

COS-D407. *Scientific Modeling and Model Validation*

Hands-on excercises

Week 5

University of Helsinki, Finland

26.10.2020–09.12.2020

Lecturer: Christina Bohk-Ewald

Source: <https://github.com/christina-bohk-ewald/2020-COS-D407-scientific-modeling-and-model-validation>

Table of content:

- 1. Some preparations in R**
- 2. Download, load, and prepare required input data**
- 3. Analyze and compare D_x estimates for Finland from different sources**
- 4. Estimate COVID-19 infections in Finland based on different D_x estimates**
- 5. Time for you to think both creatively and critically about the meaning of these results**

1. Some preparations in R

1.1 Open a new script for week 5 in R (e.g., *week-5.R*) and save it to a folder of your choice (e.g., *course-COS-D407*).

1.2 Create a filepath to this folder from where you would like to load data and to where you would like to save your outcome. For example,

```
the.course-COS-D407.path <- c("C:/course-COS-D407")
```

1.3 You can then set the working directory to this path

```
setwd(the.course-COS-D407.path)
```

2. Download, load, and prepare required input data

In week 5 we analyze the sensitivity of the demographic scaling model's COVID-19 infection estimates for Finland with respect to D_x estimates as input. We will start with collecting, loading, and comparing D_x estimates from different sources and will continue with estimating and analyzing COVID-19 infection estimates for Finland with the demographic scaling model based on these different D_x estimates.

Note that you know already the steps 2.1 through 2.7 from the previous lab session, step 2.8 is new.

2.1 Download confirmed cases and reported deaths attributable to COVID-19 as of today

Please go to the website of the Johns Hopkins University CSSE. The files

- *time_series_covid19_confirmed_global.csv*
- *time_series_covid19_deaths_global.csv*

contain confirmed cases and reported deaths, respectively, for many countries on a daily basis since January 22, 2020. Please download these two files and save them in your project folder.

2.2 Load COVID-19 data

Please load the numbers of confirmed cases and reported deaths from COVID-19 in R using the function *read.csv* of the R-package *openxlsx*.

```
require(openxlsx)
```

```
confirmed <- read.csv("time_series_covid19_confirmed_global.csv",header=TRUE,  
stringsAsFactors = FALSE)  
confirmed[1:2,1:6]
```

```
## Province.State Country.Region      Lat      Long X1.22.20 X1.23.20  
## 1                Afghanistan 33.93911 67.70995      0      0  
## 2                Albania 41.15330 20.16830      0      0
```

```
deaths <- read.csv("time_series_covid19_deaths_global.csv",header=TRUE,  
stringsAsFactors = FALSE)  
deaths[1:2,((ncol(deaths)-5):ncol(deaths))]
```

```
## X9.10.20 X9.11.20 X9.12.20 X9.13.20 X9.14.20 X9.15.20  
## 1      1420      1420      1420      1420      1425      1426  
## 2       324       327       330       334       338       340
```

2.3 Download and load abridged life tables

Please go to the UNWPP2019 website, download abridged life tables for both sexes together, save them into your project folder, and then load them into R.

```
lt_1950_2020 <- read.xlsx("WPP2019_MORT_F17_1_ABRIDGED_LIFE_TABLE_BOTH_SEXES.xlsx",
sheet = 1, startRow = 17)
```

Brief data description. The data object *lt_1950_2020* contains abridged life tables for both sexes for all UN countries.

Explore this data object and find out how large Finnish remaining life expectancy at birth has been 1950-55 through 2015-19.

```
## lt_1950_2020[1:2,]

colnames(lt_1950_2020)

## [1] "Index"
## [2] "Variant"
## [3] "Region,.subregion,.country.or.area.*"
## [4] "Notes"
## [5] "Country.code"
## [6] "Type"
## [7] "Parent.code"
## [8] "Period"
## [9] "Age.(x)"
## [10] "Age.interval.(n)"
## [11] "Central.death.rate.m(x,n)"
## [12] "Probability.of.dying.q(x,n)"
## [13] "Probability.of.surviving.p(x,n)"
## [14] "Number.of.survivors.l(x)"
## [15] "Number.of.deaths.d(x,n)"
## [16] "Number.of.person-years.lived.L(x,n)"
## [17] "Survival.ratio.S(x,n)"
## [18] "Person-years.lived.T(x)"
## [19] "Expectation.of.life.e(x)"
## [20] "Average.number.of.years.lived.a(x,n)"

lt_1950_2020[which(lt_1950_2020[, "Region,.subregion,.country.or.area.*"] == "Finland" &
lt_1950_2020["Age.(x)"] == 0), c("Period", "Expectation.of.life.e(x)"]]
```

```
##      Period Expectation.of.life.e(x)
## 4823 1950-1955      66.401073999999994
## 10349 1955-1960      68.189361000000005
## 15875 1960-1965      69.067266000000004
## 21401 1965-1970      69.716247999999993
## 26927 1970-1975      70.930030000000002
## 32453 1975-1980      72.714505000000003
## 37979 1980-1985      74.326365999999993
## 43505 1985-1990      74.788223000000002
## 49031 1990-1995      75.841427999999993
## 54557 1995-2000      77.144569000000004
## 60083 2000-2005      78.400789000000003
## 65609 2005-2010      79.548135000000002
## 71135 2010-2015      80.705350999999993
## 76661 2015-2020      81.646075999999994
```

