Rédigé par :
Ibrahim Souleymane AMADOU
Elève ingénieur statisticien économiste

Ibrahim Souleymane Amadou

2023-07-19

Contents

1 Préliminaires

```
#vider la mémoire
rm(list=ls())
#lancer le garbage collector
gc()
```

```
##           used  (Mb) gc trigger  (Mb) max used  (Mb)
## Ncells 4757594 254.1    8364035 446.7  6627853 354.0
## Vcells 7160922  54.7   12255594  93.6 10138138  77.4
```

2 Partie 1

2.1 Préparation & mise en forme

2.1.1 Importation et mise en forme

```
# importation de la base
Base_Partie_1<-read_excel("F:/ISE/projet_R/data/Base_Partie 1.xlsx")

projet<-Base_Partie_1

# Selection des variables et detection des valeurs manquantes
projet %>%
  select(!key) %>%    # selection des variables mentionnées
  miss_var_summary(order = T) %>%  # NA par variable
  kable()
```

variable	n_miss	pct_miss
q17	131	52.4
q19	120	48.0
q14b	1	0.4
q16	1	0.4
q1	0	0.0
q2	0	0.0
q23	0	0.0
q24	0	0.0
q24a_1	0	0.0
q24a_2	0	0.0
q24a_3	0	0.0
q24a_4	0	0.0
q24a_5	0	0.0
q24a_6	0	0.0
q24a_7	0	0.0
q24a_9	0	0.0
q24a_10	0	0.0
q25	0	0.0
q26	0	0.0
q12	0	0.0
q20	0	0.0
filiere_1	0	0.0
filiere_2	0	0.0
filiere_3	0	0.0
filiere_4	0	0.0
q8	0	0.0
q81	0	0.0
gps_menlatitude	0	0.0
gps_menlongitude	0	0.0
submissiondate	0	0.0
start	0	0.0
today	0	0.0

```r
# Verification de NA pour la variable key
projet %>%
  select(key) %>%
  miss_var_summary()  #Pas de NA
```

```
## # A tibble: 1 x 3
##   variable n_miss pct_miss
##   <chr>     <int>    <dbl>
## 1 key           0        0
```

Il n' y a pas de NA pour la variable key

2.1.2 Creation de variables

```r
# Rename
projet<-projet %>%
          rename(region=q1,departement=q2,sexe=q23)

# Creation de la variable sexe_2
projet<-projet %>%
```

```r
         mutate(sexe_2=if_else(sexe=="Femme",1,0))
```

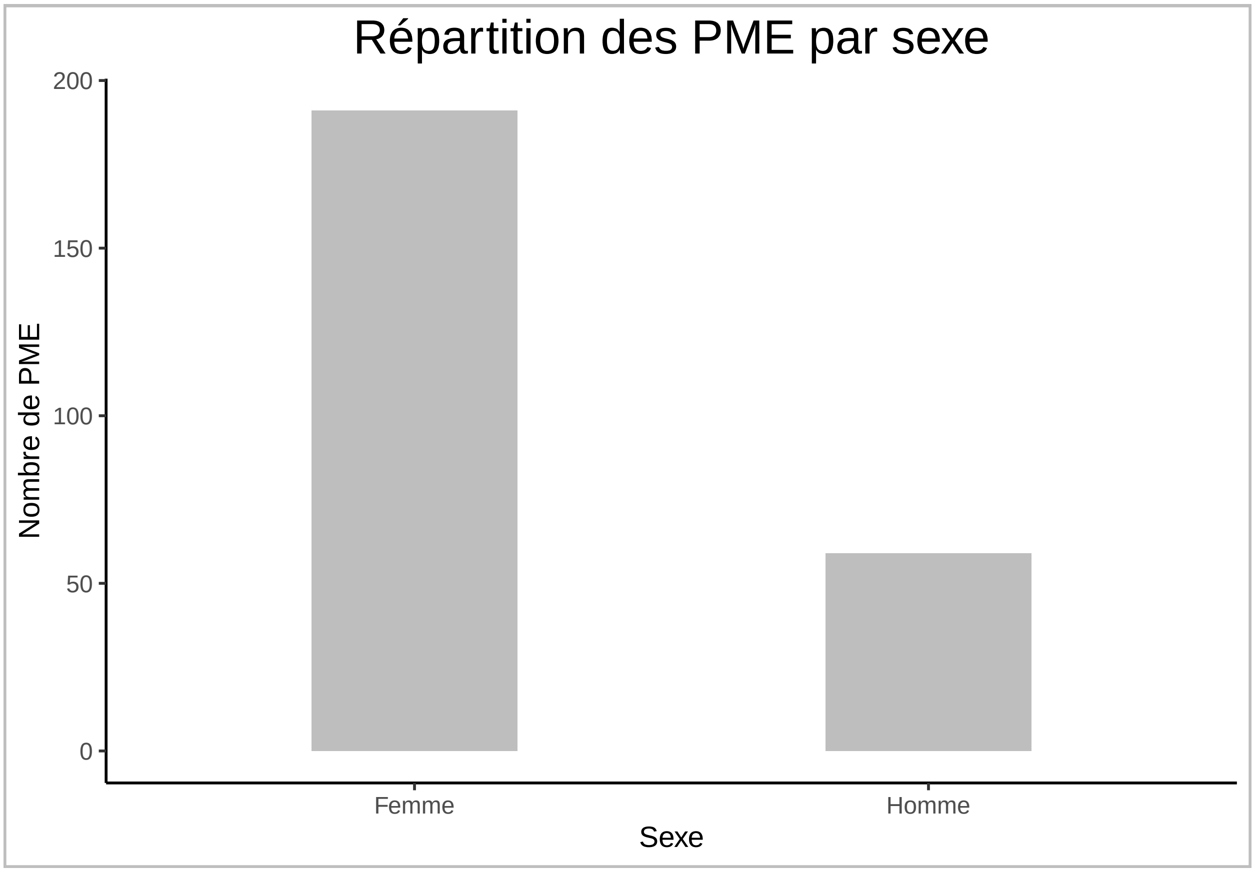
```r
# Creation du dataframe langues
langues<-projet %>%
         select(key,starts_with("q24a_"))

langues<-langues %>%
  mutate(parle=rowSums(langues[-1],na.rm=T)) %>%
  select(c("key","parle"))

# Fusion des bases projet et langues
base_fusion <- merge(projet, langues, by = "key")
```

