1 Quotients de mortalité et espérance de vie

Charger dans un **data.frame** le contenu du fichier **table−mortalite−france.csv** (`tmort <−read.csv2("table-mortalite-france.csv")`).

Remarque. Lorsqu'on veut charger dans la session **R** un fichier, il faut s'assurer que fichier est bien situé dans le répertoire courant de la session. Pour afficher les noms des fichiers de type **csv** du répertoire courant : **dir**(`pattern="*.csv"`). Si vous souhaitez examiner la liste de tous les fichiers du répertoire courant : **dir**(). Pour connaître le chemin qui conduit au répertoire courant **getwd**() (pour *get working directory*), et si vous souhaitez modifier le répertoire courant « à la main », essayez **setwd**() (pour *set working directory*).

Sélection et projection

1. Sélection de sous-tableaux de données. La fonction **subset**() permet de sélectionner élégamment un sous-ensemble de lignes vérifiant une condition donnée, par exemple
   subset(`tmort,ANNEE==1806 &DEPARTEMENT=="SEINE"`)
   sélectionne les lignes de `tmort` qui vérifient les deux conditions `ANNEE==1806` et `DEPARTEMENT=="SEINE"`

   Solution: La fonction **subset** permet de choisir dans un **data.frame** à la fois des lignes (ici celles pour lesquelles, la variable `DEPARTEMENT` vaut `"SEINE"` et la variable `ANNEE` vaut 1806) et des colonnes si on utilise l'argument `select`.
   Par exemple si on veut seulement choisir les colonnes `Age`, `Qmort` et `Survie` et les lignes correspondant au département de la `"LOIRE"` et à l'`ANNEE` 1856, on écrira

```
tselection <- subset(tmort, ANNEE==1856 &
                            DEPARTEMENT=="LOIRE",
                            select=c("Age","Qmort","Survie"))
tseine1806 <- subset(tmort, ANNEE==1856 &
                            DEPARTEMENT=="SEINE",
                            select=c("Age","Qmort","Survie"))
```

   Dans le langage des bases de données ou des systèmes d'information, en fait dans le langage de requête `SQL`, la commande `SELECT` permet de réaliser ces opérations qui s'appellent sélection et projection. Plutôt que sur des **data.frame**, les requêtes des bases de données relationnelles opèrent sur des tables relationnelles.
   Un environnement de calcul statistique comme **R** permet de réaliser des opérations de type « bases de données »de deux façons : soit en travaillant dans l'espace de travail `R` sur des **data.frame** à l'aide de **subset**,**merge**, **aggregate**, soit en confiant le travail à un système de gestion de bases de données via un paquetage comme `RODBC` (R-*open data base connectivity*). Cette dernière manière de procéder permet de traiter des données de très grandes tailles.

2. Sélectionner les lignes de `tmort` correspondant aux trois départements d'Île de France (`"SEINE","SEINE-ET-OISE","SEINE-ET-MARNE"`)

   Solution: On utilise la fonction **subset** mais la clause de choix est une disjonction (un OU) réalisée avec l'opérateur | :
   `DEPARTEMENT=="SEINE"| DEPARTEMENT=="SEINE-ET-MARNE"| DEPARTEMENT=="SEINE-ET-OISE".`

```
tidf <- subset(tmort,       DEPARTEMENT=="SEINE" |
                            DEPARTEMENT=="SEINE-ET-MARNE" |
                            DEPARTEMENT=="SEINE-ET-OISE",
                            select=c("DEPARTEMENT","Age",
```

```
                                        "Qmort","Survie")
)
```

3. Sélectionner les lignes de `tmort` correspondant aux trois départements d'Île de France et à l'année 1806. Que pensez-vous de l'air de Paris ?

Solution: Quand on veut sélectionner à la fois sur l'année et le département, il faut veiller sur la manière dont on évalue l'expression : quel est l'opérateur prioritaire ? Et ou OU ? En fait c'est la conjonction. Si on écrit
`ANNEE==1856 &DEPARTEMENT=="LOIRE"| DEPARTEMENT=="ALLIER"`
on choisit les lignes correspondant soit à `ANNEE==1856 &DEPARTEMENT=="LOIRE"`
soit à `DEPARTEMENT=="ALLIER"`, ce n'est pas forcément ce que l'on cherche. Pour éviter les mauvaises surprises, utilisez des parenthèses.

```
tidf1806 <- subset(tmort, ANNEE==1806 & (
                          DEPARTEMENT=="SEINE" |
                          DEPARTEMENT=="SEINE-ET-MARNE" |
                          DEPARTEMENT=="SEINE-ET-OISE"),
                          select=c("DEPARTEMENT","Age",
                                   "Qmort","Survie")
)
```

