Code-Annexe Stats Assus

2025-04-29

```r
knitr::opts_chunk$set(echo = TRUE)
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```r
library(knitr)
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```r
library(lmtest)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
library(statmod)
library(glmnet)
```

```
## Loading required package: Matrix

## Loaded glmnet 4.1-8
```

```r
library(MASS)
library(stats)
library(mgcv)
```

```
## Loading required package: nlme

##
## Attaching package: 'nlme'

## The following object is masked from 'package:dplyr':
##
##     collapse

## This is mgcv 1.9-1. For overview type 'help("mgcv-package")'.
```

```r
library(pscl)
```

```
## Warning: package 'pscl' was built under R version 4.3.2
```

```
## Classes and Methods for R originally developed in the
## Political Science Computational Laboratory
## Department of Political Science
## Stanford University (2002-2015),
## by and under the direction of Simon Jackman.
## hurdle and zeroinfl functions by Achim Zeileis.
```

```r
data = source("/Users/markus/Downloads/assurance_complet.R")
```

1. Analyse descriptive

1.1 État du DataFrame

```r
head(dat)
```

```
##                                    pcs       RUC          cs reves
## 1                            Retraites  7500.000 Moyenne Sup  7500
## 2 Cadres et prof. intellectuelles sup. 25000.000        Aise 60000
## 3 Cadres et prof. intellectuelles sup. 25000.000        Aise 90000
## 4                            Retraites  5500.000 Moyenne Inf  5500
## 5                             Ouvriers  5113.636 Moyenne Inf 11250
## 6                            Retraites  6500.000 Moyenne Inf  6500
##          crevpp region habi                 Ahabi        Atyph agecat
## 1 4eme quartile      1    8 Paris + Agglomeration    Locataire  61-96
## 2 4eme quartile      1    8 Paris + Agglomeration Proprietaire  51-60
## 3 4eme quartile      1    8 Paris + Agglomeration Proprietaire  51-60
## 4 3eme quartile      1    8 Paris + Agglomeration    Locataire  61-96
## 5 2eme quartile      1    8 Paris + Agglomeration    Locataire  41-50
## 6 4eme quartile      1    8 Paris + Agglomeration    Locataire  61-96
##                 Acompm nbpers        enfants            Anat            Bauto
## 1        Personne seule      1 Pas d'enfants Menage francais Pas de vehicule
## 2          Autre menage      3 Pas d'enfants Menage francais Au - 1 vehicule
## 3          Autre menage      3 Pas d'enfants Menage francais Au - 1 vehicule
## 4        Personne seule      1 Pas d'enfants     Non declare Pas de vehicule
## 5 Couple avec enfant(s)      3 Au - 1 enfant Menage francais Au - 1 vehicule
## 6        Personne seule      1 Pas d'enfants Menage francais Pas de vehicule
##   Nbadulte Sinistre1 Sinistre2 Sinistre3 Police1 Police2 Police3 durPolice1
## 1        1       0.6       0.0      0.00    6.31   5.925   0.695  9.0559139
## 2        3       0.0       0.0      2.71    0.00   0.690   0.785  0.0000000
## 3        3       0.0       0.0      2.71    0.00   0.690   0.785  0.0000000
## 4        1       0.0       0.0      3.45    1.32   5.445   0.000  5.6901285
## 5        2       0.4       2.0      0.00    0.40   8.430   1.470  0.6378342
## 6        1       0.0       1.1      0.00    0.00   2.500   2.850  0.0000000
##   Duree NSin censure Sinistre0
## 1   906    2       1  17.58972
## 2    30    3       1  20.92726
## 3    30    4       1  22.02196
## 4   569    2       1  18.01244
## 5    64    5       1  12.11505
## 6    30    2       1  19.60981
```

```r
str(dat)
```

```
## 'data.frame':    5352 obs. of  27 variables:
##  $ pcs       : Factor w/ 8 levels "Agr. exploitants",..: 8 4 4 8 6 8 4 6 3 8 ...
##  $ RUC       : num  7500 25000 25000 5500 5114 ...
##  $ cs        : Factor w/ 4 levels "Aise","Modeste",..: 4 1 1 3 3 3 1 4 1 3 ...
##  $ reves     : num  7500 60000 90000 5500 11250 ...
##  $ crevpp    : Factor w/ 4 levels "1er quartile",..: 4 4 4 3 2 4 4 3 4 3 ...
##  $ region    : chr  "1" "1" "1" "1" ...
##  $ habi      : chr  "8" "8" "8" "8" ...
##  $ Ahabi     : Factor w/ 5 levels "Communes rurales",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ Atyph     : Factor w/ 3 levels "Locataire","Non declare",..: 1 3 3 1 1 1 3 1 1 1 ...
##  $ agecat    : Factor w/ 4 levels "21-40","41-50",..: 4 3 3 4 2 4 4 1 4 4 ...
##  $ Acompm    : Factor w/ 4 levels "Autre menage",..: 4 1 1 4 2 4 4 2 4 1 ...
##  $ nbpers    : num  1 3 3 1 3 1 1 3 1 3 ...
##  $ enfants   : Factor w/ 2 levels "Au - 1 enfant",..: 2 2 2 2 1 2 2 1 2 2 ...
##  $ Anat      : Factor w/ 3 levels "Au moins une personne etrangere",..: 2 2 2 3 2 2 2 2 2 1 ...
##  $ Bauto     : Factor w/ 2 levels "Au - 1 vehicule",..: 2 1 1 2 1 2 2 1 2 1 ...
##  $ Nbadulte  : num  1 3 3 1 2 1 1 2 1 3 ...
##  $ Sinistre1 : num  0.6 0 0 0 0.4 0 0.4 0 1.6 0 ...
##  $ Sinistre2 : num  0 0 0 0 2 1.1 0 0 0 0 ...
##  $ Sinistre3 : num  0 2.71 2.71 3.45 0 0 1.1 0.71 0 0.92 ...
##  $ Police1   : num  6.31 0 0 1.32 0.4 ...
##  $ Police2   : num  5.92 0.69 0.69 5.45 8.43 ...
##  $ Police3   : num  0.695 0.785 0.785 0 1.47 ...
##  $ durPolice1: num  9.056 0 0 5.69 0.638 ...
##  $ Duree     : num  906 30 30 569 64 ...
##  $ NSin      : num  2 3 4 2 5 2 4 4 2 4 ...
##  $ censure   : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Sinistre0 : num  17.6 20.9 22 18 12.1 ...
```

```r
# 1. Statistiques descriptives générales
summary(dat)
```

```
##                                   pcs            RUC
##  Retraites                          :1372   Min.   :  277.8
##  Ouvriers                           :1327   1st Qu.: 3823.5
##  Professions intermediaires         : 887   Median : 5500.0
##  Employes                           : 806   Mean   : 6277.5
##  Cadres et prof. intellectuelles sup.: 480   3rd Qu.: 7812.5
##  Autres pers. sans activite prof.    : 183   Max.   :35294.1
##  (Other)                            : 297
##            cs            reves                  crevpp        region
##  Aise        : 629   Min.   :    1000   1er quartile :1260   Length:5352
##  Modeste     : 901   1st Qu.:    8500   2eme quartile:1194   Class :character
##  Moyenne Inf:2346   Median :   11250   3eme quartile:1403   Mode  :character
##  Moyenne Sup:1476   Mean   :   14880   4eme quartile:1495
##                     3rd Qu.:   16250
##                     Max.   :3416250
##
##      habi                                    Ahabi                 Atyph
##  Length:5352        Communes rurales               :1352   Locataire    :1963
##  Class :character   Paris + Agglomeration          : 697   Non declare :   73
```

```
##  Mode  :character   Un. urb. de 10 000 a 99 999 hab.:1031   Proprietaire:3316
##                     Un. urb. de 100 000 hab. et +   :1623
##                     Un. urb. de 2 000 a 9 999 hab.  : 649
##
##
##    agecat                       Acompm         nbpers
##  21-40:1616   Autre menage        :1921   Min.   : 1.000
##  41-50:1483   Couple avec enfant(s):1379   1st Qu.: 2.000
##  51-60: 854   Couple sans enfant   :1293   Median : 3.000
##  61-96:1399   Personne seule       : 759   Mean   : 3.038
##                                            3rd Qu.: 4.000
##                                            Max.   :10.000
##
##          enfants                              Anat
##  Au - 1 enfant:1379   Au moins une personne etrangere: 115
##  Pas d'enfants:3973   Menage francais                :4866
##                       Non declare                    : 371
##
##
##
##
##              Bauto         Nbadulte       Sinistre1         Sinistre2
##  Au - 1 vehicule:4909   Min.   :1.000   Min.   :  0.000   Min.   : 0.0000
##  Pas de vehicule: 443   1st Qu.:2.000   1st Qu.:  0.000   1st Qu.: 0.0000
##                         Median :2.000   Median :  0.000   Median : 0.0000
##                         Mean   :2.388   Mean   :  1.243   Mean   : 0.1615
##                         3rd Qu.:3.000   3rd Qu.:  0.000   3rd Qu.: 0.0000
##                         Max.   :8.000   Max.   :355.000   Max.   :31.1000
##
##    Sinistre3         Police1           Police2           Police3
##  Min.   : 0.000   Min.   : 0.0000   Min.   :  0.000   Min.   : 0.000
##  1st Qu.: 0.000   1st Qu.: 0.5375   1st Qu.:  3.829   1st Qu.: 0.520
##  Median : 0.705   Median : 1.9500   Median :  9.060   Median : 1.420
##  Mean   : 1.837   Mean   : 3.7507   Mean   : 13.017   Mean   : 2.110
##  3rd Qu.: 2.560   3rd Qu.: 5.0170   3rd Qu.: 17.950   3rd Qu.: 2.833
##  Max.   :40.220   Max.   :54.9850   Max.   :124.109   Max.   :34.743
##
##    durPolice1            Duree             NSin            censure
##  Min.   :0.000e+00   Min.   :   0.0   Min.   : 0.000   Min.   :0.0000
##  1st Qu.:0.000e+00   1st Qu.:  23.0   1st Qu.: 0.000   1st Qu.:0.0000
##  Median :0.000e+00   Median :  42.5   Median : 4.000   Median :1.0000
##  Mean   :5.191e+08   Mean   : 249.1   Mean   : 4.249   Mean   :0.6179
##  3rd Qu.:2.000e+00   3rd Qu.: 233.0   3rd Qu.: 6.000   3rd Qu.:1.0000
##  Max.   :2.778e+12   Max.   :9877.0   Max.   :30.000   Max.   :1.0000
##
##    Sinistre0
##  Min.   : 0.9652
##  1st Qu.:13.2769
##  Median :17.2558
##  Mean   :16.1732
##  3rd Qu.:19.2718
##  Max.   :27.3074
##
```

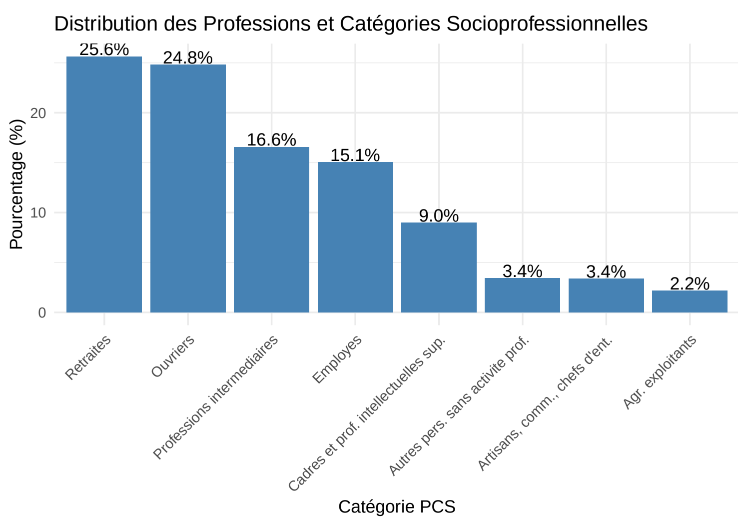
```r
sum(is.na(dat))
```

```
## [1] 0
```

1.2 Analyse des variables catégorielles clés

a) Professions et Catégories Socioprofessionnelles

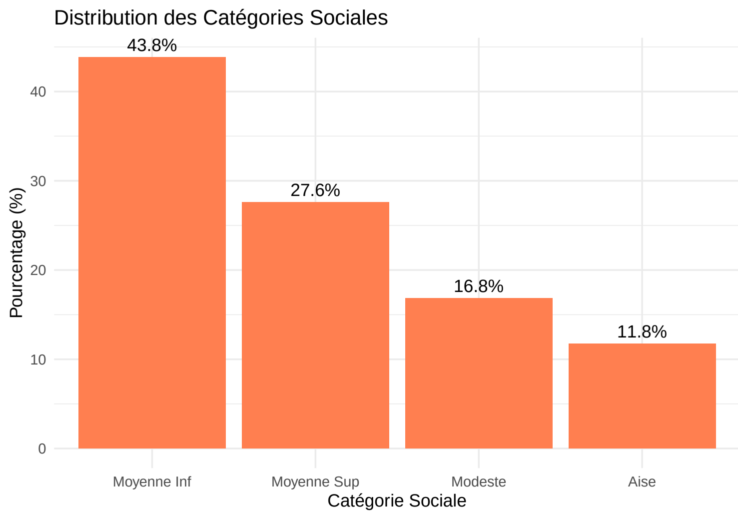
```r
# Analyse de la variable pcs (Professions et Catégories Socioprofessionnelles)
tab_pcs <- table(dat$pcs)
prop_pcs <- prop.table(tab_pcs) * 100
df_pcs <- data.frame(Categorie = names(tab_pcs), Count = as.numeric(tab_pcs),
                     Percentage = as.numeric(prop_pcs))

# Visualisation de pcs
ggplot(df_pcs, aes(x = reorder(Categorie, -Percentage), y = Percentage)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  geom_text(aes(label = sprintf("%.1f%%", Percentage)), vjust = -0.1) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Distribution des Professions et Catégories Socioprofessionnelles",
       x = "Catégorie PCS", y = "Pourcentage (%)")
```



b) Catégories Sociales

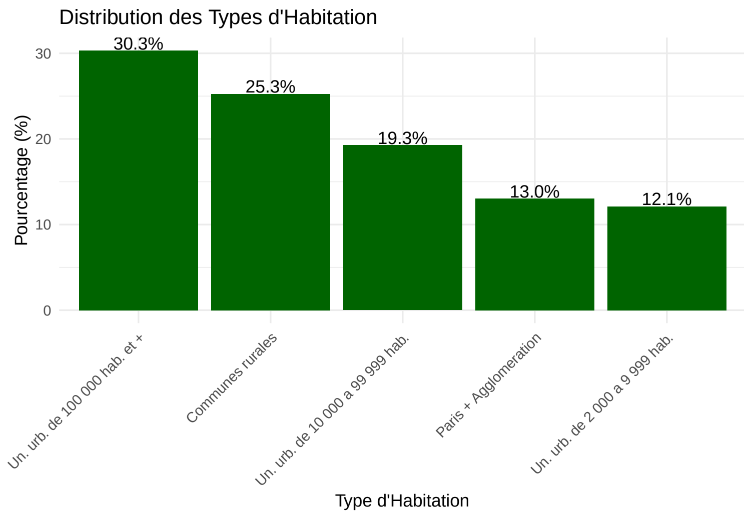
```r
# Analyse de la variable cs (Catégorie Sociale)
tab_cs <- table(dat$cs)
prop_cs <- prop.table(tab_cs) * 100
df_cs <- data.frame(Categorie = names(tab_cs), Count = as.numeric(tab_cs),
                    Percentage = as.numeric(prop_cs))

# Visualisation de cs
ggplot(df_cs, aes(x = reorder(Categorie, -Percentage), y = Percentage)) +
  geom_bar(stat = "identity", fill = "coral") +
```

```r
  geom_text(aes(label = sprintf("%.1f%%", Percentage)), vjust = -0.5) +
  theme_minimal() +
  labs(title = "Distribution des Catégories Sociales",
       x = "Catégorie Sociale", y = "Pourcentage (%)")
```



c) Le Type d'habitation

```r
# Analyse de la variable Ahabi (Type d'habitation)
tab_ahabi <- table(dat$Ahabi)
prop_ahabi <- prop.table(tab_ahabi) * 100
df_ahabi <- data.frame(Categorie = names(tab_ahabi), Count = as.numeric(tab_ahabi),
                       Percentage = as.numeric(prop_ahabi))

# Visualisation de Ahabi
ggplot(df_ahabi, aes(x = reorder(Categorie, -Percentage), y = Percentage)) +
  geom_bar(stat = "identity", fill = "darkgreen") +
  geom_text(aes(label = sprintf("%.1f%%", Percentage)), vjust = -0.1) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Distribution des Types d'Habitation",
       x = "Type d'Habitation", y = "Pourcentage (%)")
```



