Analyse de données médico-administratives avec R: Exploration des données Open MEDIC et Open DAMIR

Nolwenn Le Meur - EHESP

Sequence 2 - Mai-Juin 2023

Pour ces exercices je vous propose de travailler directement sous le format Rmarkdown.

I. Exploration des données Open MEDIC

L'offre de données Open Medic est constituée d'un ensemble de bases annuelles, portant sur l'usage du médicament, délivré en pharmacie de ville (2018-2021). Toutes les données sont extraites du système national interrégimes de l'Assurance Maladie (Sniiram). Les données sur le médicament sont restituées au travers de la classification ATC.

Vous disposez d'un jeux de données Open_Medic de 2021 qui permet d'étudier les dépenses annuelles de médicaments (montants remboursés - REM - et remboursables - BSE) ainsi que le nombre de boîtes délivrées, en fonction d'éléments descriptifs sur les bénéficiaires (tranche d'âge, sexe, région de résidence selon la nouvelle nomenclature Insee) ou de l'information sur la spécialité du prescripteur.

Le traitement des données a été opéré de manière à garantir la confidentialité des informations sur les bénéficiaires ainsi que sur les professionnels de santé. Notamment, certaines modalités ont été floutées (par les valeurs inconnues 9,99,999,etc..) lorsque le seuil critique de 10 bénéficiaires n'était pas respecté.

source: http://open-data-assurance-maladie.ameli.fr/medicaments/index.php

1. Lecture des données Open MEDIC

Vous disposez du fichier "OPEN_MEDIC_2021.CSV" pour étudier la consommation médicamenteuse dans votre région. https://assurance-maladie.ameli.fr/etudes-et-donnees/open-medic-base-complete-depenses-medicaments-2021

a. Dans RStudio, créez un nouveau projet R et fichier Rmarkdown du nom de *"openMedic.Rmd"*.

b. Utilisez tour à tour les fonctions *read.csv()* et *fread()* de la library *data.table* pour lire les données du fichier "OPEN_MEDIC_2021.CSV". Faites en sorte de créer 2 objets R: *med2021_csv* et *med2021*, respectivement. Quelles différences faites vous?

```
med2021_csv <- read.csv("../openMedic/OPEN_MEDIC_2021.CSV", header=T,  dec = ",",  sep=";")
```

```
library(data.table)
med2021 <- fread("../openMedic/OPEN_MEDIC_2021.CSV", header=T, dec = ",")
## Attention code CIP13 en 64byte
## transfo en caractère pour l'obtenir en entier
med2021$CIP13 <- as.character(med2021$CIP13)
med2021$BEN_REG <- factor(med2021$BEN_REG)
```

b. Supprimez l'objet *med2021_csv* créé par la fonction *read.csv()* en utilisant la fonction *rm()*.

```
rm(med2021_csv)
```

c. Observez le type de l'objet *med2021* issu de la fonction *fread()* en utilisant la fonction *str()*. Est-ce que les différentes variables sont au bon format?

Les lignes de commandes ci-dessous peuvent vous être utiles pour corriger quelques problèmes d'importation de format de données

```
# BSE lue comme une chaîne de caractère
## conversion nombre avec délimitation des milliers en point (.)
# elimination des points
med2021$BSE2 <- gsub("[.]", "", med2021$BSE)
## conversion des décimales avec virgule (syst FR) par des points (syst en)
med2021$BSE2  <- gsub(",", "\\.", med2021$BSE2)
# conversion de caractères en numérique possible
med2021$BSE2 <- as.numeric(med2021$BSE2)

#med2021$sexe <- factor(med2021$sexe, levels=c(1,2,9),
#                       labels=c("MASCULIN","FEMININ","INCONNU"))
```

d. Utilisez la fonction *summary()* pour un rapide résumé statistique des variables. Que remarquez vous?

```
summary(med2021)
```

```
##      ATC1              l_ATC1              ATC2              L_ATC2
##  Length:1820538     Length:1820538     Length:1820538     Length:1820538
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##      ATC3              L_ATC3              ATC4              L_ATC4
##  Length:1820538     Length:1820538     Length:1820538     Length:1820538
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##      ATC5              L_ATC5              CIP13             l_cip13
##  Length:1820538     Length:1820538     Length:1820538     Length:1820538
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    TOP_GEN             GEN_NUM            age              sexe
##  Length:1820538     Min.   :   0.0   Min.   : 0.00   Min.   :1.000
##  Class :character   1st Qu.:   0.0   1st Qu.:20.00   1st Qu.:1.000
##  Mode  :character   Median :   9.0   Median :20.00   Median :2.000
##                     Mean   : 284.6   Mean   :35.41   Mean   :1.545
##                     3rd Qu.: 549.0   3rd Qu.:60.00   3rd Qu.:2.000
##                     Max.   :1242.0   Max.   :99.00   Max.   :9.000
##
##     BEN_REG          PSP_SPE          BOITES            REM
##  11     :232146   Min.   : 1.00   Min.   :   -694   Length:1820538
```

```
##  93     :176607   1st Qu.: 1.00   1st Qu.:     39   Class :character
##  76     :163646   Median :14.00   Median :     96   Mode  :character
##  84     :161205   Mean   :40.36   Mean   :   1253
##  75     :148664   3rd Qu.:90.00   3rd Qu.:    373
##  44     :140102   Max.   :99.00   Max.   :6388400
##  (Other):798168
##      BSE
##  Length:1820538
##  Class :character
##  Mode  :character
##
##
##
##
```

e. Pour régulation (erreur de remboursements), l'assurance maladie enregistre des retraits de boîtes d'où les valeurs négatives qui doivent être supprimées

