Analysing an imbalanced stroke prediction dataset using machine learning techniques

Analysing an Imbalanced Stroke Prediction Dataset Using Machine Learning Techniques

Viswapriya Subramaniyam Elangovan[1] ,Rajeswari Devarajan[1] , Osamah I. Khalaf[2], Mhd Saeed Sharif[3] and Wael Elmedany[4]

[1]Department of Data Science and Business Systems, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chennai-603203, Tamilnadu, India
[2]Department of Solar, Al-Nahrain Research Center for Renewable Energy, Al-Nahrain University, Jadriya, Baghdad, Iraq
[3]Intelligent Technologies Research Group, Computer Science and DT, ACE, UEL University Way, London, UK.
[4]College of Information Technology | University of Bahrain, Bahrain

ABSTRACT

A stroke is a medical condition characterized by the rupture of blood vessels within the brain which can lead to brain damage. Various symptoms may be exhibited when the brain's supply of blood and essential nutrients is disrupted. To forecast the possibility of brain stroke occurring at an early stage using Machine Learning (ML) and Deep Learning (DL) is the main objective of this study. Timely detection of the various warning signs of a stroke can significantly reduce its severity. This paper performed a comprehensive analysis of features to enhance stroke prediction effectiveness. A reliable dataset for stroke prediction is taken from the Kaggle website to gauge the effectiveness of the proposed algorithm. The dataset has a class imbalance problem which means the total number of negative samples is higher than the total number of positive samples. The results are reported based on a balanced dataset created using oversampling techniques. The proposed work used Smote and Adasyn to handle imbalanced problem for better evaluation metrics. Additionally, the hybrid Neural Network and Random Forest (NN-RF) utilizing the balanced dataset by Adasyn oversampling achieves the highest F1-score of 75% compared to the original unbalanced dataset and other benchmarking algorithms. The proposed algorithm with balanced data utilizing hybrid NN-RF achieves an accuracy of 84%. Advanced ML techniques coupled with thorough data analysis enhance stroke prediction. This study underscores the significance of data-driven methodologies, resulting in improved accuracy and comprehension of stroke risk factors. Applying these methodologies to medical fields can enhance patient care and public health outcomes. By integrating our discoveries, we can enhance the efficiency and effectiveness of the public health system.

Keywords: Machine learning; neural network; random forest; stroke prediction; imbalanced data

1. Introduction

A stroke ranks among the top causes of mortality and represents a severe global public health threat. As proposed in [1]. An acute stroke is broken blood vessels that lead to causes of hemiplegia, impairment, and unawareness. A stroke can occur at any time. The two common types of strokes are acute ischemic stroke and haemorrhagic stroke. If there is a reduction or cessation of blood flow to brain cells, it results in the death of these cells within a matter of minutes, ultimately leading to fatality. The prevalence and mortality of stroke continue to increase [2]. Timely identification and prevention of a stroke have become crucial to mitigate its negative outcomes. Symptoms of stroke disease may vary, and it can develop either slowly or quickly. It can be detected through face swelling, arm raising test, the way the person speaks, and the time he takes to respond, if these factors lead to abnormalities, in this scenario, it is advisable for the person to seek immediate medical attention at a hospital. The majority of the people who have experienced a stroke come under the age of 70. Six and a half million people die from stroke annually [3]. Therefore, predicting this fatal disease as soon as possible is extremely important.

Analysing an imbalanced stroke prediction dataset using machine learning techniques

The field of medical sciences has witnessed remarkable enhancements owing to the progression of technological innovations over the years. Significantly, the Internet of Things (IoT) has simplified the collection of healthcare-related data through the accessibility of affordable wearable devices [4]. Many unprocessed medical data are derived from these devices to uncover informative patterns through diverse ML techniques. Furthermore, the insights acquired are subsequently employed for decision-making within the healthcare sector, demonstrating their effectiveness as cost-saving factors [5]. ML models can be effectively employed on Electronic Health Records (EHR) to proficiently forecast the likelihood of a stroke occurrence for each patient. It is based on the features of the patient's records. Patients' EHRs encompass, gather, and store numerous aspects of their medical conditions. The significant risk factors of stroke are age, hypertension, high blood pressure, smoking, and presence of heart disease, obesity, and lifestyle. However, all the accumulated features in EHRs could potentially contribute to stroke detection.

In recent years, the increasing intricacy of dimensionality and the skewed distribution of samples has posed noteworthy challenges to machine learning. The minority class has the least number of samples, whereas the majority class has more samples in case of unbalanced data [6]. The performance of the classification algorithm will be biased with majority classes [7]. Numerous discussions are conducted at various levels to overcome this constraint, including algorithmic approaches, cost-sensitive techniques, and data-level interventions. At the algorithmic level, the classification models are dynamic to improve the learning method on minority classes by assigning limits separately for both classes. Nevertheless, this approach remains fixed once put into action due to its classifier-specific nature. Cost-sensitive models address class imbalanced challenges by assigning different classification costs to different classes. Aiming to optimize misclassification costs results in an overly high price for identifying samples. Data-level techniques can be seen as data preparation strategies employing diverse sampling methods to achieve class balance within training data. This approach remains highly adaptable and independent of any specific classifier, fulfilling the fundamental criteria of traditional classifiers [8]. Data level methods are categorized as under-sampling, oversampling, and hybrid sampling techniques. Under-sampling involves the removal of certain samples from the majority class, reducing time costs. However, it comes with the drawback of potentially losing information. Random over-sampling will replicate the existing minority class samples to match them with the majority class count. This duplication of samples, however, can potentially lead to issues of overfitting. Recognizing the overfitting challenge stemming from random over-sampling and aiming to maintain dataset balance, the synthetic minority oversampling technique SMOTE was introduced [9].

Numerous studies have made advancements and attained enhanced classification outcomes through the utilization of SMOTE. However, within the SMOTE technique, the incorporation of specific noisy samples during the creation of new instances can yield irrational results. As a result, this can compromise the overall classification performance of the classifier. To address this concern, Adasyn [10] arose as a technique that creates new minority samples, positioning them near the original samples that were misclassified based on the k-nearest neighbor classifier. The hybrid strategy merges the initial two models to eliminate negative class samples and duplicate positive class samples, aiming to mitigate overfitting. This hybrid data-level technique is used to address datasets with diverse distributions. However, it comes with a significant computational burden and is not suitable for multi-classification models. Oversampling predominantly encompasses techniques rooted in structure preservation and interpolation. The classification effect of minority samples has been improved by these effective approaches [11]. The oversampling technique has been utilized in this study to handle the unbalanced data.

The primary contributions of this paper encompass the following – (a) the comprehensive insight of various features for stroke prediction is provided, (b) the oversampling technique has been used for solving the imbalanced problem, (c) the classification algorithms were validated on the unbalanced dataset and also on the balanced dataset to check the impact of the oversampling technique.

Presented below is the structure of the rest of the paper. In section 2, the related study discusses how the proposed system is different from the existing system. In section 3, the detailed description of the proposed system and data sets used. It also includes

an elaborate description of the operating model. In section 4, the findings from the experiment and subsequent analysis have also been discussed neatly with the proposed algorithm and clear explanation. In section 5, there is a discussion about the conclusion, perspectives, and other future work that should be done.

2. Related work

There is an increase in the number of researches to predict stroke disease through ML techniques. As proposed by [12], a stroke prediction model has been developed using a Deep Neural Network with ant-lion optimization algorithm based on an imbalanced dataset. The imbalanced issue has been solved by random oversampling. They compared the proposed classifier algorithm including the oversampling technique with other existing ML algorithms and found the proposed model gave a better accuracy of 99.5%.

According to [13], suggested a stroke prediction model formulated on Auto HPO. In this paper, the missing values and inaccurate values are handled through statistical and non-parametric methods. Auto HPO was used to compute the hyperparameter optimization to select instances randomly from the majority samples to minimize the imbalance ratio. A deep neural network has classified with an accuracy of 71.6%.

A recent study revealed that the accuracy of a stroke prediction model can be improved by using major risk factors that were identified from the dataset and classification was done with a perceptron neural network [14]. They used an under-sampling technique to handle an imbalanced dataset. Among several machine learning classifiers, they concluded neural networks achieved a better accuracy of 78% with only limited sets of features.

