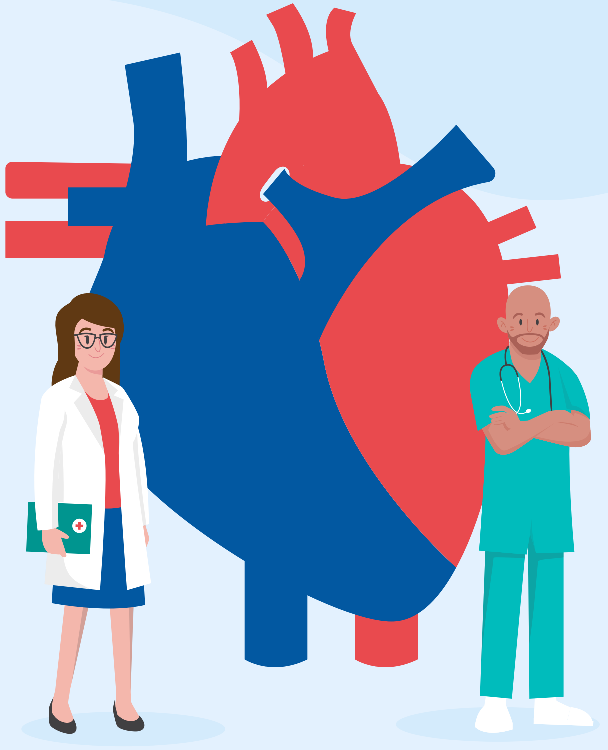
SC1015 Mini-Project

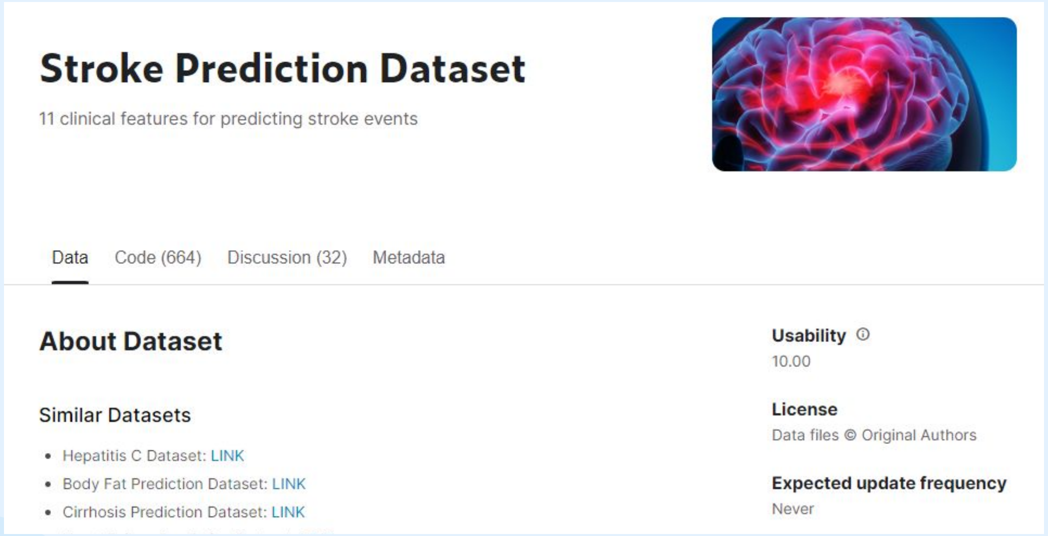
Jaslyn Tan (U2121916G)
Sim Ian Leng (U2122663C)
Gerald Ong (U2122864K)



About the Dataset

Dataset: Stroke Prediction Dataset

Link: https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset



Stroke Prediction Dataset

11 clinical features for predicting stroke events

About Dataset

Similar Datasets

- Hepatitis C Dataset: LINK
- Body Fat Prediction Dataset: LINK
- Cirrhosis Prediction Dataset: LINK

Usability
10.00

License
Data files © Original Authors

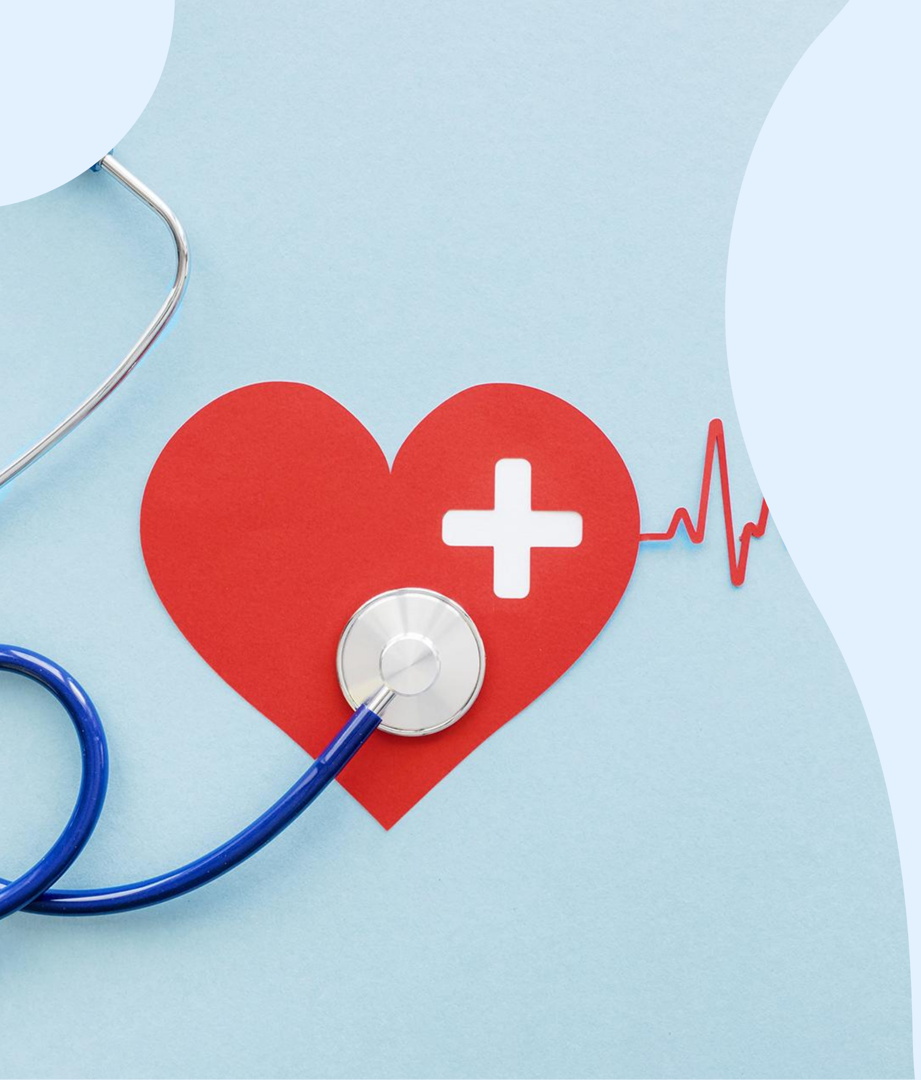
Expected update frequency
Never

Practical Motivation

Real-life Significance

Disease prediction has the potential to benefit stakeholders such as the government and insurance companies.

Furthermore, healthcare service providers can also **shift to more preventive care**, not only improving patients' quality of life but also potentially saving money in the healthcare system.



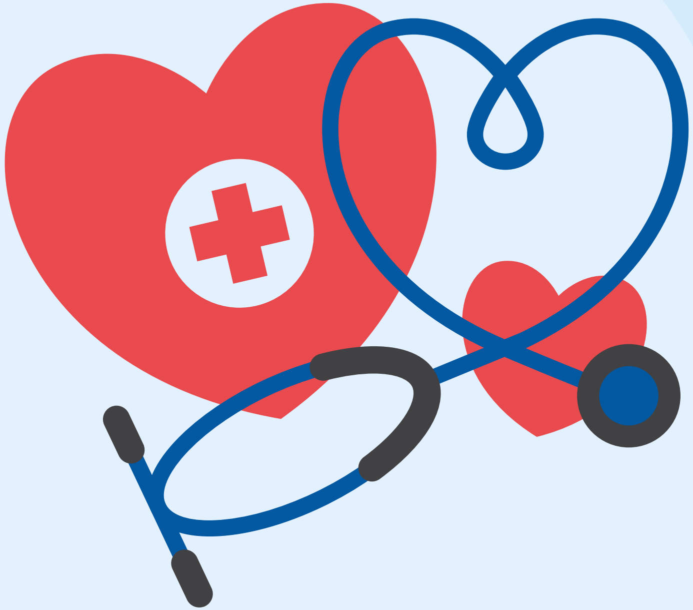
Problem Statement

Are we able to predict whether a patient is likely to get a stroke based on his/her lifestyle?

02

Data Cleaning

- Remove duplicate values
- Remove anomalies
- Drop unnecessary columns
- Remove null values
- Change object to categorical types

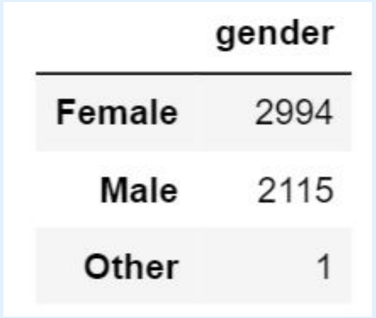


Remove duplicate values → Drop unnecessary columns → **Remove anomalies**

→ Remove null values → Convert object to categorical types

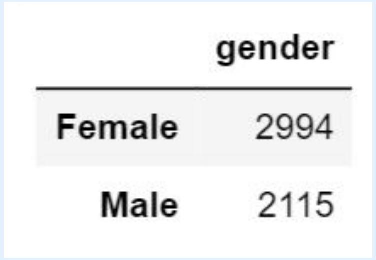
Detected anomaly: Under the gender column, there is a category "Other".

