# PRML COURSE PROJECT

## STROKE PREDICTION - PROJECT REPORT

Group Members:
Aditya Raj (B20CS089)
Tanmay (B20AI047)
Akhilesh Bhasale (B20EE006)

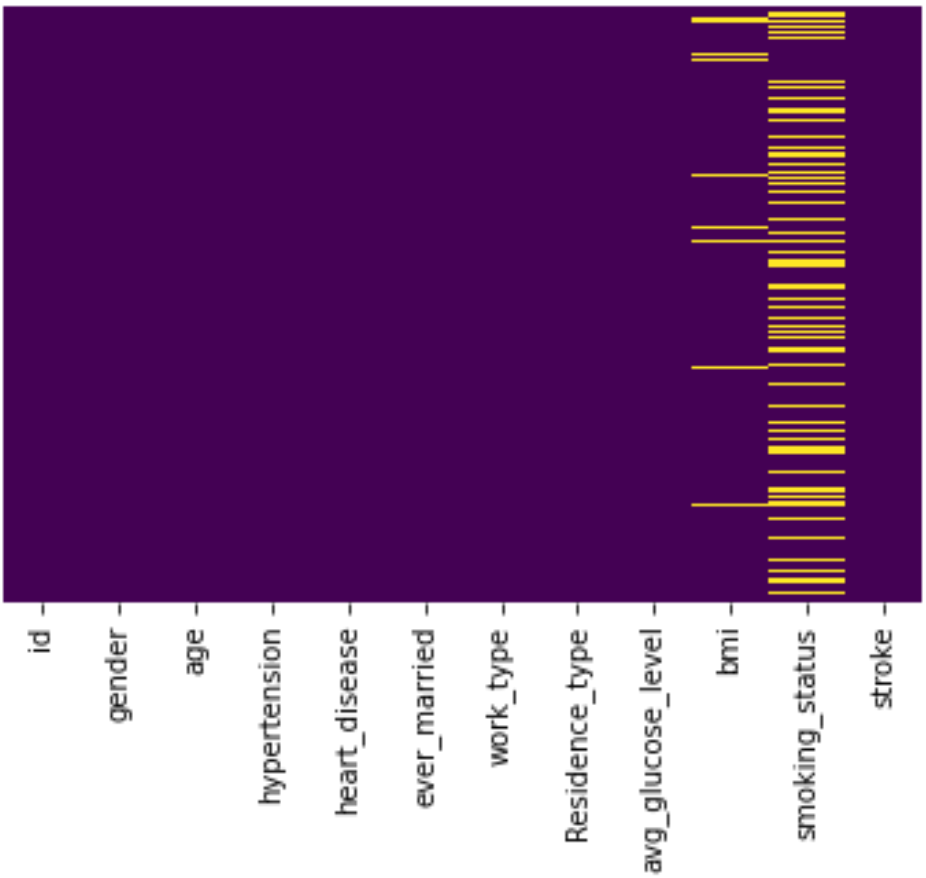
01

# ABSTRACT:

The objective of this project is to predict whether a patient is at a risk of stroke or not. According to WHO, strokes account for approximately 11% of total deaths.
This dataset is used to predict whether a patient is likely to get a stroke based on input parameters like gender, age, various diseases, and smoking status. Each row provides relevant information about the patient.