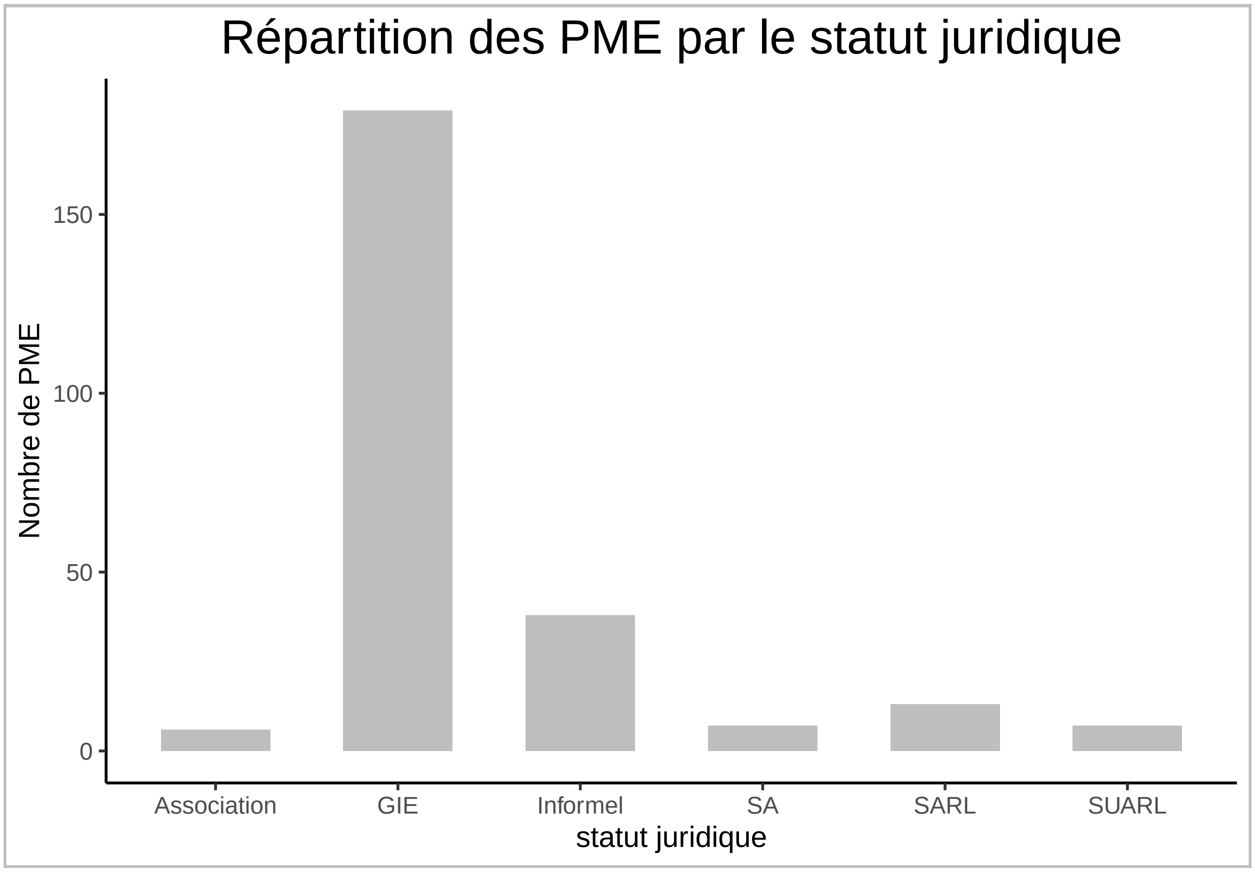
2.2 Analyses descriptives

```r
# Repartition des PME par sexe
# Création du graphique avec ggplot2
ggsexe<-ggplot(projet, aes(x = sexe)) +
  geom_bar(fill = "gray", width = 0.4) +
  theme_classic()+
  labs(title = "Répartition des PME par sexe",
       x = "Sexe",
       y = "Nombre de PME") +
  theme(panel.grid = element_blank())+
  theme(plot.title = element_text(size = 18, hjust = 0.5, vjust = 1),
        plot.background = element_rect(color = "gray", size = 1))
  #theme(plot.margin = margin(0.5, 0.5, 0.5, 0.5))
ggsexe
```



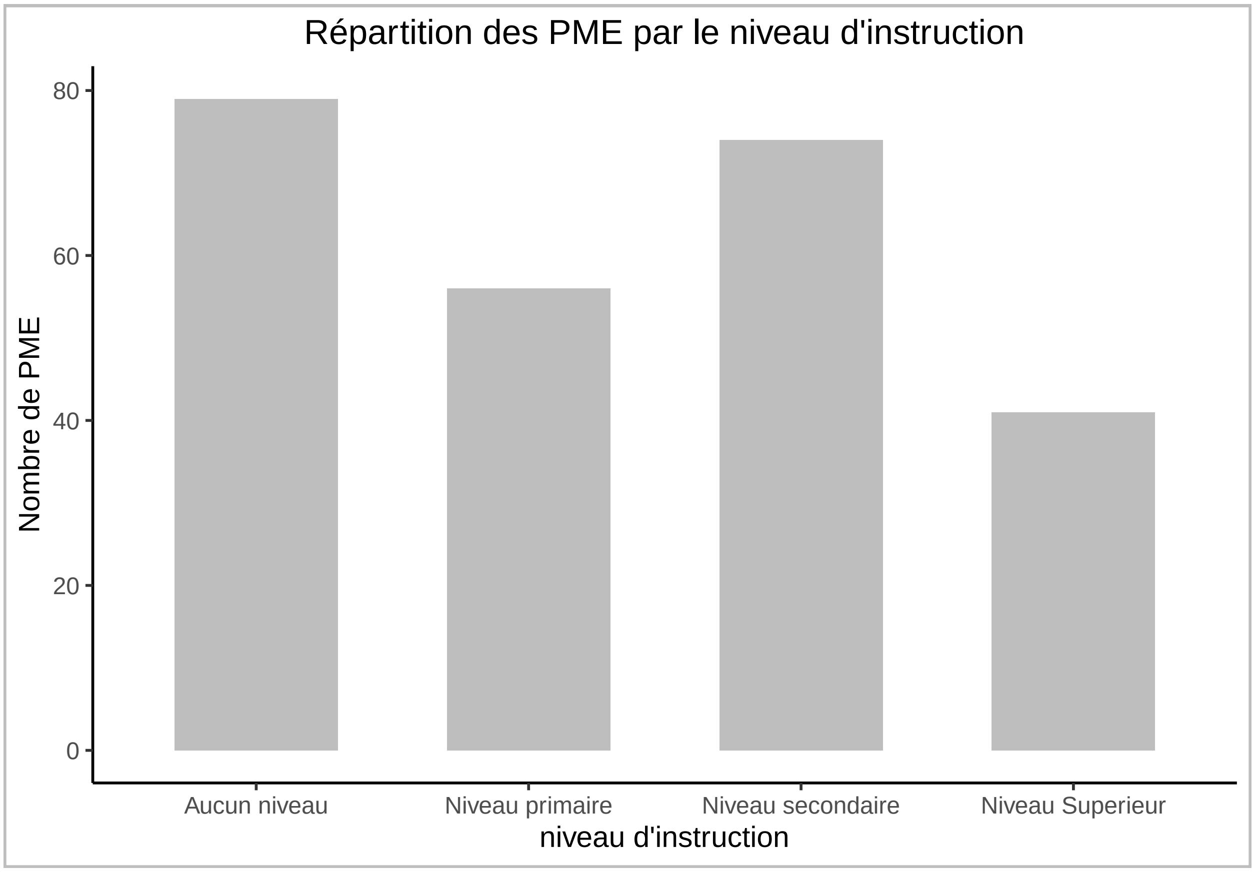
On remarque que les femmes détiennent un nombre plus élevé de petites et moyennes entreprises (PME) que les hommes.

```
# Repartition par le statut juridique
ggstatut<-ggplot(projet, aes(x = q12)) +
  geom_bar(fill = "gray", width = 0.6) +
  theme_classic()+
  labs(title = "Répartition des PME par le statut juridique",
       x = "statut juridique",
       y = "Nombre de PME") +
  theme(panel.grid = element_blank())+
  theme(plot.title = element_text(size = 18, hjust = 0.5, vjust = 1),
        plot.background = element_rect(color = "gray", size = 1))
ggstatut
```

