SC1015 Mini Project

Stroke Prediction

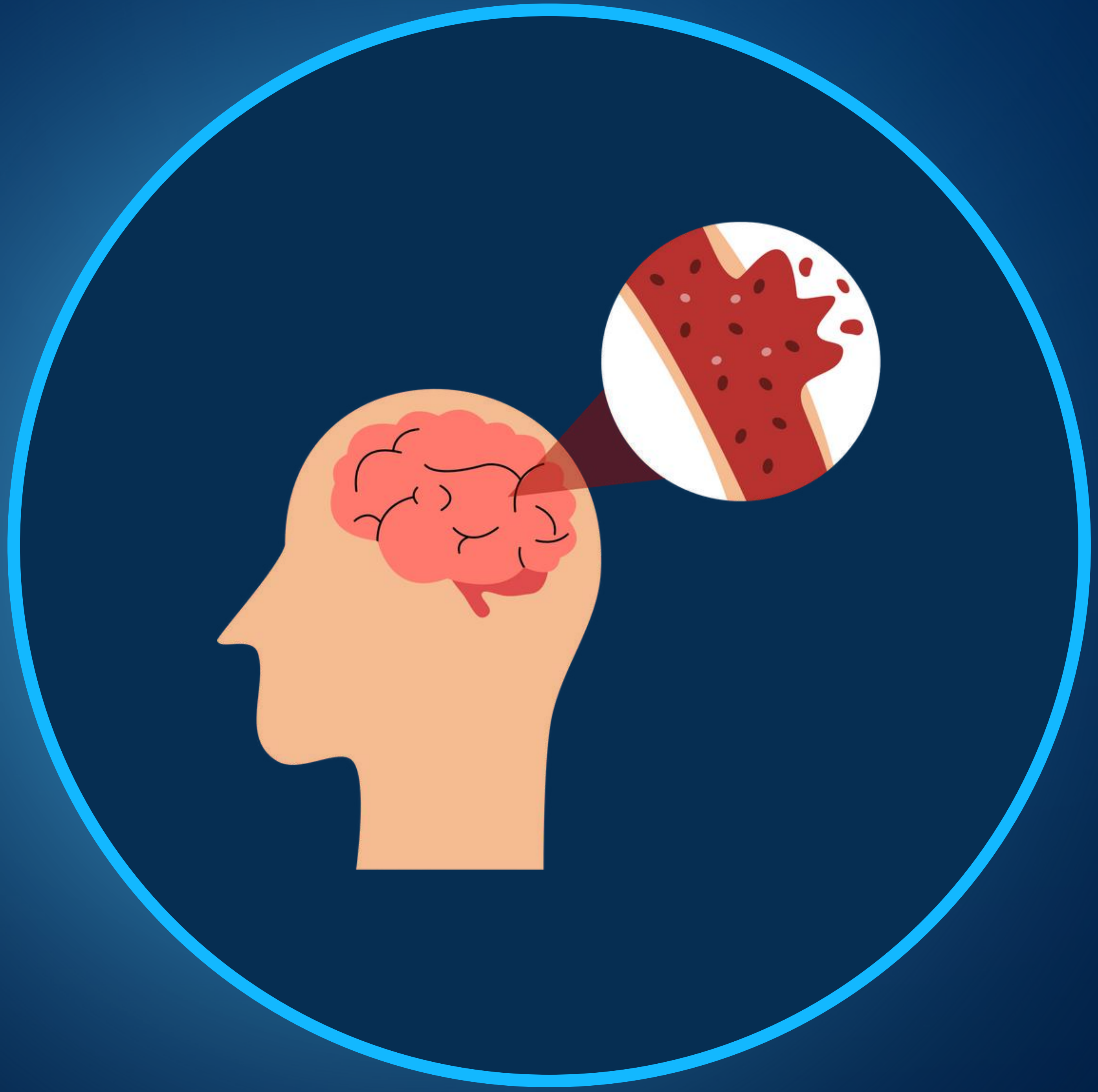
FCS2 Team 9
Tan Jin Yong (U2322914J)
Ryan Ching Kay Joon (U2321286F)

Content

The Problem

World Health Organization (WHO) in 2022:

1. Stroke is the 2nd leading cause of death
2. 1 in 4 people aged >25 estimated to have a stroke in their lifetime
3. From 1990 to 2019, 70% increase risk of stroke
4. 43% increase in death due to stroke



Practical Motivation

- People do not know the contributing factors to getting a stroke
- ~15 million people worldwide suffer a stroke yearly
- Early Intervention can reduce stroke occurrence
- Improve qualtiy of life
- Reduce cost spent on treatment

Problem Definition

Are we able to identify the contributing factors that causes a stroke to happen?

The Dataset

Stroke Prediction Dataset by Fedesoriano

Stroke Prediction Dataset

11 clinical features for predicting stroke events

Last Updated: 3 years ago (Version 1)

About this Dataset

This dataset is used to predict whether a patient is likely to get stroke based on the input parameters like gender, age, various diseases, and smoking status. Each row in the data provides relavant information about the patient.

The Dataset

5110 row entries, 12 columns of variables → 4867 row entries, 11 + 1 columns of variables

```
Shape of data is: (5110, 12)


<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5110 entries, 0 to 5109
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   id                 5110 non-null   int64
 1   gender             5110 non-null   object
 2   age                5110 non-null   float64
 3   hypertension       5110 non-null   int64
 4   heart_disease      5110 non-null   int64
 5   ever_married       5110 non-null   object
 6   work_type          5110 non-null   object
 7   Residence_type     5110 non-null   object
 8   avg_glucose_level  5110 non-null   float64
 9   bmi                4909 non-null   float64
 10  smoking_status     5110 non-null   object
 11  stroke             5110 non-null   int64
dtypes: float64(3), int64(4), object(5)
memory usage: 479.2+ KB
None
```

```
Shape of data is: (4867, 12)


<class 'pandas.core.frame.DataFrame'>
Index: 4867 entries, 0 to 5109
Data columns (total 12 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   gender                    4867 non-null   object
 1   age                       4867 non-null   int64
 2   hypertension              4867 non-null   int64
 3   heart_disease             4867 non-null   int64
 4   ever_married              4867 non-null   object
 5   work_type                 4867 non-null   object
 6   Residence_type            4867 non-null   object
 7   avg_glucose_level         4867 non-null   float64
 8   bmi                       4867 non-null   float64
 9   smoking_status            4867 non-null   object
 10  stroke                    4867 non-null   int64
 11  smoking_status_numerical  4867 non-null   int64
dtypes: float64(2), int64(5), object(5)
memory usage: 494.3+ KB
None
```

The Dataset - Cleaning

1. Converted ‘age’ from float to int data type
2. Dropped ‘age’ rows with ‘age’ = 0
3. Dropped ‘id’ column
4. Mapped smoking _status to numerical values

```python
#Drop the unnecessary column and row

#The ID row is of no use for us
cleancsv = datacsv.drop(['id'], axis=1)

#We notice some of the BMI values are NaN. It would not make sense for us to replace it
#it will affect the overall dataset. Hence it would be better to remove the whole row
cleancsv = cleancsv.drop(cleancsv[cleancsv['bmi'].isna()].index)

#We shall convert the age from float to int, as we notice there are weird values of e.g.
cleancsv['age'] = cleancsv['age'].astype('int64')
cleancsv = cleancsv[cleancsv['age'] != 0]

#We convert the status of smoking into numerical values
#Mapping
smoking_mapping = {'never smoked': 1, 'smokes': 2, 'formerly smoked': 3, 'Unknown': 0}
#Apply mapping
cleancsv['smoking_status_numerical'] = cleancsv['smoking_status'].map(smoking_mapping)
```

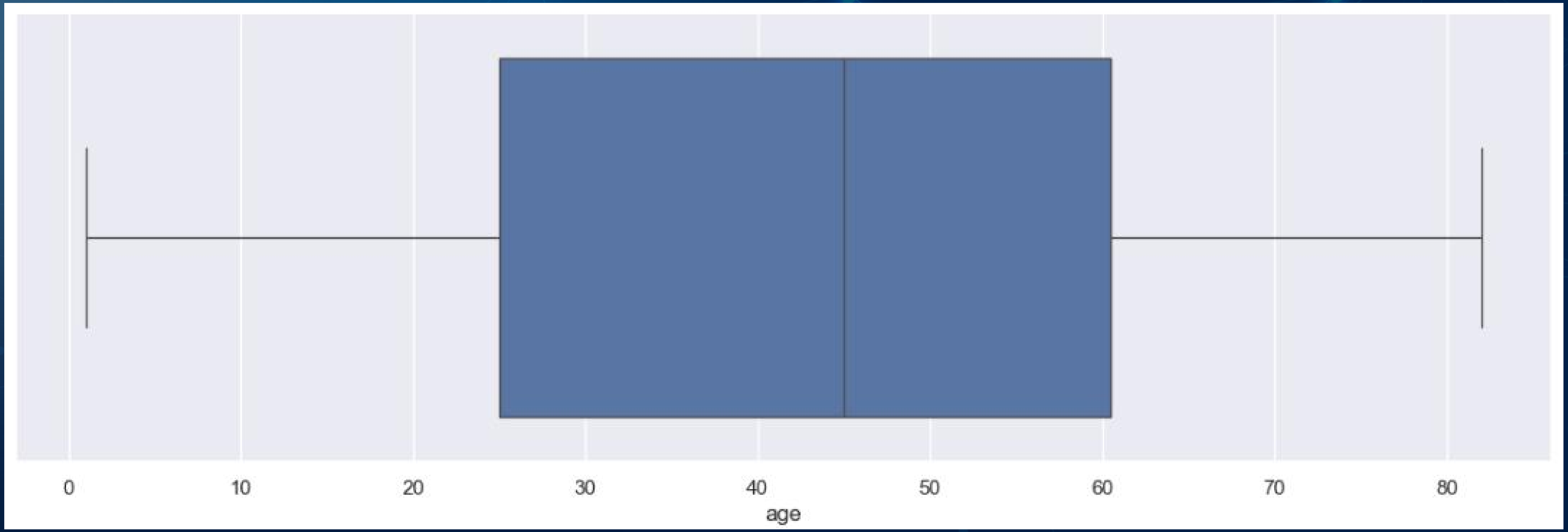
The Dataset

A	B	C	D	E	F	G	H	I	J	K	L
id	gender	age	hypertensic	heart_disea	ever_marri	work_type	Residence_	avg_glucos	bmi	smoking_st	stroke
9046	Male	67	0	1	Yes	Private	Urban	228.69	36.6	formerly sn	1
51676	Female	61	0	0	Yes	Self-employ	Rural	202.21	N/A	never smok	1
31112	Male	80	0	1	Yes	Private	Rural	105.92	32.5	never smok	1
60182	Female	49	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
1665	Female	79	1	0	Yes	Self-employ	Rural	174.12	24	never smok	1
56669	Male	81	0	0	Yes	Private	Urban	186.21	29	formerly sn	1
53882	Male	74	1	1	Yes	Private	Rural	70.09	27.4	never smok	1
10434	Female	69	0	0	No	Private	Urban	94.39	22.8	never smok	1
27419	Female	59	0	0	Yes	Private	Rural	76.15	N/A	Unknown	1
60491	Female	78	0	0	Yes	Private	Urban	58.57	24.2	Unknown	1
12109	Female	81	1	0	Yes	Private	Rural	80.43	29.7	never smok	1
12095	Female	61	0	1	Yes	Govt_job	Rural	120.46	36.8	smokes	1
12175	Female	54	0	0	Yes	Private	Urban	104.51	27.3	smokes	1
8213	Male	78	0	1	Yes	Private	Urban	219.84	N/A	Unknown	1
5317	Female	79	0	1	Yes	Private	Urban	214.09	28.2	never smok	1
58202	Female	50	1	0	Yes	Self-employ	Rural	167.41	30.9	never smok	1
56112	Male	64	0	1	Yes	Private	Urban	191.61	37.5	smokes	1
34120	Male	75	1	0	Yes	Private	Urban	221.29	25.8	smokes	1
27458	Female	60	0	0	No	Private	Urban	89.22	37.8	never smok	1
25226	Male	57	0	1	No	Govt_job	Urban	217.08	N/A	Unknown	1
70630	Female	71	0	0	Yes	Govt_job	Rural	193.94	22.4	smokes	1

