Modèles de régression

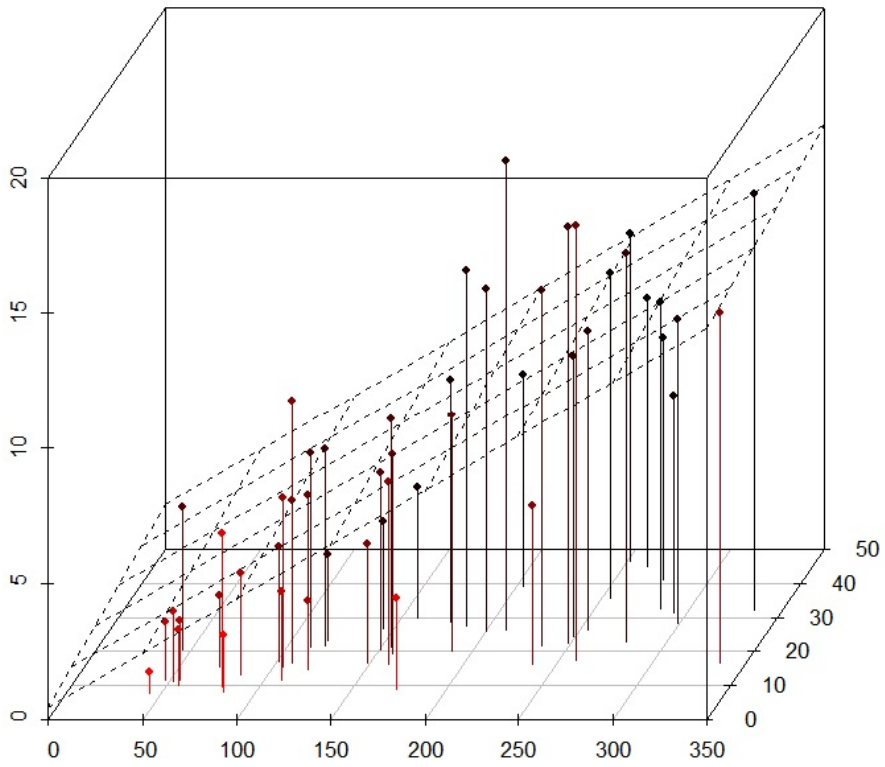
Christophe Chesneau

Université de Caen

Modèles de régression

Christophe Chesneau

http://www.math.unicaen.fr/~chesneau/



Caen, le 04 Octobre 2016

Table des matières

1 Régression linéaire multiple (*rlm*) 9

1.1 Contexte . . . 9

1.2 Estimations . . . 11

1.3 Coefficients de détermination . . . 13

1.4 Lois des estimateurs . . . 13

1.5 Intervalles de confiance . . . 14

1.6 Tests statistiques . . . 15

2 Études des hypothèses standards 17

2.1 Motivation . . . 17

2.2 Analyses du/des nuages de points . . . 17

2.3 Analyses graphiques des résidus . . . 19

2.4 Outils de vérification . . . 20

2.4.1 Indépendance de ϵ et $X_1, \ldots, X_p$ . . . 21

2.4.2 Indépendance de $\epsilon_1, \ldots, \epsilon_n$ . . . 22

2.4.3 $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$ . . . 24

2.4.4 Normalité de $\epsilon_1, \ldots, \epsilon_n$ . . . 25

2.5 Une solution possible : transformer Y . . . 27

3 Autres aspects du modèle de *rlm* 29

3.1 Détection des valeurs anormales . . . 29

3.2 Multicolinéarité . . . 31

3.3 Stabilité du modèle . . . 34

3.4 Sélection de variables . . . 35

3.5 Traitement de variables qualitatives . . . 39

4 Méthode des moindres carrés généralisés (*mcg*) 43

4.1 Contexte . . . 43

4.2 Quelques résultats . . . 44

4.3 Hétéroscédasticité des erreurs et *mcg* . . . 45
4.4 Cas de données groupées . . . 46
4.5 Méthode des *mcqg* . . . 48
4.6 Autocorrélation des erreurs et *mcg* . . . 51

5 Régression non-linéaire **57**
5.1 Contexte . . . 57
5.2 Régression polynomiale . . . 58
5.3 Résidus partiels . . . 59
5.4 Méthodes itératives . . . 62
5.5 Extension : régression non-paramétrique . . . 65

6 Régression logistique **69**
6.1 Contexte . . . 69
6.2 Transformation logit . . . 70
6.3 Variable latente . . . 71
6.4 Estimation . . . 73
6.5 Significativité de la régression . . . 75
6.6 Rapport des côtes . . . 77
6.7 Intervalles de confiance . . . 78
6.8 Pertinence du modèle . . . 79
6.9 Détection des valeurs anormales . . . 81
6.10 Sélection de variables . . . 83
6.11 Qualité du modèle . . . 84
6.12 Cas des données groupées . . . 86

7 Régression polytomique **89**
7.1 Contexte . . . 89
7.2 Régression multinomiale (ou polytomique non-ordonnée) . . . 90
7.2.1 Contexte . . . 90

7.2.2 Estimation 90
7.2.3 Significativité du modèle 91
7.2.4 Sélection de variables 93
7.2.5 Qualité du modèle 93
7.3 Régression polytomique ordonnée 94

8 Régression de Poisson **97**
8.1 Contexte 97
8.2 Significativité de la régression 99
8.3 Intervalles de confiance 101
8.4 Pertinence du modèle 101
8.5 Détection des valeurs anormales 104
8.6 Sélection de variables 105
8.7 Dispersion anormale 106
8.8 Variable de décalage (*offset*) 107

9 Modèles de régression à effets mixtes **111**
9.1 Introduction aux modèles de *rlm* à effets mixtes 111
9.2 Compléments et extensions 114

10 Jeux de données **117**

11 Annexe : *emv* **119**
11.1 Méthode 119
11.2 Résultats asymptotiques 120
11.3 Test global 120
11.4 Test partiel 120
11.5 Algorithme de Newton-Raphson et *emv* 121

Index **123**

~ Note ~

Ce document résume les notions abordées dans le cours *Modèles de Régression* du M2 orienté statistique de l'université de Caen.

Un des objectifs est de donner des pistes de réflexion à la construction de modèles prédictifs à partir de données.

Les méthodes statistiques y sont décrites de manière concise, avec les commandes R associées.

Pour compléter, des études utilisant des modèles de régression sont disponibles ici :

`http://www.math.unicaen.fr/~chesneau/etudes-reg.pdf`

Quatre points au centre de ce document :

- **Tous les modèles sont faux**
- **Certains modèles sont meilleurs que d'autres**
- **Le modèle le meilleur ne peut jamais être connu avec certitude**
- **Plus simple est le modèle, mieux c'est (principe KISS)**

Je vous invite à me contacter pour tout commentaire : `christophe.chesneau@gmail.com`

Quelques ressources en lien avec ce cours :

- Cours de Régression M2 de Bernard Delyon :

  `https://perso.univ-rennes1.fr/bernard.delyon/regression.pdf`

- Pratique de la régression linéaire multiple de Ricco Rakotomalala :

  `http://eric.univ-lyon2.fr/~ricco/cours/cours/La_regression_dans_la_pratique.pdf`

- Cours de Régression linéaire de Arnaud Guyader :

  `http://www.lsta.lab.upmc.fr/modules/resources/download/labsta/Pages/Guyader/Regression.pdf`

- CookBook R de Vincent Isoz et Daname Kolani :

  `http://www.sciences.ch/dwnldbl/divers/R.pdf`

Bonne lecture !

1 Régression linéaire multiple (*rlm*)

1.1 Contexte

Problématique : Dans une population, on souhaite prévoir les valeurs d'une variable quantitative Y à partir des valeurs de p autres variables $X_1, \ldots, X_p$. Cela revient à expliquer les variations de Y à partir de celles de $X_1, \ldots, X_p$. On dit alors que l'on souhaite "expliquer Y à partir de $X_1, \ldots, X_p$", Y est appelée "variable à expliquer" et $X_1, \ldots, X_p$ sont appelées "variables explicatives".

Données : Les données dont on dispose sont n observations de $(Y, X_1, \ldots, X_p)$ notées $(y_1, x_{1,1}, \ldots, x_{p,1}), \ldots, (y_n, x_{1,n}, \ldots, x_{p,n})$.

Les données se présentent généralement sous la forme d'un tableau :

Y	X_1	$\ldots$	X_p
y_1	$x_{1,1}$	$\ldots$	$x_{p,1}$
y_2	$x_{1,2}$	$\cdots$	$x_{p,2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y_n	$x_{1,n}$	$\ldots$	$x_{p,n}$

Modèle de régression linéaire multiple (*rlm*) : Si une liaison linéaire entre Y et $X_1, \ldots, X_p$ est envisageable, on peut considérer le modèle de régression linéaire multiple : il existe $p+1$ coefficients inconnus $\beta_0, \ldots, \beta_p$ tels que

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \epsilon,$$

où ϵ est une quantité représentant une somme d'erreurs.

On modélise Y, $X1$, $X2$ et $X3$ par une *rlm* en faisant :

```
reg = lm(Y ~ X1 + X2 + X3)
```

Objectif : Un objectif est d'estimer $\beta_0, \dots, \beta_p$ à l'aide des données afin de prédire la valeur moyenne de Y pour une nouvelle valeur de $(X_1, \dots, X_p)$.

Modélisation : On modélise les variables considérées comme des variables aléatoires réelles (*var*) (définies sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$). À partir de celles-ci, le modèle de *rlm* est caractérisé par les points suivants. Pour tout $i \in \{1, \dots, n\}$,

- $(x_{1,i}, \dots, x_{p,i})$ est une réalisation du vecteur aléatoire réel $(X_1, \dots, X_p)$,
- sachant que $(X_1, \dots, X_p) = (x_{1,i}, \dots, x_{p,i})$, y_i est une réalisation de

$$Y_i = \beta_0 + \beta_1 x_{1,i} + \dots + \beta_p x_{p,i} + \epsilon_i,$$

où ϵ_i est une *var* modélisant une somme d'erreurs.

On appelle erreurs les *var* $\epsilon_i, \dots, \epsilon_n$.

Remarque : Pour tout $x = (x_1, \dots, x_p) \in \mathbb{R}^p$, sous l'hypothèse que $\mathbb{E}(\epsilon | \{(X_1, \dots, X_p) = x\}) = 0$, le modèle de *rlm* peut s'écrire comme

$$\mathbb{E}(Y | \{(X_1, \dots, X_p) = x\}) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p.$$

Ainsi, sachant que $(X_1, \dots, X_p) = x$, la valeur moyenne de Y est une combinaison linéaire de $(x_1, \dots, x_p)$.

Écriture matricielle : Le modèle de *rlm* s'écrit sous la forme matricielle :

$$Y = X\beta + \epsilon,$$

où

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix}, \quad X = \begin{pmatrix} 1 & x_{1,1} & \cdots & x_{p,1} \\ 1 & x_{1,2} & \cdots & x_{p,2} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{1,n} & \cdots & x_{p,n} \end{pmatrix}, \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \quad \epsilon = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}.$$

Hypothèses standards : On suppose que X est de rang colonnes plein (donc $(X^t X)^{-1}$ existe), ϵ et $X_1, \dots, X_p$ sont indépendantes et $\epsilon \sim \mathcal{N}_n(0_n, \sigma^2 \mathbb{I}_n)$ où $\sigma > 0$ est un paramètre inconnu.

En particulier, cette dernière hypothèse entraîne que

- $\epsilon_1, \ldots, \epsilon_n$ sont indépendantes,
- $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n) = \sigma^2$,
- $\epsilon_1, \ldots, \epsilon_n$ suivent chacune une loi normale (qui est $\mathcal{N}(0, \sigma^2)$).

1.2 Estimations

Emco : L'estimateur des moindres carrés ordinaires (*emco*) de β est

$$\widehat{\beta} = (X^t X)^{-1} X^t Y.$$

Il est construit de sorte que l'erreur d'estimation entre $X\widehat{\beta}$ et Y soit la plus petite possible au sens $\|.\|^2$:

$$\widehat{\beta} = \underset{\beta \in \mathbb{R}^{p+1}}{\text{Argmin}} \|Y - X\beta\|^2,$$

où $\|.\|$ désigne la norme euclidienne de $\mathbb{R}^n$:

$$< a, b >= a^t b = b^t a = \sum_{i=1}^{n} a_i b_i, \qquad \|a\|^2 =< a, a >= a^t a = \sum_{i=1}^{n} a_i^2.$$

Pour tout $j \in \{0, \ldots, p\}$, la $j+1$-ème composante de $\widehat{\beta}$, notée $\widehat{\beta}_j$, est l'*emco* de β_j.

***Emco* et *emv* :** L'*emco* de β est l'estimateur du maximum de vraisemblance (*emv*) de β.

En effet, la vraisemblance associée à $(Y_1, \ldots, Y_n)$ est

$$L(\beta, z) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{\|z - X\beta\|^2}{2\sigma^2}\right), \qquad z \in \mathbb{R}^n.$$

Par conséquent

$$\underset{\beta \in \mathbb{R}^{p+1}}{\text{Argmax}}\, L(\beta, Y) = \underset{\beta \in \mathbb{R}^{p+1}}{\text{Argmin}} \|Y - X\beta\|^2 = \widehat{\beta}.$$

Estimateur de la prédiction : En posant $x_\bullet = (1, x_1, \ldots, x_p)$, la valeur prédite moyenne de Y lorsque $(X_1, \ldots, X_p) = (x_1, \ldots, x_p) = x$ est définie par

$$y_x = x_\bullet \beta = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p.$$

Un estimateur de y_x est

$$\widehat{Y}_x = x_\bullet \widehat{\beta} = \widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_p x_p.$$

Estimateur de σ^2 : Un estimateur de σ^2 est

$$\widehat{\sigma}^2 = \frac{1}{n-(p+1)} \|Y - X\widehat{\beta}\|^2.$$

Il vérifie $\mathbb{E}(\widehat{\sigma}^2) = \sigma^2$. De plus, $\widehat{\sigma}^2$ et $\widehat{\beta}$ sont indépendants.

Estimations ponctuelles : En pratique, on considère les réalisations de $\widehat{\beta}_0, \widehat{\beta}_1, \ldots, \widehat{\beta}_p, \widehat{\sigma}^2$ correspondantes aux données. On travaille donc avec des réels.

```
reg = lm(Y ~ X1 + X2 + X3)
```

On obtient les estimations ponctuelles de β_0, β_1, β_2 et β_3 par la commande R :

```
reg
```

Pour isoler l'estimation ponctuelle de β_2 (par exemple), on exécute :

```
reg$coeff[3]
```

Les valeurs prédites moyennes de Y prises aux valeurs des données de $X1$, $X2$ et $X3$ s'obtiennent en faisant :

`predict(reg)` (ou `fitted(reg)`)

La valeur prédite moyenne de Y pour la valeur $(X1, X2, X3) = (1.2, 2.2, 6)$ est donnée par les commandes R :

```
predict(reg, data.frame(X1 = 1.2, X2 = 2.2, X3 = 6))
```

Si le coefficient β_0 n'a pas de sens dans la modélisation, on l'enlève en faisant :

```
reg = lm(Y ~ X1 + X2 + X3 - 1)
```

1.3 Coefficients de détermination

Coefficients de détermination : On appelle coefficient de détermination la réalisation R^2 de

$$\widehat{R}^2 = 1 - \frac{\|\widehat{Y} - Y\|^2}{\|\overline{Y}1_n - Y\|^2},$$

où $\widehat{Y} = X\widehat{\beta}$ et $\overline{Y} = (1/n)\sum_{i=1}^n Y_i$ (et 1_n désigne le vecteur colonne à n composantes égales à 1).

Ce R^2 est un coefficient réel toujours compris entre 0 et 1.

Il mesure de la qualité du modèle de *rlm*; plus R^2 est proche de 1, meilleur est le modèle.

Comme le R^2 dépend fortement de p, on ne peut pas l'utiliser pour comparer la qualité de 2 modèles de *rlm* qui diffèrent quant au nombre de variables explicatives. C'est pourquoi on lui préfère sa version ajustée présentée ci-dessous.

Coefficients de détermination ajusté : On appelle coefficient de détermination ajusté la réalisation $\overline{R}^2$ de

$$\widehat{\overline{R}}^2 = 1 - \frac{\|\widehat{Y} - Y\|^2/(n-(p+1))}{\|\overline{Y}1_n - Y\|^2/(n-1)} = 1 - \frac{n-1}{n-(p+1)}(1-\widehat{R}^2).$$

Les coefficients R^2 et $\overline{R}^2$ sont donnés par les commandes R :

```
summary(reg)
```

1.4 Lois des estimateurs

Loi de $\widehat{\beta}$: On a

$$\widehat{\beta} \sim \mathcal{N}_{p+1}\left(\beta, \sigma^2(X^tX)^{-1}\right).$$

La matrice de covariance estimée de $\widehat{\beta}$, qui est aussi la réalisation de $\widehat{\sigma}^2(X^tX)^{-1}$, est donnée par les commandes R :

```
vcov(reg)
```

Loi de $\widehat{\beta}_j$: Pour tout $j \in \{0, \dots, p\}$, en notant $[(X^tX)^{-1}]_{j+1,j+1}$ la $j+1$-ème composante diagonale de $(X^tX)^{-1}$, on a

$$\widehat{\beta}_j \sim \mathcal{N}\left(\beta_j, \sigma^2[(X^tX)^{-1}]_{j+1,j+1}\right), \qquad \frac{\widehat{\beta}_j - \beta_j}{\sigma\sqrt{[(X^tX)^{-1}]_{j+1,j+1}}} \sim \mathcal{N}(0,1).$$

Degrés de liberté : Dans ce qui suit, on travaillera avec le nombre de degrés de liberté :

$\nu = n - (p+1)$.

Loi associée à $\widehat{\sigma}^2$: On a

$$(n-(p+1))\frac{\widehat{\sigma}^2}{\sigma^2} \sim \chi^2(\nu).$$

Apparition de la loi de Student : Pour tout $j \in \{0, \dots, p\}$, en posant $\widehat{\sigma}(\widehat{\beta}_j) = \widehat{\sigma}\sqrt{[(X^tX)^{-1}]_{j+1,j+1}}$, on a

$$\frac{\widehat{\beta}_j - \beta_j}{\widehat{\sigma}(\widehat{\beta}_j)} \sim \mathcal{T}(\nu).$$

1.5 Intervalles de confiance

Intervalle de confiance pour β_j : Pour tout $j \in \{0, \dots, p\}$, un intervalle de confiance pour β_j au niveau $100(1-\alpha)\%$, $\alpha \in]0,1[$, est la réalisation i_{β_j} de

$$I_{\beta_j} = \left[\widehat{\beta}_j - t_\alpha(\nu)\widehat{\sigma}\sqrt{[(X^tX)^{-1}]_{j+1,j+1}},\ \widehat{\beta}_j + t_\alpha(\nu)\widehat{\sigma}\sqrt{[(X^tX)^{-1}]_{j+1,j+1}}\right],$$

où $t_\alpha(\nu)$ est le réel vérifiant $\mathbb{P}(|T| \geq t_\alpha(\nu)) = \alpha$, avec $T \sim \mathcal{T}(\nu)$.

```
confint(reg, level = 0.95)
```

Les estimateurs $\widehat{\beta}_0, \dots, \widehat{\beta}_p$ étant corrélés, on peut aussi s'intéresser aux ellisoïdes de confiance pour un couple (β_i, β_j). Pour (β_2, β_4) par exemple, les commandes R associées sont :

```
library(ellipse)
plot(ellipse(reg, c(3, 5), level = 0.95))
```

Intervalle de confiance pour y_x : Soient y_x la prédiction moyenne de Y quand $(X_1, \dots, X_p) = (x_1, \dots, x_p) = x$ et $x_\bullet = (1, x_1, \dots, x_p)$.

Un intervalle de confiance pour y_x au niveau $100(1-\alpha)\%$, $\alpha \in]0,1[$, est la réalisation i_{y_x} de

$$I_{y_x} = \left[\widehat{Y}_x - t_\alpha(\nu)\widehat{\sigma}\sqrt{x_\bullet (X^tX)^{-1}x_\bullet^t},\ \widehat{Y}_x + t_\alpha(\nu)\widehat{\sigma}\sqrt{x_\bullet (X^tX)^{-1}x_\bullet^t}\right],$$

où $t_\alpha(\nu)$ est le réel vérifiant $\mathbb{P}(|T| \geq t_\alpha(\nu)) = \alpha$, avec $T \sim \mathcal{T}(\nu)$.

```
predict(reg, data.frame(X1 = 1.2, X2 = 2.2, X3 = 6),
interval = "confidence")
```

1.6 Tests statistiques

p-valeur : On considère des hypothèses de la forme :

$$H_0 : \text{A} \qquad \text{contre} \qquad H_1 : \text{contraire de A}$$

La p-valeur est le plus petit réel $\alpha \in]0,1[$ calculé à partir des données tel que l'on puisse se permettre de rejeter H_0 au risque $100\alpha\%$. Autrement écrit, la p-valeur est une estimation ponctuelle de la probabilité critique de se tromper en rejetant H_0/affirmant H_1 alors que H_0 est vraie.

Degrés de significativité : Le rejet de H_0 sera

- "significatif" si p-valeur $\in]0.01, 0.05]$, symbolisé par $*$,
- "très significatif" si p-valeur $\in]0.001, 0.01]$, symbolisé par $**$,
- "hautement significatif" si p-valeur < 0.001, symbolisé par $***$,
- ("presque significatif" si p-valeur $\in]0.05, 0.1]$, symbolisé par . (un point)).

Test de Student : Soit $j \in \{0, \ldots, p\}$. L'objectif du test de Student est d'évaluer l'influence de X_j sur Y.

On considère les hypothèses :

$$H_0 : \beta_j = 0 \qquad \text{contre} \qquad H_1 : \beta_j \neq 0.$$

On calcule la réalisation t_{obs} de

$$T_* = \frac{\widehat{\beta}_j}{\widehat{\sigma}(\widehat{\beta}_j)}.$$

On considère une *var* $T \sim \mathcal{T}(\nu)$.

Alors la p-valeur associée est

$$\text{p-valeur} = \mathbb{P}(|T| \geq |t_{obs}|).$$

Si

- $*$, l'influence de X_j sur Y est "significative",
- $**$, l'influence de X_j sur Y est "très significative",
- $***$, l'influence de X_j sur Y est "hautement significative".

Test global de Fisher : L'objectif du test global de Fisher est d'étudier la pertinence du lien linéaire entre Y et $X_1, \ldots, X_p$.

On considère les hypothèses :

$$H_0 : \beta_1 = \beta_2 = \ldots = \beta_p = 0 \qquad \text{contre} \qquad H_1 : \text{il y a au moins un coefficient non nul.}$$

On calcule la réalisation f_{obs} de

$$F_* = \frac{\widehat{R}^2}{1-\widehat{R}^2}\frac{n-(p+1)}{p}.$$

On considère une *var* $F \sim \mathcal{F}(p, \nu)$.

Alors la p-valeur associée est

$$\text{p-valeur} = \mathbb{P}(F \geq f_{obs}).$$

Notons que ce test est moins précis que le test de Student car il ne précise pas quels sont les coefficients non nuls.

Il est toutefois un indicateur utile pour déceler d'éventuelles problèmes (comme des colinéarités entre $X_1, \ldots, X_p$).

Les tests statistiques précédents sont mis en œuvre par les commandes R :

```
summary(reg)
```

2 Études des hypothèses standards

2.1 Motivation

Questions :

1. Comment valider les hypothèses standard du modèle de *rlm* avec les données ?
2. Peut-on améliorer les estimations des paramètres ?

Rappel : Les hypothèses suivantes ont été formulées :

- ϵ et $X_1, \ldots, X_p$ sont indépendantes,
- $\epsilon_1, \ldots, \epsilon_n$ sont indépendantes,
- $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$,
- $\epsilon_1, \ldots, \epsilon_n$ suivent des lois normale centrées.

Commandes R clés : Une première analyse de la validation de ces hypothèses doit être graphique.

Les commandes R clés sont :

```
par(mfrow = c(2, 2))
plot(reg, 1:4)
```

L'enjeu des graphiques affichés sera expliqué par la suite.

Des tests statistiques rigoureux viendront ensuite confirmer/infirmer cette première analyse visuelle.

2.2 Analyses du/des nuages de points

Pertinence du modèle : Pour certain problème, le modèle de *rlm* n'est pas le plus adapté.

Il est parfois judicieux de transformer Y et $X_1, \ldots, X_p$, puis de les modéliser par une *rlm*.

Ainsi, on considère un modèle de la forme :

$$f(Y) = \beta_0 + \beta_1 g_1(X_1) + \ldots + \beta_p g_p(X_p) + \epsilon,$$

où $f, g_1, \ldots, g_p$ désignent des transformations/fonctions à choisir.

Choix des transformations : Les p nuages de points :

$$\{(x_{j,i}, y_i);\ i \in \{1, \ldots, n\}\}, \qquad j \in \{1, \ldots, p\}$$

peuvent nous aiguiller sur les transformations candidates.

Pour tout $j \in \{1, \ldots, p\}$, une approche intuitive consiste à déterminer des fonctions f et g_j telles que le nuage de points $\{(g_j(x_{j,i}), f(y_i));\ i \in \{1, \ldots, n\}\}$ soit ajustable par une droite.

