S1_5_Autour_de_la_modélisation

Table des matières

1 Introduction **2**

2 Codages des variables explicatives quantitatives **2**
2.A Problématique . . . 2
2.B Exemple R sur les modèles additifs généralisés (GAM) . . . 2
2.B.1 Modèle additif généralisé (GAM) avec splines pour une variable explicative quantitative . . . 3
2.B.2 Modèle additif généralisé (GAM) avec splines pour une régression logistique . . . 5
2.B.2.1 Découpage en classes . . . 6
2.B.2.2 Polynômes orthogonaux . . . 7

3 Codage des variables explicatives catégorielles **9**
3.A Exemple : ABO . . . 10
3.B Exemple R . . . 11
3.B.1 Avec une variable codée 0/1 . . . 11
3.B.2 Avec une variable codée -1/1 . . . 13

4 Choix des variables explicatives **16**
4.A Principe . . . 16

5 À propos des termes d'interaction **17**

6 Données manquantes **18**
6.A Types de données manquantes . . . 18
6.B Gestion des données manquantes . . . 18
6.C Exemple R avec le package `mice` . . . 18

7 Bootstrap **21**
7.A Principe . . . 21
7.B Conditions à respecter pour utiliser le bootstrap : . . . 22
7.C Diagnostics et avantages . . . 23
7.D Exemple R . . . 24

1 Introduction

Modèle : "simplifie" la réalité.

1. comment coder les variables explicatives (quantitatives et catégorielles) ?
2. interactions entre variables explicatives.
3. sélection automatique de variables explicatives.
4. données manquantes.
5. techniques de ré-échantillonnage.

2 Codages des variables explicatives quantitatives

2.A Problématique

On considère le modèle linéaire suivant :

$$Y = \alpha_0 + \alpha_1 X_1 + \cdots + \alpha_p X_p + \epsilon$$

- Y = mesure des capacités cognitives d'un sujet
- X1 = âge du sujet (en années)

Comme ce modèle fait l'hypothèse d'une relation linéaire, il fait l'hypothèse que la variation entre 60 et 80 ans est la même qu'entre 20 et 40 ans.

On peut apprécier la forme de la relation entre la variable a expliquer Y (NB : dans le cas d'une régression logistique, il s'agit de logit$(P(Y = 1))$) et la variable explicative X_1 en utilisant des méthodes non paramétriques (loess, splines, etc.).

Si la relation n'est pas linéaire, on peut envisager plusieurs solutions :

1. Convert X_i en variable catégorielle (ex. quartiles, quintiles, etc.) : perte d'information, perte de puissance mais interprétation simple des résultats.
2. Ajouter des modèles polynomiaux en X_i (ex. X_i et X_i^2) : permet de modéliser des relations non linéaires simples (ex. relations quadratiques) mais interprétation complexe
3. Recourir à un "modèle additif généralisé" (GAM) : permet de modéliser des relations non linéaires complexes (ex. splines, loess, etc.) mais interprétation complexe

2.B Exemple R sur les modèles additifs généralisés (GAM)

Les modèles additifs généralisés (ou GAM, pour generalized additive models) offrent une méthode flexible pour décrire une relation non-linéaire entre des prédicteurs et une variable réponse.

La moyene de Y dépend de la somme de fonctions des variables explicatives, mais il n'y a pas d'hypothèse de linéarité.

Dans ce cadre, on laisse des "degrés de liberté" aux courbes de régression (et à la relation entre Y et X_i) pour qu'elles puissent s'adapter aux données.

Par exemple : si un seul degré de liberté, la relation est linéaire.

Avec deux degrés de liberté, la relation est quadratique, etc.

En gros :

- modèle linéaire généralisé = somme des lignes = sommes des courbes à degrés de liberté 1
- modèle additif généralisé = somme des courbes = sommes des courbes à degrés de liberté k

2.B.1 Modèle additif généralisé (GAM) avec splines pour une variable explicative quantitative

On considère la durée de l'entretien `smp$duree.interv` comme variable à expliquer et une liste de variables explicatives parmi lesquelles l'âge `smp$age`.

On trace un *spline* représentant la forme de la relation entre ces deux variables.

```
mod <- gam(duree.interv~s(age, k=20, fx=TRUE),data=smp)
plot(mod)
```

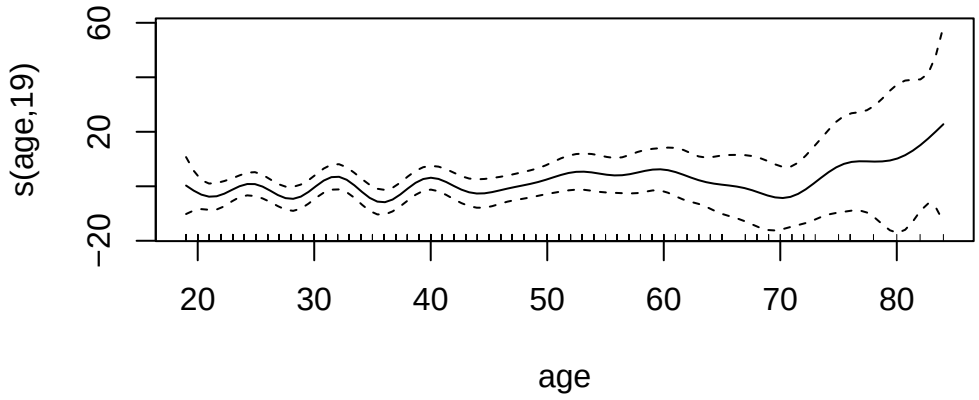

Figure 1: Spline représentant la forme de la relation entre la durée de l'entretien et l'âge des détenus.

On peut augmenter le nombre de paramètre du *spline* en augmentant le paramètre `k` (nombre de noeuds).

Si on laisse par défaut, `mgcv` choisit automatiquement une valeur de `k`

Pour savoir ce qu'il a choisi :

```
mod <- gam(duree.interv~s(age),data=smp)
summary(mod)
```

```
Family: gaussian
Link function: identity

Formula:
duree.interv ~ s(age)

Parametric coefficients:
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  62.2016     0.7066   88.02   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
       edf Ref.df     F p-value
s(age)   1      1 5.911  0.0153 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.00657   Deviance explained = 0.79%
GCV = 372.52  Scale est. = 371.51    n = 744
```

interprétation :

- edf = 1 : le *spline* est linéaire (1 degré de liberté)
- p-value = 0.0153 : la relation entre âge et durée de l'entretien est significative
- ça signifie que `mgcv` a choisi `k=2` (car pour un *spline* cubique, le degré de liberté est égal à `k-1`)
- Donc le *spline* est linéaire (car 2-1=1)
- donc la relation entre âge et durée de l'entretien est modélisée comme linéaire
- On peut vérifier en forçant `k=2` :

```
mod1 <- gam(duree.interv~s(age,k=2,fx=TRUE),data=smp)
```

```
Warning in smooth.construct.tp.smooth.spec(object, dk$data, dk$knots): basis dimension, k, incr
```

```
plot(mod1)
```

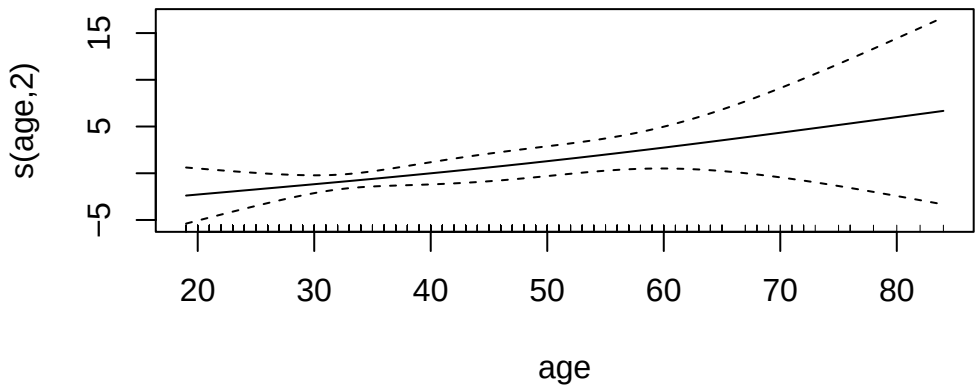

Figure 2: Spline représentant la forme de la relation entre la durée de l'entretien et l'âge des détenus, avec k=2.

