Stroke Prediction Classification Model

For my capstone, I am going to create a classification model that looks at 10 attributes of an individual, and predicts whether they have or have not had a stroke. If I can determine features that correlate with stroke risk, I can share my findings with the medical community, in the hopes that they can use this information to preempt future potential strokes. Doctors can inform their patients whether they are at high risk for a stroke.These findings can also be used to make mailers for health care providers to send out, or to make posters for hospitals to hang on their walls.

I am going to use a dataset that I found on Kaggle. The dataset contains 5110 observations with 12 attributes. The first attribute is an id variable, and the last attribute is a binary variable that is equal to 1 if the patient had a stroke and is equal to 0 if the patient did not have a stroke.

To create the classification model, I'm going to start by cleaning the data. If there is no stroke event attribute for an observation, it will be removed. If other attributes are missing for an observation, I'll consider imputing the median or the mean. Otherwise, I'll remove the observation. I will also check for outliers. After the data is cleaned, I'll do some exploratory data analysis, which will involve looking at correlations between features and creating various visualizations, including scatter plots and line graphs. Once the EDA is done, I'll build a pipeline. In the pipeline, I will scale the data, perform cross validation, and identify the best hyperparameters and features to use in the final model. I'll split the data into training data and test data. The model will be formed on the training data and tested on the test data.

The biggest challenge I can foresee is that roughly 95% of the patients we've observed did not have strokes. This means that the model may end up being overly cautious. Therefore, we should care more about the model's precision and recall scores as opposed to its accuracy score.