# Final Report

This project sought out to answer a single question: Which single clinical feature is best at predicting a stroke event? A solution was successfully discovered, and a model capable of predicting a stroke was created. Careful steps were constructed to get to this point.

The dataset used for this project was collected from Kaggle. The csv file was fairly clean, which was the reason the dataset was picked initially. There was only one column with missing values (BMI), and these values were filled with the median value of the column. From here, exploratory data analysis was conducted.

The exploratory data analysis phase of the project was where the solution was revealed. Absolute and relative frequencies were calculated for each feature against the target feature, then graphed using histograms. Age, heart disease, average glucose level, hypertension, and marriage status showed significant results. When examining correlation coefficients, the highest values corresponded to these exact features (Figure 1). Age seems to be the clinical feature best at predicting a stroke event.

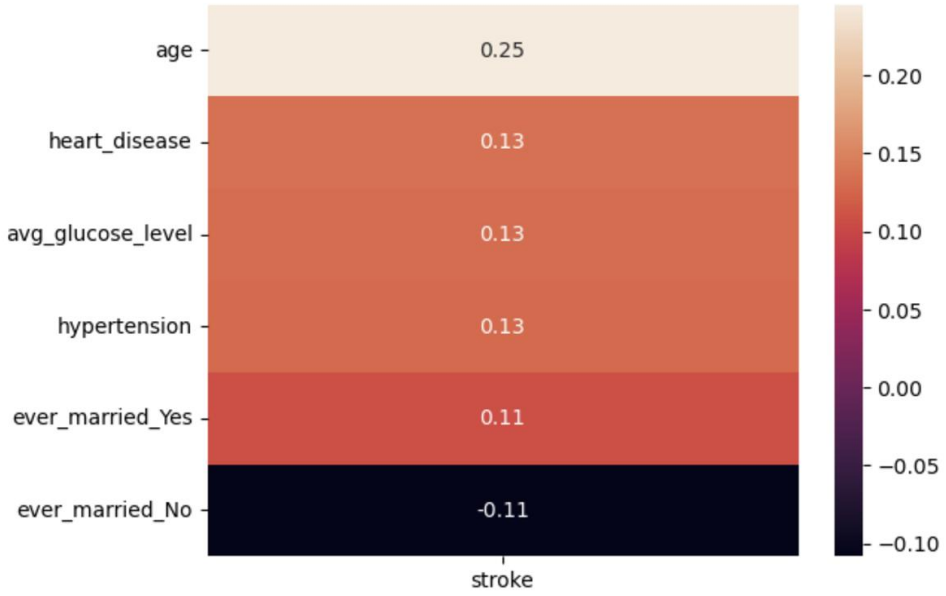


Figure 1

The preprocessing step was fairly simple. Predictive models cannot be built on categorical data, so One Hot Encoding was employed. To ensure our model better

comprehended all features (with their varying units of measurements and wide range of possible values), data was scaled using the StandardScaler provided by scikit-learn. Data was then split into train and test sets, opting for an 80/20 split, respectively. Variables were stored for ease of use in later work. All these measures ensure efficiency in the modeling phase and prevent data leakage.

When attempting to build a model, logistic regression seemed like a reasonable preliminary step, considering a binary target feature is present. The model was fit on the train data and then used to make predictions. Accuracy was extremely high, but upon further examination, the model did not perform well (Figure 2). The model simply classified all the data as belonging to class 0 (the class corresponding to no stroke event). This defeats the purpose of creating a predictive model because even a human is capable of being lazy. This information prompted two thoughts: 1. All future models will have to be assessed by a classification report and 2. We potentially have a class imbalance that needs to be addressed.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.94 | 1.00 | 0.97 | 960 |
| 1 | 0.00 | 0.00 | 0.00 | 62 |
| accuracy | | | 0.94 | 1022 |
| macro avg | 0.47 | 0.50 | 0.48 | 1022 |
| weighted avg | 0.88 | 0.94 | 0.91 | 1022 |

Figure 2

Luckily, classification reports are easy to compute, and the logistic regression model provided by scikit-learn features a class_weight argument capable of solving the class imbalance issue. Setting this class_weight argument to ‘balanced’ produced immediate improvements (Figure 3). To verify the model performed as anticipated, a resampling algorithm was imported from imblearn. imblearn offers many options for over-sampling and under-sampling algorithms. The SMOTE algorithm seemed adequate for this analysis since the minority class has a low number of instances. Over-sampling is preferred in situations such as this as opposed to under-sampling the majority class.

# Final Report

Once the SMOTE algorithm finished running, a basic logistic regression model with default parameters was fit on the resampled data and predictions were made. The resulting classification report was almost identical to the classification report for the logistic regression model with the class_weight parameter set to 'balanced'. The only difference was a slight increase in accuracy (76.125 vs 75.832). Nearly identical metrics confirm that both models performed as expected.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.98 | 0.76 | 0.85 | 960 |
| 1 | 0.17 | 0.76 | 0.28 | 62 |
| accuracy | | | 0.76 | 1022 |
| macro avg | 0.57 | 0.76 | 0.57 | 1022 |
| weighted avg | 0.93 | 0.76 | 0.82 | 1022 |

Figure 3

Other models were also tested to see if further improvements could be made. A decision tree, random forest, and k-nearest neighbors classifier were imported from scikit-learn and fit on the resampled data (not all of these models have the class_weight argument seen in the logistic regression model). The results did not best the two logistic regression models previously built. One more route was explored. PyCaret is a library that creates many machine learning models on given data with only a few lines of code. After loading in PyCaret and feeding in the original DataFrame, a list of models was presented. The models deemed superior were those with high accuracy, which is misleading given the class imbalance. Therefore, the model with the highest recall (Naive Bayes) and the model with the highest precision (AdaBoost Classifier) were examined. The results were disappointing.

To further add to the disappointment, hyperparameter tuning yielded no improvements to the two logistic regression models highlighted throughout this report. Grid Search and Randomized Grid Search techniques were utilized. Principal Component Analysis (PCA) was performed in an attempt to extract the components responsible for over 70% of the variance seen in the data. The idea behind employing

this dimensionality reduction technique is to reduce noise and build a model that generalizes better to unseen data, essentially limiting overfitting. However, the subsequent model generalized too well and did not properly represent our data, leading to poor performance metrics.

To conclude, feeding train data into the SMOTE object to perform over-sampling and then using that train data to build a logistic regression model with default parameters was the best performing model. A pipeline will be constructed in a model metrics text file with the appropriate steps for extra clarity and easy implementation. If opting for simplicity, the logistic regression model provided by scikit-learn features a class_weight argument that, when set to ‘balanced’, performs a similar process to the SMOTE object resulting in an almost identical classification report.

So, it seems that age is the single clinical feature best at predicting a stroke event, but is it really that great of a predictor? Not exactly. With a correlation coefficient of around 0.25, the relationship between age and stroke appears to be a weak one. This is not necessarily a bad thing. The medical community is aware that there is no sole indicator capable of signaling that a stroke event is imminent. Rather, there exists a group of indicators used in unison to inform patients that they are at high risk of having a stroke. Even then, this methodology is not entirely sound as seen with the limitations of the predictive model built. It is entirely possible there are better indicators out there that have not been discovered yet. In order to find this solution and build a more robust model, better data is needed.