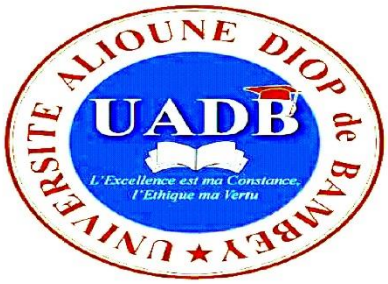


03 DÉCEMBRE 2020

DEVOIR DE MODÈLE LINÉAIRE GÉNÉRALISÉ

CHEIKH OMAR BA ET AMADOU MAMOUDOU LY
UNIVERSITÉ ALIOUNE DIOP

Exercice 1:

On considère le modèle de régression $y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \varepsilon_i \;\; 1 \leq i \leq n$

que l'on écrit sous la forme $Y = X\beta + \varepsilon$. Les $x_{i,j}$sont des variables exogènes du modèle, les ε_i sont des variables aléatoires indépendantes, de loi normale centrée admettant la même variance σ^2. On a observé :

$$X'X = \begin{pmatrix} 30 & 20 & 0 \\ 20 & 20 & 0 \\ 0 & 0 & 10 \end{pmatrix}, \; X'Y = \begin{pmatrix} 15 \\ 20 \\ 10 \end{pmatrix}, \;\; Y'Y = 59.5$$

1. Déterminons *n*, la moyenne des $x_{i,2}$, le coefficient de corrélation des $x_{i,1}$et des $x_{i,2}$.
   a) Déterminons n

$$X'X = \begin{pmatrix} n & \sum x_{i,1} & \sum x_{i,2} \\ \sum x_{i,1} & \sum x_{i,1}^2 & \sum x_{i,1}x_{i,2} \\ \sum x_{i,2} & \sum x_{i,1}x_{i,2} & \sum x_{i,2}^2 \end{pmatrix} = \begin{pmatrix} 30 & 20 & 0 \\ 20 & 20 & 0 \\ 0 & 0 & 10 \end{pmatrix}$$

   Par identification: n = 30

   b) La Moyenne de $x_{i,2}$

$$\bar{x}_2 = \frac{1}{n}\sum_{i=1}^{n} x_{i_1,2} = \frac{1}{30} \times 0 = 0 \Longrightarrow$$

$$\boxed{\bar{x}_2 = 0}$$

   c) Le coefficient de corrélation x_1 et x_2 :

$$r(x_1, x_2) = \frac{Cov(x_1, x_2)}{\sqrt{V(x_1)}\sqrt{V(x_2)}} = \frac{\sum x_{i,1}x_{i,2} - n\bar{x}_1\bar{x}_2}{\sqrt{\sum x_{i,1}^2 - n\bar{x}_1^2}\sqrt{\sum x_{i,2}^2 - n\bar{x}_2^2}} = 0$$

$$\boxed{r(x_1, x_2) = 0}$$

2. Estimer β_0, β_1, β_2, σ^2 par la méthode des moindres carrés ordinaires.
   On sait que :

$$\hat{\beta}_{MC0} = (X'X)^{-1}X'Y$$

   Calculons d'abord l'inverse de $X'X$:

$$X'X = \frac{1}{det(X'X)} \;\; {}^{t_{Com}}(X'X) = \begin{pmatrix} 0{,}1 & -0{,}1 & 0 \\ -0{,}1 & 0{,}15 & 0 \\ 0 & 0 & 0{,}1 \end{pmatrix}$$

$$\hat{\beta}_{MC0} = \begin{pmatrix} 0,1 & -0,1 & 0 \\ -0,1 & 0,15 & 0 \\ 0 & 0 & 0,1 \end{pmatrix} \begin{pmatrix} 15 \\ 20 \\ 10 \end{pmatrix} = \begin{pmatrix} -0,5 \\ 1,5 \\ 1 \end{pmatrix}$$

Alors, on a :

$$\hat{\beta}_0 = -0,5\,, \hat{\beta}_1 = 1,5 \; et \; \hat{\beta}_2 = 1$$

Un estimateur non biaisé $\hat{\sigma}^2 \; de \; \sigma^2$ s'écrit :

$$\hat{\sigma}^2 = \frac{\|Y - X\hat{\beta}\|^2}{n-p-1} = \frac{\|Y\|^2 - \|X\hat{\beta}\|^2}{27}$$ ce qui s'écrit encore :

$$\hat{\sigma}^2 = \frac{Y'Y - Y'X(X'X)^{-1}X'Y}{27} = \frac{59,5 - 32,5}{27} = 1$$

$$\Rightarrow \quad \hat{\sigma}^2 = 1$$

3. Calculons pour β_1 un intervalle de confiance à 95% et tester $\beta_2 = 0,8$ a niveau 10%

Puisqu'on que :

$$\frac{\hat{\beta}_2 - \beta_2}{\hat{\sigma}_2} = \frac{\hat{\beta}_2 - \beta_2}{\hat{\sigma}\sqrt{(X'X)^{-1}_{2,2}}} \hookrightarrow t_{n-3} = t_{27}$$

On en déduit qu'un intervalle de confiance à 95% pour β_2 est :

$$I(\beta_1) = [\hat{\beta}_1 - t_{27}(0,975)\hat{\sigma}\sqrt{(X'X)^{-1}_{1,1}}\,;\, \hat{\beta}_1 + t_{27}(0,975)\hat{\sigma}\sqrt{(X'X)^{-1}_{1,1}}]$$

C'est-à-dire :

$$I(\beta_2) \approx [1,5 - 2,05\sqrt{0,15}\,;\, 1,5 + 2,05\sqrt{0,15}] \approx [0,71\,;\, 2,29].$$

Pour tester l'hypothèse $H_0 : \beta_2 = 0,8$ contre $H_1 : \beta_2 \neq 0,8$ au niveau 10%, on calcule de même un intervalle de confiance à 90% de β_2 :

$$I(\beta_2) = [\hat{\beta}_2 - t_{27}(0,95)\hat{\sigma}\sqrt{(X'X)^{-1}_{2,2}}\,;\, \hat{\beta}_2 + t_{27}(0,95)\hat{\sigma}\sqrt{(X'X)^{-1}_{2,2}}]$$

Ce qui donne :

$$I(\beta_2) \approx [1 - 1,70\sqrt{0,1}\,;\, 1 - 1,70\sqrt{0,1}] \approx [0,46\,;\, 1,54],$$

Donc on accepte au niveau 10% l'hypothèse selon laquelle $\beta_2 = 0,8$.

4. Tester $\beta_0 + \beta_1 = 3$ contre $\beta_0 + \beta_1 \neq 3$ au niveau 5%

On sait que :

$$\frac{(\widehat{\beta}_1 + \widehat{\beta}_2 - 3) - (\beta_1 + \beta_2 - 3)}{\hat{\sigma}_{\widehat{\beta_2}+\beta_2 - 3}} \sim t_{27}$$

Avec :

$$\hat{\sigma}_{\widehat{\beta}_1+\widehat{\beta}_2} = \sqrt{\hat{\sigma}_1^2 + 2Cov(\widehat{\beta}_1 + \widehat{\beta}_2) + \hat{\sigma}_2^2}$$
$$= \hat{\sigma}\sqrt{(X'X)_{1,1}^{-1} + 2(X'X)_{1,2}^{-1} + (X'X)_{2,2}^{-1}}$$

AN :

$$\hat{\sigma}_{\widehat{\beta}_1+\widehat{\beta}_2} = 1 \times \sqrt{0{,}15 + 2 \times (0) + 0{,}1}$$

c'est-à-dire :

$\hat{\sigma}_{\widehat{\beta}_1+\widehat{\beta}_2} = 0{,}5$. Donc un intervalle de confiance à 95% pour $\beta_1 + \beta_2$ est :

$$I(\beta_1 + \beta_2) = [2.5 - 0{,}5t_{27}(0{,}975) ; 2.5 + 0{,}5t_{27}(0{,}975)]$$
$$\approx [1{,}47 ; 3{,}53]$$

Par conséquent, au niveau 5% , on accepte $H_0: \beta_0 + \beta_1 = 3$ contre $H_1: \beta_0 + \beta_1 = 3$.

