Travaux dirigés - Régression linéaire multiple - Correction

Julien Chiquet et Guillem Rigaill

15 octobre 2015

Vote au présidentielles US de 2000 en Georgie

1. Préliminaires

Chargement des package qui seront utiles.

```
library(ggplot2); library(GGally)
library(car)
library(leaps)
```

- Charger le jeu de données `gavote` à la page [http://julien.cremeriefamily.info/reglin/gavote.txt]

```
## Lecture du tableau de données
gavote <- read.table(file = "gavote.txt", header=TRUE)
```

- Créer la variable `undercount` (proportion de bulletins de vote considérés comme nuls) et l'ajouter à la table. Nous allons nous intéresser à la prédiction de cette variable par les autres. Elle porte donc le statut de *variable réponse*, ou *à expliquer*. Supprimer les variables `votes` et `ballots` du tableau (expliquer pourquoi).

```
## Création de la variable `undercount` (proportion de votes nuls)
gavote$undercount <- (gavote$ballots - gavote$votes)/gavote$ballots
```

- Créer les variables `pergore`, `perbush` et `perother` (pourcentage de votants pour Gore, Bush et pour d'autres candidats). Les ajouter à la table. Supprimer les variables `gore`, `bush` et `other` du tableau.

```
## Création des variables `pergore`, `perbush`et ``perothers` (proportion de votes par candidat)
gavote$pergore <- gavote$gore/gavote$votes
gavote$perbush <- gavote$bush/gavote$votes
gavote$perother <- gavote$other/gavote$votes

## Suppression de variables redondantes/inutiles
gavote$gore <- NULL
gavote$bush <- NULL
gavote$other <- NULL
```

On supprime également les variables `votes` et `ballots` car prédire la proportion de bulletins nuls à partir de ces variables n'a aucun intérêt et risque de masquer l'effet des autres.

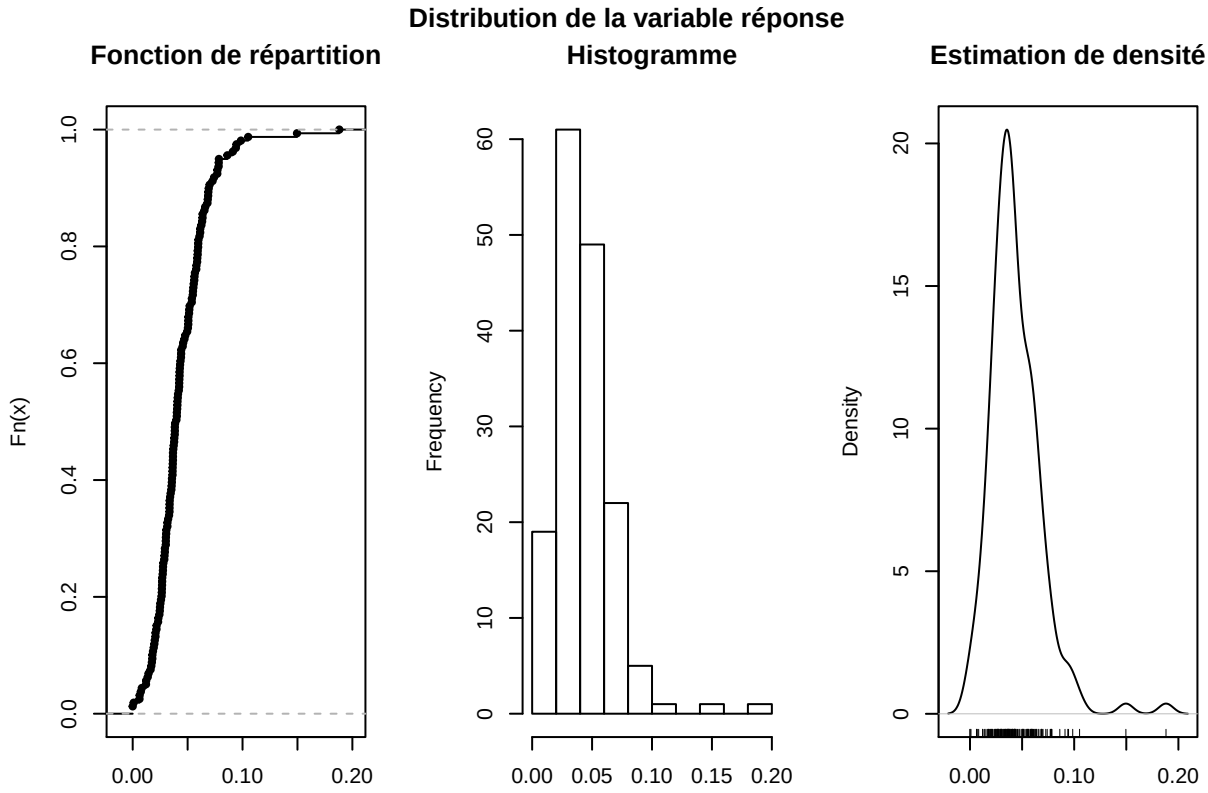
```
gavote$votes   <- NULL
gavote$ballots <- NULL
```

2. Analyse descriptive

Faire une analyse descriptive des données. Cette analyse vous aidera dans vos choix de modèles et pour l'interprétation des résultats.