2.4 Load global pattern over age of COVID-19 deaths

Dudel et al. (2020) provide data on age-specific death counts attributable to COVID-19 in their supplementary material. These data have served as a basis for calculating a global average pattern over age for total death counts as input for the demographic scaling model. You can download this *global average pattern over age* from the GitHub repository for this course.

```
global_age_dist_deaths <- source("global_age_dist_deaths.R")
## global_age_dist_deaths
```

Brief data description. The data object *global_age_dist_deaths* contains the global pattern over 10-year age groups of COVID-19 deaths.

2.5 Load infection fatality rates from Verity et al. (2020)

Verity and colleagues (2020) report infection fatality rates by 10-year age groups for Hubei province, China, on page 5. Please create a data object *ifr_by_age_china_verity* that contains these data or download it from the GitHub repository for this course.

```
ifr_by_age_china_verity <- read.table("infection-fatality-rates-by-age-china-Verity.txt",
header=FALSE, stringsAsFactors = FALSE)
```

```
ifr_by_age_china_verity
```

```
##      V1      V2      V3      V4
## 1  0 1.6e-05 1.85e-06 0.000249
## 2 10 7.0e-05 1.50e-05 0.000500
## 3 20 3.1e-04 1.40e-04 0.000920
## 4 30 8.4e-04 4.10e-04 0.001850
## 5 40 1.6e-03 7.60e-04 0.003200
## 6 50 6.0e-03 3.40e-03 0.013000
## 7 60 1.9e-02 1.10e-02 0.039000
## 8 70 4.3e-02 2.50e-02 0.084000
## 9 80 7.8e-02 3.80e-02 0.133000
```

Brief data description. The data object *ifr_by_age_china_verity* contains the modal estimate as well as the lower and upper bound of the 95 percent credible interval of the infection fatality rates of Hubei province, China, by 10-year age groups.

2.6 Load infection fatality rates from Salje et al. (2020)

Salje and colleagues (2020) report infection fatality rates by 10-year age groups for France. Please create a data object *ifr_by_age_france_salje* that contains these data or download it from the GitHub repository for this course.

```
ifr_by_age_france_salje <- read.table("infection-fatality-rates-by-age-france-Salje.txt",
header=FALSE, stringsAsFactors = FALSE)
```

```
ifr_by_age_france_salje <- ifr_by_age_france_salje/100
```

```
ifr_by_age_france_salje
```

```
##      V1      V2      V3      V4
## 1 0.0 0.00001 1.0e-06 0.00002
## 2 0.1 0.00001 1.0e-06 0.00002
## 3 0.2 0.00007 3.0e-05 0.00010
## 4 0.3 0.00020 1.0e-04 0.00040
## 5 0.4 0.00060 3.0e-04 0.00090
```

```
## 6 0.5 0.00200 1.0e-03 0.00360
## 7 0.6 0.00900 5.0e-03 0.01400
## 8 0.7 0.02400 1.4e-02 0.03700
## 9 0.8 0.10100 6.0e-02 0.15600
```

Brief data description. The data object *ifr_by_age_france_salje* contains the modal estimate as well as the lower and upper bound of the 95 percent credible interval of the infection fatality rates of France by 10-year age groups.

2.7 Infection fatality rates from Levin et al. (2020)

Levin and colleagues (2020) propose an exponential model function to determine infection fatality rates by single years of age. Please create a data object *ifr_by_age_levin* that contains these data for 10-year age groups.

```
ifr_by_age_levin <- c(0)
current_ifr_sum <- c(0)
exp_IFR <- exp(-7.53 + 0.119 * (seq(0.5,90.5,1))) / 100
for(group in 1:8){
  pos <- (1+10*(group-1)):(10+10*(group-1))
  current_ifr_sum[group] <- sum(exp_IFR[pos])
  ifr_by_age_levin[group] <- current_ifr_sum[group]/10
} ## group
current_ifr_sum[9] <- sum((exp_IFR[(pos[length(pos)]+1):(length(exp_IFR))]))
ifr_by_age_levin[9] <- current_ifr_sum[9]/length((pos[length(pos)]+1):(length(exp_IFR)))

round(ifr_by_age_levin,5)
```

```
## [1] 0.00001 0.00003 0.00011 0.00037 0.00120 0.00396 0.01300 0.04275 0.15094
```

Brief data description. The data object *ifr_by_age_levin* contains the central estimate of infection fatality rates by 10-year age groups.

