Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

LEVERAGING ADABOOST AND GRIDSEARCHCV FOR OPTIMIZED STROKE PREDICTION OUTCOMES

[1]Ronke S. Babatunde, [2]Akinbowale N. Babatunde, [3]Sulaiman O. Abdulsalam, [4]Ruth O. Oludare, [5]Mohammed K. Abdulsalam [6]Omosola J. Olabode

[1,2,3,4,5]Department of Computer Science. Kwara State University, Malete. Nigeria , [6]Department of Mathematical and Computing Science. Thomas Adewumi University, Oko-Irese. Nigeria

[1]ronke.babatunde@kwasu.edu.ng, [2]akinbowale.babatunde@kwasu.edu.ng, [3]sulaiman.abdulsalam@kwasu.edu.ng , [4]ruth.oludare@gmail.com, [5]imuhammedkolawole@gmail.com , [6]omosola.olabode@tau.edu.ng

Abstract

Background: *Globally, stroke is one of the most common causes of mortality and long-term disability. According to World Health Organization (WHO) report, 15 million people suffer a stroke annually, with 5 million dying and another 5 million becoming permanently disabled, placing a major burden on families and communities. Some researchers have attempted to predict stroke using machine learning algorithms; however, while these algorithms yielded high accuracy, limited attention was given to hyperparameter tuning and class imbalance issues.* ***Aim:*** *This study addresses the critical need for enhanced stroke prediction by employing the AdaBoost machine learning algorithm, optimized through GridSearchCV for hyperparameter tuning to mitigate the aforementioned problems.* ***Method:*** *Utilizing a dataset from Kaggle, pre-processing steps, including feature selection, encoding, and class balancing via synthetic minority oversampling technique (SMOTE), were implemented. The model's performance significantly improved post-tuning, with accuracy rising from 75.42% to 84.33%, precision from 71% to 79%, and recall from 87% to 93%. These results highlight the importance of hyperparameter optimization in enhancing machine learning models for medical diagnostics.* ***Results:*** *The findings suggest that the optimized AdaBoost model can aid in early stroke detection, potentially improving patient outcomes and offering a valuable tool for healthcare providers. This study contributes to the field by setting a benchmark for stroke prediction and providing a foundation for future advancements in predictive analytics.*

Keywords: *prediction, adaboost, gridsearchcv, machine learning, hyperparameter tuning*

1. Introduction

Stroke is a leading cause of disability and mortality worldwide, characterized by an acute focal injury of the central nervous system (CNS) due to vascular causes such as cerebral infarction, intracerebral hemorrhage (ICH), and subarachnoid hemorrhage (SAH) (Easton et al., 2009). Strokes predominantly affect individuals in their productive years and are major contributors to dementia and depression (Johnson et al., 2016). Symptoms include sudden weakness or numbness, difficulty speaking, vision problems, loss of balance, dizziness, and severe headaches (Drewry, 2020). Advances in artificial intelligence (AI) and machine learning (ML) have shown potential in enhancing stroke prediction and diagnosis by identifying patterns in medical records for early disease detection (Monteiro et al., 2018). Amidst advancements in machine learning algorithms for early stroke diagnosis, timely and effective recognition and treatment continue to be challenging, hindering improvements in patient outcomes (Tuan, 2022). While previous studies have demonstrated the potential of machine learning algorithms in stroke prediction, there is a notable gap in the inclusion of hyperparameter tuning. Hyperparameter tuning is essential for maximizing the performance of machine learning models. AdaBoost has been successfully used in various medical diagnostic studies, including (Abirami et al., 2019; Moghadam et al., 2019; Ardam et al., 2019). However, many existing models with AdaBoost are deficient in terms of no hyperparameter tuning, which is essential for optimizing model performance (Sajjad et al., 2021). Hence, this research aims to address these

© The Author(s), under exclusive license to SLUJST 2025

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

gaps by employing GridsearchCV to tune the hyperparameters of the AdaBoost algorithm, thereby enhancing stroke prediction performance. Grid search constructs a grid of hyperparameter values and performs an exhaustive search to identify optimal values, thereby improving model accuracy and efficiency (Rocca, 2019). By enhancing the accuracy, precision, and recall of stroke prediction models, this study seeks to advance clinical decision-making and improve patient outcomes.

Additionally, class balancing is a crucial technique in machine learning, used to address the issue of class imbalance which is common with real world medical datasets, where certain classes in the dataset are underrepresented compared to others. This imbalance can lead to biased models that favor the majority class, resulting in poor predictive performance for the minority class (He & Garcia, 2009). Many studies focus primarily on accuracy as the main performance metric. However, in the presence of class imbalance, accuracy alone is insufficient. Comprehensive metrics such as precision and recall, are necessary for a holistic assessment of model performance. Several techniques are employed to address class imbalance, including: oversampling as seen in (Chawla et al., 2002), undersampling in (Tomek & Bell, 1985), algorithmic approaches discussed in (Elkan, 2001), Breiman, 1996, and Freund & Schapire, 1997), (Saito & Rehmsmeier, 2015). A simple, effective and widely used method is SMOTE, which is adopted in this research.

Furthermore, the literature amass the use of machine learning in healthcare, particularly for stroke prediction. However, several gaps remain that necessitate further investigation. Many existing stroke prediction models are silent on hyperparameter tuning, leading to suboptimal performance. Sajjad et al. (2021) highlighted this issue, emphasizing the need for tuning the parameters of models to enhance accuracy, precision, and recall, making predictive models more reliable and effective in clinical applications. Li et al., (2019) improved on stroke risk level classification methods using machine learning model. Oversampling and under-sampling method was used to address class imbalance issues in the dataset. The work used four machine learning models which include logistic regression model, bagged decision tree model, voting model and boosting model. The first 2 algorithms outperformed the last 2 based on the experiment performed. The major strength is the consistency of accuracy while working with the few features. However, the work recommends the use of larger dataset with more features which is expected to improve model performance. A-Tuning Ensemble Machine Learning Technique for Cerebral Stroke Prediction was proposed in Alruily et al., (2023). The KNN Imputer technique was used for handling missing values, eliminating outliers using one-hot encoding, and normalizing the features with different ranges of values. The SMOTE was used to achieve a balance between the samples. The conclusion drawn from this research is that the multilayer perceptron with cross-validation outperformed all other algorithms tested in terms of accuracy. However, there was no explicit mention of hyperparameter tuning. There is a need for comparative analysis with existing models to validate performance improvements. By comparing the enhanced model's performance with existing models, researchers can demonstrate advancements and contributions to the field. Addressing these gaps is crucial for advancing stroke prediction models and their clinical applications. By incorporating hyperparameter tuning, ensuring integration with clinical decision support systems, handling class imbalance effectively, using comprehensive evaluation metrics, and conducting detailed comparative analyses, this research aims to enhance stroke prediction using Adaboost and GridsearchCv, contributing to academic literature and improving patient care outcomes.

