Capstone 2: Stroke Prediction Analysis - Project Report

Can we predict strokes in patients based on their physical and personal attributes? If so, what attributes are going to be most important in aiding in this prediction? Accurately predicting events such as strokes, could be an indispensable tool for doctors and people at risk. According to the World Health Organization (WHO), strokes are the second leading cause of death globally, responsible for approximately 11% of total deaths. Machine learning and AI can be controversial topics in many instances today; however, it is hard to dispute the value of a tool that can help predict strokes before they happen. In this data analysis, we predict strokes in patients using a variety of classification algorithms, identify the most important patient attributes, and highlight the difficulty of successfully utilizing a healthcare prediction model like ours.

The data we are using is publicly available and can be found on Kaggle under *Stroke Prediction Dataset*. The dataset is a collection of both categorical and numerical features that describe an individual patient. The categorical features include gender, marital status, work type, residence type, hypertension, heart disease, smoking status, and stroke. The other features are age, average glucose level, and BMI. The data includes 5110 observations, where only 249 patients, or under 5%, experienced a stroke. Therefore, there is a class imbalance present and we will need to take this into account to attempt to effectively model the data.

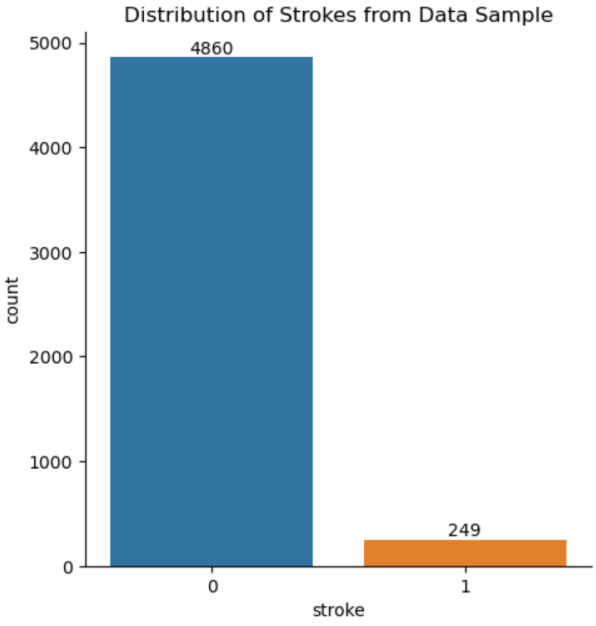

Fig 1: Distribution of stroke and non-stroke classes in the data

Before we could model, we needed to perform a few data-cleaning exercises. First of all, we removed the single “other” observation from the gender column because there was only one. Then, we replaced all missing values from the BMI column with the column’s median. We chose the median here because the BMI is close to normally distributed. Then, we encoded the data using a combination of one-hot encoding for our binary features and dummy encoding for the

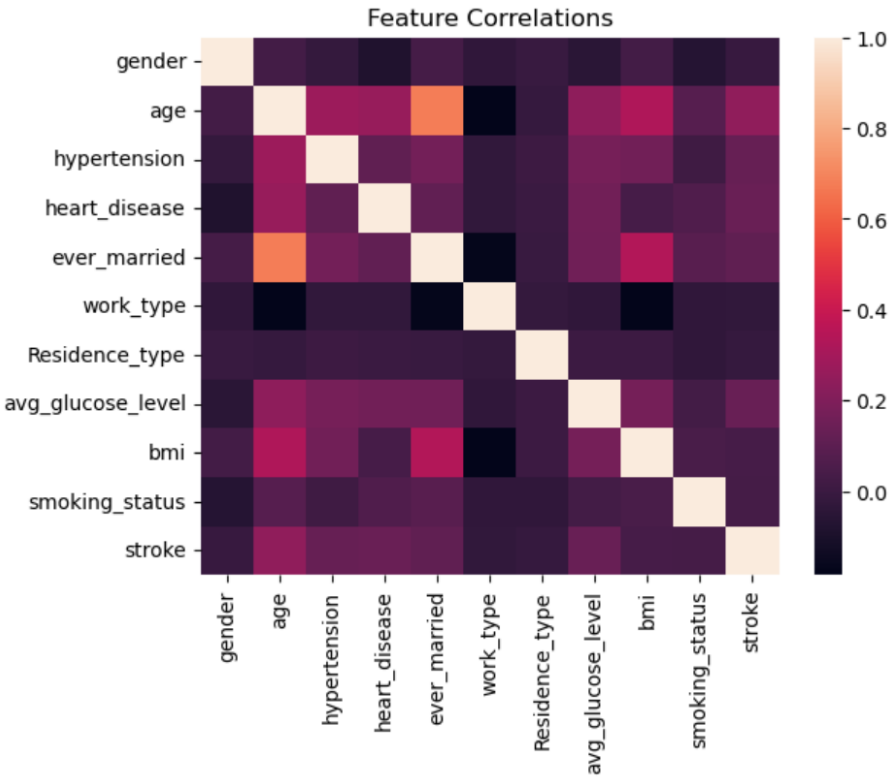

Fig 2: Correlation of features in data

remaining categorical features. Lastly, before modeling, we split the data into training and testing sets using an 80:20 split. This split left our data with approximately 4000 training observations and 1000 testing observations. Here, the X data contains 17 features and the y data is a single array.

