DEVOIR R

Cheikh Oumar DIALLO

2025-02-25

#importation des packages

library(haven)  
library(dplyr)

##   
## Attachement du package : 'dplyr'

## Les objets suivants sont masqués depuis 'package:stats':  
##   
## filter, lag

## Les objets suivants sont masqués depuis 'package:base':  
##   
## intersect, setdiff, setequal, union

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ forcats 1.0.0 ✔ readr 2.1.5  
## ✔ ggplot2 3.5.1 ✔ stringr 1.5.1  
## ✔ lubridate 1.9.4 ✔ tibble 3.2.1  
## ✔ purrr 1.0.2 ✔ tidyr 1.3.1

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(gtsummary)

# chargement des bases

menage<- haven::read\_dta("../data/base\_menage.dta")  
fbschil<- haven::read\_dta("../data/food\_comp\_child\_baseline.dta")  
fendchil<- haven::read\_dta("../data/food\_comp\_child\_endline.dta")  
fbasemth<- haven::read\_dta("../data/food\_comp\_mother\_baseline.dta")  
fdmth<- haven::read\_dta("../data/food\_comp\_mother\_endline.dta")

#verification des doublons dans la base baseline

# Identifier les lignes dupliquées  
doublons <- fbschil %>%  
 group\_by\_all() %>%  
 filter(n() > 1) %>%  
 ungroup()  
  
# Afficher les doublons  
doublons

## # A tibble: 0 × 17  
## # ℹ 17 variables: regionid <dbl>, communeid <dbl>, villageid <dbl>, hhid <chr>,  
## # round <dbl+lbl>, s1\_q0 <dbl+lbl>, s1\_q1 <dbl+lbl>, s1\_q2 <dbl+lbl>,  
## # energ\_kcal <dbl>, protein\_g <dbl>, lipid\_tot\_g <dbl>, calcium\_mg <dbl>,  
## # iron\_mg <dbl>, zinc\_mg <dbl>, vit\_b6\_mg <dbl>, vit\_b12\_mcg <dbl>,  
## # vit\_c\_mg <dbl>

#verification des doublons dans la base endeline

# Identifier les lignes dupliquées  
doublons <-fendchil %>%  
 group\_by\_all() %>%  
 filter(n() > 1) %>%  
 ungroup()  
  
# Afficher les doublons  
doublons

## # A tibble: 0 × 17  
## # ℹ 17 variables: regionid <dbl>, communeid <dbl>, villageid <dbl>, hhid <chr>,  
## # round <dbl+lbl>, s1\_q0 <dbl+lbl>, s1\_q1 <dbl+lbl>, s1\_q2 <dbl+lbl>,  
## # energ\_kcal <dbl>, protein\_g <dbl>, lipid\_tot\_g <dbl>, calcium\_mg <dbl>,  
## # iron\_mg <dbl>, zinc\_mg <dbl>, vit\_b6\_mg <dbl>, vit\_b12\_mcg <dbl>,  
## # vit\_c\_mg <dbl>

#verification des doublons dans la base menage

# Identifier les lignes dupliquées  
doublons <-menage %>%  
 group\_by\_all() %>%  
 filter(n() > 1) %>%  
 ungroup()  
  
# Afficher les doublons  
doublons

## # A tibble: 2 × 21  
## regionid communeid villageid hhid hhsize poly hh\_primary s1\_q2 s1\_q4a  
## <dbl> <dbl> <dbl> <chr> <dbl> <dbl+l> <dbl+lbl> <dbl+l> <dbl>  
## 1 1 1 15 4953484… 6 0 [Non] 0 [Non] 1 [Mal… 65  
## 2 1 1 15 4953484… 6 0 [Non] 0 [Non] 1 [Mal… 65  
## # ℹ 12 more variables: s2\_q1 <dbl+lbl>, s2\_q2 <dbl+lbl>, s2\_q4 <dbl+lbl>,  
## # s29\_q1 <dbl+lbl>, demgrp1 <dbl>, demgrp2 <dbl>, demgrp3 <dbl>,  
## # demgrp4 <dbl>, demgrp5 <dbl>, dependencyratio <dbl>, hfias\_score <dbl>,  
## # T1 <dbl+lbl>

