Page 1 of 15

Coronavirus COVID-19
A descriptive Comparison between Countries

Countries: AT, Germany, Iceland, IT, Spain, UK, US

Data importet from Johns Hopkins CSSE: https://github.com/CSSEGISandData/COVID-19

Population data from: https://covid.ourworldindata.org/data/ecdc/locations.csv

Statistical Analysis:
Version from: April 25, 2020

Contents

List of Tables

List of Figures

```r
# Options
options(replace.assign = TRUE, width = 80)

# disabling scientific notation in R;
options(scipen = 999)

# Set digit option;
options(digits = 2)

pdf.options(family = "Helvetica-Narrow")

# Define knitr chunk options;
opts_chunk$set(fig.width  = 5,
               fig.height = 5,
               out.width  = '12cm',
               fig.pos    = 'H',
               fig.align  = 'center',
               fig.path   = './figure/Plot-',
               fig.keep   = 'all',
               cache      = FALSE,
               echo       = TRUE,        # FALSE: no output of R code;
               message    = FALSE,       # FALSE: no output of R messages;
               warning    = FALSE,       # FALSE: no output of warnings;
               dev        = 'pdf')
```

```r
### Load R packages and Read SAS xpt files;
source("impsas_xpt.R")
```

```r
### Read chunks;
read_chunk("00_chunks.R")

### Read R functions;
source("00_r_functions.R")
```

1   COVID-19, Data Handling

Data from United Kingdom: Anguilla, Bermuda, British Virgin Islands, Cayman Islands, Channel Islands, Falkland Islands (Islas Malvinas), Gibraltar, Isle of Man, Montserrat, Turks and Caicos Islands - not included (but population data for UK used; dataset *POPDATA*).

Data from Netherlands: Aruba, Bonaire, Sint Eustatius and Saba, Curacao, Sint Maarten - not included (but population data for Netherlands used; dataset *POPDATA*).

Data from France: French Guiana, French Polynesia, Guadeloupe, Martinique, Mayotte, New Caledonia, Reunion, Saint Barthelemy, Saint Pierre and Miquelon, St Martin - not included (but population data for France used; dataset *POPDATA*).

Minor corrections of inconsistencies for State Iceland:
If STATE="Iceland" and DATE="15Mar20"d then DEATH=0; * instead of N=5;
If STATE="Iceland" and DATE="20Mar20"d then DEATH=1; * instead of N=0;

Currently following countries (with population data) are in the derived dataset *ALLSTATE*:

Table 1: Listing of Countries/States in Dataset ALLSTATE

	STATE	POPUL
1	Austria	9006400
2	Belgium	11589616
3	France	65273512
4	Germany	83783945
5	Hungary	9660350
6	Iceland	341250
7	Italy	60461828
8	Netherlands	17134873
9	Norway	5421242
10	Portugal	10196707
11	Russia	145934460
12	Spain	46754783
13	Sweden	10099270
14	Switzerland	8654618
15	US	331002647
16	United Kingdom	67886004

All counties/states can be found in the derived (long dataset) dataset: *ALLDAYS* (w/o population data)

```
## Derive Date Today - will be used in headings with \Sexpr{myDATE};
myDATE <- TODAY[1, 2]
```

2 COVID-19, Analysis from 2020-04-24

2.1 COVID-19, Total Confirmed Cases, by Country

```
cap <- "COVID 19 - Day 1 is first day with >=1 case in country"

#Select Sates (subset), e.g.;
df1STATES <- ALLSTATE %>% filter(STATE == "Austria" | STATE == "Germany" | STATE == "Iceland" |
           STATE == "Italy" | STATE == "Spain" | STATE == "United Kingdom" |
           STATE == "US")
# Dataset for CFR (%) - select countries;
df1TODAY <- TODAY %>% filter(STATE == "Austria" | STATE == "Germany" | STATE == "Iceland" |
           STATE == "Italy" | STATE == "Spain" | STATE == "United Kingdom" |
           STATE == "US")

# Master dataset (all countries/states in database)
#df1STATES <- ALLSTATE
#df1TODAY <- TODAY

xLab <- "Day"
yLab <- "Confirmed Cases"

ggplot(df1STATES, aes(x = DAY, y = CONFIRM)) +
  geom_line(aes(color = STATE), size = 1) +
  guides(color = guide_legend("State")) +
  labs(x = xLab, y = yLab) +
  theme_pubr()
```


