Use of Different Machine Learning Techniques to Predict Risk of Stroke

Showmitra Roy, Asif Ahmed Tanjid, Sunjida Kabir Shepa, Md. Rahanur Nishan
Computer Science and Engineering Department, AIUB
Dhaka, Bangladesh
royshowmitra@gmail.com
asifahmednirob44@gmail.com
sanjidashifabd2001@gmail.com
rahanurnishanp@gmail.com

***Abstract*—A stroke is a severe medical condition that results from the abrupt stoppage of blood flow to a part of the brain. The discontinuation of blood flow can gradually lead to the death of brain cells which could cause disability and even death depending on the region of the brain that is affected. Since stroke is life-threatening, it is necessary to detect the risk of stroke at an early stage so that this valuable information can be used for the prediction of stroke and help to promote a healthy lifestyle. Therefore, we explore different machine-learning models in order to achieve a robust framework that can be used to evaluate and predict the risk of stroke. A dataset with relevant features is used for this research which is preprocessed to make it suitable for the experiment. Then various machine learning models are selected such as Naive Bayes (NB), Decision Tree (DT), Logistic Regression (LR), K-Nearest Neighbors (KNN), and Support Vector Machine (SVM). The models are applied to the dataset to predict the risk of stroke which gives the accuracy of the models. The results of the experiment show that we were able to achieve a total of 96.05% accuracy for SVM and LR for 6 features whereas the other models varied from 87% to 95%. On the other hand, the results for all features showed over 90% accuracy for all the models and 96.05% accuracy for SVM, LR, and NB which is an improved accuracy compared to previous works on the same dataset.**

***Keywords*—Stroke, risk prediction, machine learning, models, data analysis.**

I. INTRODUCTION

Blood circulation is the most important bodily function since it supplies the body's vital organs with the proper amount of O2 and nutrients needed to operate. When anything prevents blood flow to a portion of the brain or when a blood artery in the brain bursts, a stroke, also known as a brain attack, happens. The brain either ages or suffers harm in both scenarios. A stroke may result in permanent brain damage, chronic disability, or even fatality. Stroke is a medical disorder. A stroke is a medical emergency that requires immediate medical attention. According to the World Health Organization (WHO), fifteen million people worldwide experience strokes every year, with one victim passing away every four to five minutes. According to the Centers for Disease Control and Prevention (CDC), stroke is the sixth most common cause of death in the United States [1]. About 11% of people die from noncommunicable diseases like stroke each year. Approximately 795,000 Americans experience the incapacitating symptoms of strokes on a regular basis [2]. It is the fourth most common cause of death in India. There are two types of strokes: ischemic and hemorrhagic. In a hemorrhagic stroke, a weak blood vessel bursts and bleeds into the brain, while clots stop the drainage in a chemical stroke. Stroke can be prevented by living a healthy, balanced lifestyle that includes giving up bad habits like smoking and drinking, maintaining a normal body mass index (BMI), a normal blood glucose level, and having good kidney and heart function. Prevention of stroke is crucial, and it must be treated quickly to prevent permanent harm or death.

With the advancement of medical technology, it is now possible to use ML techniques to predict the onset of a stroke. The algorithms used in ML are useful because they enable precise prediction and appropriate analysis. Most of the earlier research on strokes concentrated on, among other things, heart attack forecasting. There haven't been many studies on brain stroke. This paper's main goal is to show how machine learning (ML) can be used to predict when a brain stroke will occur. The most significant element of the techniques used, and the conclusions drawn, is that Random Forest performed the best of the four different classification algorithms examined, outperforming them all in terms of accuracy metrics. The model's training on textual data rather than actual brain images has some drawbacks. This study illustrates the application of five ML classification techniques.

