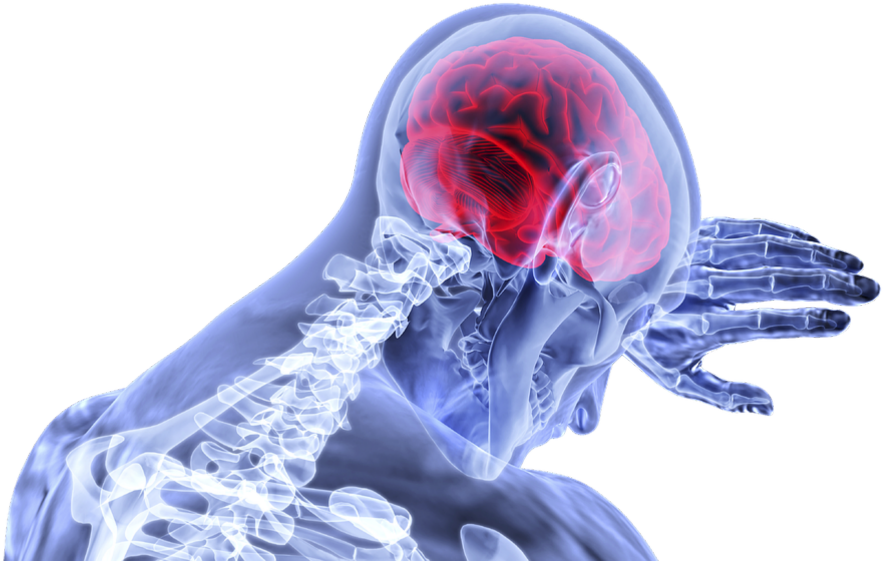


BRAIN Stroke PREDICTION

Team Members :

Aayushi Gautam

Priyanshu Raman

Aishwarya Pakala

Vaishanv Baswe

Shravika Reddy Gudepu

Executive Summary

The World Health Organization (WHO) defines stroke as “rapidly developing clinical signs of focal (or global) disturbance of cerebral function, lasting more than 24 hours or leading to death, with causes that are of vascular origin. Symptoms of stroke include trouble walking, speaking and understanding, as well as paralysis or numbness of the face, arm, or leg. It is a dreaded situation for any patient. To understand the long term effects of lifestyle and the activities that the patients were involved in before they suffered stroke can conclude on interpreting the trends that led to maximum cases.

There are various risk factors associated with the onset of stroke in an individual. However, there has been limited use of data mining on patients’ medical records to study the inter dependency of different risk factors of stroke. It has now become easier to collect healthcare from multiple sources and also insights obtained from mining stroke data will be useful for decision making to improve health care.

In this project, our goal is to perform an analysis of patients’ records to identify the impact of risk factors on stroke prediction. We attempt to analyze the correlation between different risk factors for stroke prediction using ggplot and pie charts.We aim to identify the risk factors associated with stroke. We employed different classification methods involving Decision Tree, Random Forest and Logistic Regression. We also compare the accuracy and Roc curve for every predicted model to provide the most appropriate method. By analysing the final results, we define the tendency of a potential case and help in early diagnosis and treatment that can minimize brain damage.

Project Motivation

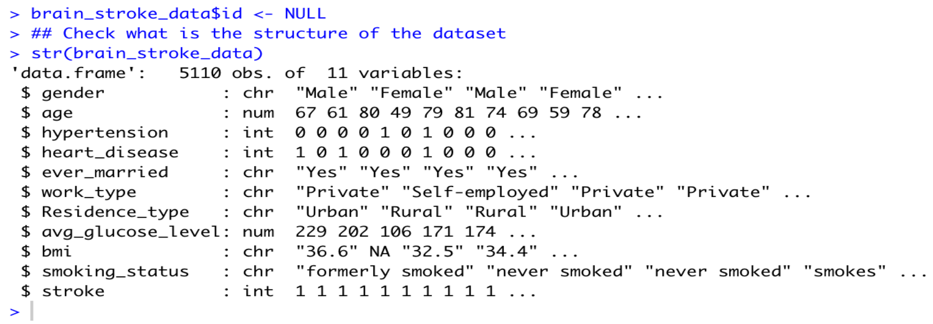
Stroke is the 2nd leading cause of death globally, responsible for approximately 11% of total deaths. We are creating a model which predicts whether a person will have a stroke or not. This can be very useful for the healthcare industry. If certain factors are found in patients, the hospital can recognize heart strokes early. This could be extremely helpful for the patients as they can receive intensive healthcare before their condition gets serious. Knowing whether a person will have a stroke in the future could aid in saving a lot of lives and money.

Data Description

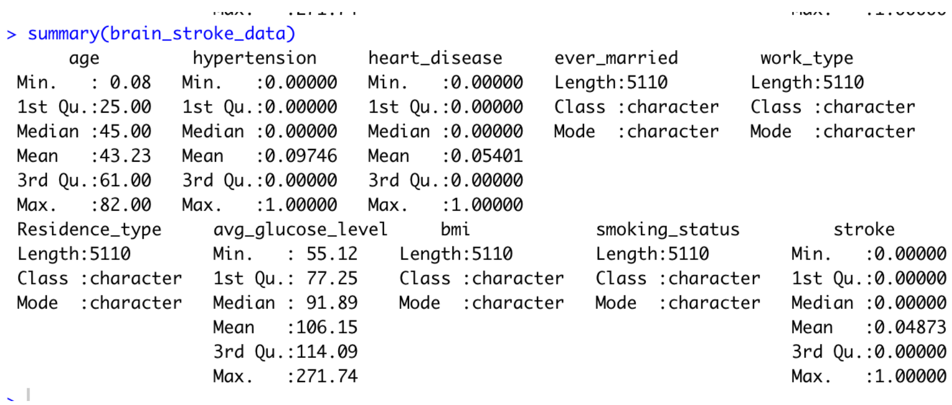
The healthcare dataset consists of 5100 records of both male and female patients and is a second hand dataset from Kaggle.This dataset contains 12 attributes in which 11 are input variables . These variables include the patient id, gender(Male or Female), age, hypertension (binary status if the person has hypertension or not) , heart disease (binary status if the person has heart disease or not) , ever married (status as Yes or No), work type (Categorical values - Self employed, Government,private), residence type(If a person resides in urban or rural), average glucose level, BMI, smoking status (smokes, formerly smoked or never smoked) and stroke (If a person has suffered from stroke or not).

In our report, we consider the 10 variables (excluding the patient id) as the input variables, and the binary status of the stroke variable as the output variable for predicting a model. This data is further cleaned and transformed into a new dataset.