1.3 Analyse des variables numériques

a) Revenus

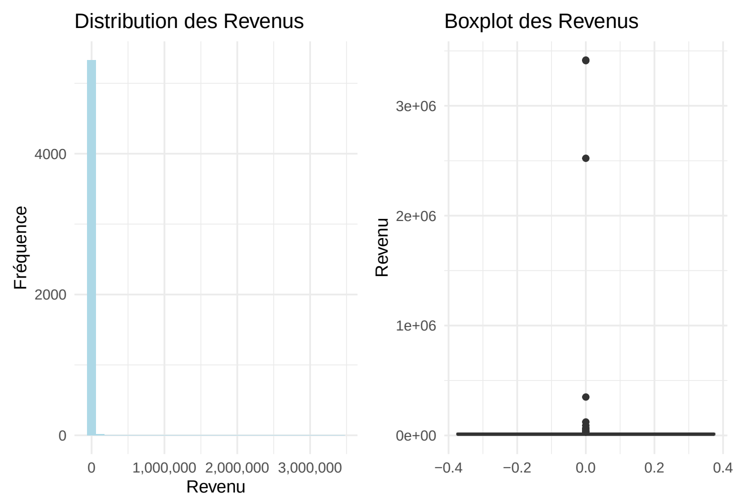
```
# Statistiques descriptives des variables numériques principales
summary(dat$reves)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    1000    8500   11250   14880   16250 3416250
```

```
library(patchwork)
```

```
## 
## Attaching package: 'patchwork'

## The following object is masked from 'package:MASS':
## 
##     area
```

```
hist_reves <- ggplot(dat, aes(x = reves)) +
  geom_histogram(bins = 30, fill = "lightblue") +
  theme_minimal() +
  labs(title = "Distribution des Revenus", x = "Revenu", y = "Fréquence") +
  scale_x_continuous(labels = scales::comma_format())

box_reves <- ggplot(dat, aes(y = reves)) +
  geom_boxplot(fill = "lightblue") +
  theme_minimal() +
  labs(title = "Boxplot des Revenus", y = "Revenu")

# Affichage côte à côte
hist_reves + box_reves
```

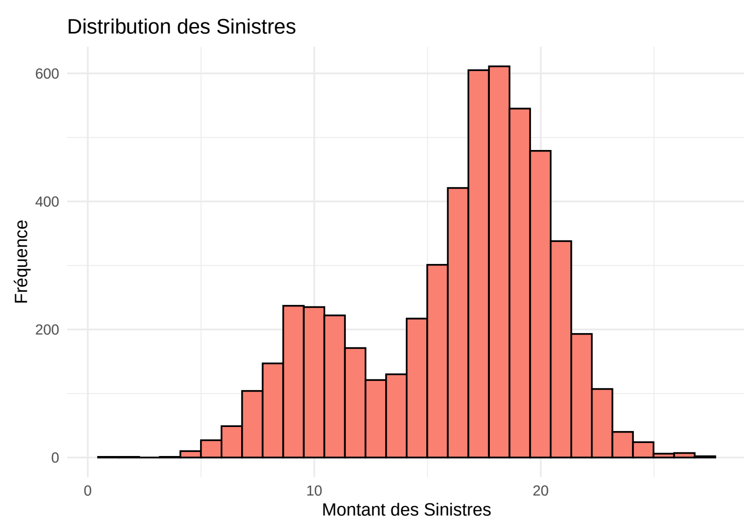


b) Sinistres

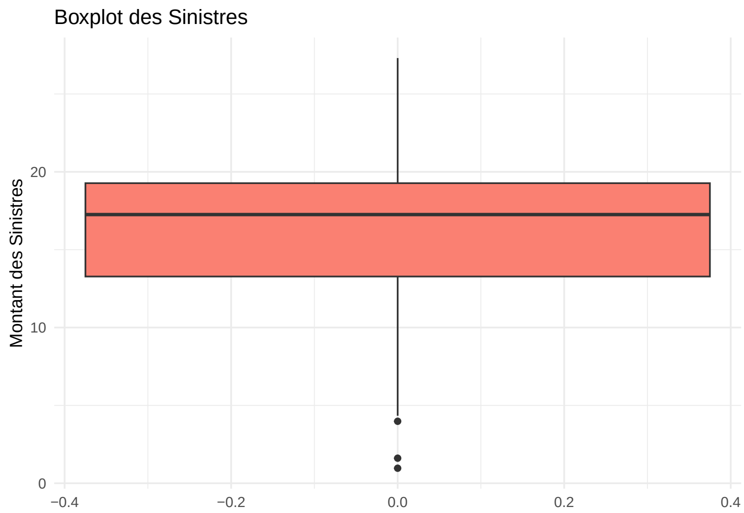
```r
# Statistiques descriptives des variables numériques principales
summary(dat$Sinistre0)
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.9652 13.2769 17.2558 16.1732 19.2718 27.3074
```

```r
# Analyse de la distribution de Sinistre0
ggplot(dat, aes(x = Sinistre0)) +
  geom_histogram(bins = 30, fill = "salmon", color = "black") +
  theme_minimal() +
  labs(title = "Distribution des Sinistres", x = "Montant des Sinistres", y = "Fréquence")
```



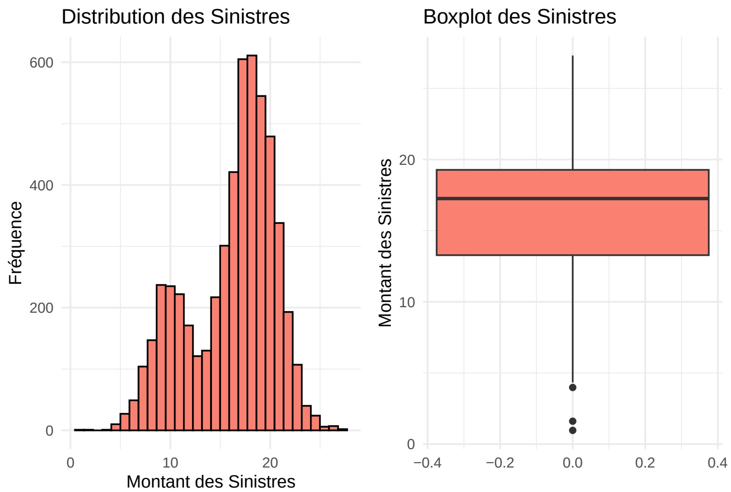

```r
# Boxplot des sinistres
ggplot(dat, aes(y = Sinistre0)) +
  geom_boxplot(fill = "salmon") +
  theme_minimal() +
  labs(title = "Boxplot des Sinistres", y = "Montant des Sinistres")
```




```r
# Résumé statistique
summary(dat$Sinistre0)
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.9652 13.2769 17.2558 16.1732 19.2718 27.3074
```

```r
# Histogramme des sinistres
hist_sinistre <- ggplot(dat, aes(x = Sinistre0)) +
  geom_histogram(bins = 30, fill = "salmon", color = "black") +
  theme_minimal() +
  labs(title = "Distribution des Sinistres", x = "Montant des Sinistres", y = "Fréquence")

# Boxplot des sinistres
box_sinistre <- ggplot(dat, aes(y = Sinistre0)) +
  geom_boxplot(fill = "salmon") +
  theme_minimal() +
  labs(title = "Boxplot des Sinistres", y = "Montant des Sinistres")

# Affichage côte à côte
hist_sinistre + box_sinistre
```



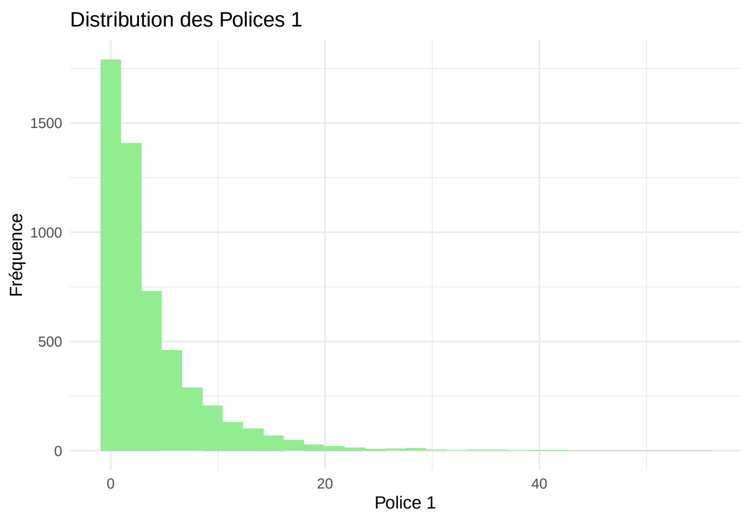
c) Polices

```
# Analyse des variables Police

police_stats <- data.frame(
  Police = c("Police1", "Police2", "Police3"),
  Moyenne = c(mean(dat$Police1), mean(dat$Police2), mean(dat$Police3)),
  Médiane = c(median(dat$Police1), median(dat$Police2), median(dat$Police3)),
  Écart_type = c(sd(dat$Police1), sd(dat$Police2), sd(dat$Police3)),
  Min = c(min(dat$Police1), min(dat$Police2), min(dat$Police3)),
  Max = c(max(dat$Police1), max(dat$Police2), max(dat$Police3))
)

police_stats
```

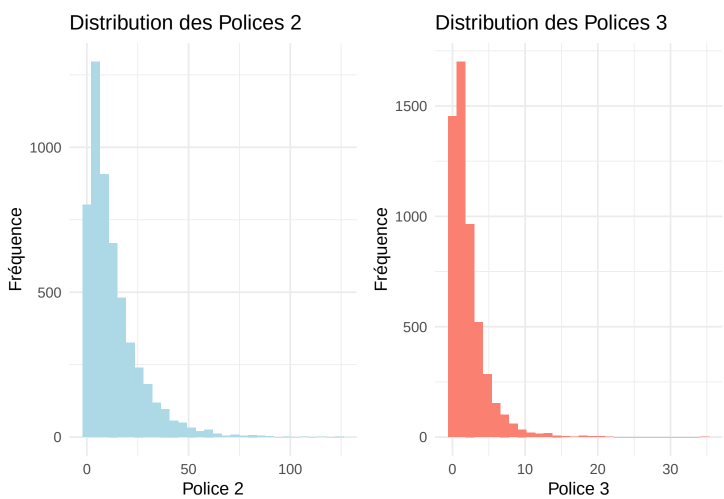
```
##    Police   Moyenne Médiane Écart_type Min     Max
## 1 Police1  3.750700    1.95   5.020503   0  54.985
## 2 Police2 13.017457    9.06  13.261083   0 124.109
## 3 Police3  2.110487    1.42   2.422230   0  34.743
```

```
# Histogrammes des polices
hist_pol1 <- ggplot(dat, aes(x = Police1)) +
  geom_histogram(bins = 30, fill = "lightgreen") +
  theme_minimal() +
  labs(title = "Distribution des Polices 1", x = "Police 1", y = "Fréquence")

hist_pol2 <- ggplot(dat, aes(x = Police2)) +
  geom_histogram(bins = 30, fill = "lightblue") +
  theme_minimal() +
  labs(title = "Distribution des Polices 2", x = "Police 2", y = "Fréquence")

hist_pol3 <- ggplot(dat, aes(x = Police3)) +
  geom_histogram(bins = 30, fill = "salmon") +
  theme_minimal() +
  labs(title = "Distribution des Polices 3", x = "Police 3", y = "Fréquence")
```

```r
# Affichage en ligne
(hist_pol1)
```

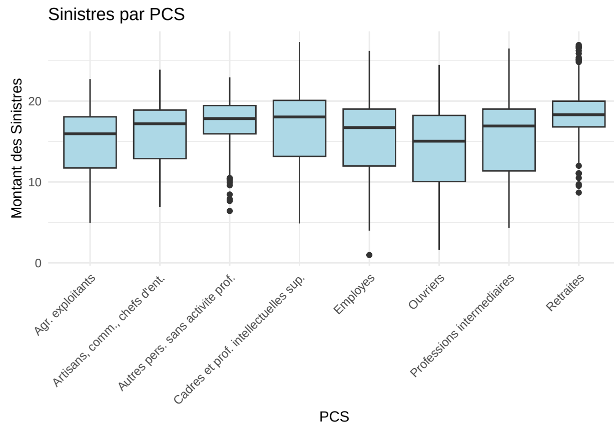



```r
( hist_pol2 | hist_pol3)
```

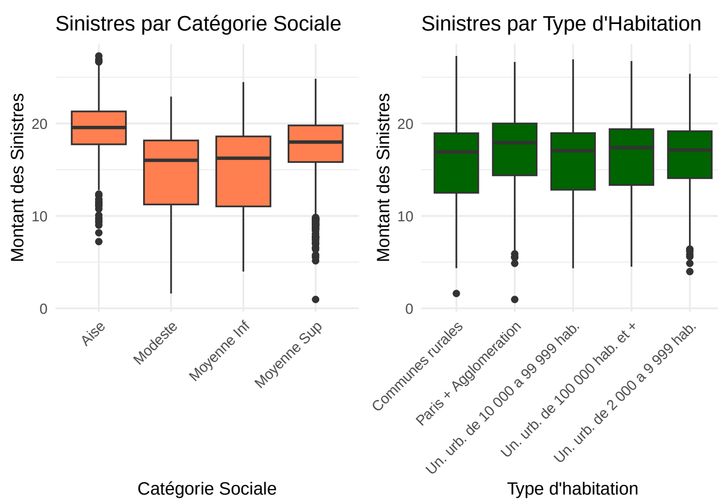


1.4 Analyse des relations

```r
# Boxplot par PCS
box_pcs <- ggplot(dat, aes(x = pcs, y = Sinistre0)) +
  geom_boxplot(fill = "lightblue") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Sinistres par PCS", x = "PCS", y = "Montant des Sinistres")

( box_pcs )
```




```r
# Boxplot par Catégorie Sociale
box_cs <- ggplot(dat, aes(x = cs, y = Sinistre0)) +
  geom_boxplot(fill = "coral") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Sinistres par Catégorie Sociale", x = "Catégorie Sociale", y = "Montant des Sinistres")

# Boxplot par Type d'Habitation
box_hab <- ggplot(dat, aes(x = Ahabi, y = Sinistre0)) +
  geom_boxplot(fill = "darkgreen") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "Sinistres par Type d'Habitation", x = "Type d'habitation", y = "Montant des Sinistres")

# Affichage en grille
( box_cs | box_hab)
```




```r
# Relation entre variables numériques
# Matrice de corrélation
num_vars_corr <- c("reves", "Sinistre0", "Police1", "Police2", "Police3", "NSin", "Duree")
corr_matrix <- cor(dat[, num_vars_corr], use = "complete.obs")
print(round(corr_matrix, 2))
```

```
##            reves Sinistre0 Police1 Police2 Police3  NSin Duree
## reves       1.00      0.01    0.00    0.01    0.01  0.01  0.01
## Sinistre0   0.01      1.00    0.00   -0.28   -0.10 -0.22  0.26
## Police1     0.00      0.00    1.00    0.15    0.03  0.09  0.01
## Police2     0.01     -0.28    0.15    1.00    0.27  0.30 -0.19
## Police3     0.01     -0.10    0.03    0.27    1.00  0.17 -0.08
## NSin        0.01     -0.22    0.09    0.30    0.17  1.00 -0.17
## Duree       0.01      0.26    0.01   -0.19   -0.08 -0.17  1.00
```

```
# Visualisation de la matrice de corrélation
#corrplot(corr_matrix, method = "circle", type = "upper", tl.col = "black",
        # title = "Matrice de Corrélation des Variables Numériques")
```