2.8 Load age-specific death counts from the COVerAGE-DB

Based on R code provided by Dudel and colleagues on GitHub:

```
library(tidyverse)
library(httr)

#
## Load data:
#

## Online source
url <- 'https://osf.io/7tnfh/download'

## Create a folder *Data*
if (!"Data" %in% dir()){
  dir.create("Data")
}

## Filename and where to save it on your computer
filename <- 'Data/Output_5.zip'

## Download the data
GET(url, write_disk(filename, overwrite = TRUE))
```

```

## Unzip the downloaded and saved zip file
unzip(filename)

## Load the data in R
Dat <- read_csv(filename,skip=3)

#
## Select data for Finland
#

## Determine country of interest
countrylist <- c("Finland")
region <- c("All")

## Select data for country of interest
dat <- Dat %>% filter(Country %in% countrylist & Region %in% region)

## Select data by sex (here: both)
dat <- dat %>% filter(Sex=="b")

## Keep only variables of interest
dat <- dat %>% select(Country,Date,Sex,Age,Deaths)

```

Brief data description. The data object *dat* contains death counts (column 5) by 5-year age groups (column 4), sex (column 3), date (column 2), and country (here: Finland; column 1).

Try to explore the Finnish data from the COVerAGE-DB a bit further. For what ages are they available? For what dates are they available?

```

dat

## # A tibble: 2,541 x 5
##   Country Date      Sex    Age Deaths
##   <chr>   <chr>   <chr> <dbl> <dbl>
## 1 Finland 06.04.2020 b         0      0
## 2 Finland 06.04.2020 b         5      0
## 3 Finland 06.04.2020 b        10      0
## 4 Finland 06.04.2020 b        15      0
## 5 Finland 06.04.2020 b        20      0
## 6 Finland 06.04.2020 b        25      0
## 7 Finland 06.04.2020 b        30      0
## 8 Finland 06.04.2020 b        35      0
## 9 Finland 06.04.2020 b        40     0.3
## 10 Finland 06.04.2020 b        45     0.5
## # ... with 2,531 more rows

unique(dat$Age)

## [1] 0 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90
## [20] 95 100

head(unique(dat$Date))

## [1] "06.04.2020" "09.04.2020" "20.04.2020" "22.04.2020" "24.04.2020"
## [6] "27.04.2020"

```

```
tail(unique(dat$Date))
```

```
## [1] "10.10.2020" "11.10.2020" "13.10.2020" "16.10.2020" "17.10.2020"  
## [6] "18.10.2020"
```

Something to think about. How can you select COVID-19 related data from the COVer-AGE-DB for another country?

3. Analyze and compare D and D_x estimates from different sources

An easy way to compare the Finnish D and D_x estimates based on data of the (1) JHU & the global age pattern as well as the (2) COVerAGE-DB is to plot them.

However, to do this, you first need to prepare the data a bit so that they are in the same format by age and time: that is, deaths need to be available by 10-year age groups and by single days using the same date format.

3.1 Source basic functions of the demographic scaling model

You can find the basic functions of the demographic scaling model in the file *basic-functions-week-5.R* in the GitHub repository for this course. They contain the functions:

- *to_ungroup* and *to_ungroup_spar* to interpolate IFR estimates of Verity et al. (2020) and Salje et al. (2020) into single years of age
- *get_ungrouped_ex_2015_2020* to ungroup remaining life expectancy
- *map_ifr_betw_ref_and_one_coi_thanatAge* and *map_ifr_betw_assigned_ref_and_one_coi_thanatAge* to scale IFRs from a reference country (here: China; France) onto a country of interest (here: Finland) based on remaining life expectancy
- *aggregate_mapped_ifr_10y* to aggregate scaled IFRs into 10-year age groups
- *disaggregate_deaths_one_coi_10y* to disaggregate total deaths into 10-year age

You may have a look at these basic functions if you wish. But it is also fine to just source (or load) them via the file *basic-functions-week-5.R*:

```
source("basic-functions-week-5.R")
```

3.2 JHU CSSE and global age pattern. Disaggregate Finnish total death counts of JHU CSSE into 10-year age groups using the global age pattern

To disaggregate the Finnish total death counts of the JHU CSSE into 10-year age groups (using the global age pattern), you need the basic function *disaggregate_deaths_one_coi_10y* of the demographic scaling model:

```
deaths_by_global_age <- disaggregate_deaths_one_coi_10y(coi="Finland")
```

```
## colSums(deaths_by_global_age)
```

If you like you could have a closer look at the data object *deaths_by_global_age* and perhaps check if the sum of the disaggregated death counts matches the total death counts of JHU CSSE.