II. LITERATURE REVIEW

The medical field has gained massive attention from researchers due to its necessity. Researchers have developed various tools and methods in order to monitor and predict various spectrum of diseases that have an impact on the health of human beings. Thus, the risk of stroke is an important matter that has gained the researchers' interest which has led to the use of a machine learning based approach to predict stroke. In this section, we will illustrate the different works that have utilized machine learning techniques for stroke risk prediction.

Firstly, the authors of the paper [3] have used eight different machine learning algorithms such as Naive Bayes (NB), Random Forest (RF), Logistic Regression (LR), K-Nearest Neighbors (KNN), Stochastic Gradient Descent (SGD), Decision Tree (DT), Multilayer Perceptron (MLP), Majority Voting, and Stacking in order to accurately detect a stroke. For their research, the authors used the dataset [4] from Kaggle which contained information regarding the 11 key features of stroke from 3254 participants. The accuracy of the Naive Bayes was 84%, Random Forest was 79%, Logistic Regression was 81%, K-Nearest Neighbors was 88%, Stochastic Gradient Descent was 91%, Decision Tree was 92%, Multilayer Perceptron was 93%, Majority Voting was 97%, and Stacking was 98%. Stacking had the highest accuracy of 98% in stroke detection in their research paper.

For the research paper [4], the authors utilized a large population-based Electronic Medical Claims database of around 800,000 patients to compare Deep Neural Network (DNN) with three other machine learning algorithms such as Gradient-boosted Decision Tree (GBDT), Logistic Regression (LR), and Support Vector Machine (SVM) to predict 5-year stroke occurrence. The accuracy for Deep Neural Network was 87.3%, Gradient-boosted Decision Tree was 86.8%, Logistic Regression was 86.6%, and Support Vector Machine was 83.9%. Here, Deep Neural Network has the highest accuracy of stroke prediction.

For our research, we have taken the research paper [3] as our primary source of reference since the authors utilize a similar approach to our approach as well as using a Kaggle Dataset.

III. MATERIALS AND METHODS

A. *Dataset Description*

In this paper, the dataset was acquired from Kaggle [5]. The dataset is titled," Stroke Prediction Dataset" and contains 12 features or attributes. The number of participants was 5110, and the attributes are noted below:

, Unique id: This refers to the unique id given to the participants.

, Gender [6]: This feature refers to the gender of the participants. The percentage of men is 41% and the percentage of is 59%.

, Age (years) [6]: This feature refers to the age of the participants.

, Hypertension [7]: This feature refers to the hypersensitiveness of the participants. The number of participants who have hypertension is 498.

, Heart disease [8]: This feature refers to the presence of heart disease among the participants. The number of people who suffer from heart disease is 276.

, Ever married [9]: This feature refers to the marital status of the participants. The percentage of married participants is 66% which is about 3353 participants.

, Work Type [10]: This feature refers to the work status of the participants. This has 3 categories that are private, which is 57%, self-employed which is 16% and others which is 27%.

, Residence Type [11]: This feature refers to the living status of the participants. This has 2 categories, urban which is 51%, and rural which is 49%.

, Avg Glucose Level [12]: This feature captures the average glucose level of the participants.

, BMI[(Kg/m2)][13]: This feature refers to the body mass index of the participants.

, Smoking Status [14]: This feature refers to whether the participants smoke or not. This has 3 categories, never smoked which is 37%, smokes which is 30%, and formerly smoked which is 33%.

, Stroke: This feature represents whether the participant previously suffered a stroke or not. The number of participants who suffered a stroke was 249. Most features are nominal, but the age, average glucose level, and BMI are numerical.

It was not necessary for us to train the models since these are pre-trained models. Thus, the dataset was used for testing to test its accuracy.

B. *Methodology*

For our research, a basic approach was used to conduct the research. First, the stroke prediction Kaggle dataset was loaded in Jupiter as Python was used as the coding language. Then, data pre-processing was used on the dataset. Next, five machine learning models were selected and applied to the dataset to get results. Lastly, an evaluation of the results was conducted to reach a conclusion.

