On Breast Cancer Detection: A Machine Learning Approach using Scikit-learn on the Wisconsin Diagnostic Dataset

Suchithra P S and Shubha N B

MCA 2nd semerster

Chanakya University Devanahalli,Bengaluru

Abstract*—This study presents a comparative analysis of several supervised machine learning algorithms on the Wisconsin Diagnostic Breast Cancer (WDBC) dataset. The algorithms examined include Logistic Regression, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Tree, Random Forest, and Multilayer Perceptron (MLP). The dataset, comprising 569 samples and 30 features, represents measurements derived from digitized images of fine needle aspirate (FNA) of breast masses. Using a 70/30 train-test split and SMOTE for class balance, the models were evaluated using accuracy, precision, recall, and F1-score. MLP emerged as the top performer with an accuracy of 99.04%, highlighting the significance of deep learning architectures in medical diagnostics. This research reaffirms the capability of machine learning to contribute meaningfully to early detection and diagnosis of breast cancer

KEYWORDS- Breast cancer; machine learning; classification; logistic regression; decision trees; neural networks; SVM; KNN; Wisconsin Diagnostic Breast Cancer dataset

1.INTRODUCTION

Breast cancer is one of the most common cancer along with lung and bronchus cancer, prostate cancer, colon cancer, and pancreatic cancer among others[2]. Representing 15% of all new cancer cases in the United States alone[1], it is a topic of research with great value. The utilization of data science and machine learning approaches in medical fields proves to be prolific as such approaches may be considered of great assistance in the decision making process of medical practitioners. With an unfortunate increasing trend of breast cancer cases[1], comes also a big deal of data which is of sig nificant use in furthering clinical and medical research, and much more to the application of data science and machine learning in the aforementioned domain. Prior studies have seen the importance of the same research topic[17, 21], where they proposed the use of machine learning (ML) algorithms for the classification of breast cancer using the Wisconsin Diagnostic Breast Cancer (WDBC) dataset[20], and even tually had significant results. This paper presents yet another study on the said topic, but with the introduction of our recently-proposed GRU-SVM model[4]. The said ML algorithm combines a type of recurrent neural network (RNN), the gated recurrent unit (GRU)[8] with the support vector machine (SVM)[9]. Along with the GRU-SVM model, a number of MLalgorithms is presented in Section 2.4, which were all applied on breast cancer classification with the aid of WDBC[20].

2.METHODOLOGY

2.1 Machine Intelligence Library

Google TensorFlow was used to implementthemachinelearning algorithms in this study, with the aid of other scientific computing libraries: matplotlib[12], numpy[19], and scikit-learn[15].

2.2 Dataset

The WDBC dataset from Scikit-learn contains 569 samples (212 malignant and 357 benign). Each sample has 30 numerical features computed from a digitized image of an FNA of a breast mass.

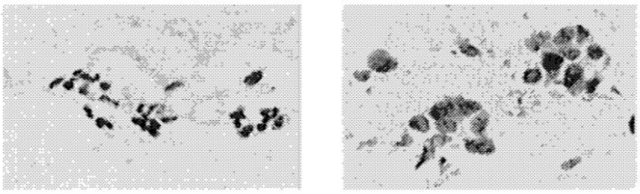

Figure 1: Image from [20] as cited by [21]. Digitized im ages of FNA: (a) Benign, (b) Malignant.

There are 569 data points in the dataset: 212– Malignant, 357 Benign. Accordingly, the dataset features are as follows:(1)radius,(2)texture,(3)perimeter,(4)area,(5)smoothness,(6)compactness, (7)concavity, (8)concavepoints, (9)symmetry,and(10)fractaldimension.Witheachfeaturehavingthreeinformation[20]:(1)mean,(2)standarderror,and(3)“worst”or largest(mean of the three largest values) computed . Thus,having a total of 30 dataset features.

2.3 Preprocessing The dataset was initially loaded using the load_breast_cancer() function from Scikit-learn. It was then converted into a pandas DataFrame format, with the target variable appended as a new column. A thorough inspection of the dataset was carried out, including viewing the head, tail, and random samples, followed by the computation of descriptive statistics. The class labels were encoded numerically, with 0 representing malignant and 1 representing benign tumors. To visualize the relationships between important features, a pairplot was generated using Seaborn. Additionally, a correlation heatmap was plotted to examine the interdependence among the features.

