Le modèle linéaire

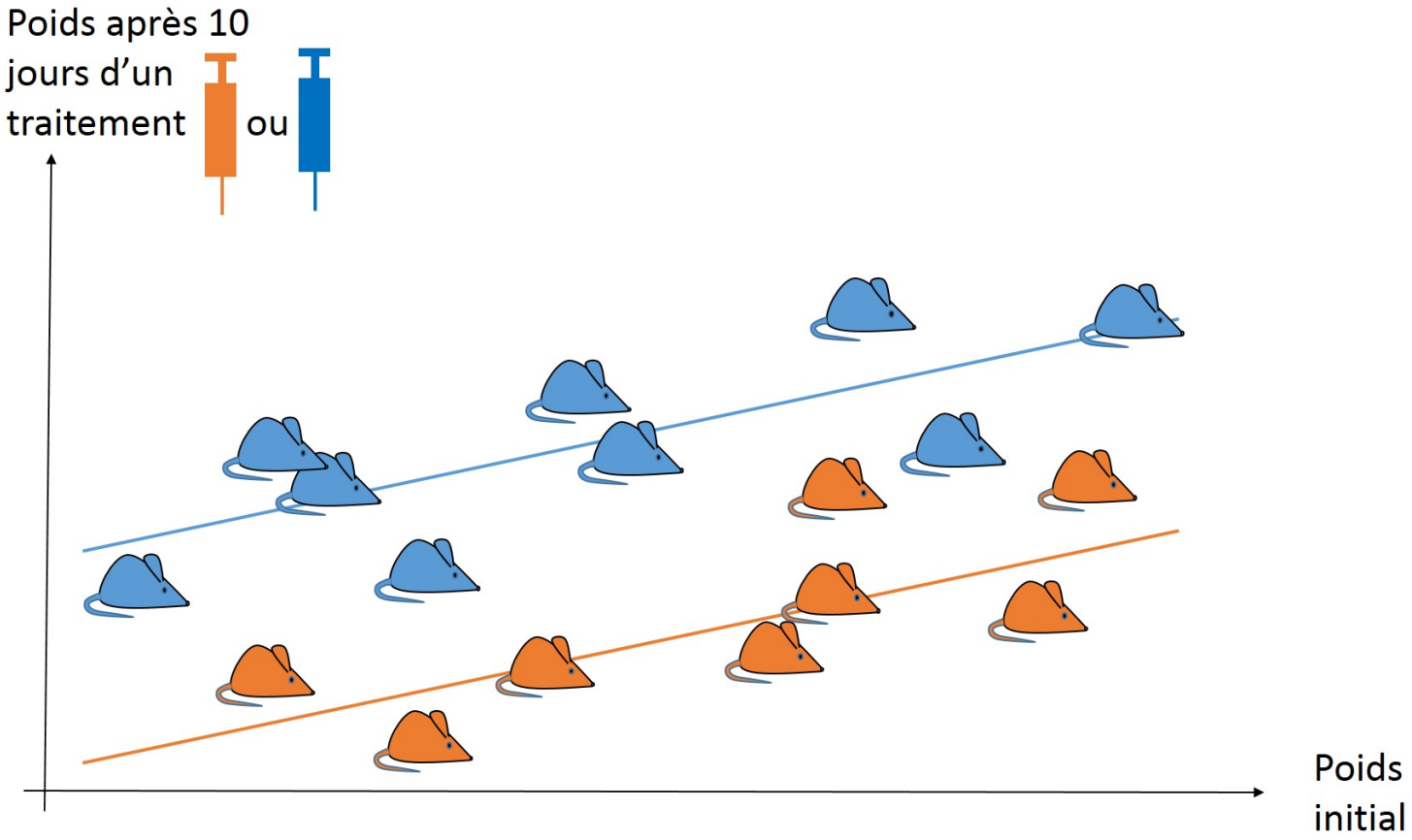
Modélisation d'une variable quantitative en fonction de variables quantitatives ou qualitatives

M. L. Delignette-Muller
VetAgro Sup

5 janvier 2023

Illustration

Ex. : modélisation d'une variable quantitative (le poids 10 jours ap. traitement) en fonction de deux variables explicatives, une quantitative (le poids initial) et une qualitative (le traitement).



Objectifs pédagogiques

- Connaître les bases théoriques du modèle linéaire.
- Connaître le cadre d'utilisation classique du modèle linéaire.
- Savoir vérifier ses conditions d'utilisation.
- Savoir interpréter les coefficients estimés et utiliser un modèle en inférence.
- Avoir un aperçu des extensions du modèle linéaire.

Plan

Exemple inspiré de la littérature

Han *et al.* 2001,
Response Surface Modeling for the Inactivation of *Escherichia coli* O157 :H7 on Green Peppers (*Capsicum annuum* L.) by Chlorine Dioxide Gas Treatments.

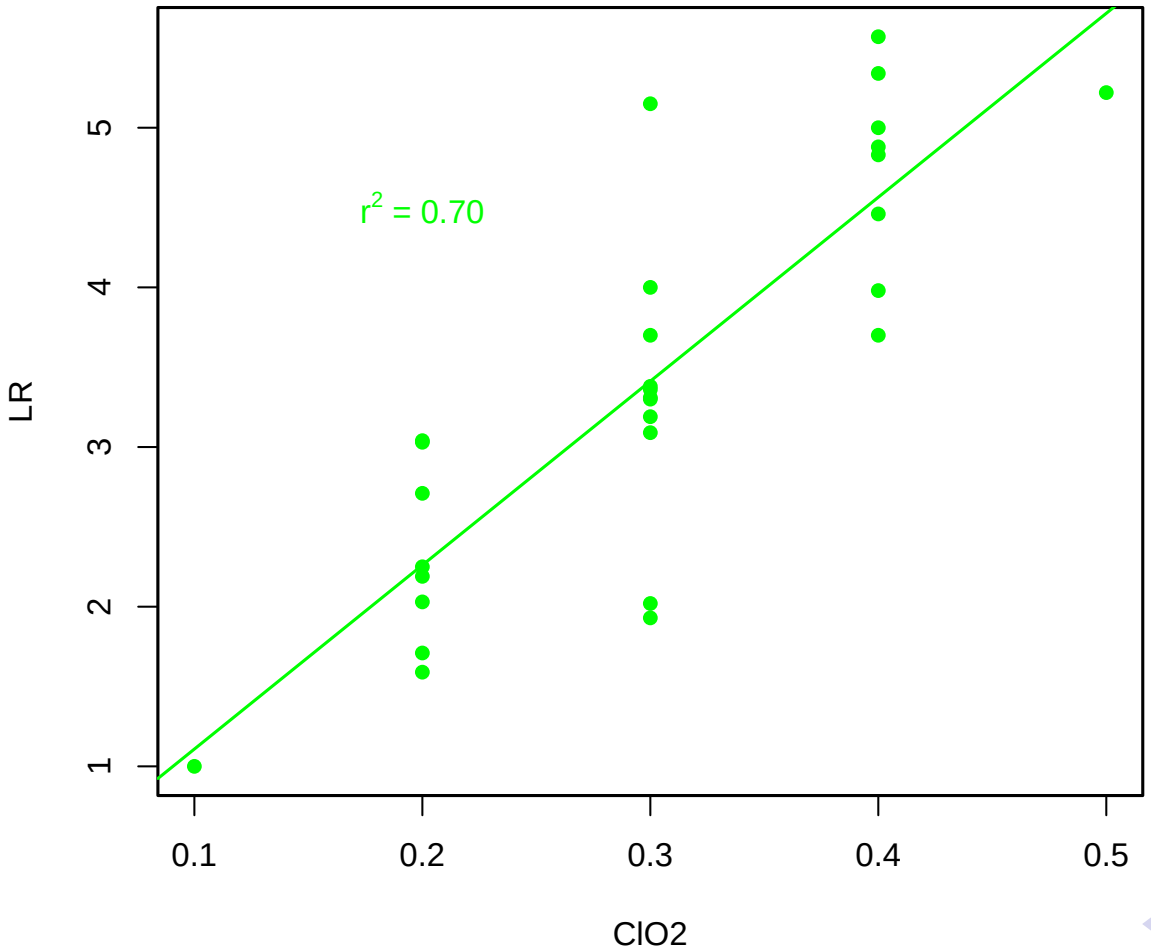
- **4 variables contrôlées** :
  - concentration en dioxyde de chlore $Cl0_2$ $(mg.l^{-1})$,
  - température T $(^\circ C)$,
  - temps de traitement t (min) et
  - humidité relative HR (%).
- **Variable observée notée** LR : la réduction bactérienne en UFC (en log) pour 5 g de poivre.

Visualisation du jeu de données

```
> d <- read.table("DATA/han2001.txt", header = TRUE,
                  stringsAsFactors = TRUE)
> str(d)
'data.frame':        29 obs. of  5 variables:
 $ T   : int  10 20 10 20 10 20 10 20 10 20 ...
 $ HR  : int  65 65 85 85 65 65 85 85 65 65 ...
 $ ClO2: num  0.2 0.2 0.2 0.2 0.4 0.4 0.4 0.4 0.2 0.2 ...
 $ t   : num  15 15 15 15 15 15 15 15 65 65 ...
 $ LR  : num  1.59 1.71 2.19 2.25 3.7 3.98 4.88 5 2.03 2.71
> head(d)
   T HR ClO2  t   LR
1 10 65  0.2 15 1.59
2 20 65  0.2 15 1.71
3 10 85  0.2 15 2.19
4 20 85  0.2 15 2.25
```

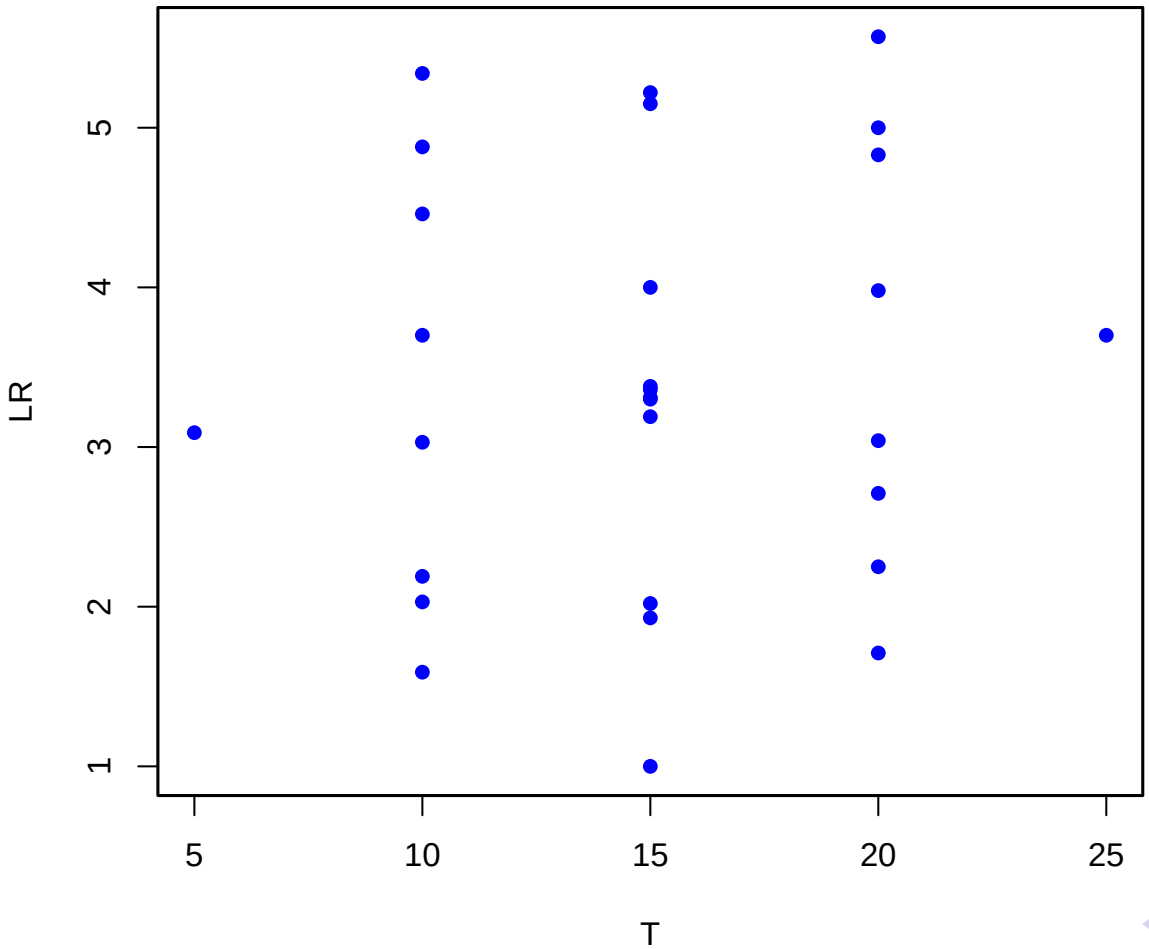
Impact du dioxyde de chlore ?

Diagramme de dispersion et régression simple en ignorant les autres variables de contrôle



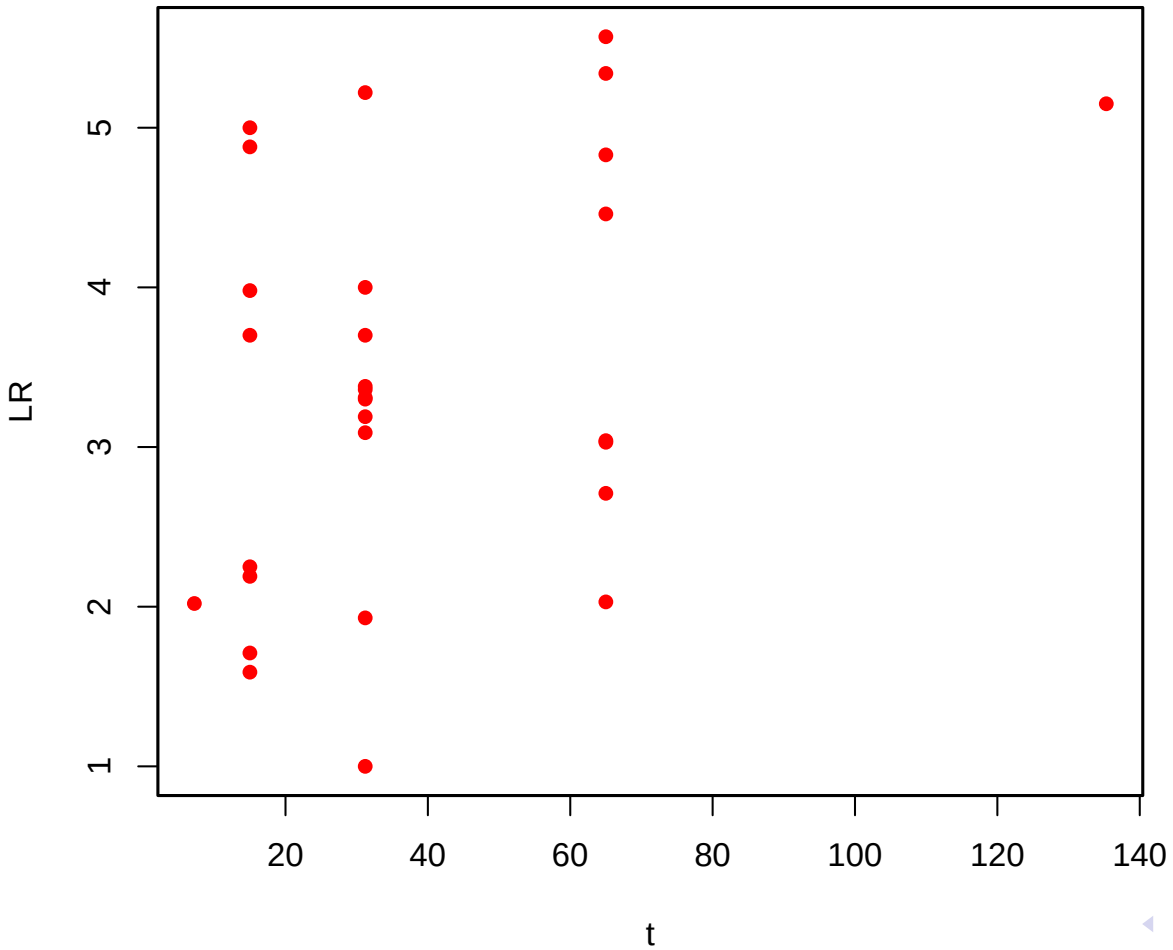
Impact de la température ?

Diagramme de dispersion en ignorant les autres variables de contrôle



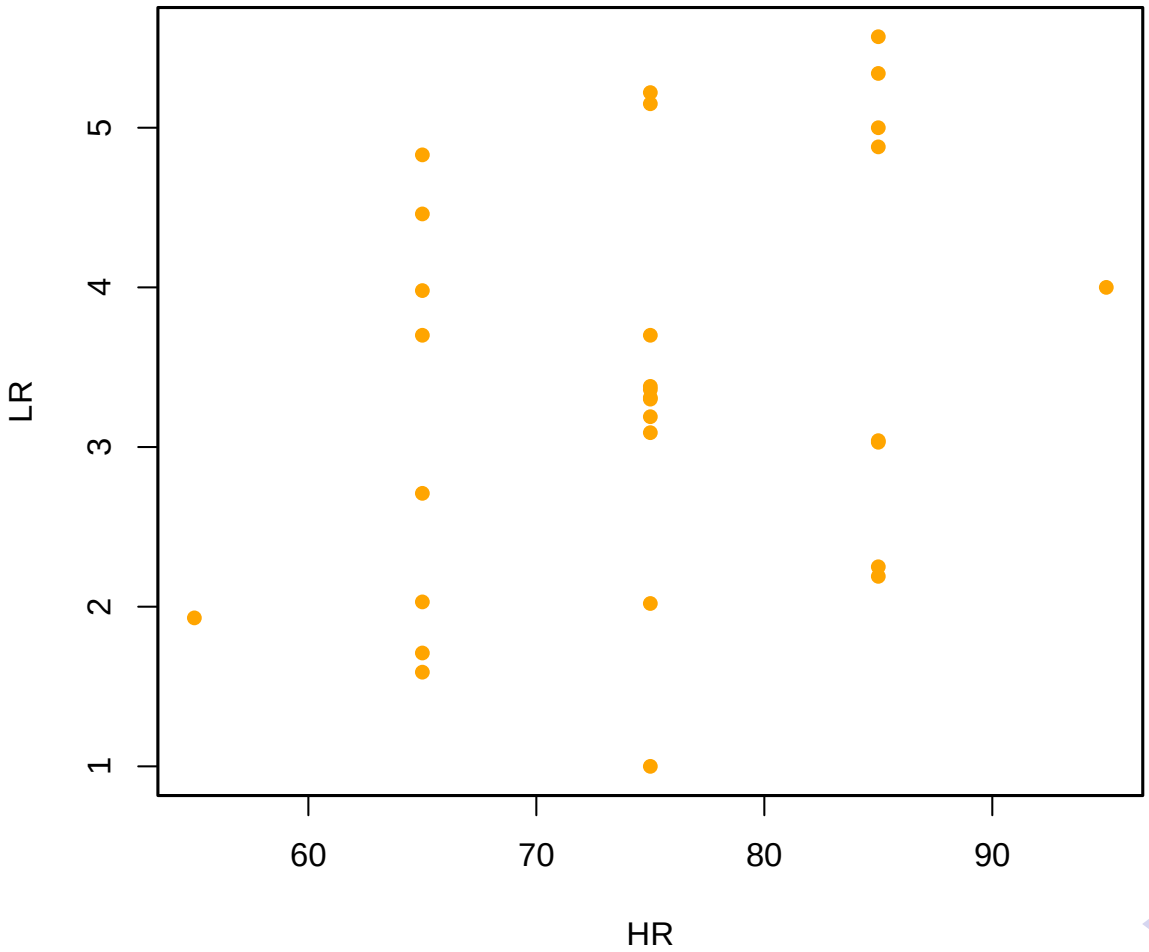
Impact du temps de traitement ?

Diagramme de dispersion en ignorant les autres variables de contrôle



Impact de l'humidité relative ?

Diagramme de dispersion en ignorant les autres variables de contrôle



Prise en compte de plusieurs variables de contrôle

L'effet le plus évident est celui du dioxyde de chlore, mais

- les autres variables n'ont-elles pas d'effet ?
- Serait-il intéressant de les intégrer dans un modèle pour mieux prédire l'effet d'un traitement ?
- Ne pourrait-on pas atteindre un pourcentage de variance expliquée (r^2) supérieur à 70% ?

$\Rightarrow$ intérêt de la **régression multiple**.

Le modèle de régression multiple

Simple extension du modèle de régression linéaire simple permettant la prise en compte de plusieurs variables de contrôle (appelées aussi régresseurs ou covariables).

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_p X_{pi} + \epsilon_i$$
avec $\epsilon_i \sim N(0, \sigma)$

Partie déterministe : relation linéaire

Partie stochastique : modèle gaussien

ϵ_i aléatoires, indépendants, suivant une loi normale (loi de Gauss) de variance résiduelle σ^2 constante.

Méthode d'estimation des paramètres

Comme en régression linéaire simple.
Maximisation de la vraisemblance (maximisation de $Pr(Y|\beta_0, \beta_1, \ldots, \beta_p, \sigma)$) qui revient dans le cadre du modèle gaussien à la **minimisation de la Somme des Carrés des Ecarts (SCE)**

$$SCE = \sum_{i=1}^{n} e_i^2 \text{ avec } e_i = Y_i - \hat{Y}_i$$

Problème d'optimisation auquel correspond une solution analytique : $b = (X'X)^{-1}X'y$

$$\text{avec } b = \begin{pmatrix} b_1 \\ b_2 \\ \ldots \\ b_p \end{pmatrix}, X = \begin{pmatrix} 1 & x_{11} & \ldots & x_{p1} \\ 1 & x_{12} & \ldots & x_{p2} \\ \ldots & \ldots & \ldots & \ldots \\ 1 & x_{1n} & \ldots & x_{pn} \end{pmatrix} \text{ et } y = \begin{pmatrix} y_1 \\ y_2 \\ \ldots \\ y_n \end{pmatrix}$$

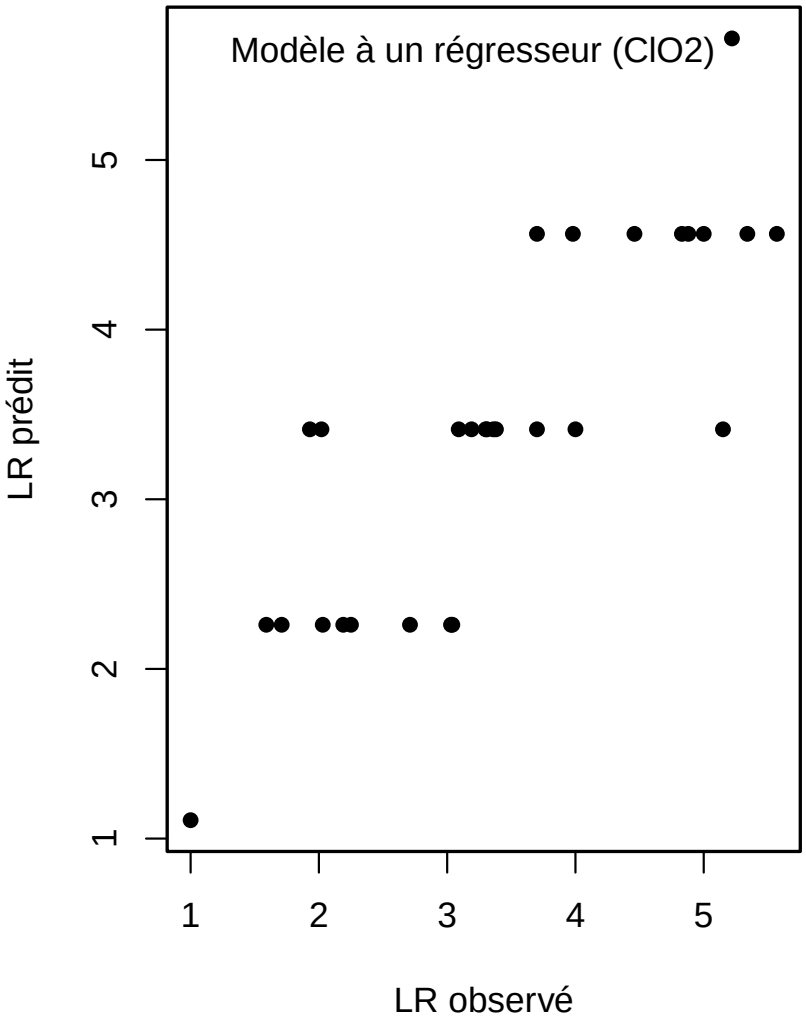
Régression linéaire multiple avec **R**

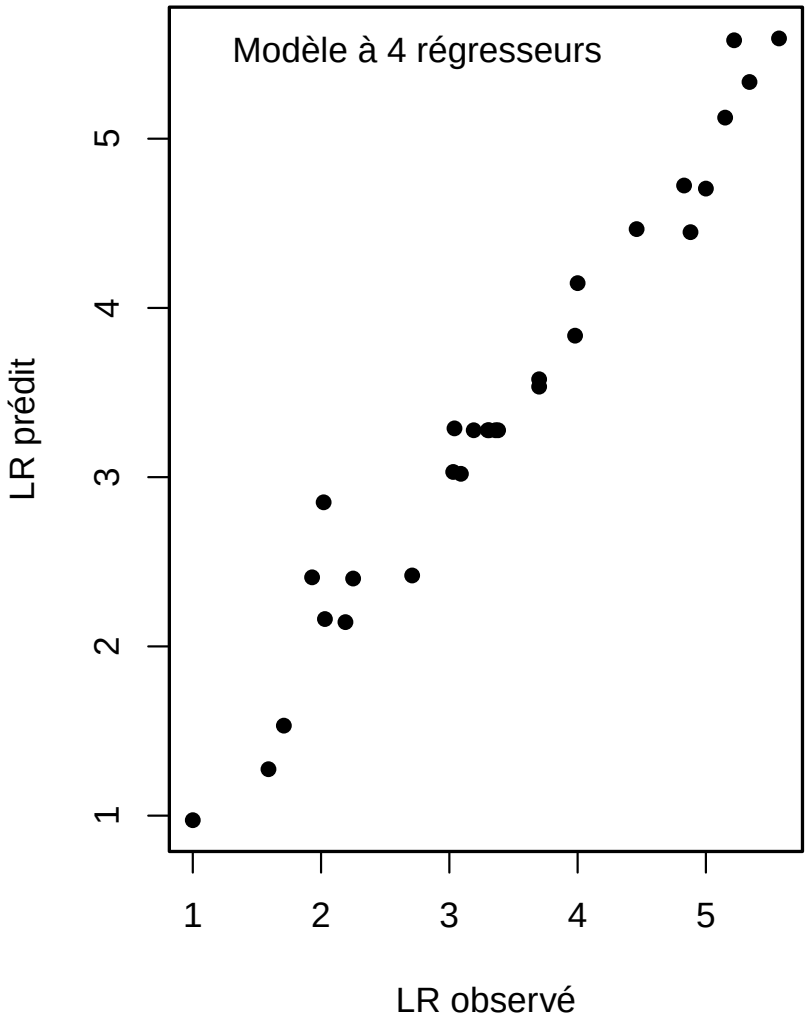
```
> m <- lm(LR ~ T + HR + ClO2 + t, data = d)
> m

Call:
lm(formula = LR ~ T + HR + ClO2 + t, data = d)

Coefficients:
(Intercept)            T           HR         ClO2            t
    -4.3783       0.0258       0.0435      11.5208       0.0177
```

Régression linéaire multiple avec **R**

```
> summary(m)

Call:
lm(formula = LR ~ T + HR + ClO2 + t, data = d)

Residuals:
    Min      1Q  Median      3Q     Max
-0.8313 -0.0872  0.0269  0.1215  0.4323

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -4.37830    0.48624   -9.00  3.7e-09
T            0.02575    0.01110    2.32    0.029
HR           0.04346    0.00555    7.83  4.6e-08
ClO2        11.52083    0.55483   20.76  < 2e-16
t            0.01775    0.00188    9.45  1.5e-09

Residual standard error: 0.272 on 24 degrees of freedom
Multiple R-squared:  0.961,         Adjusted R-squared:  0.954
F-statistic:  147 on 4 and 24 DF,  p-value: <2e-16
```

Comparaison prédits / observés pour les 2 modèles





Intérêt de la prise en compte de plusieurs régresseurs

Dans cet exemple

- chaque régresseur apporte une contribution significative à l'explication de la variable observée.
- On passe de 70% de variance expliquée (r^2) avec seulement ClO_2 à 96% avec les 4 régresseurs.

