

TD Econométrie - Séance 4 - Analyse des cigarettes

R.R.

Inspection des données

1. Chargez le fichier « cigarettes_pour_regression.txt » dans un data frame.

```
#charger les données
#dans ce format, le séparateur est tabulation, la première ligne contient
#le nom des variables, le point décimal est ".", la première colonne
#le nom des observations
#changement de dossier
setwd("../votre dossier de travail ...")
cigarettes <- read.table(file="cigarettes_pour_regression.txt", sep="\t", header=TRUE, dec=".", row.names=1)
```

2. Affichez les observations.

```
#vérification -- affichage des valeurs
print(cigarettes)
```

##	TAR	NICOTINE	WEIGHT	CO
## Alpine	14.1	0.86	0.9853	13.6
## Benson_Hedges	16.0	1.06	1.0938	16.6
## Camellights	8.0	0.67	0.9280	10.2
## Carlton	4.1	0.40	0.9462	5.4
## Chesterfield	15.0	1.04	0.8885	15.0
## GoldenLights	8.8	0.76	1.0267	9.0
## Kent	12.4	0.95	0.9225	12.3
## Kool	16.6	1.12	0.9372	16.3
## L_M	14.9	1.02	0.8858	15.4
## LarkLights	13.7	1.01	0.9643	13.0
## Marlboro	15.1	0.90	0.9316	14.4
## Merit	7.8	0.57	0.9705	10.0
## MultiFilter	11.4	0.78	1.1240	10.2
## NewportLights	9.0	0.74	0.8517	9.5
## Now	1.0	0.13	0.7851	1.5
## OldGold	17.0	1.26	0.9186	18.5
## PallMallLight	12.8	1.08	1.0395	12.6
## Raleigh	15.8	0.96	0.9573	17.5
## SalemUltra	4.5	0.42	0.9106	4.9
## Tareyton	14.5	1.01	1.0070	15.9
## TrueLight	7.3	0.61	0.9806	8.5
## ViceroyRichLight	8.6	0.69	0.9693	10.6
## VirginiaSlims	15.2	1.02	0.9496	13.9
## WinstonLights	12.0	0.82	1.1184	14.9

Affichez le nombre de lignes et de colonnes du data frame.

```
#nombre de lignes et de colonnes dans le data.frame
print(dim(cigarettes))
```

```
## [1] 24 4
```

3. Affichez les noms des observations et des variables.

```
#Affichage des étiquettes des cigarettes
```

```
print(rownames(cigarettes))
```

```
## [1] "Alpine"          "Benson_Hedges"    "Camellights"
## [4] "Carlton"         "Chesterfield"     "GoldenLights"
## [7] "Kent"            "Kool"              "L_M"
## [10] "LarkLights"      "Marlboro"         "Merit"
## [13] "MultiFilter"     "NewportLights"    "Now"
## [16] "OldGold"         "PallMallLight"    "Raleigh"
## [19] "SalemUltra"      "Tareyton"         "TrueLight"
## [22] "ViceroyRichLight" "VirginiaSlims"    "WinstonLights"
```

```
#noms des variables
```

```
print(colnames(cigarettes))
```

```
## [1] "TAR"          "NICOTINE" "WEIGHT"    "CO"
```