Figures below shows the class and summaries of each variable-



```
> brain_stroke_data$id <- NULL
> ## Check what is the structure of the dataset
> str(brain_stroke_data)
'data.frame':   5110 obs. of  11 variables:
 $ 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              : chr  "36.6" NA "32.5" "34.4" ...
 $ smoking_status   : chr  "formerly smoked" "never smoked" "never smoked" "smokes" ...
 $ stroke           : int  1 1 1 1 1 1 1 1 1 1 ...
> 
```



```
> summary(brain_stroke_data)
      age         hypertension       heart_disease      ever_married        work_type        
 Min.   : 0.08   Min.   :0.00000   Min.   :0.00000   Length:5110        Length:5110       
 1st Qu.:25.00   1st Qu.:0.00000   1st Qu.:0.00000   Class :character   Class :character  
 Median :45.00   Median :0.00000   Median :0.00000   Mode  :character   Mode  :character  
 Mean   :43.23   Mean   :0.09746   Mean   :0.05401                                        
 3rd Qu.:61.00   3rd Qu.:0.00000   3rd Qu.:0.00000                                        
 Max.   :82.00   Max.   :1.00000   Max.   :1.00000                                        
 Residence_type     avg_glucose_level      bmi            smoking_status         stroke       
 Length:5110        Min.   : 55.12    Length:5110        Length:5110        Min.   :0.00000  
 Class :character   1st Qu.: 77.25    Class :character   Class :character   1st Qu.:0.00000  
 Mode  :character   Median : 91.89    Mode  :character   Mode  :character   Median :0.00000  
                    Mean   :106.15                                          Mean   :0.04873  
                    3rd Qu.:114.09                                          3rd Qu.:0.00000  
                    Max.   :271.74                                          Max.   :1.00000  
> 
```

Data pre-processing and cleaning

This dataset contains various null and unknown values. The columns that have null values are age, bmi (which are integer variables) and smoking status (character variables in which null values are given as unknown). To get rid of the null values, we performed imputations on them. We used the mean of the variable as the imputed value.

We made bar charts to exhibit the missing values before and after cleaning the data.

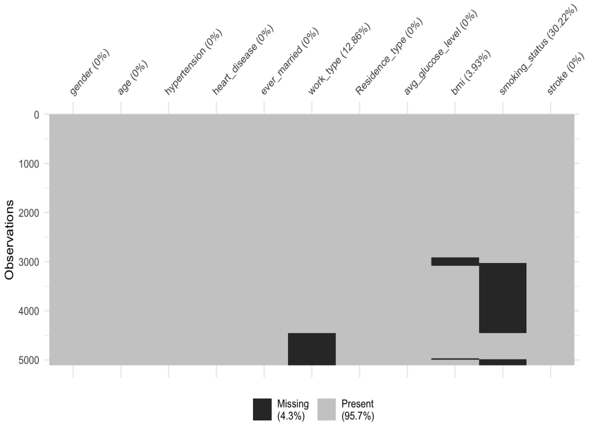


Fig 1: Before cleaning the data

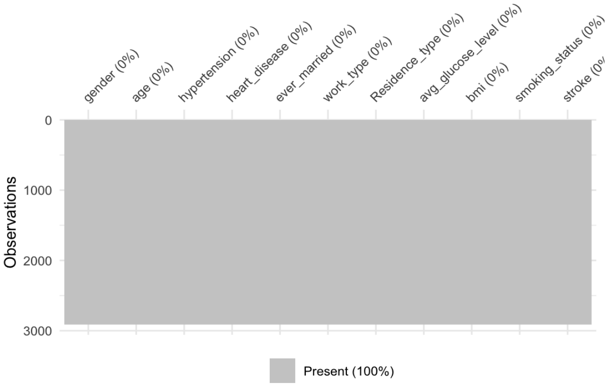


Fig 2 : After cleaning the data

We can see that after imputing the values, there are no missing values in the data.

Exploratory data analysis

To better understand the data, we have generated various charts. These charts show us how various factors are responsible for causing a stroke in a patient.

1. Pie chart representing patients who had stroke based on hypertension.

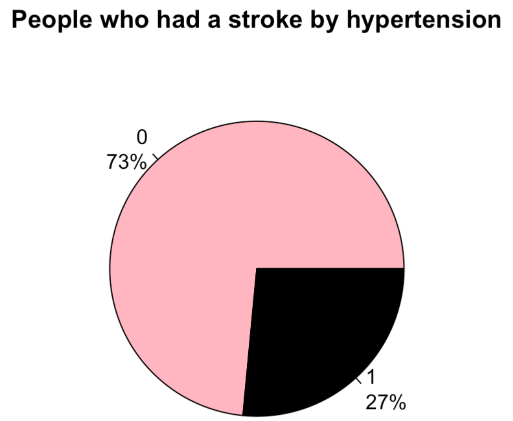


Fig 3: Pie Chart 1

Label	
0	Don’t have hypertension
1	Have Hypertension

From the pie chart above we observe that 73% of people who have a stroke do not suffer from hypertension and 27% of people who have a stroke suffer from hypertension.

2. Pie chart representing stroke patients based of their smoking status

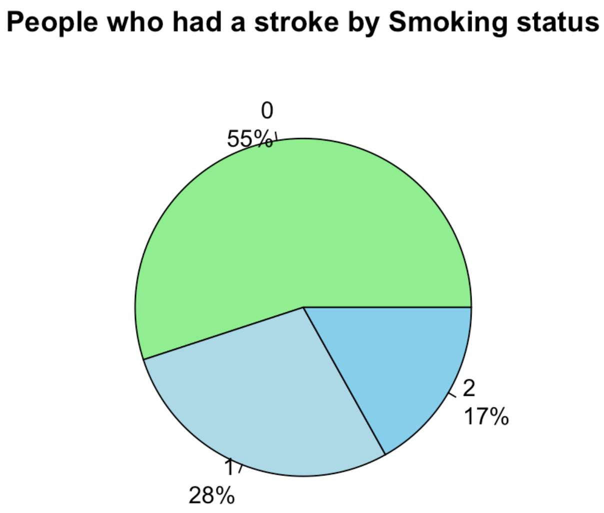


Fig 4: Pie Chart 2

Label	
0	Never smoked
1	Formerly smoked
2	Smoked

From the pie chart above we observe that 55% of patients suffering from stroke never smoked, 28% did smoke in the past and 17% are present day smokers.

3. Pie chart of stroke patients based on their gender

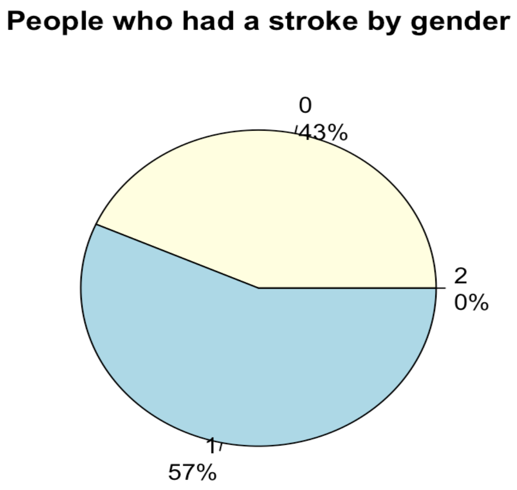


Fig 5: Pie Chart 3

Label	
0	Male
1	Female
2	Others

From pie chart 3 we understand that amongst all stroke patients 44% were male and 56% were female.