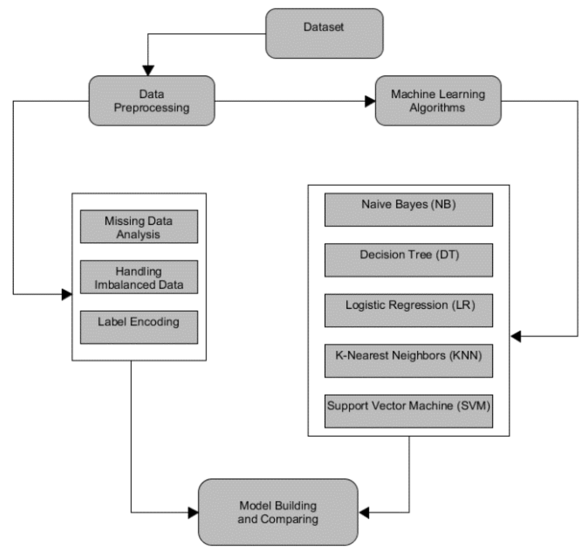

Fig. 1. Steps to our approach

C. *Data Pre-processing*

The initial data is in a raw state and might not be suitable for the accurate prediction of the risk of stroke. Therefore, it might degrade the final prediction quality as data might contain missing or duplicate values. These data are referred to as noisy data and need to be pre-processed. Data pre-processing is necessary to make the data appropriate by redundant values reduction, feature selection as well as data discretization.

In the same way, the data from our Kaggle dataset was not suitable for applying machine learning algorithms. So, we had to pre-process the data to make it suitable. There was initially 12 key features as shown in the following table of figure 2:

The algorithms were applied two times, one time with 6 key features and another time with 11 key features. For our approach, we decided to first remove the unique id feature of the dataset since it had no influence on the risk of stroke. Therefore, we removed the unique id feature using the following algorithm in python:

	id	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi
0	9046	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6
1	51676	Female	61.0	0	0	Yes	Self-employed	Rural	202.21	NaN
2	31112	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5
3	60182	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4
4	1665	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0
...	...	...	...	...	...	...	...	...	...	...
5105	18234	Female	80.0	1	0	Yes	Private	Urban	83.75	NaN
5106	44873	Female	81.0	0	0	Yes	Self-employed	Urban	125.20	40.0
5107	19723	Female	35.0	0	0	Yes	Self-employed	Rural	82.99	30.6
5108	37544	Male	51.0	0	0	Yes	Private	Rural	166.29	25.6
5109	44679	Female	44.0	0	0	Yes	Govt_job	Urban	85.28	26.2

5110 rows × 12 columns

Fig. 2. Table with 12 key features

In [464]:

```
strokesData = strokesData.dropna()
```

In [465]:

```
if 'id' in strokesData :
    strokesData = strokesData.drop('id', axis=1)

strokesData
```

Fig. 3. Dropna function to remove id feature.

In Figure 3 Dropna function is used on the strokes Data to remove id feature since it was in strokes Data. Then, the following algorithm was applied to scale the non-numeric value into a matrix as well as removing null values in order to predict the risk of stroke:

Fig. 5. Algorithm to use on six key features.

	hypertension	heart_disease	avg_glucose_level	bmi	sm
0	0	1	228.69	36.6	for
2	0	1	105.92	32.5	
3	0	0	171.23	34.4	
4	1	0	174.12	24.0	
5	0	0	186.21	29.0	for
...	...	...	...	...	
5104	0	0	103.08	18.6	
5106	0	0	125.20	40.0	
5107	0	0	82.99	30.6	
5108	0	0	166.29	25.6	for
5109	0	0	85.28	26.2	

4909 rows × 6 columns

features as shown in the table in figure.6.