3.3 COVerAGE-DB. Group Finnish death counts of COVerAGE-DB into 10-year age groups

Now you need to convert the COVID-19 death counts of the COVerAGE-DB from 5-year age groups into 10-year age groups:

```
## length(dat$Deaths[which(dat$Date=="15.09.2020")])  
## unique(dat$Age)  
## length(unique(dat$Age))  
## unique(dat$Age)[15]
```



```

#
## For one specific date (here: Sept 15, 2020):
#

dat_db <- c(0)
start <- seq(1,15,2)
end <- seq(2,16,2)
for(group in 1:8){
  dat_db[group] <- sum(dat$Deaths[which(dat$Date=="15.09.2020")][start[group]:end[group]])
}
dat_db[9] <- sum(dat$Deaths[which(dat$Date=="15.09.2020")][17:21])

sum(dat_db)

```

```
## [1] 338.9
```

```

#
## And over time:
#

dat_db_10y <- matrix(NA,nr=length(seq(0,80,10)),nc=length(unique(dat$Date)))
colnames(dat_db_10y) <- unique(dat$Date)
rownames (dat_db_10y) <- seq(0,80,10)

start <- seq(1,15,2)
end <- seq(2,16,2)
for(date in 1:length(unique(dat$Date))){
  current_date <- unique(dat$Date)[date]
  for(group in 1:8){
    dat_db_10y[group,date] <- sum(dat$Deaths[which(dat$Date==current_date)]
                                [start[group]:end[group]])
  } ## group
  dat_db_10y[9,date] <- sum(dat$Deaths[which(dat$Date==current_date)][17:21])
} ## date

## colSums(dat_db_10y)

```

Brief data description. The data object *dat_db_10y* contains COVID-19 deaths by 10-year age groups (rows) and single days (columns) from the COVERAGE-DB.

3.4 Bring data objects into the same date format

As a last step of data preparation, you need to bring the data objects that contain the death counts caused by COVID-19 into the *same* date format:

```

#
## Data based on COVERAGE-DB:
#

dat$Date <- as.Date(dat$Date,format="%d.%m.%Y")
head(unique(dat$Date))

## [1] "2020-04-06" "2020-04-09" "2020-04-20" "2020-04-22" "2020-04-24"
## [6] "2020-04-27"

```

```

tail(unique(dat$Date))

## [1] "2020-10-10" "2020-10-11" "2020-10-13" "2020-10-16" "2020-10-17"
## [6] "2020-10-18"

colnames(dat_db_10y)[1:ncol(dat_db_10y)] <- as.character(unique(dat$Date))
tail(colnames(dat_db_10y))

## [1] "2020-10-10" "2020-10-11" "2020-10-13" "2020-10-16" "2020-10-17"
## [6] "2020-10-18"

#
## Data based on JHU CSSE and global age pattern:
#

dates <- seq(as.Date("22/01/2020", format = "%d/%m/%Y"),
by = "days", length = (ncol(deaths)-4) )
head(dates)

## [1] "2020-01-22" "2020-01-23" "2020-01-24" "2020-01-25" "2020-01-26"
## [6] "2020-01-27"

tail(dates)

## [1] "2020-09-10" "2020-09-11" "2020-09-12" "2020-09-13" "2020-09-14"
## [6] "2020-09-15"

colnames(deaths)[5:ncol(deaths)] <- as.character(dates)
tail(colnames(deaths))

## [1] "2020-09-10" "2020-09-11" "2020-09-12" "2020-09-13" "2020-09-14"
## [6] "2020-09-15"

colnames(deaths_by_global_age)[1:ncol(deaths_by_global_age)] <- as.character(dates)
tail(colnames(deaths_by_global_age))

## [1] "2020-09-10" "2020-09-11" "2020-09-12" "2020-09-13" "2020-09-14"
## [6] "2020-09-15"

```

3.5. Visualize D and D_x estimates from different sources

How well do the Finnish data on COVID-19-related death counts from the COVerAGE-DB compare with the death counts from JHU CSSE?

