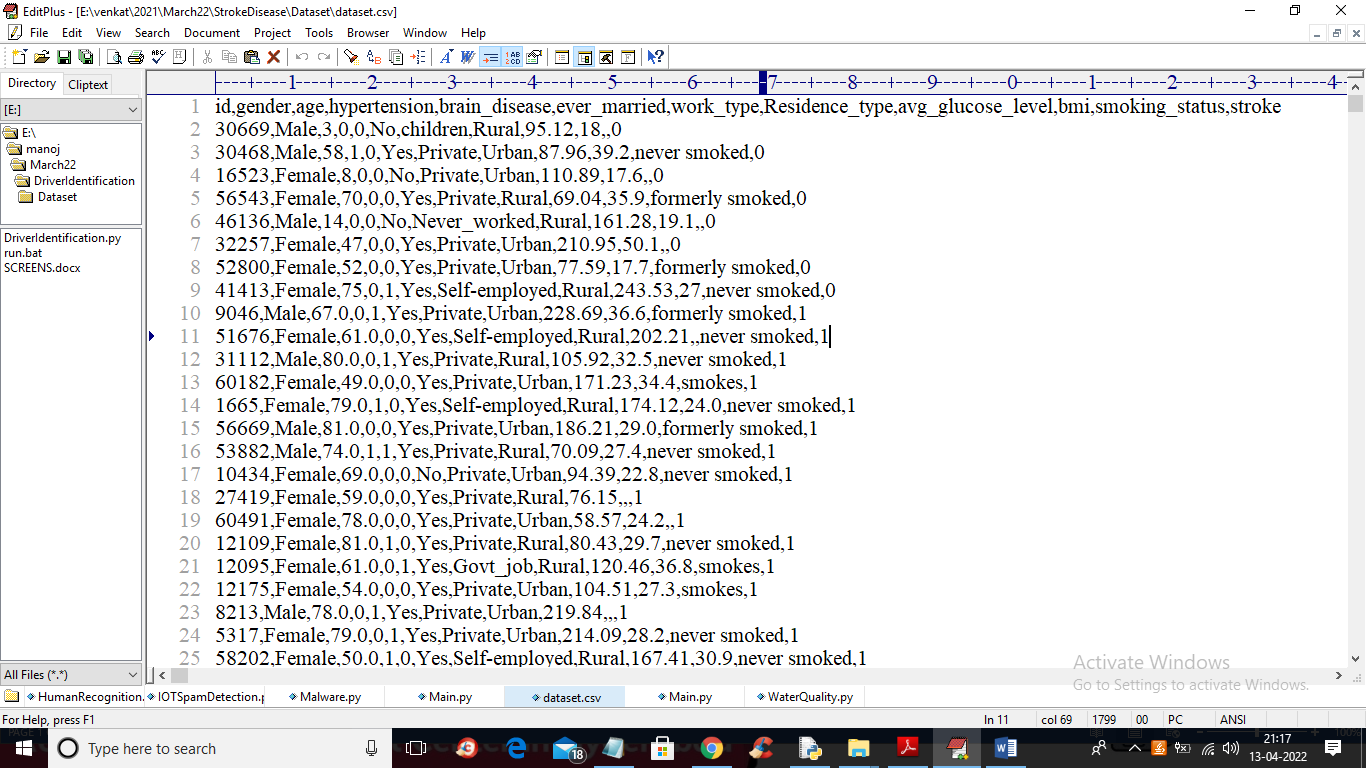
STROKE RISK PREDICTION WITH HYBRID DEEP TRANSFER LEARNING FRAMEWORK

In this paper author is using various machine learning algorithms such as J48, Naïve Bayes, KNN and Random Forest to predict stroke disease and Random Forest is giving best accuracy and in extension we have added ANN algorithm.

To train above algorithms author has collected his own dataset from hospitals and internet but we cannot collect that much data so I have downloaded STROKE dataset from KAGGLE website and below is the URL of that dataset