4. Scatter plot of stroke distribution based on BMI and age

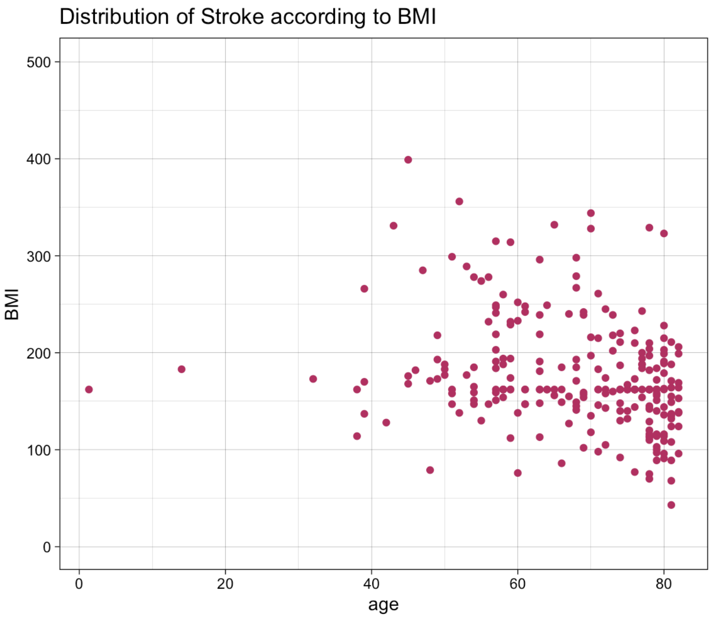


Fig 6: Distribution of Stroke patients with respect to BMI and age

5. Bar chart showing different age groups who have stroke

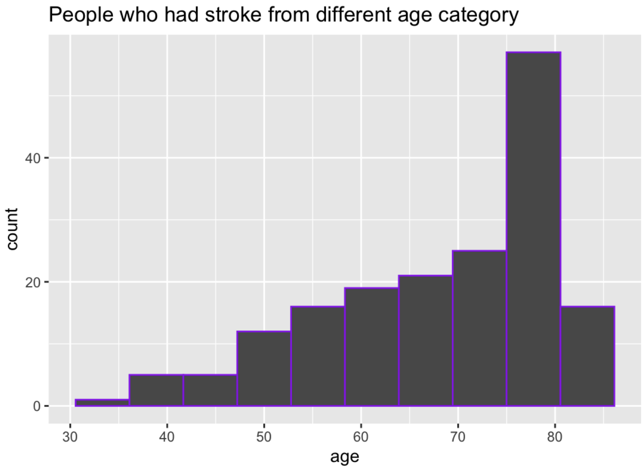


Fig 7: Stroke frequency of different age groups

The above plot shows the stroke frequency in different age ranges, according to the observation the maximum number of patients belong to the age range of 75 years to 80 years.

Box plots

1.

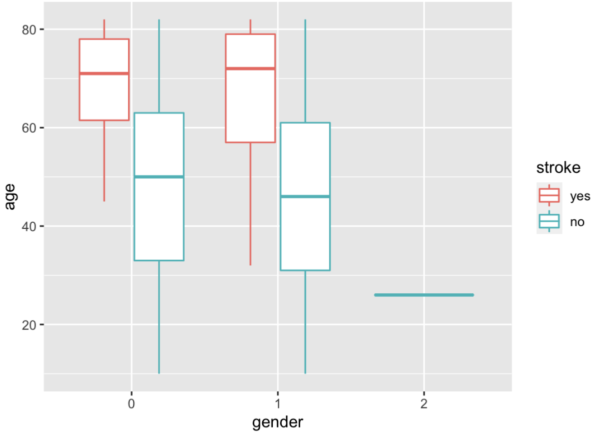


Fig 8: Box plot representing the patients on the basis of age and gender

According to the observation from the box plot (fig8), among all the females from different age groups, mostly stroke patients were over 60 years and below 80 years of age. Similarly the age range for male patients was mostly over 55 years and below 80 years of age.

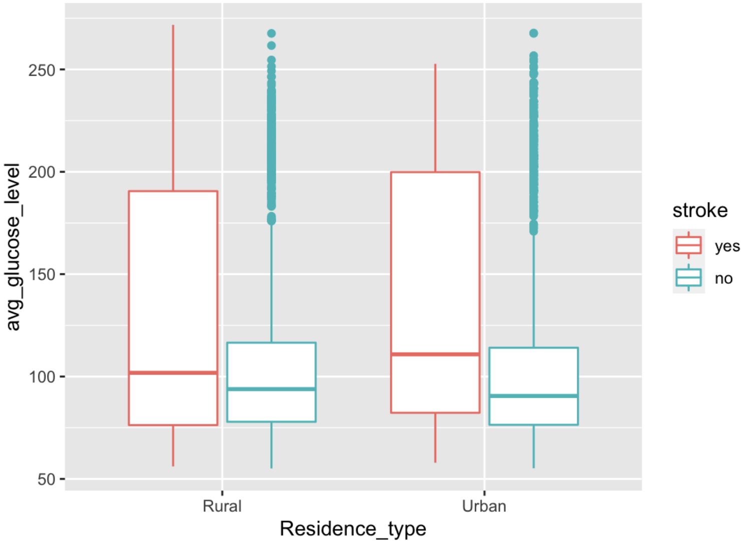


Fig 9: This represents the patients on the basis of average glucose level and residence type.

According to the observation from the box plot (fig9), among all the people, who reside in urban regions , the mean glucose level of the person who has suffered a stroke is high. If a person has higher, he is more prone to stroke than the people with lower glucose level. The people who have low glucose levels are less likely prone to stroke irrespective of the region they reside

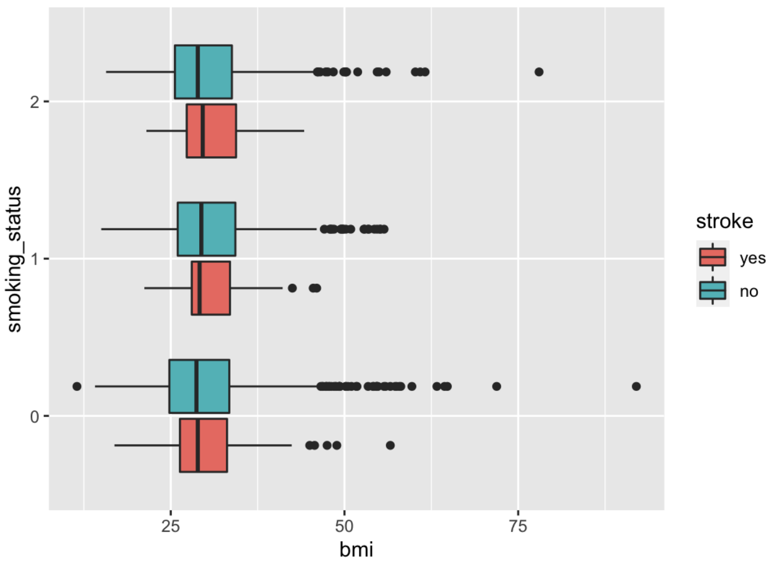


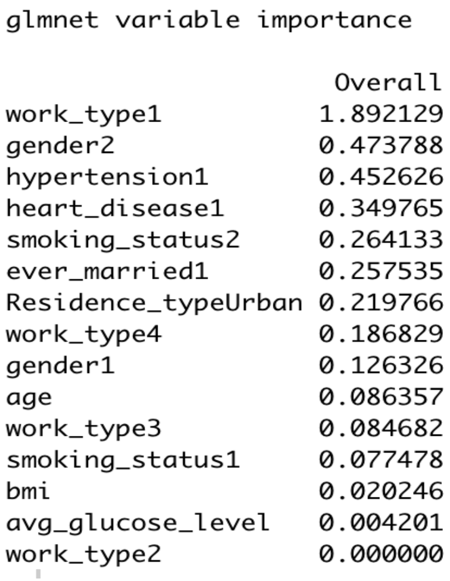
Fig 10: Box plot representing smoking status and bmi of a person

According to the above plot, 0 represents the people who never smoked, 1 represents the set of people who formerly smoked and 2 represents the set of people who smokes. People who have smoked and have higher BMI are more likely prone to stroke.