4. Calculez les statistiques descriptives pour chaque variable.

```
#Stat. descriptives simples
```

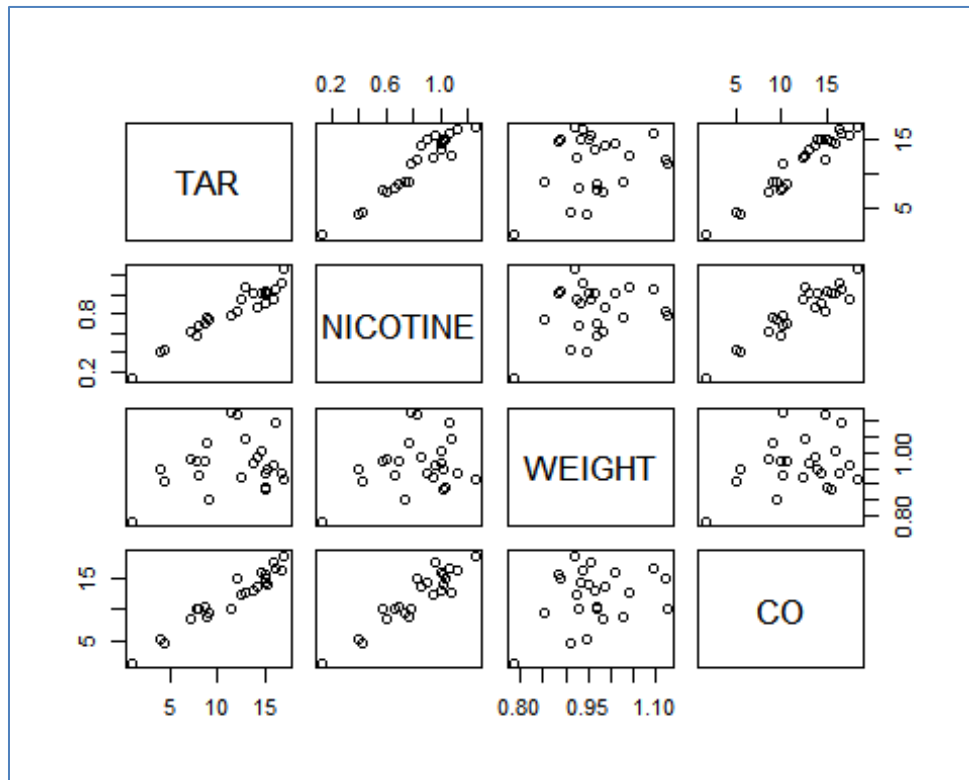
```
print(summary(cigarettes))
```

```
##      TAR          NICOTINE      WEIGHT          CO
##  Min.   : 1.00    Min.   :0.1300   Min.   :0.7851   Min.   : 1.500
## 1st Qu.: 8.45    1st Qu.:0.6850   1st Qu.:0.9215   1st Qu.: 9.875
## Median :12.60    Median :0.8800   Median :0.9535   Median :12.800
## Mean   :11.48    Mean   :0.8283   Mean   :0.9622   Mean   :12.071
## 3rd Qu.:15.03    3rd Qu.:1.0200   3rd Qu.:0.9907   3rd Qu.:15.100
## Max.   :17.00    Max.   :1.2600   Max.   :1.1240   Max.   :18.500
```

5. Réalisez les graphiques nuages de points en croisant deux à deux les variables.

```
#Nuages de points deux à deux
```

```
pairs(cigarettes)
```



Plusieurs variables sont fortement corrélées, en particulier avec la variable cible (endogène) CO.

On distingue quelques points atypiques, par ex. une marque présente une très faible valeur de CO.

Régression linéaire multiple

- Réalisez une régression linéaire multiple expliquant la variable CO à partir de toutes les autres.

```
#Régression linéaire multiple
modele <- lm(CO ~ TAR + NICOTINE + WEIGHT, data = cigarettes)
print(modele)

##
## Call:
## lm(formula = CO ~ TAR + NICOTINE + WEIGHT, data = cigarettes)
##
## Coefficients:
## (Intercept)      TAR      NICOTINE      WEIGHT
##    -0.5517     0.8876     0.5185     2.0793

#objet summary
sm <- summary(modele)
print(sm)
```

```
##
## Call:
## lm(formula = CO ~ TAR + NICOTINE + WEIGHT, data = cigarettes)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.1083 -0.8046 -0.1199  1.0095  2.0501
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.5517     2.9713   -0.186  0.854569
## TAR           0.8876     0.1955    4.540  0.000199 ***
## NICOTINE      0.5185     3.2523    0.159  0.874941
## WEIGHT        2.0793     3.1784    0.654  0.520431
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.16 on 20 degrees of freedom
## Multiple R-squared:  0.935, Adjusted R-squared:  0.9252
## F-statistic: 95.86 on 3 and 20 DF, p-value: 4.85e-12
```

Le seul coefficient significatif à 5% celui de TAR ($p\text{-value} = \Pr(>|t|) < 0.05$). On pouvait s'y attendre, la variable est corrélée avec CO dans le graphique.

Etonnement, NICOTINE qui est manifestement très corrélée avec CO également (cf. graphique ci-dessus) n'apparaît pas comme pertinent dans la régression. Pourquoi ?

8. Affichez le champ `$coefficients` de l'objet issu de `summary()`.

```
#coefficients
print(sm$coefficients)

##              Estimate Std. Error    t value    Pr(>|t|)
## (Intercept) -0.5516976  2.9712809 -0.1856767  0.8545685010
## TAR          0.8875803  0.1954817  4.5404782  0.0001990908
## NICOTINE     0.5184696  3.2523311  0.1594148  0.8749410220
## WEIGHT       2.0793442  3.1784171  0.6542075  0.5204306639
```

Quel est le type de cet objet ?

```
#classe de $coefficients
print(class(sm$coefficients))

## [1] "matrix"
```

Quelles sont ses dimensions ?

```
#dimensions
print(dim(sm$coefficients))

## [1] 4 4
```

9. Affichez les écarts-type des coefficients estimés.

```
#ecarts-type des coefficients estimés
print(sm$coefficients[,2])

## (Intercept)          TAR      NICOTINE      WEIGHT
##  2.9712809    0.1954817    3.2523311    3.1784171
```

