

# R + LATEX

Une très brève introduction

# **Objectifs & Philosophie**

# Objectifs & Philosophie

- ▶ Installation: RStudio + R + L<sup>A</sup>T<sub>E</sub>X

# Objectifs & 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 & 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 Languages, 1 Software**



# **Installation**

## **2 Languages, 1 Software**



1. R : [www.cran.rstudio.com](http://www.cran.rstudio.com)

# Installation

## 2 Languages, 1 Software



1. R : [www.cran.rstudio.com](http://www.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 Languages, 1 Software



1. R : [www.cran.rstudio.com](http://www.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 Languages, 1 Software

The screenshot shows the RStudio interface with the following details:

- Project:** (None)
- Environment:** Shows "Environment is empty".
- Files:** Displays several files: JfMorin.tex\*, Class3.tex, beamerthemeCement\_WorkshopR.sty, CreateMaps.R, Class2.tex, and mapWorld.
- R Script:** The main pane contains R code for mapping:

```
## MAP THE ADDITIVE SCALE
library(RColorBrewer)
library(mapprojtools)
library(ggplot2)
library(rworldmap)

# Replace Badly coded countries in df
Data$region[Data$region == "Russian Federation"] <- "Russia"
Data$region[Data$region == "United Kingdom"] <- "UK"
Data$region[Data$region == "Congo"] <- "Republic of Congo"
Data$region[Data$region == "DR Congo"] <- "Democratic Republic of the Congo"

# 
mapWorld = map_data(map='world')
mapWorld = merge(Data, mapWorld, by='region', all.y=TRUE)
mapWorld = mapWorld[order(mapWorld$order), ] # <---

# Fix missing code in World
mapWorld$region[mapWorld$region == "Denmark"] <- "Greenland"

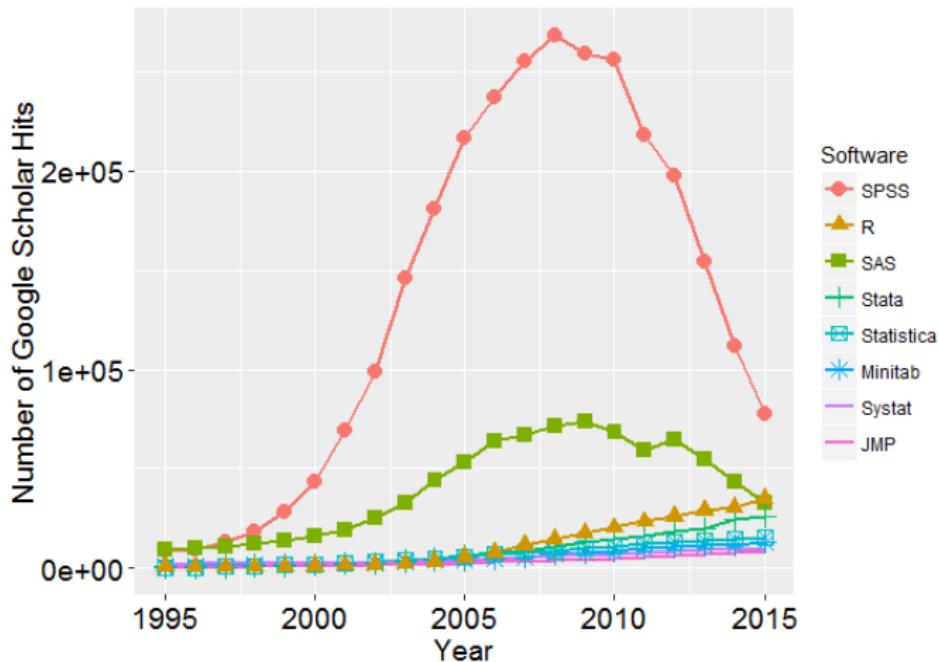
96:53  (Untitled) ±
```
- Console:** Shows the command `./CreateMaps.R` being run, with the output:

```
>
>
>
>
>
>
>
>
> 1+1
[1] 2
> |
```

# **Le monde de l'Open Source**

# Pourquoi R ?

# Pourquoi R ?



# 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
5. *Open source*: Développer par et pour les chercheurs

# Pourquoi R ? Les raisons de détester

1. Programmer du code = Courbe d'apprentissage raide
2. Développement éclectique. Par moment... 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 automatique des trucs comme les tableaux, les 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
- ▶ Certains journaux 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}
```

# L<sup>A</sup>T<sub>E</sub>X: C'est une blague?!



**LATEX: Non.**



# 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 \documentclass{article}
2 \usepackage[utf8]{inputenc}
3 \usepackage[T1]{fontenc}
4 \usepackage{cjk}
5 \usepackage{amsmath}
6 \usepackage{amssymb}
7 \usepackage{amsthm}
8 \usepackage{mathbbm}
9 \usepackage{color}
10 \usepackage{listings}
11 \usepackage{fancyvrb}
12 \usepackage{tikz}
13 \usepackage{tikz-qtree}
14 \usepackage{tikz-qtree-compat}
15 \usepackage{tikz-qtree}
16 \usepackage{tikz-qtree}
17 \usepackage{tikz-qtree}
18 \usepackage{tikz-qtree}
19 \usepackage{tikz-qtree}
20 \usepackage{tikz-qtree}
21 \usepackage{tikz-qtree}
22 \usepackage{tikz-qtree}
23 \usepackage{tikz-qtree}
```

