République du Sénégal

Un peuple - Un but - Une foi

Agence Nationale de la Statistique et de la Démographie (ANSD)

Ecole Nationale de la Statistique et de l'Analyse Économique (ENSAE)

PROJET STATISTIQUE SUR R ET PYTHON

RENDU DU PROJET DE COURS

Rédigé par :
SOSSOU Toussaint Régis
Élève Ingénieur Statisticien Économiste

Sous la supervision de :
M. HEMA Aboubacar
Analyst researcher

Juillet 2023

Contents

PARTIE 1

I. Préparation des données

I.1 Description de la base

I.2 Importation et mise en forme

```
#Importation des librairies nécessaires
library("readxl")
library("gtsummary")
library("flextable")
library("dplyr")
library("ggplot2")
library("sf")
library("leaflet")
library("DT")
library("ggspatial")
library("webshot")
library("gt")
```

Importation des bibliothèques nécessaires

```
#Importation de la base de données dans un objet de type data.frame nommé projet
projet<-read_xlsx("Base_Partie 1.xlsx")
```

Importation de la base de données dans un objet de type data.frame nommé projet

```
#Tableau résumant les valeurs manquantes par variable
flextable::flextable(data.frame(variables=colnames(projet),
                                'valeurs manquantes'=apply(projet, MARGIN = 2,
                                                           function(x) sum(is.na(x)))))
```

Tableau résumant les valeurs manquantes par variable

variables	valeurs.manquantes
key	0
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

variables	valeurs.manquantes
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

```
#apply(projet, MARGIN = 2, function(x) sum(is.na(x)) calcule les valeurs manquantes
#par variables et flextable vient associer au nom de chaque variable sa valeur manquante.
```

```
if (all(!is.na(projet$key))) {
  cat("La variable Key ne possède pas de valeurs manquantes")
} else {
  cat(paste("Le nombre de valeurs manquantes de la variable key est",
              sum(is.na(projet$key))))
  cat("\n Les types de PME concernées sont : \n",
      paste(projet$q12[is.na(projet$key)]), sep=" ")
}
```

Vérification de la contenance pour la variable key de données manquantes dans la base projet et identification le cas échéant du statut juridique de la (ou des) PME concernées.

```
## La variable Key ne possède pas de valeurs manquantes
```

I.3 Création de variables

```
if(is.na(match("q1", colnames(projet)))){
  cat("la base 'Projet' ne contient pas la variable 'q1'")
}else{
  projet <- projet %>%
  rename(region = q1)
  cat("Variable 'q1' renommée en 'région'")
}
```

Renommons la variable q1 en region

```
## Variable 'q1' renommée en 'région'
```

```
if(is.na(match("q2", colnames(projet)))){
  print("la base 'Projet' ne contient pas la variable 'q2'")
}else{
  projet <- projet %>%
  rename(departement = q2)
  print("Variable 'q2' renommée en 'département'")
}
```

Renommons la variable q2 en departement

```
## [1] "Variable 'q2' renommée en 'département'"
```

```
if(is.na(match("q23", colnames(projet)))){
  cat("la base 'Projet' ne contient pas la variable 'q23'")
}else{
  projet <- projet %>%
  rename(sexe = q23)
  cat("Variable 'q23' renommée en 'sexe'")
}
```

Renommons la variable q23 en sexe

```
## Variable 'q23' renommée en 'sexe'
```

```
#code de questionr::irec()
# projet$sexe_2 <- projet$sexe
# projet$sexe_2[projet$sexe == "Femme"] <- "1"
# projet$sexe_2[projet$sexe == "Homme"] <- "0"
# projet$sexe_2 <- as.numeric(projet$sexe_2)
if(is.na(match("sexe",colnames(projet)))){
  cat(" la base 'projet' ne contient pas la variable 'sexe'")
}else{
  projet$sexe_2 <- ifelse(projet$sexe == "Femme", 1, 0)
  cat("Création de la variable sexe éffectué avec succès.")
}
```

Création de la variable sexe_2 qui vaut 1 si sexe égale à Femme et 0 sinon.

```
## Création de la variable sexe éffectué avec succès.
```

```r
langues<-projet[c("key",grep("^q24a_", colnames(projet), value = TRUE))]
#la fonction grep() va aller lire dans les variables du dataframe projet
# les variables dont le nom commence par "q24a_" et l'on y ajoute la variable
#key avec le c() et enfin, on passe en paramètre le resultat au dataframe projet
# Cela permet de créer un nouveau dataframe "langues" avec les colonnes d'intérêt
```

Création d'un data.frame nommé langues qui prend les variables key et les variables correspondantes décrites plus haut.

```r
# La nouvelle colonne contient la somme des valeurs de chaque ligne pour les
#colonnes qui commencent par "q24a_" dans le dataframe "langues"
langues$parle <- rowSums(langues[, grepl("^q24a_", colnames(langues))])
```

Création d'une variable parle qui est égale au nombre de langue parlée par le dirigeant de la PME.

```r
#ici on va écraser le dataframe langue en conservant uniquement les deux
# variables key et parle
langues<-langues[,c("key", "parle")]
```

Sélection unique des variables key et parle, et affectation au dataframe langues.

```r
projet<-base::merge(projet, langues, by = "key", all = TRUE)
```

Merge des data.frame projet et langues

II. Analyses descriptives

Répartion des PME suivant le sexe.

```r
gtsummary::tbl_summary(projet, include=sexe) %>% modify_header(label~"**Total**"
                                                    ) %>% modify_caption(
                                                      caption = "**Répartition des PME suivant le sexe*
```

