

# COURS DE PROJET STATISTIQUE AVEC R : TP1

DIOP Ousseynou (ISEP 3)

2024-04-08

## Contents

Bibliothèques . . . . .	1
<b>Préparation des données</b>	<b>2</b>
Description des données . . . . .	2
Importation et mise en forme . . . . .	2
Création de variable . . . . .	2
<b>Analyses descriptives</b>	<b>3</b>
<b>Un peu de cartographie</b>	<b>9</b>
Transformation de la base en données géographiques . . . . .	9
Première carte . . . . .	10
Avec le Sénégal . . . . .	11
Carte brute Sénégal . . . . .	11
Carte Sénégal niveau région . . . . .	12
Carte Sénégal niveau département . . . . .	13
Carte Sénégal niveau commune . . . . .	14
Représentation spatiale des PME suivant le sexe . . . . .	15
Représentation spatiale des PME suivant le niveau d'instruction . . . . .	16
Autre Statistique spatiale . . . . .	17

## Bibliothèques

```
library(haven)
library(readxl)
library(dplyr)
library(ggplot2)
library(sf)
library(ggspatial)
```

# Préparation des données

## Description des données

Voir tp

## Importation et mise en forme

```
way <- here::here() #chemin relatif de notre base de données
past <- paste0(way, "/Base_Projet.xlsx") #chemin complet pour acceder a la base de données
projet_nd <- read_xlsx(past) #importation de la base
projet <- data.frame(projet_nd) #transformation de la base en data_frame
```

Pour connaitre le nombre de lignes (i.e. le nombre de PME) et colonnes (i.e. nombre de variables) de la base projet, il est possible d'utiliser les fonctions comme **dim** ou **str**.

```
dim(projet) # donne respectivement le nombre de ligne et de colonne
```

```
## [1] 250 33
```

Le nombre de PME dans la base de données *projet* est de 250, de même le nombre de variables est de 33.

**NB** : la fonction *str* donne les caractéristiques de la base (des variables).

Vérifions s'il existe des valeurs manquantes pour la variable *key*.

```
unique(is.na(projet$key)) #la fonction is.na vérifie s'il y a des valeurs manquantes sur la variable key
```

```
## [1] FALSE
```

Le résultat montre que toutes les observations ont un identifiant.

## Création de variable

```
#renommons q1 en region
projet <- projet %>%
  rename(region = q1,
         departement = q2,
         sexe = q23) #la fonction rename renomme la variable q1, q2 et q23 de la base projet en region,

#creation de la variable sexe2
projet$sexe2 <- ifelse(projet$sexe=="Femme",1,0) # si la valeur de sexe est Femme alors met 1 pour ma v

#création du data_frame langue

langue <- projet %>%
  select(key,q24a_1,q24a_2,q24a_3,q24a_4,q24a_5,q24a_6,q24a_7,q24a_9,q24a_10) #Selection des variables
```

```

#creation de la variable parle

langue <- langue %>%
  mutate(parle =q24a_1+q24a_2+q24a_3+q24a_4+q24a_5+q24a_6+q24a_7+q24a_9+q24a_10) #La fonction mutate pe

#selection de parle et key
langue <- langue %>%
  select(key,parole)

#Merge des df langue et projet

projet_lg <- merge(projet,langue, by="key") #la variable Key est la variable clé de la fusion des deux

```

## Analyses descriptives

```

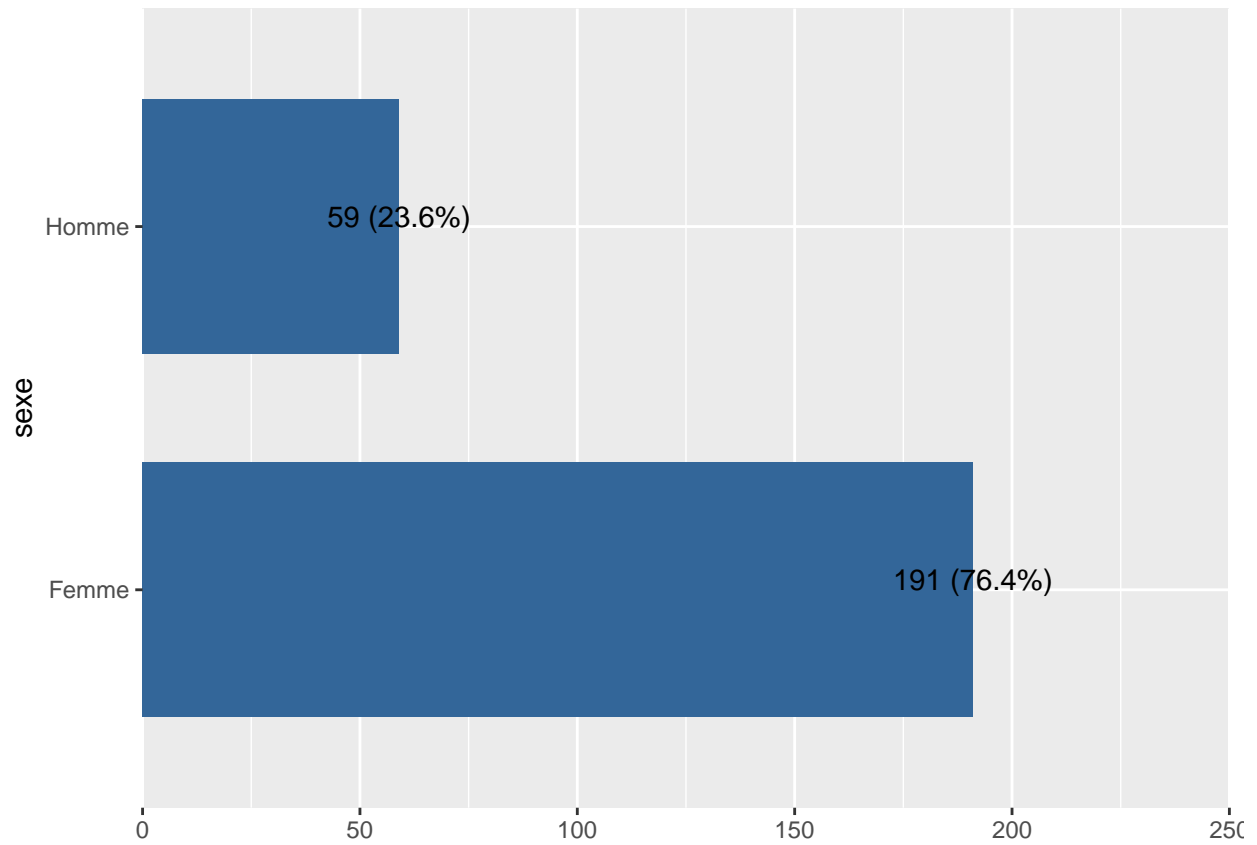
univarie <- function(data, var, include_na = TRUE) {
  if (!is.data.frame(data)) {
    stop("la base de données n'est pas un dataframe")
  }
  nom <- deparse(substitute(var))#Ici nous avons recuperer uniquement le nom de la variable pour éviter
  if (!nom %in% names(data)) {
    stop("cette variable n'existe pas dans la base")
  }

  if (!include_na) {
    data <- na.omit(data)#supprimer les ligne qui ont des NA
  }

  # Calcul des statistiques descriptives
  summary_stats <- summary(data[[nom]])
  # Tracé des fréquences
  graphe <- sjPlot::plot_frq(data, coord.flip = TRUE, nom, show.na = include_na)
  show(graphe)
  return(summary_stats)
}

