Predicting Cancer Recurrence with Cellular, Tumor, and Patient Characteristics:

Prognostic Wisconsin Breast Cancer Database

Rachel Rowey
M.Sc. Candidate in Biomedical Engineering, Brown University
GitHub Repository: https://github.com/rachelrowey/Rowey-DATA1030-Project.git
Submitted: December 13, 2024

DATA1030: Hands-On Data Science
Instructor: Andras Zsom, PhD
Brown University Data Science Institute
Fall 2024

I. Introduction

Motivation

Cancer is a widely studied and complex disease affecting 20 million people globally each year.[1] Among its subtypes, breast cancer is one of the most prevalent, impacting 1 in 8 women in the U.S.[2] Recovery and remission rates vary significantly by stage, with five-year survival rates ranging from 90-98% for stage 2 breast cancer to just 31% for stage 4.[3,4] Despite advancements in treatment and improved survival rates, tumor recurrence remains a critical challenge. Recurrence rates for ER-positive tumors, axillary lymph node involvement, and triple-negative breast cancer are approximately 42%, 25%, and 40%, respectively.[5] These high rates emphasize the need for accurate prediction methods to detect recurrence early, enabling better treatment planning and outcomes. The ML model in this paper aims to accurately predict recurrence risk in breast cancer patients based on cellular, tumor, and patient characteristics.

Dataset

The dataset for this project is the Wisconsin Prognostic Breast Cancer (WPBC) database, available in the UC Irvine Machine Learning Repository.6 It contains follow-up data from breast cancer patients treated at the University of Wisconsin General Surgery Department since 1984. The cohort includes only cases of invasive breast cancer without evidence of distant metastases at diagnosis, simplifying the analysis to focus on local recurrence. Features were computed from digitized images of fine needle aspirates of breast masses, describing various cellular nuclei characteristics. Additional features include patient information, such as the number of affected lymph nodes, tumor diameter, and time to excision.

Previous Work

Previous research on this dataset primarily focused on predicting the time to recurrence using Recurrence Surface Approximation (RSA), a novel linear programming approach. RSA incorporates both recurrent and nonrecurrent cases to predict recurrence timing, achieving a mean error of 13.9 months. This method also assessed the predictive power of various features, showing that cytological measurements from FNA images were more effective prognostic indicators than traditional factors like tumor size or lymph node status. The features were analyzed and consolidated using the Xcyt software system, enabling the modeling of recurrence as a function-approximation problem.

II. Exploratory Data Analysis (EDA)

Overview

This dataset is small with 197 data points, each representing an individual breast cancer patient. This data is therefore independent and identically distributed (IID) since there is no grouping. It also contains 33 features, 32 of which are continuous and 1 is ordinal. The columns first include ID number, time, lymph node status, and tumor size. ID number is omitted as one of

the features. Additionally, there are ten cell characteristics of interest, with three cell nuclei imaged per patient, yielding 30 features. These include radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension. The target variable is outcome, a binary, categorical variable that either has a value of R (recur), or N (nonrecur). There are 150 instances of N and 47 instances of R, making this a slightly imbalanced dataset, as seen in **Figure 1**.

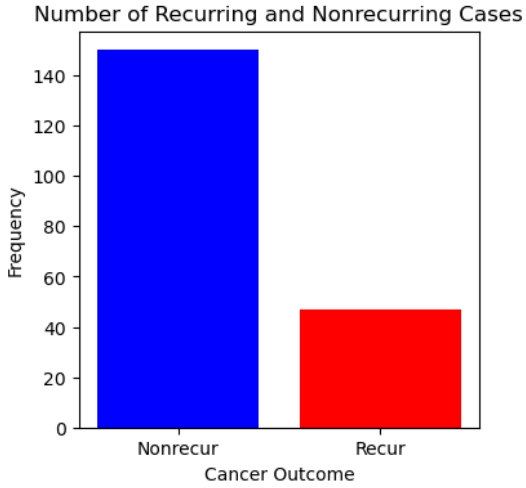
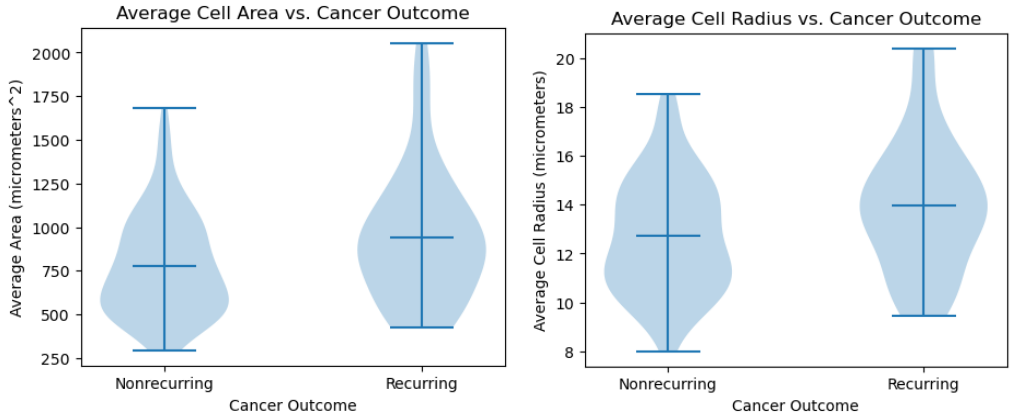


