Introdução ao uso de dados geoespaciais no R

9 Visualização de dados geoespaciais

Maurício H. Vancine
Milton C. Ribeiro
UNESP - Rio Claro
Laboratório de Ecologia Espacial e Conservação (LEEC)
25/10/2021-05/11/2021

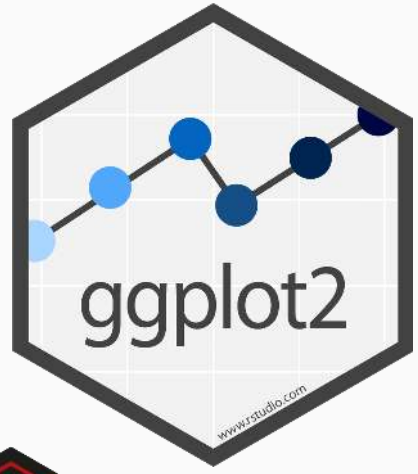

Title
Mexico - Major Habitat Types
Inset map
Main map
Gulf of Mexico
Pacific Ocean
Orientation indicator
Scale indicator
Source note
Graphics
Legend
Major Habitat Types
Tropical and Subtropical Moist Broadleaf Forests
Tropical and Subtropical Dry Broadleaf Forests
Tropical and Subtropical Coniferous Forests
Temperate Conifer Forests
Tropical and Subtropical Grasslands, Savannas and Shrublands
Flooded Grasslands and Savannas
Montane Grasslands and Shrublands
Mediterranean Forests, Woodlands, and Scrubs
Deserts and Xeric Shrublands
Mangroves
Map created January, 2015
ESRI Virtual Campus

Gimond (2021)

9 Visualização de dados geoespaciais

Conteúdo

1. Elementos de um mapa
2. Pacotes para produção de mapas
3. Mapas estáticos
4. Mapas animados
5. Mapas interativos



ggplot2, tmap, mapsf, leaflet, mapview

9 Visualização de dados geoespaciais

Script

`09_script_intro_geoespacial_r.R`

1. Elementos de um mapa

Elementos

- Mapa principal
- Mapa secundário
- Título
- Legenda
- Barra de escala
- Indicador de orientação (Norte)
- Gride de coordenadas
- Descrição do Sistema de Referência de Coordenadas
- Informações de origem
- Elementos auxiliares

Gimond (2021)

2. Pacotes para produção de mapas

Principais pacotes

- ggplot2: cria mapas elegantes usando a Gramática de Gráficos
- ggspatial: estrutura de dados espaciais para ggplot2
- ggmap: visualização espacial com ggplot2
- tmap: cria mapas temáticos
- mapsf: cria mapas temáticos com dados vetoriais
- linemap: cria mapas de linhas
- cartography: cartografia temática
- cartogram: métodos de visualização para cartogramas
- googleway: acessa Google Maps APIs para recuperar dados e traçar mapas
- rasterVis: métodos de visualização para dados raster
- mapview: visualização interativa de dados espaciais em R
- leaflet: cria Web Maps interativos com Leaflet
- plotly: crie gráficos interativos da Web por meio de 'plotly.js'
- mapedit: edição interativa de dados geoespaciais em R

3. Mapas estáticos

Mapas estáticos são mapas simples e fixos para visualização de dados

Principais

1. `plot()` (vários pacotes)
2. ggplot2
3. ggmap
4. tmap
5. mapsf
6. linemap
7. cartography
8. cartogram
9. googleway
10. rasterVis

3. Mapas estáticos

Função `plot()`

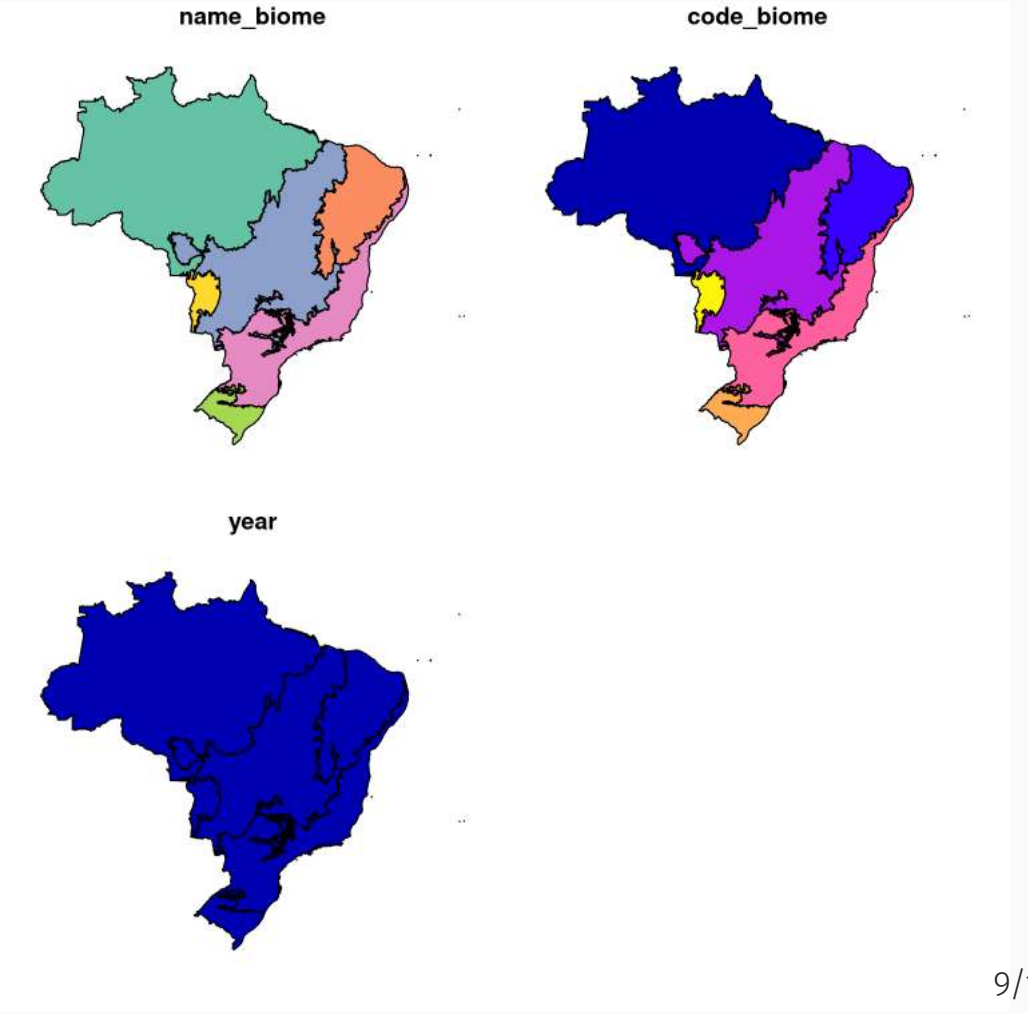
```
# biomas
biomas <- geobr::read_biomes(showProgress = FALSE) %>%
  dplyr::filter(name_biome != "Sistema Costeiro")
biomas
```

```
## Simple feature collection with 6 features and 3 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -73.98304 ymin: -33.75115 xmax: -28.84785 ymax: 5.269581
## Geodetic CRS:  SIRGAS 2000
##       name_biome code_biome year                           geom
## 1       Amazônia          1 2019 MULTIPOLYGON (((-44.08515 - ...
## 2       Caatinga          2 2019 MULTIPOLYGON (((-41.7408 -2 ...
## 3        Cerrado          3 2019 MULTIPOLYGON (((-43.39009 - ...
## 4 Mata Atlântica          4 2019 MULTIPOLYGON (((-48.70814 - ...
## 5          Pampa          5 2019 MULTIPOLYGON (((-52.82472 - ...
## 6       Pantanal          6 2019 MULTIPOLYGON (((-57.75946 - ...
```

3. Mapas estáticos

Função `plot()`

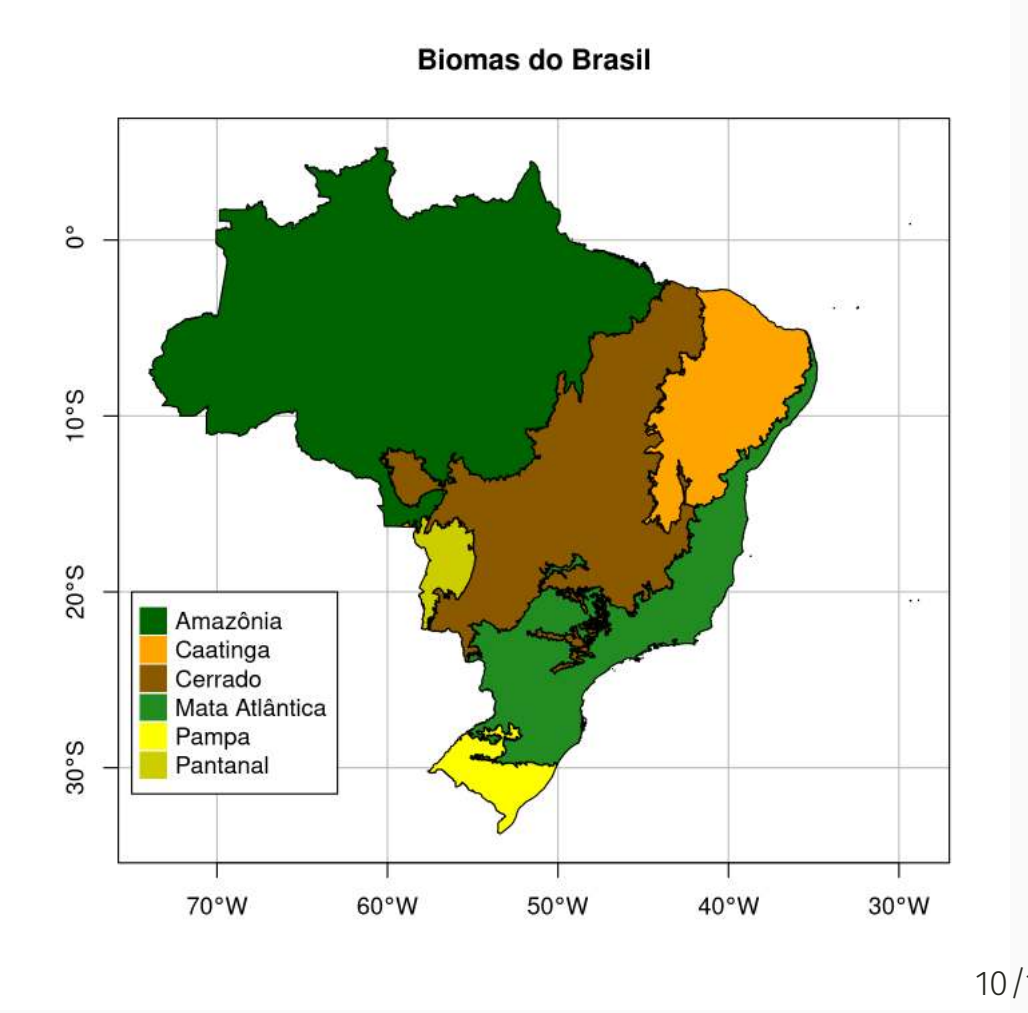
```
# plot
plot(biomas)
```



3. Mapas estáticos

Função `plot()`

```
# plot
plot(biomas$geom,
     col = c("darkgreen", "orange", "orange4",
             "forestgreen", "yellow", "yellow3"),
     main = "Biomas do Brasil", axes = TRUE,
     graticule = TRUE)

# legend
legend(x = -75, y = -20, pch = 15, cex = 1,
       pt.cex = 2.5, legend = biomas$name_biome,
       col = c("darkgreen", "orange", "orange4",
               "forestgreen", "yellow", "yellow3"))
```



3. Mapas estáticos

Função `plot()`

```
# rio claro
rc_2020 ← geobr::read_municipality(code_muni = 3543907, year = 2020, showProgress = FALSE) %>%
  sf::st_transform(crs = 4326)
rc_2020
```

```
## Simple feature collection with 1 feature and 7 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -47.76521 ymin: -22.55203 xmax: -47.46188 ymax: -22.24368
## Geodetic CRS:  WGS 84
##     code_muni name_muni code_state abbrev_state name_state code_region name_region                           geom
## 493   3543907 Rio Claro         35           SP  São Paulo           3     Sudeste MULTIPOLYGON (((-47.46875 - ...
```

3. Mapas estáticos

Função `plot()`

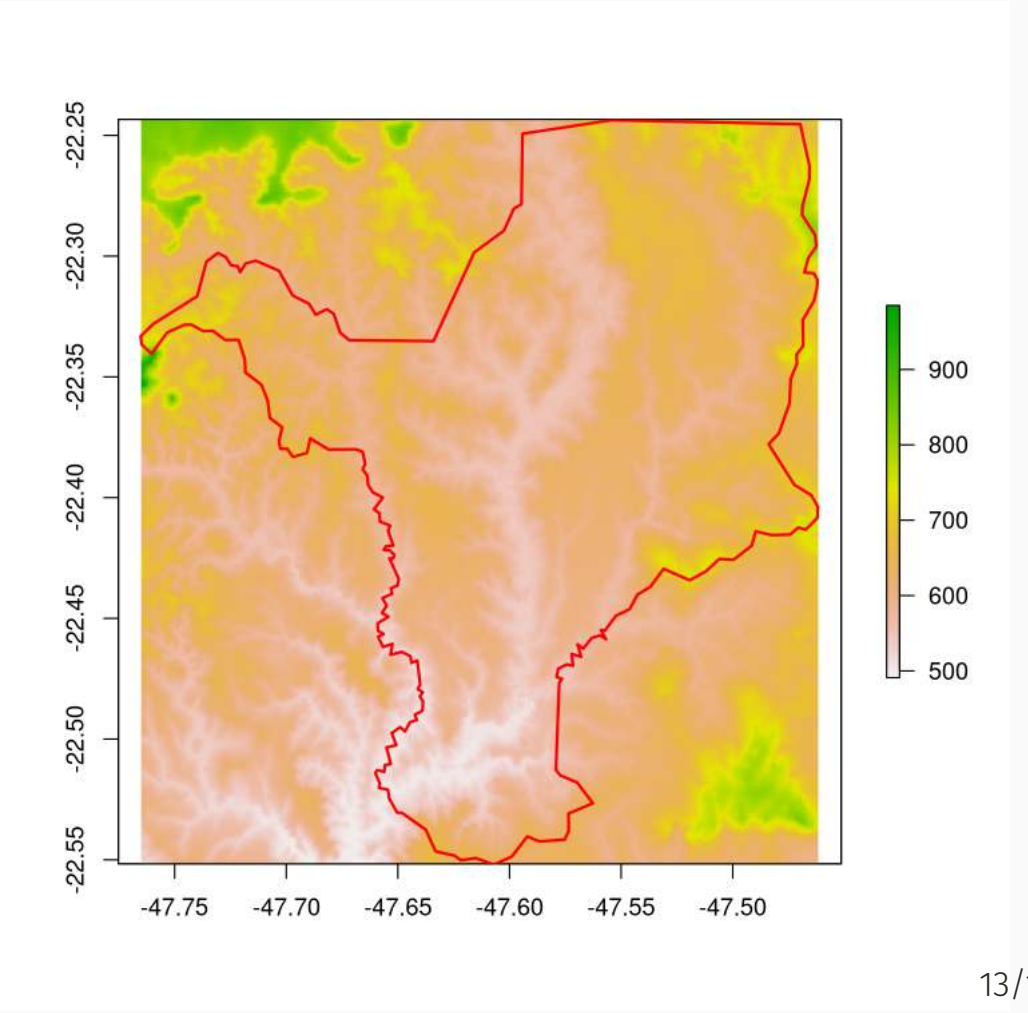
```
# importar raster
dem_rc ← raster::raster(here::here("03_dados", "raster", "srtm_27_17.tif")) %>%
  raster::crop(rc_2020)
dem_rc
```

```
## class      : RasterLayer
## dimensions : 370, 364, 134680  (nrow, ncol, ncell)
## resolution : 0.0008333333, 0.0008333333  (x, y)
## extent     : -47.765, -47.46167, -22.55167, -22.24333  (xmin, xmax, ymin, ymax)
## crs        : +proj=longlat +datum=WGS84 +no_defs
## source     : memory
## names      : srtm_27_17
## values     : 491, 985  (min, max)
```

3. Mapas estáticos

Função `plot()`

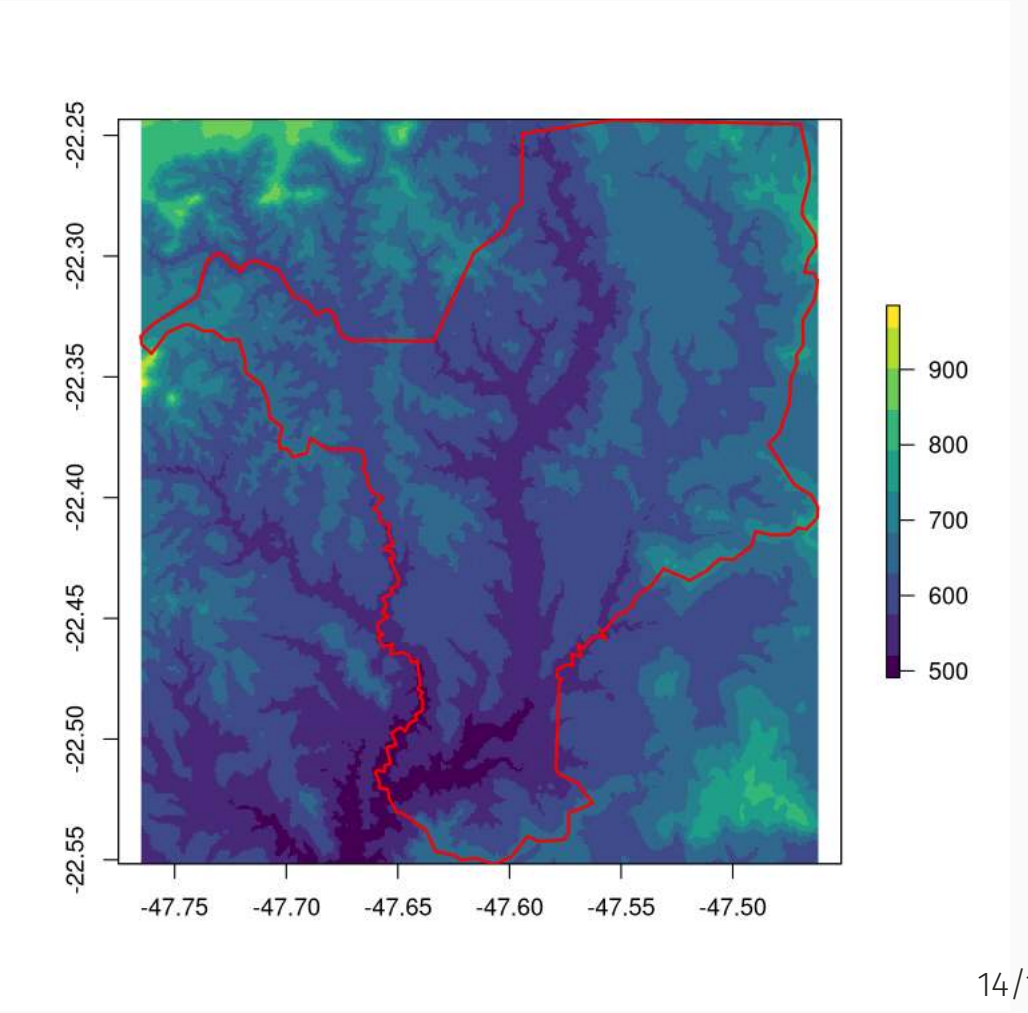
```
# plot
plot(dem_rc)
plot(rc_2020$geom, col = NA, border = "red", lwd = 2
```



3. Mapas estáticos

Função `plot()`

```
# plot
plot(dem_rc, col = viridis::viridis(10))
plot(rc_2020$geom, col = NA, border = "red", lwd = 2
```



3. Mapas estáticos

Função `plot()`

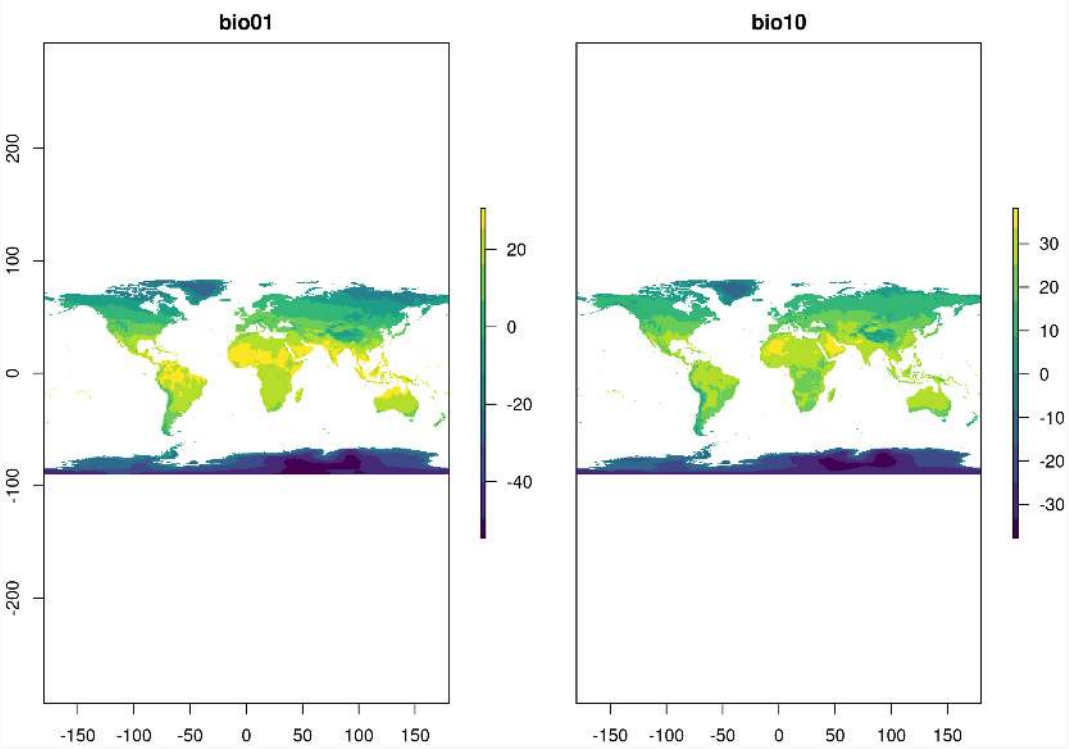
```r
# listar arquivos
files <- dir(path = here::here("03_dados", "raster"), pattern = "wc", full.names = TRUE) %>%
  grep(".tif", ., value = TRUE)

# importar
bioclim <- raster::stack(files)

# names
names(bioclim) <- c("bio01", paste0("bio", 10:19), paste0("bio0", 2:9))
bioclim
```

```
## class      : RasterStack
## dimensions : 1080, 2160, 2332800, 19  (nrow, ncol, ncell, nlayers)
## resolution : 0.1666667, 0.1666667  (x, y)
## extent     : -180, 180, -90, 90  (xmin, xmax, ymin, ymax)
## crs        : +proj=longlat +datum=WGS84 +no_defs
## names      :       bio01,       bio10,       bio11,       bio12,       bio13,       bio14,       bio15,       bio1
## min values :  -54.724354,  -37.781418,  -66.311249,    0.000000,    0.000000,    0.000000,    0.000000,    0.00000
## max values :    30.98764,    38.21617,    29.15299, 11191.00000,  2381.00000,   484.00000,   229.00169,  5284.0000
```

3. Mapas estáticos

Função `plot()`

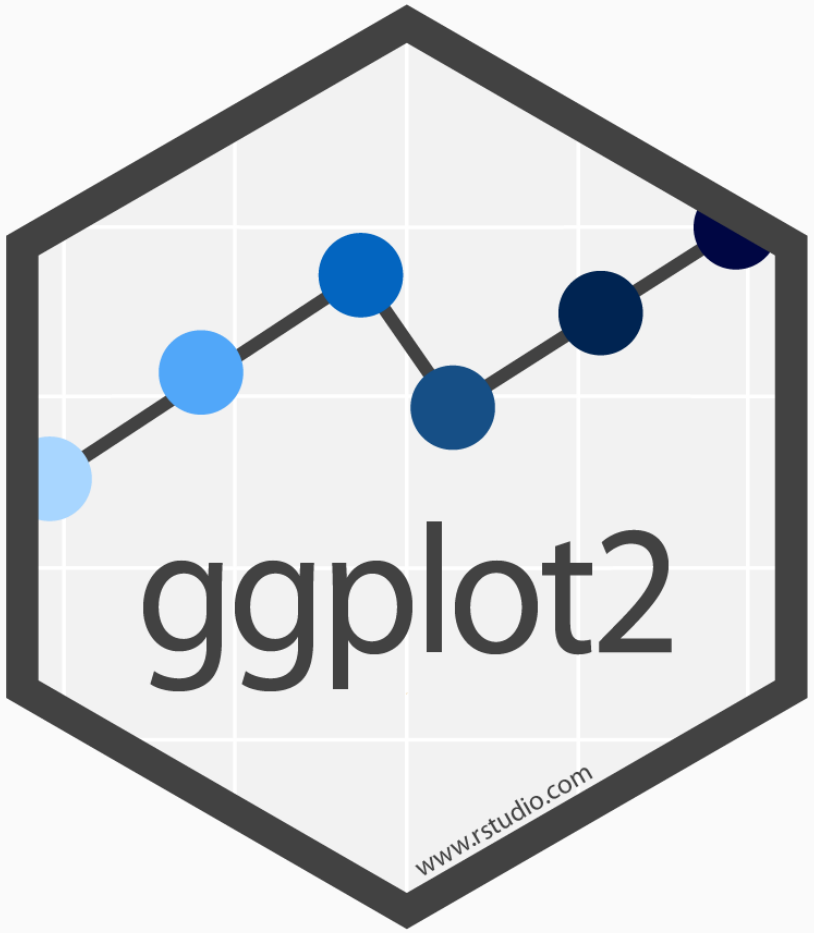
```
# plot
plot(bioclim[[1:2]], col = viridis::viridis(10))
```