```

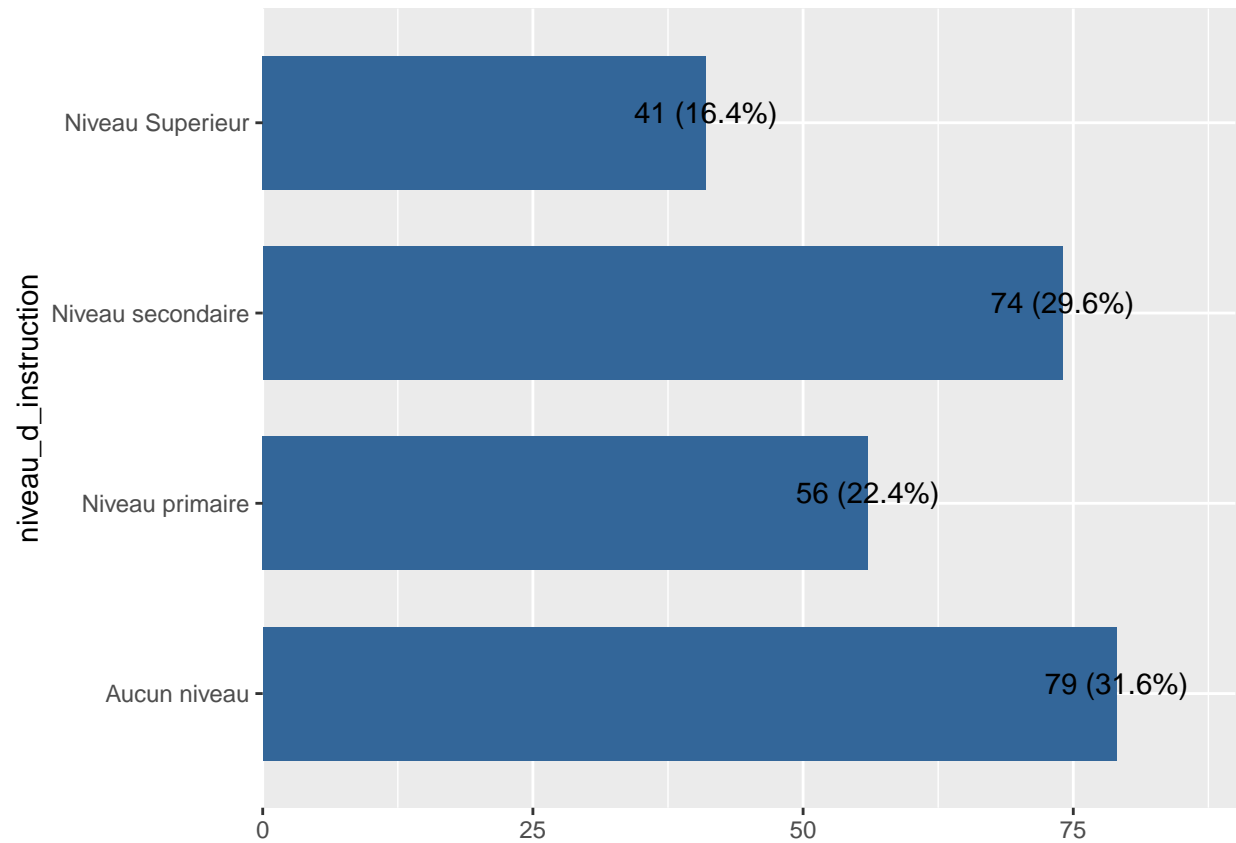
```
univarie(projet_lg,sexe)
```



```
##      Length      Class    Mode  
##         250 character character
```

