TP- Aide

Jm Bécu

Aide TP

Test environnement travail

```
#test package irene
villes <- read.csv('./DonneesGPSvilles.csv',header=TRUE,dec='.',sep=';',quote="\"")
coord <- cbind(villes$longitude,villes$latitude)
dist <- distanceGPS(coord)
voisins <- TSPnearest(dist)
print(voisins)
```

```
## $longueur
## [1] 4303.568
##
## $chemin
##  [1]  1  8 11 18 15 19  6 20  3 10 17 16  7 13 21  4  9 14 12  2 22  5
```

En théorie vous devez avoir obtenu cette sortie. Si non, vous ne pourrez pas faire le TP.

Si ça ne marche pas essayez de les réinstaller puis re-testez:

install.packages('Rcpp') install.packages(c('maps','sp','microbenchmark','TSP')) install.packages('./TSPpackage_1.0.zip', repos = NULL, type = "win.binary")

Questions

**0/ Regression Lineaire

La régression linéaire résout un modèle tel que

$y = ax + b + \epsilon$ où $\epsilon \sim \mathcal{N}(0, \sigma^2)$ et x n'a pas de distribution associée.

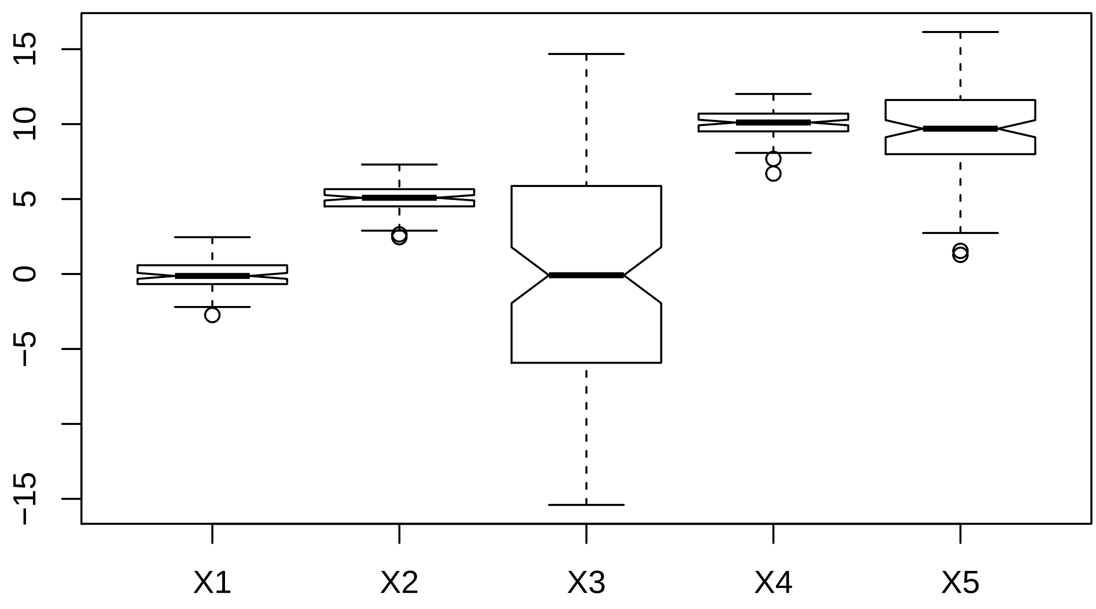
Donc faire une régression des moindres carrés est faisable si on vérifie 3 hypothèses :

- $y = ax \dots$: Il existe une **relation linéaire** entre y et x. Hypothèse testé avec le test de student sur coefficient a. $H_0 : a = 0$ doit être rejetée.
- $y = \dots + b$: Il existe un **biais constant** b non nul. Hypothèse testée avec test de student sur b. Cette hypothèse ne remet pas en cause le modèle mais simplement sur l'existence de b selon $H_0 : b = 0$.
- $y = \dots + \epsilon$: Il existe un **bruit gaussien** $\epsilon \sim \mathcal{N}(0, \sigma^2)$ qui affecte y et qui est indépendant de x. Hypothèse testée par shapiro-wilk sur les résidus. En effet si les deux première hypothèse sont vérifiée et que le modèle est bon alors $R = y - \hat{y} = y - (\hat{a}x + \hat{b}_{\text{si } b \neq 0})$ et R est donc un estimateur $\hat{\epsilon}$. $H_0 : \epsilon \sim \mathcal{N}$ ne doit pas être rejetée.