Figure 1. Case distribution of the target variable.

Feature Visualizations

After the initial overview, I created visualizations between the features and target variable to identify correlations. The visuals seen in **Figure 2** illustrate relationships between cell characteristics and the target variable, showing slight correlations. The area, radius, and perimeter means for nonrecurring cases and recurring cases are 775.31 micrometers^2, 12.71 micrometers, 84.47 micrometers, and 941.45 micrometers^2, 13.95 micrometers, 92.92 micrometers, respectively.

a) b)



c)

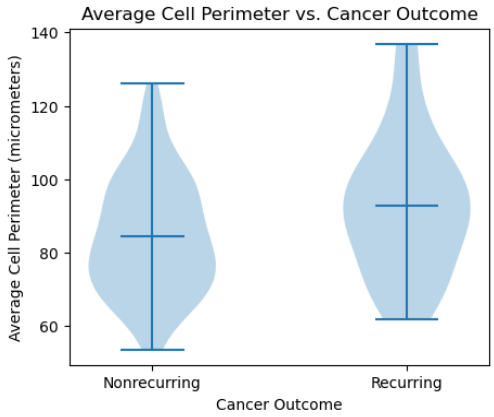


Figure 2. Violin plots of average cell a) area, b) radius, and c) perimeter vs. cancer outcome.

Additionally, **Figure 3** displays the correlation between tumor size and cancer outcome. The range of values is similar for each class, but the excised tumor diameter is slightly larger for recurring cases, with a mean of 3.46 cm, in contrast to 2.64 cm for nonrecurring cases.

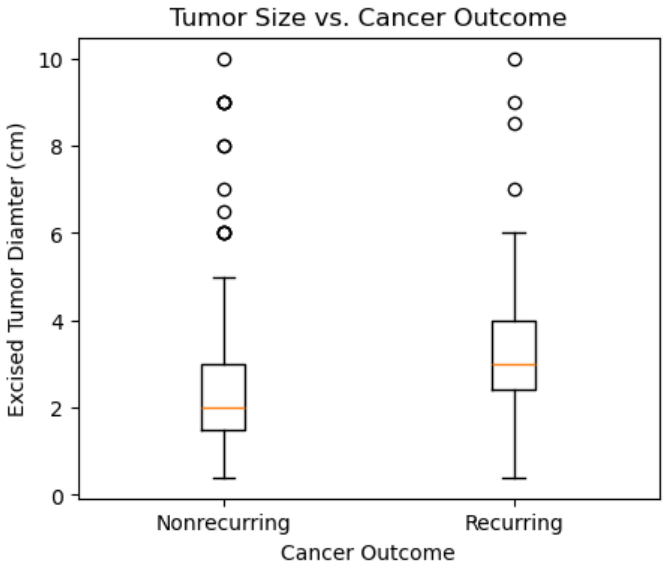


Figure 3. Box plot of excised tumor diameter vs. cancer outcome.

Finally, **Figure 4** displays lymph node status in correlation with cancer outcome. I expected to see the proportion of recurring cases increase as the affected lymph nodes increase, but surprisingly there is no clear trend observed here.

