

Créer une carte avec ggplot2

Introduction :

Ggplot2 est un package de visualisation utilisé aussi bien en R qu'en python. On l'utilise pour faire des graphiques à partir d'une base de données, mais on peut s'en servir pour bien plus que cela.

Ici, nous allons utiliser ggplot2 afin de visualiser des cartes, et de les rendre plus attractives. Pour cela en plus de ggplot2 nous allons utiliser le package map et data map entre autres.

L'objectif est de comprendre comment fonctionne ggplot2 avec les cartes afin de pouvoir plus tard créer des cartes intéressantes et de visualiser sur google map les informations que l'on souhaite faire ressortir sur une carte.

Avec ggplot2, nous pouvons par exemple faire une carte météo, une carte sur les épidémies ou encore des cartes sur des critères sociaux démographiques. L'utilité de ggplot2 est de faire ressortir l'information que l'on veut faire passer de manière visuelle. Dans un case study, David Kahle a fait ressortir sur carte map la weathermap des crimes à Houston aux Etats-Unis.

Cas pratique

Dans ce cas pratique, nous allons faire appel à plusieurs packages: ggplot2, maps, mapdata, dplyr et viridis

dans un premier temps, nous allons installer et lancer les packages :

```
install.packages("maps")
library('maps')
install.packages("mapdata")
library('mapdata')
install.packages('dplyr')
library('dplyr')
install.packages("viridis")
library('viridis')
library('ggplot2')
```

Une fois les packages lancés, nous allons pouvoir commencer.

Une carte sans et une carte avec ggplot2 :

La carte sans :

```
library(maps)
library(mapdata)
```

```
map('worldHires')
```

La carte avec :

```
install.packages("maps")
library(maps)
install.packages("mapdata")
library('mapdata')
install.packages('dplyr')
library('dplyr')
install.packages("viridis")
library('viridis')
library('ggplot2')
```

```
world_map <- map_data("world")
ggplot(world_map, aes(x = long, y = lat, group = group)) +
  geom_polygon(fill="lightgray", colour = "white")
```

On remarque déjà que la carte du monde avec ggplot2 est bien plus esthétique

Nous pouvons aussi sélectionner juste certain pays qui nous intéressent :

```
# Quelques pays de l'UE
some.eu.countries <- c(
  "Portugal", "Spain", "France", "Switzerland", "Germany",
  "Austria", "Belgium", "UK", "Netherlands",
  "Denmark", "Poland", "Italy",
  "Croatia", "Slovenia", "Hungary", "Slovakia",
  "Czech republic"
)
# Récupérer les données cartographiques
some.eu.maps <- map_data("world", region = some.eu.countries)

# Calculer le centroïde comme étant la longitude et la latitude moyennes
# Utilisé comme coordonnée pour les noms de pays
region.lab.data <- some.eu.maps %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat))

ggplot(some.eu.maps, aes(x = long, y = lat)) +
  geom_polygon(aes( group = group, fill = region))+
  geom_text(aes(label = region), data = region.lab.data, size = 3, hjust = 0.5)+
  scale_fill_viridis_d()+
  theme_void()+
  theme(legend.position = "none")
```