Introduction aux mégadonnées en sciences sociales (FAS 1001)

Cours 8: *Web scraping*

Nadjim Fréchet

12 mars 2024

Retour sur le TP3 😮

- Pas encore terminé de corriger. Vous aurez les notes ce soir ou demain.
- Pour les travaux de sessions. Je vais tout faire pour terminer les corrections pour le début de la semaine prochaine.
- Pour le TP4. Je vous donne une semaine de plus! 🥳

> **ATTENTION**
>
> **RAPPEL DE LA DATE DE REMISE: lundi 25 mars avant minuit**. Et oui, une semaine de plus.

Invité de la semaine 😎

- **Benjamin Guinaudeau**
- As du codage. Spécialisation en analyses textuelles automatisées, Machine learning et data science.
- **Article avec des données de TikTok: APSA ITP Best Journal Article Award**
- **MAINTENANT PROFESSEUR À L'UNIVERSITÉ LAVAL À QUÉBEC!**

Lecture de la semaine 📚🤭

- Chapitre 7 du **livre** de Alexander (**2023**).
- Introduction aux APIs. C'est quoi un API (ou *Application Programming Interface*) selon vous?
- Certains dans cette classe ont utilisé un API (sous forme d'un fichier `JSON`).
- Accès (ou appel) à des données (entreprises, groupes, particuliers, etc.) disponibles sous certaines conditions (clairement explicitées).
- Souvent accessible avec des packages (R, Python ou autres) développés par des programmeurs ou les propriétaires des données.

Exemple d'API 📈 TheyWorkForYou

- Site pour chercher des données parlementaires britanniques.
- **Exemple de clé**
- **Package de disponible pour télécharger les données**

Exemple d'API 📈 TheyWorkForYou

```
# Load package

library(twfy)
library(tidyverse)

# Set API key

my_key <- "key"

set_api_key(my_key)

# Chercher les données de David Cameron

# getDebates(type ="commons", person = 10777)
```

C’est bien quand c’est accessible…

- Qu’est-ce qu’on fait quand les données ne sont pas accessibles?
- ***Web scraping*** direct?
- Discussion sur la légalité du scraping. Est-ce une pratique sans limite?

Web scraping

- Plusieurs packages de disponibles dans différentes langues de programmation. Exemple `BeautifulSoup` dans `Python`.
- Dans R pour le scraping de page statique, le package **`rvest`** est un excellent outil de travail.
- Pour *scraper* des données tirées du web, un peu d'exploration de page s'impose. On peut le faire de différentes manières, notamment avec **SelectorGadget**.
- `SelectorGadget` permet de chercher des éléments **HTML** de la page web. D'autres manières possibles, notamment de regarder directement le code source de la page.

Exemple avec une page *Wikipedia*

- Pratiquez-vous sur une page *Wikipédia*! Les éléments d'une page *Wikipédia* restent plutôt stables à travers le temps.
- Exemple avec la page *Wikipédia* **suivante**.

```
#### Scraping de données de sondages canadiens ####

# 1 - Libraries ----

library(tidyverse)
library(lubridate) # Package pour gérer des données temporelles
library(rvest)     # Package pour le scraping

# 2 - Scraping ----

Polls_data <- read_html("https://en.wikipedia.org/wiki/Opinion_poll
```

Exemple avec une page *Wikipedia*

- On cherche les éléments de la page qui seront nécessaires pour notre travail.

```
Polls_data_1 <- Polls_data |>
  ### Chercher l'élément "table" ###
  html_elements("table") |>
  ### Chercher le deuxième élément (ou table) de la liste ###
  pluck(2) |>
  ### Transformation en matrice de données ###
  html_table(fill = T)
```