J'ai 1820538 enregistrements

```
med2021 <- med2021 %>% filter(BOITES > 0)
```

Après nettoyage des boîtes en erreur j'ai 1820308enregistrements

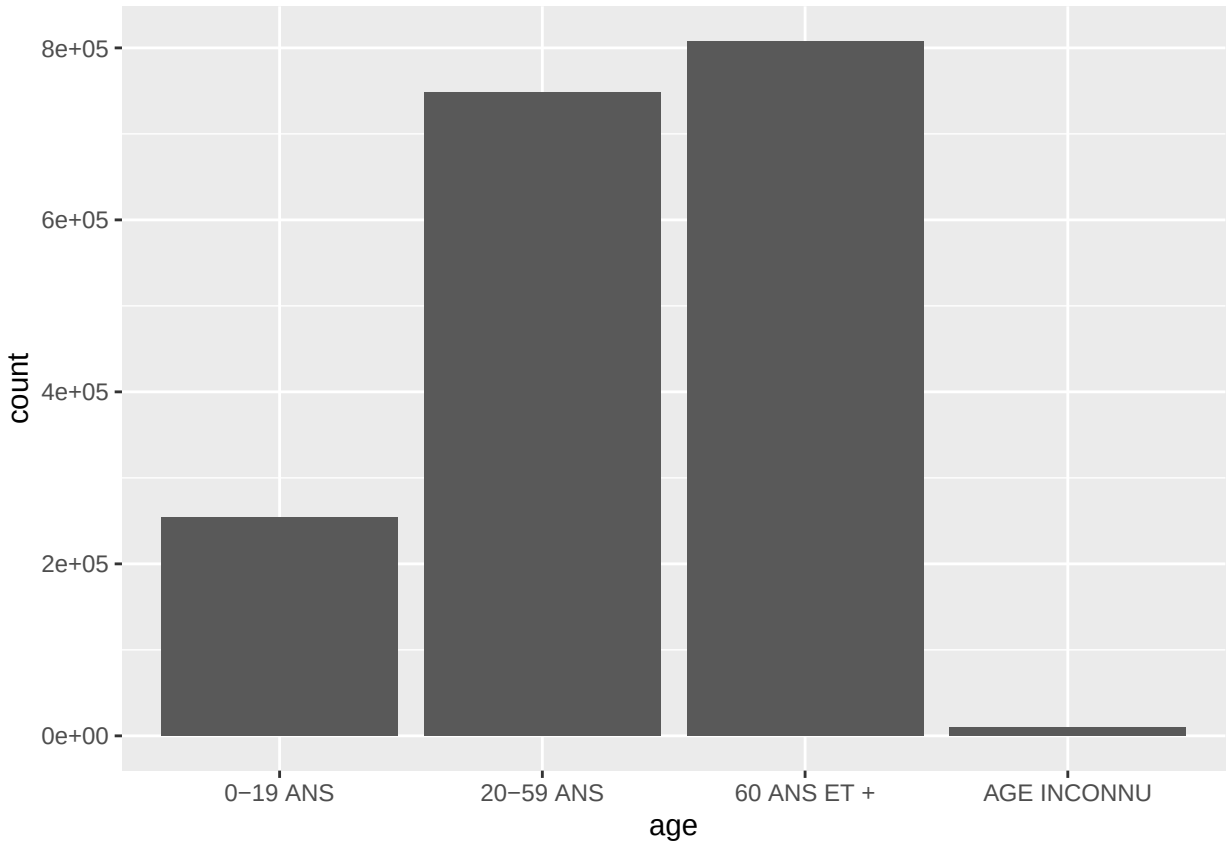
2. Description de l'âge des consommants

a. Documentez la variable **age** en la transformant au format *factor* et en ajoutant les étiquettes (labels): "0-19 ANS","20-59 ANS","60 ANS ET +" et "AGE INCONNU".

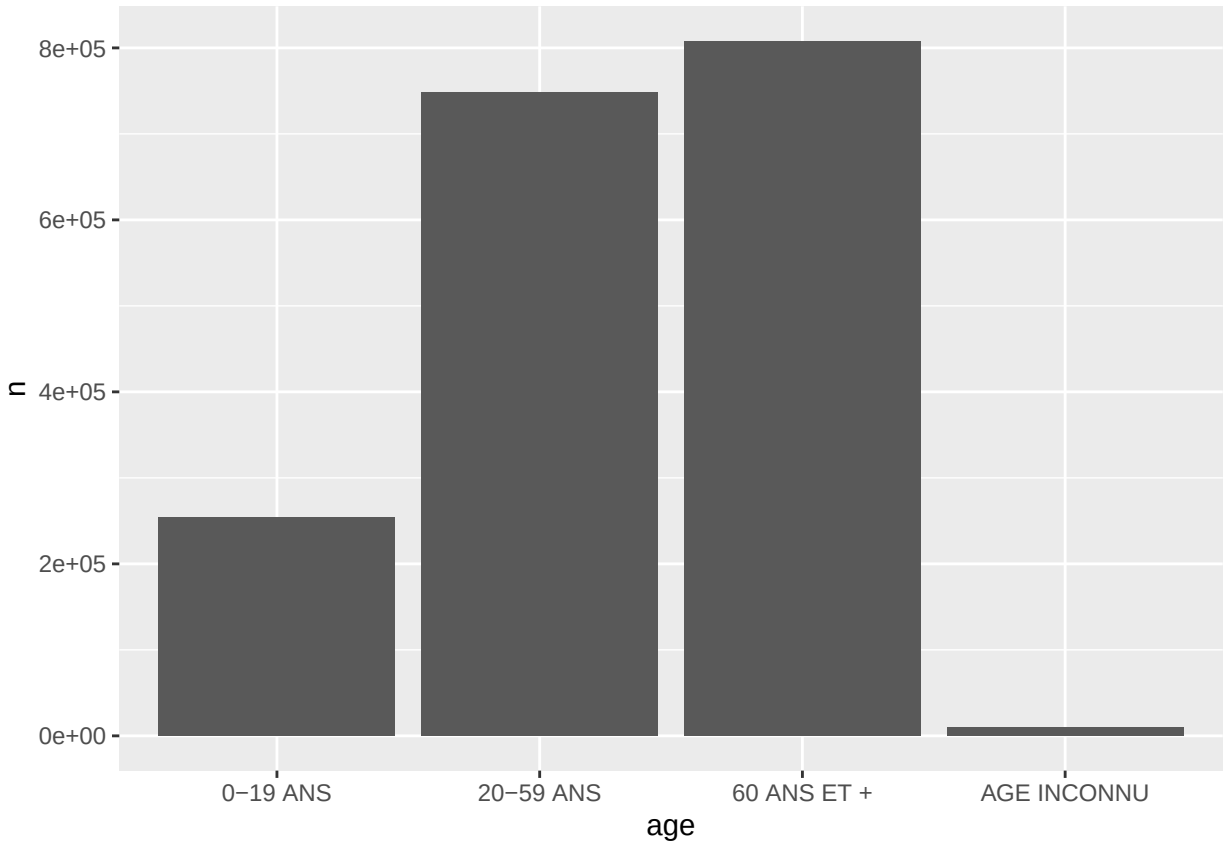
```
med2021$age <- factor(med2021$age, levels= c( 0, 20, 60, 99),
                      labels=c("0-19 ANS","20-59 ANS","60 ANS ET +","AGE INCONNU"))
```

