

République du Sénégal

Un Peuple-Un But-Une Foi



Ministère de l'Economie, du Plan et de la Coopération

Agence nationale de la Statistique et de la Démographie



Ecole nationale de la Statistique et de l'Analyse économique Pierre Ndiaye



TP 1 R :

Travaux pratiques 1 de R

Rédigé par :

KONLAMBIGUE Youdan-yamin

Elève ingénieur statisticien économiste en troisième année de classe préparatoire

Professeur :

M. HEMA

Année scolaire 2023-2024

Contents

1	Préparation des données	3
1.1	Importation et mise en forme	3
1.2	Création de variables	3
2	Analyses descriptives	3
3	Un peu de cartographie	10

```
library(sf)
library(labelled) # for general functions to work with labelled data
library(tidyverse) # general wrangling
library(readxl) # manipuler les fichiers excel
library(gtsummary) # to demonstrate automatic use of variable labels in
library(ggstats)
library(ggplot2)
library(ggspatial)

rm(list = ls())
```

1 Préparation des données

1.1 Importation et mise en forme

- Importation

```
projet <- read_excel("Base_Projet.xlsx")
```

- La base de données contient 33 variables collectées sur 250 PME.
- Il y'a 0 valeurs manquantes sur la variable `key`.

1.2 Création de variables

- Renommage des variables et création de `sexe_2`

```
projet <- projet %>% rename("region" = "q1",
                           "departement" = "q2",
                           "sexe" = "q23") %>%
  mutate(sexe_2 = if_else(sexe == "Femme", 1, 0))
```

- Data.frame `languages`

```
languages <- projet %>%
  select("key", starts_with("q24a_")) %>%
  mutate(parle = sum(c_across(2:ncol(.)))) %>%
  select("key", "parle")
```

- Merge des data.frame `projet` et `languages`

```
length(unique(projet$key)) == nrow(projet)
```

```
## [1] TRUE
```

```
projet <- left_join(projet, languages, by = "key")
```

2 Analyses descriptives

- fonction `univarie()`

```
univarie <- function(data, var_cont, var_cat, stat) {
  data <- data
  var <- c(var_cont, var_cat)
  tbl <- data %>% tbl_summary(include = all_of(var) ,
                             statistic = list(
                               all_continuous() ~ paste0(stat[1], ": {" ,stat[1],"}", stat[2], ": {" ,stat[2],"}",
```

```

        all_categorical() ~ "{p} %"),
        sort = all_categorical() ~ "frequency"
    )

for (i in var_cont){
  ggplot(data) +
  aes(x = data[i]) +
  geom_density(adjust = .5)

  ggplot(data) +
  aes(x = data[i]) +
  geom_histogram(
    fill = "lightblue",
    colour = "black",
    binwidth = 1
  ) +
  xlab(i) +
  ylab("Effectifs")
}

for (i in var_cat){
  ggplot(data) +
  aes(x = forcats::fct_infreq(data[i]),
      y = after_stat(prop), by = 1) +
  geom_bar(stat = "prop",
          fill = "#4477AA", colour = "black") +
  geom_text(
    aes(label = after_stat(prop) |>
        scales::percent(accuracy = .1)),
    stat = "prop",
    nudge_y = .02
  ) +
  theme_minimal() +
  theme(
    panel.grid = element_blank(),
    axis.text.y = element_blank()
  ) +
  xlab(NULL) + ylab(NULL) +
  ggtitle(i)
}
}

