

# R + LATEX

Une très brève introduction

# **Objectifs et philosophie**

# Objectifs et philosophie

- ▶ Installation: RStudio +  $\mathbb{R}$  + L<sup>A</sup>T<sub>E</sub>X

# Objectifs et philosophie

- ▶ Installation: RStudio +  $\mathbb{R}$  +  $\text{\LaTeX}$
- ▶ Résumé de la journée:  $\mathbb{R}$  et  $\text{\LaTeX}$ 
  1. Le monde de l'*Open Source*
  2. Quelques bases de programmation  $\mathbb{R}$
  3. Faire des graphiques en  $\mathbb{R}$  avec  $\text{\LaTeX}$

# Objectifs et philosophie

- ▶ Installation: RStudio +  $\mathbb{R}$  +  $\text{\LaTeX}$
- ▶ Résumé de la journée:  $\mathbb{R}$  et  $\text{\LaTeX}$ 
  1. Le monde de l'*Open Source*
  2. Quelques bases de programmation  $\mathbb{R}$
  3. Faire des graphiques en  $\mathbb{R}$  avec  $\text{\LaTeX}$
- ▶ Philosophie: Les outils *avant* la méthode

# **Installation**

## **2 langages, 1 logiciel**



# **Installation**

## **2 langages, 1 logiciel**



1. R : <https://cran.rstudio.com>

# Installation

## 2 langages, 1 logiciel



1. R : <https://cran.rstudio.com>
2. L<sup>A</sup>T<sub>E</sub>X: [www.latex-project.org/get/](http://www.latex-project.org/get/)

# Installation

## 2 langages, 1 logiciel



1. R : <https://cran.rstudio.com>
2. L<sup>A</sup>T<sub>E</sub>X: [www.latex-project.org/get/](http://www.latex-project.org/get/)
3. RStudio: [www.rstudio.com/products/rstudio/download/](http://www.rstudio.com/products/rstudio/download/)

# Installation

## 2 langages, 1 logiciel

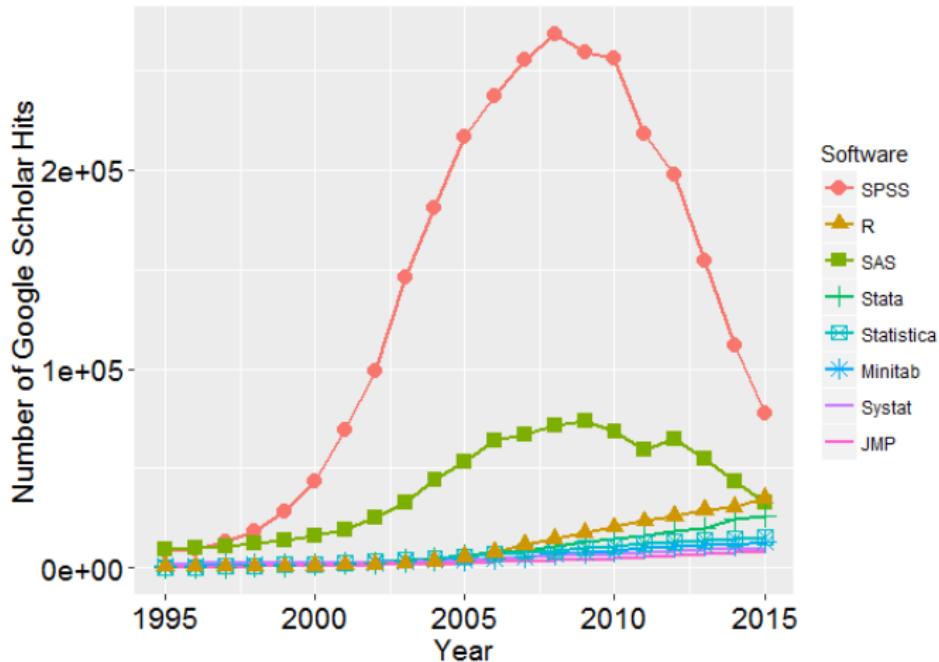
The screenshot shows the RStudio interface with the following details:

- File Explorer:** Shows multiple files: JFMorin.tex\*, Class3.tex, beamerthemeCement\_WorkshopR.sty, CreateMaps.R, Class2.tex, and mapWorld.
- Code Editor:** Displays R code for mapping. The code includes:
  - Setting the additive scale for colors.
  - Replacing country names in the DataRegion dataset.
  - Merging Data and mapWorld datasets.
  - Fixing missing code in mapWorld.
- Environment:** Shows "Environment is empty".
- Plots:** None.
- Packages:** None.
- Help:** None.
- Viewer:** None.
- Console:** Shows the command `1+1` and its result `[1] 2`.
- Status Bar:** Shows the path `/Dropbox (Personal)/ENSEIGNEMENT/Cours/POL7004-AnalyseQuantitative/POL7004-2017-Fall/`.

# Le monde de l'*Open Source*

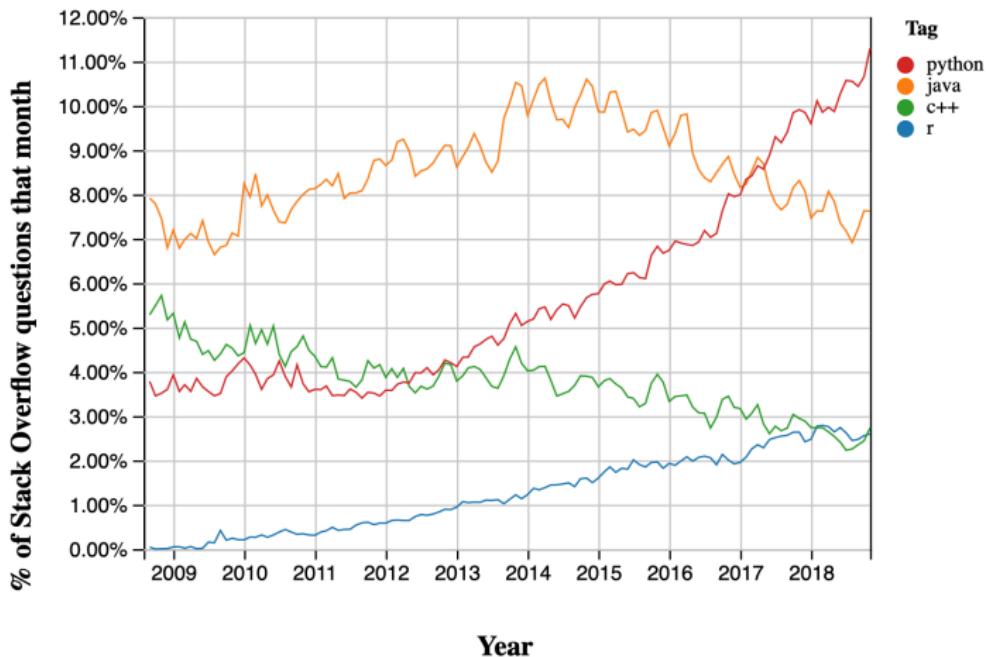
# Pourquoi R ?

# Pourquoi R ?



# Pourquoi R ?

## Stack Overflow Traffic to Programming Languages



# Pourquoi R ? Les raisons de l'aimer

1. Gratisssss
2. Disponible pour tous les systèmes d'exploitation
3. Graphiques + L<sup>A</sup>T<sub>E</sub>X
4. Popularité + *packages* (extensions)
5. *Open Source* (code source ouvert): développé par et pour les chercheurs

# Pourquoi R ? Les raisons de détester

1. Programmer du code = courbe d'apprentissage raide
2. Développement éclectique, parfois chaotique

# Pourquoi L<sup>A</sup>T<sub>E</sub>X?

# Pourquoi L<sup>A</sup>T<sub>E</sub>X? Raisons d'aimer

- ▶ Bibliographie: BibT<sub>E</sub>X
- ▶ Table des matières, tableaux, etc.
- ▶ S'occupe automatiquement des tableaux, graphiques, etc.
- ▶ De beaux gabarits
- ▶ Code + *Open Source* = une large communauté d'experts en ligne

# Pourquoi L<sup>A</sup>T<sub>E</sub>X? Raisons d'aimer

- ▶ Difficile à apprendre... Très difficile. Mais les bases sont simples
- ▶ Incompatible avec Word
- ▶ Pas d'autocorrecteur
- ▶ Pas de « suivi des corrections » et de trucs comme ça
- ▶ Le document final est uniquement disponible après la compilation du code
- ▶ Certaines revues scientifiques n'acceptent pas les soumissions en L<sup>A</sup>T<sub>E</sub>X... d'autres les encouragent

# L<sup>A</sup>T<sub>E</sub>X: Un beau tableau

Table 1. Length of Bananas and Apples

Quantile	Bananas	Apples
0%	59	44
50%	69	64
100%	77	71

# LATEX: Le code du beau tableau

```
\begin{table}
  \centering
  \caption{Length of Bananas and Apples}
  \begin{tabular}{lrr}
    Quantile & Bananas & Apples\\ \hline
    0\%      & 59      & 44 \\
    50\%     & 69      & 64 \\
    100\%    & 77      & 71 \\
  \end{tabular}
  \label{tab:bananasapples}
\end{table}
```

# LATEX

Tableau 1: Tests des hypothèses

	Vote pour le NPD						
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Évaluation du chef NPD					3.87*** (0.22)	3.81*** (0.24)	3.17**** (0.52)
Droite idéologique			-2.86*** (0.46)	-3.24*** (0.53)			-2.66*** (0.57)
Québec	0.69*** (0.09)	0.61*** (0.16)		0.92** (0.34)		0.56** (0.17)	0.93** (0.35)
Femme		0.05 (0.09)		-0.08 (0.19)		-0.03 (0.10)	-0.08 (0.20)
Francophone		-0.02 (0.17)		-0.37 (0.35)		-0.29 (0.18)	-0.63 (0.37)
allophone		-0.17 (0.15)		-0.38 (0.34)		-0.18 (0.17)	-0.22 (0.36)
Moins de 34 ans		-0.03 (0.15)		-0.17 (0.34)		-0.13 (0.16)	-0.26 (0.36)
Plus de 55 ans		-0.23* (0.10)		-0.33 (0.21)		-0.24* (0.11)	-0.23 (0.22)
Haut revenu		-0.33** (0.12)		-0.36 (0.24)		-0.30* (0.13)	-0.32 (0.25)
Faible revenu		0.30* (0.15)		0.33 (0.31)		0.40* (0.17)	0.49 (0.33)
Pas de diplôme secondaire		-0.23 (0.15)		0.04 (0.36)		-0.12 (0.17)	0.03 (0.38)
Diplôme universitaire		0.13 (0.10)		-0.61** (0.21)		-0.12 (0.11)	-0.79*** (0.22)
-constante	-1.05*** (0.05)	-0.86*** (0.11)	0.34 (0.20)	0.96** (0.35)	-3.17*** (0.15)	-2.95*** (0.19)	-1.21* (0.51)
N	2,745	2,464	655	610	2,636	2,381	602
Log Likelihood	-1,650.11	-1,487.30	-383.02	-346.16	-1,412.88	-1,276.31	-317.77
AIC	3,304.22	2,996.60	770.04	716.31	2,829.77	2,576.62	661.54

Source : Étude électorale canadienne, 2011.

Note : Régression logistique binomiale.

