Factors Affecting Brain Stroke: A Statistical Analysis

Pratik Prakash Brahmapurkar

i. Abstract

When an area of the brain loses its blood supply, a catastrophic medical condition known as a stroke may develop, which poses a significant risk to the patient's life. (NHS, 2022)

The second leading cause of death and disability around the world is stroke. Some brain cells die quickly when an artery gets blocked or breaks because they don't get enough oxygen. Strokes are also one of the main reasons why people get dementia and feel sad. Low- and middle-income countries have the most stroke-related deaths and disability-adjusted life years. 3–5 In the past 40 years, the number of strokes in low- and middle-income countries has gone up. In this time, the number of strokes in countries with high incomes has dropped by 42%. Strokes happen 15 years earlier and kill more people in low- and middle-income countries than in high-income countries. Strokes mostly happen to people when they are at their best. This developing disaster hasn't gotten much attention, even though it has a big impact on the social and economic progress of countries. (Johnson, et al., 2016)

Several different machine learning models would be constructed, and then a comprehensive conclusion would be drawn, along with some suggestions for future study and areas where further investigation is needed.

Contents

1. Introduction

Stroke has a big effect on the economy and society all over the world. Stroke is getting more and more attention from the media, patients and caregivers, service improvements, and research. It is thought that 4.5 million people die every year from a stroke, and over 9 million people live with the effects of a stroke. If they live to be 85, almost one in four men and almost one in five women age 45 can expect to have a stroke. Stroke happens to about 2-2.5 people out of every thousand people. Over 5 years, there is a 15–40% chance that it will happen again. By 2023, about 30% more people will have their first stroke than in 1983. There are about 5 out of every thousand people who have it. One year after having a stroke, 65 percent of survivors are able to live on their own. Stroke is the leading cause of disability in adults. (Wolfe, 2000)

A sudden stroke is much more than ever been a medical emergency that needs a quick response from EMTs and neurologists. Negative ideas about how to treat strokes are now out of date, because the future looks bright for stroke patients. (Hill & Hachinski, 1998)

2. Causes

Brain attacks or strokes are sudden problems with the nervous system that cause sudden weakness or numbness on one side of the body, trouble speaking or understanding language, sudden blindness, or sudden trouble walking. This happens when a blockage in an artery cuts off blood flow to a part of the brain, and the symptoms are related to that part of the brain. FAST is an acronym that was made to make people more aware of stroke symptoms and help people in the community recognise them better. (Kulkarni, 2021)

The brain's tissues are damaged by a lack of blood supply. Stroke symptoms appear in the portions of the body whose functions are regulated by the affected regions of the cerebral cortex. Those who have a stroke and seek medical attention as soon as possible have a better prognosis. Therefore, knowing the symptoms of a stroke is essential in order to take rapid action in the event of a stroke. (Kohli, 2021)

There are two main reasons why people have strokes: a blocked artery (ischemic stroke) or a blood vessel that leaks or bursts (hemorrhagic stroke). Some people may have a transient ischemic attack (TIA), which is a short-term blockage of blood flow to the brain that doesn't cause long-term symptoms. (Brown, 2022)

A. Ischemic stroke: This is the type of stroke that happens most often. It happens when the blood vessels in the brain get narrowed or blocked, causing very little blood to flow to the brain (ischemia). Blood vessels can get blocked or narrowed when fatty deposits build up in them or when blood clots or other debris travel through the bloodstream, usually from the heart, and get stuck in the blood vessels in the brain.
B. Transient ischemic attack (TIA): A transient ischemic attack (TIA), which is sometimes called a "ministroke," is a short period of time when stroke-like symptoms happen. A TIA happens when part of the brain doesn't get enough blood for a short time. This can last as less as 5 min. A TIA happens when a clot or piece of debris slows or stops blood flow to a nerves, just like an ischemic stroke. The symptoms alone can't tell you if someone is having a stroke or a TIA.
C. Hemorrhagic stroke: A hemorrhagic stroke occurs when an artery in the brain bursts open or releases blood, causing bleeding to occur. The blood from that artery causes an increase in

pressure in the skull and causes the brain to enlarge, causing damage to brain cells and other structures. (Kohli, 2021)

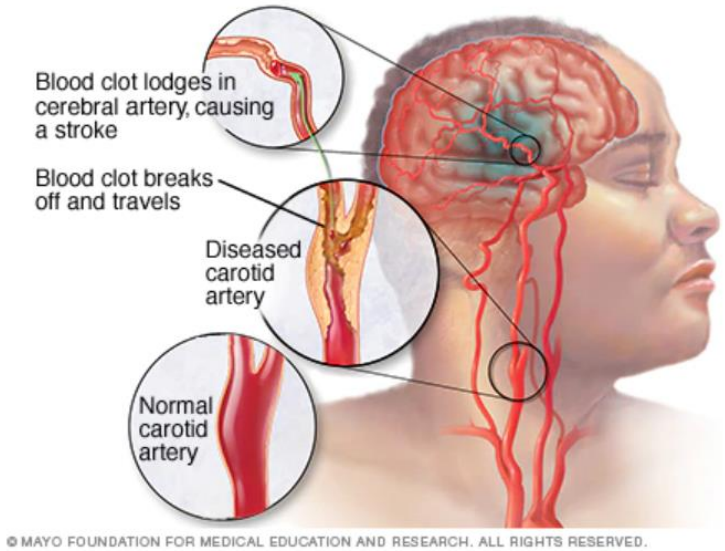

Figure 1 Shows Ischemic stroke

Other Causes of Brain Stroke are: (Kulkarni, 2021)

1. High Blood Pressure: High blood pressure is still one of the leading causes of heart disease, brain disease, and death. Because of this, early diagnosis and treatment go a long way toward preventing complications. Several changes to the way you eat, such as eating less sodium and alcohol and eating more fruits, vegetables, legumes, and low-fat dairy products and less meat, sweets, and saturated fats. Stopping smoking, starting a regular aerobic exercise routine, and living a stress-free life can do a lot to lower blood pressure and keep its complications from happening.
2. Diabetes: Diabetes is still one of the main causes of death, and it is also one of the main causes of stroke, heart disease, and peripheral arterial disease. Diabetes is less likely to happen if you eat a lot of fruits, veggies, nuts, whole grains, and olive oil. It is very important to find people who have pre-diabetes, also called IFG and IGT. Patients with a family history of the disease and other risk factors must be tested often so that they can be treated early and avoid problems.
3. Smoking: Smoking is among the major causes of strokes as well as heart attacks, bronchitis, and lung cancer. The harmful levels of SPM in the air we breathe are very bad for our health. Add to that the bad things that happen when individuals smoke, and have a double whammy. Studies are currently being done to find out how air pollution affects the arteries in the brain and how well it works. This would not be surprising if these studies show that air pollution has a serious negative effect on these parameters.
4. Higher Cholesterol Level: A brain attack is more likely to happen if you have high cholesterol. High blood cholesterol can cause problems, but they can be avoided if they are found and treated quickly. Avoiding a high-fat diet and working out regularly are two of the most important ways to avoid problems.

