PROJET LOGICIEL STATISTIQUES R

NFEGUE ZOFOA Stewart

PARTIE1 Importation et mise en forme

Importation de la base de données

library(readxl)  
  
# Importer la base de données Excel  
projet <- read\_excel("Base\_Partie 1.xlsx")

Faites un tableau qui resume les valeurs manquantes par variable

# Calcul du nombre de valeurs manquantes par variable  
valeurs\_manquantes <- colSums(is.na(projet))  
  
# Création du tableau résumant les valeurs manquantes  
tableau\_valeurs\_manquantes <- data.frame(Variable = names(valeurs\_manquantes), Valeurs\_Manquantes = valeurs\_manquantes)  
  
# Affichage du tableau résumant les valeurs manquantes  
print(tableau\_valeurs\_manquantes)

## Variable Valeurs\_Manquantes  
## key key 0  
## q1 q1 0  
## q2 q2 0  
## q23 q23 0  
## q24 q24 0  
## q24a\_1 q24a\_1 0  
## q24a\_2 q24a\_2 0  
## q24a\_3 q24a\_3 0  
## q24a\_4 q24a\_4 0  
## q24a\_5 q24a\_5 0  
## q24a\_6 q24a\_6 0  
## q24a\_7 q24a\_7 0  
## q24a\_9 q24a\_9 0  
## q24a\_10 q24a\_10 0  
## q25 q25 0  
## q26 q26 0  
## q12 q12 0  
## q14b q14b 1  
## q16 q16 1  
## q17 q17 131  
## q19 q19 120  
## q20 q20 0  
## filiere\_1 filiere\_1 0  
## filiere\_2 filiere\_2 0  
## filiere\_3 filiere\_3 0  
## filiere\_4 filiere\_4 0  
## q8 q8 0  
## q81 q81 0  
## gps\_menlatitude gps\_menlatitude 0  
## gps\_menlongitude gps\_menlongitude 0  
## submissiondate submissiondate 0  
## start start 0  
## today today 0

Vérifier s’il y a les des valeurs manquantes pour la variable key

# D'après le tableau\_valeurs\_manquantes, la variable KEY n'a aucune valeur manquante.

Création des variables

Renommer les variables

# Renommer les variables  
#Renommer la variable "q1" en "region"  
 colnames(projet)[colnames(projet) == "q1"] <- "region"  
 print(projet)

## # A tibble: 250 × 33  
## key region q2 q23 q24 q24a\_1 q24a\_2 q24a\_3 q24a\_4 q24a\_5 q24a\_6  
## <chr> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 uuid:68bf… Diour… Bamb… Femme 65 0 1 0 1 0 0  
## 2 uuid:d70b… Thiès Mbour Femme 52 1 1 0 0 1 0  
## 3 uuid:0ac1… Thiès Mbour Femme 65 1 1 0 0 0 0  
## 4 uuid:c52c… Thiès Mbour Femme 38 1 1 0 0 1 0  
## 5 uuid:ac17… Zigui… Bign… Homme 40 1 1 1 0 0 1  
## 6 uuid:5780… Zigui… Ouss… Femme 43 1 1 1 0 0 0  
## 7 uuid:c306… Thiès Thiès Femme 53 0 1 0 1 0 0  
## 8 uuid:74e6… Zigui… Zigu… Homme 33 1 0 0 0 0 0  
## 9 uuid:2ee0… Diour… Bamb… Femme 67 0 1 0 1 0 0  
## 10 uuid:5c80… Saint… Daga… Homme 35 1 1 0 0 0 0  
## # ℹ 240 more rows  
## # ℹ 22 more variables: q24a\_7 <dbl>, q24a\_9 <dbl>, q24a\_10 <dbl>, q25 <chr>,  
## # q26 <dbl>, q12 <chr>, q14b <chr>, q16 <chr>, q17 <chr>, q19 <chr>,  
## # q20 <chr>, filiere\_1 <dbl>, filiere\_2 <dbl>, filiere\_3 <dbl>,  
## # filiere\_4 <dbl>, q8 <chr>, q81 <chr>, gps\_menlatitude <dbl>,  
## # gps\_menlongitude <dbl>, submissiondate <dttm>, start <dttm>, today <dttm>