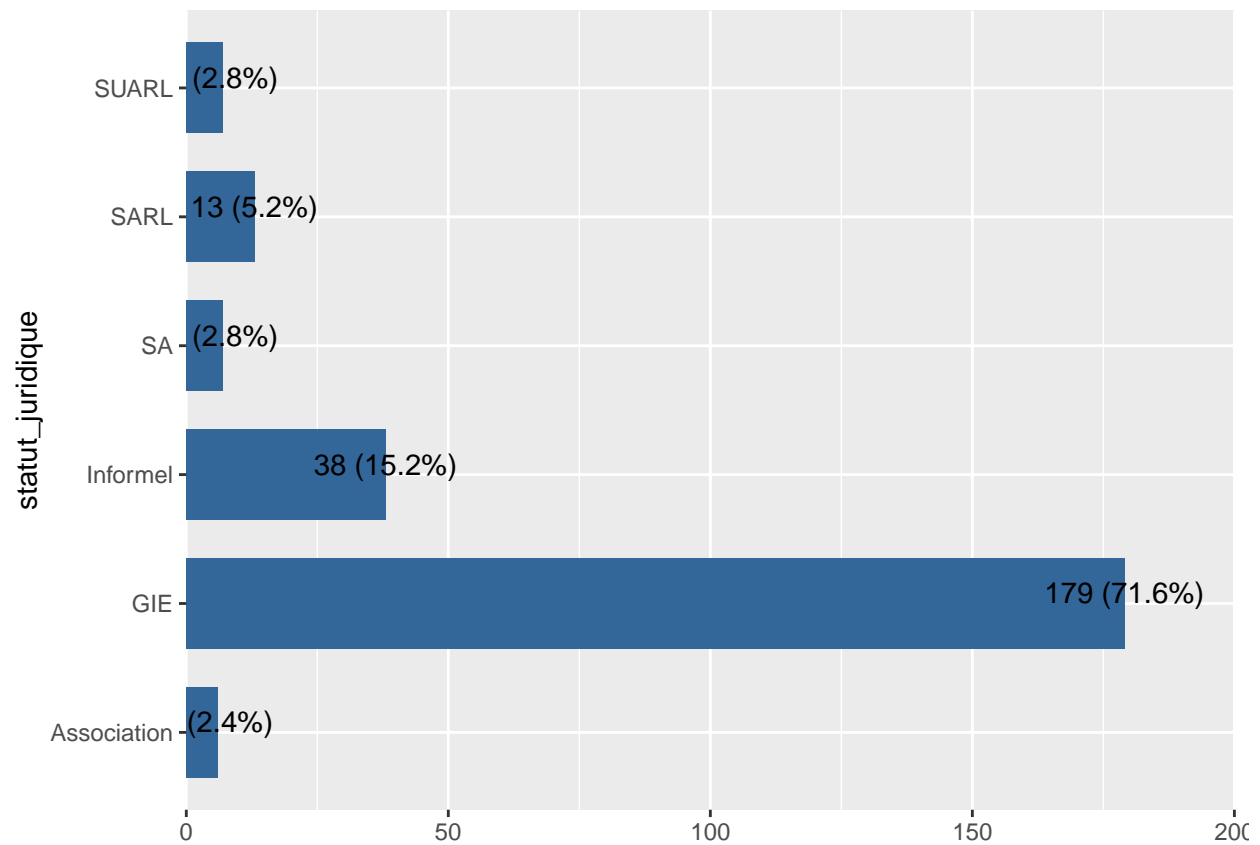
```
projet_lg <- projet_lg %>%  
  rename(niveau_d_instruction=q25)
```

```
univarie(projet_lg,niveau_d_instruction)#appel de la fonction univarie
```



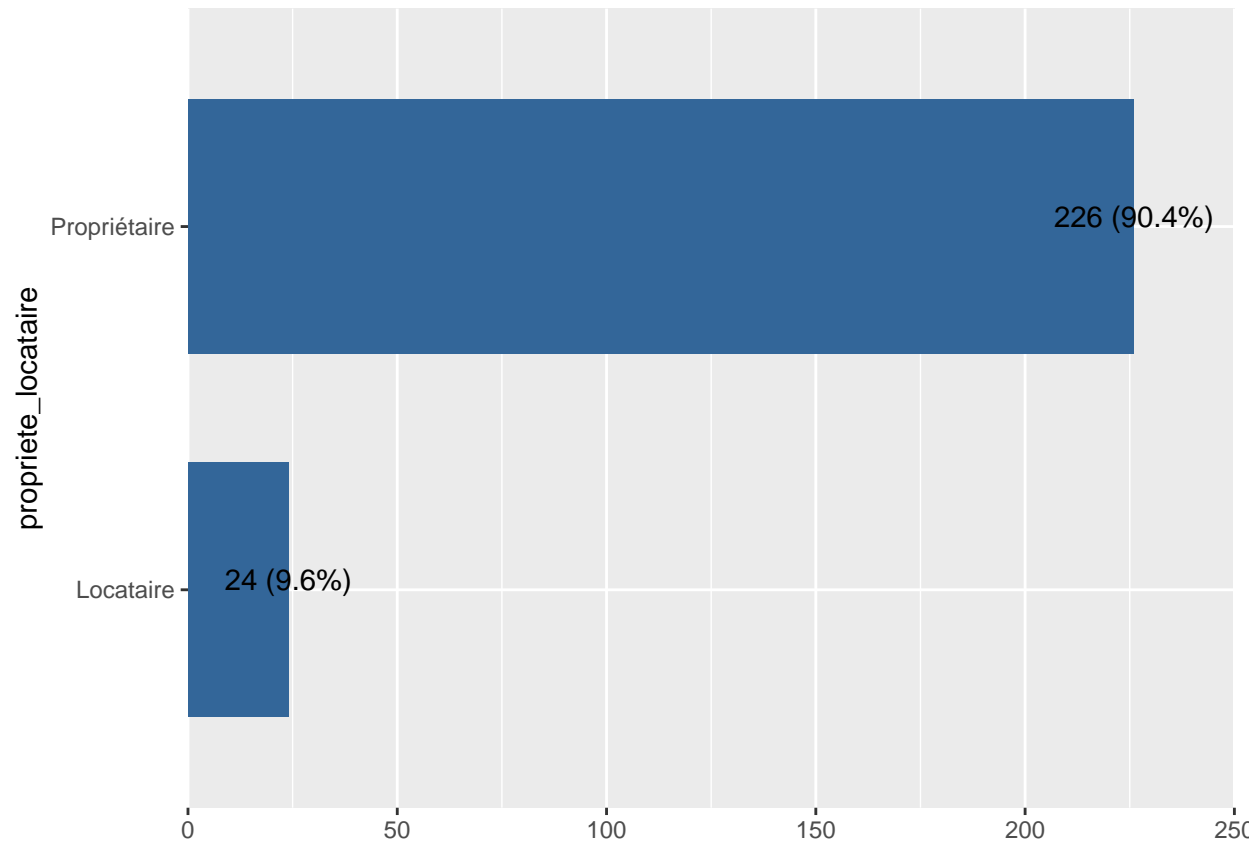
```
##      Length      Class    Mode
##      250 character character
```

```
projet_lg <- projet_lg %>%
  rename(statut_juridique=q12)
univarie(projet_lg,statut_juridique)
```



```
##      Length      Class    Mode
##      250 character character
```

```
projet_lg <- projet_lg %>%
  rename(propriete_locataire=q81)
univariate(projet_lg,propiete_locataire)
```



```
##      Length      Class      Mode
##          250 character character
```

```
bivarie <- function(data, var1, var2) {
  if (!is.data.frame(data)) {
    stop("La base de données n'est pas un dataframe")
  }

  var1_nom <- deparse(substitute(var1)) # Ici nous avons recuperer uniquement le nom de la variable pour
  if (!var1_nom %in% names(data)) {
    stop("La variable", var1_nom, "n'existe pas dans la base de données")
  }

  var2_nom <- deparse(substitute(var2))
  if (!var2_nom %in% names(data)) {
    stop("La variable", var2_nom, "n'existe pas dans la base de données")
  }

  # Extraction des variables
  var1_data <- data[[var1_nom]]
  var2_data <- data[[var2_nom]]

  # Tableau croisé pour les deux variables avec fréquences
  cross_table1 <- addmargins(table(var1_data, var2_data)) # ajouter les sommes marginales avec addmargins
  cross_table <- addmargins(prop.table(table(var1_data, var2_data)) * 100, margin = 1)
```

```

# Affichage du tableau croisé avec fréquences en pourcentage

cat("\nTableau croisé entre", var1_nom, "et", var2_nom, "avec fréquences:\n")
print(cross_table1)
cat("\nTableau croisé entre", var1_nom, "et", var2_nom, "avec fréquences en pourcentage:\n")
return(list(cross_table = cross_table))
}

