INDIVIDUAL PROJECT

The dataset used in this assignment was sourced from the Kaggle website: (https://www.kaggle.com/datasets/omkargowda/suicide-rates-overview-1985-to-2021).

QUESTION A

Conduct descriptive summaries for Suicides dataset.

#The dataset is first loaded into R.

suicides = read.csv(file.choose(),header=T)

#First, let's have a glimpse into the first and last few observations of the dataset.

head(suicides)

```
> head(suicides)
  country year    sex         age suicides_no population suicides.100k.pop country.year HDI.for.year gdp_for_year.... gdp_per_capita....       generation
1 Albania 1987   male 15-24 years          21     312900              6.71  Albania1987           NA    2,15,66,24,900                796     Generation X
2 Albania 1987   male 35-54 years          16     308000              5.19  Albania1987           NA    2,15,66,24,900                796           Silent
3 Albania 1987 female 15-24 years          14     289700              4.83  Albania1987           NA    2,15,66,24,900                796     Generation X
4 Albania 1987   male   75+ years           1      21800              4.59  Albania1987           NA    2,15,66,24,900                796 G.I. Generation
5 Albania 1987   male 25-34 years           9     274300              3.28  Albania1987           NA    2,15,66,24,900                796          Boomers
6 Albania 1987 female   75+ years           1      35600              2.81  Albania1987           NA    2,15,66,24,900                796 G.I. Generation
```

tail(suicides)

```
> tail(suicides)
                       country year    sex       age suicides_no population suicides.100k.pop                country.year HDI.for.year gdp_for_year....
31751                   Turkey 2017 female 75+ years          NA   82089826       0.000000000                  Turkey2017    0.7953429         8.59E+11
31752                  Ukraine 2017 female 75+ years         256   44831135       0.571031717                 Ukraine2017    0.7854583         1.12E+11
31753           United Kingdom 2017 female 75+ years         104   66058859       0.157435356          United Kingdom2017    0.9147349         2.70E+12
31754 United States of America 2017 female 75+ years         501  325122128       0.154095940 United States of America2017    0.9186199         1.95E+13
31755                  Uruguay 2017 female 75+ years          14    3422200       0.409093566                 Uruguay2017    0.8167449      64233966861
31756               Uzbekistan 2017 female 75+ years           3   32388600       0.009262518              Uzbekistan2017    0.6912577      62081323299
      gdp_per_capita....      generation
31751          10589.668 G.I. Generation
31752           2638.326 G.I. Generation
31753          40857.756 G.I. Generation
31754          60109.656 G.I. Generation
31755          18690.894 G.I. Generation
31756           1916.765 G.I. Generation
```

#Then, let's take a look at the variables in the data as well as their types:

str(suicides)

```
> str(suicides)
'data.frame':	31756 obs. of  12 variables:
 $ country           : chr  "Albania" "Albania" "Albania" "Albania" ...
 $ year              : int  1987 1987 1987 1987 1987 1987 1987 1987 1987 1987 ...
 $ sex               : chr  "male" "male" "female" "male" ...
 $ age               : chr  "15-24 years" "35-54 years" "15-24 years" "75+ years" ...
 $ suicides_no       : int  21 16 14 1 9 1 6 4 1 0 ...
 $ population        : int  312900 308000 289700 21800 274300 35600 278800 257200 137500 311000 ...
 $ suicides.100k.pop : num  6.71 5.19 4.83 4.59 3.28 2.81 2.15 1.56 0.73 0 ...
 $ country.year      : chr  "Albania1987" "Albania1987" "Albania1987" "Albania1987" ...
 $ HDI.for.year      : num  NA NA NA NA NA NA NA NA NA NA ...
 $ gdp_for_year....  : chr  "2,15,66,24,900" "2,15,66,24,900" "2,15,66,24,900" "2,15,66,24,900" ...
 $ gdp_per_capita....: num  796 796 796 796 796 796 796 796 796 796 ...
 $ generation        : chr  "Generation X" "Silent" "Generation X" "G.I. Generation" ...
```

*#It seems like the variable **year** fits the type factor better than integer. The variables **country, sex, age** and **generation** also seem to fit the type factor better than character. Meanwhile, the variable **gdp_for_year....** was wrongly assigned as a character type instead of numeric.*

*#**year** will be changed into a factor type variable.*

suicides$year <- as.factor(suicides$year)

*#**gdp_for_year....** will have commas removed from its values in order to be changed into a numeric type variable.*

suicides$gdp_for_year.... <- as.numeric(gsub(",", "", suicides$gdp_for_year....))

#All character type variables in the suicide dataset will be changed into factor ones.

suicides[sapply(suicides,is.character)] <- lapply(suicides[sapply(suicides,is.character)], as.factor)

#Now that the changes to variable types have been made, let's check the properties of the fixed dataset!

summary(suicides)

```
> summary(suicides)
       country           year            sex                age          suicides_no        population        suicides.100k.pop       country.year  
 Austria    :  430   2009    : 1068   female:15878   15-24 years:5298   Min.   :    0.0   Min.   :2.780e+02   Min.   :  0.000   Albania1987:   12  
 Iceland    :  430   2001    : 1056   male  :15878   25-34 years:5298   1st Qu.:    3.0   1st Qu.:1.288e+05   1st Qu.:  0.370   Albania1988:   12  
 Mauritius  :  430   2010    : 1056                  35-54 years:5298   Median :   25.0   Median :5.468e+05   Median :  4.285   Albania1989:   12  
 Netherlands:  430   2000    : 1032                  5-14 years :5266   Mean   :  237.1   Mean   :7.217e+06   Mean   : 11.717   Albania1992:   12  
 Argentina  :  420   2002    : 1032                  55-74 years:5298   3rd Qu.:  132.0   3rd Qu.:2.909e+06   3rd Qu.: 14.560   Albania1993:   12  
 Belgium    :  420   2003    : 1032                  75+ years  :5298   Max.   :22338.0   Max.   :1.411e+09   Max.   :515.093   Albania1994:   12  
 (Other)    :29196   (Other):25480                                      NA's   :1200                                           (Other)    :31684  
  HDI.for.year   gdp_for_year....    gdp_per_capita....            generation  
 Min.   :0.378   Min.   :4.692e+07   Min.   :   251     Boomers        :5646  
 1st Qu.:0.727   1st Qu.:1.055e+10   1st Qu.:  3765     G.I. Generation:4056  
 Median :0.800   Median :5.585e+10   Median : 10062     Generation X   :7720  
 Mean   :0.794   Mean   :5.722e+11   Mean   : 17589     Generation Z   :1470  
 3rd Qu.:0.874   3rd Qu.:2.865e+11   3rd Qu.: 25622     Millenials     :5844  
 Max.   :0.975   Max.   :5.100e+13   Max.   :126352     Silent         :7020  
 NA's   :19456                                                                 
```

str(suicides)

```
> str(suicides)
'data.frame':	31756 obs. of  12 variables:
 $ country           : Factor w/ 114 levels "Albania","Antigua and Barbuda",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ year              : Factor w/ 36 levels "1985","1986",..: 3 3 3 3 3 3 3 3 3 3 ...
 $ sex               : Factor w/ 2 levels "female","male": 2 2 1 2 2 1 1 1 2 1 ...
 $ age               : Factor w/ 6 levels "15-24 years",..: 1 3 1 6 2 6 3 2 5 4 ...
 $ suicides_no       : int  21 16 14 1 9 1 6 4 1 0 ...
 $ population        : int  312900 308000 289700 21800 274300 35600 278800 257200 137500 311000 ...
 $ suicides.100k.pop : num  6.71 5.19 4.83 4.59 3.28 2.81 2.15 1.56 0.73 0 ...
 $ country.year      : Factor w/ 2649 levels "Albania1987",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ HDI.for.year      : num  NA NA NA NA NA NA NA NA NA NA ...
 $ gdp_for_year....  : num  2.16e+09 2.16e+09 2.16e+09 2.16e+09 2.16e+09 ...
 $ gdp_per_capita....: num  796 796 796 796 796 796 796 796 796 796 ...
 $ generation        : Factor w/ 6 levels "Boomers","G.I. Generation",..: 3 6 3 2 1 2 6 1 2 3 ...
```

#Afterwards, descriptive statistics will be conducted for the suicide dataset. Let's start with the dataset's numerical variables.

library(psych)
numerical_suvar <- suicides[sapply(suicides, function(x) is.integer(x) | is.numeric(x))]
describe(numerical_suvar)

```
                   vars     n         mean           sd       median      trimmed          mad         min         max        range  skew kurtosis
suicides_no           1 30556 2.371400e+02 8.679600e+02 2.500000e+01 7.322000e+01 3.706000e+01        0.00 2.23380e+04 2.233800e+04 10.62   167.38
population            2 31756 7.217454e+06 5.799323e+07 5.468325e+05 1.488271e+06 7.708519e+05      278.00 1.41110e+09 1.411100e+09 21.54   508.15
suicides.100k.pop     3 31756 1.172000e+01 2.159000e+01 4.280000e+00 7.280000e+00 6.350000e+00        0.00 5.15090e+02 5.150900e+02  7.36   119.85
HDI.for.year          4 12300 7.900000e-01 1.000000e-01 8.000000e-01 8.000000e-01 1.100000e-01        0.38 9.80000e-01 6.000000e-01 -0.49    -0.03
gdp_for_year....      5 31756 5.722487e+11 2.544261e+12 5.584969e+10 1.571756e+11 8.006663e+10 46919625.00 5.10000e+13 5.099995e+13 13.00   220.64
gdp_per_capita....    6 31756 1.758895e+04 1.946486e+04 1.006200e+04 1.404223e+04 1.149756e+04      251.00 1.26352e+05 1.261010e+05  1.92     4.56
                             se
suicides_no        4.970000e+00
population         3.254351e+05
suicides.100k.pop  1.200000e-01
HDI.for.year       0.000000e+00
gdp_for_year....   1.427739e+10
gdp_per_capita.... 1.092300e+02
```

#For categorical data, two methods will be used to retrieve their descriptive statistics.

