**Final Project**

This project seeks to create a model using the dataset (‘healthcare-dataset-stroke-data’) so as to predict if a person is more likely to have a stroke based on features in the dataset such as gender, age, blood pressure, blood sugar, and marriage status, etc. I plan to use multiple machine learning models I have learned from this course as well as two other models I found online. I am aiming to find the best model to perform the classifications of the likely to have a stroke (stroke=1) and likely not to have a stroke (stroke =0).

This project has three sections. Section 1 focuses on cleaning and preparing the data (dropping null values, converting categorical values to numeric values) and analyzing the correlations between the features and the lable class. Section 2 focuses on using KNN, Logistic, and Random forest models to classify the stroke lable, and compute the associated performance matrices. Section 3 focuses on oversampling the data first and then applying different machine learning models so as to compare the results.

**Section 1:**

First I checked the dataset’s basic info and its null values. I dropped the ‘id’ column since it is not a valid feature to perform classification. Then, I filled the null values in the ‘bmi’ column using the mean value of ‘bmi’. Then I converted all categorical values to numeric values such as Female=1, Male=0, Yes=1, No=0 in 'ever\_married' column, etc. Finally, I checked the correlations between all the features and the ‘stroke’ lable.

Chart, bar chart

Description automatically generated

Based on the above graph, ‘age’ has the highest correlation with ‘stroke’ compared to other features. This means that older people have higher chance of having a stroke. ‘hypertension’,’avg\_glucose\_level’ and ‘heart\_disease’ also have relatively high correlations with ‘stroke’.

Chart, scatter chart

Description automatically generated

I also checked the relations between avg\_glucose\_level and age based on the ‘stroke’ lable. Based on the above graph, we can see that older people are more likely to have higher avg\_glucose\_level (above 100), and are more likely to have a stroke.

**Section 2:**

The datapoints for the ‘stroke’ column is very imbalanced (see the graph below). For this section, I wanted to try KNN, Logistic and Random Forest models without oversampling the data.

Chart, bar chart

Description automatically generated

**KNN classification:**

I tried k\_values=1-10. Based on the accuracy rate, k=10 gives the best accuracy rate.

Chart, line chart

Description automatically generated

When K=10, compute the MAE, RMSE, confusion matrix and TP,FP,TN,FN,accuracy, TPR and TNR values.

MAE: 0.050489236790606656

RMSE: 0.2246981014397021

Chart

Description automatically generated

Graphical user interface, application

Description automatically generated

**Logistic model:**

MAE: 0.050489236790606656

RMSE: 0.2246981014397021

Chart, treemap chart

Description automatically generated

Graphical user interface, application, website

Description automatically generated

**Random Forest:**

MAE: 0.05244618395303327

RMSE: 0.22901131839503758

Chart, treemap chart

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

From the above figures, we can see that the performance matrices such as accuracy rates, MAE and RMSE for each model are all fairly high. However, from the confusion matrix graphs and the TN /TNR table, we can tell the TNR, which represents the rate of successfully identifying stroke=1, is extremely low (basically 0). Therefore, none of the models performed well in terms of predicting the likelihood of having a stroke. This made me realize that the extremely imbalanced class distribution causes some performance matrices such as accuracy rate to be poor measures for evaluating the classification models.

**Section 3:**

In order to solve the above issue, I used SMOTE Class to perform over-sampling. The values in the ‘stroke’ column after oversampling are plotted as follows:

Chart, bar chart

Description automatically generated

I created a function run\_model, and re-ran KNN(K=10), Logistic, and Random Forest models. The results are:

**KNN(K=10):**

**Chart, treemap chart

Description automatically generated**

**Graphical user interface, text, application

Description automatically generated**

**Logistic regression:**

Chart, treemap chart

Description automatically generated

Text

Description automatically generated

**Random Forest Tree:**

Chart, treemap chart

Description automatically generated

Graphical user interface, text

Description automatically generated

I also tried to use StratifiedKfold to make 10 folds and perform SMOTE and Random Forest with each fold. The best TNR I got is below:

Text

Description automatically generated

The best TNR is 0.36 which is still not optimal.

**I did another attempt with SVC model to see if I can get a better result with TNR:**

SVC model with class\_weight='balanced':

Chart, treemap chart

Description automatically generated

Graphical user interface

Description automatically generated

In conclusion, the main purpose of this project was to predict the likelihood of a person having a stroke. Considering all the machine learning models I tried, I think the logistic model provided a relatively better result than the other models, because the TNR rate for the logistic model was 79%, which was much higher than the other models. I also think that perhaps this dataset did not include the most appropriate features for the ‘stroke’ classification. I am looking forward to doing more research on this and to finding out if other features (such as alcohol consumption, family history, etc.) would contribute to a better classification result.