```

```
bivarie(projet_lg,statut_juridique,sexe)
```

```

##
## Tableau croisé entre statut_juridique et sexe avec fréquences:
##      var2_data
## var1_data  Femme Homme Sum
## Association    3     3   6
## GIE            149    30 179
## Informel       32     6  38
## SA              1     6   7
## SARL           2    11  13
## SUARL          4     3   7
## Sum           191    59 250
##
## Tableau croisé entre statut_juridique et sexe avec fréquences en pourcentage:

```

```

## $cross_table
##      var2_data
## var1_data  Femme Homme
## Association  1.2   1.2
## GIE          59.6  12.0
## Informel     12.8   2.4
## SA           0.4   2.4
## SARL         0.8   4.4
## SUARL        1.6   1.2
## Sum         76.4  23.6

```

```
bivarie(projet_lg,niveau_d_instruction,sexe)
```

```

##
## Tableau croisé entre niveau_d_instruction et sexe avec fréquences:
##      var2_data
## var1_data  Femme Homme Sum
## Aucun niveau    70     9  79
## Niveau primaire  48     8  56
## Niveau secondaire 56    18  74
## Niveau Supérieur 17    24  41
## Sum           191    59 250
##
## Tableau croisé entre niveau_d_instruction et sexe avec fréquences en pourcentage:

```

```

## $cross_table
##      var2_data

```



```
## var1_data      Femme Homme
##   Aucun niveau    28.0   3.6
##   Niveau primaire  19.2   3.2
##   Niveau secondaire 22.4   7.2
##   Niveau Supérieur   6.8   9.6
##   Sum              76.4  23.6
```

```
bivarie(projet_lg,propiete_locataire,sexe)
```

```
##
## Tableau croisé entre propriete_locataire et sexe avec fréquences:
##           var2_data
## var1_data      Femme Homme Sum
##   Locataire       16     8  24
##   Propriétaire    175    51 226
##   Sum             191    59 250
##
## Tableau croisé entre propriete_locataire et sexe avec fréquences en pourcentage:
```

```
## $cross_table
##           var2_data
## var1_data      Femme Homme
##   Locataire       6.4   3.2
##   Propriétaire    70.0  20.4
##   Sum             76.4  23.6
```

## Un peu de cartographie

### Transformation de la base en données géographiques

```
projet_map<- sf::st_as_sf(projet_lg, coords = c("gps_menlongitude", "gps_menlatitude")) #changer les do
class(projet_map) #class de la base de données
```

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

```
sf::st_crs(projet_map) <- 4326 #système de coordonnées
str(projet_map)
```

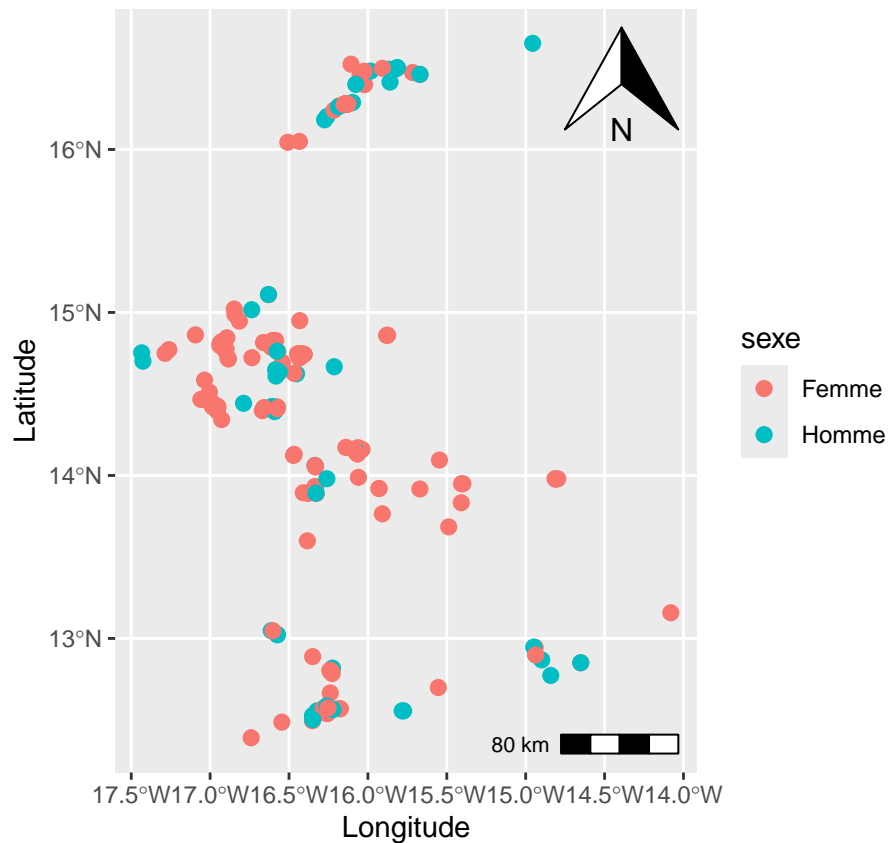
```
## Classes 'sf' and 'data.frame':  250 obs. of  34 variables:
## $ key                : chr  "uuid:004b9117-d180-4031-a6af-6b4efabb5f53" "uuid:007d8eb4-45eb-44f4-a
## $ region             : chr  "Diourbel" "Thiès" "Saint-Louis" "Diourbel" ...
## $ departement        : chr  "Bambey" "Tivaouane" "Dagana" "Mbacké" ...
## $ sexe               : chr  "Femme" "Femme" "Homme" "Femme" ...
## $ q24                : num  62 60 58 60 63 61 45 45 65 60 ...
## $ q24a_1             : num  0 0 1 1 0 1 0 0 1 1 ...
## $ q24a_2             : num  1 1 1 1 1 1 1 1 1 1 ...
## $ q24a_3             : num  0 0 0 0 0 0 0 0 0 0 ...
## $ q24a_4             : num  0 1 0 0 0 0 1 1 0 0 ...
## $ q24a_5             : num  0 0 0 0 1 0 0 0 0 0 ...
```

```
## $ q24a_6 : num 0 0 0 0 0 0 0 0 0 0 ...
## $ q24a_7 : num 0 0 0 0 0 0 0 0 0 0 ...
## $ q24a_9 : num 0 0 0 0 0 0 0 0 0 0 ...
## $ q24a_10 : num 0 0 0 0 0 0 0 0 0 1 ...
## $ niveau_d_instruction: chr "Aucun niveau" "Niveau secondaire" "Niveau secondaire" "Niveau primaire" ...
## $ q26 : num 20 10 30 25 21 25 12 7 5 30 ...
## $ statut_juridique : chr "GIE" "GIE" "GIE" "GIE" ...
## $ q14b : chr "Non" "Non" "Non" "Oui" ...
## $ q16 : chr "Non" "Oui" "Non" "Oui" ...
## $ q17 : chr NA "Bon état" NA "Bon état" ...
## $ q19 : chr "Mauvais état" NA "Mauvais état" NA ...
## $ q20 : chr "Oui" "Non" "Oui" "Oui" ...
## $ filiere_1 : num 1 1 0 1 0 0 0 0 1 0 ...
## $ filiere_2 : num 0 0 0 0 0 0 1 1 0 0 ...
## $ filiere_3 : num 0 0 1 0 1 1 0 0 1 1 ...
## $ filiere_4 : num 0 1 0 0 0 0 0 0 0 0 ...
## $ q8 : chr "Aucun" "Transformation d'autres céréales" "Transformation du riz" "Transformation d'autres produits" ...
## $ propriete_locataire : chr "Propriétaire" "Locataire" "Propriétaire" "Locataire" ...
## $ submissiondate : POSIXct, format: "2021-06-05 15:33:51" "2021-06-15 01:10:46" ...
## $ start : POSIXct, format: "2021-06-04 15:14:14" "2021-06-08 14:40:28" ...
## $ today : POSIXct, format: "2021-06-04" "2021-06-08" ...
## $ sexe2 : num 1 1 0 1 0 1 1 1 1 0 ...
## $ parle : num 1 2 2 2 2 2 2 2 2 3 ...
## $ geometry : sfc_POINT of length 250; first list element: 'XY' num -16.6 14.8
## - attr(*, "sf_column")= chr "geometry"
## - attr(*, "agr")= Factor w/ 3 levels "constant","aggregate",...: NA NA NA NA NA NA NA NA NA ...
## ..- attr(*, "names")= chr [1:33] "key" "region" "departement" "sexe" ...
```

