Basic data manipulation and visualization

Aishwarya

January 21, 2019

Let us write a function to subset a given dataset

```
library(ggplot2)
sub <- function(data, ...){
  arg <- list(...)
  a = data[0]

  for(i in arg){

    a <- cbind(a, data[i])
    }

return(a)
}
```

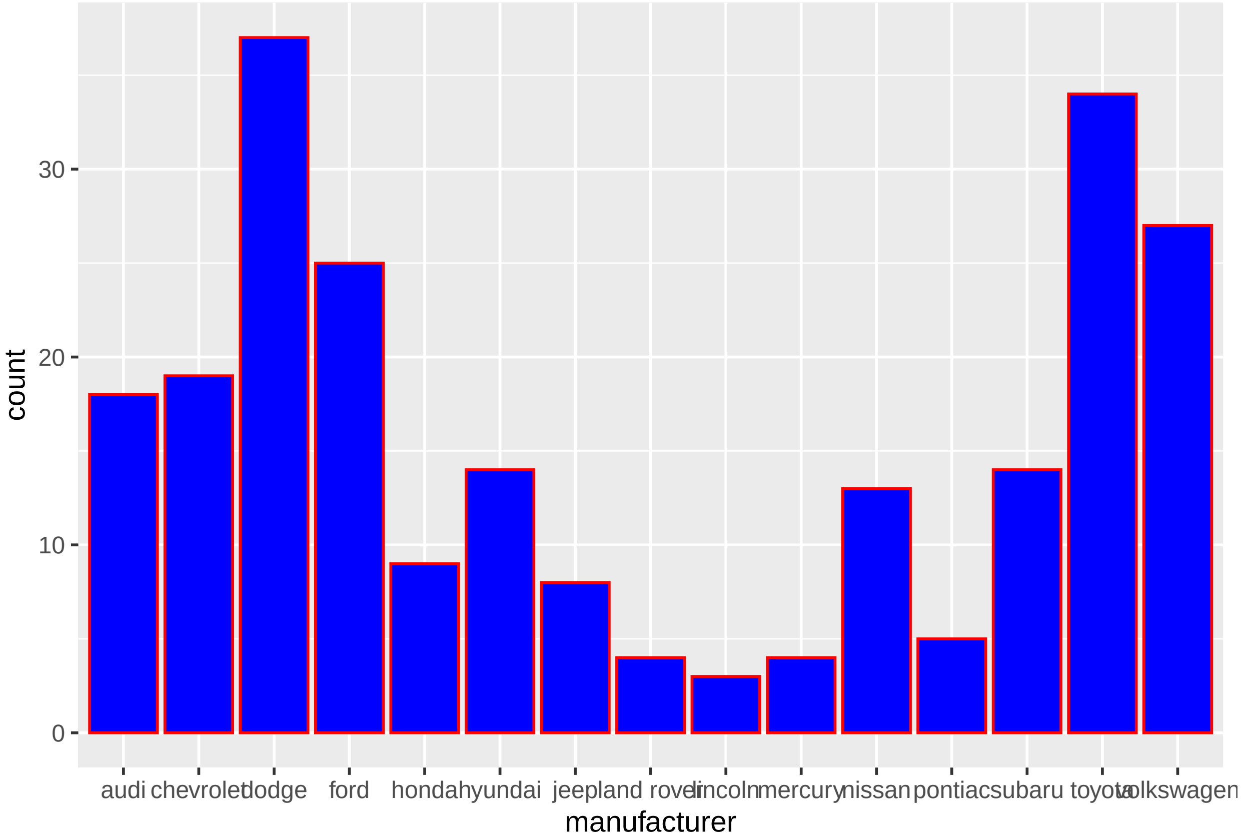
Testing it on mpg dataset

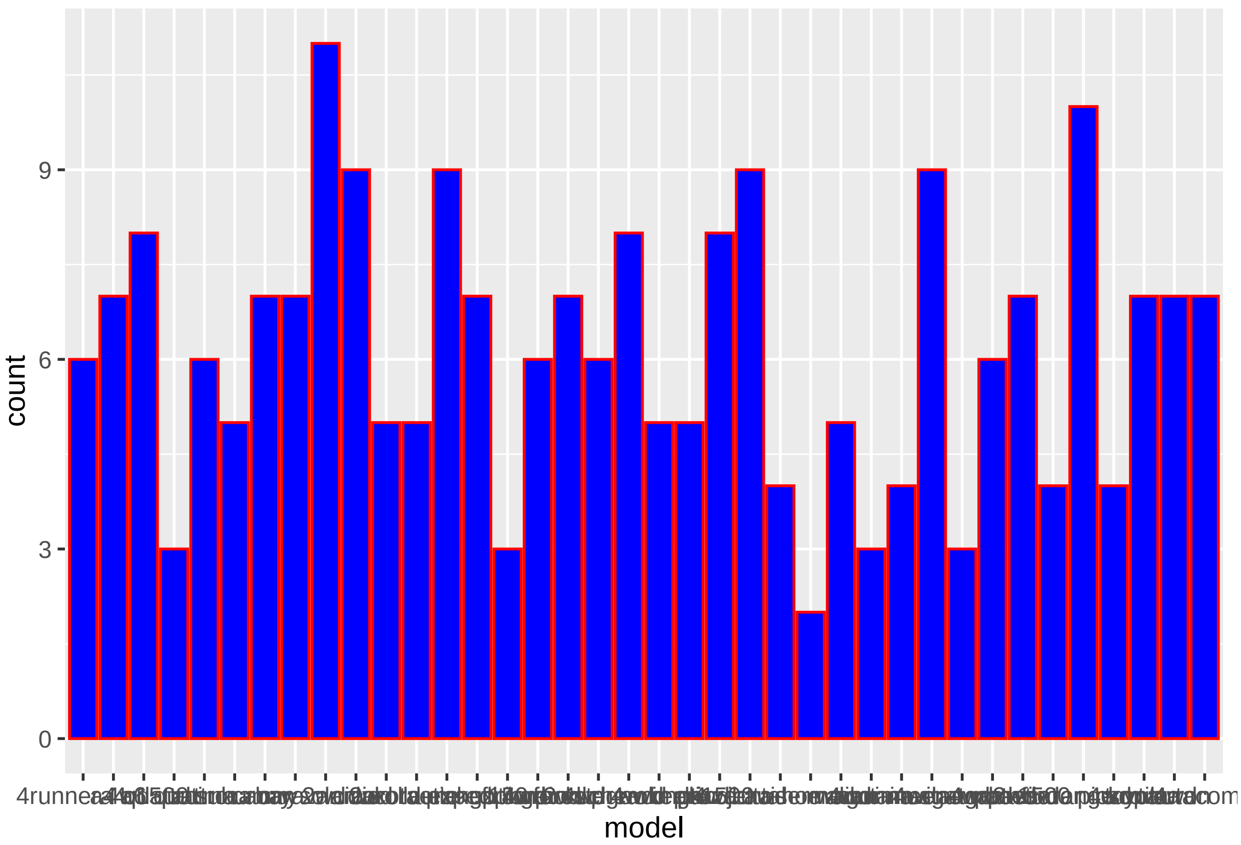
```
sub(mpg, "drv" ,3, "cyl", 1, 4)
```

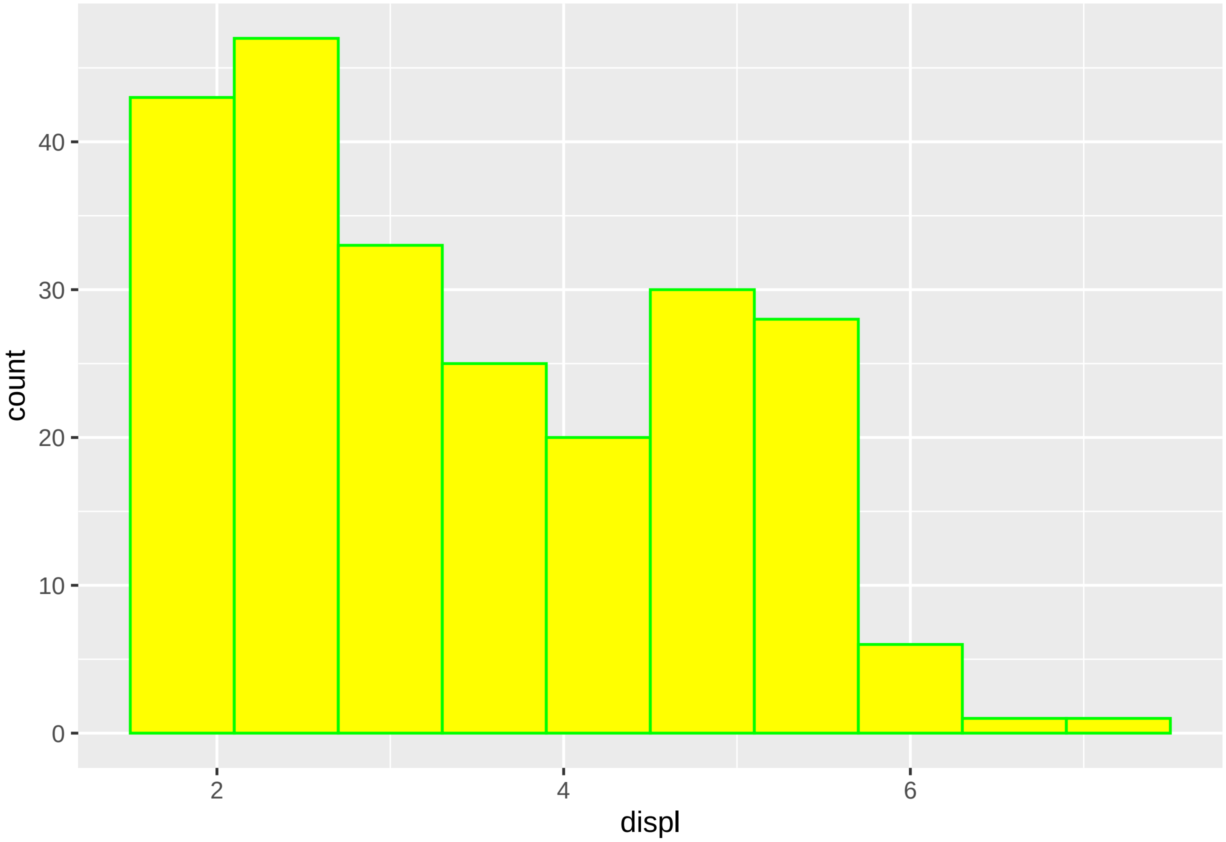
```
##     drv displ cyl manufacturer year
## 1     f   1.8   4         audi 1999
## 2     f   1.8   4         audi 1999
## 3     f   2.0   4         audi 2008
## 4     f   2.0   4         audi 2008
## 5     f   2.8   6         audi 1999
## 6     f   2.8   6         audi 1999
## 7     f   3.1   6         audi 2008
## 8     4   1.8   4         audi 1999
## 9     4   1.8   4         audi 1999
## 10    4   2.0   4         audi 2008
## 11    4   2.0   4         audi 2008
## 12    4   2.8   6         audi 1999
## 13    4   2.8   6         audi 1999
## 14    4   3.1   6         audi 2008
## 15    4   3.1   6         audi 2008
## 16    4   2.8   6         audi 1999
## 17    4   3.1   6         audi 2008
## 18    4   4.2   8         audi 2008
## 19    r   5.3   8    chevrolet 2008
## 20    r   5.3   8    chevrolet 2008
## 21    r   5.3   8    chevrolet 2008
## 22    r   5.7   8    chevrolet 1999
## 23    r   6.0   8    chevrolet 2008
## 24    r   5.7   8    chevrolet 1999
## 25    r   5.7   8    chevrolet 1999
## 26    r   6.2   8    chevrolet 2008
## 27    r   6.2   8    chevrolet 2008
## 28    r   7.0   8    chevrolet 2008
## 29    4   5.3   8    chevrolet 2008
## 30    4   5.3   8    chevrolet 2008
## 31    4   5.7   8    chevrolet 1999
## 32    4   6.5   8    chevrolet 1999
## 33    f   2.4   4    chevrolet 1999
## 34    f   2.4   4    chevrolet 2008
## 35    f   3.1   6    chevrolet 1999
## 36    f   3.5   6    chevrolet 2008
## 37    f   3.6   6    chevrolet 2008
## 38    f   2.4   4        dodge 1999
## 39    f   3.0   6        dodge 1999
## 40    f   3.3   6        dodge 1999
## 41    f   3.3   6        dodge 1999
## 42    f   3.3   6        dodge 2008
## 43    f   3.3   6        dodge 2008
## 44    f   3.3   6        dodge 2008
## 45    f   3.8   6        dodge 1999
## 46    f   3.8   6        dodge 1999
## 47    f   3.8   6        dodge 2008
## 48    f   4.0   6        dodge 2008
## 49    4   3.7   6        dodge 2008
## 50    4   3.7   6        dodge 2008
## 51    4   3.9   6        dodge 1999
## 52    4   3.9   6        dodge 1999
## 53    4   4.7   8        dodge 2008
## 54    4   4.7   8        dodge 2008
## 55    4   4.7   8        dodge 2008
## 56    4   5.2   8        dodge 1999
## 57    4   5.2   8        dodge 1999
## 58    4   3.9   6        dodge 1999
## 59    4   4.7   8        dodge 2008
## 60    4   4.7   8        dodge 2008
## 61    4   4.7   8        dodge 2008
## 62    4   5.2   8        dodge 1999
## 63    4   5.7   8        dodge 2008
## 64    4   5.9   8        dodge 1999
## 65    4   4.7   8        dodge 2008
## 66    4   4.7   8        dodge 2008
## 67    4   4.7   8        dodge 2008
## 68    4   4.7   8        dodge 2008
## 69    4   4.7   8        dodge 2008
## 70    4   4.7   8        dodge 2008
## 71    4   5.2   8        dodge 1999
## 72    4   5.2   8        dodge 1999
## 73    4   5.7   8        dodge 2008
## 74    4   5.9   8        dodge 1999
## 75    r   4.6   8         ford 1999
## 76    r   5.4   8         ford 1999
## 77    r   5.4   8         ford 2008
## 78    4   4.0   6         ford 1999
## 79    4   4.0   6         ford 1999
## 80    4   4.0   6         ford 1999
## 81    4   4.0   6         ford 2008
## 82    4   4.6   8         ford 2008
## 83    4   5.0   8         ford 1999
## 84    4   4.2   6         ford 1999
## 85    4   4.2   6         ford 1999
## 86    4   4.6   8         ford 1999
## 87    4   4.6   8         ford 1999
## 88    4   4.6   8         ford 2008
## 89    4   5.4   8         ford 1999
## 90    4   5.4   8         ford 2008
## 91    r   3.8   6         ford 1999
## 92    r   3.8   6         ford 1999
## 93    r   4.0   6         ford 2008
## 94    r   4.0   6         ford 2008
## 95    r   4.6   8         ford 1999
## 96    r   4.6   8         ford 1999
## 97    r   4.6   8         ford 2008
## 98    r   4.6   8         ford 2008
## 99    r   5.4   8         ford 2008
## 100   f   1.6   4        honda 1999
## 101   f   1.6   4        honda 1999
## 102   f   1.6   4        honda 1999
## 103   f   1.6   4        honda 1999
## 104   f   1.6   4        honda 1999
## 105   f   1.8   4        honda 2008
## 106   f   1.8   4        honda 2008
## 107   f   1.8   4        honda 2008
## 108   f   2.0   4        honda 2008
## 109   f   2.4   4      hyundai 1999
## 110   f   2.4   4      hyundai 1999
## 111   f   2.4   4      hyundai 2008
## 112   f   2.4   4      hyundai 2008
## 113   f   2.5   6      hyundai 1999