#The first method is by using the prop table to see the frequency and proportional percentage of the categorical variable's values.

```
library(SmartEDA)
factor_suvar <- suicides[sapply(suicides, is.factor)]
View(ExpCTable(factor_suvar,Target=NULL,margin=1,clim=3000,round=2,bin=NULL,per=T))
```

	Variable	Valid	Frequency	Percent	CumPercent
1	country	Albania	264	0.83	0.83
2	country	Antigua and Barbuda	372	1.17	2.00
3	country	Argentina	420	1.32	3.32
4	country	Armenia	346	1.09	4.41
5	country	Aruba	168	0.53	4.94
6	country	Australia	408	1.28	6.22
7	country	Austria	430	1.35	7.57
8	country	Azerbaijan	192	0.60	8.17
9	country	Bahamas	276	0.87	9.04
10	country	Bahrain	252	0.79	9.83
11	country	Barbados	300	0.94	10.77
12	country	Belarus	300	0.94	11.71
13	country	Belgium	420	1.32	13.03
14	country	Belize	336	1.06	14.09
15	country	Bosnia and Herzegovina	24	0.08	14.17
16	country	Brazil	420	1.32	15.49
17	country	Brunei Darussalam	48	0.15	15.64
18	country	Bulgaria	408	1.28	16.92
19	country	Cabo Verde	12	0.04	16.96
20	country	Canada	396	1.25	18.21
21	country	Chile	420	1.32	19.53
22	country	China, Hong Kong SAR	48	0.15	19.68
23	country	Colombia	420	1.32	21.00
24	country	Costa Rica	408	1.28	22.28
25	country	Croatia	310	0.98	23.26
26	country	Cuba	336	1.06	24.32
27	country	Cyprus	226	0.71	25.03
28	country	Czech Republic	322	1.01	26.04
29	country	Czechia	48	0.15	26.19
30	country	Denmark	312	0.98	27.17

Showing 1 to 30 of 2,819 entries, 5 total columns

#The second method is calculating the modes of suicide's categorical variables by using a mode table.

```
varmode <- function(x){
  a = table(x)
  return(a[which.max(a)])
}
sapply(factor_suvar,varmode)
```

```
sapply(factor_suvar,varmode)
       country.Austria                year.2009             sex.female          age.15-24 years country.year.Albania1987   generation.Generation X
                   430                     1068                  15878                     5298                       12                      7720
```

QUESTION B

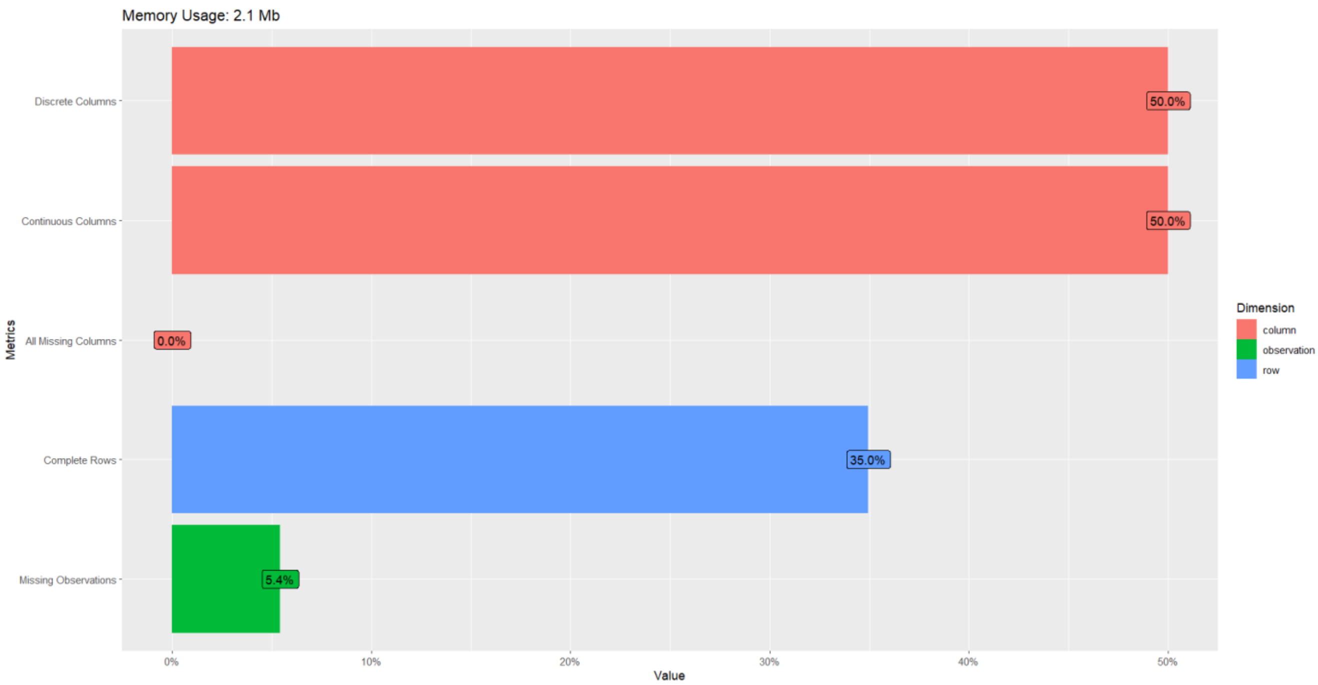
Produce suitable data representations and visualizations to describe the shape and pattern of the data.

#Let's look at the basic details of the dataset (e.g. its numbers of rows and columns etc.).

```
library(DataExplorer)
introduce(suicides)
```

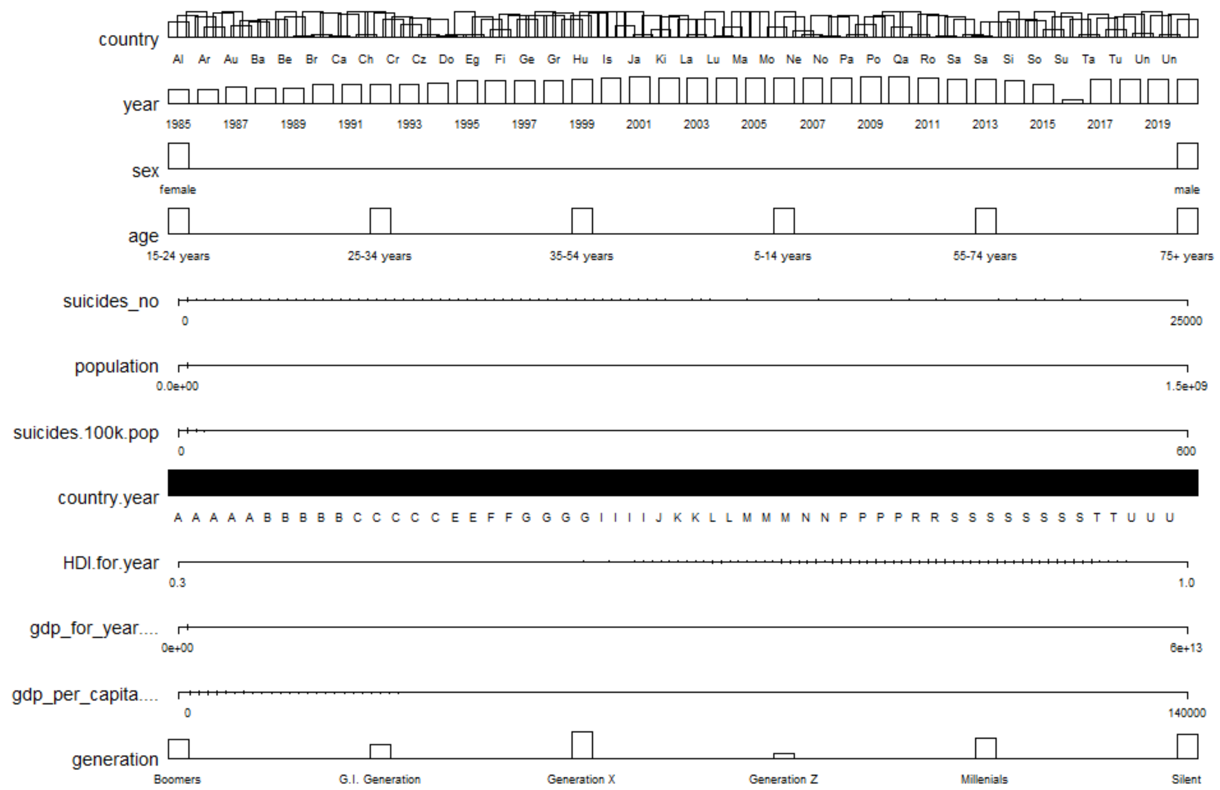
```
> introduce(suicides)
   rows columns discrete_columns continuous_columns all_missing_columns total_missing_values complete_rows total_observations memory_usage
1 31756      12                6                  6                   0                20656         11100             381072      2246008
```

```
plot_intro(suicides)
```



#Then, let's have a quick glance at the distributions of the values in suicide's columns.

```
library(Hmisc)
datadensity(suicides)
```



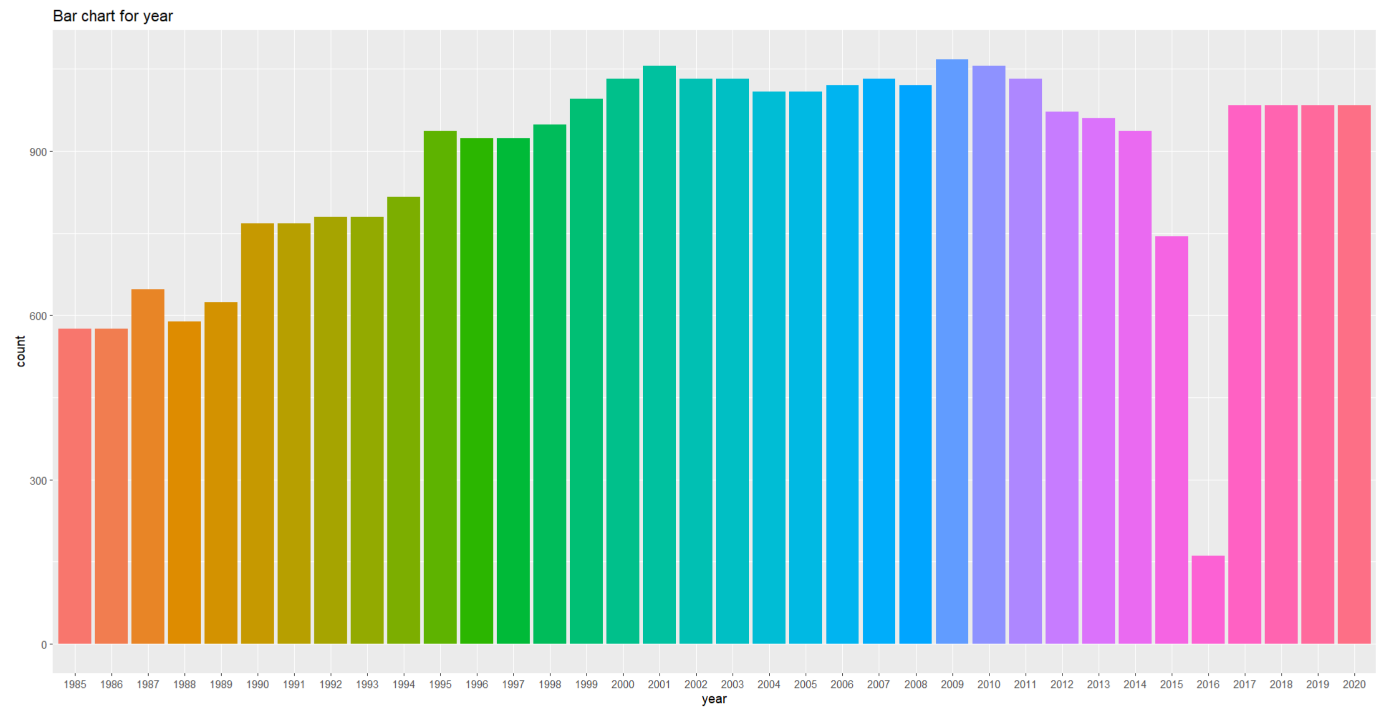
#DATA VISUALISATION

```
attach(suicide)
library(ggplot2)
library(dplyr)
```