Table 2: **Répartition des PME suivant le sexe**

Total	**N = 250**
sexe	
Femme	191 (76%)
Homme	59 (24%)

Répartion des PME suivant le niveau d'instruction.

```r
gtsummary::tbl_summary(projet, include=q25, label = list(q25~"Niveau d'instruction")
            ) %>% modify_header(label~"**Total**") %>% modify_caption(
              caption = "**Répartition des PME suivant
              le niveau d'instruction**")
```

Table 3: **Répartition des PME suivant le niveau d'instruction**

Total	**N = 250**
Niveau d'instruction	
Aucun niveau	79 (32%)
Niveau primaire	56 (22%)
Niveau secondaire	74 (30%)
Niveau Superieur	41 (16%)

Répartion des PME suivant le statut juridique.

```
gtsummary::tbl_summary(projet, include=q12, label = list(q12~"Statut juridique")
            ) %>% modify_header(label~"**Total**") %>% modify_caption(
              caption = "**Répartition des PME suivant
              le statut juridique**")
```

Table 4: **Répartition des PME suivant le statut juridique**

Total	**N = 250**
Statut juridique	
Association	6 (2.4%)
GIE	179 (72%)
Informel	38 (15%)
SA	7 (2.8%)
SARL	13 (5.2%)
SUARL	7 (2.8%)

Répartion des PME suivant le propriétaire/locataire.

```
gtsummary::tbl_summary(projet, include=q81, label = list(q81~"propriétaire/locataire")
            )%>% modify_header(label~"**Total**"
                               ) %>%
  modify_caption(caption = "**Répartition
                 des PME suivant le type de logement**")
```

Table 5: **Répartition des PME suivant le type de logement**

Total	**N = 250**
propriétaire/locataire	
Locataire	24 (9.6%)
Propriétaire	226 (90%)

Répartion des PME suivant le statut juridique et le sexe.

```
gtsummary::tbl_cross(projet,
    row =  q12,
    col = sexe,
    percent = "row",
    label = list(q12~"Statut juridique",sexe~"sexe")
```

```
  ) %>% modify_caption(caption =
                         "**Répartition des PME suivant
                       le statut juridique et le sexe**")
```

Table 6: **Répartition des PME suivant le statut juridique et le sexe**

	Femme	Homme	Total
Statut juridique			
Association	3 (50%)	3 (50%)	6 (100%)
GIE	149 (83%)	30 (17%)	179 (100%)
Informel	32 (84%)	6 (16%)	38 (100%)
SA	1 (14%)	6 (86%)	7 (100%)
SARL	2 (15%)	11 (85%)	13 (100%)
SUARL	4 (57%)	3 (43%)	7 (100%)
Total	191 (76%)	59 (24%)	250 (100%)

Répartion des PME suivant le niveau d'instruction et le sexe.

```
gtsummary::tbl_cross(projet,
    row =  q25,
    col = sexe,
    percent = "row",
    label = list(q25~"Niveau d'instruction",sexe~"**sexe**")
  ) %>% modify_caption(caption =
                         "**Répartition des PME suivant
                       le niveau d'instruction et le sexe**")
```

Table 7: **Répartition des PME suivant le niveau d'instruction et le sexe**

	Femme	Homme	Total
Niveau d'instruction			
Aucun niveau	70 (89%)	9 (11%)	79 (100%)
Niveau primaire	48 (86%)	8 (14%)	56 (100%)
Niveau secondaire	56 (76%)	18 (24%)	74 (100%)
Niveau Superieur	17 (41%)	24 (59%)	41 (100%)
Total	191 (76%)	59 (24%)	250 (100%)

Répartion des PME suivant le Propriétaire/locataire et le sexe.

```
gtsummary::tbl_cross(projet,
    row =  q81,
    col = sexe,
    percent = "row",
    label = list(q81~"Propriétaire/locataire",sexe~"**sexe**")
  ) %>% modify_caption(caption =
                         "**Répartition des PME suivant
                       le niveau d'instruction et le sexe**")
```

Table 8: **Répartition des PME suivant le niveau d'instruction et le sexe**

	Femme	Homme	Total
Propriétaire/locataire			
Locataire	16 (67%)	8 (33%)	24 (100%)
Propriétaire	175 (77%)	51 (23%)	226 (100%)
Total	191 (76%)	59 (24%)	250 (100%)

Statistiques descriptives de notre choix sur les autres variables (à définir).

```
# Répartition des PME suivant le sexe, le niveau d'instruction, le statut juridique et le propriétaire

tbl1 <- projet %>% gtsummary::tbl_summary(include = c("sexe", "q25", "q12", "q81"))
# Répartition des PME suivant le statut juridique et le sexe,   le niveau d'instruction et le sexe, •
tbl2 <- projet %>% tbl_summary(
  include = c("q25", "q12", "q81"),
  by ="sexe", label=list(q12~ "Statut juridique",
                         q25~ "Niveau d'instruction",
                         q81~ "Propriétaire/locataire")) %>%
  add_overall() %>%
  modify_header(label ~ "**Caractéristiques**") %>%
  modify_spanning_header(c("stat_1", "stat_2") ~ "**Sexe**")

## Empillement l'un sur l'autre

gtsummary::tbl_stack(
  list(tbl1, tbl2),
  group_header = c("Modèle univarié", "Modèle bivaré") ## intitulé des groupes de tableau associés
)
```