10. Pour chaque coefficient, calculez son intervalle de confiance au niveau 95%.

```
#quantile de la loi de Student
qs <- qt(0.975,24-3-1)

#bornes basses
print("Bornes basses")

## [1] "Bornes basses"

print(sm$coefficients[,1]-qs*sm$coefficients[,2])

## (Intercept)          TAR      NICOTINE      WEIGHT
## -6.7496811    0.4798127 -6.2657743 -4.5507177

#bornes hautes
print("Bornes hautes")

## [1] "Bornes hautes"

print(sm$coefficients[,1]+qs*sm$coefficients[,2])

## (Intercept)          TAR      NICOTINE      WEIGHT
##  5.646286    1.295348    7.302713    8.709406
```

Analyse des résidus

11. Récupérez les résidus de la régression (\$residuals). Calculez sa moyenne. Que constatez-vous ?

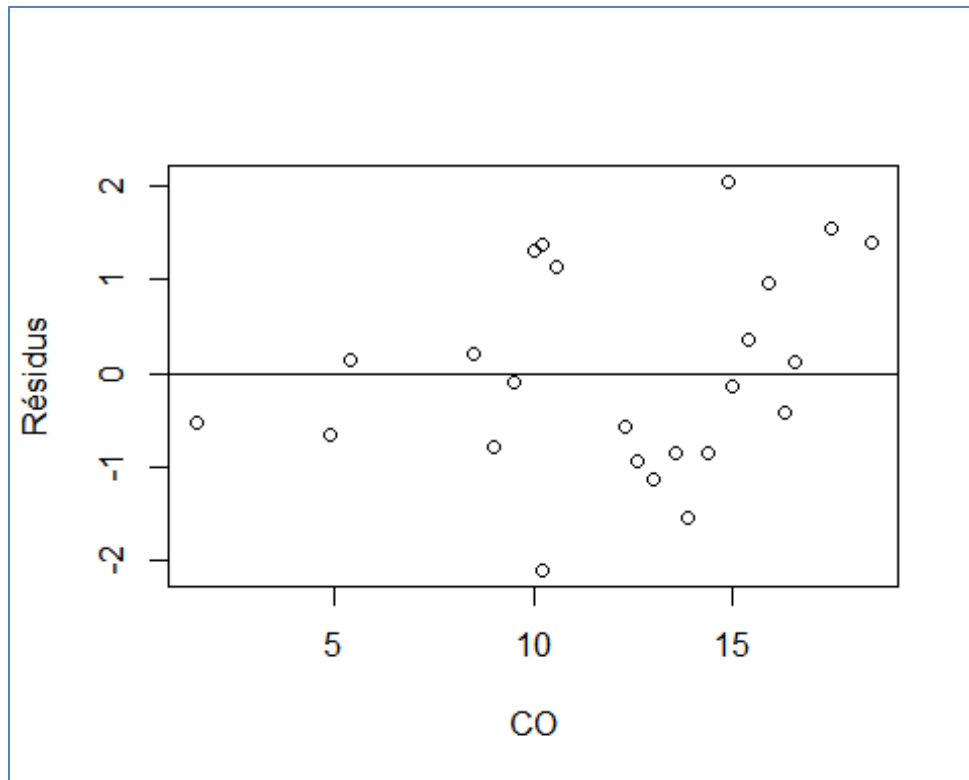
```
#Résidus
e <- modele$residuals #ou encore e <- residuals(modele)
print(mean(e))

## [1] -6.128453e-17
```

Dans la régression avec constante, la moyenne des résidus est nulle forcément, sinon problème.

12. Construisez le graphique nuage de points en croisant en abscisse la variable cible (CO) et en ordonnée le résidu (plot). Y a-t-il des éléments saillants dans le graphique ?

