# Stroke Prediction

## B. E. Information Technology

By


| | |
|---|---|
| Meet Popat | 37 |
| Yashraj Rai | 38 |
| Deep Vakharia | 39 |
| Aayush Doshi | 40 |

**Group No. - 19**

**Dr. Vandana Patil**

Designation

Department of Information Technology
St. Francis Institute of Technology
(Engineering College)

University of
Mumbai 2021-2022

# DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources.

We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in this submission.

We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

1. _____

(Signature)

- - - - - - - - - - - - - - - - - - - - - - - - - - - -

(Name of student and Roll No.)

2. _____

(Signature)

- - - - - - - - - - - - - - - - - - - - - - - - - - - -

(Name of student and Roll No.)

3. _____

(Signature)

- - - - - - - - - - - - - - - - - - - - - - - - - - - -

(Name of student and Roll No.)

Date:

# CERTIFICATE

This R programming Mini-project Stroke Prediction by is complete in all respects and was successfully demonstrated on 18th May 2022.

Name : _____

Signature :_____

(Internal examiner)

Name : _____

Signature :_____

(External examiner)

Date: 18th May 2022

Place: SFIT, Mumbai

Class- BEITA
Batch- A3

Group No- 1
Project Name –Stroke Prediction

# INDEX


| Sr. No. | Content | Page No. |
|---|---|---|
| 1. | Project Overview | 5-6 |
| 2. | Data Extraction | 7-11 |
| 3. | Exploratory Data Analysis | 12-18 |
| 4. | Application of Mining Algorithm | 19-30 |
| 5. | Data Visualization and Interpretation | 31-35 |
| 6. | Conclusion | 36-37 |
| 7. | Acknowledgement | 38 |


**Signature:**

**Ms. Vandana Patil**
**(Internal Guide)**

# Project Overview

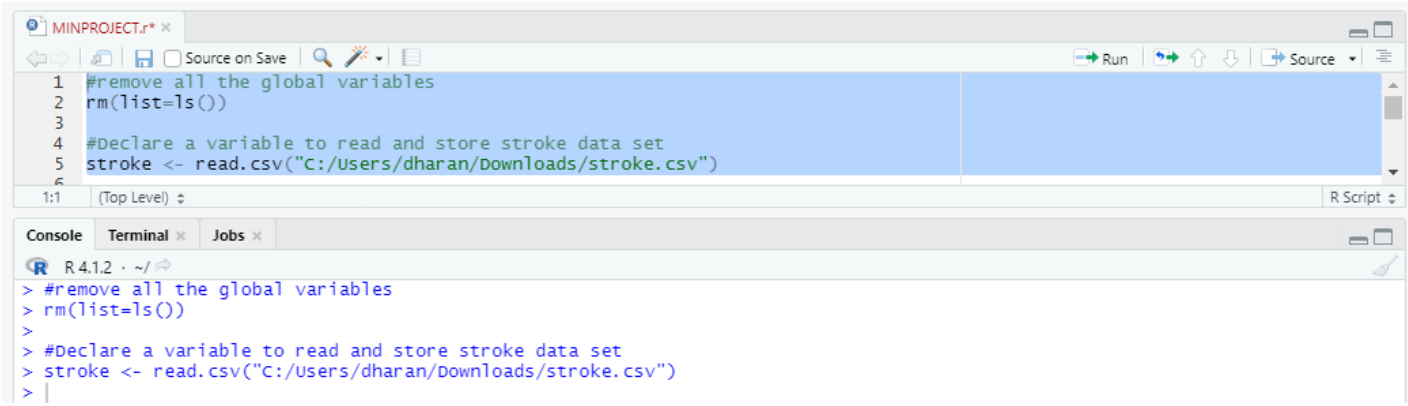
# PROJECT OVERVIEW

| | | |
|---|---|---|
| 1. | Project Title - | **Stroke Prediction** |
| 2. | Data set Name - | **Stroke Prediction Dataset** |
| 3. | Introduction of Data set - | Context<br>- According to the World Health Organization (WHO) stroke is the 2nd leading cause of death globally, responsible for approximately 11% of total deaths.<br>- This dataset is used to predict whether a patient is likely to get a stroke based on the input parameters like gender, age, various diseases, and smoking status. Each row in the data provides relevant information about the patient.<br><br>Attribute Information<br><br>1. id: unique identifier<br>2. gender: "Male", "Female" or "Other"<br>3. age: age of the patient<br>4. hypertension: 0 if the patient doesn't have hypertension, 1 if the patient has hypertension<br>5. heart_disease: 0 if the patient doesn't have any heart diseases, 1 if the patient has a heart disease<br>6. ever_married: "No" or "Yes"<br>7. work_type: "children", "Govt_jov", "Never_worked", "Private" or "Self-employed"<br>8. Residence_type: "Rural" or "Urban"<br>9. avg_glucose_level: average glucose level in blood<br>10. bmi: body mass index<br>11. smoking_status: "formerly smoked", "never smoked", "smokes" or "Unknown"*<br>12. stroke: 1 if the patient had a stroke or 0 if not<br><br>Note: "Unknown" in smoking_status means that the information is unavailable for this patient |
| 4. | Length of Data set | No. of observations (rows) – 5110<br>No of columns - 12 |
| 5. | Name of the source website - | Kaggle- https://www.kaggle.com/ |
| 6. | URL - | https://www.kaggle.com/fedesoriano/stroke-prediction-dataset |

