Stroke Prediction

Habib Khalil

2022-12-22

Introduction

The dataset set we will analyse in this project is to find out what factor increases the chances of having a stroke if any. The data set has 11 clinical features(columns), I will use these parameters and see if there is any correlation between pre-existing health factors and stroke and secondly if there are environmental factors affecting the chances of having a stroke. The data set is already collected and is somewhat pre-processed with some unknown values in the smoking and na values BMI column, which will be dealt with in the data processing phase of the project as mentioned above it has 11 clinical features(columns). The data set is comprised of slightly more females than males with 5110 observations. The reason we have chosen this data set is that stroke kills around 11% of the world population making it the second cause of death. The questions we are going to answer are as follows: Does any factor in the data set affect the chances of getting a stroke? We will divide this question into 3 sub-questions stated below and group variables together on similarities. which will be addressed by using observations in mentioned clinical features(columns) during statistical analysis.

1. Does underlying health conditions increase or decrease chances of getting a stroke? By using data from following columns.

a. Hypertension
b. Heart disease
c. Average glucose level
d. BMI

2. Does social/environmental status increase or decrease chances of getting a stroke? By using data from following columns.

a. Residence Type
b. Work type
c. Ever married
d. Smoking Status

3. Does human biology affect chances of getting a stroke? By using data from following columns.

a. Age
b. Gender Null Hypothesis: No variables in the data set has an influence on chances of getting a stroke. Alternate Hypothesis: Variables in the data set does have an influence on chances of getting a stroke.

Data set was obtained from Kaggle.com, the link is provided, but the auther wants to remain anonymous and has released the datsset for educational purposes. https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset (https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset)

```
suppressMessages(library(tidyverse))
```

```
## Warning: package 'ggplot2' was built under R version 4.2.2
```

```
## Warning: package 'readr' was built under R version 4.2.2
```

```
## Warning: package 'purrr' was built under R version 4.2.2
```

```
## Warning: package 'stringr' was built under R version 4.2.2
```

```
suppressMessages(library(naniar))
```

```
## Warning: package 'naniar' was built under R version 4.2.2
```

```
suppressMessages(library(ggplot2))
suppressMessages(library(grid))
suppressMessages(library(forcats))
suppressMessages(library(gridExtra))
```

```
## Warning: package 'gridExtra' was built under R version 4.2.2
```

```
suppressMessages(library(dplyr))
suppressMessages(library(tidyr))
suppressMessages(library(scales))
suppressMessages(library(caret))
```

```
## Warning: package 'caret' was built under R version 4.2.2
```

```
suppressMessages(library(MLmetrics))
```

```
## Warning: package 'MLmetrics' was built under R version 4.2.2
```

```
suppressMessages(library(imbalance))
```

```
## Warning: package 'imbalance' was built under R version 4.2.2
```

```
suppressMessages(library(gridExtra))
suppressMessages(library(patchwork))
```

```
## Warning: package 'patchwork' was built under R version 4.2.2
```

Loading the data and data exploration

data set has been downloaded form the website in a CSV format and using read.csv data is loaded on to R.

```
#install.packages("RCurl")
#library(RCurl)
#dataset_url <- "https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset"
#destfile <- "stroke-prediction-dataset.csv"
#download.file(dataset_url, destfile = destfile)

#multiple attempts were made to download directly from the website but due to some issue, I w
as unable to succeed in my attempts  therefore data was downloaded manually and accessed as b
elow.

df <- read.csv("healthcare-dataset-stroke-data.csv")

# we will take a glimpse of the data set to get a general idea of the what sort of data we ar
e dealing with.
glimpse(df)
```

```
## Rows: 5,110
## Columns: 12
## $ id                <int> 9046, 51676, 31112, 60182, 1665, 56669, 53882, 10434…
## $ gender            <chr> "Male", "Female", "Male", "Female", "Female", "Male"…
## $ age               <dbl> 67, 61, 80, 49, 79, 81, 74, 69, 59, 78, 81, 61, 54, …
## $ hypertension      <int> 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1…
## $ heart_disease     <int> 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0…
## $ ever_married      <chr> "Yes", "Yes", "Yes", "Yes", "Yes", "Yes", "Yes", "No…
## $ work_type         <chr> "Private", "Self-employed", "Private", "Private", "S…
## $ Residence_type    <chr> "Urban", "Rural", "Rural", "Urban", "Rural", "Urban"…
## $ avg_glucose_level <dbl> 228.69, 202.21, 105.92, 171.23, 174.12, 186.21, 70.0…
## $ bmi               <chr> "36.6", "N/A", "32.5", "34.4", "24", "29", "27.4", "…
## $ smoking_status    <chr> "formerly smoked", "never smoked", "never smoked", "…
## $ stroke            <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1…
```

There are 12 columns 5110 rows, we have both categorical and continuous variables with mixture of character and numeric atomic classes of object. we will attempt to change these classes for ease of use.

```
#factoring categorical variables for ease of access and uniformity, useful for modelling func
tions of aov, lm, etc.

df$gender<-as.factor(df$gender)
df$ever_married<-as.factor(df$ever_married)
df$work_type<- factor(df$work_type)
df$Residence_type<- factor(df$Residence_type)
df$smoking_status <- factor(df$smoking_status)
df$heart_disease <- as.factor(df$heart_disease)
df$hypertension <- as.factor(df$hypertension)
df$stroke <- as.factor(df$stroke)


#as bmi is a continuous variables and stored as character, it will be change appropriately.
df$bmi <- as.numeric(df$bmi)
```

```
## Warning: NAs introduced by coercion
```

Categorical and numerical variable are present in the dataset as following: Categorical variables are gender, ever_married, work_type, Residence_type, and smoking_status. Variables with binary numerical as in 1(Yes) and 0(No) are present in hypertension,heart_disease, and stroke. Variables with continuous numerical class are age, avg_glucose_level and bmi.

Finding missing values and cleaning data. summarizing and understanding data variables including the category within the variables.

using summary function for overview of the data set, categories of categorical variables with its count and mean, median etc for numeric variables will be displayed.

```
summary(df)
```

```
##        id            gender          age        hypertension heart_disease
##  Min.   :   67   Female:2994   Min.   : 0.08   0:4612       0:4834
##  1st Qu.:17741   Male  :2115   1st Qu.:25.00   1: 498       1: 276
##  Median :36932   Other :   1   Median :45.00
##  Mean   :36518                 Mean   :43.23
##  3rd Qu.:54682                 3rd Qu.:61.00
##  Max.   :72940                 Max.   :82.00
##
##  ever_married         work_type    Residence_type avg_glucose_level
##  No :1757     children     : 687   Rural:2514     Min.   : 55.12
##  Yes:3353     Govt_job     : 657   Urban:2596     1st Qu.: 77.25
##               Never_worked :  22                  Median : 91.89
##               Private      :2925                  Mean   :106.15
##               Self-employed: 819                  3rd Qu.:114.09
##                                                   Max.   :271.74
##
##       bmi                smoking_status stroke
##  Min.   :10.30   formerly smoked: 885   0:4861
##  1st Qu.:23.50   never smoked   :1892   1: 249
##  Median :28.10   smokes         : 789
##  Mean   :28.89   Unknown        :1544
##  3rd Qu.:33.10
##  Max.   :97.60
##  NA's   :201
```

From the summary we can observe 3 possible issues, first one is in gender column as we have one patient with “Other” gender category, this needs to be removed as one record will not have any affect on the analysis. second possible issue can be found in smoking status column titled as “Unknown”, making it a significant percentage of the data set, we will explore this at later stage and decision will be taken after that, and the third possible issue is in bmi column with 201 NA values and possible solution is to replace NA with mean bmi as it is small percentage of the whole data set.

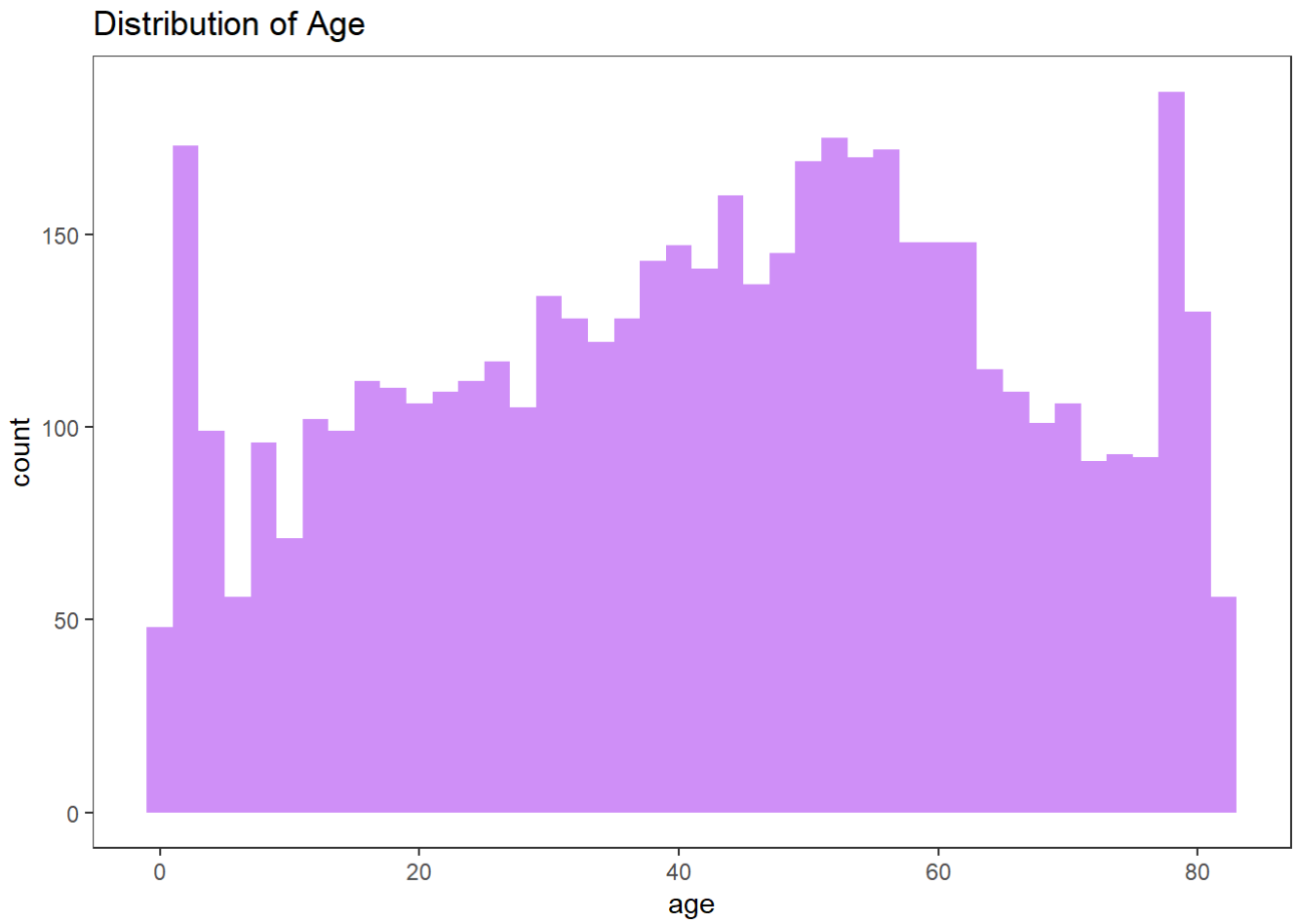
We will now look at one variable at a time and deal with any missing, outliers and to have better understanding, we will start with continuous varibles.

Variable id: please note id column has no affect on the analysis therefore it will be dropped.

```
# id column is dropped

df$id <-NULL
```

Variable: Age

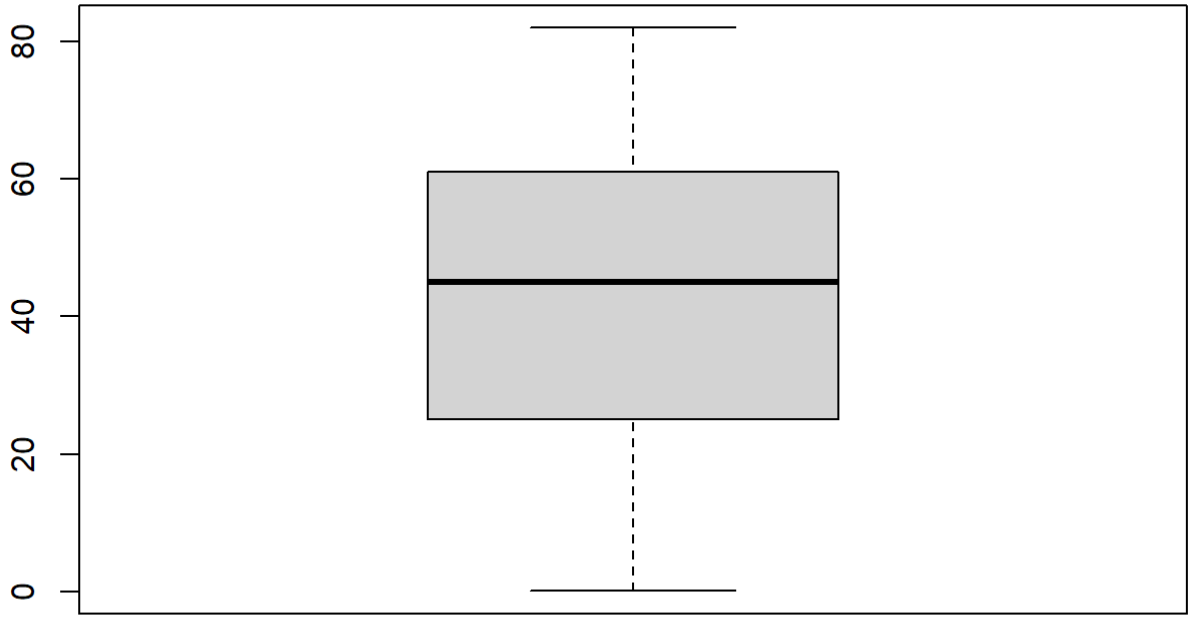
```
ggplot(df, aes(x = age)) +
geom_histogram(fill = 'purple',alpha = 0.5, binwidth = 2) +
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
labs(title = "Distribution of Age")
```




```
summary(df$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.08   25.00   45.00   43.23   61.00   82.00
```

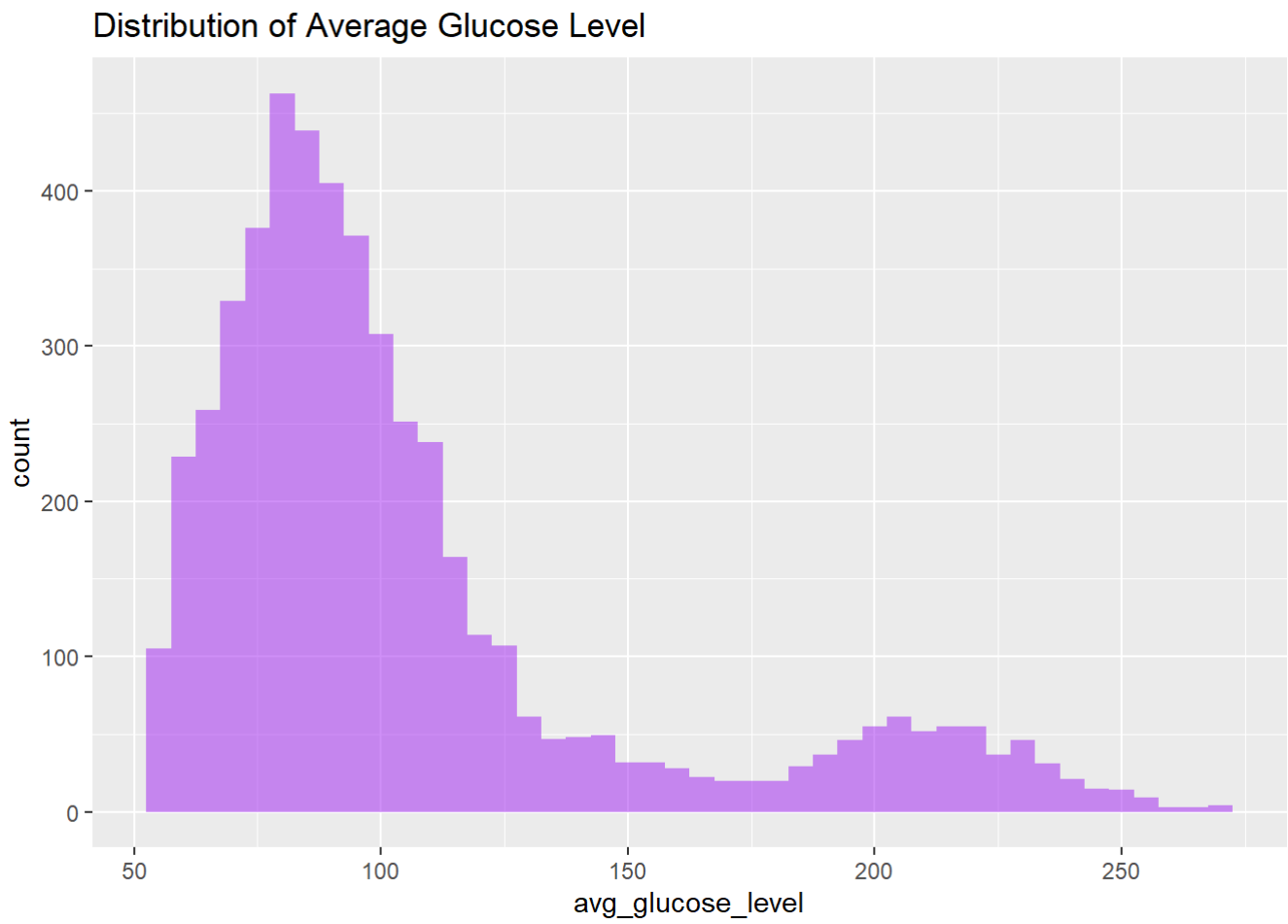
```
boxplot(df$age)
```




```
#The age range is quite good, patient are of all ages and mean age is 43.22 years old althoug
h general knowledge is that stroke is related to older age. and there are no outliers and goo
d inter-quratile range.
```

