Breast cancer prediction using neural networks

1. Introduction

Breast cancer is the most prevalent cancer globally [1,2], as well as the second leading cause of death among women [3]. Its lethal prognosis is partly due to the generally late established diagnosis, putting emphasis on the pivoting point of an early and accurate diagnosis to improve patient outcome and survival rate [1,3]. Traditional diagnostic methods, such as mammography, MRI and biopsies, are time- and financially consuming, in addition to complex interpretations [1,2]. However, recent involvement of improved computational analysis tools [1-4] have successfully reduced the mortality rate up to 30% [2]. Such tools include machine learning (ML) and deep learning (DL) [2], as powerful assistant tools in medical diagnostics enabling the potential to analyze complex datasets, images and uncover subtle patterns that may be difficult for human observers to spot [2,4,5].

As part of the course “Machine Learning in Biotechnology”, this project aims to develop and evaluate a deep learning model on a real-world medical dataset. The objective is to develop a predictive system capable of classifying breast cancer tumors as either benign or malignant based on a set of diagnostic features. For this purpose, we have chosen two well-established Breast Cancer Wisconsin Datasets [6, 7].

The primary focus of our investigation is the implementation of a Fully Connected Neural Network (FCNN) for the diagnosis of breast cancer. A FCNN is a type of artificial neural network where the neurons in adjacent layers are fully connected, forming a dense web of connections. The fundamental strategy of a FCNN is to learn hierarchical feature representations from the data. The input data is passed through a series of hidden layers. Each neuron in these layers computes a weighted sum of its inputs, adds a bias, and then passes the result through a non-linear activation function (such as ReLU, Rectified Linear Unit). This process allows the network to transform the data in a way that makes complex, non-linear relationships separable. The network's primary statistical assumption is that it does not assume any specific underlying distribution of the data, making it a powerful non-parametric tool. The learning process is achieved through an algorithm called backpropagation, where the model's prediction error (calculated by a loss function) is propagated backward through the network [8]. An optimization algorithm (such as Adam, Adaptive Moment Estimation) then iteratively adjusts the network's weights and biases via gradient descent to minimize this error, effectively "learning" the patterns that differentiate benign from malignant tumors.

To provide a comprehensive performance benchmark and show the FCNN's effectiveness, its results are compared against three classical machine learning models: Logistic Regression, Random Forest, and Gradient Boosting. Performance is assessed using key metrics including accuracy, precision, recall, and F1-score, providing an evaluation of the FCNN's potential as a diagnostic tool [8].

As the WDBC dataset is extensively used for these types of projects we also wanted to try something new, which was to try to predict recurrent cancer with the help of another matching dataset, which included all cases from the original dataset.

The secondary focus of our investigation was the implementation of a FCNN for the prediction of breast cancer severity.

2. Materials and Methods

2.1. Datasets

This study used two publicly available datasets:

1) the **Wisconsin Diagnostic Breast Cancer (WDBC)** [6]**,** a descriptive data set of 30 characteristic numeric measurements of nuclear cells, across 569 participant samples. The features are computed from digitized images of fine needle aspirates (FNA), a minimally invasive biopsy of breast mass. The measurements describe characteristics of the cell nucleus, such as radius, texture, perimeter, and smoothness. The target variable, Diagnosis, is binary, classifying tumors as either Malignant (M) or Benign (B). Class distribution: 357 benign, 212 malignant.
2) the **Wisconsin Prognostic Breast Cancer (WPBC) dataset**[7]. Each record represents data of a breast cancer case follow-up. The patients are exhibiting invasive breast cancer, but no evidence of distant metastases at the time of diagnosis. Most patients are also included in the WDBC dataset and were classified as Malignant. This dataset contains 198 instances, each with the same 30 numeric features computed from FNA images as in the WDBC dataset plus 3 additional numeric variables: Time (recurrence time if Recurrent, disease-free time if Non-recurrent); Tumor Size (diameter of the excised tumor in centimeters) and Lymph Node Status (number of positive axillary lymph nodes observed at time of surgery). The target variable, Outcome, is binary, classifying tumors as either Recurrent (R) or Non-recurrent (NC). Class distribution: 151 Non-recurrent, 47 Recurrent.

A **Merged dataset** was also created for this study where the value of the target variable Outcome from the WPBC dataset was copied to the WDBC dataset to the corresponding patient ID. Then, an additional categorical variable, Severity, was created, where Severity = 0 corresponds to all occurrences classified as Benign in WDBC, Severity = 1 corresponds to all occurrences classified as Malign in WDBC and Non-recurrent in WPBC, Severity = 2 corresponds to all occurrences classified as Malign in WDBC and Recurrent in WPBC, Severity = 3 corresponds to all occurrences classified as Malign in WDBC and not present in WPBC. The Severity variable was created for the purpose of this project and has not been published before.

2.2. Data Preprocessing

Prior to model training, a series of preprocessing steps were performed. The categorical diagnosis labels in the WDBC dataset were encoded into a numerical format, with Malignant (M) mapped to 1 and Benign (B) to 0, using the LabelEncoder from Scikit-learn. The WDBC dataset was split into a training (80%) and a test (20%) dataset. Subsequently, all 30 numeric features were standardized using the StandardScaler from Scikit-learn. This process transformed the features to have a mean of 0 and a standard deviation of 1, ensuring that features with larger scales did not disproportionately influence model training. The standardization was performed separately on the training and test datasets to avoid data leakage.

2.3. Model Implementation

The primary focus of this study was the development and evaluation of a deep learning model for breast cancer classification. To measure its performance, several classical machine learning models were also implemented as baseline comparisons.

2.3.1. Primary Model: Fully Connected Neural Network (FCNN)

A sequential deep learning model was constructed using the TensorFlow Keras API. The network architecture was designed for binary classification and consisted of the following layers:

- An input layer connected to a first hidden layer with 128 neurons and a Rectified Linear Unit (ReLU) activation function.
- A dropout layer with a rate of 0.3, serving as a regularization technique to prevent overfitting.
- A second hidden layer with 64 neurons, also using ReLU activation.
- A second dropout layer with a rate of 0.2.
- A final output layer with a single neuron and a sigmoid activation function, which produces a probability score between 0 and 1 for the binary classification task.

The FCNN was compiled using the Adam optimizer and binary_crossentropy as the loss function, which is standard for binary classification problems.

2.3.2. Baseline Models for Comparison

To benchmark the performance of the FCNN, three well-established machine learning models were implemented using the Scikit-learn library:

- Logistic Regression: A linear model configured with max_iter set to 1000 to ensure convergence.
- Random Forest: An ensemble model consisting of 400 decision trees (n_estimators=400).
- Gradient Boosting: A sequential ensemble model built with 600 estimators (n_estimators=600), a learning rate of 0.01, and a maximum tree depth of 4.

For all baseline models, the random_state was fixed to ensure reproducibility.

2.4. Experimental Design and Model Evaluation during Training

The performance of the FCNN and the baseline models was evaluated using a repeated stratified 5-fold cross-validation procedure. The training dataset is divided into 5 subsets (folds) of roughly equal size. In each iteration of the cross-validation, 1 fold (20% of the data) is used for testing and the remaining 4 folds (80% of the data) are combined and used for training. Each training fold was further splitted (90% for training : 10% for validation) for the calculation and tracking of training and validation metrics and plotting FCNN training history. The proportion of classes (Malignant/Benign) is maintained in each of the 5 folds. This process is repeated 5 times, with a different fold being used as the testing set in each iteration. This way, every sample in the dataset gets to be in the testing set exactly once, providing a more robust evaluation of the model's performance than a single train-test split. This entire validation process was executed three times, each with a different random seed (1, 7, and 42), to ensure the stability and reproducibility of the results.

Within each fold, the FCNN was trained for a maximum of 1000 epochs with a batch size of 32. Its training was also regulated by an early stopping callback. This technique monitored the validation loss (on a 10% split of the fold's training data) and terminated training if no improvement was observed for 5 consecutive epochs, restoring the model weights from the best-performing epoch.

