Test de management et d’analyse des données avec R

Djerakei MISTALENGAR

2025-02-25

Installation et chargement des packages

```
# Vérifier et installer les packages nécessaires

packages <- c("haven", "utils", "dplyr", "tidyverse", "gtsummary", "survey", "knitr")

for (pkg in packages) {
  if (!require(pkg, character.only = TRUE)) install.packages(pkg, dependencies = TRUE)
  library(pkg, character.only = TRUE)
}
```

```
# Supprimer toutes les variables de l'environnement
rm(list = ls())
```

Chargement des données

```
# Chargement des fichiers

mbl <- haven::read_dta("../Données/food_comp_mother_baseline.dta")

mel <- haven::read_dta("../Données/food_comp_mother_endline.dta")



str(mbl)
```

```
tibble [4,256 x 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… ..@ 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… ..@ 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… ..@ 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… ..@ 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”

```
str(mel)
```

tibble [4,256 x 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… ..@ 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… ..@ 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, 0, 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… ..@ 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”

```
cbl <- haven::read_dta("../Données/food_comp_child_baseline.dta")

cel <- haven::read_dta("../Données/food_comp_child_endline.dta")


str(cbl)
```

tibble [4,256 x 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… ..@ 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… ..@ 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… ..@ 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… ..@ 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”

```
str(cel)
```

tibble [4,256 x 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… ..@ 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… ..@ 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… ..@ 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… ..@ 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”

```
men <- haven::read_dta("../Données/base_menage.dta")

str(men)
```

tibble [1,065 x 21] (S3: tbl_df/tbl/data.frame) $ regionid : num [1:1065] 2 2 2 2 2 2 2 2 2 2 … ..- attr(, “label”)= chr “Region ID” ..- attr(, “format.stata”)= chr “%8.0g” $ communeid : num [1:1065] 25 25 25 25 25 25 25 25 25 25 … ..- attr(, “label”)= chr “Commune ID” ..- attr(, “format.stata”)= chr “%8.0g” $ villageid : num [1:1065] 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 … ..- attr(, “label”)= chr “Village ID” ..- attr(, “format.stata”)= chr “%8.0g” $ hhid : chr [1:1065] “4948484848535052” “4948484848535053” “4948484848535055” “4948484848535056” … ..- attr(, “label”)= chr “Household ID” ..- attr(, “format.stata”)= chr “%45s” $ hhsize : num [1:1065] 4 8 11 9 16 6 31 8 23 5 … ..- attr(, “label”)= chr “Household size” ..- attr(, “format.stata”)= chr “%10.0g” $ poly : dbl+lbl [1:1065] 0, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1… ..@ label : chr “Polygamous household?” ..@ format.stata: chr “%8.0g” ..@ labels : Named num [1:2] 0 1 .. ..- attr(, “names”)= chr [1:2] “Non” “Oui” $ hh_primary : dbl+lbl [1:1065] 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0… ..@ label : chr “Household head completed primary education” ..@ format.stata: chr “%9.0g” ..@ labels : Named num [1:2] 0 1 .. ..- attr(, “names”)= chr [1:2] “Non” “Oui” $ s1_q2 : dbl+lbl [1:1065] 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1… ..@ label : chr “Male head of household” ..@ format.stata: chr “%8.0g” ..@ labels : Named num [1:2] 0 1 .. ..- attr(, “names”)= chr [1:2] “Female” “Male” $ s1_q4a : num [1:1065] 37 52 67 38 75 41 56 52 63 26 … ..- attr(, “label”)= chr “Age head of household” ..- attr(, “format.stata”)= chr “%8.0g” $ s2_q1 : dbl+lbl [1:1065] 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0… ..@ label : chr “Is head of household literate in local language” ..@ format.stata: chr “%8.0g” ..@ labels : Named num [1:2] 0 1 .. ..- attr(, “names”)= chr [1:2] “Non” “Oui” $ s2_q2 : dbl+lbl [1:1065] 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0… ..@ label : chr “Is head of household literate in French?” ..@ format.stata: chr “%8.0g” ..@ labels : Named num [1:2] 0 1 .. ..- attr(, “names”)= chr [1:2] “Non” “Oui” $ s2_q4 : dbl+lbl [1:1065] 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1… ..@ label : chr “Has head of household been to school (formal or informal)?” ..@ format.stata: chr “%8.0g” ..@ labels : Named num [1:2] 0 1 .. ..- attr(, “names”)= chr [1:2] “Non” “Oui” $ s29_q1 : dbl+lbl [1:1065] 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0… ..@ label : chr “Est-ce qu’un membre de votre ménage a pris un prêt ou fait un emprunt en argent” ..@ format.stata: chr “%8.0g” ..@ labels : Named num [1:2] 0 1 .. ..- attr(, “names”)= chr [1:2] “Non” “Oui” $ demgrp1 : num [1:1065] 1 0 0 0 1 0 5 1 1 1 … ..- attr(, “label”)= chr “Number of children 0-36 months” ..- attr(, “format.stata”)= chr “%9.0g” $ demgrp2 : num [1:1065] 1 2 2 1 1 1 5 0 2 1 … ..- attr(, “label”)= chr “Number of children 36-72 months” ..- attr(, “format.stata”)= chr “%9.0g” $ demgrp3 : num [1:1065] 0 3 3 3 7 2 11 4 9 1 … ..- attr(, “label”)= chr “Number of adults 6-14 years” ..- attr(, “format.stata”)= chr “%9.0g” $ demgrp4 : num [1:1065] 1 2 5 5 6 3 10 3 11 2 … ..- attr(, “label”)= chr “Number of adults 14-65 years” ..- attr(, “format.stata”)= chr “%9.0g” $ demgrp5 : num [1:1065] 0 1 1 0 1 0 0 0 0 0 … ..- attr(, “label”)= chr “Number of elders 65+ years” ..- attr(, “format.stata”)= chr “%9.0g” $ dependencyratio: num [1:1065] 2 3 1.2 0.8 1.67 … ..- attr(, “label”)= chr “Dependency ratio” ..- attr(, “format.stata”)= chr “%9.0g” $ hfias_score : num [1:1065] 18 21 0 4 1 3 12 0 14 0 … ..- attr(, “label”)= chr “HFIAS Score (0-27)” ..- attr(, “format.stata”)= chr “%9.0g” $ T1 : dbl+lbl [1:1065] 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1… ..@ label : chr “Treatment: 1st level comparison” ..@ format.stata: chr “%12.0g” ..@ labels : Named num [1:2] 0 1 .. ..- attr(, “names”)= chr [1:2] “Controle” “Intervention” - attr(*, “label”)= chr “One row per household”

Partie 1 : Gestion et nettoyage des bases de données