5. Calculer $\bar{y}$ et déduire le coefficient de détermination ajusté R^2

La moyenne empirique des y_i se déduit de la première composante du vecteur $X'Y$, donc $\bar{y} = \frac{15}{30} = 0{,}5$. Par définition, le coefficient de détermination ajusté R^2 vaut :

$$R^2 = 1 - \frac{n-1}{n-p}\frac{\|Y - \widehat{Y}\|^2}{\|Y - \bar{y}\ 1\|^2} = 1 - (n-1)\frac{\hat{\sigma}^2}{\|Y - \bar{y}\ 1\|^2}$$

Donc :

$$R^2 = 1 - \frac{29}{Y'Y - 30\bar{y}^2} = 1 - \frac{29}{59{,}5 - 30 \times (0{,}5) \times (0{,}5)} \approx 0{,}44$$

$$R^2 \approx 0{,}44$$

6. Construire un intervalle de prévision à 95% de y_{n+1} si $x_{n+1,1} = 3$ et $x_{n+1,1} = 0{,}5$.

Si nous remplaçons dans $y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \varepsilon_i$ ⇨

$$\hat{y}_{n+1} = \hat{\beta}_0 - \hat{\beta}_1 x_{n+1,1} + \hat{\beta}_2 x_{n+1,2}$$

Cela donne :

$$\hat{y}_{n+1} = -0{,}5 + 1{,}5 \times 3 + 1 \times 0{,}5 = 4{,}5 = \frac{9}{2}$$

$$\hat{y}_{n+1}\hat{y}_{n+1} = \frac{9}{2}$$

et un intervalle de prévision à 95% pour y_{n+1}est :

$$IC(y_{n+1}) = [\hat{y}_{n+1} \pm t_{27}\,(0{,}975)\sqrt{1 + (1\ 3\ 0{,}5)\begin{pmatrix} 0{,}1 & -0{,}1 & 0 \\ -0{,}1 & 0{,}15 & 0 \\ 0 & 0 & 0{,}1 \end{pmatrix}\begin{pmatrix} 1 \\ 3 \\ 0{,}5 \end{pmatrix}}]$$

ce qui donne numériquement :

$$IC(y_{n+1}) = [1{,}69\ ;\ 7{,}31]$$

Exercice 2:

Le fichier "data.txt" contient les valeurs de 5 variables : Y , X_1, X_2, X_3 et X_4.

Notre base de données est composée de cinq colonnes, dont la première colonne représente la variable de sortie Y et les quatre autres colonnes restantes représentent respectivement les variables explicatives X_1, X_2, X_3 et X_4.

Comme nous sommes face à un problème de régression linéaire, nous aurons une variable à expliquer qui représente Y et une ou plusieurs variables explicatives $(X_1, \ldots\ldots, X_4)$.

Linear nous dit que notre modèle pour Y est une combinaison linéaire des prédicteurs X.

La régression signifie simplement que nous essayons de mesurer la relation entre une variable de réponse et (une ou plusieurs) variables prédictives. Dans le cas du SLR, la réponse et le prédicteur sont *des* variables *numériques*.

Nous allons de ce pas tenter de fournir des réponses aux questions posées.

1. Créer le vecteur Y contenant la variable que nous voulons modéliser et la matrice X contenant les 4 variables explicatives $X_1, \ldots\ldots, X_4$.

Nous allons charger la base grâce au bout de code suivant :

```
###On charge la base de donnée
dt=read.table(file.choose() ,sep=" ",header =TRUE)
dt
```

Créons-Y et la matrice X

```
####la matrice X contenant les 4 variables explicatives X1, . . . ,X4
X=as.matrix(dt[,2:5])
X
```

Représentons Y en fonction de chacune des autres variables : observe-t-on des liens linéaires ?

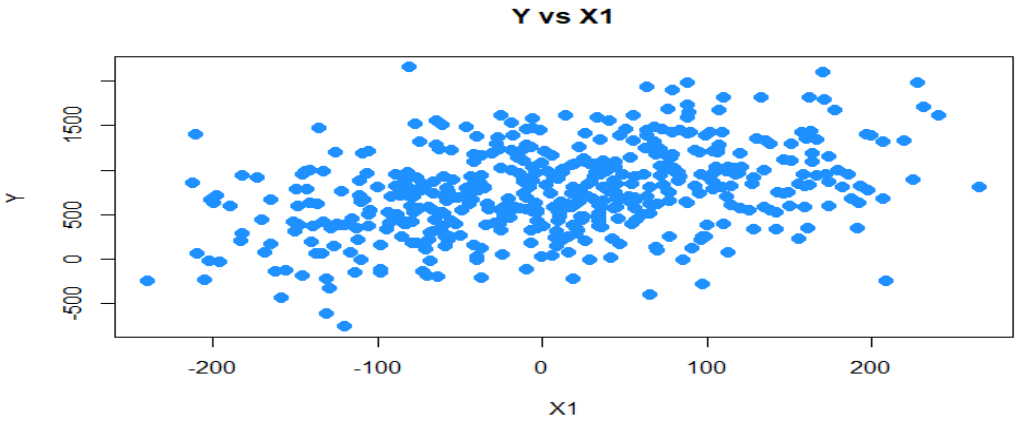
Ensuite nous allons opérer une représentation entre Y et chaque colonne (X_1, X_2, X_3, X_4) de la matrice X.

- Y *et* X_1

##Code

```
### Je Représente Y en fonction de chacune des autres variables
##### Y en fonction X1
plot(Y ~ X1, data = dt,
     xlab = "X1",
     ylab = "Y",
     main = "Y vs X1",
     pch = 20,
     cex = 2,
     col = "dodgerblue")
```

##Sortie



Figur1 : Nuage de point de Y et X_1

- $Y\ et\ X_2$

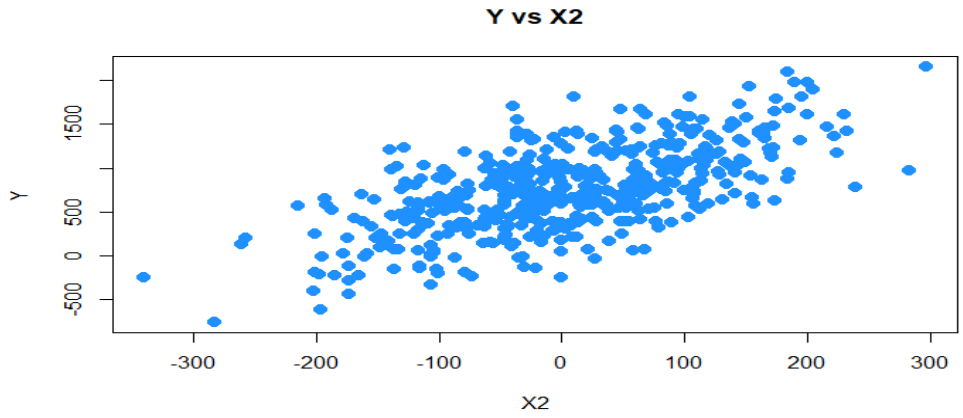
```
plot(Y ~ X3, data = dt, xlab = "X3",
    ylab = "Y",  main = "Y vs X3",
    pch = 20,  cex = 2,
    col = "dodgerblue")
```