b. Que representent les graphiques issus des 2 commandes qui suivent? Quelles sont les différences entre les 2 approches?

```
library(dplyr)
#command 1
ggplot(med2021, aes(age)) + geom_bar(stat="count")
```




```
#command 2
tabage<- med2021 %>% count(age)
ggplot(tabage, aes(x=age, y=n)) + geom_bar(stat="identity")
```



La fonction ci-dessous dénombre les patients dans chaque classe d'âge et dans chaque région.

```
# Frequence absolue
med2021 %>% group_by(BEN_REG) %>% count(age)
```

```
## # A tibble: 51 x 3
## # Groups:   BEN_REG [14]
##    BEN_REG age              n
##    <fct>   <fct>        <int>
##  1 5       0-19 ANS     11549
##  2 5       20-59 ANS    33025
##  3 5       60 ANS ET +  31851
##  4 11      0-19 ANS     30929
##  5 11      20-59 ANS    99783
##  6 11      60 ANS ET + 101399
##  7 11      AGE INCONNU     35
##  8 24      0-19 ANS     13791
##  9 24      20-59 ANS    42619
## 10 24      60 ANS ET +  48230
## # i 41 more rows
```

c. Complétez la fonction pour ajouter la fréquence relative au tableau

```
# Frequence relative
freqRegAge<- med2021 %>% group_by(BEN_REG) %>%
  count(age) %>%
  mutate("freq" = round(n /sum(n)*100,2))

# Bonus: Par région, nombre de classe ATC5 unique - éventail de prescription médicamenteuse en région
```

```r
med2021 %>% group_by(BEN_REG) %>% summarise("ATC5"=length(unique(ATC5)))
```

```
## # A tibble: 14 x 2
##    BEN_REG  ATC5
##    <fct>   <int>
##  1 5         955
##  2 11       1089
##  3 24        999
##  4 27       1015
##  5 28       1020
##  6 32       1048
##  7 44       1052
##  8 52       1017
##  9 53       1008
## 10 75       1055
## 11 76       1061
## 12 84       1065
## 13 93       1062
## 14 99       1138
```

d. Dénombrez pour chaque sexe, les patients dans chaque classe d'âge (fréquence absolue et relative)

```r
med2021$sexe <- factor(med2021$sexe, labels=c("MASCULIN","FEMININ","INCONNU"))
# ATTENTION: ne pas re-executer si déjà fait plus haut
#table(med2021$AGE, med2021$sexe)
#prop.table(table(med2021$AGE, med2021$sexe),1)*100
med2021 %>% group_by(sexe) %>% count(age) %>% mutate(freq = n / sum(n)*100)
```

```
## # A tibble: 12 x 4
## # Groups:   sexe [3]
##    sexe     age              n   freq
##    <fct>    <fct>        <int>  <dbl>
##  1 MASCULIN 0-19 ANS    118470 13.9
##  2 MASCULIN 20-59 ANS   339547 39.8
##  3 MASCULIN 60 ANS ET + 391880 45.9
##  4 MASCULIN AGE INCONNU   3794  0.444
##  5 FEMININ  0-19 ANS    134658 14.0
##  6 FEMININ  20-59 ANS   408672 42.4
##  7 FEMININ  60 ANS ET + 415805 43.2
##  8 FEMININ  AGE INCONNU   3910  0.406
##  9 INCONNU  0-19 ANS       542 15.2
## 10 INCONNU  20-59 ANS      271  7.59
## 11 INCONNU  60 ANS ET +    311  8.71
## 12 INCONNU  AGE INCONNU   2448 68.5
```

e. Dénombrez pour chaque groupe d'âge, les patients de chaque sexe (fréquence absolue et relative)

```r
med2021 %>% group_by(age) %>% count(sexe) %>% mutate(freq = n / sum(n)*100)
```

```
## # A tibble: 12 x 4
## # Groups:   age [4]
##    age         sexe          n    freq
##    <fct>       <fct>     <int>   <dbl>
##  1 0-19 ANS    MASCULIN 118470 46.7
##  2 0-19 ANS    FEMININ  134658 53.1
##  3 0-19 ANS    INCONNU     542  0.214
```

```
##  4 20-59 ANS   MASCULIN 339547 45.4
##  5 20-59 ANS   FEMININ  408672 54.6
##  6 20-59 ANS   INCONNU     271  0.0362
##  7 60 ANS ET + MASCULIN 391880 48.5
##  8 60 ANS ET + FEMININ  415805 51.5
##  9 60 ANS ET + INCONNU     311  0.0385
## 10 AGE INCONNU MASCULIN   3794 37.4
## 11 AGE INCONNU FEMININ    3910 38.5
## 12 AGE INCONNU INCONNU    2448 24.1
```

3. Décrire la prescription d'anxiolytiques dans votre région.

Votre objectif est de décrire le niveau de prescription d'anxiolytiques (ATC3 = N05B) par les médecins généralistes libéraux dans votre région.