To answer this question, you can compare the numbers of total and age-specific death counts for a specific date (here: September 15, 2020):

```

#
## Compare total death counts of the same date (here: September 15, 2020):
#

## deaths_by_global_age[, "2020-09-15"]
deaths_by_global_age[, "2020-09-15"][1:5]

##           0           10           20           30           40
## 0.002543368 0.053752682 0.266485690 0.717046646 1.542833210

deaths_by_global_age[, "2020-09-15"][6:9]

##           50           60           70           80

```

```
## 5.838341 22.486834 84.161560 223.930603
```

```
## dat_db_10y[, "2020-09-15"]  
dat_db_10y[, "2020-09-15"][1:5]
```

```
## 0 10 20 30 40  
## 0.0 0.0 0.4 1.4 4.2
```

```
dat_db_10y[, "2020-09-15"][6:9]
```

```
## 50 60 70 80  
## 11.1 26.2 67.6 228.0
```

```
sum(deaths_by_global_age[, "2020-09-15"])
```

```
## [1] 339
```

```
sum(dat_db_10y[, "2020-09-15"])
```

```
## [1] 338.9
```

Here you can see that the JHU CSSE and the COVERAGE-DB provide similar total COVID-19 related death counts, approximately 339, for Finland on September 15, 2020.

You can, of course, also plot the D estimates from the different sources over time:

```
today <- Sys.Date()  
dates_all <- seq(as.Date("22/01/2020", format = "%d/%m/%Y"), today, by = "days")  
head(dates_all)
```

```
## [1] "2020-01-22" "2020-01-23" "2020-01-24" "2020-01-25" "2020-01-26"
```

```
## [6] "2020-01-27"
```

```
tail(dates_all)
```

```
## [1] "2020-11-18" "2020-11-19" "2020-11-20" "2020-11-21" "2020-11-22"
```

```
## [6] "2020-11-23"
```

```
par(fig = c(0,1,0,1), las=1, mai=c(0.8,0.8,0.8,0.4))
```

```
plot(x=-100,y=-100,xlim=c(0,length(dates_all)),ylim=c(0,350),xlab="Calendar date",  
ylab="",main="COVID-19-related total death counts\n based on different sources",axes=FALSE)
```

```
## dates_all[which(dates_all %in% dates)]
```

```
xxx <- which(dates_all %in% dates)
```

```
## dates_all[xxx]
```

```
lines(x=xxx,y=colSums(deaths_by_global_age),lwd=1,col="black")
```

```
points(x=xxx,y=colSums(deaths_by_global_age),lwd=4,pch=19,col="black")
```

```
## dates_all[which(dates_all %in% unique(dat$Date))]
```

```
xx <- which(dates_all %in% unique(dat$Date))
```

```
## dates_all[xx]
```

```
lines(x=xx,y=colSums(dat_db_10y),lwd=1,col="blue")
```

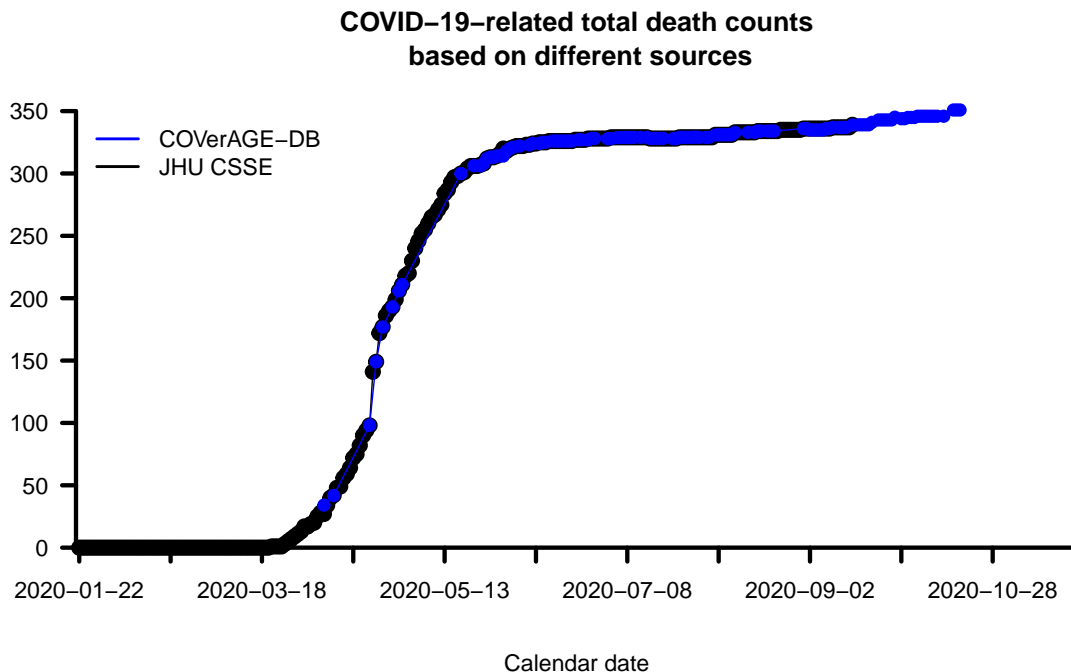
```
points(x=xx,y=colSums(dat_db_10y),lwd=2,pch=19,col="blue")
```

```
axis(side=1,at=c(seq(1,length(dates_all),28),length(dates_all)),
```

```
labels=dates_all[c(seq(1,length(dates_all),28),length(dates_all))],lwd=3,pos=0)
```

```
axis(side=2,at=seq(0,350,50),labels=TRUE,lwd=3,pos=0)
```

```
legend(x=0,y=350,c("COVerAGE-DB","JHU CSSE"),
col=c("blue","black"),lwd=2,bty="n")
```



