Devoir de R SOMA Ben Idriss

2025-02-25

Commencons par importer les librairies nécessaires pour notre travail

# importation des librairies   
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(gtsummary)

# Partie 1: Gestion et nettoyage des bases de données

child\_baseline <- haven::read\_dta("../Données/food\_comp\_child\_baseline.dta")  
  
child\_endline <- haven::read\_dta("../Données/food\_comp\_child\_endline.dta")  
  
mother\_baseline <- haven::read\_dta("../Données/food\_comp\_mother\_baseline.dta")  
  
mother\_endline <- haven::read\_dta("../Données/food\_comp\_mother\_endline.dta")  
  
menage <- haven::read\_dta("../Données/food\_comp\_child\_baseline.dta")

### organisation et structure des différentes bases

print(child\_baseline)

## # A tibble: 4,256 × 17  
## regionid communeid villageid hhid round s1\_q0 s1\_q1 s1\_q2 energ\_kcal  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl>  
## 1 2 25 1000 4948… 1 [Bas… 1 [Bre… 1 [Yes] 2 [chi… 355.  
## 2 2 25 1000 4948… 1 [Bas… 2 [Lun… 1 [Yes] 2 [chi… 224.  
## 3 2 25 1000 4948… 1 [Bas… 3 [Din… 1 [Yes] 2 [chi… 334.  
## 4 2 25 1000 4948… 1 [Bas… 4 [Sna… 1 [Yes] 2 [chi… 494.  
## 5 2 25 1000 4948… 1 [Bas… 1 [Bre… 1 [Yes] 2 [chi… 235.  
## 6 2 25 1000 4948… 1 [Bas… 2 [Lun… 1 [Yes] 2 [chi… 216.  
## 7 2 25 1000 4948… 1 [Bas… 3 [Din… 1 [Yes] 2 [chi… 142.  
## 8 2 25 1000 4948… 1 [Bas… 4 [Sna… 1 [Yes] 2 [chi… 406.  
## 9 2 25 1000 4948… 1 [Bas… 1 [Bre… 1 [Yes] 2 [chi… 165.  
## 10 2 25 1000 4948… 1 [Bas… 2 [Lun… 1 [Yes] 2 [chi… 157.  
## # ℹ 4,246 more rows  
## # ℹ 8 more variables: 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>

str(child\_baseline)

## tibble [4,256 × 17] (S3: tbl\_df/tbl/data.frame)  
## $ regionid : num [1:4256] 2 2 2 2 2 2 2 2 2 2 ...  
## ..- attr(\*, "label")= chr "Region ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ communeid : num [1:4256] 25 25 25 25 25 25 25 25 25 25 ...  
## ..- attr(\*, "label")= chr "Commune ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ villageid : num [1:4256] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...  
## ..- attr(\*, "label")= chr "Village ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ hhid : chr [1:4256] "4948484848535052" "4948484848535052" "4948484848535052" "4948484848535052" ...  
## ..- attr(\*, "label")= chr "Household ID"  
## ..- attr(\*, "format.stata")= chr "%45s"  
## $ round : dbl+lbl [1:4256] 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Survey round : Baseline, Endline"  
## ..@ format.stata: chr "%10.0g"  
## ..@ labels : Named num [1:2] 1 2  
## .. ..- attr(\*, "names")= chr [1:2] "Baseline" "Endline"  
## $ s1\_q0 : dbl+lbl [1:4256] 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2,...  
## ..@ label : chr "eating occasion"  
## ..@ format.stata: chr "%27.0g"  
## ..@ labels : Named num [1:4] 1 2 3 4  
## .. ..- attr(\*, "names")= chr [1:4] "Breakfast" "Lunch" "Dinner" "Snacks"  
## $ s1\_q1 : dbl+lbl [1:4256] 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Meal consumed? Y/N"  
## ..@ format.stata: chr "%9.0g"  
## ..@ labels : Named num [1:2] 0 1  
## .. ..- attr(\*, "names")= chr [1:2] "No" "Yes"  
## $ s1\_q2 : dbl+lbl [1:4256] 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,...  
## ..@ label : chr "Who consummed?"  
## ..@ format.stata: chr "%16.0g"  
## ..@ labels : Named num [1:3] 1 2 3  
## .. ..- attr(\*, "names")= chr [1:3] "mother" "child" "mother and child"  
## $ energ\_kcal : num [1:4256] 355 224 334 494 235 ...  
## ..- attr(\*, "label")= chr "Consommation en ernergie (kcal)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ protein\_g : num [1:4256] 12.74 8.01 12.72 16.65 4.45 ...  
## ..- attr(\*, "label")= chr "Consommation en proteine (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ lipid\_tot\_g: num [1:4256] 3.14 1.96 3.42 8.84 4.76 ...  
## ..- attr(\*, "label")= chr "Consommation en lipide (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ calcium\_mg : num [1:4256] 40.1 24.8 47 60.1 93.3 ...  
## ..- attr(\*, "label")= chr "Consommation en calcium (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ iron\_mg : num [1:4256] 5.39 3.42 4.9 6.83 1.54 ...  
## ..- attr(\*, "label")= chr "Consommation en fer (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ zinc\_mg : num [1:4256] 2.41 1.52 2.35 3.13 1.63 ...  
## ..- attr(\*, "label")= chr "Consommation en zinc (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b6\_mg : num [1:4256] 0.167 0.105 0.168 0.207 0.116 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B6 (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b12\_mcg: num [1:4256] 0.00616 0.00377 0.00785 0.00959 0.00688 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B12 (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_c\_mg : num [1:4256] 0.031282 0.019173 0.03986 0.048689 0.000167 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine C (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"

print(child\_endline)

## # A tibble: 4,256 × 17  
## regionid communeid villageid hhid round s1\_q0 s1\_q1 s1\_q2 energ\_kcal  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl>  
## 1 2 25 1000 4948… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 1193.  
## 2 2 25 1000 4948… 2 [End… 2 [Lun… 1 [Yes] 2 [chi… 535.  
## 3 2 25 1000 4948… 2 [End… 3 [Din… 1 [Yes] 2 [chi… 1701.  
## 4 2 25 1000 4948… 2 [End… 4 [Sna… 1 [Yes] 2 [chi… 531.  
## 5 2 25 1000 4948… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 188.  
## 6 2 25 1000 4948… 2 [End… 2 [Lun… 0 [No] 2 [chi… NA   
## 7 2 25 1000 4948… 2 [End… 3 [Din… 1 [Yes] 2 [chi… 176.  
## 8 2 25 1000 4948… 2 [End… 4 [Sna… 0 [No] 2 [chi… NA   
## 9 2 25 1000 4948… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 1304.  
## 10 2 25 1000 4948… 2 [End… 2 [Lun… 1 [Yes] 2 [chi… 336.  
## # ℹ 4,246 more rows  
## # ℹ 8 more variables: 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>

str(child\_endline)