#suppression des doublons dans la Base ménage

# Lire le fichier .dta  
menage <-haven::read\_dta("../data/base\_menage.dta")  
  
# Supprimer les doublons  
menage\_sans\_doublons <- distinct(menage)  
  
# Sauvegarder la base sans doublons  
write\_dta(menage\_sans\_doublons, "menage\_sans\_doublons.dta")  
  
# Confirmation  
cat("menage sans doublons enregistrée sous 'menage\_sans\_doublons.dta'")

## menage sans doublons enregistrée sous 'menage\_sans\_doublons.dta'

#comparaison des noms des variables entre baseline et Endline

# Lire les deux bases .dta  
fbschil<- haven::read\_dta("../data/food\_comp\_child\_baseline.dta")  
fendchil<- haven::read\_dta("../data/food\_comp\_child\_endline.dta")  
  
# Comparer les noms des variables  
if (setequal(names(fbschil), names(fendchil))) {  
 cat("Les noms des variables sont identiques.\n")  
} else {  
 cat("Les noms des variables ne sont pas identiques.\n")  
  
 # Afficher les variables différentes  
 cat("Variables présentes uniquement dans fbschil :\n")  
 print(setdiff(names(fbschil), names(fendchil)))  
  
 cat("Variables présentes uniquement dans fendchil :\n")  
 print(setdiff(names(fendchil), names(fbschil)))  
  
 # Harmoniser les colonnes (ajouter les colonnes manquantes avec NA)  
 for (var in setdiff(names(fbschil), names(fendchil))) {  
 fendchil[[var]] <- NA  
 }  
 for (var in setdiff(names(fendchil), names(fbschil))) {  
 fbschil[[var]] <- NA  
 }  
  
 # Réordonner les colonnes pour qu'elles soient dans le même ordre  
 fendchil <- fendchil[, names(fbschil)]  
  
 cat("Les noms des variables ont été harmonisés.\n")  
}

## Les noms des variables sont identiques.

# calcul de la consommation moyenne

# Calcul de la consommation d'énergie moyenne  
conso\_energie\_moyenne <- fbasemth %>%  
 summarise(energie\_moyenne = mean(protein\_g, na.rm = TRUE))  
# Affichage du résultat  
print(conso\_energie\_moyenne)

## # A tibble: 1 × 1  
## energie\_moyenne  
## <dbl>  
## 1 22.1

## Exploration des données

# Afficher les noms des variables  
colnames(fbasemth)

## [1] "regionid" "communeid" "villageid" "hhid" "round"   
## [6] "s1\_q0" "s1\_q1" "s1\_q2" "V1" "protein\_g"   
## [11] "lipid\_tot\_g" "calcium\_mg" "iron\_mg" "V9" "vit\_b6\_mg"   
## [16] "vit\_b12\_mcg" "vit\_c\_mg"

# Aperçu des premières lignes  
tibble::glimpse(fbasemth)

## Rows: 4,256  
## Columns: 17  
## $ regionid <dbl> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2…  
## $ communeid <dbl> 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25…  
## $ villageid <dbl> 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000, 1000…  
## $ hhid <chr> "4948484848535052", "4948484848535052", "4948484848535052"…  
## $ round <dbl+lbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, …  
## $ s1\_q0 <dbl+lbl> 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, …  
## $ s1\_q1 <dbl+lbl> 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, …  
## $ s1\_q2 <dbl+lbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, …  
## $ V1 <dbl> 680.4425, NA, 634.2752, 255.7698, 562.8450, NA, NA, NA, 23…  
## $ protein\_g <dbl> 23.313662, NA, 21.622943, 8.252958, 11.696869, NA, NA, NA,…  
## $ lipid\_tot\_g <dbl> 5.301461, NA, 4.872112, 8.061300, 6.486809, NA, NA, NA, 1.…  
## $ calcium\_mg <dbl> 62.73322, NA, 57.09909, 22.04590, 116.28838, NA, NA, NA, 1…  
## $ iron\_mg <dbl> 10.590628, NA, 9.896879, 0.912000, 2.716486, NA, NA, NA, 3…  
## $ V9 <dbl> 4.5072635, NA, 4.1896121, 0.4560000, 3.3820429, NA, NA, NA…  
## $ vit\_b6\_mg <dbl> 0.30578630, NA, 0.28354130, 0.04560000, 0.22481194, NA, NA…  
## $ vit\_b12\_mcg <dbl> 0.0086916752, NA, 0.0077976741, 0.0000000000, 0.0082258058…  
## $ vit\_c\_mg <dbl> 4.414822e-02, NA, 3.960726e-02, 0.000000e+00, 2.000516e-04…

## Calcul de la consommation d’énergie moyenne par repas

# Calcul de la consommation d'énergie moyenne  
conso\_energie\_moyenne <- fbasemth %>%  
 summarise(energie\_moyenne = mean(protein\_g, na.rm = TRUE))  
  