`plot(w)`

ou

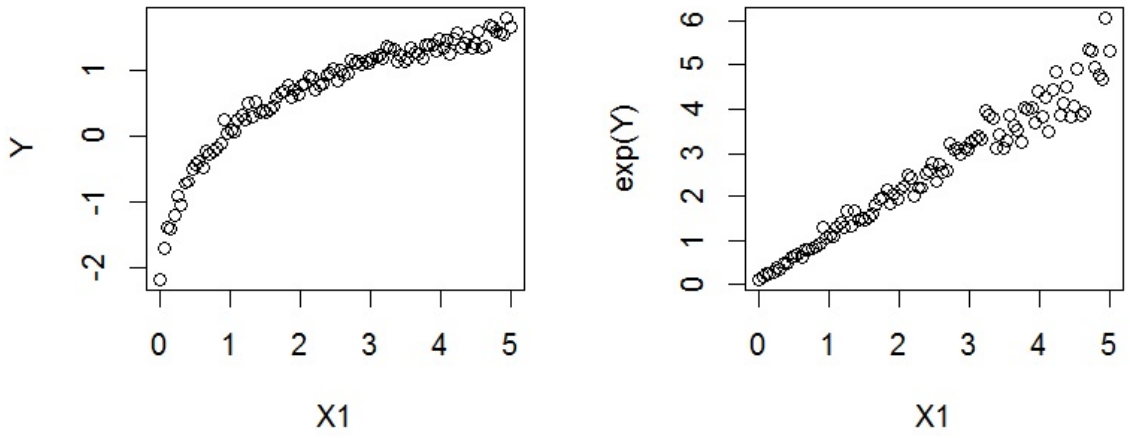
`pairs(w)` (ou, par exemple : `pairs(~ Y + X1 + X4)`).

Pour avoir des idées de fonctions $f, g_1, \ldots, g_p$ pertinentes, on peut utiliser les commandes R :

`scatterplotMatrix(w)` (ou, par exemple : `scatterplotMatrix(~ Y + X1 + X4)`)

Nous verrons par la suite les limites de cette approche et étudierons les méthodes alternatives (dans le chapitre *Régression non-linéaire*).

Exemple : Dans l'exemple-ci dessous, on cherche à expliquer Y à partir de X_1 :



Vu le nuage de points, il est préférable de considérer la transformation $\exp(Y)$ et de faire une régression linéaire sur X_1, soit

$$\exp(Y) = \beta_0 + \beta_1 X_1 + \epsilon.$$

On obtiendra des estimations de β_0 et β_1 avec un meilleur $\overline{R}^2$.

Un exemple de *rlm* avec variables transformées est

```
reg = lm(log(Y) ~ sqrt(X1) + exp(X2) + I(X3^4))
```

2.3 Analyses graphiques des résidus

Résidus : Pour tout $i \in \{1, \ldots, n\}$, on appelle i-ème résidu la réalisation e_i de

$$\widehat{\epsilon}_i = Y_i - \widehat{Y}_i,$$

où $\widehat{Y}_i = \widehat{\beta}_0 + \widehat{\beta}_1 x_{1,i} + \ldots + \widehat{\beta}_p x_{p,i}$.

On appelle résidus les réels $e_1, \ldots, e_n$.

Ces résidus vont nous permettre de valider ou non les hypothèses initiales.

```
residuals(reg)
```

Résidus standardisés : Pour tout $i \in \{1, \ldots, n\}$, on appelle i-ème résidu standardisé la réalisation e_i^* de

$$\widehat{\epsilon}_i^* = \frac{\widehat{\epsilon}_i}{\widehat{\sigma}\sqrt{1 - [X(X^tX)^{-1}X^t]_{i,i}}}.$$

On appelle résidus standardisés les réels $e_1^*, \ldots, e_n^*$.

```
rstandard(reg)
```

Lois : Pour tout $i \in \{1, \ldots, n\}$, si les hypothèses initiales sont vérifiées, on a

$$\widehat{\epsilon}_i \sim \mathcal{N}\left(0, \sigma^2(1 - [X(X^tX)^{-1}X^t]_{i,i})\right), \qquad \frac{\widehat{\epsilon}_i}{\sqrt{1 - [X(X^tX)^{-1}X^t]_{i,i}}} \sim \mathcal{N}(0, \sigma^2)$$

et

$$\widehat{\epsilon}_i^* = \frac{\widehat{\epsilon}_i}{\widehat{\sigma}\sqrt{1 - [X(X^tX)^{-1}X^t]_{i,i}}} \sim \mathcal{T}(\nu).$$

Analyse graphique principale : On trace le nuage de points : $\{(i, e_i);\ i \in \{1, \ldots, n\}\}$. Si

- le nuage de points n'a aucune structure particulière,
- il y a une symétrie dans la répartition des points par rapport à l'axe des abscisses,

alors on admet que $\epsilon \sim \mathcal{N}_n(0_n, \sigma^2\mathbb{I}_n)$.

```
plot(residuals(reg))
```

Si problème :

1. Si le nuage de points a l'allure d'une route sinueuse ou d'un mégaphone, on soupçonne que ϵ et $X_1, \ldots, X_p$ sont dépendantes ou/et les *var* $\epsilon_1, \ldots, \epsilon_n$ sont dépendantes (si cela a du sens), ou/et $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$ n'est pas vérifiée.
2. S'il y a une asymétrie dans la répartition des points par rapport à l'axe des abscisses, l'hypothèse de normalité de $\epsilon_1, \ldots, \epsilon_n$ est à étudier.

2.4 Outils de vérification

En cas de doute, il convient de vérifier, dans l'ordre :

- l'indépendance de ϵ et $X_1, \ldots, X_p$,
- l'indépendance de $\epsilon_1, \ldots, \epsilon_n$,
- l'égalité $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$,
- la normalité de $\epsilon_1, \ldots, \epsilon_n$.

2.4.1 Indépendance de ϵ et $X_1, \ldots, X_p$

Graphique "Scale-Location" : On trace le nuage de points :

$$\{(e_i, y_i - e_i);\ i \in \{1, \ldots, n\}\}.$$

Notons que $y_i - e_i$ est la réalisation de $\widehat{Y}_i = Y_i - \widehat{\epsilon}_i$.

Si on ne peut pas ajuster le nuage de points par une "ligne" (droite ou courbe), on admet que ϵ et $X_1, \ldots, X_p$ sont indépendantes.

```
plot(reg, 1)
```

Si problème : Si on peut ajuster le nuage de points par une "ligne" (droite ou courbe), on soupçonne que ϵ et $X_1, \ldots, X_p$ sont dépendantes. Le lien linéaire entre Y et $X_1, \ldots, X_p$ peut être remis en question.

Le modèle de régression non-linéaire est une alternative à étudier ; l'ajout de nouvelles variables dépendantes de $X_1, \ldots, X_p$, comme les transformations polynomiales X_1^2, $X_2^2 \ldots$, peut être plus approprié.

La "ligne rouge" affichée par les commandes `plot(reg, 1)` est un ajustement du nuage de points qui utilise une méthode non-linéaire avancée, appelée régression locale ou LOESS. Dès lors, cet ajustement doit être médiocre sinon cela traduit une dépendance (polynomiale) entre ϵ et $X_1, \ldots, X_p$. De plus, la moyenne des valeurs de la "ligne rouge" affichée doit être quasi nulle, caractérisant ainsi l'hypothèse $\mathbb{E}(\epsilon_1) = \ldots = \mathbb{E}(\epsilon_n) = 0$.

Test de Rainbow : Pour conclure à la non-linéarité du modèle de régression, on préconise le test de Rainbow : si p-valeur < 0.05, on rejette la linéairté du modèle de *rlm* et on admet qu'un modèle de régression non-linéaire est plus adapté aux données.

```
library(lmtest)

raintest(reg)
```

2.4.2 Indépendance de $\epsilon_1, \ldots, \epsilon_n$

Motivation : Si les observations de $Y, X_1, \ldots, X_p$ portent sur des individus tous différents et que le modèle de *rlm* a du sens, $\epsilon_1, \ldots, \epsilon_n$ sont indépendantes.

Par conséquent, si on distingue une structure dans le nuage des points des résidus (route sinueuse, mégaphone...),

- soit le modèle n'est pas adapté,
- soit il faut se tourner vers la vérification de l'hypothèse $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$.

En revanche, si les observations de $Y, X_1, \ldots, X_p$ présentent une dépendance temporelle, la dépendance de $\epsilon_1, \ldots, \epsilon_n$ est à étudier.

Corrélogramme : Pour étudier l'indépendance de $\epsilon_1, \ldots, \epsilon_n$, partant des résidus $e_1, \ldots, e_n$, la première approche consiste à tracer le corrélogramme.

Celui-ci représente les estimations ponctuelles de la fonction d'autocorrélation (*acf*) définie par

$$\rho(h) = \frac{\mathbb{C}(\epsilon_i, \epsilon_{i+h})}{\sigma(\epsilon_i)\sigma(\epsilon_{i+h})}, \qquad i \in \{1, \ldots, n-h\}, \qquad h \in \{1, \ldots, n-1\},$$

sous forme de bâtons.

La liaison linéaire entre ϵ_i et ϵ_{i+h} est mesurée.

On peut aussi calculer un intervalle de confiance pour $\rho(h)$ au delà duquel la dépendance est remise en cause.

Si les bâtons sont de tailles et de signes alternés (ou presque) et qu'aucun d'entre eux ne dépassent les bornes de l'intervalle de confiance (ou presque), on admet l'indépendance de $\epsilon_1, \ldots, \epsilon_n$.

```
acf(residuals(reg))
```

Corrélogramme partiel : Le corrélogramme partiel vient compléter l'étude précédente ; il représente les estimations ponctuelles de la fonction d'autocorrélation partielle (*pacf*) sous forme de bâtons. Cette fonction mesure la liaison linéaire entre ϵ_i et ϵ_{i+h} une fois retirés les liens transitant par les variables intermédiaires $\epsilon_{i+1}, \ldots, \epsilon_{i+h-1}$.

L'interprétation est la même que pour l'*acf*.

```
pacf(residuals(reg))
```

Si problème : Ainsi, si les sommets des bâtons peuvent être rejoints par une ligne courbe "sans pic" ou si plusieurs bâtons dépassent les bornes de l'intervalle de confiance, une dépendance peut-être soupçonnée.

Cela peut être confirmé avec le test de Ljung-Box.

Test de Ljung-Box (ou du portemanteau) : On considère les hypothèses :

$$H_0 : \rho(1) = \ldots = \rho(n) = 0 \qquad \text{contre} \qquad H_1 : \text{au moins une corrélation n'est pas nulle.}$$

Partant des résidus $e_1, \ldots, e_n$, on peut utiliser le test de Ljung-Box : si p-valeur < 0.05, on admet qu'au moins une corrélation n'est pas nulle, donc que $\epsilon_1, \ldots, \epsilon_n$ ne sont pas indépendantes.

```
library(lawstat)

Box.test(residuals(reg), type = "Ljung")
```

Structure de dépendance : Si la dépendance $\epsilon_1, \ldots, \epsilon_n$ est avérée ; le modèle de *rlm* n'est pas adapté.

Afin de trouver une alternative, il est intéressant d'identifier, si possible, la structure de dépendance associée.

La structure *AR(1)* présentée ci-après est l'une des plus répandue.

Structure *AR(1)* : On dit que $\epsilon_1, \ldots, \epsilon_n$ ont une structure auto-régressive de degré 1 (*AR(1)*) s'il existe :

- $\rho \in]-1, 1[$,
- n *var iid* $u_1, \ldots, u_n$ suivant chacune la loi normale $\mathcal{N}(0, v^2)$ avec $v > 0$,

tels que, pour tout $i \in \{1, \ldots, n\}$,

$$\epsilon_i = \rho \epsilon_{i-1} + u_i.$$

Le réel ρ mesure la dépendance de $\epsilon_1, \ldots, \epsilon_n$;

- si $\rho = 0$, pour tout $i \in \{1, \ldots, n\}$, $\epsilon_i = u_i$, donc $\epsilon_1, \ldots, \epsilon_n$ sont indépendants,
- si $\rho \neq 0$, on admet la structure *AR(1)* ; $\epsilon_1, \ldots, \epsilon_n$ ne sont pas indépendants.

Test de Durbin-Watson : On considère les hypothèses :

$$H_0 : \rho = 0 \qquad \text{contre} \qquad H_1 : \rho \neq 0.$$

Partant des résidus $e_1, \ldots, e_n$, on peut utiliser le test de Durbin-Watson : si p-valeur < 0.05, alors on admet que $\rho \neq 0$, entraînant la structure $AR(1)$ de $\epsilon_1, \ldots, \epsilon_n$.

```
library(lmtest)

dwtest(reg)
```

Si problème : Dans le cas d'une structure *AR(1)* sur $\epsilon_1, \ldots, \epsilon_n$, on est capable d'estimer efficacement β. Cela sera présenté dans le chapitre *Méthode des moindres carrés généralisés*.

2.4.3 $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$

Graphique "Scale-Location" : On considère le nuage de points :

$$\left\{ (\sqrt{|e_i^*|}, y_i - e_i);\ i \in \{1, \ldots, n\} \right\}.$$

Si on ne distingue aucune structure, on peut admettre que $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$.

```
plot(reg, 3)
```

Test de White / Test de Breusch-Pagan : Admettons que $\epsilon_1, \ldots, \epsilon_n$ soient indépendantes. Pour étudier l'égalité $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$, on peut utiliser le test de White ou le test de Breusch-Pagan.

L'idée du test de White est de tester l'existence d'un lien linéaire entre $\widehat{\epsilon}^2$ et les p^2 variables :

- $X_1, \ldots, X_p$,
- les carrés : $X_1^2, \ldots, X_p^2$,
- les produits croisés : $X_1X_2, X_1X_3, \ldots, X_{p-1}X_p$.

Si p-valeur > 0.05, on admet que $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$.

Plus simplement, on peut utiliser le test de Breusch-Pagan qui teste l'existence d'un lien linéaire entre $\widehat{\epsilon}^2$ et $X_1, \ldots, X_p$ seules.

```
library(lmtest)

bptest(reg)
```

Méthode de Glejser : La méthode de Glejser consiste à étudier l'existence d'un lien linéaire entre $|\widehat{\epsilon}|$ et des transformations (subjectives) de $X_1, \ldots, X_p$. Si au moins une variable influe très significativement sur $|\widehat{\epsilon}|$, on rejette $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$.

```
e = residuals(reg)

reg2 = lm(abs(e) ~ sqrt(X1) + X2 + log(X3))

summary(reg2)
```

Si problème : On propose 2 solutions :

- Une *rlm* avec Y transformée (comme $\ln(Y)$, $\sqrt{Y}$ ou $1/Y$) peut engendrer des nouvelles erreurs $\epsilon_1, \ldots, \epsilon_n$ vérifiant $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$.

  Dès lors, on peut utiliser ce nouveau modèle pour une étude statistique.
- Si, pour tout $i \in \{1, \ldots, n\}$, on a une idée de la valeur de $\mathbb{V}(\epsilon_i)$ ou que celle-ci est estimable, alors nous verrons une solution dans le chapitre *Méthode des moindres carrés généralisés*.

2.4.4 Normalité de $\epsilon_1, \ldots, \epsilon_n$

QQ plot : Admettons que $\epsilon_1, \ldots, \epsilon_n$ soient indépendantes et $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$.

Dans un premier temps, pour étudier la normalité de $\epsilon_1, \ldots, \epsilon_n$, on trace le nuage de points QQ plot associé (ou diagramme Quantile-Quantile).

Si le nuage de points est bien ajusté par la droite $y = x$, alors on admet la normalité de $\epsilon_1, \ldots, \epsilon_n$.

Principe du QQ plot : Le principe du QQ plot dans le cadre d'une *rlm* est décrit ci-après.

1. Pour tout $i \in \{1, \ldots, n\}$, si $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$, alors

$$\widehat{\epsilon}_i^* = \frac{\widehat{\epsilon}_i}{\widehat{\sigma}\sqrt{1 - [X(X^tX)^{-1}X^t]_{i,i}}} \sim \mathcal{T}(\nu).$$

   On considère alors la fonction de répartition F associée à la loi $\mathcal{T}(\nu)$.

2. D'autre part, un estimateur de la fonction de répartition de $\widehat{\epsilon}_1^*$ dans le cas général est la fonction de répartition empirique : $\widehat{G}(x) = (1/n)\sum_{i=1}^n \mathbf{1}_{\{\widehat{\epsilon}_i^* \leq x\}}$. Soit $G(x)$ sa réalisation.
3. Par conséquent, si $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$, alors on a $F(x) \simeq G(x)$ et, a fortiori, $x \simeq F^{-1}(G(x))$. Le graphique QQ plot consiste à tracer le nuage de points :

$$\left\{(F^{-1}(G(e_i^*)), e_i^*);\ i \in \{1, \ldots, n\}\right\}.$$

Si $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$, alors, pour tout $i \in \{1, \ldots, n\}$, $(F^{-1}(G(e_i^*)), e_i^*) \simeq (e_i^*, e_i^*)$ et les points du nuage seront presque sur la droite d'équation $y = x$.

Notons que l'on trace le QQ plot à l'aide des résidus standardisés $e_1^*, \ldots, e_n^*$ et de la loi de Student $\mathcal{T}(\nu)$.

```
plot(qt(ppoints(rstandard(reg)), reg$def), sort(rstandard(reg)))
```

Si ν est suffisament grand, on peut utiliser la loi normale $\mathcal{N}(0,1)$ car $\mathcal{T}(\nu) \approx \mathcal{N}(0,1)$. On parle alors de QQ norm.

```
plot(reg, 2)
```

soit encore :

```
qqnorm(rstandard(reg))
```

ou plus joli :

```
library(car)
qqPlot(reg)
```

Test de Shapiro-Wilk : Pour conclure à la normalité de $\epsilon_1, \ldots, \epsilon_n$, partant des résidus $e_1, \ldots, e_n$, on préconise le test de Shapiro-Wilk : si p-valeur > 0.05, on admet l'hypothèse de normalité.

```
shapiro.test(residuals(reg))
```

Si problème : Une *rlm* avec Y transformée (comme $\ln(Y)$, $\sqrt{Y}$ ou $1/Y$) peut engendrer des nouvelles erreurs $\epsilon_1, \ldots, \epsilon_n$ suivant chacune une loi normale.

Dès lors, on peut utiliser ce nouveau modèle pour une étude statistique.

2.5 Une solution possible : transformer Y

Bilan : Comme écrit précédemment, s'il y a un problème de normalité ou d'égalité de variances pour les erreurs, un nouveau modèle de *rlm* avec Y transformée peut engendrer des nouvelles erreurs $\epsilon_1, \ldots, \epsilon_n$ vérifiant les hypothèses standards.

Les transformations les plus utilisées sont : $\ln(Y)$, Y^2, Y^3, $\sqrt{Y}$ et $1/Y$.

Transformations avancées : On peut également utiliser des transformations qui s'adaptent aux données via un nouveau paramètre inconnu à estimer. Le schéma standard est décrit ci-après.

Soit T_λ une fonction dépendant d'un réel λ. On fait une *rlm* sur $T_\lambda(Y)$ et $X_1, \ldots, X_p$ et on estime λ de sorte que la loi de $\epsilon_1, \ldots, \epsilon_n$ soit aussi proche que possible d'une loi normale. Cette estimation se fait avec l'*emv* $\widehat{\lambda}$ de λ.

On considère alors un nouveau modèle de *rlm* :

$$T_{\hat{\lambda}}(Y) = \beta_0 + \beta_1 X_1 + \ldots \beta_p X_p + \epsilon,$$

où $\beta_0, \ldots, \beta_p$ sont $p+1$ coefficients inconnus et $\epsilon \sim \mathcal{N}(0, \sigma^2)$ avec σ^2 inconnu.

Les transformations les plus célèbres sont :

- la transformation logarithmique :

$$\ell_\alpha(Y) = \log(Y + \alpha).$$

Le "meilleur" réel α peut être estimé.

```
library(MASS)
reg = lm(Y ~ X1 + X2 + X3)
graph = logtrans(reg, alpha = seq(0.1, 10, length = 20))
hat_alpha = graph$x[which.max(graph$y)]
reg2 = lm(log(Y + hat_alpha) ~ X1 + X2 + X3)
summary(reg2)
```

- la transformation puissance de Box-Cox :

$$bc_\lambda(Y) = \begin{cases} \dfrac{Y^\lambda - 1}{\lambda} & \text{si } \lambda \neq 0, \\ \log(Y) & \text{sinon.} \end{cases}$$

(sous l'hypothèse que $Y > 0$).

Le "meilleur" réel λ peut être estimé.

```
library(car)
reg = lm(Y ~ X1 + X2 + X3)
reg2 = powerTransform(reg)
summary(reg2)
reg3 = lm(bcPower(Y, coef(reg2)) ~ X1 + X2 + X3)
summary(reg3)
```

- la transformation de Yeo et Johnson qui supporte le cas où Y peut prendre des valeurs négatives :

$$yj_\lambda(Y) = \begin{cases} \dfrac{(Y+1)^\lambda - 1}{\lambda} & \text{si } \lambda \neq 0 \text{ et } Y \geq 0, \\ \log(Y+1) & \text{si } \lambda = 0 \text{ et } Y \geq 0, \\ -\dfrac{(1-Y)^{2-\lambda} - 1}{2-\lambda} & \text{si } \lambda \neq 2 \text{ et } Y < 0, \\ -\log(1-Y) & \text{si } \lambda = 2 \text{ et } Y < 0. \end{cases}$$

Le "meilleur" réel λ peut être estimé.

```
library(car)
reg = lm(Y ~ X1 + X2 + X3)
reg2 = powerTransform(reg, family = "yjPower")
summary(reg2)
reg3 = lm(yjPower(Y, coef(reg2)) ~ X1 + X2 + X3)
summary(reg3)
```

3 Autres aspects du modèle de *rlm*

3.1 Détection des valeurs anormales

Objectif : La détection de valeurs anormales dans les données est cruciale car ces valeurs peuvent avoir une influence négative dans les estimations et, a fortiori, dans les prévisions (effet levier de la fonction de régression).

Méthodes :

- Méthode des résidus standardisés,
- Critère des distances de Cook.

Méthode des résidus standardisés : Pour tout $i \in \{1, \ldots, n\}$, si

$$|e_i^*| > 2,$$

on envisage l'anormalité de la i-ème observation.

Cette règle repose sur la construction d'un intervalle de confiance nous assurant qu'il y a (environ) 95 chances sur 100 que la i-ème observation vérifie $|e_i^*| \leq 2$.

```
e = rstandard(reg)
plot(e)
e[abs(e) > 2]
```

Critère des distances de Cook : Pour tout $i \in \{1, \ldots, n\}$, on défini la distance de Cook de la i-ème observation par

$$d_i = \frac{[X(X^tX)^{-1}X^t]_{i,i}}{(p+1)(1-[X(X^tX)^{-1}X^t]_{i,i})}(e_i^*)^2.$$

Si

$$d_i > 1,$$

on envisage l'anormalité de la i-ème observation.

Dans la littérature, les seuils $4/n$ et $4/(n-(p+1))$ sont parfois utilisés au lieu de 1.

On peut montrer que d_i est la réalisation de

$$D_i = \frac{\|\widehat{Y} - \widehat{Y}_{-i}\|_n^2}{(p+1)\widehat{\sigma}^2},$$

où $\widehat{Y}_{-i} = (X\widehat{\beta})_{-i}$ qui correspond au calcul de $X\widehat{\beta} = X(X^tX)^{-1}X^tY$ avec X et Y privés de la i-ème observation.

Ce critère mesure donc l'influence d'une observation sur l'erreur de prévision.

```
plot(reg, 4)
cooks.distance(reg)[cooks.distance(reg) > 1]
```

Admettons que les valeurs associées aux individus 4 et 26 soient anormales.

On refait l'analyse sans ces individus avec les commandes R :

```
reg2 = lm(Y ~ X1 + X2 + X3, subset = - c(4, 26))
```

Ou alors :

```
ww = w[ - c(4, 26), ]
attach(ww)
reg = lm(Y ~ X1 + X2 + X3)
```

Peu importe la méthode et le résultat, il faut toujours s'assurer auprès du spécialiste de l'étude qu'une ou plusieurs observations peuvent être retirées des données.

Régression robuste : S'il y a une ou plusieurs valeurs considérées comme anormales mais qui ont lieu d'être dans l'analyse, on peut améliorer la prédiction en faisant ce que l'on appelle de la "régression robuste". Cela consiste à estimer β par

$$\widehat{\beta} = \underset{\beta\in\mathbb{R}^{p+1}}{\text{Argmin}} \sum_{i=1}^{n} \rho_k(Y_i - x_i\beta),$$

où $x_i = (1, x_{1,i}, \ldots, x_{p,i})$ et ρ_k est la fonction de Huber définie par

$$\rho_k(u) = \begin{cases} u^2 & \text{si } |u| \leq k, \\ 2k|u| - k^2 & \text{sinon.} \end{cases}$$

Cette fonction vaut u^2 lorsque $|u|$ est petit, et est d'ordre $|u|$ ensuite.

par conséquent, elle donne moins de poids aux valeurs anormales, améliorant ainsi le réalisme de la prédiction.

```
library(MASS)
```

La régression robuste avec la fonction de Huber prise en $k = 15$ se fait par les commandes R :

```
reg = rlm(Y ~ X1 + X2 + X3, psi = psi.huber, k = 15)
summary(reg)
```

Observations influentes : Pour identifier les observations qui influent le plus dans les estimations (celles dont une faible variation des valeurs induisent une modification importante des estimations), plusieurs outils complémentaires existent : les DFBETAS, les DFFITS, les rapports de covariance et les distances de Cook.