#Renommer la variable "q2" en "departement"  
 colnames(projet)[colnames(projet) == "q2"] <- "departement"  
 print(projet)

## # A tibble: 250 × 33  
## key region departement q23 q24 q24a\_1 q24a\_2 q24a\_3 q24a\_4 q24a\_5  
## <chr> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 uuid:68bff… Diour… Bambey Femme 65 0 1 0 1 0  
## 2 uuid:d70b3… Thiès Mbour Femme 52 1 1 0 0 1  
## 3 uuid:0ac18… Thiès Mbour Femme 65 1 1 0 0 0  
## 4 uuid:c52cf… Thiès Mbour Femme 38 1 1 0 0 1  
## 5 uuid:ac177… Zigui… Bignona Homme 40 1 1 1 0 0  
## 6 uuid:57809… Zigui… Oussouye Femme 43 1 1 1 0 0  
## 7 uuid:c3065… Thiès Thiès Femme 53 0 1 0 1 0  
## 8 uuid:74e60… Zigui… Ziguinchor Homme 33 1 0 0 0 0  
## 9 uuid:2ee01… Diour… Bambey Femme 67 0 1 0 1 0  
## 10 uuid:5c801… Saint… Dagana Homme 35 1 1 0 0 0  
## # ℹ 240 more rows  
## # ℹ 23 more variables: q24a\_6 <dbl>, q24a\_7 <dbl>, q24a\_9 <dbl>, q24a\_10 <dbl>,  
## # q25 <chr>, q26 <dbl>, q12 <chr>, q14b <chr>, q16 <chr>, q17 <chr>,  
## # q19 <chr>, q20 <chr>, filiere\_1 <dbl>, filiere\_2 <dbl>, filiere\_3 <dbl>,  
## # filiere\_4 <dbl>, q8 <chr>, q81 <chr>, gps\_menlatitude <dbl>,  
## # gps\_menlongitude <dbl>, submissiondate <dttm>, start <dttm>, today <dttm>

#Renommer la variable "q23" en "sexe"  
 colnames(projet)[colnames(projet) == "q23"] <- "sexe"  
 print(projet)

## # A tibble: 250 × 33  
## key region departement sexe q24 q24a\_1 q24a\_2 q24a\_3 q24a\_4 q24a\_5  
## <chr> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 uuid:68bff… Diour… Bambey Femme 65 0 1 0 1 0  
## 2 uuid:d70b3… Thiès Mbour Femme 52 1 1 0 0 1  
## 3 uuid:0ac18… Thiès Mbour Femme 65 1 1 0 0 0  
## 4 uuid:c52cf… Thiès Mbour Femme 38 1 1 0 0 1  
## 5 uuid:ac177… Zigui… Bignona Homme 40 1 1 1 0 0  
## 6 uuid:57809… Zigui… Oussouye Femme 43 1 1 1 0 0  
## 7 uuid:c3065… Thiès Thiès Femme 53 0 1 0 1 0  
## 8 uuid:74e60… Zigui… Ziguinchor Homme 33 1 0 0 0 0  
## 9 uuid:2ee01… Diour… Bambey Femme 67 0 1 0 1 0  
## 10 uuid:5c801… Saint… Dagana Homme 35 1 1 0 0 0  
## # ℹ 240 more rows  
## # ℹ 23 more variables: q24a\_6 <dbl>, q24a\_7 <dbl>, q24a\_9 <dbl>, q24a\_10 <dbl>,  
## # q25 <chr>, q26 <dbl>, q12 <chr>, q14b <chr>, q16 <chr>, q17 <chr>,  
## # q19 <chr>, q20 <chr>, filiere\_1 <dbl>, filiere\_2 <dbl>, filiere\_3 <dbl>,  
## # filiere\_4 <dbl>, q8 <chr>, q81 <chr>, gps\_menlatitude <dbl>,  
## # gps\_menlongitude <dbl>, submissiondate <dttm>, start <dttm>, today <dttm>

