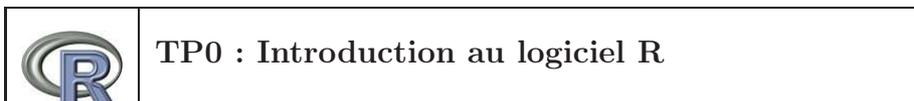


Année : 2008-2009 1er semestre
Niveau : MASTER IS 1 ère année
Cours : Logiciel R
Enseignant : A. Illig



Les travaux pratiques du module LOGICIEL R s'effectueront sous l'interface d'utilisation LINUX (voir Annexe A) et ont pour objectif la découverte et l'apprentissage du logiciel R (voir Annexe B) destiné à l'analyse statistique et graphique.

Table des matières

1	Session introductive	3
1.1	Organisation du travail	3
1.2	Démarrer/quitter une session en R	4
1.3	Exercice d'application	5
2	Aide, librairies, données dans R	7
2.1	Aide en ligne, aide html	7
2.2	Utilisation de librairies	7
2.3	Données R	8
2.4	Illustration graphique	8
A	A propos de LINUX	11
B	A propos de R	13

1 Session introductive

1.1 Organisation du travail

Nous allons diviser l'espace de travail en trois sous-espaces (ou fenêtres) comme résumé sur la figure FIG. 1 :

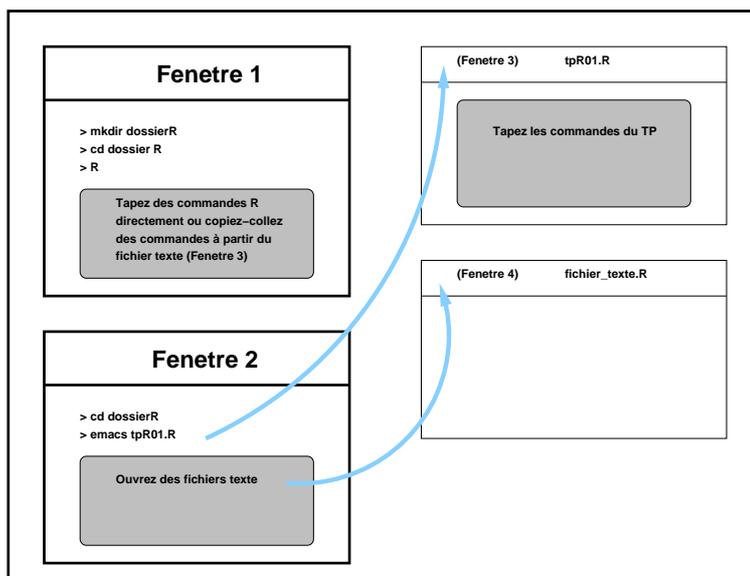


FIG. 1 – Division de l'espace de travail

1. Préalablement à l'ouverture de la première session de travail sous R, vous devez créer un répertoire de travail. Pour cela, ouvrez une première fenêtre *xterm*. Dans cette fenêtre, que nous appellerons Fenêtre 1, tapez ensuite la commande :

```
mkdir dossierR
```

et, pour aller dans le répertoire (ou dossier) nouvellement créé, tapez :

```
cd dossierR
```

Puis, lancez le programme R en tapant directement dans la Fenêtre 1 :

```
R
```

2. Ensuite, ouvrez une deuxième fenêtre *xterm*. Cette nouvelle fenêtre sera appelée fenêtré 2.
3. Enfin, à partir de la Fenêtré 2, placez vous dans le répertoire *dossierR*, puis ouvrez (au moyen d'un éditeur de texte *emacs*, *xemacs*, *asedit*...) un fichier texte appelé *tpR1.R*, dans lequel seront stockées toutes les commandes R à exécuter. Ainsi, la succession de commandes :

```
cd dossierR
emacs tpR1.R
```

tapées dans la Fenêtre 2, positionne d'abord l'utilisateur dans le répertoire de travail depuis lequel le logiciel R a été lancé dans la Fenêtre 1, puis, ouvre une fenêtre *emacs* que nous appellerons Fenêtre 3.