# DATA PRE-PROCESSING AND ANALYSIS:

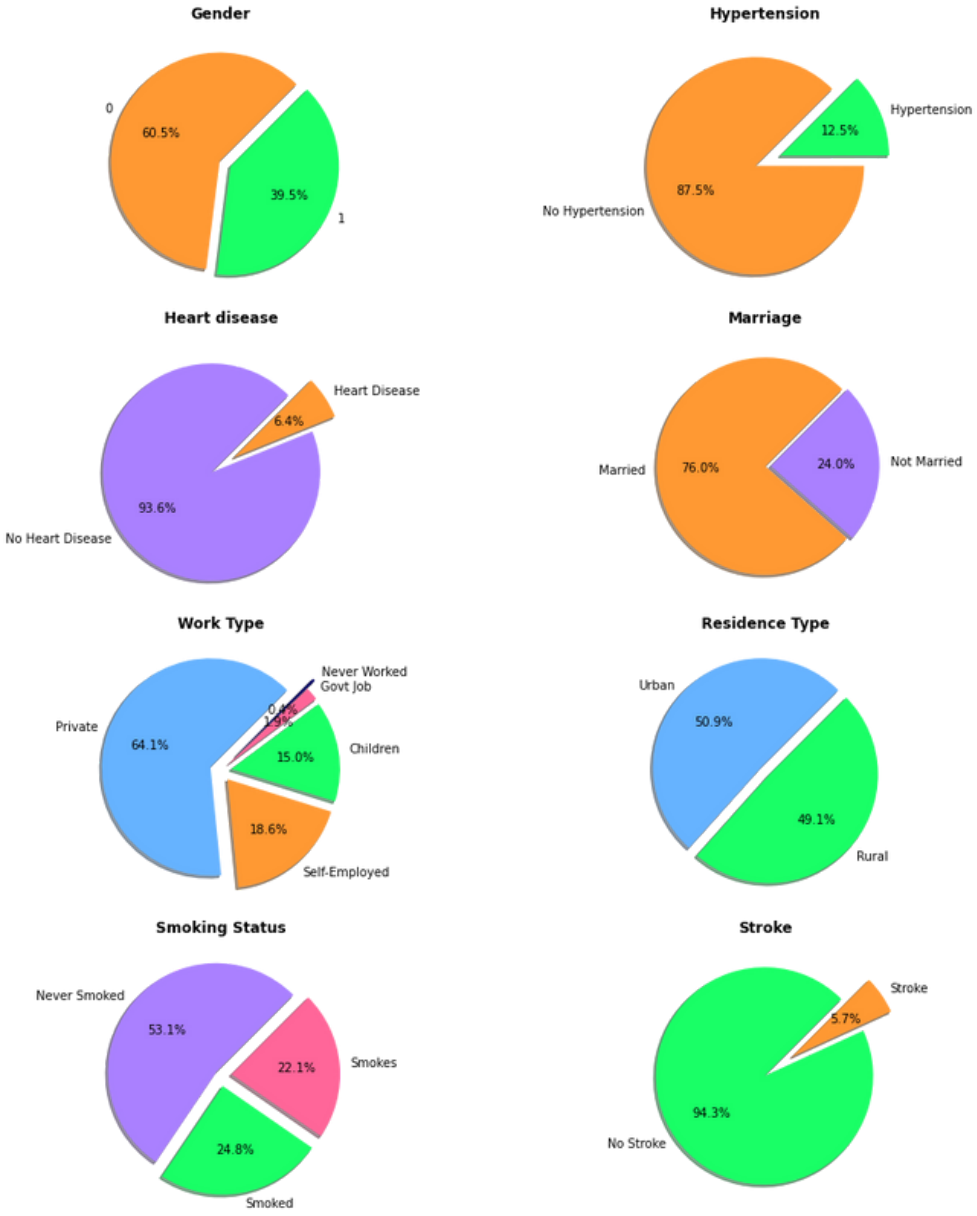
- We will start by processing the information so that we can take care of any issues that can cause problems when we try to use our models to predict data.
- We will first look for any missing values. We can see that some values are missing from the BMI and smoking status section so we eliminate those values from our data. The following plot is a heat map showing the missing values.
- Furthermore we also count the outlier values for BMI since outliers increase the variability in our data, which decreases statistical power.




*Heat map of missing values.*

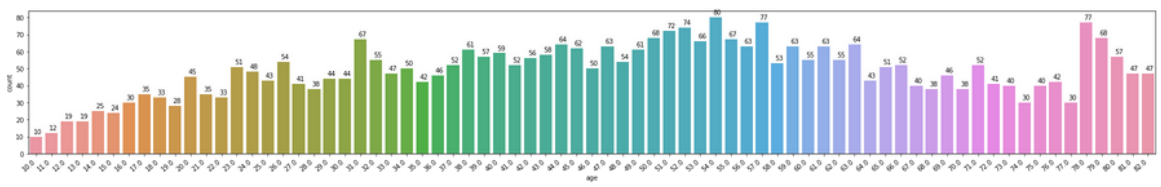
- Consequently, excluding outliers can cause our results to become statistically significant. We replace those values with the mean of all remaining BMI values.
- After looking at the gender values we can notice that there are three genders but only one value belongs to the third category. Thus we can replace this with a ‘Male’ or ‘Female’ value to reduce variability.
- Now we can observe the data and create plots to find out some sort of co-relation. We use the corr() function for this and plot it’s result. This allows us to select highly co-related features for evaluation. The following is a plot that represents the co-relation graph.

