Predicting Stroke Risk Based on an Optimized Machine Learning Model

Felix Liu[1]

Sendelta International Academy, 518103, Shenzhen, China

Abstract. Stroke risk prediction is critical for identifying at-risk populations in healthcare as well as for early diagnosis and resource optimization in stroke management. This study's main goal was to create a reliable stroke prediction model by analyzing a publically accessible dataset containing 4981 records and 11 variables using machine learning techniques. Initial data preprocessing consisted of converting categorical variables to numerical values and addressing class imbalance through methods such as class weighting and the Synthetic Minority Oversampling Technique (SMOTE). Three basic models - logistic regression, decision trees, and random forests - were implemented to establish base performance. To improve prediction accuracy and address imbalances, optimized ensemble models were built using stacking and hyperparameter tuning through grid search and cross-validation. This paper evaluates the performance of the model based on composite indicators. While the base model demonstrated high accuracy and recall, the optimized model had a superior performance with an AUC score of 0.94, showing that the capacity to distinguish between stroke and non-stroke cases has significantly improved. By offering a useful predictive tool that can precisely estimate the risk of stroke and direct creative ways to early intervention, this study advances the field of healthcare analytics.

1. Introduction

Stroke risk assessment is an important issue in the global health field, and precise predictive models are required to identify potential high-risk populations and enhance clinical decision-making. Mastering various factors that may lead to stroke is crucial for taking preventive measures and improving patients' rehabilitation outcomes. With the rise of the incidence rate of stroke, especially among the elderly, the demand for efficient prediction tools has become an unprecedented urgency. This study's objective is to use health data analysis to forecast the likelihood of stroke, and to assist healthcare providers in making more informed decisions.

Recent literature highlights various methodologies employed in stroke prediction, underscoring advancements in machine learning applications within healthcare. In the first article, data estimation, feature selection, and machine learning prediction models were

Corresponding author: Felixliu0813@outlook.com

combined to propose a new stroke prediction method that outperformed the traditional Cox proportional hazards model on the CHS dataset, revealing new potential risk factors for stroke [1]. In the second article, a classification model was built using artificial intelligence techniques, utilizing a backpropagation neural network classification algorithm, PCA for dimensionality reduction, and a decision tree algorithm for feature selection [2]. In the third article, the application of Support Vector Machines (SVMs) in stroke prediction was explored and various SVM kernel functions were trained and tested on data collected from the International Stroke Trial Database [3]. The fourth article proposes employing a Bayesian Rule List (BRL) approach to formulate a collection of "if-then" rules that simplify a complex, high-dimensional feature space into clear and comprehensible decision-making statements. This approach preserves the interpretability of medical scoring systems while achieving predictive performance comparable to leading machine learning algorithms [4]. Together, these studies illustrate the evolving landscape of stroke risk prediction and highlight how novel strategies can improve both accuracy and reliability.

The paper aims to create a robust and accurate stroke prediction model that tackles the challenges posed by data imbalances and the need to optimize machine learning techniques and provides clear, actionable insights for clinical practice. The dataset and approach used will be described in depth in the subsequent sections, followed by the results and analysis of both base and optimized models, ultimately demonstrating the effectiveness of the proposed approach in improving stroke prediction accuracy.

2. Dataset and Methodology

2.1 Dataset Description

The purpose of this dataset is to predict the risk of stroke, aiming to study and predict various factors affecting strokes to support clinical decision-making. By analyzing patients' health data, helps to assess the likelihood of a stroke and assists doctors in identifying high-risk patients. The dataset comes from Kaggle (https://www.kaggle.com/datasets/jillanisofttech/brain-stroke-dataset/data).

It contains 4,981 records and includes 11 feature variables as shown in Table 1:

Table 1. Variables and Attributes

Variable Name	Attributes
gender	Contains two categories representing males and females.
age	Age ranges from 0.08 to 82 years, with an average age of 43.42 years and a standard deviation of 22.66.
hypertension	Indicates whether an individual has high blood pressure, with 0 representing no and 1 representing yes.
heart_disease	Indicates whether an individual has heart disease, with 0 representing no and 1 representing yes.
ever_married	Indicates whether the individual has ever been married, with values of "Yes" and "No."
work_type	Contains four different types of work categories.
Residence_type	Indicates the type of residential environment, with two categories.
avg_glucose_level	The average glucose level is 105.94, with a range between 55.12 and 271.74.
bmi	The average BMI is 28.49, ranging from 14.0 to 48.9.
smoking_status	Indicates the individual's smoking status, with four categories.
stroke	The target variable indicates whether the individual has had a

	stroke, with 0 representing no and 1 representing yes.

