





Enhancing Breast Cancer Classification with XGBoost: A Comparative Study of Machine Learning Models

 Maher Abbass ^{*α},  Ali Daher ^{†α},  Bassel Fakhry ^{‡α}, and  Moustapha Ghandour ^{§α}

^α American University of Beirut (AUB)

ABSTRACT

Abstract: Breast cancer classification is pivotal in medical diagnosis, guiding treatment strategies and prognostic assessments. In this research, we evaluate the efficacy of various machine learning models, including convolutional neural networks (CNNs), XGBoost, and a hierarchical XGBoost approach, in discerning breast cancer subtypes. Additionally, we focus on addressing class imbalances, particularly in minority classes, to ensure comprehensive classification performance. Our results showcase the CNN model's performance, achieving an accuracy of 78%. While demonstrating high precision for the majority class, the CNN model struggles with recall and F1-score for minority classes. In contrast, the XGBoost model exhibits superior accuracy of 93% and demonstrates promising performance in classifying minority classes, evidenced by an average F1 score of 0.4. Furthermore, we explore a hierarchical XGBoost approach, yielding a balanced accuracy of 75.92%. However, overfitting is observed in a secondary model aimed at classifying minority classes. Our comparative analysis underscores the robustness of XGBoost in handling imbalanced datasets, culminating in enhanced breast cancer classification accuracy. These findings underscore the potential of XGBoost as a valuable tool in clinical settings for precise breast cancer subtype classification, thereby facilitating tailored treatment strategies and improved patient outcomes.

1 INTRODUCTION

Breast cancer classification is a critical task in medical diagnosis and treatment planning. Here's a brief description of the four types of breast cancer mentioned:

1. Breast Invasive Ductal Carcinoma (IDC):

- IDC is the most common type of breast cancer, representing about 80% of all cases.
- It begins in the milk ducts of the breast and invades nearby tissue as it grows.
- Diagnosis often involves a biopsy to examine tissue samples under a microscope.

2. Breast Mixed Ductal and Lobular Carcinoma:

- This type of breast cancer contains a combination of both ductal and lobular cancer cells.
- It's less common than pure IDC or invasive lobular carcinoma (ILC).

*✉ mna73@aub.edu.lb

†✉ aad40@aub.edu.lb

‡✉ bif02@aub.edu.lb

§✉ mmg30@aub.edu.lb

- Treatment approaches may vary depending on the proportion of ductal and lobular components.

3. Breast Invasive Lobular Carcinoma (ILC):

- ILC begins in the lobules, which are the milk-producing glands of the breast.
- It tends to spread more diffusely throughout the breast and may be challenging to detect on imaging tests.
- Diagnosis often requires a combination of imaging studies and biopsy.

4. Breast Invasive Mixed Mucinous Carcinoma:

- This is a rare subtype of breast cancer characterized by a mix of invasive cancer cells and mucinous (gelatinous) cells.
- It may present as a distinct mass on imaging studies, but definitive diagnosis requires biopsy and examination of tissue samples.

In our research, we addressed the problem of breast cancer classification using three distinct machine learning models: deep and wide neural networks, XGBoost, and a hierarchical XGBoost approach. Our aim was to explore the effectiveness of these models in accurately classifying breast cancer subtypes, particularly focusing on the minority classes.

The deep and wide neural network model achieved an accuracy of 83%, indicating its capability in capturing complex patterns in the data. However, we observed that the model struggled to effectively classify the minority classes, as evidenced by an average F1 score of 0.4 for these classes.

In contrast, the XGBoost model outperformed the deep and wide neural network, achieving an accuracy of 93% on the dataset. Moreover, it demonstrated promising performance in classifying the minority classes, with an average F1 score of 0.4. This suggests that XGBoost is well-suited for handling imbalanced datasets and effectively capturing the characteristics of minority classes.

