TP R de Statistiques sur l'analyse multivariée: AFC, ACP, CAH, k-means et AFCM

Emmanuel Rachelson and Matthieu Vignes

9 octobre 2013, SupAero - ISAE

1 Présidentielles 2008 - AFC

Récupérer les données, préparer R

- Télécharger les données depuis :
  `http://carlit.toulouse.inra.fr/wikiz/images/e/eb/PresidentielleREMOVETXT.CSV.TXT`.
- Les charger dans R: `read.csv("Presidentielle.CSV",row.names=1)`.
- Installer les packages R `"vcd"`, `"ade4"`, `"graphics"` et `"FactoMineR"` (fonction `install.packages()`)

Un coup d'oeil sur les données

- Quelles sont les variables du tableau de données ?
- Quelle est leur nature ? Leurs modalités ?
- En rappelant brièvement ce qu'est une Analyse Factorielle des Correspondances, dire sur quelles modalités va porter l'analyse.
- Avec la fonction `summary()`, obtenir les stats descriptives des variables.
- Représenter graphiquement les données. Idées: utiliser les fonctions `assoc()` (package `"vcd"`), `mosaicplot()` (package `"graphics"`) et `table.value()` (package `"ade4"`).

Réalisation de l'AFC

- Avec la fonction `CA()` du package `"FactoMineR"`, effectuer l'analyse factorielle des correspondances.
- Construire le diagramme des valeurs propres. Le diagramme des valeurs propres cumulées. Par combien d'axes l'information est-elle résumée de manière très satisfaisante ?

- Représenter les plans factoriels définis par les 4 premiers axes principaux.
- Des commentaires sur la qualité des représentations ainsi obtenues ? Par exemple en analysant les sorties `...$(col/row)$(coord/cos2/contrib)` de l'AFC effectuée avec la fonction `CA()`.
- Etudier les sorties de la fonction `dimdesc()` pour avoir une description automatique des axes de l'AFC dans notre étude.

2 Données hôtels - clustering hiérarchique ascendant, ACP et *K-means*

Récupérer les données, préparer R

- Télécharger les données depuis :

  `http://carlit.toulouse.inra.fr/wikiz/images/6/64/HotelsREMOVETXT.CSV.TXT`.
- Les charger dans R: `read.csv("Hotels.CSV",row.names=1)`.
- Installer les packages R `"cluster"`, `"fpc"`, `"vegan"` et `"FactoMineR"` (fonction `install.packages()`)

Prise en main des données et premières analyses

- Quelles sont les différentes variables ? Quelle est leur nature ?
- Qui sont les individus et les variables sur qui on va faire porter la classi cation hiérarchique ascendante ?
- Obtenir les statistiques descriptives, les covariances et les corrélations entre les variables quantitatives du jeu de donnéees.
- Créer ensuite le graphique en étoile des hôtels (voir la fonction `stars()`).

CAH

- Faire la classification hiérarchique ascendante des observations en utilisant les distances euclidienne et Manhattan et les liaisons simple, complète et de Ward. Aide: library `"cluster"` et fonctions `agnes(), as.hclust(), as.dendogram(), split(), cutree()...`
- Faire la classification hiérarchique ascendante des variables en utilisant les distances euclidienne et Manhattan et les liaisons simples, complètes et de Ward.

K-means

- Obtenir la classification des hôtels en trois groupes à l'aide de la méthode des *K-means* qui portera sur toutes les variables du tableau (`kmeans()`. Représenter graphiquement les trois groupes sur le premier plan factoriel d'une ACP (`PCA()` du package `"FactoMineR"`). Qu'observe-t-on ? Comment se répartissent les groupes ?

- Obtenir la classification des hôtels en 3 groupes à l'aide de la méthode des *K-means* qui portera cette fois sur les coordonnées des hôtels dans le premier plan factoriel. Représenter graphiquement ces trois nouveaux groupes sur le premier plan factoriel. Qu'observe-t-on ? Comment se répartissent les groupes ?

- Quelles sont les différences de classement entre les 2 classements des 2 poins précédents ? Le premier plan factoriel traduit-il fidèlement l'ensemble des données ? On pourra se réferrer au diagramme des valeurs propres.

- On décide de vérifier si l'attribution des étoiles est conforme aux critères de constitution des groupes par la méthode des *K-means*. Puisqu'il existe 6 catégories d'étoiles, de 0 à 5, classer les hôtels en 6 groupes à l'aide de la méthode des *K-means* portant cette fois-ci sur toutes les variables à l'exclusion de la variable prix. Attention les groupes obtenus ne sont pas nécessairement numérotés par ordre croissant des étoiles.