# L<sup>A</sup>T<sub>E</sub>X: Code (Partie 2)

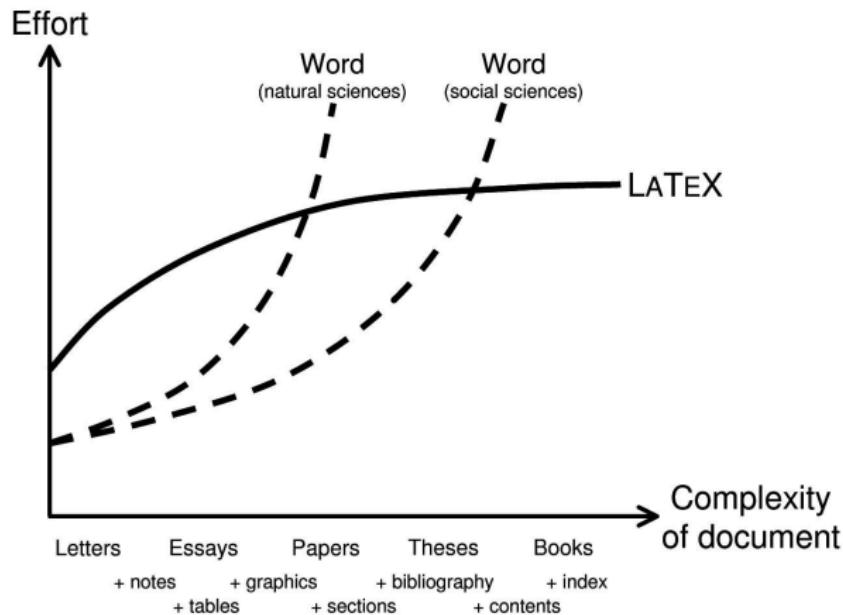
<...>

```
1 \documentclass{article}
2 \usepackage[utf8]{inputenc}
3 \usepackage[T1]{fontenc}
4 \usepackage{amsmath}
5 \usepackage{amsfonts}
6 \usepackage{amssymb}
7 \usepackage{pstricks}
8 \usepackage{color}
9 \usepackage{listings}
10 \usepackage{float}
11 \usepackage{fancyvrb}
12 \usepackage{array}
13 \usepackage{longtable}
14 \usepackage{booktabs}
15 \usepackage{url}
16 \usepackage{hyperref}
17 \usepackage{mathbbm}
18 \usepackage{bm}
19 \usepackage{mathtools}
20 \usepackage{mathbbm}
21 \usepackage{bm}
22 \usepackage{mathtools}
23 \usepackage{bm}
```

# LATEX



# LATEX



# R + LATEX

<...>

```
1 \begin{document} \begin{center} \textbf{R} \textbf{+} \textbf{LATEX} \end{center} \end{document}
```

# 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

# R Les bases de la programmation

# R = Language de programmation

- ▶ Opérateurs de calcul
- ▶ Opérations d'assignement
- ▶ Opérateurs logique
- ▶ Instructions de contrôle

# Opérateurs de calcul

- +
- -
- /
- %%

# Opérateurs logique

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

# Instructions de contrôle

- ▶ if... else
- ▶ for loop

# R Structure de données

# Structure de données

- ▶ Constantes

# Structure de données

- ▶ Constantes
- ▶ Vecteurs

# Structure de données

- ▶ Constantes
- ▶ Vecteurs
- ▶ Data frames

# Constantes

<...>

```
1 #include<iostream>    // Standard Input/Output  
2 #include<string>      // Standard C++ String Class  
3  
4 int main() {  
5     std::cout << "Hello World!"  
6 }
```

# Vecteurs

<...>

```
1 #> vecteur de 1000 éléments de type double contenant des valeurs
2 #> aléatoires entre 0 et 1. Les deux derniers éléments
3 #> sont en rouge.
```

# Data Frames

<...>

```
1 [object Object] [object Object] [object Object] [object Object] [object Object]
```

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	
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	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				

# Fonctions

# Fonctions de base R

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

# Fonction R : mean()

<...>

```
1 mean(c(1, 2, 3, 4, 5))
```

# Créer une fonction en R

# Fonction R : meanGirls()



# Fonction R : meanGirls()

<...>

```
1 function() {  
2   #  
3   #  
4   #}
```

# Fonction R : meanGirlsPlus()

<...>

```
1 #> Fonction qui calcule la moyenne des filles
2 #> et qui ajoute une valeur supplémentaire
3 #> à cette moyenne
4 #> 
5 #> Fonction qui calcule la moyenne des filles et ajoute une valeur supplémentaire
6 #> à cette moyenne
7 #>
8 #> Fonction qui calcule la moyenne des filles
9 #>
```

# Maintenant?

# Maintenant? Plus de fonctions R ...

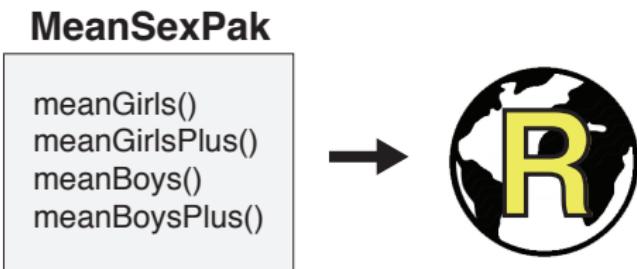
```
meanGirls()  
meanGirlsPlus()  
meanBoys()  
meanBoysPlus()
```