Mais on peut forcer une valeur de `k` différente : par exemple ici on force `k=4`

```
mod2 <- gam(duree.interv~s(age,k=4,fx=TRUE),data=smp)
plot(mod2)
```

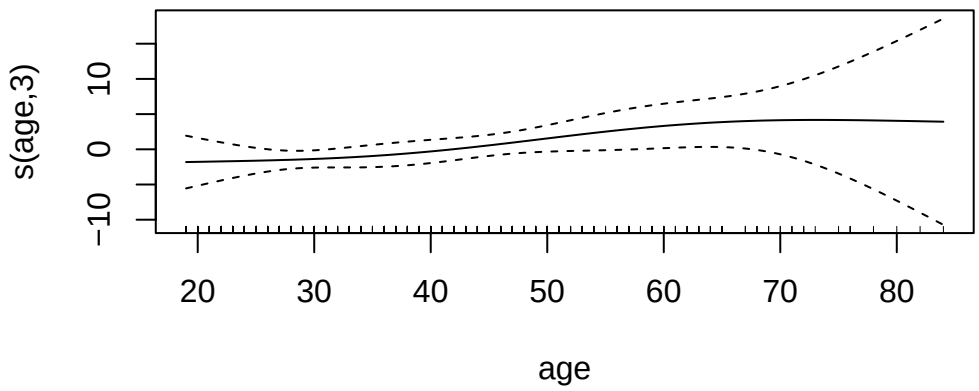

Figure 3: Spline représentant la forme de la relation entre la durée de l'entretien et l'âge des détenus, avec k=4.

En gros : plus `k` est grand, plus le *spline* est flexible.

2.B.2 Modèle additif généralisé (GAM) avec splines pour une régression logistique

On essaie d'expliquer l'abus de substance `smp$abus.subst` (0/1) en fonction de l'âge `smp$age`.

```
mod_logit <- gam(abus.subst~s(age), data=smp, family=binomial)
plot(mod_logit)
```

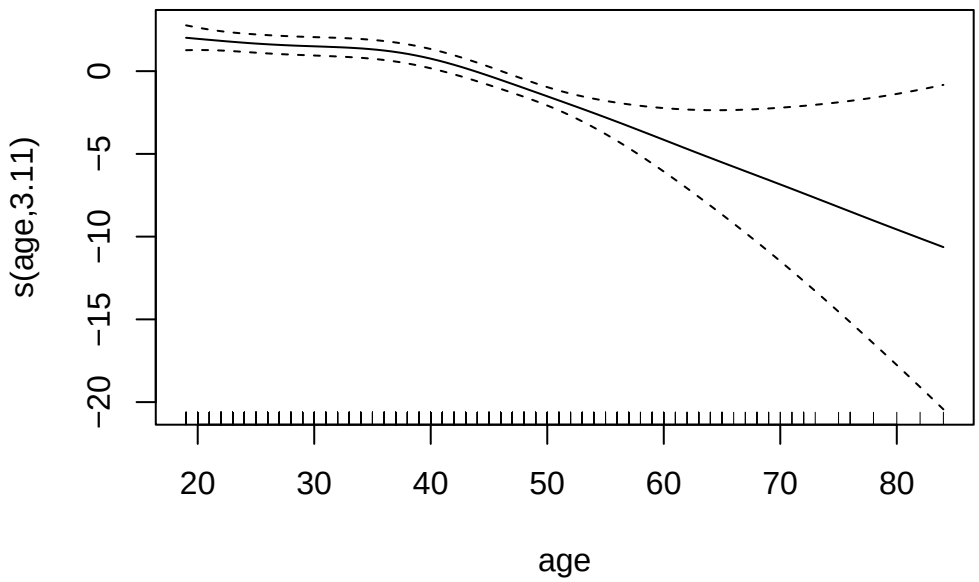

Figure 4: Spline représentant la forme de la relation entre le logit de la probabilité de présenter un abus de substance et l'âge des détenus.

La spline pointe un écart statistiquement significatif à la linéarité l'évolution est stable jusqu'à 40 ans puis semble s'infléchir.

Le nombre de degrés de liberté effectif est de 3.114 donc supérieur à 1. Il est codé "edf" dans le tableau et est écrit sur l'axe des y du graphique.

Et donc là : que faire ?

- Si l'âge est un facteur de confusion accessoire d'intérêt marginal : on le laisse comme ça
- Si l'âge est un facteur de confusion important : il faut adopter une autre stratégie
  - On peut le recouper en classes : plus l'échantillon est grand, plus on peut faire de classes

2.B.2.1 Découpage en classes La fonction `cut()` permet de découper une variable quantitative en classes.

```
smp$age.4f <- cut(
    smp$age,
    breaks=c(-Inf, 25, 35, 45, Inf),
    labels=c("<25","25-35","35-45",">45"))
table(smp$age.4f, useNA="ifany")
```

```
  <25 25-35 35-45   >45  <NA>
  140   217   202   238     2
```

Les effectifs par classe sont suffisants pour inclure cette variable dans un modèle de régression logistique.

On peut ensuite l'inclure dans un modèle de régression logistique classique :

```
mod_logit2 <- glm(
    abus.subst ~
    age.4f +
    factor(type.centre),
    data=smp,
    family="binomial")
summary(mod_logit2)
```

```
Call:
glm(formula = abus.subst ~ age.4f + factor(type.centre), family = "binomial",
    data = smp)

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)           -1.0570     0.3886  -2.720 0.006525 **
age.4f25-35           -0.2397     0.2274  -1.054 0.291920
age.4f35-45           -0.8364     0.2405  -3.478 0.000505 ***
age.4f>45             -3.3742     0.4537  -7.436 1.04e-13 ***
factor(type.centre)2   0.6805     0.3781   1.800 0.071901 .
```

```
factor(type.centre)3   1.0808     0.3554   3.041 0.002355 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 921.27  on 796  degrees of freedom
Residual deviance: 757.98  on 791  degrees of freedom
  (2 observations deleted due to missingness)
AIC: 769.98

Number of Fisher Scoring iterations: 6
```

Représentation en table des résultats :

```
library(gtsummary)
tbl <-
    mod_logit2 %>%
    tbl_regression(exponentiate = TRUE) %>%
    as_gt()
tbl
```

Characteristic	OR	95% CI	p-value
age.4f			
<25	—	—	
25-35	0.79	0.50, 1.23	0.3
35-45	0.43	0.27, 0.69	<0.001
>45	0.03	0.01, 0.08	<0.001
factor(type.centre)			
1	—	—	
2	1.97	0.97, 4.31	0.072
3	2.95	1.52, 6.20	0.002

Abbreviations: CI = Confidence Interval, OR = Odds Ratio

2.B.2.2 Polynômes orthogonaux Le problème avec les polynômes "classiques" (ex. X_i et X_i^2) est que les termes sont corrélés entre eux.

