Part A, Topic 1 - COVID-19

MONASH University

Lance Belen

May 16, 2025

Contents

1 Data Wrangling and Integration

First, load the packages needed.

```
library(readxl)
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.4     v tibble    3.2.1
## v purrr     1.0.4     v tidyr     1.3.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

Then, read and open the three datasets, Covid-data.csv, CountryLockdowndates.csv and WorldwideVaccineData.csv

```
covid <- read.csv("Covid-data.csv")
lockdown <- read.csv("CountryLockdowndates.csv")
vaccine <- read.csv("WorldwideVaccineData.csv")
```

1.1 Exploring the datasets

Covid-data.csv

```
cat("Number of rows:", nrow(covid), "\n")
```

```
## Number of rows: 1575
```

```
cat("Number of columns:", ncol(covid), "\n")
```

```
## Number of columns: 8
```

```
cat("Number of missing values:", sum(is.na(covid)), "\n")
```

```
## Number of missing values: 13
```

```
knitr::kable(head(covid), caption = "First 6 rows of the dataset")
```

Table 1: First 6 rows of the dataset

location	date	total_cases	new_cases	total_deaths	new_deaths	gdp_per_capita	population
Australia	2019-12-31	0	0	0	0	44648.71	25499881
Australia	2020-01-01	0	0	0	0	44648.71	25499881
Australia	2020-01-02	0	0	0	0	44648.71	25499881
Australia	2020-01-03	0	0	0	0	44648.71	25499881
Australia	2020-01-04	0	0	0	0	44648.71	25499881
Australia	2020-01-05	0	0	0	0	44648.71	25499881

```
cat("Structure of the dataset:\n")
```

```
## Structure of the dataset:
```

```
str(covid)
```

```
## 'data.frame':    1575 obs. of  8 variables:
##  $ location      : chr  "Australia" "Australia" "Australia" "Australia" ...
##  $ date          : chr  "2019-12-31" "2020-01-01" "2020-01-02" "2020-01-03" ...
##  $ total_cases   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ new_cases     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ total_deaths  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ new_deaths    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ gdp_per_capita: num  44649 44649 44649 44649 44649 ...
##  $ population    : int  25499881 25499881 25499881 25499881 25499881 25499881 25499881 25499881 2549
```

```
cat("Summary of the dataset:\n")
```

```
## Summary of the dataset:
```

```
summary(covid)
```

```
##    location             date            total_cases          new_cases
##  Length:1575        Length:1575        Min.   :      0.0   Min.   :-29726
##  Class :character   Class :character   1st Qu.:     21.5   1st Qu.:     1
##  Mode  :character   Mode  :character   Median :  58226.0   Median :   205
##                                        Mean   : 180451.9   Mean   :  2971
##                                        3rd Qu.: 173133.0   3rd Qu.:  1880
##                                        Max.   :3363056.0   Max.   : 66625
##
##   total_deaths      new_deaths       gdp_per_capita    population
##  Min.   :     0   Min.   :-1918.0   Min.   :15309   Min.   :2.550e+07
##  1st Qu.:     0   1st Qu.:    0.0   1st Qu.:26677   1st Qu.:6.046e+07
##  Median :  2837   Median :    5.0   Median :38606   Median :6.789e+07
##  Mean   : 14060   Mean   :  183.8   Mean   :35140   Mean   :2.652e+08
##  3rd Qu.: 25100   3rd Qu.:  149.0   3rd Qu.:42201   3rd Qu.:2.075e+08
##  Max.   :135605   Max.   : 4928.0   Max.   :54225   Max.   :1.439e+09
##  NA's   :6        NA's   :7
```

CountryLockdowndates.csv

