Analytical Notebook

Stefan Zimmermann

11 7 2020

Install and load R-Packages

```r
# install and load packages
loadpackage <- function(x){
  for( i in x ){
    #  require returns TRUE invisibly if it was able to load package
    if( ! require( i , character.only = TRUE ) ){
      #  If package was not able to be loaded then re-install
      install.packages( i , dependencies = TRUE )
    }
    #  Load package (after installing)
    library( i , character.only = TRUE )
  }
}

# load packages
loadpackage( c("readr", "knitr", "dplyr", "tidyr", "sparklyr", "ggplot2"))
```

Load Covid-Datasets

The datasets UID_ISO_FIPS_LookUp_Table.csv and time_series_covid19_confirmed_global.csv are loaded.

```r
# use url to current dataset
url_data1 <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/UID_IS
url_data2 <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_c

# load datasets
data1 <- read_csv(url_data1)
data2 <- read_csv(url_data2)

# harmonise ID-Variables in both datasets
names(data1)[names(data1) == "Long_"] <- "Long"
names(data2)[names(data2) == "Country/Region"] <- "Country_Region"
names(data2)[names(data2) == "Province/State"] <- "Province_State"

# Since we are only interested in countries and not in regions
# we keep only a subset of the dataset without regions.
data1 <-subset(data1, is.na(data1$Province_State))
data2 <-subset(data2, is.na(data2$Province_State))

data2 <-
  reshape(
    data = as.data.frame(data2),
    varying = list(names(data2)[5:length(data2)]),
    timevar = "day",
    v.names = "count",
    idvar = c("Country_Region"),
    direction = "long",
    times = names(data2)[5:length(data2)]
  )

data2$date <- as.Date(data2$day, format = "%m/%d/%y")
data2$datecount <- data2$date-min(data2$date)
days <- unique(data2$day)
weeks <- days[seq(1, length(days), 7)]

write_csv(data1, path = '../input/data1.csv')
write_csv(data2, path = '../input/data2.csv')

# setting up spark
sc <- spark_connect(master = "local",
                    version = "2.4.3")

data1 <- copy_to(sc, data1, overwrite = T)
data2 <- copy_to(sc, data2, overwrite = T)
src_tbls(sc)
```

Data Cleaning

Here the data sets are prepared for merging and reshaping. The ID variables must be standardized. Since we are only interested in countries and not in regions # we keep only a subset of the dataset without regions. Then we drop countries without information and we keep all important variables. We reshape the dataset to long format

```r
data_merge <- merge(data2, data1, by = "Country_Region")

data_merge <- data_merge %>%
  select(Country_Region, Country_Region, day, date, count, datecount, Population)

write_csv(data_merge, path = '../input/data_merge.csv')
data_merge <- copy_to(sc, data_merge, overwrite = T)

# Germany, China,Japan, United Kingdom, US, Brazil, Mexico
my_data <- data_merge %>%
  filter(Country_Region=="Germany" | Country_Region=="Mexico" |
         Country_Region=="United Kingdom" | Country_Region=="US" |
         Country_Region=="Brazil" | Country_Region=="China" |
         Country_Region=="Japan") %>%
  mutate(rate = (count/Population)*100) %>%
  mutate(count2 = round((count/1000))) %>%
  collect() %>%
  print()
```

```
## # A tibble: 1,032 x 8
##    Country_Region day   date        count datecount Population   rate
##    <chr>          <chr> <date>      <dbl>     <dbl>      <dbl>  <dbl>
##  1 Brazil         6/21~ 2020-06-21 1.08e6       151  212559409 0.510
##  2 Brazil         2/4/~ 2020-02-04 0.            13  212559409 0
##  3 Brazil         2/8/~ 2020-02-08 0.            17  212559409 0
##  4 Brazil         1/22~ 2020-01-22 0.             0  212559409 0
##  5 Brazil         6/19~ 2020-06-19 1.03e6       149  212559409 0.486
##  6 Brazil         2/6/~ 2020-02-06 0.            15  212559409 0
##  7 Brazil         5/9/~ 2020-05-09 1.56e5       108  212559409 0.0734
##  8 Brazil         5/13~ 2020-05-13 1.90e5       112  212559409 0.0895
##  9 Brazil         5/4/~ 2020-05-04 1.09e5       103  212559409 0.0511
## 10 Brazil         5/29~ 2020-05-29 4.65e5       128  212559409 0.219
## # ... with 1,022 more rows, and 1 more variable: count2 <dbl>
```