Figure 1: COVID 19 - Day 1 is first day with $>=$1 case in country

2.2 COVID-19, Total Confirmed Cases, Day 1 first day with >=100 cases, by Country

```
cap <- "COVID 19 - Day 1 is first day with >=100 cases in country"

#df1STATES <- ALLSTATE
xLab <- "Day"
yLab <- "Confirmed Cases"

ggplot(df1STATES, aes(x = DAY100, y = CONFIRM)) +
  geom_line(aes(color = STATE), size = 1) +
  guides(color = guide_legend("State")) +
  labs(x = xLab, y = yLab) +
  theme_pubr()
```


Figure 2: COVID 19 - Day 1 is first day with >=100 cases in country

2.3 COVID-19, Total Confirmed Cases per million people, by Country

```
cap <- "COVID 19 - Day 1 is first day with >=1 case per million people in country"

#df1STATES <- ALLSTATE
xLab <- "Day"
yLab <- "Confirmed cases per million people"

ggplot(df1STATES, aes(x = DAY1M, y = CONF_MIO)) +
  geom_line(aes(color = STATE), size = 1) +
  guides(color = guide_legend("State")) +
  labs(x = xLab, y = yLab) +
  theme_pubr()
```
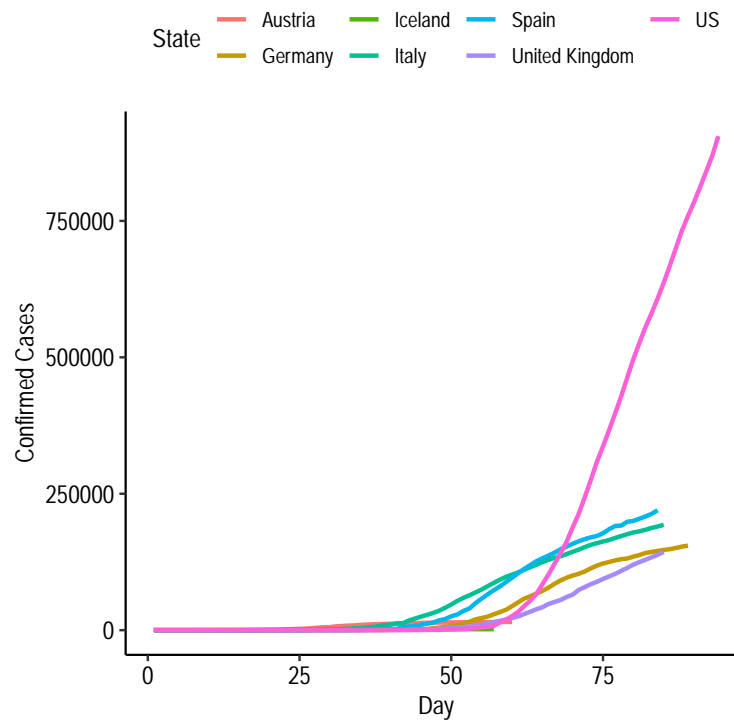

Figure 3: COVID 19 - Day 1 is first day with >=1 case per million people in country