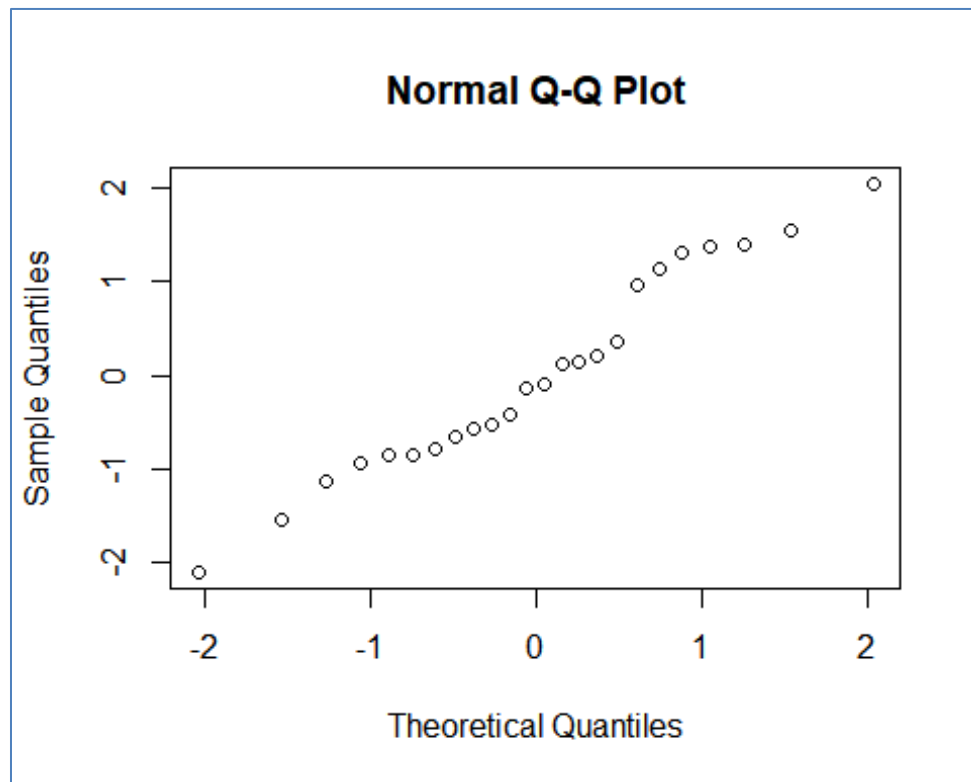
```
#Graphique des résidus
plot(cigarettes$CO,e,ylab="Résidus",xlab="CO")
abline(h=0)
```



Pas d'éléments réellement choquants. Peut-être : 1 valeur de C0 très faible, 2 résidus élevés en valeur absolue (1 négatif autour de C0 = 10 ; 1 positif autour de CO = 15).

13. Réalisez la droite de Henry pour vérifier la compatibilité des résidus avec l'hypothèse de normalité (qqnorm). Que constatez-vous ?

#Droite de Henry
`qqnorm(e)`



On a quelque chose qui ressemble fortement à une droite. La compatibilité avec la loi normale est viable.

14. Test de Jarque-Bera

```
#asymétrie
g1 <- mean(e^3)/(mean(e^2)^1.5)
print(g1)

## [1] 0.1608395

#aplatissement
g2 <- mean(e^4)/(mean(e^2)^2)-3
print(g2)

## [1] -0.8123179

#stat. de test du test de normalité de Jarque-bera
Tjb <- ((24-3-1)/6)*(g1^2+(g2^2)/4)
print(Tjb)

## [1] 0.6361148

#p-value du test de Jarque-Bera
print(pchisq(Tjb,2,lower.tail = FALSE))

## [1] 0.727561
```

Pour un test à $\alpha = 5\%$, la p -value $> \alpha$, on ne peut pas rejeter l'hypothèse de normalité des résidus. Ce résultat est cohérent avec la droite de Henry ci-dessus.

Détection des points atypiques et influents

15. Calculez le résidu studentisé de la régression.

```
#Résidus studentisés
res.student <- rstudent(modele)
print(res.student)
```

##	Alpine	Benson_Hedges	CamellLights	Carlton
##	-0.8050342	0.1177432	1.2568137	0.1268819
##	Chesterfield	GoldenLights	Kent	Kool
##	-0.1343193	-0.7313460	-0.5032631	-0.3699537
##	L_M	LarkLights	Marlboro	Merit
##	0.3223152	-1.0210174	-0.8339836	1.2051387
##	MultiFilter	NewportLights	Now	OldGold
##	-2.3367957	-0.0833822	-0.6190598	1.4347442
##	PallMallLight	Raleigh	SalemUltra	Tareyton
##	-0.9685787	1.5279745	-0.6023223	0.8557047
##	TrueLight	ViceroyRichLight	VirginiaSlims	WinstonLights
##	0.1920721	1.0249592	-1.4249196	2.2090210

16. Calculez le seuil critique pour le résidu studentisé pour un risque de 10%.

```
#Seuil critique
#risque alpha = 0.1
alpha <- 0.1
#calcul du seuil à partir de La loi de Student à (n-p-2) ddl ==> n = 24 obs., p = 3 explicatives
seuil.student <- qt(1-alpha/2, 24-3-2)
print(seuil.student)
```

```
## [1] 1.729133
```

Attention en degré de liberté de la loi de Student : $n - p - 2$!