```
colnames(mbl)
```

1. Vérifiez la présence de doublons dans les bases de données Baseline, Endline et ménage. Supprimez les doublons si nécessaire. [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”

```
View(mbl)
```

```
# Fonction pour vérifier et supprimer les doublons
clean_data <- function(df, key_vars) {
  df <- df %>%
    mutate(dupli = duplicated(df[key_vars]))  # Identifier les doublons
  print(sum(df$dupli))  # Nombre de doublons

  df <- df %>%
    distinct(across(all_of(key_vars)), .keep_all = TRUE)  # Supprimer les doublons

  return(df)
}
```

```
mbl<-clean_data(mbl, "hhid")
```

[1] 3192

```
print(mbl)
```

```
mel<-clean_data(mel, "hhid")
```

[1] 3192

```
cbl<-clean_data(cbl, "hhid")
```

[1] 3192

```
cbl
```

```
cel<-clean_data(cel, "hhid")
```

[1] 3192

```
# Vérification des noms de variables dans chaque base de données
names(mbl)  # Baseline des mères
```

[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” “dupli”

```
names(mel)  # Endline des mères
```

[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” “dupli”

```
names(cbl)  # Baseline des enfants
```

[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” “dupli”

```
names(cel)  # Endline des enfants
```

[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” “dupli”
```

```
names(men)  # Base des ménages
```

```
[1] “regionid” “communeid” “villageid” “hhid”
[5] “hhsize” “poly” “hh_primary” “s1_q2”
[9] “s1_q4a” “s2_q1” “s2_q2” “s2_q4”
[13] “s29_q1” “demgrp1” “demgrp2” “demgrp3”
[17] “demgrp4” “demgrp5” “dependencyratio” “hfias_score”
[21] “T1”
```

```

# Comparer les noms de variables pour identifier les différences

# Comparer les noms entre Baseline et Endline pour les mères
setdiff(names(mbl), names(mel))
```