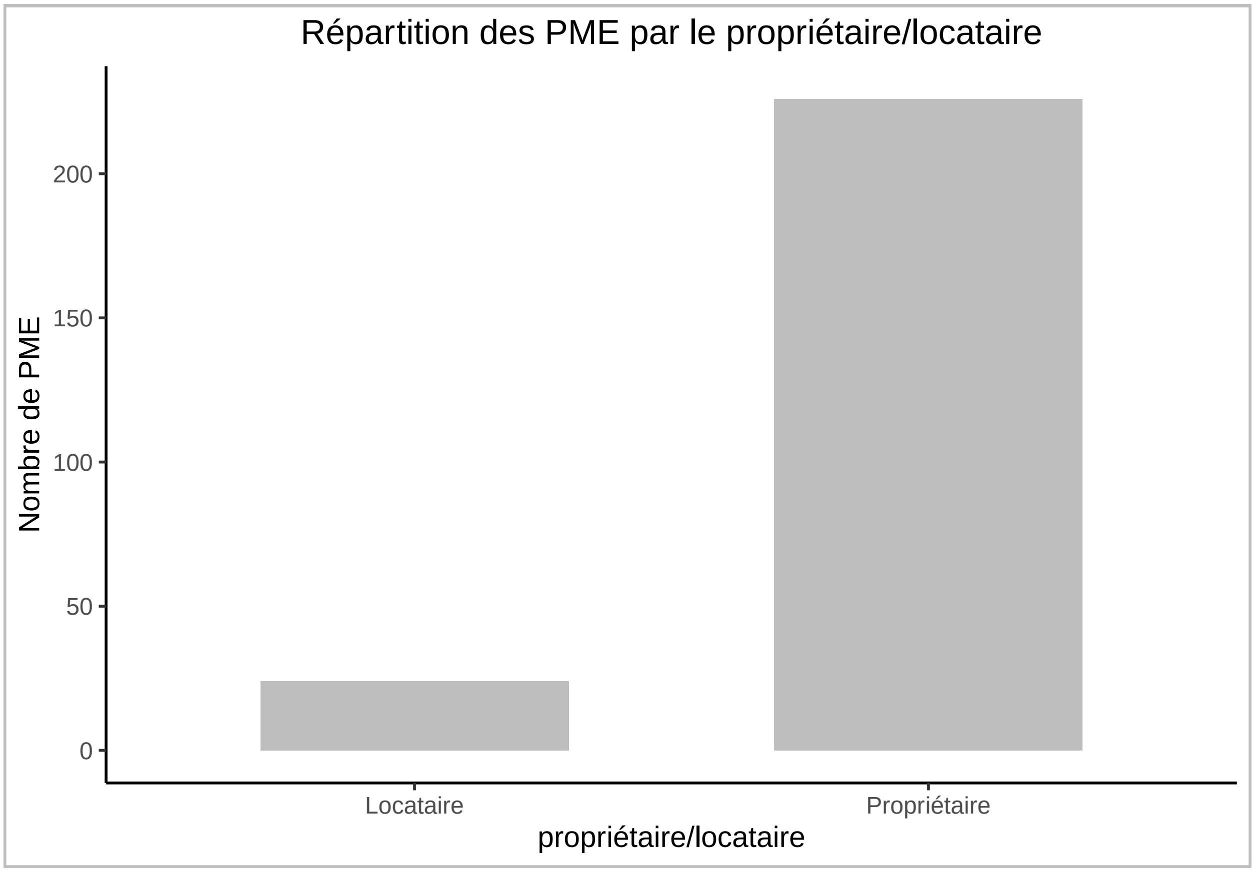


La grande majorité des PME adoptent la forme juridique de GIE, tandis que seulement un petit nombre choisissent d'être constituées en associations ou en SA.

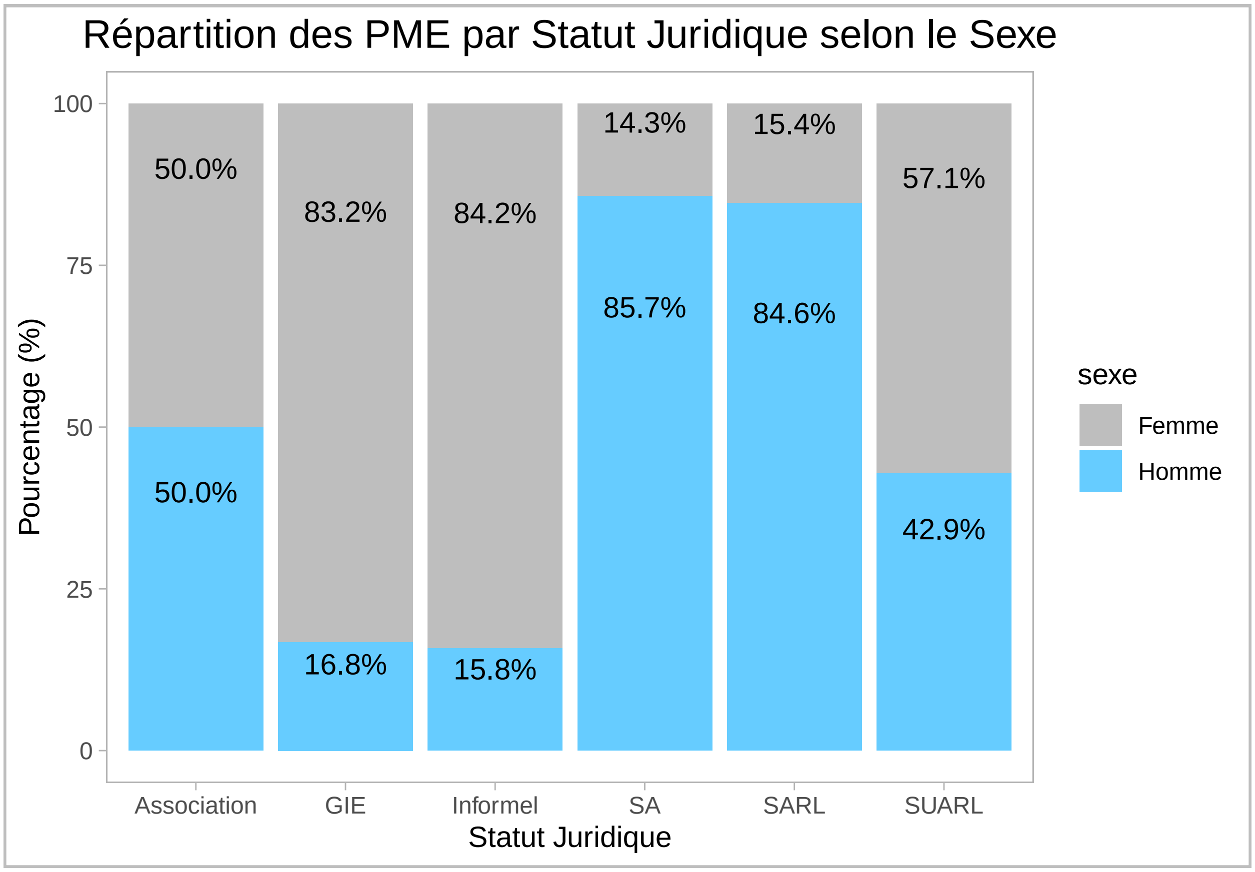
```
# Repartition par le niveau d'instruction
ggniv<-ggplot(projet, aes(x = q25)) +
  geom_bar(fill = "gray", width = 0.6) +
  theme_classic()+
  labs(title = "Répartition des PME par le niveau d'instruction",
       x = "niveau d'instruction",
       y = "Nombre de PME") +
  theme(panel.grid = element_blank())+
  theme(plot.title = element_text(size = 13, hjust = 0.5, vjust = 1),
        plot.background = element_rect(color = "gray", size = 1))
ggniv
```