17. Quelles sont les marques de cigarette atypiques au sens de ce seuil ?

```
#détection des cigarettes en dehors des tuyaux
#vecteur de booléen indiquant les atypiques
atypiques.rstudent <- (res.student < -seuil.student | res.student > +seuil.student)
ab.student <- cigarettes[atypiques.rstudent,]
print(ab.student)
```

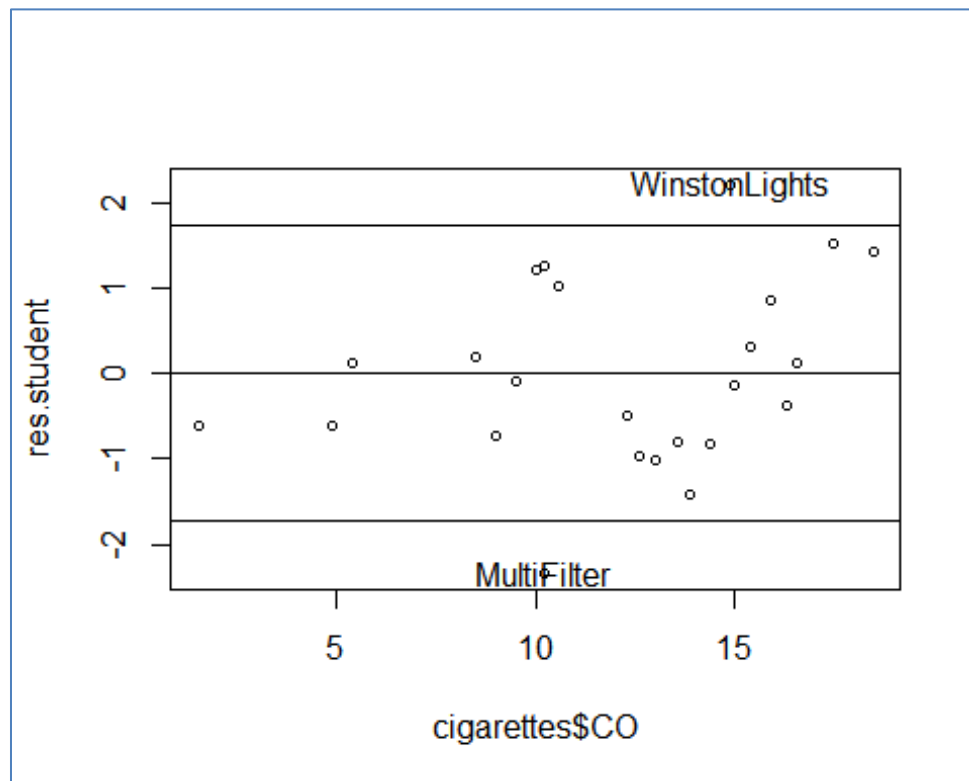
##		TAR	NICOTINE	WEIGHT	CO
##	MultiFilter	11.4	0.78	1.1240	10.2
##	WinstonLights	12.0	0.82	1.1184	14.9

18. Construisez le graphique nuage de points croisant en abscisse la variable cible CO et en ordonnée le résidu studentisé (plot). Insérez dans le graphique les lignes matérialisant les seuils critiques (text). Faites apparaître nommément les cigarettes atypiques.


```

#mettre en évidence les points atypiques dans le graphique des résidus
#construction du graphique des résidus studentisés
plot(cigarettes$CO, res.student, cex=0.75)
abline(h=-seuil.student)
abline(h=+seuil.student)
abline(h=0)
text(cigarettes$CO[atypiques.rstudent], res.student[atypiques.rstudent], rownames(cigarettes)[atypiques.rstudent])

```



19. Calculez le levier de chaque observation.

```

#levier
indicateurs <- influence.measures(modele)

#quels sont les descripteurs disponibles
attributes(indicateurs)

## $names
## [1] "infmat" "is.inf" "call"
##
## $class
## [1] "infl"

#on s'intéresse à la matrice infmat
print(indicateurs$infmat)

##               dfb.1_      dfb.TAR      dfb.NICO      dfb.WEIG
## Alpine          -0.007274016 -0.312949909  0.2972719267 -0.039062192
## Benson_Hedges   -0.039521859  0.016699581 -0.0126680153  0.039361311

