

# Projeto final

Tarla Lemos de Andrade

30/06/2020

## Introdução

Com o cenário atual de crise sanitária causado pela pandemia do novo corona vírus, o excesso de informação e o número de óbitos diários deixaram de causar impacto. Por isso é de grande importância a construção de representações visuais que permitam com que as pessoas tomem consciência da realidade e continuem praticando as ações preventivas. Com este objetivo, neste trabalho serão desenvolvidos visualizações que permitam a análise dos casos do Brasil.

## Visualização 1: Taxa de letalidade dos estados ao longo do tempo

A proposta era fazer uma visualização que permitisse analisar e comparar a gravidade da doença entre os estados. Para quantificar a gravidade escolhi a taxa de letalidade, que não só carrega esta informação, mas também o contexto político e social do local. Esta taxa é calculada da seguinte forma:

$$\text{Taxa de letalidade} = \frac{\text{Quantidade de mortes}}{\text{Casos confirmados}}$$

Para representar as três informações de interesse (tempo, estados e a taxa de mortalidade), optei por um Heatmap.

**Dados** Os dados foram coletados no site <https://brasil.io/home/>, que coleta os dados das Secretarias Estaduais de Saúde diariamente. O script de extração pode ser conferido no github <https://github.com/turicas/brasil.io>.

```
library(readr)
library(dplyr)
library(ggplot2)
library(lubridate)
library(tidyr)

covid_br <- read_csv("caso_full129-06.csv.gz") %>%
  dplyr::filter(!is.na(city)) %>%
  dplyr::filter(date > '2020-04-02') %>%
  dplyr::group_by(state, date) %>%
  dplyr::summarise(taxaDeMortes = sum(last_available_death_rate))

legenda <- c('0 a 5', '5.1 a 10', '11 a 15', '16 a 20',
             '21 a 25', '26 a 30', '31 a 35', 'mais de 36')
intervalos <- c(-Inf, 5, 10, 15, 20, 25, 30, 35, +Inf)
covid_br$tipo <- cut(covid_br$taxaDeMortes, breaks=intervalos, labels=legenda)
```

```

paletinha <- colorRampPalette(c('#3dfe22','#030d02'))

png("Covid_heatmap.png",width=1000, height=600)
ggplot(covid_br, aes(x=date,
                     y=state))+
  geom_tile(aes(fill=tipo), color = "#d3d3d3", size = .5)+
  scale_x_date(date_breaks = "1.5 week", date_labels = '%d-%b')+
  coord_cartesian(expand = FALSE)+
  theme_grey(base_size = 10) +
  theme(text = element_text(size = 12),
        legend.position = "bottom",
        legend.title = element_text(size = 10),
        legend.text.align = unit(.6, "npc"),
        legend.key.size = unit(.03, "npc"),
        legend.key.height = unit(.04, "npc"),
        axis.text.y = element_text(size=8))+
  scale_fill_manual(values = RColorBrewer::brewer.pal(9, "Reds"))+
  labs(fill="Taxa de letalidade (%)",
       x=NULL,
       y=NULL,
       title = 'Taxa de letalidade (mortes / confirmados) por estado',
       subtitle = paste0("Do dia 02/04/2020 ao dia 29/06/2020"),
       caption = 'Fonte: Secretarias de Saúde das Unidades Federativas,
                 dados tratados por Álvaro Justen e colaboradores/Brasil.IO.')

dev.off()

