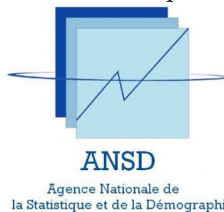


RÉPUBLIQUE DU SÉNÉGAL  
Un peuple- un But- une Foi



\*\*\*\*\*

Agence Nationale de la Statistique et de la Démographie



École Nationale de la Statistique et de l'Analyse Économique Pierre Ndiaye



Projet statistique avec R

---

TP 2 de R

---

*Rédigé par :*

ZAONGO INOUSSA

ÉLÈVES INGÉNIEUR STATISTICIEN ÉCONOMISTE

*Sous la supervision de :*

M.Hema Aboubacar

INGÉNIEUR DES TRAVAUX STATISTIQUE

©Avril-2024

# Sommaire

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Importation et mise en forme</b>	<b>3</b>
2.1	Importation des données . . . . .	3
2.2	Nombre de lignes et de colonnes . . . . .	4
<b>3</b>	<b>Recodage et Labelisation</b>	<b>4</b>
3.1	Recodage des variables . . . . .	4
3.2	Labelisation des variables . . . . .	9
<b>4</b>	<b>Analyses descriptives</b>	<b>10</b>
4.1	Analyses descriptives univariés . . . . .	10
4.2	Analyses descriptives bivariées . . . . .	21
<b>5</b>	<b>Conclusion</b>	<b>31</b>

```
#liste des library
```

```
library(labelled)
library(readxl)
library(here)
```

```
## here() starts at C:/Users/LENOVO/Desktop/ISEP3_année scolaire 2023-2024/SEMESTRE 6/Statistique_5/proj
```

```
library(ggplot2)
library(knitr)
library(ggplot2)
library(dplyr)
```

## 1 Introduction

L'agriculture, en tant que pilier fondamental de nos sociétés, joue un rôle crucial dans la fourniture de denrées alimentaires, la préservation de l'environnement et le développement économique. Dans ce contexte, la collecte et l'analyse de données agricoles revêtent une importance capitale pour comprendre les tendances, évaluer les performances et orienter les politiques agricoles.

Le projet présenté dans ce rapport se concentre sur la labélisation et le recodage des variables, ainsi que sur l'analyse de diverses données agricoles. L'unité statistique principale utilisée est la production agricole, qui constitue une mesure essentielle pour évaluer les performances et les tendances dans le secteur agricole.

Ce projet vise à fournir une méthodologie robuste pour la gestion des données agricoles, en mettant l'accent sur la standardisation et la cohérence des variables. La labélisation et le recodage des variables permettent de rendre les données plus compréhensibles et faciles à analyser, facilitant ainsi la prise de décision et la communication des résultats.

Dans ce rapport, nous présenterons d'abord les objectifs du projet, puis nous décrirons les méthodes utilisées pour la labélisation et le recodage des variables. Ensuite, nous explorerons différentes analyses de variables agricoles, mettant en lumière les tendances, les corrélations et les insights pertinents pour les décideurs politiques, les chercheurs et les acteurs du secteur agricole.

## 2 Importation et mise en forme

### 2.1 Importation des données

```
getwd() # Pour savoir dans quel répertoire nous travaillons
```

```
## [1] "C:/Users/LENOVO/Desktop/ISEP3_année scolaire 2023-2024/SEMESTRE 6/Statistique_5/projet_statistique"
```

```
library(here) # Pour rendre le code portable entre différents environnements et systèmes d'exploitation
```

```
base_tp2 = readxl::read_xlsx(here("Base TP2.xlsx"))
```

```
str(base_tp2) # Pour voir la structure de la base de données
```

```
## tibble [53 x 30] (S3: tbl_df/tbl/data.frame)
```

```
## $ region : num [1:53] 5 5 2 1 5 3 3 2 2 4 ...
```

```
## $ departement : num [1:53] 53 53 22 12 52 31 32 22 22 41 ...
```

```
## $ sexe : num [1:53] 2 1 2 1 2 1 2 1 1 1 ...
```

```
## $ age : num [1:53] 35 50 35 25 60 36 25 56 55 80 ...
```

```
## $ sit_mat : num [1:53] 1 1 1 1 1 1 1 1 1 1 ...
```

```
## $ si_chef_men : num [1:53] 1 2 1 3 1 1 3 2 2 2 ...
```

```
## $ ethnie      : num [1:53] 10 1 1 3 77 3 3 1 1 2 ...
## $ occupation  : num [1:53] 22 1 1 22 1 1 15 1 1 1 ...
## $ formation   : num [1:53] 4 1 3 5 3 99 1 1 1 1 ...
## $ niveau_alphabs : num [1:53] NA 0 NA NA NA NA 1 0 0 0 ...
## $ types_varietes : chr [1:53] "2" "1" "1" "1" ...
## $ types_varietes_1: num [1:53] 0 1 1 1 1 1 1 1 0 1 ...
## $ types_varietes_2: num [1:53] 1 0 0 0 0 1 0 0 1 0 ...
## $ criteres_var   : chr [1:53] "1 12 13" "1 4 6 7 11 13" "1 5 6" "1 4 6 7 14 15" ...
## $ criteres_var_1 : num [1:53] 1 1 1 1 1 1 1 0 1 1 ...
## $ criteres_var_2 : num [1:53] 0 0 0 0 1 1 0 0 1 0 ...
## $ criteres_var_3 : num [1:53] 0 0 0 0 1 0 0 0 0 0 ...
## $ criteres_var_4 : num [1:53] 0 1 0 1 1 0 1 0 0 0 ...
## $ criteres_var_5 : num [1:53] 0 0 1 0 0 0 0 0 0 0 ...
## $ criteres_var_6 : num [1:53] 0 1 1 1 1 0 0 0 0 0 ...
## $ criteres_var_7 : num [1:53] 0 1 0 1 1 0 0 0 0 0 ...
## $ criteres_var_8 : num [1:53] 0 0 0 0 1 0 0 0 0 0 ...
## $ criteres_var_9 : num [1:53] 0 0 0 0 1 0 0 0 0 0 ...
## $ criteres_var_10: num [1:53] 0 0 0 0 0 0 0 0 0 0 ...
## $ criteres_var_11: num [1:53] 0 1 0 0 1 0 0 1 1 1 ...
## $ criteres_var_12: num [1:53] 1 0 0 0 0 0 0 0 0 0 ...
## $ criteres_var_13: num [1:53] 1 1 0 0 1 1 0 0 0 0 ...
## $ criteres_var_14: num [1:53] 0 0 0 1 1 1 0 0 0 0 ...
## $ criteres_var_15: num [1:53] 0 0 0 1 0 0 0 0 0 0 ...
## $ criteres_var_16: num [1:53] 0 0 0 0 0 0 0 0 0 0 ...
```

## 2.2 Nombre de lignes et de colonnes

```
Nb_ligne = nrow(base_tp2) # pour compter le nombre de ligne
Nb_colone = ncol(base_tp2) # pour compter le nombre de colonne

print(paste("le nombre de ligne est : ", Nb_ligne))

## [1] "le nombre de ligne est : 53"

print(paste("le nombre de colone est : ", Nb_colone))

## [1] "le nombre de colone est : 30"
```

## 3 Recodage et Labelisation

La phase de recodage et de labelisation consiste à restructurer et à attribuer des valeurs significatives aux différentes modalités des variables dans une base de données. Cette étape permet d’améliorer la lisibilité et l’interprétation des données, facilitant ainsi les analyses ultérieures. En assignant des libellés clairs et compréhensibles, cette démarche vise à rendre les variables plus accessibles et informatives pour les utilisateurs.

### 3.1 Recodage des variables

Nous allons recoder toutes les variables de la base en suivant la description donnée dans le fichier word. Il faut constater que la variable “types-varietes” qui est une variable à choix multiple a été disloqué en deux variables types-varietes-1 et types-varietes-2 donc nous allons recoder ces deux variables en 1 pour “Oui” et 0 pour “Non” même si elle ne sont pas décrite dans le fichier word. De même la variable “criteres-var” a été disloqué en 16 variables et nous allons recoder ses variables comme suit : 1 pour “Oui” et 0 pour “Non” plutôt que de recoder la variable elle-même qui est un condensé de ses variables. En s’efforçant de recoder cette variable, les modalités de la variables risquent d’être encombrant et incompréhensible.

### 3.1.1 transformation des variables en facteur

Nous allons transformer les variables qui sont de type numeric en facteur d'abord car Cela permet de s'assurer que les valeurs numériques sont traitées comme des catégories plutôt que des valeurs continues.

```
colonnes_a_convertir <- c("sexe", "sit_mat", "si_chef_men", "ethnie", "occupation", "formation", "niveau_alp  
  
base_tp2[colonnes_a_convertir] <- lapply(base_tp2[colonnes_a_convertir], as.factor) # Transforme toutes
```

### 3.1.2 Recodage de la variable Sexe

```
base_tp2$sexe = dplyr::recode_factor(  
  base_tp2$sexe,  
  `1` = "Homme",  
  `2` = "Femme"  
)
```

### 3.1.3 Recodage de la variable sit-mat

```
base_tp2$sit_mat = dplyr::recode_factor(  
  base_tp2$sit_mat,  
  `1` = "Marié(e)",  
  `2` = "Veuf(ve)",  
  `3` = "Divorcé(e)",  
  `4` = "Séparé(e)",  
  `5` = "Célibataire"  
)
```

### 3.1.4 Recodage de la variable si-chef-men

```
base_tp2$si_chef_men = dplyr::recode_factor(  
  base_tp2$si_chef_men,  
  `1` = "Femme du chef de ménage",  
  `2` = "Chef du ménage",  
  `3` = "fils-fille du chef de ménage",  
  `99` = "Autres"  
)
```

### 3.1.5 Recodage de la variable ethnie

```
base_tp2$ethnie = dplyr::recode_factor(  
  base_tp2$ethnie,  
  `1` = "Wolof",  
  `2` = "Pulaar/Toucouleur",  
  `3` = "Sérère",  
  `4` = "Mandika/Bambara",  
  `5` = "Soninké",  
  `6` = "Diola",  
  `7` = "Manjack",  
  `8` = "Bainouk",  
  `9` = "Maures",
```

```

`10` = "Balante",
`77` = "Autre"

)

```

### 3.1.6 Recodage de la variable occupation

Nous remarquons que dans la variable “occupation” il y’a des modalités qui ne correspondent pas aux modalités décrites dans le fichier word donc nous avons décidé de les recoder en NA pour faciliter la suite du travail. Il s’agit dans ce cas de recoder les valeurs proprement dit , c’est à dire changer les valeurs des modalités.

```

base_tp2$occupation <- dplyr::recode(
  base_tp2$occupation,
  `1` = "1",
  `2` = "2",
  `3` = "3",
  `4` = "4",
  `5` = "5",
  `6` = "6",
  .default = NA_character_ # Remplacer les modalités non recodées par NA
)