## 114   f   2.5   6      hyundai 1999
## 115   f   3.3   6      hyundai 2008
## 116   f   2.0   4      hyundai 1999
## 117   f   2.0   4      hyundai 1999
## 118   f   2.0   4      hyundai 2008
## 119   f   2.0   4      hyundai 2008
## 120   f   2.7   6      hyundai 2008
## 121   f   2.7   6      hyundai 2008
## 122   f   2.7   6      hyundai 2008
## 123   4   3.0   6         jeep 2008
## 124   4   3.7   6         jeep 2008
## 125   4   4.0   6         jeep 1999
## 126   4   4.7   8         jeep 1999
## 127   4   4.7   8         jeep 2008
## 128   4   4.7   8         jeep 2008
## 129   4   5.7   8         jeep 2008
## 130   4   6.1   8         jeep 2008
## 131   4   4.0   8   land rover 1999
## 132   4   4.2   8   land rover 2008
## 133   4   4.4   8   land rover 2008
## 134   4   4.6   8   land rover 1999
## 135   r   5.4   8      lincoln 1999
## 136   r   5.4   8      lincoln 1999
## 137   r   5.4   8      lincoln 2008
## 138   4   4.0   6      mercury 1999
## 139   4   4.0   6      mercury 2008
## 140   4   4.6   8      mercury 2008
## 141   4   5.0   8      mercury 1999
## 142   f   2.4   4       nissan 1999
## 143   f   2.4   4       nissan 1999
## 144   f   2.5   4       nissan 2008
## 145   f   2.5   4       nissan 2008
## 146   f   3.5   6       nissan 2008
## 147   f   3.5   6       nissan 2008
## 148   f   3.0   6       nissan 1999
## 149   f   3.0   6       nissan 1999
## 150   f   3.5   6       nissan 2008
## 151   4   3.3   6       nissan 1999
## 152   4   3.3   6       nissan 1999
## 153   4   4.0   6       nissan 2008
## 154   4   5.6   8       nissan 2008
## 155   f   3.1   6      pontiac 1999
## 156   f   3.8   6      pontiac 1999
## 157   f   3.8   6      pontiac 1999
## 158   f   3.8   6      pontiac 2008
## 159   f   5.3   8      pontiac 2008
## 160   4   2.5   4       subaru 1999
## 161   4   2.5   4       subaru 1999
## 162   4   2.5   4       subaru 2008
## 163   4   2.5   4       subaru 2008
## 164   4   2.5   4       subaru 2008
## 165   4   2.5   4       subaru 2008
## 166   4   2.2   4       subaru 1999
## 167   4   2.2   4       subaru 1999
## 168   4   2.5   4       subaru 1999
## 169   4   2.5   4       subaru 1999
## 170   4   2.5   4       subaru 2008
## 171   4   2.5   4       subaru 2008
## 172   4   2.5   4       subaru 2008
## 173   4   2.5   4       subaru 2008
## 174   4   2.7   4       toyota 1999
## 175   4   2.7   4       toyota 1999
## 176   4   3.4   6       toyota 1999
## 177   4   3.4   6       toyota 1999
## 178   4   4.0   6       toyota 2008
## 179   4   4.7   8       toyota 2008
## 180   f   2.2   4       toyota 1999
## 181   f   2.2   4       toyota 1999
## 182   f   2.4   4       toyota 2008
## 183   f   2.4   4       toyota 2008
## 184   f   3.0   6       toyota 1999
## 185   f   3.0   6       toyota 1999
## 186   f   3.5   6       toyota 2008
## 187   f   2.2   4       toyota 1999
## 188   f   2.2   4       toyota 1999
## 189   f   2.4   4       toyota 2008
## 190   f   2.4   4       toyota 2008
## 191   f   3.0   6       toyota 1999
## 192   f   3.0   6       toyota 1999
## 193   f   3.3   6       toyota 2008
## 194   f   1.8   4       toyota 1999
## 195   f   1.8   4       toyota 1999
## 196   f   1.8   4       toyota 1999
## 197   f   1.8   4       toyota 2008
## 198   f   1.8   4       toyota 2008
## 199   4   4.7   8       toyota 1999
## 200   4   5.7   8       toyota 2008
## 201   4   2.7   4       toyota 1999
## 202   4   2.7   4       toyota 1999
## 203   4   2.7   4       toyota 2008
## 204   4   3.4   6       toyota 1999
## 205   4   3.4   6       toyota 1999
## 206   4   4.0   6       toyota 2008
## 207   4   4.0   6       toyota 2008
## 208   f   2.0   4   volkswagen 1999
## 209   f   2.0   4   volkswagen 1999
## 210   f   2.0   4   volkswagen 2008
## 211   f   2.0   4   volkswagen 2008
## 212   f   2.8   6   volkswagen 1999
## 213   f   1.9   4   volkswagen 1999
## 214   f   2.0   4   volkswagen 1999
## 215   f   2.0   4   volkswagen 1999
## 216   f   2.0   4   volkswagen 2008
## 217   f   2.0   4   volkswagen 2008
## 218   f   2.5   5   volkswagen 2008
## 219   f   2.5   5   volkswagen 2008
## 220   f   2.8   6   volkswagen 1999
## 221   f   2.8   6   volkswagen 1999
## 222   f   1.9   4   volkswagen 1999
## 223   f   1.9   4   volkswagen 1999
## 224   f   2.0   4   volkswagen 1999
## 225   f   2.0   4   volkswagen 1999
## 226   f   2.5   5   volkswagen 2008
## 227   f   2.5   5   volkswagen 2008
## 228   f   1.8   4   volkswagen 1999
## 229   f   1.8   4   volkswagen 1999
## 230   f   2.0   4   volkswagen 2008
## 231   f   2.0   4   volkswagen 2008
## 232   f   2.8   6   volkswagen 1999
## 233   f   2.8   6   volkswagen 1999
## 234   f   3.6   6   volkswagen 2008
```

Now, writing a function to plot each column of dataset.If it’s a continuous variable (numeric), create a histogram. If it’s a categorical variable (character or factor), create a bar plot.

