**A Multi-Algorithmic Approach to Stroke Risk Prediction Using Machine Learning**

**Abstract**

Stroke remains a major public health concern and a leading cause of death and long-term disability worldwide. Early prediction of stroke risk can greatly improve preventive care and patient outcomes. This study employed a retrospective analysis using the Kaggle Stroke Prediction Dataset to develop effective machine learning models for stroke prediction. The dataset underwent thorough preprocessing, including data cleaning, transformation of categorical variables into numerical format, and exploratory data analysis. Feature selection was performed using the ANOVA F-test and Chi-squared test to identify the most significant predictors. To address the class imbalance problem, the Synthetic Minority Over-sampling Technique (SMOTE) was applied. Four classification algorithms, Logistic Regression, Decision Tree, Random Forest, and XGBoost were trained and evaluated using performance metrics such as accuracy, precision, recall, and F1-score. Hyperparameter tuning was conducted using Grid Search to optimize each model. Among the models, Random Forest and XGBoost achieved the highest accuracy of 91%, outperforming the others. The findings underscore the effectiveness of ensemble learning methods in predicting stroke risk and demonstrate the potential of integrating machine learning into healthcare systems for early detection and improved clinical decision-making.

Keywords: Stroke Prediction; Machine Learning; Random Forest; XGBoost; SMOTE; Feature Selection; Logistic Regression; Medical Diagnosis; Class Imbalance

1.0 Introduction

Stroke remains one of the foremost causes of disability and death worldwide. According to the World Health Organization (WHO), approximately 17.9 million people lose their lives each year to cardiovascular diseases (CVD), with strokes and heart attacks accounting for nearly 80% of these fatalities. This trend is on the rise, highlighting the urgent need for early detection and prevention strategies. A stroke occurs when the blood supply to the brain is disrupted either through a blockage (ischemic stroke) or the rupture of a weakened blood vessel (hemorrhagic stroke). Common signs include weakness in the muscles, numbness, difficulty speaking, vision problems, and in more severe cases, loss of consciousness [1].

There are two major types of strokes: ischemic, caused by blood clots obstructing vessels, and hemorrhagic, which results from a ruptured blood vessel bleeding into the brain [14,15]. Preventive measures such as avoiding smoking and excessive alcohol consumption, maintaining a healthy body mass index (BMI), managing blood glucose levels, and ensuring good cardiovascular and renal health play an essential role in reducing stroke risk. The ability to predict strokes in advance is crucial to minimizing long-term damage or mortality [2,13].

In recent years, advances in machine learning have enabled the development of intelligent systems capable of predicting stroke occurrences with considerable accuracy. These algorithms support data-driven decision-making in clinical settings, offering timely insights that can guide preventive care [3,12]. Despite the critical importance of early stroke prediction, a review of existing literature has revealed that relatively few studies have applied machine learning models to this domain. This limited exploration presents a significant research opportunity to improve stroke detection using data-driven approaches.

The key contributions of this study are as follows:

* We developed a robust predictive pipeline for identifying individuals at risk of stroke.
* We conducted a comparative analysis of four widely used classification algorithms.
* We integrated statistical techniques for effective feature selection to enhance model performance.

## Dataset

The dataset used in this study was curated and released by *Fedesoriano* on the Kaggle platform. It comprised of a total of 5,110 instances and included 12 features relevant to stroke prediction. Among these, the categorical variables include gender, heart-disease, hypertension, ever-married, work-type, residence-type, smoking-status, and the stroke outcome, which serves as the target variable. The continuous variables are id (a unique identifier for each record), age, average glucose level, and body mass index (BMI). The dataset integrates a mix of demographic, lifestyle, and health-related attributes, providing a rich foundation for training and evaluating machine learning models in the context of stroke risk prediction.

**2.0 Related Literature**

Stroke is a complex and life-threatening condition that has drawn considerable attention in recent years due to its rising global incidence and significant burden on healthcare systems. Although numerous risk factors have been identified including hypertension, heart disease, diabetes, and lifestyle habits, the ability to predict stroke events remains a crucial area of study. With the increasing availability of healthcare data, researchers have turned to machine learning techniques to improve predictive accuracy and support early intervention.

Several studies have proposed the application of machine learning algorithms for stroke prediction, yet this field remains underexplored compared to other cardiovascular conditions. [4] developed a model incorporating Decision Tree, Naïve Bayes, and Artificial Neural Networks to predict stroke risk using patient data. Their study emphasized the importance of data preprocessing steps, such as handling missing values and label encoding, and highlighted the potential of Naïve Bayes as the most effective classifier in their comparison.

