# ps5_Markdown

Mengqi Shi

2023-02-13

1 Load and check data (5pt) You first task is to do a very simple data check: 1. (1pt) For solving the problems, and answering the questions, create a new rmarkdown docu- ment with an appropriate title. See https://faculty.washington.edu/otoomet/info201-book/ r-markdown.html#r-markdown-rstudio-creating.

```
library(readr)
gapminder <- read_delim("~/Desktop/INFO 201/ps5_Markdown/gapminder.csv")
```

```
## Rows: 13055 Columns: 25
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr  (6): iso3, name, iso2, region, sub-region, intermediate-region
## dbl (19): time, totalPopulation, fertilityRate, lifeExpectancy, childMortali...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

2. (2pt) Load data. How many rows/columns do we have?

There are `nrow(gapminder)` rows and `ncol(gapminder)` columns.

```
nrow(gapminder)
```

```
## [1] 13055
```

```
ncol(gapminder)
```

```
## [1] 25
```

3. (2pt) Print a small sample of data. Does it look OK? Yes.

```
head(gapminder, 3)
```

```
## # A tibble: 3 x 25
##   iso3  name  iso2  region sub-r~1 inter~2  time total~3 ferti~4 lifeE~5 child~6
##   <chr> <chr> <chr> <chr>  <chr>   <chr>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 ABW   Aruba AW    Ameri~ Latin ~ Caribb~  1960   54211    4.82    65.7      NA
## 2 ABW   Aruba AW    Ameri~ Latin ~ Caribb~  1961   55438    4.66    66.1      NA
## 3 ABW   Aruba AW    Ameri~ Latin ~ Caribb~  1962   56225    4.47    66.4      NA
```

```
## # ... with 14 more variables: youthFemaleLiteracy <dbl>,
## #   youthMaleLiteracy <dbl>, adultLiteracy <dbl>, GDP_PC <dbl>,
## #   accessElectricity <dbl>, agriculturalLand <dbl>, agricultureTractors <dbl>,
## #   cerealProduction <dbl>, fertilizerHa <dbl>, co2 <dbl>,
## #   greenhouseGases <dbl>, co2_PC <dbl>, pm2.5_35 <dbl>, battleDeaths <dbl>,
## #   and abbreviated variable names 1: `sub-region`, 2: `intermediate-region`,
## #   3: totalPopulation, 4: fertilityRate, 5: lifeExpectancy, ...
```

2 Descriptive statistics (15pt) 1. (3pt) How many countries are there in the dataset? Analyze all three: iso3, iso2 and name.

There are `length(unique(gapminder$iso3))` iso3, `length(unique(gapminder$iso2))` iso2, `length(unique(gapminder$na`

```
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
```

```
length(unique(gapminder$iso3))
```

```
## [1] 253
```

```
length(unique(gapminder$iso2))
```

```
## [1] 249
```

```
length(unique(gapminder$name))
```

```
## [1] 250
```

2. If you did this correctly, you saw that there are more names than iso-2 codes, and there are even more iso3 -codes. What is going on? Can you find it out? The name are not one-on-one, so there are more iso3 and iso2 codes.

(a) (5pt) Find how many names are there for each iso-2 code. Are there any iso-2 codes that correspond to more than one name? What are these countries? Namibia.

```
gapminder%>%
  group_by(iso2, name)%>%
  summarise(result=n(),
        isDubl=ifelse(n()>1,T,F))
```

```
## ‘summarise()‘ has grouped output by ’iso2’. You can override using the
## ‘.groups‘ argument.

## # A tibble: 250 x 4
## # Groups:   iso2 [249]
##    iso2  name                 result isDubl
##    <chr> <chr>                 <int> <lgl>
##  1 AD    Andorra                  60 TRUE
##  2 AE    United Arab Emirates     60 TRUE
##  3 AF    Afghanistan              60 TRUE
##  4 AG    Antigua and Barbuda      60 TRUE
##  5 AI    Anguilla                  1 FALSE
##  6 AL    Albania                  60 TRUE
##  7 AM    Armenia                  60 TRUE
##  8 AO    Angola                   60 TRUE
##  9 AQ    Antarctica                1 FALSE
## 10 AR    Argentina                60 TRUE
## # ... with 240 more rows
```

(b) (5pt) Now repeat the same for name and iso3-code. Are there country names that have more than one iso3-code? What are these countries? Hint: two of these entitites are CHANISL and NLD CURACAO.

These are CHANISL, GBM, KOS, and NLD CURACAO

```
gapminder%>%
   group_by(iso3, name)%>%
   summarise(result=n(),
          isDubl=ifelse(n()>1,T,F))
```

```
## ‘summarise()‘ has grouped output by ’iso3’. You can override using the
## ‘.groups‘ argument.

## # A tibble: 253 x 4
## # Groups:   iso3 [253]
##    iso3  name                 result isDubl
##    <chr> <chr>                 <int> <lgl>
##  1 ABW   Aruba                    60 TRUE
##  2 AFG   Afghanistan              60 TRUE
##  3 AGO   Angola                   60 TRUE
##  4 AIA   Anguilla                  1 FALSE
##  5 ALA   Åland Islands             1 FALSE
##  6 ALB   Albania                  60 TRUE
##  7 AND   Andorra                  60 TRUE
##  8 ARE   United Arab Emirates     60 TRUE
##  9 ARG   Argentina                60 TRUE
## 10 ARM   Armenia                  60 TRUE
## # ... with 243 more rows
```

3. (2pt) What is the minimum and maximum year in these data? The minimum year is `min(gapminder$time, na.rm = TRUE)` and maximum year is `max(gapminder$time, na.rm = TRUE)`.

```r
min(gapminder$time, na.rm = TRUE)
```

```
## [1] 1960
```

```r
max(gapminder$time, na.rm = TRUE)
```

```
## [1] 2019
```

3 CO2 emissions (30pt) Next, let's analyze CO2 emissions. 1. (2pt) How many missing co2 emissions are there for each year? Analyze both missing CO2 and co2_PC. Which years have most missing data?