On peut utiliser des polynômes orthogonaux pour éviter ce problème.

La fonction `poly()` permet de créer des polynômes orthogonaux en choisissant des degrés de polynome pour lesquels les termes sont orthogonaux donc non corrélés.

```
dt <- na.omit(smp[,c("abus.subst","age","type.centre")])
mod3 <- glm(
    abus.subst ~
    poly(age, degree = 3) + factor(type.centre),
```

```
    data=dt,
    family="binomial")
```

```
Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

Warning : le processus de convergence est fragile, car les polynômes de degré élevé peuvent causer des problèmes numériques.

Avec un degré 2, ça passe mieux :

```
dt <- na.omit(smp[,c("abus.subst","age","type.centre")])
mod3 <- glm(
    abus.subst ~
    poly(age, degree = 2) + factor(type.centre),
    data=dt, family="binomial")
summary(mod3)
```

```
Call:
glm(formula = abus.subst ~ poly(age, degree = 2) + factor(type.centre),
    family = "binomial", data = dt)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)             -2.8028     0.4042  -6.934 4.10e-12 ***
poly(age, degree = 2)1 -68.6238    10.7444  -6.387 1.69e-10 ***
poly(age, degree = 2)2 -32.6108     7.9009  -4.127 3.67e-05 ***
factor(type.centre)2     0.6178     0.3824   1.616  0.10616
factor(type.centre)3     1.0559     0.3591   2.941  0.00328 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 921.27  on 796  degrees of freedom
Residual deviance: 740.34  on 792  degrees of freedom
AIC: 750.34

Number of Fisher Scoring iterations: 7
```

Le seul truc interprétable sont les p-values associées aux termes polynomiaux (pas les coefficients).

On visualise la relation proposée par le modèle en utilisant la fonction `predict()`.

```
ages=seq(20,50,1)
avgpred <- sapply(
    ages,
    function(age) {
```

```
        dt$age <- age
        mean(predict(mod3, newdata=dt, type="response"))}
        )
plot(ages, avgpred,
    type="l",
    xlim=c(20,50), ylim=c(0,1),
    las=1,
    xlab="Âge", ylab="Prédiction moyenne")
```

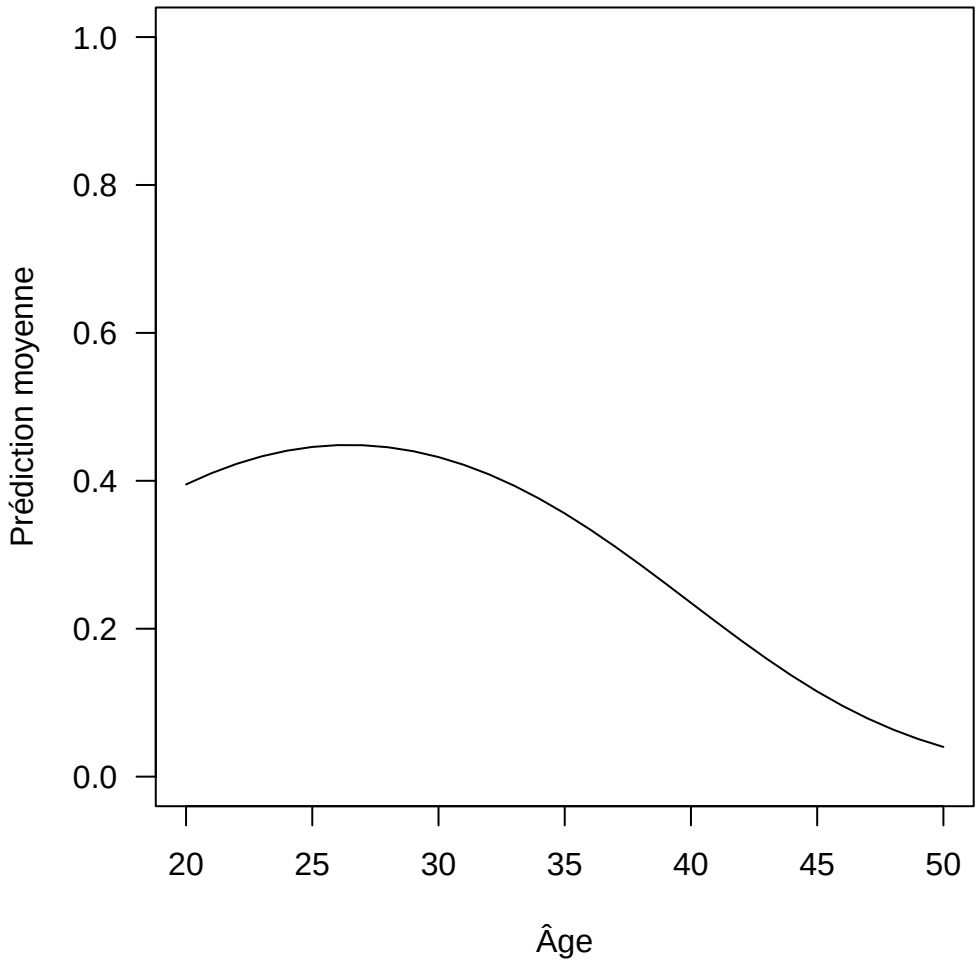

Figure 5: Prédiction moyenne du risque d'abus de substance en fonction de l'âge des détenus, standardisée sur le type de centre.

Cette approche de prédiction moyenne consiste à évaluer le pourcentage de risque d'abus de substance selon le modèle mod3, pour chaque âge et après « standardisation » sur le type de centre (car le type de centre est inclus dans `mod3`).

3 Codage des variables explicatives catégorielles

Dans le modèle linéaire, les variables explicatives catégorielles sont classiquement

- Transformées en variables binaires
- Ou en variables quantitatives discrètes (le problème est que l'écart entre les modalités n'est pas forcément le même)

Dans le cas de variables catégorielles comptant \2 classes : il faut recoder par une série de variables binaires.

3.A Exemple : ABO

4 groupes sanguins : A, B, AB, O

- Approche classique : codage 0/1 pour chaque groupe sanguin (3 variables binaires, une de référence)
  - Groupe A : $X_1 = 1$ et $X_2 = 0$ et $X_3 = 0$
  - Groupe B : $X_1 = 0$ et $X_2 = 1$ et $X_3 = 0$
  - Groupe AB : $X_1 = 0$ et $X_2 = 0$ et $X_3 = 1$
  - Groupe O (référence) : $X_1 = 0$ et $X_2 = 0$ et $X_3 = 0$

En tableau :

Groupe sanguin	X_1	X_2	X_3
A	1	0	0
B	0	1	0
AB	0	0	1
O (référence)	0	0	0

Équation du modèle linéaire :

$$Y = a_0 + a_1 Z_1 + a_2 Z_2 + ... + a_p Z_p + b_1 X_1 + b_2 X_2 + b_3 X_3 + \epsilon$$

Dans ce modèle : le coefficient b_1 sera interprété comme la différence moyenne de Y entre le groupe A et le groupe O (référence) à $Z_1, Z_2, ..., Z_p$ fixés.

- D'autres codages sont possibles : notamment "-1 / 1"
  - Groupe A : $X_1 = 1$ et $X_2 = X_3 = 0$
  - Groupe B : $X_2 = 1$ et $X_1 = X_3 = 0$
  - Groupe AB : $X_3 = 1$ et $X_1 = X_2 = 0$
  - Groupe O (référence) : $X_1 = X_2 = X_3 = -1$

Équation du modèle linéaire :

$$Y = a_0 + a_1 Z_1 + a_2 Z_2 + ... + a_p Z_p + b_1 X_1 + b_2 X_2 + b_3 X_3 + \epsilon$$

Dans ce modèle : le coefficient b_1 sera interprété comme la différence entre l'effet observé dans le groupe "A" et la moyenne non pondérée des effets observés dans les autres groupes (B, AB et O) à $Z_1, Z_2, ..., Z_p$ fixés.

Moyenne non pondérée : il s'agit de la moyenne des moyennes de chaque groupe, calculée sans tenir compte des effectifs respectifs de ces groupes (chaque groupe a le même poids dans le calcul de la moyenne globale).

3.B Exemple R

3.B.1 Avec une variable codée 0/1

variable catégorielle : `smp$profession` (8 modalités)