Si besoin est, pour identifier les observations influentes, on fait :

```
summary(influence.measures(reg))
```

3.2 Multicolinéarité

Problème : Si au moins une des variables parmi $X_1, \ldots, X_p$ a une liaison (presque) linéaire avec d'autres, alors $\det(X^tX) \simeq 0$. Par conséquent, les éléments de la matrice :

$$(X^tX)^{-1} = \frac{1}{\det(X^tX)} com(X^tX)^t$$

seront très grands (à cause de la quantité $1/\det(X^tX) \simeq \infty$).

Comme, pour tout $j \in \{1, \ldots, p\}$, $\widehat{\beta}_j \sim \mathcal{N}\left(\beta_j, \sigma^2[(X^tX)^{-1}]_{j+1,j+1}\right)$, la variance de $\widehat{\beta}_j$ explose.

Conséquence : Cela entraîne une grande instabilité dans l'estimation de β_j et fausse tous les tests statistiques.

En particulier, si au moins une variable parmi $X_1, \ldots, X_p$ a une liaison linéaire avec d'autres, il est possible qu'aucune variable ne montre d'influence significative sur Y et cela, en dépit de

- toute logique,

- du test de Fisher qui peut quand même indiquer une influence significative globale des coefficients (car il prend en compte toutes les variables).

Dans le contexte d'une étude, s'il est vraisemblable que certaines variables de $X_1, \ldots, X_p$ soient liées, il faut étudier ces éventuelles multicolinéarités avant de valider des résultats statistiques.

Méthodes :

- Règle de Klein,
- Facteur d'inflation de la variance (*vif*).

Règle de Klein : On calcule la matrice carré $p \times p$ composée des estimations ponctuelles des corrélations :

$$\rho_{i,j} = \frac{\mathbb{C}(X_i, X_j)}{\sigma(X_i)\sigma(X_j)}.$$

Si une ou plusieurs valeurs au carré sont proches de R^2, alors on soupçonne que les variables associées sont colinéaires.

```
c = cor(cbind(X1, X2, X3), cbind(X1, X2, X3))

c^2
```

Vif : Pour tout $j \in \{1, \ldots, p\}$, on appelle j-ème facteur d'inflation de la variance (*vif*) le réel :

$$V_j = \frac{1}{1 - R_j^2},$$

où R_j^2 désigne le coefficient de détermination de la *rlm* de X_j sur les autres variables.

On peut montrer que la variance estimée de $\widehat{\beta}_j$ est proportionnelle à V_j.

Ainsi, plus le lien linéaire entre X_j et les autres variables est fort, plus R_j^2 est proche de 1, plus V_j est grand et plus l'estimation de β_j est instable.

Critère pratique : Si

$$V_j \geq 5,$$

on admet que X_j a un lien linéaire avec les autres variables.

```
library(car)

vif(reg)
```

Si problème : On propose 3 solutions :

1. On regroupe les variables colinéaires pour n'en former qu'une.

   Par exemple, si on soupçonne que X_j et X_k sont colinéaires, on peut considérer la nouvelle variable $Z = a + b(X_j + X_k)$ (ou $Z = a + b(X_j - X_k)$), avec a et b arbitrairement choisis.

2. On élimine une ou plusieurs des variables colinéaires (en concertation avec un spécialiste des données pour savoir si cela a du sens).

3. On considère un autre estimateur de β :
   - l'estimateur Ridge,
   - l'estimateur LASSO.

Estimateur Ridge : L'estimateur ridge est défini par

$$\widehat{\beta}^* = (X^t X + \lambda \mathbb{I}_p)^{-1} X^t Y,$$

où λ désigne une constante positive.

Il vérifie

$$\widehat{\beta}^* = \underset{\beta \in \mathbb{R}^{p+1}}{\text{Argmin}} \left\{ \|Y - X\beta\|^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}.$$

En general, on le calcule pour plusieurs valeurs de λ.

Une constante λ convenable est estimable avec plusieurs méthodes, dont la méthode du maximum de vraisemblance.

```
library(MASS)
reg = lm.ridge(Y ~ X1 + X2 + X3, lambda = seq(0, 100, 1))
select(reg)
```

Si cela renvoie une valeur estimée pour λ de 4 (par exemple), on considère :

```
regridge = lm.ridge(Y ~ X1 + X2 + X3, lambda = 4)
summary(regridge)
```

Estimateur LASSO : L'estimateur LASSO est défini par

$$\widehat{\beta}^* = \underset{\beta\in\mathbb{R}^{p+1}}{\text{Argmin}} \left\{ \|Y - X\beta\|^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\},$$

où λ désigne une constante positive.

```
library(lars)
X = cbind(1, X1, X2)
reglasso = lars(X, Y, type = "lasso")
summary(reglasso)
```

3.3 Stabilité du modèle

Objectif : Il est intéressant de savoir si le modèle de *rlm* est stable sur l'intégralité de l'échantillon.

Si ce n'est pas le cas, les coefficients de régression du modèle peuvent varier significativement suivant les valeurs de $X_1, \ldots, X_p$.

Pour ce faire, pour tout $n_1 \in \{2, \ldots, n-2\}$, on considère les hypothèses :

$$H_0 : Y = X\beta + \epsilon \qquad \text{contre}$$

$$H_1 : \text{il existe } \beta_1 \neq \beta_2 \text{ tels que } Y^{(1)} = X^{(1)}\beta_1 + \epsilon^{(1)} \text{ et } Y^{(2)} = X^{(2)}\beta_2 + \epsilon^{(2)},$$

avec $Y = (Y^{(1)}, Y^{(2)})^t$, $Y^{(1)}$ de taille n_1, $Y^{(2)}$ de taille $n_2 = n - n_1$, $X = (X^{(1)}, X^{(2)})^t$ et $\epsilon = (\epsilon^{(1)}, \epsilon^{(2)})^t$.

Test de Chow : Pour conclure à la stabilité du modèle, on préconise le test de Chow pour chaque $n_1 \in \{2 + p, \ldots, n - (2 + p)\}$ (de sorte à ce que les degrés de liberté : $\nu_1 = n_1 - (p + 1)$ et $\nu_2 = n_2 - (p + 1)$, vérifient $\nu_1 \geq 1$ et $\nu_2 \geq 1$).

Avant de mettre en œuvre celui-ci, il faut préalablement vérifier que $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$ (par exemple, quand le test de Breusch-Pagan donne p-valeur > 0.05).

Si la plus petite p-valeur est > 0.05, on admet la stabilité du modèle.

```
library(strucchange)
p.value = as.vector(NULL)
n = length(Y)
for(i in 3:(n - 3)) {
p.value[i] = sctest(Y ~ X1, type = "Chow", point = i)$p.value
}
p.value[which.min(p.value)]
```

Si problème : La rupture structurelle est peut-être due à la présence d'une variable qualitative qu'il convient de prendre en compte dans une nouvelle modélisation.

3.4 Sélection de variables

Objectif : Il est intéressant de déterminer la meilleure combinaison des variables $X_1, \ldots, X_p$ qui explique Y.

Or l'approche qui consiste à éliminer d'un seul coup les variables non significatives n'est pas bonne ; certaines variables peuvent être corrélées à d'autres, ce qui peut masquer leur réelle influence sur Y.

Pour retirer les variables $X1$ et $X2$ du modèle (si envie est) :

```
reg2 = update(reg, .~. - X1 - X2)
```

Approches :

- Approche exhaustive,
- Approche en arrière,
- Approche en avant,
- Approche pas à pas.

Approche exhaustive (exhaustive) : On calcule les $\overline{R}^2$ des 2^p *rlm* différentes définies avec toutes les combinaisons possibles de $X_1, \ldots, X_p$.

La meilleure combinaison sera celle ayant le plus grand $\overline{R}^2$.

Par exemple, avec $p = 3$, on considère les $2^3 = 8$ *rlm* :

$$Y = \beta_0 + \epsilon, \qquad Y = \beta_0 + \beta_1 X_1 + \epsilon, \qquad Y = \beta_0 + \beta_2 X_2 + \epsilon$$

$$Y = \beta_0 + \beta_3 X_3 + \epsilon, \qquad Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \epsilon, \qquad Y = \beta_0 + \beta_1 X_1 + \beta_3 X_3 + \epsilon$$

$$Y = \beta_0 + \beta_2 X_2 + \beta_3 X_3 + \epsilon, \qquad Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \epsilon.$$

Cette approche est fastidieuse à mettre en œuvre si p est grand.

```
library(leaps)
v = regsubsets(Y ~ X1 + X2 + X3, w, method = "exhaustive")
plot(v, scale = "adjr2")
```

Les carrés noires de la ligne la plus haute indique la meilleure combinaison de variables explicatives en termes de $\overline{R}^2$.

Approche en arrière (backward) : On part d'une *rlm* avec toutes les variables $X_1, \ldots, X_p$ et on étudie leur significativité.

On retire la moins significative (donc celle qui a la plus grande p-valeur).

Puis on refait une *rlm* avec les variables restantes et on retire de nouveau la moins significative.

On itère ce processus jusqu'à n'avoir que des variables significatives.

Exemple :

1. On considère la *rlm* avec toutes les variables $X_1, \ldots, X_p$:

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \epsilon,$$

et on étudie la significativité de chacune des variables $X_1, \ldots, X_p$.

2. On retire la moins significative. Admettons que ce soit X_3. On considère la *rlm* :

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_4 X_4 + \ldots + \beta_p X_p + \epsilon,$$

et on étudie la significativité de chacune des variables $X_1, X_2, X_4, \ldots, X_p$.

3. On élimine la moins significative. On itère ce processus jusqu'à n'avoir que des variables significatives.

On peut définir cette approche avec le $\overline{R}^2$ en retirant à chaque étape la variable dont le retrait du modèle conduit à la plus grande augmentation du $\overline{R}^2$.

```
library(leaps)
v = regsubsets(Y ~ X1 + X2 + X3, w, method = "backward")
plot(v, scale = "adjr2")
```

Approche en avant (forward) : On considère les p régressions simples possibles avec une seule variable explicative X_1 ou X_2 ou $\dots X_p$ et on étudie leur significativité.

On retient la plus significative (donc celle qui a la plus petite p-valeur).

On fait alors les $p-1$ *rlm* contenant la variable retenue et une seule autre variable parmi les $p-2$ restantes.

On garde alors la plus significative parmi ces dernières.

On itère ce processus jusqu'à qu'aucune variable ne soit retenue.

Exemple :

1. On considère les p régressions linéaires simples :

$$Y = \beta_0 + \beta_j X_j + \epsilon, \qquad j \in \{1, \dots, p\},$$

et on étudie la significativité de chacune des variables $X_1, \dots, X_p$.

2. On garde la plus significative. Admettons que ce soit X_3. On considère alors les $p-1$ *rlm* :

$$Y = \beta_0 + \beta_3 X_3 + \beta_j X_j + \epsilon, \qquad j \in \{1, \dots, p\} - \{3\},$$

et on étudie la significativité de chacune des variables $X_1, X_2, X_4, \dots, X_p$.

3. On garde la plus significative. On itère ce processus jusqu'à qu'aucune variable ne soit retenue.

On peut définir cette approche avec le $\overline{R}^2$ en ajoutant à chaque étape la variable dont l'ajout dans le modèle conduit à la plus grande augmentation du $\overline{R}^2$.

```
library(leaps)
v = regsubsets(Y ~ X1 + X2 + X3, w, method = "forward")
plot(v, scale = "adjr2")
```

Approche pas à pas (stepwise) : Cette approche est un mélange des approches en arrière et en avant. On vérifie que l'ajout d'une variable ne provoque pas la suppression d'une variable déjà introduite.

Cp, AIC et BIC : Il existe d'autres critères que le $\overline{R}^2$:

- le critère Cp (de Mallows) : réalisation de

$$Cp = \frac{\|Y - \widehat{Y}\|^2}{\widehat{\sigma}^2} - (n - 2(p+1)),$$

- le critère AIC : réalisation de

$$AIC = 2(p+1) - 2l,$$

où $l = \max_{\beta \in \mathbb{R}^{p+1}} \ell(\beta)$, le maximum de la log-vraisemblance du modèle,

- le critère BIC : réalisation de

$$BIC = (p+1)\ln(n) - 2l.$$

Ces critères reposent sur un compromis "biais - parcimonie".

Plus petits ils sont, meilleur est le modèle.

Contrairement au $\overline{R}^2$, ces critères s'étendent à d'autres types de modèles, notamment les *modèles linéaires généralisés*.

```
AIC(reg)
BIC(reg)
```

```
library(leaps)
v = regsubsets(Y ~ X1 + X2 + X3, w, method = "backward")
plot(v, scale = "bic")
```

```
library(stats)
```

Pour utiliser l'approche pas à pas avec le AIC, puis obtenir les résultats statistiques associés au modèle sélectionné :

```
reg2 = step(reg, direction = "both", k = 2)
summary(reg2)
```

Pour considérer le BIC, on prend `k = log(length(Y))`

Comparaison de 2 modèles : Pour tester l'influence d'une ou plusieurs variables dans un modèle, tout en prenant en considération les autres variables, on peut utiliser le test ANOVA : si p-valeur > 0.05, alors les variables étudiées ne contribuent pas significativement au modèle.

Si on veut tester $H_0 : \beta_2 = \beta_4 = 0$ en sachant qu'il y a les variables $X1$ et $X3$ dans le modèle, on effectue :

```
reg1 = lm(Y ~ X1 + X2 + X3 + X4)
reg2 = lm(Y ~ X1 + X3)
anova(reg1, reg2)
```

3.5 Traitement de variables qualitatives

Variables qualitatives : Supposons qu'une ou plusieurs variables $X_1, \ldots, X_p$ soient qualitatives. Disons uniquement X_1 de modalités $\{m_1, m_2, m_3\}$ pour simplifier.

Alors on transforme chacune de ses modalités, sauf une, en une variable binaire de la forme :

$$X_1 m_1 = \mathbf{1}_{\{X_1 = m_1\}},$$

laquelle vaut 1 si X_1 a la modalité m_1, et 0 sinon. "Sauf une" sinon la somme de celle-ci ferait 1, injectant de la dépendance superflue dans la modélisation.

En pratique, on supprime celle qui correspond à la situation la plus courante ou plus simplement, celle dont le nom de la modalité se classe premier dans l'ordre alphabétique (c'est ce que fait R).

```
X1 = as.factor(X1)
summary(X1)
```

On considère alors ces nouvelles variables dans le modèle de *rlm* :

$$Y = \beta_0 + \beta_1 X_1 m_1 + \beta_2 X_1 m_2 + \beta_3 X_2 + \beta_4 X_3 + \epsilon,$$

où $\beta_0, \ldots, \beta_4$ sont 5 coefficients inconnus et $\epsilon \sim \mathcal{N}(0, \sigma^2)$ avec $\sigma > 0$ inconnu.

L'analyse de ce modèle s'appelle l'ANCOVA (ANalysis of COVAriance).

```
X1 = as.factor(X1)
reg = lm(Y ~ X1 + X2)
```

Sous-nuages de points : Lorsqu'il y a uniquement 2 variables explicatives $X1$ et $X2$, avec $X1$ qualitative et $X2$ quantitative, on peut tracer le nuage de points de $(X2, Y)$, diviser en sous-nuages correspondants aux modalités de $X1$.

```
reg = lm(Y ~ X1 * X2)
plot(X2[X1 == "m1"], Y[X1 == "m1"], pch = 15, ylab = "Y", xlab = "X2",
col = "green")
points(X2[X1 == "m2"], Y[X1 == "m2"], pch = 16, col = "blue")
points(X2[X1 == "m3"], Y[X1 == "m3"], pch = 17, col = "red")
legend(x = 120, y = 65, c("X1 = m1", "X1 = m2", "X1 = m3"),
col = c("green", "blue", "red"), pch = c(15, 16, 17))
```

L'allure de ces sous-nuages de points est un indicateur sur la possible liaison linéaire entre Y, $X1$ et $X2$.

De plus, si les points correspondants aux différentes modalités sont mélangés, alors il y a une interaction envisageable entre $X1$ et $X2$ sur Y.

Interactions : Il est courant que des variables qualitatives interagissent avec d'autres variables présentent dans le modèle.

Pour prendre en compte cet aspect, il faut introduire des nouvelles variables :

$$X_1 m_1 : X_2, \qquad X_1 m_2 : X_2.$$

Ce qui nous amène au modèle :

$$Y = \beta_0 + \beta_1 X_1 m_1 + \beta_2 X_1 m_2 + \beta_3 X_2 + \beta_4 X_3 + \beta_5 X_1 m_1 : X_2 + \beta_6 X_1 m_2 : X_2 + \epsilon,$$

où $\beta_0, \ldots, \beta_6$ sont 7 coefficients inconnus et $\epsilon \sim \mathcal{N}(0, \sigma^2)$ avec $\sigma > 0$ inconnu.

La commande $X1 * X2$ prend en compte $X1$ seule, $X2$ seule et les interactions $X_1 m_1 : X_2$ et $X_1 m_2 : X_2$.

```
reg = lm(Y ~ X3 + X1 * X2)
```

ou, en prenant en compte plus d'interactions,

```
reg = lm(Y ~ X1 * X2 * X3)
```

Sur l'hypothèse $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$: Lorsqu'une ou plusieurs variables qualitatives sont présentent dans le modèle de régression, certaines méthodes vues précédemment pour vérifier $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$ ne sont plus adaptées (test de Breusch-Pagan...).

On préférera

- une analyse graphique avec
  - toujours le nuage de points $\left\{(\sqrt{|e_i^*|}, y_i - e_i);\ i \in \{1, \ldots, n\}\right\}$: si il n'y a pas de structure, on admet l'égalité des variances,
  - des boites à moustaches pour chacune modalité : si les boites ont a peu près la même étendue, on admet l'égalité des variances,
- le test de Bartlett : si p-valeur > 0.05, on admet l'égalité des variances,
- ou le test de Levene : si p-valeur > 0.05, on admet l'égalité des variances.

```
reg = lm(Y ~ X1 * X2)
e = residuals(reg)
boxplot(e ~ X1)
library(stats)
bartlett.test(e, X1)
```

ou, alternativement,

```
library(lawstat)
levene.test(e, X1)
```

4 Méthode des moindres carrés généralisés (*mcg*)

4.1 Contexte

Écriture matricielle : On rappelle le modèle de *rlm* s'écrit sous la forme matricielle :

$$Y = X\beta + \epsilon,$$

où

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix}, \quad X = \begin{pmatrix} 1 & x_{1,1} & \cdots & x_{p,1} \\ 1 & x_{1,2} & \cdots & x_{p,2} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{1,n} & \cdots & x_{p,n} \end{pmatrix}, \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \quad \epsilon = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}.$$

Hypothèses : On suppose que $(X^tX)^{-1}$ existe, ϵ et $X_1, \ldots, X_p$ sont indépendantes et

$$\epsilon \sim \mathcal{N}_n(0_n, \sigma^2\Omega),$$

où Ω désigne une matrice symétrique, définie positive et inversible.

La définition de Ω inclue

- une possible dépendance dans les *var* $\varepsilon_1, \ldots, \varepsilon_n$,
- une possible inégalité des variances de $\varepsilon_1, \ldots, \varepsilon_n$.

Dans le cas où $\Omega = \mathbb{I}_n$, on retrouve le modèle de *rlm* avec les hypothèses standards.

Idée : Il existe une matrice $\Omega^{-1/2}$ telle que $\Omega^{-1/2}\Omega^{-1/2} = \Omega^{-1}$.

Par conséquent, on peut transformer le modèle initial comme :

$$\Omega^{-1/2}Y = \Omega^{-1/2}X\beta + \Omega^{-1/2}\epsilon,$$

avec $\Omega^{-1/2}\epsilon \sim \mathcal{N}_n(0_n, \sigma^2\mathbb{I}_n)$.

On est donc dans le cadre standard avec une nouvelle écriture matricielle qui utilise $\Omega^{-1/2}Y$ et $\Omega^{-1/2}X$.

Point clés : Toutes les formules du chapitre précédent *Régression linéaire multiple* sont valables avec

- $\Omega^{-1/2}Y$ à la place de Y,
- $\Omega^{-1/2}X$ à la place de X.

Partant de Ω, on détermine $\Omega^{-1/2}$ avec la fonction :

```
fnMatSqrtInverse = function(mA) {
ei = eigen(mA)
d = ei$values
d = (d + abs(d)) / 2
d2 = 1 / sqrt(d)
d2[d == 0] = 0
return(ei$vectors %*% diag(d2) %*% t(ei$vectors))
}
```

On obtient $\Omega^{-1/2}$ en faisant :

```
fnMatSqrtInverse(Omega)
```

Ensuite, on peut transformer Y et X comme :

```
Yo = fnMatSqrtInverse(Omega) %*% Y
X = cbind(1, X1, X2)
Xo = fnMatSqrtInverse(Omega) %*% X
```

4.2 Quelques résultats

Emcg : L'estimateur des moindres carrés généralisés (*emcg*) est

$$\widehat{\beta} = (X^t\Omega^{-1}X)^{-1}X^t\Omega^{-1}Y.$$

C'est aussi l'*emv* de β.

Estimateur de σ^2 : Un estimateur de σ^2 est

$$\widehat{\sigma}^2 = \frac{1}{n-(p+1)}(Y - X\widehat{\beta})^t \Omega^{-1}(Y - X\widehat{\beta}).$$

Il vérifie $\mathbb{E}(\widehat{\sigma}^2) = \sigma^2$, et $\widehat{\sigma}^2$ et $\widehat{\beta}$ sont indépendantes.

Loi de $\widehat{\beta}$: On a

$$\widehat{\beta} \sim \mathcal{N}_{p+1}(\beta, \sigma^2 (X^t \Omega^{-1} X)^{-1}).$$

Loi associée à $\widehat{\sigma}^2$: On a

$$(n-(p+1))\frac{\widehat{\sigma}^2}{\sigma^2} \sim \chi^2(\nu).$$

On transforme Y et X comme :

```
Yo = fnMatSqrtInverse(Omega) %*% Y
X = cbind(1, X1, X2)
Xo = fnMatSqrtInverse(Omega) %*% X
```

On fait une régression linéaire multiple avec les variables transformées :

```
reg = lm(Yo ~ Xo[ ,1]+Xo[ ,2]+Xo[ ,3] - 1)
```

4.3 Hétéroscédasticité des erreurs et *mcg*

Hétéroscédasticité des erreurs : On parle d'hétéroscédasticité des erreurs quand

$$\Omega = \text{diag}(w_1, \ldots, w_n),$$

où $w_1, \ldots, w_n$ sont des réels positifs dont au moins 2 diffèrent.

Notons que $\mathbb{V}(\epsilon_i) = \sigma^2 w_i \ \ (= \mathbb{E}(\epsilon_i^2))$.

En pratique : On admet l'hétéroscédasticité des erreurs lorsque

- il y a indépendance (admis via l'analyse des graphiques *acf* et *pacf* par exemple),
- il n'y a pas l'égalité des variances (par exemple, quand le test de Breusch-Pagan donne p-valeur < 0.05).

Estimation : Lorsque $w_1, \ldots, w_n$ sont connus, les formules précédentes s'appliquent avec

$$\Omega^{-1} = \text{diag}(w_1^{-1}, \ldots, w_n^{-1}).$$

En particulier, l'*emcg* de β est $\widehat{\beta} = (X^t\Omega^{-1}X)^{-1}X^t\Omega^{-1}Y = \text{Argmin}_{\beta\in\mathbb{R}^{p+1}} \|Y - X\beta\|^2_{(1/w)}$, où $\|.\|_{(1/w)}$ désigne la norme euclidienne pondérée de $\mathbb{R}^n$: $\|a\|^2_{(1/w)} = \sum_{i=1}^n (1/w_i)a_i^2$.

Par exemple, si $\Omega = \text{diag}(1, 2, 3, 4, 5)$, les commandes R associées à l'*emcg* $\widehat{\beta}$ sont :

```
w = c(1, 2, 3, 4, 5)
regw = lm(Y ~ X1 + X2 + X3, weights = 1/w)
```

4.4 Cas de données groupées

Contexte : Les n individus sont répartis en q groupes $\mathcal{G}_1, \ldots, \mathcal{G}_q$:

	$\mathcal{G}_1$	$\mathcal{G}_2$	$\ldots$	$\mathcal{G}_q$
Effectif	n_1	n_2	$\ldots$	n_q

Ainsi, $n = \sum_{g=1}^q n_g$. On n'a pas en notre possession les observations $(y_1, x_{1,1}, \ldots, x_{p,1}), \ldots, (y_n, x_{1,n}, \ldots, x_{p,n})$ de $(Y, X_1, \ldots, X_p)$ sur les n individus ; pour chacun des q groupes $\mathcal{G}_1, \ldots, \mathcal{G}_q$, on dispose uniquement des moyennes des observations des individus.

