Section: Goals and Hypotheses

The goal of our script is to identify the attributes that affect a person's risk of stroke. With that we can then focus on attempting to cluster the attributes to find the patients with these attributes that most effectively might yield a successful predictor of future strokes. Since strokes are preventable, the people who fit into a highly predictive cluster could be alerted that they are at a high risk for stroke. Then they could be informed of any preventative steps they could take to decrease this risk. Based on our preliminary research, we hypothesize that people who have any of the following are at a higher risk for stroke: history of hypertension or heart disease, a history of or a status of currently smoking, a high average glucose level and a high BMI. Furthermore, if they have more than one of these risk factors, then their risk for having a stroke increases even more. We will graph each of these risk factors individually to compare the number of patients who had a stroke with those who did not to be able to reinforce our preliminary research findings. We will then be able to use machine learning to pass these different attributes to the program which will return if it thinks that a person is likely to have a stroke or not.

It is important to remember that only 249 patients in this study had a stroke, compared to the almost 5,000 who did not have a stroke, so coming to any significant conclusions based on this dataset might be hard. That is only a small percentage of patients who had a stroke. But we can compare the lifestyles of the patients within that group to try to come to conclusions about potential risk factors.

Biased Article

Why We Need Accountable Algorithms by Cathy O'Neil

Algorithms separate people into different groups of "winners and losers", based on factors such as class, gender, race. Why is that? It is because algorithms are trained. They are trained on curated historical data, historical information of successes and failures. They examine data, but there are two major problems with data. Data can be very messy, so one has to make sure it is clean and relevant because issues such as spelling errors or vague information can drastically change how the data is read. Another issue with data is that information is not equally distributed. People in the minority and people on welfare or people with criminal records will have more data in the system then others. Recidivism risk scores for example are used by courts to compare a criminal defendant's profile to another's to see if they are expected to return to prison. A higher score can equate to a longer term. Criminal defendants have no way to protest or understand how their scores are determined. Just one example of how algorithms compare people as just data points, and not as human beings. It's also important to keep track of the trade-off between the number of false positives and the number of false negatives produced from data algorithms. In an ideal world these trade-offs would be made clear and errors would be kept track of, O'Neil believes these errors are not given the attention they should be.

Our history is overwhelmingly filled with racism, sexism and xenophobia, so O'Neil states we must assume algorithms inherently contain the same unethical biases. Data science is just codifying these prejudices, making them less visible to the human eye, but behind the curtain, they're more apparent in society than ever. Ethical nondiscriminatory constraints are expensive. It's expensive to add layers and to keep monitoring layers of the algorithm. No one wants to be the company at a competitive disadvantage when it comes to profit margins, so they all use similar algorithms that are inherently prejudiced. O'Neil believes that unless there are standards for anti-discrimination and fairness laws, we will not be able to move forward into a ethically-just and fair world.

Citation:

O'Neil, C. (2017, August 7). *Why We Need Accountable Algorithms*. Cato Unbound. https://www.cato-unbound.org/2017/08/07/cathy-oneil/why-we-need-accountable-algorithms.

Data Analysis: Using the Classification Learner to Develop a Stroke Prediction Model

At this point, the strokedata_prepped table was used to develop a machine learning model. The entire table was passed to the application. This was done with the knowledge that the application is capable of performing a self-assessment of itself. The model application was instructed to separate 15% of the data in "strokedata_prepped" to use as an assessment tool. As models were developed, this randomly separated data was used to assess the accuracy of each model iteration.

Once again, the medical risk is based on smoking history, heart disease, hypertension and the scores are as follows:

- No risk factors (i.e. not a smoker, no heart disease, and no hypertension) === 0
- 1 risk factor === 1
- 2 risk factors === 2
- All risk factors === 3

Now, some brief additional context to outline what was provided to the classification learner. The data table mentioned above was the only data provided. As just noted, a section of this data was separated out of assessment. The “gender”, “age”, “work_type”, “avg_glucose_level”, “bmi”, and “medical_risk” features were all provided as “predictors” - the data being used to predict which classification a subject should receive. The “response” that the model was trying to classify subjects into was the “stroke” column of the table.

With little experience and understanding of the classification learner models, the data was used to train a couple functions that were capable of handling the type of information provided. Given the mixture of categorical and numerical data, there were a limited number of models capable of properly using both. Also, options may have been limited because principle component analysis (PCA) was enabled, which will be discussed more shortly.