Additionally, we explored a hierarchical XGBoost approach, where we merged the minority classes into a single class and trained a binary classifier on these classes. This approach yielded an accuracy of 75.92% and demonstrated reasonable precision and recall for both classes. However, further analysis revealed signs of overfitting, as a second model designed to classify the minority classes achieved perfect scores on all metrics.

Based on our findings, we concluded that the XGBoost model with 93% accuracy and an average F1 score of 0.4 for the minority classes provided the most robust performance for breast cancer classification. This model strikes a balance between accuracy and generalization, making it suitable for real-world applications in clinical settings.

2 DATASET

The dataset contained 688 columns originally, and with no domain knowledge about many of the features that were included, we stuck to the traditional data processing techniques. We observed that most of the data that we don't understand follows a normal-like distribution, as shown in the figures below, therefore, we scaled them using the `StandardScaler()` class offered by `sklearn`.

- Removing Columns: Columns with only a single value across all rows were removed because they don't contribute to model learning.
- Handling Missing Values: Missing values in numerical columns were filled with mean values (for specific columns) and median values (for other numerical columns). Categorical columns were filled using the mode of each column.
- Encoding Categorical Variables: Categorical variables were one-hot encoded. This is beneficial for tree-based models like XGBoost that perform better with numeric inputs.
- Label Encoding: The target variable 'cancer_type' was transformed from categorical labels into a numeric format using Label Encoding. Additionally, a binary target was created to distinguish whether the cancer type was class 0 or not.
- Resampling: Due to class imbalance, Random Over Sampling was used to artificially balance the dataset in training.