Please describe the level and temporal development of the Finnish total death counts caused by COVID-19 from the COVerAGE-DB and the JHU CSSE. What do you think: how large will be the impact of using COVID-19-related death counts from these two different sources on the COVID-19 infection estimates of the demographic scaling model?

You can now also plot the D_x estimates from the different sources for a specific date:

```
par(fig = c(0,1,0,1), las=1, mai=c(0.8,0.8,1.2,0.4))

plot(x=-100,y=-100,xlim=c(-5,80),ylim=c(0,250),xlab="Chronological age",
ylab="",main="COVID-19 death counts by 10-year age groups\n September 15, 2020",axes=FALSE)

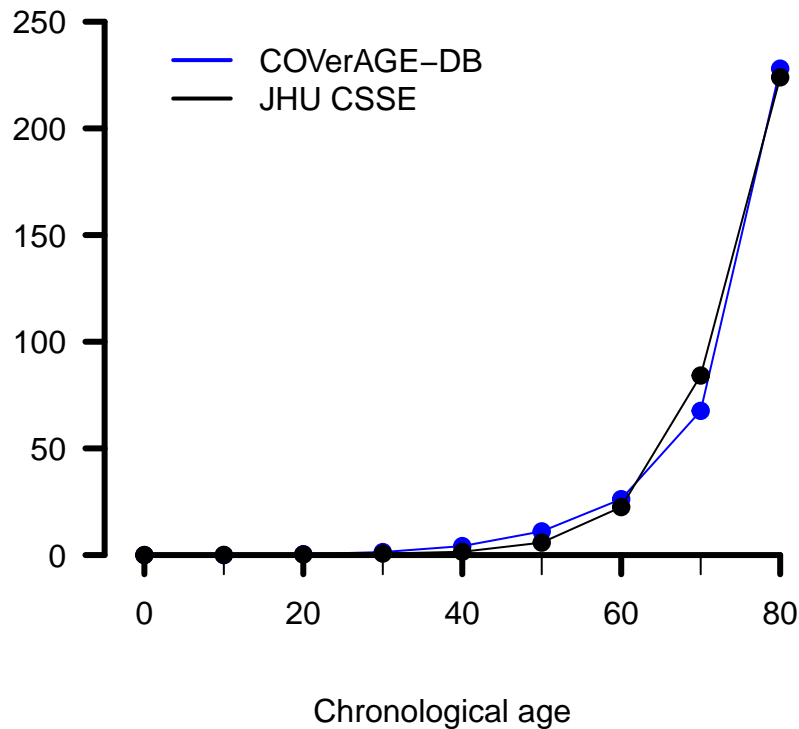
lines(x=seq(0,80,10),y=dat_db_10y[, "2020-09-15"],lwd=1,col="blue")
points(x=seq(0,80,10),y=dat_db_10y[, "2020-09-15"],lwd=2,pch=19,col="blue")

lines(x=seq(0,80,10),y=deaths_by_global_age[, "2020-09-15"],lwd=1,col="black")
points(x=seq(0,80,10),y=deaths_by_global_age[, "2020-09-15"],lwd=2,pch=19,col="black")

axis(side=1,at=seq(0,80,10),labels=FALSE,lwd=1,pos=0)
axis(side=1,at=seq(0,80,20),labels=TRUE,lwd=3,pos=0)
axis(side=2,at=seq(0,250,50),labels=TRUE,lwd=3,pos=-5)

legend(x=0,y=250,c("COVerAGE-DB","JHU CSSE"),
col=c("blue","black"),lwd=2,bty="n")
```

COVID-19 death counts by 10-year age groups September 15, 2020



Please describe and compare the level of the age-specific COVID-19 death counts as of September 15, 2020, based on data of the COVerAGE-DB and the JHU CSSE. Would you expect similar results for other countries and other points of time?