2.4 Models and Mathematical Formulations

This section presents the machine learning(ML)algorithms used inthestudy. The Stochastic Gradient Descent(SGD) learning algorithm was used for all the ML algorithms presented in this section except for GRU-SVM,NearestNeighbor search,andSupportVec tor Machine.The code implementations may be found on line at

https://github.com/Shubha28-collab/machine-learning-model.git

2.4.1 Logistic Regression
Logistic Regression is a linear model used for binary classification. It computes the probability that a data point belongs to a specific class using the sigmoid function: function: h(x) = 1 / (1 + exp(-w^T * x + b))

Here, w and b are the learned weights and bias, and is the input feature vector.

2.4.2 Linear Regression Despite being a regression algorithm, linear regression was adapted for classification in this study. The regression function is:

h_theta(x) = sum(theta_i * x_i) + b

To perform classification, a threshold was applied:

f(h_theta(x)) = 1 if h_theta(x) >= 0.5, otherwise 0

The mean squared error (MSE) loss was minimized:

L(y, theta, x) = (1/N) * sum((y_i - (theta_i * x_i + b))^2)

The parameters theta were learned using the stochastic gradient descent (SGD) algorithm.

2.4.3 Multilayer Perceptron (MLP) The perceptron model was originally introduced by Rosenblatt and expanded into the MLP by incorporating hidden layers with activation functions such as ReLU. The model is represented as:

h_theta(x) = sum(theta_i * x_i) + b f(h_theta(x)) = max(0, h_theta(x))

For this study, a three-layer MLP with architecture 500-500-500 and ReLU activation was used. The cross-entropy function was the loss function used for training.

2.4.4 K-Nearest Neighbors (KNN) KNN is a non-parametric method that assigns class labels based on the majority vote of the nearest k neighbors. Distance is calculated using either Manhattan (L1) or Euclidean (L2) norms:

L1: ||p - q|| = sum(|p_i - q_i|) L2: ||p - q|| = sqrt(sum((p_i - q_i)^2))

This algorithm is purely geometric and does not require training.

2.4.5 Decision Tree A decision tree classifies data by learning simple decision rules inferred from data features. It splits the data to reduce impurity using metrics like Gini Index:

Gini(D) = 1 - sum(p_k^2)

Where p_k is the probability of class k at a node.

2.4.6 Random Forest Random Forest is an ensemble technique that constructs multiple decision trees during training and outputs the mode of their predictions. It helps improve accuracy and control overfitting by averaging results from many trees.

2.4.7 Support Vector Machine (SVM) SVM aims to find the optimal hyperplane that separates classes by maximizing the margin between them. It uses the hinge loss function and is formulated as:

f(x) = sign(w^T * x + b) Loss = (1/2)||w||^2 + C * sum(max(0, 1 - y_i(w^T * x_i + b))^2)

Where C is the regularization parameter and y_i are the class labels.

2.5 Pipeline and Grid Search Implemented using `ImbPipeline`, each model followed this sequence:

- `SMOTE` → `StandardScaler` → classifier
- `GridSearchCV` for hyperparameter tuning

Example parameter grids:

- Logistic Regression: `C = [0.1, 1, 10]`
- SVM: `C = [0.1, 1, 10]`, `kernel = ['linear', 'rbf']`
- KNN: `n_neighbors = [3, 5, 7]`
- MLP: `hidden_layer_sizes = [(50,), (100,)]`, `alpha = [0.0001, 0.001]`

3. RESULTS AND DISCUSSION

3.1 Performance Metrics Each classifier was evaluated based on its cross-validated accuracy and confusion matrix. The best models were then tested on the test split.

MLP Results:

- Accuracy: 99.04%
- Precision, Recall, F1-score: all above 98%

3.2 Pairplot Visualization of Features

A pairplot of all 30 features in the WDBC dataset was generated to visualize their pairwise relationships. The resulting matrix reveals clusters and potential separability between benign and malignant classes. Diagonal plots show the feature distributions, while off-diagonal plots illustrate the scatter relationships between features. Features like mean radius, mean perimeter, and mean area exhibit strong linear relationships and cluster separations. These insights help validate the dataset's structure and the feasibility of effective classification using ML models. This visualization supports feature selection and confirms that several features contribute significantly to the classification task.