Group	**Characteristic**	**N = 250**	**Femme**, N = 191	**Homme**, N = 59
Modèle univarié	sexe			
	Femme	191 (76%)		
	Homme	59 (24%)		
	q25			
	Aucun niveau	79 (32%)		
	Niveau primaire	56 (22%)		
	Niveau secondaire	74 (30%)		
	Niveau Superieur	41 (16%)		
	q12			
	Association	6 (2.4%)		
	GIE	179 (72%)		
	Informel	38 (15%)		
	SA	7 (2.8%)		
	SARL	13 (5.2%)		
	SUARL	7 (2.8%)		
	q81			
	Locataire	24 (9.6%)		
	Propriétaire	226 (90%)		
Modèle bivaré	Niveau d'instruction			
	Aucun niveau	79 (32%)	70 (37%)	9 (15%)

Group	Characteristic	N = 250	Femme, N = 191	Homme, N = 59
	Niveau primaire	56 (22%)	48 (25%)	8 (14%)
	Niveau secondaire	74 (30%)	56 (29%)	18 (31%)
	Niveau Superieur	41 (16%)	17 (8.9%)	24 (41%)
	Statut juridique			
	Association	6 (2.4%)	3 (1.6%)	3 (5.1%)
	GIE	179 (72%)	149 (78%)	30 (51%)
	Informel	38 (15%)	32 (17%)	6 (10%)
	SA	7 (2.8%)	1 (0.5%)	6 (10%)
	SARL	13 (5.2%)	2 (1.0%)	11 (19%)
	SUARL	7 (2.8%)	4 (2.1%)	3 (5.1%)
	Propriétaire/locataire			
	Locataire	24 (9.6%)	16 (8.4%)	8 (14%)
	Propriétaire	226 (90%)	175 (92%)	51 (86%)

```
## Agencement l'une à coté de l'autre
gtsummary::tbl_merge(
  list(tbl1, tbl2),
  tab_spanner = c("Modèle bivarié", "Modèle multivarié") ## intitulé des groupes de tableau associés
)
```

Characteristic	N = 250	Overall, N = 250	Femme, N = 191	Homme, N = 59
sexe				
Femme	191 (76%)			
Homme	59 (24%)			
q25				
Aucun niveau	79 (32%)			
Niveau primaire	56 (22%)			
Niveau secondaire	74 (30%)			
Niveau Superieur	41 (16%)			
q12				
Association	6 (2.4%)			
GIE	179 (72%)			
Informel	38 (15%)			
SA	7 (2.8%)			
SARL	13 (5.2%)			
SUARL	7 (2.8%)			
q81				
Locataire	24 (9.6%)			
Propriétaire	226 (90%)			
Niveau d'instruction				
Aucun niveau		79 (32%)	70 (37%)	9 (15%)
Niveau primaire		56 (22%)	48 (25%)	8 (14%)
Niveau secondaire		74 (30%)	56 (29%)	18 (31%)
Niveau Superieur		41 (16%)	17 (8.9%)	24 (41%)
Statut juridique				
Association		6 (2.4%)	3 (1.6%)	3 (5.1%)
GIE		179 (72%)	149 (78%)	30 (51%)
Informel		38 (15%)	32 (17%)	6 (10%)
SA		7 (2.8%)	1 (0.5%)	6 (10%)
SARL		13 (5.2%)	2 (1.0%)	11 (19%)
SUARL		7 (2.8%)	4 (2.1%)	3 (5.1%)

Characteristic	**N = 250**	**Overall**, N = 250	**Femme**, N = 191	**Homme**, N = 59
Propriétaire/locataire				
Locataire		24 (9.6%)	16 (8.4%)	8 (14%)
Propriétaire		226 (90%)	175 (92%)	51 (86%)

```
# # Statistiques descriptives
#
# gtsummary::tbl_stack(
#   list(tbl_filiere_1 , tbl_filiere_2 , tbl_filiere_3 , tbl_filiere_4),
#   group_header = c("arachide", "anacarde", "mangue", "riz") ## intitulé des groupes de tableau associ
# )
#
# ## Agencement l'une à coté de l'autre
# gtsummary::tbl_merge(
#   list(tbl_filiere_1, tbl_filiere_2, tbl_filiere_3, tbl_filiere_4),
#   tab_spanner = c("arachide", "anacarde", "mangue", "riz") ## intitulé des groupes de tableau associé
# )
```

III. Un peu de cartographie

Transformation du data.frame en données géographiques dont l'objet sera nommé projet_map.

```
# Jointure spatiale entre les données du projet et les données géospatiales du Sénégal
# 1. Lecture des données géospatiales du Sénégal avec la fonction st_read()
# et spécification du système de coordonnées de référence (CRS)
#senegal <- st_read("gadm41_SEN_1.shp")
# 2. Conversion des données du projet en un objet spatial "sf"
# en utilisant la fonction st_as_sf()
# - Les colonnes "gps_menlongitude" et "gps_menlatitude" contiennent les coordonnées spatiales
# - Le CRS est spécifié à l'aide de la fonction st_crs() pour correspondre au CRS des données du Sénéga
# - Les données projet sont transformées en objet spatial "sf" avec st_as_sf()
# projet_sf <- st_as_sf(projet,
#                       coords = c("gps_menlongitude", "gps_menlatitude"),
#                       crs = st_crs(senegal))
# 3. Jointure spatiale entre les objets spatiaux "projet_sf" et "senegal"
# en utilisant la fonction st_join()
# - La fonction st_join() effectue la jointure spatiale entre les polygones du Sénégal (senegal)
#   et les points du projet (projet_sf) en attribuant à chaque point son emplacement spatial
#   en fonction de la région du Sénégal dans laquelle il se trouve
#code final
projet_map <- st_join(st_as_sf(projet,
                     coords = c("gps_menlongitude", "gps_menlatitude"),
                     crs=st_crs(st_read("gadm41_SEN_1.shp"))),
                     st_read("gadm41_SEN_1.shp"))
```