Variable: Average glucose level

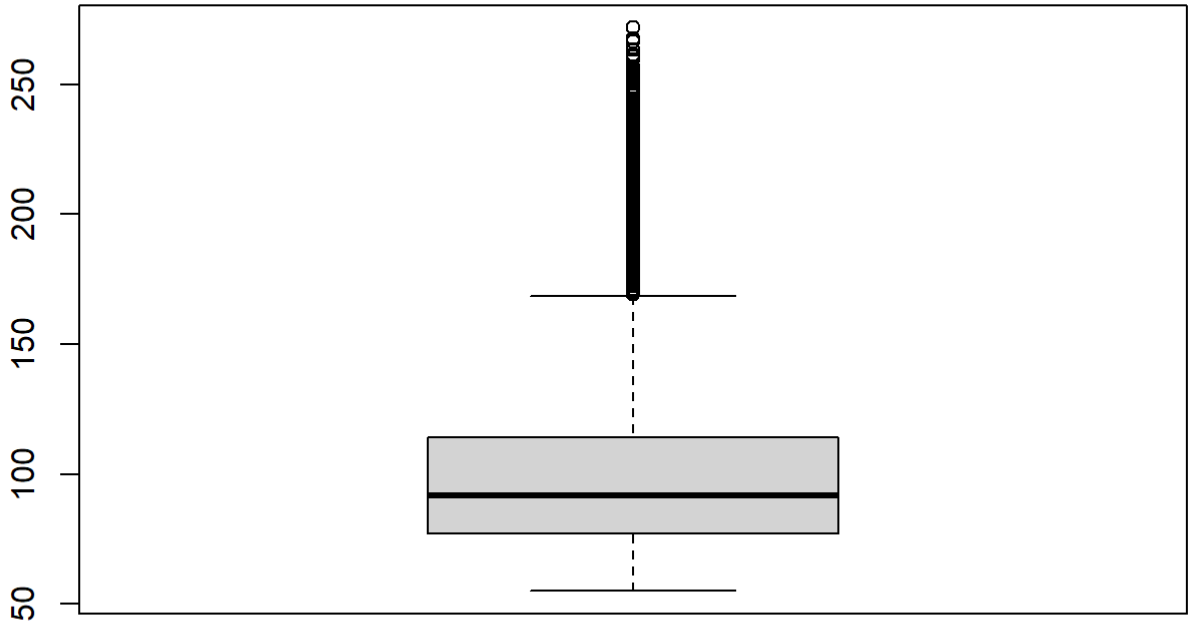
```
ggplot(df, aes(x = avg_glucose_level)) +
geom_histogram(binwidth = 5, fill = 'purple',alpha = 0.5) +
labs(title = "Distribution of Average Glucose Level")
```



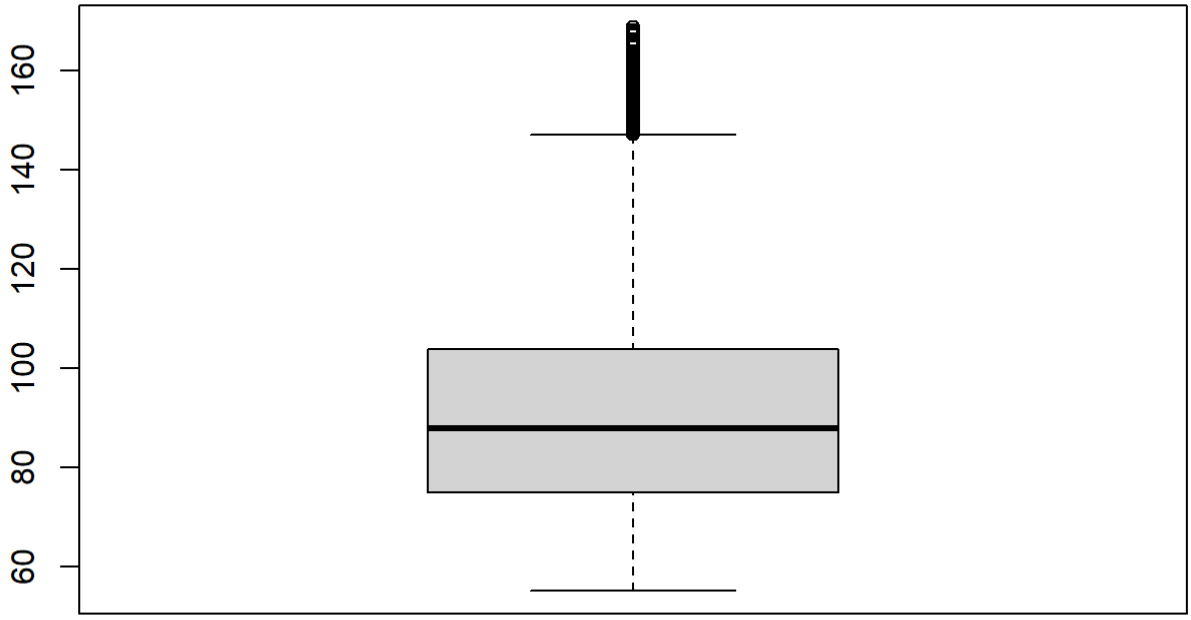

```
#we have right skewed data distribution

summary(df$avg_glucose_level)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   55.12   77.25   91.89  106.15  114.09  271.74
```

```
#difference between 3rd quartile and max is really big, max value is almost 2.5 time the 3rd quartile, we will visualize this in box plot

boxplot(df$avg_glucose_level)
```




```
# as we can see these can be consider outliers and needs to be removed for sake of this proje
ct,

#Removing any outliers
quartiles<- quantile(df$avg_glucose_level, probs= c(0.25,0.75), na.rm = FALSE)
IQR <- IQR(df$avg_glucose_level)
Lower <- quartiles[1] - 1.5*IQR
Upper <- quartiles[2] + 1.5*IQR
df <- subset(df, df$avg_glucose_level > Lower & df$avg_glucose_level < Upper)


boxplot(df$avg_glucose_level)
```



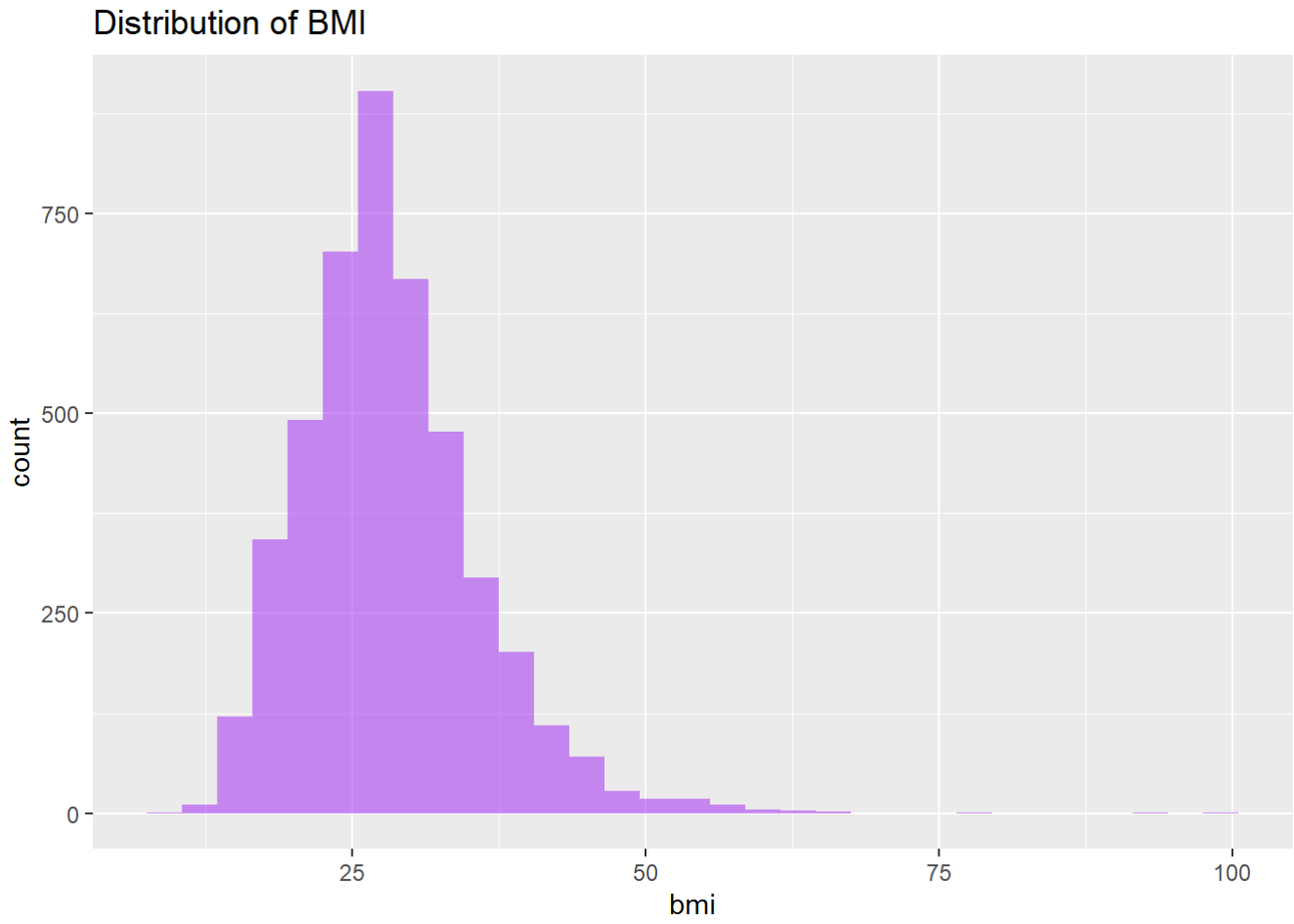

```
#as we can see now it is the difference between the max and 3rd quartile is not to high.
summary(df$avg_glucose_level)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   55.12   75.06   87.96   91.44  103.93  168.68
```

```
# the mean has dropped to 91.44 mg/dL and max value has dropped by almost a 100mg/dL
```

Variable: BMI

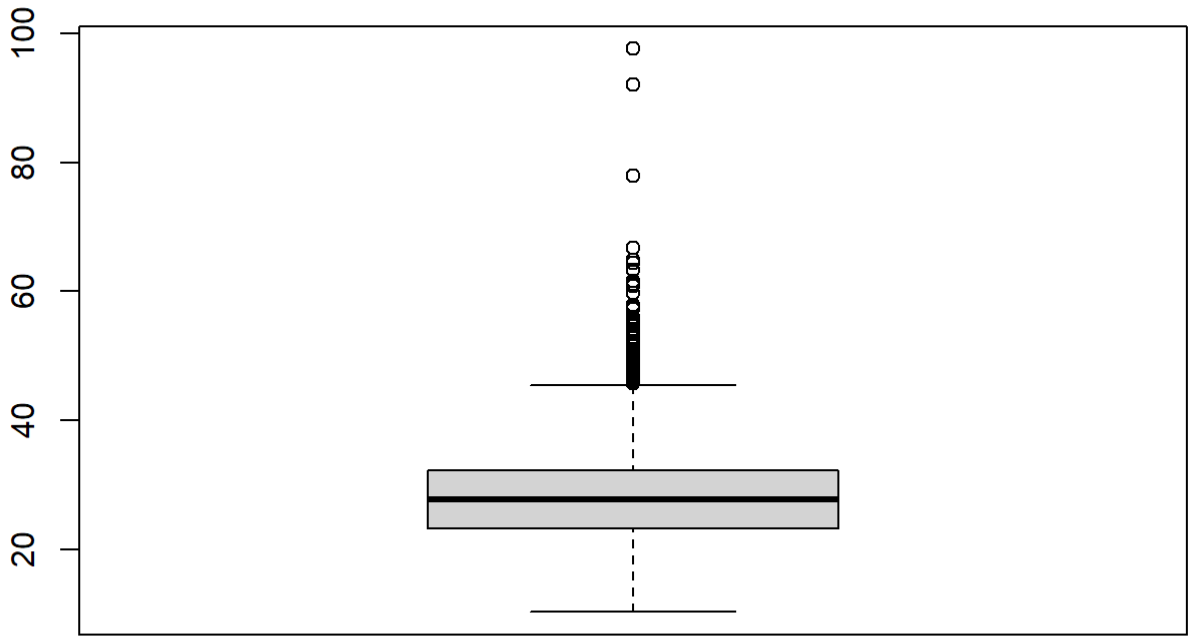
```
# 201 n/a in the BMI column. we will deal with N/A values of bmi by replacing it with mean bm
i from the dataset.
df$bmi[is.na(df$bmi)] <- round(mean(df$bmi, na.rm = TRUE))

ggplot(df, aes(x = bmi)) +
geom_histogram( binwidth = 3, fill = 'purple',alpha = 0.5) +
labs(title = "Distribution of BMI")
```



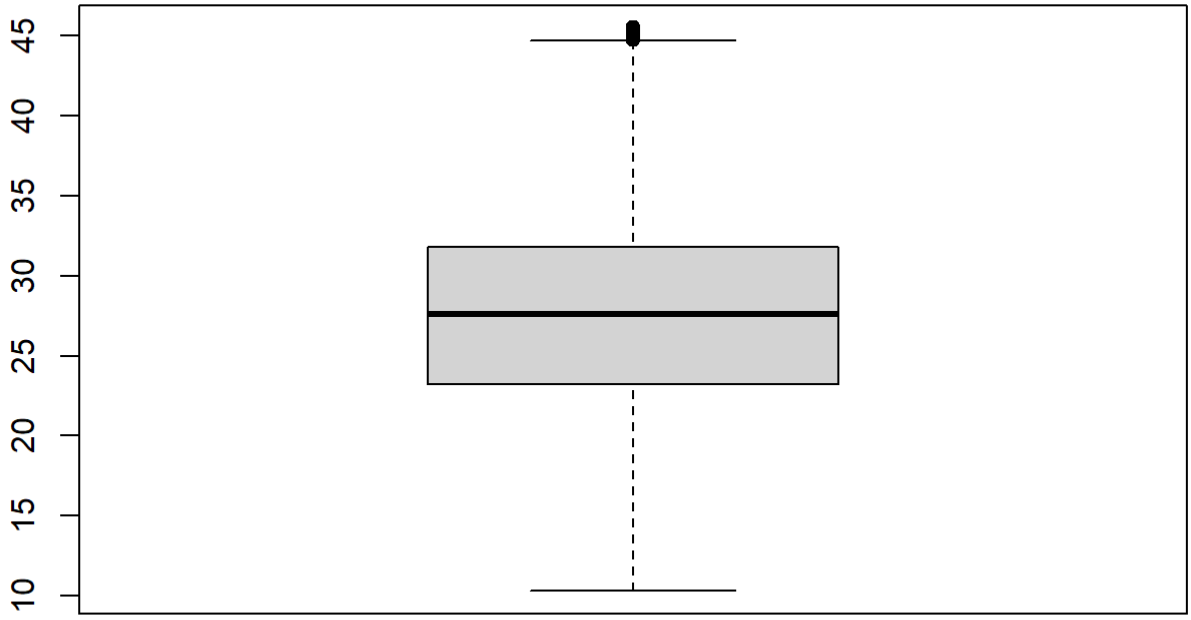

```
# the data is skewed to the right as the peak of the graph lies on the left side.

summary(df$bmi)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    10.3    23.3    27.8    28.3    32.2    97.6
```

```
#we have extreme obesity for small percentage of patients, but mean bmi being 28.9.

boxplot(df$bmi)
```




```
#the difference between max and 3rd quartile is over 65BMI, which really big, we will also re
move the outliers for the purpose of this project



#Removing any outliers
quartiles<- quantile(df$bmi, probs= c(0.25,0.75), na.rm = FALSE)
IQR <- IQR(df$bmi)
Lower <- quartiles[1] - 1.5*IQR
Upper <- quartiles[2] + 1.5*IQR
df <- subset(df, df$bmi > Lower & df$bmi < Upper)

boxplot(df$bmi)
```



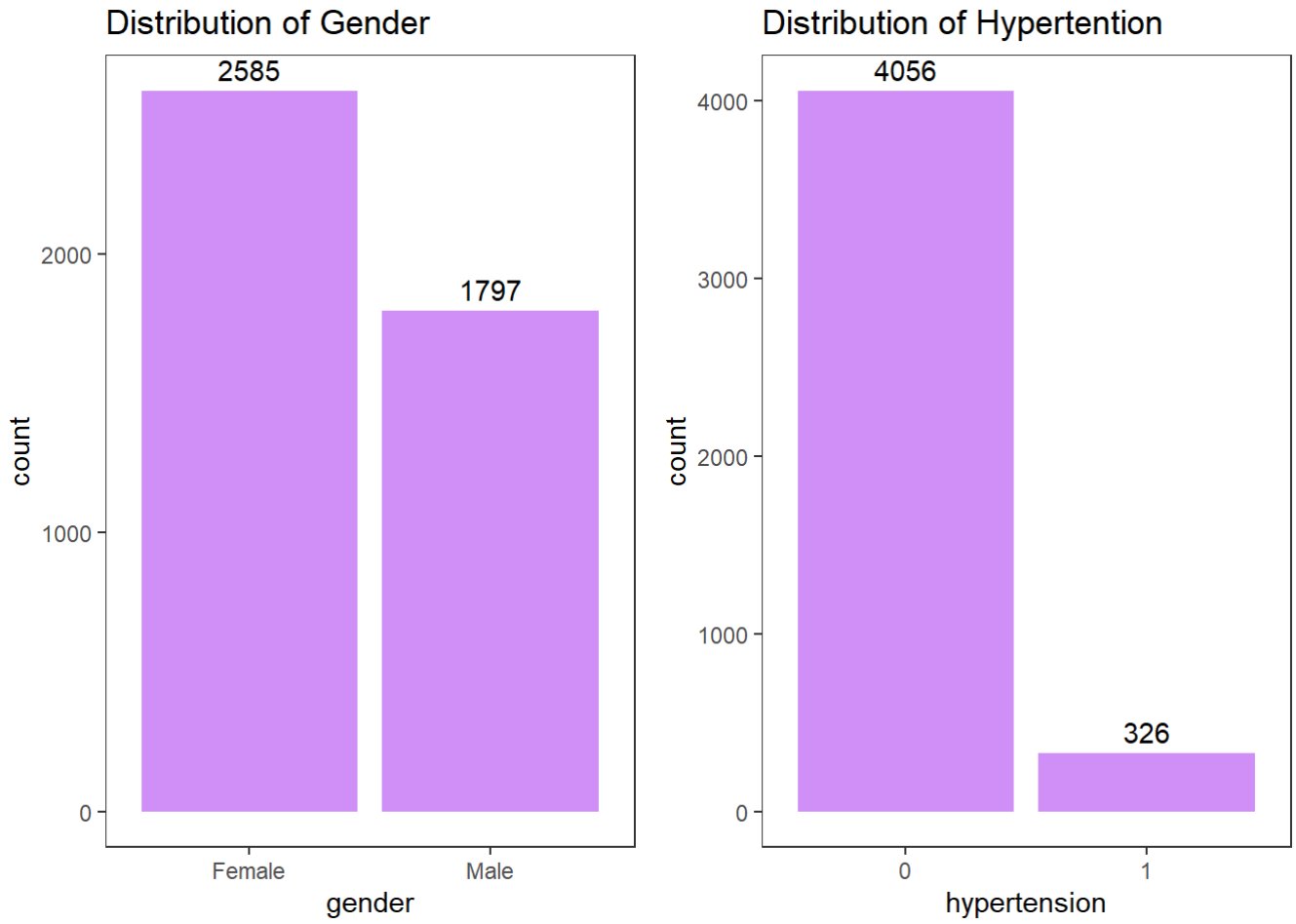

```
summary(df$bmi)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.30   23.20   27.60   27.74   31.80   45.50
```

```
#we have now alot better bmi readings than before, the diffrence between max and 3rd quartile is only 12bmi come to 65bmi. please refer to summary and boxplot.
```