Commençons par nous intéresser à la variable à expliquer (*undercount*) et à sa répartition.

```
par(mfrow=c(1,3))
plot(ecdf(gavote$undercount), main="Fonction de répartition", xlab="")
hist(gavote$undercount,main="Histogramme",xlab="")
plot(density(gavote$undercount),main="Estimation de densité", xlab="")
rug(gavote$undercount)
title(outer=TRUE, main ="\nDistribution de la variable réponse", sub="Pourcentage de votes nuls")
```

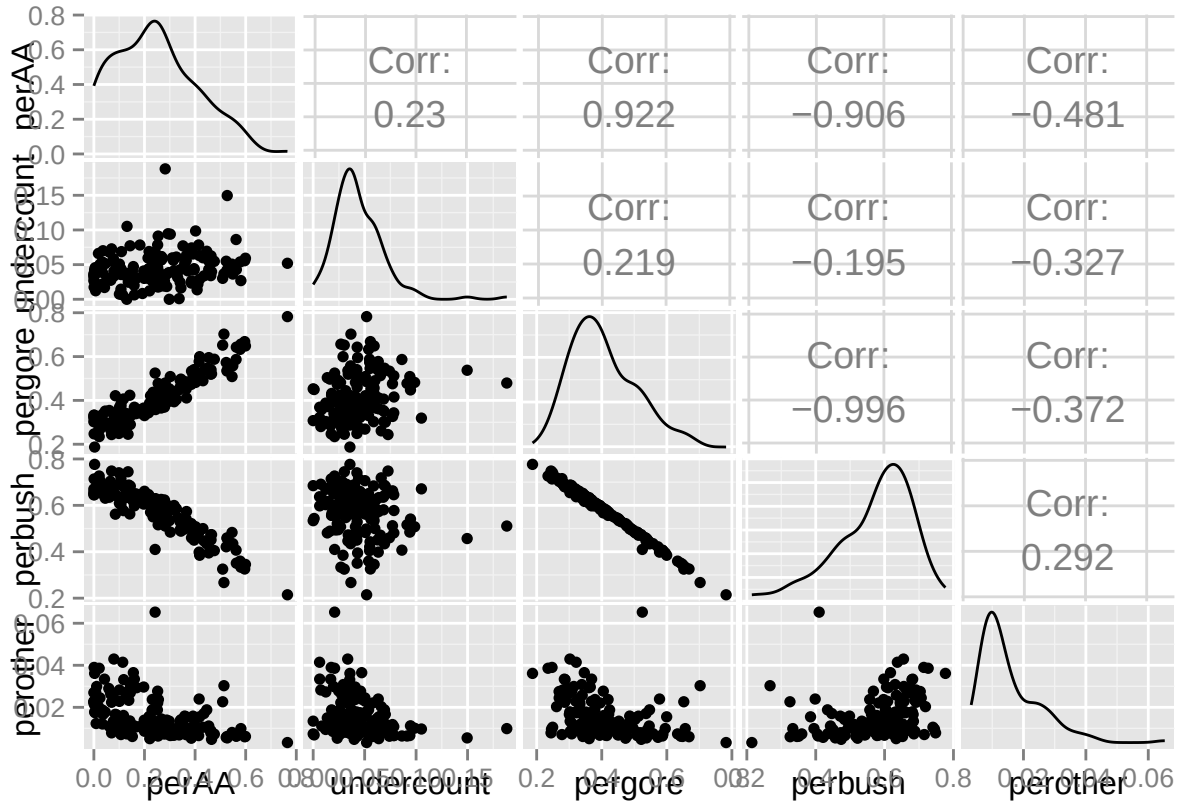


Dans le tableau, on trouve des variables factorielles et des variables quantitatives. L'analyse descriptive doit être fait en conséquence.

```
are.factor  <- sapply(gavote, is.factor)
```

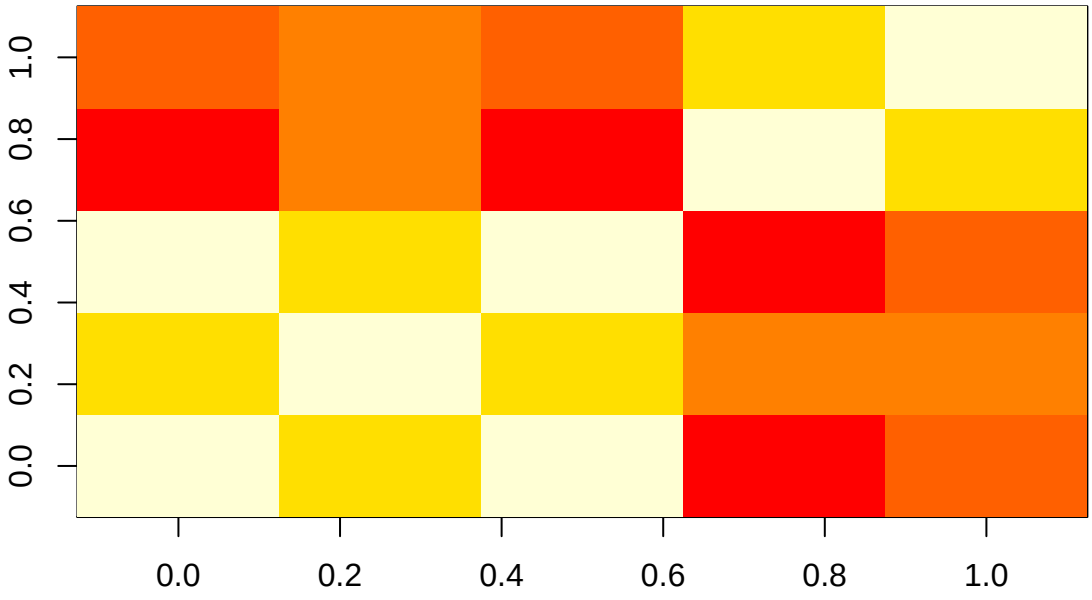
Représentons les graphes pair à pair entre les variables numériques

```
ggpairs(gavote, columns = which(!are.factor))
```