2.Proposition de tarification

2.1 Tarification a priori

```
# Transformation des variables "reves" et "Police2" en appliquant une transformation logarithmique
dat$log_reves <- log(dat$reves)
dat$log_Police2 <- log(dat$Police2)

# Séparation de l'échantillon en train et test sets
set.seed(1)
train_indices <- sample(1:nrow(dat), size = 0.7 * nrow(dat))
dat_train <- dat[train_indices, ]
dat_test <- dat[-train_indices, ]


modele_apriori <- lm(log(Sinistre0) ~ . - Police1 - Police2 - Police3 - log_Police2, data = dat_train)
```

```
# Modèle Forward
modele_forward <- stepAIC(modele_apriori, direction = "forward", trace=0)
summary(modele_forward)
```

```
##
## Call:
## lm(formula = log(Sinistre0) ~ (pcs + RUC + cs + reves + crevpp +
##     region + habi + Ahabi + Atyph + agecat + Acompm + nbpers +
##     enfants + Anat + Bauto + Nbadulte + Sinistre1 + Sinistre2 +
##     Sinistre3 + Police1 + Police2 + Police3 + durPolice1 + Duree +
##     NSin + censure + log_reves + log_Police2) - Police1 - Police2 -
##     Police3 - log_Police2, data = dat_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.33091 -0.08640  0.00917  0.09986  0.61141
##
## Coefficients: (5 not defined because of singularities)
##                                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)                            2.843e+00  2.118e-01  13.423  < 2e-16
## pcsArtisans, comm., chefs d'ent.      -8.304e-03  2.355e-02  -0.353   0.7244
## pcsAutres pers. sans activite prof.   -1.248e-02  2.346e-02  -0.532   0.5949
## pcsCadres et prof. intellectuelles sup. -1.518e-02  2.134e-02  -0.711   0.4770
```

```
## pcsEmployes                            -2.404e-02  2.007e-02  -1.198   0.2311
## pcsOuvriers                            -3.061e-02  1.932e-02  -1.584   0.1132
## pcsProfessions intermediaires          -2.324e-02  2.011e-02  -1.156   0.2479
## pcsRetraites                           -1.917e-02  2.215e-02  -0.865   0.3869
## RUC                                     1.152e-05  2.477e-06   4.650 3.43e-06
## csModeste                               1.171e-02  2.508e-02   0.467   0.6406
## csMoyenne Inf                           1.828e-03  1.845e-02   0.099   0.9211
## csMoyenne Sup                           3.660e-03  1.384e-02   0.264   0.7914
## reves                                  -1.374e-07  5.146e-07  -0.267   0.7896
## crevpp2eme quartile                     1.927e-02  1.249e-02   1.543   0.1230
## crevpp3eme quartile                     8.117e-03  1.774e-02   0.458   0.6473
## crevpp4eme quartile                     1.222e-02  2.237e-02   0.546   0.5849
## region2                                -1.587e-02  1.824e-02  -0.870   0.3843
## region3                                -7.910e-04  1.992e-02  -0.040   0.9683
## region4                                -9.319e-03  1.890e-02  -0.493   0.6220
## region5                                -6.438e-03  1.845e-02  -0.349   0.7271
## region7                                 8.237e-03  1.913e-02   0.431   0.6668
## region8                                -1.355e-02  1.871e-02  -0.724   0.4692
## region9                                -5.984e-03  1.915e-02  -0.312   0.7547
## habi1                                  -8.084e-03  1.098e-02  -0.736   0.4617
## habi2                                   7.929e-03  1.294e-02   0.613   0.5402
## habi3                                  -1.445e-02  1.251e-02  -1.155   0.2483
## habi4                                  -1.835e-02  1.137e-02  -1.614   0.1065
## habi5                                  -5.288e-03  1.127e-02  -0.469   0.6390
## habi6                                  -3.956e-03  1.063e-02  -0.372   0.7097
## habi7                                  -2.682e-03  8.213e-03  -0.327   0.7440
## habi8                                  -1.521e-02  1.914e-02  -0.795   0.4269
## AhabiParis + Agglomeration                     NA         NA      NA       NA
## AhabiUn. urb. de 10 000 a 99 999 hab.          NA         NA      NA       NA
## AhabiUn. urb. de 100 000 hab. et +             NA         NA      NA       NA
## AhabiUn. urb. de 2 000 a 9 999 hab.            NA         NA      NA       NA
## AtyphNon declare                       -6.564e-03  2.335e-02  -0.281   0.7786
## AtyphProprietaire                       7.521e-04  6.035e-03   0.125   0.9008
## agecat41-50                            -4.101e-03  8.087e-03  -0.507   0.6121
## agecat51-60                            -9.618e-03  9.879e-03  -0.974   0.3304
## agecat61-96                            -9.024e-03  1.442e-02  -0.626   0.5316
## AcompmCouple avec enfant(s)            -6.188e-01  1.103e-02 -56.084  < 2e-16
## AcompmCouple sans enfant               -2.807e-04  1.001e-02  -0.028   0.9776
## AcompmPersonne seule                    1.553e-03  1.689e-02   0.092   0.9267
## nbpers                                  7.735e-03  6.561e-03   1.179   0.2385
## enfantsPas d'enfants                           NA         NA      NA       NA
## AnatMenage francais                    -9.940e-03  1.739e-02  -0.571   0.5677
## AnatNon declare                        -1.915e-02  2.018e-02  -0.949   0.3426
## BautoPas de vehicule                   -1.865e-03  1.044e-02  -0.179   0.8582
## Nbadulte                               -5.347e-03  5.963e-03  -0.897   0.3700
## Sinistre1                              -6.947e-04  3.532e-04  -1.967   0.0493
## Sinistre2                               2.576e-03  2.117e-03   1.217   0.2237
## Sinistre3                               6.056e-04  1.021e-03   0.593   0.5531
## durPolice1                              1.131e-13  7.252e-14   1.560   0.1188
## Duree                                   2.236e-06  5.730e-06   0.390   0.6964
## NSin                                    4.804e-04  8.536e-04   0.563   0.5736
## censure                                 1.435e-02  1.099e-02   1.306   0.1915
## log_reves                              -3.907e-04  2.516e-02  -0.016   0.9876
##
```

```
## (Intercept)                             ***
## pcsArtisans, comm., chefs d'ent.
## pcsAutres pers. sans activite prof.
## pcsCadres et prof. intellectuelles sup.
## pcsEmployes
## pcsOuvriers
## pcsProfessions intermediaires
## pcsRetraites
## RUC                                     ***
## csModeste
## csMoyenne Inf
## csMoyenne Sup
## reves
## crevpp2eme quartile
## crevpp3eme quartile
## crevpp4eme quartile
## region2
## region3
## region4
## region5
## region7
## region8
## region9
## habi1
## habi2
## habi3
## habi4
## habi5
## habi6
## habi7
## habi8
## AhabiParis + Agglomeration
## AhabiUn. urb. de 10 000 a 99 999 hab.
## AhabiUn. urb. de 100 000 hab. et +
## AhabiUn. urb. de 2 000 a 9 999 hab.
## AtyphNon declare
## AtyphProprietaire
## agecat41-50
## agecat51-60
## agecat61-96
## AcompmCouple avec enfant(s)             ***
## AcompmCouple sans enfant
## AcompmPersonne seule
## nbpers
## enfantsPas d'enfants
## AnatMenage francais
## AnatNon declare
## BautoPas de vehicule
## Nbadulte
## Sinistre1                               *
## Sinistre2
## Sinistre3
## durPolice1
## Duree
```

```
## NSin
## censure
## log_reves
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1576 on 3694 degrees of freedom
## Multiple R-squared:   0.76,  Adjusted R-squared:  0.7566
## F-statistic: 229.3 on 51 and 3694 DF,  p-value: < 2.2e-16
```

```
# Évaluation du modèle sur l'échantillon de test
predictions <- predict(modele_forward, newdata = dat_test)
mse <- mean((log(dat_test$Sinistre0) - predictions)^2)
cat("Mean Squared Error (MSE) on test set:", mse, "\n")
```

```
## Mean Squared Error (MSE) on test set: 0.02398579
```

```
# Modele Backward
modele_backward <- stepAIC(modele_apriori, direction = "backward", trace=0)
summary(modele_backward)
```

```
##
## Call:
## lm(formula = log(Sinistre0) ~ RUC + Acompm + censure, data = dat_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.34226 -0.08536  0.00903  0.10136  0.59834
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 2.818e+00  5.835e-03 482.951   <2e-16 ***
## RUC                         1.138e-05  9.285e-07  12.261   <2e-16 ***
## AcompmCouple avec enfant(s) -6.074e-01  6.614e-03 -91.836   <2e-16 ***
## AcompmCouple sans enfant    -6.733e-03  7.132e-03  -0.944   0.3452
## AcompmPersonne seule        -7.325e-03  8.514e-03  -0.860   0.3897
## censure                     1.210e-02  6.841e-03   1.768   0.0771 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1573 on 3740 degrees of freedom
## Multiple R-squared:  0.7579, Adjusted R-squared:  0.7575
## F-statistic:  2341 on 5 and 3740 DF,  p-value: < 2.2e-16
```

```
# Évaluation du modèle sur l'échantillon de test
predictions <- predict(modele_backward, newdata = dat_test)
mse <- mean((log(dat_test$Sinistre0) - predictions)^2)
cat("Mean Squared Error (MSE) on test set:", mse, "\n")
```

```
## Mean Squared Error (MSE) on test set: 0.02358581
```

```r
# Modèle Stepwise
modele_stepwise <- stepAIC(modele_apriori, direction = "both", trace=0)
summary(modele_stepwise)
```

```
##
## Call:
## lm(formula = log(Sinistre0) ~ RUC + Acompm + censure, data = dat_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.34226 -0.08536  0.00903  0.10136  0.59834
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 2.818e+00  5.835e-03 482.951   <2e-16 ***
## RUC                         1.138e-05  9.285e-07  12.261   <2e-16 ***
## AcompmCouple avec enfant(s) -6.074e-01  6.614e-03 -91.836   <2e-16 ***
## AcompmCouple sans enfant    -6.733e-03  7.132e-03  -0.944   0.3452
## AcompmPersonne seule        -7.325e-03  8.514e-03  -0.860   0.3897
## censure                     1.210e-02  6.841e-03   1.768   0.0771 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1573 on 3740 degrees of freedom
## Multiple R-squared:  0.7579, Adjusted R-squared:  0.7575
## F-statistic:  2341 on 5 and 3740 DF,  p-value: < 2.2e-16
```

```r
# Évaluation du modèle sur l'échantillon de test
predictions <- predict(modele_stepwise, newdata = dat_test)
mse <- mean((log(dat_test$Sinistre0) - predictions)^2)
cat("Mean Squared Error (MSE) on test set:", mse, "\n")
```

```
## Mean Squared Error (MSE) on test set: 0.02358581
```

```r
# Fonction pour calculer le R² sur l'échantillon test
test_r_squared <- function(model, test_data) {
  y_true <- log(test_data$Sinistre0)
  y_pred <- predict(model, newdata = test_data)
  ss_res <- sum((y_true - y_pred)^2)
  ss_tot <- sum((y_true - mean(y_true))^2)
  1 - (ss_res / ss_tot)
}

# Création d'un dataframe comparatif
comparaison_modeles <- data.frame(
  Modèle = c("Forward","Backward", "Stepwise"),
  MSE_Test = c(
    mean((log(dat_test$Sinistre0) - predict(modele_forward, dat_test))^2),
    mean((log(dat_test$Sinistre0) - predict(modele_backward, dat_test))^2),
    mean((log(dat_test$Sinistre0) - predict(modele_stepwise, dat_test))^2)

  ),
```

```r
  R2_Test = c(
    test_r_squared(modele_forward, dat_test),
    test_r_squared(modele_backward, dat_test),
    test_r_squared(modele_stepwise, dat_test)

  )
)

# Affichage du tableau

kable(comparaison_modeles,
      digits = 6,
      col.names = c("Modèle", "MSE", "R²"),
      caption = "Comparaison des performances des modèles")
```

Table 1: Comparaison des performances des modèles

Modèle	MSE	R^2
Forward	0.023986	0.752725
Backward	0.023586	0.756849
Stepwise	0.023586	0.756849

```r
#Présentation du modèle retenu
summary(modele_stepwise)
```

```
##
## Call:
## lm(formula = log(Sinistre0) ~ RUC + Acompm + censure, data = dat_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.34226 -0.08536  0.00903  0.10136  0.59834
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 2.818e+00  5.835e-03 482.951   <2e-16 ***
## RUC                         1.138e-05  9.285e-07  12.261   <2e-16 ***
## AcompmCouple avec enfant(s) -6.074e-01  6.614e-03 -91.836   <2e-16 ***
## AcompmCouple sans enfant    -6.733e-03  7.132e-03  -0.944   0.3452
## AcompmPersonne seule        -7.325e-03  8.514e-03  -0.860   0.3897
## censure                      1.210e-02  6.841e-03   1.768   0.0771 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1573 on 3740 degrees of freedom
## Multiple R-squared:  0.7579, Adjusted R-squared:  0.7575
## F-statistic:  2341 on 5 and 3740 DF,  p-value: < 2.2e-16
```

```r
# Tests avant de proposer un modèle a posteriori

# Test de Breusch-Pagan pour l'hétéroscédasticité
```

```r
bp_test <- bptest(modele_stepwise)
print(bp_test)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  modele_stepwise
## BP = 96.108, df = 5, p-value < 2.2e-16
```

```r
# Test de Shapiro-Wilk pour la normalité des résidus
shapiro_test <- shapiro.test(residuals(modele_stepwise))
print(shapiro_test)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(modele_stepwise)
## W = 0.94343, p-value < 2.2e-16
```

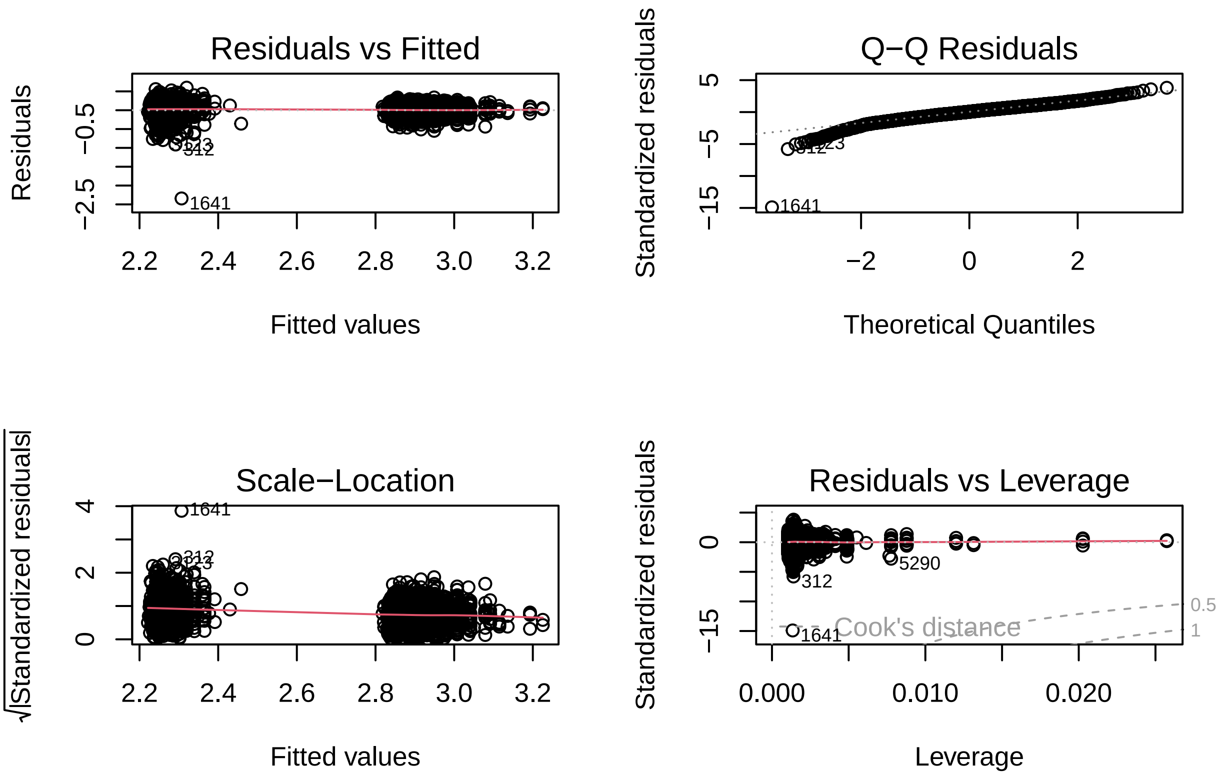
```r
# Test de Durbin-Watson pour l'autocorrélation
dw_test <- dwtest(modele_stepwise)
print(dw_test)
```

```
##
##  Durbin-Watson test
##
## data:  modele_stepwise
## DW = 2.016, p-value = 0.6879
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
# Vérifier la multicolinéarité
vif_values <- vif(modele_stepwise)
print(vif_values)
```

```
##             GVIF Df GVIF^(1/(2*Df))
## RUC     1.816418  1        1.347746
## Acompm  1.216259  3        1.033168
## censure 1.672862  1        1.293392
```

```r
# Graphiques
par(mfrow = c(2, 2))
plot(modele_stepwise)
```




```r
par(mfrow = c(1, 1))
```

```r
# Visualiser les données du point 1641
point_1641 <- dat_train[1641, ]
point_1641
```

```
##                                             pcs       RUC          cs reves
## 2770 Cadres et prof. intellectuelles sup. 3064.516 Moyenne Inf  9500
##            crevpp region habi                              Ahabi         Atyph agecat
## 2770 1er quartile       9    7 Un. urb. de 100 000 hab. et + Proprietaire   51-60
##            Acompm nbpers       enfants              Anat             Bauto Nbadulte
## 2770 Autre menage       4 Pas d'enfants Menage francais Au - 1 vehicule        4
##      Sinistre1 Sinistre2 Sinistre3 Police1 Police2 Police3 durPolice1 Duree
## 2770         2         0      9.89     7.6   17.86    3.96   0.156495    16
##      NSin censure Sinistre0 log_reves log_Police2
## 2770   10       0  14.83813  9.159047    2.882564
```

```r
# Modèle GLM Gamma initial avec toutes les variables
modele_gamma_complet <- glm(Sinistre0 ~ . - Police1 - Police2 - Police3 - log_Police2,
                            family = Gamma(link = "log"),
                            data = dat_train)

# Modèle minimal
modele_gamma_null <- glm(Sinistre0 ~ 1, family = Gamma(link = "log"), data = dat_train)
```

```r
# Comparaison
modele_gamma_forward <- step(modele_gamma_null, scope = list(lower = formula(modele_gamma_null),
                                                             upper = formula(modele_gamma_complet)),
                             direction = "forward", trace = 0)
modele_gamma_backward <- step(modele_gamma_complet, direction = "backward", trace = 0)

modele_gamma_both <- step(modele_gamma_complet, direction = "both", trace = 0)
```

```r
# Fonction pour calculer le R² pour GLM Gamma
test_r_squared_gamma <- function(model, test_data) {
  y_true <- test_data$Sinistre0
  y_pred <- predict(model, newdata = test_data, type = "response")
  ss_res <- sum((y_true - y_pred)^2)
  ss_tot <- sum((y_true - mean(y_true))^2)
  1 - (ss_res / ss_tot)
}

# Créer un dataframe comparatif complet
comparaison_glm <- data.frame(
  Modèle = c("Forward GLM Gamma", "Backward GLM Gamma", "Stepwise GLM Gamma"),
  AIC = c(
    AIC(modele_gamma_forward),
    AIC(modele_gamma_backward),
    AIC(modele_gamma_both)
  ),
  BIC = c(
    BIC(modele_gamma_forward),
    BIC(modele_gamma_backward),
    BIC(modele_gamma_both)
  ),
  Déviance = c(
    deviance(modele_gamma_forward),
    deviance(modele_gamma_backward),
    deviance(modele_gamma_both)
  ),
  MSE_Test = c(
    mean((dat_test$Sinistre0 - predict(modele_gamma_forward, dat_test, type = "response"))^2),
    mean((dat_test$Sinistre0 - predict(modele_gamma_backward, dat_test, type = "response"))^2),
    mean((dat_test$Sinistre0 - predict(modele_gamma_both, dat_test, type = "response"))^2)
  ),
  R2_Test = c(
    test_r_squared_gamma(modele_gamma_forward, dat_test),
    test_r_squared_gamma(modele_gamma_backward, dat_test),
    test_r_squared_gamma(modele_gamma_both, dat_test)
  )
)

# Afficher le tableau formaté
kable(comparaison_glm,
      digits = 6,
      col.names = c("Modèle", "AIC", "BIC", "Déviance", "MSE", "R²"),
      caption = "Comparaison des performances des modèles GLM Gamma")
```