\*p<0.05 ; \*\*p<0.01 ; \*\*\*p<0.001

# L<sup>A</sup>T<sub>E</sub>X: Code (Partie 1)

<...>

```
1 % Table created by stargazer v.5.1 by Marek Hlavac, Harvard University. E-mail: hlavac
2 % Date and time: Wed, Jan 07, 2015 - 22:20:00
3 \begin{table}[]\centering
4 \caption{Tests des hypothèses}
5 \label{}
6 \scriptsize
7 \begin{tabular}{@{\extracolsep{5pt}}lcccccc}
8 \\[-1.8ex]\hline \\[-1.8ex]
9 \\[-1.8ex] & \multicolumn{7}{c}{Vote pour le NPD} \\
10 \\[-1.8ex] & (1) & (2) & (3) & (4) & (5) & (6) & (7)\\
11 \hline \\[-1.8ex]
12 Évaluation du chef NPD & & & & 3.87$^{***}$ & 3.81$^{***}$ & 3.17$^{***}$ \\
13 & & & & (0.22) & (0.24) & (0.52) \\
14 Droite idéologique & & & & -$2.86$^{***}$ & -$3.24$^{***}$ & & & -$2.66$^{***}$ \\
15 & & & & (0.46) & (0.53) & & & (0.57) \\
16 Québec & 0.69$^{***}$ & 0.61$^{***}$ & & 0.92$^{**}$ & 0.56$^{**}$ & 0.93$^{**}$ \\
17 & (0.09) & (0.16) & & (0.34) & & (0.17) & & (0.35) \\
18 Femme & 0.05 & & -$0.08 & & -$0.03 & & -$0.08 \\
19 & & (0.09) & & (0.19) & & (0.10) & & (0.20) \\
20 Francophone & & -$0.02 & & -$0.37 & & -$0.29 & & -$0.63 \\
21 & & (0.17) & & (0.35) & & (0.18) & & (0.37) \\
22 Allophone & & -$0.17 & & -$0.38 & & -$0.18 & & -$0.22 \\
23 & & (0.15) & & (0.34) & & (0.17) & & (0.36) \\
```

# LATEX: Code (Partie 2)

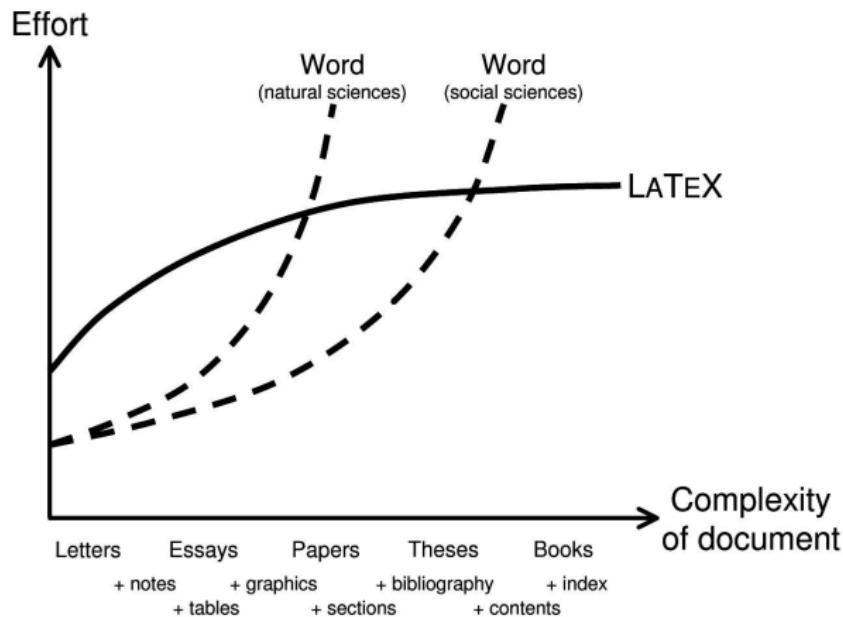
<...>

```
1 Moins de 34 ans & & $-$0.03 & & $-$0.17 & & $-$0.13 & $-$0.26 \\
2 & & (0.15) & & (0.34) & & (0.16) & (0.36) \\
3 Plus de 55 ans & & $-$0.23$^{*} \$ & & $-$0.33 & & $-$0.24$^{*} \$ & $-$0.23 \\
4 & & (0.10) & & (0.21) & & (0.11) & (0.22) \\
5 Haut revenu & & $-$0.33$^{**} \$ & & $-$0.36 & & $-$0.30$^{*} \$ & $-$0.32 \\
6 & & (0.12) & & (0.24) & & (0.13) & (0.25) \\
7 Faible revenu & & 0.30$^{*} \$ & & 0.33 & & 0.40$^{*} \$ & 0.49 \\
8 & & (0.15) & & (0.31) & & (0.17) & (0.33) \\
9 Pas de diplôme secondaire & & $-$0.23 & & 0.04 & & $-$0.12 & 0.03 \\
10 & & (0.15) & & (0.36) & & (0.17) & (0.38) \\
11 Diplôme universitaire & & 0.13 & & $-$0.61$^{**} \$ & & $-$0.12 & $-$0.79$^{***} \$ \\
12 & & (0.10) & & (0.21) & & (0.11) & (0.22) \\
13 \_constante & $-$1.05$^{***} \$ & $-$0.86$^{***} \$ & 0.34 & 0.96$^{**} \$ & $-$3.17$^{***} \$ & $-$1.412.88 \\
14 & (0.05) & (0.11) & (0.20) & (0.35) & (0.15) & (0.19) & (0.51) \\
15 N & 2,745 & 2,464 & 655 & 610 & 2,636 & 2,381 & 602 \\
16 Log-vraisemblance & $-$1,650.11 & $-$1,487.30 & $-$383.02 & $-$346.16 & $-$1,412.88 & $-$1,270.00 \\
17 AIC & 3,304.22 & 2,996.60 & 770.04 & 716.31 & 2,829.77 & 2,576.62 & 661.54 \\
18 \hline \\[-1.8ex] \\
19 \multicolumn{8}{l}{\emph{Source}: Étude électorale canadienne, 2011.} \\
20 \multicolumn{8}{l}{\emph{Note}: Régression logistique binomiale.} \\
21 \multicolumn{8}{l}{\$^{*} \$p\$ < \$0.05; \$^{**} \$p\$ < \$0.01; \$^{***} \$p\$ < \$0.001} \\
22 \end{tabular} \\
23 \end{table}
```

# L<sup>A</sup>T<sub>E</sub>X



# L<sup>A</sup>T<sub>E</sub>X



# R + LATEX

<...>

```
1 stargazer(model1, model2, model3, model4, model5, model6, model7)
```

# R + LATEX

Tableau 1: Tests des hypothèses

	Vote pour le NPD						
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Évaluation du chef NPD					3.87*** (0.22)	3.81*** (0.24)	3.17**** (0.52)
Droite idéologique		-2.86*** (0.46)	-3.24*** (0.53)				-2.66*** (0.57)
Québec	0.69*** (0.09)	0.61*** (0.16)		0.92** (0.34)		0.56** (0.17)	0.93** (0.35)
Femme		0.05 (0.09)		-0.08 (0.19)		-0.03 (0.10)	-0.08 (0.20)
Francophone		-0.02 (0.17)		-0.37 (0.35)		-0.29 (0.18)	-0.63 (0.37)
allophone		-0.17 (0.15)		-0.38 (0.34)		-0.18 (0.17)	-0.22 (0.36)
Moins de 34 ans		-0.03 (0.15)		-0.17 (0.34)		-0.13 (0.16)	-0.26 (0.36)
Plus de 55 ans		-0.23* (0.10)		-0.33 (0.21)		-0.24* (0.11)	-0.23 (0.22)
Haut revenu		-0.33** (0.12)		-0.36 (0.24)		-0.30* (0.13)	-0.32 (0.25)
Faible revenu		0.30* (0.15)		0.33 (0.31)		0.40* (0.17)	0.49 (0.33)
Pas de diplôme secondaire		-0.23 (0.15)		0.04 (0.36)		-0.12 (0.17)	0.03 (0.38)
Diplôme universitaire		0.13 (0.10)		-0.61** (0.21)		-0.12 (0.11)	-0.79*** (0.22)
-constante	-1.05*** (0.05)	-0.86*** (0.11)	0.34 (0.20)	0.96** (0.35)	-3.17*** (0.15)	-2.95*** (0.19)	-1.21* (0.51)
N	2,745	2,464	655	610	2,636	2,381	602
Log Likelihood	-1,650.11	-1,487.30	-383.02	-346.16	-1,412.88	-1,276.31	-317.77
AIC	3,304.22	2,996.60	770.04	716.31	2,829.77	2,576.62	661.54

Source : Étude électorale canadienne, 2011.

Note : Régression logistique binomiale.

\*p<0.05 ; \*\*p<0.01 ; \*\*\*p<0.001

# **LATEX: c'est tout?**

# L<sup>A</sup>T<sub>E</sub>X: c'est tout?

- ▶ PowerPoint: le Saint-Graal des présentations orales?

# L<sup>A</sup>T<sub>E</sub>X: c'est tout?

- ▶ PowerPoint: le Saint-Graal des présentations orales?
- ▶ NON! Beamer

# L<sup>A</sup>T<sub>E</sub>X: c'est tout?

- ▶ PowerPoint: le Saint-Graal des présentations orales?
- ▶ NON! Beamer
- ▶ Google Docs: l'Excalibur du travail en ligne instantané?

# L<sup>A</sup>T<sub>E</sub>X: c'est tout?

- ▶ PowerPoint: le Saint-Graal des présentations orales?
- ▶ NON! Beamer
- ▶ Google Docs: l'Excalibur du travail en ligne instantané?
- ▶ NON!!! Overleaf

# Overleaf

The screenshot shows the Overleaf interface with a LaTeX project titled "Exploration of empirical Bayes hierarchical modeling for the analysis of genome-wide association study data".

**Project Overview:** The left sidebar lists files: blo.cls, biorrefs.bst, colorsty, fig1.pdf, fig2.pdf, fig3.pdf, fig4.pdf, refs.bib, and samplebibtex.tex. There are buttons for "DOWNLOAD AS ZIP" and "Save to Dropbox".

**Document Content:**

**Title:** Exploration of empirical Bayes hierarchical modeling for the analysis of genome-wide association study data

**Authors:** ELIZABETH A. HERON\*, COLM O'DUSHLAINNE, RICARDO SEGURADO, LOUISE GALLAGHER, MICHAEL GILL

**Text:** [4pt]

**Text:** % Author addresses

**Text:** Neuropsychiatric Genetics Research Group and Department of Psychiatry, Trinity College Dublin, Trinity Centre for Health Sciences, James's Street, Dublin 8, Ireland

**Text:** [2pt]

**Text:** % E-mail address for correspondence  
eaheron@tcd.ie

**Footnote:** 33 % Add a footnote for the corresponding author if one has been  
34 % Identified in the author list  
35 |footnotetext|To whom correspondence should be addressed.