On peut chercher à visualiser ces quotients de mortalité en affichant des `o` pour la `SEINE`, des + pour la `SEINE-ET-OISE`, des `x` pour la `SEINE-ET-MARNE`.

```
subset(tidf1806,DEPARTEMENT=="SEINE",select=c("Age","Qmort"))
plot(subset(tidf1806,DEPARTEMENT=="SEINE",
            select=c("Age","Qmort")), pch="o")
points(subset(tidf1806,DEPARTEMENT=="SEINE-ET-MARNE",
            select=c("Age","Qmort")), pch="x")
points(subset(tidf1806,DEPARTEMENT=="SEINE-ET-OISE",
            select=c("Age","Qmort")), pch="+")
```

On constate qu'à tous les âges, le quotient de mortalité est supérieur dans le département urbain de la Seine. Les « courbes » ont toutes les trois la même allure. Une mortalité juvénile très importante, une mortalité minimale entre 15 et 20 ans. Un quotient de mortalité important qui croît rapidement au delà de 45 ans.

Notez l'usage de **plot** puis de **points**.

Paris, comme les grandes villes de l'époque est un mouroir.

4. Sélectionner les lignes correspondant aux départements et aux années où le quotient de mortalité juvénile est supérieur à 30%.

Solution:

```
bads<- subset(tmort,Age==0 & Qmort>.3,select=c("DEPARTEMENT","ANNEE"))
```

On récupère une **data.frame** formé par deux colonnes et autant de lignes qu'il y a de couples `DEPARTEMENT`,`ANNEE` où la mortalité juvénile dépasse 30%

5. En utilisant l'argument `select` de la fonction **subset**, donner la liste des départements où le quotient de mortalité juvénile est supérieur à 30% au moins une fois dans le siècle. On ne veut pas de doublons dans cette liste !

Solution:

```
unique(sort(bads$DEPARTEMENT))
```

6. Construire une matrice dont les lignes correspondent aux années de recensement et les colonnes aux couples (année,département) pour lesquels la mortalité juvénile est supérieure à 30%.

Solution: Ceci n'est pas vraiment une bonne idée. Pour compter le nombre de mauvais départements par année, le plus simple est d'utiliser la commande importante **aggregate**. L'objectif est pour chaque valeur possible de `ANNEE` dans le **data.frame** `bads` (en fait, tous les recensements jusqu'à 1896) de constituer un vecteur des noms des départements à forte mortalité juvénile cette année là, et de compter le nombre d'éléments de ce vecteur grâce à **length**. .

```
res<-aggregate(bads$DEPARTEMENT,by=list(bads$ANNEE),FUN="length")
```

aggregate, dans sa forme la plus simple, a trois arguments : un premier vecteur qui est ici `bads$DEPARTEMENT`, sur lequel on veut effectuer des regroupements et calculer des statistiques ; une liste de vecteurs qui vont guider les regroupements, ces vecteurs doivent tous être de même longueur que le vecteur sur lequel on effectue les regroupements ; enfin un argument nommé `FUN` auquel on affecte une chaîne de caractères qui désigne la fonction à appliquer aux vecteurs formés par regroupement, ici c'est a fonction **length**().

`res` est un **data.frame**. Les colonnes de ce **data.frame** correspondent aux vecteurs utilisés pour guider le regroupement, elles sont nommées automatiquement `Group.1`, `Groupe.2`, ... , enfin une colonne nommée `x` contient le resultat de l'appel de la fonction affectée à `FUN` sur chaque regroupement. Pour rendre les objets plus lisibles, on peut renommer ces colonnes.

```
names(res) <-  c("ANNEE","NbreDepartements")
```

Pour un graphique très sommaire **plot**(`res`). Que pensez-vous de

```
 barplot(res[[2]],
         names.arg=res[[1]],
         legend=c("Nombre de departements dont\n
                   le Quotient de mortalite juvenile\n
                   depasse 30%")
)
```

7. Calculer pour chaque année de recensement le nombre de départements où la mortalité juvénile dépasse 30% à partir de la matrice construite à la question précédente.

Solution: Voir solution de la question précédente

8. Montrer sur un graphique l'évolution du nombre de départements où la mortalité juvénile est supérieure à 30%.

Solution: Voir solution de la question précédente

Calcul de la fonction de survie en une année en un département

On veut calculer la fonction de survie en 1806 dans le département de la `SEINE`. Utiliser le sous-tableau construit dans la section précédente.

1. En utilisant les quotients de mortalité et les formules données en cours, calculer la proportion d'individus survivant au delà de leur cinquième, dixième, quinzième, ... anniversaire.

