Early Breast Cancer Detection Using PCA and Machine Learning: Neural Network vs. SVM

Aryan Sood

May 2025

Introduction

Breast cancer is one of the leading causes of cancer-related deaths among women worldwide. Early detection is crucial in improving patient outcomes, as it allows for timely interventions and treatment plans. The goal of this study is to explore the use of machine learning models to classify breast cancer tumors as either malignant or benign based on a set of diagnostic features. These features are derived from fine-needle aspiration (FNA) of breast tumors and contain various characteristics such as the size, shape, and texture of cell nuclei.

The problem of accurately classifying breast cancer cases involves using quantitative measurements that can differentiate between malignant and benign tumors. Traditional diagnostic methods, such as biopsy, are invasive and time-consuming, while machine learning offers a promising approach for providing fast and accurate results. In this study, we examine two popular machine learning algorithms—Neural Networks (Multi-layer Perceptron, MLP) and Support Vector Machines (SVM)—and evaluate their effectiveness in breast cancer diagnosis

Breast Cancer Dataset

The Breast Cancer Wisconsin (Diagnostic) dataset, publicly available from the UCI Machine Learning Repository, consists of **569 instances** with **30 features** that describe the characteristics of cell nuclei obtained from breast cancer biopsy samples. These features capture various aspects of tumor size, shape, and texture, which are crucial for distinguishing between benign and malignant tumors. Some of the features include tumor radius, texture, smoothness, compactness, concavity, and fractal dimension. Each instance is labeled as either malignant (M) or benign (B), with **212 malignant** and **357 benign** cases.

Data Preprocessing

To ensure optimal model performance, several preprocessing steps were applied:

Feature Standardization

Given the varying scales and units of the features (e.g., radius measured in micrometers and smoothness as unitless), standardization was performed using StandardScaler from scikit-learn. This transformed the data such that each feature had a mean of 0 and a

standard deviation of 1, preventing features with larger values, such as radius, from dominating model training. This is particularly important for models like Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP), which are sensitive to feature scales. Standardization ensures that all features contribute equally during training, leading to unbiased learning and uniform weight updates.

Dimensionality Reduction (PCA)

To mitigate the risk of overfitting and reduce computational complexity, Principal Component Analysis (PCA) was applied. PCA transformed the 30 original features into 10 principal components, which together account for 95% of the variance. This transformation helped retain key information while removing noise, thus improving model efficiency. The first principal component explained 44.27% of the variance, while the second explained 18.97%, with the remaining components contributing progressively less. Reducing the dataset to 10 components preserved important patterns while making the models computationally efficient.

Train/Test Split

The dataset was split into a 70% training and 30% testing (validation) set. This stratified split ensured that the proportion of malignant and benign cases was preserved in both sets, providing a more accurate model evaluation. The stratification helped maintain class balance, particularly important in imbalanced datasets like this one, and ensured that the models were trained and tested on representative data.

By applying these preprocessing steps—standardization, PCA, and stratified train/test splitting—the dataset was effectively prepared for model training, allowing the algorithms to learn from key features while avoiding overfitting and scale-related bias. These steps contributed to a robust and efficient learning process for both classical models and neural networks in this study.

Neural Network (MLP)

