Capstone Proposal

Domain Background

Stroke is an ailment affecting the normal blood supply to the brain. According to the World Health Organisation stroke is the second leading cause of death globally, accounting for 10.2% of deaths in 2016.

Based on numbers of deaths alone, incidence appears to increase dramatically with age. A recent publication of the International Journal of Scientific & Engineering Research outlines some existing models for cardiovascular risk assessment, which are used to predict strokes. There may be scope to improve on these using Machine Learning techniques.

I was personally drawn to this problem by a hunger to understand the affliction more deeply. Several people close to me have either passed away from a stroke, or are now living with a disability brought on as a result.

Problem Statement

Train & deploy a Machine Learning model that can effectively & efficiently predict stroke incidence. Following this, determine what feature(s) may be causally related to strokes.

Datasets and Inputs

My Capstone Proposal is driven by the Stroke Prediction Dataset available on Kaggle.

The dataset is available as a single CSV file. This dataset was chosen as it contains features related to both a person's lifestyle (e.g. `ever_married`, `smoking_status`, `Residence_type`) and medical condition (e.g. `hypertension`, `heart_disease`, `avg_glucose_level`), along with the all-important `stroke` label.

Disregarding `id` and the label, we are left with ten features that should provide us with concrete insights in to what causes the ailment.

It is worth noting that approximately 95% of people in the dataset did not have a stroke. It will therefore be critcal that we either rebalance our dataset, or choose a scoring algorithm such as F1 score that punishes both false negatives and false positives.

Solution Statement

Split the dataset into a training & testing set, and train an ML model to predict the likelihood of a patient having a stroke, based on features given in the dataset. Deploy and test the model against the test dataset, and further tune the model if necessary.

Next, compare the effectiveness of our new model against the specified benchmark (outlined below). Follow this with a discussion on the merits of different approaches.

Benchmark Model

See data_exploration.ipynb for a benchmark model. Logistic Regression was used, with the F1 score (more info below) algorithm testing its' effectiveness.

With the help of some Grid Search, the most effective model obtained had an F1 score of `0.2513`. This is quite a low score, that we should be able to beat with some further engineering.

Note that the dataset was not resampled for this benchmark; the only significant alteration of the underlying values was to replace null `bmi` values with the median.

Logistic Regression was chosen as a benchmark as it is regularly used for binary classification, and was very straightforward & simple to train and test.

Evaluation Metrics

A crucial aspect of the final model's performance is its' ability to correctly predict instances of stroke (`stroke` = 1). Due to the imbalanced nature of the label values, accuracy will not be a good measure. Both precision and recall are useful when attempting to quantify the ability of a model to predict a single class. They are handily combined in to a single metric called F1 Score. See this link for more information on the algorithm in `scikit-learn`.

Project Design

Next steps:

- Explore other models and compare their effectiveness' against benchmark (initial research suggests RandomForest and CatBoost).
- If significant improvements on benchmark are not seen, consider balancing dataset.
- Deploy model in Amazon Sagemaker, and test with Lambda.
- What feature(s) determine stroke incidence? Opportunity for visualisations i.e. Correlation Matrix
- Discuss possible enhancements to model, involving a complementary dataset if possible.

As the dataset is relatively small (~ 50K data points), I do not see value in testing a computationally-expensive neural network. One may be tested for sake of completeness.

Note that other solutions online binned the continuous features of the given dataset, for example a bmi of between 0 and 19 is replaced with 'Underweight'. To me this is just throwing away information, so unless I see a good reason I will not do same.