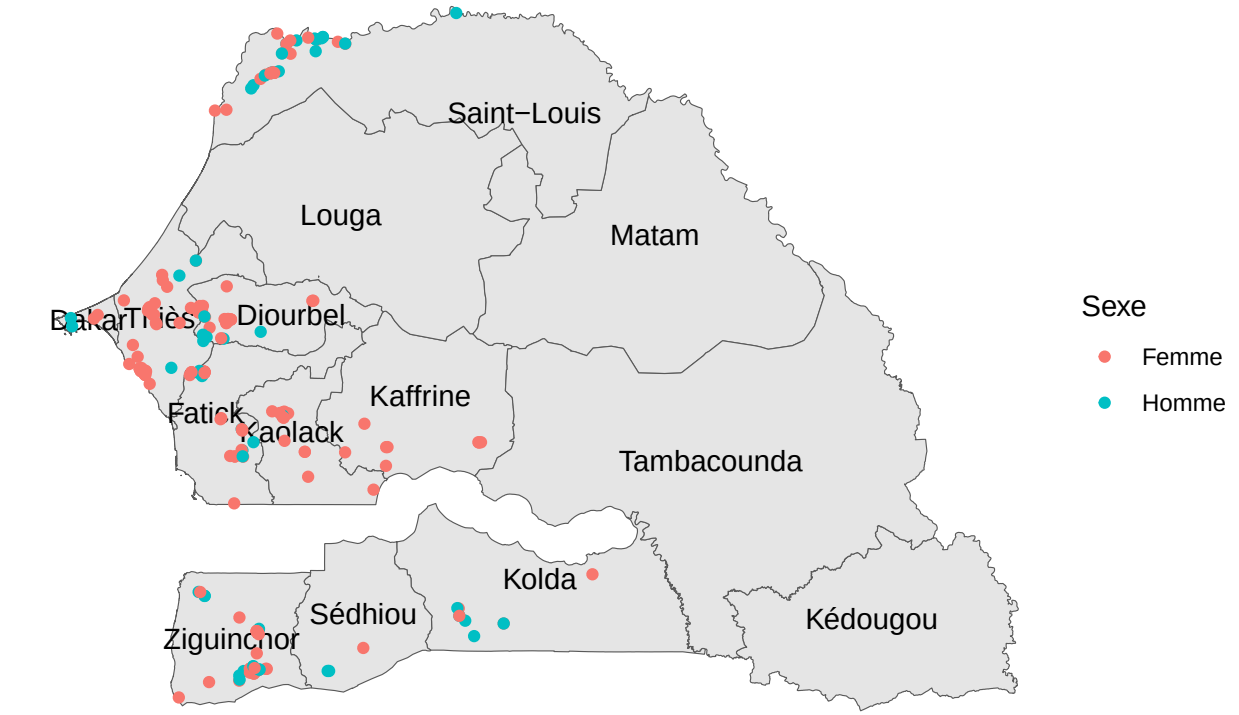
```
## Reading layer `gadm41_SEN_1' from data source
##   `C:\Users\starlab\Desktop\Projet_R\gadm41_SEN_1.shp' using driver `ESRI Shapefile'
## Simple feature collection with 14 features and 11 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS:  WGS 84
## Reading layer `gadm41_SEN_1' from data source
```

```
##   `C:\Users\starlab\Desktop\Projet_R\gadm41_SEN_1.shp' using driver `ESRI Shapefile'
## Simple feature collection with 14 features and 11 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS:  WGS 84
```

Représentation spatiale des PME suivant le sexe.

```
ggplot() +
  geom_sf(data=st_read("gadm41_SEN_1.shp"))+
  geom_sf_text(data=st_read("gadm41_SEN_1.shp"), aes(label=NAME_1))+
  geom_sf(data=projet_map, aes(color=sexe), size=1.5)+
  labs(title = "Repartition des PME suivant le sexe",
       subtitle = "Carte du Sénégal",
       color = "Sexe", x = NULL, y = NULL) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 1),
    plot.subtitle = element_text(hjust = 1))+
  theme_void()
```

```
## Reading layer `gadm41_SEN_1' from data source
##   `C:\Users\starlab\Desktop\Projet_R\gadm41_SEN_1.shp' using driver `ESRI Shapefile'
## Simple feature collection with 14 features and 11 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS:  WGS 84
## Reading layer `gadm41_SEN_1' from data source
##   `C:\Users\starlab\Desktop\Projet_R\gadm41_SEN_1.shp' using driver `ESRI Shapefile'
## Simple feature collection with 14 features and 11 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS:  WGS 84
```



Représentation spatiale des PME suivant le niveau d'instruction.