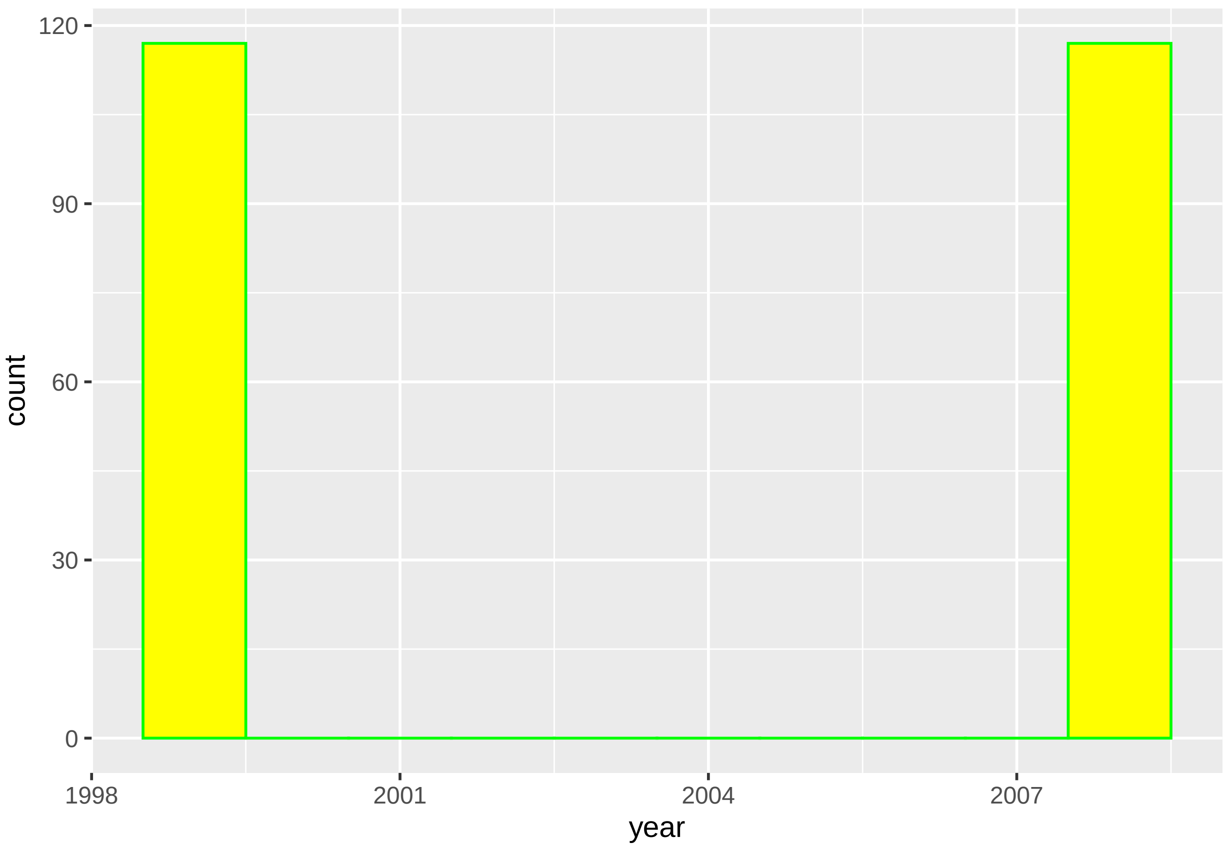
```
plot <- function(data){
  for(i in names(data)){
    for(x in data[i]){
      if(is.numeric(x)){
        gh <- ggplot(data, mapping = aes(x = x)) + geom_histogram(bins = 10, fill ="yellow" , color ="g

        print(gh)
      }
      else{
        gb <- ggplot(data, mapping = aes(x = x)) + geom_bar(fill = "blue", color = "red") + labs(x=i)
        print(gb)
      }
    }
  }

}

plot(mpg)
```