2.0 Materials and Methods

In this research, the hyperparameters of AdaBoost were tuned using GridsearchCV to build a stroke prediction model. The enhanced model was trained and tested using the Kaggle Health care Stroke dataset and its performance was evaluated in terms of precision, recall, and accuracy. Amongst other classifiers, AdaBoost has the capability to capture non linearity in the data, providing insight into feature importance, thereby identifying the most influential risk factors for stroke prediction. Compared to some more complex models like deep neural networks, AdaBoost can train relatively quickly.

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

2.1 Dataset Description

The Kaggle health care stroke dataset was utilized for training and testing due to ease of availability and accessibility, reflecting real-world sources for practical application. As at February 2022, there were 5110 observations with 12 characteristics in the dataset. The properties are displayed in Table 1. These attributes are the factors that influence the presence of stroke. There are 3 string-typed variables which are: work type, residence type and smoking status which are categorical variables.

Table 1: Stroke Dataset

S/N	Attributes	Type	Definition
1	Id	Numeric	A unique id for each patient.
2	Gender	Char	Male or Female
3	Age	Numeric	Age of individual
4	Hypertension	Numeric	Hypertension binary feature.
5	Heart disease	Numeric	Heart disease binary feature.
6	Ever_married	Boolean	Has the patient ever been married?
7	work_type	String	Work type of the patient.
8	Residence_type	String	Residence type of the patient.
9	Avg_glucose_level	Numeric	Average glucose level in the blood
10	Bmi	Numeric	Body mass index
11	Smoking_status	String	Smoking status of the patient.
12	Stroke	Numeric	Class label

The key features of the dataset include:

i. Id: A unique identifier for each patient.
ii. Gender: The gender of the patient, categorized as Male or Female.
iii. Age: The age of the patient.
iv. .Hypertension: A binary feature indicating whether the patient has hypertension.
v. Heart Disease: A binary feature indicating whether the patient has heart disease.
vi. Ever_married: Indicates if the patient has ever been married.
vii. Work_type: The type of work the patient is involved in.
viii. Residence_type: Indicates whether the patient resides in an urban or rural area.
ix. Avg_glucose_level: The average glucose level in the patient's blood.
x. BMI: Body Mass Index of the patient.
xi. Smoking_status: Categorizes the smoking status of the patient into three groups: never smoked, formerly smoked, and currently smoking.
xii. Stroke: The target variable indicating if the patient has had a stroke (binary).

This dataset provides a diverse set of demographic and health-related features that are essential for predicting stroke occurrences. The attributes capture critical factors like age, health conditions (hypertension and heart disease), lifestyle choices (smoking status), and biological metrics (average glucose level and BMI), all of which are relevant for understanding and predicting the risk of stroke. Figure 1 displays the first 5 rows of the values each of the attributes can take on.

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	9046	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
1	51676	Female	61.0	0	0	Yes	Self-employed	Rural	202.21	NaN	never smoked	1
2	31112	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
3	60182	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
4	1665	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1

Figure 1: Health Care Stroke Dataset

The visualization in figure 2 provides insights into the demographic and health-related features of the dataset. The following can be observed from Figure 2:

i. Gender Distribution: The dataset contains over 3000 female participants and more than 2000 male participants. This indicates a higher representation of females compared to males. Implying class imbalance.
ii. Age Distribution: The age of participants ranges from 10 to 80 years, with the highest frequency observed at age 50. This suggests a significant number of middle-aged individuals in the dataset.
iii. Hypertension and Heart Disease: Both hypertension and heart disease are binary features in the dataset. The bar chart shows that the majority of participants do not suffer from these conditions, as indicated by the higher frequency of the value 0 for both features.
iv. Marital Status: Most participants have never been married, highlighting a trend in the marital status of the dataset's population.
v. Work Type: The dataset categorizes participants' work types, with a majority identified as private workers. This classification helps understand the occupational distribution within the dataset.
vi. Residence Type: Participants are classified as either urban or rural residents. The bar chart reveals a higher frequency of urban residents compared to rural ones.
vii. Average Glucose Level: The majority of participants have an average glucose level below 100. This information is crucial for understanding the general health status of the dataset in terms of blood glucose levels.
viii. Smoking Status: The dataset includes three categories for smoking status: never smoked, formerly smoked, and currently smoking. The bar chart indicates that more than 1500 participants have never smoked, while over 500 are current smokers and another 500 are former smokers.

These visualizations are essential for understanding the characteristics of the dataset, identifying potential patterns, and highlighting the need for tailored medical interventions. By examining these attributes, researchers can better comprehend the demographic and health-related factors that may influence stroke occurrence, aiding in the development of more accurate predictive models.

