Introduction to R

Second course - Answers

Jean-Baptiste Guiffard and Florence Lecuit

October 18, 2024

Load libraries

```
library(tidyverse) #tidyverse contains ggplot2 library
```

For the first two exercises, we will use a built-in dataset in R (already loaded), called **iris**. This dataset gives petal and sepal lengths and widths for different flowers.

```
head(iris) #to see the first rows of the dataset
```

Exercise 1 : Analyzing different types of variables

1. How many variables are in the dataset? What type of variables?

```
ncol(iris)
```

```
## [1] 5
```

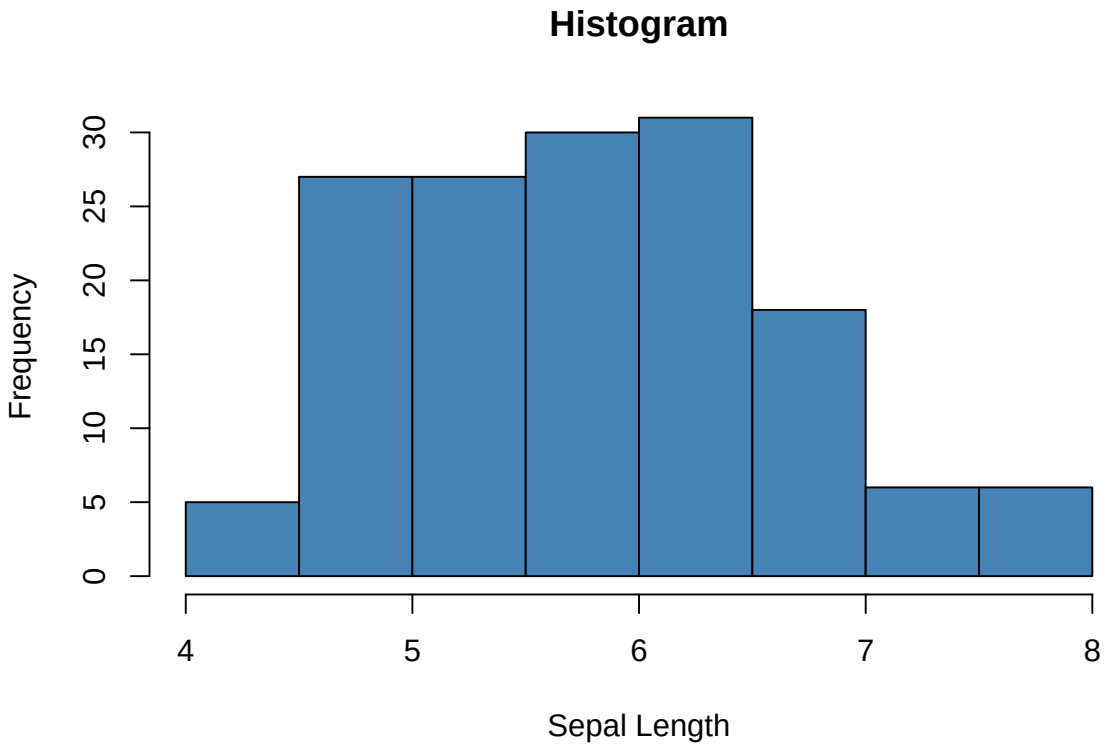
```
str(iris)
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1
```

