

Aide sur le logiciel R

Pour un bon apprentissage de R, nous pensons qu'il faut des bases solides et vous pouvez voir le livre *Statistique avec R*¹ pour bien comprendre le fonctionnement de R. Cependant, pour ce MOOC, il ne s'agit pas d'être un utilisateur averti mais de connaître quelques notions essentielles qui sont reprises ici.

Table des matières

1	Installer R	2
2	Ouvrir et fermer R	2
3	L'invite de commande	2
4	L'affectation, les objets et les fonctions	3
5	Sélection	3
6	Installer un package	4
7	Installer un menu déroulant	4
7.1	Installer le menu déroulant général : le package Rcmdr	4
7.2	Installer un complément au menu déroulant général : le package RcmdrPlugin.FactoMineR	4
8	Importer (ou saisir) des données	5
8.1	Importer des données depuis R	5
8.2	Importer des données avec le menu déroulant	5
9	Graphiques	6
10	Analyses statistiques	7

1. Statistique avec R. (2013). Cornillon *et al.*, 296 pages, 3ème édition, Presses Universitaires de Rennes

1 Installer R

Il suffit d'aller sur la page du CRAN <http://cran.r-project.org/> et d'installer la version de R propre au système d'exploitation de l'ordinateur (Windows, Mac OS X, Linux). Une documentation complète sur l'installation y est disponible. Vous pouvez utiliser l'environnement Rstudio qui facilite l'utilisation de R. Suivre le lien www.rstudio.com.

2 Ouvrir et fermer R

Sous Windows et Mac OS X, il suffit de chercher R parmi les programmes installés. Sous Linux, on tape simplement R dans une fenêtre de commande. Pour fermer R, on peut utiliser le menu ou taper la commande `q()`. En sortie, le logiciel propose de sauvegarder la session de travail. Ceci permet, lors d'une ouverture ultérieure, de récupérer l'ensemble des objets construits. Mais le plus souvent, on préférera sauvegarder les instructions plutôt que l'ensemble des objets. Pour cela, il convient d'écrire puis de sauvegarder les instructions dans un fichier texte appelé script.

3 L'invite de commande

Quand on lance R, une fenêtre s'ouvre et attend les instructions avec un prompt de la forme `>`. On peut déjà utiliser R comme une machine à calculer, par exemple :

```
> 2+3.2  
[1] 5.2
```

Le `[1]` signifie que la première (et unique) coordonnée du vecteur de résultat vaut `5.2`. Quand R attend la suite de l'instruction, l'invite de commande devient : `+`. Par exemple, si vous tapez :

```
> 1 -
```

R attend la deuxième partie de la soustraction et l'invite de commande est `+`. Pour soustraire 2 à 1, on tape :

```
+ 2  
[1] -1
```

Généralement, il s'agit d'un oubli de `)` ou `"`. Il suffit de proposer une (ou plusieurs) parenthèse(s) ou double guillemet pour terminer la commande.

La combinaison `Ctrl + c` (ou l'icône STOP) permet de stopper toute commande. Il est possible de rappeler les commandes grâce aux flèches vers le haut ou le bas.

4 L'affectation, les objets et les fonctions

R fait des calculs via des fonctions (elles seront notées en marron) :

```
> sqrt(2)
[1] 1.414214
```

On peut affecter les résultats dans des objets plus ou moins complexes grâce à `<-` ou `=`. Ainsi créons le vecteur `x` des valeurs entières de 1 à 6 :

```
> x <- 1:6
> x
[1] 1 2 3 4 5 6
```

L'affichage du contenu est simplement obtenu en tapant le nom de l'objet.

5 Sélection

Pour construire le vecteur `y` des coordonnées 2 et 4 du vecteur `x`, on tape :

```
> y <- x[c(2,4)]
> y
[1] 2 4
```

Les colonnes 1 et 3 d'une matrice `mat` puis les lignes 2 et 5 sont sélectionnées respectivement par :

```
> mat <- matrix(1:15,ncol=3) #création de la matrice
> mat[,c(1,3)]              #sélection des colonnes 1 et 3
> mat[c(2,5),]              #sélection des lignes 2 et 5
```

En combinant, les lignes 4 et 2 pour les colonnes 2 et 3 s'obtiennent comme suit :

```
> mat[c(4,2),c(2,3)]
```

On peut également affecter des objets de nature différente dans des listes :

```
> maliste <- list(vecteur=x,matrice=mat)
```