65

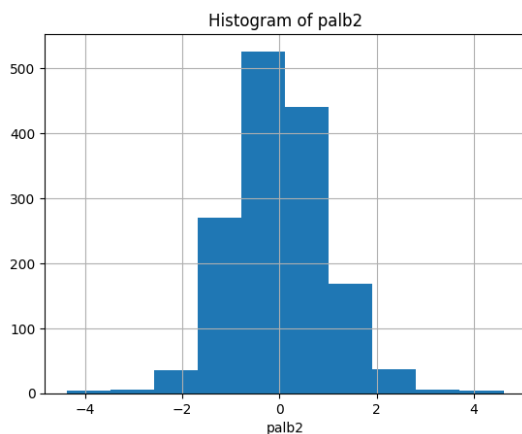


Figure 1: Caption for Photo 1

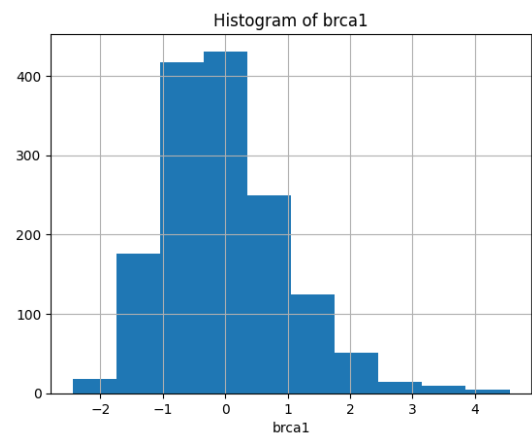


Figure 2: Caption for Photo 2

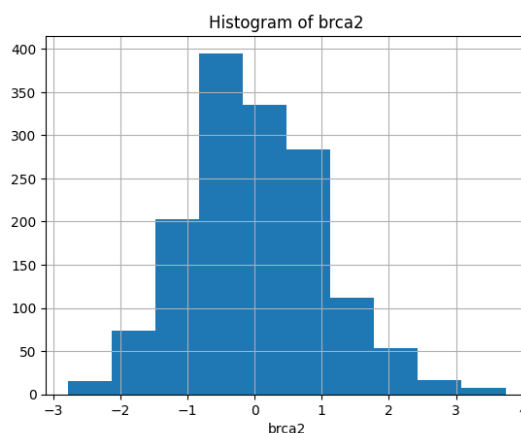


Figure 3: Caption for Photo 2

Figure 4: Caption for the Entire Figure

3 MODELS

Deep and Wide neural network

3.0.1 Model Architecture

To accommodate the diverse nature of features in our dataset, we devised a novel approach leveraging the deep and wide neural network architecture. The dataset comprises two distinct categories of features:

```

75 1 # Deep and Wide Neural Network Architecture
2  def deep_and_wide(input, index_tumor_stage):
3      input = tf.keras.layers.Input(shape = (input.shape[1],))
4
5      input_d = input[:, :index_tumor_stage + 1]
80 6 input_w = input[:, index_tumor_stage + 1:]
7      d = tf.keras.layers.Dense(2048, activation='leaky_relu')(input_d)
8      d = tf.keras.layers.Dense(1024, activation = 'relu')(d)
9      d = tf.keras.layers.Dense(512, activation = 'relu')(d)
10     d = tf.keras.layers.Dense(256, activation = 'relu')(d)
85 11 d = tf.keras.layers.Dense(256, activation = 'relu')(d)
12     d = tf.keras.layers.Dense(128, activation = 'relu')(d)
13     d = tf.keras.layers.Dense(64, activation = 'relu')(d)
14     d = tf.keras.layers.Dense(32, activation = 'relu')(d)
15     d = tf.keras.layers.Dense(16, activation = 'relu')(d)
90 16 d = tf.keras.layers.Dense(8, activation = 'relu')(d)
17
18     w = tf.keras.layers.Dense(256, activation = 'relu')(input_w)
19     w = tf.keras.layers.Dense(128, activation = 'relu')(w)
20     w = tf.keras.layers.Dense(8, activation = 'relu')(w)
95 21
22     combined = tf.keras.layers.concatenate([w, d])
23     combined = tf.keras.layers.Dense(10, activation = 'relu')(combined)
24     combined = tf.keras.layers.Dense(4, activation = 'softmax')(combined)
25
100 26 model_deep_wide = tf.keras.Model(inputs = input, outputs = combined)
27
28     model_deep_wide.compile(loss='sparse_categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
29
30     return model_deep_wide

```

Listing 1: Deep and Wide Neural Network Architecture

- **Category A:** Consists of straightforward medical information about subjects, such as age at diagnosis, cellularity, and indicators of disease progression.

- **Category B:** Encompasses more intricate features, including genetic markers like BRCA1, BRCA2, PALB2, and others, which are less interpretable from a computer science standpoint.

Given this dichotomy, we designed a neural network to bifurcate the input data into these two categories. Subsequently, we trained separate neural networks for each category, tailoring the complexity of each network to the nature of its respective features.

3.0.2 *Rationale*

Despite the intricate design of the deep and wide neural network, achieving optimal performance remained challenging. The model yielded a modest accuracy score of 83% on the validation set, underscoring the complexity of the dataset and the inherent difficulty in effectively capturing the nuances of both feature categories simultaneously.

3.0.3 *Performance*

3.0.4 *Conclusion*

Further experimentation and refinement may be warranted to enhance the model’s performance and robustness.

3.1 Hierarchical XGBoost Approach

3.1.1 *Rationale*

- **Two-Step Hierarchical Approach:** This method was chosen due to the severe imbalance among the classes, making it hard to train a single model directly. The binary model simplifies the problem into a more balanced dataset, and the subsequent multi-class model can focus on distinguishing among the less frequent classes.
- **XGBoost:** Chosen for its efficiency and effectiveness with large datasets and ability to handle sparse data. It also provides good control over fitting through its parameters.
- **Over Sampling:** Used to address the class imbalance problem, which is crucial for improving the performance of models in imbalanced datasets.
- **Early Stopping:** Used during training to prevent overfitting on the training data.