Mais le modèle théorique est-il valable, tant pour sa partie déterministe que pour sa partie stochastique ?
$\Rightarrow$ Vérification *a posteriori* des conditions d'utilisation.

Examen des résidus

Même principe qu'en régression linéaire simple : vérification *a posteriori* du modèle d'erreur gaussien (partie stochastique du modèle).

- **graphe des résidus** = examen des résidus en fonction de la variable prédite
- **diagramme Quantile-Quantile** des résidus (vérification de la normalité de leur distribution)
- + examen des **résidus en fonction de chaque régresseur** (notamment pour contrôler la linéarité du modèle)

Graphe des résidus

```
> plot(residuals(m) ~ fitted(m), pch = 16)
```

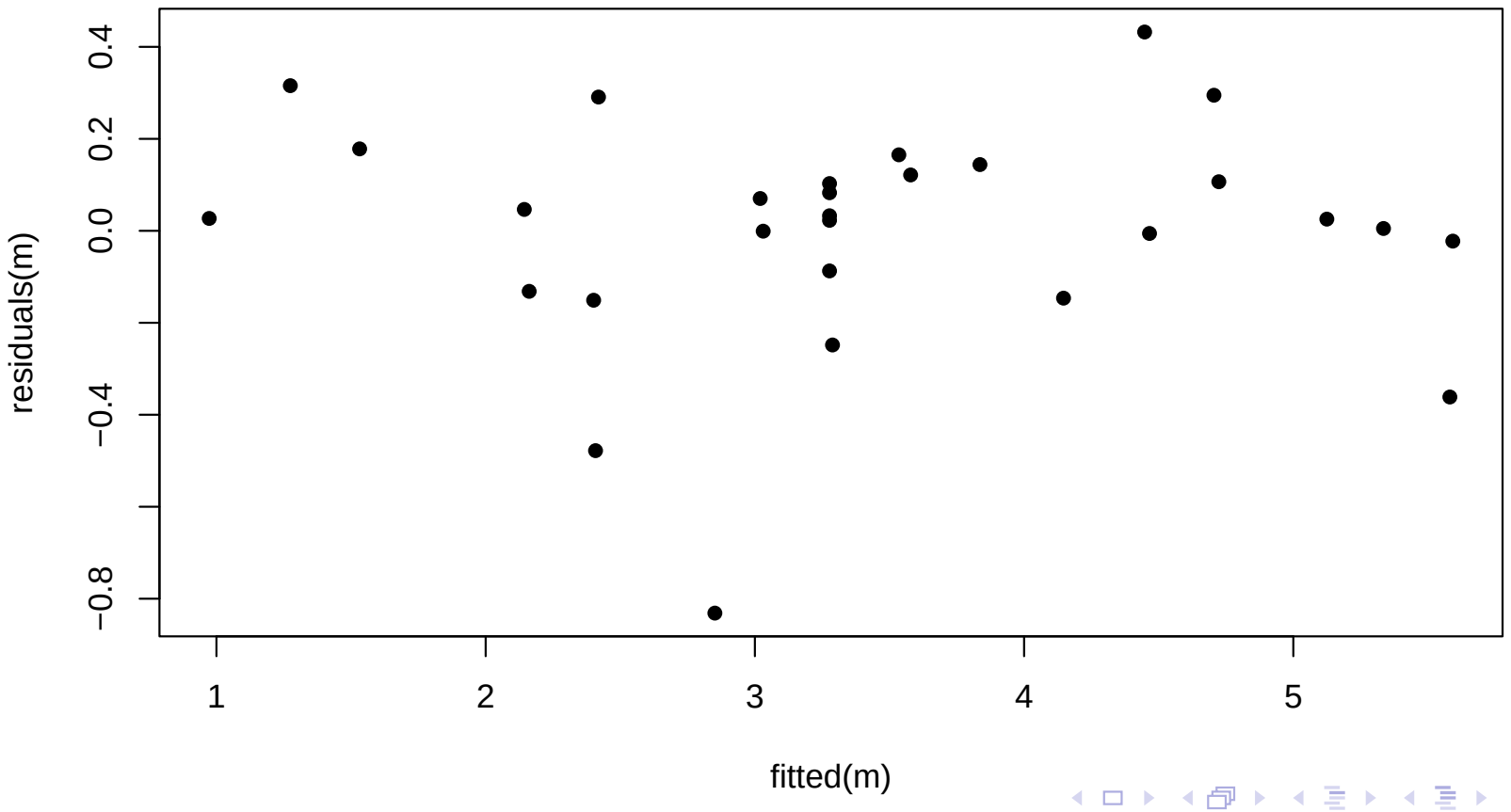
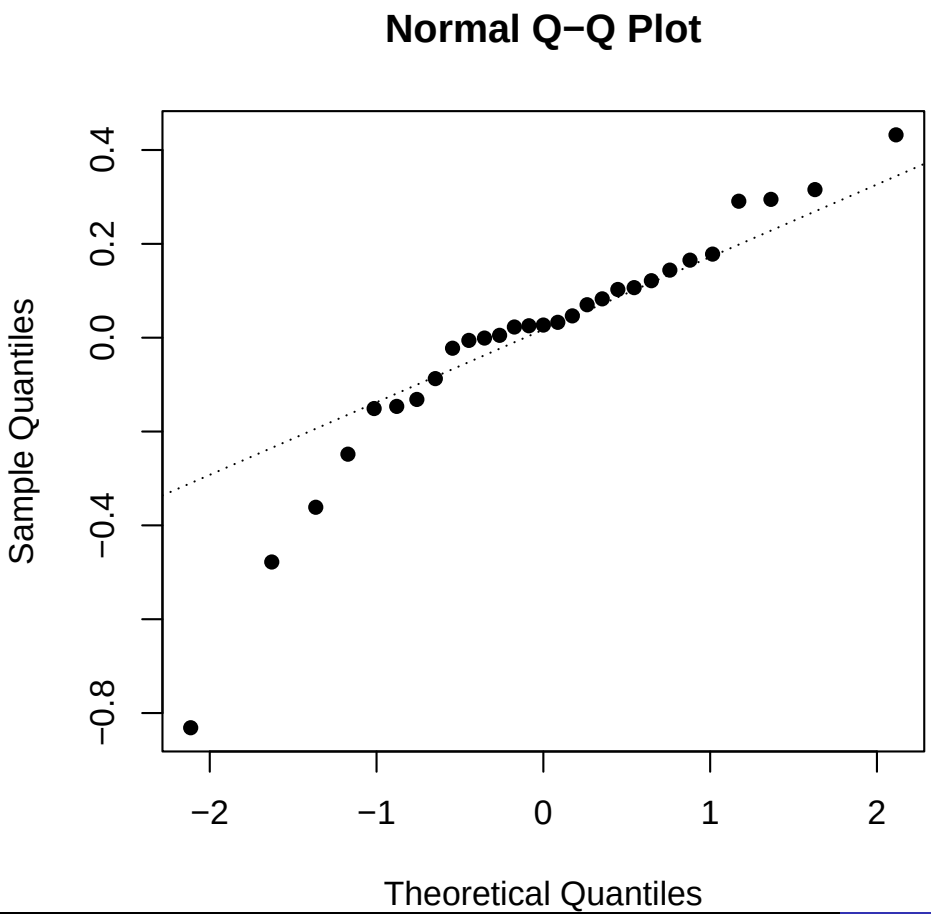


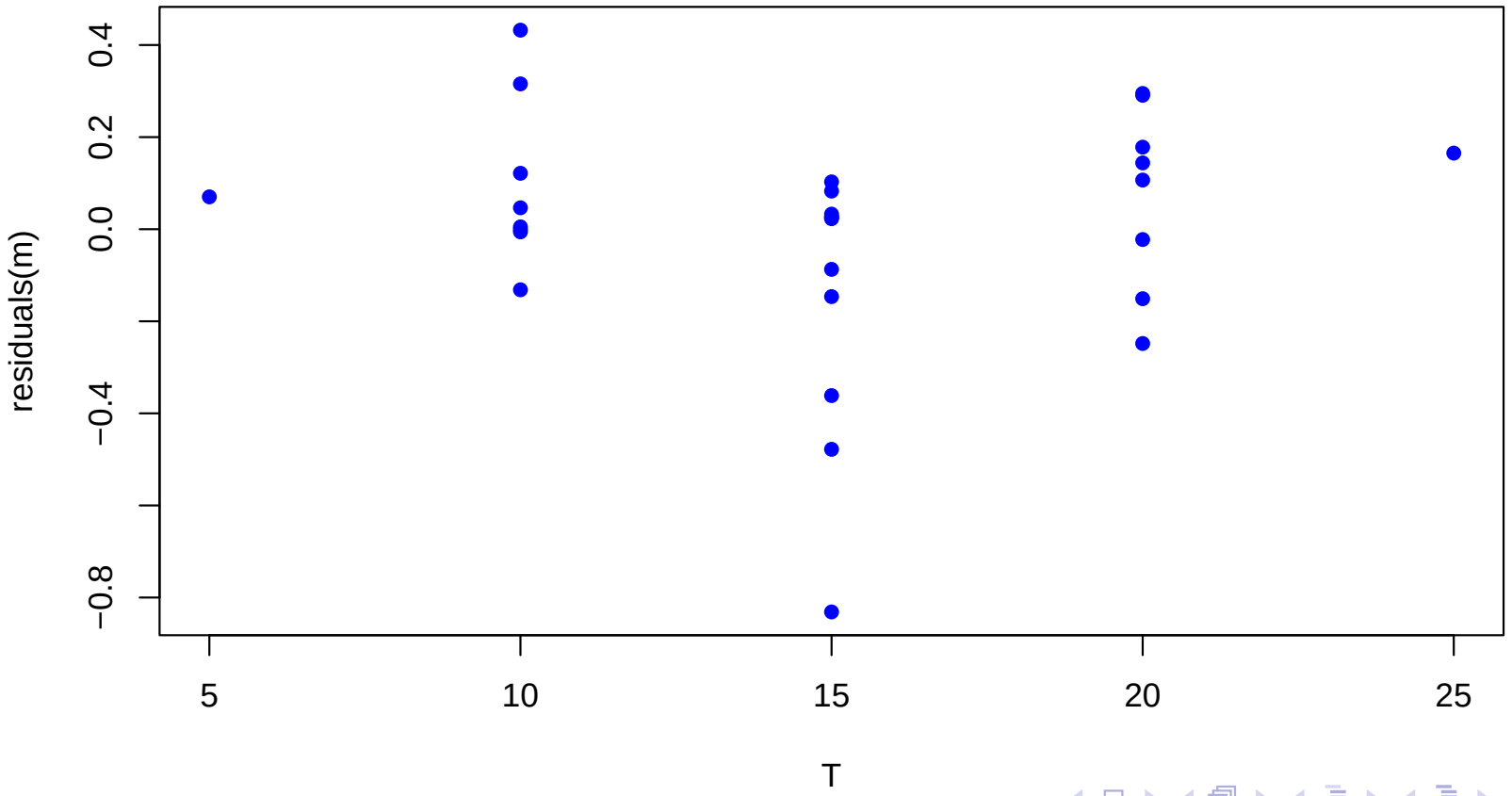
Diagramme Quantile - Quantile des résidus

```
> qqnorm(residuals(m), pch = 16)
> qqline(residuals(m), lty = 3)
```



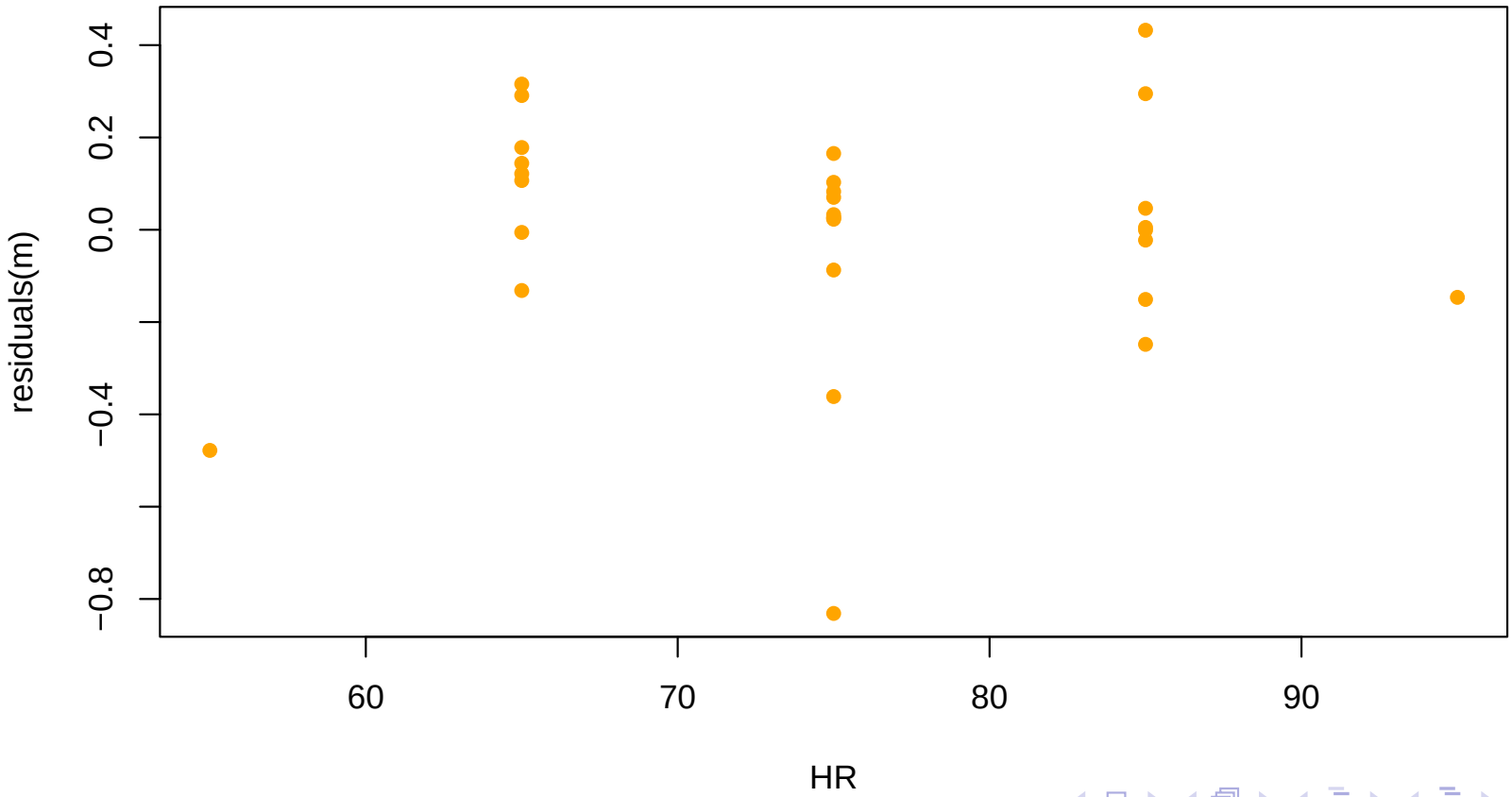
Résidus en fonction de chaque régresseur (1)

```
> plot(residuals(m) ~ T, data = d, pch = 16, col = "blue")
```



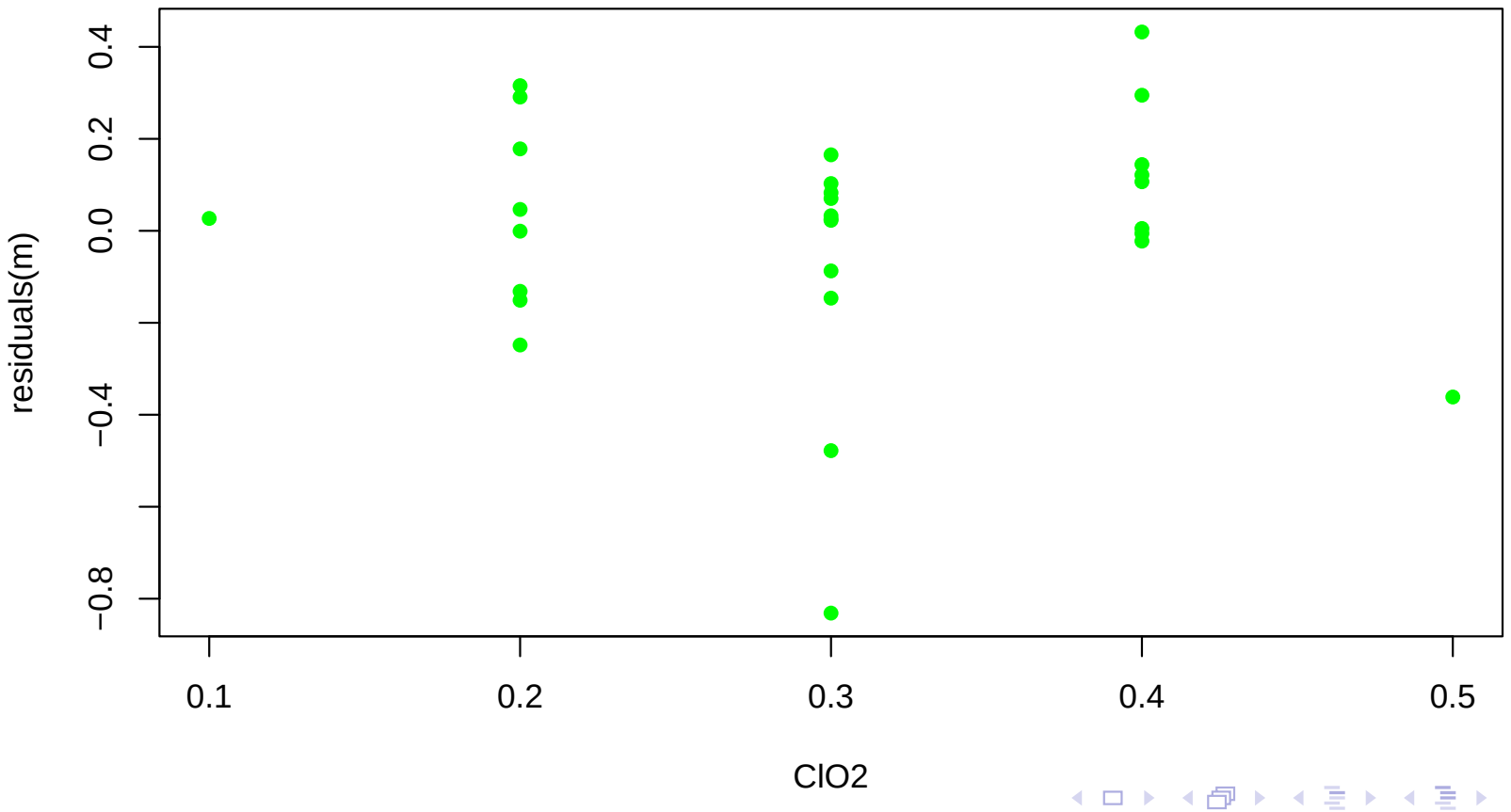
Résidus en fonction de chaque régresseur (2)

```
> plot(residuals(m) ~ HR, data = d, pch = 16, col = "orange
```



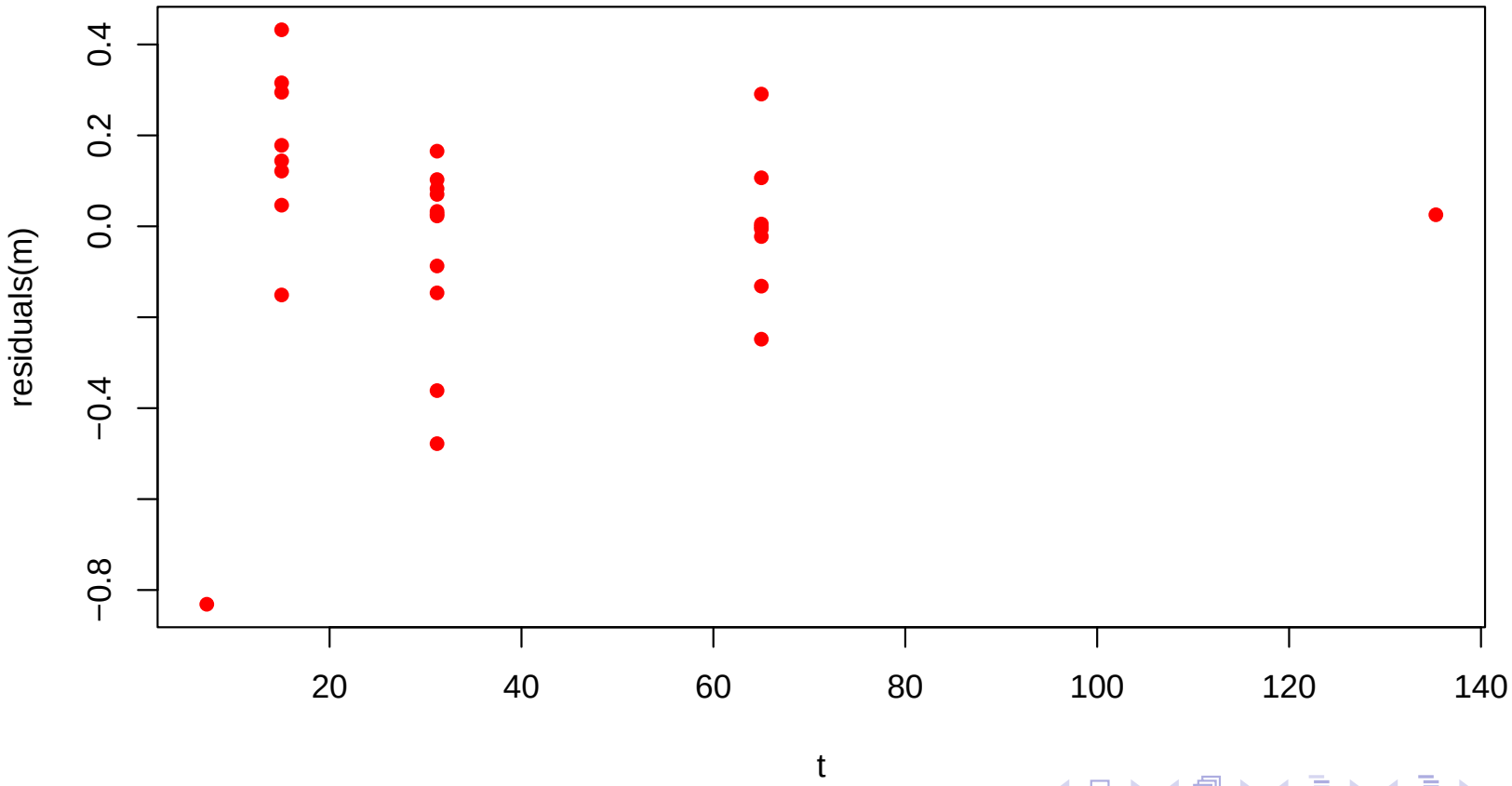
Résidus en fonction de chaque régresseur (3)

```
> plot(residuals(m) ~ ClO2, data = d, pch = 16, col ="green
```



Résidus en fonction de chaque régresseur (4)

```
> plot(residuals(m) ~ t, data = d, pch = 16, col = "red")
```



Détection des données influentes - "Jacknife" ou eustachage

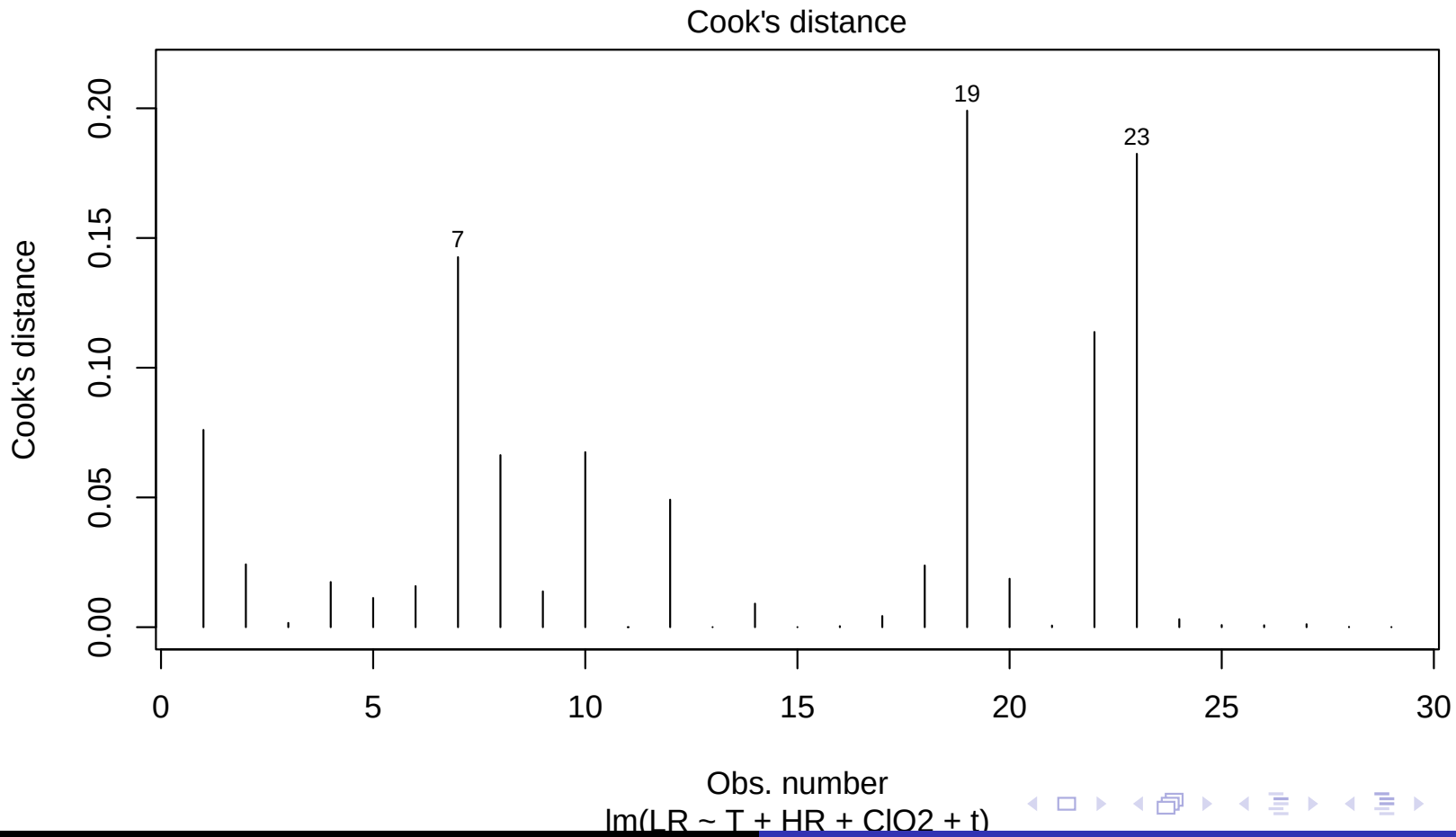
Pour chaque observation $n^\circ i$ (i allant de 1 à n)

- réestimation des paramètres du modèle par régression sur le **jeu de données sans l'observation** $n^\circ i$
- quantification globale de l'impact de l'observation $n^\circ i$ sur l'ensemble des paramètres du modèle - **distance de Cook**

Graphe représentant les distances de Cook en fonction du numéro de l'observation, permettant de détecter d'éventuelles observations nettement plus influentes que les autres.