```

Ici maintenant nous allons faire le recodage proprement parlant en donnant du sens aux modalités.

```

base_tp2$occupation = dplyr::recode_factor(
  base_tp2$occupation,
  `1` = "Agriculture, Elevage,Sylviculture,Pêche",
  `2` = "Activités extractives",
  `3` = "Activités de fabrication (Artisanat)",
  `4` = "Activité de transformation",
  `5` = "Production et distribution d'électricité et de gaz",
  `6` = "Production et distribution d'eau,assainissement,traitement d
  .default = NA_character_ # Remplacer les modalités non recodées pa
)

```

### 3.1.7 Recodage de la variable formation

```

base_tp2$formation = dplyr::recode_factor(
  base_tp2$formation,
  `1` = "Non scolarisé",
  `2` = "Elémentaire",
  `3` = "Moyen",
  `4` = "Secondaire",
  `5` = "Licence",
  `6` = "Master",
  `7` = "Doctorat",
  `99` = "Ne sait pas"
)

```

### 3.1.8 Recodage de la variable niveau\_alphabs

```
base_tp2$niveau_alphabs = dplyr::recode_factor(
  base_tp2$niveau_alphabs,
  `0` = "Sans niveau",
  `1` = "Sait lire dans une langue",
  `2` = "Sait lire et écrire dans une langue"
)
```

### 3.1.9 Recodage de la variable types-varietes

```
base_tp2$types_varietes = dplyr::recode_factor(
  base_tp2$types_varietes,
  `1` = "Traditionnelles",
  `2` = "Améliorées",
  `1 2` = "Traditionnelles et Améliorées "
)
```

### 3.1.10 Recodage de la variable “types-varietes-1” et “types-varietes-2”

```
base_tp2$types_varietes_1 = dplyr::recode_factor(
  base_tp2$types_varietes_1,
  `0` = "Non",
  `1` = "Oui"
)

base_tp2$types_varietes_2 = dplyr::recode_factor(
  base_tp2$types_varietes_2,
  `0` = "Non",
  `1` = "Oui"
)
```

### 3.1.11 Recodage de la variable criteres-var

On constate que les modalités de la variables criteres-var ce sont disloqués en des variables dichotomique donc nous allons les recoder toutes et laisser le recodage de la variable criteres-var elle même qui semble être complexe de par sa structure.

```
base_tp2$criteres_var_1 = dplyr::recode_factor(
  base_tp2$criteres_var_1,
  `0` = "Non",
  `1` = "Oui"
)

base_tp2$criteres_var_2 = dplyr::recode_factor(
  base_tp2$criteres_var_2,
  `0` = "Non",
  `1` = "Oui"
)

base_tp2$criteres_var_3 = dplyr::recode_factor(
  base_tp2$criteres_var_3,
  `0` = "Non",
  `1` = "Oui"
)
```

```

    )
base_tp2$criteres_var_4 = dplyr::recode_factor(
  base_tp2$criteres_var_4,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_5 = dplyr::recode_factor(
  base_tp2$criteres_var_5,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_6 = dplyr::recode_factor(
  base_tp2$criteres_var_6,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_7 = dplyr::recode_factor(
  base_tp2$criteres_var_7,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_8 = dplyr::recode_factor(
  base_tp2$criteres_var_8,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_9 = dplyr::recode_factor(
  base_tp2$criteres_var_9,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_10 = dplyr::recode_factor(
  base_tp2$criteres_var_10,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_11 = dplyr::recode_factor(
  base_tp2$criteres_var_11,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_12 = dplyr::recode_factor(
  base_tp2$criteres_var_12,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_13 = dplyr::recode_factor(
  base_tp2$criteres_var_13,
  `0` = "Non",
  `1` = "Oui"
)
base_tp2$criteres_var_14 = dplyr::recode_factor(
  base_tp2$criteres_var_14,

```



```

        `0` = "Non",
        `1` = "Oui"
    )
base_tp2$criteres_var_15 = dplyr::recode_factor(
    base_tp2$criteres_var_15,
    `0` = "Non",
    `1` = "Oui"
)
base_tp2$criteres_var_16 = dplyr::recode_factor(
    base_tp2$criteres_var_16,
    `0` = "Non",
    `1` = "Oui"
)

```

## 3.2 Labelisation des variables

La labellisation des variables est une pratique fondamentale en analyse de données qui consiste à attribuer des étiquettes descriptives aux variables d'un ensemble de données. Cela permet de rendre les données plus compréhensibles et accessibles, et facilite leur manipulation et leur analyse.

```

var_label(base_tp2$sexe) <- "Sexe"
var_label(base_tp2$age) <- "Age"
var_label(base_tp2$sit_mat) <- "Situation matrimoniale"
var_label(base_tp2$si_chef_men) <- "Statut dans le ménage"
var_label(base_tp2$ethnie) <- "Ethnie"
var_label(base_tp2$occupation) <- "Occupation"
var_label(base_tp2$formation) <- "Formation"
var_label(base_tp2$niveau_alphabs) <- "Niveau d'alphabétisation"
var_label(base_tp2$criteres_var) <- "Quelles sont les critères de choix des variétés de sésame?"
var_label(base_tp2$types_varietes) <- "Quelles sont les variétés que vous utilisez pour la production de sésame?"

var_label(base_tp2$criteres_var_1) <- "Quelles sont les critères de choix des variétés de sésame? : rendement"
var_label(base_tp2$criteres_var_2) <- "Quelles sont les critères de choix des variétés de sésame? : tailles des graines"
var_label(base_tp2$criteres_var_3) <- "Quelles sont les critères de choix des variétés de sésame? : résistantes aux maladies/ravageurs"
var_label(base_tp2$criteres_var_4) <- "Quelles sont les critères de choix des variétés de sésame? : tolérantes aux sécheresses"
var_label(base_tp2$criteres_var_5) <- "Quelles sont les critères de choix des variétés de sésame? : rendements élevés"
var_label(base_tp2$criteres_var_6) <- "Quelles sont les critères de choix des variétés de sésame? : tolérantes aux inondations"
var_label(base_tp2$criteres_var_7) <- "Quelles sont les critères de choix des variétés de sésame? : faible charge de travail"
var_label(base_tp2$criteres_var_8) <- "Quelles sont les critères de choix des variétés de sésame? : facile à transformer"
var_label(base_tp2$criteres_var_9) <- "Quelles sont les critères de choix des variétés de sésame? : haut teneur en huile"
var_label(base_tp2$criteres_var_10) <- "Quelles sont les critères de choix des variétés de sésame? : haut rendement après transformation"
var_label(base_tp2$criteres_var_11) <- "Quelles sont les critères de choix des variétés de sésame? : demande sur le marché"
var_label(base_tp2$criteres_var_12) <- "Quelles sont les critères de choix des variétés de sésame? : autres"

```

```

Bon goût"
var_label(base_tp2$criteres_var_13) <- "Quelles sont les critères de choix des variétés de sésame? :
Belle couleur"
var_label(base_tp2$criteres_var_14) <- "Quelles sont les critères de choix des variétés de sésame? :
Haut rendement en fourrages"
var_label(base_tp2$criteres_var_15) <- "Quelles sont les critères de choix des variétés de sésame? :
Qualité du fourrage"
var_label(base_tp2$criteres_var_16) <- "Quelles sont les critères de choix des variétés de sésame? :
Autres à spécifier"

var_label(base_tp2$types_varietes_1) <- "Quelles sont les variétés que vous utilisez pour la production
Traditionnelles"
var_label(base_tp2$types_varietes_2) <- "Quelles sont les variétés que vous utilisez pour la production
Améliorées"

```

## 4 Analyses descriptives

### 4.1 Analyses descriptives univariés

Dans cette partie nous allons essayer de créer des fonctions pour faire les analyses descriptives.

```

univariate_stats <- function(data, variable) {
  # Vérifier si la variable est factorielle
  if (!is.factor(data[[variable]])) {
    stop("La variable doit être de type factoriel")
  }

  # Calculer les statistiques
  stats <- table(data[[variable]])
  stats_percentage <- prop.table(stats) * 100

  # Créer le tableau des statistiques
  stats_table <- data.frame(Nombre = as.numeric(stats),
                           Pourcentage = stats_percentage,
                           Variable = names(stats))

  # Afficher le tableau avec kable
  cat("Tableau des statistiques univariées pour la variable", variable, ":\n")
  print(kable(stats_table, format = "markdown"))

  # Générer un graphique circulaire avec ggplot2 si la variable a deux modalités
  if (length(levels(data[[variable]])) == 2) {
    # Ajouter les pourcentages au dataframe
    stats_table$prop <- paste(round(stats_percentage, 2), "%")

    # Tracer le graphique circulaire
    pie_chart <- ggplot(stats_table, aes(x = "", y = Nombre, fill = Variable)) +
      geom_bar(stat = "identity", width = 1) +
      coord_polar("y", start = 0) +
      geom_text(aes(label = prop), position = position_stack(vjust = 0.5), color = "white") +
      theme_void() +
      labs(title = paste("Graphique circulaire de", variable)) +
      labs(fill = "Légende") # Modifier le titre de la légende
  }
}

```

```

print(pie_chart)
} else {
  # Tracer un graphique en barres si la variable a plus de deux modalités
  if (length(levels(data[[variable]])) > 2) {
    stats_table$prop <- paste(round(stats_percentage, 2), "%")
    bar_chart <- ggplot(stats_table, aes(x = Variable, y = Nombre, fill = Variable)) +
      geom_col() +
      geom_text(aes(label = prop), position = position_stack(vjust = 0.5), color = "black")+
      labs(title = paste("Graphique en barres de", variable)) +
      labs(x = "Variable", y = "Nombre") +
      labs(fill = "Légende") + # Modifier le titre de la légende
      theme_minimal()

    print(bar_chart)
  } else {
    cat("La variable n'a pas exactement deux modalités, le graphique circulaire n'est pas généré.\n")
  }
}
}