Créer la variable sexe\_2

#Avec la fonction ifelse, on peut coder la variable femme en 1 et si c'est un homme, ça donne la valeur 0  
  
 projet$sexe\_2 <- ifelse(projet$sexe == "Femme", 1,0)

Créer un dataframe nommé langues

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

langues = select(projet,key,starts\_with("q24a\_"))%>%data.frame()  
#la fonction starts\_with permet de prendre les variables qui commencent par "q24a\_"

Créer une variable parle égale au nombre de langues parlée par le dirigeant

library(dplyr)  
  
projet <- projet%>%  
 mutate(parle = rowSums(across(starts\_with("q24a\_")) > 0))  
  
#la fonction mutate permet d'ajouter la variable parle à la base projet et tient compte du nombre des langues parlées par le dirigeant

Sélectionner les variables key et parle, l’objet de reetour sera langues

library(dplyr)  
langues <- projet %>%  
 select(key, parle)

Merger les data.frame projet et langues

library(dplyr)  
  
 merge\_df <- merge(projet, langues, by = "row.names")

Analyses Descriptives

Le but de ces codes est de pouvoir faire les analyses descriptives de certaines variables par filière. Pour chaque filière, on recense les PME engagée d’où (filiere\_1 ==1 par exemple qui prend les PME dans la filière 1) Ensuite, on choisit les variables concernées avec la fonction select de dplyr et on fait des statistiques par rapport à chaque filiere avec la fonction tbl\_summary. A la fin, on merge les tableaux pour une seule sortie

# Charger les packages nécessaires  
library(dplyr)  
library(gtsummary)

## #StandWithUkraine

F1<- subset(projet,filiere\_1==1)%>%  
dplyr::select(sexe\_2,q25,q12,q81,filiere\_1)%>%  
gtsummary::tbl\_summary(  
 by= filiere\_1,  
 statistic=list(  
 all\_categorical() ~ "{n} / {N} ({p}%)"  
 ),  
 missing = "no",  
 percent="column"  
)%>%  
 modify\_header(label ~ "\*\*variable\*\*" )%>%  
bold\_labels()  
   
  
F2<- subset(projet,filiere\_2==1)%>%  
dplyr::select(sexe\_2,q25,q12,q81,filiere\_2)%>%  
gtsummary::tbl\_summary(  
 by= filiere\_2,  
 statistic=list(  
 all\_categorical() ~ "{n} / {N} ({p}%)"  
 ),  
 missing = "no",  
 percent="column"  
)%>%  
 modify\_header(label ~ "\*\*variable\*\*" )%>%  
bold\_labels()  
  
F3<- subset(projet,filiere\_3==1)%>%  
dplyr::select(sexe\_2,q25,q12,q81,filiere\_3)%>%  
gtsummary::tbl\_summary(  
 by= filiere\_3,  
 statistic=list(  
 all\_categorical() ~ "{n} / {N} ({p}%)"  
 ),  
 missing = "no",  
 percent="column"  
)%>%  
 modify\_header(label ~ "\*\*variable\*\*" )%>%  
bold\_labels()  
  
F4<- subset(projet,filiere\_4==1)%>%  
dplyr::select(sexe\_2,q25,q12,q81,filiere\_4)%>%  
gtsummary::tbl\_summary(  
 by= filiere\_4,  
 statistic = list(  
 all\_categorical()~ "{n} / {N} ({p}%)"  
 ),  
 missing = "no",  
 percent="column"  
)%>%  
 modify\_header(label ~ "\*\*variable\*\*" )%>%  
bold\_labels()  
  