## tibble [4,256 × 17] (S3: tbl\_df/tbl/data.frame)  
## $ regionid : num [1:4256] 2 2 2 2 2 2 2 2 2 2 ...  
## ..- attr(\*, "label")= chr "Region ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ communeid : num [1:4256] 25 25 25 25 25 25 25 25 25 25 ...  
## ..- attr(\*, "label")= chr "Commune ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ villageid : num [1:4256] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...  
## ..- attr(\*, "label")= chr "Village ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ hhid : chr [1:4256] "4948484848535052" "4948484848535052" "4948484848535052" "4948484848535052" ...  
## ..- attr(\*, "label")= chr "Household ID"  
## ..- attr(\*, "format.stata")= chr "%45s"  
## $ round : dbl+lbl [1:4256] 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,...  
## ..@ label : chr "Survey round : Baseline, Endline"  
## ..@ format.stata: chr "%10.0g"  
## ..@ labels : Named num [1:2] 1 2  
## .. ..- attr(\*, "names")= chr [1:2] "Baseline" "Endline"  
## $ s1\_q0 : dbl+lbl [1:4256] 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2,...  
## ..@ label : chr "eating occasion"  
## ..@ format.stata: chr "%27.0g"  
## ..@ labels : Named num [1:4] 1 2 3 4  
## .. ..- attr(\*, "names")= chr [1:4] "Breakfast" "Lunch" "Dinner" "Snacks"  
## $ s1\_q1 : dbl+lbl [1:4256] 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Meal consumed? Y/N"  
## ..@ format.stata: chr "%9.0g"  
## ..@ labels : Named num [1:2] 0 1  
## .. ..- attr(\*, "names")= chr [1:2] "No" "Yes"  
## $ s1\_q2 : dbl+lbl [1:4256] 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,...  
## ..@ label : chr "Who consummed?"  
## ..@ format.stata: chr "%16.0g"  
## ..@ labels : Named num [1:3] 1 2 3  
## .. ..- attr(\*, "names")= chr [1:3] "mother" "child" "mother and child"  
## $ energ\_kcal : num [1:4256] 1193 535 1701 531 188 ...  
## ..- attr(\*, "label")= chr "Consommation en ernergie (kcal)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ protein\_g : num [1:4256] 24.94 18.38 58.84 13.89 5.04 ...  
## ..- attr(\*, "label")= chr "Consommation en proteine (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ lipid\_tot\_g: num [1:4256] 51.915 3.205 90.268 3.917 0.496 ...  
## ..- attr(\*, "label")= chr "Consommation en lipide (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ calcium\_mg : num [1:4256] 479.73 41.32 1122.63 74.44 3.29 ...  
## ..- attr(\*, "label")= chr "Consommation en calcium (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ iron\_mg : num [1:4256] 11.5857 8.709 7.2277 6.4328 0.0321 ...  
## ..- attr(\*, "label")= chr "Consommation en fer (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ zinc\_mg : num [1:4256] 4.9181 3.3562 1.4184 2.7147 0.0102 ...  
## ..- attr(\*, "label")= chr "Consommation en zinc (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b6\_mg : num [1:4256] 0.4678 0.2251 0.2028 0.3523 0.0018 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B6 (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b12\_mcg: num [1:4256] 0.00396 0.01364 0 0.00736 0.00309 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B12 (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_c\_mg : num [1:4256] 6.8299 0.1201 16.904 0.0764 0 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine C (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"

print(mother\_baseline)

## # A tibble: 4,256 × 17  
## regionid communeid villageid hhid round s1\_q0 s1\_q1 s1\_q2 V1  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl>  
## 1 2 25 1000 494848484… 1 [Bas… 1 [Bre… 1 [Yes] 1 [mot… 680.  
## 2 2 25 1000 494848484… 1 [Bas… 2 [Lun… 0 [No] 1 [mot… NA   
## 3 2 25 1000 494848484… 1 [Bas… 3 [Din… 1 [Yes] 1 [mot… 634.  
## 4 2 25 1000 494848484… 1 [Bas… 4 [Sna… 1 [Yes] 1 [mot… 256.  
## 5 2 25 1000 494848484… 1 [Bas… 1 [Bre… 1 [Yes] 1 [mot… 563.  
## 6 2 25 1000 494848484… 1 [Bas… 2 [Lun… 0 [No] 1 [mot… NA   
## 7 2 25 1000 494848484… 1 [Bas… 3 [Din… 0 [No] 1 [mot… NA   
## 8 2 25 1000 494848484… 1 [Bas… 4 [Sna… 0 [No] 1 [mot… NA   
## 9 2 25 1000 494848484… 1 [Bas… 1 [Bre… 1 [Yes] 1 [mot… 237.  
## 10 2 25 1000 494848484… 1 [Bas… 2 [Lun… 1 [Yes] 1 [mot… 553.  
## # ℹ 4,246 more rows  
## # ℹ 8 more variables: protein\_g <dbl>, lipid\_tot\_g <dbl>, calcium\_mg <dbl>,  
## # iron\_mg <dbl>, V9 <dbl>, vit\_b6\_mg <dbl>, vit\_b12\_mcg <dbl>, vit\_c\_mg <dbl>

str(mother\_baseline)

## tibble [4,256 × 17] (S3: tbl\_df/tbl/data.frame)  
## $ regionid : num [1:4256] 2 2 2 2 2 2 2 2 2 2 ...  
## ..- attr(\*, "label")= chr "Region ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ communeid : num [1:4256] 25 25 25 25 25 25 25 25 25 25 ...  
## ..- attr(\*, "label")= chr "Commune ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ villageid : num [1:4256] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...  
## ..- attr(\*, "label")= chr "Village ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ hhid : chr [1:4256] "4948484848535052" "4948484848535052" "4948484848535052" "4948484848535052" ...  
## ..- attr(\*, "label")= chr "Household ID"  
## ..- attr(\*, "format.stata")= chr "%45s"  
## $ round : dbl+lbl [1:4256] 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Survey round : Baseline, Endline"  
## ..@ format.stata: chr "%10.0g"  
## ..@ labels : Named num [1:2] 1 2  
## .. ..- attr(\*, "names")= chr [1:2] "Baseline" "Endline"  
## $ s1\_q0 : dbl+lbl [1:4256] 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2,...  
## ..@ label : chr "eating occasion"  
## ..@ format.stata: chr "%27.0g"  
## ..@ labels : Named num [1:4] 1 2 3 4  
## .. ..- attr(\*, "names")= chr [1:4] "Breakfast" "Lunch" "Dinner" "Snacks"  
## $ s1\_q1 : dbl+lbl [1:4256] 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Meal consumed? Y/N"  
## ..@ format.stata: chr "%9.0g"  
## ..@ labels : Named num [1:2] 0 1  
## .. ..- attr(\*, "names")= chr [1:2] "No" "Yes"  
## $ s1\_q2 : dbl+lbl [1:4256] 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Who consummed?"  
## ..@ format.stata: chr "%16.0g"  
## ..@ labels : Named num [1:3] 1 2 3  
## .. ..- attr(\*, "names")= chr [1:3] "mother" "child" "mother and child"  
## $ V1 : num [1:4256] 680 NA 634 256 563 ...  
## ..- attr(\*, "label")= chr "Consommation en ernergie (kcal)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ protein\_g : num [1:4256] 23.31 NA 21.62 8.25 11.7 ...  
## ..- attr(\*, "label")= chr "Consommation en proteine (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ lipid\_tot\_g: num [1:4256] 5.3 NA 4.87 8.06 6.49 ...  
## ..- attr(\*, "label")= chr "Consommation en lipide (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ calcium\_mg : num [1:4256] 62.7 NA 57.1 22 116.3 ...  
## ..- attr(\*, "label")= chr "Consommation en calcium (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ iron\_mg : num [1:4256] 10.591 NA 9.897 0.912 2.716 ...  
## ..- attr(\*, "label")= chr "Consommation en fer (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ V9 : num [1:4256] 4.507 NA 4.19 0.456 3.382 ...  
## ..- attr(\*, "label")= chr "Consommation en zinc (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b6\_mg : num [1:4256] 0.3058 NA 0.2835 0.0456 0.2248 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B6 (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b12\_mcg: num [1:4256] 0.00869 NA 0.0078 0 0.00823 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B12 (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_c\_mg : num [1:4256] 0.0441 NA 0.0396 0 0.0002 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine C (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"

print(mother\_endline)