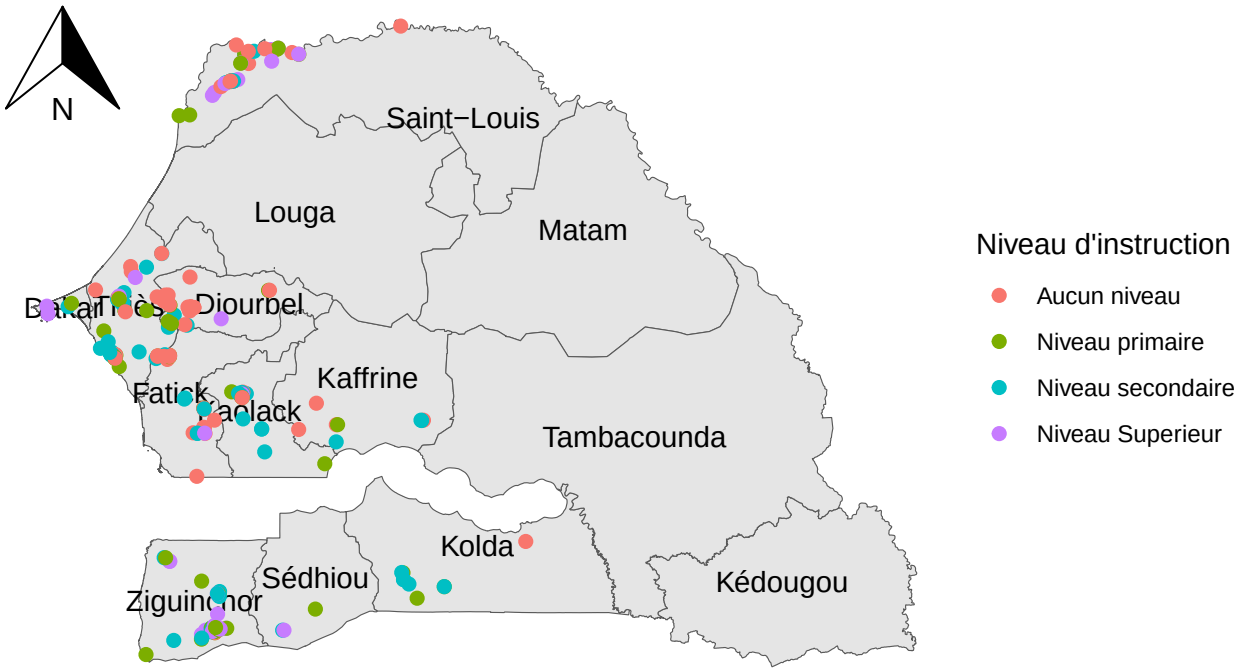
```
ggplot() +
  geom_sf(data=st_read("gadm41_SEN_1.shp"))+
  geom_sf_text(data=st_read("gadm41_SEN_1.shp"), aes(label=NAME_1))+
  geom_sf(data=projet_map, aes(color=q25), size=2)+
  labs(title = "Repartition des PME suivant le niveau d'instructions",
       subtitle = "Carte du Sénégal",
       color = "Niveau d'instruction",x = NULL, y = NULL) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 1),
    plot.subtitle = element_text(hjust = 1))+
  theme_void()+
  annotation_north_arrow(location = "tl", scale = 0.05)
```

```
## Reading layer `gadm41_SEN_1' from data source
##   `C:\Users\starlab\Desktop\Projet_R\gadm41_SEN_1.shp' using driver `ESRI Shapefile'
## Simple feature collection with 14 features and 11 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS:  WGS 84
## Reading layer `gadm41_SEN_1' from data source
##   `C:\Users\starlab\Desktop\Projet_R\gadm41_SEN_1.shp' using driver `ESRI Shapefile'
## Simple feature collection with 14 features and 11 fields
```

```
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS:  WGS 84
```

Repartition des PME suivant le niveau d'instructions

Carte du Sénégal



Analyse spatiale de notre choix : Analyse spatiale de l'état des routes passant devant les entreprises .