```r
library(dplyr)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0     v purrr   1.0.1
## v tibble  3.1.8     v stringr 1.5.0
## v tidyr   1.3.0     v forcats 1.0.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
na_count_co2 <- gapminder %>%
  group_by(time) %>%
  summarize(count_na = sum(is.na(co2)))
na_count_co2
```

```
## # A tibble: 61 x 2
##     time count_na
##    <dbl>    <int>
##  1  1960       60
##  2  1961       60
##  3  1962       58
##  4  1963       57
##  5  1964       51
##  6  1965       51
##  7  1966       51
##  8  1967       51
##  9  1968       51
## 10  1969       51
## # ... with 51 more rows
```

```r
na_count_co2PC <- gapminder %>%
  group_by(time) %>%
  summarize(count_na = sum(is.na(co2_PC)))
na_count_co2PC
```

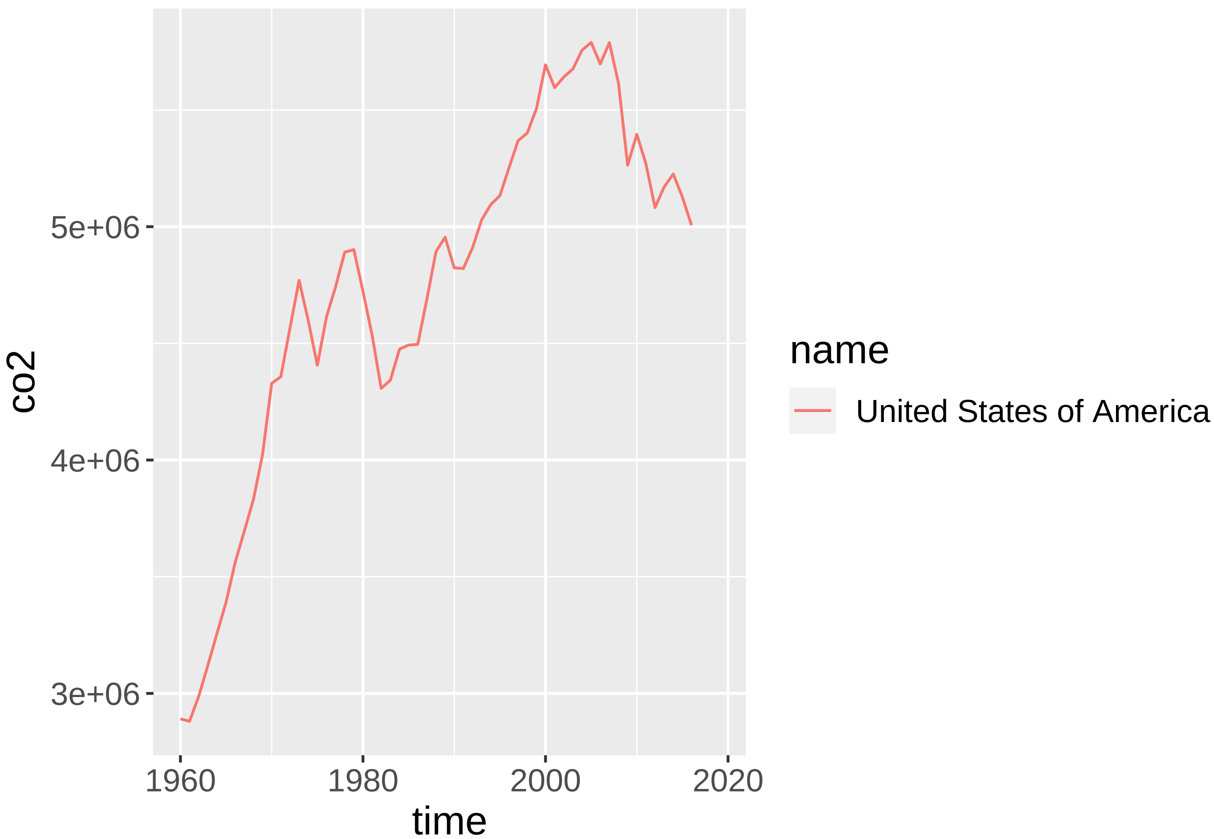
```
## # A tibble: 61 x 2
##     time count_na
##    <dbl>    <int>
```

```
## 1  1960       60
## 2  1961       60
## 3  1962       58
## 4  1963       57
## 5  1964       51
## 6  1965       51
## 7  1966       51
## 8  1967       51
## 9  1968       51
## 10  1969       51
## # ... with 51 more rows
```

2. (5pt) Make a plot of total CO2 emissions over time for the U.S, China, and India. Add a few more countries of your choice. Explain what do you see.

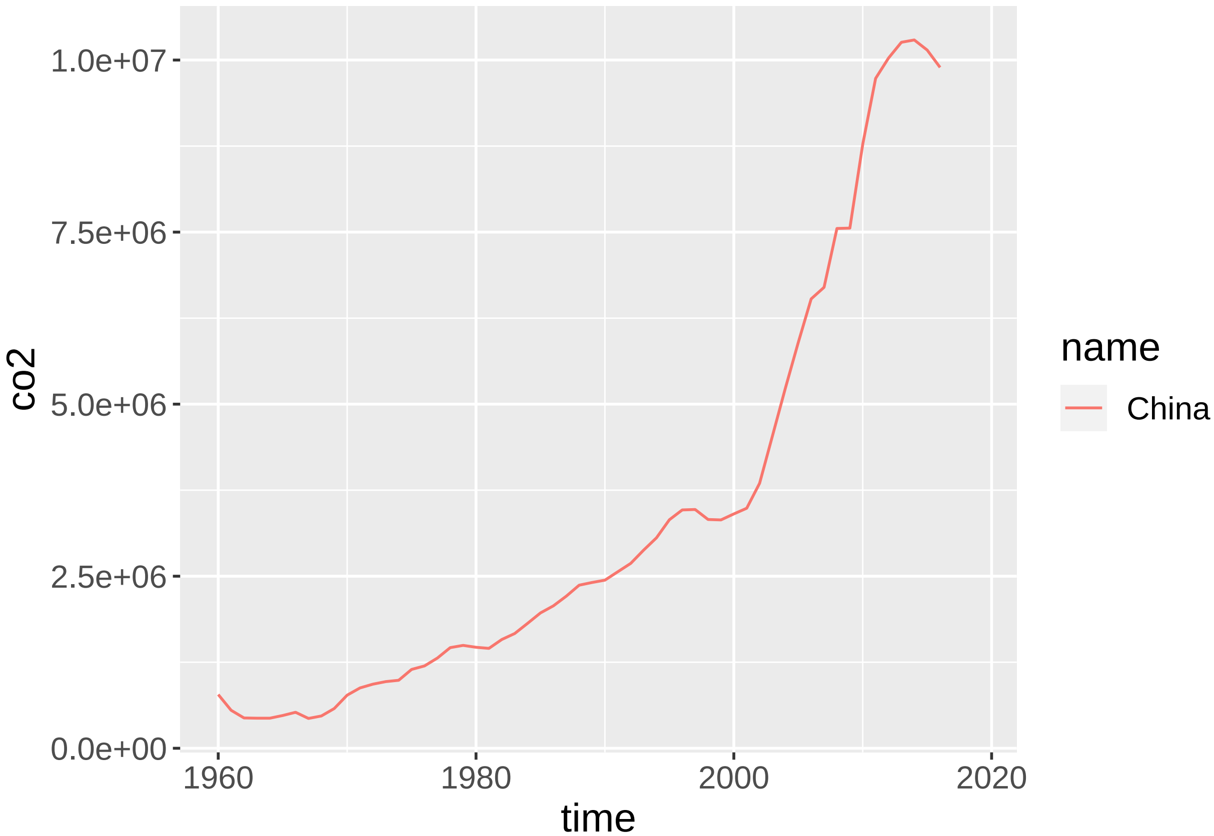
```
library(ggplot2)
gapminder%>%
  filter(name == 'United States of America') %>%
  ggplot(aes(time, co2, color=name)) +
geom_line() +
  theme(text = element_text(size=15))
```