- Optionnel : étudier un critère qui permet de choisir le nombre de groupes à constituer dans un *K-means* lorsqu'on n'a pas de connaissance *a priori* pour privilégier un nombre de groupes particulier. En général, l'idée est de calculer les partitions et un critère de *fit* pour les classifications obtenues entre 2 valeurs maximum et minimum du nimbre de classes et de retenir le nombre de classes pour lequel le critère est optimisé. On pourra regarder la fonction `cascadeKM()` du package `"vegan"` (avec un `plot()` associé) et en particulier les critères de Calinski et Harabasz ou SSI (*Simple Structure Index*). Le premier crière est défini par $\frac{SSB/(K-1)}{SSW/(n-K)}$, avec n le nombre de valeurs dans le jeu de données, K le nombre de groupes, SSW la somme du carré des distances intra-groupes et et SSB la somme du carré des distances inter-groupes. C'est une statistique F de comparaison de variances. Le second critère est une combinaison multiplicative de 3 indices qui mesurent l'aspect compact des classes.

3 Deux exemples d'AFCM

Jeu de données fictif sur les préférences selon le sexe et le revenu

On joue ici avec les données fictives ci-dessous pour comprendre le fonctionnement de l'Analyse Factorielle des Correspondances multiples (AFCM). Rapidement, une AFCM est une AFC sur un tableau complet c'est à dire où chaque variable est "décomposée" en chacune de ses modalités et où chaque individu prend la valuer 1 ou 0 sur cette variable composite selon qu'il a la modalité pour la variable étudiée. Le jeu de données que nosu analyserons comporte les réponses de 10 personnes aux 3 questions suivantes:

1. êtes vous un homme ou une femme ?
2. Quel est votre niveau de revenus: moyen ou élevé ?
3. Quel est votre dessert favori: fruit (A), glace (B) ou chocolat (C) ?

	Sexe	Revenu	Préférence
1	F	M	A
2	F	M	A
3	F	F	B
4	F	F	C
5	F	F	C
6	M	F	C
7	M	F	B
8	M	M	B
9	M	M	B
10	M	M	A

1. Créer les données dans R:

```
Sexe <- rep(c("F", "M"), c(5, 5))
Revenu <- rep(c("M", "E", "M"), c(2, 5, 3))
Pref <- c("A", "A", "B", "C", "C", "C", "B", "B", "B", "A")
Resultats <- data.frame(cbind(Sexe, Revenu, Pref))
```

2. "A l'oeil", vous voyez quelque chose se dégager ?
3. Quelle forme a `Resultats` ? Obtenir la table de contingence avec la fonction `table()` et le tableau de Burt ainsi que le tableau disjonctif complet (en vous renseignant pour savoir ce que c'est !). Vous pourrez soit créer les fonctions vous-mêmes, soit utiliser les fonction `acm.burt()` et `acm.disjonctif()` du package `"ade4"` permettent de créer les tableaux de Burt et disjonctif complet. La fonction `tab.disjonctif()` du package `"FactoMineR"` permet de créer le disjonctif complet.

4. Représenter graphiquement les données avec `mosaicplot()` ou `assoc()`.

5. Réaliser l'AFCM du tableau `Resultats` en faisant soit une AFC sur le tableau de Burt soit une AFC sur le tableau disjonctif complet (fonction `CA()`. Limiter l'analyse aux 4 premières valeurs propres. Pourquoi ne pas aller voir plus loin ?

6. Les sorties obtenues par ces 2 méthodes sont différentes; quels liens ? Et les sorties graphiques ?

7. Réaliser une AFCM du tableau initial `Resultats` avec la fonction `MCA()` du package `"FactoMineR"`. A laquelle des 2 approches ci-avant s'apparente-t-elle ?

Admission d'étudiants à l'université de Californie *Berkeley*

Le jeu de données `UCBAdmissions` est un tableau tridimensionnel avec 4526 observations de 3 variables sur les étudiants ayant postulé Ã l'admission à l'université de Californie *Berkeley* en 1973. Les modalités des variables étudiées sont rassemblées dans le tableau ci-dessous:

Variable #	Name(Eng)	Levels(Eng)	Nom(Fr)	Modalités(Fr)
1	Admit	admitted, rejected	Résultat d'admission	admis, rejeté
2	Gender	male, female	Sexe	homme, femme
3	Dept	A, B, C, D, E, F	Département	A, B, C, D, E, F

1. Le jeu de données `UCBAdmissions` est directement disponible dans R. Représenter les données avec les fonctions `assoc()` et `mosaicplot()`.

2. Procéder à l'analyse des correspondances multiples avec la fonction `MCA()` du package `"FactoMineR"`. Quel est le problème ? Afficher le code de la fonction `MCA()`. La premièere instruction de cette fonction est `X <- as.data.frame(X)`. La fonction `MCA()` réalise donc l'AFCM du tableau `data.frame(UCBAdmissions)`. Affcher ce tableau et identifier la cause de l'échec de la première analyse factorielle des correspondances multiples.

3. Transformer la table `UCBAdmissions` en un tableau disjonctif complet à l'aide d'une fonction `expand.dft()` que vous créerez et qui permet de retrouver le tableau des observations à partir du tableau de contingence.

4. Procéder alors à l'analyse factorielle des données multiples du jeu de données.