2. What graphs can you use to observe the distribution of a numeric variable?

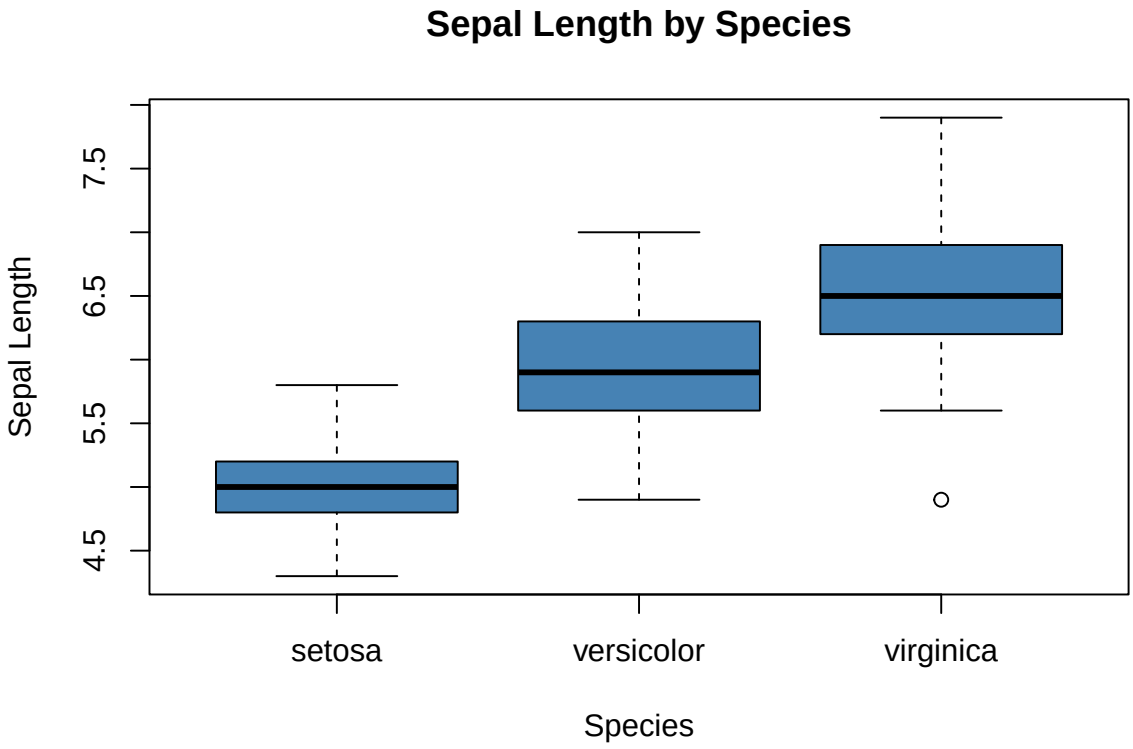
3. Create a histogram of the variable *Sepal.Length*. What does this graph show us?

```
hist(x=iris$Sepal.Length,
     col='steelblue',
     main='Histogram',
     xlab='Sepal Length',
     ylab='Frequency')
```



4. Create a boxplot of the variable *Sepal.Width*. Group by flower species. What do you observe?

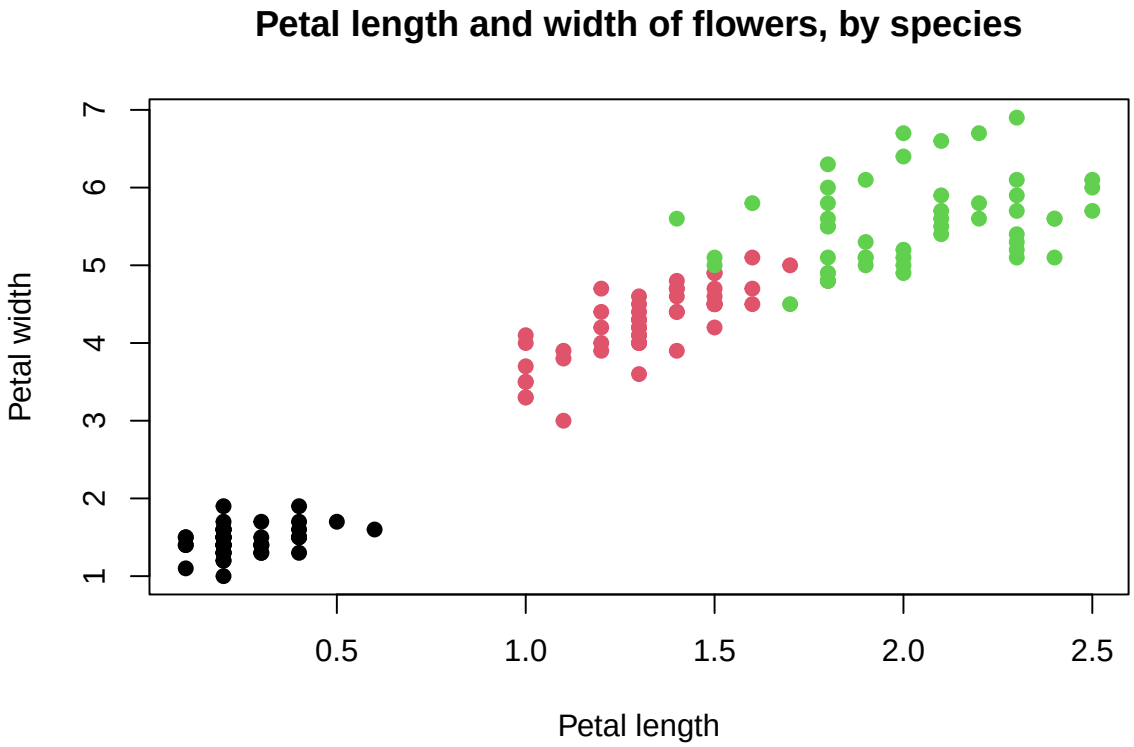
```
boxplot(Sepal.Length~Species,
        data=iris,
        main='Sepal Length by Species',
        xlab='Species',
        ylab='Sepal Length',
        col='steelblue',
        border='black')
```



