Predicting Stroke Outcomes

Luca DeJesu, Mahir Morar, JeMarra Rivers, Colin Liu, Chase Dannen

May 9th, 2021

Introduction

Stroke is the leading cause of long-term adult disability and the fifth leading cause of death in the US [1], yet, according to a free recall survey, up to 60 percent of respondents could not name a single risk factor for stroke [2]. There are certain risk factors for stroke that can be categorized as modifiable and non-modifiable. Some non-modifiable risk factors include age and sex, and some modifiable risk factors include hypertension, smoking, and BMI [3]. So, there are many risk factors for stroke, yet plenty of people are unable to identify what exactly can cause an ischemic or hemorrhagic stroke.

In order to help other people identify what can cause a stroke, machine learning methods can be employed to predict incidence of strokes and highlight the attributes that weighed on what was predicted. In order to further strengthen confidence in one method, another can be used with a comparison of results using common machine learning evaluation metrics. Our goal is to train two machine learning classifiers on a stroke dataset that provides 5110 instances of patients that either had a stroke or did not, based on 10 identified risk factors [4]. More specifically, we will employ a Gaussian Naive Bayes classifier, a decision tree classifier, and an ROC curve to compare the two methods. The attributes predicted to affect stroke will be presented, and then the comparison of the methods will follow. By doing these things, we can help others better understand the main risk factors for stroke and provide some confidence in the machine learning algorithms used to reach these conclusions. Additionally, we can take a look at our dataset, and make a prediction as to what attributes may contribute the most. Based on some analysis of the data done later on, our team has a hypothesis that heart disease and age will be strong predictors, and that a decision trees approach should lead us to better prediction accuracy.

Problem Definition

Using a stroke dataset with 5110 instances of stroke based on 10 different recognized risk factors, we will utilize a Gaussian Naive Bayes classifier with standard scaling and a decision tree algorithm in order to predict stroke outcome based on the attributes. An interesting point in the dataset is that it includes several of both of the two main categories of risk factors identified earlier: "modifiable" (hypertension, marriage history, work and residence type, smoking status, bmi) and "non-modifiable" (gender, age). When later analyzing which attributes indicated stroke in the set, we can refer back to these categories and provide some evidence for the prevalence of them. Rather than implement a customized approach, our team decided to utilize the well known Sci-Kit learn library and the Gaussian Naive Bayes classifier provided by them, as well as the DecisionTreeClassifier for the decision trees algorithm. Accuracy of the predictions will be

displayed, as well as graphical representations of the risk-factor prevalence in the dataset prior to classification (the attributes), and finally the two machine learning algorithms will be compared on accuracy. The evaluation metric to compare the two algorithms will be a graphical Receiver Operator Characteristic curve (ROC curve). Our aim is to visualize what can be indicative of strokes based on the data set, predict strokes on this set using two common machine learning algorithms, and compare the accuracy of the two.
The inputs for both algorithms come from the data set and are listed as follows:

- ID: a unique identifier for each stage
- Gender: "Male", "Female" or "Other"
- Age: age of the patient
- Hypertension: 0 if the patient doesn't have hypertension, 1 if the patient has hypertension
- Heart_disease: 0 if the patient doesn't have any heart diseases, 1 if the patient has a heart disease
- Ever_married: "No" or "Yes"
- Work_type: "children", "Govt_job", "Never_worked", "Private" or "Self-employed"
- Residence_type: "Rural" or "Urban"
- Avg_glucose_level: average glucose level in blood
- bmi: body mass index
- Smoking_status: "formerly smoked", "never smoked", "smokes" or "Unknown" (Unknown means that info is unavailable for that patient)
- Stroke: 1 if the patient had a stroke or 0 if not

The output for Naive Bayes:
The Gaussian Naive Bayes classifier will output the accuracy in prediction on the training set, and the testing set. The testing set will be used to compare against the actual set, wherein the actual incidence of stroke will be compared with the algorithms' predicted stroke incidence.

