Using Machine Learning to Predict the Post-Operative Life Expectancy of Lung Cancer Patients

Sushant Thyagaraj
sushantt@gmail.com

June 1, 2017

Introduction

Lung cancer is the leading cause of cancer death in humans, for both men and women. According to the Center for Disease Control and Prevention, 212,584 people are diagnosed with lung cancer annually.[1] With the exponential surge in technological development in the 21st century, computational tools, including machine learning, are becoming increasingly useful techniques to target many such diseases in the ways of disease detection, disease prevention, and many other purposes. The increasing precision of computational algorithms for the identification of trends through data shows promise as a useful diagnostic tool in the medical field.

In this paper, a tool was developed to predict the post-operative life expectancy of lung cancer patients using the computational methods of logistic regression, gradient boosting, and neural network models. These methods were used specifically to predict whether a lung cancer patient will survive one year after he or she has had thoracic surgery. The results of each of the techniques were then measured and compared based on accuracy and performance.

It was concluded that the neural network method produced the predictor tool with the highest accuracy rate. The gradient boosting algorithm was slightly less accurate but still produced good results. Finally, the logistic regression algorithm came last in performance and was unable to generate a good fit for the data. Another conclusion from this project was addressing the many issues that come with public, open-access data, ranging from substantial noise to missing records, which all contribute to lost data integrity within the dataset.

Data

The data used in this paper was taken from the Machine Learning Repository belonging to the University of California, Irvine.[2] The data was collected by Wroclaw Thoracic Surgery Centre in Poland.[3] The data is sourced from patients who underwent major lung resections for primary lung cancer in the years 2007-2011.More information on the data is presented in the following table:

---

[1] Heart Disease Facts & Statistics. Centers for Disease Control and Prevention. 2015 Oct [accessed 2016 Jul 25]. http://www.cdc.gov/heartdisease/facts.htm

[2] https://archive.ics.uci.edu/ml/datasets/Echocardiogram

[3] UCI Machine Learning Repository: Echocardiogram Data Set. UCI Machine Learning Repository: Echocardiogram Data Set. [accessed 2016 Jul 25]. https://archive.ics.uci.edu/ml/datasets/echocardiogram

Attribute	Description
DGN	Diagnosis - specific combination of ICD-10 codes for primary and secondary as well multiple tumors if any
PRE4	Forced vital capacity
PRE5	Volume that has been exhaled at the end of the first second of forced expiration
PRE6	Performance status - Zebra scale
PRE7	Pain before surgery
PRE8	Haemoptysis before surgery
PRE9	Dyspnoea before surgery
PRE10	Cough before surgery
PRE11	Weakness before surgery
PRE14	T in clinical TNM - size of the original tumor
PRE17	Type 2 DM - diabetes mellitus
PRE19	MI up to 6 months
PRE25	PAD - peripheral arterial diseases
PRE30	Smoking
PRE32	Asthma
AGE	Age at surgery
Risk1Y	1 year survival period - True value if died, False if alive

Table 1: List of Variables and Details

According to the documentation of the data, a few of the variables measured could be ignored. The DGN was solely a diagnosis identification number, so it played no role in the outcome of the classification. Therefore, it was excluded from the data.

The dataset contains a total of 470 patients, with 17 values ideally being recorded for all patients. The data consisted of roughly 7,990 data points, which was a good amount to develop a model for.

The detailed modeling techniques that were used in this project were, specifically: logistic regression, gradient boosting, and neural networks.

Logistic regression is a statistical technique that is commonly used in dataset analysis. Logistic regression is used primarily to explain the relationship between a dependent, binary variable and other independent variables, based on a logistic curve. Since the thoracic surgery

data had a response variable that was binary (1 for dead, 0 for alive), logistic regression was deemed a relevant tool for predictive analysis.

Gradient Boosting, or simply boosting, was also used as an ensemble learning method. In boosting, bootstrap sampling was not involved; instead, each tree was fit on modified version of the original data set where basis models were iteratively added to reduce the selected loss function. Boosting was applied to the dataset to determine whether this type of ensemble learning would enhance the predictive capabilities of the logistic regression algorithm.