The dataset has no missing values, making it highly complete, which reduces the workload for data cleaning and preprocessing. Despite the completeness of the dataset, the target variable suffers from a severe category imbalance problem, with only about 5% of the sample representing stroke cases. Therefore, this problem is optimized in later sections.

2.2 Data Preprocessing

Before building the prediction model, necessary preprocessing steps were applied to the dataset [5,6]. First, categorical variables were converted to numerical values. For example, "Yes" was changed to 1 and "No" to 0 in the "ever_married" column. "Male" was changed to 1 and "Female" to 0 in the "gender" column. One-hot encoding was used for multi-class variables, such as work_type, residence_type, and smoking_status, to transform each category into binary columns. Finally, no missing value treatment was required since the dataset is complete, reducing the effort in data cleaning. After preprocessing, all categorical variables were successfully transformed into numerical form for the model to understand. Multi-class variables were encoded to ensure the model did not misunderstand the relationships between categories. The complete dataset ensures effective training without missing data affecting predictions. This preprocessing step improves model performance and accuracy, enabling the model to more successfully identify patterns in the data, reducing errors, and increasing prediction reliability. This sets the foundation for subsequent analysis and ensures model robustness when handling complex data.

3. Model Building

3.1 Splitting the Training and Testing Set

To ensure that the procedure is dependable and repeatable, it is essential to separate the dataset into training and test sets for later model training and assessment. First, the dataset was divided into target variables and feature matrices. The target variable is the stroke column that we want to predict, whereas the feature matrix includes all of the input variables. The dataset was randomly divided into a training set (75%) and a testing set (25%), using the train_test_split function. This approach efficiently distributes data to guarantee that the model can pick up significant characteristics during training and verify its capacity for generalization on the test set. To ensure repeatability in data splitting, a random seed of random_state=3 was set so that the results of each run remain consistent, aiding debugging and result verification [7].

3.2 Base Models

In the pursuit of effective stroke prediction, 3 foundational machine learning models can be employed, each with its unique strengths and weaknesses. The logistic regression model is a widely used method for applications involving binary classification, estimating the probability of an event occurring [8]. It applies a sigmoid function to convert linear regression outputs into probabilities and predicts classes based on a defined threshold. Known for its simplicity and efficiency, logistic regression may struggle with imbalanced data. A decision tree classifier, on the other hand, chooses the feature that best divides the classes at each node to create a tree-like model for classification or regression tasks, recursively partitioning the data. Despite being simple to understand, decision trees have a tendency to overfit, particularly in small or imbalanced datasets. The random forest model addresses some of

these limitations by being an ensemble of multiple decision trees, classifying by majority voting. It builds several trees from random subsets of the training data, enhancing stability and accuracy through this ensemble approach. Random forests can handle high-dimensional data and resist overfitting, but they may still have difficulty recognizing minority classes in imbalanced datasets. Together, these base models provide a robust foundation for addressing the stroke prediction task [9,10].

3.3 Model Optimization

3.3.1 Addressing Data Imbalance

Since in this dataset, the positive category sample (stroke cases) represents only 5% of the sample, the model is susceptible to the category imbalance problem, causing it to be more inclined to predict the negative category sample, thus ignoring the small number of stroke cases. To address this issue, class weighting adjustment was applied during model training by increasing the weight of the positive class so that it contributes more to the loss calculation. This was achieved by setting class_weight='balanced' to automatically balance class weights. Additionally, resampling techniques were used, such as undersampling to decrease negative class samples and oversampling (e.g., Synthetic Minority Oversampling Technique (SMOTE)) to increase the number of positive class samples. By creating synthetic samples for the minority class, SMOTE improves the model's capacity to learn from these instances. However, it can potentially lead to overfitting by creating very similar synthetic examples. On the other hand, undersampling may risk losing valuable information by discarding negative class samples, which could also affect the model's performance. Thus, while these techniques can mitigate imbalance issues, careful consideration is necessary to avoid introducing new biases or losing critical data.

