Analyse de données médico-administratives avec R: Exploration des données OpenCCAM de Scan Santé

Nolwenn Le Meur - EHESP

Sequence 1 - Mai 2023

Vous vous intéressez aux procédures qui peuvent être effectués en ambulatoire ou en chirurgie conventionnelle.

Vous disposez d'un premier d'un jeu données qui liste par établissement de soins les actes réalisés en France en 2020, le nombre d'actes, la DMS, l'activité ambulatoire, le département et la région. Ces données sont issues de ScanSante et librement disponibles sur https://www.scansante.fr/open-ccam/open-ccam-2020

I. Lecture et nettoyage des données

Dans un premier temps, il vous est demandé de lire les données, de vérifier leur qualité, et de sélectionner votre population d'intérêt.

a. Lire le fichier *"Open_ccam_20.csv"* avec la fonction *read.csv()* dans R pour créer l'objet R *ccam20* et vérifiez le type des variables. Attention les variables sont séparées pas des ";" et les données "manquantes" sont notées "." .

```
## Lecture des fichiers
ccam20 <- read.csv("Open_ccam_20.csv", header=T, sep=";", na.strings = ".", dec=",")
## avec la library data.table
# library(data.table)
# ccam20 <- fread("Open_ccam_20.csv")
```

b. Utliser la fonction skim() de la librarie skimr pour un premier diagnostic.

```
library(skimr)
skim(ccam20)
```

Table 1: Data summary

Name	ccam20
Number of rows	313491
Number of columns	10
Column type frequency:	
character	4
numeric	6
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
finess	0	1	8	9	0	1293	0
FinessGeo	0	1	8	9	0	1501	0
acte	0	1	8	8	0	4914	0
dep	0	1	1	2	0	102	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
nb_sejsea	0	1.00	167.97	713.37	11	18.00	35.00	97.00	44413.0	
nb_actes	0	1.00	224.59	1023.59	11	19.00	37.00	108.00	50068.0	
dms_globale	0	1.00	6.30	7.60	0	0.76	3.96	9.19	116.7	
nb_sej_0_nuit	173871	0.45	160.80	827.66	11	17.00	32.00	83.00	44237.0	
nb_acte_ambu	173871	0.45	164.17	831.68	11	18.00	33.00	85.00	44237.0	
reg	0	1.00	51.54	29.22	1	27.00	52.00	76.00	93.0	

c. Que font les lignes de codes ci-dessous? Executez pas à pas les lignes pour mieux les comprendre.

```
## region et département
ccam20$reg <- as.factor(ccam20$reg)
ccam20$dep <- as.factor(ccam20$dep)
# acte
ccam20$acte <- substr(ccam20$acte, 1, 7)
ccam20$acte <- as.factor(ccam20$acte)

# FINESS
ccam20$FinessGeo<- ifelse(nchar(ccam20$FinessGeo)==8,
                          paste("0",ccam20$FinessGeo,sep=""), ccam20$FinessGeo)
```

Vous souhaitez faire le focus sur un groupe de procédures. Nous vous proposons de travailler sur les actes endovasculaires mais vous pouvez utiliser votre propre liste d'actes réalisés en ambulatoire et en conventionnel.

d. Vous devez pour ce faire lire le fichier "liste_ccam2020.csv" (séparateur de colonnes ",") qui comporte les actes endovasculaires d'intérêt.

```
# lecture de codes CCAM pour les actes endovasculaires d'intérêt
ccam_endo <- read.csv("liste_ccam2020.csv", sep=",")
```

e. Puis vous devez sélectionner dans l'objet *ccam20* les lignes correspondant aux actes endovasculaires que vous venez de lire pour créer une sous table *endo*. Executez la ligne de commande ci-dessous pour comprendre le rôle de la function *%in%*.

```
# Sélection des actes endovasculaires dans la base ccam20
endo <- ccam20[ccam20$acte%in%ccam_endo$CDC_ACT, ]
```

f. Essayez de trouver une manière de faire la même chose avec la fonction filter de librairie *dplyr*

```
# endo <- ccam20[ccam20$acte%in%ccam_endo[,1], ]

## Avec la fonction subset
# endo <- subset(ccam20, ccam20$acte%in%ccam_endo$CDC_ACT)

## Avec la fonction filter de la librairie dplyr
```

```
library(dplyr)
endo <- ccam20 %>% filter(acte%in%ccam_endo$CDC_ACT)
```

g. Sauvegarder cette nouvelle base de donnée au format *.Rdata avec la fonction save()

```
save(endo, file="endovasculaire_20.Rdata" )
```

II. Statistiques descriptives

Pour l'exercice je vous invite à suavegarder votre script R et fermer votre application pour la réouvrir. Vous redémarrez ainsi comme après un long week-end…

Votre base de données *endo* n'est sans doute pas chargée dans votre environemment, vous devez l'importer avec la fonction *load()*.

```
load("endovasculaire_20.Rdata")
```

1. Quels établissements sont au-dessus de la DMS nationale pour les actes d'intérêts?

a. Calculez les indicateurs statistiques usuels pour la DMS en utilisant la fonction *summary()* et la fonction *ci()* de la library *epiDisplay*. Commentez les résultats.

```
# Affiche d'un premier niveau de résumé statistique
summary(endo$dms_globale)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   3.022   4.704   5.875   7.548  70.769
```

```
# charger la library epiDisplay et ses fonctions
library(epiDisplay)
# calcul de intervalle de confiance sur la DMS
endoci <- ci(endo$dms_globale)
endoci
```

```
##     n     mean       sd        se lower95ci upper95ci
##  1806 5.874781 4.576003 0.1076781  5.663594  6.085967
```

b. Calculez les indicateurs statistiques usuels pour la dms par région en utilisant la fonction *by()* ou la syntaxe de *dplyr*.

```
# utilisation de la fonction by pour l'affichage de résumés stat par région
by(endo$dms_globale, endo$reg, summary)
```

```
## endo$reg: 1
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.806   2.000   3.047   5.083   9.206  13.912
## ------------------------------------------------------------
## endo$reg: 4
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.150   3.527   6.487   6.636   8.959  20.676
## ------------------------------------------------------------
## endo$reg: 11
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.3333  3.0822  5.3844  7.4579  9.2252 70.7692
## ------------------------------------------------------------
## endo$reg: 24
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.438   3.923   5.242   5.855   7.286  14.417
```

```
## ------------------------------------------------------------
## endo$reg: 27
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.5556  3.3667  5.6087  6.4830  8.7436 20.2143
## ------------------------------------------------------------
## endo$reg: 28
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   2.292   3.467   4.418   6.142  15.062
## ------------------------------------------------------------
## endo$reg: 32
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.045   3.253   4.967   5.657   7.292  16.500
## ------------------------------------------------------------
## endo$reg: 44
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.7308  3.3036  5.5258  6.3272  8.5402 18.6667
## ------------------------------------------------------------
## endo$reg: 52
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.3333  2.4365  4.1333  4.9319  6.3967 17.2727
## ------------------------------------------------------------
## endo$reg: 53
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.2308  2.6894  3.8462  4.5439  5.8421 12.8718
## ------------------------------------------------------------
## endo$reg: 75
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.8235  2.9143  4.4545  5.4661  7.0741 18.5000
## ------------------------------------------------------------
## endo$reg: 76
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.364   3.161   4.783   5.982   7.552  33.105
## ------------------------------------------------------------
## endo$reg: 84
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.09091  3.03399  4.88861  6.05982  8.34286 25.45454
## ------------------------------------------------------------
## endo$reg: 93
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.8235  3.0833  4.3782  5.4682  6.3695 29.5000
```

```
# avec dplyr
# endo %>% group_by(reg) %>% summarise("Moyenne"= mean(dms_globale, na.rm=T))
```

d. Observez l'utilisation de la fonction *ifelse()* ci-dessous. Que fait cette ligne de commmande.

```
endo$Top <- ifelse(endo$dms_globale > endoci$upper95ci, 1, 0)
```

```
#endo$Top <- ifelse(endo$dms_globale > endoci$upper95ci, 1, 0)
#créez une variable binaire *Top* dans le table *endo* avec la modalité 1 lorsque #l'établissement a un

# Autre méthode par indexation
#endo$Top <- 0
#endo$Top[endo$dms_globale > endoci$upper95ci] <- 1
```

```
## Si pas accès à epiDisplay et utilisation de la moyenne des DMS
# endo$Top <- ifelse(endo$dms_globale > mean(endo$dms_globale, na.rm=TRUE), 1, 0)
## ou
# endo$Top[endo$dms_globale > mean(endo$dms_globale, na.rm=T)] <- 1
## ou
# endo[endo$dms_globale > mean(endo$dms_globale, na.rm=T), "Top"] <- 1
```

e. Avec la function *table()* créer la table de contingence qui dénombre les établissements au-dessus la borne supérieure de l'intervalle de confiance de la DMS nationale pour 1 acte dans chaque région.

```
## Si 1 acte - 1 hôpital par rapport à la dms nationale
table(endo$Top, endo$reg)
```

```
##
##       1   4  11  24  27  28  32  44  52  53  75  76  84  93
##   0   9  12 133  33  42  72 125  79  74  65 142  94 146 134
##   1   4  12 111  20  31  27  75  57  29  20  67  59  84  50
```

```
## Si n actes par hôpital
meanbyfiness <- endo %>% group_by(FinessGeo, reg) %>%
  summarise("mean_etab"= mean(dms_globale, na.rm=T))
```