4. Estimate total numbers of COVID-19 infections in Finland based on different D_x estimates

To estimate the total numbers of COVID-19 infections in Finland based on different D_x estimates you can adopt the demographic scaling model.

4.1 Scale original IFR_x estimates from a reference country onto a country of interest

For the sake of convenience (and simplicity), you can just scale the original IFR_x estimates of Verity et al. (2020) based on remaining life expectancy in order to get IFR_x estimates that better match the Finnish context with respect to age structure, preconditions, and medical services.

```
#
## 1. Ungroup original IFR estimates:
#

ungrouped_mode_ifr_by_single_age_china_sp <- to_ungroup(to_ungroup=
  ifr_by_age_china_verity[,2],nr_grouped_years=10)

#
```

```

## 2. Scale original IFRs onto a COI (here: Finland) via remaining life expectancy:
#

mapped_mode_ifr_thanatAge_verity <- map_ifr_betw_ref_and_one_coi_thanatAge(coi="Finland",
  lt_1950_2020=lt_1950_2020,
  ungrouped_ifr_by_single_age_china_sp=ungrouped_mode_ifr_by_single_age_china_sp)

## and fill in the few NA values:

pos_na <- which(is.na(mapped_mode_ifr_thanatAge_verity[1,]))
if(length(pos_na)>0){
  for(pos in 1:length(pos_na)){
    if(pos_na[pos] < 6){
      mapped_mode_ifr_thanatAge_verity[1,pos_na[pos]] <-
        min(mapped_mode_ifr_thanatAge_verity[1,],na.rm=TRUE)
    }
    if(pos_na[pos] >= 6){
      mapped_mode_ifr_thanatAge_verity[1,pos_na[pos]] <-
        mapped_mode_ifr_thanatAge_verity[1,pos_na[pos]-1]
    }
  } ## for pos
} ## if

#
## 3. Put scaled IFRs into 10-year age groups:
#

mapped_mode_ifr_thanatAge_verity_10y <- aggregate_mapped_ifr_10y(disaggregated_mapped_ifr=
  mapped_mode_ifr_thanatAge_verity)

```

4.2 Estimate total numbers of COVID-19 infections in Finland over time

You can now estimate the total numbers of COVID-19 infections in Finland over time based on the scaled IFR_x estimates and the different D_x estimates.

```

#
## Based on data of COVerAGE-Db:
#

inf_mode_scaled_verity_COVerAGE <- colSums( dat_db_10y /
  mapped_mode_ifr_thanatAge_verity_10y )

#
## Based on data of JHU CSSE:
#

inf_mode_scaled_verity_JHU <- colSums( deaths_by_global_age /
  mapped_mode_ifr_thanatAge_verity_10y )

```

The data objects `inf_mode_scaled_verity_COVerAGE` and `inf_mode_scaled_verity_JHU` contain the total numbers of COVID-19 infection estimates for Finland based on age-specific death counts of the COVerAGE-DB and the JHU CSSE (& the global age pattern), respectively.

You can now visualize the total numbers of COVID-19 infections in Finland over time in order to analyze the