BI models used

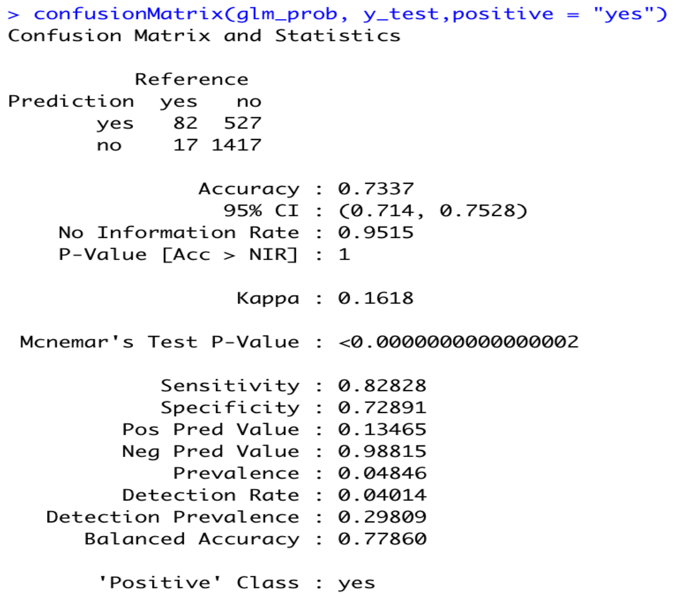
1. Logistic Regression

We have split our data into subsets and used each subset as a training dataset to predict a model and another one as a validation dataset to validate the predicted model. As we have a lot of unbalanced data, we used an over sampling method to get the accurate results.



```
glmnet variable importance

                    Overall
work_type1         1.892129
gender2            0.473788
hypertension1      0.452626
heart_disease1     0.349765
smoking_status2    0.264133
ever_married1      0.257535
Residence_typeUrban 0.219766
work_type4         0.186829
gender1            0.126326
age                0.086357
work_type3         0.084682
smoking_status1    0.077478
bmi                0.020246
avg_glucose_level  0.004201
work_type2         0.000000
```

The above output shows us the importance of different variables in our model. Work type1



```
> confusionMatrix(glm_prob, y_test,positive = "yes")
Confusion Matrix and Statistics

          Reference
Prediction  yes   no
       yes   82  527
       no    17 1417

               Accuracy : 0.7337
                 95% CI : (0.714, 0.7528)
    No Information Rate : 0.9515
    P-Value [Acc > NIR] : 1

                  Kappa : 0.1618

 Mcnemar's Test P-Value : <0.0000000000000002

            Sensitivity : 0.82828
            Specificity : 0.72891
         Pos Pred Value : 0.13465
         Neg Pred Value : 0.98815
             Prevalence : 0.04846
         Detection Rate : 0.04014
   Detection Prevalence : 0.29809
      Balanced Accuracy : 0.77860

       'Positive' Class : yes
```

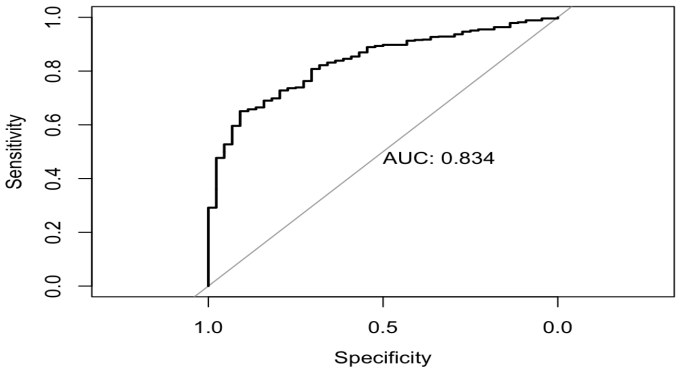


Fig 11: ROC curve for logistic regression

The accuracy rate for the predicted model is 73.37%. This model give higher importance to worktype with coefficient of 1.89 followed by gender (female) with 0.47 . The least importance is given to average glucose level. The area under the curve is 0.834 which is better

2. Decision Tree

A decision tree is a decision support tool that uses a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making.

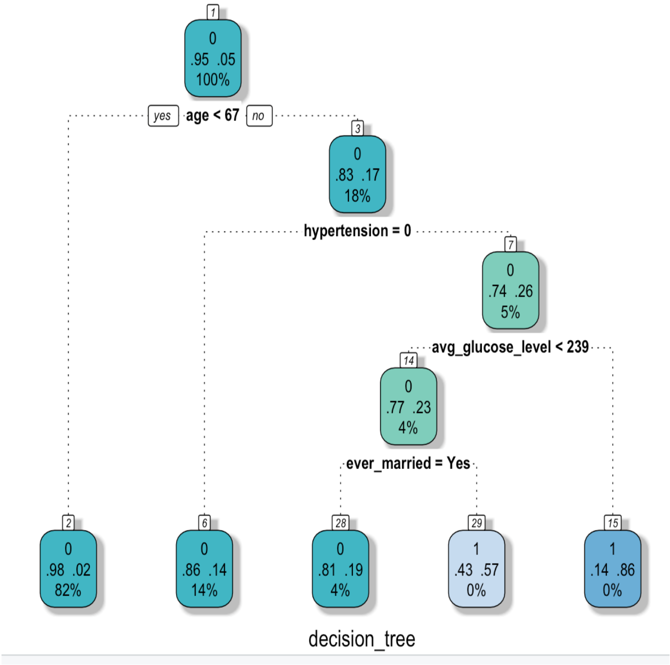


Fig 12: Decision tree obtained

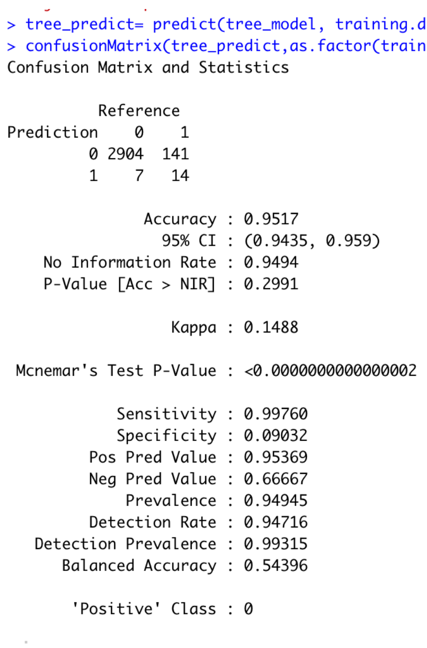
From the above decision tree, following observations can be made -

a) From the first node (the root node), we can say that there is a **98%** chance that a person younger than 67 years, **will not** have a stroke. Data with patients older than 67 years, proceeds to the next step.

b) The second node includes all the people that have hypertension. If a person does not have hypertension, there is an **86%** chance that they **will not** have a stroke. If a person has hypertension, it moves to the next step.

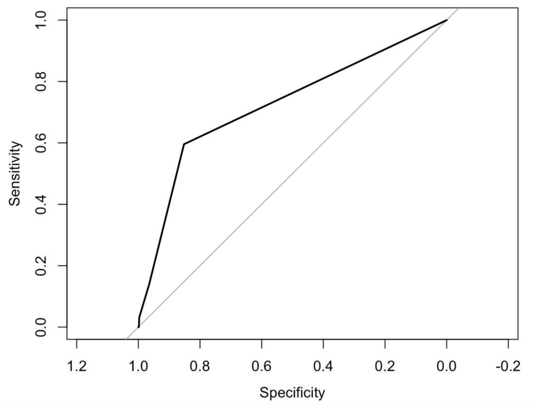
c) We can see from this node that if a person has an average glucose level less than 239, there is a **14%** chance the person **will** have a stroke. If their glucose level is more than 239, it proceeds to the next step.