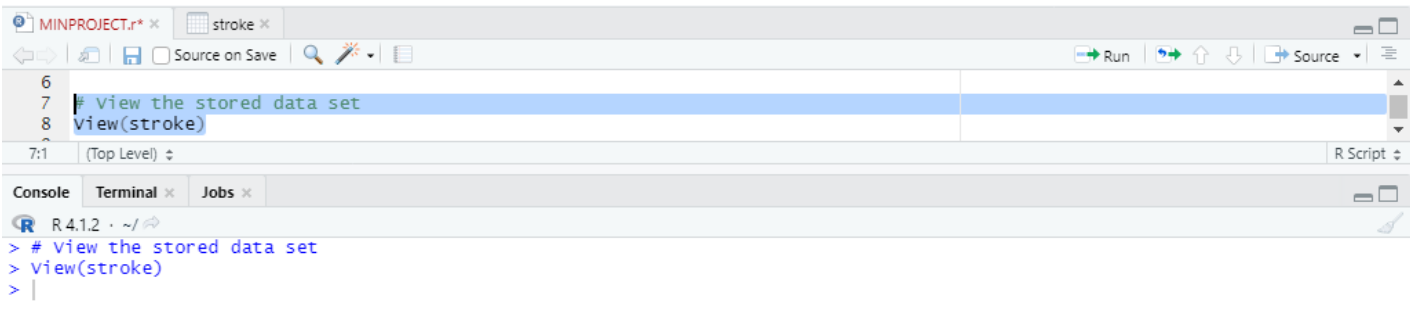
# Data Extraction

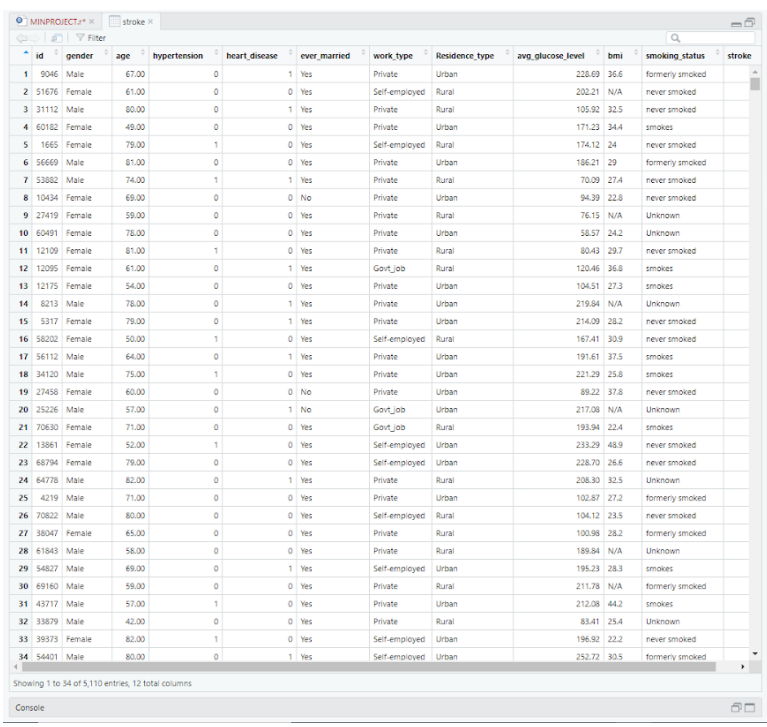
# Data Extraction

## 1. Import Data (.csv)



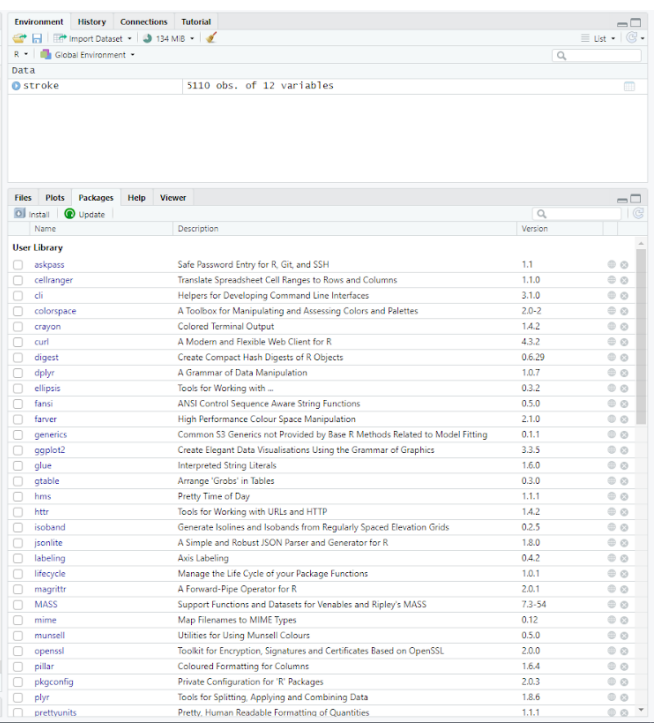

## 2. Viewing the data , (view) (heads) (tail)



| | id | gender | age | hypertension | heart_disease | ever_married | work_type | Residence_type | avg_glucose_level | bmi | smoking_status | stroke |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 9046 | Male | 67.00 | 0 | 1 | Yes | Private | Urban | 228.69 | 36.6 | formerly smoked | |
| 2 | 51676 | Female | 61.00 | 0 | 0 | Yes | Self-employed | Rural | 202.21 | N/A | never smoked | |
| 3 | 31112 | Male | 80.00 | 0 | 1 | Yes | Private | Rural | 105.92 | 32.5 | never smoked | |
| 4 | 60182 | Female | 49.00 | 0 | 0 | Yes | Private | Urban | 171.23 | 34.4 | smokes | |
| 5 | 1665 | Female | 79.00 | 1 | 0 | Yes | Self-employed | Rural | 174.12 | 24 | never smoked | |
| 6 | 56669 | Male | 81.00 | 0 | 0 | Yes | Private | Urban | 186.21 | 29 | formerly smoked | |
| 7 | 53882 | Male | 74.00 | 1 | 1 | Yes | Private | Rural | 70.09 | 27.4 | never smoked | |
| 8 | 10434 | Female | 69.00 | 0 | 0 | No | Private | Urban | 94.39 | 22.8 | never smoked | |
| 9 | 27419 | Female | 59.00 | 0 | 0 | Yes | Private | Rural | 76.15 | N/A | Unknown | |
| 10 | 60491 | Female | 78.00 | 0 | 0 | Yes | Private | Urban | 58.57 | 24.2 | Unknown | |
| 11 | 12109 | Female | 81.00 | 1 | 0 | Yes | Private | Rural | 80.43 | 29.7 | never smoked | |
| 12 | 12095 | Female | 61.00 | 0 | 1 | Yes | Govt_job | Rural | 120.46 | 36.8 | smokes | |
| 13 | 12175 | Female | 54.00 | 0 | 0 | Yes | Private | Urban | 104.51 | 27.3 | smokes | |
| 14 | 8213 | Male | 78.00 | 0 | 1 | Yes | Private | Urban | 219.84 | N/A | Unknown | |
| 15 | 5317 | Female | 79.00 | 0 | 1 | Yes | Private | Urban | 214.09 | 28.2 | never smoked | |
| 16 | 58202 | Female | 50.00 | 1 | 0 | Yes | Self-employed | Rural | 167.41 | 30.9 | never smoked | |
| 17 | 56112 | Male | 64.00 | 0 | 1 | Yes | Private | Urban | 191.61 | 37.5 | smokes | |
| 18 | 34120 | Male | 75.00 | 1 | 0 | Yes | Private | Urban | 221.29 | 25.8 | smokes | |
| 19 | 27458 | Female | 60.00 | 0 | 0 | No | Private | Urban | 89.22 | 37.8 | never smoked | |
| 20 | 25226 | Male | 57.00 | 0 | 1 | No | Govt_job | Urban | 217.08 | N/A | Unknown | |
| 21 | 70630 | Female | 71.00 | 0 | 0 | Yes | Govt_job | Rural | 193.94 | 22.4 | smokes | |
| 22 | 13861 | Female | 52.00 | 1 | 0 | Yes | Self-employed | Urban | 233.29 | 48.9 | never smoked | |
| 23 | 68794 | Female | 79.00 | 0 | 0 | Yes | Self-employed | Urban | 228.70 | 26.6 | never smoked | |
| 24 | 64778 | Male | 82.00 | 0 | 1 | Yes | Private | Rural | 208.30 | 32.5 | Unknown | |
| 25 | 4219 | Male | 71.00 | 0 | 0 | Yes | Private | Urban | 102.87 | 27.2 | formerly smoked | |
| 26 | 70822 | Male | 80.00 | 0 | 0 | Yes | Self-employed | Rural | 104.12 | 23.5 | never smoked | |
| 27 | 38047 | Female | 65.00 | 0 | 0 | Yes | Private | Rural | 100.98 | 28.2 | formerly smoked | |
| 28 | 61843 | Male | 58.00 | 0 | 0 | Yes | Private | Rural | 189.84 | N/A | Unknown | |
| 29 | 54827 | Male | 69.00 | 0 | 1 | Yes | Self-employed | Urban | 195.23 | 28.3 | smokes | |
| 30 | 69160 | Male | 59.00 | 0 | 0 | Yes | Private | Rural | 211.78 | N/A | formerly smoked | |
| 31 | 43717 | Male | 57.00 | 1 | 0 | Yes | Private | Urban | 212.08 | 44.2 | smokes | |
| 32 | 33879 | Male | 42.00 | 0 | 0 | Yes | Private | Rural | 83.41 | 25.4 | Unknown | |
| 33 | 39373 | Female | 82.00 | 1 | 0 | Yes | Self-employed | Urban | 196.92 | 22.2 | never smoked | |
| 34 | 54401 | Male | 80.00 | 0 | 1 | Yes | Self-employed | Urban | 252.72 | 30.5 | formerly smoked | |

Showing 1 to 34 of 5,110 entries, 12 total columns



```r
#head and tail functions
head(stroke)
tail(stroke)
```

```
> #head and tail functions
> head(stroke)
     id gender age hypertension heart_disease ever_married     work_type Residence_type avg_glucose_level  bmi
1  9046   Male  67            0             1          Yes       Private          Urban            228.69 36.6
2 51676 Female  61            0             0          Yes Self-employed          Rural            202.21  N/A
3 31112   Male  80            0             1          Yes       Private          Rural            105.92 32.5
4 60182 Female  49            0             0          Yes       Private          Urban            171.23 34.4
5  1665 Female  79            1             0          Yes Self-employed          Rural            174.12   24
6 56669   Male  81            0             0          Yes       Private          Urban            186.21   29
   smoking_status stroke
1 formerly smoked      1
2    never smoked      1
3    never smoked      1
4          smokes      1
5    never smoked      1
6 formerly smoked      1
> tail(stroke)
        id gender age hypertension heart_disease ever_married     work_type Residence_type avg_glucose_level  bmi
5105 14180 Female  13            0             0           No      children          Rural            103.08 18.6
5106 18234 Female  80            1             0          Yes       Private          Urban             83.75  N/A
5107 44873 Female  81            0             0          Yes Self-employed          Urban            125.20   40
5108 19723 Female  35            0             0          Yes Self-employed          Rural             82.99 30.6
5109 37544   Male  51            0             0          Yes       Private          Rural            166.29 25.6
5110 44679 Female  44            0             0          Yes      Govt_job          Urban             85.28 26.2
      smoking_status stroke
5105         Unknown      0
5106    never smoked      0
5107    never smoked      0
5108    never smoked      0
5109 formerly smoked      0
5110         Unknown      0
>
```

## 3. Using dimension function to get to know more about the data

```r
#to check dimension of the data set
dim(stroke)
```

```
> #to check dimension of the data set
> dim(stroke)
[1] 5110   12
>
```