```

```
## Camellights      0.103072842 -0.209493674  0.1608099548 -0.067260624
## Carlton         0.004712591 -0.013104839 -0.0012137939  0.008306493
## Chesterfield    -0.031493480 -0.004581011 -0.0053652160  0.036407301
## GoldenLights     0.154417052  0.218964583 -0.1846045951 -0.160721742
## Kent            -0.052049201  0.089117547 -0.1054756298  0.069698261
## Kool            -0.039613402 -0.032513283  0.0013076328  0.054970848
## L_M             0.079718491  0.023814654 -0.0008303217 -0.088730412
## LarkLights      0.001585024  0.120871085 -0.1612834966  0.035028003
## Marlboro        -0.151236528 -0.399689403  0.3592648351  0.108804448
## Merit           -0.001668270  0.090231526 -0.1670523168  0.102565082
## MultiFilter      1.066309955 -0.274093400  0.3942149435 -1.225015025
## NewportLights   -0.025172722  0.015573351 -0.0152699105  0.025044063
## Now            -0.403947154 -0.102743819  0.2049852480  0.273993017
## OldGold         0.179157637 -0.372221039  0.5443835325 -0.354133679
## PallMallLight   0.288617300  0.538596558 -0.5649064659 -0.196623758
## Raleigh         0.147730615  0.638380856 -0.5549434297 -0.096350384
## SalemUltra     -0.084905923  0.050552156  0.0078896160  0.028097721
## Tareyton        -0.083225134  0.015241129  0.0148887112  0.070569311
## TrueLight       -0.011060363 -0.024490066  0.0120087376  0.021336064
## ViceroyRichLight -0.023925121 -0.122870283  0.0805826669  0.059239152
## VirginiaSlims   -0.104528751 -0.169015536  0.0890555584  0.123648970
## WinstonLights   -0.954528028  0.240701536 -0.3256172454  1.077491904
##                  dffit      cov.r      cook.d      hat
## Alpine          -0.36527775  1.2946257  0.0339544205  0.17073128
## Benson_Hedges   0.05608100  1.5018754  0.0008270487  0.18491182
## Camellights     0.38481248  0.9756425  0.0359775671  0.08571162
## Carlton        0.05678186  1.4686351  0.0008477501  0.16685544
## Chesterfield    -0.05272054  1.4115110  0.0007307419  0.13349209
## GoldenLights    -0.31590879  1.3036985  0.0255436513  0.15724567
## Kent            -0.16578650  1.2907586  0.0071377900  0.09789596
## Kool            -0.13636091  1.3550670  0.0048582398  0.11960820
## L_M             0.12657369  1.3864989  0.0041930994  0.13360991
## LarkLights      -0.29345872  1.0734606  0.0214838777  0.07630555
## Marlboro        -0.45768063  1.3835125  0.0531774394  0.23146025
## Merit           0.39006700  1.0102180  0.0371967516  0.09482777
## MultiFilter     -1.38484974  0.6039106  0.3920197606  0.25992124
## NewportLights   -0.03576440  1.4514831  0.0003364802  0.15538649
## Now            -0.57176743  2.1003985  0.0843300908  0.46034869
## OldGold         0.84001591  1.0924928  0.1675397193  0.25528101
## PallMallLight   -0.65782478  1.4794832  0.1085189838  0.31566149
## Raleigh         0.75166755  0.9591703  0.1324143176  0.19484847
## SalemUltra     -0.25548669  1.3430757  0.0168553813  0.15248468
## Tareyton        0.23247757  1.1332941  0.0136948072  0.06873642
## TrueLight       0.06262032  1.3477388  0.0010299226  0.09607980
## ViceroyRichLight 0.28018918  1.0639337  0.0195770222  0.06953294
## VirginiaSlims   -0.43010756  0.8924817  0.0439821752  0.08350342
## WinstonLights   1.22624895  0.6436587  0.3148452146  0.23555978
```

#on récupère la colonne "hat" qui correspond au levier

```
res.hat <- indicateurs$informat[, "hat"]
```

```
print(res.hat)
```

```
##           Alpine      Benson_Hedges      Camellights      Carlton
##           0.17073128      0.18491182      0.08571162      0.16685544
##           Chesterfield      GoldenLights      Kent      Kool
```

```
##      0.13349209      0.15724567      0.09789596      0.11960820
##      L_M      LarkLights      Marlboro      Merit
##      0.13360991      0.07630555      0.23146025      0.09482777
##      MultiFilter      NewportLights      Now      OldGold
##      0.25992124      0.15538649      0.46034869      0.25528101
##      PallMallLight      Raleigh      SalemUltra      Tareyton
##      0.31566149      0.19484847      0.15248468      0.06873642
##      TrueLight ViceroyRichLight      VirginiaSlims      WinstonLights
##      0.09607980      0.06953294      0.08350342      0.23555978
```

20. Quels sont les points atypiques au sens du levier ?

```
#Le seuil est défini par  $2x(p+1)/n \Rightarrow p = 3$  expl.,  $n = 24$  obs.
seuil.hat <- 2*(3+1)/24
print(seuil.hat)

## [1] 0.3333333

#Les points atypiques au sens du levier
atypiques.levier <- (res.hat > seuil.hat)
ab.hat <- cigarettes[atypiques.levier,]
print(ab.hat)

##      TAR NICOTINE WEIGHT  CO
## Now      1      0.13 0.7851 1.5
```