	A	B	C	D	E	F	G	H	I	J	K	L
1	gender	age	hypertensic	heart_disea	ever_marri	work_type	Residence_	avg_glucos	bmi	smoking_st	stroke	smoking_status_numerical
2	Male	67	0	1	Yes	Private	Urban	228.69	36.6	formerly sn	1	3
3	Male	80	0	1	Yes	Private	Rural	105.92	32.5	never smok	1	1
4	Female	49	0	0	Yes	Private	Urban	171.23	34.4	smokes	1	2
5	Female	79	1	0	Yes	Self-employ	Rural	174.12	24	never smok	1	1
6	Male	81	0	0	Yes	Private	Urban	186.21	29	formerly sn	1	3
7	Male	74	1	1	Yes	Private	Rural	70.09	27.4	never smok	1	1
8	Female	69	0	0	No	Private	Urban	94.39	22.8	never smok	1	1
9	Female	78	0	0	Yes	Private	Urban	58.57	24.2	Unknown	1	0
10	Female	81	1	0	Yes	Private	Rural	80.43	29.7	never smok	1	1
11	Female	61	0	1	Yes	Govt_job	Rural	120.46	36.8	smokes	1	2
12	Female	54	0	0	Yes	Private	Urban	104.51	27.3	smokes	1	2
13	Female	79	0	1	Yes	Private	Urban	214.09	28.2	never smok	1	1
14	Female	50	1	0	Yes	Self-employ	Rural	167.41	30.9	never smok	1	1
15	Male	64	0	1	Yes	Private	Urban	191.61	37.5	smokes	1	2
16	Male	75	1	0	Yes	Private	Urban	221.29	25.8	smokes	1	2
17	Female	60	0	0	No	Private	Urban	89.22	37.8	never smok	1	1
18	Female	71	0	0	Yes	Govt_job	Rural	193.94	22.4	smokes	1	2
19	Female	52	1	0	Yes	Self-employ	Urban	233.29	48.9	never smok	1	1
20	Female	79	0	0	Yes	Self-employ	Urban	228.7	26.6	never smok	1	1
21	Male	82	0	1	Yes	Private	Rural	208.3	32.5	Unknown	1	0

Exploratory Data Analysis

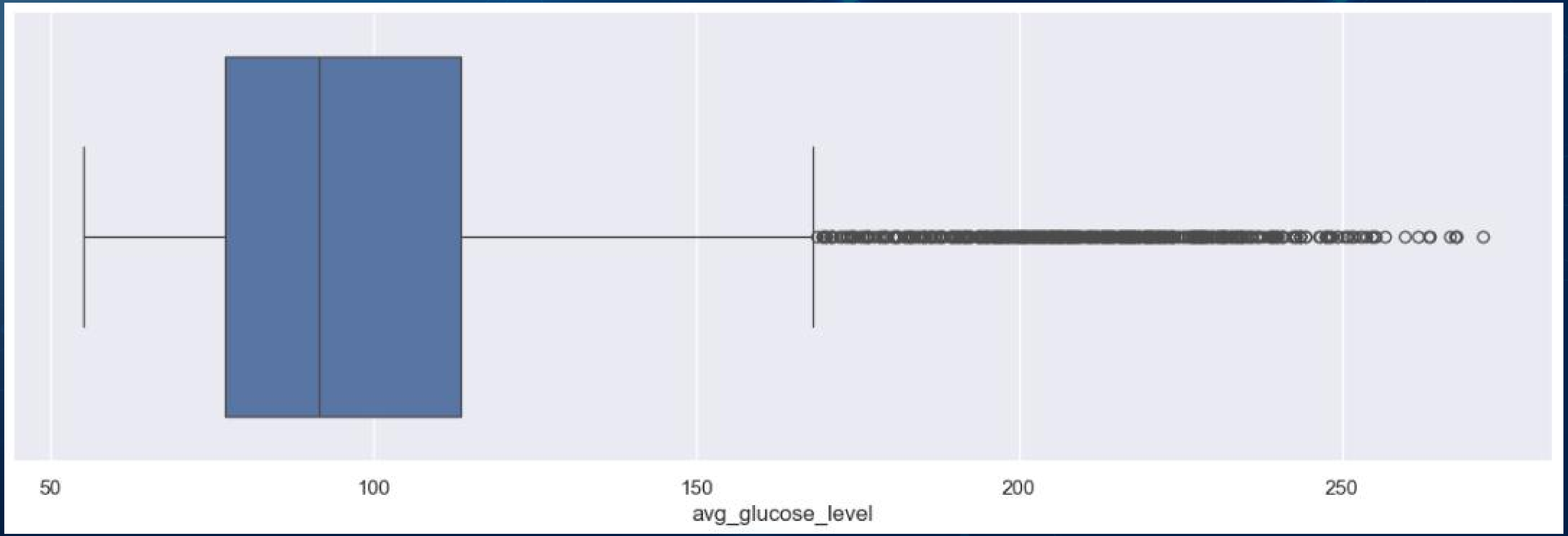
Exploratory Data Analysis



Age

A diverse dataset for age group, with 50% of the data lying between 25 to 60 years old

Exploratory Data Analysis



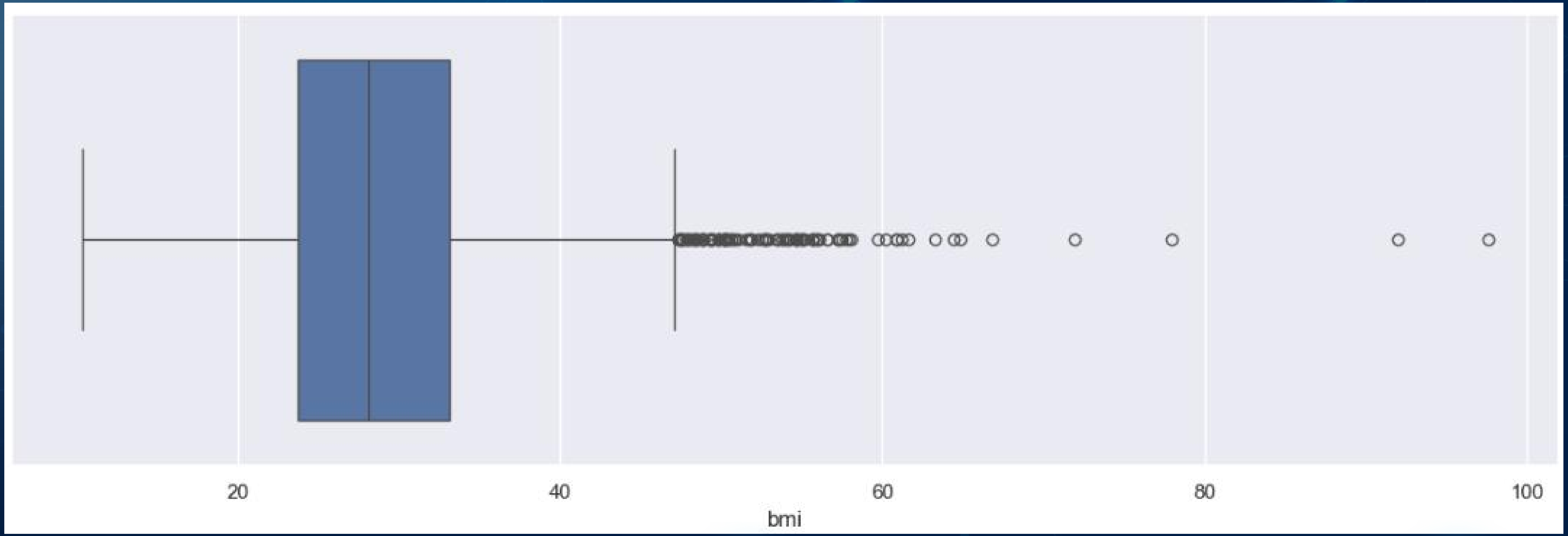
Age

A diverse dataset for age group, with 50% of the data lying between 25 to 60 years old

Average Glucose Level

Significant amount of outliers with average glucose level >168.38mg/dL
Normal range is 72 to 108mg/dL

Exploratory Data Analysis



Age
A diverse dataset for age group, with 50% of the data lying between 25 to 60 years old

Average Glucose Level
Significant amount of outliers with average glucose level >168.38mg/dL
Normal range is 72 to 108mg/dL

Body Mass Index (BMI)
Significant amount of outliers, most of which points to them being obese with a BMI of >47.2kg/m^2

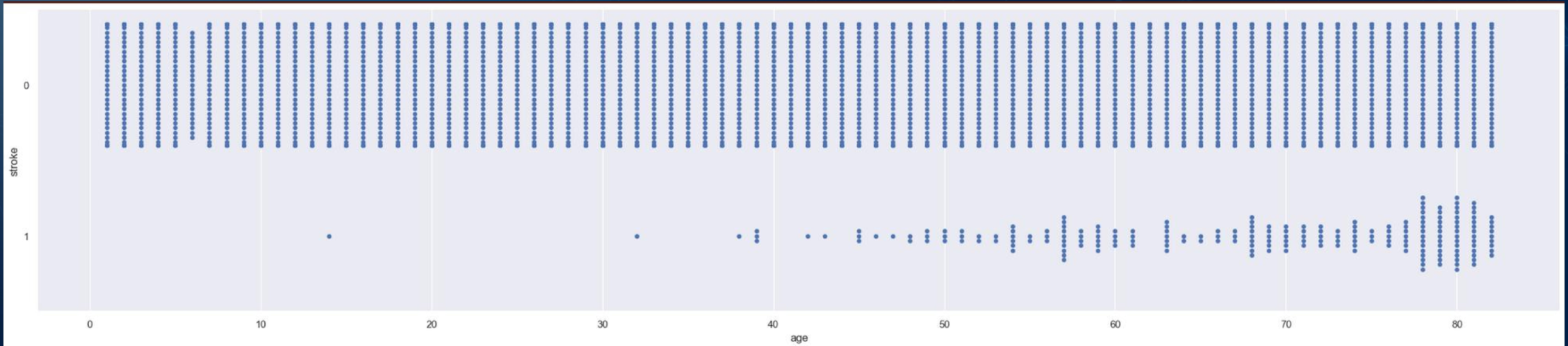
BMI Chart (Metric)

Weight (kg)

Height (cm)	40	45	50	55	60	65	70	75	80	85	90	95	100	105	110	115	120	125	130	135	140	145	150
145	19	21	24	26	29	31	33	36	38	40	43	45	48	50	52	55	57	59	62	64	67	69	71
147.5	18	21	23	25	28	30	32	34	37	39	41	44	46	48	51	53	55	57	60	62	64	67	69
150	18	20	22	24	27	29	31	33	36	38	40	42	44	47	49	51	53	56	58	60	62	64	67
152.5	17	19	21	24	26	28	30	32	34	37	39	41	43	45	47	49	52	54	56	58	60	62	64
155	17	19	21	23	25	27	29	31	33	35	37	40	42	44	46	48	50	52	54	56	58	60	62
157.5	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48	50	52	54	56	58	60
160	16	18	20	21	23	25	27	29	31	33	35	37	39	41	43	45	47	49	51	53	55	57	59
162.5	15	17	19	21	23	25	27	28	30	32	34	36	38	40	42	44	45	47	49	51	53	55	57
165	15	17	18	20	22	24	26	28	29	31	33	35	37	39	40	42	44	46	48	50	51	53	55
167.5	14	16	18	20	21	23	25	27	29	30	32	34	36	37	39	41	43	45	46	48	50	52	53
170	14	16	17	19	21	22	24	26	28	29	31	33	35	36	38	40	42	43	45	47	48	50	52
172.5	13	15	17	18	20	22	24	25	27	29	30	32	34	35	37	39	40	42	44	45	47	49	50
175	13	15	16	18	20	21	23	24	26	28	29	31	33	34	36	38	39	41	42	44	46	47	49
177.5	13	14	16	17	19	21	22	24	25	27	29	30	32	33	35	37	38	40	41	43	44	46	48
180	12	14	15	17	19	20	22	23	25	26	28	29	31	32	34	35	37	39	40	42	43	45	46
182.5	12	14	15	17	18	20	21	23	24	26	27	29	30	32	33	35	36	38	39	41	42	44	45
185	12	13	15	16	18	19	20	22	23	25	26	28	29	31	32	34	35	37	38	39	41	42	44
187.5	11	13	14	16	17	18	20	21	23	24	26	27	28	30	31	33	34	36	37	38	40	41	43
190	11	12	14	15	17	18	19	21	22	24	25	26	28	29	30	32	33	35	36	37	39	40	42
192.5	11	12	13	15	16	18	19	20	22	23	24	26	27	28	30	31	32	34	35	36	38	39	40
195	11	12	13	14	16	17	18	20	21	22	24	25	26	28	29	30	32	33	34	36	37	38	39

Underweight (<18.5) | Healthy (18.5 - 24.9) | Overweight (25 - 29.9) | Obese Class I (30 - 34.9) | Obese Class II (35 - 39.5) | Obese Class III (>40)