# Affichage du résultat  
print(conso\_energie\_moyenne)

## # A tibble: 1 × 1  
## energie\_moyenne  
## <dbl>  
## 1 22.1

#Sauvegarde des nouvelles bases

# i) Sauvegarder les changements pour food\_comp\_mother\_baseline.dta  
write\_dta(fbasemth, "mother\_baseline\_v1.dta")  
  
# ii) Sauvegarder les changements pour food\_comp\_mother\_endline.dta  
write\_dta(fdmth, "mother\_endline\_v1.dta")  
  
# iii) Sauvegarder les changements pour food\_comp\_child\_baseline.dta  
write\_dta(fbschil, "child\_baseline\_v1.dta")  
  
# iv) Sauvegarder les changements pour food\_comp\_child\_endline.dta  
write\_dta(fendchil, "child\_endline\_v1.dta")  
  
# v) Sauvegarder les changements pour base\_menage.dta  
write\_dta(menage\_sans\_doublons, "base\_menage\_final.dta")

#Fusion des basées

#baseline  
# Jointure complète entre les bases des mères et des enfants sur l'identifiant du ménage  
bfinal <- full\_join(fbschil, fbschil, by = "hhid")

## Warning in full\_join(fbschil, fbschil, by = "hhid"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 1 of `x` matches multiple rows in `y`.  
## ℹ Row 1 of `y` matches multiple rows in `x`.  
## ℹ If a many-to-many relationship is expected, set `relationship =  
## "many-to-many"` to silence this warning.

# Aperçu du résultat  
head(bfinal)

## # A tibble: 6 × 33  
## regionid.x communeid.x villageid.x hhid round.x s1\_q0.x s1\_q1.x s1\_q2.x  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+l> <dbl+l>  
## 1 2 25 1000 4948484848… 1 [Bas… 1 [Bre… 1 [Yes] 2 [chi…  
## 2 2 25 1000 4948484848… 1 [Bas… 1 [Bre… 1 [Yes] 2 [chi…  
## 3 2 25 1000 4948484848… 1 [Bas… 1 [Bre… 1 [Yes] 2 [chi…  
## 4 2 25 1000 4948484848… 1 [Bas… 1 [Bre… 1 [Yes] 2 [chi…  
## 5 2 25 1000 4948484848… 1 [Bas… 2 [Lun… 1 [Yes] 2 [chi…  
## 6 2 25 1000 4948484848… 1 [Bas… 2 [Lun… 1 [Yes] 2 [chi…  
## # ℹ 25 more variables: energ\_kcal.x <dbl>, protein\_g.x <dbl>,  
## # lipid\_tot\_g.x <dbl>, calcium\_mg.x <dbl>, iron\_mg.x <dbl>, zinc\_mg.x <dbl>,  
## # vit\_b6\_mg.x <dbl>, vit\_b12\_mcg.x <dbl>, vit\_c\_mg.x <dbl>, regionid.y <dbl>,  
## # communeid.y <dbl>, villageid.y <dbl>, round.y <dbl+lbl>, s1\_q0.y <dbl+lbl>,  
## # s1\_q1.y <dbl+lbl>, s1\_q2.y <dbl+lbl>, energ\_kcal.y <dbl>,  
## # protein\_g.y <dbl>, lipid\_tot\_g.y <dbl>, calcium\_mg.y <dbl>,  
## # iron\_mg.y <dbl>, zinc\_mg.y <dbl>, vit\_b6\_mg.y <dbl>, vit\_b12\_mcg.y <dbl>, …

#renommer les variables

# Renommer toutes les colonnes de consommation en ajoutant le suffixe \_b  
bfinal <- bfinal %>%  
 rename\_with(~ paste0(., "\_b"), .cols = matches("energ\_kcal|protein\_g|lipid\_tot\_g|calcium\_mg|iron\_mg|vit\_b6\_mg|vit\_b12\_mcg|vit\_c\_mcg"))  
  
# Vérifier les nouveaux noms des colonnes  
colnames(bfinal)

## [1] "regionid.x" "communeid.x" "villageid.x" "hhid"   
## [5] "round.x" "s1\_q0.x" "s1\_q1.x" "s1\_q2.x"   
## [9] "energ\_kcal.x\_b" "protein\_g.x\_b" "lipid\_tot\_g.x\_b" "calcium\_mg.x\_b"   
## [13] "iron\_mg.x\_b" "zinc\_mg.x" "vit\_b6\_mg.x\_b" "vit\_b12\_mcg.x\_b"  
## [17] "vit\_c\_mg.x" "regionid.y" "communeid.y" "villageid.y"   
## [21] "round.y" "s1\_q0.y" "s1\_q1.y" "s1\_q2.y"   
## [25] "energ\_kcal.y\_b" "protein\_g.y\_b" "lipid\_tot\_g.y\_b" "calcium\_mg.y\_b"   
## [29] "iron\_mg.y\_b" "zinc\_mg.y" "vit\_b6\_mg.y\_b" "vit\_b12\_mcg.y\_b"  
## [33] "vit\_c\_mg.y"

