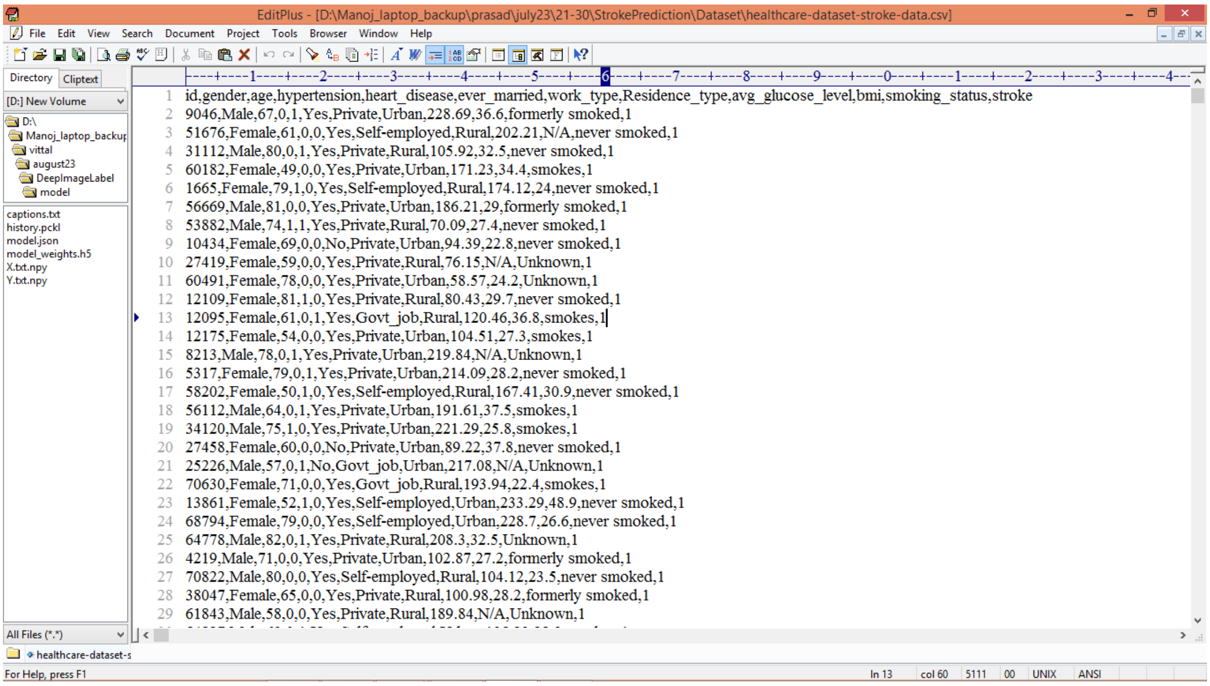
# Automated Stroke Prediction Using Machine Learning: An Explainable and Exploratory Study With a Web Application for Early Intervention

Stroke often causes due to blood flow stop to brain and this is one of the deadly diseases. Patient life can be saved and stroke can be avoided by timely and accurate detection. Existing detection technique requires heavy resources and they make time for prediction. To overcome from this problem many machine learning algorithms were introduced as they are very accurate in medical diseases prediction but existing techniques were suffering from data leakage such as improper handling or missing values, improper categorical data calculation etc. No existing techniques were employing any Explainable model (XAI) which can show which features are helping most in detecting stroke so doctor can give priority on such features for faster recovery. These explainable features can be Smoking, Age, BMI and may be other features.

So author of this paper employing different processing techniques such as Removing missing values, Imbalance data handling using SMOTE and relevant features selection using CHI2 algorithm. All this processed features will get trained on 6 different algorithms such as Random Forest, KNN, SVM, Logistic Regression, XGBOOST and Naive Bayes. In all algorithm Random Forest is giving high accuracy and each algorithm performance is evaluated in terms of accuracy, precision, recall and FSCORE.

For easy understanding of features author employing various graph on Strokes patient data. Best algorithm will be input to SHAPELY Explainable (XAI) algorithm to explain about features which are contributing most in predicting correct label.

To train all algorithms we are using STROKE dataset from KAGGLE and below screen showing all dataset details


EditPlus - [D:\Manoj_laptop_backup\prasad\july23\21-30\StrokePrediction\Dataset\healthcare-dataset-stroke-data.csv]
File Edit View Search Document Project Tools Browser Window Help
Directory Cliptext
[D:] New Volume
D:\
Manoj_laptop_backup
vittal
august23
DeepImageLabel
model
captions.txt
history.pckl
model.json
model_weights.h5
X.txt.npy
Y.txt.npy
All Files (*.*)
healthcare-dataset-s
For Help, press F1
In 13 col 60 5111 00 UNIX ANSI

```
id,gender,age,hypertension,heart_disease,ever_married,work_type,Residence_type,avg_glucose_level,bmi,smoking_status,stroke
9046,Male,67,0,1,Yes,Private,Urban,228.69,36.6,formerly smoked,1
51676,Female,61,0,0,Yes,Self-employed,Rural,202.21,N/A,never smoked,1
31112,Male,80,0,1,Yes,Private,Rural,105.92,32.5,never smoked,1
60182,Female,49,0,0,Yes,Private,Urban,171.23,34.4,smokes,1
1665,Female,79,1,0,Yes,Self-employed,Rural,174.12,24,never smoked,1
56669,Male,81,0,0,Yes,Private,Urban,186.21,29,formerly smoked,1
53882,Male,74,1,1,Yes,Private,Rural,70.09,27.4,never smoked,1
10434,Female,69,0,0,No,Private,Urban,94.39,22.8,never smoked,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 smoked,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 smoked,1
58202,Female,50,1,0,Yes,Self-employed,Rural,167.41,30.9,never smoked,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 smoked,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
13861,Female,52,1,0,Yes,Self-employed,Urban,233.29,48.9,never smoked,1
68794,Female,79,0,0,Yes,Self-employed,Urban,228.7,26.6,never smoked,1
64778,Male,82,0,1,Yes,Private,Rural,208.3,32.5,Unknown,1
4219,Male,71,0,0,Yes,Private,Urban,102.87,27.2,formerly smoked,1
70822,Male,80,0,0,Yes,Self-employed,Rural,104.12,23.5,never smoked,1
38047,Female,65,0,0,Yes,Private,Rural,100.98,28.2,formerly smoked,1
61843,Male,58,0,0,Yes,Private,Rural,189.84,N/A,Unknown,1
```