The performance of all models was quantified using four standard metrics: Accuracy, Precision, Recall, and F1-Score. The final reported scores represent the average performance across all 15 folds (5 folds x 3 seeds).

2.5. Final Model Evaluation during Testing

Following cross-validation which is primarily for model selection and comparison, the final chosen models were trained again on the entire initial training dataset (80% split) and then evaluated on the initial test set (20% split) to report the final performance metrics. This final evaluation on the initial test set is distinct from the cross-validation and provides the most unbiased performance estimate for reporting. Confusion matrices and Receiver Operating Characteristic (ROC) curves with Area Under the Curve (AUC) values were generated for comprehensive comparison.

2.5. Complementing WDBC Analysis with a Novel Severity Classification

The implementation of the FCNN using the WDBC dataset was complemented by introducing a novel approach to predict patient outcomes from image data. This was achieved through the creation of a merged dataset (Section 2.1) and a new categorical variable, "Severity". This novel 'Severity' variable allows for a more detailed classification of malignant cases based on recurrence information, providing a richer dataset for predicting different levels of disease severity. The models developed using the merged dataset leverage this new variable to explore the prediction of these distinct severity classes.

2.5.1. Experimental Design

The experimental design involved training and evaluating three different neural network models to predict breast cancer severity based on the provided dataset. The primary challenge addressed was the severe class imbalance observed in the 'Severity' variable. The rationale for training three distinct models was to explore different approaches to handling the class imbalance and assess their impact on model performance, particularly in predicting the less frequent severity classes:

1. **Model 1** (All Severity Classes): This model was trained using the full dataset, including the minority Severity 3 class. Class weights were applied during training to give more importance to the minority classes and mitigate the imbalance. This approach aimed to see if the model could learn to differentiate all four severity levels, despite the data distribution.

2. **Model 2** (Severity 0, 1, 2 with Adaptive Synthetic Sampling (ADASYN)): Due to the extremely low count of Severity 3 samples and the difficulty in reliably predicting this class, Severity 3 instances were removed from the dataset for this and the subsequent model. This simplified the classification problem to three classes (0, 1, and 2). ADASYN was then applied to the training data only. ADASYN is an oversampling technique that generates synthetic samples for minority classes, focusing on regions of the feature space that are harder to learn, thus aiming for a more balanced and potentially more informative training set.

3. **Model 3** (Severity 0, 1, 2 with Synthetic Minority Over-sampling Technique (SMOTE)): Similar to Model 2, Severity 3 instances were removed, reducing the problem to three classes. For this model, SMOTE was applied to the training data. SMOTE is another oversampling technique that generates synthetic samples along the line segments connecting minority class instances to their nearest neighbors. This approach aims to create a more balanced training distribution by interpolating new samples for the minority classes.

For all three models, a consistent data splitting strategy was employed: 60% for training, 16% for validation, and 24% for testing (percentages refer to the relevant data subset, either the original full dataset or the dataset with Severity 3 removed). This allowed for an unbiased evaluation of the models on unseen data.

Models were evaluated during the training process using the validation set. Key metrics monitored included the validation loss and accuracy. To prevent overfitting and optimize training, callbacks were utilized:

Early Stopping: Monitored the validation loss and halted training if there was no significant improvement for a specified number of epochs (patience=10), restoring the model weights from the epoch with the best validation loss. This ensured that the model did not continue training excessively after reaching its optimal performance on unseen data.

ReduceLROnPlateau: Monitored the validation loss and reduced the learning rate of the optimizer if the validation loss stopped improving for a specified number of epochs (patience=5). This helped the model converge more effectively by taking smaller steps in the parameter space when the loss plateaued.

These techniques allowed for dynamic adjustment of the training process based on the model's performance on the validation set, promoting generalization and preventing the models from becoming overly specialized to the training data.

2.5.1. Model Evaluation

The final trained models were evaluated on the test sets to assess their generalization performance on unseen data. For Model 1, the evaluation was conducted on the test set containing all four severity classes (0, 1, 2, and 3). For Model 2 (ADASYN) and Model 3 (SMOTE), the evaluation was performed on the test set from which Severity 3 instances had been removed, as these models were trained to classify only the three remaining classes (0, 1, and 2).

To gain a deeper understanding of the models' performance across different classes, confusion matrices were generated. Additionally, per-class metrics were provided including precision, recall, and F1-score, as well as macro and weighted averages.

2.6. Software and Libraries

All data analysis and modeling were performed in Python (version 3.x). The primary libraries used were pandas for data manipulation, scikit-learn for implementing the baseline models and preprocessing, tensorflow with keras for the FCNN implementation, and matplotlib and seaborn for data visualization. The experiments were conducted within a Google Colab environment. We also used some help with coding from Microsoft Copilot and Google Gemini.

3. Results

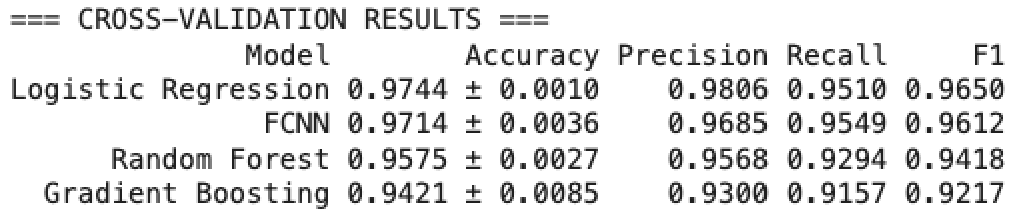
The performance of the primary model, a FCNN, was evaluated and benchmarked against three classical machine learning models: Logistic Regression, Random Forest, and Gradient Boosting.

3.1. Overall Performance of the FCNN and Baseline Models during Cross-Validation

The FCNN demonstrated strong and stable performance in classifying breast cancer tumours, achieving a mean accuracy of 0.9714 ± 0.0036 across the repeated cross-validation runs.

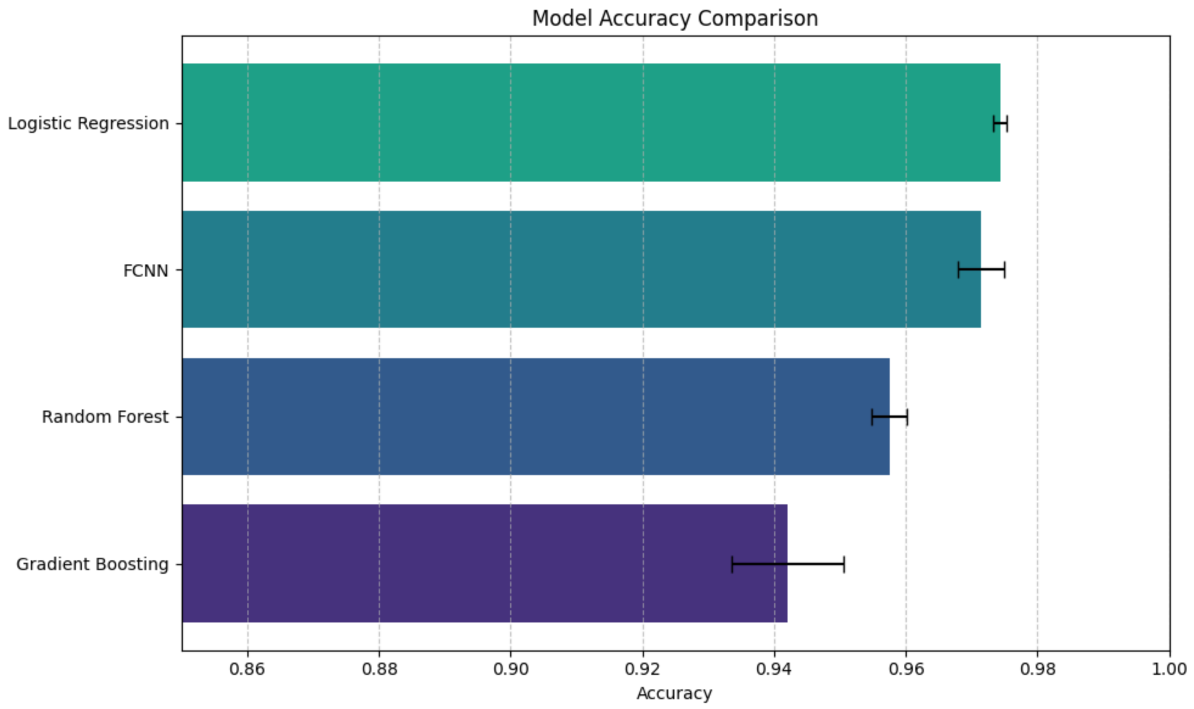
We compared these results against the baseline models, as summarized in Table 1 and visualized in Figure 1. The Logistic Regression model slightly outperformed the FCNN, achieving the highest mean accuracy of 0.9744 ± 0.0010 and also leading in precision (0.9806) and F1-score (0.9650). In contrast, the other two baseline models, Random Forest and Gradient Boosting (ensemble models, recorded lower mean accuracies of 0.9575 and 0.9421, respectively. All models display low standard deviations (n=3).