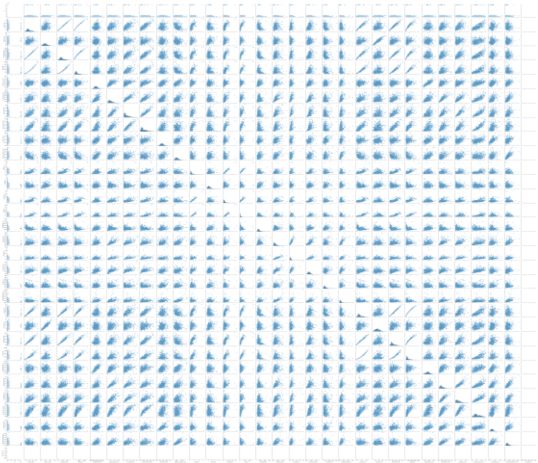

Figure 2: Plotted using matplotlib[12].pairplot

3.3 Focused Pairplot on Selected Features

Additionally, a more focused pairplot was created using only four key features: mean radius, mean texture, mean perimeter, and mean area. This visualization (Figure X) clearly differentiates between the malignant (class 0) and benign (class 1) tumors using hue coloring. The histograms along the diagonal indicate distinct feature distributions across classes. The scatter plots in the off-diagonal cells reveal a positive correlation among the selected features, with visible class separation. The focused visualization highlights the discriminative power of a small subset of features, which reinforces the importance of targeted feature selection for optimizing classification performance.

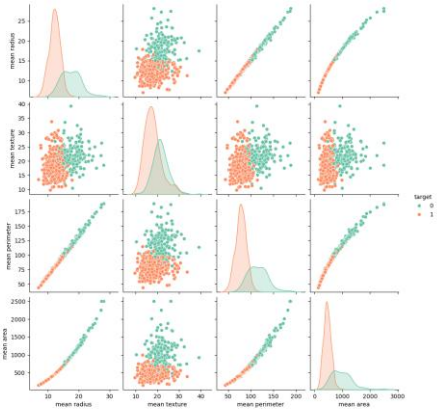

Figure 3: Plotted using matplotlib[12].pairplot.

3.4 Confusion Matrix Analysis for SVM

The confusion matrix for the Support Vector Machine (SVM) classifier demonstrates its effectiveness in distinguishing between malignant and benign tumors. The model correctly identified 61 malignant and 103 benign cases, while it misclassified 3 malignant cases as benign and 4 benign cases as malignant. This indicates a strong performance, with a high true positive and true negative rate. The matrix also shows that the SVM model maintains a balanced precision and recall, making it a reliable choice for medical diagnostic applications where both false negatives and false positives must be minimized.

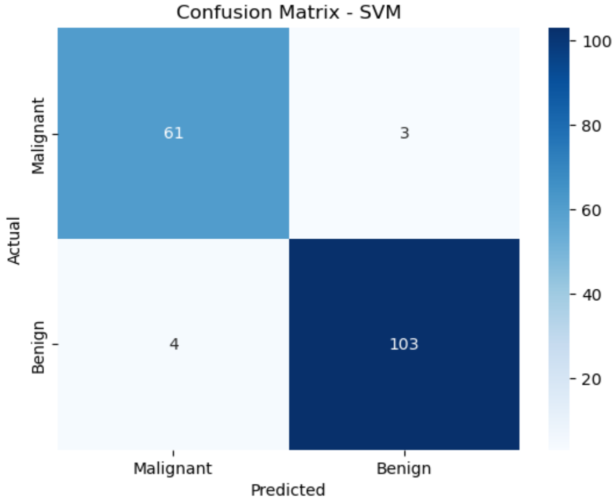

Figure 4: Plotted using matplotlib[12].confusion matrix showing actual and predicted result of benign and malignant cancer cell.