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

```
## ou on converse la table d'origine et ajoute d'un colonne mean_etab
## où la moyenne est répété pour chaque établissement
meanbyfiness2 <- endo %>% group_by(FinessGeo) %>%
  mutate("mean_etab"= mean(dms_globale, na.rm=T))
## si que des NAs dans dms_globale pour 1 hôpital
# meanbyfiness <- meanbyfiness[meanbyfiness$mean!="NaN",]

# ligne unique par établissement unique
# library base R (default) dmsFiness2 <- unique(meanbyfiness2[,c("FinessGeo", "reg", "mean_etab")])
# méthode dplyr
dmsFiness <- distinct(meanbyfiness2[,c("FinessGeo", "reg","mean_etab")])

# nouveau calcul de la dms national et de l'intervalle de confiance
cinat<- ci(meanbyfiness$mean_etab)
# nouvelle variable TOP
meanbyfiness$Top <- ifelse(meanbyfiness$mean_etab > cinat$upper95ci, 1, 0)
table(meanbyfiness$Top, meanbyfiness$reg)
```

```
##
##      1  4 11 24 27 28 32 44 52 53 75 76 84 93
##   0  2  2 30  7  8 13 23 14 15 10 26 20 32 29
##   1  1  3 22  3  5  4  9 11  4  3  9 10 14  8
```

f. Utilisez la fonction *prop.table()* pour obtenir la proportion de ces établissements par région.

```
## proportion par région
round(prop.table(table(endo$Top, endo$reg), margin = 2)*100,2)
```

```
##
##         1     4    11    24    27    28    32    44    52    53    75    76
##   0 69.23 50.00 54.51 62.26 57.53 72.73 62.50 58.09 71.84 76.47 67.94 61.44
##   1 30.77 50.00 45.49 37.74 42.47 27.27 37.50 41.91 28.16 23.53 32.06 38.56
```

```
##
##        84    93
##   0 63.48 72.83
##   1 36.52 27.17
```

2. Quel est le mode de financement des établissements qui dépassent la DMS nationale pour les actes endovasculaires?

Pour obtenir le mode de financement des hopitaux (mft - mode fixation tarifs) nous devons apparier nos données CCAM aux données du répertoire Finess. Ces informations sont librement disponibles sur le site Open-data.gov.

(source: https://www.data.gouv.fr/fr/datasets/finess-extraction-du-fichier-des-etablissements/)

a. Lire le fichier *"etalab_stock_et_20201231.csv"* dans R pour créer l'objet R *finess*

b. Vérifier le type de la variable *mft* qui code le mode de financement des établissements.

c. Vérifier la longueur des codes *FinessGeo* dans les 2 bases à fusionner

d. Apparier les données CCAM pour les actes endovasculaires aux données Finess (variables *nofinesset* et *mft* uniquement) dans un objet nommé *endofiness* grâce à la fonction *merge()*.

```
## Jointure entre base avec clé de jointure de nom différent
## Si les 2 clés de jointures ont le même nom pas besoin de by.x et by.y mais juste by
endofiness <- merge(endo, finess[, c("nofinesset", "mft")], by.x="FinessGeo",by.y="nofinesset", all.x=TR
## colnames(endofiness)
```

e. Trouver une autre façon de le faire avec la librarie dplyr et sa syntaxe

```
## Avec dplyr
endofiness <- left_join(endo, finess[, c("nofinesset", "mft")],
                        by=c("FinessGeo"=="nofinesset"))
```

f. Que fait le code chunk suivant?

```
# useNA = "always pour visualiser si il y a de NA
table(endofiness$mft, useNA = "always")
levels(endofiness$mft)
endofiness$mft <- droplevels(endofiness$mft)
table(endofiness$mft, useNA = "always")
endofiness$mft <- factor(endofiness$mft,
                             labels=c("public", "public","non lucratif", "privé",
                                      "non lucratif", "privé",
                                      "public","privé","indéterminé"))
```

```
# Table de contingence de la variable mode de paiement
# Si table de contingence retourne des modalités à 0 on les retire en les forçant à NA
# Table de contingence de la variable mode de paiement sans les 0
# Ajouter les labels (étiquettes) à la variable mft
```

g. Utilisez les fonctions *table* et *prop.table()* pour obtenir le dénombrement et la proportion des établissements au-dessus de la DMS nationale (variable *Top*) par type de financements (variable *mft*).

```
## Proportion avec total colonne=100
prop.table(table(endofiness$Top, endofiness$mft), 2)*100
```

```
##
##       public non lucratif    privé indéterminé
##   0 39.92042     81.83613 77.14286    75.00000
```

```
##   1 60.07958     18.16387 22.85714    25.00000
```

```
## Proportion avec total ligne=100
prop.table(table(endofiness$Top, endofiness$mft), 1)*100
```

```
##
##         public non lucratif      privé indéterminé
##   0 25.9482759   71.4655172  2.3275862   0.2586207
##   1 70.1238390   28.4829721  1.2383901   0.1547988
```

3. Est-ce que les établissements dépassant la dms nationale pour les actes endovasculaires proposent aussi cette intervention en ambulatoire?

a. Créez une variable binaire *Ambu* dans le table *endofiness* avec la modalité 1 lorsque l'établissement présente des informations dans la variable *nb_sej_0_nuit* et 0 autrement.

```
# Création de vecteur de 0
endofiness$Ambu <- 0
# Remplacement des 0 par 1 si dans colonne nb_sej_0_nuit j'ai une valeur (pas de na !is.na() )
endofiness$Ambu[!is.na(endofiness$nb_sej_0_nuit)] <- 1

## Si je n'ai pas de NA ( !is.na() ) alors 1 sinon 0
# endofiness$Ambu <- ifelse(!is.na(endofiness$nb_sej_0_nuit), 1, 0)
## Si j'ai des NA alors 0 sinon 1 (avec étiquette)
# endofiness$Ambu <- ifelse(is.na(endofiness$nb_sej_0_nuit), "pas ambu", "ambu")
#endofiness$Ambu <- as.factor(endofiness$Ambu)
```

b. Utilisez les fonctions *table* et *prop.table()* pour obtenir le dénombrement et la proportion des établissements au-dessus de la DMS nationale pour 1 acte (variable *Top*) selon la présence ou nom d'une activité ambulatoire pour l'acte.

```
endofiness$Top <- factor(endofiness$Top, levels=c(0,1),
                         labels=c("dms", "sup dms"))
tabTopAmbu <- table("Dépassement"=endofiness$Top,
                    "Ambulatoire"=endofiness$Ambu)
tabTopAmbu
```

```
##            Ambulatoire
## Dépassement    0    1
##     dms     1011  149
##     sup dms  619   27
```

```
# 1 tiers des établissements qui ne font pas de l'ambulatoire
# depasse la DMS pour au moins 1 acte contre 15% pour les autres
prop.table(tabTopAmbu, 2)*100
```

```
##            Ambulatoire
## Dépassement        0        1
##     dms     62.02454 84.65909
##     sup dms 37.97546 15.34091
```