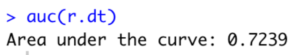
d) If a person is married, there is a **43%** chance that they **could suffer** a stroke. If a person is unmarried, there is an **81%** chance that they **will not** have a stroke.



```
> tree_predict= predict(tree_model, training.df,type="class")
> confusionMatrix(tree_predict,as.factor(training.df$stroke))
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 2904  141
         1    7   14
                                             
               Accuracy : 0.9517             
                 95% CI : (0.9435, 0.959)    
    No Information Rate : 0.9494             
    P-Value [Acc > NIR] : 0.2991             
                                             
                  Kappa : 0.1488             
                                             
 Mcnemar's Test P-Value : <0.0000000000000002
                                             
            Sensitivity : 0.99760            
            Specificity : 0.09032            
         Pos Pred Value : 0.95369            
         Neg Pred Value : 0.66667            
             Prevalence : 0.94945            
         Detection Rate : 0.94716            
   Detection Prevalence : 0.99315            
      Balanced Accuracy : 0.54396            
                                             
       'Positive' Class : 0                  
```

By looking at the confusion matrix, we can say that our model has an accuracy of 95.17% .





```
> auc(r.dt)
Area under the curve: 0.7239
```

Fig 13: ROC curve for decision tree

Since we do not get an accurate representation of the data just by accuracy, we have also computed the ROC curve. The area under the curve (AUC) is 0.72, which means that our model is satisfactory.

3. Random Forest

Random forest is a collection of decision trees. A decision tree is built using all the variables of interest, while random forest is built on the whole dataset running multiple decision trees and then selecting the classes which hold more importance.

Decision trees are easy for interpretation of patterns, but may not provide the best of accuracy. Random forest was done to better understand the effect on ROC and accuracy values for the model. Also for unexpected validation Random forest is discussed to be a better model.

The below bar chart(fig 13) shows the importance of variables that are playing a role in driving the tests.

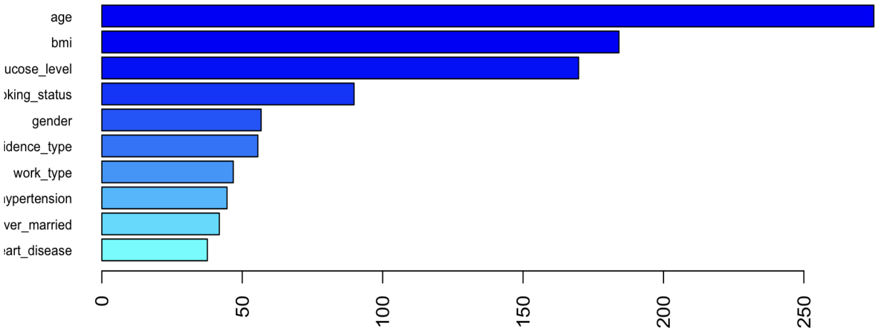


Fig 14: Importance of variables

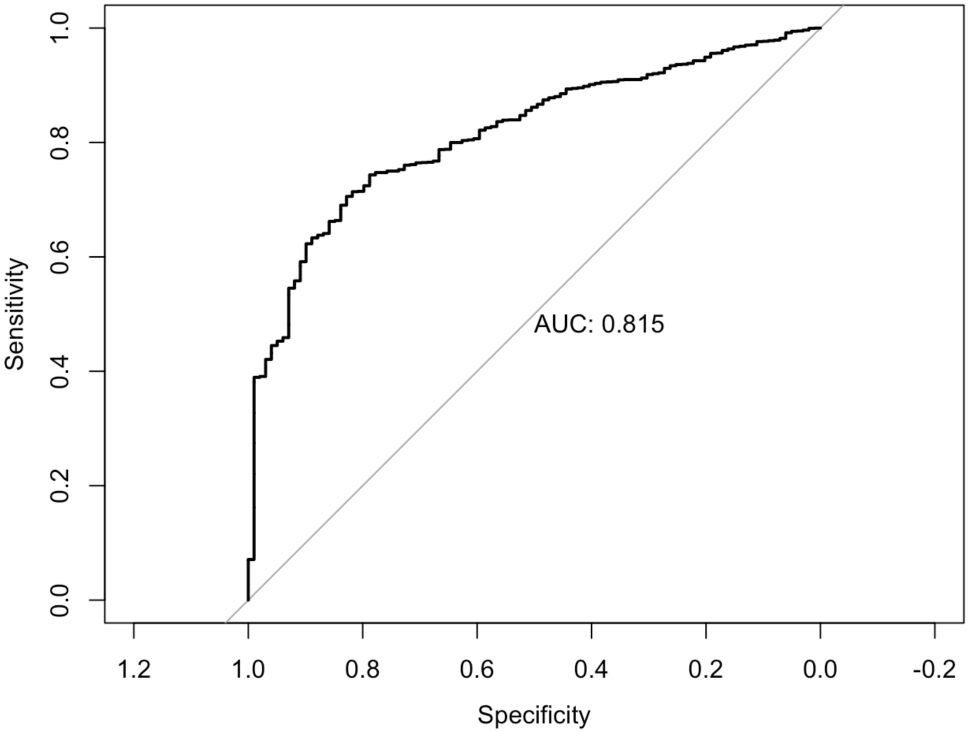
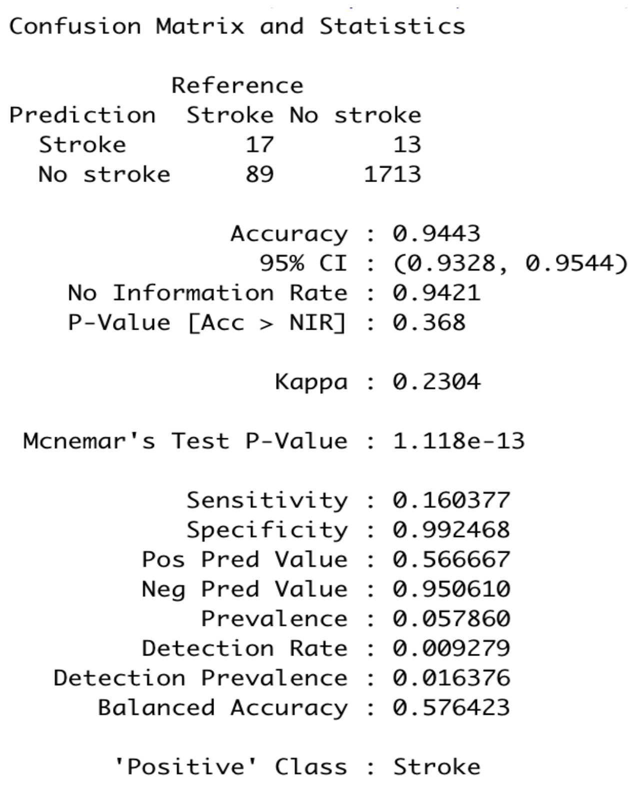

1.0
0.8
0.6
0.4
0.2
0.0
Sensitivity
AUC: 0.815
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
Specificity

Fig 14:

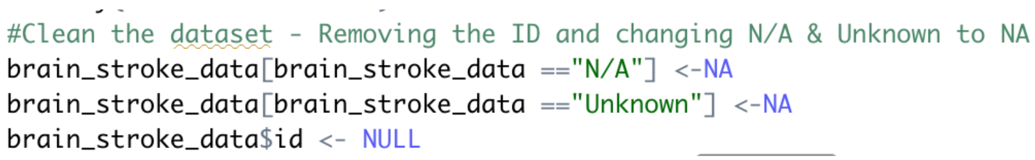


```
Confusion Matrix and Statistics

           Reference
Prediction  Stroke No stroke
  Stroke        17        13
  No stroke     89      1713

               Accuracy : 0.9443
                 95% CI : (0.9328, 0.9544)
    No Information Rate : 0.9421
    P-Value [Acc > NIR] : 0.368

                  Kappa : 0.2304

 Mcnemar's Test P-Value : 1.118e-13

            Sensitivity : 0.160377
            Specificity : 0.992468
         Pos Pred Value : 0.566667
         Neg Pred Value : 0.950610
             Prevalence : 0.057860
         Detection Rate : 0.009279
   Detection Prevalence : 0.016376
      Balanced Accuracy : 0.576423

       'Positive' Class : Stroke
```

The accuracy was 94.43% but because we are not relying upon accuracy rate for any of our models, we got the ROC values for area under the curve to be 0.815, which means the model is working satisfactorily.