**Abstract:**

**Text:** 38 |In the analysis of genome-wide association (GWA) data, the aim is  
39 to detect statistical associations between single nucleotide  
40 polymorphisms (SNPs) and the disease or trait of interest. These  
41 SNPs, or the particular regions of the genome they implicate, are  
42 then considered for further study. We demonstrate through a  
43 comprehensive simulation study that the inclusion of additional,  
44 biologically relevant information through a 2-level

**Biostatistics** tab is selected, showing the journal's information and submission details.

# R Les bases de la programmation

# R = langage de programmation

- ▶ Opérateurs de calcul
- ▶ Opérateurs d'assignement
- ▶ Opérateurs logiques
- ▶ Instructions de contrôle

# Opérateurs de calcul

- +
- -
- \*
- /
- %%

# Opérateurs logiques

- ==
- !=
- >=
- <=
- <
- >
- &
- |
- %in%

# Instructions de contrôle

- ▶ if... else
- ▶ for loop

# R Structure des données

# Structure des données

- ▶ Constantes

# Structure des données

- ▶ Constantes
- ▶ Vecteurs

# Structure des données

- ▶ Constantes
- ▶ Vecteurs
- ▶ *Data frames* (tableaux de données)

# Constantes

<...>

```
1 variableString <- "Banana"  
2 variableNumerical <- 1492  
3 variableBoolean <- TRUE
```

# Vecteurs

<...>

```
1 vecteurString <- c(variableString, "Apple", "Orange", "Sand Paper")
2 vecteurNumerical <- c(variableNumerical, 1604, 2011, 0328424)
3 vecteurBoolean <- c(variableBoolean, FALSE, TRUE, TRUE)
```

# *Data frames*

<...>

```
1 Data <- data.frame(vectorString, vectorNumerical, vectorBoolean, c(23,17,32,56))
```

Constantes

Vecteurs

Data frames

Constantes

Vecteurs

Data frames

1 

Constantes

1 

Vecteurs

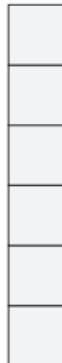


Data frames

## Constantes

1 

## Vecteurs

1	
2	
3	
4	
5	
6	

## Data frames

	v1	v2	v3	v4
1				
2				
3				
4				
5				
6				

# aFruit <- “banana”

Constantes



Vecteurs

1	
2	
3	
4	
5	
6	

Data frames

	v1	v2	v3	v4
1				
2				
3				
4				
5				
6				

# **fruits[1] <- “banana”**

Constantes



Vecteurs

1	A yellow banana icon with a brown stem, positioned at the top of the vector.
2	
3	
4	
5	
6	

Data frames

	v1	v2	v3	v4
1				
2				
3				
4				
5				
6				

# Data[1,1] <- “banana”

Constantes



Vecteurs

1	A yellow banana icon with a brown stem, positioned at the top of the first column.
2	
3	
4	
5	
6	

Data frames

	v1	v2	v3	v4
1	A yellow banana icon with a brown stem, positioned at the top-left cell of the matrix.			
2				
3				
4				
5				
6				

# Data\$fruits[1] <- “banana”

Constantes



Vecteurs

1	
2	
3	
4	
5	
6	

Data frames

	v1	v2	v3	v4
1				
2				
3				
4				
5				
6				

# Fonctions

# Fonctions de base R

- ▶ length()
- ▶ min()
- ▶ max()
- ▶ sum()
- ▶ median()
- ▶ mean()

# Fonction R : mean()

<...>

```
1 mean(yourVector)
```

# Créer une fonction en R

# Fonction R : meanGirls()



# Fonction R : meanGirls()

<...>

```
1 meanGirls <- function(Data){  
2     result <- sum(Data$age[Data$woman==1])/length(Data$age[Data$woman==1])  
3     return(result)  
4 }
```

# Fonction R : meanGirlsPlus()



```
1 meanGirlsPlus <- function(Data, star=FALSE){  
2     if(star == FALSE){  
3         result <- sum(Data$age[Data$woman==1])/length(Data$age[Data$woman==1])  
4     } else {  
5         result <- sum(Data$age[Data$woman==1])/length(Data$age[Data$woman==1])  
6         result <- paste("*****", result, "*****")  
7     }  
8     return(result)  
9 }
```

# Maintenant?

# Maintenant? Plus de fonctions R ...

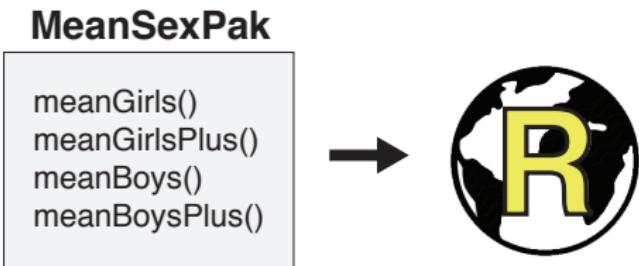
```
meanGirls()  
meanGirlsPlus()  
meanBoys()  
meanBoysPlus()
```

# Maintenant? Un *package* R

## MeanSexPak

```
meanGirls()  
meanGirlsPlus()  
meanBoys()  
meanBoysPlus()
```

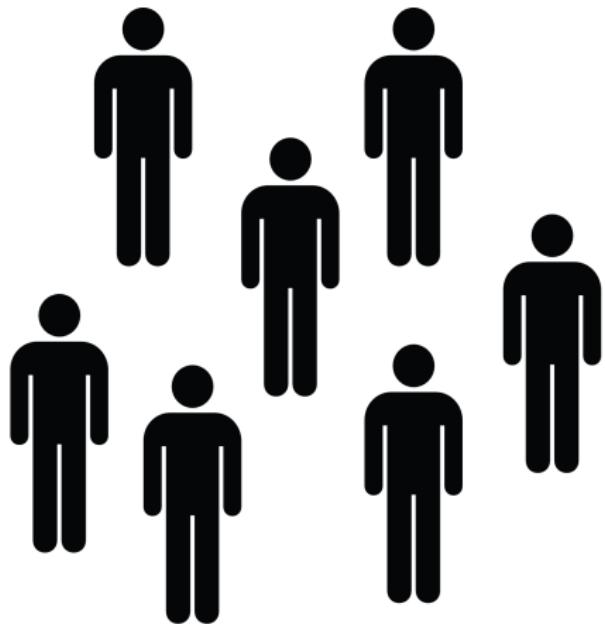
# Maintenant? La publication d'un *package*



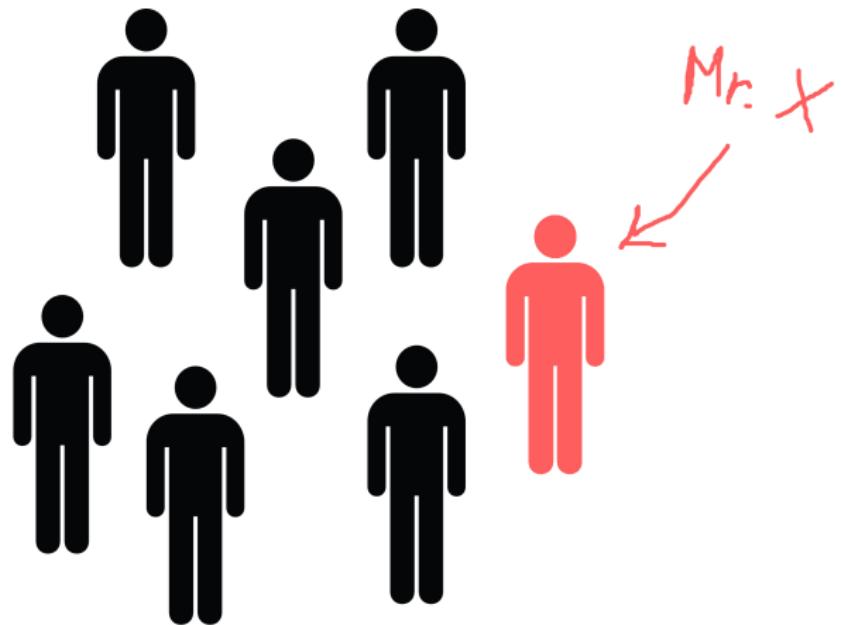
# Maintenant? Diffusion à la communauté



# Communauté $\mathbb{R}$



# M. X



# Ensuite? M. X installe le *package*

<...>

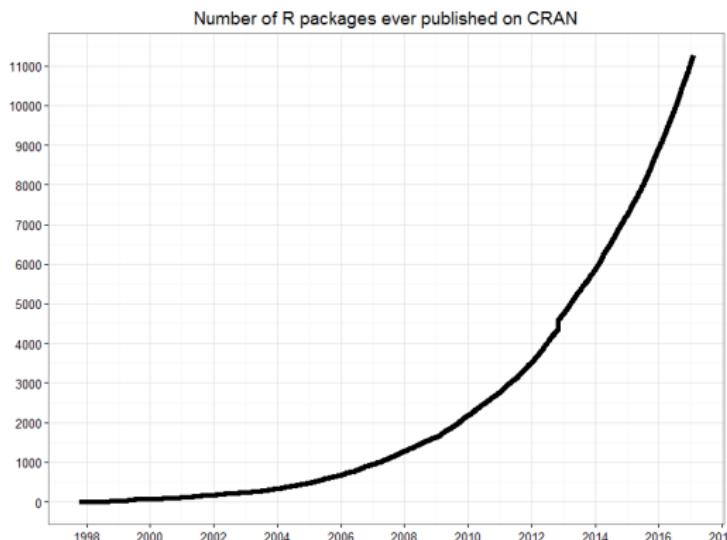
```
1 |install.packages("MeanSexPak")
```

# Ensuite? M. X utilise le *package*

<...>

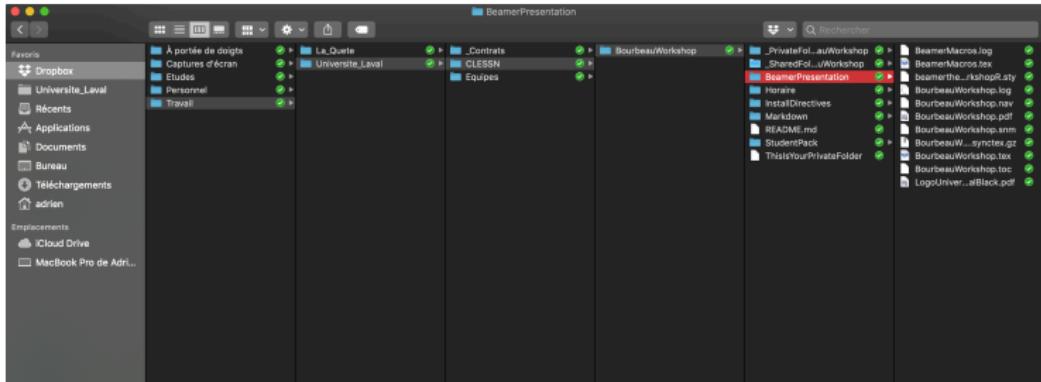
```
1 library(MeanSexPak)
2
3 # Calculate the mean age of the girls
4 girlsMeanAge <- meanGirls(MrXOwnData)
```

# Nombre de *packages R*



**Assez de blabla... C'est  
le temps de coder!**

# La logique de Dropbox



Assez de blabla... C'est le temps de coder!

# PLUS JAMAIS



mémoire.xml



mémoire.v2.xml



mémoire.final.xml



mémoire.dernier  
final.xml

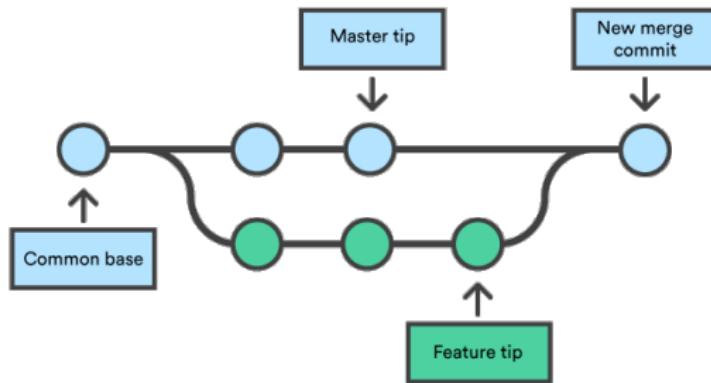


mémoire.dernier  
final.sûr.xml



mémoire.final  
putaindemerde.final.xml

# Version control



Assez de blabla... C'est le temps de coder!