3. Mapas estáticos

Função `plot()`

```
# diretorio
dir.create(here::here("03_dados", "mapas"))

# exportar mapa
png(filename = here::here("03_dados", "mapas", "mapa_biomas.png"),
    width = 20, height = 20, units = "cm", res = 300)
plot(biomas$geom,
     col = c("darkgreen", "orange", "orange4", "forestgreen", "yellow", "yellow3"),
     main = "Biomas do Brasil", axes = TRUE, graticule = TRUE)
legend(x = -75, y = -20, pch = 15, cex = 1, pt.cex = 2.5, legend = biomas$name_biome,
       col = c("darkgreen", "orange", "orange4", "forestgreen", "yellow", "yellow3"))
dev.off()
```



ggplot2

3. Mapas estáticos

ggplot2, **ggspatial** e **ggmap**

Integrado ao **tidyverse**, possui uma sintaxe própria

Principal pacote para **visualização de dados** no R

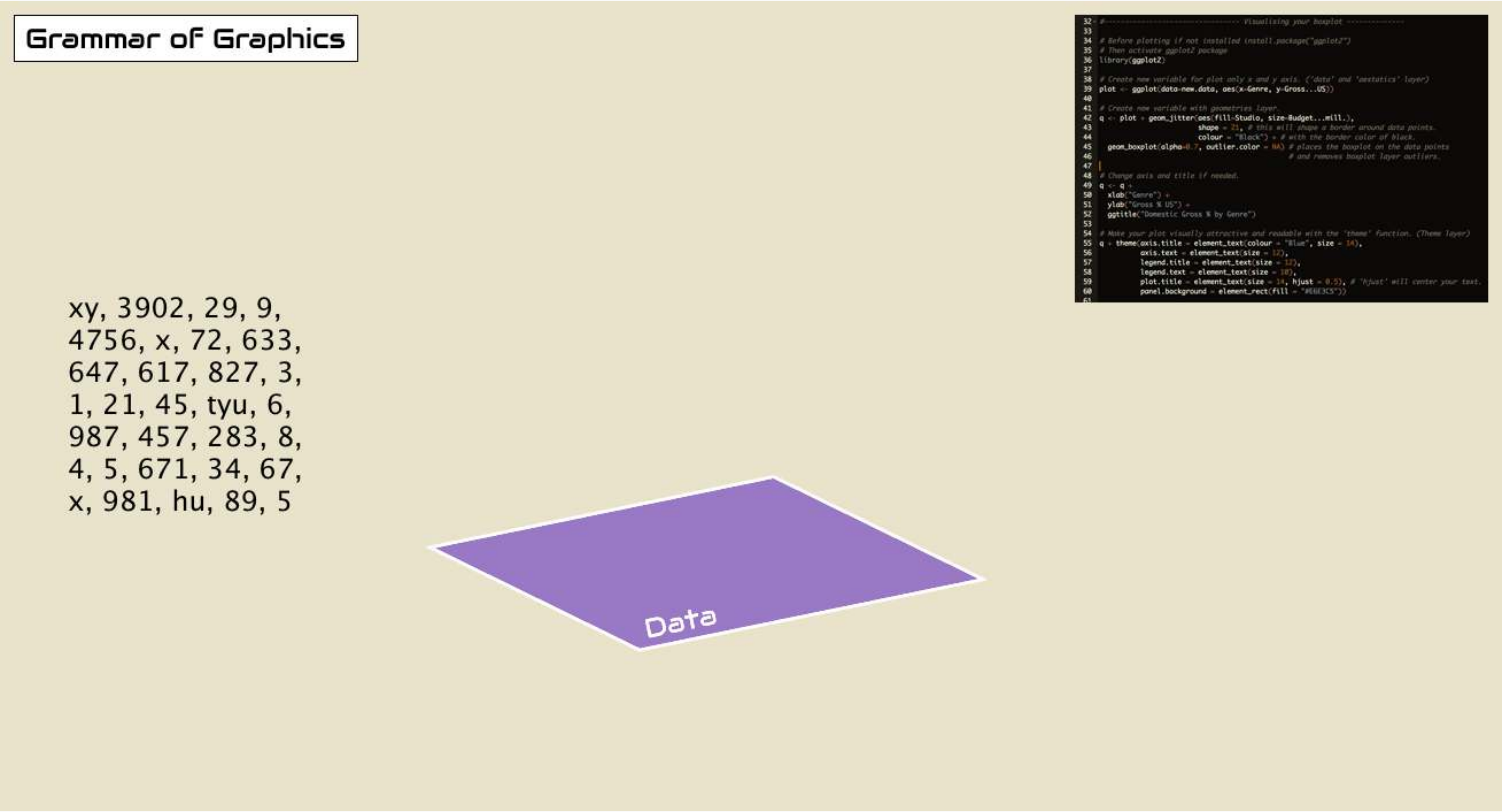
Estrutura:

```
ggplot( ... ) +
aes( ... ) +
geom_( ... ) +
stats_( ... ) +
coord_( ... ) +
facet_( ... ) +
theme_( ... )
```

ggplot2, @allison_horst

3. Mapas estáticos

ggplot2, **ggspatial** e **ggmap**

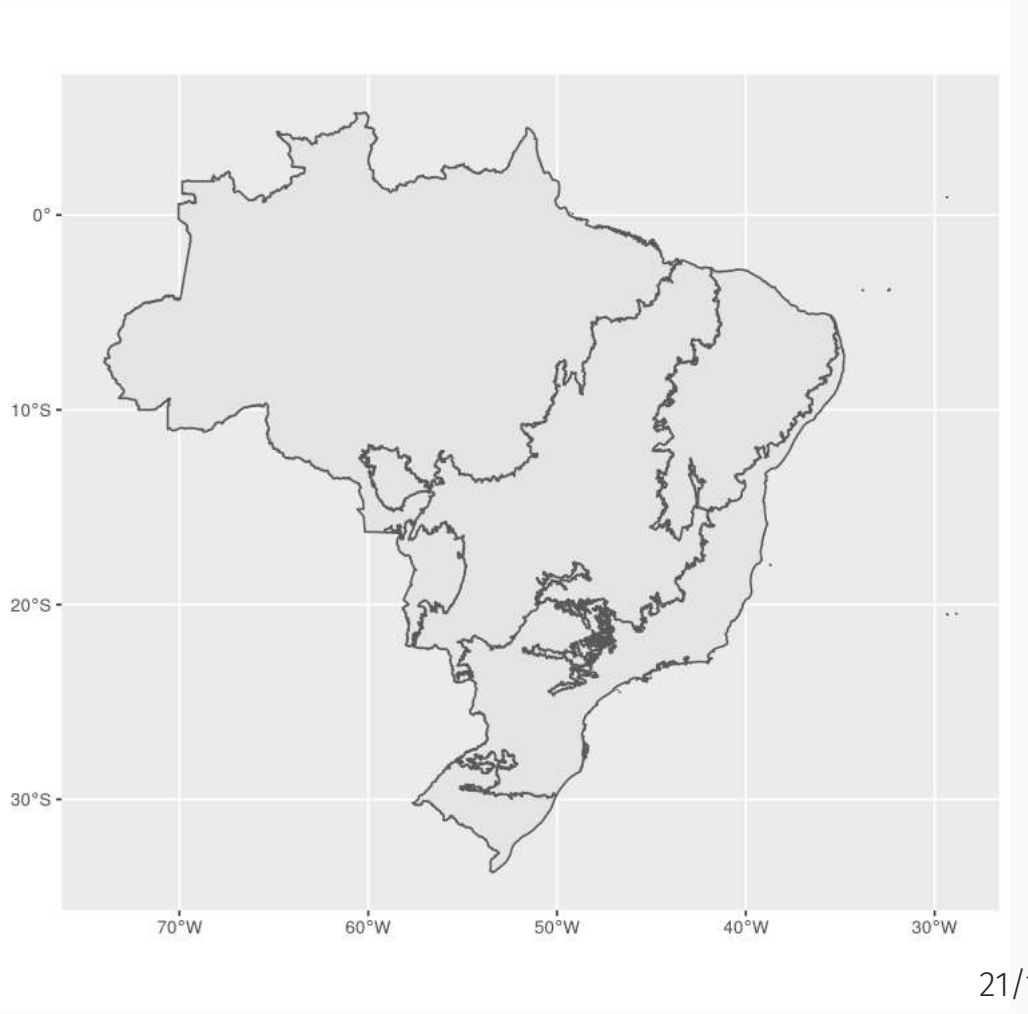


[Think About the Grammar of Graphics When Improving Your Graphs]

3. Mapas estáticos

ggplot2 e **ggspatial**

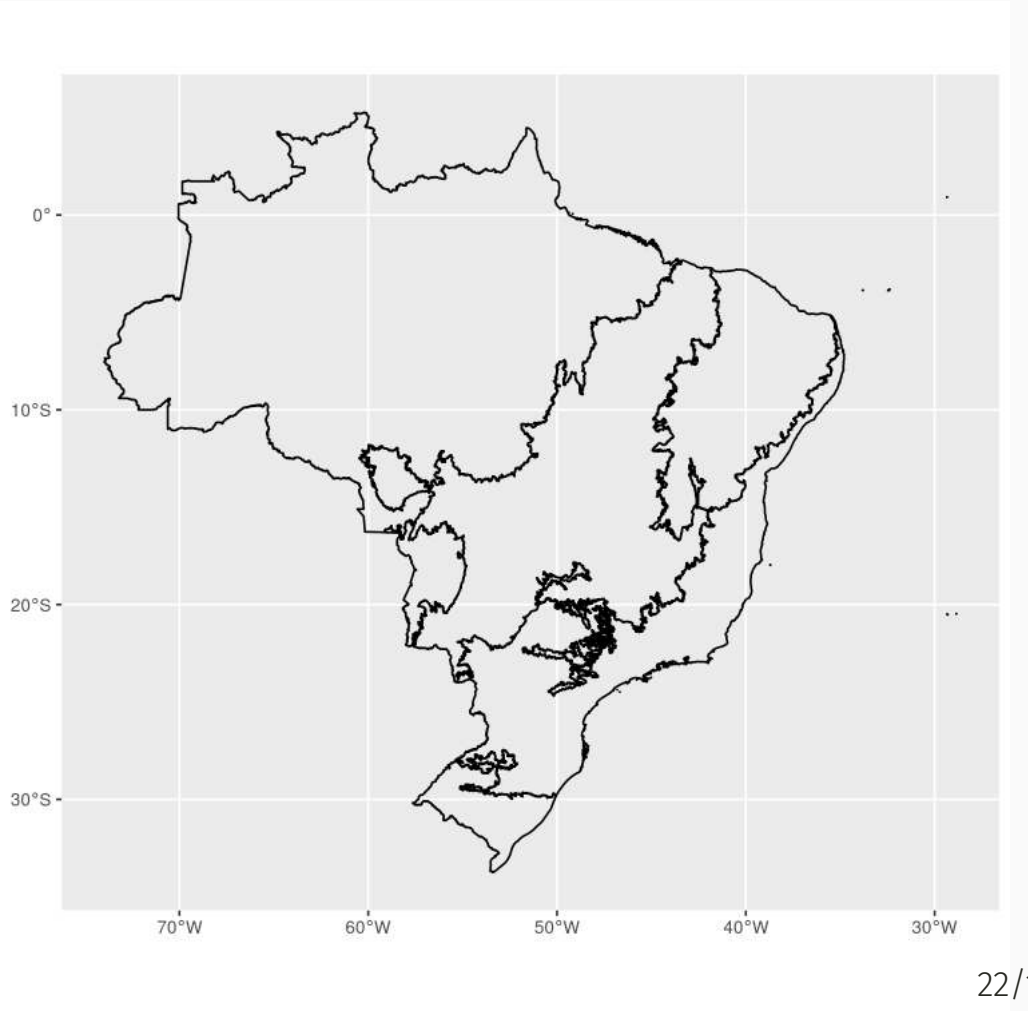
```
# dados
ggplot() +
  geom_sf(data = biomas)
```



3. Mapas estáticos

ggplot2 e **ggspatial**

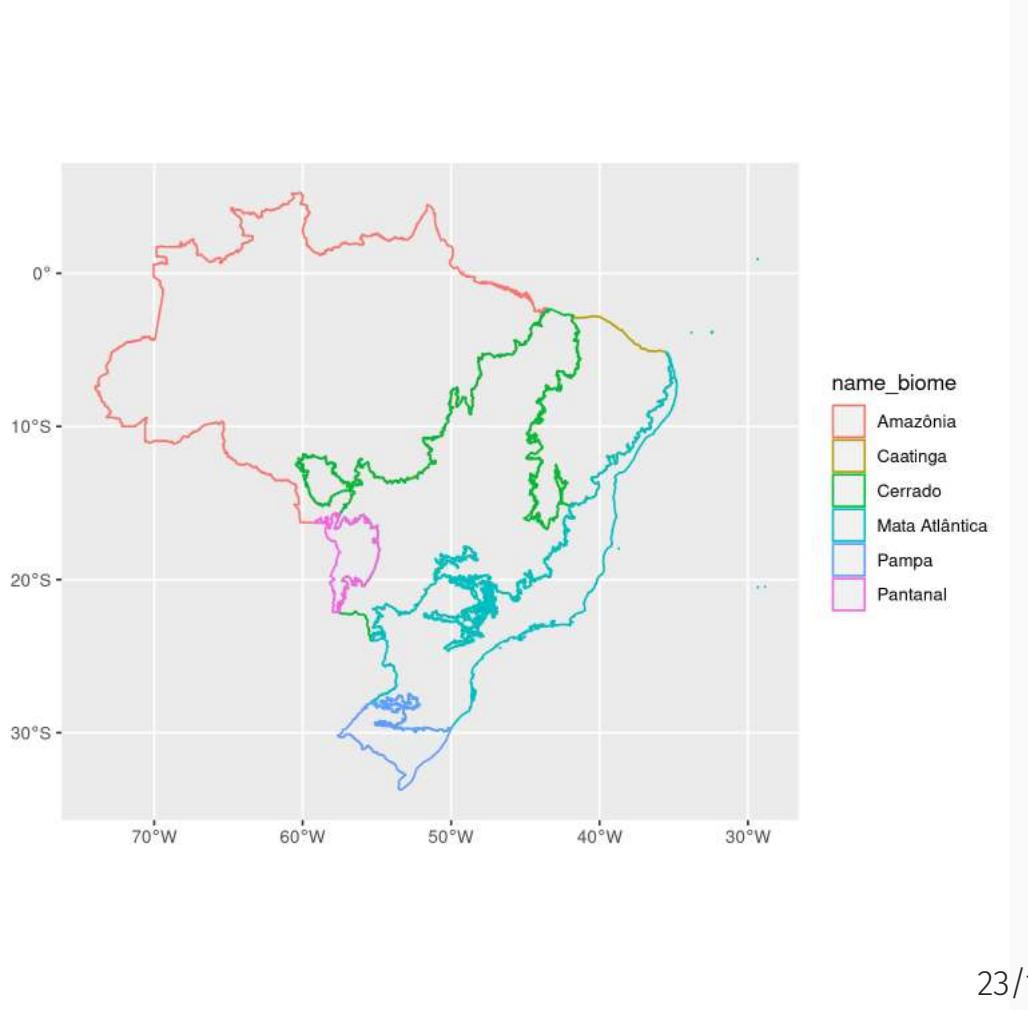
```
# cor e preenchimento
ggplot() +
  geom_sf(data = biomas, color = "black", fill = NA)
```



3. Mapas estáticos

ggplot2 e **ggspatial**

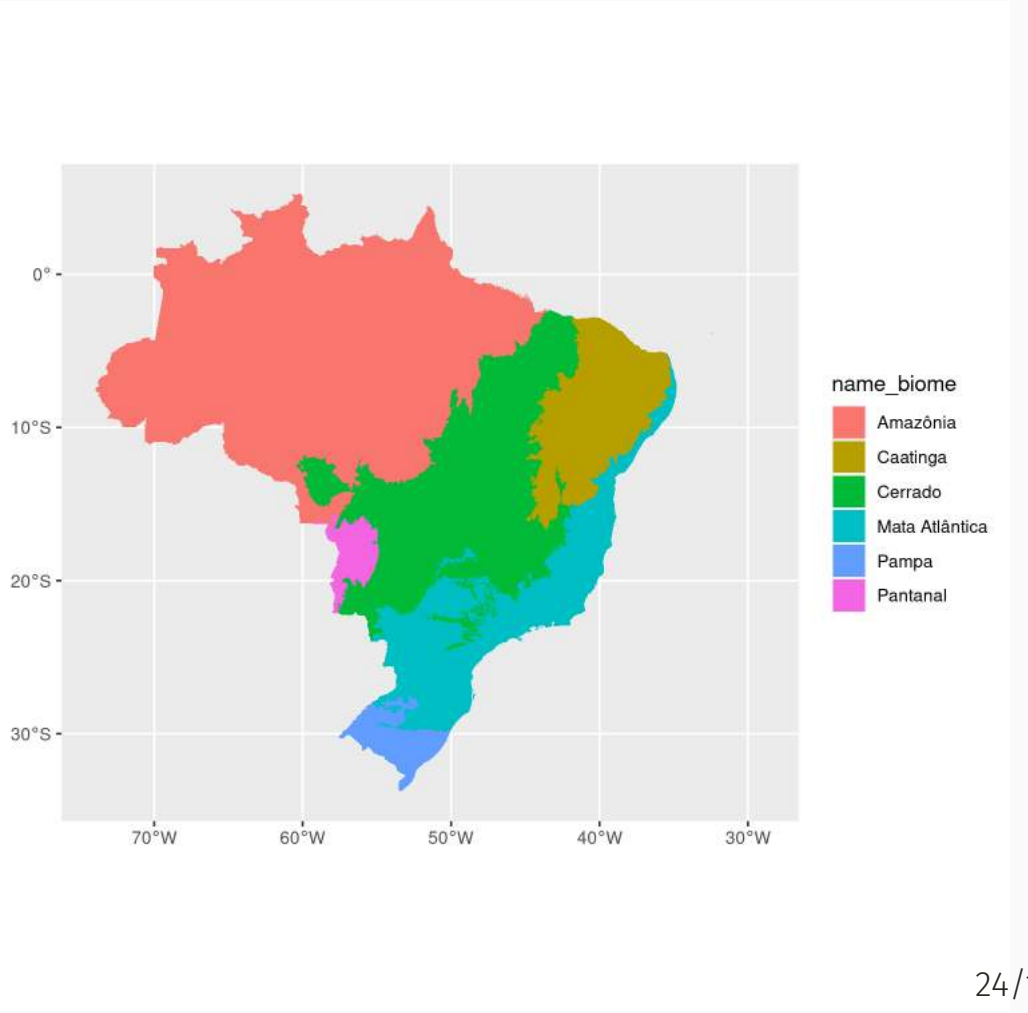
```
# cor e preenchimento
ggplot() +
  geom_sf(data = biomas,
          aes(color = name_biome),
          fill = NA)
```



3. Mapas estáticos

ggplot2 e **ggspatial**

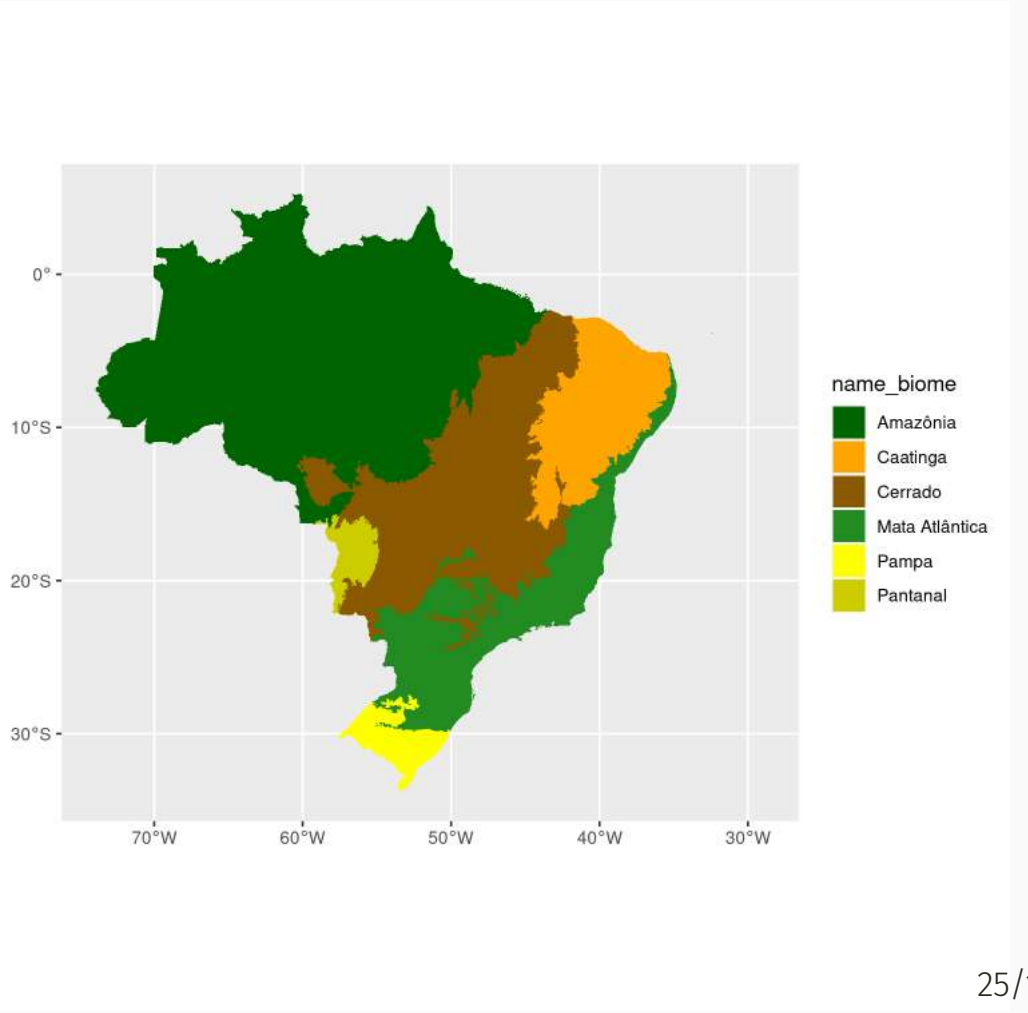
```
# cor e preenchimento
ggplot() +
  geom_sf(data = biomas,
          aes(fill = name_biome),
          color = NA)
```



3. Mapas estáticos

ggplot2 e **ggspatial**

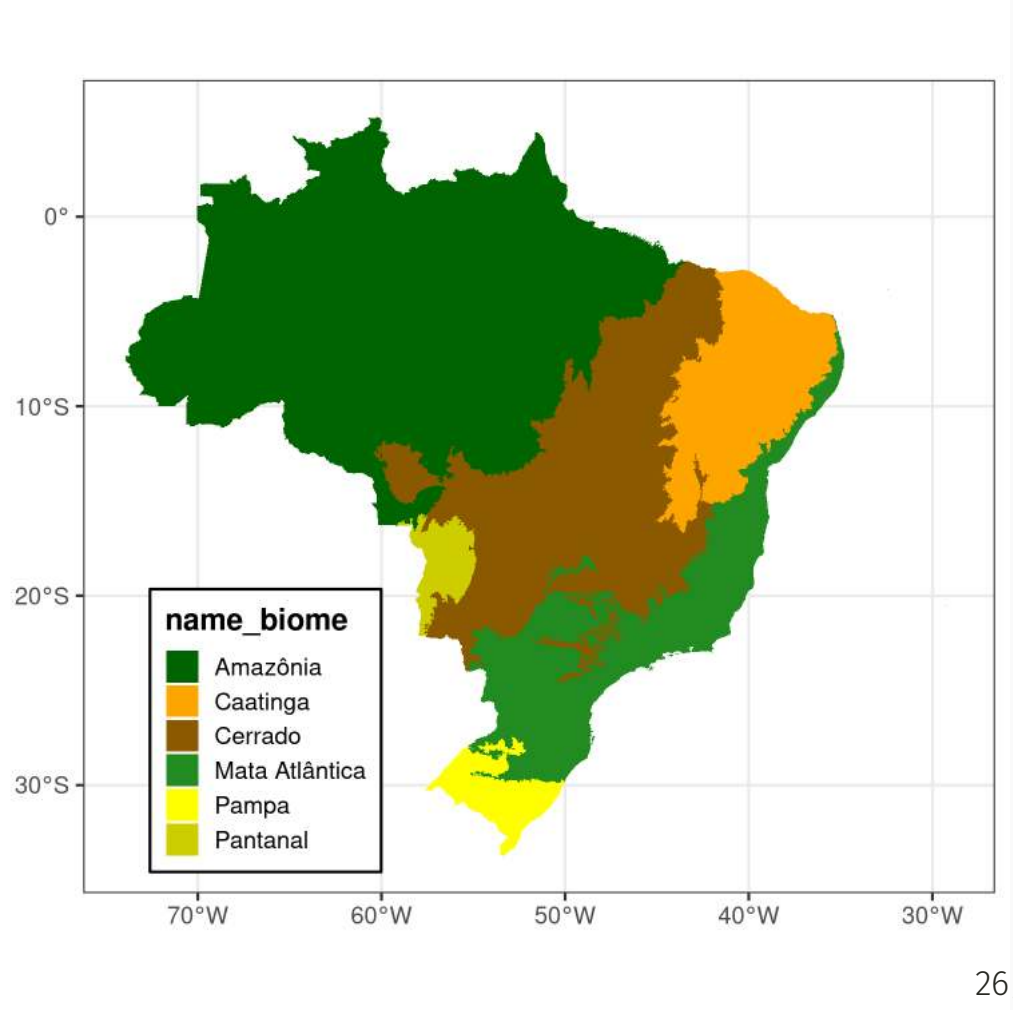
```
# definir cor
ggplot() +
  geom_sf(data = biomas, aes(fill = name_biome), colo
  scale_fill_manual(values = c("darkgreen", "orange"
                               "forestgreen", "yellow
```



