# Quelques notes sur les statistiques

P.Gaignon

June 11, 2025

## Contents

# List of Figures



# A propos de ce document :

Ce document est destiné à une utilisation non commerciale. Le but est juste de partager les connaissances que j'ai pu acquises en statistiques aux travers de mon expérience. L'objectif n'est pas de faire un document de référence / cours, mais un partage d'expérience. J'essaye autant que possible d'ajouter les références des données et références utilisés.

Pour chaque partie, il est précisé les packages nécessaires à installer pour faire tourner le code proposé. Il est également important d'installer les dépendances des packages pour que tout fonctionne. Des commentaires de code sont également ajoutés pour expliquer certains éléments particuliers lors de la mise en pratique sous R.

Attention, il ne s'agit pas d'un document apprenant à coder en R, mais donnant les éléments nécessaires pour réaliser certaines analyses. La majorité des analyses sont réalisées à partir de données déjà présentes dans R.

# I) Quelques bases statistiques

# II) Quelques définitions en vrac (et qu'on peut retrouver plus bas)

## a Des notions mathématiques

**Moyenne :**

**Variance :**

**Covariance :** La covariance d'un couple de variable aléatoire (X,Y) est défini comme $cov(X,Y) = E(XY) - E(X)E(Y)$. Si X et Y sont indépendants, alors on a $E(XY) = E(X)E(Y)$, donc la covariance est nulle.

**Coefficient de corrélation linéaire** : Pour deux variables aléatoires X et Y, de variances non nulles, on peut définir leur coefficient de corrélation linéaire par $Cor(X,Y) = \frac{cov(X,Y)}{\sqrt{V(X)V(Y)}}$

**Moyenne harmonique** $\tilde{\mu}$ : La moyenne harmonique $\tilde{\mu}$ d''un échantillon de taille n associées aux valeurs $\{x_1, x_2, ....x_n\}$ est le nombre dont l'inverse est la moyenne arithmétique des inverses des dites valeurs :

$$\tilde{\mu} = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + ... \frac{1}{x_n}}$$

Si à chaque $x_i$ est associé un poids $w_i$ spécifique, son estimation devient alors :

$$\tilde{\mu} = \frac{\sum_{i=1}^{n} w_i}{\sum_{i=1}^{n} w_i x_i}$$

**La suite arithmétique** : on a $U_{n+1}$=$U_n$+r, avec *r* sa raison arithmétique. Alors la somme des termes vaut $\sum_{k=0}^{n} U_k = \frac{(U_0+U_n)(n+1)}{2} = (n+1)U_0 + \frac{rn(n+1)}{2}$

**La suite géométrique** : on a $U_n = U_o q^n$ avec q sa raison (différente de 0). On a alors que si q est différent de 1, la somme des termes vaut $\sum_{k=0}^{n} U_k = (\frac{q^{n+1}-1}{q-1})U_0$

**La suite arithmético-géométrique** : on a $U_{n+1} = aU_n + b$, avec a et b différents de 0. On peut alors calculer également définir c, solution unique de $ac+b = c$, avec a différent de 1, tel que $U_n = (U_0 - c) * a^n + c$, pour tout n un entier positif.

**La suite récurrente linéaire d'ordre 2** : On défini alors $U_{n+2} = aU_{n+1} + bU_n$. On définit alors le polynome P caractéristique de $U_n$ : $X^2 - aX - b$. Si b est différent de 0, et $\mu$ et $\lambda$ les racines de P, Alrs il exsite $\alpha$ et $\beta$, des nombres complexes, tels que s'il sont différents, $U_n = \alpha\lambda^n + \beta\mu^n$, et $(\alpha n + \beta)\lambda^n$ sinon.

**Propriété du coefficient binomial** : Il permet de compte au nombre d'arrangements en sélectionnant k élément parmi n. On définit alors $C_n^k = \frac{n!}{k!(n-k)!}$. On peut notamment noter que $C_n^k = C_n^{n-k}$ et que $C_{n+1}^{k+1} = C_n^k + C_n^{k+1}$

## b Quelques notions de probabilités

### b.1 Elements généraux

En dehors du théorème central limite sur lequel se base une grande partie des statistiques, plusieurs élèments de probabilités trouvent sens dans les analyses statistiques. La première est la définition de l'indépendance. Si on considère deux évènements A et B indépendants, alors $P(A \cap B) = P(A) \times P(B)$. A l'inverse, on sait que A est indépendant de B si $P(A|B) = P(A)$, c'est-à-dire que la réalisation de A ne dépend pas de B.

### b.2 Lois discrètes :

- Loi uniforme définie sur {1,... n}, alors pour k appartenant à cet ensemble, P(X=k)=$\frac{1}{n}$, E(X)=$\frac{n+1}{2}$, V(X)=$\frac{n^2-1}{12}$
- Loi de Bernouilli/Binomiale, de paramètre (n,p), où n le nombre d'essai, et p la probabiltié que cela arrive sur 1 événement, défini sur [0;n] -> [0,1], P(X=k)=$C_k^n p^k (1-p)^{n-k}$, E(X) = np, V(X)=np(1-p)
- Loi Multinomiale, de paramètre c (le nombre de modalité possible supérieur ou égal à 2 et un entier positif) et $n_1, ....n_c$ le nombre maximales obtenues pour chaque valeur tel que $n = \sum_{k=1}^{c} n_k$. On définit alors $p_k = \frac{n_k}{n}$ On a alors $P(X_1 = k_1, ..., X_c = k_c) = \frac{n!}{k_1!x...xk_c!} \times p_1^{k_1} \times ... \times p_c^{k_c}$
- Loi de Poisson, de paramètre $\lambda$, défini sur [0;+$\infty$[ -> [0,1], P(X=k)=$\frac{\lambda^k}{k!}e^{-\lambda}$ . E(X)=$\lambda$, V(X)=$\lambda$
- Loi de Parcal / Loi Binomiale Négative : Combien de tirage pour obtenir k fois le même éléments : $P(X = n) = C_{k-1}^{k-1} p^k (1-p)^{n-k}$, $E(X) = \frac{k}{p}$ et $V(X) = \frac{k(1-p)}{p^2}$
- Loi géométrique de paramètre p qui compte le nombre d'essai jusqu'à un succès, on A P(X)=$p(1-p)^{x-1}$, avec E(X)=$\frac{1}{p}$ et V(X)=$\frac{q}{p^2}$
- Loi Hypergéomtréique de paramètre (N,n,p), travaillant sur une population N dont on extrait une sous-population n, avec p la probabilité de l'événement d'intérêt. L'ensemble de définition de X dépend alors des valeurs de n et p choisies : {max(0;n-Nq),....,min(n,Np)} On a alors P(X=k)=$\frac{C_N^k C_{N-Np}^{n-x}}{C_N^n}$, avec E(X)=$np$ et V(X)=$\frac{N-n}{N-1}np(1-p)$. Il est à noté que si N $\hookrightarrow \infty$, alors la loi tend vers une loi Binomiale classique.

### b.3 Lois continues :

- Loi Normale : de moyenne m et d'écart type$\sigma$, défini sur ]-$\infty$;+$\infty$[ -> [0,1], f(x)=$\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}(\frac{t-m}{\sigma})^2}$
- Loi Uniforme, définie sur [a,b]. si $k \in [a, b]$, alors P(X$\leq$k)=$\frac{k-a}{b-a}$, avec E(X)=$\frac{b+a}{2}$ et V(x)=$\frac{(b-a)^2}{12}$
- Loi Exponentielle de paramètre $\lambda$ définie sur [0,$\infty$[. f(x)=$\lambda e^{-\lambda x}$ pour tout x positif, et E(X)=$\frac{1}{\lambda}$ et V(X)=$\frac{1}{\lambda^2}$.
- Loi Gamma, définie sur [0;+$\infty$]. $f(x) = \frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha-1} e^{\frac{-x}{\beta}}$, avec $\Gamma(\alpha) = \int_0^{+\infty} \frac{x^{\alpha-1}}{\beta^\alpha} e^{\frac{-x}{\beta}} dx$. On peut alors l'écrire $f(x) = \frac{1}{\beta^\alpha (\alpha-1)!} x^{\alpha-1} e^{\frac{-x}{\beta}}$. E(X)=$\alpha\beta$ et V(X)=$\alpha\beta^2$
- Loi du khi-deux définie sur ]0,+$\infty$[ de paramètre $\nu$. Il s'agit d'un cas particulier de la loi Gamma, avec $\alpha = \nu/2$ et $\beta = 2$, $\nu$ représente le nombre de degrés de liberté. On a alors $f(x) = \frac{1}{2^{\nu/2}\Gamma(\nu/2)} x^{\frac{\nu}{2}-1} e^{\frac{-x}{2}}$. On a alors E(X)=$\nu$ et V(X)=2$\nu$.

## c Sur des variables qualitatives

**Odds ratio (OR) :** également appelé rapport de cotes.

On peut l'oobtenir à partir d'un tableau de contingence, ici réussite à un examen selon le fait d'avoir révisé ou pas (Cf : Episode 11 de la chaine le risque $\alpha$)

| | Réussite à l'examen | Echec à l'examen |
|---|---|---|
| Révision | 450 | 50 |
| Pas de révision | 350 | 150 |

Table 1: Tableau de contingence sur la réussite à un examen

Dans ce cas, OR vaut 450/50 sur 350/150, soit 9/2.33, donc OR = 3.86.

**Risque Relatif (RR) :** Ratio des risques entre le traitement et le contrôle. Dans le cas du tableau exposé pour l'OR, c'est 450/500 et 350/500. Ce qui donne un RR de 0.9/0.7 = 1.28. Il y a une augmentation de 28% des chances de réussir ses examens en révisant.

