CS M148 Project 3 Report

Han Goh

June 1, 2021

1 Executive Summary

The goal of this project was to develop a model that predicts whether a patient is likely to experience a stroke, given features including age, medical conditions, smoking status, and more. Such an endeavor is well justified as stroke accounts for 11% of total deaths, making it the second leading cause of death around the world.

As the data was already collected and given to us by the UCLA hospital, this report picks up the process at the data engineering step, and pushes to model fitting and analysis.

For data engineering, the main challenge was accounting for the natural class imbalance within the data, with the vast majority as non-stroke patients. The problems of class imbalance continued through to the model building and analysis stage, where metrics needed to be carefully chosen to accurate describe its performance. Another important step was feature selection and analyzing feature importance, where age and smoking status seemed to be most correlated in predicting stroke.

The model that performed best in terms of precision and recall was a tuned Gradient Boosting model. As will be discussed later, whether this "best" model is suitable for real-life application needs further consideration, as there are more to a model's viability than just its predictive power.

2 Background and Introduction

As mentioned earlier, stroke is one of the leading causes of death worldwide, and it can also lead to long-term disabilities otherwise. However, the CDC states that the consequences of a stroke can be diminished “when emergency treatment begins quickly”.[1]

One of the main ways to increase the chances of quicker treatment is to be able to recognize the symptoms of a stroke. However, a better safeguard will be to predict whether someone is prone to stroke in the first place, so that they (and those around them) can be better prepared in the event that it occurs. The CDC lists “high blood pressure, high cholesterol, smoking, obesity, and diabetes” as leading causes, many of which are included as features in the dataset. It is important to note that this dataset does not include information about race and ethnicity, which are also stated to highly influence the likelihood of stroke for those living in the United States.

[1] CDC: Stroke Facts and Statistics

3 Methodology

This section will detail the steps taken to engineer the data and fit an optimized model to best form predictions about strokes.

3.1 Initial Analysis

The first step for this project was to analyze the given dataset. After plotting the histograms for the numerical values, it was clear that class imbalance will be of importance (see Figure 1 below). The other histograms can be seen in the Jupyter notebook attached.

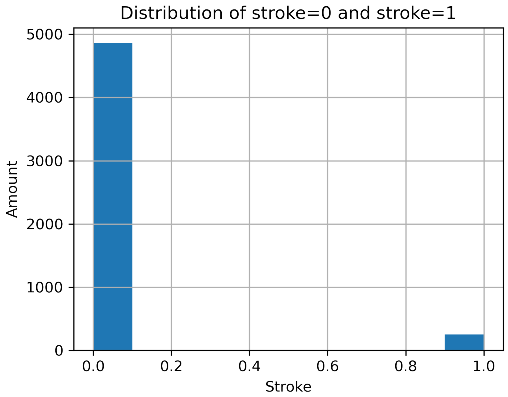

Figure 1: Class Distribution of stroke=0 and stroke=1

With such an unbalanced dataset, it is important to consider what metric will be best in determining model performance. Conventionally, the F1-score is a better indication of performance over accuracy for such datasets. But in disease prediction, the cost of false negatives should be prioritized—this leads to recall as the chosen metric[2].

Then, a correlation heatmap (see Figure 2 on the next page) was used to gauge which features may be most important in predicting stroke. Features ages and average glucose level (indicative of diabetes) are most correlated to stroke, which aligns with the predictors the CDC mentions. There is co-linearity between age and all the other numerical features, but this was overlooked as they are all deemed important predictors by the CDC.

[2] Problems arose with recall, as some models would end up predicting every point as stroke=1. More discussion on this in the model optimization section.