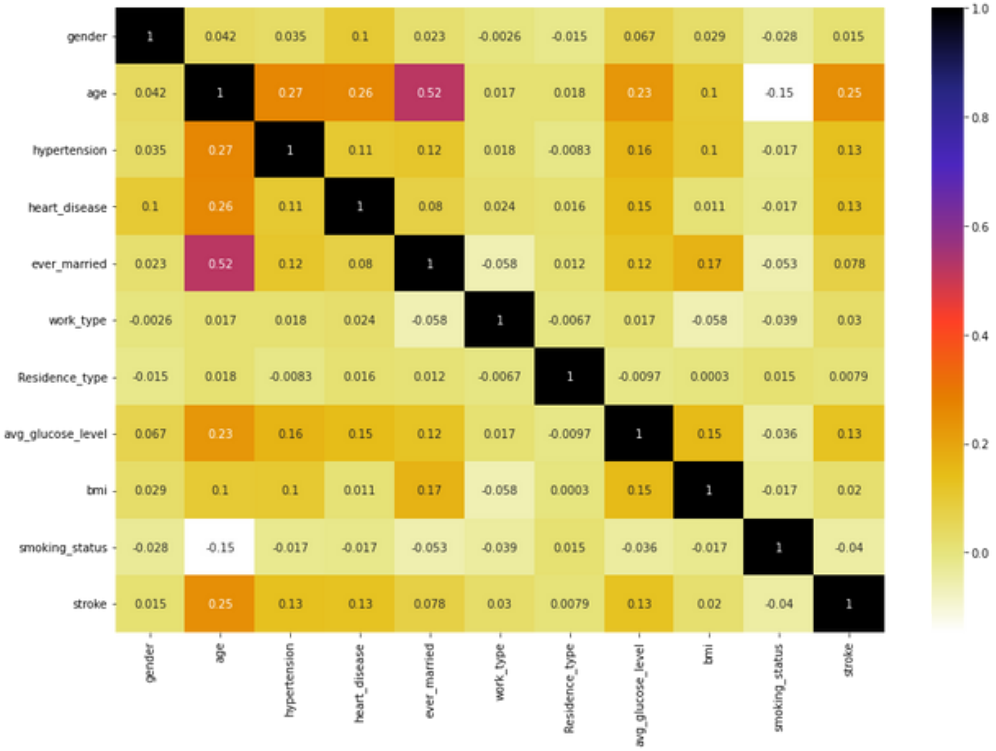
# 03




*The above is the pie chart distribution of the data set.*

We also have some numerical features like age that have the following representation.



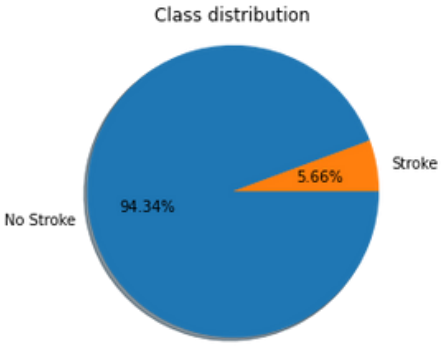


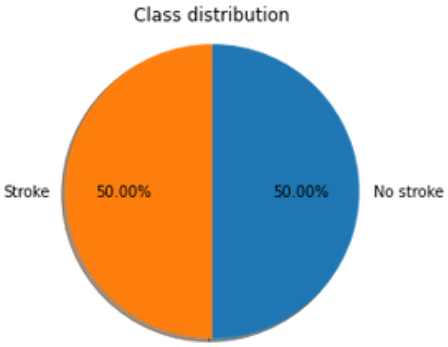

*The grid plotted above is the co-relation heatmap.*

# EXPERIMENTATION AND IDEAS:

- We can now move on to creating a model for training and testing.
- We have a large number of observations in our training data. Number features < number of observations as well. Furthermore there is a mix of categorical and numerical data. So we decided to use XGBoost and LightGBM.
- Stroke prediction is essentially a classification problem. Thus the chosen classifiers were Decision Tree Classifier, MLP Classifier, Random Forest Classifier, Histogram-gradient Boosting Classifier, and Naive-Bayes.

