Predicting Pathological Complete Response (PCR) and Relapse-Free Survival (RFS) in Breast Cancer Treatment

Bala Krishnan Sekar*, Tamal Chowdhury*, Arvin Corotana*, Hamza Elshafie*, Evangelos Vagianos*
*Department of Physics and Astronomy
University of Nottingham

***Abstract*—Chemotherapy is a crucial treatment method for breast cancer, effectively targeting cancer cells. However, it also brings challenges, especially due to its effects on healthy cells and the toxicities it can cause. Attaining Pathological Complete Response (pCR) is a key determinant for successful cure and extended patient lifespan. To enhance predictive capabilities for achieving pCR and estimating potential Relapse-free Survival (RFS) time, we leveraged several machine learning models. Our study utilized a dataset comprising 400 observations and 117 features, consisting of both clinical and MRI features. Logistic regression worked best on the dataset for PCR, while Random Forest performed the best for RFS. These findings contribute to the ongoing efforts in refining treatment decisions for breast cancer patients, aiming to minimize adverse effects and optimize therapeutic outcomes.**

***Index Terms*—Breast Cancer, PCR, RFS, Machine Learning**

I. Introduction

Breast cancer is a prevalent form of cancer in women, and chemotherapy is a commonly employed treatment method aimed at halting the reproduction of cancer cells. This strategy helps prevent the cells from proliferating and spreading throughout the body. While chemotherapy is a potent but toxic process, it has the potential to either completely cure the tumor or enhance the effectiveness of surgical interventions [1]. The attainment of a pathological complete response (pCR) at surgery is crucial for achieving a cure and extending the patient's lifespan. Relapse-free survival (RFS) time is the duration during which a patient survives without experiencing any symptoms or signs of cancer recurrence after the completion of primary treatment.

Unfortunately, only a quarter of individuals undergoing chemotherapy achieve pCR, as many grapple with the adverse effects of the treatment. Consequently, if we could predict the likelihood of achieving pCR before initiating chemotherapy, it would be possible to avoid undesirable outcomes and enhance treatment decision-making.

Our dataset comprises of 400 observations with a total of 117 features, including 10 clinical features. After dividing the data into training and test sets, we applied various data preprocessing techniques. Subsequently, we implemented multiple models, selecting the best-performing one for predicting outcomes in real-world scenarios.

The aims were to develop a model for each task of classification and regression. The classification task is to predict whether pathological complete response will be achieved with the given information. The regression task consists of predicting the relapse-free survival time.

II. Pre-processing

We separated pre-processing in to five different tasks. First, we checked missing values, identified in our dataset as 999. For classification, we initially noticed that only five samples were missing the pCR outcome. Given the small number, we chose to remove these from our dataset. We then examined the MRI-based features and found no missing values. However, we did find missing data in several other clinical features: ChemoGrade, Proliferation, HistologyType, LNStatus, HER2, TrippleNegative, and PgR. We imputed these missing values using the mode of each respective feature.

For regression, we noticed there were no missing values in RFS and for the other missing values, similar to classification all missing values were imputed using the mode of its respective feature column. Next, we conducted exploratory data analysis to understand the distribution of each feature and its impact on the outcome. This revealed a higher number of patients who did not achieve pCR compared to those who did. To address this imbalance and prevent model bias, we employed the Synthetic Minority Oversampling Technique (SMOTE) [2].

Furthermore, we focused on outlier detection and removal. This was carried out using the IQR method. 1.5 times of 75 percentile and 25 percentile were used as upper and lower limits respectively. Finally, we normalised the data using the min-mix scaler to transform all the feature values in the range between 0 and 1.

For feature selection, first we explored the correlations between the features and the corresponding target variable, which revealed a generally low degree of correlation between any of the features and the target. Subsequently, we then shifted our focus to explore the correlations between the features with each other, where we identified several instances of high correlation. Based on these insights, we established a criterion for feature elimination, where any feature exhibiting over three instances of a correlation exceeding 0.85 (for classification) or 0.8 (for regression) with other features was removed to reduce redundancy in our data.

