

# Análisis gráfico con ggplot2. Parte final

## Otros gráficos con ggplot2

Hemos visto la enorme versatilidad de ggplot2 para generar gráficos de todo tipo. En esta parte final del Módulo 3 te presentamos algunos paquetes que permiten extender un poco más el uso de ggplot.

### - Mapas con ggplot

Necesito instalar los siguientes paquetes:

```
library("ggplot2")
library("cowplot")
library("sf")
library("rnaturalearth")
library("scales")
```

Limpio la zona de dibujo

```
theme_set(theme_bw())
```

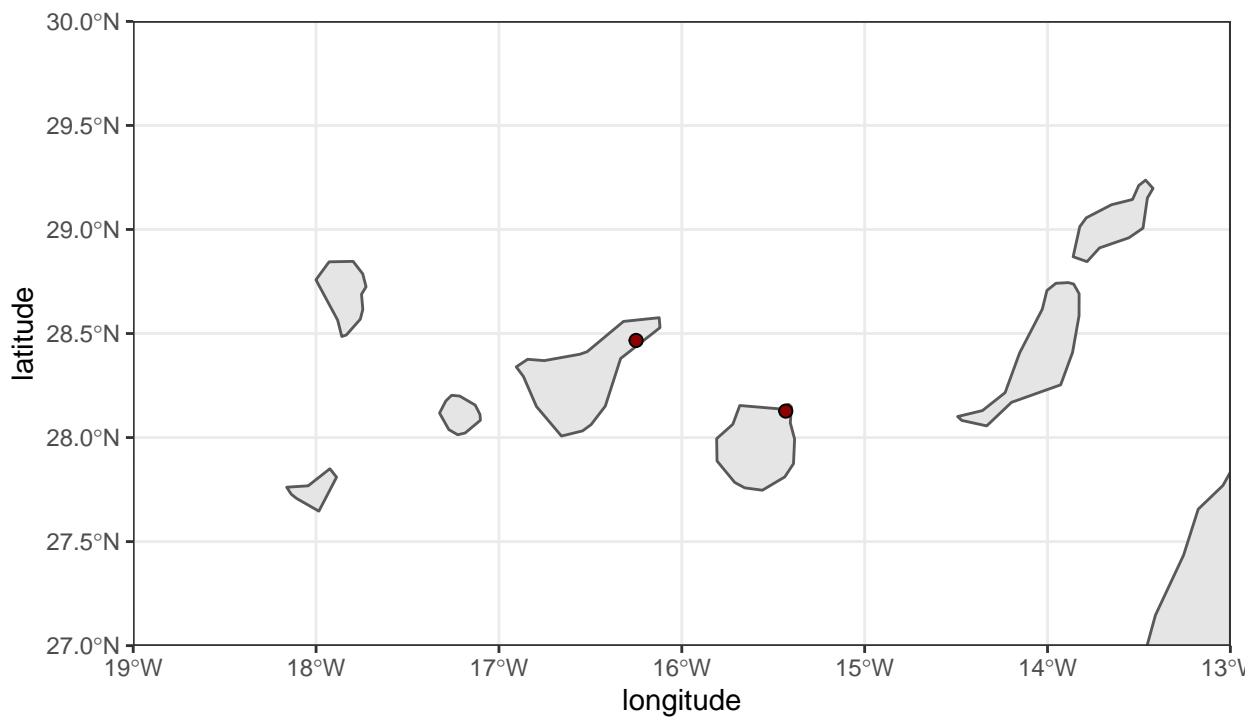
Comenzamos cargando el Mapamundi básico y con un zoom llegar a Canarias, marcando la localización de las dos capitales de provincia.

```
world <- ne_countries(scale = "medium", returnclass = "sf")
class(world)

## [1] "sf"           "data.frame"

sites <- data.frame(longitude = c(-15.431389, -16.25),
                     latitude = c(28.127222, 28.466667))

ggplot(data = world) +
  geom_sf() +
  geom_point(data = sites, aes(x = longitude, y = latitude), size = 2,
             shape = 21, fill = "darkred") +
  coord_sf(xlim = c(-19, -13), ylim = c(27, 30), expand = FALSE)
```