Another study,[3] employed six machine learning classifiers including Logistic Regression, Support Vector Machines, K-Nearest Neighbors, and Random Forests on a dataset extracted from Kaggle. Their results indicated that Naïve Bayes achieved the highest accuracy of 82%, outperforming other models in terms of precision, recall, and F1-score. Their work also drew attention to the need for handling class imbalance, a challenge that was addressed using under sampling techniques.

Deep learning methods to detect critical findings from head CT scans were explored, demonstrating how medical imaging can be integrated with machine learning for stroke diagnosis[5]. However, they noted that image-based approaches often require greater computational resources and extensive labeled datasets, making them less feasible for widespread use in resource-limited settings. In reviewing existing literature, it was evident that most models rely on structured clinical data and emphasize classification accuracy. However, many approaches face limitations such as poor generalizability, lack of real-time deployment, or restricted scope to specific stroke types. Furthermore, while some research explored advanced models, relatively few studies rigorously compare multiple classifiers using standardized preprocessing pipelines and statistical feature selection.

Our work has addressed this gap by developing a multi-model framework that integrates Decision Tree, Logistic Regression, Random Forest, and XGBOOST classifiers. Unlike previous studies that often overlook feature selection, we apply both ANOVA F-test and Chi-square tests to select meaningful predictors. In addition, we use SMOTE to tackle data imbalance, an issue commonly ignored in earlier works. By incorporating grid search for hyperparameter tuning and evaluating performance through precision, recall, F1-score, and accuracy, this study contributes a comprehensive and methodologically sound approach to stroke risk prediction.

3.0 **Materials and Method**

Our study has followed a three-phase approach, beginning with data preprocessing, followed by feature engineering and data splitting, and concluding with model development and performance evaluation, as depicted in Fig. 1. During the preprocessing phase, we observed that the BMI variable had some missing entries. These gaps were addressed by calculating the average BMI across the dataset and using that value to fill in the missing data points.

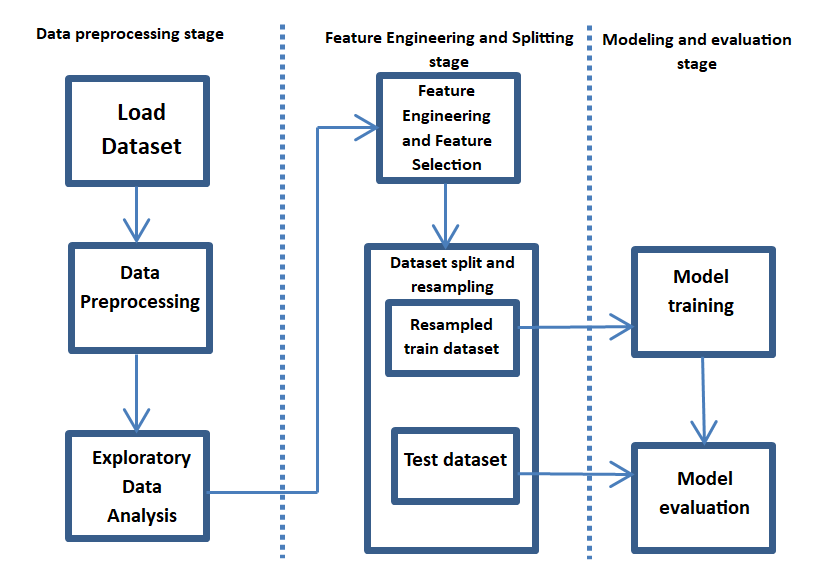


Fig. 1: Methodology overview of multi-algorithmic framework for Stroke Prediction

As part of our exploratory data analysis, we examined the distribution of the target variable to assess whether the dataset is balanced. Figure 2 reveals a significant class imbalance: 95.1% of the instances (4,861 out of 5,110) correspond to the "no stroke" class, while only 4.9% (249 out of 5,110) represent cases of stroke. Additionally, we analyzed the distribution of selected features in relation to the target variable to gain deeper insights. We explored the relationship between stroke occurrence and features such as heart disease status and hypertension status.

