GNING_Ibrahima_projet_R_ENSAE 2023

Ibrahima GNING

2023-07-23

Partie 1

1 Préparation des données

1.1 Importation et mise en forme

Chargement des bibliothèques

```
library(readxl)
library(dplyr)
library(janitor)
library(gt)
library(gtsummary)
library(sf)
library(leaflet)
library(raster)
library(readxl)
library(ggplot2)
```

Lecture du fichier Base_Partie 1 avec la fontion read_excel du package readxl qui va la transformer en data.frame et l'assigner à l'objet projet

```
projet <- readxl::read_excel("Base_Partie 1.xlsx")
```

Ce code permet de selectionner toutes les variables sauf la variable key dans la base projet avec la fontion select de la bibliothèque dplyr et l'assigner à l'objet projet_selection

```
projet_selection <- dplyr::select(projet, -key)
```

on calcule le nombre de valeurs manquantes par variable avec la fonction sapply qui parcourt toutes les lignes ; la fonction is.na permet de verifier s'il y'a ou pas une valeur manquante et la fontion sum somme le nombre valeurs manquantes.Enfin on crée un dataframe résumant les valeurs manquantes par variable

```
vm <- sapply(projet_selection, function(x) sum(is.na(x)))
tableau<- data.frame(variables= names(vm),Valeurs_manquantes = vm)
tableau %>%
  gt() %>%
  tab_header(title =md("**projet**"),
```

```
            subtitle = md("Le nombre de valeurs manquantes pour chaque
variable")) %>%
  tab_source_note("projet")
```

Table 1: **projet**

Le nombre de valeurs manquantes pour chaque variable

variables	Valeurs_manquantes
q1	0
q2	0
q23	0
q24	0
q24a_1	0
q24a_2	0
q24a_3	0
q24a_4	0
q24a_5	0
q24a_6	0
q24a_7	0
q24a_9	0
q24a_10	0
q25	0
q26	0
q12	0
q14b	1
q16	1
q17	131
q19	120
q20	0
filiere_1	0
filiere_2	0
filiere_3	0
filiere_4	0
q8	0
q81	0
gps_menlatitude	0
gps_menlongitude	0
submissiondate	0
start	0
today	0

projet

```
# Utiliser sapply pour vérifier les valeurs manquantes dans la variable 'key'
resultats_manquants <- sapply(projet$key, function(x) sum(is.na(x)) > 0)
```

```
# Obtenir les indices des lignes où les valeurs sont manquantes
indices_manquants <- data.frame(indices_manquants =
which(resultats_manquants))

# Afficher le tableau des indices des lignes avec des valeurs manquantes
indices_manquants %>%
  gt() %>%
  tab_header(title =md("**projet**"),
             subtitle = md("indices des lignes avec des valeurs manquantes"))
%>%
  tab_source_note("projet")
```

Table 2: **projet**

indices des lignes avec des valeurs manquantes

	indices_manquants

projet

1.2 Création de variables

names(projet) donnes tous les noms des variables et on identifie le nom q1 puis le rempace par region et la meme logique est appliquée pour q1 et q23

```
names(projet)[names(projet) == "q1"] <- "region"
names(projet)[names(projet) == "q2"] <- "departement"
names(projet)[names(projet) == "q23"] <- "sexe"
```

Dans ce code, ifelse() est utilisé pour évaluer une condition. Si la condition projet$sexe == “Femme” est vraie, la valeur 1 est assignée à sexe_2,sinon la valeur 0 est assignée.

```
projet$sexe_2 <- ifelse(projet$sexe == "Femme", 1, 0)
```

Le code recherche les variables ayant comme prefice q24a_ et les selectionne pour former avec ces variables et key le dataframe langues

```
langues <- projet[, c("key", grep("^q24a_", names(projet), value = TRUE))]
```

on somme toutes les variables en ligne de la base langes pour obtenir le nombre de langue parlée par le dirigeant de la PME.

```
langues$parle <- rowSums(langues[, -1])
```

on selection dans langues les variables key et parle pour generer un dataframe de meme nom

```
langues <- dplyr::select(langues, key, parle)
```

Dans ce code, la fonction merge() est utilisée pour fusionner projet et langues en utilisant la variable commune key. Le résultat de la fusion est stocké dans projet_langues.

```
projet_langues <- merge(projet, langues, by = "key")
```

2 Analyses descriptives

La durée entre la date de soumission des informations de la PME et date de début de l'enrégistrement des informations de la PME par l'enquêteur en heure.

```
projet_langues$duree <- difftime(projet_langues$submissiondate,
                                 projet_langues$start,units="hours")
```

Convertir en format numérique

```
projet_langues$duree <- as.numeric(projet_langues$duree)
```

filiere 1

```
#Conversion de la variable 'filiere_1' en facteur avec des niveaux
#personnalisés
projet_langues$filiere_1<-factor(projet_langues$filiere_1, levels=
c(0,1),labels = c("Non","Oui"))
# Création d'un tableau récapitulatif (summary) avec des statistiques
#spécifiées
tab11<-
projet_langues%>%tbl_summary(include=c(sexe,q25,q12,q81,q24,parle,duree),
                                    # Variables à inclure dans le tableau
                             by=filiere_1,percent = "column",
                             # Regroupement par la variable 'filiere_1'
                             label= list(q25 ~ "Niveau d'instruction ",
                                         q12 ~ "Statut juridique",
                                         q81 ~ "propriétaire ou locataire",
                                         q24~ "Age du dirigeant"),
                             # Étiquette pour les variables
                             type = list(parle ~ "continuous2",
                                         duree ~ "continuous2"),
                             # Type de sommaire pour la variable 'parle'et
'duree'
                             statistic=list(sexe~"{p}%",
                                            q25~"{p}%",
                                            q12~"{p}%",
                                            q12~"{p}%",
                                            q81~"{p}%",
                                            q24~"{median}",
                                            parle~"{mean}",
                                            duree~"{max}"),
                             # Statistique des variables
                             duree ~ scales::label_number(suffix = " hours")
                             #afficher l'unité de la variable duree
                             ) %>% add_n()
tab11
```

Characteristic	**N**	**Non**, N = 142[1]	**Oui**, N = 108[1]
sexe	250		

Characteristic	**N**	**Non**, N = 142[1]	**Oui**, N = 108[1]
Femme		69%	86%
Homme		31%	14%
Niveau d'instruction	250		
Aucun niveau		25%	40%
Niveau primaire		23%	21%
Niveau secondaire		28%	31%
Niveau Superieur		23%	7.4%
Statut juridique	250		
Association		2.8%	1.9%
GIE		70%	73%
Informel		11%	21%
SA		3.5%	1.9%
SARL		8.5%	0.9%
SUARL		4.2%	0.9%
propriétaire ou locataire	250		
Locataire		8.5%	11%
Propriétaire		92%	89%
Age du dirigeant	250	54	58
parle	250		
Mean		2.62	2.32
duree	250		
Maximum		664 hours	479 hours