2. Assurez-vous que les noms des variables sont cohérents entre les bases de données Baseline et Endline [1] “V1” “V9”

```
setdiff(names(mel), names(mbl))
```

```
[1] “energ_kcal” “zinc_mg”
```

```
# Comparer les noms entre Baseline et Endline pour les enfants
setdiff(names(cbl), names(cel))
```

```
character(0)
```

```
setdiff(names(cel), names(cbl))
```

```
character(0)
```

```
# Renommer la colonne 'V1' en 'energ_kcal'

colnames(mel)[colnames(mel) == "V1"] <- "energ_kcal"

colnames(mbl)[colnames(mbl) == "V1"] <- "energ_kcal"

# Vérifier les noms des variables après renaming
names(mbl)
```

```
[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” “V9” “vit_b6_mg”
[16] “vit_b12_mcg” “vit_c_mg” “dupli”
```

```
names(mel)
```

```
[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” “dupli”
```

```

# Vérifier les données manquantes dans chaque base
sum(is.na(mbl))  # Pour Baseline des mères
```

Veuillez vérifier soigneusement les données et corriger les données manquantes de certaines variables si possibles. [1] 615

```
sum(is.na(mel))  # Pour Endline des mères
```

[1] 702

```
sum(is.na(cbl))  # Pour Baseline des enfants
```

[1] 306

```
sum(is.na(cel))  # Pour Endline des enfants
```

[1] 486

```
sum(is.na(men))  # Pour la base des ménages
```

[1] 1

```
# Pour la base `mbl` (Baseline des mères), par exemple
colSums(is.na(mbl))  # Compter les NA pour chaque variable
```

regionid communeid villageid hhid round s1_q0 1 2 0 0 0 0 s1_q1 s1_q2 energ_kcal protein_g lipid_tot_g calcium_mg 0 0 68 68 68 68 iron_mg V9 vit_b6_mg vit_b12_mcg vit_c_mg dupli 68 68 68 68 68 0

```
colSums(is.na(mel))  # Idem pour la base `mel` (Endline des mères)
```

regionid communeid villageid hhid round s1_q0 0 0 0 0 0 0 s1_q1 s1_q2 energ_kcal protein_g lipid_tot_g calcium_mg 0 0 78 78 78 78 iron_mg zinc_mg vit_b6_mg vit_b12_mcg vit_c_mg dupli 78 78 78 78 78 0

```
# Répéter pour les autres bases
colSums(is.na(cbl))
```

regionid communeid villageid hhid round s1_q0 0 0 0 0 0 0 s1_q1 s1_q2 energ_kcal protein_g lipid_tot_g calcium_mg 0 0 34 34 34 34 iron_mg zinc_mg vit_b6_mg vit_b12_mcg vit_c_mg dupli 34 34 34 34 34 0

```
colSums(is.na(cel))
```