```

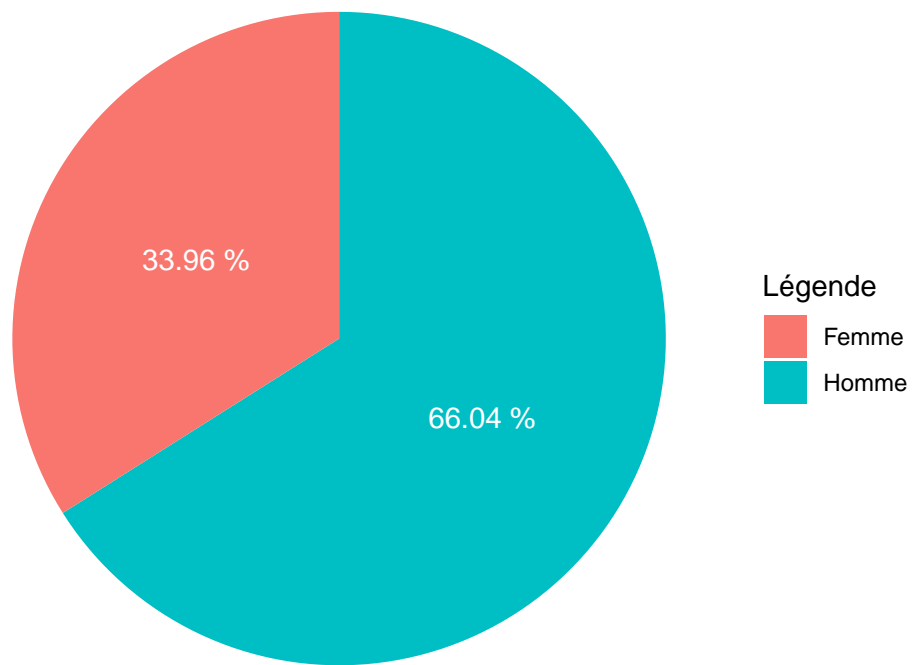
#### 4.1.1 Statistique univarié sur les variables en appelant la fonction univariate\_stats

```
univariate_stats(base_tp2, "sexe")
```

```
## Tableau des statistiques univariées pour la variable sexe :
##
##
## | Nombre|Pourcentage.Var1 | Pourcentage.Freq|Variable |
## |-----:|:-----|-----:|:-----|
## |      35|Homme          |          66.03774|Homme     |
## |      18|Femme          |          33.96226|Femme     |

```

## Graphique circulaire de sexe



```
univariate_stats(base_tp2,"sit_mat")
```

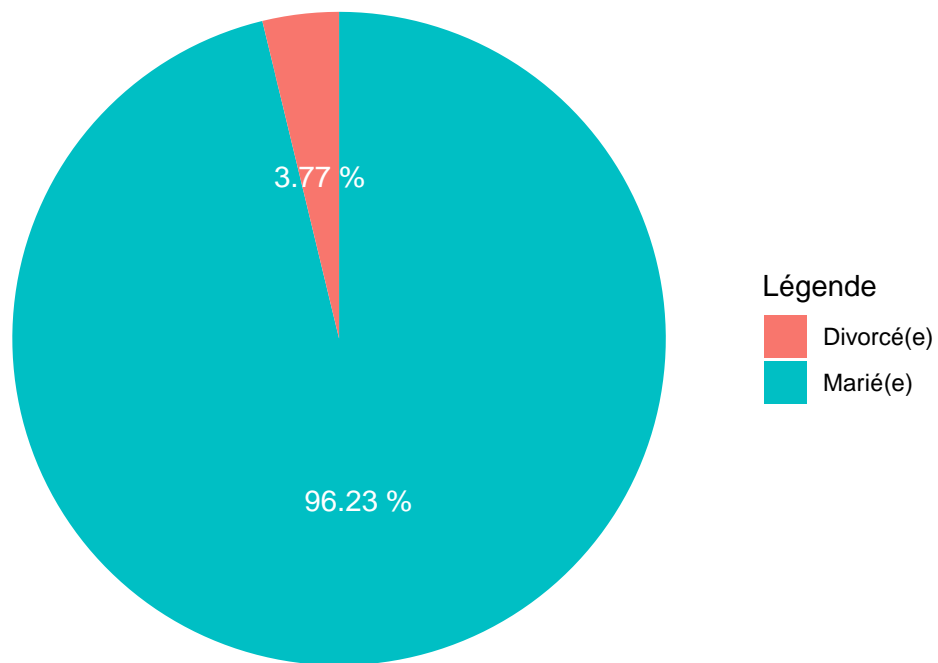
```
## Tableau des statistiques univariées pour la variable sit_mat :
```

```
##
```

```
##
```

##		Nombre		Pourcentage.Var1		Pourcentage.Freq		Variable	
##		-----:		-----:		-----:		-----:	
##		51		Marié(e)		96.226415		Marié(e)	
##		2		Divorcé(e)		3.773585		Divorcé(e)	

Graphique circulaire de sit\_mat



```
univariate_stats(base_tp2, "si_chef_men")
```

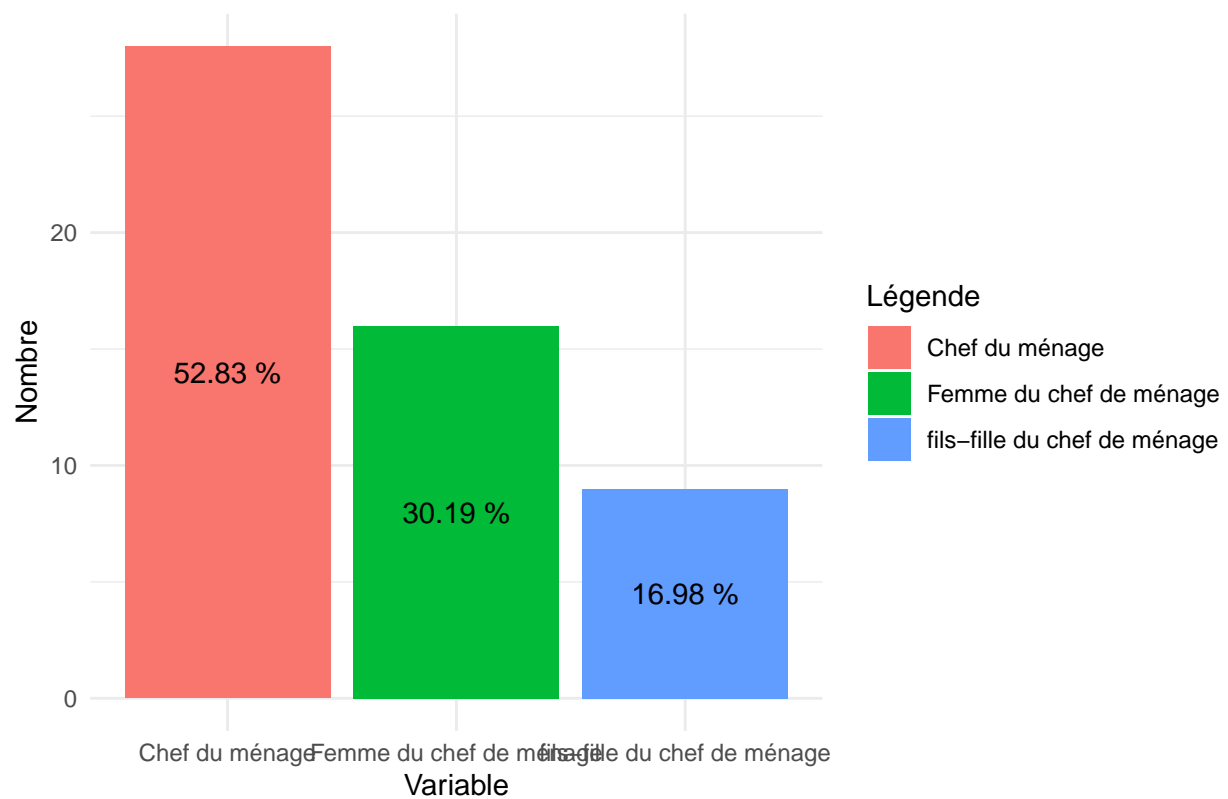
```
## Tableau des statistiques univariées pour la variable si_chef_men :
```

```
##
```

```
##
```

##	Nombre	Pourcentage.Var1	Pourcentage.Freq	Variable
##	16	Femme du chef de ménage	30.18868	Femme du chef de ménage
##	28	Chef du ménage	52.83019	Chef du ménage
##	9	fils-fille du chef de ménage	16.98113	fils-fille du chef de ménage

Graphique en barres de si\_chef\_men



```
univariate_stats(base_tp2, "ethnie")
```

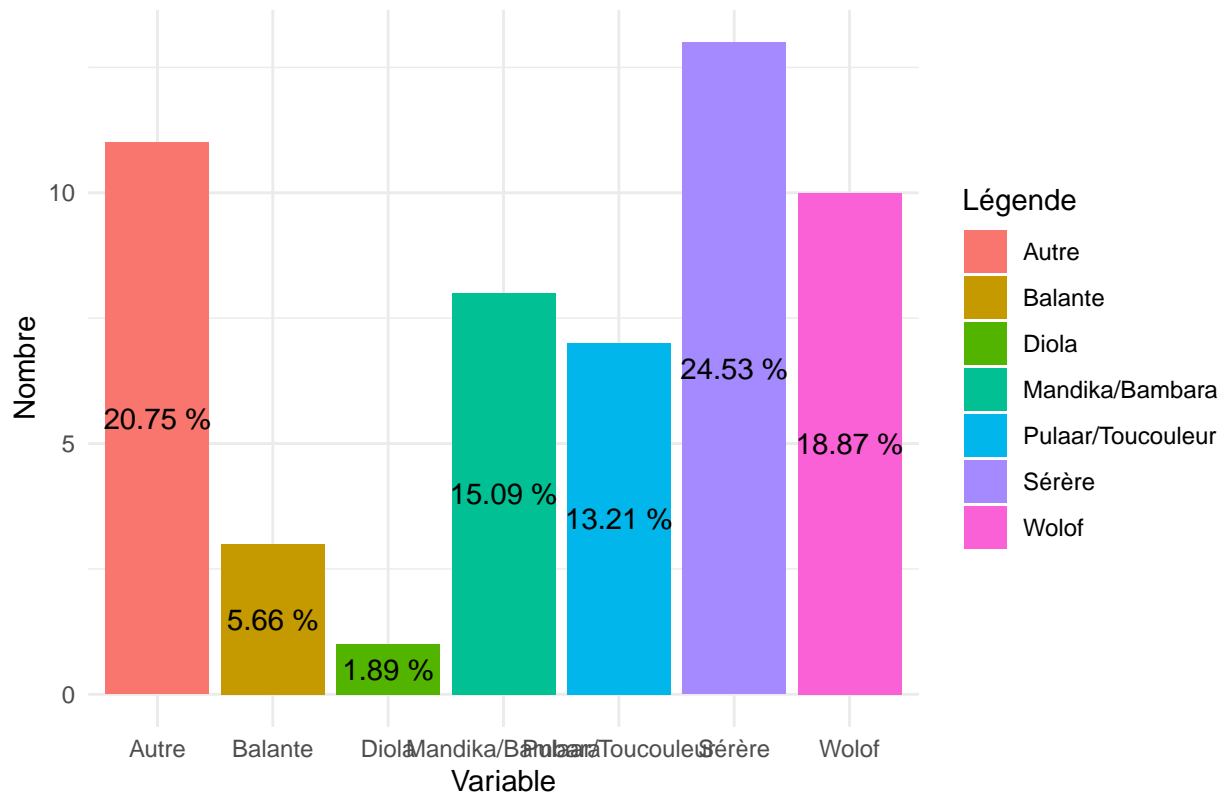
```
## Tableau des statistiques univariées pour la variable ethnie :
```

```
##
```

```
##
```