5. Atrial Fibrillation: Atrial fibrillation is among the most common causes of stroke that people don't know about. This is a very familiar problem with the way the heart beats. It happens more often after age 70 and when there are other heart problems. This condition can be found with the help of a Holter monitor and a loop recorder. It is very important to recognise this condition because it requires a stronger blood thinner (anticoagulants) to prevent strokes as well as other medicines to keep the heart rate under control.

3. Symptoms and Treatment

Getting better after a stroke can be a hard and emotional process that is different for each person. The chance of getting better depends on where the lesion is, how big it is, how much tissue it affects, how long it has been since it was treated, and other things. But experts have found a general pattern in how motor skills get better after a stroke. (Bence, 2022)

A stroke can cause sudden numbness or weakness, especially on one side of the body. If your vision in one or both eyes changes, or if you have trouble swallowing, No idea why I have a bad headache, Trouble getting dizzy, walking, or keeping your balance, Confused, can't talk or understand others. The FAST test can help find signs. It means: (DerSarkissian, 2022)
FACE, hanging down, Request a smile.
ARMS that are weak or numb.
ARMS that are weak or numb.
Speech, can the person say something simple? Do they have trouble speaking or mumble?
Time to call 911

Figure 2 Shows FAST test helps spot symptoms for Stroke (DerSarkissian, 2022)

4. Methodology

A. **Data Collection and Loading**: The.csv file for the Stroke Prediction Dataset can be downloaded from Kaggle. The location where the Stroke Prediction data is saved has been set as the work directory in R-studio. The CSV file is then read into R-studio, and the target variable is "stroke." There are 5110 observations inside the data frame, which also has 12 attributes. Furthermore, all of the needed libraries have already been loaded.

B. **Summarizing the Data: T**he summary function is in responsible of analysing the whole set of data. It was observed that there were no NA values in the data. This makes the data set clean, and the colSums(is.na) function was used to analyze each and every column. Cross table function was also used to analyze the data with respect to stroke column.

C. Data Cleaning: The data which was downloaded from kaggle was a clean data set with no NA values. The gender of one patient who had previously identified as 'other' was altered to Male. But in BMI column there consisted almost all the numeric value but there were few text value as well. The text value from the BMI column was removed and then the blank field was replaced by the mean BMI values. The categorical variables like gender, hypertension, heart_disease, ever_married, work_type, Residence_type, smoking_status and stroke were converted to factors for classification purpose.

D. Descriptive Statistic: A straightforward method for computing descriptive statistics on datasets is provided by the Crosstable package, which is based around a single function called crosstable(). The Cross Table function allows for the simultaneous analysis of two variables. Every other characteristic is weighed against the goal variable, which is the stroke.

E. **Visualization using GGPLOT2:** Bar plots are being constructed with respect to the target variable stroke. The graph gets the function geom text(). It requires that mappings be made for x, y, and the text itself. By setting vjust, it can move the text above or below the tops of the bars (vertical justification). Putting names on a bar graph that shows counts instead of values. hjust = 0.5 puts the plot titles in the middle. stat = "identity" shows the number of cases in each group by the height of the bar.

F. Splitting the dataset: The data is split into two sets, called a training set and a test set, with an 80:20 split. To start, the Index must be trained using the createDataPartition() method, with the target value stroke as the training variable. We'll keep p=0.8 because we want 80% of the training set and 20% of the remaining set for the test set. In the same way, the partitioning is done based on what each model needs.

G. Logistic Regression: Logistic Regression is a type of classification model in the world of Machine Learning. This implies that logistic regression models have a fixed number of parameters that change based on the number of input features. Based on the other parameters, they make

categorical predictions, like whether a patient had a brain stroke or not. All the variables are used except the id column. In this case the type is used as response.

H. **Decision Tree**: To grow a tree, you have to choose what parts to use and how to split them, and you also have to know when to stop. Since trees grow at random, we will need to cut them down to make them look nice. Similarly, in this case decision tree is used to identify the important variables which are important part of the decision. Here, the split of data is done 50-50 and Accuracy is been observed for both test as well as train data set.

I. **Naives Bayes:** Naive Bayes models are a group of classification algorithms that are very fast and easy to use. They are often good for very high-dimensional datasets. They are very useful as a quick and dirty base point for a classification task because they are fast and have few parameters that can be changed. The major objective naïve_bayes() finds the class of each characteristic in the dataset and, depending on what the user chooses, assumes that each feature could have a different distribution.

J. **Random Forest:** Random forest, as its name suggests, is made up of a lot of different decision trees that work together as a whole. Each tree in the random forest predicts a class, and the class that gets the most votes is the one that our model predicts. "proximity = TRUE" in random forest means that two cases are "close" or "near" to each other. For every pair of cases, observations, or sample points, the distance between them is calculated. If two cases are at the same tree's end node, they are one step closer to each other. At the end of the run of all trees, the distances are normalised by dividing by the number of trees. Proximities are used to fill in missing data, find outliers, and make low-dimensional views of the data that are clearer.

K. **Deep Neural Network:** A Deep Neural Network (DNN) is an artificial neural network (ANN) that has many hidden layers of units between the input and output layers. Deep Neural Networks can model complex, non-linear relationships. Most DNNs are built as feedforward networks, but recurrent neural networks have been used in research with great success.

L. **eXtreme Gradient Boosting (XG Boost):** eXtreme Gradient Boosting is an implementation of gradient-boosted decision trees that is designed for speed and performance. The name xgboost, on the other hand, refers to the engineering goal of pushing the limit of computational power for boosted tree algorithms. This is why so many individuals use xgboost. Caret package is used to do Cross Validation and Hyper - parameters tuning using grid search technique. First, the trainControl() function is used to set the method of cross-validation to be used and the type of search, such as "grid" or "random." Then train the model by using the train() function with tuneGrid as one of the arguments. Then finally xgboost model to make predictions about the testing data and predict the "stroke" rate and performance measures.

M. **Comparison of all models:** Each model, or any machine learning method, contains a number of properties that allow it to analyse data in a variety of ways. The data that is given into these algorithms is often varied depending on the stage of the experiment that was completed before. Different models are compared on the basis of it’s accuracy level.

5. Descriptive Statistics

Stroke	Female	Male
No	2853	2008
Yes	141	108
% Yes	4.709419	5.10397

Table 1 Shows Stroke content with respect to gender

Stroke	Hypertension	
	No	Yes
No	4429	432
Yes	183	66
% Yes	3.96791	13.25301

Table 2 Shows stroke with respect to Hypertension

Stroke	Heart Disease	
	No	Yes
No	4632	229
Yes	202	47
% Yes	4.178734	17.02899

Table 3 Shows stroke with respect to Heart Disease

Stroke?	Formerly Smoked	Never Smoked	Smokes	Unknown
No	815	1802	747	1497
Yes	70	90	42	47
% Yes	7.90960452	4.756871036	5.323194	3.044041