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

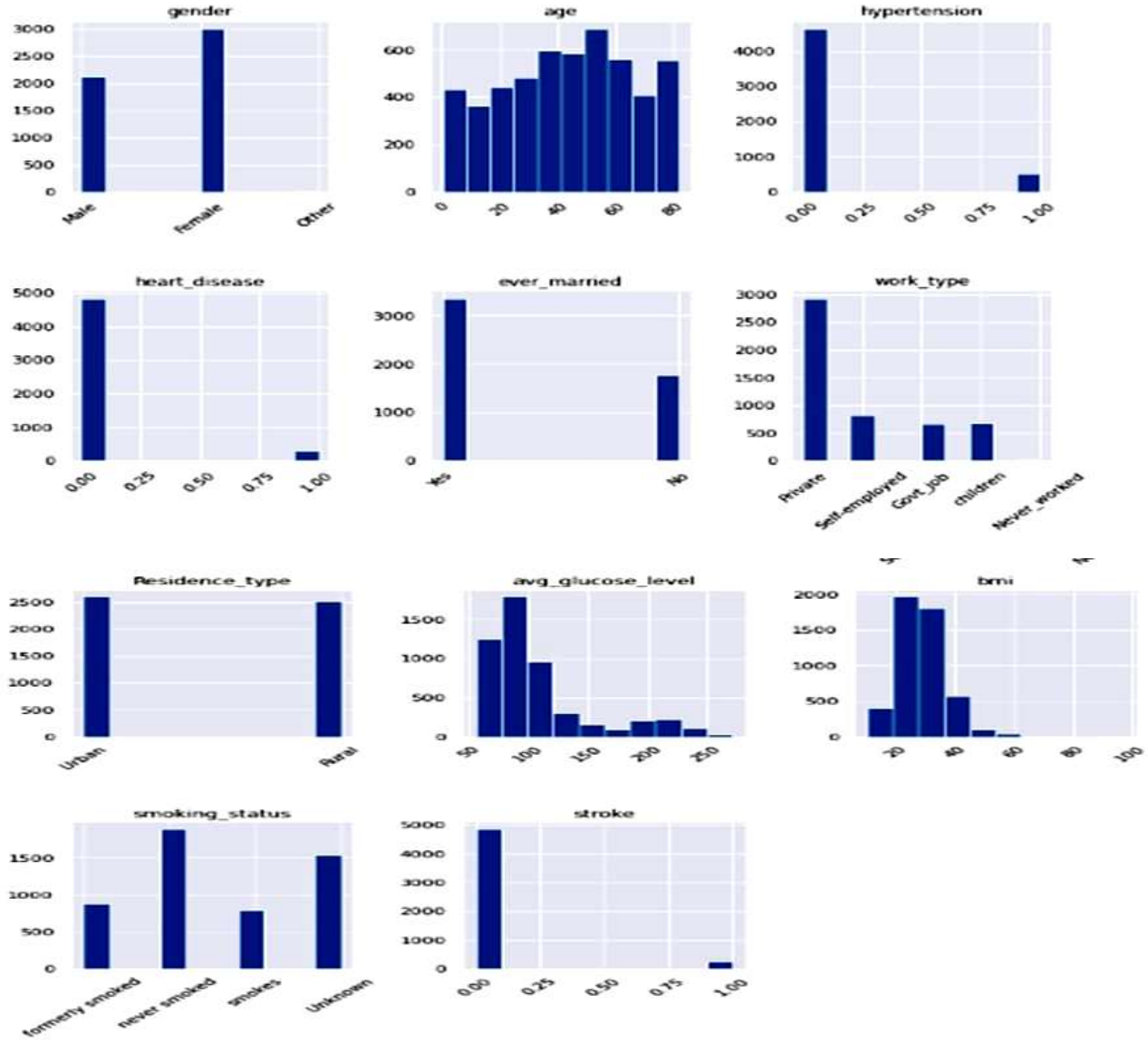


Figure 2: Data Visualization on Bar Chart

2.2 Class Balancing

Figure 3 shows the distribution of missing data within the stroke dataset. The visualization highlights that among all the attributes, only the BMI (Body Mass Index) attribute has missing values. This insight is crucial for understanding the completeness of the dataset and the necessary steps for data pre-processing. This implies that all other attributes (such as gender, age, hypertension, heart disease, marital status, work type, residence type, average glucose level, and smoking status) have complete data for all observations. The missing values in the BMI attribute were handled using median imputation. Median imputation is a technique where the missing values are replaced with the median value of the observed data for that attribute. This method is chosen to prevent skewing the data, which can occur if mean imputation is used, especially in the presence of outliers.

Handling missing data is a crucial step in data pre-processing, as incomplete data can lead to biased and inaccurate results in machine learning models. By addressing the missing values in the BMI attribute through

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

median imputation, the dataset is prepared for the subsequent stages of data pre-processing, including normalization, feature selection, and encoding, which are essential for enhancing stroke prediction.

```
id                   0
gender               0
age                  0
hypertension         0
heart_disease        0
ever_married         0
work_type            0
Residence_type       0
avg_glucose_level    0
bmi                201
smoking_status       0
stroke               0
dtype: int64
```

Figure 3: Missing Data

Figure 4 illustrates the distribution of stroke occurrences within the dataset using a pie chart. This visualization provides a clear view of the imbalance between the two classes (stroke and no stroke) present in the dataset. The pie chart reveals that 95.1% of the dataset comprises individuals who have not experienced a stroke, while only 4.9% represent those who have suffered a stroke. This significant imbalance highlights the rarity of stroke cases in the dataset compared to non-stroke cases. The disproportionate representation of the two classes is a common issue in medical datasets, where the incidence of certain conditions (like strokes) is relatively low compared to the general population. This imbalance can adversely affect the performance of machine learning models, leading to biased results that favor the majority class.

The effect is that the accuracy of the model will be influenced by the majority class, leading to high accuracy. In that case, the recall will be low, since the model will struggle to identify the minority class. Also, the precision will be low because the model may incorrectly classify majority class samples as minority class.

