COVID assignment

2022-07-08

Project Files and Importing Data

First of all we need to start with finding an appropriate source of data and do our research to establish if the data is suitable for our research. It is of crucial importance to use reliable, credible and un-biased source for any kind of analysis we are about to perform.

As suggested during the lectures, I have used the same data from Johns Hopkins located on github and started with reading them in into the R Studio.

Throughout the assignment I will use two R libraries:

```
library(tidyverse)
library(lubridate)
```

```
## Read in the data from github

global_cases <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_1
global_deaths <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_
US_cases <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_da
US_deaths <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_d
```

Tidying and Transforming Data

When we have our data read in, we can proceed with cleaning and tidying them up. We will remove information that we do not need for the purpose of our analysis (such as latitude, longitude) and just like in video putting dates, deaths and cases in their own columns.

```
global_cases <- global_cases %>%
pivot_longer(cols = -c('Province/State', 'Country/Region', Lat, Long), names_to = "date", values_to = "
select(-c(Lat, Long))
global_deaths <- global_deaths %>%
pivot_longer(cols = -c('Province/State', 'Country/Region', Lat, Long), names_to = "date", values_to = "
select(-c(Lat, Long))
```

Afterwards we have a further look at our dataset and we can continue with additional tidying up. For ease of analysis, we can join global_cases data set with global_deaths dataset. As well we can notice that date column is a “character” vector, so we can as well mutate that from character to date object.

```
global <- global_cases %>%
full_join(global_deaths) %>%
rename(Country_Region = 'Country/Region', Province_State = 'Province/State') %>%
mutate(date = mdy(date))
```

```
## Joining, by = c("Province/State", "Country/Region", "date")
```

To help us with our analysis we can filter out all “zero” cases.

```
global <- global %>% filter(cases > 0)
```

Additionally we can see that our data set contains two separate columns:

- Province_State
- Country_Region

We can combine those into one for ease of analysis.

```
global <- global %>%
unite ("Combined_Key", c(Province_State, Country_Region), sep = "," , na.rm = TRUE, remove = FALSE)
```

To help with our analysis we can as well add to our data set information about total population in our countries. We will use data from Johns Hopkins Universit in github and join it with our existing global dataset.

```
uid_lookup_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/
uid <- read_csv(uid_lookup_url) %>%
select(-c(Lat, Long_, Combined_Key, code3, iso2, iso3, Admin2))
```

```
## Rows: 4317 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (7): iso2, iso3, FIPS, Admin2, Province_State, Country_Region, Combined_Key
## dbl (5): UID, code3, Lat, Long_, Population
##
## 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.
```

```
global <- global %>%
left_join(uid, by = c("Province_State", "Country_Region")) %>%
select(-c(UID,FIPS)) %>%
select(Province_State, Country_Region, date, cases, deaths, Population, Combined_Key)
```

After we have tidied up our global data, we can move on to our US data, where we will perform more or less the same tasks as we did with previous data sets.

```
US_cases <- US_cases %>%
pivot_longer(cols = -(UID:Combined_Key), names_to = "date" , values_to = "cases") %>%
select(Admin2:cases) %>%
mutate(date = mdy(date)) %>%
select(-c(Lat, Long_))
```

```
US_deaths <- US_deaths %>%
pivot_longer(cols = -(UID:Combined_Key), names_to = "date" , values_to = "deaths") %>%
select(Admin2:deaths) %>%
mutate(date = mdy(date)) %>%
select(-c(Lat, Long_))
```

```
## Warning: 3342 failed to parse.
```

Finally we can join our two data sets together:

```
US <- US_cases %>%
full_join(US_deaths)
```

```
## Joining, by = c("Admin2", "Province_State", "Country_Region", "Combined_Key",
## "date")
```

And as a matter of consistency we will filter out zero cases even in our US data set.

```
US <- US %>% filter (cases > 0)
```