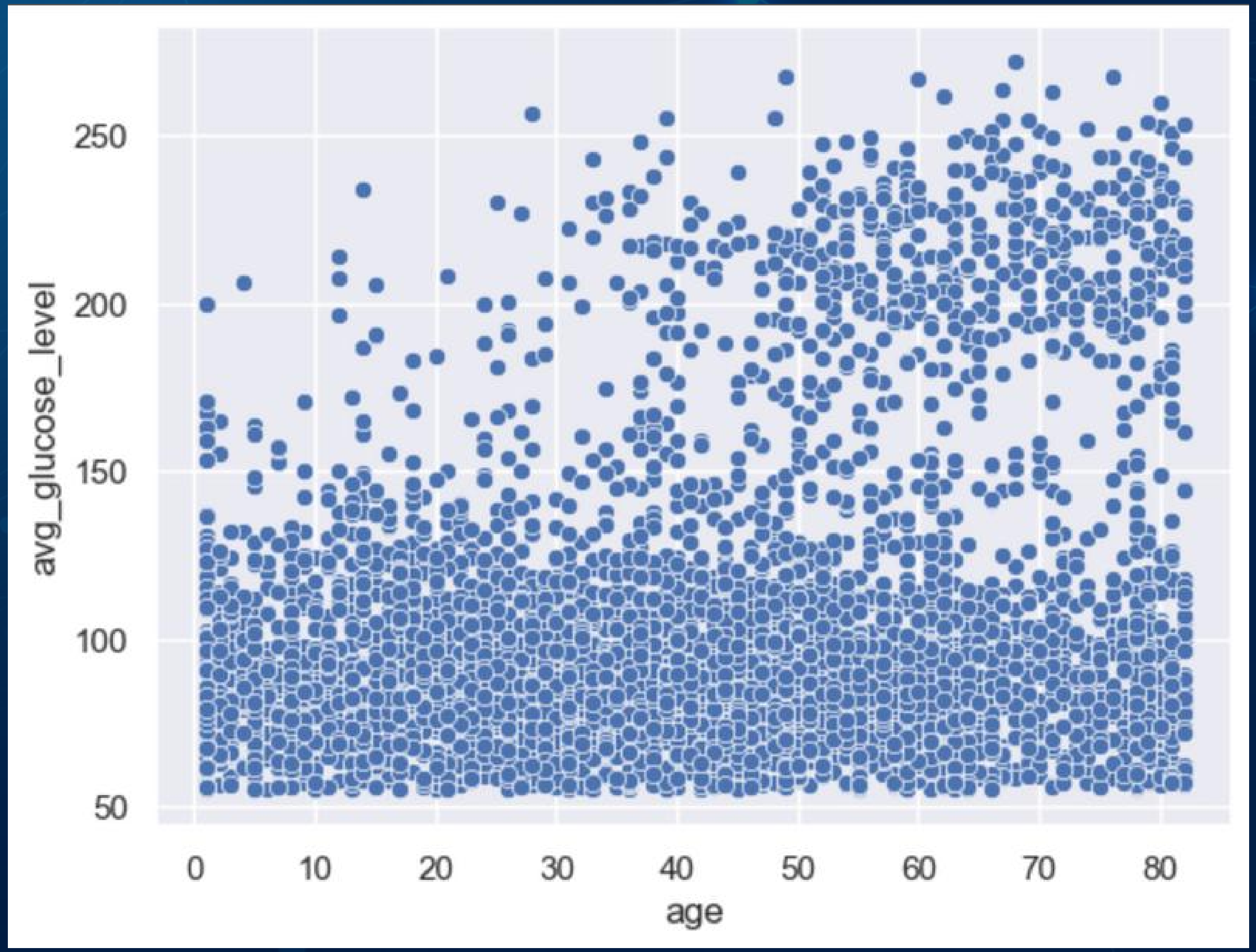
Exploratory Data Analysis



Age against Stroke

Higher the age, increased likelihood of getting stroke

Exploratory Data Analysis



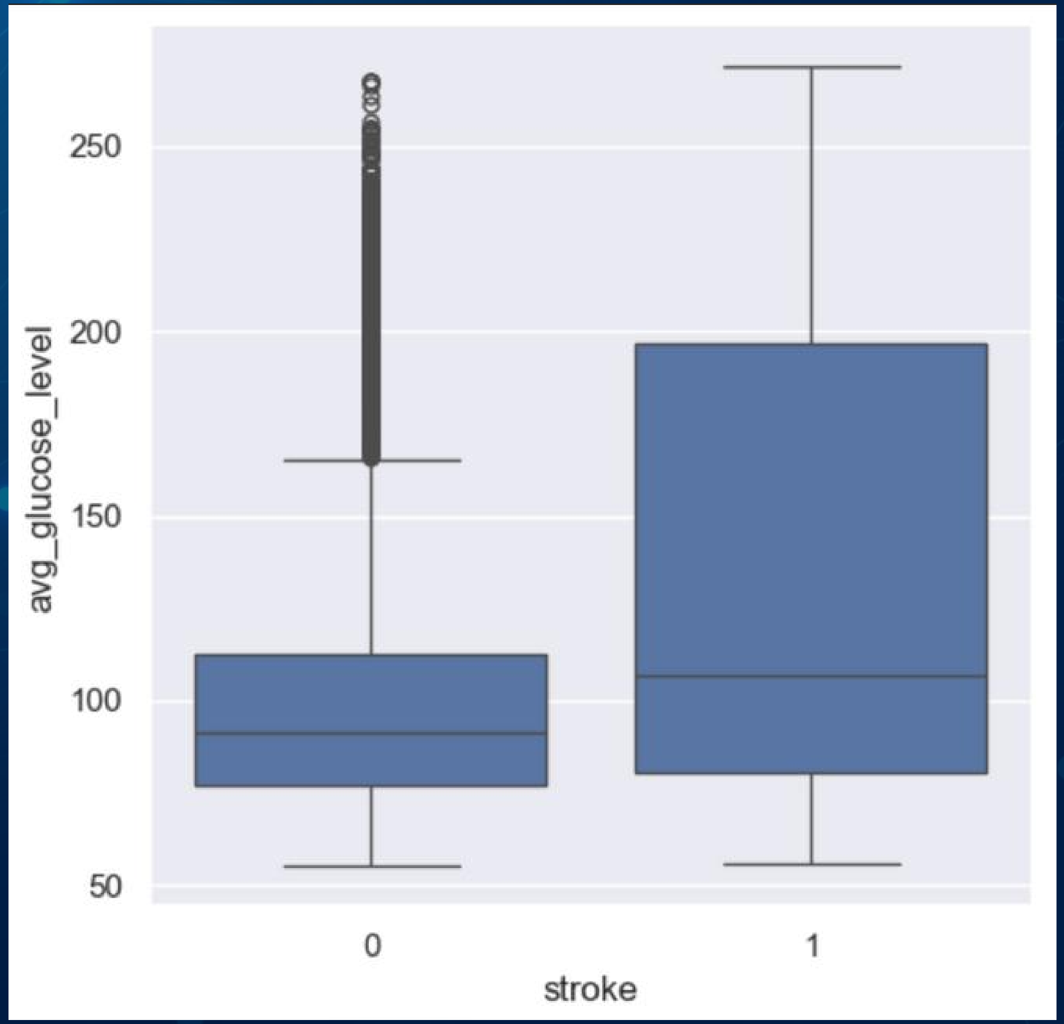
Age against Stroke

Higher the age, increased likelihood of getting stroke

Age against Glucose

Higher the age, increased average levels of glucose

Exploratory Data Analysis

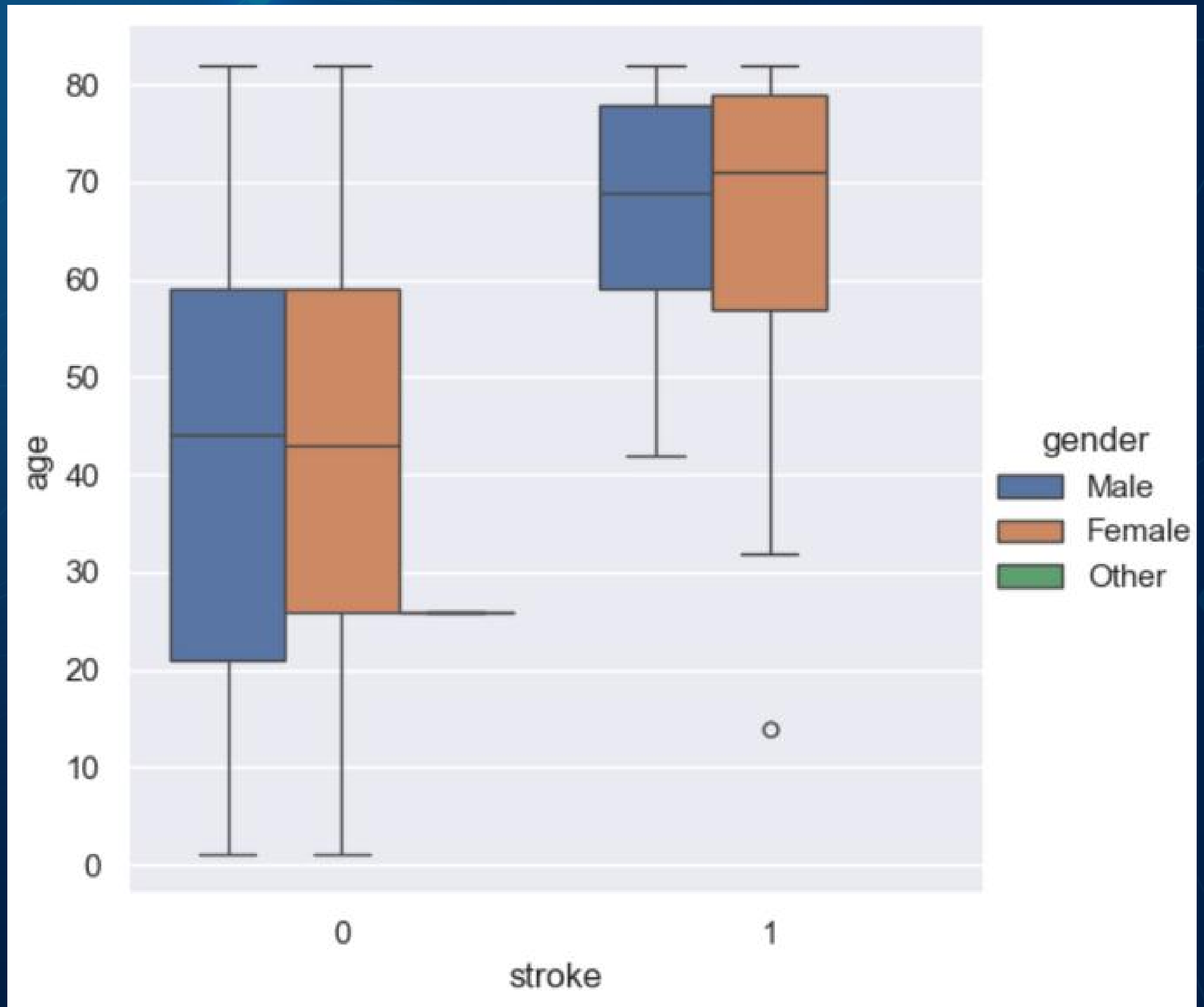


Age against Stroke
Higher the age, increased likelihood of getting stroke

Age against Glucose
Higher the age, increased average levels of glucose

Stroke against Glucose
Those with stroke has a overall higher median level of glucose level