```

- application

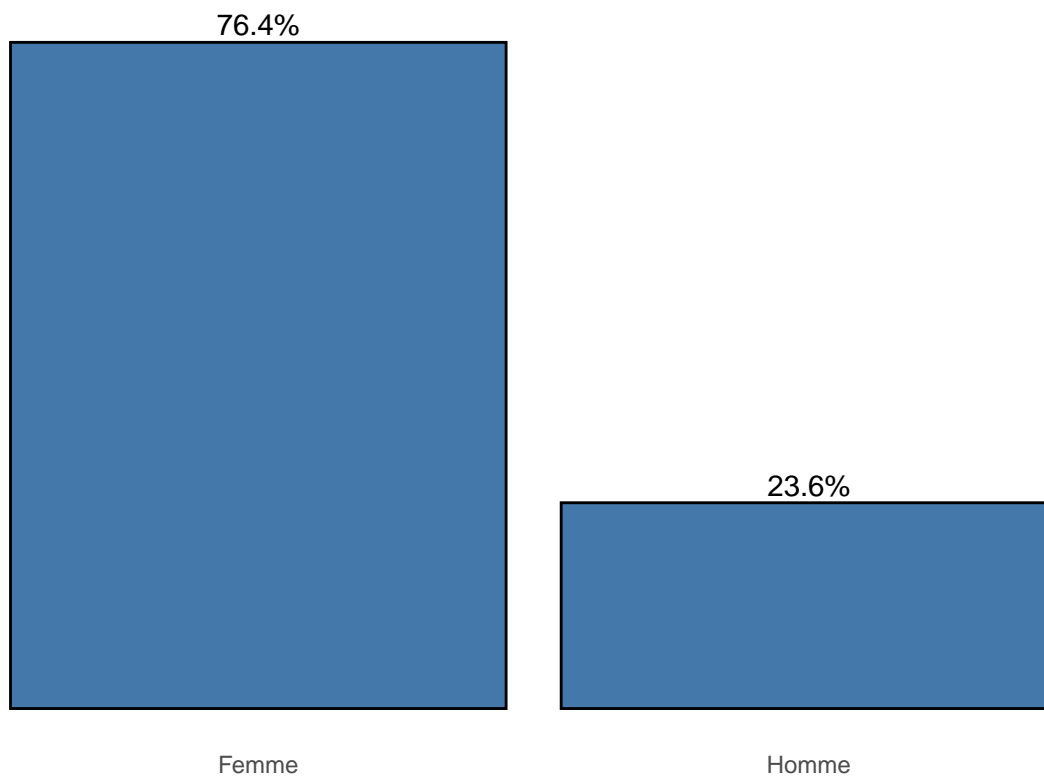
```

var_cat <- c("sexe", "q25", "q12")
var_cont <- c()

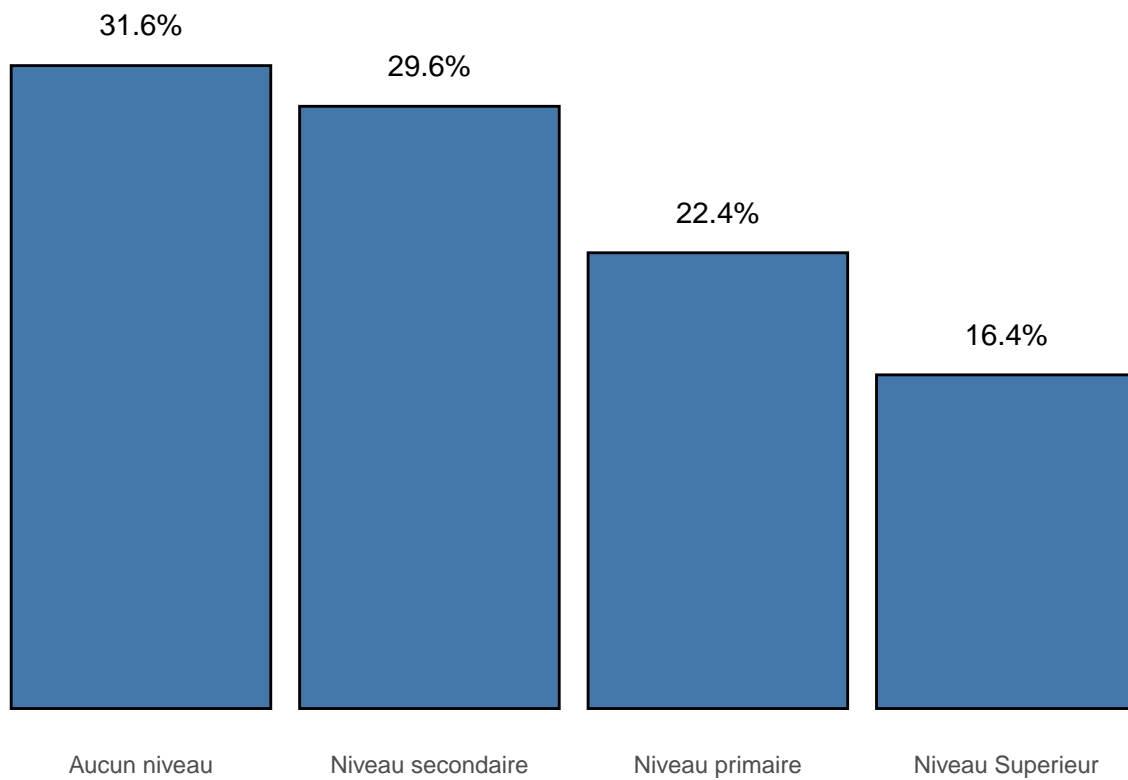
stat <- c("mean", "sd")
univarie(projet, var_cont, var_cat , stat )