# Maintenant? Un package R

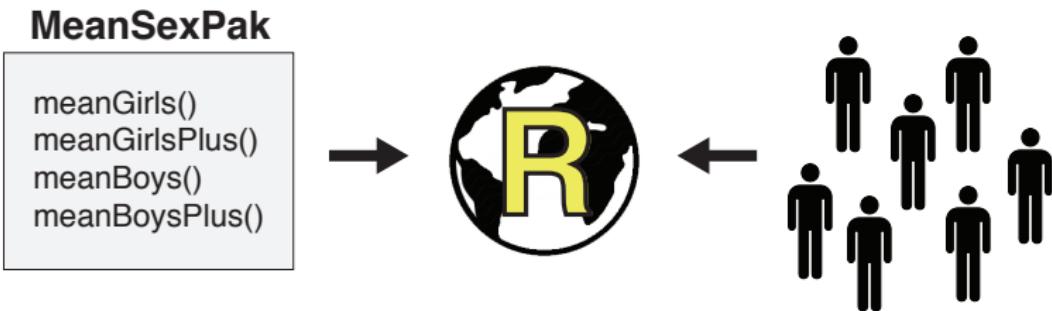
## MeanSexPak

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

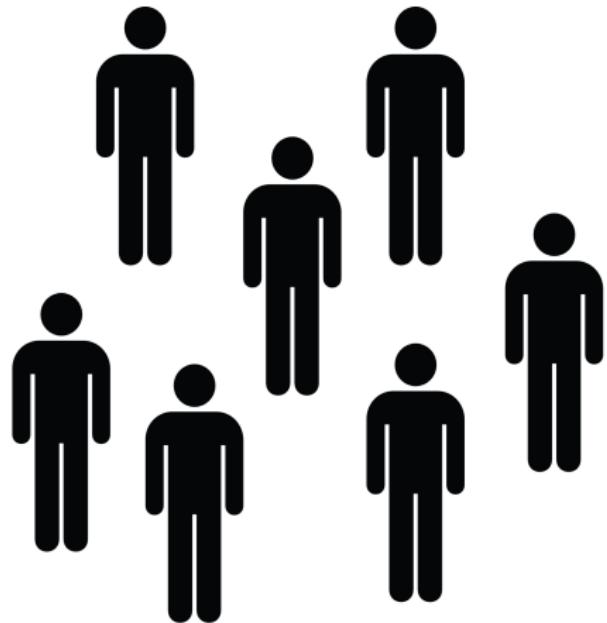
# Maintenant? La publication d'un package



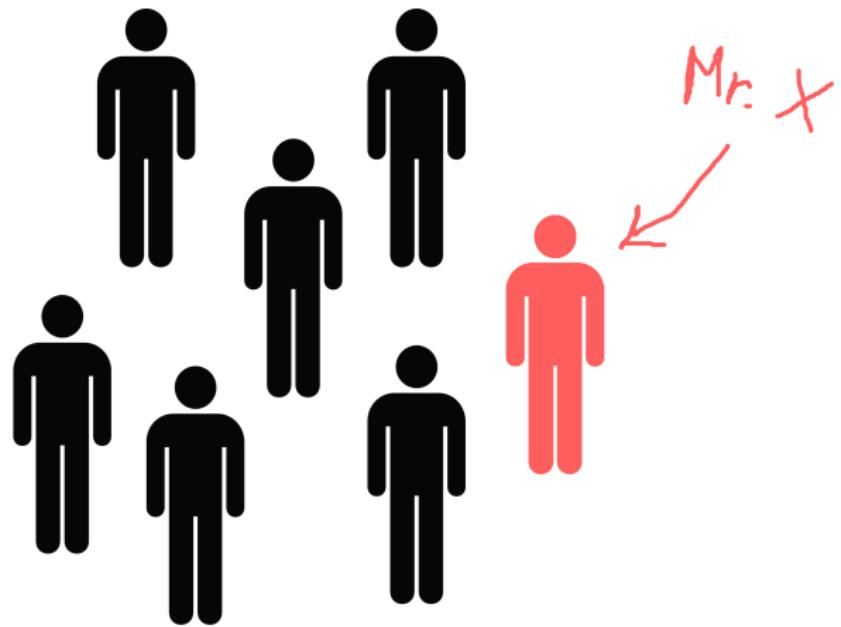
# Maintenant? Diffusion à la communauté



# Communauté R



# Mr. X



# Then? Mr. X Installs the Package

<...>

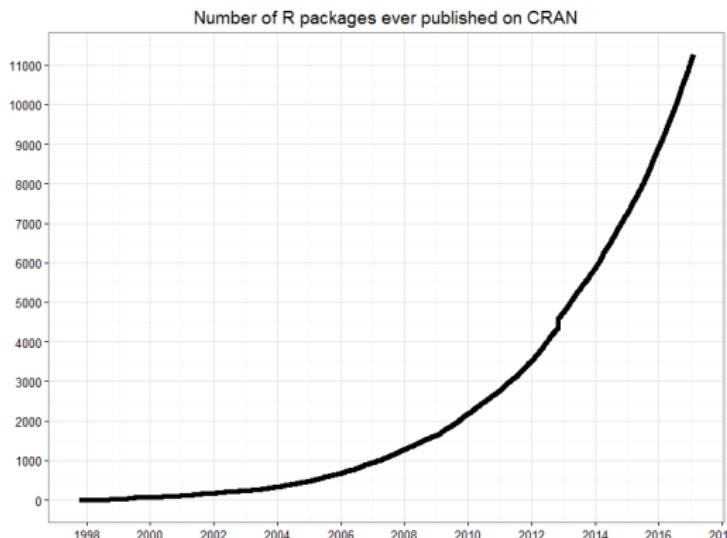
```
1 [REDACTED]
```

# Ensuite? Mr. X utilise le package

<...>

```
1 package com.mr.x;
2
3 import org.openqa.selenium.WebDriver;
4 import org.openqa.selenium.WebElement;
5 import org.openqa.selenium.support.FindBy;
6 import org.openqa.selenium.support.PageFactory;
```

