Exploratory Data Analysis (for) Stroke Data Set

Pascal Visser

2022-09-14

Contents

Loading libraries

```
# load libraries
library(tidyverse)
library(naniar)
library(readr)
```

```
library(ggplot2)
library(lemon)
library(knitr)
library(pander)
library(cowplot)
library(dplyr)
```

1. Intro

A stroke is a life-threatening medical condition, where there is no or poor blood flow to (parts of) the brain. This causes parts of the brain to stop functioning and die off in minutes time. A stroke or brain attack is labelled as the second leading cause of death worldwide and is responsible for an annual mortality of about 5.5 million.

This dataset consists of people that had a stroke and people that have not. The variables are values that indicate their lifestyle, health and environment factors. The goal of this research is to calculate the chance of a stroke based of their lifestyle. This can be accomplished by machine learning. This gives the following research question:

Is it possible to produce an accurate algorithm with machine learning, to calculate the risk of a stroke based on lifestyle variables of people with stroke history and people that never had a stroke?

2. Dataset

The data is as said, lifestyle information of people with stroke history or not. In this section the data will be talked trough and the variables explained. First the data is loaded in:

```
# load in the data
strokedata <- read.csv("Data/Stroke_dataset.csv", header = T)
kable(strokedata[1:6, 1:6], format = 'simple', caption = "Head of data")
```

Table 1: Head of data

id	gender	age	hypertension	heart_disease	ever_married
9046	Male	67	0	1	Yes
51676	Female	61	0	0	Yes
31112	Male	80	0	1	Yes
60182	Female	49	0	0	Yes
1665	Female	79	1	0	Yes
56669	Male	81	0	0	Yes

```
cat("The dataset is", dim(strokedata)[1], "rows long, and has", dim(strokedata)[2], "columns")
```

```
## The dataset is 5110 rows long, and has 12 columns
```

Above, the first six rows of the data are shown. Also is given how many records there are. The codebook will clarify the variables and their values. The codebook will be loaded in:

```r
# load codebook
codebook <- read.table("Data/codebook.txt", header = T, sep = ';')
kable(codebook, format = 'pipe', caption = "Codebook of stroke data")
```

Table 2: Codebook of stroke data

Column	Description	value	datatype
id	unique identifier of patient	number	int
gender	male of female	sex	chr
age	age of patient	number	dbl
hypertension	suffering from hypertension?	binairy	int
heart_disease	suffering from heart disease?	binary	int
ever_married	married or married in the past?	binary	chr
work_type	sort of work	string	chr
Residence_type	type of residence	string	chr
avg_glucose_level	average level of glucose	number	dbl
bmi	body mass index	number	chr
smoking_status	smoking history	string	chr
stroke	stroke history	binary	int

The codebook gives a detailed description the columns of the dataset and column content about the value/datatypes. This is necessary to understand the data.

Now that the data is loaded in, there can be looked at the negatives of the data. The biggest flaws are the datatypes of the columns that are not equal. For example: the columns age is the type double instead of integer. The column BMI is character, while a double datatype is much more logical. Also, the columns hypertension and hearth disease are a yes or no question, in these columns the yes or no is represented by 0 or 1. The columns of marriage history is a yes or no question too, but here yes or no is represented by text. This also needs modifying.

The character variables are can be converted to factor classes.

Additionally, it's good to know how many NA's are present in the dataset. These missing values can screw the data. Also, values like 'unknown' can be declared as a missing value

```r
# check unique values of character values

cat("Gender:", unique(strokedata$gender),"\n",
    "Married:", unique(strokedata$ever_married), "\n",
    "Work type:", unique(strokedata$work_type), "\n",
    "Residence type:", unique(strokedata$Residence_type), "\n",
    "Smoking:", unique(strokedata$smoking_status))
```

```
## Gender: Male Female Other
##  Married: Yes No
##  Work type: Private Self-employed Govt_job children Never_worked
##  Residence type: Urban Rural
##  Smoking: formerly smoked never smoked smokes Unknown
```

Above reveals the Smoking status variable has a value "Unknown". This need to be addressed in the modifying section. Also, the column Gender has one other record. For the N/A the scan function of the 'naniar' package is used.

```r
# scan for missing values
nas <- miss_scan_count(data = strokedata, search = list("N/A", "Unknown"))
kable(nas, format = 'pipe', caption = "Number of missing values per variable")
```

Table 3: Number of missing values per variable

Variable	n
id	0
gender	0
age	0
hypertension	0
heart_disease	0
ever_married	0
work_type	0
Residence_type	0
avg_glucose_level	0
bmi	201
smoking_status	1544
stroke	0

The column BMI is responsible for 201 missing values, these missing values are likely responsible for the column to be parsed as character datatype. the column smoking_status has 1544 missing values. 1544 values is a big number and need to be dealt with in the modifying.

3 Dataset modifying

As talked in the previous section, some column types makes no sense, these types need to be changed to more sensible types. The columns that will be converted are:

- age, dbl -> int
- bmi, chr -> dbl

And additionally, the contents of the column ever_married will be converted to binary, and the smoking_status unknown will be cleared.

First the column age will be cast to integer, because a double type with decimals makes no sense in age, secondly, the column bmi will be cast to double, to keep the decimal precision:

```r
# cast columns to another type
strokedata <- transform(strokedata, age = as.integer(age))
strokedata <- transform(strokedata, bmi = as.double(bmi))
```

Now the ever_married column need to be changed to binary and a integer datatype:

```r
# replace yes and no by 1 or 0
strokedata$ever_married[strokedata$ever_married == "Yes"] <- "1"
strokedata$ever_married[strokedata$ever_married == "No"] <- "0"
strokedata <- transform(strokedata, ever_married = as.integer(ever_married))
```

The character classes will be cast to factor:

```r
strokedata$Residence_type <- as.factor(strokedata$Residence_type)
strokedata$gender <- as.factor(strokedata$gender)
strokedata$work_type <- as.factor(strokedata$work_type)
```

The record with gender 'Other' will be replaced with an gender value

```r
# seek the most present value inside the column
table(strokedata$gender)
```

```
##
## Female   Male  Other
##   2994   2115      1
```

```r
# remove record where gender is other
strokedata$gender[strokedata$gender == 'Other'] <- "Male"
```

It has been chosen to replace the other value with male, because of the balance inside the variable.