3. Mapas estáticos

ggplot2 e **ggspatial**

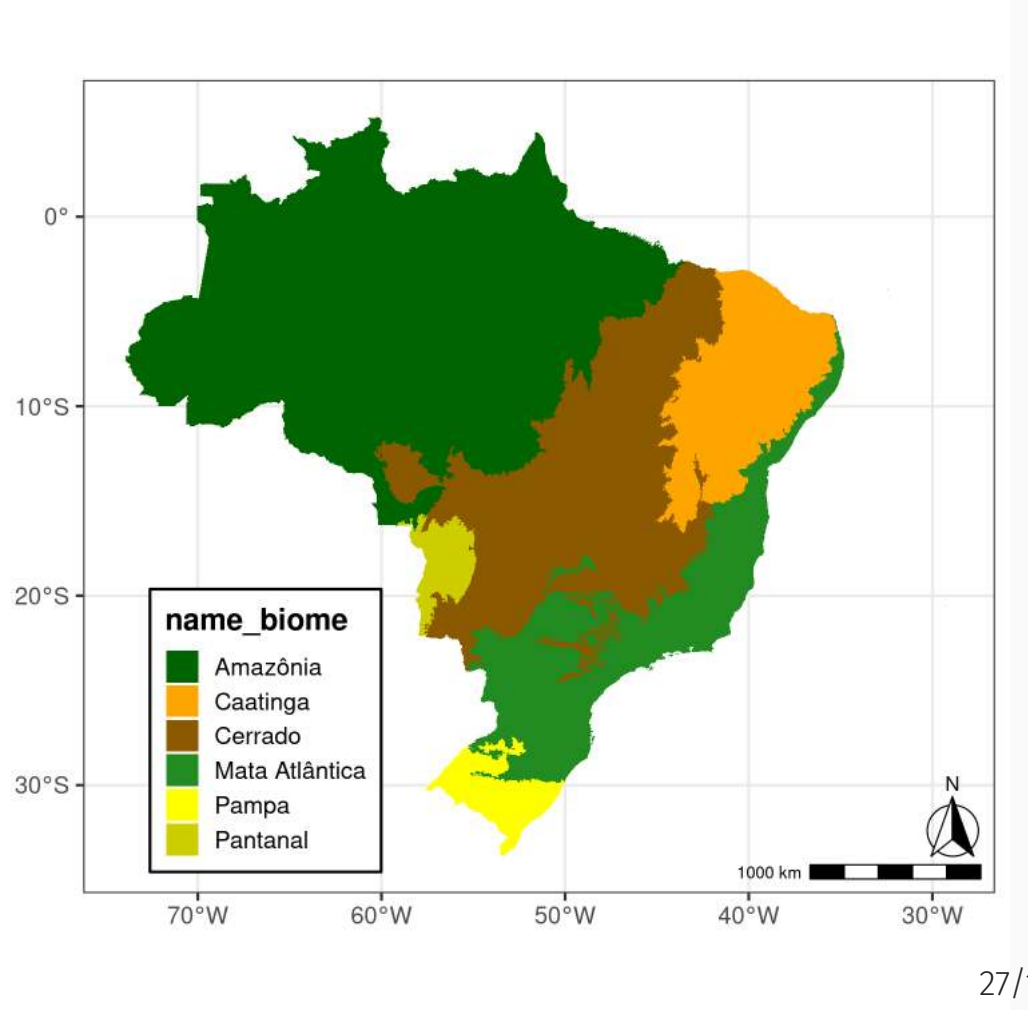
```
# tema
ggplot() +
  geom_sf(data = biomas, aes(fill = name_biome), colo
  scale_fill_manual(values = c("darkgreen", "orange"
                               "forestgreen", "yellow
  theme_bw(base_size = 15) +
  theme(legend.title = element_text(size = 15, face =
        legend.position = c(.2, .2),
        legend.background = element_rect(colour = "bl
```



3. Mapas estáticos

ggplot2 e ggspatial

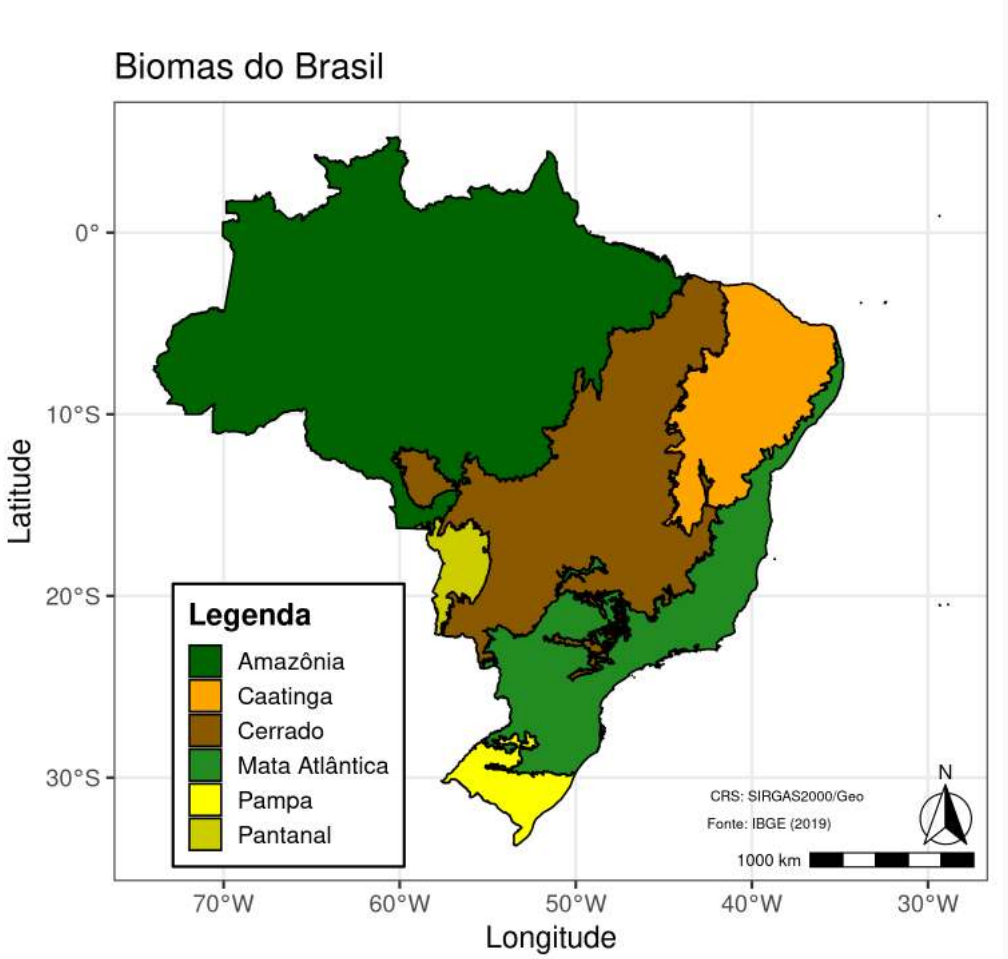
```
# barra de escala e norte
ggplot() +
  geom_sf(data = biomas, aes(fill = name_biome), colo
  scale_fill_manual(values = c("darkgreen", "orange"
                               "forestgreen", "yellow
  theme_bw(base_size = 15) +
  theme(legend.title = element_text(size = 15, face =
        legend.position = c(.2, .2),
        legend.background = element_rect(colour = "bl
  annotation_scale(location = "br") +
  annotation_north_arrow(location = "br", which_north
                         pad_x = unit(0, "cm"), pad_y
                         style = north_arrow_fancy_or
```



3. Mapas estáticos

ggplot2 e ggspatial

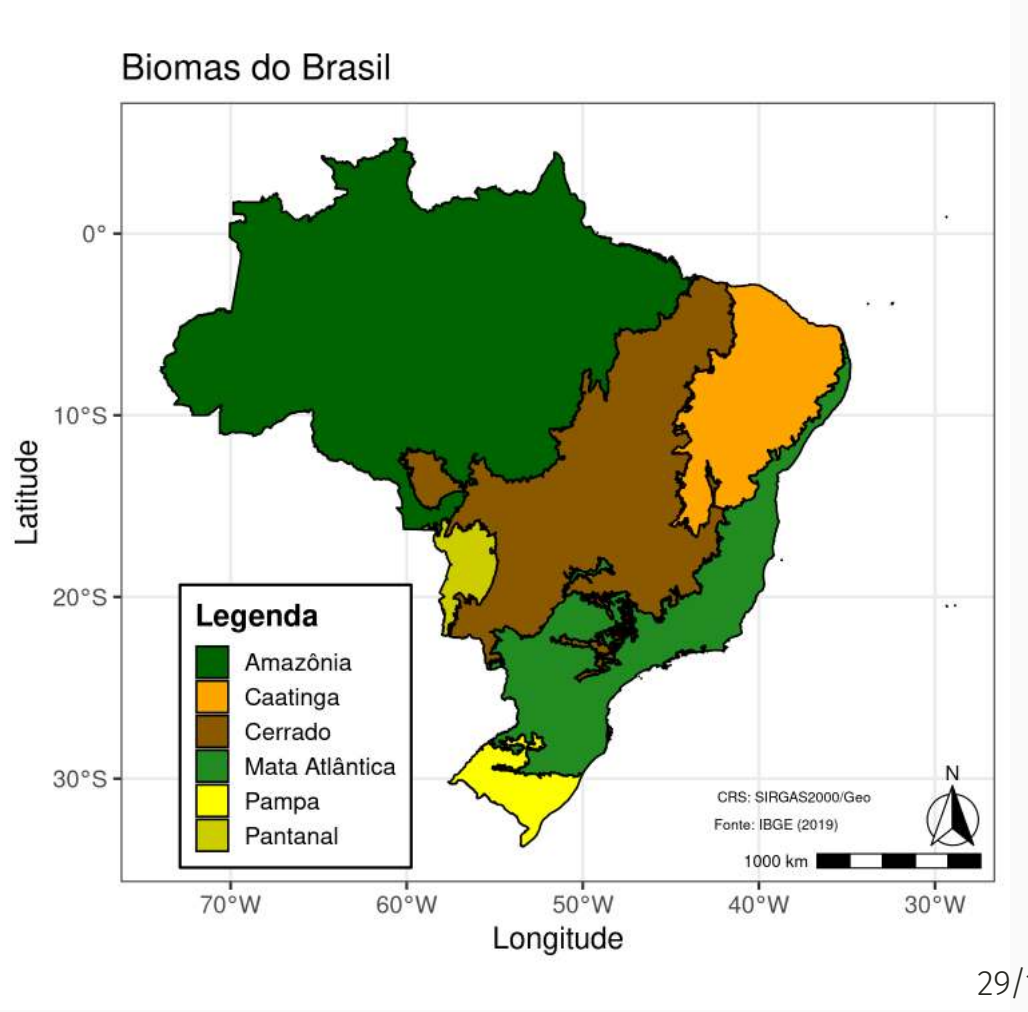
```
# nomes e anotacoes
ggplot(data = biomas) +
  aes(fill = name_biome) +
  geom_sf(color = "black") +
  scale_fill_manual(values = c("darkgreen", "orange"
                               "forestgreen", "yellow
  theme_bw(base_size = 15) +
  theme(legend.title = element_text(size = 15, face =
        legend.position = c(.2, .2),
        legend.background = element_rect(colour = "b
  annotation_scale(location = "br") +
  annotation_north_arrow(location = "br", which_north
                         pad_x = unit(0, "cm"), pad_y
                         style = north_arrow_fancy_or
  annotate(geom = "text", label = "CRS: SIRGAS2000/Ge
  annotate(geom = "text", label = "Fonte: IBGE (2019
  labs(title = "Biomas do Brasil", fill = "Legenda",
```



3. Mapas estáticos

ggplot2 e ggspatial

```
# atribuicao
map_biomas_ggplot2 ← ggplot(data = biomas) +
  aes(fill = name_biome) +
  geom_sf(color = "black") +
  scale_fill_manual(values = c("darkgreen", "orange"
                               "forestgreen", "yellow
  theme_bw(base_size = 15) +
  theme(legend.title = element_text(size = 15, face =
        legend.position = c(.2, .2),
        legend.background = element_rect(colour = "b
  annotation_scale(location = "br") +
  annotation_north_arrow(location = "br", which_north
                         pad_x = unit(0, "cm"), pad_y
                         style = north_arrow_fancy_or
  annotate(geom = "text", label = "CRS: SIRGAS2000/Ge
  annotate(geom = "text", label = "Fonte: IBGE (2019
  labs(title = "Biomas do Brasil", fill = "Legenda",
map_biomas_ggplot2
```



3. Mapas estáticos

ggplot2 e ggspatial

```
# dados
dem_rc_da ← raster::rasterToPoints(dem_rc) %>%
  tibble::as_tibble()
head(dem_rc_da)
```

.pull-right[

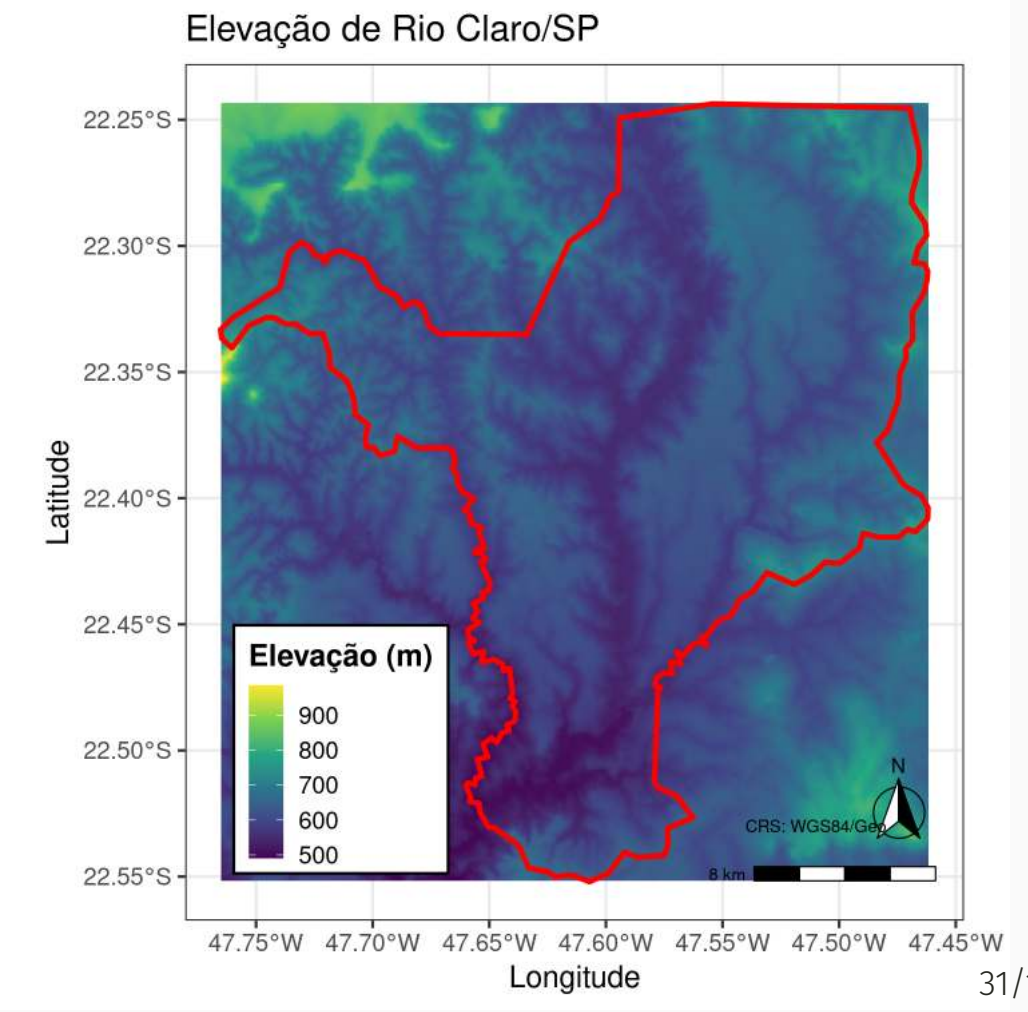
```
## ¤[90m# A tibble: 6 × 3¤[39m
##       x     y srtm_27_17
##   ¤[3m¤[90m<dbl>¤[39m¤[23m ¤[3m¤[90m<dbl>¤[39m¤[23m      ¤[
## ¤[90m1¤[39m -¤[31m47¤[39m¤[31m.¤[39m¤[31m8¤[39m -¤[31m22¤[3
## ¤[90m2¤[39m -¤[31m47¤[39m¤[31m.¤[39m¤[31m8¤[39m -¤[31m22¤[39m¤[31m.¤[39m¤[31m2¤[39m        856
## ¤[90m3¤[39m -¤[31m47¤[39m¤[31m.¤[39m¤[31m8¤[39m -¤[31m22¤[39m¤[31m.¤[39m¤[31m2¤[39m        856
## ¤[90m4¤[39m -¤[31m47¤[39m¤[31m.¤[39m¤[31m8¤[39m -¤[31m22¤[39m¤[31m.¤[39m¤[31m2¤[39m        856
## ¤[90m5¤[39m -¤[31m47¤[39m¤[31m.¤[39m¤[31m8¤[39m -¤[31m22¤[39m¤[31m.¤[39m¤[31m2¤[39m        853
## ¤[90m6¤[39m -¤[31m47¤[39m¤[31m.¤[39m¤[31m8¤[39m -¤[31m22¤[39m¤[31m.¤[39m¤[31m2¤[39m        852
```

]

3. Mapas estáticos

ggplot2 e ggspatial

```
# plot
map_dem_rc_ggplot2 ← ggplot() +
  geom_raster(data = dem_rc_da, aes(x = x, y = y, fi
  geom_sf(data = rc_2020, col = "red", fill = NA, si
  scale_fill_viridis_c() +
  coord_sf() +
  theme_bw(base_size = 15) +
  theme(legend.title = element_text(size = 15, face
        legend.position = c(.2, .2),
        legend.background = element_rect(colour = "b
  annotation_scale(location = "br",
                   pad_x = unit(.5, "cm"), pad_y = u
  annotation_north_arrow(location = "br", which_nort
                         pad_x = unit(.4, "cm"), pad
                         style = north_arrow_fancy_o
  annotate(geom = "text", label = "CRS: WGS84/Geo", 
  labs(title = "Elevação de Rio Claro/SP", fill = "E
map_dem_rc_ggplot2
```



3. Mapas estáticos

ggplot2 e ggspatial

```
# exportar
ggsave(
  filename = here::here("03_dados", "mapas", "mapa_biomas_ggplot2.png"),
  plot = map_biomas_ggplot2,
  width = 20,
  height = 20,
  units = "cm",
  dpi = 300
)
```

3. Mapas estáticos

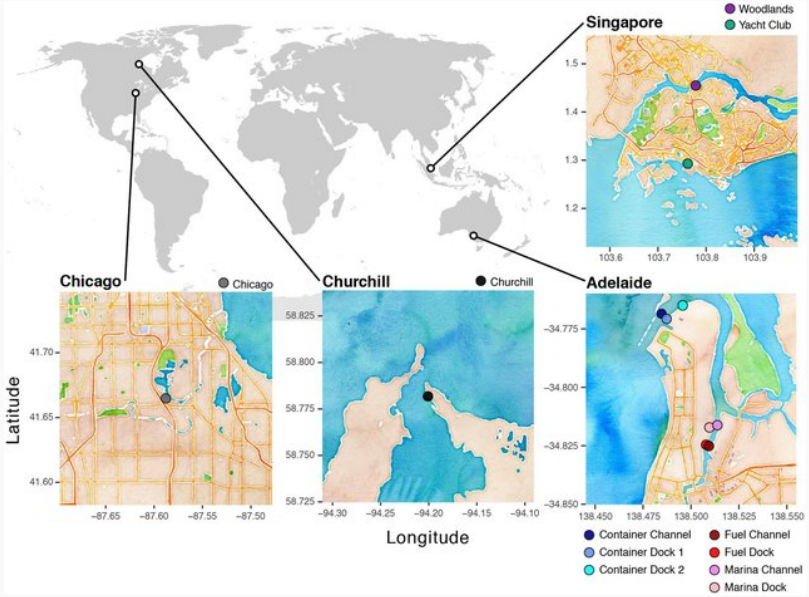
ggplot2 e ggspatial

```
# exportar
ggsave(
  filename = here::here("03_dados", "mapas", "mapa_dem_rc_rc_ggplot2.png"),
  plot = map_dem_rc_ggplot2,
  width = 20,
  height = 20,
  units = "cm",
  dpi = 300
)
```

3. Mapas estáticos

ggmap

```
# pacote
# install.packages("ggmap")
library(ggmap)

# limite
brasil ← c(left = -80, bottom = -35, right = -30, top = 5)
rio_claro ← c(left = -47.67, bottom = -22.48, right = -47.46, top = -22.32)
```

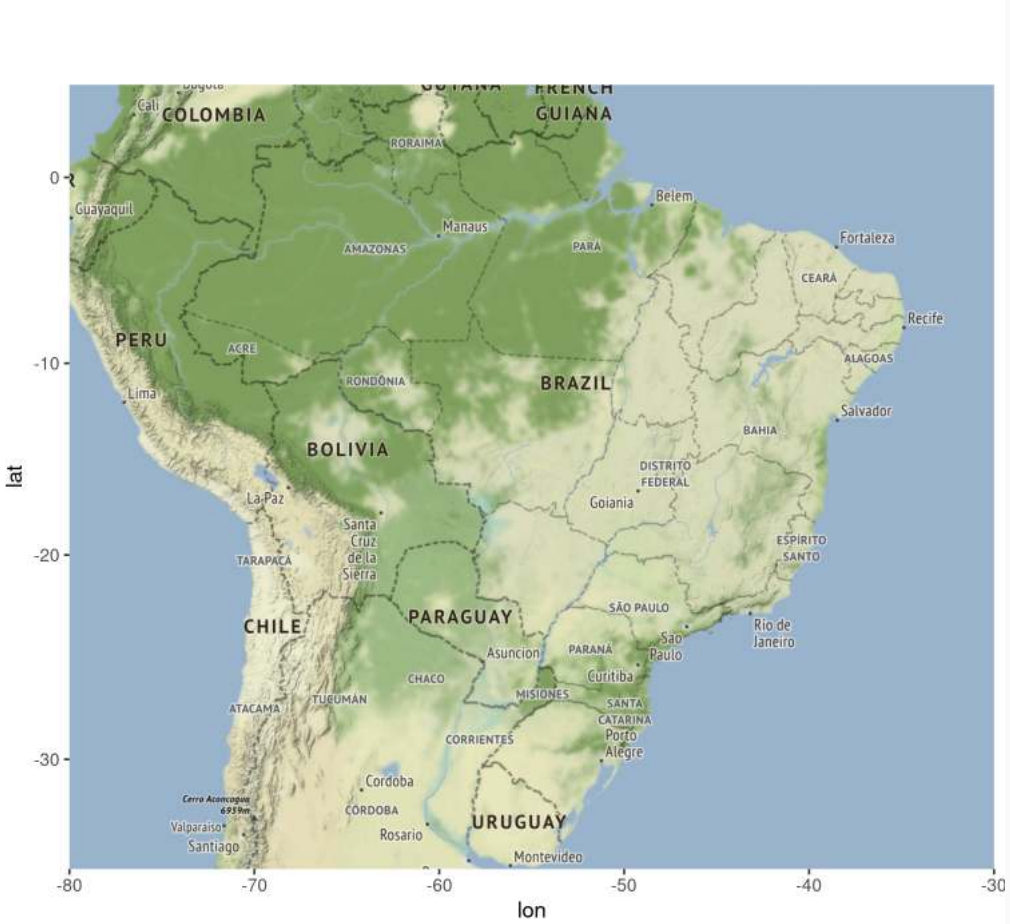


ggmap

3. Mapas estáticos

ggmap

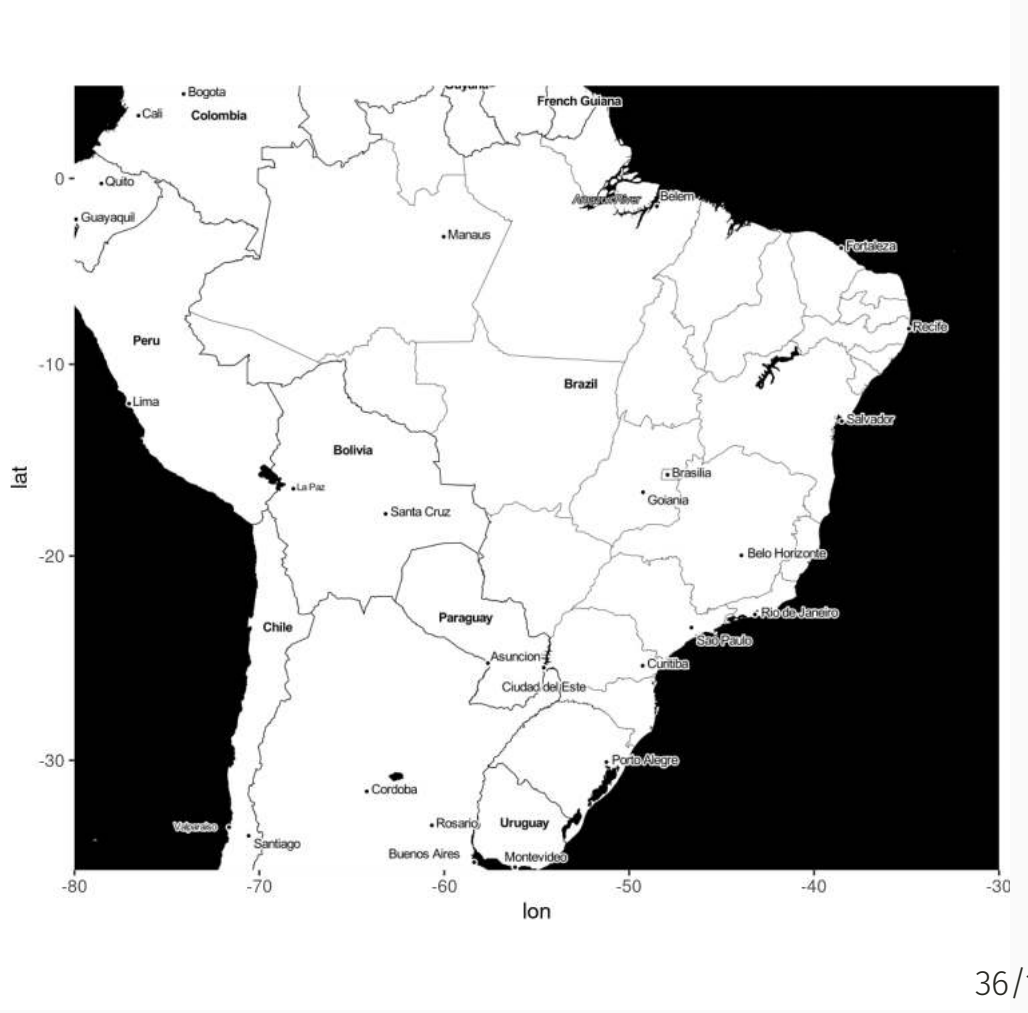
```
# plot
get_stamenmap(bbox = brasil,
              maptype = "terrain",
              zoom = 5) %>%
  ggmap()
```



