



STT 1000 - STATISTIQUES

ARTHUR CHARPENTIER



Logiciels de Statistique

cf [comparaison des logiciels de statistiques](#) sur wikipedia

La connaissance d'un logiciel de programmation/calculs/analyse de données est devenu un élément incontournable pour les chercheurs, praticiens, industriels et donc des étudiants.

Pourquoi R ? Notamment parce qu'il est gratuit, libre, multiplateforme et bénéficie de la plus grande communauté au monde.



Logiciel R/RStudio

RStudio: Logiciel libre, multi-plateforme, gratuit ; Interface graphique permettant un environnement de développement intégré (IDE).

Un IDE n'est pas une interface graphique au sens de SPSS ou SAS, qui permettrait d'utiliser le logiciel à travers des menus et des boîtes de dialogue : il s'agit d'un environnement facilitant la saisie, l'exécution de code, la visualisation des résultats, etc.

Foire aux Questions

- ▶ *Ai-je besoin d'installer les logiciels R et RStudio sur une machine personnelle ?*

Oui, ça sera plus simple / **Non**, il existe de versions en ligne (<https://rstudio.cloud/>)

- ▶ *Ai-je besoin de maîtriser parfaitement le logiciel R pour réussir le cours avec succès ?*

Non, la maîtrise de R n'est pas une fin en soi (dans ce cours). R est juste un outils (très pratique)

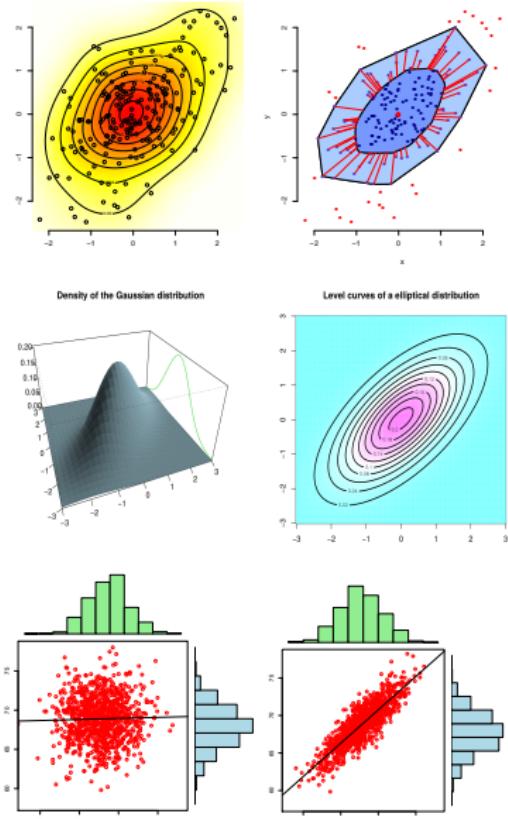
- ▶ *R/RStudio sera-t-il utilisé comme outil d'évaluation ?*

Oui, deux devoirs au cours du semestre seront à préparer et à remettre. Ces devoirs nécessitent l'utilisation du logiciel Rstudio. De plus, les examens pourront contenir des sorties logicielles qu'il conviendra d'interpréter,

S/R Language

Statistiques et graphiques

```
1 > x = c(0,1,2,3,4,5,6,7,8,9,10)
2 > mean(x)
3 [1] 5
4 > quantile(x, .95)
5 95%
6 9.5
7 > quantile(x, .95, type=1)
8 95%
9 10
10 > quantile(x, .95, type=4)
11 95%
12 9.45
13 > quantile(x, .95, type=5)
14 95%
15 9.95
```



RStudio

The screenshot shows the RStudio interface. The top bar includes tabs for 'Untitled1' and 'Addins', and buttons for 'Go to file/function', 'Run', 'Source', and 'File'. The main area has tabs for 'Console' and 'Terminal'. The 'Console' tab displays the R startup message:

```
R version 3.6.3 (2020-02-29) -- "Holding the Windsock"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support is running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/RData]
```