## Premiere carte

```
graphe_1 <- ggplot(projet_map) +
  aes(fill = sexe, colour = sexe) +
  geom_sf(size = 2.35) +
  scale_fill_manual(values = c(femme = "#F8766D", homme = "#61FFEC")) +
  xlab("Longitude") +
  ylab("Latitude") +
  annotation_scale(location = "br") +
  annotation_north_arrow(location = "tr", which_north = "true")

graphe_1
```



## Avec le Sénégal

```
#IMPORTATION DES DONNEES DU SENEGAL
past_sn0 <- paste0(way, "/gadm41_SEN_0.shp")
past_sn1 <- paste0(way, "/gadm41_SEN_1.shp")
past_sn2 <- paste0(way, "/gadm41_SEN_2.shp")
past_sn3 <- paste0(way, "/gadm41_SEN_3.shp")

sn_map0 <- read_sf(past_sn0)
sn_map1 <- read_sf(past_sn1)
sn_map2 <- read_sf(past_sn2)
sn_map3 <- read_sf(past_sn3)
```

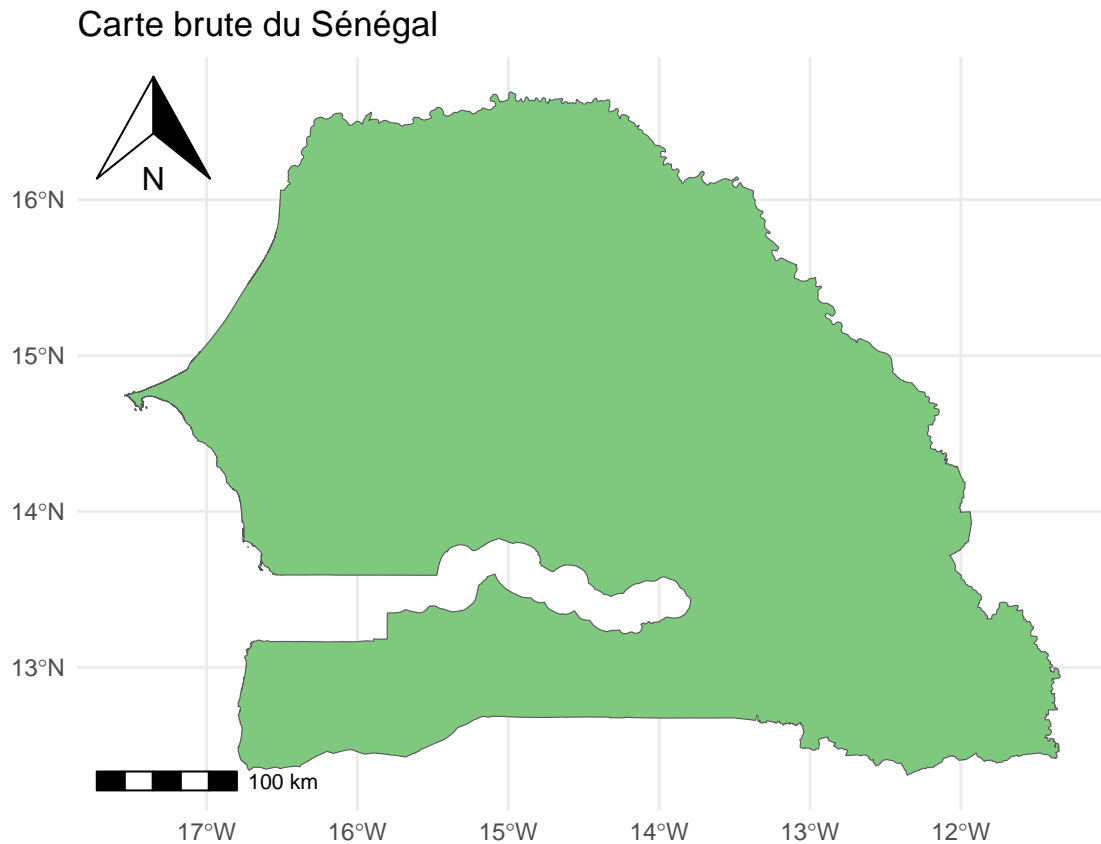
## Carte brute Sénégal

```
library(ggplot2)

# Créer la carte brute du Sénégal
carte_0 <- ggplot(sn_map0) +
  aes(fill = COUNTRY) +
  geom_sf(size = 1.2) +
  scale_fill_brewer(palette = "Accent", direction = 1) +
```

```
labs(title = "Carte brute du Sénégal") +
theme_minimal() +
theme(legend.position = "none") +
annotation_scale(location = "bl") +
annotation_north_arrow(location = "tl", which_north = "true")

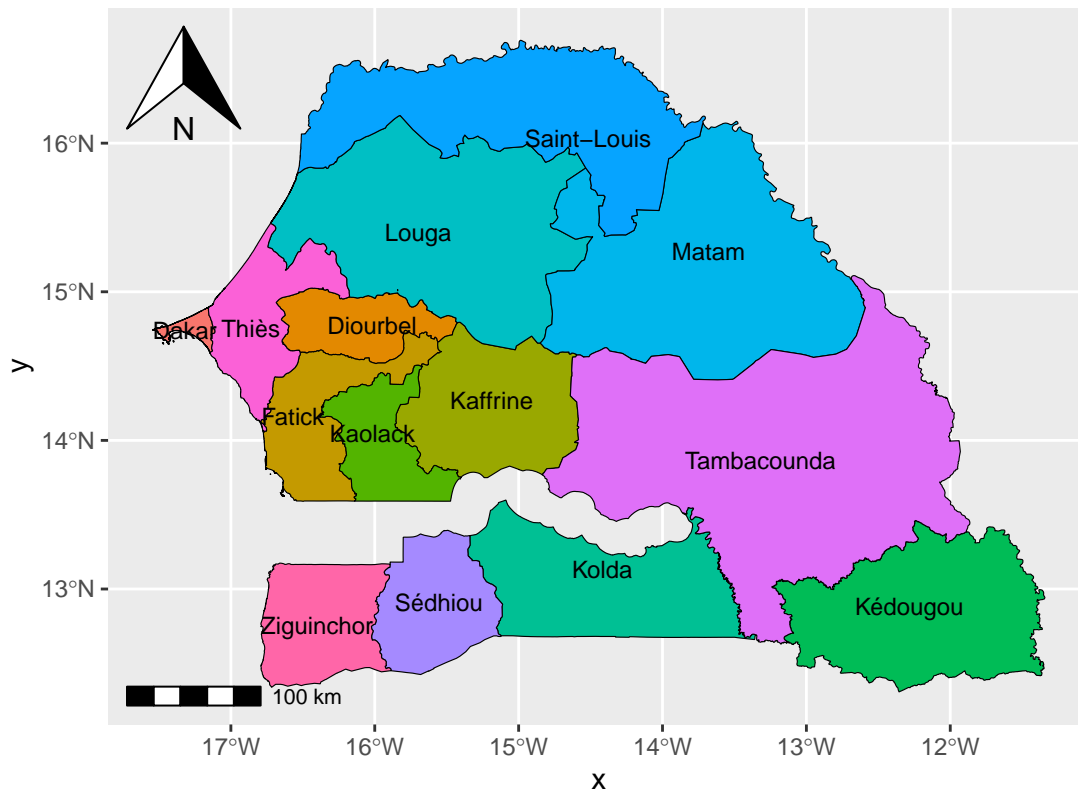
print(carte_0)
```



Carte Sénégal niveau région

```
carte_1 <- ggplot() +
  geom_sf(data = sn_map1, aes(fill = NAME_1), color = "black", size = 0.8) +
  geom_sf_text(data = sn_map1, aes(label = NAME_1), size = 3, color = "black") + # Ajouter les noms de
  theme(legend.position = "none")+ # Masquer la légende
  labs(title = "Carte du Sénégal niveau region") +
  annotation_scale(location = "bl") +
  annotation_north_arrow(location = "tl", which_north = "true")
print(carte_1)
```