```

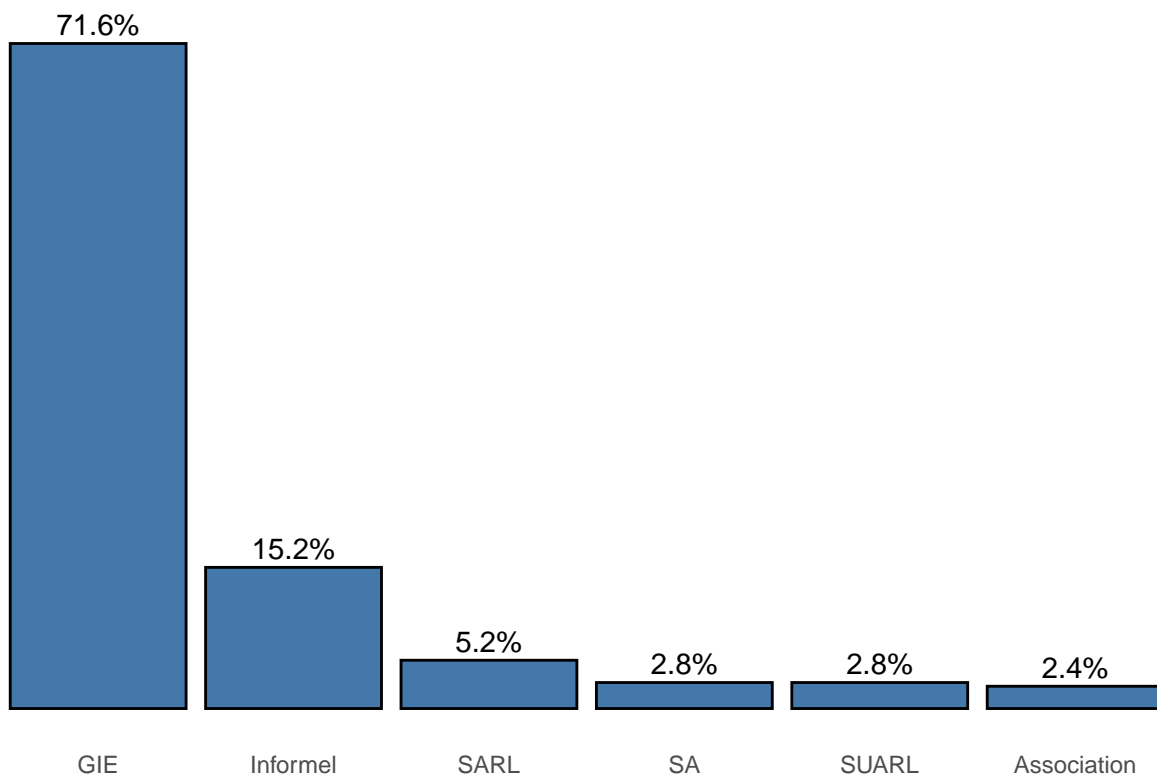
sexe



sexe



sexe



- fonction bivarie()

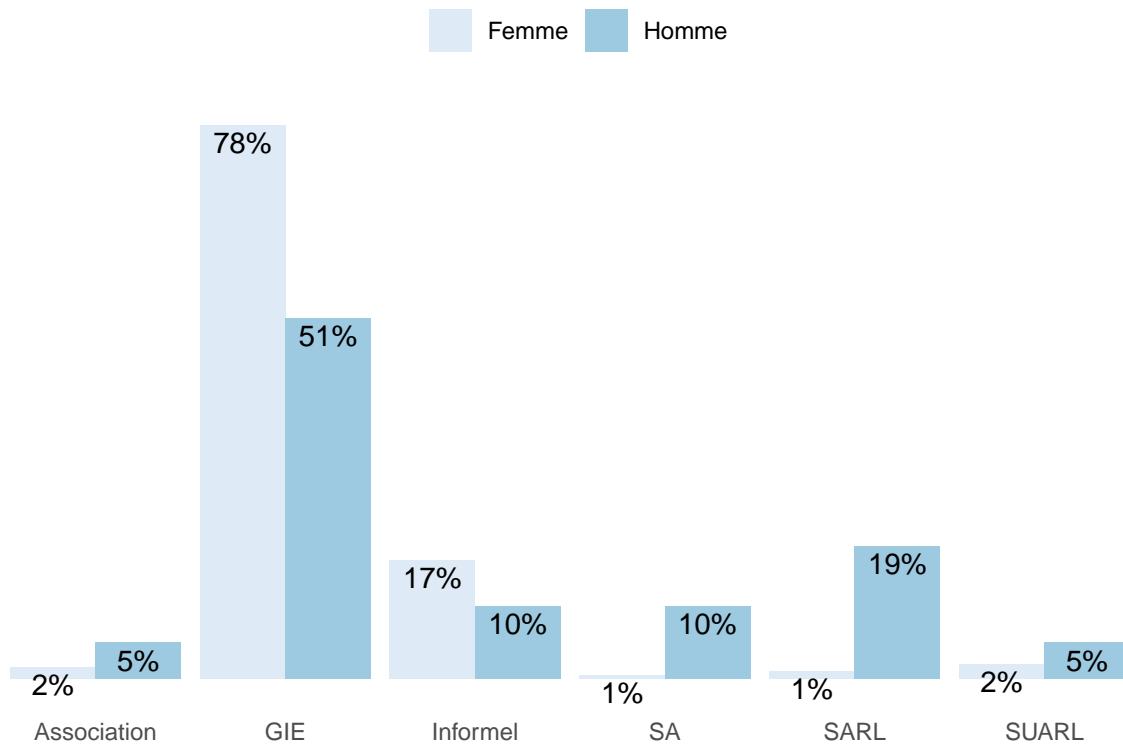
```
bivarie <- function(data, var1, var2){  
  ggplot(data) +  
  aes(  
    x = var1,  
    y = after_stat(prop),  
    fill = var2,  
    by = as_factor(as.character(var2)),  
    label = scales::percent(after_stat(prop), accuracy = 1)  
  ) +  
  geom_bar(  
    stat = "prop",  
    position = position_dodge(.9)  
  ) +  
  geom_text(  
    aes(y = after_stat(prop) - 0.01),  
    stat = "prop",  
    position = position_dodge(.9),  
    vjust = "top"  
  ) +  
  scale_y_continuous(labels = scales::percent)+  
  
  theme_light() +  
  xlab("") +  
  ylab("") +  
  labs(fill = "") +
```

```
ggtitle(paste0(var1, " selon ", var2)) +
theme(
  panel.grid = element_blank(),
  panel.border = element_blank(),
  axis.text.y = element_blank(),
  axis.ticks = element_blank(),
  legend.position = "top"
) +
scale_fill_brewer()
}
```