In our exploratory data analysis, we achieved several tasks including highlighting the class imbalance, identifying correlating features, and digging into the distribution of the response variable (stroke) within sub-features. First, as mentioned earlier, there is a class imbalance present in our response variable of approximately 19:1. A visual of this imbalance can be found in Fig 1. Also, we looked at the correlations between all of the features and plotted them in a heatmap. This heatmap can be found in Fig 2. It can be seen here that none of the features are strongly correlated with stroke. Furthermore, age appears to be the most correlated but it only has a value of approximately 0.25. This picture does not tell the whole story about the relationship between stroke status and the other features; however, it is a hint that this problem isn't going to be a straightforward modeling process.

In the modeling stage, we trained and tested several different algorithms to assess the performance for the specific problem. The algorithms selected for this project are random forest classifier, support vector machine, logistic regressor, XGBoost classifier, and CatBoost classifier.

To assess the performance of the models, we recorded the area under the curve (AUC), the

	Model Name	AUC	F1 Score	F1.5 Score
0	RandomForest	0.823824	0.000000	0.000000
1	SVM	0.583401	0.000000	0.000000
2	LogisticRegression	0.864012	0.000000	0.000000
3	XGBoost	0.667490	0.098765	0.082019
4	CatBoost	0.829965	0.000000	0.000000
5	RandomForest_scaled	0.818893	0.120482	0.101246
6	SVM_scaled	0.661744	0.164948	0.148997
7	LogisticRegression_scaled	0.810333	0.201439	0.210162
8	XGBoost_scaled	0.732745	0.025316	0.020767
9	CatBoost_scaled	0.831048	0.141414	0.128895

Table 1: Performance results of models for both original data and upsampled data

F1-score, and the F1.5-score. The F1.5-score was recorded because it places a higher emphasis on recall rather than precision. The recall is more important in data like this because it is important that we minimize the false-negative predictions. Furthermore, false-negative predictions in this scenario are patients who have a stroke status of 1 but we predict them as 0. This scenario is also referred to as a Type II error and can be detrimental in health-related instances.

We modeled and trained baseline models for each algorithm using a grid search cross-validation method for parameter tuning. However, the classification results were very poor and all of the models except the XGBoost classifier predicted all observations as non-stroke. In a class-imbalanced environment, this classification may achieve high accuracy; however, it has a recall of zero and is useless to the project. The XGBoost algorithm predicted only 1 stroke correctly, which is only a marginal improvement.

Given the poor performance and high-class imbalance of the baseline models, we upsampled the training data using a synthetic minority oversampling technique (SMOTE) to remove the class imbalance in the data and re-trained our models. Removing the imbalance from the data improved all five models. In fact, the recall is greater than zero for all models. The full set of results can be seen in Table 1 where the “scaled” data is the oversampled data. We selected Logistic Regression, SVM, and CatBoost using the upsampled training data as our best models (in that order) for having the maximum F1-score, F1.5-score, and AUC combinations. However, the classification results are still barely better than guessing. The Logistic Regression model, the best of its class, correctly predicted 14 strokes, missed 48 strokes, and misclassified 63 non-strokes as strokes. These results tell us a few things about the data: First, is that predicting a stroke may be quite difficult; Secondly, we may be missing key features that help predict strokes in patients.

	Model	F1-Score	F1.5-Score	AUC (w/ F1 Threshold)	AUC (w/ F1.5 Threshold)	F1 Threshold	F1.5 Threshold
0	SVM	0.243137	0.303008	0.665625	0.665625	0.010	0.010
1	Logistic Regression	0.313167	0.398884	0.763693	0.763693	0.266	0.266
2	CatBoost	0.319728	0.411801	0.782678	0.800874	0.138	0.122

Table 2: Performance results of top models following classification threshold tuning

Then, we performed some additional analysis on the top models. First, we wanted to identify the most important features of each model. For all three models, the top features were very similar with only a couple exceptions. The top six that showed up the most were: never smoked, smokes, formerly smoked, unknown smoking status, age, and private work type. Furthermore, we found there were some correlations between smoking status, working in the private sector, and stroke status, respectively. Upon further analysis of smoking versus strokes, the results were as we predicted that individuals who smoke or formerly smoked, have a higher proportion of strokes. Also, individuals in the private sector have a higher chance to have a stroke than others. We performed a hypothesis test on this discovery and found the results to be statistically insignificant and were unable to reject the null hypothesis. A full set of stroke proportions can be found in Fig 3. The top features with the most strokes are not a surprise, however, it is a surprise that our models did not emphasize these features more.