Before removing:



	gender
Female	2994
Male	2115
Other	1

After removing:



	gender
Female	2994
Male	2115

Remove duplicate values → Drop unnecessary columns → Remove anomalies →

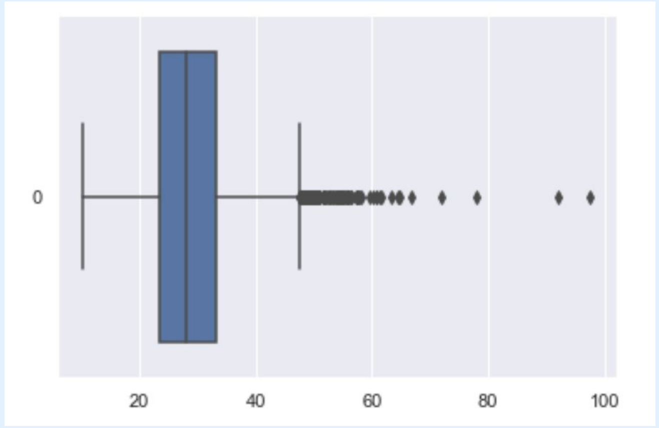
Remove null values → Convert object to categorical types

01 **Identified 201 null values under the column ‘bmi’**

From our understanding, we can either

- Remove the rows with null values
- Fill up with mean/median

In our case, we chose to fill the null values up as filling up is better than removing data from the dataset. Further research tells us that we should fill up the null values using the median since there are many outliers in the dataset.



Remove duplicate values → Drop unnecessary columns → Remove anomalies →

Remove null values → Convert object to categorical types

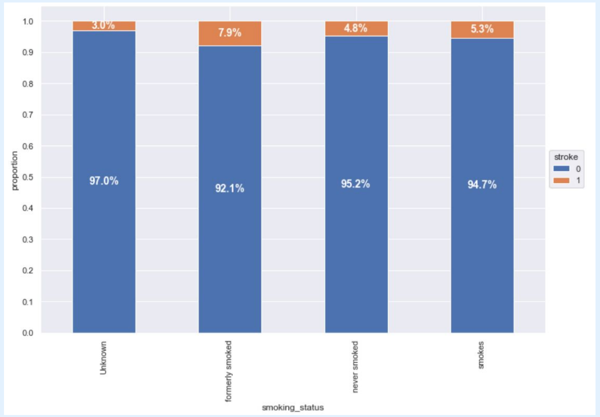
02 Identified an ‘Unknown’ category under the smoking_status column

Consequence: Result in ambiguity for the ML, where it will wrongly conclude that people with unknown smoking status are less likely to suffer from a stroke.

Action taken: Remove null values by reclassification (into other categories)

Based on the probability of getting a stroke according to smoking_status,

- Someone who has never smoked is least likely to suffer from a stroke (4.8%)
- Someone who formerly smoked is the most likely to suffer from a stroke (7.9%)



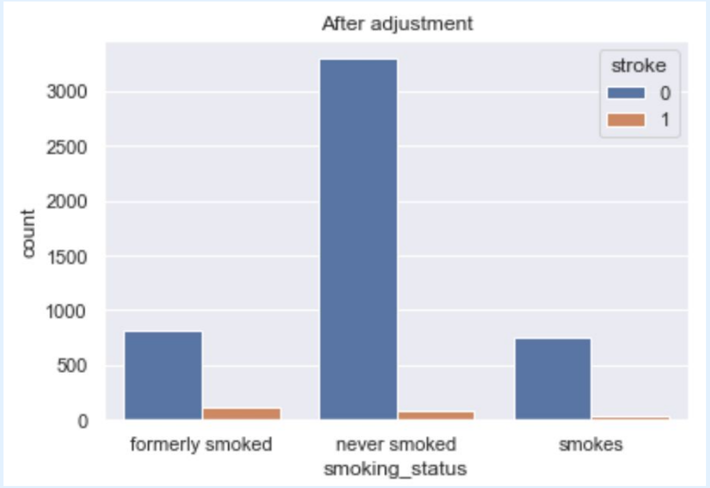
Remove duplicate values → Drop unnecessary columns → Remove anomalies →

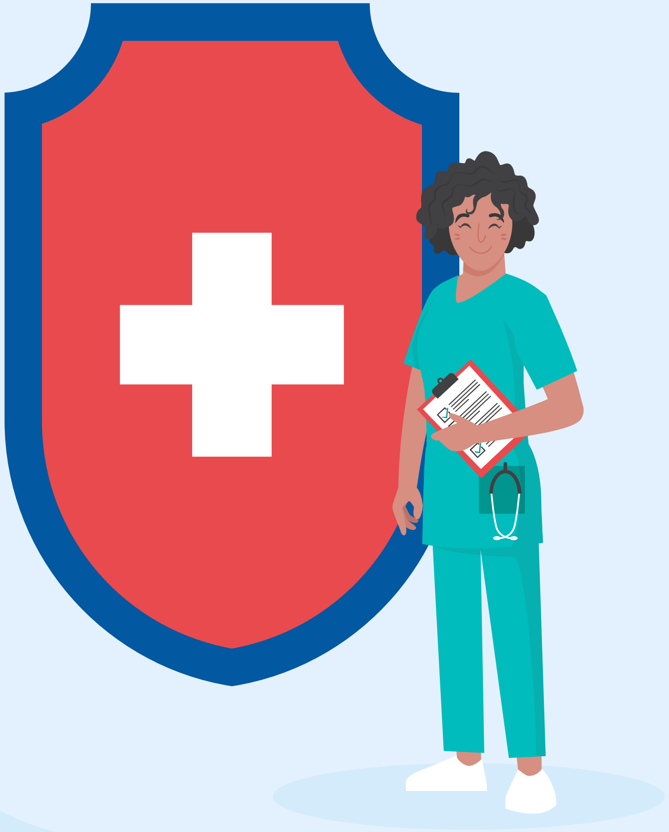
Remove null values → Convert object to categorical types

02 **Identified an 'Unknown' category under the smoking_status column**

Therefore, we will:

- For those who suffered from a stroke, set the category to 'formerly smoked'
- For those who did not suffer from a stroke, set the category to 'never smoked'



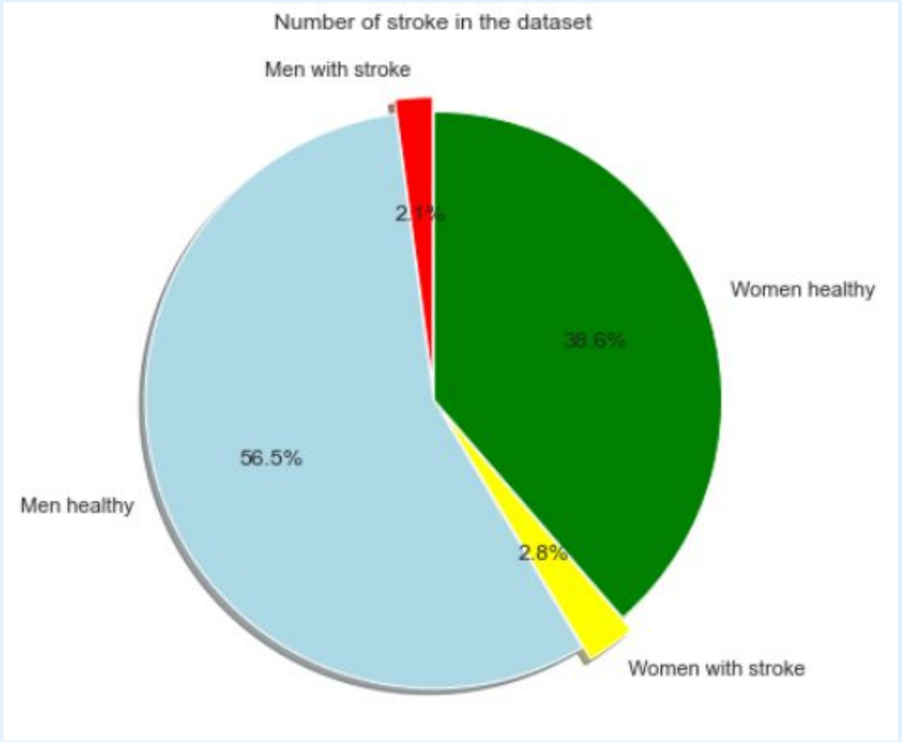


03

Exploratory Data Analysis

Data Overview

- Assessment Metric: **Stroke Rate**
- 3 **Continuous** Variables
    - Age
    - Average Glucose Levels
    - BMI
- 7 **Categorical** Variables
    - Gender
    - Hypertension
    - Heart Disease
    - Residence Type
    - Ever Married
    - Work Type
    - Smoking Status