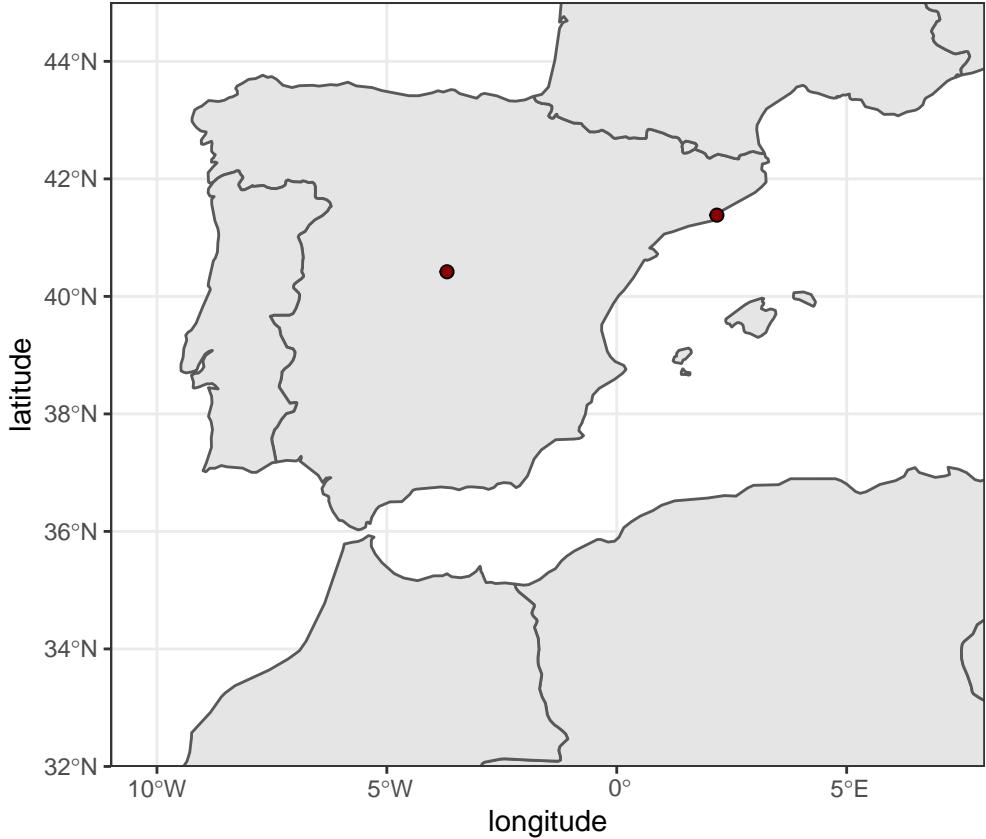
Hago lo mismo con la península y Madrid/Barcelona

```

sites <- data.frame(longitude = c(2.176944,-3.691944),
                     latitude = c(41.3825,40.418889))

ggplot(data = world) +
  geom_sf() +
  geom_point(data = sites, aes(x = longitude, y = latitude), size = 2,
             shape = 21, fill = "darkred") +
  coord_sf(xlim = c(-11, 8), ylim = c(32, 45), expand = FALSE)