Table 1: Cross-Validation Results Summary



```
=== CROSS-VALIDATION RESULTS ===
                 Model        Accuracy Precision Recall     F1
Logistic Regression 0.9744 ± 0.0010    0.9806 0.9510 0.9650
               FCNN 0.9714 ± 0.0036    0.9685 0.9549 0.9612
      Random Forest 0.9575 ± 0.0027    0.9568 0.9294 0.9418
  Gradient Boosting 0.9421 ± 0.0085    0.9300 0.9157 0.9217
```

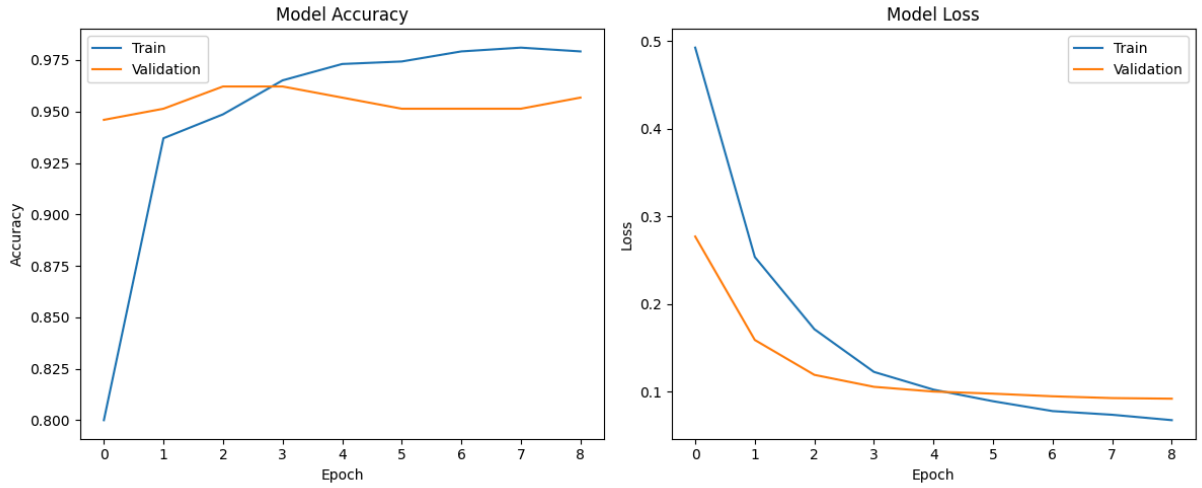
Figure 1: Cross-Validation Model Accuracy Comparison



3.2. FCNN Training Dynamics

The training history of the FCNN was analyzed to ensure stable learning and to check for overfitting (Figure 2). The model's training and validation loss curves decreased rapidly and converged, while the accuracy curves increased steadily and plateaued at a high level. The close alignment between the training and validation curves for both metrics indicates that the regularization techniques (dropout) and early stopping were effective in preventing overfitting, leading to a well-generalized model.

Figure 2: FCNN Model Accuracy and Loss History



3.3. Performance of the FCNN and Baseline Models on the Test Set

Following cross-validation, the final trained models were evaluated on the independent test set (20% of the WDBC dataset) to provide an unbiased assessment of their expected performance on unseen data. The performance metrics on this final test set are summarized in Table 2 and the corresponding confusion matrices are presented in Figures 3-6. Receiver Operating Characteristic (ROC) curves were also generated and are presented in Figure 7. In the context of breast cancer diagnosis, minimizing false negatives – instances where a malignant tumor is incorrectly classified as benign – is critically important to avoid missed diagnoses. Recall, which measures the proportion of actual positive cases that are correctly identified, is a key metric for assessing a model's ability to minimize false negatives.

The analysis of the ROC curves and their corresponding AUC values (Figure 7) reveal high discriminatory power across all models, indicating their strong ability to separate the two classes.

On the final test set, the **FCNN** demonstrated the highest accuracy at 0.9825. Notably, the FCNN achieved perfect precision (1.0000), indicating that all instances predicted as Malignant by the FCNN on this test set were indeed Malignant (zero false positives). Its recall was 0.9524, meaning it correctly identified approximately 95.24% of the actual malignant cases. This corresponds to a low number of false negatives (2 cases) in the confusion matrix. The FCNN's F1-score, which balances precision and recall, was 0.9756.

The **Random Forest** model also performed exceptionally well on the final test set, achieving an accuracy of 0.9737. Like the FCNN, the Random Forest model also showed perfect precision (1.0000) on this test set (zero false positives). Its recall was

0.9286, correctly identifying approximately 92.86% of the actual malignant cases, indicating a relatively low rate (3 cases) of false negatives. Its F1-score was 0.9630.

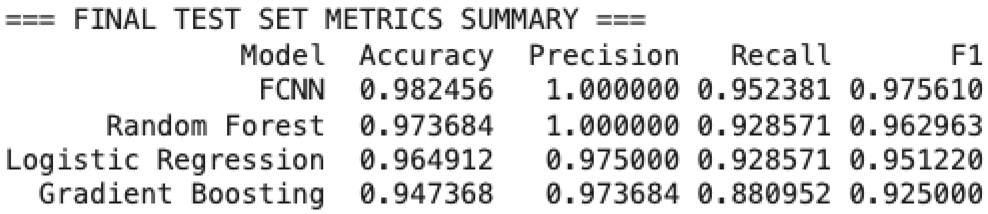
The **Logistic Regression** model achieved an accuracy of 0.9649 on the test set, with a precision of 0.9750. Its recall was also 0.9286, similar to the Random Forest, correctly identifying about 92.86% of malignant cases. The F1-score was 0.9512.

The **Gradient Boosting** model showed an accuracy of 0.9474. Its precision was 0.9737. However, its recall was lower at 0.8810, meaning it missed a higher proportion of actual malignant cases (5 cases) compared to the other models, resulting in more false negatives on this test set. The F1-score was 0.9250.

The confusion matrices (Figures 3-6) visually confirm these results, showing the count of false negatives for each model. While all models demonstrated strong performance, the **FCNN** achieved the highest recall, indicating it was the most effective among the evaluated models at minimizing false negatives on this test set, which is crucial for reliable cancer diagnosis.