Données : Les données se présentent généralement sous la forme d'un tableau :

Groupe	Effectif	Y	X_1	$\ldots$	X_p
$\mathcal{G}_1$	n_1	$\overline{y}_1$	$\overline{x}_{1,1}$	$\ldots$	$\overline{x}_{p,1}$
$\mathcal{G}_2$	n_2	$\overline{y}_2$	$\overline{x}_{1,2}$	$\ldots$	$\overline{x}_{p,2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathcal{G}_q$	n_q	$\overline{y}_q$	$\overline{x}_{1,q}$	$\ldots$	$\overline{x}_{p,q}$

où, pour tout $g \in \{1, \ldots, q\}$,

$$\overline{y}_g = \frac{1}{n_g} \sum_{i=n_{g-1}+1}^{n_{g-1}+n_g} y_i, \qquad \overline{x}_{j,g} = \frac{1}{n_g} \sum_{i=n_{g-1}+1}^{n_{g-1}+n_g} x_{j,i}.$$

Modélisation : Le modèle de *rlm* sur les variables $Y, X_1, \ldots, X_p$ nous donne :

$$Y_i = \beta_0 + \beta_1 x_{1,i} + \ldots + \beta_p x_{p,i} + \epsilon_i, \qquad i \in \{1, \ldots, n\},$$

avec $\beta_0, \beta, \ldots, \beta_p$ inconnus, $\epsilon_1, \ldots, \epsilon_n$ *iid*, $\epsilon_1 \sim \mathcal{N}(0, \sigma^2)$.

Cependant, ce modèle n'est pas exploitable, ayant accès uniquement aux moyennes des observations.

Sous la forme moyenne, il devient

$$\overline{Y}_g = \beta_0 + \beta_1 \overline{x}_{1,g} + \ldots + \beta_p \overline{x}_{p,g} + \overline{\epsilon}_g, \qquad g \in \{1, \ldots, q\}.$$

avec

$$\overline{Y}_g = \frac{1}{n_g} \sum_{i=n_{g-1}+1}^{n_{g-1}+n_g} Y_i, \qquad \overline{\epsilon}_g = \frac{1}{n_g} \sum_{i=n_{g-1}+1}^{n_{g-1}+n_g} \epsilon_i \sim \mathcal{N}\left(0, \frac{\sigma^2}{n_g}\right).$$

Le modèle peut donc s'écrire sous la forme matricielle :

$$\overline{Y} = \overline{X}\beta + \overline{\epsilon},$$

avec

$$\overline{Y} = \begin{pmatrix} \overline{Y}_1 \\ \overline{Y}_2 \\ \vdots \\ \overline{Y}_q \end{pmatrix}, \qquad \overline{X} = \begin{pmatrix} 1 & \overline{x}_{1,1} & \cdots & \overline{x}_{p,1} \\ 1 & \overline{x}_{1,2} & \cdots & \overline{x}_{p,2} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \overline{x}_{1,q} & \cdots & \overline{x}_{p,q} \end{pmatrix}, \qquad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \qquad \overline{\epsilon} = \begin{pmatrix} \overline{\epsilon}_1 \\ \overline{\epsilon}_2 \\ \vdots \\ \overline{\epsilon}_q \end{pmatrix}.$$

On a $\overline{\epsilon} \sim \mathcal{N}_q(0, \sigma^2\Omega)$ avec

$$\Omega = \text{diag}(n_1^{-1}, \ldots, n_q^{-1}).$$

Les erreurs sont donc héteroscédastiques pour l'indice g avec pour poids $w_g = n_g^{-1}$.

Estimation : Avec ces notations, l'*emcg* de β est

$$\widehat{\beta} = (\overline{X}^t \Omega^{-1} \overline{X})^{-1} \overline{X}^t \Omega^{-1} \overline{Y}.$$

```
effectif = c(12, 13, 12, 11, 8, 9)

regw = lm(Y ~ X1 + X2 + X3, weights = 1 / effectif)
```

4.5 Méthode des *mcqg*

Motivation : Lorsque Ω est inconnue, l'*emcg* de β décrit précédemment n'est pas réaliste.

Afin d'atténuer l'hétéroscedasticité, on préconise 2 méthodes :

- Transformer la variable Y,
- Utiliser les moindres carrés quasi-généralisés (*mcqg*).

Moindres carrés quasi-généralisés (*mcqg*) / Estimateurs de type "sandwich" : La méthode des *mcqg* consiste à estimer ponctuellement Ω à l'aide des données. Dans ce qui suit, on se placera dans le cadre de l'hétéroscédasticité : $\Omega = \text{diag}(w_1, \ldots, w_n)$. On se concentrera donc sur l'estimation de w_i à l'aide des données.

Méthode I : On estime ponctuellement w_i par e_i^2.

On considère alors l'estimateur $\widehat{\beta} = (X^t\widetilde{\Omega}^{-1}X)^{-1}X^t\widetilde{\Omega}^{-1}Y$ de β, avec $\widetilde{\Omega} = \text{diag}(e_1^2, \ldots, e_n^2)$.

```
reg = lm(Y ~ X1 + X2 + X3)

e = residuals(reg)

reg2 = lm(Y ~ X1, weights = 1 / e^2)

summary(reg2)
```

Cette méthode donne des résultats satisfaisants quand n est très grand. En revanche, elle n'est pas fiable quand n est modeste. De plus, elle est sensible aux valeurs (presque) anormales. C'est pourquoi on lui préférera les méthodes à venir.

Pour valider les hypothèses de base, on peut étudier le modèle transformé en faisant :

On calcule $\Omega^{-1/2}$:

```
omegasqrtinv = (1 / e) * diag(n)
```

On calcule les transformations $\Omega^{-1/2}Y$ et $\Omega^{-1/2}X$:

```
Yo = omegasqrtinv %*% Y
X = cbind(1, X1, X2, X3)
Xo = omegasqrtinv %*% X
```

On fait une *rlm* sur Yo et les colonnes de Xo :

```
reg3 = lm(Yo ~ Xo[ ,1] + Xo[ ,2] + Xo[ ,3] + Xo[ ,4] - 1)
summary(reg3)
```

On vérifie la validité de la méthode avec une analyse graphique :

```
par(mfrow = c(2, 2))
plot(reg3, 1:4)
```

On peut également préciser certains points avec les tests statistiques déjà vus.

Méthode II : On modélise $\log(\widehat{\epsilon}_i^2)$ par une *rlm* à partir de $X_1, \ldots, X_p$:

$$\log(\widehat{\epsilon}_i^2) = \theta_0 + \theta_1 x_{1,i} + \ldots + \theta_p x_{p,i} + u_i,$$

avec $u_1, \ldots, u_n$ *n var iid*, $u_1 \sim \mathcal{N}(0, \delta^2)$, $\theta_0, \ldots, \theta_p$ et δ inconnus.

L'exponentielle de l'observation de $\log(\widehat{\epsilon}_i^2)$ donne une estimation ponctuelle de w_i.

On considère alors l'estimateur $\widehat{\beta} = (X^t\widetilde{\Omega}^{-1}X)^{-1}X^t\widetilde{\Omega}^{-1}Y$ de β, avec $\widetilde{\Omega}$ la matrice diagonale composée des estimations ponctuelles de $(w_1, \ldots, w_n)$.

```
reg = lm(Y ~ X1 + X2 + X3)
e = residuals(reg)
rege = lm(log(e^2) ~ X1 + X2 + X3)
reg2 = lm(Y ~ X1, weights = exp(-fitted(rege)))
summary(reg2)
```

Pour valider les hypothèses de base, on peut étudier le modèle transformé en faisant :

On calcule $\Omega^{-1/2}$:

```
omegasqrtinv = exp(-fitted(rege) / 2) * diag(n)
```

On calcule les transformations $\Omega^{-1/2}Y$ et $\Omega^{-1/2}X$:

```
Yo = omegasqrtinv %*% Y
```

```
X = cbind(1, X1, X2, X3)
```

```
Xo = omegasqrtinv %*% X
```

On fait une *rlm* sur Yo et les colonnes de Xo :

```
reg3 = lm(Yo ~ Xo[ ,1] + Xo[ ,2] + Xo[ ,3] + Xo[ ,4] - 1)
```

```
summary(reg3)
```

On vérifie la validité de la méthode avec une analyse graphique :

```
par(mfrow = c(2, 2))
```

```
plot(reg3, 1:4)
```

On peut également préciser certains points avec les tests statistiques déjà vus.

Méthode III : On suppose que w_i s'écrit sous la forme :

$$w_i = (\beta_0 + \beta_1 x_{1,i} + \ldots + \beta_p x_{p,i})^{2\delta},$$

où $\beta_0, \beta_1, \ldots, \beta_p$ sont les coefficients inconnus du modèle de *rlm* initial :

$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \epsilon$, et un δ désigne un nouveau paramètre inconnu.

On considère alors l'*emv* $(\widehat{\beta}_0, \widehat{\beta}_1, \ldots, \widehat{\beta}_p, \widehat{\delta})$ de $(\beta_0, \beta_1, \ldots, \beta_p, \delta)$.

L'observation de $(\widehat{\beta}_0 + \widehat{\beta}_1 x_{1,i} + \ldots + \widehat{\beta}_p x_{p,i})^{2\widehat{\delta}}$ donne une estimation ponctuelle de w_i.

On considère alors l'estimateur $\widehat{\beta} = (X^t \widetilde{\Omega}^{-1} X)^{-1} X^t \widetilde{\Omega}^{-1} Y$ de β, avec $\widetilde{\Omega}$ la matrice diagonale composée des estimations ponctuelles de $(w_1, \ldots, w_n)$.

```
library(nlme)
```

```
reg = gls(Y ~ X1 + X2, weights = varPower(), method = "ML")
```

```
summary(reg)
```

On vérifie la validité de la méthode avec une analyse graphique :

```
plot(predict(reg), residuals(reg, type = "pearson"))
```

Si les points du nuage sont uniformément répartis et qu'il n'y a pas de structure apparente, il n'y a rien à traiter. En particulier, l'hétéroscédasticité est bien corrigée.

Méthode IV : On utilise la méthode connue sous le nom de "HC3" qui estime ponctuellement ω_i par $e_i^2/(1-[X(X^tX)^{-1}X^t]_{i,i})^2$.

On considère alors l'estimateur $\widehat{\beta} = (X^t\widetilde{\Omega}^{-1}X)^{-1}X^t\widetilde{\Omega}^{-1}Y$ de β, avec $\widetilde{\Omega}$ la matrice diagonale composée des estimations ponctuelles de $(w_1,\dots,w_n)$.

Les commandes associées sont :

```
library(lmtest)
library(car)
reg = lm(Y ~ X1 + X2 + X3)
coeftest(reg, vcov = hccm)
```

4.6 Autocorrélation des erreurs et *mcg*

Autocorrélation des erreurs : On parle d'autocorrélation des erreurs lorsqu'il existe $(i,j) \in \{1,\dots,n\}^2$ avec $i \neq j$ tel que $\mathbb{C}(\epsilon_i,\epsilon_j) \neq 0$.

En pratique : On admet l'autocorrélation des erreurs lorsqu'il y a de la dépendance (admis via l'analyse des graphiques *acf* et *pacf* par exemple).

Dans ce cas, il est difficile de traiter le problème dans sa généralité ; on est obligé de supposer une structure simple de dépendance sur les erreurs et de la vérifier ensuite à l'aide des données.

Rappel : structure *AR(1)* : On dit que $\epsilon_1,\dots,\epsilon_n$ ont une structure *AR(1)* si il existe $\rho \in]-1,1[$ et n *var iid* $u_1,\dots,u_n$ suivant chacune la loi normale $\mathcal{N}(0,v^2)$ avec $v>0$, tels que

$$\epsilon_i = \rho\epsilon_{i-1} + u_i.$$

Le réel ρ mesure la dépendance de $\epsilon_1,\dots,\epsilon_n$.

Test de Durbin-Watson : Pour conclure à la structure *AR(1)* de $\epsilon_1, \ldots, \epsilon_n$, partant des résidus $e_1, \ldots, e_n$, on préconise le test de Durbin-Watson : si p-valeur < 0.05, alors on admet que $\rho \neq 0$, entraînant la dépendance.

```
library(lmtest)

dwtest(reg)
```

Quelques résultats théoriques : Si $\epsilon_1, \ldots, \epsilon_n$ ont une structure *AR(1)*, alors

$$\epsilon \sim \mathcal{N}_n(0_n, \sigma^2 \Omega(\rho)),$$

avec $\sigma^2 = \upsilon^2$ et

$$\Omega(\rho) = \left(\frac{1}{1-\rho^2} \rho^{|i-j|} \right)_{(i,j) \in \{1,\ldots,n\}^2} = \frac{1}{1-\rho^2} \begin{pmatrix} 1 & \rho & \cdots & \rho^{n-1} \\ \rho & 1 & \cdots & \rho^{n-2} \\ \cdots & \cdots & \cdots & \cdots \\ \rho^{n-1} & \rho^{n-2} & \cdots & 1 \end{pmatrix}.$$

On a

$$\Omega^{-1}(\rho) = \begin{pmatrix} 1 & -\rho & 0 & \cdots & 0 & 0 & 0 \\ -\rho & 1+\rho^2 & -\rho & \cdots & 0 & 0 & 0 \\ 0 & -\rho & 1+\rho^2 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1+\rho^2 & -\rho & 0 \\ 0 & 0 & 0 & \cdots & -\rho & 1+\rho^2 & -\rho \\ 0 & 0 & 0 & \cdots & 0 & -\rho & 1 \end{pmatrix}.$$

Les commandes R ci-dessous calculent Ω et Ω^{-1} avec $\rho = 0.67$ (par exemple) :

```
n = length(Y)
rho = 0.67
omega = matrix(rep(0, n^2), n, n)
for (i in 1:n){
for (j in 1:n){
omega[i, j] = (1 / (1 - rho^2)) * rho^(abs(i - j))
}}
omega
invomega = solve(omega)
invomega
```

Estimation : L'*emcg* de β est

$$\widehat{\beta} = (X^t \Omega(\rho)^{-1} X)^{-1} X^t \Omega(\rho)^{-1} Y.$$

Toutefois, lorsque ρ est inconnu, cet estimateur n'est pas réaliste puisqu'il en dépend.

Solution 1 : méthode des *mco* : Une première idée est d'estimer ρ par la méthode des *mco*.

On considère le modèle de *rlm* sous les hypothèses standards, l'*emco* et les résidus associés. Une modélisation possible de l'équation caractérisant la structure *AR(1)* est :

$$\widehat{\epsilon}_i = 0 + \rho e_{i-1} + u_i.$$

On est donc dans le cadre d'une régression linéaire simple avec $\beta_0 = 0$ et $\beta_1 = \rho$.

L'*emco* de $\beta_1 = \rho$ est donné par :

$$\widehat{\rho} = \frac{\sum_{i=2}^{n} \widehat{\epsilon}_i e_{i-1}}{\sum_{i=2}^{n} e_{i-1}^2}.$$

```
n = length(Y)
e = residuals(reg)
rho = lm(e[-1] ~ e[-n] - 1)$coeff[1]
rho
```

ou

```
rho = sum(e[-1] * e[-n]) / sum(e[-n]^2)
rho
```

D'où l'estimateur de β :

$$\widehat{\beta} = (X^t \Omega(\widehat{\rho})^{-1} X)^{-1} X^t \Omega(\widehat{\rho})^{-1} Y.$$

Dans le but d'utiliser la commande `lm`, laquelle nous donne des outils de vérifications d'hypothèses puissants, nous allons coder à la main ce qu'il faut :

```
omega = matrix(rep(0, n^2), n, n)
for (i in 1:n){
for (j in 1:n){
omega[i, j] = (1 / (1 - rho^2)) * rho^(abs(i - j))
}}
Yo = fnMatSqrtInverse(omega) %*% Y
X = cbind(1, X1, X2, X3)
Xo = fnMatSqrtInverse(omega) %*% X
reg = lm(Yo ~ Xo[ ,1] + Xo[ ,2] + Xo[ ,3] + Xo[ ,4] - 1)
summary(reg)
```

On vérifie la validité de la méthode avec une analyse graphique :

```
par(frow = c(2, 2))
plot(reg, 1:4)
```

On peut également préciser les choses avec les tests statistiques déjà vus.

Méthode de Cochrane-Orcutt : Il existe une méthode itérative permettant d'estimer ponctuellement ρ. Cette méthode est connue sous le nom de Cochrane-Orcutt.

```
library(orcutt)
reg2 = cochrane.orcutt(reg)
reg2
```

Solution 2 : Maximum de vraisemblance : On peut aussi traiter la structure des erreurs *AR(1)* avec la méthode du maximum de vraisemblance.

En posant $\theta = (\beta, \sigma, \rho)$, la vraisemblance associée à $(Y_1, \ldots, Y_n)$ est

$$L(\theta, z) = \frac{1}{(2\pi\sigma^2 \det(\Omega)(\rho))^{n/2}} \exp\left(-\frac{(z - X\beta)^t \Omega(\rho)^{-1}(z - X\beta)}{2\sigma^2}\right), \qquad z \in \mathbb{R}^n.$$

L'*emv* $\widehat{\theta} = (\widehat{\beta}, \widehat{\sigma}, \widehat{\rho})$ de θ est

$$\widehat{\theta} = \underset{\theta \in \mathbb{R}^{p+3}}{\text{Argmax}}\, L(\theta, Y).$$

Après calculs, il vient

$$\widehat{\beta} = (X^t \Omega(\rho)^{-1} X)^{-1} X^t \Omega(\rho)^{-1} Y,$$

$$\widehat{\sigma^2} = \frac{(Y - X\beta)^t \Omega(\rho)^{-1}(Y - X\beta)}{n}$$

et

$$\widehat{\rho} = \underset{\rho \in]-1,1[}{\text{Argmin}} \left((\det(\Omega(\rho)))^{1/n} (Y - X\widehat{\beta})^t \Omega(\rho)^{-1} (Y - X\widehat{\beta}) \right).$$

Comme $\widehat{\beta}$ dépend du coefficient inconnu ρ, on peut utiliser son estimation $\widehat{\rho}$ dans la définition de $\widehat{\beta}$:

$$\widetilde{\beta} = (X^t \Omega(\widehat{\rho})^{-1} X)^{-1} X^t \Omega(\widehat{\rho})^{-1} Y.$$

```
library(nlme)
```

On a *corARMA(p = 1, q = 0)=AR(1)*, donc

```
reg = gls(Y ~ X1 + X2 + X3, correlation = corARMA(p = 1, q = 0),
method = "ML")
summary(reg)
```

Cette méthode a l'avantage de nous donner beaucoup d'informations, notamment les degrés de significativité de $X_1, \ldots, X_p$, le AIC et le BIC.

On vérifie la validité de la méthode avec une analyse graphique : {(valeurs prédites$_i$, résidus de Pearson$_i$) ; $i \in \{1, \ldots, n\}$} :

```
plot(predict(reg), residuals(reg, type = "pearson"))
```

Si les points du nuage sont uniformément répartis, l'hétéroscédasticité est bien corrigée.

Extension : La structure des erreurs peut être plus complexe qu'un *AR(1)*.

Dans ce cas, les coefficients inconnus de cette structure sont encore estimables en passant par la méthode du maximum de vraisemblance.

```
library(nlme)
reg = gls(Y ~ X1 + X2 + X3, correlation = corARMA(p = 2, q = 3),
method = "ML")
summary(reg)
```

5 Régression non-linéaire

5.1 Contexte

Problématique : Dans une population, on souhaite expliquer une variable quantitative Y à partir de p autres variables $X_1, \ldots, X_p$. Si une liaison non-linéaire entre ces variables est envisageable, on peut considérer le modèle de régression non-linéaire multiple : il existe

- $q+1$ coefficients inconnus $\beta_0, \ldots, \beta_q$,
- une fonction f supposée connue (dans un premier temps),

tels que

$$Y = f(X_1, \ldots, X_p, \beta_0, \ldots, \beta_q) + \epsilon,$$

où ϵ est une variable d'erreur.

Données : Les données dont on dispose sont n observations de $(Y, X_1, \ldots, X_p)$ notées $(y_1, x_{1,1}, \ldots, x_{p,1}), \ldots, (y_n, x_{1,n}, \ldots, x_{p,n})$.

Les données se présentent généralement sous la forme d'un tableau :

Y	X_1	$\ldots$	X_p
y_1	$x_{1,1}$	$\ldots$	$x_{p,1}$
y_2	$x_{1,2}$	$\cdots$	$x_{p,2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y_n	$x_{1,n}$	$\ldots$	$x_{p,n}$

Objectif : Un objectif est d'estimer $\beta_0, \ldots, \beta_q$ à l'aide des données afin de prédire la valeur moyenne de Y pour une nouvelle valeur de $(X_1, \ldots, X_p)$.

Modélisation : On modélise les variables considérées comme des *var* (définies sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$). À partir de celles-ci, le modèle de régression non-linéaire multiple est caractérisé par les points suivants.

Pour tout $i \in \{1, \ldots, n\}$,

- $(x_{1,i}, \ldots, x_{p,i})$ est une réalisation du vecteur aléatoire réel $(X_1, \ldots, X_p)$,
- sachant que $(X_1, \ldots, X_p) = (x_{1,i}, \ldots, x_{p,i})$, y_i est une réalisation de

$$Y_i = f(x_{1,i}, \ldots, x_{p,i}, \beta_0, \ldots, \beta_q) + \epsilon_i,$$

où ϵ_i est une *var* modélisant une somme d'erreurs.

Remarque : Pour tout $x = (x_1, \ldots, x_p) \in \mathbb{R}^p$, sous l'hypothèse que $\mathbb{E}(\epsilon | \{(X_1, \ldots, X_p) = x\}) = 0$, le modèle de régression non-linéaire peut s'écrire comme

$$\mathbb{E}(Y | \{(X_1, \ldots, X_p) = x\}) = f(x_1, \ldots, x_p, \beta_0, \ldots, \beta_q).$$

Hypothèses : On supposera dans la suite que ϵ et $X_1, \ldots, X_p$ sont indépendantes et

$\epsilon = (\epsilon_1, \ldots, \epsilon_n)^t \sim \mathcal{N}_n(0_n, \sigma^2 \mathbb{I}_n)$ où $\sigma > 0$ est un paramètre inconnu.

En particulier, l'hypothèse $\epsilon \sim \mathcal{N}_n(0_n, \sigma^2 \mathbb{I}_n)$ entraîne que

- $\epsilon_1, \ldots, \epsilon_n$ sont indépendantes,
- $\mathbb{V}(\epsilon_1) = \ldots = \mathbb{V}(\epsilon_n)$,
- $\epsilon_1, \ldots, \epsilon_n$ suivent des lois normale centrées.

5.2 Régression polynomiale

Définition : On parle de régression polynomiale quand f est définie comme un polynôme par rapport à $X_1, \ldots, X_p$.

Cette représentation polynomiale peut être motivée par l'intuition, le contexte de l'expérience d'où émanent les données ou des critères statistiques précis (comme les résidus partiel présentés ci-après).

Exemple : On suppose que f s'écrit sous la forme :

$$f(X_1, X_2, \beta_0, \beta_1, \beta_2, \beta_3, \beta_4) = \beta_0 + \beta_1 X_1 + \beta_2 X_1^2 + \beta_3 X_1^3 + \beta_4 X_2,$$

ce qui correspond au modèle :

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_1^2 + \beta_3 X_1^3 + \beta_4 X_2 + \epsilon.$$

```
reg = lm(Y ~ I(X1) + I(X1^2) + I(X1^3) + X2)
ou
reg = lm(Y ~ poly(X1, 3, raw = T) + X2)
summary(reg)
```

On peut atténuer les effets de dépendance entre les variables explicatives puissances en considérant une base de polynômes obtenue par le processus d'orthonormalisation de Schmidt.

```
reg = lm(Y ~ poly(X1, 3) + X2)
summary(reg)
```

5.3 Résidus partiels

Motivation : Il est souvent intéressant d'analyser le lien réel entre Y et une variable explicative X_j, laquelle est linéaire par hypothèse dans le modèle *rlm*.

Or l'approche intuitive qui consiste à étudier visuellement le nuage de points :

$$\{(x_{j,i}, y_i);\ i \in \{1, \ldots, n\}\}, \qquad j \in \{1, \ldots, p\}$$

et son ajustement par une droite n'est pas fiable.

En effet, toutes les autres variables explicatives peuvent aussi influencer les valeurs de Y et ainsi brouiller notre impression.

C'est pourquoi, on cherche à mesurer l'influence seule de X_j sur Y, laquelle peut ne pas être linéaire.

Cela va nous permettre de choisir des fonctions $g_1, \ldots, g_p$ convenables telles que le modèle de régression :

$$Y = \beta_0 + \beta_1 g_1(X_1) + \ldots + \beta_p g_p(X_p) + \epsilon,$$

soit mieux adapté au problème.

On est donc dans le cadre de la régression non-linéaire avec

$$f(X_1, \ldots, X_p, \beta_0, \ldots, \beta_p) = \beta_0 + \beta_1 g_1(X_1) + \ldots + \beta_p g_p(X_p).$$

Résidus partiels : Pour tout $(i, j) \in \{1, \ldots, n\} \times \{1, \ldots, p\}$, on appelle (j, i)-ème résidus partiel la réalisation $e_{j,i}^{\triangle}$ de

$$\widehat{\epsilon}_{j,i}^{\triangle} = \widehat{\beta}_j x_{j,i} + \widehat{\epsilon}_j.$$

Intérêt des résidus partiels : L'intérêt des résidus partiels est d'enlever l'effet estimé de toutes les variables explicatives autres que X_j sur Y.

