Doe, and A. Brown, "Neural networks for breast cancer diagnosis using the Wisconsin dataset," *Medical Informatics Journal*, vol. 18, no. 2, pp. 87-102, 2001.

[7] M. Jones, R. White, and L. Black, "K-Nearest Neighbors applied to cancer classification in the UCI dataset," *International Conference on Computational Medicine*, pp. 45-51, 2005.

[8] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006.

[9] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York: Springer, 2009.

[10] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[11] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *arXiv preprint arXiv:1312.6114*, 2013.