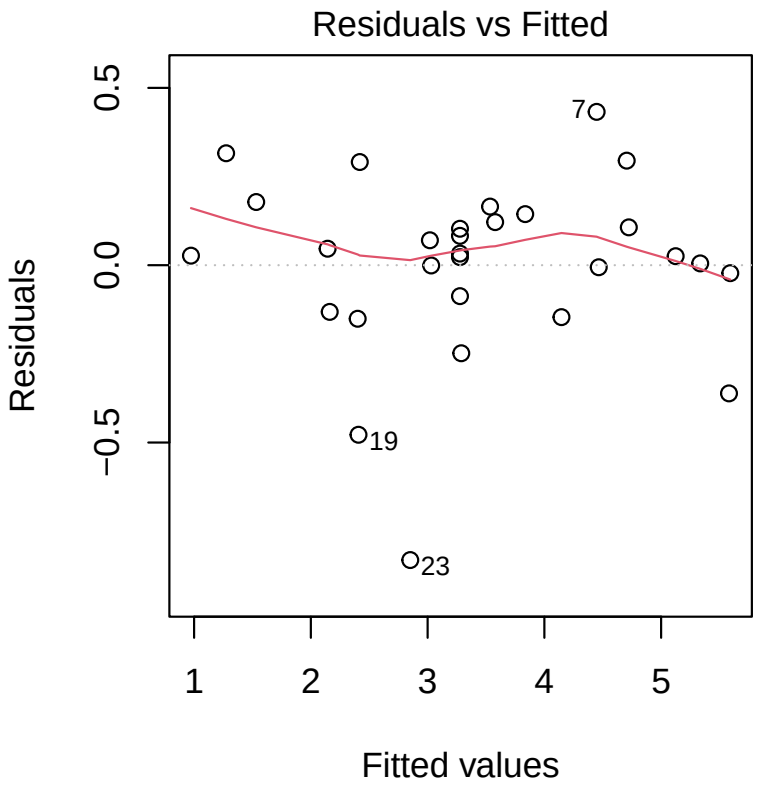
Graphe des distances de Cook

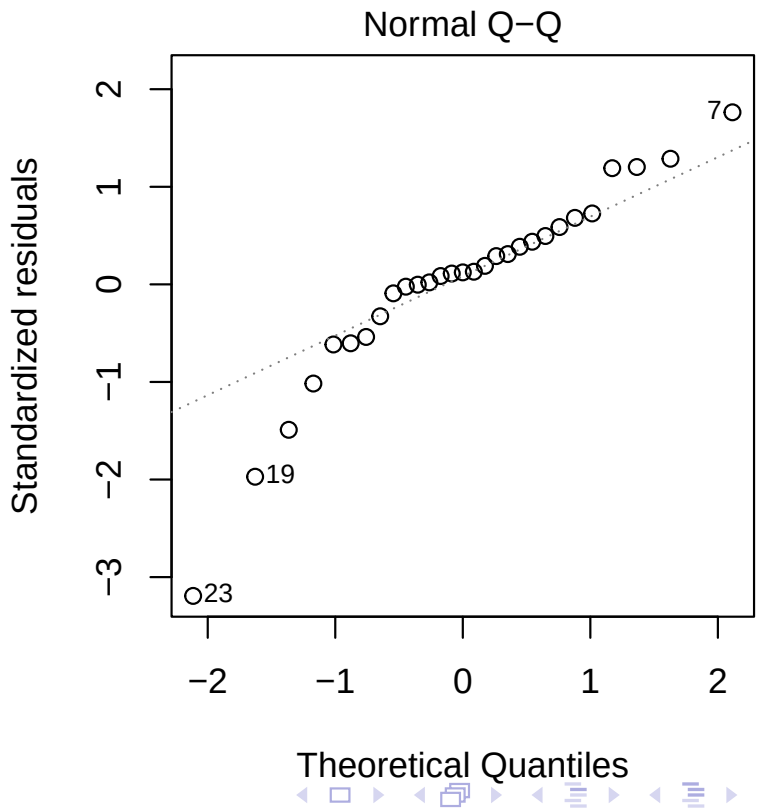
```
> plot(m, which = 4)
```



Repérage possible aussi des observations extrêmes sur le graphe des résidus et le diagramme Quantile- Quantile

```
> plot(m, which = 1); plot(m, which = 2)
```





Que faire en cas de non respect du modèle d'erreur ?

- **Non normalité des résidus et/ou hétéroscedasticité** (variance non constante) : tenter une transformation de variable ou changer la partie stochastique du modèle (modèle généralisé)
- **Non indépendance des résidus** ou non linéarité : tenter une transformation de variable ou utiliser un modèle déterministe plus complexe ou prendre en compte une autocorrélation des résidus (notamment sur séries temporelles).
- **Observations influentes** : examiner de plus près ces observations, et tester la robustesse des conclusions issues de la régression à l'incorporation ou non des ces observations.

Prédiction à partir du modèle

Même principe qu'en régression linéaire.

Intervalle de confiance sur la moyenne (marge d'erreur sur la moyenne = incertitude sur la partie déterministe)

```
> predict(m, data.frame(T = 10, HR = 80, ClO2 = 0.2, t = 60),
          interval="confidence")
```

```
   fit  lwr  upr
1 2.72 2.51 2.94
```

Intervalle de prédiction (marge d'erreur sur une observation prédite, intégrant aussi la partie stochastique)

```
> predict(m, data.frame(T = 10, HR = 80, ClO2 = 0.2, t = 60),
          interval="prediction")
```

```
   fit  lwr  upr
1 2.72 2.12 3.33
```

approché souvent par $\hat{Y}_0 \pm 2 \times \sigma$

Coefficients du modèle et interprétation

```
> coef(m)
(Intercept)           T          HR        ClO2           t
    -4.3783      0.0258      0.0435     11.5208      0.0177
```

$$LR = -4.38 + 0.0258\,T + 0.0435\,HR + 11.5\,ClO_2 + 0.0177\,t$$

Exemple d'interprétation :
Toutes conditions égales par ailleurs, quand on augmente la concentration en ClO_2 de 0.1, on augmente le log de la réduction décimale (LR) de 1.15.

Coefficients du modèle dans le résumé et tests de nullité de chacun d'eux associés

```
> summary(m)$coefficients
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -4.3783    0.48624   -9.00 3.66e-09
T             0.0258    0.01110    2.32 2.91e-02
HR            0.0435    0.00555    7.83 4.58e-08
ClO2         11.5208    0.55483   20.76 7.63e-17
t             0.0177    0.00188    9.45 1.45e-09
```

Chaque valeur de p correspond au test de l'égalité à 0 du coefficient associé, les autres étant pris en compte dans le modèle.

Intervalles de confiance sur les coefficients (ou paramètres) du modèle

```
> confint(m)
                 2.5 %  97.5 %
(Intercept) -5.38184 -3.3747
T            0.00285  0.0487
HR           0.03201  0.0549
ClO2        10.37571 12.6660
t            0.01387  0.0216
```

A noter ici que tous les coefficients ont un intervalle de confiance ne comprenant pas 0 (corrobore les résultats affichés dans le résumé du modèle).

Cas de régresseurs non contrôlés ou données non issues d'un plan d'expérience

Ciliberti *et al.* 2011,
The Nile monitor (*Varanus niloticus* ; Squamata : Varanidae) as a sentinel species for lead and cadmium contamination in sub-Saharan wetlands.

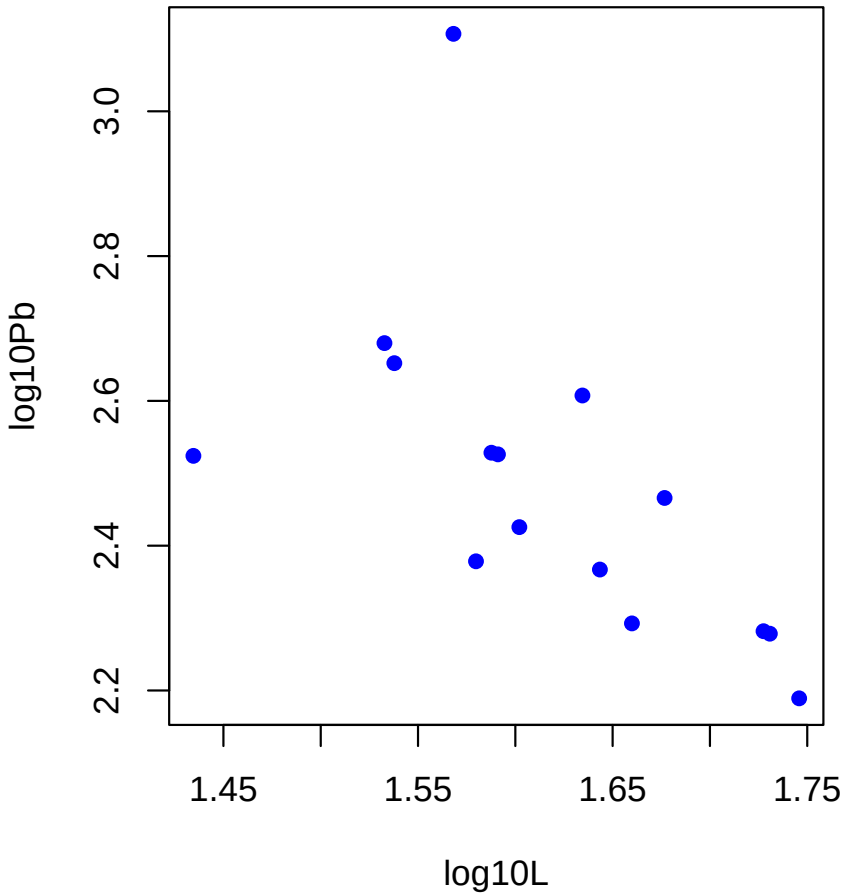
Modélisation de la teneur en plomb (en log_{10} : $log10Pb$) dans le rein de varans en fonction

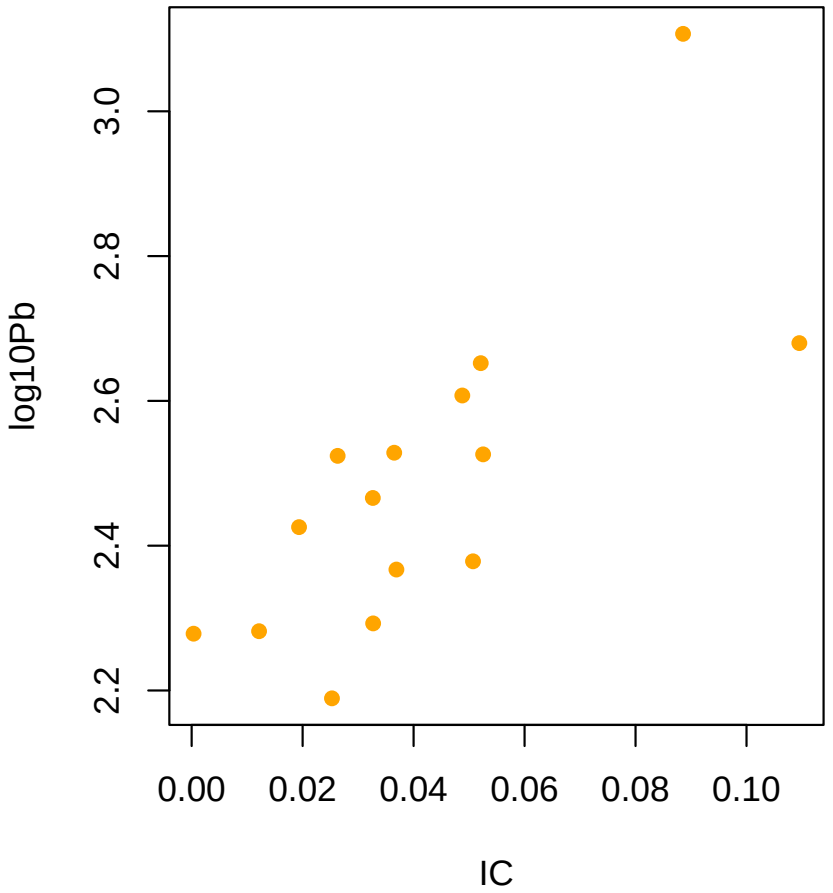
- de la **longueur museau - cloaque** (L)
- et de l'**indice corporel** (IC - part de masse graisseuse).

Visualisation du jeu de données

```
> dvar <- read.table("DATA/Pbvarans.txt", header = TRUE,
                     stringsAsFactors = TRUE)
> str(dvar)
'data.frame':        15 obs. of  4 variables:
 $ log10M : num  -0.466 0.33 0.292 0.215 0.225 ...
 $ log10L : num  1.43 1.66 1.68 1.63 1.64 ...
 $ IC     : num  0.0263 0.0327 0.0327 0.0488 0.0369 ...
 $ log10Pb: num  2.52 2.29 2.47 2.61 2.37 ...
> head(dvar)
   log10M log10L     IC log10Pb
1 -0.4660   1.43 0.0263    2.52
2  0.3304   1.66 0.0327    2.29
3  0.2923   1.68 0.0327    2.47
4  0.2148   1.63 0.0488    2.61
5  0.2253   1.64 0.0369    2.37
```

Données en fonction de chaque régresseur potentiel





Modèle linéaire incluant les 2 régresseurs

```
> m.IC.L <- lm(log10Pb ~ log10L + IC, data = dvar)
> summary(m.IC.L)

Call:
lm(formula = log10Pb ~ log10L + IC, data = dvar)

Residuals:
    Min      1Q  Median      3Q     Max
-0.2048 -0.0976  0.0225  0.0613  0.3522

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    3.478      0.921    3.78   0.0026
log10L        -0.740      0.546   -1.36   0.2001
IC             4.940      1.652    2.99   0.0113

Residual standard error: 0.148 on 12 degrees of freedom
Multiple R-squared:  0.63,      Adjusted R-squared:  0.569
F-statistic: 10.2 on 2 and 12 DF,  p-value: 0.00255
```

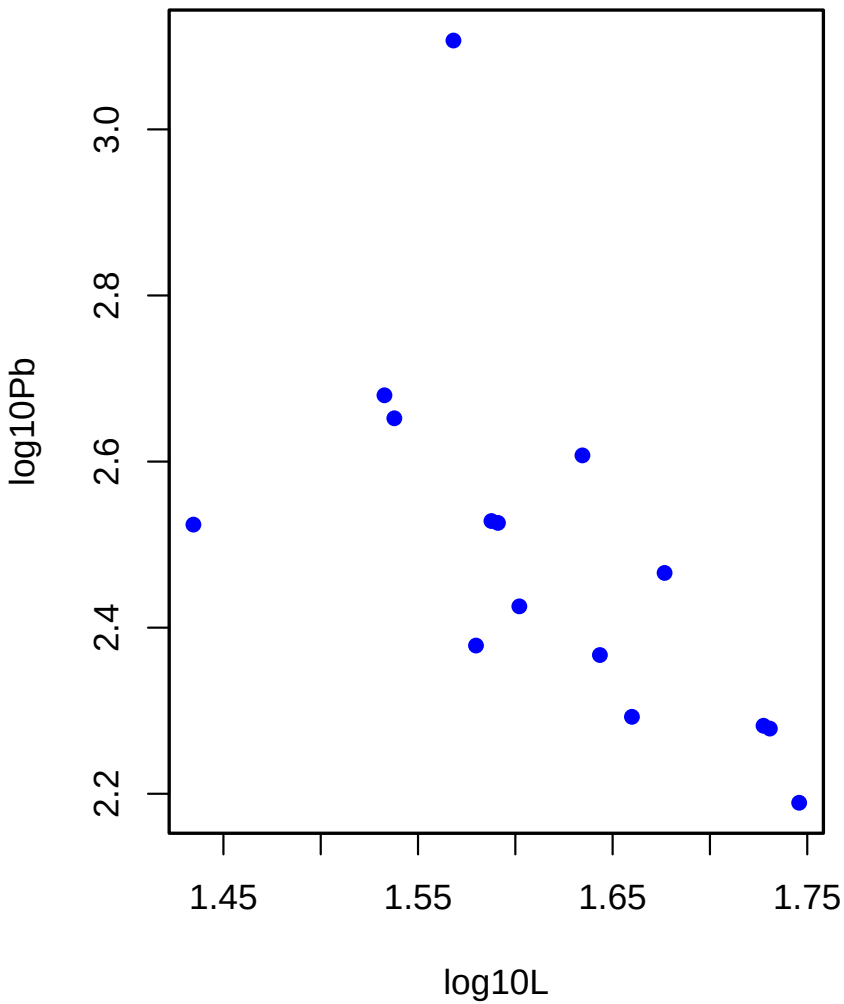
Interprétation des coefficients du modèle

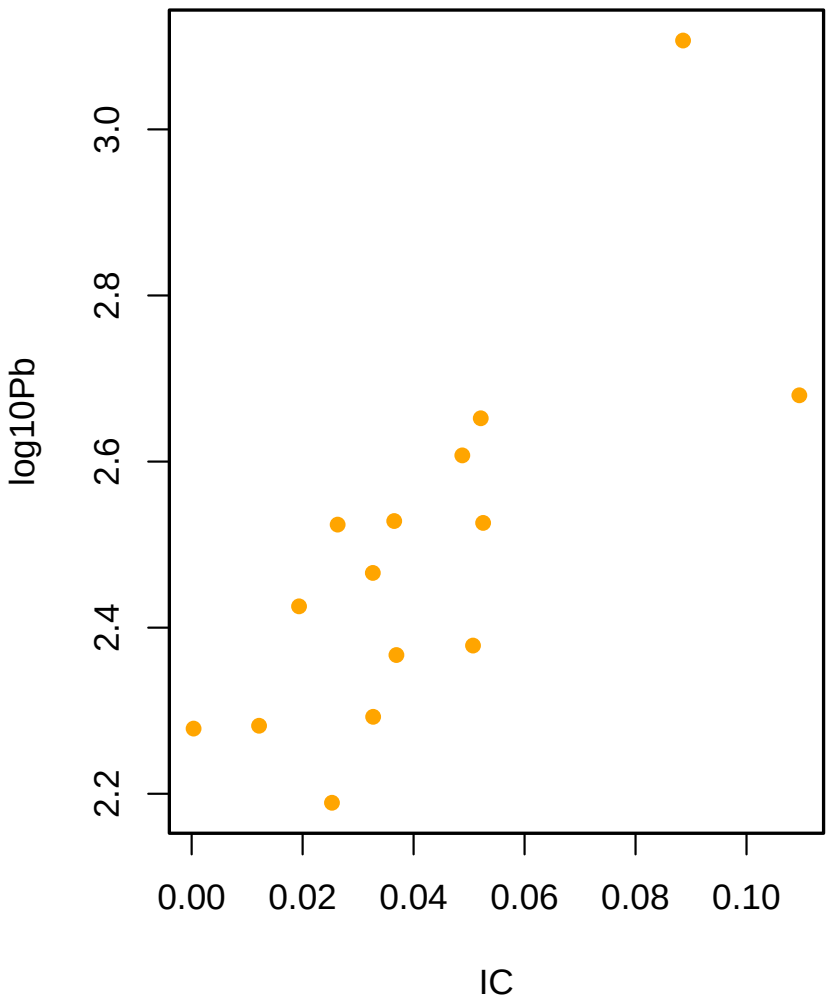
```
> coef(m.IC.L)
(Intercept)       log10L           IC
       3.48        -0.74         4.94
```

Interprétation en prenant en compte la variabilité de chaque régresseur dans le jeu de données
(ici en multipliant par l'écart type du régresseur) :

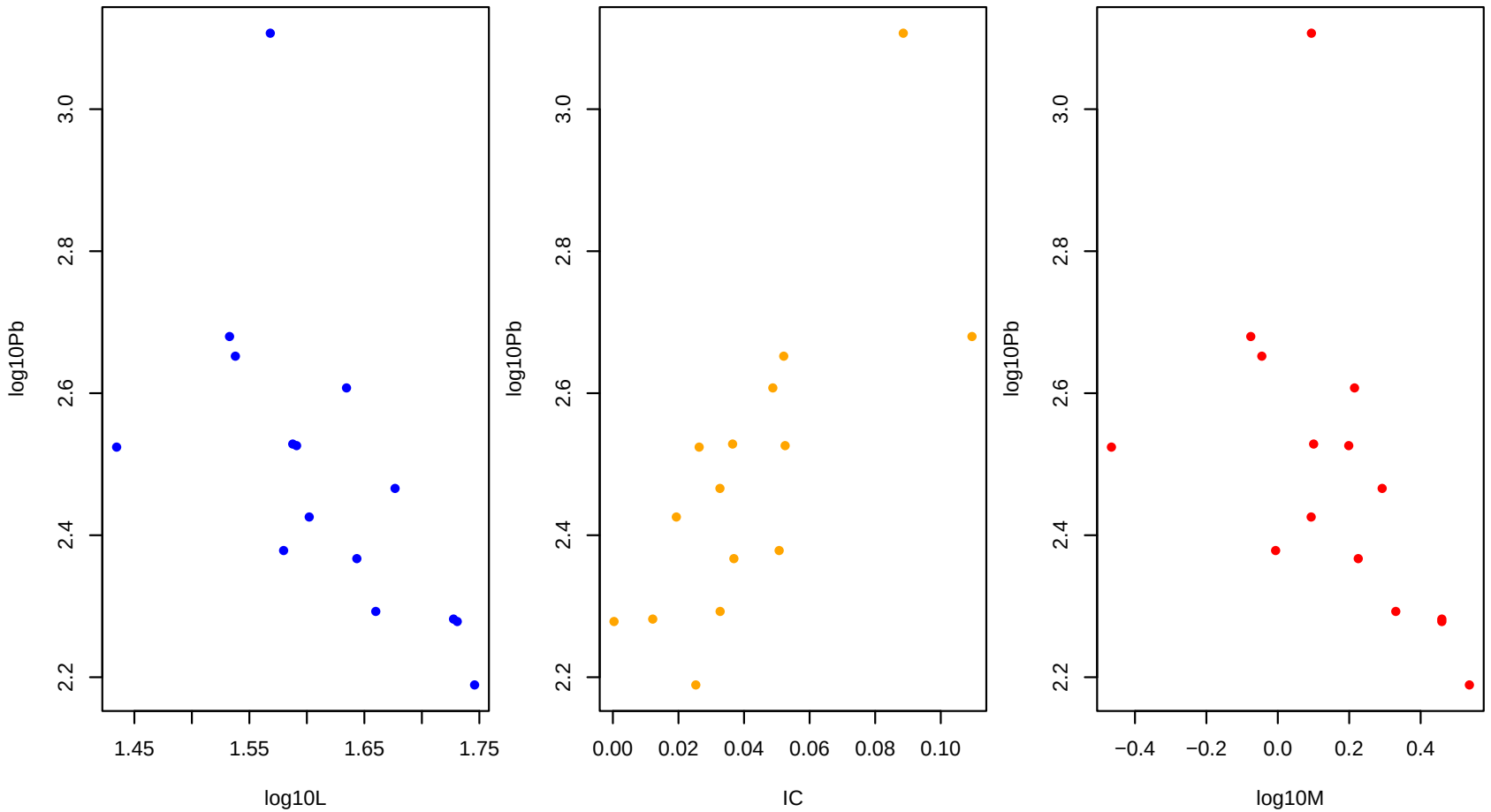
```
> coef(m.IC.L)["log10L"] * sd(dvar$log10L)
 log10L
-0.0627
> coef(m.IC.L)["IC"] * sd(dvar$IC)
   IC
0.138
```

Variable observée en fonction de chacun des régresseurs





Inclusion d'un 3ème régresseur : la masse corporelle M



Modèle linéaire incluant 3 régresseurs

```
> m.IC.L.M <- lm(log10Pb ~  IC + log10L + log10M, data = dvar)
> summary(m.IC.L.M)

Call:
lm(formula = log10Pb ~ IC + log10L + log10M, data = dvar)

Residuals:
     Min       1Q   Median       3Q      Max
-0.1599 -0.1180  0.0109  0.0626  0.3240

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    8.085      4.544    1.78    0.103
IC             3.838      1.961    1.96    0.076
log10L        -3.653      2.866   -1.27    0.229
log10M         0.927      0.895    1.04    0.323

Residual standard error: 0.147 on 11 degrees of freedom
Multiple R-squared:  0.663,        Adjusted R-squared:  0.571
F-statistic: 7.22 on 3 and 11 DF,  p-value: 0.006
```

Interprétation des coefficients du modèle

```
> coef(m.IC.L.M)
```

(Intercept)	IC	log10L	log10M
8.085	3.838	-3.653	0.927

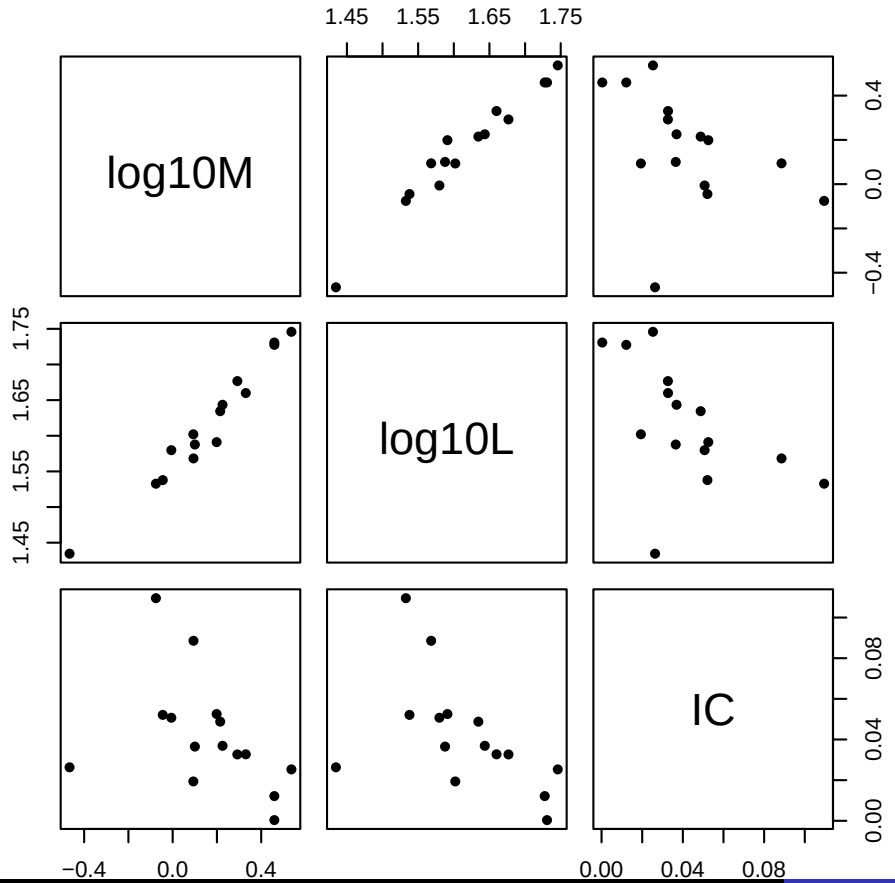
Signe contre-intuitif du coefficient associé au régresseur $log10M$.
D'où cela vient-il ?
A noter par ailleurs qu'aucun coefficient ne semble significativement différent de 0.

```
> confint(m.IC.L.M)
```

	2.5 %	97.5 %
(Intercept)	-1.917	18.09
IC	-0.479	8.15
log10L	-9.962	2.66
log10M	-1.044	2.90

Difficulté d'interprétation des coefficients du modèle due à la corrélation entre les régresseurs *log*10*M* et *log*10*L*

```
> pairs(dvar[1:3], pch = 16)
```



Difficulté liée aux régresseurs non contrôlés (données non issues d'un plan d'expérience)

Corrélation potentielle entre les régresseurs induisant

- une difficulté d'interprétation des coefficients du modèle
- et des coefficients non significativement différents de 0.

Nécessaire **sélection**,
si possible *a priori*,
des régresseurs inclus dans le modèle en **évitant les corrélations fortes entre régresseurs**.

A vous de jouer !

Consigne

Traitez l'exercice 1 en vous servant du guide sur notre site d'enseignement : `https://biostatistique.vetagro-sup.fr/guideRmodlin.pdf`.

Plan

Principe de parcimonie

Un modèle ne doit pas contenir plus de paramètres qu'il n'en faut pour décrire correctement les données.

- Il est dangereux d'essayer de faire rentrer dans la partie déterministe ce qui est stochastique.
- La surparamétrisation induit un manque de robustesse et rend le modèle peu généralisable.
- ATTENTION la part de variance expliquée (r^2) augmente toujours lorsqu'on ajoute des régresseurs.
- Les tests de modèles emboîtés et des critères d'ajustement pénalisés sont plus adaptés pour aider à sélectionner *a posteriori* les régresseurs.

Test F des modèles emboîtés

- Soit SCE_p la SCE du modèle le plus complet à p paramètres,
- soit SCE_q la SCE d'un modèle simplifié du modèle complet à q paramètres ($q < p$),

La statistique

$$F = \frac{(SCE_q - SCE_p) \times (n-p)}{SCE_p \times (p-q)}$$

suit la loi de Fisher et Snedecor de degré de liberté $p - q$ et $n - p$ sous l'hypothèse d'équivalence des 2 modèles.
Le rejet de H_0 indique que le modèle complet décrit significativement mieux les données que le modèle partiel.