2.4 COVID-19, Active Cases per million people, by Country

No of Active Cases = confirmed minus death minus recovered.

```
cap <- "COVID 19 - Day 1 is first day with >=1 case per mio - Active = confirmed - death - recoverd"

#df1STATES <- ALLSTATE
xLab <- "Day"
yLab <- "Active Cases per million people"

ggplot(df1STATES, aes(x = DAYDIS, y = DIS_MIO)) +
  geom_line(aes(color = STATE), size = 1) +
  guides(color = guide_legend("State")) +
  labs(x = xLab, y = yLab) +
  theme_pubr()
```
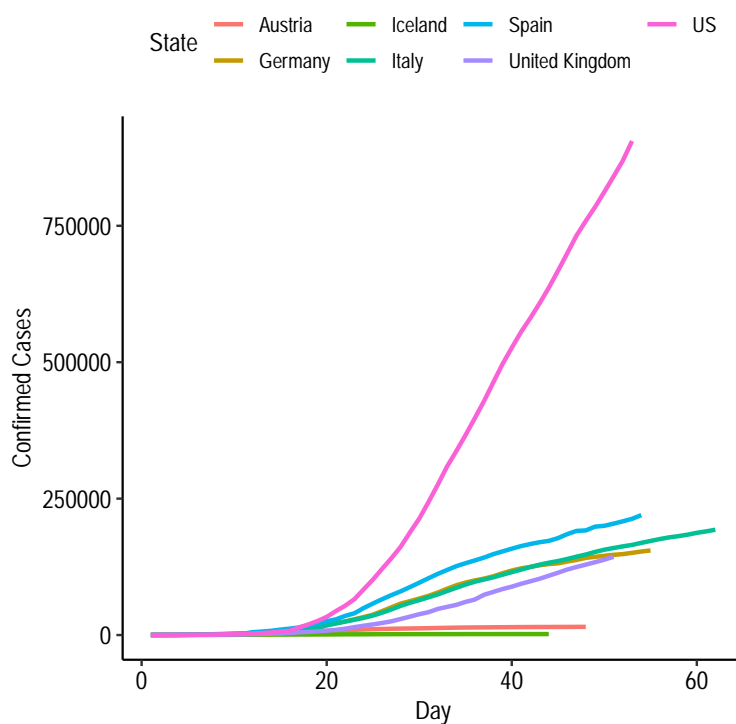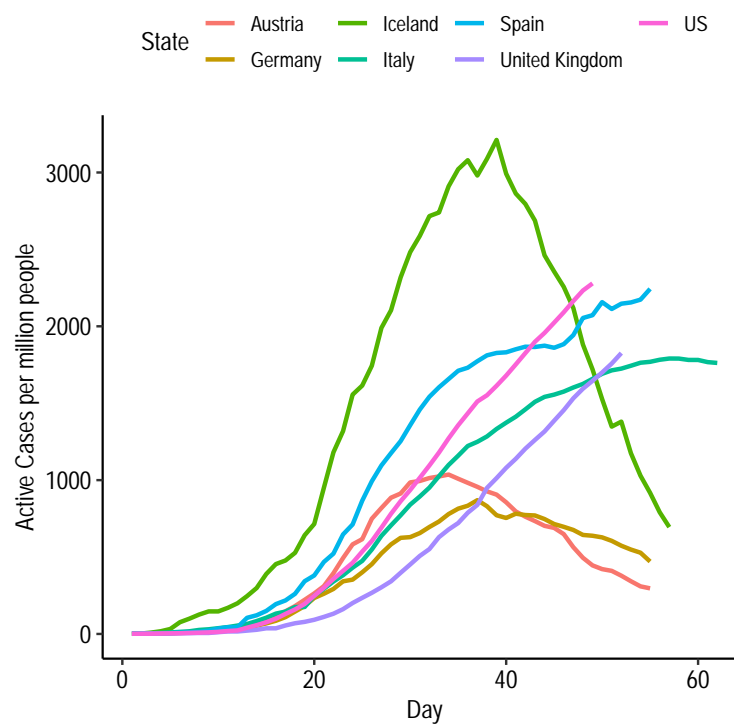