```
## Warning: Removed 3 rows containing missing values (`geom_line()`).
```



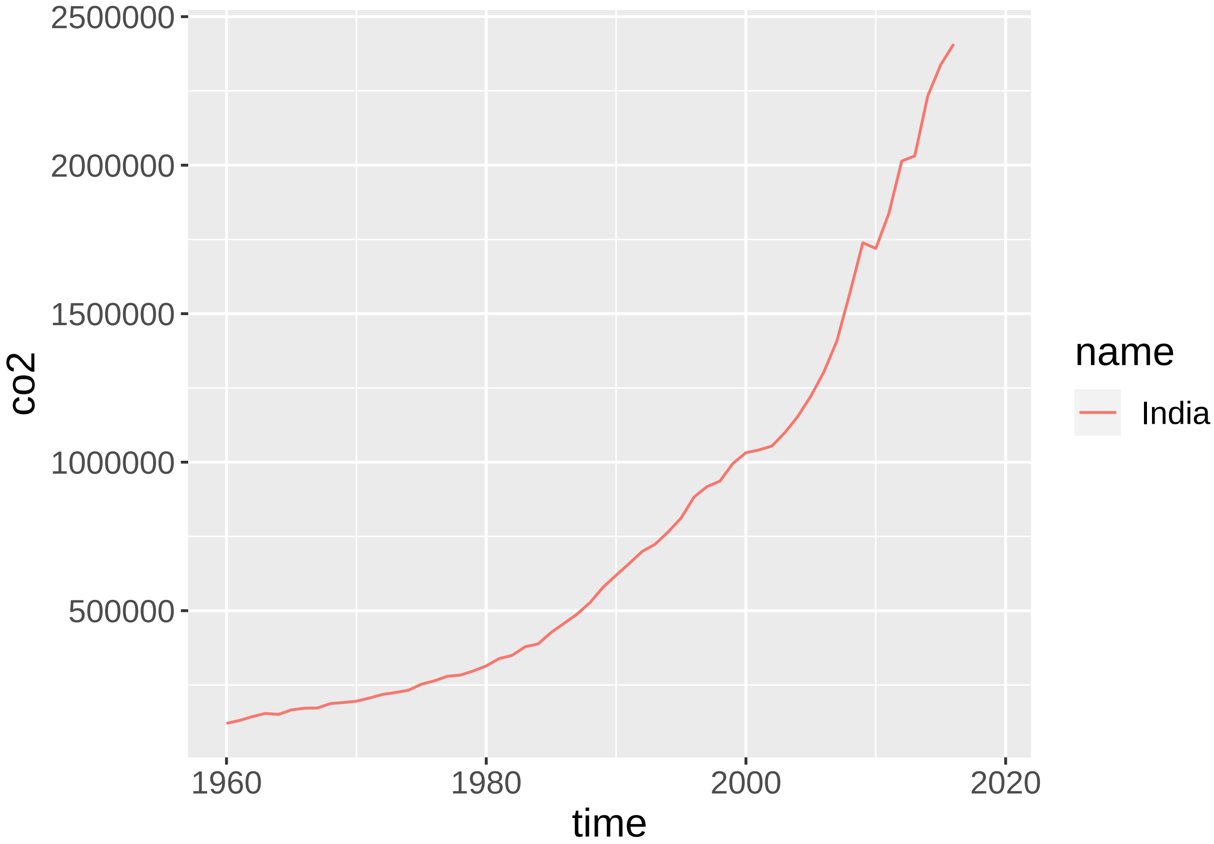
```r
gapminder%>%
  filter(name == 'China') %>%
  ggplot(aes(time, co2, color=name)) +
geom_line() +
  theme(text = element_text(size=15))
```