En effet, on peut montrer que

$$\widehat{\epsilon}_{j,i}^{\triangle} = Y_i - (\widehat{\beta}_1 x_{1,i} + \ldots + \widehat{\beta}_{j-1} x_{j-1,i} + \widehat{\beta}_{j+1} x_{j+1,i} + \ldots + \widehat{\beta}_p x_{p,i}).$$

Ainsi, la nature de la liaison existante entre X_j et Y se représente avec le nuage de point :

$$\left\{(x_{j,i}, e_{j,i}^{\triangle});\ i \in \{1, \ldots, n\}\right\}.$$

S'il peut s'ajuster simplement par une droite, alors la liaison entre X_j et Y est linéaire, ce qui rejoint l'hypothèse de départ.

Les autres cas sont problématiques et traduisent une liaison non-linéaire entre X_j et Y.

```
reg = lm(Y ~ X1 + X2 + X3)
library(car)
crPlots(reg)
```

Si problème : Si le nuage de points a une allure courbe, il est avantageux de considérer une transformation de X_j correspondante à l'équation de la courbe qui ajuste au mieux le nuage.

Plus précisément, si le nuage de points peut être ajusté par la courbe d'équation $y = g_j(x)$, alors

- soit on remplace X_j par $g_j(X_j)$ dans le modèle et on refait les analyses,
- soit on ajoute la nouvelle variable $g_j(X_j)$ dans le modèle et on refait les analyses ; les 2 variables X_j et $g_j(X_j)$ sont donc prises en compte, ajoutant ainsi un nouveau coefficient inconnu à estimer.

Transformations : Les transformations les plus courantes sont :

- les transformations polynomiales :

```
reg = lm(Y ~ poly(X1, 3) + X2 + I(X3^4))
summary(reg)
```

- les transformations avec des fonctions usuelles :

```
reg = lm(Y ~ exp(X1) + log(X2) + 2^X3)
summary(reg)
```

- les transformations de Box-Cox :

$$g_j(u) = bc_{\lambda_j}(u) = \begin{cases} \dfrac{u^{\lambda_j} - 1}{\lambda_j} & \text{si } \lambda_j \neq 0, \\ \log(u) & \text{sinon.} \end{cases}$$

```
library(car)
reg = lm(Y ~ bcPower(X1, 2.7) + bcPower(X2, 1.8))
summary(reg)
```

Si les $\lambda_1, \ldots, \lambda_p$ sont inconnus (ce qui est généralement le cas), on peut utiliser la méthode de Box-Tidwell qui donne des *emv* de ceux-ci.

```
library(car)
boxTidwell(Y ~ X1 + X2 + X3)
```

Puis on fait une *rlm* entre Y et les transformations de Box-Cox de $X1$ et $X2$ aux puissances estimées.

Si une variable explicative $X4$ est présente dans l'analyse mais on ne souhaite pas la transformer, on la prend en compte dans les estimations de $\lambda_1, \lambda_2, \lambda_3$ en faisant

```
boxTidwell(Y ~ X1 + X2 + X3, other.x = ~ X4)
```

5.4 Méthodes itératives

Estimateur : Partant du modèle sous sa forme générale :

$$Y = f(X_1, \ldots, X_p, \beta_0, \ldots, \beta_q) + \epsilon,$$

on souhaite construire un estimateur $\widehat{\beta} = (\widehat{\beta}_0, \ldots, \widehat{\beta}_q)^t$ tel que

$$\widehat{\beta} = \underset{\beta \in \mathbb{R}^{q+1}}{\text{Argmin}} \sum_{i=1}^{n} (Y_i - f(x_{1,i}, \ldots, x_{p,i}, \beta_0, \ldots, \beta_q))^2.$$

Or un tel estimateur n'a pas de forme analytique.

Ainsi, pour avoir une estimation ponctuelle de β, on utilise des algorithmes itératifs, notamment :

- l'algorithme de Gauss-Newton,
- l'algorithme de Newton-Raphson.

Algorithme de Gauss-Newton : Soit $b = (b_0, \ldots, b_q)^t$ la réalisation de $\widehat{\beta}$ correspondante aux données. On pose

$$\eta(b) = \begin{pmatrix} f(x_{1,1}, \ldots, x_{p,1}, b) \\ \vdots \\ f(x_{1,n}, \ldots, x_{p,n}, b) \end{pmatrix}.$$

On part d'une valeur initiale de b soit $b^{(0)}$ que l'on suppose proche de la solution.

Un développement limité autour de cette valeur donne :

$$\eta(b) \simeq \eta(b^{(0)}) + D\eta(b^{(0)})(b - b^{(0)}).$$

Dans $\|y - \eta(b)\|^2$, on remplace $\eta(b)$ par la partie principale du développement limité. On se ramène donc à chercher $b^{(1)}$ qui minimise :

$$b \to \|y - (\eta(b^{(0)}) + D\eta(b^{(0)})(b - b^{(0)}))\|^2.$$

On pose $y = (y_1, \ldots, y_n)^t$, $z = y - \eta(b^{(0)})$, $\gamma = b - b^{(0)}$ et $X = D\eta(b^{(0)})$.

Il s'agit maintenant de minimiser $\gamma \to \|z - X\gamma\|^2$. On obtient $\gamma^{(1)} = (X^tX)^{-1}X^tz$, ce qui entraine $\gamma^{(1)} = \left(D\eta(b^{(0)})^tD\eta(b^{(0)})\right)^{-1} D\eta(b^{(0)})^t(y - \eta(b^{(0)}))$. On en déduit que

$$b^{(1)} = b^{(0)} + \left(D\eta(b^{(0)})^tD\eta(b^{(0)})\right)^{-1} D\eta(b^{(0)})^t(y - \eta(b^{(0)})).$$

Par conséquent, l'algorithme est le suivant :

$$b^{(i+1)} = b^{(i)} + \left(D\eta(b^{(i)})^tD\eta(b^{(i)})\right)^{-1} D\eta(b^{(i)})^t(y - \eta(b^{(i)})).$$

On itère le processus jusqu'à avoir stabilisation :

$$b^{(m+1)} \simeq b^{(m)}.$$

```
library(MASS)
s = c(b0 = 90, b1 = 95, b2 = 120)
reg = nls(Y ~ b0 + b1 * exp(-X1 / b2), start = s)
summary(reg)
```

Graphiques :

```
plot(X1, Y)
lines(X1, predict(reg))
```

Algorithme de Newton-Raphson : Soit $b = (b_0, \ldots, b_q)^t$ la réalisation de $\widehat{\beta}$ correspondante aux données. On pose $\eta(b) = \begin{pmatrix} f(x_{1,1}, \ldots, x_{p,1}, b) \\ \vdots \\ f(x_{1,n}, \ldots, x_{p,n}, b) \end{pmatrix}$, $y = (y_1, \ldots, y_n)^t$ et $\mathcal{L}_2(b) = \|y - \eta(b)\|^2$. Pour une étape $b^{(i)}$ que l'on suppose proche de la solution (qui minimise $\mathcal{L}_2$) on développe $\text{grad}_b\mathcal{L}_2$ au voisinage de $b^{(i)}$:

$$\text{grad}_b\mathcal{L}_2(b) \simeq \text{grad}_b\mathcal{L}_2(b^{(i)}) + D\text{grad}_b\mathcal{L}_2(b^{(i)})(b - b^{(i)}).$$

On suppose que l'étape suivante $b^{(i+1)}$ sera proche de la solution, laquelle vérifie $\text{grad}_b\mathcal{L}_2(b) = 0$. Donc on cherche $b^{(i+1)}$ tel que : $0 \simeq \text{grad}_b\mathcal{L}_2(b^{(i)}) + D\text{grad}_b\mathcal{L}_2(b^{(i)})(b^{(i+1)} - b^{(i)})$.

Par conséquent, l'algorithme est le suivant :

$$b^{(i+1)} = b^{(i)} - \left(\text{Hess}\,\mathcal{L}_2(b^{(i)})\right)^{-1} \text{grad}_b\mathcal{L}_2(b^{(i)}),$$

avec $\text{Hess}\,\mathcal{L}_2(b^{(i)}) = \text{grad}_b(-2D\eta(b)^t(y - \eta(b)) = 2\left(D\eta(b)^t D\eta(b) - \sum_{i=1}^{n}(y_i - \eta_i(b))\,\frac{\partial^2\eta_i(b)}{\partial b \partial b^t}\right)$ et

$$\frac{\partial^2\eta_i(b)}{\partial b \partial b^t} = \begin{pmatrix} \frac{\partial^2\eta_i(b)}{\partial b_1 \partial b_1} & \cdots & \frac{\partial^2\eta_i(b)}{\partial b_1 \partial b_p} \\ \cdots & \cdots & \cdots \\ \frac{\partial^2\eta_i(b)}{\partial b_p \partial b_1} & \cdots & \frac{\partial^2\eta_i(b)}{\partial b_p \partial b_p} \end{pmatrix}.$$

On itère le processus jusqu'à avoir stabilisation :

$$b^{(m+1)} \simeq b^{(m)}.$$

Estimateur de σ^2 : Un estimateur de σ^2 est

$$\widehat{\sigma}^2 = \frac{1}{n - (q+1)} \sum_{i=1}^{n} \left(Y_i - f(x_i, \widehat{\beta})\right)^2.$$

Validation des hypothèses : Pour valider les hypothèses standards, voici une proposition d'analyse graphique, avec les mêmes interprétations que pour la *rlm* :

```
e = residuals(reg)
par(mfrow = c(2, 2))
plot(e)
acf(e)
qqnorm(scale(e))
qqline(e)
plot(fitted(reg), e)
```

5.5 Extension : régression non-paramétrique

Problématique : Dans une population, on souhaite expliquer une variable quantitative Y à partir de p autres variables $X_1, \ldots, X_p$.

Si une liaison non-linéaire entre ces variables est envisageable, on peut considérer le modèle de régression non-paramétrique : il existe une fonction inconnue f telle que

$$Y = f(X_1, \ldots, X_p) + \epsilon,$$

où ϵ est une variable d'erreur.

Données : Les données dont on dispose sont n observations de $(Y, X_1, \ldots, X_p)$ notées $(y_1, x_{1,1}, \ldots, x_{p,1}), \ldots, (y_n, x_{1,n}, \ldots, x_{p,n})$.

Les données se présentent généralement sous la forme d'un tableau :

Y	X_1	$\ldots$	X_p
y_1	$x_{1,1}$	$\ldots$	$x_{p,1}$
y_2	$x_{1,2}$	$\cdots$	$x_{p,2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y_n	$x_{1,n}$	$\ldots$	$x_{p,n}$

Objectif : Un objectif est d'estimer la fonction f (et non des paramètres) à l'aide des données afin de prédire la valeur moyenne de Y pour une nouvelle valeur de $(X_1, \ldots, X_p)$.

Estimateur de Nadaraya-Watson : On se limite au cas d'une seule variable explicative X_1. On appelle estimateur de Nadaraya-Watson l'estimateur :

$$\widehat{f}_n(x) = \frac{\sum_{i=1}^{n} K_h(x - x_{1,i}) Y_i}{\sum_{i=1}^{n} K_h(x - x_{1,i})},$$

où

$$K_h(u) = \frac{1}{h} K\left(\frac{u}{h}\right).$$

Dans cette écriture, K désigne un noyau (*kernel*) et h une fenêtre.

L'idée de la construction de $\widehat{f}_n$ est la suivante : en notant $g(x, y)$ la densité de (X_1, Y) et $g(x)$ celle de X_1, on a

$$f(x) = \mathbb{E}(Y|X_1 = x) = \int_{-\infty}^{\infty} y g(y|x) dy = \frac{\int_{-\infty}^{\infty} y g(x, y) dy}{g(x)} = \frac{\int_{-\infty}^{\infty} y g(x, y) dy}{\int_{-\infty}^{\infty} g(x, y) dy}.$$

En utilisant les propriétés particulières de K, on montre que

- un estimateur de $\int_{-\infty}^{\infty} y g(x, y) dy$ est $(1/n) \sum_{i=1}^{n} K_h(x - x_{1,i}) Y_i$,
- un estimateur de $\int_{-\infty}^{\infty} g(x, y) dy$ est $(1/n) \sum_{i=1}^{n} K_h(x - x_{1,i})$.

Il existe de nombreuses méthodes pour choisir h de manière à ce que $\widehat{f}_n$ soit un bon estimateur de f.

```
library(np)
reg = npreg(Y ~ X1)
summary(reg)
plot(reg)
points(X1, Y)
predict(reg)
predict(reg, newdata = data.frame(X1 = 1.2))
```

Pour valider les hypothèses standards, voici une proposition d'analyse graphique :

```
e = residuals(reg)
plot(e)
acf(e)
qqnorm(e)
qqline(e)
plot(fitted(reg), e)
```

Estimateur par splines : On se limite au cas d'une seule variable explicative X_1. Le point de départ est la décomposition de f à l'aide de fonctions polynomiales par morceaux appelées splines. Ces splines dépendent de paramètres appelés nœuds. Dès lors, on suppose qu'il existe $q+1$ coefficients inconnus $\beta_0, \ldots, \beta_q$ tels que

$$f(x) = \beta_0 + \beta_1 B_1(x) + \ldots + \beta_q B_q(x),$$

où $B_1(x), \ldots, B_q(x)$ désignent les fonctions splines (ou des combinaisons de splines).

On est alors en mesure d'estimer $\beta_0, \ldots, \beta_q$ ainsi que les nœuds des splines (avec la méthode des moindres carrés pénalisés). On appelle estimateur par splines un estimateur de la forme :

$$\widehat{f}(x) = \widehat{\beta}_0 + \widehat{\beta}_1 \widehat{B}_1(x) + \ldots + \widehat{\beta}_q \widehat{B}_q(x),$$

où, pour tout $j \in \{1, \ldots, q\}$, $\widehat{B}_j(x)$ reprend la construction de $B_j(x)$ en y injectant les estimations des nœuds.

```
library(stats)
reg = smooth.spline(X1, Y, nknots = 5)
plot(X1, Y)
lines(reg)
u = seq(3, 30, by = 0.1)
predict(reg, u)
```

6 Régression logistique

6.1 Contexte

Problématique : On considère une population $\mathcal{P}$ divisée en 2 groupes d'individus G_1 et G_2 distinguables par des variables $X_1, \dots, X_p$.

Soit Y la variable qualitative telle que $Y(\omega) = 1$ si un individu ω extrait au hasard dans $\mathcal{P}$ appartient à G_1 et $Y(\omega) = 0$ sinon.

On souhaite expliquer Y à partir de $X_1, \dots, X_p$.

Données : Les données dont on dispose sont n observations de $(Y, X_1, \dots, X_p)$ notées

$(y_1, x_{1,1}, \dots, x_{p,1}), \dots, (y_n, x_{1,n}, \dots, x_{p,n})$.

Les données se présentent généralement sous la forme d'un tableau :

Y	X_1	$\dots$	X_p
y_1	$x_{1,1}$	$\dots$	$x_{p,1}$
y_2	$x_{1,2}$	$\dots$	$x_{p,2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y_n	$x_{1,n}$	$\dots$	$x_{p,n}$

où, pour tout $(i, j) \in \{1, \dots, n\} \times \{1, \dots, p\}$, $x_{j,i}$ est l'observation de la variable X_j sur le i-ème individu et y_i indique le groupe dans lequel il appartient : $y_i \in \{0, 1\}$.

Modélisation : On modélise les variables considérées comme des *var* (définies sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$).

Pour tout $i \in \{1, \dots, n\}$,

- $(x_{1,i}, \dots, x_{p,i})$ est une réalisation du vecteur aléatoire réel $(X_1, \dots, X_p)$,
- sachant que $(X_1, \dots, X_p) = (x_{1,i}, \dots, x_{p,i}) = x_i$, y_i est une réalisation de

$$Y_i \sim \mathcal{B}(p(x_i)), \qquad p(x_i) = \mathbb{P}(\{Y = 1\} | \{(X_1, \dots, X_p) = x_i\}).$$

Objectif : Un objectif est d'estimer la probabilité inconnue qu'un individu ω vérifiant $(X_1, \ldots, X_p) = x$ appartienne au groupe G1 :

$$p(x) = \mathbb{P}(\{Y = 1\} | \{(X_1, \ldots, X_p) = x\}), \qquad x = (x_1, \ldots, x_p),$$

à l'aide des données.

Remarque : $p(x)$ est aussi la valeur moyenne de Y :

$$p(x) = \mathbb{E}(Y | \{(X_1, \ldots, X_p) = x\}).$$

6.2 Transformation logit

Problème : Si on exprime $p(x)$ avec $x = (x_1, \ldots, x_p)$ comme

$$p(x) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p,$$

alors au moins 2 problèmes surviennent :

- on a $p(x) \in [0, 1]$ alors que $\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p \notin [0, 1]$ a priori,
- quand $p(x)$ tend vers 0 ou 1, on doit avoir $\frac{\partial}{\partial x_j} p(x)$ qui tend vers 0. Or $\frac{\partial}{\partial x_j} p(x) = \beta_j$ ne tend pas vers 0 a priori.

Transformation logit : On appelle transformation logit la fonction :

$$\text{logit}(y) = \log\left(\frac{y}{1-y}\right) \in \mathbb{R}, \qquad y \in]0, 1[.$$

Son inverse est la fonction :

$$\text{logit}^{-1}(y) = \frac{\exp(y)}{1 + \exp(y)} \in]0, 1[, \qquad y \in \mathbb{R}.$$

Régression logistique : On appelle régression logistique la modélisation :

$$\text{logit}(p(x)) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p,$$

où $\beta_0, \ldots, \beta_p$ désignent $p+1$ coefficients inconnus.

Ainsi, $p(x)$ et $\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$ sont liés par la transformation logit ; on parle de lien logit.

On en déduit l'expression de $p(x)$:

$$p(x) = \text{logit}^{-1}(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p) = \frac{\exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p)}{1 + \exp(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p)},$$

Objectif : Notre objectif est d'estimer $\beta_0, \ldots, \beta_p$ à partir des données. Pour ce faire, on utilise la méthode du maximum de vraisemblance.

6.3 Variable latente

Variable latente : Dans de nombreux contextes, il existe une variable latente Y_* telle que :

- Y peut se modéliser comme :

$$Y = \begin{cases} 1 & \text{si } Y_* \geq 0, \\ 0 & \text{sinon.} \end{cases}$$

- une liaison linéaire entre Y_* et $X_1, \ldots, X_p$ est envisageable.

Typiquement, $Y_* = T - S$, où T désigne une variable quantitative et S une variable de seuil : $Y = 1$ correspond alors à $T \geq S$.

Par exemple, un individu sera considéré comme malade : $Y = 1$ pour présence de maladie, lorsque sa fièvre, variable T, dépasse un certain seuil, variable S. Donc $Y_* = T - S$.

Modélisation : On peut modéliser Y_* par une *rlm* :

$$Y_* = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \gamma,$$

où $\beta_0, \ldots, \beta_p$ sont $p+1$ coefficients inconnus et γ est une *var*

- symétrique (γ et $-\gamma$ suivent la même loi) de densité f_γ et de fonction de répartition F_γ,
- indépendante de $X_1, \ldots, X_p$.

Donc

$$\begin{aligned} p(x) &= \mathbb{P}(\{Y=1\}|\{(X_1,\ldots,X_p)=x\}) = \mathbb{P}(\{Y_*\geq 0\}|\{(X_1,\ldots,X_p)=x\}) \\ &= \mathbb{P}(\beta_0+\beta_1 x_1+\ldots+\beta_p x_p+\gamma\geq 0) \\ &= \mathbb{P}(-\gamma\leq \beta_0+\beta_1 x_1+\ldots+\beta_p x_p) = \mathbb{P}(\gamma\leq \beta_0+\beta_1 x_1+\ldots+\beta_p x_p) \\ &= F_\gamma\left(\beta_0+\beta_1 x_1+\ldots+\beta_p x_p\right). \end{aligned}$$

On en déduit que

$$F_\gamma^{-1}(p(x)) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p.$$

On définit ainsi une infinité de fonctions de lien dépendant de la loi de γ.

Par exemple, si γ suit la loi logistique $\mathcal{L}(1)$, alors

$$f_\gamma(x) = \frac{e^x}{(1+e^x)^2}, \qquad F_\gamma(y) = \frac{e^y}{1+e^y}, \qquad F_\gamma^{-1}(y) = \text{logit}(y).$$

On retrouve le modèle de régression logistique.

Liens : Les autres fonctions de lien les plus utilisées sont :

- le lien probit : $\gamma \sim \mathcal{N}(0,1)$:

$$\text{probit}(y) = F_\gamma^{-1}(y), \qquad F_\gamma(y) = \int_{-\infty}^{y} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t^2}{2}\right) dt.$$

Pour travailler avec ce lien, dans les commandes R à venir, il faut préciser :

```
family = binomial(link = "probit")
```

- le lien cloglog : $\gamma \sim \text{Gompertz}(0,1)$:

$$\text{cloglog}(y) = F_\gamma^{-1}(y), \qquad F_\gamma(y) = 1 - \exp(-\exp(y)).$$

Pour travailler avec ce lien, dans les commandes R à venir, il faut préciser :

```
family = binomial(link = "cloglog")
```

- le lien cauchit : $\gamma \sim \text{Cauchy}(0,1)$:

$$\text{probit}(y) = F_\gamma^{-1}(y), \qquad F_\gamma(y) = \frac{1}{\pi}\arctan(y) + \frac{1}{2}.$$

Pour travailler avec ce lien, dans les commandes R à venir, il faut préciser :

```
family = binomial(link = "cauchit")
```

Dans la suite, on considère uniquement le lien logit en raison de sa simplicité.

6.4 Estimation

Vraisemblance : Soit $\beta = (\beta_0, \ldots, \beta_p)$. La vraisemblance associée à $(Y_1, \ldots, Y_n)$ est

$$L(\beta, z) = \prod_{i=1}^{n} p(x_i)^{z_i}(1 - p(x_i))^{1-z_i}, \qquad z = (z_1, \ldots, z_n) \in \{0,1\}^n,$$

avec $x_i = (x_{1,i}, \ldots, x_{p,i})$.

Emv : Les estimateurs du maximum de vraisemblance de $\beta_0, \ldots, \beta_p$, notés $\widehat{\beta}_0, \ldots, \widehat{\beta}_p$, vérifient les équations :

$$\sum_{i=1}^{n} \left(y_i - \text{logit}^{-1}(\widehat{\beta}_0 + \widehat{\beta}_1 x_{1,i} + \ldots + \widehat{\beta}_p x_{p,i}) \right) = 0$$

et, pour tout $j \in \{1, \ldots, p\}$,

$$\sum_{i=1}^{n} x_{j,i} \left(y_i - \text{logit}^{-1}(\widehat{\beta}_0 + \widehat{\beta}_1 x_{1,i} + \ldots + \widehat{\beta}_p x_{p,i}) \right) = 0.$$

Dans le cas général, il n'y a pas d'expression analytique pour $\widehat{\beta}_0, \ldots, \widehat{\beta}_p$; ils peuvent être approchés avec l'algorithme de Newton-Raphson.

Estimation : Une estimation de $p(x)$ avec $x = (x_1, \ldots, x_p)$ est la réalisation de

$$\widehat{p}(x) = \text{logit}^{-1}(\widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_p x_p) = \frac{\exp(\widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_p x_p)}{1 + \exp(\widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_p x_p)}.$$

```
library(stats)
reg = glm(Y ~ X1 + X2 + X3, family = binomial)
reg
predict.glm(reg, data.frame(X1 = 15, X2 = 12.6, X3 = 4), type = "response")
```

La commande `predict.glm` utilise, par défaut, la transformation Logit de $\widehat{p}(x)$.

C'est pourquoi il faut préciser `type = "response"` lorsque que l'on veut faire de la prédiction sur $p(x)$ uniquement.

Pour tracer la ligne logistique, on exécute les commandes R :

```
plot(X1, Y)
curve(predict(reg, data.frame(X1 = x), type = "response"), add = T)
```

Prédiction du groupe : On appelle prédiction du groupe d'un individu ω vérifiant $(X_1, \ldots, X_p) = x$ la réalisation de

$$\widehat{Y}_x = \mathbf{1}_{\{\widehat{p}(x) \geq 1 - \widehat{p}(x)\}} = \begin{cases} 1 & \text{si } \widehat{p}(x) \geq \dfrac{1}{2}, \\ 0 & \text{sinon.} \end{cases}$$

```
pred.prob = predict.glm(reg, data.frame(X1 = 15, X2 = 12.6, X3 = 4),
type = "response")
pred.mod = factor(ifelse(pred.prob > 0.5, "1", "0"))
pred.mod
```

Variance / écart-type : En posant $\widehat{\beta} = (\widehat{\beta}_0, \ldots, \widehat{\beta}_p)^t$, on a

$$\widehat{V}(\widehat{\beta}) = \left(-\frac{\partial^2}{\partial \beta^2} \log L(\beta, Y) \right)^{-1} |_{\beta=\widehat{\beta}} = (X^t W X)^{-1},$$

où

$$X = \begin{pmatrix} 1 & x_{1,1} & \cdots & x_{p,1} \\ 1 & x_{1,2} & \cdots & x_{p,2} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{1,n} & \cdots & x_{p,n} \end{pmatrix}$$

et $W = \text{diag}\left(\widehat{p}(x_1)(1-\widehat{p}(x_1)), \ldots, \widehat{p}(x_n)(1-\widehat{p}(x_n))\right)$.