As suggested by [15], the stacking algorithm outperforms other classifiers for the stroke prediction model and the most relevant features have been identified based on the ranking method. The stacking method achieved the best accuracy of 98% According to [16], a stroke prediction model has been developed based on EHR data and they reduced the feature space by using principle component analysis and dimensional reduction. Since the dataset is imbalanced, it has been solved by using a random under-sampling technique. Moreover, among several classification algorithms such as Decision Tree (DT), and Random Forest (RF), the multi-layer perceptron model achieved a better accuracy of 75.02%

Another study discussed stroke detection using rough set theory to identify the most relevant features of stroke disease and random downsampling has been used to solve imbalanced data [17]. The proposed rough set feature selection technique was compared with other feature selection techniques and achieved a higher correlation value of 0.675. According to [18]. The artificial neural network with a stochastic gradient descent algorithm outperformed the various existing ML algorithms for stroke prediction with a dataset collected from Sugam Hospital, Kumbakonam has achieved a better accuracy of 95%

Another study [19] analyzed machine learning algorithms for stroke prediction such as Logistic Regression (LR), RF, naïve Bayes, and support vector machine using a stroke prediction dataset. They concluded RF outperformed other algorithms with Auc. of 0.81 with under sampling to solve imbalanced data.

According to [20], EHR is collected from the Medical Information Mart for Intensive Care (MIMIC-III). After data preprocessing, they used LR and support vector machine for classification purposes. They also included an optimization process with these ML algorithms. They found support vector machine outperformed other algorithms.

As per the study mentioned in [21], authors used a cyberbullying dataset and four resampling techniques such as random oversampling, under sampling, smote, and hybrid smote tomek. They found accuracy had not much been improved only with resampling techniques. Somehow, smote tomek enhanced recall values. The support vector classifier outperformed other classifier models on all metrics with a margin of 0.99.

A recent study [22] focussed on the importance of drug mechanisms using machine learning classifiers to predict effective drug combinations by using real data and shown enhanced results. Here the dataset is imbalanced, so they used random under-sampling. The different classification algorithms such as naïve Bayes, RF, knn, and LR with an average accuracy of 89%

As given by [23], the classification of monkeypox skin lesions can be done by the convolutional neural network. It achieved a higher accuracy of 95% and also this model has been optimized by the grey optimization algorithm.

As per the article [24], the prediction of a heart attack on the MIMIC-III dataset using several machine learning algorithms gives better accuracy for RF. However, they have used a novel balanced technique named under sampling-clustering-oversampling with RF outperforms other ML classifiers with 75% accuracy. They showed gradual improvement between the original and balanced datasets.

The outcomes derived from the diverse techniques illustrate that various aspects can influence the findings of the effectiveness of the prediction model. These diverse aspects encompass chosen features, data cleaning procedures, handling of null values, data variability, and data normalization. Hence it is essential for the analysts, how these aspects using the EHR dataset are analyzed and how they affect the efficiency of the final stroke prediction model.

The existing research on stroke disease prediction models still needs to be improved by using proper sampling techniques for handling imbalanced datasets. Research in related domains that identify an imbalance ratio in the dataset affects the performance of the ML framework. Using the oversampling technique to get balanced data instead of using downsampling is important since it might lead to the loss of essential information to develop the prediction model. Hence, it is more necessary for ML practitioners in the medical field to handle class imbalance problems by using the oversampling technique. This will lead to a higher F1 score without any loss of data. If the input dataset is highly imbalanced, it is better to focus on the F1-score value than accuracy since it is biased with the majority samples.

3. Materials and methods

3.1. Dataset description

The records encompass vital signs, diagnoses, and medical examination outcomes of a patient. Further, the medical diagnosis appears favorable as EHR are optimally utilized. According to the recorded statistics, the utilization of EHR in US hospitals surged from 12.5% to 75.5% between 2009 and 2014 [25]. The stroke prediction can be done by using signals, scan images, or simply with health records [26]. The used EHR dataset is available from Kaggle, an open dataset. The dataset comprised a total of 43400 samples that consisted of 11 inbound features and 1 outcome feature. The 11 input features are identifier, gender, age, hypertension, heart disease, marital status, kind of occupation, residence area, average glucose level, Body Mass Index and smoking status. Stroke and non-stroke cases are distinguished and represented as binary classes for the output feature. [27].The personal information of the patient can be identified by using two attributes namely age and gender. The most important clinical records are available in the remaining 8 attributes. The Kaggle dataset sources are shown in **Table 1**. The dataset description is shown in Table 2. Since the dataset is unbalanced, there are 1.8% instances of stroke presence and 98.1% of absence of stroke. There are 7 categorical attributes and 5 numerical attributes. In this dataset 783 have the presence of stroke while 42617 have the absence of stroke. The patient ID has been excluded for analysis and study from the features [28].

3.2. Exploratory data analysis

Table 2 consists of various risks that are mentioned with the encoded values. The analysis of features such as age, gender, work type, and stroke to find the number of occurrences under two different categories that lead to a person having a stroke or not has been discussed in this section. From the dataset, the number of people who had a stroke is 783 and who did not have a stroke is 42617 as shown in **Figure 1**.

Table 1. Dataset Description

Description	Kaggle Dataset
Total Records	43400
No of Features	11
No of Classes	2
% of Present records	1.80
% of Negative records	98.1

Analysing an imbalanced stroke prediction dataset using machine learning techniques

Table 2. Exploratory data analysis

	Attribute Name	Description	Range of Values
1	Gender	Gender of the person [1: Male, 0: Female]	0, 1
2	Age	Age of the person in years	20-80
3	Hypertension	No hypertension-0 Suffering hypertension-1	0, 1
4	Heart disease	No heart disease-0 Suffering heart disease-1	0, 1
5	Ever married	Not married -0 Married -1	0, 1
6	Work-type	Children, private, never worked, govt job, self-employed	1,2,3,4,5
7	Residence area	Rural or urban	0, 1
8	Avg-Glucose	Average Glucose Level in numerical	55-280
9	BMI	Body mass index in numerical	10.1 – 98.7
10	Smoking-status	Never smoked formerly smoked	0, 1
11	Stroke status	No Stroke – 0, Stroke – 1	0, 1

This shows the dataset is highly imbalanced. Since the number of observations of one class is significantly lesser than the number of observations of another class, with this imbalanced dataset, the predictive model could be biased and inaccurate. So, this should be balanced by using the oversampling method. The dataset consists of categorical and numerical features. The class distribution of categorical features is shown in **Figure 2**. The blue represents no stroke class and the orange represents the stroke class

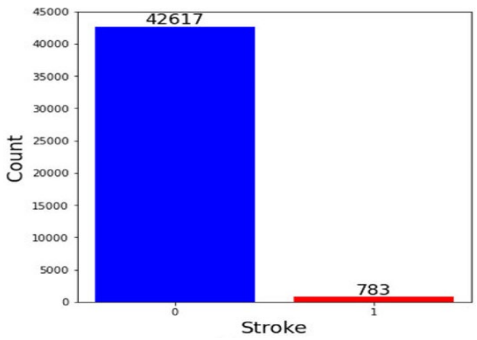
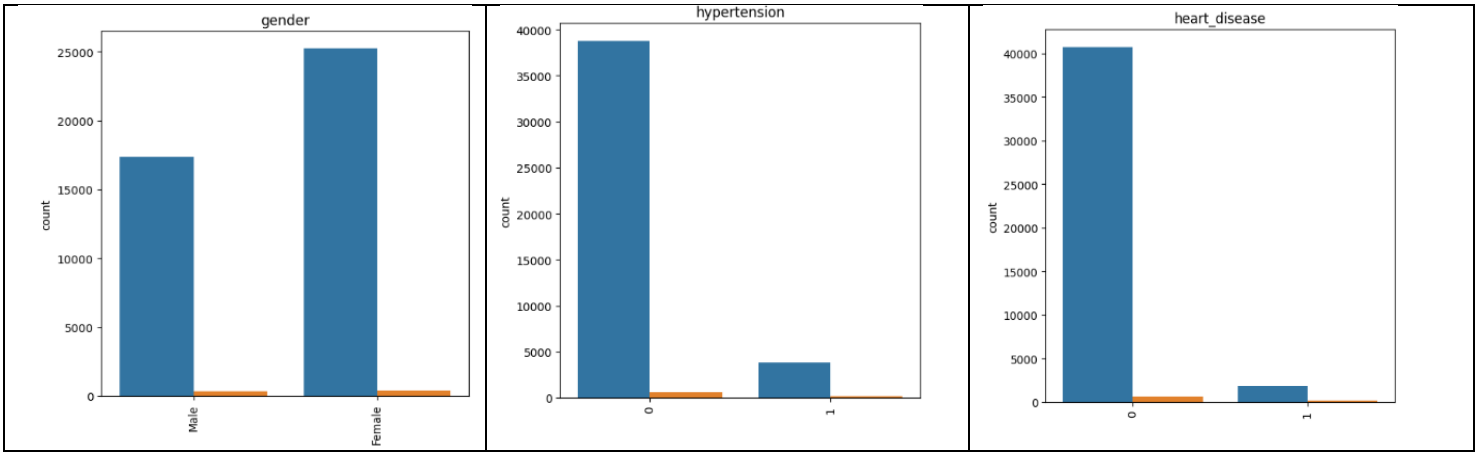