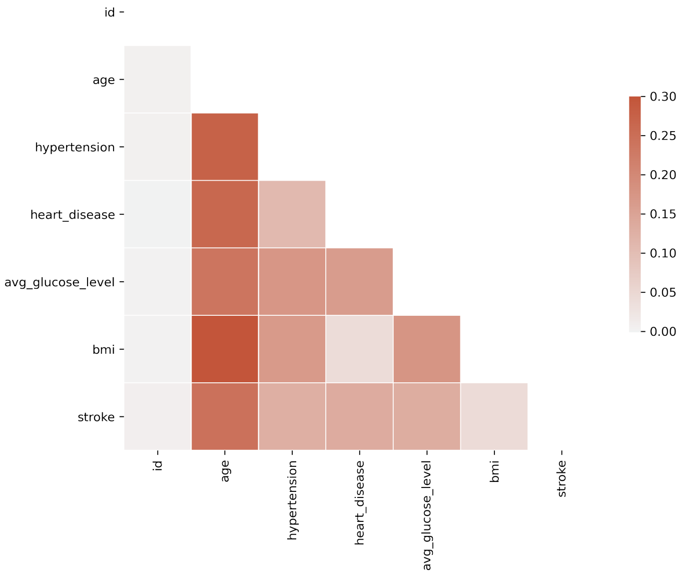

Figure 2: Correlation Heatmap

3.2 Pipeline: Scale, Imputation, Augmentation, Balance

As we can see above in Figure 2 (and with our intuition), the 'id' is not relevant to the dataset and was dropped. For feature augmentation, heart_disease was squared to increase its presence, as its correlation seemed a bit lower than the others. BMI was not squared as there were imputed values (median was chosen to keep imputation simple), so squaring it will increase the bias that was introduced into the data. Hypertension and heart_disease was crossed based on intuition and prior research on the subject. In addition, StandardScaler() was used to normalize the values as nothing special seemed to be needed, and the categorical features were one-hot encoded as they were non-ordinal and multi-categorical.

After the pipelining, the class imbalance was artificially corrected by using the SMOTEENN sampler from the imlearn library. The resulting distribution can be seen below in Table 1:

Label	Original	Resampled
0	3899	3187
1	189	3766

Table 1: Class Distributions Before and After Resampling

3.3 Logistic Regression and PCA

With this resampled data, an OLS regression was used to confirm intuitions based on research and correlation values on feature (un)importance [3]. The most notable results were that age had the largest coefficient, indicating its importance, and that features work_type_Never_worked and work_type_Private are likely unimportant, as they had p-values > 0.05.

The two features were dropped and PCA was used to reduce the dimensionality of the dataset. To confirm its validity, three Logistic Regression models were used: one trained on the original dataset, one trained on the reduced dataset with the two features dropped, and the last on the dataset formed by PCA on the reduced data. As seen in Table 2 and Figure 3 below, they performed comparably, supporting the feature drop. This reduced the number of features from 22 to 10.

Metric	Original Dataset	Reduced Dataset	PCA Dataset
Accuracy	0.702	0.704	0.666
Precision	0.155	0.156	0.108
Recall	0.917	0.917	0.800
F1 score	0.265	0.266	0.190

Table 2: Metrics of Logistic Regression on Original, Reduced, and PCA Dataset

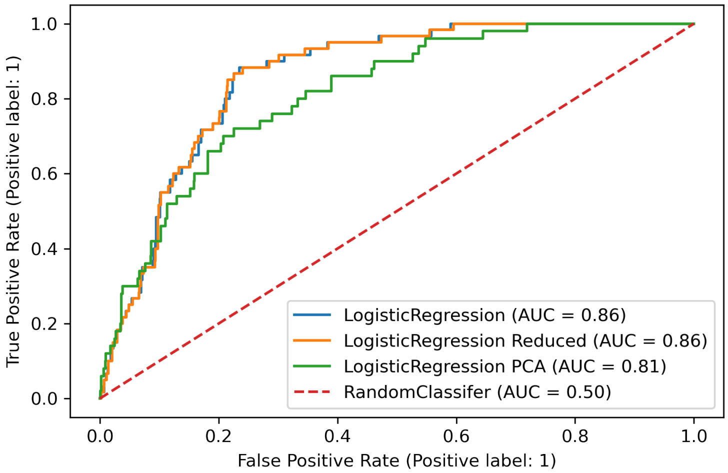

Figure 3: ROC curves of Logistic Regression on Original, Reduced, and PCA Dataset

[3] The full result can be seen in the notebook

3.4 Model Fitting and Evaluation

Leveraging OLS regression and PCA greatly reduced the dimensionality while keeping most of its variance. Models were then trained on this PCA dataset, with the intention to maximize its recall and evaluated via cross validation.

As the data has been artificially balanced, a choice must be made: test on the balanced data, or test on the original distribution. For this project, all models were tested on the original distribution as:

- Testing on the original data best mimics the real-world distribution of these classes
- Testing on the balanced data gives overly optimistic metrics, almost all > 0.95. This indicated that balanced data will not give accurate measures of performance.

For this reason, parameter tuning was not done with the popular GridSearchCV, as it does support using two different distributions for training and testing [4]. Rather (also due to computer power limitations) just one train and test set were used to tune model parameters, and they were eventually checked with all other models in the cross validation step.