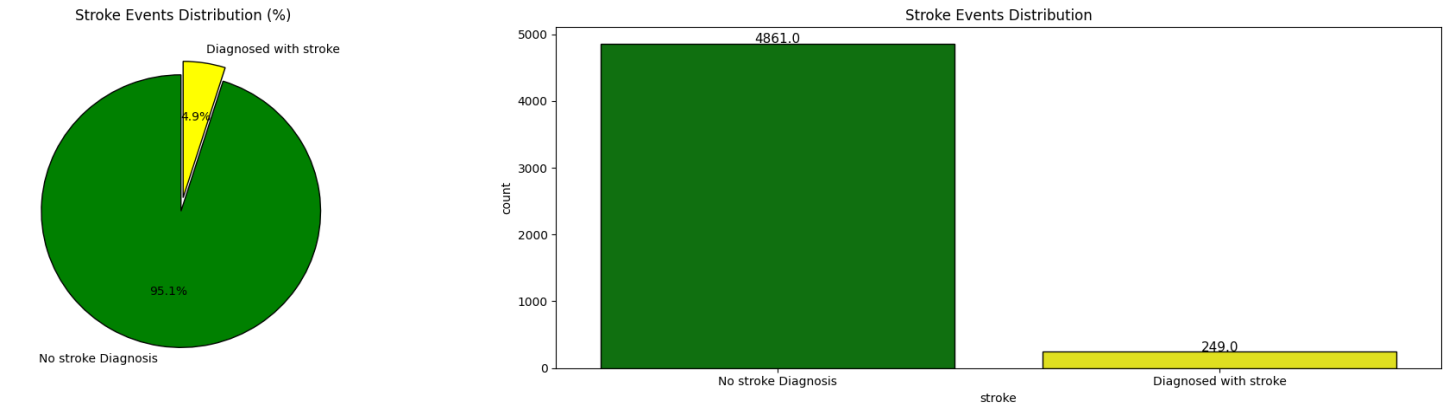


Fig. 2: Class distribution of the target variable

As shown in Fig. 3, the relationship between stroke occurrence and the presence of hypertension and heart disease. Among individuals without hypertension, 96% did not experience a stroke, while 4% did. In contrast, 86.7% of those with hypertension had no stroke, whereas 13.3% experienced one. A similar trend was observed with heart disease status: 96.8% of individuals without heart disease did not have a stroke, and 3.2% did. However, among those with heart disease, 83.0% had no stroke, while 17.0% experienced a stroke. These findings suggest that both hypertension and heart disease are associated with an increased likelihood of stroke in the dataset.

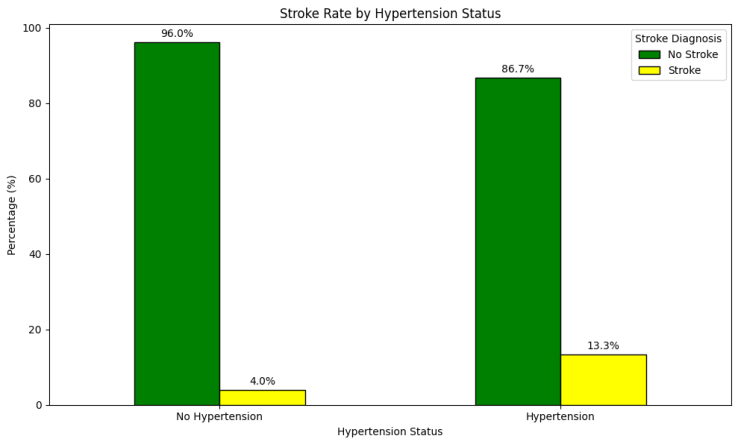
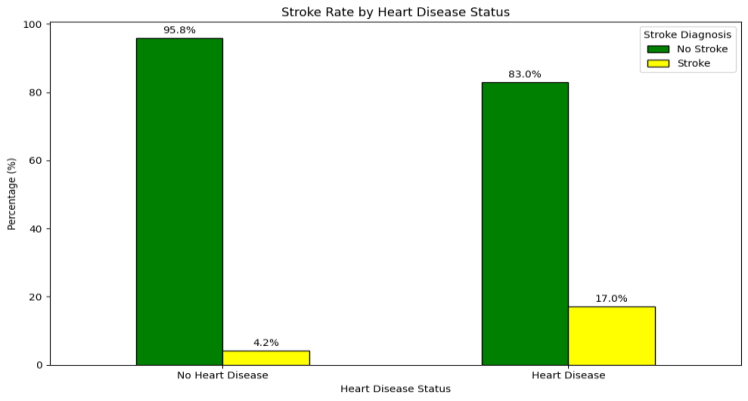


Fig. 3: Relationship between stroke occurrence and the presence of hypertension and heart disease

## Feature Engineering and Splitting stage

In the feature engineering and data splitting stage, we carried out several key steps including feature transformation, feature selection, data partitioning, and resampling to address class imbalance. To prepare categorical variables like gender, smoking status, work type, residence type, and marital status for compatibility with machine learning algorithms, we applied label encoding to convert them into numeric form. For feature selection, we employed statistical methods tailored to data type: the Chi-square test was used for categorical features, while the ANOVA F-score was applied to continuous variables, as illustrated in Fig. 4. Notably, our analysis indicated that gender did not have a significant impact on predicting stroke outcomes.

