Clustering spatial et Introduction à R

Université Paris-Est Créteil Val-de-Marne
Serge Lhomme

Maître de conférences en Géographie
http://sergelhomme.fr
serge.lhomme@u-pec.fr

13 mars 2018

Clustering spatial

Introduction

Le partitionnement de données (ou data clustering en anglais) est une méthode descriptive d'analyse de données. Elle vise à diviser un ensemble de données en différents « paquets » (groupes) homogènes.

Plus précisément, les données de chaque sous-ensemble (groupe) partagent des caractéristiques communes, qui correspondent le plus souvent à des critères de proximité (similarité) que l'on définit en introduisant des mesures de distance entre objets.

Pour obtenir un bon partitionnement, il convient de :

- minimiser l'inertie intra-classe pour obtenir des groupes (cluster en anglais) les plus homogènes possibles ;
- maximiser l'inertie inter-classe afin d'obtenir des sous-ensembles bien différenciés.

Le clustering spatial va alors prendre en considération les coordonnées géographiques qui seront considérées comme des variables.

Clustering spatial

Pour faire quoi ?

Il faut bien comprendre que faire du clustering spatial, ce n'est pas simplement cartographier les résultats d'un clustering portant sur des données diverses et variées. C'est prendre en compte l'espace dans le clustering.

L'espace n'étant pas neutre, il peut conditionner les résultats d'un simple clustering. Par exemple, des lieux proches pourront avoir tendance à se ressembler et le clustering dégagera alors de grandes zones pouvant être considérées comme homogènes.

Plus simplement, pour diverses raisons, on peut parfois chercher à limiter le nombre d'entités géographiques étudiées. Regrouper des lieux en se fondant uniquement sur leur localisation peut alors avoir du sens, plutôt que d'utiliser des unités administratives supérieures mal adaptées.

Clustering spatial

Différences entre clustering et discrétisation

La discrétisation est une opération, très utile en cartographie, qui permet de découper en différentes classes une variable qualitative ou quantitative. Cette opération simplifie l'information en regroupant les objets géographiques présentant les mêmes caractéristiques en classes distinctes.

La discrétisation et le clustering poursuivent le même objectif : minimiser la variance intra-classe et maximiser la variance inter-classe.

Néanmoins, la discrétisation peut être considérée comme un cas particulier de clustering limité à une seule variable.

Le passage à une série de variables entraine une problématique relative au calcul de la similarité entre les objets. De plus, les corrélations éventuelles entre les variables ajoutent une problématique supplémentaire.

Pour certaines personnes, ces deux termes sont équivalents.

Clustering spatial

Classification supervisée Vs classification non-supervisée

Le clustering est un ensemble de méthodes de classification parmi d'autres. On distingue les classifications supervisées et non-supervisées.

Pour une classification supervisée, on dispose d'éléments déjà classés, on cherche alors à classer un nouvel élément le mieux possible.

Pour une classification non-supervisée, on dispose d'éléments non classés et on cherche à les regrouper en différentes classes.

Le clustering est une méthode de classification non-supervisée.

Le clustering est donc confronté au problème du choix du nombre de classes.

Clustering spatial

Différentes méthodes

Il existe de multiples méthodes de partitionnement de données (clustering), parmi lesquelles :

- Les méthodes fondées sur les centroides telles que les algorithmes des k-moyennes (centres mobiles) ;
- Les méthodes de regroupement hiérarchique (CAH) ;
- Des algorithmes de maximisation de l'espérance (EM) ;
- Des algorithmes fondés sur la densité tels que DBSCAN ou OPTICS ;
- Des méthodes connexionnistes telles que les cartes auto-adaptatives.

Clustering spatial

Le cas particulier des semis de points

Ce cours va traiter du cas particulier de la répartition d'un ensemble de lieux qui correspondent aux différentes localisations d'un phénomène. On parle de semis de points.

Ces lieux peuvent être des habitations, des commerces, des personnes, des clients...

