Stroke Prediction

Deep Vasan, Tatvam Shah, Rhythm Jain
B20CS083, B20CS077, B20EE094

Abstract – This is a report of our Course Project aimed at Stroke Prediction using Machine Learning. We took a Dataset which contained general information about the person like age, marital status, residential type, smoking status and health related features like previous history of suffering from hypertension, heart disease and some general health indicators like BMI. We used various classification algorithms and compared their results in this report.

I. Introduction

In the modern world, the number of diseases and ailments have increased. With the improvements in the methods of diagnosis, machine learning has played a wonderful part in the early identification and treatment of several diseases and medical conditions such as Cancer Detection, Respiratory Disease Detection and so on. On similar lines, our project aims to analyze and make the use of machine learning techniques to classify the vulnerability of a person to succumb to stroke on the basis of a few inputs describing the lifestyle of the individual and his previous medical conditions.

.

Dataset

Healthcare-Dataset-Stroke-Data :
The train dataset contains 1556 rows where each row has 12 columns containing :

- id column containing the id of the individual
- gender column stating the gender of the individual
- age column stating the age of the individual
- hypertension column giving the history of the individual whether suffered from hypertension in the past
- heart disease column giving the history of the individual whether suffered from heart disease in the past
- ever married column giving the marital status of the individual
- work type column stating the employment type of the individual (Government job, Private, Self Employed etc)
- residence type stating the residence type of the individual (Rural or Urban)
- avg_glucose_level giving the average blood glucose level of the individual
- bmi providing the body mass index of the individual
- smoking_status column stating the smoking status of the individual (smokes, formerly smoked, never smoked etc)
- stroke column (label column) providing the stroke occurrence status of the individual

The dataset has been split into train, validation and test with validation size 0.16 and test size of 0.2.

II. Methodology

Overview

There are various classification algorithms present out of which we shall implement the following

- *Random Forest Classifier*
- *Support Vector Machine Classifier*
- *XGBoost Classifier*
- *Multilayer Perceptron Classifier*
- *K - Nearest Neighbor Classifier*

Exploring the dataset and pre-processing

The dataset was analyzed and it was observed that there were 201 NULL values in the bmi column. The dataset was highly skewed, that is the number of rows with output labels as 1 were much greater than the number of rows with output as 1, which might make the model biased towards the prediction 1, which could be a severe issue(discussed in analysis part). Hence, to improve the model, SMOTE algorithm was used for oversampling, so that the dataset has an approximately equal amount of 0 and 1 labels, thus, leading to a better and more safer model(according to the point of view of diagnosis). These were replaced by the mean of the corresponding columns. Label Encoder was used to encode the columns with string values which were gender, ever_married, work_type, residence_type, and smoking_status into numerical values. The columns age, bmi and average_glucose_level were scaled using Min-Max Scaler. Heatmaps, histograms and stacked barcharts were plotted for visualization of data.

Hyperparameter Tuning for optimal hyperparameters

For the best results across the models, hyperparameter tuning was performed on the models using manual hyperparameter tuning as well as RandomizedSearchCV. Different hyperparameters were chosen for the models such that it would lead to better performance as well as reduce the chances of overfitting.

Implementation of classification algorithms

- *Random Forest Classifier :* Random Forest Classifiers use boosting ensemble methods to train upon various decision trees and produce aggregated results.It is one of the most used machine learning algorithms.
  - Tree with max_depth of 9 was used along with the hyperparameters obtained from RandomizedSearchCV.
- *KNN (k - nearest neighbors) :*KNN are supervised algorithms which classify on the basis of distance from similar points.Here k is the number of nearest neighbors to be considered in the majority voting process.
  - KNN Classifier with n-neighbors = 5 was used along with the hyperparameters obtained from RandomizedSearchCV.
- *Support Vector Machine :*In SVM , data points are plotted into n-dimensional graphs which are then classified by drawing hyperplanes.
  - Linear SVM Classifier was used along with the hyperparameters obtained from RandomizedSearchCV.
- *Multilayer Perceptron :*MLP is a feedforward Neural Network which uses backpropagation to update weights and improve the results.
  - MLP Classifier was used along with the hyperparameters obtained from RandomizedSearchCV.
- *XGBoost :* XGBoost is a scalable and highly accurate implementation of gradient boosting that pushes the limits of computing power for boosted tree algorithms, being built largely for energizing machine learning model performance and computational speed.
  - XGBClassifier was used along with the hyperparameters obtained from RandomizedSearchCV.