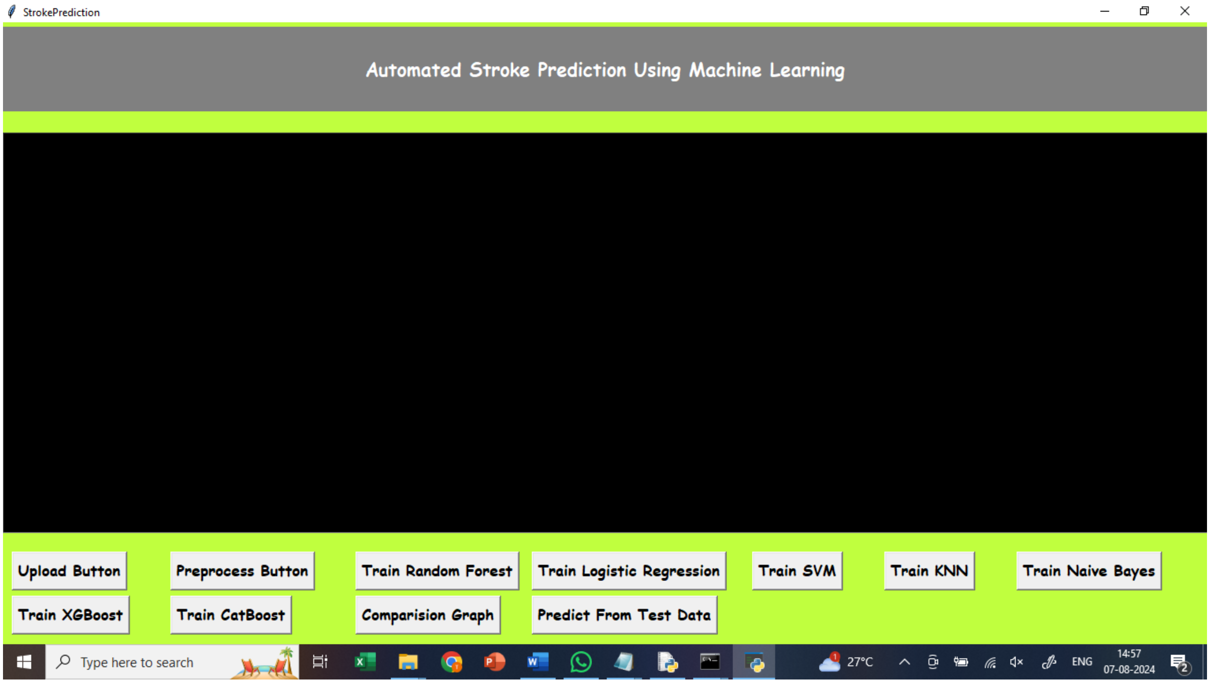
In above screen first row represents dataset column names and remaining rows represents dataset values and by using above dataset we will test all algorithm performance.

## Extension Concept

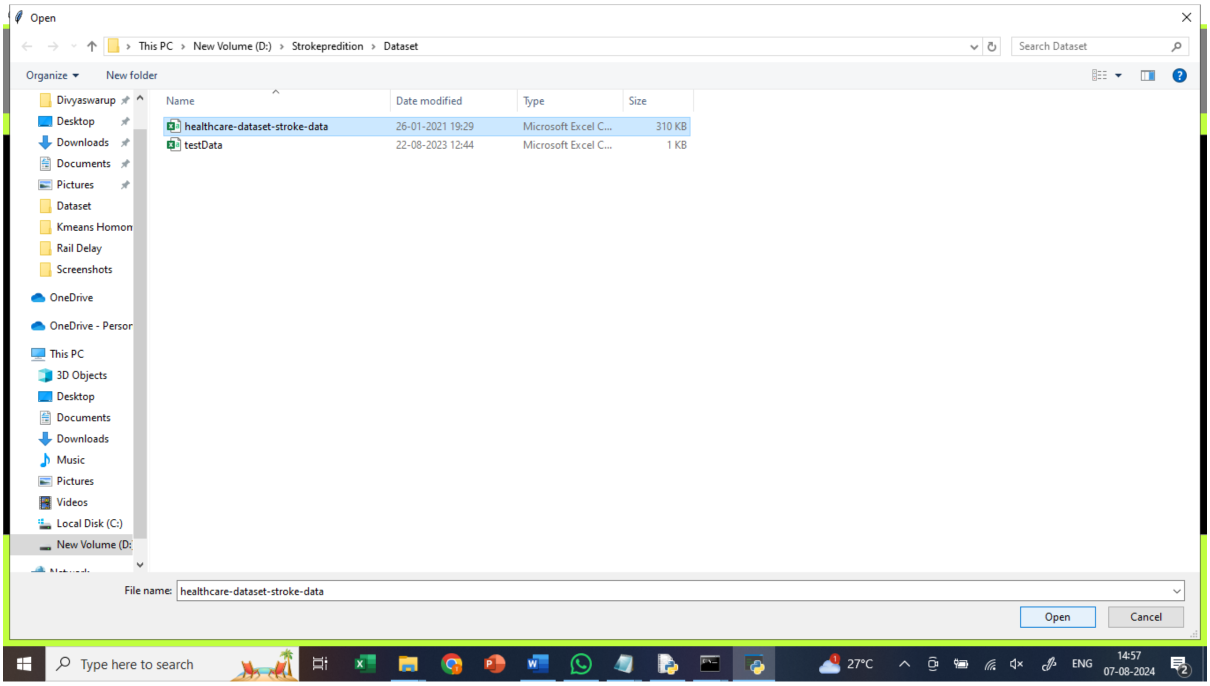
As extension we are employing CATBOOST classifier which will use forest of weak classifiers or group of multiple classifiers and then train each classifier and vote out best classifier for final prediction and using multiple classifier will help CATBOOST in enhancing prediction accuracy.

