PRINT ISSN 1119-8362
Electronic ISSN 2659-1502

Full-text Available Online at
https://www.ajol.info/index.php/jasem
http://www.bioline.org.br/ja

J. Appl. Sci. Environ. Manage.
Vol. 26 (6) 1031-1037 June 2022

Early Prediction of Cerebrovascular Disease using Boosting Machine Learning Algorithms to Assist Clinicians

[*1]ABDULLAHI, SD; [2]MUHAMMAD, SA

*[*1]Department of Mathematics and Computer Science, Federal University Kashere. Gombe State, Nigeria*
[2]Department of Computer Science, Faculty of Computing Federal University Dutse Jigawa State.
**Corresponding Author Email: daudasaniaa008@gmail.com, Other Author Email: msirajoa@fud.edu.ng*

ABSTRACT: Clinicians are required to make an early prediction of diseases to save a life, especially cerebrovascular diseases. The objective of this research is to use mathematical models such as boosting machine learning algorithms as a tool to be applied by clinicians for cerebrovascular disease. This paper particularly, considered XGBoost, AdaBoost, LightGBM, and CatBoost Classifiers to predict cerebrovascular disease using age, gender, BMI, hypertension, heart disease, residence type, ever married, smoking status, and average glucose level of the patients. Synthetic Minority Over-Sampling Technique Edited Nearest Neighbors Under-sampling (SMOTE-ENN) and Feature Engineering were applied to the dataset to enhance the performance of the algorithms. The result obtained showed that XGBoost Classifier is the best model with an accuracy of 98% and an AUC of 0.983.

DOI: https://dx.doi.org/10.4314/jasem.v26i6.6

Impact factor: http://sjifactor.com/passport.php?id=21082

Google Analytics: https://www.ajol.info/stats/bdf07303d34706088ffffbc8a92c9c1491b12470

Dates: Received: 24 February 2022; Revised: 30 May 2022; Accepted: 06 June 2022

Keywords: Cerebrovascular; Machine Learning Algorithms; Data preprocessing; Imbalanced data

Cerebrovascular Disease (Stroke) occurs when the blood stopped from flowing to the brain which causes the brain cells to die, within a short moment it can cause brain damage, long-term disability, and sometimes death (Aggarwal *et al.,* 2020; Kim *et al*., 2020). Machine learning is a sub-area of artificial intelligence that deals with learning from data, gaining popularity in clinical medicine because of its ability to infer meaningful knowledge from large datasets. Most of the clinical datasets are imbalanced as such the algorithms tend to be biased towards the majority class. The minority class of data is usually overlooked or oversampling of the minority class will create synthetic data, therefore, Boosting classifiers (Sridharan *et al*., 2021) may be used for the minority class. In the work of Nwosu *et al*. (2019), they perform an analysis of the stroke risk factors from patients' electronic health records to uncover the inter-dependence of the stroke risk factors, Principal Component analysis was employed, and showed that the risk factors are not highly correlated and as such the feature subspace cannot be reduced while reducing the feature subspace will help improve the model performance and also take less time to train the model and the used of statistical methods like chi-squared were used and reduced the feature space to six (6) which comprises age, heart_disease, average_glucose_level, hypertension, work_type, and ever_married with a performance accuracy of 97.6% using two-class Boosted Decision Tree as shown by (Ray *et al.,* 2020). In the report of Wu and Fang (2020). They developed machine learning models for predicting stroke with imbalanced data, random under-sampling Technique (RUS), Random over-sampling Technique (ROS), and Synthetic Minority over-sampling Technique (SMOTE) are the sampling technique used. Among them, SMOTE was able to give good balancing results as compared to the others with Random forest having an accuracy performance of 78% but sample among older Chinese. In the work of Alberto and Rodríguez (2021). The cross-industry standard process for data mining (CRISP-DM) is used as a guideline to develop stroke prediction using 5110 records with highly imbalanced data problems, SMOTE Technique, XGBoost, support vector machine, Neural Network, K-Nearest Neighbor, Naïve Bayes, Logistic Regression, Random Forest, and Decision Tree were used. The researchers concluded

**Corresponding Author Email: daudasaniaa008@gmail.com*

that Random Forest is the best model with an accuracy of 92% and fewer classification errors compared to the other algorithms, so also when hypertension and heart disease are present there is a high chance for the person to suffer from stroke. The objective of this research is to use mathematical models such as boosting machine learning algorithms as a tool to be applied by a clinician for the early prediction of cerebrovascular disease by considering XGBoost, AdaBoost, LightGBM, and CatBoost Classifiers to predict cerebrovascular disease using age, gender, BMI, hypertension, heart disease, residence type, ever married, smoking status, and average glucose level of the patients.