## # A tibble: 4,256 × 17  
## regionid communeid villageid hhid round s1\_q0 s1\_q1 s1\_q2 energ\_kcal  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl>  
## 1 2 25 1000 4948… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 2217.  
## 2 2 25 1000 4948… 2 [End… 2 [Lun… 1 [Yes] 1 [mot… 1021.  
## 3 2 25 1000 4948… 2 [End… 3 [Din… 1 [Yes] 1 [mot… 3038.  
## 4 2 25 1000 4948… 2 [End… 4 [Sna… 1 [Yes] 1 [mot… 716.  
## 5 2 25 1000 4948… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 618.  
## 6 2 25 1000 4948… 2 [End… 2 [Lun… 0 [No] 1 [mot… NA   
## 7 2 25 1000 4948… 2 [End… 3 [Din… 1 [Yes] 1 [mot… 675.  
## 8 2 25 1000 4948… 2 [End… 4 [Sna… 0 [No] 1 [mot… NA   
## 9 2 25 1000 4948… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 876.  
## 10 2 25 1000 4948… 2 [End… 2 [Lun… 1 [Yes] 1 [mot… 414.  
## # ℹ 4,246 more rows  
## # ℹ 8 more variables: 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>

str(mother\_endline)

## tibble [4,256 × 17] (S3: tbl\_df/tbl/data.frame)  
## $ regionid : num [1:4256] 2 2 2 2 2 2 2 2 2 2 ...  
## ..- attr(\*, "label")= chr "Region ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ communeid : num [1:4256] 25 25 25 25 25 25 25 25 25 25 ...  
## ..- attr(\*, "label")= chr "Commune ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ villageid : num [1:4256] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...  
## ..- attr(\*, "label")= chr "Village ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ hhid : chr [1:4256] "4948484848535052" "4948484848535052" "4948484848535052" "4948484848535052" ...  
## ..- attr(\*, "label")= chr "Household ID"  
## ..- attr(\*, "format.stata")= chr "%45s"  
## $ round : dbl+lbl [1:4256] 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,...  
## ..@ label : chr "Survey round : Baseline, Endline"  
## ..@ format.stata: chr "%10.0g"  
## ..@ labels : Named num [1:2] 1 2  
## .. ..- attr(\*, "names")= chr [1:2] "Baseline" "Endline"  
## $ s1\_q0 : dbl+lbl [1:4256] 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2,...  
## ..@ label : chr "eating occasion"  
## ..@ format.stata: chr "%27.0g"  
## ..@ labels : Named num [1:4] 1 2 3 4  
## .. ..- attr(\*, "names")= chr [1:4] "Breakfast" "Lunch" "Dinner" "Snacks"  
## $ s1\_q1 : dbl+lbl [1:4256] 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1,...  
## ..@ label : chr "Meal consumed? Y/N"  
## ..@ format.stata: chr "%9.0g"  
## ..@ labels : Named num [1:2] 0 1  
## .. ..- attr(\*, "names")= chr [1:2] "No" "Yes"  
## $ s1\_q2 : dbl+lbl [1:4256] 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Who consummed?"  
## ..@ format.stata: chr "%16.0g"  
## ..@ labels : Named num [1:3] 1 2 3  
## .. ..- attr(\*, "names")= chr [1:3] "mother" "child" "mother and child"  
## $ energ\_kcal : num [1:4256] 2217 1021 3038 716 618 ...  
## ..- attr(\*, "label")= chr "Consommation en ernergie (kcal)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ protein\_g : num [1:4256] 48.7 34.8 92 19.1 16.5 ...  
## ..- attr(\*, "label")= chr "Consommation en proteine (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ lipid\_tot\_g: num [1:4256] 89.6 5.94 183.6 4.73 1.63 ...  
## ..- attr(\*, "label")= chr "Consommation en lipide (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ calcium\_mg : num [1:4256] 818.8 72.1 1781.3 131.6 10.8 ...  
## ..- attr(\*, "label")= chr "Consommation en calcium (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ iron\_mg : num [1:4256] 22.659 16.558 11.345 8.877 0.105 ...  
## ..- attr(\*, "label")= chr "Consommation en fer (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ zinc\_mg : num [1:4256] 9.5691 6.4618 2.2278 3.5867 0.0334 ...  
## ..- attr(\*, "label")= chr "Consommation en zinc (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b6\_mg : num [1:4256] 0.8748 0.4327 0.314 0.5277 0.0059 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B6 (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b12\_mcg: num [1:4256] 0.00948 0.02232 0 0.01281 0.01016 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B12 (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_c\_mg : num [1:4256] 11.547 0.183 26.738 0.133 0 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine C (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"

1. Verifions la presence de doublons dans chaque base

# Fonction pour vérifier et supprimer les doublons  
remove\_duplicates <- function(data, key) {  
 if (any(duplicated(data[[key]]))) {  
 cat("Des doublons ont été trouvés dans", deparse(substitute(data)), "\n")  
 cat("Nombre de doublons:", sum(duplicated(data[[key]])), "\n")  
 data <- data %>% distinct(!!sym(key), .keep\_all = TRUE)  
 cat("Les doublons ont été supprimés.\n")  
 } else {  
 cat("Aucun doublon trouvé dans", deparse(substitute(data)), "\n")  
 }  
 return(data)  
}  
  
# Vérifier et supprimer les doublons pour chaque jeu de données  
child\_baseline <- remove\_duplicates(child\_baseline, "hhid")

## Des doublons ont été trouvés dans child\_baseline   
## Nombre de doublons: 3192   
## Les doublons ont été supprimés.

child\_endline <- remove\_duplicates(child\_endline, "hhid")

## Des doublons ont été trouvés dans child\_endline   
## Nombre de doublons: 3192   
## Les doublons ont été supprimés.

mother\_baseline <- remove\_duplicates(mother\_baseline, "hhid")

## Des doublons ont été trouvés dans mother\_baseline   
## Nombre de doublons: 3192   
## Les doublons ont été supprimés.

mother\_endline <- remove\_duplicates(mother\_endline, "hhid")

## Des doublons ont été trouvés dans mother\_endline   
## Nombre de doublons: 3192   
## Les doublons ont été supprimés.

menage <- remove\_duplicates(menage, "hhid")

## Des doublons ont été trouvés dans menage   
## Nombre de doublons: 3192   
## Les doublons ont été supprimés.

Il n’y a pas de doublons dans une des bases

print("noms de colonnes de child\_baseline" )

## [1] "noms de colonnes de child\_baseline"

colnames(child\_baseline)

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

print("noms de colonnes de child\_endlineline" )

## [1] "noms de colonnes de child\_endlineline"

colnames(child\_endline)

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

print("noms de colonnes de mother\_baseline" )

## [1] "noms de colonnes de mother\_baseline"

colnames(mother\_baseline)

## [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"

print("noms de colonnes de mother\_endline" )

## [1] "noms de colonnes de mother\_endline"

colnames(mother\_endline)

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

Verifions que toutes les bases endline et baseline ont le meme nom de colonnes

# Créer une liste des datasets  
datasets <- list(  
 child\_baseline = child\_baseline,  
 child\_endline = child\_endline,  
 mother\_baseline = mother\_baseline,  
 mother\_endline = mother\_endline,  
 menage = menage  
)  
  