Now what is left are the missing values, BMI has N/A and smoking status has "unknown". For BMI the Na's will be replaced with the mean of the column:

```r
# replace the missing values with the column mean
strokedata$bmi[is.na(strokedata$bmi)] <- mean(strokedata$bmi, na.rm = T)

# keep the digit precision as before
strokedata$bmi <- round(strokedata$bmi, digits = 1)
```

The smoking status 'unknown' is harder to replace because the variables are character, first we need to know how many records there are of the column:

```r
# Count the unique variables in the gender column
table(strokedata$smoking_status)
```

```
##
## formerly smoked    never smoked          smokes         Unknown
##             885            1892             789            1544
```

Their are 885, 1892 and 789 records of not unknowns. Based on these numbers, we can calculate the probability of occurring in the smoking status variable.

```r
# Calculate the probability of formerly smoker, current smokers and non-smokers given that there's only

prob.Formerly <- 885 / (885 + 1892 + 789)
prob.Never <- 1892 / (885 + 1892 + 789)
prob.Smoke <- 789 / (885 + 1892 + 789)
```

With the probabilities, we can replace the unknowns based on there weight in the column.

```r
# Replacing 'Unknown' in smoking_status by the other 3 variables according to their weightage
strokedata$rand <- runif(nrow(strokedata))

strokedata <- strokedata%>%mutate(Probability = ifelse(rand <= prob.Formerly, "formerly smoked", ifelse

strokedata <- strokedata%>%mutate(smoking.status = ifelse(smoking_status == "Unknown", Probability, smol

# View the new Smoking Status column's unique values and their counts
table(strokedata$smoking.status)
```

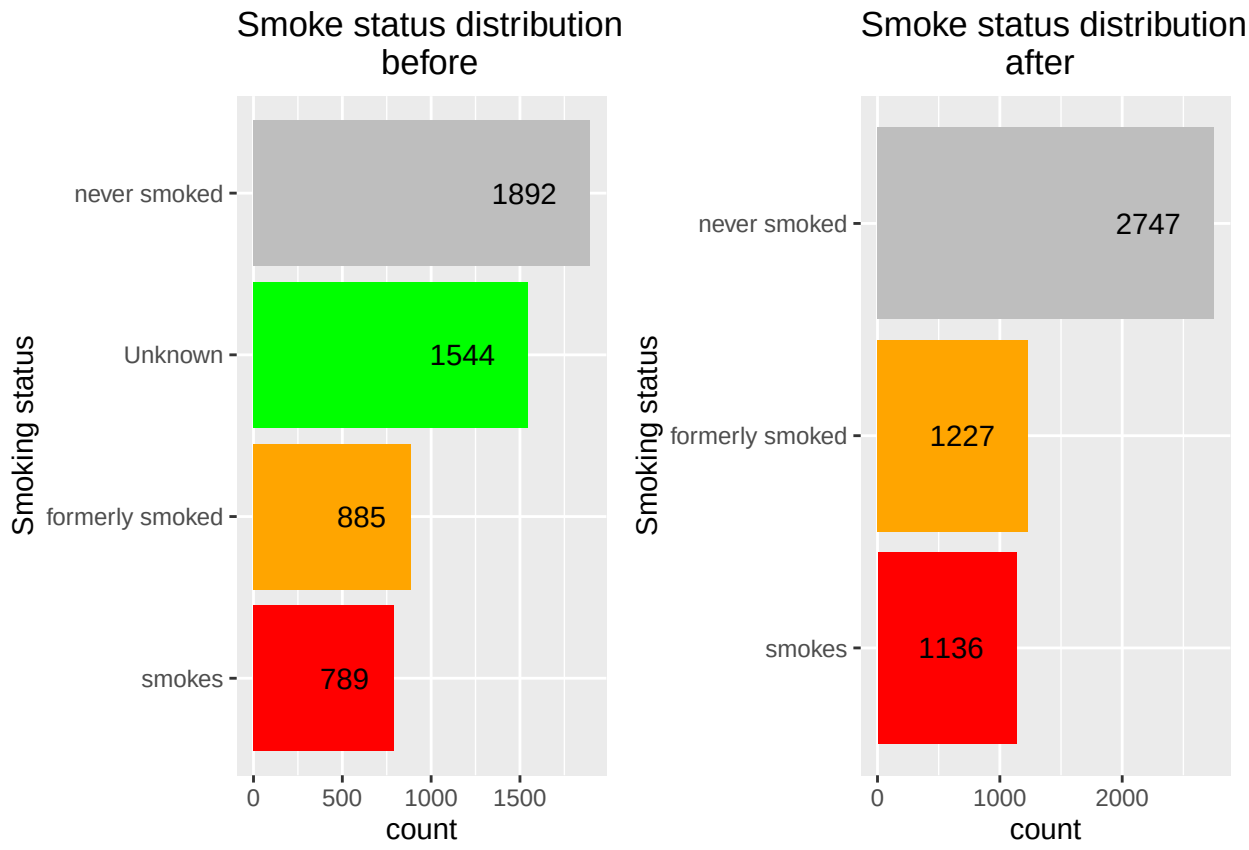
```
##
## formerly smoked    never smoked          smokes
##            1227            2747            1136
```

```r
# Remove columns that are not needed
health <- subset(strokedata, select = -c(rand,Probability,smoking_status))

# revise the column name of smoking status
colnames(health)[12] <- "smoking_status"

# 'health' is the final modified dataset which will be used for the EDA section below.
```

Now we have a column without the unknowns, but with increased other variables based on there weight. Which is shown in the plots below:



In the plots, the effect is shown of replacing the unknown value by other values based on weight. With this method, the distribution of values is not affected.

4. Exploratory Data analysis

In this part, the modified data is visualised into useful graphs. These graphs will tell about the distribution of variables and the relations to each other.

