

# Exercice - pour apprendre du vocabulaire en **R**

Thibault LAURENT

Mise à jour: 02 novembre 2021

## Contents

<b>1 Un peu de vocabulaire en R</b>	<b>2</b>
1.1 Les premières fonctions à apprendre	2
1.2 Les opérateurs importants d'affectation et sélection	2
1.3 Les opérateurs importants de comparaison	2
1.4 Les fonctions mathématiques	2
1.5 Les tests sur les booléens et les ensembles	3
1.6 Manipulation de <b>matrix</b>	3
1.7 Manipulation sur les vecteurs	3
1.8 Manipulation sur les <b>list/data.frame</b>	3
1.9 Manipulation sur les <b>factor</b>	3
1.10 Manipulation de chaînes de caractères	3
1.11 Statistiques	4
1.12 Calcul algébrique	4
1.13 Interroger le répertoire de travail	4
1.14 Lecture/écriture de données	4

Ce document a été généré directement depuis **RStudio** en utilisant l'outil Markdown. La version *.pdf* se trouve ici.

Dans ce document, nous avons répertorié par catégorie, un certain nombre de commandes/fonctions de base, dont un grand nombre ont déjà été présenté dans la première partie du cours.

**Objectif de l'exercice :** chaque participant va choisir un paragraphe et pour chacune des commandes/fonctions de base incluses dans un paragraphe, expliquer leur but et les illustrer par quelques exemples. Vous pourrez bien entendu vous inspirer de l'aide en ligne, mais l'intérêt de l'exercice est de trouver des exemples originaux. Si des commandes/fonctions de base n'ont pas encore été vues en cours, l'aide en ligne devrait vous fournir toute l'information dont vous avez besoin pour comprendre ces fonctions.

**Dans quel format rendre l'exercice :** ici, nous vous recommandons d'utiliser l'outil **R** Markdown qui est accessible depuis **RStudio**. Vous pourrez ainsi créer facilement un fichier au format *.html*, lisible depuis n'importe quel explorateur internet. Vous trouverez dans ce document les informations pour utiliser **R** Markdown : [http://www.thibault.laurent.free.fr/presentation%20inge\\_stat/presentation%2008-07-13/Presentation%20R%20Markdown%20SL%2011072013.pdf](http://www.thibault.laurent.free.fr/presentation%20inge_stat/presentation%2008-07-13/Presentation%20R%20Markdown%20SL%2011072013.pdf)

Pour customiser un document Markdown, on recommande la lecture de cette page <https://holtzy.github.io/Pimp-my-rmd/>

**Où rendre l'exercice ?** vous pourrez déposer le fichier *.html* dans une nouvelle conversation du "Forum **R** (chapitres 1 et 2)". Vous créerez à chaque fois un nouveau post intitulé "Exercice - paragraphe 1.X" comme je le ferai moi-même par un exemple.

**Quel paragraphe choisir ?** vous choisirez le paragraphe que vous souhaitez excepté le paragraphe 1.1. Ce n'est pas grave si certains d'entre vous choisissent le même paragraphe, du moment que vous changez

vos exemples, ainsi que vos commentaires. Je pousse cependant les personnes qui ne sont pas complètement débutantes, à choisir les paragraphes qui contiennent le plus de fonctions qui n'ont pas déjà été vues.

**Faut-il décrire en profondeur chaque fonction ?** Non. Certaines fonctions contiennent énormément de paramètres d'entrée et il est impossible de les décrire tous. L'idée est de donner quelques lignes de description par fonction et deux/trois exemples d'utilisation simples.

**Notation :** cet exercice ne sera pas noté, mais il est fortement conseillé de le faire et de participer à ce travail collectif. A moins que je ne vois des explications qui soient vraiment fausses, je ne prendrais pas nécessairement la peine de vous donner un feedback individuel. En revanche, je vous incite à regarder ce qu'auront fait les autres participants afin que cela vous aide à assimiler du vocabulaire en **R**. N'hésitez pas non plus à réagir aux posts des uns et des autres.

**Date limite de remise :** il n'y a pas de limite pour rendre cet exercice.

## 1 Un peu de vocabulaire en R

### 1.1 Les premières fonctions à apprendre

**Remarque :** j'utiliserai ce paragraphe comme exemple. De votre côté, vous ne pourrez pas le choisir...

```
?  
str
```

### 1.2 Les opérateurs importants d'affectation et sélection

```
%in%, match  
=, <-  
$, [, [[, head, tail, subset
```

### 1.3 Les opérateurs importants de comparaison

```
all.equal, identical  
!=, ==, >, >=, <, <=  
is.na  
is.finite
```

### 1.4 Les fonctions mathématiques

```
*, +, -, /, ^, %%, %/%  
abs, sign  
acos, asin, atan, atan2  
sin, cos, tan  
ceiling, floor, round, trunc, signif  
exp, log, log10, log2, sqrt  
max, min, prod, sum  
cummax, cummin, cumprod, cumsum, diff  
pmax, pmin  
range  
mean, median, cor, sd, var  
rle
```

## 1.5 Les tests sur les booléens et les ensembles

```
&, |, !, xor  
all, any  
intersect, union, setdiff, setequal  
which
```

## 1.6 Manipulation de matrix

```
c, matrix  
  
length, dim, ncol, nrow  
cbind, rbind  
names, colnames, rownames  
t  
diag  
as.matrix
```

## 1.7 Manipulation sur les vecteurs

```
c  
rep, rep_len  
seq, seq_len, seq_along  
rev  
sample  
choose, factorial, combn  
(is/as).(character/numeric/logical/...)
```

## 1.8 Manipulation sur les list/data.frame

```
list, unlist  
data.frame, as.data.frame  
split  
  
lapply, sapply, vapply  
apply  
tapply  
replicate
```

## 1.9 Manipulation sur les factor

```
factor, levels, nlevels  
reorder, relevel  
cut, findInterval  
interaction  
options(stringsAsFactors = FALSE)
```

## 1.10 Manipulation de chaînes de caractères

```
grep, agrep  
gsub  
strsplit
```

```
chartr  
nchar  
tolower, toupper  
substr  
paste  
library(stringr)
```

## 1.11 Statistiques

```
duplicated, unique  
merge  
order, rank, quantile  
sort  
table, ftable
```

## 1.12 Calcul algébrique

```
crossprod, tcrossprod  
eigen, qr, svd  
%*%, %o%, outer  
rcond  
solve
```

## 1.13 Interroger le répertoire de travail

```
ls, exists, rm  
getwd, setwd  
q  
source  
install.packages, library, require
```

## 1.14 Lecture/écriture de données

```
data  
count.fields  
read.csv, write.csv  
read.delim, write.delim  
read.fwf  
readLines, writeLines  
load, save  
library(foreign)
```