```
## Warning: Removed 3 rows containing missing values (‘geom_line()‘).
```




```r
gapminder%>%
  filter(name == 'India') %>%
  ggplot(aes(time, co2, color=name)) +
geom_line() +
  theme(text = element_text(size=15))
```

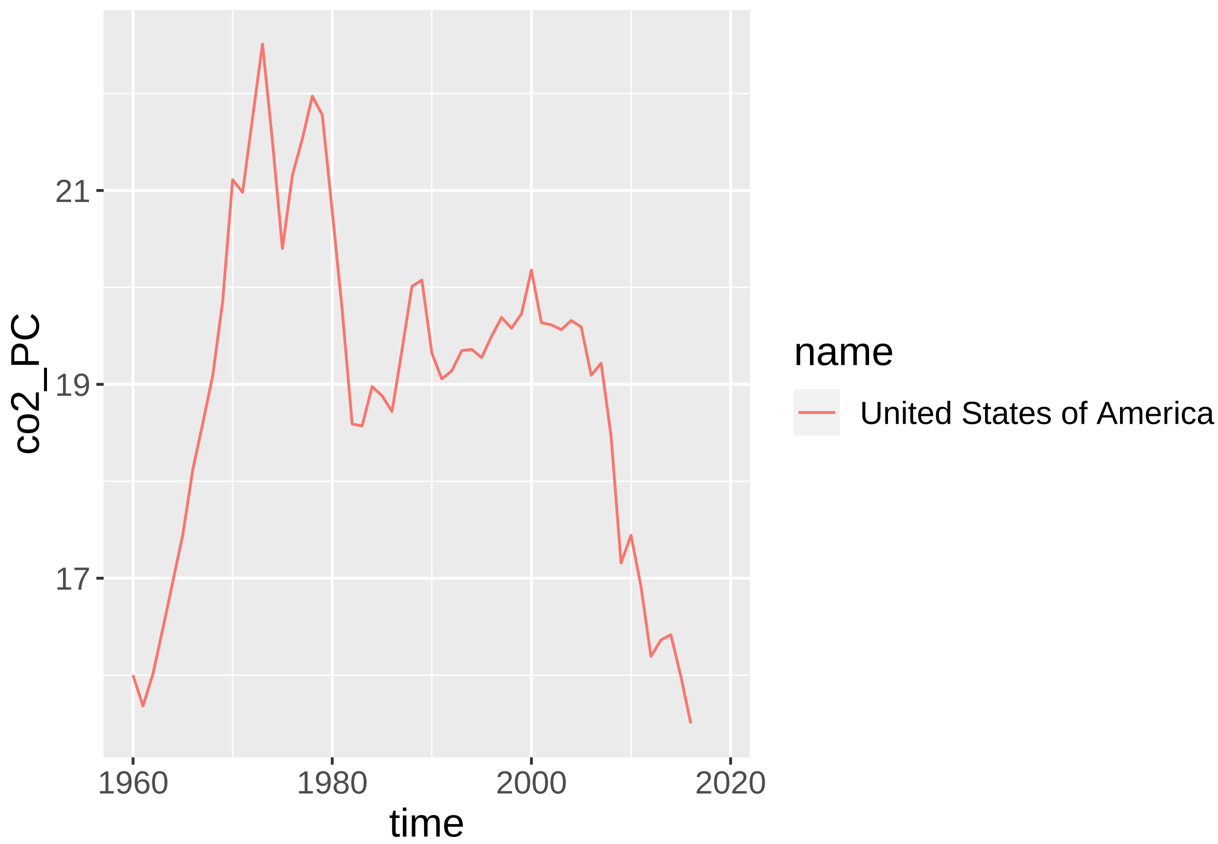
```
## Warning: Removed 3 rows containing missing values (‘geom_line()‘).
```




3. (5pt) Now let's analyze the CO2 emissions per capita (co2_PC ). Make a similar plot of the same countries. What does this figure suggest?

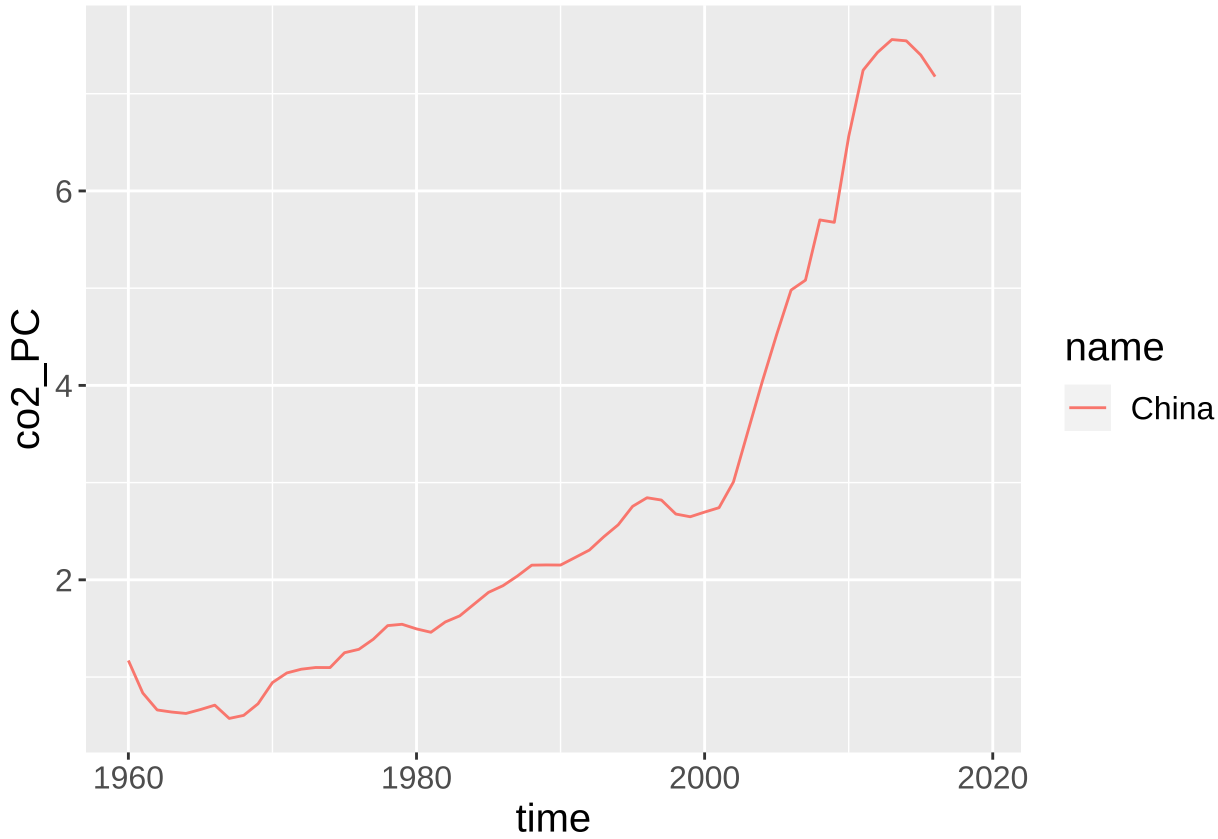
```
library(ggplot2)
gapminder%>%
  filter(name == 'United States of America') %>%
  ggplot(aes(time, co2_PC, color=name)) +
geom_line() +
  theme(text = element_text(size=15))
```

```
## Warning: Removed 3 rows containing missing values (`geom_line()`).
```



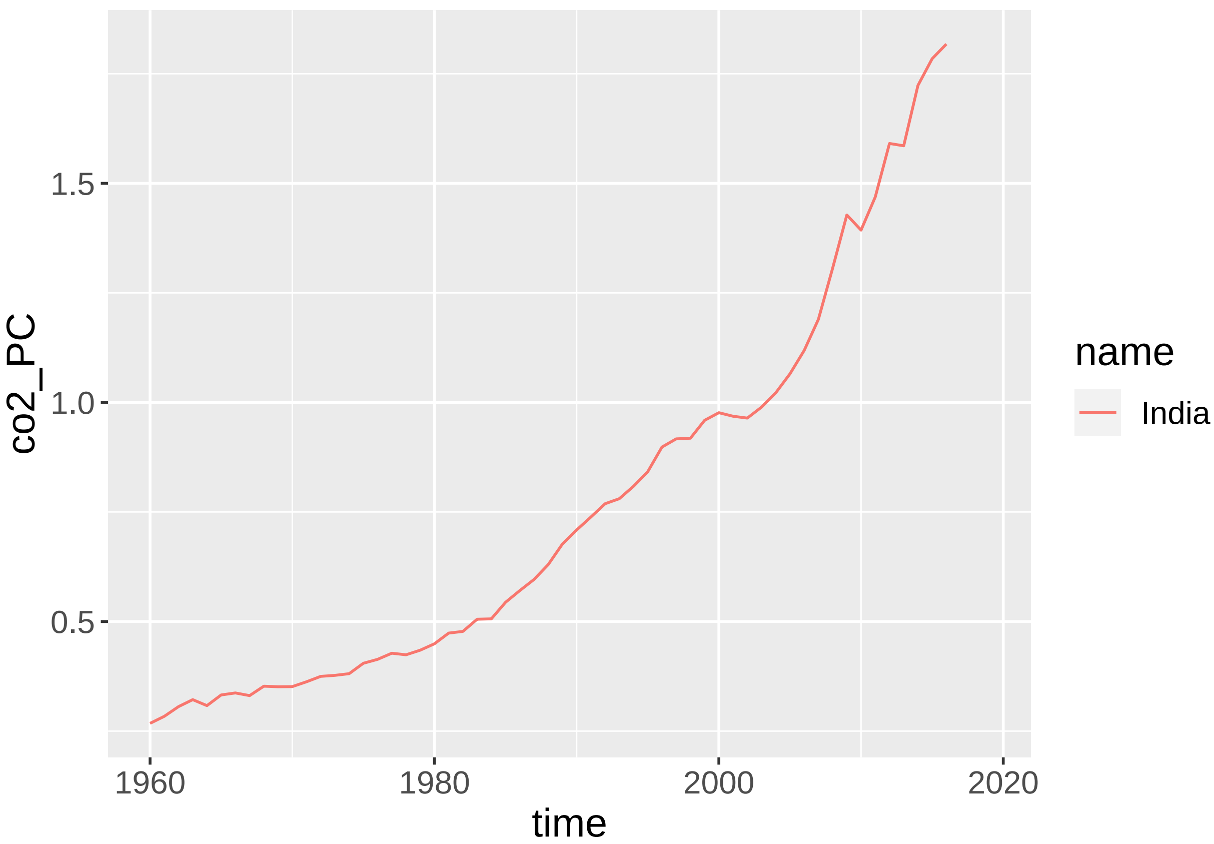

