Predicting the Survivability of Breast Cancer Patients after Neoadjuvant Chemotherapy Using Machine Learning

Linda Banh
Dept. of Electrical Engineering
lbanh@stanford.edu

Robel Daniel
Dept. of Computer Science
robeld@stanford.edu

Preston Ng
Dept. of Computer Science
plng@stanford.edu

Abstract

Breast cancer is the most common type of cancer in the United States [1], and in 15-20% of these cases, these breast cancer patients receive neoadjuvant chemotherapy (NAC) to improve survival. This project was designed around improving methods for predicting survivability in breast cancer NAC patients using characteristics observed at the time of diagnosis. After NAC treatment, breast cancer patients are typically reported as free of cancer if they have a complete response or a partial/no response, otherwise. This paper attempts to correlate survival with patient responses using the Kaplan-Meier Survival Analysis Curve and explores prediction models based on a patient's response to NAC.

The Jupyter Notebook that was used for this project can be viewed here: goo.gl/rPgXci.

1 Introduction and Motivation

Breast cancer is the most common type of cancer in the United States, with an estimated 268,670 new cases expected by the National Cancer Institute in 2018 [1]. In about 15-20% of cases, breast cancer patients receive neoadjuvant chemotherapy (NAC), chemotherapy before surgery, to improve chances of survival. Generally, doctors rely on the residual cancer burden (RCB) score, which is a strong predictor for the patient's likelihood of survival [2].

This score is comprised of six features [3]:

- diameters of primary tumor bed (d_1 and d_2)
- proportion of primary tumor bed that contains invasive carcinoma (which is dependent on overall percentage of carcinoma [$\%CA$] and percentage of in situ carcinoma [$\%CIS$])
- number of axillary lymph nodes containing metastatic carcinoma (LN)
- the diameter of largest metastasis in lymph node (d_{met})

However, because electronic medical records (EMRs) tend to be unstructured and have missing data, there was not enough information to calculate the RCB score. In addition, pathology reports were written in free text form, so it was challenging to get a decently-sized dataset, even after using regular expressions for RCB features. Since there were barriers in creating labels with RCB, a patient's response to NAC was used instead. Patients were labeled as follows:

- 0: Partial/No Response
- 1: Complete Response

A partial/no response meant that there were still some signs of cancer for the patient after NAC at the time of their last check-up, and a complete response meant that the patient had no signs of cancer after NAC at the time of their last check-up. These responses were evaluated by looking at a patient's overall AJCC (American Joint Committee on Cancer) staging status in their EMR.

1.1 Kaplan-Meier Survival Analysis

The Kaplan-Meier Survival Analysis Curve was used to demonstrate the claim that patients with complete responses after NAC will have a higher chance of survival in comparison to patients who did not. This analysis measures the fraction of subjects living for a certain amount of time after a specified time [4]. In this case, this specified timeline was the date of diagnosis until the last time of contact (including the death date if patient is deceased at last time of contact).

Patients who did not have had a death event before the last date of contact were labeled as censored observations. To calculate survivability ($\hat{S}(t)$), the following probabilities were calculated:

$$\hat{S}(t) = \prod_{i:t_i \leq t} \left(1 - \frac{d_i}{n_i}\right),$$

t_i is a time when at least one event happened, d_i the number of deaths that happened at time t_i, and n_i the individuals known to survive (have not yet had a death or have been censored) at time t_i. This was calculated for two groups of subjects (i.e. complete response vs. partial/no response). This is shown in Figure 1.

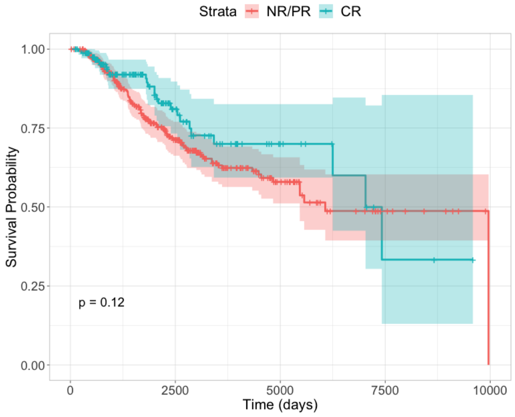

Figure 1: This Kaplan-Meier Survival Analysis curve shows some correlation between complete response and likelihood of survival, with p-value $= 0.12$.