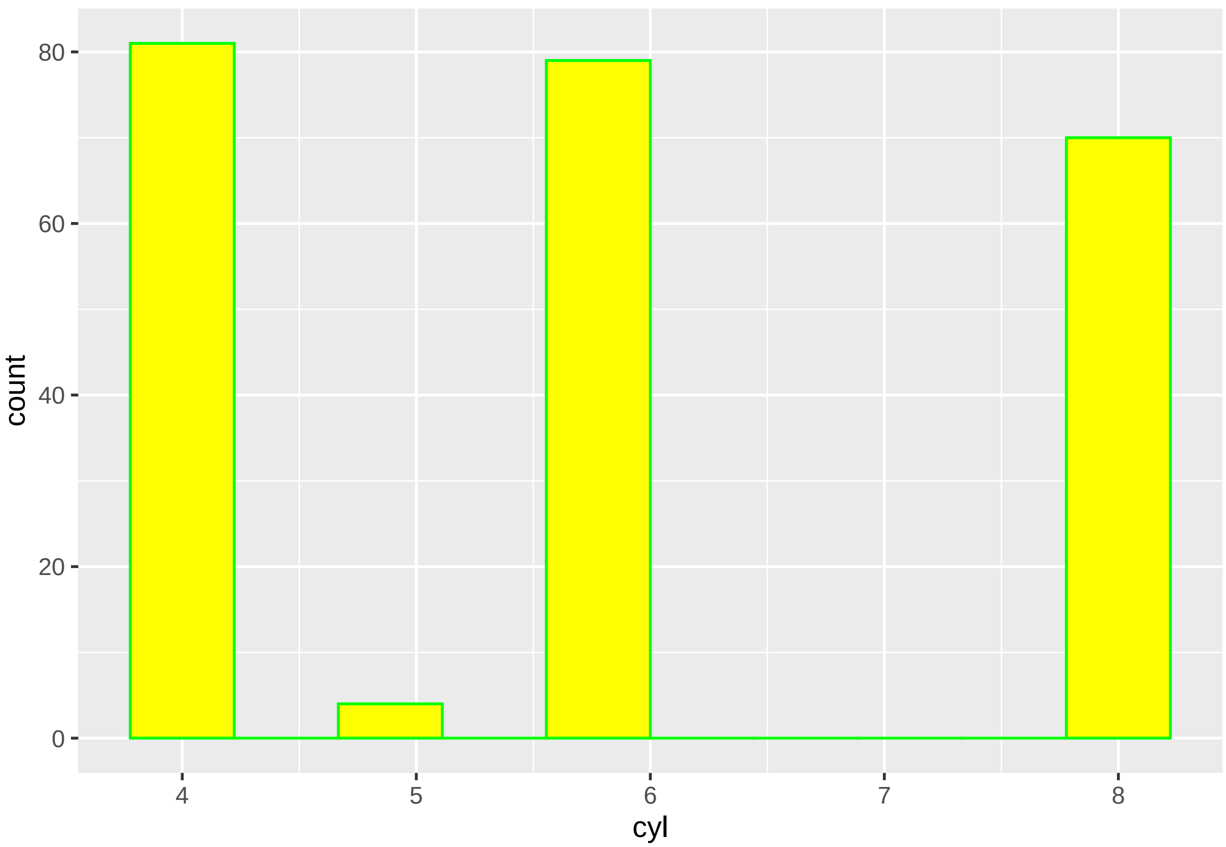


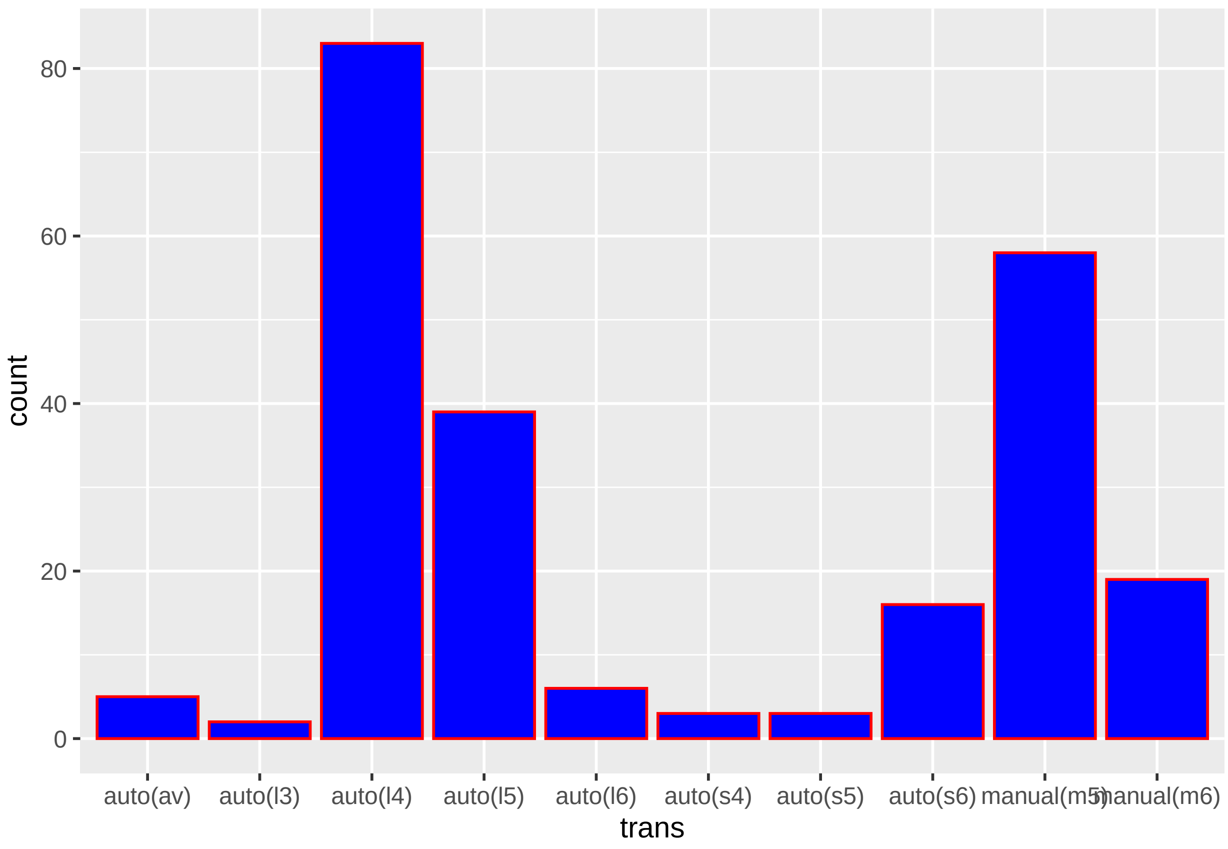

count
9
6
3
0
model



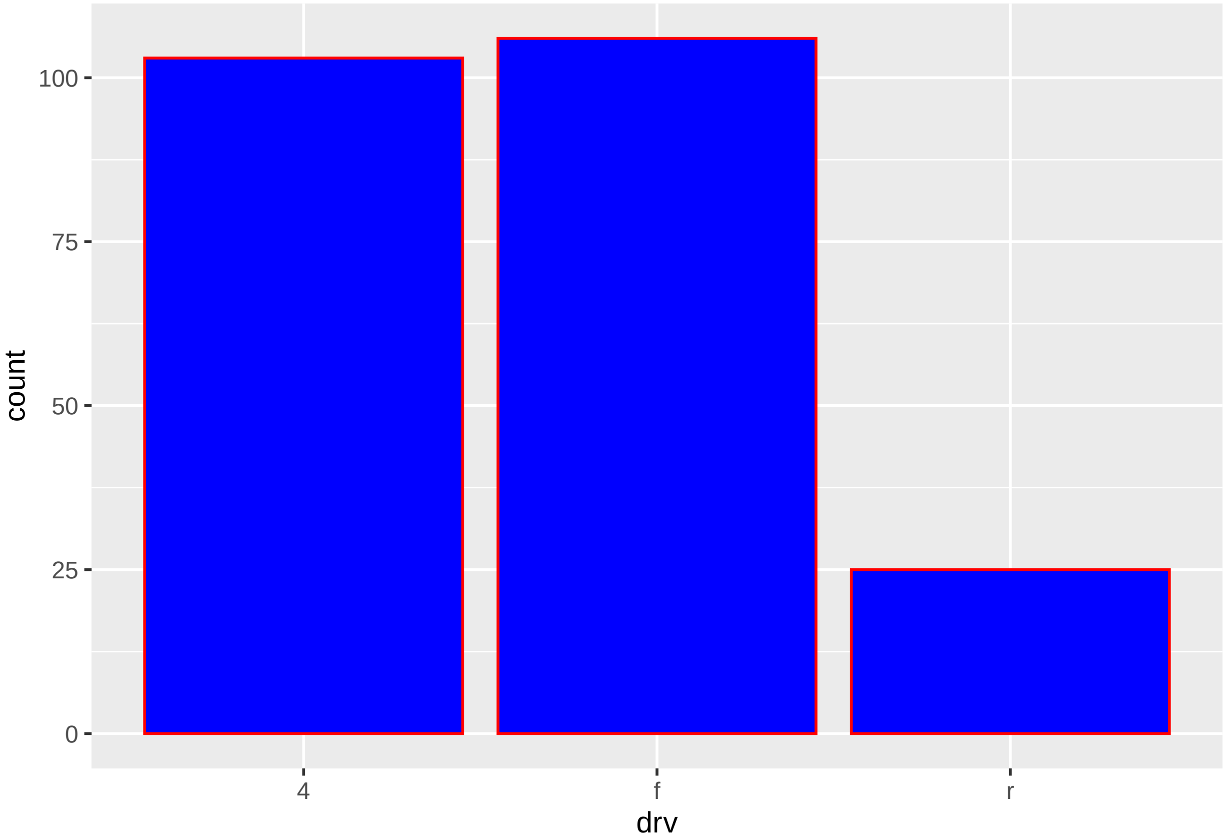
40
30
20
10
0
count
2
4
6
displ

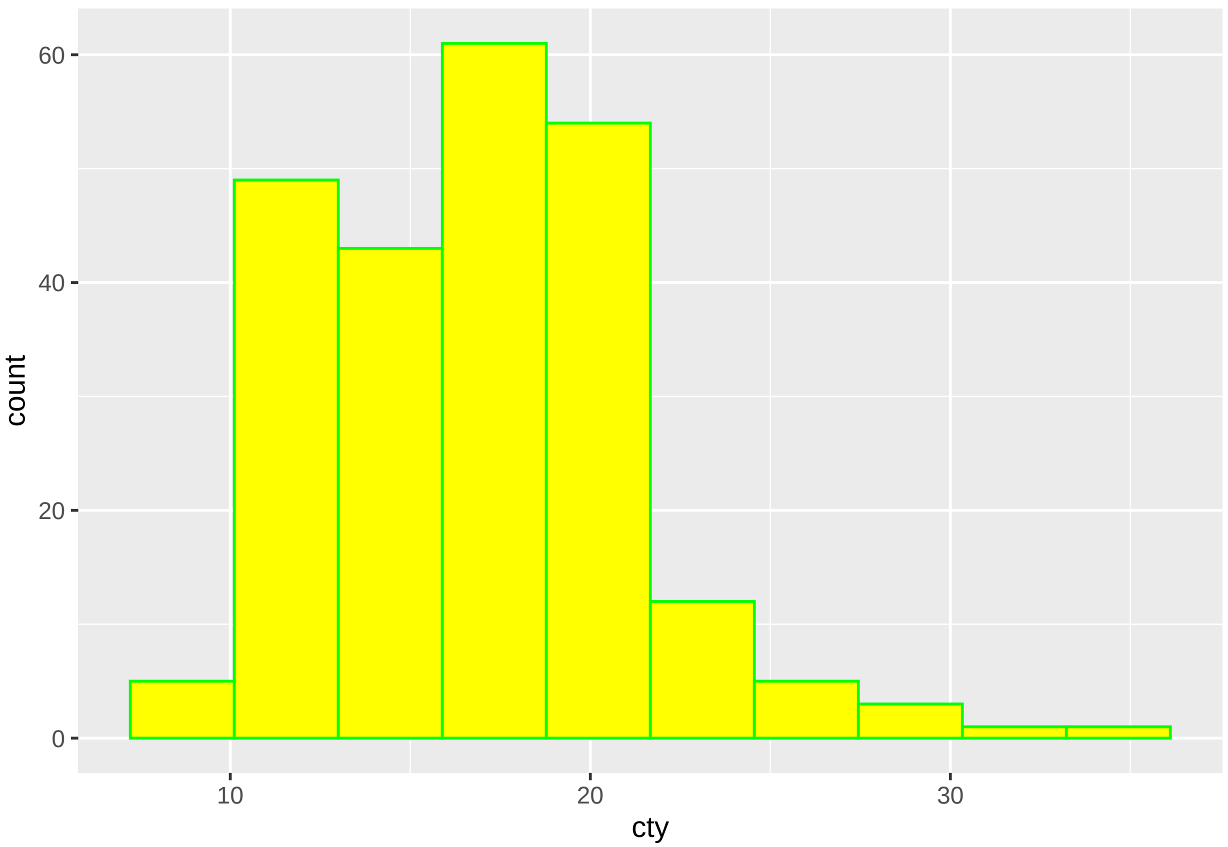

120
90
60
30
0
count
1998
2001
2004
2007
year

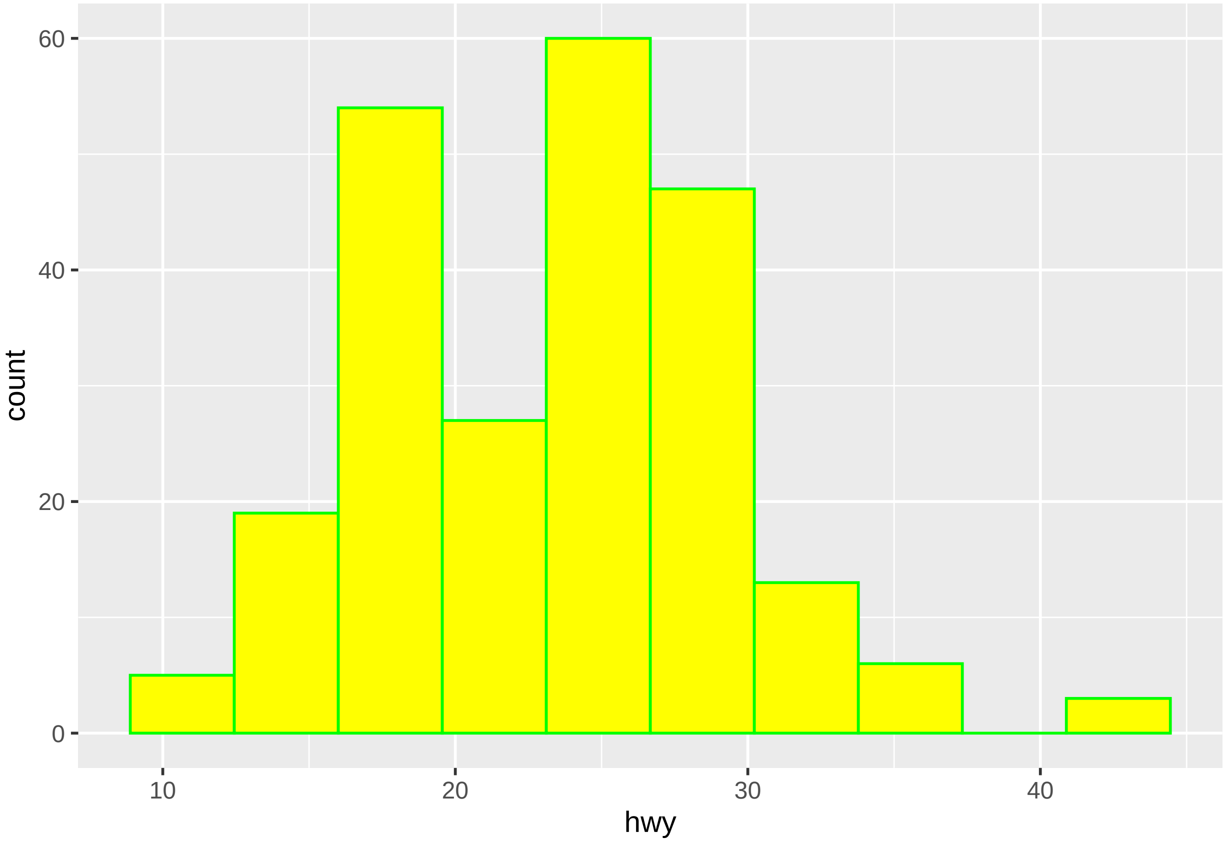

80
60
40
20
0
count
4
5
6
7
8
cyl

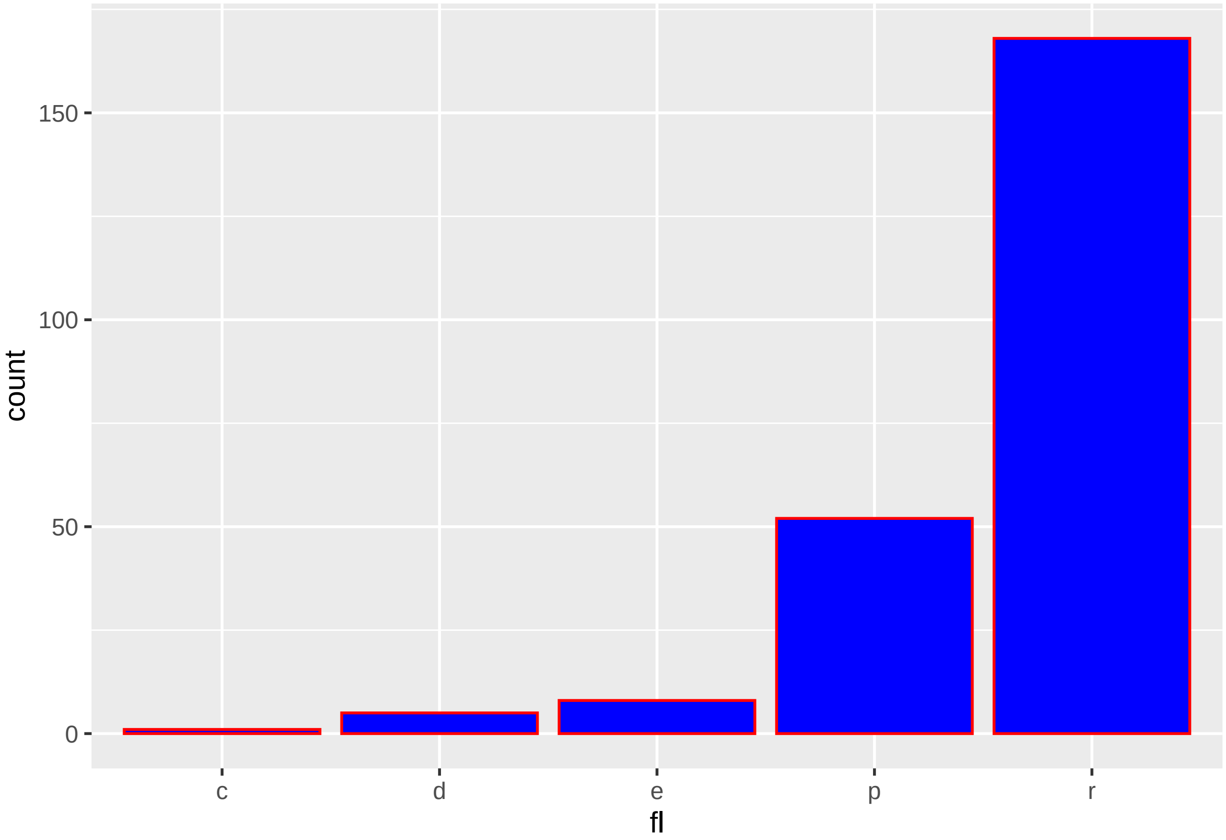