We can look now at the summary for both of our data sets:

```
summary(global)
```

```
##  Province_State     Country_Region          date                cases
##  Length:244187      Length:244187      Min.   :2020-01-22   Min.   :       1
##  Class :character   Class :character   1st Qu.:2020-10-16   1st Qu.:     865
##  Mode  :character   Mode  :character   Median :2021-05-26   Median :   12793
##                                        Mean   :2021-05-23   Mean   :  729723
##                                        3rd Qu.:2021-12-31   3rd Qu.:  186211
##                                        Max.   :2022-08-04   Max.   :91961519
##
##      deaths          Population         Combined_Key
##  Min.   :      0   Min.   :8.090e+02   Length:244187
##  1st Qu.:      6   1st Qu.:7.892e+05   Class :character
##  Median :    153   Median :7.133e+06   Mode  :character
##  Mean   :  12111   Mean   :2.922e+07
##  3rd Qu.:   2781   3rd Qu.:2.914e+07
##  Max.   :1032820   Max.   :1.380e+09
##                    NA’s   :4993
```

```
summary(US)
```

```
##     Admin2          Province_State     Country_Region     Combined_Key
##  Length:2769446     Length:2769446     Length:2769446     Length:2769446
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##       date                 cases               deaths
##  Min.   :2020-01-22   Min.   :      1   Min.   :    0.0
##  1st Qu.:2020-11-02   1st Qu.:    458   1st Qu.:    6.0
##  Median :2021-06-04   Median :   2085   Median :   36.0
##  Mean   :2021-06-03   Mean   :  11697   Mean   :  172.9
##  3rd Qu.:2022-01-03   3rd Qu.:   6911   3rd Qu.:  111.0
##  Max.   :2022-08-04   Max.   :3320781   Max.   :32807.0
```

Lastly I have decided to work further with the global data set, therefore I will add extra column that shows daily individual increases in cases and deaths to our global dataset and new column that shows number of deaths per million of population.

```
world <- global %>% mutate(deaths_per_mill = deaths *1000000/ Population)
world <- world %>%
mutate(new_cases = cases - lag(cases), new_deaths = deaths - lag(deaths))
```

Analyzing and Modeling data

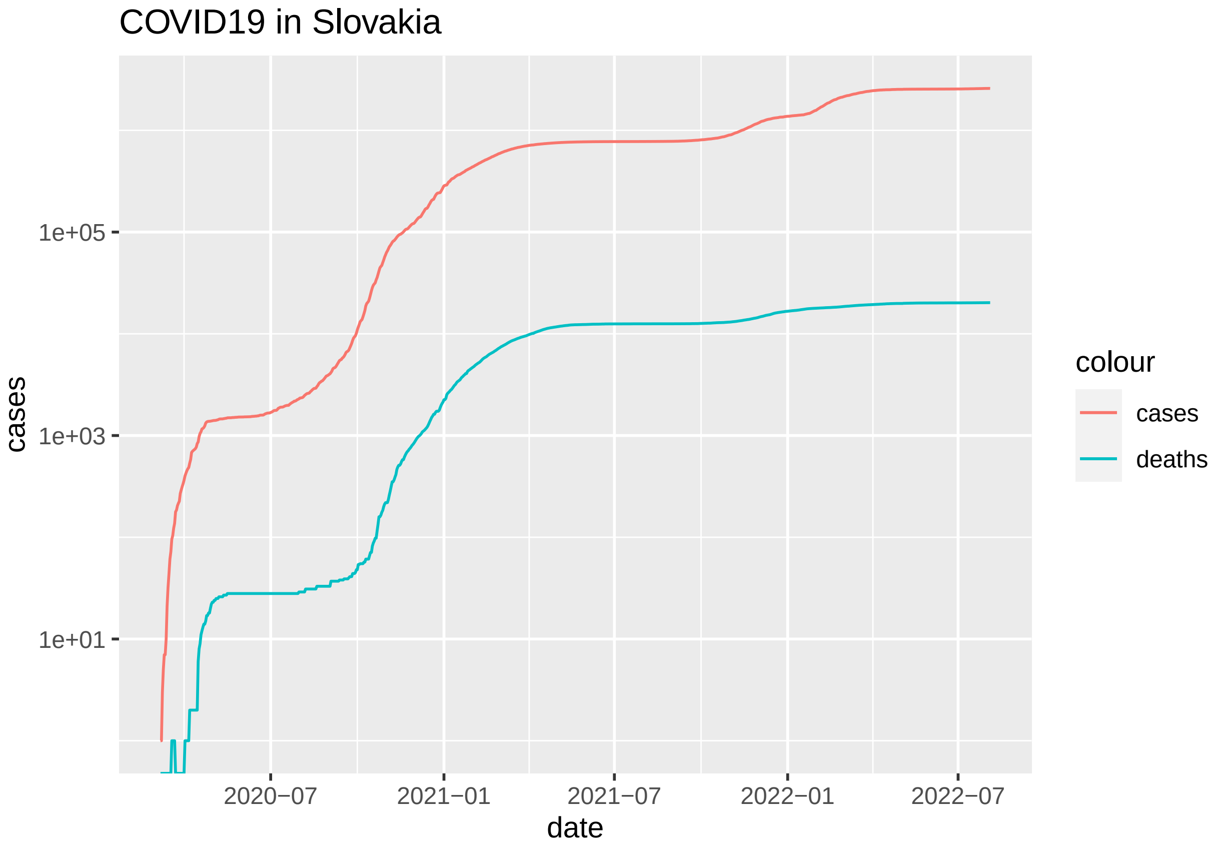
I have decided to have a further into the COVID situation in my home country Slovakia, therefore I will create a new data set called “Slovakia” from our world dataset.

```
Slovakia <- world %>% filter(Country_Region == "Slovakia")
```