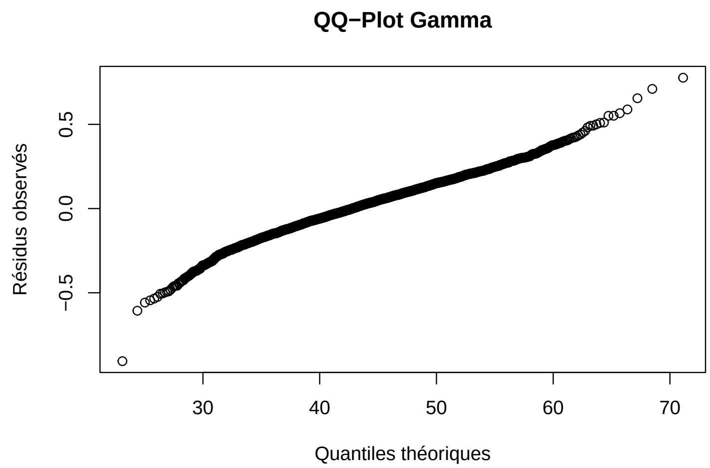
Table 2: Comparaison des performances des modèles GLM Gamma

Modèle	AIC	BIC	Déviance	MSE	R^2
Forward GLM Gamma	17081.56	17125.15	87.38382	4.390765	0.750685
Backward GLM Gamma	17081.56	17125.15	87.38382	4.390765	0.750685
Stepwise GLM Gamma	17081.56	17125.15	87.38382	4.390765	0.750685

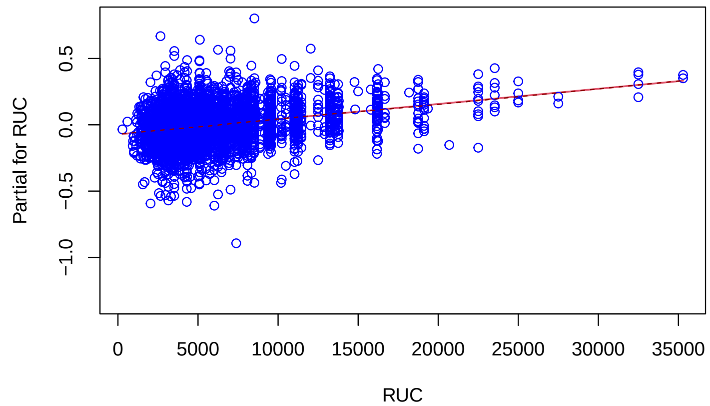
```r
summary(modele_gamma_both)
```

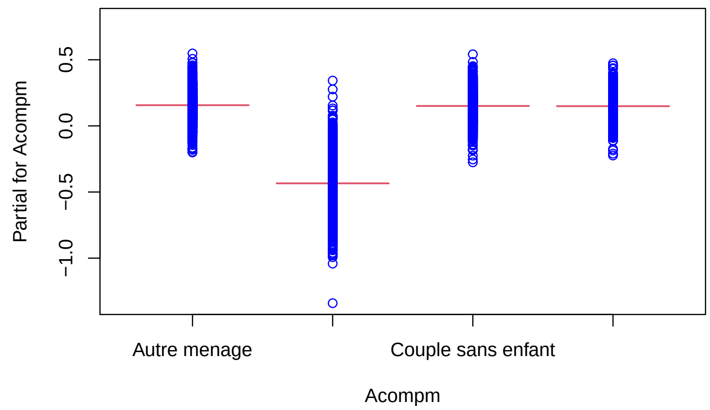
```
##
## Call:
## glm(formula = Sinistre0 ~ RUC + Acompm + censure, family = Gamma(link = "log"),
##     data = dat_train)
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 2.825e+00  5.491e-03 514.549   <2e-16 ***
## RUC                         1.147e-05  8.737e-07  13.125   <2e-16 ***
## AcompmCouple avec enfant(s) -5.911e-01  6.224e-03 -94.969   <2e-16 ***
## AcompmCouple sans enfant    -5.984e-03  6.711e-03  -0.892   0.3727
## AcompmPersonne seule        -7.451e-03  8.012e-03  -0.930   0.3524
## censure                      1.136e-02  6.437e-03   1.764   0.0778 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Gamma family taken to be 0.02190342)
##
##     Null deviance: 336.313  on 3745  degrees of freedom
## Residual deviance:  87.384  on 3740  degrees of freedom
## AIC: 17082
##
## Number of Fisher Scoring iterations: 4
```

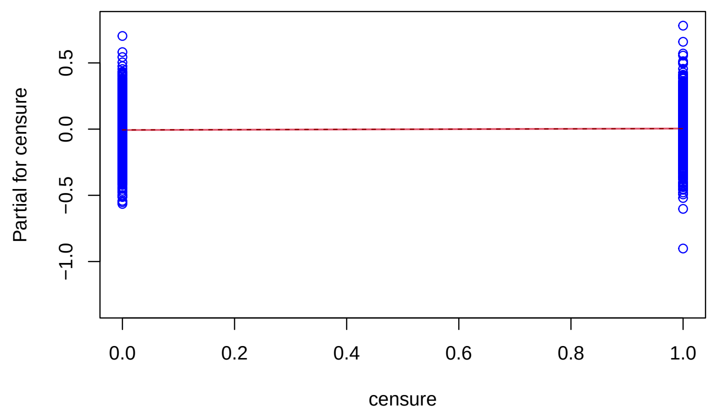
```r
res_pearson <- residuals(modele_gamma_both, type = "pearson")
qq.plot <- qqplot(qgamma(ppoints(res_pearson), shape = gamma.shape(modele_gamma_both)$alpha), res_pears
                  main = "QQ-Plot Gamma", xlab = "Quantiles théoriques", ylab = "Résidus observés")
abline(0, 1, col = "red")
```




```r
# Validité du lien logarithmique

termplot(modele_gamma_both, partial.resid = TRUE, col.res = "blue", smooth = panel.smooth)
```







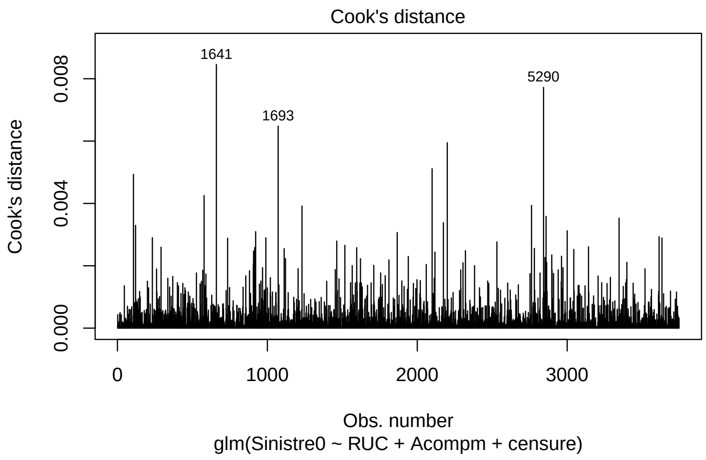

```r
#Test de Cameron & Trivedi
res <- residuals(modele_gamma_both, type = "pearson")

# Régression pondérée des résidus au carré sur les prédits
modele_cam_triv <- lm(res^2 ~ fitted(modele_gamma_both), weights = 1/fitted(modele_gamma_both))

statistique_test <- summary(modele_cam_triv)$r.squared * length(res)
p_value <- pchisq(statistique_test, df = 1, lower.tail = FALSE)

cat("Statistique de test Cameron & Trivedi :", statistique_test, "\n")
```

```
## Statistique de test Cameron & Trivedi : 315.1727
```

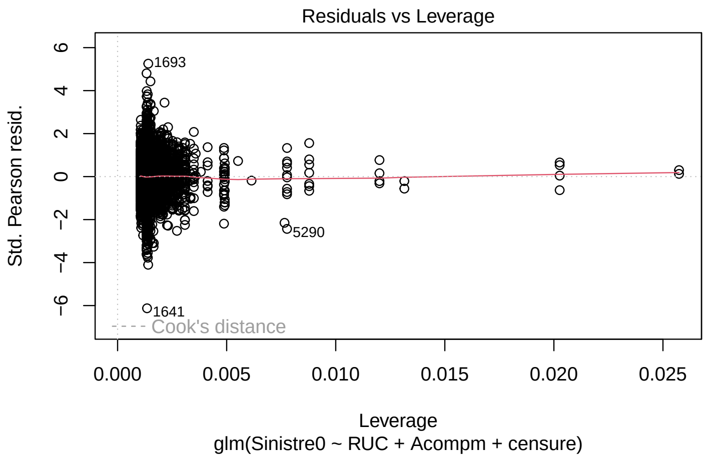
```r
cat("p-value :", p_value, "\n")
```

```
## p-value : 1.630885e-70
```

```r
#Absence de points influents
plot(modele_gamma_both, 4)  # Distance de Cook
```




```r
plot(modele_gamma_both, 5)  # Leviers vs résidus standardisés
```




```r
# Création du modèle GLM avec régularisation Lasso pour distribution Gamma

# Préparation des données pour glmnet
X_train <- model.matrix(Sinistre0 ~ . - Police1 - Police2 - Police3 - log_Police2 - 1,
                        data = dat_train)
y_train <- dat_train$Sinistre0

X_test <- model.matrix(Sinistre0 ~ . - Police1 - Police2 - Police3 - log_Police2 - 1,
                       data = dat_test)
y_test <- dat_test$Sinistre0

# Transformation pour approximer une régression Gamma
```

```r
log_y_train <- log(y_train)

# Validation croisée avec family="gaussian" sur données transformées
set.seed(123)
cv_lasso <- cv.glmnet(X_train, log_y_train,
                      family = "gaussian",
                      alpha = 1,
                      nfolds = 10)

# Lambda optimal
lambda_min <- cv_lasso$lambda.min
lambda_1se <- cv_lasso$lambda.1se

# Modèles avec lambda optimal
modele_lasso_min <- glmnet(X_train, log_y_train,
                          family = "gaussian",
                          alpha = 1,
                          lambda = lambda_min)

modele_lasso_1se <- glmnet(X_train, log_y_train,
                          family = "gaussian",
                          alpha = 1,
                          lambda = lambda_1se)


# Prédictions sur l'échantillon test

# Correction pour le biais de transformation logarithmique
log_predictions <- predict(modele_lasso_1se, newx = X_test)
sigma2 <- mean((log_y_train - predict(modele_lasso_1se, newx = X_train))^2)
predictions <- exp(log_predictions + sigma2/2)  # Correction pour distribution log-normale

# 1. MSE et MAE
mse <- mean((y_test - predictions)^2)
cat("\nMSE:", mse, "\n")
```

```
##
## MSE: 4.502466
```

```r
# 2. R² (coefficient de détermination)
r2 <- 1 - sum((y_test - predictions)^2) / sum((y_test - mean(y_test))^2)
cat("R²:", r2, "\n")
```

```
## R²: 0.7443425
```