- application

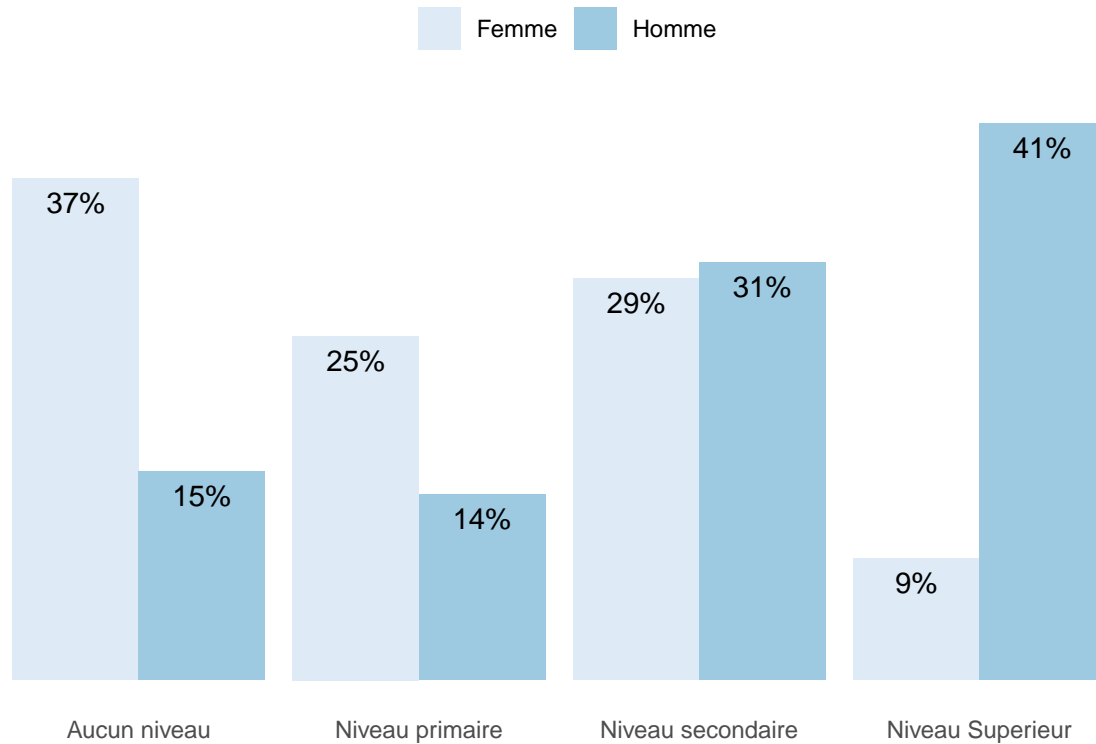
```
projet %>% bivarie( projet$q12, projet$sexe)
```

GIE selon Femme



```
projet %>% bivarie( projet$q25, projet$sexe)
```


Aucun niveau selon Femme



```
ggplot(projet) +
  aes(
    x = q12,
    y = after_stat(prop),
    fill = sexe,
    by = as_factor(sexe),
    label = scales::percent(after_stat(prop), accuracy = 1)
  ) +
  geom_bar(
    stat = "prop",
    position = position_dodge(.9)
  ) +
  geom_text(
    aes(y = after_stat(prop) - 0.01),
    stat = "prop",
    position = position_dodge(.9),
    vjust = "top"
  ) +
  scale_y_continuous(labels = scales::percent)+

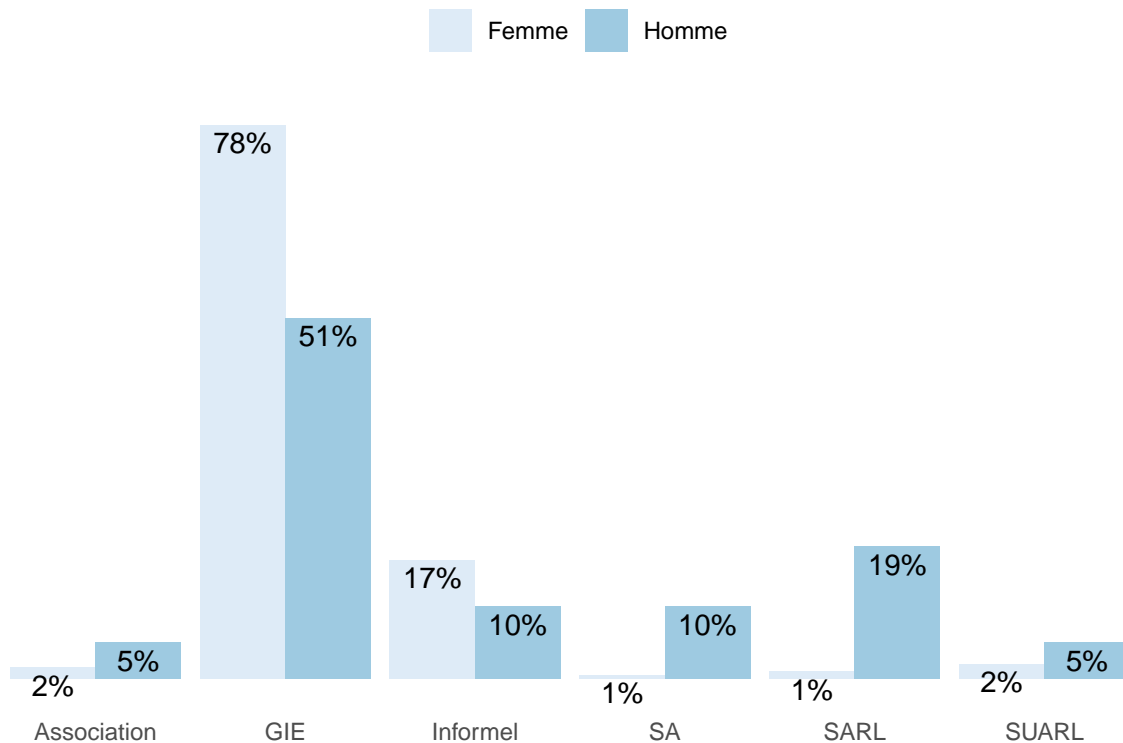
  theme_light() +
  xlab("") +
  ylab("") +
  labs(fill = "") +
  ggtitle(paste0("q12", " selon ", "sexe")) +
  theme(
```

```

panel.grid = element_blank(),
panel.border = element_blank(),
axis.text.y = element_blank(),
axis.ticks = element_blank(),
legend.position = "top"
) +
scale_fill_brewer()

