

Introduction aux fonctions du langage R

Pierre Legendre
Département de sciences biologiques
Université de Montréal

Octobre 2004

R est un système (progiciel) d'analyse statistique et graphique, multi-plateforme et gratuit, créé par Ross Ihaka et Robert Gentleman. R est à la fois un logiciel et un dialecte du langage S de par AT&T Bell Laboratories. S est disponible sous la forme du logiciel S-PLUS commercialisé par la compagnie *Insightful*.

1. Installer R sur un ordinateur

- Aller à l'adresse <http://cran.r-project.org/> sur l'Internet. Ce site se trouve à Vienne.
- Cliquer sur "CRAN" dans la section "Download" (colonne de gauche de la page). Choisir un site de distribution.
- Choisir la version désirée dans la section "Precompiled Binary Distributions". Suivre les instructions. Pour les machines fonctionnant sous Windows, choisir "base" puis télécharger l'exécutable "rw2000.exe". Pour Mac OS X, télécharger le disque-image "R.dmg". Des versions sont également disponibles pour Mac OS 9.x (version fossile), Linux et Unix.

Des manuels sont disponibles dans la section Documentation => Manuals, y compris le guide *R pour débutant* d'Emmanuel Paradis. La documentation interne de R est en anglais.

2. Le langage R et ses commandes

Lorsqu'on démarre R, une fenêtre appelée « console » apparaît à l'écran. On tape une ou plusieurs lignes de commande dans la section du bas de cette fenêtre. Les commandes sont exécutées lorsqu'on appuie sur la touche « retour » ; les commandes exécutées sont transférées dans le haut de la fenêtre. Celle-ci peut être sauvegardée, par exemple sous le nom « RConsole.txt ». On pourra plus tard recopier des portions de ce fichier-texte vers la console afin d'exécuter à nouveau les commandes.

Attribuer un nom à un objet

Les objets utilisés dans le cadre de ce cours se limiteront aux vecteurs, aux matrices et aux « data.frame » (forme de matrice contenant plusieurs types de vecteurs). Un nombre est un vecteur contenant un seul élément dans le langage R.

Pour réaliser les différentes analyses statistiques, nous utiliserons des fonctions du langage R. Certaines de ces fonctions permettent de calculer la moyenne par exemple, tandis que d'autres permettent de construire des graphiques.

On peut construire un vecteur en langage R en utilisant la fonction « c » qui permet de combiner des nombres pour former un vecteur. Par exemple, si on veut construire un vecteur contenant les nombres 1,2 et 3 et que l'on choisit d'appeler ce vecteur « vec1 », la ligne de commande est la suivante :

```
> vec1 <- c(1,2,3)
```

Le « > » en début de ligne est le caractère de sollicitation. Pour donner un nom à un objet, il suffit d'utiliser l'opérateur « assigner » qui est constitué de « < » suivi de « - » :

```
> vec1 <- c(1,2,3)
```

avec ou sans espaces. On peut également écrire :

```
> vec1 = c(1,2,3)
```

Il est important de faire la différence entre le nom d'un objet et la fonction qui permet de le calculer. Par exemple, supposons qu'on désire calculer la moyenne des valeurs qui se trouvent dans un vecteur (objet) « toto ». On importe l'objet dans le langage R à l'aide de « read.table » puis on calcul sa moyenne que l'on peut appeler « moy1 » :

```
> toto <- read.table("essai1.txt", header=T, row.names=1)
```

« read.table » est la fonction qui permet d'importer dans R des données qui se trouvent dans un fichier. « toto » est le nom que l'on donne au tableau de données lues à partir du fichier texte « essai1.txt ».

```
> moy1 <- mean(toto)
```

« moy1 » est le nom donné dans cet exemple à la moyenne des valeurs qui se trouvent dans l'objet « toto ». On la calcule à l'aide de la fonction « mean ». Lors de l'appel d'une fonction, les arguments sont placés entre parenthèses, y compris le nom des objets utilisés par la fonction.

Important : « toto » et « moy1 » sont des noms choisis de façon arbitraire par l'utilisateur. Par contre, « read.table » et « mean » sont des fonction qui permettent de réaliser des calculs. On ne peut ni changer ni traduire leur nom.