The “All Quick-To-Train” option was used to train three models that could handle the data. All three were variations of decision tree models. With limited experience, it was challenging to understand exactly what options to select to get the best modeling results. Training more than one model was recommended and was useful in showing how slightly different models yield different outputs from the same data. Additionally, the decision was made to enable PCA. It was slightly unclear if there would be any repercussions of this decision (since PCA does not work with categorical features), it was an understood concept that would seemingly simplify the relationship of data to the classification output. The only downside of enabling the feature was a reduced understanding of how the model related the inputs and outputs. However, the model functionality was already a bit of a mystery, so modifying the analysis process to include PCA did not seem unreasonable. The use of the categorical data, which was a concern when deciding how to prepare models, seems to have not been an issue since the model descriptions reported “All 2 categorical predictors are used in the model. PCA is not applied to categoricals.”

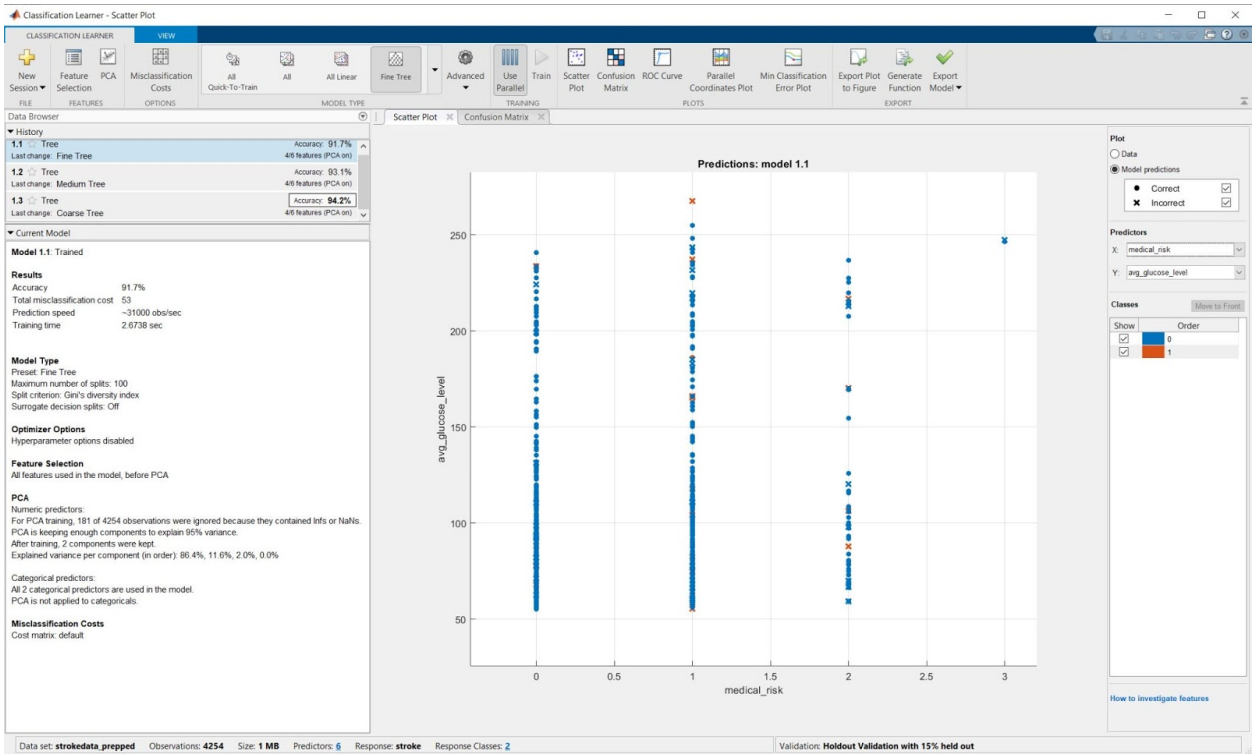
The data was provided and options desired for our model were selected. Now, the provided data was used to train the models with our data with the goal of predicting the stroke response data. According to MATLAB the accuracy is high, for all three tree models (91.7% Fine Tree, 93.1% Medium Tree, 94.2% Coarse Tree). At first glance, it seemed the data yielded positive results for our work. Perfect prediction would have been great, but given the basic data provided to the model 90% seemed exceptional. The next step was looking at the assessment tools for each model that provided more context about these accuracy percentages. This was where it became clear that the models trained were not quite as valuable as the accuracy number had presented.

