Réaliser des cartes thématiques avec Mapsf

Coralie Cottet

2023-07-27

Nous allons montrer comment réaliser des cartes avec R à l’aide du package **mapsf** (T.Giraud) et du packages **sf**. Le package “mapsf” est un outil très utile pour manipuler et visualiser des données spatiales en R, tandis que le package “sf” fournit une infrastructure pour stocker et manipuler ce type de données. Ces deux outils sont une solution simple et pratique pour réaliser ce type d’analyse.



**Documentation** (Timothé Giraud):

* <https://rcarto.github.io/ined2022/07_mise_en_page.html>
* <http://riatelab.github.io/mapsf>
* <http://rgeomatic.hypotheses.org/2077>

**Installation des packages**:

install.packages(mapsf)  
install.packages(sf)  
install.packages(dplyr)

library(mapsf)  
library(sf)  
library(dplyr)

Nous allons réaliser une carte de Paris avec les données du site de Paris Data [[Lien](http://opendata.paris.fr/pages/catalogue/?disjunctive.theme&disjunctive.publisher)]

# Importation des données

Il existe plusieurs formats de fond de carte:

* **GeoJSON** est un format de données géospatiales basé sur JSON (JavaScript Object Notation) qui permet de stocker des données géographiques sous forme de caractères.
* **Shapefile** est un format de données géospatiales propriétaire développé par *Esri* (Environmental Systems Research Institute). Il est constitué de plusieurs fichiers qui stockent des informations sur les entités géographiques telles que les points, les lignes et les polygones, ainsi que des attributs associés à ces entités.

Ici le fond importé est en format *GeoJSON*, il correspond aux limites des arrondissements. On va lui ajouter celui des voies d’eau en couche d’habillage.

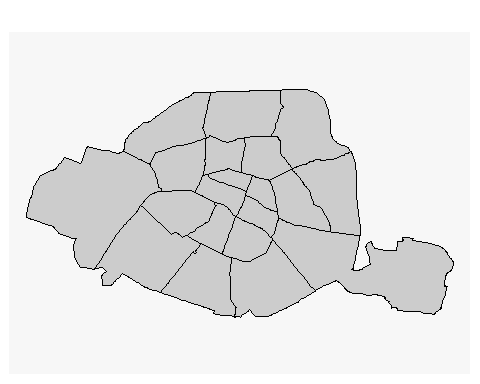
arrondissements <- st\_read(dsn = "https://opendata.paris.fr/explore/dataset/arrondissements/download/?format=geojson&timezone=Europe/Berlin&lang=fr")

Reading layer `OGRGeoJSON' from data source   
 `https://opendata.paris.fr/explore/dataset/arrondissements/download/?format=geojson&timezone=Europe/Berlin&lang=fr'   
 using driver `GeoJSON'  
Simple feature collection with 20 features and 9 fields  
Geometry type: POLYGON  
Dimension: XY  
Bounding box: xmin: 2.224078 ymin: 48.81558 xmax: 2.469761 ymax: 48.90216  
Geodetic CRS: WGS 84

cours\_deau<-st\_read(dsn="https://opendata.paris.fr/api/explore/v2.1/catalog/datasets/plan-de-voirie-voies-deau/exports/geojson?lang=fr&timezone=Europe%2FBerlin")

Reading layer `OGRGeoJSON' from data source   
 `https://opendata.paris.fr/api/explore/v2.1/catalog/datasets/plan-de-voirie-voies-deau/exports/geojson?lang=fr&timezone=Europe%2FBerlin'   
 using driver `GeoJSON'  
Simple feature collection with 58 features and 25 fields  
Geometry type: MULTIPOLYGON  
Dimension: XY  
Bounding box: xmin: 2.224081 ymin: 48.81924 xmax: 2.450555 ymax: 48.90207  
Geodetic CRS: WGS 84