```



A continuación somos más específicos y, a partir del repositorio de mapas del centro nacional de información geográfica, representamos las provincias españolas.

```

MF<-readRDS("gadm36_ESP_2_sf.rds") # Fichero de provincias
regiones = st_as_sf(MF, coords = c("x", "y"), crs = 28992, agr = "constant")

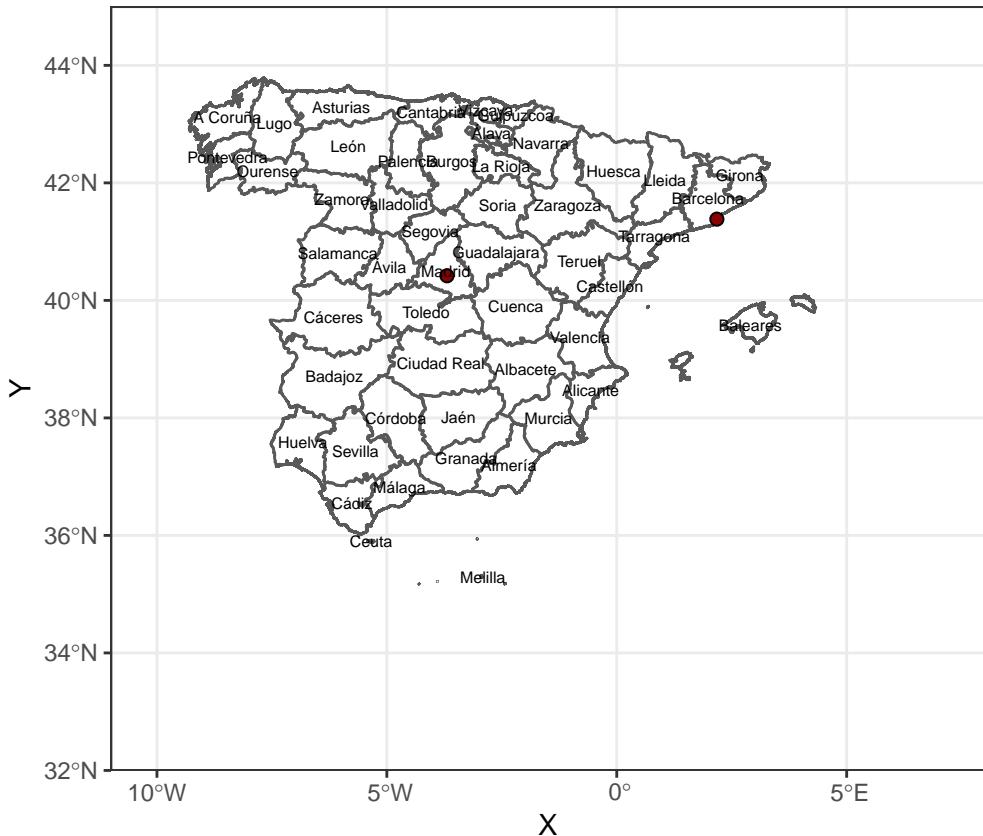
Prov <-regiones[,14]

Prov <- cbind(Prov, st_coordinates(st_centroid(Prov)))
Prov$ID <- c(regiones$NAME_2) # Nombre de provincias

sites <- st_as_sf(sites, coords = c("longitude", "latitude"),
                  crs = 4326, agr = "constant")

ggplot(data = world) +
  geom_sf(data = Prov, fill = NA) +
  geom_sf(data = sites, size = 2, shape = 21, fill = "darkred") +
  geom_text(data = Prov, aes(X, Y, label = ID), size = 2) +
  coord_sf(xlim = c(-11, 8), ylim = c(32, 45), expand = FALSE)

```



Con este mapa como base puedo colorear cualquier información en relación a las provincias españolas. A modo de ejemplo, dibujo por colores el porcentaje de personas que en junio de 2020 habían contactado con los servicios sanitarios por síntomas de covid.

```

Provin <- read.csv("Provin.csv", sep=";")
Prov$sintomas <- Provin$sintomas

Peni <- ggplot(data = world) +
  ggtitle("Contacto con los servicios sanitarios por síntomas de COVID-19") +
  geom_sf(fill = "white") +
  geom_sf(data = Prov, fill = NA) +
  geom_sf(data = Prov, aes(fill = sintomas)) +
  scale_fill_viridis_c(trans = "sqrt", alpha = .4,
                        labels = percent) +
  geom_text(data = Prov, aes(X, Y, label = scales::percent(round(sintomas,digits = 2))),
            size = 3) +
  coord_sf(xlim = c(-11, 8), ylim = c(32, 45), expand = FALSE) +
  theme(legend.title=element_blank(), axis.text.x=element_blank(),
        axis.text.y=element_blank(), axis.title.x=element_blank(),
        axis.title.y=element_blank(), legend.position = "none")

# Islas Canarias

Prov$nudge_y <- .5
Prov$nudge_y[Prov$ID == "Santa Cruz De Tenerife"] <- 0.5