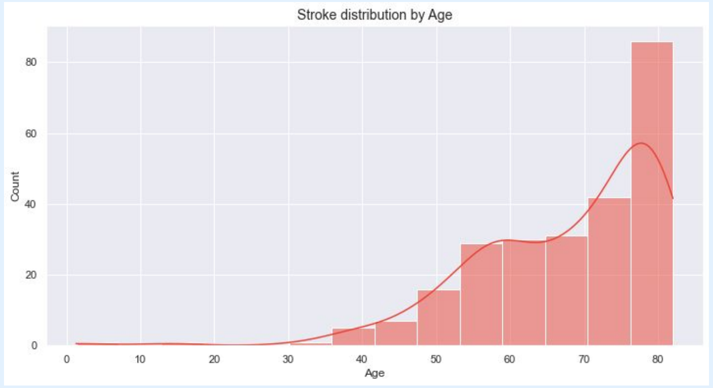


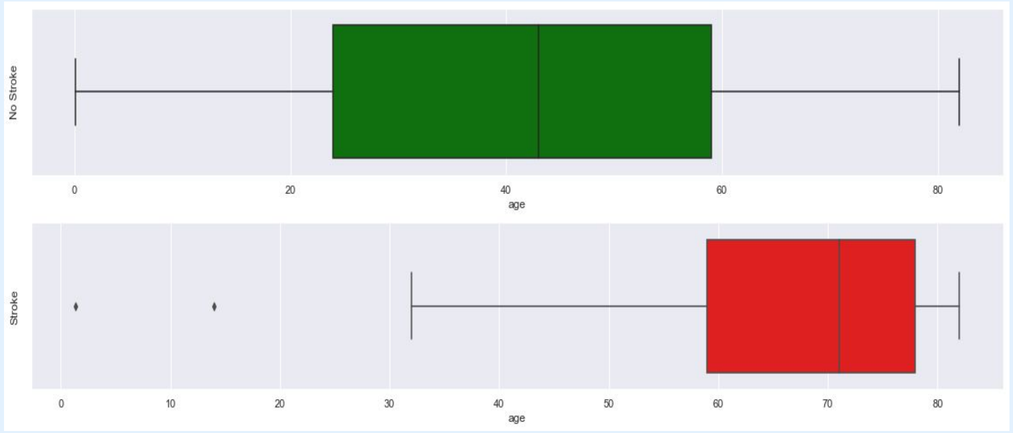
Age

Stroke count increases with age

A larger proportion of stroke patients are of a higher age.

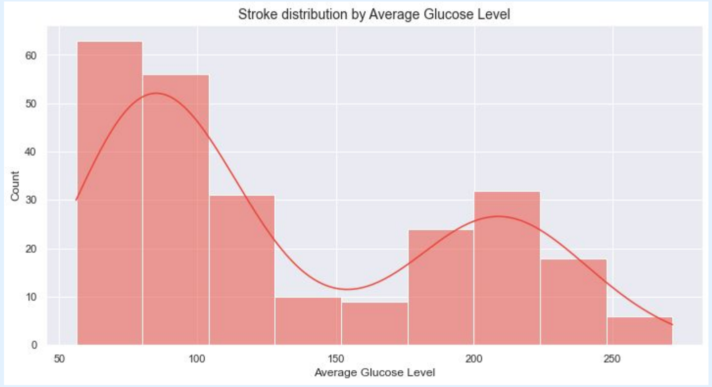
The median age for patients with stroke is significantly higher than that of patients without stroke.



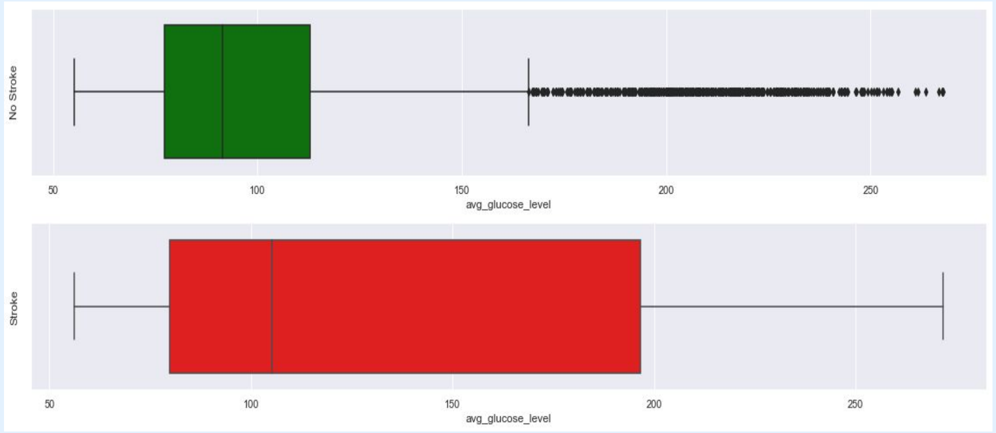


Average Glucose Level

Stroke patients have higher average glucose levels



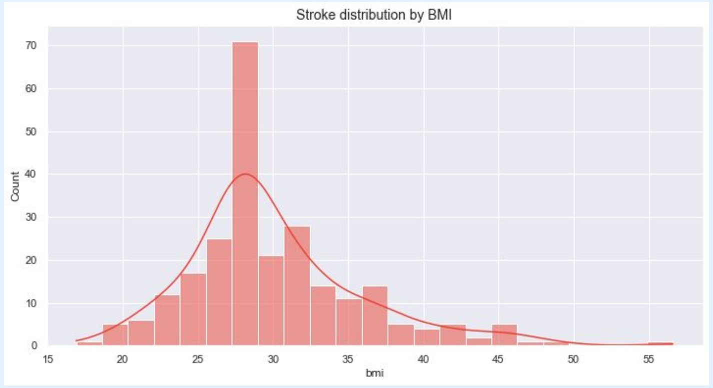
Median average glucose level is higher among strong patients than non-stroke patients.

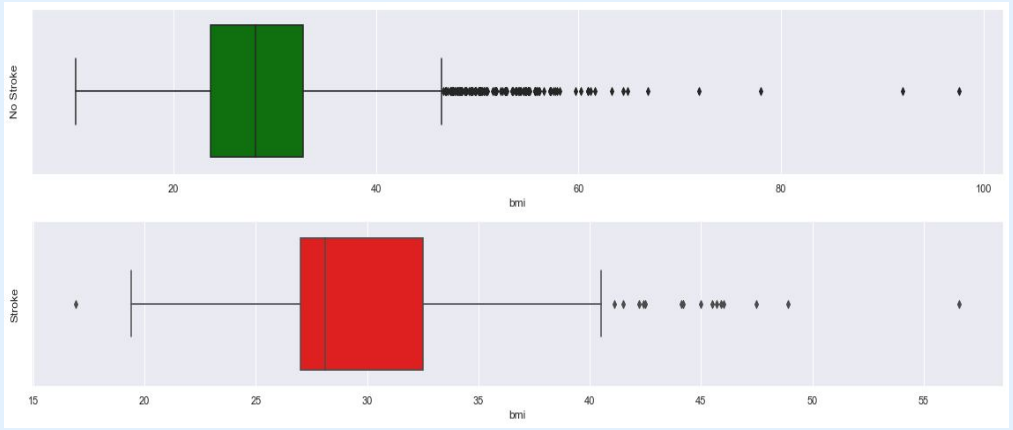


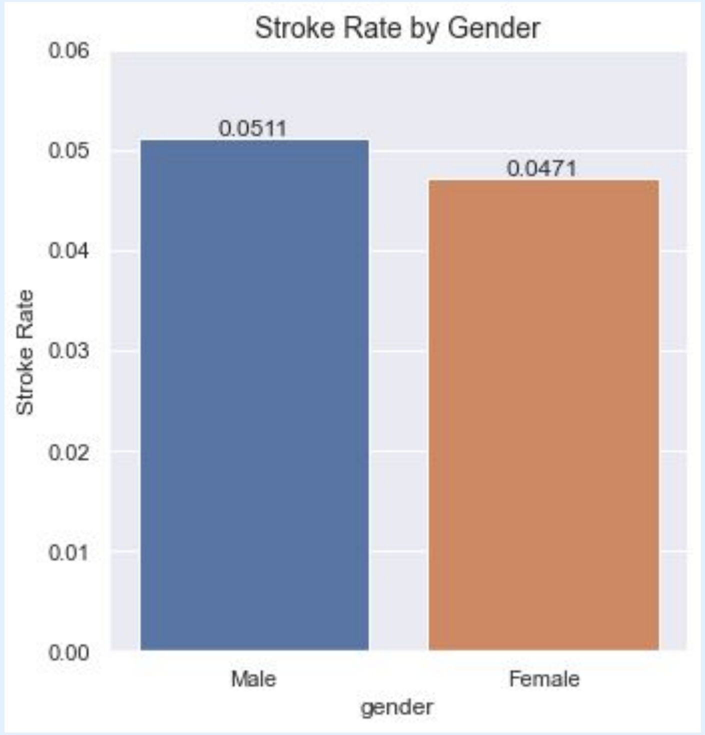
Body Mass Index

Stroke patients have a higher BMI than non-stroke patients

Patients with stroke have a higher median Body Mass Index as compared to patients who have not suffered from a stroke.