III. Statistiques inferentielles (optionel)

1. Analyses univariées

a. Réalisez un test de Chi2 ou un test Exact de Fisher pour tester l'hypothèse d'indépendence entre dépassement de la DMS nationale et la pratique de cette intervention en ambulatoire.

```r
chisq.test(tabTopAmbu)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tabTopAmbu
## X-squared = 34.444, df = 1, p-value = 4.388e-09
```

```r
chisq.test(endofiness$Top, endofiness$Ambu)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  endofiness$Top and endofiness$Ambu
## X-squared = 34.444, df = 1, p-value = 4.388e-09
```

```r
fisher.test(tabTopAmbu)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  tabTopAmbu
## p-value = 4.749e-10
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.1865430 0.4547102
## sample estimates:
## odds ratio
##  0.2961341
```

```r
fisher.test(endofiness$Top, endofiness$Ambu)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  endofiness$Top and endofiness$Ambu
## p-value = 4.749e-10
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.1865430 0.4547102
## sample estimates:
## odds ratio
##  0.2961341
```

Vous vous demandez si la DMS des établissements est fonction de l'activité soit la variable *nb_actes*?

b. Créez la variable binaire *nbactes_eleve* sur la base de la médiane de la distribution des *nb_actes* des établissements.

```r
summary(endofiness$nb_actes)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   11.00   17.00   28.00   43.32   51.00  510.00
```

```r
med_acte <- median(endofiness$nb_actes)
endofiness$nbactes_eleve <- ifelse(endofiness$nb_actes > median(endofiness$nb_actes), 1, 0)
#endofiness$nbactes_eleve <- ifelse(endo$nb_actes > med_acte, 1, 0)
```

c. Calculez les indicateurs statistiques usuels pour la DMS selon la variable *nbactes_eleve*.

```
# par exemple
by(endofiness$dms_globale, endofiness$nbactes_eleve, epiDisplay::ci)
```

```
## endofiness$nbactes_eleve: 0
##    n     mean       sd        se lower95ci upper95ci
##  927 6.077555 5.372363 0.1764516  5.731264  6.423846
## ------------------------------------------------------------
## endofiness$nbactes_eleve: 1
##    n     mean       sd        se lower95ci upper95ci
##  879 5.660933 3.538199 0.1193405  5.426707  5.895159
```

```
endofiness %>% group_by(nbactes_eleve) %>% summarise(mean(dms_globale))
```

```
## # A tibble: 2 x 2
##   nbactes_eleve `mean(dms_globale)`
##           <dbl>               <dbl>
## 1             0                6.08
## 2             1                5.66
```

d. Testez l'hypothèse de l'indépendence entre la DMS et un nombre d'actes élevés avec le test statistique adapté. Interpréter les résultats.

```
by(endofiness$dms_globale, endofiness$nbactes_eleve, shapiro.test)
```

```
## endofiness$nbactes_eleve: 0
##
##  Shapiro-Wilk normality test
##
## data:  dd[x, ]
## W = 0.69367, p-value < 2.2e-16
##
## ------------------------------------------------------------
## endofiness$nbactes_eleve: 1
##
##  Shapiro-Wilk normality test
##
## data:  dd[x, ]
## W = 0.87922, p-value < 2.2e-16
```

```
wilcox.test(endofiness$dms_globale ~ endofiness$nbactes_eleve)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  endofiness$dms_globale by endofiness$nbactes_eleve
## W = 405084, p-value = 0.8332
## alternative hypothesis: true location shift is not equal to 0
```

2. ANOVA and Co

Vous vous interrogez ensuite sur l'existence de différences de DMS entre établissements aux modes de financements différents.

a. Calculez la moyenne des DMS par mode de financement.

```
by(endofiness$dms_globale,  endofiness$mft, mean)
```

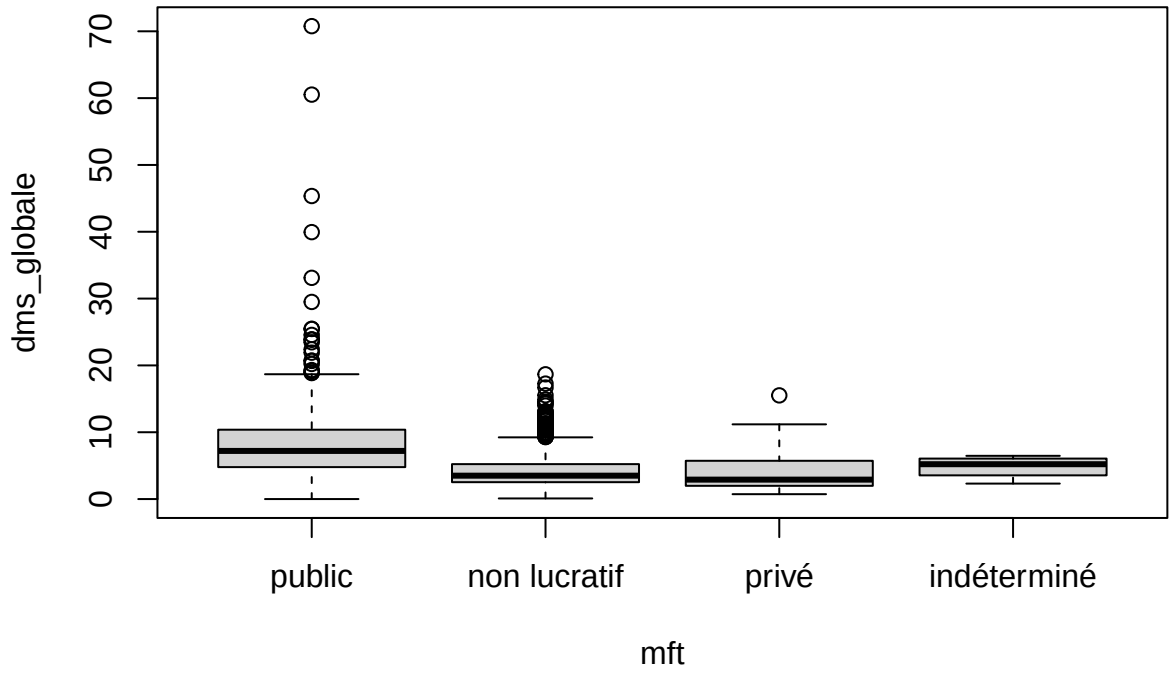
```
## endofiness$mft: public
## [1] 8.18202
## ------------------------------------------------------------
## endofiness$mft: non lucratif
## [1] 4.216218
## ------------------------------------------------------------
## endofiness$mft: privé
## [1] 4.296468
## ------------------------------------------------------------
## endofiness$mft: indéterminé
## [1] 4.801158
```

b. Appliquez le test statistique adapté pour explorer l'hypothèse d'une indépendence entre mode de financement et DMS. Vous pouvez utiliser la librarie *rstatix* pour une variété de tests.

```
# Analyse non paramétrique
kruskal.test(dms_globale ~ mft, data=endofiness)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  dms_globale by mft
## Kruskal-Wallis chi-squared = 455.68, df = 3, p-value < 2.2e-16
```

```
boxplot(dms_globale ~ mft, data=endofiness)
```




```
library(rstatix)
```

```
##
## Attaching package: 'rstatix'

## The following object is masked from 'package:MASS':
##
##     select

## The following object is masked from 'package:stats':
##
##     filter
```