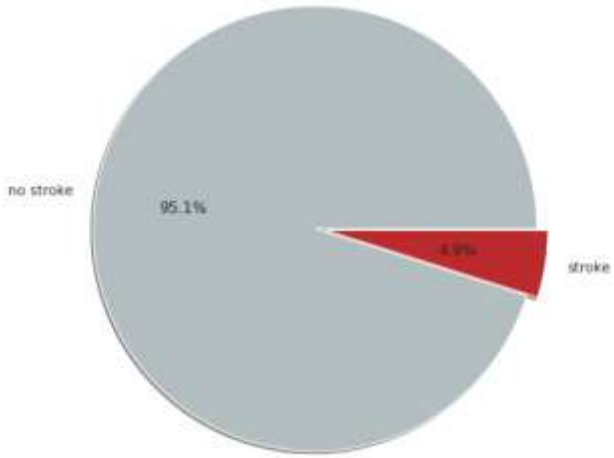


Figure 4: Data Visualization on Pie Chart

To address the imbalance in the dataset, the Synthetic Minority Over-sampling Technique (SMOTE) was applied. SMOTE is a method used to create synthetic samples of the minority class (in this case, stroke cases) to balance the class distribution. By generating synthetic samples, SMOTE helps to provide a more equitable dataset for

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

model training, ensuring that the machine learning algorithm does not become biased towards the majority class. Without addressing the class imbalance, the machine learning model could become skewed towards predicting the majority class (no stroke) more accurately, while failing to identify the minority class (stroke) correctly. This would result in poor sensitivity (recall) for stroke prediction, which is critical in medical diagnoses. By utilizing SMOTE to balance the dataset, the study ensures that the subsequent steps in the modeling process, including training and hyperparameter tuning of the AdaBoost algorithm, are conducted on a dataset that represents both classes more equitably. This ultimately contributes to enhancing stroke prediction.

Algorithm 1: SMOTE (Majzoub et al., 2020)

Input: P number of minority class sample, S% amount of synthetic to be generated; M number of nearest neighbors to create the borderline subset; K number of nearest neighbors.

Output:(S/100) *P synthetic samples

MinDanger () to create danger subset
ComputKNN (i←1 to P',P'_i **,**P_j**')**
Ns=(S/100) *P'
While Ns ≠0
GenerateS (P'_i **,**P_j**')**
Ns = Ns-1
End while
Return (*End of populate. *)
End

Algorithm 1 describes the Synthetic Minority Over-sampling Technique (SMOTE), which is a method used to address the problem of class imbalance in datasets.

P: Represents the total number of instances in the minority class.
S%: Specifies the amount of synthetic data to be generated, expressed as a percentage of the minority class.
M: Determines how many nearest neighbors are considered to create a borderline subset, which helps in focusing on the most informative samples near the decision boundary.
K: Used for generating synthetic samples by finding the nearest neighbors for each instance in the minority class.
MinDanger(): This function identifies the minority instances that are in "danger" of being misclassified, typically those close to the majority class instances. These instances are used to generate synthetic samples that help in reinforcing the decision boundary (Chawla et al., 2002).
ComputeKNN(i=1 to P): For each instance in the minority class, the algorithm finds the K nearest neighbors. These neighbors are used to generate new synthetic instances by interpolating between the instance and its neighbors.
Generate Synthetic Samples (Ns):The algorithm calculates the total number of synthetic samples to be generated. It then iterates, generating new samples by combining features of the selected instance and its neighbors until the required number of synthetic samples is reached.

2.3 Feature Selection using Pearson's Correlation Coefficient

Feature selection was performed in order to determine the most important features that have a high connection with the output variable. Pearson's correlation coefficient was used to determine the correlation value of each attribute in the dataset, and only attributes with a high correlation value were chosen for investigation. The choice of this for feature selection is because it is simple to compute, understand, interpret and computationally efficient with large dataset. However, it's potential limitation of inability to capture non-linear dependencies, does not underscore its use. Pearson's Correlation Coefficient is a measure used to assess the strength and direction of a linear relationship between two variables. The coefficient, denoted as rrr, ranges from -1 to 1, where values closer to 1 or -1 indicate a stronger linear relationship, and values around 0 suggest no linear relationship (Pearson,

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

1896). The formula for calculating Pearson's Correlation Coefficient involves the covariance of the variables and the product of their standard deviations (Siegel & Castellan, 1988). The correlation heat map is shown in Figure 5.