Solution: Si $q_0, q_5, \ldots, q_{85}$ désignent les quotients de mortalité pour les différentes classes d'âge, la fonction de survie $\bar{F}$ qui s'en déduit est :

$$\bar{F}(5i) = \prod_{j=0}^{i-1}(1 - q_{5j}) \quad \text{et } \bar{F}(0) = 1\,.$$

On étend brutalement (ce n'est pas la seule manière possible) $\bar{F}(5i + k) = \bar{F}(i)$ pour $0 \leq k < 5$.

```
format(tseine1806,digits=2)
aux <- cumprod(1-tseine1806$Qmort)
aux <- c(1,aux[1:(length(aux)-1)])
tseine1806riche <- cbind(tseine1806, aux)
names(tseine1806riche) <- c(names(tseine1806),"pSurvie")
format(tseine1806riche,digits=2)
```

2. Calculer l'espérance de vie

Solution: Si X est une variable (aléatoire) à valeur dans N, et si la loi de X admet pour fonction de répartition $F = 1 - \bar{F}$, alors

$$\mathbf{E}[X] = \sum_{i\in\mathbb{N}} \bar{F}(i)\,.$$

Ici

$$\sum_{i\in\mathbb{N}} \bar{F}(i) = \sum_{i\in\mathbb{N}} \sum_{k:\ 0\leq k<5} \bar{F}(5i + k) = 5\sum_{i\in\mathbb{N}} \bar{F}(5i)\,.$$

Il suffit donc d'invoquer la fonction **sum** pour calculer l'espérance de vie à la naissance (ou du moins une approximation car nous disposons seulement d'une version grossière de la fonction de survie).

```
# Calcul de l'espérance de vie à la naissance
# C'est une surestimation (assez grossière)
EdVSeine1806 <- 5 * sum(tseine1806$pSurvie)
```

Fusion-jointure de tableaux

Nous voulons mener le calcul précédent pour toutes les années et tous les département. Parce qu'il ne faut pas trop sucharger les machines, on pourra au préalable sélectionner l'année 1806.

1. A l'aide de la fonction **merge**, former un tableau **data.frame** dont chaque ligne correspond à un département, une année et deux âges avec les quotients de mortalité associé. *Suggestion* : utiliser **merge** et fusionner le tableau `tmort` avec lui même (!!!) sur les colonnes `ANNEE` et `DEPARTEMENT`. Puis sélectionner les lignes où `Age.y` < `Age.x`, ces lignes interviendront dans le calcul des fonctions de survie.

Solution: La fonction **merge** permet de composer de nouveaux **data.frames** à partir de **data.frame** en assemblant de manière *cohérente* les lignes d'un **data.frame** avec celles d'un autre (éventuellement comme c'est le cas ici, avec une copie de lui même).

Ici nous voulons calculer la fonction de survie, puis l'espérance de vie à la naissance pour chaque département et chaque année. Nous allons d'abord former un nouveau **data.frame** qui, pour chaque `ANNÉE`, `DEPARTEMENT`, `Age.x`, `Age.y` contiendra le quotient de mortalité à l'age `Age.y` si `Age.x` > `Age.y`. On procède en deux temps, car **merge** n'offre que des possibilités limités (en jargon bases de données, **merge** ne permet que des équi-jointures et pas de θ-jointures)

```
# construction d'un tableau
# avec ANNEE, DEPARTEMENT, AGE.x, AGE.y, Qmort.y
# où  AGE.x >= AGE.y
#
tm <-merge(x=tmort,y=tmort,by.x=c("DEPARTEMENT","ANNEE"),
                    by.y=c("DEPARTEMENT","ANNEE"))
#
# sélection des lignes qui rentrent dans le calcul de
# la fonction de survie à Age.x
#
tms<-subset(tm,Age.x>Age.y,
               select=c("DEPARTEMENT","ANNEE","Age.x","Qmort.y"))
```

2. Calculer la fonction de survie en chaque âge, chaque année de recensement, en chaque département. On pourra utiliser la fonction **aggregate**, on aggrège sur l'âge, le département et l'année, la fonction à appliquer à `1-T$Qmort` est **prod** (`T` est le nom du **data.frame** ontenu à la question précédente). Vous devez obetnir un **data.frame** dont les colonnes sont l'âge, le département, l'année et le taux de survie correspondant.

Solution:

```
#
# Le calcul des fonctions de survie est une aggrégation.
# Pour chaque couple  DEPARTEMENT,ANNEE on veut faire le produit
# des variables Qmort.y
#
survies<-aggregate(1-tms$Qmort.y,
                    by=list(Age=tms$Age,
                            Dep=tms$DEPARTEMENT,
                            An=tms$ANNEE),
                    FUN="prod")
names(survies)
#
```