The output for Decision Trees:

- The left of a given node shows the progression of true samples and the right shows the progression of false samples
- Top row shows the value at which the node splits
- Gini is the percentage of samples that go in one direction (i.e. 0 means all samples go in the same direction)
- Samples is the amount of patients that made it to that node to be split
- Value is the split of the samples for true/false

Algorithm Definition

Gaussian Naive Bayes

All Naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable, which in our case is stroke.

$$P(Y \; = \; k \mid x1, x2 \ldots x10) = \frac{P(x_1 \mid Y{=}k) * P(x_2 \mid Y{=}k) \ldots P(x_{10} \mid Y{=}k) * P(Y{=}k)}{P(x_1) * P(x_2) \ldots . P(x_{10})}$$

where K = 1 for stroke, 0 for no stroke, and $(x1, x2 \ldots x10)$ are the 10 attributes in our data set: gender, age, hypertension, heart disease, ever married, work type, residence type, average glucose level, body mass index, and smoking status.

Fig.1: Our classifier will compute predictions using the above formula, after the conditional probabilities and prior probabilities are computed in the training phase.

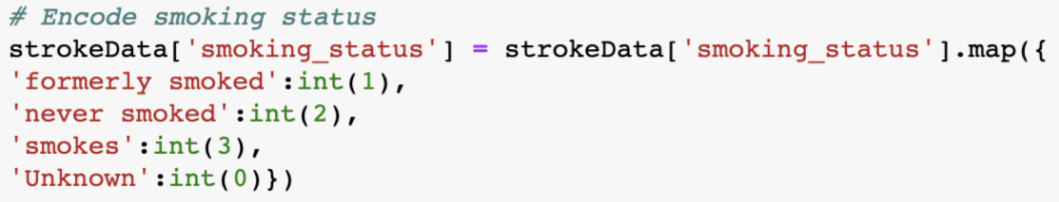
Gaussian Naive Bayes was chosen for this task, as we have continuous inputs. We thus make the assumption that our continuous variables are distributed according to a normal Gaussian distribution. The following formula is how each probability density is calculated for each feature in order to make the assumption they are distributed according to a Gaussian Distribution:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

Fig.2: The probability density of feature Xi given class y (stroke or no stroke)

The parameters in this equation for mean (mu) and variance are estimated using maximum likelihood.

This is how the Sci-Kit learn implementation of the Gaussian Naive Bayes algorithm makes predictions: using the data, and the priors of whether or not stroke occurred.
The training of this algorithm on the training data split involves getting all of the conditional probabilities (the probabilities that an instance takes on a value for a certain attribute) for the attribute values as well as the prior probability for the class attribute, stroke incidence. For the textual attributes, we have encoded them into numbers to conform to what the classifier expects as input. Here is an example of the encoding for smoking status:

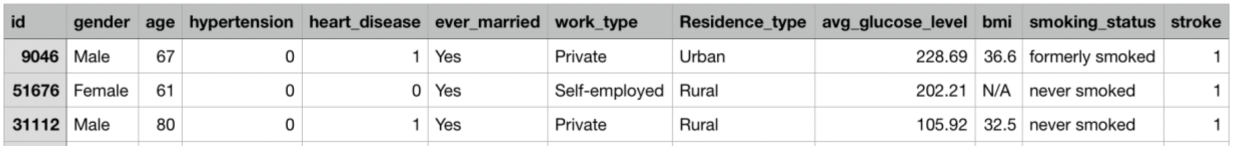


```
# Encode smoking status
strokeData['smoking_status'] = strokeData['smoking_status'].map({
'formerly smoked':int(1),
'never smoked':int(2),
'smokes':int(3),
'Unknown':int(0)})
```

Fig.3: Smoking status encoded into 4 distinct integers

Finally, for the predictions, the test split of the data set will be fed into the classifier using the probabilities calculated by the training set as mentioned above. The split for testing will consist of 1533 randomly selected examples.

To get a better understanding of how exactly one of our data set instances will be processed, we can trace an example. Here is the top of our dataset, before encoding:



id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
9046	Male	67	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
51676	Female	61	0	0	Yes	Self-employed	Rural	202.21	N/A	never smoked	1
31112	Male	80	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1

Fig.4: The first three instances of the dataset. Note that "N/A" and other nulls were filled with the mean of all values afterward.