## SCREEN SHOTS

To run project double click on ‘run.bat’ file to get below screen



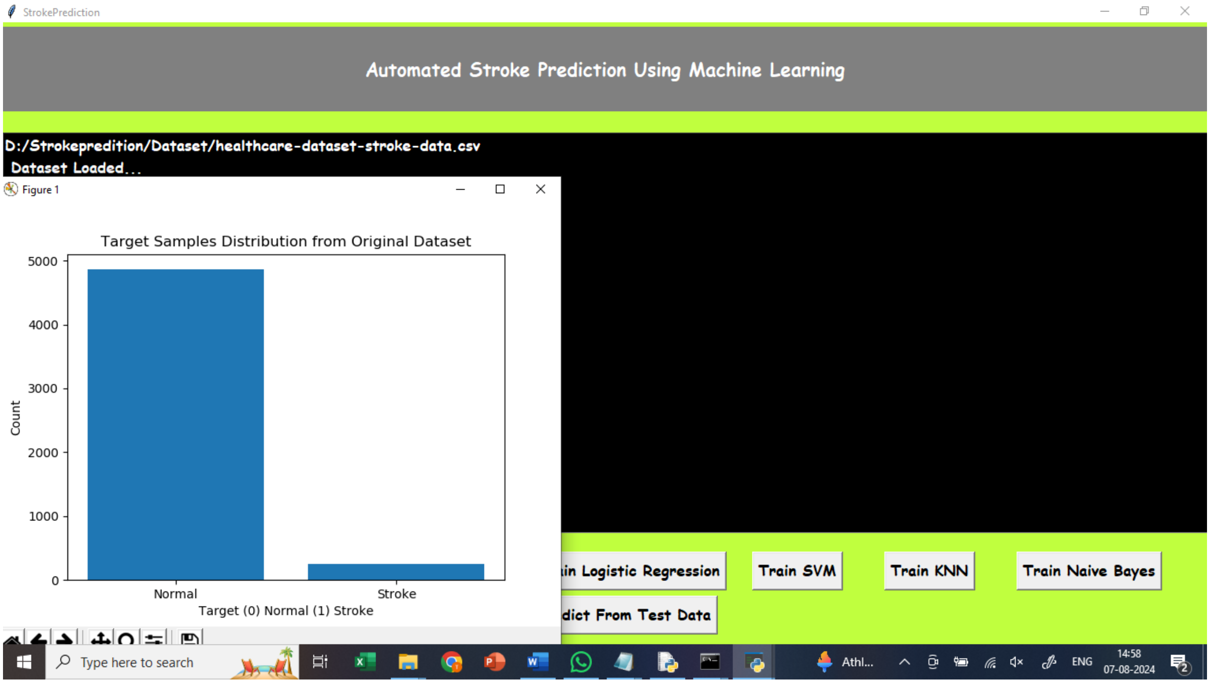
In above screen, click on upload button



| Name | Date modified | Type | Size |
|---|---|---|---|
| healthcare-dataset-stroke-data | 26-01-2021 19:29 | Microsoft Excel C... | 310 KB |
| testData | 22-08-2023 12:44 | Microsoft Excel C... | 1 KB |

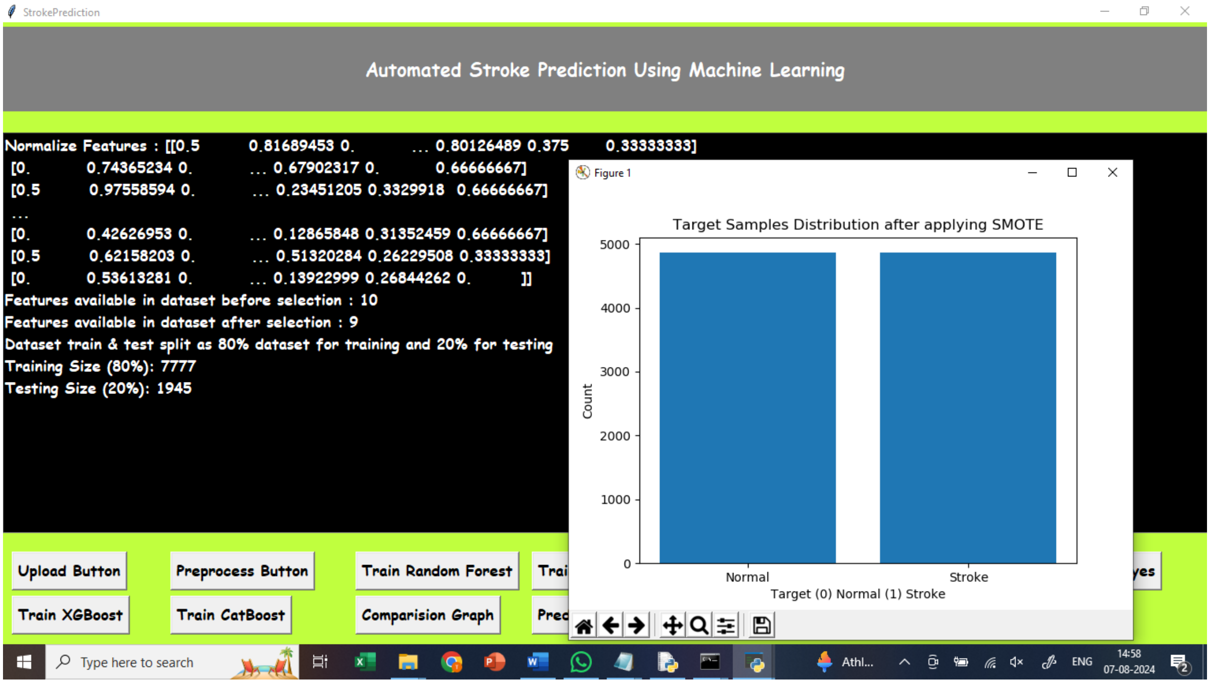
File name: healthcare-dataset-stroke-data