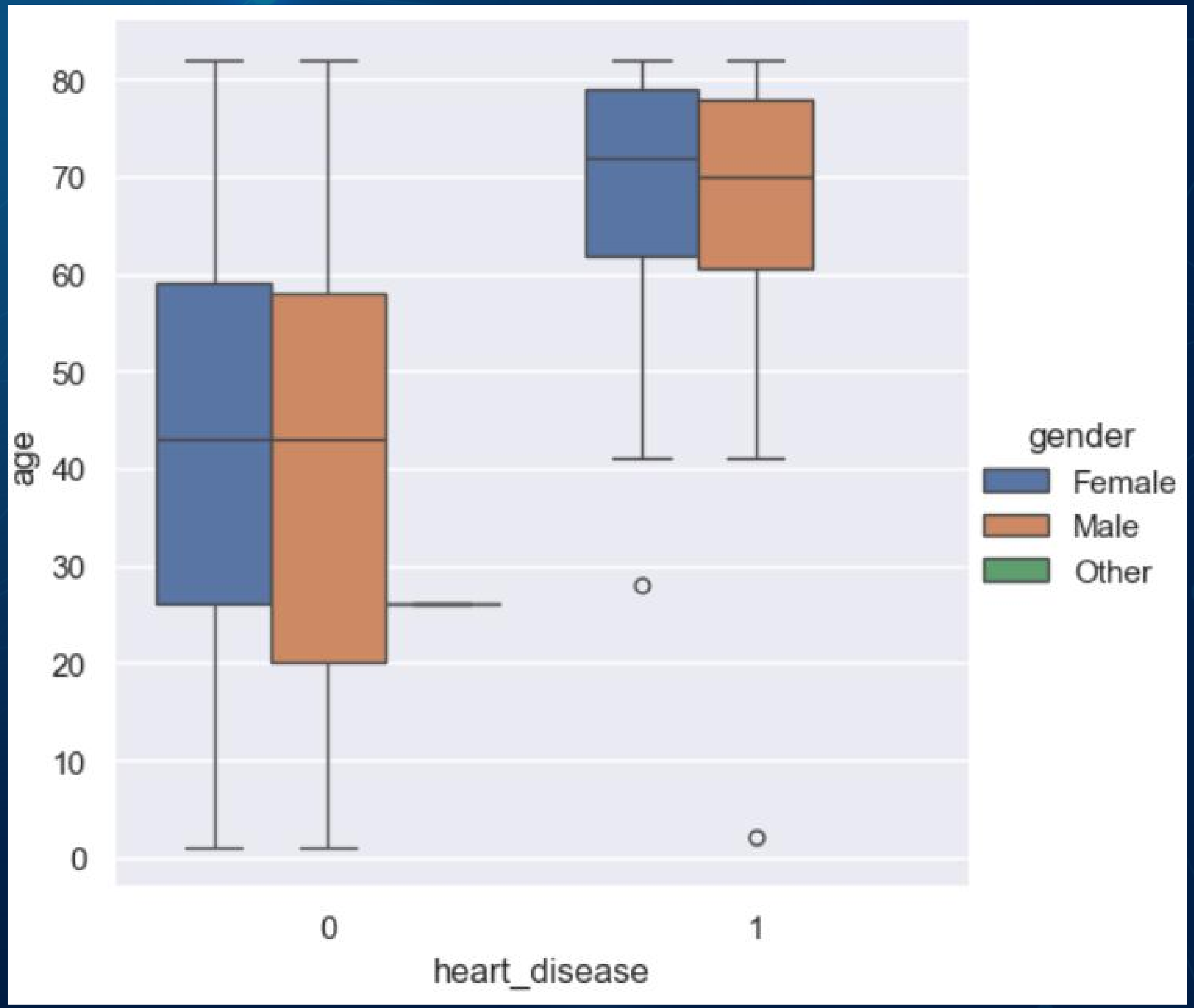
Exploratory Data Analysis



Stroke against Age (Categorized)

Higher the age, increased likelihood of getting stroke

Exploratory Data Analysis



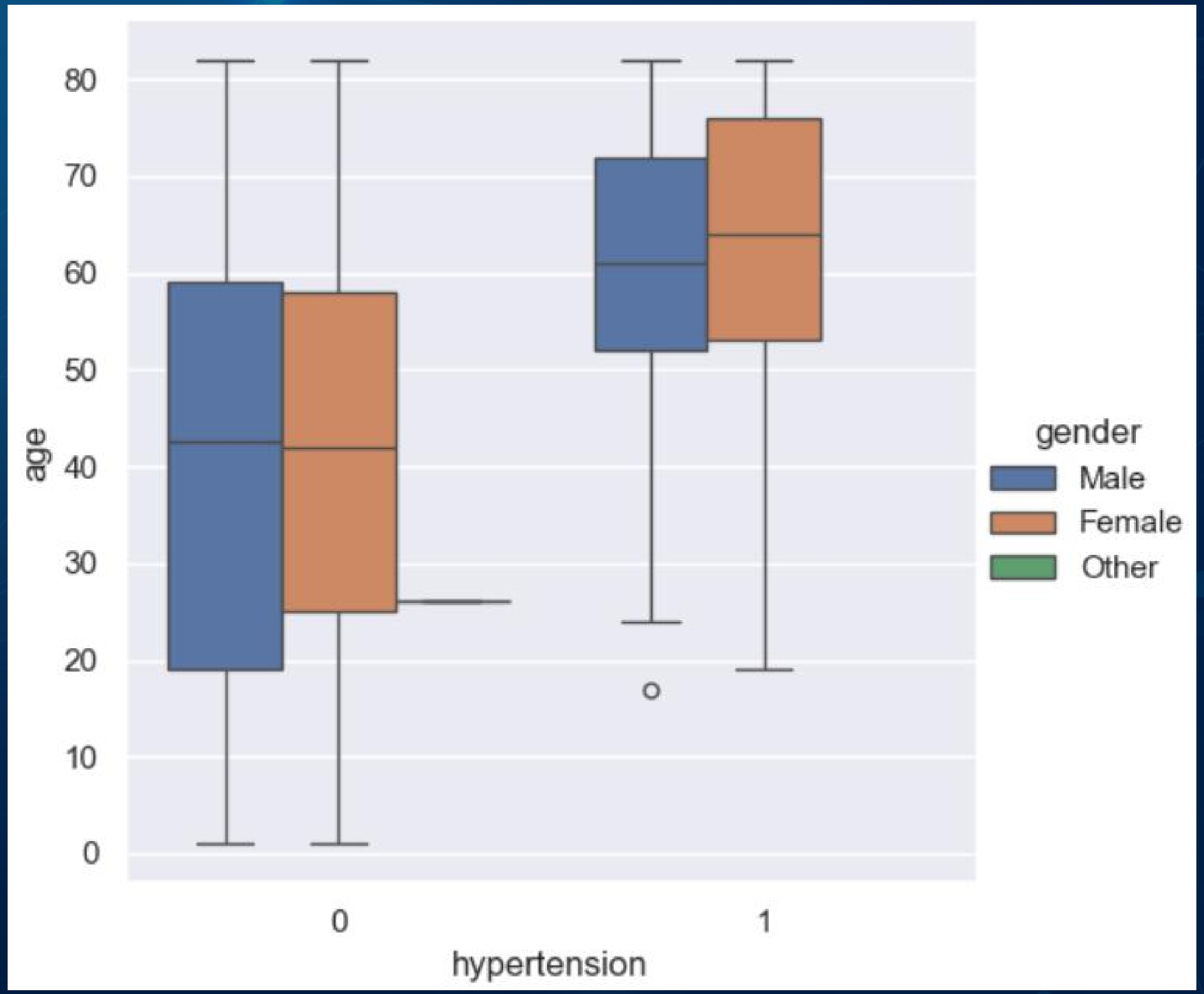
Stroke against Age (Categorized)

Higher the age, increased likelihood of getting stroke

Age against Heart Disease

Higher the age, increased average levels of glucose

Exploratory Data Analysis



Stroke against Age (Categorized)

Higher the age, increased likelihood of getting stroke

Age against Heart Disease

Higher the age, increased average levels of glucose

Age against Hypertension

Higher the age, increased risk of hypertension

EDA Conclusions

Age

Older ages tend to have a higher risk for Heart Disease, Hypertension, Average Glucose Level, and Stroke chances

Glucose

Having an increased level of Average Glucose Level does not directly translates to having higher chance of getting Stroke

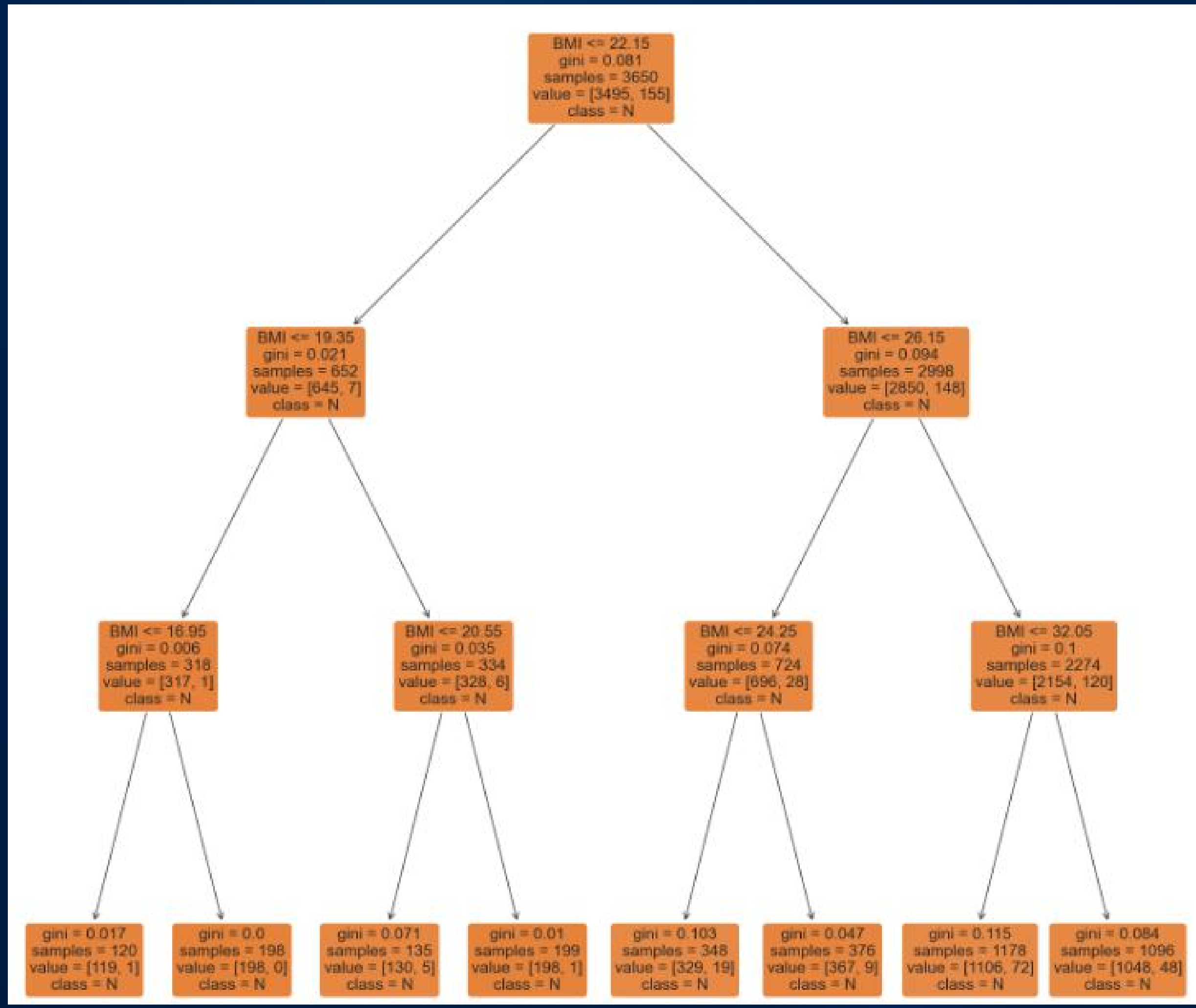
Gender

On average, older females have higher chances of getting hypertension and stroke, compared to their male counterparts. But females may develop stroke at an earlier age than males