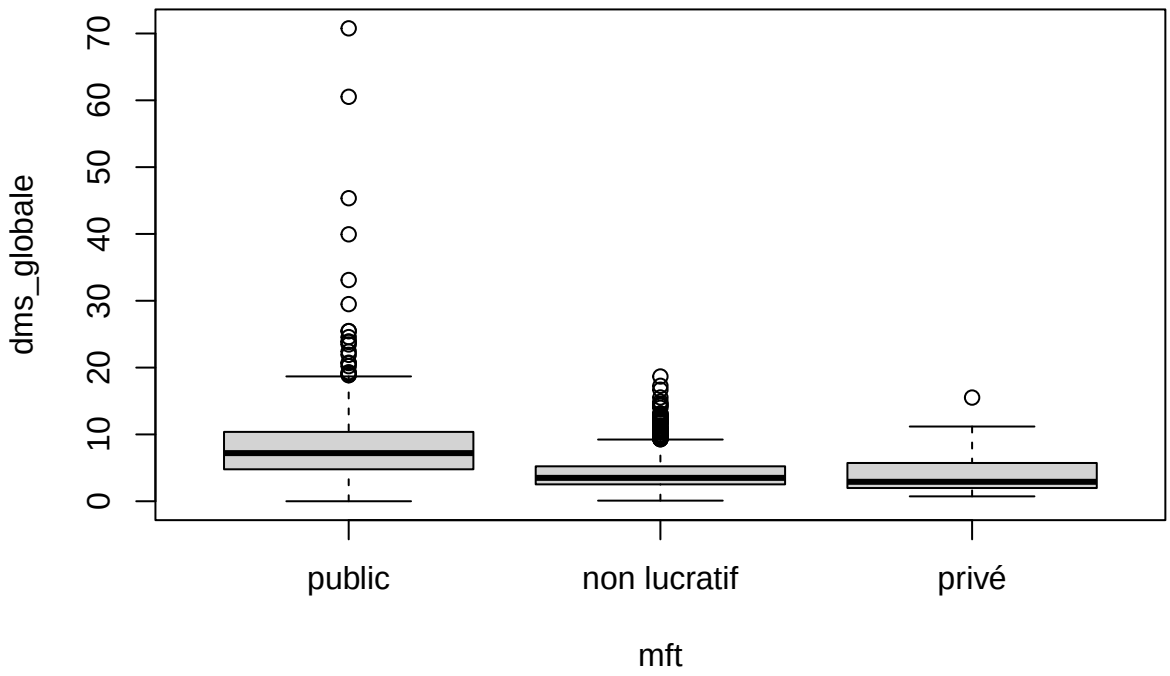
```
dunn_test(dms_globale ~ mft, data=endofiness)
```

```
## # A tibble: 6 x 9
##   .y.         group1 group2    n1    n2 statistic        p    p.adj p.adj.signif
## * <chr>       <chr>  <chr>  <int> <int>     <dbl>    <dbl>    <dbl> <chr>
## 1 dms_globale public non l~   754  1013   -21.1   3.17e-99 1.90e-98 ****
## 2 dms_globale public privé    754    35    -6.27  3.62e-10 1.81e- 9 ****
## 3 dms_globale public indét~   754     4    -1.32  1.88e- 1 7.50e- 1 ns
## 4 dms_globale non l~ privé   1013    35    -0.390 6.96e- 1 1   e+ 0 ns
## 5 dms_globale non l~ indét~  1013     4     0.711 4.77e- 1 1   e+ 0 ns
## 6 dms_globale privé  indét~    35     4     0.802 4.22e- 1 1   e+ 0 ns
```

```
# ANOVA -  Tukey HSD test même si méthodologiquement très limite
#aovmft <- aov(dms_globale ~ mft, data=endofiness)

# vérification a posteriori de la validité de l'ANOVA
#plot(aovmft)
#summary(aovmft)
#TukeyHSD(aovmft)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  dms_globale by mft
## Kruskal-Wallis chi-squared = 455.15, df = 2, p-value < 2.2e-16
```




```
## # A tibble: 3 x 9
##   .y.         group1 group2    n1    n2 statistic        p    p.adj p.adj.signif
## * <chr>       <chr>  <chr>  <int> <int>     <dbl>    <dbl>    <dbl> <chr>
## 1 dms_globale public non l~   754  1013   -21.1   4.09e-99 1.23e-98 ****
## 2 dms_globale public privé    754    35    -6.27  3.68e-10 7.37e-10 ****
## 3 dms_globale non l~ privé   1013    35    -0.391 6.96e- 1 6.96e- 1 ns
```

3. Régressions

A partir de vos données uniquement, vous chercher ensuite à expliquer les raisons possibles d'un dépassement de la DMS nationale.

a. Utilisez la méthode *glm()* pour réaliser une régression logistique et explorer les possibles liens entre DMS au dessus de la DMS nationale (variable *Top*) et les variables *nb_actes*, *mft*, *Ambu*.

```
## Changement de catégorie de référence pour la variable mft
#endofiness$mft <- relevel(endofiness$mft, ref="privé")
m1 <- glm(Top ~  nb_actes + mft + Ambu, data=endofiness, family = binomial())
summary(m1)
```

```
##
## Call:
## glm(formula = Top ~ nb_actes + mft + Ambu, family = binomial(),
##     data = endofiness)
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      0.511921   0.090500   5.657 1.54e-08 ***
```

```
## nb_actes        -0.000896   0.001444  -0.620 0.535026
## mftnon lucratif -1.887120   0.111637 -16.904  < 2e-16 ***
## mftprivé        -1.533674   0.414996  -3.696 0.000219 ***
## mftindéterminé  -1.130322   1.189169  -0.951 0.341851
## Ambu            -1.062614   0.237530  -4.474 7.69e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2355.3  on 1805  degrees of freedom
## Residual deviance: 1989.5  on 1800  degrees of freedom
## AIC: 2001.5
##
## Number of Fisher Scoring iterations: 4
```

b. Utilisez la fonction *logistic.display()* de la libary *epiDisplay* pour un affichage synthetique des OR.

```
logistic.display(m1, simplified = T)
```

```
##
##                        OR  lower95ci upper95ci     Pr(>|Z|)
## nb_actes        0.9991044 0.99627987 1.0019368 5.350263e-01
## mftnon lucratif 0.1515075 0.12173303 0.1885645 4.200689e-64
## mftprivé        0.2157416 0.09565087 0.4866074 2.193344e-04
## mftindéterminé  0.3229292 0.03139674 3.3214680 3.418511e-01
## Ambu            0.3455514 0.21693339 0.5504259 7.691657e-06
```

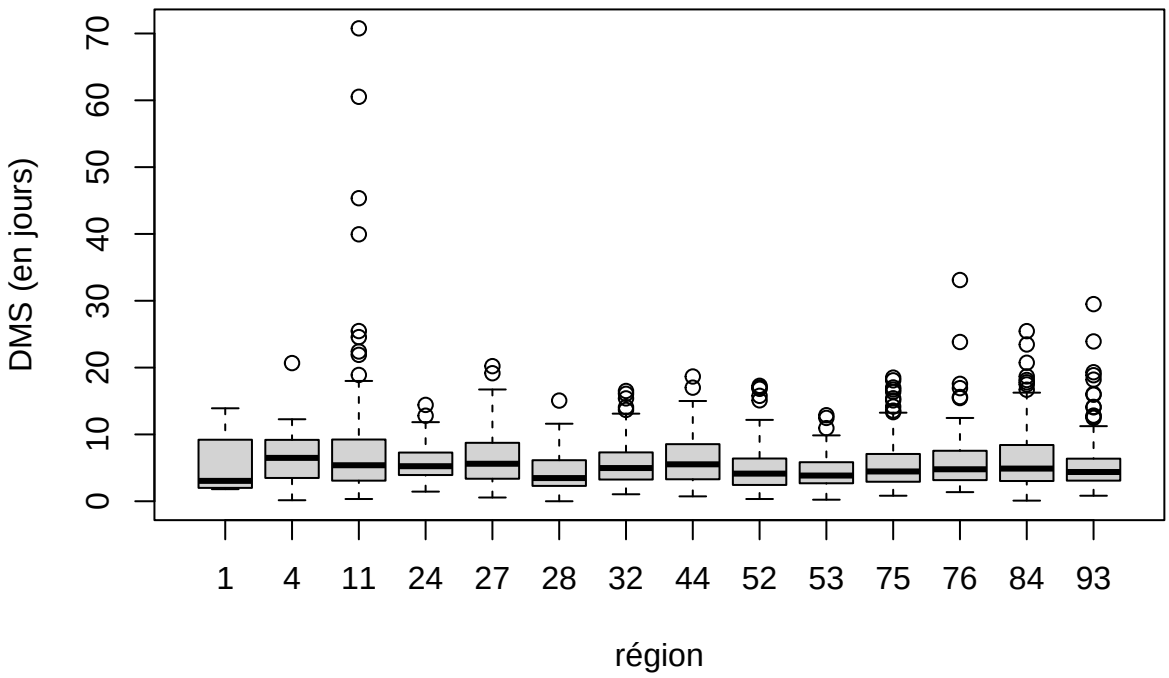
IV. Représentations graphiques et cartes

R dispose de nombreuses fonctionnalités graphiques de base mais la libraire *ggplot2* s'impose aujourd'hui grâce à l'optimisation des paramètres pour un respect des règles de technique de visualisation.

