Analyse spatiale et territoriale du logement social

Formation Carthageo-Geoprisme 2022

Claude Grasland, Université de Paris Cité

Section 1

Localisation spatiale

Retour sur sf

Nous revenons sur le **package sf (spatial features)** que nous avons déjà rencontré au moment de la création de cartes thématiques par IRIS ou communes à l'aide du package `mapsf`.

Ici le package sf va être utilisé pour cartographier non pas des zones mais des localisations ponctuelles. Il pourra être à nouveau couplé avec le logiciel de **cartogaphie statique** comme mapsf , afin par exemple de placer les localisations des logements sociaux au dessus du fonds de carte des IRIS ou communes.

Mais il pourra aussi servir de base à des **cartographies dynamiques** permettant de placer les points sur des réseaux de rue et plus généralement sur des "tuiles" cartographiques permettant d'effectur des zoom. On utilisera à cet effet d'autres packages comme `leaflet` ou sa version simplifiée `mapview`.

Données ponctuelles

Nous reprenons le fichier de localisation établi au chapitre précédent et nous ne conservons que 6 variables:

```
logt <- readRDS("data/sel_logt.RDS") %>%
        select(adresse=result_id,
               X,Y,
               date = CONSTRUCT)
```

adresse	X	Y	date
94019_0037_00005	666779.5	6855840	1971
94019_0037_00001	666716.7	6855829	1971
94019_0037_00001	666716.7	6855829	1971

Données IRIS

Nous chargeons par ailleurs le fichier des IRIS en ne gardant que la zone d'étude :

```
map_iris <- readRDS("data/map_iris.RDS") %>%
          filter(INSEE_COM %in% c("94011","94019",
                                  "94071","94055"))
```

INSEE_COM	NOM_COM	IRIS	CODE_IRIS	NOM_IRIS	TYP_IRIS	DEPT
94011	Bonneuil-sur-Marne	0105	940110105	Saint-Exupéry	H	94
94055	Ormesson-sur-Marne	0103	940550103	Centre Sud	H	94
94071	Sucy-en-Brie	0103	940710103	La Cité Verte	H	94

Agrégation par commune

Rappel : on peut agréger les géométries d'un fonds sf. Ici on va créer le fonds de carte des communes.

```
map_com <- map_iris %>% group_by(INSEE_COM,NOM_COM) %>%
                  summarise() %>%
                  st_cast("MULTIPOLYGON")
```

```
## `summarise()` has grouped output by 'INSEE_COM'. You can ov
## `.groups` argument.
```

Vérification de la projection

Nous savons que les coordonnées X,Y du fichier logement sont projetées en EPS 2154. Mais quelle est la projection de notre fonds IRIS ? S'agit-il de la même ?

```
st_crs(map_iris)$proj4string
```

```
## [1] "+proj=lcc +lat_0=46.5 +lon_0=3 +lat_1=49 +lat_2=44 +x_
```

```
st_crs(2154)$proj4string
```

```
## [1] "+proj=lcc +lat_0=46.5 +lon_0=3 +lat_1=49 +lat_2=44 +x_
```

A priori il s'agit bien de la même de sorte que les coordonnées X,Y devraient bien se superposer sur le fonds IRIS