The assessment information of each model will be explained in more detail. In short, the models predicted non-stroke patients very well, but generally failed to actually predict strokes. The Fine Tree model, which had the lowest accuracy, predicted the most strokes from assessment data set, but it was only 3 patients who fell into this category among the more than 600 subjects that were used for assessment. While looking through the models and assessing them, which will follow this, was valuable, it was quickly apparent that the general conclusion from this work was that the data available was not appropriate for trying to predict if a person is likely to have a stroke.

Intuition might be “predicting someone to have a stroke from lifestyle choices alone would be much too convenient” to not have been discovered already. Whether or not this is true, the failure in this case seemed to be far more dependent on the data used. As noted early on in the data exploration, there were not many patients in the study who suffered a stroke. There was only a small chance that the data available could be used to decisively group these few patients from the rest of the group of non-stroke patients which had significantly more room for variation. It might be unlikely that the risk factors provided here could ever be used to consistently predict strokes. Strokes are a complex occurrence in the body, and the data analyzed here is not even directly linked to the response itself. With a significantly larger dataset, it might be possible to find a general cluster of stroke patients according to the features assessed in this work. Even if the accuracy of the model was much lower, it would still be a

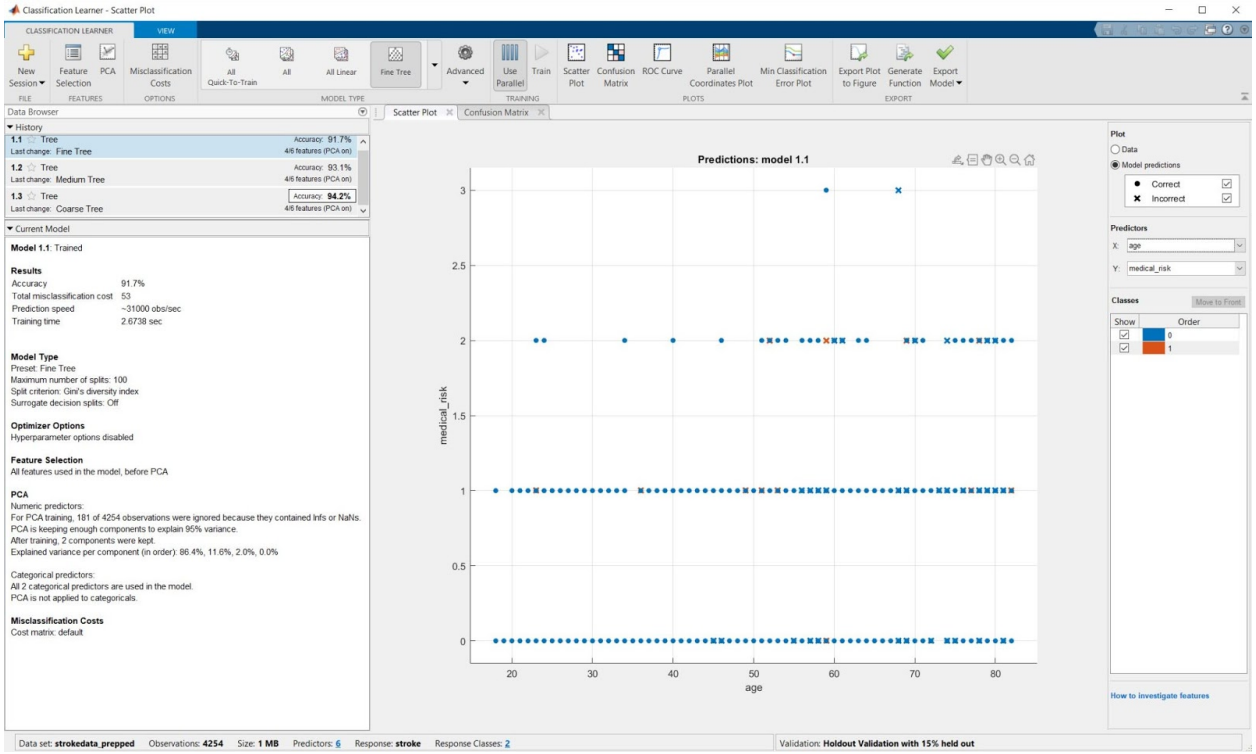
valuable tool for identifying a significantly smaller subset of the general population that should consider making changes to their lifestyle for their health. Thus, the conclusion for this set of data is more inconclusive than failure, and the approach might be valuable with more information.

