

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 *R pour la statistique et la science des données*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 et Rstudio	2
2	L'affectation, les objets et les fonctions	3
3	Sélection	3
4	Installer un package	4
5	Importer (ou saisir) des données	4
6	Analyses statistiques	5

1. R pour la statistique et la science des données. (2018). Husson *et al.*, 416 pages, Presses Universitaires de Rennes

1 Installer R et Rstudio

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. On préconise d'utiliser ensuite l'environnement Rstudio qui facilite l'utilisation de R. Suivre le lien www.rstudio.com. Son installation suppose d'avoir préalablement installé R.

L'environnement RStudio se présente sous la forme d'une fenêtre globale (voir Fig. 1) scindée en 4 sous-fenêtres distinctes :

- la fenêtre de scripts (en haut à gauche),
- la console (en bas à gauche),
- la fenêtre d'environnement et d'historique (en haut à droite),
- la fenêtre des fichiers, graphes, packages et d'aide (en bas à droite).

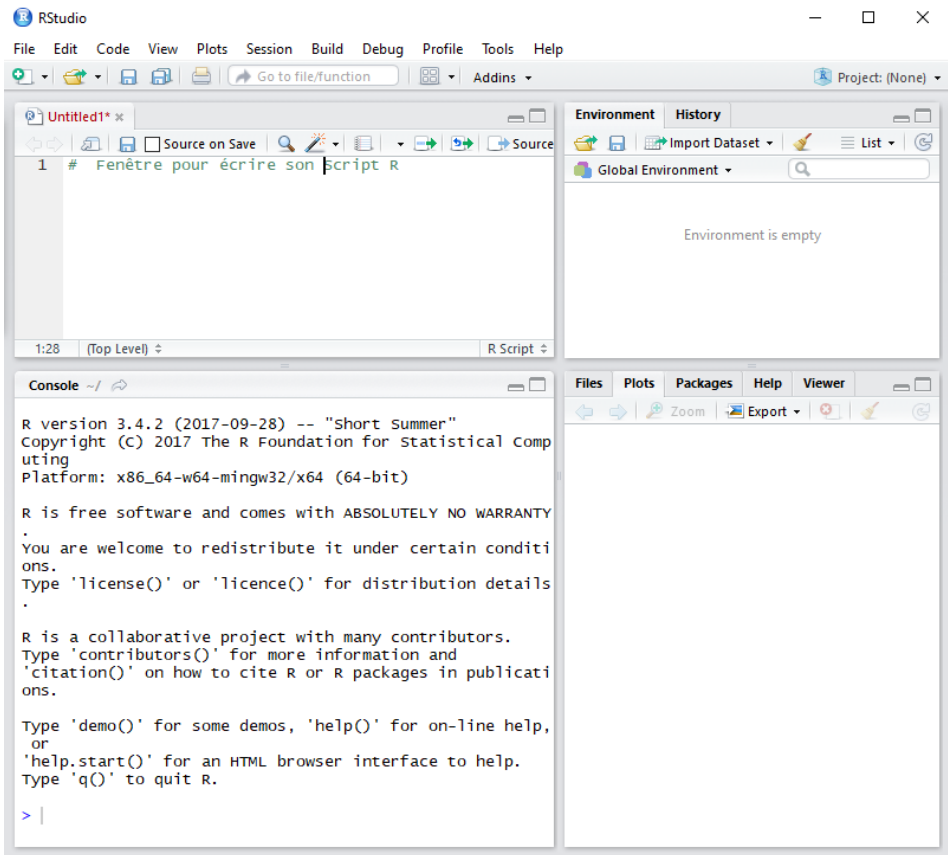


FIGURE 1 – Environnement RStudio.

Pour exécuter du code R écrit dans la fenêtre de scripts, on appuie sur le bouton **Run**. Le code s’affiche dans la console R avec un prompt de la forme : `>`. Par exemple, si on exécute `2+3.2`, on obtient dans la console :

```
> 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 d’une instruction, l’invite de commande devient : `+`. Par exemple, si on exécute `1-`, alors s’affiche dans la console :

```
> 1-
+
```

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

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

Généralement, il s’agit d’un oubli de `)` ou `"`. Il suffit alors de proposer une (ou plusieurs) parenthèse(s) ou double guillemet pour terminer la commande, ou bien taper « Escap » pour récupérer la main.

2 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.

3 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
```

4 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**. Dans RStudio, on peut installer un package à l'aide de l'onglet **Tools -> Install Packages** du menu déroulant, ou encore dans la fenêtre en bas à droite via l'onglet **Packages** puis **Install**.

Une autre possibilité est d'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)
```

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

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

5 Importer (ou saisir) des données

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)
```

6 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 la fenêtre de commandes. 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.