Variable: Gender and hypertension

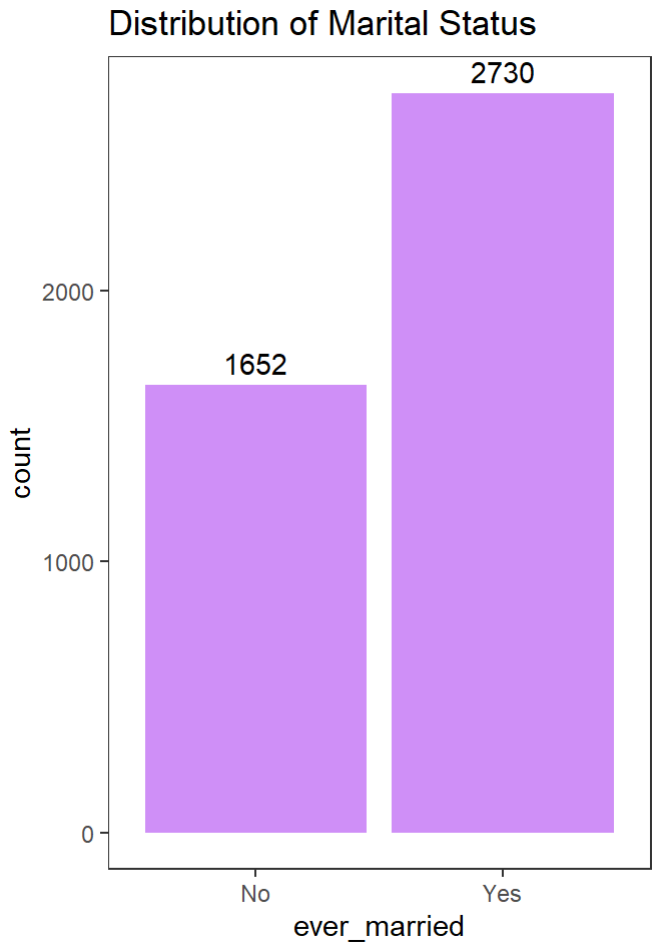
```r
#there is 3 categories in the gender column and the category "Other" has only one data, there
fore we will drop the "Other" category in the gender column.
df  <- df[df$gender != "Other", , drop=FALSE]
df$gender <- factor(df$gender)


#plotting of the variable to visualize disparity if any

gen<-df%>%
  ggplot(aes(gender))+
  geom_bar(fill = 'purple',alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Distribution of Gender")



hyp<-df%>%
  ggplot(aes(hypertension))+
  geom_bar(fill = 'purple',alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Distribution of Hypertention")



grid.arrange(gen, hyp, ncol = 2, nrow = 1)
```

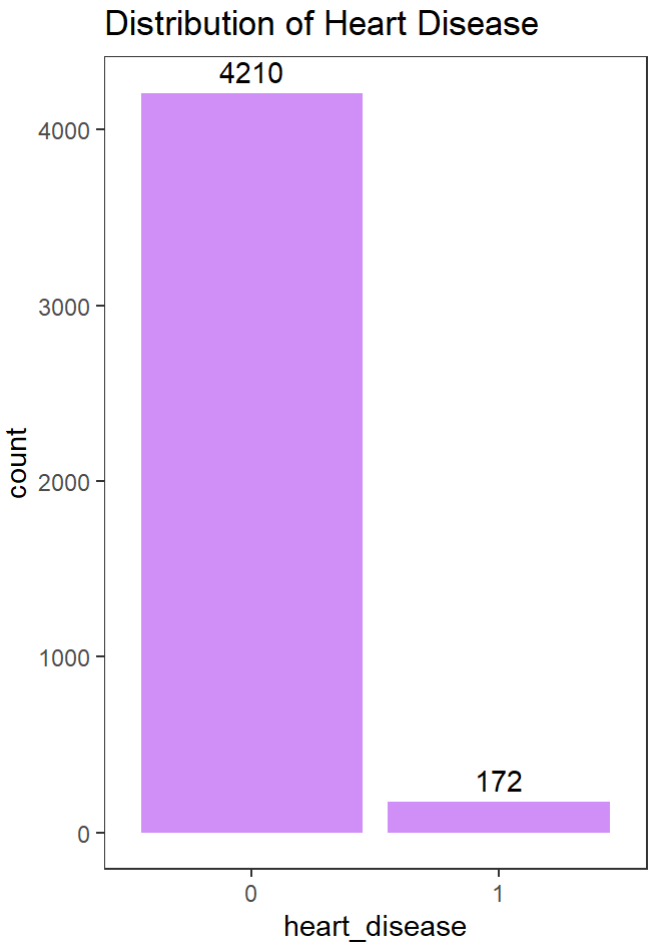



```
#there are almost 800 more female than male.
# around 7 percent of the patients have hypertension.
```

Variables: Heart disease and Marital Status

```
mar<-df%>%
  mutate(ever_married = fct_infreq(ever_married)) %>%
  mutate(ever_married = fct_rev(ever_married)) %>%
  ggplot(aes(ever_married))+
  geom_bar(fill = 'purple',alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Distribution of Marital Status")


hear<-df%>%
  ggplot(aes(heart_disease))+
  geom_bar(fill = 'purple',alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Distribution of Heart Disease")

grid.arrange(mar, hear, ncol = 2, nrow = 1)
```

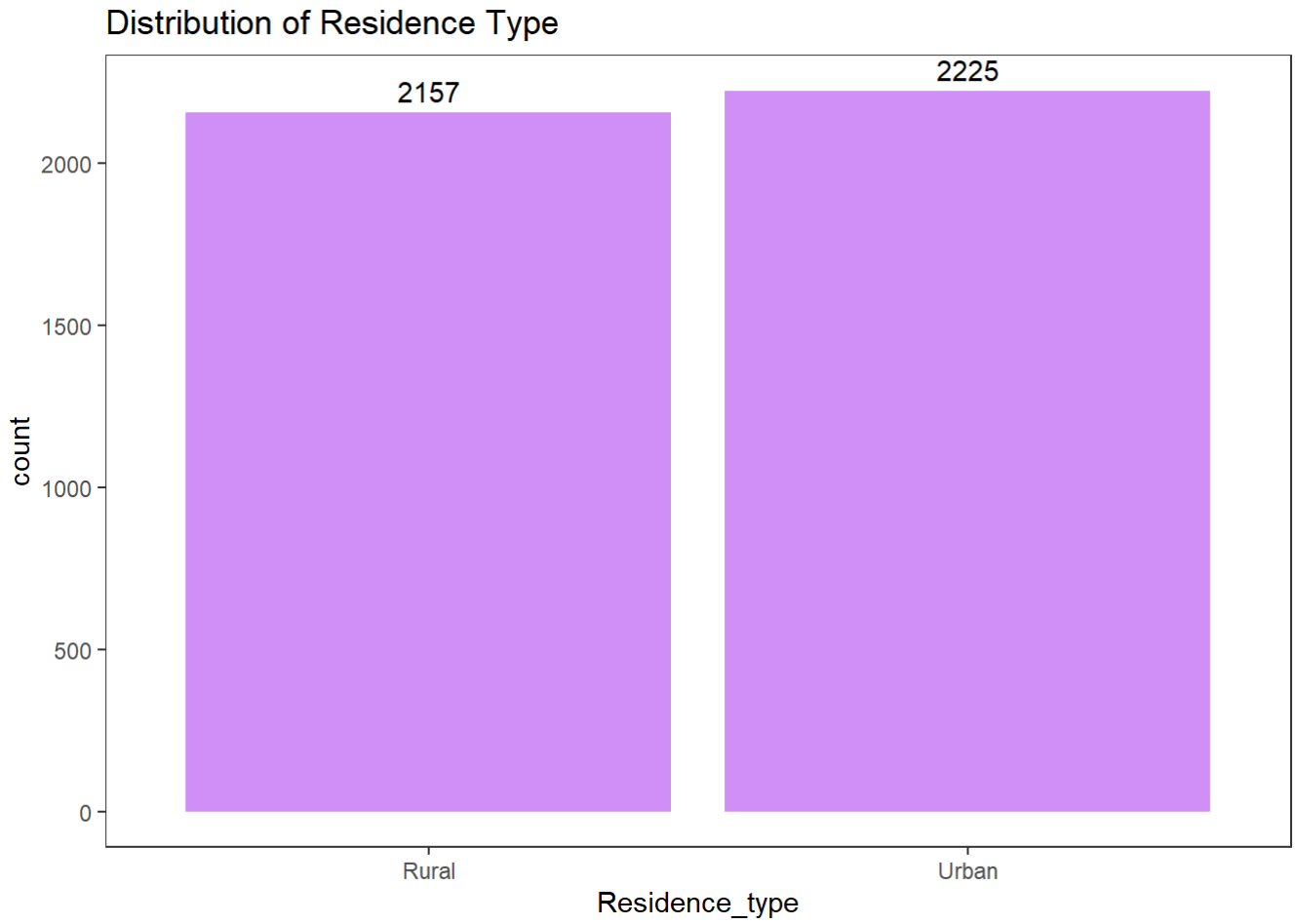




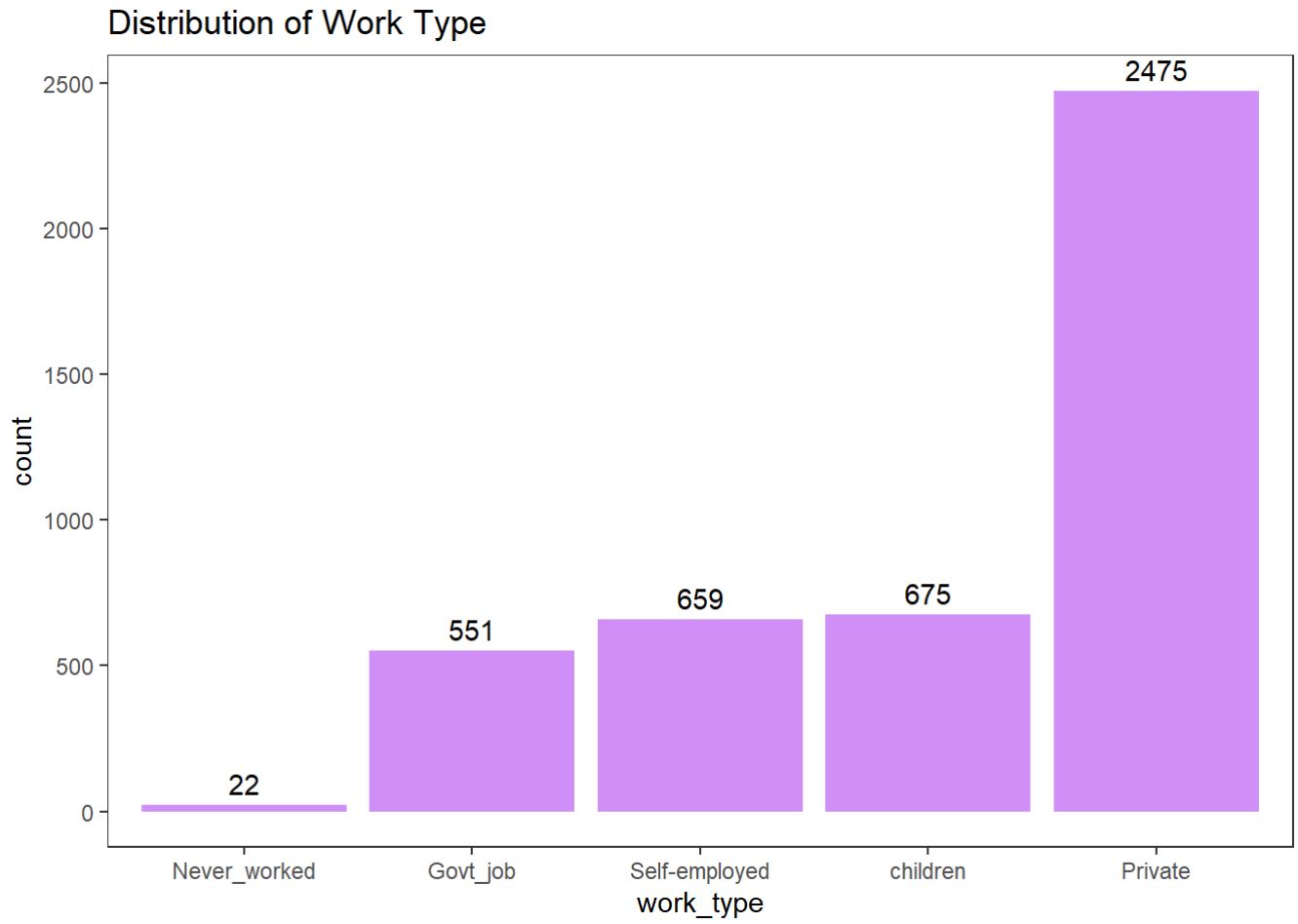

```
# around 62 percent of the patients have been married or are married
# around 4 percent of the patients have heart disease
```

Variable: Work type and Residence type

```
df%>%
  ggplot(aes(Residence_type))+
  geom_bar(fill = 'purple',alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Distribution of Residence Type")
```




```
 # the distribution of patients living in rural and urban are almost identical.

df%>%
  mutate(work_type = fct_infreq(work_type)) %>%
  mutate(work_type = fct_rev(work_type)) %>%
  ggplot(aes(work_type))+
  geom_bar(fill = 'purple',alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Distribution of Work Type")
```

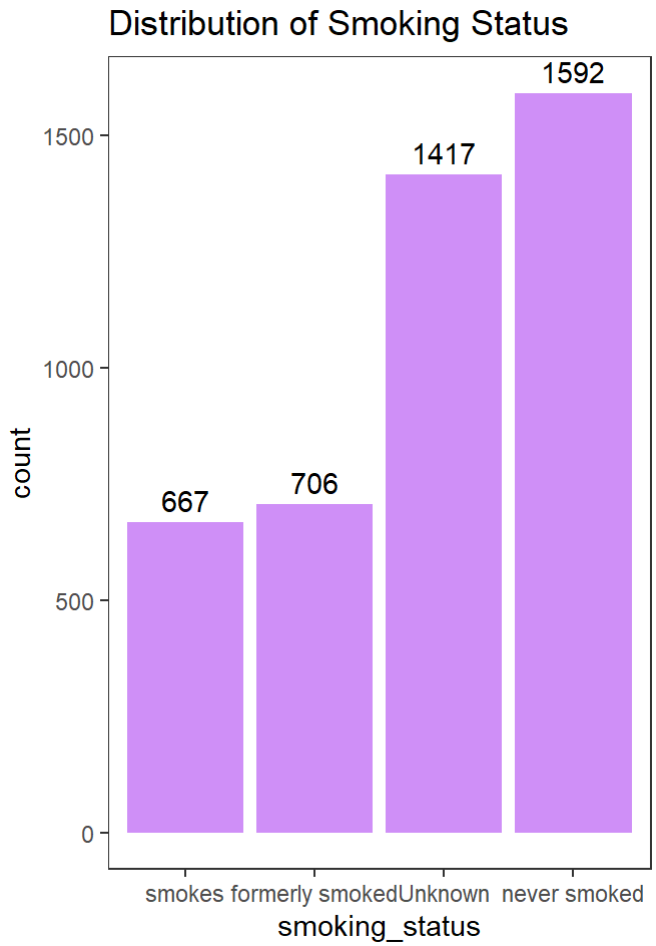


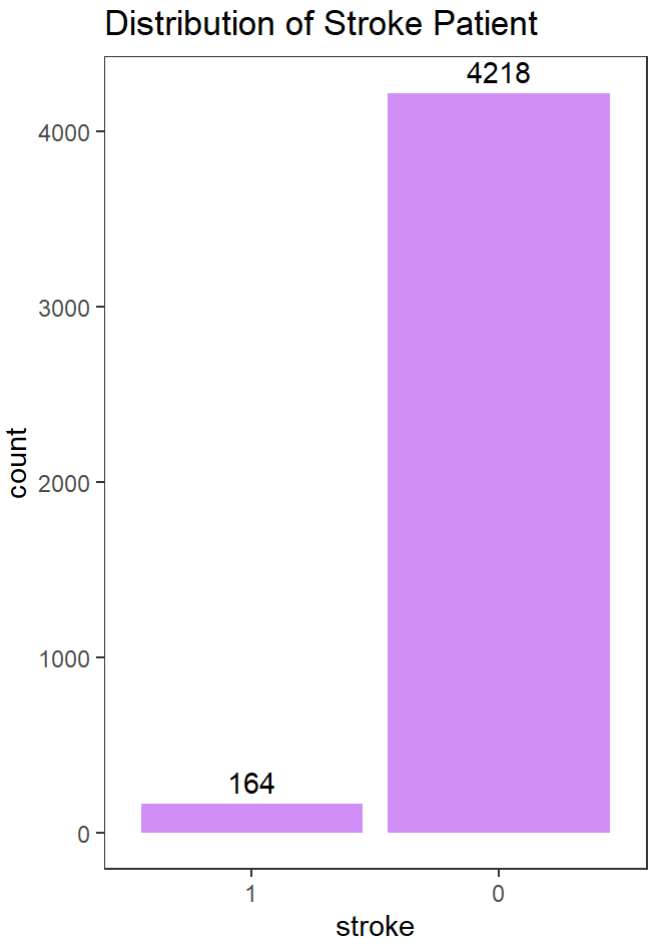

```
 # over 60% of the patients have worked in private sector, 17% self employed, 14% government job, and the rest are either children or never worked
```

Variable: Smoking status and target variable stroke

```
smok<-df%>%
  mutate(smoking_status = fct_infreq(smoking_status)) %>%
  mutate(smoking_status = fct_rev(smoking_status)) %>%
  ggplot(aes(smoking_status))+
  geom_bar(fill = 'purple',alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Distribution of Smoking Status")




stro<-df%>%
  mutate(stroke = fct_infreq(stroke)) %>%
  mutate(stroke = fct_rev(stroke)) %>%
  ggplot(aes(stroke))+
  geom_bar(fill = 'purple',alpha = 0.5)+
  geom_text(stat='count', aes(label=..count..), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Distribution of Stroke Patient")




grid.arrange(smok, stro, ncol = 2, nrow = 1)
```

```
## Warning: The dot-dot notation (`..count..`) was deprecated in ggplot2 3.4.0.
## ℹ Please use `after_stat(count)` instead.
```