3.3.2 Ensemble Optimization

To improve model performance, stacking was applied by combining predictions from multiple base models (logistic regression, decision tree classifier, and random forest) with a final meta-model to make the final prediction. Stacking utilizes each model's capabilities, enhancing prediction robustness when individual models perform poorly. Additionally, grid search and cross-validation were used. GridSearchCV selects the best-performing hyperparameter combinations by evaluating various combinations on the training data using cross-validation techniques (e.g., 5-fold cross-validation). Hyperparameters for each base model were tuned, including the number of trees, maximum features, and maximum depth for random forests; the regularization strength (C) and penalty type (L1 or L2) for logistic regression; and the maximum depth and minimum samples split for decision trees. In cross-validation, the training data is divided into subsets, the model is trained on certain subsets and validated on others, and the performance of each combination of hyperparameters is evaluated on both the training and validation sets. However, the computational cost of grid search can be quite high, especially with large datasets, as it may require extensive computational resources and time to evaluate all possible combinations of hyperparameters. This trade-off between thoroughness and efficiency must be carefully considered in the modeling process.

3.4 Evaluation Metrics

Once the optimal model was selected, predictions were made on the test set using the following five metrics: accuracy, precision, recall, F1-score, and AUC score, ensuring the model's improved recognition of the positive class while maintaining good generalization ability.

Accuracy is the percentage of accurately predicted samples among all samples, and it indicates how accurate the model's predictions are overall. However, in imbalanced datasets, accuracy might not fully reflect the model's capacity to identify minority classes (e.g., stroke cases). The formula is:

$$\frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

Where FP stands for false positive, FN for false negative, TP for true positive, and TN for true negative. TP is a true positive, TN is a true negative, FP is a false positive, and FN is a false negative.

Precision is the proportion of predicted positive samples that are actually positive. A high precision means that the model has a low false positive rate, useful when false positives have high costs. The formula is:

$$\frac{TP}{TP+FP} \tag{2}$$

Recall is the proportion of actual positive samples that are correctly predicted. A high recall indicates that the model effectively captures most positive cases (e.g., stroke patients). The formula is:

$$\frac{TP}{TP+FN} \tag{3}$$

The F1-score is the harmonic mean of precision and recall, making it an ideal metric for handling class imbalance and reflecting the overall performance in predicting the positive class. The formula is:

$$2 \times \frac{Precision \times Recall}{Precision+Recall} \tag{4}$$

The AUC score measures the effectiveness of a binary classification model across various classification thresholds. It represents the area under the Receiver Operating Characteristic (ROC) curve. A higher AUC score, ranging from 0.5 to 1, indicates better discrimination ability. An AUC of 1.0 signifies flawless classification, while a score of 0.5 suggests that the model performs no better than random chance. In particular, higher AUC scores reflect improved model performance in distinguishing between positive and negative classes, especially in unbalanced datasets.