<https://www.kaggle.com/fedesoriano/stroke-prediction-dataset>

Below screen showing some dataset details



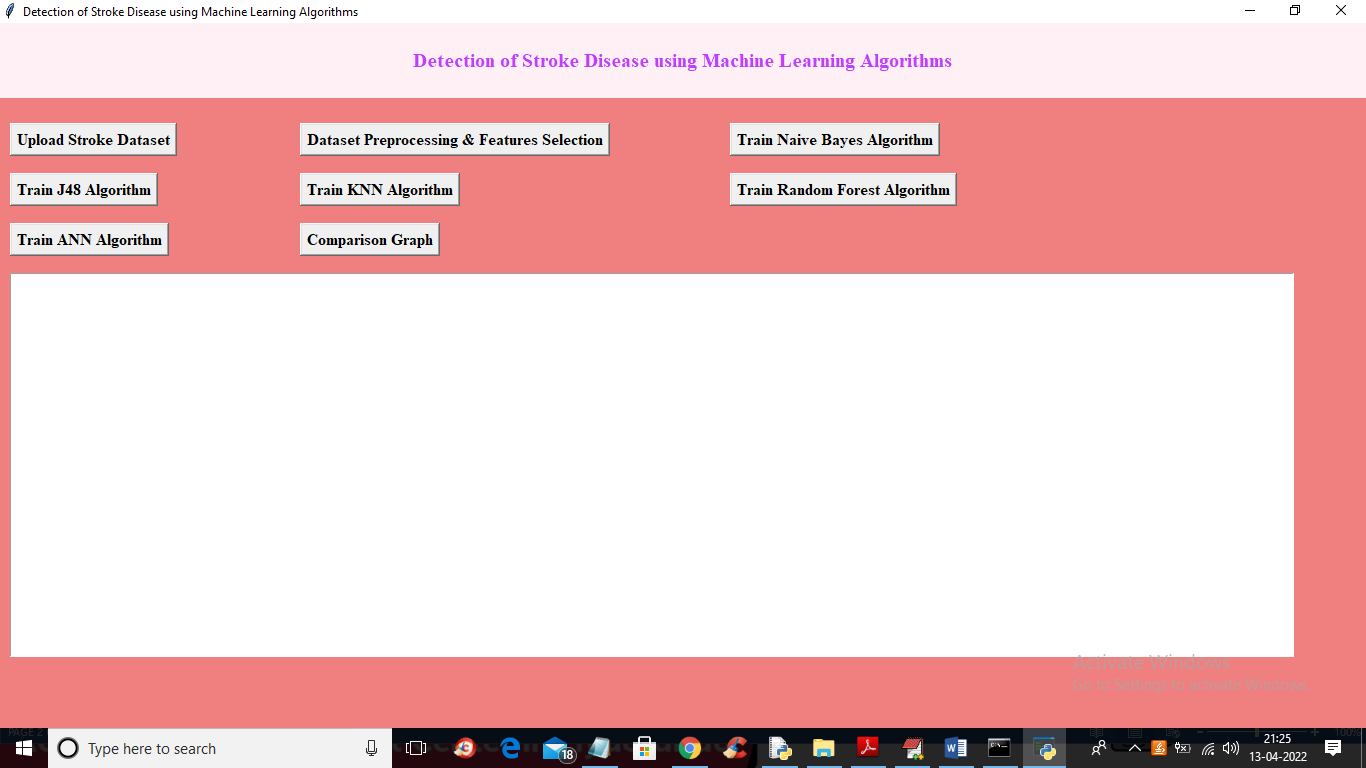
In above screen first row represents dataset column names and remaining rows are the dataset values and in last column we have stroke column labels as 0 or 1 where 0 means NO STROKE and 1 means STROKE. In above dataset we can see there are lots of missing values and contains non-numeric values so we need to apply label encoder to convert non-numeric into numeric dataset and then we select features from it.

To implement this project we have designed following modules

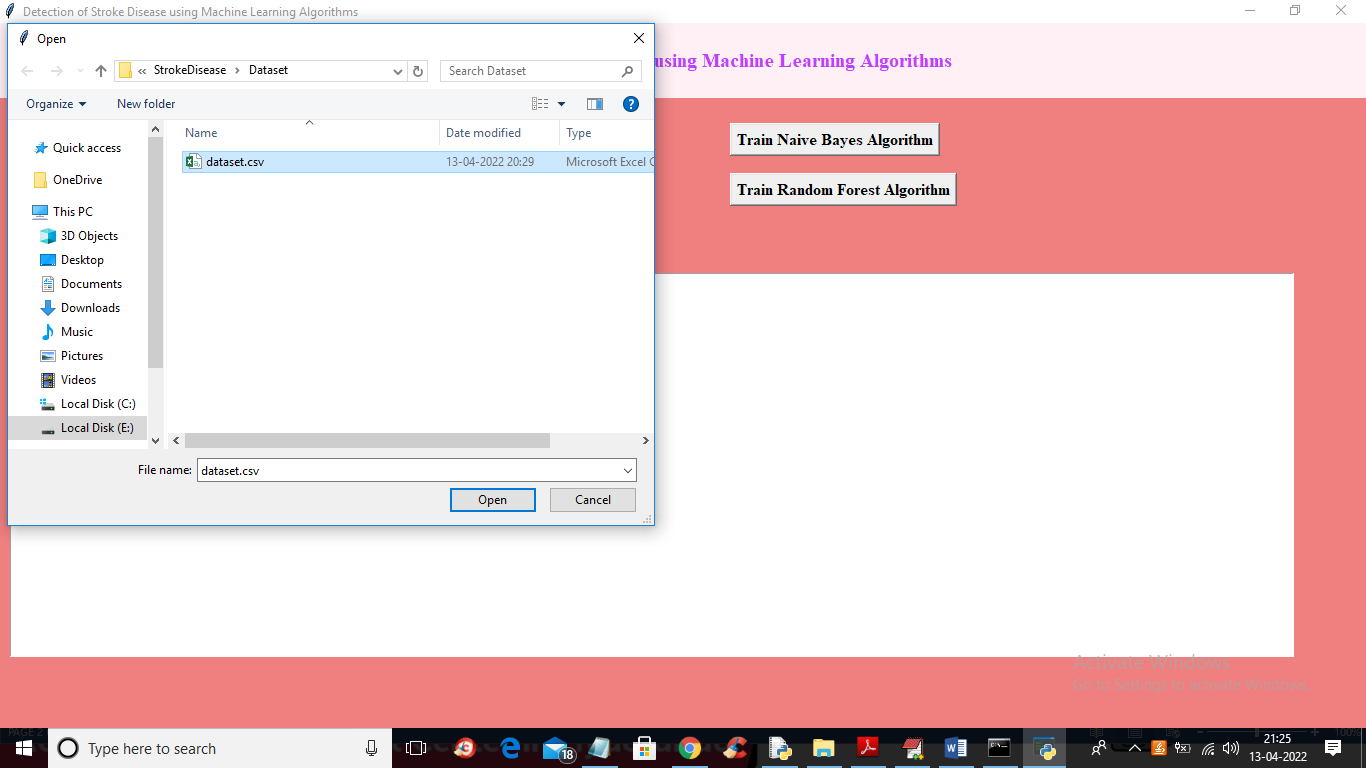
1. Upload Stroke Dataset: using this module we will upload dataset to application
2. Dataset Preprocessing & Features Selection: using this module we will clean dataset by replacing missing values with 0 and then apply label encoding algorithm to convert non-numeric values to numeric values and then select features from dataset and then split dataset into train and test where application used 80% data for training and 20% for testing
3. Train Naive Bayes Algorithm: above training data will be input to Naïve Bayes algorithm to train a model and this model will be applied on test data to calculate accuracy
4. Train J48 Algorithm: above training data will be input to J48 algorithm to train a model and this model will be applied on test data to calculate accuracy
5. Train KNN Algorithm: above training data will be input to KNN algorithm to train a model and this model will be applied on test data to calculate accuracy
6. Train Random Forest Algorithm: above training data will be input to Random Forest algorithm to train a model and this model will be applied on test data to calculate accuracy
7. Train ANN Algorithm: above training data will be input to ANN algorithm to train a model and this model will be applied on test data to calculate accuracy
8. Comparison Graph: using this module we will plot accuracy comparison graph between all algorithms