2.2 Tarification a posteriori

```r
library(AER)
```

```
## Loading required package: sandwich
```

```
## Loading required package: survival
```

```r
#Test Endogenéité

dat$Police1 <- ifelse(dat$Police1 == 0, 1e-5, dat$Police1)
dat$Police2 <- ifelse(dat$Police2 == 0, 1e-5, dat$Police2)
dat$Police3 <- ifelse(dat$Police3 == 0, 1e-5, dat$Police3)

model_iv_all <- ivreg(
  log(Sinistre0) ~ pcs + RUC + cs + log(reves) + crevpp + Ahabi +
    Atyph + agecat + Acompm + nbpers + enfants + Anat + Bauto +
    Nbadulte + Duree + NSin + censure + Police1 + log(Police2) + Police3
  | . -Police1 -log(Police2) -Police3 + region + habi,
  data = dat
)

summary(model_iv_all, diagnostics = TRUE)
```

```
##
## Call:
## ivreg(formula = log(Sinistre0) ~ pcs + RUC + cs + log(reves) +
##     crevpp + Ahabi + Atyph + agecat + Acompm + nbpers + enfants +
##     Anat + Bauto + Nbadulte + Duree + NSin + censure + Police1 +
##     log(Police2) + Police3 | . - Police1 - log(Police2) - Police3 +
##     region + habi, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.33539 -0.08862  0.01137  0.10766  0.63822
##
## Coefficients:
##                                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)                             2.840e+00  1.123e-01  25.284  < 2e-16
## pcsArtisans, comm., chefs d'ent.        4.018e-04  2.092e-02   0.019 0.984679
## pcsAutres pers. sans activite prof.     5.467e-03  2.189e-02   0.250 0.802777
## pcsCadres et prof. intellectuelles sup.  1.411e-03  1.902e-02   0.074 0.940842
## pcsEmployes                            -8.527e-03  1.765e-02  -0.483 0.628965
## pcsOuvriers                            -1.553e-02  1.692e-02  -0.917 0.358967
## pcsProfessions intermediaires          -1.814e-02  1.787e-02  -1.015 0.310016
## pcsRetraites                           -6.651e-03  1.992e-02  -0.334 0.738436
## RUC                                     8.873e-06  2.598e-06   3.416 0.000641
## csModeste                              -3.132e-03  2.088e-02  -0.150 0.880782
## csMoyenne Inf                           2.866e-03  1.620e-02   0.177 0.859594
## csMoyenne Sup                           9.566e-03  1.231e-02   0.777 0.437228
## log(reves)                              3.437e-03  1.354e-02   0.254 0.799636
## crevpp2eme quartile                     1.128e-02  1.208e-02   0.934 0.350515
## crevpp3eme quartile                     1.213e-02  1.781e-02   0.681 0.495873
## crevpp4eme quartile                     1.254e-02  2.166e-02   0.579 0.562770
## AhabiParis + Agglomeration             -8.133e-03  1.113e-02  -0.731 0.464982
## AhabiUn. urb. de 10 000 a 99 999 hab.  -5.690e-03  7.373e-03  -0.772 0.440317
## AhabiUn. urb. de 100 000 hab. et +      1.091e-05  6.684e-03   0.002 0.998698
## AhabiUn. urb. de 2 000 a 9 999 hab.     4.677e-03  8.283e-03   0.565 0.572308
## AtyphNon declare                        1.544e-02  2.223e-02   0.694 0.487512
## AtyphProprietaire                       6.213e-03  6.626e-03   0.938 0.348452
## agecat41-50                             3.031e-03  8.251e-03   0.367 0.713382
```

```
## agecat51-60                            -1.151e-02  1.099e-02  -1.047 0.295222
## agecat61-96                            -5.283e-03  1.502e-02  -0.352 0.725083
## AcompmCouple avec enfant(s)            -6.091e-01  9.840e-03 -61.898  < 2e-16
## AcompmCouple sans enfant               -3.629e-03  1.139e-02  -0.319 0.750000
## AcompmPersonne seule                   -2.046e-02  2.168e-02  -0.944 0.345414
## nbpers                                  1.822e-02  1.280e-02   1.423 0.154753
## AnatMenage francais                    -6.396e-03  1.779e-02  -0.359 0.719267
## AnatNon declare                        -3.501e-02  2.044e-02  -1.713 0.086712
## BautoPas de vehicule                   -2.876e-03  9.667e-03  -0.297 0.766102
## Nbadulte                               -9.812e-03  8.565e-03  -1.146 0.252032
## Duree                                   5.119e-06  8.342e-06   0.614 0.539440
## NSin                                    1.967e-03  1.071e-03   1.836 0.066462
## censure                                 4.371e-03  1.040e-02   0.420 0.674228
## Police1                                -2.225e-03  7.082e-03  -0.314 0.753391
## log(Police2)                           -2.654e-02  2.356e-02  -1.127 0.259838
## Police3                                -4.810e-03  1.187e-02  -0.405 0.685356
##
## (Intercept)                            ***
## pcsArtisans, comm., chefs d'ent.
## pcsAutres pers. sans activite prof.
## pcsCadres et prof. intellectuelles sup.
## pcsEmployes
## pcsOuvriers
## pcsProfessions intermediaires
## pcsRetraites
## RUC                                    ***
## csModeste
## csMoyenne Inf
## csMoyenne Sup
## log(reves)
## crevpp2eme quartile
## crevpp3eme quartile
## crevpp4eme quartile
## AhabiParis + Agglomeration
## AhabiUn. urb. de 10 000 a 99 999 hab.
## AhabiUn. urb. de 100 000 hab. et +
## AhabiUn. urb. de 2 000 a 9 999 hab.
## AtyphNon declare
## AtyphProprietaire
## agecat41-50
## agecat51-60
## agecat61-96
## AcompmCouple avec enfant(s)            ***
## AcompmCouple sans enfant
## AcompmPersonne seule
## nbpers
## AnatMenage francais
## AnatNon declare                        .
## BautoPas de vehicule
## Nbadulte
## Duree
## NSin                                   .
## censure
## Police1
```

```
## log(Police2)
## Police3
##
## Diagnostic tests:
##                                   df1  df2 statistic p-value
## Weak instruments (Police1)        11 5305    13.289  <2e-16 ***
## Weak instruments (log(Police2))   11 5305     1.478   0.132
## Weak instruments (Police3)        11 5305    23.816  <2e-16 ***
## Wu-Hausman                         3 5310     0.956   0.413
## Sargan                            12   NA     5.708   0.930
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1671 on 5313 degrees of freedom
## Multiple R-Squared: 0.7242,  Adjusted R-squared: 0.7222
## Wald test: 384.7 on 38 and 5313 DF,  p-value: < 2.2e-16
```

```r
# Transformation Police2
dat <- dat %>%
  mutate(log_Police2 = log1p(Police2))  # log(1 + Police2)

# Recréer les échantillons train/test
set.seed(1)
train_indices <- sample(1:nrow(dat), size = 0.7 * nrow(dat))
dat_train <- dat[train_indices, ]
dat_test <- dat[-train_indices, ]

# Modèle linéaire complet
modele_posteriori_complet <- lm(
  log(Sinistre0) ~ pcs + RUC + cs + log(reves) + crevpp + Ahabi +
    Atyph + agecat + Acompm + nbpers + enfants + Anat + Bauto +
    Nbadulte + Duree + NSin + censure + Police1 + log_Police2 + Police3,
  data = dat_train
)

# Modèle stepAIC
library(MASS)
modele_stepwise <- step(modele_posteriori_complet, direction = "both", trace = 0)

# Modèle GLM avec famille Gamma
model_glm_gamma <- glm(
  Sinistre0 ~ pcs + RUC + cs + log(reves) + crevpp + Ahabi +
    Atyph + agecat + Acompm + nbpers + enfants + Anat + Bauto +
    Nbadulte + Duree + NSin + censure + Police1 + log(Police2) + Police3,
  family = Gamma(link = "log"),
  data = dat_train
)

# Version avec sélection AIC pour GLM
model_glm_step <- step(model_glm_gamma, direction = "both", trace = 0)

# Fonction pour calculer les métriques incluant R²
calculate_metrics <- function(model, test_data) {
  # Prédictions
```

```r
  if (inherits(model, "glm") && !inherits(model, "lm")) {
    # Pour les GLM
    pred <- predict(model, newdata = test_data, type = "response")
    observed <- test_data$Sinistre0
  } else if (inherits(model, c("lm", "glmnet"))) {
    # Pour les modèles linéaires
    pred <- exp(predict(model, newdata = test_data))
    observed <- test_data$Sinistre0
  }

  # Calcul des erreurs
  residuals <- observed - pred
  mse <- mean(residuals^2)
  rmse <- sqrt(mse)
  mae <- mean(abs(residuals))

  # Calcul du R²
  ss_total <- sum((observed - mean(observed))^2)
  ss_residual <- sum(residuals^2)
  r_squared <- 1 - (ss_residual / ss_total)

  # AIC et BIC
  if (inherits(model, c("lm", "glm"))) {
    aic <- AIC(model)
    bic <- BIC(model)
  } else {
    aic <- NA
    bic <- NA
  }

  return(c(
    MAE = mae,
    MSE = mse,
    RMSE = rmse,
    R_squared = r_squared,
    AIC = aic,
    BIC = bic
  ))
}

# Évaluation des modèles
results_complet <- calculate_metrics(modele_posteriori_complet, dat_test)
results_stepwise <- calculate_metrics(modele_stepwise, dat_test)

results_glm <- calculate_metrics(model_glm_gamma, dat_test)
results_glm_step <- calculate_metrics(model_glm_step, dat_test)

# Création du tableau de comparaison
tableau_comparaison <- rbind(
  "Modèle Linéaire Complet" = results_complet,
  "Modèle StepAIC" = results_stepwise,
  "Modèle GLM Gamma" = results_glm,
  "Modèle GLM Gamma StepAIC" = results_glm_step
```

```r
)

# Affichage du tableau
kable(
  round(tableau_comparaison, 6),
  caption = "Comparaison des performances des modèles a posteriori",
  digits = 6)
```

Table 3: Comparaison des performances des modèles a posteriori

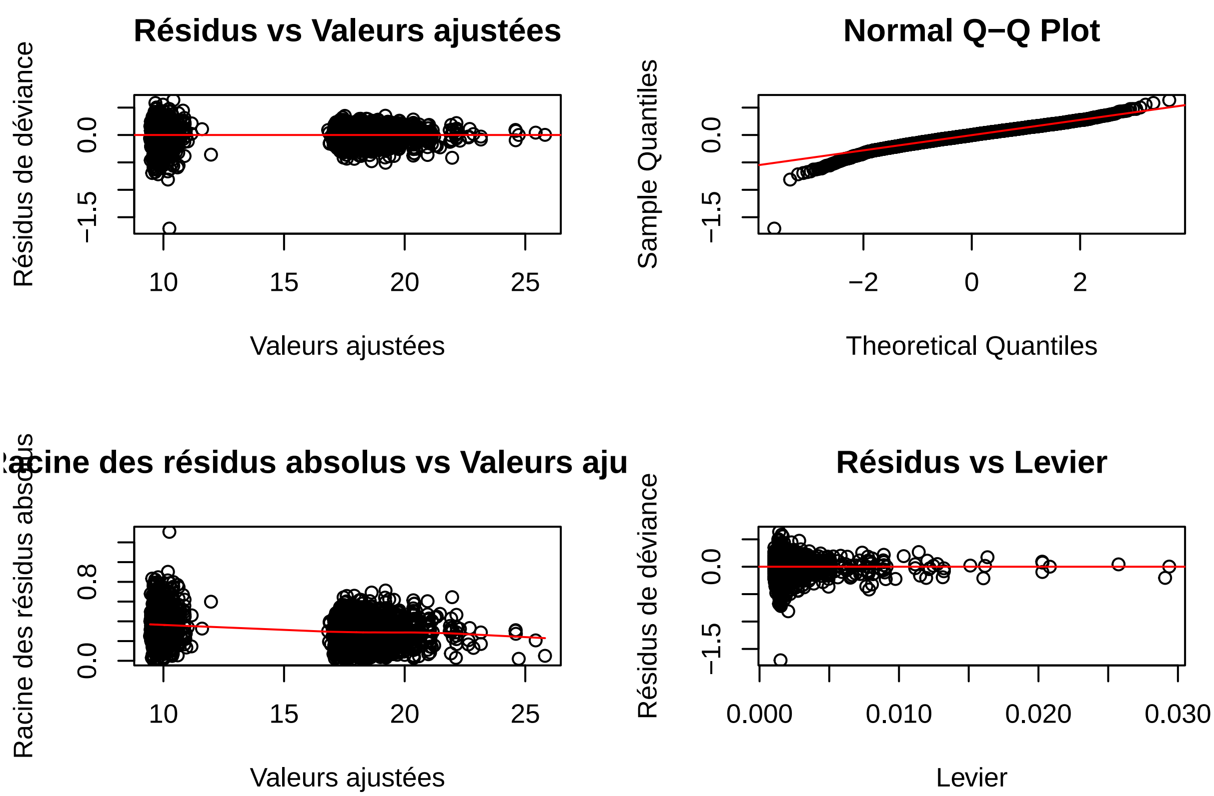
	MAE	MSE	RMSE	R_squared	AIC	BIC
Modèle Linéaire Complet	1.692183	4.455078	2.110705	0.747033	-3174.195	-2925.057
Modèle StepAIC	1.688181	4.436672	2.106341	0.748078	-3220.295	-3170.467
Modèle GLM Gamma	1.683447	4.404143	2.098605	0.749925	17127.384	17376.521
Modèle GLM Gamma StepAIC	1.680565	4.392196	2.095757	0.750604	17080.690	17130.517

```r
summary(model_glm_step)
```

```
##
## Call:
## glm(formula = Sinistre0 ~ RUC + Acompm + censure + Police1, family = Gamma(link = "log"),
##     data = dat_train)
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 2.822e+00  5.894e-03 478.681   <2e-16 ***
## RUC                         1.146e-05  8.734e-07  13.126   <2e-16 ***
## AcompmCouple avec enfant(s) -5.907e-01  6.225e-03 -94.890   <2e-16 ***
## AcompmCouple sans enfant    -5.211e-03  6.724e-03  -0.775   0.4384
## AcompmPersonne seule        -6.002e-03  8.054e-03  -0.745   0.4562
## censure                      1.155e-02  6.436e-03   1.794   0.0729 .
## Police1                      8.457e-04  4.837e-04   1.748   0.0805 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Gamma family taken to be 0.02189052)
##
##     Null deviance: 336.313  on 3745  degrees of freedom
## Residual deviance:  87.317  on 3739  degrees of freedom
## AIC: 17081
##
## Number of Fisher Scoring iterations: 4
```

```r
#Validation Modèle GLM Gamma StepAIC

# Analyse des résidus
residus <- residuals(model_glm_step, type = "deviance")
valeurs_ajustees <- predict(model_glm_step, type = "response")

# Graphiques de diagnostic
```

```r
par(mfrow = c(2, 2))
plot(valeurs_ajustees, residus, main = "Résidus vs Valeurs ajustées",
     xlab = "Valeurs ajustées", ylab = "Résidus de déviance")
abline(h = 0, col = "red")

# QQ-plot des résidus
qqnorm(residus)
qqline(residus, col = "red")

# Racine des résidus absolus vs valeurs ajustées (hétéroscédasticité)
plot(valeurs_ajustees, sqrt(abs(residus)),
     main = "Racine des résidus absolus vs Valeurs ajustées",
     xlab = "Valeurs ajustées", ylab = "Racine des résidus absolus")
lines(lowess(valeurs_ajustees, sqrt(abs(residus))), col = "red")

# Résidus vs levier
plot(hatvalues(model_glm_step), residus,
     main = "Résidus vs Levier",
     xlab = "Levier", ylab = "Résidus de déviance")
abline(h = 0, col = "red")
```




```r
par(mfrow = c(1, 1))

# Test de spécification du lien
preds <- predict(model_glm_step, type = "link")
```

```r
model_reset <- update(model_glm_step, . ~ . + I(preds^2) + I(preds^3))
anova(model_glm_step, model_reset, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Sinistre0 ~ RUC + Acompm + censure + Police1
## Model 2: Sinistre0 ~ RUC + Acompm + censure + Police1 + I(preds^2) + I(preds^3)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      3739     87.317
## 2      3737     87.160  2  0.15709  0.02752 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Test de l'endogénéité
model_police1 <- lm(Police1 ~ region + habi + pcs + RUC + cs + log(reves) +
                    crevpp + Ahabi + Atyph + agecat + Acompm + nbpers +
                    enfants + Anat + Bauto + Nbadulte + Duree + NSin + censure,
                    data = dat_train)
model_police2 <- lm(log(Police2) ~ region + habi + pcs + RUC + cs + log(reves) +
                    crevpp + Ahabi + Atyph + agecat + Acompm + nbpers +
                    enfants + Anat + Bauto + Nbadulte + Duree + NSin + censure,
                    data = dat_train)
model_police3 <- lm(Police3 ~ region + habi + pcs + RUC + cs + log(reves) +
                    crevpp + Ahabi + Atyph + agecat + Acompm + nbpers +
                    enfants + Anat + Bauto + Nbadulte + Duree + NSin + censure,
                    data = dat_train)

resid_police1 <- residuals(model_police1)
resid_police2 <- residuals(model_police2)
resid_police3 <- residuals(model_police3)

# Ajout des résidus au modèle principal
dat_train$resid_police1 <- resid_police1
dat_train$resid_police2 <- resid_police2
dat_train$resid_police3 <- resid_police3

# Modèle avec les résidus
model_hausman <- update(model_glm_step, . ~ . + resid_police1 + resid_police2 + resid_police3)
anova(model_glm_step, model_hausman, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Sinistre0 ~ RUC + Acompm + censure + Police1
## Model 2: Sinistre0 ~ RUC + Acompm + censure + Police1 + resid_police1 +
##     resid_police2 + resid_police3
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      3739     87.317
## 2      3736     87.305  3 0.012009   0.9082
```

```r
# Interactions clés
model_glm_inter <- update(model_glm_step, . ~ . + RUC:log(reves) + Police1:Nbadulte)
```

```r
# Modèle GAM

model_gam <- gam(Sinistre0 ~ s(RUC) + s(log(reves)) + Acompm + Nbadulte + s(Police1),
                family = Gamma(link = "log"), data = dat_train)
```

```r
# Évaluation des trois nouveaux modèles
results_glm_inter <- calculate_metrics(model_glm_inter, dat_test)
results_gam <- calculate_metrics(model_gam, dat_test)

# Tableau de comparaison incluant tous les modèles
tableau_comparaison_complet <- rbind(
  "Modèle Linéaire Complet" = results_complet,
  "Modèle StepAIC" = results_stepwise,
  "Modèle GLM Gamma" = results_glm,
  "Modèle GLM Gamma StepAIC" = results_glm_step,
  "Modèle GLM Gamma avec Interactions" = results_glm_inter,
  "Modèle GAM" = results_gam
)

# Affichage du tableau complet
kable(
  round(tableau_comparaison_complet, 6),
  caption = "Comparaison des performances des modèles a posteriori",
  digits = 6)
```

```
## Warning: 'xfun::attr()' is deprecated.
## Use 'xfun::attr2()' instead.
## See help("Deprecated")

## Warning: 'xfun::attr()' is deprecated.
## Use 'xfun::attr2()' instead.
## See help("Deprecated")
```