##	Nombre	Pourcentage.Var1	Pourcentage.Freq	Variable
##	10	52.83	18.867925	Wolof
##	7	30.19	13.207547	Pulaar/Toucouleur
##	13	68.17	24.528302	Sérère
##	8	41.90	15.094340	Mandika/Bambara
##	1	5.28	1.886793	Diola
##	3	15.87	5.660377	Balante
##	11	58.11	20.754717	Autre

Graphique en barres de ethnie



```
univariate_stats(base_tp2,"occupation")
```

```
## Tableau des statistiques univariées pour la variable occupation :
```

```
##
```

```
##
```

```
## | Nombre|Pourcentage.Var1 | Pourcentage.Freq|Variable
```

```
## |-----:|:-----:|-----:|:-----:|
```

```
## | 49|Agriculture, Elevage,Sylviculture,Pêche | 100|Agriculture, Elevage,Sylviculture,Pêche
```

```
## La variable n'a pas exactement deux modalités, le graphique circulaire n'est pas généré.
```

```
univariate_stats(base_tp2,"formation")
```

```
## Tableau des statistiques univariées pour la variable formation :
```

```
##
```

```
##
```

```
## | Nombre|Pourcentage.Var1 | Pourcentage.Freq|Variable |
```

```
## |-----:|:-----:|-----:|:-----:|
```

```
## | 30|Non scolarisé | 56.603774|Non scolarisé |
```

```
## | 10|Elémentaire | 18.867925|Elémentaire |
```

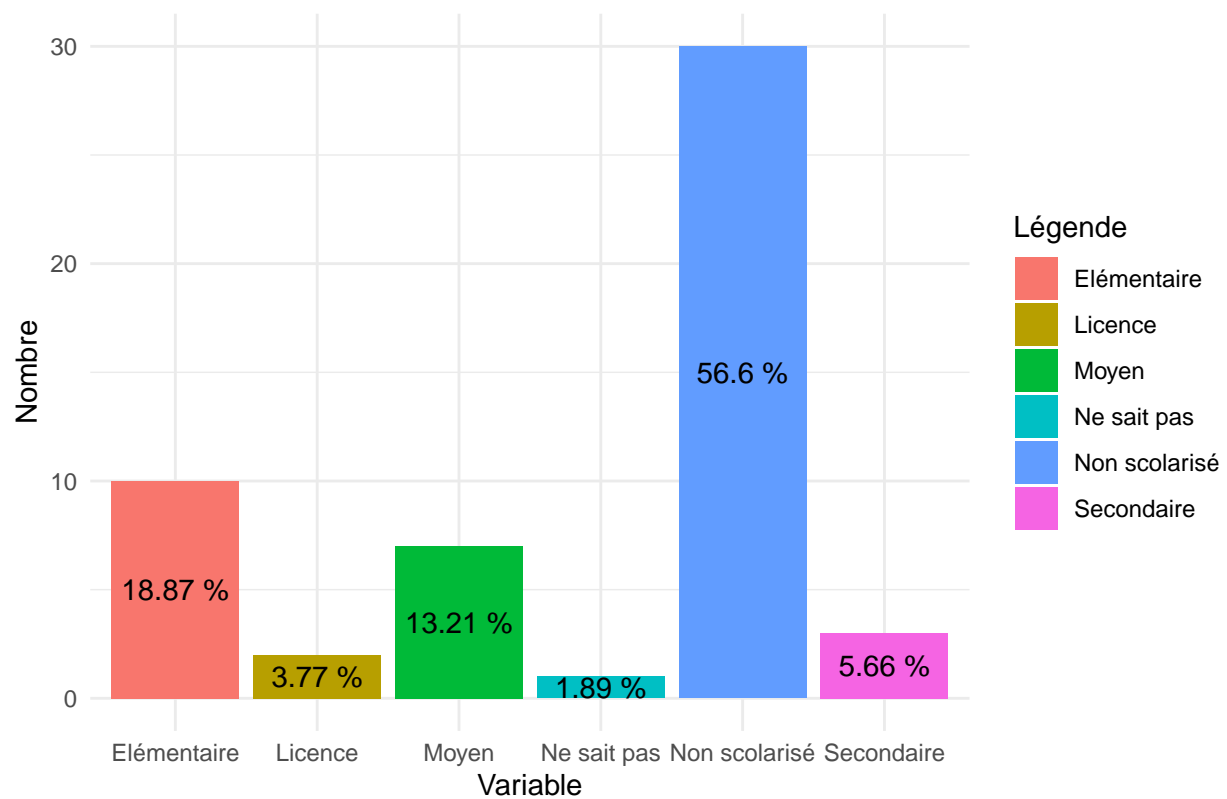
```
## | 7|Moyen | 13.207547|Moyen |
```

```
## | 3|Secondaire | 5.660377|Secondaire |
```

```
## | 2|Licence | 3.773585|Licence |
```

```
## | 1|Ne sait pas | 1.886793|Ne sait pas |
```

Graphique en barres de formation



```
univariate_stats(base_tp2, "niveau_alphabs")
```

```
## Tableau des statistiques univariées pour la variable niveau_alphabs :
```

```
##
```

```
##
```

```
## | Nombre|Pourcentage.Var1 | Pourcentage.Freq|Variable
```

```
## |-----:|:-----:|-----:|:-----:
```

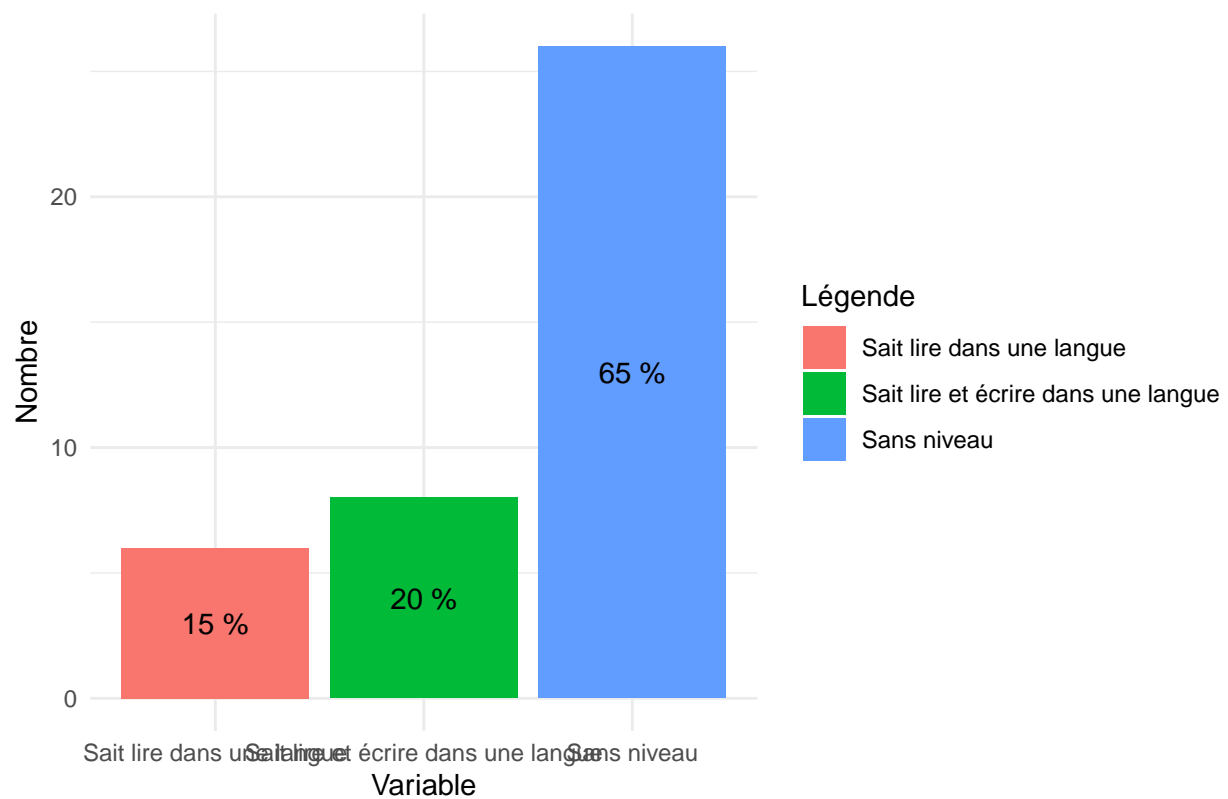
```
## |      26|Sans niveau | 65|Sans niveau
```

```
## |       6|Sait lire dans une langue | 15|Sait lire dans une langue
```

```
## |       8|Sait lire et écrire dans une langue | 20|Sait lire et écrire dans une langue
```



Graphique en barres de niveau\_alphabs



```
univariate_stats(base_tp2, "types_varietes_1")
```

```
## Tableau des statistiques univariées pour la variable types_varietes_1 :
```

```
##
```

```
##
```

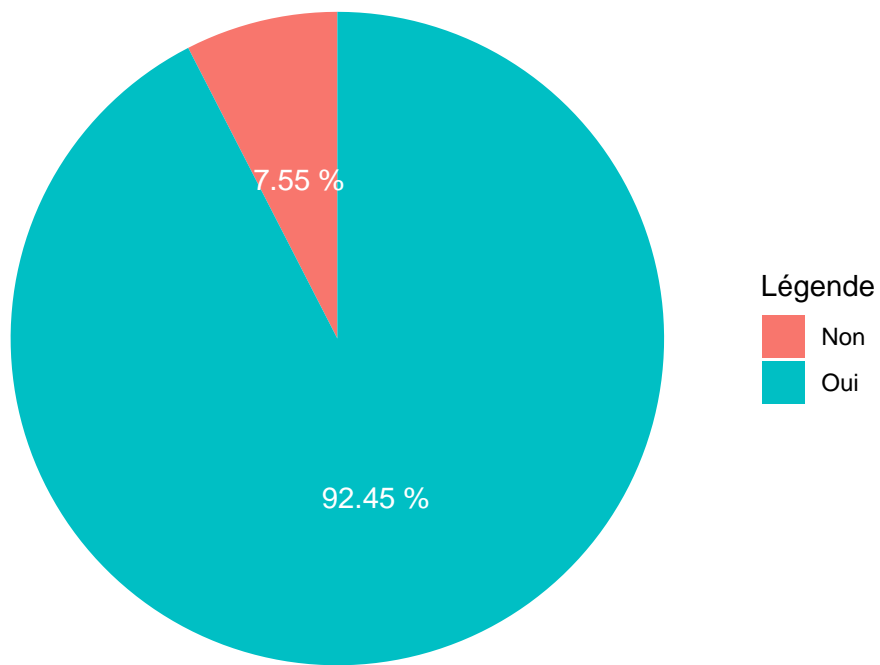
```
## | Nombre|Pourcentage.Var1 | Pourcentage.Freq|Variable |
```

```
## |-----:|:-----:|:-----:|:-----:|
```

```
## |      4|Non          |      7.54717|Non      |
```

```
## |     49|Oui          |     92.45283|Oui      |
```