```

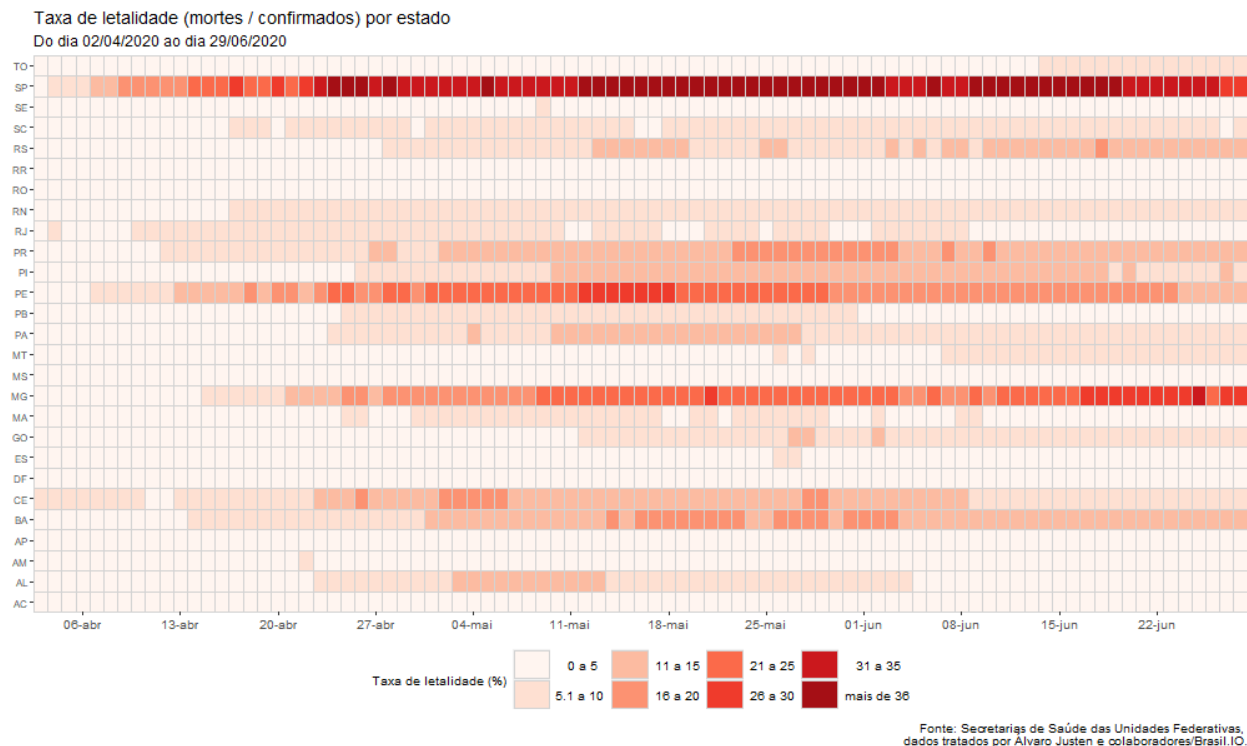


Figure 1: Heatmap

## Visualização 2: Evolução do Covid-19 no Brasil

A ideia era construir uma visualização que mostrasse o avanço da contaminação pelos municípios e estados. Assim, como os dados geográficos eram importantes para passar esta informação, optei por criar um mapa iterativo.

O gif do mapa pode ser acessado pelo link: <https://photos.app.goo.gl/4uTaUp4bdjeg2qEs5>

### Dados

A base de dados é a mesma da visualização anterior.

Obs: Os casos importados não estão representados no mapa.

```
library(readr)
library(dplyr)
library(ggplot2)
library(ggspatial)
library(geobr)
library(lubridate)
library(animation)
library(RColorBrewer)

covid_dados <- read_csv("caso_full29-06.csv.gz") %>%
  dplyr::filter(!is.na(city)) %>%
  dplyr::filter(!is.na(city_ibge_code)) %>%
  dplyr::select(date,state,city,city_ibge_code,last_available_confirmed)
colnames(covid_dados) <- c('data', 'estado', 'cidade',
                           'code_muni', 'casosAcumulados' )

datas <- unique(covid_dados$data)
inicio_data <- min(datas)
datas_seq <- inicio_data + days(0:(length(datas)-1))

#dados do mapa e código dos municípios
estados <- read_state(year=2018)
municipios <- read_municipality(year=2018) #>%

#Dados para a legenda
legenda <- c('1 a 100', '101 a 250', '251 a 500', '501 a 750 mil',
             '751 a 1 mil', '1001 a 5 mil', '5001 a 10 mil',
             '10001 a 50 mil', 'mais de 50 mil')
intervalos <- c(1,100,250,500,750,1000,5000,10000,50000,+Inf)
covid_dados$situation <- cut(covid_dados$casosAcumulados,
                             breaks=intervalos,
                             labels=legenda)

#função para interpolar cores
paletinha <- colorRampPalette(c('#3dfe22', '#030d02'))

saveGIF(ani.height=700,ani.width=700,
        ani.res=120,interval=.10,{
  for(i in 1:length(datas)){

    #Selecionando os dados em função da data
    data_da_vez <- datas_seq[i]
    dados_data_da_vez <- dplyr::filter(covid_dados,data==data_da_vez)
    data_da_vez_ok <- filter(dados_data_da_vez,
```

```

                                dados_data_da_vez$code_muni %in%
                                municipios$code_muni)

#Juntando os dados do mapa e do dataset
juntos <- inner_join(municipios,dados_data_da_vez,by='code_muni')

plot <- ggplot(juntos)+
  geom_sf(aes(fill=situation, color=situation),cex=.1)+
  geom_sf(data=municipios,
          color=alpha("gray80",0.2),
          fill="transparent",cex=.1)+
  geom_sf(data=estados,color="gray40",
          fill="transparent",cex=.5)+
  scale_fill_manual(values = paletinha(9),
                    limits = legenda)+
  scale_color_manual(values = paletinha(9),
                    limits = legenda)+

  theme_bw()+
  guides(color=FALSE)+
  labs(fill="Número\nde\ncasos\nAcumulados",
       x=NULL,
       y=NULL,
       title="Evolução de casos de Covid-19 no Brasil",
       subtitle = paste0("Total de Casos em ",
                        day(data_da_vez),"/",
                        month(data_da_vez),"/",
                        year(data_da_vez)),
       caption ='Fonte: Secretarias de Saúde das Unidades Federativas, dados
       tratados por Álvaro Justen e colaboradores/Brasil.IO.')+
  theme(panel.grid = element_blank(),
        panel.border = element_blank(),
        panel.background = element_rect(fill = '#d8e4d3'),
        axis.text = element_blank(),
        axis.ticks = element_blank(),
        legend.position = c(0.88,.21),
        legend.key.size = unit(.3,'line'),
        legend.text = element_text(size = 8),
        legend.title = element_text(size = 9))

  print(plot)
}
})