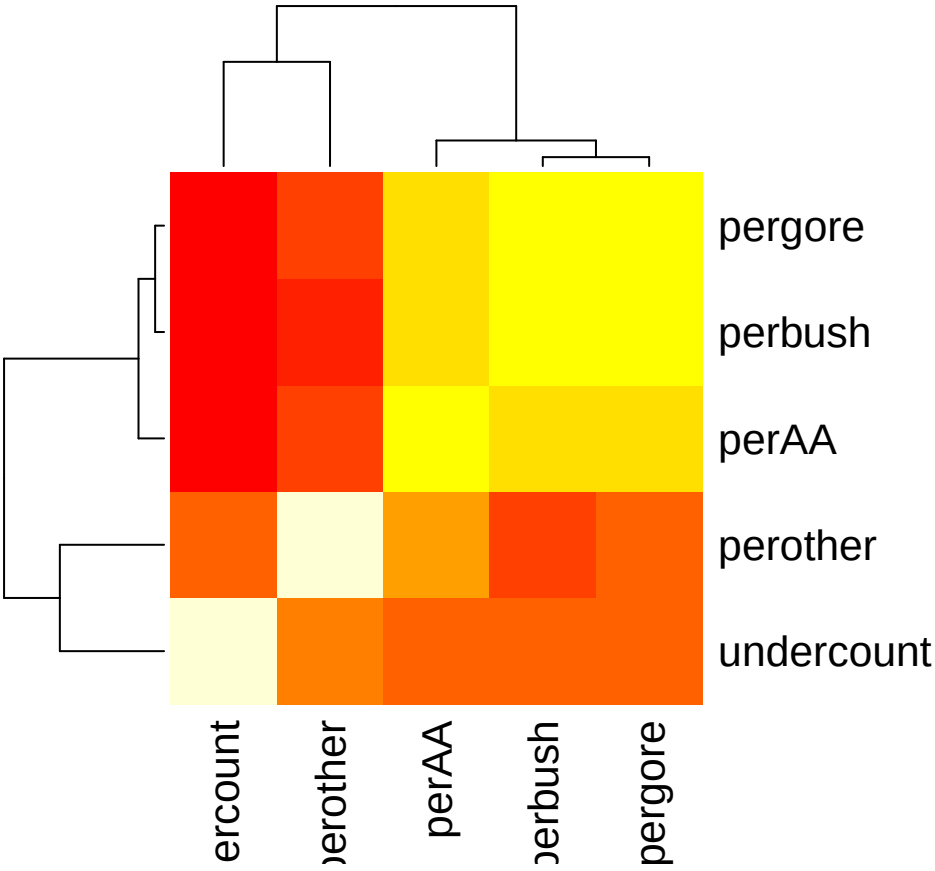


Voyons la matrice des corrélation entre variables quantitative:

```
image(cor(gavote[, !are.factor]))
```



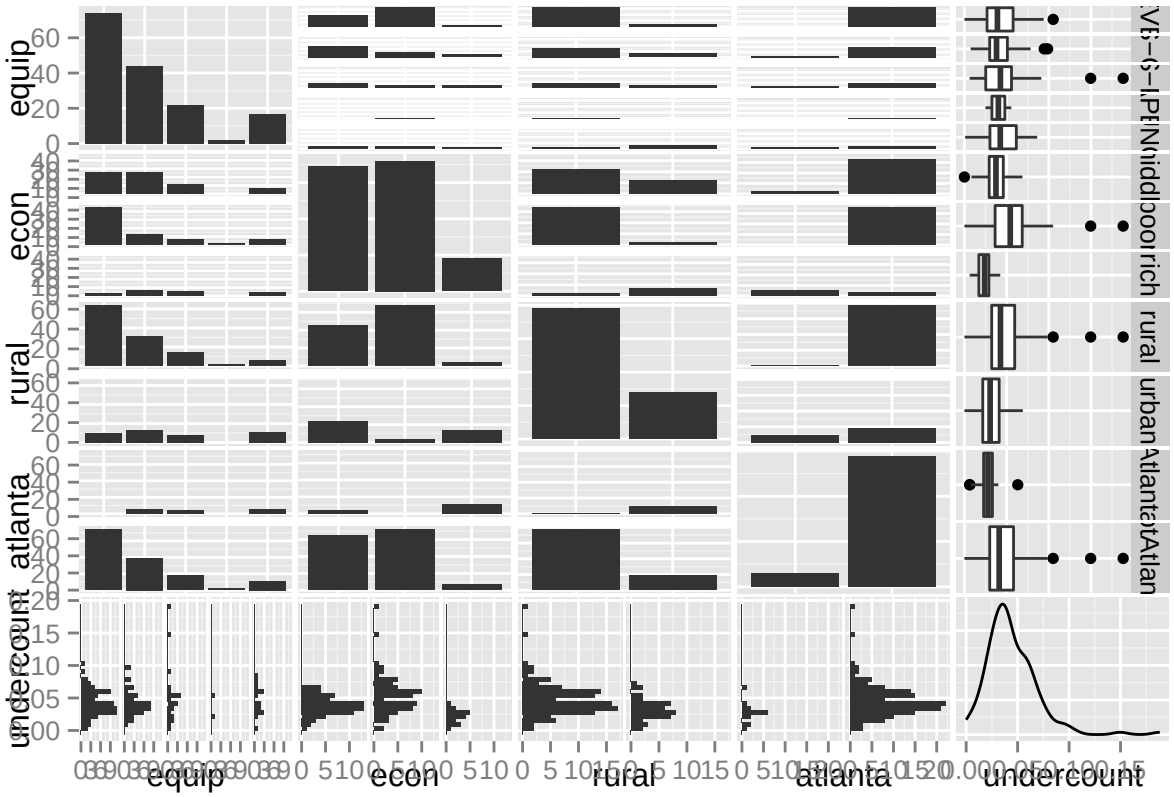

```
heatmap(abs(cor(gavote[, !are.factor])))
```



Enfin, voyons comment se répartit la variable à expliquer dans les différentes catégories de variables catégorielles.

```
ggpairs(gavote, columns = which(colnames(gavote) == "undercount" | are.factor))
```

```
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
```