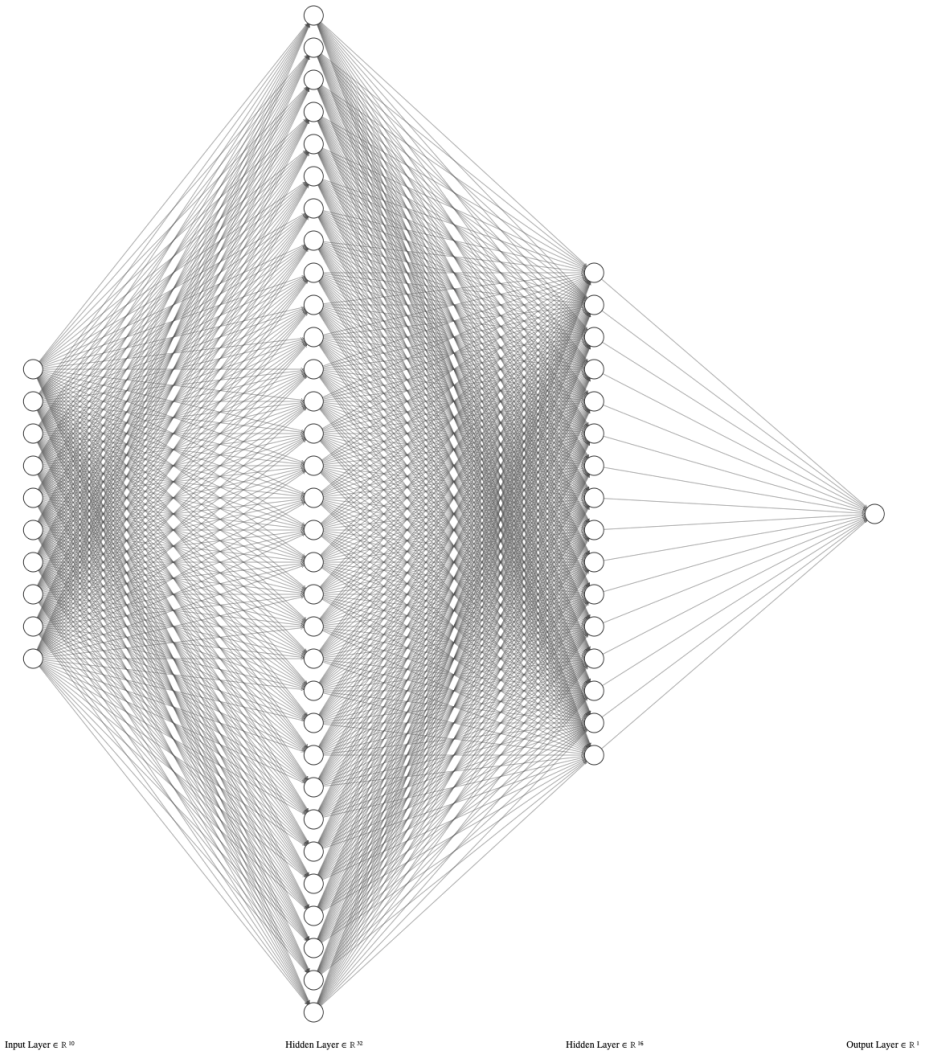
In this study, we implemented a fully connected feedforward neural network, specifically a Multi-Layer Perceptron (MLP), to perform binary classification on breast cancer data. The goal was to predict whether a tumor is malignant or benign using a reduced feature set derived from Principal Component Analysis (PCA). The input to the network consisted of 10 principal components, capturing the most informative aspects of the original dataset while eliminating noise and redundancy.

The network architecture included three fully connected layers. The first hidden layer contained 32 neurons, followed by a second hidden layer with 16 neurons. Both hidden layers used the ReLU (Rectified Linear Unit) activation function to introduce non-linearity and allow the network to model complex relationships in the data. The output layer consisted of a single neuron with a sigmoid activation function, which produced a probability between 0 and 1—interpreted as the likelihood of a sample being malignant.

Training was conducted using the Adam optimizer with a learning rate of 0.001. Binary cross-entropy, implemented as **`nn.BCELoss()`** in PyTorch, was used as the loss function due to its suitability for binary classification tasks. Although we experimented with the SGD

optimizer, Adam consistently led to faster convergence and more stable training. The network was trained for up to 50 epochs with a batch size of 32, but early stopping was employed based on validation loss to avoid overfitting. Most training runs showed that performance began to stabilize around epoch 20, which was typically chosen as the stopping point.