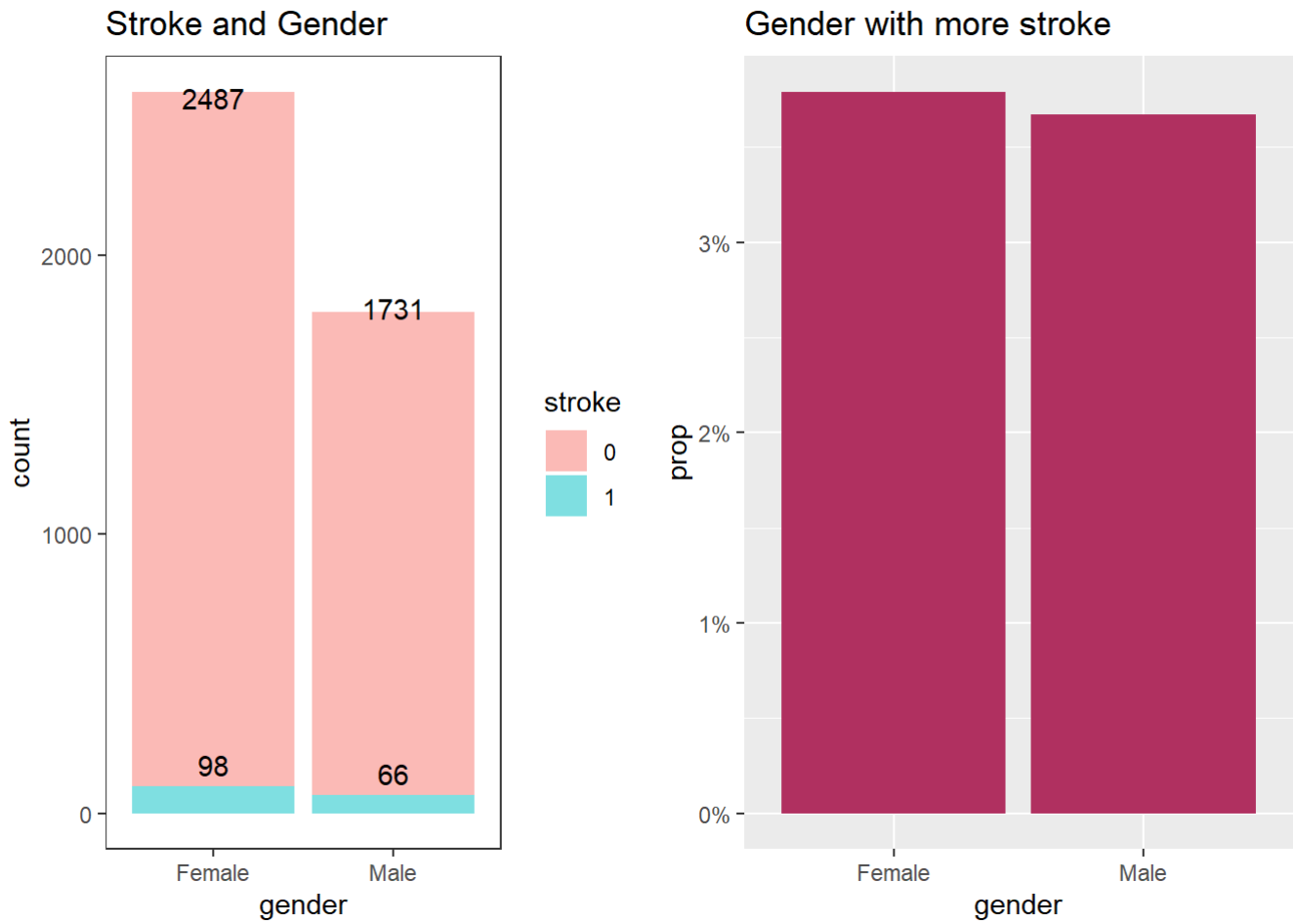

```
# 30% of patient's smoking status is unknown, this can cause an issue as smoking increases th
e chances of stroke by 6 times. 15% has smokes,17% formerly smoked and and last 37% never smo
ked. note: unknown category will be looked at later stage
#only about 5% of the patients have had stroke
```

Bivariate Analysis

We will now examine each variable with the target variable and also explore what category has more stroke patients using prop function to have a better understanding of the data and target variables, and use t-test for continuous variable and chi squared for categorical variable finding out individual impact on the target varible.

Gender vs Stroke

```r
#creating visual bar chart of both stroke and gender.
V1 <- df%>%
  ggplot(aes(gender,fill = stroke)) +
  geom_bar(alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Stroke and Gender", X = "Stroke", Y = "Count")


#creating visual bar chart proportion of gender that has highest stroke case.
gen <- df%>%
  group_by(gender)%>%
  summarise(prop = sum(stroke == "1")/length(gender))
V1p <- gen%>%
  ggplot(aes(x =gender, y = prop))+
  geom_col(fill = "maroon")+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "Gender with more stroke ")

grid.arrange(V1, V1p, ncol = 2, nrow = 1)
```



```
#141 female had stroke and 108 male, as we previously mentioned that there are around 900 mor
e female than male in the data set, in proportion chart we can observe there are slightly mor
e female who had stroke than male.



chisq.test(df$gender,df$stroke, correct = FALSE)
```

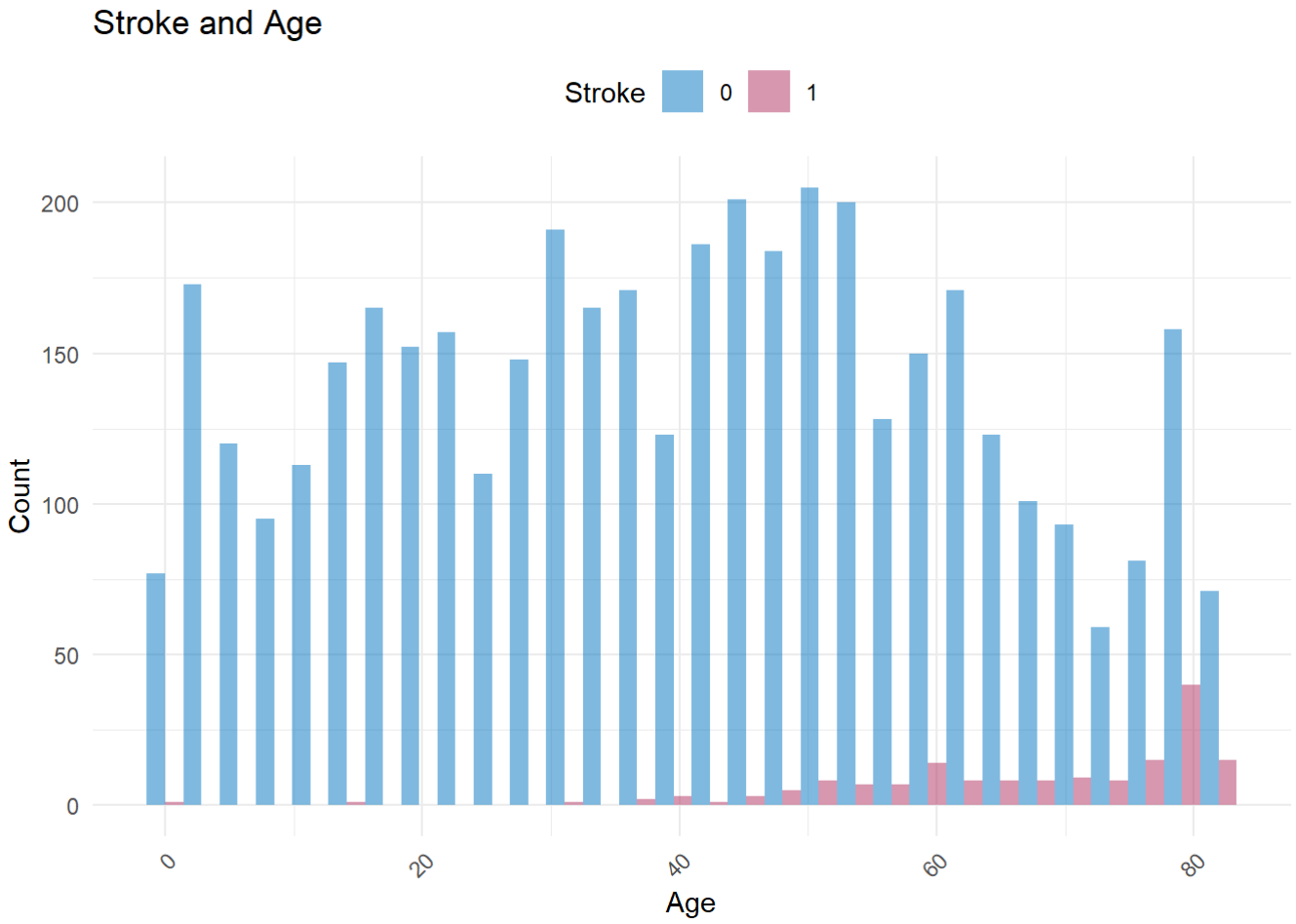
```
##
##  Pearson's Chi-squared test
##
## data:  df$gender and df$stroke
## X-squared = 0.041191, df = 1, p-value = 0.8392
```

```
# x-squared = 0.42127 indicated that there is no association between gender and stroke. the
p.value being over 0.5 and greater than 0.05.
```

Age vs Stroke

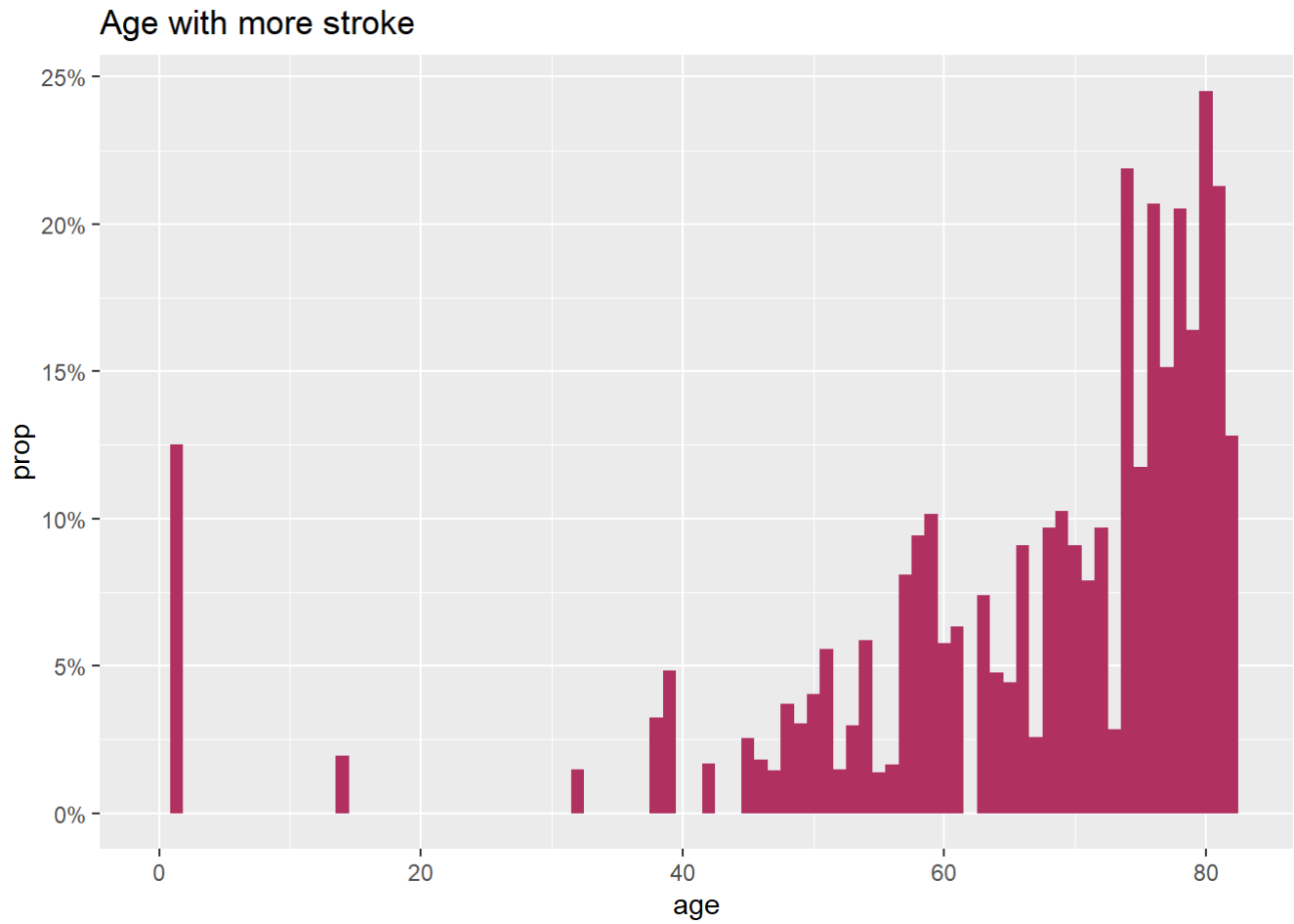
```
#creating visual bar chart of both stroke and age.
df %>%
  ggplot(aes(age)) +
  geom_histogram(aes(fill = stroke), alpha = 0.5, position = "dodge") +
  scale_fill_manual(values = c("#0073C2FF", "maroon")) +
  theme_minimal() +
  theme(legend.position = "top",
        axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Stroke and Age",
       x = "Age",
       y = "Count",
       fill = "Stroke")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#creating visual bar chart proportion of age that has highest stroke cases.
age12<-df%>%
  group_by(age)%>%
  summarise(prop = sum(stroke == "1")/length(age))
age12%>%
  ggplot(aes(x =age, y = prop))+
  geom_col(fill = "maroon", width = 1)+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "Age with more stroke ")
```

```
## Warning: `position_stack()` requires non-overlapping x intervals
```



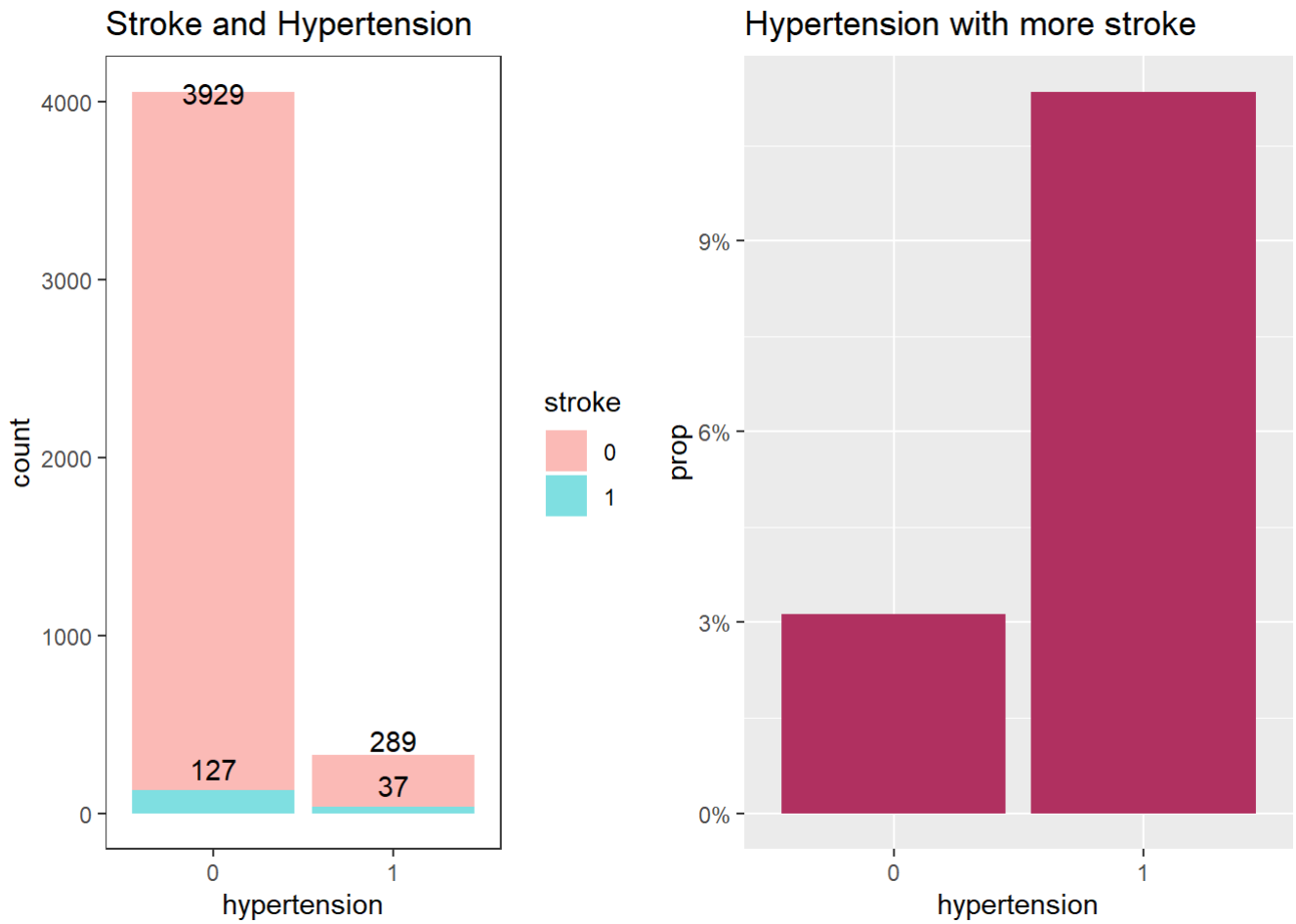

```
#there is a clear correlation between age and stroke, the older the patient gets the higher t
he chances of having a stroke specially from age 40 and over.



t.test(age~stroke, data= df)
```

```
##
##  Welch Two Sample t-test
##
## data:  age by stroke
## t = -23.596, df = 196.39, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal
to 0
## 95 percent confidence interval:
##  -29.31090 -24.78928
## sample estimates:
## mean in group 0 mean in group 1
##        39.88479        66.93488
```

```
#The mean age of individuals who have had a stroke is found to be significantly higher, with
a sample mean of 66.93488, compared to those who have not had a stroke, with a sample mean of
39.88479. this proves that age has significant impact on chances of getting a stroke
```

Hypertensionn vs Stroke

```
#creating visual bar chart of both stroke and hypertension.
H1<-df%>%
ggplot(aes(hypertension,fill = stroke)) +
  geom_bar(alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Stroke and Hypertension", X = "Stroke", Y = "Count")


#creating visual bar chart proportion of hypertension that has highest stroke case.
hyperten <- df%>%
  group_by(hypertension)%>%
  summarise(prop = sum(stroke == "1")/length(hypertension))
H1p<-hyperten%>%
  ggplot(aes(x =hypertension, y = prop))+
  geom_col(fill = "maroon")+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "Hypertension with more stroke ")

grid.arrange(H1, H1p, ncol = 2, nrow = 1)
```