Fig. 1. Class distribution of stroke



Analysing an imbalanced stroke prediction dataset using machine learning techniques

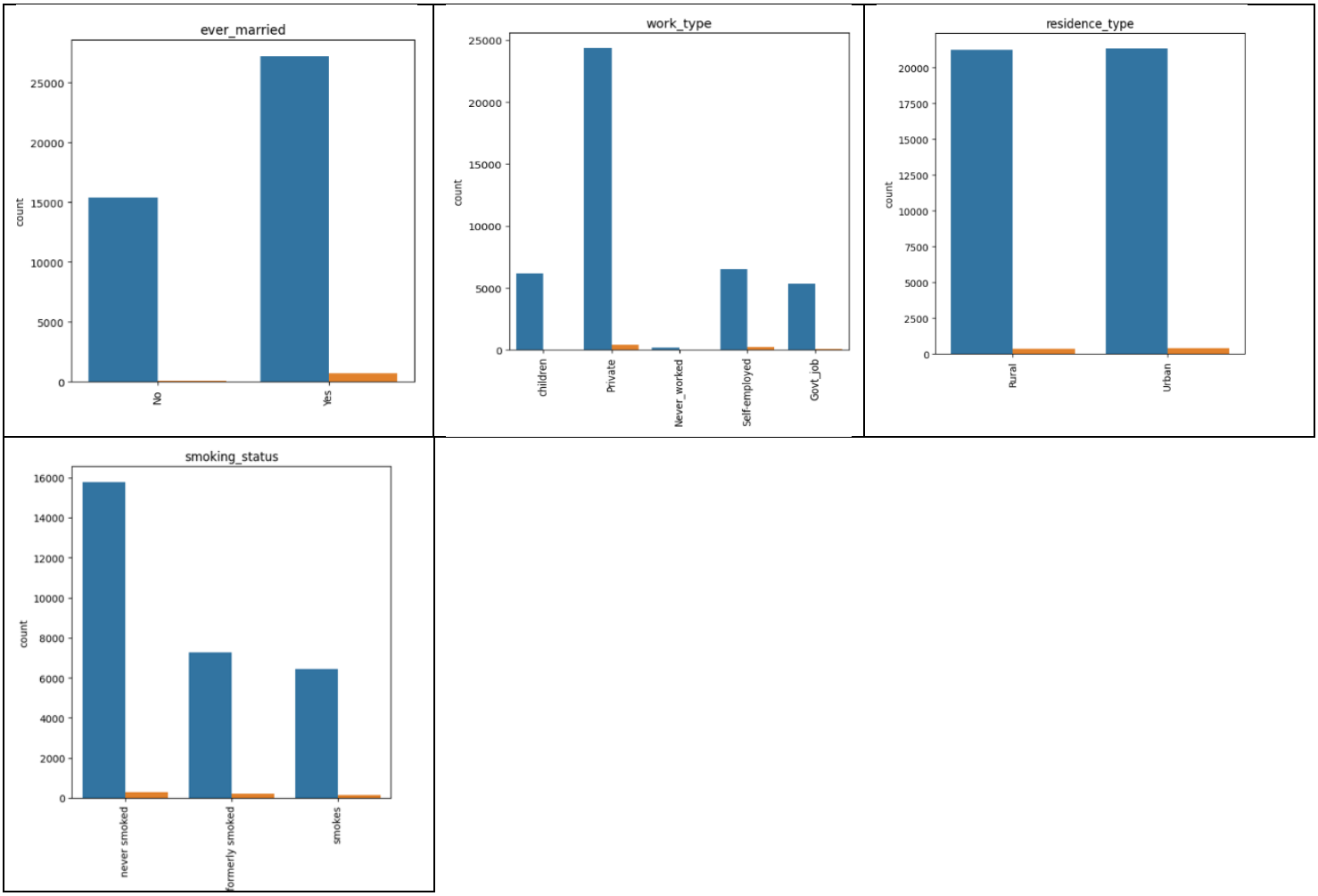


Fig. 2. Class distribution of categorical features

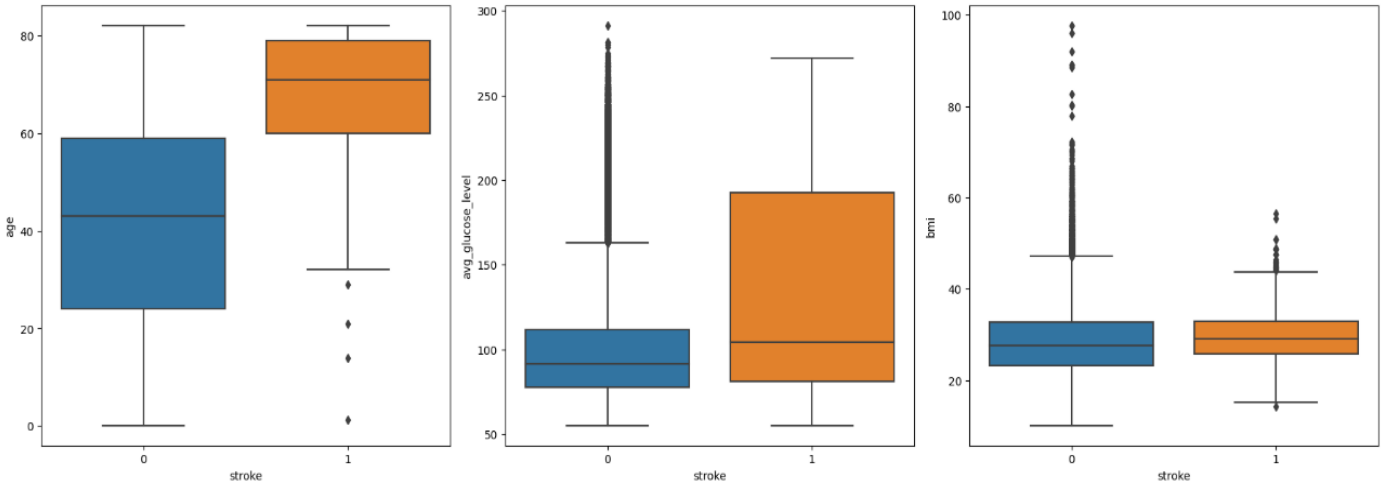


Fig. 3. Box plot of numerical features

The approximate proportion of men and women who get stroke is 23% and 26% respectively. This indicates that men have a higher chance of getting stroke disease than women, although the disease affects both men and women. The features such as hypertension and heart disease do not have a high impact on stroke. People who are married have a higher stroke rate. The class distribution of work type is evident that a majority of participants (75%) are employed in the private sector, and 42% of them

had a stroke. The distribution of participants across the two classes based on their residence is shown as approximately more of the urban people suffered from stroke than rural people. The class distribution of smoking status feature is evident that smoking habit does not have a major impact on stroke disease.

From the box plot representation in **Figure 3**, it is shown that People aged more than 60 years tend to have a stroke. Some outliers can be seen as people below age 20 are having a stroke it might be possible that it's valid data as stroke also depends on our eating and living habits. Another observation is people not having strokes also consist of people aged greater than 60 years. Based on the box plot representation, it's apparent that individuals who have experienced a stroke tend to exhibit an average glucose level exceeding 100. While there are clear outliers in cases of patients without stroke, it's plausible that these outliers could represent legitimate records. There is not a notable observation that clearly illustrates how BMI impacts the likelihood of experiencing a stroke. The histogram representation of numerical features was plotted to identify any potential relationship between the feature and stroke is shown in **Figure 4**. It shows that the risk of experiencing a stroke increased as patients age increased. Elderly patients were more prone to experiencing a stroke compared to younger patients. The highest percentage of patients who experienced a stroke fell within the BMI range of 25 to 35, surpassing patients from other groups. Elevated BMI does not result in an increased risk of stroke. Stroke cases occur in the range of values of glucose levels from 50 – 120 and also 200 – 250. Usually, Diabetes was identified in patients with readings exceeding 200mg/dL. Patients with readings falling between 140–199mg/dL were also categorized as having pre-diabetes. Diabetes stands as one of the contributing factors to the occurrence of a stroke, and individuals with pre-diabetes exhibit an elevated risk of experiencing a stroke.