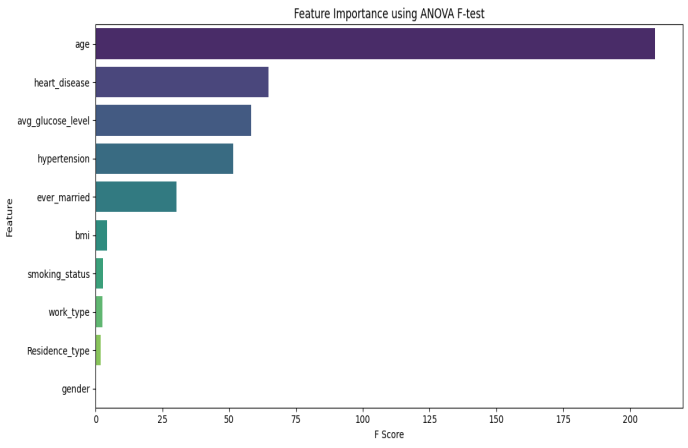
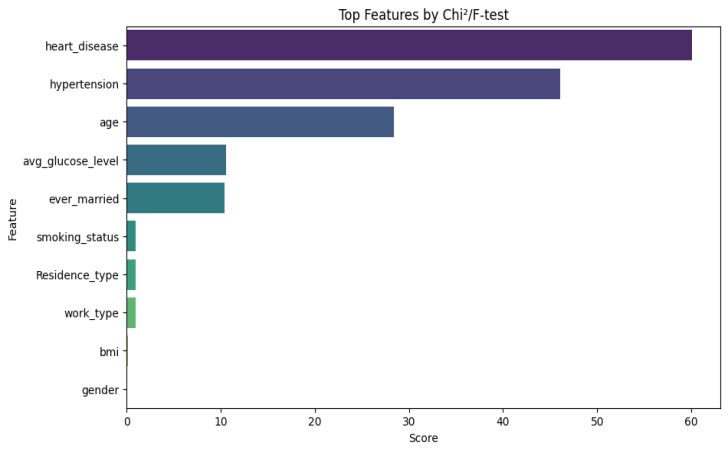


Fig. 4: Feature selection using Chi-square and ANOVA F-test

Following feature selection, we split the dataset into 70% training and 30% testing subsets. After the split, the training set consisted of 3,417 instances labeled "no stroke" and only 160 instances labeled "stroke," highlighting a pronounced class imbalance. To mitigate this, we applied the Synthetic Minority Over-sampling Technique (SMOTE) to the training data, generating a balanced dataset prior to model training, as shown in Fig. 5.

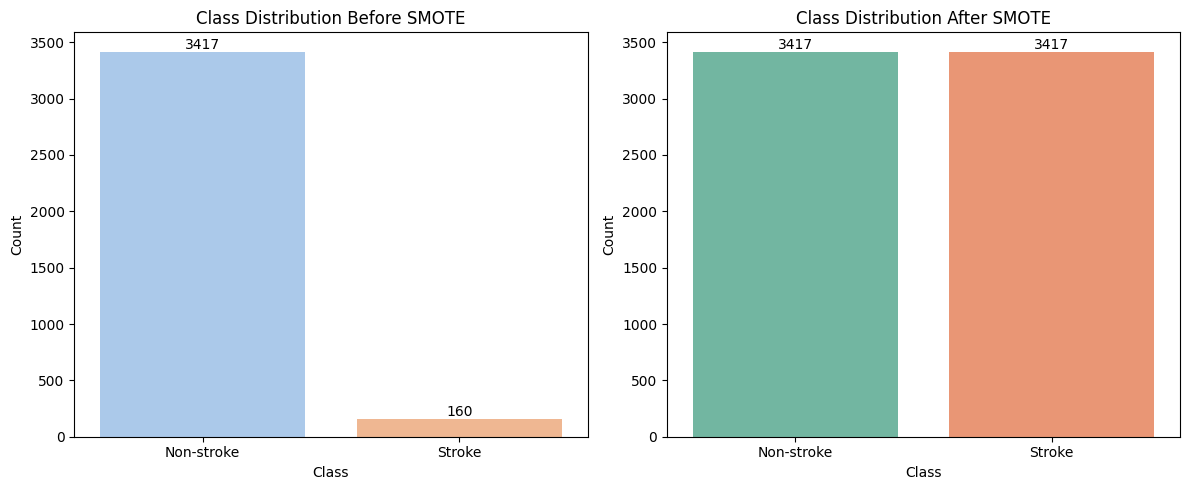


Fig. 5: Class balancing using SMOTE

## Modelling and Evaluation Stage

### Decision Tree Classifier

A decision tree typically comprises a root node, several internal (nonterminal) nodes, and terminal (leaf) nodes. The root and internal nodes together form the decision-making structure of the tree, while the terminal nodes correspond to the final classification outcomes as shown in Fig. 6. The root node encompasses the full range of possible classes for the dataset. Nodes positioned at the same depth in the tree are collectively referred to as a layer. Each node is defined by the subset of classes it aims to differentiate, the features used for discrimination, and the decision rule that governs the classification process [6].