4.1 Summary

First the summary of the values numeric values, we have three columns with important numeric values, age, glucose level and BMI. The summary of these variables tells about the distribution of values and possible outliers

```
#summary of subset data
pander(summary(health[c(3,9:10)],), caption = "Summary of numeric values")
```

Table 4: Summary of numeric values

age	avg_glucose_level	bmi
Min. : 0.00	Min. : 55.12	Min. :10.30
1st Qu.:25.00	1st Qu.: 77.25	1st Qu.:23.80
Median :45.00	Median : 91.89	Median :28.40
Mean :43.22	Mean :106.15	Mean :28.89
3rd Qu.:61.00	3rd Qu.:114.09	3rd Qu.:32.80
Max. :82.00	Max. :271.74	Max. :97.60

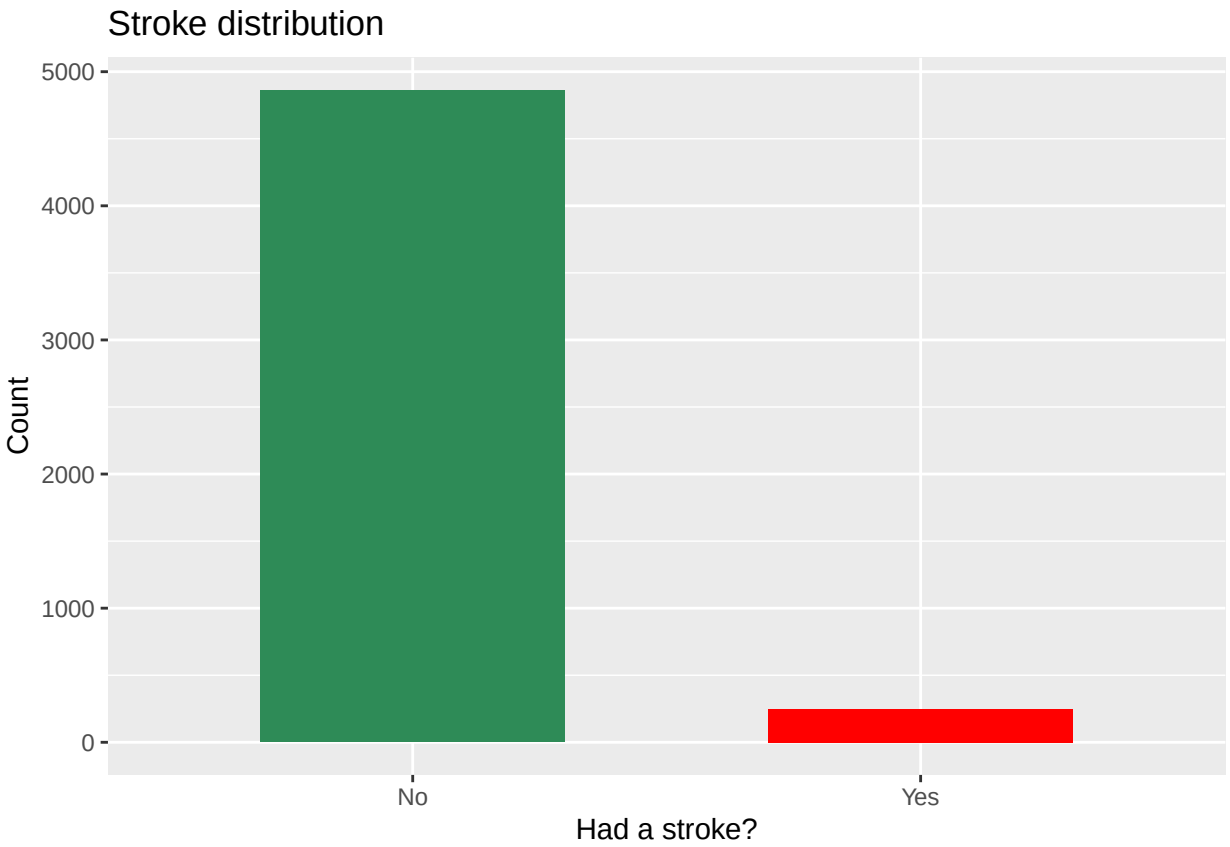
Looking at age, the minimum value is 0, which is a bit odd. maybe this is a unknown. Glucose looks normal, but BMI has a very low minimum, which is also odd.

4.2 Distribution of classifier

The research is about classifying patients changes of stroke based on there lifestyle and health. The figure below is the distribution of the stroke incidents in the data.

```
# Plot stroke distribution
fig1 <- ggplot(health, aes(x=factor(stroke))) +
  geom_bar(width = 0.6, fill = c("seagreen", "red")) +
  labs(title = "Stroke distribution", x = "Had a stroke?", y  = "Count")

# plot the graph
fig1 + scale_x_discrete(breaks=c("0", "1"), labels=c("No", "Yes"))
```