3.3. Data preprocessing

Usually, the data that were collected is said to be raw data. To remove the redundant feature and fill in the missing values, the computation of pre-processing techniques can be handled. This technique is a significant process in every proposed structure that also eliminates the duplicate values in the dataset. In this phase, several steps are applied.

1. Some values are missing in smoking status and BMI features. The mean value must be calculated to fill in these missing values.

2. Transforming the string values into integer values of the corresponding features using Encoder. After the exploratory data analysis, the removal of outliers and duplicate values is required. The missing values should be filled in. It is noted that the BMI value is missing for about 3%. These missing values should be filled with mean values. Around 30% of the missing value is found in the feature smoking status.

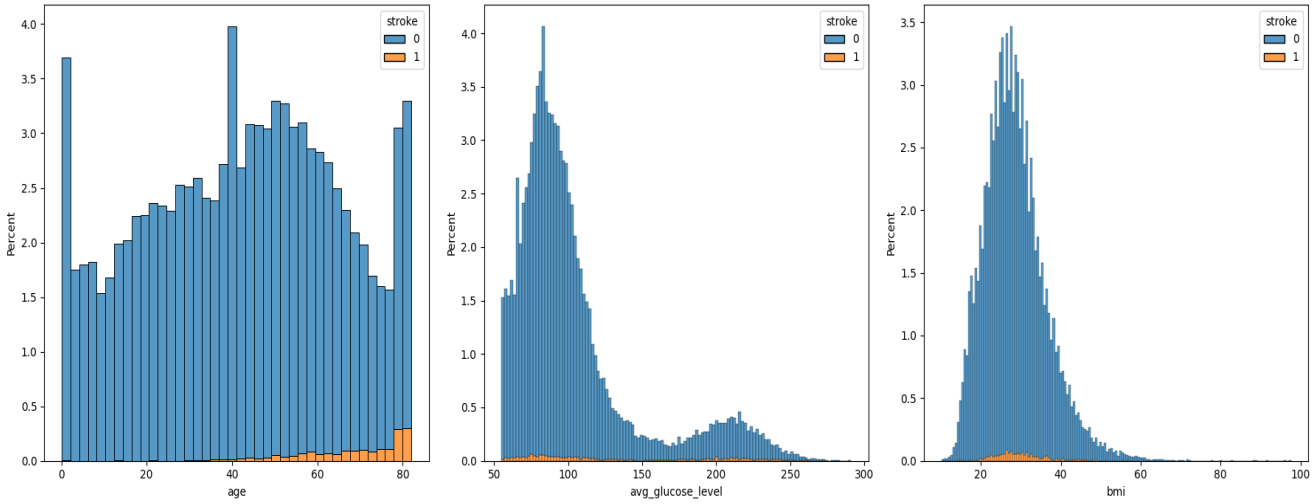


Fig. 4. Histogram plot of numerical features

However, data cleaning is highly required for this imbalanced dataset. Since the BMI feature belongs to numerical value the missing values can be filled by mean. The smoking status feature belongs to categorical data so it should be filled with 'No Info' for the missing one. The next step is to transform the string values into integer values using Encoder. After this step, the data is completely transferred into a numerical dataset.

3.4. Oversampling modeling

The oversampling method such as random oversampling duplicates minority samples to balance with majority samples. This method has a disadvantage that will lead to an overfitting problem. The most commonly used oversampling technique is SMOTE which generates synthetic samples and also overcomes the disadvantage of the random oversampling method [29]. Adasyn is an extension of SMOTE that generates instances by dense regions.

3.4.1. SMOTE

The basic idea of SMOTE is it first chooses a minority sample and finds its nearest neighbor. Then it randomly chooses the nearest minority sample. Subsequently, choose a point at random along the line connecting two samples to create the new synthetic minority sample. The generation of simulated points can be done in this manner and can be re-balanced easily [30]. The difference between the sample under consideration and its corresponding nearest neighbor is taken. Then it should be multiplied by an unpredictable number generated, normalized, and then finally added sequentially to the original vector under consideration.

It consists of two significant steps.

1. For each chosen minority sample, a neighborhood is defined, which identifies k nearest neighbors.

2. The N number of neighbor samples is selected, and new samples are generated in between the existing samples by interpolation method.

Assume a minority sample x_i, N randomly chosen samples from its neighborhood and p=1, ... N, a new sample x_i*p is expressed as $x_i*p = x_i + u (x_ip - x_i)$ where u is a number that takes a value between 0 and 1 which is generated randomly. It is faster and provides better classification accuracy. However, this SMOTE technique has issues regarding noise sample generation, which means there is a chance of generating new samples that belong to the majority samples. To overcome this issue, there are variants of smote have been developed. One among them is the Adasyn oversampling technique.

3.4.2. Adasyn

Adasyn is an adaptable approach for generating data that can evolve based on feedback and results which generates samples adaptively to mitigate the effects of imbalanced data in classification tasks. The dataset is illustrated as D and the number of samples represented as m. x_i, yi (i=1,2, . . . m) xi is an input sample and y_i is an output sample. $y_i \epsilon$ 0,1 is a class label, y_i=0 is mentioned as an underrepresented class, and y_i=1 is mentioned as an over-represented class. The number of samples in the underrepresented class is assigned as m_0 and the number of samples in the overrepresented class is assigned as m1, where, m0 is less than m_1, $m_0+m_1=m$ [31]. The scatterplot of numerical features using Adasyn is shown in **Figure 5**.

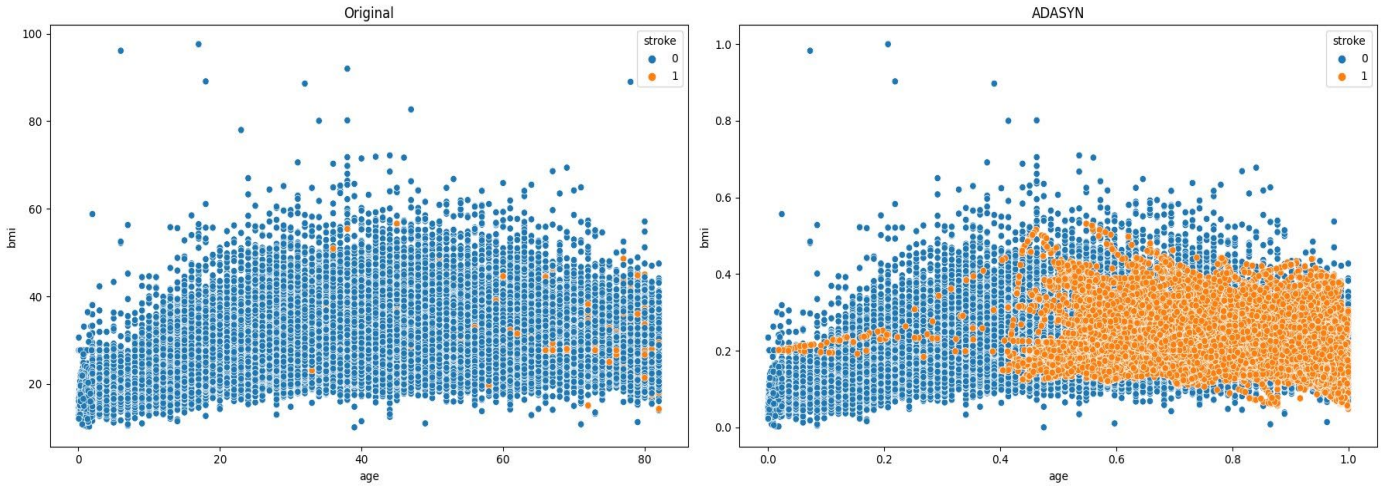


Fig. 5. Scatterplot of numerical features

The used dataset is highly imbalanced. Out of 43400 samples, it is divided as 42617 and 783 for no stroke class and stroke class respectively. The dataset distribution between no stroke and stroke classes are 1.8% and 98.1% respectively. It is not advisable to use the under-sampling method to solve class imbalance, though it may lose the required information of samples. So here two techniques of oversampling such as smote and adasyn have been used. Among these, adasyn is more suitable for this dataset while performing classification. **Table 3** describes the dataset before and after the oversampling method. The dataset is divided into 80% for training (34094 for no stroke class and 626 for stroke class) and 20% testing (8523 for no stroke class and 157 for stroke class). It also shows the count of samples before and after smote and adasyn methods during training. For smote, the dataset for both classes are balanced, which has 34094 for no stroke and stroke. For adasyn, the number of samples for no stroke is 34094, and for stroke is 34289.