3. Mapas estáticos

ggmap

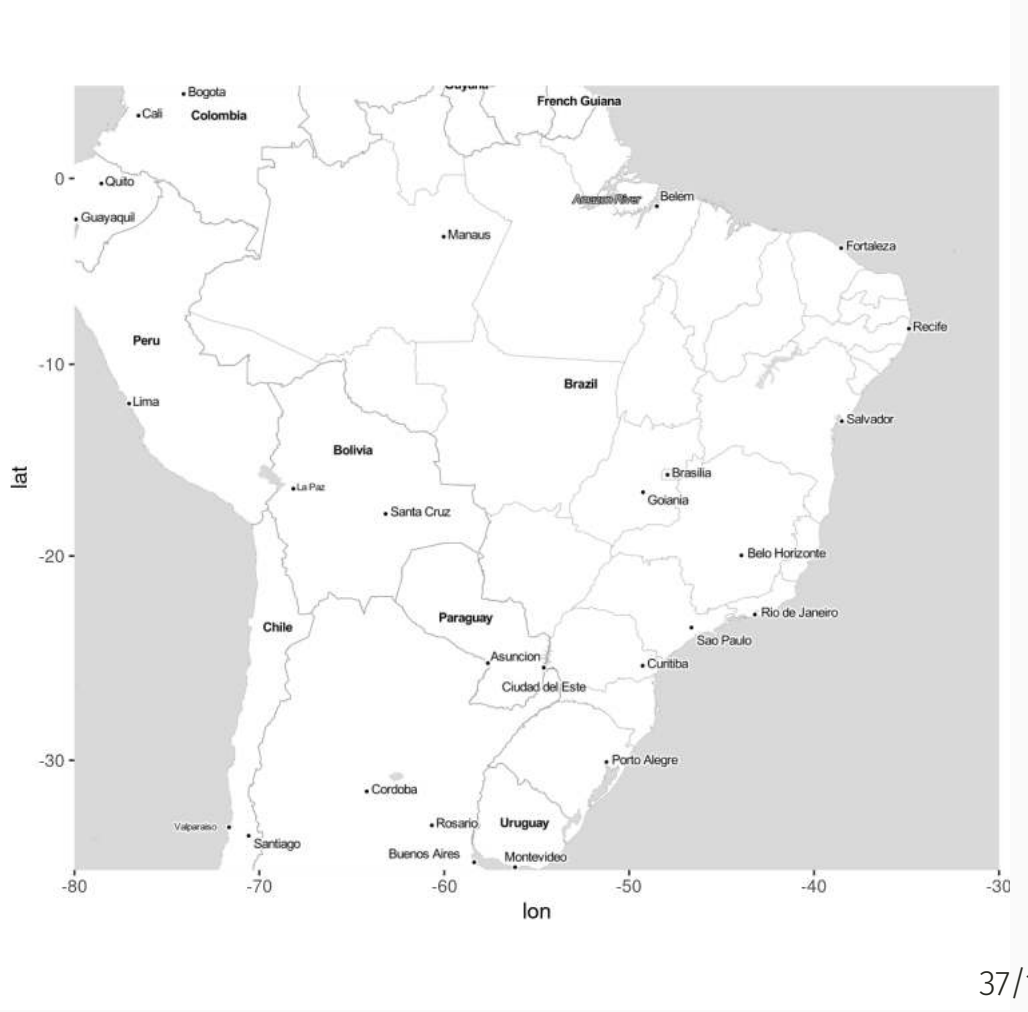
```
# plot
get_stamenmap(bbox = brasil,
              maptype = "toner",
              zoom = 5) %>%
  ggmap()
```



3. Mapas estáticos

ggmap

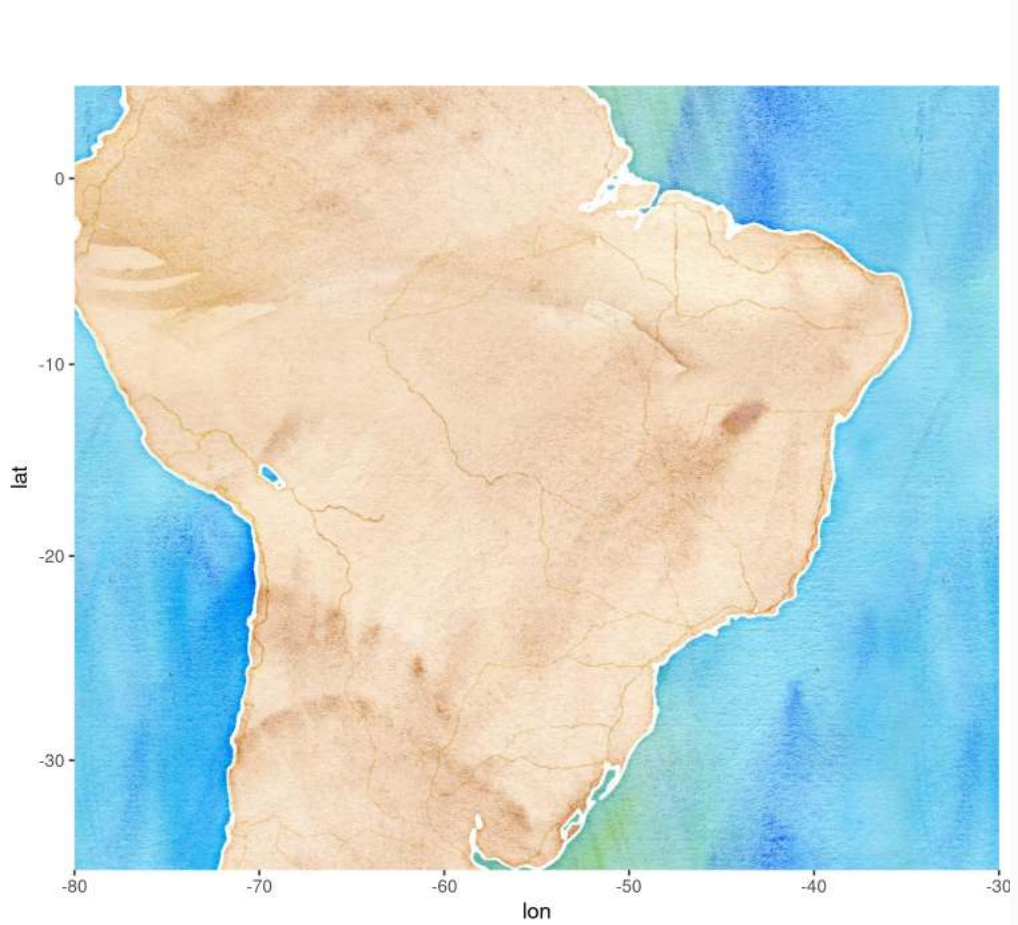
```
# plot
get_stamenmap(bbox = brasil,
              maptype = "toner-lite",
              zoom = 5) %>%
  ggmap()
```



3. Mapas estáticos

ggmap

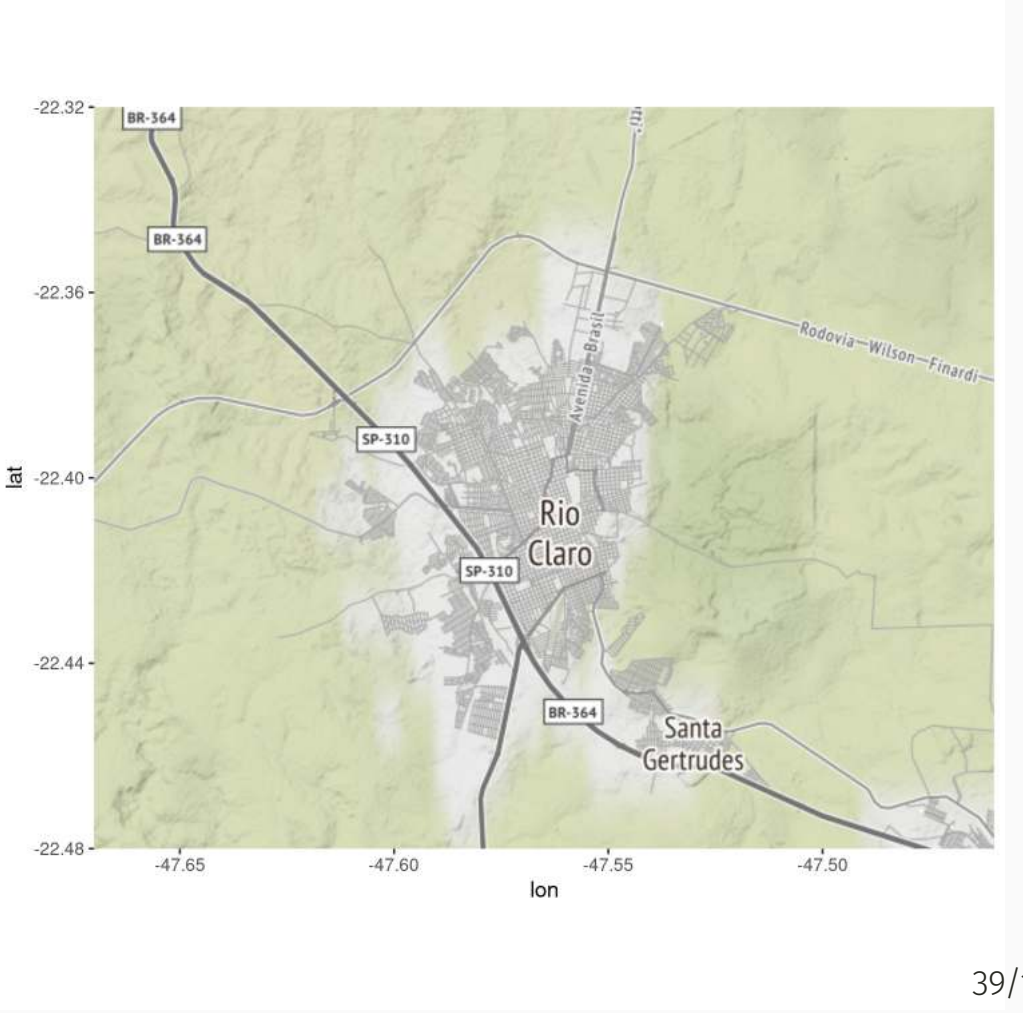
```
# plot
get_stamenmap(bbox = brasil,
              maptype = "watercolor",
              zoom = 5) %>%
  ggmap()
```

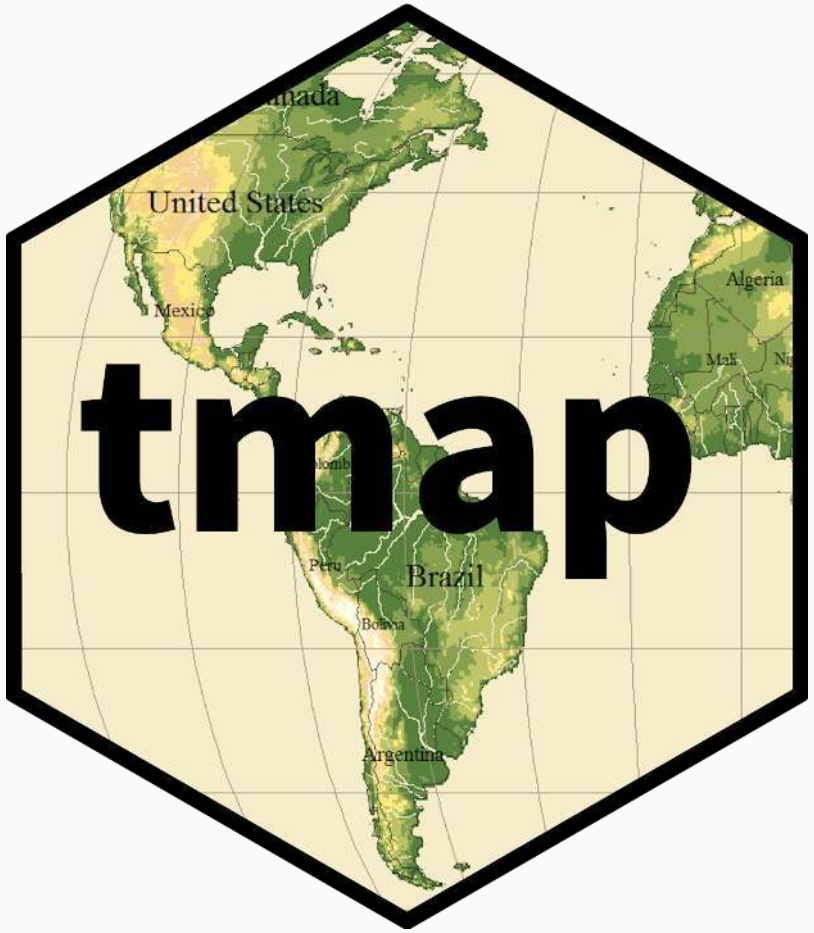


3. Mapas estáticos

ggmap

```
# plot
get_stamenmap(bbox = rio_claro,
              maptype = "terrain",
              zoom = 12) %>%
  ggmap()
```



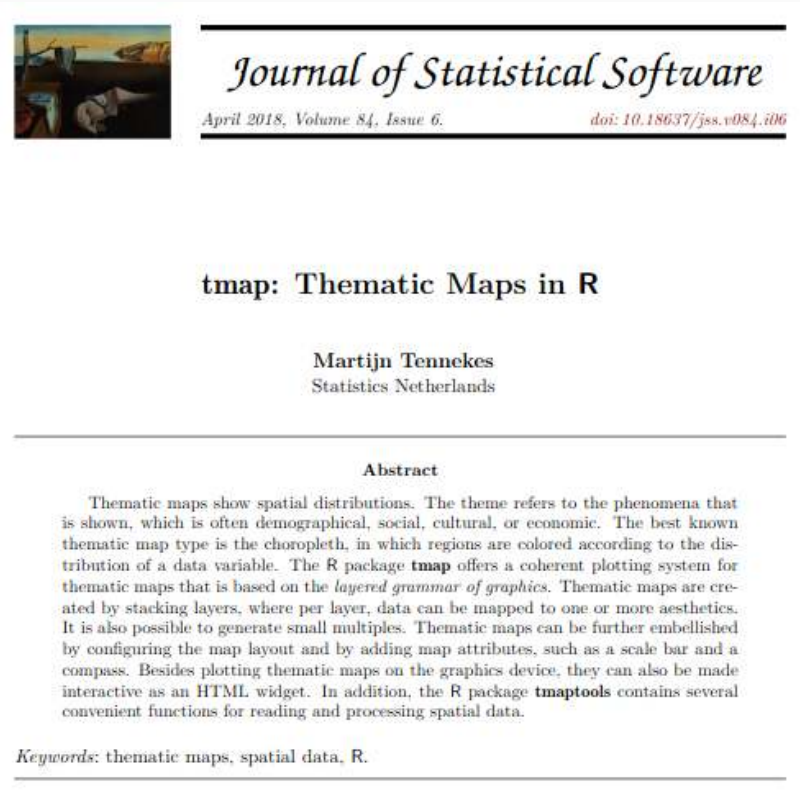

tmap
United States
Mexico
Algeria
Brazil
Peru
Bolivia
Argentina

tmap

3. Mapas estáticos

tmap

Artigo



Journal of Statistical Software
April 2018, Volume 84, Issue 6. *doi: 10.18637/jss.v084.i06*

tmap: Thematic Maps in R

Martijn Tennekes
Statistics Netherlands

Abstract

Thematic maps show spatial distributions. The theme refers to the phenomena that is shown, which is often demographical, social, cultural, or economic. The best known thematic map type is the choropleth, in which regions are colored according to the distribution of a data variable. The R package **tmap** offers a coherent plotting system for thematic maps that is based on the *layered grammar of graphics*. Thematic maps are created by stacking layers, where per layer, data can be mapped to one or more aesthetics. It is also possible to generate small multiples. Thematic maps can be further embellished by configuring the map layout and by adding map attributes, such as a scale bar and a compass. Besides plotting thematic maps on the graphics device, they can also be made interactive as an HTML widget. In addition, the R package **tmaptools** contains several convenient functions for reading and processing spatial data.

Keywords: thematic maps, spatial data, R.

Tennekes (2018)

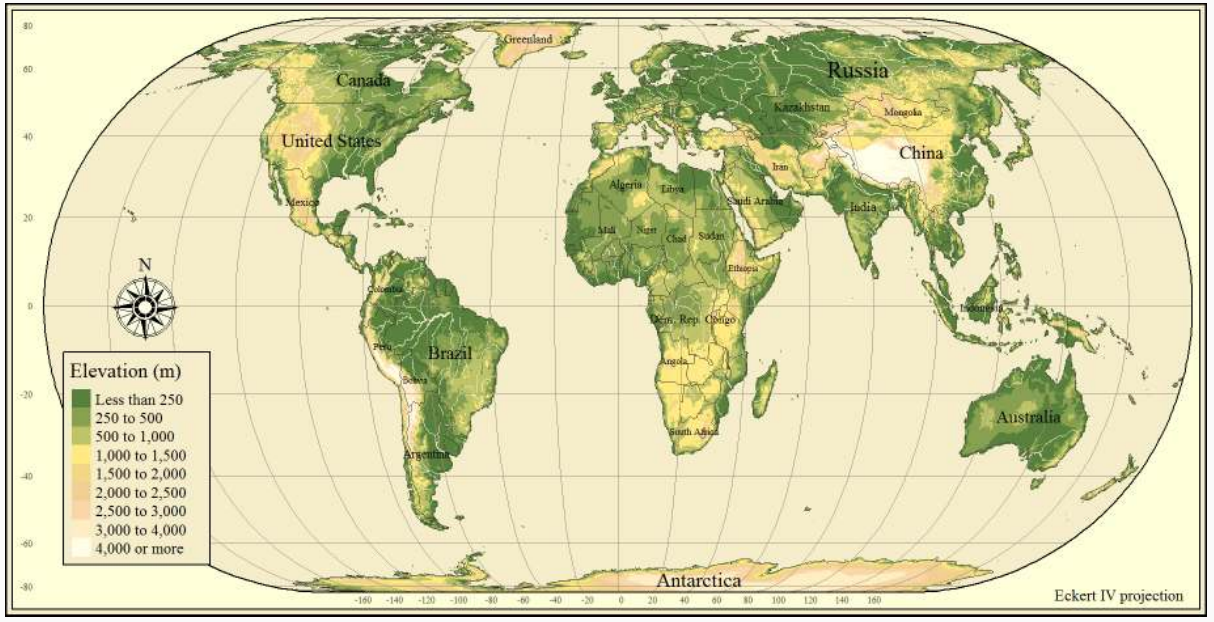
tmap

Sintaxe baseada no **ggplot2** e na **Gramática dos Gráficos**

Dois modos: **plot** (estáticos) e **visualização** (interativos)

Estrutura:

```
tm_shape( ... ) +
tm_*( ... ) +
tm_facets( ... ) +
tm_layout( ... )

tm_style( ... )
tmap_arrange( ... )
tm_mode( ... )
```

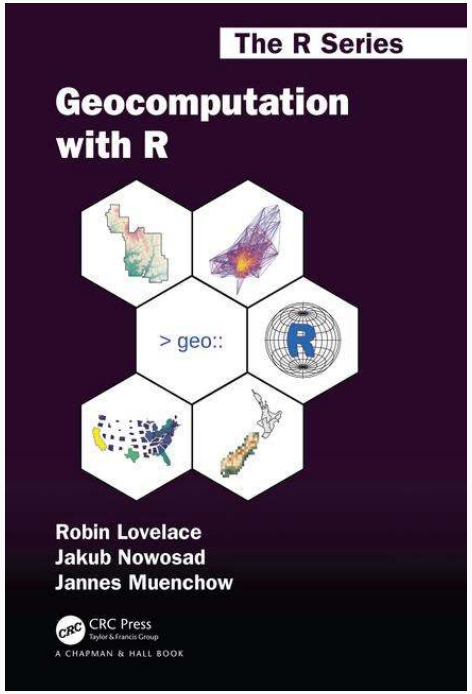


tmap

3. Mapas estáticos

tmap

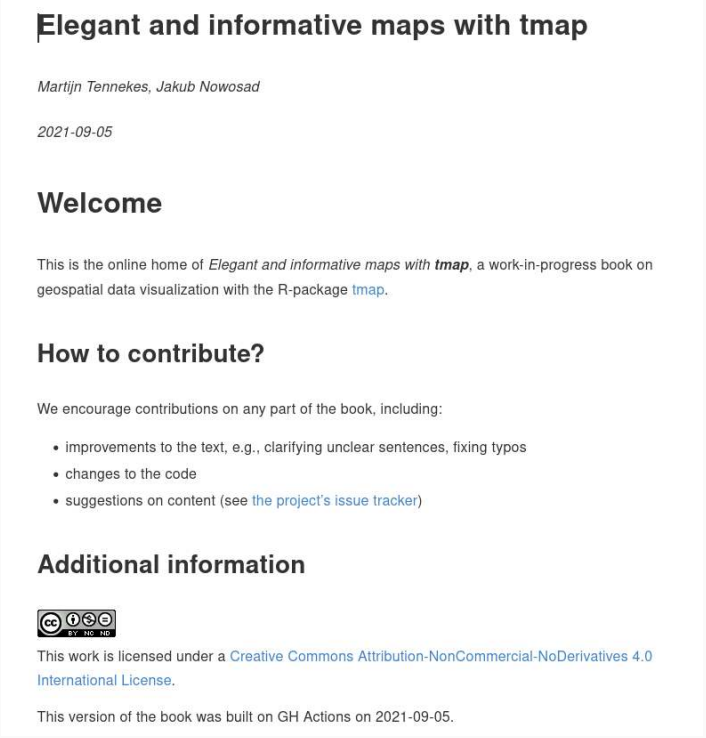
Geocomputation with R (2019)



Lovelace et al. (2019)

tmap

Elegant and informative maps with tmap



Elegant and informative maps with tmap

Martijn Tennekes, Jakub Nowosad

2021-09-05

Welcome

This is the online home of *Elegant and informative maps with* ***tmap***, a work-in-progress book on geospatial data visualization with the R-package tmap.

How to contribute?

We encourage contributions on any part of the book, including:

- improvements to the text, e.g., clarifying unclear sentences, fixing typos
- changes to the code
- suggestions on content (see the project's issue tracker)

Additional information

This version of the book was built on GH Actions on 2021-09-05.

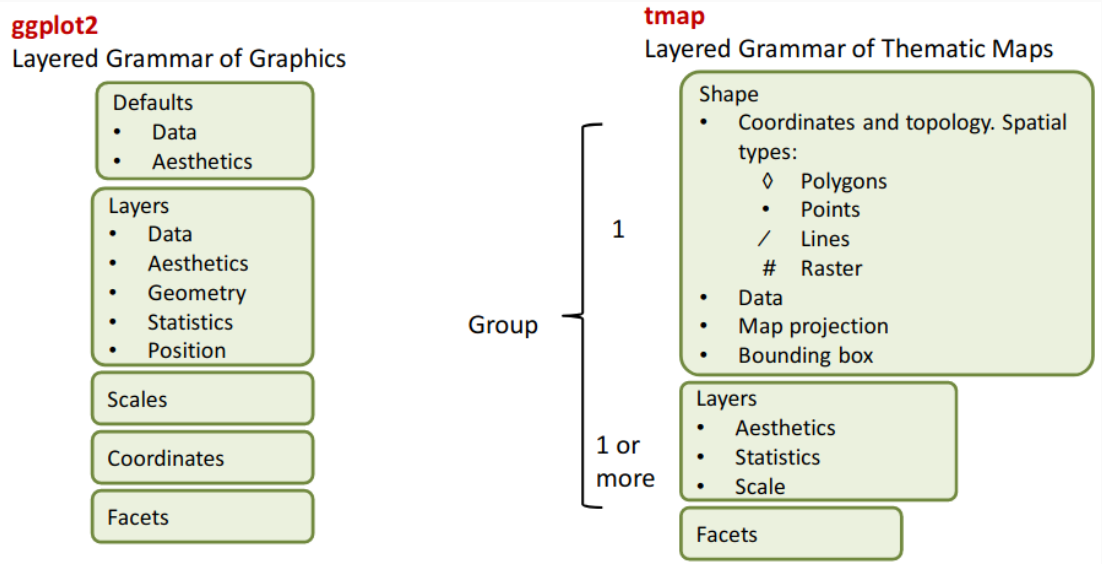
Tennekes & Nowosad (inacabado)

3. Mapas estáticos

tmap

	(Spatial) data	Layers (geometry, mapping, and scaling)	Small multiples	Layout
ggplot2	ggplot(...) +	geom_...(...) + scale_...(...) +	facet_wrap(...) +	theme(...)
tmap	tm_shape(...) +	tm_...(...) +	tm_facets(...) +	tm_layout(...)