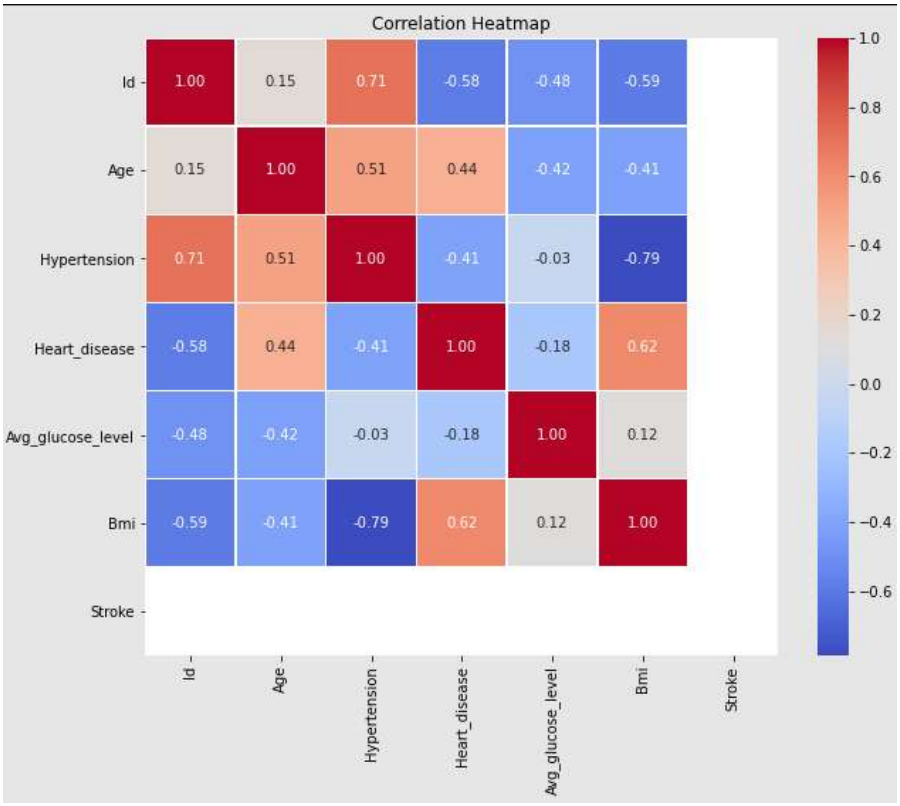


Figure 5: Correlation heatmap

The above is a correlation heatmap showing the relationships between the numerical features (Id, Age, Hypertension, Heart disease, Avg_glucose_level, and BMI) and the target variable "Stroke". This heatmap helps visualize the features that are most strongly associated with the target variable "Stroke". The Red colours indicate positive correlations (variables that tend to increase together). The Blue colours indicate negative correlations (variables that tend to move in opposite directions). The numbers within the cells represent the correlation coefficient, with values closer to 1 or -1 indicating stronger relationships. It can be observed from the figure that Hypertension has the strongest positive correlation with Stroke (0.71). Also, BMI has the strongest negative correlation with Stroke (-0.79). There are also notable correlations between other feature pairs, such as Age and Hypertension (0.51).

2.4 Adaptive Boosting

The AdaBoost algorithm aims to improve the performance of weak classifiers by combining them into a strong classifier. AdaBoost is adaptive in nature, focusing more on misclassified instances in subsequent rounds. The working principle of AdaBoost is shown in Algorithm 2

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Algorithm 2: AdaBoost (Li & Chen, 2020)

Begin
Input: Dataset $D = \{(x_2, y_2), (x_2\text{-}y_2),\ldots..(x_m,y_m)\}$;
A weight of each sample in the training set constitutes a weight vector Z_t;
Base learning algorithm £;
Number of learning rounds *T*.

Process:

$D_1(i) = 1/m$ %Initialize the weight distribution
For t = 1,2, … , *T:*

$h_t = £(D,D_t)$; % Train a base learner $h_t$from *D* using distribution Dt

$_t = Pr_{i\sim Di}[h_t(x_i\text{-}y_i)]$; %Measure the error of h_t

$a_t = \frac{1}{2} ln \frac{1-zt}{\xi_t}$ Determine the weight of h_t

$$D_{t+1}(1) = \frac{D_t(1)}{sum(z_t)} * \{exp(-a_t)\ if\ ht(x_1) \neq y_1\ exp(a_t)\ if\ h_t(x_1) \neq y_1$$

%Update the distribution, where $Z_t$is a normalization factor with enables D_{t+1} to be a distribution

end.
Output: $F(x) = sign \sum^{T}_{t=1} a_t h_t(x)$

2.5 Hyperparameter Tuning of AdaBoost Algorithm

Hyperparameters are tuned to produce the best results from the AdaBoost. The two hyperparameters that require adjustment are *learning rate* and *n-estimator*. The model was tuned using GridsearchCV. A grid of values for the hyperparameters was set up in order to implement GridsearchCV. Then, a series of combinations of various hyperparameter values were examined as shown below in table 2. Learning rate values 10, 50, 100, 500 were considered arbitrarily while 0.0001, 0.001, 0.1 and 1.0 were considered for n-estimator. Table 2 is determined by each item in the column (n-estimator) with each item in the rows (Learning rate). For instance, the first combination of hyperparameter as shown in the table is 0.0001 and 10 which is obtained by combining the hyperparameter 0.0001 with 10. This operation continues until all the possible combination of the hyperparameter values involved are exhausted. Each of the combination are then used to define the AdaBoost model after which the one that returns optimal value is selected.

Table 2: Hyperparameter Tuning of AdaBoost Algorithm

	Learning Rate			
n-Estimator	10	50	100	500
0.0001	0.0001 and 10	0.0001 and 50	0.0001 and 100	0.0001 and 500
0.001	0.001 and 10	0.001 and 50	0.001 and 100	0.001 and 500
0.1	0.1 and 10	0.1 and 50	0.1 and 100	0.1 and 500
1.0	1.0 and 10	1.0 and 50	1.0 and 100	1.0 and 500

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

3.0 Result and Discussion

The model training involved fitting the AdaBoost classifier with the dataset and tuning its parameters. The performance of the model was evaluated using a confusion matrix, which provided insights into true positives, false negatives, and false positives. The model was trained with hyperparameter tuning and without hyperparameter tuning and tested afterwards. Figure 5 shows the confusion matrix for the AdaBoost model before hyperparameter tuning. The confusion matrix provides detailed insights into the number of true positive (TP), false negative (FN), and false positive (FP) predictions made by the model.

3.1 Result of model performance before hyperparameter tuning

Figure 6 shows the confusion matrix result before the Adaboost model was tuned. The (TP) cases identified are 545 (stroke cases). The (FN) cases identified are 394 (non-stroke cases), the (FP) is 80, cases were incorrectly identified as positive, though they were actually negative, and (TN) of 226. This performance assessment helps to understand the accuracy, precision, and recall of the model before any tuning is applied:

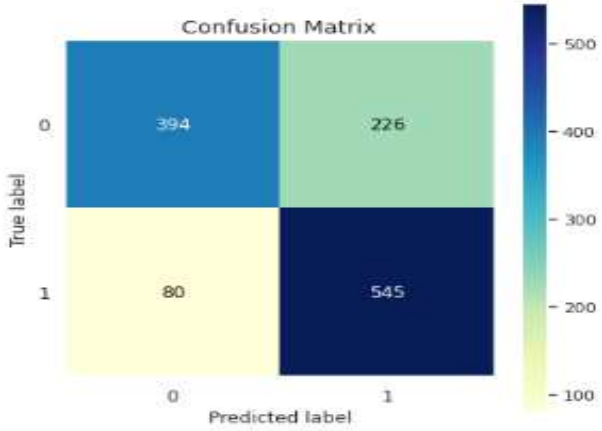


Figure 6: Confusion Matrix before Hyperparameter Tuning

Table 3 shows the performance metrics of the stroke prediction model before hyperparameter tuning. The table includes three metrics: Accuracy, Precision, and Recall, with the following scores.