Table 4 Shows stroke with respect to Smokers

6. Visualization

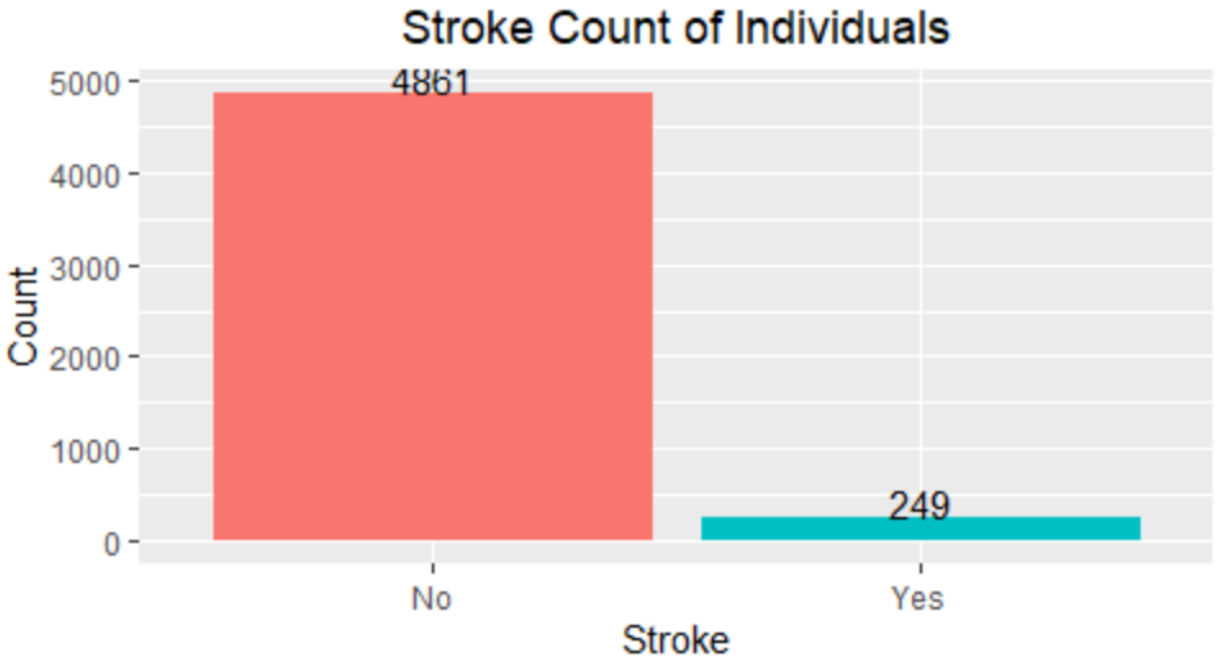

Figure 3 Shows with and without stroke count

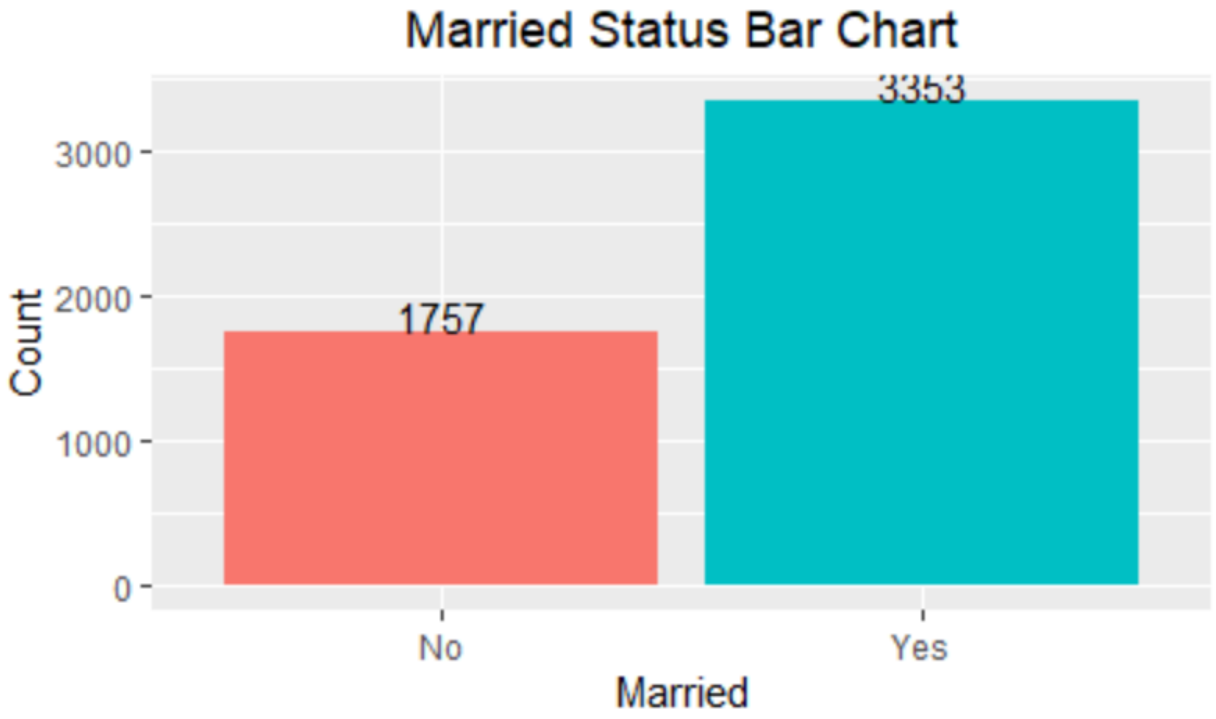

Figure 4 Shows Married Status

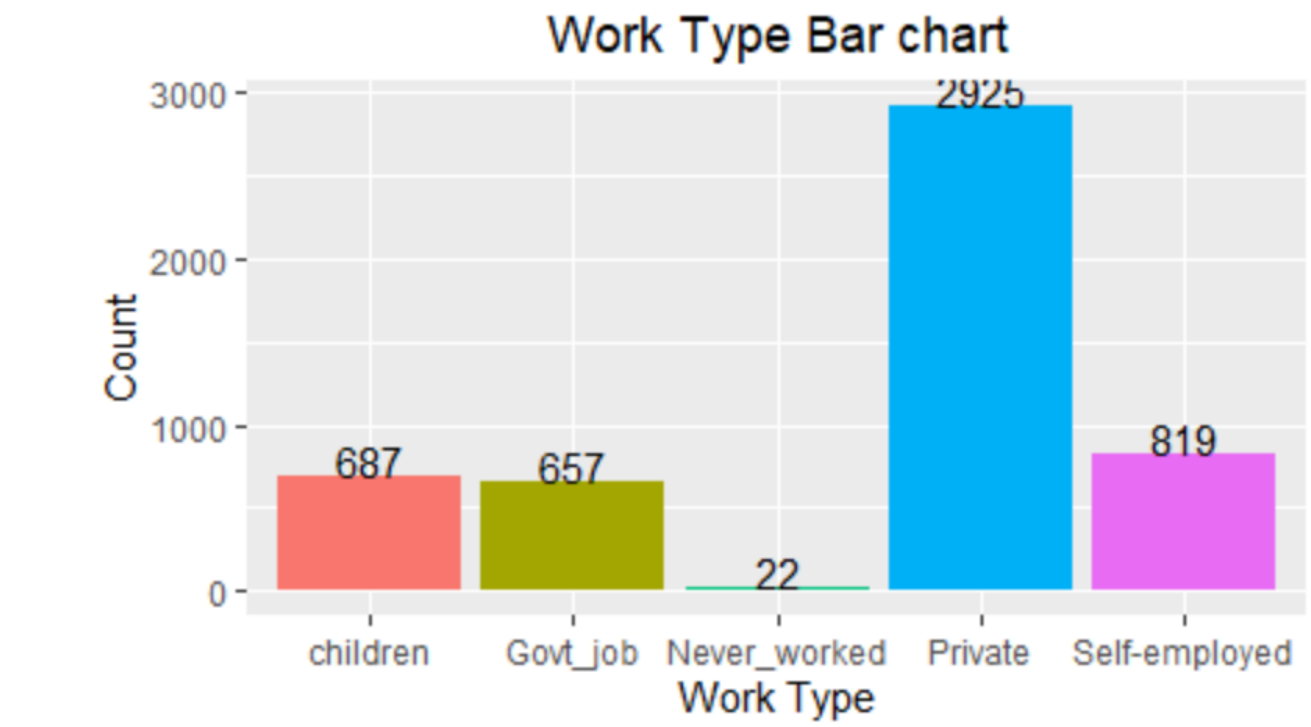

Figure 5 Shows Work Type Status

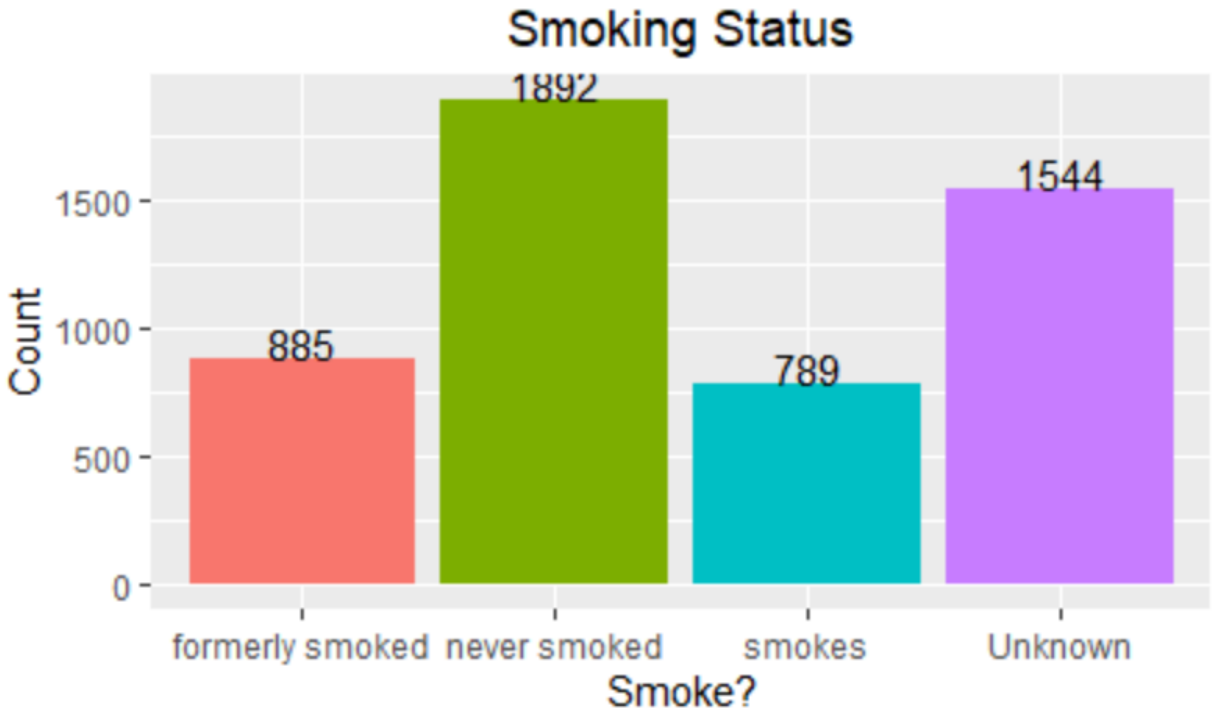

Figure 6 Shows Individuals Smoking status

7. Machine Learning Models for Classification

A. **Logistic Regression**

```
Call:
glm(formula = formula1, family = "binomial", data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.0578  -0.3249  -0.1669  -0.0919   3.4964

Coefficients:
                             Estimate Std. Error z value             Pr(>|z|)
(Intercept)                -6.667629   0.804542  -8.287 < 0.0000000000000002 ***
genderMale                  0.056646   0.157950   0.359              0.71987
hypertension1               0.260486   0.189539   1.374              0.16935
heart_disease1              0.266213   0.215213   1.237              0.21610
ever_marriedYes            -0.203994   0.254845  -0.800              0.42344
work_typeGovt_job          -1.155043   0.865063  -1.335              0.18181
work_typeNever_worked     -10.574538 351.472436  -0.030              0.97600
work_typePrivate           -0.966265   0.843792  -1.145              0.25215
work_typeSelf-employed     -1.428731   0.870796  -1.641              0.10086
Residence_typeUrban         0.147237   0.154501   0.953              0.34060
smoking_statusnever smoked -0.253973   0.198457  -1.280              0.20064
smoking_statussmokes        0.194369   0.234916   0.827              0.40801
smoking_statusUnknown      -0.083547   0.231876  -0.360              0.71862