# Extraire les noms de colonnes pour chaque dataset  
column\_names <- lapply(datasets, colnames)  
  
# Vérifier si tous les datasets ont les mêmes colonnes  
all\_equal <- Reduce(setequal, column\_names)  
  
if (all\_equal) {  
 cat("Toutes les bases ont les mêmes noms de colonnes.\n")  
} else {  
 cat("Les bases ont des différences dans les noms de colonnes.\n")  
}

## Les bases ont des différences dans les noms de colonnes.

### 3. Verifions si ya des données manquantes

# Créer une liste des datasets  
datasets <- list(  
 child\_baseline = child\_baseline,  
 child\_endline = child\_endline,  
 mother\_baseline = mother\_baseline,  
 mother\_endline = mother\_endline,  
 menage = menage  
)  
  
# Vérifier le nombre total de valeurs manquantes dans chaque base  
  
for (name in names(datasets)) {  
 missing\_values <- sum(is.na(datasets[[name]]))  
 cat(name, "contient", missing\_values, "valeurs manquantes.\n")  
}

## child\_baseline contient 306 valeurs manquantes.  
## child\_endline contient 486 valeurs manquantes.  
## mother\_baseline contient 615 valeurs manquantes.  
## mother\_endline contient 702 valeurs manquantes.  
## menage contient 306 valeurs manquantes.

On constate que les bases mother\_baseline et mother\_endline contiennent plus de valeurs manquantes que les autres bases

### visualisation des données manquantes

child\_baseline[!complete.cases(child\_baseline), ]

## # A tibble: 34 × 17  
## regionid communeid villageid hhid round s1\_q0 s1\_q1 s1\_q2 energ\_kcal  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+> <dbl+l> <dbl>  
## 1 2 26 1008 49484… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## 2 2 50 1124 49495… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## 3 1 43 114 49495… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## 4 1 43 114 49495… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## 5 2 33 1201 49504… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## 6 2 11 1304 49514… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## 7 2 12 1463 49525… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## 8 2 12 1463 49525… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## 9 2 12 1463 49525… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## 10 2 13 1490 49525… 1 [Bas… 1 [Bre… 0 [No] 2 [chi… NA  
## # ℹ 24 more rows  
## # ℹ 8 more variables: 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>

child\_endline[!complete.cases(child\_baseline), ]

## # A tibble: 34 × 17  
## regionid communeid villageid hhid round s1\_q0 s1\_q1 s1\_q2 energ\_kcal  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl>  
## 1 2 26 1008 4948… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 435.   
## 2 2 50 1124 4949… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 2.10  
## 3 1 43 114 4949… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 255.   
## 4 1 43 114 4949… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 826.   
## 5 2 33 1201 4950… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 840.   
## 6 2 11 1304 4951… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 358.   
## 7 2 12 1463 4952… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 603.   
## 8 2 12 1463 4952… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 0   
## 9 2 12 1463 4952… 2 [End… 1 [Bre… 0 [No] 2 [chi… NA   
## 10 2 13 1490 4952… 2 [End… 1 [Bre… 1 [Yes] 2 [chi… 363.   
## # ℹ 24 more rows  
## # ℹ 8 more variables: 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>

mother\_baseline[!complete.cases(child\_baseline), ]

## # A tibble: 34 × 17  
## regionid communeid villageid hhid round s1\_q0 s1\_q1 s1\_q2 V1  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl>  
## 1 2 26 1008 494848564… 1 [Bas… 1 [Bre… 0 [No] 1 [mot… NA   
## 2 2 50 1124 494950524… 1 [Bas… 1 [Bre… 0 [No] 1 [mot… NA   
## 3 1 43 114 494952485… 1 [Bas… 1 [Bre… 0 [No] 1 [mot… NA   
## 4 1 43 114 494952485… 1 [Bas… 1 [Bre… 0 [No] 1 [mot… NA   
## 5 2 33 1201 495048494… 1 [Bas… 1 [Bre… 1 [Yes] 1 [mot… 63.0  
## 6 2 11 1304 495148524… 1 [Bas… 1 [Bre… 0 [No] 1 [mot… NA   
## 7 2 12 1463 495254514… 1 [Bas… 1 [Bre… 0 [No] 1 [mot… NA   
## 8 2 12 1463 495254514… 1 [Bas… 1 [Bre… 0 [No] 1 [mot… NA   
## 9 2 12 1463 495254514… 1 [Bas… 1 [Bre… 1 [Yes] 1 [mot… 368.   
## 10 2 13 1490 495257484… 1 [Bas… 1 [Bre… 0 [No] 1 [mot… NA   
## # ℹ 24 more rows  
## # ℹ 8 more variables: protein\_g <dbl>, lipid\_tot\_g <dbl>, calcium\_mg <dbl>,  
## # iron\_mg <dbl>, V9 <dbl>, vit\_b6\_mg <dbl>, vit\_b12\_mcg <dbl>, vit\_c\_mg <dbl>

mother\_endline[!complete.cases(child\_baseline), ]

## # A tibble: 34 × 17  
## regionid communeid villageid hhid round s1\_q0 s1\_q1 s1\_q2 energ\_kcal  
## <dbl> <dbl> <dbl> <chr> <dbl+l> <dbl+l> <dbl+l> <dbl+l> <dbl>  
## 1 2 26 1008 4948… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 661.   
## 2 2 50 1124 4949… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 4.05  
## 3 1 43 114 4949… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 603.   
## 4 1 43 114 4949… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 1467.   
## 5 2 33 1201 4950… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 1167.   
## 6 2 11 1304 4951… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 776.   
## 7 2 12 1463 4952… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 874.   
## 8 2 12 1463 4952… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 0   
## 9 2 12 1463 4952… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 261.   
## 10 2 13 1490 4952… 2 [End… 1 [Bre… 1 [Yes] 1 [mot… 491.   
## # ℹ 24 more rows  
## # ℹ 8 more variables: 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>

1. la consommation moyenne d’énergie a chauqe repas dans la base . Pour cela nous allons utiliser mother\_baseline

str(mother\_baseline)