We can take the first instance as an example, namely identification number 9046 in Fig.4, as an example of the test set (whether or not it was actually selected by the test set is not known, it is a random selection). At the point of testing classification, the prior probability of stroke or not stroke has been calculated, and the conditional probabilities for each attribute have been computed. The following is how the conditional probabilities are calculated for the attribute value of an instance, 't':

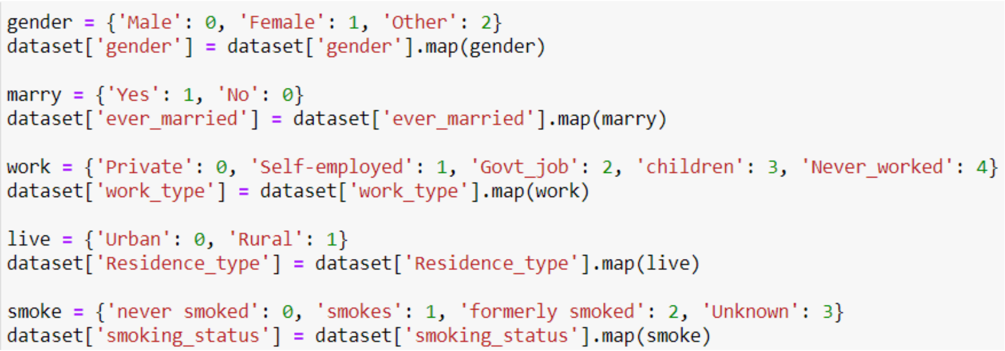
$$\hat{P}(t|c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}},$$

Fig.5 The formula for conditional probability for attribute 't'. It is simply the number of distinct times this occurred in the training set (Tct) divided by the sum of all values occurring (the total instances V)

Now, using these probabilities, the algorithm will classify this instance as either 1 for predicted stroke, or 0 for no predicted stroke, based on a new example seen by the test split of the dataset. In this instance, if the algorithm predicts a '1', then it correctly classified it, as this man had a stroke. To summarize, the main question being asked and predicted upon in this instance is the following: given that a person is male, aged 67 years, does not have hypertension, does have heart disease, was married, works privately, resides in an urban neighborhood, has an average glucose level of 228.69, a BMI of 36.6, and has smoked previously, will they have a stroke?

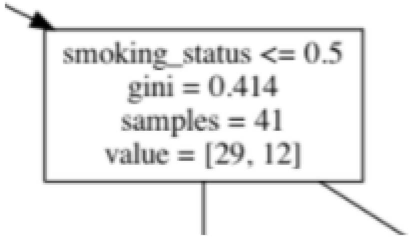
Decision Trees

This is how the Sci-Kit learn implementation of the Decision Tree algorithm makes predictions: using the data, and the priors of whether or not stroke occurred. Before the training data is passed through the Ski-Kit learn implementation however, the data of many attributes are encoded into numerical values similar to the above Naïve Bayes process.



```
gender = {'Male': 0, 'Female': 1, 'Other': 2}
dataset['gender'] = dataset['gender'].map(gender)

marry = {'Yes': 1, 'No': 0}
dataset['ever_married'] = dataset['ever_married'].map(marry)

work = {'Private': 0, 'Self-employed': 1, 'Govt_job': 2, 'children': 3, 'Never_worked': 4}
dataset['work_type'] = dataset['work_type'].map(work)

live = {'Urban': 0, 'Rural': 1}
dataset['Residence_type'] = dataset['Residence_type'].map(live)

smoke = {'never smoked': 0, 'smokes': 1, 'formerly smoked': 2, 'Unknown': 3}
dataset['smoking_status'] = dataset['smoking_status'].map(smoke)
```

Once that is completed and the data is placed into the appropriate vector variables, the training of this algorithm participates in a loop process, starting with the determination of the "best" decision attribute. The algorithm determines which attribute has the largest contribution in the determination of the target function, creating new descendants of the established node which repeats the determination process with the remaining samples that have arrived at the node. In the visualization on the decision tree, this attribute is noted at the top of the node where the name of the attribute can be seen along with a comparative function, where the subsequent connected nodes represent the sample pools that are registered as true or false of that function. The following is an example of one of the stroke decision tree's node:



Following the attribute and its comparative function are other variables that assist with the visualization of the training data's split and taken paths. The gini value directly under the attribute notes the percentage of samples that would go in one direction. For example, a gini value 0.0 represents that all of the samples received the same result but a gini value of 0.653 would represent that approximately 65 percent of the samples that reached the node would receive the same result. Below the gini value is the sample count that proceeded to the current node followed by the value variable which shows how that sample count is divided between true or false according to the nodes' function.