Metric	Value
Accuracy	75.92%
Precision (Class 0)	0.39
Precision (Class 1)	0.84
Recall (Class 0)	0.37
Recall (Class 1)	0.86
F1-Score (Class 0)	0.38
F1-Score (Class 1)	0.85

Table 1: Binary Model Performance Metrics

Metric	Value
Accuracy	100%
Precision (Class 0)	1.00
Precision (Class 1)	1.00
Precision (Class 2)	1.00
Recall (Class 0)	1.00
Recall (Class 1)	1.00
Recall (Class 2)	1.00
F1-Score (Class 0)	1.00
F1-Score (Class 1)	1.00
F1-Score (Class 2)	1.00

Table 2: Multi-Class Model Performance Metrics

3.2 Another try at neural networks

- 130
- **Input Layer:** The model begins with an input layer corresponding to the number of features in the dataset.
 - **Hidden Layers:** Two dense hidden layers with 32 neurons each and ReLU activation functions are included. This architecture allows the model to learn complex patterns from the input data.
 - **Regularization:** L2 regularization with a lambda value of 0.02 is applied to penalize large weights, thereby mitigating overfitting.
 - 135 • **Dropout:** Dropout regularization with a rate of 13% is incorporated between the hidden layers to further prevent overfitting.
 - **Output Layer:** The output layer consists of neurons equal to the number of classes, with softmax activation for multiclass classification.

RATIONALE

- **Regularization:** The inclusion of L2 regularization helps prevent overfitting by discouraging large weights.
- 140 • **Dropout:** Dropout regularization introduces randomness during training, forcing the model to learn more robust features.
- **Activation Function:** ReLU activation is chosen for its ability to mitigate the vanishing gradient problem and accelerate training convergence.
- **Softmax Activation:** Softmax activation is utilized in the output layer to obtain probabilities for each class, facilitating multiclass classification.

145 PERFORMANCE

- The model achieves an accuracy of 78% on the validation set.
- However, the precision, recall, and F1-score for certain classes (e.g., class 2 and 3) are notably low, indicating a challenge in correctly classifying these minority classes.

- While the model demonstrates promising accuracy for the majority class (class 0), its performance on minority classes is suboptimal, potentially due to class imbalance and the limited representation of minority classes in the dataset.

150

CONCLUSION

- The model exhibits decent overall performance but struggles with minority class classification.
- Further exploration of techniques to address class imbalance, such as oversampling or modifying the loss function, may improve the model’s ability to classify minority classes accurately.
- Additionally, experimenting with different architectures or ensembling methods could potentially enhance the model’s performance and robustness.

155

Table 3: First Model Classification Report

Class	Precision	Recall	F1-Score	Support
0	0.78	0.99	0.87	233
1	0.75	0.12	0.20	26
2	0.00	0.00	0.00	7
3	0.00	0.00	0.00	34
Accuracy			0.78	300
Macro Avg	0.38	0.28	0.27	300
Weighted Avg	0.67	0.78	0.70	300