The R code that generated this plot can be viewed here: https://goo.gl/NTnRS8

2 Related Work

As mentioned in the introduction, previous approaches to predicting breast cancer patient survivability post-NAC involve using the RCB score [2]. This value was shown to be a useful predictor in the prognosis of breast cancer recurrence; the simple formula for RCB is given by

$$1.4(f_{inv}d_{prim})^{0.17} + [4(1 - 0.75^{LN})d_{met}]^{0.17} \quad (1)$$

and the fields used (such as primary tumor bed area, cancer cellularity, and cancer-positive lymph node count) were correlated with higher recurrence. However, the unstandardized and incomplete nature of many pathology reports makes RCB difficult to calculate from reports in a database. If the doctor does not record even one of the attributes necessary to calculate RCB, then this value is impossible to find. Even if they do record all the inputs necessary, the nature of pathology reports makes it very difficult to extract the meaningful information automatically.

Other approaches for predicting breast cancer survivability have also been done, but differ from our work in key ways. For example, Delen, Walker, and Kadam have found success with decision trees; however, their data doesn't focus on NAC patients, and they use an arbitrary definition of survival[5]; Bellaachia and Guven take a similar approach as Delen et al [6]. Khan, Choi, Shin, and Kim use fuzzy decision trees, on the general SEER breast cancer dataset that Bellaachia and Guven used; they also did not use NAC patients only and had lower accuracy [7], which is also true for the decision tree model proposed by Liu, Wang, and Zhang [8]. In general, past researchers have used large datasets and have not gone into detail with how they chose features and why.

Our approach, focuses on neoadjuvant chemotherapy breast cancer patients using patient response as a label (proven to be correlated with survival through the Kaplan-Meier estimator) with discerning feature selection, and using bootstrap random forests to avoid overfitting. This differs from the few previous papers on using statistical/ML methods to predict breast cancer patient survivability.

3 Methods/Experiments

3.1 Data and Feature Selection

Data was received from the OncoShare database [9], a project between Stanford and Palo Alto Medical Foundation (PAMF) founded in 2008. Their goal is to use ”big data” to improve breast cancer care. This database is comprised of the statewide, population-based California Cancer Registry, EMRs from Stanford University Hospital, and multiple sites of the community-based PAMF healthcare system. Records detail genomic sequencing results from clinical testing laboratories, patient-reported data on cancer care preferences, pathology/radiology reports, and more.

For this project, feature selection was a notable challenge since there were more than 200 features, many of which were sparse (missing information) or not relevant for patient response prediction. Thankfully, the OncoShare database had a codebook detailing the specifics of each column of the EMRs. However, looking through the codebook, more than a dozen seemed relevant for prediction models, and this drew some concern for over-fitting features. Therefore, after some discussion with Professor Itakura (oncologist from Stanford School of Medicine), the long list of features was reduced to the following:

- primary site (location where the tumor originated)
- laterality of the tumor (side of the body in which the tumor originated)
- tumors cell type
- tumor behavior (malignant, in situ, benign, or uncertain)
- sequence of all reportable neoplasms during the patient's lifetime determined by the central registry
- estrogen-receptor characteristics of tumor
- progesterone-receptor characteristics of tumor
- actual number of tumors
- site specific information
- the type of diagnostic/staging procedure

3.2 Preprocessing

Generally, EMRs are missing information or are sparse; thus, a large part of this project was dedicated to pre-processing. To pre-process the data, a combination of Microsoft Excel and pandas [10] was used. Since this project focused on NAC breast cancer patients, the data needed to be screened and parsed for this requirement. From the 24,301 patient records received, 2,139 were NAC breast cancer patients. From these 2,139 records, only 340 records were usable and did not have missing data for the features selected in *Data and Feature Selection*. Looking through the 340 NAC patients, 109 patients exhibited a complete response to NAC (labeled as 1), and the remaining 231 patients showed partial/no responses (labeled as 0).

Afterwards, some data needed to be re-encoded. Since many features had categorical data, a way to overcome this was to re-encode the categorical data to discrete numbers starting at 0. An example of this is shown in Table 1, on page 4 for *SITE_02* (location where tumor originated).