regionid communeid villageid hhid round s1_q0 0 0 0 0 0 0 s1_q1 s1_q2 energ_kcal protein_g lipid_tot_g calcium_mg 0 0 54 54 54 54 iron_mg zinc_mg vit_b6_mg vit_b12_mcg vit_c_mg dupli 54 54 54 54 54 0

```
colSums(is.na(men))
```

```
   regionid       communeid       villageid            hhid
          0               1               0               0
     hhsize            poly      hh_primary           s1_q2
          0               0               0               0
     s1_q4a           s2_q1           s2_q2           s2_q4
          0               0               0               0
     s29_q1         demgrp1         demgrp2         demgrp3
          0               0               0               0
    demgrp4         demgrp5 dependencyratio     hfias_score
          0               0               0               0
         T1
          0
```

```
# Imputation par la médiane pour les nutriments
mbl$protein_g[is.na(mbl$protein_g)] <- median(mbl$protein_g, na.rm = TRUE)
mel$protein_g[is.na(mel$protein_g)] <- median(mel$protein_g, na.rm = TRUE)

mbl$lipid_tot_g[is.na(mbl$lipid_tot_g)] <- median(mbl$lipid_tot_g, na.rm = TRUE)
mel$lipid_tot_g[is.na(mel$lipid_tot_g)] <- median(mel$lipid_tot_g, na.rm = TRUE)
```

```r
# Imputation par 0 pour les variables de type 'absence de consommation' comme `s1_q0`
mbl$s1_q0[is.na(mbl$s1_q0)] <- 0
mel$s1_q0[is.na(mel$s1_q0)] <- 0

# Vérification après traitement
colSums(is.na(mbl))  # Vérifier les NA après traitement
```

```
regionid communeid villageid hhid round s1_q0 1 2 0 0 0 0 s1_q1 s1_q2 energ_kcal protein_g lipid_tot_g
calcium_mg 0 0 68 0 0 68 iron_mg V9 vit_b6_mg vit_b12_mcg vit_c_mg dupli 68 68 68 68 68 0
```

```r
colSums(is.na(mel))
```

```
regionid communeid villageid hhid round s1_q0 0 0 0 0 0 0 s1_q1 s1_q2 energ_kcal protein_g lipid_tot_g
calcium_mg 0 0 78 0 0 78 iron_mg zinc_mg vit_b6_mg vit_b12_mcg vit_c_mg dupli 78 78 78 78 78 0
```

```r
# Imputation dans les données des enfants
cbl$protein_g[is.na(cbl$protein_g)] <- median(cbl$protein_g, na.rm = TRUE)
cel$protein_g[is.na(cel$protein_g)] <- median(cel$protein_g, na.rm = TRUE)

# Imputation des valeurs manquantes pour les nutriments spécifiques
cbl$calcium_mg[is.na(cbl$calcium_mg)] <- median(cbl$calcium_mg, na.rm = TRUE)
cbl$iron_mg[is.na(cbl$iron_mg)] <- median(cbl$iron_mg, na.rm = TRUE)
cbl$zinc_mg[is.na(cbl$zinc_mg)] <- median(cbl$zinc_mg, na.rm = TRUE)
cbl$vit_b6_mg[is.na(cbl$vit_b6_mg)] <- median(cbl$vit_b6_mg, na.rm = TRUE)
cbl$vit_b12_mcg[is.na(cbl$vit_b12_mcg)] <- median(cbl$vit_b12_mcg, na.rm = TRUE)
cbl$vit_c_mg[is.na(cbl$vit_c_mg)] <- median(cbl$vit_c_mg, na.rm = TRUE)


# Imputation par la médiane pour les variables avec des NA
cbl$energ_kcal[is.na(cbl$energ_kcal)] <- median(cbl$energ_kcal, na.rm = TRUE)
cel$calcium_mg[is.na(cel$calcium_mg)] <- median(cel$calcium_mg, na.rm = TRUE)
# Imputation par la médiane pour les nutriments manquants
cel$iron_mg[is.na(cel$iron_mg)] <- median(cel$iron_mg, na.rm = TRUE)
cel$zinc_mg[is.na(cel$zinc_mg)] <- median(cel$zinc_mg, na.rm = TRUE)
cel$vit_b6_mg[is.na(cel$vit_b6_mg)] <- median(cel$vit_b6_mg, na.rm = TRUE)
cel$vit_b12_mcg[is.na(cel$vit_b12_mcg)] <- median(cel$vit_b12_mcg, na.rm = TRUE)
cel$vit_c_mg[is.na(cel$vit_c_mg)] <- median(cel$vit_c_mg, na.rm = TRUE)


cbl$lipid_tot_g[is.na(cbl$lipid_tot_g)] <- median(cbl$lipid_tot_g, na.rm = TRUE)
cel$lipid_tot_g[is.na(cel$lipid_tot_g)] <- median(cel$lipid_tot_g, na.rm = TRUE)

# Et répéter pour d'autres variables comme calcium, zinc, etc.


# Imputation dans la base des ménages
men$hhsize[is.na(men$hhsize)] <- median(men$hhsize, na.rm = TRUE)


# Vérification après traitement
colSums(is.na(cbl))  # Vérifier les NA après traitement
```

```
regionid communeid villageid hhid round s1_q0 0 0 0 0 0 0 s1_q1 s1_q2 energ_kcal protein_g lipid_tot_g
calcium_mg 0 0 0 0 0 0 iron_mg zinc_mg vit_b6_mg vit_b12_mcg vit_c_mg dupli 0 0 0 0 0 0
```

```
colSums(is.na(cel))
```

regionid communeid villageid hhid round s1_q0 0 0 0 0 0 0 s1_q1 s1_q2 energ_kcal protein_g lipid_tot_g calcium_mg 0 0 54 0 0 0 iron_mg zinc_mg vit_b6_mg vit_b12_mcg vit_c_mg dupli 0 0 0 0 0 0

```
colSums(is.na(men))
```

```
   regionid       communeid       villageid            hhid 
          0               1               0               0 
     hhsize            poly      hh_primary           s1_q2 
          0               0               0               0 
     s1_q4a           s2_q1           s2_q2           s2_q4 
          0               0               0               0 
     s29_q1         demgrp1         demgrp2         demgrp3 
          0               0               0               0 
    demgrp4         demgrp5 dependencyratio     hfias_score 
          0               0               0               0 
         T1 
          0 
```