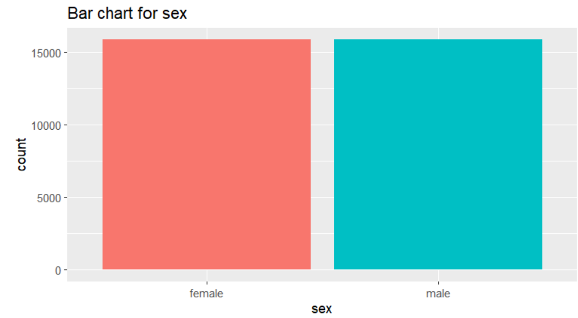
#VISUALISATION OF CATEGORICAL DATA

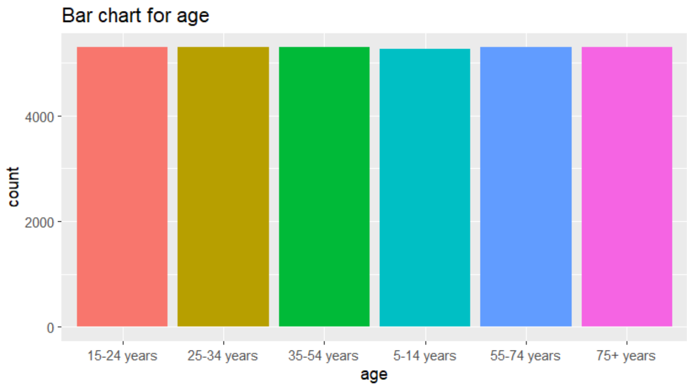
#BARCHARTS

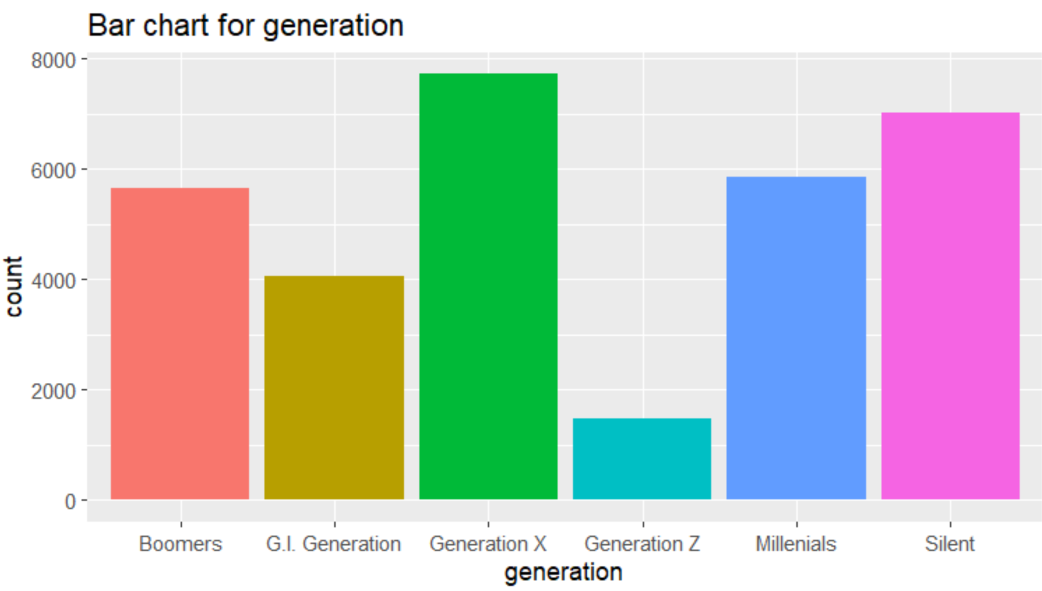
```
facvarcols <- names(select(factor_suvar,-matches("country")))
#Country-related variables have too many unique values to be properly visualised in a barchart.
barchart <- function(var) {
  ggplot(suicides, aes_string(x = var)) +
    geom_bar(aes_string(fill=var), stat = "count") +
    ggtitle(paste("Bar chart for",var))+
    theme(legend.position = "none")
}

lapply(facvarcols, barchart)
```



#variables sex and age have equal distribution

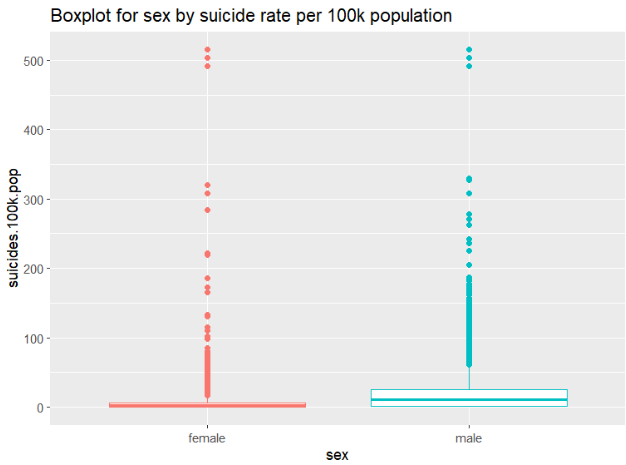


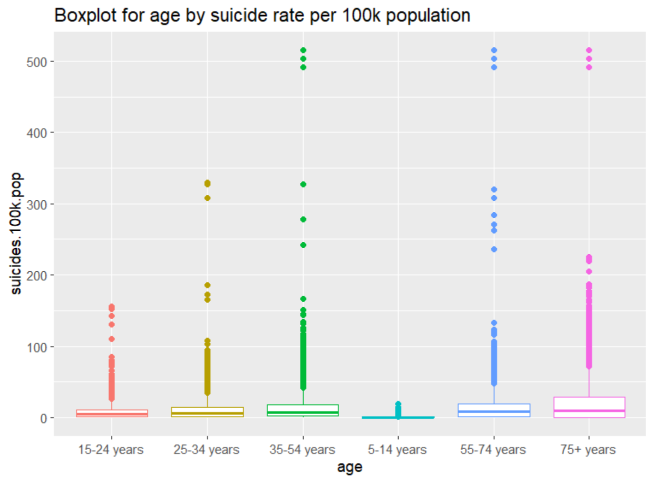


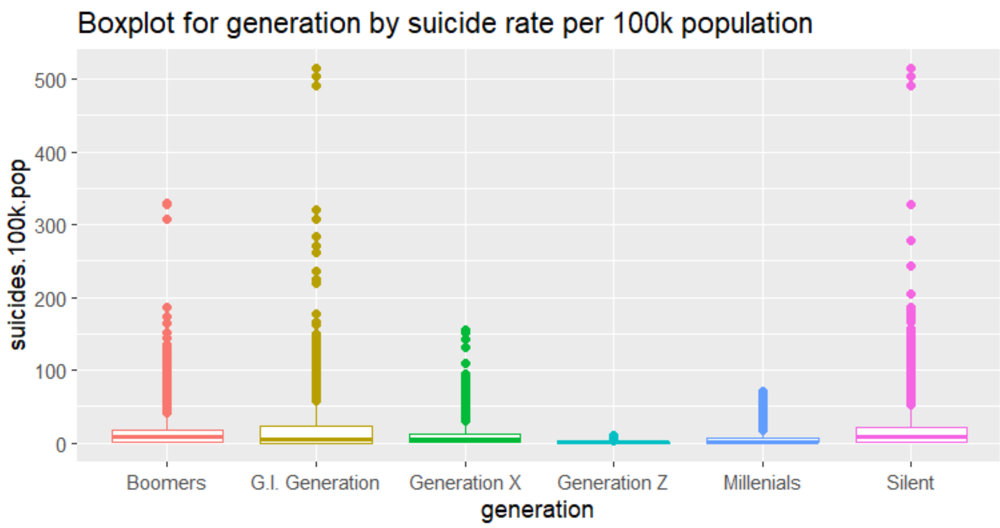


#BOX PLOTS

```
boxplots <- function(var) {
  ggplot(suicides, aes_string(x = var, y= "suicides.100k.pop")) +
    geom_boxplot(aes_string(color=var)) +
    ggtitle(paste("Boxplot for",var,"by suicide rate per 100k population")) +
    theme(legend.position = "none")
}

lapply(c("sex","age","generation"), boxplots) #year has been excluded for having no bearing on y
```





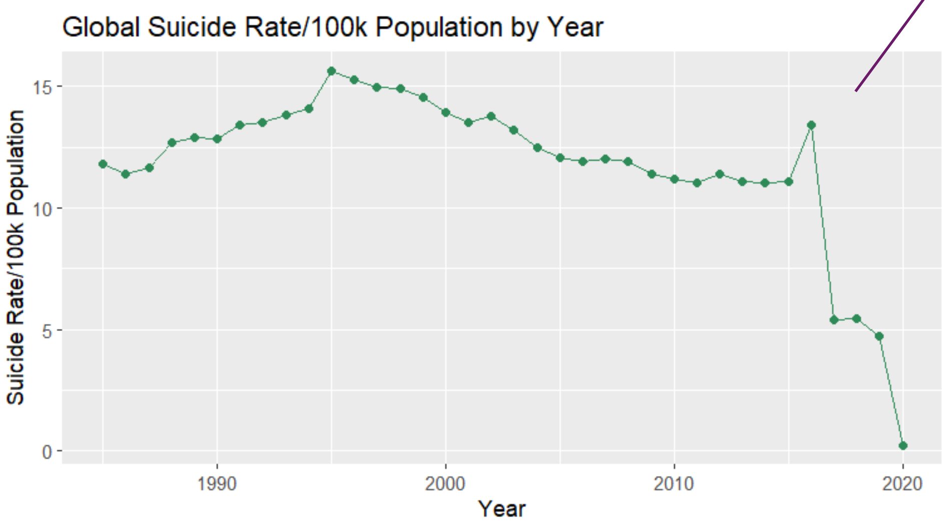


#BRIEF EXPLORATORY DATA ANALYSIS OF CATEGORICAL DATA
#PART 1: SUICIDE RATES PER 100K POPULATION, YEAR BY YEAR

```
totalyr <- suicides %>%
  group_by(year) %>%
  summarise(yearly = mean(suicides.100k.pop))

ggplot(totalyr, aes(x=as.numeric(as.character(year)), y=yearly)) +
```

```
  geom_line(color="seagreen") +
  geom_point(color="seagreen") +
  ggtitle("Global Suicide Rate/100k Population by Year") +
  xlab("Year") + ylab("Suicide Rate/100k Population")
```

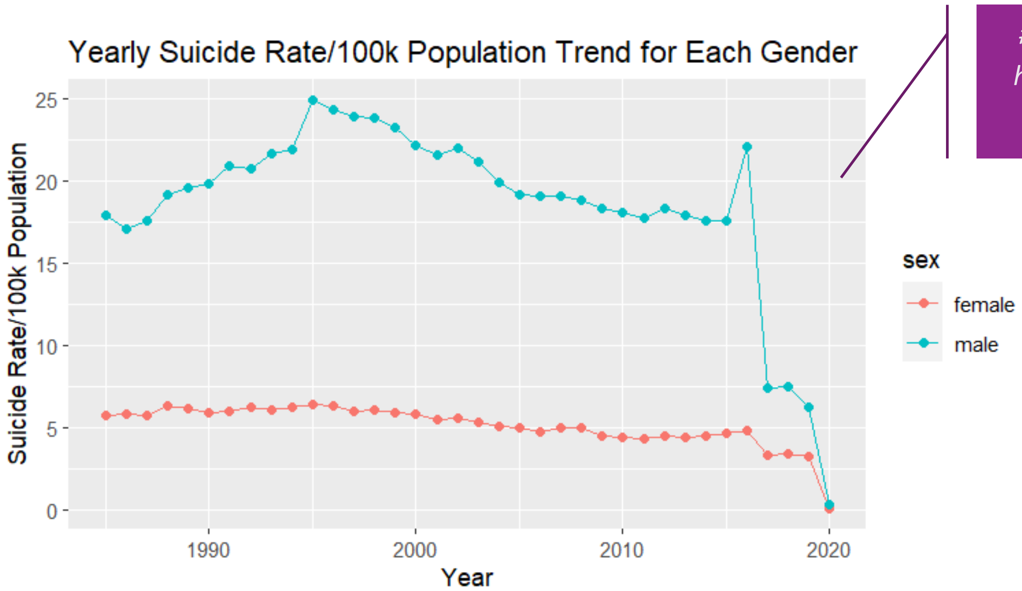
#The global suicide rate seems to decrease over the years.