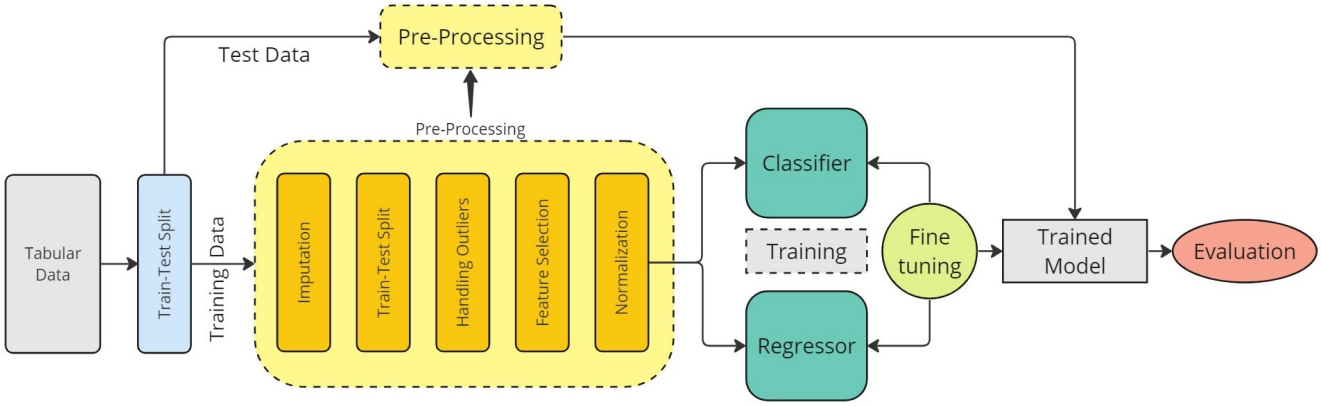

Fig. 1. Training and evaluation Pipeline

III. Method

Multiple models were developed in an attempt to capture the relationship between the given features and the label of either a pCR or not. Gridsearch was used to tune the hyper-parameters in each of the models, which tests different values of the hyper-parameters against the performance metric.

A. Classification for Pathological Complete Response

1) Logistic Regression: Logistic regression for binary classification, estimates the probability that a given sample falls into a particular category rather than providing single direct predictions [3]. It uses the Bernoulli distribution and a sigmoid function to create probabilities between 0 and 1, with a typical threshold of 0.5 to determine class allocation. The loss function is determined on the coefficient parameter vector β which is minimised using gradient descent.

2) Support Vector Machines (SVM): For classification, the SVM algorithm builds a model that assign samples to categories based on their position to some hyperplane, adjusted by some hyper-parameter C for non-perfectly separable data which relaxes the margin constraint. To handle non-linear boundaries SVM uses kernels like polynomial, linear, and RBF to extend the feature space where data can be linearly separated.

3) K Nearest Neighbours (KNN): KNN is a simple algorithm that considers the nearby data-points, expressed through number K, with respect to their distance from the respective unlabelled data. The K nearby data points have a label which is used to vote for the category that the data-point in question is, forming a majority voting system. Odd number K is common to always achieve a majority with smaller K generally resulting in noisier predictions and larger K leading to smoother decision boundaries but less capable of capturing local patterns.

4) Decision Tree: Decision Trees are a non-parametric supervised learning method, which can be used classification and regression. In Decision Trees, we make decisions and then split the variables until we find a suitable class for the split data [4]. In each decision trees, we have decision tree nodes and leaves. Leaf nodes are the final nodes of the decision tree after which, decision tree algorithm won't split the data. Decision tree algorithm tries to minimize the mean squared error between predicted values in a leaf node and the actual target values for those data points.

5) Random Forest: Random forests are an ensemble method as they combine the prediction of multiple models, decision trees, leading to better generalisation. Randomness in the form of bagging is used to train the trees on different subsets of the training data meaning the trees may fit the data they have been shown with the combination lowering the chance of over-fitting as they are not trained the same. Different predictions may be generated from tree as more randomness is introduced into the system by choosing random features to feed the trees that are not the same in training, called random feature selection. After this process, votes from the different decision trees are taken with the majority leading to the final decision [5].

6) Multi-layer Perceptron: Multi-layer Perceptron Neural Network is a neural network with multiple layers, and all its layers are connected. It learns through back-propagation and updates the parameters to make better prediction. There is an input layer, a series of hidden layers and an output layer.

B. Regression for Relapse-Free Survival

1) Lasso: Lasso regression, is in essence a linear regression model with an added $l1$ penalty term, which is equal to the sum of the absolute values of the coefficients vector. This penalty term has the effect of forcing some of the coefficients to zero when the tuning parameter λ is sufficiently large, which essentially performs feature selection. Predictions are then made by using the retained features in a linear combination similar to linear regression and the model is optimized for MAE.

2) Decision Tree: In the regression context, the fundamental structure of Decision Trees, which includes decision nodes

and leaves, remains the same as that in classification. However, the objective differs for regression, where the algorithm instead focuses on minimizing the mean squared error (MAE) between the predicted values in a leaf and the actual target values of the data points within that leaf. Predictions are made by traversing the tree with a new data point's features until a leaf node is reached, where the prediction is the mean of the target values of the training samples in that leaf.

3) Random Forest: Random forest for regression is much the same compared to classification with the basis of predictions being from an ensemble of decision trees. Instead of a majority voting system after the bagging and random feature selection there will be an averaging of the predictions from all decision trees.

