Stroke Data Analysis

Nathan Gin

67117388

Stats 170A HW 4

Abstract

Being able to determine whether or not someone had a stroke before they are brought to a hospital allows the doctors to set up and provide the adequate care that the subjects need. There are multiple methods used to help determine whether someone had a stroke or not such as RACE score or Electroencephalography changes. Through testing the accuracy of different models, we were able to see which model best fits the given data. The best fitting models were the logistic regression model with no EEG variables and the KNN model with 7 nearest neighbors who both had an accuracy close to 75%. The worst model, in this case, was the decision tree classifier at an accuracy of only about 45%.

Introduction

The motivation for this study is to help with a pre-hospital diagnosis of strokes in patients so they can receive adequate and appropriate care upon arrival. The data I am looking at consists of 100 patients that were taken into the hospital for care and diagnosis of if they had a stroke or not. I am looking at the association between different clinical variables and stroke status. The clinical variables include RACE (Rapid Arterial Occlusion Evaluation), a tool used to help predict if a patient had a stroke before being admitted to the hospital on a score from 1 to 5, age (in years), LKW (Last Known Well in hours), gender (male or female), and 100 different Electroencephalography (EEG) signals. There are 100 EEG signals taken from a 3 minute period immediately after brain ischemia for each patient. I am testing the accuracy of different models on the data to find the one that performs the best based on the data set given.

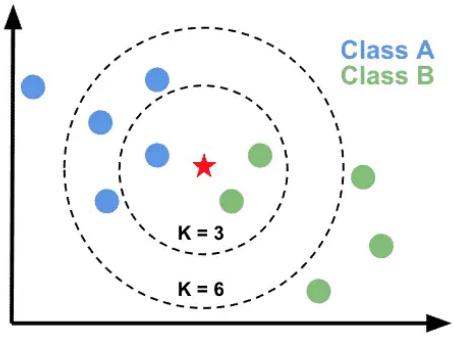
Statistical Methods

In this study, we will be comparing two different logistic regression models and 3 nonlinear classification models all fit using R. These nonlinear classification models include K-nearest neighbors, a decision tree, and Random Forest. The two logistic regression models include one with just the clinical variables and no EEG variables and one with the clinical and all the EEG variables. Logistic regression is the most fitting for this data because we are testing it against stroke status which can only be yes or no. We tested the accuracy of the different models by separating the data into training data and testing data. We used 70% of the data to train the classifiers and the remaining 30% of the data to test how accurate our classifiers were. This accuracy compares the amount the classifier identified correctly from the testing data.

Logistic Regression:

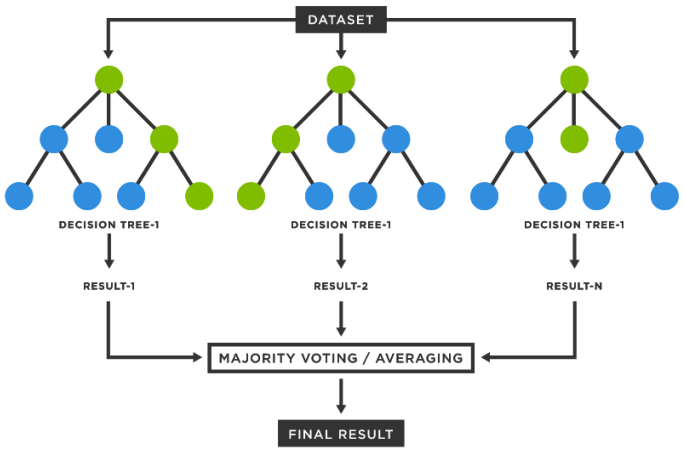
$$log(\frac{P(Outcome)}{1-P(Outcome)}) = \beta 0 + \beta 1X1 + \beta 2X2 + \beta 3X3 + \dots$$

The K-nearest neighbor classifier takes into consideration a certain point and determines what value it should be based on K neighbors from the given point. So if k = 1, it would take on the given value of the nearest neighbor to itself. If k = 9, it would take the 9 nearest neighbors and take on the value that the majority of the values have. In the example on the right, the point we are determining is the star. If k = 3, then it would be labeled as Class B, and if k = 6, it would be labeled as Class A.



The decision tree classifier looks at all of the data and then begins to split it one by one based on the variables within it. For the top of the tree, it may first separate by gender, then by RACE score greater than 2, then by Age greater than 60, etc. This is just one of the many ways a decision tree could be made in this scenario and depending on the route it chooses, it will change what the resulting stroke status value will be. A single decision tree will follow this splitting process until it cannot split anymore or reaches a set depth and at each leaf will be the resulting status for that path. The leaf will represent the probability of yes or no for the stroke status of that path. One benefit of a decision tree is it is easy to visualize and understand but at the same time, it suffers because it can have a high variance for smaller data sets making completely different trees depending on how it is split.

