Predictive Modeling for Breast Cancer Survivability Using Machine Learning Techniques

Rahi Krishna
Dept. of Health Informatics
New York University
rk4748@nyu.edu

Tanmay G Dadhania
Dept. of Computer Science
New York University
tgd8275@nyu.edu

Tarun Sharma
Dept. of Computer Science
New York University
ts5098@nyu.edu

***Abstract*—The research investigates the application of various machine learning algorithms to predict breast cancer survivability using data from the Surveillance Epidemiology and End Results (SEER) database. The study emphasizes preprocessing, feature importance analysis, and model performance through rigorous hyperparameter tuning. Key findings indicate that hormonal status and cancer staging are vital predictors of survivability. The Random Forest model exhibited superior performance, achieving an accuracy of 89.06%. This paper outlines the methodology, experiments, and results of the study, providing insights into the effectiveness of different algorithms in a clinical predictive context.**

***Index Terms*—breast cancer, machine learning, predictive modeling, feature importance, SEER database**

I. Introduction

Breast cancer remains one of the most prevalent and deadly cancers affecting women worldwide. Despite advances in medical technology and treatment methodologies, the prognosis for breast cancer patients varies widely based on numerous factors, including genetic, demographic, and clinical characteristics. Predictive modeling using machine learning offers a promising approach to understanding these outcomes by analyzing large datasets to identify patterns and predictors of survivability.

The Surveillance Epidemiology and End Results (SEER) program collects and publishes cancer incidence and survival data from population-based cancer registries covering approximately 34.6% of the U.S. population. This dataset is rich with detailed information about patient demographics, tumor characteristics, treatment details, and survival outcomes, making it an invaluable resource for clinical research, particularly in oncology.

In this study, we leverage the SEER database to develop a predictive model that classifies breast cancer patients into different survivability outcomes based on their clinical and demographic features. The primary objective is to apply advanced data mining techniques, including feature selection and machine learning algorithms, to build a model that can accurately predict the likelihood of survival at various stages following a breast cancer diagnosis.

II. Materials and Methods

A. Data Source

Data for this study was sourced from the SEER database, maintained by the National Cancer Institute. It includes comprehensive patient data, which was preprocessed to handle missing values, standardize numerical inputs, and reduce dimensionality using PCA, preparing it for further analysis.

B. Data Preprocessing

The preprocessing steps were implemented using Python libraries such as Pandas and Scikit-learn. The dataset was first loaded from a Google Drive location, showcasing the integration of cloud storage in data handling. The initial preprocessing involved:

- Identifying and classifying survival status based on the criteria that if survival months were greater than or equal to 60 and the patient was alive, they were classified as 'survived'.
- Handling missing values by imputing the median for numerical features and the mode for categorical features.
- Applying *StandardScaler* for normalization and *PCA* for dimensionality reduction to retain 95% of the variance in the dataset, which was crucial for managing high-dimensional data.

C. Feature Encoding and Selection

Categorical variables were encoded using one-hot encoding to prepare them for machine learning algorithms. Feature selection was performed based on information gain, calculated using mutual information criteria, to identify the most predictive features. This involved:

$$IG(T, a) = H(T) - H(T \mid a) \tag{1}$$

where $H(T)$ is the entropy of the target variable and $H(T \mid a)$ is the conditional entropy given feature a.

D. Model Development and Evaluation

Each model was rigorously tested and evaluated based on its ability to accurately classify breast cancer survivability outcomes. Below, we delve into the specifics of each model's implementation and mathematical foundations.

1) K-Nearest Neighbors (KNN): The K-Nearest Neighbors algorithm is a non-parametric method used for classification by analyzing the labels of the nearest samples in the feature space. The KNN algorithm implemented can be mathematically described as follows:

$$y(x) = \frac{1}{k} \sum_{i \in N_k(x)} y_i \tag{2}$$

where $N_k(x)$ denotes the set of k nearest neighbors to point x, and y_i are the labels of these neighbors. Distance between points is calculated using the Euclidean metric:

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{n} (x_{il} - x_{jl})^2} \tag{3}$$

where x_i and x_j are two points in the feature space, and n is the number of features.