# Git = un logiciel libre

```
BeamerPresentation — adrien@MBP-de-Adrien-2 — ..rPresentation — -zsh
→ BourbeauWorkshop git:(master) ✘ ls
BeamerPresentation           Markdown
Horaire                     README.md
InstallDirectives          StudentPack
ThisIsYourPrivateFolder
_PrivateFolder_BourbeauWorkshop
_SharedFolder_BourbeauWorkshop
→ BourbeauWorkshop git:(master) ✘ cd BeamerPresentation
→ BeamerPresentation git:(master) ✘ ls
BeamerMacros.log            BourbeauWorkshop.pdf
BeamerMacros.tex            BourbeauWorkshop.snm
BourbeauWorkshop.log         BourbeauWorkshop.synctex.gz
BourbeauWorkshop.nav         BourbeauWorkshop.tex
BourbeauWorkshop.toc
LogoUniversiteLavalBlack.pdf
beamertHEMEcement_WorkshopR.sty
→ BeamerPresentation git:(master) ✘ git pull
Already up to date.
→ BeamerPresentation git:(master) ✘ git add .
→ BeamerPresentation git:(master) ✘ git commit -a -m "autres diapo sur Git et Github"
[master aec4e24] autres diapo sur Git et Github
 3 files changed, 8 insertions(+)
→ BeamerPresentation git:(master) ✘ git push
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 306.82 KiB | 1.70 MiB/s, done.
Total 6 (delta 5), reused 0 (delta 0)
remote: Resolving deltas: 100% (5/5), completed with 5 local objects.
To https://github.com/clesn/BourbeauWorkshop.git
 c243e47..aec4e24  master -> master
→ BeamerPresentation git:(master) ✘ █
```

Assez de blabla... C'est le temps de coder!

# GitHub = un site web

The screenshot shows a GitHub repository page for 'clessn / BourbeauWorkshop'. The top navigation bar includes 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the header, there are tabs for 'Code', 'Issues', 'Pull requests', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. A message states 'No description, website, or topics provided.' with an 'Edit' button. The repository statistics are: 38 commits, 2 branches, 0 releases, and 7 contributors. A dropdown menu shows 'Branch: master'. Buttons for 'Create new file', 'Upload files', 'Find file', and 'Clone or download' are visible. A list of recent commits is shown:

Commit	Message	Time Ago
AdriCloud ajout de ma diapo sur la logique de dropbox	Latest commit ce27e4d 35 minutes ago	
BeamerPresentation	ajout de ma diapo sur la logique de dropbox	35 minutes ago
Horaire	Horaire final	5 hours ago
InstallDirectives	Suite de la traduction du Beamer et ajout des macros.tex	3 days ago
Markdown	Suppression de fichiers non-nécessaires	2 days ago
StudentPack	Ajout des fichiers originaux créés sur mon ordi	9 days ago
.DS_Store	ajout de ma diapo sur l'arborescence de Dropbox	42 minutes ago
.gitignore	Add LaTeX stuff to the .gitignore	8 days ago
README.md	Initial commit	9 days ago

Assez de blabla... C'est le temps de coder!

# Quelques bases

## Common Git Commands



- `$git config`
- `$git init`
- `$git clone <path>`
- `$git add <file_name>`
- `$git commit`
- `$git status`
- `$git remote`
- `$git checkout <branch_name>`
- `$git branch`
- `$git push`
- `$git pull`
- `$git merge <branch_name>`
- `$git diff`
- `$git reset`
- `$git revert`
- `$git tag`
- `$git log`

# La vraie base

In case of fire



1. git commit



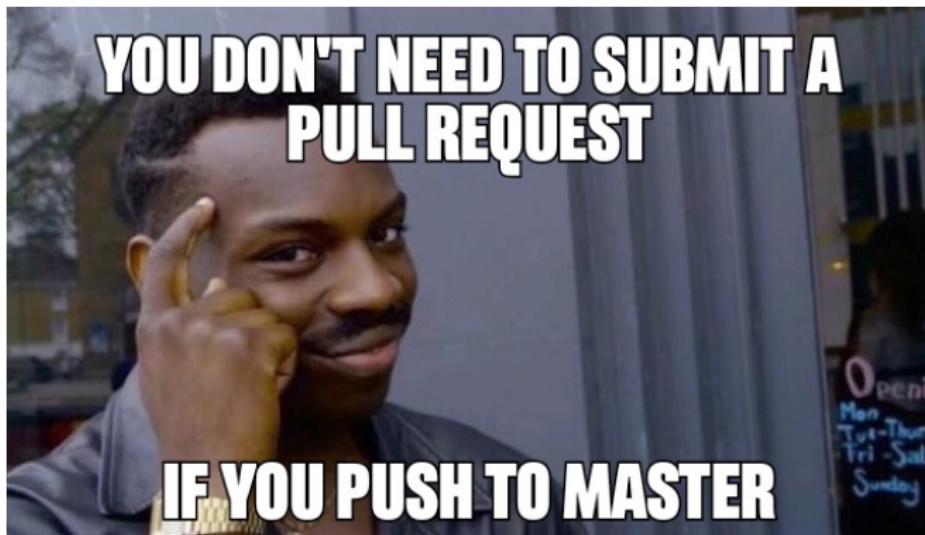
2. git push



3. leave building

Programmers protocol

# N'empêche, il faut réfléchir..!



# Vim et Lady Gaga



I Am Developer

@iamdevloper



Always enjoy seeing someone  
trying to exit Vim for the first time.

Lady Gaga ✅ @ladygaga

AAAAAAAAAAAAAHHHHRHRGRGRGRRGUR

BHJB

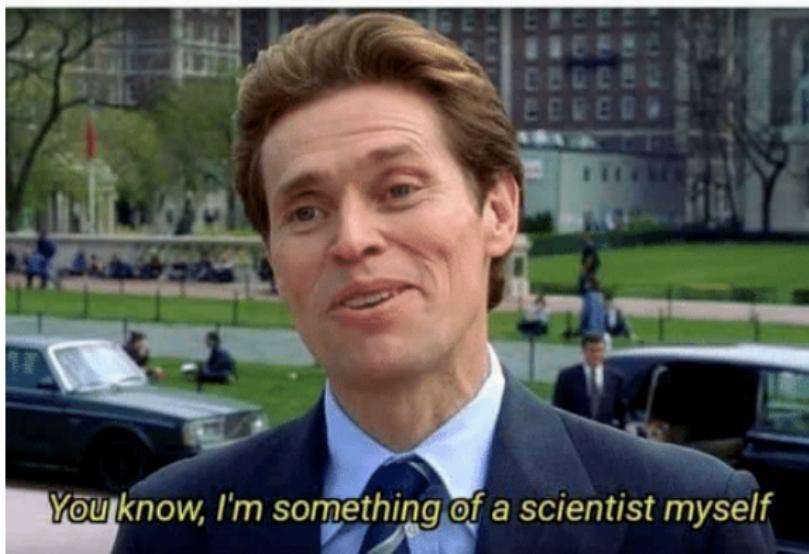
EORWPSOJWPJORGWOIRGWSGODEWPGOHE

PW09GJEDPOKSD!!!!!!!!!!!!!!

0924QU8T63095JRGHWPE09UJ0PWHRGW

# Vim = un éditeur de texte

When you finally exit vim



Finally did it!!

# VIM Adventures = un jeu pour apprendre Vim!



Assez de blabla... C'est le temps de coder!

# Wordpress et HTML

# **Plan de la présentation**

# Plan de la présentation

- Wordpress
  - 1. Astra
  - 2. Elementor

# Plan de la présentation

- ▶ Wordpress
  - 1. Astra
  - 2. Elementor
- ▶ HTML
  - 1. La création d'Internet
  - 2. Notions de base en programmation HTML

# Plan de la présentation

- ▶ Wordpress
  - 1. Astra
  - 2. Elementor
- ▶ HTML
  - 1. La création d'Internet
  - 2. Notions de base en programmation HTML
- ▶ Démonstration

# **Wordpress : Astra & Elementor**

# Wordpress : Astra & Elementor

- Wordpress c'est quoi?
  - Système de gestion de contenus (SGC ou CMS)
  - Images, vidéos, PDF
  - Pages, boutons, hyperliens

# Wordpress : Astra & Elementor

- ▶ Wordpress c'est quoi?
  - ▶ Système de gestion de contenus (SGC ou CMS)
  - ▶ Images, vidéos, PDF
  - ▶ Pages, boutons, hyperliens
- ▶ Pas idéal à lui seul
  - ▶ D'où l'utilisation de thèmes

# **Wordpress: Astra**

# Wordpress: Astra

- ▶ Astra c'est quoi?
  - ▶ Un thème construit pour Wordpress
  - ▶ Une collection de *templates* et de *stylesheets*
  - ▶ Offre également des sites web à personnaliser

# Wordpress: Astra

- ▶ Astra c'est quoi?
  - ▶ Un thème construit pour Wordpress
  - ▶ Une collection de *templates* et de *stylesheets*
  - ▶ Offre également des sites web à personnaliser
- ▶ Pourquoi Astra?
  - 1 Facilité d'utilisation: inclut Elementor
  - 2 Produit des sites web très rapides
  - 3 Présence de tutoriels complets en ligne

# **Wordpress: Elementor**

# Wordpress: Elementor

- Elementor c'est quoi?
  - ▶ Constructeur de pages ou *page builder*
  - ▶ Existance propre, mais intégré à Astra
  - ▶ Offre une interface facile d'utilisation

# **La création d'Internet**

# La création d'Internet

- ▶ Internet
  - ▶ Entre 1975 et 1980
  - ▶ Réseau de réseaux

# La création d'Internet

- ▶ Internet
  - ▶ Entre 1975 et 1980
  - ▶ Réseau de réseaux
- ▶ World Wide Web : 1989
  - ▶ Pas la même chose qu'Internet!
  - ▶ Système utilisé pour accéder à Internet
  - ▶ On accède au WWW par des navigateurs web:
    - ▶ Chrome, Safari, Firefox, Opera, Internet Explorer, etc.
  - ▶ Existence d'autres systèmes comme:
    - ▶ E-mail, messagerie instantanée

# **WWW: URL, HTML et HTTP?**

# WWW: URL, HTML et HTTP?

- ▶ URL = Uniform Resource Locator
  - ▶ Adresse du site web

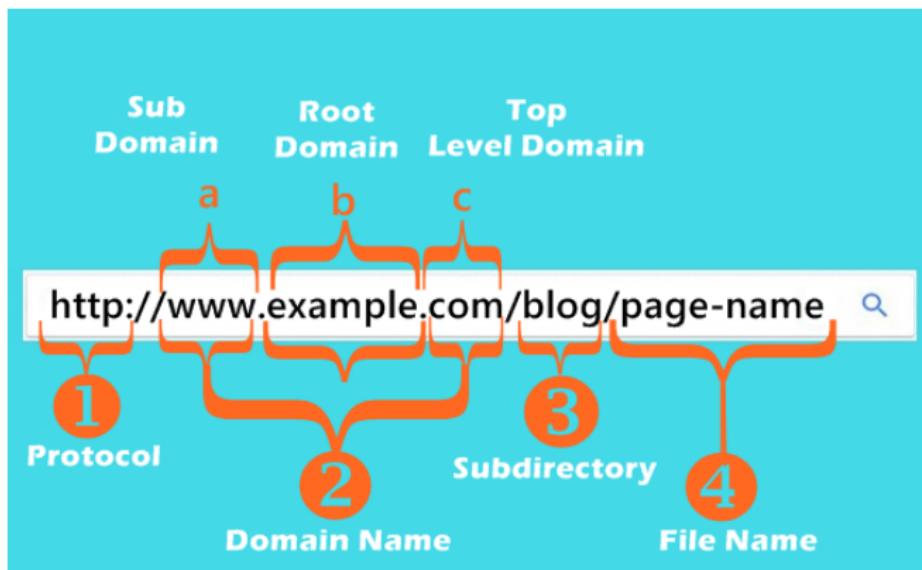
# WWW: URL, HTML et HTTP?