Test des modèles emboîtés avec **R** sur exemple précédent

```
> anova(m.IC.L, m.IC.L.M)
Analysis of Variance Table

Model 1: log10Pb ~ log10L + IC
Model 2: log10Pb ~ IC + log10L + log10M
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     12 0.263
2     11 0.239  1    0.0233 1.07   0.32
```

L'ajout du régresseur $log10M$ ne permet pas de décrire significativement mieux les données.

Comparaison du modèle à 3 régresseurs avec le modèle prenant en compte uniquement *IC*

```
> m.IC <- lm(log10Pb ~ IC, data = dvar)
> anova(m.IC, m.IC.L.M)
Analysis of Variance Table

Model 1: log10Pb ~ IC
Model 2: log10Pb ~ IC + log10L + log10M
  Res.Df   RSS Df Sum of Sq    F Pr(>F)
1     13 0.303
2     11 0.239  2    0.0635 1.46   0.27
```

L'ajout des régresseurs $log10M$ et $log10L$ ne permet pas de décrire significativement mieux les données.

La vraisemblance (likelihood) et la déviance

La **vraisemblance** est généralement exprimée en log :

$$logLik = ln(Pr(y \mid \beta, \sigma^2))$$

La **déviance** est le double de la différence de vraisemblance du modèle saturé (modèle décrivant exactement les données) et du modèle ajusté.

$$Dev = 2 \times (logLik_{saturated.model} - logLik) = -2logLik + constante$$

La **déviance ne peut être utilisée que pour comparer plusieurs modèles ajustés sur un même jeu de données**, donc la calculer à une constante près ($-2logLik$) est suffisant.
Plus la déviance est faible, meilleur est l'ajustement aux données.

Le critère d'information pénalisé d'Akaïké (AIC)

L'AIC fait partie des critères d'information, dont l'objectif est d'aider à trouver un modèle qui soit un bon compromis entre qualité d'ajustement et complexité, donc un **modèle parcimonieux**. Le plus connu d'entre eux est l'AIC.
L'AIC est une **déviance pénalisée par un terme de complexité du modèle** égal au double du nombre de paramètres estimés.

$$AIC = -2logLik + 2 \times p$$

avec p le nb. de paramètres du modèle (constante et σ compris)
Plus ce critère est faible, meilleur le modèle est considéré.

Origine de l'AIC

Les capacités théoriques de l'AIC à choisir un bon modèle ont été démontrées dans le cadre de l'**utilisation du modèle en prédiction**.
La **pénalisation de la déviance** dans la définition de l'AIC sert à **corriger le biais d'optimisme** (sous-estimation des erreurs en prédiction) dû au fait que la déviance est calculée sur les données qui ont servi à ajuster le modèle, et non sur des données indépendantes, ce qui constituerait une vraie validation en prédiction.
L'AIC est donc adapté aux cas on l'on souhaite construire un modèle prédictif.

Autres critères d'information pénalisés

- **Critère d'Akaïké corrigé**
  Il s'agit d'une correction de l'AIC recommandée quand n le nombre d'observations est faible par rapport au nombre de paramètres à estimer du modèle p ($< 40 \times p$)

$$AICc = -2logLik + 2p + \frac{2p(p+1)}{n-p-1}$$

- **Critère d'information bayésien de Schwarz**
  Le critère d'information de Scharz propose une pénalisation plus sévère, et tend donc à sélectionner des modèles plus simples.

$$BIC = -2logLik + ln(n) \times p$$

.

Comparaison des AIC de tous les modèles possibles

```
> m.IC.M <- lm(log10Pb ~  IC + log10M, data = dvar)
> m.L.M <- lm(log10Pb ~  log10L + log10M, data = dvar)
> m.L <- lm(log10Pb ~  log10L, data = dvar)
> m.M <- lm(log10Pb ~  log10M, data = dvar)
> AIC(m.IC.L.M, m.IC.L, m.IC, m.IC.M, m.L.M, m.L, m.M)
```

	df	AIC
m.IC.L.M	5	-9.51
m.IC.L	4	-10.11
m.IC	3	-9.97
m.IC.M	4	-9.44
m.L.M	4	-7.03
m.L	3	-3.76
m.M	3	-1.36

Le modèle à deux régresseurs avec IC et L semblerait meilleur, mais de si peu.

Comparaison des BIC de tous les modèles possibles

```
> AIC(m.IC.L.M, m.IC.L, m.IC, m.IC.M, m.L.M, m.L, m.M,
      k = log(nrow(dvar)))
         df    AIC
m.IC.L.M  5 -5.967
m.IC.L    4 -7.280
m.IC      3 -7.850
m.IC.M    4 -6.609
m.L.M     4 -4.194
m.L       3 -1.633
m.M       3  0.769
> # code équivalent
> #BIC(m.IC.L.M, m.IC.L, m.IC, m.IC.M, m.L.M, m.L, m.M)
```

Le modèle à un seul régresseur IC serait choisi, là encore de très peu.

Méthodes de sélection automatique de variables

Il n'est pas toujours possible d'écrire tous les modèles quand le nombre de régresseurs potentiel augmente
(trop grand nombre de modèles).

Il existe des méthodes automatiques de sélection de variables basées sur l'un des critères vus précédemment (test F, AIC, BIC).
La **sélection** peut-être

- **descendante** (on part du modèle le plus complet),
- **ascendante** (on part du modèle le plus simple et on indique le modèle le plus complet),
- **"bidirectionnelle"** à chaque étape on s'autorise à ajouter ou enlever une variable (le plus efficace pour améliorer le critère), en partant du modèle nul, du modèle complet, ou d'un modèle intermédiaire.

Exemple de sélection descendante sur la base de l'AIC

```
> selecdesc <- step(m.IC.L.M, trace = 0)
> selecdesc

Call:
lm(formula = log10Pb ~ IC + log10L, data = dvar)

Coefficients:
(Intercept)           IC       log10L
       3.48         4.94        -0.74
```

Le modèle à deux régresseurs IC et L serait choisi.

Regardons de plus près comment ça fonctionne

```
> step(m.IC.L.M)
Start:  AIC=-54.1
log10Pb ~ IC + log10L + log10M

          Df Sum of Sq   RSS   AIC
- log10M   1    0.0233 0.263 -54.7
<none>                 0.239 -54.1
- log10L   1    0.0353 0.275 -54.0
- IC       1    0.0833 0.323 -51.6

Step:  AIC=-54.7
log10Pb ~ IC + log10L

          Df Sum of Sq   RSS   AIC
<none>                 0.263 -54.7
- log10L   1    0.0402 0.303 -54.5
- IC       1    0.1957 0.458 -48.3

Call:
lm(formula = log10Pb ~ IC + log10L, data = dvar)

Coefficients:
```

Exemple de sélection ascendante sur la base de l'AIC

```
> m.0 <- lm(log10Pb ~  1, data = dvar)
> selecasc <- step(m.0, scope = log10Pb ~  IC + log10L + log10M,
                   direction = "forward", trace = 0)
> selecasc

Call:
lm(formula = log10Pb ~ IC + log10L, data = dvar)

Coefficients:
(Intercept)           IC        log10L
       3.48         4.94         -0.74
```

Le modèle à deux régresseurs IC et L serait à nouveau choisi.
Il n'est néanmoins pas du tout systématique d'obtenir le même résultat avec les deux méthodes.

Regardons comment ça fonctionne en partant de m.IC

```
> step(m.IC, scope = log10Pb ~  IC + log10L + log10M, direction = "forward")
Start:  AIC=-54.5
log10Pb ~ IC

         Df Sum of Sq   RSS   AIC
+ log10L  1    0.0402 0.263 -54.7
<none>                0.303 -54.5
+ log10M  1    0.0282 0.275 -54.0

Step:  AIC=-54.7
log10Pb ~ IC + log10L

         Df Sum of Sq   RSS   AIC
<none>                0.263 -54.7
+ log10M  1    0.0233 0.239 -54.1

Call:
lm(formula = log10Pb ~ IC + log10L, data = dvar)

Coefficients:
(Intercept)           IC       log10L
      3.48         4.94         -0.74
```

Exemple de sélection "bidirectionnelle" sur la base de l'AIC

```
> m.0 <- lm(log10Pb ~  1, data = dvar)
> selecboth <- step(m.0, scope = log10Pb ~  IC + log10L + log10M,
                    direction = "both", trace = 0)
> selecboth

Call:
lm(formula = log10Pb ~ IC + log10L, data = dvar)

Coefficients:
(Intercept)           IC        log10L
       3.48         4.94         -0.74
```

A nouveau dans cet exemple très simple le même modèle final est obtenu, ce qui n'est pas du tout systématique.

Regardons comment ça fonctionne en partant de m.IC

```
> step(m.IC, scope = log10Pb ~  IC + log10L + log10M, direction = "both")
Start:  AIC=-54.5
log10Pb ~ IC

          Df Sum of Sq   RSS   AIC
+ log10L   1     0.040 0.263 -54.7
<none>                 0.303 -54.5
+ log10M   1     0.028 0.275 -54.0
- IC       1     0.408 0.710 -43.7

Step:  AIC=-54.7
log10Pb ~ IC + log10L

          Df Sum of Sq   RSS   AIC
<none>                 0.263 -54.7
- log10L   1    0.0402 0.303 -54.5
+ log10M   1    0.0233 0.239 -54.1
- IC       1    0.1957 0.458 -48.3

Call:
lm(formula = log10Pb ~ IC + log10L, data = dvar)
```

Limites de ces méthodes

- Les différentes méthodes (AIC, BIC, test F) donnent souvent des résultats différents. Laquelle utiliser ?
- On dit qu'une **différence d'AIC** (ou BIC) commence à donner des arguments en faveur d'un modèle si elle dépasse 2 et devient **vraiment probante si elle dépasse 10** donc une procédure automatique qui ne tient pas compte de l'ampleur de différences d'AIC est-elle raisonnable ?

Il ne semble pas raisonnable d'utiliser aveuglément une méthode automatique pour choisir un modèle (surtout un modèle explicatif) et l'expertise biologique semble capitale dans ce choix.

Quelques références très citées pour aller plus loin sur ce sujet délicat

Aucune de ces références ne propose une stratégie complètement automatisable, qui ne fasse pas à un moment appel à l'expertise biologique, et toutes proposent des stratégies relativement complexes. $\Rightarrow$ Les auteurs ne suivent souvent pas la totalité des conseils donnés dans ces articles.

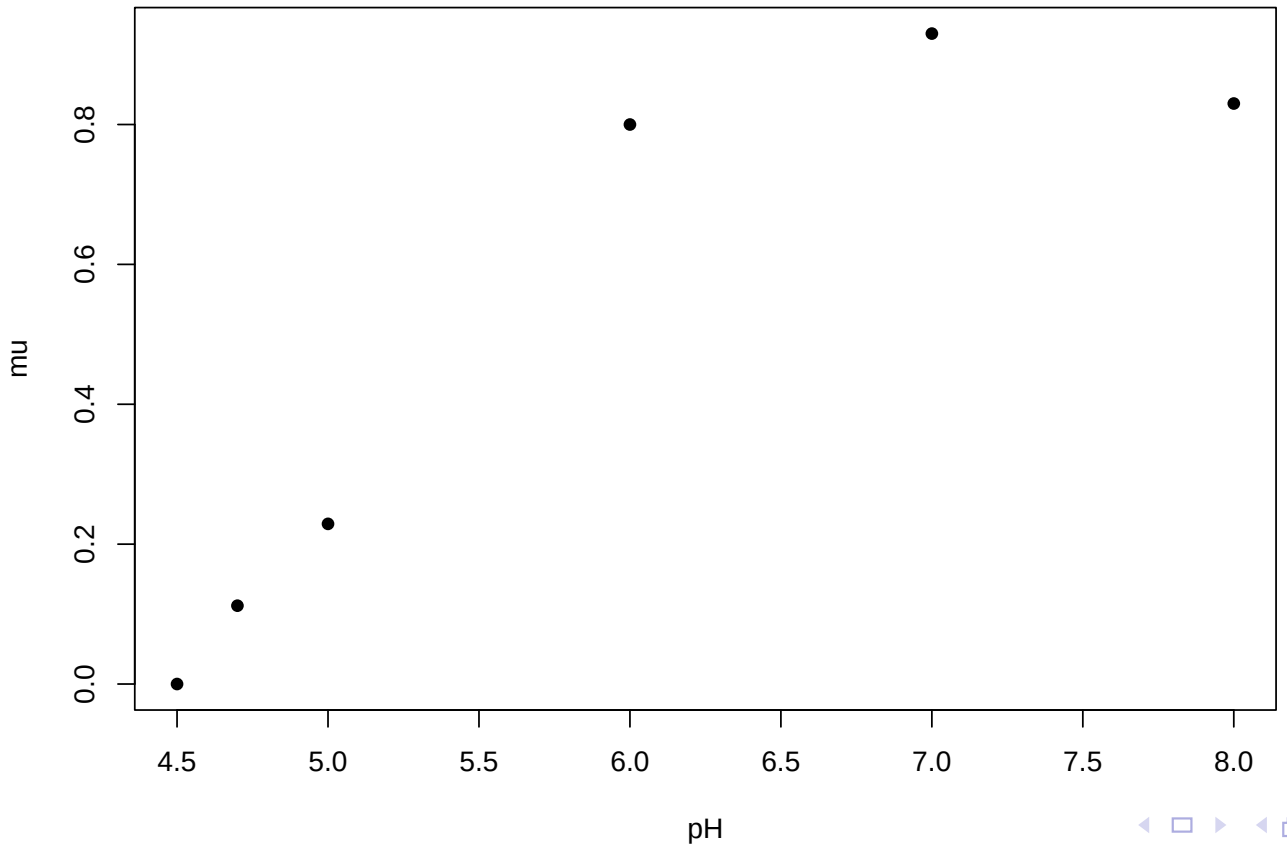
- Gelman, A., Hill, J. (2006). Data analysis using regression and multilevel/hierarchical models. Cambridge university press.
- Bursac, Z., Gauss, C. H., Williams, D. K., Hosmer, D. W. (2008). Purposeful selection of variables in logistic regression. Source code for biology and medicine, 3(1), 1-8.
- Hosmer, D. W., Lemeshow, S., Sturdivant, R. X. (2013). Applied logistic regression (Vol. 398). John Wiley and Sons.
- Harrison, X. A. *et al.* (2018). A brief introduction to mixed effects modelling and multi-model inference in ecology. PeerJ, 6, e4794.

A retenir

- Il n'est pas prudent d'utiliser plus de paramètres que nécessaire pour décrire correctement les données (**principe de parcimonie**)
  car la **surparamétrisation induit un manque de robustesse** et un modèle peu généralisable.
- Les tests des modèles emboîtés et critères d'information peuvent être utiles mais **ne peuvent en aucun cas remplacer l'oeil expert du biologiste**.
- Une stratégie possible : **classer les modèles suivant leur AIC(c)** et faire son **choix entre les meilleurs sur des arguments biologiques**.
- Il est indispensable de s'assurer de la **pertinence biologique** du modèle ET du respect de ses **conditions d'utilisation** !

Que faire lorsque la relation n'est pas linéaire ?

Exemple de la modélisation de l'effet du pH du milieu de culture sur un taux de croissance microbien.



Ajustement d'un modèle polynomial

Essai d'ajustement d'un modèle polynomial de degré 2 :

$$\mu = \beta_0 + \beta_1 \times pH + \beta_2 \times pH^2 + \epsilon \text{ avec } \epsilon \sim N(0, \sigma)$$

On reste dans le cadre de la régression linéaire multiple avec deux régresseurs corrélés, pH et pH^2.

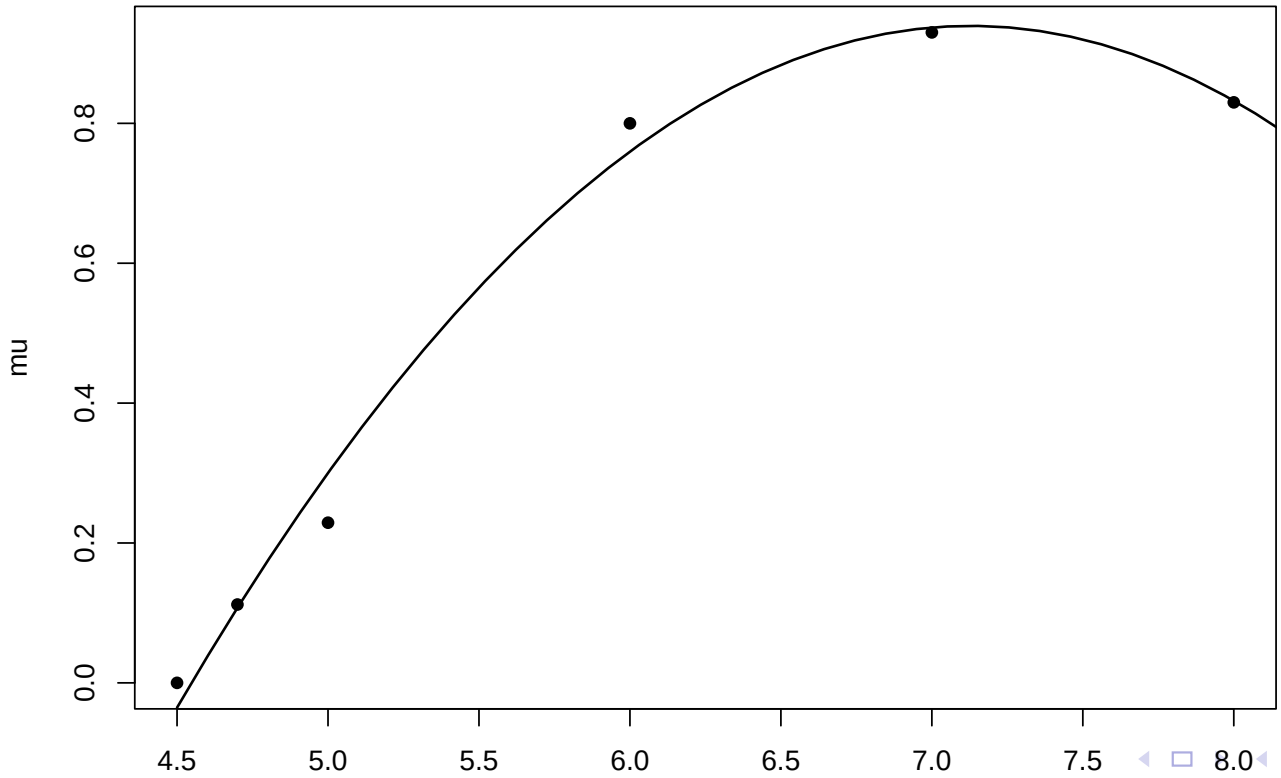
```
> (mpH2 <- lm(mu ~ pH + I(pH^2), data = dpH))

Call:
lm(formula = mu ~ pH + I(pH^2), data = dpH)

Coefficients:
(Intercept)           pH      I(pH^2)
     -6.229        2.011        -0.141
```

Résumé des résultats d'ajustement du modèle

```
> summary(mpH2)

Call:
lm(formula = mu ~ pH + I(pH^2), data = dpH)

Residuals:
       1        2        3        4        5        6
 0.03542  0.00474 -0.07111  0.04038 -0.00704 -0.00239

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -6.2286     0.6753   -9.22   0.0027
pH            2.0109     0.2267    8.87   0.0030
I(pH^2)      -0.1410     0.0183   -7.72   0.0045

Residual standard error: 0.0517 on 3 degrees of freedom
Multiple R-squared:  0.991,        Adjusted R-squared:  0.984
F-statistic:  159 on 2 and 3 DF,  p-value: 0.000907
```

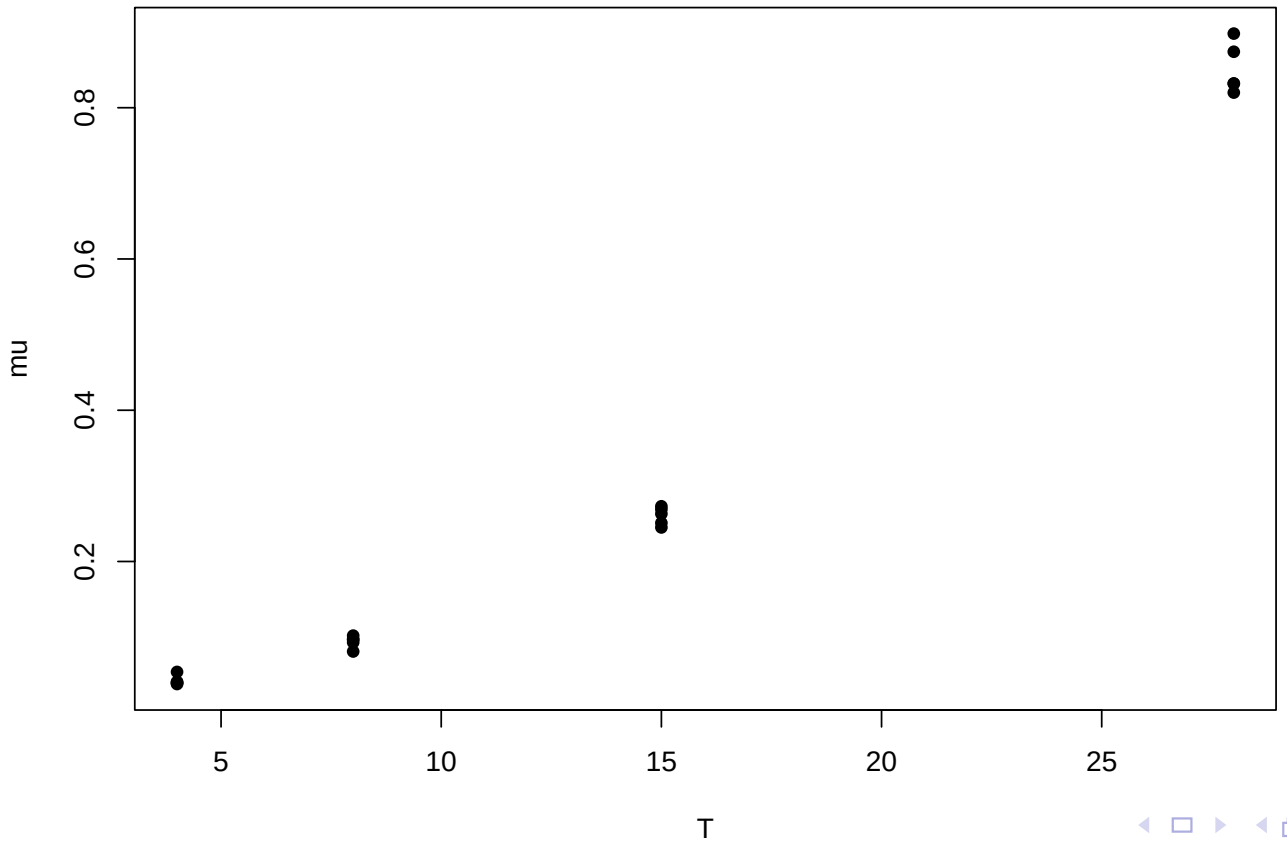
Visualisation de l'ajustement du modèle

```
> plot(mu ~ pH, pch = 16, data = dpH)
> xpH <- data.frame(pH = seq(4.5, 9.5, length.out = 50))
> lines(xpH$pH, predict(mpH2, newdata = xpH), lwd = 1.5)
```



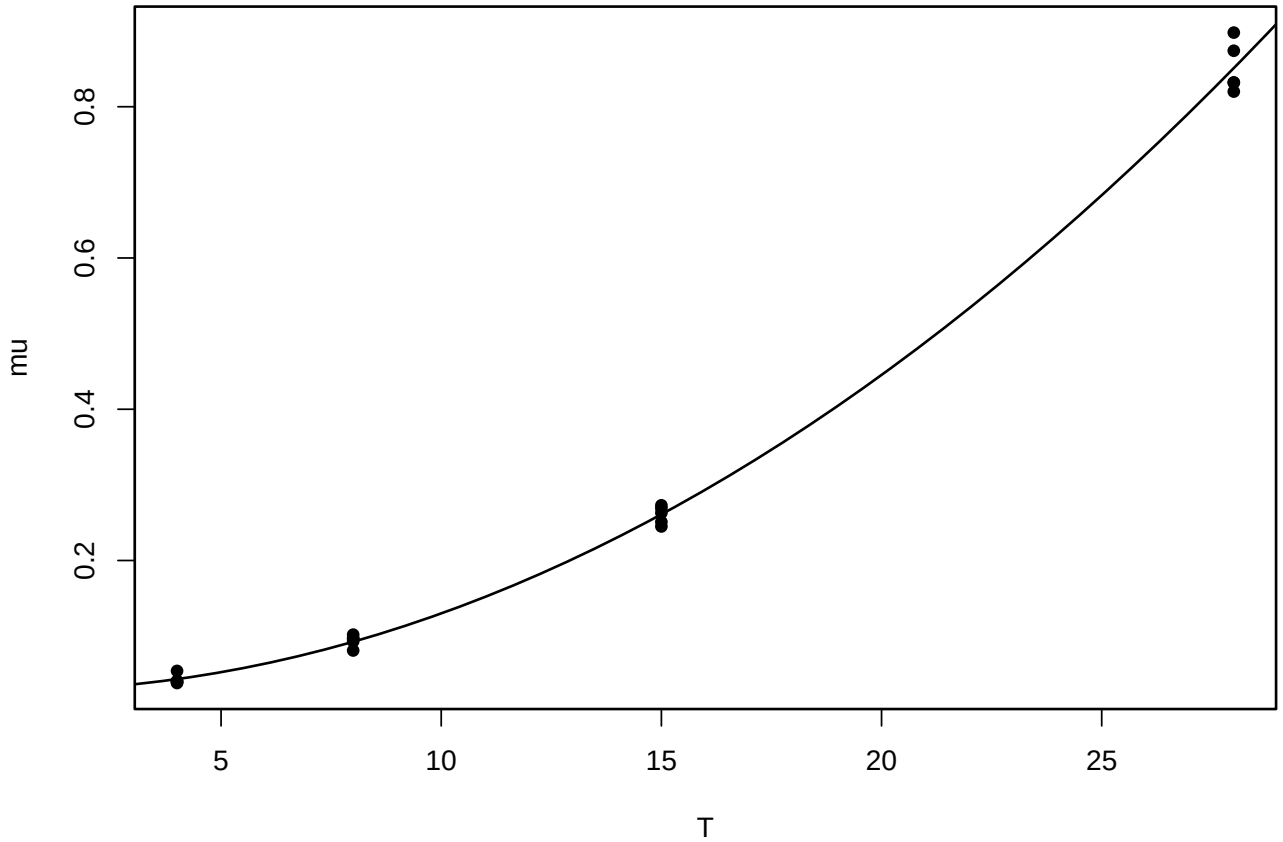
Autre exemple nécessitant un méthode alternative

Exemple de la modélisation de l'effet de la température du milieu de culture sur un taux de croissance microbien.