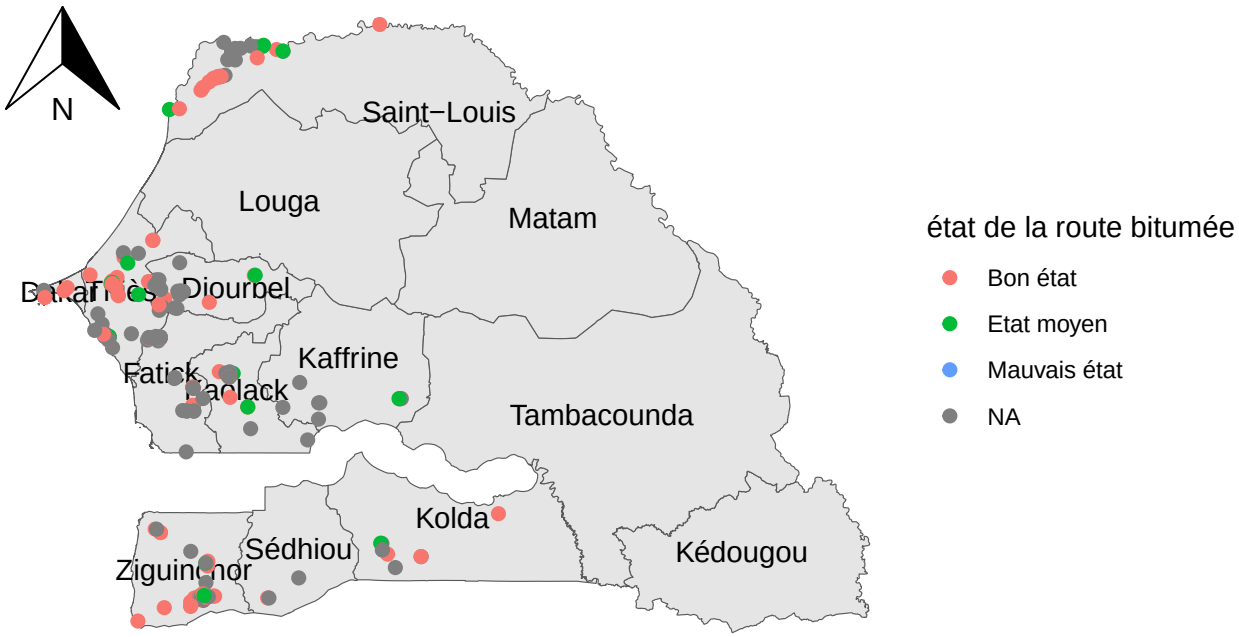
```
ggplot() +
  geom_sf(data=st_read("gadm41_SEN_1.shp"))+
  geom_sf_text(data=st_read("gadm41_SEN_1.shp"), aes(label=NAME_1))+
  geom_sf(data=projet_map, aes(color=q17), size=2)+
  labs(title = "état de la route bitumée passant devant les PME ",
       subtitle = "Carte du Sénégal",
       color = "état de la route bitumée",x = NULL, y = NULL) +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 1),
    plot.subtitle = element_text(hjust = 1))+
  theme_void()+
  annotation_north_arrow(location = "tl", scale = 5)
```

```
## Reading layer `gadm41_SEN_1' from data source
##   `C:\Users\starlab\Desktop\Projet_R\gadm41_SEN_1.shp' using driver `ESRI Shapefile'
## Simple feature collection with 14 features and 11 fields
## Geometry type: MULTIPOLYGON
```

```
## Dimension:     XY
## Bounding box:  xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS:  WGS 84
## Reading layer `gadm41_SEN_1' from data source
##   `C:\Users\starlab\Desktop\Projet_R\gadm41_SEN_1.shp' using driver `ESRI Shapefile'
## Simple feature collection with 14 features and 11 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS:  WGS 84
```

état de la route bitumée passant devant les PME

Carte du Sénégal



Partie 2

I. Nettoyage et gestion des données.

```
#Importation de la base Base_Partie 2.xlsx
partie2<-readxl::read_xlsx("Base_Partie 2.xlsx")
```

Renommons la variable "country_destination" en "destination" et définissons les valeurs négatives comme manquantes.

```
# Renommons la variable "country_destination" en "destination"
partie2 <- rename(partie2, destination = country_destination)
#Remplacement par des valeurs manquantes
partie2$destination <- ifelse(partie2$destination < 0, NA, partie2$destination)
```

Création d'une nouvelle variable contenant les tranches d'âge de 5 ans en utilisant la variable "age".

La base de donnée contenant des valeurs abbérantes pour la variable age (999), nous imputons à toutes les valeurs supérieures à 100 la médiane des ages.

```
#Imputation aux valeurs abbérantes de la variable age de la médiane de la série
median_age <- median(subset(partie2, age<100)$age)
partie2 <- partie2%>%
  mutate(age = ifelse(age < 0 | age > 100, median(subset(partie2, age>100)$age), age))
#suppressions de l'objet après utilisation
rm(median_age)
```

Création de la variable proprement dite

```
# Création de la nouvelle variable "age_group" en découpant la variable "age" en tranches de 5 ans
age_intervals <- seq(0, 100, by = 5)
age_labels <- paste0("]", age_intervals[-length(age_intervals)], ", ", age_intervals[-1], "]")

partie2 <- partie2 %>%
  mutate(age_interval = cut(age, breaks = age_intervals, labels = age_labels, right = FALSE))
#supression des objets après utilisation
rm(age_intervals, age_intervals)
```

Création d'une nouvelle variable contenant le nombre d'entretiens réalisés par chaque agent recenseur

```
# Variable nombre d'entretien par agent recenceur
partie2<- partie2 %>% group_by(enumerator) %>% mutate(nbr_entretien = n())
#  retirons les regroupements temporaires effectués par group_by()
partie2 <- ungroup(partie2)
```

Création d'une nouvelle variable qui affecte aléatoirement chaque répondant à un groupe de traitement (1) ou de controle (0).

```
#création de la variable aléatoire
partie2<- partie2 %>% mutate(group_traitement =sample(0:1, nrow(partie2),
                                                      replace = TRUE) )
```

Fusion de 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.

```
# fusion avec la taille de la population
# chargement la feuille disrict
district <- read_excel("Base_Partie 2.xlsx",sheet = "district")
fusion<-merge(partie2, district , by="district")
```

Calcul de la durée de l'entretien et indiquer la durée moyenne de l'entretien par enquêteur.

```
# Convertion des colonnes en format de date et heure
fusion$starttime <- as.POSIXct(fusion$starttime)
fusion$endtime <- as.POSIXct(fusion$endtime)
# Calcul de la durée de l'entretien en minutes
```

```r
fusion$duree_entretien <- as.numeric(difftime(fusion$endtime, fusion$starttime, units = "mins"))

# Calcul de la durée moyenne de l'entretien par enquêteur
duree_moy_p_enqueteur <- tapply(fusion$duree_entretien, fusion$enumerator, mean)
```