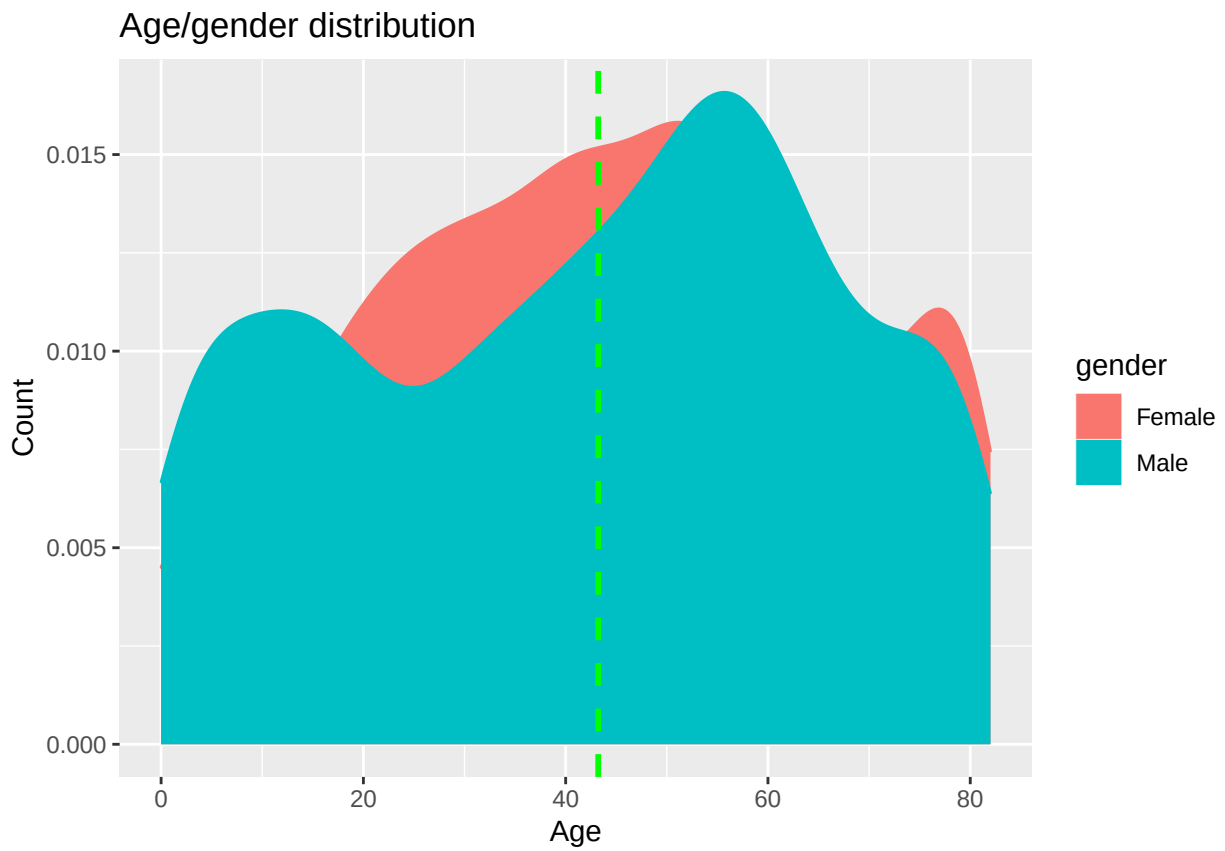


The number of stroke in the dataset is very low. the vast majority of the data didn't had a stroke. this needs to be balanced later.

4.3 Age and gender distribution

There are 5110 people in the dataset from which there is data, the gender and age distribution is as follows:

```
# make age and gender plot
ggplot(health, aes(x=age, fill=gender, color=gender)) +
    geom_density() +
    labs(title = "Age/gender distribution", x = "Age", y  = "Count") +
    geom_vline(aes(xintercept=mean(age)), color = 'green', lwd = 1, linetype = 'dashed')
```



Most people are around 40 - 60 years old, with some more females round the 20- 40 mark.

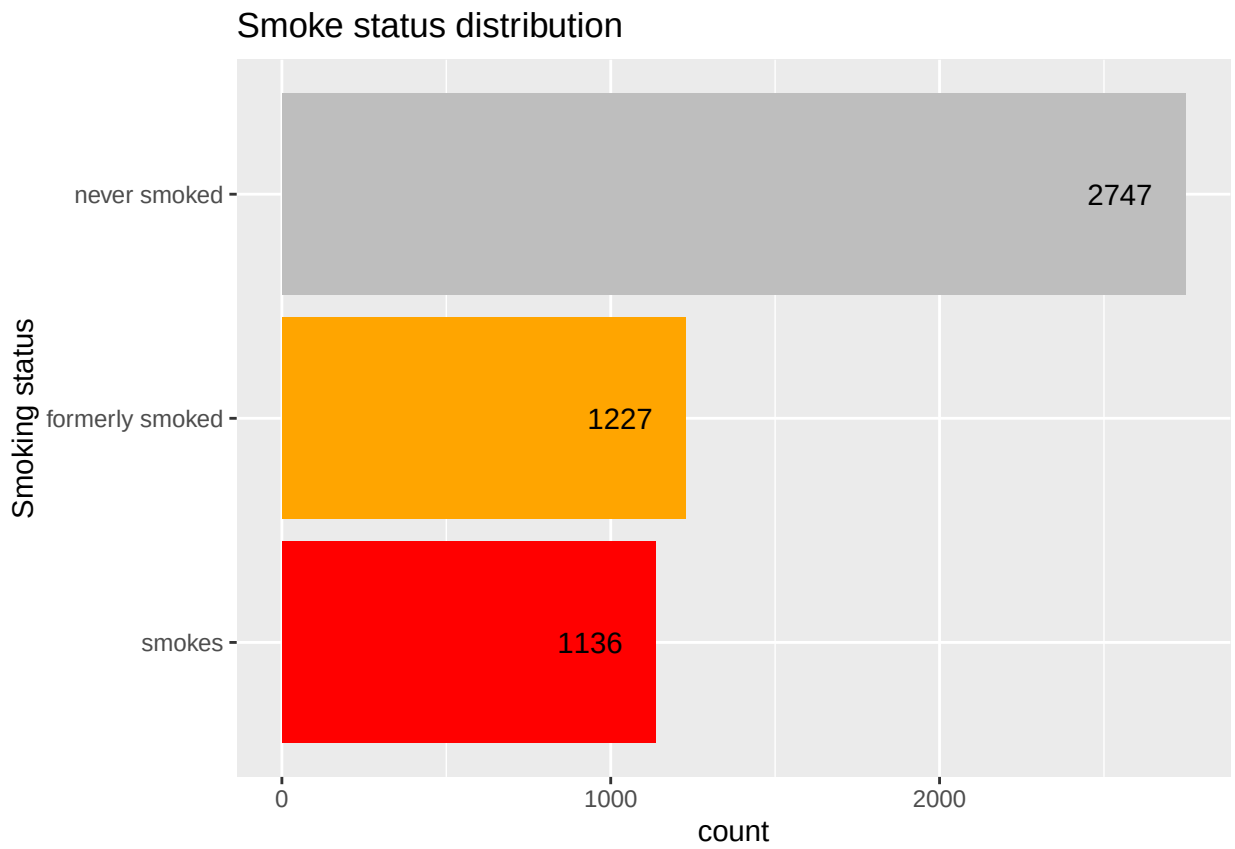
4.4 health related variables

This sections tells about the heath related variables such as, smoke, BMI, hypertension, heart disease and glucose levels

4.4.1 Smoke

Smoking can affect your health in a bad why, therefore it is good to know how many people (formerly) are smoking.

```
health %>%
  group_by(smoking_status) %>%
    summarise(count = length(smoking_status)) %>%
      mutate(smoking_status = factor(smoking_status)) %>%
        ggplot(aes(x = fct_reorder(smoking_status, count), y = count)) +
        geom_col(fill = c("Orange", "gray", "red")) +
        geom_text(aes(label = count, x = smoking_status, y = count), size = 4, hjust = 1.5) +
        coord_flip() +
        labs(x = "Smoking status", title = "Smoke status distribution")
```



Most of the people never have smoke, but a fairly amount of the people smokes or formerly smoked. This can indicate a bad overall health, which possible can contribute towards strokes. The term smokes is a bit too general, because it says nothing about how much someone smokes on a daily basis. The same applies to formerly smokes.