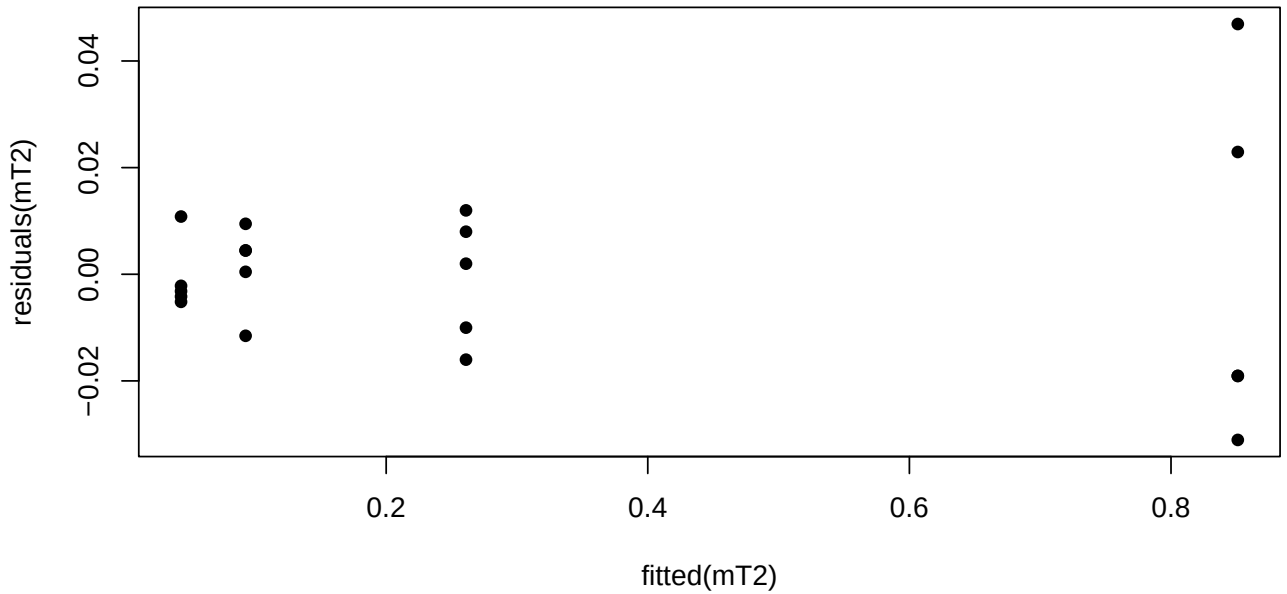
Essai d'ajustement d'un modèle polynomial de degré 2

```
> mT2 <- lm(mu ~ T + I(T^2), data = dT)
```



Regardons de plus près les résidus

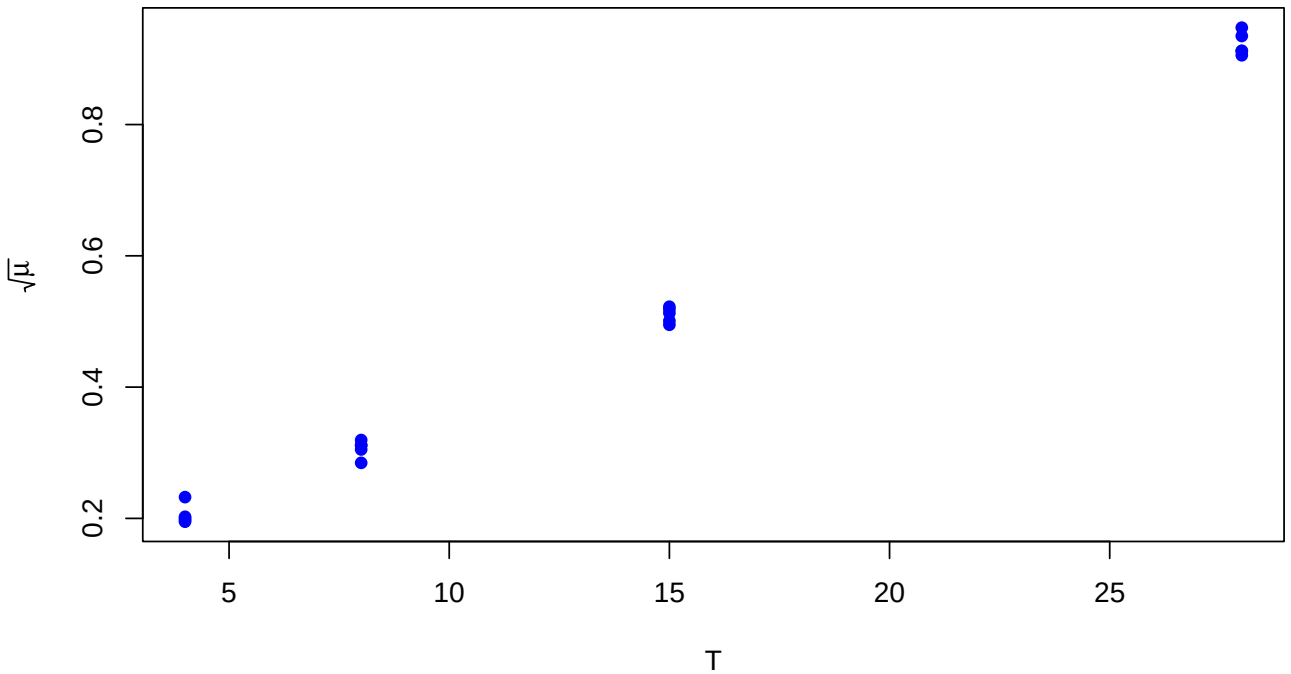
```
> plot(residuals(mT2) ~ fitted(mT2), pch = 16)
```



On observe une nette augmentation de l'amplitude des résidus lorsque le taux de croissance augmente.

Une alternative envisageable : la transformation de variable

Transformation du taux de croissance en racine carrée qui sur cet exemple linéarise la relation et stabilise la variance.

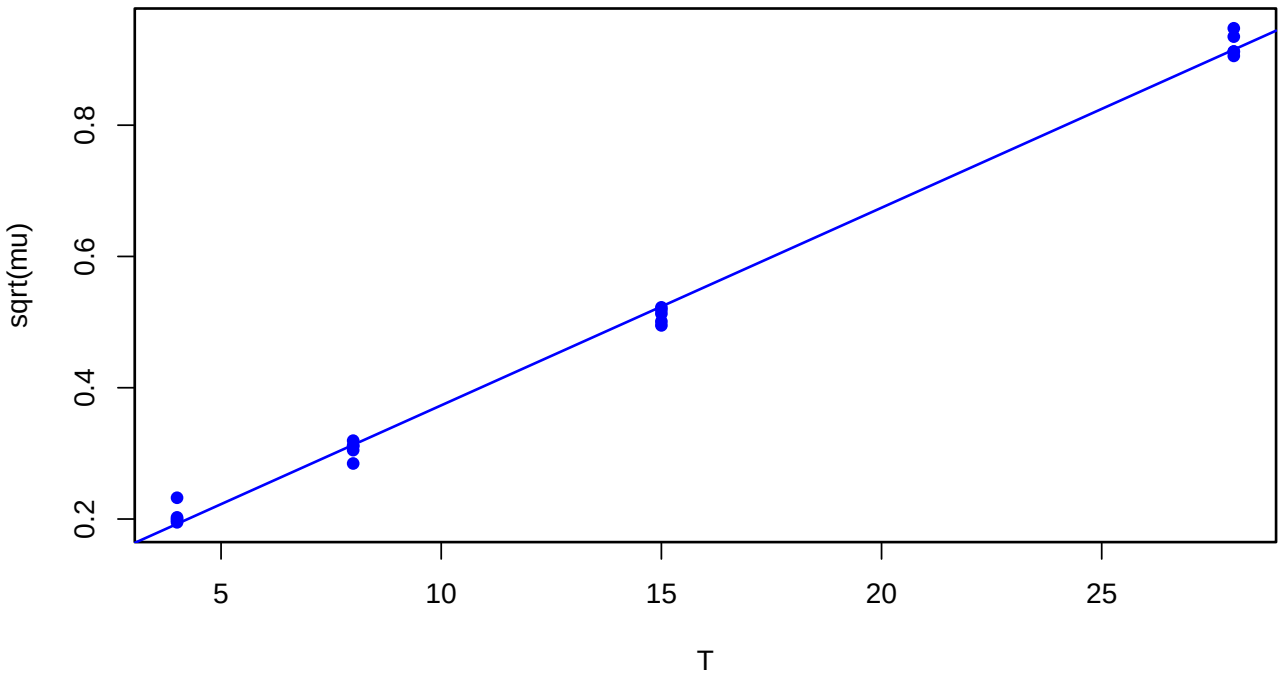


Ajustement d'un modèle linéaire sur $\sqrt{\mu}$

$$\sqrt{\mu} = \beta_0 + \beta_1 \times T + \epsilon \text{ avec } \epsilon \sim N(0, \sigma)$$

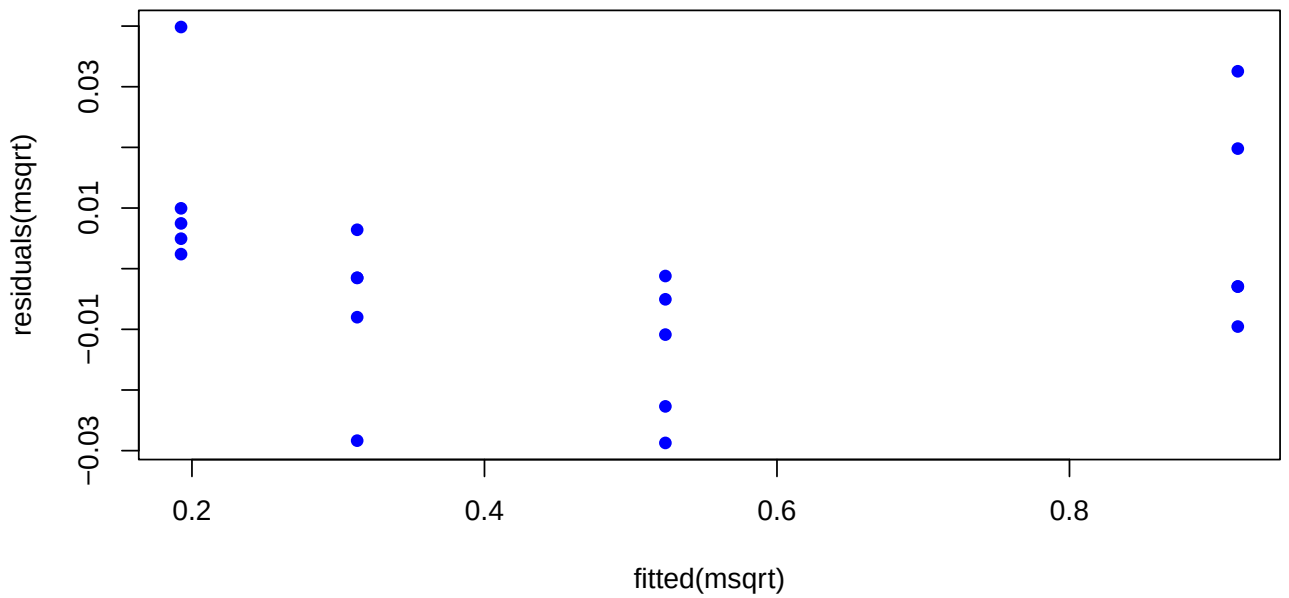
On reste dans le cadre de la régression linéaire multiple avec un régresseur, T, et une variable observée transformée, $\sqrt{\mu}$.

```
> msqrt <- lm(sqrt(mu) ~ T, data = dT)
```



Regardons les résidus avec cette alternative

```
> plot(residuals(msqrt) ~ fitted(msqrt), pch = 16, col = "blue")
```



Ce n'est pas parfait mais mieux : on n'a plus d'hétéroscédasticité mais une légère tendance à la non linéarité.

Transformations de variables classiques à explorer

Quelques **transformations de variables classiquement utilisées** pour tenter de se ramener à un **modèle linéaire gaussien** :

- tranformation **logarithmique** (quand la variable est strictement positive et varie sur plusieurs ordres de grandeurs),
- transformation **en racine carrée** (notamment quand les données sont des comptages),
- tranformation **logit** ($logit(y) = \ln \frac{y}{1-y}$ lorsque y est une variable strictement comprise entre 0 et 1).

On n'arrive pas à se ramener à un modèle linéaire gaussien dans tous les cas. On utilise un **modèle non linéaire** et/ou un **modèle linéaire généralisé** (non gaussien) dans les autres cas.

A vous de jouer !

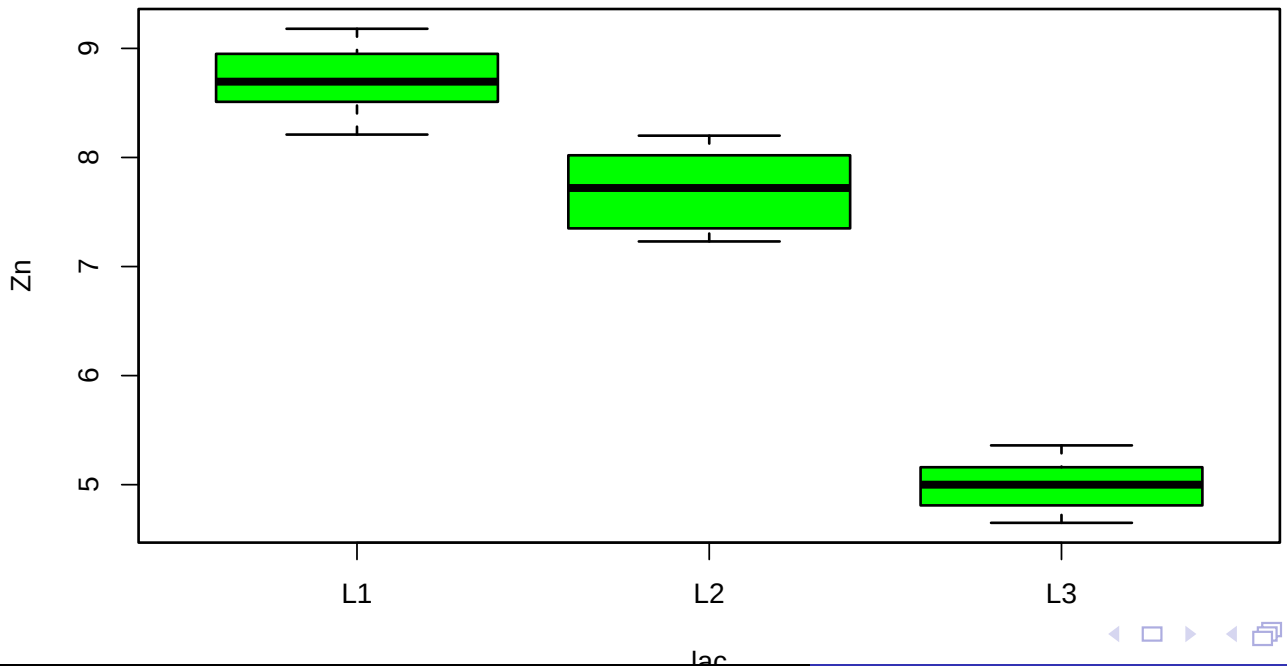
Consigne

Reprenez l'exercice 1 pour comparez les deux modèles avec l'AIC et le BIC, puis traitez les exercices 2 et 3, puis pour les plus rapides les exercices 7 et 8.

Plan

Quand la variable explicative est qualitative - ANOVA et modèle linéaire

Exemple de l'étude de la concentration en zinc dans des moules en fonction du lac d'origine
(trois lacs correspondant à trois niveaux d'exposition différents).



Codage d'une variable explicative qualitative dans un modèle linéaire

Codage d'une variable explicative qualitative à partir de variables indicatrices.

- Ex. de codage d'une variable à 2 modalités :
  $sexe = 1$ si femelle, 0 si mâle.
- Ex. de codage d'une variable à 3 modalités
  (cf. exemple précédent) :
  $X_1 = 1$ si lac L1
  $X_2 = 1$ si lac L2
  et donc si $X_1 = X_2 = 0$ c'est le lac L3.
- Dans le cas général il suffit de $p - 1$ variables indicatrices pour coder une variable à p modalités.

le modèle d'ANOVA 1 : un modèle linéaire gaussien

- Ecriture classique d'un modèle d'ANOVA 1 avec $p = 3$:
  $Y_{ij} = \mu + \alpha_i + \epsilon_{ij}$ avec $\epsilon_{ij} \sim N(0, \sigma)$ et $\alpha_1 + \alpha_2 + \alpha_3 = 0$
- Ecriture sous forme d'un modèle linéaire à l'aide des deux variables indicatrices X_1 et X_2 codant pour les modalités 1 et 2 du facteur étudié :
  $Y_k = \beta_0 + \beta_1 X_1 k + \beta_2 X_2 k + \epsilon_k$ avec $\epsilon_k \sim N(0, \sigma)$
- Lien entre les deux écritures :
  $\mu + \alpha_1 = \beta_0 + \beta_1$
  $\mu + \alpha_2 = \beta_0 + \beta_2$
  $\mu + \alpha_3 = \beta_0$
  $\mu = \beta_0 + \frac{\beta_1 + \beta_2}{3}$

Contrainte nécessaire sur les α_i dans une écriture de type ANOVA

- Ecriture sous forme de modèle linéaire avec 4 paramètres : β_0, β_1, β_2, σ
- Ecriture de type ANOVA avec 5 paramètres (un de trop) : μ, α_1, α_2, α_3, σ
  paramètres non identifiables, d'où la nécessité d'une contrainte.
  Contraintes classiques :
  - contrainte de type somme : $\sum \alpha_i = 0$
    (μ moyenne globale)
  - contrainte de type cellule de référence
    (ex. $\alpha_1 = 0$, μ moyenne du groupe de référence)

Définition des contraintes dans **R**

- **Contrainte définie par défaut de type cellule de référence** :

  ```
  > options(contrasts = c("contr.treatment", "contr.treatment"))
  ```

  On peut choisir le groupe de référence en changeant l'ordre des modalités du facteur.
  Ex. pour prendre le lac L2 comme groupe de référence :

  ```
  dZn$lac <- factor(dZn$lac, levels = c("L2", "L3", "L1"))
  ```

- **Contrainte de type somme aussi utilisable à l'aide de l'option suivante** :

  ```
  options(contrasts = c("contr.sum", "contr.sum"))
  ```

ANOVA avec la fonction `lm()`

Deux façons d'obtenir le tableau d'ANOVA :

```
> a <- aov(Zn ~ lac, data = dZn)
> summary(a)
```

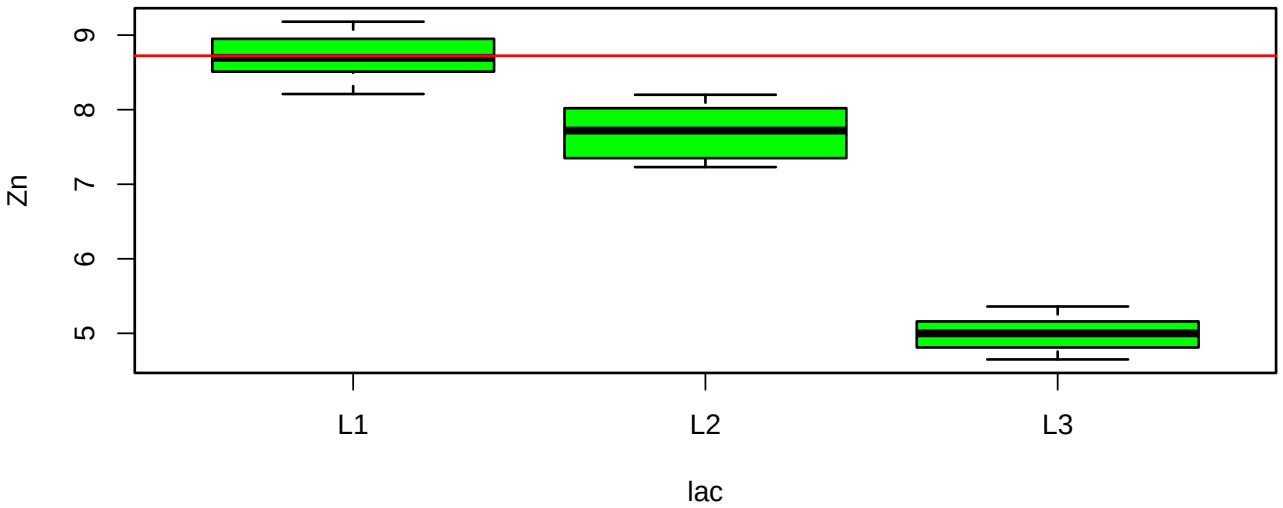
ou

```
> m <- lm(Zn ~ lac, data = dZn)
> anova(m)
```

```
Analysis of Variance Table

Response: Zn
          Df Sum Sq Mean Sq F value Pr(>F)
lac        2   74.1    37.1     378 <2e-16
Residuals 27    2.6     0.1
```

Interprétation des coefficients du modèle linéaire avec la contrainte de type cellule de référence

```
> coef(m)
(Intercept)        lacL2        lacL3
       8.72        -1.04        -3.73
```

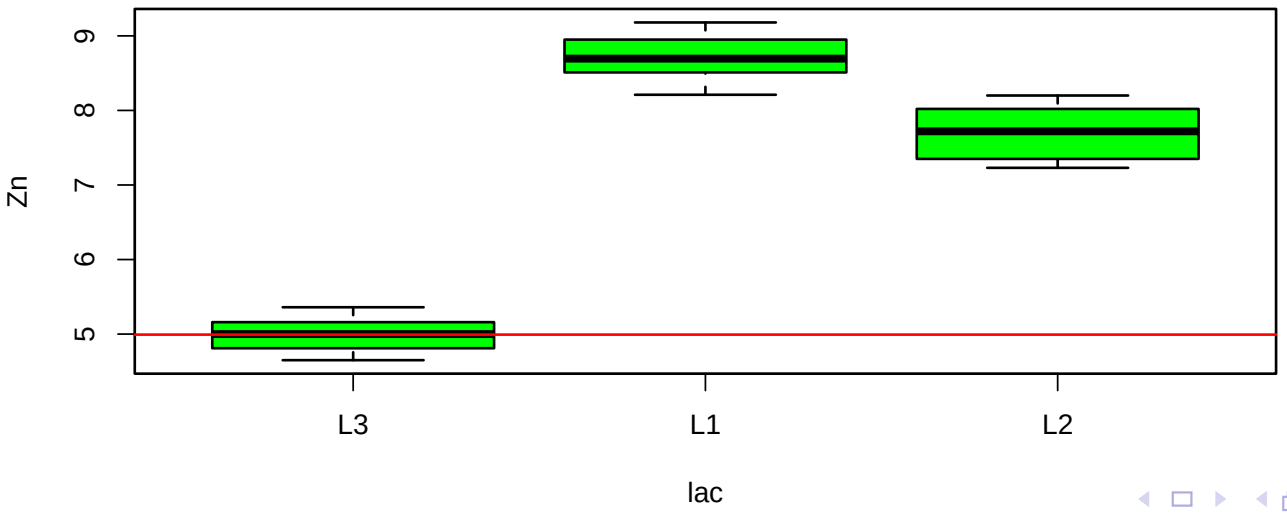


Contrainte de type cellule de référence avec changement du groupe de référence

On peut par exemple définir le site le moins contaminé comme référence :

```
> dZn$lac <- factor(dZn$lac, levels = c("L3", "L1", "L2"))
> m <- lm(Zn ~ lac, data = dZn)
> coef(m)
```

```
(Intercept)        lacL1        lacL2
       4.99         3.73         2.69
```

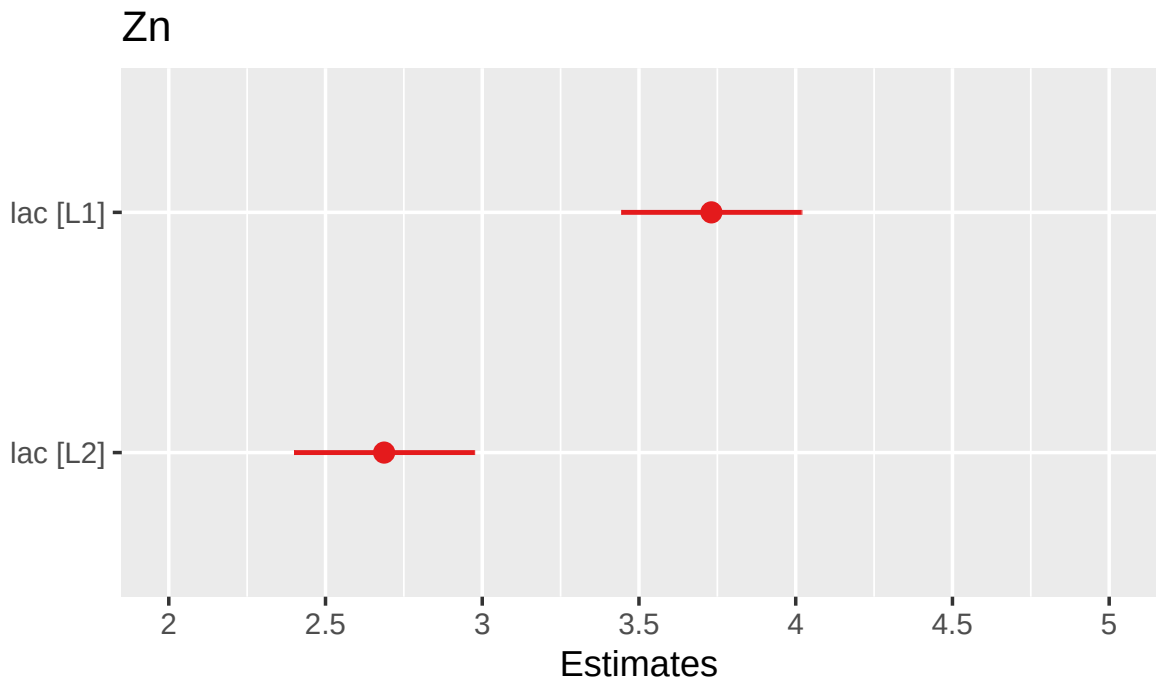


Intervalles de confiance associés au différences estimées par rapport au groupe de référence

```
> confint(m)
                2.5 % 97.5 %
(Intercept)  4.79   5.20
lacL1        3.44   4.02
lacL2        2.40   2.97
```

Représentation des différences estimées avec leur intervalle de confiance à l'aide du package `sjPlot`

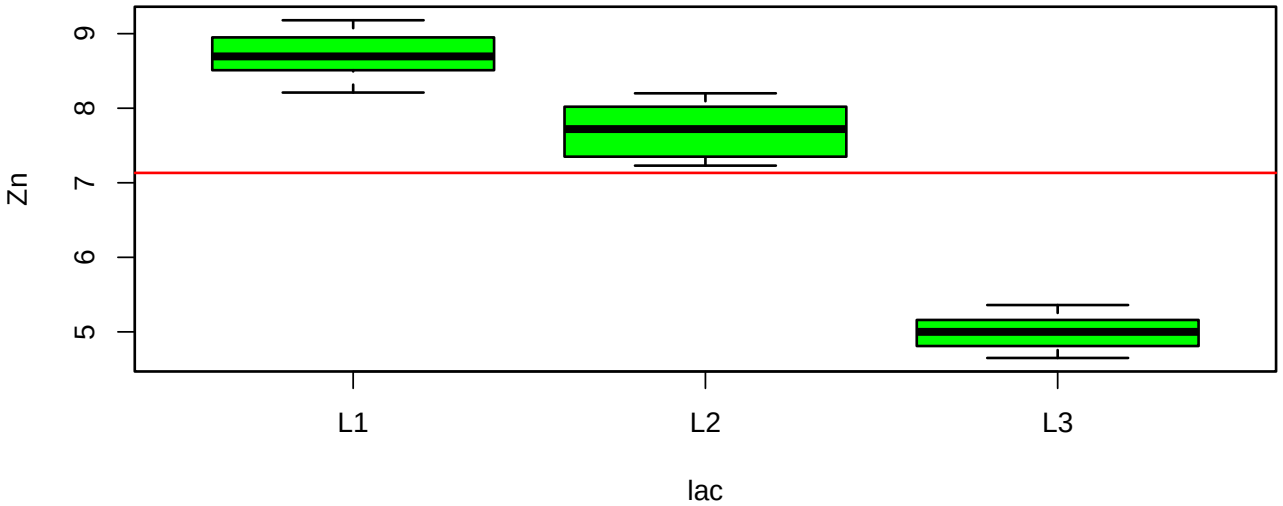
```
> library(sjPlot)
> plot_model(m)
```



Interprétation des coefficients du modèle linéaire avec la contrainte de type somme (peu utilisée)

```
> options(contrasts = c("contr.sum", "contr.sum"))
> m <- lm(Zn ~ lac, data = dZn)
> coef(m)
```

```
(Intercept)        lac1        lac2
      7.132       1.592       0.548
```



Deux variables explicatives qualitatives - ANOVA 2 cas du modèle croisé fixe avec répétitions

Exemple de l'étude de l'action de l'adrénaline et de la pitressine sur l'équilibre hydrique de têtards de crapauds.