```

```

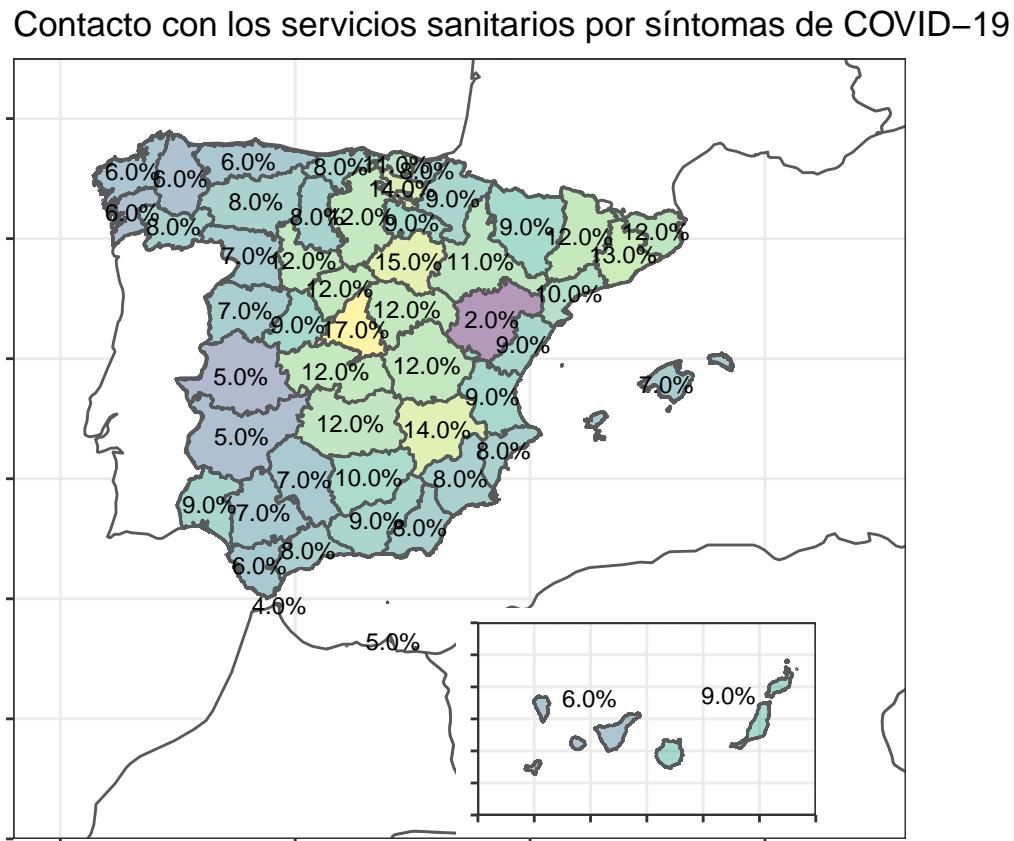
Cana <-ggplot(data = world) +
  geom_sf(data = Prov, fill = NA) +
  geom_sf(data = Prov, aes(fill = sintomas)) +
  scale_fill_viridis_c(trans = "sqrt", alpha = .4,
                        labels = percent, breaks = c(.04, .16)) +
  geom_text(data = Prov, aes(X, Y,label = scales::percent(round(sintomas,digits =2))),
            size = 3, nudge_y = Prov$nudge_y)+ 
  coord_sf(xlim = c(-19, -13), ylim = c(27, 30), expand = FALSE) +
  theme(legend.position = "none", axis.text.x=element_blank(),
        axis.text.y=element_blank(),axis.title.x=element_blank(),
        axis.title.y=element_blank())

# Combino ambos gráficos

plot.with.inset <-
  ggdraw() +
  draw_plot(Peni) +
  draw_plot(Cana, x = 0.5, y = .01, width = .3, height = .3)

plot.with.inset