Implemented: tm_polygons, tm_symbols, tm_lines, tm_raster, tm_text, tm_fill, tm_borders, tm_bubbles, tm_squares, tm_dots, tm_rgb, tm_markers, tm_iso



3. Mapas estáticos

tmap

```
# poligonos
tm_shape(biomas) +
  tm_polygons()
```

3. Mapas estáticos

tmap

```
# bordas
tm_shape(biomas) +
  tm_borders()
```

3. Mapas estáticos

tmap

```
# prenchimento
tm_shape(biomas) +
  tm_fill()
```

3. Mapas estáticos

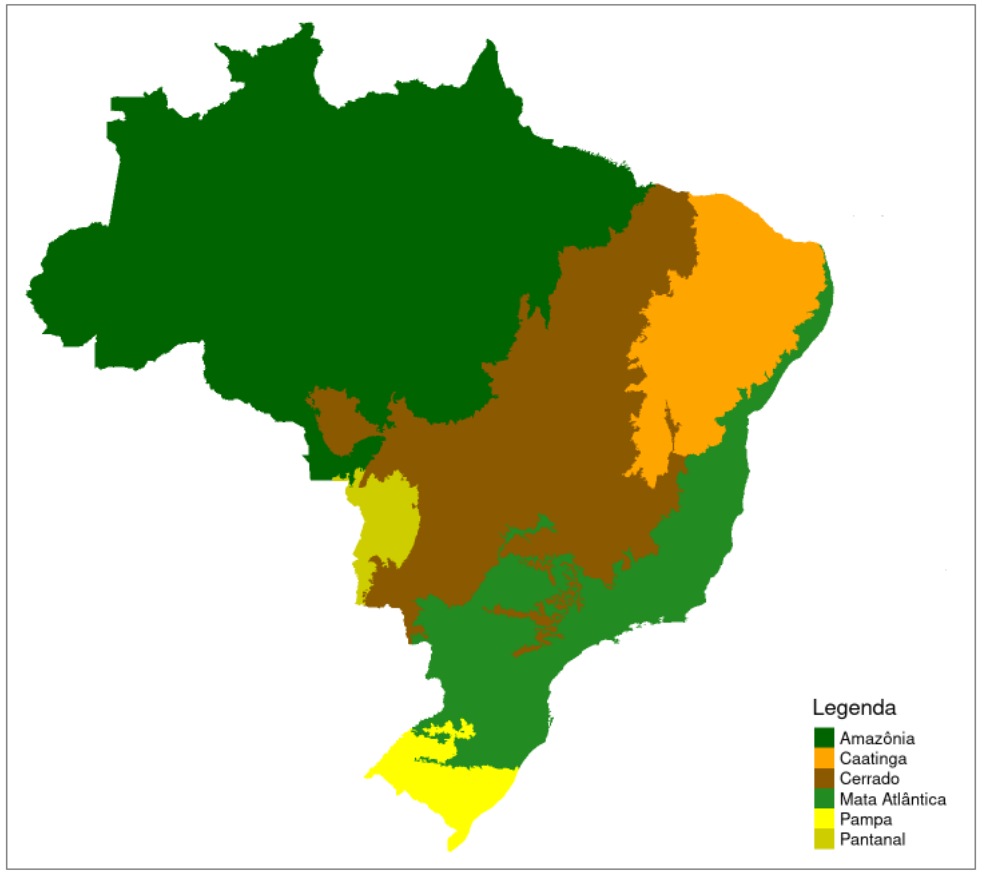
tmap

```
# cores
tm_shape(biomas) +
  tm_fill(col = "name_biome", title = "Legenda")
```

3. Mapas estáticos

tmap

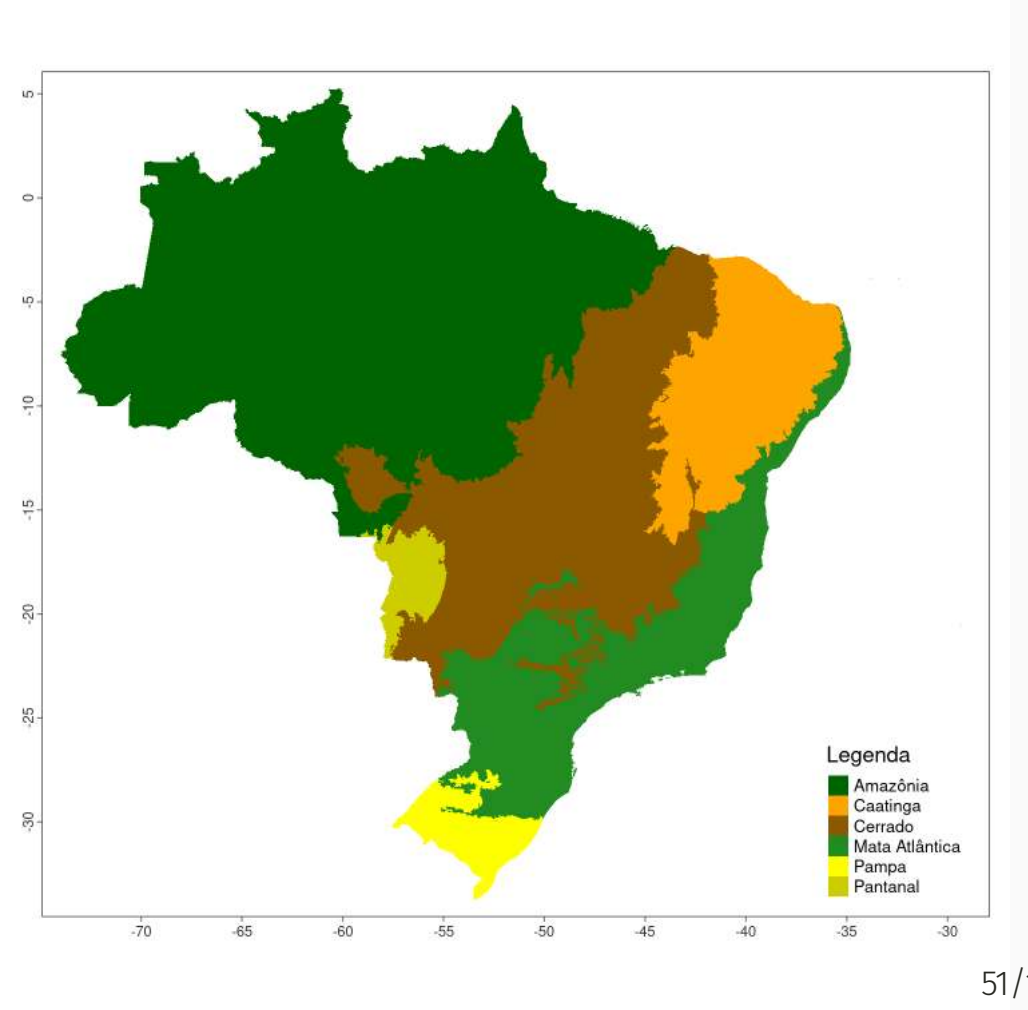
```
# definir cores
tm_shape(biomas) +
  tm_fill(col = "name_biome",
          pal = c("darkgreen", "orange", "orange4",
          title = "Legenda")
```



3. Mapas estáticos

tmap

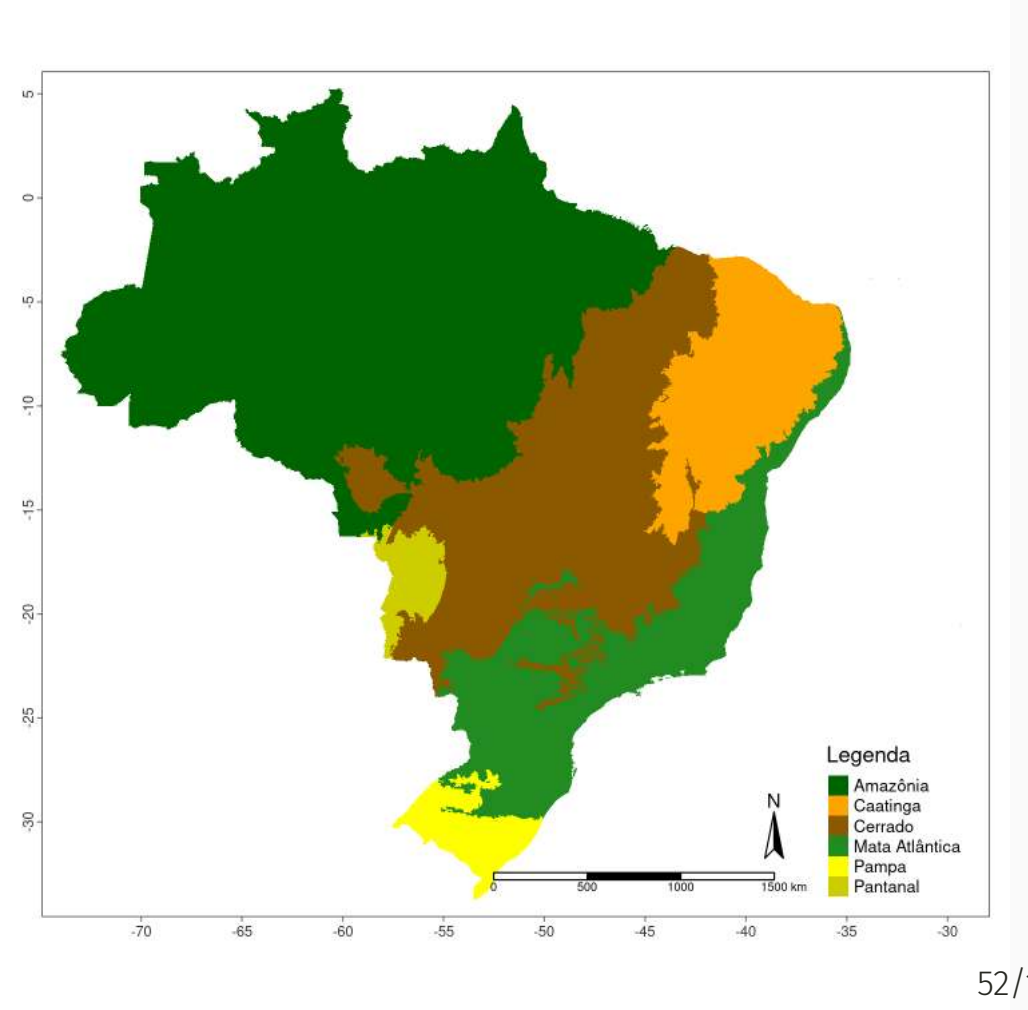
```
# gride coordenadas
tm_shape(biomas) +
  tm_fill(col = "name_biome",
          pal = c("darkgreen", "orange", "orange4",
          title = "Legenda") +
  tm_grid(lines = FALSE,
          labels.format = list(big.mark = ""),
          labels.rot = c(0, 90))
```



3. Mapas estáticos

tmap

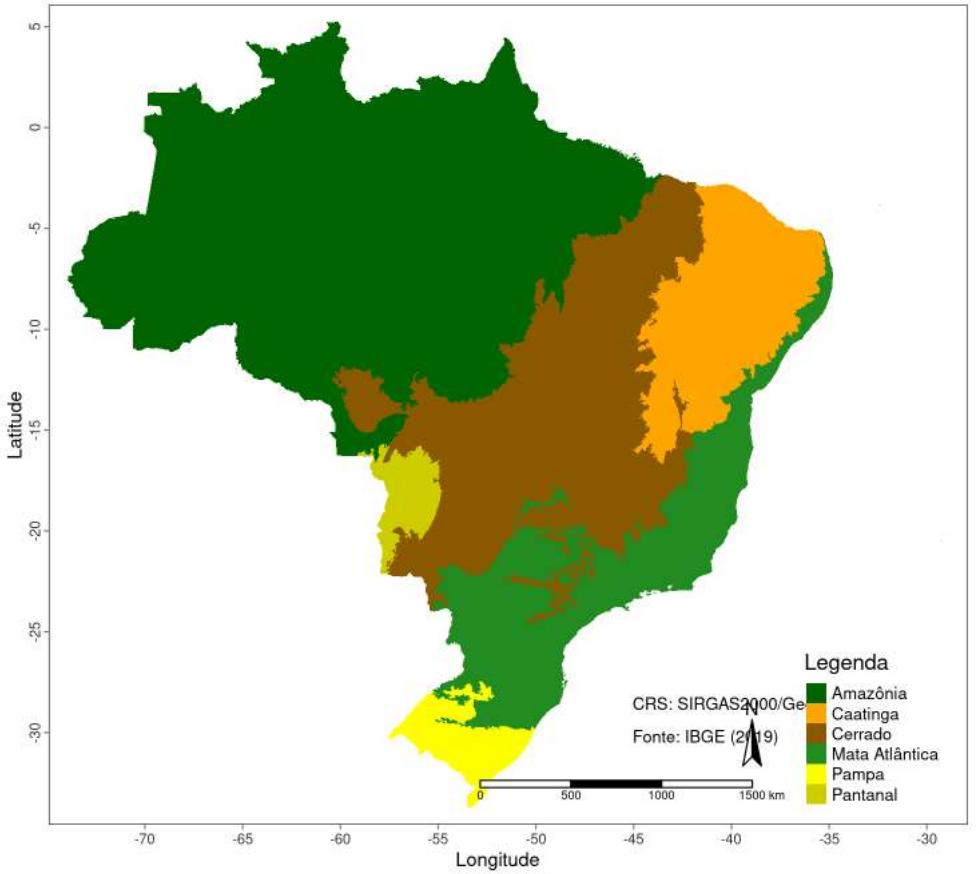
```
# barra de escala e norte
tm_shape(biomas) +
  tm_fill(col = "name_biome",
          pal = c("darkgreen", "orange", "orange4",
          title = "Legenda") +
  tm_grid(lines = FALSE,
          labels.format = list(big.mark = ""),
          labels.rot = c(0, 90)) +
  tm_compass() +
  tm_scale_bar()
```



3. Mapas estáticos

tmap

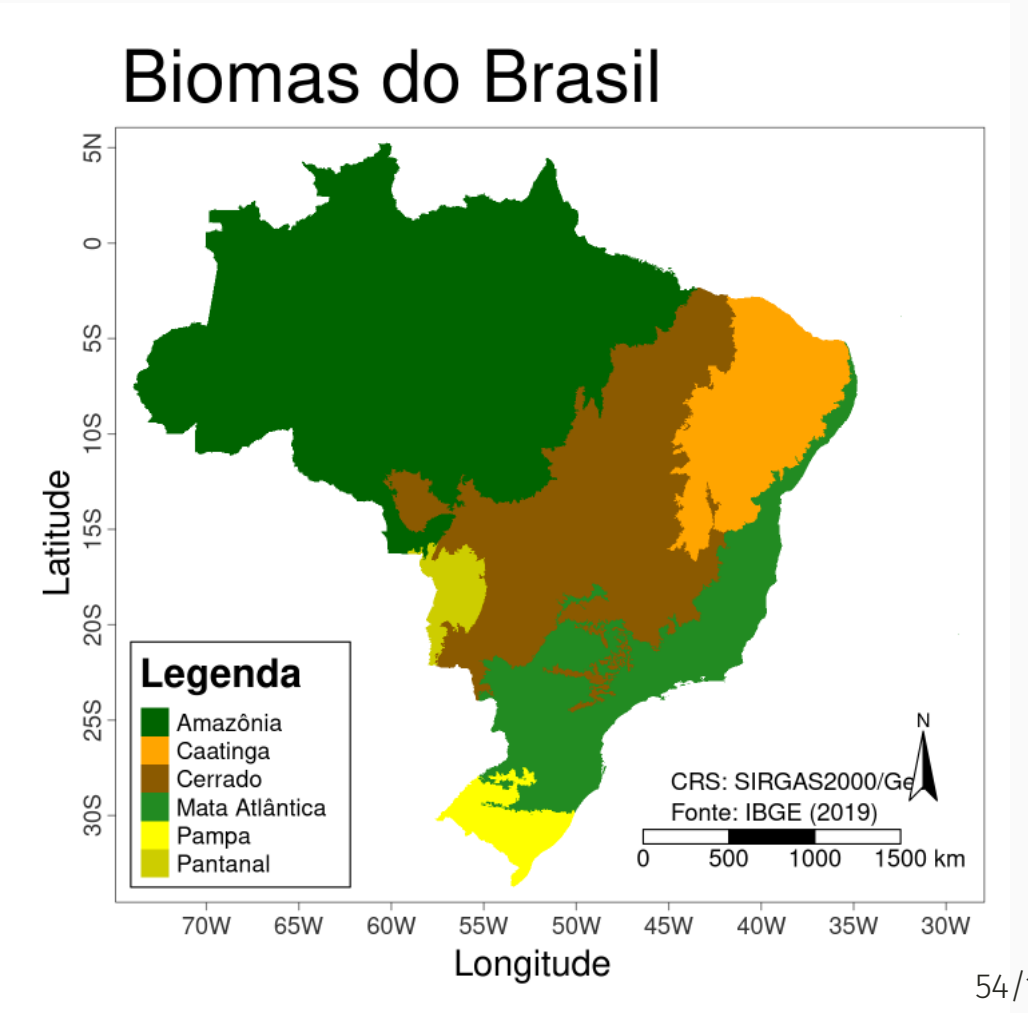
```
# nomes
tm_shape(biomas) +
  tm_fill(col = "name_biome",
          pal = c("darkgreen", "orange", "orange4",
          title = "Legenda") +
  tm_grid(lines = FALSE,
          labels.format = list(big.mark = ""),
          labels.rot = c(0, 90)) +
  tm_compass() +
  tm_scale_bar() +
  tm_xlab("Longitude") +
  tm_ylab("Latitude") +
  tm_credits("CRS: SIRGAS2000/Geo", position = c(.63
  tm_credits("Fonte: IBGE (2019)", position = c(.63,
```



3. Mapas estáticos

tmap

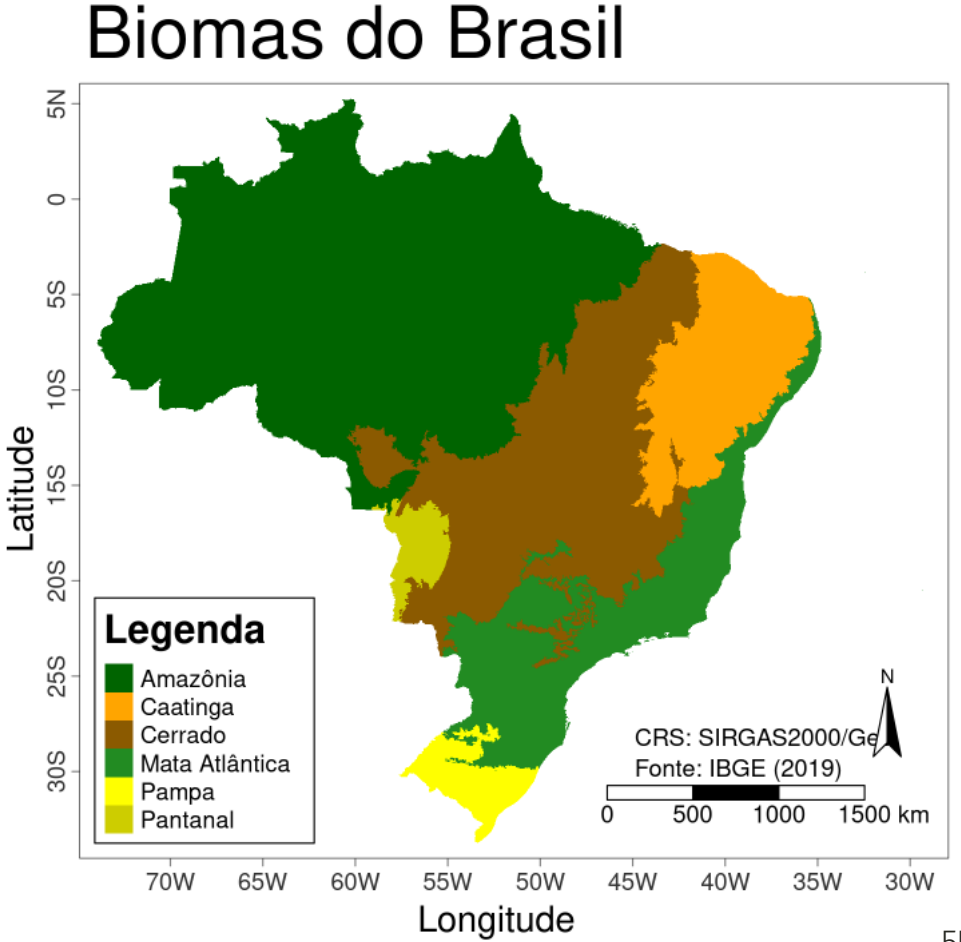
```
# titulos
tm_shape(biomas) +
  tm_fill(col = "name_biome",
          pal = c("darkgreen", "orange", "orange4",
          title = "Legenda") +
  tm_graticules(lines = FALSE,
                labels.format = list(big.mark = ""),
                labels.rot = c(0, 90),
                labels.size = 1) +
  tm_compass(size = 3) +
  tm_scale_bar(size = 1) +
  tm_xlab("Longitude", size = 1.5) +
  tm_ylab("Latitude", size = 1.5) +
  tm_credits("CRS: SIRGAS2000/Geo", position = c(.63
  tm_credits("Fonte: IBGE (2019)", position = c(.63,
  tm_layout(main.title = "Biomas do Brasil",
            main.title.position = c(.1, .95),
            main.title.size = 3,
            title.fontface = "bold",
            legend.frame = TRUE,
            legend.position = c("left", "bottom"),
```



3. Mapas estáticos

tmap

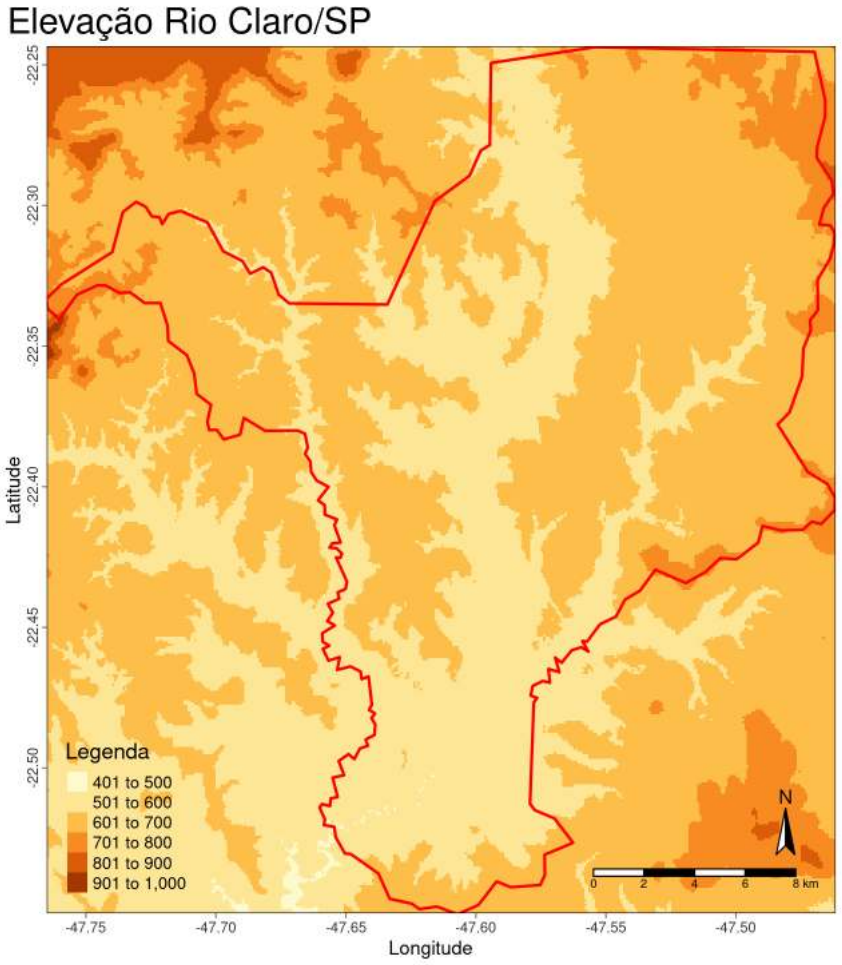
```
# atribuicao
map_biomas_tmap ← tm_shape(biomas) +
  tm_fill(col = "name_biome",
          pal = c("darkgreen", "orange", "orange4",
          title = "Legenda") +
  tm_graticules(lines = FALSE,
                labels.format = list(big.mark = ""),
                labels.rot = c(0, 90),
                labels.size = 1) +
  tm_compass(size = 3) +
  tm_scale_bar(size = 1) +
  tm_xlab("Longitude", size = 1.5) +
  tm_ylab("Latitude", size = 1.5) +
  tm_credits("CRS: SIRGAS2000/Geo", position = c(.63
  tm_credits("Fonte: IBGE (2019)", position = c(.63,
  tm_layout(main.title = "Biomas do Brasil",
            main.title.position = c(.1, .95),
            main.title.size = 3,
            title.fontface = "bold",
            legend.frame = TRUE,
            legend.position = c("left", "bottom"),
```



3. Mapas estáticos

tmap

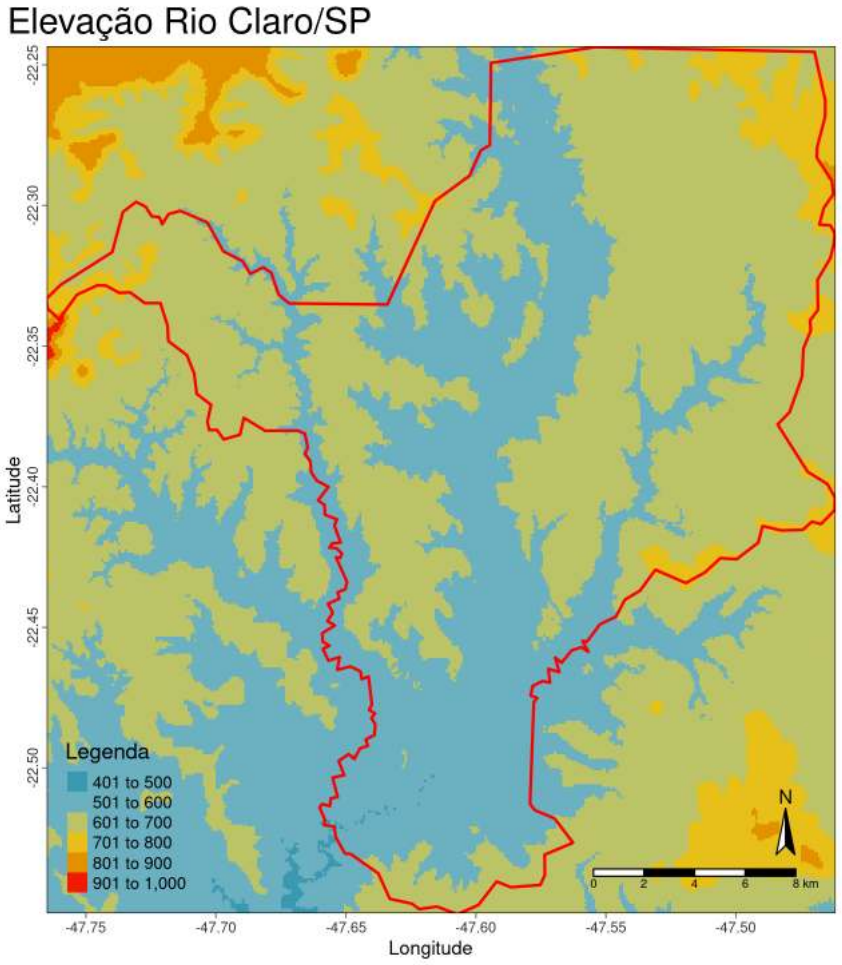
```
# plot
map_dem_rc_tmap ← tm_shape(dem_rc) +
  tm_raster(title = "Legenda") +
  tm_shape(rc_2020) +
  tm_borders(col = "red", lwd = 2) +
  tm_grid(lines = FALSE,
          labels.format = list(big.mark = ""),
          labels.rot = c(0, 90)) +
  tm_compass() +
  tm_scale_bar() +
  tm_xlab("Longitude") +
  tm_ylab("Latitude") +
  tm_layout(legend.position = c("left", "bottom"),
            main.title = "Elevação Rio Claro/SP")
map_dem_rc_tmap
```



