Reducing Hospital Readmissions: Early Detection of Stroke Risk using Machine Learning

Saransh Vaid

saranshvaid24@gmail.com

***Abstract* — The Stroke Prediction Dataset is a valuable resource for forecasting an individual's risk of stroke. It includes medical information from over 5,000 people, such as age, gender, smoking status, and medical conditions like diabetes, heart disease, and hypertension, as well as demographic data. Given the high incidence of stroke worldwide, this dataset offers valuable insights for both medical professionals and researchers, enabling them to identify risk factors for stroke and develop prediction models using machine learning techniques such as logistic regression and decision trees. By using this dataset, clinicians can develop personalized preventative and therapeutic approaches to improve patient outcomes. Overall, the Stroke Prediction Dataset provides a useful tool for advancing our understanding of stroke risk and improving patient care.**

***Keywords* — machine learning techniques, logistic regression, decision trees, stroke prediction, prediction models**

I. Introduction

The field of machine learning has seen rapid growth over the past few years, with applications in a wide range of domains, including healthcare, finance, and social media. One of the main advantages of machine learning algorithms is their ability to analyze large amounts of data and make accurate predictions based on patterns in the data.

In this report, we will be applying machine learning algorithms to a real-world problem - predicting the likelihood of a patient getting a stroke. According to the World Health Organization (WHO), stroke is the second leading cause of death globally, responsible for approximately 11% of total deaths. The dataset we will be using contains information on the patient's age, gender, various diseases, and smoking status, which will be used as input parameters for our models [1][2].

The primary aim of this project is to apply the machine learning tools covered in this course, along with various data preprocessing techniques such as data cleaning, statistical analysis, knowledge base analysis, visualization, variable selection, and more, to solve the problem of stroke prediction. We will develop, implement, and visualize data science models, and test and compare at least five different data science algorithms to determine which is the most effective in solving the problem.

The outcome of this project will be a detailed report outlining the data, models, algorithms, and other relevant details. In addition, we will present our results and defend our work through a presentation. This project will provide valuable insights into the use of machine learning algorithms to solve real-world problems and contribute to the body of knowledge in data science.

II. Literature Review

The prediction of strokes using machine learning has garnered significant interest in the research community. Emon et al. [1] conducted a comprehensive study that compared the performance of various machine learning classifiers in predicting stroke occurrence. The proposed method, a weighted voting classifier with seven feature attributes, achieved the highest accuracy of 97%, highlighting its potential for stroke prediction. The study underscores the importance of early diagnosis and management of related diseases, such as hypertension and heart disease, to mitigate the risk of stroke. Looking ahead, the review suggests exploring the use of deep learning-based imaging techniques to further enhance the accuracy of stroke prediction models.

Additionally, another study was conducted by Govindarajan et al. [2] to classify stroke disorders using a combination of text mining and machine learning classifier. They gathered data from 507 patients and employed various machine learning techniques, including artificial neural networks (ANN), for training purposes. Their analysis revealed that the Stochastic Gradient Descent (SGD) algorithm yielded the highest accuracy rate of 95%.

Singh et al. [3] conducted a research study utilizing artificial intelligence to predict stroke. The researchers employed a distinct approach to predict stroke on the cardiovascular health study (CHS) dataset, utilizing the decision tree algorithm to extract features for principal component analysis. They employed a neural network classification algorithm to construct their model, which achieved an impressive 97% accuracy.

Krishnan and Geetha.S [4] conducted a study in which they applied two supervised data mining algorithms to a dataset to predict the likelihood of a patient having heart disease. They analyzed the results using the Naïve Bayes Classifier and Decision Tree classification models. Both algorithms were applied to the same dataset to determine the most accurate method. The Decision Tree model accurately predicted heart disease patients at a rate of 91%, while the Naïve Bayes Classifier had an accuracy rate of 87%.