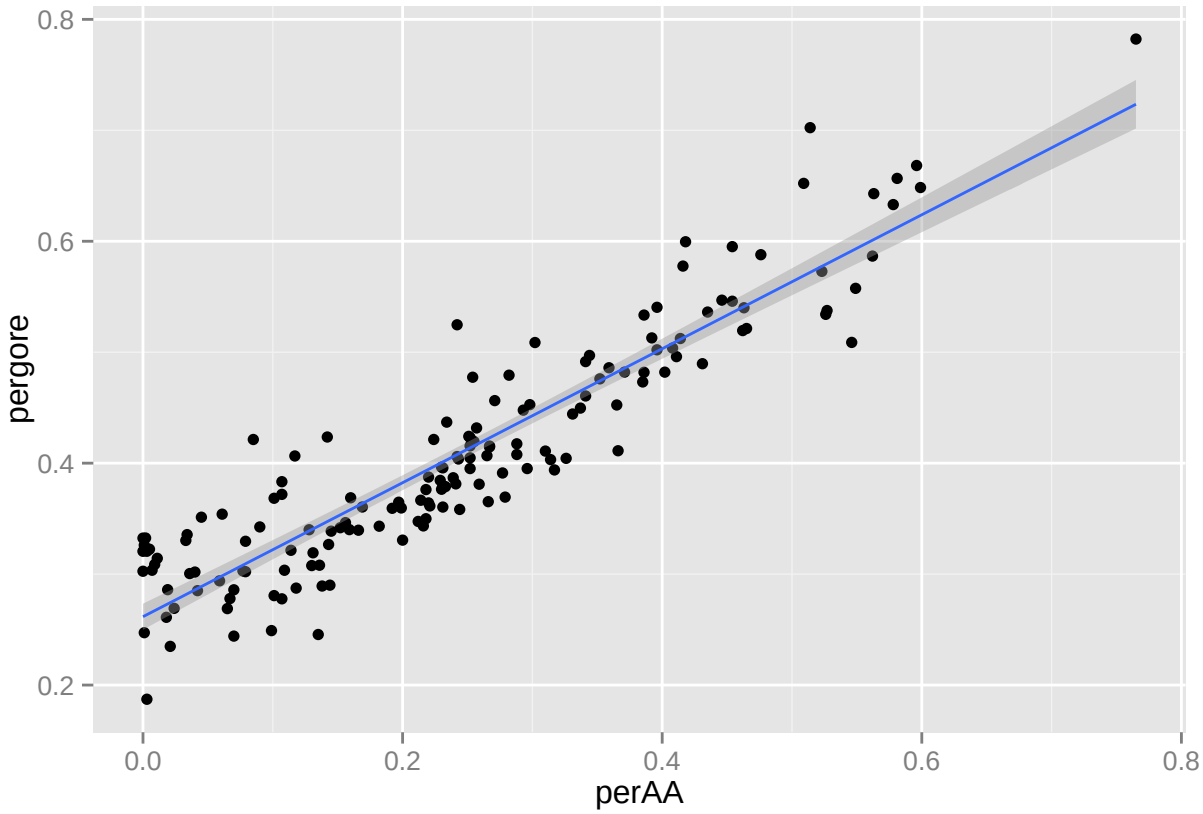


3. Quelques modèles linéaires simples

Votes parmi les Afro-Américains

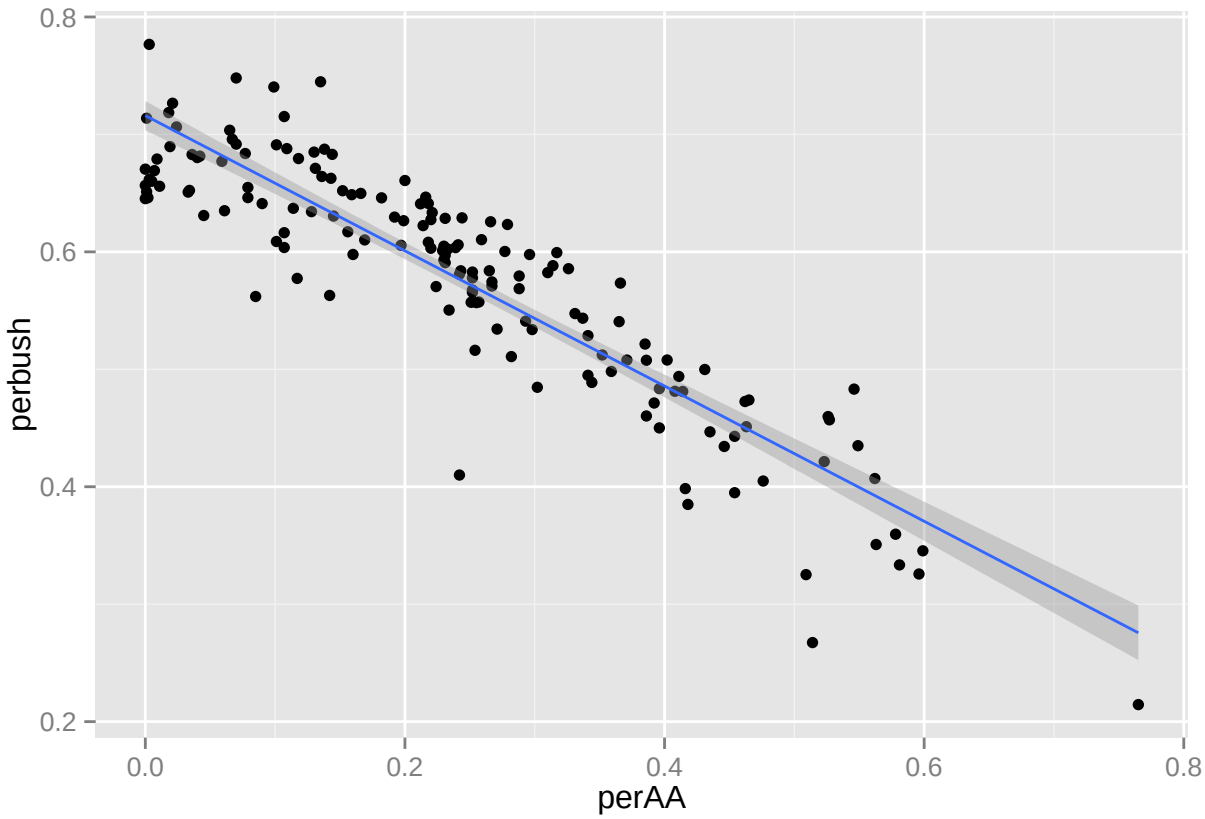
- Tracer le diagramme de dispersion croisant `pergore` avec `perAA`. Ajuster un modèle de régression simple et tracer la droite de régression entre `pergore` et `perAA`.

```
ggplot(gavote, aes(x=perAA,y=pergore)) + geom_point() + stat_smooth(method="lm", formula=y~x)
```