3.3 Choosing Machine Learning Models

After the data was pre-processed, the NAC patient response predictions were tested using logistic regression, k-nearest neighbors (KNN), and bootstrapped random forests.

Logistic regression is a binary classifier, and makes a prediction ($\hat{y}$) using the sigmoid function: $h_\theta(x) = \frac{1}{1+\exp(-\theta^T x)}$ [11]. This classifier makes a decision based on the probability of what is observed given a feature set ($P(y=1|x) = h_\theta(x)$, $P(y=0|x) = 1 - h_\theta(x)$). To put this quantitatively,

$$\hat{y} = \begin{cases} 0, & P(y=0|x) \geq P(y=1|x) \\ 1, & otherwise \end{cases}$$

It was chosen as a baseline since it is known to produce reliable classification for binary data and is a good fit for this problem since we aim to classify patients with complete or not a complete response. KNN is a simple, non-parametric machine learning technique that classifies data based on a majority vote from its k neighbors; because of this, KNN seemed like a good choice in balancing the bias-variance trade-off since the dataset was so small, with a size of 340. In addition, it made sense intuitively that patients who exhibited similar characteristics at the time of diagnosis might also have similar responses after NAC.

Lastly, bootstrapped random forest was chosen. Bootstrapped random forest is an algorithm that utilizes decision trees to split on features depending on a threshold.

$$S_p(j,t) = (\{x : x_j < t\}, \{x : x_j \geq t\})$$

Best splits are selected via Gini Impurity:

$$I_G(p) = \sum_{i=1}^{J} p_i \left(1 - p_i\right)$$

and each tree is run with ”bagged” subsamples (randomly sampled datapoints with replacement). All trees are averaged afterwards [12]. Because all randomly generated trees are bagged and averaged afterwards, bootstrap random forest is helpful in reducing variance and thus, this seemed appropriate for the problem since there were many features but relatively few patients.

All three algorithms were implemented using scikit-learn [13] and NumPY [14].

Tumor Location Origin	Description	After Re-Encoding
C500	Nipple (areolar)	0
C501	Central portion of breast (subareolar) area w 1 cm around areolar complex	1
C502	Upper inner quadrant (UIQ) of breast	2
C503	Lower inner quadrant (LIQ) of breast	3
...	...	...

Table 1: Re-encoding of Labels Example for SITE_02

3.4 Testing/Measuring Performance

For training, validation, and testing, the NAC breast cancer patient dataset was split 80/20% for train/test, and then the 80% was utilized for 5-fold cross validation (CV). Afterwards, each model's performance was evaluated using the following metrics: train/test accuracy, precision, recall, and specificity.

4 Experiments/Results

As stated in *Introduction and Motivation*, the objective is to predict whether a patient exhibits a complete response or partial/no response at the time of a NAC patients last follow-up, which is determined by their overall AJCC stage in their EMR. To test this, logistic regression (Figure 2), k-nearest neighbors (Figure 3, 4), bootstrapped random forests (Figure 5, 6) algorithms were implemented and each model's performance was evaluated (Table 2).

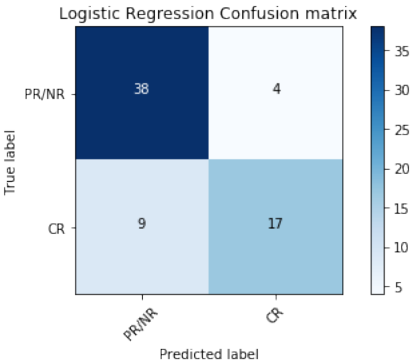

Figure 2: Logistic Regression Confusion Matrix

4.1 Discussion/Challenges

Since the averaged CV accuracy and the train accuracy (from the 80/20 split) were quite similar (with a +/- 2% margin), the train and test accuracy from the 80/20 split will be used to compare the models in this discussion. (Other information such as confusion matrices and train/test accuracy plots can be viewed in the figures discussed in *Experiments/Results*.)