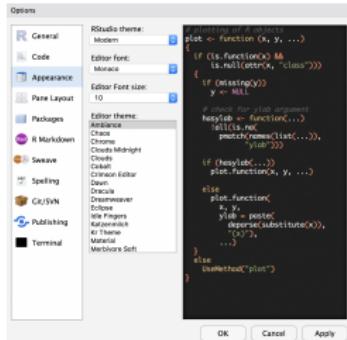
The right side of the interface is the 'Environment' browser, which lists global variables:

Object	Type	Description
A	list	10 obs. of 2 variables
acp	list	List of 5
anova	list	List of 13
AOV	list	List of 13
apartments	list	1000 obs. of 6 variables ALTREIC num [1:45, 1:3] -62.7 807 907.8 -53.7 213...
B	list	7 obs. of 2 variables
b2	list	28 obs. of 5 variables
base	list	891 obs. of 8 variables
BT	list	List of 2
C	list	num [1:3, 1:558] 0.1677 -0.0525 0.3725 0...
coh	list	List of 7
cors	list	num [1:4, 1:4] 1 0.503 0.856 0.493 0.503 -
d	list	28 obs. of 2 variables
dat14boot	list	List of 11
database	list	6 obs. of 32 variables
Davis	list	200 obs. of 5 variables
dc	list	12 obs. of 12 variables
dd	list	50 obs. of 5 variables

Below the environment browser are tabs for 'Files', 'Plots', 'Packages', 'Help', and 'Viewer'.

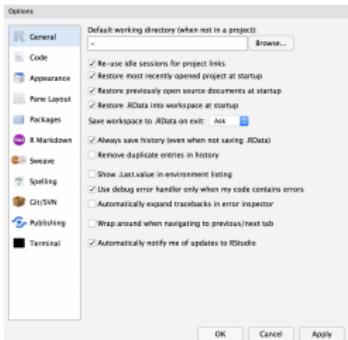
RStudio: appearance

Ma version de RStudio est sombre
on peut la paramétrer...



The screenshot shows the 'Appearance' tab selected in the left sidebar of the Options dialog. Under 'Editor theme', the 'Ambiance' theme is chosen. The 'Editor font' is set to 'Monaco' and the 'Editor Font size' is 10. Other themes listed include 'Aero', 'Clouds', 'Clouds-Dark', 'Cobalt', 'Crimson Editor', 'Dark', 'Dracula', 'Dracula-Darculavore', 'Eclipse', 'Ice Fingers', 'Kittens', 'KrTheme', 'Material', and 'Mercurius Soft'. At the bottom are 'OK', 'Cancel', and 'Apply' buttons.

```
RStudio theme: Modern
Editor font: Monaco
Editor Font size: 10
Editor theme: Ambiance
  plot <- function(x, y, ...)
  {
    if (is.function(x) &
        is.null(attr(x, "class")))
    {
      if (is.missing(y))
        y <- NULL
      # check for ylab argument
      if (!is.function(y))
        warning("y is not a function")
      else
        y <- substitute(y)
      if (is.list(x))
        plot.function(x, y, ...)
      else
        plot.function(
          x, y,
          ylab = y,
          deparse(substitute(x)),
          ...)
    }
    else
      UseMethod("plot")
  }
```



The screenshot shows the 'General' tab selected in the left sidebar of the Options dialog. It contains various configuration options like 'Default working directory', 'Restore idle sessions for project links', and 'Always save history (even when not saving RData)'. At the bottom are 'OK', 'Cancel', and 'Apply' buttons.

```
RStudio theme: Modern
Editor font: Monaco
Editor Font size: 10
Editor theme: Ambiance
  plot <- function(x, y, ...)
  {
    if (is.function(x) &
        is.null(attr(x, "class")))
    {
      if (is.missing(y))
        y <- NULL
      # check for ylab argument
      if (!is.function(y))
        warning("y is not a function")
      else
        y <- substitute(y)
      if (is.list(x))
        plot.function(x, y, ...)
      else
        plot.function(
          x, y,
          ylab = y,
          deparse(substitute(x)),
          ...)
    }
    else
      UseMethod("plot")
  }
```

RStudio

The screenshot shows the RStudio interface. The left pane contains an R Markdown file named 'Untitled1.Rmd' with the following content:

```
title: "Untitled"
author: "Arthur Charpentier"
date: "19/11/2020"
output: html_document
...
````{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
````

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.
```