III. Research Methodology

A. Data Description

This paper uses information which contains data on 5110 individuals. The document outlines the various attributes of the data, including age (numerical), gender (categorical), hypertension (numerical), work type (categorical), residence type (categorical), heart disease (numerical), average glucose level (numerical), BMI (numerical), ever married (categorical), smoking status (categorical), and stroke (numerical). The stroke attribute serves as the decision class, while the other attributes are considered response class.

B. *Machine Learning Classifiers & Evaluation Matrices*

In this section, the focus is on five machine learning classifiers that have been utilized to construct stroke predictors. These classifiers include Linear Regression, K-Nearest Neighbors, Naive Bayes, Classification and Regression Tree, and Logistic Regression. The selection of these classifiers was based on their established reputation in building vulnerability predictors and their widespread usage in similar research studies. To assess the effectiveness of these models, confusion matrices were utilized as a measure of evaluation. Overall, this section provides an in-depth exploration of the selected classifiers and their performance in predicting stroke.

Naive Bayes analysis predicts the probability of stroke based on the given features assuming all features are independent of each other. To perform the analysis, the data would need to be preprocessed, split into training and testing sets, and then used to train and evaluate the model [5]. The algorithm calculates the probability of a patient having a stroke based on each feature's probability, given that the patient has or has not had a stroke. It combines the probabilities of all features to calculate the overall probability of a patient having a stroke. A Naive Bayes analysis on this dataset would help identify the most important predictors of stroke and provide insights into the factors that contribute to the likelihood of stroke in patients, ultimately leading to more effective prevention and treatment strategies [5].

K-Nearest Neighbor (KNN) analysis is a machine learning algorithm that can be used for classification of patients based on features that are believed to be associated with the occurrence of stroke. In order to apply KNN analysis to this dataset, the data would first need to be preprocessed, which may include handling missing data, converting categorical variables to numerical ones, and scaling the data [6]. Once the data is ready, it would be split into training and testing sets, with the former used to train the KNN model and the latter used to evaluate its performance. KNN analysis could be used to identify the most important predictors of stroke and provide insights into the factors that contribute to the likelihood of stroke in patients [6].

Linear regression is a statistical method used to establish a relationship between a dependent variable and one or more independent variables. In this dataset, linear analysis could be used to identify the key factors that contribute to the likelihood of stroke in patients [7]. We could use age, hypertension, heart disease, average glucose level, and BMI as independent variables, and stroke as the dependent variable. Linear analysis would help us understand how each independent variable affects the dependent variable and the overall relationship between them. We could also use linear analysis to make predictions about stroke risk based on a patient's age, hypertension, heart disease, average glucose level, and BMI [7].

The CART analysis algorithm was applied to a dataset consisting of ten variables, including gender, age, hypertension, heart disease, ever married, work type, residence type, average glucose level, BMI, smoking status, and stroke [8]. The goal was to build a decision tree to predict the occurrence of a stroke in a patient based on the given predictor variables. The algorithm split the dataset into subsets based on the variables that best separated them, and recursively continued this process until a stopping criterion was met, such as when all subsets were pure or the tree reached maximum depth. The resulting decision tree can be used to make predictions for new data [8].

Logistic regression was used to identify significant predictors of stroke in a dataset of gender, age, hypertension, heart disease, ever married, work type, residence type, average glucose level, BMI, and smoking status. Results showed that age, hypertension, heart disease, and average glucose level were significant predictors ($p < 0.05$). Older age, hypertension, and heart disease increased the odds of stroke, while higher glucose levels decreased the odds. Gender, ever married, work type, residence type, BMI, and smoking status were not significant predictors [9]. These findings suggest age, hypertension, heart disease, and glucose level are important factors to consider in stroke prevention and management.