Tout au long du TP, vous pourrez exécuter les commandes R dans la Fenêtre 1 et ouvrir autant de fichiers texte que vous le souhaitez depuis la Fenêtre 2.

1.2 Démarrer/quitter une session en R

Pour lancer le programme R dans un environnement LINUX, il suffit de taper directement la commande R dans une fenêtre *xterm*. Quand le prompt `>` apparaît, vous pouvez commencer à taper des commandes. Pour quitter le programme R, il suffit de taper la commande `q()`.

Par exemple, vous pouvez utiliser R comme une calculatrice en tapant dans la Fenêtre 1 :

```
> 5+26
```

Lorsque vous tapez sur la touche *Enter*, le résultat apparaît :

```
> 5+26
[1] 31
```

Le résultat est précédé de `[1]` pour indiquer à l'utilisateur que le résultat qui suit les crochets est le premier (ici le seul) des résultats attendus. Certaines commandes renvoient plusieurs résultats :

```
> options(width=40)
> 1:20
[1] 1 2 3 4 5 6 7 8 9 10 11 12
[13] 13 14 15 16 17 18 19 20
```

Le résultat est la suite des entiers de 1 à 20 écrite sur deux lignes : le premier résultat (le chiffre 1) suit le crochet `[1]` et le treizième (le nombre 13) suit le crochet `[13]`. Quelques exemples supplémentaires à propos de la calculatrice de R :

```
> # Le symbole * est la multiplication de R.
> # Tout ce qui suit le symbole # est considéré
> # comme un commentaire et n'est pas exécuté
> # par R.
> 5 *3
[1] 15
> 3-8
[1] -5
> 12/2
[1] 6
```

```
> 2^{-3}
[1] 0.125
```

Dans la Fenêtre 3, vous pouvez par exemple taper les commandes R suivantes :

```
# Premier TP de R
# Vecteurs et matrices
Vect=c(1,1,5,7)
Vect
Vect[3]
Mat=matrix(Vect,ncol=2,nrow=2)
Mat
Mat[2,1]
```

Lorsque vous sauvez pour la première fois les commandes tapées dans la Fenêtre 3, un fichier texte du nom *tpR1.R* est créé dans le répertoire *dossierR*. Pour exécuter les commandes du fichier *tpR1.R*, faites un copier-coller ligne à ligne dans la Fenêtre 1. Il est aussi possible de taper directement dans la Fenêtre 1 la commande suivante :

```
> source("tpR1.R")
```

Cette commande permet d'effectuer (sans affichage écran) toutes les commandes du fichier *tpR1.R*. Vous ne verrez cependant pas s'afficher les valeurs de **Vect** et **Mat** bien qu'elles soient stockées en mémoire.

1.3 Exercice d'application

Une personne souhaite emprunter aujourd'hui une somme S au taux d'intérêt mensuel i . Le remboursement de l'emprunt doit s'effectuer en n mensualités d'un montant R . Le montant des mensualité s'obtient en égalisant la somme empruntée P et la somme des mensualités actualisées :

$$P = R(1+i)^{-1} + \dots + R(1+i)^{-n}$$

1. Montrez que $P = R \frac{1 - (1+i)^{-n}}{i}$.
2. Calculez R pour $P = 1500$, $n = 12$ et $i = 1\%$.

2 Aide, librairies, données dans R

2.1 Aide en ligne, aide html

Pour obtenir des renseignements sur une fonction R donnée, par exemple la fonction `plot`, on peut faire appel à l'aide en ligne dans la Fenêtre 1 :

```
> ?plot
```

ou bien

```
> help(plot)
```

Par contre, la commande

```
> help.search(plot)
```

va rechercher toutes les fonctions R où le mot-clef `plot` est utilisé dans le descriptif de cette fonction.