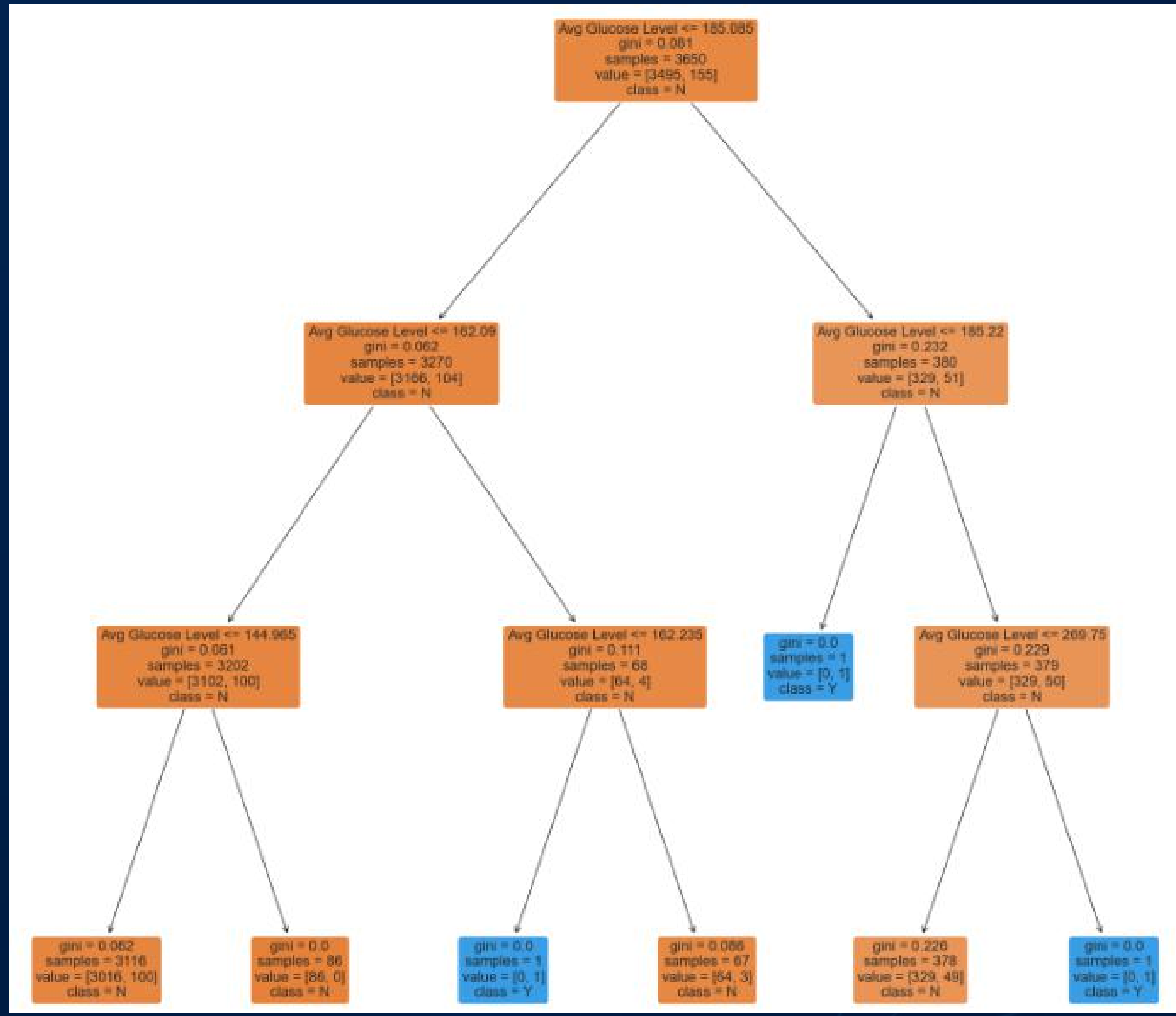
Machine Learning

Classification Tree

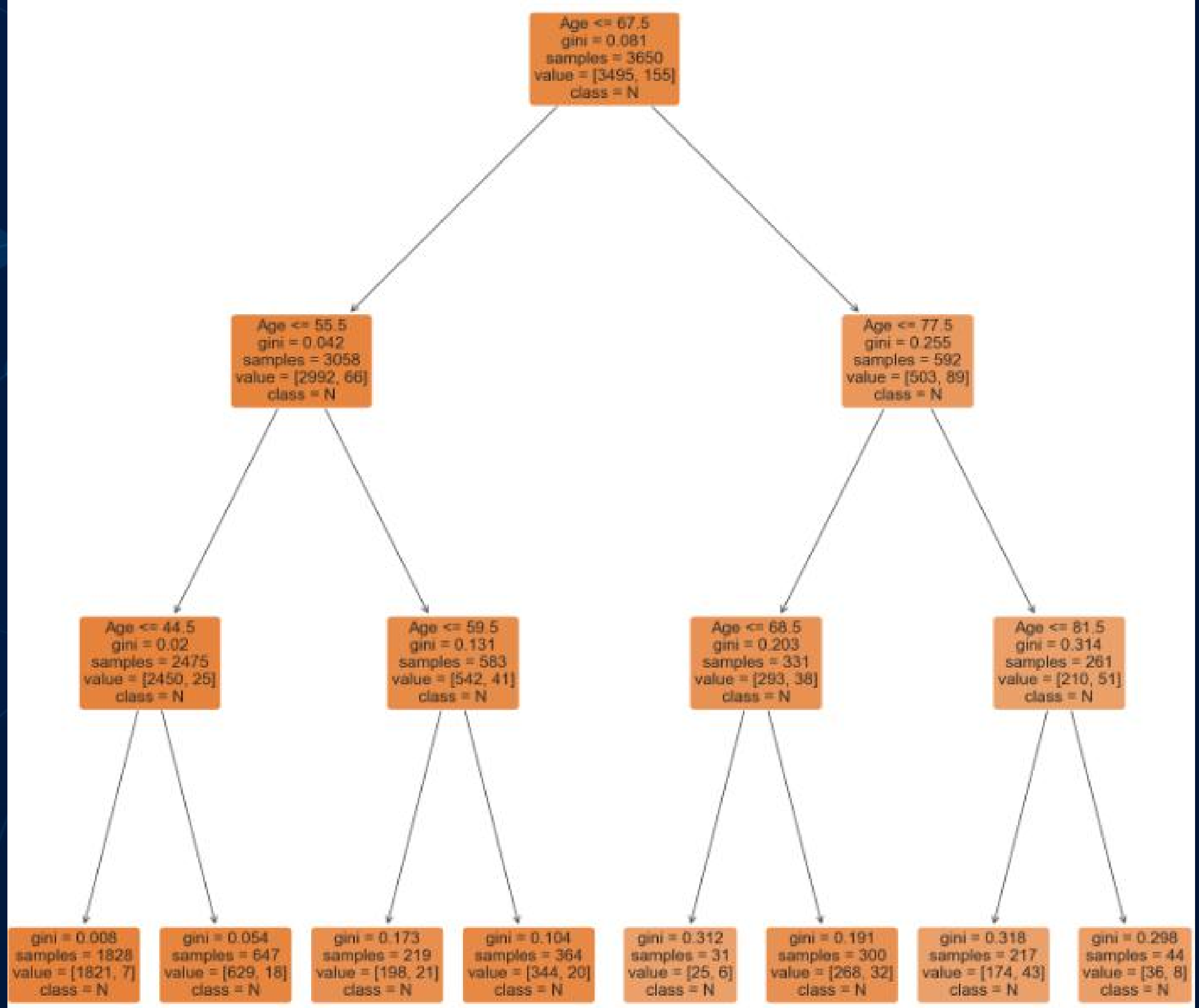
Stroke against



BMI

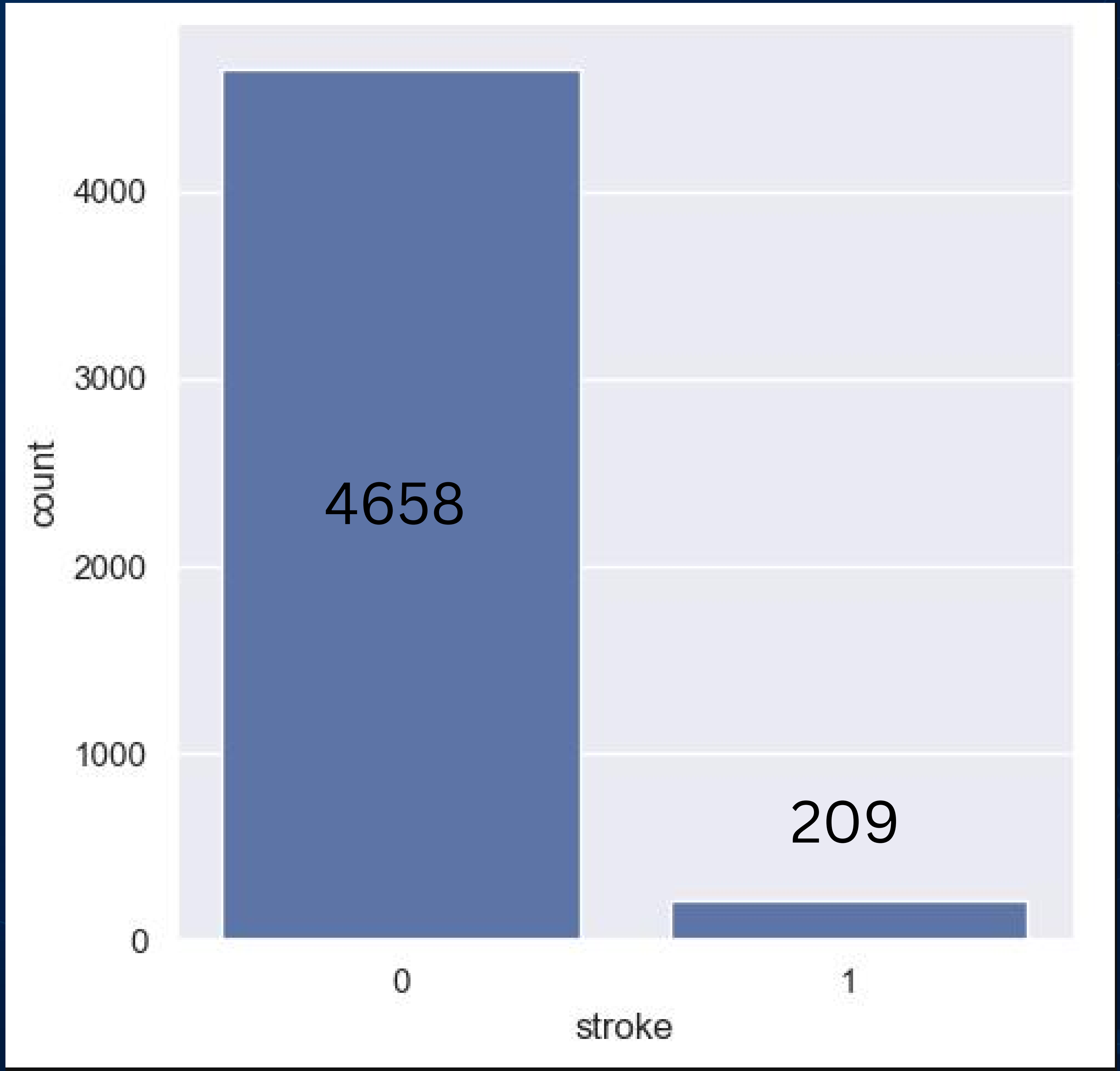


Avg_glucose_level



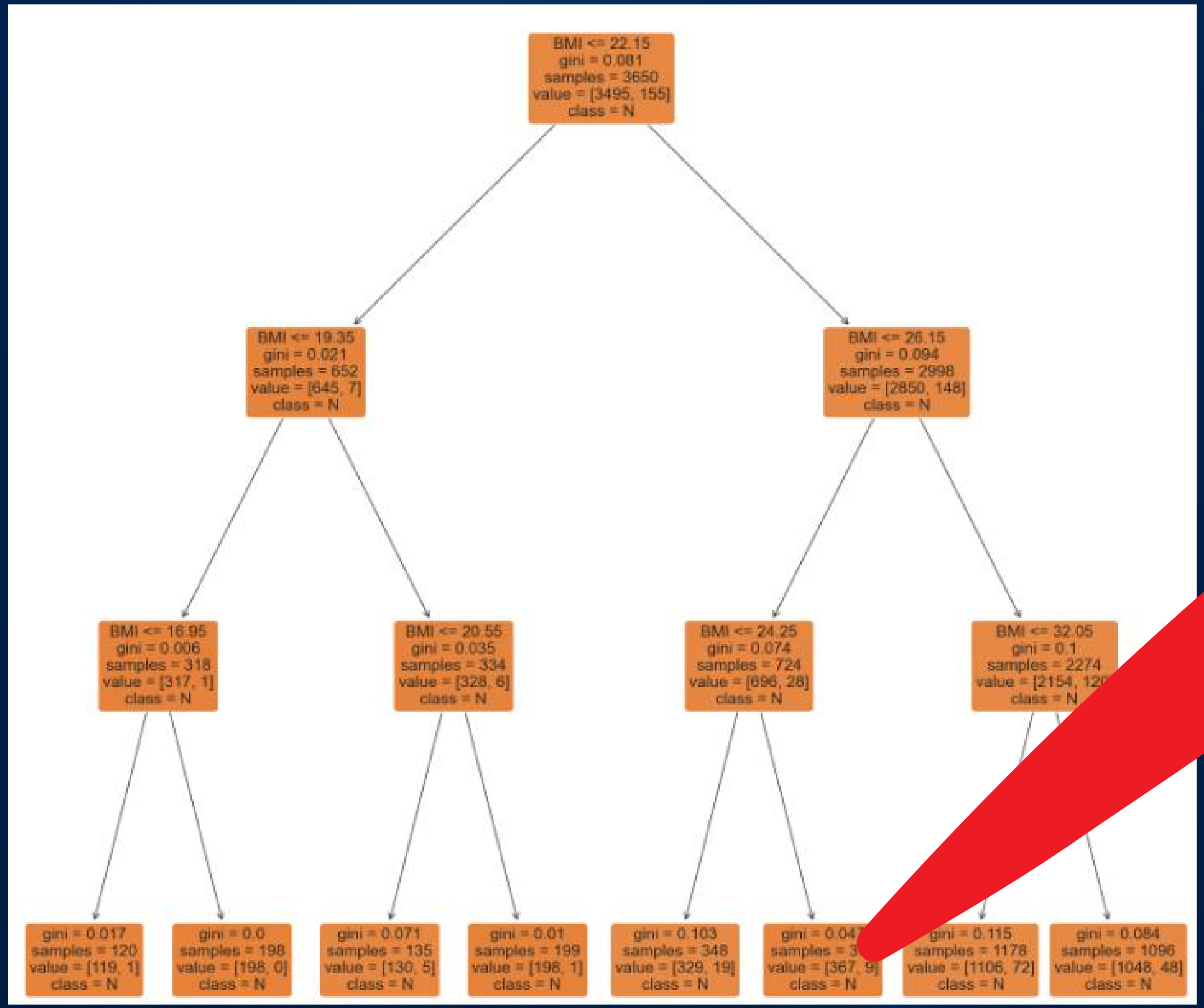
age

Classification Tree

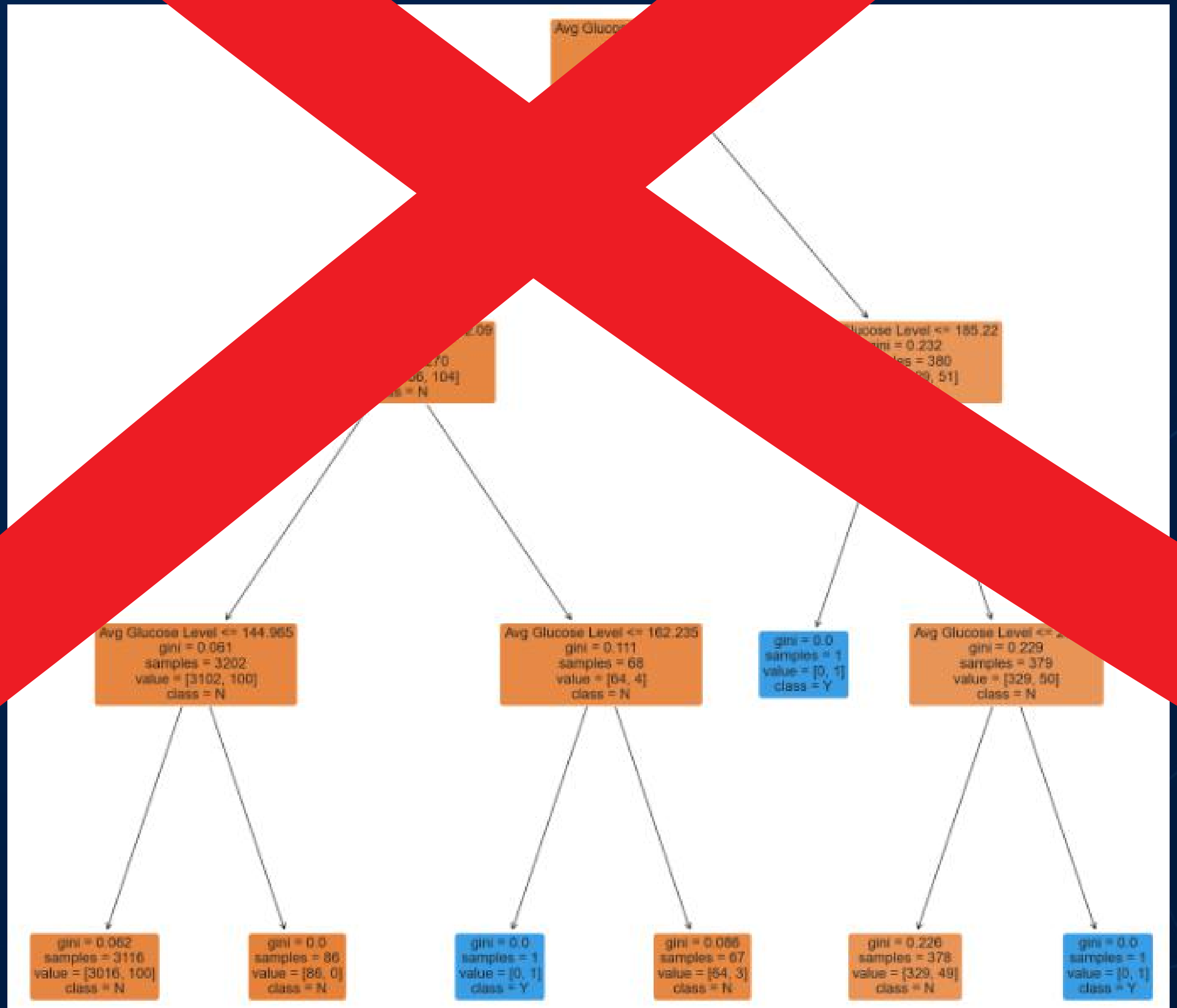


Classification Tree

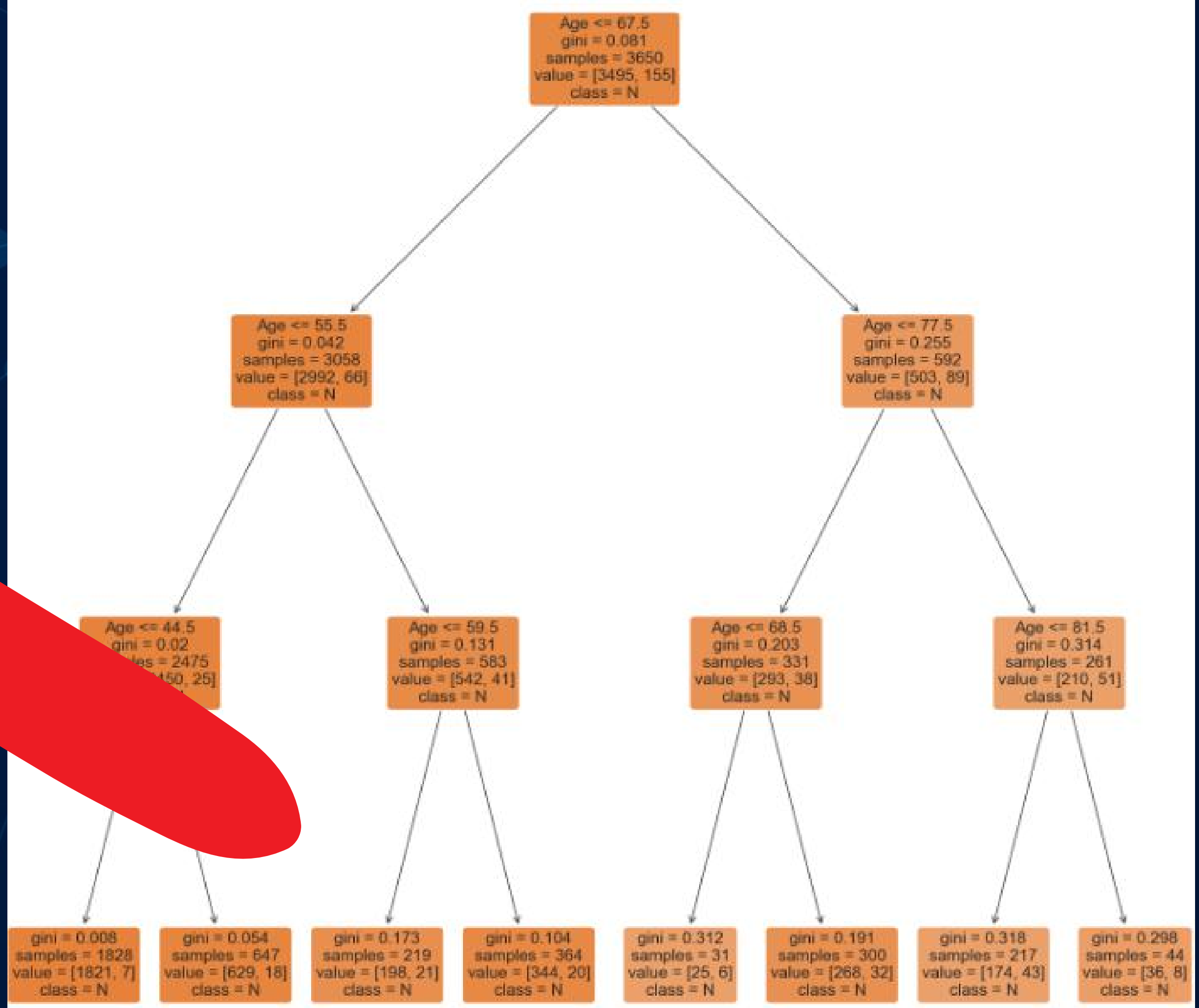
Stroke again



BMI