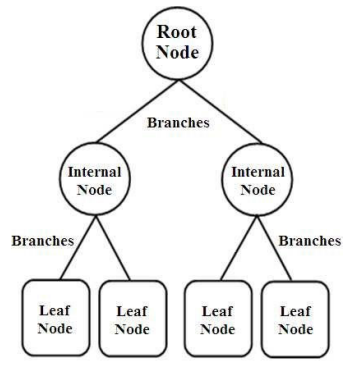


Fig. 6: Decision tree example [7].

During the model training phase using the Decision Tree algorithm, we applied hyperparameter optimization to enhance performance. This was achieved through a grid search strategy, where we systematically explored combinations of key hyperparameters. The parameters considered in the search included the splitting criterion, maximum tree depth, minimum number of samples required to split an internal node, minimum number of samples required at a leaf node, and the maximum number of features considered for splitting. The search space for these parameters is outlined in Table 1. Based on this tuning process, the optimal configuration identified was as follows: criterion = entropy, max\_depth = 20, min\_samples\_split = 2, min\_samples\_leaf = 1, and max\_features = None.

Table 1: Hyperparameter search space for Decision Tree classifier

|  |  |  |  |
| --- | --- | --- | --- |
| SN | Hyperparameter | Search Space | Best Parameter |
| 1 | Criterion | Gini,entropy | Entropy |
| 2 | Max\_depth | None,5,10,15,20,25,30,40,50 | 20 |
| 3 | Min\_sample\_split | 2,5,10,20 | 2 |
| 4 | Min\_sample\_leaf | 1,2,4,6,8 | 1 |
| 5 | Max\_features | None, sqrt, log2 | None |

Following model training with the optimal hyperparameter configuration, we assessed the performance of the Decision Tree using the test dataset. The evaluation was carried out based on the metrics outlined in Table 9. The detailed results, including the model's performance across various metrics, are presented in Table 4 and illustrated in Fig. 7. Overall, the Decision Tree classifier achieved an accuracy score of 88%, indicating a strong ability to correctly distinguish between stroke and non-stroke cases.

Table 2: Evaluation score for Decision Tree classifier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SN | Accuracy | Precision | Recall | F1-Score |
| 1 | 88% | 56% | 59% | 57% |

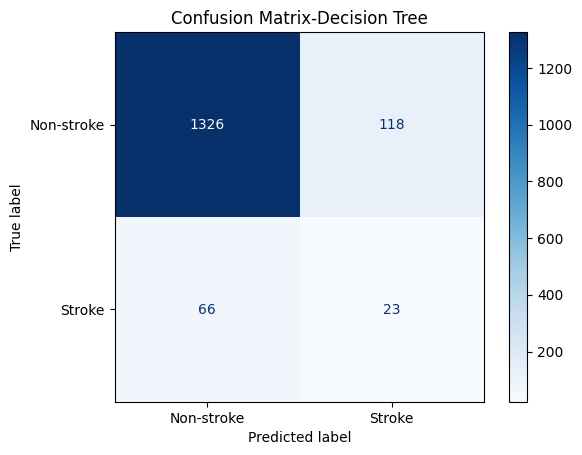


Fig. 7: Confusion Matrix for Decision Tree Classifier

### Logistic Regression Classifier

Logistic regression is commonly used to examine how predictor variables influence categorical outcomes. In most cases, the outcome is binary such as the presence or absence of a disease making it suitable for binary logistic regression models [8].

We trained our model using the Logistic Regression classifier and carried out hyperparameter tuning to optimize its performance. This was done using the grid search method, exploring a predefined set of values for key hyperparameters, as detailed in Table 3. The hyperparameters tuned included the regularization strength (C), the type of penalty applied, the solver used for optimization, and the class weight setting. After evaluating various combinations, the best configuration was found to be: C = 0.1, penalty = l2, solver = liblinear, and class\_weight = None.

Table 3: Hyperparameter search space for Logistic Regression Classifier

|  |  |  |  |
| --- | --- | --- | --- |
| SN | Hyperparameter | Search Space | Best-hyperparameter |
| 1 | C | 0.01, 0.1, 1, 10, 100 | 0.1 |
| 2 | Penalty | L1, L2 | L2 |
| 3 | Solver | Liblinear, saga | Liblinear |
| 4 | Class\_weight | None, balanced | None |

Using the optimal configuration obtained from the hyperparameter tuning process, we trained the Logistic Regression model and evaluated its performance on the test dataset. The evaluation was conducted based on the metrics outlined in Table 9. The model's results are presented in Table 4 and Figure 8. Overall, the Logistic Regression model achieved an accuracy of 80%, demonstrating a moderate ability to classify stroke outcomes in the dataset.