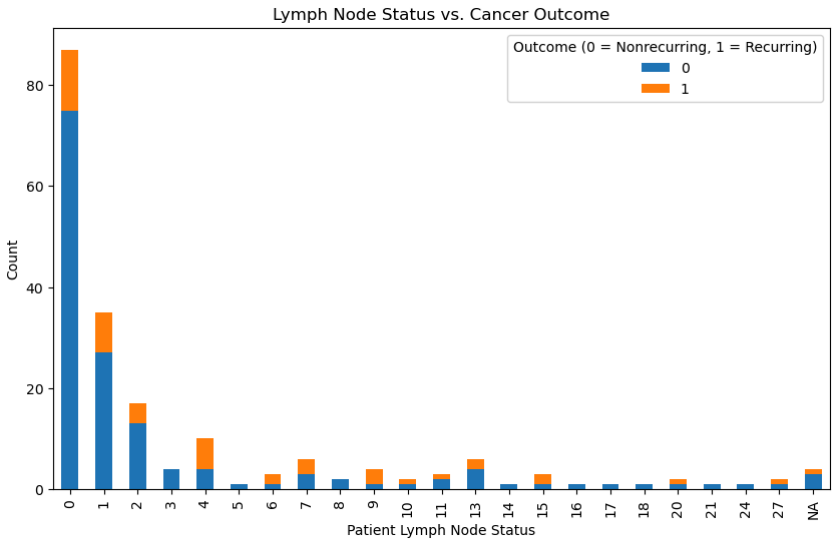


Figure 4. Stacked bar graph of lymph node status vs. cancer outcome.

These visualizations allow for a comprehensive understanding of the dataset before moving forward with further analysis.

III. Methods

Splitting Strategy

This dataset is small, IID, and slightly imbalanced, and is therefore split using a stratified KFold split. The first step is to use train_test_split() to split the data into 20% test and 80% 'other', while stratifying for the outcome variable, y. Next, StratifiedKFold is applied to 'other' to create multiple folds that act as training and validation data, stratifying for y. Since this dataset is small, we use 5 splits to maximize the data and allow for training and cross-validation on multiple different subsets. Stratification allows for sufficient representation of the target variable classes in each iteration.

Data Preprocessing

The first preprocessing step was feature engineering. Each of the ten cell characteristics had three repeat measurements, so I created ten new features by averaging the repeats, allowing for a more straightforward analysis of each feature's effect on recurrence, and resulting in 43 total features. The next step was to handle missing values present in the lymph node status column. I replaced the question marks with 'NA' and created an 'NA' category for the lymph node status feature. Lymph node status is categorical, specifically ordinal, data and therefore was encoded using OrdinalEncoder(). Finally, the numerical features were standardized using StandardScaler() to ensure consistent means of zero and standard deviations of one. This ensures that all variables are on a comparable scale for accurate analysis.

Cross Validation Pipeline & Hyperparameter Tuning

The cross validation pipeline is created with both a preprocessor and a classifier. The preprocessor accounts for categorical and numerical features, as discussed previously. Four classifiers were used, including logistic regression, random forest, SVC, and XGBoost classifier. A parameter grid is defined for the classifier, containing the hyperparameters being tuned and their values. A summary of algorithms, hyperparameters, and their values can be seen in **Table 1.** The model is evaluated across multiple random states, in this case 5 random states, to account for variability in splitting. Within each random state, the splitting strategy described previously is implemented. Then, GridSearchCV() uses the pipeline and parameter grid to tune the hyperparameters. Ultimately, this pipeline collects the best hyperparameter values, CV scores, and test scores.

Model Performance Metrics & Uncertainty

To evaluate model performance, the primary metric used was accuracy, which measures the proportion of correct predictions to the number of total predictions. While there is potential for differing viewpoints, I chose to assign equal weight towards false positives and negatives. This ML model helps individuals plan treatments, and it is unclear exactly what that might entail. A false positive could lead an individual to receive excess cancer treatment, which can be detrimental to other health aspects, while a false negative could lead to catching the tumor later than intended. However, it is standard for individuals in remission to undergo regular screening and therefore unlikely for the tumor to progress past stage 2 without detection. Therefore, placing equal weight on these two cases seems to be the most logical approach. It is notable that accuracy can be misleading for a slightly imbalanced dataset such as this one, which is why baseline score analysis is essential. I also calculated F2 scores for a more comprehensive look, which inherently takes imbalance into account and allowed me to explore the scenario of prioritizing recall in the case that a patient wanted to prioritize a false negative. Ultimately, the accuracy and F2 scores both agreed on an optimal model.

Additionally, the scores' standard deviations across various random states were calculated in order to account for splitting uncertainties, seen in **Table 1**.