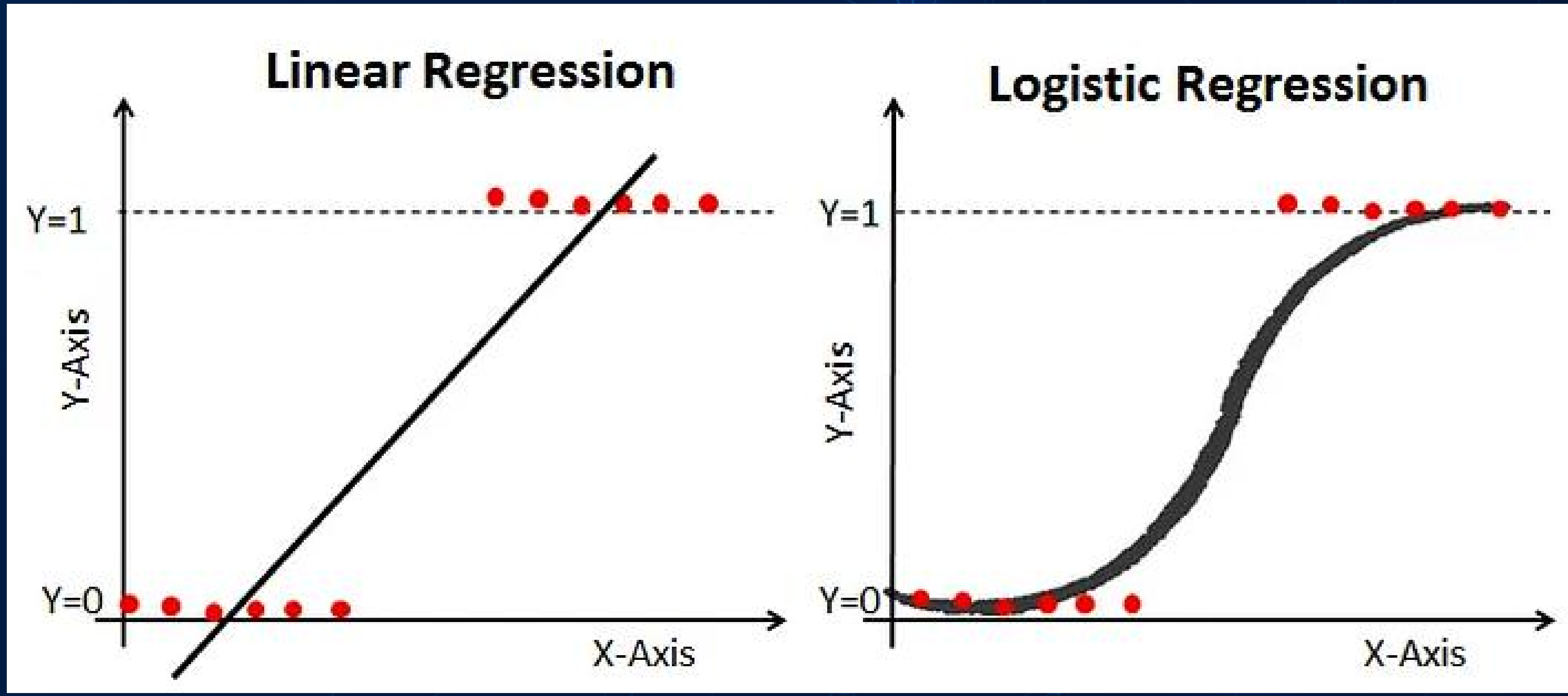
Avg_glucose_level



age

Logistic Regression

- Works best for binary classification tasks
- Utilizes multi variables for classification
- Analyze co-efficient of LR model to identify leading factors that are significant predictors of stroke risk



Resampling

- To handle and resolve the imbalance in the dataset
- Over-sample the lacking variable
- Under-sample the dominant variable