# Nombre de packages R



**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 *Template* et de *Stylesheets*
  - ▶ Offre également des sites web à personnaliser

# Wordpress : Astra

- ▶ Astra c'est quoi ?
  - ▶ Un thème construit pour Wordpress
  - ▶ Une collection de *Template* et de *Stylesheets*
  - ▶ Offre également des sites web à personnaliser
- ▶ Pourquoi Astra ?
  - 1 Facilité d'utilisation: inclus 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

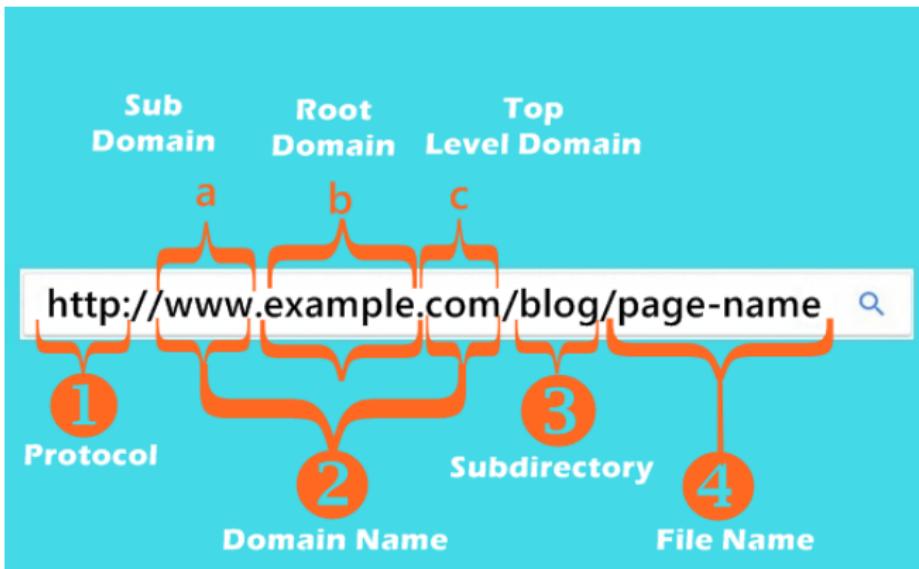
# WWW : URL, HTML et HTTP ?

- ▶ URL = Uniform Resource Locator
  - ▶ Adresse du site web
- ▶ HTTP = Hypertext Transfer Protocol
  - ▶ Transmettre et formater les commandes
  - ▶ Indique 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
  - ▶ Indique 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

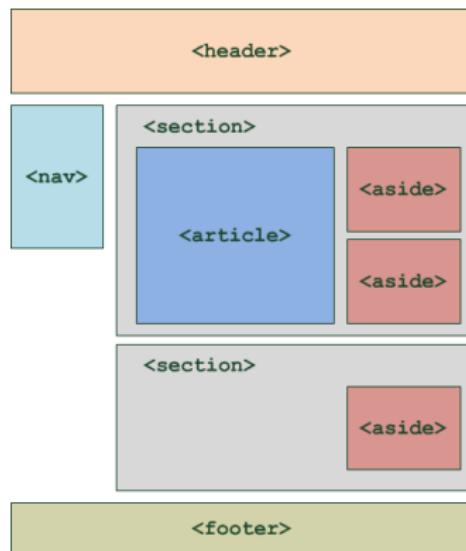
# Explorons une URL



# **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 de recherche en R

# Avec R, penser autrement les possibilités de recherche

# Plan de la présentation

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

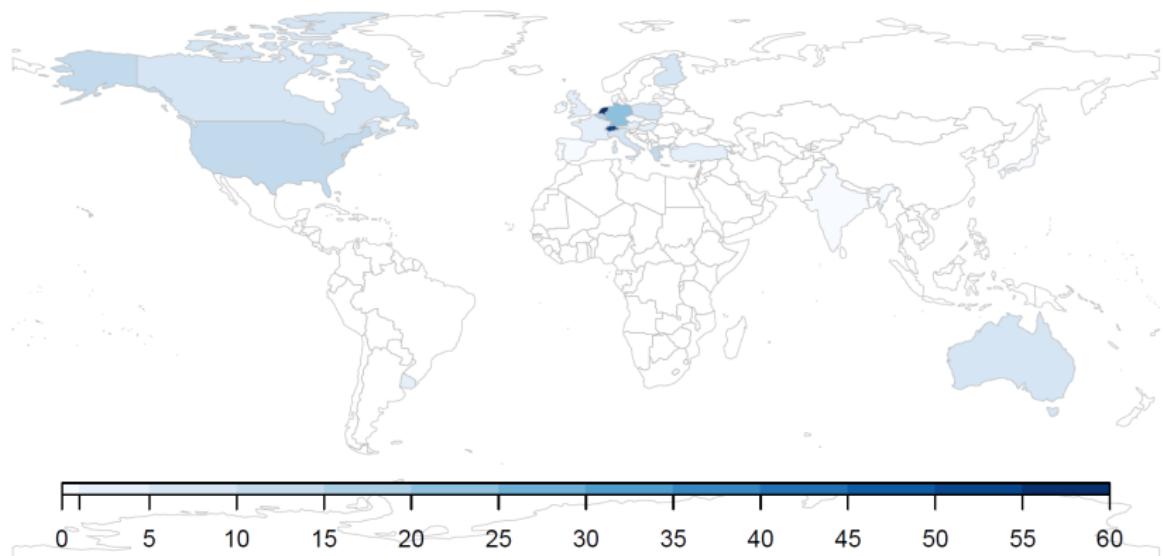
# Scoping review: Cartographier la littérature scientifique