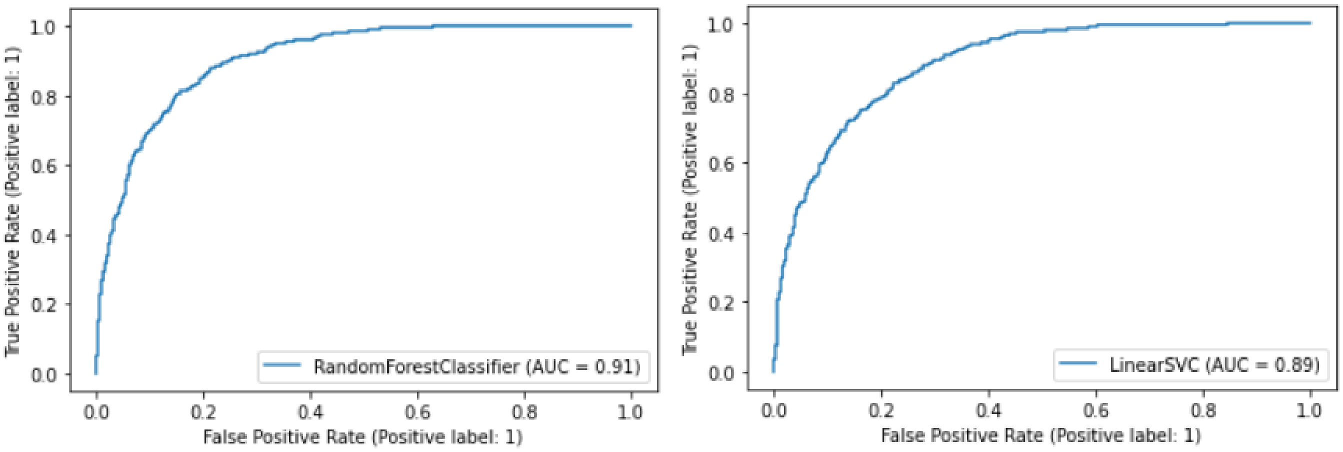
III. Evaluation of models

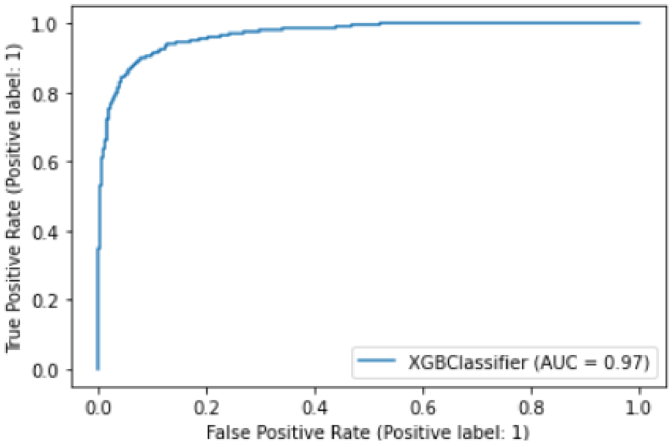
The models implemented were evaluated using techniques like - Classification report : precision , recall , f1 score and AUC , ROC plots , accuracy score, type -1 error and type - 2 error, most important being type - 2 error in this case. The roc auc score of SVC is not done since SVC doesn't have the predict_proba attribute which is used for calculating roc auc score. According to the type 2 error , linear SVC is the best model.