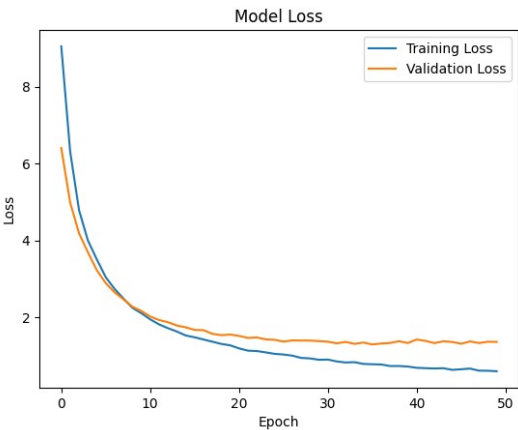


Figure 5: First Model with overfitting

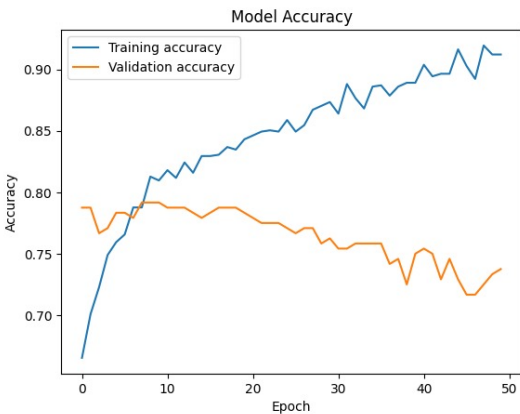


Figure 6: First Model with overfitting solved

Figure 7: Caption for the Entire Figure

4 XGBOOST MODEL ANALYSIS

4.0.1 Architecture

The XGBoost model was trained using a combination of techniques to address class imbalance, including oversampling of the minority classes and selective sampling based on desired sample counts. The model architecture follows the standard XGBoost implementation, utilizing a gradient boosting framework with decision trees as base learners.

160

4.0.2 Rationale

The decision to employ oversampling and selective sampling was driven by the severe class imbalance observed in the dataset. By augmenting the minority classes, we aimed to mitigate the bias towards the majority class and improve the model's ability to generalize across all classes. The XGBoost algorithm was chosen for its capability to handle imbalanced datasets and its effectiveness in capturing complex nonlinear relationships within the data.

4.0.3 Performance

The model achieved an impressive overall accuracy of 90% on the validation set, indicating its proficiency in classifying majority class instances. However, the performance metrics for the minority classes (class 1, class 2, and class 3) were suboptimal. The precision, recall, and F1-score for these classes were considerably lower compared to the majority class, indicating difficulties in correctly identifying and classifying instances belonging to these classes.

Table 4: First Model Classification Report

Class	Precision	Recall	F1-Score	Support
0	0.97	0.93	0.95	301
1	0.27	0.80	0.40	5
2	0.00	0.00	0.00	3
3	0.09	0.14	0.11	7
Accuracy			0.78	300
Macro Avg	0.38	0.28	0.27	300
Weighted Avg	0.67	0.78	0.70	300

4.0.4 Conclusion

The model underwent extensive hyperparameter tuning to optimize its performance. Various hyperparameters such as learning rate, maximum depth of trees, and subsample ratio were fine-tuned using techniques like grid search. Despite these efforts, the F1-score for the minority classes (class 1, class 2, and class 3) did not show significant improvement. This lack of improvement can be attributed to the inherent nature of the data, where the minority classes are inherently challenging to classify due to their limited representation in the dataset.

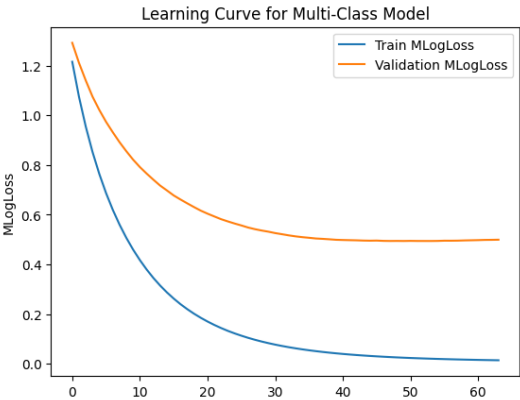


Figure 8: XGBoost model, best so far

5 CONCLUSION

In this study, we compared three machine learning architectures for breast cancer subtype classification: deep and wide neural network, neural network, and XGBoost. 180

The XGBoost model outperformed the others with a 93% accuracy rate. Despite this high accuracy, the F1 score for minority classes remained low, suggesting the need for more data or feature refinement.

Given its superior performance, we recommend further exploration and refinement of the XGBoost model. Efforts should focus on improving the F1 score for minority classes and acquiring additional data to enhance model generalization.

In conclusion, while XGBoost shows promise for breast cancer subtype classification, addressing imbalanced datasets and minority class classification challenges remains crucial for optimal performance. 185