a. Sélectionnez les données de votre région (exemple Bretagne code 53) en utilisant l'indexation ou la fonction *filter()*

```
#bzh<- med2021[med2021$BEN_REG == 53, ]
bzh <- med2021 %>% filter(BEN_REG == 53)
```

b. Sélectionnez le sous-ensemble de prescription d'anxiolytiques (ATC3 = N05B)

```
#bzhn05b <- bzh[bzh$ATC3 == "N05B", ]
bzhn05b <- bzh %>% filter(ATC3 == "N05B")
```

c. Sélectionnez le sous-ensemble de médicament prescrit par des médecins généralistes libéraux (PSP_SPE == 1)

```
#bzhn05bg<- bzhn05b[bzhn05b$PSP_SPE == 1,]
bzhn05bg <- bzhn05b %>% filter(PSP_SPE == 1)
```

d. Combinez l'ensemble des commandes a-b-c en une seule ligne

```
#bzhn05bg<- med2021[med2021$BEN_REG == 53 & med2021$ATC3 == "N05B" & med2021$PSP_SPE == 1, ]
bzhn05bg <- med2021 %>% filter(BEN_REG == 53 & ATC3 == "N05B" & PSP_SPE == 1)
```

e. Pour chaque sexe, calculez le nombre de boîtes prescriptes par classe d'âge

```
#by(bzhn05bg$BOITES, list(bzhn05bg$sexe, bzhn05bg$AGE), sum)
bzhn05bg_count <-  bzhn05bg %>% group_by(age, sexe) %>% summarise("boites"=sum(BOITES)) %>% mutate("pou
```

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

```
bzhn05bg_count
```

```
## # A tibble: 6 x 4
## # Groups:   age [3]
##   age         sexe      boites pourcentage
##   <fct>       <fct>      <int>       <dbl>
## 1 0-19 ANS    MASCULIN    7570        31.2
## 2 0-19 ANS    FEMININ    16716        68.8
## 3 20-59 ANS   MASCULIN  591445        45.4
## 4 20-59 ANS   FEMININ   711243        54.6
## 5 60 ANS ET + MASCULIN  643886        32.1
## 6 60 ANS ET + FEMININ  1364162        67.9
```

f. Utilisez la library *gtsummary* pour faire un tableau qui résume le niveau de prescriptions des médecins généralistes statistique chez les hommes et les femmes et par classe d'âge. Essayez d'afficher, les

moyennes, les écart-types et intervalle de confiance à la place des médianes et intervalles inter-quartiles.

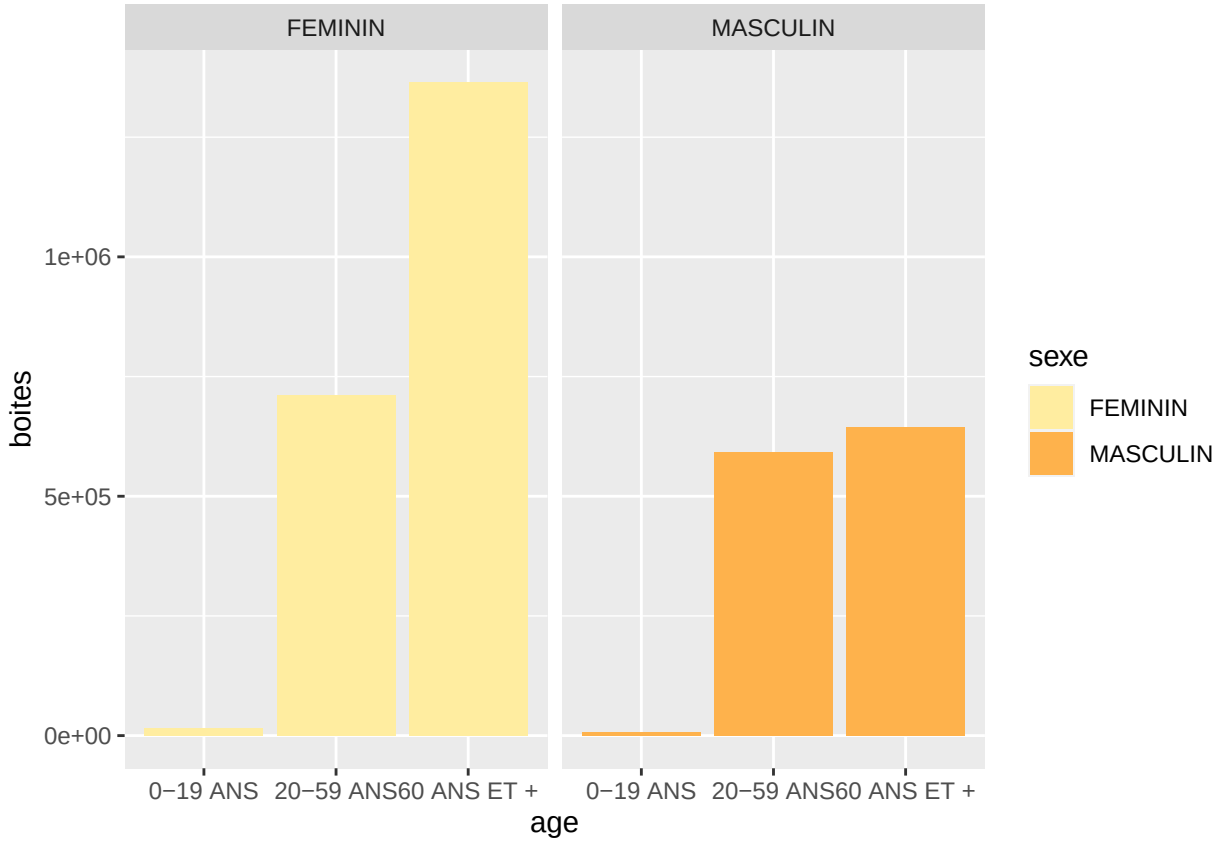
```
library(gtsummary)
bzhn05bg <- droplevels(bzhn05bg)
bzhn05bg %>% select(age, sexe, BOITES) %>%
  tbl_summary(by=age)

med2021 %>% select(age, sexe, BOITES) %>%
  tbl_summary(by=age,
              statistic = all_continuous() ~"{mean} ({sd})") %>% add_ci()
```

