

Introduction à R

R est un logiciel de statistique dédié à l'analyse des données et à leur visualisation. Il contient une collection d'outils pour la statistique, un environnement graphique et un langage de programmation orienté objet. La plupart des entités créées en R sont permanentes. Ces entités sont les objets "données, résultats, fonctions", et sont stockées dans le répertoire `.RData` créé par défaut. Le résultat d'une procédure statistique peut être ainsi réutilisé lors de différentes sessions. Il est donc important de créer un répertoire pour chaque projet statistique effectué en R.

On ouvre une session de R par la commande :

```
$ R
```

Pour clôturer une session, utiliser :

```
> q()
```

L'historique d'une session est conservé dans le fichier `.Rhistory`.

R possède une documentation en ligne accessible par :

```
> help.start()
```

Techniquement, R est un langage fonctionnel. Les commandes élémentaires sont constituées d'expressions et d'affectations. Par exemple :

```
> 2 + 5                                [1] 12 6 10 8
[1] 7                                  > a[2:4]
> a <- c(9,3,7,5)                       [1] 3 7 5
> a                                     > a[a>6]
[1] 9 3 7 5                             [1] 9 7
> a + 3
```

R peut être complété en écrivant de nouvelles fonctions. Voici un exemple où l'on souhaite calculer la statistique $\text{stat.log}(x) = -\frac{1}{n} \sum_{i=1}^n \ln x_i$ où $\forall i, x_i > 0$. On pourra définir une fonction de la façon suivante (même si l'on peut faire bien plus rapide en pratique) :

```
> stat.log <- function(x)
+ {
+   n <- length(x)
+   s <- 0
+   for(i in (1:n)) { s <- s + log(x[i]) }
+   -s/n
+ }
```

La fonction `stat.log` pourra être désormais utilisée comme une fonction standard de R. D'un point de vue pratique, on peut éditer ses fonctions dans un éditeur externe (nedit, emacs, ...) puis faire du copier/coller vers R ou bien utiliser la commande `source`.

Quelques commandes utiles de R

<code>help(mean)</code>	aide sur la commande <code>mean</code>
<code>x <- c(3,14,15,9)</code>	créé un vecteur ligne $x = (3, 14, 15, 9)$
<code>n <- length(x)</code>	taille du vecteur x
<code>sum(x^2)</code>	$\sum_i x_i^2$
<code>mean(x)</code>	moyenne empirique de l'échantillon x
<code>round(x)</code>	valeurs de x arrondies à l'entier le plus proche
<code>seq(from=1,to=10,by=2)</code>	séquence $(1 + 2k ; k \text{ entier}, 1 + 2k \leq 10)$
<code>rep(x,3)</code>	concaténation de 3 répliques du vecteur x
<code>solve(a,b)</code>	solution du système linéaire $ax = b$
<code>diag(x)</code>	matrice diagonale de diagonale x
<code>var(x)</code>	variance estimée $s_n'^2$
<code>sqrt(x)</code>	racine carrée de x , élément par élément.
<code>summary(x)</code>	moyenne, médiane, quartiles et valeurs extrêmes
<code>hist(x)</code>	histogramme de x
<code>sort(x)</code>	tri de x par valeurs croissantes
<code>qqnorm(x)</code>	graphe de probabilités pour la loi normale
<code>plot(x,y)</code>	trace le nuage de points $\{(x_i, y_i)\}_i$
<code>abline(b,a)</code>	superpose au graphique précédent la droite d'équation $y = ax + b$
<code>points(x,z)</code>	superpose au graphique précédent le nuage de points $\{(x_i, z_i)\}_i$
<code>lines(x,z)</code>	superpose au graphique précédent la ligne polygonale reliant les points $\{(x_i, z_i)\}_i$
<code>lm(y~x)</code>	régression linéaire de y sur x
<code>lm(y~x)\$coefficients[2]</code>	pente de la droite de régression
<code>lm(y~x)\$coefficients[1]</code>	ordonnée à l'origine de la droite de régression
<code>lines(x,fitted.values(lm(y~x)))</code>	superpose au graphique précédent la droite de régression
<code>postscript("nom.eps")</code>	redirection de la sortie graphique vers le fichier <code>nom.eps</code>
<code>dev.off()</code>	termine la redirection graphique vers un fichier
<code>par()</code>	contrôle des paramètres graphiques
<code>par(mfcol=c(2,1))</code>	graphique à 2 lignes et 1 colonnes
<code>cat("bonjour","\n")</code>	imprime à l'écran le mot <code>bonjour</code> et retourne à la ligne
<code>source("nom.R")</code>	charge les commandes R contenues dans le fichier <code>nom.R</code> dans R
<code>if, else</code>	structure de contrôle ou d'itération
<code>for, while, repeat</code>	... voir introduction à R d'O. Gimenez, page 71