  
filiere <- tbl\_merge(list(F1, F2, F3, F4))  
filiere

## Table printed with `knitr::kable()`, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| **variable** | **1**, N = 108 | **1**, N = 61 | **1**, N = 89 | **1**, N = 92 |
| --- | --- | --- | --- | --- |
| **sexe\_2** | 93 / 108 (86%) | 40 / 61 (66%) | 68 / 89 (76%) | 77 / 92 (84%) |
| **q25** |  |  |  |  |
| Aucun niveau | 43 / 108 (40%) | 13 / 61 (21%) | 26 / 89 (29%) | 11 / 92 (12%) |
| Niveau primaire | 23 / 108 (21%) | 17 / 61 (28%) | 24 / 89 (27%) | 26 / 92 (28%) |
| Niveau secondaire | 34 / 108 (31%) | 15 / 61 (25%) | 25 / 89 (28%) | 32 / 92 (35%) |
| Niveau Superieur | 8 / 108 (7.4%) | 16 / 61 (26%) | 14 / 89 (16%) | 23 / 92 (25%) |
| **q12** |  |  |  |  |
| Association | 2 / 108 (1.9%) | 3 / 61 (4.9%) |  | 2 / 92 (2.2%) |
| GIE | 79 / 108 (73%) | 35 / 61 (57%) | 73 / 89 (82%) | 77 / 92 (84%) |
| Informel | 23 / 108 (21%) | 12 / 61 (20%) | 5 / 89 (5.6%) | 3 / 92 (3.3%) |
| SA | 2 / 108 (1.9%) | 2 / 61 (3.3%) | 3 / 89 (3.4%) | 3 / 92 (3.3%) |
| SARL | 1 / 108 (0.9%) | 6 / 61 (9.8%) | 6 / 89 (6.7%) | 5 / 92 (5.4%) |
| SUARL | 1 / 108 (0.9%) | 3 / 61 (4.9%) | 2 / 89 (2.2%) | 2 / 92 (2.2%) |
| **q81** |  |  |  |  |
| Locataire | 12 / 108 (11%) | 7 / 61 (11%) | 11 / 89 (12%) | 9 / 92 (9.8%) |
| Propriétaire | 96 / 108 (89%) | 54 / 61 (89%) | 78 / 89 (88%) | 83 / 92 (90%) |

Un peu de cartographie

#importation des bibliothèques  
  
library(ggplot2)  
library(sf)

## Linking to GEOS 3.9.3, GDAL 3.5.2, PROJ 8.2.1; sf\_use\_s2() is TRUE

library(st)

## Le chargement a nécessité le package : sda

## Le chargement a nécessité le package : entropy

## Le chargement a nécessité le package : corpcor

## Le chargement a nécessité le package : fdrtool

Transformer le data.frame en données géographiques dont l’objet sera nommé projet\_map.

senegal <- st\_read("gadm41\_SEN\_0.shp")

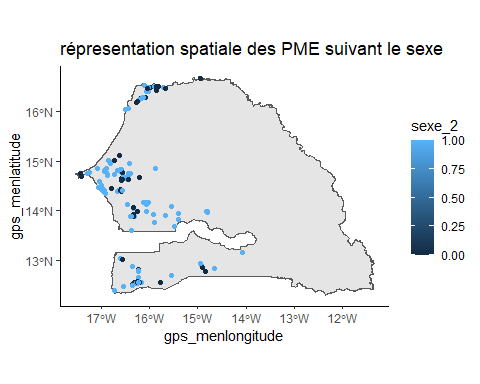
## Reading layer `gadm41\_SEN\_0' from data source   
## `C:\Users\Stewart Frenzy\Desktop\NfegueStewart\gadm41\_SEN\_0.shp'   
## using driver `ESRI Shapefile'  
## Simple feature collection with 1 feature and 2 fields  
## Geometry type: MULTIPOLYGON  
## Dimension: XY  
## Bounding box: xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207  
## Geodetic CRS: WGS 84

projet\_map<- st\_as\_sf(projet,coords=c("gps\_menlongitude","gps\_menlatitude"),crs=st\_crs(senegal))  
projet\_map<- st\_join(projet\_map,senegal)  
projet\_map