4. Niveau de consommations par âge et par sexe

a. Utilisez la librarie *ggplot2* et la fonction *geom_bar()* pour représenter vos résultats. Utilisez la page d'aide pour embellir votre graphique

```
library(ggplot2)
#ggplot(data = bzhn05bg_count, aes(x=AGE, y=boites, fill=sexe)) + geom_bar(stat="identity")
# Variable sexe avec Femme en référence pour correspondance de colour
bzhn05bg_count$sexe <- relevel(bzhn05bg_count$sexe, ref = "FEMININ")
p <- ggplot(data = bzhn05bg_count, aes(x=age, y=boites, fill=sexe))
p + geom_bar(stat="identity") + scale_fill_brewer(palette="YlOrRd") + facet_wrap(.~ sexe)
```

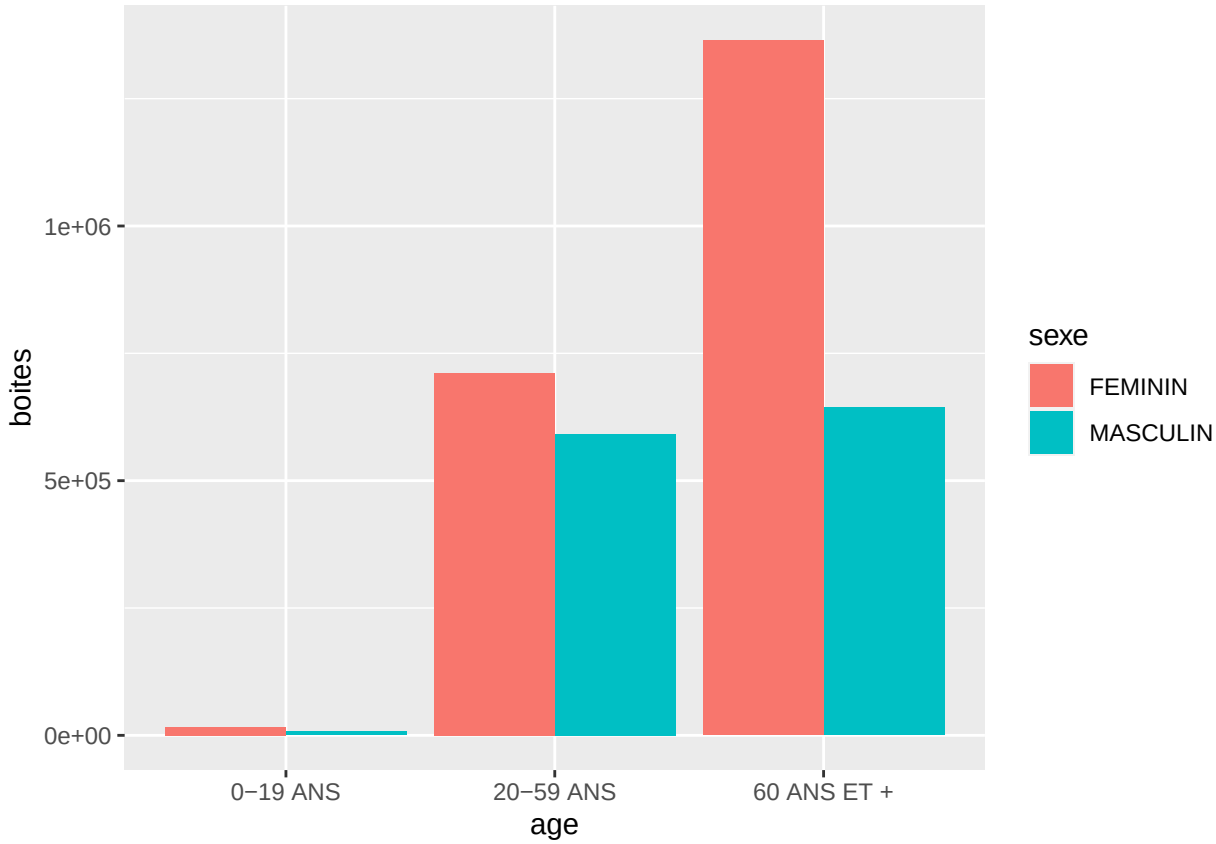



```
#ggplot(data = bzhn05bg_count, aes(x=sexe, y=boites, fill=AGE)) + #geom_bar(stat="identity")
```

b. Refaites votre graphique avec les barres adjacentes

```
library(ggplot2)
ggplot(data = bzhn05bg_count, aes(x=age, y=boites, fill=sexe)) +
```