Cependant, le RR n'est pas toujours calculable, contrairement à l'OR. Ils sont tous deux des tailles d'effet. Si les risques sont rares, OR et RR sont souvent très proches.

**Needed Number to Treat (NNT)** : Principe assez simple, combien de sujet à traiter pour changer le résultat de 1 (Nombre de personne à soigner avec le traitement pour en soigner une de plus que dans le groupe contrôle). Il se calcule de la manière suivante :

$$NNT = \frac{1}{\frac{NbreSuccès_T raitement_t}{n_T} - \frac{NbreSuccès_C}{n_C}}$$

si on prend le cas du tableau exprimé pour les OR, on obtien un NNT de 1/(0.9-0.7)=5. Il faut donc que 5 personnes révisent pour qu'une de plus ait son examen.

Cependant, parfois quelques utilisations abusives du NNT, et pas mal de problème dans la définition de son intervalle de confiance comme décrit par Hutton (2000). Hutton (2000) a proposé une autre définition du NNT à partir de $\pi_T$, proportion de succès dans le groupe Traitement :

$$NNT = \frac{1-\pi_t 1-1/OR}{\pi_T(1-\pi_T)(1-1/OR)}$$

Le problème principal est l'estimation de l'intervalle de confiance. Cet intervalle n'est en effet pas symétrique car le NNT ne suit pas une loi normale. De plus, par définition, le NNT ne peut valoir 0. Du coup, quand le NNT s'approche de zéro, des problèmes conceptuels apparaissent et rendent son utilisation très difficiles. De même, quand le NNT tend vers des grandes valeurs, son interprétation reste compliquée.

## d Quelques courbes de références

### d.1 Courbes Concaves

#### d.1.1 Courbes exponentielles Une très connu est la courbe exponentielle :

$Y = ae^{kX}$

où on peut moduler $a$ et $k$ pour renforcer ou non l'importance de l'effet exponentiel.

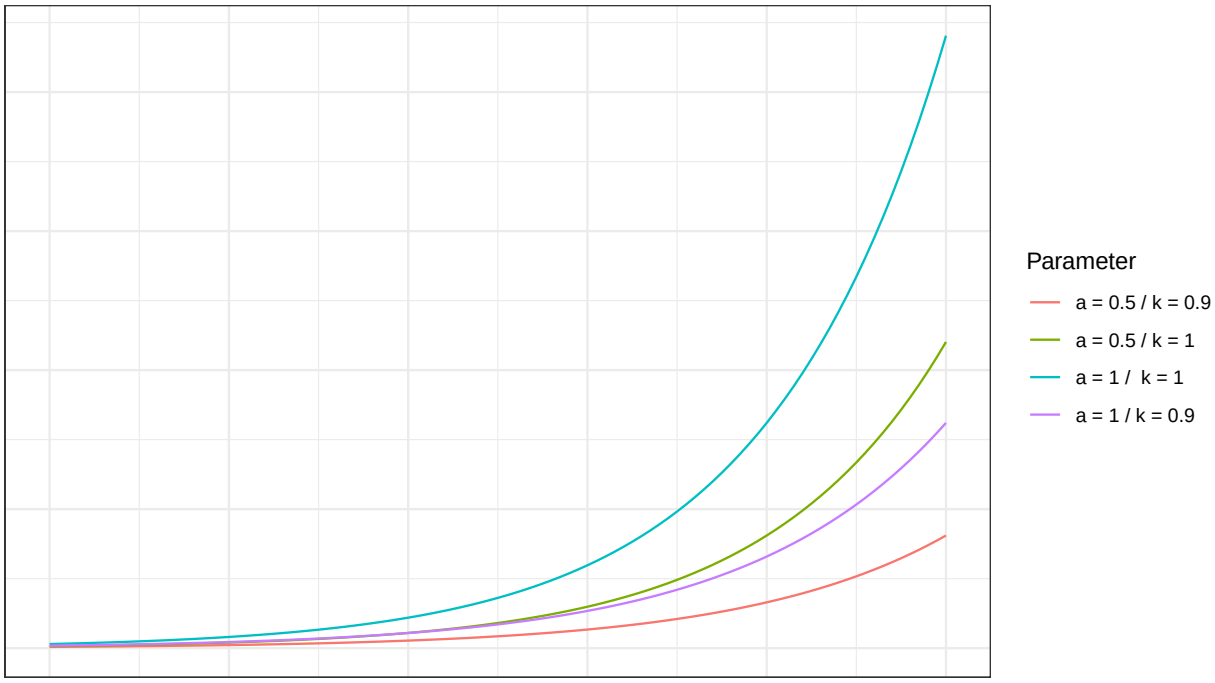



Figure II).1: Exemples de courbes exponetielles selon leurs paramètres

On peut aussi imaginer des valeurs négatives de $k$ pour avoir des courbes qui tendent vers 0.

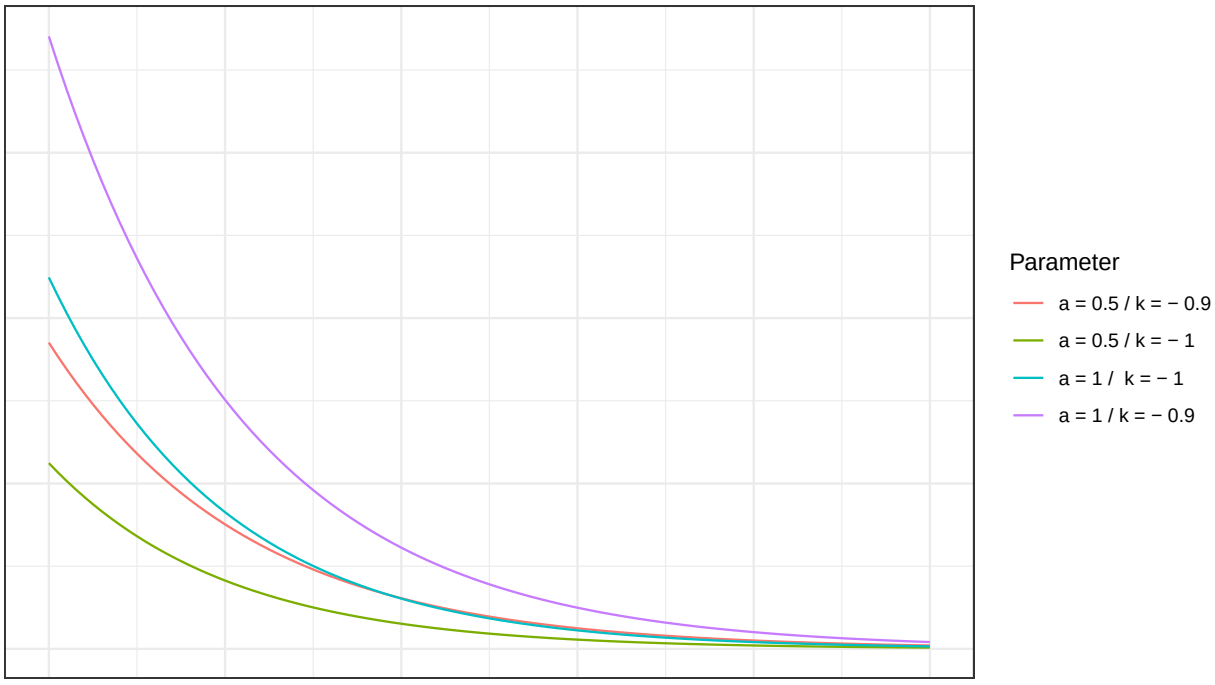



Figure II).2: Exemples de courbes exponetielles avec des coefficients négatifs

**d.1.2 Courbe asymptotique** On peut représenter des courbes avec un modèle asymptotique pour Y quand X tend vers l'infini.

$Y = a - (a - b)e^{-cX}$

avec :

- $a$ le maximum possible à atteindre
- $b$ la valeur de Y à X=0
- $c$ l'augmentation relative de Y par rapport à X.

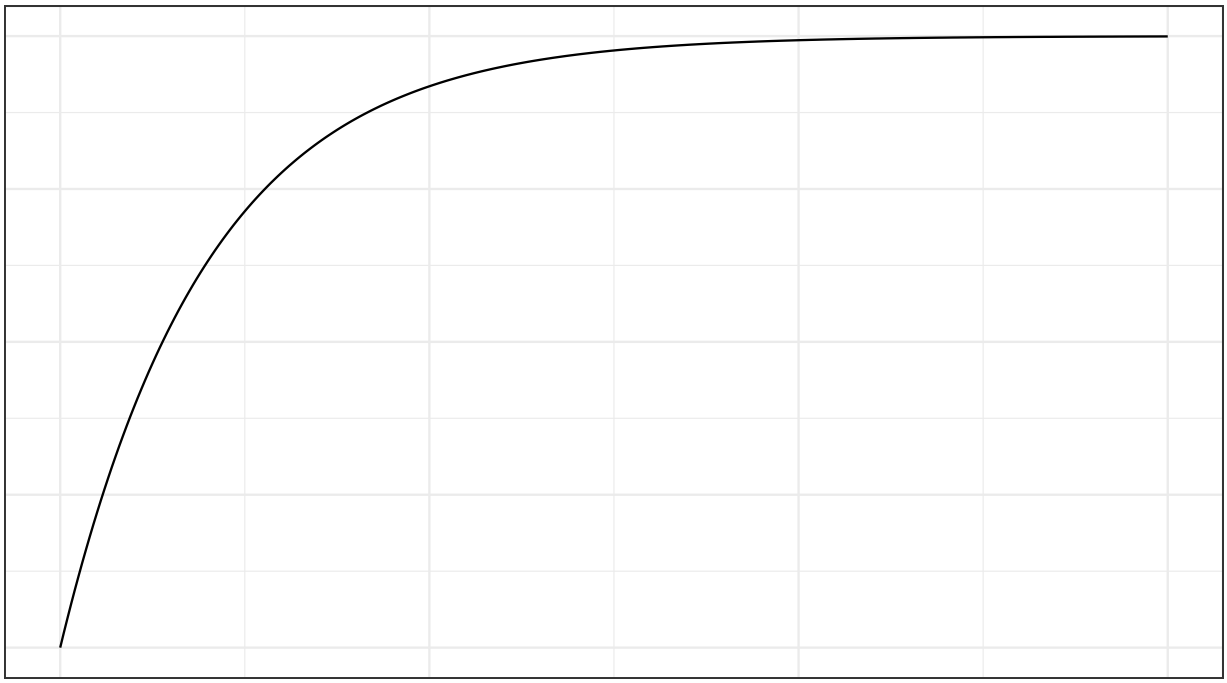


Figure II).3: Exemple de modèle asymptotique

Dans le cas particulier où $b$=0, on parle souvent d'équation exponentielle négative, car on obtient alors la formule suivant :

$Y = a[1 - e^{-cX}]$

On a donc Y=0 quand X=0, on parle alors de $c$ comme le coefficient d'extinction.