Si les hypothèse sur ax et ϵ vont dans le bon sens alors le modèle supposé est validé et son estimation pourra être utilisée pour prédire y selon de nouveaux x.

1/ SI ensemble de 5 vecteur comment faire matrice avec 5 colonne pour boxplot

```
X1 <- rnorm(100)
X2 <- rnorm(100,5)
X3 <- rnorm(100,0,8)
X4 <- rnorm(100,10)
X5 <- rnorm(100,10,3)


mat <- cbind(X1,X2,X3,X4,X5)
par(mfrow=c(1,1))
boxplot(mat,notch=TRUE)
```



2/ Somme de loi normale (pour le t-test branch nearest voir TD 2, exo 3, q2)

Si $X_1 \sim \mathcal{N}(m_1, \sigma_1^2)$ et $X_2 \sim \mathcal{N}(m_2, \sigma_2^2)$ alors

$$Yplus = X_1 + X_2 \sim N(m_1 + m_2, sqrt(\sigma_1^2 + \sigma_2^2)$$

et

$$Yminus = X_1 + X_2 \sim N(m_1 - m_2, sqrt(\sigma_1^2 + \sigma_2^2)$$

```
X1 <- rnorm(1000,mean=1,sd= 1)
X2 <- rnorm(1000,mean=2,sd= 0.5)

t.test(X1-X2)
```

```
##
##   One Sample t-test
##
## data:  X1 - X2
## t = -27.679, df = 999, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -1.0644557 -0.9235136
## sample estimates:
##  mean of x
## -0.9939847
```

```
t.test(X1,X2,paired=TRUE)
```

```
##
##   Paired t-test
##
## data:  X1 and X2
## t = -27.679, df = 999, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.0644557 -0.9235136
## sample estimates:
## mean of the differences
##              -0.9939847
```

Le test pairé est le même que celui fait sur la différence.

3/ microBenchmark

Remarque : Le package **multcomp** est appelé de manière transparente dans la fonction **microbenchmark**. Les résultats de celui-ci apparaissent dans la colonne **cld** (deernière colonne) de la sortie. Donc l'interprétation est expliquée ci-dessous.

Tout les membres d'un meme groupe n'ont pas de différence significative pour leurs moyenne et les groupes $\{a, b, c, d, \dots\}$ sont rangés de manière croisante.

Exemple - si variables X et variable Y sont dans le groupe a alors $m_X \simeq m_Y$ où plutot qu'il n'a pas pu être mis en évidence une différence significative entre les deux. - si variables X et variable Y sont dans le groupe a et **b** alors $m_X \neq m_Y$ significativement. Et comme $\{a, b, c, d, \dots\}$ sont rangés de manière croisante alors $m_a < m_b$ donc $m_X < m_Y$

4/ Test fisher

La loi de fisher $\mathcal{F}(d_1, d_2)$ est définie sur $[0; +\infty]$ avec deux paramètre d_1 et d_2. Comme pour le test du χ^2 on pout l'intepréter comme une "distance" entre deux modèles ce qui explique les hypothèse (au sens mathématique du terme) :

$(H_0) : F = 0$ contre $(H_1) : F > 0$ où F est la statistique de Fisher à tester.

```
  summary(mod2)
```

```
##
## Call:
## lm(formula = yobs2 ~ X)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.08825 -0.05976 -0.01614  0.05420  0.16833
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.86001    0.01437   59.84   <2e-16 ***
## X           15.95615    0.02409  662.48   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.071 on 98 degrees of freedom
## Multiple R-squared:  0.9998, Adjusted R-squared:  0.9998
## F-statistic: 4.389e+05 on 1 and 98 DF,  p-value: < 2.2e-16
```

Dans le cadre de la sortie de la fonction **lm(**$Y \sim X$**)** cette approche teste la pertinence du modèle de régression avec les variables contre le modèle sans ces variables.