#PART 2: YEARLY SUICIDE RATE/100K POPULATION TREND PER GENDER

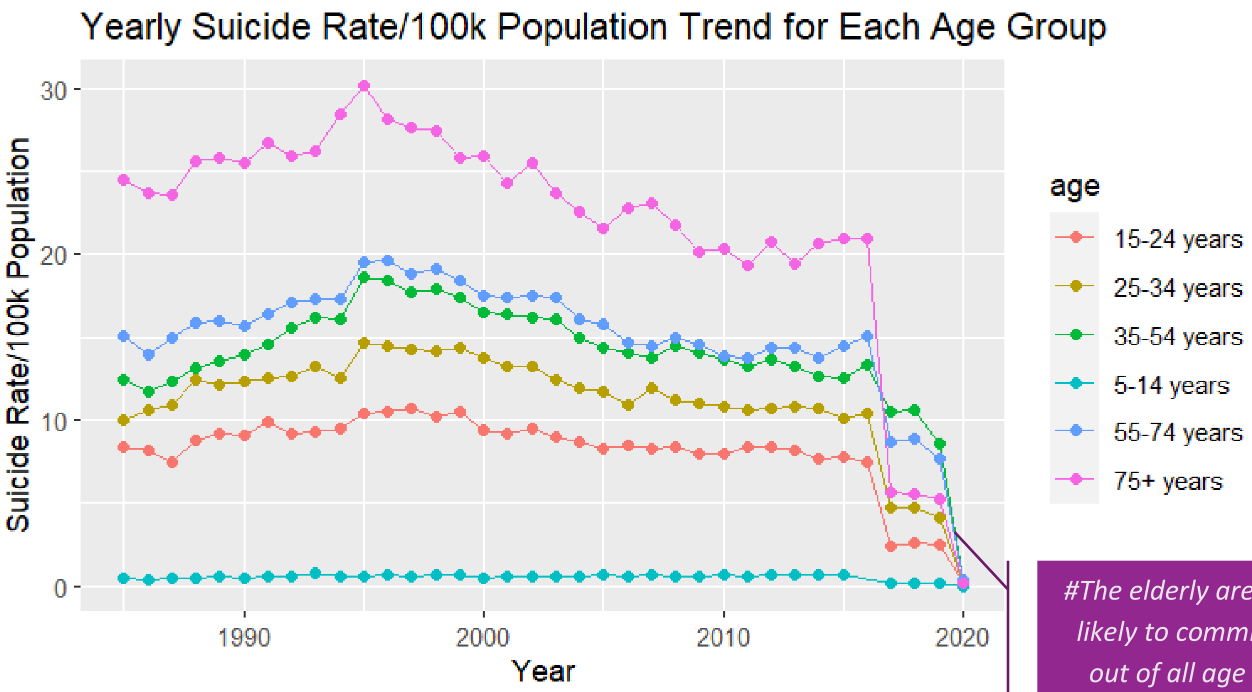
```
sexyr <- suicides %>%
  group_by(year, sex) %>%
  summarise(yearly = mean(suicides.100k.pop))

ggplot(sexyr, aes(x=as.numeric(as.character(year)), y=yearly, group=sex, color=sex)) +
  geom_line() +
  geom_point() +
  ggtitle("Yearly Suicide Rate/100k Population Trend for Each Gender") +
  xlab("Year") + ylab("Suicide Rate/100k Population")
```

#Men generally have a higher suicide rate than women.



#PART 3: YEARLY SUICIDE RATE/100K POPULATION TREND PER AGE GROUP

```
ageyr <- suicides %>%
  group_by(year, age) %>%
  summarise(yearly = mean(suicides.100k.pop))

ggplot(ageyr, aes(x=as.numeric(as.character(year)), y=yearly, group=age, color=age)) +
  geom_line() +
  geom_point() +
  ggtitle("Yearly Suicide Rate/100k Population Trend for Each Age Group") +
  xlab("Year") + ylab("Suicide Rate/100k Population")
```



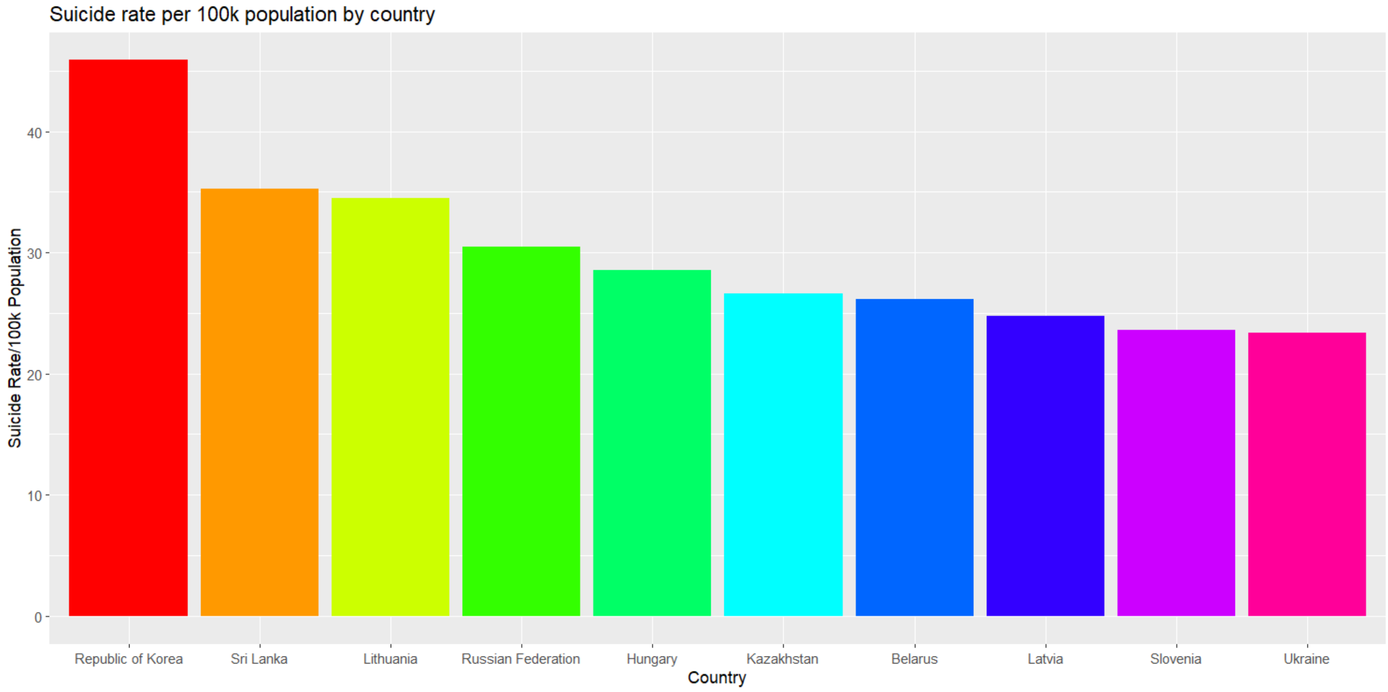
#The elderly are the most likely to commit suicide out of all age groups.

#PART 4: TOP 10 COUNTRIES WITH THE HIGHEST YEARLY SUICIDE RATES

```
countrycide <- suicides %>%
  group_by(country) %>%
  summarise(mean_suic100k = mean(suicides.100k.pop)) %>%
  arrange(desc(mean_suic100k)) %>%
  top_n(10)
```

#The yearly average of suicide rate per 100k population were taken from all countries. However, only the top 10 countries with the highest yearly suicide rates would be charted.

```
ggplot(countrycide, aes(x = reorder(country, -mean_suic100k), y = mean_suic100k)) +
  geom_bar(stat = "identity", fill = rainbow(10)) +
  labs(title ="Suicide rate per 100k population by country", x = "Country", y = "Suicide Rate/100k Population")
```

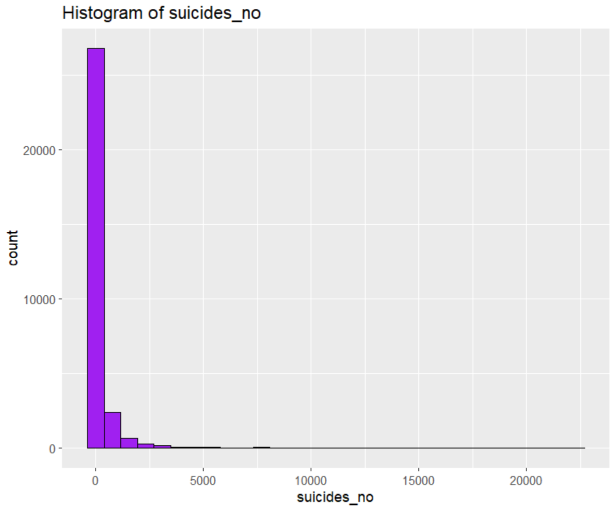


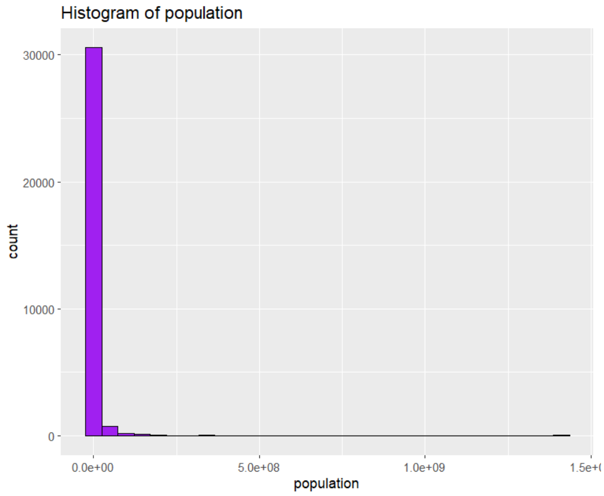
#The country with the highest yearly suicide rate per 100k population is the Republic of Korea/South Korea.