Enfin, une aide au format *html* est également disponible par la commande :

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

qui lance une fenêtre internet depuis laquelle, il est directement possible de rechercher les informations sur la fonction `plot`. En cas d'échec :

```
> options(helphtml=true)
```

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

2.2 Utilisation de librairies

Pour des études (statistiques) spécifiques, il peut être nécessaire de faire appel à un *package* de R bien précis. La commande

```
> library()
```

affiche les *packages* disponibles. Par exemple, pour charger le *package* `graphics` :

```
> library(graphics)
```

Ce *package* est utile pour certaines représentations graphiques comme nous l'apprendrait la commande :

```
> help(graphics)
```

ou la liste des fonctions utilisables dans ce *package* :

```
> library(help=graphics)
```

Pour obtenir la liste des *packages* utilisés, il suffit de taper la commande :

```
> search()
```

2.3 Données R

R est un logiciel qui met à disposition de l'utilisateur une quantité importantes de données qui peuvent être utilisées pour une étude statistique. La liste des données s'affiche par la commande :

```
> data()
```

Pour obtenir des précisions sur les données `cars`, on fait appel à la fonction `help()` :

```
> help(cars)
```

Les données `cars` sont chargées par la commande,

```
> data(cars)
```

On accède alors aux noms des variables composant l'ensemble des données `cars` :

```
> names(cars)
```

Les données `speed` et `dist` sont appelées par `cars$speed` et `cars$dist`. Si l'on "attache" ces données

```
> attach(cars)
```

les variables sont directement accessibles par `speed`, `dist` et la commande

```
> search()
```

permet d'afficher les données (et autres `list` ou `data.frame`) attachées ainsi que les `packages` utilisés. Si l'on détache ces données

```
> detach(cars)
```

elles n'apparaissent plus lors de l'appel à la commande `search()`.

2.4 Illustration graphique

Terminons cette introduction par un exemple graphique :

```
> search()
> # Le package graphics est déjà chargé
> # mais les données cars ne sont pas
> # encore attachées
> attach(cars)
> # Affichons la liste des couleurs
> # disponibles en R
> colors()
> # Ouvrons un dispositif graphique
> x11()
> # Choisissons la couleur du fond d'écran
```

```
> par(bg="turquoise1")
> # Effectuons la représentation graphique
> plot(speed,dist,type="p",pch=25,
col="violetered2",col.main="blue",
col.axis="blue",xlab="Vitesse",
ylab="Distance de freinage",
main="Joli graphique")
> # Détachons les données cars
> detach(cars)
> search()
```