Table 3. Dataset before and after oversampling method

	Dataset			
Phase	Training (80%)		Testing (20%)	
Classes	No-Stroke	Stroke	No-Stroke	Stroke
All Dataset	34094	626	8523	157
Before Oversampling	34094	626	8523	157
After Smote	34094	34094	8523	157
After Adasyn	34094	34289	8523	157

3.5 System Architecture

Once data has been processed, it becomes available for model construction. Model development involves utilizing various data preparation and ML techniques. Some of the techniques employed include DT, LR, RF, and Artificial Neural network (ANN). To compare the performance of these models, accuracy metrics such as accuracy score, precision score, recall score, and F1 score are used. **Figure 6** illustrates a block schematic of the proposed system architecture.

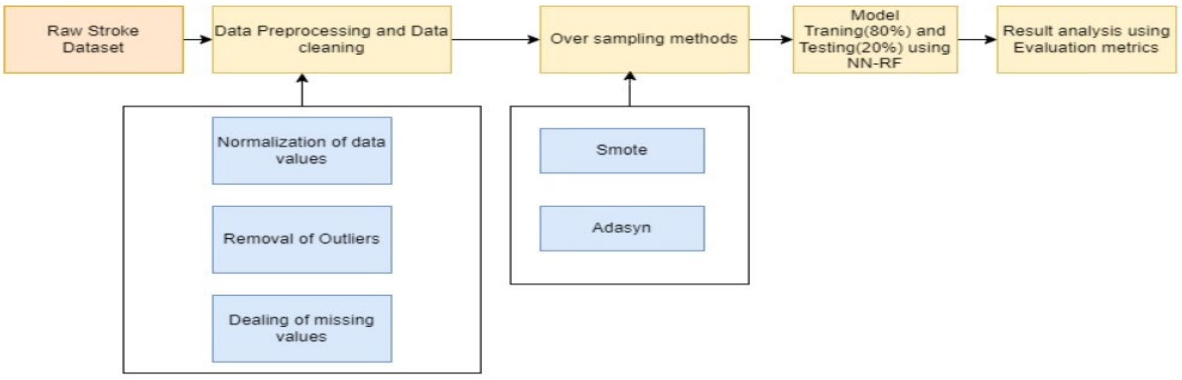


Fig. 6. Architecture of proposed model

3.6 Classification algorithm

The detailed analysis of three classification approaches for stroke prediction is discussed in this section. They are LR, DT, and NN-RF. The benefits of combining neural network with RF for classification are as follows,

1. There are a lot of various interpretability tools and techniques in a tree-based model that can be used by a single unique library.

2. It is easy to combine the neural network and decision forest by using this library, whereas the output of the neural network can be consumed by a tree-based model.

3. It is possible to solve ranking problems in addition to classification and regression using this model.

3.6.1 Logistic Regression

One of the supervised classification algorithms is the LR algorithm. Based on the logistic function, the individuals are classified. This LR suits well for the environment where the data points do not fit properly. This can be achieved with the help of the Sigmoid function. The equation for simple linear regression, which is used to model the linear relationship between two variables, is expressed in Equation (1).

$$y = b_0 + b_1 * x \quad (1)$$

So, the function is then transformed by applying the sigmoid function to it, and it is shown in Equation (2).

$$p = \frac{1}{1 + e^{-y}} \quad (2)$$

Now the value of y is calculated by using Equation (3).

$$\ln\left(\frac{p}{1-p}\right) = b_0 + b_1 * x \quad (3)$$

Implement the gradient descent algorithm to optimize the coefficient and intercept. The purpose of this model is to evaluate the extent to which each independent variable contributes to the probability of the outcome variable, by quantifying their statistical significance and estimating their effect size.

3.6.2 Decision tree

The algorithm can be trained on a dataset containing labelled examples of different categories, which can then be used to predict the category of new, unseen examples based on the learned patterns and features extracted from the training set. One of the robust approaches popularly adopted in ML is DT. The efficient combination of a series of basic tests represents the successive DT model. In each test, the numerical value will be compared to a threshold value. It is mainly used for grouping purposes. Each tree is composed of nodes and branches. The node represents features to be classified.

3.6.3 Artificial Neural Network

It is simply referred to as Neural Network (NN). Its structure and functionalities are similar to the human brain. It is one of the types of ML algorithms that can learn complex patterns in data and make predictions based on the learning data. Neuron is the basic building block of NN. It consists of multiple layers of neurons. Neurons in each layer have been connected and associated with their weights. The output has been produced with the help of the activation layer. These are trained with labelled data and produce predicted target output.

3.6.4 Random Forest

RF is a versatile and most commonly used ML algorithm that belongs to the ensemble model. It can perform classification and regression. It consists of multiple decision trees, where each tree is trained independently. There is less chance of getting overfitting problems in this RF model.

3.6.5 Neural network-random forest

This classification model is a combination of the artificial neural network but the SoftMax layer is replaced with a decision forest, which consists of several decision trees [32]. The input features are considered as X and the output feature is represented as y. A decision tree is a two-way tree comprised of both branching vertices and outcome vertices. Here, N represents the branching vertex index within the decision tree, while L signifies the collection of outcome vertex indices 1, . . ., L. Each outcome vertex, denoted as lϵL, is associated with a probabilistic distribution across the output space Y. Additionally, every decision node, denoted as nϵN, is allocated a boundary rule. Furthermore, the mapping function X→R is established within the artificial Neural Network [33]. This function plays a crucial role in shaping the behavior of the boundary rule associated with the decision trees.

Algorithm:

Input – dataset S= (x_1, y_2) . . .. (x_n, y_n), features F, and quantity of trees in forest

Output – Stroke or not Stroke

1. Function NN-RF (S, F)

2. Initialize parameters of Decision Forest Classifier such as number of trees, number of nodes in each tree, min and max depth of a tree

3. For i less than mindepth to maxdepth, do

4. Assign nodes for each categorical feature and numerical feature

5. End for

6. To calculate out of bag accuracy, initialize the dataset S for training construct NN-RF with K trees train NN-RF using dataset S for each decision tree Ti in NN-RF do

7. Si=bootstrapped dataset used for training T_i

8. Si_c=S/Si

9. Pi= predictions on dataset Sic using Ti, calculate average performance or error using predictions

10. P=p_1, p_2, ...p_k

The corresponding pseudocode has been given below. Initially, data has been loaded, then if any values of the dataset are null, it will be replaced with mean. After preprocessing the data, it has been divided into training and testing phases. Though the dataset is highly imbalanced, both smote and adasyn oversampling have been applied. Classification has been done with the proposed NN-RF classifier.

Pseudocode:

BEGIN
Data← load dataset
Find mean and replace if data. Value is equal to NAN or null
Preprocessing:
Encode data if data. dtypes=object
X←data. Drop ['stroke']
Y←data. Stroke
Split data as x_1, x_2, y_1, y_2
S_x←smote (data_x)
A_x←adasyn (data_x)
Classifier←train model using A_x
Predict←test [data (x_2, y_2)]

The learning rate is assigned as 0.001, the batch size is fit to 150, and the total number of epochs is 100. The number of trees is 300 and their corresponding depth is 10. It is observed from the given algorithm, that it is associated with training set S, features samples, number of trees were treated as input.