La statistique F représente donc ici une distance entre la qualité de prédiction du modèle linéaire basé sur les variables explicatives dans X où ($\hat{y} = y - x\beta$) contre le modèle prédit sans rien du tout ($\hat{y} = y$).

Si la "distance" entre les deux modèles est nulle (ou presque) alors le modèle linéaire n'apporte aucune information utile pour modèliser y (car sans les avriables explicatives on obtient la même chose) et plus cette distance est significativementgrande plus le modèle apporte.

4/ Pairwise-t.test

Vecteur **results** et **methods** de taille $n = 250$ ou **results**$=3.14, 5, 69, \ldots$ et **methods**$='branch',' branch',' NN', \ldots$

Si pval proche de 1 : Aucune différence entre méthode donc boxplot "proche" Si pval proche de 0 : significativement différent entre méthode donc boxplot "disjoint"

5/ Selection de variable

La colonne **Pr(**$> |t|$**)** obtenue par **summary(lm(**y X**))** correspond à la p-value pour un test de Student où

$$(H_0) : \hat{\beta}_i = 0 \text{ contre } (H_0) : \hat{\beta}_i \neq 0$$

Si on suppose

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

alors quand on estime $\hat{\beta}$ par **lm**

$$\hat{\beta} = (X^T X)^{-1}(X^T Y)$$

ce qui revient a estimer les coefficients ou la corrélation entre les variables explicatives et Y est pénalisée par la corrélation des variables entres elles.

Donc si vous rajoutez des variables explicatives ou si vous en retirez :

-$X^T Y$: Ne varie pas donc la relation des variablesdans X une a une avec Y ne change pas

-$(X^t X)^{-1}$ est affectée de façon plus ou moins importante.

- $\hat{\beta}$ (Estimate) est modifié a cause de $(X^t X)^{-1}$
- p-value associées aux variables sont pour le test $(H_0) : \hat{\beta}_i = 0$ pour la i-ème variable. Donc elle seront affectées elles aussi.

AIC : Test importance des vaiables. Si Résidus ne changent pas entre deux 'ensemble' de variable alors les variables qui diffèrent entre les deux modèles ne sont pas 'utiles' significativement.

Regression

Modèle linéaire univarié

$$Y = aX + b + \epsilon$$

où $\epsilon \sim \mathcal{N}(0, \sigma^2)$‘

```
varNoise <- 2
```

- Exemple $Y = 15X + 1 + \epsilon$ où $\epsilon \sim \mathcal{N}(0, 2)$

```
n <- 100
 X <- runif(n)
a <- 15
b <- 1
noi <- rnorm(n,0,varNoise)
yobs <- a*X +b + noi
```

Ici X suit une loi uniforme. En effet aucun a priori sur X n'est nécessaire mais Y et X doivent être linéairement corrélés.

```
mod <- lm(yobs~X) # estimation modèle linéaire par regression des moindres carrés
mod
```

```
##
## Call:
## lm(formula = yobs ~ X)
##
## Coefficients:
## (Intercept)            X
##      0.6168      16.1063
```

```
estB <- mod$coefficients[1]  #  b soit non nul significativement (h0 pas d'intercept)
estA <- mod$coefficients[2] #  a soit non nul significativement
#(H0 : modéle linaire -> qualité modele)
```

lm$(y \sim X)$ applique régression sur Y expliqué par X. L'intercept correspond à **b**. Le coefficient à **a**. (Si on est en multivarié il y a plusieurs coefficients)

```
# residu : bruit du modele  + partie non expliquée de yobs
# pour que modele soit bon residu = bruit (ou presque)
yhat <- estA * X + estB
res <- yobs - yhat # doivent suivre un loi gaussienne
```

Les résidus $r = y - (\hat{a}X + \hat{b}) = y - \hat{y}$.