SCREEN SHOTS

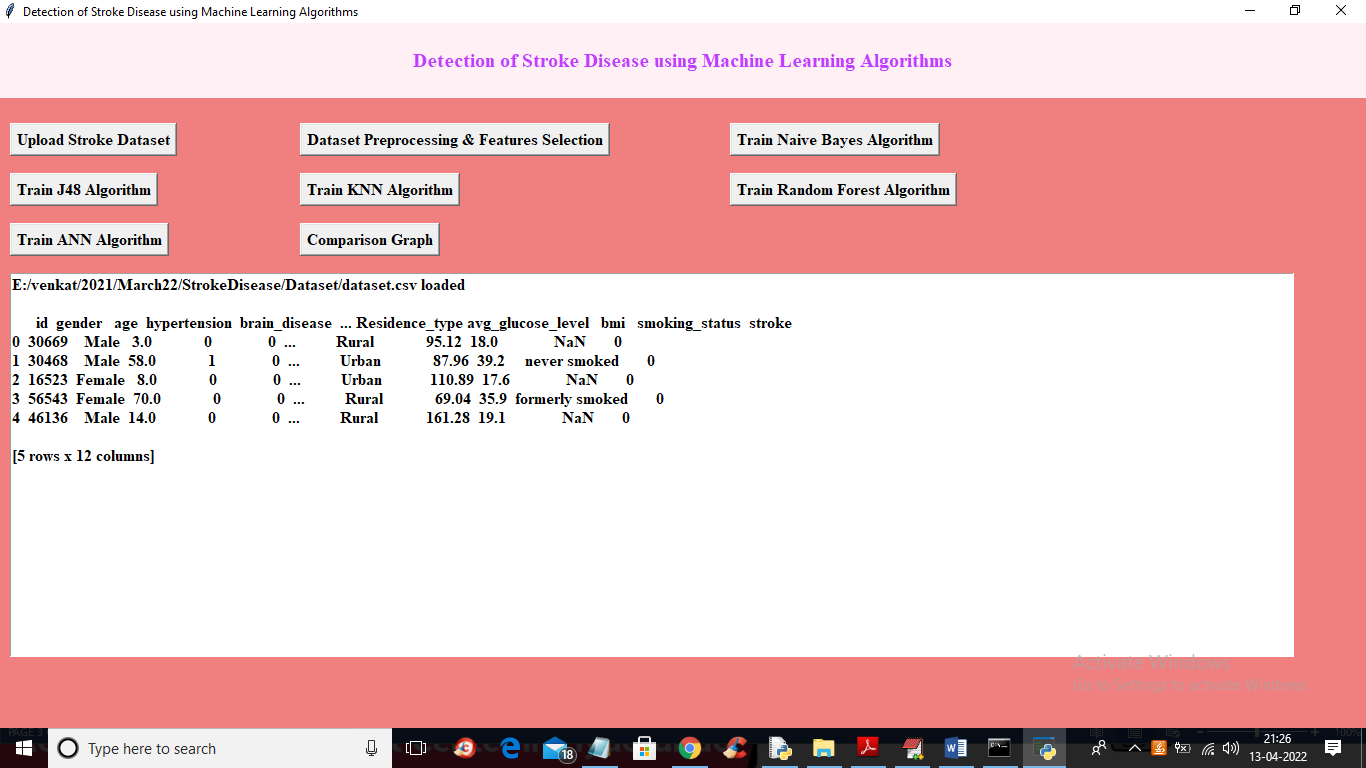
To run project double click on ‘run.bat’ file to get below output



