Analyse de données

Gwenlake

Présentation

Enseignant

Guillaume Béguec, guillaume.beguec@gwenlake.com

- Formation universitaire : Mathématiques appliquées et statistique.
- Data scientist à Gwenlake : Modélisation de données, labelling, clustering, traitement de données textuelles.
- Secteurs : Médical, maritime, banques, organismes publics, …

Objectifs du cours

- Initiation aux outils R pour de l'analyse de données.
- Importation et manipulations d'objets dataframe.
- Premières visualisations.

Rappels rapides des bases de R

Les objets R

Les variables et types de base:

- Texte (**character**), nombre (**numeric, integer**), booléen (**logical**), **NULL, list, dates**

Les Structures de données :

- Vecteurs : une liste d'éléments **du même type**. Exemple : *c(1, 5, 6)*
- List : ensemble d'éléments de type différents. Exemple : list(1, "a", True)
- Matrices : tableaux homogènes. Chaque ligne et colonne sont du **même type**.
- DataFrame : liste de colonnes (variables) **de même longueur** et **de types mixtes.**
  - Exemple: data.frame(prenom = c("Ana", "Ben", "Léa"), age = c(23, 31, 27), ville = c("Paris", "Lyon", "Nice"))
- Factors : variable de catégories avec une liste de valeurs possibles (levels)
  - Exemple : Taille de vêtements : factor(c("M", "M", "M", "L", "L"), levels=c("S","M","L"))

Rappels rapides des bases de R

Fonctions

- Une fonction prend des valeurs en entrée (arguments) et renvoie un résultat.
- Création d'une fonction en R : *f <- function(x) { x * 2 }*.
  - f(3) renvoie 6.
- Intérêt : éviter de répéter du code et réutiliser facilement la même opération ailleurs.

Packages

- Un package est un ensemble de fonctions/données qui ajoutent des capacités à R.
- Installer (une fois) : *install.packages("dplyr")*
- Charger et utiliser (à chaque session) : *library(dplyr)*
- Intérêt : des outils prêts à l'emploi pour aller plus vite (nettoyage, graphes, stats).
- Exemples : Dyplr, tidyverse, factominer, …

Rappels rapides des bases de R

Charger et lire un CSV

- Un CSV est un fichier texte tabulaire où les valeurs sont séparées par des virgules (ou des points-virgules).
- Lire en base R :
  - *df <- read.csv("donnees.csv")*
  - si séparateur **;** df <- *read.csv2("donnees.csv") ou df <- read.csv("donnees.csv",* , sep=";"*)*
- Lire avec tidyverse (un package) : df <- readr::read_csv("donnees.csv").
- Intérêt : importer facilement des données pour les analyser dans R.

Manipulation de DataFrames

Fonctions et opérations de base

- names() : noms des colonnes, nrow() : nombre de lignes, ncol(): nombre de colonnes.
- df[1,2]: extrait la première ligne, deuxième colonne.
- df[1,] ou df[,1]: extrait toute la première ligne ou toute la première colonne.
- df$col : Extrait la colonne « col » du dataframe df.
- df$col2 <- sapply(df$col1, as.character) : créé ou réécrit dans col2 une variable texte de col1.
- df2 <- subset(df2, nationality %in% c("FRA", "GER")) : recréé un dataframe où la variable nationality vaut « FRA » (Française) ou « GER » (allemande).
- order(-a1$Total) : Récupère les indices de la variable total triée par ordre décroissant.

Exemple : Top 10 de la variable gold (nombre de médailles d'or)

```
top10 <- dataframe_athetes[order(- dataframe_athetes $Total, ]
top10 <- head(top10, 10)
```

Manipulation de DataFrames

Le package dyplr

- Manipulation de dataframe sous forme de pipeline avec les opérateurs |> ou %>%
- df |> names(), df |> nrow(), df |> ncol()
- df |> summarise(n = n()) : compter les lignes
- df |> slice(1) |> select(2) : première ligne, deuxième colonne
- df <- df |> mutate(col2 = as.character(col1)) : créé ou réécrit dans col2 une variable texte de col1.
- df2 <- df2 |> filter(nationality %in% c("FRA", "GER"))
- df |> arrange(desc(Total)) : tri dans l'ordre décroissant

Exemple : Top 10 de la variable gold (nombre de médailles d'or)

```
top10 <- dataframe_athetes |>
  arrange(desc(Total)) |>
  slice_head(n = 10)
```