- ▶ URL = Uniform Resource Locator
  - ▶ Adresse du site web
- ▶ HTTP = Hypertext Transfer Protocol
  - ▶ Transmettre et formater les commandes
  - ▶ Indiquer aux serveurs et aux navigateurs comment agir
  - ▶ HTTPS = sa version sécurisée

# WWW: URL, HTML et HTTP?

- ▶ URL = Uniform Resource Locator
  - ▶ Adresse du site web
- ▶ HTTP = Hypertext Transfer Protocol
  - ▶ Transmettre et formater les commandes
  - ▶ Indiquer aux serveurs et aux navigateurs comment agir
  - ▶ HTTPS = sa version sécurisée
- ▶ HTML = Hypertext Markup Language
  - ▶ Le *markup language* standard
  - ▶ Indique comment formater et afficher la **page** web

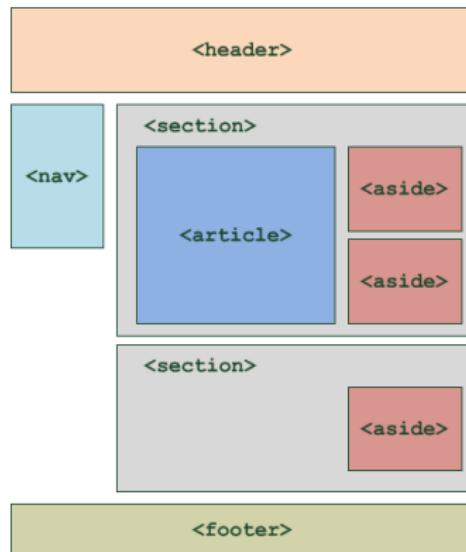
# Explorons une URL



# **HTML, quelques notions de base**

# HTML, quelques notions de base

- La structure d'une page HTML



# **HTML, quelques notions de base**

# HTML, quelques notions de base

- ▶ La structure du langage HTML
  - ▶ À la façon de LaTeX
  - ▶ "<body> </body>" = ouvre et ferme le document
  - ▶ "<p> </p>" = ouvre et ferme un paragraphe
  - ▶ "<h1-6> </h1-6>" = un entête
  - ▶ "<ul> </ul>" = une liste non ordonnée
  - ▶ "<em> </em>" = une section en *italique*
  - ▶ "<strong> </strong>" = une section en **gras**

# Démonstration

**Possibilités pour  
pousser plus loin  
nos recherches**

# Avec R, penser autrement les possibilités de recherche

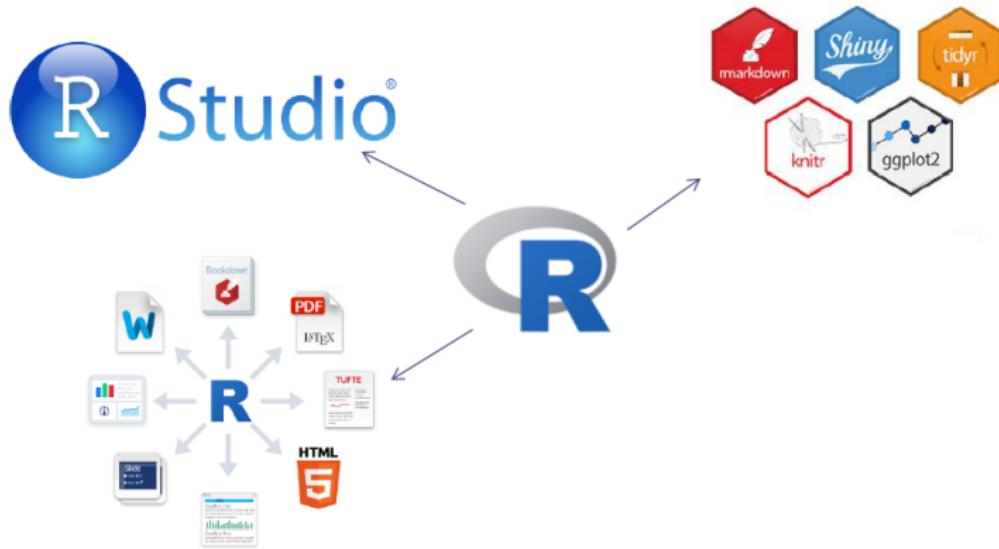
# Plan de la présentation

- ▶ Des outils qui s'offrent à nous
  - ▶ Shiny
  - ▶ MTurk
- ▶ Utiliser R dans la systématisation des revues de littérature
  - ▶ *Scoping review*

# Shiny: Qu'est-ce qu'une *Shiny app*?

- ▶ Environnement d'application web pour R
- ▶ Permet de transformer des analyses en applications web réactives
- ▶ Ne nécessite aucune connaissance de HTML, CSS, ou JavaScript
- ▶ Partageable avec des gens qui n'ont pas R

# Shiny: dans un processus commun et synergique



# Shiny: un exemple très simple avec ggplot2

```
library(shiny)

# Define server logic required to draw a histogram
server <- function(input, output) {

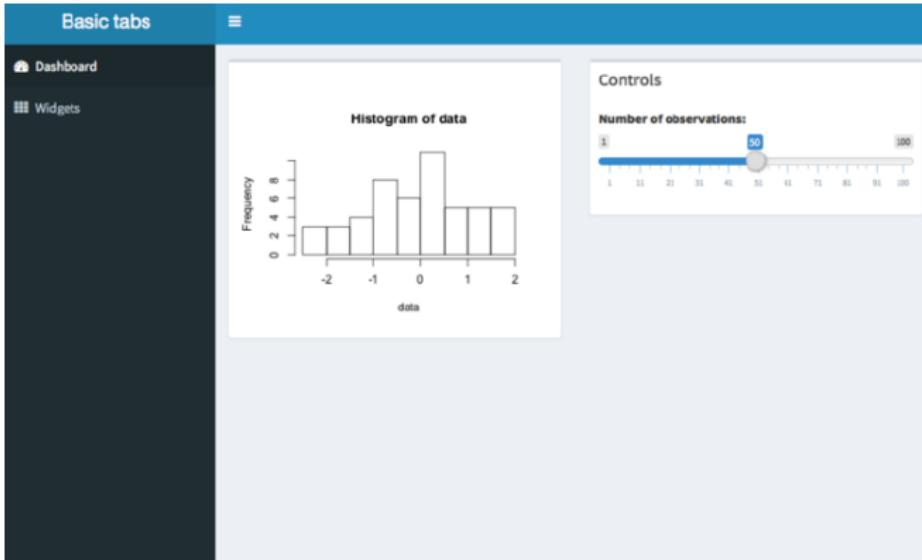
  output$plot_try <- renderPlot(
    ggplot(filter(mtcars, cyl %in% input$NbCylinder), aes(x = mpg, y = disp)) +
      geom_point(alpha=0.8, fill = "lightgray", width = 4) +
      theme_classic() +
      ggtitle(input>Title, paste0("Vous avez choisi ", input$NbCylinder, " cylindres")) +
      scale_x_continuous(name="Nombre de cylindres") +
      scale_y_continuous(name="Blabla")

  )
}
```

# Shiny: un exemple très simple avec ggplot2

```
19 # Define UI for application that draws a histogram
20
21 dashboardPage( # pour construire la page
22   dashboardHeader(title = "Test",
23     dropdownMenu(type = "messages", badgeStatus = "success",
24       messageItem("Support Team",
25         "This is the content of a message.",
26         time = "5 mins"
27       ),
28       messageItem("Support Team",
29         "This is the content of another message.",
30         time = "2 hours"
31     ),
32     messageItem("New User",
33       "Can I get some help?",
34       time = "Today"
35   )),
36   #haut de la fenetre
37   dashboardSidebar(),
38   dashboardBody(textInput(inputId ="Title", label = "Il faut ajouter le titre du graphique"),
39     selectInput(inputId = "NbCylinder", label = "Quel est le nombre de cylindres?",
40       choice = unique(mtcars$cyl)),
41     plotOutput("plot_try", width = "80%"))
42 )
```

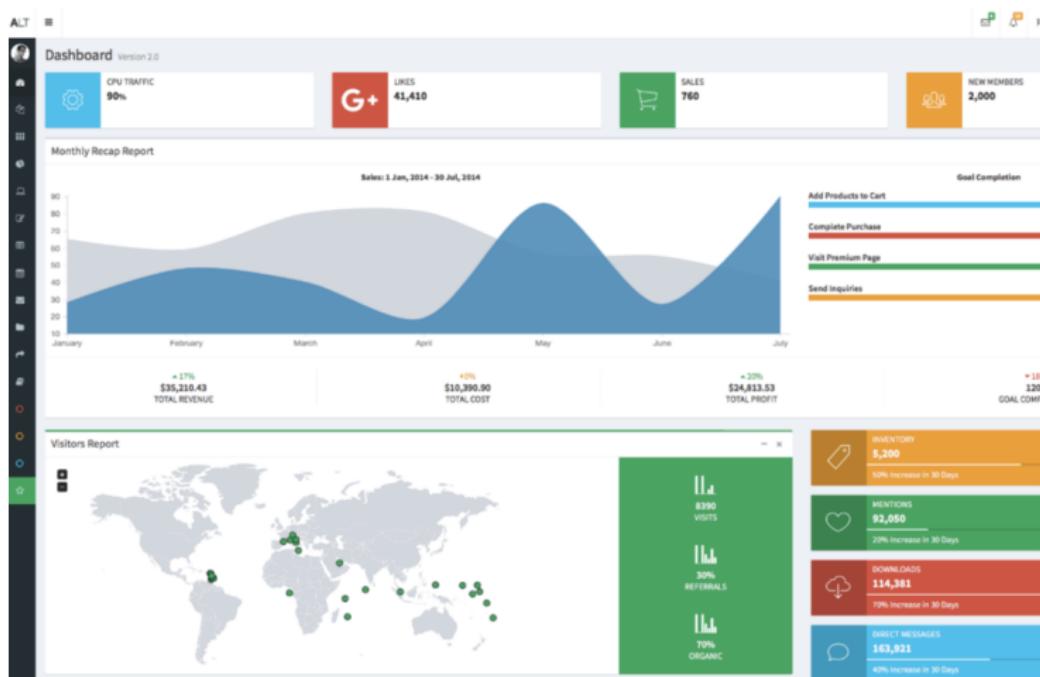
# Shiny: un exemple très simple avec ggplot2