Dans la suite, on posera $\widehat{\sigma}(\widehat{\beta}_j) = \sqrt{\widehat{V}(\widehat{\beta}_j)}$, racine carrée de la $j+1$-ème composante du vecteur $\widehat{V}(\widehat{\beta})$.

6.5 Significativité de la régression

Test de Wald : Soit $j \in \{0, \ldots, p\}$. L'objectif du test de Wald est d'évaluer l'influence (ou la contribution) de X_j sur Y.

On considère les hypothèses :

$$H_0 : \beta_j = 0 \qquad \text{contre} \qquad H_1 : \beta_j \neq 0.$$

On calcule la réalisation z_{obs} de

$$Z_* = \frac{\widehat{\beta}_j}{\widehat{\sigma}(\widehat{\beta}_j)}.$$

On considère une *var* $Z \sim \mathcal{N}(0,1)$.

Alors la p-valeur associée est

$$\text{p-valeur} = \mathbb{P}(|Z| \geq |z_{obs}|).$$

Ce test repose sur le fait que l'*emv* à pour loi asymptotique la loi normale.

Si $*$, l'influence de X_j sur Y est significative, si $**$, elle est très significative, si $***$, elle est hautement significative.

```
summary(reg)
```

Déviance : La déviance du modèle est la réalisation de

$$D = 2\sum_{i=1}^{n}\left(Y_i \log\left(\frac{Y_i}{\widehat{p}(x_i)}\right) + (1-Y_i)\log\left(\frac{1-Y_i}{1-\widehat{p}(x_i)}\right)\right)$$

avec $x_i = (1, x_{1,i}, \ldots, x_{p,i})$.

C'est 2 fois la différence entre les log-vraisemblances évaluées en Y_i et $\widehat{p}(x_i)$.

Loi de D : Si le modèle est bien adapté au problème (ou exact), la loi limite (ou exacte) de D est $\chi^2(n-(p+1))$.

Test de la déviance : Soit $j \in \{0, \ldots, p\}$. L'objectif du test de la déviance est d'évaluer l'influence (ou la contribution) de X_j sur Y.

La p-valeur associée utilise la loi du Chi-deux : si $*$, l'influence de X_j sur Y est significative, si $**$, elle est très significative et si $***$, elle est hautement significative.

```
anova(reg, test = "Chisq")
```

Test lr : On considère les hypothèses :

$$H_0 : \beta_1 = \beta_2 = \ldots = \beta_p = 0 \qquad \text{contre} \qquad H_1 : \text{il y a au moins un coefficient non nul.}$$

Pour ce faire, on utilise le test du rapport de vraisemblance (lr pour likelihood ratio) (asymptotique). La p-valeur associée utilise la loi du Chi-deux : si $*$, l'influence d'au moins une variable est significative, si $**$, très significative et si $***$, hautement significative.

```
library(rms)
reg2 = lrm(Y ~ X1 + X2)
reg2
```

Autre solution :

```
reg = glm(Y ~ X1 + X2, family = binomial)
reg0 = glm(Y ~ 1, family = binomial)
anova(reg0, reg, test = "Chisq")
```

6.6 Rapport des côtes

Rapport des côtes : On appelle rapport des côtes (ou odds ratio) de 2 valeurs x_* et x_0 de $X = (X_1, \ldots, X_p)$ le réel :

$$RC(x_*, x_0) = \frac{\frac{p(x_*)}{1-p(x_*)}}{\frac{p(x_0)}{1-p(x_0)}} = \frac{p(x_*)(1-p(x_0))}{(1-p(x_*))p(x_0)}.$$

Interprétation : Soient $j \in \{1, \ldots, p\}$ et $e_j = (0, \ldots, 0, 1, 0, \ldots, 0)$ (le 1 se situe à la j-ème composante).

Si X_j augmente d'une unité, alors le rapport des côtes est

$$RC_j = RC(x + e_j, x) = \exp(\beta_j), \qquad x \in \mathbb{R}^p.$$

Par conséquent,

- si $RC_j > 1$, l'augmentation d'une unité de X_j entraîne une augmentation des chances que $\{Y = 1\}$ se réalise,
- si $RC_j = 1$, l'augmentation d'une unité de X_j n'a pas d'impact sur Y,
- si $RC_j < 1$, l'augmentation d'une unité de X_j entraîne une augmentation des chances que $\{Y = 0\}$ se réalise.

Estimateur de RC_j : Un estimateur de RC_j est

$$\widehat{RC}_j = \exp(\widehat{\beta}_j).$$

Avec l'observation de celui-ci, on peut interpréter l'influence de X_j sur $\{Y = 1\}$ en la comparant à 1, comme on l'a fait précédemment.

```
exp(coef(reg))
```

6.7 Intervalles de confiance

Intervalle de confiance pour β_j : Soit $j \in \{0, \ldots, p\}$. Un intervalle de confiance pour β_j au niveau $100(1-\alpha)\%$, $\alpha \in]0,1[$, est la réalisation i_{β_j} de

$$I_{\beta_j} = \left[\widehat{\beta}_j - z_\alpha \widehat{\sigma}(\widehat{\beta}_j),\ \widehat{\beta}_j + z_\alpha \widehat{\sigma}(\widehat{\beta}_j)\right],$$

où z_α est le réel vérifiant $\mathbb{P}(|Z| \geq z_\alpha) = \alpha$, avec $Z \sim \mathcal{N}(0,1)$.

```
confint.default(reg, level = 0.95)
```

Intervalle de confiance pour $p(x)$: Un intervalle de confiance pour $p(x)$, avec $x = (x_1, \ldots, x_p)$, au niveau $100(1-\alpha)\%$, $\alpha \in]0,1[$, est la réalisation $i_{p(x)}$ de

$$I_{p(x)} = \left[\text{logit}^{-1}\left(\text{logit}\,\widehat{p}(x) - z_\alpha \widehat{\sigma}(\text{logit}\,\widehat{p}(x))\right),\ \text{logit}^{-1}\left(\text{logit}\,\widehat{p}(x) + z_\alpha \widehat{\sigma}(\text{logit}\,\widehat{p}(x))\right)\right],$$

où z_α est le réel vérifiant $\mathbb{P}(|Z| \geq z_\alpha) = \alpha$, avec $Z \sim \mathcal{N}(0,1)$.

Pour avoir un intervalle de confiance pour $p(x)$ au niveau 95%, on propose :

```
logitp = predict.glm(reg, data.frame(X1 = 1), se.fit = TRUE)
iclogit = c(logitp$fit - 1.96 * logitp$se.fit,
logitp$fit + 1.96 * logitp$se.fit)
ic = exp(iclogit) / (1 + exp(iclogit))
ic
```

Intervalle de confiance pour RC_j : Soit $j \in \{1, \ldots, p\}$. Un intervalle de confiance pour RC_j au niveau $100(1-\alpha)\%$, $\alpha \in]0,1[$, est la réalisation i_{RC_j} de

$$I_{RC_j} = \left[\exp\left(\widehat{\beta}_j - z_\alpha \widehat{\sigma}(\widehat{\beta}_j)\right),\ \exp\left(\widehat{\beta}_j + z_\alpha \widehat{\sigma}(\widehat{\beta}_j)\right)\right],$$

où z_α est le réel vérifiant $\mathbb{P}(|Z| \geq z_\alpha) = \alpha$, avec $Z \sim \mathcal{N}(0,1)$.

```
exp(confint.default(reg))
```

6.8 Pertinence du modèle

Méthodes : Afin d'étudier la pertinence du modèle de régression logistique, on préconise les méthodes suivantes :

- La "règle du pouce",
- Test de Hosmer-Lemeshow,
- Test des résidus de Pearson,
- Test des résidus de la déviance.

La règle du pouce : L'espérance d'une *var* suivant une loi du Chi-deux est égale à ses degrés de libertés. Par conséquent, si le modèle est pertinent, on doit avoir D proche de son espérance $\nu = n - (p+1)$, soit encore,

$$\frac{D}{\nu} \simeq 1.$$

Si tel est le cas, cela est satisfaisant.

```
deviance(reg) / df.residual(reg)
```

ou alors, on a ces 2 valeurs avec la commande `summay(reg)` ; il est donc aisée de voir si elles sont proches ou pas.

Test de Hosmer-Lemeshow : Pour évaluer la pertinence du modèle de régression logistique, on préconise le test de Hosmer-Lemeshow. La p-valeur associée utilise la loi du Chi-deux : si p-valeur > 0.05, on admet que le modèle est bien adapté aux données.

```
library(ResourceSelection)
hoslem.test(Y, fitted(reg), g = 10)
```

Résidus de Pearson : Pour tout $i \in \{1, \dots, n\}$, on appelle i-ème résidus de Pearson la réalisation de

$$\widehat{\epsilon}_i^{\circ} = \frac{Y_i - \widehat{p}(x_i)}{\sqrt{\widehat{p}(x_i)(1-\widehat{p}(x_i))}}.$$

Celui-ci est notée e_i°.

```
e = residuals(reg, type = "pearson")
```

Loi associée : Si le modèle étudié est pertinent (ou exact), alors la loi limite (ou exacte) de $\sum_{i=1}^{n}(\widehat{\epsilon}_i^{\circ})^2$ est $\chi^2(n-(p+1))$.

Test des résidus de Pearson : L'objectif du test des résidus de Pearson est d'évaluer la pertinence du modèle.

On considère les hypothèses :

$$H_0 : p(x) = \text{logit}^{-1}(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p) \qquad \text{contre}$$

$$H_1 : p(x) \neq \text{logit}^{-1}(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p).$$

On calcule

$$\chi^2_{obs} = \sum_{i=1}^{n}(e_i^{\circ})^2.$$

On considère une *var* $K \sim \chi^2(\nu)$.

Alors la p-valeur associée est

$$\text{p-valeur} = \mathbb{P}(K \geq \chi^2_{obs}).$$

Si p-valeur > 0.05, alors on admet que le modèle est bien adapté aux données.

```
s2 = sum(residuals(reg, type = "pearson")^2)
ddl = df.residual(reg)
pvaleur = 1 - pchisq(s2, ddl)
pvaleur
```

Test des résidus de la déviance : Le test des résidus de la déviance est similaire à celui des résidus de Pearson, mais avec les déviances résiduelles :

pour tout $i \in \{1, \ldots, n\}$, la i-ème déviance résiduelle est la réalisation de

$$D_i = \text{Sign}(Y_i - \widehat{p}(x_i))\sqrt{2\left(Y_i \log\left(\frac{Y_i}{\widehat{p}(x_i)}\right) + (1-Y_i)\log\left(\frac{1-Y_i}{1-\widehat{p}(x_i)}\right)\right)}.$$

et la déviance est la réalisation de

$$D = \sum_{i=1}^{n} D_i^2.$$

Soit χ^2_{obs} cette réalisation et $K \sim \chi^2(\nu)$.

Alors la p-valeur associée est

$$\text{p-valeur} = \mathbb{P}(K \geq \chi^2_{obs}).$$

```
pvaleur = 1 - pchisq(deviance(reg), df.residual(reg))
pvaleur
```

Pseudo R^2 : Plusieurs "pseudo R^2" existent pour la régression logistique (pseudo R^2 de McFadden, de Nagelkerke ...).

Plus ils sont proches de 1, meilleur est le modèle.

Toutefois, ces R^2 sont souvent petits et difficiles à interpréter ; ils sont généralement considérés comme corrects si $R^2 > 0.2$.

```
library(rms)
reg2 = lrm(Y ~ X1 + X2 + X3)
reg2
```

6.9 Détection des valeurs anormales

Objectif : La détection de valeurs anormales dans les données est cruciale car elles peuvent avoir une influence négative dans les estimations et, a fortiori, dans les prévisions (effet levier de la fonction de régression).

Méthodes :

- Méthode des résidus normalisés de Pearson,
- Critère des Distances de Cook.

Résidus normalisés de Pearson : Pour tout $i \in \{1, \dots, n\}$, on appelle i-ème résidus standardisé de Pearson la réalisation $e_i^\star$ de

$$\widehat{\epsilon}_i^\star = \frac{\widehat{\epsilon}_i^o}{\sqrt{1 - [W^{1/2}X(X^tWX)^{-1}X^tW^{1/2}]_{i,i}}}.$$

où $W = \text{diag}\left(\widehat{p}(x_1)(1-\widehat{p}(x_1)), \dots, \widehat{p}(x_n)(1-\widehat{p}(x_n))\right)$.

On appelle résidus standardisés les réels $e_1^\star, \ldots, e_n^\star$.

```
rstandard(reg, type = "pearson")
```

Méthode des résidus normalisés de Pearson : Pour tout $i \in \{1, \ldots, n\}$, si

$$|e_i^\star| > 2,$$

on envisage l'anormalité de la i-ème observation.

```
e = rstandard(reg, type = "pearson")
plot(e)
e[abs(e) > 2]
```

Critère des distances de Cook : Pour tout $i \in \{1, \ldots, n\}$, on défini la distance de Cook de la i-ème observation par

$$d_i = \frac{[W^{1/2}X(X^tWX)^{-1}X^tW^{1/2}]_{i,i}}{(p+1)(1-[W^{1/2}X(X^tWX)^{-1}X^tW^{1/2}]_{i,i})}(e_i^\star)^2.$$

Si

$$d_i > 1,$$

on envisage l'anormalité de la i-ème observation. Le seuil $4/(n-(p+1))$ est parfois utilisé.

```
plot(reg, 4)
cooks.distance(reg)[cooks.distance(reg) > 1]
```

Admettons que les valeurs associées aux individus 8 et 20 soient anormales.

On refait l'analyse sans ces individus avec les commandes R :

```
reg2 = glm(Y ~ X1 + X2 + X3, subset = - c(8, 20), family = binomial)
```

Peu importe la méthode et le résultat, il faut toujours s'assurer auprès du spécialiste de l'étude que une ou plusieurs observations peuvent être retirées des données.

6.10 Sélection de variables

Objectif : Il est intéressant de déterminer la meilleure combinaison des variables $X_1, \ldots, X_p$ qui explique Y.

Or l'approche qui consiste à éliminer d'un seul coup les variables dont les coefficients associés ne sont pas significativement différents de 0 n'est pas bonne ; certaines variables peuvent être corrélées à d'autres ce qui peut masquer leur réelle influence sur Y.

Méthodes : La méthode pas à pas utilisant le AIC ou BIC déjà vue avec la *rlm*, fonctionne aussi avec la régression logistique.

```
library(stats)
reg = glm(Y ~ X1 + X2 + X3, family = binomial)
```

Pour faire l'approche pas à pas avec AIC :

```
step(reg, direction = "both", k = 2)
```

Pour faire l'approche pas à pas avec BIC :

```
step(reg, direction = "both", k = log(length(Y)))
```

Comparaison de 2 modèles : Pour tester l'influence d'une ou plusieurs variables dans le modèle de régression logistique, tout en prenant en considération les autres variables, on peut utiliser le test de la déviance (équivalent à celui de l'ANOVA dans le cas du modèle de *rlm*).

Si p-valeur > 0.05, alors les variables étudiées ne contribuent pas significativement au modèle.

Si on veut tester $H_0 : \beta_2 = \beta_4 = 0$ en sachant qu'il y a les variables $X1$ et $X3$ dans le modèle, on effectue

```
reg1 = glm(Y ~ X1 + X2 + X3 + X4, family = binomial)
reg2 = glm(Y ~ X1 + X3, family = binomial)
anova(reg1, reg2)
```

6.11 Qualité du modèle

Méthodes : Pour évaluer la qualité du modèle de régression logistique, on préconise :

- Le taux d'erreur,
- La courbe ROC.

Prédiction du groupe : On appelle i-ème prédiction du groupe la réalisation $\widetilde{y}_i$ de

$$\widehat{Y}_i = \mathbf{1}_{\{\widehat{p}(x_i)\geq 1-\widehat{p}(x_i)\}} = \begin{cases} 1 & \text{si } \widehat{p}(x_i) \geq \dfrac{1}{2}, \\ 0 & \text{sinon.} \end{cases}$$

```
pred.prob = predict(reg, type = "response")
pred.mod = factor(ifelse(pred.prob > 0.5, "1", "0"))
pred.mod
```

Matrice de confusion : On appelle matrice de confusion la matrice :

$$\begin{aligned} MC &= \begin{pmatrix} \sum_{i=1}^{n} \mathbf{1}_{\{y_i=\widetilde{y}_i=0\}} & \sum_{i=1}^{n} \mathbf{1}_{\{y_i=0\}\cap\{\widetilde{y}_i=1\}} \\ \sum_{i=1}^{n} \mathbf{1}_{\{y_i=1\}\cap\{\widetilde{y}_i=0\}} & \sum_{i=1}^{n} \mathbf{1}_{\{y_i=\widetilde{y}_i=1\}} \end{pmatrix} \\ &= \begin{pmatrix} \text{Nombre de 0 prédit, 0 en réalité} & \text{Nombre de 1 prédit, 0 en réalité} \\ \text{Nombre de 0 prédit, 1 en réalité} & \text{Nombre de 1 prédit, 1 en réalité} \end{pmatrix}. \end{aligned}$$

```
mc = table(Y, pred.mod)
mc
```

Taux d'erreur : On appelle le taux d'erreur (de prédiction) le réel :

$$t = \frac{1}{n}\left(\sum_{i=1}^{n} \mathbf{1}_{\{y_i=0\}\cap\{\widetilde{y}_i=1\}} + \sum_{i=1}^{n} \mathbf{1}_{\{y_i=1\}\cap\{\widetilde{y}_i=0\}}\right).$$

Ce taux est la proportion des modalités prédites qui diffèrent des modalités observées (c'est aussi la somme des 2 valeurs non-diagonales de la matrice de confusion divisée par n).

Plus t est proche de 0, meilleur est la qualité prédictive modèle.

On convient que la qualité prédictive du modèle est mauvaise lorsque $t > 0.5$.

```
t = (mc[1, 2] + mc[2, 1]) / sum(mc)

t
```

Courbe ROC (Receiver Operating Characteristic curve) : Soit $\tau \in [0, 1]$. On appelle i-ème prédiction du groupe au niveau τ la réalisation $\widetilde{y}_i(\tau)$ de

$$\widehat{Y}_i(\tau) = \begin{cases} 1 & \text{si } \widehat{p}(x_i) \geq \tau, \\ 0 & \text{sinon.} \end{cases}$$

À partir de ces prédictions, on défini :

- la fréquence de fausse alarme (”1-specificity”) :

$$ffa(\tau) = \frac{\sum_{i=1}^{n} \mathbf{1}_{\{y_i=0\} \cap \{\widetilde{y}_i(\tau)=1\}}}{\sum_{i=1}^{n} \mathbf{1}_{\{y_i=0\}}}.$$

- la fréquence de bonne détection (”sensitivity”) :

$$fbd(\tau) = \frac{\sum_{i=1}^{n} \mathbf{1}_{\{y_i=1\} \cap \{\widetilde{y}_i(\tau)=1\}}}{\sum_{i=1}^{n} \mathbf{1}_{\{y_i=1\}}}.$$

On appelle courbe ROC la courbe passant par les points :

$$\{(ffa(\tau), fbd(\tau));\ \tau \in [0, 1]\}.$$

Plus la courbe longe les axes $x = 0$ et $y = 1$, meilleur est le modèle.

```
library(Epi)

ROC(form = Y ~ X1, plot = "ROC")
```

(Autrement dit, plus l'aire sous la courbe ROC est proche de 1, meilleur est le modèle).

6.12 Cas des données groupées

Contexte : On suppose que les n individus sont répartis en q groupes $\mathcal{G}_1, \dots, \mathcal{G}_q$:

	$\mathcal{G}_1$	$\mathcal{G}_2$	$\dots$	$\mathcal{G}_q$
Effectif	n_1	n_2	$\dots$	n_q

Ainsi, $n = \sum_{u=1}^{q} n_u$. On n'a pas en notre possession les observations $(y_1, x_{1,1}, \dots, x_{p,1}), \dots, (y_n, x_{1,n}, \dots, x_{p,n})$ de $(Y, X_1, \dots, X_p)$ sur les n individus ; pour chacun des groupes $\mathcal{G}_1, \dots, \mathcal{G}_q$, on dispose du nombre d'individus réalisant $\{Y = 1\}$. Ce nombre est une variable quantitative discrète N.

Données : Les données se présentent généralement sous la forme d'un tableau :

Groupe	X_1	$\dots$	X_p	Effectif	N
$\mathcal{G}_1$	$x_{1,1}$	$\dots$	$x_{p,1}$	n_1	$\sum_{i=1}^{n_1} y_i$
$\mathcal{G}_2$	$x_{1,2}$	$\dots$	$x_{p,2}$	n_2	$\sum_{i=n_1+1}^{n_1+n_2} y_i$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathcal{G}_q$	$x_{1,q}$	$\dots$	$x_{p,q}$	n_q	$\sum_{i=n_{q-1}+1}^{n_{q-1}+n_q} y_i$

Modélisation : On modélise les variables considérées comme des *var* (définies sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$).

Pour tout $g \in \{1, \dots, q\}$,

- $(x_{1,g}, \dots, x_{p,g})$ est une réalisation du vecteur aléatoire réel $(X_1, \dots, X_p)$,
- sachant que $(X_1, \dots, X_p) = (x_{1,g}, \dots, x_{p,g}) = x_g$, $\sum_{i=n_{g-1}+1}^{n_{g-1}+n_g} y_i$ est une réalisation de

$$N_g \sim \mathcal{B}\left(n_g, p(x_g)\right), \qquad p(x_g) = \mathbb{P}(\{Y = 1\} | \{(X_1, \dots, X_p) = x_g\}).$$

Cette loi est utilisée dans tous les outils théoriques, notamment pour définir la vraisemblance associée à $(N_1, \ldots, N_q)$ (et non $(Y_1, \ldots, Y_n)$) :

$$L(\beta, z) = \prod_{g=1}^{q} \binom{n_g}{z_g} p(x_g)^{z_g} (1 - p(x_g))^{n_g - z_g}, \qquad z = (z_1, \ldots, z_g) \in \prod_{g=1}^{q} \{0, \ldots, n_g\}.$$

Objectif : L'objectif est toujours le même : estimer la probabilité (ou proportion) inconnue

$$p(x) = \mathbb{P}(\{Y = 1\} | \{(X_1, \ldots, X_p) = x\}), \qquad x = (x_1, \ldots, x_p),$$

à l'aide des données.

En notant Nb le nombre d'individus, les commandes R correspondantes sont :

```
reg = glm(cbind(Yg, Nb - Yg) ~ X1 + X2 + X3 + X4, family = binomial)
```

ou

```
reg = glm(Yg / n ~ X1 + X2 + X3 + X4, family = binomial, weights = n)
summary(reg)
```

Analyse statistique : Les méthodes présentées dans les sections précédentes sont toujours valables avec les mêmes commandes R.

7 Régression polytomique

7.1 Contexte

Problématique : Dans une population, on souhaite expliquer une variable qualitative Y à m modalités $u_1, \dots, u_m$ à partir de p autres variables $X_1, \dots, X_p$.

Données : Les données dont on dispose sont n observations de $(Y, X_1, \dots, X_p)$ notées

$(y_1, x_{1,1}, \dots, x_{p,1}), \dots, (y_n, x_{1,n}, \dots, x_{p,n})$.

Les données se présentent généralement sous la forme d'un tableau :

Y	X_1	$\dots$	X_p
y_1	$x_{1,1}$	$\dots$	$x_{p,1}$
y_2	$x_{1,2}$	$\dots$	$x_{p,2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y_n	$x_{1,n}$	$\dots$	$x_{p,n}$

Modélisation : On modélise les variables considérées comme des *var* (définies sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$).

Pour tout $i \in \{1, \dots, n\}$,

- $(x_{1,i}, \dots, x_{p,i})$ est une réalisation du vecteur aléatoire réel $(X_1, \dots, X_p)$,
- sachant que $(X_1, \dots, X_p) = (x_{1,i}, \dots, x_{p,i}) = x_i$, y_i est une réalisation d'une *var* Y_i dont la loi est donnée par

$$\mathbb{P}(Y_i = u_k) = \mathbb{P}(\{Y = u_k\} | \{(X_1, \dots, X_p) = x_i\}).$$

Objectif : Pour tout $k \in \{1, \dots, m\}$, un objectif est d'estimer la probabilité inconnue

$$p_k(x) = \mathbb{P}(\{Y = u_k\} | \{(X_1, \dots, X_p) = x\}), \qquad x = (x_1, \dots, x_p),$$

à l'aide des données.

7.2 Régression multinomiale (ou polytomique non-ordonnée)

7.2.1 Contexte

Hypothèse : On suppose que les modalités de Y sont sans lien hiérarchique/ordre.

Notons que le cas $k = 2$, $u_1 = 0$ et $u_2 = 1$ correspond à la régression logistique.