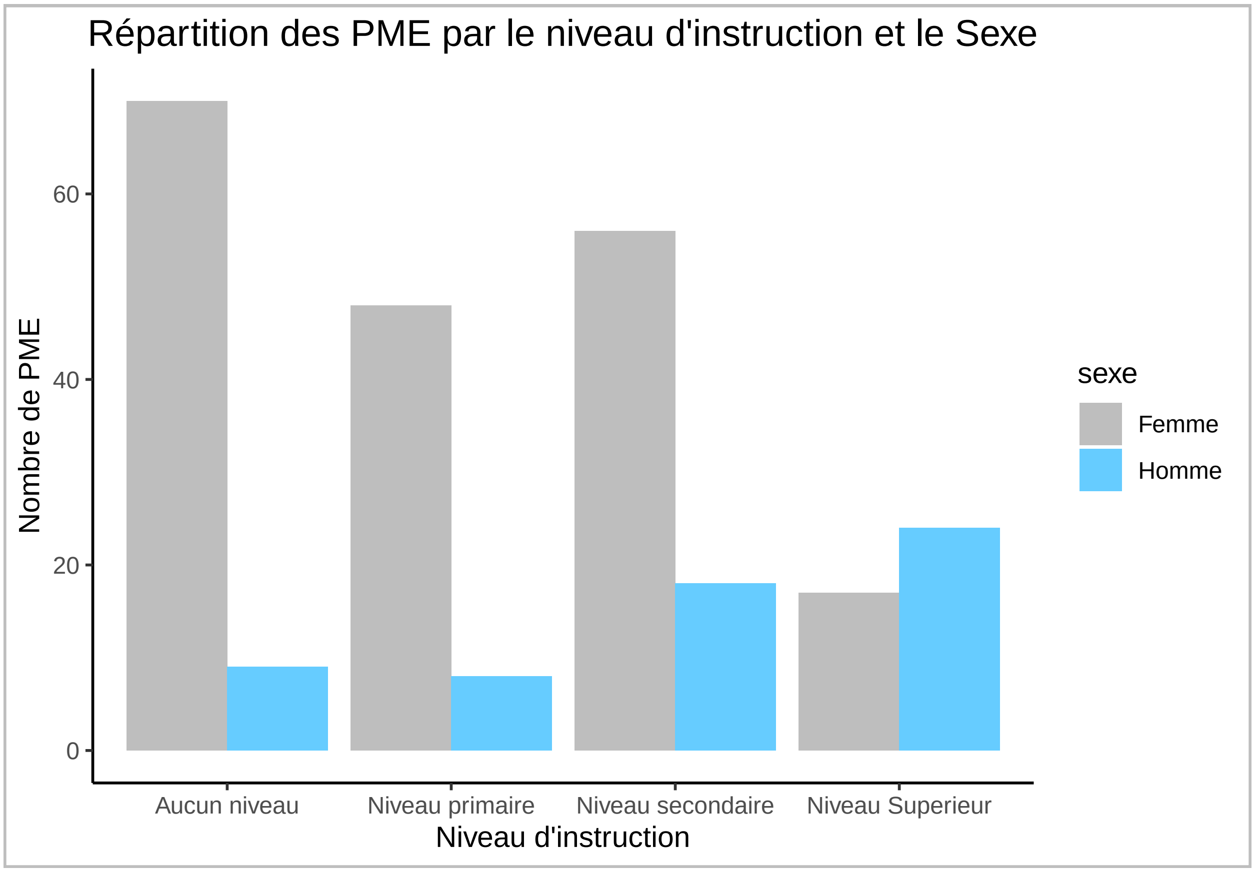

```r
# Repartition par le propriétaire/locataire
gg2<-ggplot(projet, aes(x = q81, fill = q81)) +
  geom_bar(fill = "gray", width = 0.6) +
  theme_classic()+
  labs(title = "Répartition des PME par le propriétaire/locataire",
       x = "propriétaire/locataire",
       y = "Nombre de PME") +
  theme(panel.grid = element_blank())+
  theme(plot.title = element_text(size = 13, hjust = 0.5, vjust = 1),
        plot.background = element_rect(color = "gray", size = 1))+
  scale_fill_manual(values = c("#FF9999", "#66CCFF"))
gg2
```




```r
# Repartition par le statut juridique et le sexe
data_group<-projet %>%
              group_by(q12,sexe) %>%
              summarize(n_pme=n()) %>%
              mutate(percent = n_pme / sum(n_pme) * 100) %>%
              ungroup()
ggcrois1<-ggplot(data_group, aes(x = q12,y = percent, fill = sexe)) +
  geom_bar(stat = "identity", position = "stack", width = 0.9) +
  geom_text(aes(label = sprintf("%.1f%%", percent)), position = position_stack(vjust = 0.8), vjust=0.5)
  theme_light()+
  labs(title = "Répartition des PME par Statut Juridique selon le Sexe",
       x = "Statut Juridique",
       y = "Pourcentage (%)") +
  theme(panel.grid = element_blank())+
  theme(plot.title = element_text(size = 15, hjust = 0.5, vjust = 1),
        plot.background = element_rect(color = "gray", size = 1))+
  scale_fill_manual(values = c("gray", "#66CCFF"))
ggcrois1
```



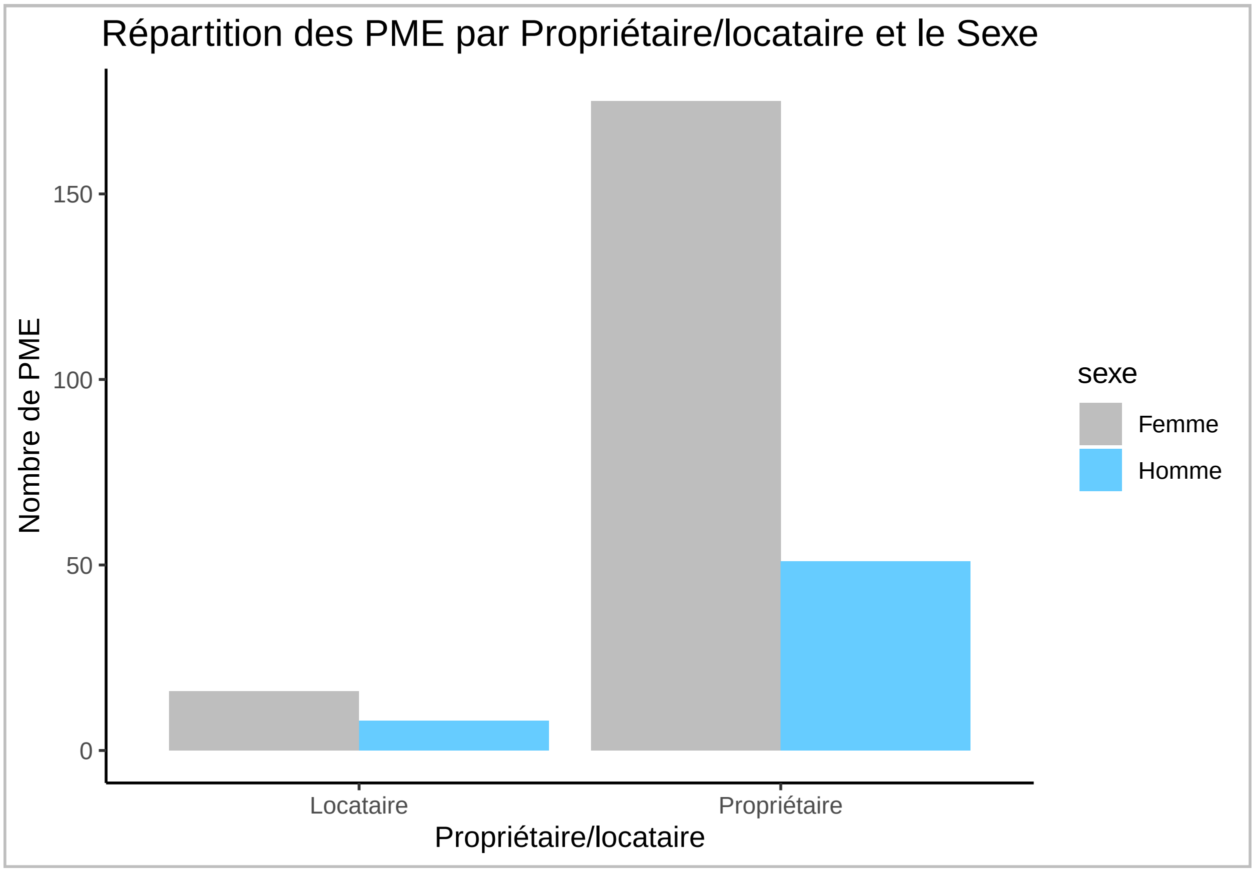

```r
# Repartition par le niveau d'instruction et le sexe
data_group<-projet %>%
              group_by(q25,sexe) %>%
              summarize(n_pme=n()) %>%
              ungroup()
ggcrois2<-ggplot(data_group, aes(x = q25,y = n_pme, fill = sexe)) +
  geom_bar(stat = "identity", position = "dodge", width = 0.9) +
  theme_classic()+
  labs(title = "Répartition des PME par le niveau d'instruction et le Sexe",
       x = "Niveau d'instruction",
       y = "Nombre de PME") +
  theme(panel.grid = element_blank())+
  theme(plot.title = element_text(size = 14, hjust = 0.5, vjust = 1),
        plot.background = element_rect(color = "gray", size = 1))+
  scale_fill_manual(values = c("gray", "#66CCFF"))
ggcrois2
```




```
# Repartion par le Propriétaire/locataire suivant le sexe
data_group<-projet %>%
              group_by(q81,sexe) %>%
              summarize(n_pme=n()) %>%
              ungroup()
```

```
## `summarise()` has grouped output by 'q81'. You can override using the `.groups`
## argument.
```

```
ggcrois3<-ggplot(data_group, aes(x = q81,y = n_pme, fill = sexe)) +
  geom_bar(stat = "identity", position = "dodge", width = 0.9) +
  theme_classic()+
  labs(title = "Répartition des PME par Propriétaire/locataire et le Sexe",
       x = "Propriétaire/locataire",
       y = "Nombre de PME") +
  theme(panel.grid = element_blank())+
  theme(plot.title = element_text(size = 14, hjust = 0.5, vjust = 1),
        plot.background = element_rect(color = "gray", size = 1))+
  scale_fill_manual(values = c("gray", "#66CCFF"))
ggcrois3
```




```r
# Stat desc sur les autres variables
projet %>%
  select(!c(key,q25,q81)) %>%
  tbl_summary(
    by = filiere_1,
    missing = "always",
    missing_text = "Missing",
    statistic =list(
      all_continuous2()~c("{median} ({p25} - {p75}","{mean} ({sd})", "{min} -   {max}"),
      all_categorical()~"{n}/{N} ({p}%)"
    ),
    percent = "column"
  ) %>%
  add_overall() %>%
  add_stat_label()
```

```
## 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 code chunk header.
```