age                         0.074568   0.006490  11.490 < 0.0000000000000002 ***
avg_glucose_level           0.004156   0.001332   3.121              0.00181 **
bmi                         0.003559   0.012546   0.284              0.77666
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1597.1  on 4088  degrees of freedom
Residual deviance: 1280.9  on 4073  degrees of freedom
AIC: 1312.9

Number of Fisher Scoring iterations: 14
```

Figure 7 Shows Logistic Regression Model Summary

```
> confusionMatrix(class_pred1, test$stroke)
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 972  49
         1   0   0

               Accuracy : 0.952
                 95% CI : (0.937, 0.9643)
    No Information Rate : 0.952
    P-Value [Acc > NIR] : 0.5379

                  Kappa : 0

 Mcnemar's Test P-Value : 0.000000000007025

            Sensitivity : 1.000
            Specificity : 0.000
         Pos Pred Value : 0.952
         Neg Pred Value :   NaN
             Prevalence : 0.952
         Detection Rate : 0.952
   Detection Prevalence : 1.000
      Balanced Accuracy : 0.500

       'Positive' Class : 0
```

Figure 8 Shows Accuracy and Confusion Matrix for Logistic Regression

B. Decision Tree

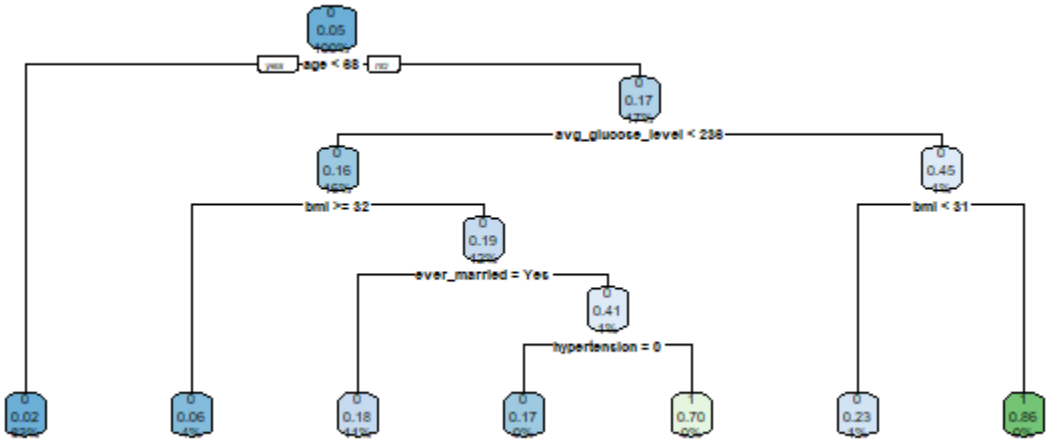

Figure 9 shows Decision Tree split

```
Classification tree:
rpart(formula = stroke ~ . - id, data = train)

Variables actually used in tree construction:
[1] age
[2] avg_glucose_level
[3] bmi
[4] ever_married
[5] hypertension

Root node error: 123/2559 = 0.048066

n= 2559

        CP nsplit rel error xerror
1 0.01355      0   1.00000      1
2 0.01084      3   0.95935      1
3 0.01000      6   0.92683      1
      xstd
1 0.087973
2 0.087973
3 0.087973
```