### d.2 Courbes sigmoïdales

$Y = c + \frac{d-c}{1+e^{b(X-e)}}$

Avec :

- $d$ : la valeur maximale d'asymptote
- $c$ : la valeur minimale d'asymptote
- $e$ : la valeur de X pour laquelle on est à mi-chemin entre $c$ et $d$
- $b$ : représente la pente au niveau du point d'inflexion .

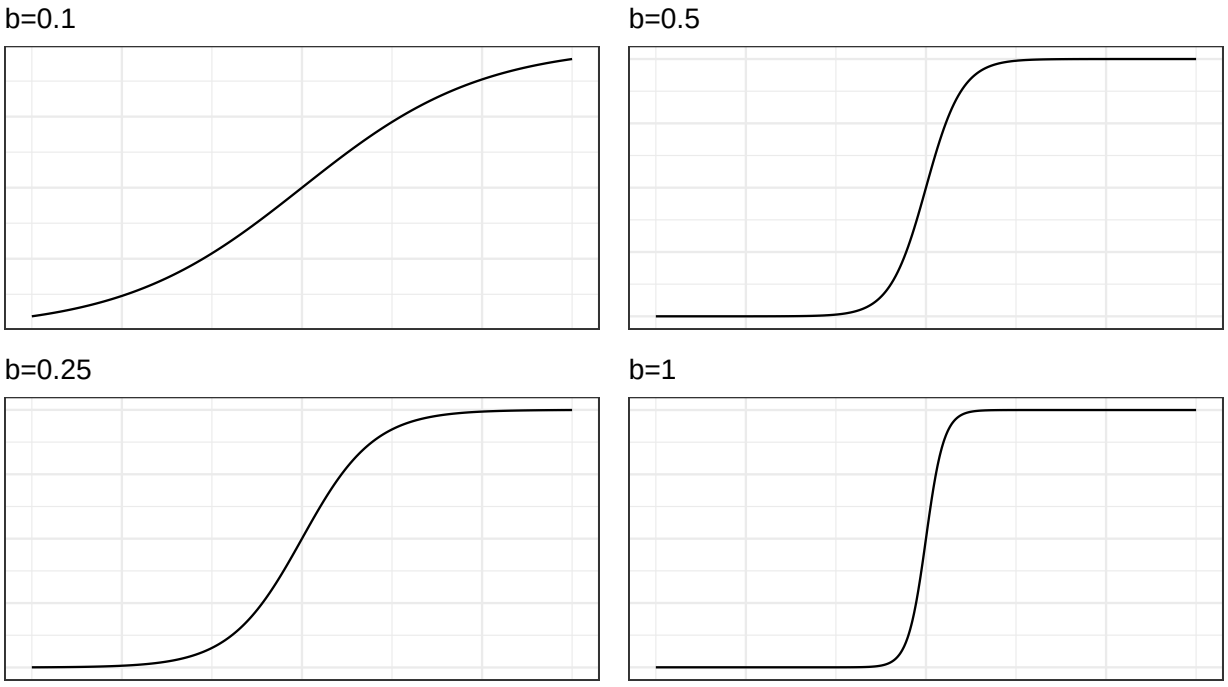



Figure II).4: Exemple du plusieurs courbes de régression logistique pour différentes valeurs de b

Cette fonction est utilisée dans le cadre d'une régression logistique à quatre paramètres. Dans une cas d'une régression logistique bimodale, on restreint alors $c$ à 0 et $d$ à 1. Il reste alors deux facteurs à estimer.

# III) Analyse Factorielle

# IV) Analyse Factorielle

Pour les analyses factorielles, un des packages des plus utilisés est *FactoMineR*, par sa simplicité, ses graphiques et ses extensions en clique-bouton en lien avec Rcommander. Pour information, nombreuses des méthodes implémentées dans ce package ont été codées par leur créateur (AFM, ACM, etc..). Le package *factoextra* sera aussi utilisé pour sa complémentarité avec *FactoMineR* et permettant la réalisation simple de beaux graphiques

catdes() for categories description dimdesc() for dimension description condes() for Continuous variables descriptions plotellipses() for confidence ellipses around categories after PCA or MCA

## a Notion d'inertie

L'inertie d'un jeu de données peut être considérée comme la quantité d'informations contenue au sein du jeu de données. On définit alors l'inertie $I$ de l'ensemble d'un groupe de données à partir de son centre de gravité $g$ :

$$I = \frac{1}{n} \sum_i^n d(e_i, g)^2,$$

avec $d$ la distance de chaque individu $e_i$ au centre de gravité $g$. Lors de la classification, notamment hiérarchique ascendante, on peut définir l'inertie intra-classe $I_a$ (information contenue dans l'ensemble des groupes) et l'inertie inter-classe $I_e$ (information non contenue dans les groupes) à partir des centres de gravité partiels $g_i$ de chacun des $k$ groupes :

$$I_e = \frac{1}{n} \sum_i^k n_i d(g_i, g)^2$$

$$I_a = \frac{1}{n} \sum_i^k \sum_j^{n_i} d(e_j, g_i)^2$$

Si une inertie est nulle, cela signifie que tous les individus sont identiques. Par définition, l'inertie $I$ équivaut à la somme des variances des $j$ variables du jeu de données. Si toutes les données sont centrées-réduites, alors l'inertie vaut $j$

## b ACP : Analyse en Composante Principale

### b.1 Cadre

La réalisation d'une ACP permet de répondre à de multiples objectifs : Permettre une représentation d'un jeu de données complexe en limitant le nombre de dimensions tout en conservant un maximum d'information, mais également d'étudier des corrélations multiples de façon simultanée.

Pour présenter l'ACP, on s'aidera du jeu de données *decathlon* présent dans le package *FactoMineR*. Il représente pour plusieurs individus leur résultats sur les dix épreuves d'un décathlon, leur classement

à la fin des épreuves, le nombre de points associés et le cadre dans lequel le décathlon a été réalisé.

```
library(FactoMineR)
data(decathlon)
kable(head(decathlon[,1:7]))
```

| | 100m | Long.jump | Shot.put | High.jump | 400m | 110m.hurdle | Discus |
|---|---|---|---|---|---|---|---|
| SEBRLE | 11.04 | 7.58 | 14.83 | 2.07 | 49.81 | 14.69 | 43.75 |
| CLAY | 10.76 | 7.40 | 14.26 | 1.86 | 49.37 | 14.05 | 50.72 |
| KARPOV | 11.02 | 7.30 | 14.77 | 2.04 | 48.37 | 14.09 | 48.95 |
| BERNARD | 11.02 | 7.23 | 14.25 | 1.92 | 48.93 | 14.99 | 40.87 |
| YURKOV | 11.34 | 7.09 | 15.19 | 2.10 | 50.42 | 15.31 | 46.26 |
| WARNERS | 11.11 | 7.60 | 14.31 | 1.98 | 48.68 | 14.23 | 41.10 |

Une question qui est posée est si certains sport sont plus discriminants pour chercher à atteindre les premières places. C'est dans ce cadre qu'on utilisera l'ACP pour regarder les liens entre variables.

```
res <- PCA(decathlon,graph=F,
           quanti.sup = 11:12, # Les variables 11 et 12 sont quantitatives supplémentaires
           quali.sup=13) # La variable 13  qualitative ne participera pas à la construction des axes.
```

On peut appliquer plusieurs fonctions au résultat obtenu :

- *dimdesc()* fpour la description des dimensions par les variables
- *plotellipses()* pour tracer des ellipses de confiances sur le plan factoriel pour les variables qualitatives

### b.2 Réduire ou ne pas réduire, telle est la question

Avant de lancer l'ACP, une question qui peut se poser est de réduire ou non les données utilisées. Le centrage est automatique pour permettre de ramener toutes les variables avec une même moyenne (à savoir 0). La réduction n'est pas automatique et le fait de la réalisation ou non influe fortement sur les résultats. La réduction induit que chaque variable voit sa variance réduite à 1 et donc que toutes les variables apporteront à l'ACP la même quantité de données. Cela permet notamment de comparer des variables quantitatives non comparables en temps normal, comme c'est le cas ici. Comparer un temps aux 100 mètres en seconde et une longueur de saut en mètre n'a pas vraiment de sens et donc la question ne se pose pas. Cependant, si les variables sont dans la même unité, la question revient à savoir si on souhaite permettre que chaque variable apporte la même quantité d'information ou si on souhaite que celles avec de plus grandes variances soient de bases discriminantes. .