Naturally, first of all I would like to look at the progression of COVID19 infections and related deaths in the country througout the pandemic starting early months of 2020 till the date.

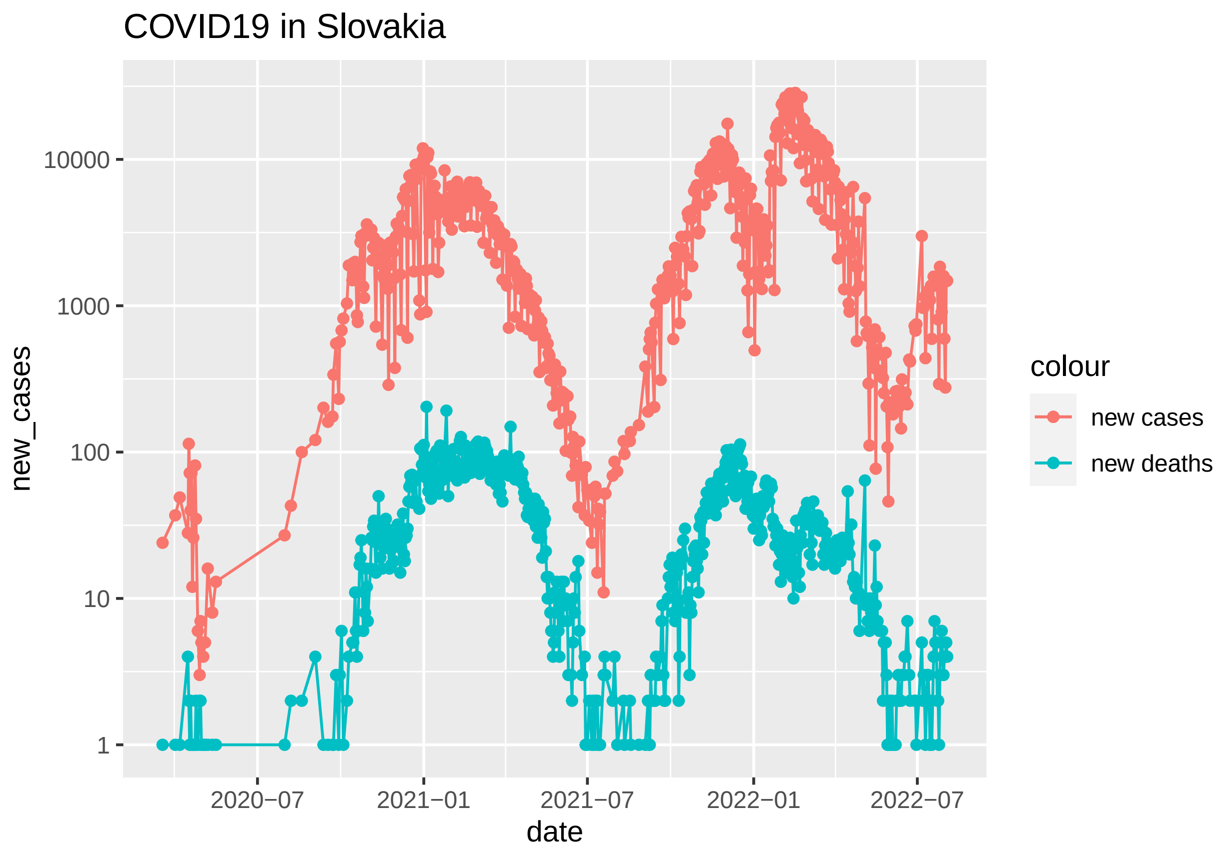
```
Slovakia %>%
ggplot(aes(x=date, y=cases)) +
geom_line(aes(color = "cases")) +
geom_line(aes(y=deaths, color = "deaths")) +
scale_y_log10() +
labs(title = "COVID19 in Slovakia")
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
```