Figure 10 Shows Variable used in decision Tree

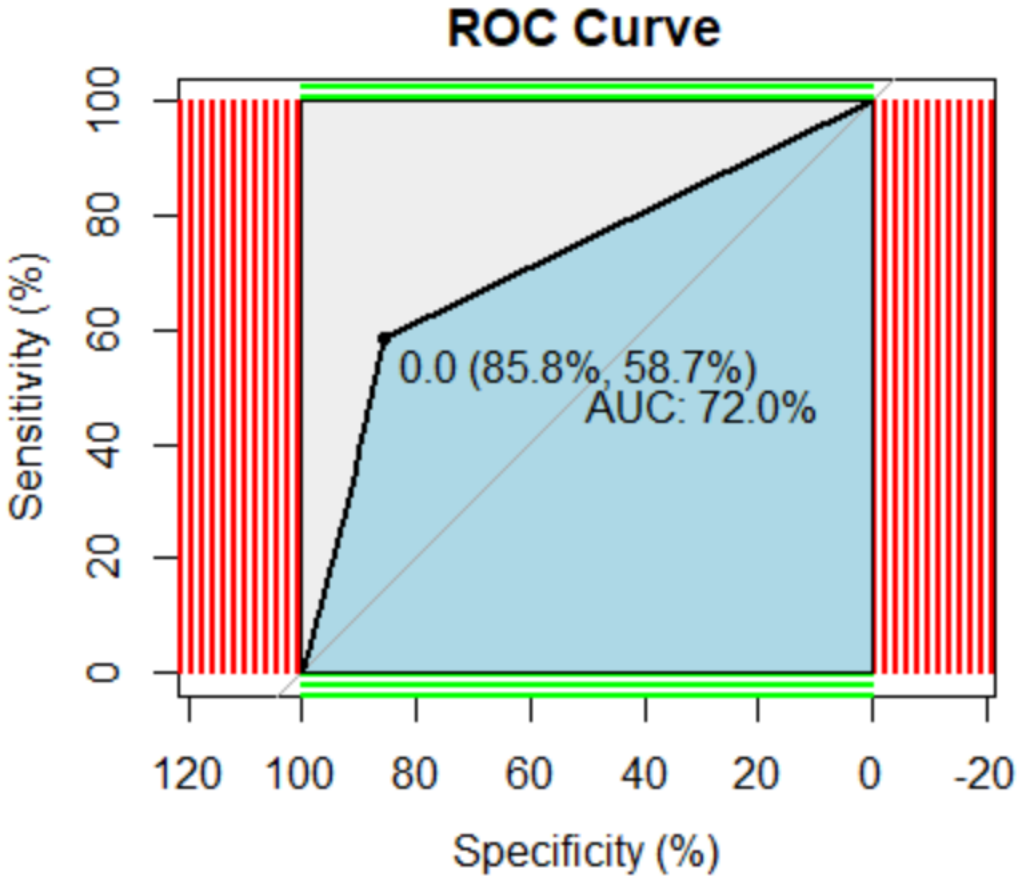

Figure 11 Shows ROC Curve for Decision Tree Model

```
> confusionMatrix(p, train$stroke, positive='1
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 2432  110
         1    4   13

               Accuracy : 0.9555
                 95% CI : (0.9467, 0.9631)
    No Information Rate : 0.9519
    P-Value [Acc > NIR] : 0.2176

                  Kappa : 0.1761

 Mcnemar's Test P-Value : <0.0000000000000002

            Sensitivity : 0.105691
            Specificity : 0.998358
         Pos Pred Value : 0.764706
         Neg Pred Value : 0.956727
             Prevalence : 0.048066
         Detection Rate : 0.005080
   Detection Prevalence : 0.006643
      Balanced Accuracy : 0.552025

       'Positive' Class : 1
```

Figure 12 Shows Confusion Matrix and Accuracy of Decision Tree

C. **Naives Bayes**

```
> (tab1 <- table(p1, train$stroke))

p1     0    1
  0 3708  160
  1  166   42
> sum(diag(tab1)) / sum(tab1)
[1] 0.9200196
```

Figure 13 Shows Confusion Matrix and Accuracy for Naives Bayes model

D. Random Forest

```
> confusionMatrix(p1, train$stroke)
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 3406    7
         1    0  165

               Accuracy : 0.998
                 95% CI : (0.996, 0.9992)
    No Information Rate : 0.9519
    P-Value [Acc > NIR] : < 0.0000000000000002

                  Kappa : 0.9782

 Mcnemar's Test P-Value : 0.02334

            Sensitivity : 1.0000
            Specificity : 0.9593
         Pos Pred Value : 0.9979
         Neg Pred Value : 1.0000
             Prevalence : 0.9519
         Detection Rate : 0.9519
   Detection Prevalence : 0.9539
      Balanced Accuracy : 0.9797

       'Positive' Class : 0
```