Il faut remarquer que le code reste relativement le même, il n'y que :

Y ~ X(X_1, X_2, X_3, X_4) , xlab = "X", et main = "Y vs X3"

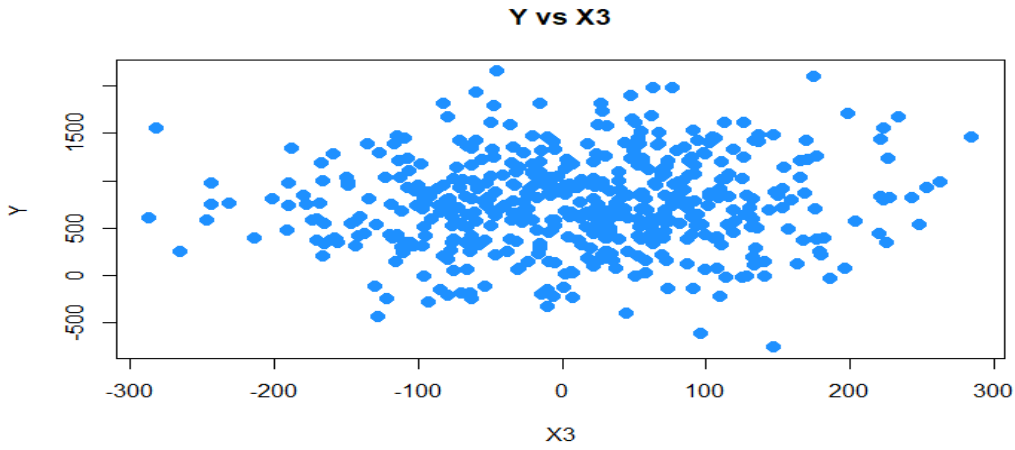
Le xlab c'est pour préciser ce qui sera en abscisse, alors ylab c'est l'opposé.

- pch: des valeurs numériques (de 0 à 25) ou des caractères ("+", ".", ";", etc) spécifiant les symboles de points (ou formes).
- cex: valeurs numériques indiquant la taille du point.
- col: nom de couleur pour les points.



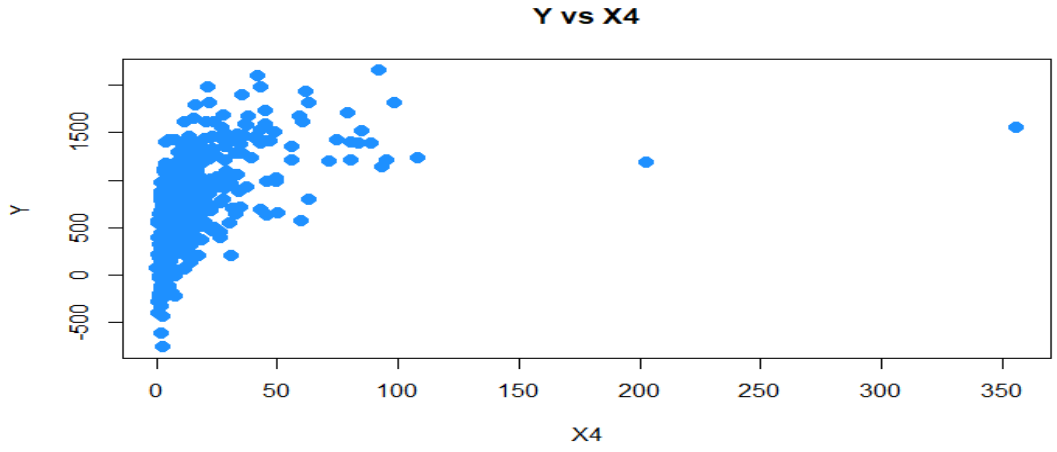
Figur2 : Nuage de point de Y et X_2

❖ $Y\ et\ X_3$



Figur3 : Nuage de point de Y et X2

❖ $Y\ et\ X_4$



Figur4 : Nuage de point de Y et X2

Nous remarquons qu'avec le nuage de la figure 4 on a des points qui s'éloignent complétement des autres.

On observe un lien linéaire avec les variables X_1, X_2, X_3., ce qui n'est pas le cas avec X4.

2. Réaliser la régression linéaire de Y sur l'ensemble des variables

Pour répondre à cette question on va se servir de la fonction « **lm** », comme suit :

```
50  #"2
51  ####Réalisation de la régression linéaire de Y sur l'ensemble des variables
52  regLm1=lm(Y~X, data = dt)
```

Que vaut le R^2 ?

La commande « summary() » permet de répondre à cette question.

```
> summary(regLm1)

Call:
lm(formula = Y ~ X, data = dt)

Residuals:
    Min      1Q  Median      3Q     Max
-1920.2  -113.0    47.3   162.6   309.9

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 570.58133   11.61731  49.115   <2e-16 ***
X1            2.10135    0.09920  21.182   <2e-16 ***
X2            3.11306    0.09700  32.092   <2e-16 ***
X3            0.23003    0.09659   2.381   0.0176 *
X4            9.04170    0.39660  22.798   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 215 on 495 degrees of freedom
Multiple R-squared:  0.7972,    Adjusted R-squared:  0.7956
F-statistic: 486.5 on 4 and 495 DF,  p-value: < 2.2e-16
```

Certains coefficients sont-ils non significatifs ? La régression paraît-elle acceptable ?

Tous les coefficients sont significatifs car la p-value est inférieure à α.

La régression paraît-elle acceptable ?

La régression est acceptable car le R^2 est largement supérieur à 0,5.

3. Pour *i* allant de 1 à 4, réalisons la régression de *Y* sur les variables

   $X_j = 1,\ldots..,4, j \neq i$ et représenter les résidus en fonction de X_i . Des liens linéaires plus nets apparaissent-ils ? Voit-on d'autres liens ?

Si nous considérons notre modèle comme « Réponse = Prédiction + Erreur », nous pouvons alors l'écrire comme :

$$y = \hat{y} + e$$

Nous définissons ensuite un résidu comme étant la valeur observée moins la valeur prédite.

$$e_i = y_i - \hat{y}_i$$

Ceci peut être réalisé à partir du code suivant:

##Code

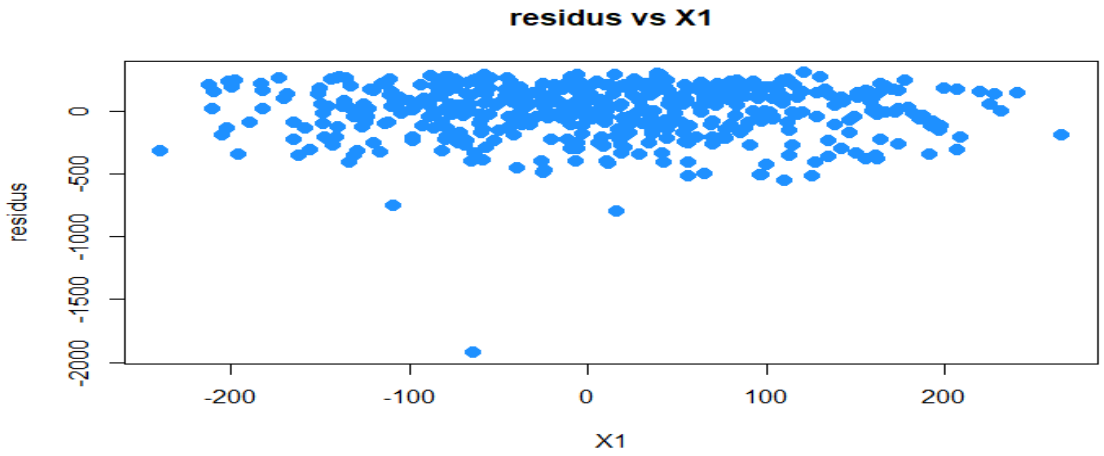
```r
#"3
#####réalisation la régression de Y sur les variables {Xj j =1, . . . . , 4, j /= i}
regLm2=lm(Y~X1+X2+X3+X4,data = dt)
summary(regLm2)
e=resid(regLm2)
#### la représentation  les résidus en fonction de Xi
##### e en fonction X1
plot(X1,e,
     xlab = "X1",
     ylab = "residus",
     main = "residus vs X1",
     pch = 20,
     cex = 2,
     col = "dodgerblue")

##### e en fonction X2
plot(X2,e,
     xlab = "X2",
     ylab = "residus",
     main = "residus vs X2",
     pch = 20,
     cex = 2,
     col = "dodgerblue")

##### e en fonction X3
plot(X3,e,
     xlab = "X3",
     ylab = "residus",
     main = "residus vs X3",
     pch = 20,
     cex = 2,
     col = "dodgerblue")
##### e en fonction X4
plot(X4,e,
     xlab = "X4",
     ylab = "residus",
     main = " residus vs X4",
     pch = 20,
     cex = 2,
     col = "dodgerblue")
```

##Sortie

- $\varepsilon \; et \; X_1$



Figur5 : Nuage de point de $\varepsilon \; et \; X_1$

- $\varepsilon \; et \; X_2$

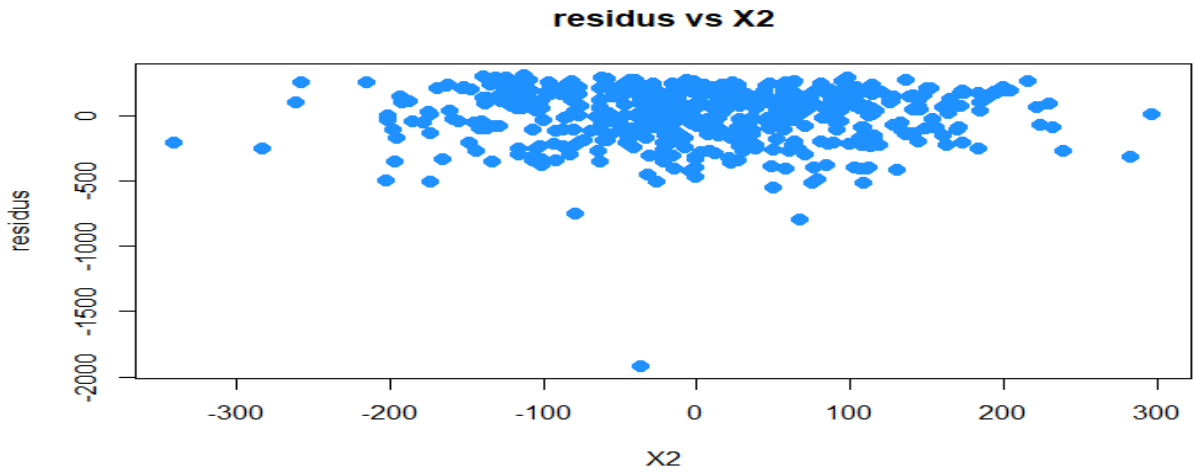


Figure6 : Nuage de point de ε et X_1

- ε et X_3

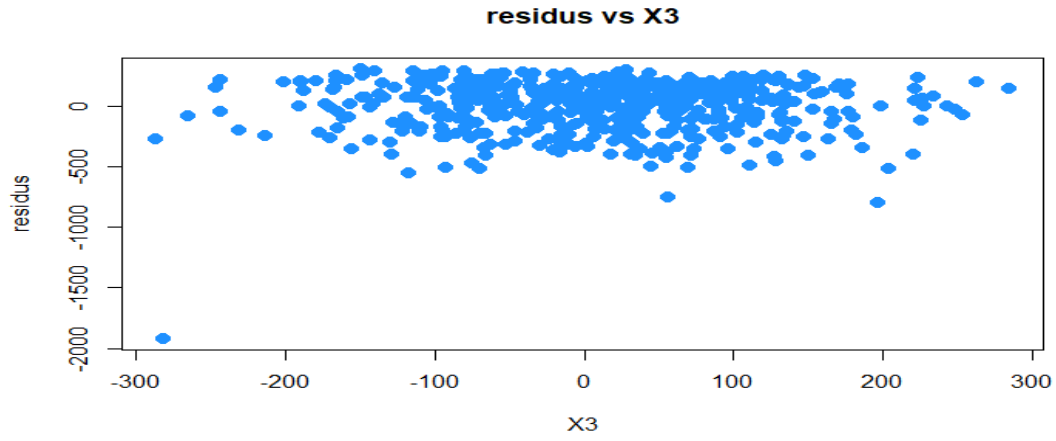


Figure7 : Nuage de point de ε et X_1

- ε et X_4

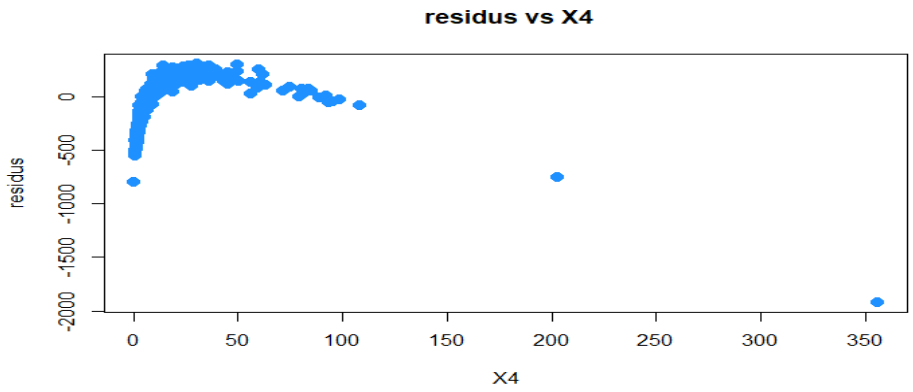


Figure8 : Nuage de point de ε et X_1

Des liens linéaires plus nets apparaissent-ils ?

Oui, pour toutes les variables sauf X_4.

Voit-on d'autres liens?

Pour X_4 le lien n'est pas linéaire.

4. Construisons la matrice *Z* contenant les variables X_1, X_2, X_3 et $ln(X_4)$

R dispose d'une fonction « matrix » qui va permettre de construire Z.

```
105  #4
106  #### la matrice Z contenant les variables X1,X2,X3 et ln(X4)
107  Z=matrix(c(X1,X2,X3,log(X4)),ncol = 4)
108  Z
109
```

Faisons la régression de *Y* sur les variables de *Z* et comparer les résultats à ceux obtenus lors de la première régression.

