Did COVID-19 cases in Europe lead cases in the USA?

An example of TidyR and R Markdown

Paul Beaumont Department of Economics Florida State University Tallahassee, FL [beaumont@fsu.edu]

October 07, 2021

Abstract

This is an example of the type of reports that we would like to be able to produce in this class. Details to follow! For now, we will focus on the R Studio environment, the yaml, the notion of tidy data, and the coding conventions that are used here.

```
# Required libraries
library(tidyverse)
```

First, what is the difference between *data science* and *data analysis*?

Data analysts and data scientists both work with data, but the main difference lies in what they do with the data. Data analysts examine large data sets to identify trends, develop charts, and create visual presentations to help businesses make more strategic decisions. Data scientists design and construct new processes for data modeling and production using prototypes, algorithms, predictive models, and custom analysis.

As an example, we will examine the raw COVID-19 cases data and try to determine whether cases in Europe led the USA during the initial outbreak period of February and March of 2020.

The Covid data is from: Our World in Data.

```
owid_covid_data <- read_csv("owid-covid-data.csv")
```

This data comes to us in a "tidy" format. By this we mean that the data are well-organized from a *computing* perspective so that there are a unique set of identifiers for every data value in the file. Humans prefer to see "flat" files like spreadsheets but this is very often inefficient from a computing perspective because we have to continually reformat and reorganize the data for different types of analyses.

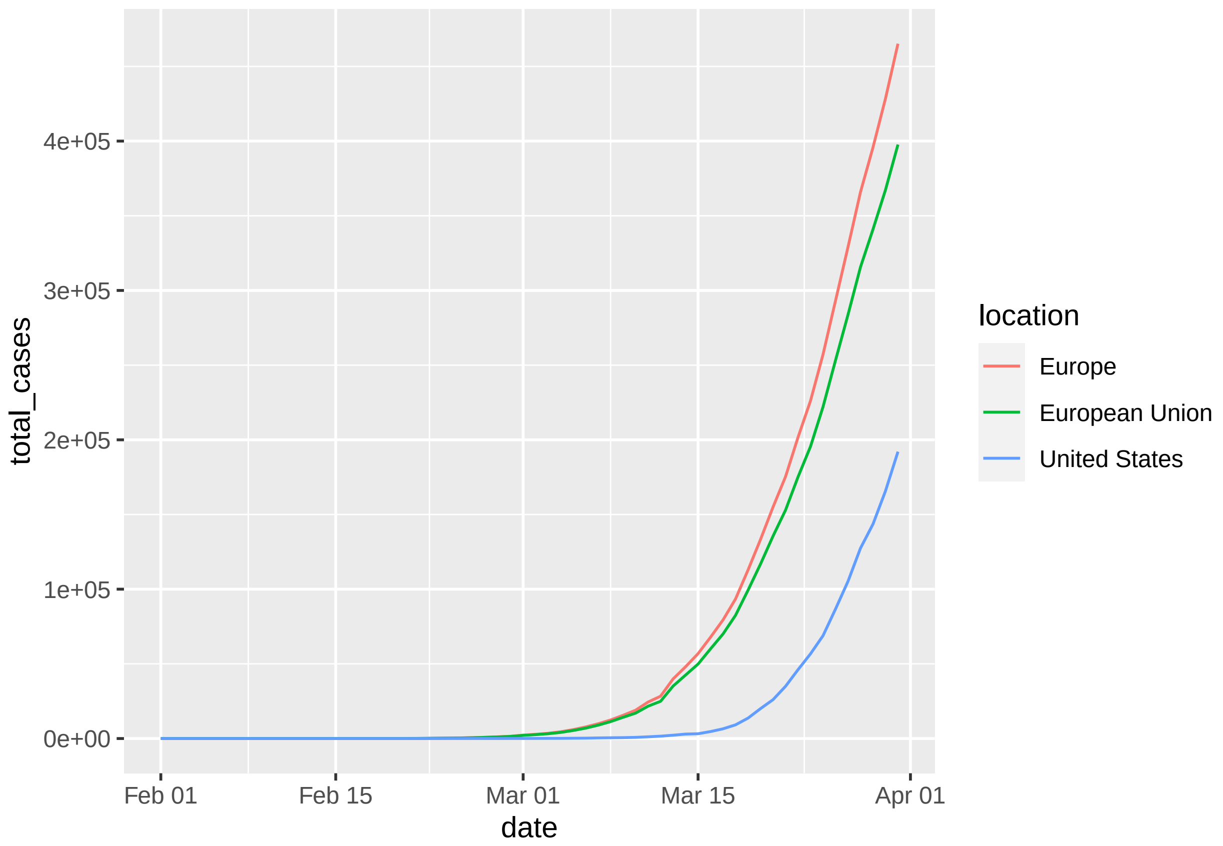
Putting the data into a tidy format can be quite tedious and needs to be carefully thought out from the beginning. This is the job of a data scientist. Their tool of choice is usually SQL, which you may have had some exposure to. We will spend quite a bit of time learning about the *tidyverse* which is a simplified, but more user friendly, version of SQL developed within R.