Table 4: Evaluation score for Decision Tree classifier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SN | Accuracy | Precision | Recall | F1-Score |
| 1 | 80% | 56% | 59% | 57% |

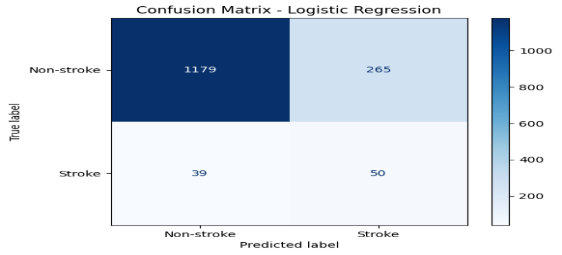


Fig. 8: Confusion Matrix for Logistic Regression

### Random Forest Classifier

Random Forest is an ensemble learning method that constructs multiple decision trees during training, where each tree is built using a random subset of features and data samples. Each tree in the forest is generated based on a random vector that is sampled independently but follows the same distribution across all trees. As the number of trees increases, the generalization error of the model converges almost surely to a stable limit [9].

The model was trained using the Random Forest algorithm, with hyperparameter tuning conducted to optimize performance. A grid search approach was applied over the search space defined in Table 5, focusing on key hyperparameters such as the number of trees (n\_estimators), maximum tree depth (max\_depth), minimum number of samples required to split an internal node (min\_samples\_split), minimum number of samples required at a leaf node (min\_samples\_leaf), and the number of features considered at each split (max\_features). After evaluating multiple combinations, the optimal configuration was identified as: n\_estimators = 200, max\_depth = None, min\_samples\_split = 2, min\_samples\_leaf = 1, and max\_features = sqrt.

Table 5: Hyperparameter search space for Random Forest Classifier

|  |  |  |  |
| --- | --- | --- | --- |
| SN | Hyperparameters | Search Space | Best-Parameter |
| 1 | N\_estimator | 50, 100, 200 | 200 |
| 2 | Max\_depth | None, 10, 20, 30 | None |
| 3 | Min\_sample\_split | 2, 5, 10 | 2 |
| 4 | Min\_sample\_leaf | 1, 2, 4 | 1 |
| 5 | Max\_features | Auto, sqrt, log2 | Sqrt |

The Random Forest model was trained using the optimal hyperparameter configuration obtained from the tuning process. Its performance was assessed using the evaluation metrics outlined in Table 9. The results of the evaluation are presented in Table 6 and Figure 9. The model achieved an accuracy of 91%, indicating strong predictive performance on the test dataset.

Table 6: Evaluation score for Random Forest Classifier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SN | Accuracy | Recall | Precision | F1-Score |
| 1 | 91% | 57% | 56% | 56% |

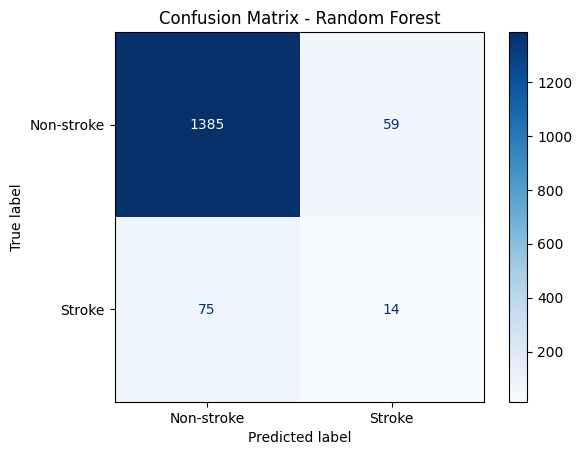


Fig. 9: Confusion Matrix for Random Forest Classifier

### XGBOOST Classifier

XGBoost is an ensemble learning algorithm designed for classification tasks. It is a powerful implementation of gradient tree boosting that incorporates advanced features such as efficient handling of sparse data and out-of-core computation, making it well-suited for large-scale and high-dimensional datasets [10].

The XGBOOST classifier was employed to train the model, with hyperparameter tuning performed using the grid search technique. The tuning process explored various values within the search space defined in Table 7, focusing on key hyperparameters including the number of estimators (n\_estimators), maximum tree depth (max\_depth), learning rate, subsample ratio, and column sampling ratio (colsample\_bytree). After evaluating multiple combinations, the optimal set of hyperparameters was identified as follows: n\_estimators = 200, max\_depth = 7, learning\_rate = 0.2, subsample = 1, and colsample\_bytree = 0.7.