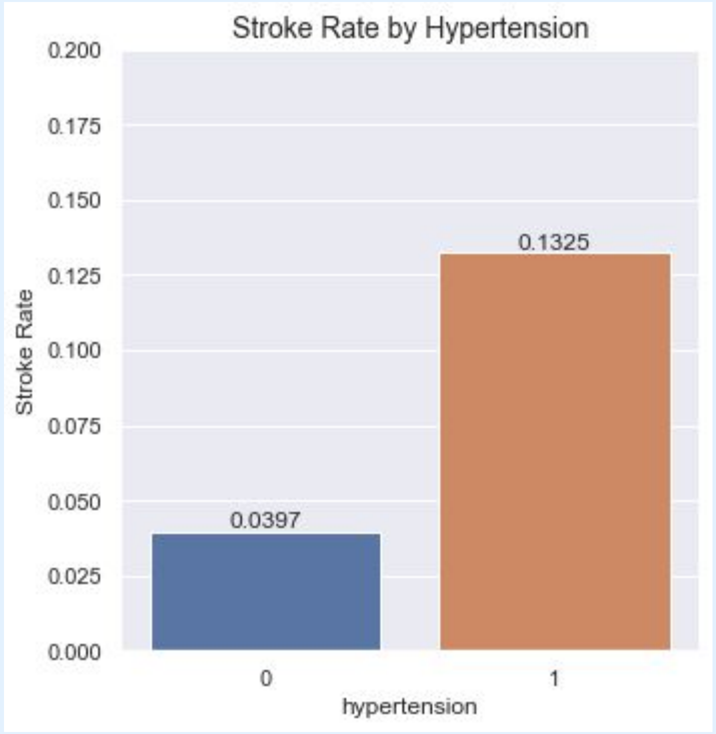




1

Gender

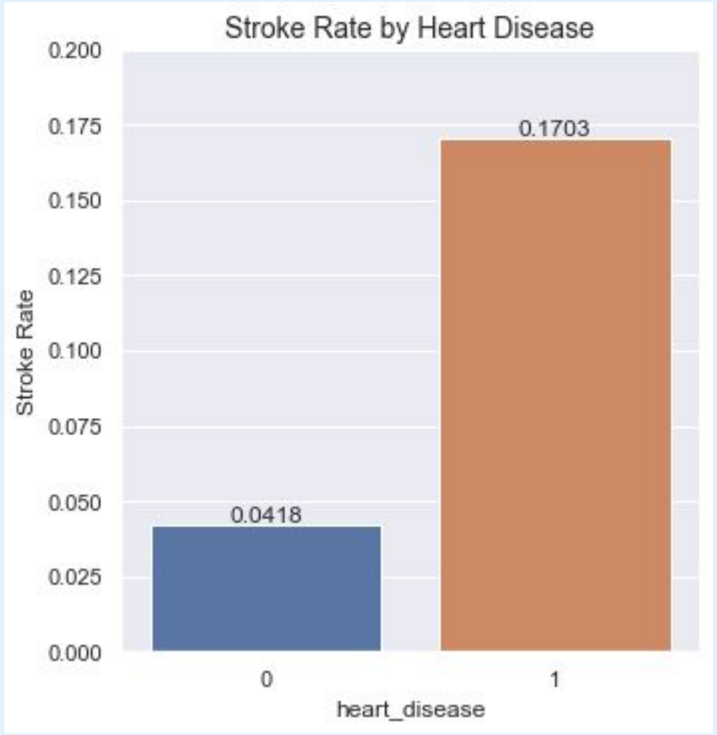
Men have a slightly higher stroke rate than women.



2

Hypertension

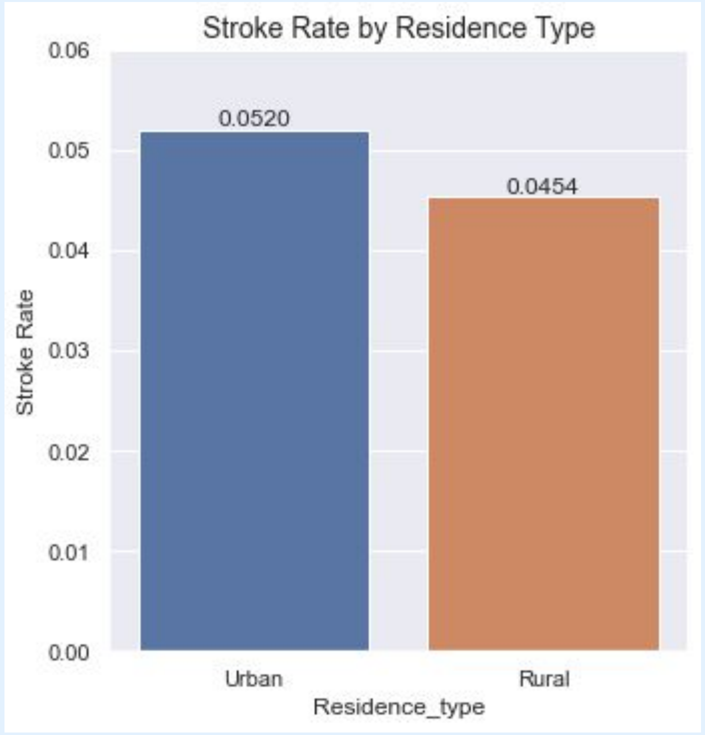
People with hypertension have a much higher stroke rate than those without.



3

Heart Disease

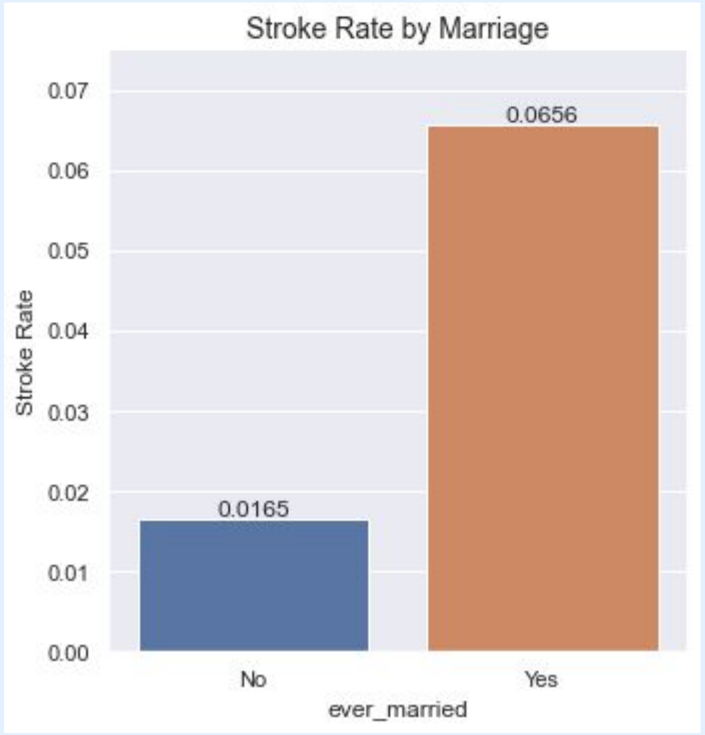
People with heart disease have a much higher stroke rate than those without.



4

Residence Type

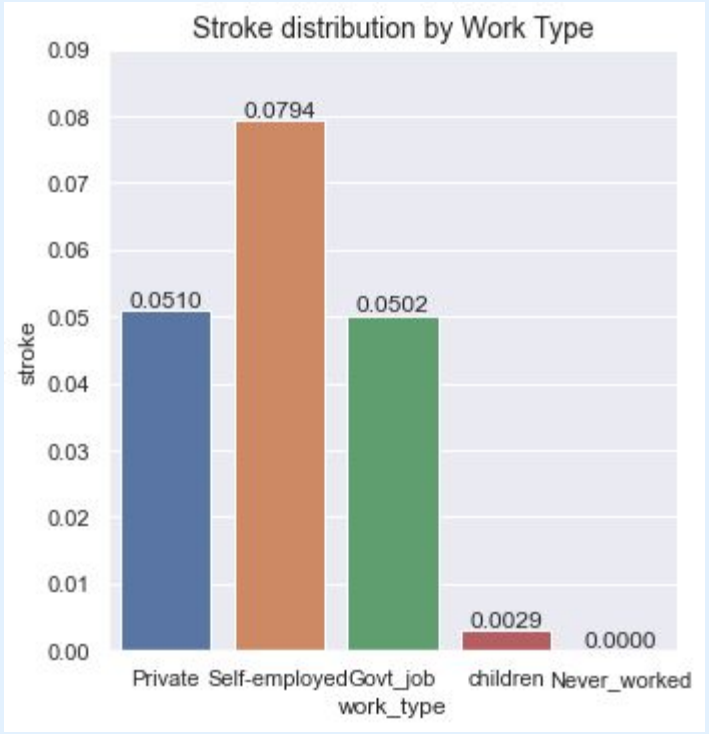
Stroke rate is slightly higher among people living in urban areas as compared to rural areas.



5

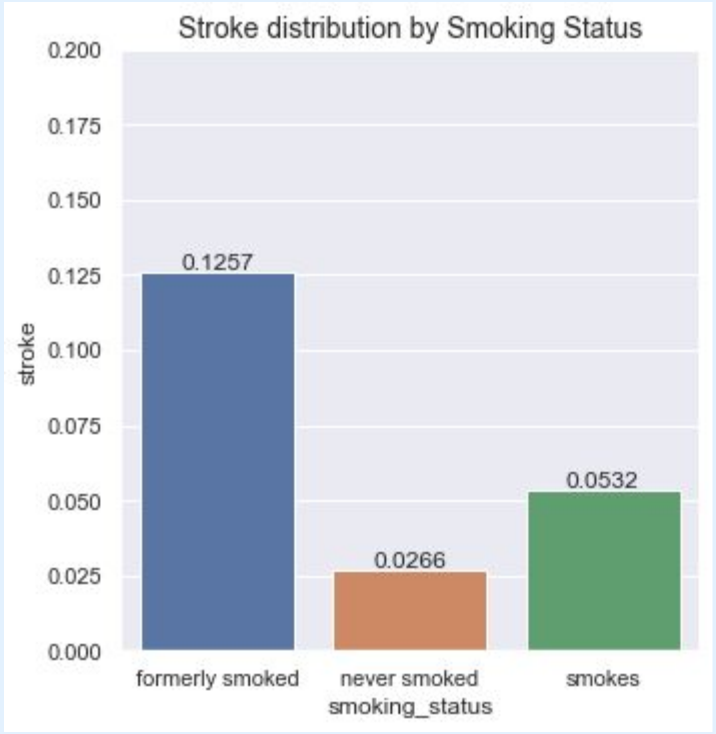
Ever Married

People who have been married have a significantly higher stroke rate than those who have not.



Work Type

Stroke rate among patients who are self-employed are the highest, while stroke rate is the lowest among patients who have never worked.