```
cat("Number of rows:", nrow(lockdown), "\n")
```

```
## Number of rows: 307
```

```
cat("Number of columns:", ncol(lockdown), "\n")
```

```
## Number of columns: 5
```

```
cat("Number of missing values:", sum(is.na(lockdown)), "\n")
```

```
## Number of missing values: 0
```

```
knitr::kable(head(lockdown), caption = "First 6 rows of the dataset")
```

Table 2: First 6 rows of the dataset

Country.Region	Province	Date	Type	Reference
Afghanistan		24/03/2020	Full	https://www.thestatesman.com/world/afghan-govt-imposes-lockdown-coronavirus-cases-increase-15-1502870945.html
Albania		08/03/2020	Full	https://en.wikipedia.org/wiki/2020_coronavirus_pandemic_in_Albania
Algeria		24/03/2020	Full	https://www.garda.com/crisis24/news-alerts/325896/algeria-government-implements-lockdown-and-curfew-in-blida-and-algiers-march-23-update-7
Andorra		16/03/2020	Full	https://en.wikipedia.org/wiki/2020_coronavirus_pandemic_in_Andorra
Angola		24/03/2020	Full	https://en.wikipedia.org/wiki/2020_coronavirus_pandemic_in_Angola
Antigua and Barbuda			None	

```
cat("Structure of the dataset:\n")
```

```
## Structure of the dataset:
```

```
str(lockdown)
```

```
## 'data.frame':    307 obs. of  5 variables:
##  $ Country.Region: chr  "Afghanistan" "Albania" "Algeria" "Andorra" ...
##  $ Province      : chr  "" "" "" "" ...
##  $ Date          : chr  "24/03/2020" "08/03/2020" "24/03/2020" "16/03/2020" ...
##  $ Type          : chr  "Full" "Full" "Full" "Full" ...
##  $ Reference     : chr  "https://www.thestatesman.com/world/afghan-govt-imposes-lockdown-coronavirus-
```

```
cat("Summary of the dataset:\n")
```

```
## Summary of the dataset:
```

```
summary(lockdown)
```

```
##  Country.Region       Province             Date               Type          
##  Length:307         Length:307         Length:307         Length:307        
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##   Reference        
##  Length:307        
##  Class :character  
##  Mode  :character  
```

WorldwideVaccineData.csv

```
cat("Number of rows:", nrow(vaccine), "\n")
```

```
## Number of rows: 187
```

```
cat("Number of columns:", ncol(vaccine), "\n")
```

```
## Number of columns: 5
```

```
cat("Number of missing values:", sum(is.na(vaccine)), "\n")
```

```
## Number of missing values: 0
```

```
knitr::kable(head(vaccine), caption = "First 6 rows of the dataset")
```

Table 3: First 6 rows of the dataset

Country	Doses.administered.per.100.people	Total.doses.administered	X..of.population.vaccinated	X..of.population.fully.vaccinated
Afghanistan	17	6445359	15	13
Albania	102	2906126	46	44
Algeria	35	15205854	19	16
Angola	64	20397115	41	22

Country	Doses.administered.per.100.people	Total.doses.administered	X..of.population.vaccinated	X..of.population.fully.vaccinated
Argentina	237	106474858	92	84
Armenia	73	2150112	38	33

```r
cat("Structure of the dataset:\n")
```

```
## Structure of the dataset:
```

```r
str(vaccine)
```

```
## 'data.frame':    187 obs. of  5 variables:
##  $ Country                          : chr  "Afghanistan" "Albania" "Algeria" "Angola" ...
##  $ Doses.administered.per.100.people: int  17 102 35 64 237 73 162 229 207 137 ...
##  $ Total.doses.administered         : num  6.45e+06 2.91e+06 1.52e+07 2.04e+07 1.06e+08 ...
##  $ X..of.population.vaccinated      : num  15 46 19 41 92 38 84 88 77 53 ...
##  $ X..of.population.fully.vaccinated: num  13 44 16 22 84 33 78 86 75 48 ...
```