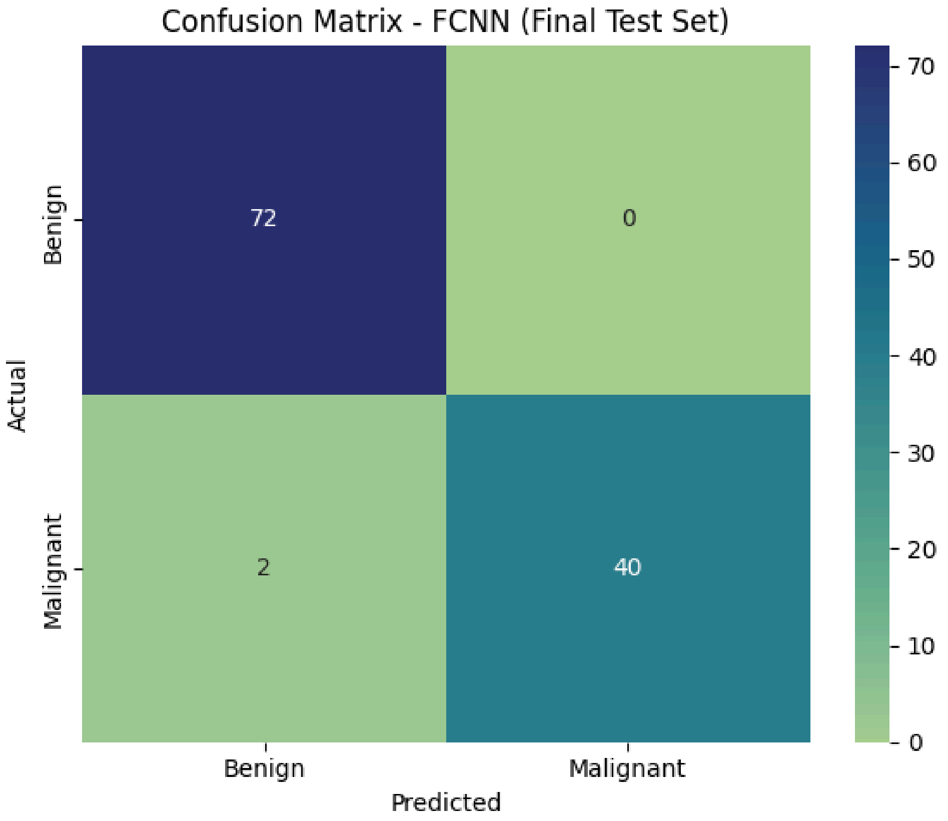
The discriminative ability of the models was assessed using ROC curves (Figure 7). The FCNN demonstrated excellent performance with an Area Under the Curve (AUC) of 0.99. This was on par with the Random Forest and Gradient Boosting models and only marginally below the perfect 1.00 AUC achieved by the Logistic Regression benchmark, confirming the FCNN's strong capability to distinguish between benign and malignant classes.

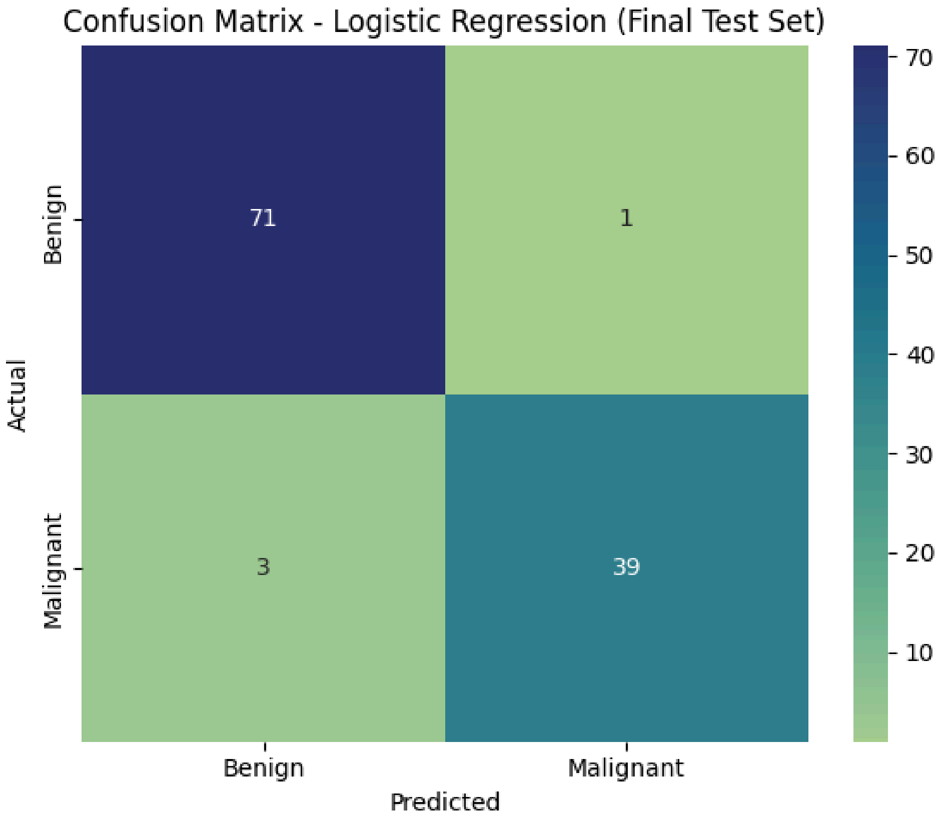
Table 2: Final Test Results Summary



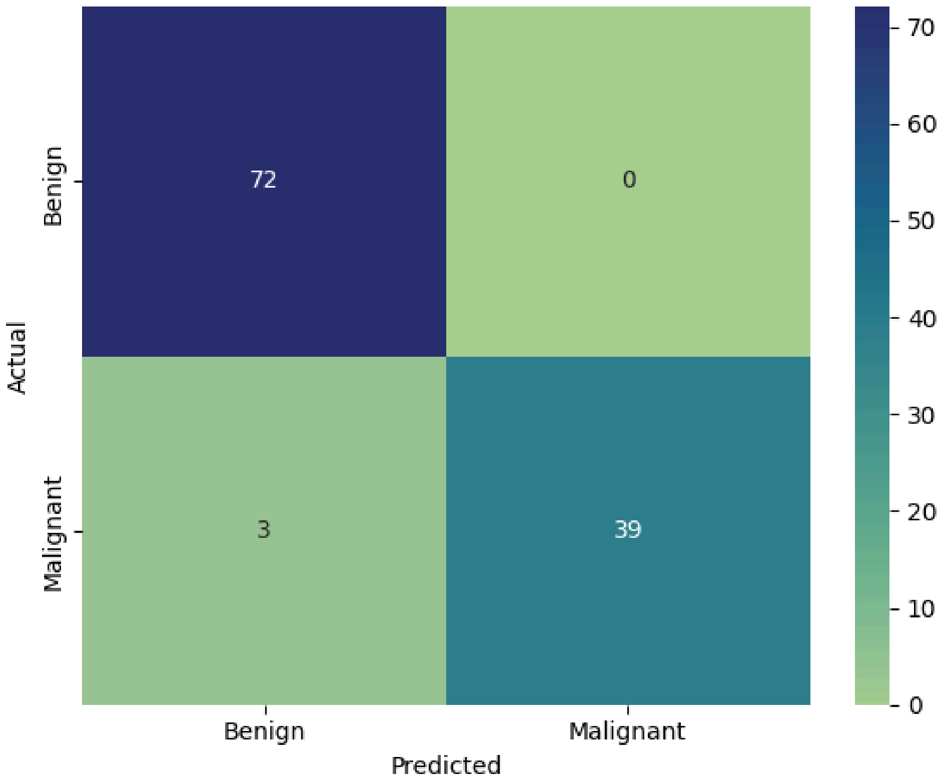
```
=== FINAL TEST SET METRICS SUMMARY ===
              Model  Accuracy  Precision   Recall       F1
               FCNN  0.982456   1.000000 0.952381 0.975610
      Random Forest  0.973684   1.000000 0.928571 0.962963
Logistic Regression  0.964912   0.975000 0.928571 0.951220
  Gradient Boosting  0.947368   0.973684 0.880952 0.925000
```

Figures 3-6: Confusion Matrices


Confusion Matrix - FCNN (Final Test Set)
Actual
Benign
Malignant
72
0
2
40
Benign
Malignant
Predicted
70
60
50
40
30
20
10
0


Confusion Matrix - Logistic Regression (Final Test Set)
Actual
Benign
Malignant
71
1
3
39
Benign
Malignant
Predicted
70
60
50
40
30
20
10

Confusion Matrix - Random Forest (Final Test Set)



Confusion Matrix - Gradient Boosting (Final Test Set)

