Анализ на починалите

Кеворкян К., Райчева Р.Костадинов К.

24/03/2022

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5 v purrr 0.3.4  
## v tibble 3.1.6 v dplyr 1.0.8  
## v tidyr 1.2.0 v stringr 1.4.0  
## v readr 2.1.2 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(easystats)

## # Attaching packages: easystats 0.4.3 (red = needs update)  
## x insight 0.16.0 x datawizard 0.3.0   
## v bayestestR 0.11.5.1 x performance 0.8.0.1   
## v parameters 0.17.0 v effectsize 0.6.0.1   
## x modelbased 0.7.2 v correlation 0.8.0   
## v see 0.7.0.1 v report 0.5.1   
##   
## Restart the R-Session and update packages in red with 'easystats::easystats\_update()'.

library(tayloRswift)  
library(readr)  
library(flextable)

##   
## Attaching package: 'flextable'

## The following object is masked from 'package:purrr':  
##   
## compose

library(janitor)

##   
## Attaching package: 'janitor'

## The following objects are masked from 'package:datawizard':  
##   
## remove\_empty, remove\_empty\_rows

## The following object is masked from 'package:insight':  
##   
## clean\_names

## The following objects are masked from 'package:stats':  
##   
## chisq.test, fisher.test

library(apyramid)  
Sys.setlocale(locale = "Bulgarian")

## [1] "LC\_COLLATE=Bulgarian\_Bulgaria.1251;LC\_CTYPE=Bulgarian\_Bulgaria.1251;LC\_MONETARY=Bulgarian\_Bulgaria.1251;LC\_NUMERIC=C;LC\_TIME=Bulgarian\_Bulgaria.1251"

cov = read\_csv("C:\\\\Users\\PC\\Desktop\\covid\_full.csv",   
 col\_types = cols(exam\_date = col\_date(format = "%d.%m.%Y"),   
 lastvac\_date = col\_date(format = "%d.%m.%Y"),   
 start\_hospis = col\_date(format = "%d.%m.%Y"),   
 end\_hospis = col\_date(format = "%d.%m.%Y")))  
  
cov = as\_tibble(cov)

dth <- read\_csv("C:\\\\Users\\PC\\Desktop\\deaths.csv", col\_types = cols(add\_day = col\_date(format = "%Y-%m-%d"),   
 death\_day = col\_date(format = "%Y-%m-%d")))  
dth = as\_tibble(dth)

# Обща характеристика на всички инфектирани в страната, независимо от изхода на инфекцията:

## общ брой

В представената база данни са налични 1 126 945 записа на официално потвърдени и регистрирани случаи с коронавирусна инфекция.

## медиана на възрастта на цялата извадка.

age = cov %>%   
 dplyr::summarise(   
 count = n(),  
 median = median(age, na.rm = TRUE),  
 IQR = IQR(age, na.rm = TRUE),  
 mean = mean(age, na.rm = TRUE)) %>%   
flextable()  
age

| count | median | IQR | mean |
| --- | --- | --- | --- |
| 1,126,945 | 48 | 29 | 47.92146 |

age\_dth = dth %>%   
 dplyr::summarise(   
 count = n(),  
 median = median(age, na.rm = TRUE),  
 IQR = IQR(age, na.rm = TRUE),  
 mean = mean(age, na.rm = TRUE)) %>%   
flextable()  
age\_dth

| count | median | IQR | mean |
| --- | --- | --- | --- |
| 36,192 | 73 | 15 | 71.52105 |

## - разпределение по пол (n, %)

gender = cov %>%   
 tabyl(gender) %>% # tabulate counts and proportions by age category  
 adorn\_pct\_formatting() %>%   
 flextable()  
gender

| gender | n | percent |
| --- | --- | --- |
| F | 605,835 | 53.8% |
| M | 521,110 | 46.2% |

gender\_dth = dth %>%   
 tabyl(sex) %>% # tabulate counts and proportions by age category  
 adorn\_pct\_formatting() %>%   
 flextable()  
gender\_dth

| sex | n | percent |
| --- | --- | --- |
| жена | 16,204 | 44.8% |
| мъж | 19,988 | 55.2% |