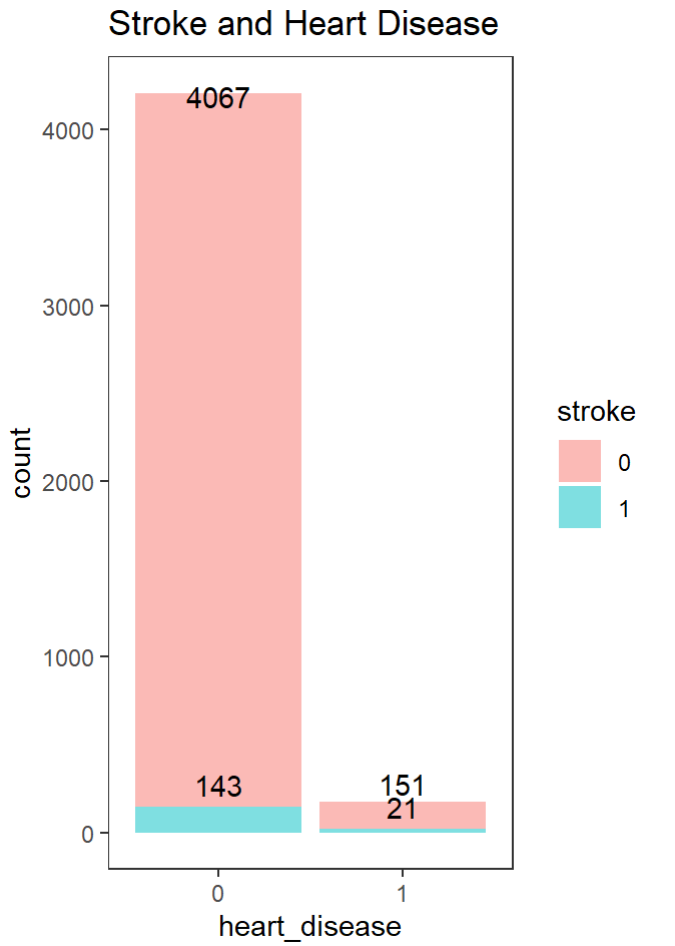
```
# those with hypertension are 3 time more likely to have a stroke than those who doesnt have hypertension




chisq.test(df$hypertension,df$stroke, correct = FALSE)
```

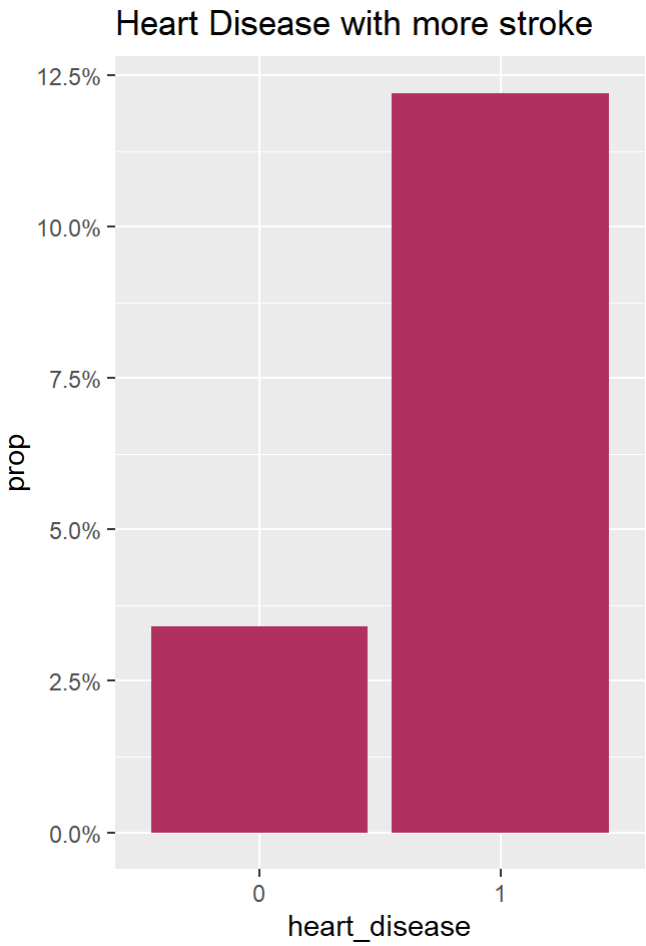
```
##
##  Pearson's Chi-squared test
##
## data:  df$hypertension and df$stroke
## X-squared = 56.575, df = 1, p-value = 5.409e-14
```

```
#x-squared and p value indicate there is strong association between hypertension and stroke. the p.value being extremely small one can reject null hypothesis.individuals with hypertensio n are more likely to have stroke compared to those without hypertension.
```

Heart Disease vs Stroke

```
#creating visual bar chart of both stroke and heart disease.
HD1<-df%>%
ggplot(aes(heart_disease,fill = stroke)) +
  geom_bar(alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Stroke and Heart Disease", X = "Stroke", Y = "Count")

#creating visual bar chart proportion of heart disease that has highest stroke case.
heart <- df%>%
  group_by(heart_disease)%>%
  summarise(prop = sum(stroke == "1")/length(heart_disease))
HD1p<-heart%>%
  ggplot(aes(x =heart_disease, y = prop))+
  geom_col(fill = "maroon")+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "Heart Disease with more stroke ")

grid.arrange(HD1, HD1p, ncol = 2, nrow = 1)
```





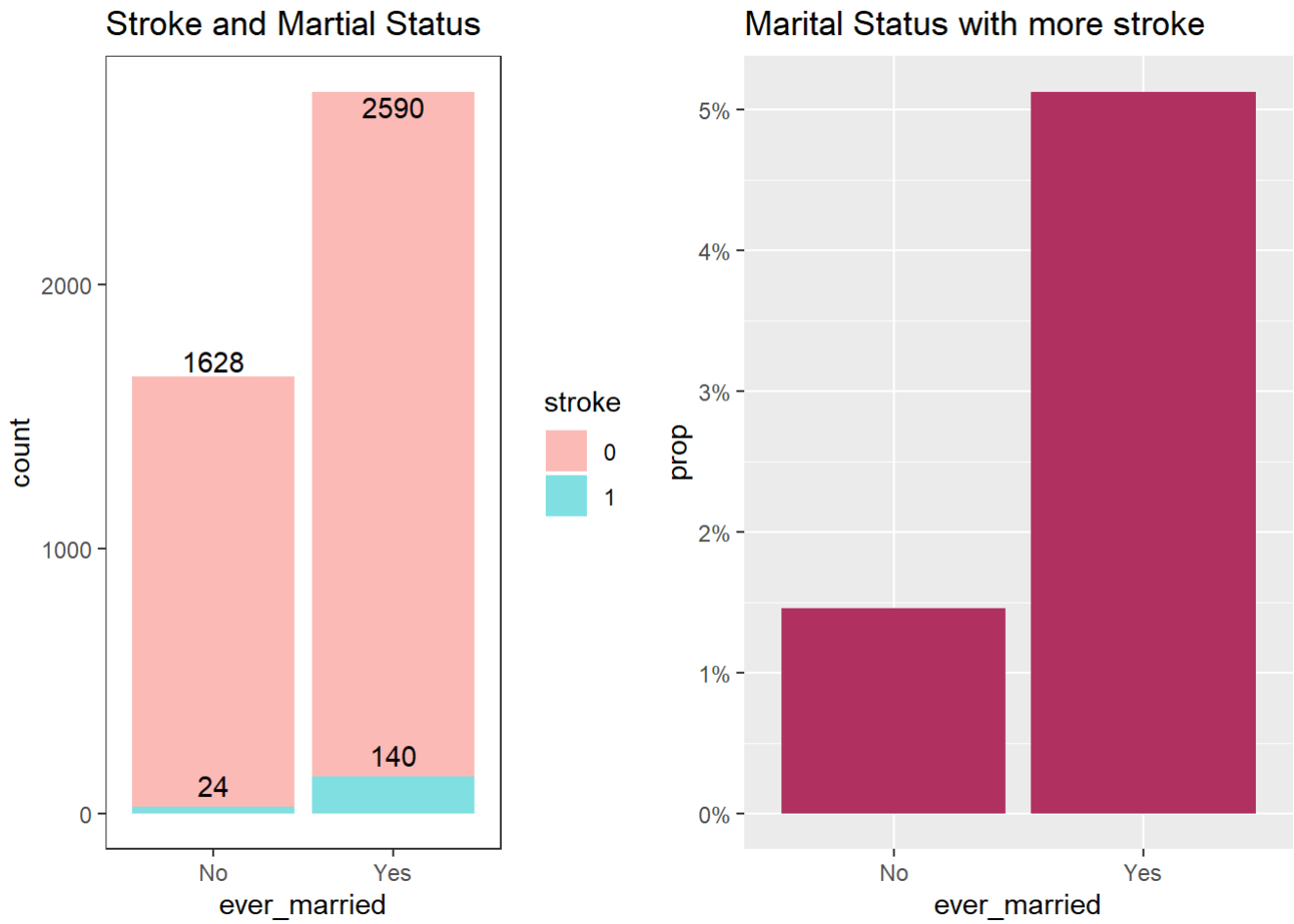

```
#patient with heart disease are 4 time more likely to have stroke than those who doesn't have heard disease.


chisq.test(df$heart_disease,df$stroke, correct = FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  df$heart_disease and df$stroke
## X-squared = 35.624, df = 1, p-value = 2.393e-09
```

```
#the p.value is extremely small one can reject the null hypothesis, safely assume that patients with heart disease are more likely to get stroke.
```

Marital status vs Stroke

```
#creating visual bar chart of both stroke and marital status.

M1<-df%>%
ggplot(aes(ever_married,fill = stroke)) +
  geom_bar(alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Stroke and Martial Status", X = "Stroke", Y = "Count")

#creating visual bar chart proportion of marital status that has highest stroke case.

marital <- df%>%
  group_by(ever_married)%>%
  summarise(prop = sum(stroke == "1")/length(ever_married))
M1p<-marital%>%
  ggplot(aes(x =ever_married, y = prop))+
  geom_col(fill = "maroon")+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "Marital Status with more stroke ")


grid.arrange(M1, M1p, ncol = 2, nrow = 1)
```



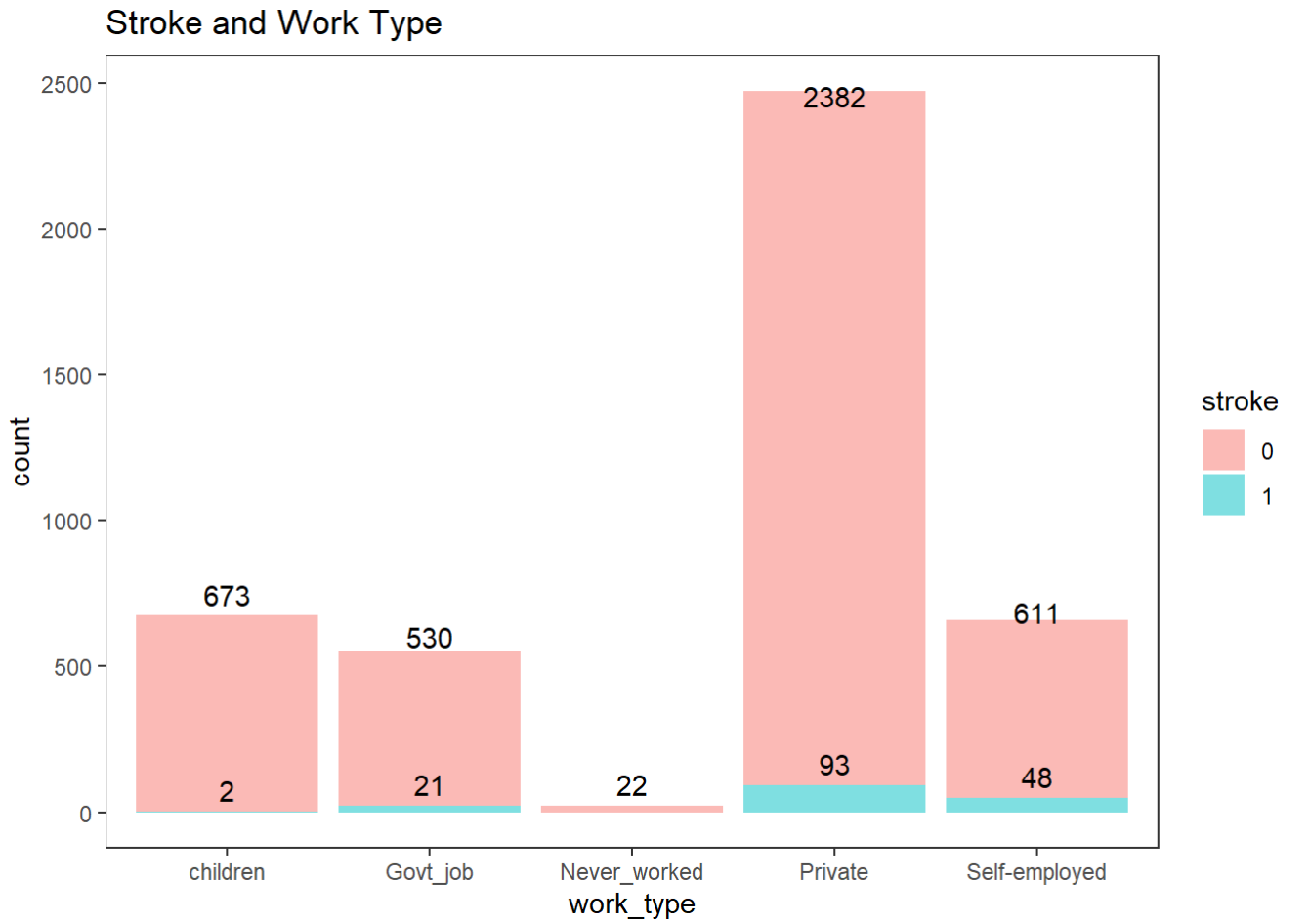

```
# being ever married increase having a stroke over 5 time than those who have not been ever m
arried

chisq.test(df$ever_married,df$stroke, correct = FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  df$ever_married and df$stroke
## X-squared = 38.593, df = 1, p-value = 5.221e-10
```

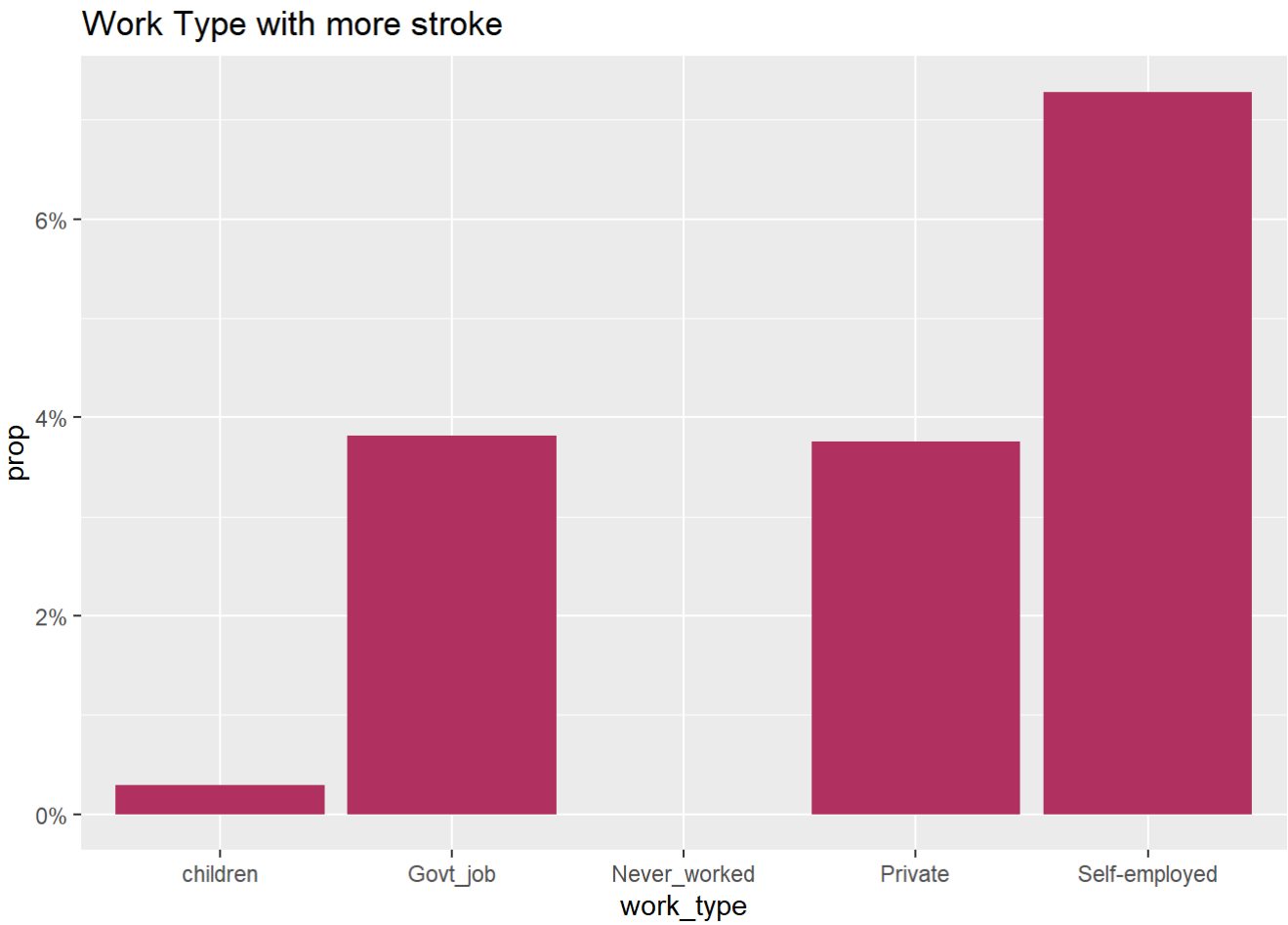
```
#those married are more likely to get stroke, there is strong  correlation as it also evident in the prop chart.
```

Work Type vs Stroke

```
#creating visual bar chart of both stroke and work type.

df%>%
ggplot(aes(work_type,fill = stroke)) +
  geom_bar( alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Stroke and Work Type", X = "Stroke", Y = "Count")
```



```
#creating visual bar chart proportion of work type that has highest stroke case.

work <-df%>%
  group_by(work_type)%>%
  summarise(prop = sum(stroke == "1")/length(work_type))
work%>%
  ggplot(aes(x =work_type, y = prop))+
  geom_col(fill = "maroon")+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "Work Type with more stroke ")
```




```
# those patient that were self employed had greater chance of getting a stroke, and those wit
h government or private job are almost identical in having a stroke, with very small number o
f children also having a stroke, this can be due other underlying health issue.



chisq.test(df$work_type,df$stroke, correct = FALSE)
```

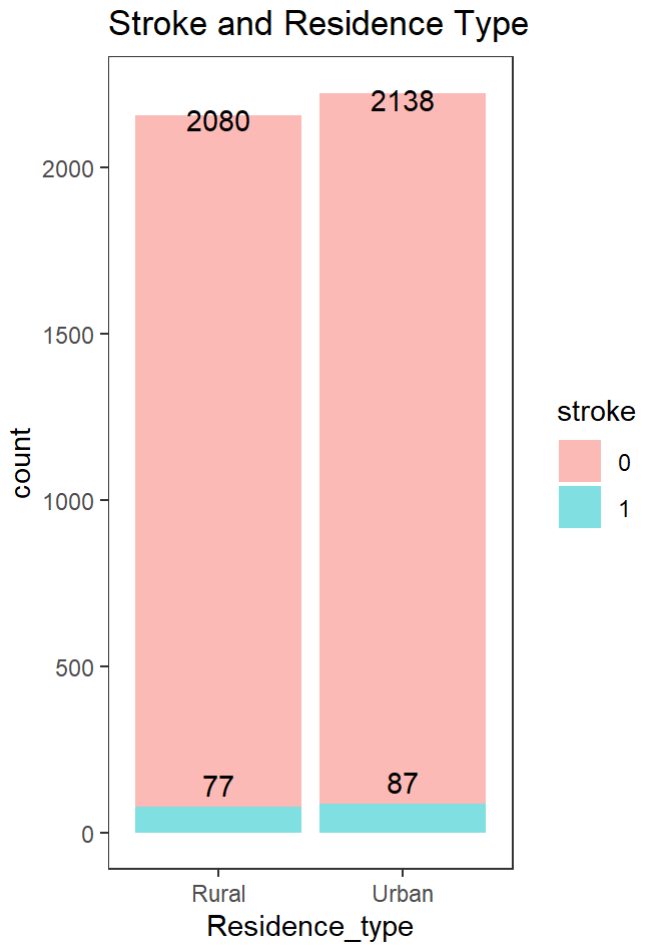
```
## Warning in chisq.test(df$work_type, df$stroke, correct = FALSE): Chi-squared
## approximation may be incorrect
```