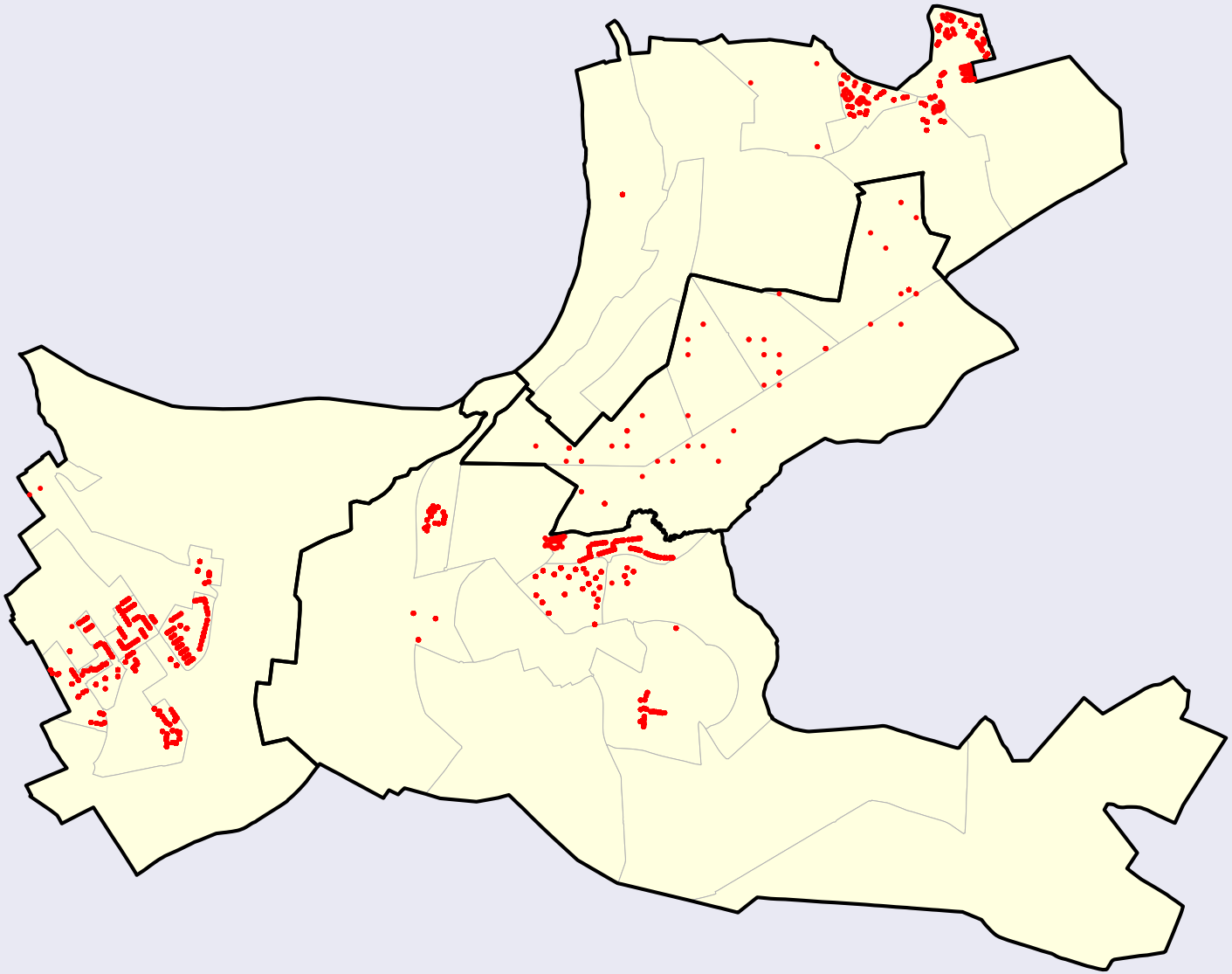
Test de superposition

Programme

```
par(mar=c(0,0,0,0))
#trace les iris
plot(map_iris$geometry,
     col="lightyellow", border="gray70",
     lwd=0.2)
# trace les communes
plot(map_com$geometry,
     col=NA, lwd=1, add=T)
# ajoute les points
points(x=logt$X,
       y=logt$Y,
       cex=0.2,
       col="red",
       pch = 16)
```

Test de superposition

Résultat



fichier des adresses

Nous allons maintenant établir un fichier de localisation des adresses en nous servant de l'identifiant unique fourni par l'INSEE.

```
adr <- logt %>% select(adresse,X,Y) %>%
                filter(duplicated(adresse) == F) %>%
                filter(is.na(X) ==F,is.na(Y)==F)
```

On constate qu'il n'y a que 652 adresses différentes alors que notre fichier fait état de 8139 logements. Une adresse regroupe donc en moyenne plus de 10 logements (habitat collectif).

Transformation en fichier sf

La transformation de notre fichier initial au format sf est facile à réaliser avec la fonction `st_as_sf()` du package sf. Mais il faut prendre garde de bien préciser le système de projection si l'on veut pouvoir ensuite l'utiliser.

```
map_adr <- st_as_sf(adr, coords = c("X","Y"))
st_crs(map_adr)<- 2154
str(map_adr)
```

```
## Classes 'sf', 'data.table' and 'data.frame': 612 obs. of  2
##  $ adresse : chr  "94019_0037_00005" "94019_0037_00001" "94
##  $ geometry:sfc_POINT of length 612; first list element:
##  - attr(*, "sf_column")= chr "geometry"
##  - attr(*, "agr")= Factor w/ 3 levels "constant","aggregate
##   ..- attr(*, "names")= chr "adresse"
```

Agrégation des logements

Notre nouveau fichier sf permet désormais d'effectuer des jointures avec le fichier des logements sociaux. A titre d'exemple on peut désormais compter le nombre de logements par adresse et leur ancienneté moyenne.

programme

```
logt_by_adr <- logt %>%
               group_by(adresse) %>%
               summarise(nblog = n(),
                         datemoy = mean(date))
```

Agrégation des logements

résultat

adresse	nblog	datemoy
	31	2014
94011_0017_00001	10	1970
94011_0017_00003	10	1970
94011_0017_00005	10	1970
94011_0019_00002	25	1992
94011_0019_00004	24	1992
94011_0022_00001	20	1966
94011_0022_00002	20	1966
94011_0022_00003	20	1966
94011_0022_00004	20	1966

Jointure

On peut désormais effectuer la jointure entre les données agrégées par adresse et le fichier sf de localisation des adresses :

```
map_logt <- inner_join(logt_by_adr,map_adr) %>% st_as_sf()
```

```
## Joining, by = "adresse"
```

Cartographie avec mapsf

On peut désormais utiliser les méthodes de cartographie déjà vues avec mapsf :