On aurait pu tracer la proportion de vote pour Bush parmi les afro-américains et faire des remarques similaires.

```
ggplot(gavote, aes(x=perAA,y=perbush)) + geom_point() + stat_smooth(method="lm", formula=y~x)
```



- Régresser `undercount` sur `perAA` puis sur `pergore`, et enfin sur `pergore` *et* `perAA`. Comparer ces modèles et interpréter les résultats.

```
M0 <- lm(undercount~1,gavote)
M11 <- lm(undercount~perAA,gavote)
M12 <- lm(undercount~pergore,gavote)
M2 <- lm(undercount~pergore+perAA,gavote)
anova(M0,M12,M2)
```

```
## Analysis of Variance Table
##
## Model 1: undercount ~ 1
## Model 2: undercount ~ pergore
## Model 3: undercount ~ pergore + perAA
##   Res.Df      RSS Df Sum of Sq      F   Pr(>F)
## 1    158 0.098477
## 2    157 0.093764  1 0.0047129 7.8845 0.005623 **
## 3    156 0.093249  1 0.0005151 0.8617 0.354701
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(M0,M11,M2)
```

```
## Analysis of Variance Table
##
## Model 1: undercount ~ 1
```

```
## Model 2: undercount ~ perAA
## Model 3: undercount ~ pergore + perAA
##   Res.Df      RSS Df Sum of Sq      F   Pr(>F)
## 1    158 0.098477
## 2    157 0.093282  1 0.0051953 8.6914 0.003689 **
## 3    156 0.093249  1 0.0000327 0.0547 0.815309
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

`pergore` n'apporte rien à `perAA` et réciproquement.

Votes selon le niveau de vie et ANOVA à un facteur

Pour étudier l'effet du facteur `econ` sur la variable `undercount`, on propose le modèle d'ANOVA 1 suivant :

$$Y_i = \mu + \mathbf{1}_{\{i \in k\}} \mu_k + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2),$$

où Y_i décrit la variable `undercount` associée au $i$ème individu, μ l'intercept et μ_k un terme additif associé au groupe k (c'est-à-dire à la modalité k de la variable `econ`).

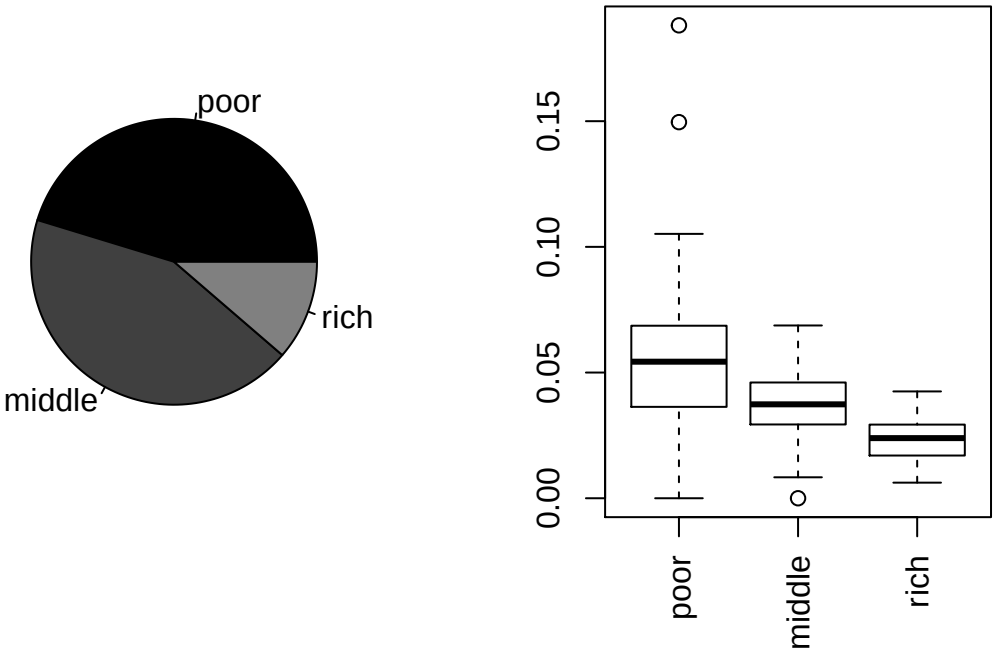
- Observer les modalités de la variables catégorielle `econ`. Recoder cette variable en classant ces modalités dans un ordre facilement interprétable.

```
gavote$econ <-factor(gavote$econ, levels=c("poor","middle","rich"))
```

- Représenter la distribution de la variable `econ`.

Pour étudier une variable catégorielle en particulier, les commandes `pie` et `barplot` fournissent un résumé interprétable.