After addressing the experience of using the classification learner overall, the intricacies of assessing the models will be examined in more detail.

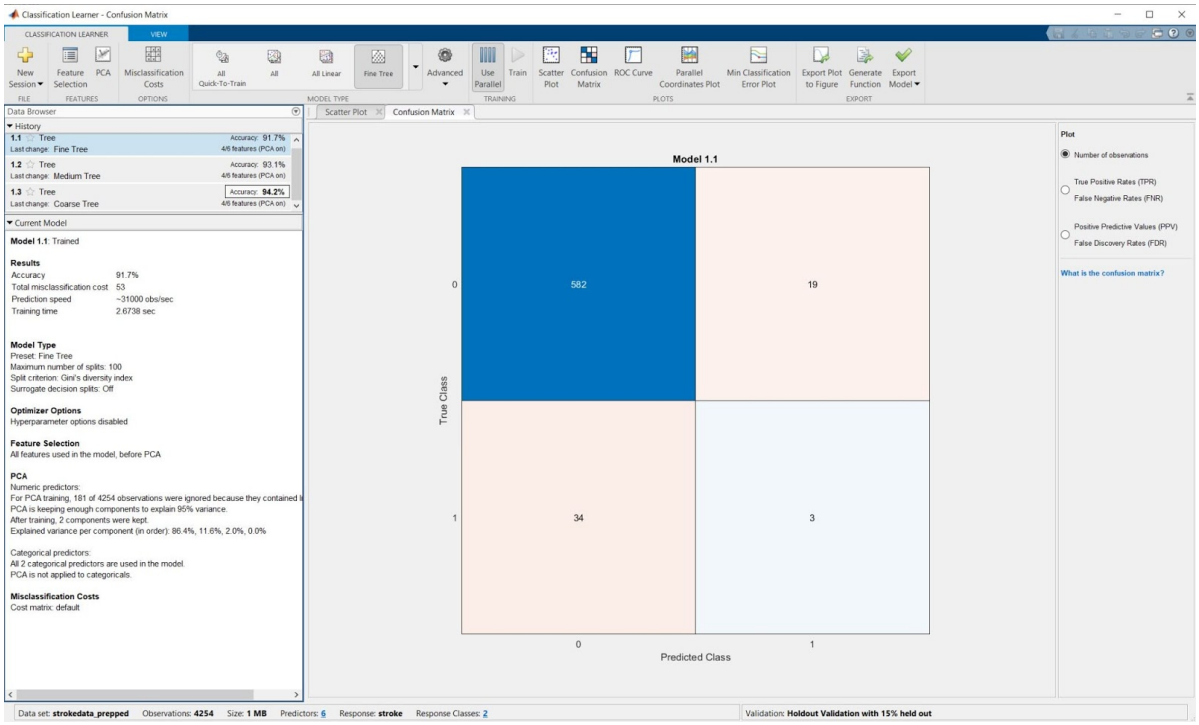


The plot of medical risk by the average glucose level was used to try to predict if a patient gets a stroke or not, since smoking, heart disease, hypertension and elevated glucose levels are believed to cause strokes. The screenshot above is the predictions model based on those factors. A red mark, or a 1, is indicative of someone having a stroke. Further, a red ‘o’ is a correct prediction, meaning the patient was expected to have a stroke and in fact did have a stroke, while a red ‘x’ is an incorrect prediction, meaning the patient was expected to have a stroke, but did not. A blue mark, or a 0 is indicative of someone not having a stroke. A blue ‘o’ is a correct prediction, meaning the patient was not expected to have a stroke and did not have one, while a blue ‘x’ is an incorrect prediction, meaning the patient was not predicted to have a stroke and still had one. Based on our research, the points where the medical risk score and the average glucose level are both higher, we would expect to see more occurrences of strokes, or red ‘o’s. That is not what we see.