4. La consommation d'énergie moyenne à chaque repas pour l'ensemble des mères lors de l'enquête de base On va calculer la moyenne de la consommation d'énergie pour l'ensemble des repas dans l'enquête de base.

```
# Calculer la consommation moyenne d'énergie en ignorant les valeurs manquantes

mean_energ <- mean(mbl$energ_kcal, na.rm = TRUE)

# Afficher la moyenne
mean_energ
```

[1] 636.7127

```
# Sauvegarder les changements dans les fichiers modifiés

# Baseline des mères
haven::write_dta(mbl, "../Données/mother_baseline_v1.dta")

# Endline des mères
haven::write_dta(mel, "../Données/mother_endline_v1.dta")

 # Baseline des enfants
haven::write_dta(cbl, "../Données/child_baseline_v1.dta")

# Endline des enfants
haven::write_dta(cel, "../Données/child_endline_v1.dta")

 # Base des ménages
haven::write_dta(men, "../Données/base_menage_final.dta")
```

Partie 2 : Empilement et Fusion des données

1) Baseline

i) Empilez les bases de données

```r
# Charger les bases de données à nouveau

mbl <- haven::read_dta("../Données/mother_baseline_v1.dta")
cbl <- haven::read_dta("../Données/child_baseline_v1.dta")
```

```r
# Effectuer un left join (on veut garder toutes les lignes de la mère)
merged_data <- right_join(cbl, mbl, by = "hhid")



# Voir
View(merged_data)
```

ii) Renommage de toutes les variables de consommation energ_kcal jusqu'à vit_c_mcg en ajoutant le suffixe _b pour faire référence à l'enquête Baseline.

```r
merged_data <- merged_data %>%
  rename_with(~ gsub("\\.x$", "_b", .), contains(c("energ_kcal", "protein_g", "lipid_tot_g", "calcium_mg

# Renommer les colonnes de consommation d'enfants avec le suffixe "_c" pour l'enquête Baseline



merged_data <- merged_data %>%
  rename_with(~ gsub("\\.y$", "_c", .), contains(c("energ_kcal", "protein_g", "lipid_tot_g", "calcium_mg

# Vérifier les noms de colonnes après renommage
names(merged_data)
```

```
[1] “regionid.x” “communeid.x” “villageid.x” “hhid”
[5] “round.x” “s1_q0.x” “s1_q1.x” “s1_q2.x”
[9] “energ_kcal_b” “protein_g_b” “lipid_tot_g_b” “calcium_mg_b” [13] “iron_mg_b” “zinc_mg”
“vit_b6_mg_b” “vit_b12_mcg_b” [17] “vit_c_mg_b” “dupli.x” “regionid.y” “communeid.y”
[21] “villageid.y” “round.y” “s1_q0.y” “s1_q1.y”
[25] “s1_q2.y” “energ_kcal_c” “protein_g_c” “lipid_tot_g_c” [29] “calcium_mg_c” “iron_mg_c” “V9”
“vit_b6_mg_c”
[33] “vit_b12_mcg_c” “vit_c_mg_c” “dupli.y”
```

iii) Création d'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.

```r
# Créer un résumé des consommations journalières totales
summary_data <- merged_data %>%
  mutate(
    total_energ_kcal = energ_kcal_b + energ_kcal_c,
    total_protein_g = protein_g_b + protein_g_c,
    total_lipid_g = lipid_tot_g_b + lipid_tot_g_c,
    total_calcium_mg = calcium_mg_b + calcium_mg_c,
    total_iron_mg = iron_mg_b + iron_mg_c,
    total_zinc_mg = zinc_mg,
    total_vit_b6_mg = vit_b6_mg_b + vit_b6_mg_c,
    total_vit_b12_mcg = vit_b12_mcg_b + vit_b12_mcg_c,
    total_vit_c_mg = vit_c_mg_b + vit_c_mg_c
  ) %>%
  select(hhid, total_energ_kcal, total_protein_g, total_lipid_g, total_calcium_mg,
         total_iron_mg, total_zinc_mg, total_vit_b6_mg, total_vit_b12_mcg, total_vit_c_mg)

# Vérifier le résultat
head(summary_data)

# Sauvegarder la base de données résumée
haven::write_dta(summary_data, "../Données/summary_daily_consumption.dta")
```