Fig. 6. Table with 6 key features

In [471]:

```
# Splitting the data in training and testing subsets
D = strokesData.values
# iloc to select specefic rows
x = strokesData.iloc[:,:-1]
y = strokesData.iloc[:,-1]
# x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.50, random_state = 1)
```

In [472]:

```
transformingColoumn = ColumnTransformer(transformers = [('encoder', OneHotEncoder(), [0,4,5,6,9])], remainder='passthrough')
x = np.array(transformingColoumn.fit_transform(x))
x

x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.50, random_state = 1)
```

In [473]:

```
sc = StandardScaler()
x_train_scaled = sc.fit_transform(x_train)
x_test_scaled = sc.fit_transform(x_test)
```

Fig. 4. Algorithm to scale non-numeric value.

First, we pre-processed the data to contain only 6 key features from the dataset in order to apply the main features related to stroke. Thus, the following algorithm was used to select only 6 key features as shown in the figure. 5:

In [232]:

```
strokesData = strokesData[['hypertension', 'heart_disease', 'avg_glucose_level', 'bmi', 'smoking_status', 'stroke']]
# strokesData['smoking_status'] = strokesData['smoking_status'].replace('formerly smoked', 1)
# strokesData['smoking_status'] = strokesData['smoking_status'].replace('smokes', 2)
# strokesData['smoking_status'] = strokesData['smoking_status'].replace('never smoked', 0)
# strokesData['smoking_status'] = strokesData['smoking_status'].replace('Unknown', 0)
strokesData = strokesData.dropna()
```

D. Machine Learning Models

We outline the models used in the classification framework for the occurrence of strokes in this section. Various classifier types are used for this purpose.

Naive Bayes: The naive Bayes (NB) classifier was considered, which ensures probability maximization if the features are highly independent. The equation for Naive Bayes is the following:

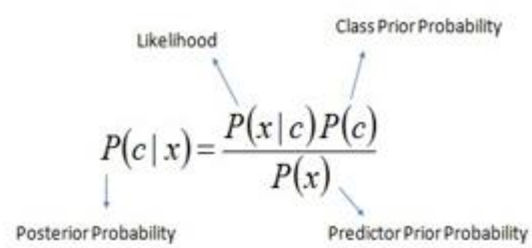

Fig. 7. Naive Bayes Equation

Binary and multi-class classification can both be done using the Naive Bayes technique. Compared to numerical input variables, naive Bayes performs better in cases of categorical input variables. It is helpful for anticipating data and making predictions based on past outcomes. The pseudocode for Naive Bayes is given below:

```
for q = 1… w // loop for each mining models element
    μ[q] = 0; // initialization of mining models elements
end for;
for j = 1… m // loop for each row
    μ[d[j,p]]++; // increment number of row for value x_{j.p} of object x_j;
    for k = 1 … p-1 // loop for each column
        μ[φ(k-1)+(d[j, k]-1)·φ(0)+ d[j, p]]++; // increment number of rows with value x_{j.k}
                                      // and value x_{j.p}, where φ(k)=s+Σ_{q=1}^{k}(|T_q|·s)
    end for;
end for;
```

Fig. 8.Pseudo-code for Naive Bayes

Decision Tree: For the development of a decision tree, we considered J48 as a single classifier and RepTree as a base classifier in the stacking method. A DT's leaf nodes indicate the classes, while its inside nodes stand in for a feature. The former is a quick and easy decision learner that creates a decision tree using information gain as an impurity measure and prunes it using reduced-error pruning, whereas J48 separates a single feature at each node using the Gini index. The pseudo-code for the decision tree is given below:

GenDecTree(Sample S, Features F)

Steps:

1. ***If*** *stopping_condition(S, F) = true* ***then***
   a. *Leaf = createNode()*
   b. *leafLabel = classify(s)*
   c. ***return*** *leaf*
2. *root = createNode()*
3. *root.test_condition =findBestSpilt(S,F)*
4. *V = {v | v a possible outcomecfroot.test_condition}*
5. ***For each*** *value v ∈ V:*
   a. S_v *= {s | root.test_condition(s) = v and s ∈ S };*
   b. *Child = TreeGrowth (*S_v*,F);*
   c. *Add child as descent of root and label the edge {root → child} as v*
6. ***return*** *root*