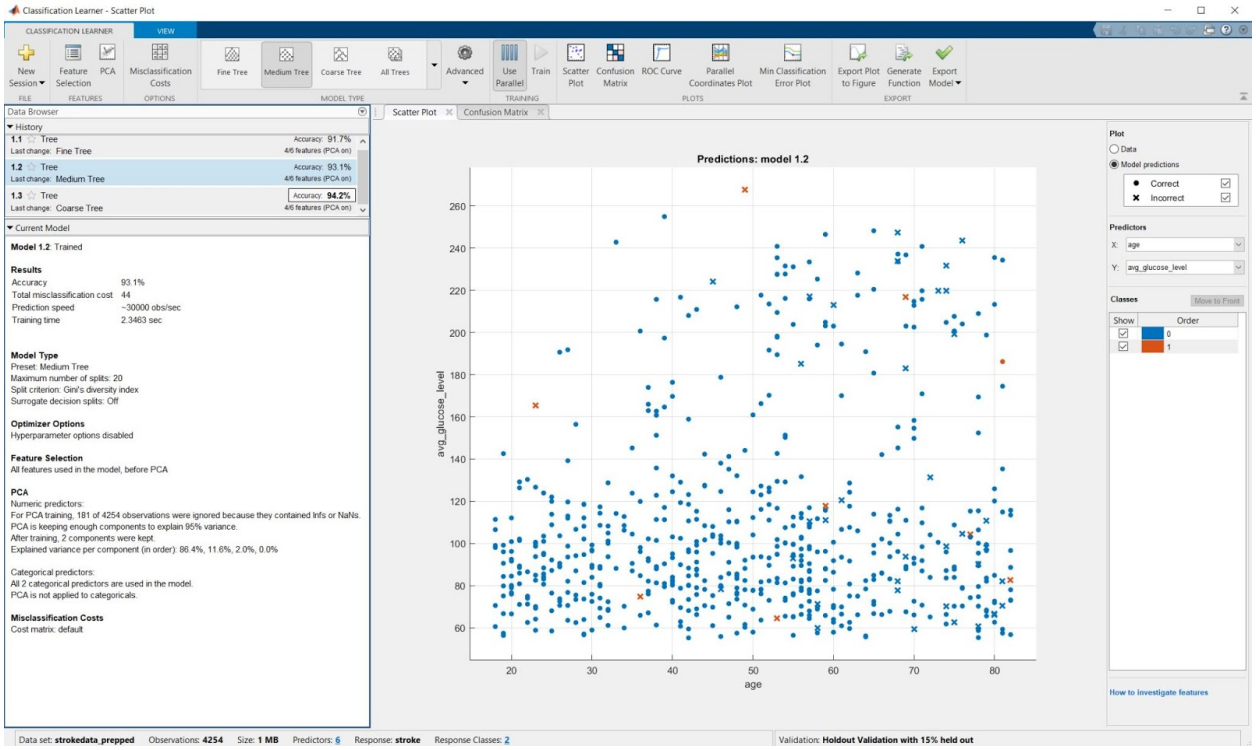
We then graphed medical risk by age, shown below, to see if being at an older age and having a higher medical risk would lead to more strokes, as one would predict based on our preliminary research. Once again, we saw a MATLAB accuracy in the 90th percentile. However, visually it does not look to be that accurate.