21. Créez un nouveau data frame excluant les observations atypiques au sens du résidu studentisé OU du levier.

```
#supprimer les points atypiques de la base
#identifier les éléments à exclure
excluded <- (atypiques.rstudent | atypiques.levier)
print(excluded)

##      Alpine      Benson_Hedges      CamellLights      Carlton
##      FALSE      FALSE      FALSE      FALSE
##      Chesterfield      GoldenLights      Kent      Kool
##      FALSE      FALSE      FALSE      FALSE
##      L_M      LarkLights      Marlboro      Merit
##      FALSE      FALSE      FALSE      FALSE
##      MultiFilter      NewportLights      Now      OldGold
##      TRUE      FALSE      TRUE      FALSE
##      PallMallLight      Raleigh      SalemUltra      Tareyton
##      FALSE      FALSE      FALSE      FALSE
##      TrueLight ViceroyRichLight      VirginiaSlims      WinstonLights
##      FALSE      FALSE      FALSE      TRUE
```

Les TRUE sont ceux à exclure c.-à-d. MULTIFILTER, NOW et WINSTONLIGHTS.

```
#nouveau data frame : on garde les non-exclus ==> !excluded
cigarettes.clean <- cigarettes[!excluded,]
print(cigarettes.clean)

##      TAR NICOTINE WEIGHT  CO
## Alpine      14.1      0.86 0.9853 13.6
```

```
## Benson_Hedges      16.0      1.06 1.0938 16.6
## Camellights         8.0      0.67 0.9280 10.2
## Carlton            4.1      0.40 0.9462  5.4
## Chesterfield       15.0      1.04 0.8885 15.0
## GoldenLights        8.8      0.76 1.0267  9.0
## Kent               12.4      0.95 0.9225 12.3
## Kool               16.6      1.12 0.9372 16.3
## L_M                14.9      1.02 0.8858 15.4
## LarkLights         13.7      1.01 0.9643 13.0
## Marlboro           15.1      0.90 0.9316 14.4
## Merit              7.8      0.57 0.9705 10.0
## NewportLights       9.0      0.74 0.8517  9.5
## OldGold            17.0      1.26 0.9186 18.5
## PallMallLight      12.8      1.08 1.0395 12.6
## Raleigh            15.8      0.96 0.9573 17.5
## SalemUltra         4.5      0.42 0.9106  4.9
## Tareyton           14.5      1.01 1.0070 15.9
## TrueLight          7.3      0.61 0.9806  8.5
## ViceroyRichLight    8.6      0.69 0.9693 10.6
## VirginiaSlims      15.2      1.02 0.9496 13.9
```

```
#dimension
print(dim(cigarettes.clean))

## [1] 21  4
```

22. Réalisez de nouveau la régression CO vs. les autres variables à partir de ce nouvel ensemble de données. Quelle est la valeur du R2 maintenant ?

```
#Nouvelle régression
modele.clean <- lm(CO ~ ., data = cigarettes.clean)
sm.clean <- summary(modele.clean)
print(sm.clean)

##
## Call:
## lm(formula = CO ~ ., data = cigarettes.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.5273 -0.7626 -0.1690  1.0397  1.5323
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.7516     3.9392   0.191   0.851
## TAR           0.9094     0.1765   5.153 7.97e-05 ***
## NICOTINE      -0.2513     3.0709  -0.082   0.936
## WEIGHT        1.1682     4.1141   0.284   0.780
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.02 on 17 degrees of freedom
## Multiple R-squared:  0.9382, Adjusted R-squared:  0.9273
## F-statistic: 86.06 on 3 and 17 DF, p-value: 1.759e-10
```

Sélection de variables

23. Testez la significativité simultanée des coefficients de NICOTINE et WEIGHT en opposant les R2 des régressions $CO = f(TAR, NICOTINE, WEIGHT)$ et $CO = f(TAR)$

```
#régression avec TAR seulement
modele.simplified <- lm(CO ~ TAR, data = cigarettes.clean)
sm.simplified <- summary(modele.simplified)
print(sm.simplified)

##
## Call:
## lm(formula = CO ~ TAR, data = cigarettes.clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.5337 -0.6918 -0.2543  1.0877  1.5280
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.79658     0.66766   2.691   0.0145 *
## TAR          0.89718     0.05296  16.942 6.35e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9669 on 19 degrees of freedom
## Multiple R-squared:  0.9379, Adjusted R-squared:  0.9346
## F-statistic: 287 on 1 and 19 DF, p-value: 6.349e-13

#F du test
FTest <- ((sm.clean$r.squared-sm.simplified$r.squared)/2)/((1-
sm.clean$r.squared)/(21-3-1))
print(FTest)

## [1] 0.04223092

#p-value
print(pf(FTest,2,21-3-1,lower.tail=FALSE))

## [1] 0.9587486
```

On ne peut pas rejeter l'hypothèse selon laquelle les coefficients de NICOTINE et WEIGHT sont simultanément nuls.