Graphique circulaire de types\_varietes\_1



```
univariate_stats(base_tp2, "types_varietes_2")
```

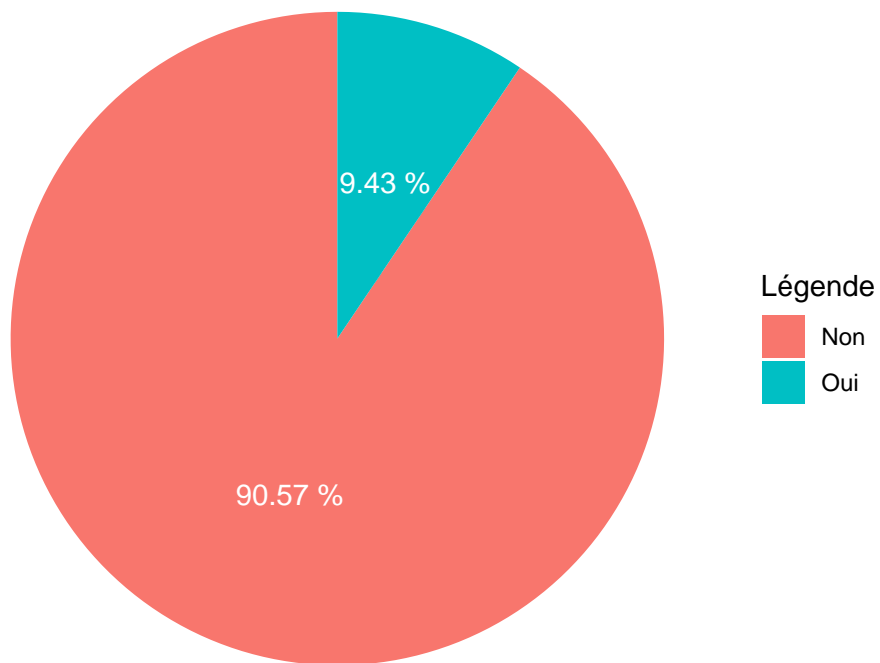
```
## Tableau des statistiques univariées pour la variable types_varietes_2 :
```

```
##
```

```
##
```

```
## | Nombre|Pourcentage.Var1 | Pourcentage.Freq|Variable |
## |-----:|:-----|-----:|:-----|
## |      48|Non          |      90.566038|Non      |
## |       5|Oui           |       9.433962|Oui      |
```

Graphique circulaire de types\_varietes\_2



```
univariate_stats(base_tp2, "criteres_var_1")
```

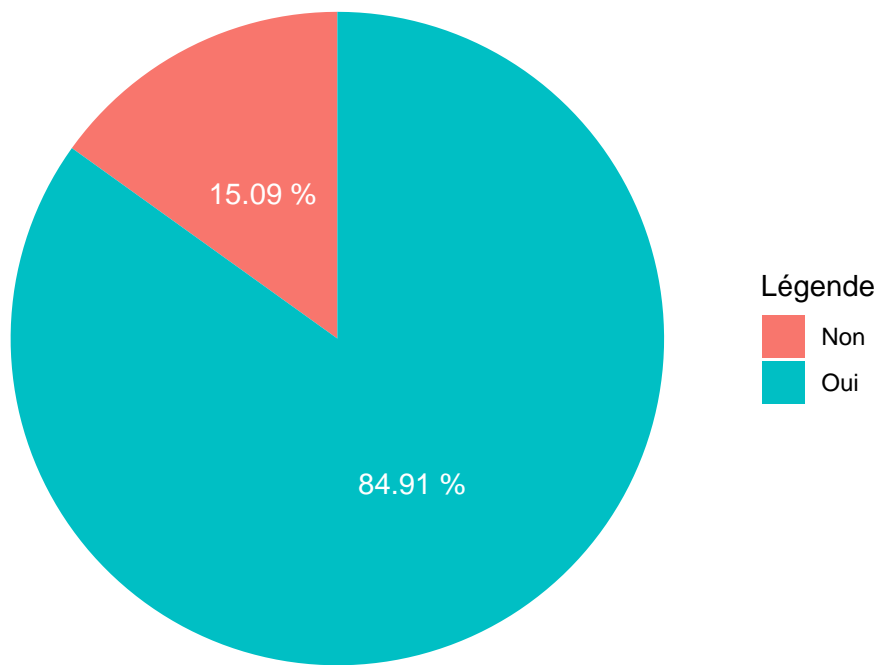
```
## Tableau des statistiques univariées pour la variable criteres_var_1 :
```

```
##
```

```
##
```

```
## | Nombre|Pourcentage.Var1 | Pourcentage.Freq|Variable |
## |-----:|:-----|-----:|:-----|
## |      8|Non          |      15.09434|Non      |
## |     45|Oui          |      84.90566|Oui      |
```

Graphique circulaire de criteres\_var\_1



```
univariate_stats(base_tp2,"criteres_var_2" )
```

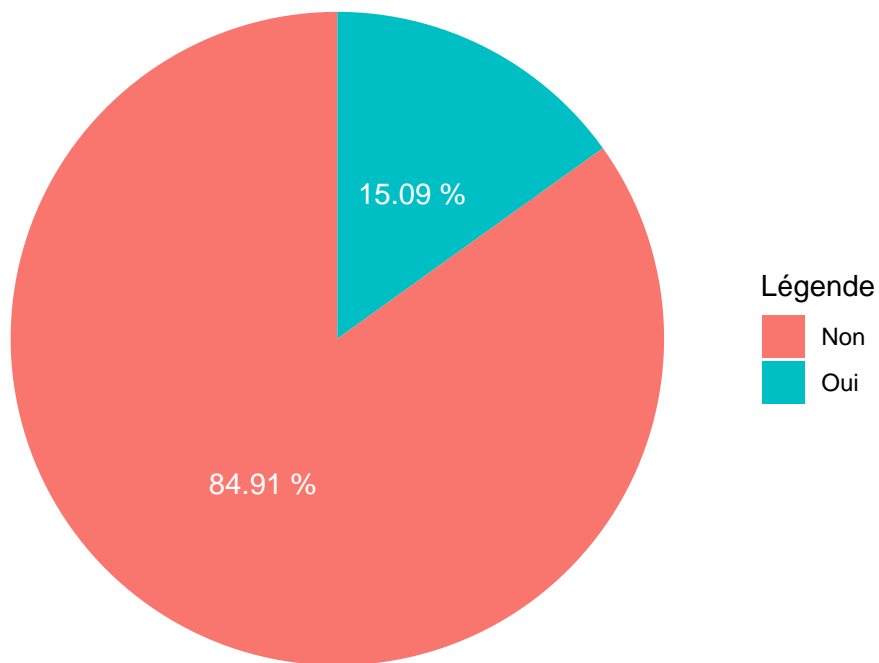
```
## Tableau des statistiques univariées pour la variable criteres_var_2 :
```

```
##
```

```
##
```

##	Nombre	Pourcentage.Var1	Pourcentage.Freq	Variable
##	45	Non	84.90566	Non
##	8	Oui	15.09434	Oui

## Graphique circulaire de criteres\_var\_2



## 4.2 Analyses descriptives bivariées

### 4.2.1 Creation de fonction pour les graphiques

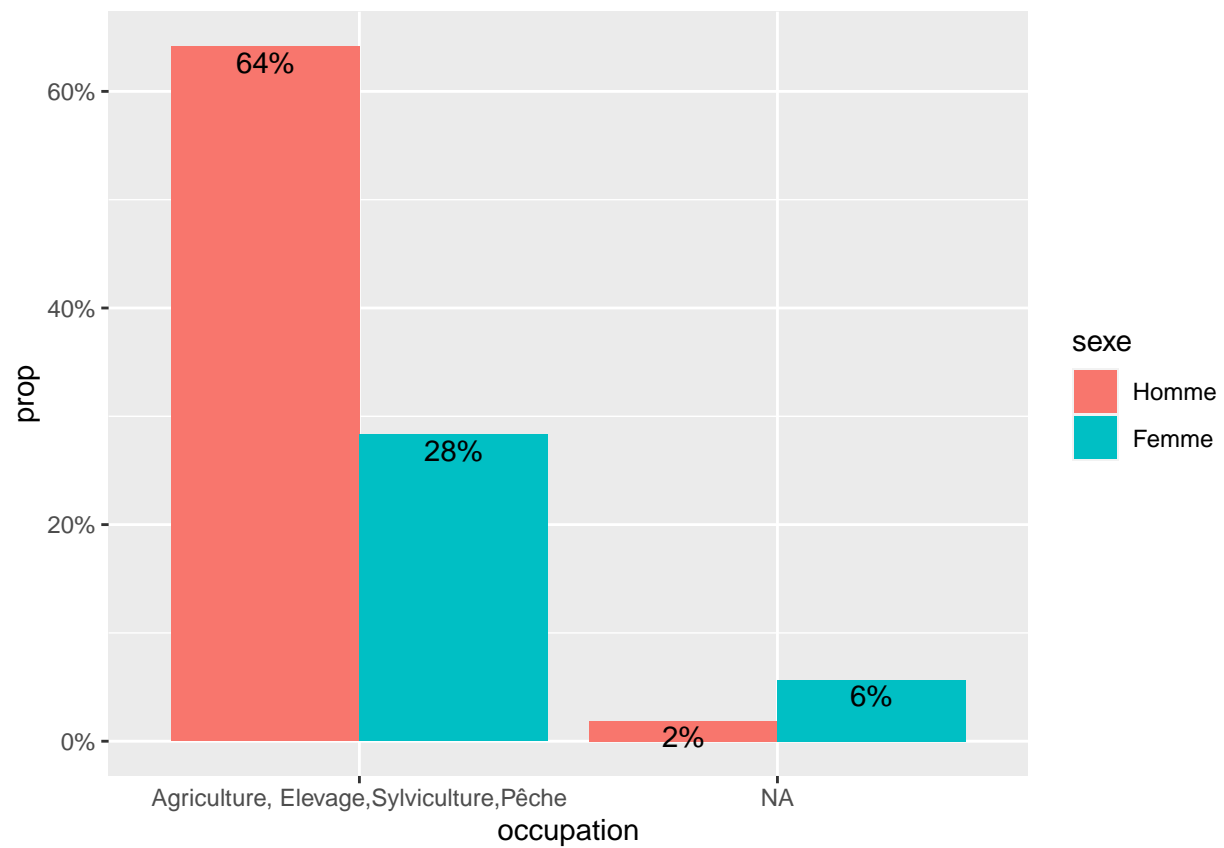
```
graph_bivar <- function(data, var1, var2){  
  
  # Calculer les proportions pour chaque niveau de var1 et var2  
  prop_table <- data %>%  
    group_by({{ var1 }}, {{ var2 }}) %>%  
    summarize(prop = n() / nrow(data))  
  
  # Créer le graphique  
  p <- ggplot(prop_table, aes(x = {{var1}} , fill = {{var2}} , y = prop, label = scales::percent(prop)  
    geom_bar(stat = "identity", position = "dodge") +  
    geom_text(aes(y = prop - 0.005), position = position_dodge(width = 0.9), vjust = "top") +  
    scale_y_continuous(labels = scales::percent)  
  
  return(p)  
}
```

```
graph_bivar(base_tp2, occupation , sexe)
```

