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

Linda Banh, Robel Daniel, Preston Ng
{lbanh, robeld, plng}@stanford.edu

Special thanks to Dr. Haruka Itakura (oncologist and assistant professor at Stanford) and CS 229 TAs for their advice and help on our project.

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. Traditionally, a patient's survivability is calculated via a residual cancer burden (RCB) score[2]:

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

Calculating RCBs is difficult because oftentimes, medical records are missing information needed to calculate the score. Therefore, in our project, we will observe complete (disappearance of all signs of cancer after treatment) or not complete response to predict survival instead.

Goal: To predict survivability of breast cancer patients after neoadjuvant chemotherapy, using overall AJCC[3] cancer staging labels (complete or not a complete response) and supervised learning algorithms.

Data and Features

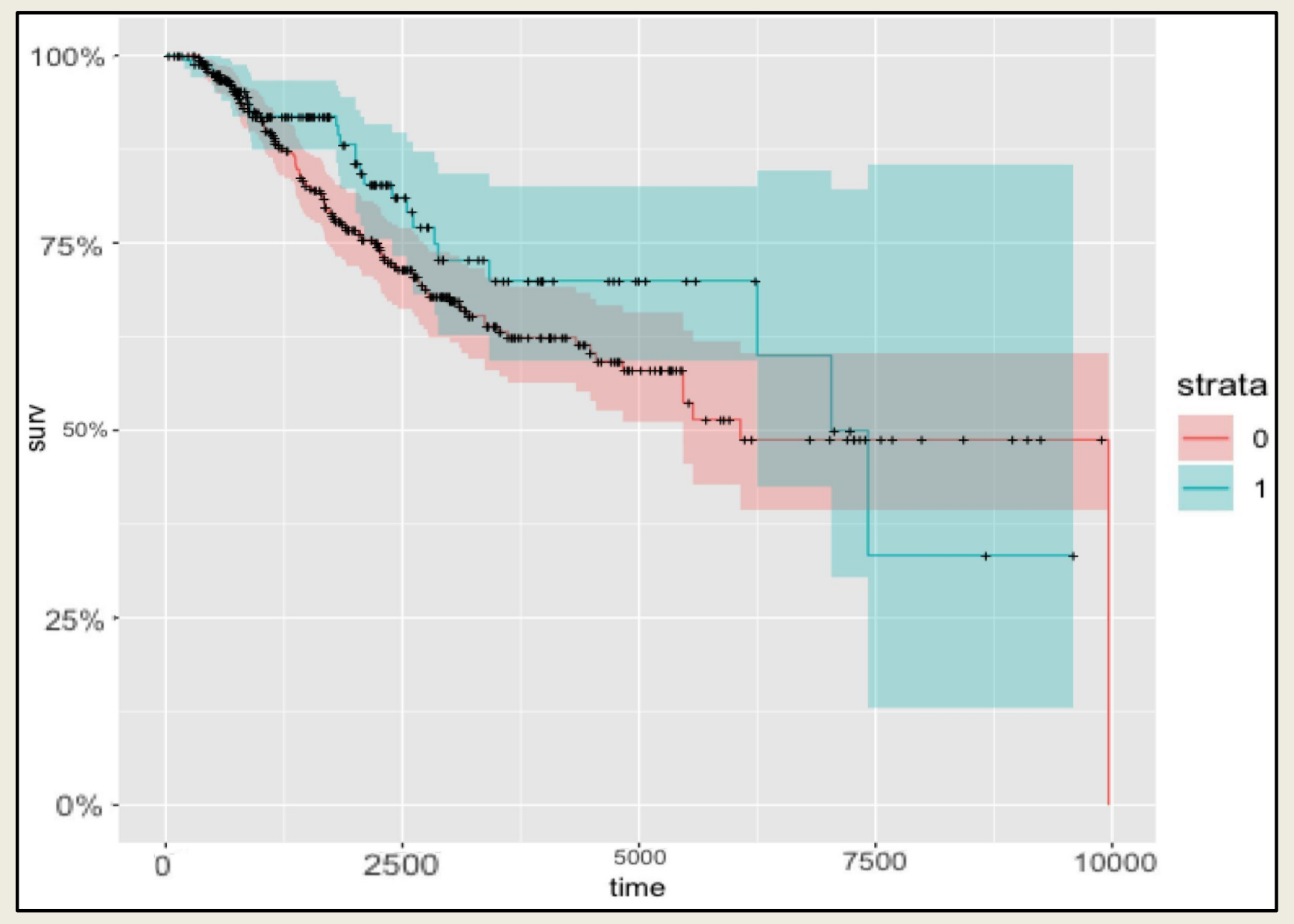

Figure 1 (left): Kaplan-Meier[4] Survival Analysis curve to show the correlation between complete response and likelihood of survival (0: partial/no response, 1: complete response) Survivability is increased for neoadjuvant patients with complete response.

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

Data: Breast cancer patient electronic health records (EHRs), provided by Oncoshare Database (with 340 NAC patients)

- ❖ Row = patient (anonymous IDs)
- ❖ Column = information about patient (ex. pathology report, tumor site, etc.)

Anon_ID	Tumor Site	Behavior of Tumor
1	C500	Benign
2	C502	In Situ
3	C501	Unknown

Figure 2 (left): Example CCR Tumor Data
*Used AJCC_P (AJCC staging label from electronic health records) and patient morbidity data for the ground truth for survival

Features: To determine features, we consulted our advisor and examined features with evidence of patient survivability correlation. These features included characteristics of the tumor(s) as well as characteristics of the patient themself. The categorical features were then mapped to discrete values indexed at 0 using a label encoder. Some features we used were:

- ➢ **site specific information about tumor and where it originated**
- ➢ **cell type and behavior of tumor (malignant, in situ, benign, or uncertain)**
- ➢ **sequence of all reportable neoplasms during the patient's lifetime**
- ➢ **tumor count**
- ➢ **overall cancer stage**

Model (Logistic Regression)