Renommez toutes les variables de l'ensemble de données en ajoutant le préfixe "endline_" à l'aide d'une boucle.

```r
# Renommons les variables avec le préfixe "endline_"
newb<-fusion
#Définition de la fonction de renommage et application grâce à un apply
newb <- lapply(names(newb), function(var) {
  if (!startsWith(var, "endline_")) {
    names(newb)[names(newb) == var] <- paste0("endline_", var)
  }
  return(newb[[var]])
})
# Convertion la liste en un nouveau data frame avec les variables renommées
newb <- as.data.frame(newb)
```

II. Analyse et visualisation des données.

Création d'un tableau récapitulatif contenant l'âge moyen et le nombre moyen d'enfants par district.

```r
# Calcul du 1er et le 3e quartile de la variable ma_variable
Q1 <- quantile(partie2$age, 0.25)
Q3 <- quantile(partie2$age, 0.75)

# Calcul de l'écart interquartile (IQR)
IQR <- Q3 - Q1

# Définition  des seuils pour exclure les valeurs aberrantes (par exemple, 1,5 fois l'IQR)
seuil_superieur <- Q3 + 1.5 * IQR
seuil_inferieur <- Q1 - 1.5 * IQR

# Filtrer les données pour exclure les valeurs aberrantes
donne_filtre <- partie2 %>% filter(age >= seuil_inferieur, age <= seuil_superieur)

tableau_1 <- donne_filtre %>%dplyr::select(district,age,children_num)%>%
  gtsummary::tbl_continuous(variable =age,
                            statistic =~ "{mean}",
                            #digits = 0,
                            include=district)
tableau_2 <- donne_filtre %>%dplyr::select(district,age,children_num)%>%
  gtsummary::tbl_continuous(variable =children_num,
                            statistic =~ "{mean}",
                            #digits = 0,
                            include=district)
tableau_recapitulatif<-tbl_merge(list(tableau_1,tableau_2),
                                 tab_spanner=c("Age Moyen des enfants","Nombre Moyen d'enfant"))%>%
  modify_caption("Age moyen & nombre moyen d'enfants par district.")%>% bold_labels()
tableau_recapitulatif
```

Table 11: Age moyen & nombre moyen d'enfants par district.

Characteristic	N = 94	N = 94
district		
1	27.7	1.00
2	26.6	0.69
3	26.1	0.00
4	26.0	0.00
5	24.3	0.50
6	23.2	0.12
7	25.2	0.00
8	24.6	1.27

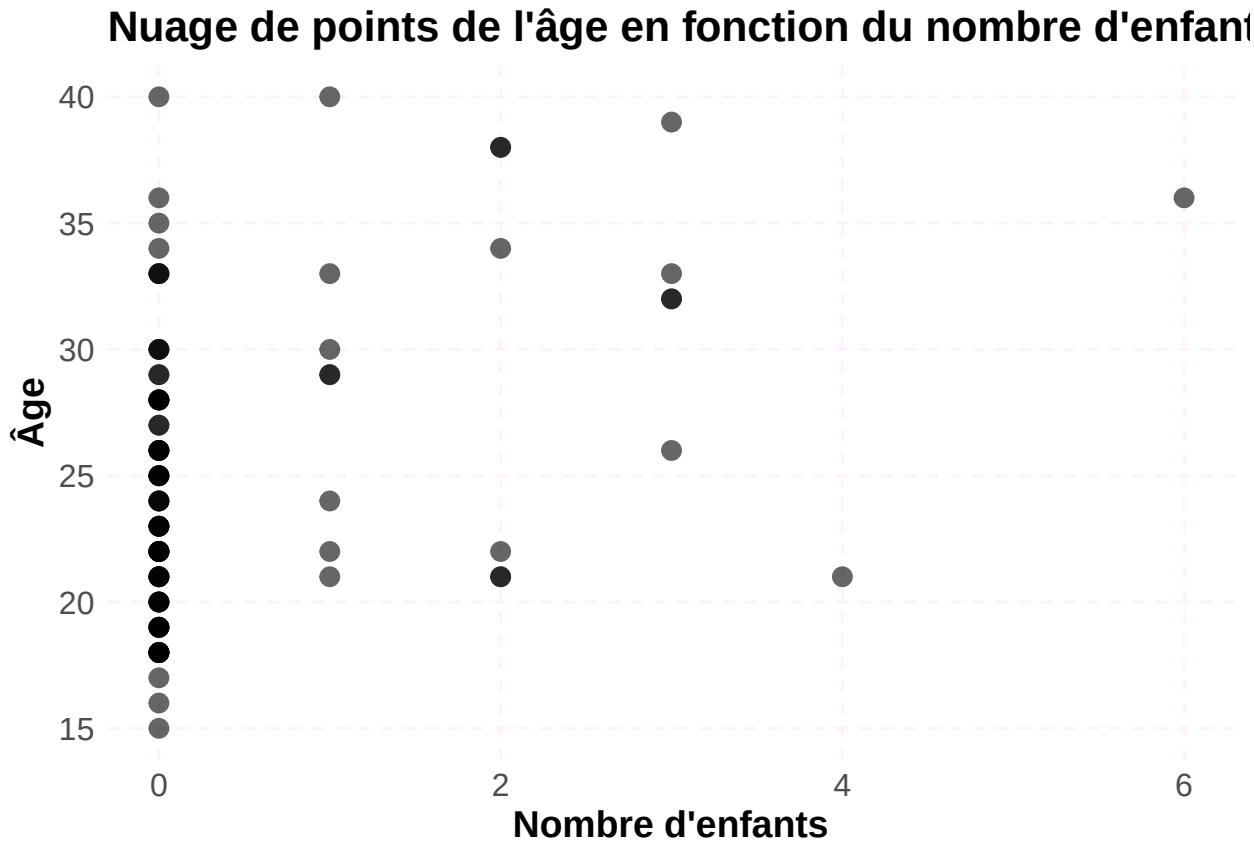
Test de si la différence d'âge entre les sexes est statistiquement significative au niveau de 5 %.

```
data_test<-partie2%>% select(sex, intention)
# Effectuons l'analyse de régression linéaire
modele <- lm(intention ~ sex, data = data_test)

# Afficher les résultats du modèle
modele%>%gtsummary::tbl_regression()
```