Figure 14 Shows Accuracy and Confusion Matrix of Random Forest Model

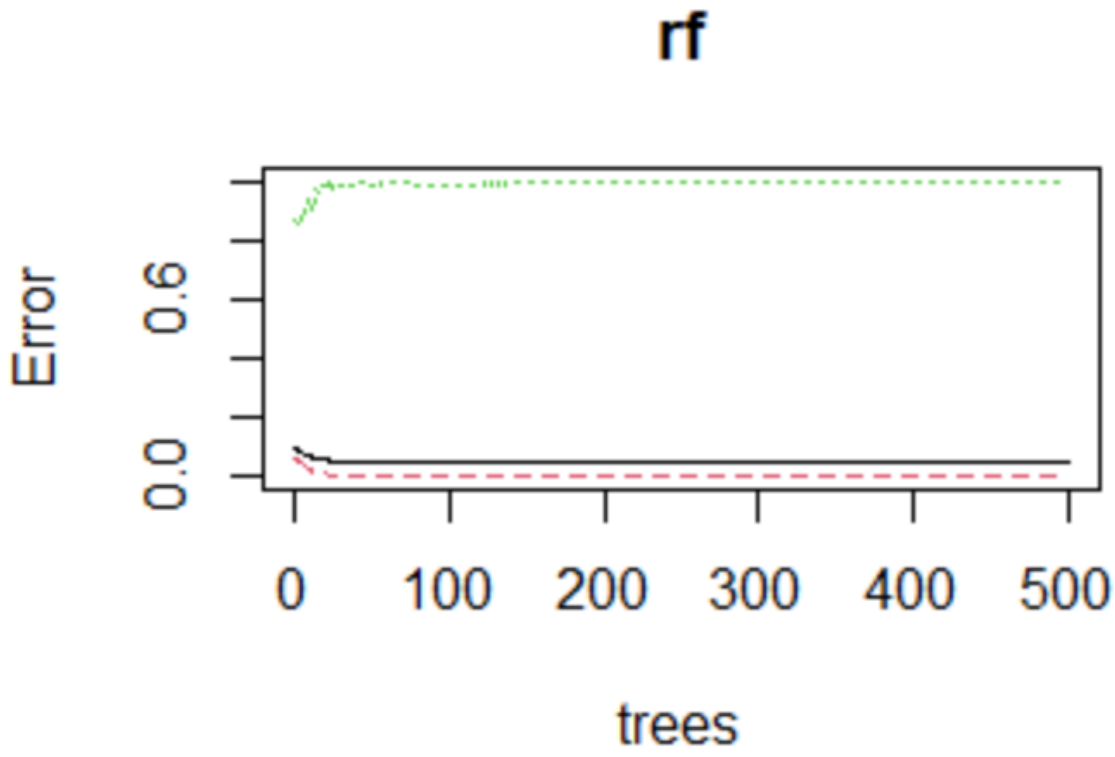

Figure 15 Shows Error rate of random forest model

E. **Deep Neural Network**

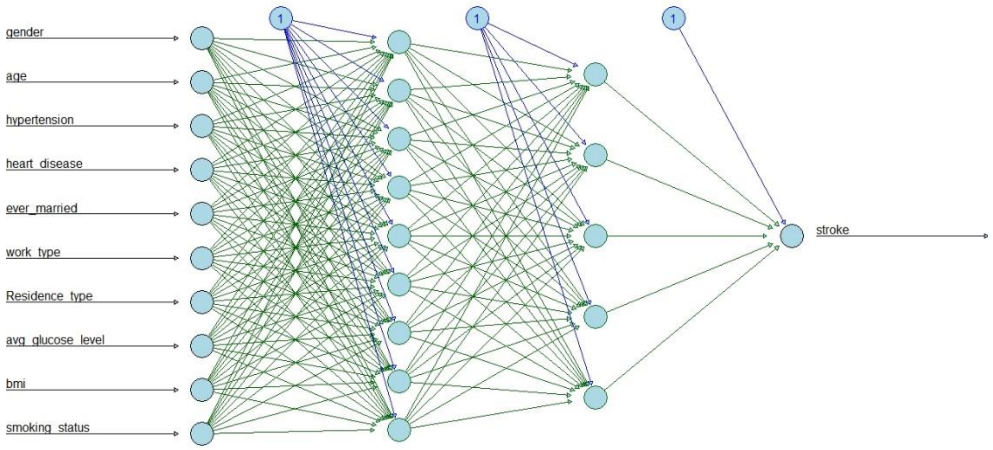

Figure 16 Shows Neural Network model visual identification

F. XG Boost (eXtreme Gradient Boosting)

```
> xgb_model
eXtreme Gradient Boosting

3578 samples
  11 predictor
   2 classes: '0', '1'

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 2863, 2862, 2862, 2862, 2863
Resampling results:

  Accuracy   Kappa
  0.9485733  0.04977373

Tuning parameter 'nrounds' was held constant at a value of
 0
Tuning parameter 'subsample' was held constant at a value
 of 0.5
```

Figure 17 Shows Accuracy and Kappa for XGBOOST Model

```
> confusionMatrix(xgb_pred, test$stroke, positive = '1')
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1445   72
         1   10    5

               Accuracy : 0.9465
                 95% CI : (0.934, 0.9572)
    No Information Rate : 0.9497
    P-Value [Acc > NIR] : 0.7433

                  Kappa : 0.0938

 Mcnemar's Test P-Value : 0.00000000001624

            Sensitivity : 0.064935
            Specificity : 0.993127
         Pos Pred Value : 0.333333
         Neg Pred Value : 0.952538
             Prevalence : 0.050261
         Detection Rate : 0.003264
   Detection Prevalence : 0.009791
      Balanced Accuracy : 0.529031

       'Positive' Class : 1
```

Figure 18 shows Accuracy and Confusion Matrix of XG Boost model

8. Finding and Results

It is clear from looking at table 1 that the number of female patients is much higher than the number of male patients. In addition, the proportion of men who have a stroke is much greater than that of women. According to Table 2, those who suffer from hypertension have an increased risk of experiencing a stroke. In the neighborhood of thirteen percent of individuals who also had hypertension, a stroke occurred. The argument holds true for heart disease as well in a similar fashion. Patients suffering from heart disease had a stroke incidence of around 17 percent. It is possible to draw the conclusion from Table 3 that patients who smoke now and who have smoked in the past have a higher risk of experiencing a stroke throughout their

lifetime. From figure 6 it can be deduced that most of the patients never smoked. From Figure 3 the number of people who have not suffered from strokes is much higher than the number of individuals who have experienced this medical emergency. Based on the data shown in figure 4, we can see that about twice as many patients had previously been married as those who have never been married. It is possible to extrapolate from figure 5 that there are nearly equal numbers of patients who work for the government, who are self-employed, and who are children. The vast majority of patients are employed by private companies, whereas a minority of them have never had a job.

From figure 7 it can be deduced that only 2 variables are significant which are age and average glucose level. As they have the significance value less than 0.05. It can also be noticed that the model is significant. The estimated value of the coefficient for the variable heart disease is b = 0.266, which indicates a positive correlation. This indicates that there is a correlation between having a higher prevalence of heart disease and having a higher risk of suffering a stroke. On the other hand, the coefficient for the variable Work Type that corresponds to never having worked is b = -10.57, which is a negative number. This suggests that a higher proportion of individuals who have never held paid employment will be linked with a lower risk of suffering a stroke.It is possible to deduce from figure 8 that the accuracy level is 0.952 with the confusion matrix being shown.

It is clear from looking at figure 9 that those with an age greater than 68 have a significantly increased risk of suffering a stroke. After that, the age factor that is more than 68 is divided by the average glucose level. This indicates that a substantial association exists between the average glucose level and age in terms of the risk of stroke. After that, the average glucose level that is over 238 is divided into BMI. People with a body mass index (BMI) of 32 or less or more than that have an increased risk of having a stroke. And then there is a distinction made based on whether or not the individual is married. The divides provide the foundation for the analysis that is carried out. The variables that were employed in the development of the tree are shown in Figure 10. The receiver operating characteristic curve is shown in Figure 11. This curve demonstrates how the True Positive Rate and the False Positive Rate vary as the criteria does. The model's accuracy is shown to be 0.955 using the confusion matrix in figure 12.

Figure 13 allows us to not only examine the confusion matrix but also reveals that the accuracy level for the Naive Bayes method is 0.92001. This information can be seen by referring to the figure.

Based on figure 14, we are able to see that the accuracy level achieved by Random Forest is quite close to 1, coming in at 0.998. In this context, we may also take note of the Kappa value, which is 0.9782. A Kappa value of 0 indicates that the raters' opinions are in random agreement with one another, while a score of 1 indicates that the raters' opinions are in full agreement with one another. The score that we received indicates that we are in virtually complete agreement. The model of the error rates for the Random Forest Model is shown in Figure 15.