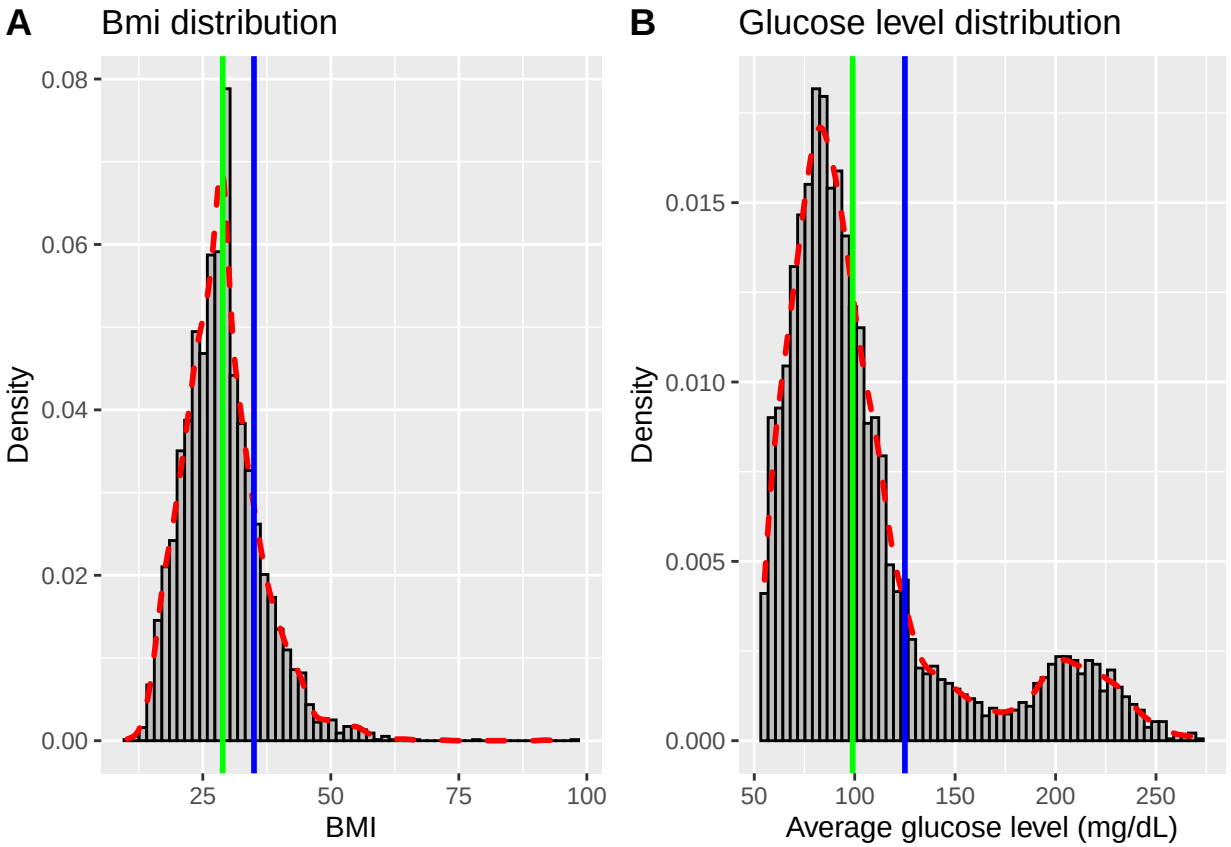
4.4.2 BMI and Glucose levels

BMI and glucose levels are also a good indication of general health. High BMI and glucose levels are considered bad for health.

```
fig2 <- ggplot(health, aes(x = bmi)) +
  geom_histogram(aes(y = ..density..), bins = 60, color = 1, fill = "gray") +
  geom_density(lwd = 1, linetype = 2, color = 'red') +
  labs(title = "Bmi distribution", x = "BMI", y = "Density") +
  geom_vline(aes(xintercept=mean(bmi)), color = 'green', lwd = 1) +
  geom_vline(xintercept = 35, color = 'blue', lwd = 1)

fig3 <- ggplot(health, aes(x = avg_glucose_level)) +
  geom_histogram(aes(y = ..density..), bins = 60, color = 1, fill = "gray") +
  geom_density(lwd = 1, linetype = 2, color = 'red') +
  labs(title = "Glucose level distribution", x = "Average glucose level (mg/dL)", y = "Density") +
  geom_vline(xintercept = 99, color = 'green', lwd = 1) +
  geom_vline(xintercept = 125, color = 'blue', lwd = 1)


plot_grid(fig2, fig3, labels = "AUTO", rel_widths = c(1,1))
```



BMI has a sort of normal distribution, with a mean of ~ 28. A BMI between 25 - 30 means overweight and 30+ means obese. Their is a fairy amount of people with a BMI higher than 35 (the blue line). This means that a reasonable amount of people in the dataset is serious overweight.

For glucose level, the distribution is a little bit right skewed with a bulge toward the end of the x axis. The mean of 106 is high. 99 mg/dL or lower is normal, 100 to 125 mg/dL indicates you have pre diabetes, and 126 mg/dL or higher indicates you have diabetes. The green line represents 99 mg/dL and the blue line 125 mg/dL. These borders shows that, according to the above information. All the people left of the green line are healthy and all the people right of the blue line have diabetes. so, their are some serious unhealty people according to these numbers.

```
table(health$stroke[health$bmi > 35 & health$avg_glucose_level > 125])
```

```
##
##   0   1
## 239  26
```

It tells that with the above information 265 people are serious unhealthy

4.4.3 Cardiovascular diseases

There are to cardiovascular diseases in the data, which are heart disease and hypertension.