1. Graphiques

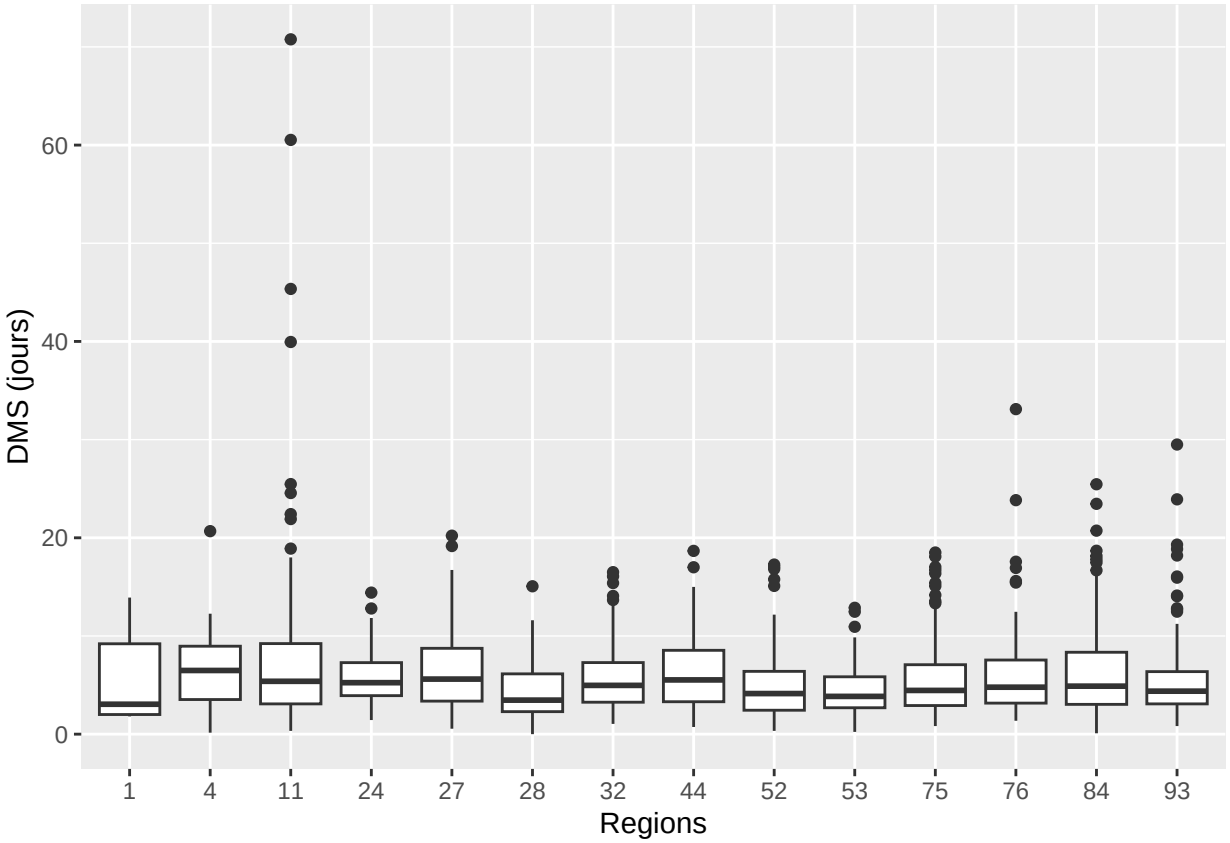
a. Utilisez la fonction *boxplot()* pour représenter les distributions des DMS par région avec des boîtes à moustaches

```
boxplot(endo$dms_globale ~ endo$reg, xlab="région", ylab="DMS (en jours)")
```



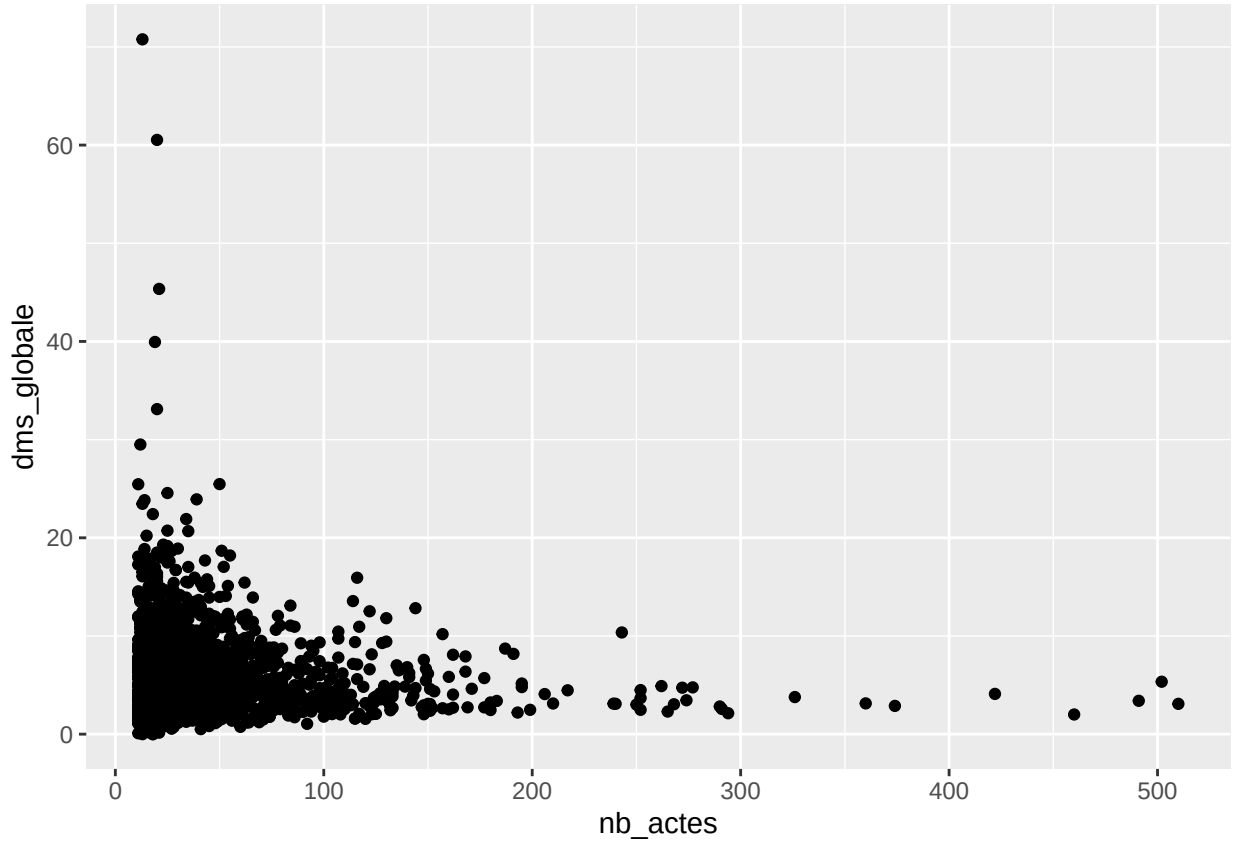
b. Utilisez la library *ggplot2* pour représenter les distributions des DMS par région avec des boîtes à moustaches

```
library(ggplot2)
ggplot(endo, aes(reg,dms_globale)) +
  geom_boxplot(aes(group=reg)) + xlab("Regions") + ylab("DMS (jours)")
```



c. Représentez le nombre d'actes en fonction de la DMS avec la syntaxe graphique incluant *geom_point()*