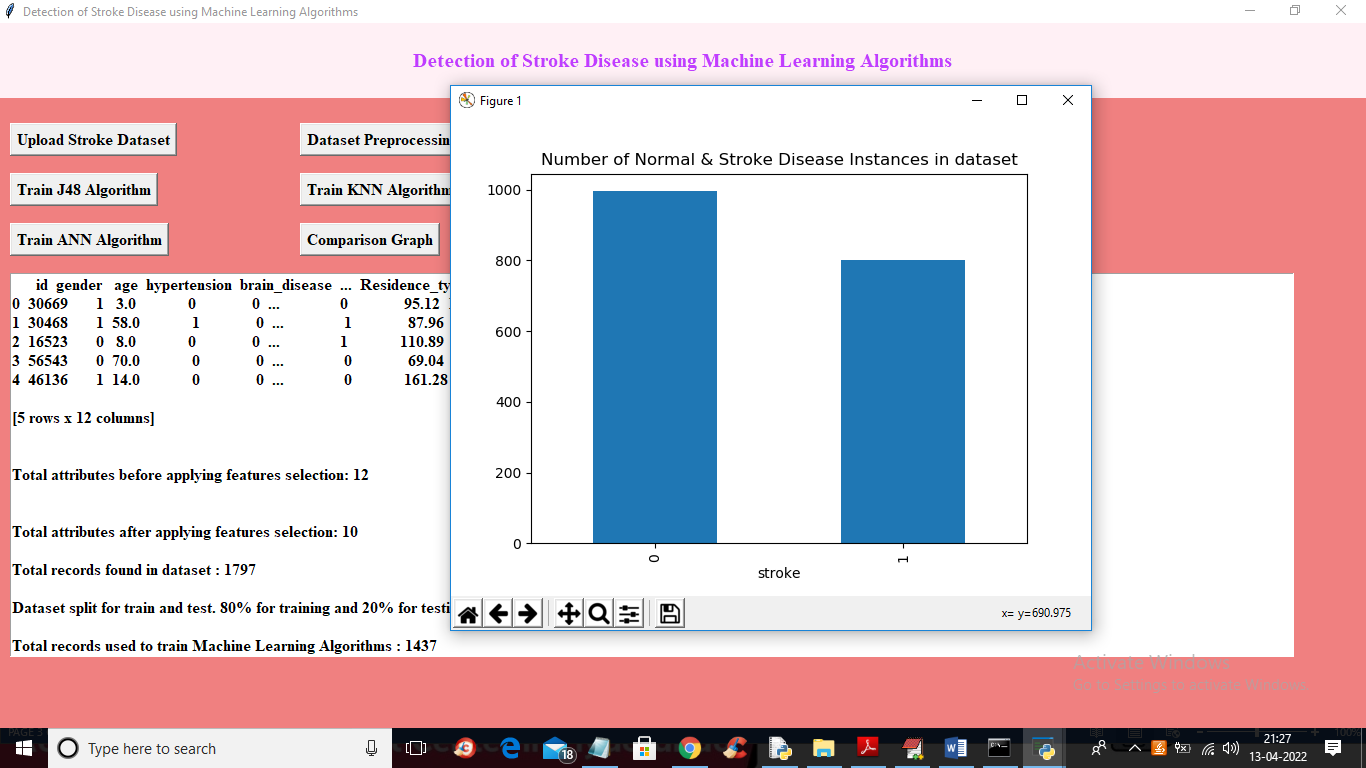
In above screen click on ‘Upload Stroke Dataset’ button to upload dataset



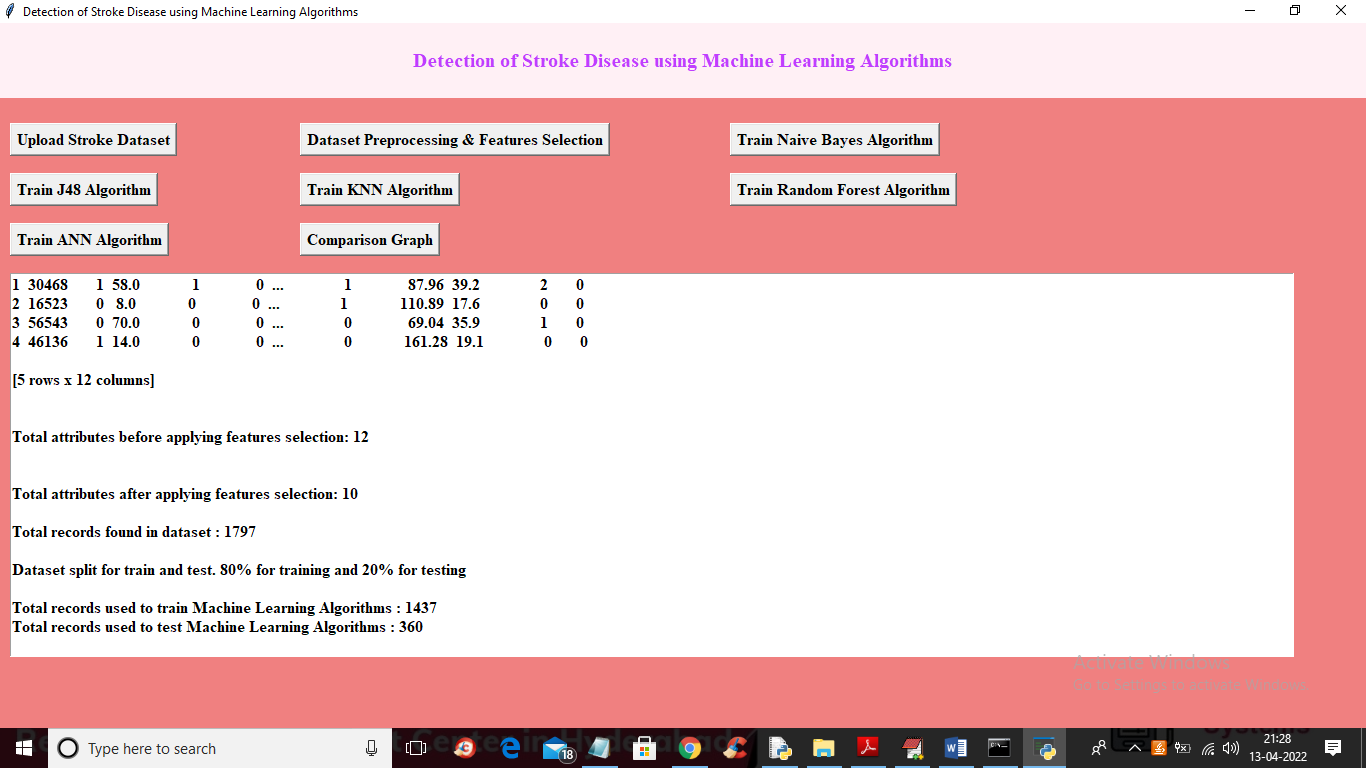
In above screen selecting and uploading dataset.csv file and then click on ‘Open’ button to load dataset and to get below output