## tibble [1,064 × 17] (S3: tbl\_df/tbl/data.frame)  
## $ regionid : num [1:1064] 2 2 2 2 2 2 2 2 2 2 ...  
## ..- attr(\*, "label")= chr "Region ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ communeid : num [1:1064] 25 25 25 25 25 25 25 25 25 25 ...  
## ..- attr(\*, "label")= chr "Commune ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ villageid : num [1:1064] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...  
## ..- attr(\*, "label")= chr "Village ID"  
## ..- attr(\*, "format.stata")= chr "%8.0g"  
## $ hhid : chr [1:1064] "4948484848535052" "4948484848535053" "4948484848535055" "4948484848535056" ...  
## ..- attr(\*, "label")= chr "Household ID"  
## ..- attr(\*, "format.stata")= chr "%45s"  
## $ round : dbl+lbl [1:1064] 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Survey round : Baseline, Endline"  
## ..@ format.stata: chr "%10.0g"  
## ..@ labels : Named num [1:2] 1 2  
## .. ..- attr(\*, "names")= chr [1:2] "Baseline" "Endline"  
## $ s1\_q0 : dbl+lbl [1:1064] 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "eating occasion"  
## ..@ format.stata: chr "%27.0g"  
## ..@ labels : Named num [1:4] 1 2 3 4  
## .. ..- attr(\*, "names")= chr [1:4] "Breakfast" "Lunch" "Dinner" "Snacks"  
## $ s1\_q1 : dbl+lbl [1:1064] 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Meal consumed? Y/N"  
## ..@ format.stata: chr "%9.0g"  
## ..@ labels : Named num [1:2] 0 1  
## .. ..- attr(\*, "names")= chr [1:2] "No" "Yes"  
## $ s1\_q2 : dbl+lbl [1:1064] 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## ..@ label : chr "Who consummed?"  
## ..@ format.stata: chr "%16.0g"  
## ..@ labels : Named num [1:3] 1 2 3  
## .. ..- attr(\*, "names")= chr [1:3] "mother" "child" "mother and child"  
## $ V1 : num [1:1064] 680 563 237 873 408 ...  
## ..- attr(\*, "label")= chr "Consommation en ernergie (kcal)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ protein\_g : num [1:1064] 23.31 11.7 8.67 9.6 20.79 ...  
## ..- attr(\*, "label")= chr "Consommation en proteine (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ lipid\_tot\_g: num [1:1064] 5.3 6.49 1.39 4.33 1.3 ...  
## ..- attr(\*, "label")= chr "Consommation en lipide (g)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ calcium\_mg : num [1:1064] 62.7 116.3 11.9 26.6 60.6 ...  
## ..- attr(\*, "label")= chr "Consommation en calcium (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ iron\_mg : num [1:1064] 10.59 2.72 3.78 12.61 3.53 ...  
## ..- attr(\*, "label")= chr "Consommation en fer (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ V9 : num [1:1064] 4.51 3.38 1.38 2.94 2.95 ...  
## ..- attr(\*, "label")= chr "Consommation en zinc (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b6\_mg : num [1:1064] 0.3058 0.2248 0.0891 0.1032 0.1811 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B6 (mg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_b12\_mcg: num [1:1064] 0.00869 0.00823 0.00349 0 0 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine B12 (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"  
## $ vit\_c\_mg : num [1:1064] 0.0441 0.0002 0.0236 1.906 1.1143 ...  
## ..- attr(\*, "label")= chr "Consommation en vitamine C (mcg)"  
## ..- attr(\*, "format.stata")= chr "%10.0g"

On voit que la variable V1 ici correpond a la consommation d’énergie en Kcal

# Calcul de la moyenne en excluant les valeurs manquantes  
mean\_energy <- mean(mother\_baseline$V1, na.rm = TRUE)  
  
# Affichage du résultat  
cat("La consommation d’énergie moyenne à chaque repas est de", round(mean\_energy, 2), "kcal.\n")

## La consommation d’énergie moyenne à chaque repas est de 636.71 kcal.

### consommation moyenne après chaque repas

View(mother\_baseline)

### consommation moyenne après chaque repas

# Moyenne de la consommation d’énergie par type de repas  
energy\_per\_meal <- mother\_baseline %>%labelled::to\_factor() %>%   
 group\_by(s1\_q0) %>%  
 summarise(mean\_energy = mean(V1, na.rm = TRUE))  
  
# Affichage du résultat  
print(energy\_per\_meal)

## # A tibble: 1 × 2  
## s1\_q0 mean\_energy  
## <fct> <dbl>  
## 1 Breakfast 637.

1. Corrigons les valeurs manquantes des variables numeriques par leur moyenne et Sauvegardons maintenat les bases

# Fonction pour remplacer les valeurs manquantes par la moyenne de la colonne  
remplace\_na\_par\_moyenne <- function(df) {  
 df %>%  
 mutate(across(where(is.double), ~ ifelse(is.na(.), mean(., na.rm = TRUE), .)))  
}  
  
  
# Appliquer la fonction à chaque base de données  
mother\_baseline <- remplace\_na\_par\_moyenne(mother\_baseline)  
mother\_endline <- remplace\_na\_par\_moyenne(mother\_endline)  
child\_baseline <- remplace\_na\_par\_moyenne(child\_baseline)  
child\_endline <- remplace\_na\_par\_moyenne(child\_endline)  
menage <- remplace\_na\_par\_moyenne(menage)  
  
  
# Sauvegarder les bases de données après correction  
write\_dta(mother\_baseline, "../Données/mother\_baseline\_v1.dta")  
write\_dta(mother\_endline, "../Données/mother\_endline\_v1.dta")  
write\_dta(child\_baseline, "../Données/child\_baseline\_v1.dta")  
write\_dta(child\_endline, "../Données/child\_endline\_v1.dta")  
write\_dta(menage, "../Données/base\_menage\_final.dta")  
  
cat("Toutes les bases ont été enregistrées avec succès !")

## Toutes les bases ont été enregistrées avec succès !

verifier\_na <- function(df, nom) {  
 missing\_counts <- colSums(is.na(df))  
 missing\_counts <- missing\_counts[missing\_counts > 0] # Afficher uniquement les colonnes avec NA  
 if (length(missing\_counts) > 0) {  
 cat("\n", nom, ": Variables avec des valeurs manquantes\n")  
 print(missing\_counts)  
 } else {  
 cat( "Aucune valeur manquante \n")  
 }  
}  
  
# Appliquer la fonction sur toutes les bases  
verifier\_na(mother\_baseline, "mother\_baseline")

## Aucune valeur manquante

verifier\_na(mother\_endline, "mother\_endline")

## Aucune valeur manquante

verifier\_na(child\_baseline, "child\_baseline")

## Aucune valeur manquante

verifier\_na(child\_endline, "child\_endline")

## Aucune valeur manquante

verifier\_na(menage, "menage")

## Aucune valeur manquante

On voit que toutes les valeurs manquantes ont été corrigées

nrow(child\_baseline)

## [1] 1064

nrow(child\_endline)

## [1] 1064

nrow(mother\_baseline)

## [1] 1064

nrow(menage)

## [1] 1064

# Partie 2 : Empilement et fusion des données

1. Baseline
2. Empilement des bases de données des mères et des enfants

on va d’abord importer les nouvelles bases

child\_baseline <- haven::read\_dta("../Données/child\_baseline\_v1.dta")  
  
child\_endline <- haven::read\_dta("../Données/child\_endline\_v1.dta")  
  
mother\_baseline <- haven::read\_dta("../Données/mother\_baseline\_v1.dta")  
  
mother\_endline <- haven::read\_dta("../Données/mother\_endline\_v1.dta")  
  
menage <- haven::read\_dta("../Données/base\_menage\_final.dta")

mother\_baseline\_unique <- mother\_baseline %>% distinct(hhid, .keep\_all = TRUE)  
child\_baseline\_unique <- child\_baseline %>% distinct(hhid, .keep\_all = TRUE)  
  
colnames(mother\_baseline)

## [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"

colnames(child\_baseline)

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

merge\_mother\_child\_baseline <- left\_join(mother\_baseline\_unique, child\_baseline\_unique, by = "hhid")  
colnames(merge\_mother\_child\_baseline)

## [1] "regionid.x" "communeid.x" "villageid.x" "hhid"   
## [5] "round.x" "s1\_q0.x" "s1\_q1.x" "s1\_q2.x"   
## [9] "V1" "protein\_g.x" "lipid\_tot\_g.x" "calcium\_mg.x"   
## [13] "iron\_mg.x" "V9" "vit\_b6\_mg.x" "vit\_b12\_mcg.x"  
## [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" "protein\_g.y" "lipid\_tot\_g.y" "calcium\_mg.y"   
## [29] "iron\_mg.y" "zinc\_mg" "vit\_b6\_mg.y" "vit\_b12\_mcg.y"  
## [33] "vit\_c\_mg.y"

1. Renommons toutes les variables de consommation energ\_kcal jusqu’à vit\_c\_mcg en ajoutant le suffixe \_b pour faire référence à l’enquête Baseline.

