

# Logiciel R et programmation

## Introduction

Ewen Gallic

Université de Rennes 1, 2016 - 2017

# Objectifs, horaires



## IT'S IN THE SYLLABUS

This message brought to you by every instructor that ever lived.

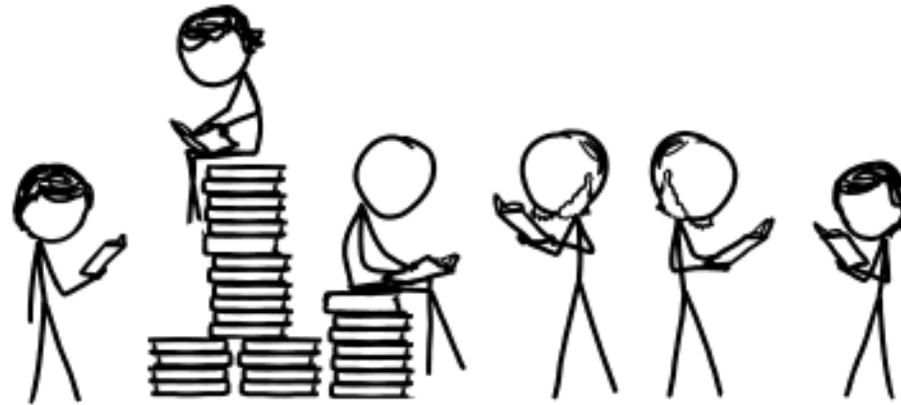
WWW.PHDCOMICS.COM

Source : <http://www.phdcomics.com/comics.php?f=1583>

# Objectifs, horaire

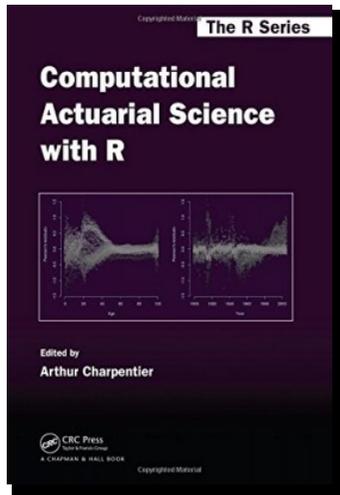
- Objectifs :
  - découvrir le langage et logiciel **R**,
  - apprendre les bases,
  - devenir autonome ;
- Horaire :
  - Attention, horaire variable selon les semaines !
  - 10 Séances ;
- Évaluation :
  - Contrôle continu.

# Références



Source : <https://what-if.xkcd.com/76/>

# Références (1)

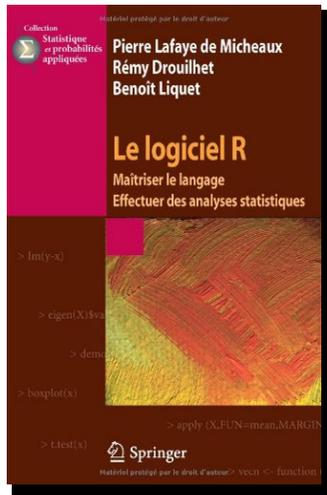


Charpentier, A. (2014). Computational actuarial science with R. Chapman and Hall.

Goulet, V. (2014). Introduction à la programmation en R



# Références (2)



Lafaye de Micheaux, P., Drouilhet, R., & Liquez, B. (2011). Le logiciel R : Maîtriser le langage - effectuer des analyses statistiques. Springer.

Paradis, E. (2002). R pour les débutants.

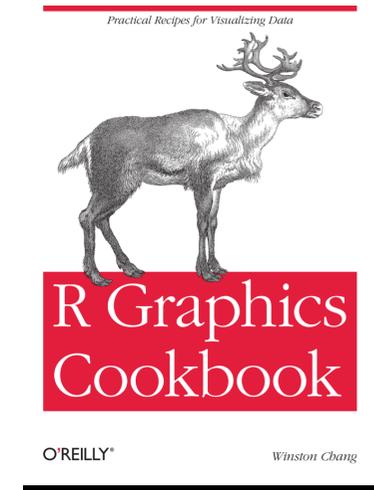


# Références (3)



Wickham, H. (2009). ggplot2 : Elegant graphics for data analysis. Springer.

Chang, W. (2013). R graphics cookbook. O'Reilly Media, Incorporated.