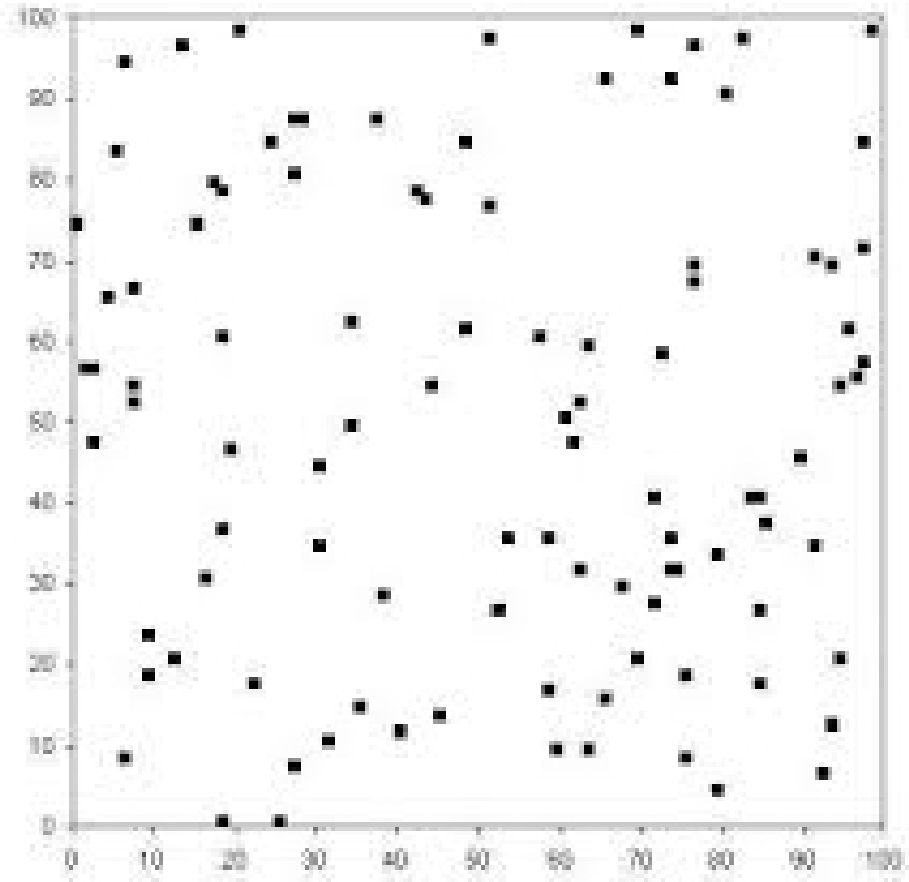
Ces lieux peuvent être traités comme des points à un certain degrés de généralisation.

Pour comparer des semis de points ou pour mettre en exergue certaines spécificités, on va être amené à étudier leur forme. On peut dans ce cadre chercher à les agréger pour simplifier l'analyse et dégager des tendances.

Dans ces analyses, l'espace est souvent considéré comme homogène, les distances entre les points étant déterminées par des distances euclidiennes.

Clustering spatial

Le cas particulier des semis de points



Clustering spatial

Algorithme des k-means : un principe assez simple

Le partitionnement en k-moyennes (ou k-means en anglais) est une méthode de clustering et un problème d'optimisation combinatoire.

Étant donnés des points et un entier k, le problème est de diviser les points en k groupes, souvent appelés clusters, de façon à minimiser une certaine fonction. On considère la distance d'un point à la moyenne des points de son cluster. La fonction à minimiser est la somme des carrés de ces distances.

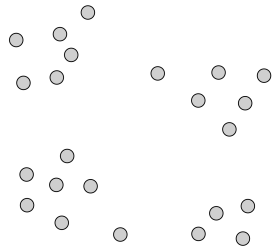
Il existe une heuristique classique pour ce problème, souvent appelée méthode des k-moyennes, utilisée pour la plupart des applications.

Le problème est aussi étudié comme un problème d'optimisation classique, avec par exemple des algorithmes d'approximation.

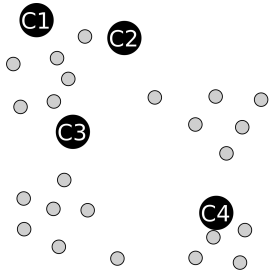
De par sa formalisation, elle est très utilisée pour effectuer des clustering sur des semis de points. Néanmoins, toutes les méthodes de clustering se formalisent bien en prenant comme exemple des semis de points.

