

Zone de chalandise

jcb

12 juillet 2015

Contents

Codes postaux du 67	1
Récupération de RPU créés	2
Liste des CP et des RPU	2
Jointure cp67 et FINESS	2
Calcul des taux de recours	2
Cartographie selon le taux de recours	3
Fusion des CP	6
Récupérer les coordonnées des polygones	7
Coordonnées du rectangle englobant	7
Cartes de base à mettre en stock	7

L'objet est de représenter graphiquement les zones de provenance des patients consultants aux urgences. Le premier découpage utilisé sont les zones définies par les codes postaux, indication figurant dans le RPU. On dispose d'un fichier Shapefile représentant les zones de CP pour toute la France (source Open Data). A partir de ce fichier on a été créés:

- **cp67.Rda** shapefile pour le bas-Rhin
- **cp68.Rda** shapefile pour le haut-Rhin
- **cp.alsace.Rda** shapefile pour l'Alsace

Voir **carto&pop.Rmd**, paragraphe *Codes Postaux* pour les détails de la création des fichiers shapefile.

Codes postaux du 67

La base de données attachée au fichier donne les indications suivantes:

```
## [1] "ID"      "LIB"      "DEP"      "SURF"      "POP2010" "X_COL6"
```

- ID: code postal
- LIB: libellé su CP (nom de la commune)
- DEP: n° du département
- SURF: surface de la zone

- POP2010: population 2010
- X_COL6: nombre de ménages

On peut calculer un **Taux de Recours**: nombre de passages dans les structures d'urgence de la zone de patients originaires d'une zone donnée divisé par la population estimée de la même zone sur une période donnée.

Il faut donc connaître le nombre de RPU enregistré durant la période pour une zone donnée.

Ici la zone est l'aire couverte par le code postal (CP).

Récupération de RPU créés

Le nombre de RPU pour un CP peut être extrait de la base RPU.

Liste des CP et des RPU

On forme `cp.hus`, un dataframe à 2 colonnes (CP, RPU) contenant le nombre de RPU pour un code postal donné résultant d'une consultation aux HUS. On a `nrow(cp.hus) = 100` => les RPU des HUS en 2014 proviennent de 100 zones postales différentes.

L'objectif est de pouvoir réaliser une jointure (merge) entre cette matrice et `data.cp` pour disposer d'une table attributaire permettant d'effectuer des calculs entre des données RPU et des données géographiques (comme calculer la taux de recours).

La fonction `chalandise` permet de créer ce dataframe.

	CP	RPU
1	67160	4682
2	67250	2186
3	67470	1640
4	67630	928
5	67510	590
6	67690	417

Jointure `cp67` et `FINESS`

On réalise un merging entre `data.cp` et `cp.hus` grace à la méthode `attrbJoin` qui utilise `match` à la place de `merge` (la fonction `merge` ne conserve pas toujours l'ordre des lignes.)

Note: la jointure ne concerne que la table attributaire (`cp67@data`) qui se voit ajouter les colonnes RPU et CP (la colonne CP est redondante avec la colonne ID).

Calcul des taux de recours

La table attributaire possède 2 colonnes, POP2010 pour la population de la zone du code postal et RPU contenant le nombre de personnes de la zone ayant eu recours au SU du secteur. Le rapport des deux fournit le taux de recours qui est stocké dans une colonne supplémentaire de la table attributaire.

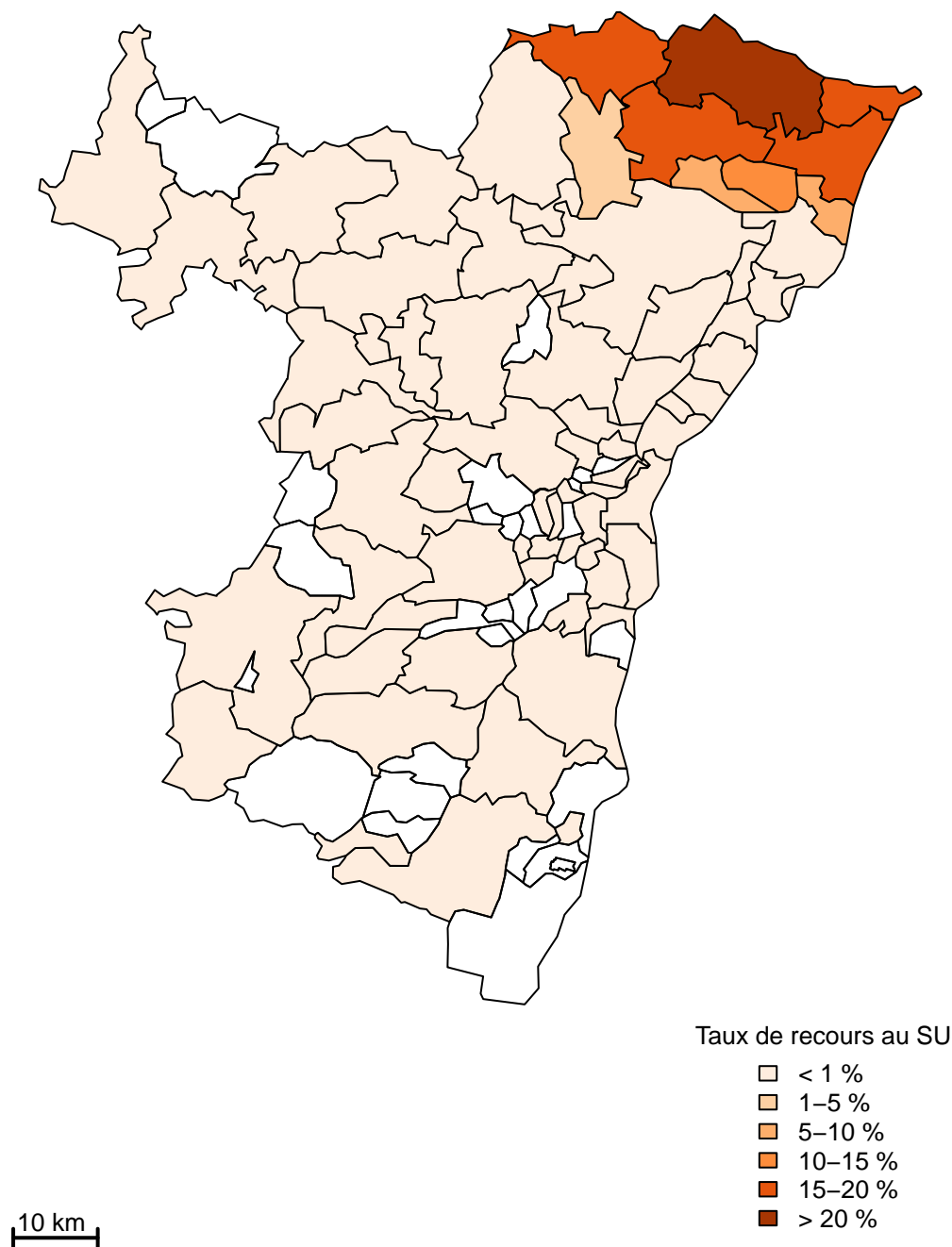
Le programme affiche en jaune les zones où au moins 1 patient à eu recours au SU de la zone et un chiffre correspondant au taux de recours.