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
```{r cars}
summary(cars)
```

The right pane shows the Environment browser with the following objects listed:

- A 18 obs. of 2 variables
- acp List of 5
- anova List of 13
- AVG List of 13
- apartments 1000 obs. of 6 variables
- AUTREIC num [1:45, 1:3] -62.7 807 907.8 ...
- B 7 obs. of 2 variables
- b2 29 obs. of 5 variables
- base 891 obs. of 8 variables
- BT List of 2
- C num [1:3, 1:58] 0.1677 -0.0525 0.3725 ...
- coh List of 7
- cars num [1:4, 1:4] 1 0.583 0.856 0.493 0.583 ...
- d 29 obs. of 2 variables
- dat4boot List of 11
- database 6 obs. of 32 variables
- Davis 200 obs. of 5 variables
- dc 12 obs. of 12 variables
- dd 59 obs. of 5 variables

The bottom right pane shows the help viewer for 't.test'.

# RStudio

The screenshot shows the RStudio interface with the following details:

- Title Bar:** Shows "Untitled.Rmd" as the active file.
- Toolbar:** Includes "Knit", "Insert", "Run", and "File" buttons.
- Code Editor:** Displays R Markdown code. The code includes a header with title, author, date, and output type, followed by a chunk setup, a note about R Markdown, and a description of how to embed R code chunks. It ends with a summary command.
- Status Bar:** Shows "8:25" and "Chunk 1: setup" along with a "R Markdown" indicator.

La fenêtre **editor** (haut à gauche)

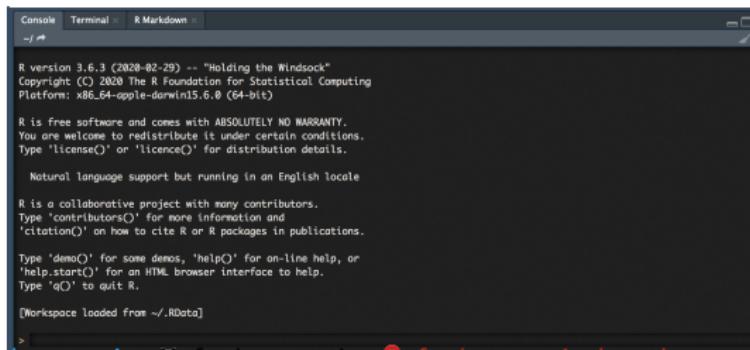
On peut y taper une succession de commandes (que l'on pourra exécuter ensuite)

```
1 # 5 observations
2 v = c(0.89367, -1.04729, 1.97133, -0.38363, 1.65414)
3 mean(v)
4 f = function(x) sum((v-x)^2)
5 optim(0, f)
```

# RStudio

La fenêtre **console** (bas à gauche) : après avoir tapé 'run',

```
1 > v = c(0.89367, -1.04729, 1.97133, -0.38363, 1.65414)
2 > mean(v)
3 [1] 0.617644
4 > f = function(x) sum((v-x)^2)
5 > optim(0, f)
6 $par
7 [1] 0.6175781
8 Warning message:
9 In optim(0, f) : one-dimensional optimization by
 Nelder-Mead is unreliable:
10 use "Brent" or optimize() directly
```



The screenshot shows the RStudio interface with the 'Console' tab selected. The window displays the standard R startup messages, including the version number (R version 3.6.3), copyright information, and instructions for redistribution. It also mentions natural language support and collaborative project details. At the bottom, it indicates that the workspace was loaded from a specific directory.

```
Console Terminal R Markdown
~/R

R version 3.6.3 (2020-02-29) -- "Holding the Windsock"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/RData]
```

La fenêtre **environment** (haut à droite)  
donne la liste des objets en mémoire

```
1 v num [1:5] 0.894 -1.047...
2 f function(x)
```

The screenshot shows the RStudio interface with the 'Environment' tab selected in the top navigation bar. The main pane displays a list of objects currently stored in memory, categorized under 'Data'. Each object is listed with its name, type, and a brief description of its contents. For example, 'A' is a '10 obs. of 2 variables' data frame containing numerical values. Other objects listed include 'acp', 'anova', 'AGV', 'apartments', 'AUTREIC', 'B', 'b2', 'base', 'BT', 'C', 'cah', 'cars', 'd', 'dot14boot', 'database', 'Davis', 'dc', and 'rid'. The 'Global Environment' dropdown menu is also visible at the top left.