```
str(smp$profession)
```

```
 Factor w/ 8 levels "agriculteur",..: 7 NA 4 6 8 6 7 2 6 6 ...
```

Pour savoir comment R code cette variable dans un modèle de régression linéaire : on utilise la fonction `contrasts()` :

```
contrasts(smp$profession)
```

```
              commerçant cadre intermédiaire employé ouvrier autre sans.emploi
agriculteur            0     0             0       0       0     0           0
commerçant             1     0             0       0       0     0           0
cadre                  0     1             0       0       0     0           0
intermédiaire          0     0             1       0       0     0           0
employé                0     0             0       1       0     0           0
ouvrier                0     0             0       0       1     0           0
autre                  0     0             0       0       0     1           0
sans.emploi            0     0             0       0       0     0           1
```

Par défaut, R utilise un codage "0/1" avec la première modalité comme référence (ici "Agriculteurs").

Pour changer la modalité de référence, on peut utiliser la fonction `relevel()` :

```
smp$profession <- relevel(smp$profession, ref="ouvrier")
contrasts(smp$profession)
```

```
              agriculteur commerçant cadre intermédiaire employé autre
ouvrier                 0          0     0             0       0     0
agriculteur             1          0     0             0       0     0
commerçant              0          1     0             0       0     0
cadre                   0          0     1             0       0     0
intermédiaire           0          0     0             1       0     0
employé                 0          0     0             0       1     0
autre                   0          0     0             0       0     1
sans.emploi             0          0     0             0       0     0
```

```
              sans.emploi
ouvrier                 0
agriculteur             0
commerçant              0
cadre                   0
intermédiaire           0
employé                 0
autre                   0
sans.emploi             1
```

Ici, la modalité de référence est “ouvrier”.

On cherche à expliquer la variable “haut risque suicidaire” `smp$hr.suicide` (0/1) en fonction de la profession `smp$profession` + autres variables de confusion.

```r
mod <- glm(
    hr.suicide~
    abus.enfant+discipline+duree.peine+profession+factor(type.centre),
    data=smp,
    family="binomial")
summary(mod)
```

```
Call:
glm(formula = hr.suicide ~ abus.enfant + discipline + duree.peine +
    profession + factor(type.centre), family = "binomial", data = smp)

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)             -0.28420    0.78455  -0.362  0.71717
abus.enfant              0.62004    0.23521   2.636  0.00839 **
discipline               0.39596    0.24881   1.591  0.11151
duree.peine             -0.32932    0.14853  -2.217  0.02661 *
professionagriculteur    2.13259    1.25662   1.697  0.08968 .
professioncommerçant    -0.17238    0.40182  -0.429  0.66793
professioncadre         -0.77484    0.77783  -0.996  0.31917
professionintermédiaire -0.92553    0.64236  -1.441  0.14963
professionemployé       -0.41118    0.36301  -1.133  0.25734
professionautre         -2.03833    1.05821  -1.926  0.05408 .
professionsans.emploi    0.27877    0.26911   1.036  0.30024
factor(type.centre)2    -0.06536    0.34254  -0.191  0.84868
factor(type.centre)3     0.31189    0.37935   0.822  0.41099
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 554.57  on 546  degrees of freedom
Residual deviance: 512.92  on 534  degrees of freedom
```

```
  (252 observations deleted due to missingness)
AIC: 538.92

Number of Fisher Scoring iterations: 5
```

Il y a bien une estimation pour 7 modalité de la variable `profession`, sauf pour la modalité de référence "ouvrier".

Interprétation : chaque coefficient de la variable `profession` est interprété par rapport à la modalité de référence "ouvrier".

Si on veut tester l'effet global de la variable `profession` dans le modèle, on utilise `drop1`:

```
drop1(mod,.~.,test="Chisq")
```

```
Single term deletions

Model:
hr.suicide ~ abus.enfant + discipline + duree.peine + profession +
    factor(type.centre)
                    Df Deviance    AIC     LRT Pr(>Chi)
<none>                   512.92 538.92
abus.enfant          1   519.72 543.72  6.7974 0.009129 **
discipline           1   515.40 539.40  2.4807 0.115250
duree.peine          1   517.84 541.84  4.9206 0.026538 *
profession           7   530.83 542.83 17.9130 0.012369 *
factor(type.centre)  2   514.88 536.88  1.9568 0.375911
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

3.B.2 Avec une variable codée -1/1

On peut utiliser la fonction `contrasts<-` pour changer le codage par défaut de R.

```
contrasts(smp$profession) <- contr.sum
contrasts(smp$profession)
```

```
             [,1] [,2] [,3] [,4] [,5] [,6] [,7]
ouvrier         1    0    0    0    0    0    0
agriculteur     0    1    0    0    0    0    0
commerçant      0    0    1    0    0    0    0
cadre           0    0    0    1    0    0    0
intermédiaire   0    0    0    0    1    0    0
employé         0    0    0    0    0    1    0
autre           0    0    0    0    0    0    1
sans.emploi    -1   -1   -1   -1   -1   -1   -1
```

Problème : le codage en matrice [,1] [,2].. n'est pas très lisible.

Renommer :

```
colnames(contrasts(smp$profession)) <-
 ↪  c("ouvrier","agriculteur","commerçant","cadre","intermédiaire","employé","autre")
contrasts(smp$profession)
```

```
              ouvrier agriculteur commerçant cadre intermédiaire employé autre
ouvrier             1           0          0     0             0       0     0
agriculteur         0           1          0     0             0       0     0
commerçant          0           0          1     0             0       0     0
cadre               0           0          0     1             0       0     0
intermédiaire       0           0          0     0             1       0     0
employé             0           0          0     0             0       1     0
autre               0           0          0     0             0       0     1
sans.emploi        -1          -1         -1    -1            -1      -1    -1
```

Modèle de régression logistique avec ce nouveau codage :

```
mod <- glm(hr.suicide ~
        abus.enfant + discipline + duree.peine + profession + factor(type.centre),
 ↪
        data=smp,
        family="binomial")
summary(mod)
```

```
Call:
glm(formula = hr.suicide ~ abus.enfant + discipline + duree.peine +
    profession + factor(type.centre), family = "binomial", data = smp)

Coefficients:
                         Estimate Std. Error z value Pr(>|z|)
(Intercept)              -0.52306    0.82744  -0.632  0.52729
abus.enfant               0.62004    0.23521   2.636  0.00839 **
discipline                0.39596    0.24881   1.591  0.11151
duree.peine              -0.32932    0.14853  -2.217  0.02661 *
professionouvrier         0.23886    0.29659   0.805  0.42060
professionagriculteur     2.37146    1.10197   2.152  0.03140 *
professioncommerçant      0.06649    0.39303   0.169  0.86567
professioncadre          -0.53598    0.69640  -0.770  0.44151
professionintermédiaire -0.68667    0.58349  -1.177  0.23926
professionemployé        -0.17232    0.36126  -0.477  0.63336
professionautre          -1.79947    0.93733  -1.920  0.05489 .
factor(type.centre)2     -0.06536    0.34254  -0.191  0.84868
factor(type.centre)3      0.31189    0.37935   0.822  0.41099
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 554.57  on 546  degrees of freedom
Residual deviance: 512.92  on 534  degrees of freedom
  (252 observations deleted due to missingness)
AIC: 538.92

Number of Fisher Scoring iterations: 5
```