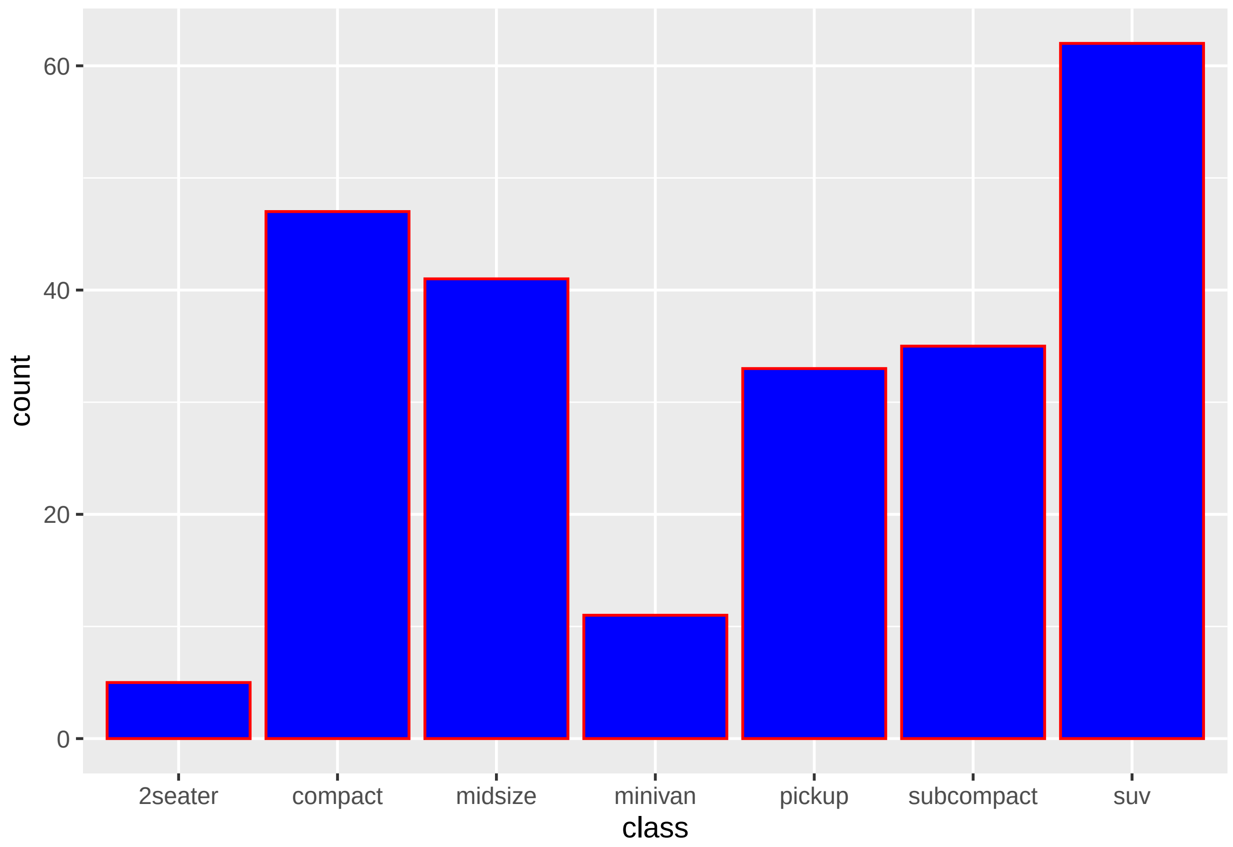
80
60
40
20
0
count
auto(av)
auto(l3)
auto(l4)
auto(l5)
auto(l6)
auto(s4)
auto(s5)
auto(s6)
manual(m5)
manual(m6)
trans


count
100
75
50
25
0
4
f
r
drv


60
40
20
0
count
10
20
30
cty

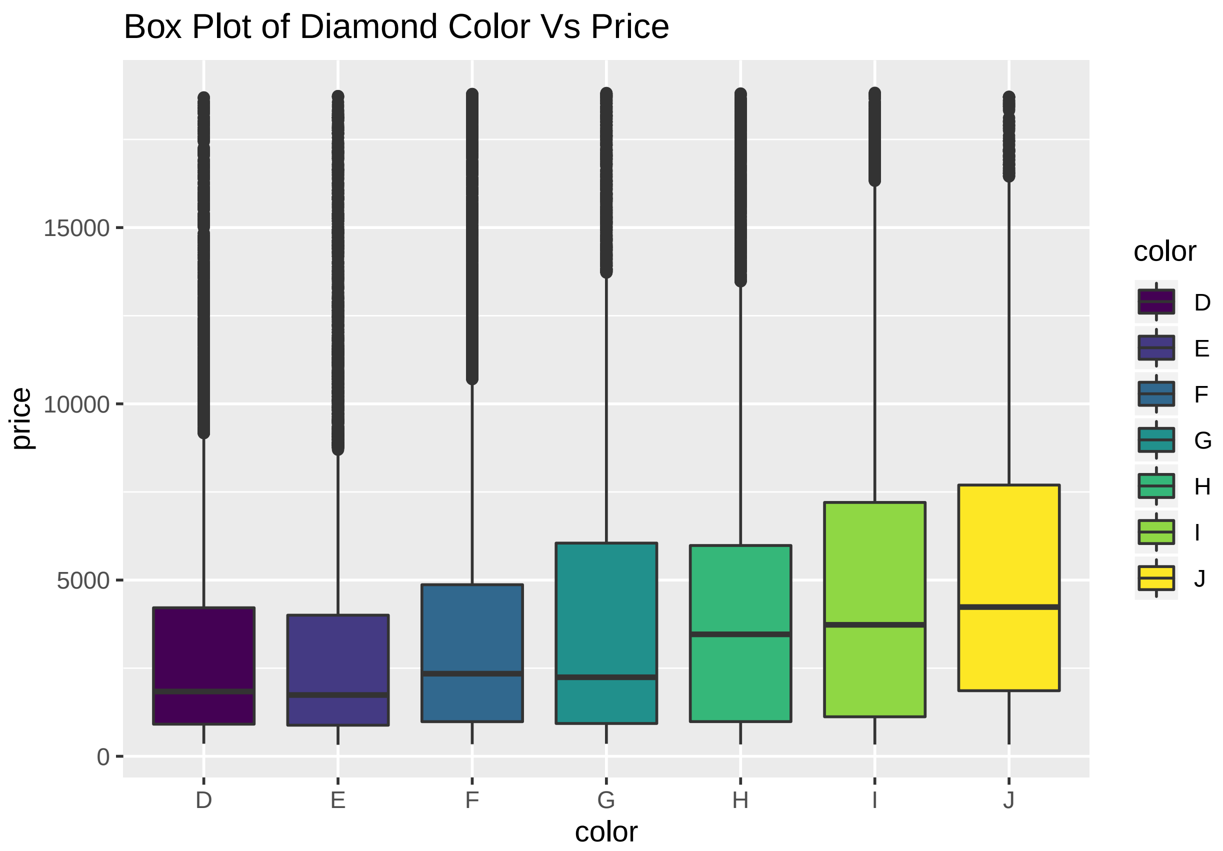

60
40
20
0
count
10
20
30
40
hwy


150
100
50
0
count
c
d
e
p
r
fl



Use side-by-side boxplots to visualize the distribution of price for each level of color.