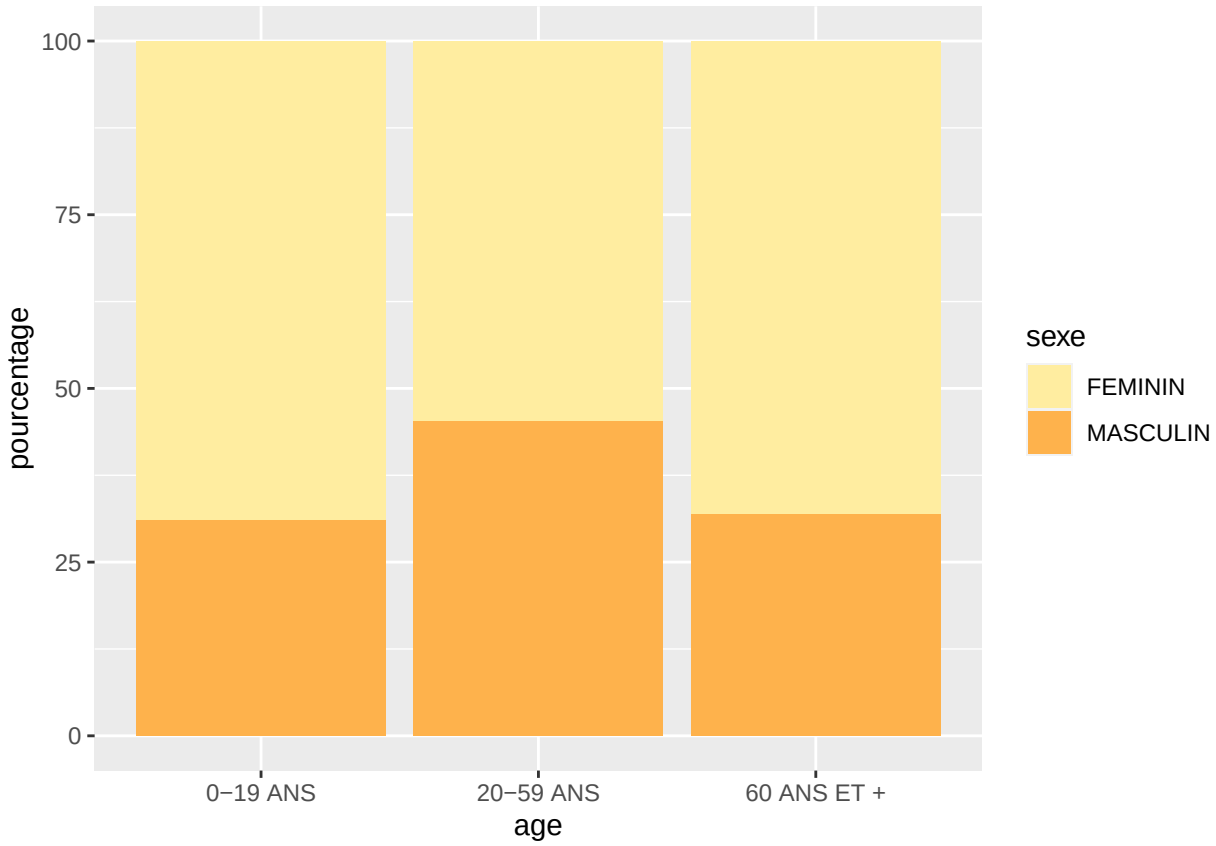
```
  geom_bar(stat="identity", position = "dodge")
```




```
#ggplot(data = freq_anxio, aes(x=AGE, y=boites, fill=sexe)) + geom_bar(stat="identity", position = "dod
```

b. Refaites votre graphique avec des proportions

```
ggplot(data = bzhn05bg_count, aes(x=age, y=pourcentage, fill=sexe)) +
  geom_bar(stat="identity")  +
  scale_fill_brewer(palette="YlOrRd")
```



5. Evolution des ventes

Vous désirez visualiser l'évolution des ventes sur plusieurs années. Votre data manager vous demande pour cela de vérifier son code avant de récupérer les données simplifiées pour les années 2018 à 2021.

a. Votre data manager a utilisé une boucle *for()* et sauvegardé les données dans l'objet R *openMed*. Que fait cette boucle *for()*.

```
openMed <- c()
for(i in 2018:2021){
    filename2open <- paste("../openMedic/OPEN_MEDIC_", i, ".CSV",sep="")
    temp <- fread(filename2open, header=T, dec = ",")
    temp <- temp[, -seq(2,12,2)]
    if(i==2019){
      colnames(temp)[10] <- "sexe"
    }
    temp <- cbind(temp,"ANNEE"=i)
  openMed <- rbind(openMed, temp)
}
# verification
# table(openMed$ANNE)
```

b. Que fait le code chunck suivant et pourquoi dans cet ordre?

```
openMed53 <- openMed %>% filter(BEN_REG == 53 & ATC3 == "N05B" & PSP_SPE%in%c(1,17))
openMed53$age <- factor(openMed53$age,
                        labels=c("0-19 ANS","20-59 ANS","60 ANS ET +"))
openMed53$sexe <- factor(openMed53$sexe, labels=c("Homme","Femme"))
```

```
# save(openMed53, file="openMed53.Rdata")
```

c. Calculez le nombre de boîtes prescriptes par classe d'âge, par année, par sexe et par spécialité de prescripteur avec la fonction *summarise()* de *dplyr*.

```
load("openMed53.Rdata")
evolution <- openMed53  %>%
  group_by(age, ANNEE, sexe, PSP_SPE) %>%
  summarise("boites"=sum(BOITES))
# Avec data.table
ev2 <- openMed53[ , .(Nb=sum(BOITES), Moy=mean(BOITES)), by=list(age, ANNEE, sexe, PSP_SPE)]
```

d. Représentez graphiquement l'évolution des ventes au cours des 3 années par classe d'âge, par sexe et par spécialité de prescripteur en utilisant la librairie *ggplot2*

```
evolution$ANNEE <- as.Date(as.character(evolution$ANNEE), "%Y")
ggplot(data = evolution, aes(y=boites, x=ANNEE,color=age, fill=age)) + geom_line() + facet_grid(.~sexe*
```

II. Exploration des données Open DAMIR

1. Acte d'echographie ccam à l'hospital en janvier 2021

Vous vous intérrogez sur l'activité des services d'échographie en établissement hospitalier. Pour cette analyse vous disposez des fichiers open DAMIR mis à disposition mensuellement par l'assurance maladie sur le site open data du gourvernement (sources: https://www.data.gouv.fr/fr/datasets/open-damir-base-complete-sur-les-depenses-dassurance-maladie-inter-regimes/)

Dans un premier temps vous découvrez la structure des fichiers. a. Lire les 10 premières lignes du fichier *"A202101.csv"* grâce à la fonction *fread()* et observer sa structure.

```
m1 <- fread("../openDamir/A202101.csv.gz", nrows=10)
```

a.Vous n'avez pas besoin de l'ensemble des variables. Relire le fichier (ensemble des lignes) en sélectionnant les variables:

- AGE_BEN_SNDS
- BEN_RES_REG
- BEN_SEX_COD
- ETE_CAT_SNDS
- ETE_REG_COD
- PRS_ACT_QTE
- PRS_NAT
- SOI_ANN
- SOI_MOI

```
library(data.table)
#namesvar <- fread("A202101.csv",nrow=1)
#colnames(namesvar)
m1 <- fread("../openDamir/A202101.csv.gz",
            select = c("AGE_BEN_SNDS", "BEN_RES_REG", "BEN_SEX_COD",
                                       "ETE_CAT_SNDS", "ETE_REG_COD", "PRS_ACT_QTE",
                                     "PRS_NAT", "SOI_ANN", "SOI_MOI"))
```

c. Selectionnez la sous-population des prescriptions hospitalières relatives (ETE_CAT_SNDS: "1101","1102") en utilisant *filter()* de la library *dplyr*

d. Sélectionnez la sous-population des échographies réalisées en 2021 (PRS_NAT: 1324) en utilisant *filter()* de la library *dplyr*

e. Dénombrer les actes d'échographie par région

```
## # A tibble: 13 x 2
##    ETE_REG_COD nbActe
##          <int>  <int>
##  1           5    335
##  2          11   1917
##  3          24    555
##  4          27    762
##  5          28    802
##  6          32   1812
##  7          44   2094
##  8          52    687
##  9          53    454
## 10          75    584
## 11          76   1578
## 12          84   1508
## 13          93    483
```

2. Evolution du volume et du taux d'actes d'echographie ccam à l'hospital pour le premier trimestre 2021

a. Vous souhaitez les mêmes statistiques pour les 3 premiers mois de l'année 2021. Que fait la code chunk ci-dessous?

```
hecho <- c()
filepath <- c("../openDamir/A202101.csv.gz",
              "../openDamir/A202102.csv.gz", "../openDamir/A202103.csv.gz")
for (i in 1:3){
  m <- fread(filepath[i], select = c("AGE_BEN_SNDS", "BEN_RES_REG", "BEN_SEX_COD",
                                     "ETE_CAT_SNDS", "ETE_REG_COD", "PRS_ACT_QTE",
                                   "PRS_NAT","SOI_ANN","SOI_MOI"))
  hosp <- m %>% filter(ETE_CAT_SNDS%in%c("1101","1102"))
  temp <- hosp %>% filter(PRS_NAT==1324, SOI_ANN==2021)
  hecho <- rbind(hecho, temp)
}
hecho$SOI_MOI <- factor(hecho$SOI_MOI, labels=c("JAN", "FEV", "MAR"))
hecho$BEN_SEX_COD <- factor(hecho$BEN_SEX_COD, labels=c("Hommes", "Femmes"))
save(hecho, file="hecho.RData")
```

b.Pour optimiser le temps de traitement, nous vous avons générer la base *hecho*. Charger la base *hecho* dans votre environnement avec la fonction *load()*

```
load("hecho.RData")
```

b. Résumer le nombre d'échographies prescrites par les établissements par région et par mois en utilisant les fonctions *group_by()* et *summarise()* de la library dplyr

```
library(knitr)
library(tidyr)
restab1<- hecho %>% group_by(ETE_REG_COD, SOI_MOI) %>% summarise("NB"=sum(PRS_ACT_QTE))
```

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

```
#bonus
restab1 <- spread(restab1, SOI_MOI, NB)
```

```
colnames(restab1) <- c("Région", "Janvier","Février","Mars")
kable(restab1, caption="Nombre d'échographies par région lors du premier trimestre 2021")
```

Table 1: Nombre d'échographies par région lors du premier trimestre 2021

Région	Janvier	Février	Mars
5	7124	5837	1252
11	29794	19100	5506
24	16689	14211	1477
27	18700	15732	2357
28	25524	20458	5592
32	46062	36786	6826
44	39550	33314	4151
52	24043	16608	2467
53	26256	18076	2207
75	37028	24784	2750
76	33179	26001	8728
84	55937	46914	10142
93	25901	22014	3877

c. Résumer le nombre d'échographies prescrites par les établissements par region, par mois et par sexe en utilisant les fonctions *group_by()* et *summarise()* de la library dplyr

```
hecho %>% group_by(ETE_REG_COD, SOI_MOI, BEN_SEX_COD) %>% summarise("NB"=sum(PRS_ACT_QTE))
```

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

```
## # A tibble: 78 x 4
## # Groups:   ETE_REG_COD, SOI_MOI [39]
##    ETE_REG_COD SOI_MOI BEN_SEX_COD    NB
##          <int> <fct>   <fct>       <int>
##  1           5 JAN     Hommes       1234
##  2           5 JAN     Femmes       5890
##  3           5 FEV     Hommes       1115
##  4           5 FEV     Femmes       4722
##  5           5 MAR     Hommes        382
##  6           5 MAR     Femmes        870
##  7          11 JAN     Hommes       6644
##  8          11 JAN     Femmes      23150
##  9          11 FEV     Hommes       4421
## 10          11 FEV     Femmes      14679
## # i 68 more rows
```

d. Visualiser ces données grâce aux méthodes de la librarie *ggplot2*.

```
library(ggplot2)
echoRegSexe <- hecho %>% group_by(ETE_REG_COD, SOI_MOI, BEN_SEX_COD) %>% summarise("NB"=sum(PRS_ACT_QTE
```

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

```
ggplot(echoRegSexe, aes(y=NB, x=SOI_MOI, fill=BEN_SEX_COD)) +
  geom_bar(stat="identity", position="dodge") +
```

```
  facet_wrap(.~ETE_REG_COD) +
  ylab("echography volume")
```


```
#p <- ggplot(echoRegSexe, aes(y=NB, x=SOI_MOI, fill=BEN_SEX_COD)) + #geom_bar(stat="identity", position
#p +  facet_wrap(.~ETE_REG_COD) +ylab("echography volume")
```

e. Vous souhaitez standardiser les données par rapport à la taille de la population par région. Vous disposez du fichier *pop-reg.csv* dans ce but. Calculez les taux d'échographie par région, par sexe et par chaque mois (jouer avec les variables qualitatives pour obtenir des regards différents sur la distribution des données).

```
# lecture du fichier et calcul pour de la population par sexe et par région
popreg<- read.csv("pop-reg.csv", header=T, sep=",")

genderpop <- popreg %>% filter(pop%in%c("Hommes", "Femmes")) %>% group_by(region, pop) %>% summarise("p
```