In above screen, upload healthcare dataset



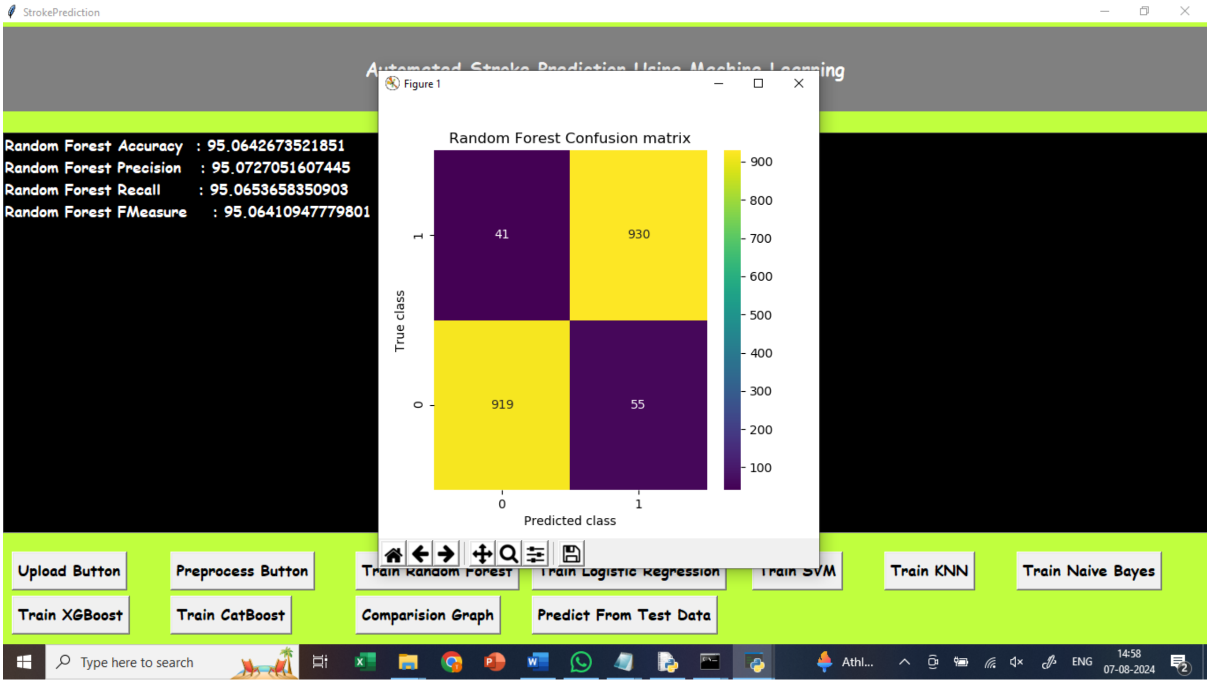
```
D:/Strokepredition/Dataset/healthcare-dataset-stroke-data.csv
 Dataset Loaded...
```

In above screen, we are getting the graph which displays the data related to normal and stroke.



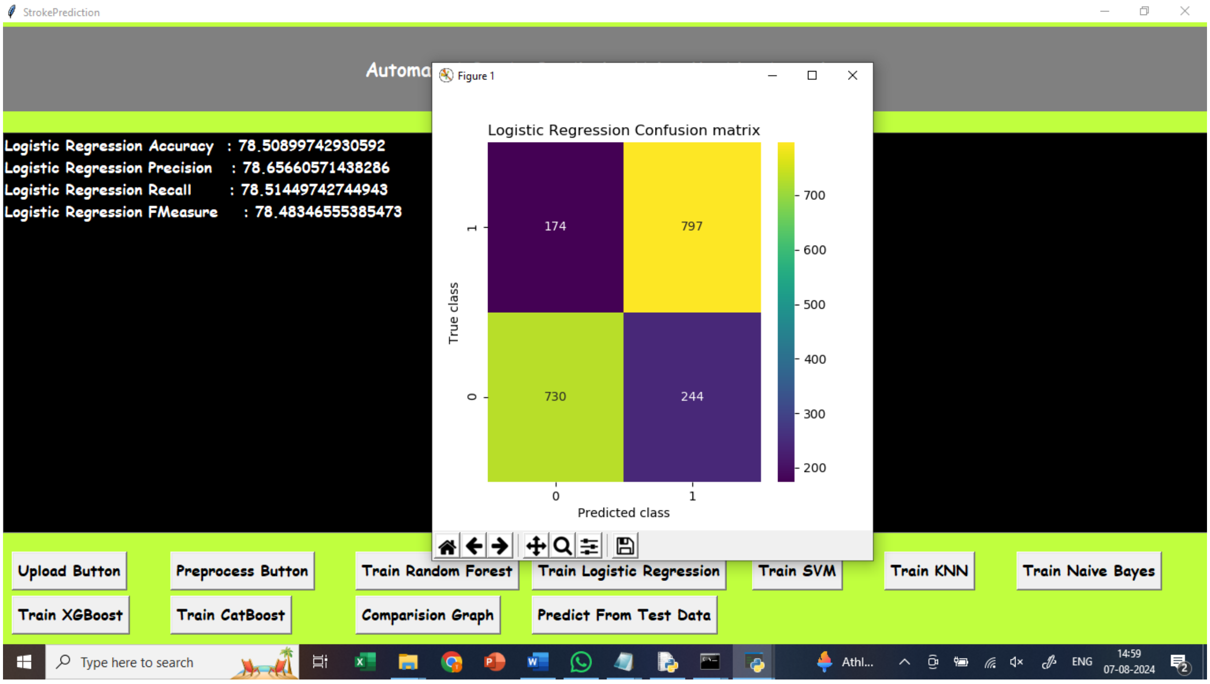
```
Normalize Features : [[0.5        0.81689453 0.         ... 0.80126489 0.375      0.33333333]
 [0.         0.74365234 0.         ... 0.67902317 0.         0.66666667]
 [0.5        0.97558594 0.         ... 0.23451205 0.3329918  0.66666667]
 ...
 [0.         0.42626953 0.         ... 0.12865848 0.31352459 0.66666667]
 [0.5        0.62158203 0.         ... 0.51320284 0.26229508 0.33333333]
 [0.         0.53613281 0.         ... 0.13922999 0.26844262 0.        ]]
Features available in dataset before selection : 10
Features available in dataset after selection : 9
Dataset train & test split as 80% dataset for training and 20% for testing
Training Size (80%): 7777
Testing Size (20%): 1945
```

In above screen, click on preprocess button, for splitting the dataset, 80% for training and 20% for testing.