iv) Sauvegarde la base de données finale

```
# Sélectionner les colonnes nécessaires
baseline_final <- merged_data %>%
  select(hhid, s1_q2.x, energ_kcal_b, protein_g_b, lipid_tot_g_b, calcium_mg_b, iron_mg_b, zinc_mg) %>%
  rename(
    s1_q2 = s1_q2.x,
    energ_kcal = energ_kcal_b,
    protein_g = protein_g_b,
    lipid_tot_g = lipid_tot_g_b,
    calcium_mg = calcium_mg_b,
    iron_mg = iron_mg_b
  )

# Vérifier le résultat
head(baseline_final)

# Sauvegarder la base de données finale
haven::write_dta(baseline_final, "../Données/baseline_final.dta")
```

```
# Charger les bases de données Endline
mel <- haven::read_dta("../Données/mother_endline_v1.dta")
cel <- haven::read_dta("../Données/child_endline_v1.dta")

# Renommer les variables de consommation pour l'enquête Endline en ajoutant le suffixe "_e"
mel <- mel %>%
  rename(
    energ_kcal_e = energ_kcal,
    protein_g_e = protein_g,
    lipid_tot_g_e = lipid_tot_g,
    calcium_mg_e = calcium_mg,
    iron_mg_e = iron_mg,
    zinc_mg_e = zinc_mg,
    vit_b6_mg_e = vit_b6_mg,
    vit_b12_mcg_e = vit_b12_mcg,
    vit_c_mg_e = vit_c_mg
  )

cel <- cel %>%
```

```r
  rename(
    energ_kcal_e = energ_kcal,
    protein_g_e = protein_g,
    lipid_tot_g_e = lipid_tot_g,
    calcium_mg_e = calcium_mg,
    iron_mg_e = iron_mg,
    zinc_mg_e = zinc_mg,
    vit_b6_mg_e = vit_b6_mg,
    vit_b12_mcg_e = vit_b12_mcg,
    vit_c_mg_e = vit_c_mg
  )

# Empiler les données des mères et des enfants pour chaque ménage
endline_merged <- bind_rows(mel, cel)

# Vérifier la base fusionnée
head(endline_merged)

# Sauvegarder la base de données empilée sous le nom "endline_merged_mother_child.dta"
haven::write_dta(endline_merged, "../Données/endline_merged_mother_child.dta")
```

Endline

```r
# Charger les bases de données des mères et des enfants
mel <- haven::read_dta("../Données/mother_endline_v1.dta")
cel <- haven::read_dta("../Données/child_endline_v1.dta")

# Fusionner les bases de données des mères et des enfants sur "hhid" (identifiant du ménage)
merged_endline <- right_join( cel,mel, by = "hhid")
```

```r
colnames(merged_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” “dupli.x” “regionid.y” “communeid.y”
[21] “villageid.y” “round.y” “s1_q0.y” “s1_q1.y”
[25] “s1_q2.y” “energ_kcal.y” “protein_g.y” “lipid_tot_g.y” [29] “calcium_mg.y” “iron_mg.y” “zinc_mg.y”
“vit_b6_mg.y”
[33] “vit_b12_mcg.y” “vit_c_mg.y” “dupli.y”
```

3) Fusionnez les données baseline_final.dta et endline_final.dta

```r
# Charger les deux bases de données
baseline_data <- haven::read_dta("../Données/baseline_final.dta")
endline_data <- haven::read_dta("../Données/endline_final.dta")

# Fusionner les données Baseline et Endline par hhid
merged_data <- left_join(baseline_data, endline_data, by = "hhid", suffix = c("_b", "_e"))

# Vérifier le résultat de la fusion
head(merged_data)
```

```r
# Sauvegarder la base de données fusionnée
haven::write_dta(merged_data, "../Données/merged_baseline_endline.dta")
```

```r
# Charger la base
base_menage <- read_dta("../Données/base_menage.dta")
```

```r
#Taille du menage
base_household_size <- base_menage %>%
  select(hhid, hhsize) %>%
  distinct()
```

```r
# Education
base_education_level <- base_menage %>%
  select(hhid, hh_primary) %>%
  distinct()
```

```r
# Ratio

base_dependence_ratio <- base_menage %>%
  select(hhid, dependencyratio) %>%
  distinct()
```

```r
# sécurité alimentaire)

base_HFIAS_score <- base_menage %>%
  select(hhid, hfias_score) %>%
  distinct()
```