```
gapminder%>%
  filter(name == 'China') %>%
  ggplot(aes(time, co2_PC, color=name)) +
geom_line() +
  theme(text = element_text(size=15))
```

```
## Warning: Removed 3 rows containing missing values (‘geom_line()‘).
```




```
gapminder%>%
  filter(name == 'India') %>%
  ggplot(aes(time, co2_PC, color=name)) +
geom_line() +
  theme(text = element_text(size=15))
```

```
## Warning: Removed 3 rows containing missing values (`geom_line()`).
```




4. (6pt) Compute average CO2 emissions per capita across the continents (assume region is the same as continent). Comment what do you see. Note: just compute averages over countries and ignore the fact that countries are of different size. Hint: Americas 2016 should be 4.80.

```
exclude_na <- gapminder %>%
  filter(!is.na(region), !is.na(co2_PC), !is.na(totalPopulation))
exclude_na %>%
  group_by(region,time)%>%
  filter(time == 2016)%>%
  summarize(average = mean(co2_PC))
```

```
## `summarise()` has grouped output by 'region'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 5 x 3
## # Groups:   region [5]
##   region    time average
##   <chr>    <dbl>   <dbl>
## 1 Africa    2016    1.22
## 2 Americas  2016    4.80
## 3 Asia      2016    6.47
## 4 Europe    2016    6.64
## 5 Oceania   2016    4.57
```

```r
exclude_na %>%
  group_by(region, time)%>%
  filter(time == 1960)%>%
  summarize(average = mean(co2_PC))
```

```
## `summarise()` has grouped output by 'region'. You can override using the
## `.groups` argument.
```

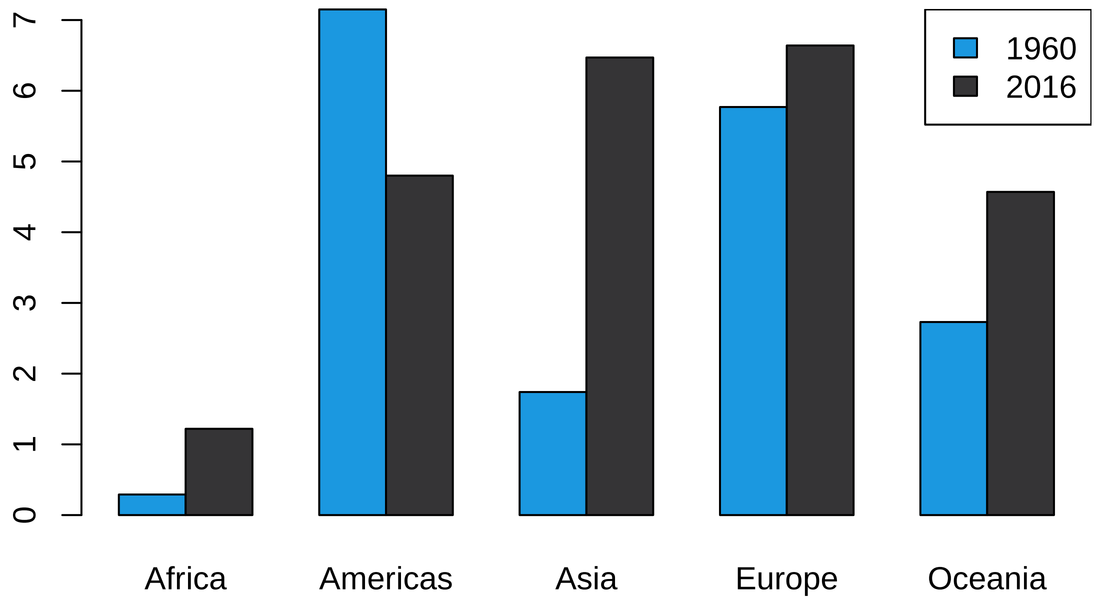
```
## # A tibble: 5 x 3
## # Groups:   region [5]
##   region    time average
##   <chr>    <dbl>   <dbl>
## 1 Africa    1960   0.291
## 2 Americas  1960   7.15
## 3 Asia      1960   1.74
## 4 Europe    1960   5.77
## 5 Oceania   1960   2.73
```

5. (7pt) Make a barplot where you show the previous results–average CO2 emissions per capita across continents in 1960 and 2016. Hint: it should look something along these lines:

```r
data <- as.matrix(data.frame(Africa = c(0.291, 1.22),
                             Americas = c(7.15, 4.80),
                             Asia = c(1.74, 6.47),
                             Europe = c(5.77, 6.64),
                             Oceania = c(2.73, 4.57)))
rownames(data) <- c("1960", "2016")
data
```

```
##      Africa Americas Asia Europe Oceania
## 1960  0.291     7.15 1.74   5.77    2.73
## 2016  1.220     4.80 6.47   6.64    4.57
```