MATERIALS AND METHODS

The proposed architecture is presented in fig 1. Has a series of steps to achieve the research aim, it starts with importing the data that is gotten from the open-source Kaggle public dataset, importing it into the programming environment using python and performing data processing and sampling data, and lastly the model evaluation. Details of each step are given in figure 1.

Data Description: The dataset used for the experiment is gotten from Kaggle's available datasets, it consists of 10 risk factors (dependent variables) where 2 are non-modifiable and 8 are modifiable, and one Target variable where 1 signifies the presence of stroke and 0 signifies the absence of Stroke. The descriptions of the variables and descriptive statistics of the numerical variables are given in Table I and Fig. 2 respectively.

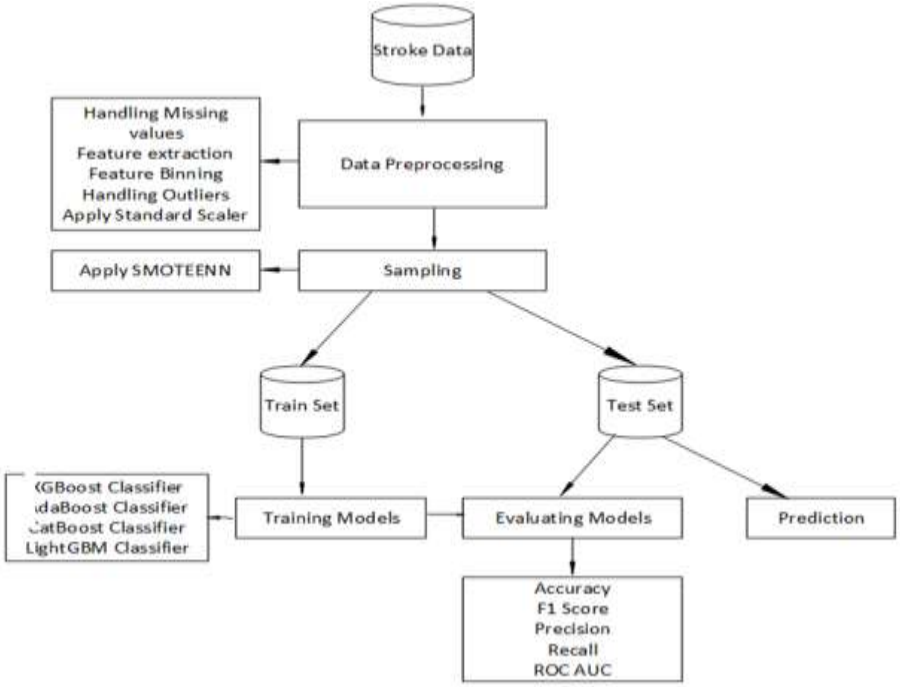

Fig. 1: The architecture of the proposed Cerebrovascular disease prediction

Table 1: Data Description

S/N	Variable Name	Description
1	Id	INTEGER, the unique id of patience
2	Gender	Object, [‘male’, ‘female’, ‘other’]
3	Age	Float, in years/months
4	hypertension	Integer, [0, 1] 0 means ‘no hypertension’ 1 signifies ‘hypertension’
5	heart_disease	Integer, [0,1] 0 means ‘no heart disease’ 1 signifies ‘heart disease’
6	ever_married	Object, [‘Yes’, ‘No’]
7	work_type	Object, [‘private’, ‘Self-employed’, ‘children’, ‘Govt_job’, ‘Never_worked’]
8	Residence_type	Object, [‘Rural’, ‘Urban’]
9	Avg_glucose_level	Float
10	Bmi	Float
11	smoking_status	Object, [‘Smoked’, ‘Never_smoked’, ‘formerly smoked’, ‘Unknown’]
12	Stroke	Integer, [0,1]