2) Naïve Bayes: The Naïve Bayes classifier is based on Bayes' theorem with an assumption of independence among predictors. The model is particularly effective for large datasets and can be described using:

$$P(y|x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n|y)}{P(x_1, \ldots, x_n)} \tag{4}$$

Assuming independence, it simplifies to:

$$P(y|x_1, \ldots, x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i|y)}{P(x_1, \ldots, x_n)} \tag{5}$$

where y is the class variable, and $x_1, \ldots, x_n$ are feature variables.

3) Decision Trees: Decision Trees are a non-linear predictive modeling approach used extensively in classification tasks. The decision to split at each node is determined by the Gini index or entropy, aiming to maximize the information gain:

$$IG(D_p, f) = I(D_p) - \sum_{j=1}^{m} \frac{N_j}{N} I(D_j) \tag{6}$$

where IG is the information gain, D_p and D_j are the datasets of the parent and j-th child node, N and N_j are the total number of samples in the parent and j-th child node, and I is the impurity measure.

4) Random Forest: Random Forest builds multiple decision trees and merges them together to improve the model's accuracy and control over-fitting. The output of the Random Forest is defined as the averaged predictions of all individual trees:

$$f(x) = \frac{1}{B} \sum_{b=1}^{B} f_b(x) \tag{7}$$

where B is the number of trees, and f_b is the prediction of the b-th tree.

5) Gradient Boosting: Gradient Boosting is an ensemble technique that builds models sequentially, each new model correcting errors made by previous models. The model is built by fitting each new instance to minimize the loss function, given by:

$$L(y, F(x)) = \sum_{i=1}^{n} L(y_i, F(x_i)) \tag{8}$$

where L is the loss function, $F(x)$ is the model prediction, and y is the actual value. Gradient boosting specifically adjusts the weights of incorrectly classified instances so that subsequent classifiers focus more on difficult cases.

These models were evaluated using cross-validation techniques to ensure the robustness of the model performance metrics, specifically looking at accuracy, precision, recall, and F1-score to assess the efficacy of each model in the context of the dataset used.

E. Hyperparameter Tuning

Hyperparameter tuning was conducted using Grid Search to find the optimal settings for each model. The process is mathematically represented as:

$$\hat{\theta} = \arg\min_{\theta} CV(\theta) \tag{9}$$

where θ represents hyperparameters and $CV(\theta)$ represents the cross-validated accuracy.

F. Hyperparameter Tuning

Hyperparameter tuning is essential for optimizing machine learning models, and in this study, Grid Search CV was used to fine-tune the parameters of Random Forest and Gradient Boosting models.

1) Grid Search CV: Grid Search CV methodically explores a specified subset of hyperparameters, conducting k-fold cross-validation for each combination to assess performance. The primary steps include:

1) **Defining Parameter Grids:** For Random Forest and Gradient Boosting, grids were set up with ranges for parameters like the number of estimators, tree depth, and learning rate.
2) **Cross-Validation:** Each parameter combination undergoes k-fold cross-validation to ensure robust evaluation, using accuracy as the scoring metric.
3) **Optimal Parameter Selection:** The combination yielding the best cross-validation score is chosen as the optimal set for the model.

The process is mathematically represented as:

$$\hat{\theta} = \arg\min_{\theta \in \Theta} (-\mathrm{CV}(\theta)) \tag{10}$$

where Θ denotes all parameter combinations, θ a specific combination, and CV is the cross-validation function returning the score for θ.

This approach ensures that the models are not only tailored to the specific characteristics of the dataset but also achieve maximum predictive accuracy.

III. Results

A. Feature Importance Analysis

An essential aspect of the modeling process was identifying which features contributed most significantly to the prediction of breast cancer survivability. The importance of each feature was evaluated using Information Gain (IG), which is rooted in the concept of entropy in information theory. This metric provides insight into how much each feature decreases the uncertainty about the target variable. Features with higher IG values contribute more to the model's predictive power.

The top five features according to their Information Gain were as follows:

1) Feature '13': InfoGain = 0.029637
2) Progesterone Status_Positive: InfoGain = 0.027755
3) Progesterone Status_Negative: InfoGain = 0.026914
4) Estrogen Status_Positive: InfoGain = 0.021380
5) 6th Stage_IIIC: InfoGain = 0.020382

These features highlight the critical role of hormonal status (specifically progesterone and estrogen) in the survivability prognosis of breast cancer patients. The presence or absence of hormonal receptors can significantly influence treatment decisions and outcomes. Furthermore, the 6th Stage_IIIC feature's importance indicates the model's sensitivity to advanced cancer stages, which are clinically known to correlate with prognosis.

