R, Rstudio et Tidyverse

Hugo Harari-Kermadec

M1 APE - Techniques quantitatives

2020

R et Rstudio

R est un logiciel de calcul scientifique libre et gratuit. Le logiciel permet de faire à peu près tout au niveau L3/M1. Il est complété au fur et à mesure des développements de la recherche dans différentes disciplines, de la biologie à l'histoire, pour des usages plus pointus. Chaque complément est appelé "package".

R et Rstudio

Rstudio est une interface qui permet de faciliter l'usage de R. Elle se découpe en 4 fenêtres :

En haut à gauche Le script : c'est un fichier texte où on écrit les instructions qu'on donne à R. On l'enregistre régulièrement pour pouvoir reproduire les opérations. C'est une des bases de la démarche scientifique.
Tapez `1+1`

R et Rstudio

Rstudio est une interface qui permet de faciliter l'usage de R. Elle se découpe en 4 fenêtres :

En haut à gauche Le script : c'est un fichier texte où on écrit les instructions qu'on donne à R. On l'enregistre régulièrement pour pouvoir reproduire les opérations. C'est une des bases de la démarche scientifique.
Tapez `1+1`

En bas à gauche Console : c'est là que Rstudio échange avec R. On envoie une instruction du script à la console (par copier-coller ou avec bouton `Run` ou `ctrl+enter`. Et R répond dans la console.
Envoyez `1+1` du script vers la console.

R et Rstudio

Rstudio est une interface qui permet de faciliter l'usage de R. Elle se découpe en 4 fenêtres :

En haut à gauche Le script : c'est un fichier texte où on écrit les instructions qu'on donne à R. On l'enregistre régulièrement pour pouvoir reproduire les opérations. C'est une des bases de la démarche scientifique.
Tapez `1+1`

En bas à gauche Console : c'est là que Rstudio échange avec R. On envoie une instruction du script à la console (par copier-coller ou avec bouton `Run` ou `ctrl+enter`. Et R répond dans la console.
Envoyez `1+1` du script vers la console.

En haut à droite Environnement : c'est là où l'on trouve les valeurs et bases de données que l'on a créé.
Définissez `a<-1+1`

R et Rstudio

Rstudio est une interface qui permet de faciliter l'usage de R. Elle se découpe en 4 fenêtres :

En haut à gauche Le script : c'est un fichier texte où on écrit les instructions qu'on donne à R. On l'enregistre régulièrement pour pouvoir reproduire les opérations. C'est une des bases de la démarche scientifique.
Tapez `1+1`

En bas à gauche Console : c'est là que Rstudio échange avec R. On envoie une instruction du script à la console (par copier-coller ou avec bouton `Run` ou `ctrl+enter`. Et R répond dans la console.
Envoyez `1+1` du script vers la console.

En haut à droite Environnement : c'est là où l'on trouve les valeurs et bases de données que l'on a créé.
Définissez `a<-1+1`

En bas à droite Annexes : on y trouve les graphiques, l'aide, les packages.

Importation

On veut importer `mini_db`.

- Dans la fenêtre environnement, cliquez sur `Import Dataset`.
- Choisissez `From text (base)` puis `mini_db`.
- On obtient une visualisation des données brutes et de leur interprétation par R. Il faut surtout vérifier

  Separator que R séparent les colonnes au bon endroit
  Heading que les noms de variables sont bien lus
  Decimal vérifiez si c'est la notation anglaise 12.1 ou française 12,1

Le module d'importation ne fait que générer un code R. Une fois que ça marche, on copie-colle ce code dans le script pour le garder.

```
mini_db <- read.csv("C:/Users/chemin/Techniques quanti/mini_db.txt", sep=";")
```

Commandes de base

View permet d'afficher la base. A ne pas utiliser si c'est une grande base.

head donne les premières données de la base. Pratique si c'est une grande base.

rm permet de jeter à la corbeille.

names donne les noms des variables.

setwd permet de définir le dossier de travail

On va donc coller tout en haut du script le chemin issu de l'importation :

```
setwd("C:/Users/chemin/Techniques quanti")
```

et simplifier l'importation :

```
mini_db <- read.csv("./mini_db.txt", sep=";")
```

Des données bien rangées

Tidyverse est une suite de packages qui servent à gérer et visualiser des données.
Il faut l'installer (une seule fois) et l'appeler (à chaque redémarrage de R)

```
install.packages("tidyverse")
library(tidyverse)
```

Les 3 commandements de tidyverse :

- chaque variable est une colonne,
- chaque observation est une ligne,
- une table différente pour chaque situation.

Des données bien rangées

Une base de données de tidyverse s'appelle un tibble.

```
pop<-as_tibble(population)
```

On importe des données directement en tibble avec

```
read_delim("population.csv",delim=";")
```

Verbes

`arrange` permet de classer les lignes

`slice` permet de choisir des lignes.

`filter` permet de choisir des lignes suivant un test.

`select` permet de choisir des variables.

`rename` permet de renommer une variable.

`mutate` permet de créer des variables et faire des opérations ligne à ligne

```
arrange(pop,age)
slice(pop,1:5) ; filter(pop,age>25)
select(pop,age) ; select(pop,-age) ;
rename(pop,Revenu=Income)
mutate(pop,Revenu_Annuel=Revenu*12)
```

Fonctions

`start_with` permet de choisir un ensemble de variables similaires, par ex. `Revenu_janvier, Revenu_fevrier...`

`end_with` idem suivant la fin.

`case_when` permet de définir une variable suivant les cas.

```
select(pop,start_with{Revenu_})
mutate(pop,Génération=case_when(age<25~jeune,
           age>=25 & age<65~ actif, age>=65~vieux))
```

Le pipe

Le pipe permet d'enchaîner des opérations sur une même base. On n'a pas besoin de rappeler à chaque verbe sur quelle base on travaille. On passe d'une opération à l'autre avec `%>%`

```
pop2<-population %>% select(-Gender)%>%
arrange(age)%>% slice(1:5)
```

produira une base pop2 avec 5 lignes.

group_by

Le verbe group_by permet de constituer des groupes dans l'échantillon et de faire une opération dans chaque groupe, comme calculer une moyenne sur les individus du groupe.

```
pop<-pop %>% group_by(Génération) %>%
mutate(Revenu_par_gen=mean(Revenu)) %>% ungroup()
```

ggplot

ggplot permet de faire de sublimes graphiques On appelle ggplot avec les options principales du graphique, puis on ajoute des éléments au graphique avec `+`

`geom_point` permet de dessiner des points

`geom_line` permet de dessiner des lignes.

`geom_smooth` permet de lisser des données.

Chacune des ces fonctions peut prendre comme option la base (data=), les variables (x= , y=), la couleur (color=" "), la taille (size=)... Il faut mettre les variables dans une fonction `aes` pour *aesthetics*