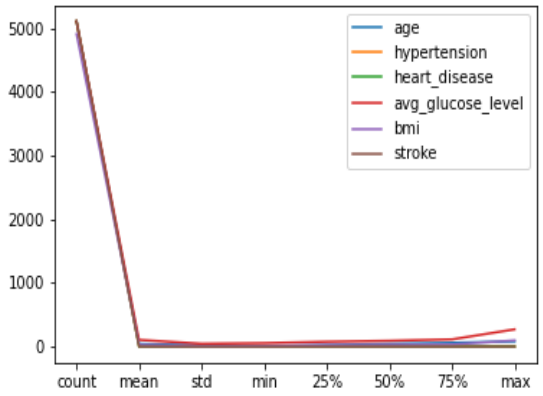

Fig 2: Descriptive Statistics of Numerical Variables.

Data Preprocessing: Machine Learning algorithms work based on data and there is the need to preprocess the data before training and evaluating the models, data processing helps in improving the performance of the models. The following steps are considered during the preprocessing phase.

i. We used KNNImputer to handle the Missing Values in bmi.
ii. smoking_status variable has a value of 'Unknown' which can signify a null value, we used the statistical mode of the variable to replace the 'Unknown'.
iii. Converting categorical features into numerical values using LabelEncoder for ordinal features like ever_married, Residence_type, and converting nominal categories into numerical values using OneHotEncoder like work_type and smoking_status.
iv. Discretization of age, bmi, and avg_glucose_level to create other categorical variables, this helps create more meaningful features and can improve the performance of the models, created age_cat with values [children, teen, adults, mid-adults, elderly], created bmi_cat with values [underweight, ideal, overweight, obesity], and created avg_glucose_level_cat with values [Very Low, Low, Normal, High, Very High].
v. Handling outliers in bmi and avg_glucose_level, we used the interquartile range method to remove outliers, which help improve the performance of the prediction.
vi. Standard Scaler is applied to the data set to transform the data on the same scale of measurement.
vii. Handling imbalanced data: imbalanced data is a scenario where the distribution of minority class '1' is very low compared to the majority class '0'. The dataset under consideration is highly imbalanced. In this research, we used a hybrid Technique called Synthetic Minority Over-Sampling Technique Edited Nearest Neighbors Under-sampling (SMOTE-ENN). It comprises under-sampling and over-sampling Techniques. The Synthetic Minority Over-Sampling Technique (SMOTE) is an over-sampling Technique that creates synthetic data and balances the distribution while the Edited Nearest Neighbor (ENN) is an under-sampling technique that performs the task of removing misclassified instances from both minority and majority classes (Lamari *et al.*, 2021). Fig. 3 and Fig. 4 show the distribution of the target class before and after applying the sampling Technique.

