2_Analysis

AnfPrak

2022-04-11

```
#######################################
#### Package / Library setup
#######################################

# Specifies which packages are used and installs / loads all that are required
lib_need <- c("tidyverse")
lib_have <- lib_need %in% rownames(installed.packages())

if(any(!lib_have)) install.packages(lib_need[!lib_have])
invisible(lapply(lib_need, library, character.only = TRUE))

rm(lib_have, lib_need)

#######################################
```

```
#######################################
#### Data import
#######################################

# Requires: cases_complete.rds (after it was merged in RKI_Merge.Rmd)
#           12411-0010.csv (state level population data)
#           12411-0015.csv (district level population data)
#
# Output:   cases_complete (df)
#           state_population (df)
#           district_population (df)


setwd(getwd())


#######################################

# Decide whether to load the data which was presented or the dataset extended
# by the missing dates after the presentation
SWITCH_data_extended <- 1

if (SWITCH_data_extended == 0) {
  cases_complete <- readRDS("03_1_cases_complete_Vortrag.rds")
} else {
  cases_complete <- readRDS("03_2_cases_complete_extended.rds")
}
```

```
# Loading state level population data
# Current as of 31.12.2020
# Source:       https://www-genesis.destatis.de/genesis/online
# Table Code:   12411-0010
state_population <- read.csv("03_5_12411-0010.csv",
                        sep = ";",
                        header = FALSE,
                        skip = 6,
                        nrows = 16,
                        fileEncoding = "latin1")

# Loading district level population data
# Current as of 31.12.2020
# Source:       https://www-genesis.destatis.de/genesis/online
# Table Code:   12411-0015
district_population <- read.csv("03_6_12411-0015.csv",
                        sep = ";",
                        header = FALSE,
                        skip = 6,
                        nrows = 476,
                        fileEncoding = "latin1")


#####################################
```

```
#####################################
#### cases_complete cleaning
#####################################

# Correct spelling for some of the state entries
unique(cases_complete$state)
```

```
##  [1] "Niedersachsen"          "Nordrhein-Westfalen"    "Schleswig-Holstein"
##  [4] "Hamburg"                "Hessen"                 "Rheinland-Pfalz"
##  [7] "Baden-Württemberg"      "Bremen"                 "Bayern"
## [10] "Brandenburg"            "Mecklenburg-Vorpommern" "Sachsen"
## [13] "Sachsen-Anhalt"         "Berlin"                 "Thüringen"
## [16] "Saarland"               "Baden-WÃ¼rttemberg"     "ThÃ¼ringen"
```

```
cases_complete$state[cases_complete$state == "ThÃ¼ringen"] <- "Thüringen"
cases_complete$state[cases_complete$state == "Baden-WÃ¼rttemberg"] <- "Baden-Württemberg"
unique(cases_complete$state)
```

```
##  [1] "Niedersachsen"          "Nordrhein-Westfalen"    "Schleswig-Holstein"
##  [4] "Hamburg"                "Hessen"                 "Rheinland-Pfalz"
##  [7] "Baden-Württemberg"      "Bremen"                 "Bayern"
## [10] "Brandenburg"            "Mecklenburg-Vorpommern" "Sachsen"
## [13] "Sachsen-Anhalt"         "Berlin"                 "Thüringen"
## [16] "Saarland"
```

```
#####################################
```

```
#####################################
#### Calculation of notification delay
#####################################
```

```r
cases_complete$ADD_delay <- cases_complete$publication_date - cases_complete$date - 1

#######################################
```

```r
#######################################
#### Adding weekday info for cycle calculation
#######################################

# First day of the week is Monday = 1. The earliest date is 2020-01-01 which is
# a Wednesday (ie. = 3)
daily_dates <- seq(as.Date("2020-01-01"),
                   as.Date("2022-02-22"),
                   by = "days")
daily_wedays <- c(3, 4, 5, 6, 7, 1, 2)

n_reps <- length(daily_dates) / 7
daily_wedays <- rep(daily_wedays, n_reps)

day_type <- data.frame(daily_dates, daily_wedays)
names(day_type) = c("date", "ADD_weekday")

cases_complete <- merge(x = cases_complete,
                        y = day_type,
                        by = "date",
                        all.x = TRUE)

rm(daily_dates, daily_wedays, n_reps, day_type)
gc()
```

```
##            used  (Mb) gc trigger   (Mb) limit (Mb)  max used   (Mb)
## Ncells  1076673  57.6    7094706  378.9         NA   8594353  459.0
## Vcells 84909513 647.9  260294140 1985.9      65536 324831642 2478.3
```

```r
#######################################
```

```r
#######################################
#### District level population data
#######################################

names(state_population) <- c("state", "ADD_state_population")
names(district_population) <- c("districtId", "V2","district_population")