Les log odds-ratios correspondants aux modalités de la variable « profession » sont maintenant relatifs à la moyenne des effets des 8 modalités

Si on veut voir “sans emploi” aussi :

```
estimable(mod,c(0,0,0,0,-1,-1,-1,-1,-1,-1,-1,0,0),conf.int=0.95)
```

```
                                    Estimate Std. Error X^2 value DF Pr(>|X^2|)
(0 0 0 0 -1 -1 -1 -1 -1 -1 -1 0 0) 0.517635  0.2952967  3.072771  1 0.07961365
                                      Lower.CI Upper.CI
(0 0 0 0 -1 -1 -1 -1 -1 -1 -1 0 0) -0.06822514 1.103495
```

4 Choix des variables explicatives

4.A Principe

Quelles variables explicatives doit-on inclure dans le modèle ?

2 types de modèles :

- Prédictifs : estimer la probabilité de survenue d'une pathologie
- Explicatif : comprendre les principaux déterminants de la survenue d'une maladie

Pour les modèles prédictifs : c'est du machine learning.

Choix des variables pour les modèles explicatifs :

- Approche hypothético-déductive : réfléchir en amont aux variables à inclure
- Graphes acycliques orientés : article
- Analyse bivariée initiale :
  - inclure les variables significatives en analyse bivariée
  - problème : risque de confusion entre variables explicatives
  - et il faut inclure dans tous les cas quelque chose de cliniquement pertinent

NB : L'analyse pas à pas ("stepwise") est fortement déconseillée pour la sélection de variables, en particulier pour les modèles explicatifs. Voici pourquoi :

1. **Biais de sélection et sur-ajustement (Overfitting)** : Le stepwise teste de nombreuses combinaisons et ne garde que celles qui "marchent" le mieux sur l'échantillon donné. Cela a tendance à capturer le bruit aléatoire (le hasard) spécifique à cet échantillon plutôt que la vraie relation biologique ou clinique. Le modèle final performe souvent très bien sur les données d'apprentissage mais mal sur de nouvelles données.
2. **P-values et intervalles de confiance invalides** : Les calculs classiques des p-values et des intervalles de confiance supposent que le modèle a été spécifié *a priori*. Quand on utilise les données pour choisir le modèle, on effectue des tests multiples implicites sans correction. En conséquence, les p-values affichées sont artificiellement trop petites (on trouve trop de résultats "significatifs") et les intervalles de confiance sont trop étroits (donnant une fausse impression de précision).
3. **Instabilité du modèle** : Le retrait ou l'ajout de quelques observations peut changer radicalement la liste des variables sélectionnées par l'algorithme.
4. **Biais des coefficients** : Les coefficients des variables retenues sont souvent surestimés (biais loin de zéro), car seules les variables ayant par hasard un effet fort dans cet échantillon passent le filtre de sélection.

Recommandation : Privilégier une sélection basée sur la connaissance du domaine (littérature, plausibilité biologique, DAGs) plutôt que sur des algorithmes automatiques basés uniquement sur la significativité statistique.

5 À propos des termes d'interaction

Dans un modèle linéaire / linéaire généralisé / modèle de Cox, les variables explicatives ont des effets qui **s'additionnent** avant transformation par la réciproque de la fonction de lien.

- Dans le cas du modèle linéaire : les effets s'additionnent directement
- Dans le cas du modèle logistique : les log-odds s'additionnent, donc les odds-ratios se multiplient
- Dans le cas du modèle de Cox : les log-hazards s'additionnent, donc les hazard-ratios se multiplient

Dans le cas d'une intercation entre deux variables explicatives, l'effet d'une variable dépend de la valeur de l'autre variable.

Par exemple, l'effet de l'alcool sur le cancer du larynx est potentialisé par le tabac.

Dans un modèle : on peut inclure un terme d'interaction entre deux variables explicatives.

La formule serait :

$$Y = a_0 + a_1 X_1 + a_2 X_2 + a_3 (X_1 * X_2) + \epsilon$$

où $X_1 * X_2$ est le terme d'interaction entre les variables explicatives X_1 et X_2.

L'interprétation des coefficients devient plus complexe en présence d'interactions.

Par exemple, dans un modèle de régression linéaire avec interaction entre deux variables explicatives X_1 et X_2, le coefficient a_1 représente l'effet de X_1 sur Y lorsque X_2 est égal à zéro, et vice versa pour a_2.

L'effet combiné des deux variables sur Y est donné par la somme des coefficients a_1, a_2 et a_3 multiplié par les valeurs respectives de X_1 et X_2.

Pour analyser l'effet de X_1 sur Y, il peut être suggéré de ne pas inclure le terme d'interaction si l'objectif principal est d'estimer l'effet moyen de X_1 sur Y.

6 Données manquantes

6.A Types de données manquantes

- Manquantes complètement aléatoires (MCAR) : la probabilité qu'une donnée soit manquante est indépendante de la valeur de la donnée elle-même et des autres variables observées.
  - Exemple : un questionnaire perdu par la poste, ou une erreur de saisie aléatoire.
- Manquantes aléatoires (MAR) : la probabilité qu'une donnée soit manquante dépend des autres variables observées, mais pas de la valeur de la donnée elle-même.
  - Exemple : les patients plus âgés sont moins susceptibles de répondre à certaines questions, mais parmi les patients du même âge, la probabilité de réponse ne dépend pas de la valeur de la donnée manquante.
- Manquantes non aléatoires (MNAR) : la probabilité qu'une donnée soit manquante dépend de la valeur de la donnée elle-même, même après avoir pris en compte les autres variables observées.
  - Exemple : les patients avec des symptômes plus graves sont moins susceptibles de répondre à une question sur leur état de santé.

6.B Gestion des données manquantes

1. Les décrire !
2. Les imputer : par la médiane, la moyenne, le mode… Ou faire de l'imputation multiple avec le package `mice`.

6.C Exemple R avec le package `mice`

```r
# sélection du jeu de données avec les variables d'intérêt
smp.imp <- smp[,c(2,7:93)]
# imputation multiple avec le package mice. par défaut : 5 imputations
smp.mice <- mice(smp.imp, seed=1)
```

```
 iter imp variable
  1   1  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  1   2  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  1   3  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  1   4  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  1   5  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  2   1  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  2   2  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  2   3  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  2   4  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  2   5  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  3   1  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
```

```
  3   2  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  3   3  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  3   4  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  3   5  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  4   1  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  4   2  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  4   3  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  4   4  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  4   5  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  5   1  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  5   2  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  5   3  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  5   4  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
  5   5  gravite  nb.enfants  scolarite  profession.b  annees.prison  preventive  crime  duree.
```

```r
# extraction de la première imputation
smp.imp.1 <- complete(smp.mice,1)
table(smp$determination,smp.imp.1$determination, useNA="ifany")
```

```
        0   1
  0   567   0
  1     0 147
  <NA>  71  14
```