```r
barplot(data,
        col = c("#1B98E0", "#353436"),
        beside = TRUE)
legend("topright",
       legend = c("1960", "2016"),
       fill = c("#1B98E0", "#353436"))
```



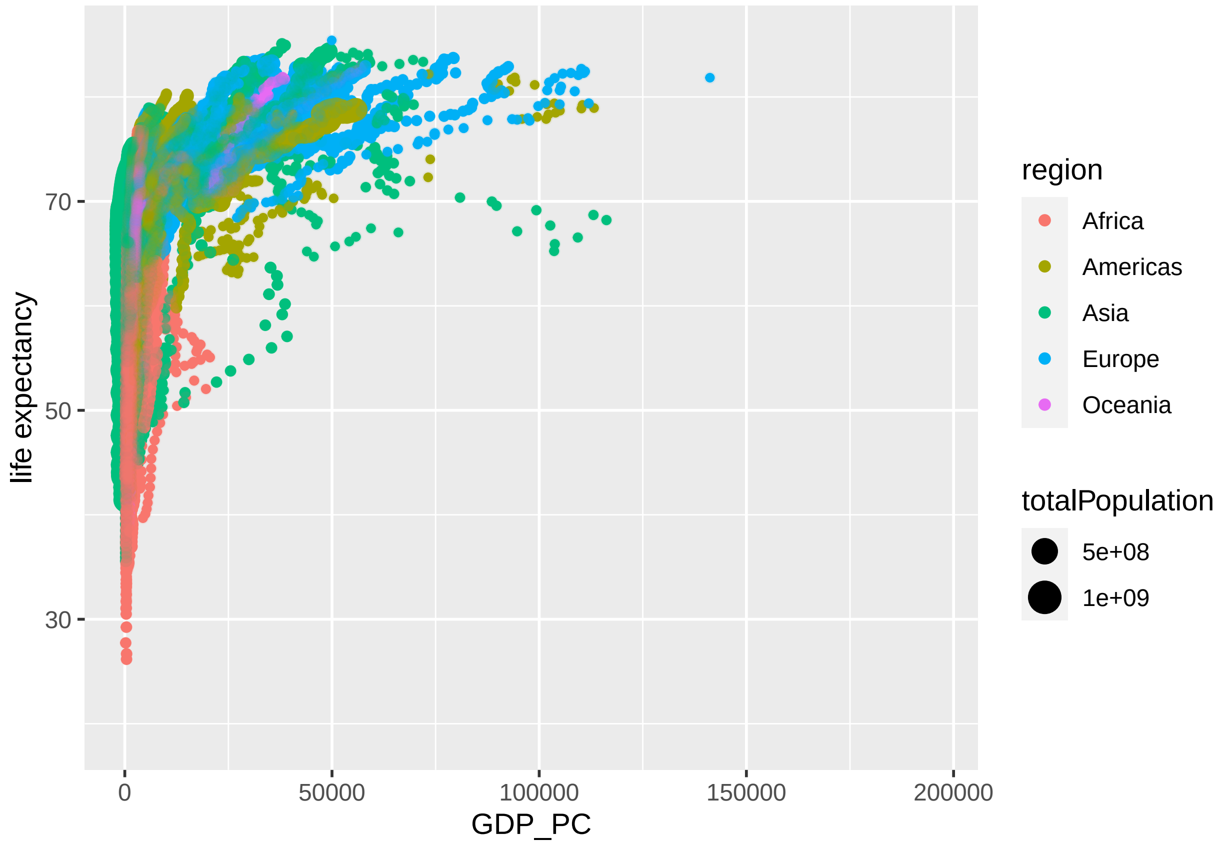

6. Which countries are the three largest, and three smallest CO2 emitters (in terms of CO2 per capita) in 2019 for each continent? (Assume region is continent).

```
##Q6
```

4 GDP per capita (50pt) Let's look at GDP per capita (GDP_PC ). 1. (8pt) Make a scatterplot of GDP per capita versus life expectancy by country, using data for 1960. Make the point size dependent on the country size, and color those according to the continent. Feel free to adjust the plot in other ways to make it better. Comment what do you see there.

```
library(ggplot2)
gdp_lifeexpectancy <- ggplot(data = subset(gapminder, !is.na(region), !is.na(name)), aes(x = GDP_PC, y =
  geom_point(aes(size = totalPopulation)) +
  geom_point(alpha = 0.1) +
  xlab("GDP_PC") +
  ylab("life expectancy")
gdp_lifeexpectancy
```

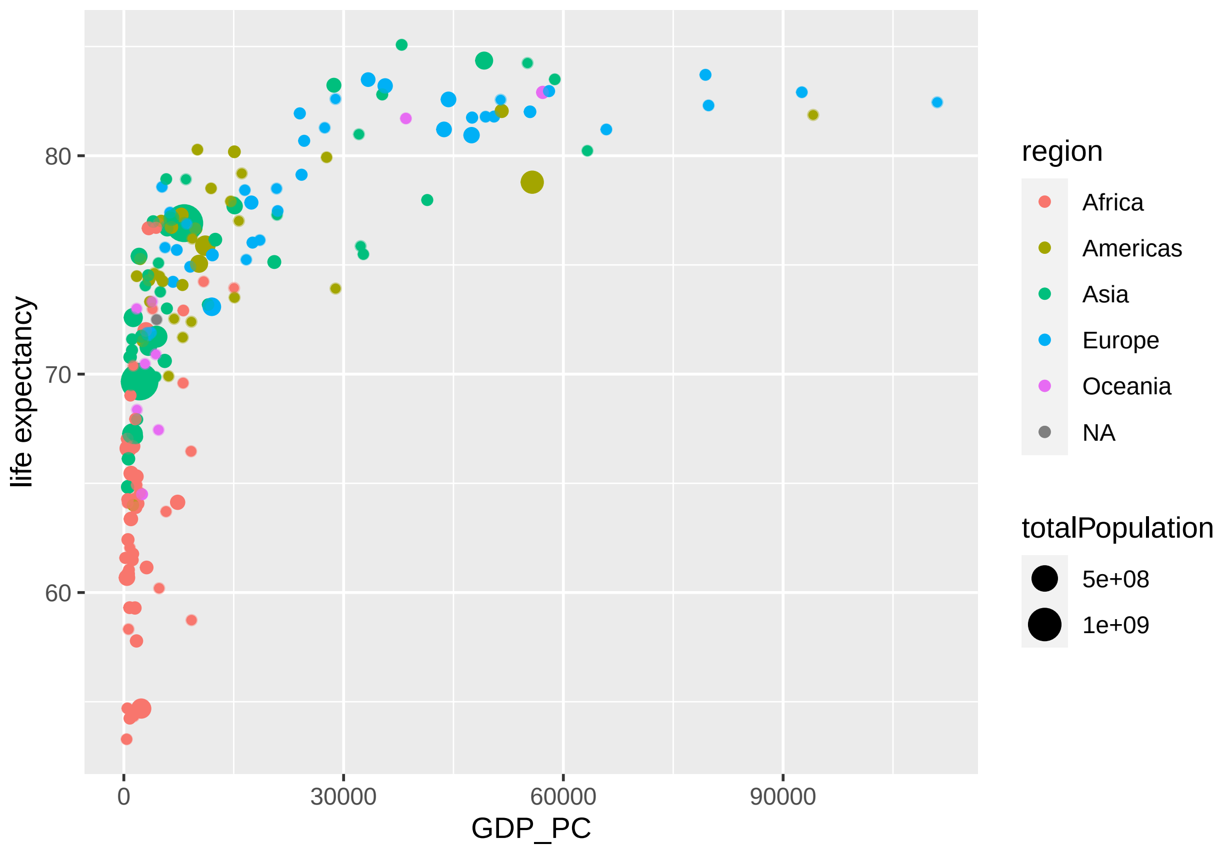
```
## Warning: Removed 3814 rows containing missing values (`geom_point()`).
## Removed 3814 rows containing missing values (`geom_point()`).
```




2. (4pt) Make a similar plot, but this time use 2019 data only.

```
gdp_lifeexpectancy2 <- ggplot(data = filter(gapminder, time == 2019), aes(x = GDP_PC, y = lifeExpectanc
  geom_point(aes(size = totalPopulation)) +
  geom_point(alpha = 0.4) +
  xlab("GDP_PC") +
  ylab("life expectancy")
gdp_lifeexpectancy2
```

```
## Warning: Removed 38 rows containing missing values (‘geom_point()‘).
## Removed 38 rows containing missing values (‘geom_point()‘).
```




3. (6pt) Compare these two plots and comment what do you see. How has world developed through the last 60 years?

4. (6pt) Compute the average life expectancy for each continent in 1960 and 2019. Do the results fit with what do you see on the figures? Note: here as average I mean just average over countries, ignore the fact that countries are of different size.