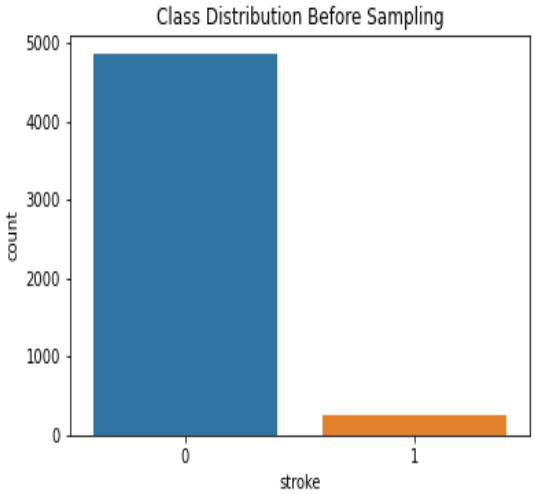

Fig. 3: Class Distribution Before Sampling

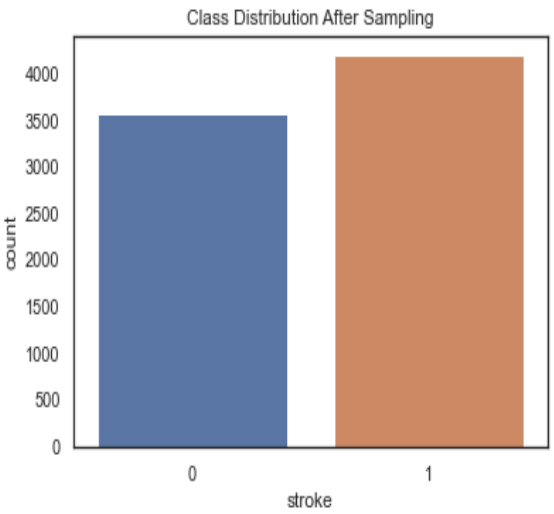

Fig. 4: Class Distribution After Applying SMOTEENN

viii. Splitting the data: after sampling the data, the dataset is split into 80% training set and 20% testing set.

Boosting Algorithms: Boosting algorithms are among the most useful and powerful techniques in predictive modeling, recently they outperform other algorithms both in machine learning tasks and Kaggle

competitions on structured data. In this paper four (4) of these algorithms are used.

i. *AdaBoost:* Adaboost proposed by Freund and Robert Schapire in 1996 is an iterative ensemble method that combines multiple weak learners to get a single strong learner sequentially. It does so by creating a decision stump. Decision stumps are decision trees with a single split. (Hu *et al.,* 2008). More weights are given to incorrectly classified samples and fewer weights to correctly classified samples, the weights are updated iteratively until the data points are correctly classified (Schapire, 2003). Below is the pseudocode of the AdaBoost algorithm.

1. Consider a Training set

2. Initialize weights and normalize the weight $\boldsymbol{D} = (\boldsymbol{x_1}, \boldsymbol{y_1}), \ldots, (\boldsymbol{x_n}, \boldsymbol{y_n}), \ldots, \boldsymbol{y} \in \{-\boldsymbol{1}, +\boldsymbol{1}\}$,

3. Repeat from $\boldsymbol{t} = \boldsymbol{1} \ldots, \boldsymbol{T}$, Executing the following steps

1) Train the Training set with the distribution D_t

2) Get the base classifier h_t which results in the least error

3) Update the weight by focusing on the incorrect sample and set the new weight

4. Output the final Strong Classifier *H*

ii. *XGBoost:* XGBoost (for "extreme gradient Boosting"), is a Gradient Boosting Decision Tree (GBDT) algorithm, it is fast, flexible, and versatile. It supports Distributed computing, parallelization of tree construction, and GPU support. It differs from other gradient boosting with the introduction of a new regularization technique to prevent overfitting. The regularization Technique to be minimized is given in Eq (1)

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \qquad 1$$

$$where\ \Omega(f_k) = \gamma T + \frac{1}{2}\lambda||\omega||^2$$

T is the number of branches in the tree, γ is a user-definable penalty, meant to prune branches, ω is the value of each leaf, Ω is the regularization function, l is a loss function that measures the residuals, that is the difference between the prediction $\hat{y}_i$ and the true value y_i.

The XGBoost algorithm runs by updating the residual of the weak learners to get an optimal model. It supports stochastic gradient boosting with subsampling at the row, column, and column per split level which increases the computational speed of the parallel algorithm (Chen and Guestrin 2016)

iii. LightGBM: LightGBM also known as Light Gradient Boosting Machine developed by a team from Microsoft in 2017. Is a GBDT algorithm that provides faster training speed, lower memory usage, it also supports parallel, distributed, and GPU learning. It uses a histogram-based algorithm in tree splitting thus reduce training time. It grows its tree leaf-wise which achieves lower loss compared to a level-wise technique in other algorithms like XGBoost. Its subsample the data instance using Gradient-based One-side Sampling (Goss). When sampling, Goss performs random sampling on instances with small gradients and keeps those instances with large gradients (Ke *et al.,* 2017).

iv. CatBoost: CatBoost (for "Categorical Boosting") is a GBDT algorithm designed by Dorogush *et al.* (2018). CatBoost not only supports categorical features but also numerical features, it has fast GPU and CPU support, and it deals with categorical features during training time whereas other GBDT uses one-hot encoding to convert categorical features into numbers during preprocessing phase. It introduced a new schema that reduced overfitting when performing tree splitting, it does that by performing a random permutation of the dataset and computing the average value of *y* for each instance with the same category. Mathematically given in Eq (2);