7

Smoking Status

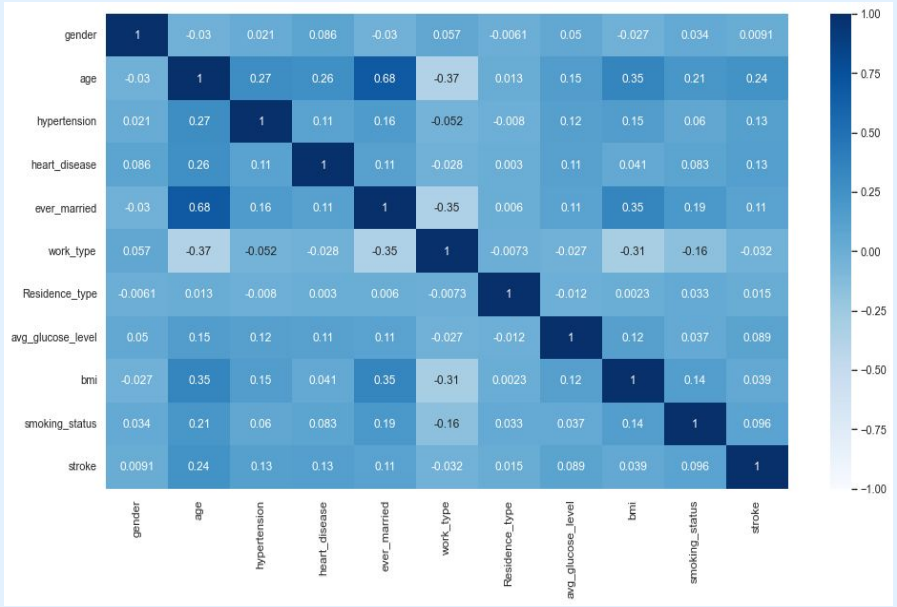
Stroke rate is the highest for patients who formerly smoked, and the lowest for patients who have never smoked.

Highest Linear Correlation

Age **1**

Hypertension **2**

Heart Disease **3**



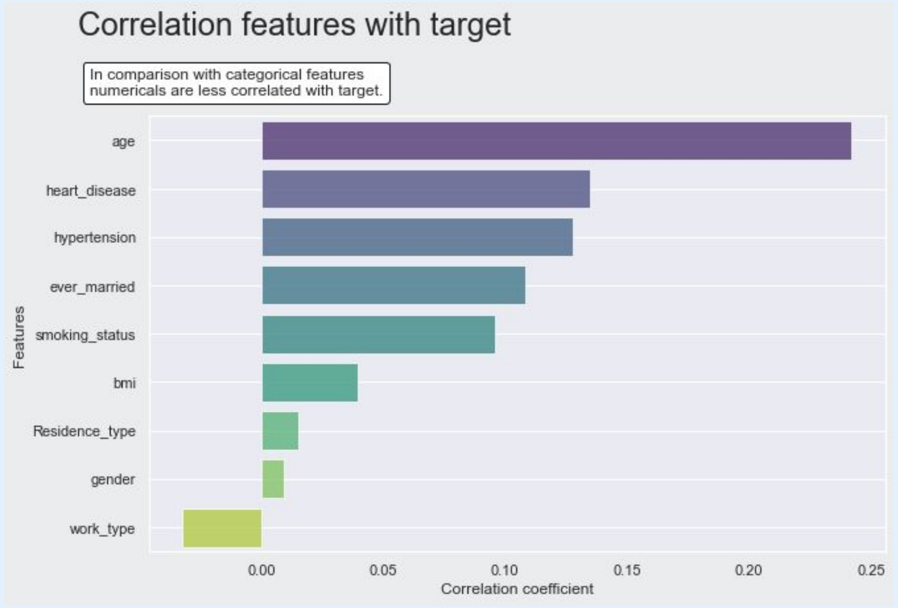
	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
gender	1	-0.03	0.021	0.086	-0.03	0.057	-0.0061	0.05	-0.027	0.034	0.0091
age	-0.03	1	0.27	0.26	0.68	-0.37	0.013	0.15	0.35	0.21	0.24
hypertension	0.021	0.27	1	0.11	0.16	-0.052	-0.008	0.12	0.15	0.06	0.13
heart_disease	0.086	0.26	0.11	1	0.11	-0.028	0.003	0.11	0.041	0.083	0.13
ever_married	-0.03	0.68	0.16	0.11	1	-0.35	0.006	0.11	0.35	0.19	0.11
work_type	0.057	-0.37	-0.052	-0.028	-0.35	1	-0.0073	-0.027	-0.31	-0.16	-0.032
Residence_type	-0.0061	0.013	-0.008	0.003	0.006	-0.0073	1	-0.012	0.0023	0.033	0.015
avg_glucose_level	0.05	0.15	0.12	0.11	0.11	-0.027	-0.012	1	0.12	0.037	0.089
bmi	-0.027	0.35	0.15	0.041	0.35	-0.31	0.0023	0.12	1	0.14	0.039
smoking_status	0.034	0.21	0.06	0.083	0.19	-0.16	0.033	0.037	0.14	1	0.096
stroke	0.0091	0.24	0.13	0.13	0.11	-0.032	0.015	0.089	0.039	0.096	1

Lowest Linear Correlation

Gender 1

Residence Type 2

Body Mass Index 3



Overall Analysis

1 Hypertension and Heart Disease

Presence of such illnesses are important stroke predictors as they imply poor health

2 Marriage

Married individuals seem to have a higher stroke rate, which may be a result of greater likelihood of conflict

3 Age

Stroke victims are often older, likely due to deteriorating health

4 Smoking Status

The stroke rate being the highest among former smokers may be due to the nature of former smokers. 43.2% of ex-smokers mentioned a current health condition as the main reason to stop smoking.

5 Work Type

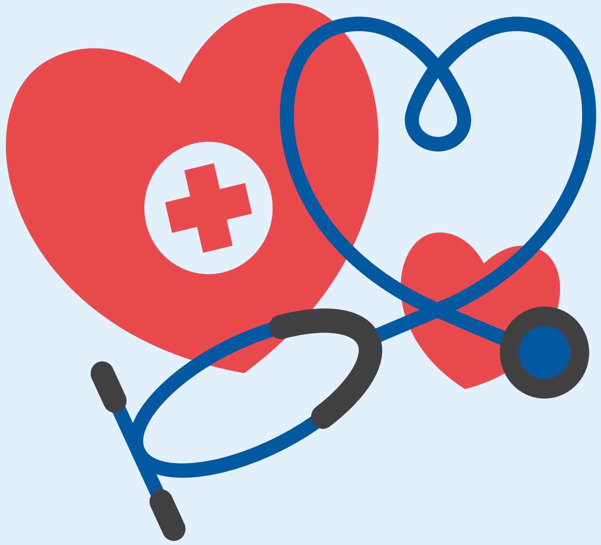
Highest stroke rate among self-employed individuals. A likely reason is that the uncertainty of such a career causes greater stress and hence a higher stroke rate.

6 Conclusion

All factors must be included in our machine learning regardless of significance. Further research in stroke prediction reveals that even factors such as gender can greatly influence stroke rate.

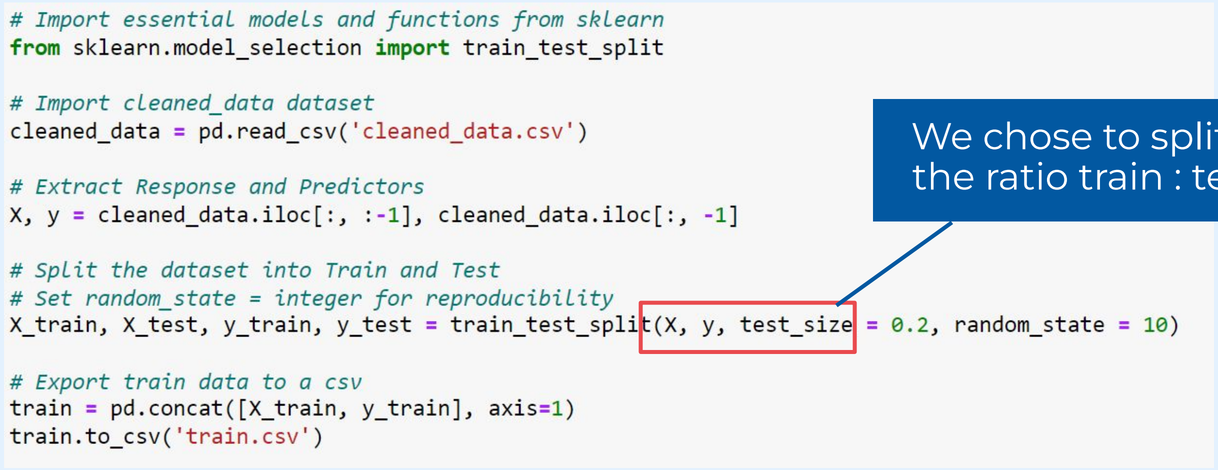
04

Data Splitting & Resampling



Data Splitting

Split data into train and test sets

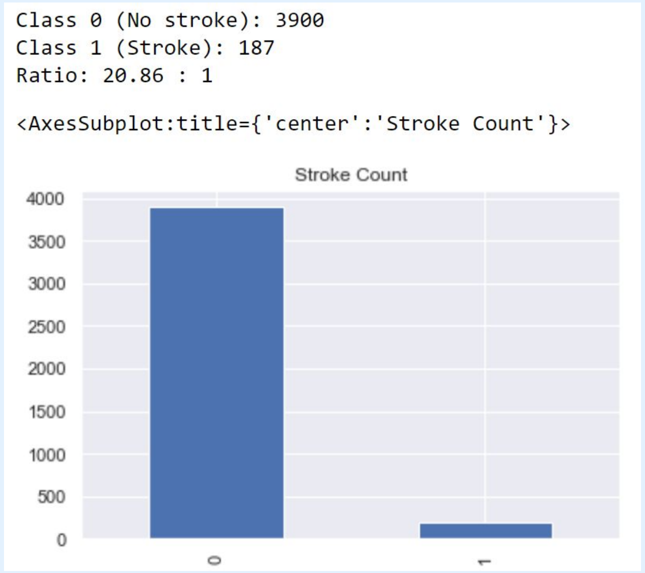


```
# Import essential models and functions from sklearn
from sklearn.model_selection import train_test_split

# Import cleaned_data dataset
cleaned_data = pd.read_csv('cleaned_data.csv')

# Extract Response and Predictors
X, y = cleaned_data.iloc[:, :-1], cleaned_data.iloc[:, -1]

# Split the dataset into Train and Test
# Set random_state = integer for reproducibility
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 10)

# Export train data to a csv
train = pd.concat([X_train, y_train], axis=1)
train.to_csv('train.csv')
```