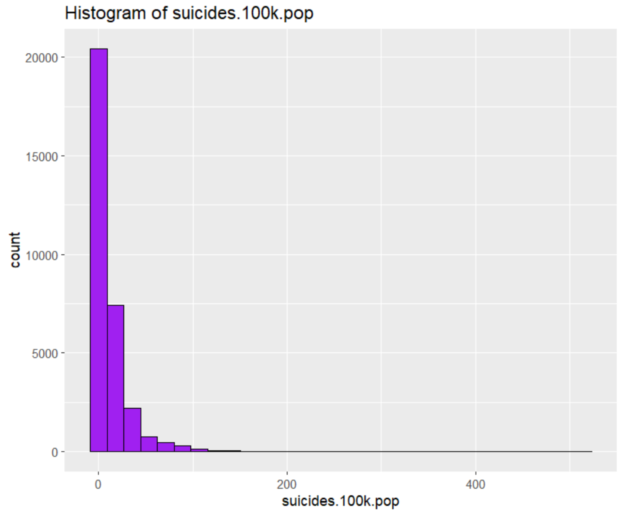
#VISUALISATION OF CATEGORICAL DATA

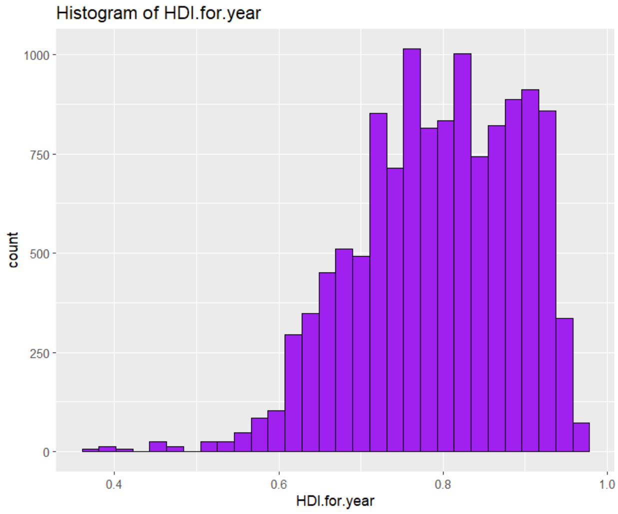
#HISTOGRAM

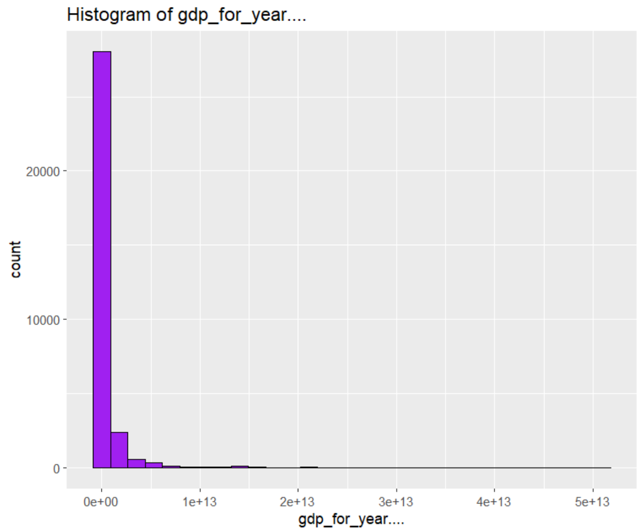
```
numvarcols <- names(numerical_suvar)
histogramm <- function(var) {
  ggplot(suicides, aes_string(x = var)) +
    geom_histogram(color = "black", fill = "purple") +
    ggtitle(paste("Histogram of",var))
}

lapply(numvarcols, histogramm)
```

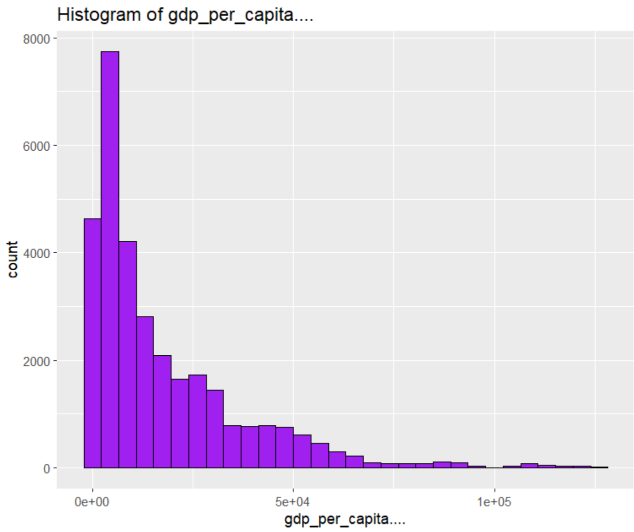

Histogram of suicides_no
count
20000
10000
0
0
5000
10000
15000
20000
suicides_no


Histogram of population
count
30000
20000
10000
0
0.0e+00
5.0e+08
1.0e+09
population





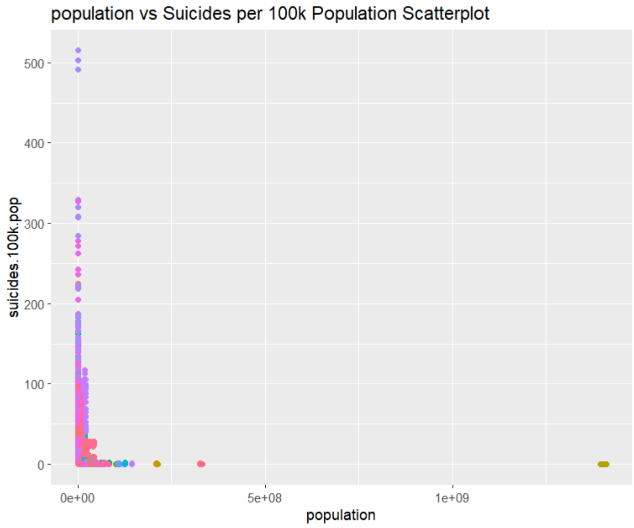


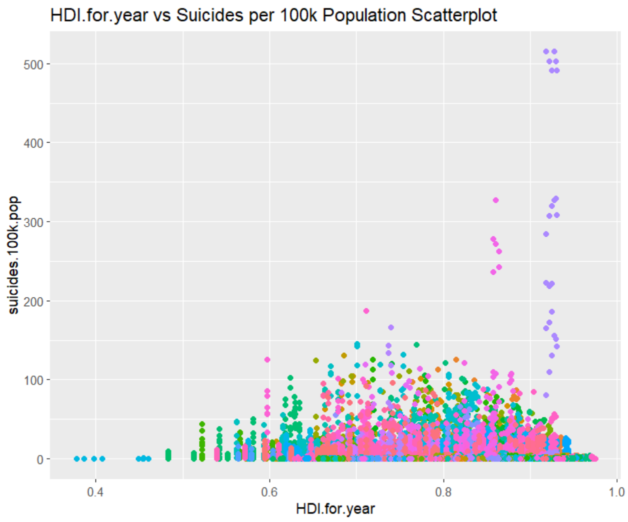


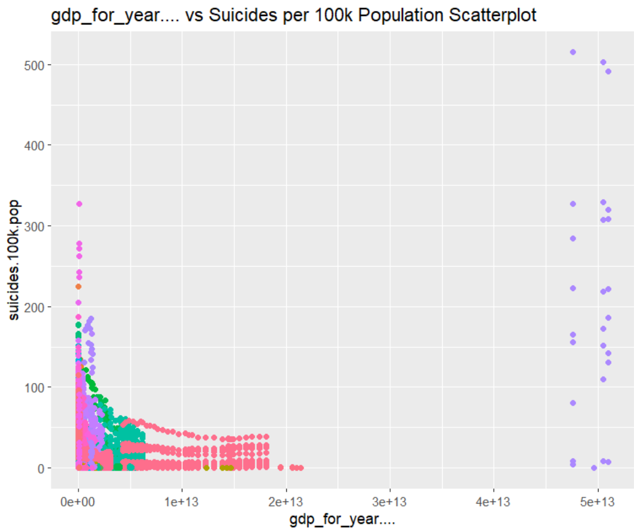
#SCATTER PLOT

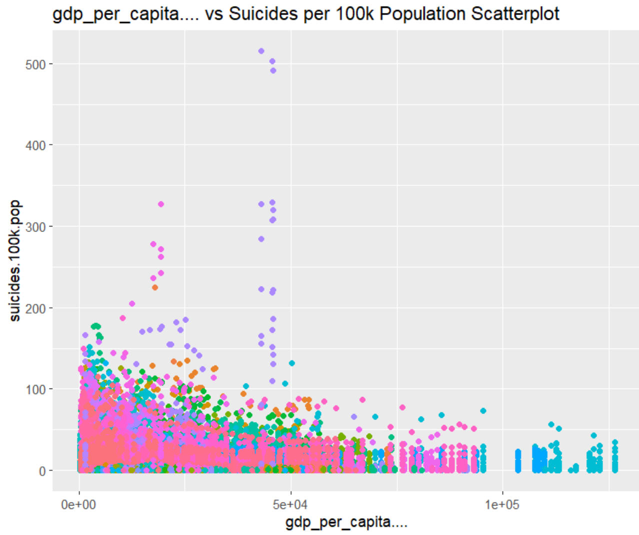
```
scnvc <- names(subset(numerical_suvar, select = -c(suicides_no, suicides.100k.pop)))
scatterplots <- function(var) {
  ggplot(suicides, aes_string(x = var[1], y = "suicides.100k.pop")) +
    geom_point(aes(colour = country)) +
    ggtitle(paste(var[1], "vs", "Suicides per 100k Population", "Scatterplot")) +
    theme(legend.position = "none")
}
lapply(scnvc, scatterplots)
```

#Scatter plots are made for each numerical variable set against the suicide rate per 100k population.