Given a data example, the algorithm neatly visualizes the path to determining the data's classification. For example, utilizing the above Naïve Bayes mentioned male who is aged 67, does not have hypertension, does have heart disease, was married, works privately, resides in an urban neighborhood, has an average glucose level of 228.69, a BMI of 36.6, and has smoked previously, will they have a stroke? With the visualized results of the decision tree algorithm, we can trace through 7 internal nodes to a leaf node that classifies the individual as a person who will likely have a stroke.

3. Experimental Evaluation

Methodology

Our methodology will be based on two main logical schemes: 1) we would like to predict stroke based on certain risk factors and view the results of these predictions using two commonly used machine learning algorithms in the healthcare industry, and 2) we would like to compare which of the algorithms predicted more accurately using machine learning evaluation metrics.

Before evaluating the outcomes of the two classifiers, we can take a look at some of the true data. This can give us an introduction to the examples and their characteristics that will be fed to the algorithms, as well as outline some predictions for which attributes will be bigger factors in stroke prediction.

After this introductory look at the data set, we can perform the predictions for Naive Bayes, decision trees, and finally compare the methods against each other with a receiver operating characteristic curve. For Gaussian Naive Bayes, we will see how many strokes were predicted compared to the actual number of strokes in the entire set. We will compare the false positive rate and false negative rate and highlight the distinctions' importance, and we will outline the overall prediction accuracy of the model. For decision trees, we will take a look at which attributes were the best at separating the data, and report those attributes, which can be compared to the actual incidence of the attributes and the corresponding stroke status. In the evaluation portion, the area under the ROC curve can give us some indication as to which classifier could predict stroke with a better ratio of true positives to false positives, for varying thresholds.

Results

Here we can show the incidences of those predicted to have stroke, what attributes/risk factors were the most prevalent. We compare the results of both algorithms and highlight what our predictions outline.

Since Naive Bayes assumes conditional independence of the attributes [5], the resultant predictions make the strong assumption that each features value is independent of any other feature value - the 'Naive' assumption.

Firstly, we can examine the dataset in greater detail. Johns Hopkins Medicine lists 3 prevalent risk factors that likely contribute to stroke that exist in our dataset: hypertension, smoking, and heart disease [6]. Let us examine how these risk factors contributed to stroke outcome in our dataset. We can later see how our decision tree discriminates and draw comparisons with the actual data.

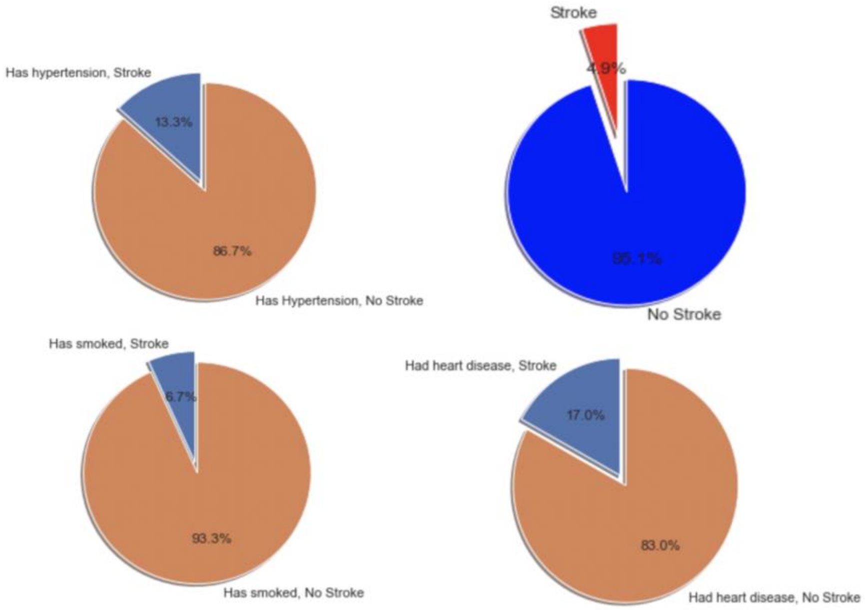
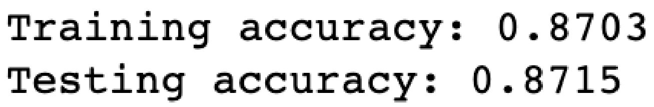


Fig.6: The pie charts above: top left is incidence of having stroke or no stroke given that hypertension exists, top right is actual overall incidence of stroke in the dataset, bottom left is incidence of having stroke or no stroke given that person has smoked currently or previously, and bottom right is incidence of having stroke or no stroke given that the person had heart disease.