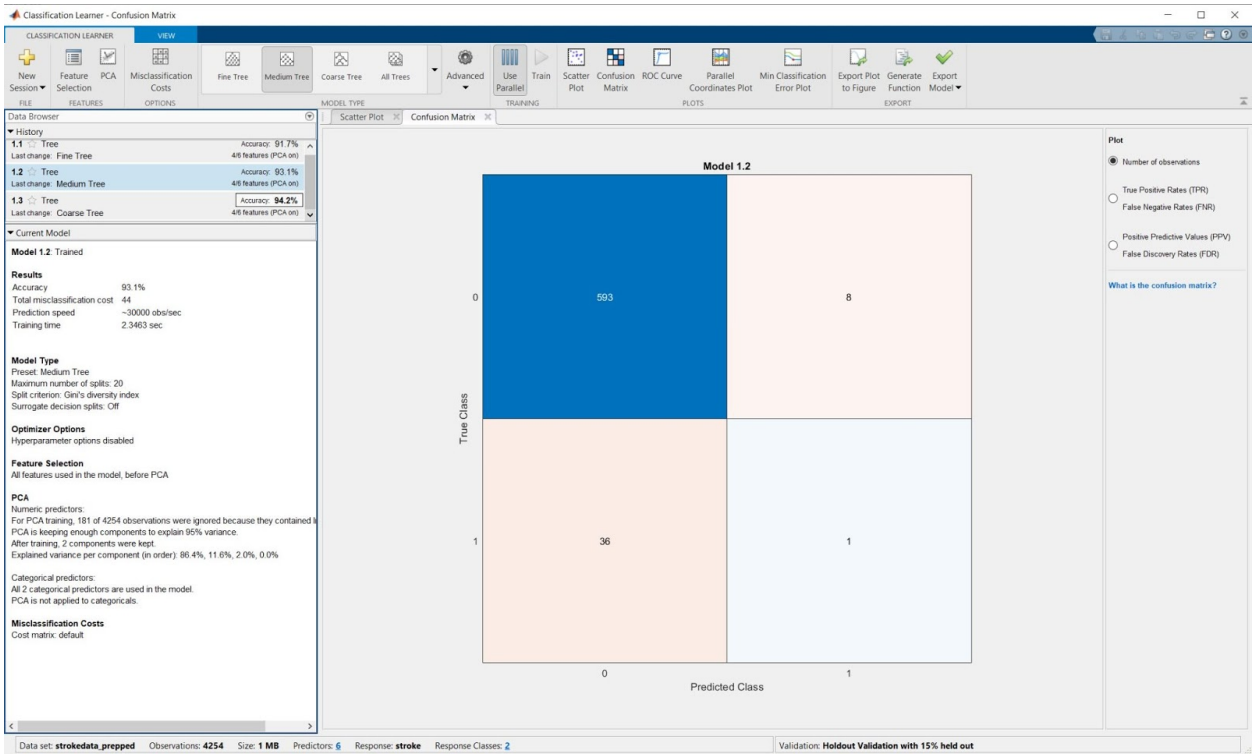


We wanted to see if we could easily quantify visually those results above. In order to do that we looked at a different plot type. We looked at the corresponding ‘Confusion Matrix’ (shown below) which displays the number of correct predictions, i.e. predicted class: 0, true class: 0, 582 predictions, and predicted class: 1, true class: 1, 3 predictions. It also displays the number of false positives, where a patient was predicted to have a stroke, but did not, predicted class: 1, true class: 0, 19 predictions, and the number of false negatives, where a patient was predicted not to have a stroke, but did, predicted class: 0, true class: 1, 34 predictions. This shows that even though according to MATLAB there is a high accuracy, the numbers of false positives and false negatives is quite high compared to the number of correct positives. This indicates that the model did not accurately predict the number of people who would have a stroke.