```
par(mfrow=c(1,2))
pie(table(gavote$econ),col=gray(0:4/4))
boxplot(undercount~econ,gavote,las=3)
```



Les boxplot illustrent clairement une tendance: plus le milieu est aisé, moins on trouve de bulletins nuls.

- Montrer que le modèle ci-dessus peut s'écrire comme un modèle linéaire de la forme

$$Y = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}.$$

Il suffit d'écrire $\mathbf{X}$ sous forme de la matrice $n \times (\text{n.group} + 1)$ suivante:

$$\mathbf{X} = \begin{pmatrix} 1 & 1 & 0 & \dots & 0 \\ 1 & \vdots & \vdots & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 0 & 1 & \dots & 0 \\ 1 & 0 & \dots & \dots & 0 \\ 1 & 0 & 1 & \dots & 0 \\ 1 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & \dots & 1 \end{pmatrix}$$

La colonne 1 est associé à l'intercept μ ; la colonne 2 à l'indicatrice des individus du premier groupe et au paramètre μ_1; la colonne 3 à l'indicatrice des individus du deuxième groupe et au paramètre μ_2; etc. Le vecteur $\boldsymbol{\beta} = (\mu, \mu_1, \dots, \mu_K)$.

- Étudier l'effet du facteur `econ`.

```
M.eco <- lm(log(1+undercount)~econ,gavote)
summary(M.eco)
```

```
##
## Call:
## lm(formula = log(1 + undercount) ~ econ, data = gavote)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.052992 -0.011944  0.000129  0.008945  0.119380
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.052992   0.002489  21.291  < 2e-16 ***
## econmiddle  -0.016048   0.003558  -4.510 1.27e-05 ***
## econrich    -0.030427   0.005566  -5.467 1.78e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02112 on 156 degrees of freedom
## Multiple R-squared:  0.1983, Adjusted R-squared:  0.188
## F-statistic: 19.29 on 2 and 156 DF,  p-value: 3.267e-08
```

Une analyse rapide conclut à un effet significatif de la variable `econ` sur la proportion de bulletins nuls: plus on appartient à un milieu aisé, moins il y a de bulletins nuls. On peut même quantifier cet effet à l'aide des valeurs des paramètres.

```
## %age de baisse dans la valeur moyenne de undercount par rapport au nniveau de référence `poor`
100/coef(M.eco)[1] * coef(M.eco)[2]
```

```
## (Intercept)
##   -30.28385
```

```
100/coef(M.eco)[1] * coef(M.eco)[3]
```

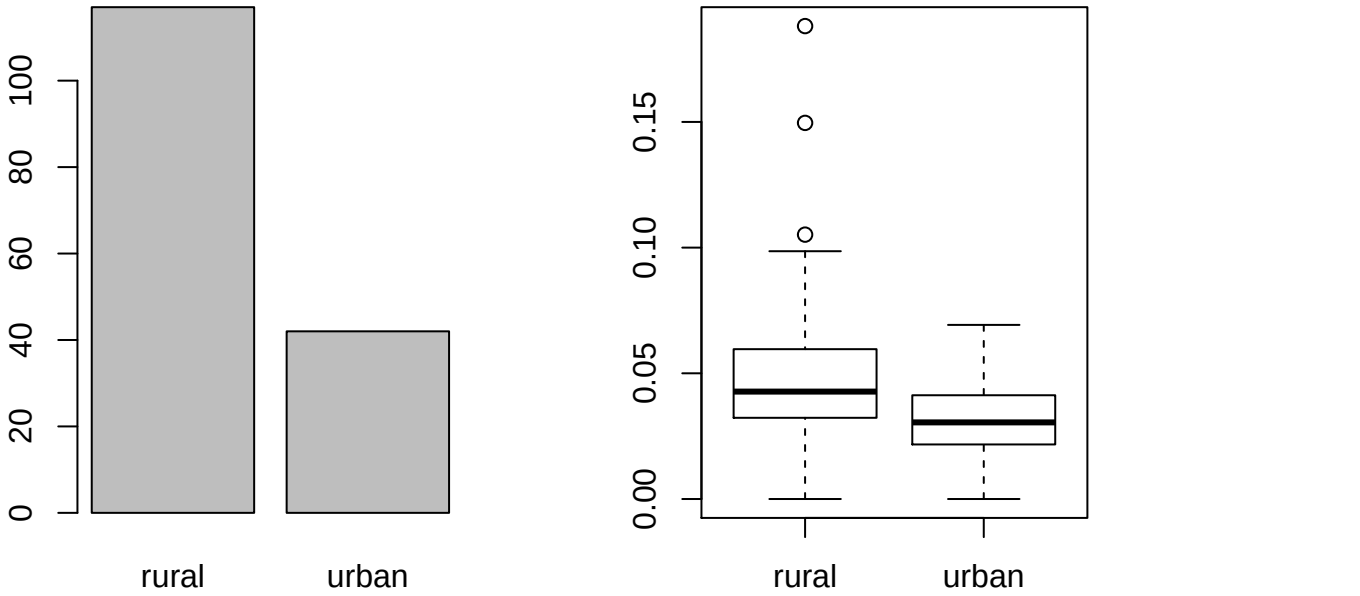
```
## (Intercept)
##   -57.41824
```

Votes selon...

Illustrer un comportement pertinent de votre choix dans le jeu de donnée en ajustant un modèle linéaire simple différent de ceux considérés jusqu'à présent.

Étudions par exemple le lien entre le vote rural/urbain et la variable à expliquer. Visuellement, il semble y avoir un effet entre sur la variable `undercount`.

```
par(mfrow=c(1,2))
barplot(sort(table(gavote$rural),decreasing=TRUE))
boxplot(undercount~rural,gavote)
```



C'est confirmé par les tests: il y a moins de bulletins nuls pour chez les votants urbains.