```

q12 selon sexe



3 Un peu de cartographie

-Transformation en données géographiques

```

projet_map <- st_as_sf(projet, coords = c("gps_menlongitude", "gps_menlatitude"), crs = 4326)
class(projet_map)

```

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

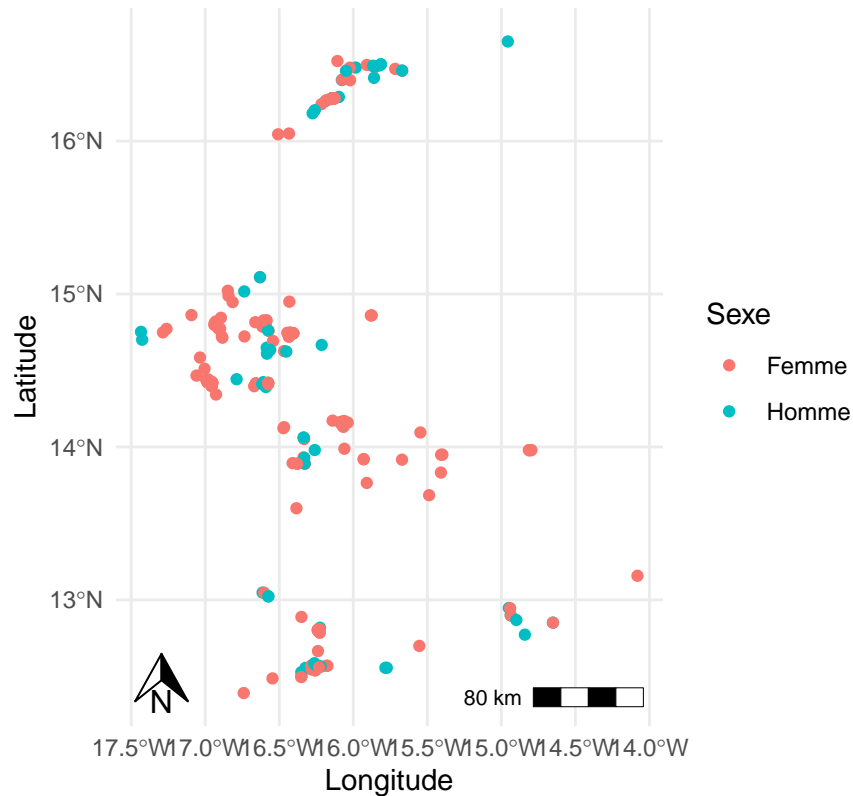
- Retrouver le code :

```

ggplot(data = projet_map) +
  geom_sf(aes(color = sexe)) +
  labs(title = "Répartition géographique des PME",
        x = "Longitude", y = "Latitude",
        color = "Sexe") +
  annotation_scale(location = "br", line_width = .5) +
  annotation_north_arrow(location = "bl", height = unit(0.7, "cm"), width = unit(0.7, "cm")) +
  theme_minimal()

```

Répartition géographique des PME



- Importation de SEN_adm et representation suivant les niveaux

```
SEN_adm0 <- st_read("data/gadm41_SEN_0.shp")
```

```
## Reading layer `gadm41_SEN_0' from data source
##   `C:\Users\LENOVO\Desktop\isep3\semestre2\Projet statistique sous R\TP 1\data\gadm41_SEN_0.shp'
##   using driver `ESRI Shapefile'
## Simple feature collection with 1 feature and 2 fields
## Geometry type: MULTIPOLYGON
## Dimension:      XY
## Bounding box:  xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS:  WGS 84
```

```
SEN_adm1 <- st_read("data/gadm41_SEN_1.shp")
```

```
## Reading layer `gadm41_SEN_1' from data source
##   `C:\Users\LENOVO\Desktop\isep3\semestre2\Projet statistique sous R\TP 1\data\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
```

```
SEN_adm2 <- st_read("data/gadm41_SEN_2.shp")
```