IV. Results

Baseline Score Comparison

To understand the significance of the test scores, we must establish a baseline score. To calculate the baseline accuracy score, we use a majority class predictor which predicts all data points as the majority class, which is class 0 (nonrecur), with 150 instances. We know TN=150, FN=47, and FP=TP=0 and use the equation accuracy = (TN + TP) / total samples, yielding a baseline accuracy of 0.761. Next, we calculate the baseline F2 score using a minority class predictor, since a majority class predictor would be undefined, which predicts class 1 (recur) for all datapoints. Class 1 has 47 instances, so we know TN=FN=0, FP=150, and TP=47 and use the

equation F2-score = (1+beta^2)*(PR/((beta^2*P) + R)), where beta=2, yielding a baseline F2 score of 0.61.

ML Model Performances & Optimal Model

The logistic regression model showed the best mean accuracy score of 0.8050, and was therefore selected as the best model, as seen in **Table 1.** Its optimal parameters were C=1, penalty=L1, and solver=saga. This model also had the best F2-score of 0.5870. It is notable that the accuracy score was 1.51 standard deviations above the baseline, while the F2-score was 0.54 standard deviations below the baseline.

Table 1: Summary of ML models, hyperparameters, and performance metrics.

Model	Tuned Parameters	Possible Parameter Values	Optimal Values	Accuracy Score (+/- std. dev.)	F2-Score (+/- std. dev.)
Logistic Regression	C Penalty Solver	[0.001, 0.01, 0.1, 1, 10] [L1, L2] [Liblinear, saga]	1 L1 saga	0.8050 +/- 0.0292	0.5870 +/- 0.0425
Random Forest	Max_depth Max_features	[1, 3, 30, 10, 30, 100] [0.25, 0.5, 0.75, 1.0]	3 0.5	0.7650 +/- 0.0122	0.1838 +/- 0.0820
SVC	C Kernel Gamma	[0.001, 0.01, 0.1, 1, 10] [linear, rbf] [scale, auto]	1 Linear Scale	0.7600 +/- 0.0339	0.2634 +/- 0.1480
XGBoost Classifier	Max_depth Reg_alpha Reg_lambda	[1, 3, 10, 30, 100] [0.01, 0.1, 1, 10, 100] [0.01, 0.1, 1, 10, 100]	3 0.1 1	0.7750 +/- 0.0474	0.3292 +/- 0.1490

Global & Local Feature Importance

The first method used to calculate global feature importance was coefficient magnitude. Considering all of the variables are scaled, we are able to directly correlate the coefficient magnitude with feature importance, resulting in **Figure 5**. Next, permutation importance was used, which determines global feature importance based on how much the model's performance decreases when a feature is shuffled. The results can be seen in **Figure 6**. Finally, SHAP values were used to assess the contribution of a feature across all data points for global importance, as seen in **Figure 7**, as well as at an individual prediction for local importance, as seen in **Figure 8**.

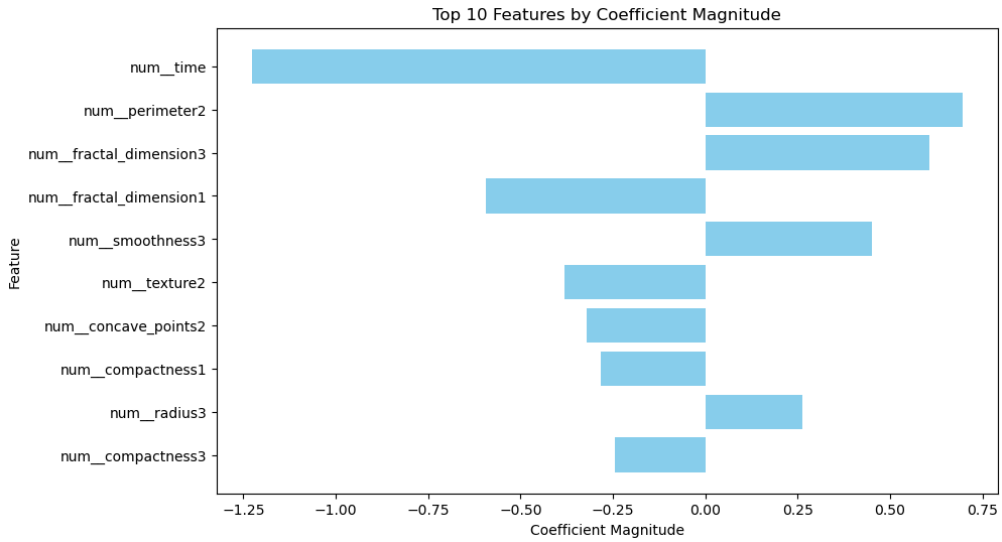


Figure 5. Global feature importance using coefficient magnitude.