On aurait pu donner n'importe quel autre nom que « moy1 » à la moyenne des valeurs qui se trouvent dans l'objet « toto ».

Le nom d'un objet commence obligatoirement par une lettre ; il peut comporter des lettres, des chiffres (0-9) et des points (.). Il faut savoir aussi que, pour les noms d'objets, R distingue les majuscules des minuscules ; ainsi, x et X pourront servir à nommer des objets distincts (voir le manuel *R pour les débutants* par Emmanuel Paradis).

Les opérateurs arithmétiques de R

Tableau 1. Les opérateurs arithmétiques du langage R.

<i>Opérateur</i>	<i>Fonction</i>
+	addition
-	soustraction
*	Multiplication (pour les matrices, ceci est la multiplication élément par élément)
/	Division (pour les matrices, ceci est la division élément par élément)
%%	modulo (reste de la division entière)
%/%	Division entière de deux nombres
%*%	Produit matriciel
^	Puissance
>	Supérieur à
<	Inférieur à
>=	Supérieur ou égal à
<=	Inférieur ou égal à
==	égal
!=	Non égal, différent
&	« et » logique
	« ou » logique

Information sur les fonctions de R

On peut obtenir des informations sur les fonctions de deux façons différentes :

> ?nom-de-la-fonction

Une fenêtre contenant une description de la fonction apparaît à l'écran. Par exemple, pour obtenir une description de la fonction « mean », on tape :

> ?mean

Si on ne connaît pas le nom de la fonction, on peut utiliser la commande « help.search ». Cette commande produit une liste de toutes les fonctions de R dont la description contient l'expression ou le terme indiqué. On peut taper par exemple :

> help.search("median")

Les fenêtres d'aide en ligne sont en anglais.

Notes et commentaires

Il est important d'ajouter des commentaires à la console de façon à se rappeler plus tard de l'objectif des calculs ou encore de la raison de telle ou telle façon de faire. Une ligne de commentaires commence par le caractère « # ». Exemple :

> # Calcul d'une régression multiple

Les lignes de commande commençant par « # » ne sont pas exécutées par R.

Importer un fichier de données en vue d'une analyse par R

On pourrait taper à l'écran toutes les données nécessaires au calcul, mais il est beaucoup plus pratique de les inscrire à l'avance dans un fichier texte, puis de les importer dans R. La fonction « read.table » permet d'importer des données et de créer un objet de type « data.frame ». Exemple :

```
> toto <- read.table("nom-du-fichier-texte", header=T, row.names=1)
```

« toto » est le nom attribué dans la page de travail R au data.frame contenant les données. Le nom-du-fichier-texte est le nom du fichier dans le dossier de votre ordinateur, y compris l'extension si le nom du fichier en comporte une (attention : certaines versions de Windows n'affichent pas les extensions à l'écran). Avant d'importer les données, il faut indiquer à R où se trouve le fichier. Il faut aller dans le menu File ou Tools, selon la version de R, puis sélectionner « Change Working Directory ». On doit naviguer sur le disque et choisir le nom du **dossier** contenant le fichier à importer.

L'argument « header » sert à indiquer s'il y a des noms en en-tête des colonnes du fichier; header=T (*true*) indique qu'il y a des noms de colonnes, header=F (*false*) qu'il n'y en a pas. L'argument « row.names » permet d'indiquer dans quelle colonne se trouvent les noms des objets (lignes). Si les noms des objets se trouvent dans la première colonne, on tape row.names=1. Les valeurs d'une ligne sont séparées par des espaces ou des tabulateurs. Par défaut, R reconnaît le point comme marque des décimales. On devra préciser « dec = "," » si la virgule marque les décimales dans le fichier à importer.