Fig. 9.Pseudo-code for Decision Tree

Logistic Regression: Logistic regression is a statistical classification method originally developed for binary tasks but extended to multiclass tasks. The model output is a binary variable where p = P (Y = 1) gives the probability that the instance belongs to the 'Stroke' class, so 1 p = P(Y = 0) captures the probability that the instance listened to the "nonstroke" class. Linear relationship between base-b log-odds and model parameters Beta i is as follows:

$$log_b\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 f_{i1} + \ldots + \beta_n f_{in}$$

The pseudo-code for Logistic Regression is given below:

Input: Training data
Begin
For i = 1 to k
For each training data instance d_i.
Set the target value for the regression to $z_i = \frac{y_i - P(1|d_j)}{[P(1|d_j)(1-P(1|d_j))]}$
Initialize the weight of instance d_j to $[P(1|d_j)(1-P(1|d_j))]$
Finalize a $f(j)$ to the data with class value (Z_j) and weight (w_j)
Classical label decision
Assign (class label: 1) if $P_{id} > 0.5$, otherwise (class label: 2)
End

Fig. 10. Pseudo-code for Logistic Regression

K-Nearest Neighbors: The K-Nearest Neighbor (K-NN) classifier is a distance-based method for calculating the similarity or difference between two instances in the dataset being studied. The simplest and most popular measurement is the Euclidean distance. Let f_{new} represent the new sample's characteristics vector, which will be used to determine if it is a stroke or not. The KNN classifier finds the K vectors (neighbors) that are closest to f_{new}. The class to which the majority of f_{new}'s neighbors belong is then assigned. The algorithm of K-Nearest Neighbors is given below:

The pseudocode of classical KNN

Input: X: training data, Y: class labels of X, K: number of nearest neighbors.
Output: Class of a test sample x.

Start
Classify (X,Y,x)
1. ***for*** each sample x ***do***
   Calculate the distance: $d(x, X) = \sqrt{\sum_{i=1}^{n}(x_i - X_i)^2}$
   end for
2. Classify x in the majority class: $C(x_i) = argmax_k \sum_{X_j \in KNN} C(X_j, Y_K)$

End

Fig. 11. Pseudo-code for KNN

Support Vector Machine: Support Vector Machine (SVM) is a supervised machine learning algorithm used for both classification and regression. SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. The algorithm for Support Vector Machine is given below:

Algorithm 1 Training an SVM
Require: $\boldsymbol{X}$ and $\boldsymbol{y}$ loaded with training labeled data, $\boldsymbol{\alpha} \Leftarrow 0$ or $\boldsymbol{\alpha} \Leftarrow$ partially trained SVM
$C \Leftarrow$ some value (10 for example)
repeat
  for all $\{\boldsymbol{x}_i, y_i\}, \{\boldsymbol{x}_j, y_j\}$ **do**
    Optimize α_i and α_j
  end for
until no changes in $\boldsymbol{\alpha}$ or other resource constraint criteria met
Ensure: Retain only the support vectors ($\boldsymbol{\alpha}_i > 0$)

Fig. 12. Pseudo-code for SVM

IV. RESULTS AND DISCUSSION

A. Experiments Setup

For the implementation of the stacking model, four base classifiers were combined. More specifically, naive Bayes, SVM, Decision Tree, Linear Regression and KNN. Their outcomes were fed into a logistic regression meta-classifier. As for the majority voting, we considered the same models with the stacking method, except for naive Bayes. The highest accuracy from the different models is 96.05% for the Support Vector Machine. The same accuracy (96.05%) was for Linear Regression. The least accurate for predicting stroke is 87.21% for Naive Bayes. The KNN model gave 95.23% accuracy which is near the highest accuracy. The accuracy of the Decision Tree is 92.14%.