# Historique



Source : <http://www.r-project.org/>

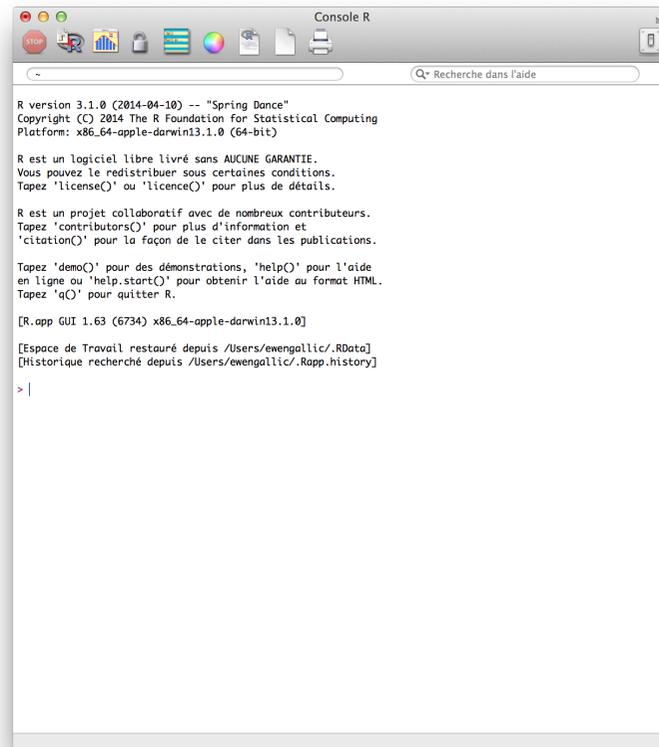
# Historique (1)

- Création de **S** dans les années 1970 :
  - AT&T Bell Laboratories,
  - John Chambers, Douglas Bates, Rick Becker, Bill Cleveland, Trevor Hastie, Daryl Pregibon et Allan Wilks,
  - **manipuler** les données,
  - effectuer des **analyses statistiques**,
  - créer des **graphiques** ;

# Historique (2)

- Création de **R** dans le milieu des années 1990 :
  - département de Statistiques de l'Université d'Auckland,
  - Ross Ihaka et Robert Gentleman,
  - inspiré du S et de Scheme,
  - distribution sous les termes de la General Public Licence,
  - développement et distribution par la **R Development Core Team**,
  - ajouts par la **communauté**.

# Console, fenêtre de script



```
R version 3.1.0 (2014-04-10) -- "Spring Dance"
Copyright (C) 2014 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin13.1.0 (64-bit)

R est un logiciel libre livré sans AUCUNE GARANTIE.
Vous pouvez le redistribuer sous certaines conditions.
Tapez 'license()' ou 'licence()' pour plus de détails.

R est un projet collaboratif avec de nombreux contributeurs.
Tapez 'contributors()' pour plus d'information et
'citation()' pour la façon de le citer dans les publications.

Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.
Tapez 'q()' pour quitter R.

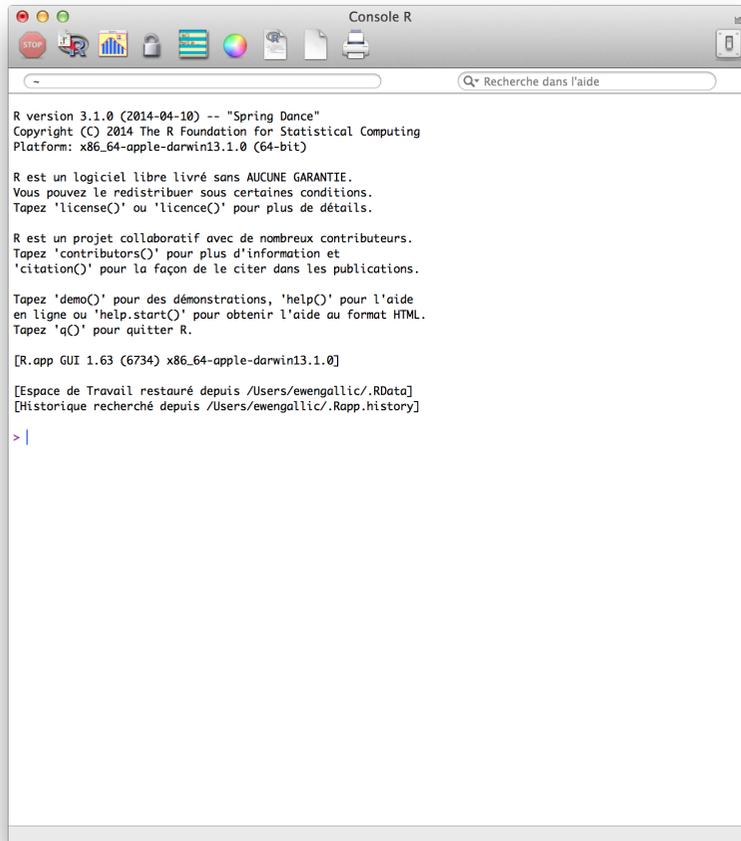
[R.app GUI 1.63 (6734) x86_64-apple-darwin13.1.0]

[Espace de Travail restauré depuis /Users/ewengallic/.RData]
[Historique recherché depuis /Users/ewengallic/.Rapp.history]

> |
```