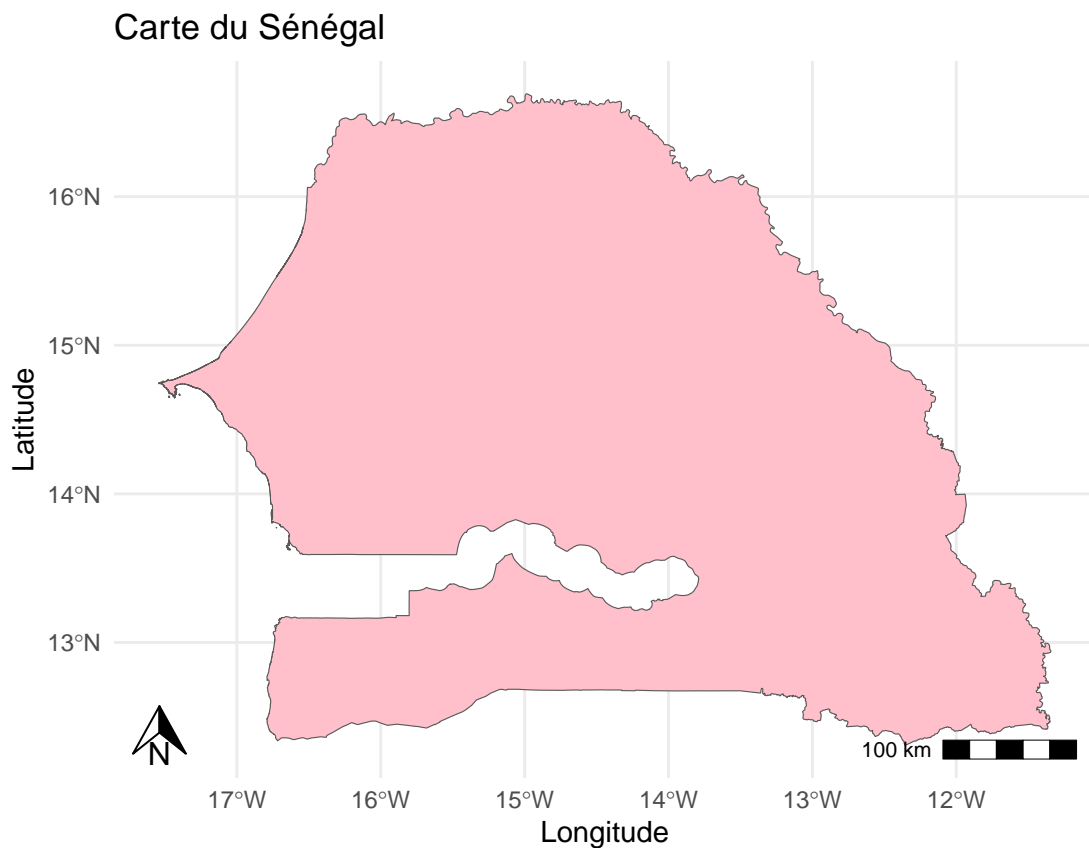
```
## Reading layer `gadm41_SEN_2' from data source
```

```
## `C:\Users\LENOVO\Desktop\isep3\semestre2\Projet statistique sous R\TP 1\data\gadm41_SEN_2.shp'
## using driver `ESRI Shapefile'
## Simple feature collection with 45 features and 13 fields
## Geometry type: MULTIPOLYGON
## Dimension: XY
## Bounding box: xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS: WGS 84

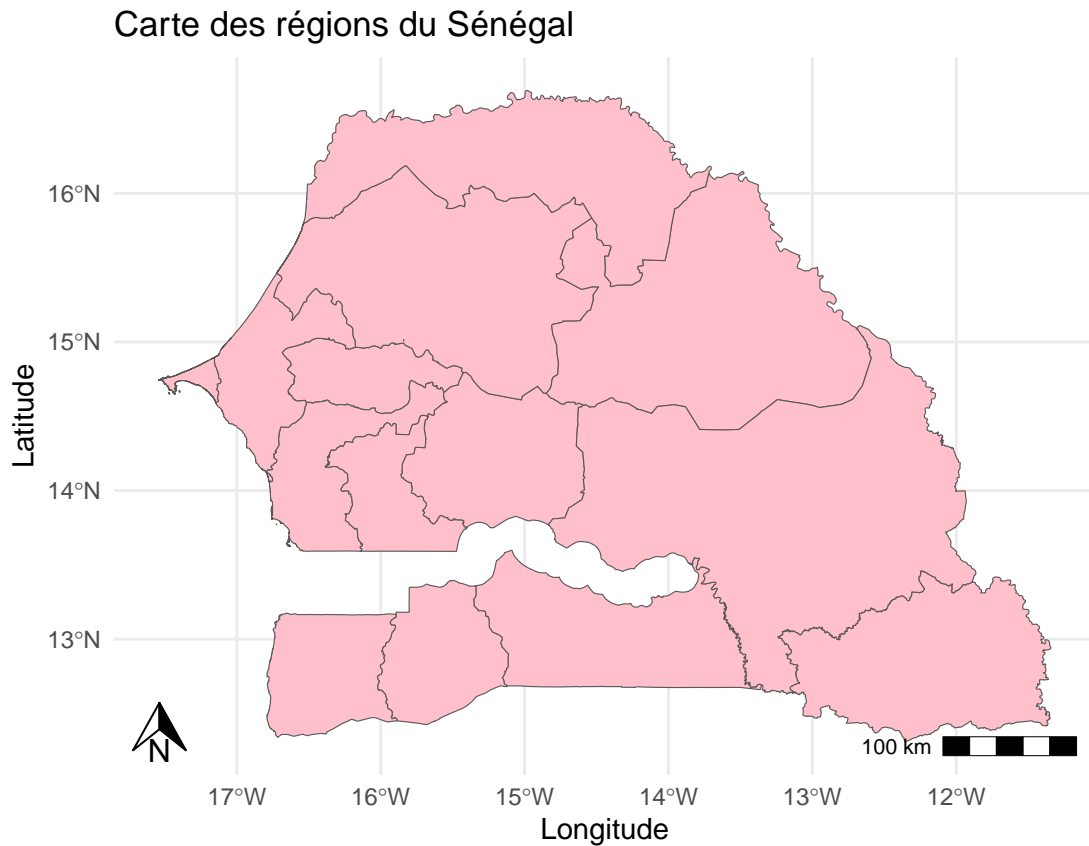
SEN_adm3 <- st_read("data/gadm41_SEN_3.shp")

## Reading layer `gadm41_SEN_3' from data source
## `C:\Users\LENOVO\Desktop\isep3\semestre2\Projet statistique sous R\TP 1\data\gadm41_SEN_3.shp'
## using driver `ESRI Shapefile'
## Simple feature collection with 123 features and 16 fields
## Geometry type: MULTIPOLYGON
## Dimension: XY
## Bounding box: xmin: -17.54319 ymin: 12.30786 xmax: -11.34247 ymax: 16.69207
## Geodetic CRS: WGS 84

ggplot(data = SEN_adm0) +
  geom_sf(fill = "pink") +
  labs(title = "Carte du Sénégal",
       x = "Longitude", y = "Latitude") +
  annotation_scale(location = "br", line_width = .5) +
  annotation_north_arrow(location = "bl", height = unit(0.7, "cm"), width = unit(0.7, "cm")) +
  theme_minimal()
```