impact of different D_x estimates. When comparing these Finnish COVID-19 infection estimates with the numbers of confirmed cases, it is a good idea to also account for an average time to death of approximately 18 days.

```
par(fig = c(0,1,0,1), las=1, mai=c(0.4,0.8,0.8,0.4))

plot(x=-100,y=-100,xlim=c(0,length(dates_all)),ylim=c(0,30),xlab="Date",ylab="",
     cex.main=0.9,main="Total numbers of COVID-19 infections, in thousand, in Finland",
     axes=FALSE)

## dates_all[which(dates_all %in% dates)]
xxx <- which(dates_all %in% dates)
## dates_all[xxx]

lines(x=xxx-18,y=inf_mode_scaled_verity_JHU/1000,lwd=1,col="black")
points(x=xxx-18,y=inf_mode_scaled_verity_JHU/1000,lwd=2,pch=19,col="black")

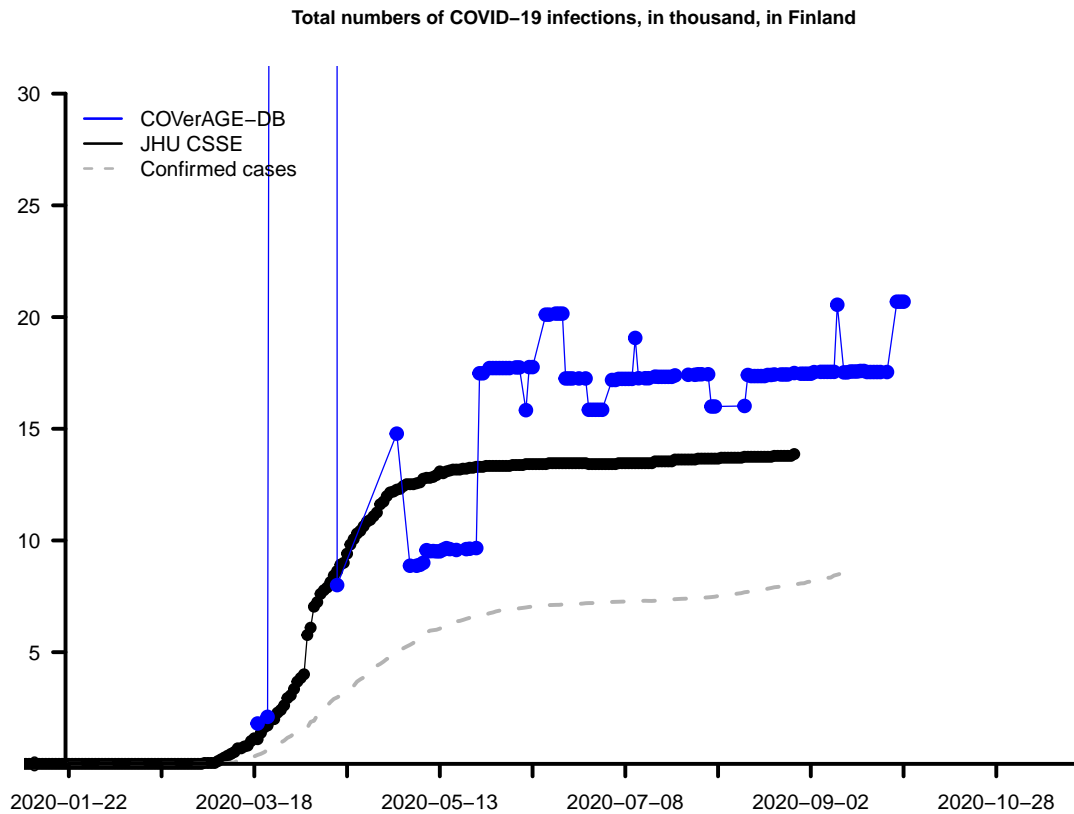
lines(x=xxx,y=confirmed[which(confirmed[, "Country.Region"]=="Finland"),
5:ncol(confirmed)]/1000,col=gray(0.7),lty=2,lwd=3)

## dates_all[which(dates_all %in% unique(dat$Date))]
xx <- which(dates_all %in% unique(dat$Date))
## dates_all[xx]

lines(x=xx-18,y=inf_mode_scaled_verity_COVerAGE/1000,lwd=1,col="blue")
points(x=xx-18,y=inf_mode_scaled_verity_COVerAGE/1000,lwd=4,pch=19,col="blue")

axis(side=1,at=c(seq(1,length(dates_all),28),length(dates_all)),
     labels=dates_all[c(seq(1,length(dates_all),28),length(dates_all))],lwd=3,pos=0)
axis(side=2,at=seq(0,30,5),labels=TRUE,lwd=3,pos=0)

legend(x=0,y=30,c("COVerAGE-DB","JHU CSSE","Confirmed cases"),
     col=c("blue","black",gray(0.7)),lty=c(1,1,2),lwd=2,bty="n")
```



Please describe and compare the level and the temporal development of these estimated total numbers of COVID-19 infections in Finland. Please also compare them to (1) the numbers of confirmed cases in Finland and, perhaps, to (2) the corresponding figures in other countries.

Please think about how you could extend this R-code in order to estimate the total numbers of COVID-19 infections for other countries and how to provide uncertainty estimates for them.

4. Time for you to think both creatively and critically about the sensitivity of the COVID-19 infection estimates with respect to the age-specific death counts. How sensitive are they? Is one result perhaps more plausible than another result?

You could perhaps also try to analyze the combined impact of IFR_x and D_x on estimates of the total numbers of COVID-19 infections. Has one input parameter a stronger impact on the results than the other input parameter? Which combinations of data for the two input parameters are perhaps more plausible than others?

What does the sensitivity analysis tell you about the results of the demographic scaling model?

What other kinds of analyses would you do in order to assess the demographic scaling model's results?