TEST DU MODELE

test du modèle : - tester significativité de a (pertinence du modele)

$$(H_0) : a = 0 \text{ contre } (H_1) : a \neq 0$$

- tester significativité de b (besoin d'un intercept) : Optionnel car ne renseigune pas sur la pertinence du modèle mais simplement pour savoir si intercept utile ou non

$$(H_0) : b = 0 \text{ contre } (H_1) : b \neq 0$$

- tester residus gaussien (modele a bien fitté ou non). En théorie ne reste que résidus gaussien ou presque.

$$(H_0) : \text{Résidus suivent loi normale contre } (H_1) : \text{Résidus ne suivent pas loi normale}$$

- Test Linéarité

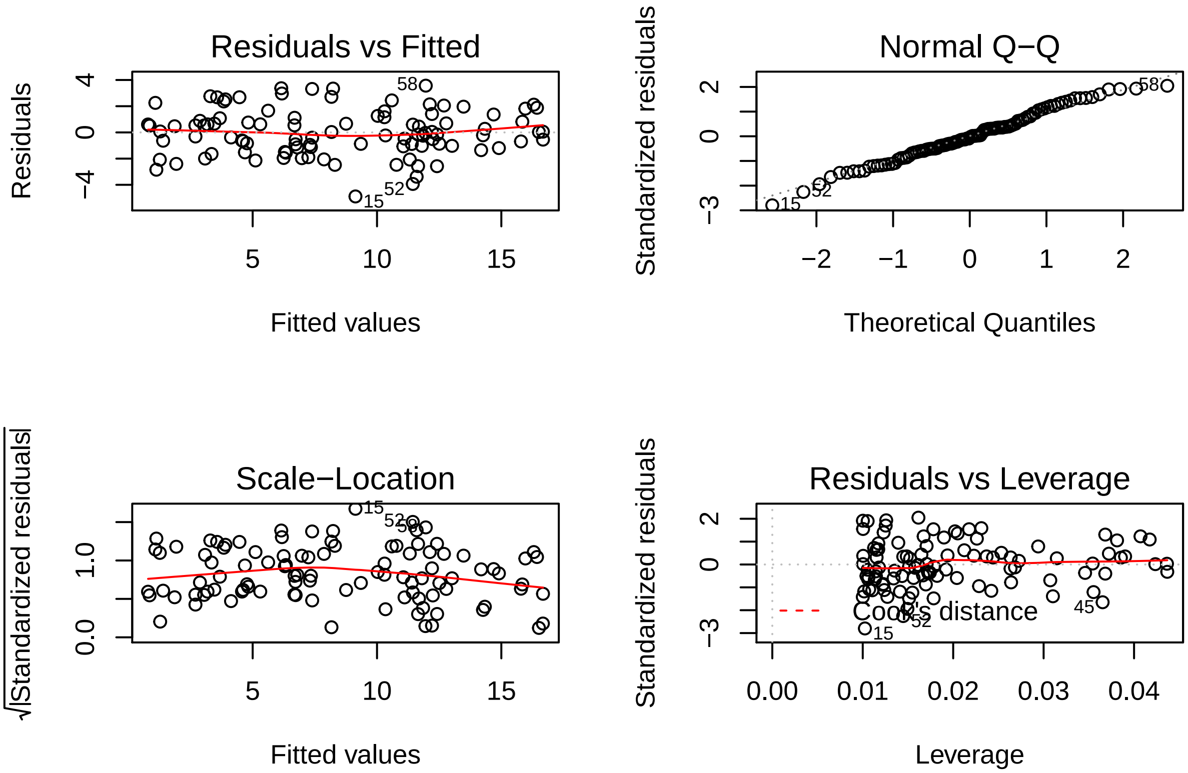
```r
summary(mod)
```

```
##
## Call:
## lm(formula = yobs ~ X)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.887 -1.079 -0.098  1.126  3.566
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.6168     0.3527   1.749   0.0834 .
## X            16.1063     0.6299  25.568   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.756 on 98 degrees of freedom
## Multiple R-squared:  0.8696, Adjusted R-squared:  0.8683
## F-statistic: 653.7 on 1 and 98 DF,  p-value: < 2.2e-16
```