[1]%; Median

Création d'un tableau récapitulatif stratifié par la variable sexe

```
tab12 <- projet_langues %>%
  dplyr::select(sexe, q25, q12, q81,filiere_1) %>%
  # Variables à inclure dans le tableau
```

```r
  tbl_strata(
    strata = sexe,# Variable utilisée pour stratifier le tableau
    .tbl_fun = ~ .x %>% # Fonction appliquée à chaque groupe stratifié

      tbl_summary(by = filiere_1, # Regroupement par la variable 'filiere_1'
                  missing = "no",# Gestion des valeurs manquantes
                  label= list(q25 ~ "Niveau d'instruction ",
                              q12 ~ "Statut juridique",
                                q81 ~ "propriétaire ou locataire")) %>%
      add_n(), # Ajouter le nombre total d'observations pour chaque groupe
    .combine_with = "tbl_stack",
    ## préciser comment combiner les tableaux de chaque groupe.
    ##Par défaut, il combine avec "tbl_merge"
    .header = "*{strata}*",
    .quiet = TRUE # permet de combiner des tableaux avec des
                  # entetes differents
  )


tab12  # Afficher le tableau récapitulatif stratifié
```

Group	Characteristic	N	**Non**, N = 98[1]	**Oui**, N = 93[1]
Femme	Niveau d'instruction	191		
	Aucun niveau		32 (33%)	38 (41%)
	Niveau primaire		28 (29%)	20 (22%)
	Niveau secondaire		26 (27%)	30 (32%)
	Niveau Superieur		12 (12%)	5 (5.4%)
	Statut juridique	191		
	Association		1 (1.0%)	2 (2.2%)
	GIE		79 (81%)	70 (75%)
	Informel		12 (12%)	20 (22%)
	SA		1 (1.0%)	0 (0%)
	SARL		2 (2.0%)	0 (0%)
	SUARL		3 (3.1%)	1 (1.1%)
	propriétaire ou locataire	191		
	Locataire		7 (7.1%)	9 (9.7%)

Group	Characteristic	N	**Non**, N = 98[1]	**Oui**, N = 93[1]
	Propriétaire		91 (93%)	84 (90%)
Homme	Niveau d'instruction	59		
	Aucun niveau		4 (9.1%)	5 (33%)
	Niveau primaire		5 (11%)	3 (20%)
	Niveau secondaire		14 (32%)	4 (27%)
	Niveau Superieur		21 (48%)	3 (20%)
	Statut juridique	59		
	Association		3 (6.8%)	0 (0%)
	GIE		21 (48%)	9 (60%)
	Informel		3 (6.8%)	3 (20%)
	SA		4 (9.1%)	2 (13%)
	SARL		10 (23%)	1 (6.7%)
	SUARL		3 (6.8%)	0 (0%)
	propriétaire ou locataire	59		
	Locataire		5 (11%)	3 (20%)
	Propriétaire		39 (89%)	12 (80%)

[1]n (%)

Cette ligne de code fusionnera les tableaux tab11 et tab12 en un seul tableau empilé pour obtenir le tableau final avec la filiere 1.

```
tab1 <- gtsummary::tbl_stack(
  list(tab11, tab12),
  quiet = TRUE)
tab1
```

Group	Characteristic	N[1]	**Non**, N = 142[1]	**Oui**, N = 108
	sexe	250		
	Femme		69%	86%
	Homme		31%	14%
	Niveau d'instruction	250		

Group	Characteristic	N[1]	Non, N = 142[1]	Oui, N = 108
	Aucun niveau		25%	40%
	Niveau primaire		23%	21%
	Niveau secondaire		28%	31%
	Niveau Superieur		23%	7.4%
	Statut juridique	250		
	Association		2.8%	1.9%
	GIE		70%	73%
	Informel		11%	21%
	SA		3.5%	1.9%
	SARL		8.5%	0.9%
	SUARL		4.2%	0.9%
	propriétaire ou locataire	250		
	Locataire		8.5%	11%
	Propriétaire		92%	89%
	Age du dirigeant	250	54	58
	parle	250		
	Mean		2.62	2.32
	duree	250		
	Maximum		664 hours	479 hours
Femme	Niveau d’instruction	191		
	Aucun niveau		32 (33%)	38 (41%)
	Niveau primaire		28 (29%)	20 (22%)
	Niveau secondaire		26 (27%)	30 (32%)
	Niveau Superieur		12 (12%)	5 (5.4%)
	Statut juridique	191		
	Association		1 (1.0%)	2 (2.2%)

Group	Characteristic	N[1]	Non, N = 142[1]	Oui, N = 108
	GIE		79 (81%)	70 (75%)
	Informel		12 (12%)	20 (22%)
	SA		1 (1.0%)	0 (0%)
	SARL		2 (2.0%)	0 (0%)
	SUARL		3 (3.1%)	1 (1.1%)
	propriétaire ou locataire	191		
	Locataire		7 (7.1%)	9 (9.7%)
	Propriétaire		91 (93%)	84 (90%)
Homme	Niveau d'instruction	59		
	Aucun niveau		4 (9.1%)	5 (33%)
	Niveau primaire		5 (11%)	3 (20%)
	Niveau secondaire		14 (32%)	4 (27%)
	Niveau Superieur		21 (48%)	3 (20%)
	Statut juridique	59		
	Association		3 (6.8%)	0 (0%)
	GIE		21 (48%)	9 (60%)
	Informel		3 (6.8%)	3 (20%)
	SA		4 (9.1%)	2 (13%)
	SARL		10 (23%)	1 (6.7%)
	SUARL		3 (6.8%)	0 (0%)
	propriétaire ou locataire	59		
	Locataire		5 (11%)	3 (20%)
	Propriétaire		39 (89%)	12 (80%)

[1]%; Median

on reprend le meme travail avec les filieres 2,3,4 pour tabi {i=2,3,4}

filiere 2

```r
#Conversion de la variable 'filiere_2' en facteur avec des niveaux
#personnalisés
projet_langues$filiere_2<-factor(projet_langues$filiere_2, levels=
c(0,1),labels = c("Non","Oui"))
# Création d'un tableau récapitulatif (summary) avec des statistiques
#spécifiées
tab21<-
projet_langues%>%tbl_summary(include=c(sexe,q25,q12,q81,q24,parle,duree),
                                    # Variables à inclure dans le tableau
                                    by=filiere_2,percent = "column",
                                    # Regroupement par la variable
'filiere_2'
                                    label= list(q25 ~ "Niveau d’instruction
",
                                                q12 ~ "Statut juridique",
                                                q81 ~ "propriétaire ou
locataire",
                                                q24~ "Age du dirigeant"),
                                    # Étiquette pour les variables
                                    type = list(parle ~ "continuous2",
                                                duree ~ "continuous2"),
                                    # Type de sommaire pour la variable
'parle'et'duree'
                                    statistic=list(sexe~"{p}%",
                                                   q25~"{p}%",
                                                   q12~"{p}%",
                                                   q12~"{p}%",
                                                   q81~"{p}%",
                                                   q24~"{median}",
                                                   parle~"{mean}",
                                                   duree~"{max}"),
                                    # Statistique des variables
                                    duree ~ scales::label_number(suffix = "
hours")
                                    #afficher l'unité de la variable duree)
                                    # Statistique des variables

) %>% add_n()
tab21
```

Characteristic	**N**	**Non**, N = 189[1]	**Oui**, N = 61[1]
sexe	250		
Femme		80%	66%
Homme		20%	34%
Niveau d’instruction	250		
Aucun niveau		35%	21%