However, purely looking at recall caused some problems in model tuning. For example, consider the following:

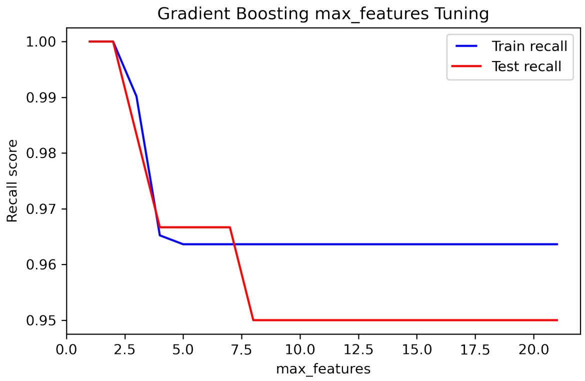

Figure 4: Recall of GB Classifier based on max_features

A max_feature value of 1 or 2 will provide a recall of 1.0, but this will make the classifier predict everything as 1. This is not a valid measure of performance, then as precision will plummet. Instead, a middle ground of max_features=5 was chosen. All model parameters were tuned with this in mind.

[4] Or I just don't know how to...

It should also be noted that sklearn's *cross_val_score* was not used, with the same reason as not using GridSearchCV. Instead, a StratifiedShuffleSplit was used on the a training with PCA reduction, but without resampling. Then, each training fold was resampled after, such that the training sets will be resampled while the test sets follow the original distribution.

In total, five models were trained: the aforementioned Logistic Regression, three ensemble methods (AdaBoost, Random Forest, Gradient Boosting), and a Neural Network. These models will be further discussed in the Results section.

4 Results

Using a 10-fold cross validation (customized, as noted before), below are the performances of the five models:

Metric (averaged)	LogReg	AdaBoost	RF	GB	NN
Accuracy	0.684	0.750	0.829	0.648	0.709
Precision	0.119	0.131	0.136	0.111	0.115
Recall	0.850	0.730	0.470	0.880	0.740
F1 score	0.209	0.222	0.211	0.197	0.198

Table 3: Averaged Metrics over 10-fold CV of Five Different Models

The Gradient Boosting classifier performed the best in terms of recall. Gradient Boosting was tuned, as its base model performed the best out of the three ensemble methods. Using the aforementioned tuning methods, the parameters of the Gradient Boosting Classifier were as follows:

- learning_rate = 0.01
- max_features = 5
- min_samples_leaf = 0.01
- min_samples_split=0.1
- n_estimators = 40
- subsample = 0.9

The Neural Network was also tuned, with its parameters as:

- activation = 'tanh'
- alpha = 0.0005
- hidden_layer_sizes = (200,200,200)
- learning_rate = 'adaptive'
- learning_rate_init = 0.1

5 Discussion

As mentioned before, recall was the main metric of focus in model evaluation and optimization. This metric was chosen because the cost of a false negative is high in stroke prediction, which possibles consequences of long-term disability or even death.

On that note, as seen in Table 3, the tuned Gradient Boosting Classifier performed best in terms of recall, with an average recall of 0.880. However, it is important to note that by maximizing recall, this classifier also has the lowest precision out of the five models. This means that, on average:

- The classifier will correctly predict 88% of all patients that experience a stroke. In other words, it will correctly identify 88% of those that get a stroke.
- The positive predictions of the classifier have a 11% chance of being correct. That is, a 11% chance that the patient will actually get a stroke.

With limited domain expertise, it is difficult to ascertain whether this recall-precision trade is correct. A recommendation would be to use this model for its high recall. Even with low precision, informing the patient about identifying signs of stroke and ways of preventing stroke are non-expensive ways to reduce the reach of stroke-related deaths.

As for which features are most likely to be the best predictors, the analysis in this project lines up with those outlined by the CDC, including but not limited to age, hypertension, smoking status, glucose level (diabetes). Keeping these in mind while diagnosing patients is likely to not cause any negative consequences.

6 Conclusion

Given a dataset, the goal of this project was to predict whether or not a patient experiences a stroke. There were several challenges, which was mainly focused around the class imbalance. Recall was chosen as the main metric (over accuracy) due to this imbalance and the fact that false negatives must be reduced in disease detection. This class imbalance was remedied by using resampling techniques, while models were still trained on the original distribution to best mimic the real-world situations. Models were tuned and evaluated in a similar manner, with the best model resulting in a recall score of 0.88.