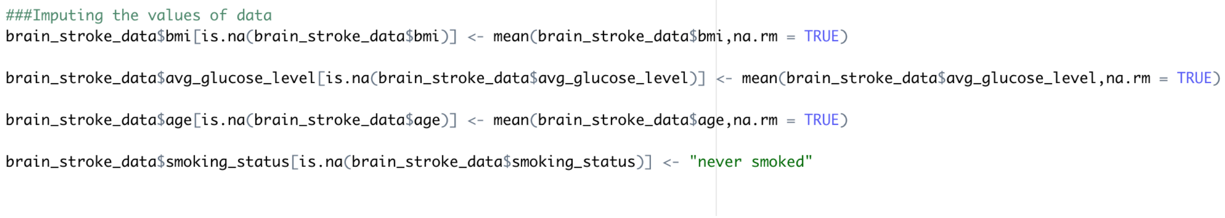
R codes used

1. Cleaning the dataset



```
#Clean the dataset - Removing the ID and changing N/A & Unknown to NA
brain_stroke_data[brain_stroke_data =="N/A"] <-NA
brain_stroke_data[brain_stroke_data =="Unknown"] <-NA
brain_stroke_data$id <- NULL
```

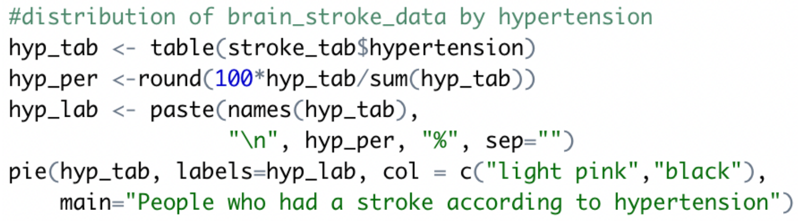
2. Imputation on the null values



```
###Imputing the values of data
brain_stroke_data$bmi[is.na(brain_stroke_data$bmi)] <- mean(brain_stroke_data$bmi,na.rm = TRUE)

brain_stroke_data$avg_glucose_level[is.na(brain_stroke_data$avg_glucose_level)] <- mean(brain_stroke_data$avg_glucose_level,na.rm = TRUE)

brain_stroke_data$age[is.na(brain_stroke_data$age)] <- mean(brain_stroke_data$age,na.rm = TRUE)

brain_stroke_data$smoking_status[is.na(brain_stroke_data$smoking_status)] <- "never smoked"
```

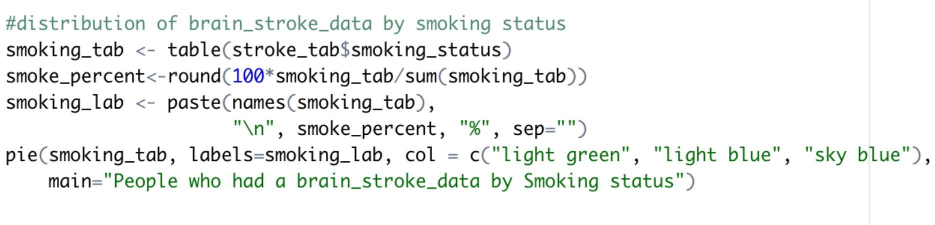
3. Plotting the various charts for stroke on the basis of different variables

- Pie chart for Hypertension



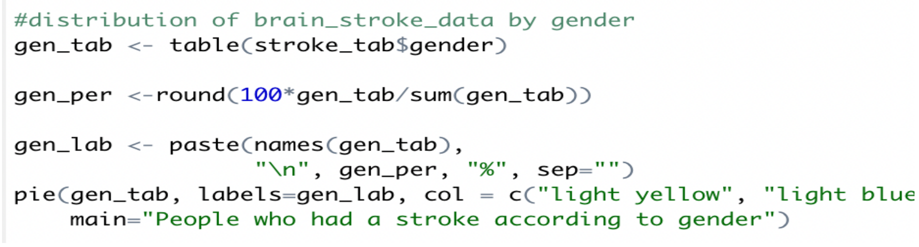
```
#distribution of brain_stroke_data by hypertension
hyp_tab <- table(stroke_tab$hypertension)
hyp_per <-round(100*hyp_tab/sum(hyp_tab))
hyp_lab <- paste(names(hyp_tab),
                 "\n", hyp_per, "%", sep="")
pie(hyp_tab, labels=hyp_lab, col = c("light pink","black"),
    main="People who had a stroke according to hypertension")
```

- Pie chart for Smoking status



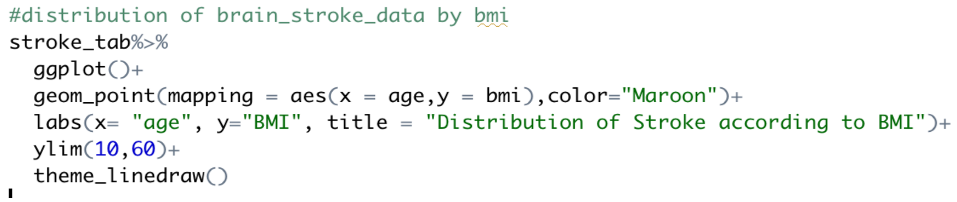
```
#distribution of brain_stroke_data by smoking status
smoking_tab <- table(stroke_tab$smoking_status)
smoke_percent<-round(100*smoking_tab/sum(smoking_tab))
smoking_lab <- paste(names(smoking_tab),
                     "\n", smoke_percent, "%", sep="")
pie(smoking_tab, labels=smoking_lab, col = c("light green", "light blue", "sky blue"),
    main="People who had a brain_stroke_data by Smoking status")
```

- <u>Pie chart for Gender</u>



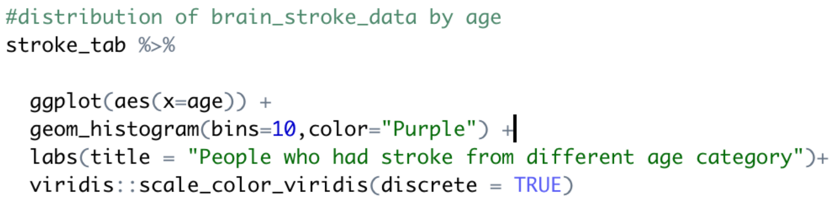
```
#distribution of brain_stroke_data by gender
gen_tab <- table(stroke_tab$gender)

gen_per <-round(100*gen_tab/sum(gen_tab))

gen_lab <- paste(names(gen_tab),
                 "\n", gen_per, "%", sep="")
pie(gen_tab, labels=gen_lab, col = c("light yellow", "light blue
    main="People who had a stroke according to gender")
```