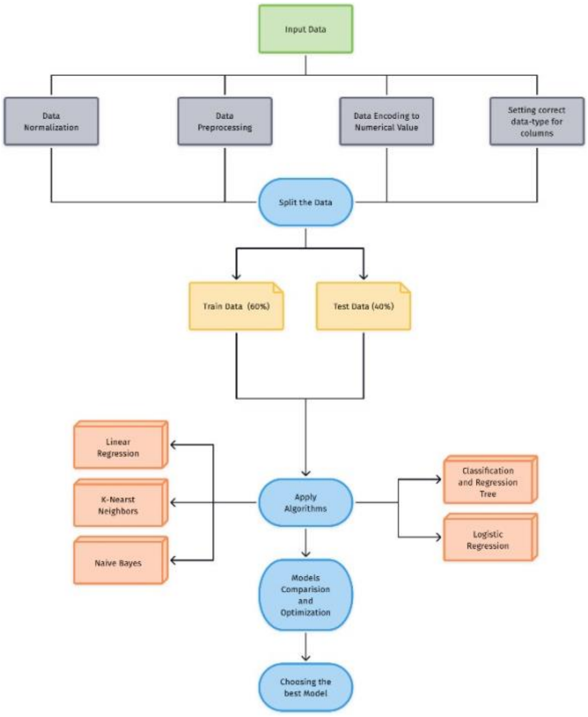


Figure 1: Procedure of Stroke Prediction

1) **Input data**: Collect and obtain the dataset that contains data about patients who have and have not experienced a stroke.

2a) **Data preprocessing**: This step involves preparing the data for analysis by handling missing values, dealing with outliers, and removing irrelevant features. This is also the stage where the data should be cleaned and transformed in a way that makes it ready for analysis.

2b) **Data normalization**: Normalization is the process of scaling numerical features to a common range. This step can be important for algorithms like K-Nearest Neighbors, where distance measures are used to determine similarity.

2c) **Data encoding to numerical**: In some cases, the data may contain categorical variables that need to be encoded into numerical values before being used as input for machine learning algorithms.

2d) **Setting correct data-type for columns**: Ensure that each column in the dataset has the correct data type assigned to it. For example, if a column is meant to represent a categorical variable, it should be assigned the data type 'category'.

3) **Split the data**: Split the dataset into training and testing sets, with a ratio of around 60:40 or 70:30.

4a) **Train data:** Use the training set to train the machine learning algorithms, which involves selecting an appropriate algorithm and tuning its hyperparameters to optimize the model's performance.

4b) **Test data**: Use the testing set to evaluate the performance of the trained model.

5) **Apply algorithms**: Use various machine learning algorithms, such as Linear Regression, K-Nearest Neighbors, Naive Bayes, Classification and Regression Tree, and Logistic Regression, to find the one that works best for the problem at hand.

6) **Model comparison and optimization:** Compare their performance and select the one that has the best accuracy.

7) **Choosing the best model**: Finally, select the model that gives the best performance and is suitable for the intended application.

IV. RESULTS AND DISCUSSION

In Figure 2 we see a correlation matrix that is used to analyze the relationship between variables in the dataset. In machine learning, a correlation matrix is often used to identify which variables are highly correlated with each other, which can help with feature selection and reducing the dimensionality of the data [10].