##Code

```
110  #### la régression de Y sur les variables de Z
111  regLm3=lm(Y~Z)
112  summary(regLm3)
```

##Sortie

```
> #### la régression de Y sur les variables de Z
> regLm3=lm(Y~Z)
> summary(regLm3)

Call:
lm(formula = Y ~ Z)

Residuals:
     Min       1Q   Median       3Q      Max
-119.106  -29.985   -0.311   28.064  132.717

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  50.92217    4.53230  11.235   <2e-16 ***
Z1            2.01242    0.01938 103.840   <2e-16 ***
Z2            3.03279    0.01895 160.004   <2e-16 ***
Z3           -0.01731    0.01894  -0.914    0.361
Z4          298.97542    1.85056 161.560   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 42 on 495 degrees of freedom
Multiple R-squared:  0.9923,    Adjusted R-squared:  0.9922
F-statistic: 1.587e+04 on 4 and 495 DF,  p-value: < 2.2e-16
```

Le R-squared est égal à 0,9923.

On constate seul la variable Z3 n'est pas significative.

Refaire la régression après avoir retiré l'une après l'autre les variables dont les coefficients ne sont pas significatifs (il faut en pratique éviter de retirer plusieurs variables en même temps : on retire d'abord la moins significative avant de refaire une régression).

##Code

```
##On retire X3
###### Comparaison entre premier modele et dernier modele
# On remarque que R2=0.9923 variables sont significatives sauf Z3 qui pas significatif
# alors que pour la premier modele R2=R2=0.7972 et que tout les variables sont significatives
Z1=matrix(c(X1,X2,log(X4)),ncol = 3)
regLm4=lm(Y~Z1)
summary(regLm4)
##### R2 ajuster n'a pas changer
```

##Sortie

```
> summary(regLm4)

Call:
lm(formula = Y ~ Z1)

Residuals:
     Min       1Q   Median       3Q      Max
-120.840  -30.093   -1.178   27.594  133.917

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  51.07129    4.52861   11.28   <2e-16 ***
Z11           2.01355    0.01934  104.12   <2e-16 ***
Z12           3.03141    0.01889  160.47   <2e-16 ***
Z13         298.82985    1.84338  162.11   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 42 on 496 degrees of freedom
Multiple R-squared:  0.9923,    Adjusted R-squared:  0.9922
F-statistic: 2.117e+04 on 3 and 496 DF,  p-value: < 2.2e-16
```

Le retrait de $\boldsymbol{X_3}$ ne change rien à la valeur de $\boldsymbol{R^2}$.

5. Ajouter la variable $ln(X4)$ à la matrice X et déterminer par une méthode automatique, en utilisant le critère d'Akaike, quel est le meilleur modèle.

##Code

```
#5
X=matrix(c(X1,X2,X3,X4,log(X4)),ncol = 5)
X
regLm5=lm(Y~X)
summary(regLm5)
```

##Sortie

```
> summary(regLm5)

Call:
lm(formula = Y ~ X)

Residuals:
     Min       1Q   Median       3Q      Max
-119.027  -29.892   -0.277   28.074  132.488

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  51.135616   5.179305   9.873   <2e-16 ***
X1            2.012485   0.019413 103.666   <2e-16 ***
X2            3.032836   0.018980 159.790   <2e-16 ***
X3           -0.017177   0.019016  -0.903    0.367
X4            0.009576   0.112088   0.085    0.932
X5          298.810242   2.677571 111.598   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 42.04 on 494 degrees of freedom
Multiple R-squared:  0.9923,    Adjusted R-squared:  0.9922
F-statistic: 1.267e+04 on 5 and 494 DF,  p-value: < 2.2e-16
```

Le meilleur modèle c'est celui qui a un plus faible AIC. D'après le résultat suivant on constate que le modèle avec suppression de X_3 a un plus faible AIC. On peut conclure que ce dernier est le meilleur modèle.

##Sortie

```
> AIC(regLm4,regLm5)
       df      AIC
regLm4  5 5162.477
regLm5  7 5165.627
>
```

6. On décide de conserver le modèle $\boldsymbol{Y = \beta_0 + \beta_1 \times X_1 + \beta_2 \times X_2 + \beta_3 \times ln(X_4) + \mathcal{E}}$. Testons la normalité des résidus.

En statistiques, les **tests de normalité** permettent de vérifier si des données réelles suivent une loi normale ou non. Les **tests de normalité** sont des cas particuliers des **tests** d'adéquation (ou **tests** d'ajustement, **tests** permettant de comparer des distributions), appliqués à une loi normale.

##Code

```
#6
###### on conserve le modele  de question 5
X=matrix(c(X1,X2,log(X4)),ncol = 3)
X
regLm=lm(Y~X)
summary(regLm)
e1=resid(regLm)
```

##Sortie

```
> e1=resid(regLm)
> shapiro.test(e1)

        Shapiro-Wilk normality test

data:  e1
W = 0.99753, p-value = 0.6728

>
```

Le modèle ci-dessus renvoie une p-value non significative. L'échantillon suit donc une loi normale.

7. Tester l'hypothèse d'homoscédasticité.

Pour répondre à cette question, on importe la Library « lmtest » et on choisit la fonction « bptest ».

##Code

```
#.7
### l'hypothèse d'homoscédasticité
library(lmtest)
bptest(regLm)
```

##Sortie

```
> library(lmtest)
Le chargement a nécessité le package : zoo

Attachement du package : 'zoo'

The following objects are masked from 'package:base':

    as.Date, as.Date.numeric

Warning messages:
1: le package 'lmtest' a été compilé avec la version R 3.5.3
2: le package 'zoo' a été compilé avec la version R 3.5.3
> bptest(regLm)

        studentized Breusch-Pagan test

data:  regLm
BP = 3.2006, df = 3, p-value = 0.3617
```

On remarque la p-value est supérieure à α, on peut conclure que les variances sont égales.

8. Donner un intervalle de confiance à 95%, puis à 99%, pour chacun des paramètres.

Pour avoir l'intervalle de confiance, on se sert de la fonction « confint », avec les paramètres : nomModèle et niveau de confiance(level).

##Code

```
#.8
###intervalle de confiance à 95%,
confint(regLm, level = 0.95)
###intervalle de confiance à 95%,
confint(regLm, level = 0.99)
####intervalle de confiance à 95% pour beta1
```

##Sortie

```
> #.8
> ###intervalle de confiance à 95%,
> confint(regLm, level = 0.95)
                  2.5 %     97.5 %
(Intercept)  42.173670  59.968901
X1            1.975552   2.051540
X2            2.994295   3.068526
X3          295.208047 302.451657
> ###intervalle de confiance à 95%,
> confint(regLm, level = 0.99)
                  0.5 %     99.5 %
(Intercept)  39.361318  62.781253
X1            1.963543   2.063549
X2            2.982563   3.080258
X3          294.063269 303.596434
```

On obtient les valeurs des β_i à 95% et 99% i= 1,…,3.

Vérifier que l'on obtient bien le même intervalle de confiance à 95% pour β_1 en utilisant les formules du cours (il faut donc l'obtenir par calculs matriciels).