3. Mapas estáticos

tmap

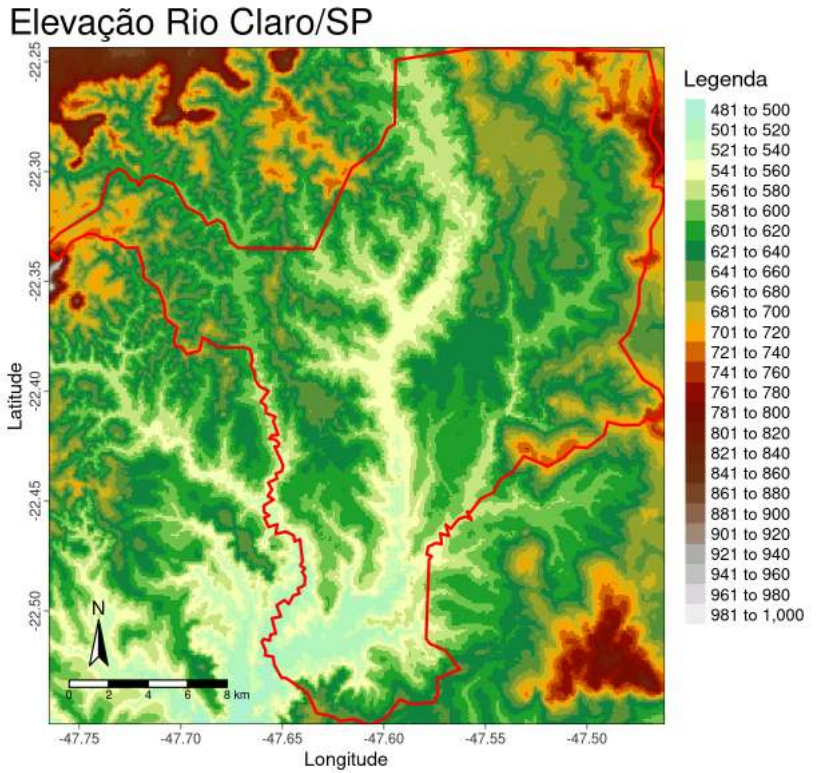
```
# plot
map_dem_rc_tmap ← tm_shape(dem_rc) +
  tm_raster(pal = wesanderson::wes_palette("Zissou1"
  tm_shape(rc_2020) +
  tm_borders(col = "red", lwd = 2) +
  tm_grid(lines = FALSE,
          labels.format = list(big.mark = ""),
          labels.rot = c(0, 90)) +
  tm_compass() +
  tm_scale_bar() +
  tm_xlab("Longitude") + tm_ylab("Latitude") +
  tm_layout(legend.position = c("left", "bottom"),
            main.title = "Elevação Rio Claro/SP")
map_dem_rc_tmap
```



3. Mapas estáticos

tmap

```
# plot
map_dem_rc_tmap ← tm_shape(dem_rc) +
  tm_raster(pal = cptcity::cpt(pal = "gmt_GMT_dem4")
  tm_shape(rc_2020) +
  tm_borders(col = "red", lwd = 2) +
  tm_grid(lines = FALSE,
          labels.format = list(big.mark = ""),
          labels.rot = c(0, 90)) +
  tm_compass(position = c("left", "bottom")) +
  tm_scale_bar(position = c("left", "bottom")) +
  tm_xlab("Longitude") +
  tm_ylab("Latitude") +
  tm_layout(legend.outside = TRUE,
            main.title = "Elevação Rio Claro/SP")
map_dem_rc_tmap
```



3. Mapas estáticos

tmap

```
# exportar
tmap::tmap_save(tm = map_biomas_tmap,
                filename = here::here("03_dados", "mapas", "mapa_biomas_tmap.png"),
                width = 20,
                height = 20,
                units = "cm",
                dpi = 300)
```

3. Mapas estáticos

tmap

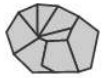
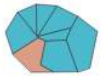
```
# exportar
tmap::tmap_save(tm = map_dem_rc_tmap,
                filename = here::here("03_dados", "mapas", "mapa_dem_rc_tmap.png"),
                width = 20,
                height = 20,
                units = "cm",
                dpi = 300)
```

mapsf

3. Mapas estáticos

mapsf

```
# pacote
# install.packages("mapsf")
library(mapsf)

# sintaxe
mf_map(x, var, type)
```

map type	type	data	polygon	point	line
Base Map	`"base"`	geometry			
Proportional Symbols	`"prop"`	stock			
Typology	`"typo"`	category			
Choropleth	`"choro"`	ratio			
Graduated Symbols	`"grad"`	stock			

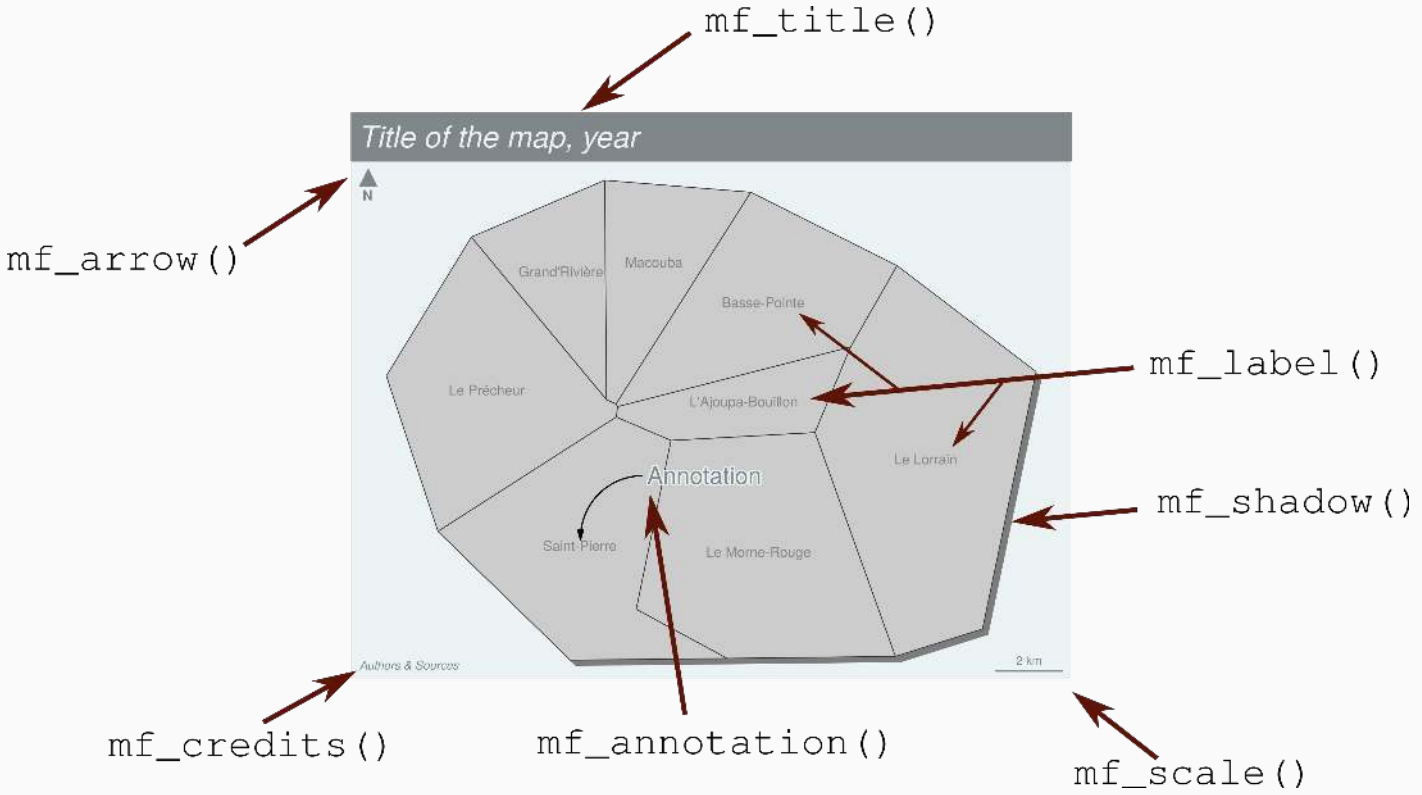
Symbols	`"symb"`	category	 
Proportional Symbols + Typo	`"prop_typo"`	stock & category	 
Proportional Symbols + Choro	`"prop_choro"`	stock & ratio	 
Symbols + Choro	`"symb_choro"`	category & ratio	 

[mapsf]

3. Mapas estáticos

mapsf

Layout



mapsf

3. Mapas estáticos

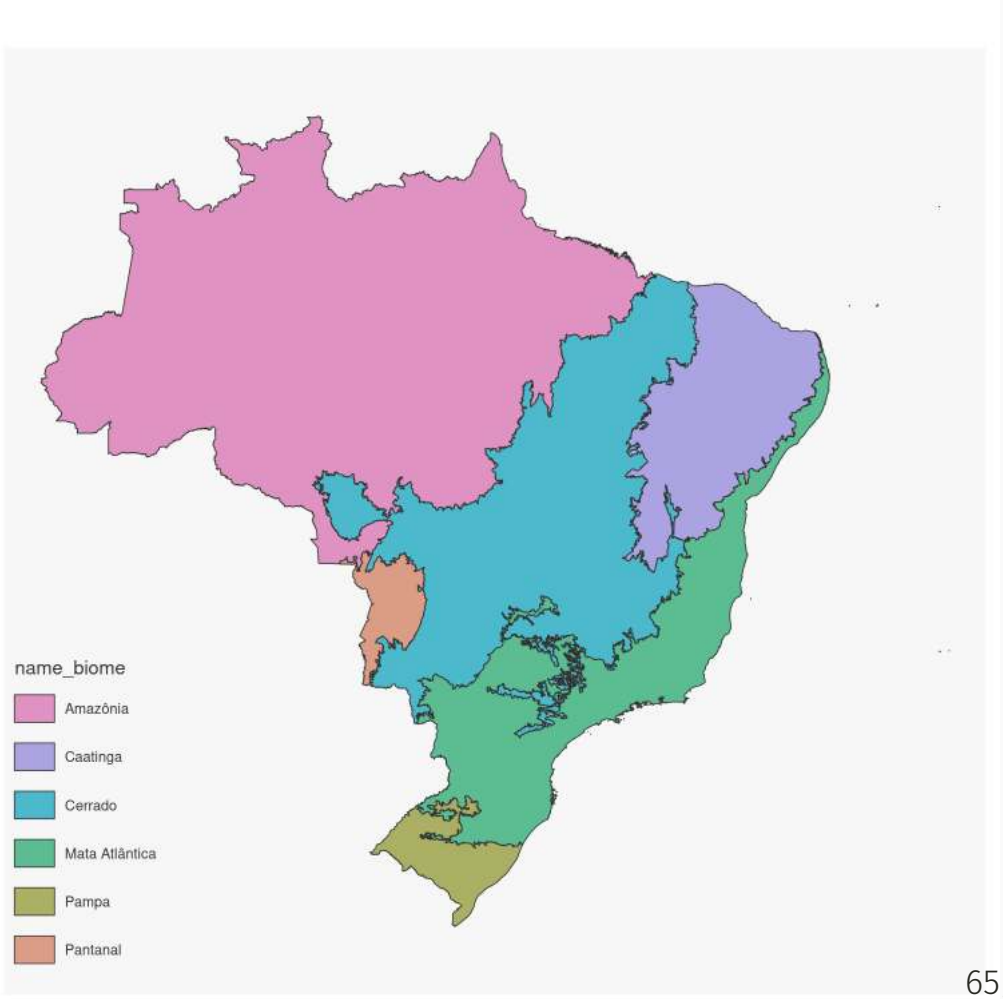
mapsf

```
# plot
mf_map(x = biomas)
```

3. Mapas estáticos

mapsf

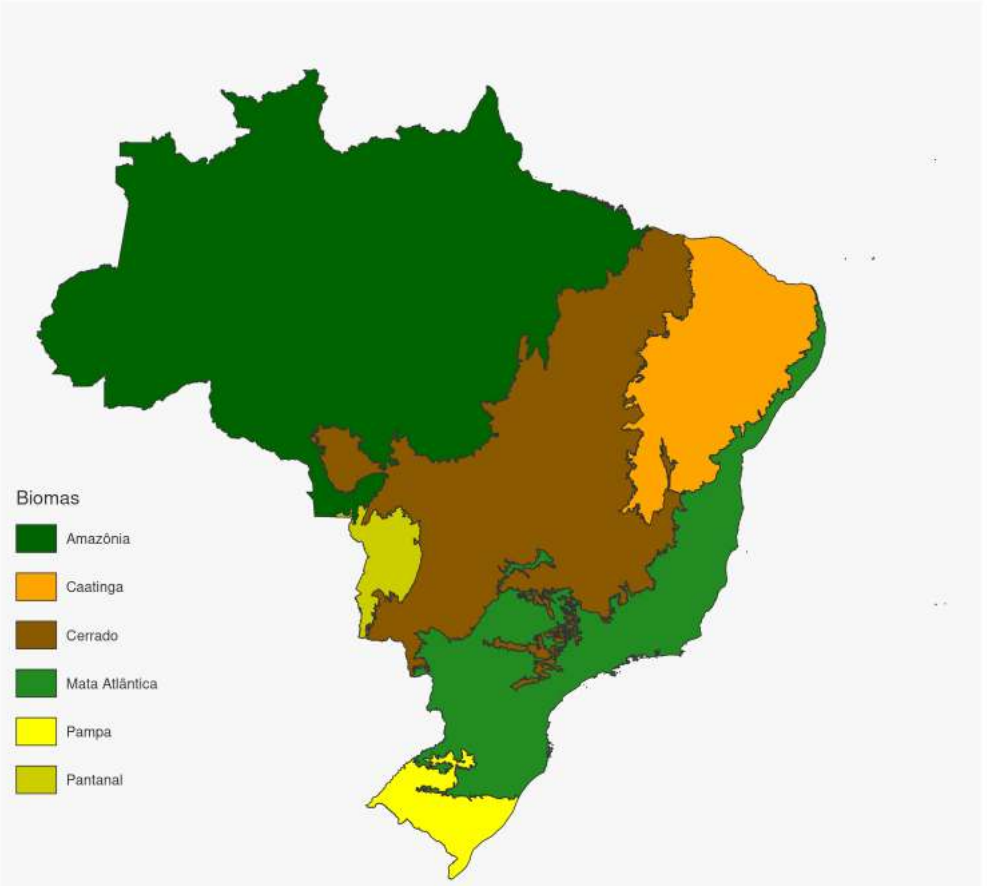
```
# plot
mf_map(x = biomas, var = "name_biome", type = "typo"
```



3. Mapas estáticos

mapsf

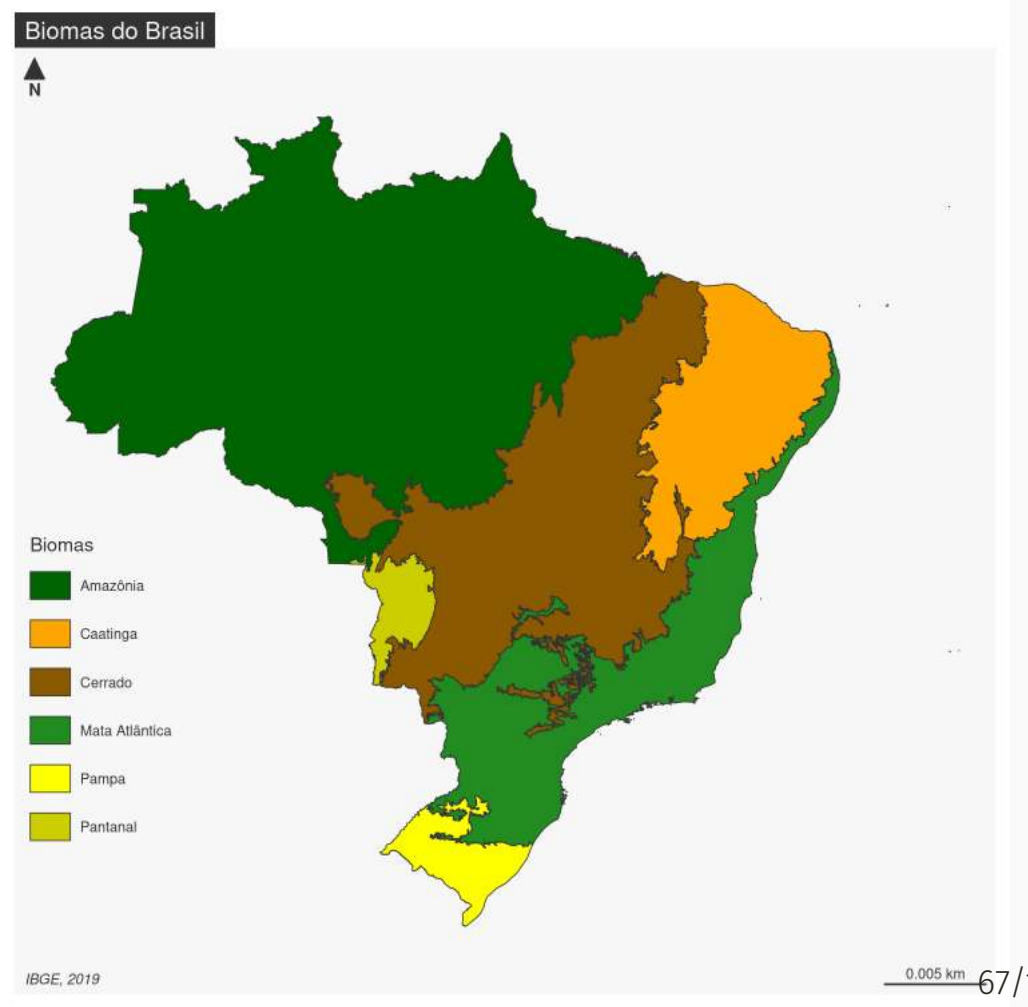
```
# plot
mf_map(x = biomas, var = "name_biome", type = "typo"
       pal = c("darkgreen", "orange", "orange4",
               "forestgreen", "yellow", "yellow3"),
       leg_title = "Biomas",
       leg_pos = c(-75, -15))
```



3. Mapas estáticos

mapsf

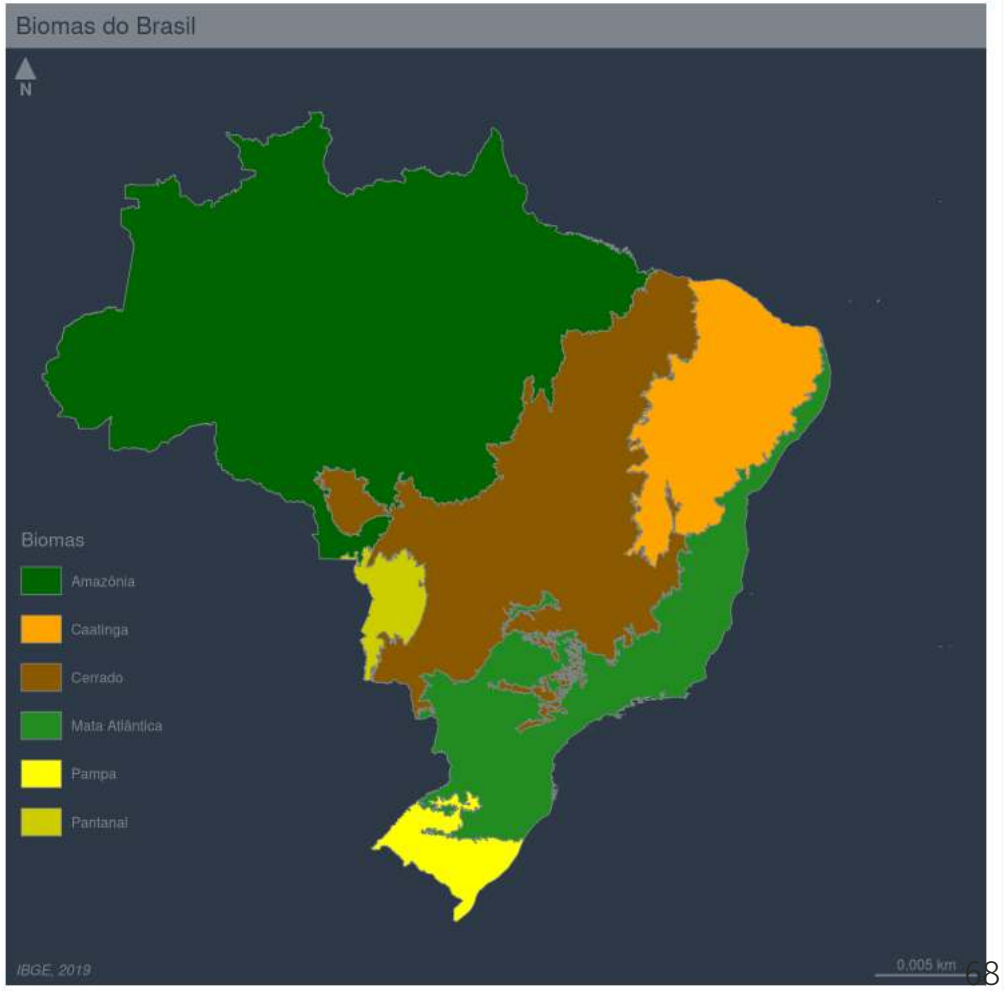
```
# plot
mf_map(x = biomas, var = "name_biome", type = "typo"
       pal = c("darkgreen", "orange", "orange4",
               "forestgreen", "yellow", "yellow3"),
       leg_title = "Biomas",
       leg_pos = c(-75, -15))
mf_title("Biomas do Brasil")
mf_credits("IBGE, 2019")
mf_scale()
mf_arrow()
```



3. Mapas estáticos

mapsf

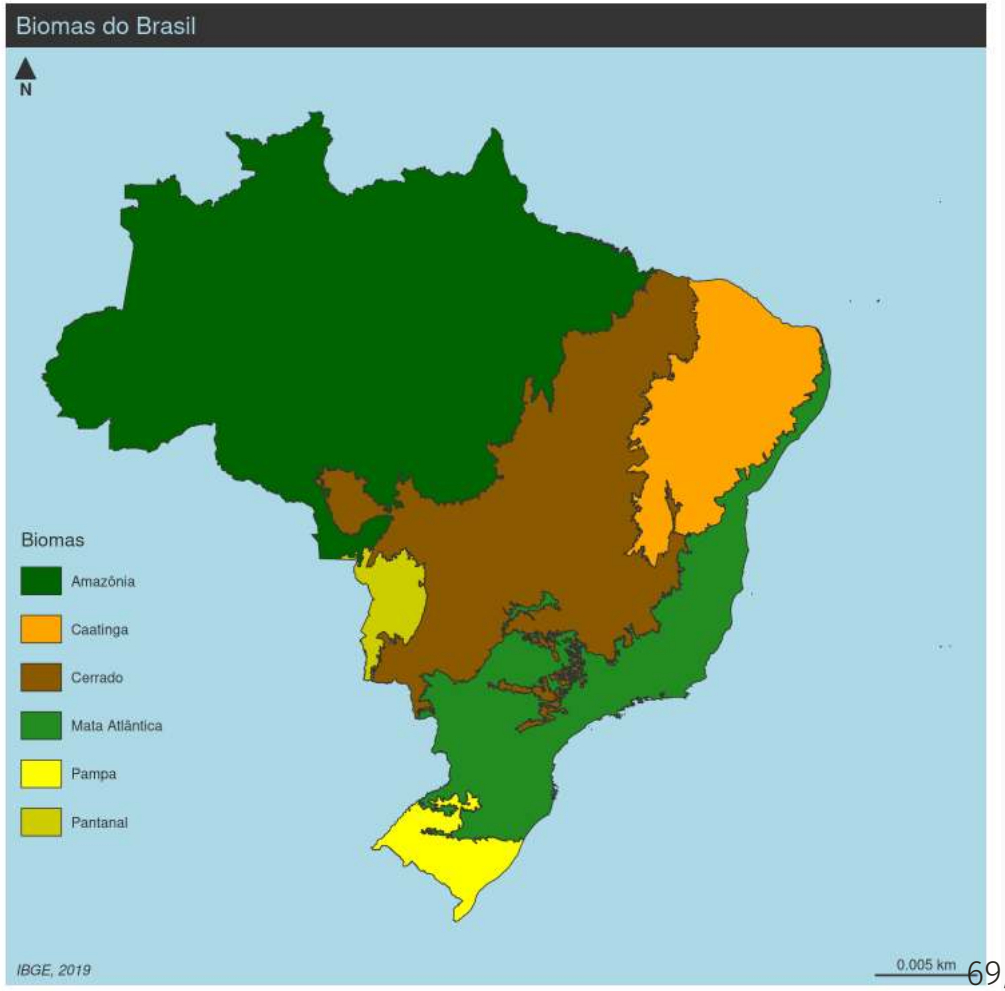
```
# plot
mf_init(x = biomas, theme = "dark")
mf_map(x = biomas, var = "name_biome", type = "typo"
       pal = c("darkgreen", "orange", "orange4",
               "forestgreen", "yellow", "yellow3"),
       leg_title = "Biomas",
       leg_pos = c(-75, -15),
       add = TRUE)
mf_title("Biomas do Brasil")
mf_credits("IBGE, 2019")
mf_scale()
mf_arrow()
```



3. Mapas estáticos

mapsf

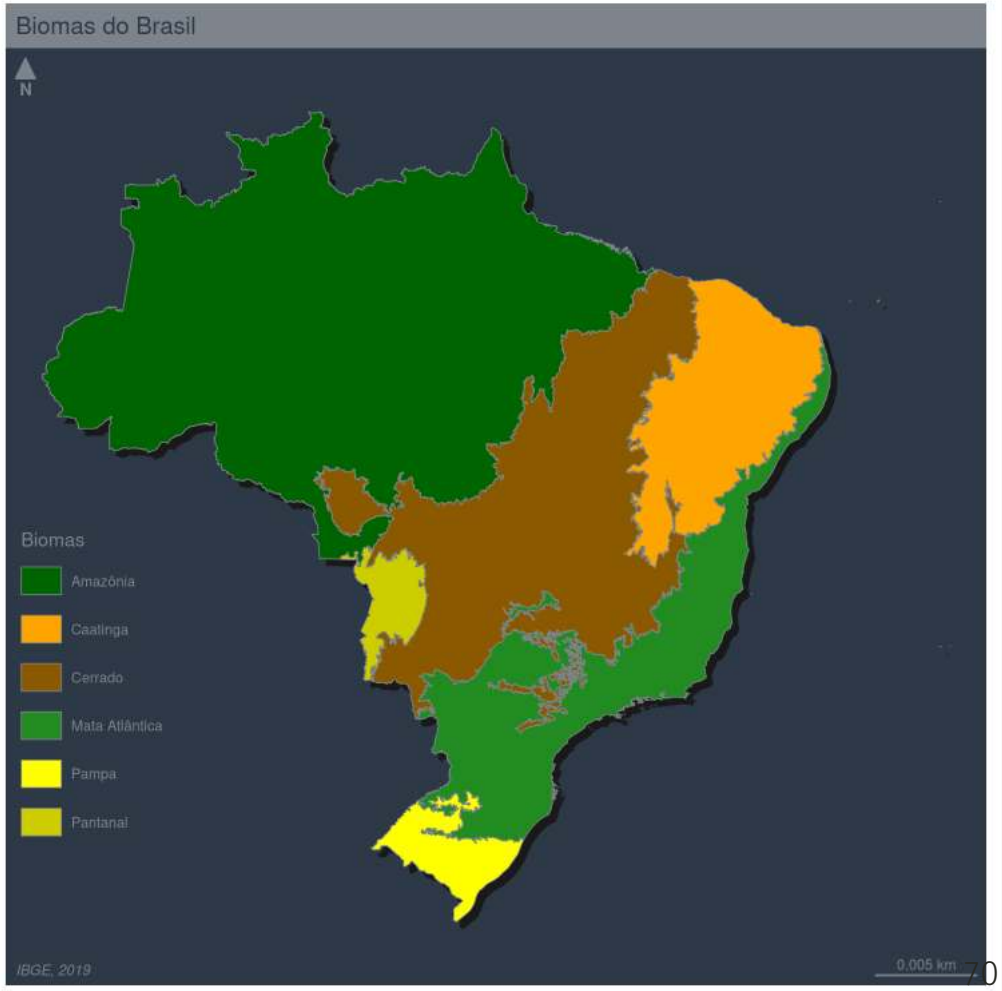
```
# plot
mf_theme(bg = "lightblue", fg = "gray20")
mf_map(x = biomas, var = "name_biome", type = "typo"
       pal = c("darkgreen", "orange", "orange4",
               "forestgreen", "yellow", "yellow3"),
       leg_title = "Biomas",
       leg_pos = c(-75, -15))
mf_title("Biomas do Brasil")
mf_credits("IBGE, 2019")
mf_scale()
mf_arrow()
```