Characteristic	N	Non, N = 189[1]	Oui, N = 61[1]
Niveau primaire		21%	28%
Niveau secondaire		31%	25%
Niveau Superieur		13%	26%
Statut juridique	250		
Association		1.6%	4.9%
GIE		76%	57%
Informel		14%	20%
SA		2.6%	3.3%
SARL		3.7%	9.8%
SUARL		2.1%	4.9%
propriétaire ou locataire	250		
Locataire		9.0%	11%
Propriétaire		91%	89%
Age du dirigeant	250	57	47
parle	250		
Mean		2.29	3.11
duree	250		
Maximum		664 hours	430 hours

[1]%; Median

Création d'un tableau récapitulatif stratifié par la variable sexe

```
tab22 <- projet_langues %>%
  dplyr::select(sexe, q25, q12, q81,filiere_2) %>%
  # Variables à inclure dans le tableau
  tbl_strata(
    strata = sexe,# Variable utilisée pour stratifier le tableau
    .tbl_fun = ~ .x %>% # Fonction appliquée à chaque groupe stratifié

      tbl_summary(by = filiere_2, # Regroupement par la variable 'filiere_2'
                  missing = "no",# Gestion des valeurs manquantes
                  label= list(q25 ~ "Niveau d'instruction ",
                              q12 ~ "Statut juridique",
```

```
                                q81 ~ "propriétaire ou locataire"))%>%
    add_n(), # Ajouter le nombre total d'observations pour chaque groupe
  .combine_with = "tbl_stack",
  ## préciser comment combiner les tableaux de chaque groupe.
  ##Par défaut, il combine avec "tbl_merge"
  .header = "*{strata}*",
  .quiet = TRUE # permet de combiner des tableaux avec des
  # entetes differents
)


tab22  # Afficher le tableau récapitulatif stratifié
```

Group	Characteristic	N	**Non**, N = 151[1]	**Oui**, N = 40[1]
Femme	Niveau d'instruction	191		
	Aucun niveau		58 (38%)	12 (30%)
	Niveau primaire		33 (22%)	15 (38%)
	Niveau secondaire		47 (31%)	9 (23%)
	Niveau Superieur		13 (8.6%)	4 (10%)
	Statut juridique	191		
	Association		2 (1.3%)	1 (2.5%)
	GIE		122 (81%)	27 (68%)
	Informel		22 (15%)	10 (25%)
	SA		1 (0.7%)	0 (0%)
	SARL		1 (0.7%)	1 (2.5%)
	SUARL		3 (2.0%)	1 (2.5%)
	propriétaire ou locataire	191		
	Locataire		13 (8.6%)	3 (7.5%)
	Propriétaire		138 (91%)	37 (93%)
Homme	Niveau d'instruction	59		
	Aucun niveau		8 (21%)	1 (4.8%)
	Niveau primaire		6 (16%)	2 (9.5%)
	Niveau secondaire		12 (32%)	6 (29%)

Group	Characteristic	N	**Non**, N = 151[1]	**Oui**, N = 40[1]
	Niveau Superieur		12 (32%)	12 (57%)
	Statut juridique	59		
	Association		1 (2.6%)	2 (9.5%)
	GIE		22 (58%)	8 (38%)
	Informel		4 (11%)	2 (9.5%)
	SA		4 (11%)	2 (9.5%)
	SARL		6 (16%)	5 (24%)
	SUARL		1 (2.6%)	2 (9.5%)
	propriétaire ou locataire	59		
	Locataire		4 (11%)	4 (19%)
	Propriétaire		34 (89%)	17 (81%)

[1]n (%)

Cette ligne de code fusionnera les tableaux tab21 et tab22 en un seul tableau empilé pour obtenir le tableau final avec la filiere 2.

```
tab2 <- gtsummary::tbl_stack(
  list(tab21, tab22),
  quiet = TRUE)
tab2
```

Group	Characteristic	N[1]	**Non**, N = 189[1]	**Oui**, N = 61
	sexe	250		
	Femme		80%	66%
	Homme		20%	34%
	Niveau d’instruction	250		
	Aucun niveau		35%	21%
	Niveau primaire		21%	28%
	Niveau secondaire		31%	25%
	Niveau Superieur		13%	26%
	Statut juridique	250		

Group	Characteristic	N[1]	Non, N = 189[1]	Oui, N = 61
	Association		1.6%	4.9%
	GIE		76%	57%
	Informel		14%	20%
	SA		2.6%	3.3%
	SARL		3.7%	9.8%
	SUARL		2.1%	4.9%
	propriétaire ou locataire	250		
	Locataire		9.0%	11%
	Propriétaire		91%	89%
	Age du dirigeant	250	57	47
	parle	250		
	Mean		2.29	3.11
	duree	250		
	Maximum		664 hours	430 hours
Femme	Niveau d’instruction	191		
	Aucun niveau		58 (38%)	12 (30%)
	Niveau primaire		33 (22%)	15 (38%)
	Niveau secondaire		47 (31%)	9 (23%)
	Niveau Superieur		13 (8.6%)	4 (10%)
	Statut juridique	191		
	Association		2 (1.3%)	1 (2.5%)
	GIE		122 (81%)	27 (68%)
	Informel		22 (15%)	10 (25%)
	SA		1 (0.7%)	0 (0%)
	SARL		1 (0.7%)	1 (2.5%)
	SUARL		3 (2.0%)	1 (2.5%)

Group	Characteristic	N[1]	**Non**, N = 189[1]	**Oui**, N = 61
	propriétaire ou locataire	191		
	Locataire		13 (8.6%)	3 (7.5%)
	Propriétaire		138 (91%)	37 (93%)
Homme	Niveau d'instruction	59		
	Aucun niveau		8 (21%)	1 (4.8%)
	Niveau primaire		6 (16%)	2 (9.5%)
	Niveau secondaire		12 (32%)	6 (29%)
	Niveau Superieur		12 (32%)	12 (57%)
	Statut juridique	59		
	Association		1 (2.6%)	2 (9.5%)
	GIE		22 (58%)	8 (38%)
	Informel		4 (11%)	2 (9.5%)
	SA		4 (11%)	2 (9.5%)
	SARL		6 (16%)	5 (24%)
	SUARL		1 (2.6%)	2 (9.5%)
	propriétaire ou locataire	59		
	Locataire		4 (11%)	4 (19%)
	Propriétaire		34 (89%)	17 (81%)

[1]%; Median

filiere 3

```
#Conversion de la variable 'filiere_3' en facteur avec des niveaux
#personnalisés
projet_langues$filiere_3<-factor(projet_langues$filiere_3, levels=
c(0,1),labels = c("Non","Oui"))
# Création d'un tableau récapitulatif (summary) avec des statistiques
#spécifiées
tab31<-
projet_langues%>%tbl_summary(include=c(sexe,q25,q12,q81,q24,parle,duree),
                                    # Variables à inclure dans le tableau
                                    by=filiere_3,percent = "column",
                                    # Regroupement par la variable
```

```
'filiere_3'
                                label= list(q25 ~ "Niveau d’instruction
",
                                            q12 ~ "Statut juridique",
                                            q81 ~ "propriétaire ou
locataire",
                                            q24~ "Age du dirigeant"),
                                # Étiquette pour les variables
                                type = list(parle ~ "continuous2",
                                            duree ~ "continuous2"),
                                # Type de sommaire pour la variable
'parle'et 'duree'
                                statistic=list(sexe~"{p}%",
                                               q25~"{p}%",
                                               q12~"{p}%",
                                               q12~"{p}%",
                                               q81~"{p}%",
                                               q24~"{median}",
                                               parle~"{mean}",
                                               duree~"{max}"),
                                # Statistique des variables
                                duree ~ scales::label_number(suffix = "
hours")
                                #afficher l'unité de la variable duree)
                                # Statistique des variables

) %>% add_n()
tab31
```