La composante de nom `matrice` de la liste `maliste` est sélectionnée par

```
> maliste$matrice
```

6 Installer un package

L'installation d'un package est à faire une seule fois. L'utilisateur aura seulement à charger le package à chaque nouvelle session de R à l'aide de la fonction **library**. Pour installer un package disponible sur le CRAN, il faut exécuter la ligne de code suivante :

```
> install.packages(dependencies=TRUE)
```

Puis choisir le site miroir le plus proche de vous et sélectionner le package à installer par exemple **FactoMineR**. Pour pouvoir utiliser le package dans une session R, il suffit d'exécuter :

```
> library(FactoMineR)
```

Il est aussi possible d'installer un package à l'aide des menus déroulants (packages -> installer le(s) package(s)...) sous Windows ou Mac OS X.

Pour connaître l'ensemble des commandes disponibles dans un package, il suffit de taper la ligne de code suivante :

```
> help(package=FactoMineR)
```

7 Installer un menu déroulant

7.1 Installer le menu déroulant général : le package Rcmdr

Le plus simple pour s'initier à R est certainement d'utiliser le package Rcmdr qui permet d'utiliser R avec un menu déroulant. L'intérêt de ce package est aussi de fournir les lignes de code correspondant aux analyses effectuées : on se familiarise ainsi avec la programmation en voyant quelles sont les fonctions employées. Pour utiliser ce package, il faut l'installer puis le lancer avec la ligne de code :

```
> library(Rcmdr)
```

7.2 Installer un complément au menu déroulant général : le package RcmdrPlugin.FactoMineR

Un menu de FactoMineR est disponible dans l'environnement Rcmdr. L'interface proposée est facile d'utilisation et permet de réaliser les analyses, de construire les graphes et de sauvegarder les résultats très facilement. Pour installer l'interface, il faut installer le package RcmdrPlugin.FactoMineR à partir du site du CRAN (voir la section installer un package).

Ensuite, pour utiliser le menu déroulant, il faut lancer Rcmdr

```
> library(Rcmdr)
```

puis charger le plug-in de FactoMineR. Pour cela, dans le menu de Rcmdr, faire **Outils ==> Charger des Plug-in(s) Rcmdr ...** et choisir le plug-in **Rcmdr-Plugin.FactoMineR**. Il faut accepter de relancer Rcmdr pour que les options soient prises en compte.

Si vous voulez avoir le menu de FactoMineR disponible à chaque lancement de Rcmdr (et ne pas avoir à refaire le chargement décrit à la ligne ci-dessus), vous pouvez faire : **Outils ==> Enregistrer les options de Rcmdr**.

Le menu déroulant de FactoMineR détecte la langue utilisée par votre ordinateur et affiche alors le menu en anglais ou en français.

8 Importer (ou saisir) des données

8.1 Importer des données depuis R

En général, la première chose à faire est de saisir ou d'importer des données. Il est facile d'importer des données saisies à partir d'un tableur (Excel par exemple). Le fichier de données contient généralement en première ligne le nom des variables et parfois les identifiants des individus en première colonne. On conseille de sauvegarder ce fichier dans un format facilement exportable de type **txt** ou **csv**. Le fichier **monfichier.csv** contient les valeurs de deux variables **x** et **y** mesurées sur 50 individus. Supposons que ce fichier se trouve à l'emplacement où travaille R. La commande permettant d'importer le fichier et de le mettre dans un objet nommé **don** est :

```
> don <- read.table("monfichier.csv", sep=";", header=TRUE, dec=",")
```

Le signe **<-** indique que le résultat de la fonction **read.table** est affecté à l'objet **don**. Si l'objet existe déjà, il est remplacé par la nouvelle version de **don**. On précise que le séparateur de colonnes est **" ; "**, que le nom des variables est présent sur la première ligne du fichier avec l'argument **header=TRUE** et que le séparateur décimal est **" , "**. On vérifie que l'importation s'est bien passée en résumant le jeu de données :

```
> summary(don)
```

8.2 Importer des données avec le menu déroulant

Pour importer les données, nous conseillons d'utiliser le menu d'importation disponible dans l'onglet **FactoMineR** car ce module d'importation permet de préciser si le nom des lignes (i.e. des individus pour l'ACP) est disponible dans la première colonne du jeu de données (ce qui est fréquent en ACP ou AFC). Pour disposer du menu déroulant de FactoMineR, voir la section précédente. Il faut alors, depuis la fenêtre Rcmdr faire **FactoMineR ==> Lire des données depuis un fichier, le presse-papier ou une URL**. La fenêtre de commande de la figure 1 s'ouvre et vous préciserez alors le nom de l'objet contenant vos données, si le nom des