# Renommer les variables de consommation  
merge\_mother\_child\_baseline <- merge\_mother\_child\_baseline %>%  
 rename\_with(~ paste0(., "\_b"), energ\_kcal:vit\_c\_mg.y)

colnames(merge\_mother\_child\_baseline)

## [1] "regionid.x" "communeid.x" "villageid.x" "hhid"   
## [5] "round.x" "s1\_q0.x" "s1\_q1.x" "s1\_q2.x"   
## [9] "V1" "protein\_g.x" "lipid\_tot\_g.x" "calcium\_mg.x"   
## [13] "iron\_mg.x" "V9" "vit\_b6\_mg.x" "vit\_b12\_mcg.x"   
## [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\_b" "protein\_g.y\_b" "lipid\_tot\_g.y\_b" "calcium\_mg.y\_b"   
## [29] "iron\_mg.y\_b" "zinc\_mg\_b" "vit\_b6\_mg.y\_b" "vit\_b12\_mcg.y\_b"  
## [33] "vit\_c\_mg.y\_b"

1. Créez une base de données qui résume les consommations journalières totales par individu (somme des 4 repas) pour l’énergie et tous les nutriments en utilisant la commande merge.

# Calculer les consommations totales par individu  
summary\_data <- merge\_mother\_child\_baseline %>%  
 group\_by(hhid) %>%  
 summarise(  
 total\_energy\_kcal = sum(energ\_kcal\_b, na.rm = TRUE),  
 total\_protein\_g = sum(protein\_g.x, protein\_g.y\_b, na.rm = TRUE),  
 total\_lipid\_g = sum(lipid\_tot\_g.x, lipid\_tot\_g.y\_b, na.rm = TRUE),  
 total\_calcium\_mg = sum(calcium\_mg.x, calcium\_mg.y\_b, na.rm = TRUE),  
 total\_iron\_mg = sum(iron\_mg.x, iron\_mg.y\_b, na.rm = TRUE),  
 total\_zinc\_mg = sum(zinc\_mg\_b, na.rm = TRUE),  
 total\_vit\_b6\_mg = sum(vit\_b6\_mg.x, vit\_b6\_mg.y\_b, na.rm = TRUE),  
 total\_vit\_b12\_mcg = sum(vit\_b12\_mcg.x, vit\_b12\_mcg.y\_b, na.rm = TRUE),  
 total\_vit\_c\_mg = sum(vit\_c\_mg.x, vit\_c\_mg.y\_b, na.rm = TRUE)  
 )  
  
# Visualiser le résumé  
print(summary\_data)

## # A tibble: 1,064 × 10  
## hhid total\_energy\_kcal total\_protein\_g total\_lipid\_g total\_calcium\_mg  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 49484848485… 355. 36.1 8.44 103.   
## 2 49484848485… 235. 16.1 11.2 210.   
## 3 49484848485… 165. 15.0 2.39 20.7  
## 4 49484848485… 1033. 21.0 9.45 58.2  
## 5 49484848485… 219. 31.9 2.00 93.1  
## 6 49484848485… 373. 33.1 4.39 33.9  
## 7 49484848485… 348. 29.3 6.66 99.3  
## 8 49484848485… 236. 31.7 8.92 121.   
## 9 49484848485… 233. 30.5 4.94 45.9  
## 10 49484848485… 377. 29.6 5.55 51.6  
## # ℹ 1,054 more rows  
## # ℹ 5 more variables: total\_iron\_mg <dbl>, total\_zinc\_mg <dbl>,  
## # total\_vit\_b6\_mg <dbl>, total\_vit\_b12\_mcg <dbl>, total\_vit\_c\_mg <dbl>

iv . Sauvegardez la base de données finale contenant seulement les hhid s1\_q2 energ\_kcal  
protein\_g lipid\_tot\_g calcium\_mg iron\_mg zinc\_mg sous le nom « baseline\_final.dta ».

# Créer la base de données finale avec une ligne pour chaque ménage  
baseline\_final <- merge\_mother\_child\_baseline %>%  
 select(hhid, s1\_q2.x, energ\_kcal\_b, protein\_g.x, lipid\_tot\_g.x, calcium\_mg.x, iron\_mg.x, zinc\_mg\_b) %>%  
 rename(  
 s1\_q2 = s1\_q2.x,  
 energ\_kcal = energ\_kcal\_b,  
 protein\_g = protein\_g.x,  
 lipid\_tot\_g = lipid\_tot\_g.x,  
 calcium\_mg = calcium\_mg.x,  
 iron\_mg = iron\_mg.x,  
 zinc\_mg = zinc\_mg\_b  
 ) %>%  
 # Garder toutes les lignes, car nous avons déjà une ligne pour la mère et une pour l'enfant  
 # Nous ne filtrons pas ici, pour garder toutes les entrées  
 select(hhid, s1\_q2, energ\_kcal, protein\_g, lipid\_tot\_g, calcium\_mg, iron\_mg, zinc\_mg)  
  
write\_dta(baseline\_final, "../Données/baseline\_final.dta")

print(baseline\_final)

## # A tibble: 1,064 × 8  
## hhid s1\_q2 energ\_kcal protein\_g lipid\_tot\_g calcium\_mg iron\_mg zinc\_mg  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 4948484848… 1 355. 23.3 5.30 62.7 10.6 2.41   
## 2 4948484848… 1 235. 11.7 6.49 116. 2.72 1.63   
## 3 4948484848… 1 165. 8.67 1.39 11.9 3.78 0.951  
## 4 4948484848… 1 1033. 9.60 4.33 26.6 12.6 3.48   
## 5 4948484848… 1 219. 20.8 1.30 60.6 3.53 1.58   
## 6 4948484848… 1 373. 20.5 2.71 20.6 2.15 2.09   
## 7 4948484848… 1 348. 16.1 3.62 53.9 6.36 2.25   
## 8 4948484848… 1 236. 22.8 5.86 74.2 10.6 2.10   
## 9 4948484848… 1 233. 21.3 3.46 30.7 9.99 1.51   
## 10 4948484848… 1 377. 18.4 3.45 32.0 10.1 2.31   
## # ℹ 1,054 more rows

1. Endline
2. empilons toutes les bases des mères et des enfants

# importation des bases   
mother\_endline <- haven::read\_dta("../Données/mother\_endline\_v1.dta")  
child\_endline <- haven::read\_dta("../Données/child\_endline\_v1.dta")

mother\_endline\_unique <- mother\_endline %>% distinct(hhid, .keep\_all = TRUE)  
child\_endline\_unique <- child\_endline %>% distinct(hhid, .keep\_all = TRUE)

merge\_mother\_child\_endline <- left\_join(  
 mother\_endline\_unique,  
 child\_endline\_unique,by="hhid"  
)