### c AC : Analyse des Correspondances

### d ACM : Analyse des Correspondances Multiples

équivalent l'ACP pour données que qualitatives

### e AFDM : Analyse Factorielle de Données Mixtes

méthode permettent d'utiliser des données quali et quanti dans un même tableau d'analyse

### f AFM: Analyse Factorielle Multiple

```
data(wine)
res <- MFA(wine, group=c(2,5,3,10,9,2), type=c("n",rep("s",5)),
    ncp=5, name.group=c("orig","olf","vis","olfag","gust","ens"),
    num.group.sup=c(1,6),graph=F)
```

Dual MFA et AFMH en supplément

### g GPA :

### h Analyse Canonique

# V) Régression

Documents sources utilisés:

- Linear Regression, via le site R-statistics.co
- Modèle Linéaire, D. Chessel & J. Thioulouse, Université de Lyon 1
- Plusieurs cours d'Agrocampus Ouest (document Perso)

## a Modèle Linéaire : Régression simple, multiple, Anova et Ancova

Le modèle linéaire classique (LM for linear model) est celui le plus utilisé. Il permet de lier une variable réponse à une ou plusieurs variables explicatives. On a tendance à découper beaucoup de modèle dedans, Anova, régression simple, régression multiple ou ANVOCA. Au final, toutes ces approches sont toutes issues d'une même théorie, le modèle linéare. Dans ce cas, la variable réponse est quantitative, c'est cela qui regroupe ces modèles, et suit une loi Normale. Du coup, quelles différences entre ces éléments ?

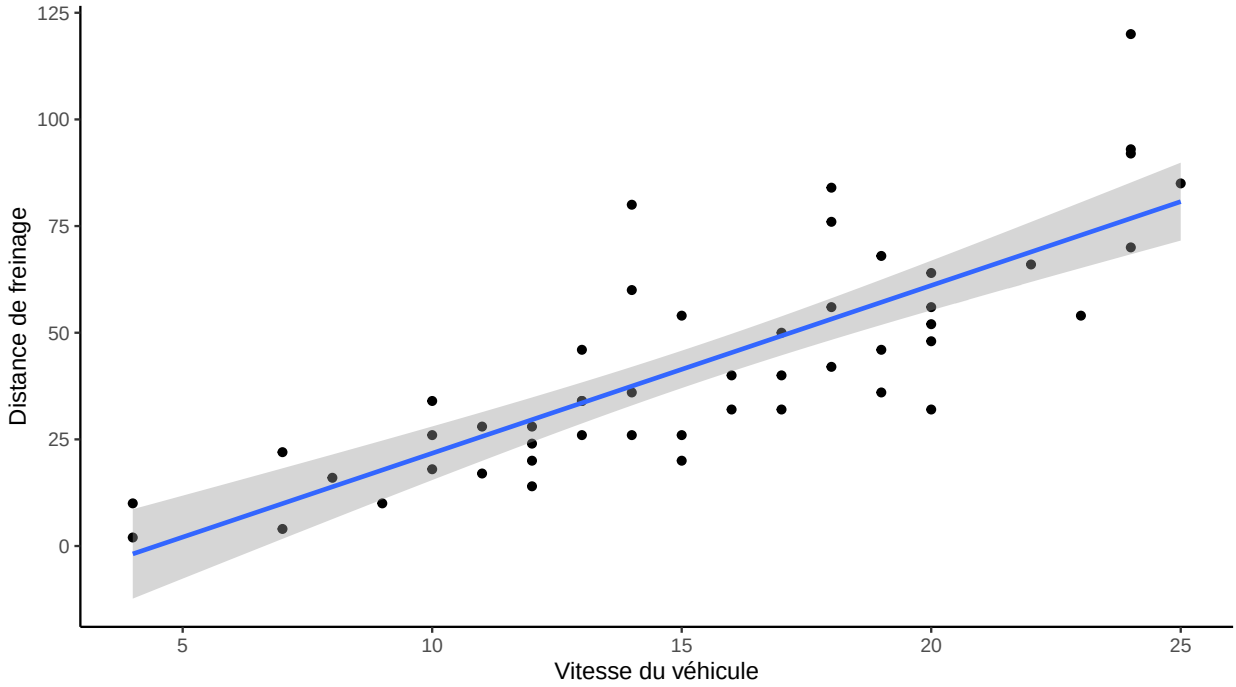
- ANOVA : Analyse de l'effet d'un ou plusieurs facteurs qualitatif
- Régression simple : Analyse de l'effet d'une variable quantitative
- Régression multiple : Analyse de l'effet de plusieurs variables quantitatives
- ANCOVA : Analyse de l'effet de plusieurs facteurs , avec des variables qualitatives et quantitatives

Un petit exemple, on cherche à la distance de freinage à la vitesse d'un véhicule (jeu de données cars)

```
library(ggplot2) # pour les graphiques
data(cars)
str(cars) # Avoir un apercu des données, et notamment de leur natures
```

```
% 'data.frame': 50 obs. of  2 variables:
%  $ speed: num  4 4 7 7 8 9 10 10 10 11 ...
%  $ dist : num  2 10 4 22 16 10 18 26 34 17 ...
```

```
mod<-lm(dist~speed,data=cars) # Modèle linéaire
p<-ggplot(data=cars,aes(x=speed,y=dist))+geom_point()+
  theme_classic()+labs(x="Vitesse du véhicule",y="Distance de freinage")+
  geom_smooth(method="lm") # Permet d'avoir une estimation de la pente de régression linéaire
print(p)
```




### a.1 Un choix anodin mais primordial : les contrastes

Un élément très important lors de toutes les régressions est le choix des contrastes. Il existe plusieurs façons d'aborder les contrastes lors des modèles, particulièrement d'Anova :

- contr.sum se traduit en statistique par $\sum_i \alpha_i = 0$ (Statistiques à la française)
- contr.treatment se traduit en statistique par $alpha_1 = 0$. Dans ce cas, l'estimation de la moyenne (l'intercept) se fait donc pour le niveau 1 de toutes les valeurs observées en ANOVA. (Statistiques à l'anglo-saxonne)
- contr.helmert : returns Helmert contrasts, which contrast the second level with the first, the third with the average of the first two, and so on

C'est important si on veut intérpréter les coefficients directement, à partir des sorties du modèle linéaire, notamment sur des variables qualitatives. MAis on peut passer cette étape à l'aide de package.

```
summary(mod)
```

```
%
% Call:
% lm(formula = dist ~ speed, data = cars)
%
% Residuals:
%     Min      1Q  Median      3Q     Max
% -29.069  -9.525  -2.272   9.215  43.201
```

```
%
% Coefficients:
%             Estimate Std. Error t value Pr(>|t|)
% (Intercept) -17.5791     6.7584  -2.601   0.0123 *
% speed         3.9324     0.4155   9.464 1.49e-12 ***
% ---
% Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
%
% Residual standard error: 15.38 on 48 degrees of freedom
% Multiple R-squared:  0.6511,  Adjusted R-squared:  0.6438
% F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

Cette sortie donne la moyenne pour une vitesse nulle est de -17.58, et on prend 3.93 m de freinage tous les 1km/h gagnés.

### b Estimation des coefficients

On va profiter du jeu de données *diamonds* pour cela. On va s'intéresser à l'effet du poids, de la taill, la couleur et de sa clareté sur le prix

```
data("diamonds")
mod=lm(price~cut+color+clarity+carat,diamonds)
library(car)
Anova(mod,type="III")
```

```
% Anova Table (Type III tests)
%
% Response: price
%                 Sum Sq    Df   F value    Pr(>F)
% (Intercept) 9.4276e+10     1  70444.50 < 2.2e-16 ***
% cut         1.6992e+09     4    317.41 < 2.2e-16 ***
% color       1.6320e+10     6   2032.46 < 2.2e-16 ***
% clarity     3.8452e+10     7   4104.51 < 2.2e-16 ***
% carat       7.2976e+11     1 545289.36 < 2.2e-16 ***
% Residuals   7.2163e+10 53921
% ---
% Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

On peut voir que toutes les variables sont ultra signigicatifes, avec des probabilités en deça de la limiete de calcul de R (2.2e-16).

Comment avoir des estimer des moyennes pour chaque niveau de facteur

```
library(emmeans)
emmeans(mod,"cut")
```

```
%  cut       emmean   SE    df lower.CL upper.CL
%  Fair        2706 29.4 53921     2648     2763
%  Good        3361 17.8 53921     3326     3396
%  Very Good   3554 12.5 53921     3530     3579
%  Premium     3575 12.0 53921     3551     3599
%  Ideal       3704 10.1 53921     3684     3724
%
% Results are averaged over the levels of: color, clarity
% Confidence level used: 0.95
```

et si on croise

```
mod2=lm(price~(cut+color+clarity)^2+carat,diamonds)
# le ^2 permet de tester toutes les combinaisons de niveau 2 au max


emmeans(mod2,c("cut","color"))
```

```
%  cut       color emmean    SE    df lower.CL upper.CL
%  Fair      D       3935 108.0 53827     3724     4146
%  Good      D       4629  57.2 53827     4517     4742
%  Very Good D       4807  43.5 53827     4722     4893
%  Premium   D       4975  42.5 53827     4892     5059
%  Ideal     D       4962  36.5 53827     4890     5033
%  Fair      E       3611  99.0 53827     3416     3805
%  Good      E       4158  46.4 53827     4067     4249
%  Very Good E       4285  32.1 53827     4222     4348
%  Premium   E       4418  32.1 53827     4355     4481
%  Ideal     E       4351  27.8 53827     4297     4406
%  Fair      F       3486  85.6 53827     3318     3654
%  Good      F       3892  45.6 53827     3803     3981
%  Very Good F       4183  32.3 53827     4120     4247
%  Premium   F       4231  30.6 53827     4171     4291
%  Ideal     F       4313  23.7 53827     4267     4360
%  Fair      G       2896  86.7 53827     2726     3066
```

```
%  Good      G       3591  45.0 53827     3503     3679
%  Very Good G       3842  31.1 53827     3781     3903
%  Premium   G       3913  26.6 53827     3861     3965
%  Ideal     G       4054  23.5 53827     4008     4100
%  Fair      H       2201  90.8 53827     2023     2379
%  Good      H       3257  51.0 53827     3157     3357
%  Very Good H       3319  34.9 53827     3251     3387
%  Premium   H       3254  29.9 53827     3195     3312
%  Ideal     H       3531  25.3 53827     3482     3581
%  Fair      I       1884 107.0 53827     1675     2093
%  Good      I       2677  58.1 53827     2563     2791
%  Very Good I       2897  42.1 53827     2815     2980
%  Premium   I       2834  37.5 53827     2760     2907
%  Ideal     I       3116  32.2 53827     3053     3180
%  Fair      J       1055 125.0 53827      809     1300
%  Good      J       1869  76.5 53827     1719     2019
%  Very Good J       2085  56.5 53827     1974     2195
%  Premium   J       1868  52.2 53827     1766     1970
%  Ideal     J       2287  51.0 53827     2187     2387
%
% Results are averaged over the levels of: clarity
% Confidence level used: 0.95
```