Based on our dataset, these three attributes each show an increase in percentage of populations having a stroke. The largest of these in this set is the existence of heart disease; of those who had heart disease, 17 percent had a stroke, versus the overall incidence of stroke being 4.9 percent.

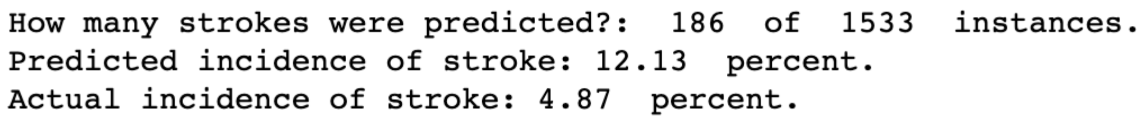
Now we can train a Gaussian Naive Bayes classifier and make stroke predictions, as outlined above in section 2. We do so, but before reporting the accuracies, we standardize the training and testing data splits, using Python's StandardScaler. We do this because some of our features are measured at different scales and thus do not contribute equally to the fit and learned function, which could result in additional bias. The results obtained on the training set and the testing set, after standardizing:



```
Training accuracy: 0.8703
Testing accuracy: 0.8715
```

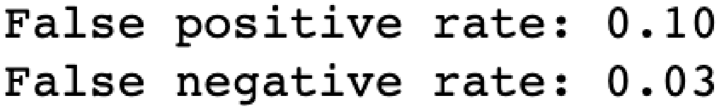
Both end up classifying strokes with about ˜87 percent accuracy, with the test set actually being slightly higher in prediction accuracy.

One interesting finding we can observe based on the results is found within false positives. For our Gaussian Naive Bayes estimator, we find that more strokes are predicted (percentage-wise) than actual incidence.



```
How many strokes were predicted?:  186  of  1533  instances.
Predicted incidence of stroke: 12.13  percent.
Actual incidence of stroke: 4.87  percent.
```

We are interested to see the false negative rate, as in this context, predicting a false positive is less dangerous than a false negative in some ways: if a person may be on a path to a stroke, they could face danger if no stroke is predicted, yet they are likely to have a stroke. To examine this model on how accurate it was at predicting false positives, we can highlight the rate of our classifier predicting a false positive and the rate of the classifier predicting a false negative:



```
False positive rate: 0.10
False negative rate: 0.03
```

Now, we can take a look at our decision trees approach. First, let's make a decision tree of depth 3 and classify a test set with it.

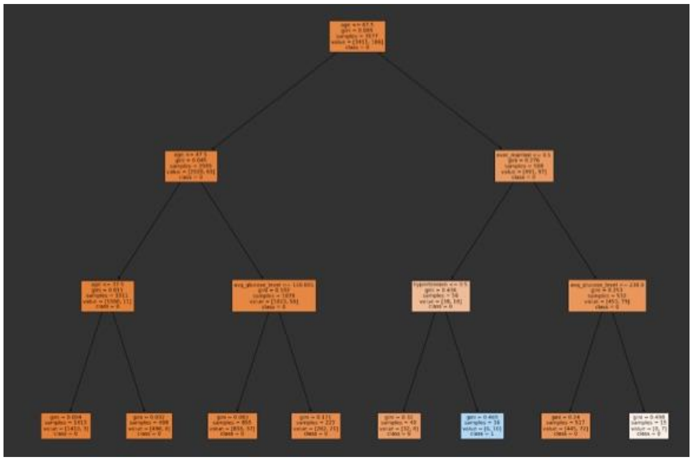


Fig. 7: A decision tree of depth 3, based on our test set. The right of the tree indicates false, the left of the tree indicates true. So, for the first node, "age <=67.5" going right means the person is older than 67.5.

Above is the decision tree with a depth of 3. Originally, the decision tree we had was too large to easily understand what story the data is telling us. As for the top 3 risk factors, the first risk factor for a stroke is age. The older an individual is, the higher their risk for a stroke gets. If someone is older than 67.5 years old, the chances increase. After that, the second risk factor is glucose level. Next would be the average glucose level. Having a high level of glucose increases the probability of an individual having a stroke. Finally, marriage is the third risk factor in determining if someone will have a stroke. If someone is married, that will play a huge role. For this specific decision tree, the shading of each node represents the confidence. The orange represents not that much of a chance of having a stroke. The light orange represents about a 50% chance of having a stroke and the blue indicates a very high confidence of having a stroke. From this decision tree, we can say with confidence that a married, 67.5 years old, With hypertension above 0.5 will likely have a stroke.
And the overall accuracy of our decision tree result:

```
Overall accuracy of the decision tree accuracy (on the test split): 0.9446
```

We compare the methods using the ROC curve in machine learning evaluation metrics. The resulting graphs can be seen below.