## по възрастови групи (n, %)

cov$age = as.numeric(cov$age)  
  
cov["age\_group"] = cut(cov$age, c(0, 14, 24,34, 44, 54, 64, 74, 84, 94, Inf),   
 c("0-14","15-24","25-34","35-44","45-54","55-64",  
 "65-74","75-84","85-94","> 95"), include.lowest=TRUE)  
  
age\_groups = cov %>%   
 tabyl(age\_group) %>% # tabulate counts and proportions by age category  
 adorn\_pct\_formatting() %>%   
 flextable()  
age\_groups

| age\_group | n | percent |
| --- | --- | --- |
| 0-14 | 56,952 | 5.1% |
| 15-24 | 81,134 | 7.2% |
| 25-34 | 150,247 | 13.3% |
| 35-44 | 195,549 | 17.4% |
| 45-54 | 206,087 | 18.3% |
| 55-64 | 190,228 | 16.9% |
| 65-74 | 151,474 | 13.4% |
| 75-84 | 77,103 | 6.8% |
| 85-94 | 17,698 | 1.6% |
| > 95 | 473 | 0.0% |

dth$age = as.numeric(dth$age)  
  
dth["age\_group"] = cut(dth$age, c(0, 14, 24,34, 44, 54, 64, 74, 84, 94, Inf),   
 c("0-14","15-24","25-34","35-44","45-54","55-64",  
 "65-74","75-84","85-94","> 95"), include.lowest=TRUE)  
  
age\_groups\_dth = dth %>%   
 tabyl(age\_group) %>% # tabulate counts and proportions by age category  
 adorn\_pct\_formatting() %>%   
 flextable()  
age\_groups\_dth

| age\_group | n | percent |
| --- | --- | --- |
| 0-14 | 15 | 0.0% |
| 15-24 | 38 | 0.1% |
| 25-34 | 198 | 0.5% |
| 35-44 | 724 | 2.0% |
| 45-54 | 1,991 | 5.5% |
| 55-64 | 5,464 | 15.1% |
| 65-74 | 12,058 | 33.3% |
| 75-84 | 11,567 | 32.0% |
| 85-94 | 4,020 | 11.1% |
| > 95 | 117 | 0.3% |

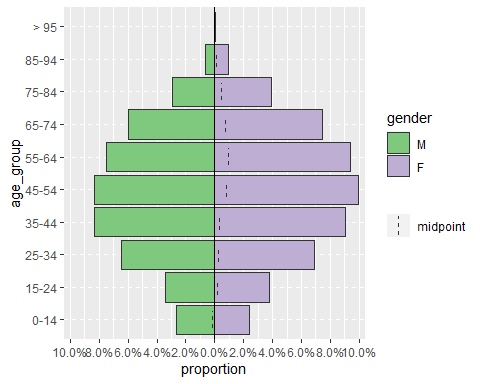
# по възраст и пол

age\_gender = cov %>% # case linelist  
 tabyl(age\_group, gender) %>% # cross-tabulate counts  
 adorn\_totals(where = "row") %>% # add a total row  
 adorn\_percentages(denominator = "col") %>% # convert to proportions  
 adorn\_pct\_formatting() %>% # convert to percents  
 adorn\_ns(position = "front") %>% # display as: "count (percent)"  
 flextable()

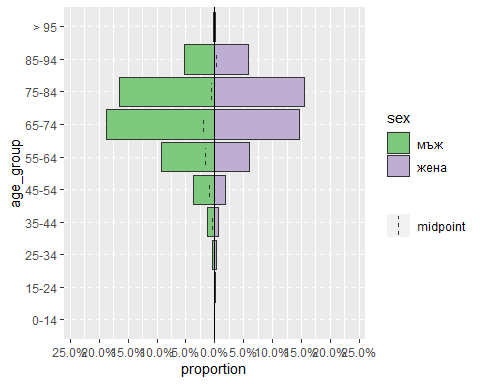
age\_gender\_dth = dth %>% # case linelist  
 tabyl(age\_group, sex) %>% # cross-tabulate counts  
 adorn\_totals(where = "row") %>% # add a total row  
 adorn\_percentages(denominator = "col") %>% # convert to proportions  
 adorn\_pct\_formatting() %>% # convert to percents  
 adorn\_ns(position = "front") %>% # display as: "count (percent)"  
 flextable()  
age\_gender\_dth

| age\_group | жена | мъж |
| --- | --- | --- |
| 0-14 | 7 (0.0%) | 8 (0.0%) |
| 15-24 | 18 (0.1%) | 20 (0.1%) |
| 25-34 | 78 (0.5%) | 120 (0.6%) |
| 35-44 | 244 (1.5%) | 480 (2.4%) |
| 45-54 | 651 (4.0%) | 1340 (6.7%) |
| 55-64 | 2142 (13.2%) | 3322 (16.6%) |
| 65-74 | 5292 (32.7%) | 6766 (33.9%) |
| 75-84 | 5605 (34.6%) | 5962 (29.8%) |
| 85-94 | 2107 (13.0%) | 1913 (9.6%) |
| > 95 | 60 (0.4%) | 57 (0.3%) |
| Total | 16204 (100.0%) | 19988 (100.0%) |