Furthermore our data was very imbalanced. The number of experimental values that suggest stroke were very less. Thus in order to train an effective model we will use SMOTE (Synthetic Minority Over-sampling Technique) to balance our data.




# RESULTS:

- We first trained our data without SMOTE and with SMOTE. SMOTE models gave us a better accuracy. The following are the model scores.
- We hyper tuned the parameters for the top 4 models. The corresponding values for the hyper tuned parameters were as follows.

| | ROC_AUC | Recall | Precision | F1_Score | Accuracy |
|---|---|---|---|---|---|
| **XGBoost** | 98.535518 | 95.152419 | 92.765105 | 93.941026 | 94.530321 |
| **LightGBM** | 98.277711 | 94.794960 | 91.573143 | 93.150745 | 93.162901 |
| **DTC** | 87.420012 | 89.431427 | 85.838787 | 87.763570 | 88.466112 |
| **Logistic Regression** | 88.224677 | 80.650936 | 78.293264 | 79.451398 | 78.894174 |
| **NeuralNetwork** | 93.845870 | 90.662848 | 85.325318 | 87.771946 | 87.871581 |
| **NaiveBayes** | 86.219840 | 80.253558 | 76.143744 | 78.138650 | 77.348395 |
| **RFC** | 97.878377 | 93.960601 | 90.851712 | 91.783398 | 92.568371 |
| **Histogram Gradient** | 97.623645 | 94.239010 | 90.156680 | 92.152217 | 92.865636 |

XGBoost Classifier

```
{'learning_rate': 0.1 , 'max_depth': 8 , 'min_child_weight': 1 , 'n_estimators': 900}
```

LGBM Classifier

```
{'eta': 0.1, 'max_depth': 6, 'n_estimators': 800}
```

Random Forest Classifier

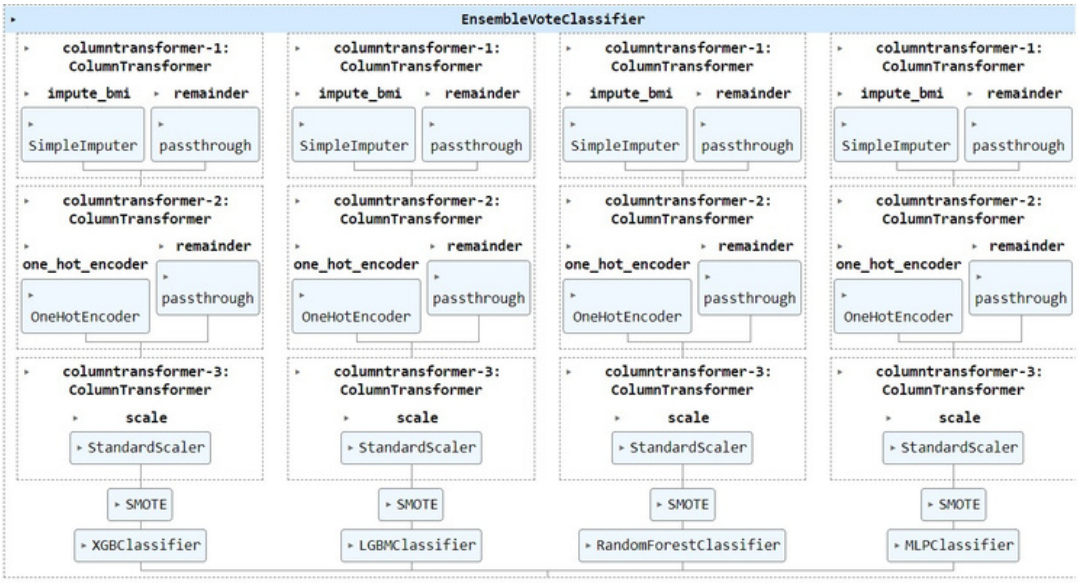
```
{'max_depth': 15, 'max_features': 'sqrt', 'n_estimators': 500}
```