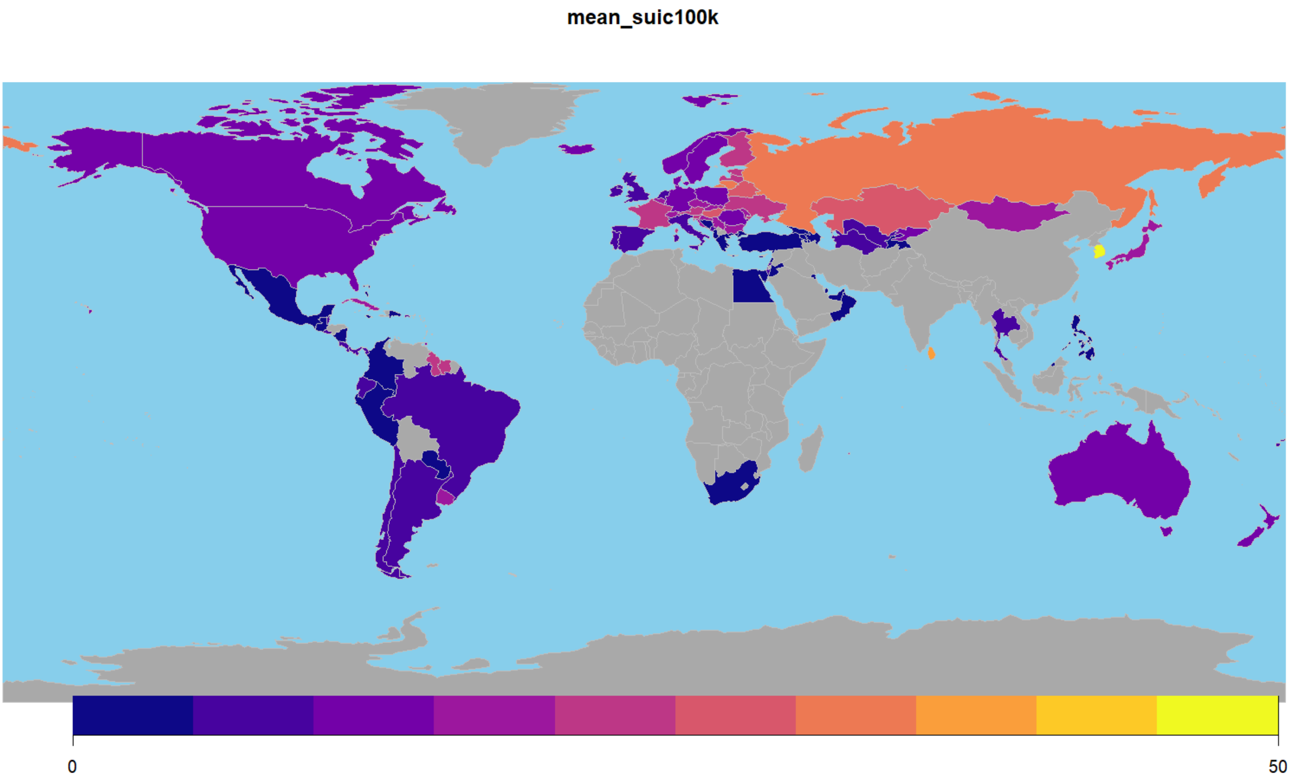






#BRIEF EXPLORATORY DATA ANALYSIS OF NUMERICAL DATA
#DISTRIBUTION OF YEARLY SUICIDE RATE/100K POPULATION BY COUNTRY

```
library(viridis)
library(rworldmap) #A world map showing the distribution of suicide rates will be created.
countrycidemap <- suicides %>%
  group_by(country) %>%
  summarise(mean_suic100k = mean(suicides.100k.pop))

worldmapsketch <- joinCountryData2Map(countrycidemap, joinCode = "NAME", nameJoinColumn =
"country") #The list of countries in the suicides dataset will be matched to a world map.
worldmap <- mapCountryData(worldmapsketch,
          nameColumnToPlot="mean_suic100k",
          colourPalette = plasma(10),
          oceanCol="skyblue",
          missingCountryCol="darkgrey", #Countries absent from the suicides dataset will be coloured grey.
          catMethod = "pretty"); worldmap
```



QUESTION C

Observe and discuss on the quality of “Suicides” dataset based on the analyses in part (a) and (b).

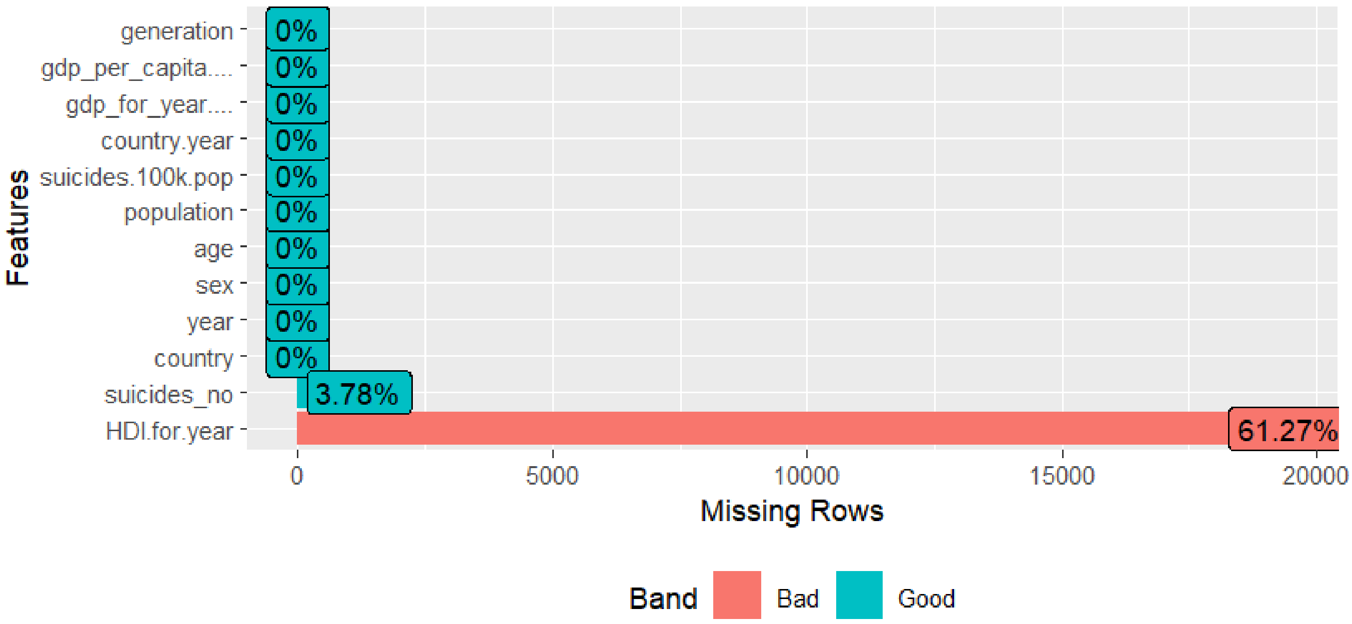
#The suicides dataset seems rather incomplete. A lot of countries, especially in the Asian and African continent, are not included in the dataset (as highlighted in grey in the world map visualisation). Also, the figures regarding the number of suicides and suicide rate per 100k population for the year of 2020 were also not filled yet, providing a misleading figure for the year’s trend (as seen as the almost zero value for 2020 in yearly suicide rate trends across categories during the categorical data visualisation). Outliers are also present for certain variables in the dataset, but cases such as South Korea’s exceptionally high suicide rate are relevant to the data analysis.

#A sizeable amount of missing value, making up 5.4% of the dataset, was also observed by using the plot_intro() function. Let’s have a closer look of the variables contributing to the amount of missing values.

profile_missing(suicides)

```
> profile_missing(suicides)
             feature num_missing pct_missing
1            country           0  0.00000000
2               year           0  0.00000000
3                sex           0  0.00000000
4                age           0  0.00000000
5        suicides_no        1200  0.03778813
6         population           0  0.00000000
7   suicides.100k.pop           0  0.00000000
8       country.year           0  0.00000000
9        HDI.for.year       19456  0.61267162
10    gdp_for_year....           0  0.00000000
11 gdp_per_capita....           0  0.00000000
12        generation           0  0.00000000
```

plot_missing(suicides)



#It seems like the variable suicides_no has a decent amount of missing value while the values for the variable hdi.for.year are mostly missing.

QUESTION D

Produce new data set by removing all the missing values in Suicides dataset. Rename the new dataset as Suicides_new.

#More than half of the variable hdi.for.year consists of missing values, hence, the variable will be dropped entirely. The 3.78% missing values of suicides_no, however, will be simply removed.

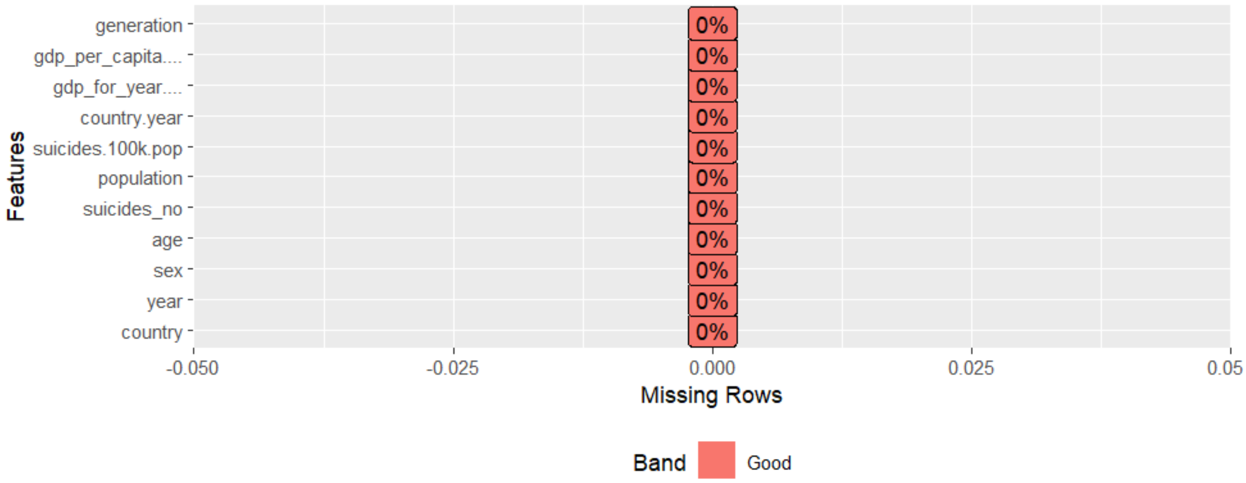
```
suicides_new <- suicides %>%
        select(-HDI.for.year) %>%
        na.omit()
```

#Now that the missing values have been removed from the suicides dataset, it's time to check the suicides_new to confirm this removal:

introduce(suicides_new)

```
> introduce(suicides_new)
   rows columns discrete_columns continuous_columns all_missing_columns
1 30556      11                6                  5                   0
  total_missing_values complete_rows total_observations memory_usage
1                    0         30556             336116      1924584
```

plot_missing(suicides_new)



QUESTION E

Apply appropriate statistical test to check on the suitability of Suicides_new dataset in pursuing further analysis. Give your comment and suggestions.

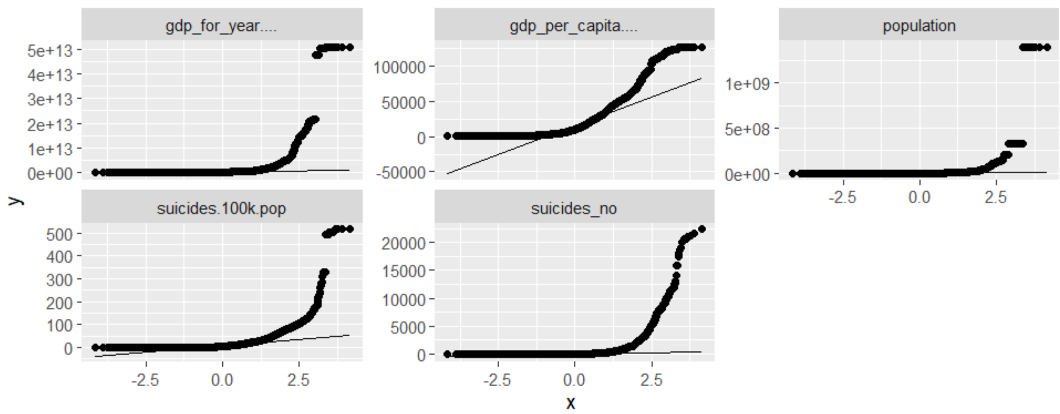
#Sometimes, statistical methods for a dataset are selected based on the data's type and distribution. According to the article "[Selection of Appropriate Statistical Methods for Data Analysis](#)", if the data follows a normal distribution and the sample size is large, parametric methods are generally preferred while if the data does not follow a normal distribution or the sample size is small, nonparametric methods should be used.