Exercise 2 : Analyzing more than one numeric variable

1. Plot petal length (y) against petal width (x). What is represented in this plot? What do you observe?
2. Change the color and symbol of the points. Try to change the points by flower species, to visualize the different groups.
3. Add axis titles and a main title to the plot.

```
plot(x=iris$Petal.Width,
     y=iris$Petal.Length,
     col=iris$Species, #color by species
     pch=19, #symbol
     main="Petal length and width of flowers, by species",
     xlab="Petal length",
     ylab="Petal width")
```



Exercice 3 : GGplot2

Load dataset

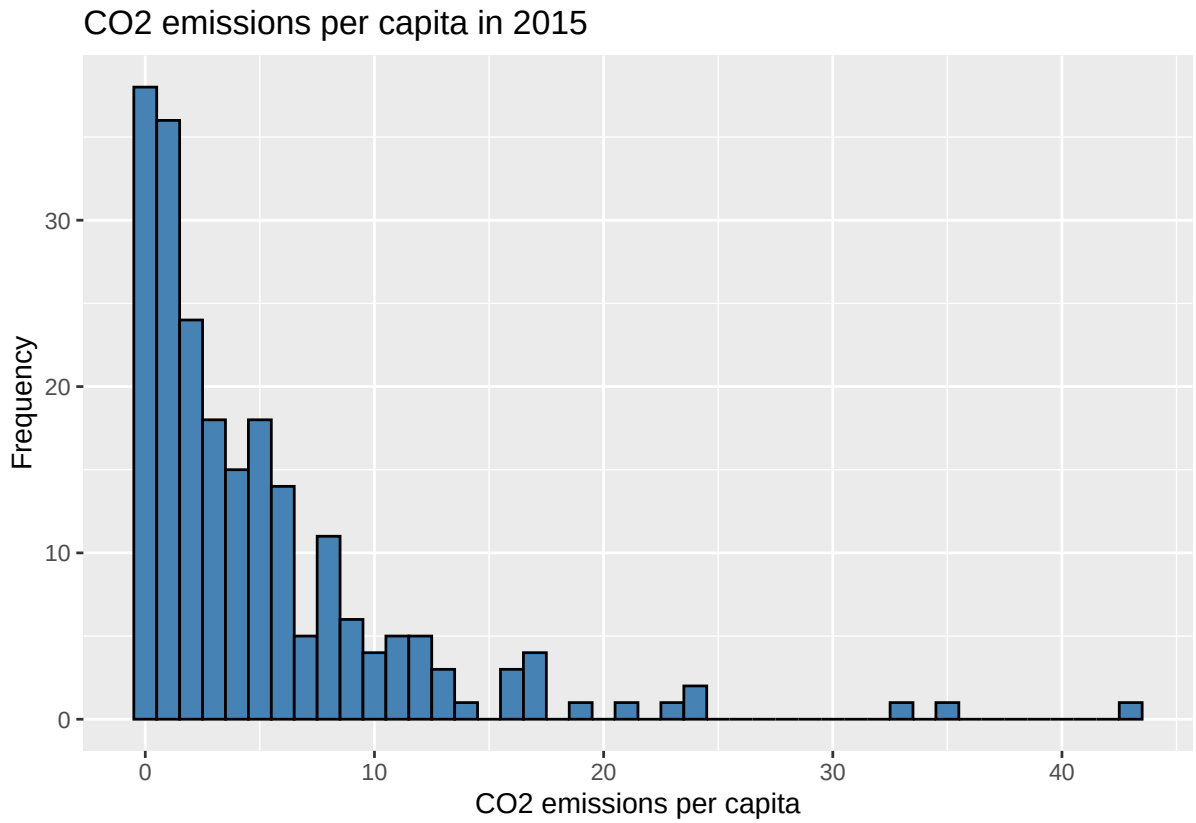
```
library(tidyverse)
data_pollution <- read.csv2('DATA/co2_clean.csv', sep=";")
```

1. Using the dataset data_pollution only for the year 2015, create a histogram showing the distribution of CO_2 emissions per capita. (Hint: Use ggplot() with geom_histogram().)

```
data_pollution_2015 <- data_pollution %>%
  filter(year == 2015)

ggplot(data=data_pollution_2015, aes(x=co2_per_capita)) +
  geom_histogram(binwidth=1, fill='steelblue', color='black') +
  labs(title='CO2 emissions per capita in 2015', x='CO2 emissions per capita', y='F
```

```
## Warning: Removed 3 rows containing non-finite outside the scale range
## (`stat_bin()`).
```