Et si on veut savoir les niveaux qui se recoupent ?

```
library(multcomp)
cld(emmeans(mod2,c("cut")))
```

```
%  cut       emmean   SE    df lower.CL upper.CL .group
%  Fair        2724 65.8 53827     2595     2853  1
%  Good        3439 28.9 53827     3382     3496   2
%  Very Good   3631 21.0 53827     3590     3673    3
%  Premium     3642 17.7 53827     3607     3676    3
%  Ideal       3802 15.8 53827     3771     3833     4
%
% Results are averaged over the levels of: color, clarity
% Confidence level used: 0.95
% P value adjustment: tukey method for comparing a family of 5 estimates
```

```
% significance level used: alpha = 0.05
% NOTE: If two or more means share the same grouping symbol,
%       then we cannot show them to be different.
%       But we also did not show them to be the same.
```

## c L'analyse de Variance

## d Modèles à effet fixe

L'objectif est d'expliquer une variable quantitative par des variables qualitatives, aussi appelées facteurs (à plusieurs modalités). Cela revient donc à expliquer la variabilité . Celle-ci se mesure par sa dispersion, et se quantifie par sa variance.

### d.1 Analyse de variance à un facteur

### d.2 Modèle Mixte

Explication des effets aléatoires

```
library(lme4)
library(emmeans)
data("Theoph")
mod<-lmer(conc~Subject+(1|Time),data=Theoph)
# mod$residuals
resid(mod)
```

```
%           1           2           3           4           5           6
% -1.058248092 -1.500667320 -0.065553428  1.004809869  0.486435289  0.481884289
%           7           8           9          10          11          12
%  0.421965733  0.603905533  0.021600835  0.565475490 -0.961608198 -0.080718408
%          13          14          15          16          17          18
% -1.217214356  2.468593253 -0.117960850  0.881576378 -0.322332706  0.072475212
%          19          20          21          22          23          24
%  0.251435217 -0.197441006 -0.596376060 -1.142036675 -0.688350630  0.855153422
%          25          26          27          28          29          30
%  1.035643117  1.448867553 -0.263667248  0.490027074  0.135962879 -0.746957090
%          31          32          33          34          35          36
% -0.455073227 -0.544929803 -1.266676047 -0.191733244 -0.902638848 -0.164551180
%          37          38          39          40          41          42
%  0.924877112  0.864958556  0.256652459  0.761460377 -0.202894055  0.034269483
%          43          44          45          46          47          48
% -0.276217580 -1.104183082 -0.804470031 -3.131706720 -0.535158370  2.248287527
```

```
%           49           50           51           52           53           54
%  1.150213350  0.843915672  0.828723590  0.494369158  0.022629817  0.039872317
%           55           56           57           58           59           60
% -1.156676309  1.121585614 -0.444910334 -0.974420640  0.694277515  0.661594666
%           61           62           63           64           65           66
%  0.446432578 -0.423515637  0.035173398 -0.117346563 -0.189737067 -0.809133529
%           67           68           69           70           71           72
%  0.949834207 -0.892585020 -1.489459361 -0.242947609  0.004517589  0.783678341
%           73           74           75           76           77           78
%  0.974732957  0.282541326  0.523111610 -0.059301365 -0.834122673  0.507602521
%           79           80           81           82           83           84
%  1.015183293 -1.803085818  0.456695298  0.692285902  0.993186130 -0.470740263
%           85           86           87           88           89           90
%  0.061324281  0.017747150 -0.583720161 -0.886478333 -0.140042932  2.882720379
%           91           92           93           94           95           96
%  1.056068876  0.541314008 -1.185359551 -0.584459323 -0.396849312 -0.248521504
%           97           98           99          100          101          102
% -0.283927924 -0.542667143 -1.098275572 -0.919995160 -0.893994756 -0.259402776
%          103          104          105          106          107          108
% -0.812776977  0.451449185  1.099659019  1.161662056  0.503197028  0.263522804
%          109          110          111          112          113          114
%  0.428682159 -1.022002581 -0.216756188  2.100824584  2.383950243  0.422336589
%          115          116          117          118          119          120
%  0.430542776  0.174527127 -0.923562567 -0.834602563 -1.244245586 -1.103032606
%          121          122          123          124          125          126
% -1.189981809 -0.354866635 -1.647285862 -1.034160203 -0.882109076  1.185486501
%          127          128          129          130          131          132
%  1.193657214  0.872275867  0.876526906  1.107643968 -0.174002206 -1.143166473
```

```
emmeans(mod,c("Subject"))
```

```
%  Subject emmean    SE    df lower.CL upper.CL
%  6         3.89 0.637 110.3     2.63     5.15
%  7         4.21 0.579  95.6     3.06     5.36
%  8         4.50 0.581  95.5     3.35     5.66
%  11        5.23 0.588  97.5     4.06     6.40
%  3         5.70 0.597  99.6     4.52     6.88
```

```
%  2         5.09 0.572  93.1     3.96     6.23
%  4         5.20 0.650 114.4     3.92     6.49
%  9         5.15 0.625 108.2     3.91     6.39
%  12        5.37 0.606 101.8     4.17     6.57
%  10        6.17 0.651 114.4     4.88     7.46
%  1         6.81 0.621 107.4     5.58     8.04
%  5         5.82 0.579  95.2     4.67     6.97
%
% Degrees-of-freedom method: kenward-roger
% Confidence level used: 0.95
```

#### d.3 Modèle Hiérarchique

l'aléatoire c'est pas assez complexe

## e Modèle Linéaire Généralisé

## f Régression Curvilinéaire

## g Régression non-linéaire

### g.1 Exemples d'applications

**g.1.1 Modèle systèmique** On s'intéresse à la dynamique de la digestion d'une molécule chez un poisson. Pour cela, on suit la position de la molécule dans les différentes parties des poissons. On obtient un modèle à 3 compartiments, où la molécule peut avancer selon le schéma présenté après. On obtient alors à 3 équations différnetielle, pour expliquer la dynamique de transfert entre compartiments, comme représenté sur la figure V).1.

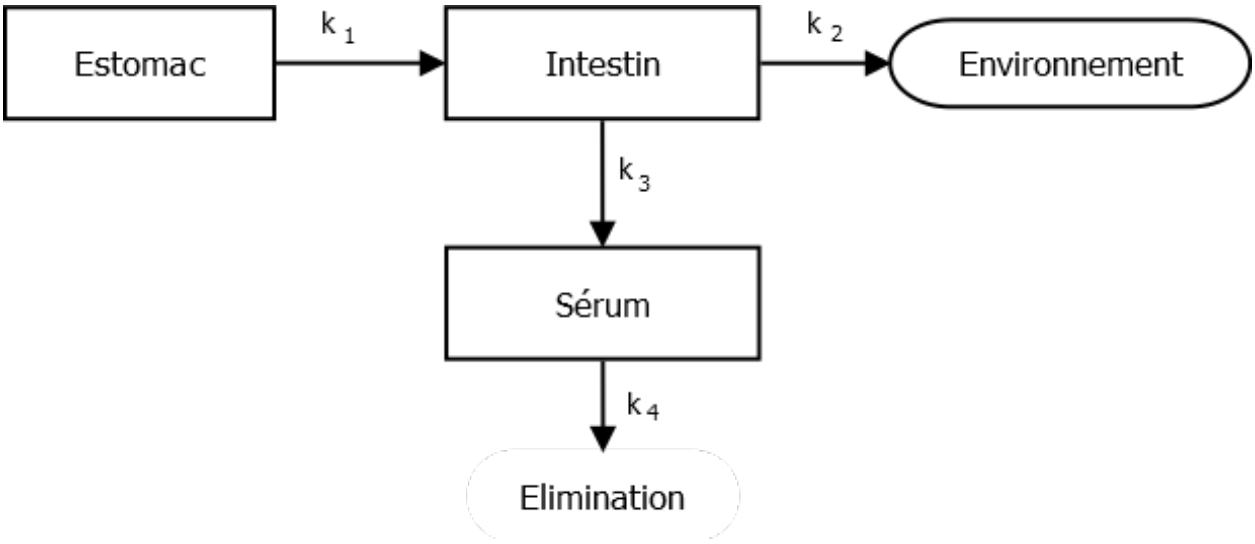



Figure V).1: Modèle cinétique biologique

On obtient alors trois équations différentielles pour exprimer l'évolution de la molécule dans les compartiments qui nous intéressent :

- Estomac : $\frac{dq_1}{dt} = -k_1 q_1(t)$
- Intestin : $\frac{dq_2}{dt} = k_1 q_1(t) - [k_2 + k_3] q_2(t)$
- Sérum : $\frac{dq_3}{dt} = k_3 q_2(t) - k_4 q_3(t)$

Si on les résoud, on obtient alors :

- Estomac : $q_1(t) = q_0 e^{-k_1 t}$
- Intestin : $q_2(t) = \frac{k_1 q_0}{k_2 + k_3 - k_1} f_1(t)$
- Sérum :$q_3(t) = f_3(t)$

où $f_1(t)$ et $f_2(t)$ sont des fonctions non-linéaires du temps. On ne peut donc pas réaliser une régression linéaire pour estimer ces deux fonctions. On obtient donc le problème que les systèmes non-linéaires, les méthodes vues précédemment ne sont donc pas applicables. Il faut chercher un autre moyen d'expliquer la variable réponse à partir des variables explicatives.