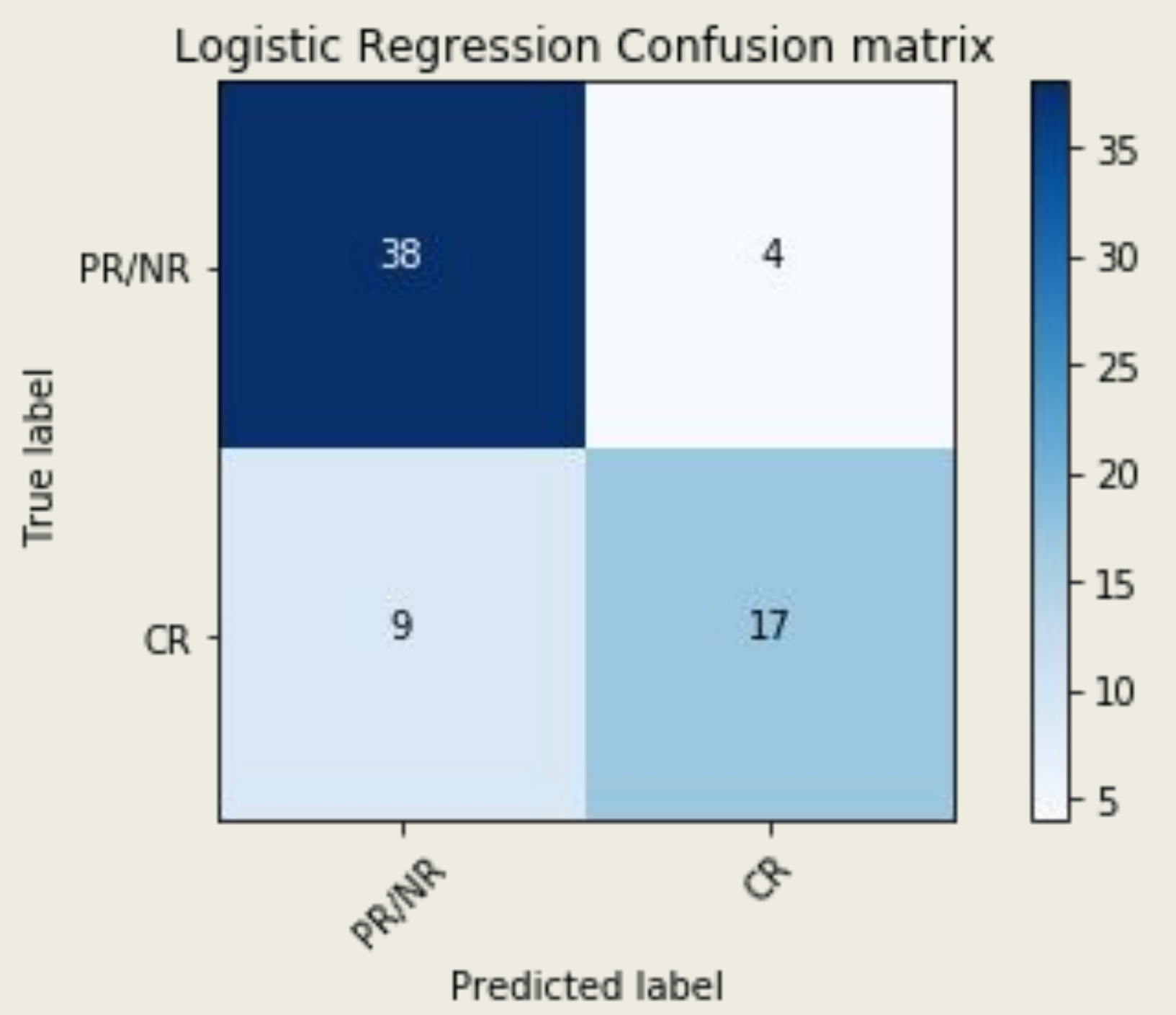

Figure 3 (left): Confusion matrix for logistic regression (used as our baseline for our machine learning analysis and is simple to compute)

$$z = w_0x_0 + w_1x_1 + \ldots + w_mx_m = \sum_{j=0}^{m} w_jx_j = \mathbf{w}^T\mathbf{x}$$

$$\phi(z) = \frac{1}{1 + e^{-z}}$$

Model (*k*-NN)