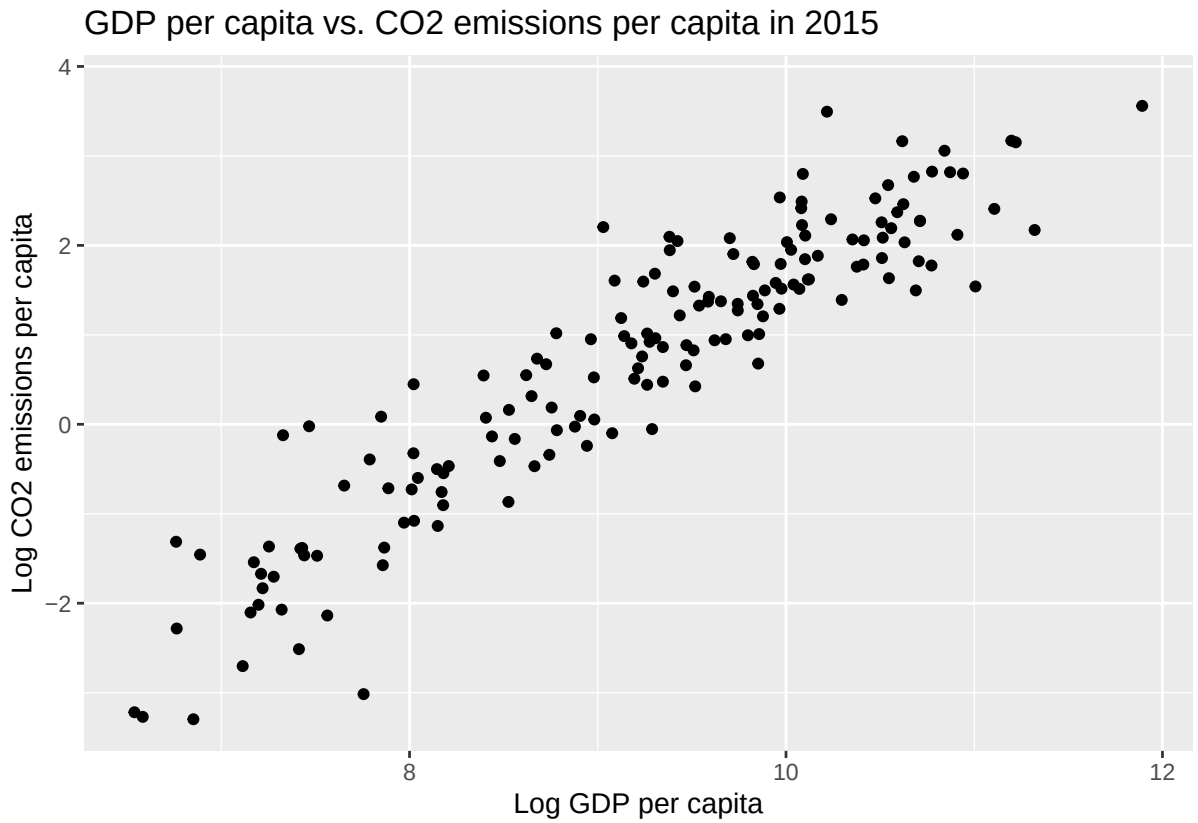


2. Create the GDP per capita variable and then create a scatter plot that shows the relationship between GDP per capita (log-transformed) and CO_2 emissions per capita (log-transformed). (Hint: Use geom_point() and log-transform the axes inside aes().)

```
data_pollution_2015 <- data_pollution_2015 %>%
  mutate(gdp_per_capita = gdp / population)

ggplot(data=data_pollution_2015, aes(x=log(gdp_per_capita), y=log(co2_per_capita)))
  geom_point() +
  labs(title='GDP per capita vs. CO2 emissions per capita in 2015', x='Log GDP per
```