Figure 4: COVID 19 - Day 1 is first day with >=1 case per mio - Active = confirmed - death - recoverd

2.5 COVID-19, Death Cases per million people, by Country

```
cap <- "COVID 19 - Day 1 is first day with >=1 fatal case in country"

#df1STATES <- ALLSTATE
xLab <- "Day"
yLab <- "Death cases per million people"

ggplot(df1STATES, aes(x = DAY_DTH, y = DTH_MIO)) +
  geom_line(aes(color = STATE), size = 1) +
  guides(color = guide_legend("State")) +
  labs(x = xLab, y = yLab) +
  theme_pubr()
```
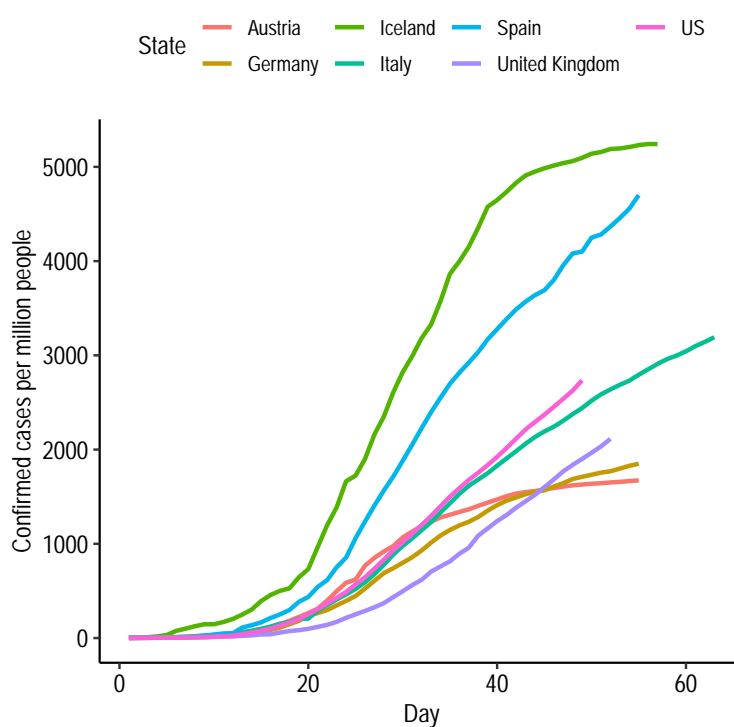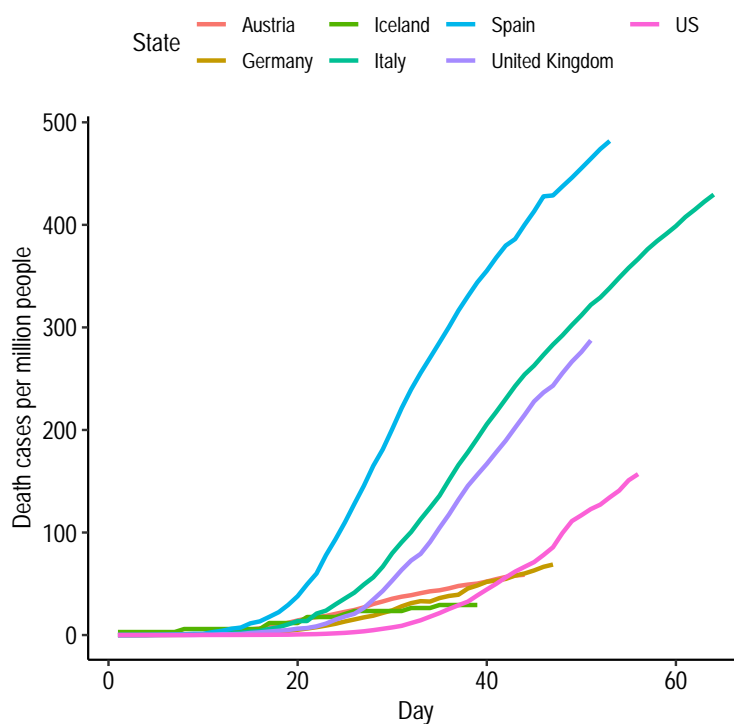