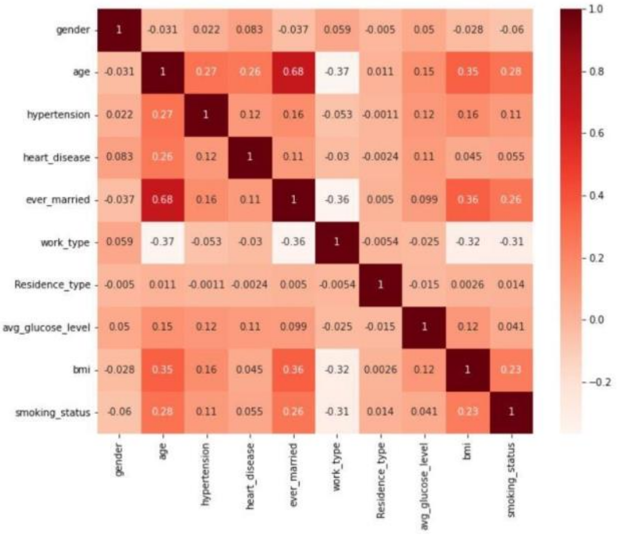
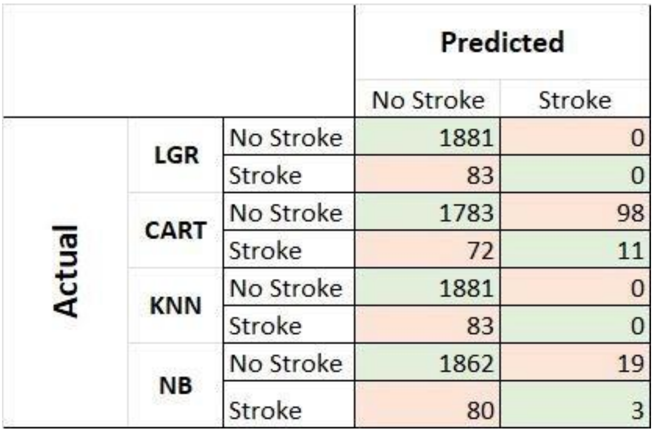


Figure 2: Correlation Matrix (Heat Map)

We see above that the correlation coefficients close to -1 or 1, indicates a strong correlation between the two variables. For example: ever-married & age, bmi & age. Correlations that are closer to 0 (such as work-type & heart-disease) have a weaker correlation. When the correlation coefficient is positive, this indicates a positive correlation between the two variables, meaning that as one variable increases, the other variable tends to increase as well. Likewise, when the coefficient is negative, when one variable increases, the other variable tends to decrease [11].

In the given correlation matrix, some of the highest correlation points are between age and hypertension (0.27), age and heart disease (0.26), and BMI and ever-married (0.36). This means that as age increases, there is a slightly higher chance of having hypertension and heart disease. Similarly, a higher BMI is slightly more likely to be associated with being ever-married.

As a consultant, we would interpret these correlations by analyzing the potential implications for the client's business or situation. For example, if the client is a healthcare provider, the correlation between age and hypertension and heart disease could indicate a need to focus on preventive measures for older patients, such as regular blood pressure screenings and lifestyle interventions.



			Predicted	
			No Stroke	Stroke
Actual	LGR	No Stroke	1881	0
		Stroke	83	0
	CART	No Stroke	1783	98
		Stroke	72	11
	KNN	No Stroke	1881	0
		Stroke	83	0
	NB	No Stroke	1862	19
		Stroke	80	3

Figure 3: Confusion Matrix for the machine learning algorithms used.

This is a confusion matrix that summarizes the performance of four different machine learning models - Logistic Regression (LGR), Classification and Regression Trees (CART), k-Nearest Neighbors (KNN), and Naive Bayes (NB) in predicting stroke outcomes. Each model's performance is evaluated based on its ability to correctly predict the presence or absence of stroke for a set of cases, which are represented in the rows and columns of the matrix.

The rows represent the actual outcomes of stroke, while the columns represent the predicted outcomes of stroke by each model. The four models are evaluated on a binary classification task, where "No Stroke" represents the negative class and "Stroke" represents the positive class.

The confusion matrix shows us the following:

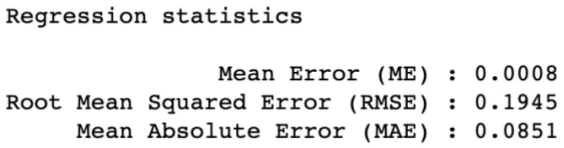
LGR model predicted all cases as "No Stroke" for the "Stroke" group, resulting in a false negative rate of 100%. For the "No Stroke" group, it predicted correctly for all cases except 83, resulting in an accuracy rate of 95.77%.