4) XGBoost: XGBoost, standing for extreme Gradient Boosting, leverages gradient boosting to sequentially refine its models, focusing on correcting previous errors, thus improving accuracy. It uses a unique tree-building technique by growing trees to a certain maximum depth specified before pruning, which helps in understanding patterns in the data. For regression, XGBoost aggregates predictions from each tree, with individual contributions based on input features, leading to a precise final prediction that reflects the overall learning of the model's trees.

5) MLP: Multi-layer Perceptrons, structured with input, hidden, and output layers, use neurons with nonlinear activation functions to model complex data relationships. For regression, they predict continuous output values by processing input data through these layers. During training, they employ backpropagation and gradient descent to minimize the difference between actual and predicted values, refining the network's weights.

IV. Training and Evaluation

For both the classification and regression problem, evaluation of the models was performed using K-fold cross-validation method which splits the training set into K folds, 5 in our case, where at each training iteration 4 folds will be used to train the model and 1 fold will be used to validate the model performance. This is repeated K times as all combinations are explored with the validation data being different for each iteration producing K results for validation using slightly different training data and a different validation set each time boosting confidence when the dataset is smaller. For hyperparameter tuning, we used gridsearch algorithm to pick the optimal parameters from a given set of parameter space.

To evaluate our model performance we used Balanced accuracy given by equation 1 for the classification task which is much more robust to class imbalance compared to normal accuracy.

$$\text{Balanced Accuracy} = \frac{1}{2}\left(\frac{\text{TP}}{\text{TP+FN}} + \frac{\text{TN}}{\text{TN+FP}}\right) \quad (1)$$

where, TP: True Positive, FN: False negative, TN: True negative and FP: False positive.

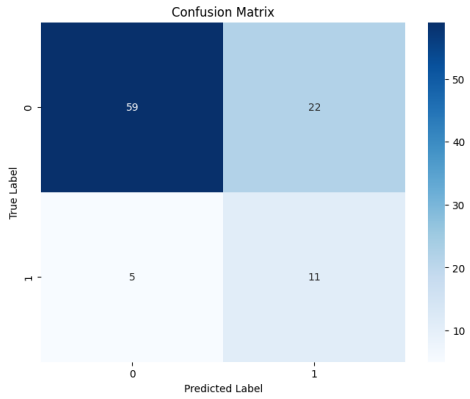

Fig. 2. Confusion matrix of Logistic Regression on test data

For the regression task we used the Mean Absolute Error (MAE) given by equation 2 as the evaluation metric, which is robust to outliers and noise in the data.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\text{Actual}_i - \text{Predicted}_i| \quad (2)$$

V. Results

A. Classification

Performance of the classification models on different folds during training is shown in Table I. Table II displays the performance on the test data. The table shows that Logistic Regression performs best on the test data in terms of balanced classification accuracy. The optimal set of parameters found for the Logistic Regression are:

C:1, penalty: L1, solver: liblinear

TABLE I
Balanced Classification Accuracy Scores in K-Fold Cross-Validation for Classification

	Fold					
Model	1	2	3	4	5	Mean
Logistic Regression	69.57	72.83	67.39	58.20	70.41	67.68
SVM	69.57	68.48	68.48	62.56	65.89	67.00
Decision Tree	71.04	72.9	69.56	74.99	73.60	72.26
Random Forest	80.65	84.35	89.13	83.33	87.40	84.97
KNN	70.65	71.74	67.39	65.77	65.92	68.29
MLP	54.35	65.22	57.61	61.26	68.14	61.32

TABLE II
Balanced Classification Accuracy Scores in Testing for Classification

Model	Accuracy (%)
Logistic Regression	**70.79**
SVM	62.11
Decision Tree	52.66
Random Forest	58.80
KNN	52.70
MLP	46.22

The confusion matrix of the logistic regression model on the test set is shown in figure 2

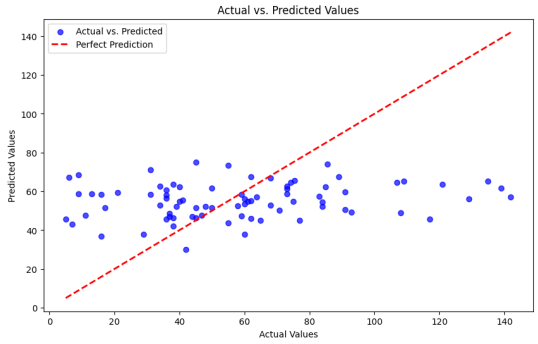

Fig. 3. Actual vs Predicted (Random Forest)