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

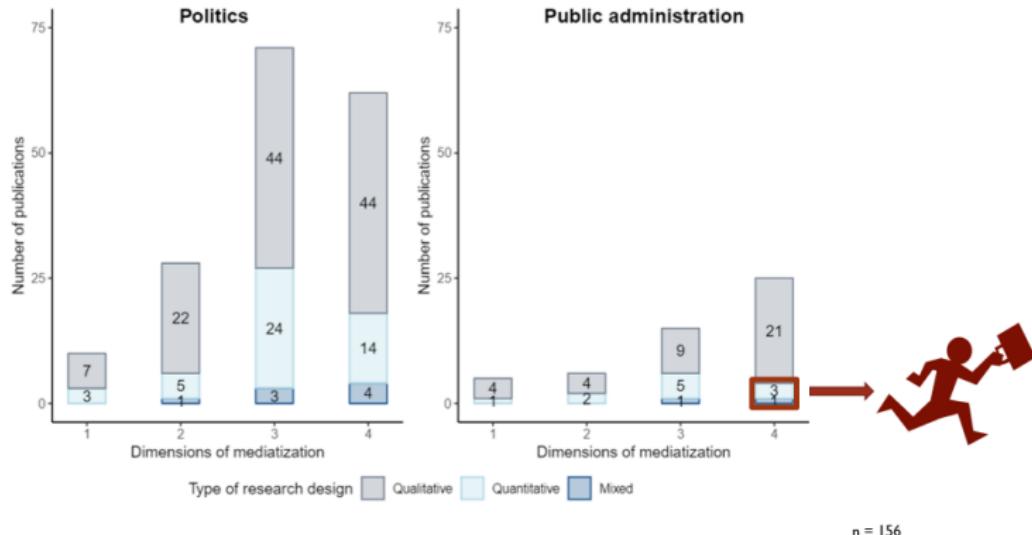
# Scoping review: Et R alors ?

Cartographier de la littérature



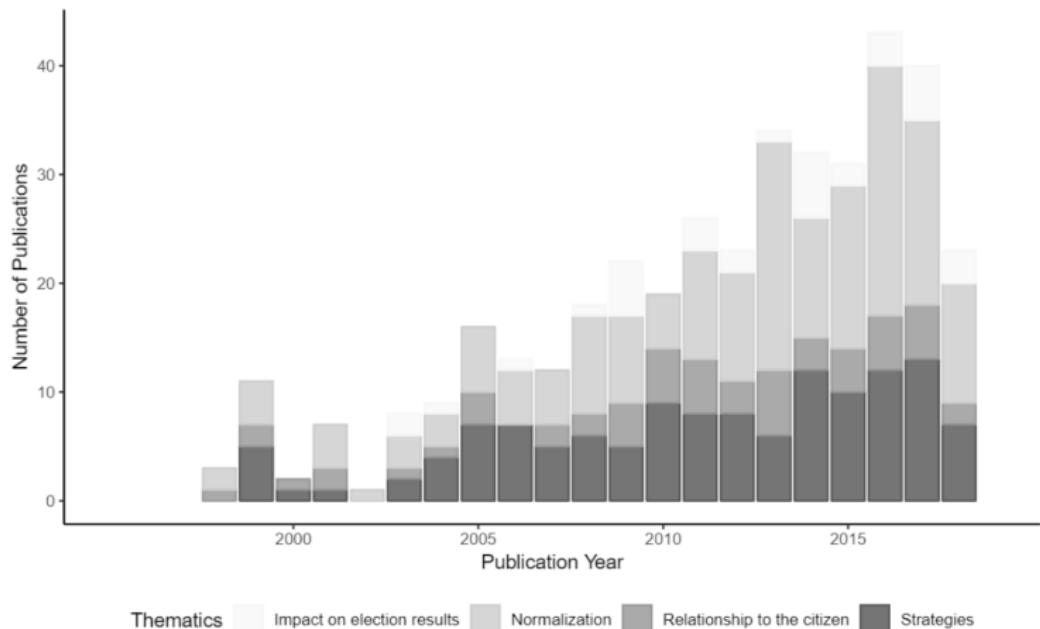
# Scoping review: Et R alors ?