piramid = apyramid::age\_pyramid(data = cov,  
 age\_group = "age\_group",  
 split\_by = "gender",  
 proportional = TRUE)  
piramid



piramid\_dth = apyramid::age\_pyramid(data = dth,  
 age\_group = "age\_group",  
 split\_by = "sex",  
 proportional = TRUE)  
piramid\_dth



vac\_piramid=apyramid::age\_pyramid(data = cov,  
 age\_group = "age\_group",  
 split\_by = "is\_f\_vac")

province = cov %>%   
 tabyl(province) %>% # tabulate counts and proportions by age category  
 adorn\_pct\_formatting() %>%   
 flextable()  
province

| province | n | percent | valid\_percent |
| --- | --- | --- | --- |
| Благоевград | 49,765 | 4.4% | 4.4% |
| Бургас | 73,863 | 6.6% | 6.6% |
| Варна | 94,464 | 8.4% | 8.4% |
| Велико Търново | 27,521 | 2.4% | 2.4% |
| Видин | 10,643 | 0.9% | 0.9% |
| Враца | 26,039 | 2.3% | 2.3% |
| Габрово | 18,544 | 1.6% | 1.6% |
| Добрич | 21,052 | 1.9% | 1.9% |
| Кърджали | 9,783 | 0.9% | 0.9% |
| Кюстендил | 20,419 | 1.8% | 1.8% |
| Ловеч | 16,353 | 1.5% | 1.5% |
| Монтана | 19,091 | 1.7% | 1.7% |
| Пазарджик | 29,373 | 2.6% | 2.6% |
| Перник | 22,240 | 2.0% | 2.0% |
| Плевен | 32,408 | 2.9% | 2.9% |
| Пловдив | 101,453 | 9.0% | 9.0% |
| Разград | 10,692 | 0.9% | 0.9% |
| Русе | 33,404 | 3.0% | 3.0% |
| Силистра | 14,618 | 1.3% | 1.3% |
| Сливен | 26,187 | 2.3% | 2.3% |
| Смолян | 12,511 | 1.1% | 1.1% |
| София | 40,013 | 3.6% | 3.6% |
| София (столица) | 286,377 | 25.4% | 25.4% |
| Стара Загора | 51,605 | 4.6% | 4.6% |
| Търговище | 12,060 | 1.1% | 1.1% |
| Хасково | 27,151 | 2.4% | 2.4% |
| Шумен | 22,180 | 2.0% | 2.0% |
| Ямбол | 17,133 | 1.5% | 1.5% |
|  | 3 | 0.0% | - |

province\_dth = dth %>%   
 tabyl(nuts2) %>% # tabulate counts and proportions by age category  
 adorn\_pct\_formatting() %>%   
 flextable()  
province\_dth

| nuts2 | n | percent |
| --- | --- | --- |
| Благоевград | 1,491 | 4.1% |
| Бургас | 1,687 | 4.7% |
| Варна | 2,415 | 6.7% |
| Велико Търново | 1,385 | 3.8% |
| Видин | 560 | 1.5% |
| Враца | 814 | 2.2% |
| Габрово | 734 | 2.0% |
| Добрич | 921 | 2.5% |
| Кърджали | 491 | 1.4% |
| Кюстендил | 869 | 2.4% |
| Ловеч | 714 | 2.0% |
| Монтана | 873 | 2.4% |
| Пазарджик | 1,606 | 4.4% |
| Перник | 650 | 1.8% |
| Плевен | 1,307 | 3.6% |
| Пловдив | 3,354 | 9.3% |
| Разград | 653 | 1.8% |
| Русе | 1,759 | 4.9% |
| Силистра | 594 | 1.6% |
| Сливен | 1,103 | 3.0% |
| Смолян | 680 | 1.9% |
| София | 1,148 | 3.2% |
| София (столица) | 4,629 | 12.8% |
| Стара Загора | 2,009 | 5.6% |
| Търговище | 654 | 1.8% |
| Хасково | 1,395 | 3.9% |
| Шумен | 1,052 | 2.9% |
| Ямбол | 645 | 1.8% |

pop\_data <- cov %>%   
 group\_by(gender, age\_group) %>%   
 summarise(counts=n()) %>%   
 mutate(  
 percent = round(100\*(counts / sum(counts, na.rm=T)),1), # % of total  
 percent = case\_when(   
 gender == "F" ~ percent,  
 gender == "M" ~ -percent, # if male, convert % to negative  
 TRUE ~ NA\_real\_))