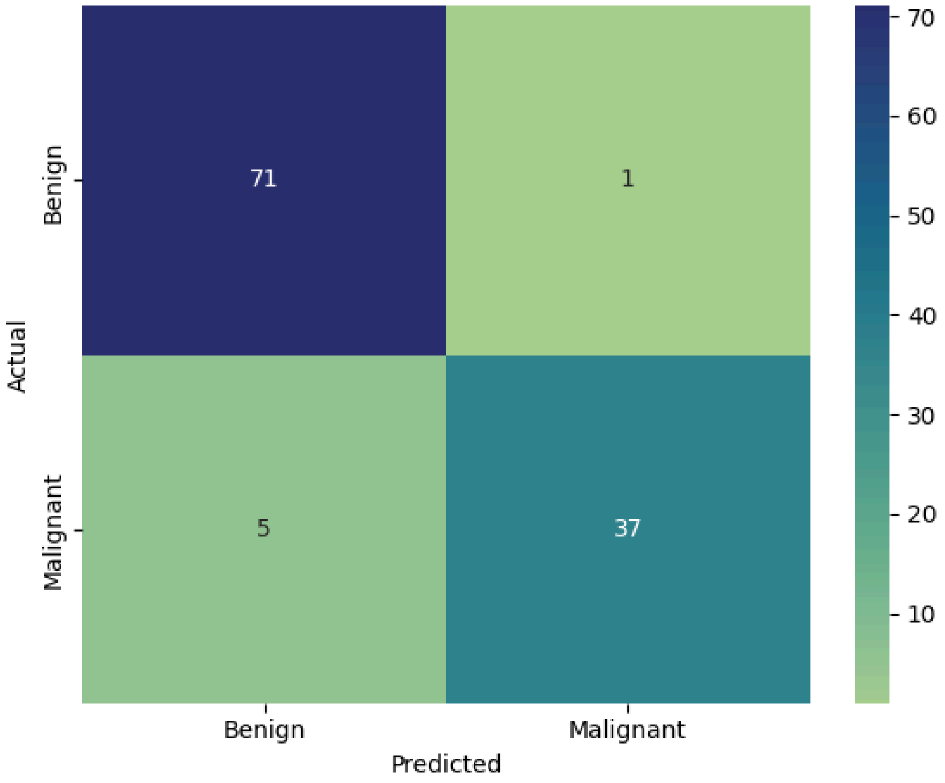
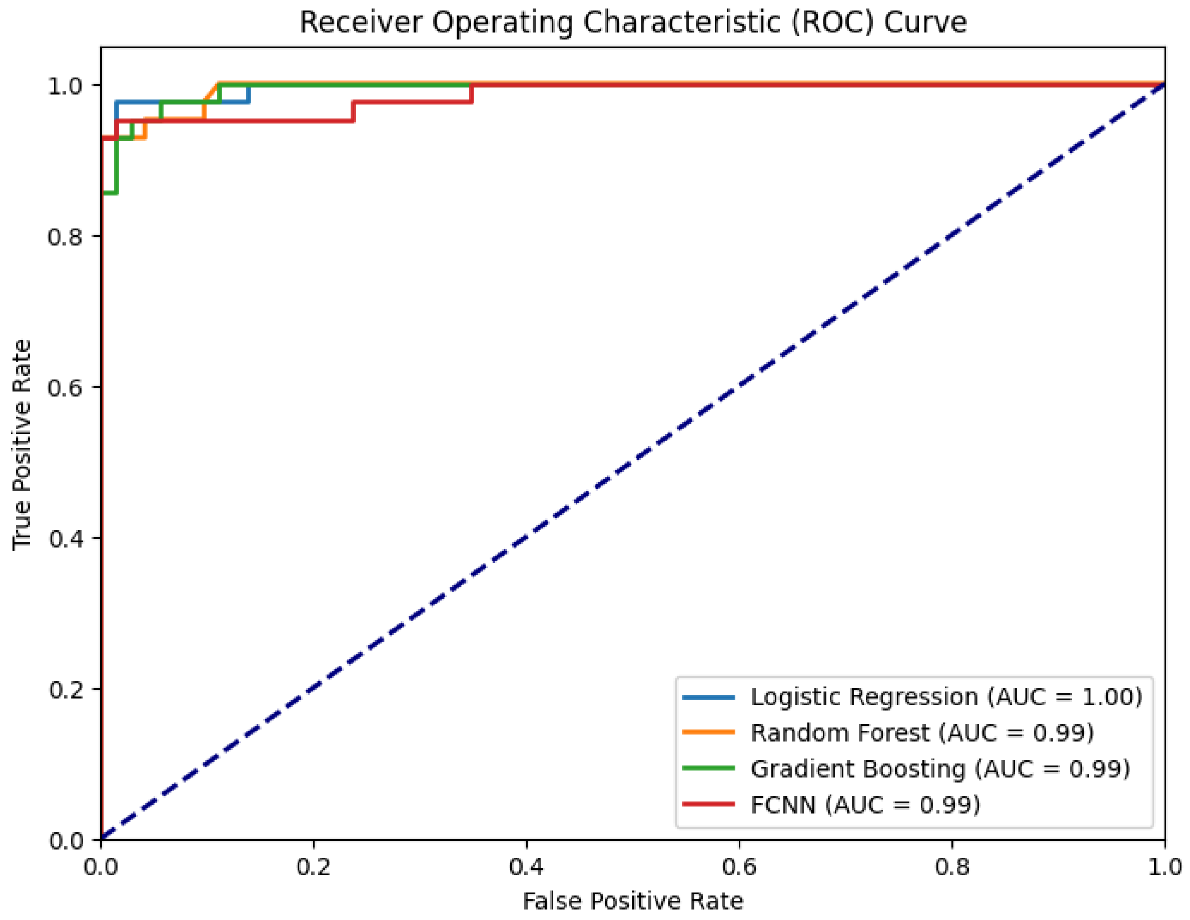


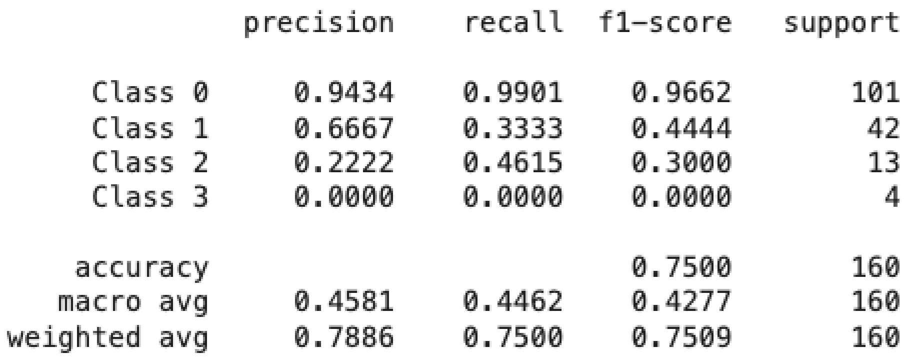
Figure 7: Receiver Operating Characteristic (ROC) Curve