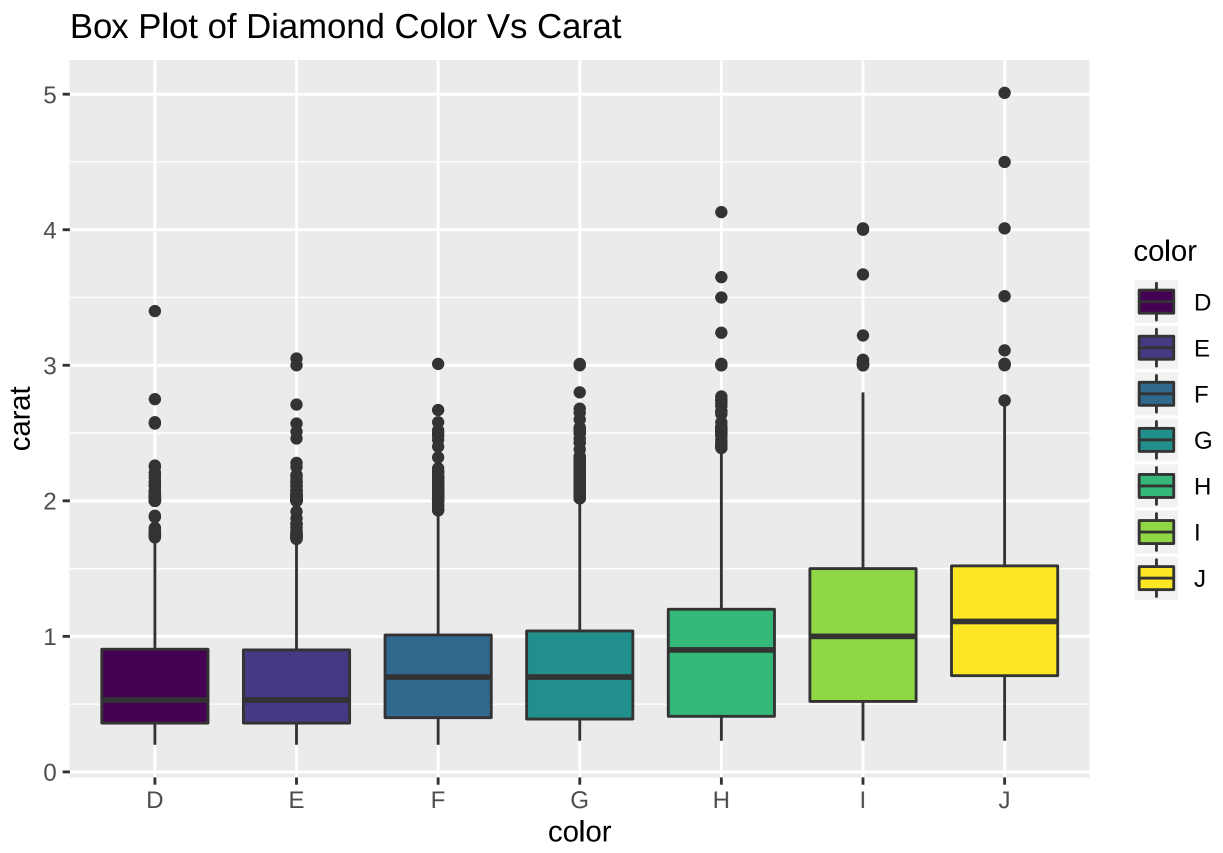
```
ggplot(diamonds, mapping= aes(x = color, y = price)) + geom_boxplot(mapping= aes(fill = color)) +
  ggtitle("Box Plot of Diamond Color Vs Price")
```



We notice that, Worst diamonds have larger spread and less outliers compared to best diamonds. People are paying high for better quality diamonds(G) rather than best(D).

Use side-by-side boxplots to visualize the distribution of carat for each level of color.

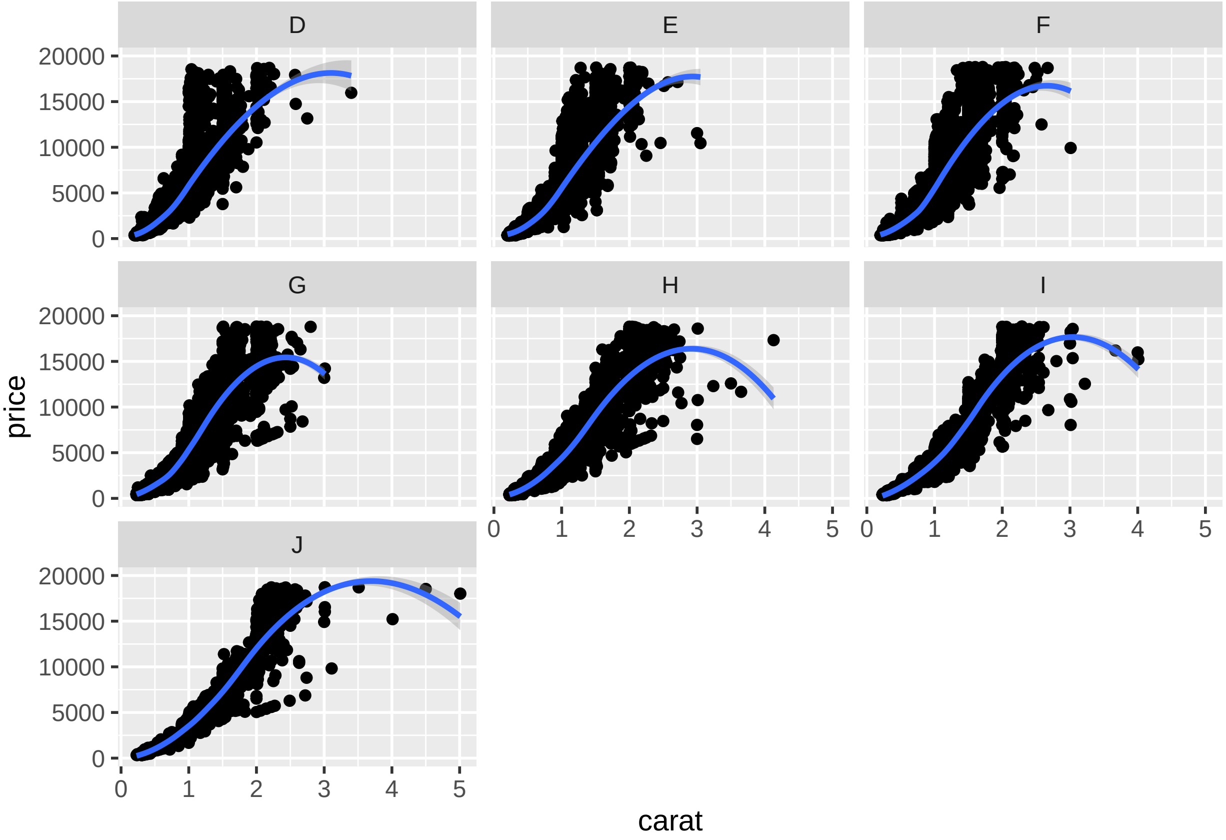
```
ggplot(diamonds, mapping = aes(x = color, y = carat, fill = color)) + geom_boxplot() +
  ggtitle("Box Plot of Diamond Color Vs Carat")
```



Best diamonds has less weight whereas worst are more heavier. From previous plot, people are paying more for Heavier and cheaper diamonds.

scatter plot of carat versus price, using either an additional aesthetic or faceting to visualize the relationship between carat and price for each level of color.

```
ggplot(data = diamonds,
       mapping = aes(x= carat, y= price)) +
  geom_point() + geom_smooth(method = loess) + facet_wrap(~color)
```



It can be said that most of the best diamonds has low weights and are sold at relatively lower prices than the better and worst diamonds. It is strange that none of the best quality diamonds weighs more than 3.5 carats and couldn't cross $16000 whereas even worst diamonds with far lesser weights costs around $18000.