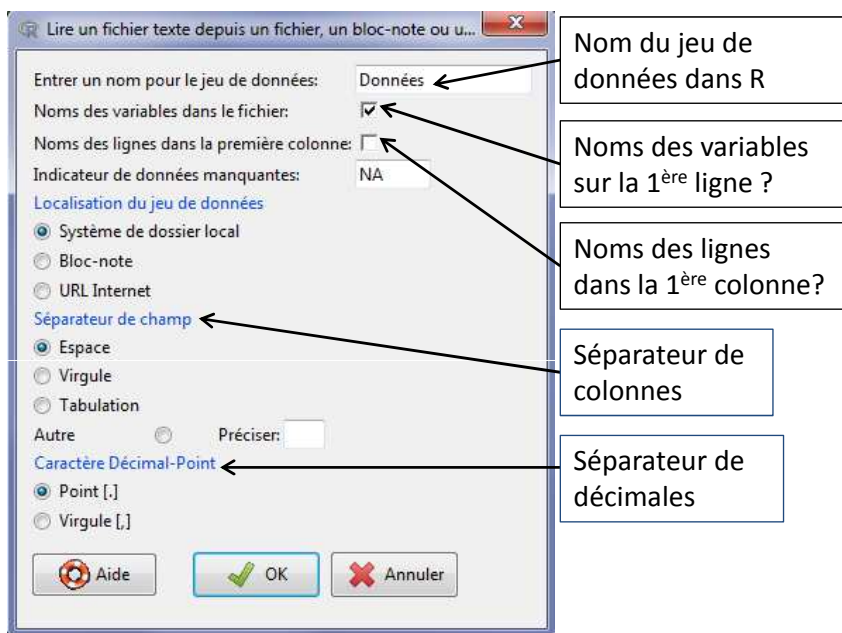


FIGURE 1 – Fenêtre d’importation de données du menu déroulant de FactoMineR

colonnes (des variables pour l’ACP) est disponible dans la première ligne du jeu de données, si le nom des lignes (des individus en ACP) est disponible dans la première colonne du jeu de données, vous préciserez aussi le séparateur de champs, i.e. le séparateur de colonnes, qui est le ";" dans les fichier csv ou la tabulation dans les fichiers txt et enfin vous préciserez le séparateur de décimal ("," ou ";").

9 Graphiques

Dans R, de nombreuses fonctions graphiques sont disponibles. Plusieurs de ces fonctions peuvent être appelées avec la même instruction : **plot**. La fonction **plot** est une fonction générique qui, selon le type d’objets, construira les graphiques adaptés à l’objet. En fait, la fonction **plot** appelle une fonction spécifique à l’objet. Par exemple, si la fonction **plot** est appelée sur un objet PCA (objet résultat d’une ACP) alors c’est la fonction **plot.PCA** qui sera appelée. Si la fonction **plot** est appelée sur un objet CA (obtenu après une analyse des correspondances) ce sera la fonction **plot.CA** qui sera appelée. Ceci est très pratique pour construire les graphiques à l’issue d’une analyse mais pour retrouver l’aide de la fonction réellement utilisée il faudra faire **help(plot.PCA)**.

10 Analyses statistiques

Différentes fonctions permettent de faire des analyses statistiques (intervalle de confiance, test du χ^2 , analyse en composantes principales, etc.). Pour voir comment utiliser une fonction, le plus simple est de se référer à l'aide de la fonction en tapant **help**(nomfonction). On voit ainsi comment elle est définie, l'ensemble de ses arguments et, en fin d'aide, des exemples d'utilisation qu'il est possible de copier-coller dans R. Si les sorties de la fonction sont affectées à un objet nommé **resultat**, il est possible de lister tous les objets présents dans **resultat** par **names**(resultat). Par exemple après l'analyse en composantes principales d'un tableau **X** :

```
> resultat <- PCA(X)
> names(resultat)
[1] "eig" "var" "ind" "svd" "call"
```

La commande **resultat\$ind** donne accès à l'objet contenant les résultats sur les individus. La fonction **summary** permet pour certaines fonctions d'obtenir un résumé des principaux résultats.