- **1er facteur fixe étudié (noté A)** : adrénaline (2 modalités)
- **2ème facteur fixe étudié (noté B)** : pitressine (2 modalités)
- **variable observée (notée Y)** : variation horaire du poids de l'animal après le traitement

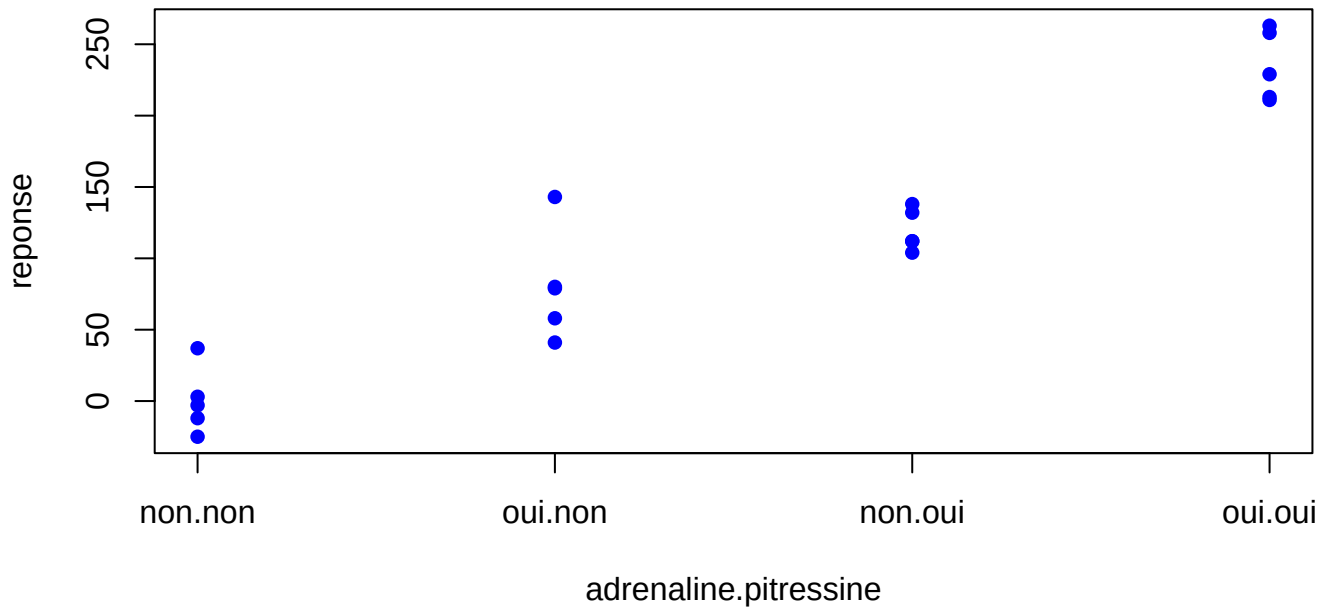
Les données : les valeurs de Y sur 4 échantillons de taille 5 correspondant au croisement des modalités des 2 facteurs.

Codage des données

```
> d <- read.table("DATA/ADREPITRE.txt", header = TRUE,
                  stringsAsFactors = TRUE)
> str(d)

'data.frame':        20 obs. of  3 variables:
 $ adrenaline: Factor w/ 2 levels "non","oui": 1 1 1 1 1 2 2 2 2 2 ...
 $ pitressine: Factor w/ 2 levels "non","oui": 1 1 1 1 1 1 1 1 1 1 ...
 $ reponse   : int  -25 -3 -12 37 3 143 41 58 80 79 ...

> head(d)

  adrenaline pitressine reponse
1        non        non     -25
2        non        non      -3
3        non        non     -12
4        non        non      37
5        non        non       3
6        oui        non     143
```

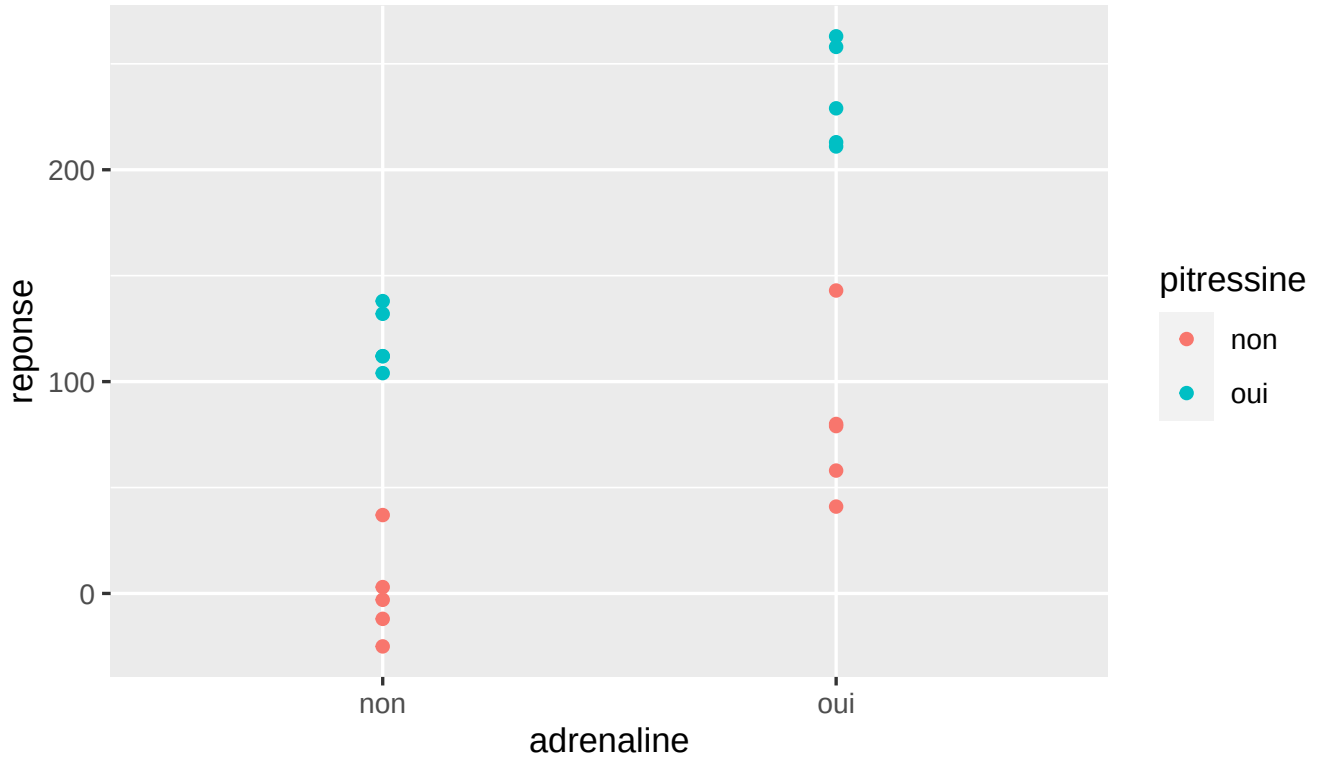
Une visualisation simple de l'ensemble des points observés

```
> par(mar = c(5, 4, 1, 1))
> stripchart(reponse ~ adrenaline:pitressine, vertical = TRUE,
    data = d, pch = 16, col = "blue", xlab = "adrenaline.pitressine")
```



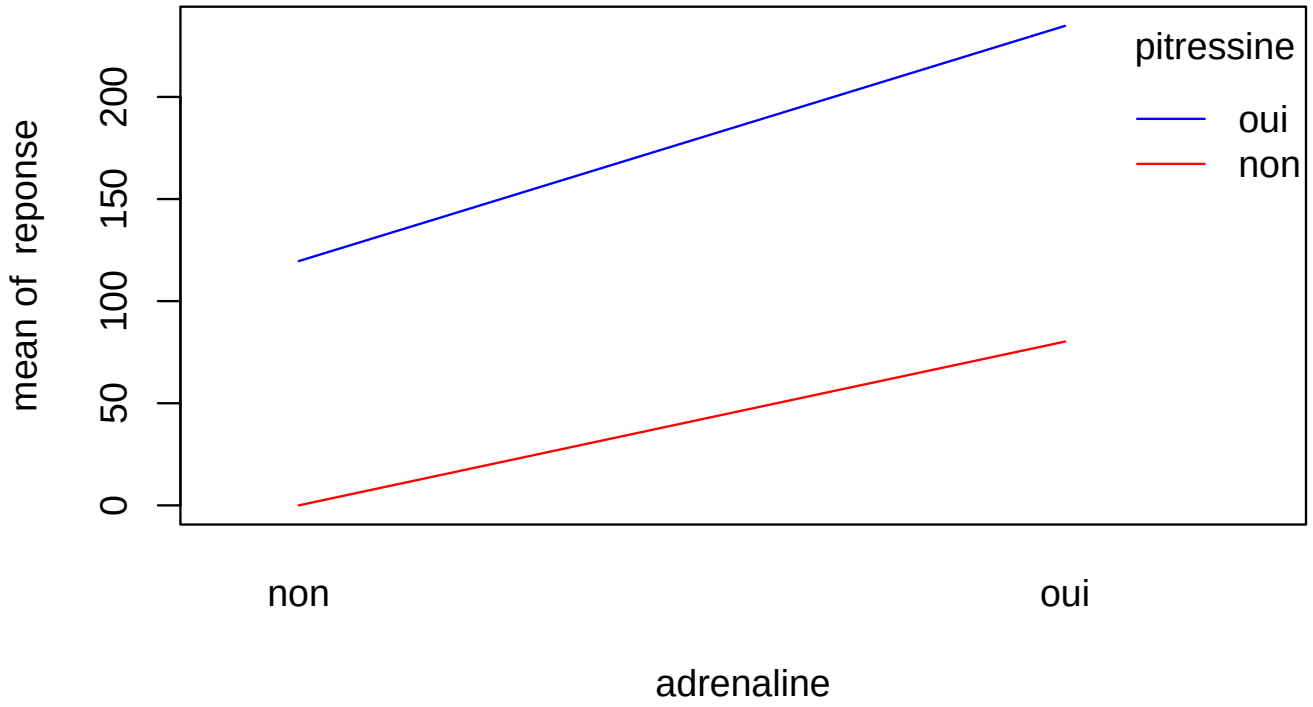
Une visualisation colorée avec ggplot2

```
> library(ggplot2)
> ggplot(data = d, aes(x = adrenaline, y = reponse,
                       colour = pitressine)) + geom_point()
```



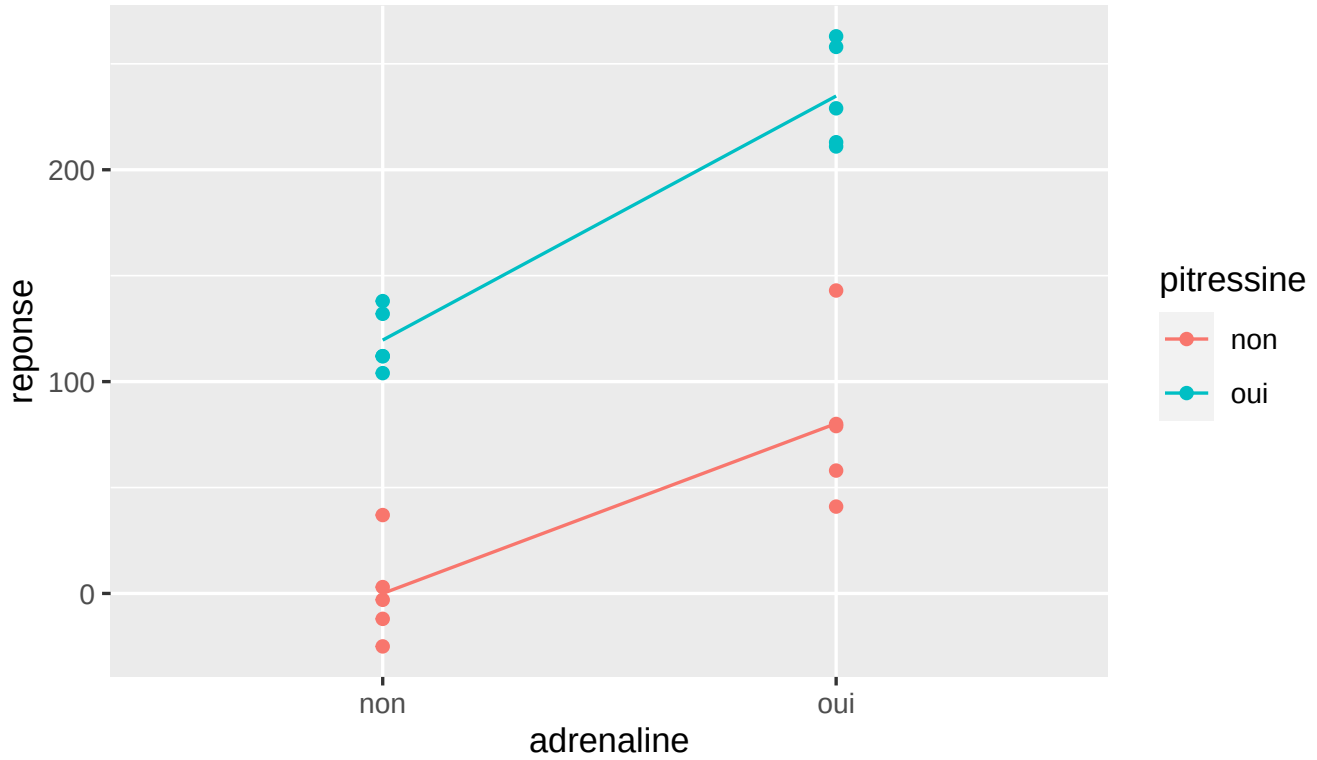
Une visualisation sous forme de graphe d'interaction

```
> par(mar = c(5, 4, 1, 1))
> within(d, interaction.plot(adrenaline, pitressine, reponse,
                             lty = c(1, 1), col = c("red","blue")))
```



Graphe d'interaction plus informatif avec ggplot2

```
> ggplot(data = d, aes(x = adrenaline, y = reponse,
                       colour = pitressine)) + geom_point() +
   stat_summary(fun = mean, geom = "line", aes(group = pitressine))
```

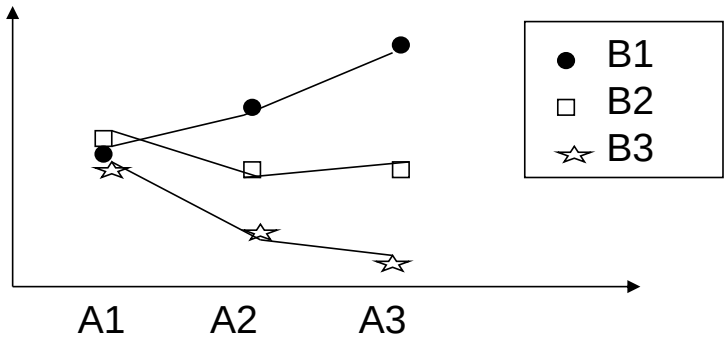


Interaction entre deux facteurs

- A et B peuvent avoir chacun un effet sur la variable observée sans qu'il y ait interaction entre les 2 facteurs (cf. ex. précédent).
- **On parle d'interaction entre A et B si l'effet de A n'est pas le même suivant la modalité de B (et réciproquement).**
- Le graphe d'interaction permet de mettre en évidence une interaction : segments non parallèles.
- Un modèle sans interaction est dit additif.
- La présence d'une interaction gêne (voire empêche) d'interpréter séparément les effets de A et B.

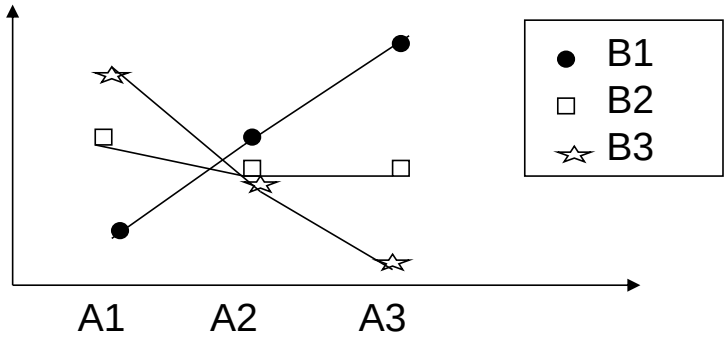
Exemples de situations quant à l'interaction

(1)



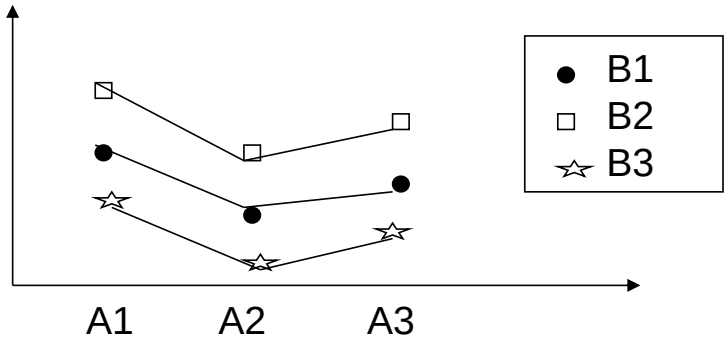
Il y a un effet de B pour la modalité A3 mais pas pour la modalité A1, donc **pas de conclusion globale possible**

(2)



L'effet de B n'est pas le même pour les modalités A1 et A3, donc **pas de conclusion globale possible**

(3)



Les segments du graphe d'interaction sont parallèles car il n'y a pas d'interaction (les effets de A et B sont dits additifs) et l'on peut tester les effets séparés de A et B

Modèles théoriques de l'ANOVA 2

Modèle avec interaction

$Y_{ij} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \epsilon_{ij}$
avec $\epsilon_{ij} \sim N(0, \sigma)$

Modèle sans interaction

$Y_{ij} = \mu + \alpha_i + \beta_j + \epsilon_{ij}$
avec $\epsilon_{ij} \sim N(0, \sigma)$

Choix du modèle basé sur la visualisation graphique des résultats et la connaissance biologique
(ATTENTION, ce n'est pas parce que l'interaction est non significative qu'on peut l'enlever).

Hypothèses que l'on peut tester en ANOVA 2

- H_0 :Absence d'interaction entre les deux facteurs
- H_0' :Absence d'action du facteur A
- H_0'' :Absence d'action du facteur B

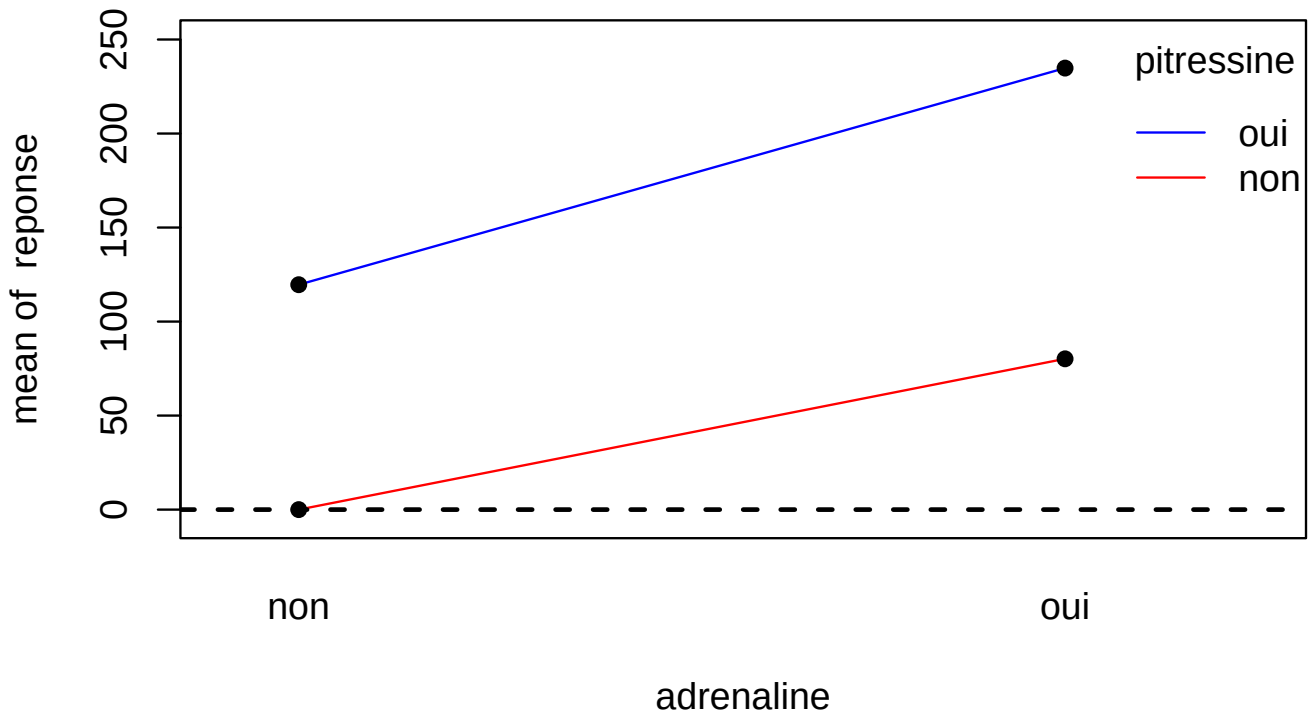
Il sera très difficile (voire impossible) de tester H_0' et H_0'' en cas d'interaction importante.

Ajustement du modèle avec interaction et interprétation des coefficients

```
> m <- lm(reponse ~ adrenaline * pitressine, data = d)
> coef(m)
                  (Intercept)                   adrenalineoui
                          0.0                            80.2
               pitressineoui adrenalineoui:pitressineoui
                        119.6                            35.0
> tapply(d$reponse, d$adrenaline:d$pitressine, mean)
non:non non:oui oui:non oui:oui
    0.0   119.6    80.2   234.8
```

Comparaison des moyennes observées à celles prédites par le modèle avec interaction

- moyennes prédites (ronds noirs).
- constante = moyenne prédite pour le groupe de référence (trait pointillé).



Ajustement du modèle sans interaction et interprétation des coefficients

```
> m <- lm(reponse ~ adrenaline + pitressine, data = d)
> coef(m)

  (Intercept) adrenalineoui pitressineoui
        -8.75         97.70        137.10

> tapply(d$reponse, d$adrenaline:d$pitressine, mean)

non:non non:oui oui:non oui:oui
    0.0   119.6    80.2   234.8
```

Comparaison des moyennes observées à celles prédites par le modèle sans interaction

- moyennes prédites (ronds noirs).
- constante = moyenne prédite pour le groupe de référence (trait pointillé).

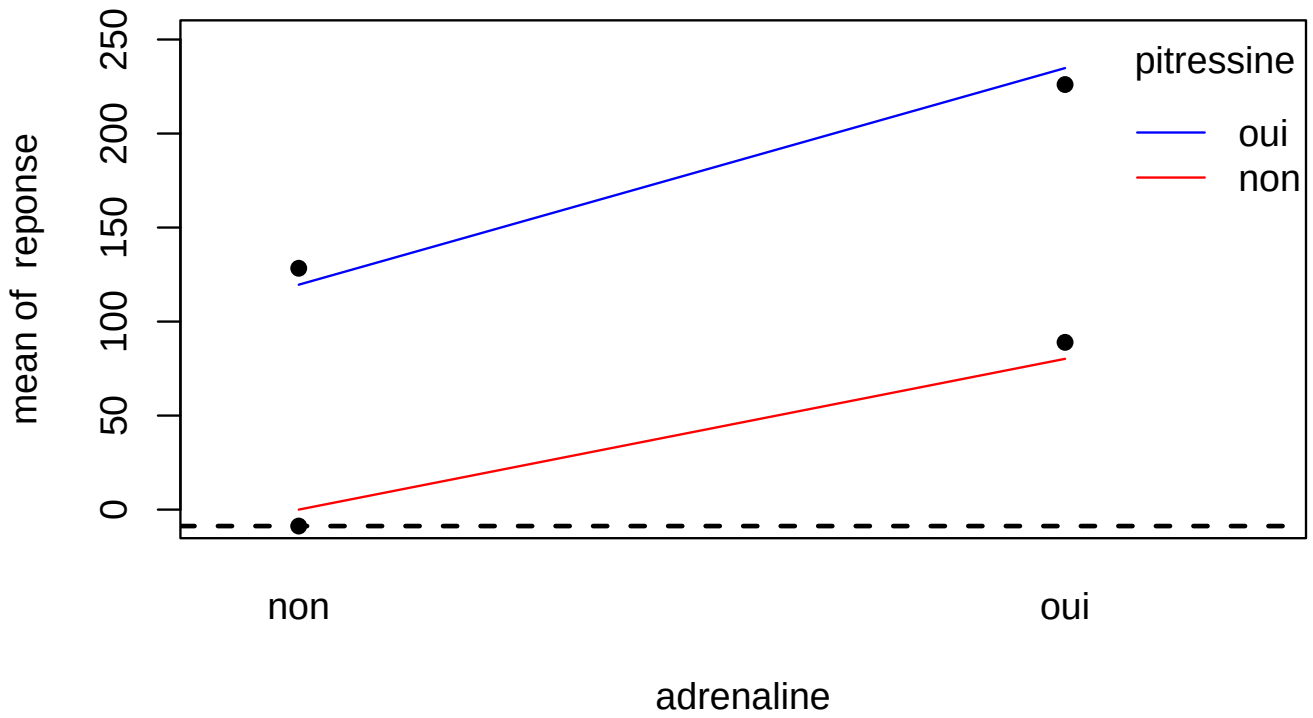


Tableau d'analyse de variance du modèle avec interaction

```
> anova(lm(reponse ~ adrenaline * pitressine, data = d))
Analysis of Variance Table

Response: reponse
                     Df Sum Sq Mean Sq F value  Pr(>F)
adrenaline            1  47726   47726   67.04 4.1e-07
pitressine            1  93982   93982  132.01 3.9e-09
adrenaline:pitressine 1   1531    1531    2.15    0.16
Residuals            16  11391     712
```

Interaction non significative et effets significatif de chacun des deux facteurs.

ATTENTION ! La réalisation de ce tableau nécessite un plan d'expérience équilibré (même nombre d'observations par groupe)

Sommes des carrés de type I dépendantes de l'ordre d'introduction des variables dans le modèle

Impact de l'ordre d'introduction des variables dans le modèle sur le tableau d'ANOVA dans un cas déséquilibré

```
> ddesequ <- d[4:20, ] # on enlève les 3 premières observations
> anova(mb1 <- lm(reponse ~ adrenaline + pitressine, data = ddesequ))
Analysis of Variance Table

Response: reponse
           Df Sum Sq Mean Sq F value  Pr(>F)
adrenaline  1  18131   18131    20.2   5e-04
pitressine  1  71175   71175    79.3 3.8e-07
Residuals  14  12563     897
> anova(mb2 <- lm(reponse ~ pitressine + adrenaline, data = ddesequ))
Analysis of Variance Table

Response: reponse
           Df Sum Sq Mean Sq F value Pr(>F)
pitressine  1  53701   53701    59.8  2e-06
adrenaline  1  35605   35605    39.7  2e-05
Residuals  14  12563     897
```

Estimation des paramètres

On peut par contre se baser sur l'estimation des paramètres pour l'inférence même dans un cas déséquilibré.
Sommes des carrés de type II indépendantes de l'ordre d'introduction des variables dans le modèle

```
> summary(mb1)$coefficients
              Estimate Std. Error t value Pr(>|t|)
(Intercept)       -5.0       15.6   -0.32 7.54e-01
adrenalineoui     95.2       15.1    6.30 1.96e-05
pitressineoui    134.6       15.1    8.91 3.85e-07
> summary(mb2)$coefficients
              Estimate Std. Error t value Pr(>|t|)
(Intercept)       -5.0       15.6   -0.32 7.54e-01
pitressineoui    134.6       15.1    8.91 3.85e-07
adrenalineoui     95.2       15.1    6.30 1.96e-05
```

Effets significatifs de l'adrénaline et de la pitressine.