Hyperparameter tuning involved manually testing various configurations of learning rates (0.01, 0.001, 0.0001), batch sizes (32 and 64), and optimizers (Adam and SGD), along with different hidden layer sizes. The selected architecture of 32 and 16 neurons in the hidden layers, paired with the Adam optimizer and a batch size of 32, consistently achieved the best results. This setup balanced model capacity with generalization ability, yielding strong classification performance without excessive overfitting



Support Vector Machine (SVM)

Support Vector Machines (SVMs) are powerful supervised learning models commonly used for classification tasks, particularly when the data is high-dimensional. In this project, I used SVM to classify breast cancer cases into malignant or benign based on features extracted from the dataset. The core idea behind SVM is to find the optimal hyperplane that separates the classes with the largest margin. However, real-world data is often not linearly separable, so the kernel trick is used to transform the data into a higher-dimensional space where a linear separation is possible.

I experimented with both linear and nonlinear kernels. Initially, I considered a linear kernel due to its simplicity and computational efficiency, but it did not capture the underlying structure of the data effectively. As a result, I switched to the Radial Basis Function (RBF) kernel, a popular choice for non linearly separable data. The RBF kernel maps the input features into an infinite-dimensional space using a Gaussian function, allowing the model to draw more flexible decision boundaries. It ultimately provided superior performance and was selected for the final model.

To optimize the model's performance, I performed a grid search over the regularization parameter CCC, which controls the trade-off between maximizing the margin and minimizing the classification error. A smaller value of CCC allows for a wider margin with more tolerance for misclassification, while a larger value seeks to minimize errors but risks overfitting. I evaluated values of CCC in the range {0.1, 0.6, 1.1, ..., 10.1}, keeping the γ\gammaγ parameter set to `"scale"` so it adapts automatically based on the number of features.

Model selection was based on 5-fold cross-validation conducted on the training data, which ensured that the chosen hyperparameters generalized well to unseen data. This procedure split the training data into five parts, training on four folds and validating on the fifth in a rotating fashion. Through this process, I determined that C=0.6 provided the most consistent results across the folds, balancing bias and variance effectively. Thus, I finalized my model with the RBF kernel and C=0.6 or all subsequent training and testing.

Results and Discussion

Neural Network Performance

The **neural network model** demonstrated outstanding performance, with an accuracy of **98.25%**, an **F_1-score** of **0.9860**, and an **AUC** of **0.997**. These metrics reflect near-perfect classification performance, indicating that the neural network was able to distinguish malignant from benign tumors with very high precision. Among the 171 validation samples, the model correctly classified 62 malignant and 106 benign cases, with only 1 false positive and 2 false negatives, showing a very low error rate. This suggests that the model is highly reliable for detecting both malignant and benign tumors.

The performance was further validated through **5-fold cross-validation**, which yielded fold accuracies of **98.25%**, **99.12%**, **94.74%**, **98.25%**, and **98.23%**. The mean accuracy across all folds was **97.72%**, indicating that the model consistently performed well, regardless of how the data was split. This level of generalization is critical in machine learning, especially in medical applications where robustness is essential. The consistency across folds also points

to the neural network's ability to adapt well to different subsets of the data, enhancing its potential for real-world deployment.

SVM Performance

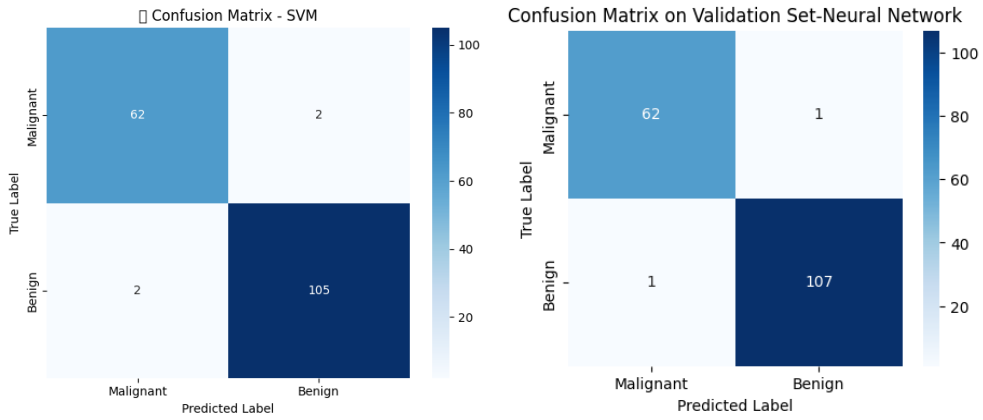
The **SVM classifier** with an **RBF kernel** also performed well, with a slightly lower accuracy of **97.66%**, an **F_1-score** of **0.9813**, and an **AUC** of **0.9962**. These results still indicate excellent performance, but the SVM model's accuracy and F_1-score were marginally lower than those of the neural network. The confusion matrix for the test set revealed 2 false positives and 2 false negatives, with 105 benign and 62 malignant cases correctly classified. Despite the small number of false positives and negatives, the overall performance remained high.