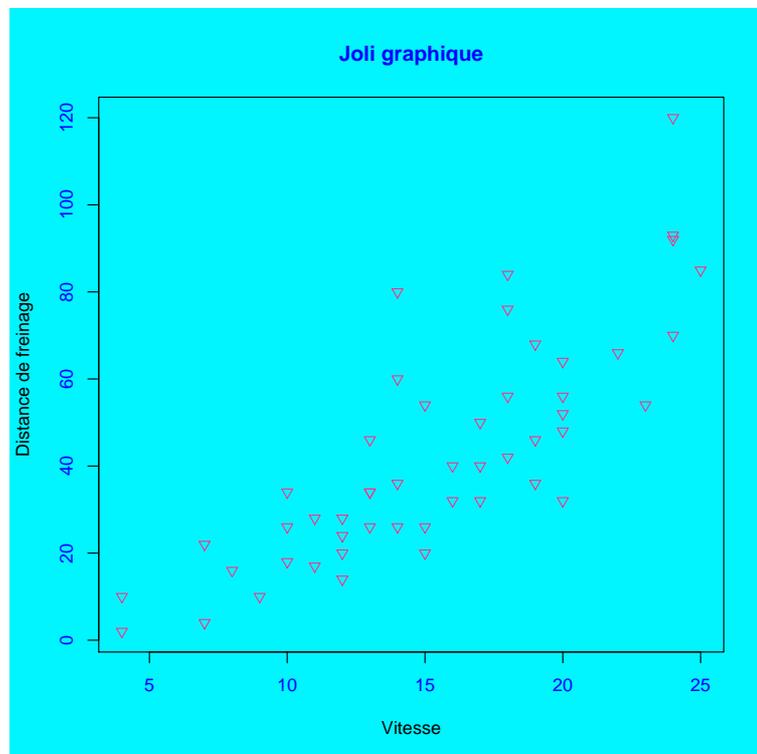


FIG. 2 – Le joli graphique

A A propos de LINUX

Le premier système UNIX a été mis au point par Ken Thompson dans les laboratoires BELL à Murray Hill dans le New Jersey aux Etats-Unis à partir de 1965. Il s'agit d'un système d'exploitation multi-utilisateurs, multi-tâches, ce qui signifie qu'il permet à un ordinateur mono ou multi-processeurs de faire exécuter simultanément plusieurs programmes par un ou plusieurs utilisateurs. Il possède un ou plusieurs interpréteurs de commandes (shell) ainsi qu'un grand nombre de commandes et de nombreux utilitaires (assembleur, compilateurs pour de nombreux langages, traitements de texte, messagerie électronique, ...). De nos jours les systèmes Unix sont très présents dans les milieux professionnels et universitaires grâce à leur grande stabilité, leur niveau de sécurité élevé et le respect des grands standards, notamment en matière de réseau.

Nous utilisons, dans le cadre de ce module, une version améliorée d'UNIX appelée LINUX. Linus B.Torvalds est à l'origine de ce système d'exploitation entièrement libre. Au début des années 90, il voulait mettre au point son propre système d'exploitation pour son projet de fin d'étude. Il avait pour intention de développer une version d'UNIX pouvant être utilisée sur une architecture de type 80386. Le premier clone d'UNIX fonctionnant sur PC a été MINIX, écrit par Andrew Tanenbaum, un système d'exploitation minimal pouvant être utilisé sur PC. Linus Torvalds décida donc d'étendre les possibilités de MINIX, en créant ce qui allait devenir LINUX. Amusées par cette initiative, de nombreuses personnes ont contribué à aider Linus Torvalds à réaliser ce système, si bien qu'en 1991 une première version du système a vu le jour. C'est en mars 1992 qu'a été diffusée la première version ne comportant quasiment aucun bug.

Pour obtenir de plus amples informations sur l'histoire et la genèse du système d'exploitation LINUX, vous pouvez consulter par exemple la page web <http://www.commentcamarche.net/unix/unixintro.php3>. Par ailleurs, vous trouverez en annexe de ce TP les principales commandes UNIX utilisables dans LINUX.

B A propos de R

R est un logiciel dédié à l'analyse statistique qui offre des facilités graphiques considérables. Il peut être vu comme une implémentation du langage S (développé par les laboratoires BELL) à la base du logiciel S-PLUS commercialisé par MathSoft. Contrairement à S-PLUS, R est un logiciel gratuit qui est téléchargeable directement sur le site internet <http://cran.r-project.org/> où se trouvent également les instructions à suivre lors de l'installation pour chaque système d'exploitation. Le logiciel R a initialement été développé dans les années 90 par deux statisticiens : Robert Gentleman et Ross Ihaka de l'Université d'Auckland en Nouvelle Zélande. Depuis, de nombreux mathématiciens ont rejoint la "R Development Core Team" qui tient à jour un site internet <http://www.r-project.org/> où sont notamment rassemblées de nombreuses informations bibliographiques sur le logiciel R. Ainsi, R est un logiciel en constante évolution comme l'atteste le nombre croissant de nouvelles librairies (aussi appelées *packages*) développées parallèlement aux avancées statistiques et mises à la disposition de tous sur le site du projet.