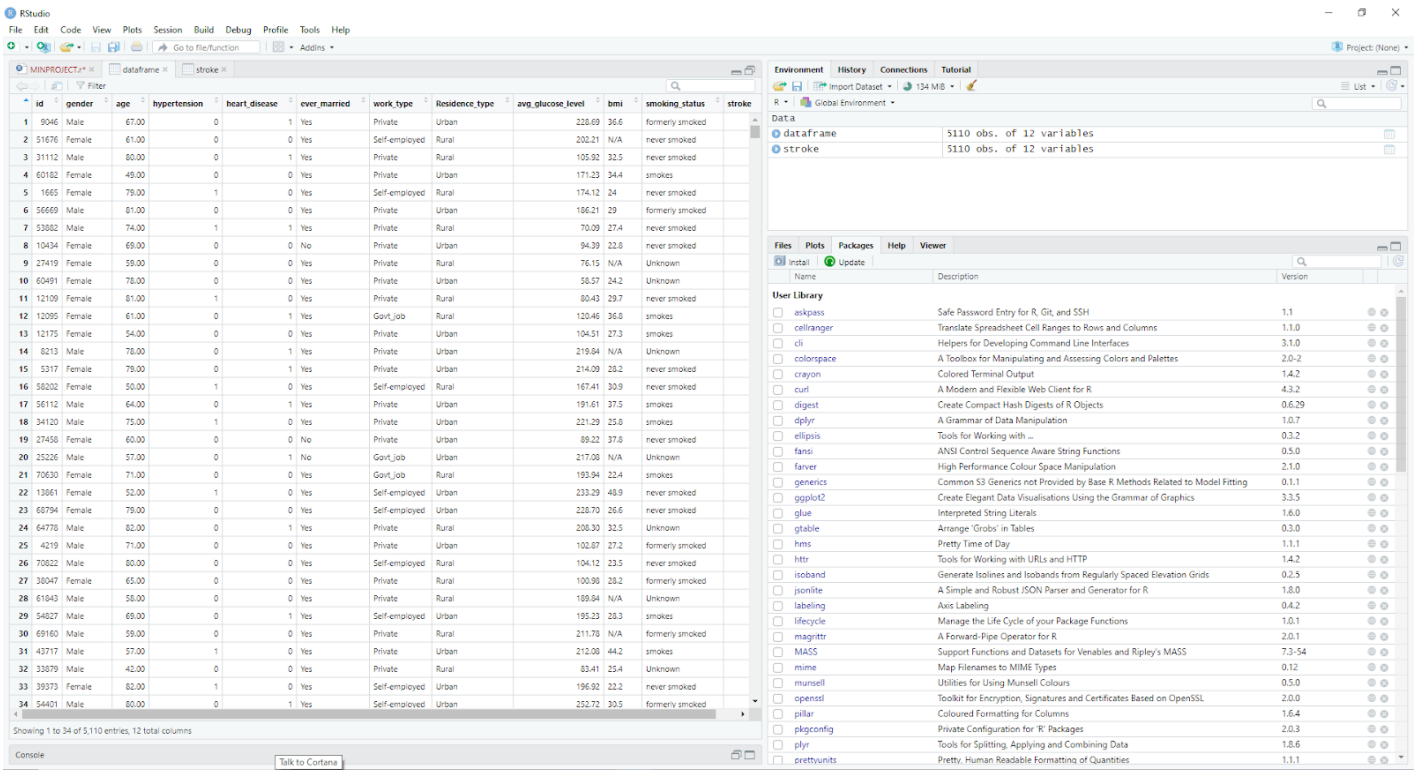
**Total number of rows : 5110**

**Total number of columns : 12**

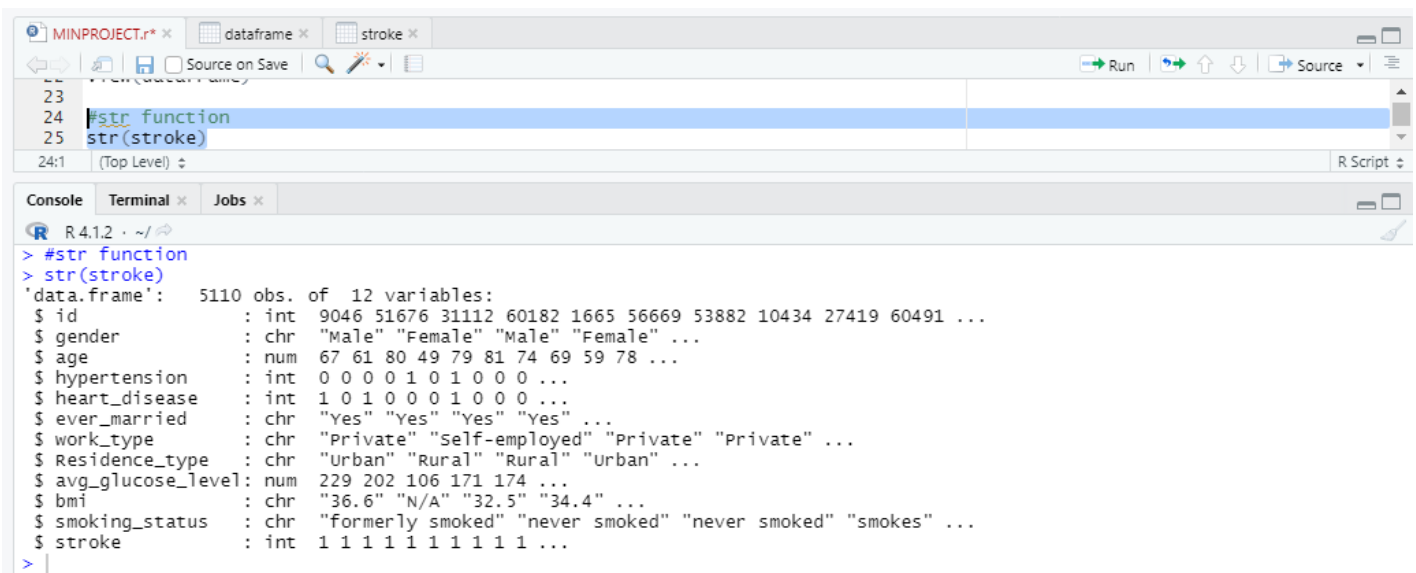
## 4. create data frame including only required data columns

```r
#creating the  data frame including only required data columns
dataframe <- data.frame(stroke)
View(dataframe)
```

```
> #creating the  data frame including only required data columns
> dataframe <- data.frame(stroke)
> View(dataframe)
>
```




## 5. Use of Str function




**As you can see here in bmi column all the values are characters but in real time it should be a numeric value so we;re converting the data type of bmi from char to float**

MINPROJECT.r* | dataframe | stroke

Source on Save | Run | Source

```r
#str function
str(stroke)

stroke$bmi <- as.numeric(as.character(stroke$bmi))
```

25:1 (Top Level) R Script

Console | Terminal | Jobs

R 4.1.2 · ~/

```
> stroke$bmi <- as.numeric(as.character(stroke$bmi))
Warning message:
NAs introduced by coercion
> #str function
> str(stroke)
'data.frame':	5110 obs. of  12 variables:
 $ id               : int  9046 51676 31112 60182 1665 56669 53882 10434 27419 60491 ...
 $ gender           : chr  "Male" "Female" "Male" "Female" ...
 $ age              : num  67 61 80 49 79 81 74 69 59 78 ...
 $ hypertension     : int  0 0 0 0 1 0 1 0 0 0 ...
 $ heart_disease    : int  1 0 1 0 0 0 1 0 0 0 ...
 $ ever_married     : chr  "Yes" "Yes" "Yes" "Yes" ...
 $ work_type        : chr  "Private" "Self-employed" "Private" "Private" ...
 $ Residence_type   : chr  "Urban" "Rural" "Rural" "Urban" ...
 $ avg_glucose_level: num  229 202 106 171 174 ...
 $ bmi              : num  36.6 NA 32.5 34.4 24 29 27.4 22.8 NA 24.2 ...
 $ smoking_status   : chr  "formerly smoked" "never smoked" "never smoked" "smokes" ...
 $ stroke           : int  1 1 1 1 1 1 1 1 1 1 ...
>
```

# Exploratory Data Analysis

# Exploratory Data Analysis

1. **As categorical variables are summarized using a frequency or proportion.So we First create a table for categorical variable hypertension,then calculate the frequency**

```
#create categorical table for variable hypertension
table(stroke$hypertension)
```

```
> #create categorical table for variable hypertension
> table(stroke$hypertension)

   0    1
4612  498
```

2. **Cross verifying  total no of observations in the column of hypertension**

```
#Cross verifying  total no of observations in the column of hypertension
length(stroke$hypertension)
```

```
> #Cross verifying  total no of observations in the column of hypertension
> length(stroke$hypertension)
[1] 5110
```

3. **To express table using proportion(dividing it by no of rows to get the proportion value)**

```
#calculate proportion
table(stroke$hypertension)/5110
```

```
> #calculate proportion
> table(stroke$hypertension)/5110

         0          1
0.90254403 0.09745597
```

4. **To calculate mean of a numeric variable age we will use mean() function**

```
#Using mean() function to calculate avg age of the patients
mean(stroke$age)
```

```
> #Using mean() function to calculate avg age of the patients
> mean(stroke$age)
[1] 43.22661
```

5. **Calculating median of a numeric variable age we use median() function**

```
#Using median() function to calculate median age of the patients
median(stroke$age)
```

```
> #Using median() function to calculate median age of the patients
> median(stroke$age)
[1] 45
```

6. **To calculate variance of a variable avg glucose level we use var() method**

```
#Using var() function to calculate variance of the avg glucose level of the patients
var(stroke$avg_glucose_level)
```

```
> #Using var() function to calculate variance of the avg glucose level of the patients
> var(stroke$avg_glucose_level)
[1] 2050.601
```

7. **To calculate standard deviation of a variable bmi we use sd() function**

```
#Using standard deviation of bmi we use sd() function
sd(stroke$bmi)
```

```
> #Using standard deviation of bmi we use sd() function
> sd(stroke$bmi)
[1] 7.761774
```

8. **To calculate minimum observation for a numeric variable avg glucose level we use min()**

```
#Using min function to find min value of avg glucose level
min(stroke$avg_glucose_level)
```

```
> #Using min function to find min value of avg glucose level
> min(stroke$avg_glucose_level)
[1] 55.12
```

9. **To find index of the minimum observation for a numeric variable avg glucose level**

```
#To find index of the minimum observation for a numeric variable avg glucose level
which.min(stroke$avg_glucose_level)
```

```
> #To find index of the minimum observation for a numeric variable avg glucose level
> which.min(stroke$avg_glucose_level)
[1] 4887
```

10. **To calculate maximum observation for a numeric variable avg_glucose_level we use max()**

```
#To calculate maximum observation for a numeric variable avg_glucose_level we use max()
max(stroke$avg_glucose_level)
```

```
> #To calculate maximum observation for a numeric variable avg_glucose_level we use max()
> max(stroke$avg_glucose_level)
[1] 271.74
```

11. **To find index of the maximum observation for a numeric variable avg_glucose_level**

```
#To find index of the maximum observation for a numeric variable avg_glucose_level
which.max(stroke$avg_glucose_level)
```

```
> #To find index of the maximum observation for a numeric variable avg_glucose_level
> which.max(stroke$avg_glucose_level)
[1] 194
```

12. **To calculate sum of all observation for a numeric variable bmi**

```
#To calculate sum of all observation for a numeric variable bmi
sum(stroke$bmi)
```

```
> #To calculate sum of all observation for a numeric variable bmi
> sum(stroke$bmi)
[1] 148670.9
```

## 13. Calculate mean for a numeric variable age with the help of sum()

```
#Calculate mean for a numeric variable age with the help of sum()
sum(stroke$age)/5110
```

```
> #Calculate mean for a numeric variable age with the help of sum()
> sum(stroke$age)/5110
[1] 43.22661
```