Characteristic	**N**	**Non**, N = 161[1]	**Oui**, N = 89[1]
sexe	250		
Femme		76%	76%
Homme		24%	24%
Niveau d’instruction	250		
Aucun niveau		33%	29%
Niveau primaire		20%	27%
Niveau secondaire		30%	28%
Niveau Superieur		17%	16%
Statut juridique	250		
Association		3.7%	0%
GIE		66%	82%

Characteristic	N	Non, N = 161[1]	Oui, N = 89[1]
Informel		20%	5.6%
SA		2.5%	3.4%
SARL		4.3%	6.7%
SUARL		3.1%	2.2%
propriétaire ou locataire	250		
Locataire		8.1%	12%
Propriétaire		92%	88%
Age du dirigeant	250	54	55
parle	250		
Mean		2.62	2.26
duree	250		
Maximum		479 hours	664 hours

[1]%; Median

Création d'un tableau récapitulatif stratifié par la variable sexe

```
tab32 <- projet_langues %>%
   dplyr::select(sexe, q25, q12, q81,filiere_3) %>%
  # Variables à inclure dans le tableau
  tbl_strata(
    strata = sexe,# Variable utilisée pour stratifier le tableau
    .tbl_fun = ~ .x %>% # Fonction appliquée à chaque groupe stratifié

      tbl_summary(by = filiere_3, # Regroupement par la variable 'filiere_3'
                  missing = "no",# Gestion des valeurs manquantes
                  label= list(q25 ~ "Niveau d'instruction ",
                              q12 ~ "Statut juridique",
                              q81 ~ "propriétaire ou locataire"))%>%
      add_n(), # Ajouter le nombre total d'observations pour chaque groupe
    .combine_with = "tbl_stack",
    ## préciser comment combiner les tableaux de chaque groupe.
    ##Par défaut, il combine avec "tbl_merge"
    .header = "*{strata}*",
    .quiet = TRUE # permet de combiner des tableaux avec des
    # entetes differents
  )
```

```
tab32  # Afficher le tableau récapitulatif stratifié
```

Group	Characteristic	N	**Non**, N = 123[1]	**Oui**, N = 68[1]
Femme	Niveau d'instruction	191		
	Aucun niveau		48 (39%)	22 (32%)
	Niveau primaire		28 (23%)	20 (29%)
	Niveau secondaire		35 (28%)	21 (31%)
	Niveau Superieur		12 (9.8%)	5 (7.4%)
	Statut juridique	191		
	Association		3 (2.4%)	0 (0%)
	GIE		87 (71%)	62 (91%)
	Informel		29 (24%)	3 (4.4%)
	SA		0 (0%)	1 (1.5%)
	SARL		1 (0.8%)	1 (1.5%)
	SUARL		3 (2.4%)	1 (1.5%)
	propriétaire ou locataire	191		
	Locataire		8 (6.5%)	8 (12%)
	Propriétaire		115 (93%)	60 (88%)
Homme	Niveau d'instruction	59		
	Aucun niveau		5 (13%)	4 (19%)
	Niveau primaire		4 (11%)	4 (19%)
	Niveau secondaire		14 (37%)	4 (19%)
	Niveau Superieur		15 (39%)	9 (43%)
	Statut juridique	59		
	Association		3 (7.9%)	0 (0%)
	GIE		19 (50%)	11 (52%)
	Informel		4 (11%)	2 (9.5%)

Group	Characteristic	N	**Non**, N = 123[1]	**Oui**, N = 68[1]
	SA		4 (11%)	2 (9.5%)
	SARL		6 (16%)	5 (24%)
	SUARL		2 (5.3%)	1 (4.8%)
	propriétaire ou locataire	59		
	Locataire		5 (13%)	3 (14%)
	Propriétaire		33 (87%)	18 (86%)

[1]n (%)

Cette ligne de code fusionnera les tableaux tab31 et tab32 en un seul tableau empilé pour obtenir le tableau final avec la filiere 3.

```
tab3 <- gtsummary::tbl_stack(
  list(tab31, tab32),
  quiet = TRUE)
tab3
```

Group	Characteristic	N[1]	**Non**, N = 161[1]	**Oui**, N = 89
	sexe	250		
	Femme		76%	76%
	Homme		24%	24%
	Niveau d'instruction	250		
	Aucun niveau		33%	29%
	Niveau primaire		20%	27%
	Niveau secondaire		30%	28%
	Niveau Superieur		17%	16%
	Statut juridique	250		
	Association		3.7%	0%
	GIE		66%	82%
	Informel		20%	5.6%
	SA		2.5%	3.4%
	SARL		4.3%	6.7%

Group	Characteristic	N[1]	Non, N = 161[1]	Oui, N = 89
	SUARL		3.1%	2.2%
	propriétaire ou locataire	250		
	Locataire		8.1%	12%
	Propriétaire		92%	88%
	Age du dirigeant	250	54	55
	parle	250		
	Mean		2.62	2.26
	duree	250		
	Maximum		479 hours	664 hours
Femme	Niveau d'instruction	191		
	Aucun niveau		48 (39%)	22 (32%)
	Niveau primaire		28 (23%)	20 (29%)
	Niveau secondaire		35 (28%)	21 (31%)
	Niveau Superieur		12 (9.8%)	5 (7.4%)
	Statut juridique	191		
	Association		3 (2.4%)	0 (0%)
	GIE		87 (71%)	62 (91%)
	Informel		29 (24%)	3 (4.4%)
	SA		0 (0%)	1 (1.5%)
	SARL		1 (0.8%)	1 (1.5%)
	SUARL		3 (2.4%)	1 (1.5%)
	propriétaire ou locataire	191		
	Locataire		8 (6.5%)	8 (12%)
	Propriétaire		115 (93%)	60 (88%)
Homme	Niveau d'instruction	59		
	Aucun niveau		5 (13%)	4 (19%)

Group	Characteristic	N[1]	**Non**, N = 161[1]	**Oui**, N = 89
	Niveau primaire		4 (11%)	4 (19%)
	Niveau secondaire		14 (37%)	4 (19%)
	Niveau Superieur		15 (39%)	9 (43%)
	Statut juridique	59		
	Association		3 (7.9%)	0 (0%)
	GIE		19 (50%)	11 (52%)
	Informel		4 (11%)	2 (9.5%)
	SA		4 (11%)	2 (9.5%)
	SARL		6 (16%)	5 (24%)
	SUARL		2 (5.3%)	1 (4.8%)
	propriétaire ou locataire	59		
	Locataire		5 (13%)	3 (14%)
	Propriétaire		33 (87%)	18 (86%)