- $Pr(> |t|)$:p-value pour test sur **b** pour **(Intercept)**
- $Pr(> |t|)$:p-value pour test sur **coefficients** pour **X** et cie. . . . Pour chaque coefficient du modèle (ici un seul qui est a) on a la p-value
- $p - value$: p-value pour modèle complet (ensemble des coefficients). Si modèle univarié (comme dans exemple) $p - value = Pr(> |t|)$ pour **X**
- Test Résidus

** Graphiquement

```r
par(mfrow=c(2,2))
plot(mod)
```



- Residuals vs Fitted : Si horizontal et homogene alors linearité et pas d'effet d'echelle
- Normal Q-Q : Compare distribution des residus par rapport a distribution normale. Un point correspond a un rapport des valeurs des memes quantile obtenus pour les deux distribution. Par exemple le point central fait le ratio entre les quantiles q_{res} et q_{norm} tel que $P(X_{res} > q_{res}) = P(X_{norm} > q_{norm}) = q_i$ où $q_i = 50\%$. Si les distribution sont identiques ou presque alors l'ensemble des points sont sur la diagonale. Sinon on observera la plupart du temps des deviation aux extremité ce qui sous-entend que les queues de distribution sont différentes.
- Scale location : Idem qur Residuals vs Fitted mais avec résidus normalisés.
- Residuals vs Leverage : Montre l'influence des echantillons (plus un point est a droite et plus il en a). Si un point est un outliers il apparaitra trés éloigné des autres et en dehors des bornes par rapport à la distance de Cook.

** test sur résidus (shapiro)

```r
#permet de voir graphiquement si ok
shapiro.test(residuals(mod)) # test bruit gaussien : H0 suit loi normale
```

```
##
##   Shapiro-Wilk normality test
##
## data:  residuals(mod)
## W = 0.98881, p-value = 0.5701
```