4. Results Analysis

4.1 Base Model Results

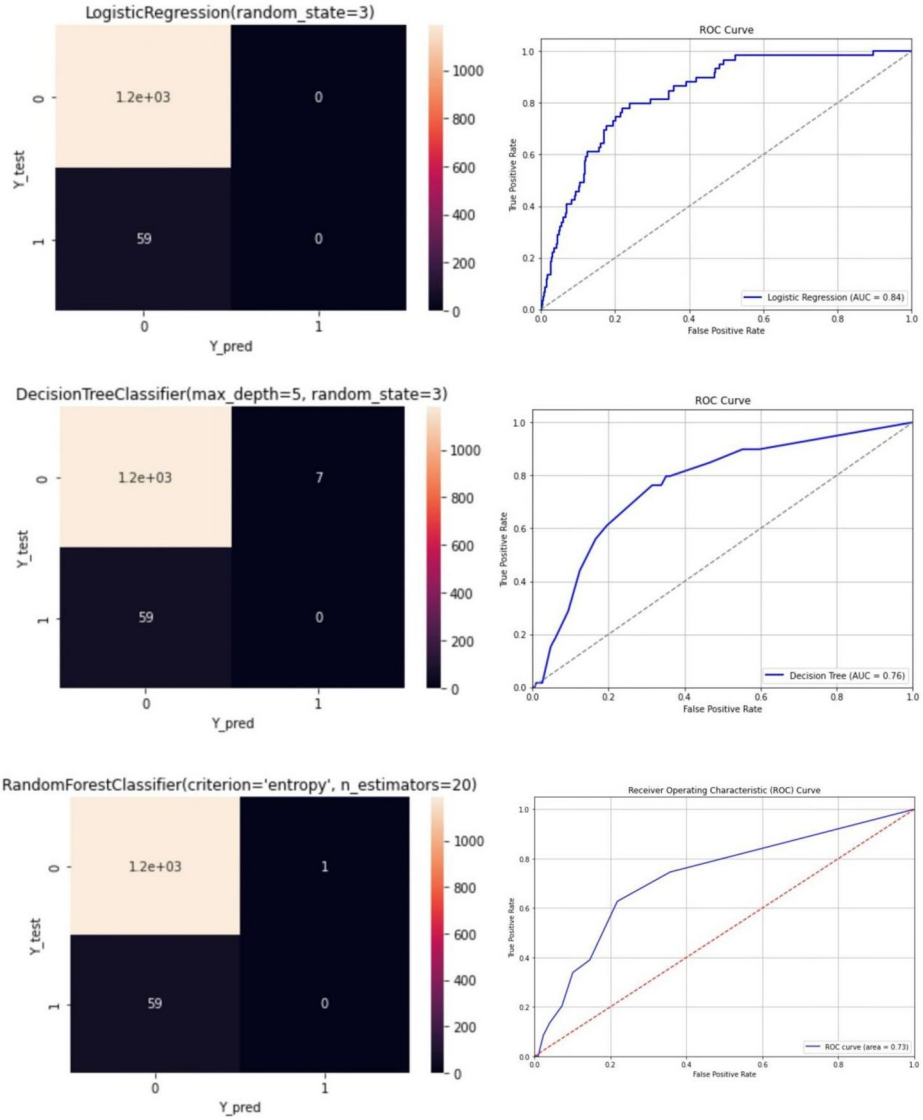

Fig. 1. Confusion matrix and ROC curve of the base models (Photo/Picture credit: Original).

As shown in Fig. 1, the performance metrics of various classification models show significant differences. Specifically, the logistic regression model demonstrated a series of excellent performance indicators: its accuracy rate was 0.95, its precision rate was also 0.95, and its recall rate reached 1.00, which shows that the model can perfectly identify all positive samples. Furthermore, its F1 score of 0.98 reflects the model's good balance between precision and recall, while its AUC score of 0.84 indicates its strong ability to distinguish between positive and negative class samples.

In comparison, the performance of the decision tree model is slightly inferior, with an accuracy rate of 0.94, a precision rate of 0.95, and a recall rate of 0.99, showing that the model's ability to identify positive classes is still very high. The F1 score is 0.97, which shows that the decision tree can still effectively capture most positive samples while maintaining

accuracy. However, the AUC score is 0.76, indicating its relatively weak classification ability under different thresholds.

In addition, the various indicators of the random forest model are: accuracy rate is 0.95, precision rate is 0.95, recall rate is also 1.00, and F1 score is 0.97, which shows that it performs well when processing positive samples. However, the AUC score of 0.74 shows that it is slightly deficient in overall classification performance. In summary, these performance indicators not only reflect the effectiveness of each model in classification tasks but also suggest that different indicators should be considered comprehensively when selecting a model to achieve the best classification effect. All three base models performed well on accuracy, precision, recall, and F1-score, indicating strong overall prediction capability. The models were highly accurate in predicting all samples, had low false positive rates, and captured the most positive samples with low false negative rates. However, the base models performed poorly on the AUC score, suggesting a weak distinction between positive and negative samples, with predictions being somewhat random.

4.2 Optimized Model Results

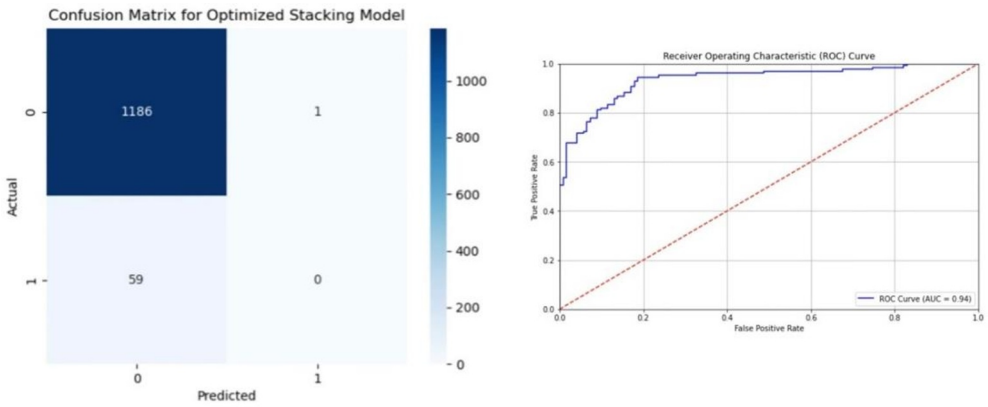

Fig. 2. Confusion matrix and ROC curve of the optimized model (Photo/Picture credit: Original).