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

```
## Prise en compte de l'outre-mer en global
genderpop$region <- factor(genderpop$region)
levels(genderpop$region)[1:5] <- "5"
genderpop <- genderpop %>% group_by(region, pop) %>% summarise("population"=sum(population))
```

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

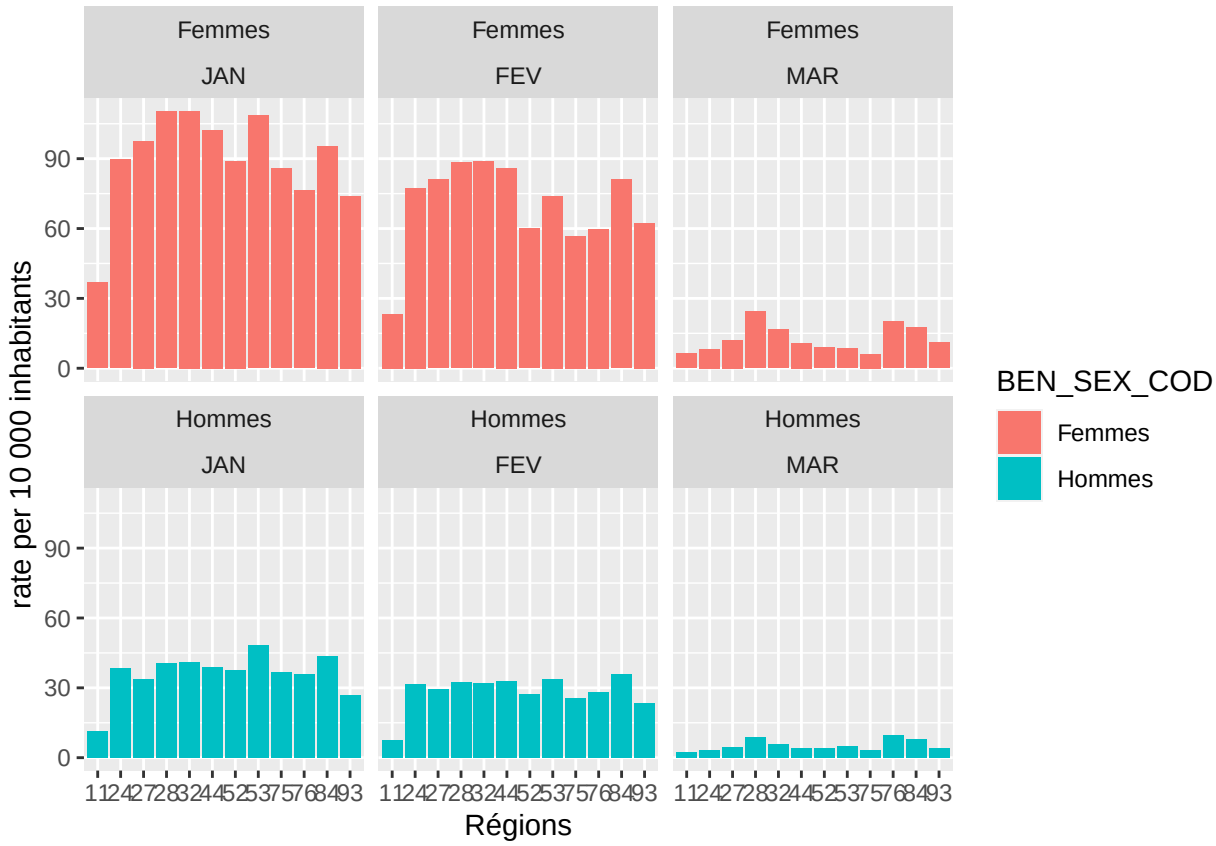
```
# levels(genderpop) <- c("Hommes", "Femmes")
# jointure avec fichier de consommation
```

```r
echoRegSexe$ETE_REG_COD <- as.factor(echoRegSexe$ETE_REG_COD)
echoRegSexe <- left_join(echoRegSexe, genderpop, by=c("ETE_REG_COD"="region", "BEN_SEX_COD"="pop"))
echoRate <- echoRegSexe %>% mutate(rate= NB/population*10000)

# éliminer la région 5 (outre-mer) non référencée dans le fichier pop
echoRate <- echoRate %>% filter(ETE_REG_COD!=5)

# par region
ggplot(echoRate, aes(y=rate, x=BEN_SEX_COD, fill=SOI_MOI)) + geom_bar(stat="identity", position="dodge")
```


```r
# par sexe et mois de soins
ggplot(echoRate, aes(y=rate, x=as.factor(ETE_REG_COD), fill=BEN_SEX_COD)) + geom_bar(stat="identity", p
```




```
library(sf)
```

```
## Linking to GEOS 3.11.0, GDAL 3.5.3, PROJ 9.1.0; sf_use_s2() is TRUE
```

```
library(mapsf)
#library(RColorBrewer)
# import ING shape files as an sf object
FrMap <- st_read(dsn="~/Documents/Projets/01_EHESP/04_Projets/00_BasesDonnees/ADMIN-EXPRESS-COG_1-1__SH

echoRate2 <- left_join(echoRate, FrMap, by=c("ETE_REG_COD"="INSEE_REG"))

sf <- st_set_geometry(echoRate2, echoRate2$geometry)
sfHJ <- sf %>% filter(BEN_SEX_COD=="Hommes", SOI_MOI=="JAN")
#-----------------

plot(st_geometry(sfHJ), col = NA, border = NA, bg = NA)
# main plot
mf_map(
  x = sfHJ,
  var = "rate",
  type = "choro",
  breaks = "equal",
  pal =  mf_get_pal(n = 5, pal = "viridis"),
  border = "grey40",
  lwd = 0.2,
  leg_pos = "right",
  leg_title = "",
  add = TRUE
```

```
)
# layout
mf_layout(title = "2021",
            credits = "Sources: SNIIRAM")
```