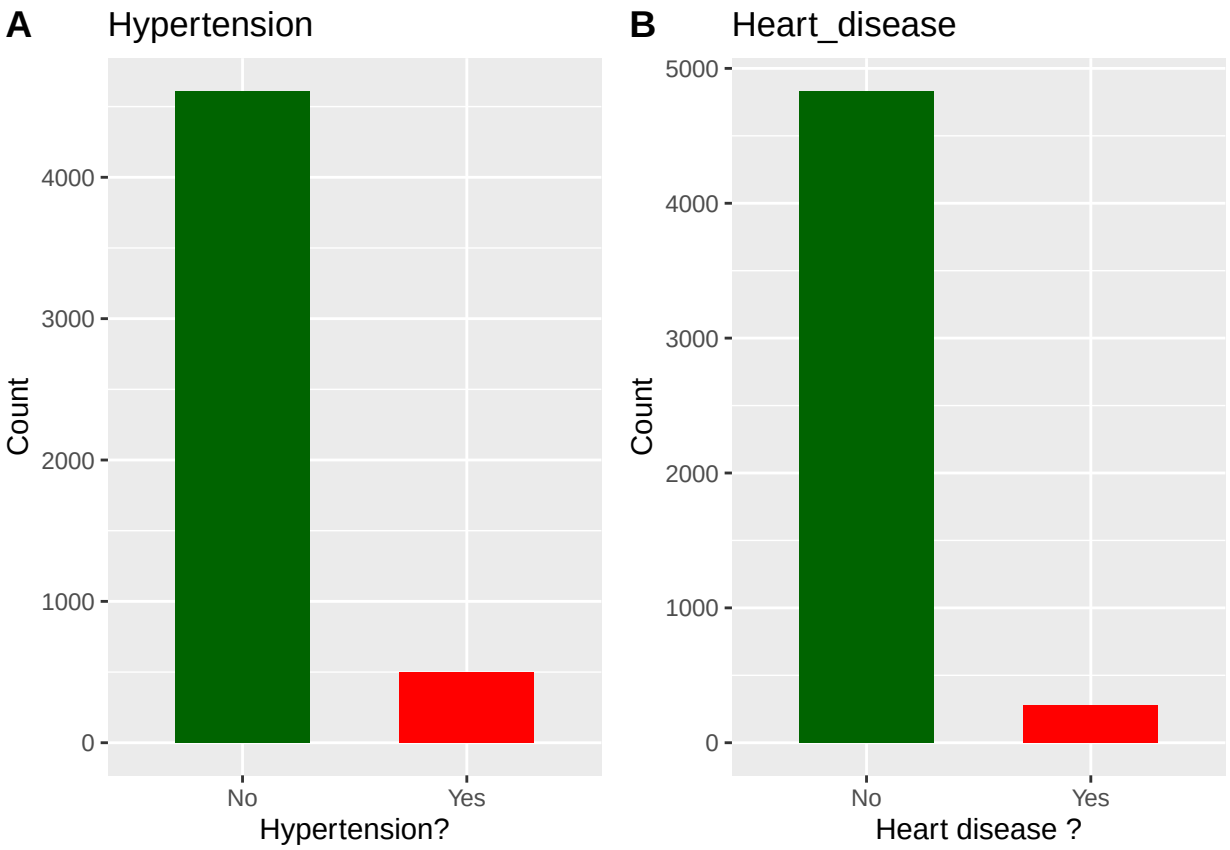
```
fig4 <- ggplot(health, aes(x=factor(hypertension))) +
  geom_bar(width = 0.6, fill = c("darkgreen", "red")) +
  labs(title = "Hypertension", x = "Hypertension?", y  = "Count")

fig4 <- fig4 + scale_x_discrete(breaks=c("0", "1"), labels=c("No", "Yes"))

fig5 <- ggplot(health, aes(x=factor(heart_disease))) +
  geom_bar(width = 0.6, fill = c("darkgreen", "red")) +
  labs(title = "Heart_disease", x = "Heart disease ?", y  = "Count")

fig5 <- fig5 + scale_x_discrete(breaks=c("0", "1"), labels=c("No", "Yes"))


plot_grid(fig4, fig5, labels = "AUTO")
```



The cardiovascular diseases are both overwhelming no, this is also imbalanced.