A neural network model, which is a learning algorithm inspired by the structure and functional aspects of biological neural networks, was also used. The computation was structured through interconnected groups of artificial neurons that processed the data using a connected approach to computation, aiding the modelling of more complex relationships between inputs and outputs, all to find patterns in data. In this dataset, an artificial neural network was developed to check for an intricate relationship between the input predictor variables and the classification variable.

Results

Data cleaning was a major part of the dataset analysis. First, as mentioned before, the variable DGN was removed from the data in accordance to the documentation from UCI.

A surprising factor of this dataset was that there were no missing values present. After traversing through the whole dataset, all values were accounted for, which was very beneficial in developing an accurate predictor.

After data cleaning was conducted, the final number of patients who were dead vs. patients who were alive was determined using a bar plot.

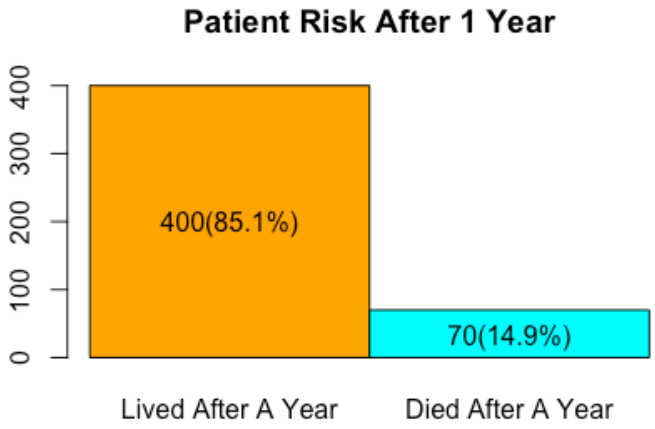

Figure 1: Bar Plot of Risk of Death After 1 Year Of Thoracic Surgery

As evident from the bar plot above, the data was heavily skewed towards the majority class, the class of interest, which in this case was the patients who had died after a year. However, this skew of data was not addressed until after the subdivision into training and testing data, as it was integral to make sure that the testing data remained unbalanced to mimic real-world data.

Before beginning model development, all numerical variables were normalized and all categorical and factor variables were binarized. The normalization essentially converted all of the variables to decimal numbers between 0 and 1, with 0 being the minimum of any value, and 1 being the maximum that could be taken.

Next, for logistic regression, in order to create the optimal model, variables had to be selected using variable selection techniques. A backward elimination algorithm which employs the Akaike Information Criterion (AIC) was used to determine which variables should be kept in the model. AIC is based on maximum likelihood and gives a penalty for each parameter in the model, AIC helps in selecting a model by trying to balance the conflicting demands of accuracy and simplicity and keeping parsimony in mind. To begin, the backward elimination algorithm starts with all of the predictor variables, and its goal is to find the model that has the smallest AIC. The algorithm tests the deletion of one variable at a time to see how it affects the AIC value. After analyzing all possible models, the algorithm selects the model fit with the smallest AIC value.

After performing backward elimination on the logistic regression model, the variables that were determined to be statistically significant were the following:

1. PRE4
2. PRE5
3. PRE7
4. PRE9
5. PRE11
6. PRE14
7. PRE17
8. PRE30

The second variable selection algorithm that was run on the data was machine learning variable selection through Random Forest. In this technique, the variable importance was measured via finding the mean decrease in the Gini index, which measures how each variable contributes to the homogeneity of the nodes and leaves in a Random Forest.[4] Variables with higher mean decreases in Gini coefficient are more significant, as they result in nodes with higher purities.[5]

The mean decrease in Gini was measured using a variable importance plot. In order to generate the plot, a few trees were created from the data, and then the mean decrease in Gini was

[4] Random Forests. Dinsdale et al. Supplemental Material. [accessed 2016 Jul 29]. https://dinsdalelab.sdsu.edu/metag.stats/code/randomforest.html

[5] Ibid.

obtained using the Gini function. The data was then graphed in a variable importance plot to visualize the mean decrease in node purity per variable.