- R est un langage **interprété** ;
- donc pas de **compilation** ;
- deux méthodes de travail :
  - dans la **console**,
  - dans une **fenêtre de script**.

# La console



```
R version 3.1.0 (2014-04-10) -- "Spring Dance"
Copyright (C) 2014 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin13.1.0 (64-bit)

R est un logiciel libre livré sans AUCUNE GARANTIE.
Vous pouvez le redistribuer sous certaines conditions.
Tapez 'license()' ou 'licence()' pour plus de détails.

R est un projet collaboratif avec de nombreux contributeurs.
Tapez 'contributors()' pour plus d'information et
'citation()' pour la façon de le citer dans les publications.

Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.
Tapez 'q()' pour quitter R.

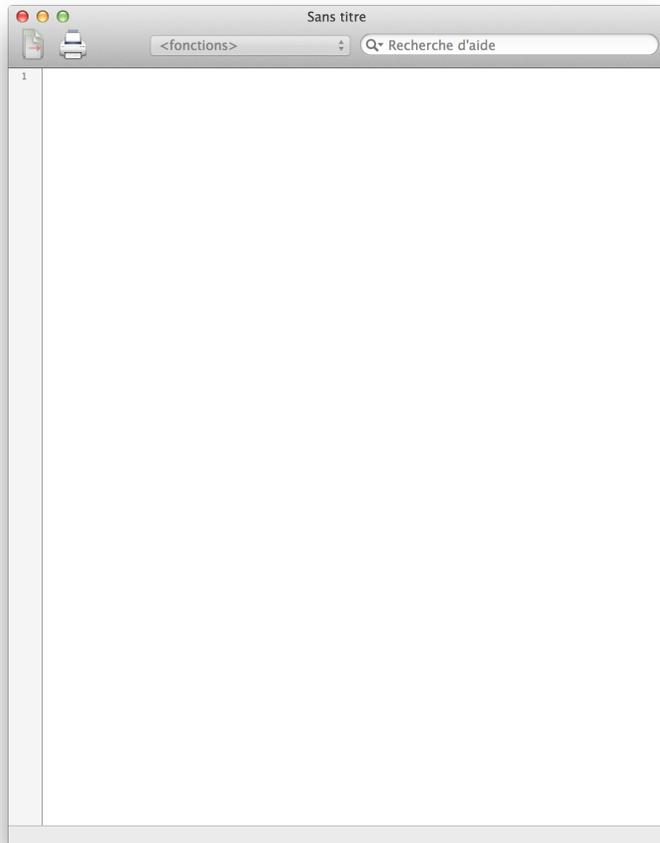
[R.app GUI 1.63 (6734) x86_64-apple-darwin13.1.0]

[Espace de Travail restauré depuis /Users/ewengallic/.RData]
[Historique recherché depuis /Users/ewengallic/.Rapp.history]

> |
```

- Écrire les expressions à la suite du symbole ("``>``") ;
- Valider avec la touche "``ENTREE``" ;
- Résultat également dans la console

# Les fenêtres de script



- Écrire le code dans un fichier ;
- Le soumettre à R ("`CTRL + R`", "`CTRL + ENTREE`", "`CMD + ENTREE`" selon la plateforme et l'éditeur).