Table 7: Hyperparameter search space for XGBOOST Classifier

|  |  |  |  |
| --- | --- | --- | --- |
| SN | Hyper-parameter | Search space | Best-hyper parameter |
| 1 | N\_estimator | 100,200 | 200 |
| 2 | Max\_depth | 3,5,7 | 7 |
| 3 | Learning\_rate | 0.01,0.1,0.2 | 0.2 |
| 4 | Subsample | 0.7,1.0 | 1 |
| 5 | Colsample\_bytree | 0.7,1.0 | 0.7 |

The model was trained using the optimal hyperparameter configuration obtained from the tuning process, utilizing the resampled training dataset. Its performance was then assessed based on the evaluation metrics outlined in Table 9. The results of this evaluation are presented in Table 8 and Figure 10. The XGBOOST model achieved an accuracy of 91%, demonstrating strong predictive performance on the test data.

Table 8: Evaluation score for XGBOOST Classifier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SN | Accuracy | Precision | Recall | F1-Score |
| 1 | 91% | 58% | 57% | 57% |

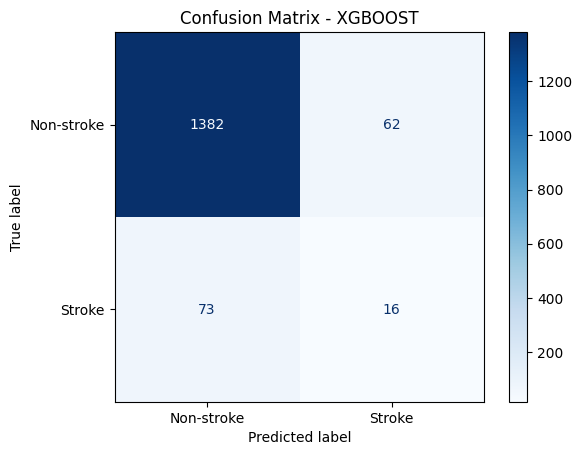


Fig. 10: Confusion Matrix for XGBOOST Classifier

### Evaluation Metrics

In this study, we employed five key evaluation metrics to assess model performance: Accuracy, Precision, Recall, F1 Score, and the Confusion Matrix. These metrics, along with their corresponding equations, are presented in Table 9. Each metric is derived from four fundamental classification outcomes: True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN). A True Positive represents a correctly predicted positive case, while a False Positive occurs when a negative case is incorrectly predicted as positive. Conversely, a True Negative refers to a correct prediction of a negative instance, and a False Negative arises when the model fails to identify a positive case, classifying it as negative instead [11].

**Table 9: Evaluation metrics and its equation**

|  |  |  |
| --- | --- | --- |
| SN | Metrics | Equation |
| 1 | Accuracy | (TP+TN)/(TP+TN+FP+FN) |
| 2 | Precision | TP/(TP+FP) |
| 3 | Recall (sensitivity) | TP/(TP+FN) |
| 4 | F1-Score | (2x(Precision x Recall)/(Precision + Recall) |
| 5 | Confusion Matrix |  |

**4.0 Discussion**

The performance of the four classification models, Decision Tree, Logistic Regression, Random Forest, and XGBOOST was evaluated based on accuracy, precision, recall, and F1-score, as presented in Table 10.

Table 10: Performance of the four classification models

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SN | Classifier | Accuracy | Precision | Recall | F1-Score |
| 1 | Decision Tree | 88% | 56% | 59% | 57% |
| 2 | Logistic Regression | 80% | 56% | 69% | 57% |
| 3 | Random Forest | 91% | 57% | 56% | 56% |
| 4 | XGBOOST | 91% | 58% | 57% | 57% |

Among all the classifiers, Random Forest and XGBOOST achieved the highest accuracy, both scoring 91%, indicating their strong capability in correctly classifying the instances in the dataset. This result suggested that ensemble-based methods, which combine multiple decision trees, can significantly improve model robustness and accuracy, especially in classification problems involving imbalanced datasets.

The Decision Tree model followed closely with an accuracy of 88%, while Logistic Regression lagged behind with 80% accuracy. Although Logistic Regression performed lower in terms of overall accuracy, it recorded the highest recall value at 69%, implying that it was more effective in identifying actual stroke cases within the dataset. This makes Logistic Regression a valuable option in scenarios where minimizing false negatives is critical such as medical diagnosis. In terms of precision, all models performed relatively close, ranging from 56% to 58%. XGBOOST slightly outperformed the others with 58%, indicating a better balance in predicting positive cases accurately without inflating false positives.

The F1-score, which balances both precision and recall, was fairly consistent across all models, with values between 56% and 57%. While Random Forest and XGBOOST had the highest accuracy, their F1-scores (56% and 57% respectively) suggest a trade-off between precision and recall, likely due to the imbalanced nature of the dataset. Overall, the ensemble models, Random Forest and XGBOOST not only led in accuracy but also demonstrated relatively balanced performance across other evaluation metrics.