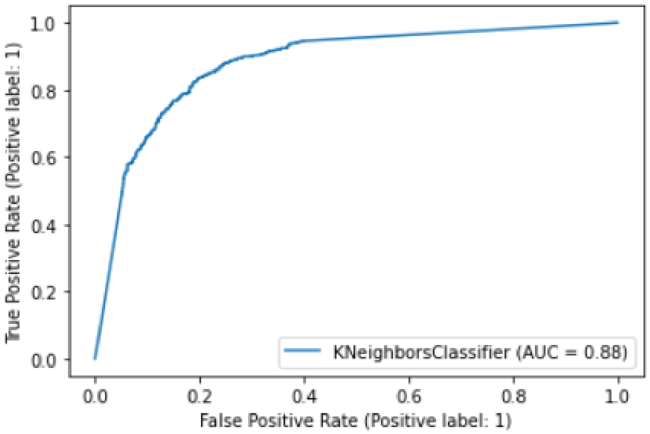
	ROC AUC Score	Type - 2 Error
Random Forest	0.9524	0.1904
Linear SVC	-	0.0851
XG Boost	0.9605	0.1470
KNN	0.9473	0.1873
MLP	0.9422	0.1595

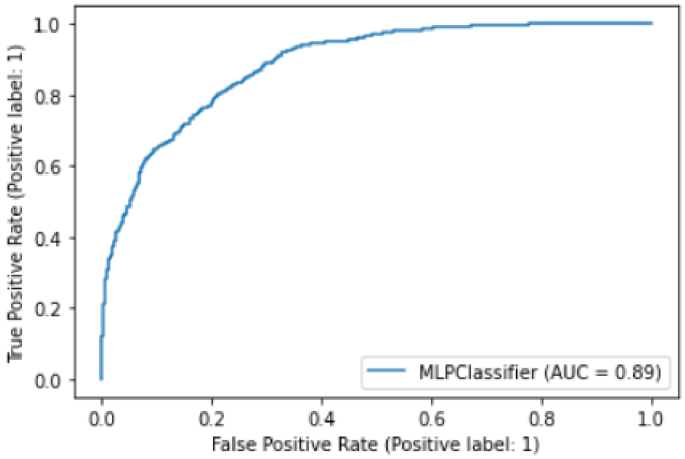
ROC - Plots

ROC plots for a) Random forest , b) Linear SVC, c) XG Boost, d) KNNClassifier, e) MLP Classifier