# Mise en pratique

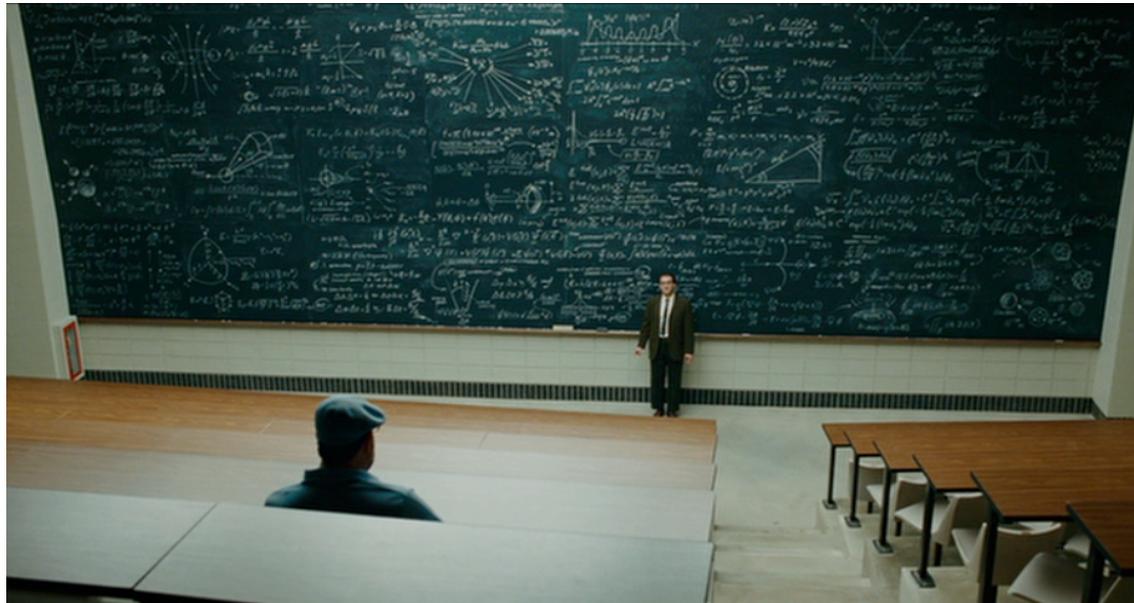
1. Écrire dans la console l'expression suivante, et valider :

```
> 2+1
```

```
## [1] 3
```

1. En faire de même dans un fichier de script.

# Variables



Source : [http://www.focusfeatures.com/a\\_serious\\_man](http://www.focusfeatures.com/a_serious_man)

# Assignment

- résultat d'une évaluation **affiché** mais pas **enregistré**;
- conservation du résultat avec "<-"" ou "->"" ;
- affichage du contenu d'une variable à l'aide de son nom.

# Assignment : exemples

```
x <- 2+1  
x
```

```
## [1] 3
```

```
x^2 -> y  
y
```

```
## [1] 9
```

# Assignment

- contenu non affiché lors de l'assignation ;

```
x <- 2+1
```

- utilisation des **parenthèses** pour forcer l'affichage du contenu.

```
(x <- 2+1)
```

```
## [1] 3
```

# Assignment

- utilisation du symbole "=" autorisée ;
- mais déconseillée.

# Conventions de nommage

- caractères alphanumériques, trait de soulignement, point ;
- ne doit pas commencer par un chiffre ;
- ne doit pas contenir d'espace (sauf si entouré de guillemets) ;
- sensibilité à la casse.

# Conventions de nommage

Dans ce cours, la convention sera la suivante pour le nommage :

- tout en minuscule ;
- espaces remplacées par des traits de soulignement.

# Conventions de nommage : exemple

```
variable_1 <- 2+1  
!variable_2 <- 3+1
```

```
## Error: objet 'variable_2' introuvable
```

```
Variable_1
```

```
## Error: objet 'Variable_1' introuvable
```

```
variable_1
```

```
## [1] 3
```

# Objets en mémoire

- les objets stockés en mémoire sont accessibles avec la fonction `ls()`

```
ls()
```

```
## [1] "encoding" "inputFile" "variable_1" "x" "y"
```

# Modification, suppression

- pour modifier un objet, on utilise à nouveau "`<-`";
- la modification d'une copie n'affecte pas l'original ;
- la suppression s'effectue avec la fonction `rm( )`.