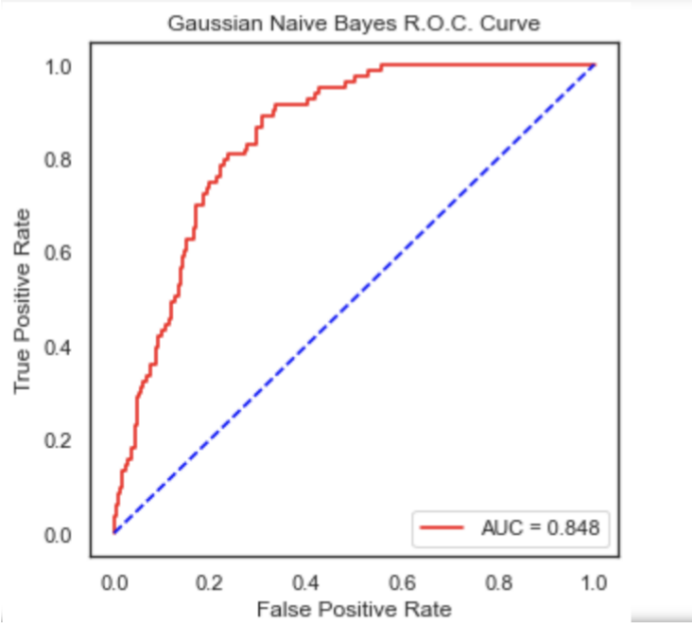


Fig.8 The Receiver Operating Characteristic Graph for the Gaussian Naive Bayes Classifier

And, we have the ROC curve for the decision trees algorithm, of depth 3:

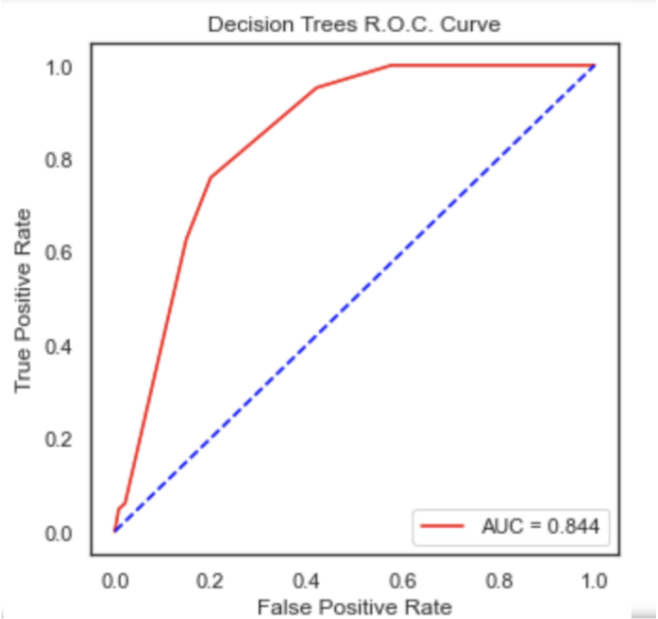


Fig. 9: The Receiver Operating Characteristic graph for our decision trees approach.

The areas under the respective curves are nearly the same, with Gaussian Naive Bayes only outperforming the decision trees by 0.004. The results of our evaluation suggest that either algorithm can predict roughly the same. With standard scaling, it is possible that our Naive Bayes approach was made slightly better by the ROC curve metric. However, it is difficult to recommend one over another only based on this metric, as both give about the same area under the curve.