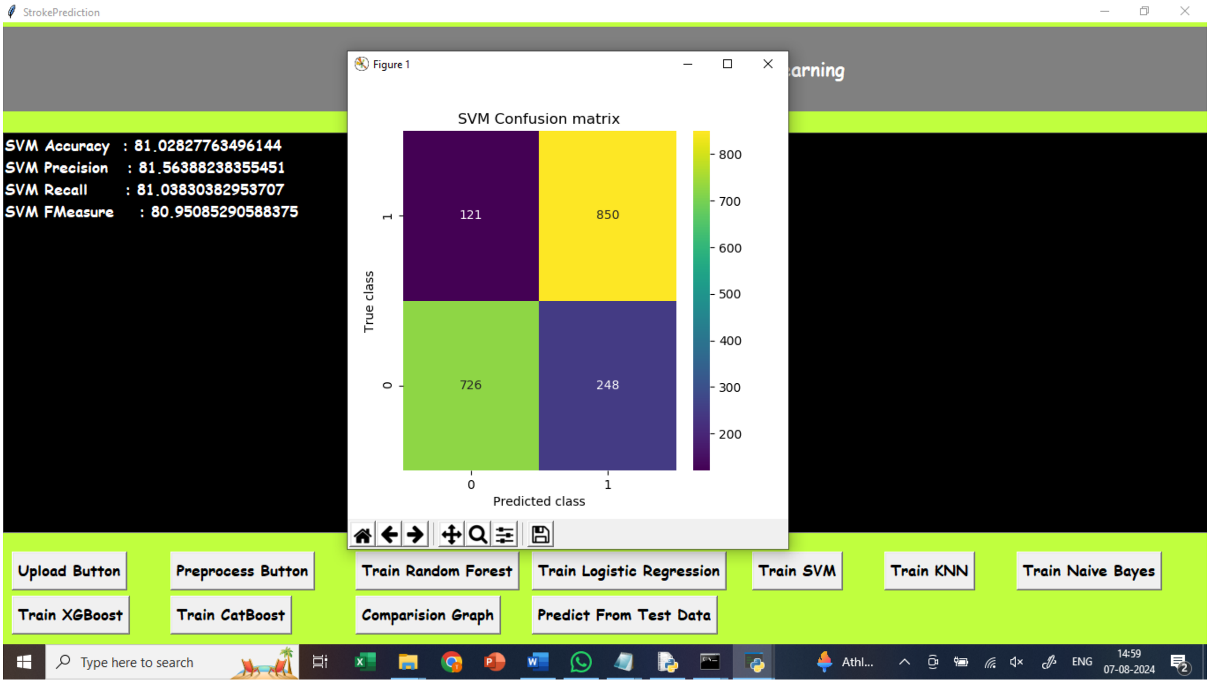
```
Random Forest Accuracy  : 95.0642673521851
Random Forest Precision  : 95.0727051607445
Random Forest Recall     : 95.0653658350903
Random Forest FMeasure    : 95.06410947779801
```

In above screen, click on Train Random Forest button, to train Random Forest and got 95% accuracy. In confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels where blue boxes represents incorrect prediction count which are very few and yellow and light green boxes represents correct prediction count.



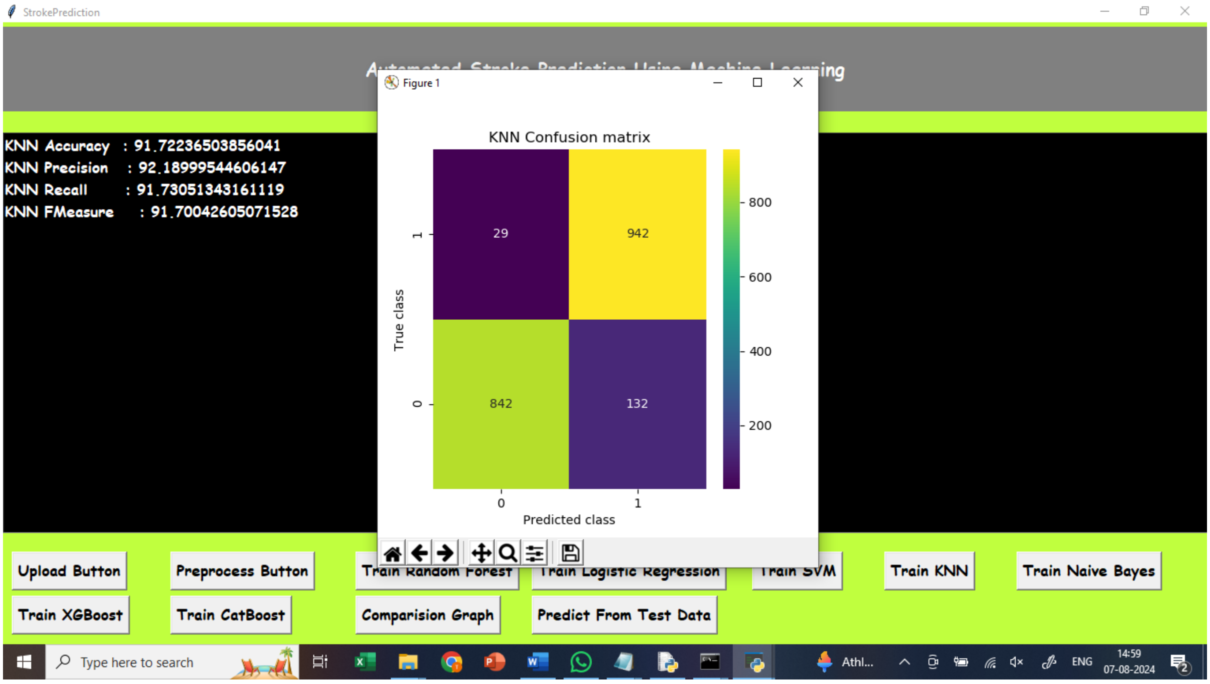
```
Logistic Regression Accuracy  : 78.50899742930592
Logistic Regression Precision  : 78.65660571438286
Logistic Regression Recall     : 78.51449742744943
Logistic Regression FMeasure    : 78.48346555385473
```

In above screen, click on Train Logistic Regression button, to train Random Forest and got 78% accuracy. In confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels where blue boxes represents incorrect prediction count which are very few and yellow and light green boxes represents correct prediction count.