#### 4.2.1.1 Application à certaines variables

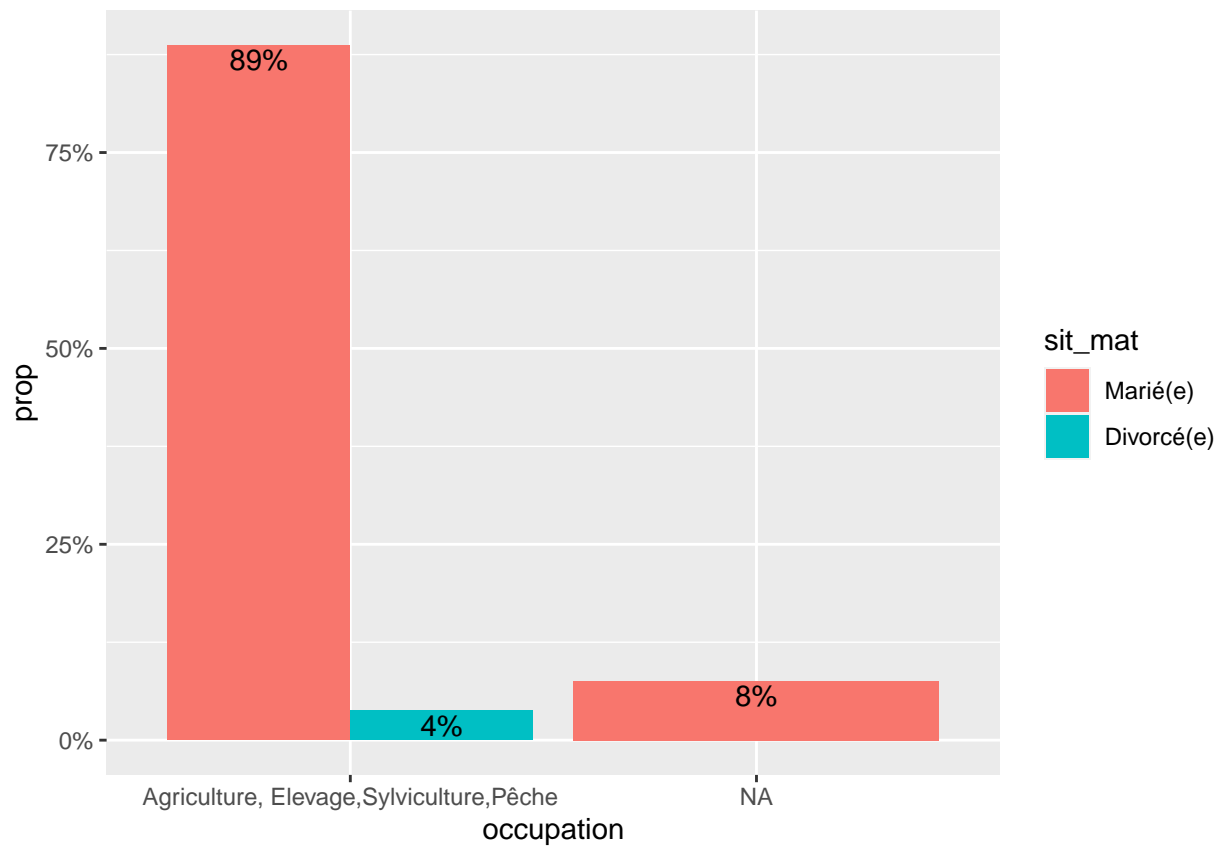
## `summarise()` has grouped output by 'occupation'. You can override using the

```
## `.groups` argument.
```



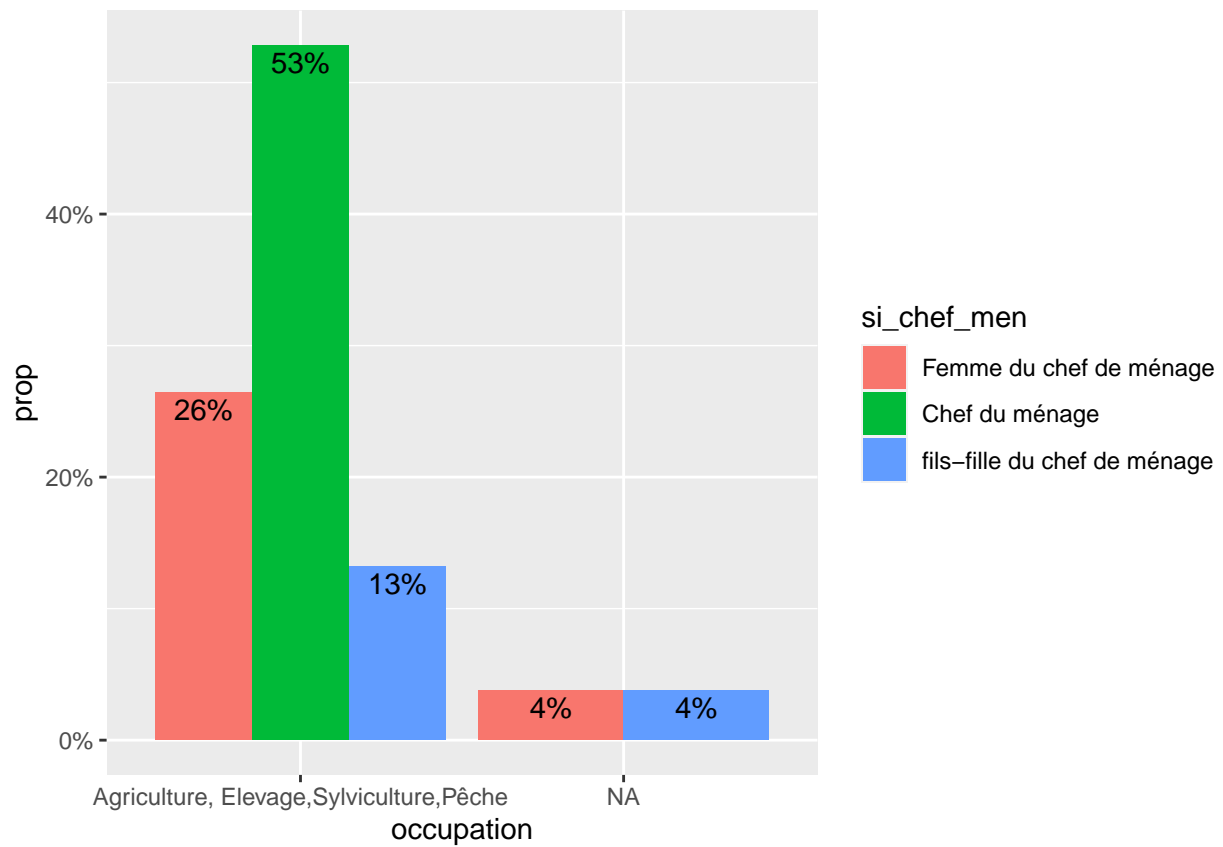
```
graph_bivar(base_tp2, occupation , sit_mat)
```

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



```
graph_bivar(base_tp2, occupation , si_chef_men)
```

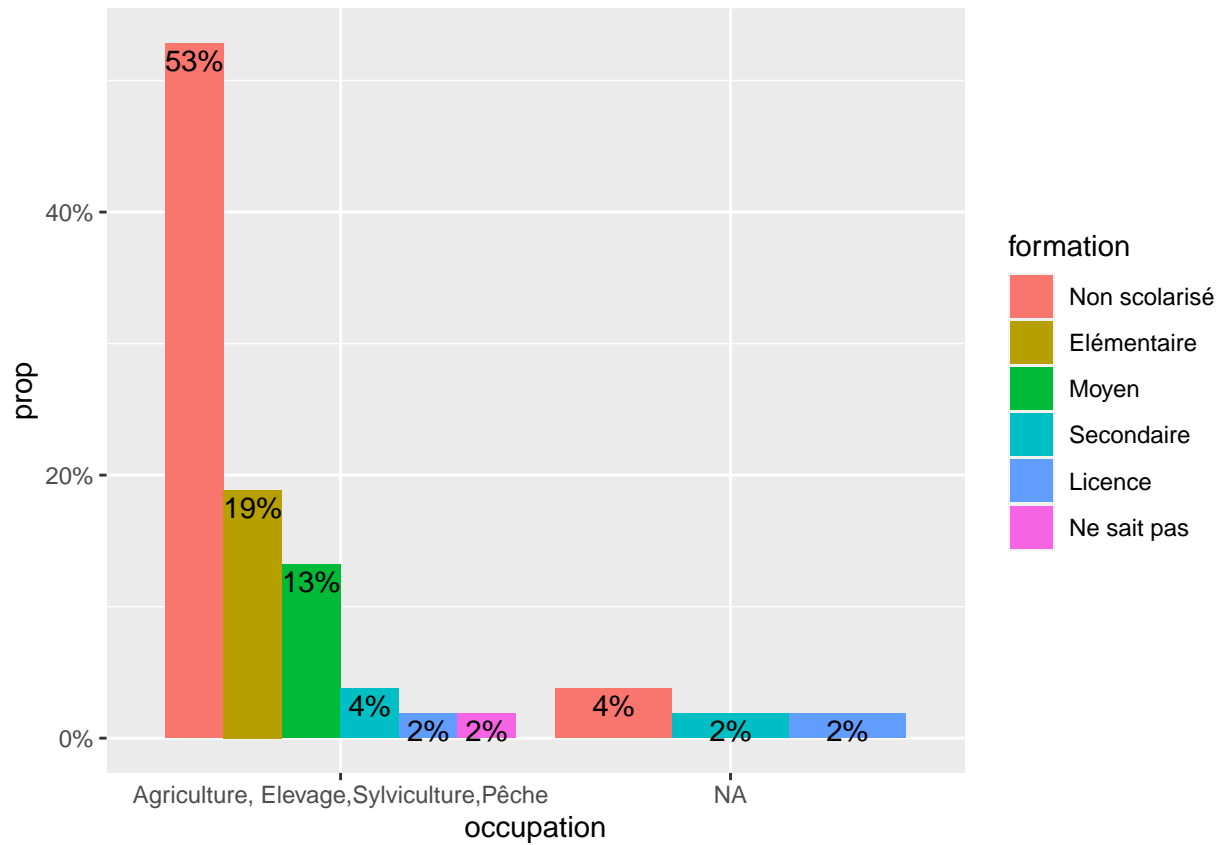
```
## `summarise()` has grouped output by 'occupation'. You can override using the  
## `.groups` argument.
```



```
graph_bivar(base_tp2, occupation , formation)
```