Characteristic	**Overall**, N = 250	**0**, N = 142
region, n/N (%)		
Dakar	1/250 (0.4%)	1/142 (0.7%)
Diourbel	34/250 (14%)	1/142 (0.7%)
Fatick	30/250 (12%)	18/142 (13%)
Kaffrine	8/250 (3.2%)	0/142 (0%)
Kaolack	21/250 (8.4%)	1/142 (0.7%)
Kolda	9/250 (3.6%)	8/142 (5.6%)
Saint-Louis	42/250 (17%)	41/142 (29%)
Sédhiou	4/250 (1.6%)	4/142 (2.8%)
Thiès	51/250 (20%)	24/142 (17%)
Ziguinchor	50/250 (20%)	44/142 (31%)
Missing	0	0
departement, n/N (%)		
Bambey	20/250 (8.0%)	0/142 (0%)
Bignona	13/250 (5.2%)	12/142 (8.5%)
Birkelane	1/250 (0.4%)	0/142 (0%)
Dagana	39/250 (16%)	39/142 (27%)
Diourbel	12/250 (4.8%)	0/142 (0%)
Fatick	15/250 (6.0%)	13/142 (9.2%)
Foundiougne	15/250 (6.0%)	5/142 (3.5%)
Goudomp	3/250 (1.2%)	3/142 (2.1%)
Kaffrine	4/250 (1.6%)	0/142 (0%)
Kaolack	16/250 (6.4%)	1/142 (0.7%)
Kolda	7/250 (2.8%)	6/142 (4.2%)
Koungheul	3/250 (1.2%)	0/142 (0%)
Mbacké	2/250 (0.8%)	1/142 (0.7%)
Mbour	22/250 (8.8%)	8/142 (5.6%)
Médina Yoro Foula	1/250 (0.4%)	1/142 (0.7%)
Nioro	5/250 (2.0%)	0/142 (0%)
Oussouye	2/250 (0.8%)	2/142 (1.4%)
Podor	1/250 (0.4%)	1/142 (0.7%)
Rufisque	1/250 (0.4%)	1/142 (0.7%)
Saint-Louis	2/250 (0.8%)	1/142 (0.7%)
Sédhiou	1/250 (0.4%)	1/142 (0.7%)
Thiès	23/250 (9.2%)	12/142 (8.5%)
Tivaouane	6/250 (2.4%)	4/142 (2.8%)
Velingara	1/250 (0.4%)	1/142 (0.7%)
Ziguinchor	35/250 (14%)	30/142 (21%)
Missing	0	0
sexe, n/N (%)		
Femme	191/250 (76%)	98/142 (69%)
Homme	59/250 (24%)	44/142 (31%)
Missing	0	0
q24, Median (IQR)	55 (45, 62)	54 (43, 61)
Missing	0	0
q24a_1, n/N (%)	159/250 (64%)	99/142 (70%)
Missing	0	0
q24a_2, n/N (%)	245/250 (98%)	137/142 (96%)
Missing	0	0
q24a_3, n/N (%)	39/250 (16%)	34/142 (24%)
Missing	0	0
q24a_4, n/N (%)	61/250 (24%)	17/142 (12%)
Missing	0	0
q24a_5, n/N (%)	43/250 (17%)	29/142 (20%)
Missing	0	0
q24a_6, n/N (%)	36/250 (14%)	28/142 (20%)
Missing	0	0
q24a_7, n/N (%)	2/250 (0.8%)	1/142 (0.7%)

2.3 Un peu de cartograhie

```
# Transformation en donnees geographiques
projet_map<-projet
coordinates(projet_map)<- c("gps_menlongitude", "gps_menlatitude")
proj4string(projet_map) <- CRS("+proj=longlat +datum=WGS84")
class(projet_map)
```

```
## [1] "SpatialPointsDataFrame"
## attr(,"package")
## [1] "sp"
```

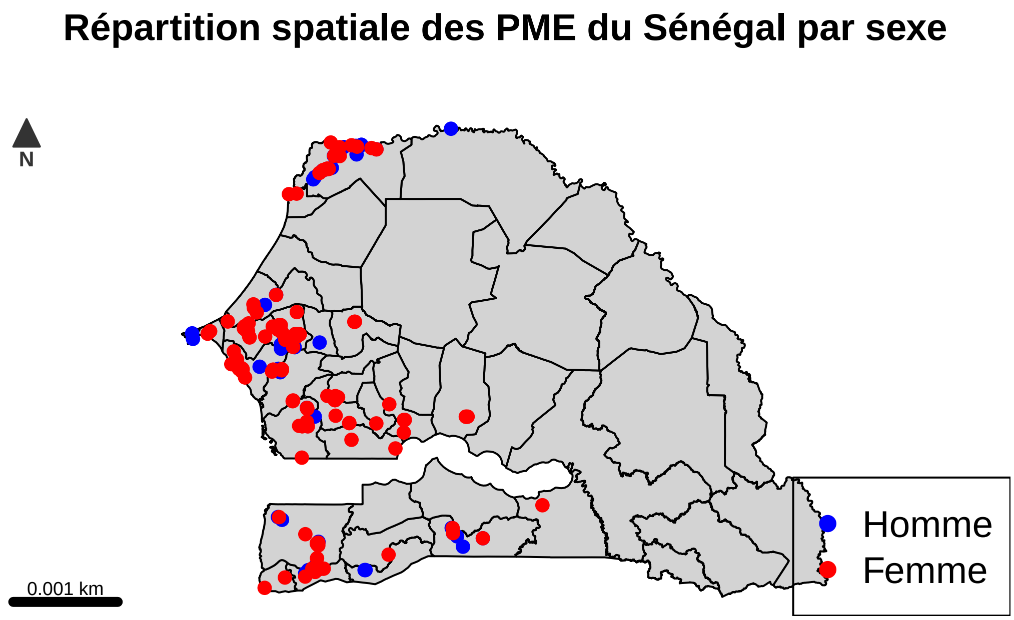
```
senegal <- st_read("F:/ISE/projet_R/data/Limite_des_départements.shp")
```

```
## Reading layer `Limite_des_départements' from data source
##   `F:\ISE\projet_R\data\Limite_des_départements.shp' using driver `ESRI Shapefile'
## Simple feature collection with 45 features and 11 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -17.53022 ymin: 12.30813 xmax: -11.34522 ymax: 16.6928
## Geodetic CRS:  WGS 84
```