Visualiser la distribution de certaines caractéristiques



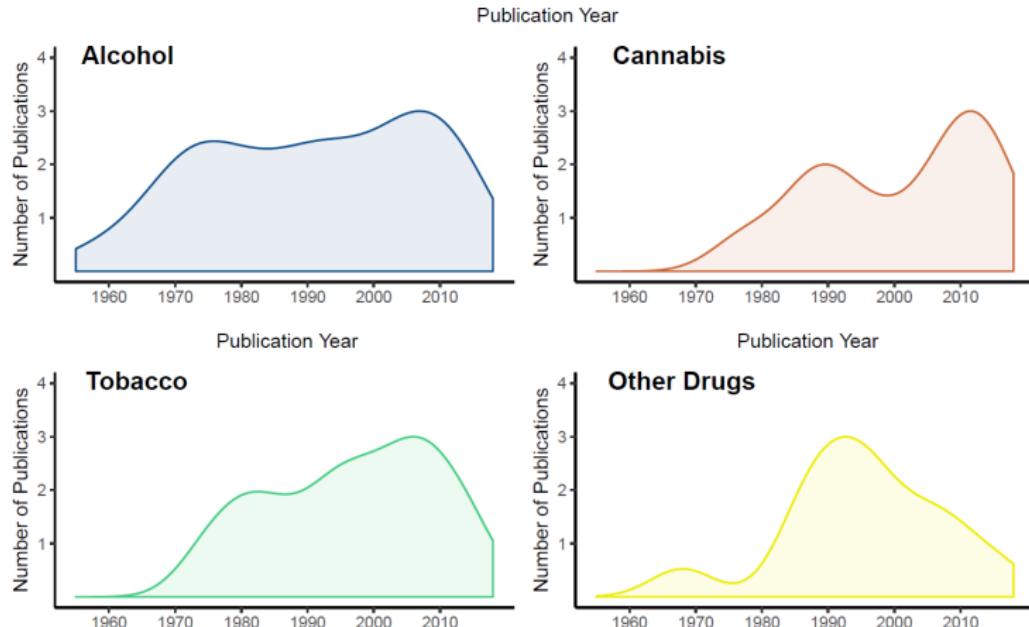
# Scoping review: Et R alors ?

Visualiser la distribution de certaines caractéristiques



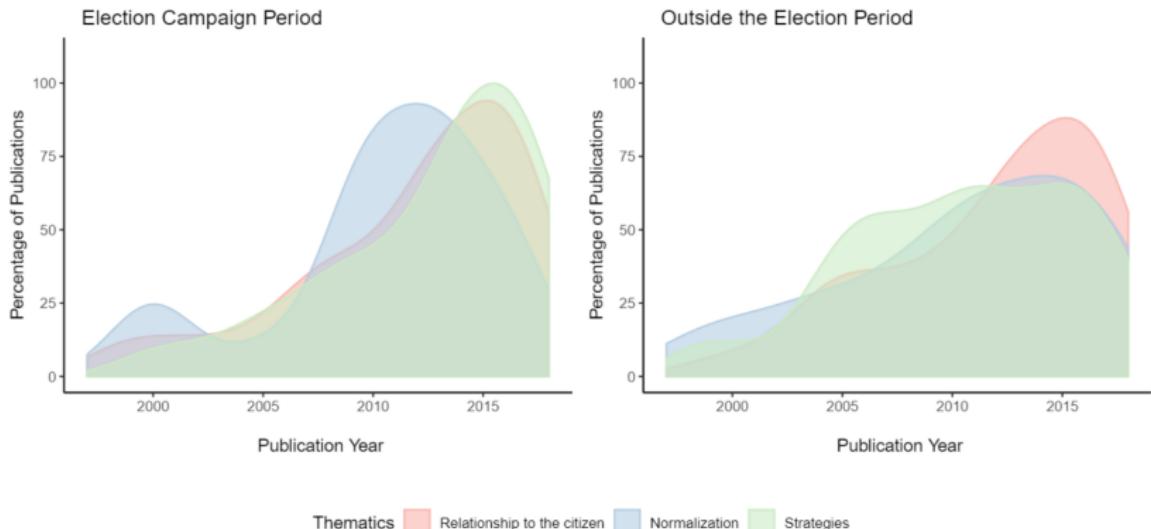
# Scoping review: Et R alors ?

Visualiser la distribution de certaines caractéristiques



# 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 :

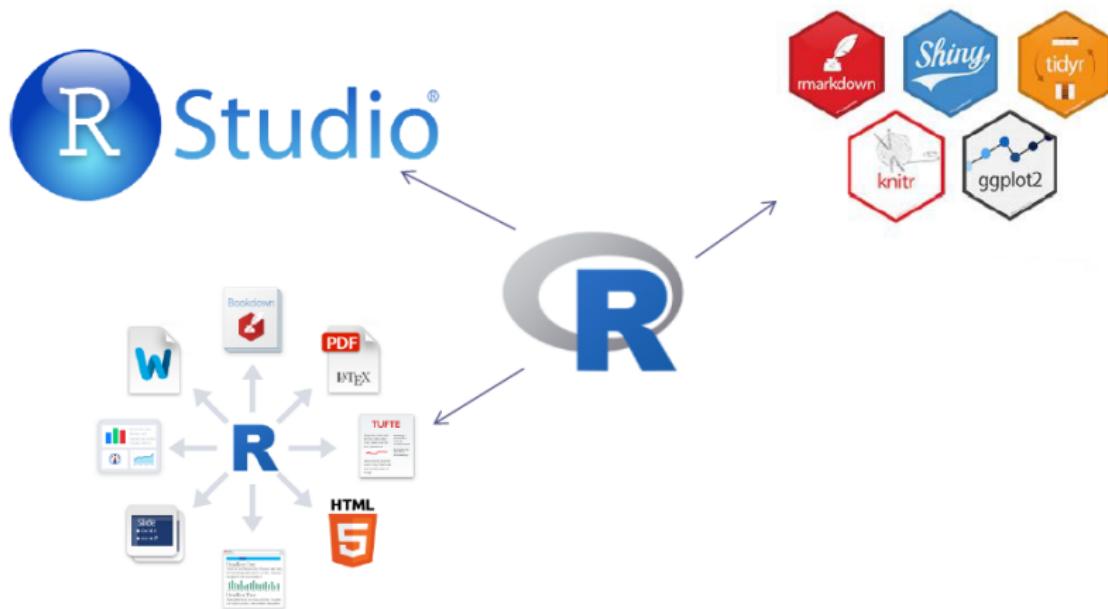
- ▶ Coder dans l'optique de pouvoir utiliser le code pour plusieurs projet
- ▶ Mise à jour automatique des visualisations graphiques lorsque les données changent
- ▶ Possibilités de visualisations graphiques