## 14. To calculate range it will display min and max value for a numeric variable bmi

```
#To calculate range it will display min and max value for a numeric variable bmi
range(stroke$bmi)
```

```
> #To calculate range it will display min and max value for a numeric variable bmi
> range(stroke$bmi)
[1] 10.3 97.6
```

## 15. To calculate percentile value with the help of quantile() for a numeric variable age

```
#To calculate percentile value with the help of quantile() for a numeric variable age
quantile(stroke$age)
```

```
> #To calculate percentile value with the help of quantile() for a numeric variable age
> quantile(stroke$age)
   0%   25%   50%   75%  100%
 0.08 25.00 45.00 61.00 82.00
```

## 16. To calculate specific percentile value for a numeric variable age

```
#To calculate specific percentile value for a numeric variable age
quantile(stroke$age,probs = 0.7)
```

```
> #To calculate specific percentile value for a numeric variable age
> quantile(stroke$age,probs = 0.7)
70%
 57
```

## 17. To calculate random multiple percentiles for a numeric variable age

```r
#To calculate random multiple percentiles for a numeric variable age
quantile(stroke$age,probs = c(0.2,0.4,0.6,0.8))
```

```
> #To calculate random multiple percentiles for a numeric variable age
> quantile(stroke$age,probs = c(0.2,0.4,0.6,0.8))
20% 40% 60% 80%
 20  38  51  65
```

## 18. To calculate correlation between numeric variables bmi and hypertension

```r
#To calculate correlation between numeric variables bmi and hypertension
cor(stroke$bmi,stroke$hypertension)
```

```
> #To calculate correlation between numeric variables bmi and hypertension
> cor(stroke$bmi,stroke$hypertension)
[1] 0.1707744
```

## 19. To calculate covariance for numeric variables bmi and hypertension

```r
#To calculate covariance between numeric variables bmi and hypertension
cov(stroke$bmi,stroke$hypertension)
```

```
> #To calculate covariance between numeric variables bmi and hypertension
> cov(stroke$bmi,stroke$hypertension)
[1] 0.3931559
```

## 20. To calculate variance for numeric variables bmi and hypertension

```r
#To calculate variance between numeric variables bmi and hypertension
var(stroke$bmi,stroke$hypertension)
```

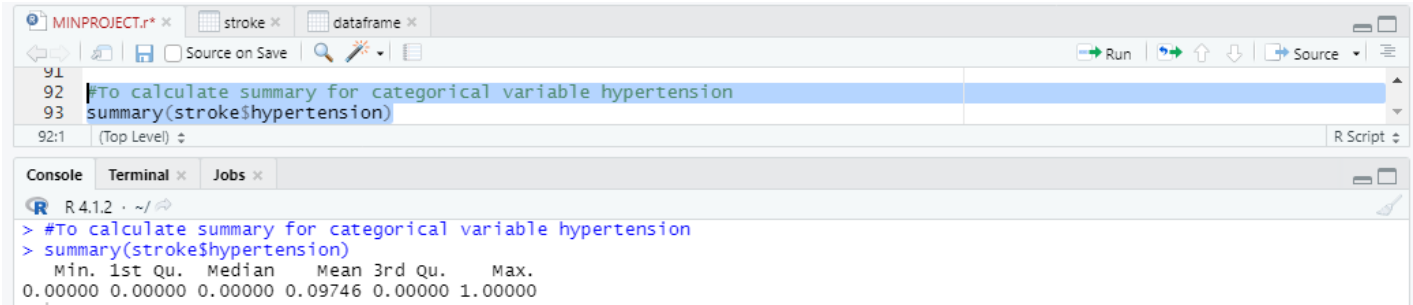
```
> #To calculate variance between numeric variables bmi and hypertension
> var(stroke$bmi,stroke$hypertension)
[1] 0.3931559
```

## 21. To calculate summary for numeric variable age

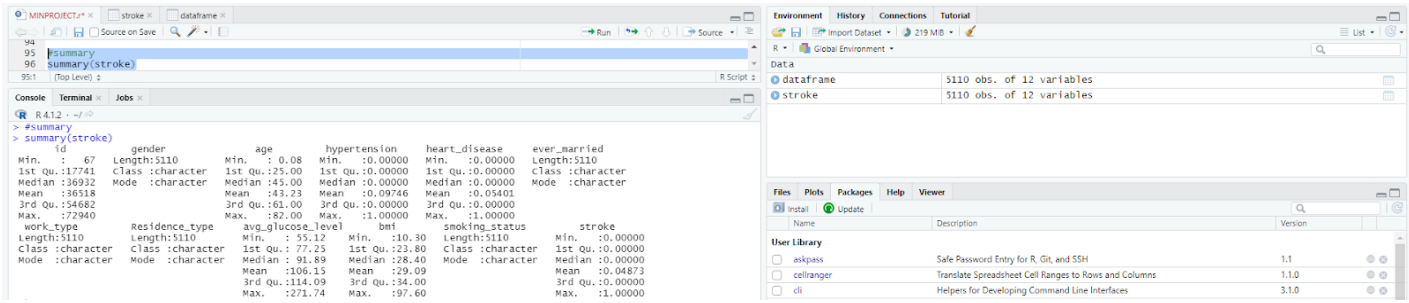
```r
#To calculate summary for numeric variable age
summary(stroke$age)
```

```
> #To calculate summary for numeric variable age
> summary(stroke$age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.08   25.00   45.00   43.23   61.00   82.00
```

## 22.To calculate summary for categorical variable hypertension




## 23.Summary of entire table



# Application of Mining Algorithm

# Application of Mining/Analytics Algorithm

## 1. Naive Bayes Algorithm

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Checking if there is any Null Values present in the data frame

```
#To check if there are any null values present in the dataframe
anyNA(dataframe)
```

```
> #To check if there are any null values present in the dataframe
> anyNA(dataframe)
[1] FALSE
```

There are no null values present in the dataframe

**Installing Required packages**

```
# Installing Packages
install.packages("e1071")
install.packages("caTools")
install.packages("caret")
```

```
> # Installing Packages
> install.packages("e1071")
Installing package into 'C:/Users/dharan/OneDrive/Documents/R/win-library/4.1'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/e1071_1.7-9.zip'
Content type 'application/zip' length 1023068 bytes (999 KB)
downloaded 999 KB

package 'e1071' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\dharan\AppData\Local\Temp\RtmpwJZFWB\downloaded_packages
> install.packages("caTools")
Installing package into 'C:/Users/dharan/OneDrive/Documents/R/win-library/4.1'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/caTools_1.18.2.zip'
Content type 'application/zip' length 316333 bytes (308 KB)
downloaded 308 KB

package 'caTools' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\dharan\AppData\Local\Temp\RtmpwJZFWB\downloaded_packages
> install.packages("caret")
Installing package into 'C:/Users/dharan/OneDrive/Documents/R/win-library/4.1'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/caret_6.0-91.zip'
Content type 'application/zip' length 3589127 bytes (3.4 MB)
downloaded 3.4 MB

package 'caret' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\dharan\AppData\Local\Temp\RtmpwJZFWB\downloaded_packages
>
```

## Loading Required packages

```
# Loading package
library(e1071)
library(caTools)
library(caret)
```

```
> # Loading package
> library(e1071)
Warning message:
package 'e1071' was built under R version 4.1.3
> library(caTools)
Warning message:
package 'caTools' was built under R version 4.1.3
> library(caret)
Loading required package: ggplot2
Loading required package: lattice
Warning messages:
1: package 'caret' was built under R version 4.1.3
2: package 'ggplot2' was built under R version 4.1.3
```

## Splitting data into training and test data

```
# Splitting data into train
# and test data
split <- sample.split(stroke, SplitRatio = 0.7)
train_cl <- subset(stroke, split == "TRUE")
test_cl <- subset(stroke, split == "FALSE")
```

```
> # Splitting data into train
> # and test data
> split <- sample.split(stroke, SplitRatio = 0.7)
> train_cl <- subset(stroke, split == "TRUE")
> test_cl <- subset(stroke, split == "FALSE")
```

## Feature Scaling

```
# Feature Scaling
train_scale <- scale(train_cl[, 3:5])
test_scale <- scale(test_cl[, 3:5])
```

```
> # Feature Scaling
> train_scale <- scale(train_cl[, 3:5])
> test_scale <- scale(test_cl[, 3:5])
```

| | |
|---|---|
| test_cl | 1704 obs. of 12 variables |
| test_scale | num [1:1704, 1:3] 0.247 1.328 0.679 1.501 0.29 ... |
| train_cl | 3406 obs. of 12 variables |
| train_scale | num [1:3406, 1:3] 1.065 0.797 1.647 1.602 1.691 ... |