Pour que le modèle soit OK - coefficients significativment non nul

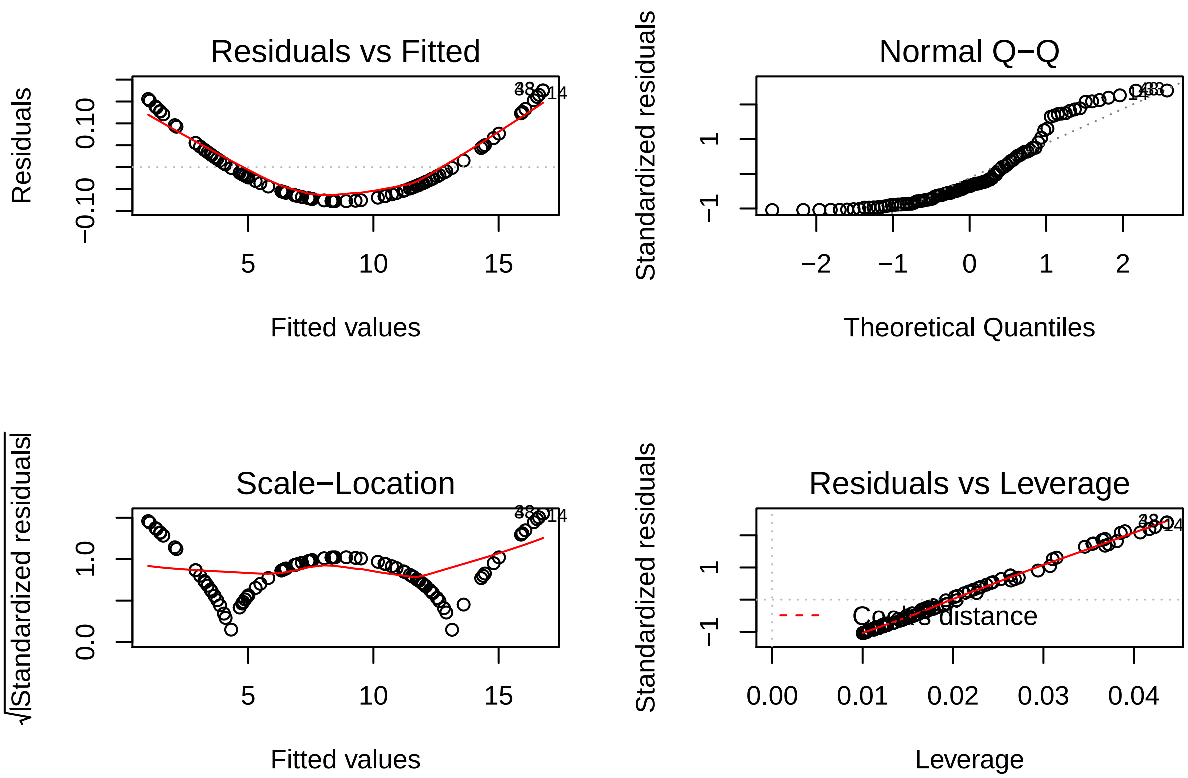
- résidus gaussien

Regression NOK

```
noi2 <- X^2
yobs2 <- a*X +b + noi2
mod2 <- lm(yobs2~X)
estB <- mod2$coefficients[1]  #  b soit non nul significativement (h0 pas d'intercept)
estA <- mod2$coefficients[2] #  a soit non nul significativement
#(HO : modéle linaire -> qualité modele)



# residu : bruit du modele  + partie non expliquée de yobs
# pour que modele soit bon residu = bruit (ou presque)
yhat2 <- estA * X + estB
res2 <- yobs2 - yhat2 # doivent suivre un loi gaussienne


#tester significativité de a (pertinence du modele) ->
#HO  a = 0 ; permet de dire si corrélation lineaire de X avec Y
#tester significativité de b (besoin d'un intercept) ->
#HO b=0; informatif pour interet de l'intercept
#tester residus gaussien (modele a bien fitté ou non) ->
#HO bruit gaussien; necessaire pour savoir si
# le modele predit bien y (ne reste que résidus gaussien ou presque)

summary(mod2)
```

```
##
## Call:
## lm(formula = yobs2 ~ X)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.07762 -0.05857 -0.02625  0.04084  0.17529
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.83372    0.01499   55.61   <2e-16 ***
## X           15.98799    0.02678  597.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07463 on 98 degrees of freedom
## Multiple R-squared:  0.9997,	Adjusted R-squared:  0.9997
## F-statistic: 3.565e+05 on 1 and 98 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2))
plot(mod2)
```




```r
#permet de voir graphiquement si ok


shapiro.test(residuals(mod2)) # test bruit gaussien : H0 suit loi normale
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(mod2)
## W = 0.85574, p-value = 1.944e-08
```

Regression Multivarié

Modèle linéaire univarié

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

où $\epsilon \sim \mathcal{N}(0, \sigma^2)$‘

```
varNoise <- 2
```

- Exemple $Y = 15X_1 + 3X_2 + 9X_3 + +0\epsilon$ où $\epsilon \sim \mathcal{N}(0,2)$

```
n <- 100
 X1 <- runif(n)
 X2 <- rnorm(n,5)
 X3 <- rnorm(n,0,8)
 X4 <- rnorm(n,50,8)
 X = cbind(X1,X2,X3,X4)
a <- 15
b <- 10
c <- 9
d <- 0
varNoise <-0.5
noi <- rnorm(n,0,varNoise)
yobsM <- a*X[,1]+b*X[,2]+c*X[,3]+d*X[,4] + noi
```

```
modM <- lm(yobsM~X) # estimation modèle linéaire par regression des moindres carrés
modM
```

```
##
## Call:
## lm(formula = yobsM ~ X)
##
## Coefficients:
## (Intercept)          XX1          XX2          XX3          XX4
##    0.014367    14.836211    10.053989     8.995821    -0.003611
```