The random forest classifier creates multiple decision trees and finds the average of them all to find the best prediction labels from the training data to use on the testing data. A single decision tree normally has high variance so taking the average of a forest of decision trees can help to return more accurate results.



The image on the left shows the idea of taking multiple decision trees and averaging them to return a more accurate result. Random forests help to solve a lot of the problems with a singular decision tree as it deals with outliers and helps to create a more appropriate average of the trees.

Results

All of the classifier models were fit with the same 70% of the data and tested on the same 30% of the remaining data. All of the results of the tests on the Stroke data can be seen in the table below.

Model	Accuracy (%)
Logistic Regression (No EEG)	76.67
Logistic Regression (With EEG)	65.22
K-nearest Neighbor (K = 7)	75.86
Decision Tree	45.83
Random Forest (Trees = 500)	66.20

Starting with the accuracy of the logistic regression model without the EEG variables, we used the model below to obtain an accuracy of 76.67%. This was the most accurate model in part because it only took into account the most important parts of the data as we looked at in the previous report with no interaction terms. Additionally, logistic regression is a good fit for the data as we are testing binary outcomes of whether the patient had a stroke or did not have a stroke.

$$log(\frac{P(Stroke = 1)}{1 - P(Stroke = 1)}) = \beta 0 + \beta 1 * RACE + \beta 2 * LKW + \beta 3 * Male + \beta 4 * Age$$

The next tested equation was when we included the 100 additionally EEG readings for each patient. This lowered the accuracy of the logistic regression model to 65.22% as it may have been overcrowding some of the other clinical variables that had a larger impact on the results such as the RACE score. Each individual EEG value may not have been a large indicator of whether or not they had a stroke and may have improved in accuracy if we used dimensionality reduction on it.

$$log(\frac{P(Stroke = 1)}{1 - P(Stroke = 1)}) = \beta 0 + \beta 1 * RACE + \beta 2 * LKW + \beta 3 * Male + \beta 4 * Age + \sum_{i=1}^{100} (\beta_i * EEG_i)$$

The KNN classifier was tested using different values for k. After testing different values of k, I found that the most accurate predictor was using 7 nearest neighbors in order to determine the value of k. Although, depending on the value of k, the accuracy went as low as 58%. The accuracy of k=7 was 75.86% which is very close to the logistic regression model and can be a good fit for data when similar values are clumped together. The association between higher RACE scores and older patients is a good example of how this can benefit the KNN classifier when predicting based on neighboring values.

The decision tree did very poorly as expected. This is because a decision tree can be prone to overfitting and having high variance depending on the path the tree took. Once overfit, the accuracy will go down as it will take into account too many variables to the point where it is not useful. This can explain the lowest accuracy of 45.83%.

The random forest classifier does improve the accuracy because it helps to tackle the issues that the single decision tree has. Instead of overfitting and having a high variance, having multiple trees will help to find a better average tree for all of the data. In this case, we created 500 decision trees with 10 variables tried at each split and averaged them out to get the answer. The accuracy was still relatively low at 66.20% which was a surprise as I expected it to be the most accurate. However, compared to the decision tree, it was over 20% more accurate.

Discussion

In conclusion, we can see that the KNN classification model and the logistic regression model without the EEG variables were the most accurate by about 10%. Behind them were the Random Forest model and the logistic regression model with the EEG variables. The worst model was the single decision tree with an accuracy of more than 30% less than the top-performing model.

However, there were many problems with the data as there are only 100 patients. Additionally, missing values when including the EEG variables forced you to either remove columns altogether leaving an even smaller dataset or find a way to fill in the blanks with an appropriate value. This could be the nearest value based on some sort of sorting or an average of the other values. This creates problems in the data but is necessary when creating the different classifiers within R. I took the average of the data to fill in the null values but creates problems when there are missing values that are factors or categorical variables such as gender or RACE score. Depending on the settings of the classifiers or if you used dimensionality reduction on the EEG variables can also lead to different results as shown with different values of k in the KNN model.

Appendix:

Image Citations

Tripadvisor, Jean-Christophe ChouinardSEO Strategist at. “K-Nearest Neighbors (KNN) in Python.” *JC Chouinard*, https://www.jcchouinard.com/k-nearest-neighbors/.

“What Is a Random Forest?” *TIBCO Software*, https://www.tibco.com/reference-center/what-is-a-random-forest.