The variables such as the number of trees, the number of nodes in each tree, and min and max depth of a tree should be initialized. Then assign nodes value for each tree using for statement. The name of the parameter, its default value, and their description are represented in **Table 4.**

Table 4. Description of parameters used for NN-RF

Name of the Parameter	Default Value	Description of the parameters
Learning Rate	0.001	Step size used by optimization algorithm during training
Batch size	150	No samples per patch
epochs	100	No of times the model will iterate during training
L2 regularization	0.001	Regularization with small weight decay
n_estimators	300	Number of trees to fit
Max_depth	10	Allow trees to grow deep enough to capture complex patterns without overfitting
Min_samples _split	2	The minimum number of samples required to split an internal node
Min_samples _leaf	1	Minimum number of samples to be at leaf node

4. Results

The experiment is done with Windows 11 OS that has RAM of 16 GB, a Hard Disk of 500GB, and Python 3.8 is used. The input dataset consists of 43400 records and 12 features that represent major risk factors of Stroke disease.

4.1 Performance metrics to evaluate the model

The ML algorithm's performance is examined by different evaluations such as recall, precision, accuracy, and F-score [34]. They are calculated as follows

True Positive (TP) – the No of persons who had stroke occurred gives the output as "stroke is predicted"

True Negative (TN) – the No of persons who do not have a stroke gives the output as "No stroke is predicted"

False Positive (FP) – the No of persons who do not suffer from a stroke but give the output as "stroke is predicted"

False Negative (FN) – the No of persons who had a stroke occurred but gives the output as "No Stroke is predicted"

The evaluation metrics such as accuracy, precision, recall, and F1-measure can be defined as shown in Equations (4), (5), (6), and (7) respectively.

Accuracy:

Accuracy is the percentage of actual predicted instances from the available instances. It is calculated by using the equation.

$$A = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

Precision:

Precision is defined as the number of correctly positive predictions from the number of positive predictions.

$$P = \frac{TP}{TP + FP} \quad (5)$$

Recall:

The recall is defined as the rate of values that measure positive predictions to the classifier correctly predicted.

$$R = \frac{TP}{TP + FN} \quad (6)$$

F1-Measure:

To find the relationship between precision and recall, the F-Measure has been used. The smaller value of precision or recall will represent the F-Measure value. It is given as follows.

$$F1 = \frac{2 * precision * recall}{precision + recall} \quad (7)$$

The other metric such as the Area under ROC curve has also been used to evaluate the prediction performance of models.

4.2 Evaluating performance of the proposed model

The outcome of the NN-RF algorithm is compared with the existing ML classification algorithms to predict the stroke in EHR. The comparison was done with and without oversampling techniques such as SMOTE and Adasyn for all five classification algorithms [35].

With the help of a confusion matrix, the evaluation of the model can be performed. This comparison takes place between this model and other ML algorithms such as DT and LR. The performance metrics of the proposed model have outperformed the other two algorithms in terms of F1 measure, accuracy, and Auc values. The comparison of performance metrics among five

Table 5. Performance metrics without Sampling

Machine Learning Classifier	Precision	Recall	F1-Score	Accuracy	Auc
DT	0.69	0.70	0.69	0.60	0.54
LR	0.69	0.70	0.69	0.59	0.53
NN	0.68	0.69	0.67	0.63	0.60
RF	0.71	0.70	0.68	0.69	0.69
NN-RF	0.72	0.72	0.72	0.74	0.75

Table 6. Performance metrics with SMOTE Technique

Machine Learning Classifier	Precision	Recall	F1-Score	Accuracy	Auc
DT	0.71	0.72	0.71	0.69	0.66
LR	0.72	0.62	0.67	0.70	0.71
NN	0.61	0.63	0.69	0.70	0.68
RF	0.73	0.72	0.71	0.73	0.74
NN-RF	0.72	0.75	0.72	0.76	0.77

Table 7. Performance metrics with Adasyn Technique

Machine Learning Classifier	Precision	Recall	F1-Score	Accuracy	Auc
DT	0.72	0.74	0.73	0.75	0.77
LR	0.75	0.65	0.70	0.76	0.77
NN	0.68	0.65	0.62	0.68	0.72
RF	0.72	0.71	0.74	0.79	0.80
NN-RF	0.75	0.76	0.75	0.84	0.86

machine learning algorithms without sampling, with the Smote technique and with the Adasyn technique are shown in **Tables 5,6, and 7** respectively. The F1 – score of LR is lesser than DT which in turn is lesser than NN-RF and their corresponding single models in case of without sampling, with SMOTE, and with Adasyn. The accuracy and Auc values of the proposed NN-RF model outperform the other models including single models such as NN and RF.

Analysing an imbalanced stroke prediction dataset using machine learning techniques

The corresponding ROC curve of the NN-RF model in terms of without sampling, with smote and with adasyn is shown in **Figures 7, 8, and 9** respectively.

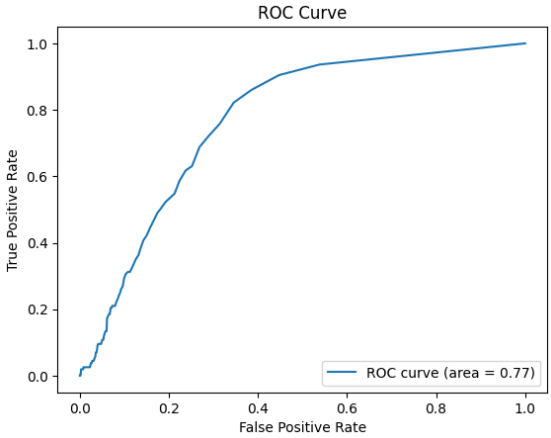


Fig. 7. Roc curve of NN-RF without sampling

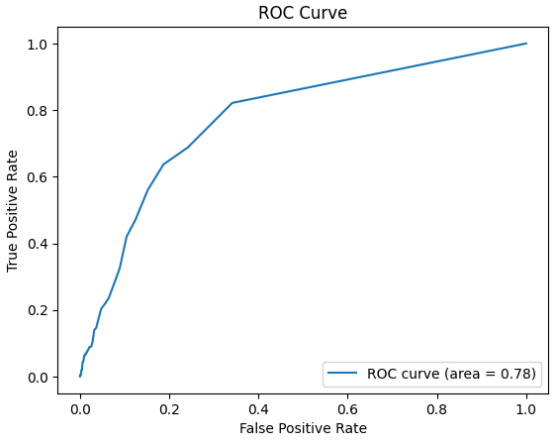


Fig. 8. Roc curve of NN-RF with Smote

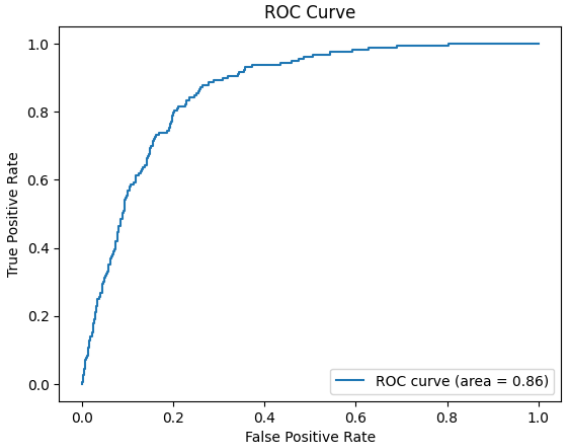


Fig. 9. Roc Curve of NN-RF with Adasyn

The important performance indicators for ML algorithms are training and execution time. Training time refers to the duration it takes for a model to undergo training on a given dataset, while execution time encompasses the overall duration required for computations, encompassing tasks like data splitting, preprocessing, and model evaluation. The training and execution time for LR are 0.45s and 6.54s respectively. For the DT algorithm, the respective times are 0.74s and 7.47s. the training time and execution time for RF are 9.8s and 17.93s. For NN algorithm, it takes 110.87s for training time and 130.85s for execution time. The training time and execution time for the proposed NN-RF are 122.67s and 149.78s