Régression multinomiale : On appelle modèle de régression multinomiale (ou polytomique non-ordonnée) la modélisation : pour tout $k \in \{2, \dots, m\}$,

$$\log\left(\frac{p_k(x)}{p_1(x)}\right) = \beta_0^{(k)} + \beta_1^{(k)} x_1 + \ldots + \beta_p^{(k)} x_p,$$

soit encore,

$$p_k(x) = \frac{\exp(\beta_0^{(k)} + \beta_1^{(k)} x_1 + \ldots + \beta_p^{(k)} x_p)}{1 + \sum_{k=2}^{m} \exp(\beta_0^{(k)} + \beta_1^{(k)} x_1 + \ldots + \beta_p^{(k)} x_p)},$$

où $\beta_0^{(k)}, \dots, \beta_p^{(k)}$ désignent $p+1$ coefficients inconnus.

Notons que, pour $k = 1$, on a

$$p_1(x) = 1 - \sum_{k=2}^{m} p_k(x).$$

Objectif : Notre objectif est d'estimer les coefficients inconnus $\beta_0, \dots, \beta_p$ à partir des données.

7.2.2 Estimation

Vraisemblance : Soit $\beta = (\beta_0^{(1)}, \dots, \beta_p^{(1)}, \dots, \beta_0^{(m)}, \dots, \beta_p^{(m)})$. La vraisemblance du modèle est

$$L(\beta, z) = \prod_{i=1}^{n} \prod_{k=1}^{m} p_k(x_i)^{\mathbf{1}_{\{z_i = u_k\}}}, \qquad z = (z_1, \dots, z_n) \in \{u_1, \dots, u_m\}^n$$

avec $x_i = (x_{1,i}, \dots, x_{p,i})$.

Cela correspond à la vraisemblance d'une loi multinomiale.

Emv : On utilise la méthode du maximum de vraisemblance pour estimer β.

Pour tout $k \in \{2, \ldots, m\}$, on a un *emv* $\widehat{\beta}^{(k)}$ de $\beta^{(k)} = (\beta_0^{(k)}, \ldots, \beta_p^{(k)})$, mis en œuvre avec l'algorithme de Newton-Raphson.

Estimation : Une estimation de $p_k(x)$ avec $k \in \{2, \ldots, m\}$ est la réalisation de

$$\widehat{p}_k(x) = \frac{\exp(\widehat{\beta}_0^{(k)} + \widehat{\beta}_1^{(k)} x_1 + \ldots + \widehat{\beta}_p^{(k)} x_p)}{1 + \sum_{k=2}^{m} \exp(\widehat{\beta}_0^{(k)} + \widehat{\beta}_1^{(k)} x_1 + \ldots + \widehat{\beta}_p^{(k)} x_p)}.$$

On en déduit une estimation de $p_k(x)$ avec $k = 1$:

$$\widehat{p}_1(x) = 1 - \sum_{k=2}^{m} \widehat{p}_k(x).$$

```
library(nnet)
reg = multinom(Y ~ X1 + X2 + X3)
summary(reg)
confint(reg, level = 0.95)
```

Prédiction : Sachant qu'un individu ω vérifie $(X_1, \ldots, X_p) = x$, à partir de $\widehat{p}_k(x)$, on peut faire de la prédiction sur :

- la probabilité que ω satisfait $Y = u_k$ pour tout $k \in \{1, \ldots, m\}$,

```
predict(reg, data.frame(X1 = 1, X2 = 25, X3 = 4), type = "probs")
```

- la modalité de Y la plus probable pour ω.

```
predict(reg, data.frame(X1 = 1, X2 = 25, X3 = 4), type = "class")
```

7.2.3 Significativité du modèle

Test de Wald : Soient $j \in \{0, \ldots, p\}$ et $k \in \{1, \ldots, m\}$. L'objectif du test de Wald est d'évaluer l'influence (ou la contribution) de X_j sur Y.

On considère les hypothèses :

$$H_0 : \beta_j^{(k)} = 0 \qquad \text{contre} \qquad H_1 : \beta_j^{(k)} \neq 0.$$

On calcule la réalisation z_{obs} de

$$Z_* = \frac{\widehat{\beta}_j^{(k)}}{\widehat{\sigma}(\widehat{\beta}_j^{(k)})}.$$

On considère une *var* $Z \sim \mathcal{N}(0,1)$.

Alors la p-valeur associée est

$$\text{p-valeur} = \mathbb{P}(|Z| \geq |z_{obs}|).$$

Si $*$, l'influence de X_j sur Y est significative, si $**$, elle est très significative et si $***$, elle est hautement significative.

```
reg = multinom(Y ~ X1 + X2 + X3)
z = summary(reg)$coeff / summary(reg)$standard.errors
pvaleur = 2 * (1 - pnorm(abs(z), 0, 1))
pvaleur
```

Déviance : La déviance du modèle est la réalisation de

$$D = 2\sum_{i=1}^{n}\sum_{k=1}^{m} \mathbf{1}_{\{Y_i=u_k\}} \log\left(\frac{\mathbf{1}_{\{Y_i=u_k\}}}{\widehat{p}_k(x_i)}\right)$$

avec $x_i = (1, x_{1,i}, \ldots, x_{p,i})$.

Test lr : On considère les hypothèses :

$$H_0 : \beta_1^{(k)} = \beta_2^{(k)} = \ldots = \beta_p^{(k)} = 0 \text{ pour tout } k \in \{1, \ldots, m\} \qquad \text{contre}$$

$$H_1 : \text{il y a au moins un coefficient non nul.}$$

Pour ce faire, on utilise le test du rapport de vraisemblance (lr pour likelihood ratio) (asymptotique). La p-valeur associée utilise la loi du Chi-deux : si $*$, l'influence d'au moins une variable est significative, si $**$, très significative et si $***$, hautement significative.

```
reg = multinom(Y ~ X1 + X2 + X3)

reg0 = multinom(Y ~ 1)

rv = reg0$deviance - reg$deviance

ddl = reg$edf - reg0$edf

pvaleur = 1 - pchisq(rv, ddl)

pvaleur
```

7.2.4 Sélection de variables

On peut également faire de la sélection de variables avec les critères AIC et BIC :

```
library(stats)
```

Pour faire l'approche pas à pas avec AIC :

```
step(reg, direction = "both", k = 2)
```

Pour faire l'approche pas à pas avec BIC :

```
step(reg, direction = "both", k = log(length(Y)))
```

7.2.5 Qualité du modèle

Prédiction de modalités : On appelle i-ème prédiction de modalité la réalisation $\widetilde{y}_i$ de

$$\widehat{Y}_i = u_{\text{Argmax}_{k\in\{1,\dots,m\}}\, \widehat{p}_k(x_i)}.$$

Matrice de confusion : On appelle matrice de confusion la matrice :

$$MC = \left(\sum_{i=1}^{n} \mathbf{1}_{\{y_i=u_k\}\cap\{\widetilde{y}_i=u_{k_*}\}}\right)_{(k,k_*)\in\{1,\dots,m\}^2}.$$

```
pr = predict(reg)

mc = table(Y, pr)

mc
```

Taux d'erreur : On appelle le taux d'erreur (de prédiction) le réel :

$$t = \frac{1}{n} \sum_{k=1}^{m} \sum_{\substack{k_*=1 \\ k_* \neq k}}^{m} \mathbf{1}_{\{y_i = u_k\} \cap \{\widetilde{y}_i = u_{k_*}\}}$$

Ce taux est la proportion des modalités prédites qui diffèrent des modalités observées.

C'est aussi la somme des $n^2 - n$ composantes non-diagonales de la matrice de confusion divisée par n.

Plus t est proche de 0, meilleur est le modèle.

On convient que la qualité du modèle est mauvaise lorsque $t > 0.5$.

```
t = (sum(mc) - sum(diag(mc))) / sum(mc)

t
```

7.3 Régression polytomique ordonnée

Hypothèse : On suppose que les modalités de Y sont ordonnées : le fait que u_{k+1} succède à u_k a du sens.

C'est souvent le cas quand les modalités de Y correspondent à des classes de valeurs.

Modélisation : On modélise les variables considérées comme des *var* (définies sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$).

On suppose qu'il existe une variable Y_* latente telle que :

- Y peut se modéliser comme :

$$Y = \begin{cases} u_1 & \text{si } Y_* \in]a_0, a_1], \\ u_2 & \text{si } Y_* \in]a_1, a_2], \\ \dots & \\ u_m & \text{si } Y_* \in]a_{m-1}, a_m], \end{cases}$$

  où $a_0 = -\infty$, $a_m = \infty$, $a_1, \dots, a_{m-1}$ sont $m-1$ coefficients inconnus,

- une liaison linéaire entre Y_* et $X_1, \dots, X_p$ est envisageable.

On peut modéliser Y_* par une *rlm* :

$$Y_* = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \gamma,$$

où $\beta_0, \ldots, \beta_p$ sont $p+1$ coefficients inconnus et γ est une *var* symétrique (γ et $-\gamma$ suivent la même loi) de densité f_γ et de fonction de répartition F_γ.

Pour tout $i \in \{1, \ldots, n\}$,

- $(x_{1,i}, \ldots, x_{p,i})$ est une réalisation du vecteur aléatoire réel $(X_1, \ldots, X_p)$,
- sachant que $(X_1, \ldots, X_p) = (x_{1,i}, \ldots, x_{p,i}) = x_i$, y_i est une réalisation de la *var* Y.

Régression polytomique ordonnée : On appelle modèle de régression polytomique ordonnée la modélisation : pour tout $k \in \{1, \ldots, m-1\}$,

$$p_k(x) = F_\gamma(a_k - (\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p)) - F_\gamma(a_{k-1} - (\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p)).$$

Objectif : Notre objectif est d'estimer les coefficients inconnus $a_1, \ldots, a_{m-1}, \beta_0, \ldots, \beta_p$ à partir des données.

Pour ce faire, on utilise la méthode du maximum de vraisemblance.

On obtient alors $\widehat{\beta}_0, \ldots, \widehat{\beta}_p$, *emv* de $\beta_0, \ldots, \beta_p$, lequel utilise l'algorithme de Newton-Raphson.

Estimation : Une estimation de $p_k(x)$ est la réalisation de

$$\widehat{p}_k(x) = F_\gamma(\widehat{a}_k - (\widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_p x_p)) - F_\gamma(\widehat{a}_{k-1} - (\widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_p x_p)).$$

```
library(MASS)
reg = polr(Y ~ X1 + X2 + X3, method = "logistic")
summary(reg)
confint(reg, level = 0.95)
```

Prédiction : Sachant qu'un individu ω vérifie $(X_1, \ldots, X_p) = x$, à partir de $\widehat{p}_k(x)$, on peut faire de la prédiction sur :

- la probabilité que ω satisfait $Y = u_k$ pour tout $k \in \{1, \ldots, m\}$,

```
predict(reg, data.frame(X1 = 1, X2 = 25, X3 = 4), type = "probs")
```

- la modalité de Y la plus probable pour ω.

```
predict(reg, data.frame(X1 = 1, X2 = 25, X3 = 4), type = "class")
```

Tests statistiques : On peut mettre en œuvre les tests statistiques usuels.

```
library(car)

Anova(reg)
```

8 Régression de Poisson

8.1 Contexte

Problématique : Dans une population, on souhaite expliquer une variable de comptage à valeurs entières Y à partir de p autres variables $X_1, \dots, X_p$.

Données : Les données dont on dispose sont n observations de $(Y, X_1, \dots, X_p)$ notées $(y_1, x_{1,1}, \dots, x_{p,1}), \dots, (y_n, x_{1,n}, \dots, x_{p,n})$.

Les données se présentent généralement sous la forme d'un tableau :

Y	X_1	$\dots$	X_p
y_1	$x_{1,1}$	$\dots$	$x_{p,1}$
y_2	$x_{1,2}$	$\dots$	$x_{p,2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
y_n	$x_{1,n}$	$\dots$	$x_{p,n}$

Objectif : Pour tout $k \in \mathbb{N}$, un objectif est d'estimer la probabilité inconnue

$$p_k(x) = \mathbb{P}(\{Y = k\} | \{(X_1, \dots, X_p) = x\}), \qquad x = (x_1, \dots, x_p),$$

à l'aide des données.

Modélisation : On modélise les variables considérées comme des *var* (définies sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$).

Pour tout $i \in \{1, \dots, n\}$,

- $(x_{1,i}, \dots, x_{p,i})$ est une réalisation du vecteur aléatoire réel $(X_1, \dots, X_p)$,
- sachant que $(X_1, \dots, X_p) = (x_{1,i}, \dots, x_{p,i}) = x_i$, y_i est une réalisation d'une *var*

$$Y_i \sim \mathcal{P}(\lambda(x_i)), \qquad \lambda(x_i) = \mathbb{E}(Y | \{(X_1, \dots, X_p) = x_i\}).$$

Régression de Poisson : On appelle régression de Poisson la modélisation : $Y \sim \mathcal{P}(\lambda(x))$ avec

$$\log(\lambda(x)) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p,$$

où $\beta_0, \ldots, \beta_p$ désignent $p+1$ coefficients inconnus.

Soit encore,

$$\lambda(x) = \exp\left(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p\right).$$

Objectif : Notre objectif est d'estimer les coefficients inconnus $\beta_0, \ldots, \beta_p$ à partir des données.

Pour ce faire, on utilise la méthode du maximum de vraisemblance.

On obtient alors $\widehat{\beta}_0, \ldots, \widehat{\beta}_p$, *emv* de $\beta_0, \ldots, \beta_p$.

Estimation : Une estimation de $\lambda(x)$ avec $x = (x_1, \ldots, x_p)$ est la réalisation de

$$\widehat{\lambda}(x) = \exp(\widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_p x_p).$$

```
library(stats)
reg = glm(Y ~ X1 + X2 + X3, family = poisson)
summary(reg)
predict.glm(reg, data.frame(X1 = 15, X2 = 12.6, X3 = 4), type = "response")
```

La commande `predict.glm` utilise, par défaut, la transformation log de $\widehat{\lambda}(x)$.

C'est pourquoi il faut préciser `type = "response"` lorsque que l'on veut faire de la prédiction sur $\lambda(x)$ uniquement.

Prédiction : Sachant qu'un individu ω vérifie $(X_1, \ldots, X_p) = x$, à partir de $\widehat{\lambda}(x)$, on peut faire de la prédiction sur

- la probabilité que ω satisfait $Y = k$ pour tout $k \in \mathbb{N}$:

$$\widehat{p}_k(x) = \exp(-\widehat{\lambda}(x)) \frac{(\widehat{\lambda}(x))^k}{k!}.$$

```
lamb = predict.glm(reg, data.frame(X1 = 15, X2 = 12.6, X3 = 4),
type = "response")
probs = dpois(0:100, lamb)
probs
```

- la valeur de Y la plus probable pour ω.

```
which.max(probs)
```

Si cela affiche 9, la valeur la plus probable est $k = 8$; c'est la 9-ème valeur en partant de 0.

Variance / écart-type : En posant $\widehat{\beta} = (\widehat{\beta}_0, \ldots, \widehat{\beta}_p)^t$, on a

$$\widehat{V}(\widehat{\beta}) = \left(-\frac{\partial^2}{\partial \beta^2} \log L(\beta, Y) \right)^{-1} |_{\beta=\widehat{\beta}} = (X^t W X)^{-1},$$

où

$$X = \begin{pmatrix} 1 & x_{1,1} & \cdots & x_{p,1} \\ 1 & x_{1,2} & \cdots & x_{p,2} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{1,n} & \cdots & x_{p,n} \end{pmatrix}$$

et $W = \text{diag}(\widehat{\lambda}(x_1), \ldots, \widehat{\lambda}(x_n))$.

Dans la suite, on posera $\widehat{\sigma}(\widehat{\beta}_j) = \sqrt{\widehat{V}(\widehat{\beta}_j)}$, racine carrée de la j-ème composante du vecteur $\widehat{V}(\widehat{\beta})$.

8.2 Significativité de la régression

Test de Wald : Soit $j \in \{0, \ldots, p\}$. L'objectif du test de Wald est d'évaluer l'influence (ou la contribution) de X_j sur Y.

On considère les hypothèses :

$$H_0 : \beta_j = 0 \qquad \text{contre} \qquad H_1 : \beta_j \neq 0.$$

On calcule la réalisation z_{obs} de

$$Z_* = \frac{\widehat{\beta}_j}{\widehat{\sigma}(\widehat{\beta}_j)}.$$

On considère une *var* $Z \sim \mathcal{N}(0, 1)$.

Alors la p-valeur associée est

$$\text{p-valeur} = \mathbb{P}(|Z| \geq |z_{obs}|).$$

Si $*$, l'influence de X_j sur Y est significative, si $**$, elle est très significative et si $***$, elle est hautement significative.

```
summary(reg)
```

Déviance : La déviance du modèle est la réalisation de

$$D = 2\sum_{i=1}^{n}\left(Y_i \log\left(\frac{Y_i}{\widehat{\lambda}(x_i)}\right) - (Y_i - \widehat{\lambda}(x_i))\right)$$

avec $x_i = (1, x_{1,i}, \ldots, x_{p,i})$.

C'est 2 fois la différence entre les log-vraisemblances évaluées en Y_i et $\widehat{\lambda}(x_i)$.

Test de la déviance : Soit $j \in \{0, \ldots, p\}$. L'objectif du test de la déviance est d'évaluer l'influence (ou la contribution) de X_j sur Y.

La p-valeur associée utilise la loi du Chi-deux : si $*$, l'influence de X_j sur Y est significative, si $**$, elle est très significative et si $***$, elle est hautement significative.

```
anova(reg, test = "Chisq")
```

Test lr : On considère les hypothèses :

$$H_0 : \beta_1 = \beta_2 = \ldots = \beta_p = 0 \qquad \text{contre} \qquad H_1 : \text{il y a au moins un coefficient non nul.}$$

Pour ce faire, on utilise le test du rapport de vraisemblance (lr pour likelihood ratio) (asymptotique). La p-valeur associée utilise la loi du Chi-deux : si $*$, l'influence d'au moins une variable est significative, si $**$, très significative et si $***$, hautement significative.

```
reg = glm(Y ~ X1 + X2, family = poisson)
reg0 = glm(Y ~ 1, family = poisson)
anova(reg0, reg, test = "Chisq")
```

8.3 Intervalles de confiance

Intervalle de confiance pour β_j : Soit $j \in \{0, \dots, p\}$. Un intervalle de confiance pour β_j au niveau $100(1-\alpha)\%$, $\alpha \in]0,1[$, est la réalisation i_{β_j} de

$$I_{\beta_j} = \left[\widehat{\beta}_j - z_\alpha \widehat{\sigma}(\widehat{\beta}_j),\ \widehat{\beta}_j + z_\alpha \widehat{\sigma}(\widehat{\beta}_j)\right],$$

où z_α est le réel vérifiant $\mathbb{P}(|Z| \geq z_\alpha) = \alpha$, avec $Z \sim \mathcal{N}(0,1)$.

```
confint.default(reg, level = 0.95)
```

Intervalle de confiance pour $\lambda(x)$: Un intervalle de confiance pour $\lambda(x)$, avec $x = (x_1, \dots, x_p)$, au niveau $100(1-\alpha)\%$, $\alpha \in]0,1[$, est la réalisation $i_{\lambda(x)}$ de

$$I_{\lambda(x)} = \left[\exp\left(\log\widehat{\lambda}(x) - z_\alpha \widehat{\sigma}(\log\widehat{\lambda}(x))\right),\ \exp\left(\log\widehat{\lambda}(x) + z_\alpha \widehat{\sigma}(\log\widehat{\lambda}(x))\right)\right],$$

où z_α est le réel vérifiant $\mathbb{P}(|Z| \geq z_\alpha) = \alpha$, avec $Z \sim \mathcal{N}(0,1)$.

Un intervalle de confiance pour $\lambda(x)$ au niveau 95% est donné par les commandes R :

```
loglamb = predict.glm(reg, data.frame(X1 = 1), se.fit = TRUE)
icloglamb = c(loglamb$fit - 1.96 * loglamb$se.fit,
loglamb$fit + 1.96 * loglamb$se.fit)
ic = exp(icloglamb)
ic
```

8.4 Pertinence du modèle

Méthodes : Afin d'étudier la pertinence du modèle de régression de Poisson, plusieurs approches sont possibles. On préconise les méthodes suivantes :

- La "règle du pouce",
- Test de Hosmer-Lemeshow,
- Test des résidus de Pearson,
- Test des résidus de la déviance.

La règle du pouce : L'espérance d'une *var* suivant une loi du Chi-deux est égale à ses degrés de libertés. Par conséquent, si le modèle est pertinent, on doit avoir D proche de son espérance, c'est à dire, $\nu = n - (p+1)$, soit

$$\frac{D}{\nu} \simeq 1.$$

Si tel est le cas, cela est rassurant.

```
deviance(reg) / df.residual(reg)
```

ou alors, on a ces 2 valeurs avec la commande `summay(reg)` ; il est donc aisée de voir si elles sont proches ou pas.

Test de Hosmer-Lemeshow : Pour évaluer la pertinence du modèle de régression logistique, on préconise le test de Hosmer-Lemeshow. La p-valeur associée utilise la loi du Chi-deux : si p-valeur > 0.05, on admet que le modèle est bien adapté aux données.

```
library(ResourceSelection)
hoslem.test(Y, fitted(reg), g = 10)
```

Résidus de Pearson : Pour tout $i \in \{1, \ldots, n\}$, on appelle i-ème résidus de Pearson la réalisation de

$$\widehat{\epsilon}_i^{\circ} = \frac{Y_i - \widehat{\lambda}(x_i)}{\sqrt{\widehat{\lambda}(x_i)}}.$$

Celui-ci est notée e_i°.

```
e = residuals(reg, type = "pearson")
```

Loi associée : Si le modèle étudié est pertinent (ou exact), alors la loi limite (ou exacte) de $\sum_{i=1}^{n} (\widehat{\epsilon}_i^{\circ})^2$ est $\chi^2(n-(p+1))$.

Test des résidus de Pearson : L'objectif du test des résidus de Pearson est d'évaluer la pertinence du modèle.

On considère les hypothèses :

$$H_0 : \lambda(x) = \exp\left(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p\right) \qquad \text{contre}$$

$$H_1 : \lambda(x) \neq \exp\left(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p\right).$$

On calcule

$$\chi^2_{obs} = \sum_{i=1}^{n} (e_i^\circ)^2.$$

On considère une *var* $K \sim \chi^2(\nu)$ (avec $\nu = n - (p+1)$).

Alors la p-valeur associée est

$$\text{p-valeur} = \mathbb{P}(K \geq \chi^2_{obs}).$$

Si p-valeur > 0.05, alors on admet que le modèle est bien adapté aux données.

```
s2 = sum(residuals(reg, type = "pearson")^2)
ddl = df.residual(reg)
pvaleur = 1 - pchisq(s2, ddl)
pvaleur
```

Test des résidus de la déviance : Le test des résidus de la déviance est similaire à celui des résidus de Pearson, mais avec les déviances résiduelles :

pour tout $i \in \{1, \ldots, n\}$, la i-ème déviance résiduelle est la réalisation de

$$D_i = \text{Sign}(Y_i - \widehat{\lambda}(x_i)) \sqrt{2 \left(Y_i \log \left(\frac{Y_i}{\widehat{\lambda}(x_i)} \right) - (Y_i - \widehat{\lambda}(x_i)) \right)}$$

et la déviance est la réalisation de

$$D = \sum_{i=1}^{n} D_i^2.$$

Soit χ^2_{obs} cette réalisation et $K \sim \chi^2(\nu)$.

Alors la p-valeur associée est

$$\text{p-valeur} = \mathbb{P}(K \geq \chi^2_{obs}).$$

```
pvaleur = 1 - pchisq(deviance(reg), df.residual(reg))
pvaleur
```

8.5 Détection des valeurs anormales

Objectif : La détection de valeurs anormales dans les données est cruciale car ces valeurs peuvent avoir une influence négative dans les estimations et, a fortiori, dans les prévisions.

Méthodes :

- Méthode des résidus normalisés de Pearson,
- Critère des Distances de Cook.

Résidus normalisés de Pearson : Pour tout $i \in \{1, \dots, n\}$, on appelle i-ème résidus standardisé de Pearson la réalisation $e_i^\star$ de

$$\widehat{\epsilon}_i^\star = \frac{\widehat{\epsilon}_i^o}{\sqrt{1 - [W^{1/2}X(X^tWX)^{-1}X^tW^{1/2}]_{i,i}}}.$$

où $W = \text{diag}(\widehat{\lambda}(x_1), \dots, \widehat{\lambda}(x_n))$.

On appelle résidus standardisés les réels $e_1^\star, \dots, e_n^\star$.

```
rstandard(reg, type = "pearson")
```

Méthode des résidus normalisés de Pearson : Pour tout $i \in \{1, \dots, n\}$, si

$$|e_i^\star| > 2,$$

on envisage l'anormalité de la i-ème observation.

```
e = rstandard(reg, type = "pearson")
plot(e)
e[abs(e) > 2]
```

Critère des distances de Cook : Pour tout $i \in \{1, \dots, n\}$, on défini la distance de Cook de la i-ème observation par

$$d_i = \frac{[W^{1/2}X(X^tWX)^{-1}X^tW^{1/2}]_{i,i}}{(p+1)(1 - [W^{1/2}X(X^tWX)^{-1}X^tW^{1/2}]_{i,i})}(e_i^\star)^2.$$

Si

$$d_i > 1,$$

on envisage l'anormalité de la i-ème observation. Le seuil $4/(n-(p+1))$ est parfois utilisé.

```
plot(reg, 4)
cooks.distance(reg)[cooks.distance(reg) > 1]
```

Admettons que les valeurs associées aux individus 4 et 26 soient anormales.