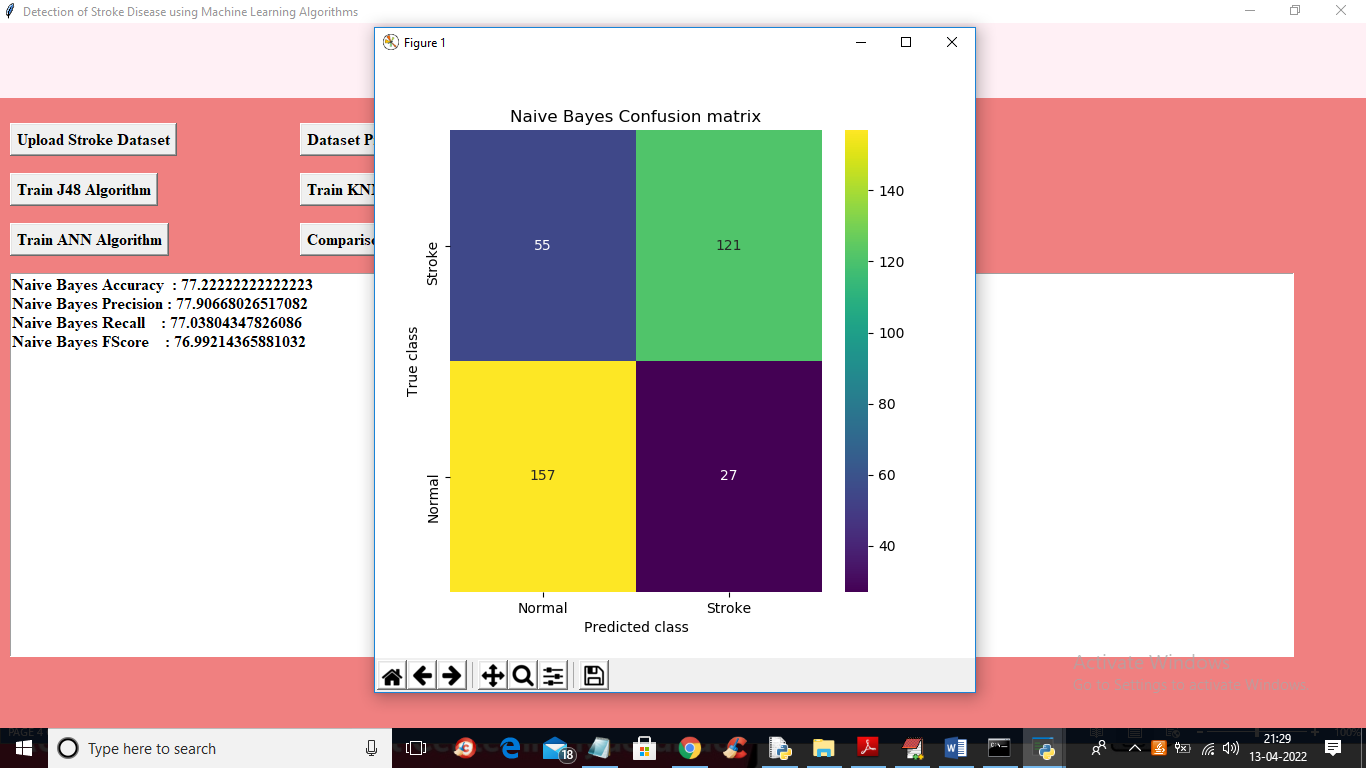
In above screen we can see dataset loaded and dataset contains so many missing and non-numeric data so click on ‘Dataset Preprocessing & Features Selection’ button to process dataset and to get below output