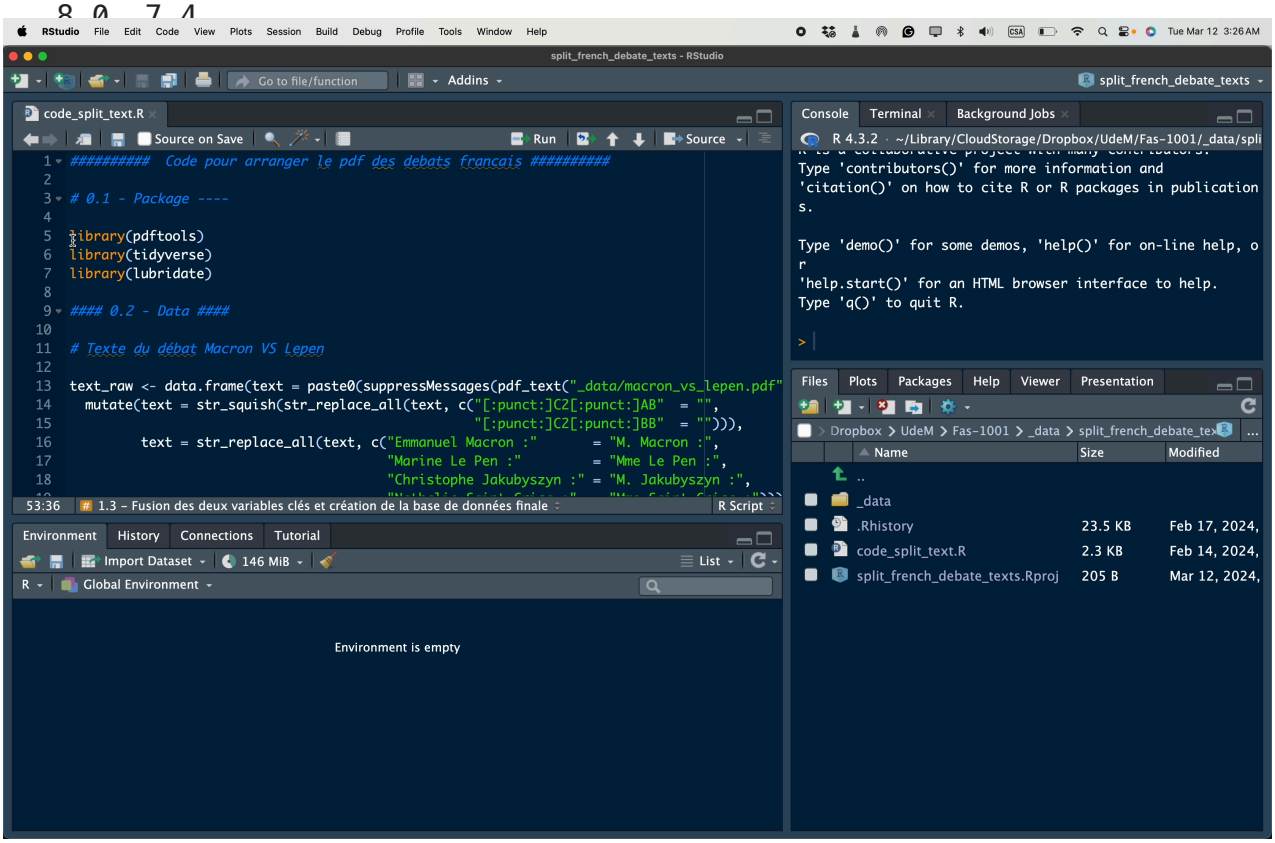
Importance de nettoyer ses données

```
Polls_data_f <- Polls_data_1 |>
  ### renommer les variables pertinentes ###
  rename(firm           = "Polling firm",
         date           = "Last dateof polling[a]",
         pred_cpc       = CPC,
         pred_lpc       = LPC,
         pred_ndp       = NDP,
         pred_bq        = BQ,
         pred_ppc       = PPC,
         pred_gpc       = GPC,
         error_margin   = "Marginof error[c]",
         sample_size    = "Samplesize[d]",
         poll_method    = "Polling method[e]",
         leading_margin = Lead) |>
  ### Nettoyage des variables ###
  filter(firm != "") |>
  # Enlever des rangées inutiles #
  slice(-c(1,235,236)) |>
  # Retirer les variable de trop #
```