Looking across all models, each algorithm performed relatively well. Bootstrap random forest performed the best (optimal performance with a depth of 7); it had a train accuracy of 99% and a

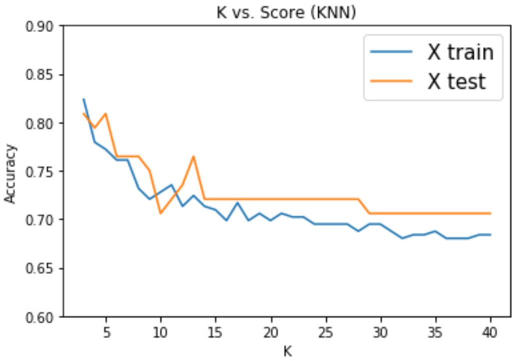

Figure 3: KNN K vs. Accuracy

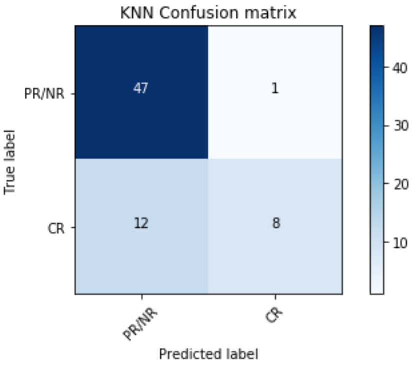

Figure 4: KNN Confusion Matrix (K=3)

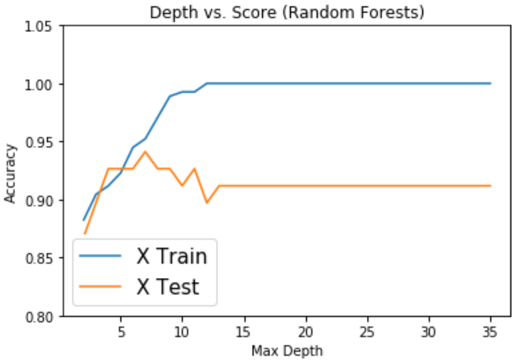

Figure 5: Random Forest Depth vs. Score

Models	5-fold CV Accuracy	Train Accuracy	Test Accuracy	Precision	Recall	Specificity
Logistic Regression	0.88	0.9044	0.8088	0.77	0.89	0.81
KNN	0.84	0.8419	0.75	0.5	0.78	0.65
Bootstrap Random Forest	0.99	0.9890	0.9265	0.81	0.91	0.95

Table 2: Comparison of Performance Metrics Across All Models

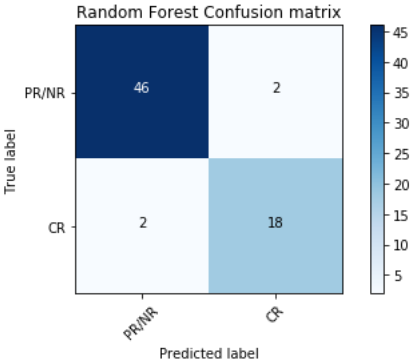

Figure 6: Bootstrap Random Forest Confusion Matrix

test accuracy of 93%. Meanwhile, KNN, with k = 3, performed the worst, with a train accuracy of 84% and test accuracy of 75%. Logistic regression was somewhere in the middle with a 90% train accuracy and 81% test accuracy. Bootstrapped random forest may have shown good performance since the algorithm was built to combat high variance from over-fitting features, since all randomly generated trees were averaged. On the other hand, KNN may have performed poorly because it was built based on the assumption that NAC patients who exhibited similar traits at the time of diagnosis would show similar responses after NAC at the time of their last checkup.

Despite the good results that were seen across all models, there may be some sampling bias, since the dataset is so small (m = 340). Even though there was a strong correlation between patients with a complete response and higher chances of survivability (as shown in Figure 1), it is hard to make a conclusion about how reliable predicting survivability via complete response is and how it compares with RCB scores (since required RCB data fields could not be collected). However, the information gathered proves that logistic regression, KNN, and bootstrap random forest can be good measures of whether a patient will have complete or not a complete response after NAC, and these models have the potential to be applied to the larger NAC breast cancer population.

5 Conclusions and Future Plans

For this project, simple supervised learning methods (logistic regression, KNN, and bootstrap random forest) were implemented to predict whether a patient would exhibit a complete or not a complete response at the time of their last visit. These responses were compared to the Kaplan-Meier Survival Analysis Curve and used to estimate their chances of survival after some specified time (measured in days).