## Fitting Naive Bayes Model to training dataset

```
> # Fitting Naive Bayes Model to training dataset
> set.seed(120)  # Setting Seed
> classifier_cl <- naiveBayes(gender ~ ., data = train_cl)
```

```
R ▾ | Global Environment ▾
Data
classifier_cl                List of  5
    $ apriori  : 'table' int [1:2(1d)] 1994 1412
     ..- attr(*, "dimnames")=List of 1
     .. ..$ Y: chr [1:2] "Female" "Male"
    $ tables   :List of 11
     ..$ id               : num [1:2, 1:2] 35989 35965 21222 21251
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y : chr [1:2] "Female" "Male"
     .. .. ..$ id: NULL
     ..$ age              : num [1:2, 1:2] 43.9 42.3 21.8 23.1
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y  : chr [1:2] "Female" "Male"
     .. .. ..$ age: NULL
     ..$ hypertension     : num [1:2, 1:2] 0.0978 0.0977 0.2971 0.2971
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y            : chr [1:2] "Female" "Male"
     .. .. ..$ hypertension: NULL
     ..$ heart_disease    : num [1:2, 1:2] 0.0361 0.0786 0.1866 0.2692
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y             : chr [1:2] "Female" "Male"
     .. .. ..$ heart_disease: NULL
     ..$ ever_married     : 'table' num [1:2, 1:2] 0.323 0.359 0.677 0.641
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y            : chr [1:2] "Female" "Male"
     .. .. ..$ ever_married: chr [1:2] "No" "Yes"
     ..$ work_type        : 'table' num [1:2, 1:5] 0.10331 0.16643 0.12989 0.11756 0.00201 ...
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y         : chr [1:2] "Female" "Male"
     .. .. ..$ work_type: chr [1:5] "children" "Govt_job" "Never_worked" "Private" ...
     ..$ Residence_type   : 'table' num [1:2, 1:2] 0.491 0.496 0.509 0.504
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y              : chr [1:2] "Female" "Male"
     .. .. ..$ Residence_type: chr [1:2] "Rural" "Urban"
     ..$ avg_glucose_level: num [1:2, 1:2] 104.7 108.9 44.3 47.3
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y                 : chr [1:2] "Female" "Male"
     .. .. ..$ avg_glucose_level: NULL
     ..$ bmi              : num [1:2, 1:2] 29.25 29.17 8.05 7.44
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y  : chr [1:2] "Female" "Male"
     .. .. ..$ bmi: NULL
     ..$ smoking_status   : 'table' num [1:2, 1:4] 0.162 0.189 0.41 0.307 0.153 ...
     .. ..- attr(*, "dimnames")=List of 2
     .. .. ..$ Y              : chr [1:2] "Female" "Male"
     .. .. ..$ smoking_status: chr [1:4] "formerly smoked" "never smoked" "smokes" "Unknown"
     ..$ stroke           : num [1:2, 1:2] 0.0476 0.0503 0.2131 0.2186
Files | Plots | Packages | Help | Viewer
```

```
> # Fitting Naive Bayes Model to training dataset
> set.seed(120)  # Setting Seed
> classifier_cl <- naiveBayes(gender ~ ., data = train_cl)
> classifier_cl

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
   Female      Male
0.5854375 0.4145625

Conditional probabilities:
        id
Y            [,1]     [,2]
  Female 35988.53 21221.66
  Male   35965.43 21251.25

        age
Y            [,1]     [,2]
  Female 43.86012 21.76021
  Male   42.26210 23.13417

        hypertension
Y              [,1]      [,2]
  Female 0.09779338 0.2971096
  Male   0.09773371 0.2970595

        heart_disease
Y              [,1]      [,2]
  Female 0.03610832 0.1866065
  Male   0.07861190 0.2692274

        ever_married
Y               No       Yes
  Female 0.3229689 0.6770311
  Male   0.3590652 0.6409348
```

```
        work_type
Y           children    Govt_job Never_worked     Private Self-employed
  Female 0.103309930 0.129889669  0.002006018 0.600300903   0.164493480
  Male   0.166430595 0.117563739  0.005665722 0.566572238   0.143767705

        Residence_type
Y            Rural     Urban
  Female 0.4914744 0.5085256
  Male   0.4957507 0.5042493

        avg_glucose_level
Y            [,1]     [,2]
  Female 104.7010 44.29455
  Male   108.9118 47.25815

        bmi
Y            [,1]     [,2]
  Female 29.25065 8.047978
  Male   29.16671 7.440742

        smoking_status
Y        formerly smoked never smoked    smokes   Unknown
  Female       0.1624875    0.4102307 0.1529589 0.2743230
  Male         0.1890935    0.3066572 0.1635977 0.3406516

        stroke
Y              [,1]      [,2]
  Female 0.04764293 0.2130630
  Male   0.05028329 0.2186063
```

## Using the classifier model for predicting test data

```
R 4.1.2 · ~/
> # Predicting on test data'
> y_pred <- predict(classifier_cl, newdata = test_cl)
> y_pred
  [1] Female Male   Female Female Female Female Male   Female Male   Female Female Male   Female Male   Male   Male
 [17] Female Female Female Male   Female Female Female Female Male   Female Female Female Male   Male   Female Female
 [33] Female Male   Female Female Male   Female Female Female Male   Female Male   Female Male   Male   Male   Female
 [49] Female Male   Female Female Female Male   Female Female Male   Female Female Female Female Male   Female Male
 [65] Male   Female Male   Male   Female Female Female Female Male   Female Female Male   Female Male   Female Female
 [81] Male   Female Female Male   Female Female Male   Female Female Female Female Female Female Male   Female Female
 [97] Male   Female Female Female Female Female Female Female Male   Female Female Female Male   Female Female Female
[113] Female Male   Female Female Female Female Female Female Male   Female Female Male   Female Male   Male   Female
[129] Female Female Female Female Female Female Female Male   Female Female Male   Female Female Female Female Female
[145] Female Female Female Female Female Male   Female Female Male   Female Female Female Female Female Female Female
[161] Male   Female Female Female Female Male   Female Female Female Female Female Female Male   Female Female Male
[177] Female Female Female Female Female Male   Female Female Female Male   Female Male   Female Female Female Female
[193] Female Female Male   Male   Female Female Female Male   Male   Female Female Male   Female Female Female Female
[209] Female Female Female Female Female Female Female Female Female Female Male   Female Female Female Female Female
[225] Female Female Female Female Female Female Female Female Female Female Female Female Female Female Female Male
[241] Female Female Female Female Female Female Female Female Female Male   Female Male   Female Female Female Female
[257] Female Female Female Female Male   Female Female Female Female Female Female Female Female Female Female Female
[273] Female Female Female Female Female Female Female Female Female Female Female Female Female Female Female Female
[289] Male   Female Female Female Female Female Female Female Female Female Female Female Female Female Female Female
[305] Female Female Female Male   Female Female Female Female Male   Male   Female Female Male   Female Female Male
[321] Female Female Female Female Female Female Male   Female Female Male   Male   Female Male   Male   Female Female
[337] Female Female Female Female Female Female Female Female Female Female Female Female Male   Female Female Female
[353] Female Male   Female Female Female Male   Female Female Female Female Female Female Female Female Female Male
[369] Female Female Female Female Female Female Male   Female Female Male   Female Male   Male   Female Female Female
[385] Male   Female Female Female Female Female Female Female Female Male   Female Male   Female Male   Female Female
[401] Female Male   Female Female Female Male   Female Female Female Female Female Female Female Female Male   Female
[417] Male   Female Female Female Female Female Female Female Female Male   Female Male   Male   Female Male   Female
[433] Female Female Female Female Female Female Female Male   Female Female Female Female Female Female Female Female
[449] Female Female Female Female Male   Female Female Male   Male   Female Female Female Female Male   Female Male
[465] Female Female Female Female Female Female Female Male   Female Female Male   Female Female Female Female Male   Female
[481] Female Female Female Female Male   Female Female Female Male   Female Female Female Female Female Male   Female
[497] Female Female Female Female Male   Male   Female Female Female Female Female Female Male   Female Female Female
[513] Female Female Female Female Female Female Female Female Female Male   Female Female Female Female Female Female
[529] Male   Male   Female Female Female Female Female Female Female Male   Male   Female Female Female Female Male   Female
[545] Female Female Male   Female Male   Female Female Female Female Female Female Female Female Female Female Female Female
[561] Female Male   Female Female Female Female Female Female Female Female Female Female Female Female Female Female Female
[577] Female Female Female Male   Female Female Female Female Female Male   Female Female Female Female Female Female Female
[593] Female Female Female Male   Male   Male   Female Female Female Female Male   Female Female Female Female Female Female
[609] Female Female Female Male   Female Female Female Female Female Female Female Female Male   Male   Female Female
[625] Male   Female Female Female Female Female Female Male   Female Female Female Female Female Female Female Male
[641] Female Female Male   Male   Female Female Female Female Female Female Female Female Male   Male   Female Female Male
[657] Female Female Female Female Female Female Female Female Female Female Female Female Female Female Male   Female
[673] Female Female Female Female Female Female Male   Female Female Female Female Female Female Female Female Female
[689] Female Female Female Female Male   Male   Female Female Male   Female Female Female Female Female Female Female
[705] Female Female Female Female Female Female Male   Female Female Female Female Female Female Male   Female Male
[721] Male   Female Female Male   Male   Female Female Female Female Female Female Female Female Male   Male   Female
[737] Female Female Female Male   Male   Female Female Female Female Female Male   Male   Male   Male   Female Female
```

```
Values
  split     logi [1:12] TRUE TRUE TRUE FALSE TRUE TRUE ...
  y_pred    Factor w/ 2 levels "Female","Male": 1 2 1 1 1 1 2 1 2 1 ...
```

## Confusion matrix

```
MINPROJECT.r* ×   stroke ×   dataframe ×
Source on Save                                  Run   Source
125
126  # Confusion Matrix
127  cm <- table(test_cl$gender, y_pred)
128  cm
128:1   (Top Level)                                   R Script
```

```
R 4.1.2 · ~/
> # Confusion Matrix
> cm <- table(test_cl$gender, y_pred)
> cm
        y_pred
         Female Male
  Female    845  155
  Male      526  177
  Other       1    0
```

```
Values
  cm        'table' int [1:3, 1:2] 845 526 1 155 177 0
```

2. **Support Vector Machine(SVM)**
   A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane that categorizes new examples.The most important question that arises while using SVM is how to decide the right hyperplan

Checking if there is any Null Values present in the data frame

```
#To check if there are any null values present in the dataframe
anyNA(dataframe)
```

```
> #To check if there are any null values present in the dataframe
> anyNA(dataframe)
[1] FALSE
```