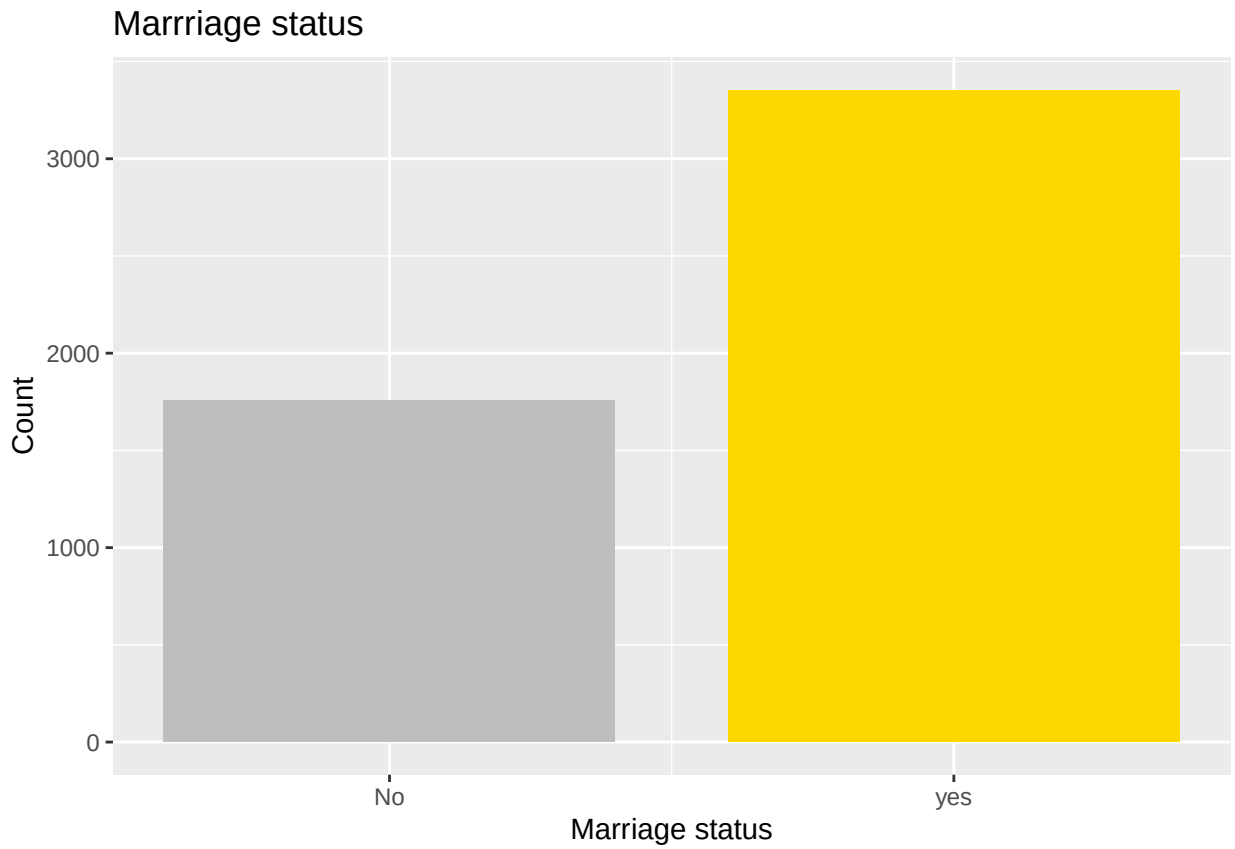
4.5 Lifestyle varibles

There are three lifestyle variables: married?, work type and residence type. These variables tells about the persons possible stress level. all these factor can contribute toward stress and possible health affection.

4.5.1 Marriage status

The below plot shows the distribution of marriage. A married life could be stressful.

```
ggplot(health, aes(x = ever_married)) +
  geom_bar(width = 0.8, fill = c("gray", "gold")) +
  scale_x_continuous(breaks = c(0,1), labels = c("No", "yes")) +
  labs(x = "Marriage status", y = "Count", title = "Marrriage status")
```



As is shown, most people have been married at some point in their lives. Their is a sizeable group with people that have not been married. This maybe could been young people, which would be logic. The code below tests this hypothesis:

```
cat("Mean age of not married: ", round(mean(health$age[health$ever_married == "0"]), 2), '\n')
```

```
## Mean age of not married:  21.98
```

```
cat("Mean age of ever married: ", round(mean(health$age[health$ever_married == "1"]), 2))
```

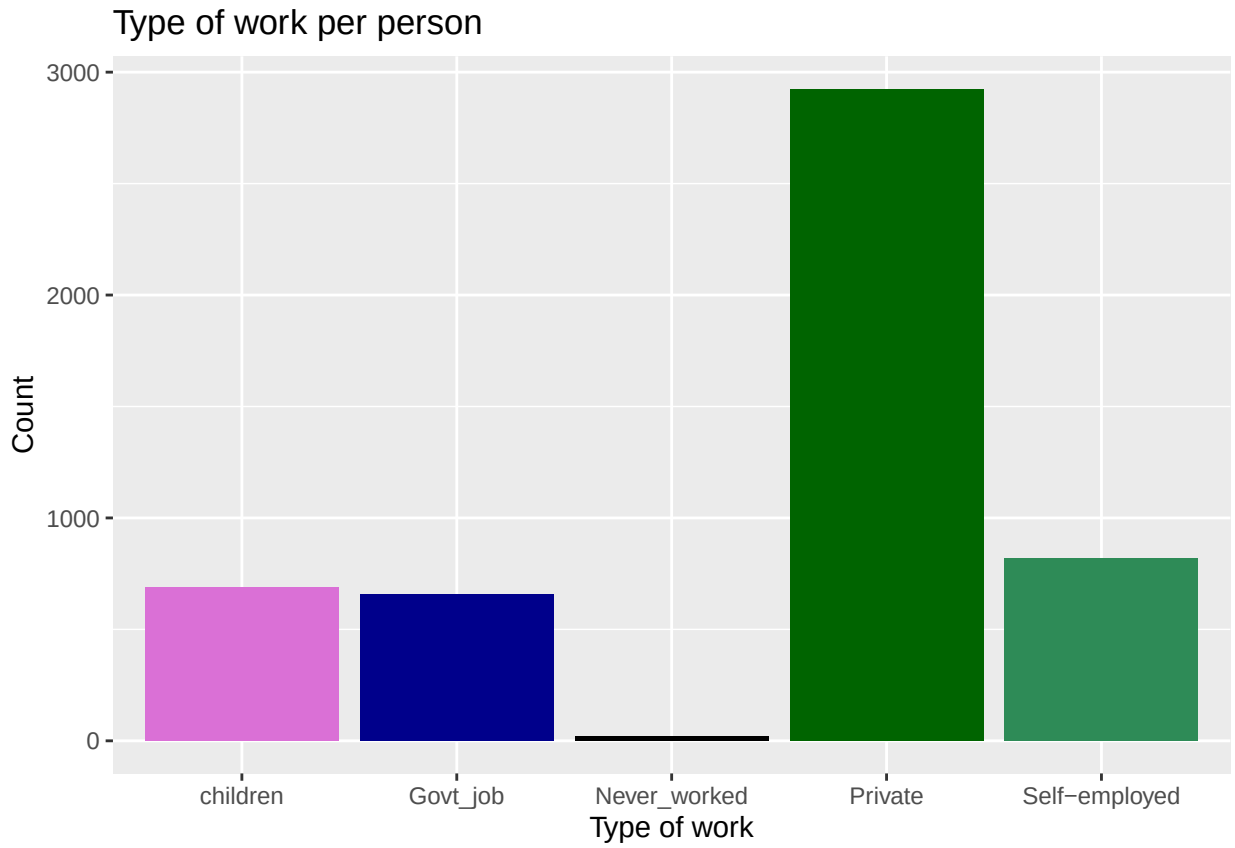
```
## Mean age of ever married:  54.34
```

As thought, the mean age is ~ 22 years old. Which means that this group mostly are young people.