- <u>Scatter plot on the basis of bmi and age</u>



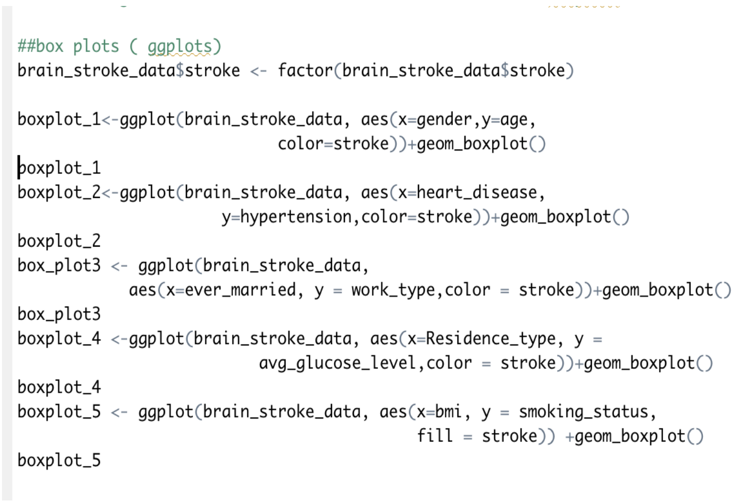
```
#distribution of brain_stroke_data by bmi
stroke_tab%>%
  ggplot()+
  geom_point(mapping = aes(x = age,y = bmi),color="Maroon")+
  labs(x= "age", y="BMI", title = "Distribution of Stroke according to BMI")+
  ylim(10,60)+
  theme_linedraw()
```

- <u>Bar chart for different age groups</u>



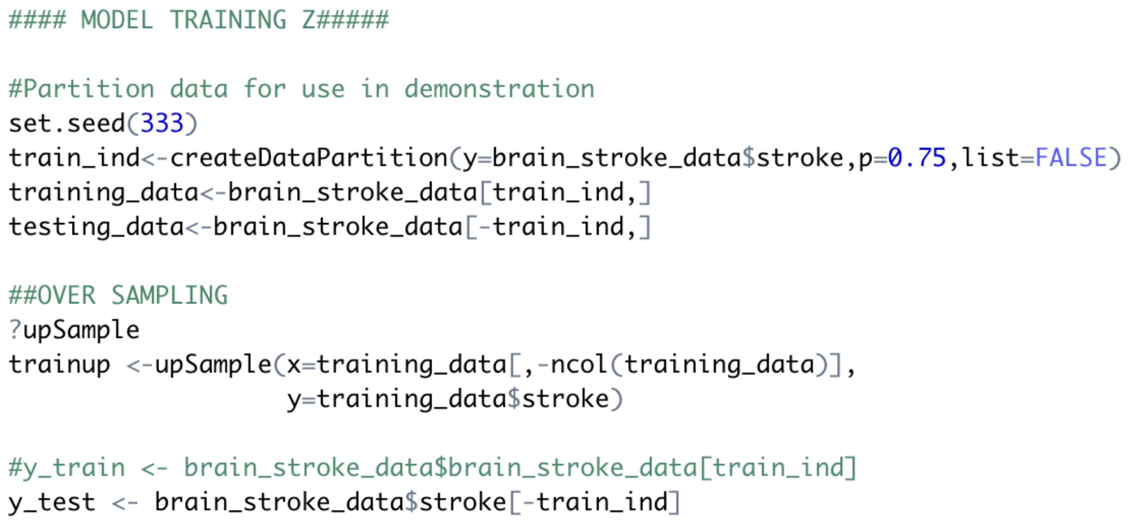
```
#distribution of brain_stroke_data by age
stroke_tab %>%

  ggplot(aes(x=age)) +
  geom_histogram(bins=10,color="Purple") +
  labs(title = "People who had stroke from different age category")+
  viridis::scale_color_viridis(discrete = TRUE)
```

- <u>Box plots with combination of two variables</u>

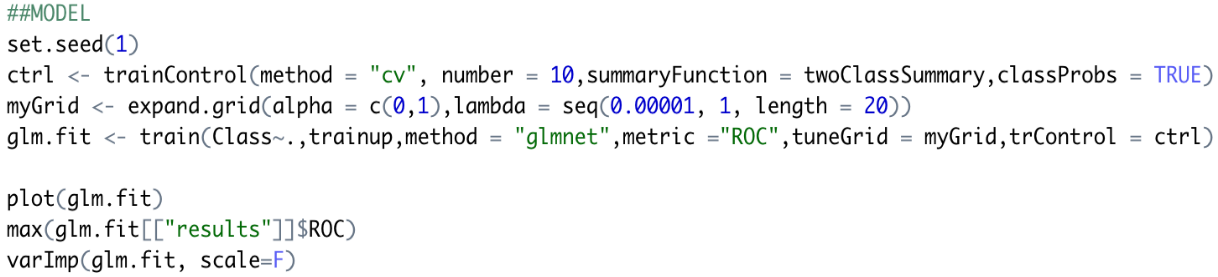


```
##box plots ( ggplots)
brain_stroke_data$stroke <- factor(brain_stroke_data$stroke)

boxplot_1<-ggplot(brain_stroke_data, aes(x=gender,y=age,
                            color=stroke))+geom_boxplot()
boxplot_1
boxplot_2<-ggplot(brain_stroke_data, aes(x=heart_disease,
                       y=hypertension,color=stroke))+geom_boxplot()
boxplot_2
box_plot3 <- ggplot(brain_stroke_data,
            aes(x=ever_married, y = work_type,color = stroke))+geom_boxplot()
box_plot3
boxplot_4 <-ggplot(brain_stroke_data, aes(x=Residence_type, y =
                          avg_glucose_level,color = stroke))+geom_boxplot()
boxplot_4
boxplot_5 <- ggplot(brain_stroke_data, aes(x=bmi, y = smoking_status,
                                          fill = stroke)) +geom_boxplot()
boxplot_5
```

4. Logistic Regression

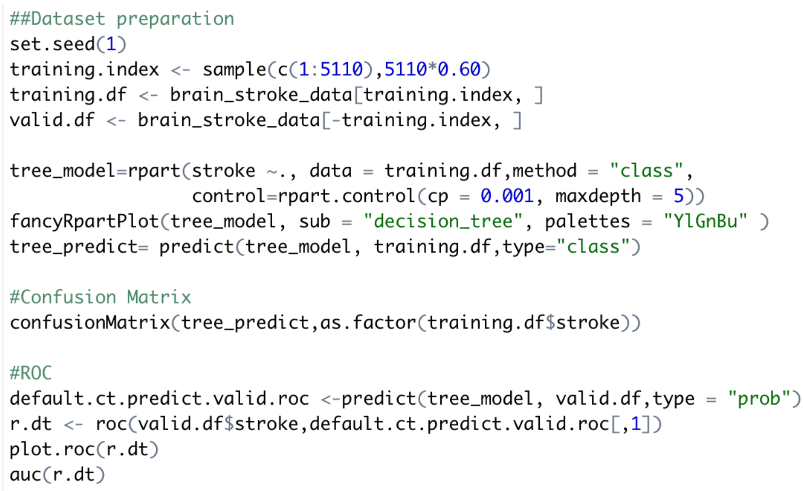


```
#### MODEL TRAINING Z#####

#Partition data for use in demonstration
set.seed(333)
train_ind<-createDataPartition(y=brain_stroke_data$stroke,p=0.75,list=FALSE)
training_data<-brain_stroke_data[train_ind,]
testing_data<-brain_stroke_data[-train_ind,]

##OVER SAMPLING
?upSample
trainup <-upSample(x=training_data[,-ncol(training_data)],
                   y=training_data$stroke)

#y_train <- brain_stroke_data$brain_stroke_data[train_ind]
y_test <- brain_stroke_data$stroke[-train_ind]
```