Exploratory Data Analysis

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

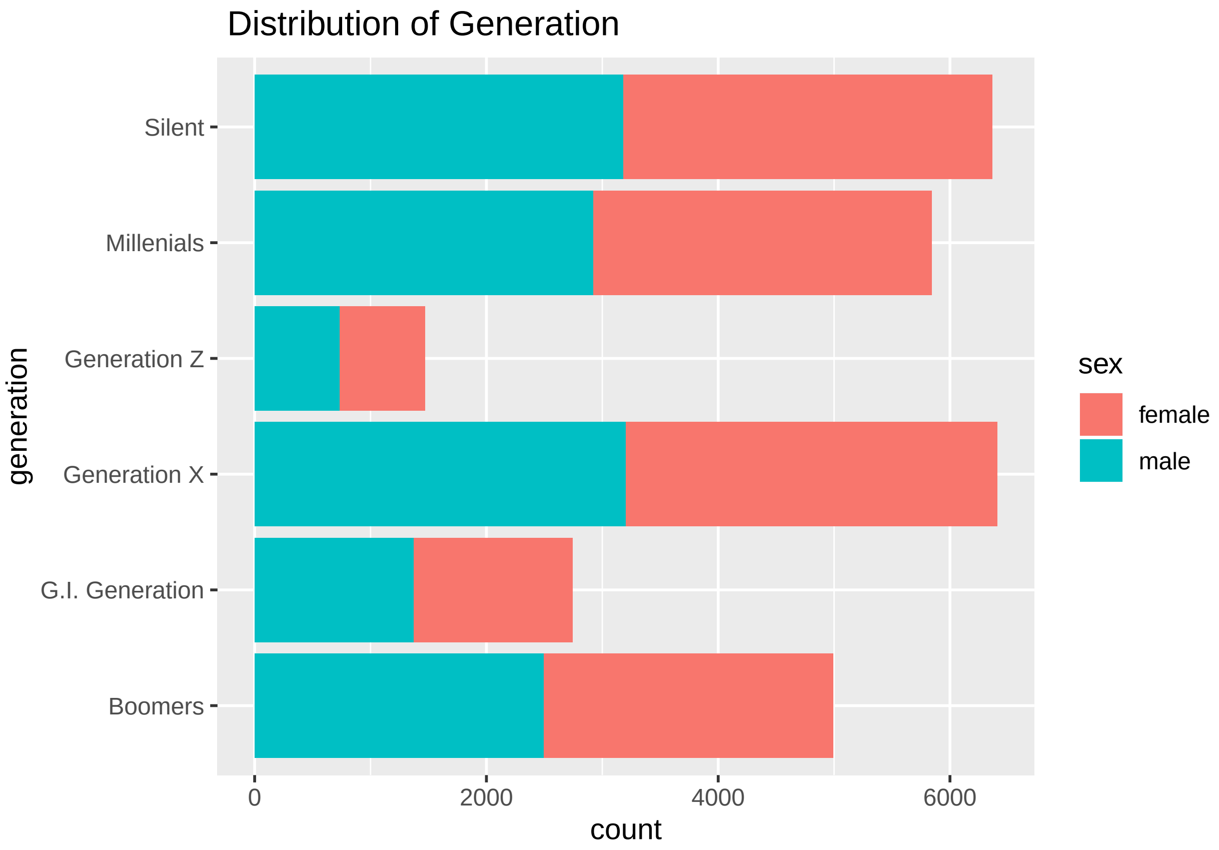
```
#reading data as data frame
data <- as.data.frame(read_csv("C:/Users/Aishwarya/Desktop/NEU/Introduction to Data management/hw3/mast
```

```
## Parsed with column specification:
## cols(
##   country = col_character(),
##   year = col_double(),
##   sex = col_character(),
##   age = col_character(),
##   suicides_no = col_double(),
##   population = col_double(),
##   `suicides/100k pop` = col_double(),
##   `country-year` = col_character(),
##   `HDI for year` = col_double(),
##   `gdp_for_year ($)` = col_number(),
##   `gdp_per_capita ($)` = col_double(),
##   generation = col_character()
## )
```

```
#Tidying
#Data is downloaded from kaggle.
#As the data is almost clean,
#basic transformation would suffice the current requirement.


#Normalising GDP variable
data <- mutate(data,
               `gdp in $100k` = `gdp_for_year ($)`/(100*1000))

#Excluding unwanted variables
data <- select(data, -`country-year`, -`gdp_for_year ($)`)



#Displaying first 10 observations
data[1:10,]
```

```
##    country year    sex         age suicides_no population
## 1  Albania 1987   male 15-24 years          21     312900
## 2  Albania 1987   male 35-54 years          16     308000
## 3  Albania 1987 female 15-24 years          14     289700
## 4  Albania 1987   male   75+ years           1      21800
## 5  Albania 1987   male 25-34 years           9     274300
## 6  Albania 1987 female   75+ years           1      35600
## 7  Albania 1987 female 35-54 years           6     278800
## 8  Albania 1987 female 25-34 years           4     257200
## 9  Albania 1987   male 55-74 years           1     137500
## 10 Albania 1987 female  5-14 years           0     311000
##    suicides/100k pop HDI for year gdp_per_capita ($)      generation
## 1               6.71           NA                796    Generation X
## 2               5.19           NA                796          Silent
## 3               4.83           NA                796    Generation X
## 4               4.59           NA                796 G.I. Generation
## 5               3.28           NA                796         Boomers
## 6               2.81           NA                796 G.I. Generation
## 7               2.15           NA                796          Silent
## 8               1.56           NA                796         Boomers
## 9               0.73           NA                796 G.I. Generation
## 10              0.00           NA                796    Generation X
##    gdp in $100k
## 1      21566.25
## 2      21566.25
## 3      21566.25
## 4      21566.25
## 5      21566.25
## 6      21566.25
## 7      21566.25
## 8      21566.25
## 9      21566.25
## 10     21566.25
```

```
#obs 1
ggplot(data) +
  geom_bar(data,mapping = aes(x = generation,fill = sex)) +
  ggtitle(" Distribution of Generation") +
  coord_flip()
```



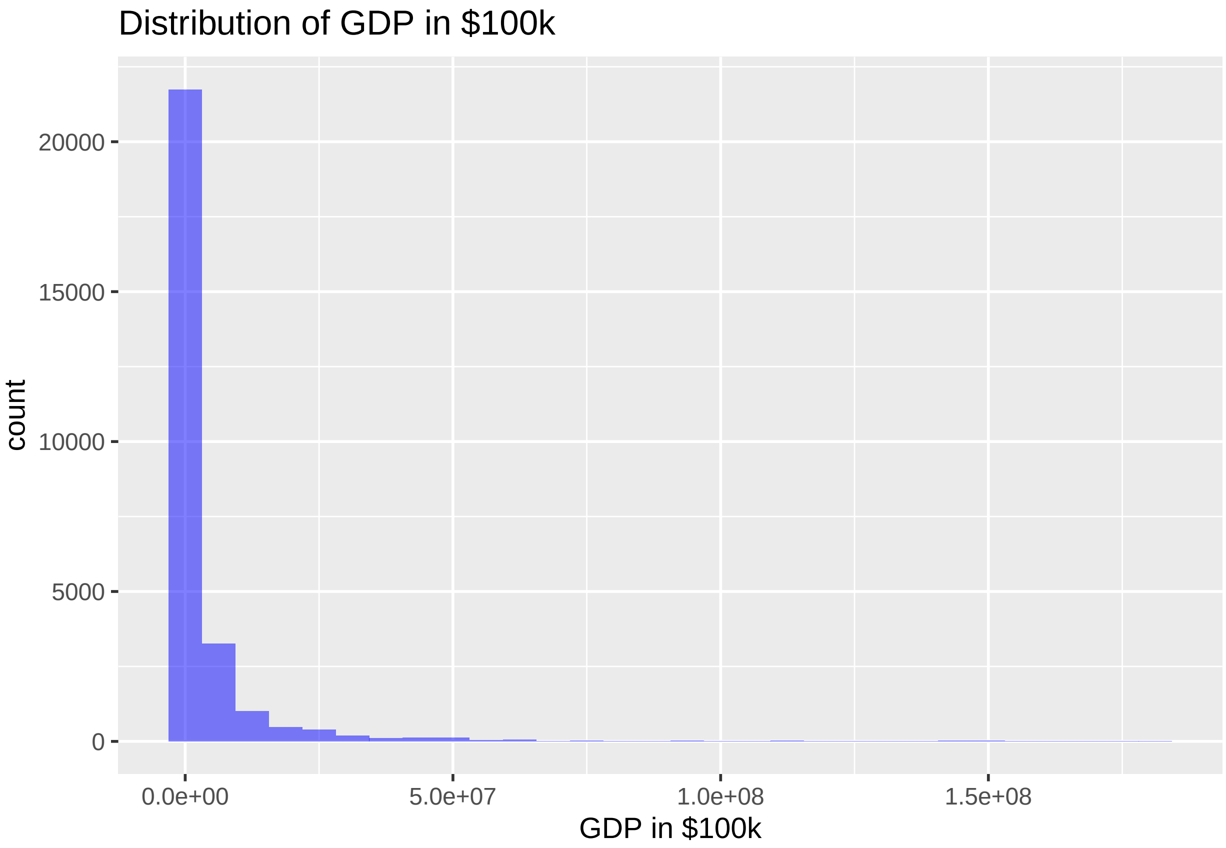
```
data %>% count(generation)
```

```
## # A tibble: 6 x 2
##   generation          n
##   <chr>           <int>
## 1 Boomers          4990
## 2 G.I. Generation  2744
## 3 Generation X      6408
## 4 Generation Z      1470
## 5 Millenials        5844
## 6 Silent            6364
```

Based on graph, Genration X and Silent have higher number of suicide rates. Calculated results also shows the same.

```
#OBS 2

ggplot(data) + geom_histogram(aes(x = `gdp in $100k`),
                              fill = 'blue', alpha= 0.5) +
  ggtitle("Distribution of GDP in $100k") +
  xlab("GDP in $100k")
```