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

- ▶ Environnement d'application web pour R
- ▶ Permet de transformer des analyses en application web réactives
- ▶ 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)

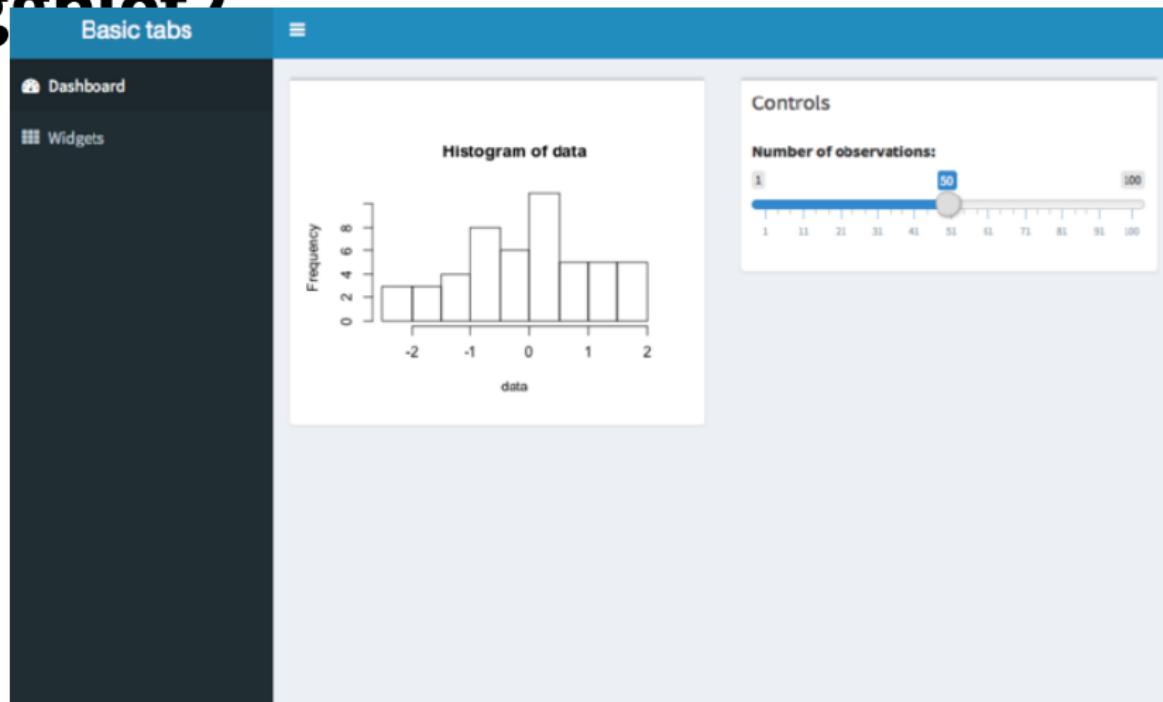
# Define server logic required to draw a histogram
server <- function(input, output) {