# Base endline

#basendeline  
# Jointure complète entre les bases des mères et des enfants sur l'identifiant du ménage  
bfinal2 <- full\_join(fdmth, fendchil, by = "hhid")

## Warning in full\_join(fdmth, fendchil, by = "hhid"): Detected an unexpected many-to-many relationship between `x` and `y`.  
## ℹ Row 1 of `x` matches multiple rows in `y`.  
## ℹ Row 1 of `y` matches multiple rows in `x`.  
## ℹ If a many-to-many relationship is expected, set `relationship =  
## "many-to-many"` to silence this warning.

# Aperçu du résultat  
head(bfinal2)

## # A tibble: 6 × 33  
## regionid.x communeid.x villageid.x hhid round.x s1\_q0.x s1\_q1.x s1\_q2.x  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+l> <dbl+l>  
## 1 2 25 1000 4948484848… 2 [End… 1 [Bre… 1 [Yes] 1 [mot…  
## 2 2 25 1000 4948484848… 2 [End… 1 [Bre… 1 [Yes] 1 [mot…  
## 3 2 25 1000 4948484848… 2 [End… 1 [Bre… 1 [Yes] 1 [mot…  
## 4 2 25 1000 4948484848… 2 [End… 1 [Bre… 1 [Yes] 1 [mot…  
## 5 2 25 1000 4948484848… 2 [End… 2 [Lun… 1 [Yes] 1 [mot…  
## 6 2 25 1000 4948484848… 2 [End… 2 [Lun… 1 [Yes] 1 [mot…  
## # ℹ 25 more variables: energ\_kcal.x <dbl>, protein\_g.x <dbl>,  
## # lipid\_tot\_g.x <dbl>, calcium\_mg.x <dbl>, iron\_mg.x <dbl>, zinc\_mg.x <dbl>,  
## # vit\_b6\_mg.x <dbl>, vit\_b12\_mcg.x <dbl>, vit\_c\_mg.x <dbl>, regionid.y <dbl>,  
## # communeid.y <dbl>, villageid.y <dbl>, round.y <dbl+lbl>, s1\_q0.y <dbl+lbl>,  
## # s1\_q1.y <dbl+lbl>, s1\_q2.y <dbl+lbl>, energ\_kcal.y <dbl>,  
## # protein\_g.y <dbl>, lipid\_tot\_g.y <dbl>, calcium\_mg.y <dbl>,  
## # iron\_mg.y <dbl>, zinc\_mg.y <dbl>, vit\_b6\_mg.y <dbl>, vit\_b12\_mcg.y <dbl>, …

#renomage des variables

# Renommer toutes les colonnes de consommation en ajoutant le suffixe \_e  
bfinal2 <- bfinal2 %>%  
 rename\_with(~ paste0(., "\_e"), .cols = matches("energ\_kcal|protein\_g|lipid\_tot\_g|calcium\_mg|iron\_mg|vit\_b6\_mg|vit\_b12\_mcg|vit\_c\_mcg"))  
  
# Vérifier les nouveaux noms des colonnes  
colnames(bfinal2)

## [1] "regionid.x" "communeid.x" "villageid.x" "hhid"   
## [5] "round.x" "s1\_q0.x" "s1\_q1.x" "s1\_q2.x"   
## [9] "energ\_kcal.x\_e" "protein\_g.x\_e" "lipid\_tot\_g.x\_e" "calcium\_mg.x\_e"   
## [13] "iron\_mg.x\_e" "zinc\_mg.x" "vit\_b6\_mg.x\_e" "vit\_b12\_mcg.x\_e"  
## [17] "vit\_c\_mg.x" "regionid.y" "communeid.y" "villageid.y"   
## [21] "round.y" "s1\_q0.y" "s1\_q1.y" "s1\_q2.y"   
## [25] "energ\_kcal.y\_e" "protein\_g.y\_e" "lipid\_tot\_g.y\_e" "calcium\_mg.y\_e"   
## [29] "iron\_mg.y\_e" "zinc\_mg.y" "vit\_b6\_mg.y\_e" "vit\_b12\_mcg.y\_e"  
## [33] "vit\_c\_mg.y"