Clustering spatial

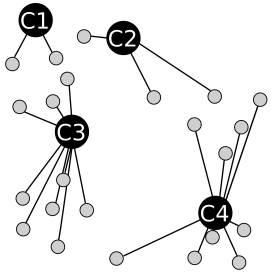
Algorithme des k-means : un principe assez simple



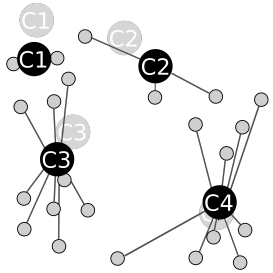
0a. Données d'entrée



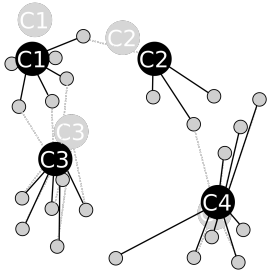
0b. intialisation



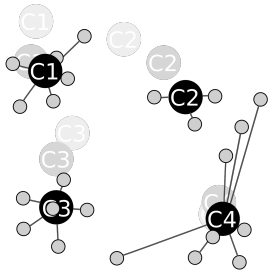
1a. assignation



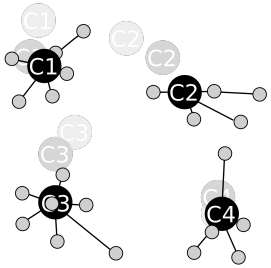
1b. calcul des points moyens



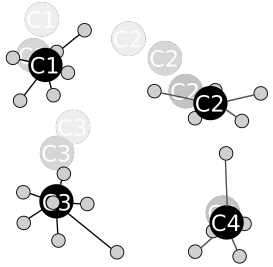
2a. assignation



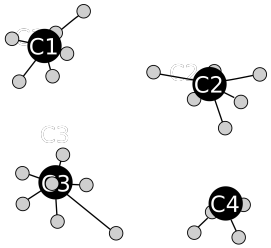
2b. calcul des points moyens



3a. assignation



3b. calcul des points moyens



4a. assignation

clusters stables (fin)

Introduction à R

Présentation

R est un langage de programmation et un logiciel libre dédié aux statistiques et à la science des données soutenu par la R Foundation for Statistical Computing.

Le langage R est largement utilisé par les statisticiens et les data miner pour le développement de logiciels statistiques et l'analyse des données.

Le projet R naît en 1993 comme un projet de recherche de Ross Ihaka et Robert Gentleman à l'université d'Auckland (Nouvelle-Zélande)

La version R 1.0.0, première version officielle du langage R, est publiée le 29 février 2000.

En 2015, plusieurs acteurs économiques importants comme IBM, Microsoft ou encore la société RStudio créent le R Consortium pour soutenir la communauté R et financer des projets autour de ce langage.

Introduction à R

Affectation et calcul

R fonctionne un peu comme une calculatrice. Si vous tapez $2 + 3$, le logiciel vous retournera la valeur 5. Néanmoins, on utilisera R davantage comme un langage de programmation en suivant les principes de l'affectation informatique.