There are no null values present in the dataframe

**Installing Required packages**

```
> install.packages('caTools')
Error in install.packages : Updating loaded packages

Restarting R session...

> install.packages("caTools")
Installing package into 'C:/Users/dharan/OneDrive/Documents/R/win-library/4.1'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/caTools_1.18.2.zip'
Content type 'application/zip' length 316333 bytes (308 KB)
downloaded 308 KB

package 'caTools' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\dharan\AppData\Local\Temp\Rtmp4i7ZuK\downloaded_packages
```

**Importing Required packages**

```
> library(caTools)
Warning message:
package 'caTools' was built under R version 4.1.3
>
```

**Taking columns 3-5 for consideration**

```
# Taking columns 3-5
dataset = stroke[3:5]
View(dataset)
```

```
> # Taking columns 3-5
> dataset = stroke[3:5]
> View(dataset)
```

MINPROJECT.r* × dataset × stroke ×

Filter

| | age | hypertension | heart_disease |
|---|---|---|---|
| 1 | 67.00 | 0 | 1 |
| 2 | 61.00 | 0 | 0 |
| 3 | 80.00 | 0 | 1 |
| 4 | 49.00 | 0 | 0 |
| 5 | 79.00 | 1 | 0 |
| 6 | 81.00 | 0 | 0 |
| 7 | 74.00 | 1 | 1 |
| 8 | 69.00 | 0 | 0 |
| 9 | 59.00 | 0 | 0 |
| 10 | 78.00 | 0 | 0 |
| 11 | 81.00 | 1 | 0 |
| 12 | 61.00 | 0 | 1 |
| 13 | 54.00 | 0 | 0 |
| 14 | 78.00 | 0 | 1 |
| 15 | 79.00 | 0 | 1 |
| 16 | 50.00 | 1 | 0 |
| 17 | 64.00 | 0 | 1 |
| 18 | 75.00 | 1 | 0 |
| 19 | 60.00 | 0 | 0 |
| 20 | 57.00 | 0 | 1 |
| 21 | 71.00 | 0 | 0 |
| 22 | 52.00 | 1 | 0 |
| 23 | 79.00 | 0 | 0 |
| 24 | 82.00 | 0 | 1 |

## Encoding the target feature as factor

MINPROJECT.r* × dataset × stroke × dataframe ×

Source on Save Run Source

```
# Encoding the target feature as factor
dataset$heart_disease = factor(dataset$heart_disease, levels = c(0, 1))
View(dataset)
```

137:1 (Top Level) R Script

R 4.1.2 · ~/

```
> # Encoding the target feature as factor
> dataset$heart_disease = factor(dataset$heart_disease, levels = c(0, 1))
> View(dataset)
```

dataset 5110 obs. of 3 variables

## Splitting the dataset into the Training set and Test set

```
# Splitting the dataset into the Training set and Test set
set.seed(123)
split1 = sample.split(stroke$age, SplitRatio = 0.7)
training_set = subset(stroke, split1 == TRUE)
test_set = subset(stroke, split1 == FALSE)
```

```
> # Splitting the dataset into the Training set and Test set
> set.seed(123)
> split1 = sample.split(stroke$age, SplitRatio = 0.7)
> training_set = subset(stroke, split1 == TRUE)
> test_set = subset(stroke, split1 == FALSE)
```

```
split1                  logi [1:5110] TRUE FALSE FALSE TRUE FALSE FALSE ...
test_set                1538 obs. of 12 variables
  $ id               : int  51676 31112 1665 56669 5317 68794 38047 61843 54827 39373 ...
  $ gender           : chr  "Female" "Male" "Female" "Male" ...
  $ age              : num  61 80 79 81 79 79 65 58 69 82 ...
  $ hypertension     : int  0 0 1 0 0 0 0 0 0 1 ...
  $ heart_disease    : int  0 1 0 0 1 0 0 0 1 0 ...
  $ ever_married     : chr  "Yes" "Yes" "Yes" "Yes" ...
  $ work_type        : chr  "Self-employed" "Private" "Self-employed" "Private" ...
  $ Residence_type   : chr  "Rural" "Rural" "Rural" "Urban" ...
  $ avg_glucose_level: num  202 106 174 186 214 ...
  $ bmi              : num  34 32.5 24 29 28.2 26.6 28.2 34 28.3 22.2 ...
  $ smoking_status   : chr  "never smoked" "never smoked" "never smoked" "formerly smoked" ...
  $ stroke           : int  1 1 1 1 1 1 1 1 1 1 ...
training_set            3572 obs. of 12 variables
  $ id               : int  9046 60182 53882 10434 27419 60491 12109 12095 12175 8213 ...
  $ gender           : chr  "Male" "Female" "Male" "Female" ...
  $ age              : num  67 49 74 69 59 78 81 61 54 78 ...
  $ hypertension     : int  0 0 1 0 0 0 1 0 0 0 ...
  $ heart_disease    : int  1 0 1 0 0 0 0 1 0 1 ...
  $ ever_married     : chr  "Yes" "Yes" "Yes" "No" ...
  $ work_type        : chr  "Private" "Private" "Private" "Private" ...
  $ Residence_type   : chr  "Urban" "Urban" "Rural" "Urban" ...
  $ avg_glucose_level: num  228.7 171.2 70.1 94.4 76.2 ...
  $ bmi              : num  36.6 34.4 27.4 22.8 34 24.2 29.7 36.8 27.3 34 ...
  $ smoking_status   : chr  "formerly smoked" "smokes" "never smoked" "never smoked" ...
  $ stroke           : int  1 1 1 1 1 1 1 1 1 1 ...
```

## Feature Scaling

```
> # Feature Scaling
> training_set[3:5] = scale(training_set[3:5])
> test_set[3:5] = scale(test_set[3:5])
```

## Training_set

MINPROJECT.r* × training_set × test_set × dataset × stroke × dataframe ×

Filter

| | id | gender | age | hypertension | heart_disease | ever_married | work_type | Residence_type | avg_glucose_level | bmi | smoking_status | stro |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 51676 | Female | 0.78226050 | -0.3126079 | -0.2447849 | Yes | Self-employed | Rural | 202.21 | 34.0 | never smoked | |
| 3 | 31112 | Male | 1.62498167 | -0.3126079 | 4.0825625 | Yes | Private | Rural | 105.92 | 32.5 | never smoked | |
| 5 | 1665 | Female | 1.58062792 | 3.1968153 | -0.2447849 | Yes | Self-employed | Rural | 174.12 | 24.0 | never smoked | |
| 6 | 56669 | Male | 1.66933541 | -0.3126079 | -0.2447849 | Yes | Private | Urban | 186.21 | 29.0 | formerly smoked | |
| 15 | 5317 | Female | 1.58062792 | -0.3126079 | 4.0825625 | Yes | Private | Urban | 214.09 | 28.2 | never smoked | |
| 23 | 68794 | Female | 1.58062792 | -0.3126079 | -0.2447849 | Yes | Self-employed | Urban | 228.70 | 26.6 | never smoked | |
| 27 | 38047 | Female | 0.95967548 | -0.3126079 | -0.2447849 | Yes | Private | Rural | 100.98 | 28.2 | formerly smoked | |
| 28 | 61843 | Male | 0.64919926 | -0.3126079 | -0.2447849 | Yes | Private | Rural | 189.84 | 34.0 | Unknown | |
| 29 | 54827 | Male | 1.13709046 | -0.3126079 | 4.0825625 | Yes | Self-employed | Urban | 195.23 | 28.3 | smokes | |
| 33 | 39373 | Female | 1.71368916 | 3.1968153 | -0.2447849 | Yes | Self-employed | Urban | 196.92 | 22.2 | never smoked | |
| 41 | 4651 | Male | 1.53627418 | -0.3126079 | -0.2447849 | Yes | Private | Rural | 78.03 | 23.9 | formerly smoked | |
| 44 | 1845 | Female | 0.87096799 | -0.3126079 | -0.2447849 | Yes | Private | Urban | 90.90 | 34.0 | formerly smoked | |
| 45 | 7937 | Male | 0.73790675 | 3.1968153 | -0.2447849 | Yes | Govt_job | Urban | 213.03 | 20.2 | smokes | |
| 48 | 47472 | Female | 0.64919926 | -0.3126079 | -0.2447849 | Yes | Private | Urban | 107.26 | 38.6 | formerly smoked | |
| 56 | 25831 | Male | 0.87096799 | -0.3126079 | 4.0825625 | Yes | Private | Rural | 196.71 | 36.5 | formerly smoked | |
| 57 | 38829 | Female | 1.71368916 | -0.3126079 | -0.2447849 | Yes | Private | Rural | 59.32 | 33.2 | never smoked | |
| 59 | 58631 | Male | 1.31450545 | 3.1968153 | -0.2447849 | Yes | Self-employed | Urban | 194.99 | 32.8 | never smoked | |
| 63 | 65842 | Female | 1.04838297 | 3.1968153 | -0.2447849 | Yes | Self-employed | Rural | 61.94 | 25.3 | smokes | |
| 65 | 7356 | Male | 1.40321294 | -0.3126079 | -0.2447849 | Yes | Private | Urban | 104.72 | 34.0 | Unknown | |
| 66 | 17013 | Male | 1.53627418 | 3.1968153 | -0.2447849 | No | Private | Urban | 113.01 | 24.0 | never smoked | |