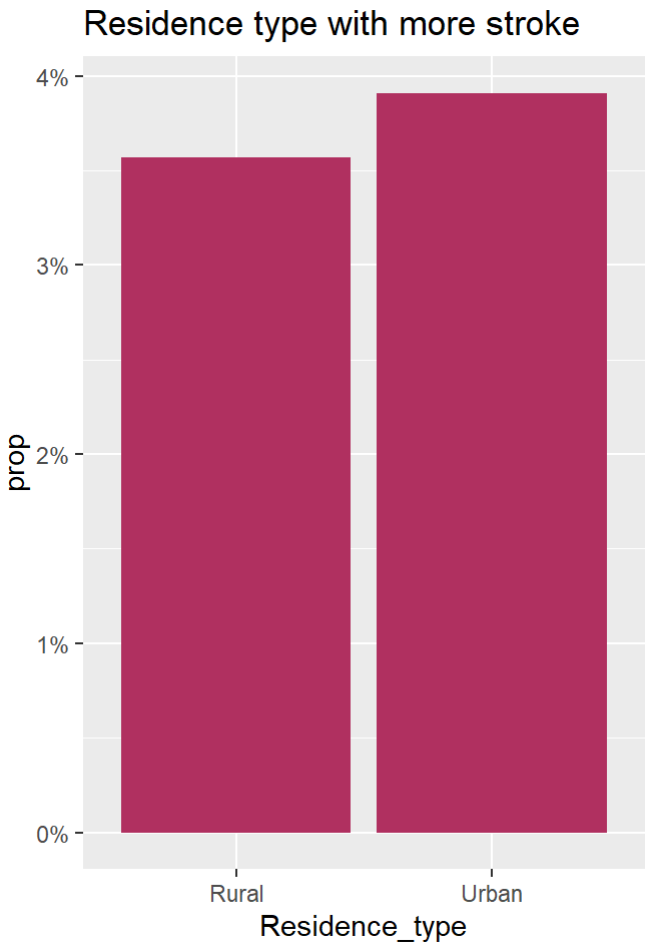
```
##
##  Pearson's Chi-squared test
##
## data:  df$work_type and df$stroke
## X-squared = 46.057, df = 4, p-value = 2.397e-09
```

```
#the results indicate there is relationship between those who work and stroke, we can safely reject null hypothesis
```

Residence Type vs Stroke

```
#creating visual bar chart of both stroke and residence type.

R1<-df%>%
ggplot(aes(Residence_type,fill = stroke)) +
  geom_bar(alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Stroke and Residence Type", X = "Stroke", Y = "Count")

#creating visual bar chart proportion of residence type that has highest stroke case.

Residence <- df%>%
  group_by(Residence_type)%>%
  summarise(prop = sum(stroke == "1")/length(Residence_type))
R1p<-Residence%>%
  ggplot(aes(x = Residence_type, y = prop))+
  geom_col(fill = "maroon")+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "Residence type with more stroke ")

grid.arrange(R1, R1p, ncol = 2, nrow = 1)
```






```
# patient living in urban areas have slighly greater chance of having a stroke


chisq.test(df$Residence_type,df$stroke, correct = FALSE)
```

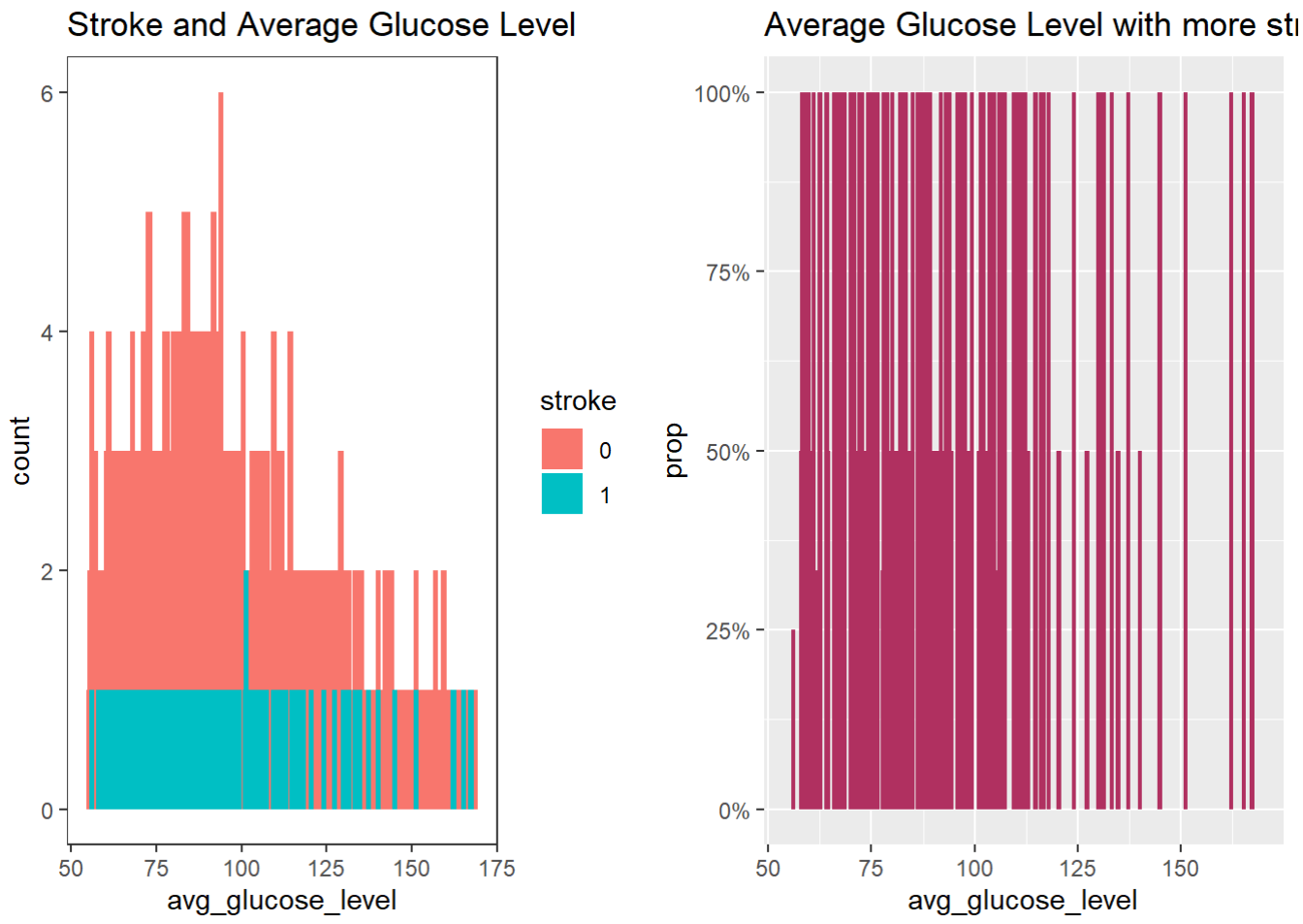
```
##
##  Pearson's Chi-squared test
##
## data:  df$Residence_type and df$stroke
## X-squared = 0.35215, df = 1, p-value = 0.5529
```

```
#p- value is over 0.55 we can safely accept null hypothesis as there is no or minimum affect
of residence type on stroke
```

Average glucose level vs Stroke

```
#creating visual bar chart of both stroke and glucose level.
G1<-df%>%
ggplot(aes(avg_glucose_level,fill = stroke)) +
  geom_bar( alpha = 2, width = 1.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Stroke and Average Glucose Level", X = "Stroke", Y = "Count")

#creating visual bar chart proportion of average glucose level that has highest stroke case.

glucose <- df%>%
  group_by(avg_glucose_level)%>%
  summarise(prop = sum(stroke == "1")/length(avg_glucose_level))
G1p<-glucose%>%
  ggplot(aes(x =avg_glucose_level, y = prop))+
  geom_col(fill = "maroon",width = 1)+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "Average Glucose Level with more stroke ")



grid.arrange(G1, G1p, ncol = 2, nrow = 1)
```

```
## Warning: `position_stack()` requires non-overlapping x intervals
## `position_stack()` requires non-overlapping x intervals
```



```
#stroke chances is distribution seems to be strogest at health range of under 100mg/dL, the r
est is almost equally distributed except few pockets around 150mg/dL that has no or low strok
e chances.



t.test(avg_glucose_level~stroke, data= df)
```

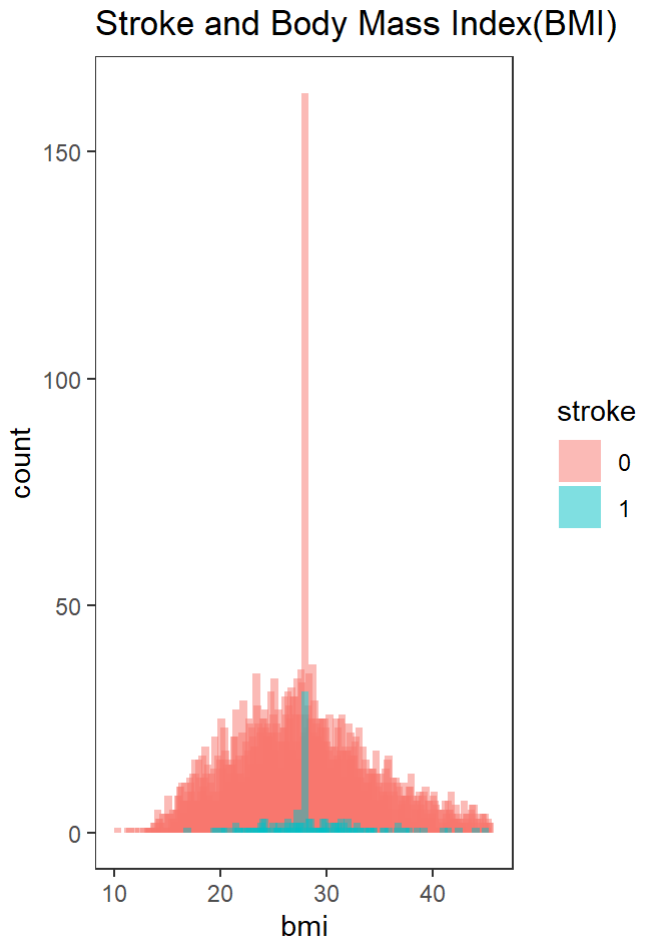
```
##
##  Welch Two Sample t-test
##
## data:  avg_glucose_level by stroke
## t = -0.013821, df = 174.26, p-value = 0.989
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal
to 0
## 95 percent confidence interval:
##  -3.813067  3.760033
## sample estimates:
## mean in group 0 mean in group 1
##        91.46330        91.48982
```

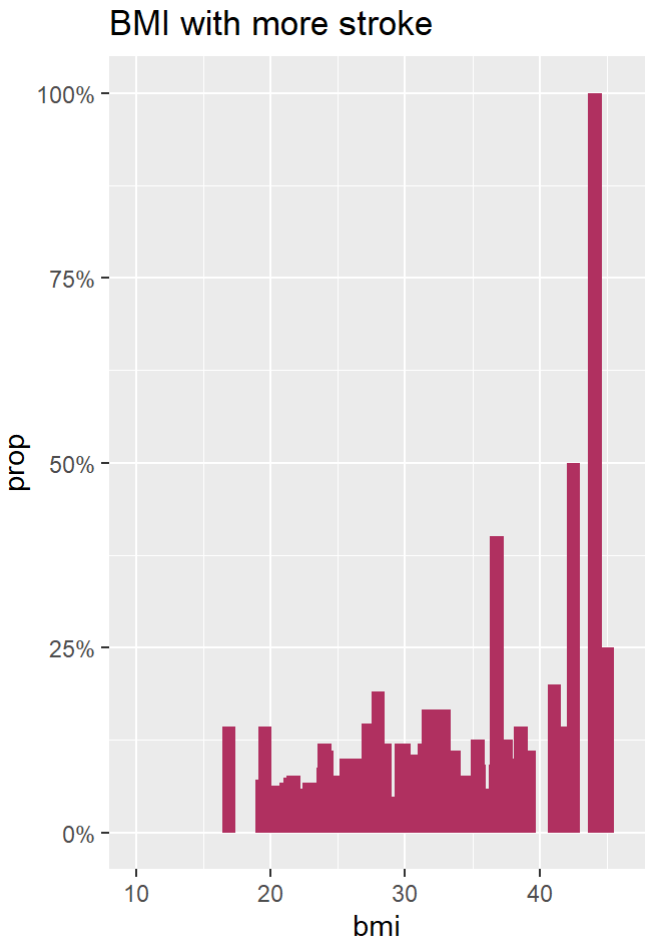
```
# with p-value of almost 1, from this we can conclude there is no link between stroke and ave
rage glucose level, with almost same mean in both groups of stroke and no stroke
```

BMI vs Stroke

```
#creating visual bar chart of both stroke and BMI.

B1<-df%>%
ggplot(aes(bmi,fill = stroke)) +
  geom_bar( alpha = 0.5,width = 0.7)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Stroke and Body Mass Index(BMI)", X = "Stroke", Y = "Count")

#creating visual bar chart proportion of BMI that has highest stroke case.
BMI <- df%>%
  group_by(bmi)%>%
  summarise(prop = sum(stroke == "1")/length(bmi))
B1p<-BMI%>%
  ggplot(aes(x =bmi, y = prop))+
  geom_col(fill = "maroon",width = 1)+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "BMI with more stroke ")

grid.arrange(B1, B1p, ncol = 2, nrow = 1)
```

```
## Warning: `position_stack()` requires non-overlapping x intervals
## `position_stack()` requires non-overlapping x intervals
```





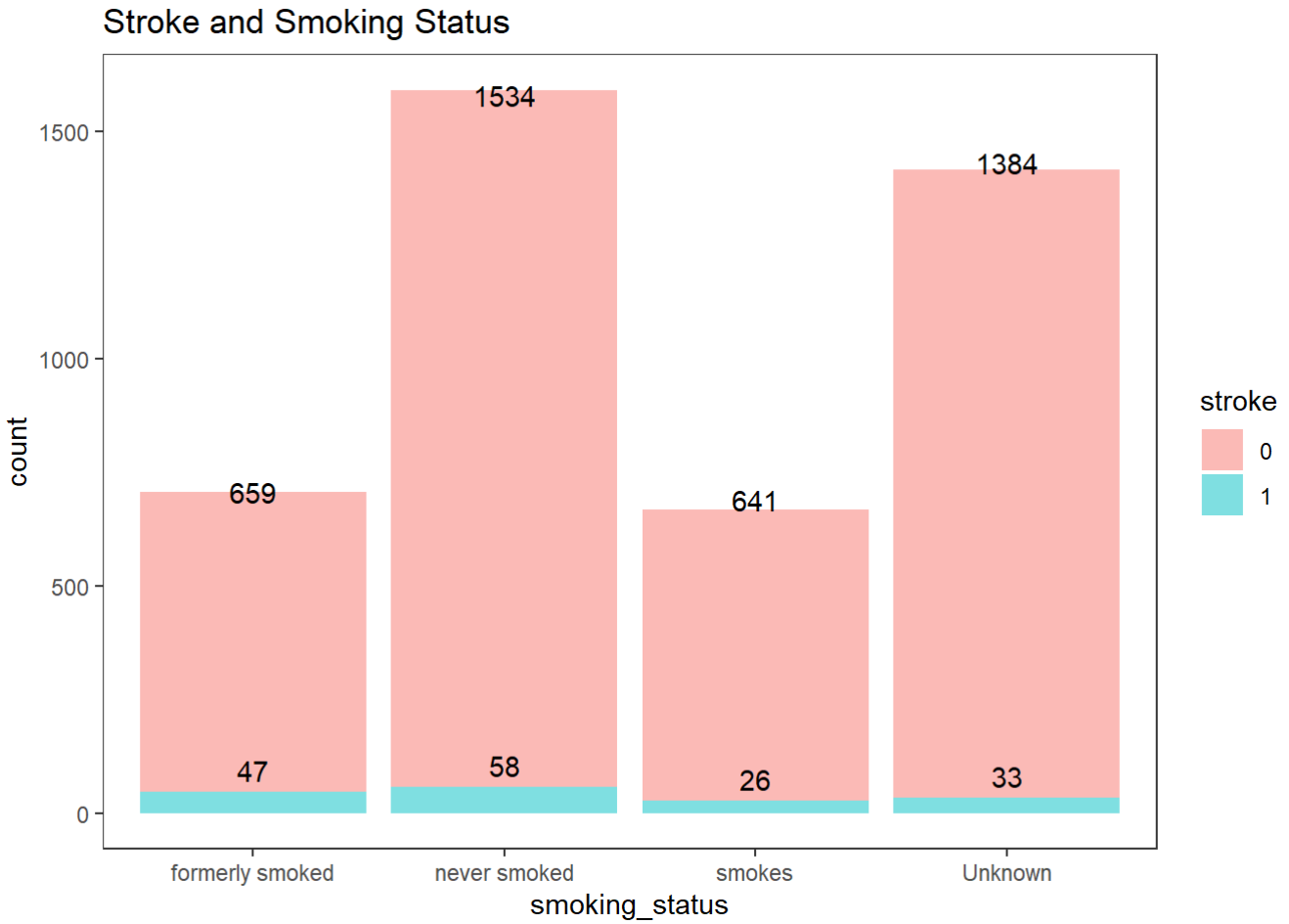

```
# vast majority of the stroke patients have BMI over 25 and below 50, which is considered ove
rweight at lower end and extremely obese on higher end, so there seems to be correlation betw
een the two.

t.test(bmi~stroke, data= df)
```

```
##
##  Welch Two Sample t-test
##
## data:  bmi by stroke
## t = -2.5251, df = 187.74, p-value = 0.01239
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal
to 0
## 95 percent confidence interval:
##  -1.7421370 -0.2139698
## sample estimates:
## mean in group 0 mean in group 1
##        27.70609        28.68415
```