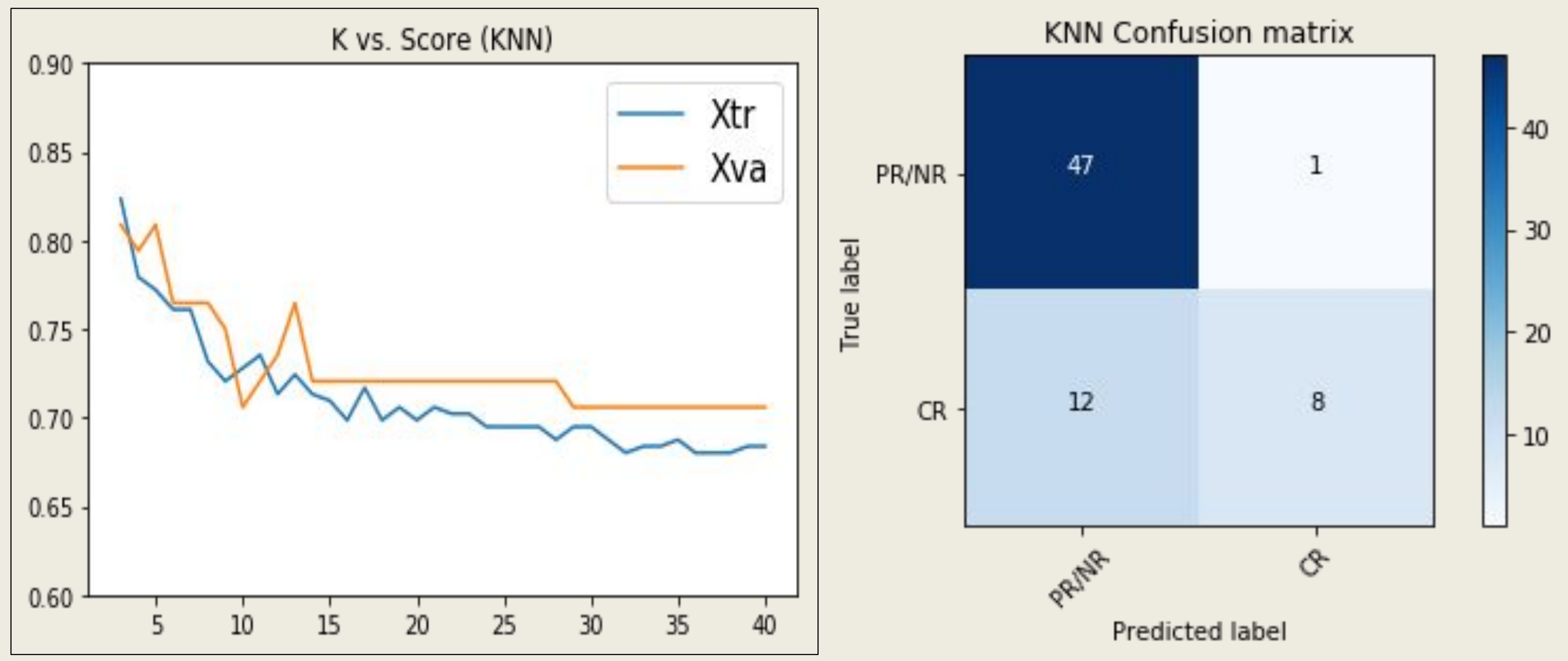

Figure 4 (above): Training vs validation curve for *k*-NN and associated confusion matrix, using k=3 (assigning a training datum to the label shared by a majority of the 3 nearest feature vectors). *k*-NN is simple and makes no assumptions about the data.

Model (Bootstrap Random Forest)