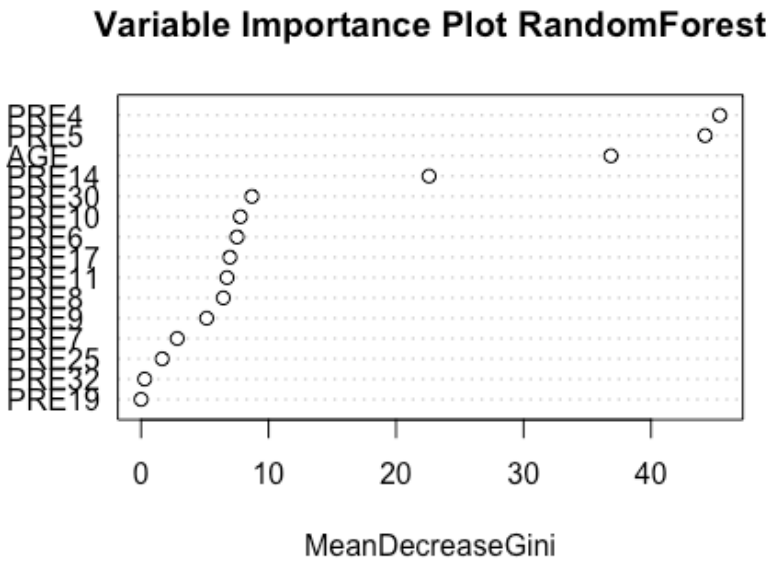

Figure 2: Variable Importance Plot Based on Random Forest

According the variable importance plot and the Gini coefficient, the "best" variables to use in the model can be found below:

1. PRE4
2. PRE5
3. AGE
4. PRE14
5. PRE30
6. PRE10

Modeling

Before modeling, the thoracic surgery data was separated into two subsets, a training dataset and a testing dataset, as is common procedure when developing a machine learning algorithm. However, due to the skew of the data towards the class of interest, the training set had to be balanced in order to minimize bias. Therefore, the minority class in the training data was oversampled so that there was a 50% balance between both the variables Died after a Year and Lived after a Year. This was all done using the ROSE package in R. The training data, balanced, and the testing data, unbalanced, were always split in a ratio of 70:30, respectively. The data samples were also completely randomized in training and test data to minimize as much bias as possible. Most importantly, only the logistic regression model used the variables that were listed as the "best" to use by the mean decrease in Gini index test and the backward selection algorithms. Both the gradient boosting and the neural network used the

whole set of variables since they had the potential to derive more complex fits and connections than logistic regression.

The models developed in this project were a logistic regression model, a gradient boosting model, and a neural network. For the first model, a generalized logistic regression model was fitted to the training data using the "glum" package in R. The second model utilized the "xgboost" package in R to build decision trees to fit the data. Lastly, a neural network learning algorithm was fitted to the training data using the package "neuralnet" in R as well. Out-of-sample performances were measured for all of the models by fitting them to the test data, with the goal being to predict the values of Risk1Y (the response variable). The predictions of each model were then compared to the actual values of the response variable in the testing data to measure accuracy, which was done using confusion matrices. The precision and recall values were also measured for each model as additional performance tests, along with F1 score and No-Information Rate.

Plots:

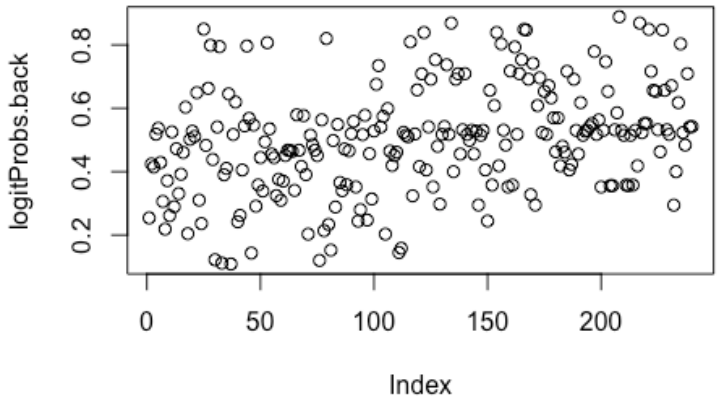

Figure 3: Predicted Probabilities For Logistic Regression

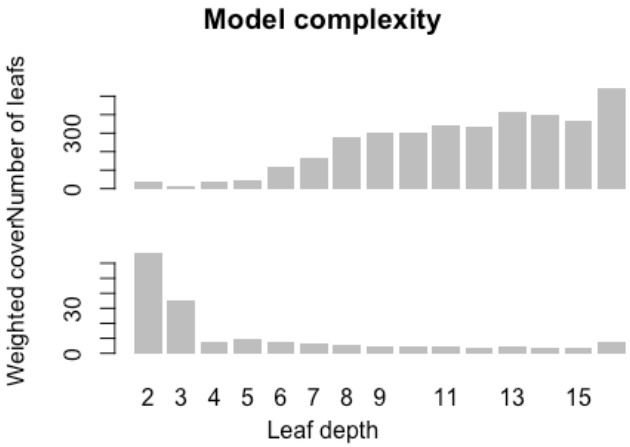

Figure 4: Measure of Model Deepness for Gradient Boosting

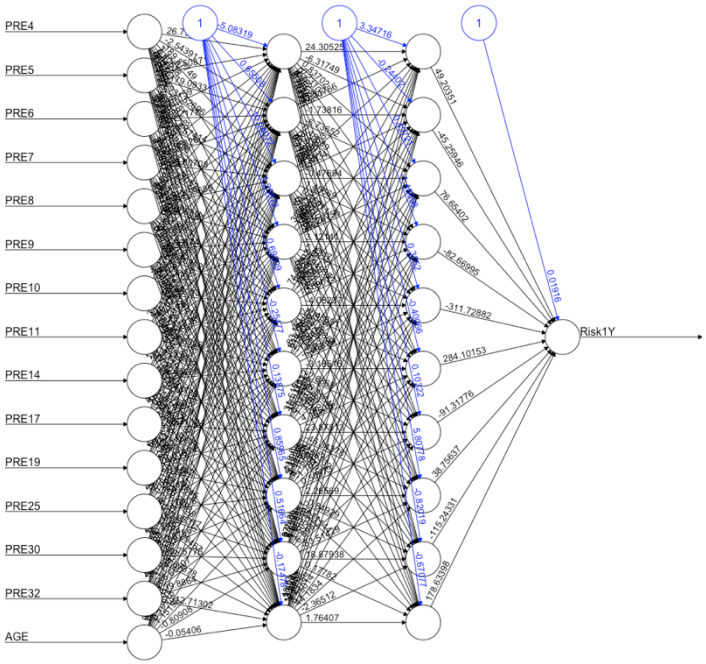

Figure 5: Visualization of Neural Network Fit

Conclusion

To summarize the process up till now, the data was first cleaned thoroughly to eliminate all issues within the dataset. Changes were made to the data in accordance to the documentation, including removing unnecessary columns. Additionally, all of the variables were either normalized or binarized depending on their type, ultimately to have a value between 0 and 1, with 0 being the absolute minimum, and 1 being the absolute maximum.

The data was then split into training and testing groups, in a ratio of 70:30. The testing and training samples were completely randomized to reduce bias and increase the predictive potential and precision of the models developed. Since the data was heavily skewed towards the class of interest, an oversampling of the minority class in the training data was conducted in order to balance the training set and develop a stronger predictive tool. The testing data remained unbalanced. Then, after dividing the data, the most useful variables for logistic regression were selected by measuring AIC in a backwards stepwise algorithm, as well as through the mean decrease of the Gini index by variable. The top variables with the greatest mean decrease were then cross validated with the variables chosen in the backwards stepwise algorithm based on AIC, and the final variables were used in the logistic regression model.

Three models were then developed on the cleaned data. They were all developed on the training data, and then applied to the test data. Accuracy and performance of individual models were measured using confusion matrices, AUC, and cross validation.

The balanced evaluation metrics measured for each of the models is listed in the following table:

	Balanced Acc.	Precision	Recall	Kappa
Logistic Regression	66.67%	65.62%	70.00%	33.33%
Gradient Boosting	80.0%	81.57%	77.5%	60.00%
Neural Networks	95.23%	98.36%	90.47%	94.17%

Table 2: Algorithms and Evaluation Metrics

Discussion

From the table above, it is clear that the neural network was able to derive a very accurate fit within the data, while the logistic regression and gradient boosting weren't able to produce as good fits.