3. Mapas estáticos

mapsf

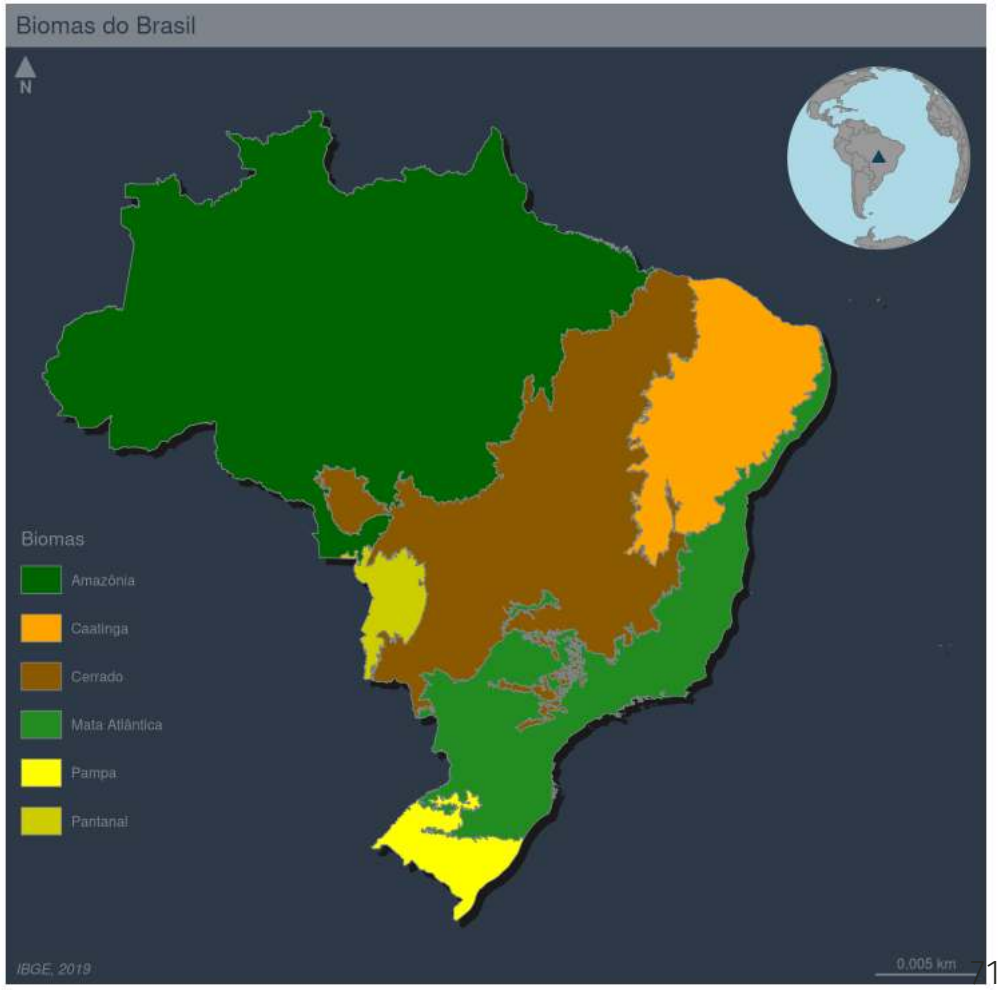
```
# plot
mf_init(x = biomas, theme = "dark")
mf_shadow(x = biomas, col = "gray10", add = TRUE)
mf_map(x = biomas, var = "name_biome", type = "typo"
       pal = c("darkgreen", "orange", "orange4",
               "forestgreen", "yellow", "yellow3"),
       leg_title = "Biomas",
       leg_pos = c(-75, -15),
       add = TRUE)
mf_title("Biomas do Brasil")
mf_credits("IBGE, 2019")
mf_scale()
mf_arrow()
```



3. Mapas estáticos

mapsf

```
# plot
mf_init(x = biomas, theme = "dark")
mf_shadow(x = biomas, col = "gray10", add = TRUE)
mf_map(x = biomas, var = "name_biome", type = "typo"
       pal = c("darkgreen", "orange", "orange4",
               "forestgreen", "yellow", "yellow3"),
       leg_title = "Biomas",
       leg_pos = c(-75, -15),
       add = TRUE)
mf_inset_on(x = "worldmap", pos = "topright")
mf_worldmap(biomas, col = "#0E3F5C")
mf_inset_off()
mf_title("Biomas do Brasil")
mf_credits("IBGE, 2019")
mf_scale()
mf_arrow()
```



3. Mapas estáticos

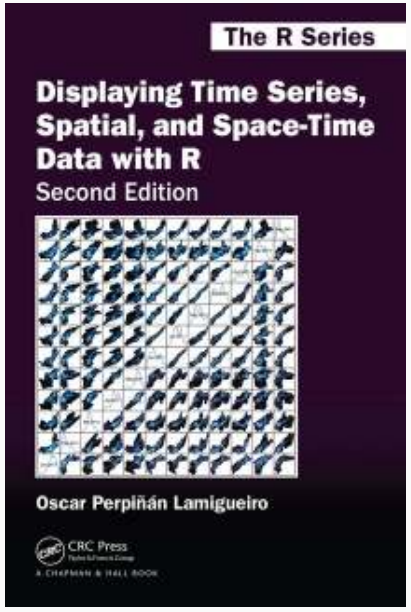
mapsf

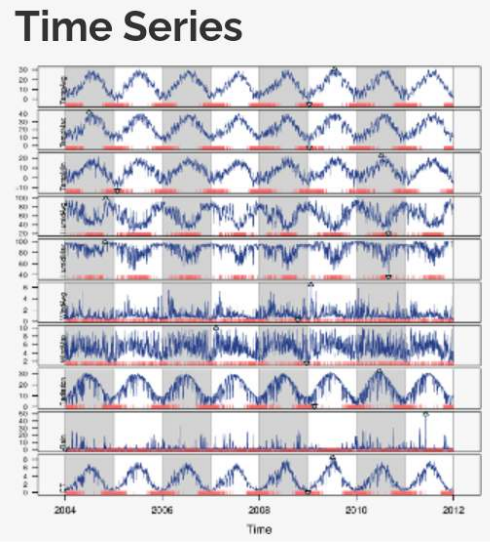
```
# export
mf_export(x = biomas,
          filename = here::here("03_dados", "mapas", "mapa_biomas_mapsf.png"),
          wi = 20, he = 20, un = "cm", res = 300)
mf_init(x = biomas, theme = "dark")
mf_shadow(x = biomas, col = "gray10", add = TRUE)
mf_map(x = biomas, var = "name_biome", type = "typo",
       pal = c("darkgreen", "orange", "orange4",
               "forestgreen", "yellow", "yellow3"),
       leg_title = "Biomas",
       leg_pos = c(-70, -17),
       add = TRUE)
mf_inset_on(x = "worldmap", pos = "topright")
mf_worldmap(biomas, col = "#0E3F5C")
mf_inset_off()
mf_title("Biomas do Brasil")
mf_credits("IBGE, 2019")
mf_scale()
mf_arrow()
dev.off()
```

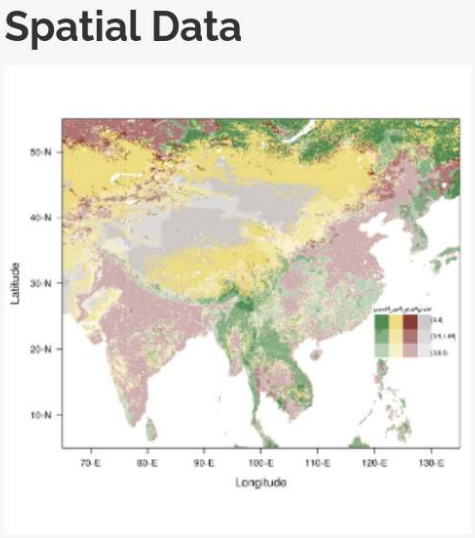
3. Mapas estáticos

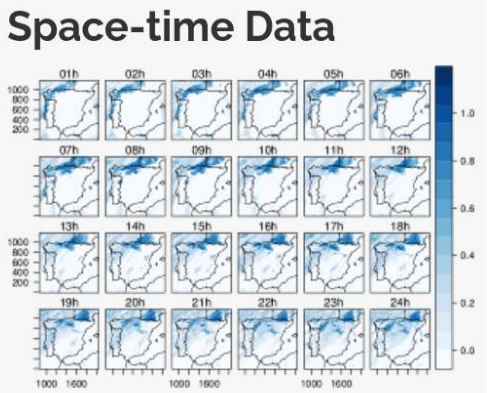
rasterVis

```
# pacote
# install.packages('rasterVis')
library(rasterVis)
```







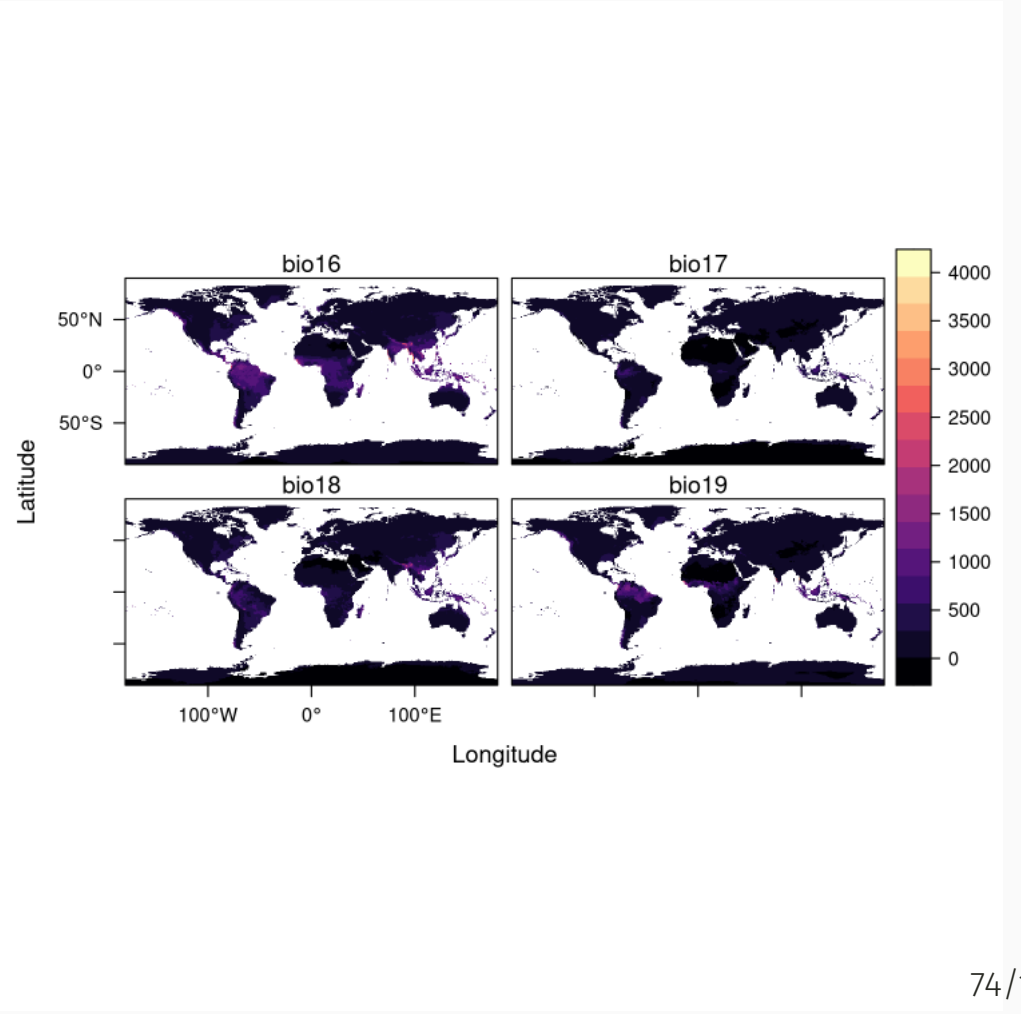


Lamigueiro (2018), Displaying time series, spatial and space-time data with R

3. Mapas estáticos

rasterVis

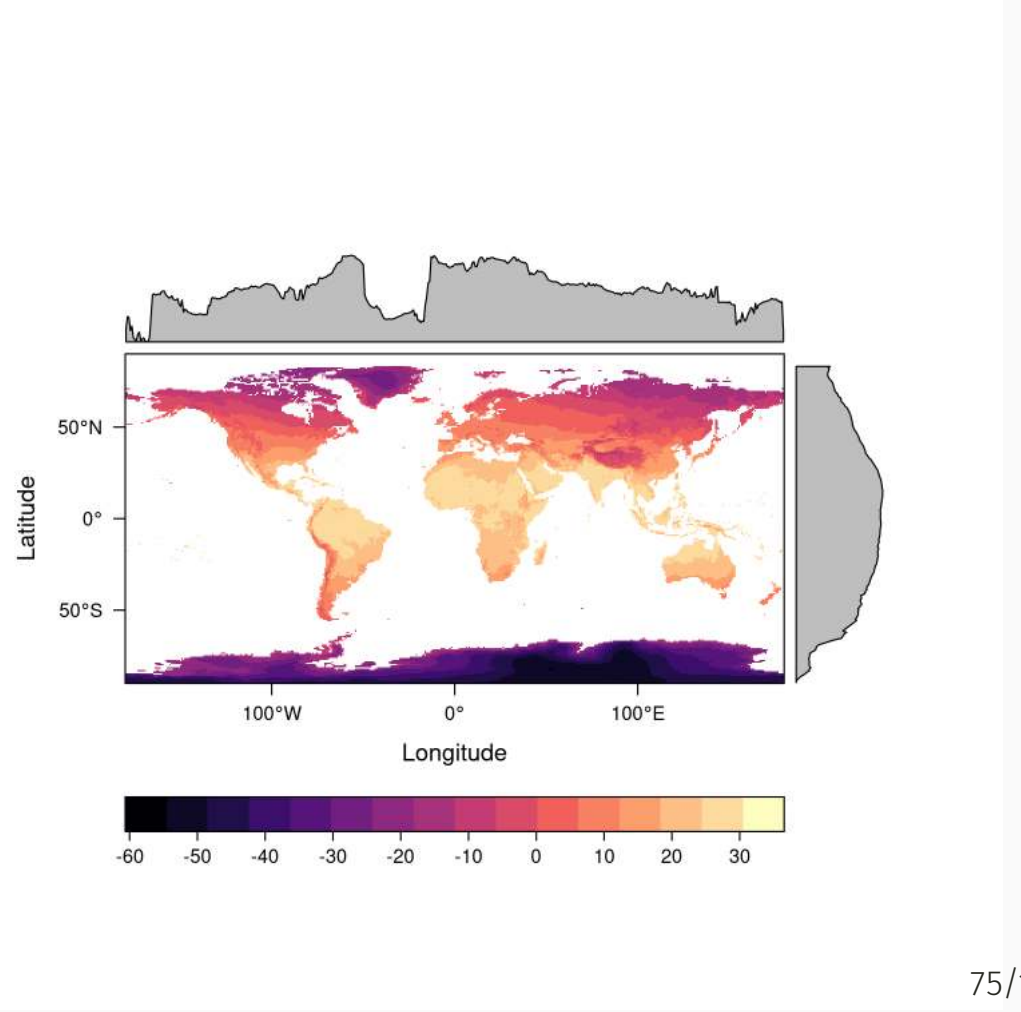
```
# plot
levelplot(bioclim[[8:11]])
```



3. Mapas estáticos

rasterVis

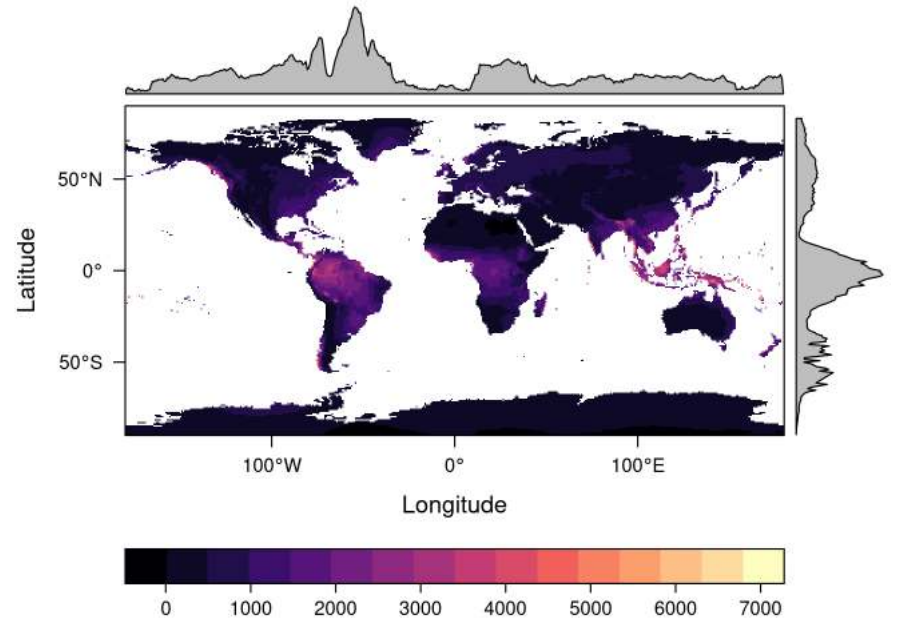
```
# plot
levelplot(bioclim,
          layers = 1,
          margin = list(FUN = "median"))
```



3. Mapas estáticos

rasterVis

```
# plot
levelplot(bioclim,
          layers = 4,
          margin = list(FUN = "median"))
```



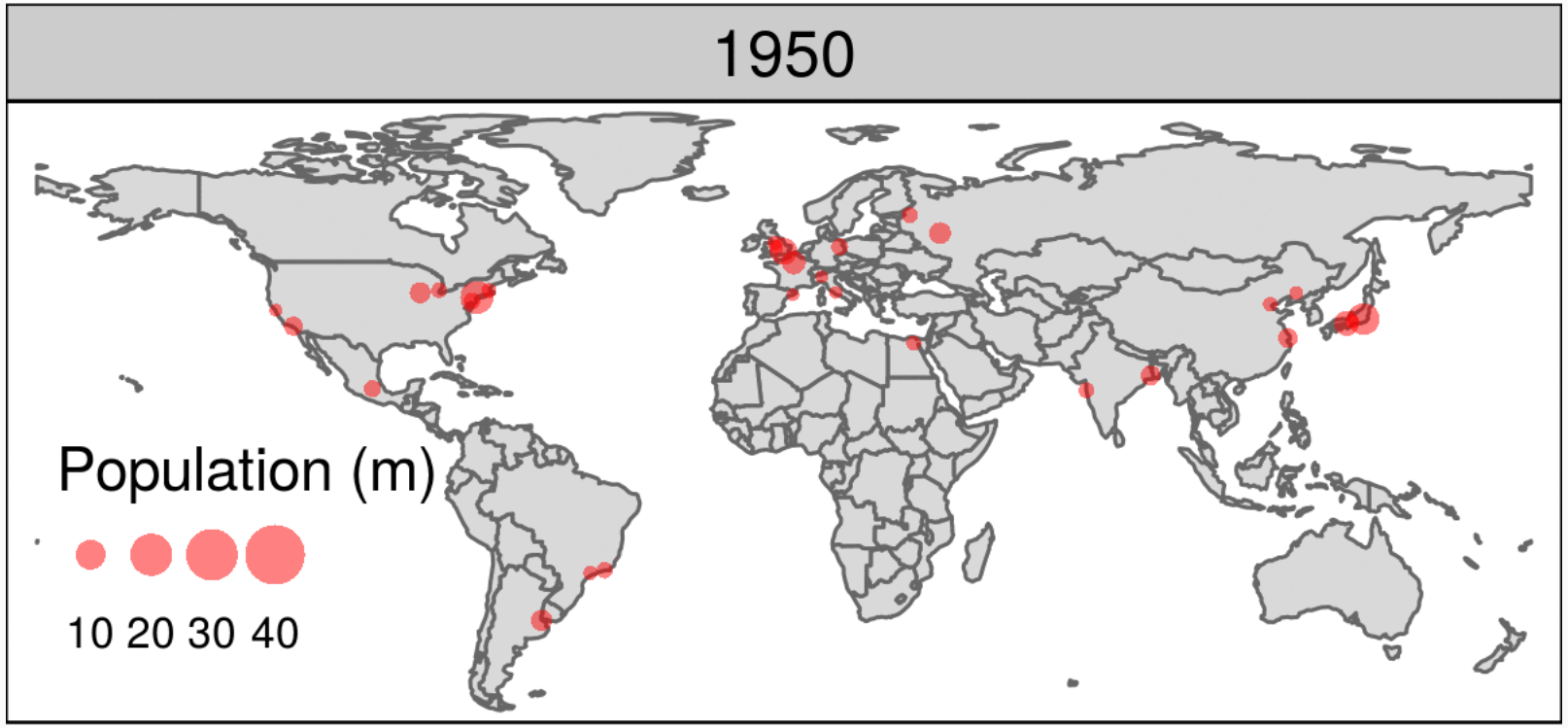
3. Mapas estáticos

rasterVis

```
# export
png(filename = here::here("03_dados", "mapas", "mapa_bioclim_rastervis.png"),
    wi = 20, he = 20, un = "cm", res = 300)
levelplot(bioclim,
          layers = 4,
          margin =
            list(FUN = "median"))
dev.off()
```

4. Mapas animados

Mapas animados são utilizados para entender a mudanças espaciais e/ou no tempo



4. Mapas animados

Mudança dos estados do Brasil ao longo do tempo

```
# dados
br_anos ← NULL
for(i in c(1872, 1900, 1911, 1920, 1933, 1940, 1950, 1960, 1970, 1980, 1991, 2001, 2010, 2020)){
  br_anos ← geobr::read_state(code_state = "all", year = i, showProgress = FALSE) %>%
    sf::st_geometry() %>%
    sf::st_as_sf() %>%
    dplyr::mutate(year = i) %>%
    dplyr::bind_rows(br_anos, .)
}
```

4. Mapas animados

Mudança dos estados do Brasil ao longo do tempo

```
# mapa facetado
map_brasil_tmap ← tm_shape(br_anos) +
  tm_polygons() +
  tm_facets(by = "year", nrow = 4)
map_brasil_tmap
```