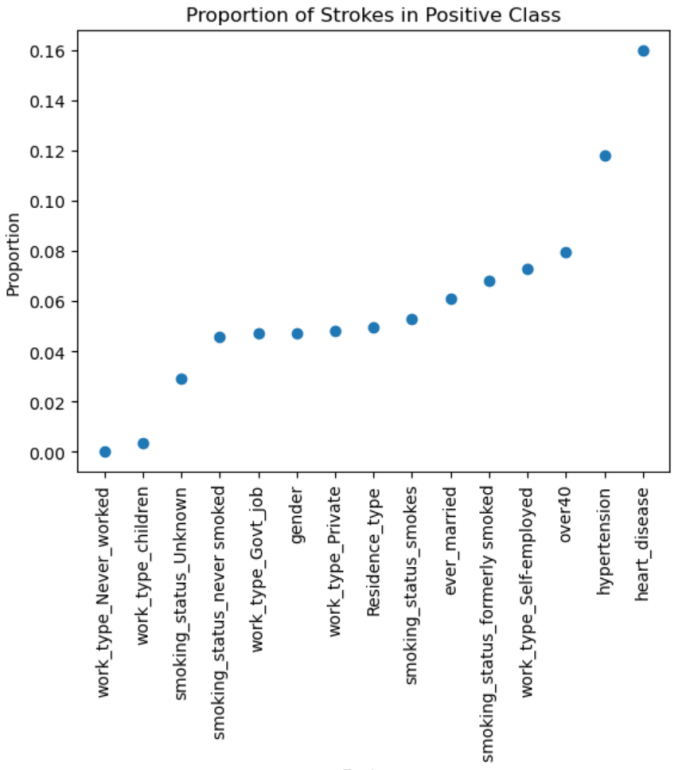

Fig 3: Proportion of strokes per feature in their respective positive classes

Lastly, we did some testing on the threshold parameter for classifying the probability predictions from the models. For all of the prior models, this parameter was set as 50% or 0.5, which is the default. But working with imbalanced data may suggest a different threshold value. Therefore, we calculated the optimal threshold for both the F1-score and F1.5-score for each model and re-calculated the metrics. A full set of results for the threshold tuning can be found in Table 2. We found that the threshold parameter is indeed important in this analysis and exposed the CatBoost model to be the best. This model correctly predicted 51 strokes, missed 11 strokes, and misclassified 212 non-strokes as strokes. The confusion matrices for the CatBoost model before and after threshold tuning can be found in Fig 4.

In future work, this model could be refined with additional observations (more patients) and additional features. Perhaps, there are other features that doctors use in practice today to determine if a patient is at risk for strokes that would also be useful for our model. Also, additional parameter tuning could be performed to our existing algorithms or perhaps additional algorithms could be tested such as neural networks.

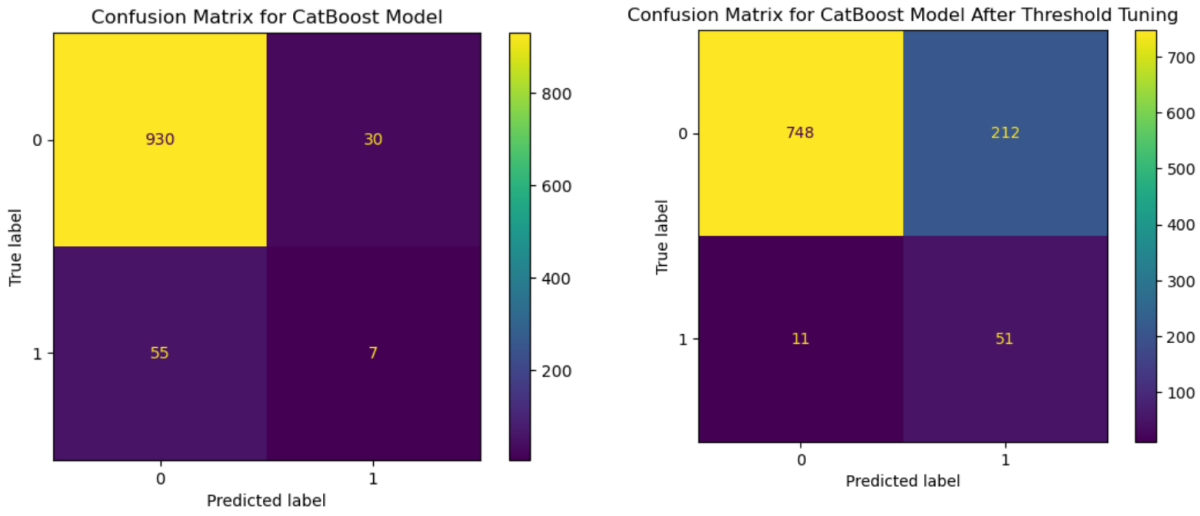

Fig 4: Confusion matrices for CatBoost model before and after threshold tuning