Let $\sigma = (\sigma, \ldots, \sigma_n)$ be the permutation, then $x_{\sigma_p,k}$ is substituted with

$$\frac{\sum_{j=1}^{p-1}\left[x_{\sigma_j,k} = x_{\sigma_p,k}\right]Y_{\sigma_j} + a.P}{\sum_{j=1}^{p-1}\left[x_{\sigma_j,k} = x_{\sigma_p,k}\right] + \boldsymbol{a}} \qquad 2$$

Where p is the prior and a which is $a > 0$ is the weight of the prior (Dorogush *et al.*, 2018; Wang and Cheng, 2021).

Evaluation Metrics: Considering the dataset is imbalanced, the accuracy metric alone is not the right to be used for imbalanced data Classification. The following evaluation metrics are used,

i. Confusion Matrix: it is a performance metric for machine learning classification problems in form of a table showing combinations of predicted and actual values.

ii. Accuracy: Accuracy measures how often a classifier correctly predicts. It is the ratio of the number of correct predictions to the total number of predictions (Jason, 2020). Mathematically defined as;

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad 3$$

iii. Precision: Precision explains how many of the correctly predicted cases turned out to be positive, in other words, Precision explains how precise our

classifier is, it is useful where False Positive (Type I Error) is a higher concern (Jason, 2020; *Sun et al.*, 2009). It is mathematically defined as;

$$Precision = \frac{TP}{TP+FP} \quad 4$$

iv. Recall: Recall Explain how many of the actual positive cases we were able to predict correctly with our model. it is useful where False Negative (Type II Error) is a higher concern (Jason, 2020; *Sun et al.*, 2009). It is mathematically defined as;

$$Recall = \frac{TP}{TP+FN} \quad 5$$

v. F1-Score: it gives a combined idea about precision and recall metrics, it is useful when both classes are of concern, mathematically, F1-Score is the harmonic mean of precision and recall (Sun *et al.*, 2009).

$$f1\_Score = 2\frac{precision*Recall}{precision+Recall} \quad 6$$

vi. AUC-ROC: Area Under the Curve (AUC) of Receiver Operating Characteristic (ROC) it is the area under the probability curve that plots True Positive Rate (TPR) against False Positive Rate (FPR) at various threshold values (Sun *et al.*, 2009).

RESULTS DISCUSSION

Data Analysis: The stroke dataset under consideration comprises eight (8) modifiable risk factors and two (2) non-modifiable risk factors, exploring the data through different analysis techniques helps us uncover patterns in the data and assist in creating more risk factors. Fig. 5 shows the frequency distributions of age, bmi, and average glucose level, there is a significant sign of stroke in elderly people, so also a higher bmi indicates a higher chance of getting a stroke. Fig. 7 shows the relationship between bmi and average glucose level, it is confirmed that people with less than 150 glucose levels are less prone to strokes than people with glucose levels more than 150 levels, people with bmi greater than 40 have low average glucose levels. Fig. 8 is a heatmap displaying the correlation coefficients of the features to detect multicollinearity, only age and ever_married have shown a slightly higher positive correlation with an r-value of 0.68. Fig. 6 shows the relationship between age and average glucose level whereas age increase leads to an increase in glucose level and is prone to stroke. *Performance Analysis* In this work, XGBoost, LightGBM, CatBoost, and AdaBoost classifiers are used for stroke prediction using the open-source Kaggle dataset. Table 2 shows the performance results of the models. The model developed with XGBoost happened to be the best in terms of accuracy, f1-score, precision, and recall as presented in Table II with an accuracy of 98.23%, followed by LightGBM, CatBoost, and AdaBoost with an accuracy of 98.07%, 97.43%, 93.96% respectively. Fig. 9 shows the confusion matrix of the best performing model, that is XGBoost model developed with fewer Type I and Type II errors, out of 1556 instances (Testing set) the model classified 716 patients with No stroke correctly and 813 patients with Stroke correctly while failed by falsely classifying 13 patients as having a stroke and 14 patients as No stroke.