  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 )
```

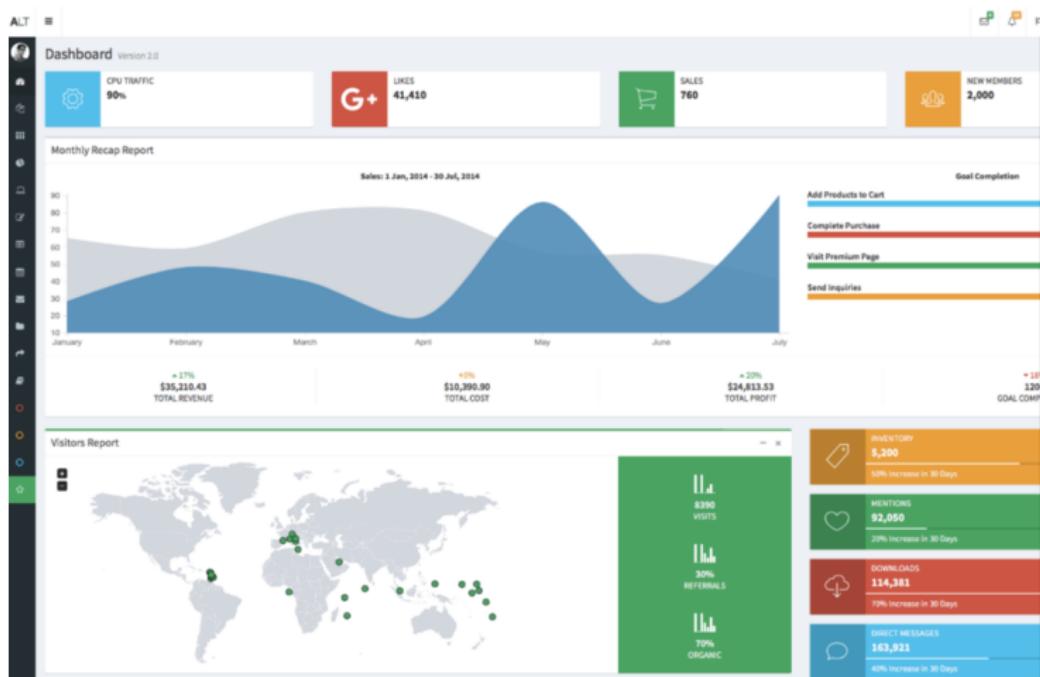
# Shiny: Un exemple très simple avec ggplot2



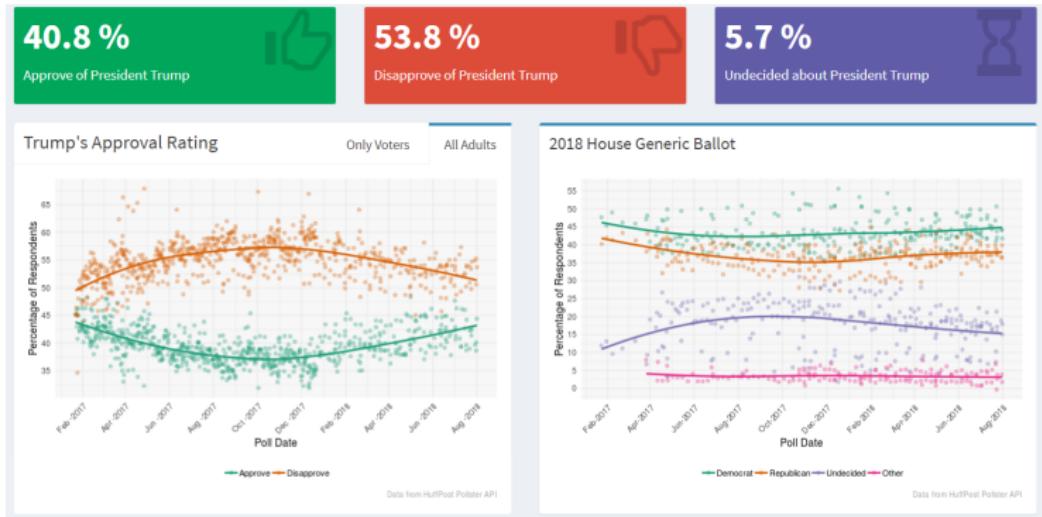
# Shiny: Un exemple très simple avec ggplot2



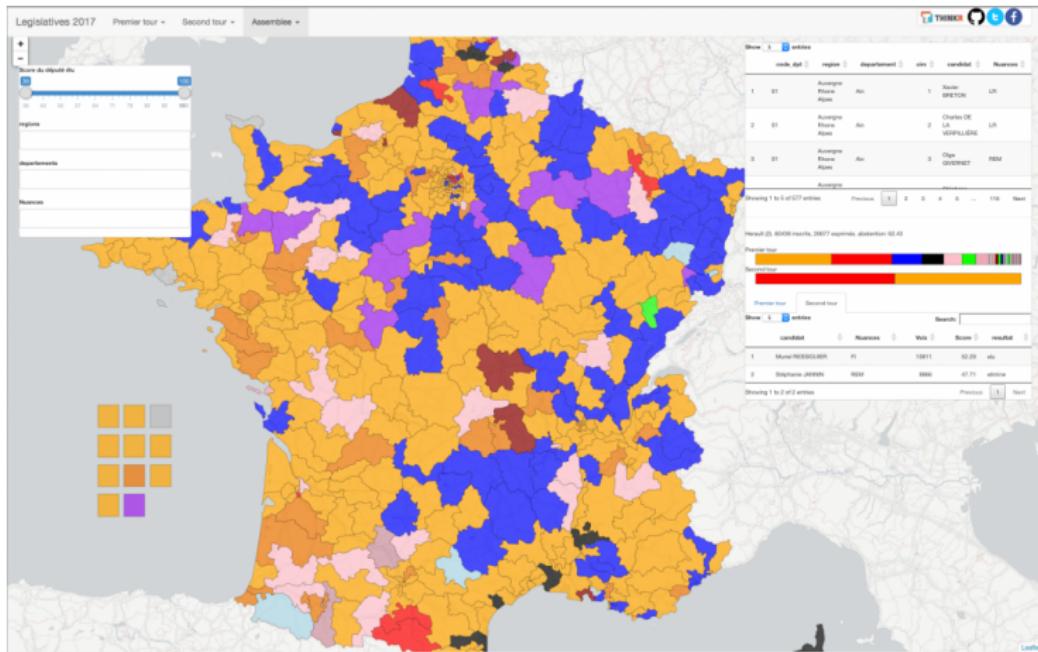
# Shiny: Des exemples plus raffinés



# Shiny: Des exemples plus raffinés



# Shiny: Des exemples plus raffinés



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

**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 débutants à avancés
- ▶ Exercices théoriques et pratique, 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, and 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 (part1); 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 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, 2 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 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 déjà répondues
- ▶ Communauté sympathique et dynamique

# Slack



- ▶ Outils de communication avec différents chaînes, publiques et privées
- ▶ Possibilité d'y joindre d'autres applications, tels 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 *clessn-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