#In order to move on to choosing the right statistical methods to be used for the case study in the next question, some tests have to be performed to see whether the variables of the data are normally distributed or not.

#There are several ways to check if a dataset follows a normal distribution:

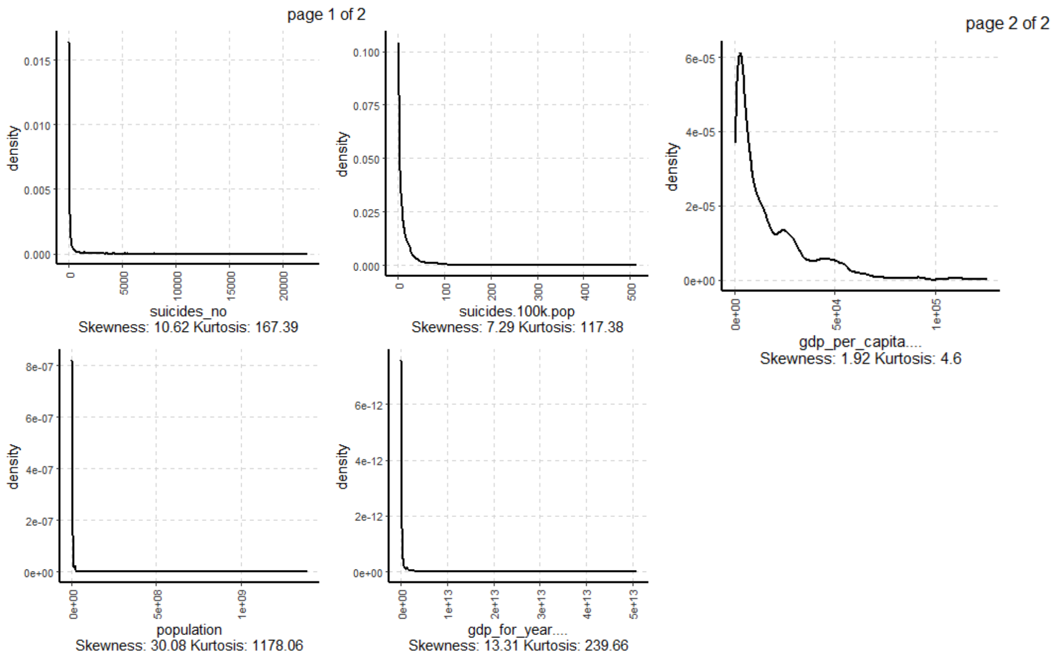
#1. VISUALLY: CREATING A QQ-PLOT

plot_qq(suicides_new) *#All of the numerical variables in the dataset are not normally distributed.*



#2. VISUALLY: DENSITY PLOT (INCLUDING SKEW AND KURTOSIS)

densityplot <- ExpNumViz(suicides_new[sapply(suicides_new, function(x) is.integer(x) | is.numeric(x))],target=NULL,nlim=10,Page=c(2,2),sample=5); densityplot



#The density of all the numerical variables in suicides_new did not follow the normal bell curve shape and almost all of them have very high skew and kurtosis values.

#3. NORMALITY TEST

library(nortest)

#Shapiro-Wilk normality test cannot be done on the numerical variables in suicides_new since the number of observations must only be between 3 and 3000. Hence, an alternative normality test needs to be chosen.

#Anderson-Darling Normality Test

lapply(suicides_new[sapply(suicides_new, function(x) is.integer(x) | is.numeric(x))],ad.test)

```
$suicides_no

        Anderson-Darling normality test

data:  X[[i]]
A = 7226.3, p-value < 2.2e-16


$population

        Anderson-Darling normality test

data:  X[[i]]
A = 9017.7, p-value < 2.2e-16


$suicides.100k.pop

        Anderson-Darling normality test

data:  X[[i]]
A = 3459.9, p-value < 2.2e-16


$gdp_for_year....

        Anderson-Darling normality test

data:  X[[i]]
A = 7817, p-value < 2.2e-16


$gdp_per_capita....

        Anderson-Darling normality test

data:  X[[i]]
A = 1861.8, p-value < 2.2e-16
```

#The p-values of the test for all numerical variables in suicides_new are less than 0.05. Thus, we have sufficient evidence to reject the null hypothesis and conclude that all numerical variables in suicides_new does not follow a normal distribution.

#CONCLUSION

#Since the variables in the suicides_new dataset are not normally distributed, in case of further analysis, parametric statistical methods such as t-tests, ANOVA and Pearson correlation analysis cannot be done.

#Suggestions for further steps are either normalizing the variables of the dataset by transforming them via methods such as log, min-max scaling and standard scaling transformations or pursue non-parametric alternatives for the parametric statistical methods instead.

QUESTION F

Create ONE case study by using Suicides_new dataset. You are required to apply appropriate statistical methods such as Two-Samples Independent t-test, Paired samples t-test, ANOVA, Correlation analysis or Chi square test to analyse the data. Comment and discuss on the findings.

#INTRODUCTION

*#Society has become increasingly aware that one's mental health is affected not only by personal risks, but also by the exposure to environmental factors affecting all those who share the same environment, such as a country of residence. The view of suicide has changed from an action stemming solely from an individual's psyche into an extension of social and economic conditions. Since a broad range of socioeconomic variables are possible to be linked to a country's suicide rate, can it be proven that **there is a statistically significant correlation between a country's GDP per capita to its suicide rate per 100k population**?*

#HYPOTHESIS

#H0: There is no significant relationship between GDP per capita and suicide rate per 100k population.
#H1: There is a significant relationship between GDP per capita and suicide rate per 100k population.

#METHODOLOGY

*#Since both the variables gdp_per_capita.... and suicides.100k.pop are not normally distributed, a non-parametric correlation analysis method, **Spearman's rank correlation coefficient**, has been chosen to investigate their relationship.*

```
cor.test(suicides_new$gdp_per_capita....,suicides_new$suicides.100k.pop, method="spearman")
```

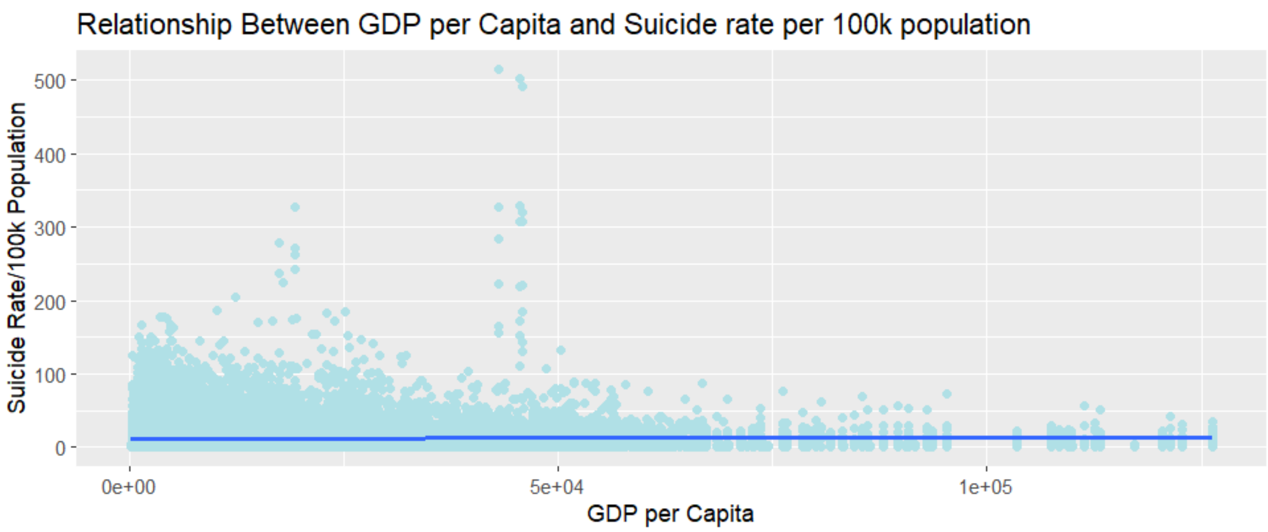
```
        Spearman's rank correlation rho

data:  suicides_new$gdp_per_capita.... and suicides_new$suicides.100k.pop
S = 4.5629e+12, p-value = 1.649e-12
alternative hypothesis: true rho is not equal to 0
sample estimates:
       rho
0.04038109
```

#RESULTS

#Since the p-value is less than 0.05, there is enough evidence to reject h0 and prove that there is a statistically significant relationship between GDP per capita and suicide rate per 100k population. This proves that a country's economy is a significant stress factor for an individual living in the country. Meanwhile, the rho of 0.04038109 shows that there is a low but positive relationship between GDP per capita and suicide rate per 100k population, meaning that while richer countries are associated with higher suicide rates, this relationship is very weak.

#Here's a look into the slight positive relationship between the two variables:

```
ggplot(suicides_new, aes_string(x = suicides_new$gdp_per_capita...., y = "suicides.100k.pop")) +
  geom_point(color="powderblue") + geom_smooth(method = lm) +
  labs(title ="Relationship Between GDP per Capita and Suicide rate per 100k population", x = "GDP per Capita", y = "Suicide Rate/100k Population")
```



QUESTION G

Compute the mean number of suicides for Mexico and Japan for Suicides_new dataset.

#MEAN NUMBER OF SUICIDES IN JAPAN

```
JPSN <- (suicides_new %>%
  filter(country %in% "Japan") %>%
  select(country, suicides_no))$suicides_no; mean(JPSN)
```

```
> JPSN <- (suicides_new %>%
+   filter(country %in% "Japan") %>%
+   select(country, suicides_no))$suicides_no; mean(JPSN)
[1] 2075.24
```

#MEAN NUMBER OF SUICIDES IN MEXICO

MXSN <-(suicides_new %>%
filter(country %in% "Mexico") %>%
select(country, suicides_no))$suicides_no; mean(MXSN)

```
> MXSN <-(suicides_new %>%
+       filter(country %in% "Mexico") %>%
+       select(country, suicides_no))$suicides_no; mean(MXSN)
[1] 328.4595
```

QUESTION H

By using the results obtained in part (g), produce the mean ratio of number of suicides for Mexico and Japan by taking the larger number as the numerator. Interpret the value.