The reason that the algorithms used weren't able to produce a perfect fit is because of issues with the data integrity itself. Specifically, the logistic regression and gradient boosting were unable to work around the noise in the data, resulting in less than adequate fits. Additionally, the dataset being open-access and public usually implies that not as much caution and care is taken during the data generation phase. Therefore, the dataset contained a substantial amount of noise, contributing to a level of variability and inconsistency within the data.

More patient records would most likely have facilitated this project, as, currently, there are only records of 470 patients. More data would help establish an even better fit for all the logistic regression and gradient boosting models, since there would be more data available for the algorithms to utilize. For the neural network, the accuracy was high enough that it may not have been worthwhile to feed any more data. Another change that might have helped model

development and accuracy would be having more features measured for each patient. More features would add more data points, which would in principle improve the fits of the models significantly.

In all, one of the main takeaways of this project is that, machine learning algorithms are highly skilled at finding the functional form of a model on data, even if the variability within the dataset is significantly high. In this thoracic surgery data, despite it being a public dataset and having noise within the data, the best algorithm, the neural network, was still able to predict with a near-perfect accuracy of 97.08%, meaning that the neural network was able to learn well enough to account for the noise in the data and work around it almost to perfection. Another takeaway is the general, underlying procedure for the development of a machine learning algorithm on any dataset, whether it be technological data, biomedical data, or any other type of data measured. The procedure is as follows:

1. Clean the raw data using various cleaning techniques like imputation, normalization, and transformation and conduct exploratory data analysis.
2. Split the data into a training set and a testing set and conduct variable selection on the training data, if necessary,
3. Develop the model using the training data.
4. Apply the model to the testing data as a predictor tool.
5. Finally, measure accuracy, precision, and recall and other metrics to determine the performance of the model.

The final takeaway detailed in this paper discusses the characteristics of open-access data. Since the data is public, there is no major incentive to record the data properly or frequently, which is why there was some noise in the dataset, accounting for the major inaccuracy in the logistic regression data. However, the neural network was able to work around the noise and still produce a very accurate fit. Regarding public data, this specific type of data also is not usually collected in a high quality setting. Researchers face many technical issues with this public data, including inadequate access to networked storage, data loss due to lack of proper organization, as well as other problems.[6] However, despite these inherent issues with data collection and data management, the best machine learning algorithm still had a predictive accuracy of about 97%, which is almost a perfect fit.

All in all, machine learning has major potential when it comes to data analytics and model development, since machine learning models have the ability to "learn" and adapt as they continue to analyze new data, which sets them apart from other methods.[7] Machine learning has the capability to produce models that target larger-scale data and deliver faster, more accurate results, all with minimum intervention. This, ultimately, is the revolutionary potential that machine learning brings to all real-world issues with data involved.

---

[6] Jahnke L, Asher A, Kermis SDC. The Problem of Data. Council Library on Information Resources; 2012. https://www.clir.org/pubs/reports/pub154/pub154.pdf

[7] Machine Learning: What it is and why it matters. Analytics, Business Intelligence and Data Management. [accessed 2016 Jul 30]. http://www.sas.com/en_id/insights/analytics/machine-learning.html

Bibliography

1. Lung Cancer Statistics. Centers for Disease Control and Prevention. https://www.cdc.gov/cancer/lung/statistics/. Published 2016. Accessed November 24, 2016.

2. Thoracic Surgery Data Set. UCI Machine Learning Repository: Data Set. https://archive.ics.uci.edu/ml/datasets/Thoracic Surgery Data. Accessed November 24, 2016.

3. Jahnke L, Asher A, Kermis SDC. The Problem of Data. Council Library on Information Resources; 2012. https://www.clir.org/pubs/reports/pub154/pub154.pdf

4. Machine Learning: What it is and why it matters. What it is and why it matters | SAS. http://www.sas.com/en_id/insights/analytics/machine-learning.html. Accessed November 24, 2016.