Figure 16 shows the neural network, where each node gets all of the given parameters as input. Most deep networks are feed-forward, which means they only go from input to output in one direction. But we can also train the model using back propagation, which means going from the output to the input. Backpropagation lets us figure out and assign the error to each neuron, which lets us adjust and fit the algorithm to work better.

The accuracy of the eXtreme Gradient Boosting (XGBoost) algorithm is shown to be 0.9465 when compared with the confusion matrix in figures 17 and 18. The Kappa score of 0.0938 indicates that there is a random agreement between the different points of view.

Machine Learning Model	Accuracy
Logistic Regression	0.952
Decision Tree	0.955
Naives Bayes	0.92
Random Forest	0.998
XGBoost	0.9465

Table 5 shows accuracy comparison between different types of Models

From Table 5 it can be observed that Random forest has the highest accuracy amongst all. And the lowest accuracy was for Naives Bayes.

9. Conclusion

After doing research on brain strokes and analysing the results, the it cane concluded that a patient's age and body mass index are the two most important criteria in determining whether or not the patient would have a stroke. Ages more than 65–70 years old may be given priority. It has also been shown that patients who smoke or who have smoked have a greater chance of experiencing a stroke than those who have never smoked. In addition, people who suffer from hypertension or heart disease are at a greater risk of experiencing a stroke. It has been determined that random forest is the most effective model for assessing this specific data, with an accuracy level of 99.8 percent. It has also been recommended that patients who are over the age of 60 or patients who have heart disease or hypertension must be diagnosed with some tests every three months. These tests could include a blood test to determine your cholesterol and blood sugar levels or a measurement of your blood pressure. (NHS, 2022) It is also possible to recommend that the patient bring along a wearable gadget similar to SWORD with them at all times. The Stroke Wearable Operative Rehabilitation Device, or SWORD, is a device that provides a patient with an exercise programme on a tablet computer. The application delivers directions to the patient, whose motions are recorded by sensors that are attached to the body. (Reuters, 2017)

References

Bence, S., 2022. *The 7 Stages of Stroke Recovery.* [Online]
Available at: https://www.verywellhealth.com/stroke-recovery-stages-5213006
[Accessed 10 May 2022].

Brown, D. R., 2022. *Stroke.* [Online]
Available at: https://www.mayoclinic.org/diseases-conditions/stroke/symptoms-causes/syc-20350113
[Accessed 9 May 2022].

DerSarkissian, C., 2022. *Stroke: What You Need to Know.* [Online]
Available at: https://www.webmd.com/stroke/ss/slideshow-stroke-overview
[Accessed 11 May 2022].

Hill, D. M. D. & Hachinski, V., 1998. Stroke treatment: time is brain. *The Lancet,* 352(S10-S14).

Johnson, W., Onuma, O., Owolabi, M. & Sachdeva, S., 2016. Stroke: a global response is needed. *Bull World Health Organ.*

Kohli, D. P., 2021. *Everything You Need to Know About Stroke.* [Online]
Available at: https://www.healthline.com/health/stroke
[Accessed 10 May 2022].

Kulkarni, D. A., 2021. *Top Causes Of Brain Stroke.* [Online]
Available at: https://www.sparshhospital.com/blog/top-causes-of-brain-stroke/
[Accessed 10 May 2022].

NHS, 2022. *Diagonosis.* [Online]
Available at: https://www.nhs.uk/conditions/stroke/diagnosis/
[Accessed 11 May 2022].

Reuters, T., 2017. *Device allows patients to work remotely with physiotherapist from comfort of home.* [Online]
Available at: https://www.cbc.ca/news/health/sword-physiotherapy-device-1.4022692#:~:text=The%20Stroke%20Wearable%20Operative%20Rehabilitation,sensors%20strapped%20to%20the%20body.
[Accessed 11 May 2022].