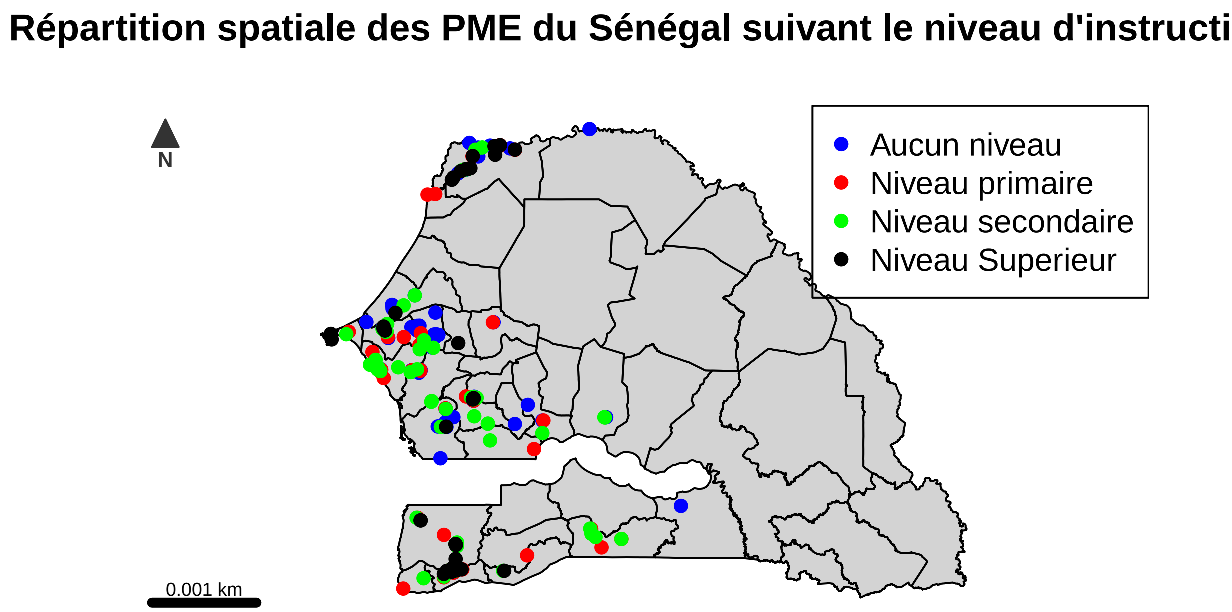
```
crs<- CRS("+proj=longlat +datum=WGS84")
sen<-senegal$geometry
#st_crs(senegal) <- crs
#ab_dept<-unique(projet$region)

centroids <- st_centroid(senegal)
centroids<-st_coordinates(centroids$geometry)

plot(sen, col = "lightgray",
     main = "Répartition spatiale des PME du Sénégal par sexe")
# Ajouter les points des PME sur la carte en fonction du sexe
points(projet_map[projet_map$sexe == "Homme",], col = "blue", pch = 16, cex = 1)
points(projet_map[projet_map$sexe == "Femme",], col = "red", pch = 16, cex = 1)
# Légendes
legend("bottomright", legend = c("Homme", "Femme"), col = c("blue", "red"), pch = 16, cex = 1.2)
# Ajouter le nord
mf_arrow()
# Ajouter l'échelle
mf_scale(pos = "bottomleft", lwd = 5,col = "black")
```

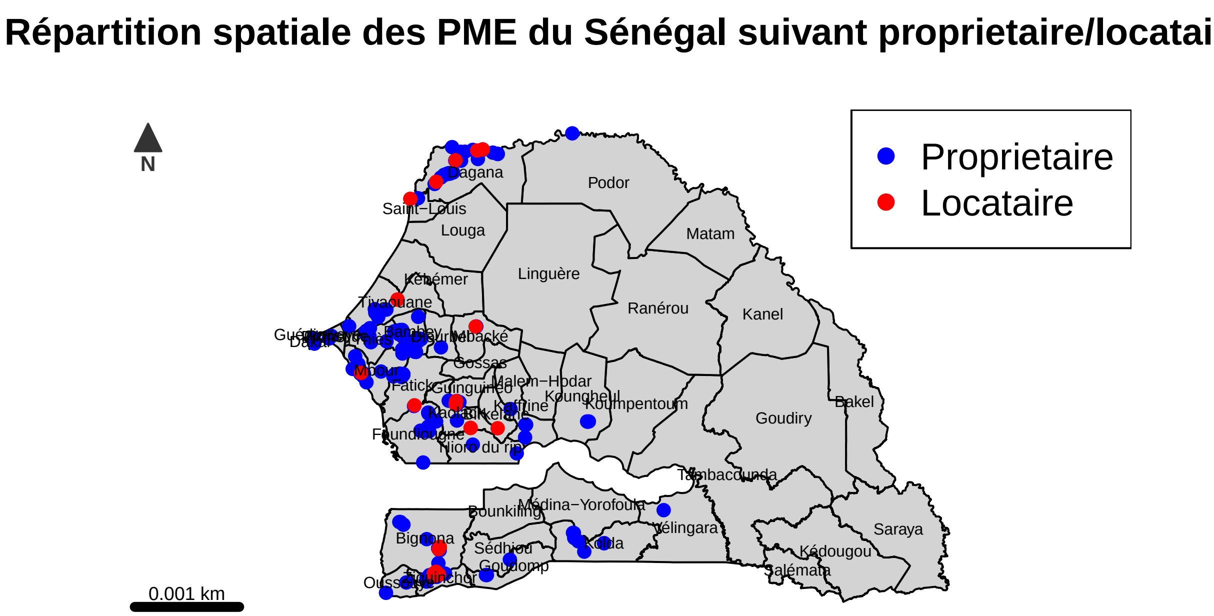



```r
# une réprésentation spatiale des PME suivant le niveau d'instruction
plot(sen, col = "lightgray",
     main = "Répartition spatiale des PME du Sénégal suivant le niveau d'instruction")
# Ajouter les points des PME sur la carte en fonction du sexe
points(projet_map[projet_map$q25 == "Aucun niveau",], col = "blue", pch = 16, cex = 1)
points(projet_map[projet_map$q25 == "Niveau primaire",], col = "red", pch = 16, cex = 1)
points(projet_map[projet_map$q25 == "Niveau secondaire",], col = "green", pch = 16, cex = 1)
points(projet_map[projet_map$q25 == "Niveau Superieur",], col = "black", pch = 16, cex = 1)
# Légendes
legend("topright", legend = c("Aucun niveau", "Niveau primaire", "Niveau secondaire", "Niveau Superieur
# Ajouter le nord
mf_arrow()
# Ajouter l'échelle
mf_scale(pos = "bottomleft", lwd = 5,col = "black")
```




```r
plot(sen, col = "lightgray",
     main = "Répartition spatiale des PME du Sénégal suivant proprietaire/locataire")
# Ajouter les points des PME sur la carte en fonction du sexe
points(projet_map[projet_map$q81 == "Propriétaire",], col = "blue", pch = 16, cex = 1)
points(projet_map[projet_map$q81 == "Locataire",], col = "red", pch = 16, cex = 1)
# Légendes
legend("topright", legend = c("Proprietaire", "Locataire"), col = c("blue", "red"), pch = 16, cex = 1.2
# Ajouter le nord
mf_arrow()
# Ajouter l'échelle
mf_scale(pos = "bottomleft", lwd = 5,col = "black")

text(centroids, labels = senegal$NOM, col = "black", cex = 0.5)
```