Characteristic	Beta	95% CI	p-value
sex	-0.92	-2.0, 0.16	0.093

Création d'un nuage de points de l'âge en fonction du nombre d'enfants.

```
# Création du nuage de points avec des effets visuels
nuage_points <- ggplot(data = donne_filtre, aes(x = children_num, y = age)) +
  geom_point(color = "black", size = 3, alpha = 0.6) +  # Couleur, taille et transparence des points
  labs(title = "Nuage de points de l'âge en fonction du nombre d'enfants",
       x = "Nombre d'enfants",
       y = "Âge") +
  theme_minimal() +  # Utiliser un thème minimal
  theme(panel.grid.major = element_line(color = "#FFF0FF", linetype = "dashed"),  # Ajouter une grille
        panel.grid.minor = element_blank(),  # Masquer les lignes de la grille secondaire
        axis.text = element_text(size = 12),  # Taille du texte des étiquettes d'axe
        axis.title = element_text(size = 14, face = "bold"),  # Taille et style du texte des titres d'a
        plot.title = element_text(size = 16, face = "bold"),  # Taille et style du titre du graphique
        legend.position = "bottom")  # Positionner la légende en bas du graphique

# Afficher le nuage de points
print(nuage_points)
```



Estimation de l'effet de l'appartenance au groupe de traitement sur l'intention de migrer.

```
data_test<-partie2%>% dplyr::select(intention,group_traitement)
# Effectuer l'analyse de régression linéaire
modele <- lm(intention ~ group_traitement, data = data_test)

# Afficher les résultats du modèle
modele%>%gtsummary::tbl_regression()
```

Characteristic	Beta	95% CI	p-value
group_traitement	-0.02	-0.72, 0.68	>0.9

Création d'un tableau de régression avec 3 modèles. La variable de résultat est toujours "intention". Modèle A : Modèle vide - Effet du traitement sur les intentions. Modèle B : Effet du traitement sur les intentions en tenant compte de l'âge et du sexe. Modèle C : Identique au modèle B mais en contrôlant le district. Les résultats des trois modèles doivent être affichés dans un seul tableau.

```
data_test<-partie2%>% select(intention,group_traitement,age,sex,district)
# Création des modèles de régression
modele_A <- lm(intention ~ group_traitement, data = data_test)
modele_B <- lm(intention ~ group_traitement + age + sex, data = data_test)
modele_C <- lm(intention ~ group_traitement + age + sex + district,
               data = data_test)
```

```r
# Création du tableau de régression avec gtsummary
tA<-modele_A%>%tbl_regression(exponentiate = FALSE)
tB<-modele_B%>%tbl_regression(exponentiate = FALSE)
tC<-modele_C%>%tbl_regression(exponentiate = FALSE)
tableau_regression <- tbl_stack(list(tA,tB,tC),group_header=c("Modèle A","Modèle B","Modèle C"))
tableau_regression
```

Group	Characteristic	Beta	95% CI	p-value
Modèle A	group_traitement	-0.02	-0.72, 0.68	>0.9
Modèle B	group_traitement	0.09	-0.62, 0.79	0.8
	age	0.00	0.00, 0.00	0.9
	sex	-0.92	-2.1, 0.23	0.12
Modèle C	group_traitement	0.14	-0.57, 0.85	0.7
	age	0.00	0.00, 0.00	>0.9
	sex	-0.88	-2.0, 0.27	0.13
	district	0.09	-0.06, 0.24	0.3

\textcolor{blue}{Partie 3 : R-shiny}

Dans cette partie, il est question de faire une application r shiny qui permet:

- de visualiser les evenements par pays (le nombre d'evenement par pays dans une carte)
- de visualiser les evenements par pays, type, annee et la localisation.

Idée de conception et fonctionnement de l'application

Pour résoudre ces deux questions à la fois, nous donnons la possibilité à l'utilisateur de sélectionner directement les variables (pays, évenements, années) qui l'intéressent dans la base de donnée fournie par l'exercice, et ceci grâce à l'interface web fournie par R-shiny. Il en va de soi qu'avec ces données que l'utilisateur entre dans l'appli, nous lui retournons l'emplacement géographique de tous les évenements sélectionnés, la carte du pays sélectionné et aussi l'année. Nous nous servons de la légende qui est interactive pour afficher les différentes statistiques concernant sa sélection.

Ainsi, lorsque l'utilisateur voudra avoir par exemple le nombre d'évenements par pays, il lui suffira de sélectionner uniquement le pays d'intérêt et de sélectionner tous les types d'évenements ainsi que toutes les années.

liste des packages utilisés dans la section R-shiny

```r
#install.packages(c(leaflet, sf, shinydashboard, rnaturalearth, rnaturalearthdata, ))
```