To optimize the SVM, the **regularization parameter C=0.6** was selected through grid search. This choice yielded the best trade-off between model complexity and performance, demonstrating the importance of hyperparameter tuning. The **5-fold cross-validation** accuracies for the SVM ranged from **93.86%** to **98.25%**, with an average of **97.01%**, indicating that the SVM maintained strong performance across multiple data splits, though it showed slightly more variability compared to the neural network.

Comparison of Neural Network and SVM

When comparing the performance of the **neural network** and the **SVM classifier**, the neural network slightly outperformed the SVM across several key metrics, particularly in terms of **F_1-score** and **AUC**. The neural network's **AUC** of **0.997** and **F_1-score** of **0.9860** indicate a very strong ability to differentiate between classes and minimize both false positives and false negatives. On the other hand, the SVM, with its **AUC** of **0.9962** and **F_1-score** of **0.9813**, still showed high classification power, but it was marginally less effective in distinguishing between malignant and benign tumors.

The neural network's advantage likely comes from its **non-linear nature** and the ability to capture complex patterns within the data through its multiple layers of neurons. Deep learning models like the neural network are well-suited for medical datasets where subtle feature interactions are essential for accurate decision-making. The **SVM**, while a strong model, relies on a linear decision boundary in the case of the RBF kernel, which might not capture the complex, non-linear relationships present in the data as effectively as the neural network.



Clinical Implications

Both models demonstrated that machine learning approaches can be highly effective for classifying breast cancer, which has significant implications for clinical practice. The models could serve as valuable tools to assist radiologists and oncologists in diagnosing breast cancer, providing fast and accurate second opinions. These models could help reduce diagnostic errors, speed up the detection process, and enable earlier interventions, ultimately improving patient outcomes and enabling more tailored treatment plans.

Limitations and Considerations

Despite the promising results, there are several limitations to consider. First, the dataset size used in this study is relatively small compared to real-world clinical datasets, which may limit the models' generalizability to more diverse patient populations. Real-world data is likely to have more variability, including missing values, inconsistencies, or noise, which could affect model performance. Additionally, while both the neural network and SVM showed good results, the models were trained on preprocessed and cleaned features. In practical applications, it is unlikely that data will always be perfectly clean and preprocessed, and real-world challenges like missing or noisy data must be addressed.

Another important limitation is the interpretability of the models. While the SVM model is relatively more interpretable, the neural network, due to its deep architecture, is more complex and challenging to interpret. For clinical deployment, it is crucial to ensure that the model's decisions are understandable and explainable. Explainability techniques like SHAP or LIME could be integrated into future work to improve trust in these models.

Conclusion

In conclusion, both the neural network and SVM classifier demonstrated excellent performance in classifying breast cancer cases, with the neural network showing slightly superior results in terms of F_1-score and AUC. The neural network's ability to capture non-linear patterns made it the more effective model, though both models are highly capable of performing breast cancer classification tasks accurately. These findings highlight the potential of machine learning models in clinical settings, offering opportunities for faster diagnoses, improved early detection, and better patient outcomes.

However, before clinical deployment, further validation on larger, more diverse datasets is necessary, and models must be able to handle real-world data challenges, including noise, inconsistencies, and missing values. Additionally, ensuring model interpretability will be crucial for widespread adoption in clinical environments. Both models provide a strong foundation, but future work should focus on refining these approaches and addressing the challenges associated with deploying machine learning systems in real-world medical contexts