We chose to split our data in the ratio train : test = 0.8 : 0.2

Problem of Data Imbalance

- Class imbalance is a problem in classification where the **distribution of classes are skewed**.
- When classifiers are faced with imbalanced datasets, where the number of negative instances far outnumbers the positive instances, the **performance drops significantly**.
- Classifiers **tend to predict the majority class** data and ignore the minority class



```
Class 0 (No stroke): 3900
Class 1 (Stroke): 187
Ratio: 20.86 : 1

<AxesSubplot:title={'center':'Stroke Count'}>
```

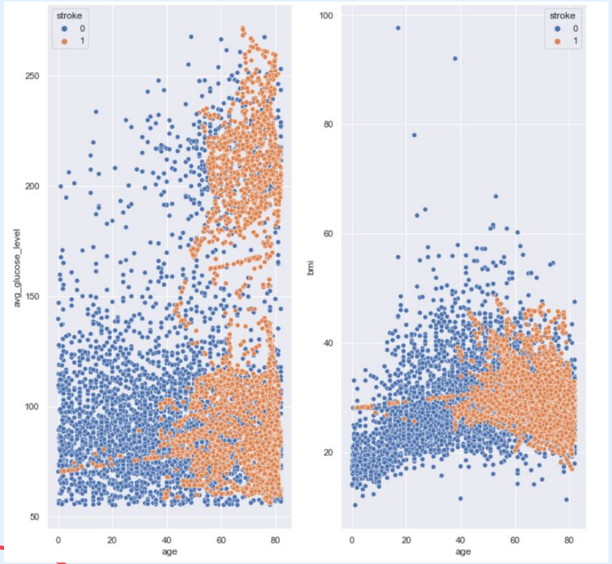
The number of instances where the patient does not have stroke is far greater (**20 times**) as compared to that where the patient has =stroke.

Oversampling Techniques

Goal: To add more samples to the minority class (‘Stroke’)

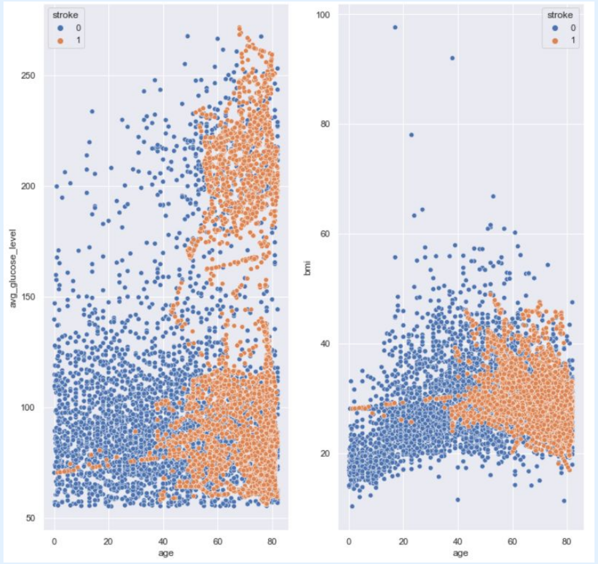
SMOTE **SMOTEENN** **ADASYN**

SMOTE



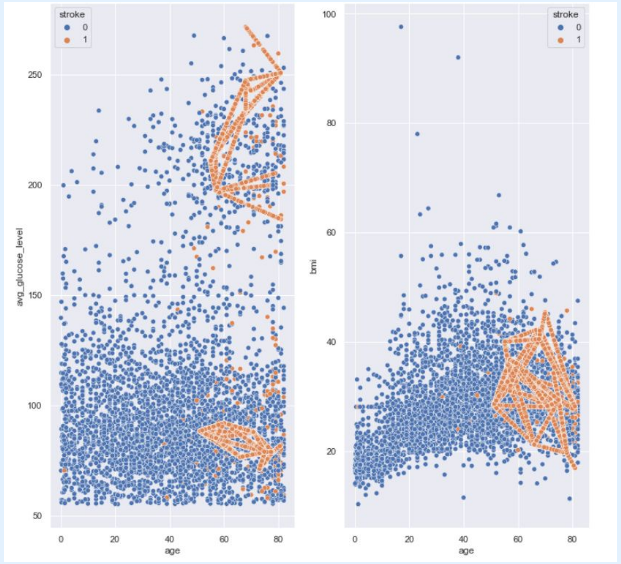
Stroke : 3900
No Stroke: 3900

SMOTEENN



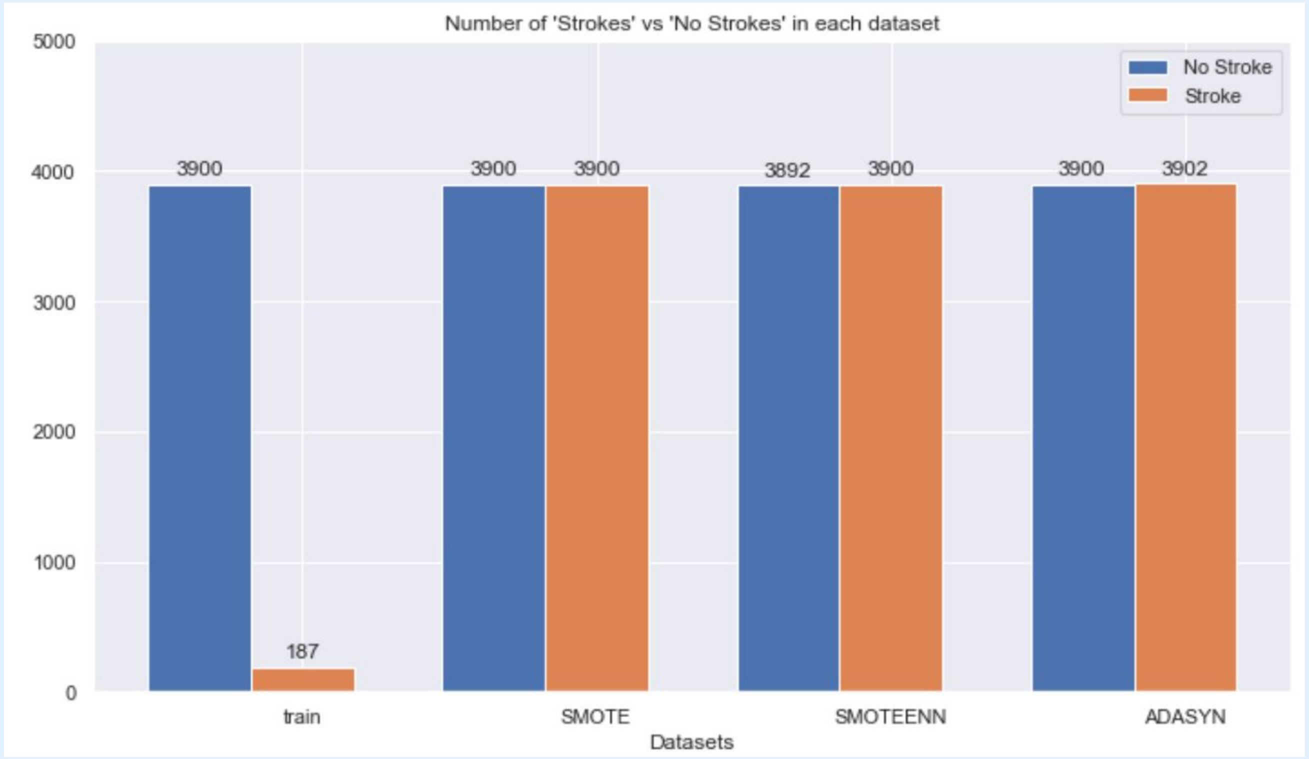
Stroke : 3892
No Stroke: 3900

ADASYN



Stroke : 3902
No Stroke: 3900

After oversampling:



05

Machine Learning

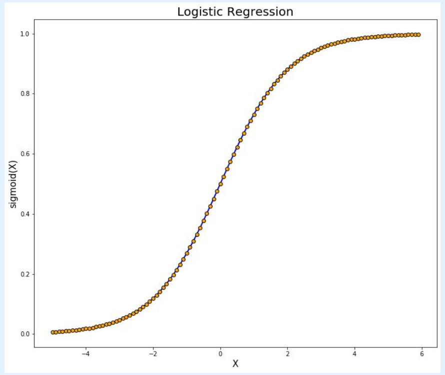
Machine Learning Problem

- **Problem Identified: Classification**
  - Classifying whether a person **would or would not suffer from a stroke**, under the influence of other variables
  - To tackle this problem, we will use **classification models (logistic regression and decision tree)**
- **Why and How does it meet our objective**
  - Classification machine learning models help us **find out which sampling method is most effective** in training our classification model
  - With that sampling method, we can **determine the better of the machine learning models used**

Goal: To determine if the model can be used to predict a patient's risk of getting a stroke accurately, based on his/her lifestyle factors.