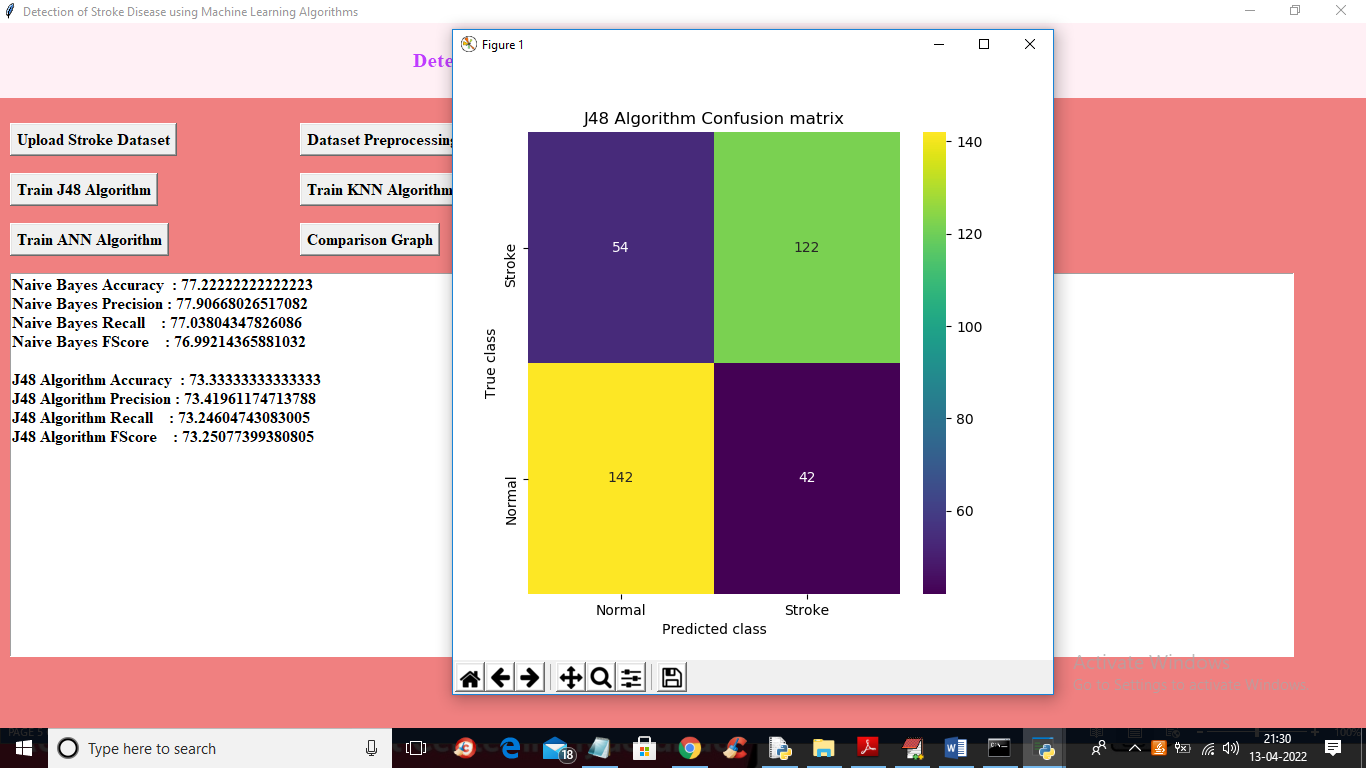
In above graph x-axis represents 0 (normal) and 1 (stroke) and y-axis represents number of instances available in those categories in dataset and now close above graph and see below screen



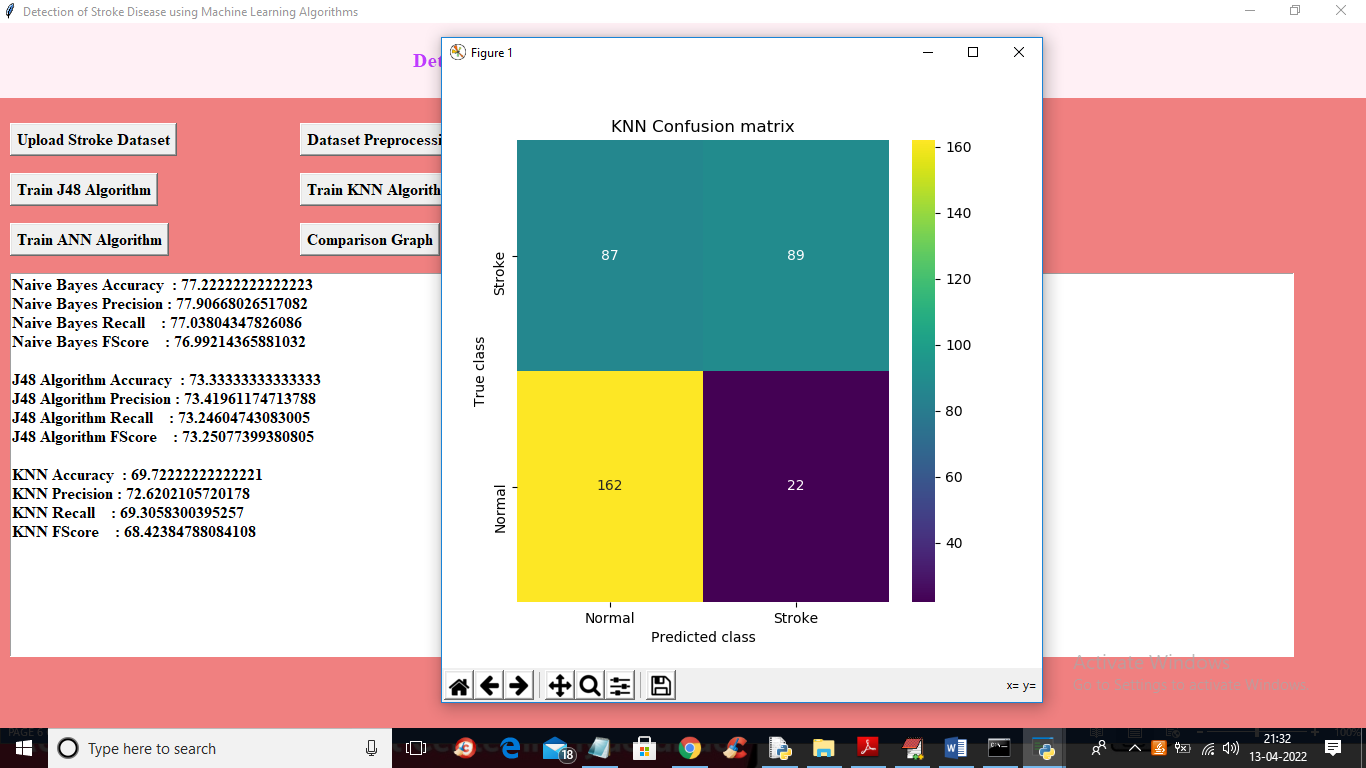
In above screen we can see all dataset converted to numeric format and then split dataset into train and test and now click on ‘Train Naïve Bayes Algorithm’ button to train Naïve Bayes on above dataset and get below output