## Simple feature collection with 250 features and 35 fields  
## Geometry type: POINT  
## Dimension: XY  
## Bounding box: xmin: -17.43384 ymin: 12.39087 xmax: -14.08056 ymax: 16.6512  
## Geodetic CRS: WGS 84  
## # A tibble: 250 × 36  
## key region departement sexe q24 q24a\_1 q24a\_2 q24a\_3 q24a\_4 q24a\_5  
## \* <chr> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 uuid:68bff… Diour… Bambey Femme 65 0 1 0 1 0  
## 2 uuid:d70b3… Thiès Mbour Femme 52 1 1 0 0 1  
## 3 uuid:0ac18… Thiès Mbour Femme 65 1 1 0 0 0  
## 4 uuid:c52cf… Thiès Mbour Femme 38 1 1 0 0 1  
## 5 uuid:ac177… Zigui… Bignona Homme 40 1 1 1 0 0  
## 6 uuid:57809… Zigui… Oussouye Femme 43 1 1 1 0 0  
## 7 uuid:c3065… Thiès Thiès Femme 53 0 1 0 1 0  
## 8 uuid:74e60… Zigui… Ziguinchor Homme 33 1 0 0 0 0  
## 9 uuid:2ee01… Diour… Bambey Femme 67 0 1 0 1 0  
## 10 uuid:5c801… Saint… Dagana Homme 35 1 1 0 0 0  
## # ℹ 240 more rows  
## # ℹ 26 more variables: q24a\_6 <dbl>, q24a\_7 <dbl>, q24a\_9 <dbl>, q24a\_10 <dbl>,  
## # q25 <chr>, q26 <dbl>, q12 <chr>, q14b <chr>, q16 <chr>, q17 <chr>,  
## # q19 <chr>, q20 <chr>, filiere\_1 <dbl>, filiere\_2 <dbl>, filiere\_3 <dbl>,  
## # filiere\_4 <dbl>, q8 <chr>, q81 <chr>, submissiondate <dttm>, start <dttm>,  
## # today <dttm>, sexe\_2 <dbl>, parle <dbl>, geometry <POINT [°]>, GID\_0 <chr>,  
## # COUNTRY <chr>

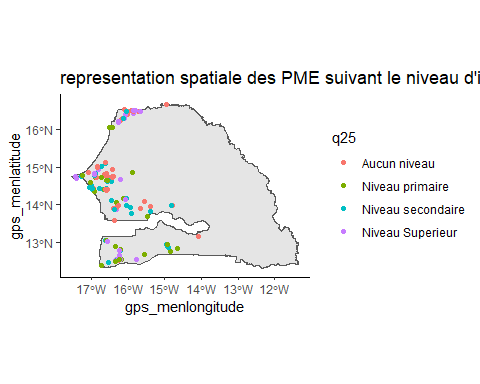
Faites une réprésentation spatiale des PME suivant le sexe

library(ggplot2)  
library(sf)  
ggplot(projet)+geom\_sf(data=senegal)+geom\_point(aes(x=gps\_menlongitude, y=gps\_menlatitude, color= sexe\_2))+ labs (title= "répresentation spatiale des PME suivant le sexe", x="gps\_menlongitude", y="gps\_menlatitude")+ theme\_classic()



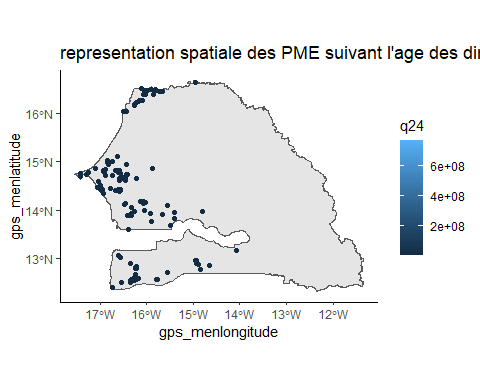
Faites une représentation spatiale suivant le niveau d’instruction

ggplot(projet)+geom\_sf(data=senegal)+geom\_point(aes(x=gps\_menlongitude, y=gps\_menlatitude,color=q25))+ labs(title="representation spatiale des PME suivant le niveau d'instruction", x="gps\_menlongitude",y= "gps\_menlatitude")+theme\_classic()