### affichons les noms des colonnes

merge\_mother\_child\_endline

## # A tibble: 1,064 × 33  
## regionid.x communeid.x villageid.x hhid round.x s1\_q0.x s1\_q1.x s1\_q2.x  
## <dbl> <dbl> <dbl> <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 2 25 1000 494848484… 2 1 1 1  
## 2 2 25 1000 494848484… 2 1 1 1  
## 3 2 25 1000 494848484… 2 1 1 1  
## 4 2 25 1000 494848484… 2 1 1 1  
## 5 2 25 1000 494848484… 2 1 1 1  
## 6 2 25 1000 494848484… 2 1 1 1  
## 7 2 25 1000 494848484… 2 1 1 1  
## 8 2 25 1000 494848484… 2 1 1 1  
## 9 2 25 1000 494848484… 2 1 1 1  
## 10 2 25 1000 494848484… 2 1 1 1  
## # ℹ 1,054 more rows  
## # 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>, s1\_q0.y <dbl>,  
## # s1\_q1.y <dbl>, s1\_q2.y <dbl>, energ\_kcal.y <dbl>, protein\_g.y <dbl>,  
## # lipid\_tot\_g.y <dbl>, calcium\_mg.y <dbl>, iron\_mg.y <dbl>, …

colnames(merge\_mother\_child\_endline)

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

1. resumons les consommations journalières totales par individu

# Chargement des bibliothèques nécessaires  
library(dplyr)  
  
# Résumer les consommations journalières totales par individu  
endline\_summary <- merge\_mother\_child\_endline %>%  
 group\_by(hhid) %>%  
 summarize(  
 energ\_kcal = sum(energ\_kcal.x, na.rm = TRUE) + sum(energ\_kcal.y, na.rm = TRUE),  
 protein\_g = sum(protein\_g.x, na.rm = TRUE) + sum(protein\_g.y, na.rm = TRUE),  
 lipid\_tot\_g = sum(lipid\_tot\_g.x, na.rm = TRUE) + sum(lipid\_tot\_g.y, na.rm = TRUE),  
 calcium\_mg = sum(calcium\_mg.x, na.rm = TRUE) + sum(calcium\_mg.y, na.rm = TRUE),  
 iron\_mg = sum(iron\_mg.x, na.rm = TRUE) + sum(iron\_mg.y, na.rm = TRUE),  
 zinc\_mg = sum(zinc\_mg.x, na.rm = TRUE) + sum(zinc\_mg.y, na.rm = TRUE),  
 s1\_q2 = first(s1\_q2.x), # Conserver la valeur de s1\_q2 de la mère  
 .groups = "drop" # Évite de garder le groupe après résumé  
 )  
  
# Vérifiez la structure de la base résumée  
print(endline\_summary)

## # A tibble: 1,064 × 8  
## hhid energ\_kcal protein\_g lipid\_tot\_g calcium\_mg iron\_mg zinc\_mg s1\_q2  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 4948484848… 3410. 73.6 142. 1299. 34.2 14.5 1  
## 2 4948484848… 806. 21.6 2.13 14.1 0.138 0.0435 1  
## 3 4948484848… 2180. 70.3 46.7 260. 31.0 11.4 1  
## 4 4948484848… 879. 10.4 5.85 15.8 23.8 2.57 1  
## 5 4948484848… 514. 20.7 2.94 846. 19.3 3.18 1  
## 6 4948484848… 1159. 50.6 17.5 2.05 0.206 0.0804 1  
## 7 4948484848… 704. 29.9 4.37 36.2 10.8 3.95 1  
## 8 4948484848… 899. 30.0 5.51 46.7 14.0 5.12 1  
## 9 4948484848… 818. 29.9 4.70 50.0 13.0 4.94 1  
## 10 4948484848… 856. 25.7 4.26 36.8 14.1 5.19 1  
## # ℹ 1,054 more rows

1. Sauvegardons la base de données finale contenant seulement les hhid s1\_q2 energ\_kcal  
   protein\_g lipid\_tot\_g calcium\_mg iron\_mg zinc\_mg sous le nom « endline\_final.dta ».

# Résumer les consommations journalières totales par individu  
endline\_summary2 <- merge\_mother\_child\_endline %>%  
 group\_by(hhid) %>%  
 summarize(  
 energ\_kcal = sum(energ\_kcal.x, na.rm = TRUE) + sum(energ\_kcal.y, na.rm = TRUE),  
 protein\_g = sum(protein\_g.x, na.rm = TRUE) + sum(protein\_g.y, na.rm = TRUE),  
 lipid\_tot\_g = sum(lipid\_tot\_g.x, na.rm = TRUE) + sum(lipid\_tot\_g.y, na.rm = TRUE),  
 calcium\_mg = sum(calcium\_mg.x, na.rm = TRUE) + sum(calcium\_mg.y, na.rm = TRUE),  
 iron\_mg = sum(iron\_mg.x, na.rm = TRUE) + sum(iron\_mg.y, na.rm = TRUE),  
 zinc\_mg = sum(zinc\_mg.x, na.rm = TRUE) + sum(zinc\_mg.y, na.rm = TRUE),  
 s1\_q2 = first(s1\_q2.x), # Conserver la valeur de s1\_q2 de la mère  
 .groups = "drop" # Évite de garder le groupe après résumé  
 )  
  
# Créer une base de données finale contenant les informations pour chaque ménage  
endline\_final <- endline\_summary2 %>%  
 select(hhid, s1\_q2, energ\_kcal, protein\_g, lipid\_tot\_g, calcium\_mg, iron\_mg, zinc\_mg)  
  
# Sauvegarder la base de données finale sous le nom "endline\_final.dta"  
write\_dta(endline\_final, "../Données/endline\_final.dta")

print(endline\_final)

## # A tibble: 1,064 × 8  
## hhid s1\_q2 energ\_kcal protein\_g lipid\_tot\_g calcium\_mg iron\_mg zinc\_mg  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 4948484848… 1 3410. 73.6 142. 1299. 34.2 14.5   
## 2 4948484848… 1 806. 21.6 2.13 14.1 0.138 0.0435  
## 3 4948484848… 1 2180. 70.3 46.7 260. 31.0 11.4   
## 4 4948484848… 1 879. 10.4 5.85 15.8 23.8 2.57   
## 5 4948484848… 1 514. 20.7 2.94 846. 19.3 3.18   
## 6 4948484848… 1 1159. 50.6 17.5 2.05 0.206 0.0804  
## 7 4948484848… 1 704. 29.9 4.37 36.2 10.8 3.95   
## 8 4948484848… 1 899. 30.0 5.51 46.7 14.0 5.12   
## 9 4948484848… 1 818. 29.9 4.70 50.0 13.0 4.94   
## 10 4948484848… 1 856. 25.7 4.26 36.8 14.1 5.19   
## # ℹ 1,054 more rows

# Fonction pour nettoyer les noms de colonnes  
clean\_column\_names <- function(data) {  
 colnames(data) <- make.names(colnames(data), unique = TRUE) # Renomme les colonnes pour les rendre valides  
 return(data)  
}  
  
# Nettoyage des noms de colonnes  
baseline\_final <- clean\_column\_names(baseline\_final)  
endline\_final <- clean\_column\_names(endline\_final)  
  
# Afficher les nouveaux noms de colonnes  
cat("Nouveaux noms des colonnes dans baseline\_final :\n")

## Nouveaux noms des colonnes dans baseline\_final :

print(colnames(baseline\_final))

## [1] "hhid" "s1\_q2" "energ\_kcal" "protein\_g" "lipid\_tot\_g"  
## [6] "calcium\_mg" "iron\_mg" "zinc\_mg"

cat("Nouveaux noms des colonnes dans endline\_final :\n")

## Nouveaux noms des colonnes dans endline\_final :

print(colnames(endline\_final))