Table 4: Comparaison des performances des modèles a posteriori

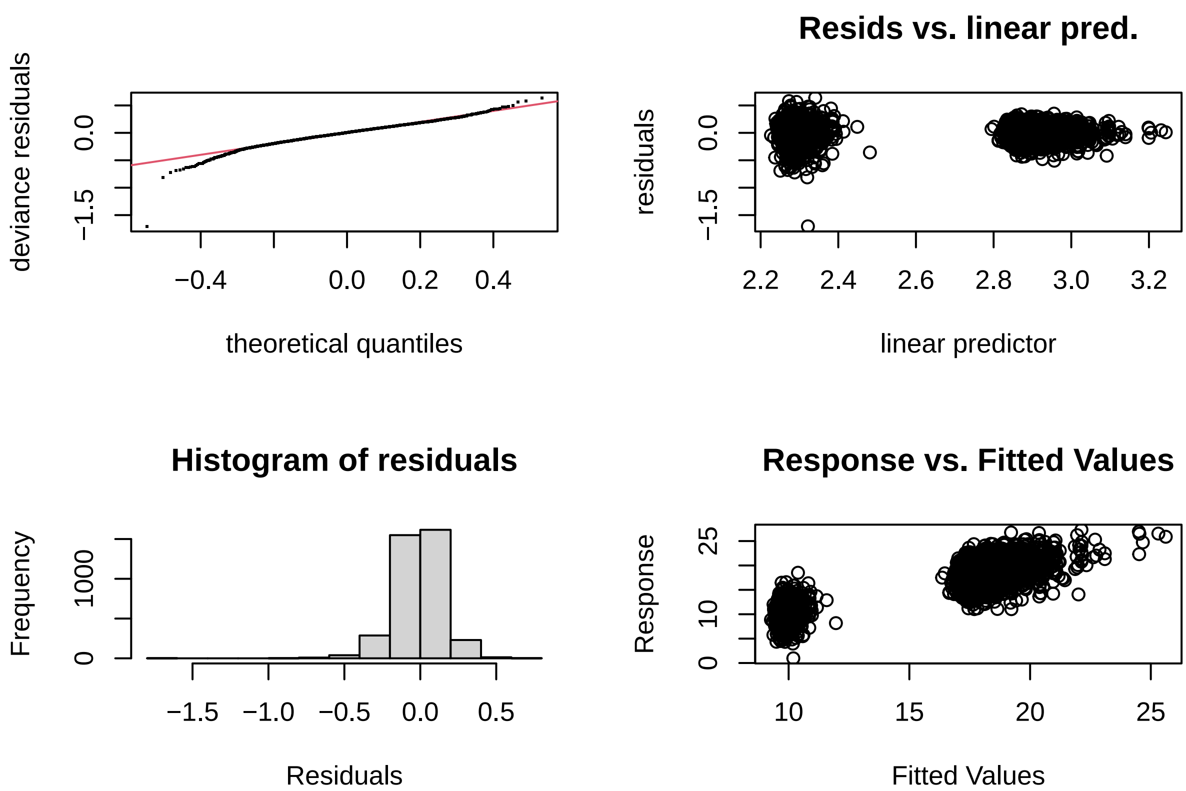
	MAE	MSE	RMSE	R_squared	AIC	BIC
Modèle Linéaire Complet	1.692183	4.455078	2.110705	0.747033	-3174.195	-2925.057
Modèle StepAIC	1.688181	4.436672	2.106341	0.748078	-3220.295	-3170.467
Modèle GLM Gamma	1.683447	4.404143	2.098605	0.749925	17127.384	17376.521
Modèle GLM Gamma StepAIC	1.680565	4.392196	2.095757	0.750604	17080.690	17130.517
Modèle GLM Gamma avec Interactions	1.680271	4.392977	2.095943	0.750559	17084.521	17146.806
Modèle GAM	1.680625	4.392068	2.095726	0.750611	17089.913	17149.291

```r
summary(model_gam)
```

```
##
## Family: Gamma
## Link function: log
##
```

```
## Formula:
## Sinistre0 ~ s(RUC) + s(log(reves)) + Acompm + Nbadulte + s(Police1)
##
## Parametric coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  2.914011   0.013560 214.898   <2e-16 ***
## AcompmCouple avec enfant(s) -0.596299   0.008100 -73.614   <2e-16 ***
## AcompmCouple sans enfant    -0.003744   0.008397  -0.446    0.656
## AcompmPersonne seule        -0.000806   0.013038  -0.062    0.951
## Nbadulte                    -0.004001   0.004129  -0.969    0.333
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                 edf Ref.df      F p-value
## s(RUC)        1.001  1.001 46.871  <2e-16 ***
## s(log(reves)) 1.001  1.002  1.939   0.164
## s(Police1)    1.532  1.904  1.620   0.148
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.749   Deviance explained =   74%
## GCV = 0.023413  Scale est. = 0.021889  n = 3746
```

```r
#Validation des termes lisses

gam.check(model_gam)
```




```
##
## Method: GCV   Optimizer: outer newton
## full convergence after 6 iterations.
## Gradient range [-9.378801e-09,1.512364e-10]
## (score 0.023413 & scale 0.0218893).
## Hessian positive definite, eigenvalue range [4.382411e-09,2.110912e-06].
## Model rank =  32 / 32
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                  k'  edf k-index p-value
## s(RUC)         9.00 1.00    1.00    0.54
## s(log(reves)) 9.00 1.00    0.97    0.04 *
## s(Police1)     9.00 1.53    1.00    0.46
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Vérification de la distribution Gamma

# Calcul des résidus de déviance pour le modèle GAM
res_dev <- residuals(model_gam, type = "deviance")
# Test de normalité (approximatif car Gamma)
shapiro.test(res_dev[sample(length(res_dev), 2000)])
```

```
##
```

```
##  Shapiro-Wilk normality test
##
## data:  res_dev[sample(length(res_dev), 2000)]
## W = 0.97913, p-value < 2.2e-16
```

```r
#Test de comparaison de liens (log vs inverse)
model_gam_inverse <- gam(Sinistre0 ~ s(RUC) + s(log(reves)) + Acompm + Nbadulte + s(Police1),
                        family = Gamma(link = "inverse"), data = dat_train)
AIC(model_gam, model_gam_inverse)
```

```
##                         df      AIC
## model_gam          9.53328 17089.91
## model_gam_inverse 10.01133 17110.96
```

```r
#Diagnostic de concurvité (multicollinéarité pour GAM)

concurvity(model_gam, full = TRUE)
```

```
##               para    s(RUC) s(log(reves)) s(Police1)
## worst    0.9795673 0.9615685     0.9579137 0.05980130
## observed 0.9795673 0.8814061     0.9423738 0.03437499
## estimate 0.9795673 0.5379512     0.6187055 0.01396122
```

3. Tobit, Tobit Généralisé et Double Hurdle

```r
#Modèle Tobit
library(censReg)

modele_tobit <- censReg(Sinistre1 ~ log_reves + Nbadulte + Bauto + Acompm + agecat + Ahabi,
                       data = dat,
                       left = 0)

summary(modele_tobit)
```

```
##
## Call:
## censReg(formula = Sinistre1 ~ log_reves + Nbadulte + Bauto +
##     Acompm + agecat + Ahabi, left = 0, data = dat)
##
## Observations:
##          Total  Left-censored     Uncensored Right-censored
##           5352           4085           1267              0
##
## Coefficients:
##                                      Estimate Std. error t value  Pr(> t)
## (Intercept)                          -19.0711     7.8835  -2.419  0.01556 *
## log_reves                             -0.2503     0.8415  -0.297  0.76610
## Nbadulte                               1.6315     0.5922   2.755  0.00587 **
## BautoPas de vehicule                  -1.2084     1.6041  -0.753  0.45125
## AcompmCouple avec enfant(s)            0.3534     1.3152   0.269  0.78817
## AcompmCouple sans enfant              -0.3387     1.3998  -0.242  0.80879
```

```
## AcompmPersonne seule                    0.2482     1.8970   0.131  0.89589
## agecat41-50                            -2.5052     1.1476  -2.183  0.02903 *
## agecat51-60                            -3.9261     1.3618  -2.883  0.00394 **
## agecat61-96                            -7.8720     1.3223  -5.953 2.63e-09 ***
## AhabiParis + Agglomeration              3.1766     1.4290   2.223  0.02622 *
## AhabiUn. urb. de 10 000 a 99 999 hab.   2.6252     1.2082   2.173  0.02979 *
## AhabiUn. urb. de 100 000 hab. et +      5.2437     1.0769   4.869 1.12e-06 ***
## AhabiUn. urb. de 2 000 a 9 999 hab.     0.1177     1.4220   0.083  0.93403
## logSigma                                3.0423     0.0211 144.167  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Newton-Raphson maximisation, 6 iterations
## Return code 8: successive function values within relative tolerance limit (reltol)
## Log-likelihood: -7100.459 on 15 Df
```

```r
library(mhurdle)
# Tobit généralisé sans corrélation
tobit_gen_nocorr <- mhurdle(
  formula = Sinistre1 ~ log_reves + Nbadulte | log_reves + Bauto + Acompm,
  data = dat,
  dist = "ln",
  h2 = FALSE,
  corr = FALSE
)

summary(tobit_gen_nocorr)
```

```
##
## Call:
## mhurdle(formula = Sinistre1 ~ log_reves + Nbadulte | log_reves +
##     Bauto + Acompm, data = dat, dist = "ln", h2 = FALSE, corr = FALSE)
##
## Frequency of 0:  0.76327
##
## Coefficients :
##                                 Estimate Std. Error  t-value  Pr(>|t|)
## h1.(Intercept)                 -1.045322   0.337955  -3.0931  0.001981 **
## h1.log_reves                    0.020512   0.036939   0.5553  0.578692
## h1.Nbadulte                     0.056579   0.018297   3.0923  0.001986 **
## h2.(Intercept)                  0.673499   0.772711   0.8716  0.383423
## h2.log_reves                   -0.057075   0.081448  -0.7008  0.483456
## h2.BautoPas de vehicule        -0.134785   0.160207  -0.8413  0.400170
## h2.AcompmCouple avec enfant(s) -0.128501   0.097047  -1.3241  0.185468
## h2.AcompmCouple sans enfant    -0.225343   0.111410  -2.0226  0.043111 *
## h2.AcompmPersonne seule        -0.392735   0.138906  -2.8274  0.004693 **
## sd.sd                           1.421212   0.011413 124.5238 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -5166.5 on 10 Df
##
## R^2 :
##  Coefficient of determination : 1
```

```
##  Likelihood ratio index       : 0.0021746
```

```r
# Tobit généralisé avec corrélation entre les deux équations
tobit_gen_corr <- mhurdle(
  formula = Sinistre1 ~ log_reves + Nbadulte | log_reves + Bauto + Acompm,
  data = dat,
  dist = "ln",
  h2 = FALSE,
  corr = TRUE
)

summary(tobit_gen_corr)
```

```
##
## Call:
## mhurdle(formula = Sinistre1 ~ log_reves + Nbadulte | log_reves +
##     Bauto + Acompm, data = dat, dist = "ln", h2 = FALSE, corr = TRUE,
##     robust = TRUE)
##
## Frequency of 0:  0.76327
##
## Newton-Raphson maximisation method
## 0 iterations, 0h:0m:0s
## g'(-H)^-1g =    NA
##
##
## Coefficients :
##                               Estimate Std. Error t-value  Pr(>|t|)
## h1.(Intercept)               -1.038508   0.338049 -3.0721  0.002126 **
## h1.log_reves                  0.018916   0.036961  0.5118  0.608809
## h1.Nbadulte                   0.059902   0.018448  3.2471  0.001166 **
## h2.(Intercept)                1.096572   0.864739  1.2681  0.204764
## h2.log_reves                 -0.063912   0.082048 -0.7790  0.436004
## h2.BautoPas de vehicule      -0.131392   0.160211 -0.8201  0.412149
## h2.AcompmCouple avec enfant(s) -0.112337   0.098382 -1.1419  0.253516
## h2.AcompmCouple sans enfant  -0.207227   0.112850 -1.8363  0.066313 .
## h2.AcompmPersonne seule      -0.363205   0.141975 -2.5582  0.010521 *
## sd.sd                         1.442892   0.048879 29.5198 < 2.2e-16 ***
## corr12                       -0.197751   0.173704 -1.1384  0.254938
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -5166.1 on 11 Df
##
## R^2 :
##  Coefficient of determination : -0.01543
##  Likelihood ratio index       : 0.0022528
```

```r
# Modèle Double Hurdle pour sinistre1
modele_double_hurdle <- mhurdle(Sinistre1 ~ log_reves + Nbadulte + Bauto |
                               Acompm + agecat + Ahabi,
                             data = dat, dist = "ln", method = "bfgs")
```

```r
# Résumé du modèle
summary(modele_double_hurdle)
```

```
##
## Call:
## mhurdle(formula = Sinistre1 ~ log_reves + Nbadulte + Bauto |
##     Acompm + agecat + Ahabi, data = dat, dist = "ln", method = "bfgs")
##
## Frequency of 0:  0.76327
##
## Coefficients :
##                                             Estimate Std. Error  t-value
## h1.(Intercept)                            -0.9993400  0.3493024  -2.8610
## h1.log_reves                               0.0162854  0.0378252   0.4305
## h1.Nbadulte                                0.0551887  0.0184922   2.9844
## h1.BautoPas de vehicule                   -0.0381560  0.0729723  -0.5229
## h2.(Intercept)                             0.1264270  0.1293031   0.9778
## h2.AcompmCouple avec enfant(s)            -0.1866981  0.1133522  -1.6471
## h2.AcompmCouple sans enfant               -0.1244059  0.1210832  -1.0274
## h2.AcompmPersonne seule                   -0.3595271  0.1391315  -2.5841
## h2.agecat41-50                            -0.0089237  0.1093186  -0.0816
## h2.agecat51-60                            -0.0709352  0.1344967  -0.5274
## h2.agecat61-96                            -0.3294722  0.1338989  -2.4606
## h2.AhabiParis + Agglomeration              0.0018045  0.1381238   0.0131
## h2.AhabiUn. urb. de 10 000 a 99 999 hab. -0.0119104  0.1249678  -0.0953
## h2.AhabiUn. urb. de 100 000 hab. et +      0.2852013  0.1100112   2.5925
## h2.AhabiUn. urb. de 2 000 a 9 999 hab.    -0.2997492  0.1498250  -2.0007
## sd.sd                                      1.4058378  0.0115380 121.8442
##                                            Pr(>|t|)
## h1.(Intercept)                             0.004224 **
## h1.log_reves                               0.666801
## h1.Nbadulte                                0.002841 **
## h1.BautoPas de vehicule                    0.601055
## h2.(Intercept)                             0.328195
## h2.AcompmCouple avec enfant(s)             0.099545 .
## h2.AcompmCouple sans enfant                0.304213
## h2.AcompmPersonne seule                    0.009764 **
## h2.agecat41-50                             0.934941
## h2.agecat51-60                             0.597907
## h2.agecat61-96                             0.013870 *
## h2.AhabiParis + Agglomeration              0.989577
## h2.AhabiUn. urb. de 10 000 a 99 999 hab.  0.924071
## h2.AhabiUn. urb. de 100 000 hab. et +      0.009529 **
## h2.AhabiUn. urb. de 2 000 a 9 999 hab.     0.045429 *
## sd.sd                                     < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -5152.6 on 16 Df
##
## R^2 :
##  Coefficient of determination : 1
##  Likelihood ratio index       : 0.0048625
```

4. Nombre de Sinistres

```
sum(dat$NSin == 0) / nrow(dat)  # Proportion de zéros
```

```
## [1] 0.2600897
```

```
# Régression de Poisson
poisson_model <- glm(NSin ~ pcs + RUC + cs + log_reves + region + Ahabi +
                       agecat + Acompm + Nbadulte,
                     family = "poisson", data = dat)
summary(poisson_model)
```

```
##
## Call:
## glm(formula = NSin ~ pcs + RUC + cs + log_reves + region + Ahabi +
##     agecat + Acompm + Nbadulte, family = "poisson", data = dat)
##
## Coefficients:
##                                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)                             -8.881e-01  2.137e-01  -4.156 3.24e-05
## pcsArtisans, comm., chefs d'ent.         9.239e-02  5.830e-02   1.585 0.113046
## pcsAutres pers. sans activite prof.      2.575e-01  6.122e-02   4.205 2.61e-05
## pcsCadres et prof. intellectuelles sup.  2.683e-01  5.290e-02   5.072 3.94e-07
## pcsEmployes                              1.955e-01  5.009e-02   3.904 9.47e-05
## pcsOuvriers                              1.498e-01  4.816e-02   3.112 0.001860
## pcsProfessions intermediaires            2.375e-01  4.997e-02   4.753 2.01e-06
## pcsRetraites                             5.268e-02  5.994e-02   0.879 0.379508
## RUC                                     -3.250e-05  5.846e-06  -5.559 2.72e-08
## csModeste                                3.184e-02  6.484e-02   0.491 0.623441
## csMoyenne Inf                            2.336e-02  5.348e-02   0.437 0.662260
## csMoyenne Sup                           -7.472e-02  4.130e-02  -1.809 0.070403
## log_reves                                2.116e-01  2.224e-02   9.515  < 2e-16
## region2                                  5.549e-02  4.898e-02   1.133 0.257249
## region3                                 -3.435e-02  5.299e-02  -0.648 0.516806
## region4                                  1.131e-01  5.046e-02   2.242 0.024969
## region5                                 -4.179e-02  4.977e-02  -0.840 0.401032
## region7                                 -1.001e-02  5.144e-02  -0.195 0.845750
## region8                                  1.372e-01  4.963e-02   2.765 0.005701
## region9                                  2.485e-01  5.069e-02   4.902 9.48e-07
## AhabiParis + Agglomeration               9.943e-02  5.137e-02   1.936 0.052925
## AhabiUn. urb. de 10 000 a 99 999 hab.    4.145e-02  2.046e-02   2.025 0.042833
## AhabiUn. urb. de 100 000 hab. et +       7.184e-02  1.890e-02   3.801 0.000144
## AhabiUn. urb. de 2 000 a 9 999 hab.      6.877e-03  2.349e-02   0.293 0.769665
## agecat41-50                             -1.325e-01  1.944e-02  -6.816 9.35e-12
## agecat51-60                             -2.021e-01  2.490e-02  -8.115 4.85e-16
## agecat61-96                             -2.468e-01  4.205e-02  -5.869 4.39e-09
## AcompmCouple avec enfant(s)              1.979e-01  2.203e-02   8.984  < 2e-16
## AcompmCouple sans enfant                -1.857e-01  2.653e-02  -6.999 2.58e-12
## AcompmPersonne seule                    -6.092e-01  4.141e-02 -14.712  < 2e-16
## Nbadulte                                 1.668e-01  1.003e-02  16.625  < 2e-16
##
## (Intercept)                             ***
## pcsArtisans, comm., chefs d'ent.
```

```
## pcsAutres pers. sans activite prof.     ***
## pcsCadres et prof. intellectuelles sup. ***
## pcsEmployes                             ***
## pcsOuvriers                             **
## pcsProfessions intermediaires           ***
## pcsRetraites
## RUC                                     ***
## csModeste
## csMoyenne Inf
## csMoyenne Sup                           .
## log_reves                               ***
## region2
## region3
## region4                                 *
## region5
## region7
## region8                                 **
## region9                                 ***
## AhabiParis + Agglomeration              .
## AhabiUn. urb. de 10 000 a 99 999 hab.   *
## AhabiUn. urb. de 100 000 hab. et +      ***
## AhabiUn. urb. de 2 000 a 9 999 hab.
## agecat41-50                             ***
## agecat51-60                             ***
## agecat61-96                             ***
## AcompmCouple avec enfant(s)             ***
## AcompmCouple sans enfant                ***
## AcompmPersonne seule                    ***
## Nbadulte                                ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 20265  on 5351  degrees of freedom
## Residual deviance: 15360  on 5321  degrees of freedom
## AIC: 29217
##
## Number of Fisher Scoring iterations: 5
```

```r
# Test de surdispersion
dispersion_test <- poisson_model$deviance / poisson_model$df.residual
cat("Dispersion:", dispersion_test, "\n")
```

```
## Dispersion: 2.886612
```