Here we can clearly see rapid growth in infections during early stages of the pandemic, followed by another sharp rise in infections during the winter of 2020/2021. From this graph however, we cannot clearly see the waves and pattern of Covid pandemic in Slovakia, as this graph is using cumulative totals, therefore for better visualization we can have a look at the graph of new cases and new deaths, that displays in much better manner the dynamics of the pandemic in the country.

```
Slovakia %>%
filter(new_cases > 0, new_deaths > 0) %>%
ggplot(aes(x=date, y=new_cases)) +
geom_point(aes(color = "new cases")) +
geom_line(aes(color = "new cases")) +
geom_point(aes(y=new_deaths, color = "new deaths")) +
geom_line(aes(y=new_deaths, color = "new deaths")) +
scale_y_log10() +
labs(title = "COVID19 in Slovakia")
```

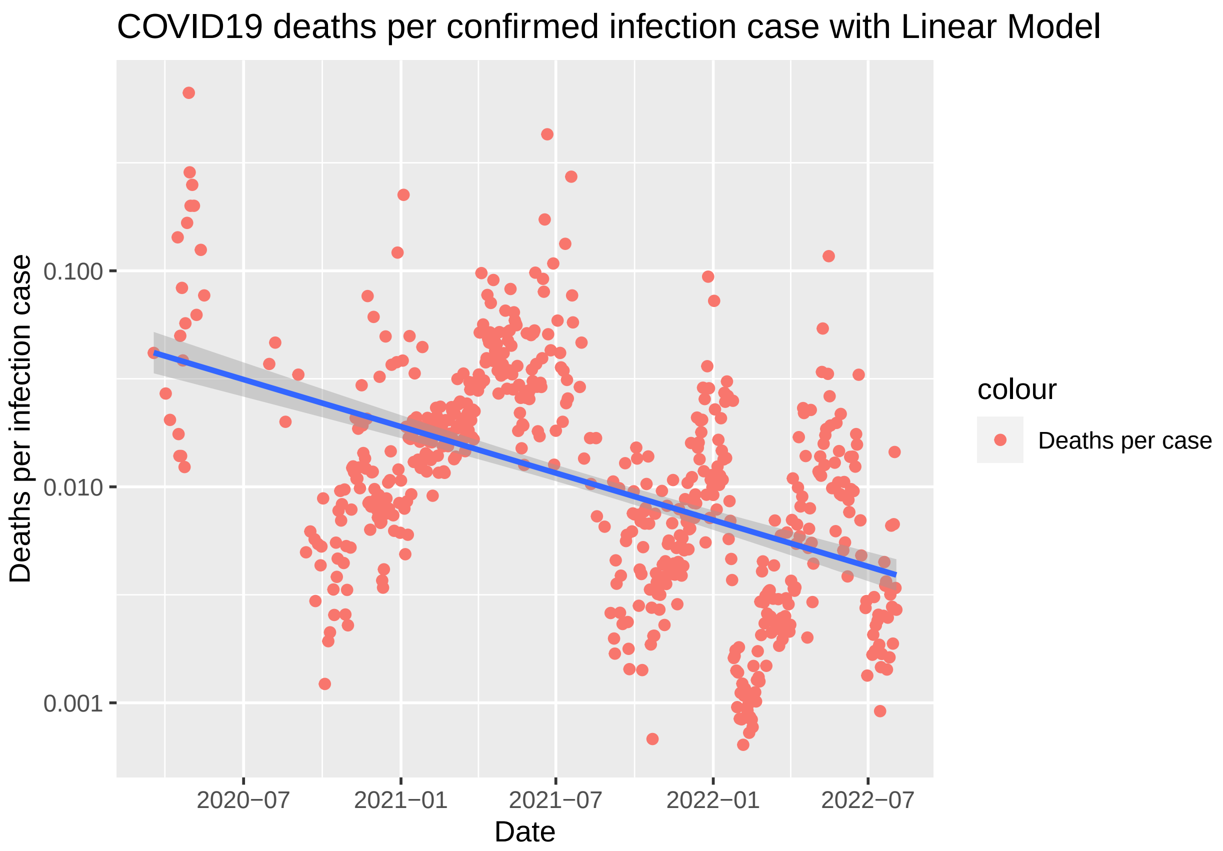


This second graph is much better visualization of the covid related situation in Slovakia, showing peaks through both winters 2020/21 and 2021/22, with decrease in