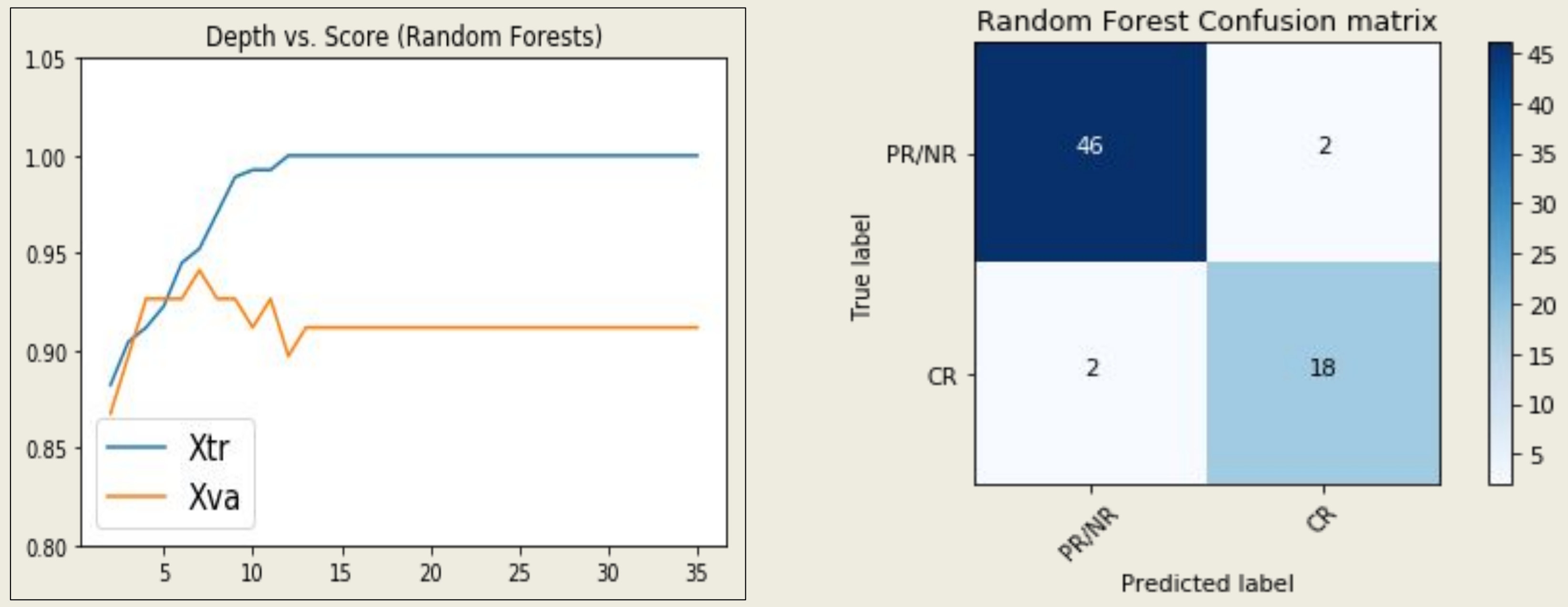

Figure 5 (above): Training vs validation curve for Bootstrap Random Forest and associated confusion matrix (bootstrap random forest is a computationally inexpensive algorithm and lets us learn about the data quickly without knowledge of labels)

$$I_G(p) = \sum_{i=1}^{J} p_i \sum_{k \neq i} p_k = \sum_{i=1}^{J} p_i(1 - p_i)$$

Results

	Train Error (m=272)	Test Error (m=68)	Precision	Recall	Specificity
Logistic Regression	0.09559	0.19118	0.77	0.89	0.81
KNN (K=3)	0.15441	0.25	0.5	0.78	0.65
Bootstrap Random Forest	0.01103	0.07353	0.81	0.91	0.95

Table 1: Test/train error and other performance metrics for logistic regression, k-NN, and bootstrap random forest models

Discussion/Challenges and Future Steps

Discussion/Challenges:
In Figure 1, Kaplan-Meier Survival Analysis shows that correlation between a complete response (CR) and survivability is higher than for patients with partial or no response. Looking at our results, we found that logistic regression, KNN (for K = 3), and bootstrapped random forest models all showed good performance, with test errors of 25% or less. Bootstrapped random forests demonstrated the best performance, with only a 7% error. However, the results may not be as reliable due to the small data set (m = 340) and sampling bias. Because our project focused on NAC patients, we were limited to a small subset of patients. Even though there was a strong correlation between patients with CR and higher chances of survivability, it is hard to make a conclusion about how reliable predicting survivability via the standard residual cancer burden (RCB) score is.

Future Steps (6 month plan):

1. Gather the required information, by talking with pathologists and radiologists, to create a large enough dataset to calculate RCB (more reliable results)
2. Test/utilize more robust natural language processing techniques to process EHRs
3. Evaluate how well a mixture of RCB features and EHR features collected at the time of first diagnosis does for predicting RCB scores and determine how the model for predicting RCB can improve

References

[1] "Common Cancer Types", *National Cancer Institute*, 2018. [Online]. Available: https://www.cancer.gov/types/common-cancers. [Accessed: 10- Dec- 2018].

[2] W. Symmans, *et al.*, "Measurement of Residual Breast Cancer Burden to Predict Survival After Neoadjuvant Chemotherapy", *Journal of Clinical Oncology*, vol. 25, no. 28, pp. 4414-4422, 2007.

[3]S. Edge, *AJCC cancer staging handbook*. New York: Springer, 2010.

[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.

Tools: Python 3, Jupyter Notebook, Numpy, Pandas, Matplotlib, Scikit-learn