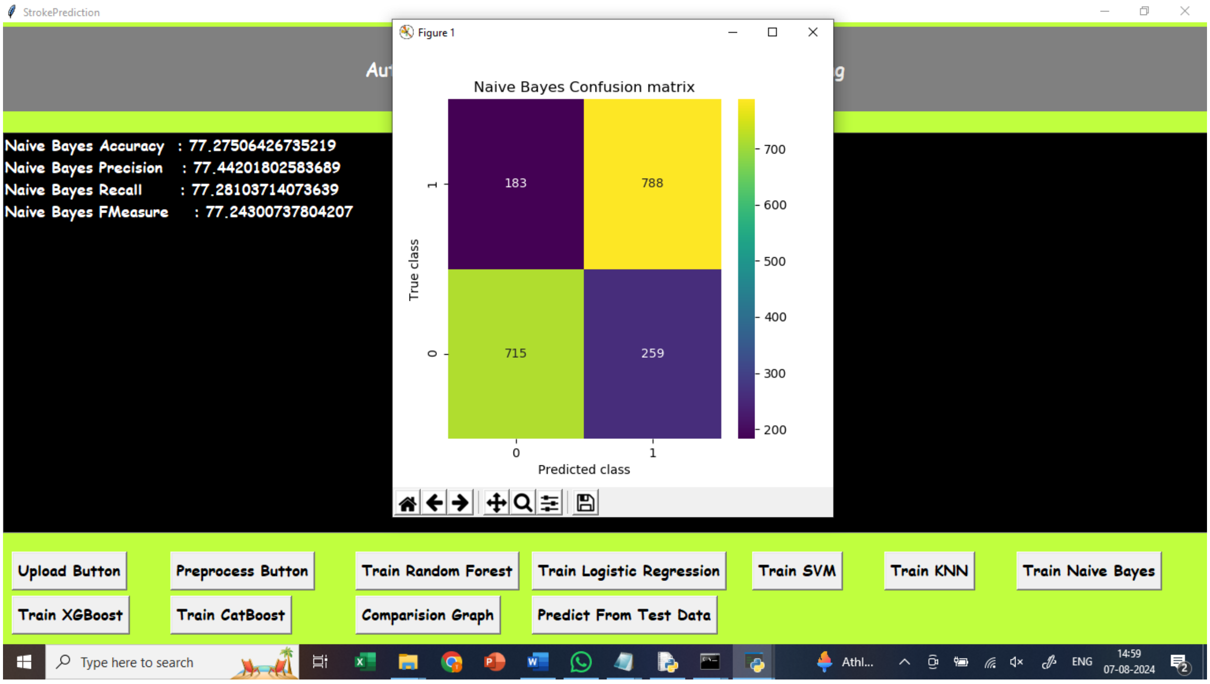
```
SVM Accuracy   : 81.02827763496144
SVM Precision   : 81.56388238355451
SVM Recall       : 81.03830382953707
SVM FMeasure     : 80.95085290588375
```

In above screen, click on Train SVM button, to train SVM and got 81% accuracy. In confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels where blue boxes represents incorrect prediction count which are very few and yellow and light green boxes represents correct prediction count.



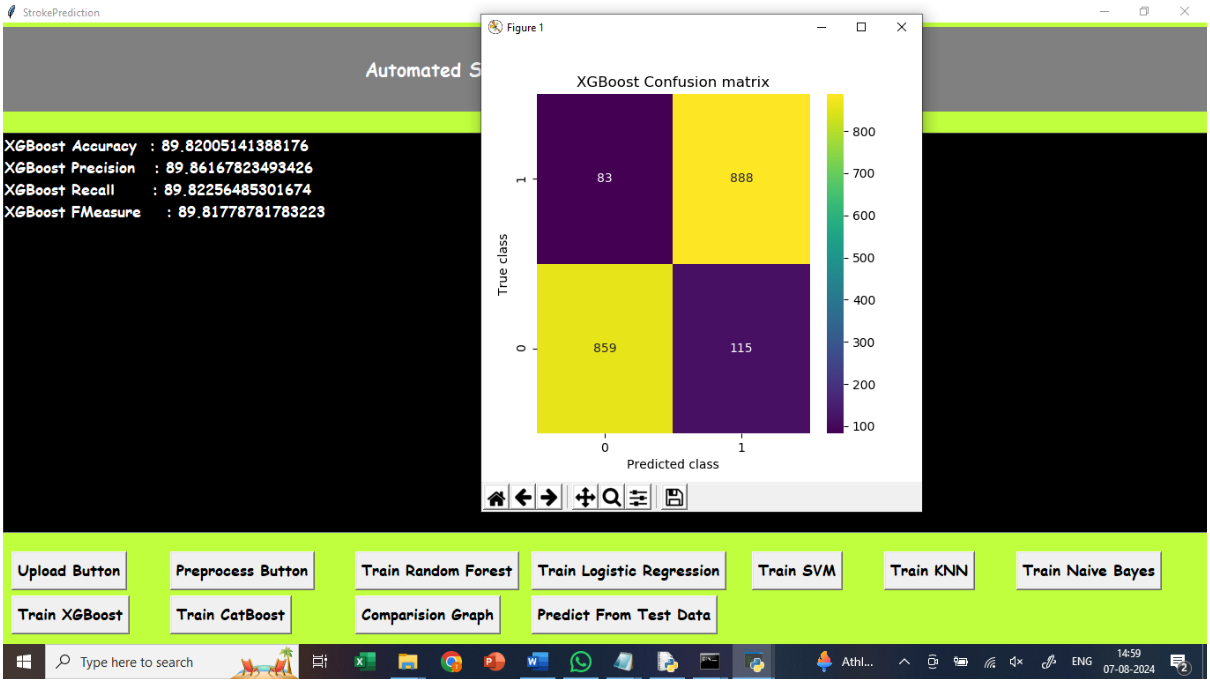
```
KNN Accuracy  : 91.72236503856041
KNN Precision  : 92.18999544606147
KNN Recall     : 91.73051343161119
KNN FMeasure    : 91.70042605071528
```

In above screen, click on Train KNN button, to train KNN and got 91% accuracy. In confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels where blue boxes represents incorrect prediction count which are very few and yellow and light green boxes represents correct prediction count.