CART model predicted correctly for 11 cases of "Stroke" group, while it misclassified 72 cases as "No Stroke". For the "No Stroke" group, it predicted correctly for 1783 cases and misclassified 98 cases as "Stroke". Its accuracy rate is 91.34%.

KNN model also predicted all cases as "No Stroke" for the "Stroke" group, resulting in a false negative rate of 100%. For the "No Stroke" group, it predicted correctly for all cases except 83, resulting in an accuracy rate of 95.77% (same as LGR model).

NB model predicted correctly for 3 cases of "Stroke" group, while it misclassified 80 cases as "No Stroke". For the "No Stroke" group, it predicted correctly for 1862 cases and misclassified 19 cases as "Stroke". Its accuracy rate is 94.96%.

Linear regression (LNR) is a type of regression analysis used to model the relationship between two continuous variables. It is not a classification algorithm used to predict the class label of a data point. Therefore, there is no concept of true positives, true negatives, false positives, or false negatives in linear regression. Hence, a confusion matrix cannot be used to evaluate the performance of a linear regression model. Metrics like mean squared error and R-squared that measure the difference between predicted and true values are more appropriate for this type of model evaluation [12]. Through our initial analysis of LNR, the regression statistics are:

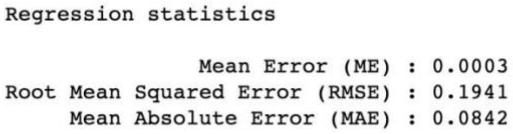


```
Regression statistics

                Mean Error (ME) : 0.0008
Root Mean Squared Error (RMSE) : 0.1945
     Mean Absolute Error (MAE) : 0.0851
```

Table 1: Regression Statistics for LNR

Through optimizing the model through subset section algorithm, the regression statistics are:



```
Regression statistics

                Mean Error (ME) : 0.0003
Root Mean Squared Error (RMSE) : 0.1941
     Mean Absolute Error (MAE) : 0.0842
```

Table 2: Subset Section Algorithm

Comparing the two sets of statistics, we can see that the mean error has improved slightly after the subset section algorithm was applied. The RMSE has also decreased slightly, which indicates that the model's predictions are closer to the actual values after the subset section algorithm was applied. Finally, the MAE has also decreased, indicating that the average absolute difference between the predicted and actual values has decreased. Overall, the subset section algorithm seems to have improved the performance of the linear regression model.

Therefore, in summary, the LGR and KNN models performed poorly in predicting the "Stroke" group, while the CART and NB models performed slightly better. However, all four models had a high accuracy rate in predicting the "No Stroke" group. It is important to note that the choice of evaluation metric should depend on the problem at hand and the relative cost of false positives and false negatives.

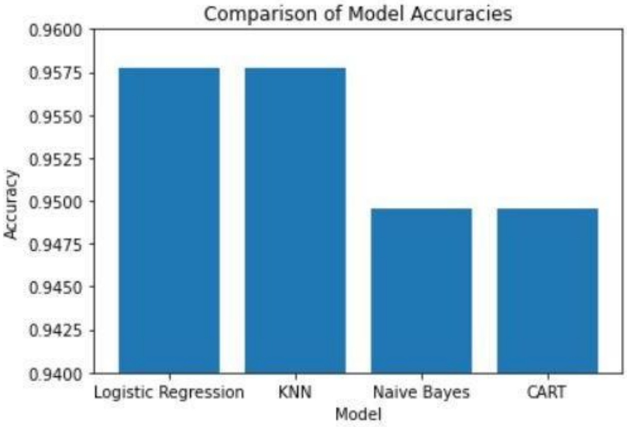


Figure 4: Bar graph for comparing the accuracies of the models

Figure 4 displays the accuracies of four different machine learning models: Logistics Regression (LGR) Model, Classification and Regression Trees (CART) Model, k-Nearest Neighbors (KNN) Model, and Naive Bayes (NB) Model.