Exemple d'affectation avec R

```
a <- 2
b <- 3
c <- a + b
```

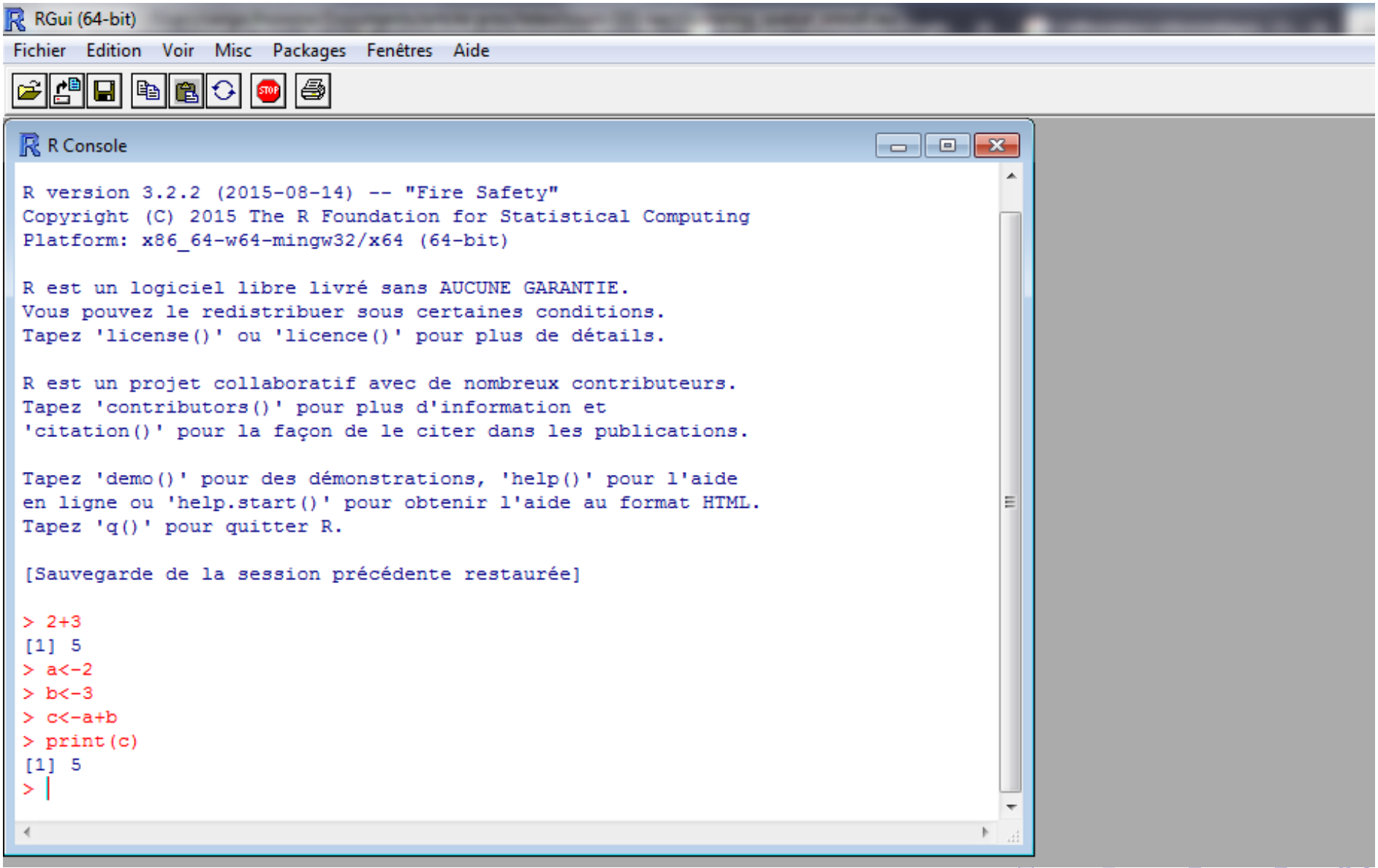
L'affichage des résultats se fera alors en utilisant une fonction : « print() ».

Affichage d'une variable avec R

```
print(c)
```

Introduction à R

Affectation et calcul



Introduction à R

Les types de données

Il existe de nombreux types de variables dans R.

Les variables de type texte

```
a <- "Texte"
```

Ces variables peuvent être ordonnées dans une liste (un vecteur) ou dans plusieurs listes pour former une matrice (un tableau de valeurs).

Les vecteurs et les matrices

```
b <- c(18, 182, 1.5, 15, 200, 5)
c <- matrix(c (18, 182, 1.5, 15, 200, 5), nrow = 2)
d <- matrix(c (18, 182, 1.5, 15, 200, 5), ncol = 2)
```

Introduction à R

Les types de données

Pour accéder à une valeur ou à un ensemble de valeurs, il faut utiliser les index des vecteurs ou des matrices.

Accès aux valeurs des vecteurs et des matrices

```
e <- b[2] + b[3]
f <- c[1,2] + c[2,3]
col <- c[,1]
ligne <- c[1,]
```

Accès avancé aux valeurs des vecteurs et des matrices

```
e <- b[c(2,4)]
f <- c[(c<15)]
g <- b[2 :5]
```

Introduction à R

Les types de données

Les data frames permettent de manipuler des tableaux bien structurés. Ce type de données est particulièrement bien adapté aux importations de fichiers textes.