Object	Type	Description
A	10 obs. of 2 variables	num [1:5, 1:2]
acp	List of 5	list
anova	List of 13	list
AGV	List of 13	list
apartments	1000 obs. of 6 variables	data frame
AUTREIC	num [1:45, 1:3] -62.7 807 907.8 -53.7 213...	vector
B	7 obs. of 2 variables	data frame
b2	20 obs. of 5 variables	data frame
base	891 obs. of 8 variables	data frame
BT	List of 2	list
C	num [1:3, 1:558] 0.1677 -0.0525 0.3725 0...	vector
cah	List of 7	list
cars	num [1:4, 1:4] 1 0.503 0.856 0.493 0.503 ...	vector
d	20 obs. of 2 variables	data frame
dot14boot	List of 11	list
database	6 obs. of 32 variables	data frame
Davis	200 obs. of 5 variables	data frame
dc	12 obs. of 12 variables	data frame
rid	50 obs. of 5 variables	data frame

# RStudio

## La fenêtre packages (bas à droite)

```
1 > install.packages("vcd")
```



[CRAN](#)  
[Mirrors](#)  
[What's new?](#)  
[Task Views](#)  
[Search](#)

[About R](#)  
[R Homepage](#)  
[The R Journal](#)

[Software](#)  
[R Sources](#)  
[R Binaries](#)  
[Packages](#)  
[Other](#)

[Documentation](#)  
[Manuals](#)  
[FAQs](#)  
[Contributed](#)

### Contributed Packages

#### Available Packages

Currently, the CRAN package repository features 17628 available packages.

[Table of available packages, sorted by date of publication](#)

[Table of available packages, sorted by name](#)

#### Installation of Packages

Please type `help("INSTALL")` or `help("install.packages")` in R for information on how to install packages from this repository. The manual [R Installation and Administration](#) (also contained in the R base sources) explains the process in detail.

[CRAN Task Views](#) allow you to browse packages by topic and provide tools to automatically install all packages for special areas of interest. Currently, 41 views are available.

#### Package Check Results

All packages are tested regularly on machines running [Debian GNU/Linux](#), [Fedora](#), macOS (formerly OS X), Solaris and Windows.

The results are summarized in the [check summary](#) (some [timings](#) are also available). Additional details for Windows checking and building can be found in the [Windows check summary](#).

Name	Description	Version
<b>System Library</b>		
abind	Combine Multidimensional Arrays	1.4-5
acepack	ACE and AVAS for Selecting Multiple Regression Transformations	1.4.1
AER	Applied Econometrics with R	1.2-9
ape	Analyses of Phylogenetics and Evolution	5.4
arm	Data Analysis Using Regression and Multilevel/Hierarchical Models	1.11-1
ash	David Scott's ASH Routines	1.0-15
askpass	Safe Password Entry for R, Git, and SSH	1.1
assertthat	Easy Pre and Post Assertions	0.2.1
backports	Reimplementations of Functions Introduced Since R 3.0.0	1.1.7
base64enc	Tools for base64 encoding	0.1-3
BenfordTests	Statistical Tests for Evaluating Conformity to Benford's Law	1.2.0
bit	Bit-Oriented Memory Efficient Tools	1.22.0

## Sample Quantiles

### Description

The generic function `quantile` produces sample quantiles corresponding to the given probabilities. The smallest observation corresponds to a probability of 0 and the largest to a probability of 1

### Usage

```
1 > quantile(x, probs = seq(0, 1, 0.25), na.rm = FALSE,
 names = TRUE, type = 7, ...)
```

### Arguments

`x`: numeric vector whose sample quantiles are wanted, or an object of a class for which a method has been defined (see also 'details'). NA and NaN values are not allowed in numeric vectors unless `na.rm` is TRUE

### References

Hyndman, R. J. and Fan, Y. (1996) Sample quantiles in statistical packages, *American Statistician* 50, 361–365. doi: 10.2307/2684934.

