

TP_2

ZINABA_Albert

2025-01-23

Contents

| | |
|--|----------|
| I-Telechargements des packages et importation des bases de données | 1 |
| I-1-Telechargement des packages | 1 |
| I-2 Importation de la base Individu | 1 |
| I-3-Importation de la base Menage | 1 |
| I-4 Structure des deux bases | 2 |
| II-Jointure des deux bases | 2 |
| II-1 Creation d'une nouvelle base de données avec un identifiant unique pour chaque individu . . . | 2 |
| II-2 Filtrage de données par l'identifiant de l'individu | 2 |

I-Telechargements des packages et importation des bases de données

I-1-Telechargement des packages

```
library(utils)
library(dplyr)
```

I-2 Importation de la base Individu

```
Individu<- utils::read.csv("../Données/ehcvm_individu_bfa2021.csv")
Individu
```

#La syntaxe ci-dessus permet a travers la fonction 'read.csv()' permet de charger #le fichier 'ehcvm_in

I-3-Importation de la base Menage

```
Menage<- utils::read.csv("../Données/ehcvm_menage_bfa2021.csv")
Menage
```

#Cette syntaxe aussi permet de charger le #fichier 'ehcvm_menage_bfa2021.csv' dans un objet 'Menage' so

I-4 Structure des deux bases

Nous allons utiliser la fonction str du package utils pour afficher la structure des deux bases.

```
utils::str(Individu)
utils::str(Menage)
```

II-Jointure des deux bases

Ici nous utilisons la fonction left_join du package dplyr pour fusionner les deux bases. On obtient une nouvelle base de données tout en gardant les identifiants des individus.

```
Base_merge<- dplyr:: left_join(Individu,Menage,by="hhid")
Base_merge
```

II-1 Creation d’une nouvelle base de données avec un identifiant unique pour chaque individu

Cette nouvelle base a été créée en ajoutant la variable “identifiant” créée par combinaison de deux variables a savoir hhid et pid, issus de la base “base_merge”. Cette nouvelle base nommée Base_Id contient a present 96 variables.

```
Base_Id <- Base_merge %>%
  dplyr:: mutate(identifiant = hhid * 100 + pid) %>%
  select(country.x,year.x,vague.x, hhid, grappe.x, menage.x, pid, identifiant, everything()) %>%
  arrange(identifiant)
Base_Id
```

II-2 Filtrage de données par l’identifiant de l’individu

A travers la fonction filter de dplyr nous filtrons certaines informations d’un individu dans la Base_Id en insérant comme argument son identifiant créée. Par exemple avec la syntaxe suivante on filtre les informations de l’individu “58600507” comme son sexe, son age, son logement et lien avec le chef de menage, sa religion, sa region entre autres.

```
Donnee_filtre <- Base_Id %>%
  dplyr:: filter(identifiant == "58600507") %>%
  select(sexe, age, logem, lien, religion, region)
Donnee_filtre
```