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

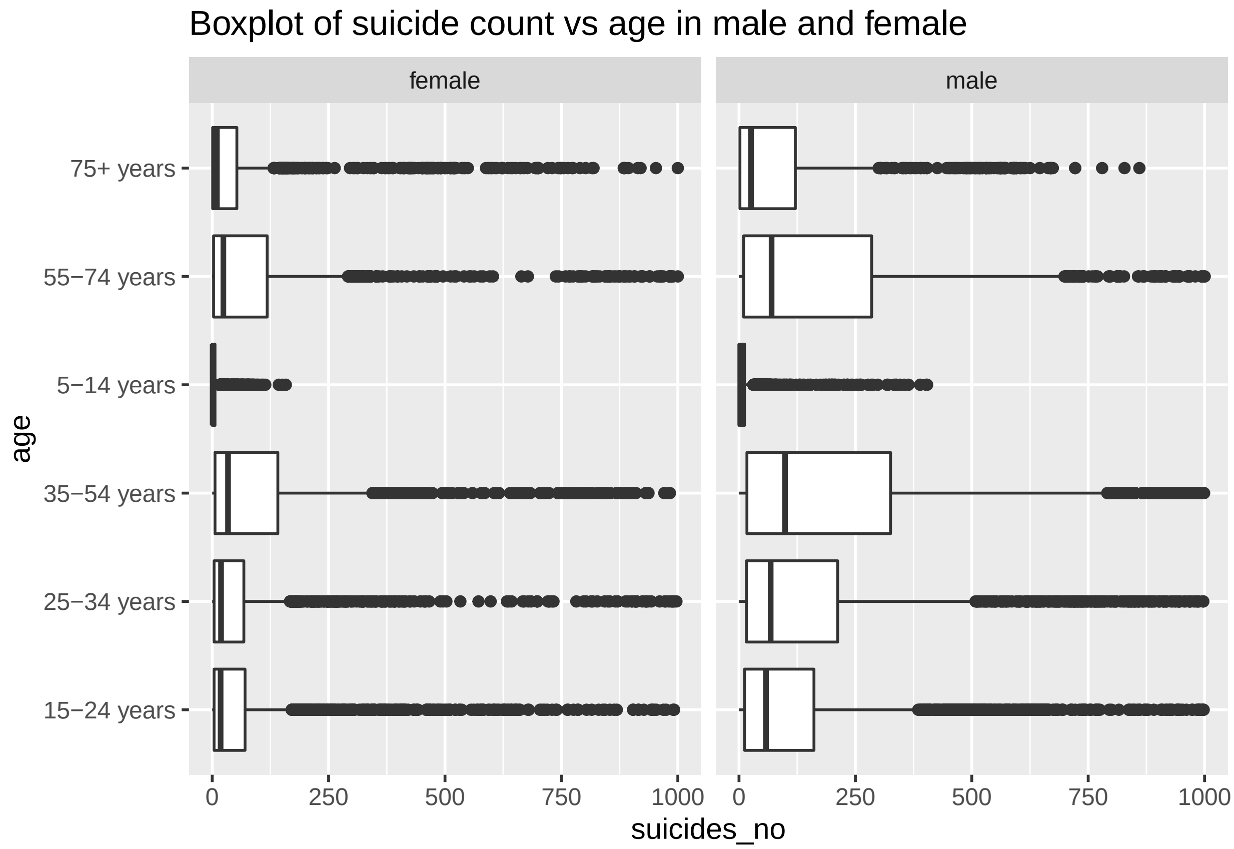


```
data %>% count(cut_interval(`gdp in $100k`, n = 10))
```

```
## # A tibble: 10 x 2
##    `cut_interval(\`gdp in $100k\`, n = 10)`     n
##    <fct>                                    <int>
##  1 [469,1.81e+07]                           26224
##  2 (1.81e+07,3.62e+07]                        888
##   3 (3.62e+07,5.44e+07]                        336
##   4 (5.44e+07,7.25e+07]                        108
##   5 (7.25e+07,9.06e+07]                         48
##   6 (9.06e+07,1.09e+08]                         48
##   7 (1.09e+08,1.27e+08]                         36
##   8 (1.27e+08,1.45e+08]                         48
##   9 (1.45e+08,1.63e+08]                         48
##  10 (1.63e+08,1.81e+08]                         36
```

```
#obs 3

ggplot(data) + geom_boxplot(aes(y = suicides_no,  x = age)) +
  facet_grid(~sex) +
 coord_flip() +
  ylim(c(0,1000)) +
  ggtitle("Boxplot of suicide count vs age in male and female")
```

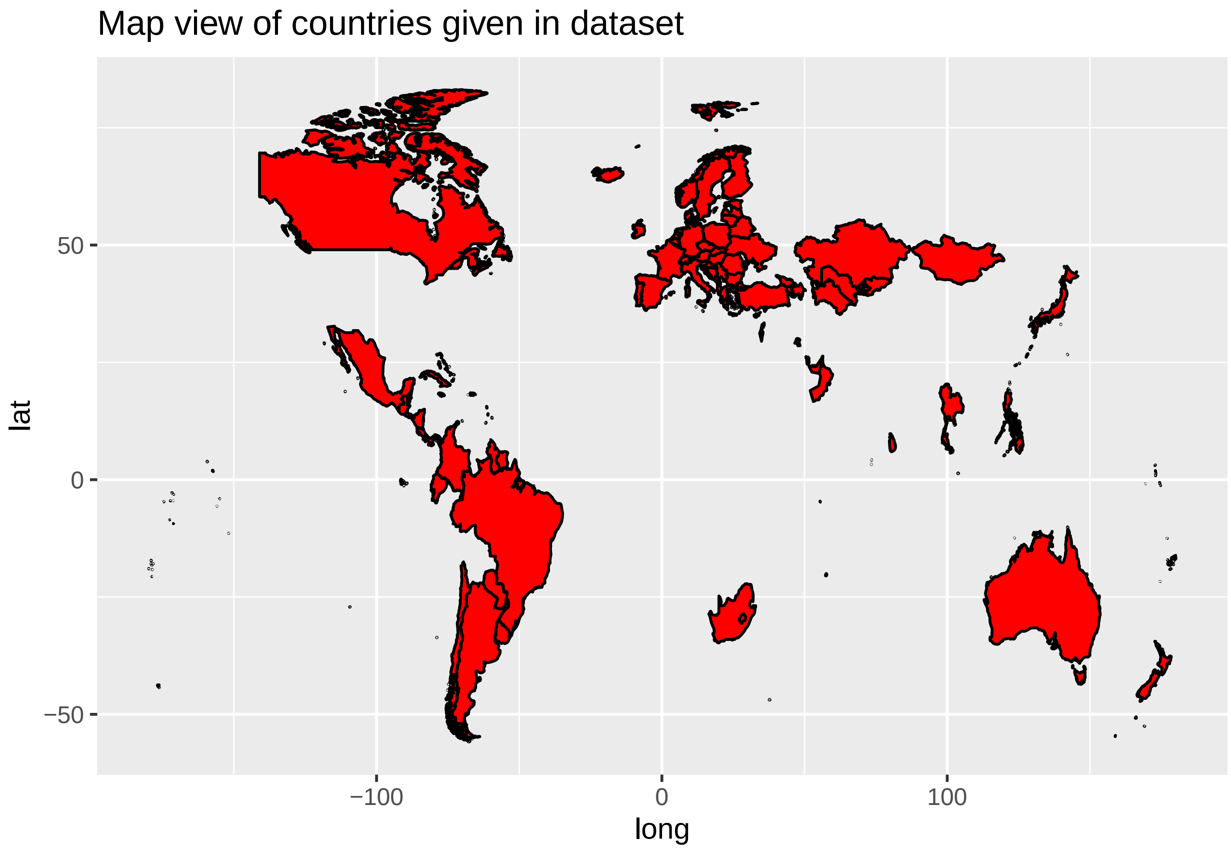
```
## Warning: Removed 1467 rows containing non-finite values (stat_boxplot).
```



There are comparetively higher no. of suicides recorded in male than female. Women are undergoing higher levels of stress at the age in between 35-74 years whereas in men higher rate is observed in between 35-54 years.

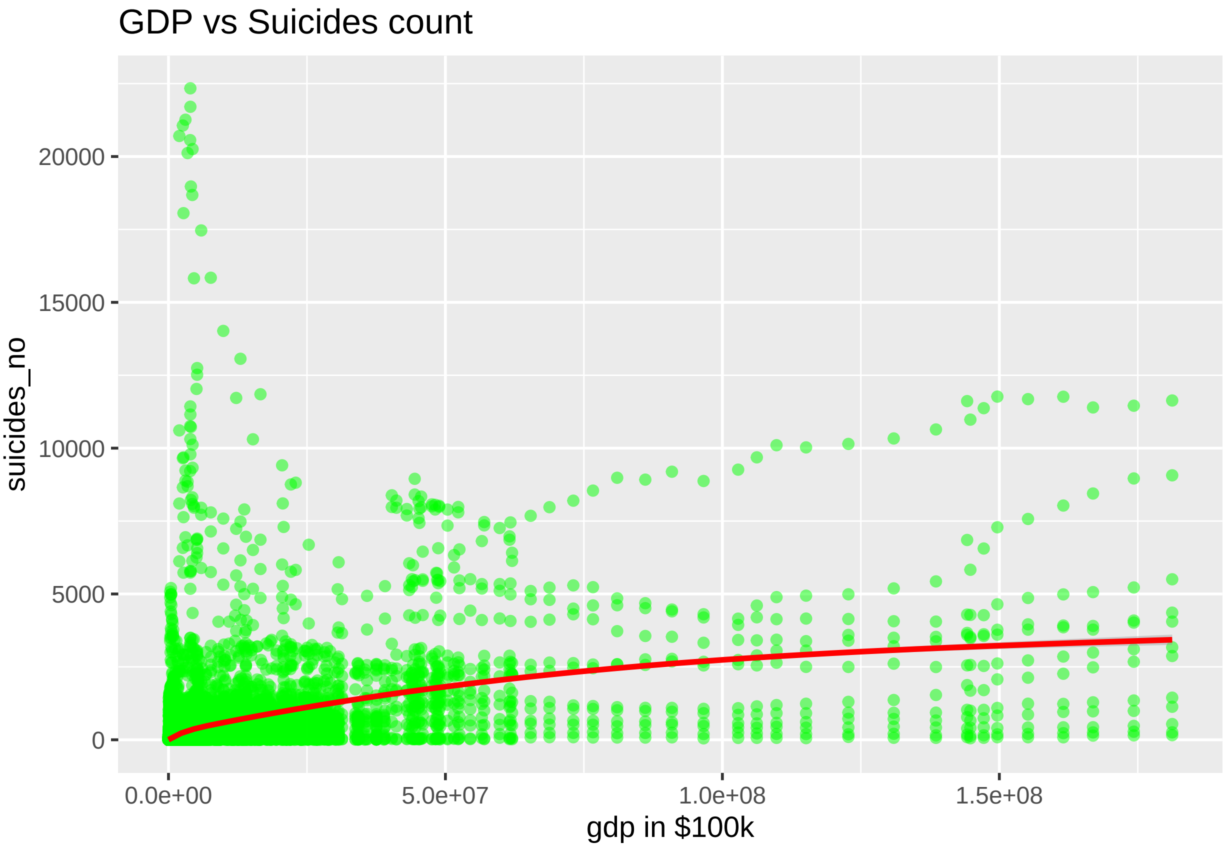
```
#obs 4

world <- ggplot2::map_data("world")
```

```r
df <- data.frame(region = c(data$country))


world_new <- world[world$region %in% df$region, ]

ggplot(world_new) +
  geom_polygon(mapping=aes(x=long, y=lat, group = group),
               fill= 'red', color = "black") +
  ggtitle("Map view of countries given in dataset")
```



```r
#obs 5
ggplot(data, mapping = aes(x = `gdp in $100k`, y = suicides_no)) +
  geom_point( position = "jitter", color = "green", alpha = 0.5) +
  geom_smooth(color = 'red') +
  ggtitle("GDP vs Suicides count")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Although there are higher no. of sucides in less earning countries, the trend seems to decrease at first and then increase proportionately with increase in gdp