Intervalles de confiance sur les paramètres

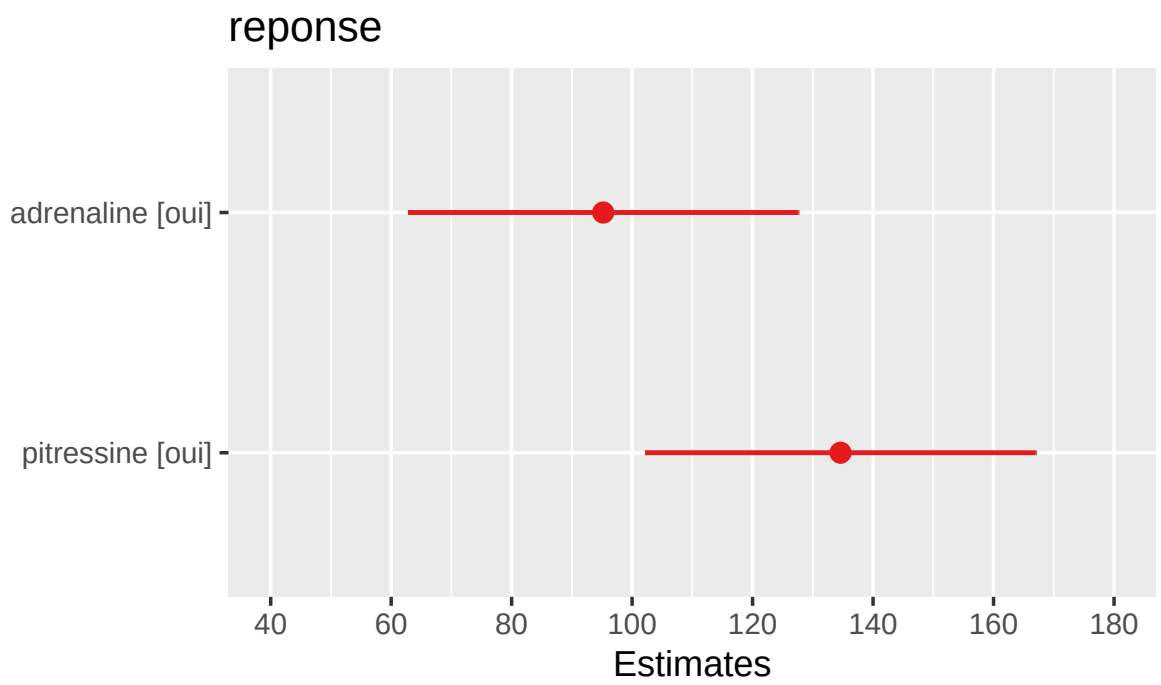
On peut aussi regarder les intervalles de confiance des paramètres pour l'inférence même dans un cas déséquilibré.

```
> confint(mb1)
                2.5 % 97.5 %
(Intercept)    -38.6   28.6
adrenalineoui   62.8  127.6
pitressineoui  102.2  167.0
> confint(mb2)
                2.5 % 97.5 %
(Intercept)    -38.6   28.6
pitressineoui  102.2  167.0
adrenalineoui   62.8  127.6
```

Même conclusion : effets significatifs de l'adrénaline et de la pitressine.

Représentation des effets estimés avec leur intervalle de confiance à l'aide du package `sjPlot`

```
> plot_model(mb1)
```



Test des modèles emboîtés

On peut aussi toujours utiliser le test des modèles emboîtés pour répondre à une question précise même dans un cas déséquilibré.

```
> madreseule <- lm(reponse ~ adrenaline, data = ddesequ)
> anova(mb1, madreseule)
Analysis of Variance Table

Model 1: reponse ~ adrenaline + pitressine
Model 2: reponse ~ adrenaline
  Res.Df   RSS Df Sum of Sq    F  Pr(>F)
1     14 12563
2     15 83737 -1    -71175 79.3 3.8e-07
> anova(mb2, madreseule)
Analysis of Variance Table

Model 1: reponse ~ pitressine + adrenaline
Model 2: reponse ~ adrenaline
  Res.Df   RSS Df Sum of Sq    F  Pr(>F)
1     14 12563
2     15 83737 -1    -71175 79.3 3.8e-07
```

Modélisation de l'effet de plusieurs facteurs qualitatifs

- **ATTENTION aux interactions entre facteurs**.
  Lorsqu'elles existent elles complexifient bien souvent l'interprétation du modèle.
  Réflexion nécessaire au préalable à la construction du modèle sur les interactions potentielles.
- Interprétation des coefficients (paramètres) du modèle pas toujours triviale, d'autant moins que le nombre de facteurs et/ou de modalités des facteurs augmente.
- L'utilisation de la fonction `lm()` est limitée à la modélisation de l'effet de **facteurs fixes** dans le cadre de **modèles croisés**.

A vous de jouer !

Consigne

Traitez l'exercice 4.

Mélange de variables explicatives qualitatives et quantitatives

Il est possible (voire courant) d'écrire un modèle linéaire avec certaines variables explicatives quantitatives (régresseurs ou covariables) et d'autres qualitatives (facteurs).
Une **interaction** entre un facteur (variable explicative qualitative) et un régresseur (variable explicative quantitative) est alors interprétée comme un **effet du facteur sur le coefficient de régression (pente)** associé au régresseur.

Exemple d'analyse de covariance - ANCOVA

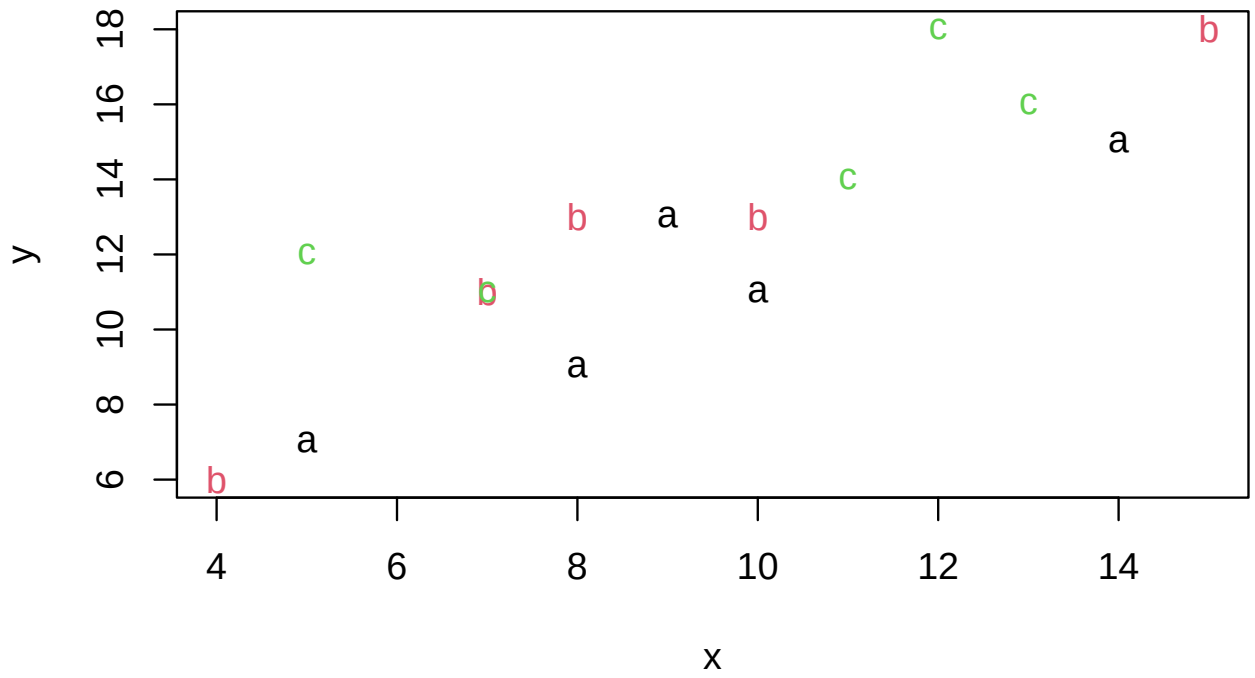
Exemple de la comparaison de l'action de trois traitements a, b, c sur un dosage sanguin, Y, en tenant compte de la valeur initiale du dosage, X, dont on se doute qu'elle impacte la valeur finale Y.
On dispose de 15 sujets randomisés en 3 groupes de 5.
Mélange d'ANOVA 1 (étude du facteur A à 3 modalités) et de régression simple (relation linéaire entre X et Y) :
effet du facteur A sur les droites de régression de Y en fonction de X.

Codage des données

```
> d <- read.table("DATA/dosageavap.txt", header = TRUE,
                  stringsAsFactors = TRUE)
> str(d)

'data.frame':        15 obs. of  3 variables:
 $ x          : int  5 8 9 10 14 4 7 8 10 15 ...
 $ y          : int  7 9 13 11 15 6 11 13 13 18 ...
 $ traitement: Factor w/ 3 levels "a","b","c": 1 1 1 1 1 2 2 2 2 2 ...

> head(d)

   x  y traitement
1  5  7          a
2  8  9          a
3  9 13          a
4 10 11          a
5 14 15          a
6  4  6          b
```

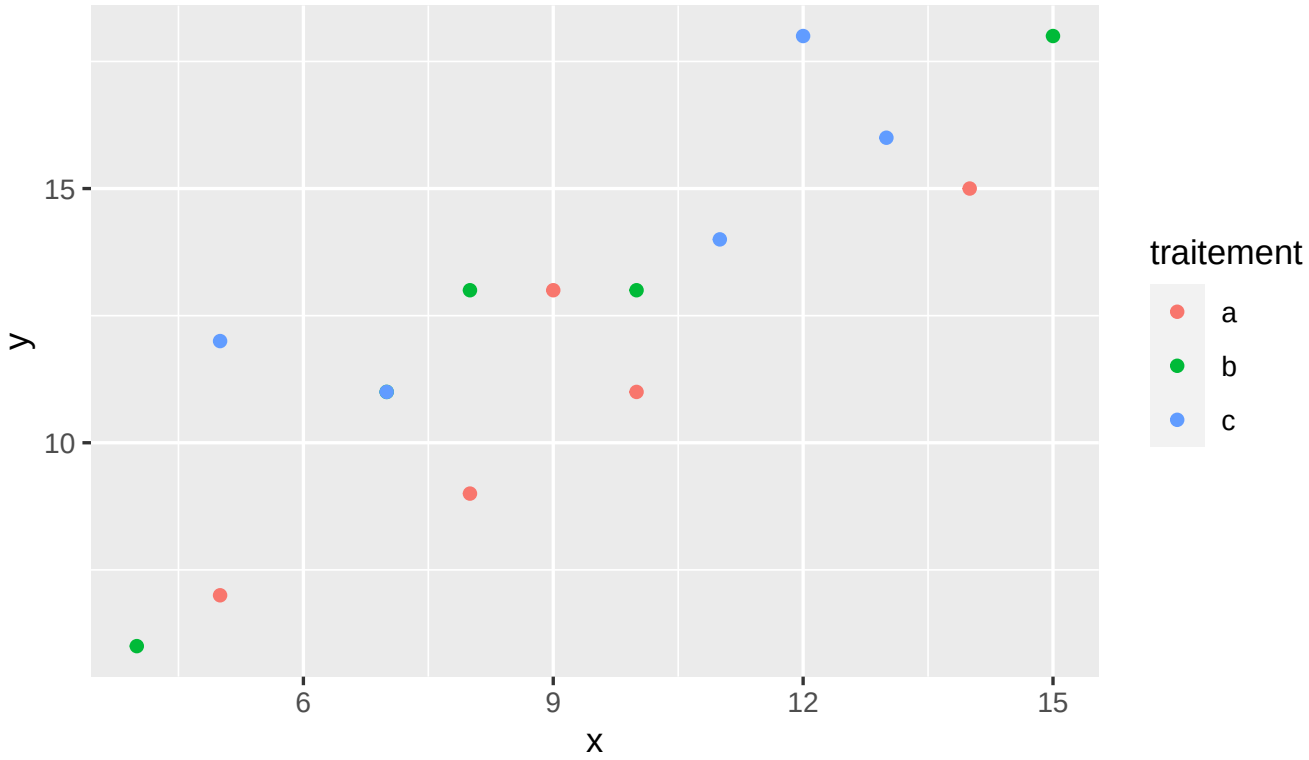
Visualisation des données

```
> plot(y ~ x, data = d, type = "n")
> text(d$x, d$y, as.character(d$traitement),
      col = as.numeric(d$traitement))
```

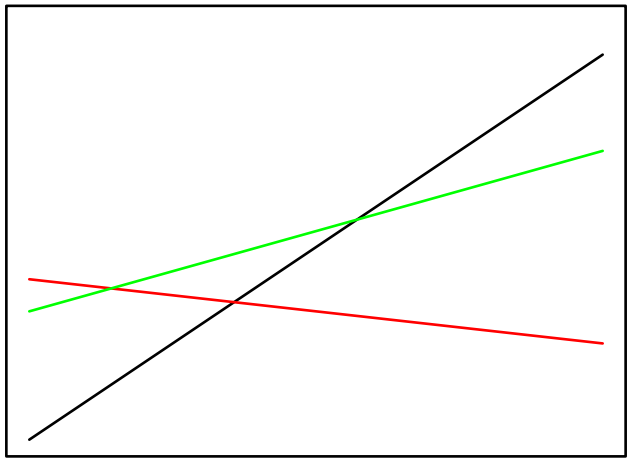


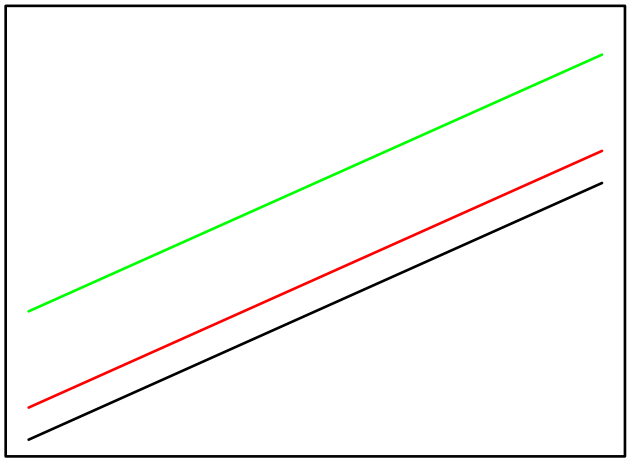
Visualisation des données avec ggplot2

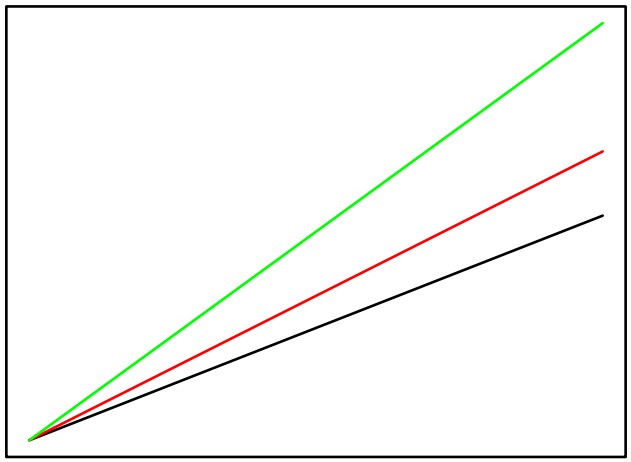
```
> ggplot(data = d, aes(x = x, y = y, colour = traitement)) +
        geom_point()
```

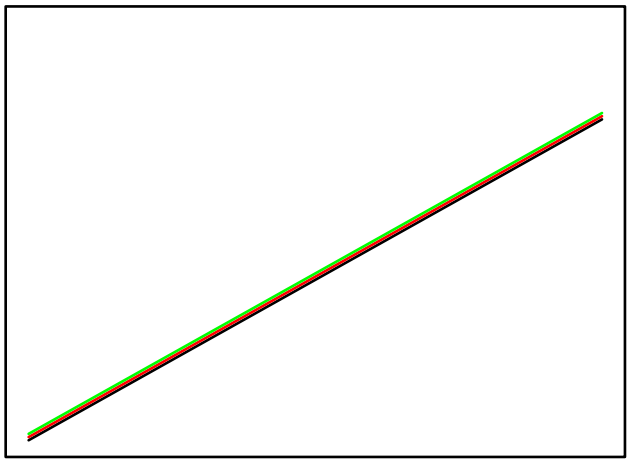


Quatre situations envisageables









Modèles théoriques de l'ANCOVA

Modèle avec interaction

$Y_{ij} = \mu + \alpha_i + \beta_i \times X_{ij} + \epsilon_{ij}$
avec $\epsilon_{ij} \sim N(0, \sigma)$

Modèle sans interaction

$Y_{ij} = \mu + \alpha_i + \beta \times X_{ij} + \epsilon_{ij}$
avec $\epsilon_{ij} \sim N(0, \sigma)$

Choix du modèle basé sur la visualisation graphique des résultats et la connaissance biologique.

Ajustement du modèle avec interaction et interprétation des coefficients

```
> mavecint <- lm(y ~ x * traitement, data = d)
> coef(mavecint)
  (Intercept)             x   traitementb   traitementc
        2.832         0.888         0.384         4.575
x:traitementb x:traitementc
        0.133        -0.180
```

Résumé de l'ajustement du modèle avec interaction

```
> summary(mavecint)

Call:
lm(formula = y ~ x * traitement, data = d)

Residuals:
   Min     1Q Median     3Q    Max
-1.360 -0.822 -0.425  0.846  2.178

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)      2.832      2.232    1.27    0.236
x                0.888      0.231    3.84    0.004
traitementb      0.384      2.845    0.13    0.896
traitementc      4.575      3.148    1.45    0.180
x:traitementb    0.133      0.296    0.45    0.664
x:traitementc   -0.180      0.319   -0.56    0.586

Residual standard error: 1.51 on 9 degrees of freedom
Multiple R-squared:  0.881,        Adjusted R-squared:  0.816
F-statistic: 13.4 on 5 and 9 DF,  p-value: 0.000604
```

Intervalles de confiance des coefficients du modèle avec interaction

```
> confint(mavecint)
                 2.5 % 97.5 %
(Intercept)   -2.218  7.882
x              0.365  1.411
traitementb   -6.052  6.820
traitementc   -2.546 11.696
x:traitementb -0.537  0.803
x:traitementc -0.903  0.542
```

De nombreux coefficients non significatifs.

Modèle sans doute surparamétré.

Ajustement du modèle sans interaction et interprétation des coefficients

Si biologiquement on ne s'attend pas à une interaction on peut construire un modèle sans interaction (en s'assurant bien sûr que les données ne mettent pas en évidence une telle interaction).

```
> msansint <- lm(y ~ x + traitement, data = d)
> coef(msansint)
```

```
(Intercept)           x traitementb traitementc
       2.81        0.89        1.56        2.84
```

Résumé de l'ajustement du modèle sans interaction

```
> summary(msansint)

Call:
lm(formula = y ~ x + traitement, data = d)

Residuals:
   Min     1Q Median     3Q    Max
-1.927 -0.908 -0.268  0.957  2.178

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)    2.809      1.252    2.24    0.046
x              0.890      0.116    7.66  9.9e-06
traitementb    1.556      0.922    1.69    0.120
traitementc    2.844      0.922    3.08    0.010

Residual standard error: 1.46 on 11 degrees of freedom
Multiple R-squared:  0.866,        Adjusted R-squared:  0.829
F-statistic: 23.7 on 3 and 11 DF,  p-value: 4.23e-05
```

Intervalles de confiance des coefficients du modèle sans interaction

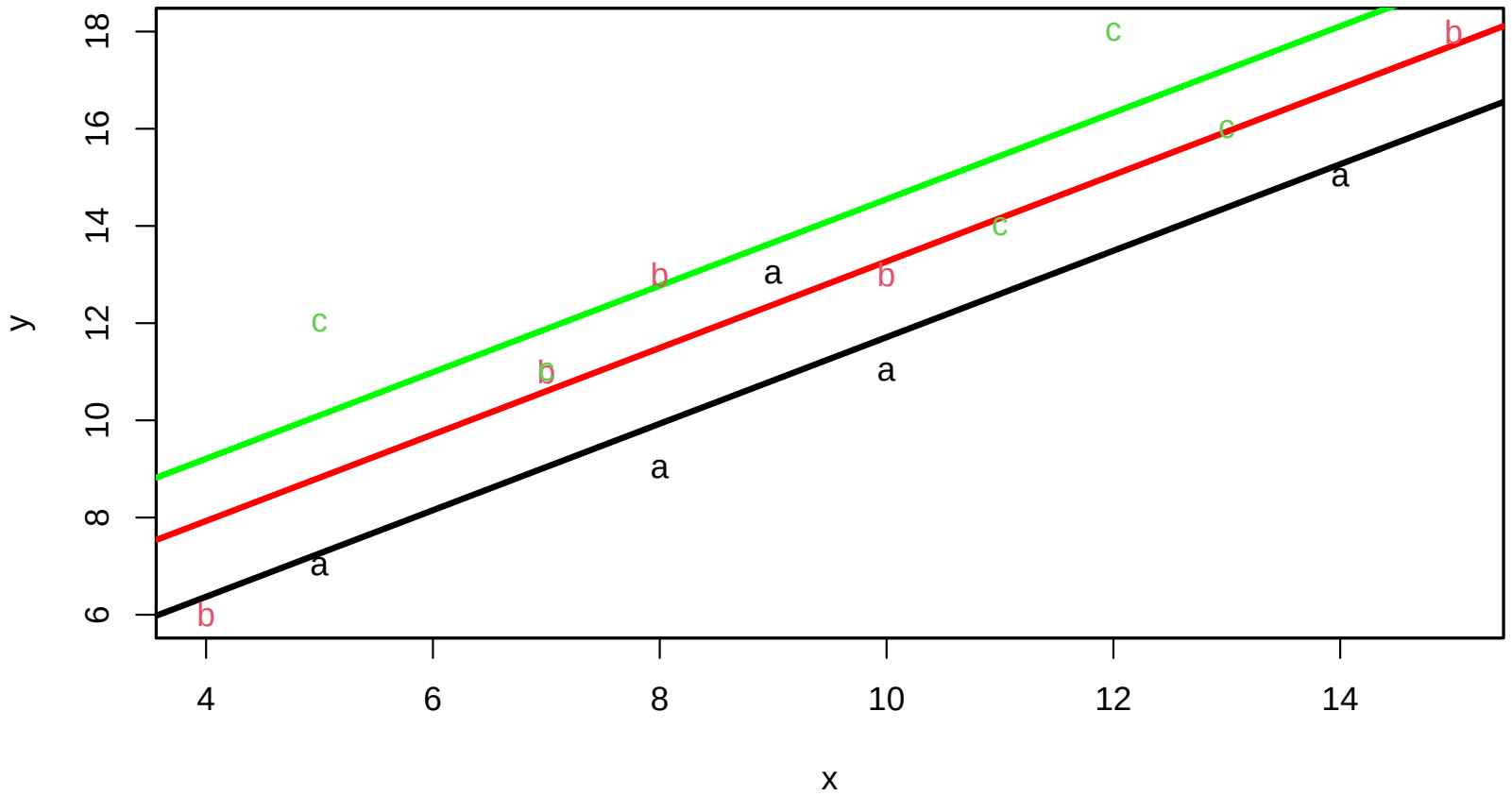
```
> confint(msansint)
                2.5 % 97.5 %
(Intercept)  0.0532   5.57
x            0.6344   1.15
traitementb -0.4730   3.59
traitementc  0.8148   4.87
```

Comparaison des deux modèles par le test F des modèles emboîtés

```
> anova(mavecint, msansint)
Analysis of Variance Table

Model 1: y ~ x * traitement
Model 2: y ~ x + traitement
  Res.Df  RSS Df Sum of Sq    F Pr(>F)
1      9 20.6
2     11 23.3 -2     -2.72 0.59   0.57
```

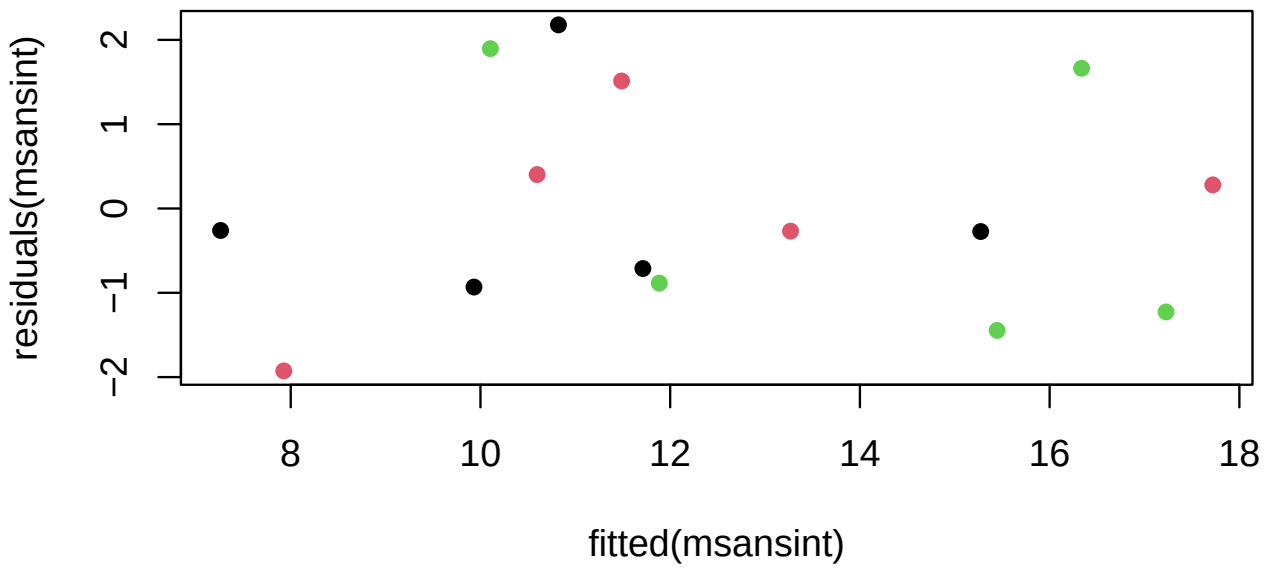
Le modèle le plus simple sans interaction semble suffisant pour bien décrire les données, donc pourrait être choisi à condition bien entendu qu'il soit en accord avec la connaissance biologique.