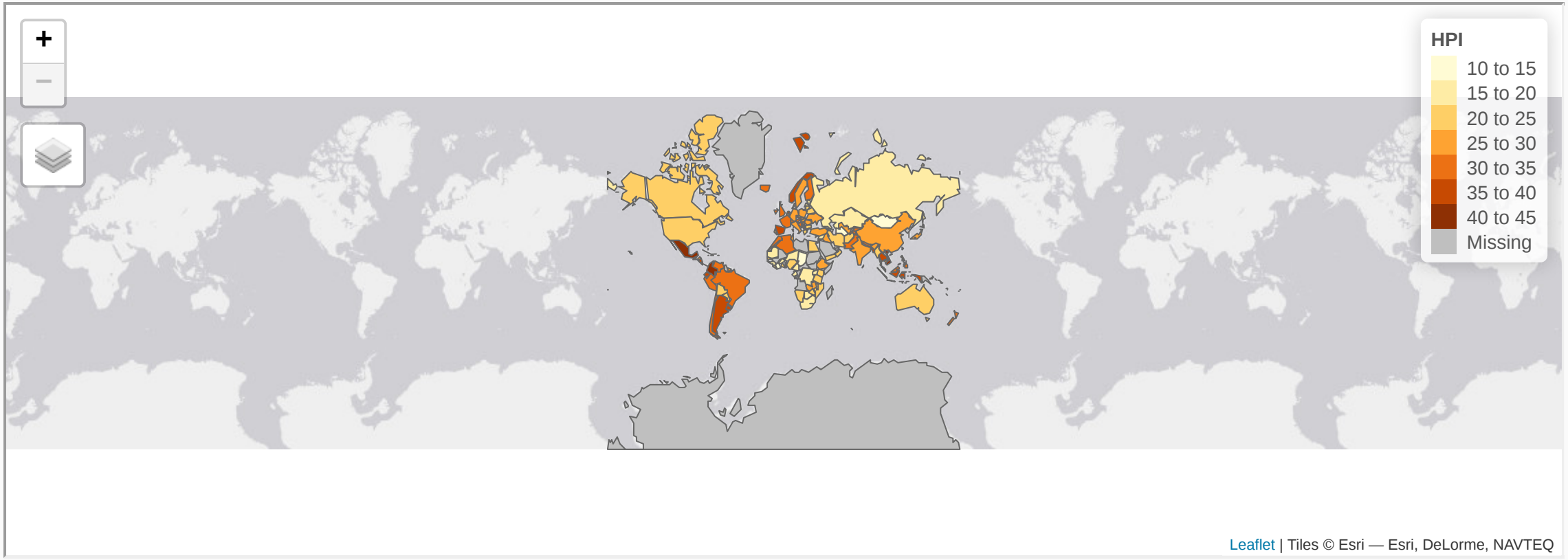
4. Mapas animados

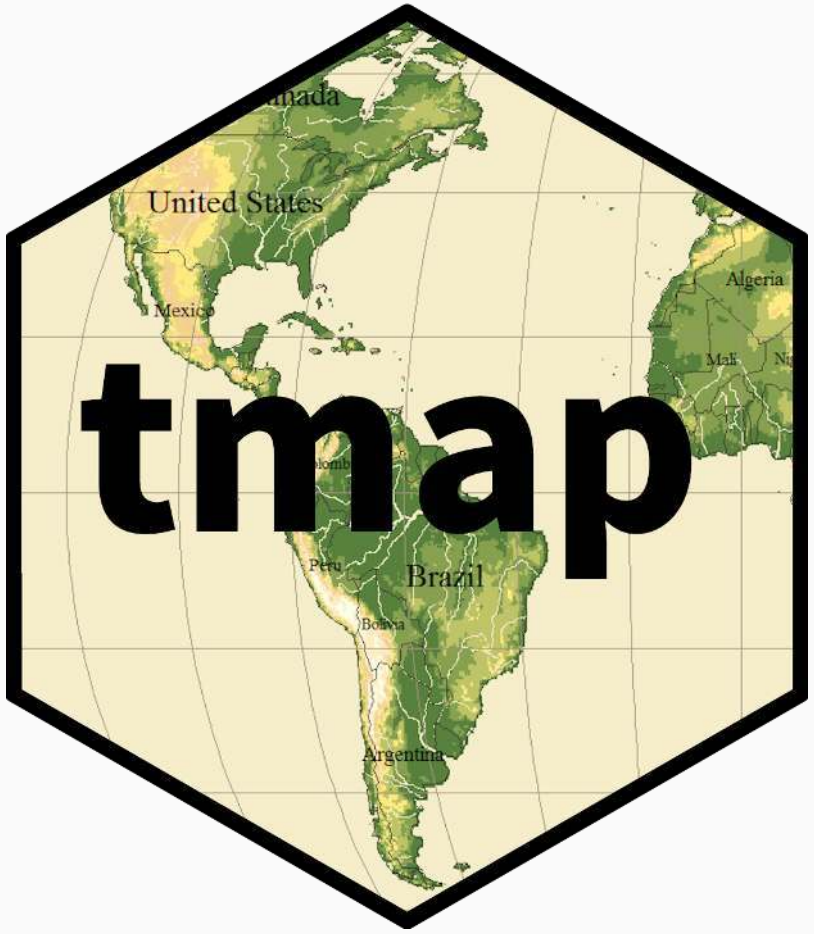
Mudança dos estados do Brasil ao longo do tempo

```
# exportar
tmap::tmap_animation(
  tm = map_brasil_tmap_ani,
  filename = here::here("03_dados",
                        "mapas",
                        "mapa_dem_rc_tmap_ani.gif"),
  delay = 30)
```

5. Mapas interativos

Mapas interativos são utilizados quando precisamos interagir




United States
Mexico
Algeria
Mali
Peru
Brazil
Bolivia
Argentina
tmap

tmap

5. Mapas interativos

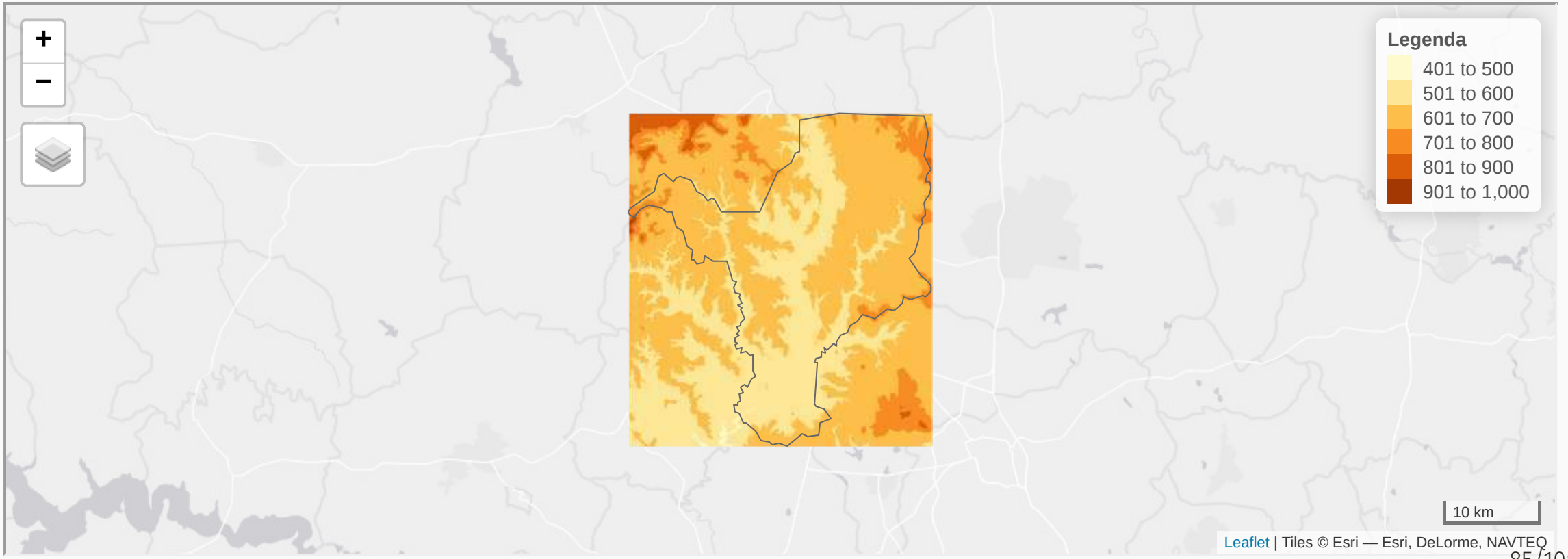
tmap

```
# mudar o modo de exibicao do tmap
tmap::tmap_mode(mode = "view")

map_biomas_tmap_int ← map_biomas_tmap
map_biomas_tmap_int
```

5. Mapas interativos

tmap

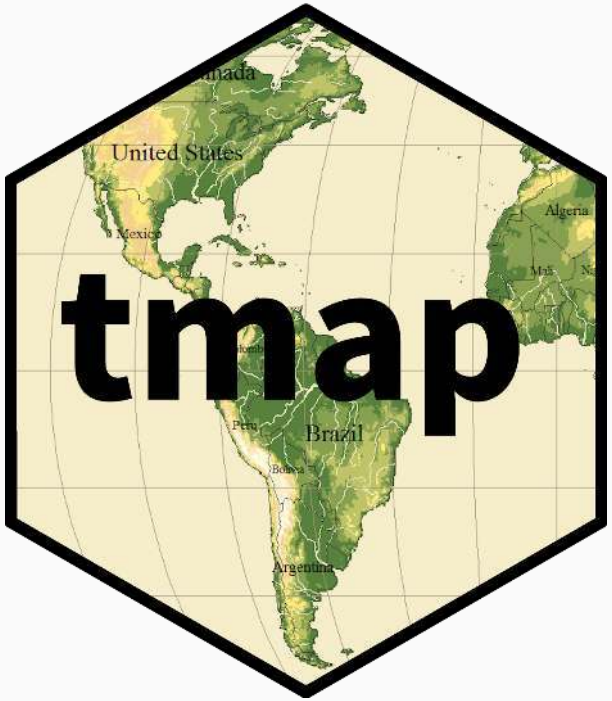
```
map_dem_rc_tmap_int ← map_dem_rc_tmap
map_dem_rc_tmap_int
```



5. Mapas interativos

tmap

```
# exportar mapa tmap interativo
tmap::tmap_save(tm = map_dem_rc_tmap_int,
                filename = here::here("dados", "mapas", "mapa_dem_rc_tmap_int.html"))
```



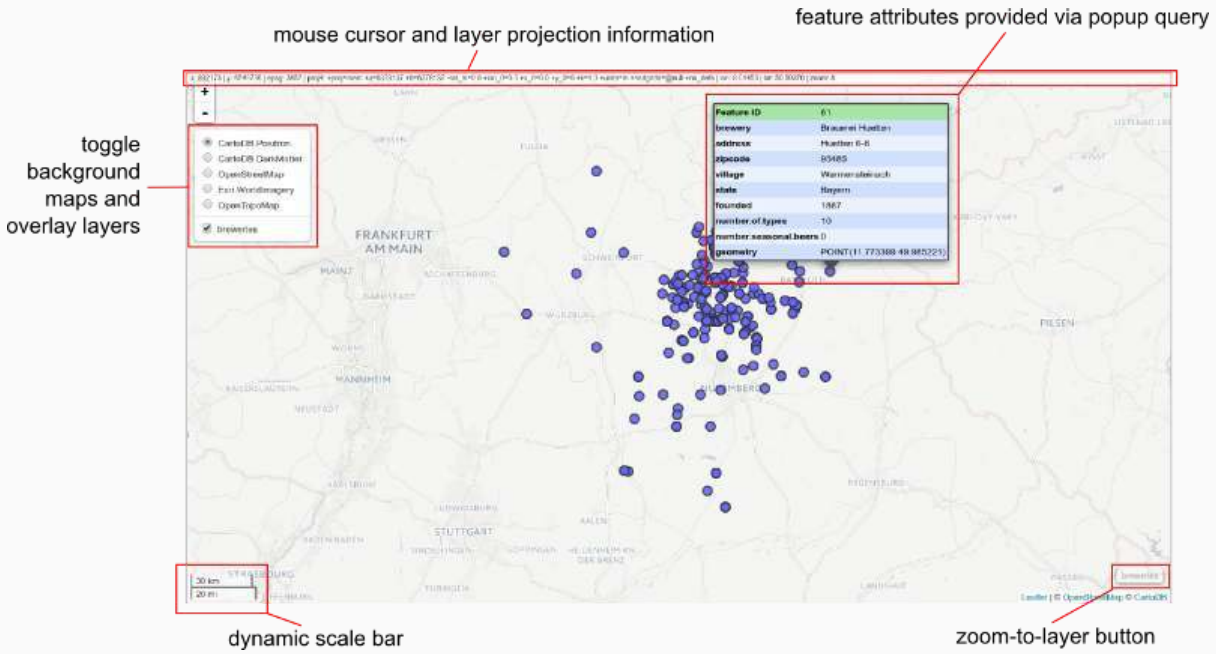
[mapview]

5. Mapas interativos

mapview

Display

```
# pacote
# install.packge("mapview")
library(mapview)
```



[mapview]

5. Mapas interativos

mapview

```
# plot
map_dem_rc_mapview_int ← mapview::mapview(dem_rc, col.regions = viridis::viridis(100))
map_dem_rc_mapview_int
```

5. Mapas interativos

mapview

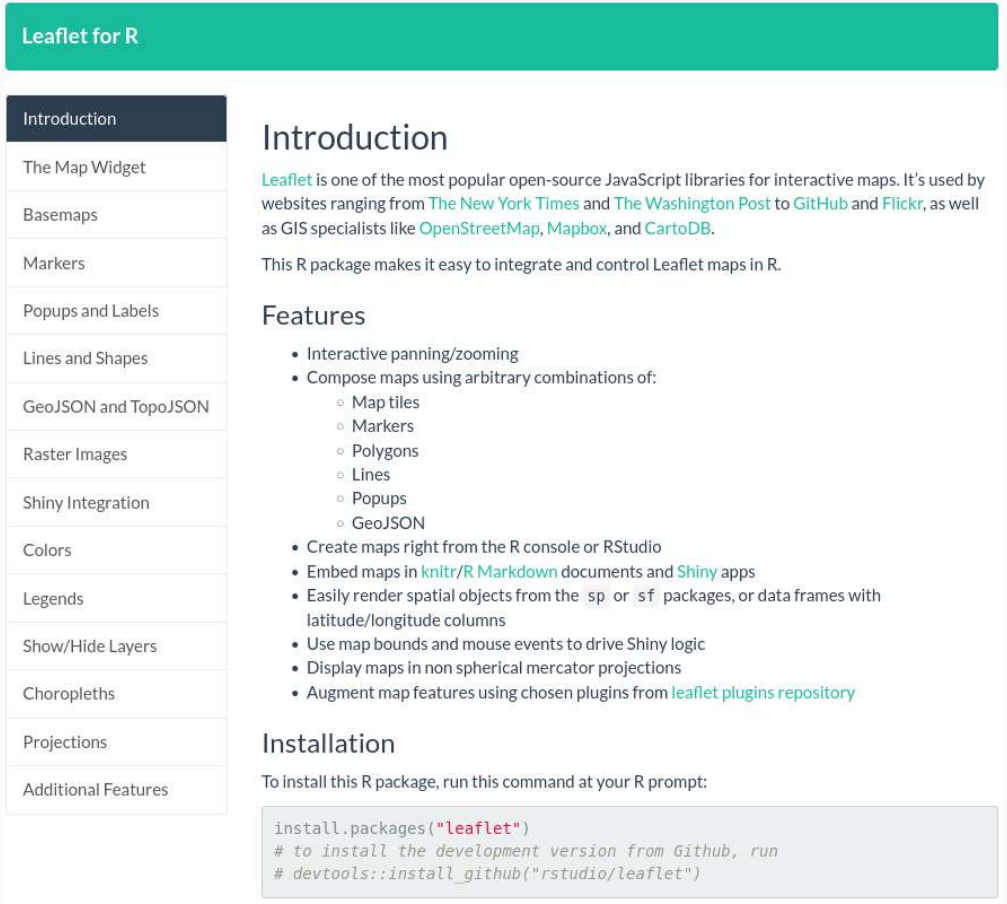
```
# exportar mapa tmap interativo
mapview::mapshot(x = map_dem_rc_mapview_int,
                 url = here::here("03_dados", "mapas", "mapa_dem_rc_mapview_int.html"))
```

leaflet

5. Mapas interativos

leaflet

```
# pacote
# install.packge("leaflet")
library(leaflet)
```



leafletjs, leaflet for R

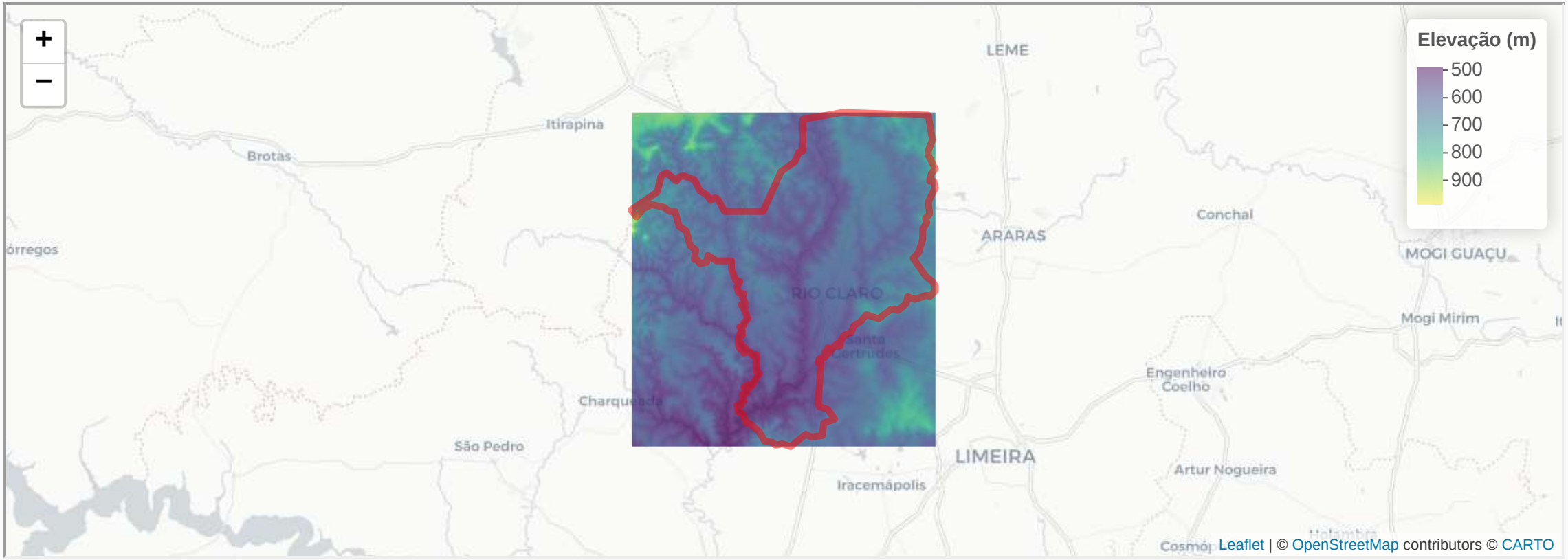
5. Mapas interativos

leaflet

```
# paleta de cores
pal ← colorNumeric(viridis::viridis(10), raster::values(dem_rc))

# mapa
map_dem_rc_leaflet_int ← leaflet() %>%
  addProviderTiles("CartoDB.Positron") %>%
  addRasterImage(dem_rc, colors = pal, opacity = .8) %>%
  addLegend(pal = pal, values = raster::values(dem_rc), title = "Elevação (m)") %>%
  addPolygons(data = rc_2020, col = "red", fill = NA)
map_dem_rc_leaflet_int
```

5. Mapas interativos

leaflet



5. Mapas interativos

leaflet

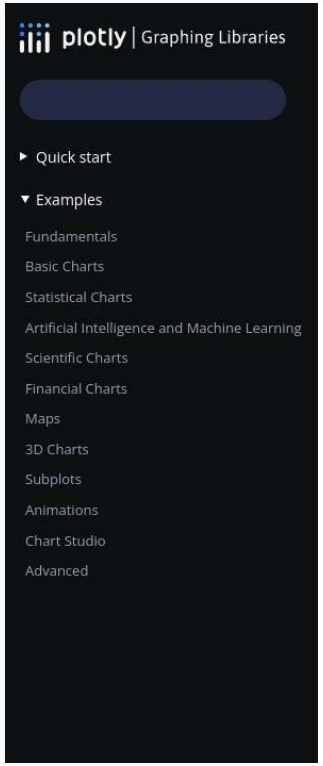
```
# exportar mapa tmap interativo
mapview::mapshot(x = map_dem_rc_leaflet_int,
                 url = here::here("03_dados", "mapas", "mapa_dem_rc_leaflet_int.html"))
```

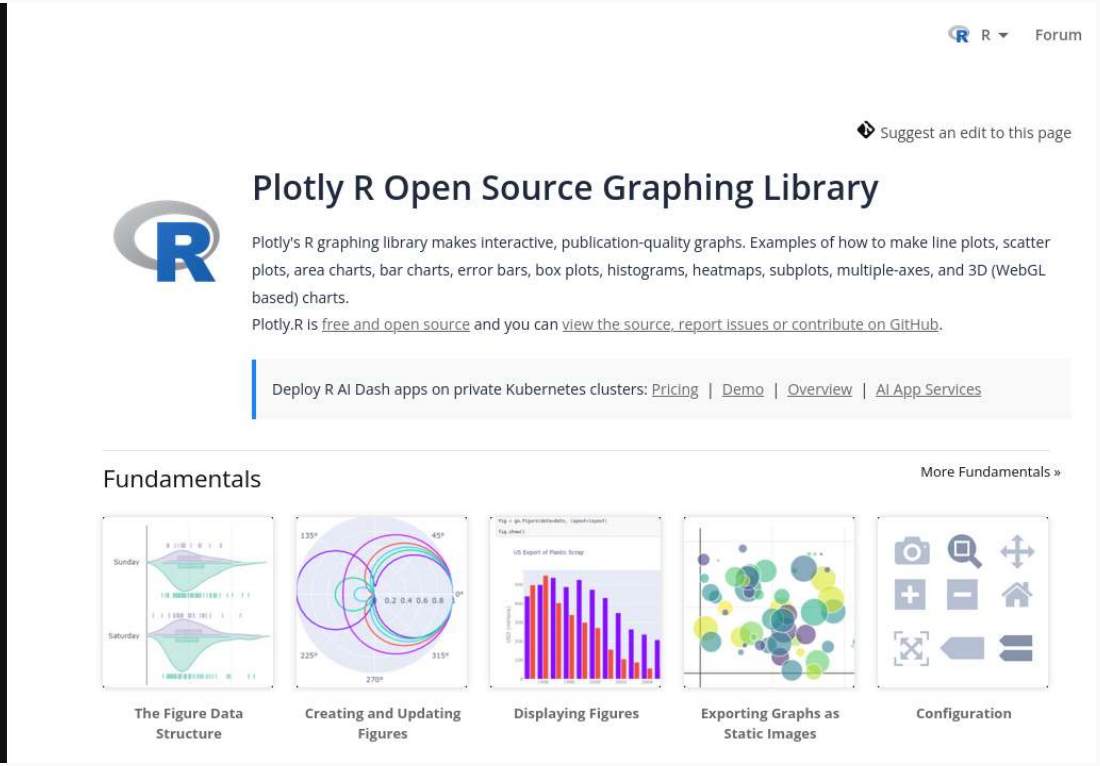
plotly

5. Mapas interativos

plotly

```
# pacote
# install.packge("plotly")
library(plotly)
```





plotly

5. Mapas interativos

plotly

```
# plot
map_rc_2020_plotly_int ← plot_geo(rc_2020)
map_rc_2020_plotly_int
```

5. Mapas interativos

plotly

```
# plot
map_rc_2020_plotly_int ← ggplotly(
  ggplot() +
    geom_sf(data = rc_2020) +
    theme_bw(base_size = 15))
map_rc_2020_plotly_int
```

5. Mapas interativos

plotly

```
# plot
map_biomas_plotly_int ← ggplotly(
  ggplot(data = biomas) +
    aes(fill = name_biome) +
    geom_sf(color = "black") +
    scale_fill_manual(values = c("darkgreen", "orange", "orange4",
                                 "forestgreen", "yellow", "yellow3")) +
    theme_bw(base_size = 15) +
    annotate(geom = "text", label = "CRS: SIRGAS2000/Geo", x = -38, y = -31, size = 2.5) +
    annotate(geom = "text", label = "Fonte: IBGE (2019)", x = -39, y = -32.5, size = 2.5) +
    labs(title = "Biomas do Brasil", fill = "Legenda", x = "Longitude", y = "Latitude"))
map_biomas_plotly_int
```

5. Mapas interativos

plotly

```
# exportar mapa tmap interativo
mapview::mapshot(x = map_biomas_plotly_int,
                 url = here::here("03_dados", "mapas", "mapa_dem_rc_leaflet_int.html"))
```

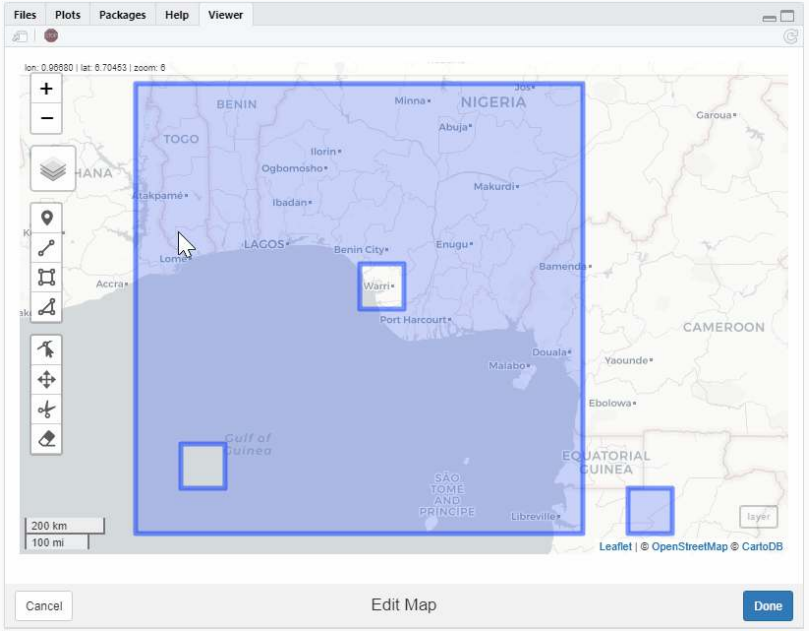
E para finalizar, vamos editar dados vetoriais no R

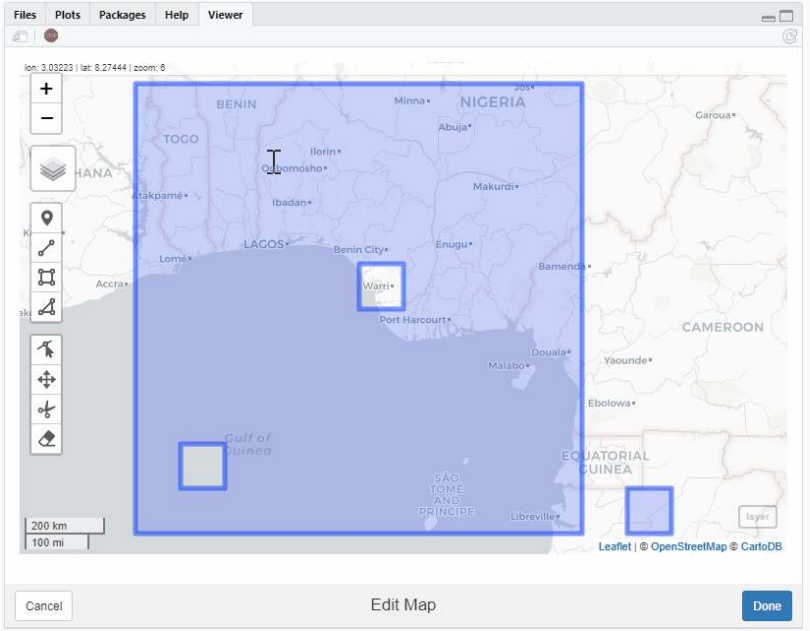
5. Mapas interativos

mapedit

Edições simples de pontos, linhas e polígonos

```
# pacote
# install.packages("mapedit")
library(mapedit)
```





mapedit

5. Mapas interativos

mapedit

```
# criar
unesp ← mapedit::drawFeatures()
unesp

# editar
unesp_editado ← mapedit::editFeatures(unesp)
unesp_editado

# mapa
mapview::mapview(unesp_editado)
```

Dúvidas?

Muito bem, finalizamos a parte teórico-prática...

Agora vem a parte do projeto, ou seja, onde vocês trabalham

Gostaria de agradecer todas e todos pela paciência e confiança 😅

Maurício Vancine

Contatos:

mauricio.vancine@gmail.com

@mauriciovancine

mauriciovancine

mauriciovancine.github.io

Slides criados via pacote xaringan e tema Metropolis. Animação dos sapos por @probzz.