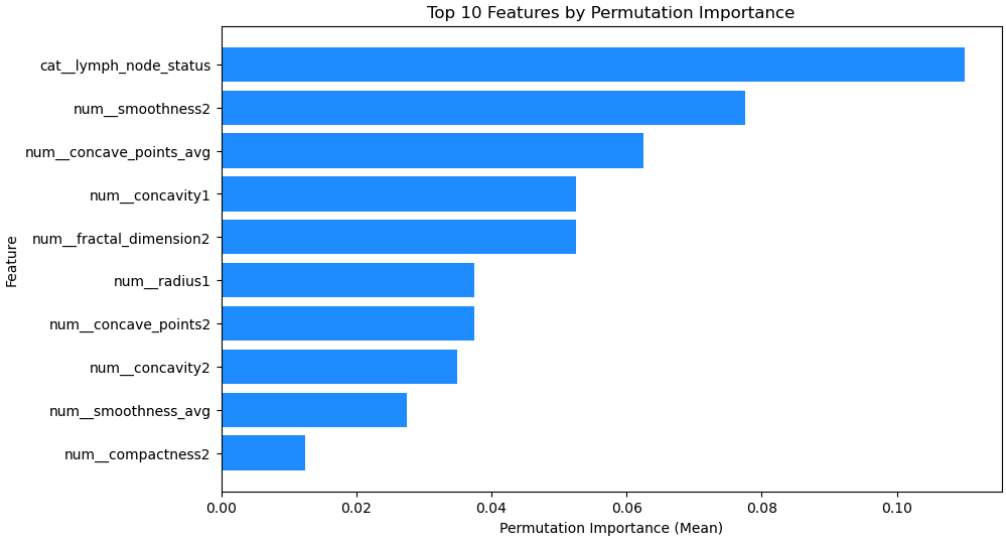


Figure 6. Global feature importance using permutation importance.

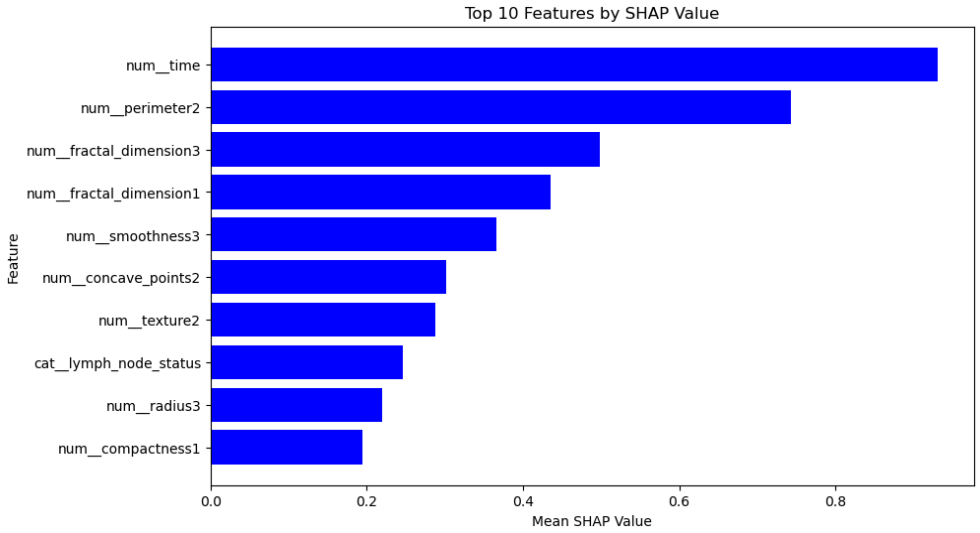


Figure 7. Global feature importance using SHAP values.