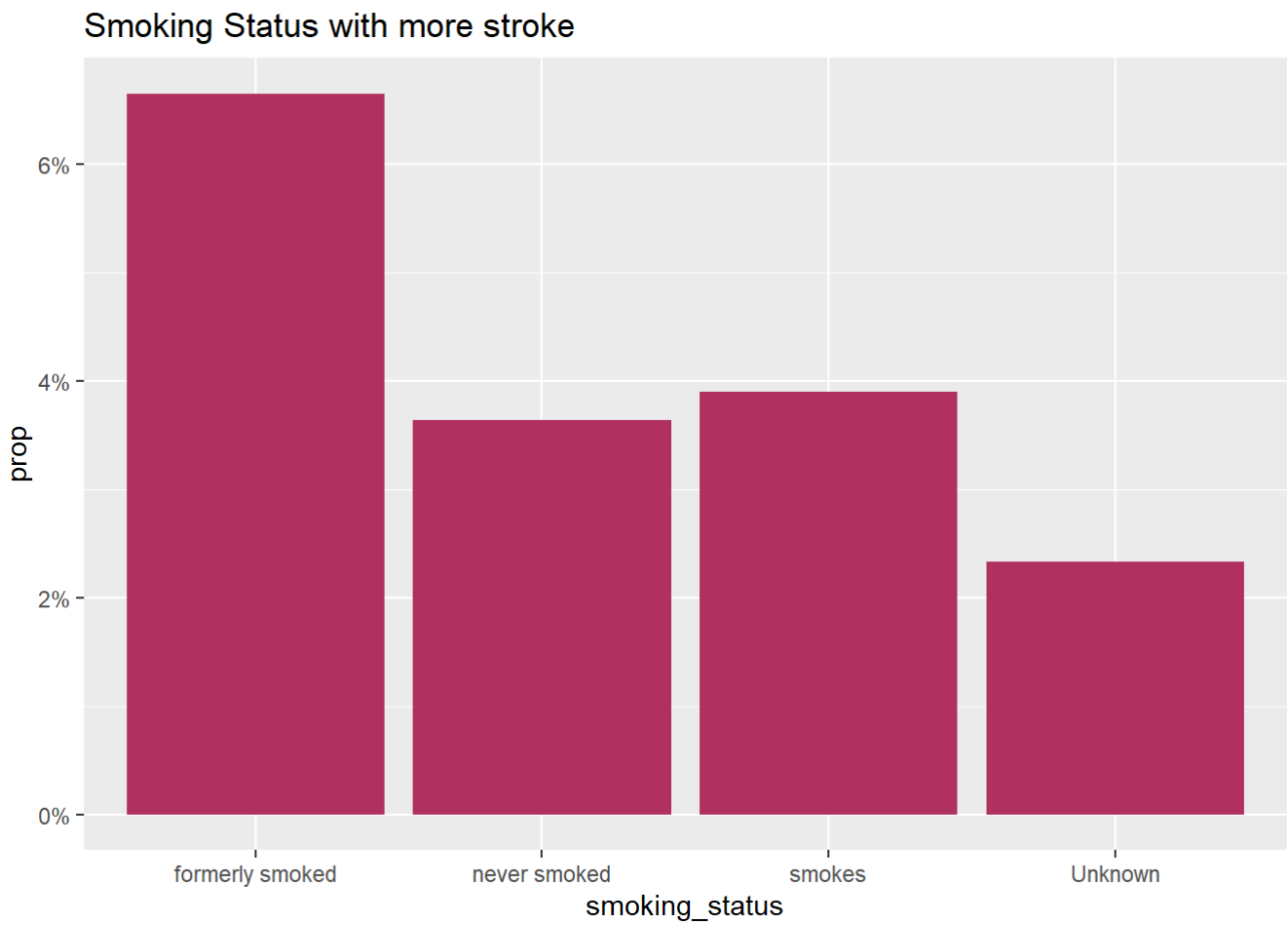
```
# the mean bmi of those had stroke seems to be higher than those who didnt have stroke. but p
-value indicates that the null hypothesis can be rejected.
```

Smoking Status vs Stroke

```
#creating visual bar chart of both stroke and smoking status.

df%>%
ggplot(aes(smoking_status,fill = stroke)) +
  geom_bar( alpha = 0.5)+
  geom_text(stat='count', aes(label=after_stat(count)), vjust=-.5)+
  theme_bw()+
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())+
  labs(title = "Stroke and Smoking Status", X = "Stroke", Y = "Count")
```




```
#creating visual bar chart proportion of smoking status that has highest stroke case.

smoking<- df%>%
  group_by(smoking_status)%>%
  summarise(prop = sum(stroke == "1")/length(smoking_status))
smoking%>%
  ggplot(aes(x =smoking_status, y = prop))+
  geom_col(fill = "maroon")+
  scale_y_continuous(labels = scales::percent_format())+
    labs(title = "Smoking Status with more stroke ")
```




```
#formerly smoked patients had the highest chance of having a stroke, those who smoke came sec
ond, third came the never smoked patients and finally the unknown category had the least chan
ce of having a stroke, therefore replacing the unknown category with any other group will not
be fare representation of data.

chisq.test(df$smoking_status,df$stroke, correct = FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  df$smoking_status and df$stroke
## X-squared = 24.598, df = 3, p-value = 1.874e-05
```

```
#with values obtained in the test we can safely reject null hypothesis and there is correlati
on between those who smoke and stroke.
```

Multivariate Analysis

Before embarking on this type statistical analysis, we are going to select age 40 and above from the data set, because the number of stroke patient gradually increases from around age 40 onwards, please refer to prop graph between stroke and age, although it reduces the number of dataset but we have very small percentage of stroke patients and patients below age 40 will just dilute the data set without bringing any meaningful contribution.

```
subset_df <- df[df$age > 40, ]
str(subset_df)
```

```
## 'data.frame':    2267 obs. of  11 variables:
##  $ gender           : Factor w/ 2 levels "Female","Male": 2 2 1 1 1 1 1 1 1 1 ...
##  $ age              : num  80 74 69 59 78 81 61 54 50 60 ...
##  $ hypertension     : Factor w/ 2 levels "0","1": 1 2 1 1 1 2 1 1 2 1 ...
##  $ heart_disease    : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 2 1 1 1 ...
##  $ ever_married     : Factor w/ 2 levels "No","Yes": 2 2 1 2 2 2 2 2 2 1 ...
##  $ work_type        : Factor w/ 5 levels "children","Govt_job",..: 4 4 4 4 4 4 2 4 5 4 ...
##  $ Residence_type   : Factor w/ 2 levels "Rural","Urban": 1 1 2 1 2 1 1 2 1 2 ...
##  $ avg_glucose_level: num  105.9 70.1 94.4 76.2 58.6 ...
##  $ bmi              : num  32.5 27.4 22.8 28 24.2 29.7 36.8 27.3 30.9 37.8 ...
##  $ smoking_status   : Factor w/ 4 levels "formerly smoked",..: 2 2 2 4 4 2 3 3 2 2 ...
##  $ stroke           : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
```

we will attempt to prove or fail to prove the hypothesis by doing following analysis. We will now do logistic regression and find the variables that has significant impact on increasing the chances of having a stroke, we will also use Anova test to observe any relationship exist between predictor(i.e. age, gender, etc..) variables and stroke variable, from results obtained from Anova we can examine the Deviance, Residual Deviance, and p-value for each variable added to the model, we can determine which predictor variables have a significant effect on stroke.

In the project proposal we formed 3 questions by grouping variables by health condition, Socio-economic and biology. the questions and variables are mentioned below

1. Does underlying health conditions increase or decrease chances of getting a stroke? By using data from following column. a. Hypertension b. Heart disease c. Average glucose level d. BMI

```
healthCon <- glm(subset_df$stroke ~ subset_df$hypertension +subset_df$heart_disease +subset_d
f$avg_glucose_level+ subset_df$bmi, family = "binomial")
summary(healthCon)
```

```
##
## Call:
## glm(formula = subset_df$stroke ~ subset_df$hypertension + subset_df$heart_disease +
##     subset_df$avg_glucose_level + subset_df$bmi, family = "binomial")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7863  -0.3879  -0.3477  -0.3135   2.6738
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -1.786940   0.564649  -3.165  0.00155 **
## subset_df$hypertension1       0.808989   0.204215   3.961 7.45e-05 ***
## subset_df$heart_disease1      0.676629   0.252005   2.685  0.00725 **
## subset_df$avg_glucose_level   0.002314   0.003644   0.635  0.52534
## subset_df$bmi                -0.042408   0.016031  -2.645  0.00816 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1136  on 2266  degrees of freedom
## Residual deviance: 1109  on 2262  degrees of freedom
## AIC: 1119
##
## Number of Fisher Scoring iterations: 5
```

```
#The variable "Average glucose level" has p-values of 0.525, which is greater than 0.05, the null hypothesis cannot be rejected for this category.

#The variable "hypertension, heart disease, and bmi" are with a p-value of less than 0.05. This means that the null hypothesis can be rejected for this subcategory and it indicates a significant effect on the variable "stroke".


#we will analysis of variance using chi squared test.
anova(healthCon, test = 'Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: subset_df$stroke
##
## Terms added sequentially (first to last)
##
##
##                              Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                          2266     1136.0
## subset_df$hypertension        1  13.3354      2265     1122.7 0.0002604 ***
## subset_df$heart_disease       1   6.1577      2264     1116.5 0.0130840 *
## subset_df$avg_glucose_level   1   0.2980      2263     1116.2 0.5851687
## subset_df$bmi                 1   7.2428      2262     1109.0 0.0071186 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#The deviance is a measure of how well the model fits the data. The smaller the residual deviance, the better the model fits the data.In the deviance results from test shows that the Null deviance is 1136 on 2266 degrees of freedom, and the Residual deviance is 1109 on 2262 degrees of freedom, which is smaller thus making it good glm a good fit for this analysis
```

2. **Does social/environmental status increase or decrease chances of getting a stroke? By using data from following columns.**
    a. Residence Type
    b. Work type
    c. Ever married
    d. Smoking Status

```
Socioeconomic <- glm(subset_df$stroke ~ subset_df$ever_married+subset_df$work_type+subset_df$Residence_type+subset_df$smoking_status, family = "binomial")
summary(Socioeconomic)
```

```
##
## Call:
## glm(formula = subset_df$stroke ~ subset_df$ever_married + subset_df$work_type +
##     subset_df$Residence_type + subset_df$smoking_status, family = "binomial")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.5837  -0.4008  -0.3579  -0.3399   2.5274
##
## Coefficients:
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                        -2.21464    0.35810  -6.184 6.23e-10 ***
## subset_df$ever_marriedYes          -0.51304    0.24823  -2.067   0.0388 *
## subset_df$work_typePrivate          0.24621    0.25017   0.984   0.3250
## subset_df$work_typeSelf-employed    0.44798    0.27279   1.642   0.1005
## subset_df$Residence_typeUrban       0.08319    0.16692   0.498   0.6182
## subset_df$smoking_statusnever smoked -0.31792    0.20835  -1.526   0.1270
## subset_df$smoking_statussmokes     -0.42431    0.26637  -1.593   0.1112
## subset_df$smoking_statusUnknown    -0.24526    0.24360  -1.007   0.3140
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1136.0  on 2266  degrees of freedom
## Residual deviance: 1125.7  on 2259  degrees of freedom
## AIC: 1141.7
##
## Number of Fisher Scoring iterations: 5
```

```
#The subcategories of the variable "work_type" have p-values of 0.3250 for "Private" and 0.10
05 for "Self-employed". As both of these p-values are greater than 0.05, the null hypothesis
cannot be rejected for these two categories.

#The subcategories of the variable "smoking_status" have p-values of 0.1270 for "never smoke
d", 0.1112 for "smokes" and 0.3140 for "Unknown". As all of these p-values are greater than
0.05, the null hypothesis cannot be rejected for these three categories.

#However, the variable "ever_married" has a subcategory "Yes" with a p-value of 0.0388, which
is less than 0.05. This means that the null hypothesis can be rejected for this subcategory a
nd it indicates a significant effect on the response variable "stroke".


anova(Socioeconomic, test = 'Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: subset_df$stroke
##
## Terms added sequentially (first to last)
##
##
##                          Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                                      2266     1136.0
## subset_df$ever_married    1   3.6224      2265     1132.4  0.05701 .
## subset_df$work_type       2   3.1833      2263     1129.2  0.20359
## subset_df$Residence_type  1   0.2341      2262     1129.0  0.62852
## subset_df$smoking_status  3   3.3022      2259     1125.7  0.34734
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#shows the results are reliable from glm for reason mentioned in question 1.
```

3. **Does human biology affect chances of getting a stroke? By using data from following columns.**

a. Age
b. Gender

```
Bio <- glm(subset_df$stroke ~ subset_df$gender + subset_df$age, family = "binomial")
summary(Bio)
```

```
##
## Call:
## glm(formula = subset_df$stroke ~ subset_df$gender + subset_df$age,
##     family = "binomial")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7173  -0.4098  -0.2884  -0.2100   2.8656
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -7.230223   0.529919 -13.644   <2e-16 ***
## subset_df$genderMale  0.135146   0.172686   0.783    0.434
## subset_df$age         0.071570   0.007548   9.482   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1136.0  on 2266  degrees of freedom
## Residual deviance: 1033.2  on 2264  degrees of freedom
## AIC: 1039.2
##
## Number of Fisher Scoring iterations: 6
```

```
#The results show that there is a significant relationship between stroke and age (p-value < 2e-16), with a positive association (coefficient = 0.071570), meaning that as age increases, the odds of having a stroke increases as well.

#However, there is no significant relationship between stroke and gender (p-value = 0.434), as the coefficient of gender is small (0.135146) and the p-value is not lower than the significance level.



anova(Bio, test = 'Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: subset_df$stroke
##
## Terms added sequentially (first to last)
##
##
##                   Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                                2266     1136.0
## subset_df$gender  1    0.185      2265     1135.8   0.6668
## subset_df$age     1  102.631      2264     1033.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#shows the results are reliable from glm for reason mentioned during question 1.
```

Modelling

We will now attempt to create predictive models, our data set is imbalance and in order to have a model accuracy, curb any biases and evaluation metrics that may result from using the current data set as is. we will create 3 predictive models one with actual data set, the second will be subset od date for patients age 40 and above and the third one for balanced oversampled dataset we will create in following steps.

we will look at our data set and its stroke proportion

step 1: we will make variables factor and over-sample the original data for balancing, we will look at our data set and its stroke proportion

```
# We will turn stroke variable to factor for classification model,
#Actual dataset
whole_dataset<-df %>%
    mutate(
        stroke = as.character(stroke),
        across(where(is.factor), as.numeric),
        stroke = factor(stroke)
    )

whole_dataset %>%
group_by(stroke) %>%
summarize(n = n()) %>%
mutate(prop = round(n / sum(n), 2))
```

```
## # A tibble: 2 × 3
##   stroke     n  prop
##   <fct>  <int> <dbl>
## 1 0       4218  0.96
## 2 1        164  0.04
```

```
# As we can see we have 5 percent patients with stroke and 95% without stroke.


# We will turn stroke variable to factor for classification model,
#over 40 dataset
overforty <-subset_df %>%
    mutate(
        stroke = as.character(stroke),
        across(where(is.factor), as.numeric),
        stroke = factor(stroke)
    )

overforty %>%
group_by(stroke) %>%
summarize(n = n()) %>%
mutate(prop = round(n / sum(n), 2))
```

```
## # A tibble: 2 × 3
##   stroke     n  prop
##   <fct>  <int> <dbl>
## 1 0       2111  0.93
## 2 1        156  0.07
```

```
# As we can see we have 8 percent patients with stroke and 92% without stroke.




oversample_dataset <- df %>%
    mutate(
        stroke = as.character(stroke),
        across(where(is.factor), as.numeric),
        stroke = factor(stroke)
    )

stroke_oversampled <- oversample(as.data.frame(oversample_dataset), classAttr = "stroke", rat
io =1, method = "MWMOTE")


#The over sampled data is now stored in the over sampled_data variable, and can be used for b
uilding a predictive model

stroke_oversampled %>%
    group_by(stroke) %>%
    summarize(n = n()) %>%
    mutate(prop = round(n / sum(n), 2))
```

```
## # A tibble: 2 × 3
##   stroke     n  prop
##   <fct>  <int> <dbl>
## 1 0       4218   0.5
## 2 1       4218   0.5
```

```
# the stroke proportion is 50% now
```

Step 2: Splitting the data into training(70%) and testing data(30%).

```
#Whole dataset
set.seed(123)
wholedatase_indices <- createDataPartition(whole_dataset$stroke, p = 0.7, list = FALSE)

# divide the data into training, validation and testing data set
whole_train_data <- whole_dataset[wholedatase_indices,]
whole_validation_data <- whole_dataset[-wholedatase_indices,]


#Define the response and predictor variables:
wholex <- subset(whole_train_data, select = -stroke)
wholey <- whole_train_data$stroke

#over 40 dataset
overforty_indices <- createDataPartition(overforty$stroke, p = 0.7, list = FALSE)

# divide the data into training, validation and testing data set
overforty_train_data <- overforty[overforty_indices,]
overforty_validation_data <- overforty[-overforty_indices,]


#Define the response and predictor variables:
fortyx <- subset(overforty_train_data, select = -stroke)
fortyy <- overforty_train_data$stroke







#oversample dataset
indices <- createDataPartition(stroke_oversampled$stroke, p = 0.7, list = FALSE)

# divide the data into training, validation and testing data set
train_data <- stroke_oversampled[indices,]
validation_data <- stroke_oversampled[-indices,]


#Define the response and predictor variables:
x <- subset(train_data, select = -stroke)
y <- train_data$stroke
```

step 3: We will now create random forest model and train in with the validation data.