```
exclude_na_life <- gapminder %>%
  filter(!is.na(region), !is.na(lifeExpectancy))
exclude_na_life %>%
  group_by(region, time)%>%
  filter(time == 1960)%>%
  summarize(average = mean(lifeExpectancy))
```

```
## `summarise()` has grouped output by 'region'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 5 x 3
## # Groups:   region [5]
##   region    time average
##   <chr>    <dbl>   <dbl>
## 1 Africa    1960    41.5
## 2 Americas  1960    58.6
## 3 Asia      1960    51.6
## 4 Europe    1960    68.3
## 5 Oceania   1960    56.4
```

```
exclude_na_life
```

```
## # A tibble: 11,618 x 25
##    iso3  name  iso2  region   `sub-region` inter~1  time total~2 ferti~3 lifeE~4
##    <chr> <chr> <chr> <chr>    <chr>        <chr>   <dbl>   <dbl>   <dbl>   <dbl>
##  1 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1960   54211    4.82    65.7
##  2 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1961   55438    4.66    66.1
##  3 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1962   56225    4.47    66.4
##  4 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1963   56695    4.27    66.8
##  5 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1964   57032    4.06    67.1
##  6 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1965   57360    3.84    67.4
##  7 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1966   57715    3.62    67.8
##  8 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1967   58055    3.42    68.1
##  9 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1968   58386    3.23    68.4
## 10 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1969   58726    3.05    68.8
## # ... with 11,608 more rows, 15 more variables: childMortality <dbl>,
## #   youthFemaleLiteracy <dbl>, youthMaleLiteracy <dbl>, adultLiteracy <dbl>,
## #   GDP_PC <dbl>, accessElectricity <dbl>, agriculturalLand <dbl>,
## #   agricultureTractors <dbl>, cerealProduction <dbl>, fertilizerHa <dbl>,
## #   co2 <dbl>, greenhouseGases <dbl>, co2_PC <dbl>, pm2.5_35 <dbl>,
## #   battleDeaths <dbl>, and abbreviated variable names
## #   1: `intermediate-region`, 2: totalPopulation, 3: fertilityRate, ...
```

```
exclude_na_life %>%
  group_by(region, time)%>%
  filter(time == 2019)%>%
  summarize(average = mean(lifeExpectancy))
```

```
## `summarise()` has grouped output by 'region'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 5 x 3
## # Groups:   region [5]
##   region    time average
##   <chr>    <dbl>   <dbl>
## 1 Africa    2019    64.1
## 2 Americas  2019    75.8
## 3 Asia      2019    74.6
## 4 Europe    2019    79.4
## 5 Oceania   2019    73.5
```

```
exclude_na_life
```

```
## # A tibble: 11,618 x 25
##    iso3  name  iso2  region   `sub-region` inter~1  time total~2 ferti~3 lifeE~4
##    <chr> <chr> <chr> <chr>    <chr>        <chr>   <dbl>   <dbl>   <dbl>   <dbl>
##  1 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1960   54211    4.82    65.7
##  2 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1961   55438    4.66    66.1
##  3 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1962   56225    4.47    66.4
##  4 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1963   56695    4.27    66.8
##  5 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1964   57032    4.06    67.1
```

```
##  6 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1965   57360    3.84    67.4
##  7 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1966   57715    3.62    67.8
##  8 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1967   58055    3.42    68.1
##  9 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1968   58386    3.23    68.4
## 10 ABW   Aruba AW    Americas Latin Ameri~ Caribb~  1969   58726    3.05    68.8
## # ... with 11,608 more rows, 15 more variables: childMortality <dbl>,
## #   youthFemaleLiteracy <dbl>, youthMaleLiteracy <dbl>, adultLiteracy <dbl>,
## #   GDP_PC <dbl>, accessElectricity <dbl>, agriculturalLand <dbl>,
## #   agricultureTractors <dbl>, cerealProduction <dbl>, fertilizerHa <dbl>,
## #   co2 <dbl>, greenhouseGases <dbl>, co2_PC <dbl>, pm2.5_35 <dbl>,
## #   battleDeaths <dbl>, and abbreviated variable names
## #   1: ‘intermediate-region‘, 2: totalPopulation, 3: fertilityRate, ...
```

```r
cat("Q4: Yes")
```

```
## Q4: Yes
```

5. (8pt) Compute the average LE growth from 1960-2019 across the continents. Show the results in the order of growth. Explain what do you see. Hint: these data (data in long form) is not the simplest to compute growth. But you may want to check out the lag() function. And do not forget to group data by continent when using lag(), otherwise your results will be messed up! See https://faculty.washington.edu/otoomet/info201-book/dplyr.html#dplyr-helpers-compute.

6. (6pt) Show the histogram of GDP per capita for years of 1960 and 2019. Try to put both histograms on the same graph, see how well you can do it!

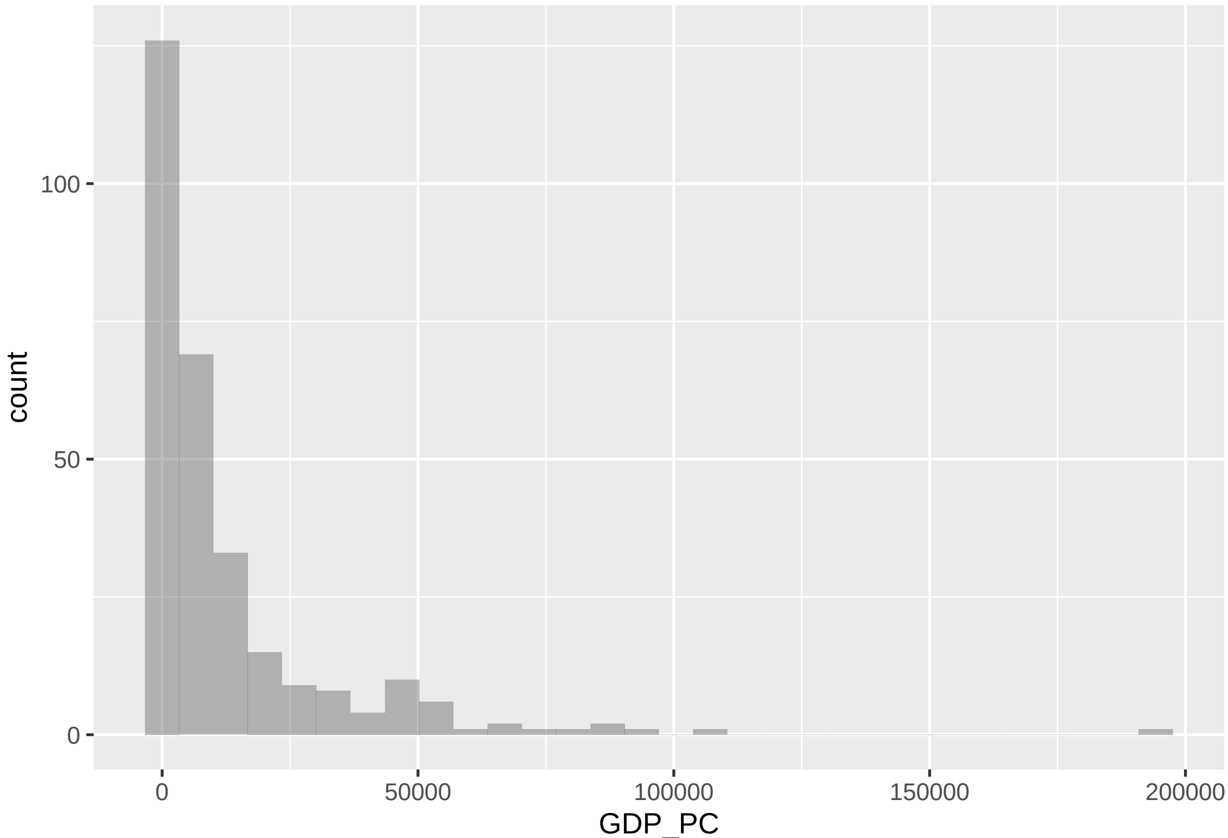
```r
plot_data <-
  gapminder %>%
    filter(time == 1960 | time == 2016)