[1]%; Median

filiere 4

```
#Conversion de la variable 'filiere_4' en facteur avec des niveaux
#personnalisés
projet_langues$filiere_4<-factor(projet_langues$filiere_4, levels=
c(0,1),labels = c("Non","Oui"))
# Création d'un tableau récapitulatif (summary) avec des statistiques
#spécifiées
tab41<-
projet_langues%>%tbl_summary(include=c(sexe,q25,q12,q81,q24,parle,duree),
                                    # Variables à inclure dans le tableau
                                    by=filiere_4,percent = "column",
                                    # Regroupement par la variable
'filiere_4'
                                    label= list(q25 ~ "Niveau d’instruction
",
                                                q12 ~ "Statut juridique",
                                                q81 ~ "propriétaire ou
locataire",
                                                q24~ "Age du dirigeant"),
                                    # Étiquette pour les variables
                                    type = list(parle ~ "continuous2",
```

```
                                        duree ~ "continuous2"),
                                    # Type de sommaire pour la variable
'parle'et 'duree'
                                    statistic=list(sexe~"{p}%",
                                                   q25~"{p}%",
                                                   q12~"{p}%",
                                                   q12~"{p}%",
                                                   q81~"{p}%",
                                                   q24~"{median}",
                                                   parle~"{mean}",
                                                   duree~"{max}"),
                                    # Statistique des variables
                                    duree ~ scales::label_number(suffix = "
hours")

                                    #afficher l'unité de la variable duree)
                                    # Statistique des variables

) %>% add_n()
tab41
```

Characteristic	**N**	**Non**, N = 158[1]	**Oui**, N = 92[1]
sexe	250		
Femme		72%	84%
Homme		28%	16%
Niveau d'instruction	250		
Aucun niveau		43%	12%
Niveau primaire		19%	28%
Niveau secondaire		27%	35%
Niveau Superieur		11%	25%
Statut juridique	250		
Association		2.5%	2.2%
GIE		65%	84%
Informel		22%	3.3%
SA		2.5%	3.3%
SARL		5.1%	5.4%
SUARL		3.2%	2.2%
propriétaire ou locataire	250		

Characteristic	N	Non, N = 158[1]	Oui, N = 92[1]
Locataire		9.5%	9.8%
Propriétaire		91%	90%
Age du dirigeant	250	54	57
parle	250		
Mean		2.17	3.04
duree	250		
Maximum		664 hours	383 hours

[1]%; Median

Création d'un tableau récapitulatif stratifié par la variable sexe

```
tab42 <- projet_langues %>%
  dplyr::select(sexe, q25, q12, q81,filiere_4) %>%
  # Variables à inclure dans le tableau
  tbl_strata(
    strata = sexe,# Variable utilisée pour stratifier le tableau
    .tbl_fun = ~ .x %>% # Fonction appliquée à chaque groupe stratifié

      tbl_summary(by = filiere_4, # Regroupement par la variable 'filiere_4'
                  missing = "no",# Gestion des valeurs manquantes
                  label= list(q25 ~ "Niveau d'instruction ",
                              q12 ~ "Statut juridique",
                              q81 ~ "propriétaire ou locataire"))%>%
      add_n(), # Ajouter le nombre total d'observations pour chaque groupe
    .combine_with = "tbl_stack",
    ## préciser comment combiner les tableaux de chaque groupe.
    ##Par défaut, il combine avec "tbl_merge"
    .header = "*{strata}*",
    .quiet = TRUE # permet de combiner des tableaux avec des
    # entetes differents
  )


tab42  # Afficher le tableau récapitulatif stratifié
```

Group	Characteristic	N	Non, N = 114[1]	Oui, N = 77[1]
Femme	Niveau d'instruction	191		
	Aucun niveau		60 (53%)	10 (13%)
	Niveau primaire		22 (19%)	26 (34%)

Group	Characteristic	N	**Non**, N = 114[1]	**Oui**, N = 77[1]
	Niveau secondaire		28 (25%)	28 (36%)
	Niveau Superieur		4 (3.5%)	13 (17%)
	Statut juridique	191		
	Association		3 (2.6%)	0 (0%)
	GIE		76 (67%)	73 (95%)
	Informel		31 (27%)	1 (1.3%)
	SA		1 (0.9%)	0 (0%)
	SARL		1 (0.9%)	1 (1.3%)
	SUARL		2 (1.8%)	2 (2.6%)
	propriétaire ou locataire	191		
	Locataire		8 (7.0%)	8 (10%)
	Propriétaire		106 (93%)	69 (90%)
Homme	Niveau d'instruction	59		
	Aucun niveau		8 (18%)	1 (6.7%)
	Niveau primaire		8 (18%)	0 (0%)
	Niveau secondaire		14 (32%)	4 (27%)
	Niveau Superieur		14 (32%)	10 (67%)
	Statut juridique	59		
	Association		1 (2.3%)	2 (13%)
	GIE		26 (59%)	4 (27%)
	Informel		4 (9.1%)	2 (13%)
	SA		3 (6.8%)	3 (20%)
	SARL		7 (16%)	4 (27%)
	SUARL		3 (6.8%)	0 (0%)
	propriétaire ou locataire	59		
	Locataire		7 (16%)	1 (6.7%)

Group	Characteristic	N	**Non**, N = 114[1]	**Oui**, N = 77[1]
	Propriétaire		37 (84%)	14 (93%)

[1]n (%)

Cette ligne de code fusionnera les tableaux tab41 et tab42 en un seul tableau empilé pour obtenir le tableau final avec la filiere 4.

```
tab4 <- gtsummary::tbl_stack(
  list(tab41, tab42),
  quiet = TRUE)
tab4
```

Group	Characteristic	N[1]	**Non**, N = 158[1]	**Oui**, N = 92
	sexe	250		
	Femme		72%	84%
	Homme		28%	16%
	Niveau d'instruction	250		
	Aucun niveau		43%	12%
	Niveau primaire		19%	28%
	Niveau secondaire		27%	35%
	Niveau Superieur		11%	25%
	Statut juridique	250		
	Association		2.5%	2.2%
	GIE		65%	84%
	Informel		22%	3.3%
	SA		2.5%	3.3%
	SARL		5.1%	5.4%
	SUARL		3.2%	2.2%
	propriétaire ou locataire	250		
	Locataire		9.5%	9.8%
	Propriétaire		91%	90%
	Age du dirigeant	250	54	57

Group	Characteristic	N[1]	Non, N = 158[1]	Oui, N = 92
	parle	250		
	Mean		2.17	3.04
	duree	250		
	Maximum		664 hours	383 hours
Femme	Niveau d'instruction	191		
	Aucun niveau		60 (53%)	10 (13%)
	Niveau primaire		22 (19%)	26 (34%)
	Niveau secondaire		28 (25%)	28 (36%)
	Niveau Superieur		4 (3.5%)	13 (17%)
	Statut juridique	191		
	Association		3 (2.6%)	0 (0%)
	GIE		76 (67%)	73 (95%)
	Informel		31 (27%)	1 (1.3%)
	SA		1 (0.9%)	0 (0%)
	SARL		1 (0.9%)	1 (1.3%)
	SUARL		2 (1.8%)	2 (2.6%)
	propriétaire ou locataire	191		
	Locataire		8 (7.0%)	8 (10%)
	Propriétaire		106 (93%)	69 (90%)
Homme	Niveau d'instruction	59		
	Aucun niveau		8 (18%)	1 (6.7%)
	Niveau primaire		8 (18%)	0 (0%)
	Niveau secondaire		14 (32%)	4 (27%)
	Niveau Superieur		14 (32%)	10 (67%)
	Statut juridique	59		
	Association		1 (2.3%)	2 (13%)