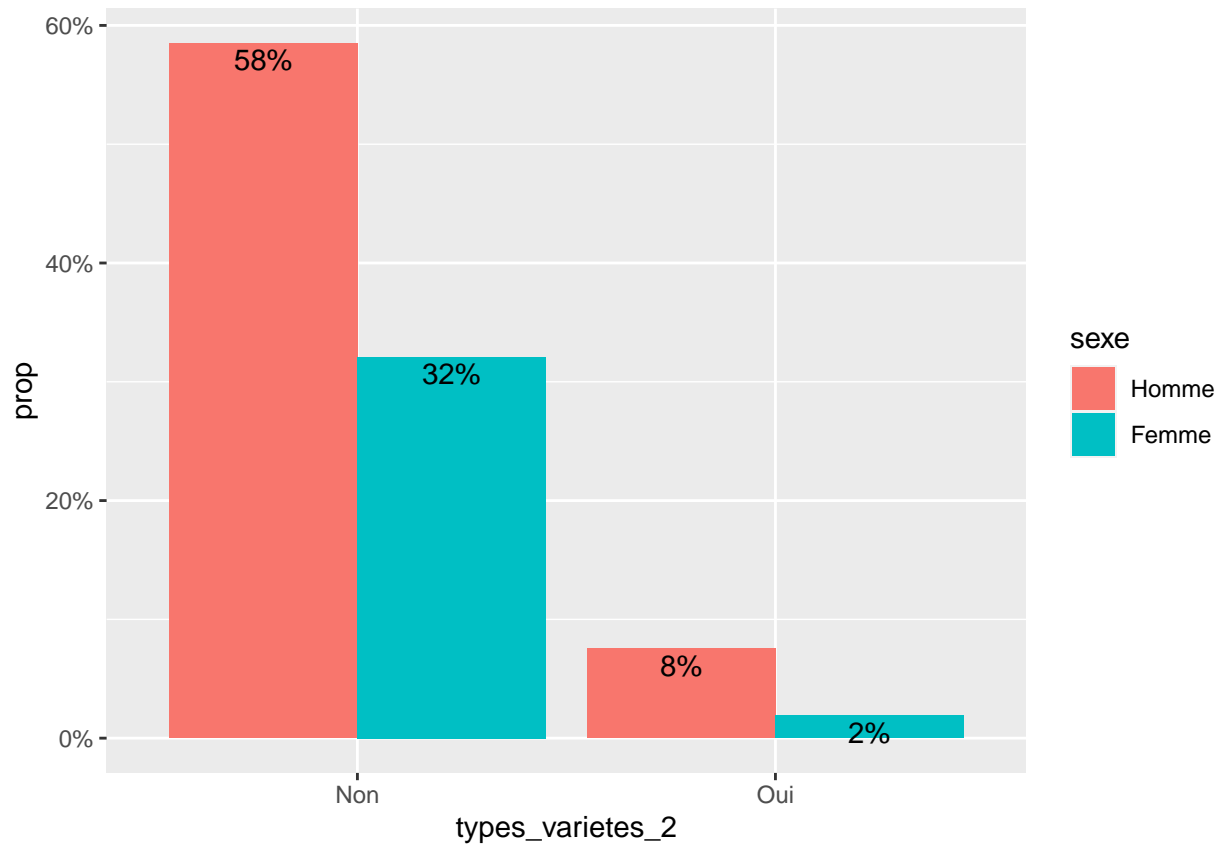
```
## `summarise()` has grouped output by 'occupation'. You can override using the
## `.groups` argument.
```





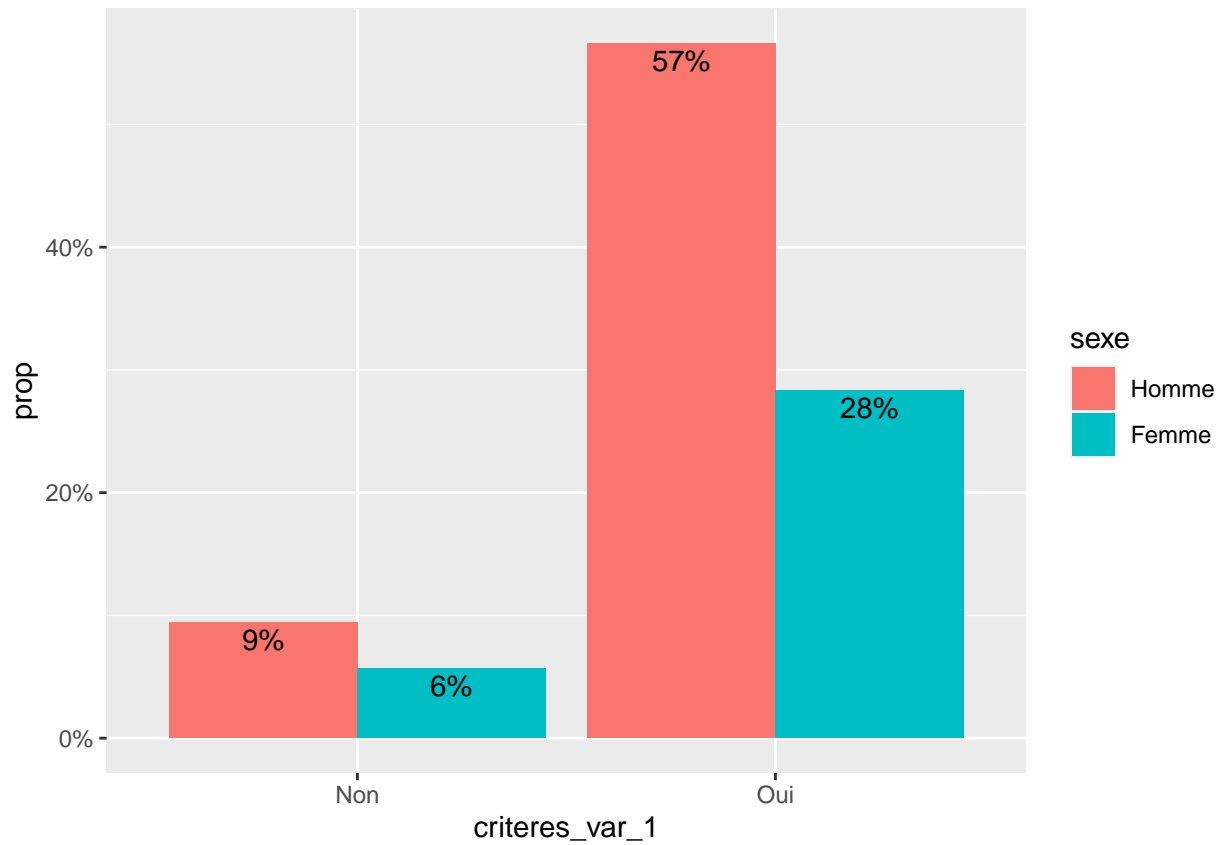
```
graph_bivar(base_tp2, types_varietes_2 , sexe)
```

```
## `summarise()` has grouped output by 'types_varietes_2'. You can override using
## the `.groups` argument.
```



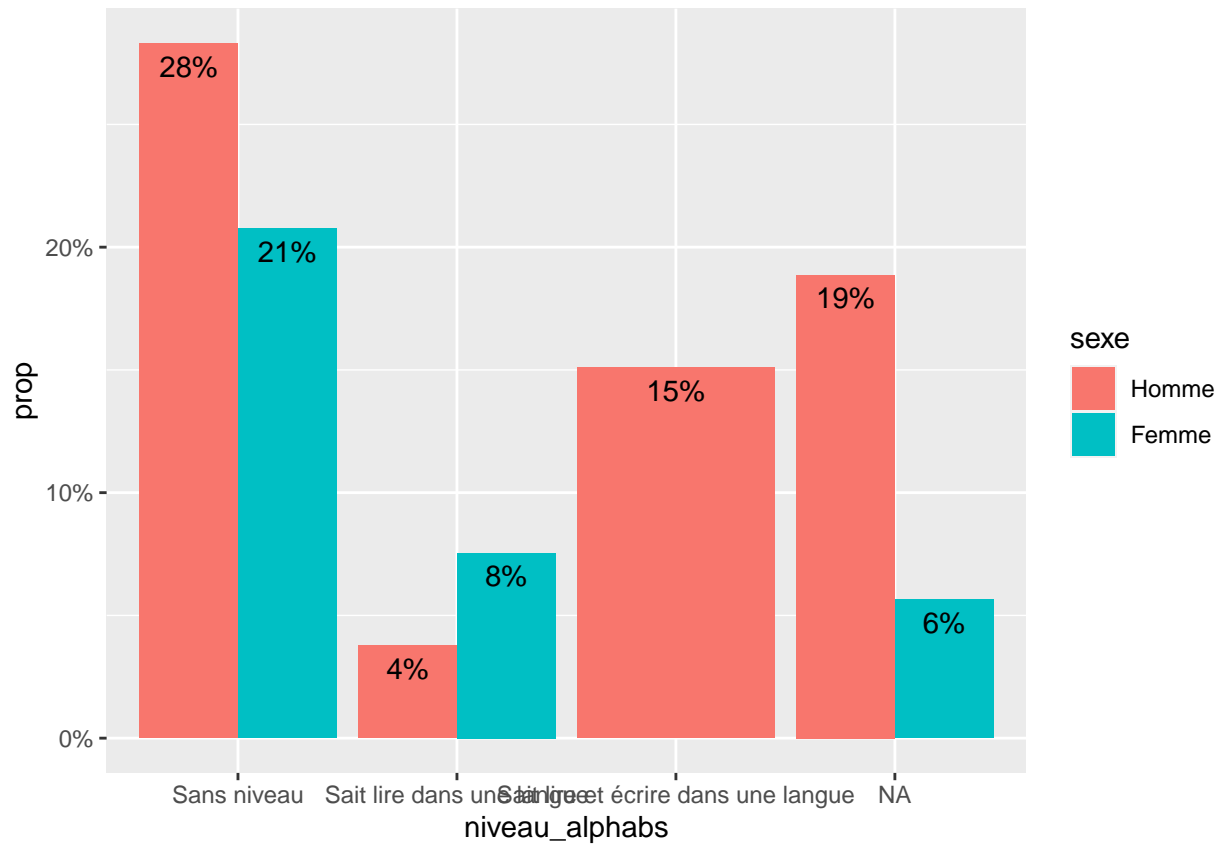
```
graph_bivar(base_tp2,criteres_var_1 , sexe)
```

```
## `summarise()` has grouped output by 'criteres_var_1'. You can override using  
## the `groups` argument.
```