24. Réalisez une sélection de variables « backward » optimisant le critère AIC.

```
#Sélection de variables
library(MASS)
modele.reduit <- stepAIC(modele.clean,direction="backward")

## Start:  AIC=4.38
## CO ~ TAR + NICOTINE + WEIGHT
##
##              Df Sum of Sq    RSS    AIC
```

```
## - NICOTINE 1 0.0070 17.682 2.3880
## - WEIGHT 1 0.0838 17.758 2.4790
## <none> 17.675 4.3797
## - TAR 1 27.6042 45.279 22.1346
##
## Step: AIC=2.39
## CO ~ TAR + WEIGHT
##
## Df Sum of Sq RSS AIC
## - WEIGHT 1 0.081 17.762 0.484
## <none> 17.682 2.388
## - TAR 1 265.058 282.740 58.600
##
## Step: AIC=0.48
## CO ~ TAR
##
## Df Sum of Sq RSS AIC
## <none> 17.762 0.484
## - TAR 1 268.34 286.103 56.848

summary(modele.reduit)

##
## Call:
## lm(formula = CO ~ TAR, data = cigarettes.clean)
##
## Residuals:
## Min 1Q Median 3Q Max
## -1.5337 -0.6918 -0.2543 1.0877 1.5280
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.79658 0.66766 2.691 0.0145 *
## TAR 0.89718 0.05296 16.942 6.35e-13 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9669 on 19 degrees of freedom
## Multiple R-squared: 0.9379, Adjusted R-squared: 0.9346
## F-statistic: 287 on 1 and 19 DF, p-value: 6.349e-13
```

OUI, ce résultat est cohérent avec le test mené précédemment. Seule la variable TAR est conservée dans le modèle.

Prédiction sur un nouveau fichier

25. Charger les données du fichier « autres_cigarettes.txt ». Combien y a-t-il de marques de cigarettes dans ce fichier ?

```
#chargement du second fichier
autres <-
read.table(file="autres_cigarettes.txt", sep="\t", header=TRUE, dec=".", row.names=1)
print(nrow(autres))
```

```
## [1] 4
```

26. Pour ces nouvelles observations, calculez les prédictions ponctuelles ainsi que leurs intervalles de confiance à 90% du modèle simplifié.

```
#prédictions (fit) incluant Les intervalles de confiance (lwr, upr)
pred <- predict(modele.reduit,newdata=autres,interval="prediction",level=0.9)
print(pred)
```

```
##           fit           lwr           upr
## Benz      14.446830 12.724455 16.169205
## GoodLook  17.945837 16.147522 19.744151
## Riverplate 9.871206 8.138635 11.603777
## Melia     5.475019 3.618546 7.331491
```

27. Sachant les vraies valeurs de l'endogène sont respectivement...

```
#vraies valeurs de l'endogene
true_endo <- c(13.5,21.3,8.25,6.0)
names(true_endo) <- c("Benz","GoodLook","RiverPlate","Melia")

#verification
quid <- (true_endo >= pred[, 'lwr']) & (true_endo < pred[, 'upr'])
print(quid)
```

```
##      Benz      GoodLook RiverPlate      Melia
##      TRUE      FALSE      TRUE      TRUE
```

Les intervalles couvrent - au niveau de confiance 90% - la "vraie" valeur de l'endogène pour BENZ, RIVERPLATE et MELIA.

28. Accolez ces nouvelles variables (prédictions et bornes des intervalles de prédiction) au jeu de données "autres_cigarettes"

```
#data frame avec la prédiction et Les résidus
autres.plus <- cbind(autres,pred)
print(summary(autres.plus))
```

```
##      TAR           NICOTINE           WEIGHT           fit
## Min.   : 4.100   Min.   :0.4000   Min.   :0.8760   Min.   : 5.475
## 1st Qu.: 7.775   1st Qu.:0.6025   1st Qu.:0.9150   1st Qu.: 8.772
## Median :11.550   Median :0.7650   Median :0.9566   Median :12.159
## Mean   :11.300   Mean   :0.7475   Mean   :0.9671   Mean   :11.935
## 3rd Qu.:15.075   3rd Qu.:0.9100   3rd Qu.:1.0087   3rd Qu.:15.322
## Max.   :18.000   Max.   :1.0600   Max.   :1.0790   Max.   :17.946
##      lwr           upr
## Min.   : 3.619   Min.   : 7.331
## 1st Qu.: 7.009   1st Qu.:10.536
## Median :10.432   Median :13.886
## Mean   :10.157   Mean   :13.712
## 3rd Qu.:13.580   3rd Qu.:17.063
## Max.   :16.148   Max.   :19.744
```

29. Sauvegardez ce nouvel ensemble de données (data frame) dans le fichier "output_regression.txt".

#sauvegarde

```
#write.table(autres.plus,file="output_regression.txt",quote=F, sep="\t",dec=".",row.names=T,col.names=T)
```