



Préconisations pour importer la table csv en R



Les packages qui peuvent être utilisés pour lire les tables csv sont *readr* ou *data.table*. Pour les charger :

```
if (!require(readr))
  install.packages("readr")
library(readr)

if (!require(data.table))
  install.packages("data.table")
library(data.table)
```

Le csv peut être lu avec la fonction `read.csv2` du package *readr* :

```
[nom_de_la_table_lue] <- read.csv2(
  "[chemin/nom_de_la_table_a_lire].csv",
  encoding = "UTF-8",
  sep = ";",
  colClasses = "character",
  quote = ""
)
```

ou avec la fonction `fread` du package *data.table* :

```
[nom_de_la_table_lue] <- fread(
  "[chemin/nom_de_la_table_a_lire].csv",
  encoding = "UTF-8",
  sep = ";",
  colClasses = "character",
  quote = ""
)
```

(ci-dessus, on lit les variables comme étant toutes de type caractère pour éviter la disparition des zéros en première position des codes géographiques ; il appartient ensuite à l'utilisateur de convertir les variables numériques, telles que les coordonnées géographiques ou la capacité).

Remarque : la fonction *write.csv* utilise un point comme séparateur décimal et une virgule comme séparateur de variables ; la fonction *write.csv2* utilise une virgule comme séparateur décimal et un point-virgule comme séparateur de variables. Ces caractéristiques ne sont pas paramétrables dans ces deux fonctions, conçues pour s'assurer que les bonnes conventions sont utilisées pour exporter un fichier valide. Les tentatives de modification sont ignorées, avec un avertissement.

C'est pourquoi la fonction *write.table* a été utilisée pour pouvoir exporter avec un point comme séparateur décimal et un point-virgule comme séparateur de variables :

```
write.table(
  [nom_de_la_table_a_exporter],
  "[chemin/nom_de_la_table_exportee].csv",
  quote = FALSE,
  sep = ";",
  na = "",
  dec = ".",
  row.names = FALSE,
  col.names = TRUE,
  fileEncoding = "UTF-8"
)
```