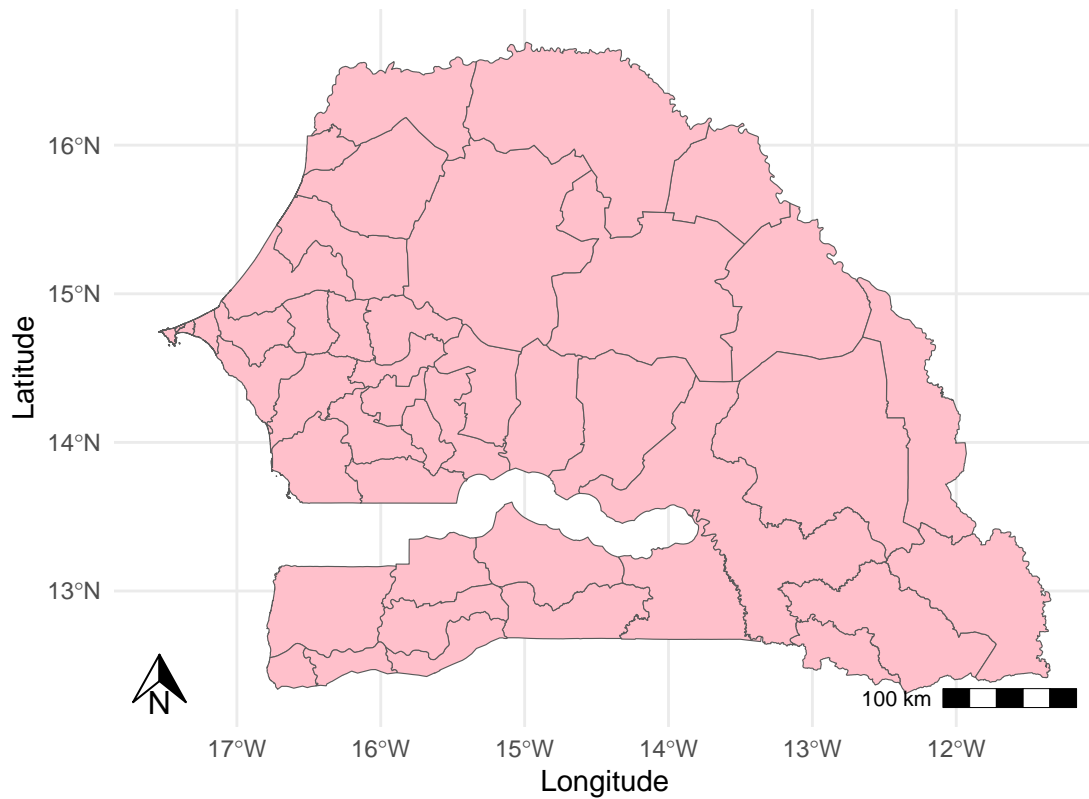


```
ggplot(data = SEN_adm1) +
  geom_sf(fill = "pink") +
  labs(title = "Carte des régions du Sénégal",
       x = "Longitude", y = "Latitude") +
  annotation_scale(location = "br", line_width = .5) +
  annotation_north_arrow(location = "bl", height = unit(0.7, "cm"), width = unit(0.7, "cm")) +
  theme_minimal()
```



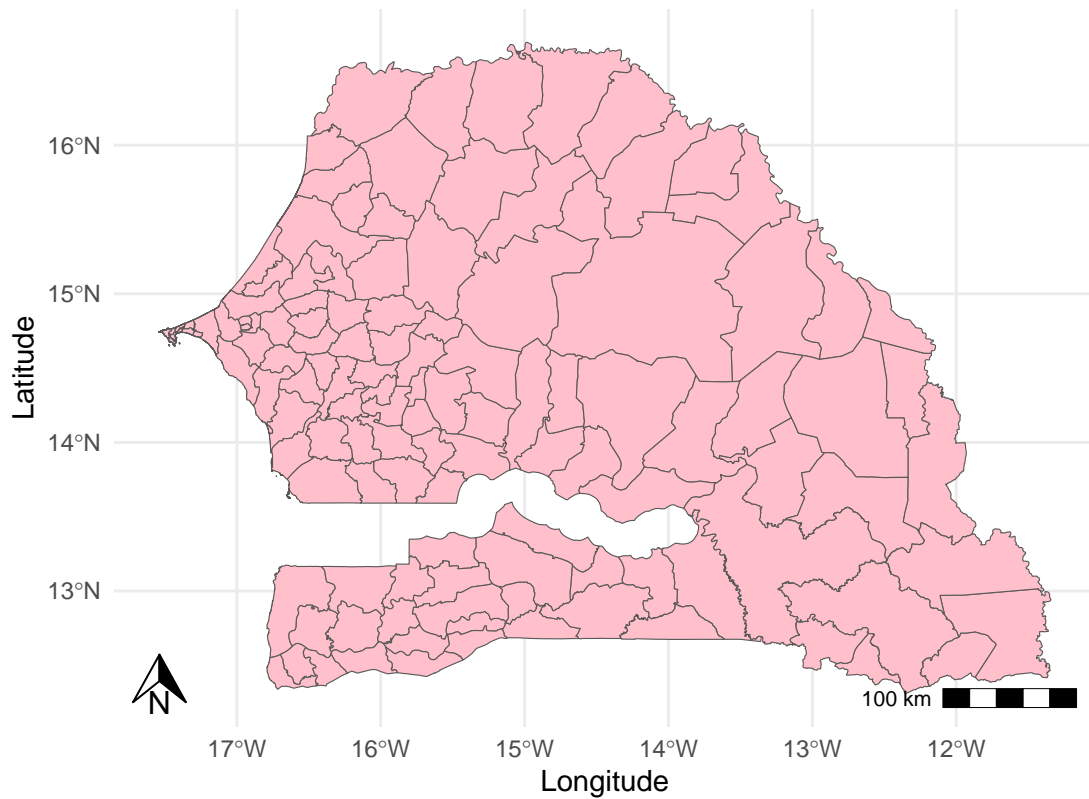
```
ggplot(data = SEN_adm2) +
  geom_sf(fill = "pink") +
  labs(title = "Carte des départements du Sénégal",
       x = "Longitude", y = "Latitude") +
  annotation_scale(location = "br", line_width = .5) +
  annotation_north_arrow(location = "bl", height = unit(0.7, "cm"), width = unit(0.7, "cm")) +
  theme_minimal()
```