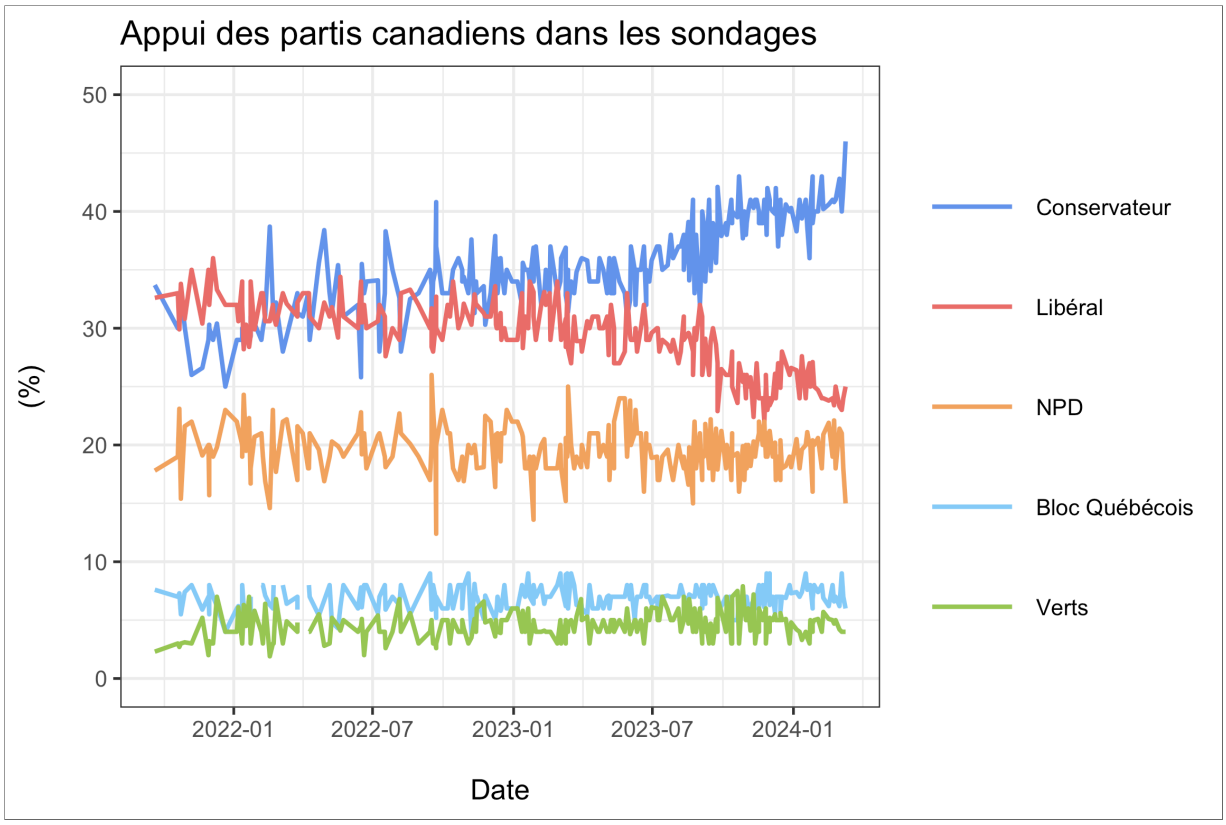
Importance de nettoyer ses données

```
1 # Visualisation de nos données
2
3 glimpse(Polls_data_f)
```

```
Rows: 233
Columns: 12
$ firm           <chr> "Mainstreet Research", "Abacus Data", "Angus
Reid", "Na…
$ date           <date> 2024-03-09, 2024-03-06, 2024-03-04, 2024-03-
01, 2024-0…
$ pred_cpc       <dbl> 46.0, 42.0, 40.0, 42.8, 41.0, 40.8, 41.0,
40.6, 40.2, 4…
$ pred_lpc       <dbl> 25.0, 24.0, 23.0, 23.3, 25.0, 23.4, 24.0,
23.8, 24.0, 2…
$ pred_ndp       <dbl> 15.0, 18.0, 21.0, 21.4, 18.0, 22.1, 19.0,
21.9, 20.9, 1…
$ pred_bq        <dbl> 6.0, 7.0, 9.0, 6.2, 7.0, 6.6, 8.0, 6.2, 6.9,
8.0, 7.4
```