Neural Network

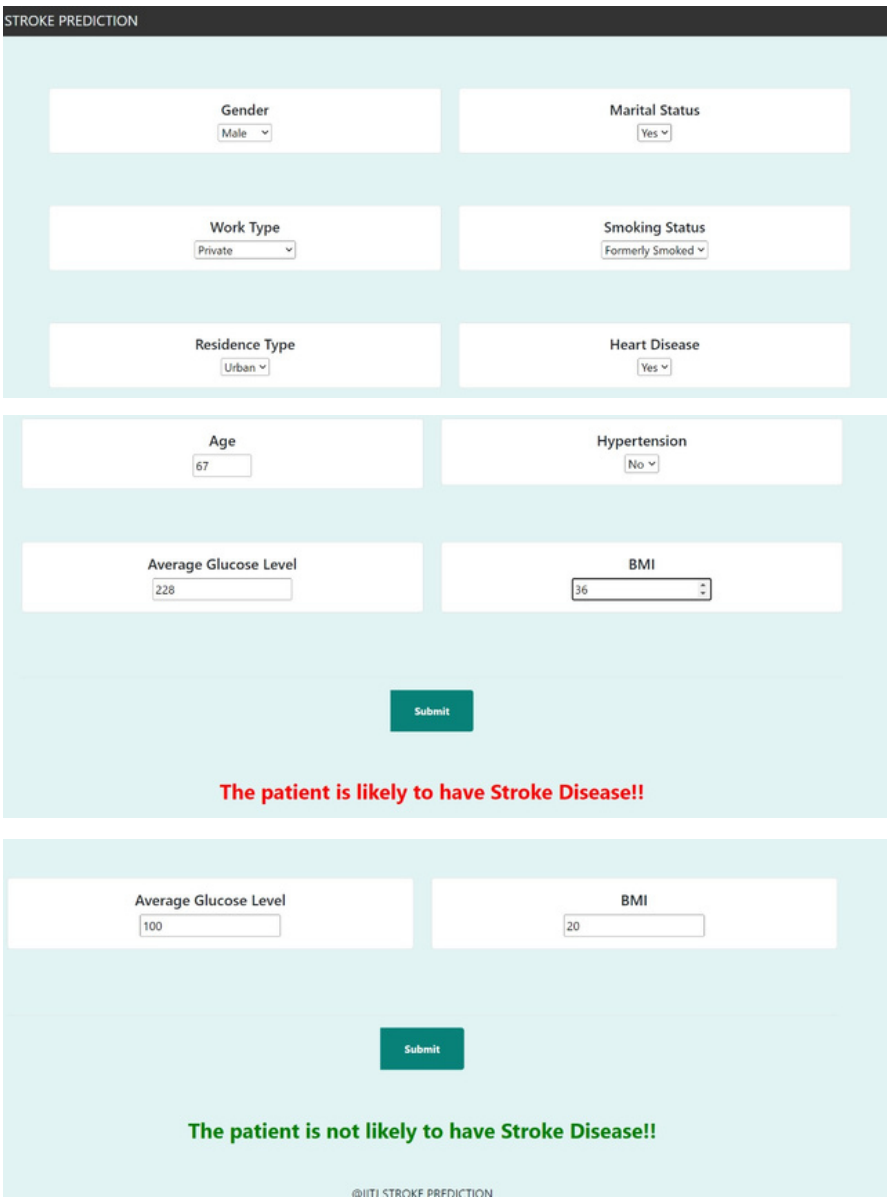
```
{'activation': 'tanh' , 'alpha': 0.0001 , 'hidden_layer_sizes': (16, 20, 22) ,
'learning_rate': 'adaptive' , 'max_iter': 1600 , 'solver': 'adam'}
```

*Hypertuned Parameters*

- Finally we constructed an end to end pipeline and hosted it on GitHub. The pipeline orchestrates the flow of data into, and out from, our machine learning model.




- After hosting the design on GitHub we get the following result. The link for the is: **https://github.com/ADITYA-1602/Stroke_Prediction_ML.git**




## Contributions:

**Aditya Raj:** Data Preprocessing, EDA, model selection, Model comparison(without SMOTE), web app(back-end), report writing and model training after hyperparameter tuning.

**Akhilesh:** SMOTE, Model comparison(with SMOTE), hyperparameter tuning(NLP/RFC), web app(front-end) pipeline(model, vote classifier), report writing and model training after hyperparameter tuning

**Tanmay:** Hyperparameter tuning(XGB/LGBM), pipeline (preprocessing, encoding, SMOTE), web app(front-end) ,report writing and model training after hyperparameter tuning.