```
M3 <- lm(undercount~rural,gavote)
anova(M3)
```

```
## Analysis of Variance Table
##
## Response: undercount
##            Df   Sum Sq   Mean Sq F value    Pr(>F)
```

```
## rural       1 0.007706 0.0077064  13.329 0.0003554 ***
## Residuals 157 0.090771 0.0005782
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4. Construction de modèles linéaires multiples

Construction d'un modèle sur base de tests

- Ajuster un modèle de régression linéaire qui explique `undercount` en fonction des variables quantitatives `perbush`, `pergore`, `perother` et `perAA`. Comment expliquer vous le comportement obtenu ? Proposer un modèle alternatif en supprimant une variable (justifier). Cette variable sera dorénavant ôter du tableau de données `gavote`.

```
summary(lm(undercount~perother+perbush+pergore+perAA, gavote))
```

```
##
## Call:
## lm(formula = undercount ~ perother + perbush + pergore + perAA,
##     data = gavote)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.050222 -0.014154 -0.002319  0.011856  0.137301
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.08515    0.03586   2.375 0.018783 *
## perother    -0.82761    0.24389  -3.393 0.000876 ***
## perbush     -0.04385    0.04642  -0.945 0.346305
## pergore           NA         NA      NA       NA
## perAA       -0.01335    0.03218  -0.415 0.678791
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02366 on 155 degrees of freedom
## Multiple R-squared:  0.1191, Adjusted R-squared:  0.102
## F-statistic: 6.985 on 3 and 155 DF,  p-value: 0.0001942
```

La matrice de design est singulière du fait des fortes corrélations entre ces variables: la matrice de $\mathbf{X}$ n'est pas de plein rang donc $\mathbf{X}^\intercal\mathbf{X}$ n'est pas inversible.

```
XtX <- cov(gavote[, colnames(gavote) %in% c("perbush","pergore", "perAA", "perother")])
rcond(XtX)
```

```
## [1] 2.308347e-18
```

On élimine une d'entre elle, par exemple `perbush`, qui est corrélée à hauteur de 0.99 à la variable `pergore`.

```
cov2cor(XtX)
```

```
##               perAA    pergore    perbush   perother
## perAA     1.0000000  0.9216525 -0.9056253 -0.4806871
## pergore   0.9216525  1.0000000 -0.9963448 -0.3723278
## perbush  -0.9056253 -0.9963448  1.0000000  0.2916860
## perother -0.4806871 -0.3723278  0.2916860  1.0000000
```

```
gavote <- gavote[, colnames(gavote) != "perbush"]
```

- Ajuster un modèle qui explique `undercount` en fonction de toutes les variables explicatives (restantes). Construire un modèle `model.test` qui intègre uniquement les variables que vous jugerez significatives.

Le modèle après log transformation est très légèrement meilleur, mais c'est quasiment anecdotique... on peut conserver la forme initiale de la réponse.

```
summary(lm(undercount~.,gavote))$adj.r.squared
```

```
## [1] 0.2392041
```

```
summary(lm(log(1+undercount)~.,gavote))$adj.r.squared
```

```
## [1] 0.2415461
```

Seuls l'intercept et les variables `equip` et `econ` sont déclarées significatives par les t-test, on construit donc un modèle en conséquence:

```
model.test <- lm(undercount~equip+econ,gavote)
summary(model.test)
```

```
##
## Call:
## lm(formula = undercount ~ equip + econ, data = gavote)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.062688 -0.012753 -0.002154  0.010154  0.117329
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.050732   0.002911  17.429  < 2e-16 ***
## equipOS-CC   0.008791   0.004315   2.037  0.04336 *
## equipOS-PC   0.020060   0.005460   3.674  0.00033 ***
## equipPAPER  -0.010485   0.015616  -0.671  0.50294
## equipPUNCH   0.012872   0.005952   2.162  0.03215 *
## econmiddle  -0.020629   0.003833  -5.382 2.73e-07 ***
## econrich    -0.039223   0.006013  -6.523 9.68e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0217 on 152 degrees of freedom
## Multiple R-squared:  0.2734, Adjusted R-squared:  0.2447
## F-statistic: 9.531 on 6 and 152 DF,  p-value: 6.926e-09
```

Cependant, en conservant aussi `rural` et/ou `perother`, on obtient un R^2 légèrement meilleur. Mais l'analyse de la variance ne déclare pas ces modèles très significativement meilleurs. Ils restent néanmoins défendables si notre objectif est la prédiction. On se contente ici du modèle ne gardant que les variables significatives.

```
model.test2 <- lm(undercount~equip+econ+rural,gavote)
summary(model.test2)$adj.r.squared
```

```
## [1] 0.2516078
```

```
anova(model.test, model.test2)
```

```
## Analysis of Variance Table
##
## Model 1: undercount ~ equip + econ
## Model 2: undercount ~ equip + econ + rural
##   Res.Df      RSS Df Sum of Sq     F Pr(>F)
## 1    152 0.071556
## 2    151 0.070434  1 0.0011213 2.404 0.1231
```

```
model.test3 <- lm(undercount~equip+econ+perother,gavote)
summary(model.test3)$adj.r.squared
```

```
## [1] 0.2474172
```

```
anova(model.test, model.test3)
```

```
## Analysis of Variance Table
##
## Model 1: undercount ~ equip + econ
## Model 2: undercount ~ equip + econ + perother
##   Res.Df      RSS Df  Sum of Sq      F Pr(>F)
## 1    152 0.071556
## 2    151 0.070829  1 0.00072694 1.5498 0.2151
```

```
model.test4 <- lm(undercount~equip+econ+rural+perother,gavote)
summary(model.test4)$adj.r.squared
```

```
## [1] 0.2518474
```