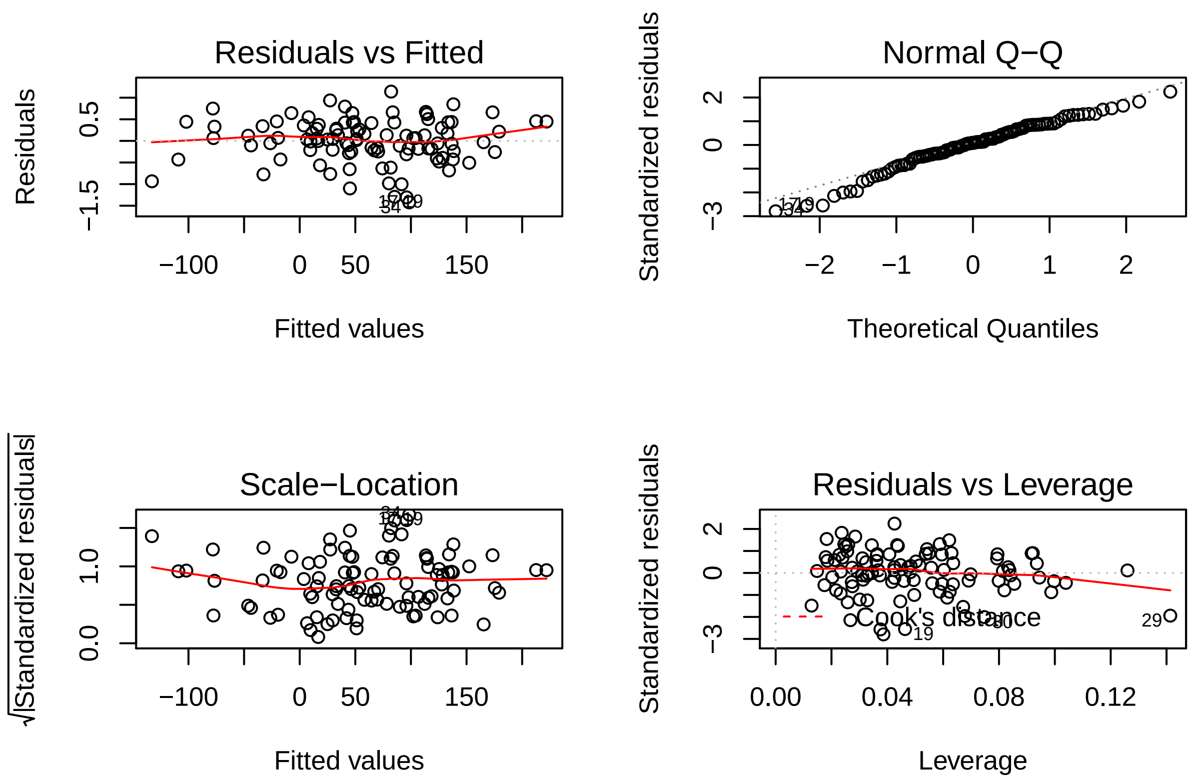
```
summary(modM)
```

```
##
## Call:
## lm(formula = yobsM ~ X)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.42333 -0.24677  0.04479  0.38119  1.14048
##
## Coefficients:
##              Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  0.014367   0.383707    0.037    0.970
## XX1         14.836211   0.185415   80.016   <2e-16 ***
## XX2         10.053989   0.049007  205.153   <2e-16 ***
## XX3          8.995821   0.006960 1292.557   <2e-16 ***
## XX4         -0.003611   0.007484   -0.483    0.631
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5195 on 95 degrees of freedom
## Multiple R-squared:  0.9999,  Adjusted R-squared:  0.9999
## F-statistic: 4.372e+05 on 4 and 95 DF,  p-value: < 2.2e-16
```

```
#(H0 : modéle linaire -> qualité modele)
```

lm$(y \sim X)$ applique régression sur Y expliqué par X. L'intercept correspond à **b**. On est en multivarié il y a plusieurs coefficients. **Ici X4 n'est pas significative pour le modèle.**

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




```
shapiro.test(residuals(modM))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(modM)
## W = 0.97591, p-value = 0.0636
```

#permet de voir graphiquement si ok