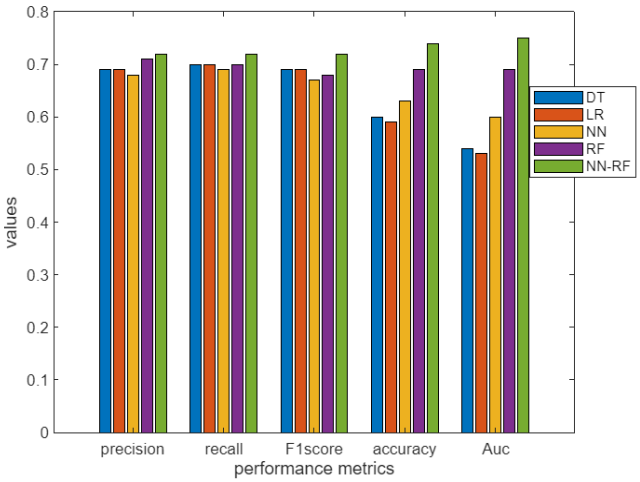


Fig. 10 Comparison of Performance metrics without sampling

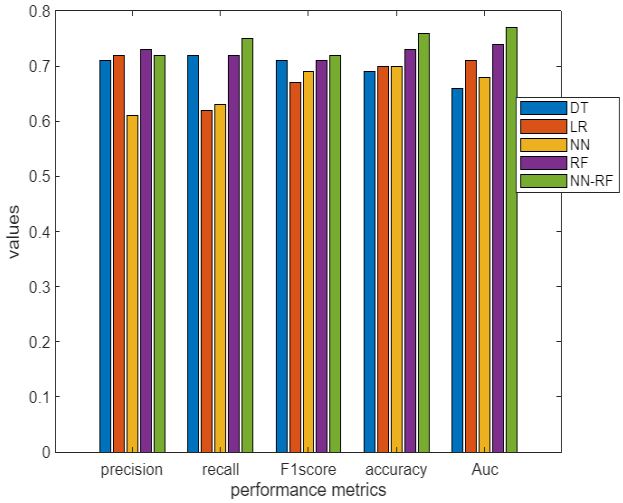


Fig. 11. Comparison of performance metrics on with smote

Analysing an imbalanced stroke prediction dataset using machine learning techniques

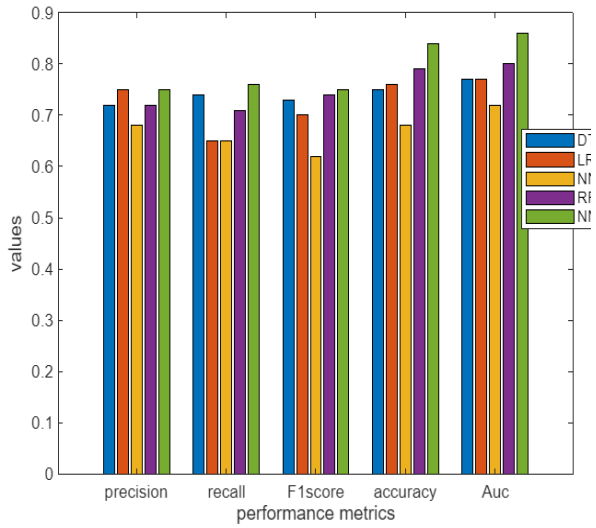


Fig. 12. Comparison of performance metrics with adasyn

5. Discussion

Among the chosen base models, the NN-RF model demonstrated the highest efficiency across all the evaluation metrics. Likewise, elevated values were attained by the LR and DT classifiers. Observing precision metrics, the SMOTE and Adasyn values are approximately similar discrimination abilities for DT and NN-RF classifier models. There is a slight difference in the Recall value between SMOTE and Adasyn of the LR classifier. A higher difference in precision and recall values for the LR model in the case of SMOTE and Adasyn techniques. Since the dataset is unbalanced, the F1 measure serves as an appropriate metric to gauge the efficiency of the ML model of the dataset. From the observation, the F1- measure of NN-RF with the Adasyn oversampling technique is 3% higher than the SMOTE technique. However, in terms of accuracy, the adasyn got 8% higher than SMOTE. The comparative graph of precision, recall, and F1 – Score, Accuracy, and Auc values for five classification algorithms is shown in **Figures 10, 11, and 12** respectively.

Here utilizing the open dataset is the main limitation. These data are characterized by finite dimensions and attributes differing from those found in hospital or institutional datasets. While the latter could potentially provide more comprehensive information and diverse features, collecting an elaborate health profile of participants, gaining access to such data is often a laborious process due to time constraints and privacy considerations.

The comparison of similar studies on similar datasets as well as with the stroke dataset has been shown in **Table 8.** It shows that the proposed model achieved higher accuracy.

Table. 8 Comparison with similar studies

Article name	Methodology	Accuracy (%)	Resampling techniques	Datasets
A predictive analytics approach for stroke prediction using machine learning and neural networks	DT, RF, and NN which was the best	77	Random down sampling	Stroke prediction dataset
Analyzing the performance of stroke prediction using ML classification algorithms	LR, DT, RF, KNN, SVM, and Naivebayes which was the best	82	Under-sampling	Stroke prediction dataset
Stroke Risk prediction using Artificial intelligence techniques through Electronic health records	LR and SVM which was the best	73	No sampling	EHR(MIMIC-III)
An Imbalanced-Data Processing Algorithm for the Prediction of Heart Attack in Stroke Patients	KNN, DT, SVM, Adaboost, and RF which was the best	70	Undersampling-Clustering-Oversampling	Stroke samples
Proposed model	NN-RF	84	Smote and Adasyn	Stroke prediction dataset

6. Conclusions and Future Work

A stroke poses a critical risk to human life that demands prevention and treatment to avert unforeseen complications. In the present era of swift ML advancement, clinical providers, and medical professionals can leverage models to handle imbalanced issues contributing to a stroke incidence. They can also evaluate the associated probability or risk of occurrence. ML can play a vital role in timely predicting strokes and minimizing their severe consequences [36]. The features present in EHR have the representation of 10 in number. The feature

space for predictive modeling cannot be reduced further without a significant loss of data. Hence, all the features had been used for the stroke prediction. This study compared various ML algorithms, such as DT, and LR, with the NN-RF algorithm separately with SMOTE and Adasyn oversampling techniques outperformed the stroke prediction model based on multiple features that encapsulate the profile of the participants. It is found that NN-RF along with Adasyn oversampling has the best performance with an F1 – measure of 75%. Also evaluating classifier performance through metrics such as AUC and accuracy is crucial for interpreting model effectiveness and showcasing their classification performance. The hybrid NN-RF classification method demonstrates superior performance compared to other approaches, achieving an AUC of 86% and an accuracy of 84%. By integrating these findings, the efficiency and effectiveness of the public health system can be enhanced.

Various paths are waiting for further work in the future. It is better to combine the electronic records dataset with a contextual understanding of various diseases and medications drawn from the publicly accessible Linked Open Data (LOD) cloud. This can be achieved by elevating the dataset to Linked Data standards and establishing interconnections with the LOD cloud. The process of interlinking is complex. However, the integration of background knowledge has the potential to enhance the performance of classification algorithms for stroke prediction, although the maintenance of these interconnections is vital.

Conflict of interest

There is no conflict of interest.

Acknowledgment

We would like to thank everyone who contributed and worked hard to achieve this work.

References

[1] N. Biswas, K. M. M. Uddin, S. T. Rikta, S. K. Dey, A comparative analysis of machine learning classifiers for stroke prediction: A predictive analytics approach, Healthcare Analytics. 2(2022) 100116. https://doi.org/10.1016/j.health.2022.100116.

[2] M. Mourguet, D. Chauveau, S. Faguer, J. B. Ruidavets, Y. Bejot, D. Ribes, A. Huart, L. Alric, L. Balardy, L. Astudillo, D. Adoue, Increased ischemic stroke, acute coronary artery disease and mortality in patients with granulomatosis with polyangiitis and microscopic polyangiitis, Journal of autoimmunity. 96 (2019) 134-14. https://doi.org/10.1016/j.jaut.2018.09.004.

[3] S. A. M. I. Ahmed Mostafa Ibrahim, D. Elzanfaly, A. Yakoub, Machine Learning Models for Predicting Brain Strokes, FCI-H Informatics Bulletin. 4 (2022) 1-6. https://doi.org/10.21608/fcihib.2022.76906.1048.

[4] Y. Ma, Y. Wang, J. Yang, Y. Miao, W. Li, Big health application system based on health internet of things and big data, IEEE Access. 5 (2016) 7885-7897. https://doi.org/10.1109/ACCESS.2016.2638449.

[5] P. Yadav, M. Steinbach, V. Kumar, G. Simon, Mining electronic health records (EHRs) A survey, ACM Computing Surveys (CSUR) 50 (2018) 1–40. https://doi.org/10.1145/3127881.

[6] H. Chen, T. Li, X. Fan, C. Luo, Feature selection for imbalanced data based on neighborhood rough sets, Information sciences. 483 (2019) 1-20. https://doi.org/10.1016/j.ins.2019.01.041.