```



y termino generando un png con el gráfico.

```
ggsave(filename = "plot.with.inset.png",  
       plot = plot.with.inset,  
       width = 17,
```

```
height = 12,  
units = "cm",  
dpi = 300)
```

## Mapas de ciudades

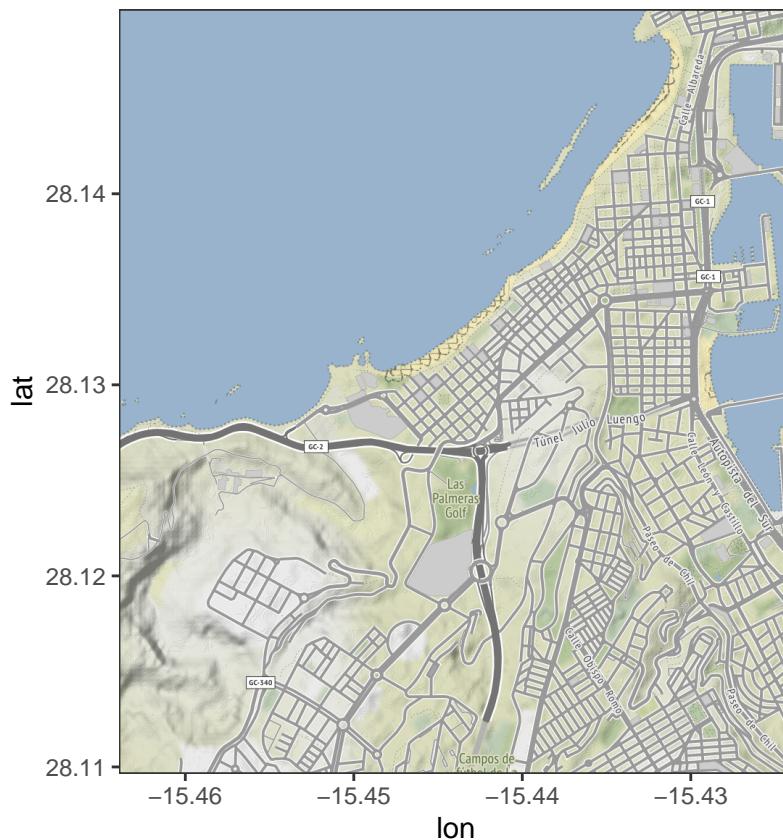
Los mapas no tienen que referir sólo regiones o provincias. Puedo generar mapas también por ciudades o barrios. Veamos cómo. Necesitaré las siguientes librerías:

```
library(tidyverse)  
library(sf)  
library(ggmap)  
library(osmdata)  
library(leaflet)
```

Los mapas los tomamos del siguiente repositorio:

<https://www.openstreetmap.org>

```
bbox_lpgc <- getbb("Guanarteme, Las Palmas de Gran Canaria")  
  
mapa_lpgc <- get_stamenmap(bbox = bbox_lpgc,  
                           zoom=15)  
ggmap(mapa_lpgc)
```



Finalmente un poco de “animación” con ganimate. Lo haremos emulando al gran Hans Rosling ya fallecido. Necesitamos los siguientes paquetes:

```
library(gapminder) # obtengo los datos  
library(ggplot2)  
library(gganimate)
```

Simulo la animación superponiendo las imágenes año tras año

```
HR <- ggplot(gapminder, aes(gdpPercap, lifeExp, size = pop, color = continent)) +  
  geom_point() +  
  scale_x_log10() +  
  theme_bw() +  
  # gganimate specific bits:  
  labs(title = 'Year: {frame_time}', x = 'GDP per capita', y = 'life expectancy') +  
  transition_time(year) +  
  ease_aes('linear')
```

ejecuta print(HR) para ver la animación.

Finalmente genero el gif animado con  
anim\_save("HR.gif")