# Some of the district classifications listed in the file are no longer in use
# (they are probably just included for historical reasons) and do not carry
# any population information anyway (verified by comparing the general population
# data above with the sum of all individual districts after omitting NAs).
district_population$district_population <- as.numeric(district_population$district_population)

# Removing old districts
district_population <- na.omit(district_population)


# Extracting the type of district from the field (separated by a ",")
district_population <- separate(data = district_population,
                                col = "V2",
```

```
                                into = c("district", "district_type"),
                                sep = "," )

# Some of the district names have "kreis" in their name rather than it being
# specified directly after a comma (thus the comma is missing). These values end
# up being NAs. This has been confirmed for all cases by manual inspection and
# can therefore be rectified manually.
district_population$district_type[is.na(district_population$district_type)] <- " Landkreis"

# Manual inspection shows that the only district with the addendum to "kreisfreie
# Stadt" is "Eisenach". This can be standardised manually.
district_population$district_type[district_population$district_type == " kreisfreie Stadt (bis 30.06.20

# rm leading spaces
district_population$district_type <- trimws(district_population$district_type)

#####################################



#########################################
#### OPTIONAL: Investigation of "mergability" of datasets
#########################################

# Trigger switch to enable verification
SWITCH_verification <- 0


if (SWITCH_verification == 1) {
  # Verifying that removing old districts from the district_population dataset
  # does not lead to deletion of population data (ie. it still adds up)
  sum(state_population$state_population) == sum(district_population$district_population)


  length(unique(cases_complete$district)) # 417 unique in cases_complete
  length(unique(cases_complete$districtId)) # 413 unique in cases_complete
  length(unique(district_population$districtId)) # 401 unique in district_pop


  # Overlap between districtId sets (cases_complete w/ district_population)
  #
  # Findings:
  #   (1) district_population treats Berlin as a single district whereas
  #       cases_complete specifies 12 individual sub-districts within Berlin
  #   (2) There are some NA for cases_complete$districtId
  #
  # Action: Recode all individual sub-districts in cases_complete as a single
  #         district (11000) [this is the official classification]
  unique(district_population$districtId)[which(!(unique(district_population$districtId) %in% unique(cas
  unique(cases_complete$districtId)[which(!(unique(cases_complete$districtId) %in% unique(district_popu


  # Classifying the district_type from cases_complete. separate() splits at
```

```
  # spaces which results in some district names being cut off. Since we only
  # care about the district_type though (which stays intact), this is not a
  # problem
  temp <- separate(data = cases_complete,
                   col = district,
                   into = c("district_type", "district"),
                   sep = " ")
  temp <- temp[,c("district_type", "district", "districtId", "landId")]


  # Findings:
  #    (1) Most district_types are either LK (Landkreis) or SK (Städtekreis?),
  #        however, there are a 4 odd entries
  #    (2) All of the odd entries can be attributed to Hannover (Region) and
  #        Aachen (the rest) which allows for manual fixing (both are LK)
  #
  # Note: Aachen used to have a dual classification where part of it was an LK
  #       and another part an SK. In 2009 these two parts were combined into
  #       a single LK (see district population table 12411-0015)
  unique(temp$district_type)
  unique(temp$district[temp$district_type == c("Region", "StadtRegion", "StädteRegion", "Städteregion")])
  unique(temp$district_type[temp$district == "Aachen"])
  unique(temp$district_type[temp$district == "Hannover"])

  rm(temp)

  # The cases_complete dataset always has a value for state and district. The
  # district name, however, does not match the format of district_population
  # The merge therefore has to happen via districtId which is not available
  # for all entries of cases_complete (but can be generated)
  any(is.na(cases_complete$state))
  any(is.na(cases_complete$district))
  any(is.na(cases_complete$districtId))

}

gc()
```

```
##            used  (Mb) gc trigger   (Mb) limit (Mb)  max used   (Mb)
## Ncells  1094040  58.5    5675765  303.2         NA   8594353  459.0
## Vcells 84950050 648.2  260294140 1985.9      65536 324831642 2478.3
```

```
########################################
```

```
########################################
#### DistrictId completion for cases_complete
#### District level population data integration
########################################

