Analysis

Huy Le Quang

7/15/2020

1.Introduction

2. Dataset

3. Descriptive analysis

xxx

```
# Load necessary packages

library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
## v ggplot2 3.3.1     v purrr   0.3.4
## v tibble  3.0.1     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts ------------------------------------------------------------------------------------ tidy
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(sparklyr)
```

```
##
## Attaching package: 'sparklyr'
## The following object is masked from 'package:purrr':
##
##     invoke
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

xxx

```
# Download data

lookup.table.url <- "https://github.com/CSSEGISandData/COVID-19/blob/master/csse_covid_19_data/UID_ISO_

lookup.table <- read.csv(url(lookup.table.url))
```

```
time.series.url <- "https://github.com/CSSEGISandData/COVID-19/blob/master/csse_covid_19_data/csse_covi
time.series <- read.csv(url(time.series.url))
```

xxx

```
# Clean the time series dataset

time.series <- time.series[-c(3, 4)] # delete unnecessary columns

names(time.series)[names(time.series) == 'Country.Region'] <- 'country'
names(time.series)[names(time.series) == 'Province.State'] <- 'state'

time.series.long <- pivot_longer(data = time.series,
                         -c(state, country),
                         names_to = "date") # reshape to long format

# Change to date format
time.series.long$date <- as.Date(gsub("X", "", time.series.long$date), "%m.%d.%y")

begin.date <- as.Date("2020-01-22")

# Calculate the number of days since the start of data collection

time.series.long <- time.series.long %>%
    mutate(day_difference = date - begin.date)

# Change variable names in lookup dataset

names(lookup.table)[names(lookup.table) == 'Country_Region'] <- 'country'
names(lookup.table)[names(lookup.table) == 'Province_State'] <- 'state'
```

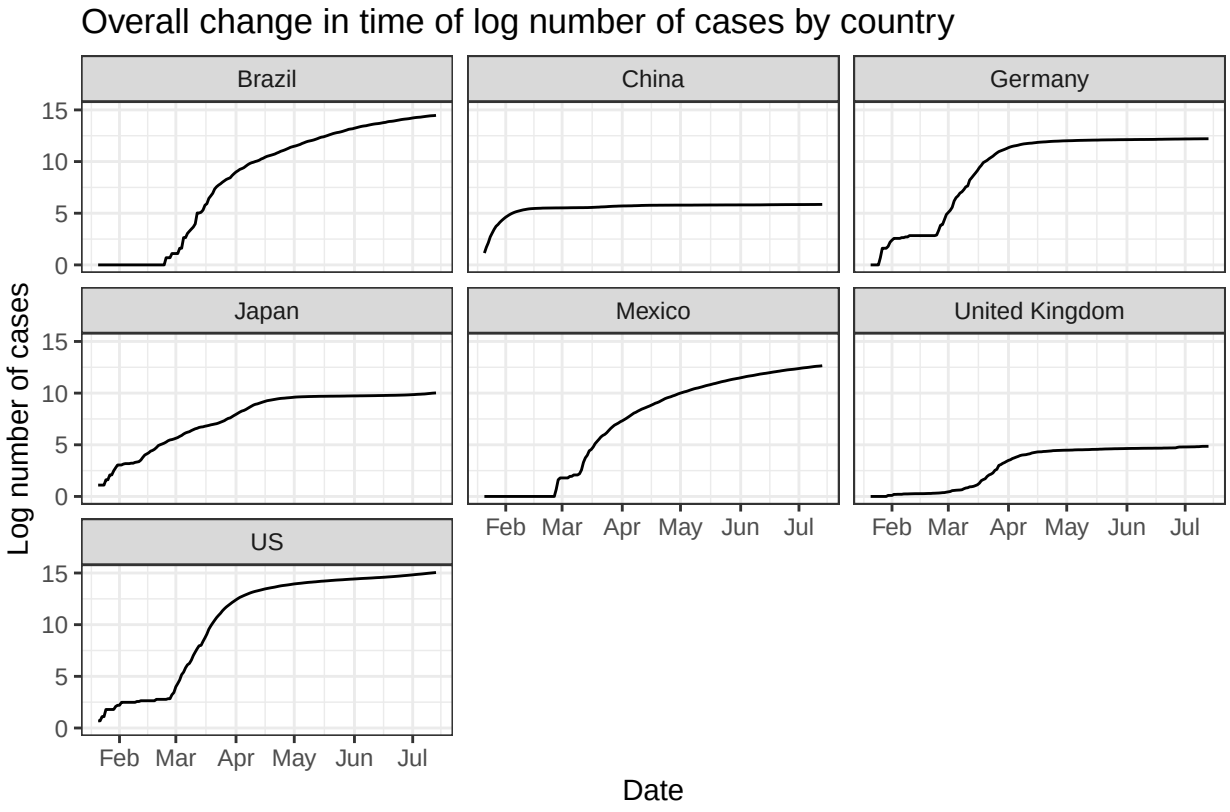
Establish Spark server and copy two datasets to this server

```
# Connect to local Spark server

sc <- spark_connect(master = "local",
                    version = "2.3")

# Copy two datasets to Spark server

time.series.long <- copy_to(sc, time.series.long, overwrite = T)
lookup.table <- copy_to(sc, lookup.table, overwrite = T)

# Merge two datasets

data_long <- full_join(time.series.long, lookup.table, by = c("state","country"))