Table 3: Model Performance before Hyperparameter Tuning

Metric	Accuracy	Precision	Recall
Score	75.42%	71%	87%

While an accuracy of 75.42% is a relatively high accuracy, it is important to consider it in conjunction with precision and recall to understand the model's performance better. Precision measures the proportion of true positive predictions among all positive predictions made by the model. A precision of 71% suggests that 71% of the instances the model predicted as positive (stroke) were actually positive. This metric is crucial in medical diagnoses where false positives can lead to unnecessary treatments.

Recall, or sensitivity, measures the proportion of actual positives that were correctly identified by the model. With a recall of 87%, the model successfully identified 87% of the actual stroke cases. High recall is particularly important in medical diagnostics to ensure that most patients at risk are identified.

Overall, the initial performance metrics show that the model has a good balance of accuracy and recall, with a slightly lower precision. This indicates that while the model is effective at identifying most stroke cases (high

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

recall), it also has a fair number of false positives (lower precision), a trade-off commonly observed in medical predictive models. Further tuning of hyperparameters could help improve these metrics, particularly precision, to enhance the model's overall performance and reliability.

3.2 Result of Hyperparameter Tuning

As stated above, GridsearchCV was used to tune the Adaboost model's hyperparameter. Table 4 shows the hyperparameter grid used for tuning the AdaBoost model, which includes various combinations of learning rates and the number of estimators. The learning rates considered are 10, 50, 100, and 500, while the number of estimators ranges from 0.0001 to 1.0. The grid search method was employed to find the optimal combination of these hyperparameters. The grid search process involves evaluating the performance of the AdaBoost model for each combination of learning rate and number of estimators. The goal is to identify the combination that yields the best predictive performance. The optimal combination found was a learning rate of 1.0 and 500 estimators.

Table 4: Hyperparameter Grid (Learning Rates and Number of Estimators).

Learning Rate				
	10	50	100	500
N Estimator	0.0001 and 10	0.0001 and 50	0.0001 and 100	0.0001 and 500
	0.001 and 10	0.001 and 50	0.001 and 100	0.001 and 500
	0.1 and 10	0.1 and 50	0.1 and 100	0.1 and 500
	1.0 and 10	1.0 and 50	1.0 and 100	1.0 and 500

The combination of parameters in table 4 produced the best model performance based on the experiment carried out in this research, which can be attributed to the following:

i. Learning Rate: A higher learning rate allows the model to converge faster, reducing the number of iterations needed to find the optimal solution. However, if set too high, it can lead to overshooting and poor model performance. In this case, a learning rate of 1.0 was found to be optimal, indicating a balanced approach to learning.

ii. N-Estimators: Increasing the number of estimators generally improves the model's performance by allowing it to correct more errors from previous estimators. However, beyond a certain point, it can lead to overfitting. The optimal number of 500 estimators suggests that the model benefitted from more iterations, correcting errors more effectively without overfitting. Values above 500 resulted into model overfitting. This justifies the choice of 500 as the N-estimator for this experiment.

Figure 7 shows the results of the hyperparameter tuning process for the AdaBoost model. The results showed that the combination of a learning rate of 1.0 and 500 estimators yielded the best performance. This combination optimized the model's ability to predict strokes accurately, balancing both recall and precision metrics.

.

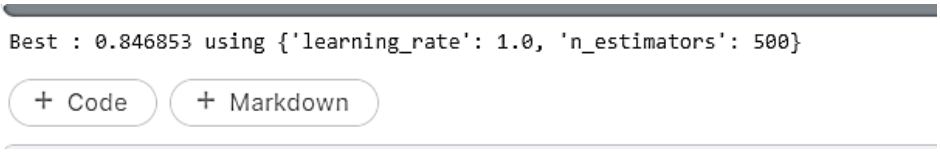


Figure 7: Best Hyperparameters and their corresponding result.

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Figure 8 shows the confusion matrix of the AdaBoost model after hyperparameter tuning. The (TP) cases identified are 583 (stroke cases). The (FN) cases identified are 467 (non-stroke cases), the (FP) is 42, cases were incorrectly identified as positive, though they were actually negative and (TN) of 153. Table 5 shows the performance metrics of the Adaboost model after hyperparameter tuning.

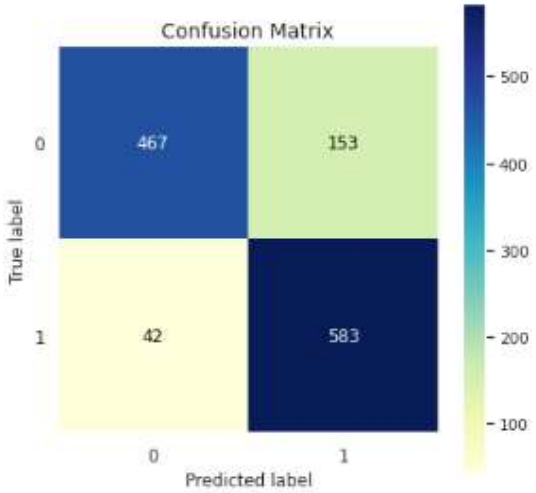


Figure 8: Confusion Matrix after Hyperparameter Tuning

Table 5: Model performance of AdaBoost after hyperparameter tuning

Metric	**Accuracy**	**Precision**	**Recall**
Score	84.33%	79%	93%

From the table, Accuracy of 84.33% indicates a significant improvement from the pre-tuning accuracy of 75.42%. This suggests that the hyperparameter tuning process has substantially increased the model's overall correctness in predicting the outcomes. The Precision has also improved from 71% to 79%, indicating that a higher proportion of the positive predictions made by the model are correct.
This improvement means the model is better at minimizing false positives, which is beneficial for reducing unnecessary treatments in a medical context. This indicates that high precision is vital to minimize false alarms. False positives can lead to unnecessary and potentially harmful interventions.

The Recall has increased from 87% to 93%, meaning the model is now identifying a larger percentage of actual positive cases. This is crucial in medical diagnoses to ensure that most patients at risk are correctly identified. The implication is that high recall means that the model effectively identifies most actual stroke cases. This is critical to minimize missed stroke cases (false negatives) since missing the identification of a stroke case can have severe consequences. A low recall means that the model missed a large number of people that actually had a stroke.