4. EVALUATION

To evaluate the performance of the implemented models, multiple metrics were considered including accuracy, precision, recall, and F1-score. These metrics provide a comprehensive understanding of each classifier's capability in correctly identifying malignant and benign tumors. The confusion matrices generated for models like MLP and SVM further illustrated how each model handled false positives and false negatives.

- **Accuracy** was used to determine the overall correctness of the model.
- **Precision** provided insight into the reliability of positive cancer predictions.
- **Recall** measured the model's ability to detect all actual cancer cases.

F1-Score offered a harmonic mean between precision and recall, especially valuable for imbalanced data.

```
n Random forest trainin..
```

	precision	recall	F1-score	support

0	1.00	1.00	1.00	169
1	1.00	1.00	1.00	286
accuracy			1.00	455
Macroavg	1.00	1.00	1.00	455
Weighted avg	1.00	1.00	1.00	455

n SVM for training..

	precision	recall	F1-score	support
0	0.96	0.80	0.87	169
1	0.89	0.98	0.94	286
accuracy			0.91	455
Macroavg	0.93	0.89	0.90	455
Weighted avg	0.92	0.91	0.91	455

n KNN trainin..

	precision	recall	F1-score	support
accuracy			0.95	455
Macroavg	0.95	0.94	0.94	455
Weighted avg	0.95	0.95	0.95	455

The models were evaluated using a 70/30 train-test split to ensure generalization, and cross-validation was employed during hyperparameter tuning for robustness. MLP achieved the highest evaluation scores overall, while SVM demonstrated balanced performance and interpretability, making it a strong alternative for deployment in clinical decision-support systems.

5. CONCLUSION AND RECOMMENDATION

This study demonstrated the effectiveness of machine learning models in classifying breast cancer cases using the WDBC dataset. Among all the models evaluated, the Multilayer Perceptron (MLP) achieved the best overall performance with an accuracy of 99.04%, while the Support Vector Machine (SVM) also provided strong and balanced classification capabilities.

The application of data preprocessing techniques like normalization and SMOTE significantly contributed to model robustness. Additionally, exploratory data analysis using pairplots and correlation heatmaps revealed meaningful insights that enhanced feature selection and model interpretability.

Recommendations:

- Future studies should consider integrating explainability frameworks like SHAP or LIME to enhance the interpretability of deep learning models in clinical settings.
- Expanding the dataset or applying transfer learning techniques could further generalize the results across diverse populations.
- Deployment of these ML models into real-world diagnostic tools should include rigorous clinical validation and testing across multiple institutions.

REFERENCES

[1] [n. d.]. ([n. d.]). https://seer.cancer.gov/statfacts/html/breast.html

[2] 2017. Cancer Statistics. (Mar 2017). https://www.cancer.gov/about-cancer/ understanding/statistics .

[4] Abien Fred Agarap. 2017. A Neural Network Architecture Combining Gated Recurrent Unit (GRU) and Support Vector Machine (SVM) for Intrusion Detection in Network Traffic Data. arXiv preprint arXiv:1709.03082 (2017).

[5] Abdulrahman Alalshekmubarak and Leslie S Smith. 2013. A novel approach combining recurrent neural network and support vector machines for time series classification. In Innovations in Information Technology (IIT), 2013 9th International Conference on. IEEE, 42–47.

[6] Yoshua Bengio, Ian J Goodfellow, and Aaron Courville. 2015. Deep learning. Nature 521 (2015), 436–444.

[7] Christopher M Bishop. 1995. Neural networks for pattern recognition. Oxford university press.

[8] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014).

[9] C. Cortes and V. Vapnik. 1995. Support-vector Networks. Machine Learning 20.3 (1995), 273–297. https://doi.org/10.1007/BF00994018

[11] Richard HR Hahnloser, Rahul Sarpeshkar, Misha A Mahowald, Rodney J Douglas, and HSebastian Seung. 2000. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. Nature 405, 6789 (2000), 947–951.

[12] J. D. Hunter. 2007. Matplotlib: A 2D graphics environment. Computing In Science &Engineering 9, 3 (2007), 90–95. https://doi.org/10.1109/MCSE.2007.55

[13] Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimiza tion. arXiv preprint arXiv:1412.6980 (2014).