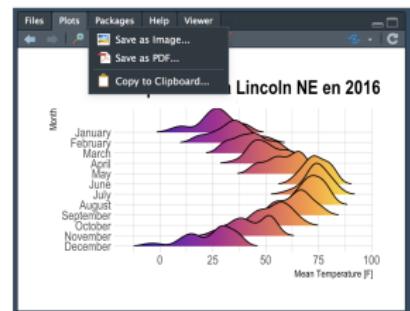
# RStudio: graphiques

Pour les graphiques (de base)

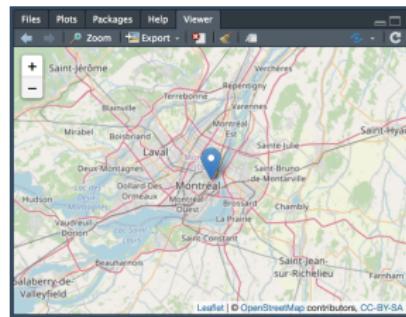
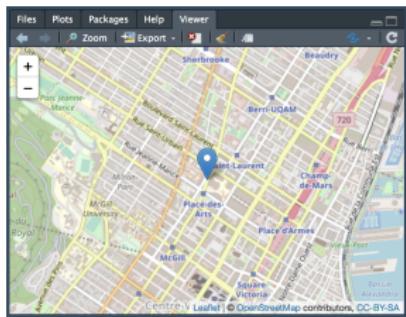
1. il faut créer la fenêtre graphique (avec la fonction `plot`)

```
1 > plot(cars)
```

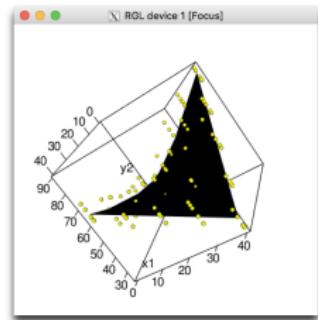
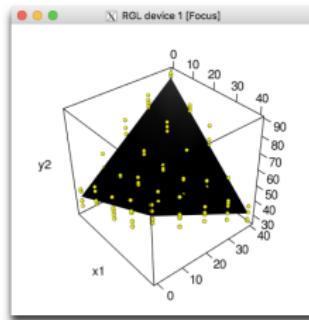
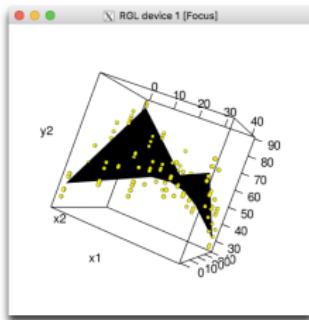
2. on peut alors superposer des courbes, des points, etc



Certains graphiques (dynamiques) apparaîtront dans la fenêtre viewer (e.g. leaflet)



ou dans une fenêtre extérieure (`rgl`)



# Fichier STT1000.RData

Les fichiers **.Rdata** contiennent des données (format lisible en R).

```
1 > url = "http://freakonometrics.free.fr/STT1000.RData"
2 > download.file(url,"STT1000.RData")
3 trying URL
4 Content type 'text/plain' length 459595 bytes (448 KB)
5 =====
6 downloaded 448 KB
7 > load("STT1000.RData")
```