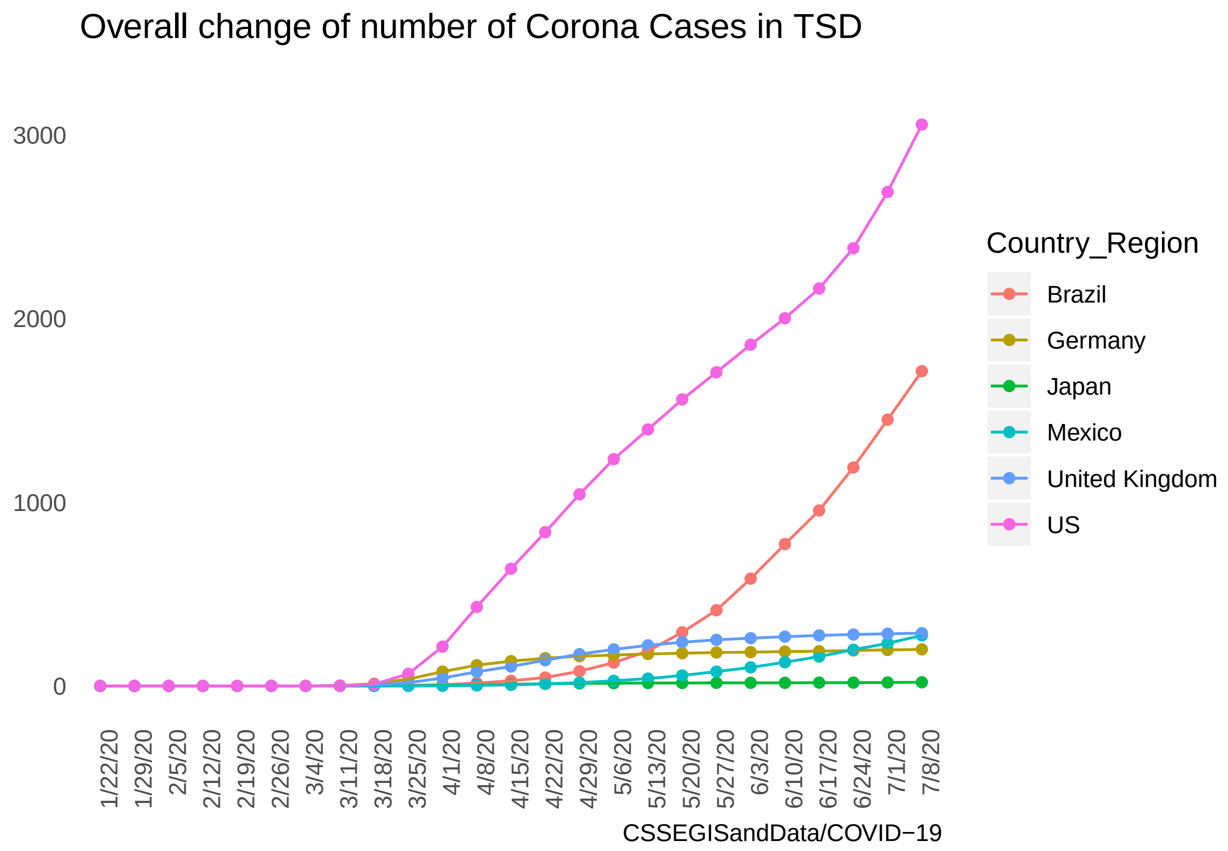
Plots

You can also embed plots, for example:

```
ggplot(data=my_data, aes(x=reorder(day, count2), y=count2, group=Country_Region, colour=Country_Region)
  geom_point()+
  geom_line()+
  scale_x_discrete(limit = weeks)+
  theme(axis.text.x = element_text(angle = 90, hjust = 1),
        axis.title=element_blank(),
        axis.ticks = element_blank(),
        strip.text = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank())+
 labs(title =  "Overall change of number of Corona Cases in TSD",
       caption = "CSSEGISandData/COVID-19")
```

```
## Warning: Removed 882 rows containing missing values (geom_point).
```

```
## Warning: Removed 882 rows containing missing values (geom_path).
```