Carte des départements du Sénégal



```
ggplot(data = SEN_adm3) +
  geom_sf(fill = "pink") +
  labs(title = "Carte des communes du Sénégal",
        x = "Longitude", y = "Latitude") +
  annotation_scale(location = "br", line_width = .5) +
  annotation_north_arrow(location = "bl", height = unit(0.7, "cm"), width = unit(0.7, "cm")) +
  theme_minimal()
```

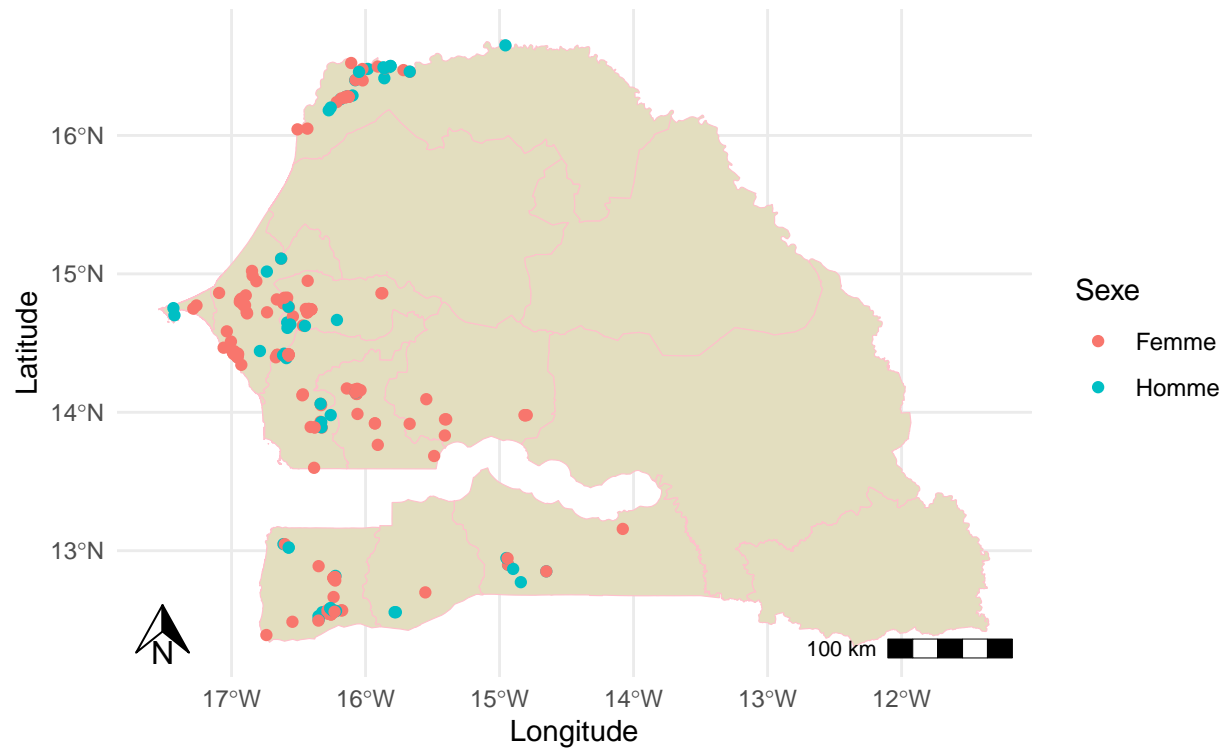
Carte des communes du Sénégal



- Représentation spatiale des PME suivant le sexe :

```
ggplot() +  
  geom_sf(data = SEN_adm1, col = "pink" , fill = "#E3DEBF" ) +  
  geom_sf(data = projet_map, aes(color = sexe)) +  
  labs(title = "Répartition géographique des PME",  
        x = "Longitude", y = "Latitude",  
        color = "Sexe") +  
  annotation_scale(location = "br", line_width = .5) +  
  annotation_north_arrow(location = "bl", height = unit(0.7, "cm"), width = unit(0.7, "cm")) +  
  theme_minimal()
```

Répartition géographique des PME



- Représentation spatiale des PME suivant le niveau d'instruction :

```
ggplot() +
  geom_sf(data = SEN_adm1, col = "pink" , fill = "#E3DEBF" )+
  geom_sf(data = projet_map, aes(color = q25)) +
  labs(title = "Répartition géographique des PME",
        x = "Longitude", y = "Latitude",
        color = "Niveau d'instruction") +
  annotation_scale(location = "br", line_width = .5) +
  annotation_north_arrow(location = "bl", height = unit(0.7, "cm"), width = unit(0.7, "cm"))+
  theme_minimal()
```


Répartition géographique des PME