number of infections (and deaths) during the summer periods. Naturally both graphs (new cases and new deaths) show similar waves and pattern, as naturally with higher number of cases there is higher number of related deaths.

Since the world and scientfic community learns more and more about COVID19, I was interested to see how my country is doing in terms of covid related deaths per one new case of infections. My initial idea was to see, since scientists and doctors learn more about this infection and ways how to treat it, if number of deaths per confirmed case of COVID19 infection is decreasing.

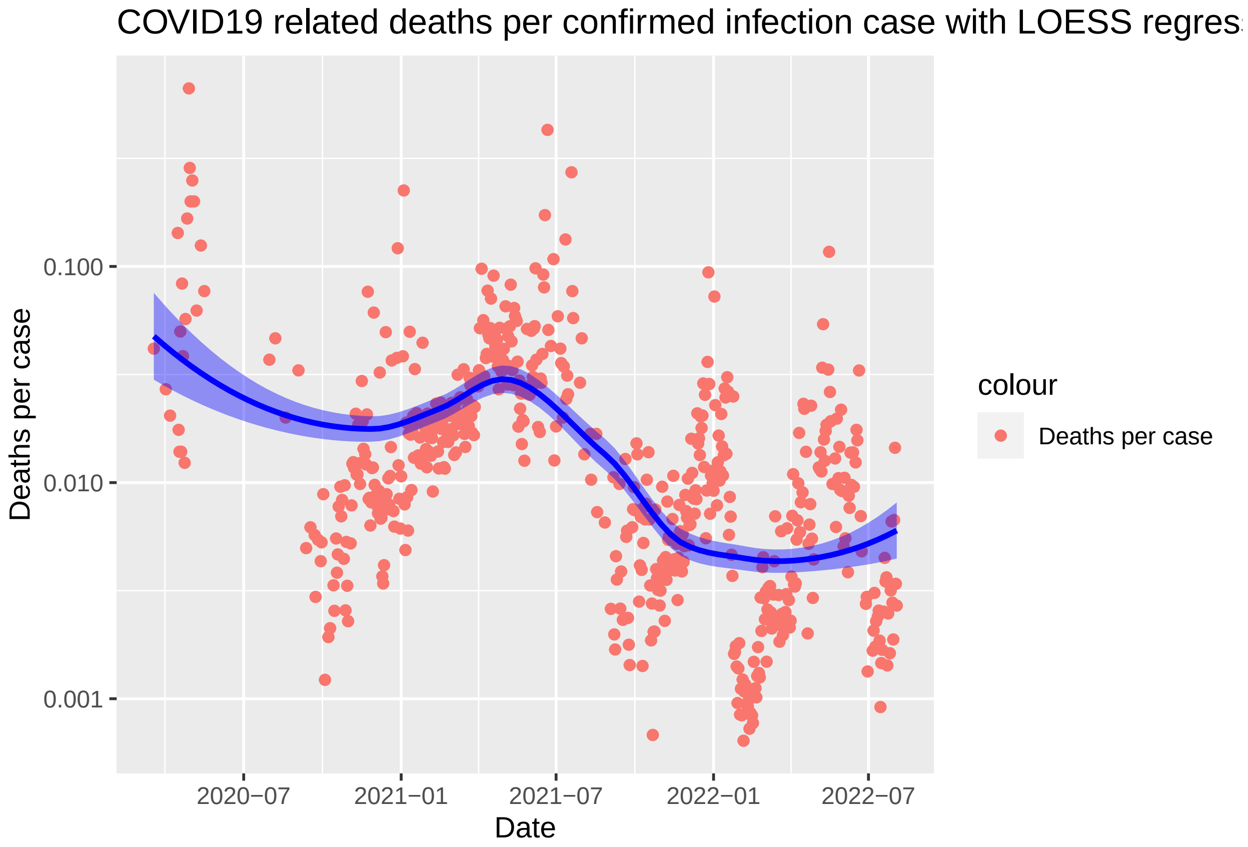
I decided to use two different methods: Linear model and LOESS regression.

```
Slovakia %>%
filter(new_deaths > 0, new_cases > 0) %>%
ggplot(aes(x=date, y= new_deaths/new_cases)) +
geom_point(aes(color="Deaths per case")) +
geom_smooth(method=lm) +
scale_y_log10() +
labs(x= "Date", y = "Deaths per infection case", title = "COVID19 deaths per confirmed infection case wi
```