Wolfe, C. D. A., 2000. The impact of stroke. *British Medical Bulletin.*

Appendix

```
#Install the required packages
#Read the Packages
library(readxl)
library(psych)
```

```
library(ggplot2)
library(tidyverse)
library(dplyr)
library(caret) #to split the data


#Set Working Directory
setwd('D:/Business Analytics/Advanced Analytics and Machine Learning')
#To remove 10E values
options(scipen = 10000)

brain_stroke <- read.csv('stroke.csv')

# :: Summarize the Data ::

summary(brain_stroke)

#Analyze the Columns with NA
colSums(is.na(brain_stroke))

#summary
summary((brain_stroke))

#Change N/A into Mean value
brain_stroke$bmi <- as.numeric((brain_stroke$bmi))
brain_stroke$bmi[is.na(brain_stroke$bmi)] <- mean(brain_stroke$bmi,na.rm = TRUE)

#Convert the categorical Variables into factor
brain_stroke$gender <- as.factor(brain_stroke$gender)
brain_stroke$hypertension <- as.factor(brain_stroke$hypertension)
brain_stroke$heart_disease <- as.factor(brain_stroke$heart_disease)
brain_stroke$ever_married <- as.factor(brain_stroke$ever_married)
brain_stroke$work_type <- as.factor(brain_stroke$work_type)
brain_stroke$Residence_type <- as.factor(brain_stroke$Residence_type)
brain_stroke$smoking_status <- as.factor(brain_stroke$smoking_status)
brain_stroke$stroke <- as.factor(brain_stroke$stroke)

brain_stroke$gender[brain_stroke$gender == 'Other'] <- 'Male'

#::: MEASURES OF ASSOSCIATION::::

#Chi Square Tests in R

#1.cross tabs Function
```

```
table(brain_stroke$stroke, brain_stroke$gender)
table(brain_stroke$stroke, brain_stroke$hypertension)
table(brain_stroke$stroke, brain_stroke$heart_disease)
table(brain_stroke$stroke, brain_stroke$ever_married)
table(brain_stroke$stroke, brain_stroke$work_type)
table(brain_stroke$stroke, brain_stroke$Residence_type)
table(brain_stroke$stroke, brain_stroke$smoking_status)

#chisq.test() function to perform the test
chisq.test(brain_stroke$stroke, brain_stroke$gender, correct = FALSE)
chisq.test(brain_stroke$stroke, brain_stroke$age, correct = FALSE)
chisq.test(brain_stroke$stroke, brain_stroke$hypertension, correct = FALSE)
chisq.test(brain_stroke$stroke, brain_stroke$heart_disease, correct = FALSE)
chisq.test(brain_stroke$stroke, brain_stroke$ever_married, correct = FALSE)
chisq.test(brain_stroke$stroke, brain_stroke$work_type, correct = FALSE)
chisq.test(brain_stroke$stroke, brain_stroke$Residence_type, correct = FALSE)
chisq.test(brain_stroke$stroke, brain_stroke$avg_glucose_level, correct = FALSE)
chisq.test(brain_stroke$stroke, brain_stroke$bmi, correct = FALSE)
chisq.test(brain_stroke$stroke, brain_stroke$smoking_status, correct = FALSE)

#:::: GGPLOT :::::

#::: Count of Strokes Frequency::::
count_stroke <- as.data.frame(table(brain_stroke$stroke))
count_stroke$Var1 <- ifelse(count_stroke$Var1 == 0, "No", 'Yes')
# Bar Chart of individuals with stroke count
ggplot(count_stroke, aes(x = Var1, y = Freq, fill = Var1)) +
  geom_bar(stat = "identity") + theme(legend.position="none") +
  geom_text(aes(label = Freq), vjust = 0) +
  labs(title="Stroke Count of Individuals",x ="Stroke", y = "Count") +
  theme(plot.title = element_text(hjust = 0.5))

#Marriage Status
count_marry <- as.data.frame(table(brain_stroke$ever_married))
ggplot(count_marry, aes(x = Var1, y = Freq, fill = Var1)) +
  geom_bar(stat = "identity") + theme(legend.position="none") +
  geom_text(aes(label = Freq), vjust = 0) +
  labs(title="Married Status Bar Chart",x ="Married", y = "Count") +
  theme(plot.title = element_text(hjust = 0.5))

# Work Type
count_work <- as.data.frame(table(brain_stroke$work_type))
ggplot(count_work, aes(x = Var1, y = Freq, fill = Var1)) +
  geom_bar(stat = "identity") + theme(legend.position="none") +
```

```
  geom_text(aes(label = Freq), vjust = 0) +
  labs(title="Work Type Bar chart",x ="Work Type", y = "Count") +
  theme(plot.title = element_text(hjust = 0.5))

#Smoking Status
count_smoke <- as.data.frame(table(brain_stroke$smoking_status))
ggplot(count_smoke, aes(x = Var1, y = Freq, fill = Var1)) +
  geom_bar(stat = "identity") + theme(legend.position="none") +
  geom_text(aes(label = Freq), vjust = 0) +
  labs(title="Smoking Status",x ="Smoke?", y = "Count") +
  theme(plot.title = element_text(hjust = 0.5))

#::::: SPLIT THE LIFEX DATA INTO TRAINING AND TEST:::

#to create a partition with 80%
set.seed(123) #generate a sequence of random numbers
index <- createDataPartition(brain_stroke$stroke, p = 0.8, list = FALSE,)
train <- brain_stroke[index, ] #first 80% for training
test <- brain_stroke[-index, ] #bottom 20% for testing

#1. :::: Logistic Regression MODEL 1 ::::
formula1 <- stroke ~ gender + hypertension + heart_disease + ever_married + work_type +
Residence_type + smoking_status + age + avg_glucose_level + bmi
model1 <- glm(formula1, data = train, family = "binomial")
#Summary of Logistic Regression MODEL 1
summary(model1)
#prediction using the model
predictions1 <- predict(model1,test,type ="response")
#Convert probabilities to 1 or 0
class_pred1 <-as.factor(ifelse(predictions1 > 0.5,1,0))
#evaluate the accuracy of the predictions
postResample(class_pred1,test$stroke)

#Confusion Matrix
confusionMatrix(class_pred1, test$stroke)

#2. :::: Decision Tree MODEL 2 ::::
library(DAAG)
library(party)
library(rpart)
library(rpart.plot)
library(mlbench)
library(caret)
library(pROC)
```

```
library(tree)

#Splitting of Data
set.seed(1234)
ind <- sample(2, nrow(brain_stroke), replace = T, prob = c(0.5, 0.5))
train <- brain_stroke[ind == 1,]
test <- brain_stroke[ind == 2,]

#Creation of Tree
tree <- rpart(stroke ~.-id, data = train)
rpart.plot(tree)
printcp(tree)

#Confusion matrix -train
p <- predict(tree, train, type = 'class')
confusionMatrix(p, train$stroke, positive='1')

p1 <- predict(tree, test, type = 'prob')
p1 <- p1[,2]
r <- multiclass.roc(test$stroke, p1, percent = TRUE)
roc <- r[['rocs']]
r1 <- roc[[1]]
plot.roc(r1,
      print.auc=TRUE,
      auc.polygon=TRUE,
      grid=c(0.1, 0.2),
      grid.col=c("green", "red"),
      max.auc.polygon=TRUE,
      auc.polygon.col="lightblue",
      print.thres=TRUE,
      main= 'ROC Curve')

#3.:::::::: Naives Bayes ::::::

library(naivebayes)
library(dplyr)
library(ggplot2)
library(psych)

set.seed(1234)
ind <- sample(2, nrow(brain_stroke), replace = T, prob = c(0.8, 0.2))
train <- brain_stroke[ind == 1,]
test <- brain_stroke[ind == 2,]
```

```
#Naive Bayes Classification
model <- naive_bayes(stroke ~ .-id, data = train, usekernel = T)
plot(model)

p <- predict(model, train, type = 'prob')

#Confusion Matrix – train data
p1 <- predict(model, train)
#Confusion Matrix
(tab1 <- table(p1, train$stroke))
#Accuracy
sum(diag(tab1)) / sum(tab1)

#Confusion Matrix – test data
p2 <- predict(model, test)
#Confusion Matrix
(tab2 <- table(p2, test$stroke))
#Accuracy
(sum(diag(tab2))/sum(tab2))


#4.::::: Random Forest :::::
library(randomForest)
library(datasets)
library(caret)

set.seed(222)
ind <- sample(2, nrow(brain_stroke), replace = TRUE, prob = c(0.7, 0.3))
train <- brain_stroke[ind==1,]
test <- brain_stroke[ind==2,]
rf <- randomForest(stroke~.-id, data=train, proximity=TRUE)
print(rf)

#Prediction & Confusion Matrix – train data
p1 <- predict(rf, train)
confusionMatrix(p1, train$stroke)

#Prediction & Confusion Matrix – test data
p2 <- predict(rf, test)
confusionMatrix(p2, test_no_id$stroke)

#Error Rate of RF
plot(rf)
```

```
#5.::::: Deep Neural Network in R ::::::
library(keras)
library(mlbench)
library(dplyr)
library(magrittr)
library(neuralnet)

data <- brain_stroke%<>% mutate_if(is.factor, as.numeric)

n <- neuralnet(stroke ~ .-id,
               data = data,
               hidden = c(9,5),
               linear.output = F,
               lifesign = 'full',
               rep=1)

plot(n,col.hidden = 'darkgreen',
     col.hidden.synapse = 'darkgreen',
     show.weights = F,
     information = F,
     fill = 'lightblue')


#5. :::::::: XG BOOST::::::::
library(xgboost)
#make this example reproducible
set.seed(0)

# set up the cross-validated hyper-parameter search
XGrid <- expand.grid(
  nrounds = 3500,
  max_depth = 7,
  eta = 0.01,
  gamma = 0.01,
  colsample_bytree = 0.75,
  min_child_weight = 0,
  subsample = 0.5
)

# pack the training control parameters
XControl <- trainControl(
  method = "cv",
  number = 5
)
```

```
# train the model for each parameter combination in the grid,
#   using CV to evaluate
xgb_model <- train(
  stroke ~ .-id,
  train,
  method = "xgbTree",
  tuneLength = 3,
  tuneGrid = XGrid,
  trControl = XControl
)
xgb_model
xgb_pred <- predict(xgb_model, newdata = test)
#Confusion Matrix
confusionMatrix(xgb_pred, test$stroke, positive = '1')
```