After some assessment, bootstrap random forest performed the best and had high accuracy, but because of a small dataset (m = 340), there may be some sampling bias. Therefore, if there was more time to work on this project, we would take the following steps:

1. Collaborate with pathologists/radiologists and other healthcare professionals to gather more RCB data. It would be helpful to have an RCB model to compare with our current model (complete or not a complete response) to see how well this model predicts survivability in NAC patients.

2. Utilize more robust natural language processing techniques that could potentially be used to process EMRs. Again, this would help create a reliable NAC breast cancer RCB dataset, which can be used to assess how accurate survivability of NAC breast cancer patients is predicted.

3. Use a mixture of RCB and EMR features in our models and see how it improves the current RCB model.

Hopefully with these changes, this can help provide clearer conclusions about our model and how it could improve in the future.

6 Contributions

As a group working on this project, we contributed equally overall. Linda Banh had large individual contributions in evaluating data features, analyzing the model performance, and writing the milestone report. Robel Daniel had large individual contributions in literature research, evaluating data features, and putting together the poster. Preston Ng had large individual contributions in preprocessing the data, implementing each machine learning model, and putting together the poster. Together, we designed the pipeline for this project, collaborated on what models were appropriate for the problem we were solving, and put together this final report.

Acknowledgments

Thank you to Professor Haruka Itakura for her contributions to this project. Not only did she give us access to the breast cancer patient database in OncoShare, but she also helped us focus on important patient features for our model designs. We would also like to thank our project TA, Ethan Steinberg, for his help in guiding us on how we should pre-process EMRs and helping design ways to handle sparse data.

References

[1] American Cancer Society, Inc., "Common cancer types," 2018. [Online]. Available: https://www.cancer.gov/types/common-cancers

[2] W. Fraser Symmans, F. Peintinger, C. Hatzis, R. Rajan, H. Kuerer, V. Valero, L. Assad, A. Poniecka, B. Hennessy, M. Green, A. U Buzdar, S. Eva Singletary, G. N Hortobagyi, and L. Pusztai, "Measurement of residual breast cancer burden to predict survival after neoadjuvant chemotherapy," *Journal of clinical oncology : official journal of the American Society of Clinical Oncology*, vol. 25, pp. 4414–22, 10 2007.

[3] Z. Nahleh, MD, D. Sivasubramaniam, MD, S. Dhaliwal, MD, V. Sundarajan, MD, and R. Komrokji, MD, "Residual cancer burden in locally advanced breast cancer: a superior tool," 2008. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2601022/

[4] E. Kaplan and P. Meier, "Nonparametric estimation from incomplete observations," *Journal of the American Statistical Association*, vol. 53, no. 282, pp. 457–481, 1958.

[5] D. Delen, G. Walker, and A. Kadam, "Predicting breast cancer survivability: a comparison of three data mining methods," *Artificial Intelligence in Medicine*, vol. 34:2, pp. 113–127, 6 2005.

[6] A. Bellaachia and E. Guven, "Predicting breast cancer survivability using data mining techniques," 1 2006.

[7] M. U. Khan, J. P. Choi, H. Shin, and M. Kim, "Predicting breast cancer survivability using fuzzy decision trees for personalized healthcare," *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 8 2008.

[8] Y.-Q. Liu, C. Wang, and L. Zhang, "Decision tree based predictive models for breast cancer survivability on imbalanced data," *2009 3rd International Conference on Bioinformatics and Biomedical Engineering*, pp. 113–127, 6 2009.

[9] A. W. Kurian and H. S. Luft, "Oncoshare," 2008. [Online]. Available: http://med.stanford.edu/oncoshare.html

[10] McKinney, Wes and others, "pandas: Data analysis python library." [Online]. Available: http://www.pandas.pydata.org/

[11] Andrew Ng, "Supervised learning," 2018. [Online]. Available: http://cs229.stanford.edu/notes/cs229-notes1.pdf

[12] Course Staff Fall 2018, "CS229 Midterm Review," 2018. [Online]. Available: http://cs229.stanford.edu/materials/cs229-mt-review.pdf

[13] Pedregosa, Fabian and Varoquaux, Gael and Gramfort, Alexandre and Michel, Vincent and others, "scikit-learn: Machine learning in python." [Online]. Available: http://www.scikit-learn.org/

[14] Hugunin, Jim and others, "Numpy: Scientific computing with python." [Online]. Available: http://www.numpy.org/