Figure 5: COVID 19 - Day 1 is first day with >=1 fatal case in country

2.6 COVID-19, Case Fatality Rate, perc., by Country - from: 2020-04-24

```
cap <- "COVID 19 - Case Fatality Rate - Total no of deaths / Total no of confrimed cases"

#df1TODAY <- TODAY
xLab <- "Country"
yLab <- "Case Fatality Rate (perc.)"

ggplot(data = df1TODAY, aes(x = STATE, y = CFR_PRC, col = STATE, fill = STATE)) +
  geom_bar(stat = "identity") +
  labs(y = yLab, x = "") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  theme(legend.position = "none")
```


Figure 6: COVID 19 - Case Fatality Rate - Total no of deaths / Total no of confrimed cases

2.7 Listing Case Fatality Rate (CFR) on 2020-04-24 - since start of COVID-19 outbreak

```
df3 <- TODAY %>%
    select(STATE, POPUL, BEG_DT, TODAY, T_DY, DEATH_DT, DEATH, CONFIRM, CFR_PRC)

my_table <- xtable(df3, caption = "Listing Case Fatality Rate",
            align = "lrrrrrrrrr", caption.placement = "top", digits = 0, label = "ListFR")

digits(my_table)[10] <- 1 # Percent (column with 1 digit);

print(my_table, caption.placement = "top", size = "small", include.rownames = TRUE,
        table.placement = "htb", tabular.environment = "longtable", floating = FALSE)
```

Table 2: Listing Case Fatality Rate

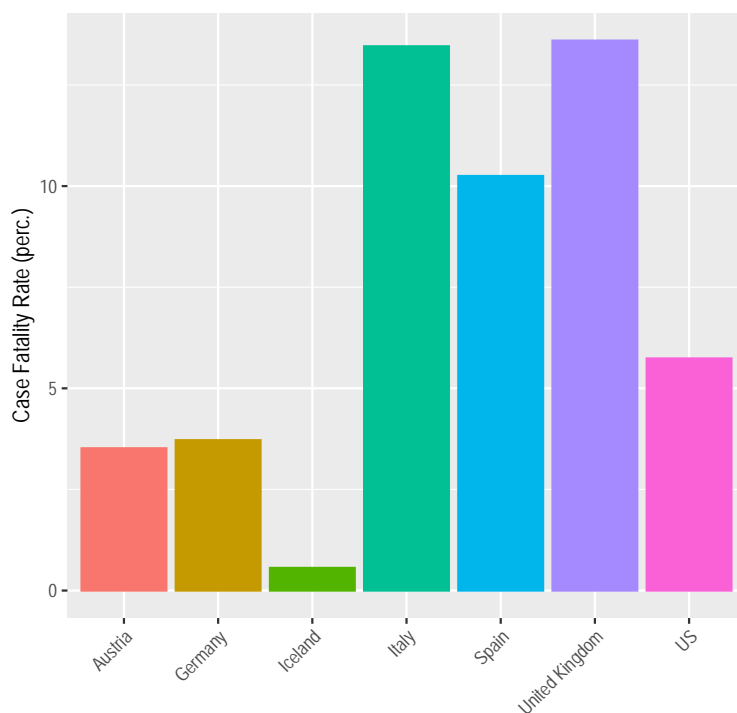|	STATE	POPUL	BEG_DT	TODAY	T_DY	DEATH_DT	DEATH	CONFIRM	CFR_PRC
1	Austria	9006400	2020-02-25	2020-04-24	60	2020-03-12 (d17)	530	15071	3.5
2	Belgium	11589616	2020-02-04	2020-04-24	81	2020-03-11 (d37)	6679	44293	15.1
3	France	65273512	2020-01-24	2020-04-24	92	2020-02-15 (d23)	22245	158636	14.0
4	Germany	83783945	2020-01-27	2020-04-24	89	2020-03-09 (d43)	5760	154999	3.7
5	Hungary	9660350	2020-03-04	2020-04-24	52	2020-03-15 (d12)	262	2443	10.7
6	Iceland	341250	2020-02-28	2020-04-24	57	2020-03-17 (d19)	10	1789	0.6
7	Italy	60461828	2020-01-31	2020-04-24	85	2020-02-21 (d22)	25969	192994	13.5
8	Netherlands	17134873	2020-02-27	2020-04-24	58	2020-03-06 (d9)	4289	36535	11.7
9	Norway	5421242	2020-02-26	2020-04-24	59	2020-03-14 (d18)	199	7463	2.7
10	Portugal	10196707	2020-03-02	2020-04-24	54	2020-03-17 (d16)	854	22797	3.7
11	Russia	145934460	2020-01-31	2020-04-24	85	2020-03-19 (d49)	615	68622	0.9
12	Spain	46754783	2020-02-01	2020-04-24	84	2020-03-03 (d32)	22524	219764	10.2
13	Sweden	10099270	2020-01-31	2020-04-24	85	2020-03-11 (d41)	2152	17567	12.3
14	Switzerland	8654618	2020-02-25	2020-04-24	60	2020-03-05 (d10)	1589	28677	5.5
15	US	331002647	2020-01-22	2020-04-24	94	2020-02-29 (d39)	51949	905358	5.7
16	United Kingdom	67886004	2020-01-31	2020-04-24	85	2020-03-05 (d35)	19506	143464	13.6