Logistic Regression without resampling

```
# Assuming X contains your predictor variables and y contains the target variable ('stroke')
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(cleancsvM_encoded, strokeCol, test_size=0.2, random_state=42)

# Create an instance of LogisticRegression
logistic_reg = LogisticRegression(random_state=42)

# Train the logistic regression model
logistic_reg.fit(X_train, y_train)

# Predict on the test set
y_pred = logistic_reg.predict(X_test)

# Calculate accuracy
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
print("")

# Generate a classification report
y_true = ['No Stroke' if label == 0 else 'Stroke' for label in y_test]
y_pred = ['No Stroke' if label == 0 else 'Stroke' for label in y_pred]

# Generate the classification report with custom class labels
print(classification_report(y_true, y_pred))
```

```
Accuracy: 0.944558521560575

              precision    recall  f1-score   support

   No Stroke       0.94      1.00      0.97       920
      Stroke       0.00      0.00      0.00        54

    accuracy                           0.94       974
   macro avg       0.47      0.50      0.49       974
weighted avg       0.89      0.94      0.92       974
```

Resampling

Accuracy: 0.944558521560575

	precision	recall	f1-score	support
No Stroke	0.94	1.00	0.97	920
Stroke	0.00	0.00	0.00	54
accuracy			0.94	974
macro avg	0.47	0.50	0.49	974
weighted avg	0.89	0.94	0.92	974

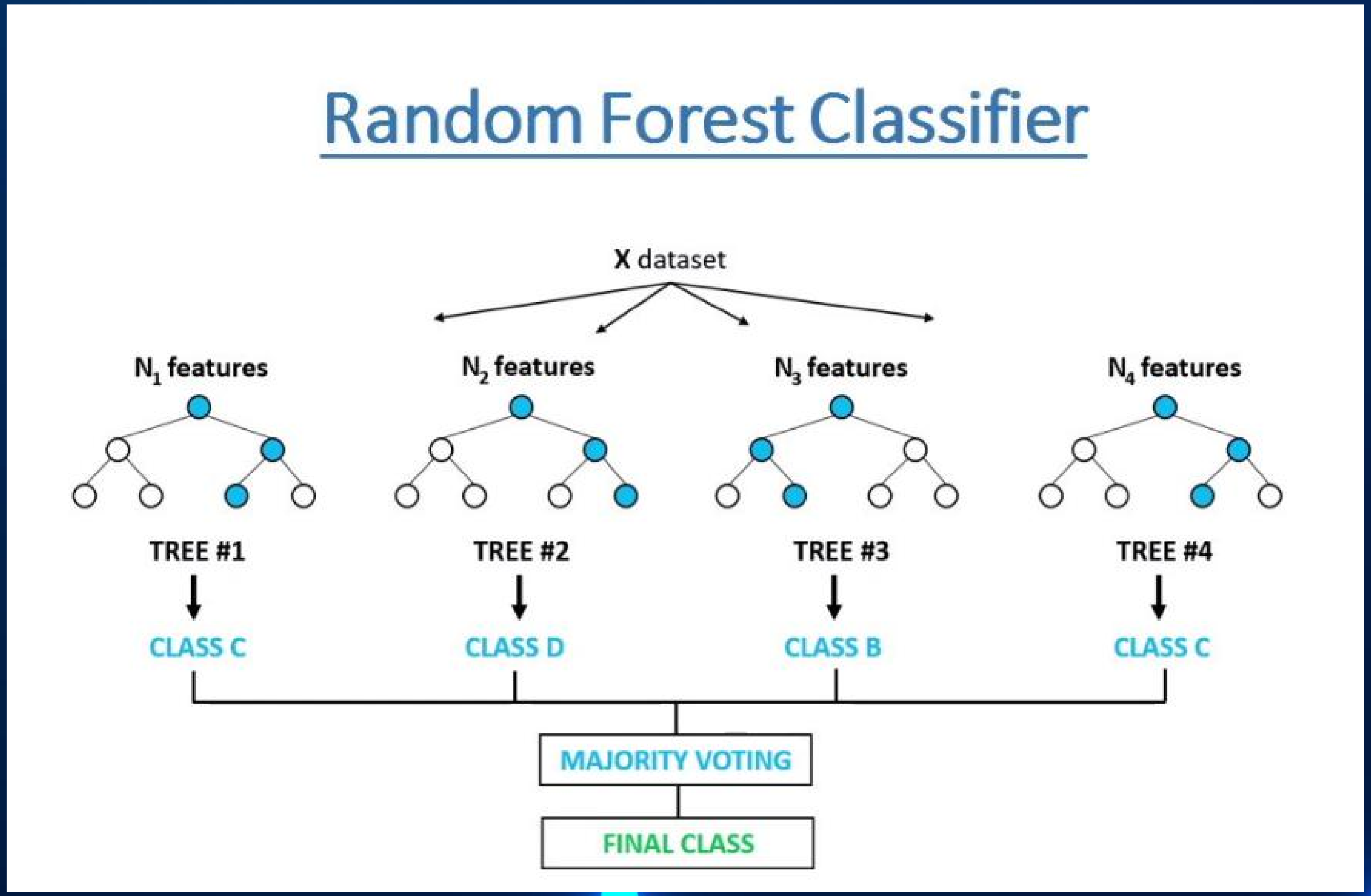
LR without Resampling

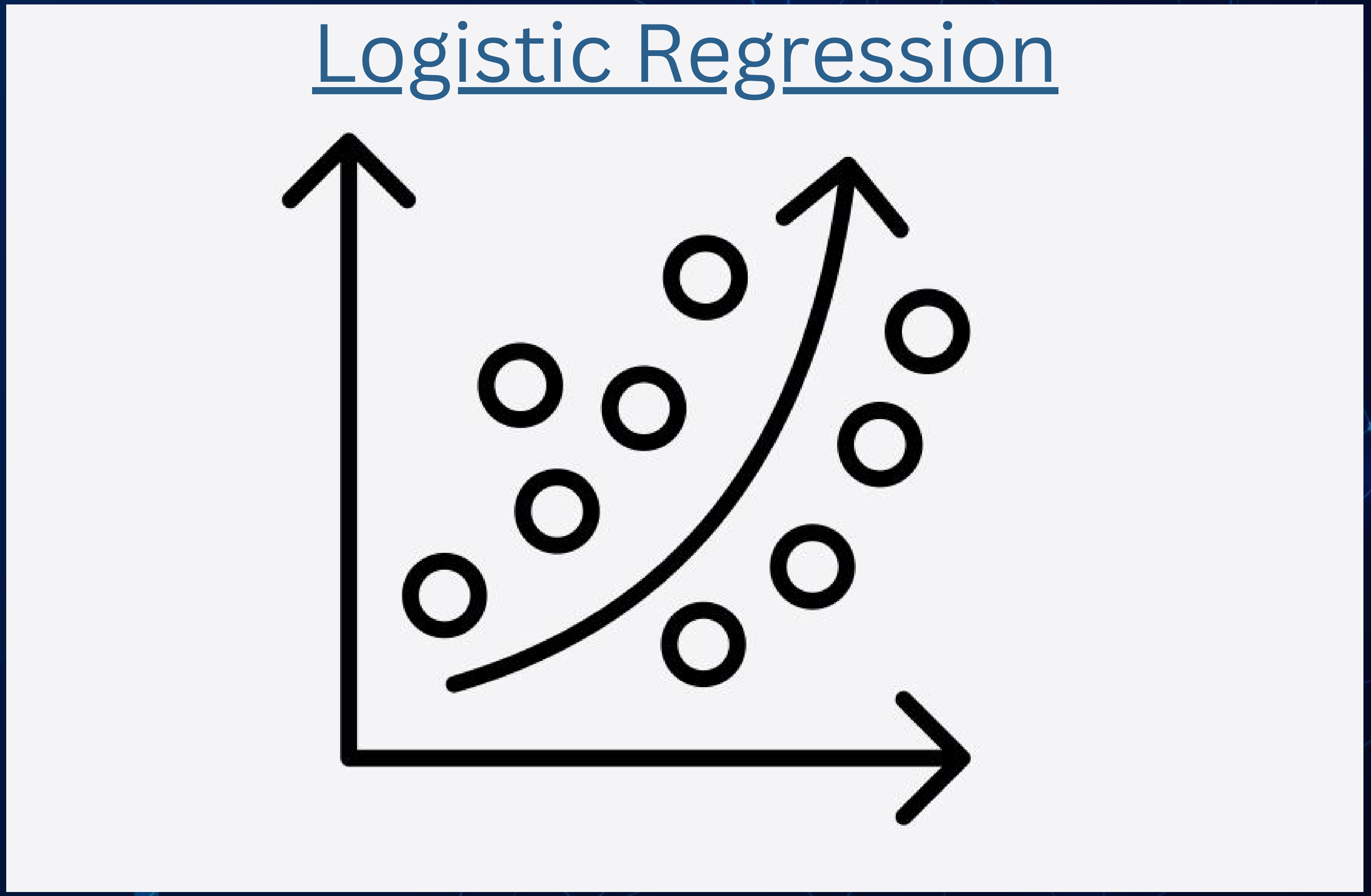
Accuracy: 0.9179184549356223

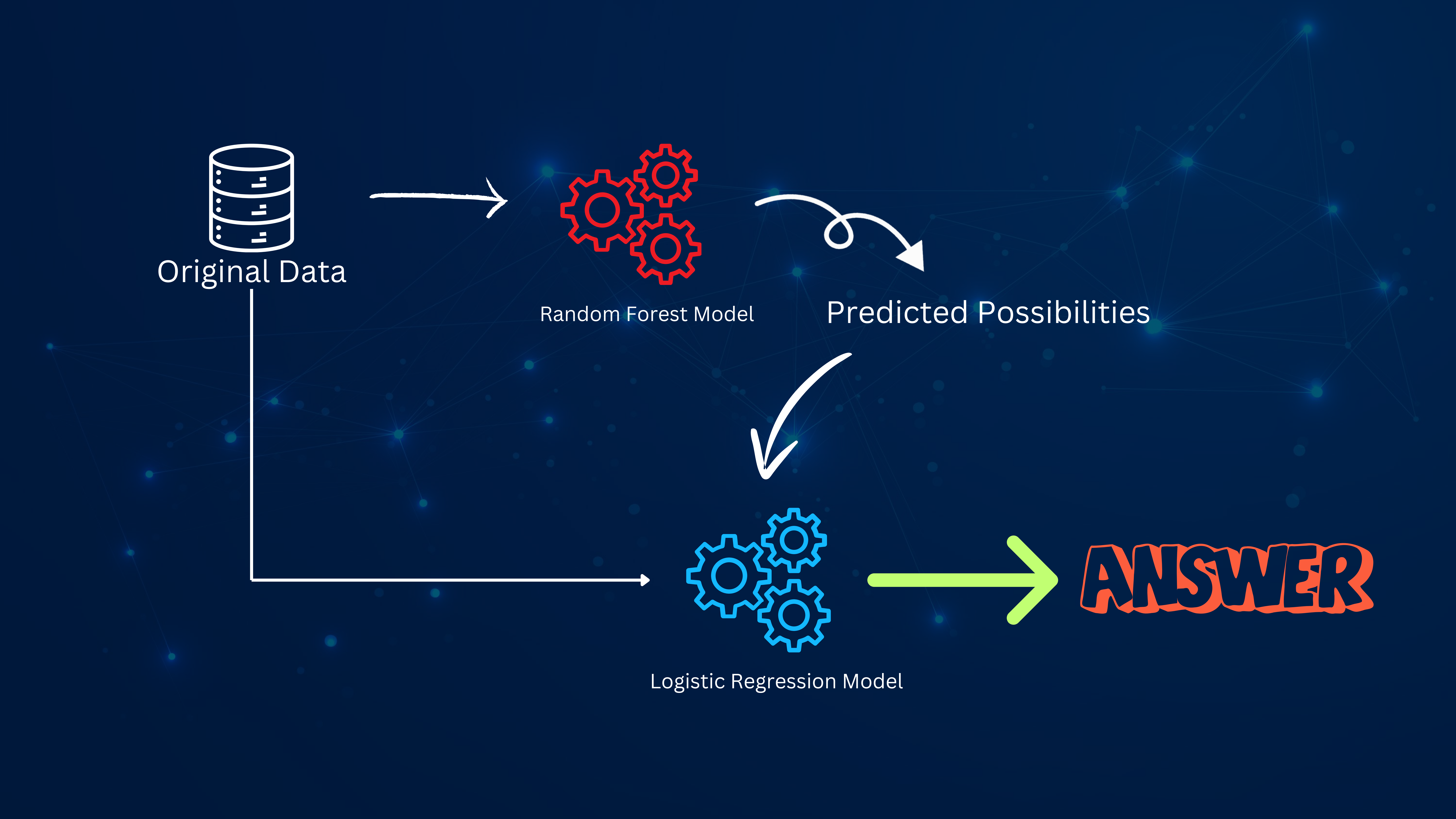
	precision	recall	f1-score	support
No Stroke	0.88	0.97	0.92	939
Stroke	0.96	0.87	0.91	925
accuracy			0.92	1864
macro avg	0.92	0.92	0.92	1864
weighted avg	0.92	0.92	0.92	1864