[7] L. Sun, T. Wang, W. Ding, J. Xu, A. Tan, Two-stage-neighborhood-based multilabel classification for incomplete data with missing labels, International Journal of Intelligent System. 37 (2022) 6773-6810. https://doi.org/10.1002/int.22861.

[8] P. Kaur, A. Gosain, Robust hybrid data-level sampling approach to handle imbalanced data during classification, Soft Computing. 24 (2020) 15715-15732. https://doi.org/10.1007/s00500-020-04901-z.

[9] N. V. Chawla, A. Lazarevic, L. O. Hall, K. W. Bowyer, SMOTEBoost: Improving prediction of the minority class in boosting. in Knowledge Discovery in Databases: PKDD 2003, springer. 2838 (2017)107-119. https://doi.org/10.1007/978-3-540-39804-2_12.

[10] H. He, Y. Bai, E. Garcia, S. A. Li, Adaptive synthetic sampling approach for imbalanced learning, IEEE international joint conference on neural networks. 2008 (2008) 1322-1328. https://doi.org/10.1109/IJCNN.2008.4633969.

[11] S. Liu, K. Zhang, Under-sampling and feature selection algorithms for S2SMLP, IEEE Access. 8 (2020) 191803-191814. https://doi.org/10.1109/ACCESS.2020.3032520.

[12] S. Bhattacharya, P. K. R. Maddikunta, S. Hakak, W. Z. Khan, A. K. Bashir, A. Jolfaei, U. Tariq, Antlion re-sampling based deep neural network model for classification of imbalanced multimodal stroke dataset, Multimedia Tools and Applications. 81(2020) 41429–41453. https://doi.org/10.1007/s11042-020-09988-y.

[13] T. Liu, W. Fan, C. Wu, A hybrid machine learning approach to cerebral stroke prediction based on imbalanced medical dataset, Artificial intelligence in medicine. 101 (2019) 101723. https://doi.org/10.1016/j.artmed.2019.101723.

[14] S. Dev, H. Wang, C. S. Nwosu, N. Jain, B. Veeravalli, D. John, A predictive analytics approach for stroke prediction using machine learning and neural networks, Healthcare Analytics. 2 (2022) 100032. https://doi.org/10.1016/j.health.2022.100032.

[15] E. Dritsas, M. Trigka, Stroke risk prediction with machine learning techniques, Sensors 22 (2022) 4670, https://doi.org/10.3390/s22134670.

[16] C. S. Nwosu, S. Dev, P. Bhardwaj, B. Veeravalli, D. John, Predicting stroke from electronic health records, 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). (2019) 5704-5707. https://doi.org/10.1109/EMBC.2019.8857234.

[17] M. S. Pathan, Z. Jianbiao, D. John, A. Nag, S. Dev, Identifying stroke indicators using rough sets, IEEE Access. 8 (2020) 210318-210327. https://doi.org/10.1109/ACCESS.2020.3039439.

[18] P. Govindarajan, R. K. Soundarapandian, A. H. Gandomi, R. Patan, P. Jayaraman, R. Manikandan, Classification of stroke disease using machine learning algorithms, Neural Computing and Applications. 32 (2020) 817-828. https://doi.org/10.1007/s00521-019-04041-y.

[19] N. Patil, and A. Sumarsono, Stroke Prediction Using Machine Learning, Journal of Research in Engineering and Computer Sciences. 2 (2024) 61-72.

[20] S. Jiang, Y. Gu, E. Kumar, Stroke Risk Prediction Using Artificial Intelligence Techniques Through Electronic Health Records, Artificial Intelligence Evolution. 4 (2023) 88-98. https://doi.org/10.37256/aie.4120232744.

[21] M. Khairy, T. M. Mahmoud, T. Abd-El-Hafeez, The effect of rebalancing techniques on the classification performance in cyberbullying datasets, Neural Comput and Applications. 36 (2024) 1049–1065. https://doi.org/10.1007/s00521-023-09084-w

[22] T. Abd El-Hafeez, M. Y. Shams, Y. A. M. M. Elshaier, H. M. Farghaly, A. E. Hasanien, Harnessing machine learning to find synergistic combinations for FDA-approved cancer drugs, Scientific Reports. 14 (2024) 2428. https://doi.org/10.1038/s41598-024-52814-w.

[23] E. H. I. Eliwa, A. M. El.Koshiry, T. AbdEl-Hafeez, H. M. Farghlay, Utilizing convolutional neural networks to classify monkeypox skin lesions, Scientific Reports. 13 (2023) 14495. https://doi.org/10.1038/s41598-023-41545-z.

[24] M. Wang, X. Yao, Y. Chen, An Imbalanced-Data Processing Algorithm for the Prediction of Heart Attack in Stroke Patients, IEEE Access. 9 (2021) 25394-25404. https://doi.org/10.1109/ACCESS.2021.3057693.

[25] B. A. Goldstein, A. M. Navar, M. J. Pencina, J. P. Ioannidis, Opportunities and challenges in developing risk prediction models with electronic health records data: A systematic review, Journal of the American Medical Informatics Association: JAMIA. 24 (2017) 198-208. https://doi.org/ 10.1093/jamia/ocw042.

[26] J. Yu, S. Park, S. H. Kwon, C. M. B. Ho, C. S. Pyo, H. Lee, AI-based stroke disease prediction system using real-time electromyography signals, Applied Sciences. 10 (2020) 67. https://doi.org/10.3390/app10196791.

[27] V. L. Feigin, M. Brainin, B. Norrving, S. Martins, R. L. Sacco, W. Hacke, M. Fisher, J. Pandian, P. Lindsay, World Stroke Organization (WSO): global stroke fact sheet, International Journal of Stroke. 17 (2022)18-29. https://doi.org/10.1177/17474930211065917.

[28] M. S. Pathan, A. Nag, M. M. Pathan, S. Dev, Analyzing the impact of feature selection on the accuracy of heart disease prediction, Healthcare Analytics. 2 (2022) 100060. https://doi.org/10.1016/j.health.2022.100060.

[29] L. Sun, M. Li, W. Ding, E. Zhang, X. Mu, J. Xu, AFNFS: Adaptive fuzzy neighborhood based feature selection with adaptive synthetic over-sampling for imbalanced data, Information Sciences. 612 (2022) 724-744. https://doi.org/10.1016/j.ins.2022.08.118.

[30] P. Vuttipittayamongkol, E. Elyan, A. Petrovski, On the class overlap problem in imbalanced data classification, Knowledge-based systems. 212 (2022) 106631. https://doi.org/10.1016/j.knosys.2020.106631.

[31] Y. Bao, S. Yang, Two Novel SMOTE Methods for Solving Imbalanced Classification Problems, IEEE Access. 11 (2023) 5816-5823. https://doi.org/10.1109/ACCESS.2023.3236794.

[32] A. Sjöberg, E. Gustavsson, A. C. Koppisetty, M. Jirstrand, Federated Learning of Deep Neural Decision Forests, Machine Learning, Optimization, and Data Science. 11943 (2019) 700-710. https://doi.org/10.1007/978-3-030-37599-7_58.

[33] J. Sun, X. Liu, X. Mei, J. Zhao, M. D. Plumbley, V. Kılıc¸, W. Wang, Deep neural decision forest for acoustic scene classification, 30th European Signal Processing Conference (EUSIPCO). (2022) 772-776. https://doi.org/10.23919/EUSIPCO55093.2022.9909575.

[34] G. Sailasya, GLA. Kumari, Analyzing the Performance of Stroke Prediction using ML Classification Algorithms, International Journal of Advanced Computer Science and Applications 12 (2021) http://dx.doi.org/10.14569/IJACSA.2021.0120662.

[35] P. cihan, Horse Surgery and Survival Prediction with Artificial Intelligence Models: Performance Comparison of Original, Imputed, Balanced, and Feature- Selected Datasets, Kafkas Universitesi Veteriner Fakultesi Dergisi. 30 (2024) 233-241. http://doi.org/ 10.9775/kvfd.2023.30908.

[36] V. S. E and R. D, A Systematic Method of Stroke Prediction Model based on Big Data and Machine Learning, Smart Technologies, Communication and Robotics (STCR), Sathyamangalam, India, (2022) 1-5. http://doi.org/ 10.1109/STCR55312.2022.10009283.