2.8 Listing Confirmed Cases, Active Cases, and Death Cases per million people on 2020-04-24

```
df4 <- TODAY %>%
    select(STATE, POPUL, CONF_MIO, DIS_MIO, DTH_MIO)

my_table <- xtable(df4, caption = "Listing Confirmed Cases, Active Cases,
                and Death Cases per million people",
        align = "lrrrrr", caption.placement = "top", digits = 0, label = "ListPerMio")

digits(my_table)[3] <- 0 # (popul));

print(my_table, caption.placement = "top", size = "small", include.rownames = TRUE,
        table.placement = "htb", tabular.environment = "longtable", floating = FALSE)
```

Table 3: Listing Confirmed Cases, Active Cases, and Death Cases per million people

	STATE	POPUL	CONF_MIO	DIS_MIO	DTH_MIO
1	Austria	9006400	1673	296	59
2	Belgium	11589616	3822	2372	576
3	France	65273512	2430	1423	341
4	Germany	83783945	1850	471	69
5	Hungary	9660350	253	178	27
6	Iceland	341250	5242	695	29
7	Italy	60461828	3192	1762	430
8	Netherlands	17134873	2132	1882	250
9	Norway	5421242	1377	1334	37
10	Portugal	10196707	2236	2032	84
11	Russia	145934460	470	428	4
12	Spain	46754783	4700	2243	482
13	Sweden	10099270	1739	1427	213
14	Switzerland	8654618	3313	703	184
15	US	331002647	2735	2279	157
16	United Kingdom	67886004	2113	1826	287