Visualisation de l'ajustement du modèle sans interaction



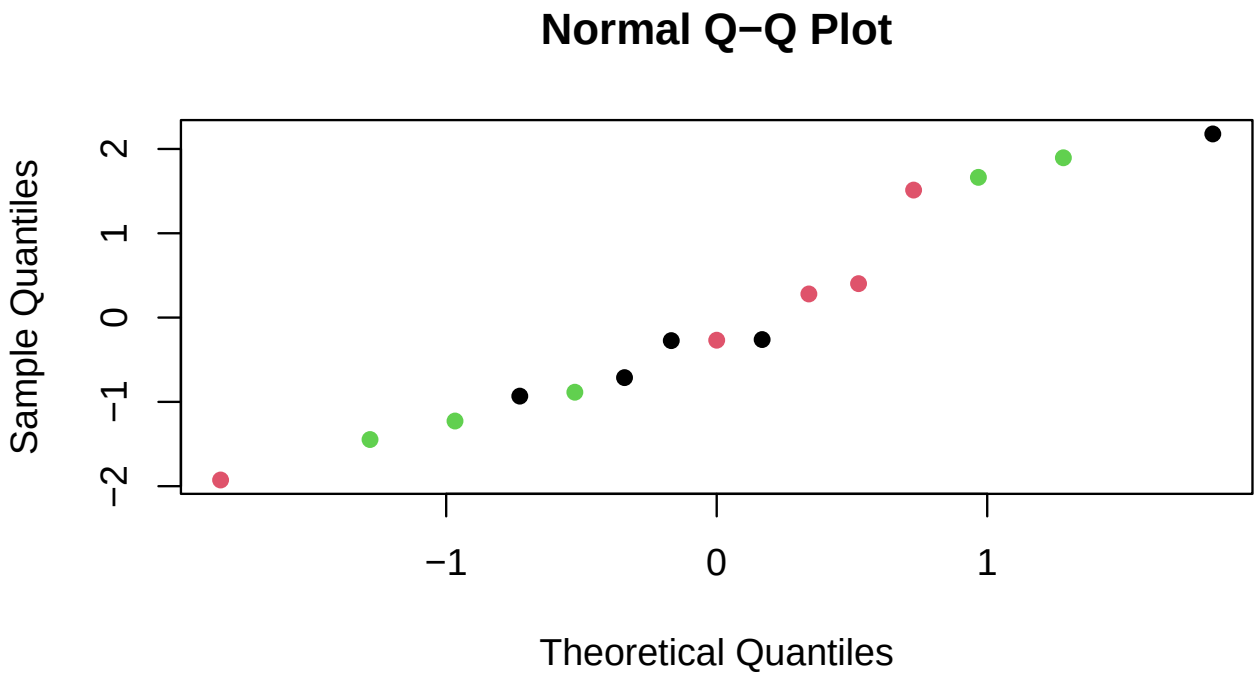
examen des résidus - graphe des résidus

```
> plot(residuals(msansint) ~ fitted(msansint),
      col = as.numeric(d$traitement), pch = 16)
```



examen des résidus - diagramme Quantile-Quantile des résidus

```
> qqnorm(residuals(msansint), col = as.numeric(d$traitement), pch = 16)
```



ATTENTION à l'impact de l'ordre d'introduction des variables dans le modèle sur un tableau d'ANCOVA

```
> anova(lm(y ~ x + traitement, data = d))
Analysis of Variance Table

Response: y
           Df Sum Sq Mean Sq F value  Pr(>F)
x           1  130.2   130.2   61.42 7.9e-06
traitement  2   20.2    10.1    4.77   0.032
Residuals  11   23.3     2.1
> anova(lm(y ~ traitement + x, data = d))
Analysis of Variance Table

Response: y
           Df Sum Sq Mean Sq F value  Pr(>F)
traitement  2   26.1    13.1    6.17   0.016
x           1  124.3   124.3   58.64 9.9e-06
Residuals  11   23.3     2.1
```

Utilisation du test des modèles emboîtés plus explicite

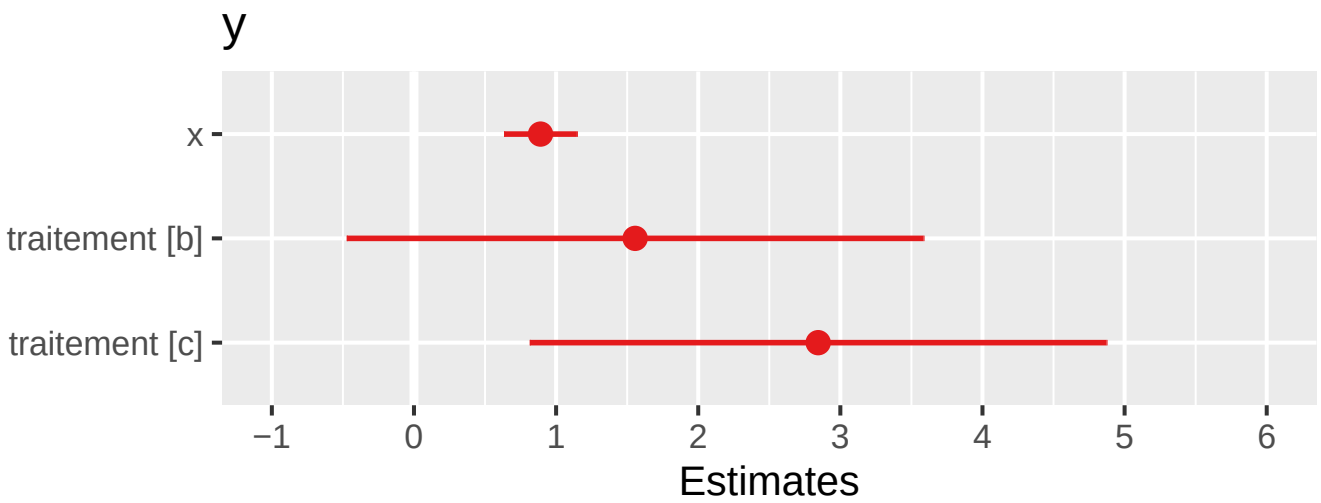
Ayant pris en compte l'effet de x, met-on en évidence un effet du traitement sur y ?

```
> mcomplet <- lm(y ~ x + traitement, data = d)
> mpartiel <- lm(y ~ x, data = d)
> anova(mcomplet, mpartiel)
Analysis of Variance Table

Model 1: y ~ x + traitement
Model 2: y ~ x
  Res.Df  RSS Df Sum of Sq    F Pr(>F)
1     11 23.3
2     13 43.6 -2     -20.2 4.77  0.032
```

Interprétation des effets estimés avec leur intervalle de confiance

ATTENTION, en se souvenant que l'effet de la covariable (x) est pour une augmentation de celle-ci d'1 unité.

```
> plot_model(mcomplet)
```



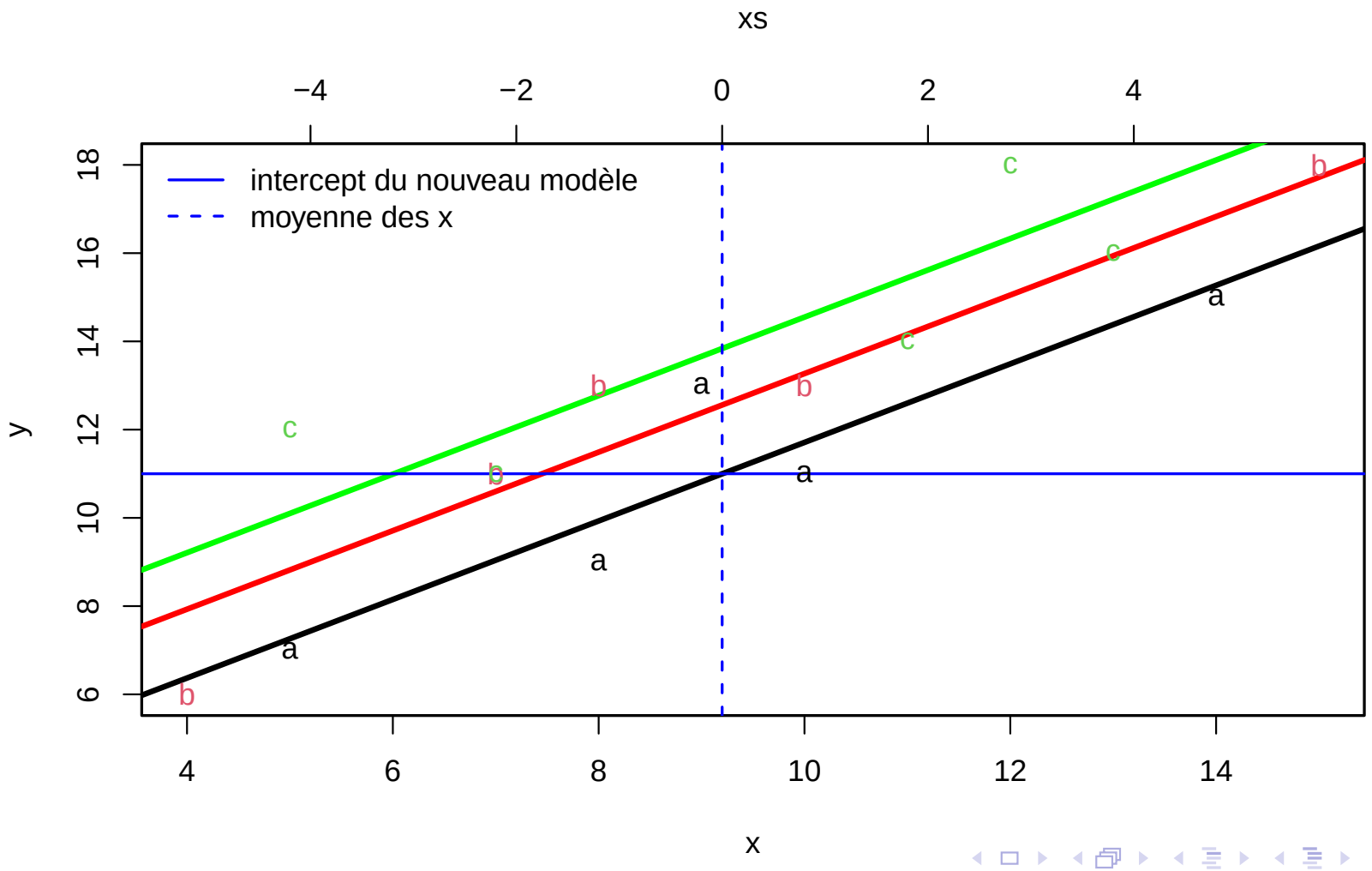
Quand cela est-il intéressant de standardiser (centrer et réduire) une covariable ? Sujet un peu controversé.

- **Lorsque la valeur 0 de la covariable n'est pas une valeur réaliste, centrer** la covariable facilite l'interprétation du terme constant (intercept). Il devient la moyenne pour le groupe de référence et une valeur moyenne de la covariable.
- **Réduire** (diviser par l'écart type) la covariable permet de **mettre les coefficients du modèle tous dans une échelle à peu près comparable**. Une pente est alors interprétée comme la différence correspondant à une **augmentation d'1 écart type de la covariable**.
- ATTENTION, dans un modèle avec interaction, selon qu'on travaille sur des covariables centrées ou non la signification des coefficients (ainsi que leur valeur) change.

Ajustement du modèle sans interaction en centrant et réduisant x le dosage avant traitement

```
> #### Modèle initial sans transformer x ####
> mcomplet <- lm(y ~ x + traitement, data = d)
> coef(mcomplet)

(Intercept)           x traitementb traitementc
       2.81        0.89        1.56        2.84

> #### Nouveau modèle en centrant et réduisant x ####
> d$xs <- scale(d$x)
> mcomplet_s <- lm(y ~ xs + traitement, data = d)
> coef(mcomplet_s)

(Intercept)          xs traitementb traitementc
      11.00        2.99        1.56        2.84
```

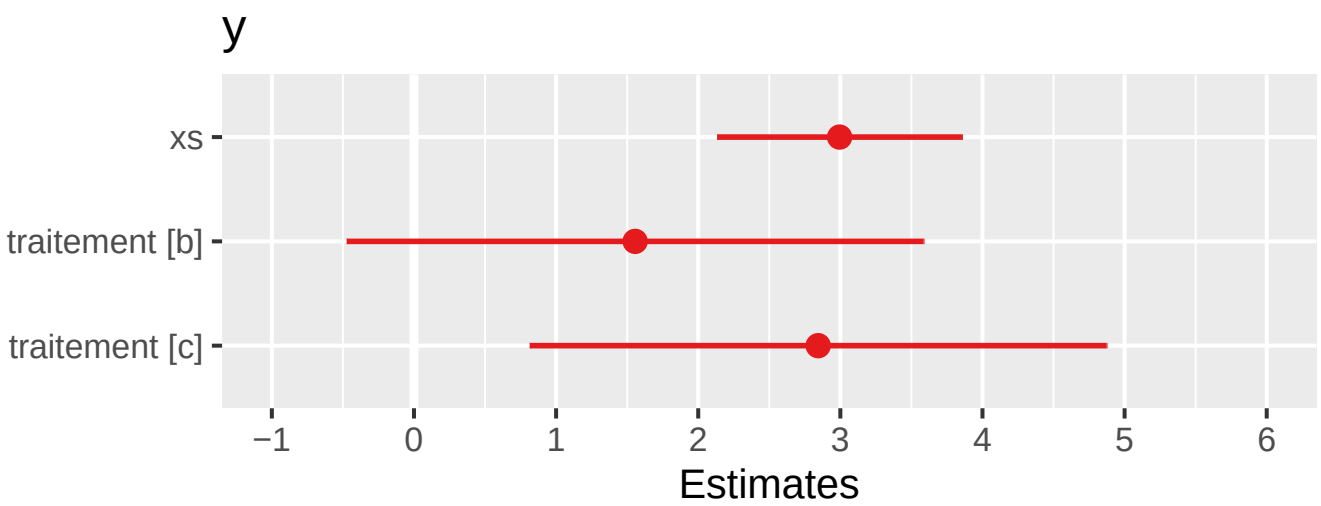
Revenons au graphe d'ajustement pour mieux comprendre et reportons-y la nouvelle intercept



Interprétation des effets estimés avec leur intervalle de confiance après standardisation de x

x ayant été centré et réduit, son effet est ici pour une augmentation d'1 écart type de x.

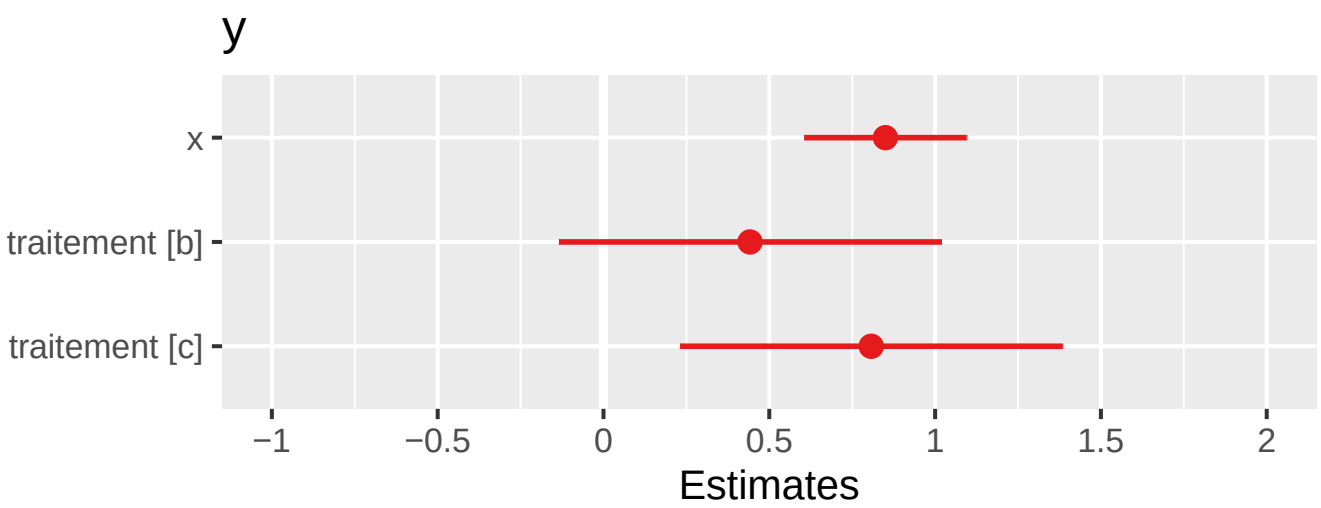
```
> plot_model(mcomplet_s)
```



Alternative : utiliser l'argument type ("std" ou "std2")

On peut mettre l'argument `type` à `"std"` ou `"std2"` pour diviser chaque coefficient de régression respectivement par SD ou 2SD de la var. explicative corresp. (cf. Gelman 2008 pour justification).

```
> plot_model(mcomplet, type = "std")
```



Gelman A (2008) "Scaling regression inputs by dividing by two standard deviations."
Statistics in Medicine 27 : 2865-2873.

Conclusion

Sur la base des exemples étudiés, on peut en théorie modéliser par un modèle linéaire l'effet sur une variable quantitative observée d'une ou plusieurs variables explicatives quantitatives et/ou qualitatives, en n'oubliant pas :

- de réfléchir à l'hypothèse de **linéarité du modèle** pour les variables explicatives quantitatives,
- de réfléchir à l'intérêt ou non de standardiser certaines variables explicatives quantitatives,
- de réflechir aux **interactions** à inclure dans le modèle,
- de ne pas inclure des variables explicatives trop fortement corrélées,
- de vérifier la **répartition aléatoire gaussienne des résidus**,
- d'utiliser des **modèles parcimonieux**.

Pour réfléchir

- Quelle autre méthode simple aurait-on pu utiliser pour prendre en compte la covariable X (dosage avant traitement) ?
- A quel modèle aurait correspondu cette alternative ?

Vers d'autres modèles de régression

- Modèle non linéaire
- Modèle linéaire généralisé
- Modèle linéaire mixte

Le modèle non linéaire

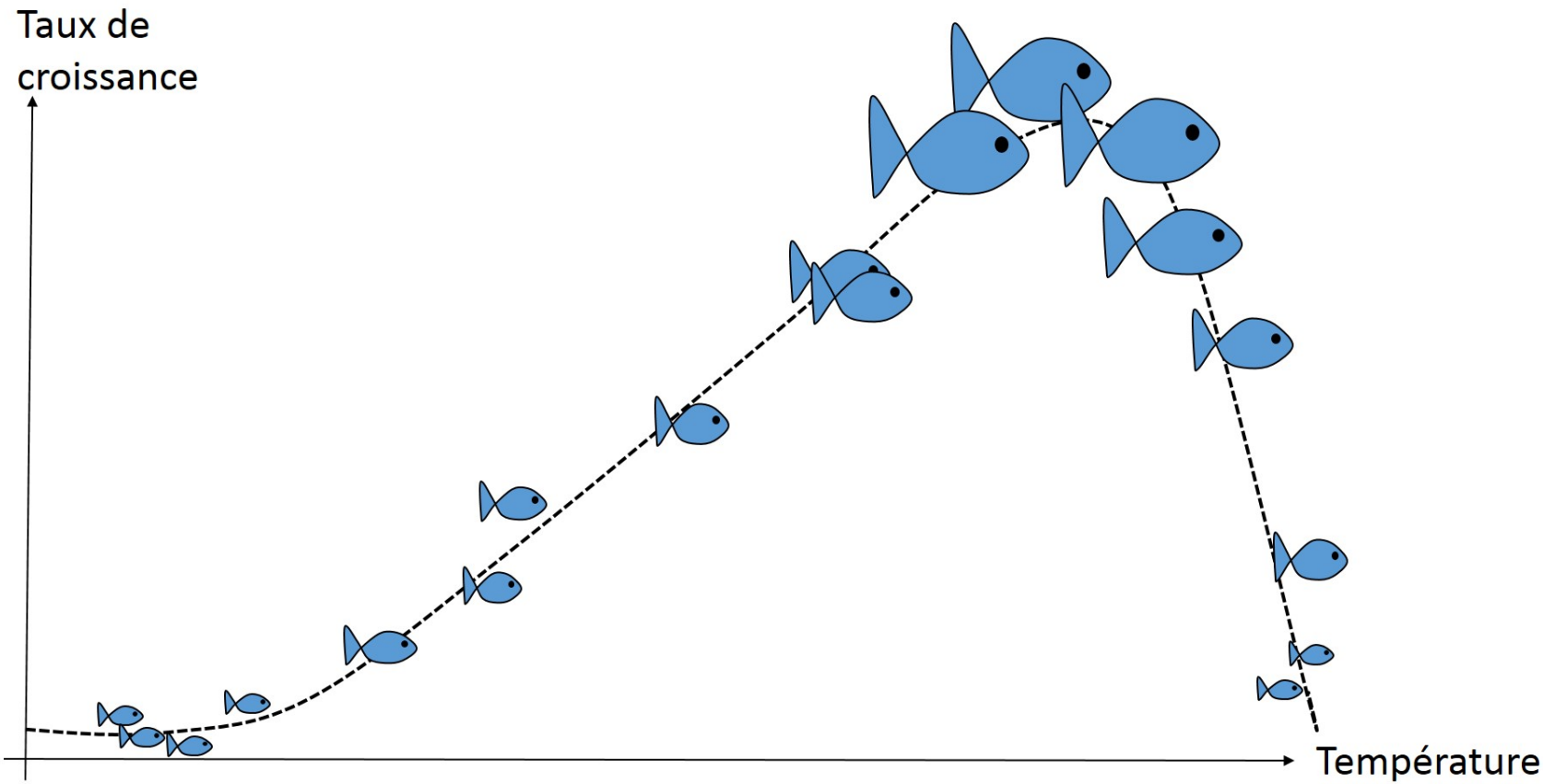
Un modèle est dit non linéaire si la variable à expliquer ne peut plus être exprimé comme une fonction linéaire des paramètres du modèle.

$$Y_i = f(X_i, \theta) + \epsilon_i$$
$$\text{avec } \epsilon_i \sim N(0, \sigma)$$

Ex. de modèle non linéaire : $Y_i = \alpha e^{\mu X_i} + \epsilon_i$ *avec* $\epsilon_i \sim N(0, \sigma)$
Ex. de modèle linéaire : $Y_i = \alpha + \beta X_i + \gamma X_i^2 + \epsilon_i$ *avec* $\epsilon_i \sim N(0, \sigma)$
Partie déterministe : fonction non linéaire des paramètres.
Partie stochastique : modèle gaussien.

fonction `nls` dans **R**

Le modèle non linéaire - illustration schématique

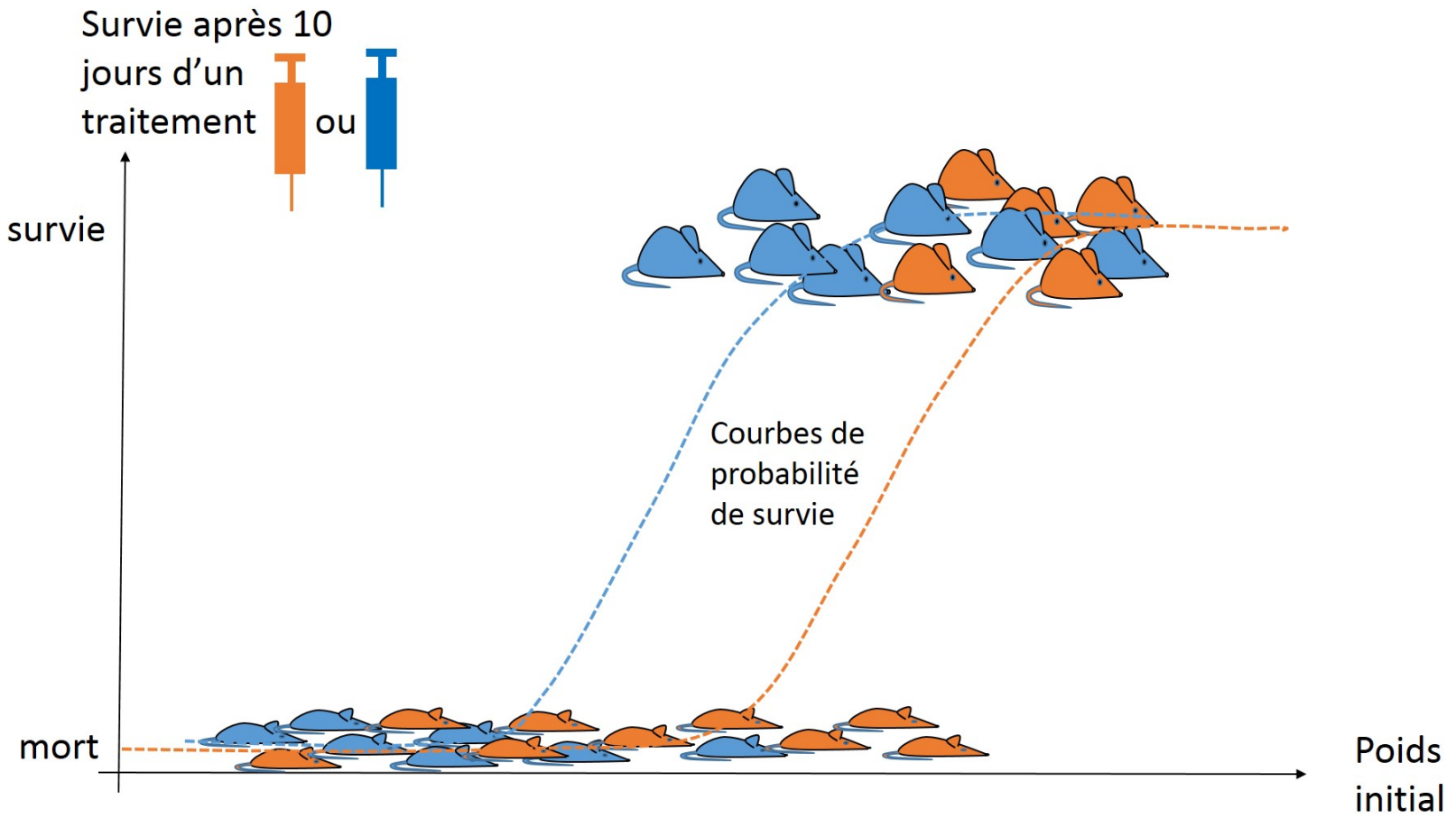


Le modèle linéaire généralisé

Un modèle linéaire généralisé permet de modéliser
l'**effet de plusieurs variables explicatives quantitatives et/ou qualitatives sur**
une variable à expliquer qualitative binaire
(ex. : malade / non malade)
ou une variable quantitative discrète
(ex. : nombre d'animaux par portée).
Partie déterministe : une transformation de la variable à expliquer (fonction de lien) est décrite par une fonction linéaire des variables explicatives .
Partie stochastique : le modèle n'est plus gaussien.

fonction `glm` dans **R**

Le modèle linéaire généralisé - illustration schématique



Le modèle linéaire mixte

Un modèle linéaire gaussien ne permet de prendre en compte que des facteurs (ou variables qualitatives) fixes, c'est-à-dire dont toutes les modalités d'intérêt sont observées. Lorsque seul un échantillon aléatoire des modalités d'un facteur sont observées, le **facteur est dit aléatoire** et l'on utilise alors un **modèle mixte** pour modéliser son effet sur la variable à expliquer.

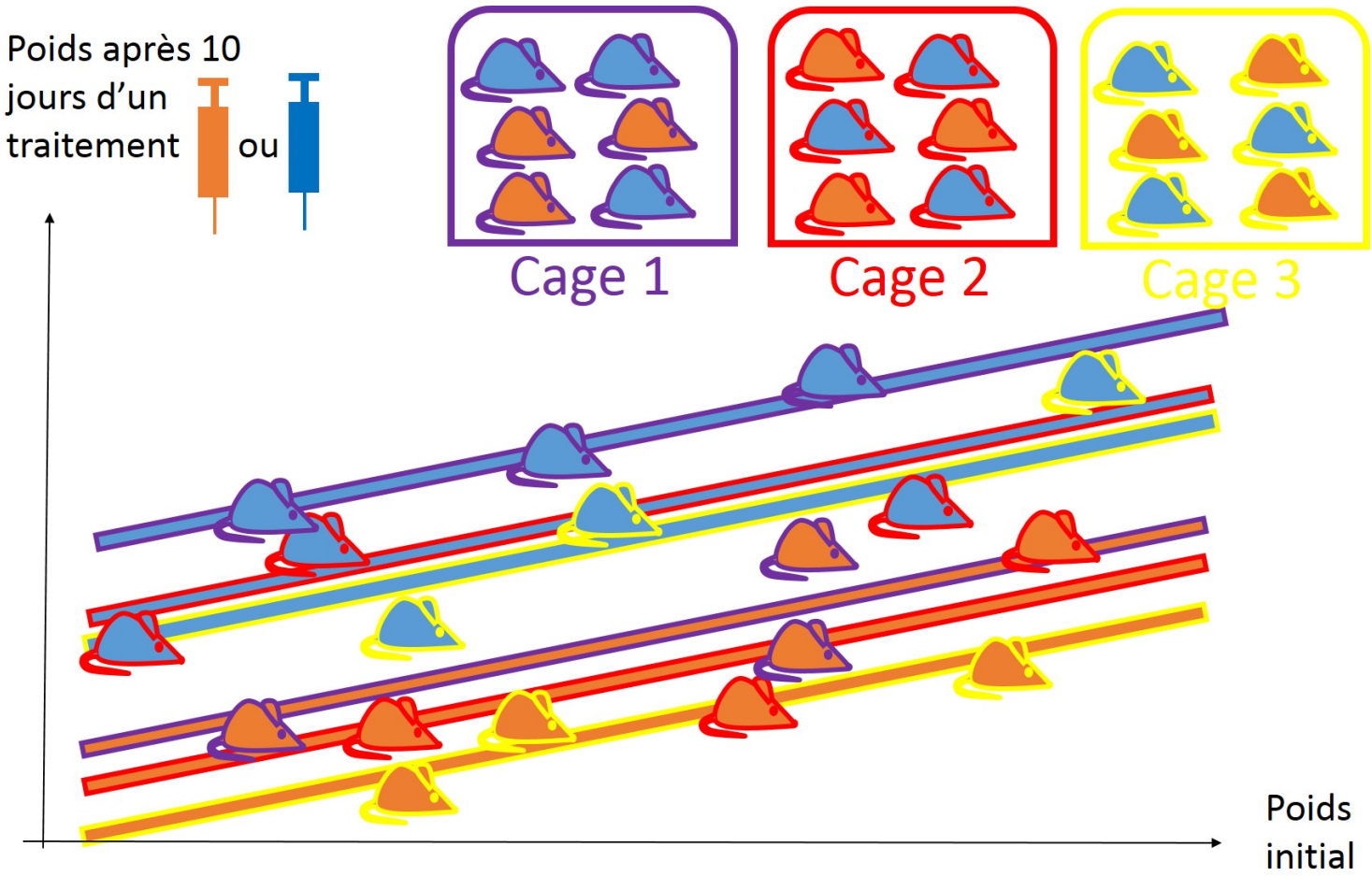
Ex. : prise en compte d'un facteur "cage" ou "élevage"

Partie déterministe : linéaire .

Partie stochastique : modèle gaussien sur les ϵ_i et modèle gaussien sur les effets des facteurs aléatoires.

fonction `lmer` du package `lme4` dans **R**

Le modèle linéaire mixte - illustration schématique



A vous de jouer !

Consigne

Traitez l'exercice 4, puis pour les plus rapides l'exercice 5.