B. Regression

Similarly, the performance of our regression models on different folds during training is shown in Table III. Table IV shows the performance on the test data. From the table, it is evident that Random Forest performs best on the test data in terms of mean absolute error. The optimal set of parameters found for the random forest model are:

max depth:20, min samples leaf:1, min samples split:2, n estimators:50

TABLE III
MEAN ABSOLUTE ERROR SCORES IN K-FOLD CROSS-VALIDATION FOR REGRESSION

	Fold					
Model	1	2	3	4	5	Mean
Lasso	21.32	19.97	20.05	19.57	19.59	20.10
Decision Tree	0.04	0.05	0.05	0.03	0.03	0.04
Random Forest	7.52	7.63	7.58	7.62	7.70	7.61
XGBoost	3.24	3.29	3.26	3.27	3.29	3.27
MLP	11.36	11.30	11.32	11.35	11.37	11.34

TABLE IV
MEAN ABSOLUTE ERROR SCORES IN TESTING FOR REGRESSION

Model	Mean Error
Lasso	25.56
Decision tree	37.41
Random Forest	**24.82**
XGBoost	27.33
MLP	26.62

The graph of actual vs predicted values on the test set for the Random Forest model is shown in figure 3.

VI. DISCUSSION

The outcomes from training evaluation along side the testing results led to the decision to move forward with the models logistic regression for classification and Random Forest for regression.

Logistic regression generally is a technique that is valued for its simplicity, making it easy to implement and even interpret the results afterwards. It best suited for datasets where there is a linear relationship between the features and the outcomes. On the other hand, however, it struggles with complex, non-linear data relationships, plus it also can be sensitive to imbalanced datasets, potentially leading to poorer model performance. Hence, why oversampling was necessary for this project.

Meanwhile, Random forests are well-known for their robustness against outliers and their ability to handle non-linear data. Yet, they come with the drawback of being computationally intensive and less interpretable as predictions are constructed from not one, but multiple decision tress, which can be challenging in scenarios where understanding the model's decision process is crucial.

Improvements could be made be trying to develop more ensemble methods for classification like XGBoost as random forest was the top performers suggesting the ensemble method may be more robust to outliers and noise as a result of imbalanced datasets. Regression may also benefit from trying to develop other models and a more extensive hyperparameter tuning method may be beneficial. In general, different methods of feature selection could have been tested, utilising statistical tests or wrapper methods. Collecting more data within this context through clinical trials would aid in the development of machine learning models as the data will help establishing relationships between features and labels. Testing for other characteristics could also be beneficial, potentially discovering other features that have more predictive power for pCR or RFS.

VII. CONCLUSIONS

In conclusion, our study aimed to enhance the efficacy of chemotherapy for individuals with a high likelihood of success by developing machine learning models to predict the surgical outcome and post-surgery survival period based on individual characteristics. Through a comprehensive process involving rigorous analysis, preprocessing, feature selection, hyperparameter tuning, and model training, we identified logistic regression and random forest models with optimized parameters as effective tools for predicting the pathological complete response (PCR) outcome and relapse-free survival (RFS).

Despite the challenges posed by a small dataset with a large number of features, our models achieved a commendable performance, with a balanced accuracy of 71% and a mean absolute error (MAE) of 24.82. It is essential to note that, in the context of our study, these results are reasonable and promising. The complexity of the dataset necessitated a careful consideration of model selection, feature relevance, and hyperparameter optimization. Our findings underscore the potential of machine learning in aiding clinical decision-making processes related to chemotherapy.

Our study contributes to the growing body of work leveraging machine learning in personalized medicine, offering insights that can potentially improve treatment outcomes for breast cancer patients. As we continue to advance in the era of precision medicine, the integration of machine learning approaches holds promise for optimizing therapeutic strategies tailored to individual patient profiles.

REFERENCES

[1] Das, A., Biswas, S., Bhattacharya, A. and Alam, E. (2021). Introduction to Breast Cancer and Awareness. doi:https://doi.org/10.1109/icaccs51430.2021.9441686.

[2] D. Elreedy and A. F. Atiya, "A comprehensive analysis of synthetic minority oversampling technique (smote) for handling class imbalance," *Information Sciences*, vol. 505, pp. 32–64, 2019. doi:10.1016/j.ins.2019.07.070

[3] J. C. Stoltzfus, "Logistic regression: A brief primer," *Academic Emergency Medicine*, vol. 18, no. 10, pp. 1099–1104, 2011. doi:10.1111/j.1553-2712.2011.01185.x

[4] Smitha. T and V. Sundaram, "Classification rules by decision tree for disease prediction," *International Journal of Computer Applications*, vol. 43, no. 8, pp. 6–12, 2012. doi:10.5120/6121-8323

[5] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. doi:10.1023/a:1010933404324