Group	Characteristic	N[1]	**Non**, N = 158[1]	**Oui**, N = 92
	GIE		26 (59%)	4 (27%)
	Informel		4 (9.1%)	2 (13%)
	SA		3 (6.8%)	3 (20%)
	SARL		7 (16%)	4 (27%)
	SUARL		3 (6.8%)	0 (0%)
	propriétaire ou locataire	59		
	Locataire		7 (16%)	1 (6.7%)
	Propriétaire		37 (84%)	14 (93%)

[1]%; Median

3 Un peu de cartographie}

Le code copie projet dans un nouvel objet nommé projet_map. Ensuite, il utilise le package sp pour définir les coordonnées spatiales de projet_map en utilisant les colonnes gps_menlongitude(longitude) et gps_menlatitude (latitude) du data frame projet.Enfin,il vérifie la classe de l'objet "projet_map" pour déterminerle type l'objet spatial représenté.

```
projet_map <- st_as_sf(projet, coords = c("gps_menlongitude",
"gps_menlatitude"))
class(projet_map)

## [1] "sf"         "tbl_df"     "tbl"        "data.frame"
```

récupérer les données géospatiales du Sénégal au niveau 0 d'administration

```
sen_region <- getData("GADM", country = "senegal", level = 1)
```

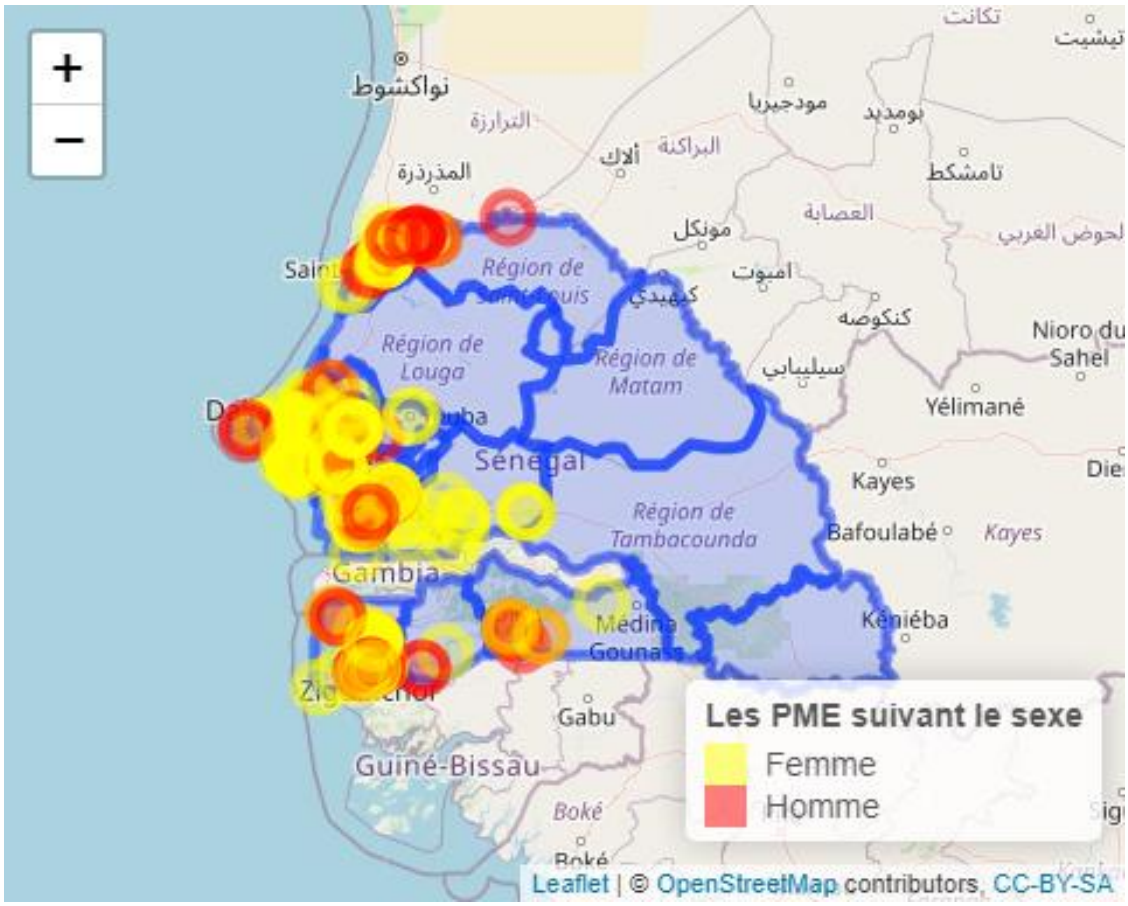
Création de la carte interactive avec des marqueurs de différentes couleurs

selon le sexe

```
m <- leaflet(sen_region) %>%
  addTiles() %>%
  addPolygons() %>%
  addCircleMarkers(
    data = projet_map,
    lat = ~st_coordinates(projet_map)[, 2],
    lng = ~st_coordinates(projet_map)[, 1],
    color = ~ifelse(sexe == "Femme", "yellow", "red"),  # Changer la couleur
en fonction du sexe
    label = ~departement
```

```
  )%>%
addLegend(
    "bottomright",   # Position de la légende (peut être "topright",
"topleft", "bottomright", ou "bottomleft")
    title = "Les PME suivant le sexe",   # Titre de la légende
    colors = c("yellow", "red"),   # Couleurs des marqueurs
    labels = c("Femme", "Homme")   # Étiquettes dans la légende
  )

# Afficher la carte
m
```