Si on veut estimer le modèle avec les données imputées, il faut utiliser la fonction `pool()` :

```r
mod.imp <- with(
        smp.mice,
                glm(
                    hr.suicide ~
                    gravite + atcd.suicide + depression+ atcd.prison
                    + contre.pers + abus.subst + determination +
                     ↪  factor(smp$type.centre),
                    family="binomial"
                    )
                )
summary(pool(mod.imp),type="all")[,c(1,3,4,7,10)]
```

```
                    term    estimate std.error      p.value        fmi
1            (Intercept) -6.91754466 0.6601498 5.418862e-24 0.02341323
2                gravite  0.68760625 0.1086462 1.447867e-09 0.12281306
3            atcd.suicide  1.62561164 0.2623188 7.823907e-09 0.17519303
4              depression  1.38815347 0.2754109 2.062645e-06 0.19918619
5             atcd.prison  0.58632019 0.2583099 2.398243e-02 0.10003463
6            contre.pers -0.08098422 0.2616474 7.570826e-01 0.06867361
7             abus.subst -0.13431550 0.2780758 6.294464e-01 0.09650375
```

```
8             determination  0.92304111 0.2930070 2.559647e-03 0.27109243
9  factor(smp$type.centre)2  0.65096200 0.4071465 1.102619e-01 0.00780281
10 factor(smp$type.centre)3  0.84577731 0.3817517 2.720780e-02 0.06065212
```

La fonction `with(glm(...))` estime le même modèle logistique sur chacun des 5 jeux de données imputées contenus dans smp.mice.

La fonction `pool()` fait une synthèse des 5 séries de résultats.

Parmi les résutats disponibles il y a : les log odds-ratios moyens, leurs erreurs types obtenues à partir de la somme des variances intra-modèles et des variances inter-modèles.

La FMI (Fraction of Missing Information) quantifie, pour chaque variable, la part de la variance dûe aux données manquantes.

7 Bootstrap

7.A Principe

Dans un modèle logistique, les p value et les IC des OR sont calculées à partir des « dérivées partielles secondes de la log vraisemblance », calculs dont la fiabilité est en principe garantie pour des tailles d'échantillons tendant vers l'infini.

Pour des échantillons de taille modérée, on peut utiliser le bootstrap pour estimer empiriquement la distribution des paramètres du modèle.

En gros, le bootstrap consiste à :

- Tirer au hasard avec remise des échantillons de taille n (taille de l'échantillon initial) parmi les n observations disponibles.
- Estimer le modèle sur chacun de ces échantillons bootstrap.
- Répéter l'opération un grand nombre de fois (ex: $k = 1000$ ou 10000).
- Appliquer la méthode des percentiles pour construire des intervalles de confiance empiriques à partir des k estimations obtenues.
  - On classe les k estimations obtenues (ex: les 1000 Odds-Ratios) du plus petit au plus grand.
  - On retire les 2,5% des valeurs les plus basses et les 2,5% des valeurs les plus élevées (les extrêmes).
  - L'intervalle entre la plus petite et la plus grande valeur restante constitue l'intervalle de confiance à 95%.

Cela revient à dire : *"Si je répétais mon étude un grand nombre de fois, dans 95% des cas, mon paramètre tomberait dans cet intervalle"*.

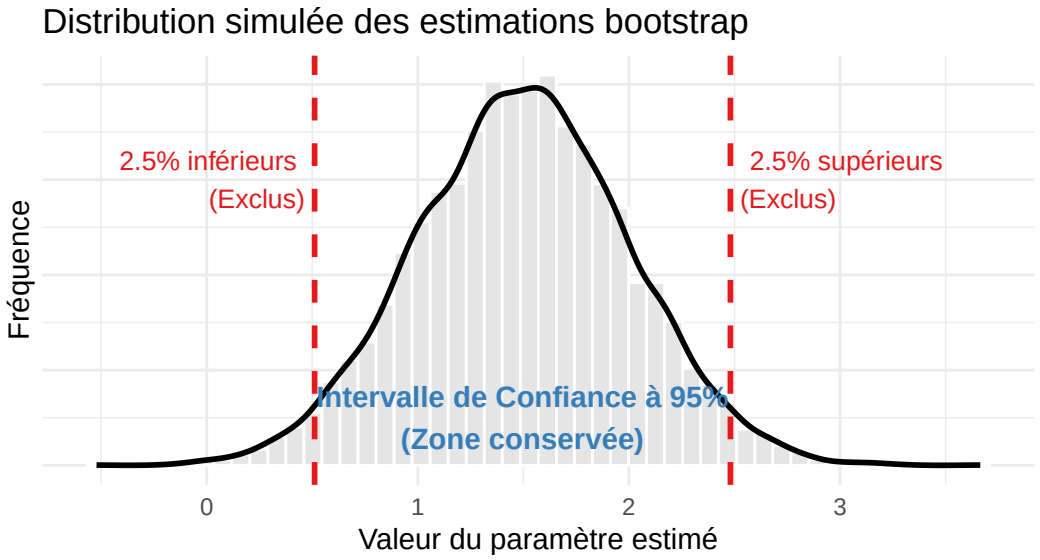

Figure 6: Illustration du principe du Bootstrap (Méthode des percentiles)

Le bootstrap présente l'avantage de ne pas reposer sur des trucs mathématiques impénétrables !

Il repose toutefois sur l'hypothèse que l'échantillon initial est représentatif de la population cible et que les observations sont indépendantes.

Les observations ré-échantillonnées peuvent correspondre à un grappe (*cluster*) de données corrélées entre elles, telles que l'ensemble des mesures répétées chez un même patient ou dans un même centre.

7.B Conditions à respecter pour utiliser le bootstrap :

1. Indépendance des observations ré-échantillonnées
2. Nombre suffisant d'observations dans l'échantillon initial pour que les fluctuations de ré-échantillonnages soient proches des fluctuations d'échantillonnages de la population
3. Fluctuations d'échantillonnages continues (non discrète) : La statistique calculée ne doit pas faire de "sauts" (être discrète) mais varier de façon fluide.

Exemple de fluctuation non continue (discrète) : La Médiane.

Si on calcule la médiane sur des données discrètes (ex: des notes entières de 0 à 10) avec un petit échantillon, la médiane ne pourra prendre que quelques valeurs spécifiques (ex: 2, 2.5, 3).

La distribution bootstrap sera "en peigne" (avec des trous) et non une courbe lisse. Cela rend les intervalles de confiance peu fiables.

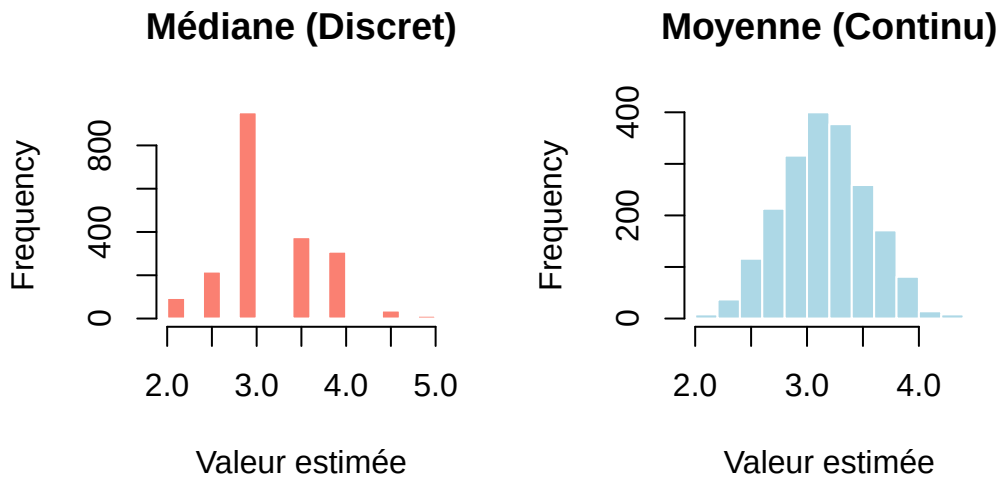

Figure 7: Comparaison : La médiane d'une variable discrète (gauche) produit une distribution bootstrap discontinue (en peigne), contrairement à la moyenne (droite) qui est plus continue.