Visualisation graphique



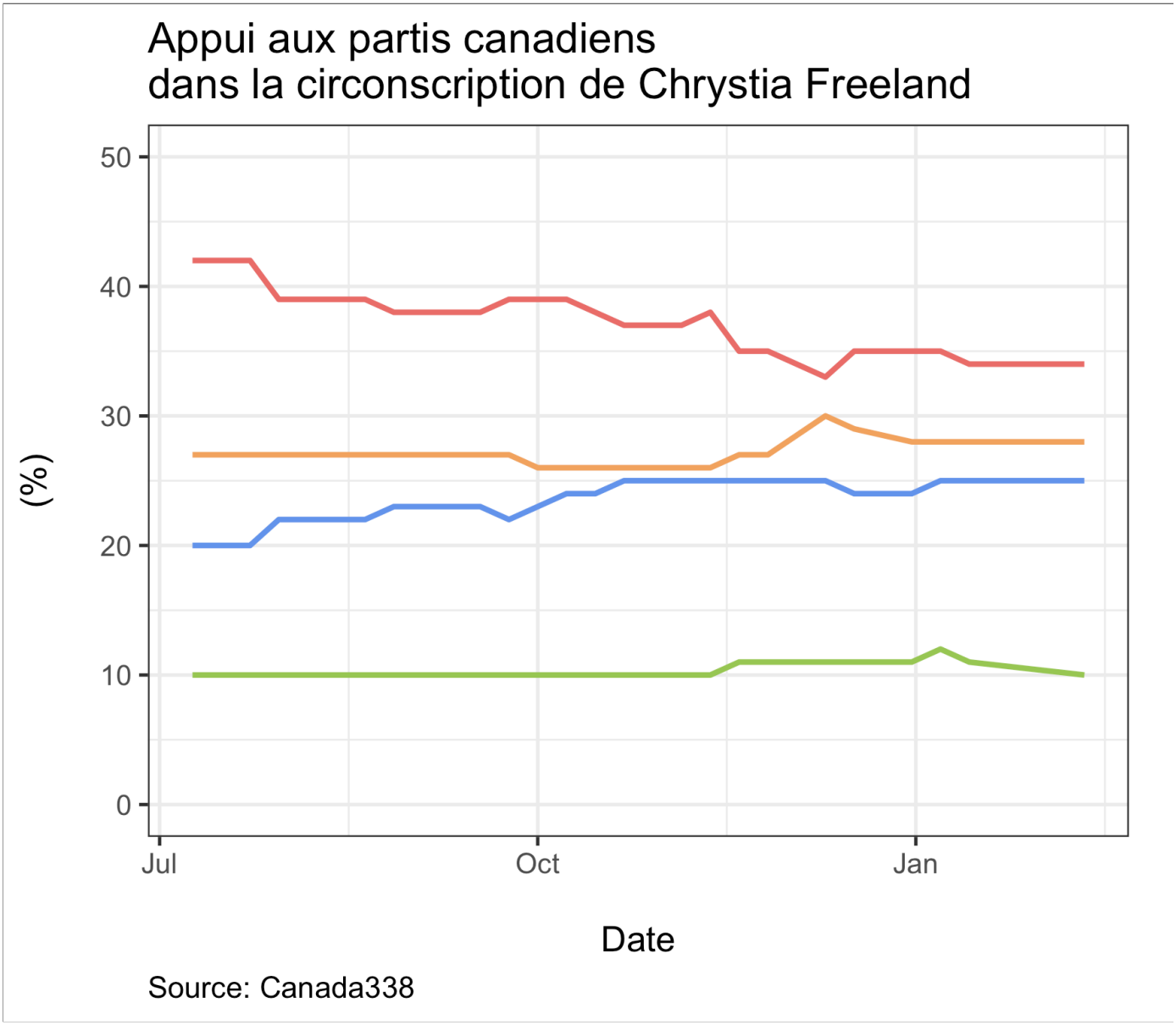
Scraping plus complexe

Autres scraping possible de pages PDF

- Voir le **chapitre 7** de Alexander (**2023**) pour différents exemples.
- Travail mené par certaines d'entre vous.

Scraping de pages PDF

Possibilité d'automatiser le processus de scraping 📈



Bibliographie

Alexander, Rohan. 2023. *Telling Stories with Data: With Applications in R*. Chapman; Hall/CRC.

Error

×