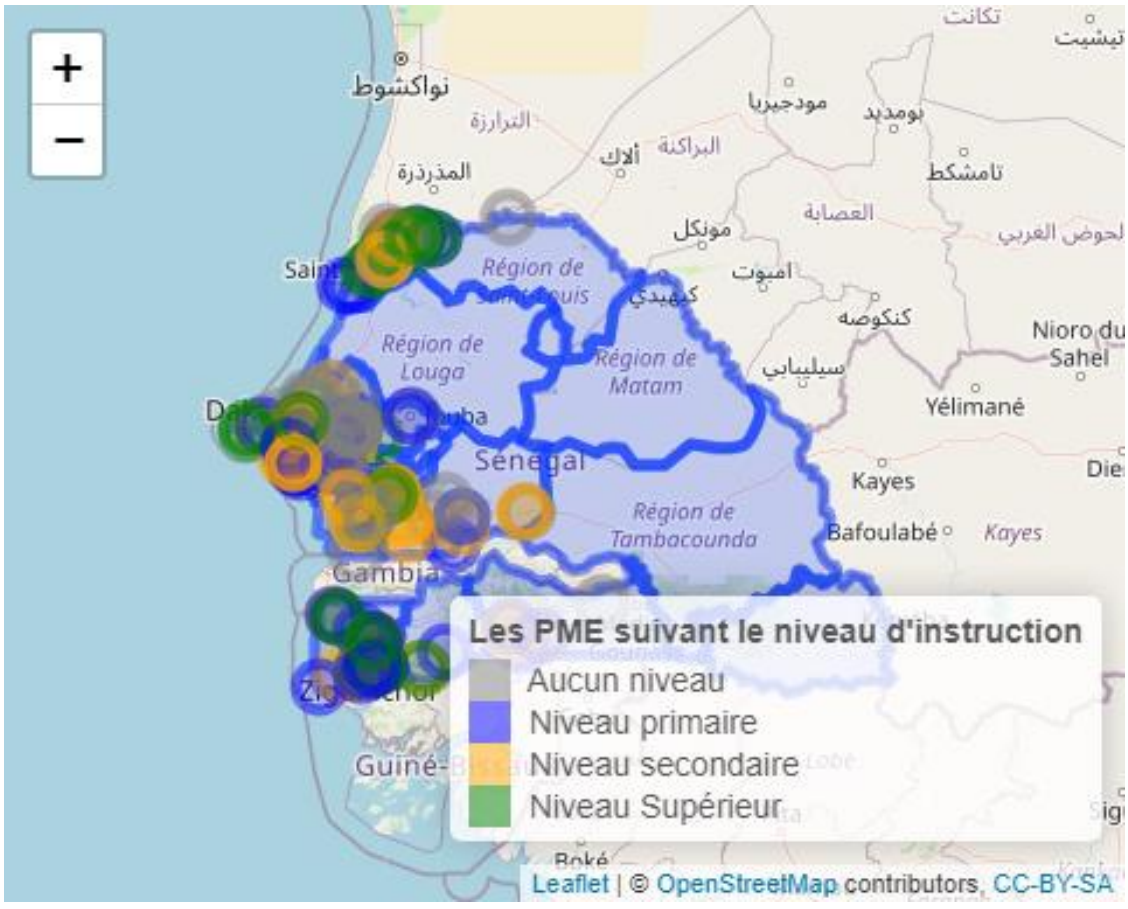


Création de la carte interactive avec des marqueurs de différentes couleurs **selon le niveau d'instruction**

```
m <- leaflet() %>%
  addTiles() %>%
  addPolygons(data = sen_region) %>%
  addCircleMarkers(
    data = projet_map,
    lat = ~st_coordinates(projet_map)[, 2],
    lng = ~st_coordinates(projet_map)[, 1],
    color = ~ifelse(q25 == "Aucun niveau", "gray",
```

```
                    ifelse(q25 == "Niveau primaire", "blue",
                           ifelse(q25 == "Niveau secondaire", "orange",
"green"))),
    label = ~departement
  ) %>%
  addLegend(
    "bottomright",   # Position de la légende
    title = "Les PME suivant le niveau d'instruction",  # Titre de la légende
    colors = c("gray", "blue", "orange", "green"),   # Couleurs des marqueurs
    labels = c("Aucun niveau", "Niveau primaire", "Niveau secondaire",
"Niveau Supérieur")   # Étiquettes dans la légende
  )


# Afficher la carte
m
```



On voit que les PME des femmes sont les plus nombreuses et que les PME se concentrent plus à Dakar et à Thies, un peu vers le nord et le sud du Sénégal. On constate aussi qu'à Dakar à Thies et le nord du Sénégal, on a a la presence de tous les niveaux d'instruction chez les dirigeants des PME mais les les dirigeants des PME des régions du sud-ouest ont généralement primaire et supeieur.

Partie 2

Nettoyage et gestion des données

```
# importation de la base
Base_Partie_2 <- readxl::read_excel("Base_Partie 2.xlsx")
# renommer country_destination" en "destination
names(Base_Partie_2)[names(Base_Partie_2) == "country_destination"] <-
"destination"
## Définir les valeurs négatives de la colonne "destination" comme manquantes
(NA)
Base_Partie_2$destination[Base_Partie_2$destination < 0] <- NA

# Créer une nouvelle variable "tranche_age" avec des tranches d'âge de 5 ans
(sous forme de texte)
Base_Partie_2$tranche_age <- cut(Base_Partie_2$age,
                                 breaks = seq(0, max(Base_Partie_2$age) +
5,
                                 by = 5), labels = FALSE, include.lowest =
TRUE)

# Convertir les identifiants numériques des tranches d'âge en intervalles
#de 5 ans sous forme de texte
Base_Partie_2$tranche_age_texte <- ifelse(is.na(Base_Partie_2$tranche_age),
                          "N/A", # Si l'âge est manquant, afficher "N/A"
                          paste0(Base_Partie_2$tranche_age * 5, "-",
                                 (Base_Partie_2$tranche_age * 5) + 4, "
ans"))
```

Dans ce code, nous utilisons la fonction group_by() du package "dplyr" pour regrouper les données par le numéro d'identification de l'enquêteur. Ensuite, nous utilisons la fonction mutate() pour ajouter une nouvelle variable "nombre_entretiens", qui contient le nombre d'entretiens réalisés par chaque enquêteur, calculé en utilisant la fonction n() qui renvoie le nombre de lignes dans chaque groupe

```
Base_Partie_2 <- Base_Partie_2 %>%
  group_by(enumerator) %>%
  mutate(nombre_entretiens = n())

# Fixer une graine (seed) pour la génération aléatoire, pour obtenir des
résultats
#reproductibles
set.seed(133)
# Créer une nouvelle variable "groupe_traitement" pour affecter aléatoirement
#les répondants à un groupe de traitement (1) ou de contrôle (0)
Base_Partie_2 <- Base_Partie_2 %>%
  group_by(id) %>%
  mutate(groupe_traitement = sample(c(0, 1), size = 1))
```

Le code est utilisé pour lire les données de la feuille “district” à partir d’un fichier Excel (“Base_Partie 2.xlsx”) et ensuite fusionner ces données avec un dataframe existant “Base_Partie_2” en utilisant la variable “district” comme clé de jointure

```
donnees_feuille_2 <- read_excel("Base_Partie 2.xlsx",
                                sheet = "district")
Base_Partie_2  <- merge(Base_Partie_2, donnees_feuille_2, by = "district",
all.x = TRUE)

# Calculer la durée de l'entretien en heures
Base_Partie_2 <- Base_Partie_2 %>%
  mutate(duree_entretien = as.numeric(endtime - starttime, units = "hours"))

# Calculer la durée moyenne de l'enquête par enquêteur

Base_Partie_2 <- Base_Partie_2 %>%
  group_by(enumerator) %>%
  mutate(duree_moy_enq = mean(duree_entretien, na.rm = TRUE))

# Récupérer les noms des colonnes de l'ensemble de données
colonne <- names(Base_Partie_2)

# Boucle pour renommer les colonnes avec le préfixe "endline_"
for (col in colonne) {
  newcolonne <- paste("endline_", col, sep = "")
  colnames(Base_Partie_2)[colnames(Base_Partie_2) == col] <- newcolonne
}
```

Analyse et visualisation des données

Tableau contenant l’âge moyen et le nombre moyen d’enfants par district

```
tabl<-Base_Partie_2%>%tbl_summary(include=c(endline_age,endline_children_num,
                                            endline_district),
                                  # Variables à inclure dans le tableau
                                  by=endline_district,percent = "column",
                                  # Regroupement par la variable
'endline_district'
                                type = list(endline_children_num =
"continuous"),
                                  statistic=list(endline_age~"{mean}",

endline_children_num~"{sum}"),
                                  # Statistique des variables

) %>% add_n()
tabl
```