Overall change of number of Corona Cases in TSD

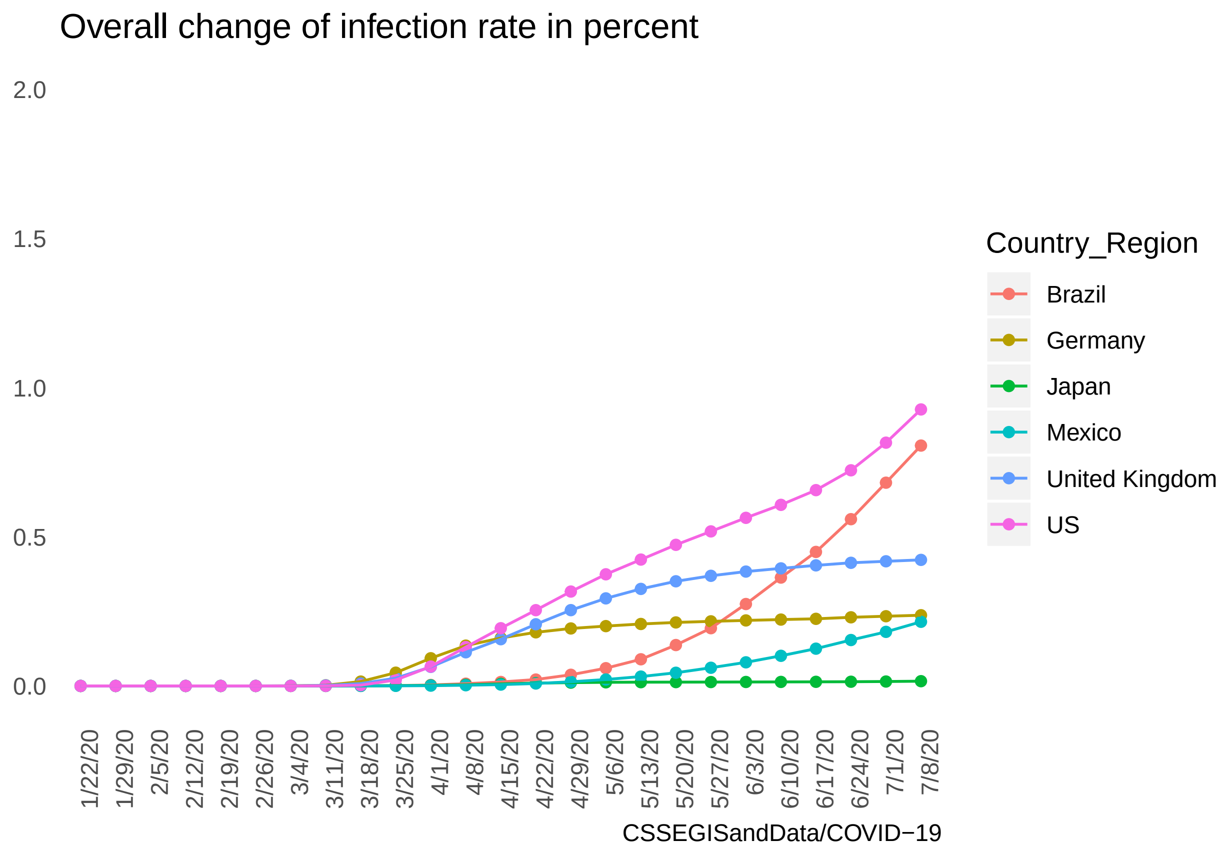
CSSEGISandData/COVID−19

```
ggsave("../output/total_change.png")
```

```
## Saving 6.5 x 4.5 in image

## Warning: Removed 882 rows containing missing values (geom_point).

## Warning: Removed 882 rows containing missing values (geom_path).
```

```
ggplot(data=my_data, aes(x=reorder(day, rate), y=rate, group=Country_Region, colour=Country_Region)) +
  geom_point()+
  geom_line()+
  scale_x_discrete(limit = weeks)+
  ylim(0,2)+
  theme(axis.text.x = element_text(angle = 90, hjust = 1),
        axis.title=element_blank(),
        axis.ticks = element_blank(),
        strip.text = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank())+
 labs(title =  "Overall change of infection rate in percent",
       caption = "CSSEGISandData/COVID-19")
```

```
## Warning: Removed 882 rows containing missing values (geom_point).

## Warning: Removed 882 rows containing missing values (geom_path).
```



```
ggsave("../output/rate_change.png")
```

```
## Saving 6.5 x 4.5 in image
## Warning: Removed 882 rows containing missing values (geom_point).

## Warning: Removed 882 rows containing missing values (geom_path).
```

Regression