2.9   Derived Dataset - long Format: head(ALLSTATE)

```
head(ALLSTATE)

##      STATE DAY DAY100 DAY1M DAYDIS DAY_DTH       DATE CONF_MIO DIS_MIO DTH_MIO
## 1 Austria   1     NA    NA     NA      NA 2020-02-25     0.22    0.22      NA
## 2 Austria   2     NA    NA     NA      NA 2020-02-26     0.22    0.22      NA
## 3 Austria   3     NA    NA     NA      NA 2020-02-27     0.33    0.33      NA
## 4 Austria   4     NA    NA     NA      NA 2020-02-28     0.33    0.33      NA
## 5 Austria   5     NA    NA     NA      NA 2020-02-29     1.00    1.00      NA
## 6 Austria   6     NA     1      1      NA 2020-03-01     1.55    1.55      NA
##   CONFIRM DEATH RECOVER DISEASE CFR_PRC   POPUL
## 1       2     0       0       2       0 9006400
## 2       2     0       0       2       0 9006400
## 3       3     0       0       3       0 9006400
## 4       3     0       0       3       0 9006400
## 5       9     0       0       9       0 9006400
## 6      14     0       0      14       0 9006400
```

2.10   Derived Dataset - long Format: str(ALLSTATE)

```
str(ALLSTATE)

## 'data.frame': 1180 obs. of  16 variables:
##  $ STATE   : Factor w/ 16 levels "Austria","Belgium",..: 1 1 1 1 1 1 1 1 1 1 ...
##   ..- attr(*, "label")= chr "State"
##  $ DAY     : num  1 2 3 4 5 6 7 8 9 10 ...
##   ..- attr(*, "label")= chr "Day (>=1 case)"
##  $ DAY100  : num  NA NA NA NA NA NA NA NA NA NA ...
##   ..- attr(*, "label")= chr "Day (>=100 cases)"
##  $ DAY1M   : num  NA NA NA NA NA 1 2 3 4 5 ...
##   ..- attr(*, "label")= chr "Day (>=1 cases per mio)"
##  $ DAYDIS  : num  NA NA NA NA NA 1 2 3 4 5 ...
##   ..- attr(*, "label")= chr "Day (>=1 cases per mio) Active"
##  $ DAY_DTH : num  NA NA NA NA NA NA NA NA NA NA ...
##   ..- attr(*, "label")= chr "Day (>=1 case fatal)"
##  $ DATE    : Date, format: "2020-02-25" "2020-02-26" ...
##  $ CONF_MIO: num  0.222 0.222 0.333 0.333 0.999 ...
##   ..- attr(*, "label")= chr "Confirmed (per mio)"
##  $ DIS_MIO : num  0.222 0.222 0.333 0.333 0.999 ...
##   ..- attr(*, "label")= chr "Active (per mio)"
##  $ DTH_MIO : num  NA NA NA NA NA NA NA NA NA NA ...
##   ..- attr(*, "label")= chr "Death (per mio)"
##  $ CONFIRM : num  2 2 3 3 9 14 18 21 29 41 ...
##   ..- attr(*, "label")= chr "Confirmed"
##  $ DEATH   : num  0 0 0 0 0 0 0 0 0 0 ...
##   ..- attr(*, "label")= chr "Death"
##  $ RECOVER : num  0 0 0 0 0 0 0 0 0 0 ...
##   ..- attr(*, "label")= chr "Recovered"
##  $ DISEASE : num  2 2 3 3 9 14 18 21 29 41 ...
##   ..- attr(*, "label")= chr "Active"
##  $ CFR_PRC : num  0 0 0 0 0 0 0 0 0 0 ...
##   ..- attr(*, "label")= chr "Death per Confirmed (perc.)"
##  $ POPUL   : num  9006400 9006400 9006400 9006400 9006400 ...
##   ..- attr(*, "label")= chr "Population"
```

A Technical Details

A.1 R Version

```
R.Version()

## $platform
## [1] "x86_64-w64-mingw32"
##
## $arch
## [1] "x86_64"
##
## $os
## [1] "mingw32"
##
## $system
## [1] "x86_64, mingw32"
##
## $status
## [1] ""
##
## $major
## [1] "3"
##
## $minor
## [1] "6.1"
##
## $year
## [1] "2019"
##
## $month
## [1] "07"
##
## $day
## [1] "05"
##
## $`svn rev`
## [1] "76782"
##
## $language
## [1] "R"
##
## $version.string
## [1] "R version 3.6.1 (2019-07-05)"
##
## $nickname
## [1] "Action of the Toes"
```