Faites une représentation spatiale suivant l’âge des dirigeants

ggplot(projet)+geom\_sf(data=senegal)+geom\_point(aes(x=gps\_menlongitude, y=gps\_menlatitude,color=q24))+ labs(title="representation spatiale des PME suivant l'age des dirigeants", x="gps\_menlongitude",y= "gps\_menlatitude")+theme\_classic()



PARTIE2

Nettoyage et gestion des données

Importation de la base

Importation du fichier data

library(readxl)  
# Importer un fichier Excel  
data <- read\_excel("Base\_Partie 2.xlsx", sheet = "data")

Importation du fichier district

library(readxl)  
district <- read\_excel("Base\_Partie 2.xlsx", sheet = "district")

Importation du fichier codebook

library(readxl)  
codebook <- read\_excel("Base\_Partie 2.xlsx", sheet = "codebook")

Renommer la variable “country\_destination” en “destination” et définir les valeurs négatives comme manquantes.

library(dplyr)  
  
# Renommer la variable "country\_destination" en "destination"  
data <- data %>% rename(destination = country\_destination)  
  
# Définir les valeurs négatives comme manquantes  
data[data < 0] <- NA

Créer une nouvelle variable contenant des tranches d’âge de 5 ans en utilisant la variable “age”.

la fonction cut permet de découper l’âge en tranche de 5 et la fonction breaks arrête lorsque le découpage en tranche de 5 a été atteint. Mutate permet de mettre la variable tranche\_age dans la base data

library(dplyr)  
  
# Créer une nouvelle variable avec des tranches d'âge de 5 ans  
data <- data %>%   
 mutate(tranche\_age = cut(age, breaks = seq(0, max(age) + 5, by = 5), labels = FALSE, right = FALSE))

Créer une nouvelle variable contenant le nombre d’entretiens réalisés par chaque agent recenseur.

library(dplyr)  
  
# Créer une nouvelle variable contenant le nombre d'entretiens réalisés par chaque agent recenseur  
data <- data %>%   
 group\_by(enumerator) %>%   
 mutate(nombre\_entretiens = n())  
  
# La nouvelle variable "nombre\_entretiens" contiendra le nombre d'entretiens réalisés par chaque agent recenseur.

library(dplyr)  
# Créer une nouvelle variable pour le groupe de traitement (1) ou de contrôle (0)  
set.seed(109) #Fixer l'aléa  
data$groupe\_traitement <- sample(c(0, 1), size = nrow(data), replace = TRUE)  
  
  
# La nouvelle variable "groupe\_traitement" contiendra des valeurs aléatoires de 0 ou 1, affectant chaque répondant à un groupe de traitement ou de contrôle.

Fusionner la taille de la population de chaque district (feuille 2) avec l’ensemble de données (feuille 1) afin que toutes les personnes interrogées aient une valeur correspondante représentant la taille de la population du district dans lequel elles vivent.

library(dplyr)  
  
# Fusionner les données en utilisant la fonction left\_join() de dplyr  
fused\_data <- left\_join(data, district, by = "district")

Calculer la durée de l’entretien et indiquer la durée moyenne de l’entretien par enquêteur.

library(dplyr)  
  
# Calculer la durée de chaque entretien en minutes  
fused\_data <- fused\_data %>%   
 mutate(duree\_entretien = difftime(endtime, starttime, units = "mins"))  
  
# Indiquer la durée moyenne de l'entretien par enquêteur  
duree\_moyenne\_par\_enqueteur <- fused\_data %>%   
 group\_by(enumerator) %>%   
 summarise(duree\_moyenne\_entretien = mean(duree\_entretien, na.rm = TRUE))

Renommez toutes les variables de l’ensemble de données en ajoutant le préfixe “endline\_” à l’aide d’une boucle.