As data analysts, we hope that our data is already tidy, or can easily be made tidy, so we really want to focus on how to manipulate tidy data to do useful analyses. For our example, we will select the total cases and the "seven day moving average of new cases" for Europe, European Union, and the United States from the beginning of the data set (2020-02-01) through 2020-03-31.

```
USEU <- owid_covid_data %>%
  filter(
    location %in% c("Europe", "European Union", "United States") &
      date >= "2020-02-01" & date <= "2020-03-31"
  ) %>%
  select(location, date, total_cases, new_cases_smoothed)
```

First we plot the total cases data. It "looks" like Europe and the EU lead the USA.

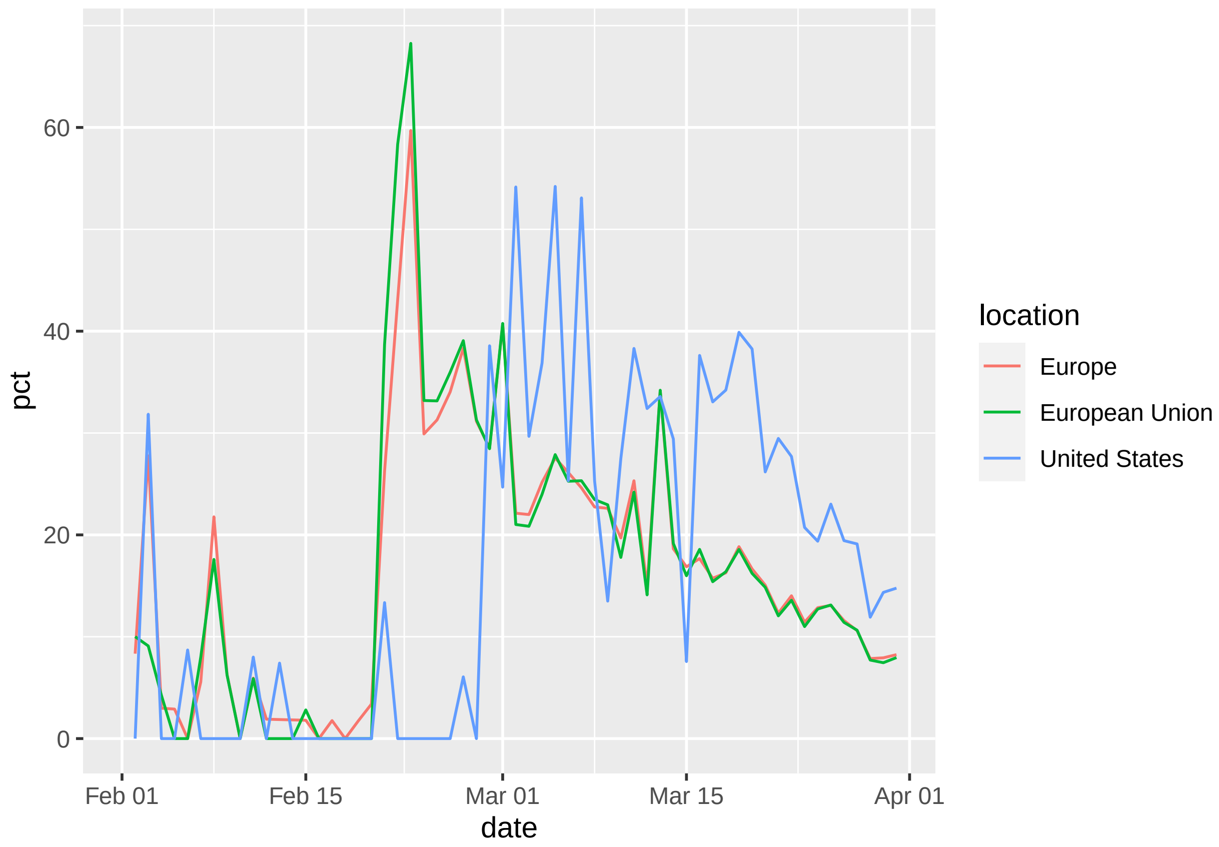
```
USEU %>%
  ggplot(mapping = aes(
    x = date,
    y = total_cases,
    group = location,
    color = location
  )) + geom_line()
```