Le programme affiche une carte où les CP sont coloriés avec une échelle de couleur progressive en fonction du taux de recours.

3

il n'a pas été possible de conserver leur niveaux de découpage, le taux de recours en Alsace ne dépassant pas 25%.

Taux de recours au SU des HUS



```
# affiche la carte des CP
plot(cp67)
```



```
a <- names(cp.hus)

# colorie les CP en rouge si ont consultés les HUS
plot(cp67, col = ifelse(cp67$ID %in% a, "red", "white"))
```

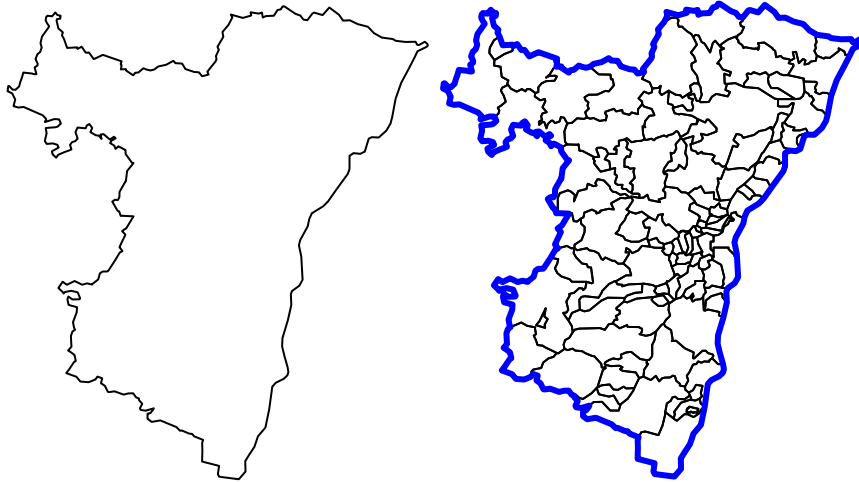


Dessiner avec des nuances de couleur selon la fréquentation

```
## palette de 5 couleurs
# greypal <- heat.colors(5)
## découpage en quintiles
# q <- quantile(cp.hus$RPU, probs = seq(0, 1, 0.2), names = TRUE)
# head(q)
# q2 <- as.character(cut(cp.hus$RPU, breaks = q, labels = greypal, include.lowest = TRUE, right = FALSE))
# plot(cp67, col = ifelse(cp67$ID %in% a, q2, "white"))
```

Fusion des CP

```
## Checking rgeos availability: TRUE
```



Récupérer les coordonnées des polygones

La méthode **coordonates** récupère les coordonnées des objets contenus dans l'objet spatial.

```
## [1] "V1" "V2"

##          V1      V2
## 2538 1046575 6846231
## 2787 1039915 6882911
## 2788 1039045 6840790
## 2789 1040266 6832372
## 2790 1049302 6830794
## 2791 1049845 6828381
```

Coordonnées du rectangle englobant

La méthode **bbox** retourne une matrice carrée contenant les coordonnées des 4 points limitant le rectangle englobant la carte.

```
##      min      max
## x  988703 1082671
## y  6789870 6895581
```

Cartes de base à mettre en stock

- codes postaux
- communes, cantons, arrondissements
- chef-lieux, préfectures, sous-préfectures
- hopitaux, SAMU, SMUR, Hélicisme, SMUR pédiatriques => cartographie de l'offre de soins
- zones de proximité
- territoires de santé