B. Evaluation

Figure 13 shows the comparison between the stroke and the other features. From figure 13 At the age of 40 to 80 the possibility of stroke is higher.

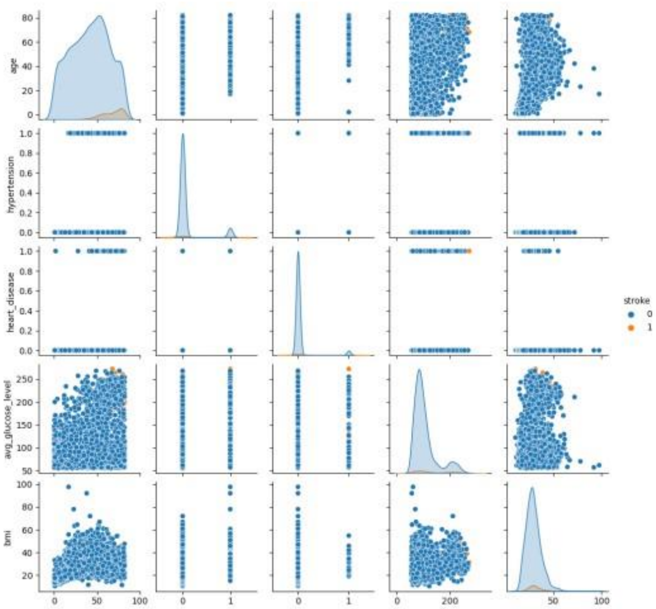

Fig. 13.Compare other features with stroke.

Moreover, the average glucose level gradually increased at this age. Having hypertension makes the high possibility of stroke. Heart disease also causes a stroke. Also, BMI gets higher at the age of 50 and it causes serious damage to the body.

In addition, comparing the average precision and recall, they are either equal, or the former is 0.2–0.3% higher than the latter. A higher difference with a precision lower than the recall is observed in the naive Bayes model. In either case, since the dataset is balanced, the F-measure is a suitable ratio that can reflect the performance (i.e., the accuracy) of the ML models on the dataset.

A limitation of this study is that it was based on a publicly available dataset. These data are of specific size and features as opposed to data from a hospital or institute. Although the latter could give more rich information data models with various features capturing a detailed health profile of the participants, acquiring access to such data is usually time-consuming and challenging for privacy reasons.

Models Accuracy Rate	
Model Name	Accuracy Score
Decision Tree	92.14%
Linear Regression	96.05%
Naive Bayes	87.21%
KNN	95.23%
SVM	96.05%

V. CONCLUSION

A stroke constitutes a threat to a human’s life that should be prevented and/or treated to avoid unexpected complications. Nowadays, with the rapid evolution of AI/ML, clinical providers, medical experts and decision-makers can exploit the established models to discover the most relevant features (or, else, risk factors) for stroke occurrence, and can assess the respective probability or risk.

In this direction, machine learning can aid in the early prediction of stroke and mitigate the severe consequences. This study investigates the effectiveness of various ML algorithms to identify the most accurate algorithm for predicting stroke based on several features that capture the participants’ profiles.

The performance evaluation of the classifiers using AUC, F-measure (which summarizes precision and recall) and accuracy is essentially suitable for the models’ interpretation, demonstrating their classification performance. In addition, they reveal the models’ validity and predictive ability in terms of the stroke class. Stacking classification outperforms the other methods, with an AUC of 96.05%, F-measure. Hence, a stacking method is an efficient approach for identifying those at high risk of experiencing a stroke in the long term. The AUC values show that the model has a high predictive ability and distinguishability among the two classes. The future purpose of this study is to enhance the ML framework via the employment of deep learning methods. Finally, a challenging but promising direction is to collect image data sensors 2022, 22, 4670 11 of 13 from brain CT scans and to evaluate the predictive ability of deep learning models in stroke occurrence.