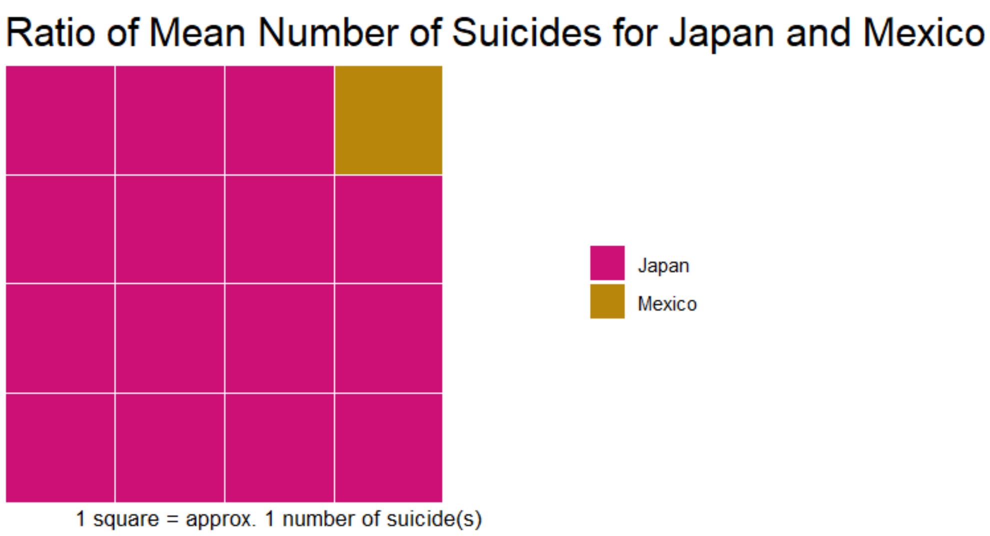
#The mean ratio of number of suicides for Japan and Mexico will be calculated as mean(number of suicides for Japan/number of suicides for Mexico).

meanratio <- mean(JPSN/MXSN); meanratio

```
[1] 15.70668
```

#INTERPRETATION OF THE MEAN RATIO

```
library(waffle)
propor = c(Japan=meanratio,Mexico=1)
waffle(propor,rows=4,size=0.5,colors = c("deeppink3", "darkgoldenrod"),
   title = "Ratio of Mean Number of Suicides for Japan and Mexico",
   xlab = "1 square = approx. 1 number of suicide(s)")
```



#This mean ratio means that on average, the number of suicides in Japan outnumbers that of Mexico by 15.70668 to 1.

QUESTION I

By considering the information in part (g) and (h), generate 50, 500, 2000 and 5000 bootstrap samples for the mean ratio. Illustrate your results by using suitable plots.

#PREPARING THE DATA TO BE SAMPLED

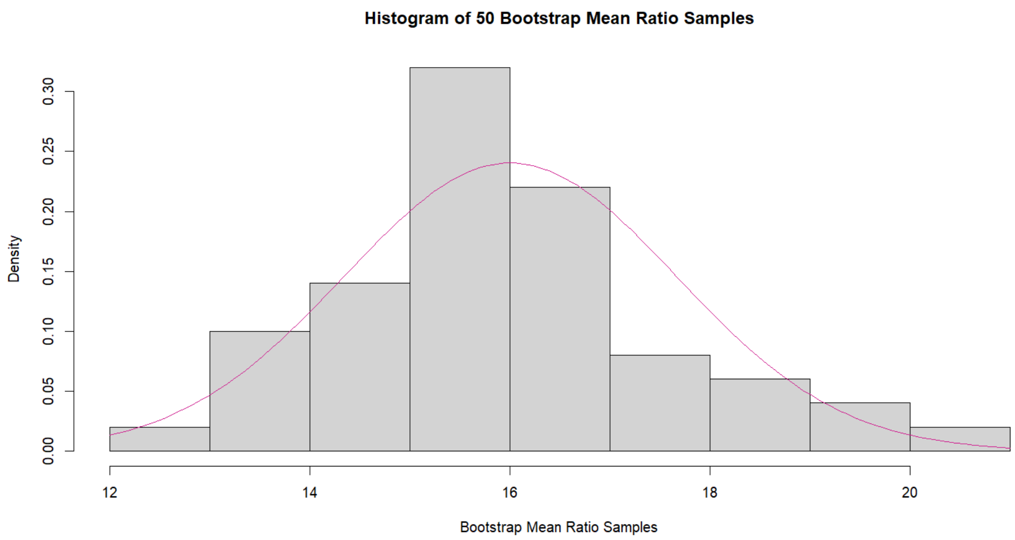
```
set.seed(100)
dt1 <- (suicides_new %>%
  filter(country %in% "Japan") %>%
  select(country, suicides_no))[, 2]

dt2 <- (suicides_new %>%
  filter(country %in% "Mexico") %>%
  select(country, suicides_no))[, 2]

dtb <- cbind(dt1,dt2)
```

#USING THE BOOT FUNCTION FOR BOOTSTRAP ESTIMATE OF STANDARD ERROR

```
library(boot)
sampler <- function(x,n){
        set.seed(100) #Setting the initial value of the random-number seed.
        results <- boot(data=x, statistic=function(d,i) mean(d[i,1]/d[i,2]), R=n)
return(results$t[,1])}  #Creating the bootstrap sampling function for the mean ratio.
```

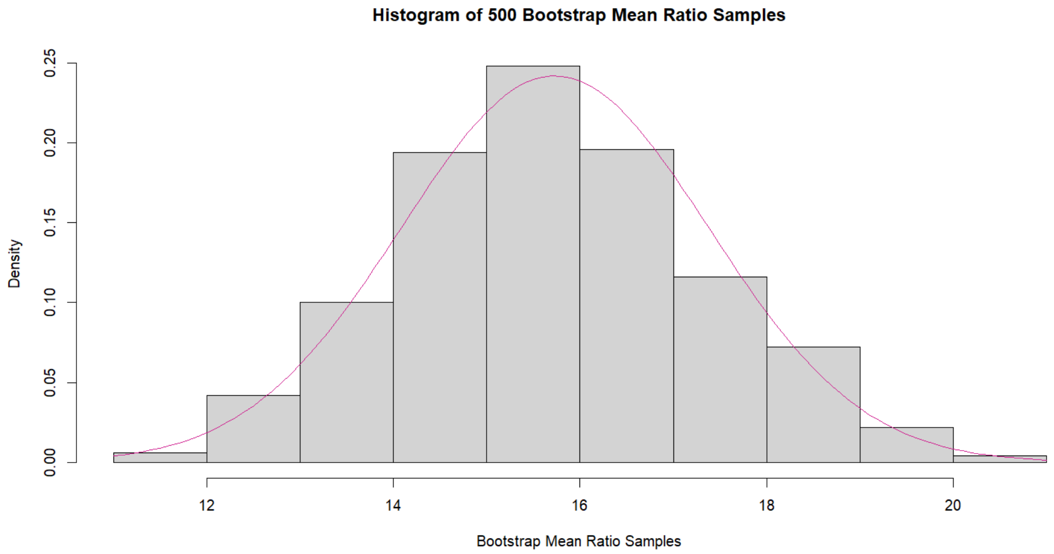
#GENERATING 50 BOOTSTRAP SAMPLES

```
sampler(dtb,50)
hist(sampler(dtb,50), main="Histogram of 50 Bootstrap Mean Ratio Samples", xlab="Bootstrap Mean Ratio Samples", freq=FALSE)
curve(dnorm(x,mean=mean(sampler(dtb,50)),sd=sd(sampler(dtb,50))), add=TRUE,col="violetred")
```



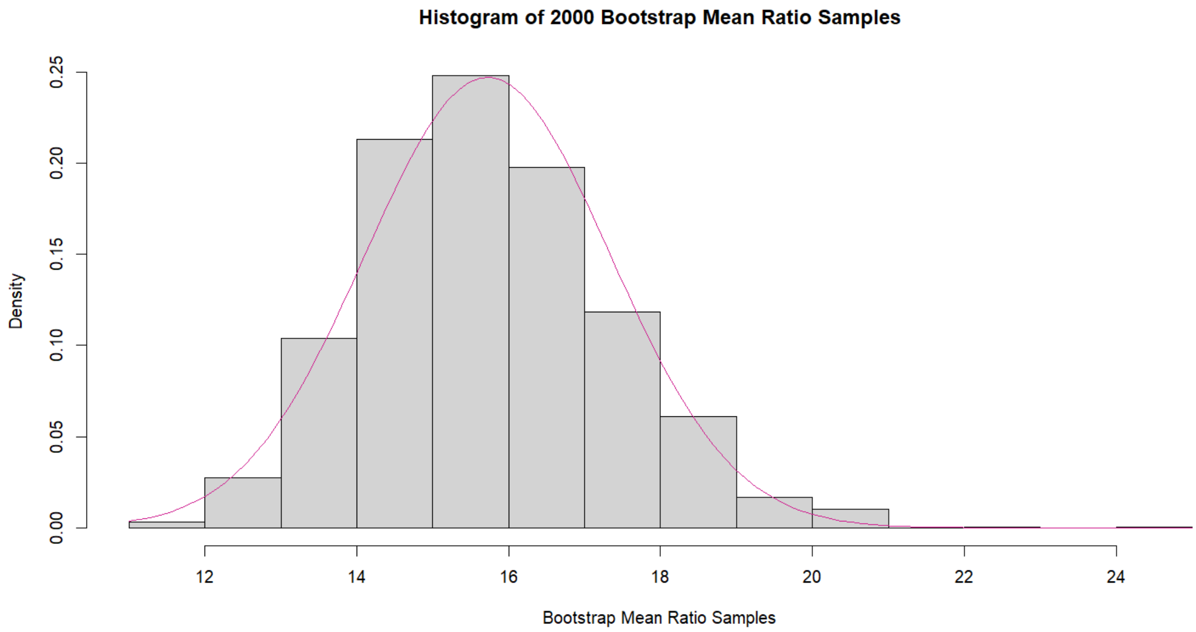
#GENERATING 500 BOOTSTRAP SAMPLES

sampler(dtb,500)
hist(sampler(dtb,500), main="Histogram of 500 Bootstrap Mean Ratio Samples", xlab="Bootstrap Mean Ratio Samples", freq=FALSE)
curve(dnorm(x,mean=mean(sampler(dtb,500)),sd=sd(sampler(dtb,500))), add=TRUE,col="violetred")



#GENERATING 2000 BOOTSTRAP SAMPLES

sampler(dtb,2000)
hist(sampler(dtb,2000), main="Histogram of 2000 Bootstrap Mean Ratio Samples", xlab="Bootstrap Mean Ratio Samples", freq=FALSE)
curve(dnorm(x,mean=mean(sampler(dtb,2000)),sd=sd(sampler(dtb,2000))), add=TRUE,col="violetred")



#GENERATING 5000 BOOTSTRAP SAMPLES

```
sampler(dtb,5000)
hist(sampler(dtb,5000), main="Histogram of 5000 Bootstrap Mean Ratio Samples", xlab="Bootstrap Mean Ratio Samples", freq=FALSE)
curve(dnorm(x,mean=mean(sampler(dtb,5000)),sd=sd(sampler(dtb,5000))), add=TRUE,col="violetred")
```