##Code

```
#### calcul de XX_bar
n = nrow(dt)
p = length(coef(regLm))
X = cbind(rep(1, n),X)
y = Y
X_X=solve(t(X) %*% X)
X_X
###Calcul de beta_chapeau
(beta_hat = solve(t(X) %*% X) %*% t(X) %*% y)
coef(regLm)
### Sigma
summary(regLm)$sigma
#### degre de liberté =496
####beta1=2.013
#### sigma_chapeau=41.99517
y_hat = X %*% solve(t(X) %*% X) %*% t(X) %*% y
e = y - y_hat
sqrt(t(e) %*% e / (n - p))
#### statistique  student = 1.960
##### avec formule du cours
c(2.013-1.960*41.99517*sqrt(2.120365e-07), 2.013+1.960*41.99517*sqrt(2.120365e-07))
```

##Sortie

```
> #### calcul de XX_bar
> n = nrow(dt)
> #### calcul de XX_bar
> n = nrow(dt)
> p = length(coef(regLm))
> X = cbind(rep(1, n),X)
> y = Y
> X_X=solve(t(X) %*% X)
> X_X
              [,1]          [,2]          [,3]          [,4]
[1,]  1.162868e-02 -1.233432e-06  8.340204e-07 -4.303072e-03
[2,] -1.233432e-06  2.120365e-07  1.066811e-08 -2.984181e-07
[3,]  8.340204e-07  1.066811e-08  2.023504e-07 -6.732006e-07
[4,] -4.303072e-03 -2.984181e-07 -6.732006e-07  1.926785e-03
> ###Calcul de beta_chapeau
> (beta_hat = solve(t(X) %*% X) %*% t(X) %*% y)
           [,1]
[1,]  51.071285
[2,]   2.013546
[3,]   3.031411
[4,] 298.829852
> coef(regLm)
(Intercept)          X1          X2          X3
  51.071285    2.013546    3.031411  298.829852
> ### Sigma
> summary(regLm)$sigma
[1] 41.99517
> #### degre de liberté =496
> ####beta1=2.013
> #### sigma_chapeau=41.99517
> y_hat = X %*% solve(t(X) %*% X) %*% t(X) %*% y
> e = y - y_hat
> sqrt(t(e) %*% e / (n - p))
         [,1]
[1,] 41.99517
> #### statistique  student = 1.960
> ##### avec formule du cours
> c(2.013-1.960*41.99517*sqrt(2.120365e-07), 2.013+1.960*41.99517*sqrt(2.120365e-07))
[1] 1.975098 2.050902
```

En conclusion, nous constatons que les intervalles de confiances sont les mêmes à 95% pour β_1.

9. Donnons une prévision et un intervalle de confiance (ou plutôt de pari) à 95% pour $Y\ si\ X_1 = X_2 = X_4 = 200$.

##Code

```
180 dtt=cbind.data.frame(Y=dt$Y,X1=dt$X1,X2=dt$X2,X4=log(dt$X4))
181 dtt
182 regLm0=lm(Y~X1+X2+X4,data = dtt)
183 summary(regLm0)
184 #.9 prévision et un intervalle de confiance
185 X_pred = data.frame(X1=200,X2=200,X4=log(200))
186 X_pred
187
188 predict(regLm0, newdata = X_pred,interval = "prediction", level = 0.95)
189
```

##Sortie

```
> #.9 prévision et un intervalle de confiance
> X_pred = data.frame(X1=200,X2=200,X4=log(200))
> X_pred
   X1  X2       X4
1 200 200 5.298317
> predict(regLm0, newdata = X_pred,interval = "prediction", level = 0.95)
       fit      lwr      upr
1 2643.358 2559.403 2727.313
```

Nous constatons que la valeur prédite rentre dans le nouvel intervalle de confiance. C'est-à-dire :

$$2643{,}358 \in [2559{,}403\ ;\ 2727{,}313]$$

Retrouver par le calcul, avec les formules du cours, l'intervalle obtenu.

##Code

```
190  ### avec formule du cours
191  Xn=as.matrix(c(1,200,200,200))
192  Xn
193  Y_n1=solve(t(Xn)%*%beta_hat)
194  Y_n1
195  c(Y_n1-1.960*41.99517*sqrt(1+t(Xn)%*%X_X%*%Xn),Y_n1+1.960*41.99517*sqrt(1+t(Xn)%*%X_X%*%Xn))
```

Retrouver le résultat obtenu lorsque l'on choisit l'option interval = "confidence".

##Code

```
197  predict(regLm0, newdata = X_pred,interval = "confidence", level = 0.95)
198
```

##Sortie

```
> predict(regLm0, newdata = X_pred,interval = "confidence", level = 0.95)
       fit      lwr      upr
1 2643.358 2627.849 2658.867
```

Exercice3 :

Le fichier "dataR.txt" disponible contient les valeurs de 6 variables : Y, X_1, X_2, X_3, X_4 *et* X_5, mesurées sur un échantillon de 500 individus. Nous voulons expliquer Y en fonction des autres variables à l'aide d'un modèle de régression linéaire.

1. Calculer la matrice de corrélations des variables explicatives et créer une matrice 5 × 5 dont le terme d'indice (i, j) est la p-valeur associée au test de nullité du coefficient de corrélation (de Pearson) entre X_i *et* X_j

##Code

```
###on charge la base de donnée
dataR=read.table(file.choose() ,sep=" ",header =TRUE)
dataR
attach(dataR)
dataR[-which(dataR$X1 < 2500),]
dataR


###1

cor(as.matrix(dataR[,2:6]))
W=as.matrix(dataR[,2:6])
W
```

##Sortie

```
> cor(W)
          X1        X2        X3         X4         X5
X1 1.0000000 0.7228088 0.3228775 0.30651587 0.49513336
X2 0.7228088 1.0000000 0.3772079 0.37486536 0.47387683
X3 0.3228775 0.3772079 1.0000000 0.34311474 0.13093242
X4 0.3065159 0.3748654 0.3431147 1.00000000 0.06642686
X5 0.4951334 0.4738768 0.1309324 0.06642686 1.00000000
```

```
> #####matrice de p-value
> rcorr(as.matrix(dataR[,2:6]),type=c("pearson"))
     X1   X2   X3   X4   X5
X1 1.00 0.72 0.32 0.31 0.50
X2 0.72 1.00 0.38 0.37 0.47
X3 0.32 0.38 1.00 0.34 0.13
X4 0.31 0.37 0.34 1.00 0.07
X5 0.50 0.47 0.13 0.07 1.00

n= 500


P
   X1     X2     X3     X4     X5
X1        0.0000 0.0000 0.0000 0.0000
X2 0.0000        0.0000 0.0000 0.0000
X3 0.0000 0.0000        0.0000 0.0034
X4 0.0000 0.0000 0.0000        0.1380
X5 0.0000 0.0000 0.0034 0.1380
>
```

Doit-on craindre un problème de multi colinéarité ?

##Code

```
install.packages("Hmisc")
library(Hmisc)
#####matrice de p-value
rcorr(as.matrix(dataR[,2:6]),type=c("pearson"))
cor.test(dataR$X1,dataR$X2,method="pearson")
# +++++++++++++++++++++++++++
# flattenCorrMatrix
# +++++++++++++++++++++++++++
# cormat : matrice des coefficients de corrélation
# pmat : matrice des p-valeurs de corrélation
flattenCorrMatrix <- function(cormat, pmat) {
  ut <- upper.tri(cormat)
  data.frame(
    row = rownames(cormat)[row(cormat)[ut]],
    column = rownames(cormat)[col(cormat)[ut]],
    cor  =(cormat)[ut],
    p = pmat[ut]
  )
}
res<-rcorr(as.matrix(dataR[,2:6]))
flattenCorrMatrix(res$r, res$P)
```

##Sortie

```
   row column         cor            p
1   X1     X2  0.70447049 0.000000e+00
2   X1     X3  0.27616546 3.472618e-10
3   X2     X3  0.33561760 1.332268e-14
4   X1     X4  0.25704319 5.688171e-09
5   X2     X4  0.33172632 2.797762e-14
6   X3     X4  0.29584487 1.544698e-11
7   X1     X5  0.44313124 0.000000e+00
8   X2     X5  0.42167086 0.000000e+00
9   X3     X5  0.02813290 5.306662e-01
10  X4     X5 -0.04923512 2.723194e-01
>
```

Nous remarquons que les variables X_1 et X_2 sont colinéaires. Il n'y a pas de multicolinéarité.