3.4. Complementing WDBC Analysis with a Novel Severity Classification

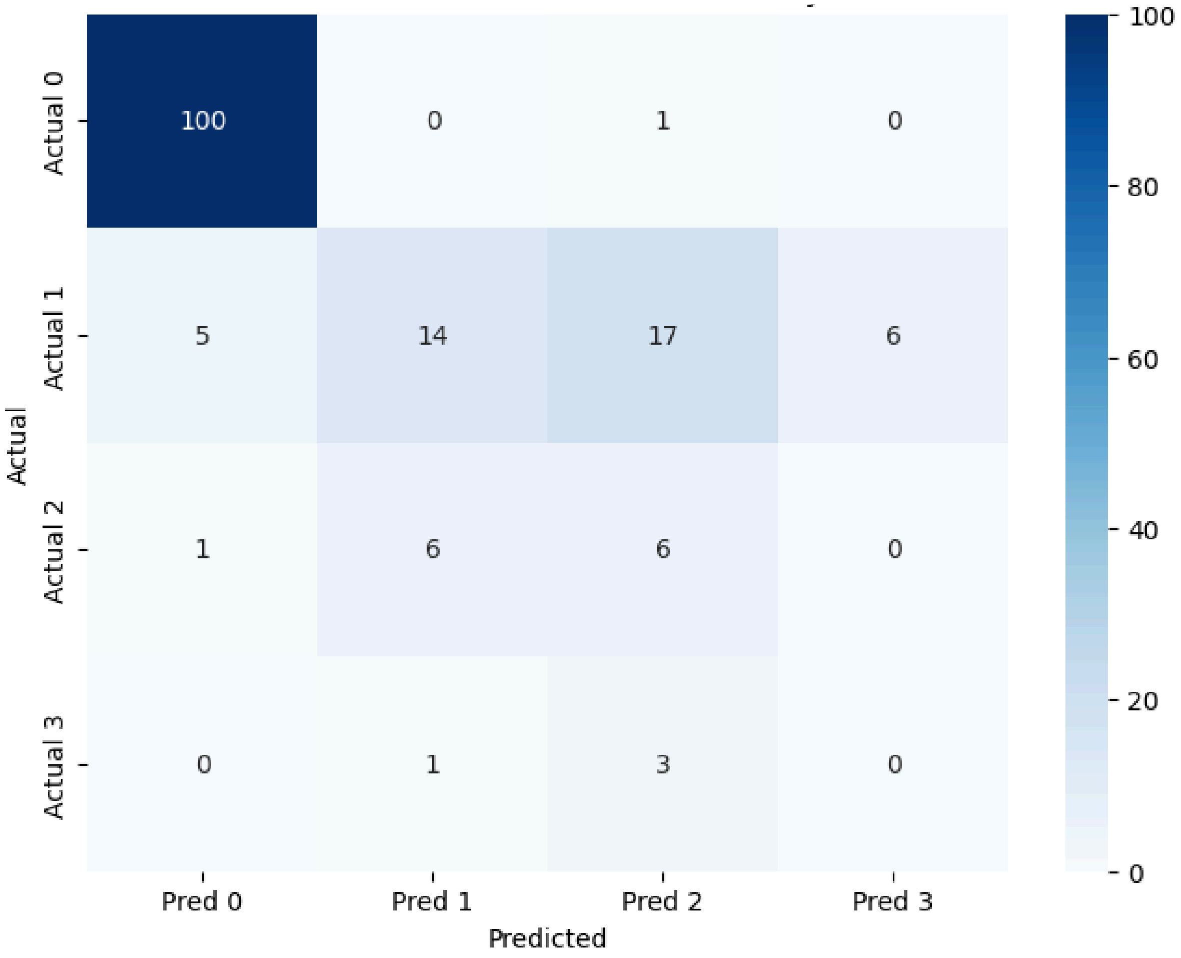
The key performance metrics on the test set for each of the three models are summarized below:

Model 1 (All Classes): This model showed good performance on Class 0 (high precision and recall), but struggled significantly with minority classes, particularly Class 3, which had zero precision, recall, and F1-score. Class 1 and Class 2 also had considerably lower recall compared to Class 0. This highlights the challenge of classifying highly imbalanced minority classes without specific handling for them.

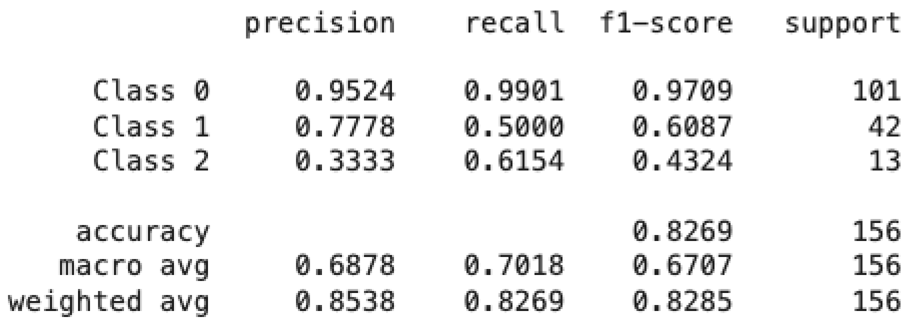


```
              precision    recall  f1-score   support

     Class 0     0.9434    0.9901    0.9662       101
     Class 1     0.6667    0.3333    0.4444        42
     Class 2     0.2222    0.4615    0.3000        13
     Class 3     0.0000    0.0000    0.0000         4

    accuracy                         0.7500       160
   macro avg     0.4581    0.4462    0.4277       160
weighted avg     0.7886    0.7500    0.7509       160
```

Confusion matrix

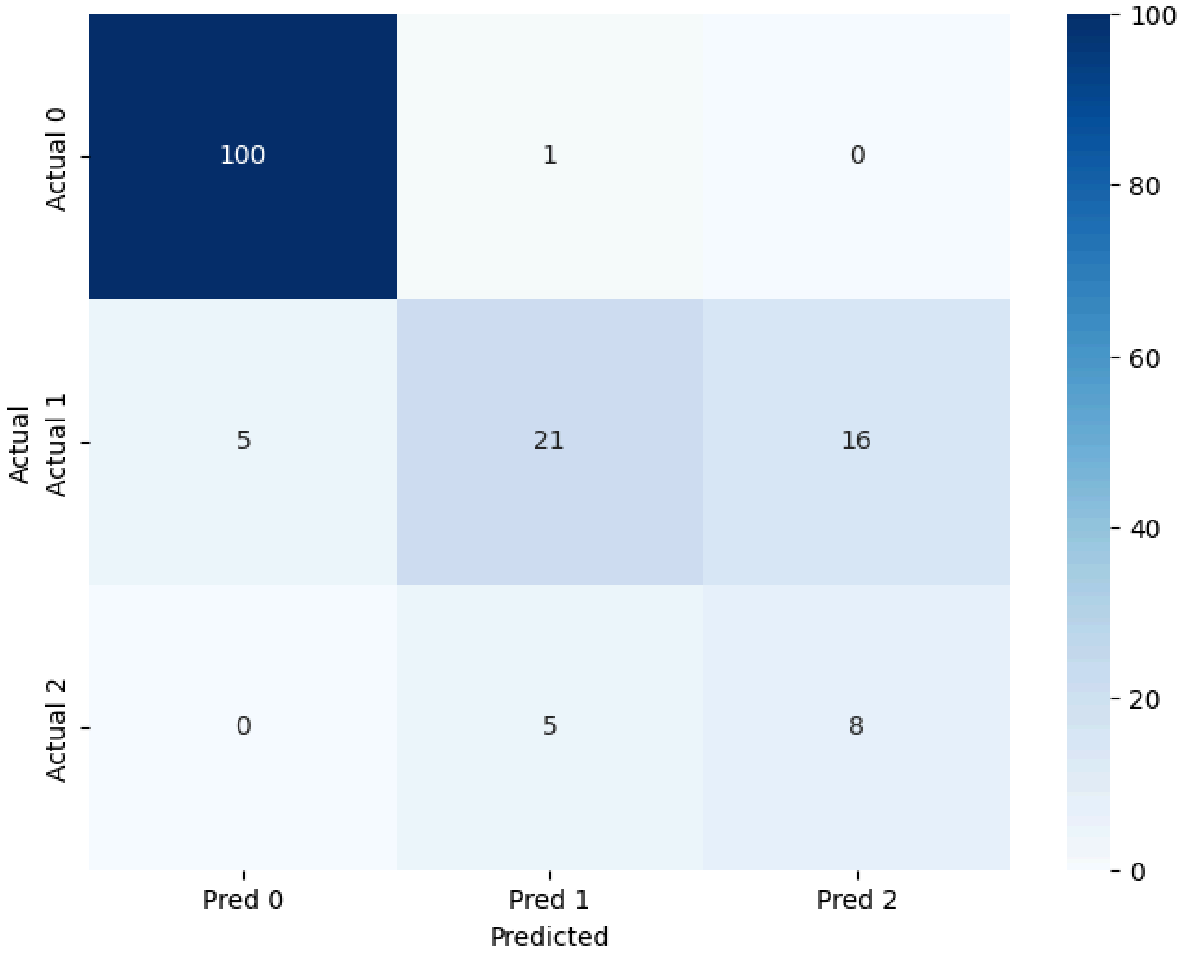


Model 2 (ADASYN): By removing Severity 3 and using ADASYN, this model showed improved overall accuracy compared to Model 1. The performance on Class 0 remained strong. There was a big improvement in recall for Classes 1 and 2 compared to Model 1, suggesting ADASYN helped in identifying more true positives for this class.