B. Hyperparameter Tuning Results

Hyperparameter tuning is an integral part of model optimization, ensuring that the models perform at their best. The Grid Search CV approach was utilized to systematically explore the hyperparameter space. For Random Forest, a comprehensive search across various depths of trees, numbers of estimators, and other parameters yielded the following optimal configuration:

- Maximum Depth: 10
- Minimum Samples per Leaf: 2
- Minimum Samples Split: 5
- Number of Estimators: 200

With these parameters, Random Forest achieved a cross-validated accuracy of 89.06%. Similarly, Gradient Boosting was fine-tuned, resulting in a slightly different set of optimal parameters:

- Learning Rate: 0.01
- Maximum Depth: 3
- Number of Estimators: 200

Gradient Boosting reached a cross-validated accuracy of 88.63%. These results not only highlight the efficacy of both models but also the importance of careful hyperparameter selection. The fine-tuning process has provided us with valuable insights into the models' behavior and their sensitivity to specific hyperparameters. It is worth noting that both models benefited from a larger number of estimators, which suggests that the dataset required complex models to capture the underlying patterns effectively.

C. Model Performance Summary

The performance of each model was evaluated based on its classification accuracy on the test set. The following table summarizes the accuracy achieved by each model after hyperparameter tuning:

TABLE I
Model Performance Summary

Model	Accuracy
KNN	0.875776
Naive Bayes	0.877019
Decision Tree	0.843478
Random Forest	0.895652
Gradient Boosting	0.889441

Random Forest emerged as the top-performing model with an accuracy of approximately 89.57%, closely followed by Gradient Boosting with an accuracy of 88.94%. These results showcase the effectiveness of ensemble methods in managing the complexity and nuances of the dataset. KNN and Naive Bayes also performed commendably, achieving accuracies above 87%, which underscores their utility in scenarios where model interpretability and computational efficiency are key considerations. The Decision Tree model demonstrated the lowest accuracy, which can be attributed to its tendency to overfit, especially in the absence of extensive parameter tuning or pruning strategies.

These findings suggest that while simpler models can provide a baseline for prediction, ensemble models that combine multiple learning algorithms significantly enhance prediction accuracy, a valuable insight for developing robust predictive tools in the medical domain.

IV. Conclusion

The study's comprehensive analysis using machine learning models has provided valuable insights into breast cancer survivability. Random Forest, with its accuracy of 89.57%, stands out as the most potent model, closely followed by Gradient Boosting. The ensemble methods demonstrated their superiority in managing the dataset's complexities, outperforming simpler models such as KNN and Naive Bayes. Notably, feature importance analysis revealed hormonal status and advanced cancer stages as critical predictors of patient outcomes. The research confirms the efficacy of machine learning techniques in medical prognosis and underscores the importance of domain expertise in interpreting model outcomes. The synergy between data-driven models and clinical knowledge holds the promise to revolutionize personalized patient care. As future work, the incorporation of additional patient data, such as genetic markers and lifestyle factors, could be explored to further enhance the model's predictive power.

References

[1] J. Brownlee, "Hyperparameter optimization with random search and grid search," Python Machine Learning, 2020, available: https://machinelearningmastery.com/.

[2] K. Taunk, S. De, S. Verma, and A. Swetapadma, "A brief review of nearest neighbor algorithm for learning and classification," *IEEE Transactions on Knowledge and Data Engineering*, 2020, available: IEEE Xplore.

[3] A. Bari, M. Chaouchi, and T. Jung, "How to use predictive analysis decision trees to predict the future," 2020, available online.

[4] R. Meltzer, "What is random forest?" 2023, updated on August 31.

[5] T. Masui, "All you need to know about gradient boosting algorithm – part 1. regression," *Towards Data Science*, January 20 2022, available: https://towardsdatascience.com/.

[6] K. Menon, "Feature selection in machine learning: All you need to know," 2019, lesson 7 of 38, Available online.

[7] R. Agarwal, "The 5 feature selection algorithms every data scientist should know," July 27 2019, available: https://towardsdatascience.com/.

[8] V. Kesti, "Ranking features based on predictive power/importance of the class labels," September 15 2019, available: https://www.analyticsvidhya.com/.

[9] M. Ved, "Feature selection in machine learning: Variable ranking and feature subset selection methods," 2019, available online.