As for the teams' hypothesis, it is slightly supported and slightly contrasted by our findings. First, the decision trees, by the evaluation metric used, did not significantly provide better results for varying thresholds. Second, the attributes our team imagined would be good predictors were good predictors in our decision tree approach. As seen in the image of the depth-3 tree, the age attribute is the first attribute to split on based on the gini index. With a gini of 0.089, this attribute (age being above or below 67.5) represents a split of almost no inequality (almost a perfect split). Heart disease was not selected in the depth 3 tree, however, but in depth-5 tree it is included on level 5

Discussion

The idea that these machine learning algorithms are capable of making moderately accurate predictions based on risk factors is supported by resulting accuracies of ~ 87 percent on our Naive Bayes classifier test set. For decision trees, our classifier with a max depth of 3 had an accuracy of 0.9446. When we allow a deeper tree, we observe the following results: a depth 5 tree gives us a resulting accuracy of 0.9432, a depth 7 tree gives an accuracy of 0.9393, and a depth 9 tree an accuracy of 0.9361. These steady decreases in accuracy may show us the result of overfitting, so we kept the tree at the highest accuracy, depth 3. Also important to note, the area under the curve for a tree of depth 5 drops to 0.82, indicating this classifier may already be less accurate. The area under the curve for a depth 9 tree drops all the way to 0.646. Medical research and current findings on stroke incidences suggest certain modifiable and non-modifiable risk factors for strokes. The main attributes revealed by research in the subject involve those listed above: . Our model suggests that the risk factor of being 67.5 years of age or above is a solid indicator, with a Gini index of 0.089 indicating that the split provided almost no inequality.

Related Work

A good comparison for our Naive Bayes approach can be found in the report “Classification algorithms for predicting sleepiness and sleep apnea severity” by Eiseman et al [9]. In this report, use three algorithms for comparison on abnormal sleep features according to polysomnography tests: support vector machines, k-nearest neighbors, and Naive Bayes (continuous features, like BMI, were also used). However, for their result, they classify predictions on the abnormal sleepiness score of the Epworth sleep scale (0-11 is normal, 11-24 is abnormal) based on these features, rather than a binary classification as in our report. Also important to note is the bad sensitivity in their findings compared to ours: this report had a sensitivity/true positive rate of 16.7 percent, while our report was able to provide a true positive rate of 84.8. This could be an indicator that our report provided better prediction accuracy for those that had stroke, whereas their report was only 16.7 percent able to classify the positives correctly. This report does, however, conclude that the utility of their sleepiness scale predictions (the Epworth sleepiness scale) are questionable based on their results.

For comparison to our Decision Tree implementation we decided to look at a work by Junjie Liang titled *Predicting borrowers’ chance of defaulting on credit loans* [10]. The project drew attention due to the similarities in data sets, specifically the continuous attributes and the discrete classifier, yet their decision to implement a random forest ensemble method. The project was provided with a labeled dataset of 150,000 borrowers and tasked with labeling and additional 100,000 borrowers as expected defaulters or not according to 10 credit risk factors. Liang reasonably chose a decision tree based learning algorithm due to the size of the data set, but differed in their decision to specifically use a random forest algorithm, which is often noted to outperform and correct the overfitting of typical decision trees. This decision resulted in good predictions on whether a borrower would default, generating an AUC score of .867 while ours AUC's revolved around .85.

Future Work

There are many risk factors for stroke, and there are many machine learning algorithms well suited for the task of making predictions based on these risk factors. Some of the risk factors for stroke that were not highlighted in this dataset include hyperlipidemia, diabetes, physical inactivity, and race. In order to have a fine-tuned understanding of everything that can lead to either an ischemic or an hemorrhagic stroke, more reports should be done using different attributes. There could be other attributes that our decision tree would better be able to discriminate based upon, and thus learn better on. A solution here would be to train these classifiers with various different datasets, and see how the decision tree discriminates. For Naive Bayes, the biggest shortcoming may be the fact that the assumption is so strong: what if the presence of one attribute does have an impact on another - such as hypertension impacting the chance of obesity? One study comparing seven machine learning algorithms found that a random forest classifier outperformed both Naive Bayes and decision trees [8].

Another direction this report could have gone in is reviewing the two broad categories of stroke risk factors (modifiable risk factor versus non-modifiable risk factor). One grey area of this dataset is the distinction between what is truly modifiable and what is not; are hypertension and heart disease modifiable, or not? Datasets with clearer distinctions, i.e. genetic risk factors versus lifestyle habits, may serve to better highlight how much control a person has over their chance of getting a stroke. The classifiers may also find better discriminatory power from different genetic risk factors that were not highlighted in this dataset.