3 Partie 2

3.1 Nettoyage et gestion des données

```r
# Importation
Base_Partie_2 <- read_excel("F:/ISE/projet_R/data/Base_Partie 2.xlsx")
projet2<-Base_Partie_2
# Changement de nom de variable et remplacement des valeurs negatives par NA
projet2<-projet2 %>%
  rename(destination=country_destination) %>%
  mutate(destination=if_else(destination<=0,NA,destination))
```

```r
# Decoupage de la variable age
projet2<-projet2 %>%
  mutate(class_age=cut(age,breaks = seq(0,max(age)+5,by=5)))
# Creation de la variable nb_entretiens
projet2<-projet2 %>%
  group_by(enumerator) %>%
  mutate(nb_entretiens=n()) %>%
  ungroup
# Creation de la variable aleas
set.seed(111) # Assure la reproductibilité des résultats
projet2<-projet2 %>%
  mutate(aleas=sample(0:1, nrow(projet2), replace = TRUE))
```

```r
# Fusion des bases
pop<-read_excel("F:/ISE/projet_R/data/Base_Partie 2.xlsx",sheet = 2)
projet2<-merge(projet2, pop, by = "district")
```

```r
# Calcul de la duree de l'entretien et de la duree moyenne
projet2<-projet2 %>%
  mutate(duree=endtime-starttime) %>%
  group_by(enumerator) %>%
  mutate(duree_moy=mean(duree)) %>%
  ungroup
```

```r
# utilisation de la fonction apply pour renommer les noms de variable
colnames(projet2) <- apply(projet2, 2, function(col) paste0("endline_", names(projet2)))
```

3.2 Analyse et visualisation des données

```r
# Tableau recapitulatif

projet2 %>%
  select(endline_district, endline_age, endline_children_num) %>%
  tbl_summary(
    type = list(endline_children_num ~ "continuous"),
    statistic = list(all_continuous() ~ "{mean}"),
    by = endline_district,
    digits = ~1
  ) %>%
  modify_header(label = "**District**") %>%
  bold_labels()
```

```
## 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 code chunk header.
```

District	**1**, N = 8	**2**, N = 27	**3**, N = 8	**4**, N = 5	**5**, N = 6	**6**, N
___endline_age___	29.6	62.6	26.1	26.0	24.3	23.
___endline_children_num___	1.5	0.9	0.0	0.0	0.5	0.1

```r
# Test ANOVA pour la difference entre les 2 groupes
test <- projet2 %>%
  select(endline_age, endline_sex) %>%
  tbl_summary(
    by = endline_sex,
    statistic = list(all_continuous() ~ "{mean}")
  ) %>%
  add_p(~ "t.test") # pvalue>0,05 : Pas de difference significative
test
```

```
## 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 code chunk header.
```

Characteristic	**0**, N = 86	**1**, N = 11	**p-value**
endline_age	26	111	0.4

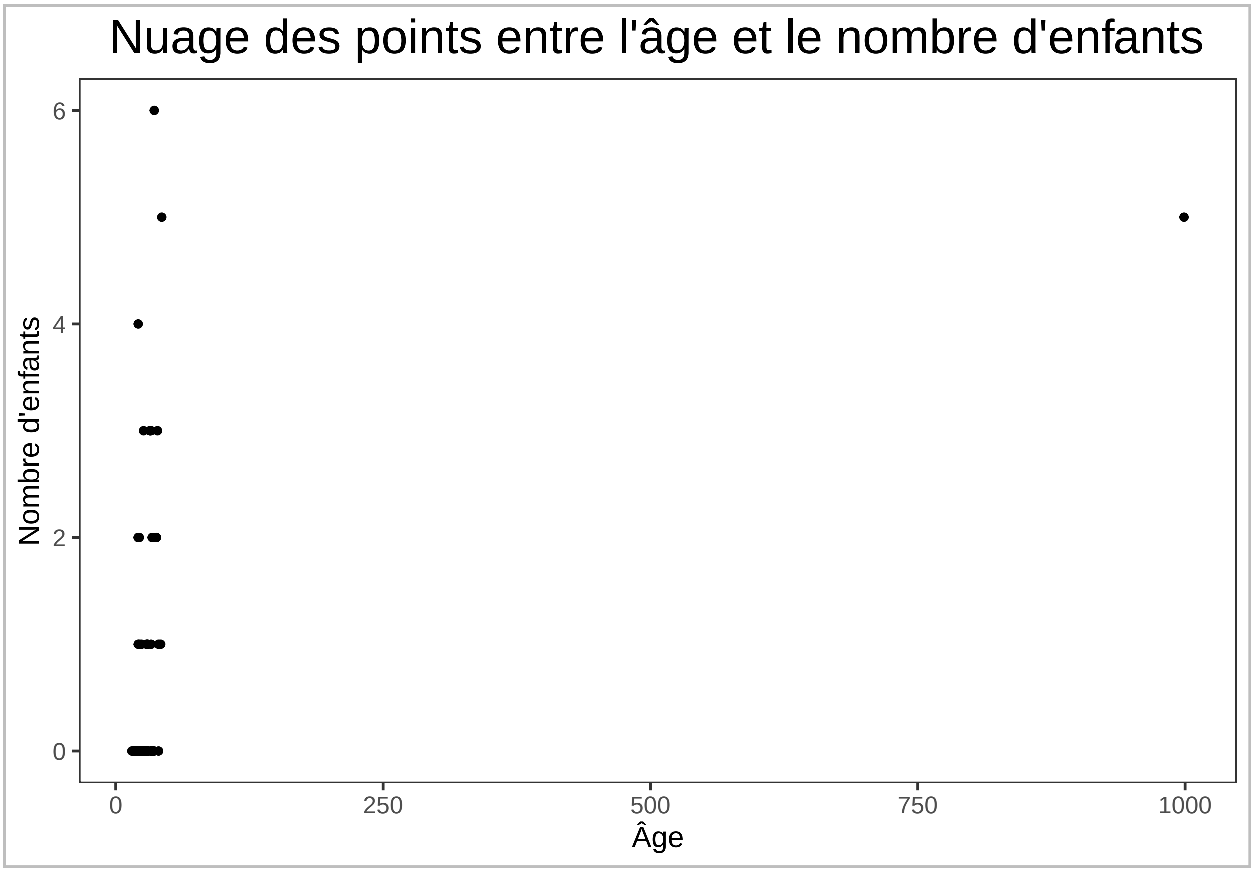
```r
# test avec le test de Student pour la difference de moyenne
test2 <- projet2 %>%
```

```
  select(endline_age, endline_sex) %>%
  tbl_summary(
    by = endline_sex,
    statistic = list(all_continuous() ~ "{mean}")
  ) %>%
  add_p(~ "fisher.test") # pvalue>0,05 : Pas de difference significative
test2
```

```
## 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 code chunk header.
```