# Shiny: un exemple très simple avec ggplot2

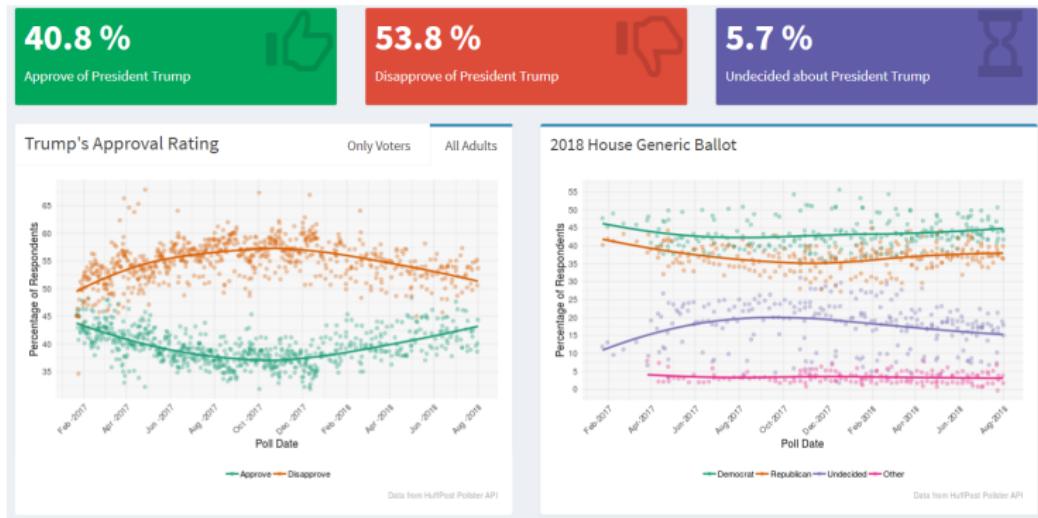


# Shiny: des exemples plus raffinés

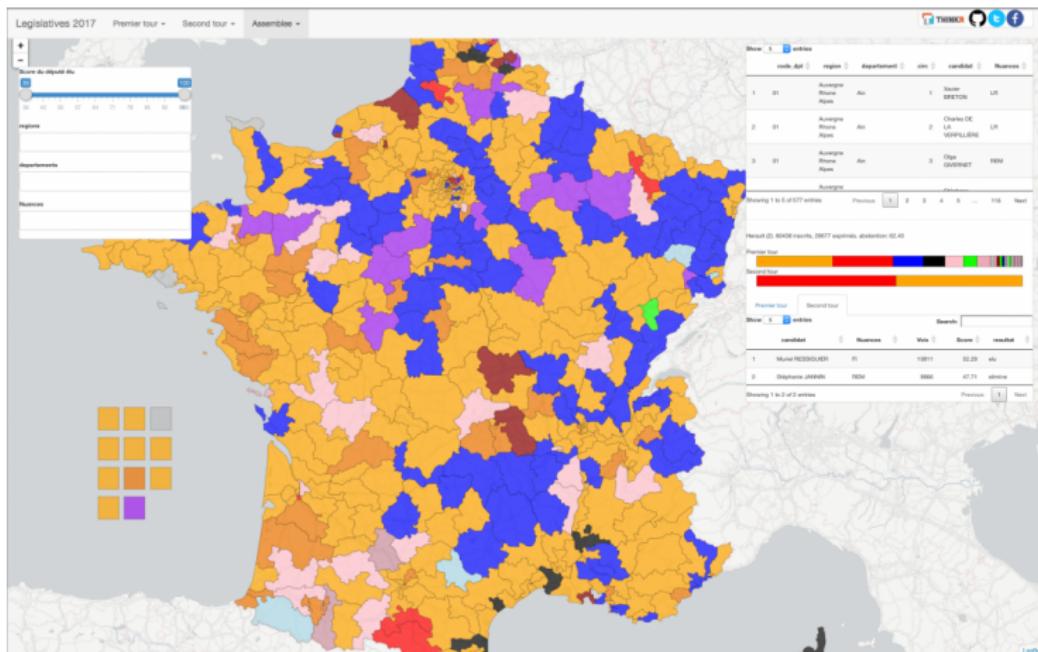


Possibilités pour pousser plus loin nos recherches

# Shiny: des exemples plus raffinés



# Shiny: des exemples plus raffinés



Possibilités pour pousser plus loin nos recherches

# **Une infinité de possibilités: un *package* continuellement en développement**

# Cent têtes valent mieux qu'une: le *crowdsourcing* en sciences sociales



# Qu'est-ce que le *crowdsourcing*?

Branche de l'innovation ouverte et de la co-création

C'est un acte consistant, pour une entreprise ou une institution, à assumer une fonction autrefois exercée par des employés et à l'externaliser vers un réseau indéfini (et généralement étendu) de personnes, sous la forme d'un appel ouvert. Cela peut prendre la forme d'une production par des pairs (lorsque le travail est effectué en collaboration), mais le travail est également souvent entrepris par des individus isolés (novices ou experts).

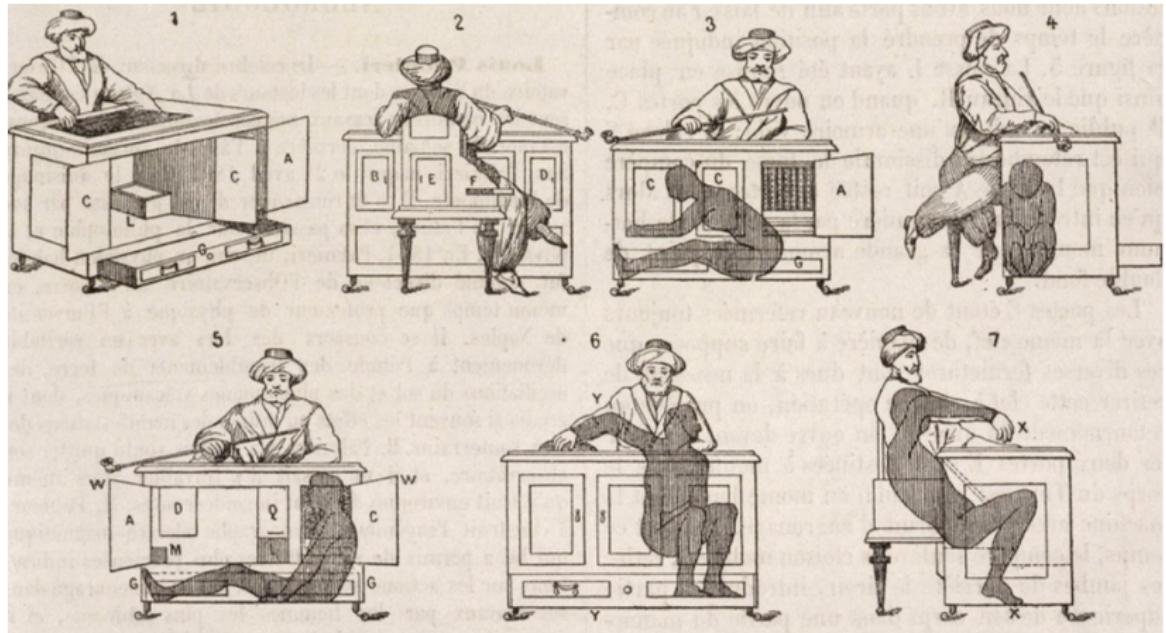
(Howe, 2016)

5051 citations

# Qu'est-ce que le *crowdsourcing*?

- Analyse de contenu;
- Sondage;
- Codage
- Collecte de données;
- Résolution d'opération mathématique;
- Génération de nouvelles idées;
- Programmation;
- Vérification;
- Reconnaissance optique de caractères;
- Correction;
- Relecture;
- Traduction;
- Micro-recherche;
- Évaluation;
- Opinions;
- Création;
- Design.

# La plateforme MTurk



# Utilisation dans la littérature

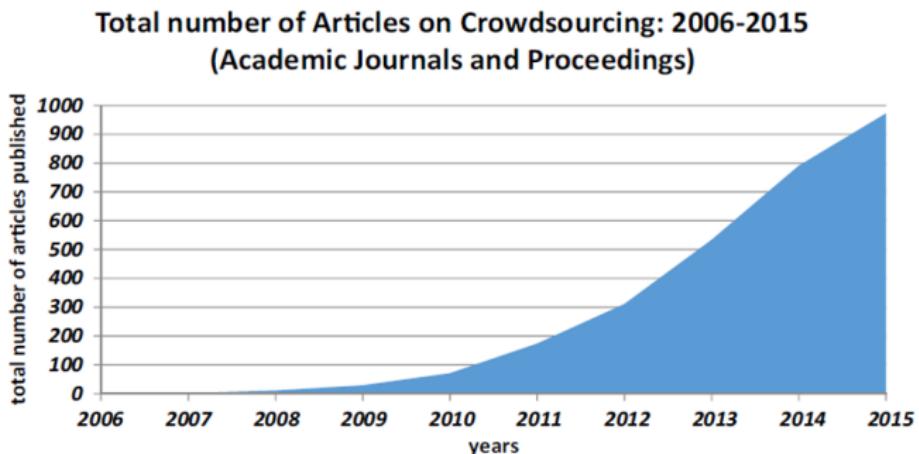
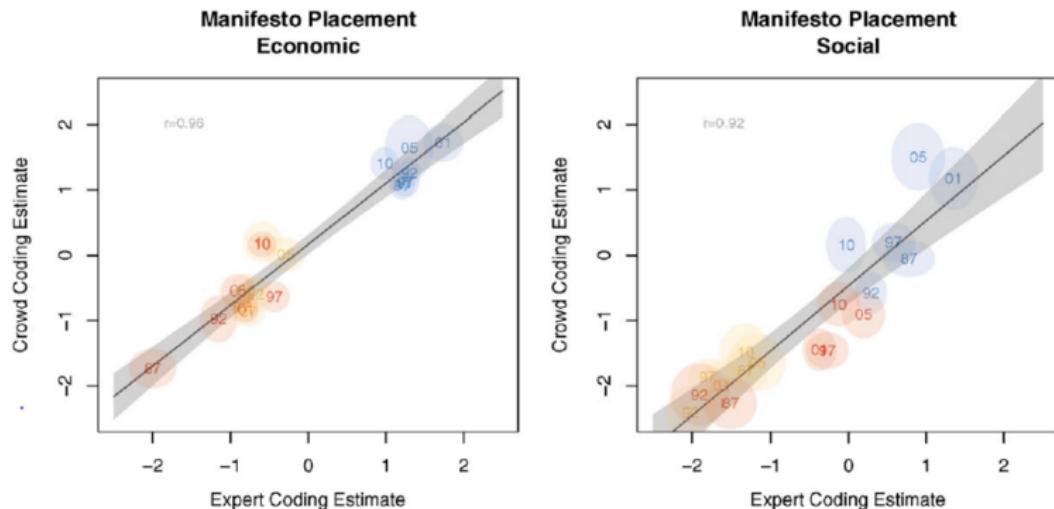


Figure 1. Academic articles and proceedings on 'crowdsourcing' in Scopus – Social Sciences & Humanities – Cumulative Source: SciVerse Scopus database (accessed 1 October 2015)

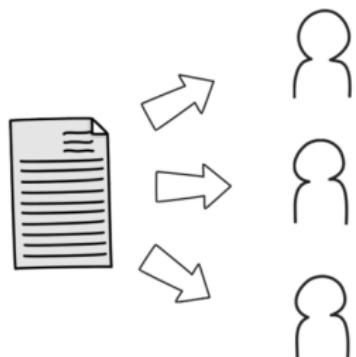
# Utilisation dans la littérature

FIGURE 3. Expert and Crowd-sourced Estimates of Economic and Social Policy Positions



# Un peu de vocabulaire

- ▶ HIT: Human Intelligence Tasks
- ▶ Assignment
- ▶ Turker
- ▶ Requester



# À savoir pour plonger dans MTurk

Paiement par assignment:

- ▶ Moyenne de 0,01 à 0,05 \$
- ▶ Recommandé ~ salaire minimum
- ▶ Frais de traitement de 20 %



# À savoir pour plonger dans MTurk

Qualifications des Turkers:

- ▶ Qualifications entraînent des frais supplémentaires
- ▶ Possibilité de sélectionner parmi plus de 60 spécificités
- ▶ Trop, c'est comme pas assez
- ▶ Tester, tester, tester



# À savoir pour plonger dans MTurk

Design HTML proposé, mais possibilité d'utiliser des plateformes externes

- ▶ Qualtrics
- ▶ Survey Monkey
- ▶ Google Forms
- ▶ Sites institutionnels



# Un exemple de la maison

amazon mturk Requester      Create      Manage      Developer

Results      Workers      Qualification Types

Manage Batches > Batch Details

A new way to summarize findings from systematic reviews 5

View the latest status of this batch, make changes, or get results.