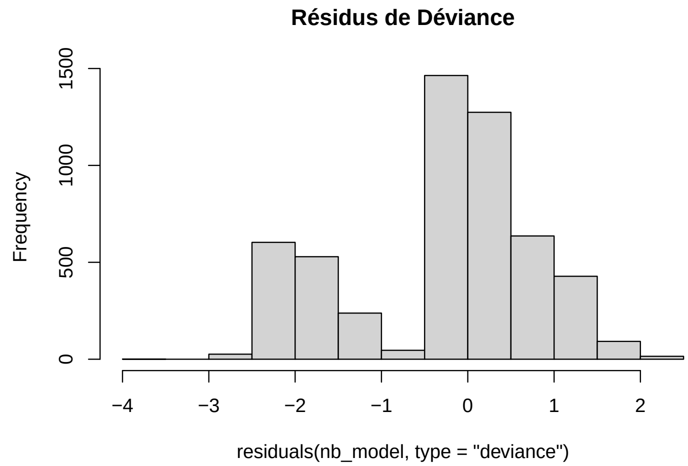
```r
# Régression Binomiale Négative

nb_model <- glm.nb(NSin ~ pcs + RUC + cs + log_reves + region + Ahabi +
                     agecat + Acompm + Nbadulte, data = dat)
summary(nb_model)
```

```
##
```

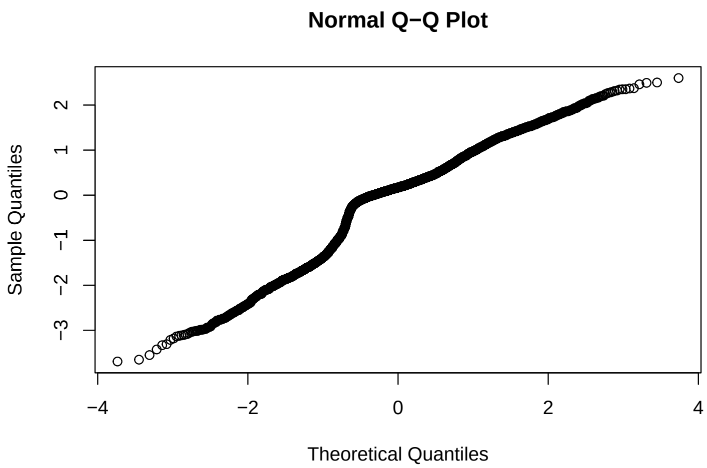
```
## Call:
## glm.nb(formula = NSin ~ pcs + RUC + cs + log_reves + region +
##     Ahabi + agecat + Acompm + Nbadulte, data = dat, init.theta = 2.025279052,
##     link = log)
##
## Coefficients:
##                                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)                            -2.047e+00  4.918e-01  -4.163 3.14e-05
## pcsArtisans, comm., chefs d'ent.        5.965e-02  1.045e-01   0.571 0.568190
## pcsAutres pers. sans activite prof.     2.464e-01  1.078e-01   2.286 0.022276
## pcsCadres et prof. intellectuelles sup.  2.090e-01  9.507e-02   2.199 0.027899
## pcsEmployes                             1.712e-01  8.943e-02   1.915 0.055515
## pcsOuvriers                             1.245e-01  8.581e-02   1.451 0.146737
## pcsProfessions intermediaires           1.996e-01  8.936e-02   2.234 0.025501
## pcsRetraites                            3.318e-02  1.027e-01   0.323 0.746642
## RUC                                    -4.087e-05  9.914e-06  -4.123 3.74e-05
## csModeste                               1.124e-01  1.085e-01   1.037 0.299907
## csMoyenne Inf                           5.533e-02  8.686e-02   0.637 0.524175
## csMoyenne Sup                          -5.703e-02  6.675e-02  -0.854 0.392895
## log_reves                               3.415e-01  5.394e-02   6.331 2.44e-10
## region2                                 5.681e-02  8.670e-02   0.655 0.512292
## region3                                -1.538e-02  9.399e-02  -0.164 0.870013
## region4                                 1.331e-01  8.990e-02   1.480 0.138898
## region5                                -1.698e-02  8.785e-02  -0.193 0.846741
## region7                                -2.229e-03  9.072e-02  -0.025 0.980395
## region8                                 1.615e-01  8.824e-02   1.830 0.067265
## region9                                 2.669e-01  9.046e-02   2.950 0.003173
## AhabiParis + Agglomeration              1.441e-01  9.081e-02   1.586 0.112626
## AhabiUn. urb. de 10 000 a 99 999 hab.   5.293e-02  3.711e-02   1.426 0.153796
## AhabiUn. urb. de 100 000 hab. et +      8.615e-02  3.434e-02   2.509 0.012119
## AhabiUn. urb. de 2 000 a 9 999 hab.     1.782e-02  4.229e-02   0.421 0.673559
## agecat41-50                            -1.330e-01  3.610e-02  -3.685 0.000229
## agecat51-60                            -2.189e-01  4.445e-02  -4.926 8.41e-07
## agecat61-96                            -2.628e-01  6.947e-02  -3.784 0.000155
## AcompmCouple avec enfant(s)             1.692e-01  4.246e-02   3.984 6.77e-05
## AcompmCouple sans enfant               -1.610e-01  4.568e-02  -3.526 0.000422
## AcompmPersonne seule                   -5.382e-01  6.849e-02  -7.858 3.92e-15
## Nbadulte                                1.512e-01  2.073e-02   7.296 2.96e-13
##
## (Intercept)                            ***
## pcsArtisans, comm., chefs d'ent.
## pcsAutres pers. sans activite prof.    *
## pcsCadres et prof. intellectuelles sup. *
## pcsEmployes                            .
## pcsOuvriers
## pcsProfessions intermediaires          *
## pcsRetraites
## RUC                                    ***
## csModeste
## csMoyenne Inf
## csMoyenne Sup
## log_reves                              ***
## region2
## region3
```

```
## region4
## region5
## region7
## region8                                 .
## region9                                 **
## AhabiParis + Agglomeration
## AhabiUn. urb. de 10 000 a 99 999 hab.
## AhabiUn. urb. de 100 000 hab. et +      *
## AhabiUn. urb. de 2 000 a 9 999 hab.
## agecat41-50                             ***
## agecat51-60                             ***
## agecat61-96                             ***
## AcompmCouple avec enfant(s)             ***
## AcompmCouple sans enfant                ***
## AcompmPersonne seule                    ***
## Nbadulte                                ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(2.0253) family taken to be 1)
##
##     Null deviance: 8575.5  on 5351  degrees of freedom
## Residual deviance: 6860.2  on 5321  degrees of freedom
## AIC: 25972
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  2.0253
##           Std. Err.:  0.0741
##
##  2 x log-likelihood:  -25907.7540
```

```r
# Histogramme des résidus de déviance
hist(residuals(nb_model, type = "deviance"), main = "Résidus de Déviance")
```



```r
# QQ-plot des résidus
qqnorm(qresid(nb_model))
```



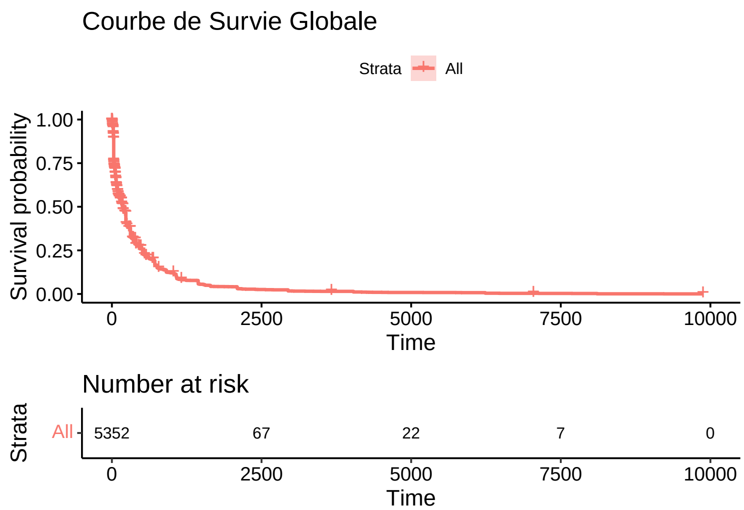

```r
# Validation Multico
vif(nb_model)
```

```
##                GVIF Df GVIF^(1/(2*Df))
## pcs        7.102602  7        1.150312
## RUC        8.182058  1        2.860430
## cs         8.583960  3        1.430918
## log_reves  5.446819  1        2.333842
## region     6.864900  7        1.147518
## Ahabi      7.037635  4        1.276228
## agecat     7.694233  3        1.405058
## Acompm     7.364517  3        1.394839
## Nbadulte   3.358009  1        1.832487
```

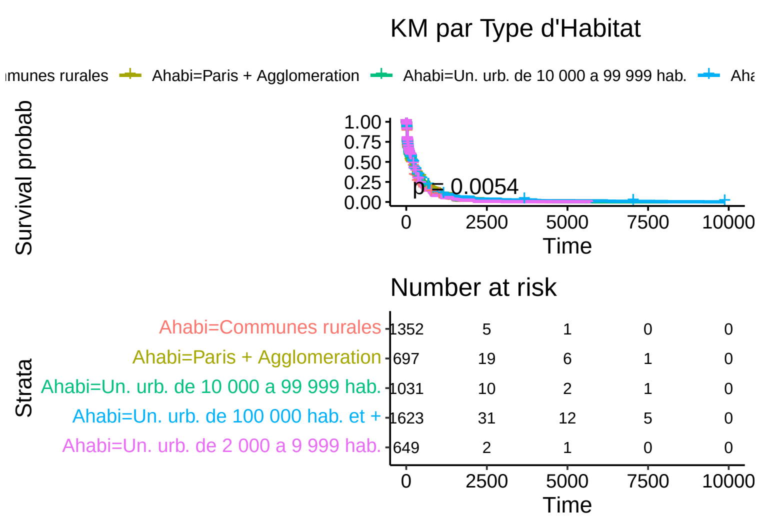
5. Modélisation de Durée

```r
# Estimateur de Kaplan-Meier (KM)

library(survival)
library(survminer)

# Objet de survie
surv_obj <- Surv(time = dat$Duree, event = dat$censure)

# KM global
km_fit <- survfit(surv_obj ~ 1)
ggsurvplot(
  km_fit,
  data = dat,
  risk.table = TRUE,
  title = "Courbe de Survie Globale",
  risk.table.height = 0.3,
```

```
  risk.table.fontsize = 3.5
)
```

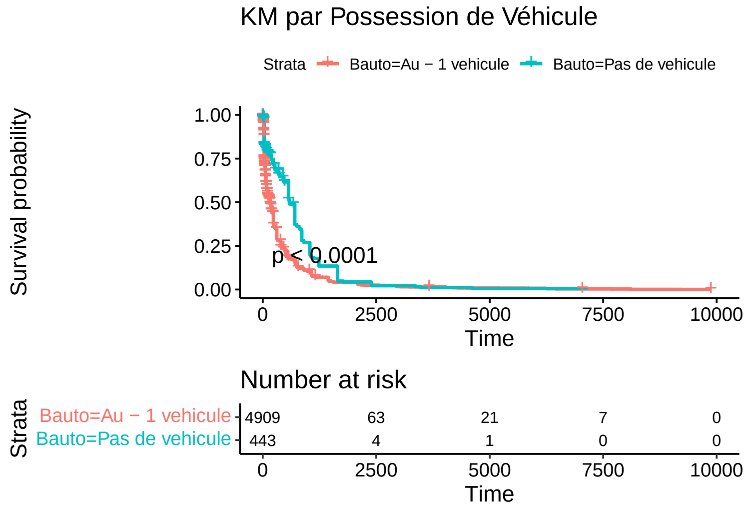



```
# KM par type d'habitat (Ahabi)
km_ahabi <- survfit(surv_obj ~ Ahabi, data = dat)
ggsurvplot(
  km_ahabi,
  data = dat,
  risk.table = TRUE,
  pval = TRUE,
  title = "KM par Type d'Habitat",
  risk.table.height = 0.5,
  risk.table.fontsize = 3.5
)
```