Characteristic	**0**, N = 86	**1**, N = 11	**p-value**
endline_age	26	111	0.2

```
# Nuage de points
points <- ggplot(projet2, aes(x = endline_age, y = endline_children_num)) +
  geom_point(size = 1) +  # Définir la taille des points
  labs(x = "Âge", y = "Nombre d'enfants", title = "Nuage des points entre l'âge et le nombre d'enfants")
  theme_bw()+ # Utiliser un thème bw pour le graphique
  theme(panel.grid = element_blank())+
  theme(plot.title = element_text(size = 18, hjust = 0.5, vjust = 1),
        plot.background = element_rect(color = "gray", size = 1))
points
```




```
# Estimation de l'effet d'appartenance au groupe traitement sur l'intention de migrer
# La variable à expliquer "intention" étant qualitative ordinale, alors on applique le modèle multilino
```

```r
projet2$endline_intention<-factor(projet2$endline_intention)
projet2$endline_aleas<-factor(projet2$endline_aleas)
model_multinomial<-multinom(endline_intention ~ endline_aleas,
                          data = projet2
                          )
```

```
## # weights:  21 (12 variable)
## initial  value 188.753284
## iter  10 value 116.186460
## iter  20 value 115.971528
## final  value 115.970744
## converged
```

```r
summary(model_multinomial)
```

```
## Call:
## multinom(formula = endline_intention ~ endline_aleas, data = projet2)
##
## Coefficients:
##   (Intercept) endline_aleas1
## 2   -2.427783     -0.9394068
## 3   -2.140106      0.5645380
## 4   -2.140061     -0.1286362
## 5   -1.916944     -1.4502696
## 6   -3.526177      1.5451686
## 7   -2.833128    -11.8013745
##
## Std. Errors:
##   (Intercept) endline_aleas1
## 2   0.6022930      1.1820036
## 3   0.5286036      0.6932361
## 4   0.5285928      0.8045063
## 5   0.4789740      1.1241954
## 6   1.0145089      1.1461729
## 7   0.7275776    279.6688664
##
## Residual Deviance: 231.9415
## AIC: 255.9415
```

```r
# Odds ratio
odds.ratio(model_multinomial)
```

```
##                           OR       2.5 %      97.5 %         p
## 2/(Intercept)     8.8232e-02  2.7099e-02  2.8730e-01 5.556e-05 ***
## 2/endline_aleas1  3.9086e-01  3.8539e-02  3.9641e+00 0.4267543
## 3/(Intercept)     1.1764e-01  4.1746e-02  3.3150e-01 5.152e-05 ***
## 3/endline_aleas1  1.7586e+00  4.5195e-01  6.8432e+00 0.4154435
## 4/(Intercept)     1.1765e-01  4.1749e-02  3.3150e-01 5.153e-05 ***
## 4/endline_aleas1  8.7929e-01  1.8169e-01  4.2553e+00 0.8729641
## 5/(Intercept)     1.4706e-01  5.7515e-02  3.7600e-01 6.276e-05 ***
## 5/endline_aleas1  2.3451e-01  2.5896e-02  2.1236e+00 0.1970329
## 6/(Intercept)     2.9417e-02  4.0276e-03  2.1490e-01 0.0005094 ***
## 6/endline_aleas1  4.6888e+00  4.9595e-01  4.4328e+01 0.1776226
## 7/(Intercept)     5.8829e-02  1.4134e-02  2.4490e-01 9.864e-05 ***
## 7/endline_aleas1  7.4942e-06 6.6093e-244 8.4976e+232 0.9663411
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
modele_A<-model_multinomial %>%
  tbl_regression()
```

```
## i Multinomial models have a different underlying structure than the models
## gtsummary was designed for. Other gtsummary functions designed to work with
## tbl_regression objects may yield unexpected results.
```

```r
modele_B<-multinom(endline_intention ~ endline_aleas + endline_age + endline_sex, data = projet2) %>%
  tbl_regression()
```

```
## # weights:  35 (24 variable)
## initial  value 188.753284
## iter  10 value 116.742875
## iter  20 value 112.762130
## iter  30 value 112.537727
## iter  40 value 112.317084
## iter  50 value 112.312904
## final  value 112.312833
## converged
```

```
## i Multinomial models have a different underlying structure than the models
## gtsummary was designed for. Other gtsummary functions designed to work with
## tbl_regression objects may yield unexpected results.
```

```r
modele_C<-multinom(endline_intention ~ endline_aleas + endline_age + endline_sex + endline_district, da
  tbl_regression()
```

```
## # weights:  42 (30 variable)
## initial  value 188.753284
## iter  10 value 122.414845
## iter  20 value 112.741425
## iter  30 value 111.013321
## iter  40 value 110.879266
## iter  50 value 110.615428
## iter  60 value 110.595571
## final  value 110.595521
## converged
```

```
## i Multinomial models have a different underlying structure than the models
## gtsummary was designed for. Other gtsummary functions designed to work with
## tbl_regression objects may yield unexpected results.
```

```r
results<-tbl_stack(
  list(modele_A, modele_B, modele_C),
  group_header = c("Modèle A : Modèle vide", "Modèle B", "Modele C")
)
results
```

```
## 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 code chunk header.
```

Group	**Characteristic**	**log(OR)**	**95% CI**	**p-value**
Modèle A : Modèle vide	endline_aleas			
	0	—	—	
	1	-0.94	-3.3, 1.4	0.4
	endline_aleas			
	0	—	—	
	1	0.56	-0.79, 1.9	0.4
	endline_aleas			
	0	—	—	
	1	-0.13	-1.7, 1.4	0.9
	endline_aleas			
	0	—	—	
	1	-1.5	-3.7, 0.75	0.2
	endline_aleas			
	0	—	—	
	1	1.5	-0.70, 3.8	0.2
	endline_aleas			
	0	—	—	
	1	-12	-560, 536	>0.9
Modèle B	endline_aleas			
	0	—	—	
	1	-1.1	-3.4, 1.3	0.4
	endline_age	-0.01	-0.06, 0.05	0.8
	endline_sex	0.96	-1.5, 3.4	0.4
	endline_aleas			
	0	—	—	
	1	0.57	-0.80, 1.9	0.4
	endline_age	0.00	-0.02, 0.01	0.8
	endline_sex	-0.41	-2.6, 1.8	0.7
	endline_aleas			
	0	—	—	
	1	-0.13	-1.7, 1.5	0.9
	endline_age	0.04	-0.05, 0.13	0.4
	endline_sex	-45	-45, -45	<0.001
	endline_aleas			
	0	—	—	
	1	-1.4	-3.6, 0.84	0.2
	endline_age	-0.01	-0.15, 0.13	0.9
	endline_sex	-32	-32, -32	<0.001
	endline_aleas			
	0	—	—	
	1	1.6	-0.62, 3.9	0.2
	endline_age	-0.04	-0.18, 0.11	0.6
	endline_sex	-33	-33, -33	<0.001
	endline_aleas			
	0	—	—	
	1	-27	-27, -27	<0.001
	endline_age	-0.07	-0.36, 0.23	0.6
	endline_sex	-31	-31, -31	<0.001
Modele C	endline_aleas			
	0	—	—	
	1	-1.1	-3.5, 1.2	0.4
	endline_age	-0.02	-0.20, 0.16	0.8
	endline_sex	0.82	-1.7, 3.4	0.5
	endline_district	-0.22	-0.74, 0.30	0.4
	endline_aleas			
	0	—	—	
	1	0.60	-0.77, 2.0	0.4

4 Partie 3

Voir le code app_shiny. Ci - dessous, les paramètres de mon compte grâce auxquels se fait le déploiement sur le net.

```
#library(rsconnect)
#rsconnect::setAccountInfo(name='amado',
#                          token='A6C18CD36EE03B9DEC387FAF19FA22E5',
#                          secret='d6D0NaAPbMmXBZoZm99t9gdH0ztsLMT8AbQQZzGa')
#rsconnect::deployApp('F:/ISE/projet_R/app_shiny/app_shiny')
```