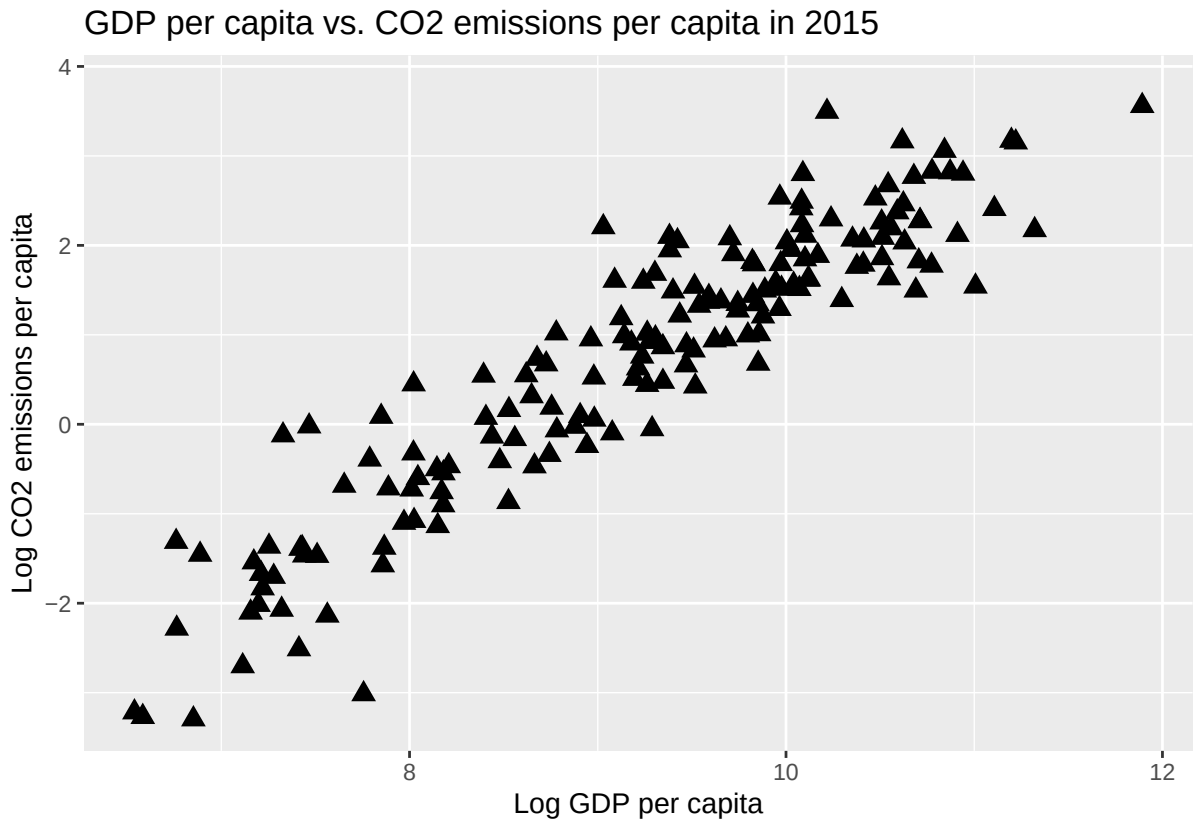
```
## Warning: Removed 57 rows containing missing values or values outside the scale r
## (`geom_point()`).
```



3. Modify the scatter plot by changing the size, shape, and color of the points. Make the points red with a black outline. (Hint: Look into geom_point() parameters like size, shape, and colour.)

```
ggplot(data=data_pollution_2015, aes(x=log(gdp_per_capita), y=log(co2_per_capita)))
  geom_point(size=3, shape=17, color='black', fill='red') +
  labs(title='GDP per capita vs. CO2 emissions per capita in 2015', x='Log GDP per
```

```
## Warning: Removed 57 rows containing missing values or values outside the scale r
## (`geom_point()`).
```



Exercice 4

```
Metadata_Country <- read.csv2('DATA/Metadata_Country.csv', sep=",")
join_pollution_wb_data <- data_pollution %>%
  dplyr::inner_join(Metadata_Country, by = c("iso_code" = "Country.Code"))
join_pollution_wb_data <- join_pollution_wb_data %>%
  filter(country != "") %>%
  filter(IncomeGroup !="")
```

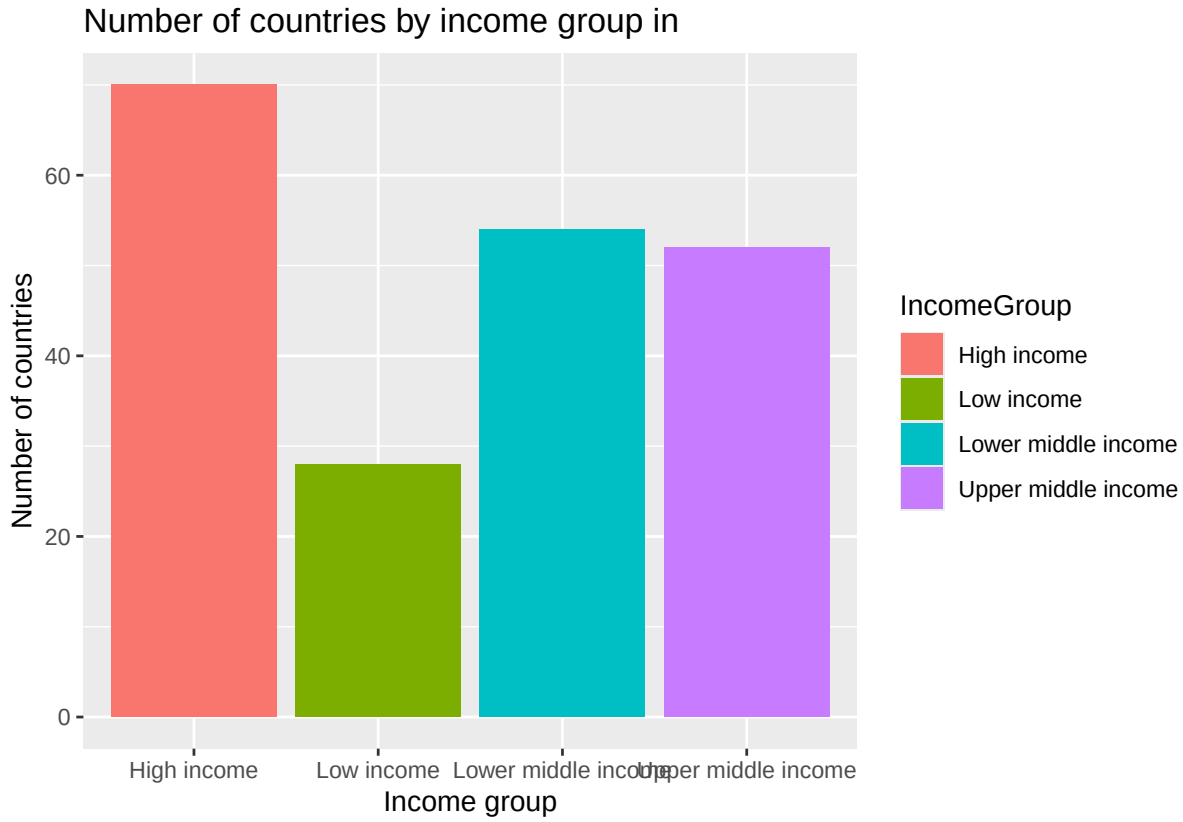
1. From this database :