LR with Resampling

Random Forest + Logistic Regression





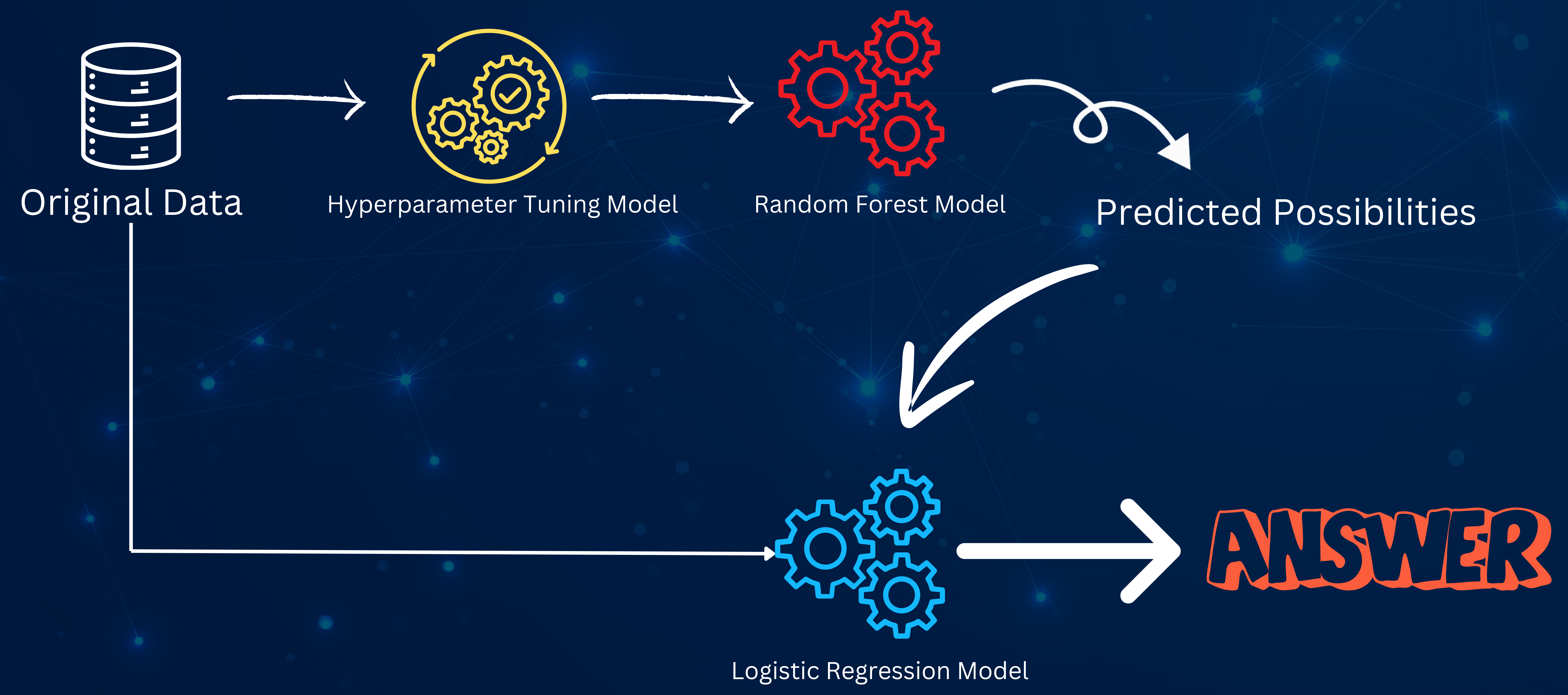

Original Data
Random Forest Model
Predicted Possibilities
Logistic Regression Model
ANSWER

Random Forest + Logistic Regression

```
Accuracy: 0.9774678111587983

Classification Report:
              precision    recall  f1-score   support

   No Stroke       0.96      0.99      0.98       939
      Stroke       0.99      0.96      0.98       925

    accuracy                           0.98      1864
   macro avg       0.98      0.98      0.98      1864
weighted avg       0.98      0.98      0.98      1864
```

Hyperparameter Tuning

- Optimize the performance of ML models
- Searches through the predefined grid of hyperparameters
- Offers the best combination that yields the best performance model


Original Data
Hyperparameter Tuning Model
Random Forest Model
Predicted Possibilities
Logistic Regression Model
ANSWER

Hyperparameter Tuning

```
# Define the parameter grid to search
param_grid = {
    'n_estimators': [100, 200, 300],  # Number of trees in the forest
    'max_depth': [None, 10, 20],       # Maximum depth of the trees
    'min_samples_split': [2, 5, 10],   # Minimum number of samples required to split an internal node
    'min_samples_leaf': [1, 2, 4]      # Minimum number of samples required to be a leaf node
}

# Create a GridSearchCV object
grid_search = GridSearchCV(RandomForestClassifier(random_state=42), param_grid, cv=5, scoring='accuracy')

# Perform grid search cross-validation
grid_search.fit(X_rf_train, y_rf_train)

# Get the best parameters and best score
best_params = grid_search.best_params_
best_score = grid_search.best_score_

print("Best Parameters:", best_params)
print("Best Score (Accuracy):", best_score)
```

```
Best Parameters: {'max_depth': None, 'min_samples_leaf': 1, 'min_samples_split': 5, 'n_estimators': 100}
Best Score (Accuracy): 0.972758879901332
```

Summary

Summary

Outcome

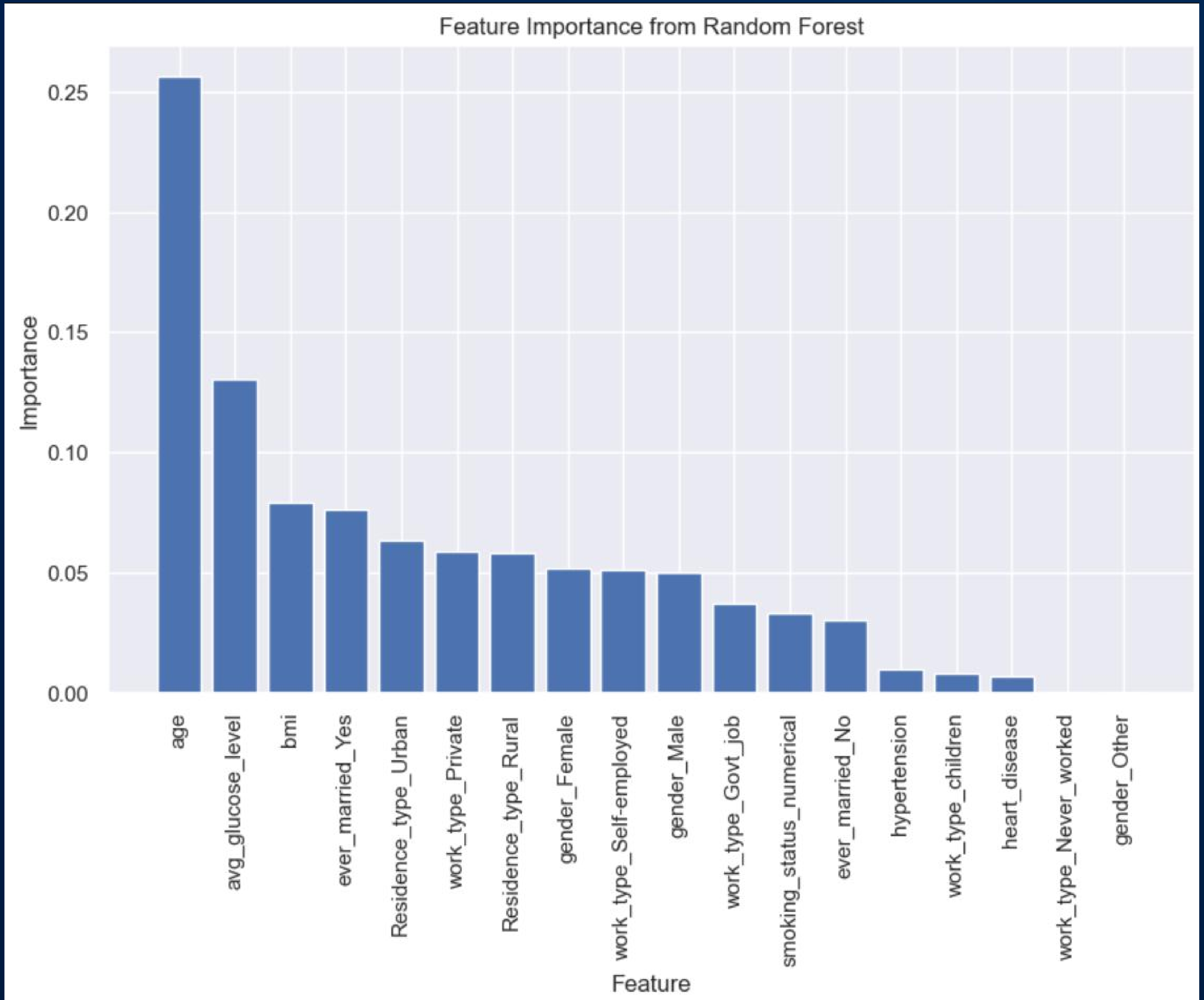
Successfully built a model capable of identifying individuals at risk of stroke

Addresses the Problem

Identifying the contributing factors in contributing to a stroke occurring

Contributing Factors

1. Age
2. Avg_Glucose_Level
3. BMI

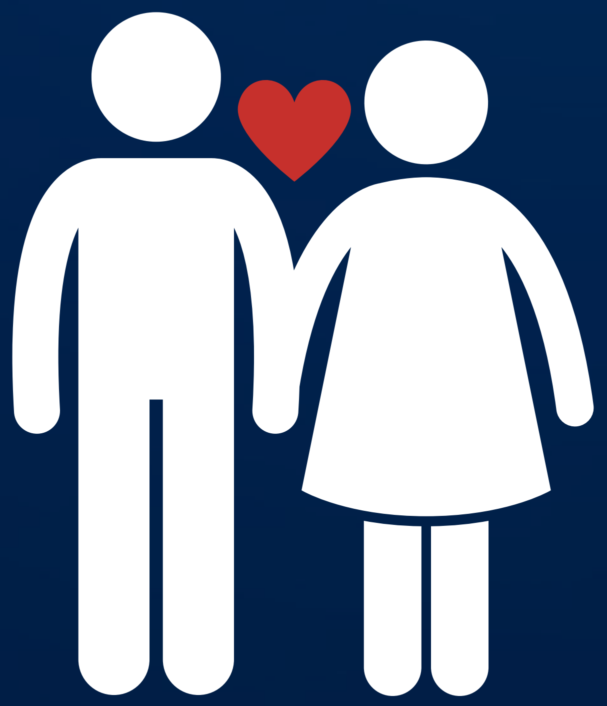


Interesting Facts

```
Significant Factors in Predicting Stroke:
work_type_Govt_job: 1.1160
Residence_type_Rural: 0.9679
gender_Female: 0.7128
work_type_Self-employed: 0.6454
Residence_type_Urban: 0.6374
gender_Male: 0.6015
hypertension: 0.4593
heart_disease: 0.4494
work_type_children: -0.3705
smoking_status_numerical: 0.2258
work_type_Private: 0.1789
ever_married_Yes: 0.1175
ever_married_No: 0.1170
work_type_Never_worked: -0.0181
bmi: 0.0176
avg_glucose_level: -0.0071
age: 0.0020
gender_Other: -0.0016
```

Gender

Marital Status



Takeaways

- Model for Prediction: The best model prediction isn’t always purely based on 1 algorithm, but a combination of a few
- Importance of feature engineering: Doing more with the given set of data
- Addressing Class Imbalance: Handling of Class Imbalance data is crucial in building a model that is confident in predictions

Our Takeaways

Future Work

- Collaboration with healthcare professionals
- Integrate predictive model into clinical practice
- Assist and facilitate early identification of individuals with risk of stroke

Future Work

Thank You

References:
https://www.world-stroke.org/assets/downloads/WSO_Global_Stroke_Fact_Sheet.pdf
https://www.who.int/srilanka/news/detail/29-10-2022-world-stroke-day-2022
https://supremevascular.com/stroke-and-stroke-screening/the-prevalence-of-stroke-in-singapore/#:~:text=Every%20year%2C%20an%20estimated%2015,and%20ischaemic%20strokes%20in%202021.
https://medium.com/analytics-vidhya/logistic-regression-using-python-a5044843a504

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