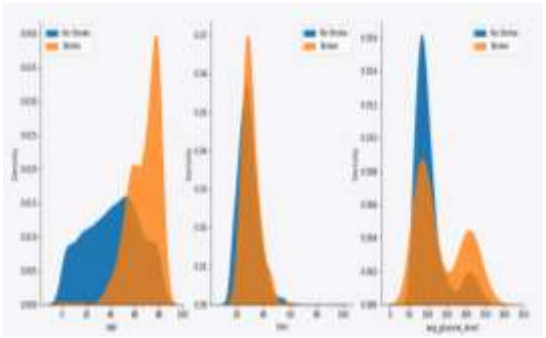

Fig. 5: Kinetic Density Estimation (KDE) of Age, bmi, and average glucose level.

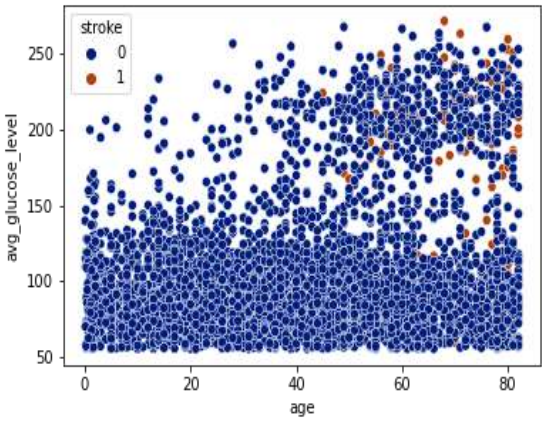

Fig. 6: Bivariate analysis of age and average glucose level

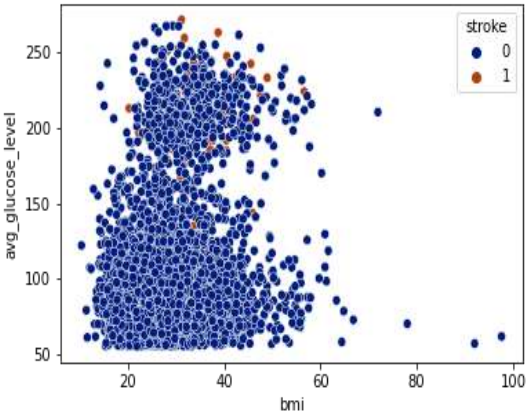

Fig. 7: Bivariate analysis of bmi and average glucose level

Table 2: Result Of The Boosting Classifiers

Model	Accuracy	F1-Score	Precision	Recall	AUC-ROC
XGBoost	98.265%	0.984	0.984	0.983	0.983
LightGBM	98.072%	0.982	0.980	0.984	0.979
CatBoost	97.429%	0.976	0.970	0.982	0.974
AdaBoost	93.959%	0.944	0.936	0.952	0.938

Table 3: Performance Comparison Of Stroke Prediction Models Using The Same Dataset

Author & Year	Title	Methods	Performance	
			Accuracy	AUC
Alberto & Rodríguez, 2021	Stroke prediction through Data Science and Machine learning algorithms	XGBoost, RF, SVM, ANN, KNN, LR, DT	RF - 92%	RF – 0.975
Sailasya & Kumari, 2021	Analyzing the Performance of Stroke Prediction using ML Classification Algorithms	RF, SVM, KNN, LR, DT, and NB	NB - 82%	Nill
Emon *et al*, 2020	Performance analysis of Machine learning approaches in stroke prediction	LR, SGD, DT, AdaBoost, DA, MLP, KNN, Voting Classifier, GBM, XGBoost	Voting Classifier - 97%	Nill
Proposed work	**An improved Stroke prediction Using Boosting Machine Learning Algorithms**	**AdaBoost, XGBoost, LightGBM, CatBoost**	**XGBoost – 98%**	**XGBoost – 0.983**

Legend: ANN – Artificial Neural Network, DT – Decision Tree, DA – Discriminant Analysis, KNN - K-Nearest Neighbor, NB - Naïve Bayes, RF – Random Forest, SVM – Support Vector Machine, SGD – Stochastic Gradient Descent, LR – Logistic Regression, GBM - Gradient Boosting Machine, Multilayer Perceptron