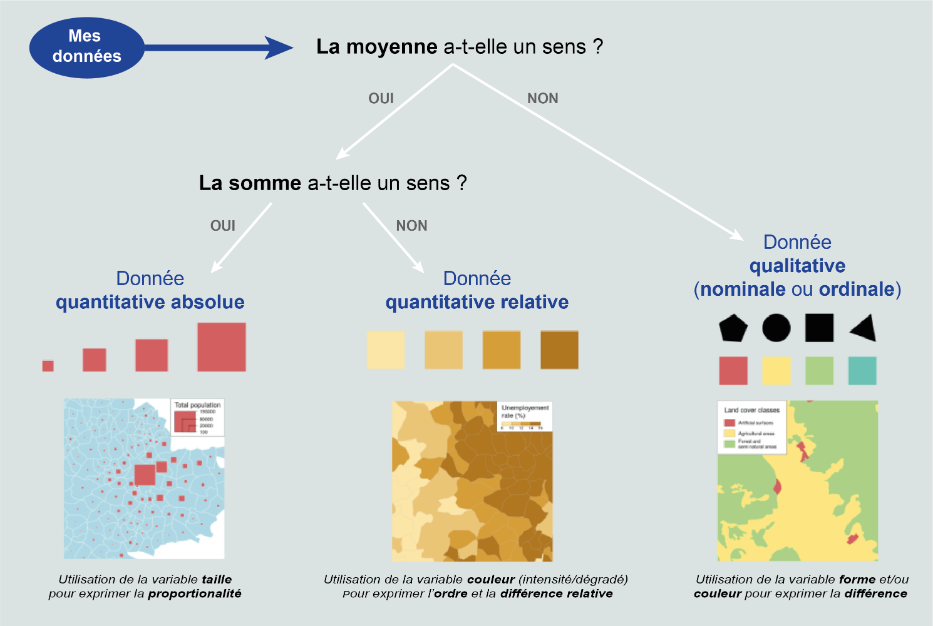
mf\_map(x = arrondissements, border = "black")



{r filename=“Affichage des cours d’eau”} mf\_map(x=cours\_deau,lwd=2,border=“lightblue”,col=“lightblue”,add=TRUE)

mf\_theme("darkula")

# Différents types de cartes



## Les cartes à symboles proportionnels

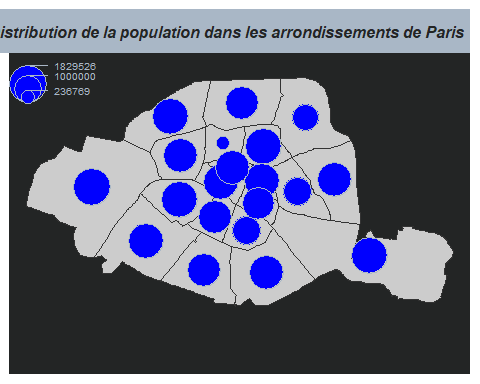
La carte en symboles proportionnels est la méthode de représentation graphique à utiliser pour visualiser des variables de stocks (variables quantitatives absolues pour lesquelles la somme et la moyenne ont une signification).

Sur la carte, on affiche l’effectif de la population de chaque arrondissement et on ajoute la légende et un titre.

#Ajout manuel des valeurs des populations de chaque arrondissements dans le tableau sous le nom "pop"  
arrondissements$pop <- c(1012687, 903036, 1369857, 1491027, 1672009, 1506475, 1637163, 1603380, 1637542, 1059282, 1494945, 1436205, 1705774, 1425805, 236769, 1668605, 1674568, 1539668, 1421827, 1829526) #INSEE,2021. La valeur 1012687 correspond à la population dans le 1er arrondissement.