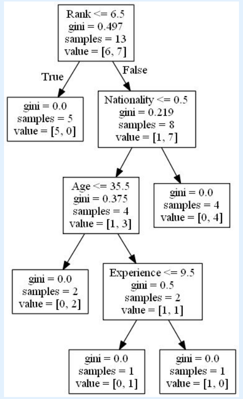
Machine Learning Models: Classification

Logistic Regression



Logistic Regression is used to calculate or **predict the probability of a binary event** occurring.

Decision Trees



Decision trees are constructed via an algorithmic approach that **identifies ways to split a data set** based on different conditions.

Why did we choose these models?

- **Supports our existing data**
  a. Both models are **supervised learning algorithms** that can be trained on data, which has **input and output labels**
  b. Both logistic models support use of **continuous data and categorical data**

- **Nature of Prediction**
  - Both Logistic Regression and Decision Trees are classification models that **solve classification problems**
  - Prediction is **categorical and binary** (Stroke vs No Stroke)

Applying the Machine Learning Model

Step 1: Importing required items

Essential libraries:

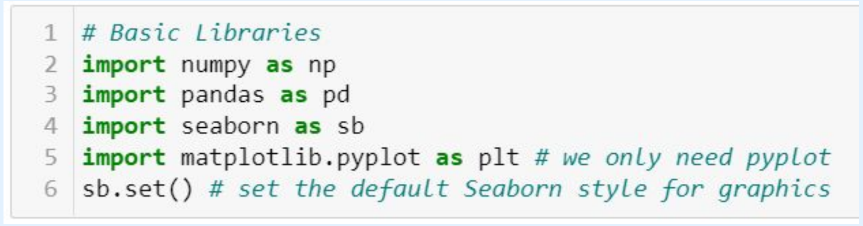
- Numpy
- Pandas
- Matplotlib
- Seaborn

Required Models (sklearn):

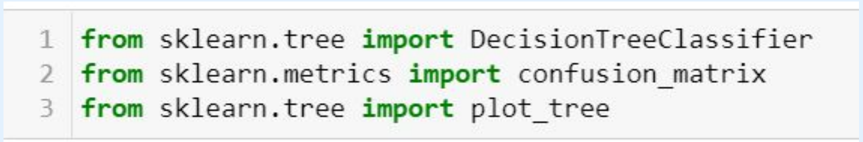
- DecisionTreeClassifier
- Logistic Regression

Datasets:

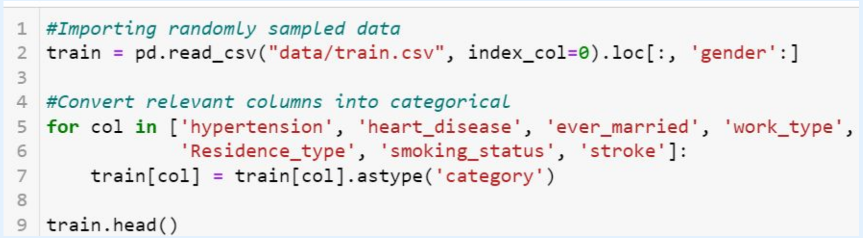
- Random Sampling
- SMOTE
- SMOTEENN
- ADASYN



```python
# Basic Libraries
import numpy as np
import pandas as pd
import seaborn as sb
import matplotlib.pyplot as plt # we only need pyplot
sb.set() # set the default Seaborn style for graphics
```



```python
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import confusion_matrix
from sklearn.tree import plot_tree
```



```python
#Importing randomly sampled data
train = pd.read_csv("data/train.csv", index_col=0).loc[:, 'gender':]

#Convert relevant columns into categorical
for col in ['hypertension', 'heart_disease', 'ever_married', 'work_type',
            'Residence_type', 'smoking_status', 'stroke']:
    train[col] = train[col].astype('category')

train.head()
```

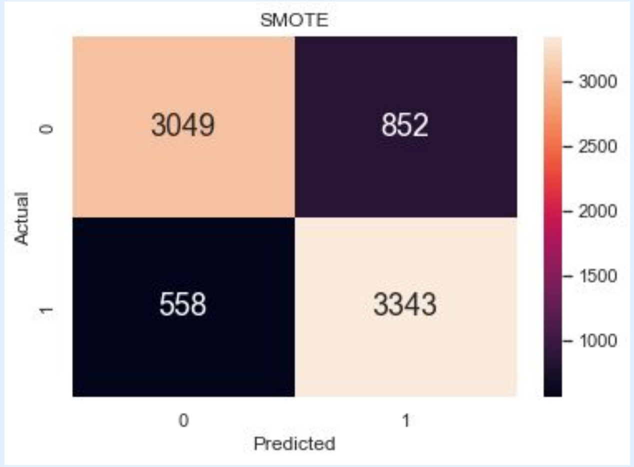
Step 2: Training and Applying the Model

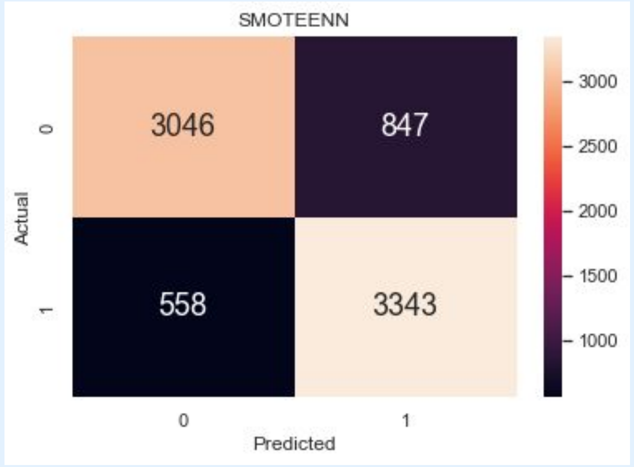
<u>For differently sampled training data</u>

A. Split the training data into predictor and response

B. Train and apply the model on training and test data

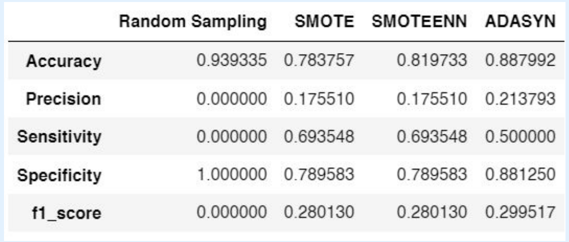
C. Plot the results in a Confusion Matrix



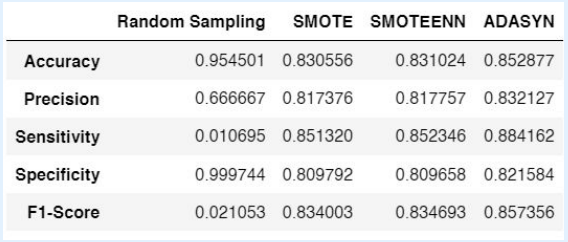


Step 3: Comparison of the different sampling methods

A. **Utilise data from the confusion matrix and formulate the performance evaluation measures**

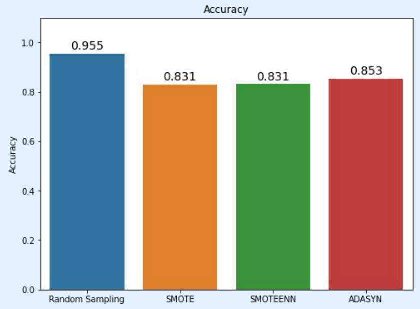


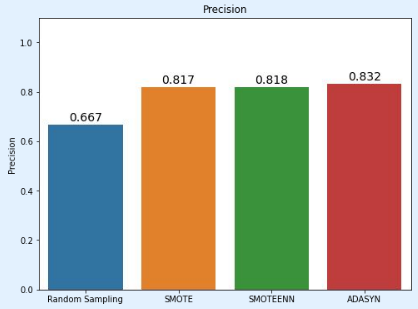
	Random Sampling	SMOTE	SMOTEENN	ADASYN
Accuracy	0.939335	0.783757	0.819733	0.887992
Precision	0.000000	0.175510	0.175510	0.213793
Sensitivity	0.000000	0.693548	0.693548	0.500000
Specificity	1.000000	0.789583	0.789583	0.881250
f1_score	0.000000	0.280130	0.280130	0.299517

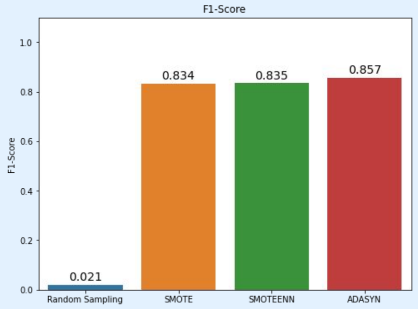


	Random Sampling	SMOTE	SMOTEENN	ADASYN
Accuracy	0.954501	0.830556	0.831024	0.852877
Precision	0.666667	0.817376	0.817757	0.832127
Sensitivity	0.010695	0.851320	0.852346	0.884162
Specificity	0.999744	0.809792	0.809658	0.821584
F1-Score	0.021053	0.834003	0.834693	0.857356

B. **Compare performance measures across the sampling methods using bar charts**







Performance Evaluation

We considered 5 performance metrics to evaluate the performance of our classification models:

- **Accuracy** (The number of classifications a model correctly predicts divided by the total number of predictions made.)

$$\text{Accuracy} = \frac{TruePositive + TrueNegative}{TotalSample}$$

- **Precision** (The ability of a classification model to identify only the relevant data points.)