- Create two variables GDP per capita and CO_2 per capita in kg ;
- Create a new database that, for the period [1990;2020], gives the average of these two variables by country;
- Delete the columns with missing values.

2. Create a bar chart showing the number of countries by their income group in 2015 (IncomeGroup). Color the bars by IncomeGroup. (Hint: Use geom_bar() with aes(fill=IncomeGroup).)

```
join_pollution_wb_2015 <- join_pollution_wb_data %>%
  filter(year ==2015)
ggplot(data=join_pollution_wb_2015, aes(x=IncomeGroup, fill=IncomeGroup)) +
```

```
  geom_bar() +
  labs(title='Number of countries by income group in ', x='Income group', y='Number
```



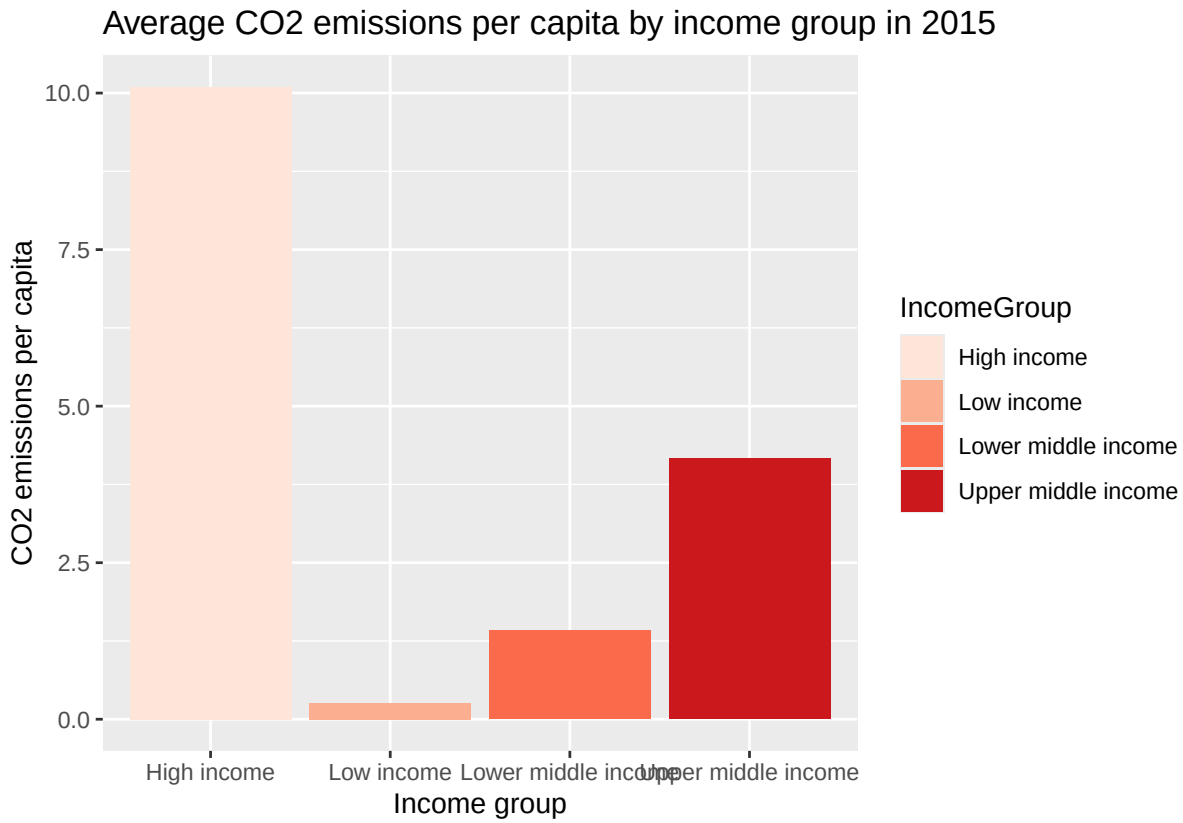
3. From the dataset in 2015, create a bar chart showing the average CO_2 emissions per capita for each income group, and color the bars using a custom palette (e.g., Reds). (Hint: Use geom_bar() and scale_fill_brewer().)

```
ggplot(data=join_pollution_wb_2015, aes(x=IncomeGroup, y=co2_per_capita, fill=Incom
  geom_bar(stat='summary', fun.y='mean') +
  scale_fill_brewer(palette='Reds') +
  labs(title='Average CO2 emissions per capita by income group in 2015', x='Income
```

```
## Warning in geom_bar(stat = "summary", fun.y = "mean"): Ignoring unknown
## parameters: `fun.y`
```

```
## Warning: Removed 1 row containing non-finite outside the scale range
## (`stat_summary()`).
```

```
## No summary function supplied, defaulting to `mean_se()`
```