A.2 Active R packages

```
sessionInfo()

## R version 3.6.1 (2019-07-05)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 7 x64 (build 7601) Service Pack 1
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=German_Austria.1252  LC_CTYPE=German_Austria.1252
## [3] LC_MONETARY=German_Austria.1252 LC_NUMERIC=C
## [5] LC_TIME=German_Austria.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] RColorBrewer_1.1-2 car_3.0-5        carData_3.0-3   texreg_1.36.23
##  [5] reporttools_1.1.2  ggthemes_4.2.0   survminer_0.4.6 ggpubr_0.2.4
##  [9] magrittr_1.5       gridExtra_2.3    xtable_1.8-4    Hmisc_4.3-0
## [13] Formula_1.2-3      survival_3.1-7   lattice_0.20-38 forcats_0.4.0
## [17] stringr_1.4.0      dplyr_0.8.3      purrr_0.3.3     readr_1.3.1
## [21] tidyr_1.0.0        tibble_2.1.3     ggplot2_3.2.1   tidyverse_1.3.0
## [25] foreign_0.8-72     knitr_1.26
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-142     fs_1.3.1         lubridate_1.7.4
##  [4] httr_1.4.1       tools_3.6.1      backports_1.1.5
##  [7] R6_2.4.1         rpart_4.1-15     DBI_1.1.0
## [10] lazyeval_0.2.2   colorspace_1.4-1 nnet_7.3-12
## [13] withr_2.1.2      tidyselect_0.2.5 curl_4.3
## [16] compiler_3.6.1   cli_2.0.0        rvest_0.3.5
## [19] htmlTable_1.13.3 xml2_1.2.2       labeling_0.3
## [22] scales_1.1.0     checkmate_1.9.4  survMisc_0.5.5
## [25] digest_0.6.23    rio_0.5.16       base64enc_0.1-3
## [28] pkgconfig_2.0.3  htmltools_0.4.0  dbplyr_1.4.2
## [31] highr_0.8        htmlwidgets_1.5.1 rlang_0.4.2
## [34] readxl_1.3.1     rstudioapi_0.10  farver_2.0.1
## [37] generics_0.0.2   zoo_1.8-6        jsonlite_1.6
## [40] zip_2.0.4        acepack_1.4.1    Matrix_1.2-17
## [43] Rcpp_1.0.3       munsell_0.5.0    fansi_0.4.0
## [46] abind_1.4-5      lifecycle_0.1.0  stringi_1.4.3
## [49] grid_3.6.1       crayon_1.3.4     haven_2.2.0
## [52] splines_3.6.1    hms_0.5.2        zeallot_0.1.0
## [55] pillar_1.4.2     ggsignif_0.6.0   reprex_0.3.0
## [58] glue_1.3.1       evaluate_0.14    latticeExtra_0.6-28
## [61] data.table_1.12.8 modelr_0.1.5    vctrs_0.2.0
## [64] cellranger_1.1.0 gtable_0.3.0     km.ci_0.5-2
## [67] assertthat_0.2.1 xfun_0.11        openxlsx_4.1.4
## [70] broom_0.5.3      KMsurv_0.1-5     cluster_2.1.0
```

A.3 PDF LaTeX

```
Sys.which("pdflatex")

##                          pdflatex
## "C:\\texlive\\2017\\bin\\win32\\pdflatex.exe"
```

A.4 System Time

```
Sys.time()
```

```
## [1] "2020-04-25 19:10:48 CEST"
```

A.5 Warranty

```
cat("This program and the derived dataset(s) come WITHOUT ANY WARRANTY")
```

```
## This program and the derived dataset(s) come WITHOUT ANY WARRANTY
```