**g.1.2 Modélisation allométrique** La modélisation allométrique est souvent utilisée pour suivre l'évolution de la part d'un animal (organe, muscles, lipides, etc..) par rapport au poids/volume total. Par exemple, on suit ici l'évolution du poids des lipides par rapport au poids d'un animal. On obtient souvent des équations de la forme :

$$\frac{d(\frac{Y}{x})}{dx} = \beta_1 \frac{Y}{x}$$

ce qui donne après résolution : $Y = \beta_0 x_1^{\beta} + \epsilon_i$. Selon la valeur de beta, plusieurs types de croissances sont distinguables (tardive, isométrique ou précose, dans l'ordre croissant)

**g.1.3 Cinétique en biochimie** Quand on étudie de la cinétique, en biochimie notamment, on étudie également des des modèles non-linéaires. (Il ne faut pas cependant la non-linéarité quand on sait faire en linéaire). Pax exemple, on s'intéresse à la décompostion du $\beta$-lactose selon différentes concentrations en NaCl

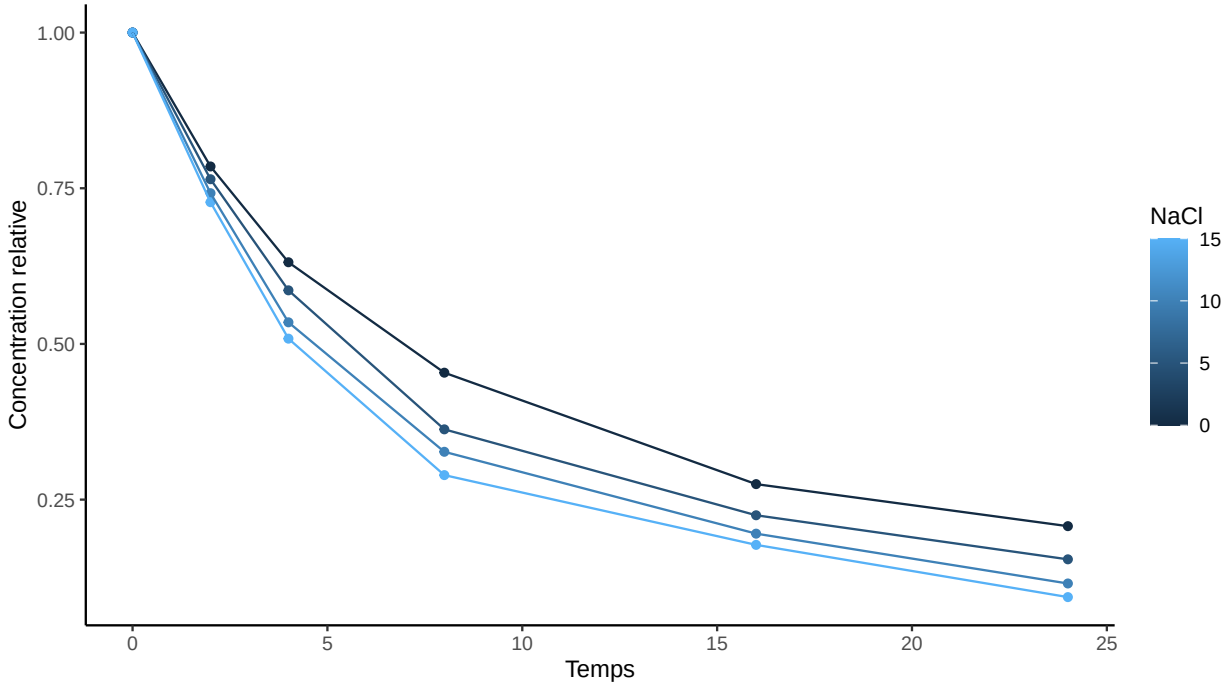



Figure V).2: Evolution de la concentration relative en fonction du temps

On s'intéresse à l'évolution de la concentration en lactose en fonction du temps, en évolution relative. Pour cela, on utilise le modèle suivant , avec $Y_t$ la concentration en $\beta$-lactose à l'instant $t$.

$$\frac{dY_t}{dt} = -kY_t^{\mu}$$

$$Y_t = Y_0[1 + (\mu - 1)kY_0^{\mu-1}t]^{\frac{1}{1-\mu}} + \epsilon_t$$

$$\epsilon \hookrightarrow N(0; \sigma)$$

où $\mu$ et $k$ dépendent des conditions expérimentales. $1 \leq \mu \leq 2$, k>0.

On a 2 paramètres inconnues : $k$ et $\mu$, 1 seule variable explicative $t$.

$$Y_t = f(t, k, \mu) + \epsilon_t$$

Il faut transformer cela, ici on teste avec le log.

$$log(Y_t) = log(Y_0) + log([1 + (\mu - 1)kY_0^{\mu-1}t]^{\frac{1}{1-\mu}} + \epsilon_t)$$

Mais cela ne marche pas. Il faut chercher la valeur qui minimise l'écart au modèle

$$SC(k,\mu) = \sum_{i=1}^{n}(Y_{t_i} - f(t_i,k,\mu))^2$$

### g.2 Base de la régression

#### g.2.1 Modèle

$$Y = f(x^1, x^2, ..., x^p; \beta_1, \beta_2, ..., \beta_q) + \epsilon, \text{ où } \epsilon \hookrightarrow N(0,\sigma)$$

$$Y = \hookrightarrow N[f(x^1, x^2, ..., x^p, \sigma]$$

Où la fonction $f$ est ma surface de réponse du modèle.

Dans le cas du modèle allométrique, on obtient la formule suivante : $Y = \beta_0 x_1^{\beta} + \epsilon_i$

Mais attention, cela est différent de $log(Y) = log(\beta_0) + \beta_1 log(x) + \epsilon$

**g.2.2 Estimation des paramètres** Pour estimer les paramètres, on utilise le critère des moindres carrés.

$$SC(\beta) = \sum_{i=1}^{n}[Y_i - f(x_i;\beta)]^2$$

On obtient que la dérivée partielle de la somme des carrés par rapport à chaque coefficient $\beta$ est nulle. Par exemple, dans le cas allométrique, on obtient les deux équations estimantes :

$$-2\sum_{i=1}^{n} x_i^{\hat{\beta}_1}[Y_i - \hat{\beta}_0 x_i^{\hat{\beta}_1}] = 0$$

$$-2\sum_{i=1}^{n} \hat{\beta}_0 x_i^{\hat{\beta}_1} log(x_i)[Y_i - \hat{\beta}_0 x_i^{\hat{\beta}_1}] = 0$$

Initialisation de l'algorithme d'estimation

On prend l'exemple de NaCl :

Pour trouver le résultat, il faut passer aux dérivées partielles (par rapport à $k$ et $\mu$). Cependant, quand on cherche à résoudre les équations à 2 inconnues, il n'y a pas de solution évidente. On va chercher la fonction à l'aide d'un algorithme pour optimiser ce critère. Le problème est de trouver $k$ et $\mu$ qui annulent la fonction. Il 'agit de l'algorithme de Newton-Raphson. Pour cela, on utilise les tangeantes et leur propriété mathématiques. On calcule la pente de la tangeante à un point, on prend le point d'intersection avec les abcisses, on calcule alors la pente de la tangeante au point associée à cette valeur de x. On continue jusqu'à ce que le point obtenu soit celui de l'intersection avec les abcisses pour la fonction. On utilise le non-linear least squares, nls dans R.On va travailer sur la concentration relative plutot que sur la concentration réele.