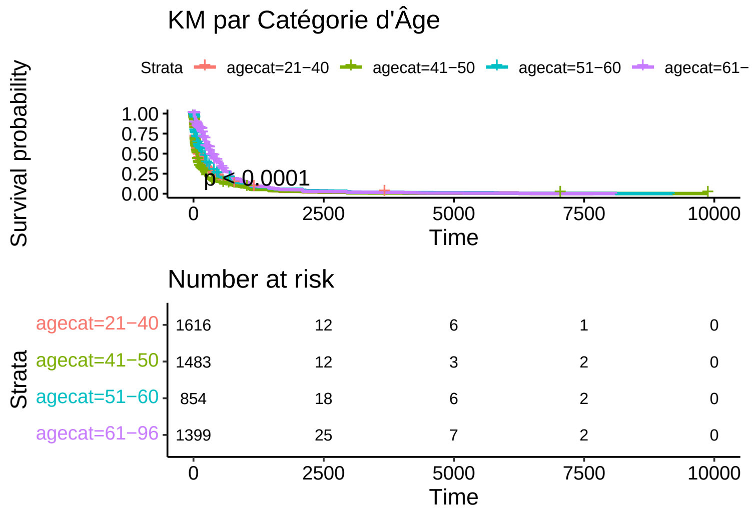



```
# KM par possession de véhicule (Bauto)
km_bauto <- survfit(surv_obj ~ Bauto, data = dat)
ggsurvplot(
```

```
  km_bauto,
  data = dat,
  risk.table = TRUE,
  pval = TRUE,
  title = "KM par Possession de Véhicule",
  risk.table.height = 0.3,
  risk.table.fontsize = 3.5
)
```




```
# KM par catégorie d'âge (agecat)
km_age <- survfit(surv_obj ~ agecat, data = dat)
ggsurvplot(
  km_age,
  data = dat,
  risk.table = TRUE,
  pval = TRUE,
  title = "KM par Catégorie d'Âge",
  risk.table.height = 0.5,
  risk.table.fontsize = 3.5
)
```




```r
# Comparaison des Courbes KM (Test du Log-Rank)
```

```r
# Test du log-rank pour Ahabi
survdiff(surv_obj ~ Ahabi, data = dat)
```

```
## Call:
## survdiff(formula = surv_obj ~ Ahabi, data = dat)
##
##                                         N Observed Expected (O-E)^2/E
## Ahabi=Communes rurales               1352      663      599     6.920
## Ahabi=Paris + Agglomeration           697      590      615     1.004
## Ahabi=Un. urb. de 10 000 a 99 999 hab. 1031      644      634     0.147
## Ahabi=Un. urb. de 100 000 hab. et +  1623     1061     1130     4.191
## Ahabi=Un. urb. de 2 000 a 9 999 hab.  649      349      329     1.172
##                                      (O-E)^2/V
## Ahabi=Communes rurales                   9.113
## Ahabi=Paris + Agglomeration              1.341
## Ahabi=Un. urb. de 10 000 a 99 999 hab.   0.194
## Ahabi=Un. urb. de 100 000 hab. et +      6.854
## Ahabi=Un. urb. de 2 000 a 9 999 hab.     1.392
##
##  Chisq= 14.7  on 4 degrees of freedom, p= 0.005
```

```r
# Test du log-rank pour Bauto
survdiff(surv_obj ~ Bauto, data = dat)
```

```
## Call:
## survdiff(formula = surv_obj ~ Bauto, data = dat)
##
##                          N Observed Expected (O-E)^2/E (O-E)^2/V
## Bauto=Au - 1 vehicule 4909     3065     2887      11.0        93
## Bauto=Pas de vehicule  443      242      420      75.5        93
##
##  Chisq= 93  on 1 degrees of freedom, p= <2e-16
```

```r
# Test du log-rank pour agecat
survdiff(surv_obj ~ agecat, data = dat)
```

```
## Call:
## survdiff(formula = surv_obj ~ agecat, data = dat)
##
##                  N Observed Expected (O-E)^2/E (O-E)^2/V
## agecat=21-40 1616      885      728    34.060     46.92
## agecat=41-50 1483      804      552   114.770    149.15
## agecat=51-60  854      591      607     0.411      0.54
## agecat=61-96 1399     1027     1420   108.949    208.40
##
##  Chisq= 283  on 3 degrees of freedom, p= <2e-16
```

```r
# Modèle 1 Cox avec Toutes les Variables
cox_model1 <- coxph(surv_obj ~ pcs + RUC + log_reves + region + Ahabi + agecat + Acompm + Nbadulte + Bau
summary(cox_model1)
```

```
## Call:
## coxph(formula = surv_obj ~ pcs + RUC + log_reves + region + Ahabi +
##     agecat + Acompm + Nbadulte + Bauto, data = dat)
##
##   n= 5352, number of events= 3307
##
##                                              coef  exp(coef)   se(coef)
## pcsArtisans, comm., chefs d'ent.        5.885e-01  1.801e+00  2.639e-01
## pcsAutres pers. sans activite prof.     1.078e-01  1.114e+00  2.725e-01
## pcsCadres et prof. intellectuelles sup. 3.537e-01  1.424e+00  2.517e-01
## pcsEmployes                             4.518e-01  1.571e+00  2.496e-01
## pcsOuvriers                             1.991e-01  1.220e+00  2.484e-01
## pcsProfessions intermediaires           5.318e-01  1.702e+00  2.486e-01
## pcsRetraites                            3.912e-01  1.479e+00  2.553e-01
## RUC                                    -1.499e-04  9.999e-01  6.827e-06
## log_reves                               7.000e-01  2.014e+00  3.168e-02
## region2                                -1.177e-01  8.890e-01  1.113e-01
## region3                                -4.479e-01  6.390e-01  1.269e-01
## region4                                -1.429e-01  8.668e-01  1.177e-01
## region5                                -3.418e-01  7.105e-01  1.143e-01
## region7                                -3.403e-01  7.115e-01  1.184e-01
## region8                                -1.867e-01  8.297e-01  1.136e-01
## region9                                -2.098e-01  8.108e-01  1.168e-01
## AhabiParis + Agglomeration             -3.100e-01  7.335e-01  1.157e-01
## AhabiUn. urb. de 10 000 a 99 999 hab.  -2.937e-02  9.711e-01  5.659e-02
## AhabiUn. urb. de 100 000 hab. et +     -1.931e-01  8.244e-01  5.282e-02
## AhabiUn. urb. de 2 000 a 9 999 hab.     2.103e-02  1.021e+00  6.684e-02
## agecat41-50                             2.516e-02  1.025e+00  5.684e-02
## agecat51-60                            -3.856e-02  9.622e-01  6.344e-02
## agecat61-96                            -1.191e-01  8.878e-01  9.380e-02
## AcompmCouple avec enfant(s)             6.594e-01  1.934e+00  6.420e-02
## AcompmCouple sans enfant               -4.046e-01  6.672e-01  6.076e-02
## AcompmPersonne seule                   -7.445e-01  4.750e-01  8.615e-02
## Nbadulte                                2.561e-01  1.292e+00  2.982e-02
## BautoPas de vehicule                   -1.282e-01  8.797e-01  7.313e-02
```

```
##                                               z Pr(>|z|)
## pcsArtisans, comm., chefs d'ent.          2.230 0.025778 *
## pcsAutres pers. sans activite prof.       0.396 0.692431
## pcsCadres et prof. intellectuelles sup.   1.406 0.159850
## pcsEmployes                               1.810 0.070261 .
## pcsOuvriers                               0.802 0.422755
## pcsProfessions intermediaires             2.140 0.032391 *
## pcsRetraites                              1.532 0.125449
## RUC                                     -21.963  < 2e-16 ***
## log_reves                                22.096  < 2e-16 ***
## region2                                  -1.058 0.290216
## region3                                  -3.529 0.000417 ***
## region4                                  -1.214 0.224601
## region5                                  -2.991 0.002784 **
## region7                                  -2.875 0.004041 **
## region8                                  -1.644 0.100270
## region9                                  -1.796 0.072442 .
## AhabiParis + Agglomeration               -2.679 0.007393 **
## AhabiUn. urb. de 10 000 a 99 999 hab.    -0.519 0.603808
## AhabiUn. urb. de 100 000 hab. et +       -3.656 0.000256 ***
## AhabiUn. urb. de 2 000 a 9 999 hab.       0.315 0.752984
## agecat41-50                               0.443 0.657966
## agecat51-60                              -0.608 0.543310
## agecat61-96                              -1.269 0.204342
## AcompmCouple avec enfant(s)              10.271  < 2e-16 ***
## AcompmCouple sans enfant                 -6.659 2.76e-11 ***
## AcompmPersonne seule                     -8.642  < 2e-16 ***
## Nbadulte                                  8.591  < 2e-16 ***
## BautoPas de vehicule                     -1.753 0.079533 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                                         exp(coef) exp(-coef) lower .95
## pcsArtisans, comm., chefs d'ent.           1.8012     0.5552    1.0737
## pcsAutres pers. sans activite prof.        1.1138     0.8978    0.6530
## pcsCadres et prof. intellectuelles sup.    1.4244     0.7021    0.8698
## pcsEmployes                                1.5711     0.6365    0.9633
## pcsOuvriers                                1.2203     0.8195    0.7500
## pcsProfessions intermediaires              1.7020     0.5875    1.0456
## pcsRetraites                               1.4788     0.6762    0.8966
## RUC                                        0.9999     1.0001    0.9998
## log_reves                                  2.0137     0.4966    1.8925
## region2                                    0.8890     1.1249    0.7148
## region3                                    0.6390     1.5650    0.4983
## region4                                    0.8668     1.1537    0.6882
## region5                                    0.7105     1.4075    0.5679
## region7                                    0.7115     1.4054    0.5642
## region8                                    0.8297     1.2052    0.6641
## region9                                    0.8108     1.2334    0.6449
## AhabiParis + Agglomeration                 0.7335     1.3634    0.5846
## AhabiUn. urb. de 10 000 a 99 999 hab.      0.9711     1.0298    0.8691
## AhabiUn. urb. de 100 000 hab. et +         0.8244     1.2130    0.7433
## AhabiUn. urb. de 2 000 a 9 999 hab.        1.0213     0.9792    0.8959
## agecat41-50                                1.0255     0.9751    0.9174
```

```
## agecat51-60                               0.9622     1.0393    0.8497
## agecat61-96                               0.8878     1.1264    0.7387
## AcompmCouple avec enfant(s)               1.9335     0.5172    1.7049
## AcompmCouple sans enfant                  0.6672     1.4987    0.5923
## AcompmPersonne seule                      0.4750     2.1054    0.4012
## Nbadulte                                  1.2919     0.7740    1.2186
## BautoPas de vehicule                      0.8797     1.1368    0.7622
##                                        upper .95
## pcsArtisans, comm., chefs d'ent.          3.0216
## pcsAutres pers. sans activite prof.       1.8999
## pcsCadres et prof. intellectuelles sup.   2.3327
## pcsEmployes                               2.5623
## pcsOuvriers                               1.9857
## pcsProfessions intermediaires             2.7703
## pcsRetraites                              2.4391
## RUC                                       0.9999
## log_reves                                 2.1427
## region2                                   1.1056
## region3                                   0.8194
## region4                                   1.0917
## region5                                   0.8889
## region7                                   0.8973
## region8                                   1.0366
## region9                                   1.0193
## AhabiParis + Agglomeration                0.9202
## AhabiUn. urb. de 10 000 a 99 999 hab.     1.0850
## AhabiUn. urb. de 100 000 hab. et +        0.9143
## AhabiUn. urb. de 2 000 a 9 999 hab.       1.1642
## agecat41-50                               1.1463
## agecat51-60                               1.0896
## agecat61-96                               1.0669
## AcompmCouple avec enfant(s)               2.1928
## AcompmCouple sans enfant                  0.7516
## AcompmPersonne seule                      0.5623
## Nbadulte                                  1.3697
## BautoPas de vehicule                      1.0152
##
## Concordance= 0.769  (se = 0.006 )
## Likelihood ratio test= 2321  on 28 df,   p=<2e-16
## Wald test            = 2553  on 28 df,   p=<2e-16
## Score (logrank) test = 2620  on 28 df,   p=<2e-16
```

```r
cat("Concordance pour le Modèle 1 :", cox_model1$concordance[1], "\n")
```

```
## Concordance pour le Modèle 1 : 5171046
```

```r
# Modèle 2 Cox avec Variables Sélectionnées
cox_model2 <- coxph(surv_obj ~ Ahabi + Bauto + agecat + pcs + log_reves, data = dat)
summary(cox_model2)
```

```
## Call:
## coxph(formula = surv_obj ~ Ahabi + Bauto + agecat + pcs + log_reves,
##     data = dat)
```

```
##
##   n= 5352, number of events= 3307
##
##                                            coef exp(coef) se(coef)      z
## AhabiParis + Agglomeration              -0.28159   0.75458  0.06041 -4.661
## AhabiUn. urb. de 10 000 a 99 999 hab.   -0.09832   0.90636  0.05598 -1.756
## AhabiUn. urb. de 100 000 hab. et +      -0.22334   0.79985  0.05073 -4.403
## AhabiUn. urb. de 2 000 a 9 999 hab.     -0.02429   0.97600  0.06634 -0.366
## BautoPas de vehicule                    -0.26736   0.76540  0.07111 -3.760
## agecat41-50                              0.11042   1.11675  0.04957  2.227
## agecat51-60                             -0.31423   0.73035  0.05579 -5.632
## agecat61-96                             -0.57388   0.56334  0.08809 -6.515
## pcsArtisans, comm., chefs d'ent.         0.27088   1.31111  0.26376  1.027
## pcsAutres pers. sans activite prof.     -0.28944   0.74869  0.27175 -1.065
## pcsCadres et prof. intellectuelles sup. -0.26699   0.76568  0.25113 -1.063
## pcsEmployes                              0.04200   1.04290  0.24920  0.169
## pcsOuvriers                              0.15859   1.17186  0.24796  0.640
## pcsProfessions intermediaires            0.01377   1.01387  0.24825  0.055
## pcsRetraites                             0.03801   1.03874  0.25401  0.150
## log_reves                                0.48041   1.61673  0.03067 15.663
##                                         Pr(>|z|)
## AhabiParis + Agglomeration              3.14e-06 ***
## AhabiUn. urb. de 10 000 a 99 999 hab.    0.07903 .
## AhabiUn. urb. de 100 000 hab. et +      1.07e-05 ***
## AhabiUn. urb. de 2 000 a 9 999 hab.      0.71426
## BautoPas de vehicule                     0.00017 ***
## agecat41-50                              0.02592 *
## agecat51-60                             1.78e-08 ***
## agecat61-96                             7.29e-11 ***
## pcsArtisans, comm., chefs d'ent.         0.30443
## pcsAutres pers. sans activite prof.      0.28684
## pcsCadres et prof. intellectuelles sup.  0.28771
## pcsEmployes                              0.86615
## pcsOuvriers                              0.52244
## pcsProfessions intermediaires            0.95577
## pcsRetraites                             0.88106
## log_reves                                < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                                         exp(coef) exp(-coef) lower .95
## AhabiParis + Agglomeration                 0.7546     1.3252    0.6703
## AhabiUn. urb. de 10 000 a 99 999 hab.      0.9064     1.1033    0.8122
## AhabiUn. urb. de 100 000 hab. et +         0.7998     1.2502    0.7241
## AhabiUn. urb. de 2 000 a 9 999 hab.        0.9760     1.0246    0.8570
## BautoPas de vehicule                       0.7654     1.3065    0.6658
## agecat41-50                                1.1167     0.8955    1.0133
## agecat51-60                                0.7303     1.3692    0.6547
## agecat61-96                                0.5633     1.7751    0.4740
## pcsArtisans, comm., chefs d'ent.           1.3111     0.7627    0.7819
## pcsAutres pers. sans activite prof.        0.7487     1.3357    0.4395
## pcsCadres et prof. intellectuelles sup.    0.7657     1.3060    0.4680
## pcsEmployes                                1.0429     0.9589    0.6399
## pcsOuvriers                                1.1719     0.8533    0.7208
```

```
## pcsProfessions intermediaires              1.0139     0.9863    0.6233
## pcsRetraites                               1.0387     0.9627    0.6314
## log_reves                                  1.6167     0.6185    1.5224
##                                          upper .95
## AhabiParis + Agglomeration                 0.8494
## AhabiUn. urb. de 10 000 a 99 999 hab.      1.0115
## AhabiUn. urb. de 100 000 hab. et +         0.8835
## AhabiUn. urb. de 2 000 a 9 999 hab.        1.1115
## BautoPas de vehicule                       0.8799
## agecat41-50                                1.2307
## agecat51-60                                0.8147
## agecat61-96                                0.6695
## pcsArtisans, comm., chefs d'ent.           2.1986
## pcsAutres pers. sans activite prof.        1.2753
## pcsCadres et prof. intellectuelles sup.    1.2526
## pcsEmployes                                1.6996
## pcsOuvriers                                1.9052
## pcsProfessions intermediaires              1.6493
## pcsRetraites                               1.7089
## log_reves                                  1.7169
##
## Concordance= 0.678  (se = 0.005 )
## Likelihood ratio test= 608.9  on 16 df,   p=<2e-16
## Wald test            = 610  on 16 df,   p=<2e-16
## Score (logrank) test = 607.3  on 16 df,   p=<2e-16
```

```
cat("Concordance pour le Modèle 2 :", cox_model2$concordance[1], "\n")
```

```
## Concordance pour le Modèle 2 : 4550783
```

```
# Modèle 3 Cox avec Variables de Base
cox_model3 <- coxph(surv_obj ~ Ahabi + Bauto + agecat, data = dat)
summary(cox_model3)
```

```
## Call:
## coxph(formula = surv_obj ~ Ahabi + Bauto + agecat, data = dat)
##
##   n= 5352, number of events= 3307
##
##                                          coef exp(coef) se(coef)       z
## AhabiParis + Agglomeration           -0.14825   0.86221  0.05828  -2.544
## AhabiUn. urb. de 10 000 a 99 999 hab. -0.11910   0.88772  0.05561  -2.142
## AhabiUn. urb. de 100 000 hab. et +    -0.23274   0.79236  0.05037  -4.620
## AhabiUn. urb. de 2 000 a 9 999 hab.   -0.02783   0.97255  0.06616  -0.421
## BautoPas de vehicule                  -0.54984   0.57704  0.06829  -8.051
## agecat41-50                            0.17599   1.19243  0.04896   3.595
## agecat51-60                           -0.25220   0.77709  0.05340  -4.723
## agecat61-96                           -0.53232   0.58724  0.04683 -11.368
##                                       Pr(>|z|)
## AhabiParis + Agglomeration            0.010970 *
## AhabiUn. urb. de 10 000 a 99 999 hab. 0.032217 *
## AhabiUn. urb. de 100 000 hab. et +    3.83e-06 ***
## AhabiUn. urb. de 2 000 a 9 999 hab.   0.673999
```

```
## BautoPas de vehicule                    8.18e-16 ***
## agecat41-50                             0.000325 ***
## agecat51-60                             2.33e-06 ***
## agecat61-96                              < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                                         exp(coef) exp(-coef) lower .95 upper .95
## AhabiParis + Agglomeration                 0.8622     1.1598    0.7691    0.9666
## AhabiUn. urb. de 10 000 a 99 999 hab.      0.8877     1.1265    0.7961    0.9899
## AhabiUn. urb. de 100 000 hab. et +         0.7924     1.2620    0.7179    0.8746
## AhabiUn. urb. de 2 000 a 9 999 hab.        0.9726     1.0282    0.8543    1.1072
## BautoPas de vehicule                       0.5770     1.7330    0.5048    0.6597
## agecat41-50                                1.1924     0.8386    1.0833    1.3125
## agecat51-60                                0.7771     1.2868    0.6999    0.8628
## agecat61-96                                0.5872     1.7029    0.5357    0.6437
##
## Concordance= 0.638  (se = 0.006 )
## Likelihood ratio test= 369.6  on 8 df,   p=<2e-16
## Wald test            = 349.9  on 8 df,   p=<2e-16
## Score (logrank) test = 360  on 8 df,   p=<2e-16
```

```r
cat("Concordance pour le Modèle 3 :", cox_model3$concordance[1], "\n")
```

```
## Concordance pour le Modèle 3 : 4127257
```

```r
#Test de proportionnalité des risques

cox.zph(cox_model1)
```

```
##            chisq df       p
## pcs        201.2  7 < 2e-16
## RUC         25.3  1 5.0e-07
## log_reves   86.4  1 < 2e-16
## region      43.1  7 3.2e-07
## Ahabi       49.8  4 4.1e-10
## agecat     136.8  3 < 2e-16
## Acompm     105.0  3 < 2e-16
## Nbadulte    12.0  1 0.00053
## Bauto        5.7  1 0.01692
## GLOBAL     449.1 28 < 2e-16
```

```r
cox.zph(cox_model2)
```

```
##            chisq df       p
## Ahabi       22.0  4 0.00020
## Bauto       14.2  1 0.00016
## agecat     174.4  3 < 2e-16
## pcs        251.8  7 < 2e-16
## log_reves  195.0  1 < 2e-16
## GLOBAL     431.3 16 < 2e-16
```

```r
cox.zph(cox_model3)
```

```
##         chisq df       p
## Ahabi    26.6  4 2.4e-05
## Bauto    18.3  1 1.9e-05
## agecat 223.4  3 < 2e-16
## GLOBAL 253.9  8 < 2e-16
```