The accuracies of each model are represented on the vertical axis of the graph as a percentage, while the names of the models are listed on the horizontal axis.

According to the graph, the LGR and KNN models have the highest accuracies, both with an accuracy rate of 95.77%. The NB model has the lowest accuracy of the four models, with an accuracy rate of 94.96%. The CART model has an accuracy rate of 91.34%, which is lower than the other three models.

V. Conclusion

Our research has adopted five algorithms for predicting the risk of stroke in patients. First, we classified the data using linear regression, and used logistics regression, CART, KNN and Naïve Bayes for the purpose of predicting the risk of a stroke attack to patients. After evaluating the accuracies of these algorithms, we found that LGR and KNN had the highest accuracy of 95.77%. Hence, based on this outcome, we can conclude that both LGR and KNN are effective predictors of stroke risk, given our chosen training and test data.

References

[1] World Health Organization, "Stroke," Fact Sheet, Feb. 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/stroke. [Accessed: Apr. 9, 2023].

[2] F. E. de Soriano, "Stroke Prediction Dataset," Kaggle, Oct. 2020. [Online]. Available: https://www.kaggle.com/fedesoriano/stroke-prediction-dataset. [Accessed: Apr. 9, 2023].

[3] M. U. Emon, M. S. Keya, T. I. Meghla, M. M. Rahman, M. S. A. Mamun, and M. S. Kaiser, "Performance Analysis of Machine Learning Approaches in Stroke Prediction," in IEEE *Fourth International Conference on Electronics*, sec. 5. pp 1468. 2020

[4] P. Govindarajan, R. K. Soundarapandian, A. H. Gandomi, R. Patan, P. Jayaraman, and R. Manikandan, "Classification of stroke disease using

machine learning algorithms," Neural Computing and Applications, vol. 32, no. 3, pp. 817–828, Feb. 2020

[5] Zhang, Z. (2004). Naive Bayes and Text Classification I. Foundations and Trends in Information Retrieval, 1(3), 1-99. doi:10.1561/1500000003

[6] Alqahtani, S. A., & Abido, M. A. (2020). K-Nearest Neighbor Algorithm in Medical Data Classification: A Survey. Journal of Healthcare Engineering, 2020, 1-18. https://doi.org/10.1155/2020/7012701

[7] Li, S., "Predicting Stroke Using Logistic Regression in Python," Towards Data Science, 22 May 2019. [Online]. Available: https://towardsdatascience.com/predicting-stroke-using-logistic-regression-in-python-60b59af42c8a. [Accessed: 05-Apr-2023]

[8] G. V. Geetha and V. Kavitha, "Stroke Prediction Using Classification and Regression Tree Analysis," in 2019 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), 2019, pp. 271-275, Doi: 10.1109/ICEECCOT46579.2019.8966091

[9] Krishnamurthi, R. V., Moran, A. E., Forouzanfar, M. H., et al. The Global Burden of Ischemic Stroke: Findings of the GBD 2010 Study. Global Heart, Volume 9, Issue 1, 2014, Pages 107-112, ISSN 2211-8160, https://doi.org/10.1016/j.gheart.2013.12.007. (https://www.sciencedirect.com/science/article/pii/S2211816013002074)

[10] S. Wagavkar, "Correlation Matrix - Introduction," Medium, Nov. 10, 2021. [Online]. Available: https://medium.com/analytics-vidhya/correlation-matrix-5e764bcee34. [Accessed: Apr. 12, 2023].

[11] J. Frost, "Interpreting Correlation Coefficients," Statistics By Jim, [Online]. Available: https://statisticsbyjim.com/basics/correlations/. [Accessed: Apr. 12, 2023]

[12] A. Smith and B. Johnson, "Performance evaluation metrics for linear regression models," IEEE Transactions on Neural Networks and Learning Systems, vol. 26, no. 10, pp. 2319-2331, Oct. 2015.