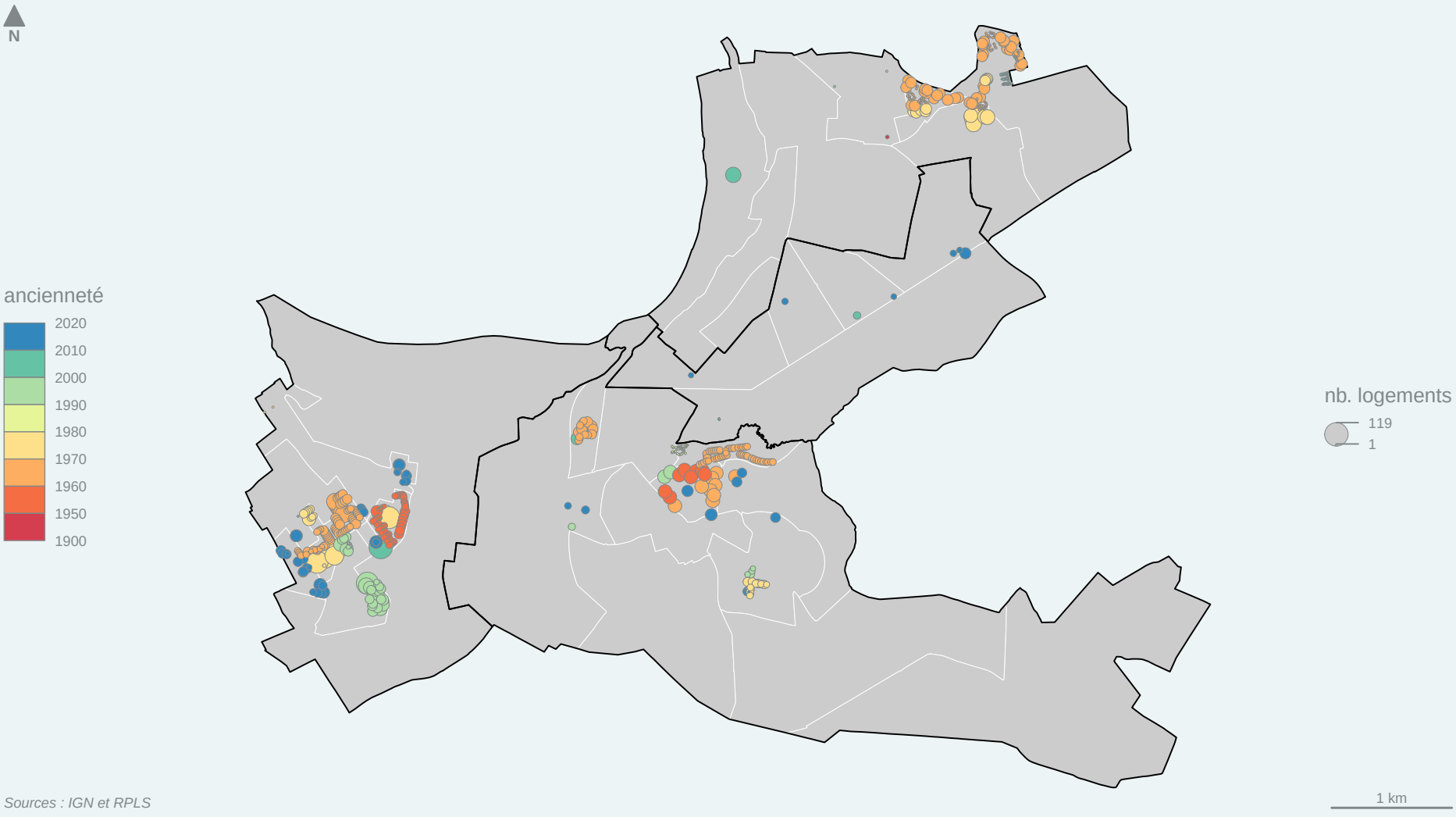
programme

```
mf_theme("agolalight")
mybreaks = c(1900, 1950, 1960, 1970, 1980, 1990, 2000, 2010, 2
mypal=brewer.pal(n = 8,name = "Spectral")
mf_map(map_iris, type = "base",
       col = "gray80",border="white", lwd=0.3)
mf_map(map_com, type = "base",
       col = NA,border="black",lwd=1,add = TRUE)
mf_prop_choro( x = map_logt,  var = c("nblog", "datemoy"),
  inches = 0.08, col_na = "grey", pal=mypal,
  breaks = mybreaks, nbreaks = 4, lwd = 0.1,
  leg_pos = c("right", "left"),leg_val_rnd = c(0,0),
  leg_title = c("nb. logements", "ancienneté"),
  add = TRUE)
```

Cartographie avec mapsf

résultat

Les logements sociaux en 2020



Sauvegarde des fichiers carographiques

On sauvegarde nos différents fichiers cartographiques au format sf relatifs à la zone d'étude.

```
saveRDS(map_com,"data/sel_map_com.RDS")
saveRDS(map_iris,"data/sel_map_iris.RDS")
saveRDS(map_logt,"data/sel_map_logt.RDS")
```