ACKNOWLEDGMENT

The authors would like to thank DR. MD Asraf Ali Sir for his diligent guidance and the thorough way he taught us everything. Without his incredibly patient and constructive criticism, this project would not have been possible. Without the help of our fellow group members, we as group partners could not have started this project.

REFERENCES

[1] “About stroke,” Nov 2022. [Online]. Available: https://www.cdc.gov/stroke/about.htm?fbclid=IwAR2VD-RTwIDLOECyPkzoTOsvzJrl8UDfGvFmjC6Zaa12MAyOrrg0zEjLyw#:,,:text=A%20stroke%2C%20sometimes%20called%20a, term%20disability%2C%20or%20even%20death

[2] Tazin, T., Alam, M. N., Dola, N. N., Bari, M. S., Bourouis, S., and Monirujjaman Khan, M., “Stroke disease detection and prediction using robust learning approaches,” Nov 2021. [Online]. Available: https://www.hindawi.com/journals/jhe/2021/7633381/?fbclid=

IwAR3XpfsouLk4D-Bnbx94mK6DG9VvRau9DZ3NRfZx3ZUkJ7kV6sggbY7iekw

[3] Dritsas, E. and Trigka, M., “Stroke risk prediction with machine learning techniques,” Sensors, vol. 22, no. 13, p. 4670, 2022.

[4] Hung, C.-Y., Chen, W.-C., Lai, P.-T., Lin, C.-H., and Lee, C.-C., “Comparing deep neural network and other machine learning algorithms for stroke prediction in a large-scale population-based electronic medical claims database,” in 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, 2017, pp. 3110–3113.

[5] Fedesoriano, “Stroke prediction dataset,” Jan 2021. [Online]. Available:

https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset

[6] Rexrode, K. M., Madsen, T. E., Yu, A. Y., Carcel, C., Lichtman, J. H., and Miller, E. C., “The impact of sex and gender on stroke,” Circulation Research, vol. 130, no. 4, pp. 512–528, 2022.

[7] Dubow, J. and Fink, M. E., “Impact of hypertension on stroke,” Current atherosclerosis reports, vol. 13, no. 4, pp. 298–305, 2011.

[8] Tsao, C. W., Aday, A. W., Almarzooq, Z. I., Alonso, A., Beaton, A. Z.,

Bittencourt, M. S., Boehme, A. K., Buxton, A. E., Carson, A. P., Commodore-Mensah, Y. et al., “Heart disease and stroke statistics—2022 update: a report from the american heart association,” Circulation, vol. 145, no. 8, pp. e153–e639, 2022.

[9] Andersen, K. and Olsen, T., “Stroke case-fatality and marital status,” Acta Neurologica Scandinavica, vol. 138, no. 4, pp. 377–383, 2018.

[10] Addo, J., Ayerbe, L., Mohan, K. M., Crichton, S., Sheldenkar, A., Chen, R., Wolfe, C. D., and McKevitt, C., “Socioeconomic status and stroke: an updated review,” Stroke, vol. 43, no. 4, pp. 1186–1191, 2012.

[11] Howard, G., “Rural-urban differences in stroke risk,” Preventive Medicine, vol. 152, p. 106661, 2021.

[12] Cai, Y., Wang, C., Di, W., Li, W., Liu, J., and Zhou, S., “Correlation between blood glucose variability and the risk of death in patients with severe acute stroke,” Revue Neurologique, vol. 176, no. 7, pp. 582–586, 2020. [Online]. Available: https:

//www.sciencedirect.com/science/article/pii/S0035378719309166

[13] Elsayed, S. and Othman, M., “The effect of body mass index (bmi) on the mortality among patients with stroke.” European Journal of Molecular and Clinical Medicine, vol. 8, no. 4, pp. 181–187, 2021.

[14] Shah, R. S. and Cole, J. W., “Smoking and stroke: the more you smoke the more you stroke,” Expert review of cardiovascular therapy, vol. 8, no. 7, pp. 917–932, 2010.