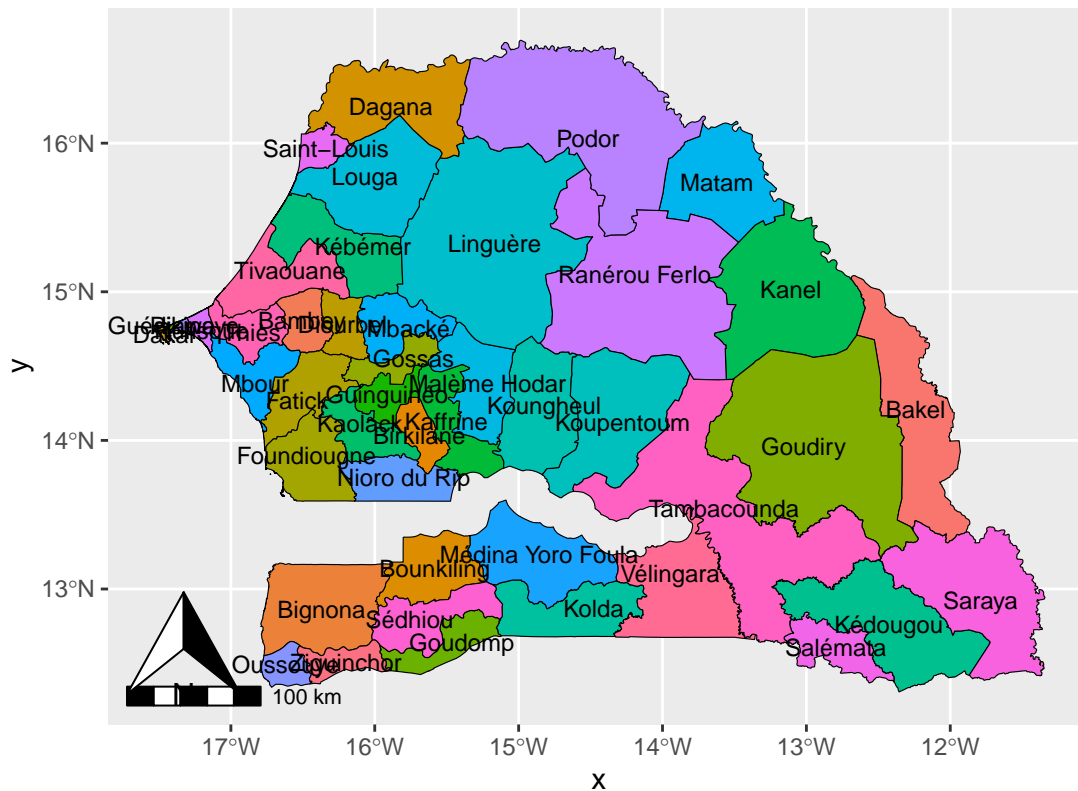
Carte du Sénégal niveau region



Carte Sénégal niveau département

```
carte_2 <- ggplot() +
  geom_sf(data = sn_map2, aes(fill = NAME_2), color = "black", size = 0.8) +
  geom_sf_text(data = sn_map2, aes(label = NAME_2), size = 3, color = "black") + # Ajouter les noms de
  theme(legend.position = "none")+ # Masquer la légende
  labs(title = "Carte du Sénégal niveau departement") +
  annotation_scale(location = "bl") +
  annotation_north_arrow(location = "bl", which_north = "true")
print(carte_2)
```

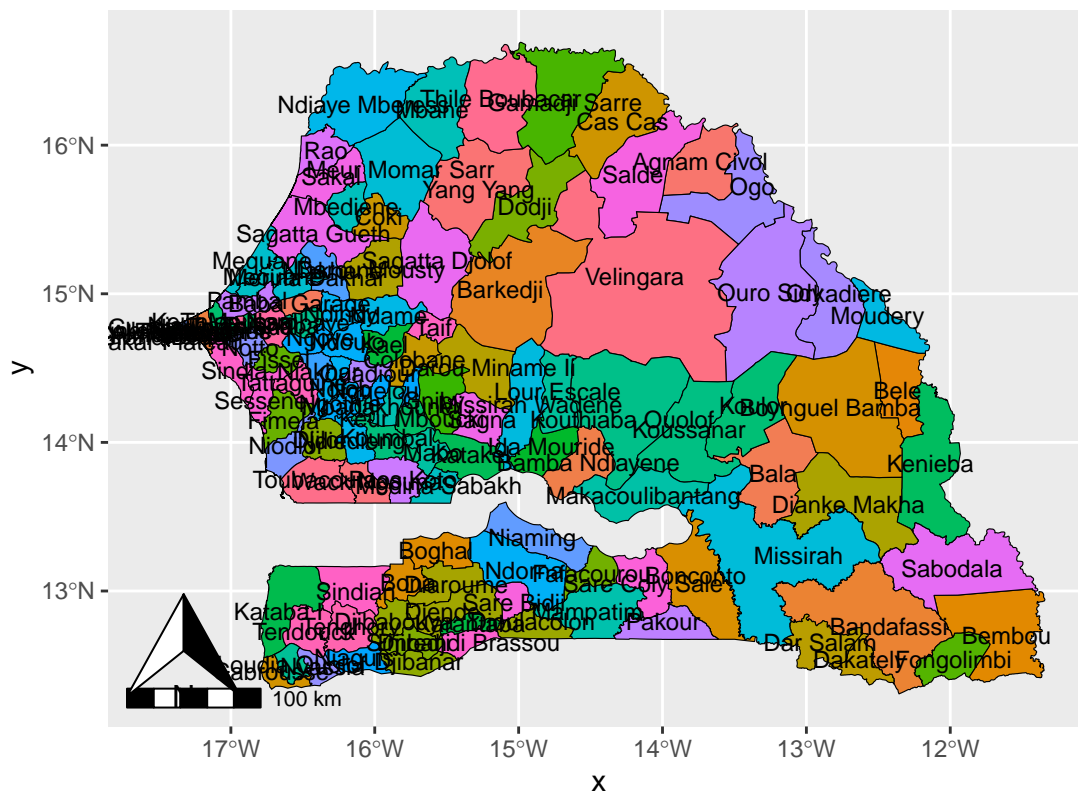
Carte du Sénégal niveau departement



Carte Sénégal niveau commune

```
carte_3 <- ggplot() +
  geom_sf(data = sn_map3, aes(fill = NAME_3), color = "black", size = 0.8) +
  geom_sf_text(data = sn_map3, aes(label = NAME_3), size = 3, color = "black") + # Ajouter les noms de
  theme(legend.position = "none")+ # Masquer la légende
  labs(title = "Carte Sénégal niveau commune") +
  annotation_scale(location = "bl") +
  annotation_north_arrow(location = "bl", which_north = "true")
print(carte_3)
```