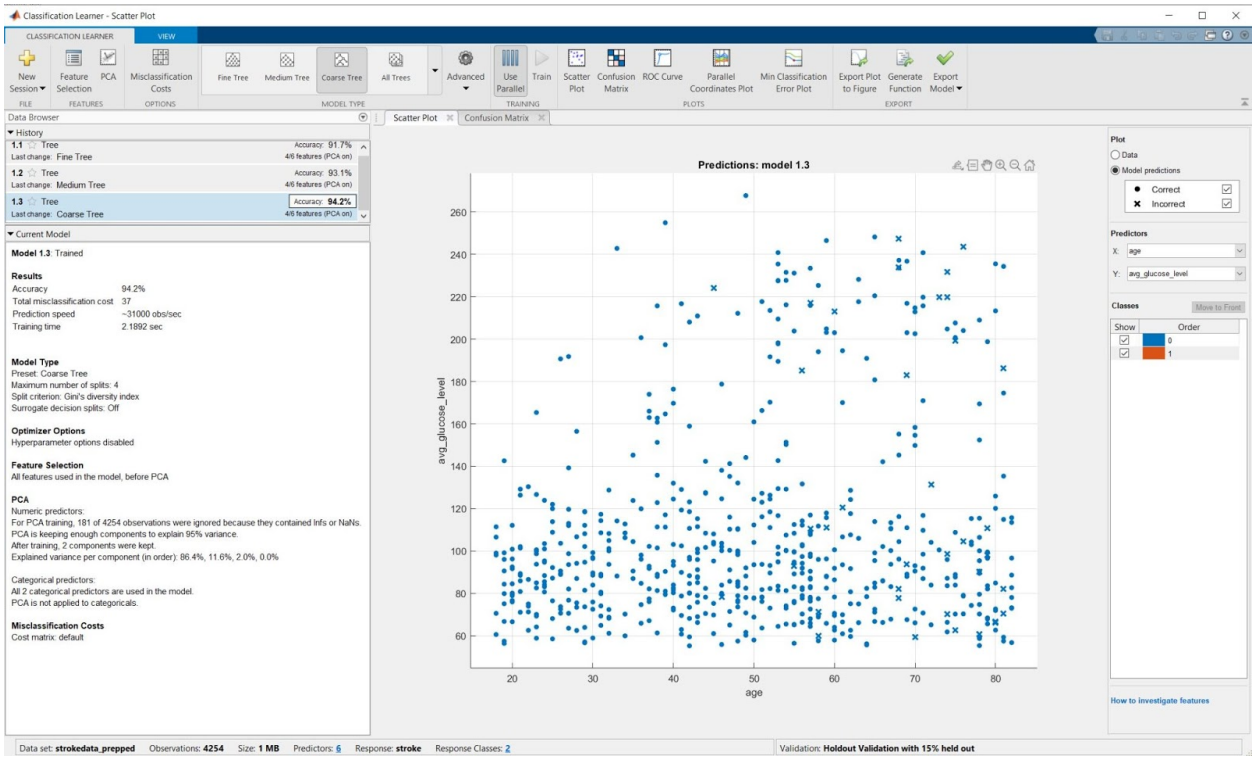


We next plotted average glucose level by age to determine how well those factors would predict stroke. We would predict, based on our preliminary research, that a high average glucose level and an older age would result in more strokes. Here we looked at the Medium Tree model which has a 93.1% accuracy according to MATLAB. However, when we look at the corresponding Confusion Matrix we see 36 false negatives and 19 false positives, with only 3 true positives. The Confusion Matrix shown below, once again shows that the model did not very accurately predict the number of strokes within the patient population based on average glucose level and age.


Classification Learner - Scatter Plot
Predictions: model 1.2
avg_glucose_level
age
Plot
Data
Model predictions
Correct
Incorrect
Predictors
X: age
Y: avg_glucose_level
Classes
Show
Order
0
1
How to investigate features
Data set: strokedata_prepped Observations: 4254 Size: 1 MB Predictors: 6 Response: stroke Response Classes: 2
Validation: Holdout Validation with 15% held out


Classification Learner - Confusion Matrix
Model 1.2
593
8
36
1
True Class
Predicted Class
Plot
Number of observations
True Positive Rates (TPR)
False Negative Rates (FNR)
Positive Predictive Values (PPV)
False Discovery Rates (FDR)
What is the confusion matrix?
Data set: strokedata_prepped Observations: 4254 Size: 1 MB Predictors: 6 Response: stroke Response Classes: 2
Validation: Holdout Validation with 15% held out

Finally we looked at the Coarse Tree model for predicting stroke based on average glucose level and age. The accuracy came back for the Coarse Tree model at 94.2% which is a very good accuracy measure. However, on the graph we do not see any red marks. Perhaps this is because the red data points are behind the blue data points.



Using the “All Quick-To-Train” method, three decision tree models were produced for the trained data. When we plotted the different risk factors of having a stroke (age, medical risk (smoking, heart disease, hypertension), average glucose level) in the classification learner to try to predict if someone has a stroke or not, MATLAB returned the decision tree models with an accuracy above 90% for all 3. This would indicate that our risk factors are indeed good predictors if someone is likely to have a stroke. However, when we looked at the Confidence Matrix produced by these different pairings, the number of false positives was high compared to the number of true positives. The number of false negatives compared to true negatives was a better ratio. This would indicate that the model does have trouble accurately predicting the likelihood of someone having a stroke based on risk factors such as their average glucose level, age, medical risk (including smoking history, heart disease, hypertension). In conclusion, according to MATLAB our model could be used with over 90% accuracy to predict if someone is likely to have a stroke. It is important though to keep in mind that false positives and false negatives still occur. However, if people are at high risk for a stroke, i.e. have any combination of the following risk factors: hypertension, history of smoking or heart disease, high average glucose level, high BMI, it would be beneficial for their health to try to change their lifestyle in someway to reduce the risk of stroke.