

## Feuille-réponse 1 Prise en main de R

### 1 Prise en main de R

Recherchez le programme R et lancez-le. La fenêtre qui s'ouvre s'appelle la *console*. Elle sert essentiellement à l'affichage des résultats. On peut y saisir une commande, telle la demande d'affichage de la valeur d'une variable (on tape le nom de la variable, puis "return"), mais il convient d'éviter de l'utiliser pour des commandes plus compliquées telles les boucles, etc. car il n'est guère possible d'en demander la réexécution.

Pour saisir une succession de commandes, il convient d'ouvrir une *fenêtre d'édition*. Pour cela utiliser la commande de menu **Fichier** – > **Nouveau script**. Il s'agit d'un assez simple éditeur de texte. Comme pour tout éditeur, il convient de commencer par donner un nom au fichier édité. Pour cela, utiliser la commande de menu **Fichier** – > **Sauvegarder-sous** et choisir le nom **Stat1.r** et éventuellement le disque et répertoire où vous voulez sauvegarder votre séance (votre clé USB par exemple). Il est essentiel de **garder les fichiers** de vos séances : cela ne vous sera pas seulement utile pour vos révisions ; dès la semaine suivante vous pourrez repartir des idées de la semaine précédente pour développer de nouvelles questions. Le nom choisi se compose de **Stat** car il s'agit du cours de Statistique, **1** car il s'agit du premier TP de la série 1, 2, ..., 10, et enfin de **.r**, qui est l'extension réservée pour les *sources* de R. Par la suite il vous suffira de double-cliquer sur un tel fichier-source pour à la fois lancer Scilab (si ce n'est pas déjà fait) puis ouvrir le source, ou *script*, dans l'éditeur de R.

À présent, vous allez taper les instructions successives dans votre fenêtre d'édition, puis marquer (en tirant la souris tout en enfonçant le bouton gauche, ou en cliquant en début de texte et par un Maj-click-gauche en fin de zone à marquer), puis demander l'exécution de la zone marquée par la commande de menu **Édition** – > **Exécuter la ligne ou sélection** (ou simplement en tapant Ctrl+R). Ainsi vous construisez progressivement une succession d'instructions et en testez l'effet en lisant dans la **console** les résultats produits. Vous pourrez ensuite réexécuter tout ou partie de votre code, dans lequel vous aurez peut-être modifié la valeur de quelques constantes.

Avant d'exécuter tout ou partie de votre code, tapez **Ctrl-S** (ou exécutez la commande de menu **Fichier-Sauvegarder**) car il arrive que R plante(!) (ou déjante(!)) c'est-à-dire s'arrête complètement et il ne vous reste plus qu'à repartir, en double-cliquant sur le fichier que vous avez fort heureusement sauvegardé ! Notez que la fenêtre d'édition s'affiche souvent par dessus la fenêtre console, ce qui gêne pour lire les résultats sur cette console. Pour cela, modifiez la place et la taille des deux fenêtres : à gauche une console étroite et à droite un éditeur plus large, les deux s'étirant du haut en bas de votre écran. Évitez tout chevauchement. Faites cela systématiquement (et immédiatement) : vous n'aurez plus à le faire lorsque vous rencontrez une difficulté et qu'il vous faudra toute votre attention sur elle.

### 2 Le fichier rubber

Nous allons travailler sur un fichier de données appelé "rubber.csv" comportant 3 variables portant les noms, **Dates**, **Prices** et **Return** correspondant aux prix et rendements journaliers du caoutchouc relevés chaque jour ouvrable sur une période de 8 années entre le 03/01/2007 et le 27/02/2015. Ce fichier se trouve à l'adresse <http://math.unice.fr/~diener/StatMass/index.htm> . Pour le charger dans votre environnement et lui donner le nom **mydata**, le sauvegarder sur votre machine, puis effectuer la commande

```
mydata<-read.csv2(file.choose())
```

puis retrouver le fichier au moyen du navigateur.

1. Pour connaître l'usage d'une commande de R, il suffit de taper dans la console son nom précédé d'un point d'interrogation. Une page de l'aide en ligne apparaît alors. En utilisant au besoin l'aide en ligne, expliquer en face de chacune des lignes suivantes ce que fait la commande.

```
head(mydata, n=10)
```

```
names(mydata)
```

```
summary(mydata)
```

```
extrait1<-mydata[1:5,2:3]
```

2. En adaptant cette dernière commande, créer un `extrait2` consistant en les 100 premiers prix puis tracer les à l'aide de `plot(extrait2)` ou, plus jolie, `plot(extrait2,type="o")`. Faites de même pour les 100 premiers rendements. Quelle différence observez-vous entre les deux graphiques ?

3. Les commandes suivantes permettent de nommer `x` et `r` les deux variables de prix et de rendements

```
x<-mydata$Prices
```

```
r<-mydata$Return
```

Combien vaut `x[1:5]` ?

Vérifier, par exemple pour  $t = 2$ , que les rendements ont bien été calculés avec la formule  $r_t = (x_t - x_{t-1})/x_{t-1}$ .

Que retourne la commande `r[1]` ? Expliquez.

Que font les commandes suivantes ?

```
min(x)
```

```
max(x)
```

```
length(x)
```

```
mean(x)
```

```
median(x)
```

```
which(x==min(x))
```

```
x[393]
```

4. A quelles dates les prix du caoutchouc étaient maximaux et minimaux ? Donnez les commandes que vous utilisez.



## 4 Choix des classes

1. On choisit à présent la largeur des barres, au moyen de l'option `breaks=`. Exécutez les instructions suivantes :

```
pas=20
```

```
bornes=seq(30,190,pas)
```

```
hist(x,breaks=bornes)
```

Que vaut `bornes` ? Que devient l'histogramme ?

2. Recommencez avec `pas` égal à 40 puis 80. Comparez avec l'histogramme initial. Calculez les effectifs des classes du dernier histogramme

3. Recommencez avec `pas` égal à 10 puis 5. Comparez avec l'histogramme initial.