## `summarise()` has grouped output by 'gender'. You can override using the  
## `.groups` argument.

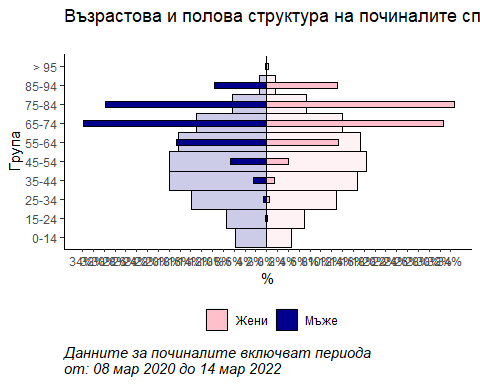
case\_data <- dth %>%   
 group\_by(sex, age\_group) %>%   
 mutate(sex = recode(sex, "жена" = 'F', "мъж" = 'M')) %>%   
 summarise(counts=n()) %>%   
 mutate(  
 percent = round(100\*(counts / sum(counts, na.rm=T)),1), # % of total  
 percent = case\_when(   
 sex == "F" ~ percent,  
 sex == "M" ~ -percent, # if male, convert % to negative  
 TRUE ~ NA\_real\_)) %>%   
 rename(gender = sex)

## `summarise()` has grouped output by 'sex'. You can override using the `.groups`  
## argument.

# combine case and population data (same column names, age\_cat values, and gender values)  
pyramid\_data <- bind\_rows("cases" = case\_data, "population" = pop\_data, .id = "data\_source")  
# Define extent of percent axis, used for plot limits  
max\_per <- max(pyramid\_data$percent, na.rm=T)  
min\_per <- min(pyramid\_data$percent, na.rm=T)

age\_levels <- c("0-14","15-24","25-34","35-44","45-54","55-64",  
 "65-74","75-84","85-94","> 95")

# begin ggplot  
##############  
ggplot()+ # default x-axis is age in years;  
  
 # population data graph  
 geom\_col(  
 data = pyramid\_data %>% filter(data\_source == "population"),  
 mapping = aes(  
 x = age\_group,  
 y = percent,  
 fill = gender),  
 colour = "black", # black color around bars  
 alpha = 0.2, # more transparent  
 width = 1)+ # full width  
   
 # case data graph  
 geom\_col(  
 data = pyramid\_data %>% filter(data\_source == "cases"),   
 mapping = aes(  
 x = age\_group, # age categories as original X axis  
 y = percent, # % as original Y-axis  
 fill = gender), # fill of bars by gender  
 colour = "black", # black color around bars  
 alpha = 1, # not transparent   
 width = 0.3)+ # half width  
   
 # flip the X and Y axes to make pyramid vertical  
 coord\_flip()+  
   
 # manually ensure that age-axis is ordered correctly  
 scale\_x\_discrete(limits = age\_levels)+ # defined in chunk above  
   
 # set percent-axis   
 scale\_y\_continuous(  
 limits = c(min\_per, max\_per), # min and max defined above  
 breaks = seq(floor(min\_per), ceiling(max\_per), by = 2), # from min% to max% by 2   
 labels = paste0( # for the labels, paste together...   
 abs(seq(floor(min\_per), ceiling(max\_per), by = 2)), "%"))+   
  
 # designate colors and legend labels manually  
 scale\_fill\_manual(  
 values = c("F" = "pink", # assign colors to values in the data  
 "M" = "darkblue"),  
 labels = c("F" = "Жени",  
 "M"= "Мъже"), # change labels that appear in legend, note order  
 ) +  
  
 # plot labels, titles, caption   
 labs(  
 title = "Възрастова и полова структура на починалите спрямо инфектираните",  
 subtitle = "",  
 x = "Група",  
 y = "%",  
 fill = NULL,  
 caption = stringr::str\_glue("Данните за починалите включват периода \nот: {format(min(dth$add\_day, na.rm=T), '%d %b %Y')} до {format(max(dth$add\_day, na.rm=T), '%d %b %Y')}")) +  
   
 # optional aesthetic themes  
 theme(  
 legend.position = "bottom", # move legend to bottom  
 panel.grid.major = element\_blank(),  
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),  
 axis.line = element\_line(colour = "black"),  
 plot.title = element\_text(hjust = 0),   
 plot.caption = element\_text(hjust=0, size=11, face = "italic"))



# разпределение спрямо място на изолация/лечение (на домашно лечение, в ковид отделение, в интензивно отдление) : todo

# ваксинационен статус в групата на оставените на домашно лечение, в ковид отделение, в интензивно отдление: todo

# изход (оздравели или починали с/от ковид) в болница(ковид отделенеи или интензивно отделение) или в дома todo