```r
summary(vaccine)
```

```
##    Country          Doses.administered.per.100.people Total.doses.administered
##  Length:187         Min.   :  0                       Min.   :1.714e+04
##  Class :character   1st Qu.: 62                       1st Qu.:1.810e+06
##  Mode  :character   Median :130                       Median :8.179e+06
##                     Mean   :131                       Mean   :6.493e+07
##                     3rd Qu.:199                       3rd Qu.:2.865e+07
##                     Max.   :343                       Max.   :3.408e+09
##  X..of.population.vaccinated X..of.population.fully.vaccinated
##  Min.   : 0.10               Min.   : 0.10
##  1st Qu.:36.50               1st Qu.:29.00
##  Median :62.00               Median :55.00
##  Mean   :56.91               Mean   :51.94
##  3rd Qu.:80.00               3rd Qu.:75.00
##  Max.   :99.00               Max.   :99.00
```

1.2 Wrangling the datasets

Deal with missing values, or wrong values such as negative values for new cases, deaths, vaccinated, etc., using KNN imputation. From the exploration above, only covid has missing values, so we will only impute that dataset using K Nearest Neighbors (KNN). By using KNN, we can fill in the missing values based on the values of the nearest neighbors in the dataset, allowing us keep the rest of the data for those entries.

```r
library(impute)
library(dplyr)
library(magrittr)

cat("Number of missing values before imputation:", sum(is.na(covid)), "\n")
```

```
## Number of missing values before imputation: 13
```

```
covid_imputed <- covid %>%
  select(where(is.numeric)) %>%
  as.matrix() %>%
  impute.knn(k = 5) %>%
  .$data %>%
  as.data.frame()
```

```
## Cluster size 1575 broken into 1378 197
## Done cluster 1378
## Done cluster 197
```

```
covid[, colnames(covid_imputed)] <- covid_imputed

cat("Number of missing values after imputation:", sum(is.na(covid)), "\n")
```

```
## Number of missing values after imputation: 0
```

We also want to remove any negative values from the dataset as they are not valid in some of the available features. First, let's check if there are indeed any entries that have such values. **Covid-data.csv**

```
covid %>%
  select(where(is.numeric)) %>%
  summarise(across(everything(), ~ any(. < 0))) %>%
  print()
```

```
##   total_cases new_cases total_deaths new_deaths gdp_per_capita population
## 1       FALSE      TRUE        FALSE       TRUE          FALSE      FALSE
```

In this case, new_cases and new_deaths are the only columns that have negative values, but those values are not valid. We will remove any rows that have negative values in those columns.

```
cat("Number of rows before removing negative values:", nrow(covid), "\n")
```

```
## Number of rows before removing negative values: 1575
```

```
covid <- covid %>%
  filter(new_cases >= 0 & new_deaths >= 0)
cat("Number of rows after removing negative values:", nrow(covid), "\n")
```

```
## Number of rows after removing negative values: 1568
```

Since CountryLockdowndates.csv doesn't have numerical values we proceed to WorldwideVaccineData.csv. **WorldwideVaccineData.csv**

```
vaccine %>%
  select(where(is.numeric)) %>%
  summarise(across(everything(), ~ any(. < 0))) %>%
  print()
```

```
##   Doses.administered.per.100.people Total.doses.administered
## 1                             FALSE                    FALSE
##   X..of.population.vaccinated X..of.population.fully.vaccinated
## 1                       FALSE                             FALSE
```

We can confirm that there are no negative values in this dataset, so no further action is needed.

1.3 Integrating the datasets

Now that we have cleaned the datasets, we can integrate them into one dataset. Firstly, the date attributes are in different formats for Covid-data.csv and CountryLockdowndates.csv, so we need to convert them to the same format.

```
covid$date <- as.Date(covid$date, format = "%Y-%m-%d")
lockdown$Date <- as.Date(lockdown$Date, format = "%d/%m/%Y")

cat("Covid-data.csv date format:\n", head(covid$date))
```

```
## Covid-data.csv date format:
##  18261 18262 18263 18264 18265 18266
```

```
cat("CountryLockdowndates.csv date format:\n", head(lockdown$Date))
```

```
## CountryLockdowndates.csv date format:
##  18345 18329 18345 18337 18345 NA
```