```

### Visualização 3: Taxa de mortalidade por Covid-19

Para avaliar a situação do Brasil com relação à contenção da doença e tratamento, propus uma visualização que permitisse comparar a taxa de mortalidade (que representa a informação de interesse) com outros países.

Para isso, construí um treemap.

**Dados** A base de dados foi coletada no site <https://ourworldindata.org/coronavirus-source-data>, que coleta dados diários do site oficial do Centro Europeu de Prevenção e controle de doenças (<https://www.ecdc.europa.eu/en>). O script pode ser acessado em <https://github.com/owid>.

```

library(readr)
library(dplyr)
library(treemap)

covid_m <- read_csv("owid-covid-data29-06.csv") %>%
  dplyr::filter(!is.na(continent)) %>%
  mutate(deaths_per_inhabitant = total_deaths / population)

covid_mundo <- covid_m[covid_m$date=='2020-06-18',]

png(filename="tree.png",width=1500, height=800)

treemap(covid_mundo,
  index=c("continent","location"),
  vSize="total_deaths_per_million",
  type="index",
  fontsize.labels=c(25,20),
  fontcolor.labels=c("#5f4f3a","white"),
  fontface.labels=c("bold","bold"),
  bg.labels=c("transparent"),
  align.labels=list(
    c("left", "top"),
    c("right", "bottom")),
  overlap.labels=0.5,
  inflate.labels=F,
  border.col='white',
  palette = c("#F1948A","#BB8FCE","#85C1E9","#82E0AA","#F7DC6F","#E59866"),
  title = "Taxa de mortalidade por Covid-19 no dia 28/06/2020 ",
  fontsize.title = 20
)

dev.off()

```

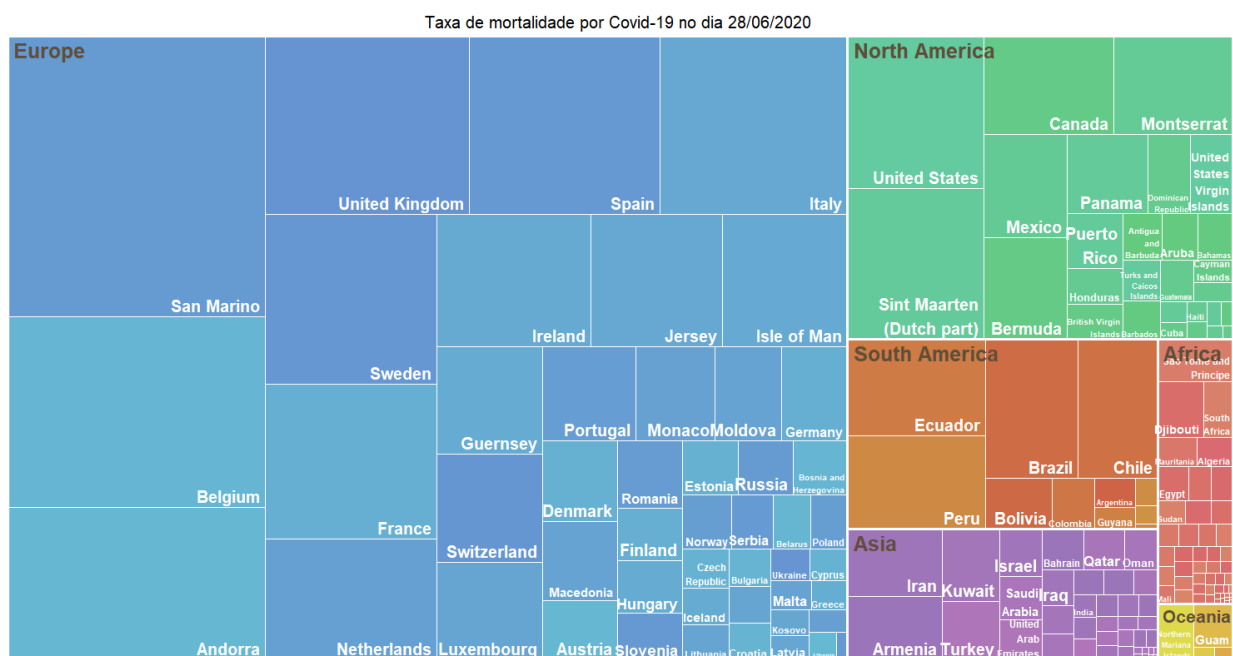


Figure 2: Treemap