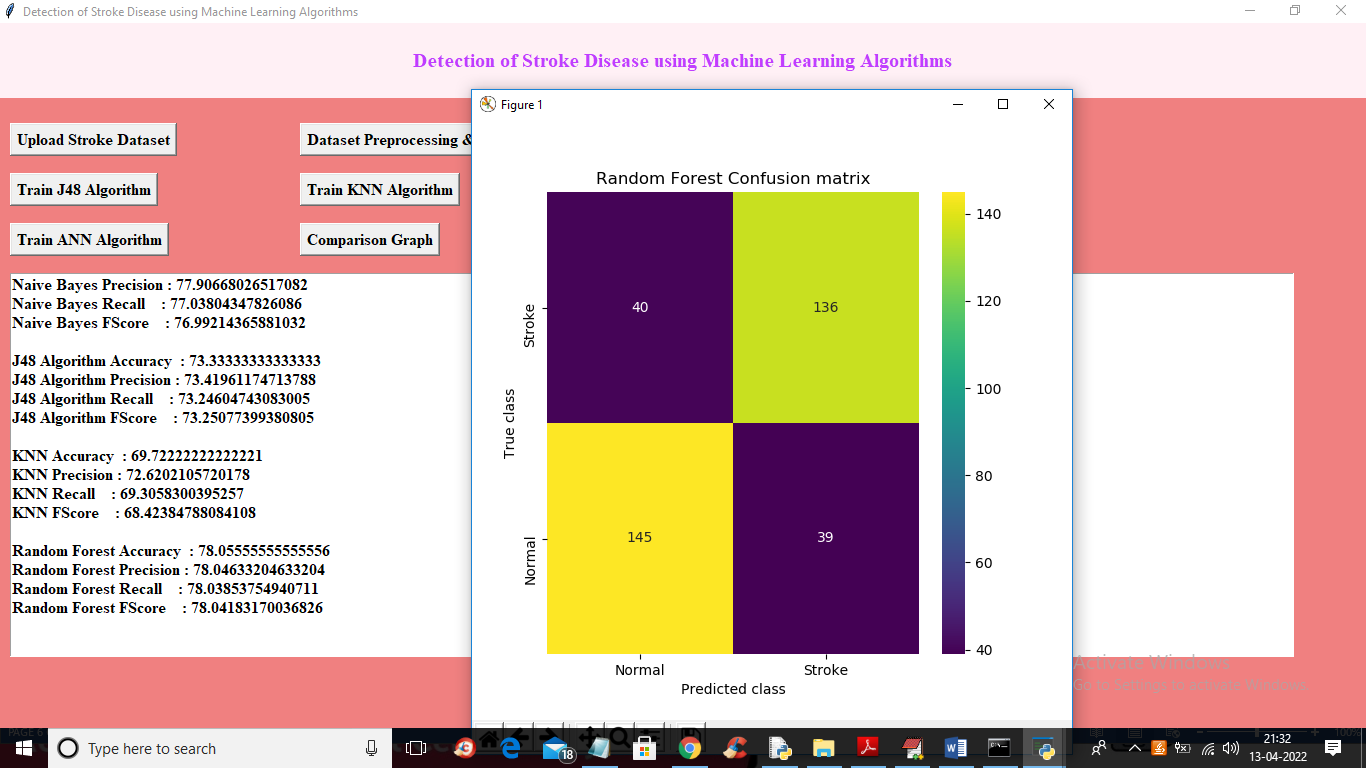
In above screen with Naïve Bayes we got 77% accuracy and in confusion matrix graph we can see number of correct and incorrect prediction by Naïve Bayes. Now click on ‘Train J48 Algorithm’ button to get below output



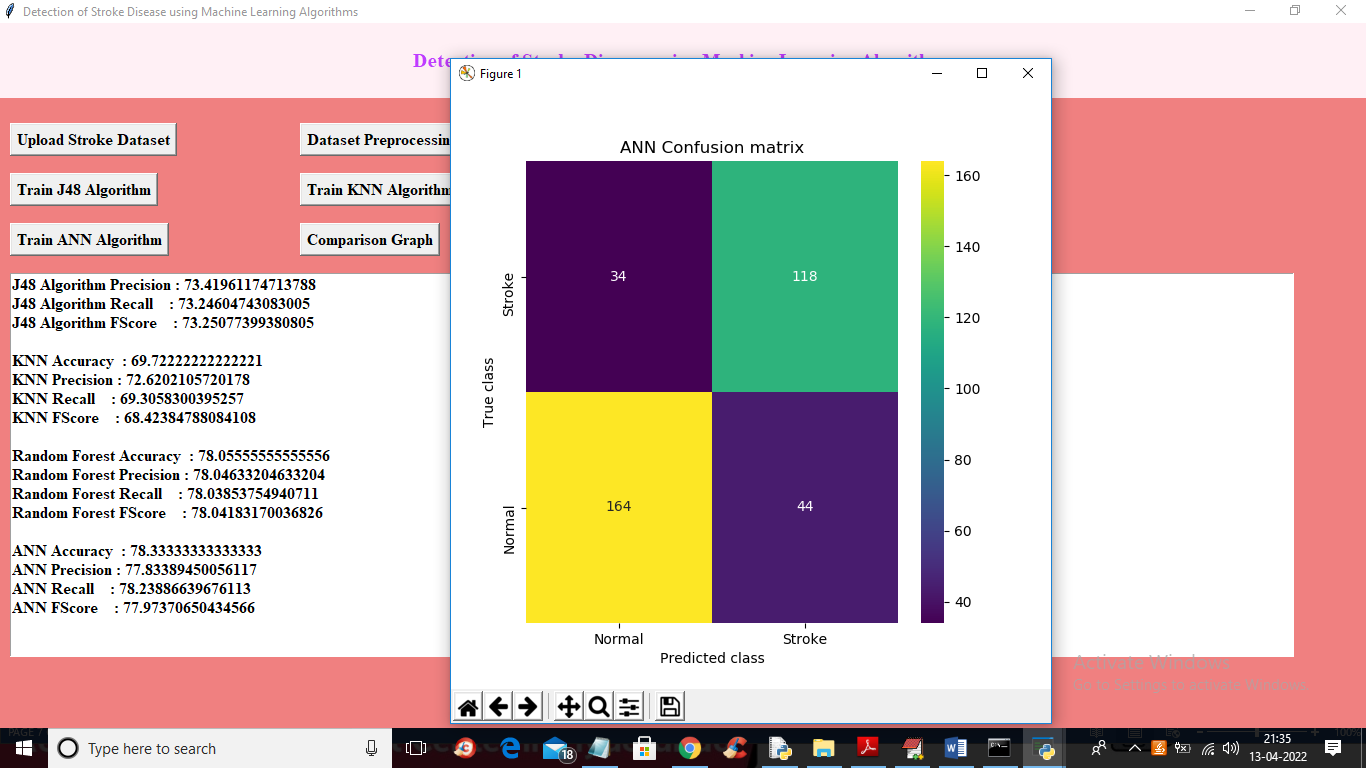
In above screen with J48 we got 73% accuracy and in confusion matrix graph we can see number of correct and incorrect prediction by J48.Now close above Graph and then click on ‘Run KNN Algorithm’ button to get below output