## [1] "hhid" "s1\_q2" "energ\_kcal" "protein\_g" "lipid\_tot\_g"  
## [6] "calcium\_mg" "iron\_mg" "zinc\_mg"

# Fusion des bases de données sur hhid  
merged\_data <- full\_join(baseline\_final, endline\_final, by = "hhid")  
  
# Vérification du nombre d'observations  
n\_baseline <- nrow(baseline\_final)  
n\_endline <- nrow(endline\_final)  
n\_merged <- nrow(merged\_data)  
  
# Afficher le nombre d'observations  
cat("Nombre d'observations dans baseline\_final :", n\_baseline, "\n")

## Nombre d'observations dans baseline\_final : 1064

cat("Nombre d'observations dans endline\_final :", n\_endline, "\n")

## Nombre d'observations dans endline\_final : 1064

cat("Nombre d'observations après fusion :", n\_merged, "\n")

## Nombre d'observations après fusion : 1064

# Sauvegarder la base de données fusionnée  
#write\_dta(merged\_data, "merged\_data.dta")

1. En utilisant la base base\_menage\_final, fusionnez les informations sur les caractéristiques des ménages telles que région, commune, village, statut matrimonial, taille du ménage, niveau d’éducation du chef de ménage, le ratio de dépendance, score HFIAS, la variable de traitement T1.

# Base de données fusionnée (Baseline et Endline)  
base\_menage\_final <- read\_dta("../Données/base\_menage\_final.dta") # Base des caractéristiques des ménages  
colnames(base\_menage\_final)

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

# Fusion des données sur la colonne hhid  
final\_data <- base\_menage\_final %>%  
 left\_join(merged\_data, by = "hhid") %>% # Fusionner avec merged\_data sur hhid  
 select(hhid,   
 regionid,   
 communeid,   
 villageid,   
 round,   
 s1\_q0,   
 s1\_q1,   
 s1\_q2,   
 energ\_kcal,   
 protein\_g,   
 lipid\_tot\_g,   
 calcium\_mg,   
 iron\_mg,   
 zinc\_mg,   
 vit\_b6\_mg,   
 vit\_b12\_mcg,   
 vit\_c\_mg) # Sélection des colonnes spécifiées

print(final\_data)

## # A tibble: 1,064 × 17  
## hhid regionid communeid villageid round s1\_q0 s1\_q1 s1\_q2 energ\_kcal  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 494848484853… 2 25 1000 1 1 1 2 355.  
## 2 494848484853… 2 25 1000 1 1 1 2 235.  
## 3 494848484853… 2 25 1000 1 1 1 2 165.  
## 4 494848484853… 2 25 1000 1 1 1 2 1033.  
## 5 494848484853… 2 25 1000 1 1 1 2 219.  
## 6 494848484853… 2 25 1000 1 1 1 2 373.  
## 7 494848484853… 2 25 1000 1 1 1 2 348.  
## 8 494848484853… 2 25 1000 1 1 1 2 236.  
## 9 494848484853… 2 25 1000 1 1 1 2 233.  
## 10 494848484853… 2 25 1000 1 1 1 2 377.  
## # ℹ 1,054 more rows  
## # ℹ 8 more variables: 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>

1. Sauvegarder la base de données finale sous le nom base\_finale.dta.

# Sauvegarder la base de données finale  
write\_dta(final\_data, "../Données/base\_finale.dta")

# Partie 3 : Statistiques descriptives interessantes

On va sortir des statiques descriptives sur les bases child\_baseline et child\_endline

colnames(child\_baseline)

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

# ————————-

# Statistiques Descriptives de l’Énergie (energ\_kcal)

# ————————-

# Statistiques descriptives pour l’énergie

energy\_summary <- child\_baseline %>%  
 select(energ\_kcal) %>%  
 tbl\_summary(  
 statistic = list(all\_continuous() ~ "{mean} ({sd}) [min: {min}, max: {max}]"),  
 missing = "no"  
 ) %>%  
 modify\_header(label = "\*\*Énergie (kcal)\*\*")  
  
# Afficher le résumé  
energy\_summary

| **Énergie (kcal)** | **N = 1,064***1* |
| --- | --- |
| energ\_kcal | 352 (235) [min: 0, max: 2,296] |
| *1*Mean (SD) [min: Min, max: Max] | |

# ————————-

# Statistiques Descriptives des Protéines (protein\_g)

# ————————-

# Statistiques descriptives pour les protéines

protein\_summary <- child\_baseline %>%  
 select(protein\_g) %>%  
 tbl\_summary(  
 statistic = list(all\_continuous() ~ "{mean} ({sd}) [min: {min}, max: {max}]"),  
 missing = "no"  
 )%>% modify\_header(label = "\*\*Proteines \*\*")  
  
# Afficher le résumé  
protein\_summary

| \*\*Proteines \*\* | **N = 1,064***1* |
| --- | --- |
| protein\_g | 11 (8) [min: 0, max: 89] |
| *1*Mean (SD) [min: Min, max: Max] | |

# ————————-

# Statistiques Descriptives des Lipides (lipid\_tot\_g)

# ————————-

# Statistiques descriptives pour les lipides

lipid\_summary <- child\_baseline %>%  
 select(lipid\_tot\_g) %>%  
 tbl\_summary(  
 statistic = list(all\_continuous() ~ "{mean} ({sd}) [min: {min}, max: {max}]"),  
 missing = "no"  
 ) %>% modify\_header(label = "\*\*Lipides \*\*")  
  
# Afficher le résumé  
lipid\_summary

| \*\*Lipides \*\* | **N = 1,064***1* |
| --- | --- |
| lipid\_tot\_g | 4.9 (8.6) [min: 0.0, max: 147.3] |
| *1*Mean (SD) [min: Min, max: Max] | |

# ————————-

# Statistiques Descriptives des Vitamines

# ————————-

# Statistiques descriptives pour les vitamines B6, B12 et C

vitamin\_summary <- child\_baseline %>%  
 select(vit\_b6\_mg, vit\_b12\_mcg, vit\_c\_mg) %>%  
 tbl\_summary(  
 statistic = list(all\_continuous() ~ "{mean} ({sd}) [min: {min}, max: {max}]"),  
 missing = "no"  
 ) %>%  
 modify\_header(label = "\*\*Vitamines\*\*")   
  
# Afficher le résumé  
vitamin\_summary

| **Vitamines** | **N = 1,064***1* |
| --- | --- |
| vit\_b6\_mg | 0.17 (0.16) [min: 0.00, max: 1.80] |
| vit\_b12\_mcg | 0.03 (0.12) [min: 0.00, max: 1.47] |
| vit\_c\_mg | 5 (16) [min: 0, max: 186] |
| *1*Mean (SD) [min: Min, max: Max] | |

vitamin\_summary <- child\_endline %>%  
 select(vit\_b6\_mg, vit\_b12\_mcg, vit\_c\_mg) %>%  
 tbl\_summary(  
 statistic = list(all\_continuous() ~ "{mean} ({sd}) [min: {min}, max: {max}]"),  
 missing = "no"  
 ) %>%  
 modify\_header(label = "\*\*Vitamines\*\*")   
  
# Afficher le résumé  
vitamin\_summary

| **Vitamines** | **N = 1,064***1* |
| --- | --- |
| vit\_b6\_mg | 0.15 (0.17) [min: 0.00, max: 1.57] |
| vit\_b12\_mcg | 0.06 (0.27) [min: 0.00, max: 3.44] |
| vit\_c\_mg | 5 (14) [min: 0, max: 181] |
| *1*Mean (SD) [min: Min, max: Max] | |

# ————————-

# Fin

# ————————-