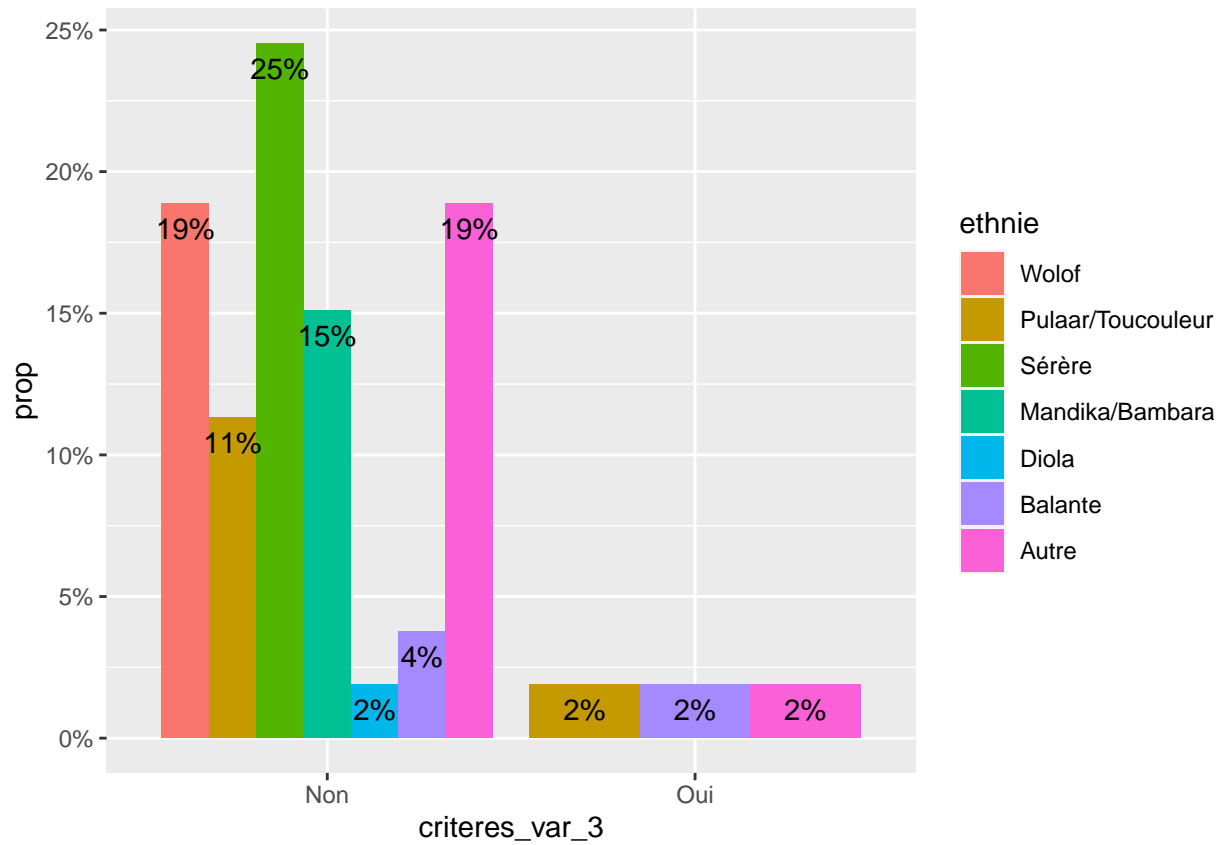
```
graph_bivar(base_tp2,niveau_alphabs , sexe)
```

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



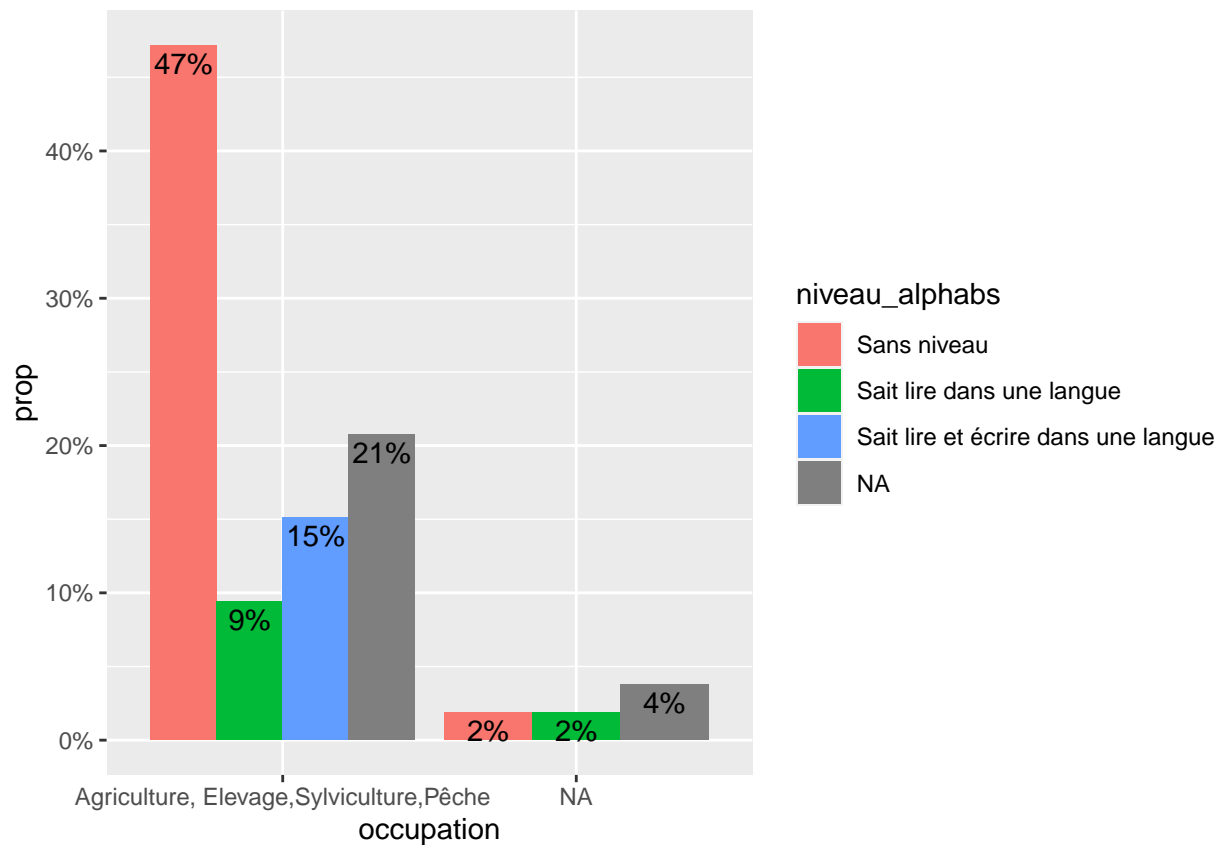
```
graph_bivar(base_tp2,criteres_var_3 , ethnie)
```

```
## `summarise()` has grouped output by 'criteres_var_3'. You can override using
## the ``.groups` argument.
```



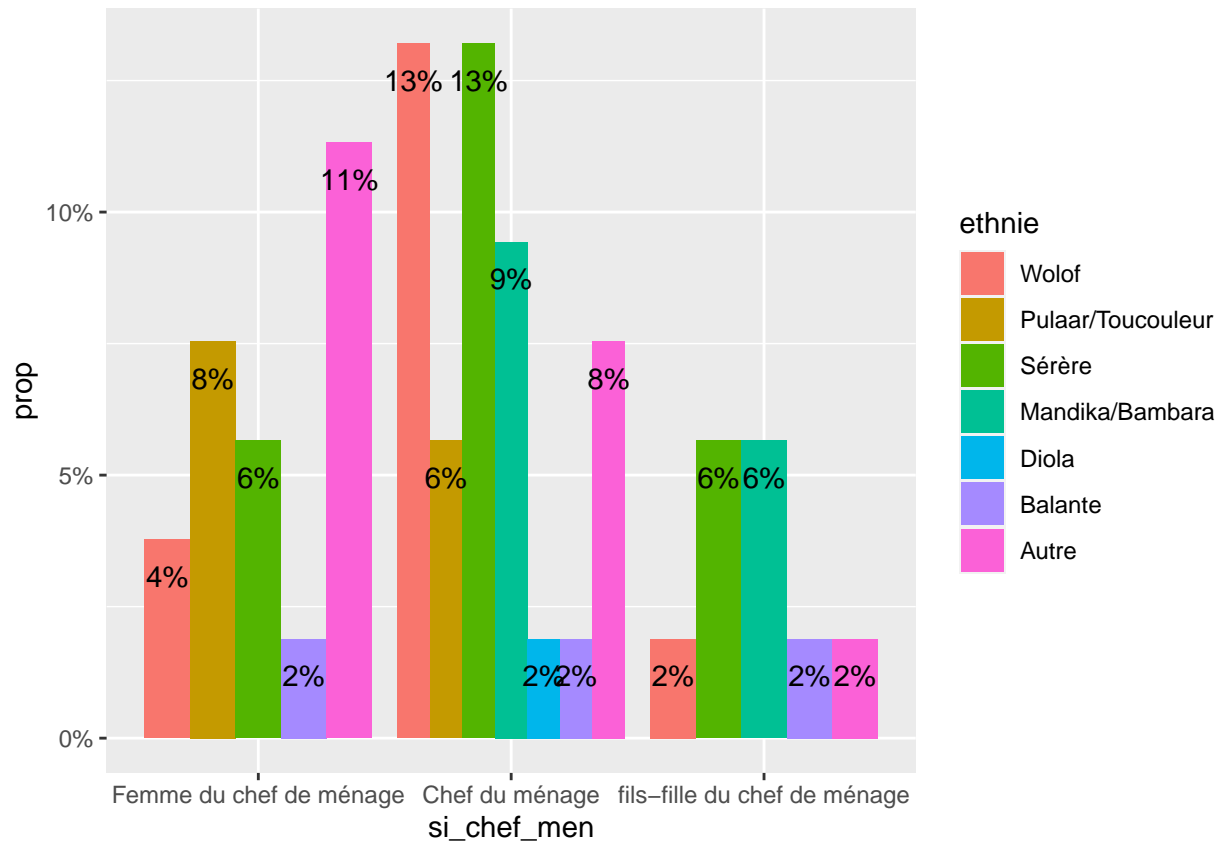
```
graph_bivar(base_tp2, occupation , niveau_alphabs)
```

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



```
graph_bivar(base_tp2, si_chef_men , ethnie)
```

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



## 5 Conclusion

Ce travail nous a permis d'acquérir plus d'outils pour le recodage et la labelisation des variables ,un plus pour les futurs ingenieurs que nous sommes.