

# Importation et Exportation des données

Joël Kazadi

2022-10-13

Dans cette section, nous allons étudier comment importer et exporter des tableaux de données avec R. Les packages à utiliser pour ces opérations varient en fonction du format du tableau de données. Nous nous focaliserons uniquement sur des fichiers en format `.csv` et `.xlsx`.

## 1 Importation

Trois cas d'importation seront passés en revue : (i) les fichiers `.csv` ; (ii) les fichiers `.xlsx` ; et (iii) les fichiers en ligne.

### 1.1 Format `.csv`

Le package requis pour réaliser l'importation d'un tableau de données en format `.csv` est `readr`.

```
# install.packages("readr")
library(readr)
setwd("C:/Users//USER/Documents/Data_Science/RStudio/")
data_iris <- read_csv("iris.csv", col_names = T)
head(data_iris, n = 4L) # ce code permet de visualiser les 4 premières lignes du tableau de données
```

```
## # A tibble: 4 x 6
##   ...1 Sepal.Length Sepal.Width Petal.Length Petal.Width Species
##   <dbl>         <dbl>         <dbl>         <dbl>         <dbl> <chr>
## 1     1           5.1           3.5           1.4           0.2 setosa
## 2     2           4.9           3             1.4           0.2 setosa
## 3     3           4.7           3.2           1.3           0.2 setosa
## 4     4           4.6           3.1           1.5           0.2 setosa
```

### 1.2 Format `.xlsx`

Le package requis pour réaliser l'importation d'un tableau de données en format `.xlsx` est `readxl`.

```
# install.packages("readxl")
library(readxl)
firms <- read_excel("Data_frame.xlsx", sheet = "time_series")
head(firms, n = 4L)
```

```
## # A tibble: 4 x 6
##   date statut effectif utility    ca charges
##   <dbl> <dbl>     <dbl>   <dbl> <dbl>   <dbl>
## 1 1970     4         2       3 3550   5185
## 2 1971     4         1       2 3975   5000
## 3 1972     1         3       2 4649   4538
## 4 1973     2         1       4 4691   4568
```

### 1.3 Importation en ligne

L'importation d'un tableau de données disponible en ligne nécessite d'indiquer préalablement le lien web où ces données ont été publiées.

```
link <- "https://archive.ics.uci.edu/ml/machine-learning-databases/00611/accelerometer.csv"
data_online <- read.csv(file = link, header = TRUE, sep = ",", dec = ".")
head(data_online) # visualisation des 6 premieres lignes
```

```
##   wconfid pctid      x      y      z
## 1      1    20 1.004  0.090 -0.125
## 2      1    20 1.004 -0.043 -0.125
## 3      1    20 0.969  0.090 -0.121
## 4      1    20 0.973 -0.012 -0.137
## 5      1    20 1.000 -0.016 -0.121
## 6      1    20 0.961  0.082 -0.121
```

```
tail(data_online) # visualisation des 6 dernieres lignes
```

```
##      wconfid pctid      x      y      z
## 152995      3   100 0.703 -0.207  0.043
## 152996      3   100 1.051  0.090 -0.262
## 152997      3   100 0.918  0.039 -0.129
## 152998      3   100 1.156 -0.094 -0.227
## 152999      3   100 0.934  0.203 -0.172
## 153000      3   100 1.199 -0.176  0.109
```

## 2 Exportation

Reprenons l'exemple du dataframe créé dans la section précédente relative à la structure des données en R.

```
data <- data.frame(Econometrie = c(18,16,17,18),
                   Statistique = c(14,13,15,19),
                   Niveau = c("Bon", "Moyen", NA, "Excellent"),
                   row.names = c("Kadima", "Kazadi", "Nsamba", "Malu"))
print(data)
```

```
##      Econometrie Statistique Niveau
## Kadima          18          14    Bon
## Kazadi          16          13  Moyen
## Nsamba          17          15   <NA>
## Malu            18          19 Excellent
```

Pour exporter l'objet `data` créé précédemment au format `.csv`, on utilise la fonction `write_csv()` du package `readxl`.

```
library(readxl)
write_csv(data, "data_grades.csv", col_names = T)
```

Pour les grands tableaux de données, il est généralement recommandé de sauvegarder le travail au format R. Voici le code :

```
save(data, file = "data_grades.RData")
```

Pour charger dans l'environnement la session R précédemment enregistré, on procède comme suit :

```
load("data_grades.RData")
```