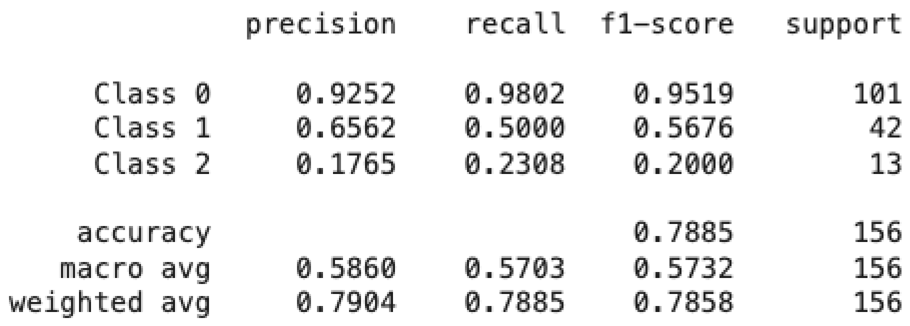


```
              precision    recall  f1-score   support

     Class 0     0.9524    0.9901    0.9709       101
     Class 1     0.7778    0.5000    0.6087        42
     Class 2     0.3333    0.6154    0.4324        13

    accuracy                         0.8269       156
   macro avg     0.6878    0.7018    0.6707       156
weighted avg     0.8538    0.8269    0.8285       156
```

Confusion matrix

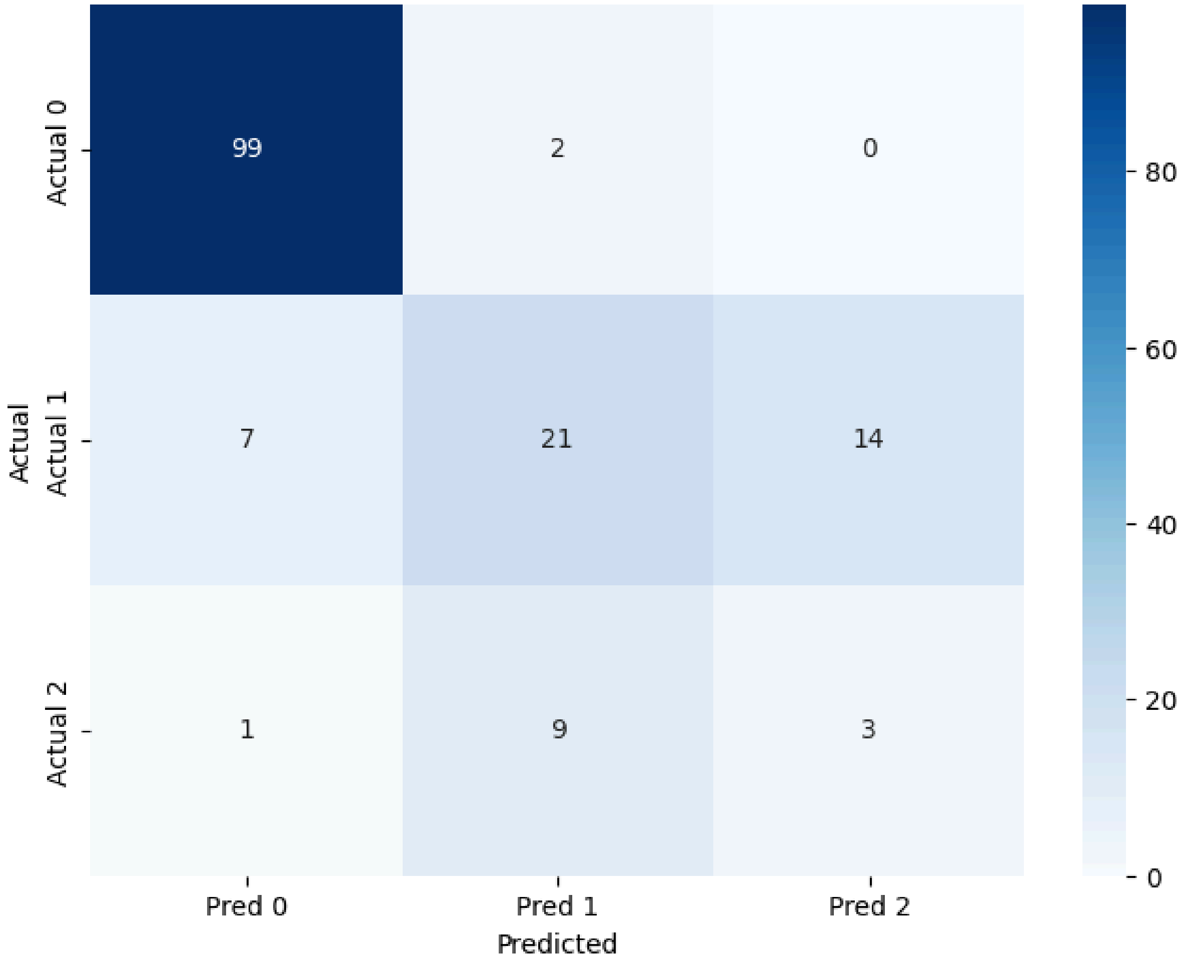


Model 3 (SMOTE): This model, also trained on filtered data (Severity 0, 1, 2) and using SMOTE also showed improved overall accuracy compared to Model 1. The performance on Class 0 remained strong. There was an improvement in recall for Class 1 but not for Class 2 compared to Model 1.



```
              precision    recall  f1-score   support

     Class 0     0.9252    0.9802    0.9519       101
     Class 1     0.6562    0.5000    0.5676        42
     Class 2     0.1765    0.2308    0.2000        13

    accuracy                         0.7885       156
   macro avg     0.5860    0.5703    0.5732       156
weighted avg     0.7904    0.7885    0.7858       156
```

Confusion matrix:



4. Discussion

In this study, we developed and evaluated a FCNN for the classification of breast cancer tumors and benchmarked its performance against three classical machine learning models. The investigation revealed that while all models performed at a high level, the results present a nuanced view of the role of model complexity in this specific diagnostic task.

4.1. Interpretation of Model Performance

The primary model, a FCNN, demonstrated excellent performance, achieving a mean accuracy of 98.25% with very low variance, indicating a stable and reliable model. Its ability to achieve this level of accuracy shows the power of neural networks to learn complex, non-linear patterns directly from data. The FCNN was the top performer compared to two ensemble models, Random Forest and Gradient Boosting. This suggests that the hierarchical feature representations learned by the deep neural network may have captured the underlying data structure more effectively than the decision-boundary-based approaches of the two tree-based ensembles for this dataset.

The simpler, linear Logistic Regression model also achieved a high accuracy (96.49%) and a perfect AUC of 1.00. The good performance of a linear model suggests that the 30 engineered features in the Wisconsin dataset are highly informative and have a strong, near-linear relationship with the diagnosis. In essence, the problem, as defined by these features, is largely linearly separable. This explains why a complex, non-linear model like the FCNN, while highly effective, did not provide a big performance boost over a simpler, well-suited algorithm.

4.2. Clinical Significance and Error Analysis

In a clinical setting like cancer diagnosis, the type of error is often more important than the overall accuracy. A false negative (classifying a malignant tumor as benign) has far more severe consequences than a false positive (classifying a benign tumor as malignant, which may lead to a follow-up biopsy).