## Test_set

MINPROJECT.r* × training_set × test_set × dataset × stroke × dataframe ×

Filter

| | id | gender | age | hypertension | heart_disease | ever_married | work_type | Residence_type | avg_glucose_level | bmi | smoking_status |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 51676 | Female | 0.78226050 | -0.3126079 | -0.2447849 | Yes | Self-employed | Rural | 202.21 | 34.0 | never smoked |
| 3 | 31112 | Male | 1.62498167 | -0.3126079 | 4.0825625 | Yes | Private | Rural | 105.92 | 32.5 | never smoked |
| 5 | 1665 | Female | 1.58062792 | 3.1968153 | -0.2447849 | Yes | Self-employed | Rural | 174.12 | 24.0 | never smoked |
| 6 | 56669 | Male | 1.66933541 | -0.3126079 | -0.2447849 | Yes | Private | Urban | 186.21 | 29.0 | formerly smoked |
| 15 | 5317 | Female | 1.58062792 | -0.3126079 | 4.0825625 | Yes | Private | Urban | 214.09 | 28.2 | never smoked |
| 23 | 68794 | Female | 1.58062792 | -0.3126079 | -0.2447849 | Yes | Self-employed | Urban | 228.70 | 26.6 | never smoked |
| 27 | 38047 | Female | 0.95967548 | -0.3126079 | -0.2447849 | Yes | Private | Rural | 100.98 | 28.2 | formerly smoked |
| 28 | 61843 | Male | 0.64919926 | -0.3126079 | -0.2447849 | Yes | Private | Rural | 189.84 | 34.0 | Unknown |
| 29 | 54827 | Male | 1.13709046 | -0.3126079 | 4.0825625 | Yes | Self-employed | Urban | 195.23 | 28.3 | smokes |
| 33 | 39373 | Female | 1.71368916 | 3.1968153 | -0.2447849 | Yes | Self-employed | Urban | 196.92 | 22.2 | never smoked |
| 41 | 4651 | Male | 1.53627418 | -0.3126079 | -0.2447849 | Yes | Private | Rural | 78.03 | 23.9 | formerly smoked |
| 44 | 1845 | Female | 0.87096799 | -0.3126079 | -0.2447849 | Yes | Private | Urban | 90.90 | 34.0 | formerly smoked |
| 45 | 7937 | Male | 0.73790675 | 3.1968153 | -0.2447849 | Yes | Govt_job | Urban | 213.03 | 20.2 | smokes |
| 48 | 47472 | Female | 0.64919926 | -0.3126079 | -0.2447849 | Yes | Private | Urban | 107.26 | 38.6 | formerly smoked |

## Fitting SVM to the Training set

R 4.1.2 · ~/

```
> classifier = svm(formula = stroke ~ .,
+                  data = test_set,
+                  type = 'C-classification',
+                  kernel = 'linear')
```

```
Data
classifier                List of  30
    $ call           : language svm(formula = stroke ~ ., data = test_set, type = "C-classification", ke…
    $ type           : num 0
    $ kernel         : num 0
    $ cost           : num 1
    $ degree         : num 3
    $ gamma          : num 0.0588
    $ coef0          : num 0
    $ nu             : num 0.5
    $ epsilon        : num 0.1
    $ sparse         : logi FALSE
    $ scaled         : logi [1:17] TRUE FALSE FALSE TRUE TRUE TRUE ...
    $ x.scale        :List of 2
     ..$ scaled:center: Named num [1:6] 3.64e+04 -5.97e-17 3.47e-17 1.25e-17 1.07e+02 ...
     .. ..- attr(*, "names")= chr [1:6] "id" "age" "hypertension" "heart_disease" ...
     ..$ scaled:scale : Named num [1:6] 20746.5 1 1 1 45.8 ...
     .. ..- attr(*, "names")= chr [1:6] "id" "age" "hypertension" "heart_disease" ...
    $ y.scale        : NULL
    $ nclasses       : int 2
    $ levels         : chr [1:2] "0" "1"
    $ tot.nSV        : int 247
    $ nSV            : int [1:2] 70 177
    $ labels         : int [1:2] 2 1
    $ SV             : num [1:247, 1:17] 0.737 -0.254 -1.673 0.978 -1.497 ...
     ..- attr(*, "dimnames")=List of 2
     .. ..$ : chr [1:247] "2" "3" "5" "6" ...
     .. ..$ : chr [1:17] "id" "genderFemale" "genderMale" "age" ...
    $ index          : int [1:247] 1 2 3 4 5 6 7 8 9 10 ...
    $ rho            : num 0.999
    $ compprob       : logi FALSE
    $ probA          : NULL
    $ probB          : NULL
    $ sigma          : NULL
    $ coefs          : num [1:247, 1] 1 1 1 1 1 1 1 1 1 1 ...
    $ na.action      : NULL
    $ fitted         : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
     ..- attr(*, "names")= chr [1:1538] "2" "3" "5" "6" ...
    $ decision.values: num [1:1538, 1] -1 -1 -1 -1 -1 ...
     ..- attr(*, "dimnames")=List of 2
     .. ..$ : chr [1:1538] "2" "3" "5" "6" ...
     .. ..$ : chr "1/0"
    $ terms          :Classes 'terms', 'formula'  language stroke ~ id + gender + age + hypertension + h…
     .. ..- attr(*, "variables")= language list(stroke, id, gender, age, hypertension, heart_disease, ev…
     .. ..- attr(*, "factors")= int [1:12, 1:11] 0 1 0 0 0 0 0 0 0 0 ...
     .. .. ..- attr(*, "dimnames")=List of 2
     .. .. .. ..$ : chr [1:12] "stroke" "id" "gender" "age" ...
     .. .. .. ..$ : chr [1:11] "id" "gender" "age" "hypertension" ...
     .. ..- attr(*, "term.labels")= chr [1:11] "id" "gender" "age" "hypertension" ...
     .. ..- attr(*, "order")= int [1:11] 1 1 1 1 1 1 1 1 1 1 ...
     .. ..- attr(*, "intercept")= num 0
     .. ..- attr(*, "response")= int 1
     .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
     .. ..- attr(*, "predvars")= language list(stroke, id, gender, age, hypertension, heart_disease, eve…
     .. ..- attr(*, "dataClasses")= Named chr [1:12] "numeric" "numeric" "character" "numeric" ...
     .. .. ..- attr(*, "names")= chr [1:12] "stroke" "id" "gender" "age" ...
    - attr(*, "class")= chr [1:2] "svm.formula" "svm"
```

## Predicting test data using a classifier model

```
> # Predicting on test data'
> y_pred1 <- predict(classifier, newdata1 = test_set)
> y_pred1
   2    3    5    6   15   23   27   28   29   33   41   44   45   48   56   57   59   63   65   66   68   70   71   74
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
  75   77   79   84   88   89   91   92   94   98  112  116  117  125  126  139  142  143  149  150  153  157  158  161
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 162  163  170  174  175  177  184  188  189  191  193  204  205  213  231  233  237  238  242  243  247  249  252  253
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 254  255  256  257  262  263  269  275  276  279  288  294  296  297  300  303  306  308  310  312  315  318  321  325
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 326  331  332  333  337  338  342  348  349  350  351  354  359  360  367  373  376  380  385  387  389  393  397  398
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 406  408  417  418  423  424  429  436  437  439  440  442  447  450  455  458  459  466  467  472  473  475  477  478
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 480  482  483  488  489  491  493  501  507  526  532  536  537  541  546  549  553  563  566  583  585  587  588  596
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 603  608  611  612  616  618  620  621  624  627  628  629  633  638  644  649  654  659  660  663  667  670  676  678
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 681  686  688  690  691  695  702  707  709  713  718  720  721  728  729  737  739  740  741  743  745  751  752  754
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 757  772  777  783  784  786  788  792  793  794  795  797  799  802  803  804  817  818  821  822  825  826  828  834
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 838  841  842  844  850  853  855  859  860  862  864  865  866  869  870  871  874  880  884  885  888  890  892  893
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 895  900  902  904  907  909  919  925  928  929  930  937  939  940  944  945  948  958  962  964  966  967  977  979
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
 984  986  989  990  992  996 1001 1002 1003 1012 1018 1019 1021 1022 1024 1025 1030 1033 1034 1035 1036 1039 1043 1044
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
1045 1050 1051 1053 1056 1058 1060 1063 1066 1067 1081 1083 1087 1089 1093 1095 1097 1098 1108 1109 1110 1111 1113 1114
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
1116 1120 1122 1123 1134 1137 1138 1141 1142 1144 1157 1160 1166 1170 1173 1179 1187 1192 1193 1195 1201 1202 1206 1209
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
1211 1213 1215 1218 1220 1221 1222 1223 1226 1227 1228 1231 1234 1236 1240 1242 1250 1252 1266 1270 1279 1289 1296 1306
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
1308 1309 1312 1316 1317 1324 1340 1347 1349 1350 1352 1355 1361 1363 1364 1369 1375 1376 1379 1381 1383 1391 1393 1394
   0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
```

## Confusion matrix

```
# Confusion Matrix
cm2 <- table(test_set$stroke, y_pred1)
cm2
```

```
R 4.1.2 · ~/
> # Confusion Matrix
> cm2 <- table(test_set$stroke, y_pred1)
> cm2
   y_pred1
      0    1
  0 1468    0
  1   70    0
```

```
y_pred1                    Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

# Data Visualization and Interpretation

**Data Visualization and Interpretation**

- **Installing packages**

```
R 4.1.2 · ~/
> install.packages("Quandl")
Installing package into 'C:/Users/dharan/OneDrive/Documents/R/win-library/4.1'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/Quandl_2.11.0.zip'
Content type 'application/zip' length 70466 bytes (68 KB)
downloaded 68 KB

package 'Quandl' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\dharan\AppData\Local\Temp\Rtmpqoui4e\downloaded_packages
>
```

```
> install.packages("ggplot2")
Installing package into 'C:/Users/dharan/OneDrive/Documents/R/win-library/4.1'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.1/ggplot2_3.3.5.zip'
Content type 'application/zip' length 4130497 bytes (3.9 MB)
downloaded 3.9 MB

package 'ggplot2' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\dharan\AppData\Local\Temp\Rtmpqoui4e\downloaded_packages
>
```

- **Importing Necessary Libraries**