4. Stabilisation asymptotique des fluctuations d'échantillonnages (c'est à dire que la distribution des fluctuations d'échantillonnages converge vers une distribution limite lorsque la taille de l'échantillon tend vers l'infini, souvent une distribution normale)

Exemple : Convergence vers la normalité.

Si la variable d'origine suit une distribution très asymétrique (ex: loi exponentielle), la distribution bootstrap de la moyenne sera elle-même asymétrique pour de petits échantillons. Elle ne deviendra "Normale" (en cloche) que si la taille de l'échantillon est suffisante.

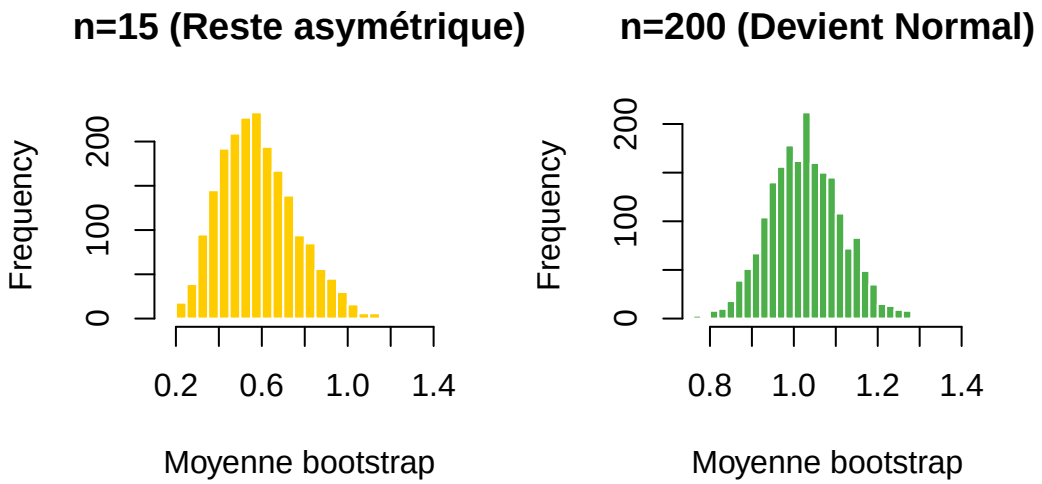

Figure 8: Illustration de la stabilisation asymptotique : Pour une variable asymétrique, la distribution bootstrap de la moyenne ne devient normale que si n est assez grand.

7.C Diagnostics et avantages

L'analyse graphique de la distribution des estimations bootstrap (histogramme, Q-Q plot) est essentielle.

1. Intérêt des diagnostics graphiques :

- **Vérifier la normalité** : Si la distribution est proche d'une courbe en cloche (Normale), les intervalles de confiance classiques sont fiables.
- **Sanity Check** : Permet de repérer des anomalies (bimodalité, valeurs aberrantes) invisibles avec un simple calcul.

2. Avantages généraux de la méthode :

- **Pédagogique** : Permet de visualiser concrètement l'incertitude et les fluctuations d'échantillonnage.
- **Priorité à la clinique** : On choisit la statistique la plus pertinente cliniquement (ex: médiane, ratio) sans être limité par la complexité mathématique de son intervalle de confiance.
- **Intelligibilité** : Offre souvent une alternative plus compréhensible aux modèles complexes (comme les modèles mixtes) pour gérer des données difficiles.
- **Interprétation facilitée de l'Intervalle de Confiance** : L'intervalle de confiance classique est souvent mal compris car sa définition est théorique et abstraite (*"si on répétait l'étude une infinité de fois..."*). Avec le bootstrap, cette répétition devient concrète et visible grâce à l'histogramme des simulations. On comprend intuitivement que l'IC correspond simplement à la zone où se concentrent la majorité (95%) des résultats simulés.

Exemple de diagnostics graphiques :

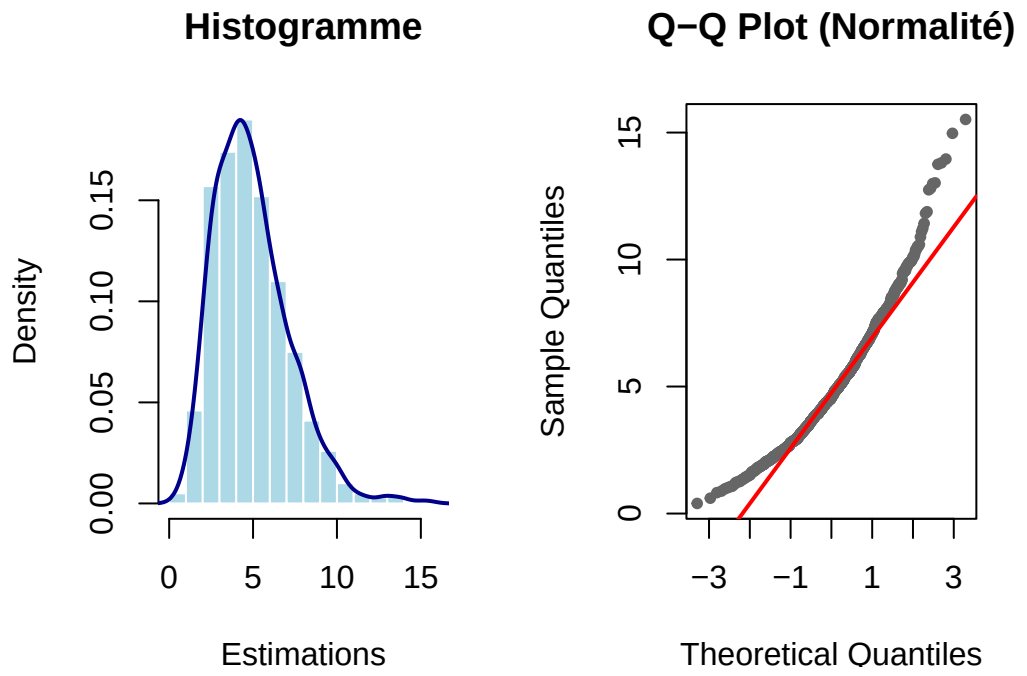

Figure 9: Diagnostics : L'histogramme et le Q-Q plot permettent de juger de la normalité de la distribution bootstrap.

7.D Exemple R

Modèle linéaire expliquant la variable « durée de l'entretien » réalisée lors de l'étude santé mentale en prison.