As shown in Fig. 2, the optimized model achieved an accuracy of 0.95, a precision of 0.95, a recall of 1.00, an F1-score of 0.98, and an AUC score of 0.94. The optimized model maintained the same high levels of accuracy, precision, recall, and F1-score as the three base models while significantly improving the AUC score to 0.94. This indicates that the model's capacity to discriminate between positive and negative classes has significantly improved, reducing misclassification and greatly enhancing overall performance on the imbalanced dataset. Through parameter optimization and data rebalancing techniques, the optimized model exhibited a markedly improved AUC score, showcasing stronger robustness in accurately distinguishing between positive and negative samples and reducing the risk of misclassification. This enhancement ensures that the model performs more consistently and is applicable in practical scenarios, particularly in fields like finance and healthcare where imbalanced datasets are common.

5. Conclusion

This paper used a systematic approach to predict stroke risk by integrating basic and advanced machine learning techniques. This paper adopts an integrated approach, covering data preprocessing and modeling of multiple basic models (logistic regression, decision tree,

random forest). To solve the problem of data imbalance, this paper uses class weight adjustment and oversampling strategies. Through superimposed ensemble optimization, this research combines the meta-model's predictions with those of the basic models to increase overall accuracy and robustness. The best hyperparameters for each model are simultaneously found using grid search and cross-validation in order to thoroughly assess its performance. According to the results, the optimized model performs noticeably better than the original model, maintaining the base model's strong performance in terms of accuracy, precision, recall, and F1 score while also greatly raising the AUC score to 0.94. This shows a significant improvement in the model's capacity to distinguish between stroke and non-stroke events, particularly when dealing with unbalanced datasets, a prevalent problem in medical data analysis.

Looking ahead, future research could focus on incorporating additional features or external datasets to refine predictive accuracy further. Investigating cutting-edge approaches like ensemble methods and deep learning may provide a more profound understanding of stroke risk factors. This study emphasizes how crucial it is to predict strokes accurately in clinical settings so that medical professionals can identify high-risk patients and take prompt action. Enhancing predictive models contributes to the evolving landscape of healthcare analytics, ultimately aiming for better patient outcomes and more effective resource allocation in stroke management.

Reference

1. A. Khosla, Y. Cao, C.C.Y. Lin, H.K. Chiu, J. Hu, H. Lee, An integrated machine learning approach to stroke prediction, in Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 183-192 (2010)
2. M.S. Singh, P. Choudhary, Stroke prediction using artificial intelligence, in 2017 8th Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON), 158-161 (2017)
3. R.S. Jeena, S. Kumar, Stroke prediction using SVM, in 2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 600-602 (2016)
4. B. Letham, C. Rudin, T.H. McCormick, D. Madigan, Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model (2024)
5. 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**, 100032 (2022)
6. M.U. Emon, M.S. Keya, T.I. Meghla, M.M. Rahman, M.S. Al Mamun, M.S. Kaiser, Performance analysis of machine learning approaches in stroke prediction, in 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), pp. 1464-1469 (2020)
7. M.S. Islam, I. Hussain, M.M. Rahman, S.J. Park, M.A. Hossain, Explainable artificial intelligence model for stroke prediction using EEG signal. Sensors, **22**(24), 9859 (2022)
8. G. Sailasya, G.L.A. Kumari, Analyzing the performance of stroke prediction using ML classification algorithms. Int. J. Adv. Comput. Sci. Appl. **12**(6) (2021)
9. M. Rajora, M. Rathod, N.S. Naik, Stroke prediction using machine learning in a distributed environment, in Distributed Computing and Internet Technology: 17th

International Conference, ICDCIT 2021, Bhubaneswar, India, January 7–10, 2021, Proceedings **17**, pp. 238-252 (2021)

10. S. Gangavarapu, L.A.K. Gorli, Analyzing the performance of stroke prediction using ML classification algorithms. Int. J. Adv. Comput. Sci. Appl. 12, (2021)