* Pour représenter la carte, on utilise la fonction map\_sf(). On doit indiquer à minima la variable liée à la statistique (var=pop), et le type de carte (type=prop).
* On va superposer 2 cartes: le fond affichant seulement les limites des arrondissements, et celle affichant sous forme de bulle proportionnelles la taille de la population dans chaque arrondissement.

mf\_map(x = arrondissements)  
mf\_map(  
 x = arrondissements,  
 var = "pop" ,  
 type = "prop",   
 leg\_title = "Population totale\12 271 794",  
 col="blue",  
 add= TRUE,  
 inches=0.2)  
  
mf\_title("Distribution de la population dans les arrondissements de Paris")



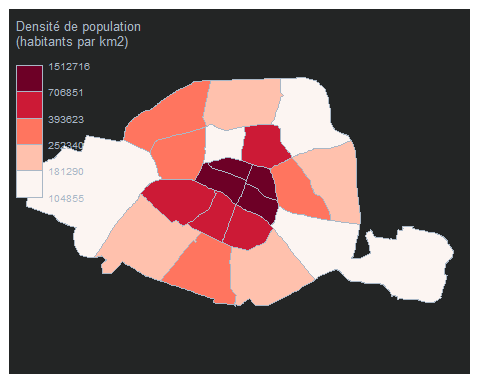
## Les cartes choroplèthes

La *carte choroplèthe* est la représentation à utiliser pour visualiser des ratios (variables quantitatives relatives pour lesquelles la moyenne a un sens, mais dont la somme n’en a pas), et qui sont des variables ordinales.

Sur la carte, on affiche la densité de population de chaque arrondissement. On utilise ici la méthode des quantiles pour discrétiser cette variable, et on applique une palette de couleurs séquentielles avec dégradé pour représenter l’ordre entre les valeurs.

#création de la variable densite   
arrondissements$DENS <- 1e6 \* arrondissements$pop / as.numeric(st\_area(arrondissements))

mf\_map(  
 x = arrondissements,  
# variable à représenter: densité de population   
var = "DENS",  
# type de carte   
type = "choro",  
# méthode de dsicrétisation  
 breaks = "quantile",  
# palette de rouges   
pal = "Reds",  
 lwd = 1,  
 leg\_title = "Densité de population\n(habitants par km2)",   
 leg\_val\_rnd = 0  
)



|  |
| --- |
| Note |
| la faible densité de population des 12e et 16e arrondissement résulte notamment des zônes *bois de Vincennes* et de *Bois de Boulogne* qui les composent. |

On peut aussi représenter un autre type de variable: qualitative nominale. Pour illuster cette fiche on va représenter les bords politiques des mairies de chaque arrondissement. Pour cela, on produit une carte choroplèthes mais avec des couleurs *sans dégradé* (palettee qualitative).

arrondissements2 = select(arrondissements, c\_ar, l\_aroff, geom\_x\_y, geometry)  
arrondissements2 = arrange(arrondissements2, c\_ar)  
  
bords\_politiques <- c("LR", "EELV", "PS", "PS", "LR", "LR", "LR", "LR", "LR", "PS", "PS", "PS", "PS", "PS", "LR", "LR", "LR", "PS", "PS", "PS")  
# Le maire du 2e arrondissement de Paris était EELV entre 2014 et 2015.  
  
  
arrondissements2 <- cbind(arrondissements2, bords\_politiques)  
  
# Choix des couleurs  
  
colors <- c("EELV" = "#00CC66", "LR" = "#0066CC", "PS" = "#FF0066")  
  
# Pour placer les carrés sur les centroïdes des polygones représentant les arrondissements.  
  
arr\_c<-st\_centroid(arrondissements2)  
mf\_map(x = arrondissements2, border = "black")  
  
# Ajouter des rectangle (pch=15) remplis pour chaque arrondissement en fonction de son bord politique  
mf\_map(  
 x = arrondissements2,  
 # var à représenter   
 var = "bords\_politiques",  
 pal=colors,  
 type="symb",  
 pch=15,  
 cex = 2,  
 lwd = .5,  
 leg\_title = "",  
 border="black"  
)

the length of pch does not match the number of modalities. The first pch is used for all modalities

mf\_layout(  
 title = "Bord politique des maires de Paris entre 2014 et 2020",  
)