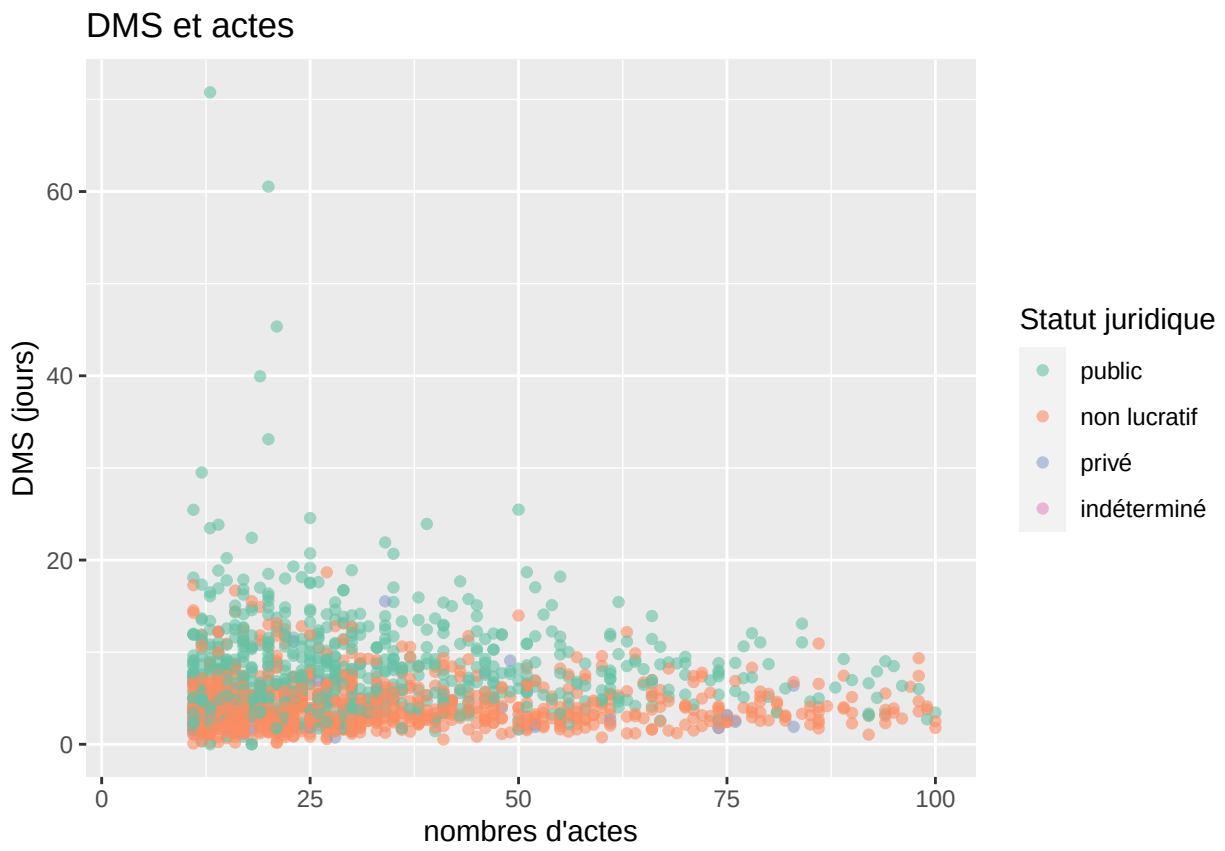
```
ggplot(endo, aes(x=nb_actes,  y=dms_globale)) + geom_point()
```



d. Représentez le nombre d'actes en fonction de la DMS et colorez les points selon les statuts juridiques

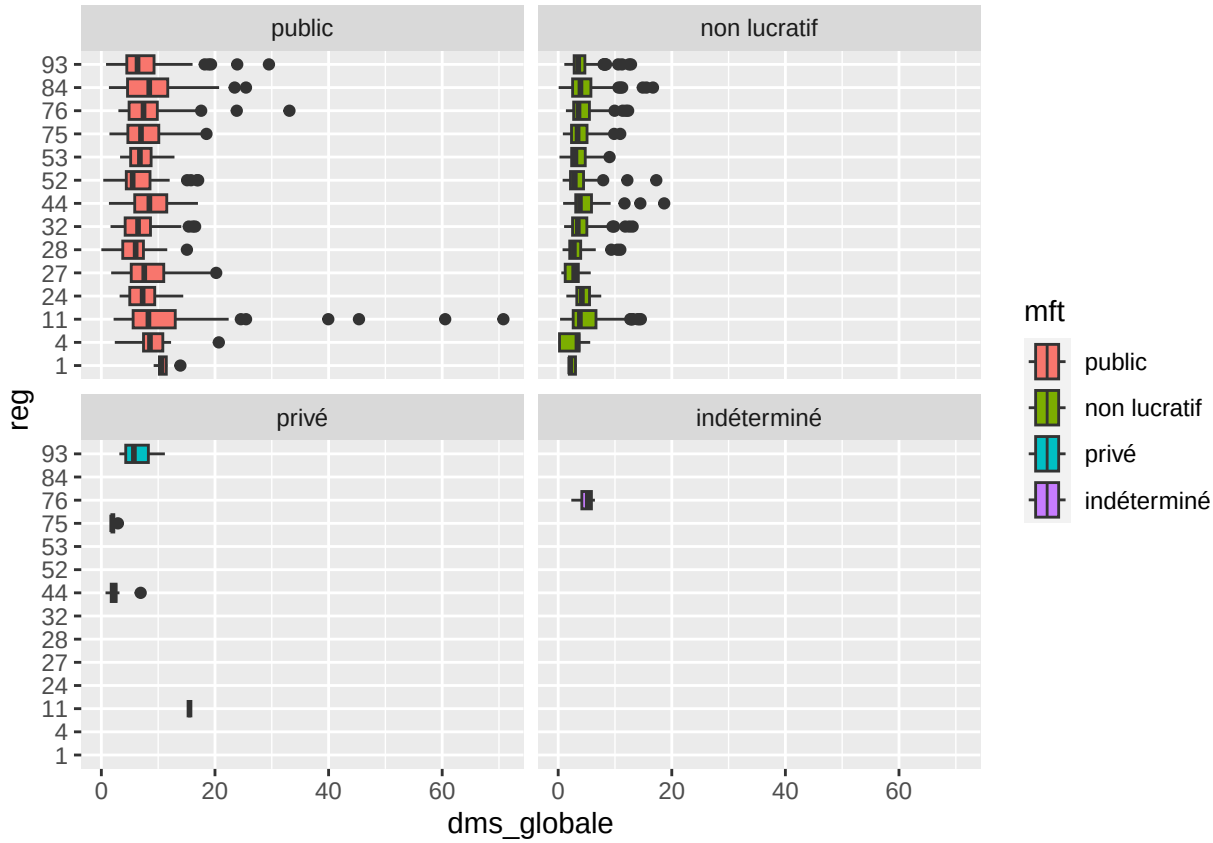
```
library(RColorBrewer)
ggplot(endofiness, aes(x=nb_actes,  y=dms_globale, colour=mft)) +
   geom_point(alpha=0.6) + scale_color_brewer(palette="Set2") +
   xlim(3,100) + labs(title="DMS et actes", y="DMS (jours)", x="nombres d'actes", colour="Statut juridi
```

```
## Warning: Removed 147 rows containing missing values (`geom_point()`).
```