2. En faisant une sélection de variables avec le critère BIC, quelles variables faudrait-il conserver ?

##Code

```
238  ####2 sélection de variables avec le critère BIC
239  Y_mod=lm(Y~.,data=dataR)
240  coef(Y_mod)
241  Y_mod_bic=step(Y_mod,direction = "backward",k = log(n))
242  coef(Y_mod_bic)
```

##Sortie

```
> ####2 sélection de variables avec le critère BIC
> Y_mod=lm(Y~.,data=dataR)
> coef(Y_mod)
(Intercept)          X1          X2          X3          X4          X5
713.0882664   0.8329416   2.1405553   1.8709773  -0.1322768  -0.2311776
> Y_mod_bic=step(Y_mod,direction = "backward",k = log(n))
Start:  AIC=5301.38
Y ~ X1 + X2 + X3 + X4 + X5

       Df Sum of Sq      RSS    AIC
- X4    1    201625 18878488 5300.5
<none>              18676863 5301.4
- X5    1    484986 19161850 5308.0
- X1    1   4812835 23489698 5409.8
- X2    1  31401478 50078341 5788.3
- X3    1  43484614 62161477 5896.4

Step:  AIC=5300.54
Y ~ X1 + X2 + X3 + X5

       Df Sum of Sq      RSS    AIC
<none>              18878488 5300.5
- X5    1    409021 19287510 5305.0
- X1    1   4694260 23572748 5405.4
- X2    1  31756558 50635046 5787.6
- X3    1  44373239 63251727 5898.9
> coef(Y_mod_bic)
(Intercept)          X1          X2          X3          X5
609.2064087   0.8203814   2.1044836   1.8426348  -0.2101862
>
```

On peut conserver toutes les variables sauf X_4.