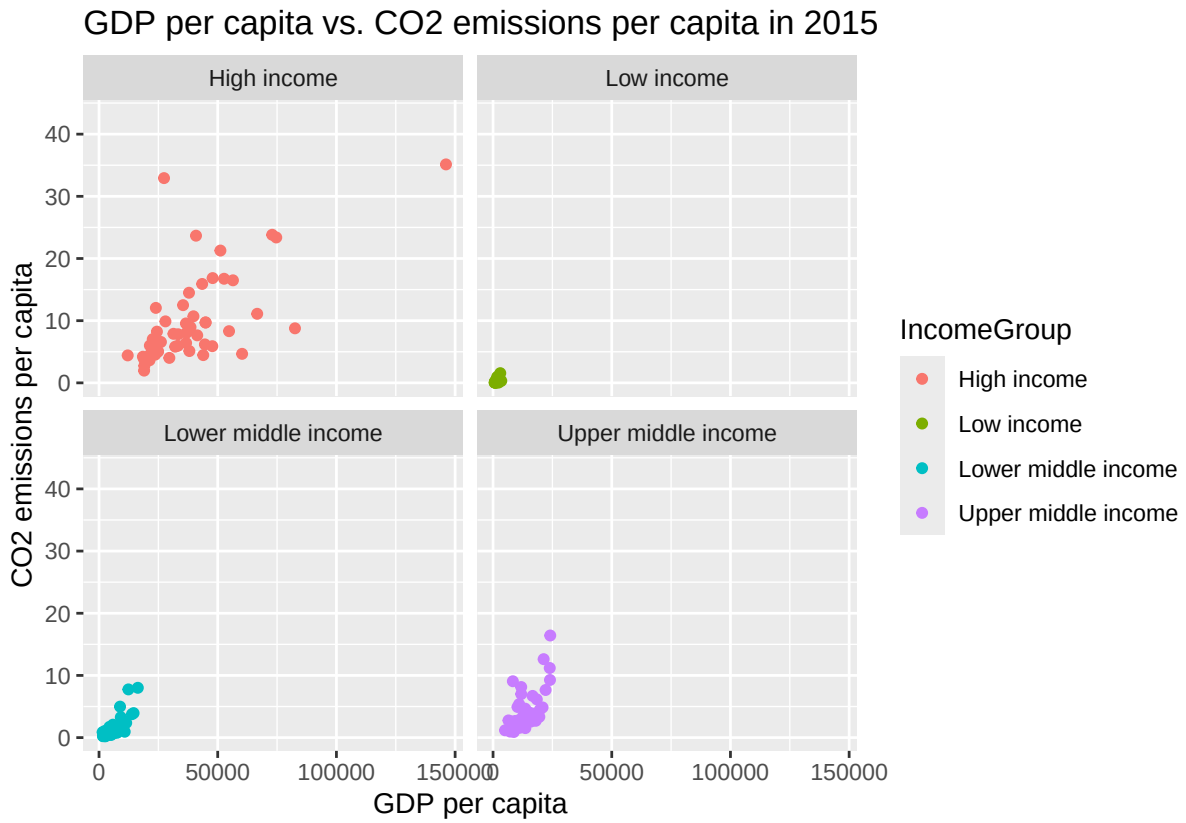


4. Use facet_wrap() to create multiple scatter plots of GDP per capita vs. CO_2 emissions per capita, one for each income group. (Hint: Facet by IncomeGroup using facet_wrap(~ IncomeGroup).)

```
join_pollution_wb_2015 <- join_pollution_wb_2015 %>%
  mutate(gdp_per_capita = gdp / population)

ggplot(data=join_pollution_wb_2015, aes(x=gdp_per_capita, y=co2_per_capita, color=I
  geom_point() +
  labs(title='GDP per capita vs. CO2 emissions per capita in 2015', x='GDP per capi
  facet_wrap(~ IncomeGroup)
```

```
## Warning: Removed 42 rows containing missing values or values outside the scale r
## (`geom_point()`).
```