On refait l'analyse sans ces individus avec les commandes R :

```
reg2 = lm(Y ~ X1 + X2 + X3, subset = - c(4, 26))
```

Peu importe la méthode et le résultat, il faut toujours s'assurer auprès du spécialiste de l'étude que une ou plusieurs observations peuvent être retirées des données.

8.6 Sélection de variables

Objectif : Il est intéressant de déterminer la meilleure combinaison des variables $X_1, \ldots, X_p$ qui explique Y.

Or l'approche qui consiste à éliminer d'un seul coup les variables dont les coefficients associés ne sont pas significativement différents de 0 n'est pas bonne ; certaines variables peuvent être corrélées à d'autres ce qui peut masquer leur réelle influence sur Y.

Méthodes : La méthode pas à pas utilisant le AIC ou BIC déjà vue avec la *rlm*, marche aussi avec la régression de Poisson.

```
library(stats)
reg = glm(Y ~ X1 + X2 + X3, family = poisson)
```

Pour faire l'approche pas à pas avec AIC :

```
step(reg, direction = "both", k = 2)
```

Pour faire l'approche pas à pas avec BIC :

```
step(reg, direction = "both", k = log(length(Y)))
```

Comparaison de 2 modèles : Pour tester l'influence d'une ou plusieurs variables dans le modèle de régression de Poisson, tout en prenant en considération les autres variables, on peut utiliser le test de la déviance (équivalent à celui de l'ANOVA dans le cas du modèle de *rlm*).

Si p-valeur > 0.05, alors les variables étudiées ne contribuent pas significativement au modèle.

Si on veut tester $H_0 : \beta_2 = \beta_4 = 0$ en sachant qu'il y a les variables $X1$ et $X3$ dans le modèle, on effectue

```
reg1 = glm(Y ~ X1 + X2 + X3 + X4, family = poisson)
reg2 = glm(Y ~ X1 + X3, family = poisson)
anova(reg1, reg2)
```

8.7 Dispersion anormale

Dispersion anormale : Par définition de la loi de Poisson, on a

$$\mathbb{E}(Y|\{(X_1,\ldots,X_p) = x\}) = \lambda(x), \qquad \mathbb{V}(Y|\{(X_1,\ldots,X_p) = x\}) = \lambda(x).$$

Ainsi la dispersion des valeurs de Y est égale à sa moyenne.

Si tel n'est pas le cas, la dispersion des valeurs est anormale.

L'impact de cette dispersion anormale est d'exagérer la significativité des coefficients ; leurs estimations ponctuelles restent quasiment inchangées.

Première approche : On peut envisager l'anormalité de la dispersion si

- la déviance D plus grande que $\nu = n - (p+1)$,

```
summary(reg)
```

- en notant $\widetilde{\lambda}(x_i)$ la réalisation de $\widehat{\lambda}(x_i)$, le nuage de points :

$$\left\{\left(\widetilde{\lambda}(x_i), (y_i - \widetilde{\lambda}(x_i))^2\right) ;\ i \in \{1,\ldots,n\}\right\}$$

n'est pas ajustable par une droite.

```
plot(log(fitted(reg)), log((Y - fitted(reg))^2))
```

Test de Cameron et Trivedi : On peut conclure à l'anormalité de la dispersion avec le test de Cameron et Trivedi :

$$H_0 : \mathbb{V}(Y|\{(X_1, \ldots, X_p) = x\}) = \lambda(x) \qquad \text{contre}$$
$$H_1 : \text{il existe } c \neq 0 \text{ tel que } \mathbb{V}(Y|\{(X_1, \ldots, X_p) = x\}) = \lambda(x) + cf(\lambda(x)),$$

pour toutes fonctions linéaires f.

Si p-valeur < 0.05, on admet la dispersion anormale.

```
library(AER)

dispersiontest(reg)
```

Si problème : Si la dispersion anormale est avérée, on peut :

- la corriger,

```
phi = sum(residuals(reg, type = "pearson")^2) / df.residual(reg)

phi

summary(reg, dispersion = phi)
```

- penser à utiliser un autre modèle (comme le modèle de régression binomiale négative qui autorise plus de souplesse dans la dispersion).

```
library(MASS)

reg = glm.nb(Y ~ X1 + X2)
```

8.8 Variable de décalage (*offset*)

Contexte : Dans de nombreux problèmes, la variable de dénombrement Y à expliquer est proportionnelle à une autre variable T.

Il est donc naturel de l'inclure dans le modèle de régression de Poisson.

Données : Les données sont de la forme :

Y	T	X_1	...	X_p
y_1	t_1	$x_{1,1}$	...	$x_{p,1}$
y_2	t_2	$x_{1,2}$	...	$x_{p,2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
y_n	t_n	$x_{1,n}$	...	$x_{p,n}$

Objectif : Pour tout $k \in \mathbb{N}$, un objectif est d'estimer la probabilité inconnue :

$$p_k(t,x) = \mathbb{P}(\{Y = k\} | \{(T, X_1, \ldots, X_p) = (t,x)\}), \qquad x = (x_1, \ldots, x_p), \qquad t \in \mathbb{R},$$

à l'aide des données.

Modélisation : On modélise les variables considérées comme des *var* (définies sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$).

Pour tout $i \in \{1, \ldots, n\}$,

- $(t_i, x_{1,i}, \ldots, x_{p,i})$ est une réalisation du vecteur aléatoire réel $(T, X_1, \ldots, X_p)$,
- sachant que $(X_1, \ldots, X_p) = (x_{1,i}, \ldots, x_{p,i}) = x_i$, y_i est une réalisation d'une *var*

$$Y_i \sim \mathcal{P}\left(\lambda(t_i, x_i)\right), \qquad \lambda(t_i, x_i) = \mathbb{E}(Y | \{(T, X_1, \ldots, X_p) = (t_i, x_i)\}).$$

Variable de décalage : On considère le modèle : $Y \sim \mathcal{P}(\lambda(t,x))$ avec

$$\log\left(\frac{\lambda(t,x)}{t}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p,$$

où $\beta_0, \ldots, \beta_p$ désignent $p+1$ coefficients inconnus.

Soit encore,

$$\lambda(t,x) = t \exp\left(\beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p\right) = \exp\left(\ln(t) + \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p\right).$$

La quantité $\ln(t)$ est appelée variable de décalage (offset).

Objectif : Notre objectif est d'estimer les coefficients inconnus $\beta_0, \ldots, \beta_p$ à partir des données.

Pour ce faire, on utilise la méthode du maximum de vraisemblance.

On obtient alors $\widehat{\beta}_0, \ldots, \widehat{\beta}_p$, *emv* de $\beta_0, \ldots, \beta_p$.

Estimation : Une estimation de $\lambda(t, x)$ avec $(t, x) = (t, x_1, \ldots, x_p)$ est la réalisation de

$$\widehat{\lambda}(t, x) = \exp(\ln(t) + \widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_p x_p).$$

```
reg = glm(Y ~ offset(log(T)) + X1 + X2, family = poisson)
summary(reg)
```

Prédiction : Sachant qu'un individu ω vérifie $(T, X_1, \ldots, X_p) = (t, x)$, à partir de $\widehat{\lambda}(t, x)$, on peut faire de la prédiction sur :

- la probabilité que ω satisfait $Y = k$ pour tout $k \in \mathbb{N}$:

$$\widehat{p}_k(t, x) = \exp(-\widehat{\lambda}(t, x)) \frac{(\widehat{\lambda}(t, x))^k}{k!}.$$

```
lamb = predict.glm(reg, data.frame(X1 = 15, X2 = 12.6, X3 = 4, T = 100),
type = "response")
probs = dpois(0:100, lamb)
probs
```

- la valeur de Y la plus probable pour ω.

```
which.max(probs)
```

9 Modèles de régression à effets mixtes

9.1 Introduction aux modèles de *rlm* à effets mixtes

Problématique : Dans une population, on souhaite expliquer une variable quantitative Y à partir de p autres variables $X_1, \ldots, X_p$.

Données : Les données sont des observations de ces variables. Il y a des mesures répétées : on a q observations/mesures pour chacun des m individus. Ainsi :

- 2 observations issues d'individus différents ne sont pas liées,
- 2 observations issues d'un même individu sont liées.

On dispose alors de $n = q \times m$ observations de $(Y, X_1, \ldots, X_p)$.

Les données se présentent généralement sous la forme d'un tableau :

Individus	Y	X_1	$\ldots$	X_p
ω_1	$y_{1,1}$	$x_{1,1,1}$	$\ldots$	$x_{p,1,1}$
	$y_{1,2}$	$x_{1,1,2}$	$\ldots$	$x_{p,1,2}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$
	$y_{1,q}$	$x_{1,1,q}$	$\ldots$	$x_{p,1,q}$
ω_2	$y_{2,1}$	$x_{1,2,1}$	$\ldots$	$x_{p,2,1}$
	$y_{2,2}$	$x_{1,2,2}$	$\ldots$	$x_{p,2,2}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$
	$y_{2,q}$	$x_{1,2,q}$	$\ldots$	$x_{p,2,q}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
ω_m	$y_{m,1}$	$x_{1,m,1}$	$\ldots$	$x_{p,m,1}$
	$y_{m,2}$	$x_{1,m,2}$	$\ldots$	$x_{p,m,2}$
	$\vdots$	$\vdots$	$\vdots$	$\vdots$
	$y_{m,q}$	$x_{1,m,q}$	$\ldots$	$x_{p,m,q}$

Pour chaque individu, on a fixé à q le nombre d'observations pour simplifier. Dans de nombreuses situations, ce nombre peut varier d'un individu à l'autre.

À cause de la liaison existante entre les observations d'un même individu, le modèle de *rlm* sous les hypothèses standards n'est pas adapté.

Modèle de *rlm* à effets aléatoires sur l'intercept : Il existe $p+1$ coefficients inconnus $\beta_0, \ldots, \beta_p$ tels que

$$Y = (\beta_0 + \xi_{ind}) + \beta_1 X_1 + \ldots + \beta_p X_p + \epsilon,$$

où ϵ est une quantité représentant une somme d'erreurs et ξ_{ind} une quantité représentant des effets aléatoires qui caractérise :

- la dépendance existante entre plusieurs observations du même individu,
- la variabilité des valeurs de $(Y, X_1, \ldots, X_p)$ pour à chaque individu.

Objectif : Un objectif est d'estimer les coefficients inconnus $\beta_0, \ldots, \beta_p$ à l'aide des données afin de prédire la valeur moyenne de Y pour une nouvelle valeur de $(X_1, \ldots, X_p)$.

Modélisation : On modélise les variables considérées comme des *var* (définies sur un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$). Pour tout $(i,u) \in \{1,\ldots,m\} \times \{1,\ldots,q\}$,

- $(x_{1,i,u}, \ldots, x_{p,i,u})$ est une réalisation du vecteur aléatoire réel $(X_1, \ldots, X_p)$,
- sachant que $(X_1, \ldots, X_p) = (x_{1,i,u}, \ldots, x_{p,i,u})$, $y_{i,u}$ est une réalisation de

$$Y_{i,u} = (\beta_0 + \xi_i) + \beta_1 x_{1,i,u} + \ldots + \beta_p x_{p,i,u} + \epsilon_{i,u},$$

où

- $\beta_0, \ldots, \beta_p$ désignent $p+1$ coefficients inconnus,
- $\epsilon_{i,u}$ est une *var* modélisant une somme d'erreurs pour la u-ème observation de ω_i,
- ξ_i est une *var* modélisant un effet aléatoire associé à ω_i.

  Cet effet vient s'ajouter à l'effet fixe β_0 (d'où le nom de "effets mixtes").

Hypothèses : En plus des hypothèses standard du modèle de *rlm* on ajoute/précise :

- $\epsilon_{1,1}, \ldots, \epsilon_{m,q}$ sont *iid* suivant chacune la loi $\mathcal{N}(0, \sigma^2)$, avec $\sigma > 0$ inconnu,

- $\xi_1, \ldots, \xi_m$ sont *iid* suivant chacune la loi $\mathcal{N}(0, v^2)$ avec $v > 0$ inconnu,
- $\epsilon_{1,1}, \ldots, \epsilon_{m,q}$ et $\xi_1, \ldots, \xi_m$ sont indépendantes.

On peut alors remarquer que, par exemple,

$$\mathbb{C}(Y_{1,1}, Y_{1,2}) = \mathbb{V}(\xi_1) = v^2 \neq 0.$$

Grâce à la présence de l'effet aléatoire, le modèle prend donc en compte la dépendance entre $Y_{1,1}$ et $Y_{1,2}$, illustrant ainsi le lien existant entre 2 observations d'un même individu.

On peut utiliser la librairie `lme4` combinée avec la librairie `lmerTest` :

```
library(lme4)
library(lmerTest)
```

En notant `ind` la variable égale à l'individu considéré :

```
reg = lmer(Y ~ X1 + X2 + (1 | ind))
```

On peut alors obtenir les estimations des paramètres inconnus et des tests de significativité en faisant :

```
summary(reg)
```

Des intervalles de confiances sont donnés par :

```
confint(reg, level = 0.95)
```

La valeur prédite moyenne de Y pour la valeur $(X1, X2) = (1.2, 2.2)$ est donnée par les commandes R :

```
predict(reg, data.frame(X1 = 1.2, X2 = 2.2))
```

On peut aussi étudier la validité des hypothèses standards :

```
qqnorm(residuals(reg))
plot(fitted(reg), residuals(reg), xlab = "Fitted", ylab = "Residuals")
```

On peut aussi utiliser la librairie `nlme` :

```
library(nlme)
```

En notant `ind` la variable égale à l'individu considéré :

```
reg = lme(Y ~ X1 + X2, random = ~ 1 | ind)
```

On peut alors obtenir les estimations des coefficients :

```
summary(reg)
```

On peut aussi étudier la validité des hypothèses standards :

```
qqnorm(residuals(reg))
plot(fitted(reg), residuals(reg), xlab = "Fitted", ylab = "Residuals")
```

9.2 Compléments et extensions

Modèle de *rlm* à effets mixtes sur une variable explicative : On peut mettre de l'effet aléatoire sur une des variables explicatives. Il existe $p+1$ coefficients inconnus $\beta_0, \ldots, \beta_p$ tels que

$$Y = \beta_0 + (\beta_1 + \xi_{ind})X_1 + \ldots + \beta_p X_p + \epsilon,$$

où ϵ est une quantité représentant une somme d'erreurs et ξ_{ind} l'effet aléatoire.

```
library(lme4)
library(lmerTest)
reg = lmer(Y ~ X1 + (X2 | ind))
summary(reg)
```

On peut également mettre des effets aléatoires sur l'intercept et les variables explicatives :

```
library(lme4)
library(lmerTest)
reg = lmer(Y ~ X1 + (X2 | ind) + (1 | ind))
summary(reg)
```

On peut également étudier la pertinence de considérer des effets aléatoires en comparant le modèle avec et le modèle sans :

```
library(lme4)
library(lmerTest)
reg1 = lmer(Y ~ X1 + (1 | ind))
reg2 = lm(Y ~ X1)
anova(reg1, reg2)
```

Modèle de *rlm* à effet aléatoire longitudinal : On peut imposer une structure *AR(1)* à l'effet aléatoire, modélisation ainsi l'évolution d'un phénomène pour un même individu :

```
library(nlme)
reg = lme(Y ~ X1, random = ~ 1 | id,
correlation = corAR1(0.8, form = ~ 1 | id))
summary(reg)
```

Autres modèles de régression à effets mixtes : Le principe d'ajouter de l'effet aléatoire s'applique au modèle de régression logistique (donc $Y \in \{0, 1\}$).

Les commandes R associées sont :

```
library(lme4)
library(lmerTest)
reg = glmer(Y ~ X1 + (X2 | ind) + (1 | ind), family = binomial)
summary(reg)
```

Une extension au modèle de régression polytomique est aussi possible :

```
library(ordinal)
reg = clmm(Y ~ X1 + (X2 | ind) + (1 | ind))
summary(reg)
```

De même pour la régression de Poisson :

```
library(lme4)

library(lmerTest)

reg = glmer(Y ~ X1 + (X2 | ind) + (1 | ind), family = poisson)

summary(reg)
```

10 Jeux de données

Adresse : Pour illustrer les concepts du cours, onze jeux de données sont disponibles :

http://www.math.unicaen.fr/~chesneau/Etude0.txt

http://www.math.unicaen.fr/~chesneau/Etude1.txt

http://www.math.unicaen.fr/~chesneau/Etude2.txt

http://www.math.unicaen.fr/~chesneau/Etude3.txt

http://www.math.unicaen.fr/~chesneau/Etude4.txt

http://www.math.unicaen.fr/~chesneau/Etude5.txt

http://www.math.unicaen.fr/~chesneau/Etude6.txt

http://www.math.unicaen.fr/~chesneau/Etude7.txt

http://www.math.unicaen.fr/~chesneau/Etude8.txt

http://www.math.unicaen.fr/~chesneau/Etude9.txt

http://www.math.unicaen.fr/~chesneau/Etude10.txt

Des propositions d'analyses sont présentées dans le document suivant :

http://www.math.unicaen.fr/~chesneau/etudes-reg.pdf

Exemple : Ci-dessous, l'exemple d'un début de traitement de données en R :

```
w = read.table("http://www.math.unicaen.fr/~chesneau/Etude2.txt",
header = T)
attach(w)
str(w)
reg = lm(Y ~ X1 + X2)
summary(reg)
par(mfrow = c(2, 2))
plot(reg, 1:4)
```

11 Annexe : *emv*

11.1 Méthode

Échantillon : On a n *var iid* $Y_1, \dots, Y_n$ suivant la même loi qu'une *var* Y.

Loi de probabilité de Y : La loi de Y est caractérisée par une fonction

$$f(\theta, z), \qquad \theta = (\theta_0, \dots, \theta_p), \qquad z \in \mathbb{R}.$$

Si Y est discrète, alors $f(\theta, z) = \mathbb{P}_\theta(Y = z)$, si Y est à densité, alors $f(\theta, z)$ est la densité associée.

Vraisemblance : La vraisemblance est

$$L(\theta, z) = \prod_{i=1}^{n} f(\theta, z_i), \qquad z = (z_1, \dots, z_n) \in \mathbb{R}^n.$$

Log-vraisemblance : La log-vraisemblance est

$$\ell(\theta) = \ln(L(\theta, z)).$$

Emv : En posant $Y = (Y_1, \dots, Y_n)$, l'*emv* est

$$\widehat{\theta} = \underset{\theta \in \mathbb{R}^{p+1}}{\operatorname{Argmax}}\, L(\theta, Y), \qquad (\text{ou } \widehat{\theta} = \underset{\theta \in \mathbb{R}^{p+1}}{\operatorname{Argmax}}\, \ell(\theta, Y)).$$

En général, on obtient $\widehat{\theta}$ en étudiant les θ qui annule les dérivées premières de la log-vraisemblance.

Matrice d'information de Fisher : On appelle matrice d'information de Fisher la matrice

$$I(\theta) = \left(-\mathbb{E}\left(\frac{\partial^2}{\partial \theta_i \partial \theta_j} \ell(\theta, Y) \right) \right)_{(i,j) \in \{0, \dots, p\}^2}.$$

11.2 Résultats asymptotiques

On a les convergences en loi suivantes :

Statistique	$\widehat{\theta}$	$(\widehat{\theta}-\theta)^t I(\theta)(\widehat{\theta}-\theta)$	$-2(\ell(\theta,Y)-\ell(\widehat{\theta},Y))$
Loi limite	$\mathcal{N}\left(0, I(\theta)^{-1}\right)$	$\chi^2(p+1)$	$\chi^2(p+1)$

11.3 Test global

On considère le test global :

$$H_0 : \ \theta = \theta_0 \qquad \text{contre} \qquad H_1 : \ \theta \neq \theta_0$$

Les statistiques utilisées pour mettre en œuvre ce test statistique sont :

Statistique (sous H_0)	de Wald : $(\widehat{\theta}-\theta_0)^t I(\theta_0)(\widehat{\theta}-\theta_0)$	de la vraisemblance : $-2(\ell(\theta_0,Y)-\ell(\widehat{\theta},Y))$
Loi limite	$\chi^2(p+1)$	$\chi^2(p+1)$

11.4 Test partiel

On suppose que le vecteur θ se divise en 2 parties : α_1 de k composantes et β_2 de $p+1-k$ composantes.

Ainsi,

$$\theta = (\alpha_1, \beta_2)$$

On considère le test partiel :

$$H_0 : \ \alpha_1 = \alpha_0 \qquad \text{contre} \qquad H_1 : \ \alpha_1 \neq \alpha_0,$$

avec α_0 un vecteur à k composantes.

On considère alors l'emv $\widehat{\theta} = (\widehat{\alpha}_1, \widehat{\beta}_2)$ et

$$\widetilde{\beta}_2 = \underset{\beta_2 \in \mathbb{R}^{p+1-k}}{\text{Argmax}} \ L((\alpha_0, \beta_2), Y).$$

Les statistiques utilisées pour mettre en œuvre ce test statistique sont :

Statistique (sous H_0)	de Wald : $(\widehat{\alpha}_1 - \alpha_0)^t \widehat{\mathbb{V}}_k (\widehat{\alpha}_1)^{-1} (\widehat{\alpha}_1 - \alpha_0)$	de la vraisemblance : $-2(\ell((\alpha_0, \widetilde{\beta}_2), Y) - \ell((\widehat{\alpha}_1, \widehat{\beta}_2), Y))$
Loi limite	$\chi^2(k)$	$\chi^2(k)$

11.5 Algorithme de Newton-Raphson et *emv*

L'algorithme de Newton-Raphson nous permettant d'approcher l'*emv* est caractérisé par la suite :

$$\theta^{(m+1)} = \theta^{(m)} - \left(\frac{\partial^2}{\partial \theta_i \partial \theta_j} \ell(\theta^{(m)}, Y) \right)^{-1}_{(i,j) \in \{0,\dots,p\}^2} \left(\frac{\partial}{\partial \theta_i} \ell(\theta^{(m)}, Y) \right)_{i \in \{0,\dots,p\}}.$$

Index

mcg, 43

AIC, 38
Algorithme de Gauss-Newton, 62
Algorithme de Newton-Raphson, 64, 121
Approche en arrière, 36
Approche en avant, 37
Approche exhaustive, 35
Approche pas à pas, 38
AR(1), 23

BIC, 38

Coefficients de détermination, 13
Coefficients de détermination ajusté, 13
Corrélogramme, 22
Corrélogramme partiel, 22
Courve ROC, 85
Cp de Mallows, 38

Degrés de significativité, 15
Dispersion anormale, 106
Distances de Cook, 29, 82, 104
Données groupées, 46, 86
Déviance, 76

Effets aléatoires, 112
Effets mixtes, 112
Emco, 11
Emv, 119
Estimateur de Nadaraya-Watson, 66
Estimateur LASSO, 34
Estimateur par splines, 67
Estimateur Ridge, 33
Estimateurs sandwich, 48

Hétéroscédasticité, 45

Interactions, 40

Lien Cauchit, 73
Lien cloglog, 73
Lien probit, 72

Matrice de confusion, 84
Mesures répétées, 111
Modèles de *rlm* à effets mixtes, 111
Modèles mixtes, 111
Moindres carrés quasi-généralisés (*mcqg*), 48
Méthode de Box-Tidwell, 61
Méthode de Cochrane-Orcutt, 54
Méthode de Glejser, 25
Méthode des résidus normalisés de Pearson, 104
Méthode des résidus standardisés, 29

Observations influentes, 31

QQ plot, 25

Rapport des côtes, 77

Règle de Klein, 32

Règle du pouce, 79, 102

Régression de Poisson, 97

Régression linéaire multiple (*rlm*), 9

Régression logistique, 69

Régression multinomiale, 90

Régression non-linéaire, 57

Régression polynomiale, 58

Régression polytomique, 89

Régression polytomique ordonnée, 94

Régression robuste, 30

Résidus, 19

Résidus normalisés de Pearson, 81

Résidus partiels, 59

Résidus standardisés, 19

Sélection de variables, 35

Taux d'erreur, 84, 94

Test ANOVA, 39

Test de Bartlett, 41

Test de Breusch-Pagan, 24

Test de Cameron et Trivedi, 107

Test de Chow, 34

Test de Durbin-Watson, 24, 52

Test de Hosmer-Lemeshow, 79, 102

Test de la déviance, 76, 100

Test de Levene, 41

Test de Ljung-Box, 23

Test de Rainbow, 21

Test de Shapiro-Wilk, 26

Test de Student, 15

Test de Wald, 75, 91, 99

Test de White, 24

Test des résidus de la déviance, 80, 103

Test des résidus de Pearson, 80, 102

Test du portemanteau, 23

Test global, 120

Test global de Fisher, 16

Test lr, 76, 100

Test partiel, 120

Transformation de Yeo et Johnson, 28

Transformation logit, 70

Transformation puissance de Box-Cox, 28, 61

Transformations polynomiales, 61

Variable de décalage, 108

Variable latente, 71

Variables qualitatives, 39

Vif, 32