En vérifiant les conditions de validité du modèle, il y a un doute sur l'absence de normalité des résidus.

```
mod <- lm(
    duree.interv~

 ↪  schizophrenie+depression+abus.subst+gravite+charactere+trauma.enfant+age+factor(type.centre
    data=smp
    )
summary(mod)
```

```
Call:
lm(formula = duree.interv ~ schizophrenie + depression + abus.subst +
    gravite + charactere + trauma.enfant + age + factor(type.centre),
    data = smp)

Residuals:
    Min      1Q  Median      3Q     Max
-38.578 -13.855  -1.769  10.922  64.405

Coefficients:
```

```
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)          42.48996    3.99998  10.623  < 2e-16 ***
schizophrenie         3.06420    2.80997   1.090 0.275911
depression            6.71269    1.64793   4.073 5.21e-05 ***
abus.subst            4.60037    1.79854   2.558 0.010760 *
gravite               1.06236    0.56548   1.879 0.060737 .
charactere            1.62547    0.93536   1.738 0.082723 .
trauma.enfant        -0.66805    1.66931  -0.400 0.689144
age                   0.20788    0.06066   3.427 0.000649 ***
factor(type.centre)2  4.09540    2.53401   1.616 0.106546
factor(type.centre)3 -1.29681    2.44159  -0.531 0.595509
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 18.02 on 645 degrees of freedom
  (144 observations deleted due to missingness)
Multiple R-squared:  0.1086,    Adjusted R-squared:  0.09619
F-statistic: 8.734 on 9 and 645 DF,  p-value: 1.919e-12
```

```r
par(mfrow=c(1,2))
hist(residuals(mod), main="Histogramme des résidus", xlab="Résidus")
qqnorm(residuals(mod), main="QQ-plot des résidus")
qqline(residuals(mod), col="red")
```

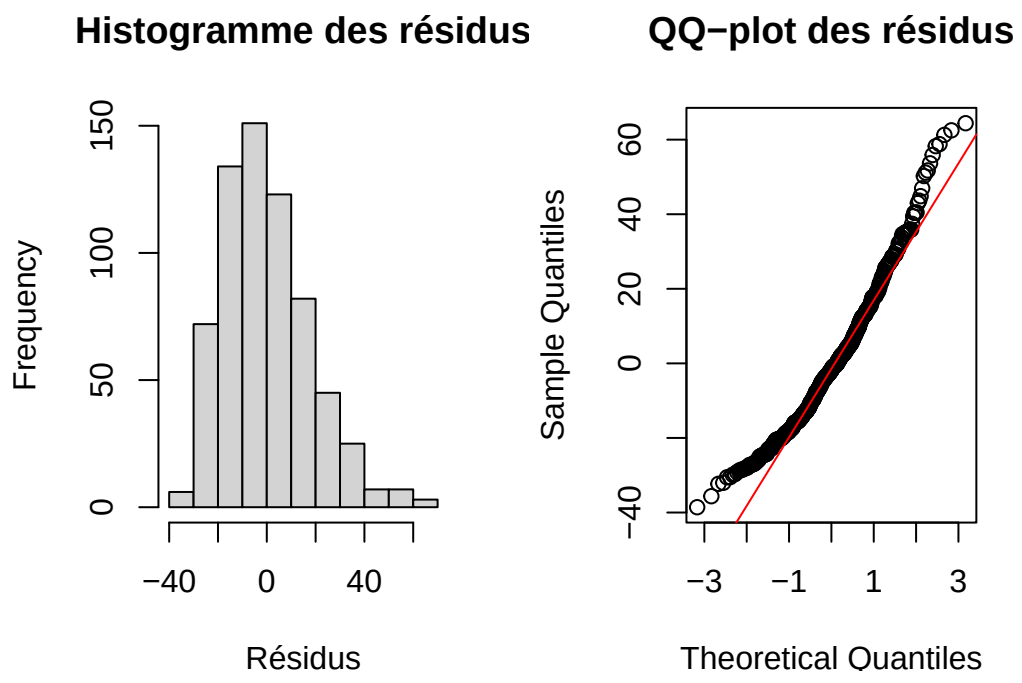

Figure 10: Histogramme et QQ-plot des résidus du modèle linéaire expliquant la durée de l'entretien. La distribution des résidus semble s'écarter de la normalité.

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

On fait une estimation par *bootstrap* en négligeant la dépendance intra-centre.

```r
lm.boot <- function(data, index) {
        smp.boot <- data[index,]
        mod <- lm(
            duree.interv ~
            schizophrenie + depression + abus.subst + gravite + charactere +
 ↪  trauma.enfant +age + factor(type.centre),
            data = smp.boot)

    coefficients(mod)
}
```

Concernant la fonction R :

Il faut définir une fonction `lm.boot` :

- entrée : jeu de données `data`
- vecteur d'indices : ìndex'
- Estime les paramètres du modèle linéaire pour chaque échantillon bootstrap.
- Les coeficients sont stockés dans `coefficients(mod)`.

Puis utilisation de la fonction `boot()` du package `boot` pour réaliser le bootstrap.

- `smp` : jeu de données original
- `lm.boot` : fonction définie précédemment à "bootstrapper", c'est à dire à appliquer sur chaque échantillon bootstrap

Si l'on est plus spécialement intéressé par la variable « schizophrénie » il faut choisir le deuxième paramètre du tableau de résultats `[,2]` car la première colonne correspond à l'ordonnée à l'origine = intercept.

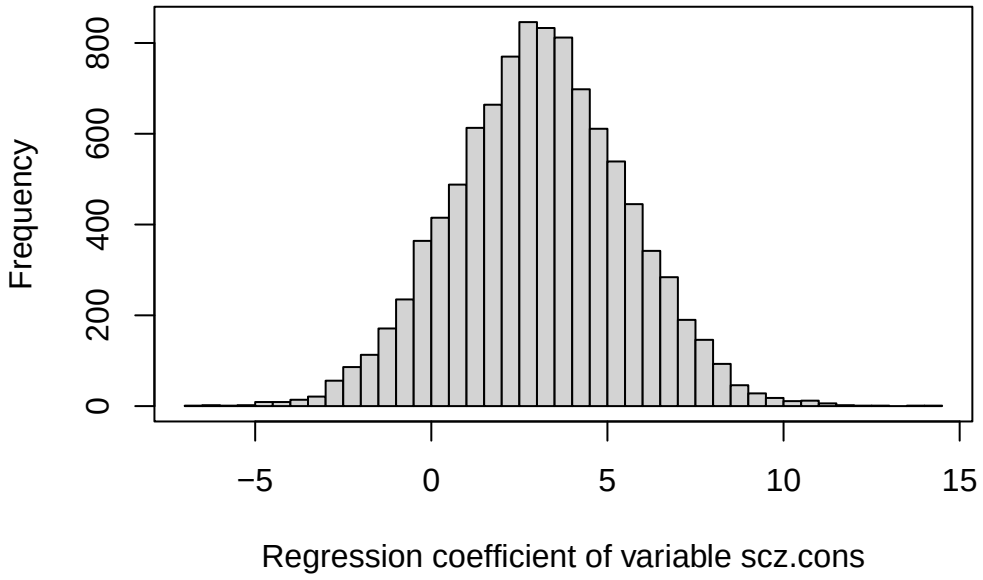

Figure 11: Histogramme des 10000 coefficients de régression de la variable « schizophrénie » estimés via bootstrap, c'est-à-dire issus de 10000 jeux de données obtenus par tirage au sort avec remise à partir du jeu de données original.

Pour obtenir l'IC à 95% de la variable « schizophrénie » et le p-value associée :

```
# index = 2 car on s'intéresse au 2ème coefficient (schizophrénie)
# type = "bca" pour méthode des percentiles corrigée (plus robuste sur petits
 ↪  échantillons)
boot.ci(resboot, index=2, type= "bca")
```

```
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 10000 bootstrap replicates

CALL :
boot.ci(boot.out = resboot, type = "bca", index = 2)

Intervals :
Level       BCa
95%   (-1.737,  7.871 )
Calculations and Intervals on Original Scale
```

```
# p-value bilatérale
2*sum(resboot$t[,2]<=0)/10000
```

```
[1] 0.2168
```