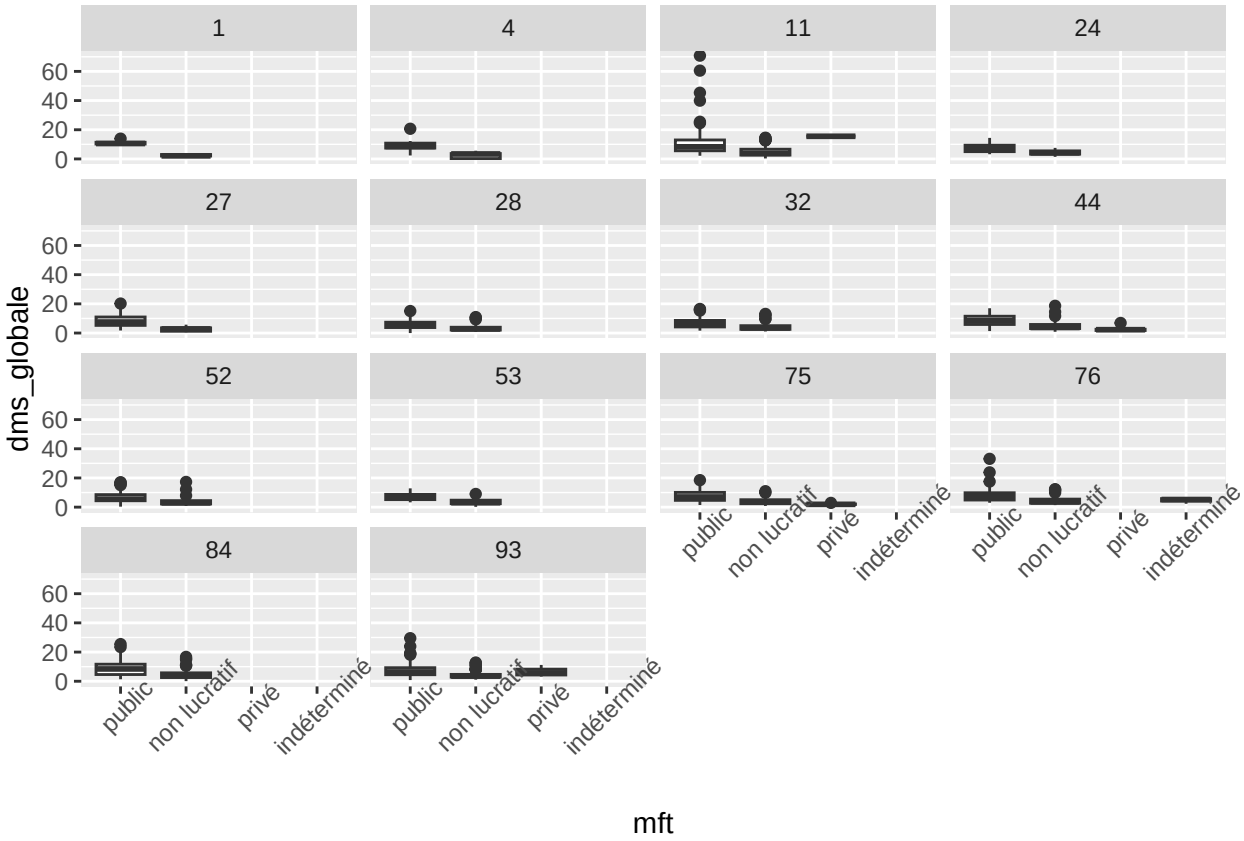


e. Utilisez une représentation en boîtes à moustaches pour représenter les distributions des DMS par région et par mode de financement en utilisant la syntaxe *facet_wrap()*

```
p <- ggplot(endofiness, aes(x=dms_globale, y=reg, fill=mft)) + geom_boxplot(aes(group=reg))
p + facet_wrap(.~ mft)
```

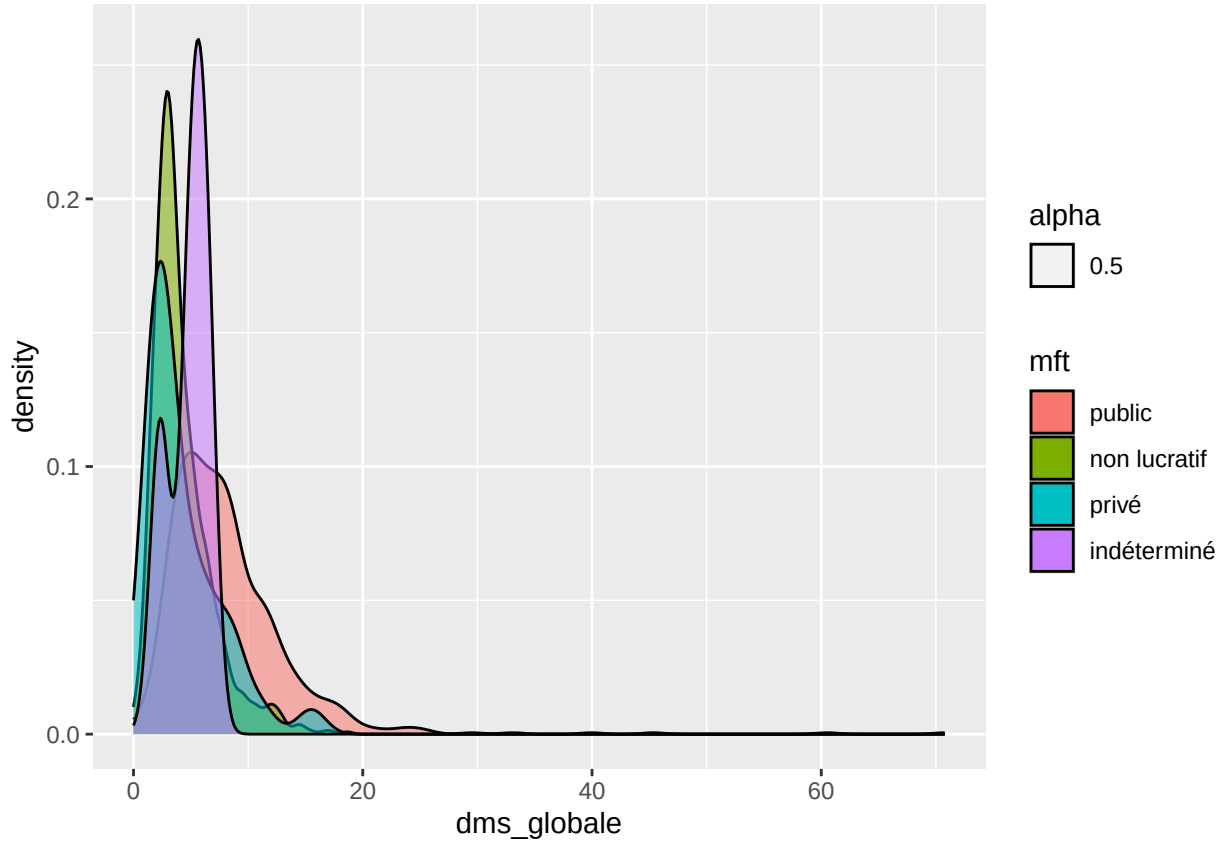



```
p <- ggplot(endofiness, aes(y=dms_globale, x=mft)) + geom_boxplot(aes(group=mft))
p + facet_wrap(.~ reg) +  theme(axis.text.x = element_text(angle = 45))
```



f. Représentez les distributions des DMS par mode de financement en utilisant *geom_density()*

```
ggplot(endofiness, aes(dms_globale, group=as.factor(mft))) +
  geom_density(aes(fill=mft, alpha = 0.5))
```



Extra: Diagramme en barre de la répartition des établissements selon leur mode de financement

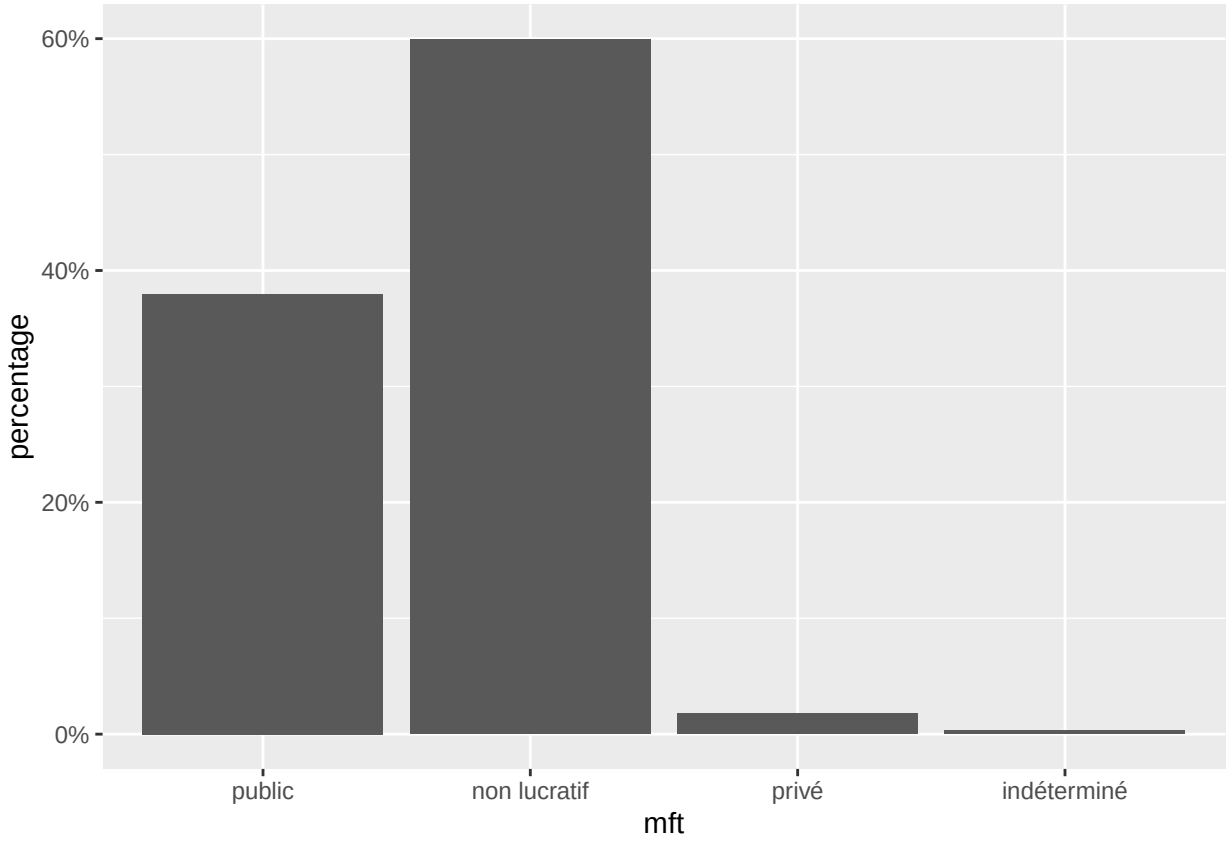
```
library(scales)  # pour l'échelle en pourcentage
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:epiDisplay':
##
##     alpha
```

```
prop_mft<- endofiness %>% select(FinessGeo, mft) %>% distinct(FinessGeo, mft) %>% count(mft) %>% mutate

ggplot(prop_mft, aes(x = mft, y = percentage)) + geom_bar(stat = "identity")+ scale_y_continuous(labels=
```