# Obtenez les noms des colonnes actuels de l'ensemble de données  
noms\_colonnes <- colnames(fused\_data)  
  
# Créer un nouveau vecteur de noms de colonnes avec le préfixe "endline\_"  
noms\_colonnes\_nouveaux <- paste("endline\_", noms\_colonnes, sep = "")  
  
# Renommer les colonnes de l'ensemble de données avec les nouveaux noms  
fused\_data <- setNames(fused\_data, noms\_colonnes\_nouveaux)

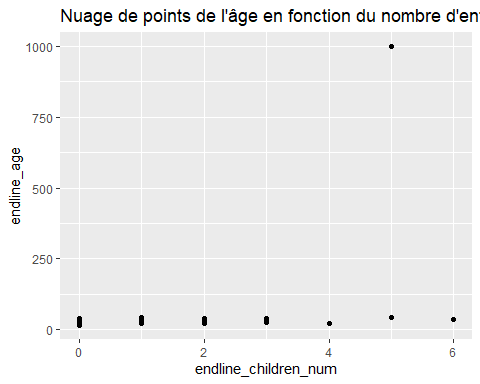
ANALYSE ET VISUALISATION DES DONNEES

Créez un tableau récapitulatif contenant l’âge moyen et le nombre moyen d’enfants par district.

library(dplyr)  
  
# Créer le tableau récapitulatif contenant l'âge moyen et le nombre moyen d'enfants par district  
tabeau\_recap <- fused\_data %>%   
 group\_by(endline\_district) %>%   
 summarise(endline\_age\_moyen = mean(endline\_age, na.rm = TRUE),  
 endline\_nb\_enfants\_moyen = mean(endline\_children\_num, na.rm = TRUE))

Créer un nuage de points de l’âge en fonction du nombre d’enfants

library(ggplot2)  
nuage <- data.frame(fused\_data$endline\_age,fused\_data$endline\_children\_num)  
  
# Créer le nuage de points  
ggplot(nuage, aes(x = fused\_data$endline\_children\_num, y = fused\_data$endline\_age)) +  
 geom\_point() +  
 labs(x = "endline\_children\_num", y = "endline\_age", title = "Nuage de points de l'âge en fonction du nombre d'enfants")



Estimation de l’effet d’appartenance à un groupe

library(gtsummary)  
library(ggplot2)  
modèle <- stats::lm(fused\_data$endline\_intention~ fused\_data$endline\_groupe\_traitement, data= fused\_data)  
modèle%>% gtsummary::tbl\_regression()

## Table printed with `knitr::kable()`, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| **Characteristic** | **Beta** | **95% CI** | **p-value** |
| --- | --- | --- | --- |
| fused\_data$endline\_groupe\_traitement | -0.49 | -1.2, 0.21 | 0.2 |

Tableau avec 3 Modèles

modele\_A <- modèle%>%gtsummary::tbl\_regression()  
modele\_B <- lm(fused\_data$endline\_intention ~ fused\_data$endline\_age+fused\_data$endline\_sex, data= fused\_data)%>%tbl\_regression()  
modele\_C<- lm(fused\_data$endline\_intention ~ fused\_data$endline\_age + fused\_data$endline\_sex + fused\_data$endline\_district, fused\_data)%>%tbl\_regression()  
  
gtsummary::tbl\_stack(  
list(modele\_A, modele\_B, modele\_C),  
group\_header = c("Modele A","Modele B","Modele C")  
)

## Table printed with `knitr::kable()`, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| **Group** | **Characteristic** | **Beta** | **95% CI** | **p-value** |
| --- | --- | --- | --- | --- |
| Modele A | fused\_dataendline\_age | 0.00 | 0.00, 0.00 | 0.9 |
|  | fused\_dataendline\_age | 0.00 | 0.00, 0.00 | >0.9 |
|  | fused\_dataendline\_district | 0.09 | -0.07, 0.24 | 0.3 |