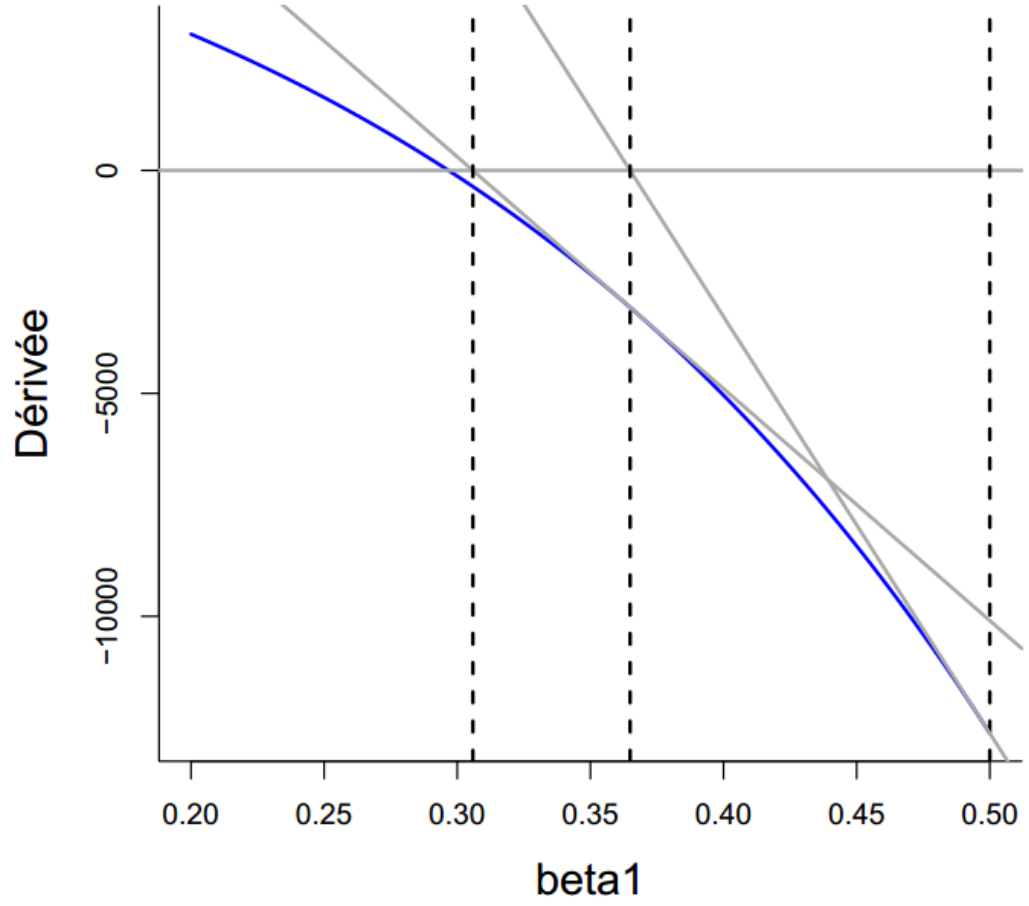



Figure V).3: Schema de résolution par l'algorithme de Newton-raphson

Pour commencer, il faut choisir initialiser $k$ et $\mu$. On décide de prendre $\mu$ au milieu de l'intervalle dépendant des conditions exprimentales, à savoir [1;2]. Pour k, on prend la pente de la tangeante à la droite de la régression jusqu'à un temps de 8 secondes. On déduit sa valeur initiale à partir de la pente à la tangeante qu'on obtient pour la régression linéaire de la tangeante. On sait à ce moment donné que l'équation à la tangeante vérifie l'équation. En résolvant l'égalité entre la tangeante et l'équation de concentratation relative (connaissant $t$ et $\mu$, on peut faire une estimation de $k$ pour sa valeur initiale).

```
head(lactose)
```

```
%   NaCl       Ct     CtRel Temps
% 1    0 5.313809 1.0000000     0
% 2    0 4.171195 0.7849727     2
% 3    0 3.353797 0.6311475     4
% 4    0 2.411540 0.4538251     8
% 5    0 1.460572 0.2748634    16
```

```
% 6    0 1.101962 0.2073770    24
```

```
y=log(lactose$CtRel)
x=lactose$Temps
lm(y[x<=8]~-1+x[x<=8]) # On obtient le l coefficient via l'équation à la tangeante
```

```
%
% Call:
% lm(formula = y[x <= 8] ~ -1 + x[x <= 8])
%
% Coefficients:
% x[x <= 8]
%   -0.1332
```

On a donc comme équation que :

$$Log(Y_t) = -0.1332 \times t$$

et à partir de l'équation précédente, pour t = 4, $Y_0$ = 1 et $\mu$ = 1.5, on obtient que :

$$log(Y_t) = log(Y_0) + log([1 + (\mu - 1)kY_0^{\mu-1}t]^{\frac{1}{1-\mu}} + \epsilon_t)$$

$$log(Y_t) = log(1) + log([1 + (1.5 - 1)kY_0^{1.5-1}t]^{\frac{1}{1-1.5}})$$

$$log(Y_t) = log([1 + (0.5)kY_0^{0.5}t]^{\frac{1}{-0.5}})$$

$$log(Y_t) = log([1 + (0.5)kY_0^{0.5}t]^{-2})$$

et donc à partir de l'équation de tangeante :

$$-0.1332 \times t = log([1 + (0.5)kt]^{-2})$$

$$10^{-0.1332t} = [1 + (0.5)kt]^{-2}$$

$$10^{-0.1332t*-0.5} = [1 + 0.5kt]$$

$$(10^{0.0666t} - 1)/(0.5t) = k$$

On trouve alors comme valeur initiale k = 0.41, que l'on ba intégrer dans le modèle pour l'aider à converger

```
nls(CtRel~(1+(mu-1)*k*Temps*1^(mu-1))^(1/(1-mu)),data=lactose,start=list(k=0.41,mu=1.5))
```

```
% Nonlinear regression model
%   model: CtRel ~ (1 + (mu - 1) * k * Temps * 1^(mu - 1))^(1/(1 - mu))
%    data: lactose
%      k     mu
```

```
% 0.1702 1.6550
%  residual sum-of-squares: 0.0422
%
% Number of iterations to convergence: 5
% Achieved convergence tolerance: 3.51e-06
```

On obtient à partir de la fonction *nls* (pour non linear least squares), les estimations les plus proches de k et $\mu$ indépendement du traitement en NaCL.

### g.3 Test d'effets non-linéaires

**g.3.1 Comparaisons de modèle** On peut alors se demander si la concentration en chlorure en sodium possède un effet. On réalise une régression pour chaque concentration en NaCl, et on récupère les coefficients $k$ et $\mu$ de chaque régression. On fait alors une régression linéaire sur les valeurs des coefficients en fonction de la concentration. On estime pour cela $k$ et $\mu$ pour chacune des concentrations de NaCL et on réalise une régression linéaire.

```
%
% Call:
% lm(formula = k ~ NaCl, data = k)
%
% Coefficients:
% (Intercept)         NaCl
%    0.140510     0.003855

% Anova Table (Type III tests)
%
% Response: k
%                Sum Sq Df F value    Pr(>F)
% (Intercept) 0.0282044  1 2187.39 0.0004569 ***
% NaCl        0.0018574  1  144.05 0.0068705 **
% Residuals   0.0000258  2
% ---
% Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

% Anova Table (Type III tests)
%
% Response: nu
%             Sum Sq Df  F value    Pr(>F)
% (Intercept) 4.5378  1 2837.071 0.0003523 ***
% NaCl        0.0462  1   28.896 0.0329079 *
```

```
% Residuals   0.0032  2
% ---
% Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

On observe qu'on a donc un effet du chlorure de potassium sur $k$ et $\mu$. Il faut donc tenir compte de la concentration de NaCl dans le modèle. On teste alors à l'aide de fisher si cette effet marqué sur les deux variables est important dans notre modèle. On étude alors l'influence de NaCL sur $k$ et $\mu$. On va donc comparer les modèles aux différentes concentrations pour voir si cet effet est significatif. On va du coup réaliser via les SCER entre les sous-modèles et le modèle simple.

```
SCER<-sum((residuals(mod))^2)
SCER0<-sum((residuals(mod0))^2)
SCER5<-sum((residuals(mod5))^2)
SCER10<-sum((residuals(mod10))^2)
SCER15<-sum((residuals(mod15))^2)
SCERss<-SCER0+SCER5+SCER10+SCER15
fis<-((SCER-SCERss)/6)/(SCERss/16)
pf(fis,6,16,lower.tail=FALSE)
```

```
% [1] 1.107859e-06
```

On obtient une p-value au test de Fisher très faible, qui montre qu'il faut tenir compte de l'effet de la concentration en Nacl sur les deux coefficients dans la cinétique du $\beta$-lactose.

#### g.3.2 Tests de nullité des coefficients ...

# VI) Régression Logistique

# VII) Sélection de modèles

Dans l'exemple, on va s'intéresser à l'épaisser de gras dorsal. Il s'agit d'une référence fixée par le marché mesuré dans les abbatoirs de porcs, et qui permet une évaluation indirecte par mesure sur la carcasse. Il existe différents type d'instruments pour essayer d'évaluer la qualité de la viande de porcs, mesurant l'épaisseur de gras et de muscle. L'objectif est d'obtenir une bonne prédiction du TMP. Il existe aussi des scanners maintenant qui apportent plus d'informations car ils étudient des zones plus larges, mais leur emploi est plus couteux. Pour cela, on dispose de différentes zones de mesure, ainsi que la mesure exacte du TMP.