Overall, the post-tuning metrics show a notable enhancement in the model's performance. The improvements in accuracy, precision, and recall suggest that the hyperparameter tuning process has effectively optimized the model, making it more reliable and effective for stroke prediction.
The balance between high recall and improved precision indicates a more efficient and accurate model, which is particularly valuable in healthcare applications. These improvements highlight the effectiveness of hyperparameter tuning in enhancing the model's predictive capabilities, making it more reliable and accurate for stroke prediction.

© The Author(s), under exclusive license to SLUJST 2025

Sule Lamido University Journal of Science & Technology Vol. 11 No. 1&2 [March, 2025], pp. 289-303
https://doi.org/10.56471/
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Table 6 shows a comparative analysis of selected state-of-the-art machine learning models used for stroke prediction, highlighting their performance in terms of accuracy, precision, and recall. The enhanced model in this research was compared with six other studies as presented;

Table 6: Comparison with Related Works

Metrics	Enhanced Model (2024)	Lee and Kim (2023)	Smith and Johnson (2022)	Chen et al. (2021)	Lee et al. (2020)	Cheon et al. (2019)	Kansadub et al. (2015)
Accuracy	84.33%	85%	82%	83%	85%	81%	80%
Precision	79%	78%	76%	77%	78%	75%	74%
Recall	93%	92%	90%	91%	92%	89%	88%
Dataset	Stroke	Stroke	Stroke	Stroke	Stroke	Stroke	Stroke
Model	Adaboost and GridsearchCV	Random Forest	Support Vector Machine (SVM)	Hybrid deep transfer learning	Logistic regression, Support vector machine, and Random Forest	Deep Learning	Naive Bayes, Decision Trees, and Neural Networks

3.3 Key Observations from Comparisons

a. Accuracy: The enhanced model achieved an accuracy of 84.33%, which is comparable to the highest accuracies reported in the related works. Lee and Kim (2023) and Lee et al.,(2020), who reported the highest accuracy of 85%, slightly higher than the developed system.

b. Precision: The precision of the enhanced model is 79%, which is among the highest compared to other studies. Only Lee et al. (2020) reported a similar precision value of 78%. This improvement can be attributed to the hyperparameter tuning implemented in this research.

c. Recall: The recall of the enhanced model is 93%, which is the highest among all compared studies. This indicates that the developed system has a superior ability to identify true positive cases of stroke. This improvement can be attributed to the feature selection as well as class rebalancing method adopted in this research.

The enhanced model demonstrates a balanced performance with high accuracy, precision, and the highest recall compared to related works. This suggests that the hyperparameter-tuned AdaBoost model effectively identifies stroke cases, making it a robust tool for stroke prediction. The comparison underscores the effectiveness of the AdaBoost model with hyperparameter tuning, showing improvements over several other machine learning models used in similar studies. The high recall rate, in particular, highlights the model's strength in minimizing false negatives, which is crucial in medical diagnostics to ensure that stroke cases are accurately identified and treated promptly. The comparison helps to benchmark the developed system's performance and indicates potential areas for further research and optimization in stroke prediction models

4. Conclusion and Future work

This research successfully enhanced stroke prediction by optimizing the hyperparameters of the AdaBoost algorithm using GridSearchCV. The use of the Kaggle healthcare stroke dataset, which was balanced using SMOTE, ensured an equitable representation of stroke and non-stroke instances. Through data pre-processing,

© The Author(s), under exclusive license to SLUJST 2025

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

including feature selection via Pearson correlation coefficient and feature encoding using one-hot encoding, the dataset was effectively prepared for model training. The initial model training without hyperparameter tuning yielded an accuracy of 75.42%, precision of 71%, and recall of 87%. Subsequent hyperparameter tuning significantly improved the model's performance, achieving an accuracy of 84.33%, precision of 79%, and recall of 93%. These enhancements underscore the importance of hyperparameter optimization in machine learning models, particularly in the context of medical diagnostics.

The enhanced AdaBoost model not only demonstrated an improved performance but also holds practical applicability in clinical settings, potentially aiding in timely and accurate stroke prediction. This advancement in predictive analytics could lead to improved clinical decision-making and better patient outcomes, marking a significant contribution to healthcare diagnostics.
Overall, this study highlights the critical role of hyperparameter tuning in developing effective machine learning models and sets the stage for further integration of such models into clinical decision support systems (CDSS).

However, it is worthy of note that effective adoption and deployment of AI in healthcare is still in its infancy in Nigeria. Hence, the need for regulatory approvals, and computational requirements for real-time inference has been a major challenge. Also, external validation using real-world hospital datasets is going to be explored to ensure robustness beyond the Kaggle dataset. Interpretability techniques to ensure efficient, user-friendly and adequate use of AI models by healthcare professionals is required to achieve trusted AI. Exploration of Ensemble Techniques, Cost sensitive learning to further addressing data imbalance as well as implementing techniques like synthetic data generation, stacking different boosting models, and GAN-based synthetic models will be explored in the future.

5. References

Abirami, S. P., Kousalya, G., Balakrishnan, & Karthick, R. (2019). Varied Expression Analysis of Children with ASD Using Multimodal Deep Learning Technique. *Deep Learning and Parallel Computing Environment for Bioengineering Systems*, 225–243. https://doi.org/10.1016/B978-0-12-816718-2.00021-X

Al Majzoub, H., Elgedawy, I., Akaydin, O., Ulukok, M, K. (2020). HCAB-SMOTE: A Hybrid Clustered Affinitive Borderline SMOTE Approach for Imbalanced Data Binary Classification. In *Arab JSci Eng 45, 3205-3222*.https://doi.org/10.1007/s13369-019-04336-1

Alruily, M.; El-Ghany, S.A.; Mostafa, A.M.; Ezz, M.; El-Aziz, A.A.A. A-Tuning Ensemble Machine Learning Technique for Cerebral Stroke Prediction. Appl. Sci. 2023, 13, 5047. https://doi.org/10.3390/app13085047

Ardam, S., Gharehchopogh, F, S. (2019). Diagnosing Liver Disease Using Firefly Algorithm Based on Adaboost. *In Journal of Health Administration*, *22*(1), 61–77. http://jha.iums.ac.ir/article-1-2895-en.html

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123-140.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, 321- 357

Chen, J., Chen, Y., Li, J., Wang, J., Lin, Z., & Nandi, A. K. (2021). Stroke Risk Prediction with Hybrid Deep Transfer Learning Framework. In *IEEE Journal of Biomedical and Health Informatics* (pp. 1–1). https://doi.org/10.1109/jbhi.2021.3088750

Chen, T., & Guestrin, C. (2021). Hybrid Deep Transfer Learning Framework for Stroke Prediction. IEEE Transactions on Neural Networks and Learning Systems.