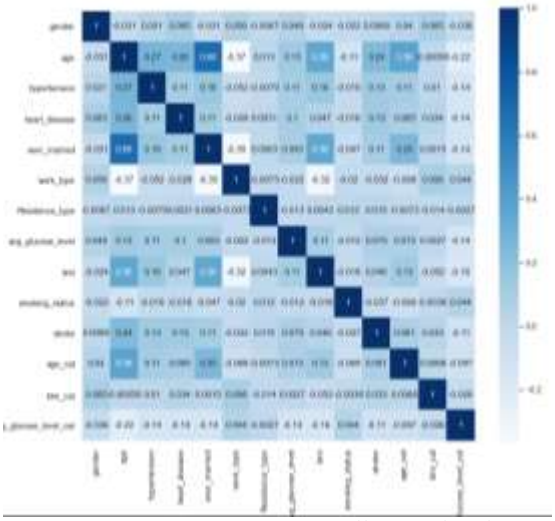

Fig. 8: Heatmap showing correlation coefficients of the features.

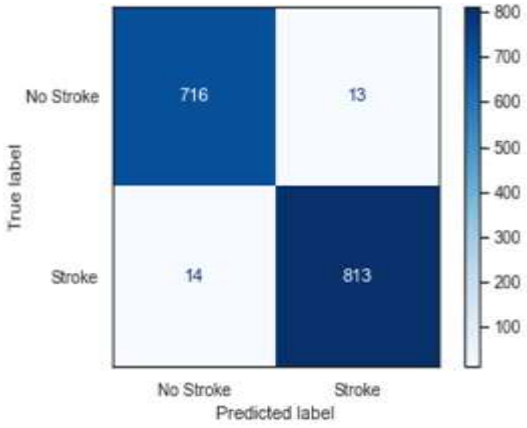

Fig. 9: Confusion Matrix of XGBOOST Classifier

Fig. 10 shows the Receiver Operating characteristic (ROC). The ROC is a probability curve that plots the true positive rate (TPR) against the false positive rate (FPR) at various thresholds, the area under the curve (AUC) measures the classifiers' ability to distinguish between classes. AUC has values ranges from 0 to 1, when AUC is 1, the model perfectly distinguishes between positive and negative classes. When AUC is 0, the model will predict negative as positive and vice versa. When AUC is 0.5, the model will fail to distinguish between classes. XGBoost has an AUC of 0.999 which signifies the model distinguishes between positive and negative classes 99% of the time.

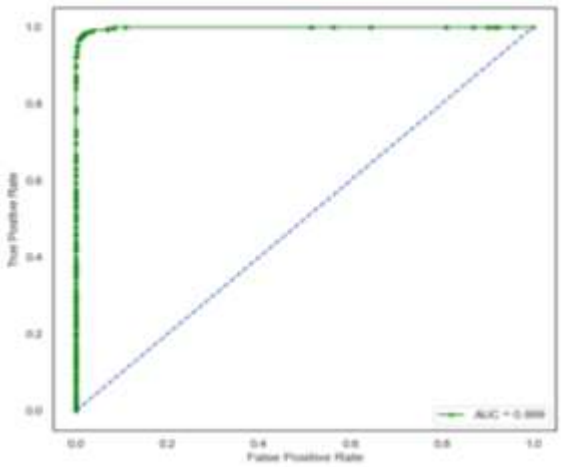

Fig. 10: Receiver operating characteristic (ROC) Curve.

Conclusion: In conclusion, Cerebrovascular disease is the second leading cause of death globally behind

heart disease, machine learning has proven impact in predicting stroke as early as possible to prevent it from causing damage. Our method considered age, bmi, average glucose level, residence type, gender, hypertension, heart disease, smoking status, and work type of an individual to predict stroke, among methods XGBoost was the best. This work used a highly imbalanced dataset, we recommend using a balanced dataset which will help in more accurate building models.

REFERENCES

Aggarwal, G; Lippi, G; Michael HB. (2020). Cerebrovascular disease is associated with an increased disease severity in patients with Coronavirus Disease 2019 (COVID-19): A pooled analysis of published literature. *Intl. J. of Strk*, *15*(4), 385–389.