Les arguments « header » et « row names » ne sont pas nécessaires si le fichier est organisé comme suit : les noms d'objets se trouvent en colonne 1 et il y a des en-têtes aux colonnes, sauf pour la colonne des noms d'objets. R est capable de reconnaître les fichiers qui comportent un élément de moins à la première ligne (en-têtes) qu'aux lignes suivantes et il les importe sans précisions supplémentaires. Par exemple, le fichier suivant :

Esp1	Esp2	Esp3	Esp4	Esp5	Esp6	Prof	Corail	Sable	Autre	
Site1	1	0	0	0	0	0	1	0	1	0
Site2	0	0	0	0	0	0	2	0	1	0
Site3	0	1	0	0	0	0	3	0	1	0
Site4	11	4	0	0	8	1	4	0	0	1
Site5	11	5	17	7	0	0	5	1	0	0
Site6	9	6	0	0	6	2	6	0	0	1
Site7	9	7	13	10	0	0	7	1	0	0
Site8	7	8	0	0	4	3	8	0	0	1
Site9	7	9	10	13	0	0	9	1	0	0
Site10	5	10	0	0	2	4	10	0	0	1

sera importé correctement par la commande

```
> recif <- read.table("fich.txt")
```

sans qu'il soit nécessaire de préciser les arguments « header » et « row names ». Il faut tout de même toujours vérifier si les données ont été importées correctement. Pour visualiser les données dans la console R, il suffit de taper le nom de l'objet qui les contient :

```
> recif
```

et appuyez sur retour. R affichera les données à l'écran.

Les librairies de R

R permet de réaliser des calculs à l'aide de fonctions préprogrammées. Ces fonctions sont stockées dans des librairies. La commande

```
> .Library
```

fournit le chemin d'accès aux librairies déjà chargées dans la machine. Si on désire utiliser une fonction d'une librairie qui est présente dans la machine mais qui n'est pas active, il faut l'activer par la commande :

```
library(nom-de-la-librairie)           par exemple           library(MASS)
```

ou encore

```
require(nom-de-la-librairie)          par exemple           require(MASS)
```

Dans cet exemple, MASS est le nom de la librairie de Venables et Ripley correspondant au contenu de leur livre *Modern Applied Statistics with S*. Cette librairie, écrite d'abord pour le langage S, a été traduite en R par Brian Ripley.

On peut apprendre l'existence de librairies contenant des fonctions préprogrammées grâce à la commande « help.search ». Par exemple :

```
> help.search("RDA")
```

indique que la librairie « *vegan* » contient une fonction permettant de calculer l'analyse canonique de redondance (RDA). On peut télécharger des librairies additionnelles, comme « *vegan* », à partir du WWW : menu Packages => Get CRAN Packages => Ready-to-use Binary. Il suffit de choisir parmi la liste des librairies disponibles.

3. Analyses statistiques en R

Presque toute la statistique contemporaine a été programmée en langage R. Jetez par exemple un coup d'œil aux fonctions de la librairie « *stat* » dont on trouve la liste dans le fichier INDEX de cette librairie. Il suffit d'appeler ces fonctions préprogrammées pour réaliser les calculs.

Des exemples d'application aux statistiques de base sont fournis dans le fichier *Calcul des statistiques de base en langage R: exemple Merle*. On y trouve des calculs de statistiques descriptives (moyenne, médiane, variance, etc.) ainsi que la fabrication d'un histogramme de distribution de fréquence et de diagrammes de dispersion.

4. Sauvegarde des résultats

Pour sauvegarder le contenu du fichier console, il suffit d'aller dans File => Save ou Save as... Sauvegardez ce fichier dans votre dossier de travail et non sur le bureau.

On peut sauvegarder les graphiques produits par R dans différents formats qui varient selon la machine utilisée. Il faut activer la fenêtre graphique, puis faire File => Save ou Save as...

5. Conseils pratiques

Plutôt que de sauvegarder le fichier console à la fin d'une session, il est préférable de copier les lignes de commande dans un fichier texte au fur et à mesure du travail.

Inscrivez beaucoup de commentaires dans votre fichier de travail. Vous serez ainsi capable de vous relire. Si les lignes de commentaires commencent par le caractère « # », vous pourrez faire exécuter à nouveau toutes vos commandes en copiant-collant une partie de votre fichier texte, commentaires compris, dans la fenêtre R.

Évitez les accents, les espaces et les virgules dans les noms d'objets : R ne sait pas gérer ces caractères.