---

You will be asked to read a summary of the systematic review of studies on programs. Then you will have to complete a survey, as your participation scientific communication.

Status
<p>Status: Pending Review</p> <div style="width: 100%; background-color: #00ff00; height: 10px; margin-bottom: 5px;"></div> <div style="display: flex; justify-content: space-between;"><span>100% submitted</span><span>100% published</span></div>

Assignments Completed: 81 / 81      Average Time per Assignment: 10 minutes 38 seconds  
Creation Time: April 16, 2019 1:29 PM PDT      Completion Time: April 16, 2019 2:29 PM PDT

# Un exemple de la maison

**Settings**

A new way to summarize findings from systematic reviews 5

[View Project](#)

Note: If you have edited the Project after publishing this Batch, you will see the latest version.

**Description:** You will be asked to read a summary of the systematic review of studies on programs. Then you will have to complete a survey, as your participation will help improve this new tool for scientific communication.

**Keywords:** survey, scientific communication  
HIT Approval Rate (%) for all Requesters' HITs greater than 80

**Qualification Requirement(s):** US Bachelor's Degree equal to true

---

**Number of Assignments per task:** 81

**Reward per Assignment:** \$2.00

---

**Batch expired on:** April 23, 2019 1:29 PM PDT

**Assignment duration:** 20 minutes

**Auto Approval Delay:** 4 days

# Un exemple de la maison

Results	
Assignments pending review:	0
Assignments approved:	81
Assignments rejected:	0

Cost Summary	
Estimated Total Reward:	\$162.00
Estimated Fees to Mechanical Turk:	\$105.30 ( <a href="#">fee details</a> )
Estimated Total Cost:	\$267.30
These costs are only an estimate until all of the assignments have been submitted and reviewed.	

# Un exemple de la maison

## Fee Details



The per assignment fees collected by Mechanical Turk for your batch are \$1.30

Mechanical Turk Fee      \$0.80

US Bachelor's Degree Fee    \$0.50

---

Fee per assignment      \$1.30

The fees collected by Mechanical Turk for your 81 assignment batch are \$105.30

To learn more about fees, please see our information on [pricing](#).

# Un exemple de la maison

Additional Fee per assignment for using

Premium Qualifications

([How do I use Premium Qualifications?](#))

Blogger: \$0.25

Born 1918 to 1960 (Age 55 or older): \$0.50

Born 1961 to 1971 (Age 45-55): \$0.50

Born 1972 to 1981 (Age 35-45): \$0.50

Born 1982 to 1986 (Age 30-35): \$0.50

Born 1987 to 1991 (Age 25-30): \$0.50

Born 1992 to 1999 (Age 18-25): \$0.50

Borrower - Auto Loans: \$0.40

Borrower - Business Loan: \$0.40

Borrower - Credit Cards: \$0.40

Borrower - Home Mortgage: \$0.40

Borrower - Personal Loan: \$0.40

Borrower - Student Loan: \$0.40

Car Owner: \$0.25

Current Residence - Owned: \$0.40

Current Residence - Rented: \$0.40

Daily Internet Usage - 1 to 4 hours: \$0.30

Daily Internet Usage - 5 to 7 hours: \$0.30

Daily Internet Usage - 7+ hours: \$0.30

Employment Industry - Banking & Financial Services: \$0.40

Employment Industry - Education: \$0.40

Employment Industry - Food & Beverage: \$0.40

Employment Industry - Government & Non-Profit: \$0.40

Employment Industry - Healthcare: \$0.40

Employment Industry - Manufacturing: \$0.40

Employment Industry - Media & Entertainment: \$0.40

Employment Industry - Retail, Wholesale & Distribution: \$0.40

Employment Industry - Software & IT Services: \$0.40

Employment Sector - Non-Profit: \$0.30

Employment Status - Full time (35+ hours per week): \$0.35

Employment Status - Part time (1-34 hours per week): \$0.35

Employment Status - Unemployed: \$0.35

Exercise - Every Day: \$0.30

Exercise - Four Plus Times a Week: \$0.30

# Un exemple de la maison

Example task from this Batch

**Survey Link Instructions** (Click to expand)

**Survey link:** [https://rc1user3cyjd9l7rtzhq.ca1.qualtrics.com/jfe/form/SV\\_0AkRMVkkwaVjzHn](https://rc1user3cyjd9l7rtzhq.ca1.qualtrics.com/jfe/form/SV_0AkRMVkkwaVjzHn)

Provide the survey code here:

e.g. 123456

You must ACCEPT the HIT before you can submit the results.

# MTurk et R

- ▶ Automatiser l'approbation des tâches
- ▶ Insérer des tâches dont on sait le résultat
- ▶ Insérer une randomisation des catégories

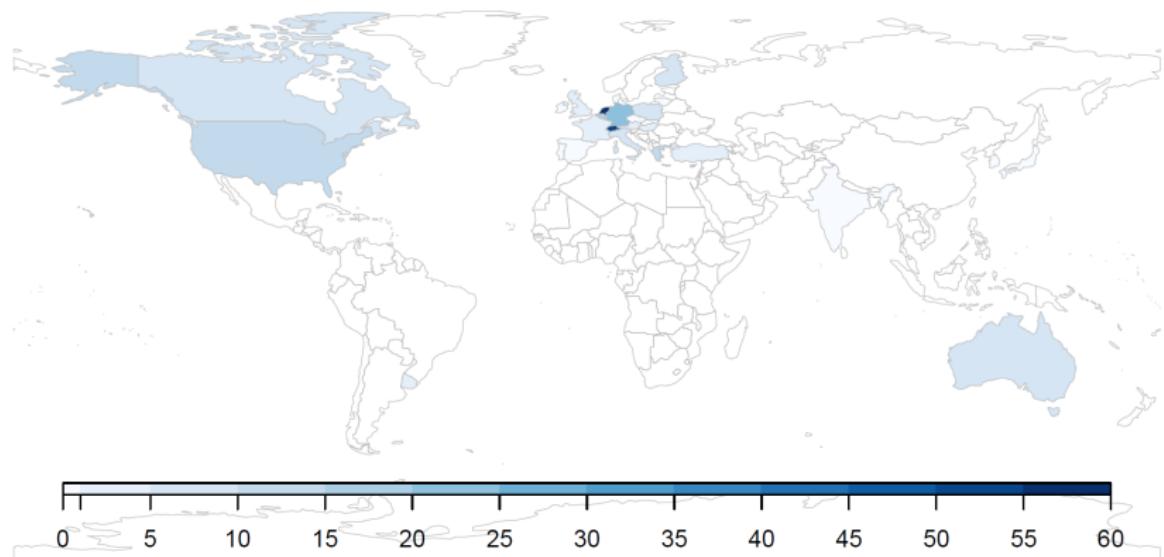
# ***Scoping review: cartographier la littérature scientifique***

Approche systématique et transparente pour appréhender la littérature

- ▶ Élaboration et déploiement d'une stratégie de recherche documentaire
- ▶ Collecte d'un large corpus de références académiques
- ▶ Tri et codage des références
- ▶ Analyses des caractéristiques de la littérature dans le champ de recherche