4.5.2 Type of work

The type of work someone preforms can contribute toward higher stress levels. The plot beneath displays the distribution of work types:

```
ggplot(health, aes(x = work_type)) +
  geom_bar(position = 'identity', fill = c("orchid", "darkblue", "black", "darkgreen","seagreen")) +
  labs(x = "Type of work", y = "Count", title = "Type of work per person")
```



looking at the graph, most people work for a private company. Children, government job and self employed are somewhat balanced. Children would be mostly women is the guess.

```
table(health$gender[health$work_type == 'children'])
```

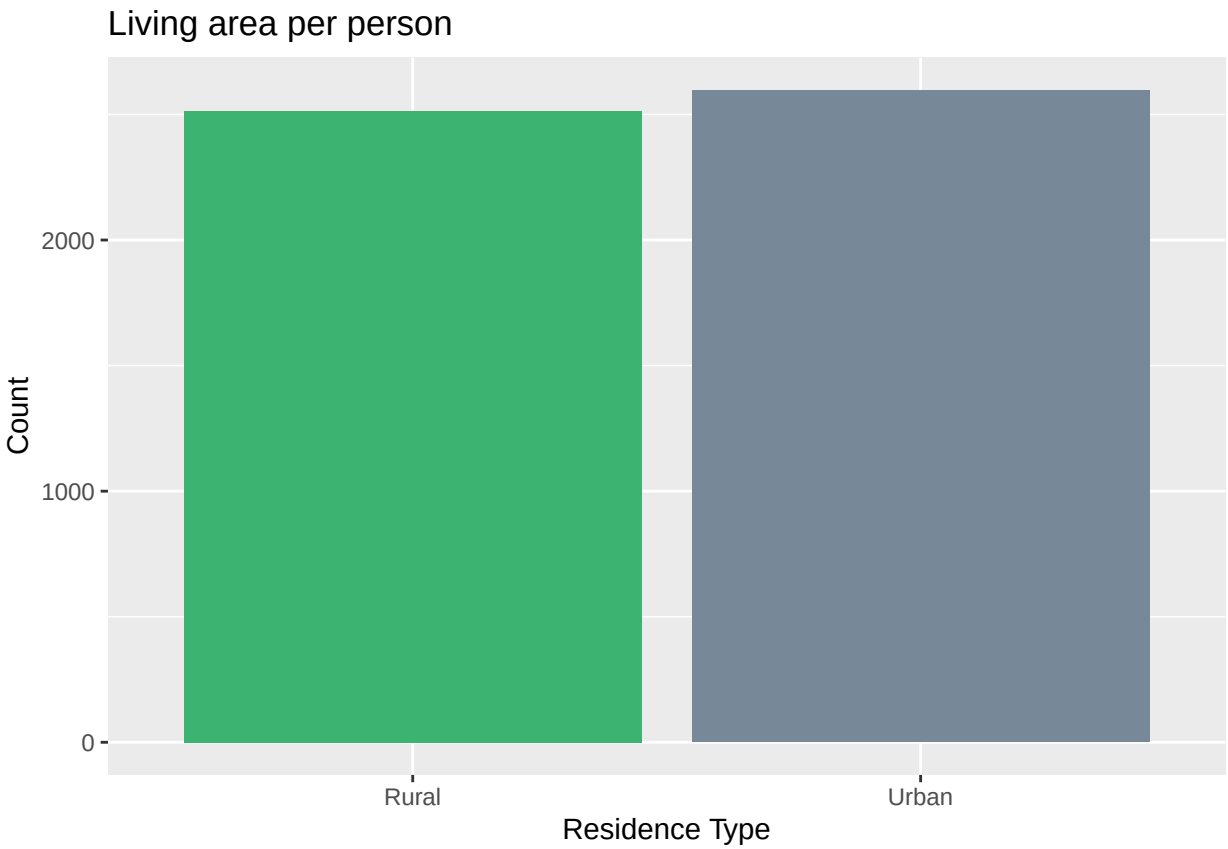
```
##
## Female   Male  Other
##    326    361      0
```

surprisingly, children is balanced between male and female. so the thought of extra stress on women with children (which possibly can lead to higher stroke chances) can be rejected.

4.5.2 Residence type

The type of living area can have influence on someone's life, living in a busy urban area maybe contributes to stress. The plot of residence type below shows the spread of living areas.

```
ggplot(health, aes(x =  Residence_type)) +
  geom_bar(position = 'identity', fill = c("mediumseagreen", "lightslategray")) +
    labs(x = "Residence Type", y = "Count", title = "Living area per person")
```



The bar plot of residence type shows a balance between the two types. There can be assumed that mostly young people live in urban areas and elderly people leave the busy areas.

```
cat("Mean age of urban: ", round(mean(health$age[health$Residence_type == "Urban"]), 2), '\n')
```

```
## Mean age of urban:  43.53
```

```
cat("Mean age of rural: ", round(mean(health$age[health$Residence_type == "Rural"]), 2))
```

```
## Mean age of rural:  42.89
```

The mean ages did not show any significant age difference between living rural or urban

5. Comparing variables

6. EDA conclusion

The data is in a good condition with several variables to work with for machine learning. The removal of the Na's made the quality of the data better, mainly trough the conversion of the term 'unknown' by smoking status. Also the casting of types helps al lot, by creating a simple but challenging dataset, But the data is still having one big flaw.

The plots shows that the distribution of certain variables skewed is, namely: stroke, hypertension and heart_disease. These variable have a very big imbalance inside. This imbalance can cause problems with machine learning because saying 'No' at these variables can give you a 95%+ accuracy. This imbalance need to be tackled before moving on into machine learning. By doing this, the dataset can be workable for machine learning.

6. Machine learning

// intro ml

// dataset changing

```
# relocate classifier
health <- health %>% relocate(stroke, .after = last_col())

# change binary values to true or false
health$stroke <- as.logical(health$stroke)
health$hypertension <- as.logical(health$hypertension)
health$heart_disease <- as.logical(health$heart_disease)
health$ever_married <- as.logical(health$ever_married)


write.csv(health, "data.csv", row.names = T)

table(health$work_type[health$stroke == "TRUE"])
```

```
##
##      children      Govt_job  Never_worked       Private Self-employed
##             2            33             0           149            65
```

// testing algorithms