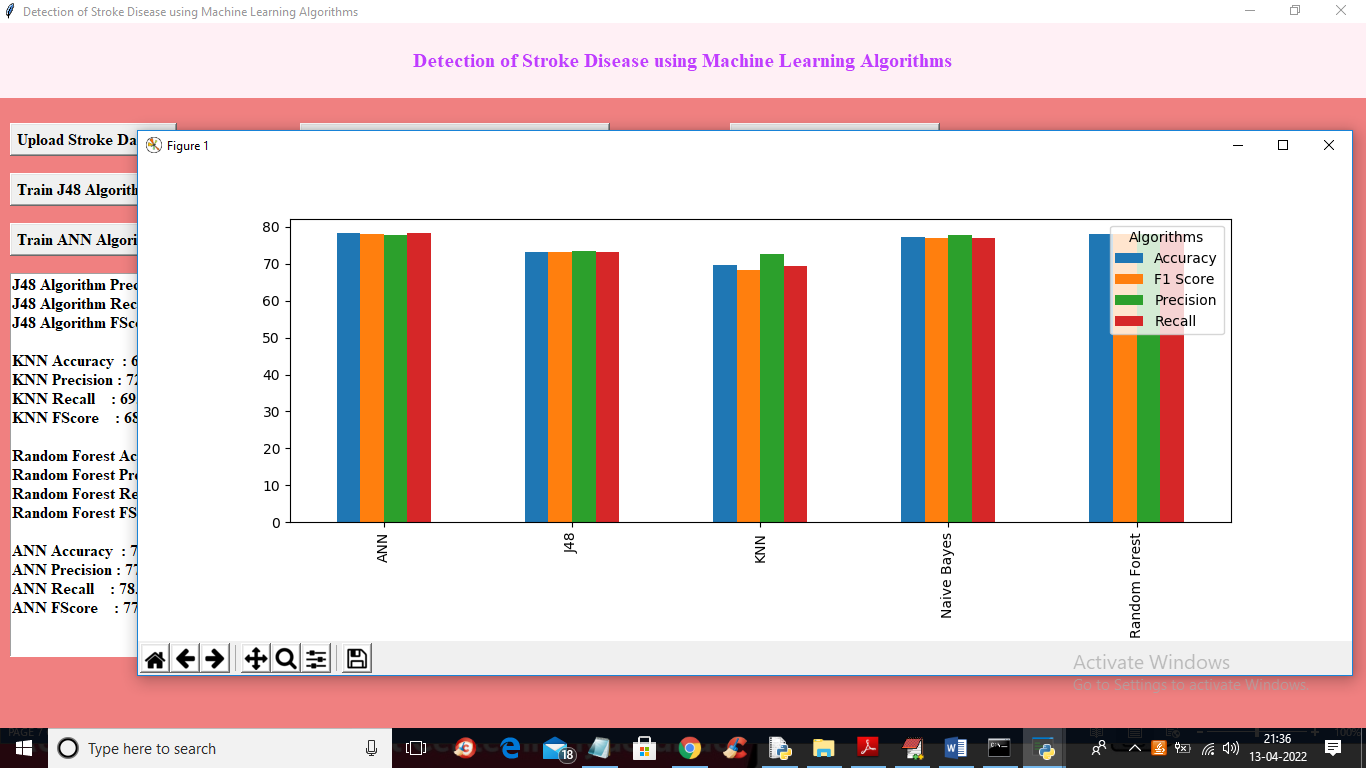
In above screen with KNN we got 69% accuracy and in confusion matrix graph we can see number of correct and incorrect prediction by KNN. Now close above Graph and then click on ‘Run Random Forest Algorithm’ button to get below output



In above screen with Random Forest we got 78% accuracy and in confusion matrix graph we can see number of correct and incorrect prediction by Random Forest. Now close above Graph and then click on ‘Run ANN Algorithm’ button to get below output



In above screen with ANN we got 78.33% accuracy and in confusion matrix graph we can see number of correct and incorrect prediction by ANN and in all algorithm ANN got high accuracy. Now close above Graph and then click on ‘Comparison Graph’ button to get below graph



In above graph x-axis represents algorithm names and y-axis represents accuracy and other metrics like precision, recall etc. different colour bar represents different metrics and in all algorithms ANN got high accuracy