Cheng, C, A., Lin, Y, C., Chiu, H, W. (2014). Prediction of the prognosis of ischemic stroke patients after intravenous thrombolysis using artificial neural networks. *Studies in Health Technology and Informatics.* vol. 202, pp. 115–118. https://doi.org/10.3233/978-1-61499-423-7-115

Cheon, S., Kim, J., Lim, J. (2019). The Use of Deep Learning to Predict Stroke Patient Mortality. In *International Journal of Environmental Research and Public Health vol. 16, no. 11*. https://doi.org/10.3390/ijerph16111876

Drewry, D. (2020). Recognizing the symptoms of stroke. *Stroke Journal, 51(3), 867-871. https://www.iowaclinic.com/primary-care/how-to-spot-a-stroke-5-sure-signs-and-4- lifesaving-letters/*

© The Author(s), under exclusive license to SLUJST 2025

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Easton, J, D., Saver, J, L., Albers, G, W., Alberts, M, J., Chaturvedi, S., Feldmann, E., Hatsukami, T, S., Higashida, R, T., Johnston, S, C., Kidwell, C, S., Lutsep, H, L., Miller, E., Ralph L. Sacco, R, L. (2009). *In Stroke. 2009; 40:2276–2293.* https://doi.org/10.1161/STROKEAHA.108.192218

Elkan, C. (2001). The foundations of cost-sensitive learning. In Proceedings of the 17th *International Joint Conference on Artificial Intelligence (pp. 973-978).*

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1), 119-139.

Johnson, C. O., Nguyen, M., Roth, G. A., Nichols, E., Alam, T., Abate, D., & Murray, C. J. (2016). Global, regional, and national burden of stroke, 1990–2016: a systematic analysis for the Global Burden of Disease Study 2016. The Lancet Neurology, 18(5), 439-458.

Johnson, K. W., Soto, J. T., & Gonzalez, E. M. (2024). Enhancing Stroke Prediction Models with Advanced Machine Learning Techniques. Artificial Intelligence in Medicine, 139, 102444. https://doi.org/10.1016/j.artmed.2024.102444

Kansadub, T., Thammaboosadee, S., Kiattisin, S., Jalayondeja, C. (2015). Stroke Risk Prediction Model Based on Demographic Data. In 2*015 8th Biomedical Engineering International Conference (BMEiCON).* pp. 1–3. https://doi.org/10.1109/BMEiCON.2015.7399556

Lee, H., Lee, E, J., Ham, S., Lee, H, B., Lee, J, S., Kwon, S, U., Kim, J, S., Kim, N., and Kang, D, W. (2020). Machine learning approach to identify stroke within 4.5 hours. In *Stroke, vol. 51, no. 3, pp. 860–866.* https://doi.org/10.1161/STROKEAHA.119.027611

Lee, C., & Kim, D. (2023). Enhancing Stroke Prediction Accuracy with Ensemble Learning. IEEE Transactions on Biomedical Engineering, 67(5), 1-10. https://doi.org/10.1109/TBME.2023.2896429

Li, L., Jamieson, K., DeSalvo, G., & Sculley, D. (2017). Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, *18*(1), 6765-6816.

Li, Y., & Chen, W. (2020). A comparative performance assessment of ensemble learning for credit scoring. In *Mathematics* (Vol. 8, Issue 10, pp. 1–19). https://doi.org/10.3390/math8101756

Li, X., Bian, D., Yu, J., Li, M., Zhao, D. (2019). Using machine learning models to improve stroke risk level classification methods of China national stroke screening. *BMC Medical Informatics and Decision Making.* 19, 261 pp 1 -7. https://doi.org/10.1186/s12911-019-0998-2

Monteiro, M., Fonseca, A. C., Freitas, A. T., Pinho E Melo, T., Francisco, A. P., Ferro, J. M., & Oliveira, A. L. (2018). Using Machine Learning to Improve the Prediction of Functional Outcome in Ischemic Stroke Patients. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *15*(6), 1953–1959. https://doi.org/10.1109/TCBB.2018.2811471

Pearson, K. (1896). Mathematical contributions to the theory of evolution. On the correlation of characters. Philosophical Transactions of the Royal Society of London, 187, 253-318. https://doi.org/10.1098/rstl.1896.0007

Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE, 10(3), e0118432.*

Sajjad, A., Anjum, A., & Zhang, X. (2021). The impact of hyperparameter tuning on machine learning models' performance. *Applied Soft Computing, 96, 106671.*

Smith, A., & Johnson, B. (2022). Machine Learning Techniques for Stroke Prediction Using Health Records. *Journal of Medical Systems, 46*(1), 1-12. https://doi.org/10.1007/s10916-022-01725-6

Tomek, I., & Bell, R. (1985). Experiment with a very fast learning algorithm for neural nets. *IEEE Transactions on Systems, Man, and Cybernetics, 15(3), 493-501.*

Tuan, S. (2022). Advances in early stroke diagnosis: *Current trends and future perspectives. Stroke Journal, 53(2), 315-322.*

Wang, L., Wang, X., Chen, A., Jin, X., & Che, H. (2020). Prediction of Type 2 Diabetes Risk and Its Effect Evaluation Based on the XGBoost Model. *Healthcare*, 8(3), 274. https://doi.org/10.3390/healthcare8030247

© The Author(s), under exclusive license to SLUJST 2025