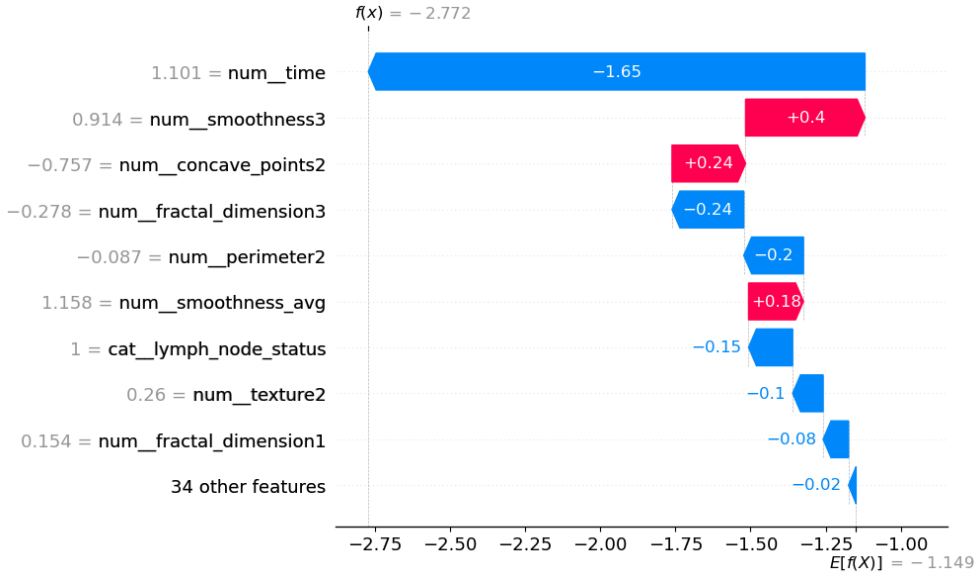


Figure 8. Local feature importance using SHAP values.

Interpretation

We synthesize the results found through global and local feature importance analysis to understand how each of the features impact recurrence risk. It is notable that perimeter, fractal dimension, smoothness, time, and concave points all appear in the top 5 features for multiple importance calculation methods. When we discuss interpretability, it is essential to take the audience into account. This model is intended for use by medical professionals, specifically oncologists. Thus, a clear understanding of the precise impact of each of the features on cancer recurrence is critical. For example, to identify the impact of perimeter on cancer recurrence, we can observe the sign of the coefficient in **Figure 5**, and conclude that an increased cell nucleus perimeter decreases recurrence risk. There is certainly more insight and support required to fully

understand a correlation like this, but this is just one example of the interpretability of this model in clinical applications.

V. Outlook

Model & Dataset Improvement

To enhance prediction accuracy, my first step would be to tune additional hyperparameters for each model, which could refine performance and better adapt to the dataset's characteristics. I could also try KNNClassifier, which we learned in class. Also, implementing more advanced feature engineering techniques could help uncover complex interactions between features and capture underlying biological correlations that may not be immediately apparent. Additionally, the dataset's relatively small size poses a challenge for building a robust predictive model. Expanding the dataset by collecting more data points would provide the model with a richer reference base, reducing overfitting and improving generalization to new patients. Furthermore, the current dataset includes many repeating feature types, which, while informative, limit the diversity of input information. Incorporating a broader range of patient characteristics, such as genetic markers, lifestyle factors, or treatment histories, could not only improve the model's predictive power but also enhance its clinical applicability, offering a more comprehensive tool for personalized treatment planning.

References

[1] National Institute of Health. *Cancer Statistics*. April 2, 2015. Accessed December 8, 2024. https://www.cancer.gov/about-cancer/understanding/statistics

[2] American Cancer Society. *Key Statistics for Breast Cancer.* 2024. Accessed December 8, 2024.https://www.cancer.org/cancer/types/breast-cancer/about/how-common-is-breast-cancer.html

[3] Komen, Susan G. *Understanding Breast Cancer Survival Rates*. May 16, 2024. https://www.komen.org/breast-cancer/facts-statistics/breast-cancer-statistics/survival-rates/

[4] National Breast Cancer Foundation, Inc. *Stage 4 Breast Cancer Overview*. Oct. 3, 2024. https://www.nationalbreastcancer.org/breast-cancer-stage-4/

[5] City of Hope. *Breast Cancer Recurrence*. 2024. https://www.cancercenter.com/cancer-types/breast-cancer/types/rare-breast-cancer-types/recurrent-breast-cancer

[6] Wolberg, W., Street, W., & Mangasarian, O. (1995). Breast Cancer Wisconsin (Prognostic) [Dataset]. UCI Machine Learning Repository. https://doi.org/10.24432/C5GK50.

[7] W. N. Street, O. L. Mangasarian, and W.H. Wolberg. An inductive learning approach to prognostic prediction. In A. Prieditis and S. Russell, editors, Proceedings of the Twelfth International Conference on Machine Learning, pages 522--530, San Francisco, 1995. Morgan Kaufmann.

[8] O.L. Mangasarian, W.N. Street and W.H. Wolberg. Breast cancer diagnosis and prognosis via linear programming. Operations Research, 43(4), pages 570-577, July-August 1995.