3. Représenter *Y* en fonction des valeurs prédites par le modèle.

❖ ***Y et*** X_1

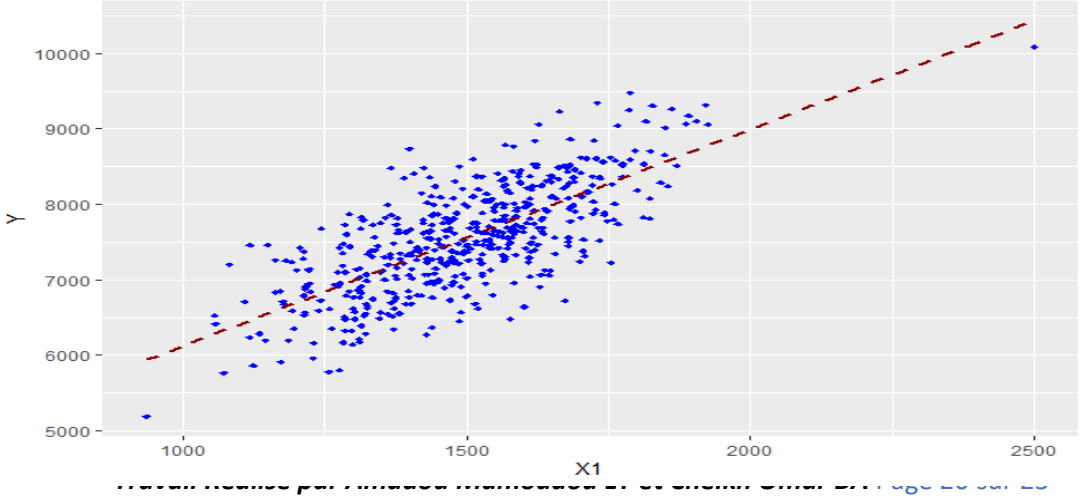


Figure9 : Nuage de point de Y et X_1

❖ Y et X_2

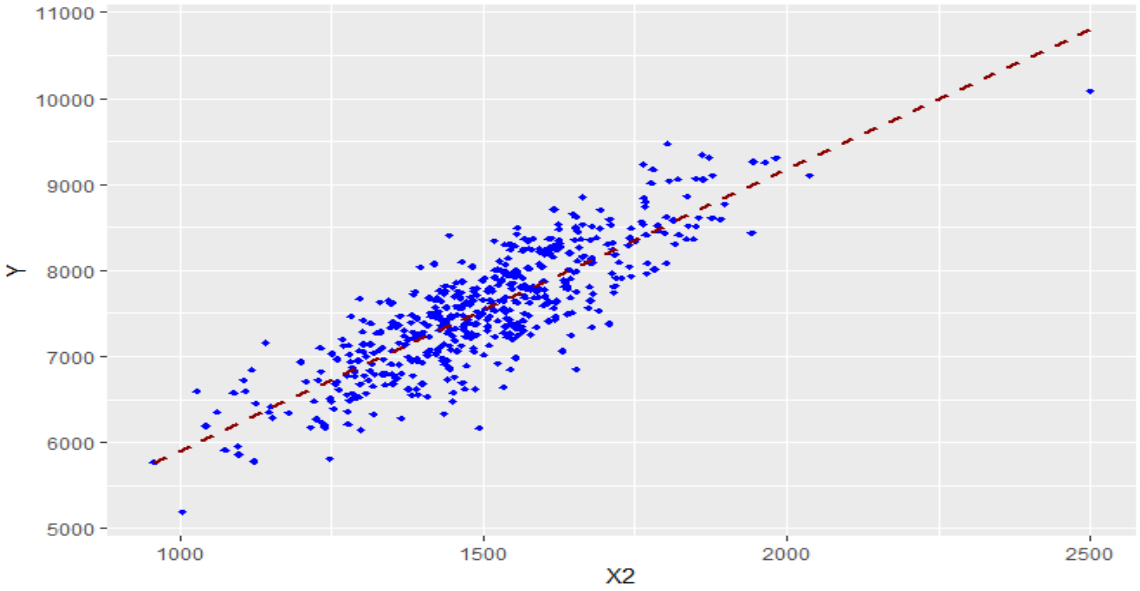


Figure10 : Nuage de point de Y et X_2

❖ Y et X_3

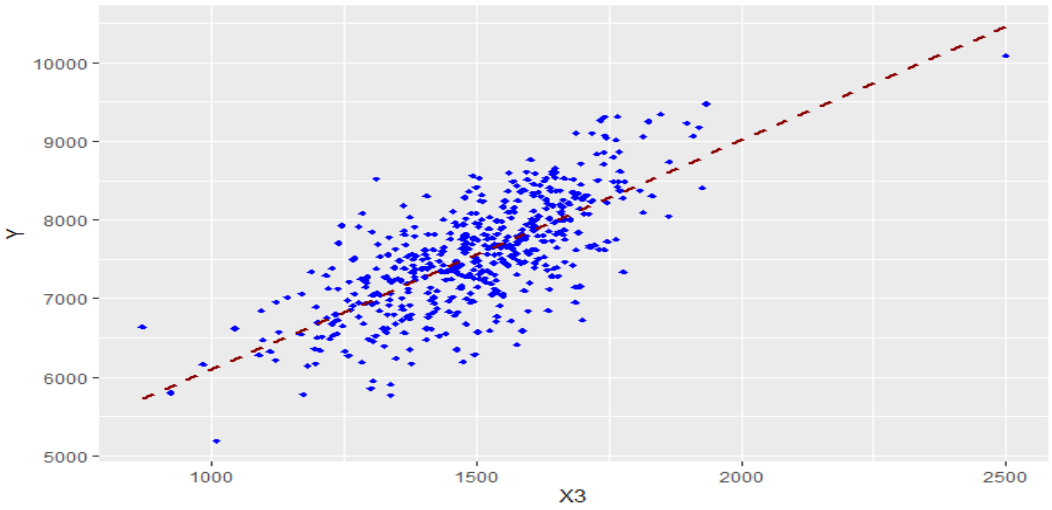


Figure11 : Nuage de point de Y et X_3

❖ Y et X_5

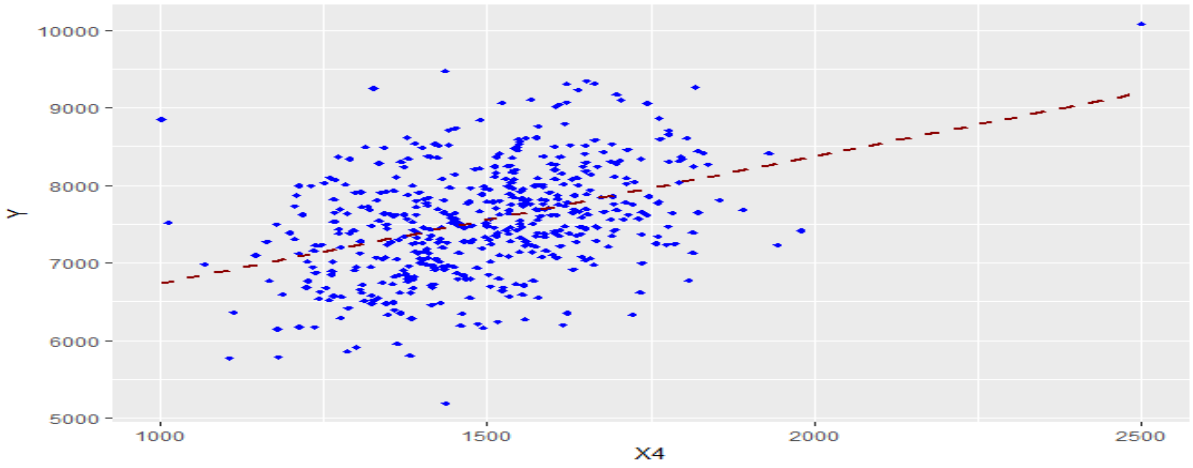


Figure12 : Nuage de point de Y et X_5

Représenter les résidus studentisés.

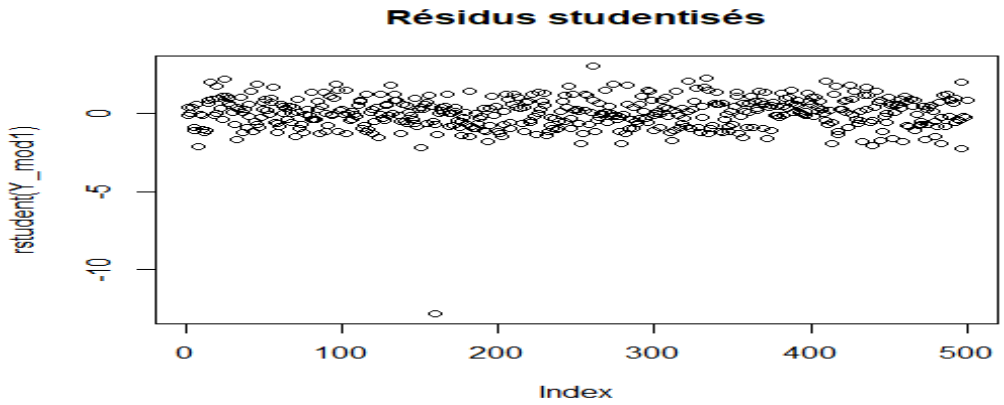


Figure13 : Nuage de point de Résidus studentisés

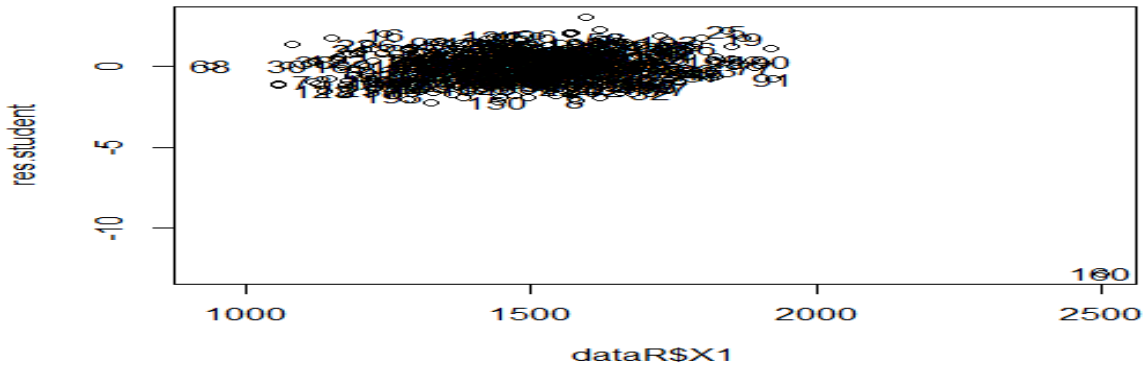


Figure1' : Nuage de point de Résidus studentisés par rapport à dataR$X1

Que remarque-t-on ?

Nous remarquons que c'est la valeur à la ligne 160 qui est le point atypique.

4. Quels sont les éventuelles valeurs anormales ?

Les valeurs anormales sont les points atypiques.

##Code

```
379  ###4 eventuelle anomalie
380  boxplot.stats(X1)$out
381  ###5
382
```

##Sortie

```
> ###4 eventuelle anomalie
> boxplot.stats(X1)$out
[1]  936.3827 2500.0000
```

5. Retirer l'observation ayant une influence trop grande, et rechercher le meilleur modèle.

##Code

```
395  ###5
396
397  dataR = dataR[-160,]  # On supprime la 160eme ligne
398
399
400  Y_mod1=lm(Y~X1+X2+X3+X5,data = dataR)
401  summary(Y_mod1)
402  r=resid(Y_mod1)
```

##Sortie

```
> r=resid(Y_mod1)
> dataR = dataR[-160,]  # On supprime la 160eme ligne
> Y_mod1=lm(Y~X1+X2+X3+X5,data = dataR)
> summary(Y_mod1)

Call:
lm(formula = Y ~ X1 + X2 + X3 + X5, data = dataR)

Residuals:
    Min      1Q  Median      3Q     Max
-431.21 -126.24   -8.57  115.89  605.78

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 166.14101   89.88371   1.848   0.0651 .
X1            0.82848    0.06428  12.890   <2e-16 ***
X2            2.08221    0.06342  32.832   <2e-16 ***
X3            1.97462    0.04803  41.113   <2e-16 ***
X5            0.05896    0.05967   0.988   0.3235
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 169.5 on 492 degrees of freedom
Multiple R-squared:  0.9411,    Adjusted R-squared:  0.9407
F-statistic:  1966 on 4 and 492 DF,  p-value: < 2.2e-16
```

Comment le R^2 a-t-il évolué ?

Le R^2 a augmenté. Car lorsqu'on avait les points atypiques le R^2 était de 92% et après suppression il passe à 94%.