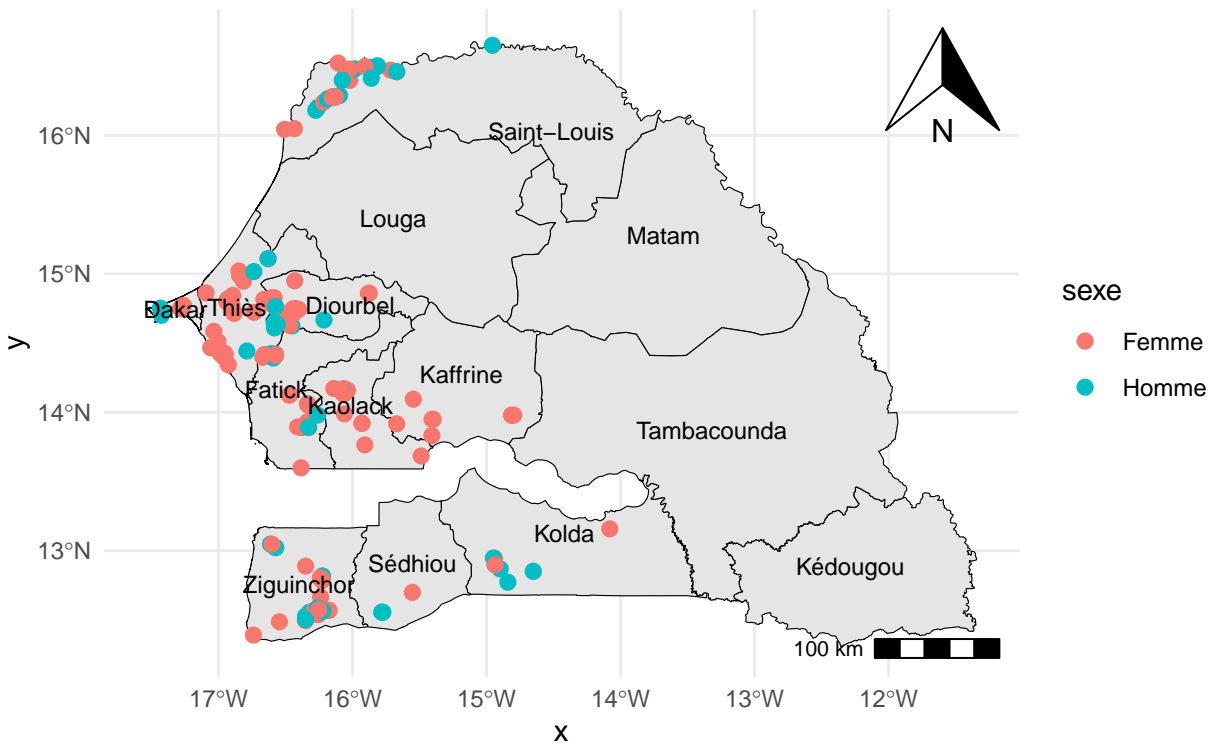
## Carte Sénégal niveau commune



## Réprésentation spatiale des PME suivant le sexe

```
carte_sen_sex <- ggplot()+
  geom_sf(data=sn_map1, color = "black", size = 0.8)+
  theme(legend.position = "none") +
  geom_sf(data=projet_map,aes(fill = sexe, colour = sexe),size = 2.35)+
  scale_fill_manual(values = c(femme = "#F8766D",
  homme = "#61FFEC")) +
  labs(title = "Représentation spatiale des PME suivant le sexe")+
  geom_sf_text(data = sn_map1, aes(label = NAME_1), size = 3, color = "black") +
  annotation_scale(location = "br") +
  annotation_north_arrow(location = "tr", which_north = "true")+
  theme_minimal()
carte_sen_sex
```

## Représentation spatiale des PME suivant le sexe

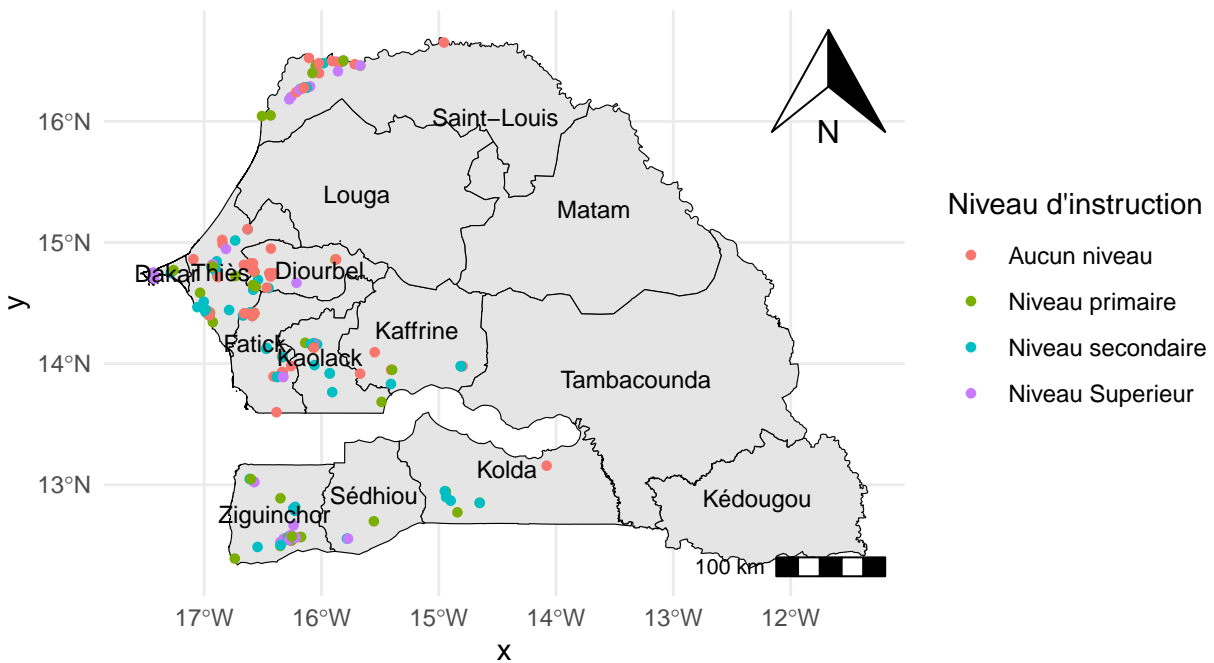


## Réprésentation spatiale des PME suivant le niveau d'instruction

```
carte_sen_niv_ins <- ggplot()+
  geom_sf(data=sn_map1, color = "black", size = 0.8)+
  theme(legend.position = "none") +
  geom_sf(data=projet_map,aes(colour = niveau_d_instruction),size = 1.2)+
  geom_sf_text(data = sn_map1, aes(label = NAME_1), size = 3, color = "black") +
  scale_color_hue(direction = 1,name = "Niveau d'instruction") +
  annotation_scale(location = "br") +
  labs(title = "Représentation spatiale des PME suivant le niveau d instruction")+
  annotation_north_arrow(location = "tr", which_north = "true")+
  theme_minimal()
carte_sen_niv_ins
```



## Représentation spatiale des PME suivant le niveau d'instruction



## Autre Statistique spatiale

```
carte_sen_pro_loc <- ggplot()+
  geom_sf(data=sn_map1, color = "black", size = 0.8)+
  theme(legend.position = "none") +
  geom_sf(data=projet_map,aes(colour = propriete_locataire),size = 1.2)+
  geom_sf_text(data = sn_map1, aes(label = NAME_1), size = 3, color = "black") +
  scale_color_hue(direction = 1,name = "Propriété/Locataire") +
  annotation_scale(location = "br") +
  labs(title = "Représentation spatiale des PME suivant les propriétaires ou locataires")+
  annotation_north_arrow(location = "tr", which_north = "true")+
  theme_minimal()
carte_sen_pro_loc
```

## Représentation spatiale des PME suivant les propriétaires ou locataires