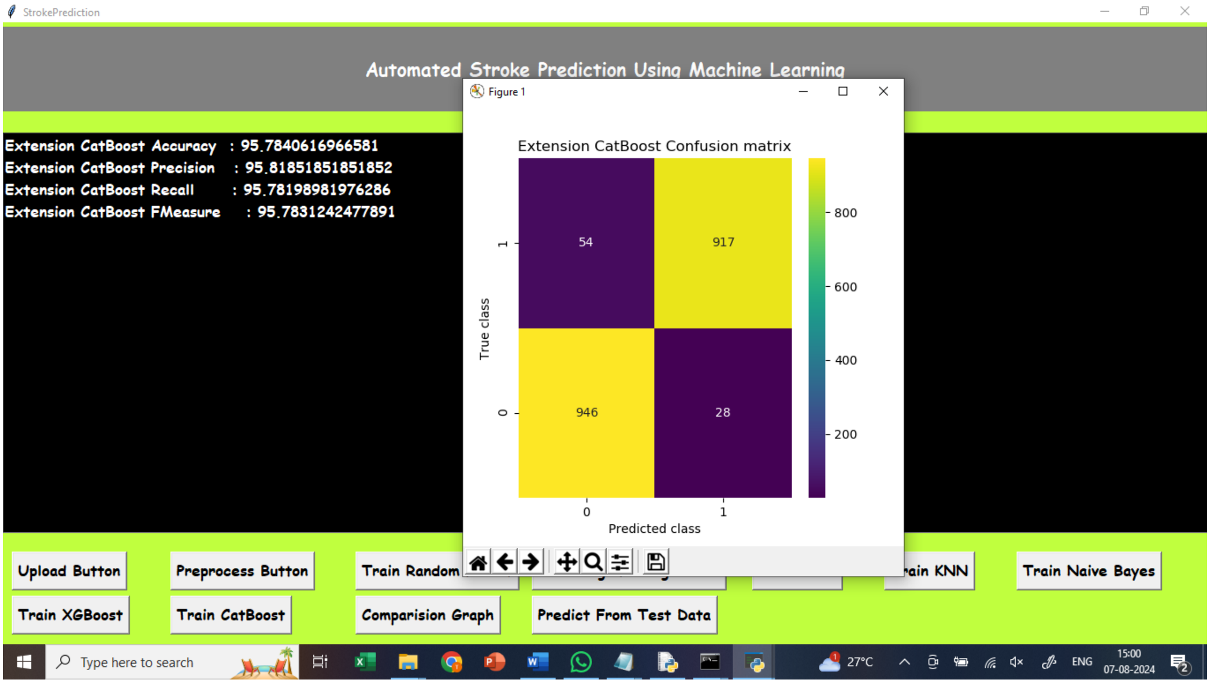
```
Naive Bayes Accuracy  : 77.27506426735219
Naive Bayes Precision  : 77.44201802583689
Naive Bayes Recall     : 77.28103714073639
Naive Bayes FMeasure    : 77.24300737804207
```

In above screen, click on Train Naïve Bayes button, to train Naïve Bayes and got 77% accuracy. In confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels where blue boxes represents incorrect prediction count which are very few and yellow and light green boxes represents correct prediction count.



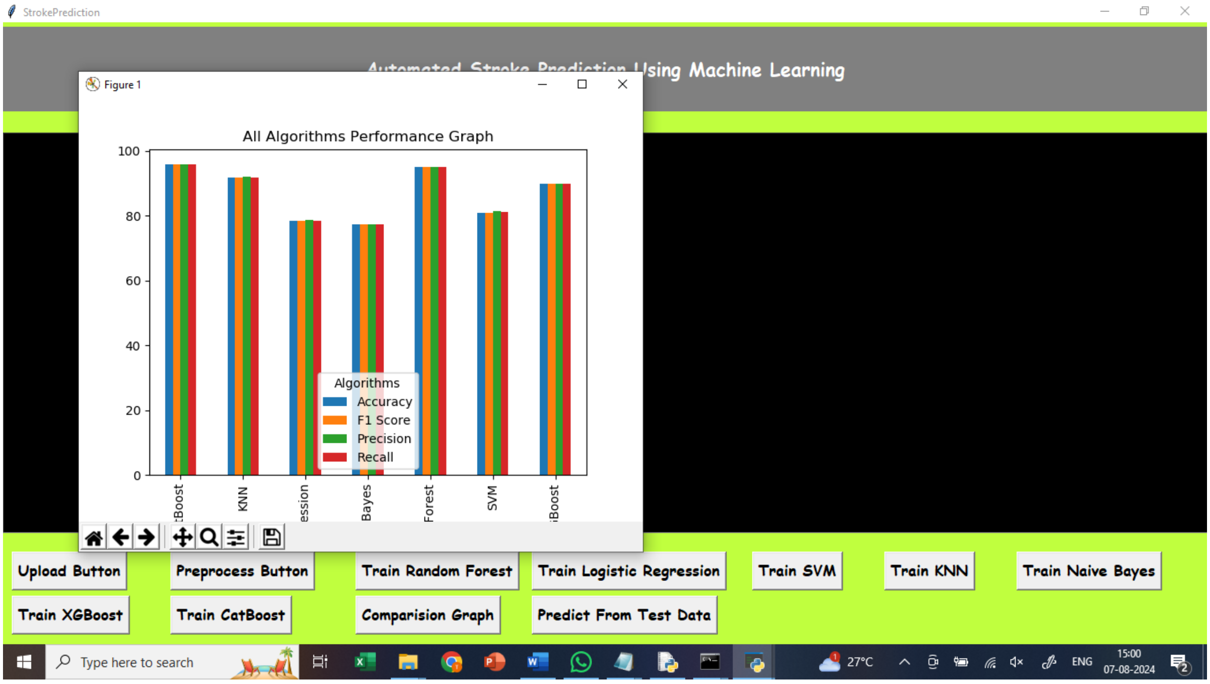
```
XGBoost Accuracy  : 89.82005141388176
XGBoost Precision  : 89.86167823493426
XGBoost Recall     : 89.82256485301674
XGBoost FMeasure    : 89.81778781783223
```

In above screen, click on XGBoost button, to train XGBoost and got 89% accuracy. In confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels where blue boxes represents incorrect prediction count which are very few and yellow and light green boxes represents correct prediction count.