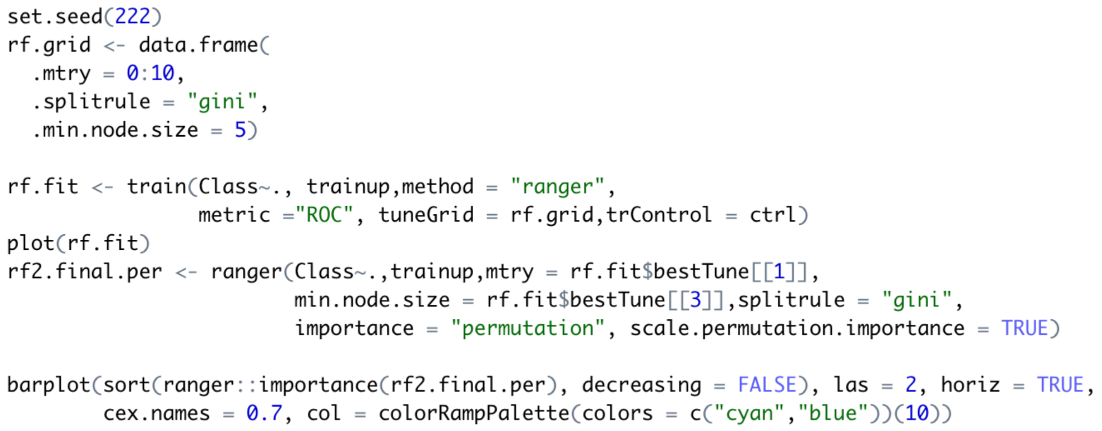
```
##MODEL
set.seed(1)
ctrl <- trainControl(method = "cv", number = 10,summaryFunction = twoClassSummary,classProbs = TRUE)
myGrid <- expand.grid(alpha = c(0,1),lambda = seq(0.00001, 1, length = 20))
glm.fit <- train(Class~.,trainup,method = "glmnet",metric ="ROC",tuneGrid = myGrid,trControl = ctrl)

plot(glm.fit)
max(glm.fit[["results"]]$ROC)
varImp(glm.fit, scale=F)
```

5. Decision Tree

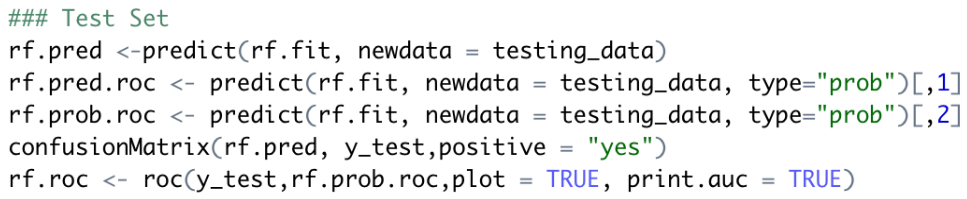


```
##Dataset preparation
set.seed(1)
training.index <- sample(c(1:5110),5110*0.60)
training.df <- brain_stroke_data[training.index, ]
valid.df <- brain_stroke_data[-training.index, ]

tree_model=rpart(stroke ~., data = training.df,method = "class",
                 control=rpart.control(cp = 0.001, maxdepth = 5))
fancyRpartPlot(tree_model, sub = "decision_tree", palettes = "YlGnBu" )
tree_predict= predict(tree_model, training.df,type="class")

#Confusion Matrix
confusionMatrix(tree_predict,as.factor(training.df$stroke))

#ROC
default.ct.predict.valid.roc <-predict(tree_model, valid.df,type = "prob")
r.dt <- roc(valid.df$stroke,default.ct.predict.valid.roc[,1])
plot.roc(r.dt)
auc(r.dt)
```

6. Random Forest



```
set.seed(222)
rf.grid <- data.frame(
  .mtry = 0:10,
  .splitrule = "gini",
  .min.node.size = 5)

rf.fit <- train(Class~., trainup,method = "ranger",
                metric ="ROC", tuneGrid = rf.grid,trControl = ctrl)
plot(rf.fit)
rf2.final.per <- ranger(Class~.,trainup,mtry = rf.fit$bestTune[[1]],
                        min.node.size = rf.fit$bestTune[[3]],splitrule = "gini",
                        importance = "permutation", scale.permutation.importance = TRUE)

barplot(sort(ranger::importance(rf2.final.per), decreasing = FALSE), las = 2, horiz = TRUE,
        cex.names = 0.7, col = colorRampPalette(colors = c("cyan","blue"))(10))
```



```
### Test Set
rf.pred <-predict(rf.fit, newdata = testing_data)
rf.pred.roc <- predict(rf.fit, newdata = testing_data, type="prob")[,1]
rf.prob.roc <- predict(rf.fit, newdata = testing_data, type="prob")[,2]
confusionMatrix(rf.pred, y_test,positive = "yes")
rf.roc <- roc(y_test,rf.prob.roc,plot = TRUE, print.auc = TRUE)
```

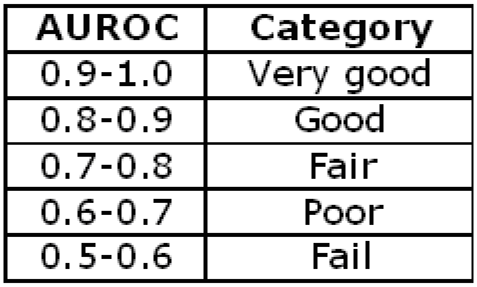
Conclusion and Implications

The dataset used for analysis here is very unbalanced. Only 249 observations out of 5110 observations are the people who suffer from stroke. This is only 4.8% of the total data. In such cases, the accuracy might not represent the analysis in a right way.

We observed different accuracy rates for different models-
Decision tree- 95.17%
Random Forest- 94.43%
Logistic Regression- 73.37%

The ROC curve is a better representation and the area under the ROC curve (AUROC) can tell us which model is the most suitable one for our dataset.

The area under the curve of the ROC plot can be used as a standard to measure the test's discriminative ability, i.e how good is the test in a given clinical situation.



AUROC	Category
0.9-1.0	Very good
0.8-0.9	Good
0.7-0.8	Fair
0.6-0.7	Poor
0.5-0.6	Fail

According to the Classification Models i.e. Decision tree, Random forest and Logistic regression we generated a comparison between all the AUC ROC values.
Decision tree-0.723
Random Forest-0.815
Logistic Regression- 0.834

So from the table above we conclude that the Decision tree generated a fair model whereas Random forest and Logistic regression performed better and were concluded to be good models for this data set.

This could be extremely helpful for the patients as they can receive intensive healthcare before their condition gets serious. Knowing whether a person will have a stroke in the future could aid in saving a lot of lives and money.

References

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

Theory-

1.https://acutecaretesting.org/en/articles/roc-curves-what-are-they-and-how-are-they-used
2.https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalanced-classification/
3.https://www.analyticsvidhya.com/blog/2016/03/practical-guide-deal-imbalanced-classification-problems/

Codes-

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

https://www.kaggle.com/aditimulye/stroke-prediction-visualization-prediction

https://www.kaggle.com/adityasharma2812/eda-stroke-predection

https://stackoverflow.com/questions/38250440/error-in-na-fail-default-missing-values-in-object-but-no-missing-values

https://neptune.ai/blog/f1-score-accuracy-roc-auc-pr-auc#:~:text=The%20first%20big%20difference%20is,assigned%20positive%20and%20negative%20classes.