# Rename variables

data_long <- data_long %>%
    rename(latitude = Lat,
           longitude = Long_,
```

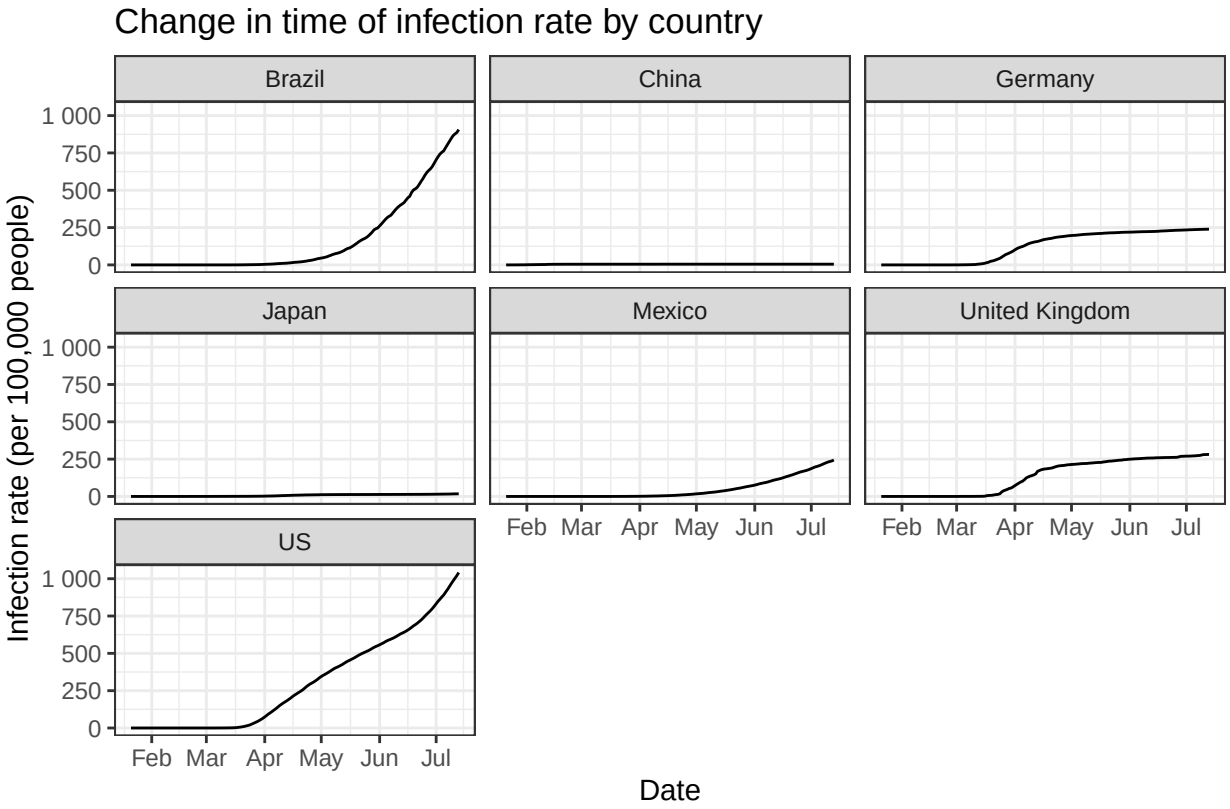
```
            population = Population,
            country_code = iso3,
            confirmed_cases = value)

# Make a smaller dataset and generate two new variables

data_final <- data_long %>%
    filter(country == "Germany"|country == "China"|country == "Japan"|
             country == "United Kingdom"|country == "US"|
             country =="Brazil"|country == "Mexico")%>%
    mutate(log.confirmed_cases = log((confirmed_cases+1)),
           infection_rate = confirmed_cases/population*100000)
```

Draw graphs

```
## Overall change in time of log number of cases

ggplot(aes(x = date, y = log.confirmed_cases),
       data = data_final) +
    stat_summary(fun.y = mean,
                 geom = "line") +
    theme_bw() +
    facet_wrap(~country) +
    labs(x = "Date", y = "Log number of cases",
         title = "Overall change in time of log number of cases by country",
         caption = "Data: JHU CSSE COVID-19 Dataset")
```




```
# Infection rate

ggplot(aes(x = date, y = infection_rate),
       data = data_final) +
    stat_summary(fun.y = mean,
                 geom = "line") +
    facet_wrap(~country) +
    scale_y_continuous(labels = scales::number_format(accuracy = 1)) +
    theme_bw() +
    labs(x = "Date", y = "Infection rate (per 100,000 people)",
         title = "Change in time of infection rate by country",
         caption = "Data: JHU CSSE COVID-19 Dataset. Infection rate per 100,000 people")
```



Regression model

```
data_final %>% lm(formula = log.confirmed_cases ~ country + population + day_difference) %>%
summary()
```

```
##
## Call:
## lm(formula = log.confirmed_cases ~ country + population + day_difference,
##     data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.9384 -1.2633 -0.0869  1.2320  6.1247
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -2.860e+00  2.474e-01  -11.56   <2e-16 ***
## countryChina            4.577e+00  2.162e-01   21.17   <2e-16 ***
## countryGermany          6.549e+00  2.455e-01   26.68   <2e-16 ***
## countryJapan            3.088e+00  2.300e-01   13.43   <2e-16 ***
## countryMexico           2.540e+00  2.296e-01   11.06   <2e-16 ***
## countryUnited Kingdom   3.714e+00  2.445e-01   15.19   <2e-16 ***
## countryUS              -3.040e+00  2.407e-01  -12.63   <2e-16 ***
## population              4.404e-08  8.958e-10   49.16   <2e-16 ***
## day_difference          2.155e-02  4.424e-04   48.72   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.027 on 8216 degrees of freedom
##   (3817 observations deleted due to missingness)
## Multiple R-squared:  0.5482, Adjusted R-squared:  0.5478
## F-statistic:  1246 on 8 and 8216 DF,  p-value: < 2.2e-16
```