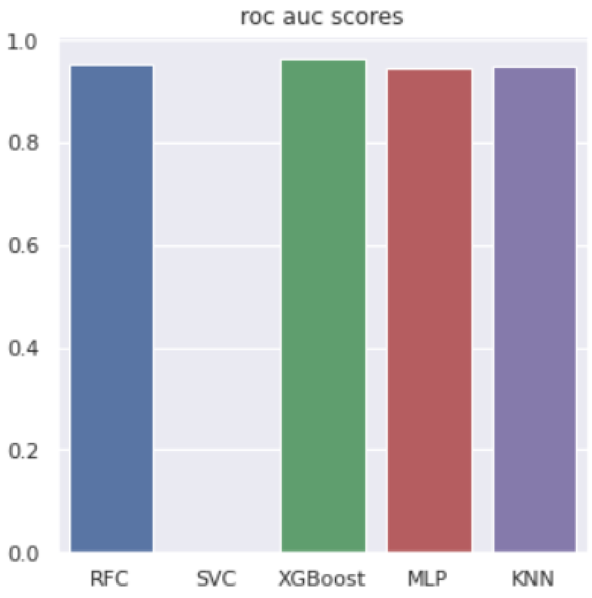


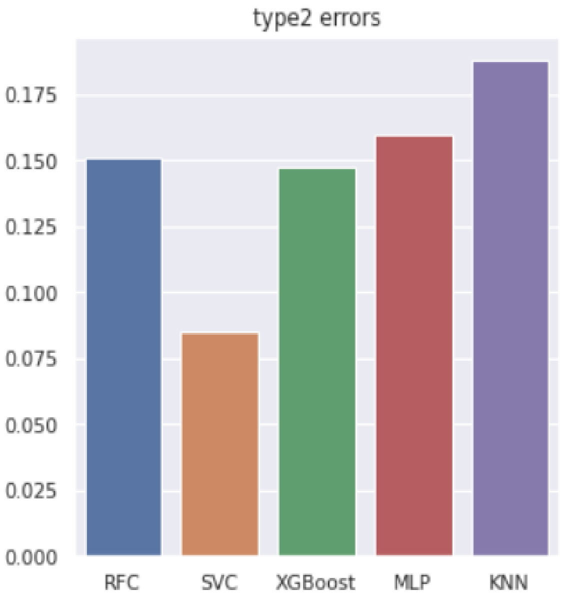


IV. Results and Analysis

It can be seen from the above table that SVC Classifier performed the best on the testing data followed by XGB Classifier and Classifier. The models gave a decent accuracy on the validation and the testing data. However, most of the predicted values were zero, including some ones which were classified as zero, leading to type - 2 error i.e., some people who were vulnerable to stroke were classified as not vulnerable, which could prove to be life threatening. Hence, we applied the SMOTE algorithm to the training dataset, which increased the number of samples in the dataset which predicted one as output labels by resampling. Thus, the model was no longer biased towards zeros and thus, even though the accuracy of the models decreased, it led to a reduction in the type - 2 error, which is more safer here (according to medical point of view).

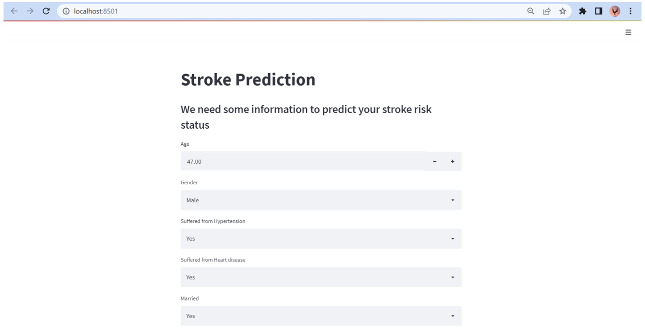
Roc scores and type 2 errors for manually tuned models :





V. Additional Work

A webapp with frontend using Flask framework and Pickle library was created and deployed at a local machine such that an individual would enter his details(attributes here) and his stroke vulnerability would be predicted. A snapshot is attached for the same.



Stroke Prediction

We need some information to predict your stroke risk status

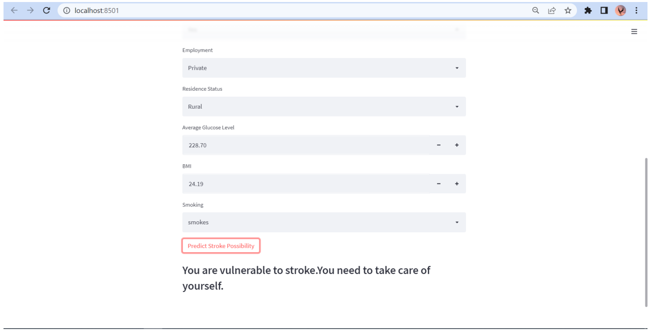
Age: 47.00

Gender: Male

Suffered from Hypertension: Yes

Suffered from Heart disease: Yes

Married: Yes



Employment: Private

Residence Status: Rural

Average Glucose Level: 228.70

BMI: 24.19

Smoking: smokes

You are vulnerable to stroke.You need to take care of yourself.

Contributions

The learning and planning was done as a team.The individual contributions are as given

- Deep Vasan (B20CS083): Random Forest Classifier, SVM Classifier, RandomizedSearchCV, Report, Model Deployment
- Tatvam (B20CS077): Data pre-processing, Data Visualization, Report, XGBoost, MLP Classifier, Hyperparameter tuning
- Rhythm Jain (B20EE094): Data pre-processing, KNN Classifier, Hyperparameter Tuning, Model Evaluation, Report

References

[1] KNN Algorithm | What is KNN Algorithm | How does KNN Function (analyticsvidhya.com)
[2] GridSearchCV or RandomSearchCV?. Comparing two sklearn hyperparameter… | by Brunna Torino | Towards Data Science
[3] SMOTE | Towards Data Science
[4] 14 Data Visualization Plots of Seaborn | by Aayush Ostwal | Towards Data Science
[5] Intuition behind ROC-AUC score. In Machine Learning, classification… | by Gaurav Dembla | Towards Data Science