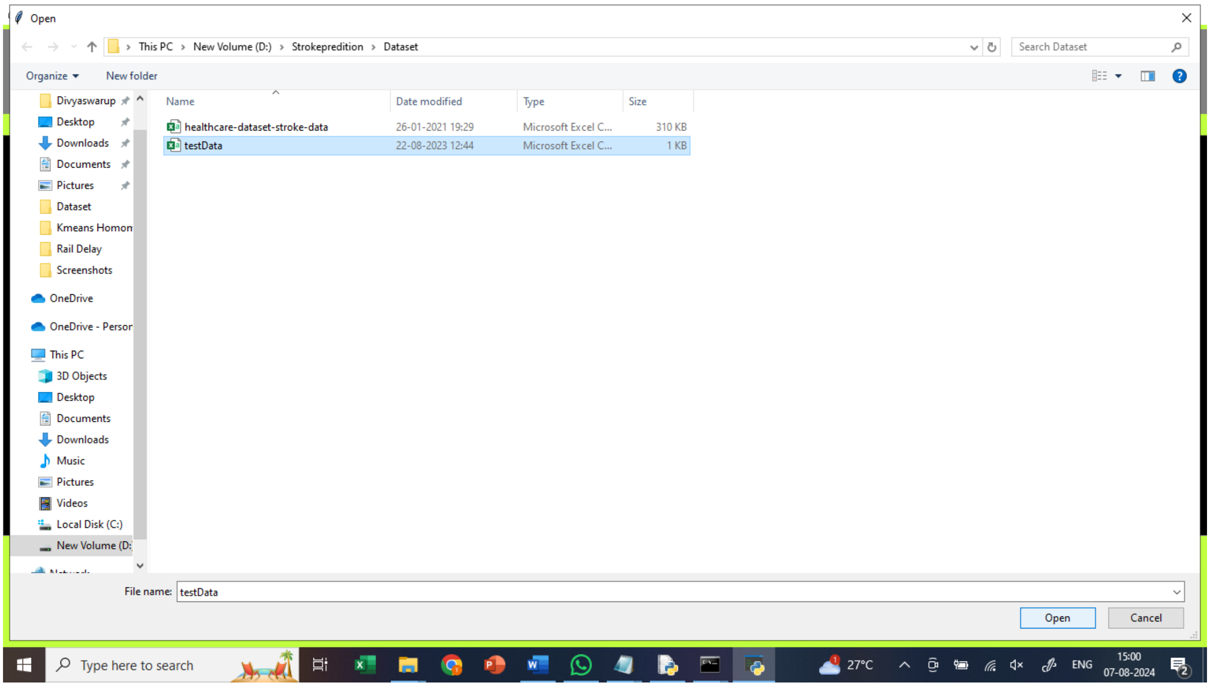


```
Extension CatBoost Accuracy  : 95.7840616966581
Extension CatBoost Precision   : 95.81851851851852
Extension CatBoost Recall      : 95.78198981976286
Extension CatBoost FMeasure     : 95.7831242477891
```

In above screen, click on Train CatBoost button, to train CatBoost and got 95% accuracy. In confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels where blue boxes represents incorrect prediction count which are very few and yellow and light green boxes represents correct prediction count.



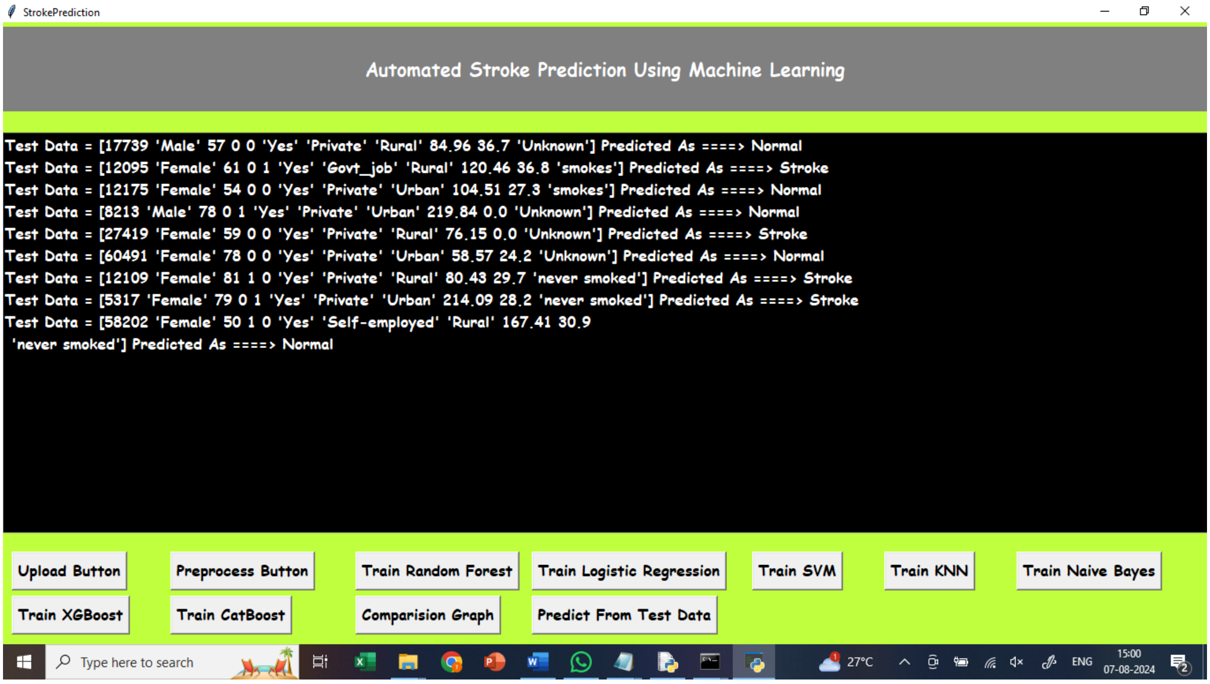
In above screen, click on comparision graph button, to compare all algorithms performance with four different parameters.



| Name | Date modified | Type | Size |
|---|---|---|---|
| healthcare-dataset-stroke-data | 26-01-2021 19:29 | Microsoft Excel C... | 310 KB |
| testData | 22-08-2023 12:44 | Microsoft Excel C... | 1 KB |

File name: testData

In above screen, click on predict from test data button, upload test data to predict whether data relates to stroke or normal



```
Test Data = [17739 'Male' 57 0 0 'Yes' 'Private' 'Rural' 84.96 36.7 'Unknown'] Predicted As ====> Normal
Test Data = [12095 'Female' 61 0 1 'Yes' 'Govt_job' 'Rural' 120.46 36.8 'smokes'] Predicted As ====> Stroke
Test Data = [12175 'Female' 54 0 0 'Yes' 'Private' 'Urban' 104.51 27.3 'smokes'] Predicted As ====> Normal
Test Data = [8213 'Male' 78 0 1 'Yes' 'Private' 'Urban' 219.84 0.0 'Unknown'] Predicted As ====> Normal
Test Data = [27419 'Female' 59 0 0 'Yes' 'Private' 'Rural' 76.15 0.0 'Unknown'] Predicted As ====> Stroke
Test Data = [60491 'Female' 78 0 0 'Yes' 'Private' 'Urban' 58.57 24.2 'Unknown'] Predicted As ====> Normal
Test Data = [12109 'Female' 81 1 0 'Yes' 'Private' 'Rural' 80.43 29.7 'never smoked'] Predicted As ====> Stroke
Test Data = [5317 'Female' 79 0 1 'Yes' 'Private' 'Urban' 214.09 28.2 'never smoked'] Predicted As ====> Stroke
Test Data = [58202 'Female' 50 1 0 'Yes' 'Self-employed' 'Rural' 167.41 30.9
 'never smoked'] Predicted As ====> Normal
```

In above screen, after uploading test data, it classified data as normal or stroke.