From this perspective, the FCNN performed exceptionally well. The confusion matrix analysis revealed that the FCNN produced the fewest false negatives overall. This is a critically important strength, demonstrating FCNN's high sensitivity (Recall) in identifying malignant cases. The other models produced more false negatives making them less reliable for this specific high-stakes application. This highlights that the FCNN provides a greater degree of safety in minimizing the most dangerous type of diagnostic error.

4.3. Limitations of the Study

While this study yielded clear results, several limitations should be acknowledged.

1. **Dataset size and complexity:** The Wisconsin dataset, while a classic benchmark, is relatively small (569 samples). Deep learning models like FCNNs typically showcase their full potential on much larger datasets where more complex, subtle patterns can be discovered. The model's performance here is likely constrained by the limited data volume.
2. **Pre-engineered features:** We used a dataset with 30 pre-calculated features. The true power of modern deep learning, particularly with Convolutional Neural Networks (CNNs), lies in their ability to perform automatic feature

extraction directly from raw data, such as histopathological images. Our study did not use this approach.

3. **Hyperparameter optimization:** The FCNN architecture and hyperparameters were manually selected and not exhaustively optimized. A more systematic search using techniques like grid search or Bayesian optimization could potentially yield a neural network with even better performance.

4.4. Complementing WDBC Analysis with a Novel Severity Classification

This study investigated the performance of three distinct neural network models in predicting breast cancer severity, addressing the significant class imbalance present in the dataset.

Model 1, trained on all four severity classes with class weights, achieved a test accuracy of 0.7500. While it performed very well in identifying Class 0 (precision 0.9434, recall 0.9901), its performance significantly degraded for the minority classes, particularly Class 3, which had a precision and recall of 0.0000. This highlights the challenge of accurately predicting extremely rare classes even with class weighting. The confusion matrix for Model 1 clearly shows that most instances of the minority classes were misclassified as Class 0.

To address the difficulty in predicting Severity 3, subsequent models focused on the remaining three classes (0, 1, and 2). Model 2, incorporating ADASYN on the training data, demonstrated improved overall performance with a test accuracy of 0.8269. The confusion matrix and classification report for Model 2 show a noticeable improvement in predicting Class 1 and Class 2 compared to Model 1. The precision and recall for Class 1 were 0.7778 and 0.5000, respectively, while for Class 2, they were 0.3333 and 0.6154. This suggests that ADASYN helped in generating more diverse synthetic samples, leading to better discrimination of the minority classes in the reduced dataset.

Model 3, utilizing SMOTE for oversampling on the three-class problem, achieved a test accuracy of 0.7885. While the overall accuracy was slightly lower than Model 2, the classification report indicates that SMOTE also contributed to improving the prediction of minority classes compared to Model 1. Class 1 had precision and recall of 0.6562 and 0.5000, and Class 2 had precision and recall of 0.1765 and 0.2308. The performance on Class 2 was notably lower than with ADASYN.

Comparing Model 2 (ADASYN) and Model 3 (SMOTE) on the reduced dataset, Model 2 with ADASYN generally exhibited better performance metrics, particularly in terms of precision and recall for the minority classes (Class 1 and 2). This suggests that, in this specific case, ADASYN's approach of generating synthetic samples in harder-to-learn areas of the feature space was more effective than SMOTE's linear interpolation method for this dataset and model architecture.

SMOTE, by simply interpolating between nearest neighbors, might not have generated enough diverse samples in these critical boundary areas. This could lead to the model having a harder time distinguishing between classes in those ambiguous regions, resulting in slightly lower performance for the minority classes compared to ADASYN. Therefore, the adaptive nature of ADASYN, which prioritizes generating synthetic samples for minority instances that are more difficult to classify, likely contributed to its slightly better performance in this specific instance.

5. Future Work and Conclusion

Based on these findings and limitations, several avenues for future research are apparent. First, applying the FCNN model to a larger, more diverse dataset would be a natural next step to validate its generalizability and explore its full potential. Second, a more advanced study could involve using a Convolutional Neural Network (CNN) to classify tumors directly from raw histopathological images, bypassing the need for manual feature engineering.

In conclusion, this study successfully demonstrates that a FCNN is a highly effective and reliable tool for breast cancer diagnosis using the Wisconsin dataset. It achieved the highest accuracy on the final test set and showed superior reliability in minimizing critical false-negative errors compared to ensemble methods. The fact that even a simpler linear model like Logistic Regression also achieved very high performance highlights an important principle in machine learning: for well-defined datasets with informative features, even simpler models can be highly effective. However, the FCNN's strong performance establishes it as a powerful and promising candidate for more complex diagnostic challenges in medical imaging.

When investigating the performance of three distinct neural network models in predicting breast cancer severity, addressing the significant class imbalance present in the dataset, removing the extremely rare Severity 3 class and employing oversampling techniques like ADASYN and SMOTE improved the model's ability to predict the remaining minority severity classes compared to training on the full, highly imbalanced dataset with only class weights. Among the oversampling methods tested, ADASYN showed an advantage in this particular application. Future work could explore other oversampling techniques, different model architectures, or feature engineering to further enhance the prediction of breast cancer severity in imbalanced datasets.

6. References

1. Shafique R, Rustam F, Choi GS, Díez I de la T, Mahmood A, Lipari V, et al. Breast Cancer Prediction Using Fine Needle Aspiration Features and Upsampling with Supervised Machine Learning. Cancers (Basel). 2023 Jan

22;15(3):681.

2. Shah H, Agrawal S, Oza P, Tanwar S. Comparative Study on Machine Learning Algorithms for Breast Cancer Diagnosis. Procedia Computer Science. 2025 Jan 1;259:1994–2003.

3. Khan J, Golilarz NA, Li JP, Kuzeli P, Addeh A, Haq AU. Breast Cancer Diagnosis using Digitized Images of FNA Breast Biopsy and Optimized Neurofuzzy System. In: 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP) [Internet]. 2020 [cited 2025 Oct 17]. p. 286–90. Available from: https://ieeexplore.ieee.org/document/9317387

4. Abubakar M, Duggan MA, Fan S, Pfeiffer RM, Lawrence S, Mutreja K, et al. Unraveling the role of stromal disruption in aggressive breast cancer etiology and outcomes. J Natl Cancer Inst. 2025 Aug 1;117(8):1673–88.

5. Street WN, Wolberg WH, Mangasarian OL. Nuclear feature extraction for breast tumor diagnosis. In: Acharya RS, Goldgof DB, editors. San Jose, CA; 1993 [cited 2025 Oct 21]. p. 861–70. Available from: http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=1008972

6. Wolberg W, Mangasarian O, Street N, Street W. Breast Cancer Wisconsin (Diagnostic) [dataset]. 1993. UCI Machine Learning Repository. Available from: https://doi.org/10.24432/C5DW2B.

7. Wolberg W, Street W, Mangasarian O. Breast Cancer Wisconsin (Prognostic) [dataset]. 1995. UCI Machine Learning Repository. Available from: https://doi.org/10.24432/C5GK50.

8. Müller AC, Guido S. Introduction to machine learning with Python: a guide for data scientists. First edition. Sebastopol, CA: O’Reilly Media; 2017.

7. Appendix

Python notebooks:

BreastCancerClassification_FCNN_v2.ipynb

https://github.com/FilipeAMarques/ML-in-Biotech-Project/blob/1293731af644228a5a507ee1e81eb13ad5d00927/BreastCancerClassification_FCNN_v2.ipynb

https://colab.research.google.com/drive/1F8ahI-_-Z2Lq5hp8DYN-O4CSXpax2-b8

https://github.com/FilipeAMarques/ML-in-Biotech-Project/blob/aa53db5eeb0054827634b43d68deef3bf44ff9c9/Model3(Severity)v2.ipynb

Python notebooks as a PDF file:

https://github.com/FilipeAMarques/ML-in-Biotech-Project/blob/83a99ff945e028926eb05476920cb3e15478cfa6/Project_BreastCancerClassification_FCNN_v2.ipynb.pdf

https://github.com/FilipeAMarques/ML-in-Biotech-Project/blob/aa53db5eeb0054827634b43d68deef3bf44ff9c9/Project_Model3(Severity)v2.ipynb.pdf