One statistical issue is that the total_cases data are clearly *non-stationary*. This is a common problem that we will have a lot to say about in the class. For now, we will deal with this issue by examining the percentage change in total_cases.

```
USEU_pct <- USEU %>%
  group_by(location) %>%
  mutate(pct = 100*(log(total_cases/lag(total_cases))))
```

```
USEU_pct %>%
  ggplot(mapping = aes(
    x = date,
    y = pct,
    group = location,
    color = location
  )) + geom_line()
```

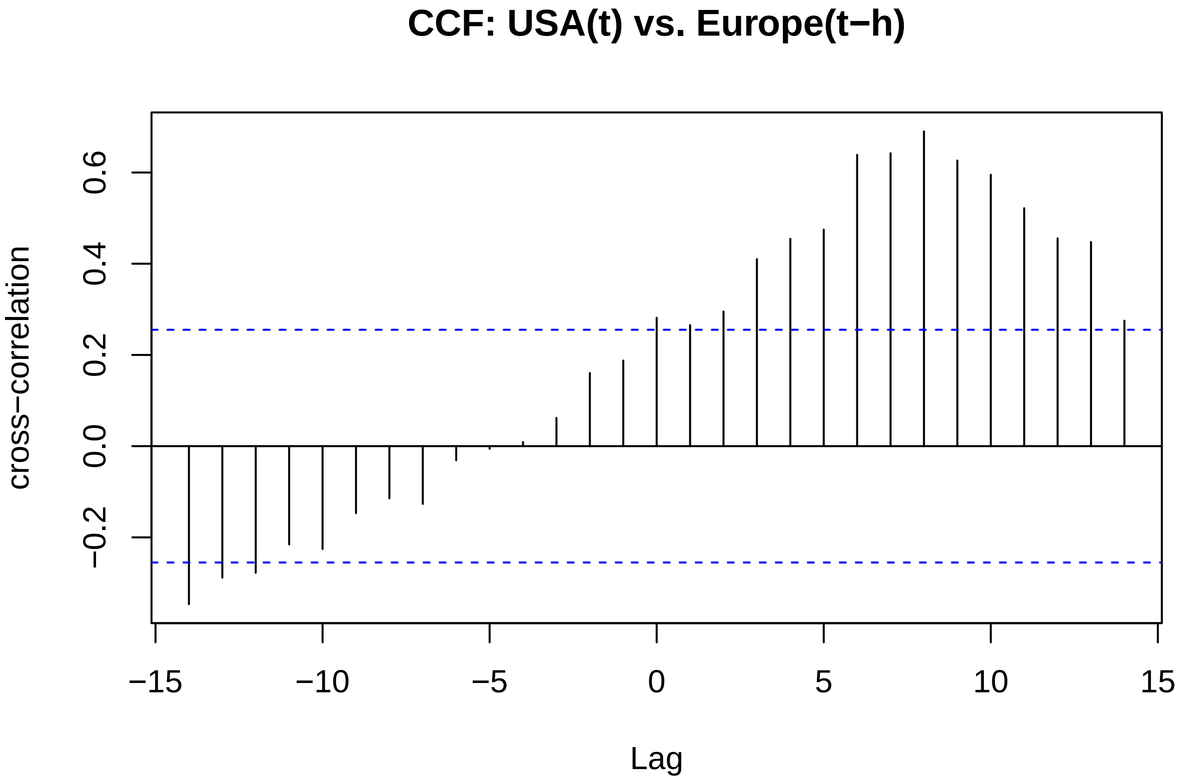


From a statisticians viewpoint, that looks much nicer.

Now we will compute the cross-correlation function for the growth rate of total cases in the USA and in Europe.

```
usa <- USEU_pct %>%
  filter(location == "United States") %>%
  ungroup %>%
  select(date, pct) %>%
  drop_na()
europe <- USEU_pct %>%
  filter(location == "Europe") %>%
  ungroup %>%
  select(date, pct) %>%
  drop_na()
```

```
ccf.out <-
  ccf(usa$pct,
      europe$pct,
      lag.max = 14,
      plot = FALSE)
# find the h for the maximum cross-correlation
h_max <- which(ccf.out$acf == max(ccf.out$acf))
max_ccf_lag <- ccf.out$lag[h_max]
max_ccf_value <- ccf.out$acf[h_max]
# plot the ccf
plot(ccf.out, ylab = "cross-correlation", main = "CCF: USA(t) vs. Europe(t-h)")
```



The CCF plot suggests that the growth rate of the number of cases in Europe leads the USA with a maximum cross-correlation of 0.69 at 8 days.