Les Data Frames

```
articles <- c( "un", "le", "la", "les")
sujets <- c( "mot", "terme", "chose", "images")
dfmots <- data.frame(articles, sujets)
dfmots2 <- data.frame(col1 = articles, col2 = sujets)
```

Appel des valeurs des Data Frames

```
print(dfmots$sujets)
print(dfmots[,1])
```

Introduction à R

L'import de données et premières fonctions

Importation de fichiers textes

```
MyTexte <- read.table(file="c :/TheData.csv", header=TRUE, sep=",")
MyData <- read.csv(file="c :/TheData.csv", header=TRUE, sep=",")
fichier <- file.choose()
```

Fonctions de base

```
res <- summary(b)
plot(d[,1],d[,2])
hist(b)
reg <- lm(d[,1] ~d[,2])
res3 <- summary(reg)
t.test(d[,1], d[,2])
```

Introduction à R

Les bibliothèques

Ce qui constitue la puissance de R, ce sont ses nombreuses bibliothèques qu'il faut télécharger.

Les librairies cartographiques

```
library(rgdal)
nuts3 <- readOGR(dsn = "data", layer = "nuts3", verbose = TRUE)
library(sp)
class(nuts3)
nuts3@proj4string
head(nuts3@data)
plot(nuts3[1, ], col = "#5C99AD", border = " #2A5F70", lwd = 4)
library(rgeos)
europeBuffer <- gBuffer(spgeom = europe, width = 50000)
```

Introduction à R

Les boucles et la programmation

Enfin, comme tout langage de programmation, R permet de répéter les mêmes instructions plusieurs fois en changeant seulement quelques paramètres. Ce sont les boucles. Ces boucles peuvent alors permettre d'effectuer des tests. Ce sont par exemple les Si.

Les boucles

```
for (i in 1 :10) {
        print(i)
        }
for (i in 1 :10) {
        if (i > 5 & i < 8) {
                print(i)
                }
        }
```

Application de R au clustering

Utilisation de la fonction kmeans

Les techniques de clustering font parties des techniques essentielles de l'analyse statistique, par conséquent R propose par défaut des fonctions de clustering, notamment la fonction « kmeans ».

Avoir des informations sur la fonction kmeans

```
help(kmeans)
```

En entrée, il faut au minimum préciser :

- la matrice (ou la data frame) qui contient les variables sur lesquelles effectuer le clustering ;
- le nombre de clusters.

En sortie, la fonction renvoie notamment les clusters associés à chaque objet, les centres des clusters.

Application de R au clustering

D'autres logiciels permettant de faire du clustering

Scilab : Logiciel libre, permettant de faire de nombreux calculs mathématiques. Il s'appuie sur un langage de programmation.

MATLAB : Logiciel propriétaire comparable à Scilab.

Tanagra : Logiciel gratuit de Data Mining spécialisé dans les méthodes de fouilles de données issues du domaine de la statistique exploratoire et de l'apprentissage automatique. Il s'appuie sur le principe de la programmation visuelle.

Stata : Logiciel libre comparable à R, spécialisé dans l'économétrie.

SAS : Logiciel propriétaire comparable à R.

SPSS : Logiciel propriétaire utilisé pour l'analyse statistique et s'appuyant sur une interface graphique comparable à Excel.

XLSTAT : Logiciel propriétaire qui propose des fonctions avancées par rapport à Excel. L'intégration entre les deux logiciels est parfaite.

Application de R au clustering

TD

1. A partir du fichier CSV des communes d'Ile-de-France, effectuez une classification k-means pour les regrouper en 4 clusters.
2. Représentez les résultats sous la forme d'un graphique.
3. Automatisez le processus à l'aide d'une boucle pour produire 10 images de classification correspondant à différents nombres de clusters.
4. A partir du shapefile des communes du département de l'Ain, calculez les centroides de ces communes, puis effectuez une classification k-means pour les regrouper en 3 clusters.
5. Représentez les résultats sous la forme de graphiques.
6. Faites en sorte que la taille des centres des clusters soit proportionnel au nombre de communes de ces clusters.
7. Pour aller plus loin, R propose une bibliothèque permettant de faire du DBSCAN, à vous de jouer...