$$\text{Precision} = \frac{TruePositives}{TruePositives + FalsePositives}$$

- **Sensitivity** (Evaluates a model's ability to predict true positives of stroke.)

$$\text{Sensitivity} = \frac{TruePositives}{TruePositives + FalseNegatives}$$

Performance Evaluation

- **Specificity** (Evaluates a model's ability to predict true negatives of stroke, equivalent to no stroke.)

$$\text{Specificity} = \frac{TrueNegatives}{TrueNegatives + FalsePositives}$$

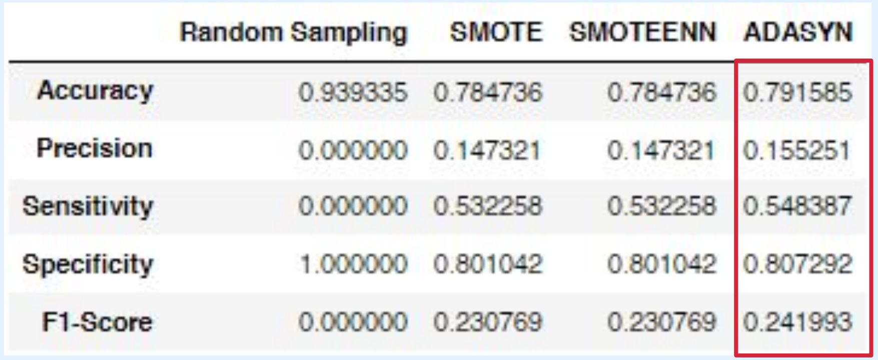
- **F1-score** (Overall measure that combines precision and recall, where a good F1-score means that you are correctly identifying the relevant class and are not influenced by false alarms.)

$$\text{F1-score} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

recall is the same as sensitivity mentioned earlier

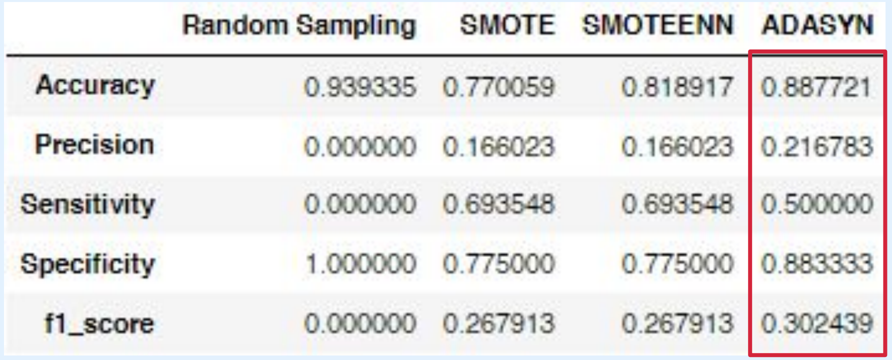
Step 4: Analysing the results

Logistic Regression



	Random Sampling	SMOTE	SMOTEENN	ADASYN
Accuracy	0.939335	0.784736	0.784736	0.791585
Precision	0.000000	0.147321	0.147321	0.155251
Sensitivity	0.000000	0.532258	0.532258	0.548387
Specificity	1.000000	0.801042	0.801042	0.807292
F1-Score	0.000000	0.230769	0.230769	0.241993

Classification Tree



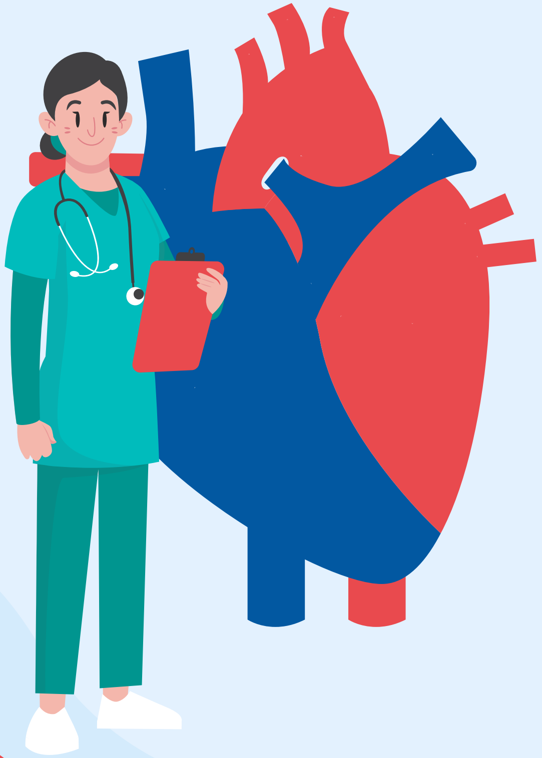
	Random Sampling	SMOTE	SMOTEENN	ADASYN
Accuracy	0.939335	0.770059	0.818917	0.887721
Precision	0.000000	0.166023	0.166023	0.216783
Sensitivity	0.000000	0.693548	0.693548	0.500000
Specificity	1.000000	0.775000	0.775000	0.883333
f1_score	0.000000	0.267913	0.267913	0.302439

ADASYN is the better oversampling method.

- On average, ADASYN produced the best results out of all oversampling methods.

Decision Tree is the better classification model.

- Overall, classification tree produced higher scores for each performance metric as compared to logistic regression.

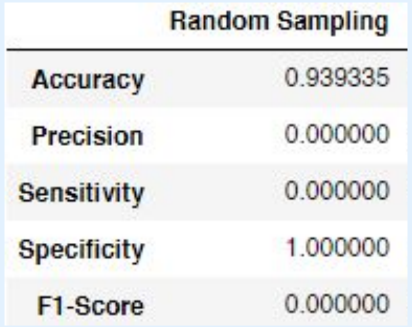


06

Conclusion

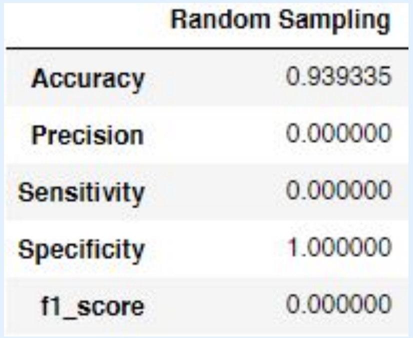
Evaluating Performance

Logistic Regression



	Random Sampling
Accuracy	0.939335
Precision	0.000000
Sensitivity	0.000000
Specificity	1.000000
F1-Score	0.000000

Classification Tree



	Random Sampling
Accuracy	0.939335
Precision	0.000000
Sensitivity	0.000000
Specificity	1.000000
f1_score	0.000000

Analysis:

For the randomly sampled train data, we have obtained the same values for each performance metric.

- Classification accuracy of 93% is high, but this is **misleading** given that the dataset we are using is imbalanced.
- An imbalanced dataset with a large number of “negative data” will give a larger proportion of True Negatives, and this will give a higher accuracy.
- This is insufficient to tell us if the model can accurately detect the “positives” in the dataset.
- Therefore, using randomly sampled data is not an accurate measure of the model’s performance.

Insights

- SMOTE and SMOTEENN produced **similar performance metrics**
  - Could be a result of similar oversampling techniques, both involves finding the K-nearest neighbor of each observation first.
- Resampling unbalanced data **improved model performance**, as seen from increased performance metrics of precision, sensitivity and f1-score.
  - This is because the classification models we have used tend to predict the majority class and ignore the minority class. Thus, the minority class is more likely to be misclassified.
- Across the different oversampling methods, **ADASYN performed best**.
- **Classification tree is a better model** as compared to logistic regression.
- Achieving a classification accuracy of 88% using ADASYN and decision trees, our model **can predict** whether a person is likely to get a stroke.

What we learned

- Techniques to **deal with null values** in the raw dataset (E.g.: Removing/Filling up with mean or median/Making predictions based on probability)
- Use of **different types of graphs/visualisation tools** to improve analysis of the dataset
- Recognise the **significance of class imbalance** in datasets
- **Resampling Techniques** including SMOTE and ADASYN
- Applying **logistic regression and decision tree** on real-life datasets
- **Performance metrics** such as precision, specificity, sensitivity, and F1-score

Moving forward

- Our opinion is that the dataset is still relatively small, and a **larger dataset** could be more practical
- Utilise **more sampling methods** to increase reliability of resampled datasets
- **Alter the hyperparameters** for algorithmic optimisation (E.g.: Depth of decision trees)
- Explore **other performance metrics** to make more informed conclusions
- Obtain **more domain knowledge** for more effective feature selection

References

- https://elitedatascience.com/imbalanced-classes
- https://www.kaggle.com/code/rafjaa/resampling-strategies-for-imbalanced-datasets/notebook
- https://www.analyticsvidhya.com/blog/2020/11/popular-classification-models-for-machine-learning/
- https://www.analyticsvidhya.com/blog/2021/07/an-introduction-to-logistic-regression/
- https://www.analyticsvidhya.com/blog/2020/12/decluttering-the-performance-measures-of-classification-models/
- https://www.analyticsvidhya.com/blog/2021/07/metrics-to-evaluate-your-classification-model-to-take-the-right-decisions/
- https://www.analyticsvidhya.com/blog/2020/10/how-to-choose-evaluation-metrics-for-classification-model/
- https://wiki.pathmind.com/accuracy-precision-recall-f1#:~:text=That%20is%2C%20a%20good%20F1,total%20failure%20when%20it's%200%20.
- https://learn.g2.com/logistic-regression
- https://blog.hexstream.com/how-to-choose-the-right-machine-learning-algorithm#:~:text=Knowledge%20of%20Data%3A%20The%20data's,place%20for%20a%20given%20analysis
- https://miro.medium.com/max/725/1*QY3CSyA4BzAU6sEPFwp9ZQ.png

Thank you.