```
> library(Quandl)
Loading required package: xts
Loading required package: zoo

Attaching package: 'zoo'

The following objects are masked from 'package:base':

    as.Date, as.Date.numeric

Warning message:
package 'Quandl' was built under R version 4.1.3
>
```

```
> library(ggplot2)
Need help getting started? Try the R Graphics Cookbook: https://r-graphics.org
Warning message:
package 'ggplot2' was built under R version 4.1.3
>
```

1. **Histogram**

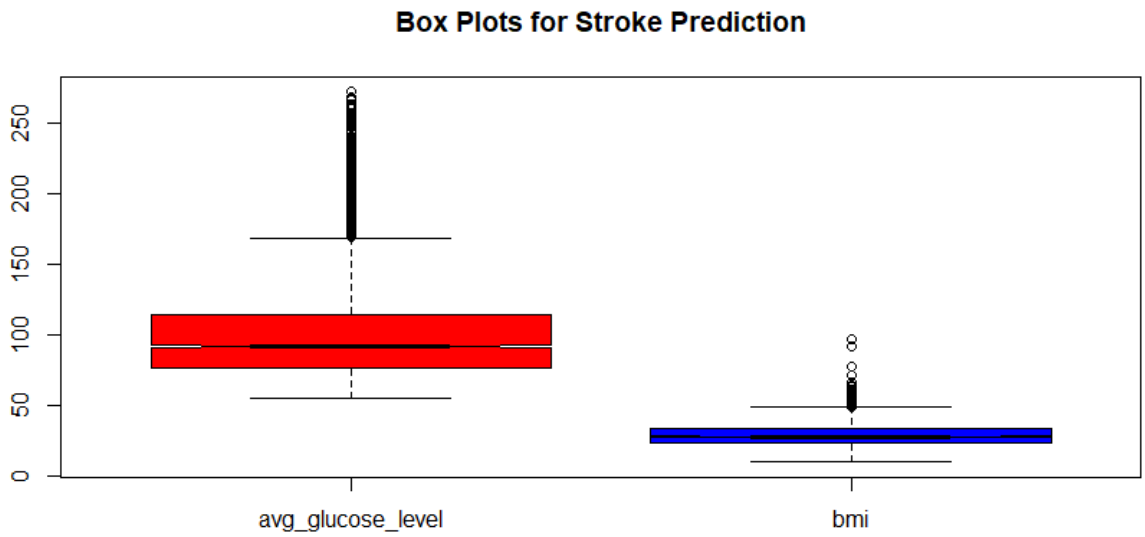
```r
#histogram
hist(stroke$age,
     main="No of stroke cases classified on the basis of age",
     xlab="age",
     xlim = c(0,100),
     col = c(0,2,4,6,8,10,12,14,16,18,20,24,26),
     las=1)
```

```
> #histogram
> hist(stroke$age,
+      main="No of stroke cases classified on the basis of age",
+      xlab="age",
+      xlim = c(0,100),
+      col = c(0,2,4,6,8,10,12,14,16,18,20,24,26),
+      las=1)
```


## 2. Boxplot

```r
#Boxplot Multiple Box plots, each representing Stroke prediction Parameter
boxplot(stroke[, 9:10],
        col=c("Red","Blue"),
        notch = TRUE,
        main ='Box Plots for Stroke Prediction')
```

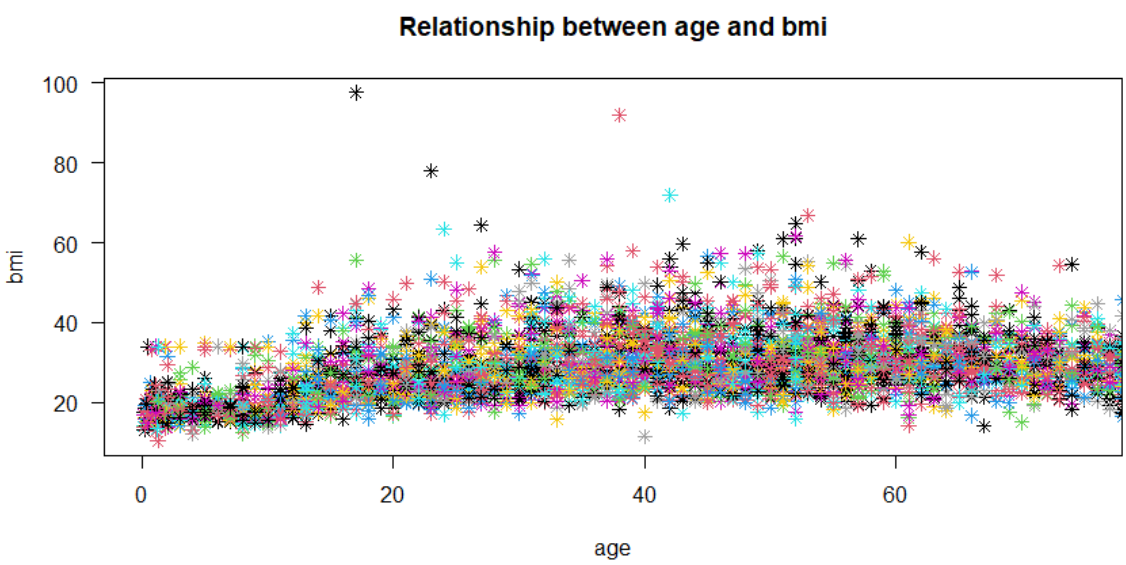
```
> #Boxplot Multiple Box plots, each representing Stroke prediction Parameter
> boxplot(stroke[, 9:10],
+         col=c("Red","Blue"),
+         notch = TRUE,
+         main ='Box Plots for Stroke Prediction')
>
```




## 3. Scatter Plot

```
#SCatterPlot
plot(stroke$age,stroke$bmi,
     main = "Relationship between age and bmi",
     xlab = "age",
     ylab = "bmi",
     xlim=c(0,75),
     las=1,
     col=1:10,
     pch =8)
```

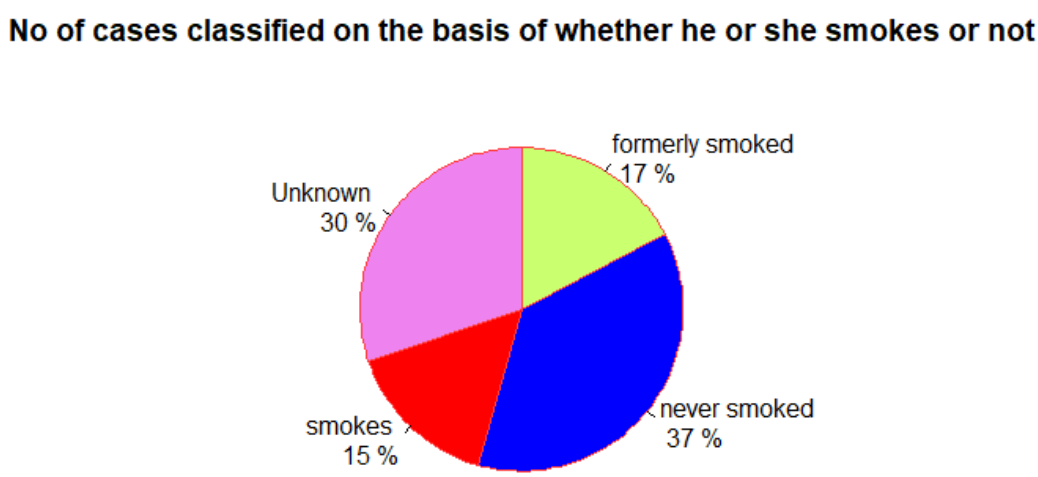
```
> #SCatterPlot
> plot(stroke$age,stroke$bmi,
+      main = "Relationship between age and bmi",
+      xlab = "age",
+      ylab = "bmi",
+      xlim=c(0,75),
+      las=1,
+      col=1:10,
+      pch =8)
```



## 4. Pie Chart

```r
#Piechart with proportion
count<-table(stroke$smoking_status)
percent<-round(count/sum(count)*100)
genderlabel<-paste(names(count),"\n",percent,"%")
pie(count,
    labels = genderlabel,
    main = "No of cases classified on the basis of whether he or she smokes or not",
    col=c("darkolivegreen1","blue","red","violet"),
    border="brown1",
    clockwise = TRUE)
```

```
> #Piechart with proportion
> count<-table(stroke$smoking_status)
> percent<-round(count/sum(count)*100)
> genderlabel<-paste(names(count),"\n",percent,"%")
> pie(count,
+     labels = genderlabel,
+     main = "No of cases classified on the basis of whether he or she smokes or not",
+     col=c("darkolivegreen1","blue","red","violet"),
+     border="brown1",
+     clockwise = TRUE)
```



# Conclusion

## Conclusion:

In this project using the R programming language, several operations on the Stroke Prediction Dataset were done in this Mini Project. The first step was to do Data Extraction, which includes importing the dataset, examining the data, and understanding the data. The "ggplot" package was used for exploratory data analysis, which includes functions like peek, select, arrange, filter, summarize and count. Finally, the dataset was subjected to Data Visualization, which included the creation of point graphs, histograms, barplots, box plots, and scatterplots with various variants and filters. The dataset has also been subjected to correlation and covariance analysis. As a result, the R Programming Language was used to understand and perform numerous operations, and the project was completed successfully.

**ACKNOWLEDGEMENT**

We would like to thank our guide, Ms. Vandana Patil, for allowing us to conduct a project on "Stroke Prediction," which has really aided our grasp of R. Her continual advice and suggestions have aided us in moving forward with our project. We would also want to thank Dr. Joanne Gomes, Head of Department (INFT), our Principal, Director, and all of the faculty members and staff for providing us with the necessary facilities for the project.