# *Scoping review: et R alors?*

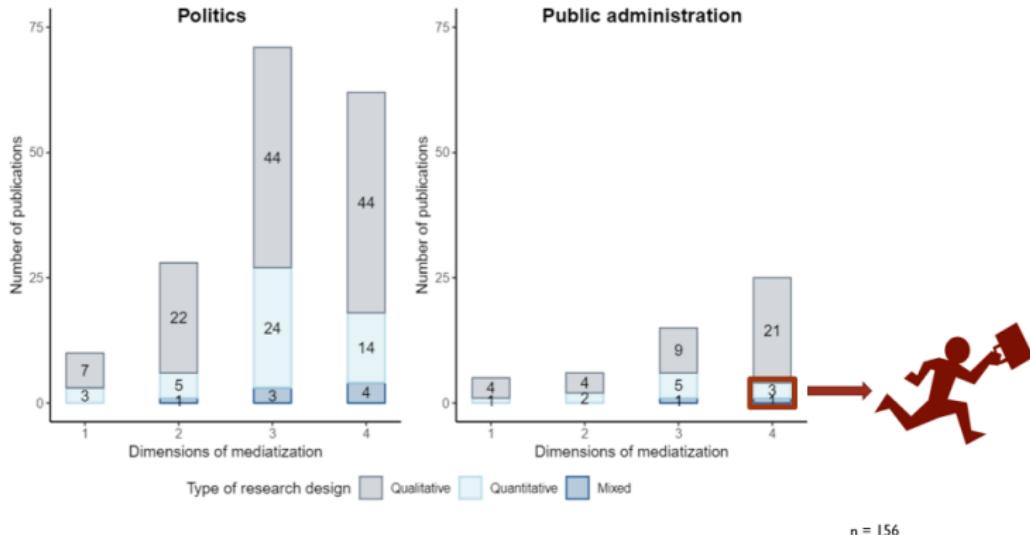
Cartographie de la littérature



Possibilités pour pousser plus loin nos recherches

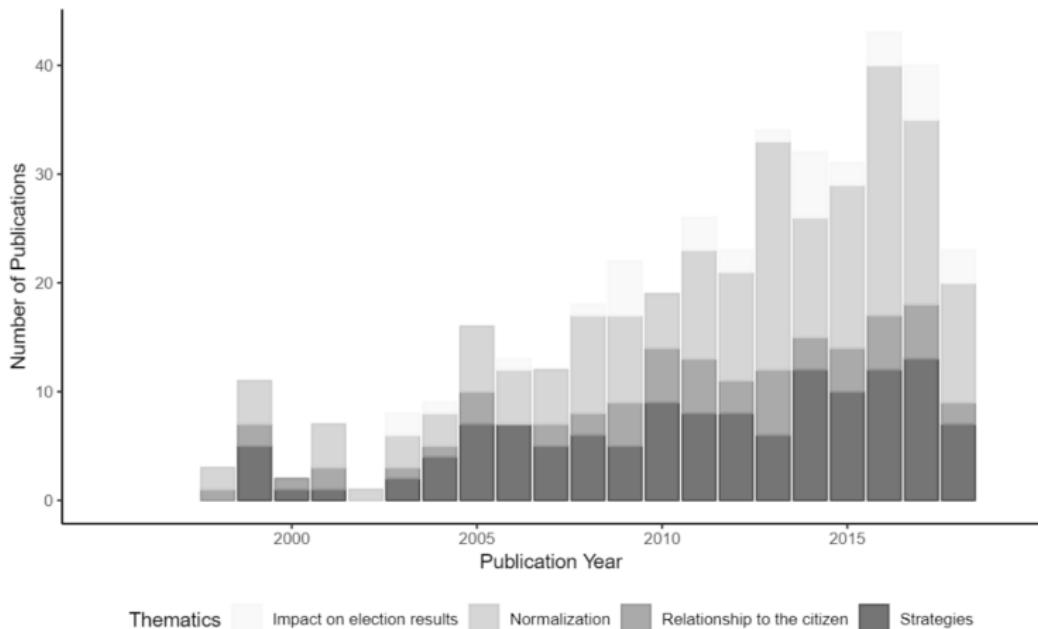
# *Scoping review: et R alors?*

Visualiser la distribution de certaines caractéristiques



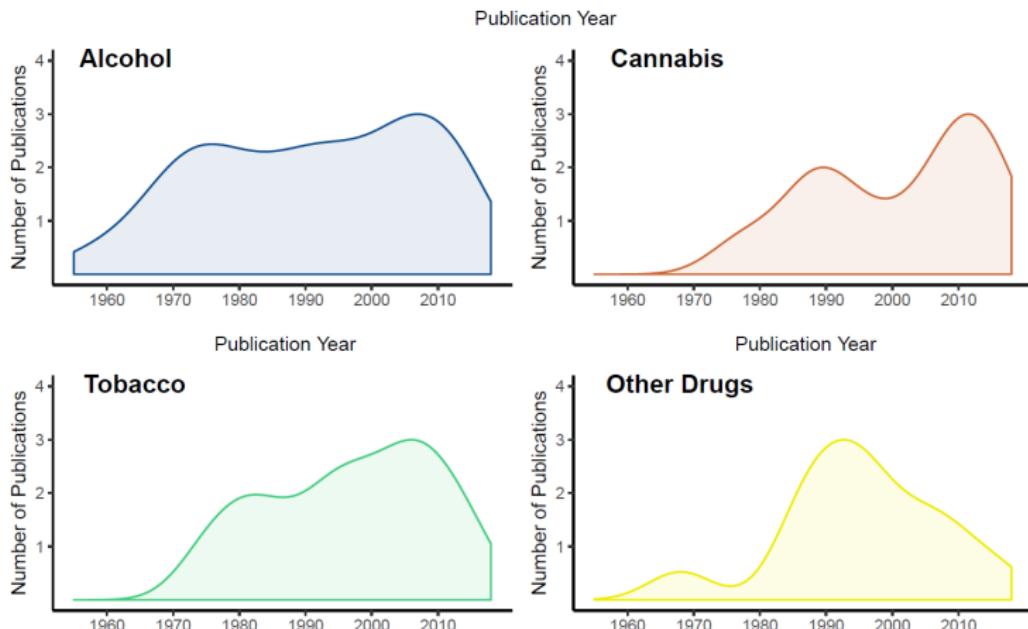
# *Scoping review: et R alors?*

Visualiser la distribution de certaines caractéristiques



# *Scoping review: et R alors?*

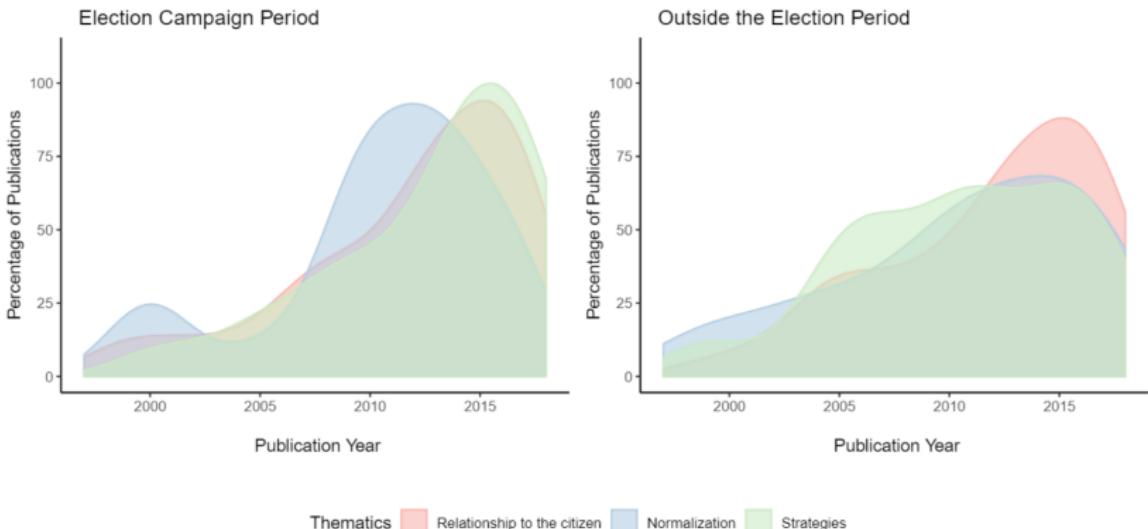
Visualiser la distribution de certaines caractéristiques



Possibilités pour pousser plus loin nos recherches

# *Scoping review: et R alors?*

Visualiser la distribution de certaines caractéristiques



# ***Scoping review: « D'accord, mais on pourrait faire ça avec Excel... »***

Oui! Mais pourquoi c'est mieux avec R :

- ▶ Coder dans l'optique de pouvoir utiliser le code pour plusieurs projets
- ▶ Mise à jour automatique des visualisations graphiques lorsque les données changent
- ▶ Possibilités de visualisations graphiques



**Et maintenant,  
comment on apprend?**

# **Contre vents et marées: apprendre R et savoir naviguer malgré les intempéries**

Et maintenant, comment on apprend?

# Plan de la présentation

- ▶ Apprentissage
  - ▶ Présentation de DataCamp
  - ▶ Choix de cours: serpents et échelles
  - ▶ DataCamp vs les autres sites
  - ▶ Livres, manuels et autres
  - ▶ Opportunités d'apprentissage
- ▶ Navigage
  - ▶ Stack Overflow
  - ▶ Slack
  - ▶ Conseils pour régler ses problèmes de programmation

# Présentation de DataCamp



DataCamp, c'est un site Internet où apprendre R, Python, Git, SQL, etc. Mais c'est beaucoup plus que cela!

- ▶ +200 instructeurs provenant de plusieurs disciplines
- ▶ +250 cours offerts, allant de débutant à avancé
- ▶ Exercices théoriques et pratiques, accompagnés de vidéos
- ▶ Une équipe dynamique qui développe constamment du nouveau contenu

# Apprentissage



Et maintenant, comment on apprend?

# Présentation de DataCamp



Plusieurs instructeurs connus, notamment Hadley Wickham,  
scientifique en chef à RStudio

- ▶ PhD en statistiques, Iowa State University
- ▶ ggplot2, plyr, dplyr, stringr, tidyverse
- ▶ L'approche tidy

# Choix de cours: serpents et échelles



Qu'est-ce que le *serpents et échelles*?

- Une liste précise et personnalisable de matériel académique pour apprendre ℝ
- Une liste des pièges à éviter pour atteindre ses objectifs
- Une façon dynamique de voir son progrès

# Choix de cours: serpents et échelles



Catégories/ Difficulté	Importation et manipulation des données	Visualisation de données	Probabilités et analyses statistiques
Débutant	Introduction to R; Importing Data in R (part 1); Cleaning Data in R; Introduction to the Tidyverse;	Data visualization in R; Data visualization with ggplot2 (part 1);	Introduction to Data; Exploratory Data Analysis; Foundation of Probability in R; Foundation of Inference;
Intermédiaire	Intermediate R; Intermediate R practice; Introduction to Text Analysis in R; Importing Data in R (part 2);	Data visualization with ggplot2 (part 2); Visualization Best Practices in R;	Correlation and regression; Multiple and Logistic Regression; Exploratory Data Analysis;
Avancé	Writing efficient R code; Importing & Cleaning Data in R: Case studies; Working with Web Data in R;	Data Visualization with ggplot2 (part 3); Communicating with Data in the Tidyverse;	Forecasting Using R; Statistical Modeling in R (part 1); Exploratory Data Analysis in R: Case Study;

Échelles pour l'apprentissage de R

# Choix de cours: serpents et échelles



Des serpents existent aux différents niveaux d'expertise

## ► Débutant

- ▶ Croire qu'il sera trop difficile d'apprendre, que c'est un objectif inatteignable
- ▶ Croire qu'il est possible d'apprendre sans pratiquer
- ▶ La peur de demander de l'aide
- ▶ Ne pas construire des bases solides avant d'aller plus loin
- ▶ La boucle infinie de tutoriels

# Choix de cours: serpents et échelles



Des serpents existent aux différents niveaux d'expertise

## ► Intermédiaire

- ▶ Croire qu'on a suffisamment de connaissances et ne pas sortir de sa zone de confort
- ▶ Vouloir apprendre plusieurs langages et n'en maîtriser aucun (R vs Python, Ruby, PHP...)
- ▶ Écrire du code mais ne pas le commenter
- ▶ Coder en n'utilisant pas un style et une planification cohérente et constante

# Choix de cours: serpents et échelles



Des serpents existent aux différents niveaux d'expertise

► Avancé

- ▶ La peur de partager son code
- ▶ Laisser le parfait être l'ennemi du bien
- ▶ Manquer d'empathie et de compréhension envers les nouveaux utilisateurs
- ▶ Douchebagisme

# DataCamp vs les autres sites

- ▶ Plus grande quantité de cours, toujours grandissante
- ▶ Instructeurs, souvent du milieu académique, reconnus et certifiés
- ▶ Prix concurrentiels et forfaits académiques disponibles
- ▶ Apprendre à son propre rythme, vs Coursera et cie.

# Autres ressources pertinentes

## ► Livres

- ▶ Statistiques en sciences humaines avec R (Guay, 2014)
- ▶ Learning R. A Step-by-Step Function Guide to Data Analysis (Cotton, 2013)
- ▶ Hands-On Programming with R (Grolemund, 2014)
- ▶ Advanced R, 2nd ed. (Wickham, 2019)

# Autres ressources pertinentes

- ▶ Ressources en ligne
  - ▶ R Bootcamp
  - ▶ Quick-R
  - ▶ R-bloggers

# Opportunités d'apprentissage

- ▶ R à Québec
- ▶ Utiliser  $\mathbb{R}$  et  $\text{\LaTeX}$  dans ses différents projets,  
professionnels et personnels

# Navigage



Et maintenant, comment on apprend?

# Stack Overflow



- ▶ Grand nombre d'usagers spécialisés dans de nombreux domaines
- ▶ Multitude de questions et réponses
- ▶ Communauté sympathique et dynamique

# Slack



- ▶ Outils de communication avec différentes chaînes, publiques et privées
- ▶ Possibilité d'y joindre d'autres applications telles que Teamline, GitHub, etc.
- ▶ Slack *La Fabrique*: Slack du département en science politique de l'UL et de ses collaborateurs
- ▶ Possibilité de demander et d'offrir de l'aide: chaîne *clssn-aide*

# Conseils pour régler ses problèmes de programmation

- ▶ Google est votre meilleur ami
- ▶ Apprendre à débugger dans la console, comprendre la logique du langage
- ▶ Relire son code: la plupart du temps, le problème n'est qu'une toute petite erreur
- ▶ Lire la documentation des *packages*