## Fichier STT1000.RData

```
1 > ls()
2 [1] "a" "alcool" "aspirine"
3 [4] "attente" "b" "babies"
```

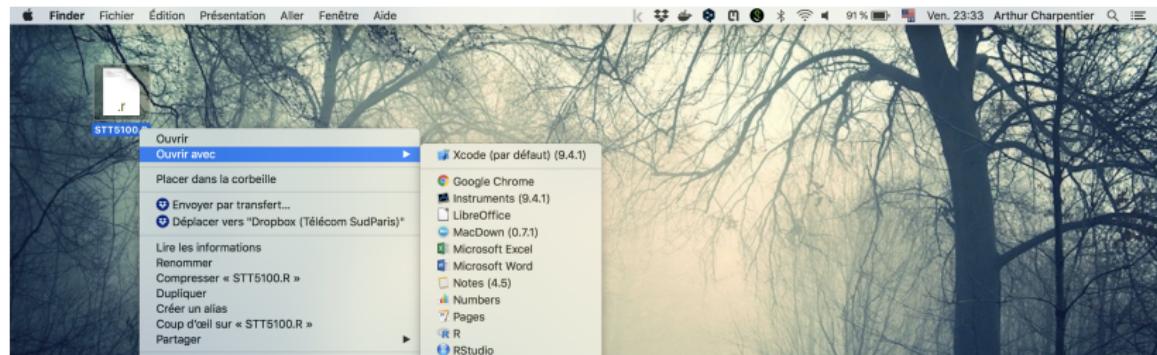
(etc)

```
1 > tail(Davis)
2 sex weight height reportedWeight reportedHeight
3 195 F 62 164 61 161
4 196 M 74 175 71 175
5 197 M 83 180 80 180
6 198 M 81 175 NA NA
7 199 M 90 181 91 178
8 200 M 79 177 81 178
9 > mean(Davis$height)
10 [1] 170.565
```

# Fichier STT1000.R

Les fichiers .R contiennent des codes R.

```
1 > url = "http://freakonometrics.free.fr/STT1000.R"
2 > download.file(url,"STT1000.R")
3 > source("STT1000.R")
```



# Base du langage R

```
1 > sum(Davis$sex=="M")
2 [1] 88
3 > mean(Davis$height [Davis$sex=="M"])
4 [1] 178.0114
5 > mean(Davis$height [Davis$sex=="F"])
6 [1] 164.7143
```

```
1 > aggregate(Davis$height , by=list(Davis$sex) , FUN=mean)
2 Group.1 x
3 1 F 164.7143
4 2 M 178.0114
```

```
1 > tapply(Davis$height , Davis$sex , mean)
2 F M
3 164.7143 178.0114
```

Pour un cours de R, parcourir le livre d'Ewen Gallic.

# Base du langage R

Par exemple, on peut programmer une descente de gradient,

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)} \text{ à partir d'une valeur initiale } x_0$$

```
1 > v = c(0.89367, -1.04729, 1.97133, -0.38363, 1.65414)
2 > f = function(x) sum((v-x)^2)
3 > df = function(x) -2*sum((v-x))
4 > d2f = function(x) 2*length(v)
5 > x = rep(2,100)
6 > for(i in 2:100) x[i]=x[i-1]-df(x[i-1])/d2f(x[i-1])
7 > x[100]
8 [1] 0.617644
```

ou

```
1 > df = function(x, h=1e-5) (f(x+h)-f(x))/h
2 > d2f = function(x, h=1e-5) (df(x+h)-df(x))/h
```

les fonctions, les boucles... un peu hors sujet ici, mais indispensable

## Et excel ?

A priori excel peut servir pour des statistiques descriptives, mais ce n'est pas ce qu'on va faire dans ce cours !

Avec un peu de courage, il doit être possible de refaire l'intégralité de ce que l'on verra dans un classeur excel...

Par exemple, on peut "programmer" une descente de gradient,

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)} \text{ à partir d'une valeur initiale } x_0$$

	E5					
1						
2						
3	0,89367	x0		df		df2
4	-1,04729	x1	0,6176	13,82356	-8,88178E-16	10
5	1,97133	x2	0,6176		0	10
6	-0,38363					
7	1,65414					

## Et excel ?

Mais les formules sont cachées dans les cellules, ce qui rend un classeur excel difficilement auditable

	A	B	C	D	E	F	G
1							
2					df		df2
3		0,89367	x0	2	2,21266	10	
4		-1,04729	x1	1,7787	5,652048	10	
5		1,97133	x2	1,2135	-1,5156016	10	
6		-0,38363	x3	1,3651	3,49743872	10	
7		1,65414	x4	1,0153	-1,277589024	10	
8			x5	1,1431	#VALEUR!		10

cf <https://www.wired.co.uk/article/spreadsheet-excel-errors>  
ou <https://www.bbc.com/news/magazine-22213219>

*“teaching Excel in actuarial undergraduate programs is important, but only in courses dedicated to operational risk”*