# Process:
#    (1) Create a map of cases_complete$districts to cases_complete$districtId
#    (2) Change value for Berlin sub districts
#    (3) Impute ID values from map. Add values as ADD_ variables to not lose data

# Using temp variable to reduce RAM usage
```

```r
# Ensuring all district names (provided by the RKI) are included
temp <- cases_complete[, c("districtId", "district")]
temp <- temp[match(unique(temp$district), temp$district),]

# Ensuring all districtIds (provided by the RKI) are included
temp2 <- cases_complete[, c("districtId", "district")]
temp2 <- temp2[match(unique(temp2$districtId), temp2$districtId),]

# Joining temporary datasets and identifying which district names are missing
# an ID (result: 6). These are added manually based on the district population
# dataset (table 12411-0015)
temp3 <- left_join(temp, temp2, by = "district")
temp3$district[is.na(temp3$districtId.y)]
```

```
## [1] "LK Neustadt a.d.Waldnaab" "LK Göttingen (alt)"
## [3] "LK Saarpfalz-Kreis"       "Städteregion Aachen"
## [5] "LK Aachen"                "StädteRegion Aachen"
```

```r
temp3$districtId.y[temp3$district == "LK Peine"] <- 3157
temp3$districtId.y[temp3$district == "StadtRegion Aachen"] <- 5334
temp3$districtId.y[temp3$district == "StädteRegion Aachen"] <- 5334
temp3$districtId.y[temp3$district == "LK Aachen"] <- 5334
temp3$districtId.y[temp3$district == "LK Göttingen (alt)"] <- 3159
temp3$districtId.y[temp3$district == "LK Saar-Pfalz-Kreis"] <- 10045


# Verifying that all district names supplied by the RKI are in the new dataset
all(temp$district %in% temp3$district)
```

```
## [1] TRUE
```

```r
# Checking whether there are any omitted districtIds from the RKI dataset
# (result: only the NA value has been omitted)
temp2$districtId[!(temp2$districtId %in% temp3$districtId.y)]
```

```
## numeric(0)
```

```r
# Omitting redundant districtId variable and copying district for later
# verification with cases_complete
temp3 <- temp3[, c("district", "districtId.y")]
temp3$ADD_district <- temp3$district
names(temp3) <- c("district", "ADD_districtId", "ADD_district")

# LK Meißen was added twice because the original file listed it once as NA and
# once with district ID as 14627 This is in first entry but programmed to be
# robust so that the first instance of LK Peine is removed
temp3 <- temp3[-c(which(is.na(temp3$ADD_districtId))), ]

# Recoding the Berlin ID to district map (combining sub districts)
temp3$ADD_districtId[temp3$ADD_districtId > 11000 & temp3$ADD_districtId < 12000] <- 11000

# Check if map is complete
all(unique(cases_complete$district) %in% temp3$district)
```

```
## [1] FALSE
```

```r
# Integrating district population data
cases_complete <- left_join(cases_complete, temp3, by = c("district"))

rm(temp, temp2, temp3)

#####################################

# District level population data integration
# Renaming districtId because merger happens via safe variable. district_clean
# are the better formatted district variables from the district_population data
names(district_population) <- c("ADD_districtId", "ADD_district_clean", "ADD_district_type", "ADD_distri
cases_complete <-left_join(cases_complete, district_population, by = "ADD_districtId")


# State level population data integration
cases_complete <- left_join(cases_complete, state_population, by = "state")

#####################################



########################################
#### OPTIONAL: Verification of successful merge
########################################

# Trigger switch to enable verification
SWITCH_verification <- 0


if (SWITCH_verification == 1) {

  # Using temp variable for safety
  temp <- cases_complete
  temp$check1 <- temp$district == temp$ADD_district
  temp$check2 <- temp$districtId == temp$ADD_districtId

  # All integrated district fields are the same as the RKI data (not surprising)
  any(isFALSE(temp$check1))

  # The only districtId values which do not correspond to the ones provided by
  # RKI are those manually replaced for the Berlin district
  temp <- temp[,c("check2", "district", "ADD_districtId")]
  temp <- na.omit(temp)
  unique(temp$district[temp$check2 == FALSE])

  # Check whether district_population and cases_complete overlap with ID. Both
  # do hence the datasets should be successfully integrated
  unique(temp$ADD_districtId)[which(!(unique(temp$ADD_districtId) %in% unique(district_population$ADD_di
  unique(district_population$ADD_districtId)[which(!(unique(district_population$ADD_districtId) %in% uni

  rm(temp)

}
```

```
############################################################
```

```
#######################################
#### Data export
#######################################

# Export depending on whether the Vortrag data or the extended data was loaded
if (SWITCH_data_extended == 0) {
  saveRDS(cases_complete, "03_7_cases_complete_Vortrag_ADD.rds")
} else {
  saveRDS(cases_complete, "03_8_cases_complete_extended_ADD.rds")
}

gc()
```

```
##             used  (Mb) gc trigger   (Mb) limit (Mb)  max used   (Mb)
## Ncells   1100987  58.8    3632490  194.0         NA   8594353  459.0
## Vcells 105632977 806.0  312432968 2383.7      65536 324831642 2478.3
```

```
#######################################
```