For more results that may lead to better prediction, more approaches with different algorithms should be taken. There are many other applicable machine learning algorithms for this data, such as deep learning networks that could reduce the misdiagnosis rate of stroke [7]. In addition to further validating machine learning predictions, different algorithms with weaker assumptions (unlike the strong conditional independence assumption of Naive Bayes) may be more useful.

Conclusion

For both algorithms, the overall prediction accuracies exceeded 85 percent: Gaussian Naive Bayes with Standard Scaling provided ~87 percent accuracy on the test set, whereas decision trees pruned to a depth of 3 provided accuracy of 94.4 percent. Based on these results, our initial hypothesis is supported slightly: the decision trees provided greater accuracy at depth 3, even when the data is normalized in Naive Bayes. However our prediction on which attributes would affect stroke most is only partially supported: age indeed was important, with the highest Gini index being found on the split of age of 67.5 years old (0.089). However, heart disease was not found to be a splitting attribute in the decision tree of depth 3, indicating that the split of inequality was lower for this attribute. Even though prior data may suggest that those who had heart disease may have an increased stroke chance (according to our data in the pie chart representation), the predicting strength based on this attribute was not firmly supported by our report.

We could say that the attributes found by our classifier could be tested again with other attributes from other sets, and cross matched to see which of the attributes was the most prevalent for those that had strokes.

One way to proceed in future reports is to run these algorithms with new datasets. As mentioned earlier, there are many identified risk factors for stroke, and some were not represented here: diabetes, aneurysm history, and race are among a few [1]. So, it would be worthwhile to re-train and re-test our classifiers on a broader range of data and cross-validate some attributes potentially. Also, different methods can be used, such as support vector machines or k-nearest neighbors. As seen in the Eiseman (et al) paper, these methods were strong predictors for classifying sleep apnea or no sleep apnea, a binary classification task that somewhat parallels our task. Our report can be one more piece of information as to whether or not classifying strokes based on Naive Bayes estimators or decision tree estimators is a viable option. Also, it can serve to be classification data for these 10 attributes.

Bibliography

[1] Boehme, A., Esenwa, C. and Elkind, M., 2021. Stroke Risk Factors, Genetics, and Prevention. Available at: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5321635/ [Accessed 10 May 2021].

[2] Nicol, M. and Thrift, A., 2005. Knowledge of risk factors and warning signs of stroke. Vascular Health and Risk Management, [online] 1(2), pp.137-147. Available at: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1993942/> [Accessed 10 May 2021].

[3] Boehme, A., Esenwa, C. and Elkind, M., 2017. Stroke Risk Factors, Genetics, and Prevention. Circulation Research, 120(3), pp.472-495.

[4] Kaggle. 2021. Stroke Prediction Dataset. [online] Available at: <https://www.kaggle.com/fedesoriano/stroke-prediction-dataset> [Accessed 10 May 2021].

[5] Mitchell, T., 2009. [online] Available at: <http://www.cs.cmu.edu/~tom/10601_sp09/lectures/NBayes-1-28-2009-ann.pdf> [Accessed 10 May 2021].

[6] The Johns Hopkins University. 2021. Risk Factors for Stroke. [online] Available at: <https://www.hopkinsmedicine.org/health/conditions-and-diseases/stroke/risk-factors-for-stroke> [Accessed 10 May 2021].

[7] Liu, T., Fan, W. and Wu, C., 2019. A hybrid machine learning approach to cerebral stroke prediction based on imbalanced medical dataset. Artificial Intelligence in Medicine, 101, p.101723.

[8] Sakr, S., Elshawi, R., Ahmed, A., Qureshi, W., Brawner, C., Keteyian, S., Blaha, M. and Al-Mallah, M., 2017. Comparison of machine learning techniques to predict all-cause mortality using fitness data: the Henry ford exercIse testing (FIT) project. BMC Medical Informatics and Decision Making, 17(1).

[9] Eiseman, N., Westover, M., Mietus, J., Thomas, R. and Bianchi, M., 2011. Classification algorithms for predicting sleepiness and sleep apnea severity. Journal of Sleep Research, 21(1), pp.101-112.

[10] Liang, J., 2021. Predicting borrowers' chance of defaulting on credit loans. [online] Stanford. Available at: <http://cs229.stanford.edu/proj2011/JunjieLiang-PredictingBorrowersChanceOfDefaultingOnCreditLoans.pdf> [Accessed 10 May 2021].