```r
TMP<-read.table("Regression/19624_DIS05.txt",sep=",",header=T) #issu des données Agrocampus
```

L'objectif est de construitre une équation (donc un modèle) de prédiction (ici du TMP) à l'aide de mesure indirecte. Il s'agit d'une mise en équation du vivant, de ce que l'on observe, de quelque chose qu'on ne maitrise pas dans le but d'expliquer et/ou prédire.

Modèle de régression linéaire :

$$Y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

où :

- $p$ est le nombre de variables (ici le nombre d'épaisseurs tissulaires mesurées)
- $\epsilon$ est l'erreur résiduelle d'écart-type $\sigma$

Comment peut-on estimer un modèle ? Comment le valide-t-on ? On va partir de modèles connus (linéaires) vers non connus (non linéaires) à partir de mesures indirectes pour obtenir les prédictions qui nous intéressent et un modèle de référence (ex: modèle de prédiction du TMP). Cependant, s'il y a beaucoup de variables, la statistique est incapable de spécifier la non-linéarité, à cause du nombre de composantes trop élevé.

Les modèles non-linéaires sont souvent avec 2 ou 3 variables, que l'on décompose toujours en deux parties, une "prévisible" (tendance moyenne) et une autre variable.

Par exemple, il y al test du CGM (Capteur de Gras Maigre) dans le cas du TMP qui associé à des lieux mesures de références.

Si l'on observe le tableau des corrélations des variablex explicatives, on observe une redondance de l'information, peut-être un sous-modèle serait-il plus intéressant (3 blocs de variables corrélées).

| | TMP | CHMUSCLE | RMUSCLE | MU23DCFR | MU34DCFR | G1CGM | GR34VLFR | CHGRAS | FRGRAS | G2CGM | GR34DCFR | GR23DCFR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TMP | 1 | 0.45 | 0.43 | 0.34 | 0.28 | −0.71 | −0.81 | −0.7 | −0.72 | −0.76 | −0.78 | −0.84 |
| CHMUSCLE | 0.45 | 1 | 0.97 | 0.68 | 0.62 | 0.01 | −0.15 | −0.21 | −0.25 | −0.11 | −0.17 | −0.26 |
| FRMUSCLE | 0.43 | 0.97 | 1 | 0.64 | 0.59 | 0.01 | −0.14 | −0.18 | −0.21 | −0.09 | −0.14 | −0.22 |
| MU23DCFR | 0.34 | 0.68 | 0.64 | 1 | 0.9 | 0.11 | −0.06 | −0.03 | −0.08 | −0.1 | −0.12 | −0.22 |
| MU34DCFR | 0.28 | 0.62 | 0.59 | 0.9 | 1 | 0.15 | −0.07 | −0.01 | −0.03 | −0.02 | −0.09 | −0.13 |
| G1CGM | −0.71 | 0.01 | 0.01 | 0.11 | 0.15 | 1 | 0.9 | 0.72 | 0.72 | 0.86 | 0.81 | 0.81 |
| CHGRAS | −0.7 | −0.21 | −0.18 | −0.03 | −0.01 | 0.72 | 0.74 | 1 | 0.98 | 0.76 | 0.76 | 0.77 |
| FRGRAS | −0.72 | −0.25 | −0.21 | −0.08 | −0.03 | 0.72 | 0.73 | 0.98 | 1 | 0.75 | 0.75 | 0.77 |
| GR34VLFR | −0.81 | −0.15 | −0.14 | −0.06 | −0.07 | 0.9 | 1 | 0.74 | 0.73 | 0.82 | 0.83 | 0.83 |
| G2CGM | −0.76 | −0.11 | −0.09 | −0.1 | −0.02 | 0.86 | 0.82 | 0.76 | 0.75 | 1 | 0.93 | 0.93 |
| GR34DCFR | −0.78 | −0.17 | −0.14 | −0.12 | −0.09 | 0.81 | 0.83 | 0.76 | 0.75 | 0.93 | 1 | 0.95 |
| GR23DCFR | −0.84 | −0.26 | −0.22 | −0.22 | −0.13 | 0.81 | 0.83 | 0.77 | 0.77 | 0.93 | 0.95 | 1 |

Figure VII).1: Corrélations entre variables du jeu de données

```
mod<-glm(TMP89~.,data=TMP[is.na(TMP$TMP89)==F,c(1,3:41)])
summary(mod)
```

```
%
% Call:
% glm(formula = TMP89 ~ ., data = TMP[is.na(TMP$TMP89) == F, c(1,
%     3:41)])
%
% Coefficients: (6 not defined because of singularities)
%                Estimate Std. Error t value Pr(>|t|)
% (Intercept)  2.159e-12        NaN     NaN      NaN
% NUMORD      -1.039e-14        NaN     NaN      NaN
% ABATT1       3.537e-12        NaN     NaN      NaN
% CC           4.846e-14        NaN     NaN      NaN
```

```
% GGENE1       2.737e-13        NaN     NaN      NaN
% GGENE2      -1.475e-13        NaN     NaN      NaN
% GGENE3       9.109e-14        NaN     NaN      NaN
% SEXECH1      1.666e-13        NaN     NaN      NaN
% CHGRAS       5.999e-14        NaN     NaN      NaN
% CHMUSCLE     5.082e-14        NaN     NaN      NaN
% SEXCGM1             NA         NA      NA       NA
% G1CGM       -1.536e-14        NaN     NaN      NaN
% G2CGM       -1.407e-14        NaN     NaN      NaN
% M2CGM       -9.701e-15        NaN     NaN      NaN
% DATEFR1      3.572e-13        NaN     NaN      NaN
% DATEFR2     -5.723e-12        NaN     NaN      NaN
% DATEFR3      3.517e-13        NaN     NaN      NaN
% DATEFR4     -5.199e-12        NaN     NaN      NaN
% DATEFR5      3.458e-13        NaN     NaN      NaN
% DATEFR6     -5.124e-12        NaN     NaN      NaN
% DATEFR7      8.271e-13        NaN     NaN      NaN
% DATEFR8      4.116e-12        NaN     NaN      NaN
% DATEFR9     -4.643e-13        NaN     NaN      NaN
% DATEFR10     3.926e-12        NaN     NaN      NaN
% DATEFR11     4.303e-12        NaN     NaN      NaN
% DATEFR12     6.480e-13        NaN     NaN      NaN
% DATEFR13    -4.916e-12        NaN     NaN      NaN
% DATEFR14     1.234e-12        NaN     NaN      NaN
% DATEFR15    -4.569e-12        NaN     NaN      NaN
% DATEFR16     1.049e-12        NaN     NaN      NaN
% DATEFR17    -4.050e-12        NaN     NaN      NaN
% DATEFR18     8.797e-13        NaN     NaN      NaN
% DATEFR19     3.506e-12        NaN     NaN      NaN
% DATEFR20     3.440e-12        NaN     NaN      NaN
% DATEFR21     3.105e-12        NaN     NaN      NaN
% DATEFR22    -8.418e-13        NaN     NaN      NaN
% DATEFR23     3.465e-12        NaN     NaN      NaN
% DATEFR24    -4.167e-13        NaN     NaN      NaN
% DATEFR25            NA         NA      NA       NA
% SEXE1       -2.106e-13        NaN     NaN      NaN
```

```
% FRGRAS      -4.851e-14        NaN     NaN      NaN
% FRMUSCLE    -5.886e-14        NaN     NaN      NaN
% GR34VLFR    -3.358e-14        NaN     NaN      NaN
% GR23DCFR     1.750e-14        NaN     NaN      NaN
% GR34DCFR     9.918e-14        NaN     NaN      NaN
% GR34DCPAFR  -9.860e-14        NaN     NaN      NaN
% MU23DCFR     2.549e-14        NaN     NaN      NaN
% MU34DCFR     2.132e-14        NaN     NaN      NaN
% MU34DCPAFR  -4.050e-14        NaN     NaN      NaN
% FILPIECE    -5.401e-16        NaN     NaN      NaN
% LONPIECE     4.321e-15        NaN     NaN      NaN
% LONGEX      -4.399e-15        NaN     NaN      NaN
% EPAPIECE    -3.359e-15        NaN     NaN      NaN
% EPAGEX       3.728e-15        NaN     NaN      NaN
% JAMPIECE    -1.397e-15        NaN     NaN      NaN
% JAMGEX       1.381e-15        NaN     NaN      NaN
% LONMUGIOS   -4.189e-15        NaN     NaN      NaN
% EPAMUGIOS    2.988e-15        NaN     NaN      NaN
% JAMMUGIOS    1.191e-15        NaN     NaN      NaN
% LONPERTPAR          NA         NA      NA       NA
% EPAPERTPAR          NA         NA      NA       NA
% JAMPERTPAR          NA         NA      NA       NA
% TMUS3P      -1.710e-14        NaN     NaN      NaN
% TMP90        9.889e-01        NaN     NaN      NaN
% TVM         -2.391e-14        NaN     NaN      NaN
% DEN                 NA         NA      NA       NA
%
% (Dispersion parameter for gaussian family taken to be NaN)
%
%     Null deviance: 7.0543e+02  on 59  degrees of freedom
% Residual deviance: 8.6030e-26  on  0  degrees of freedom
% AIC: -3416.3
%
% Number of Fisher Scoring iterations: 1
```

# VIII) Rstudio

## a quelques raccourcies

## b Rstudio

Quelques raccourcis intéressants : - *ALT* + - : écrit de lisgne d'assignation **<-** - *CTRL* + *MAJ* + *M* écrira le signe du pipe *%>%* - *CTRL* + *MAJ* + *R* vous permettra d'écrire proprement un titre de nouvelle section - *CTRL* + *ALT* + *I* insérera un code chunk R dans votre code Rmarkdown - *CTRL* + *ALT* + *X* : alors celui-là est très intéressant. Si vous avez un bout de code que vous souhaitez transformer en fonction, ce raccourci-clavier fera tout le boulot tout seul, jusqu'à deviner le nom des paramètres de la fonction, vous n'aurez qu'à entrer le nom de la fonction. - *ALT* + *L* réduit la section dans laquelle est le curseur - *ALT* + *MAJ* + *L* ouvre la section - *ALT* + *O* réduit toutes les sections - *ALT* + *MAJ* + *O* ouvre toutes les sections - *CTRL* + *I* indente correctement le code sélectionné - *CTRL* + *MAJ* + *C* commente ou dé-commente la ligne active ou les ligne

Fonction cut -> Permet de transformet une quanti en quali avec des intervalles données

Fonction smartbind de chez gtools -> faire du rbind avec des colonnes mal agencées, permet du rbind selon le nom des colonnes.

Fonction droplevels -> Fait disparaître les niveaux de facteurs non utilisés.

# Bibliographie

Hutton, J.L., 2000. Number Needed to Treat: Properties and Problems. Journal of the Royal Statistical Society. Series A (Statistics in Society) 163, 403–419.