Characteristic	N	**1**, N = 8[1]	**2**, N = 27[1]	**3**, N = 8[1]	**4**, N = 5[1]	**5**, N = 6[1]	**6**, N = 26[1]	**7**, N = 6[1]	**8**, N = 11[1]
endline_age	97	30	63	26	26	24	23	28	25
endline_children_num	97	12.00	23.00	0.00	0.00	3.00	3.00	1.00	14.00

[1]Mean; Sum

Test de Student

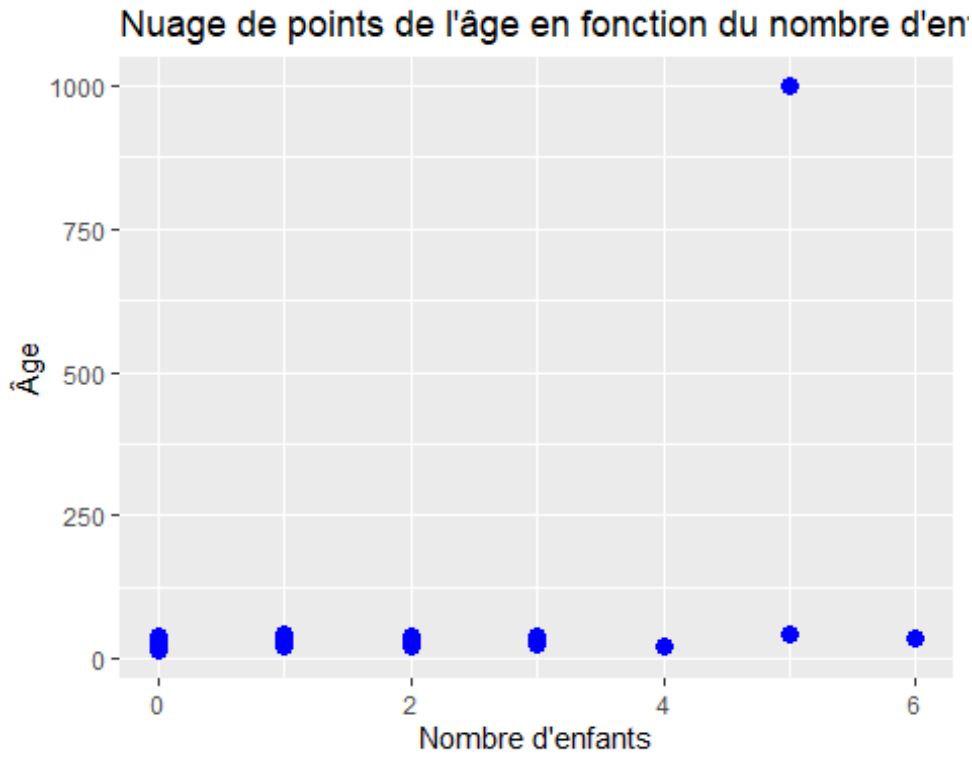
```
test <- t.test(Base_Partie_2$endline_age, Base_Partie_2$endline_sex)
print(test)
```

```
##
##  Welch Two Sample t-test
##
## data:  Base_Partie_2$endline_age and Base_Partie_2$endline_sex
## t = 3.5296, df = 96.002, p-value = 0.0006408
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  15.53332 55.45637
## sample estimates:
##  mean of x  mean of y
## 35.6082474  0.1134021
```

ll y a une différence statistiquement significative entre les âges pour les deux sexes car p-value < 5%.

Créez le nuage de points avec ggplot

```
ggplot(Base_Partie_2, aes(x = endline_children_num, y = endline_age)) +
geom_point(size = 3, color = "blue") +
  # Définir la taille et la couleur des points+
  labs(x = "Nombre d'enfants", y = "Âge") +
  ggtitle("Nuage de points de l'âge en fonction du nombre d'enfants")
```



Création du modèle de régression linéaire

```
# Création du modèle de régression linéaire
model <- lm(endline_intention ~ endline_groupe_traitement, data =
Base_Partie_2)
# Affichage des résultats de la régression linéaire
summary(model)

##
## Call:
## lm(formula = endline_intention ~ endline_groupe_traitement, data =
Base_Partie_2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1915 -1.1915 -1.0000  0.8085  5.0000
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  2.1915     0.2519   8.701 9.86e-14 ***
## endline_groupe_traitement  -0.1915     0.3508  -0.546    0.586
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.727 on 95 degrees of freedom
## Multiple R-squared:  0.003126,   Adjusted R-squared:  -0.007367
## F-statistic: 0.2979 on 1 and 95 DF,  p-value: 0.5865
```

```
#tableau
tbl_regression(model)
```

Characteristic	Beta	95% CI[1]	p-value
endline_groupe_traitement	-0.19	-0.89, 0.50	0.6

[1]CI = Confidence Interval

Les résultats de la régression linéaire suggèrent qu'il n'y a pas de preuve statistiquement significative d'un effet de l'appartenance au groupe de traitement sur l'intention de migrer (p-value> 5%)

Tableau de régression avec 3 modèles

```
# Modèle A : Modèle vide - Effet du traitement sur les intentions
model_A <- lm(endline_intention ~ endline_groupe_traitement, data =
Base_Partie_2)

# Modèle B : Effet du traitement sur les intentions en tenant compte de l'âge
et du sexe
model_B <- lm(endline_intention ~ endline_groupe_traitement + endline_age +
endline_sex, data = Base_Partie_2)

# Modèle C : Identique au modèle B mais en contrôlant le district
model_C <- lm(endline_intention ~ endline_groupe_traitement + endline_age +
endline_sex + endline_district, data = Base_Partie_2)

# Créez un objet tbl_regression pour chaque modèle
tbl_model_A <- tbl_regression(model_A)
tbl_model_B <- tbl_regression(model_B)
tbl_model_C <- tbl_regression(model_C)

# Combiner les trois tableaux dans un seul tableau en utilisant la fonction
tbl_merge
table_combine <- tbl_merge(list(tbl_model_A, tbl_model_B, tbl_model_C),
                           tab_spanner = c("model_A", "model_B","model_C"))

# Afficher le tableau combiné
table_combine
```

	model_A			model_B			model_C		
Characteristic	**Beta**	**95% CI[1]**	**p-value**	**Beta**	**95% CI[1]**	**p-value**	**Beta**	**95% CI[1]**	**p-value**
endline_groupe_traitement	-0.19	-0.89, 0.50	0.6	-0.31	-1.0, 0.40	0.4	-0.31	-1.0, 0.40	0.4
endline_age				0.00	0.00, 0.00	0.8	0.00	0.00, 0.00	>0.9

	model_A			model_B			model_C		
Characteristic	**Beta**	**95% CI[1]**	**p-value**	**Beta**	**95% CI[1]**	**p-value**	**Beta**	**95% CI[1]**	**p-value**
endline_sex				-0.98	-2.1, 0.17	0.095	-0.92	-2.1, 0.23	0.11
endline_district							0.08	-0.07, 0.24	0.3

[1]CI = Confidence Interval