Alberto, J; Rodríguez, T. (2021). Stroke prediction through Data Science and Machine Learning Algorithms. *Ml*.

Chen, T; Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the ACM SIGKDD Int. Conf. on Knowl. Disc. and Data Mining*, *13-17-August-2016*, 785–794.

Dorogush, AV; Ershov, V; Gulin, A. (2018). CatBoost: gradient boosting with categorical features support. 1–7. http://arxiv.org/abs/1810.11363

Emon, MU; Keya, MS; Meghla, TI; Rahman, MM; Mamun, MSA; Kaiser, MS. (2020). Performance Analysis of Machine Learning Approaches in Stroke Prediction. *Proc. of the 4th Intl. Conf. on Electro., Comm. and Aero. Technol., ICECA 2020*, 1464–1469.

Freund, Y; Schapire, RE; Hill, M. (1996). Experiments with a New Boosting Algorithm Rooms *f 2B-428 , 2A-424 g*.

Hu, W; Member, S; Hu, W; Maybank, S. (2008). AdaBoost-Based Algorithm for Network. *Ieee Transac. on Sys., Man, and Cybernetics*, *38*(2), 577–583.

Jason, B. (2020). Tour of Evaluation Metrics for Imbalanced Classification. Machine Learning Mastery. https://machinelearningmastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/

Ke, G; Meng, Q; Finley, T; Wang, T; Chen, W; Ma, W; Ye, Q; Liu, T. (2017). LightGBM : A Highly Efficient Gradient Boosting Decision Tree. *Nips*, 1–9.

Kim, J; Thayabaranathan, T; Donnan, GA; Howard, G; Howard, VJ; Rothwell, PM; Feigin, V; Norrving, B; Owolabi, M; Pandian, J; Liu, L; Cadilhac, DA; Thrift, AG. (2020). Global Stroke Statistics 2019. *Inter. J. of Strk*, *15*(8), 819–838.

Lamari, M; Azizi, N; Hammami, NE; Boukhamla, A; Cheriguene, S; Dendani, N; Benzebouchi, NE. (2021). SMOTE--ENN-Based Data Sampling and Improved Dynamic Ensemble Selection for Imbalanced Medical Data. *Advan. on Smart and Soft. Comput.* (pp. 37–49).

Nwosu, CS; Dev, S; Bhardwaj, P; Veeravalli, B; John, D. (2019). Predicting Stroke from Electronic Health Records. *Proc. of the Annu. Intl. Conf. of the IEEE Eng. in Med. and Bio. Soc., EMBS*, 5704–5707.

Ray, S; Alshouiliy, K; Roy, A; Alghamdi, A; Agrawal, DP. (2020). Chi-Squared Based Feature Selection for Stroke Prediction using AzureML. (2020) *Intermountain Eng. Tech. and Comp., IETC 2020*.

Sailasya, G; Kumari, GLA. (2021). Analyzing the Performance of Stroke Prediction using ML Classification Algorithms. *Intl. J. of Advanc. Comp. Sci. and App. 12*(6), 539–545.

Schapire, RE. (2003). The Boosting Approach to Machine Learning: An Overview. 149–171.

Sridharan, M; Mantyla, M; Rantala, L; Claes, M. (2021). Data balancing improves self-admitted technical debt detection. *18th Intl. Conf. on Mining Soft. Repo., MSR 2021*, 358–368.

Sun, Y; Wong, AKC; Kamel, MS. (2009). Classification of imbalanced data: A review. *Intl. J. of Patn. Recog. and AI. 23*(4), 687–719.

Wahab, K. W. (2008). The burden of stroke in Nigeria. *Intl. J. of Strk*, *3*(4), 290–292.

Wang, H; Cheng, L. (2021). CatBoost model with synthetic features in application to loan risk assessment of small businesses. http://arxiv.org/abs/2106.07954

Wu, Y; Fang, Y. (2020). Stroke prediction with machine learning methods among older chinese. *Intl. J. of Env. Res. and Pub. Hlth*, *17*(6), 1–11.