# Modification, suppression : exemples

```
premier_objet <- 2+1
premier_objet <- 2+1
second_objet <- premier_objet # On assigne la valeur de premier_objet
# à second_objet
premier_objet <- 2+2 # On change la valeur de premier_objet
# Puis on affiche le contenu des deux objets :
premier_objet # Le résultat doit être 4
```

```
## [1] 4
```

```
second_objet # Le résultat doit être 3
```

```
## [1] 3
```

# Modification, suppression : exemples

```
premier_objet # L'objet existe
```

```
## [1] 4
```

```
rm(premier_objet) # On le supprime  
premier_objet # Son affichage provoque un message d'erreur
```

```
## Error: objet 'premier_objet' introuvable
```

# Packages



Source : ["Wenger EvoGrip S17" by D-M Commons](#)

# Packages

- les `base` contiennent :
  - des jeux de **fonctions**,
  - des fichiers d'**aide**,
  - éventuellement des jeux de **données** ;
- les fonctions élémentaires sont dans le `base`;
- certains `base` sont chargés par défaut.

# Packages

Liste des packages chargés dans la session par défaut (plus **base**) :

```
getOption("defaultPackages")
```

```
## [1] "datasets" "utils"      "grDevices" "graphics"  "stats"  
## [6] "methods"
```

# Packages : installation, chargement

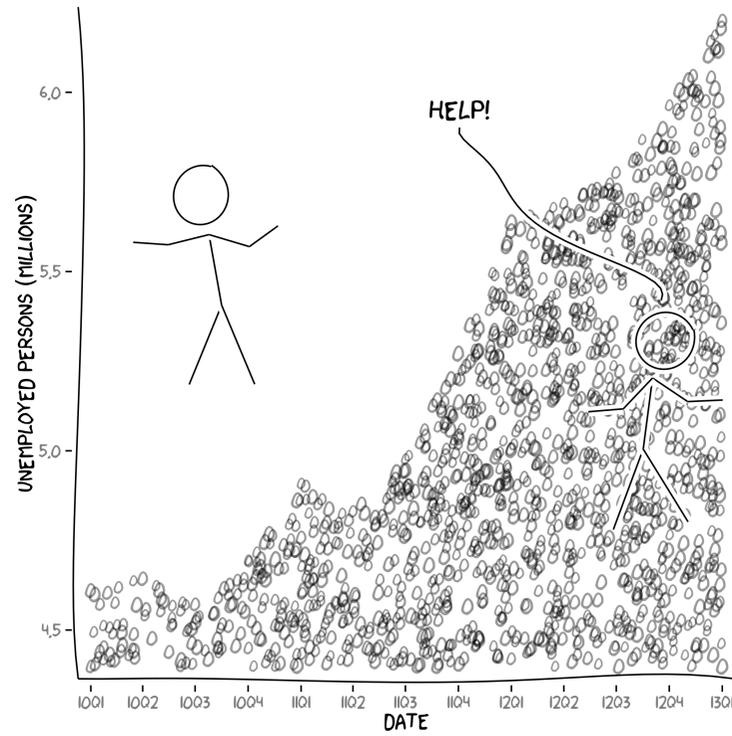
Pour installer un nouveau `package`, le plus simple est d'utiliser la fonction `install.packages()`.

```
install.packages("ggplot2")
```

Pour pouvoir utiliser les fonctions et jeux de données contenues dans un package, il faut le charger.

```
library("ggplot2")
```

# Aide



Source : <http://xkcd.r-forge.r-project.org/>

# Aide

- l'aide en ligne est **simple** d'utilisation ;
- la plupart des fichiers d'aide sont bien **documentés** ;
- **accès direct** à la fiche d'aide avec la fonction `help()` ;
- recherche de fichiers d'aide par **titre** ou **mot-clé** avec la fonction `help.search()`.

# Aide : exemples

Pour trouver des explications sur la fonction `log()` :

```
help("log")
```

```
## Help on topic 'log' was found in the following packages:  
##  
##   Package                Library  
##   base                    /Library/Frameworks/R.framework/Resources/library  
##   spam                    /Library/Frameworks/R.framework/Versions/3.1/Resources/library  
##  
##  
## Using the first match ...
```

De manière équivalente, on peut écrire :

```
?log
```

```
## Help on topic 'log' was found in the following packages:  
##
```

# Aide

Pour rechercher un sujet, un mot-clé :

```
# Pages dont le titre ou le sujet match "logarithm"  
help.search("logarithm")  
# De manière équivalente  
??logarithm  
  
# Pages dont les mots clés contiennent "data"  
help.search(keyword = "data")
```

La liste des mots-clés est disponible ici : <https://svn.r-project.org/R/trunk/doc/KEYWORDS>

# Aide

Quelques **réflexes** à avoir :

- consulter l'aide à partir du logiciel ;
- consulter des ouvrages de référence ;
- consulter les archives des ;
- consulter les questions sur .