gdp <- ggplot(plot_data, aes(x = GDP_PC, fill = time)) +
  geom_histogram(position = "identity", alpha = 0.4) +
  xlab("GDP_PC")
gdp
```

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

```
## Warning: Removed 143 rows containing non-finite values (‘stat_bin()‘).
```

```
## Warning: The following aesthetics were dropped during statistical transformation: fill
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a ‘group‘ aesthetic or to convert a numerical
##   variable into a factor?
```




7. (6pt) What was the ranking of US in terms of life expectancy in 1960 and in 2019? (When counting from top.) Hint: check out the function rank()! Hint2: 17 for 1960.

The rank in 2019 is 46 and in 1960 is 17

```
gapminder %>%
  filter(time == 2019)%>%
  arrange(lifeExpectancy) %>%
  group_by(name)%>%
  mutate(rank = rank(lifeExpectancy))
```

```
## # A tibble: 217 x 26
## # Groups:   name [214]
##    iso3  name          iso2  region sub-r~1 inter~2  time total~3 ferti~4 lifeE~5
##    <chr> <chr>         <chr> <chr>  <chr>   <chr>   <dbl>   <dbl>   <dbl>   <dbl>
##  1 CAF   Central Afr~  CF    Africa Sub-Sa~ Middle~  2019  4.75e6    4.64    53.3
##  2 TCD   Chad          TD    Africa Sub-Sa~ Middle~  2019  1.59e7    5.65    54.2
##  3 LSO   Lesotho       LS    Africa Sub-Sa~ Southe~  2019  2.13e6    3.11    54.3
##  4 NGA   Nigeria       NG    Africa Sub-Sa~ Wester~  2019  2.01e8    5.32    54.7
##  5 SLE   Sierra Leone  SL    Africa Sub-Sa~ Wester~  2019  7.81e6    4.17    54.7
##  6 SOM   Somalia       SO    Africa Sub-Sa~ Easter~  2019  1.54e7    5.98    57.4
##  7 CIV   Côte d'Ivoi~  CI    Africa Sub-Sa~ Wester~  2019  2.57e7    4.59    57.8
##  8 SSD   South Sudan   SS    Africa Sub-Sa~ Easter~  2019  1.11e7    4.62    57.8
##  9 GNB   Guinea-Biss~  GW    Africa Sub-Sa~ Wester~  2019  1.92e6    4.40    58.3
## 10 GNQ   Equatorial ~  GQ    Africa Sub-Sa~ Middle~  2019  1.36e6    4.43    58.7
```

```
## # ... with 207 more rows, 16 more variables: childMortality <dbl>,
## #   youthFemaleLiteracy <dbl>, youthMaleLiteracy <dbl>, adultLiteracy <dbl>,
## #   GDP_PC <dbl>, accessElectricity <dbl>, agriculturalLand <dbl>,
## #   agricultureTractors <dbl>, cerealProduction <dbl>, fertilizerHa <dbl>,
## #   co2 <dbl>, greenhouseGases <dbl>, co2_PC <dbl>, pm2.5_35 <dbl>,
## #   battleDeaths <dbl>, rank <dbl>, and abbreviated variable names
## #   1: `sub-region`, 2: `intermediate-region`, 3: totalPopulation, ...
```

```
gapminder %>%
  filter(time == 1960)%>%
  arrange(lifeExpectancy) %>%
  group_by(name)%>%
  mutate(rank = rank(lifeExpectancy))
```

```
## # A tibble: 216 x 26
## # Groups:   name [213]
##    iso3  name         iso2  region sub-r~1 inter~2  time total~3 ferti~4 lifeE~5
##    <chr> <chr>        <chr> <chr>  <chr>   <chr>   <dbl>   <dbl>   <dbl>   <dbl>
##  1 MLI   Mali         ML    Africa Sub-Sa~ Wester~  1960 5263733    6.97    28.2
##  2 YEM   Yemen        YE    Asia   Wester~ <NA>     1960 5315355    7.94    29.9
##  3 SLE   Sierra Leone SL    Africa Sub-Sa~ Wester~  1960 2317636    6.13    31.6
##  4 SSD   South Sudan  SS    Africa Sub-Sa~ Easter~  1960 2842724    6.72    31.7
##  5 GMB   Gambia       GM    Africa Sub-Sa~ Wester~  1960  365047    6.25    32.1
##  6 AFG   Afghanistan  AF    Asia   Southe~ <NA>     1960 8996973    7.45    32.4
##  7 TLS   Timor-Leste  TL    Asia   South-~ <NA>     1960  474532    6.32    33.7
##  8 LBR   Liberia      LR    Africa Sub-Sa~ Wester~  1960 1118657    6.41    34.3
##  9 BFA   Burkina Faso BF    Africa Sub-Sa~ Wester~  1960 4829288    6.29    34.4
## 10 BTN   Bhutan       BT    Asia   Southe~ <NA>     1960  223288    6.64    34.5
## # ... with 206 more rows, 16 more variables: childMortality <dbl>,
## #   youthFemaleLiteracy <dbl>, youthMaleLiteracy <dbl>, adultLiteracy <dbl>,
## #   GDP_PC <dbl>, accessElectricity <dbl>, agriculturalLand <dbl>,
## #   agricultureTractors <dbl>, cerealProduction <dbl>, fertilizerHa <dbl>,
## #   co2 <dbl>, greenhouseGases <dbl>, co2_PC <dbl>, pm2.5_35 <dbl>,
## #   battleDeaths <dbl>, rank <dbl>, and abbreviated variable names
## #   1: `sub-region`, 2: `intermediate-region`, 3: totalPopulation, ...
```

8. (6pt) If you did this correctly, then you noticed that US ranking has been falling quite a bit. But we also have more countries in 2019–what about the relative rank divided by the corresponding number of countries that have LE data in the corresponding year? Hint: 0.0904 for 1960.