On this graph we can see that our model says that number of deaths per covid cases is consistently decreasing since the start of the pandemic. Since this is very simple model, I wanted to see what would be the result using more advanced model, in this case LOESS regression.

```
Slovakia %>%
filter(new_deaths > 0, new_cases > 0) %>%
ggplot(aes(x=date, y= new_deaths/new_cases)) +
geom_point(aes(color = "Deaths per case")) +
geom_smooth(color = "blue", fill = "blue") +
scale_y_log10() +
labs(x="Date", y="Deaths per case", title = "COVID19 related deaths per confirmed infection case with LO
```



In this graph we achieved clearer visualization of our data. Since the start of the pandemic we can see that number of deaths per each confirmed case of COVID19 infections is in general decreasing (as suggested by the previous linear model), however for the period starting shortly prior to downloading this dataset, we can see slight upward trajectory of the blue model line. We cannot speculate about possible reasons why it is so. For further clarification we would need to do further research and/or consult specialists in related fields.

Bias Identification

My personal bias for this (still) highly discussed topic would be the fact that I have no medical background, nor do I have knowledge from the fields of microbiology or virology. All the information I have about COVID19 comes from media or from the broadcasts of local health authorities in the place live. In order to be able to fully and correctly analyse this data, one needs better knowledge of above mentioned fields or close cooperation with professionals working in those areas. Therefore it is important that I stick to the data presented here and I do not transfer my personal opinion or limited information into the analysis

```
sessionInfo()
```

```
## R version 4.2.0 (2022-04-22)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur/Monterey 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.2/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.2/Resources/lib/libRlapack.dylib
##
```

```
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] lubridate_1.8.0 forcats_0.5.1   stringr_1.4.0   dplyr_1.0.9
##  [5] purrr_0.3.4     readr_2.1.2     tidyr_1.2.0     tibble_3.1.7
##  [9] ggplot2_3.3.6   tidyverse_1.3.1
##
## loaded via a namespace (and not attached):
##  [1] lattice_0.20-45  assertthat_0.2.1 digest_0.6.29    utf8_1.2.2
##  [5] R6_2.5.1         cellranger_1.1.0 backports_1.4.1  reprex_2.0.1
##  [9] evaluate_0.15    httr_1.4.3       highr_0.9        pillar_1.7.0
## [13] rlang_1.0.2      curl_4.3.2       readxl_1.4.0     rstudioapi_0.13
## [17] Matrix_1.4-1     rmarkdown_2.14   splines_4.2.0    bit_4.0.4
## [21] munsell_0.5.0    broom_0.8.0      compiler_4.2.0   modelr_0.1.8
## [25] xfun_0.31        pkgconfig_2.0.3  mgcv_1.8-40      htmltools_0.5.2
## [29] tidyselect_1.1.2 fansi_1.0.3      crayon_1.5.1     tzdb_0.3.0
## [33] dbplyr_2.2.0     withr_2.5.0      grid_4.2.0       nlme_3.1-157
## [37] jsonlite_1.8.0   gtable_0.3.0     lifecycle_1.0.1  DBI_1.1.2
## [41] magrittr_2.0.3   scales_1.2.0     cli_3.3.0        stringi_1.7.6
## [45] vroom_1.5.7      farver_2.1.0     fs_1.5.2         xml2_1.3.3
## [49] ellipsis_0.3.2   generics_0.1.2   vctrs_0.4.1      tools_4.2.0
## [53] bit64_4.0.5      glue_1.6.2       hms_1.1.1        parallel_4.2.0
## [57] fastmap_1.1.0    yaml_2.3.5       colorspace_2.0-3 rvest_1.0.2
## [61] knitr_1.39       haven_2.5.0
```