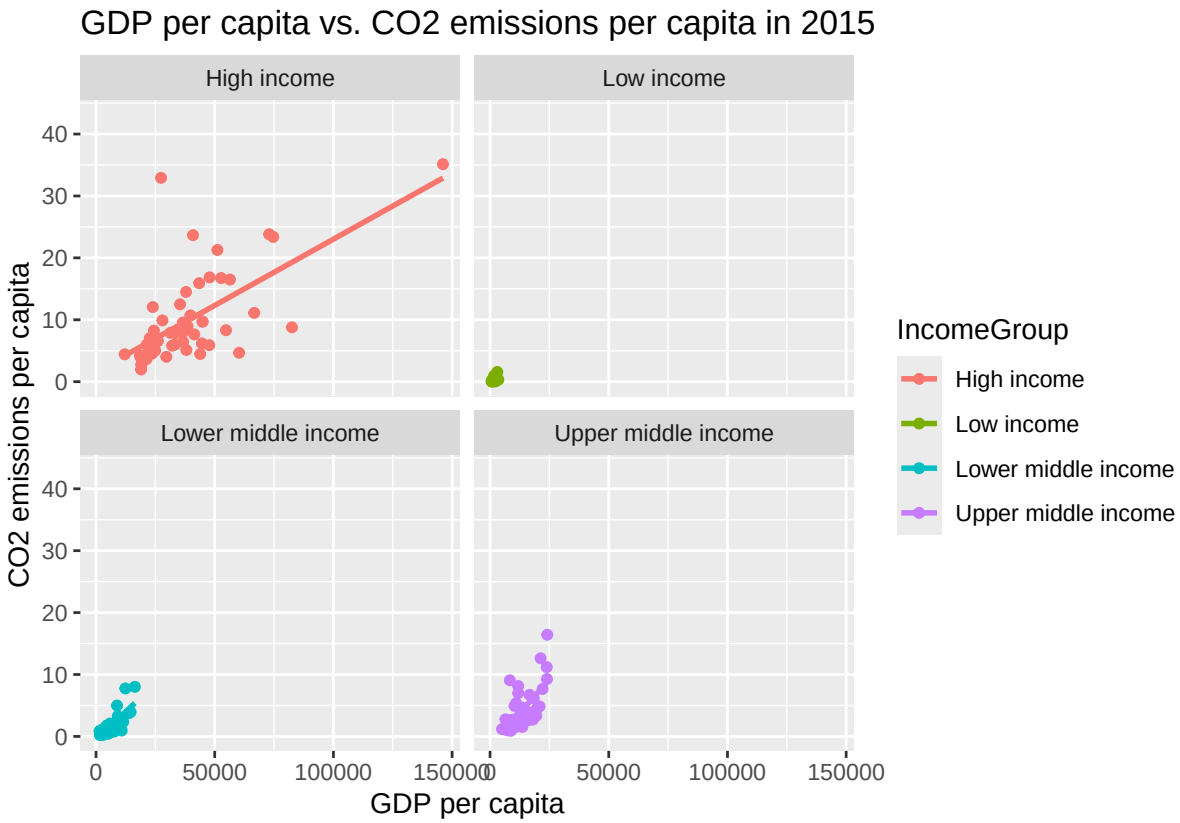
5. Add a linear regression line to the scatter plot of GDP per capita vs. CO_2 emissions per capita. Display the regression line without the confidence interval. (Hint: Use geom_smooth(method="lm", se=FALSE).)

```
ggplot(data=join_pollution_wb_2015, aes(x=gdp_per_capita, y=co2_per_capita, color=I
  geom_point() +
  geom_smooth(method='lm', se=FALSE) +
  labs(title='GDP per capita vs. CO2 emissions per capita in 2015', x='GDP per capi
  facet_wrap(~ IncomeGroup)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 42 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```

```
## Warning: Removed 42 rows containing missing values or values outside the scale r
## (`geom_point()`).
```


GDP per capita vs. CO2 emissions per capita in 2015
High income
Low income
Lower middle income
Upper middle income
CO2 emissions per capita
GDP per capita
0
10
20
30
40
0
50000
100000
150000
IncomeGroup
High income
Low income
Lower middle income
Upper middle income