2. Cartes

Vous souhaitez représenter les DMS par région.

a. Calculez la moyenne des DMS par département et stocker le résultat sous la forme du data.frame du nom de *dmsdep*

```
endofiness$dep <- factor(endofiness$dep)
dmsdep <-  by(endofiness$dms_globale, endofiness$dep, mean, na.rm=T)
dmsdep <- as.vector(dmsdep)
dmsdep <- data.frame("dep"=levels(endofiness$dep), "dms"=dmsdep)

# Avec dplyr
dmsdep <- endofiness %>% group_by(dep) %>% summarise("dms" = mean(dms_globale, na.rm=T))
dmsdep$dep <- ifelse(nchar(as.character(dmsdep$dep))==2, as.character(dmsdep$dep),
                     paste("0", as.character(dmsdep$dep), sep=""))
dmsdep$dep <- factor(dmsdep$dep)
```

Pour construire une carte vous devez obtenir les coordonnées géographiques. Vous utilisez les library *ggmap* et *maps*.

b. Installer et charger ces libraries dans R puis avec la fonction *map_data()* charger la carte de France dans un objet R nommé *france*

```
library(ggmap)
```

```
## i Google's Terms of Service: <https://mapsplatform.google.com>
## i Please cite ggmap if you use it! Use `citation("ggmap")` for details.
```

```
library(maps)
france <- map_data("france")
```

c. Vous avez besoin de la correspondance entre les numéros de départements et leurs libellés disponibles dans le fichier *dep2reg.csv*. Lire le fichier dans R et nommer votre objet *dep2reg*.

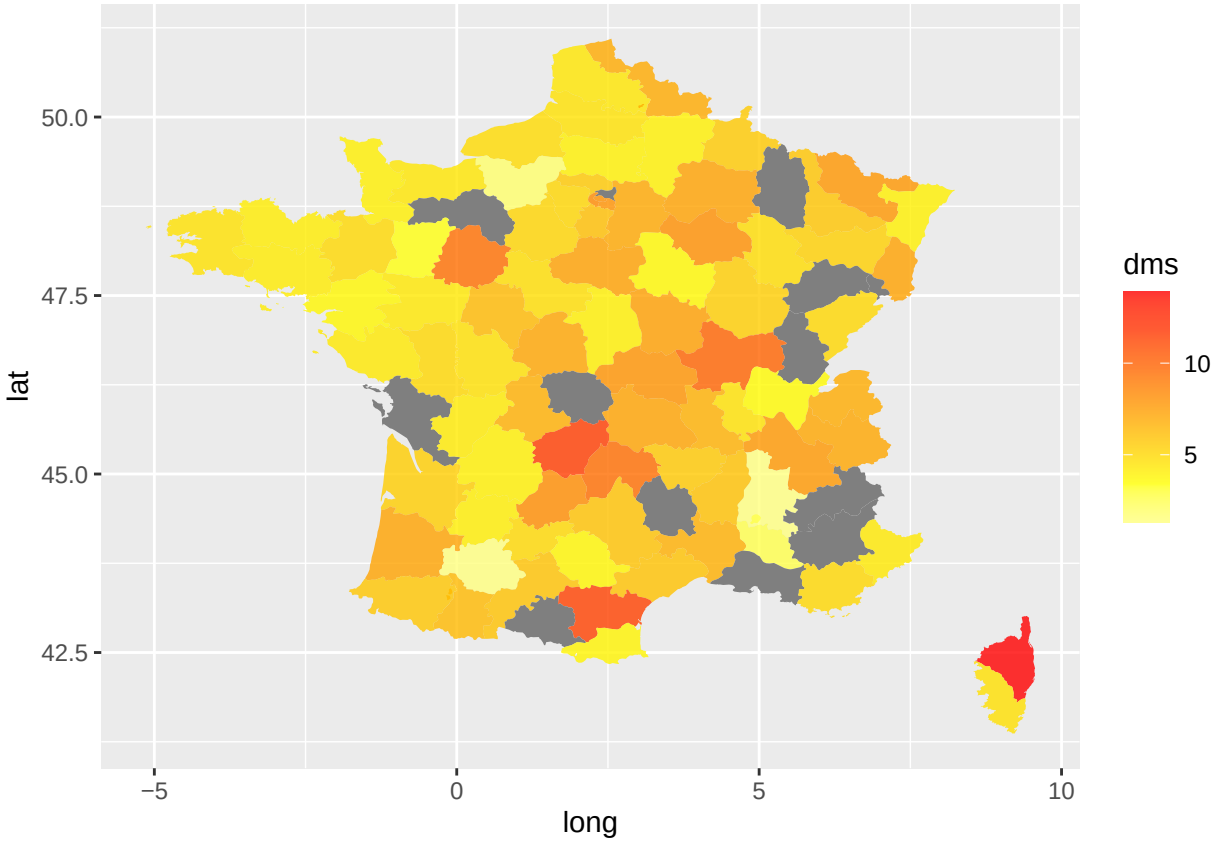
```
dep2reg <- read.csv("dep2reg.csv", header=T, sep=",")
```

d. Effectuer les jointures entre les tables *dmsdep* et *dep2reg* puis avec la table *france* pour créer l'objet *dmsdep2map*

```
dmsdep <-  merge(dmsdep, dep2reg, by ="dep", all.y=T)
dmsdep2map <- merge(france, dmsdep, by.x="region", by.y="label_dep", all.x=T)
```

e. Executer le code ci-dessous et commenter les résultats

```
dmsdep2map <- dmsdep2map[order(dmsdep2map$order), ]
ggplot(data = dmsdep2map) +
   geom_polygon(aes(x = long, y = lat, fill = dms, group = group)) +
  scale_fill_gradientn(colours =heat.colors(7, alpha=0.8, rev = T))
```




```
   coord_fixed(1.3)
```

```
## <ggproto object: Class CoordFixed, CoordCartesian, Coord, gg>
##     aspect: function
##     backtransform_range: function
##     clip: on
##     default: FALSE
```

```
##     distance: function
##     expand: TRUE
##     is_free: function
##     is_linear: function
##     labels: function
##     limits: list
##     modify_scales: function
##     range: function
##     ratio: 1.3
##     render_axis_h: function
##     render_axis_v: function
##     render_bg: function
##     render_fg: function
##     setup_data: function
##     setup_layout: function
##     setup_panel_guides: function
##     setup_panel_params: function
##     setup_params: function
##     train_panel_guides: function
##     transform: function
##     super:  <ggproto object: Class CoordFixed, CoordCartesian, Coord, gg>
```