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.2.2
```

```
## randomForest 4.7-1.1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:gridExtra':
##
##     combine
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

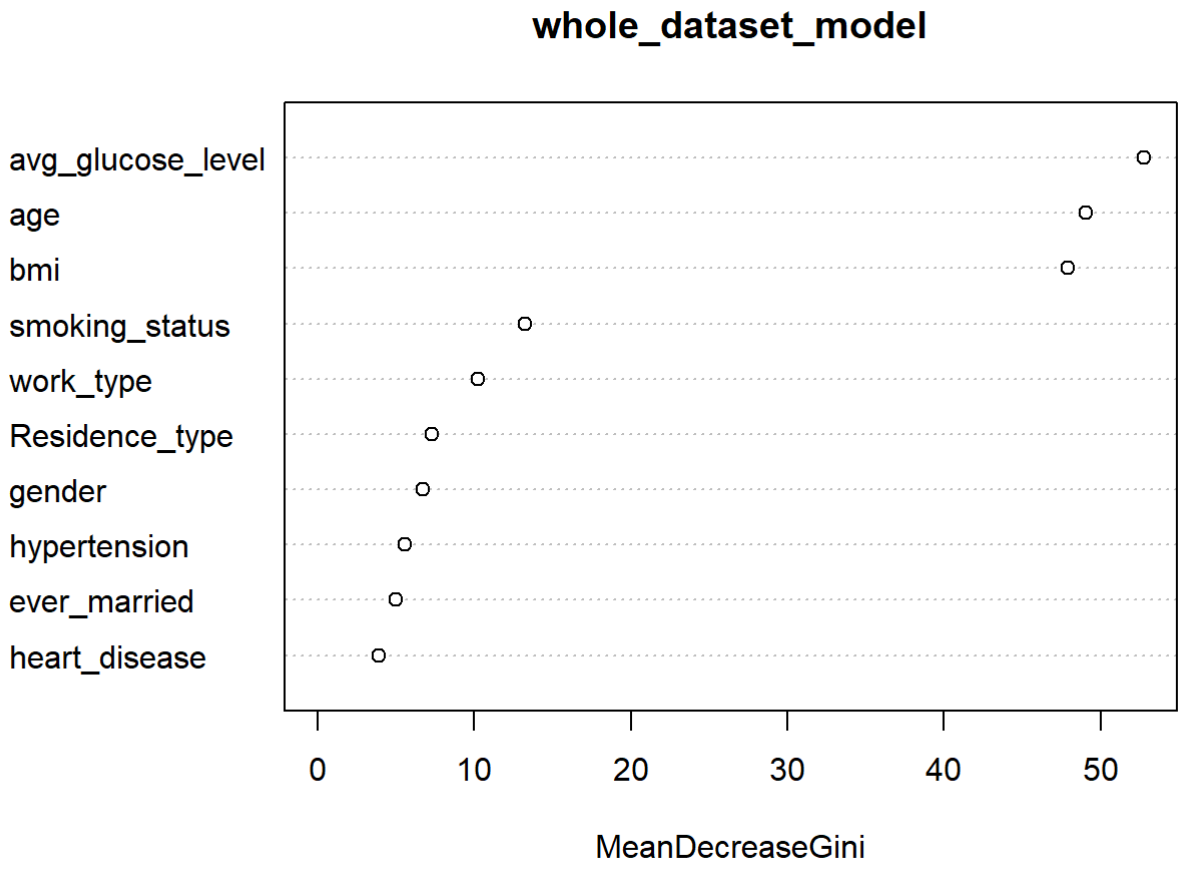
```
#whole dataset Model
whole_dataset_model <- randomForest(wholex, wholey, ntree = 500)

#we will now test the model with the testing data creating in step 3.

whole_dataset_predictions <- predict(whole_dataset_model, newdata = whole_validation_data[-nc
ol(whole_validation_data)])


#over forty Model
over_forty_model <- randomForest(fortyx, fortyy, ntree = 500)

#we will now test the model with the testing data creating in step 3.

over_forty_predictions <- predict(over_forty_model, newdata = overforty_validation_data [-nco
l(overforty_validation_data)])



#oversample_Model
oversample_model <- randomForest(x, y, ntree = 500)

#we will now test the model with the testing data creating in step 3.

predictions <- predict(oversample_model, newdata = validation_data[-ncol(validation_data)])
```

step 4: we will evaluate the performance with help of confusionMatrix and visualize the importance of each variable.

```
#Whole data set model results
confusionMatrix(whole_dataset_predictions, whole_validation_data$stroke)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 1263   48
##          1    2    1
##
##                Accuracy : 0.9619
##                  95% CI : (0.9501, 0.9716)
##     No Information Rate : 0.9627
##     P-Value [Acc > NIR] : 0.5947
##
##                   Kappa : 0.0343
##
##  Mcnemar's Test P-Value : 1.966e-10
##
##             Sensitivity : 0.99842
##             Specificity : 0.02041
##          Pos Pred Value : 0.96339
##          Neg Pred Value : 0.33333
##              Prevalence : 0.96271
##          Detection Rate : 0.96119
##    Detection Prevalence : 0.99772
##       Balanced Accuracy : 0.50941
##
##        'Positive' Class : 0
##
```

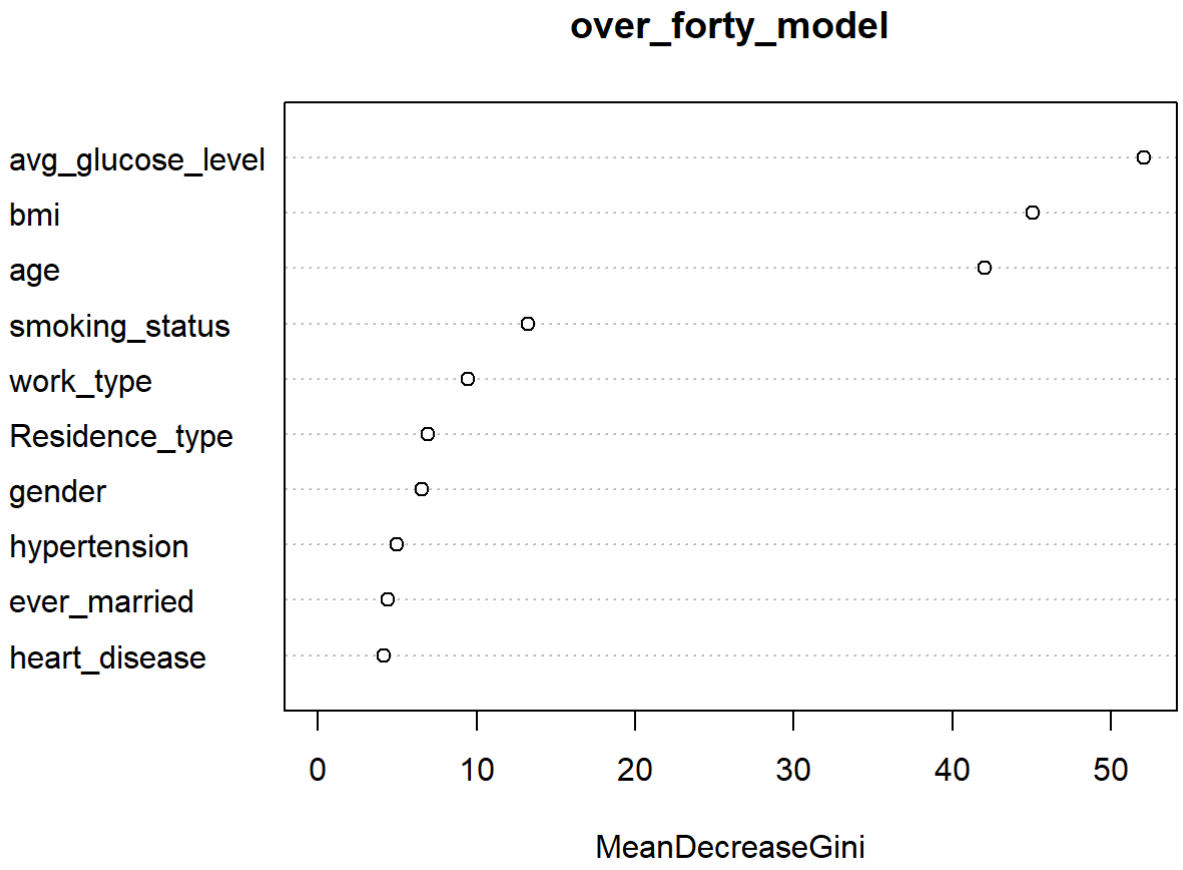
```
#the whole dataset model has been overall good metrics, with 96.19% of the test samples class
ified correctly. 95% of the time the accuracy score will fall between 0.9501 and 0.9716. high
sensitivity,low specificity and high precision

varImpPlot(whole_dataset_model)
```




```
#from this we can assume that age, bmi, and average glucose level are the best predictors of stroke variable, followed by smoking status, work type,residence, gender, hypertension, marital status and heart disease.

#over forty data set model results
confusionMatrix(over_forty_predictions, overforty_validation_data$stroke)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 632  46
##          1   1   0
##
##                Accuracy : 0.9308
##                  95% CI : (0.909, 0.9487)
##     No Information Rate : 0.9323
##     P-Value [Acc > NIR] : 0.5986
##
##                   Kappa : -0.0029
##
##  Mcnemar's Test P-Value : 1.38e-10
##
##             Sensitivity : 0.9984
##             Specificity : 0.0000
##          Pos Pred Value : 0.9322
##          Neg Pred Value : 0.0000
##              Prevalence : 0.9323
##          Detection Rate : 0.9308
##    Detection Prevalence : 0.9985
##       Balanced Accuracy : 0.4992
##
##        'Positive' Class : 0
##
```

```
#the over 40 dataset model has been overall good metrics, with slightly lesser accuracy of 9
3%,  95% of the time the accuracy score will fall between 0.90 and 0.94. Sensitivity (True Po
sitive Rate) is 0.99842, and Specificity (True Negative Rate) is 0.000.

varImpPlot(over_forty_model)
```

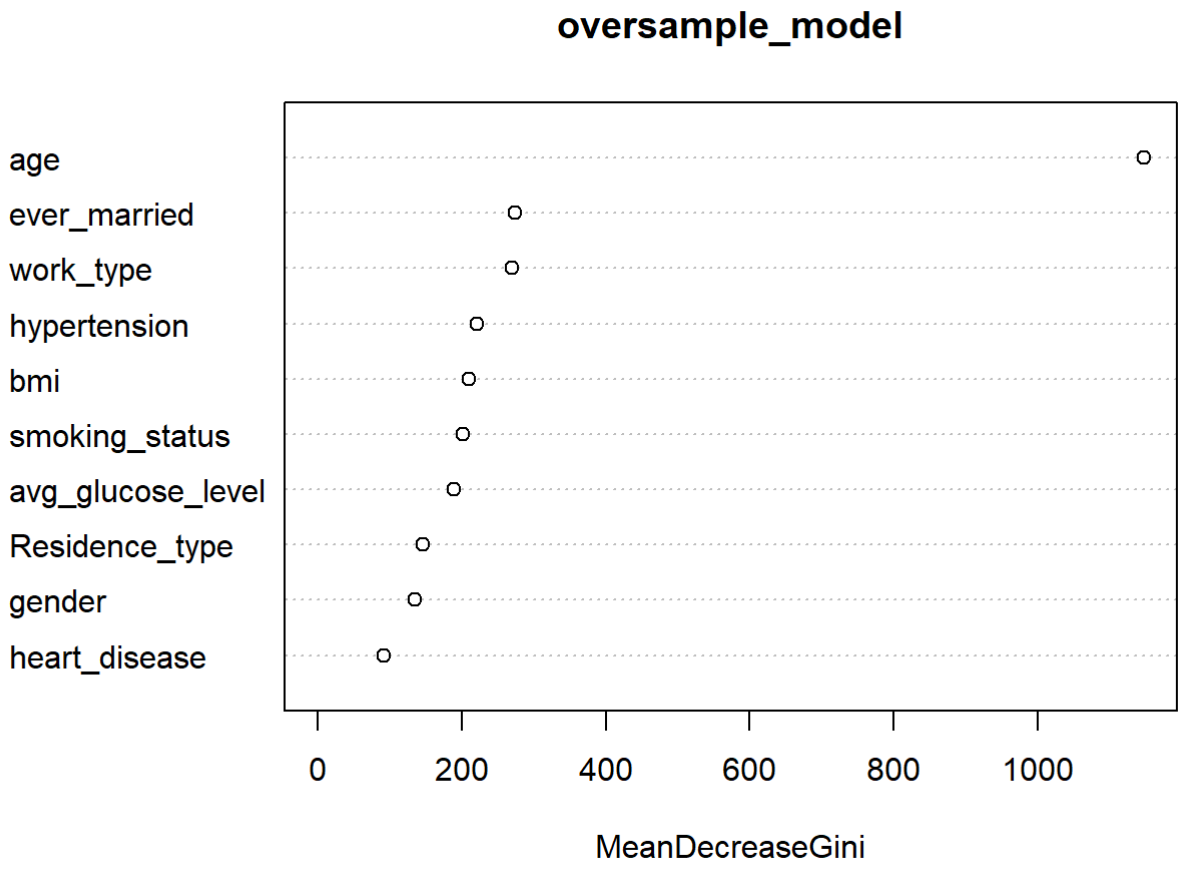



```
#from this we can assume that average glucose, bmi and age(one level down compare to whole da
ta set)  level are the best predictors of stroke variable, followed by smoking status, work t
ype,residence, gender, martial status(one level up than compare to whole data set), hypertens
ion and heart disease..



#over sampled data set results
confusionMatrix(predictions, validation_data$stroke)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 1233   43
##          1   32 1222
##
##                Accuracy : 0.9704
##                  95% CI : (0.963, 0.9766)
##     No Information Rate : 0.5
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.9407
##
##  Mcnemar's Test P-Value : 0.2482
##
##             Sensitivity : 0.9747
##             Specificity : 0.9660
##          Pos Pred Value : 0.9663
##          Neg Pred Value : 0.9745
##              Prevalence : 0.5000
##          Detection Rate : 0.4874
##    Detection Prevalence : 0.5043
##       Balanced Accuracy : 0.9704
##
##        'Positive' Class : 0
##
```

#the over sampled dataset model has been overall good metrics aswell, with higher accuracy th an over 40 model and the actual whole data set, 97% accuracy. Sensitivity (True Positive Rat e) is 0.9826, and Specificity (True Negative Rate) is 0.9628.

```
varImpPlot(oversample_model)
```




```
#from this we can assume that age are the best predictor of stroke variable, following by les
ser degree, marital status, work type, hypertension, bmi, smoking status, average glucose lev
el, residence type, gender and heart disease. common theme was that heart disease seem was lo
west predictors in all three type of models, and age were amongst the top 3 in all three mode
l. first 2 models were imbalanced dataset with only small percentage of stroke present in it,
```

Statement of results and Conclusions:

We have carried out data analysis on patient data to predict stroke, we explored the data for better understanding first by visualising. Once the data was cleansed and processed, we conducted hypothesis testing during bivariant and multivariate analysis, as outlined in the project proposal. The following variables seem to support the null hypothesis, gender, residence and average glucose level. The remaining variables (age, smoking status, hypertension, marital status, work type, heart disease, and BMI) showed significant association with the target variable of stroke, therefore, rejecting the Null Hypothesis and proving the alternative Hypothesis. We used three modified data sets for the prediction of stroke. firstly, the whole data set after cleaning during the data understanding phase, which had about 4% stroke patients and 96% none stroke patients. The second model used a subset of the whole data set of patients aged 40 and above. The reason behind this division of data was during bivariant analysis, it was evident stroke incidence increased with age after 40, and we had an insignificant number of stroke patients before age 40, which can be considered “outliers”. therefore, we wanted to prevent data dilution and only use a relevant segment of the data for better prediction, and the percentage of stroke patients was 8% in the over-forty data set, almost doubling it. Thirdly, we want to observe the effect of higher stroke patient percentage in our data on the model. therefore, the whole data set were oversampled to increase stroke patients, after the process, we had where 50% of patients with stroke and 50% didn’t have a stroke. Random forest algorithm was used as a predictive modelling tool, although attempts were made to use logistics regression but were met with poor results and issues. The results of the random forest have been provided using a confusion matrix. The accuracy of all 3 models was quite high, the model with over 40 dataset perform poorly compared to the whole data set and over the sample data set. Oversampled data set model had 97 per cent accuracy 98% sensitivity 96% specificity outperforming

other 2 data set models. The Variable Important Plot function was used to rank and compare the importance of different variables in the model, which helped identify the most important predictor of stroke, using this information we can remove any variable that will have the least impact on predicting stroke and this information can be used in future data analysis projects. The top three important variables for the whole data set model were strangely average glucose level as the topmost important variable, we have statistically proven that glucose level has no to little impact on stroke prediction, secondly age which can be agreed upon and thirdly BMI which again seems to have less impact on stroke prediction. The model with over 40 dataset made all three mentioned variables important predictors but in a slightly different order. And finally, the oversampled data set model Selected age as the main predictor of stroke and all other variables had little impact on stroke prediction. What we can deduct from the results is that the data set we have does have mostly the right predictors(variables) but with a small percentage of those variables having positive conditions, we had a small percentage of the dataset that heart disease or hypertension. If we want to make predictions on such important health issues, then we need a larger dataset with patients that are with health conditions. Especially the target variable having the highest percentage of patients with i.e. stroke then we will have better results those models can even be deployed in real-life settings.

The results of the analysis may not be fully reliable as they depend on data quality, algorithm choice and parameters, and data size. To improve reliability of the results, future analysis should have more diverse and comprehensive data, multiple different algorithms, and a larger data size. Alternative interpretations of the results may include different variable selection or different algorithms that may lead to different results and conclusions.

For non-technical team

We analysed patient dataset to predict stroke using statistical methods and machine learning. Our study found that age, smoking status, hypertension, marital status, work type, heart disease, and BMI had significant associations with the target variable (stroke). We used three different modified datasets to test our prediction models, each with different varying amounts of stroke patients. The best-performing model was the oversampled data set, which had 50% of patients with stroke and 50% without. The results showed that age was the most important predictor of stroke, while other variables had little impact. However, these results were based on a small dataset, so larger datasets with more patients with health conditions would likely lead to better results and be more useful in real-life settings.

Reflection:

From this project, I have learned the importance of having a clean and processed dataset for accurate prediction results. I also realised the impact of having a balanced dataset with sufficient patients with the target condition. My motivation for this project was to understand how health data sets work and familiarise myself so that I may eventually enter and contribute to the healthcare industry, by using what I have and yet to learn, it can have an enormous effect on early intervention and prevention of health conditions. A challenge faced was the limitations of the dataset and the need for larger datasets with sufficient patients with the target condition. If I had more time and knowledge, I could have explored other predictive models, I did attempt glm for predictive models and had issues with data structure, which I could not resolve or tried to balance the dataset differently, here again, I attempted to learn SMOTE and ADASYN which require a bit more time than I had to understand the concept and apply it. This same research question can be addressed with different target variables such as hypertension, or heart disease, another question I would have proposed is what is the top 3 predictor variable for this condition to narrow it down.

References

Anova.glm: Analysis of deviance for generalized linear model fits (no date) RDocumentation. Available at: https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/anova.glm (https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/anova.glm) (Accessed: January 25, 2023).

What is Random Forest? (no date) IBM. Available at: https://www.ibm.com/topics/random-forest (https://www.ibm.com/topics/random-forest) (Accessed: January 28, 2023).

Bhalla, D. (no date) A complete guide to Random Forest in R, ListenData. Available at: https://www.listendata.com/2014/11/random-forest-with-r.html (https://www.listendata.com/2014/11/random-forest-with-r.html) (Accessed: January 28, 2023).

Song, L., Langfelder, P. & Horvath, S. Random generalized linear model: a highly accurate and interpretable ensemble predictor. BMC Bioinformatics 14, 5 (2013). https://doi.org/10.1186/1471-2105-14-5 (https://doi.org/10.1186/1471-2105-14-5)

Zach (2022) How to create a confusion matrix in R (step-by-step), Statology. Available at: https://www.statology.org/confusion-matrix-in-r/ (https://www.statology.org/confusion-matrix-in-r/) (Accessed: January 29, 2023).

Aditya_Sharma and Kaushik_Roy_Chowdhur (2015) How to extract important variables from random forest model using varimpplot in R?, Data Science, Analytics and Big Data discussions. Available at: https://discuss.analyticsvidhya.com/t/how-to-extract-important-variables-from-random-forest-model-using-varimpplot-in-r/1325 (https://discuss.analyticsvidhya.com/t/how-to-extract-important-variables-from-random-forest-model-using-varimpplot-in-r/1325) (Accessed: January 28, 2023).