**Conclusions**

This study has demonstrated the effectiveness of a multi-model machine learning approach in predicting stroke risk based on demographic and health-related data. By comparing four classification models, Decision Tree, Logistic Regression, Random Forest, and XGBoost we established that ensemble-based methods, particularly Random Forest and XGBoost, offer superior performance, achieving an accuracy of 91%. The integration of proper feature selection techniques (ANOVA F-test and Chi-squared test) and the use of SMOTE for addressing class imbalance proved critical in enhancing model robustness. While Logistic Regression lagged in accuracy, it showed the highest recall, highlighting its utility in identifying actual stroke cases, which is valuable in clinical scenarios where early intervention is key.

The findings support the potential of machine learning in augmenting stroke risk assessment. By aiding healthcare professionals in early diagnosis and targeted prevention, this approach can contribute meaningfully to public health outcomes. Future research should focus on real-time model deployment and integration with clinical decision support systems to further drive adoption in healthcare practice.

COMPETING INTERESTS DISCLAIMER:

Authors have declared that they have no known competing financial interests OR non-financial interests OR personal relationships that could have appeared to influence the work reported in this paper.

Disclaimer (Artificial intelligence)

Author(s) hereby declare that NO generative AI technologies such as Large Language Models (ChatGPT, COPILOT, etc.) and text-to-image generators have been used during the writing or editing of this manuscript.

**References**

[1] Azam MS, Habibullah M, Rana HK. Performance analysis of various machine learning approaches in stroke prediction. International Journal of Computer Applications. 2020 Sep;175(21):11-5.

[2] Emon MU, Keya MS, Meghla TI, Rahman MM, Al Mamun MS, Kaiser MS. Performance analysis of machine learning approaches in stroke prediction. In 2020 4th international conference on electronics, communication and aerospace technology (ICECA) 2020 Nov 5 (pp. 1464-1469). IEEE.

[3] Sailasya G, Kumari GL. Analyzing the performance of stroke prediction using ML classification algorithms. International Journal of Advanced Computer Science and Applications. 2021;12(6); 539-545.

[4] Mahesh KA, Shashank H, Srikanth S, Thejas A. Prediction of stroke using machine learning. InProceedings of the Conference Paper 2020 Jun.

[5] Chilamkurthy S, Ghosh R, Tanamala S, Biviji M, Campeau NG, Venugopal VK, Mahajan V, Rao P, Warier P. Deep learning algorithms for detection of critical findings in head CT scans: a retrospective study. The Lancet. 2018 Dec 1;392(10162):2388-96.

[6] Swain PH, Hauska H. The decision tree classifier: Design and potential. IEEE transactions on geoscience electronics. 1977 Jul;15(3):142-7.

[7] Mienye ID, Jere N. A survey of decision trees: Concepts, algorithms, and applications.IEEE access. 2024 Jun 19. https://ieeexplore.ieee.org/document/10562290

[8] Nick TG, Campbell KM. Logistic regression. Topics in biostatistics. 2007:273-301.

[9] Breiman L. Random forests. Machine learning. 2001 Oct;45:5-32.

[10] Fauzan MA, Murfi H. The accuracy of XGBoost for insurance claim prediction. Int. J. Adv. Soft Comput. Appl. 2018 Jul 1;10(2):159-71.

[11] Mbonu CE, Anigbogu K, Asogwa D, Belonwu T. An Explorative Analysis of SVM Classifier and ResNet50 Architecture on African Food Classification. arXiv preprint arXiv:2505.13923. 2025 May 20.

[12] Zhou, L., Ye, P., & Wang, Y. (2025). Exploring the potential biomarkers between stroke and obstructive sleep apnea by WGCNA and machine learning. Sleep and Breathing, 29(4), 1-12.

[13] Aggelousis, N. Identifying Risk Factors for Falls Prediction in Older Adults: An Explainable Machine Learning Approach. Networks. 2023; 87, 7-50.

[14] Amal, S., Kramer, R., Sawyer, D., Rabb, J. B., Maurais, A. S., Ross, C. S., ... & Winslow, R. L. (2022). Predicting Adverse Events in the Cardiothoracic Surgery Intensive Care Unit Using Machine Learning: Results and Challenges. medRxiv,2022; 2022-12.

[15] Cao, J., You, K., Xu, P., Sun, Y., Shao, J., Zhou, Y., ... & Ye, J. (2025). Advancing predictive, preventive, and personalized medicine in eyelid diseases: a concerns-based and expandable screening system through structural dissection. EPMA Journal 2025, Volume 16; 1-14.