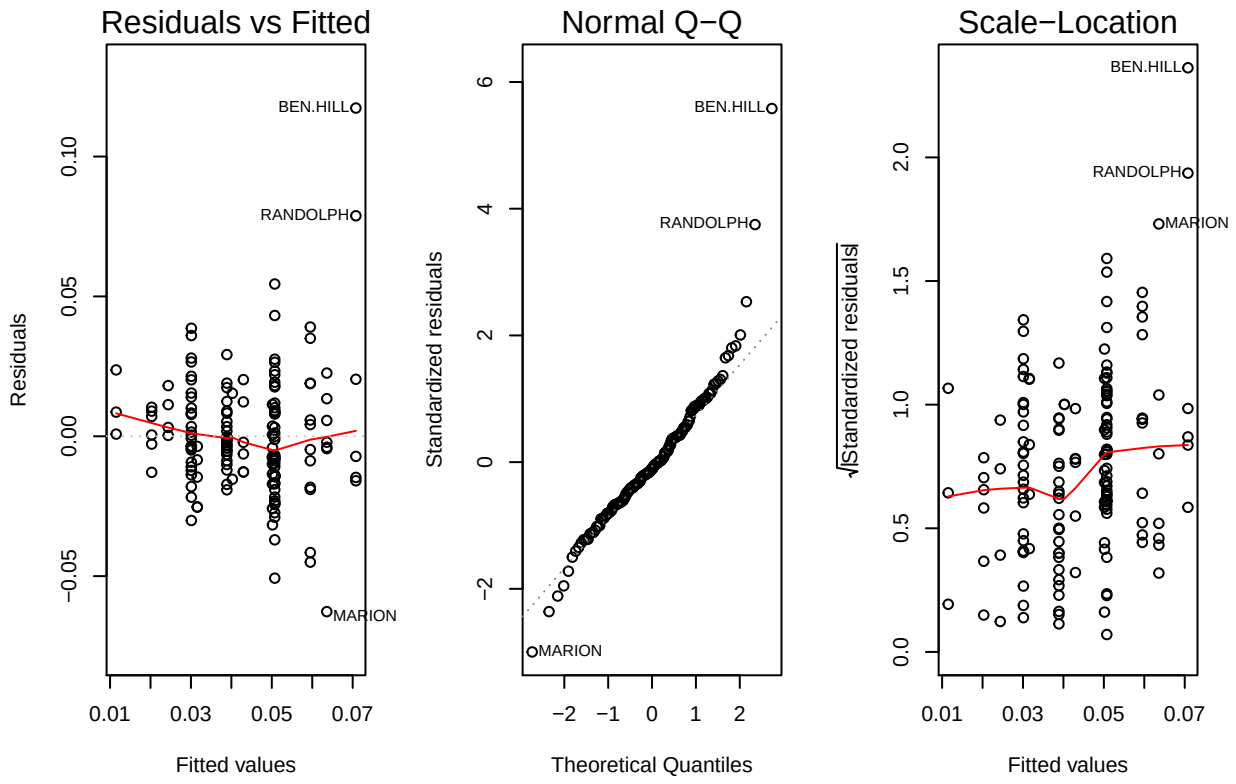
```
anova(model.test, model.test4)
```

```
## Analysis of Variance Table
##
## Model 1: undercount ~ equip + econ
## Model 2: undercount ~ equip + econ + rural + perother
##   Res.Df      RSS Df Sum of Sq      F Pr(>F)
## 1    152 0.071556
## 2    150 0.069946  2 0.0016102 1.7265 0.1814
```

- Étudier les résidus de `model.test`.

Trois viennent un peu perturber la normalité et l'homoscédasticité.

```
par(mfrow=c(1,3))
plot(model.test, which=1:3)
```



Construction par régression stepwise

- En utilisant la procédure `step`, proposer un modèle à l'aide de la procédure forward/backward utilisant les critères AIC puis BIC. On appelera `model.AIC` et `model.BIC` les modèles respectivement obtenus.

```
model.AIC <- step(lm(undercount~.,gavote), k=2)
```

```
## Start:  AIC=-1205.45
## undercount ~ equip + econ + perAA + rural + atlanta + pergore +
##     perother
##
##            Df Sum of Sq      RSS     AIC
## - atlanta   1 0.0000034 0.069708 -1207.4
## - pergore   1 0.0000868 0.069792 -1207.2
## - perAA     1 0.0001935 0.069899 -1207.0
## - perother  1 0.0006455 0.070351 -1206.0
## - rural     1 0.0007085 0.070414 -1205.8
## <none>                  0.069705 -1205.5
## - equip     4 0.0084239 0.078129 -1195.3
## - econ      2 0.0078679 0.077573 -1192.5
##
## Step:  AIC=-1207.44
```

```
## undercount ~ equip + econ + perAA + rural + pergore + perother
##
##            Df Sum of Sq      RSS     AIC
## - pergore   1 0.0000878 0.069796 -1209.2
## - perAA     1 0.0001910 0.069899 -1209.0
## - perother  1 0.0006611 0.070369 -1207.9
## - rural     1 0.0007126 0.070421 -1207.8
## <none>                  0.069708 -1207.4
## - equip     4 0.0086883 0.078397 -1196.8
## - econ      2 0.0082120 0.077920 -1193.7
##
## Step:  AIC=-1209.24
## undercount ~ equip + econ + perAA + rural + perother
##
##            Df Sum of Sq      RSS     AIC
## - perAA     1 0.0001493 0.069946 -1210.9
## - perother  1 0.0005815 0.070378 -1209.9
## - rural     1 0.0007368 0.070533 -1209.6
## <none>                  0.069796 -1209.2
## - equip     4 0.0087722 0.078568 -1198.4
## - econ      2 0.0083559 0.078152 -1195.3
##
## Step:  AIC=-1210.9
## undercount ~ equip + econ + rural + perother
##
##            Df Sum of Sq      RSS     AIC
## - perother  1 0.0004889 0.070434 -1211.8
## - rural     1 0.0008833 0.070829 -1210.9
## <none>                  0.069946 -1210.9
## - equip     4 0.0086956 0.078641 -1200.3
## - econ      2 0.0090961 0.079042 -1195.5
##
## Step:  AIC=-1211.79
## undercount ~ equip + econ + rural
##
##         Df Sum of Sq      RSS     AIC
## <none>               0.070434 -1211.8
## - rural  1 0.0011213 0.071556 -1211.3
## - equip  4 0.0084709 0.078905 -1201.7
## - econ   2 0.0150415 0.085476 -1185.0
```

```
model.BIC <- step(lm(undercount~.,gavote), k=log(nrow(gavote)))
```

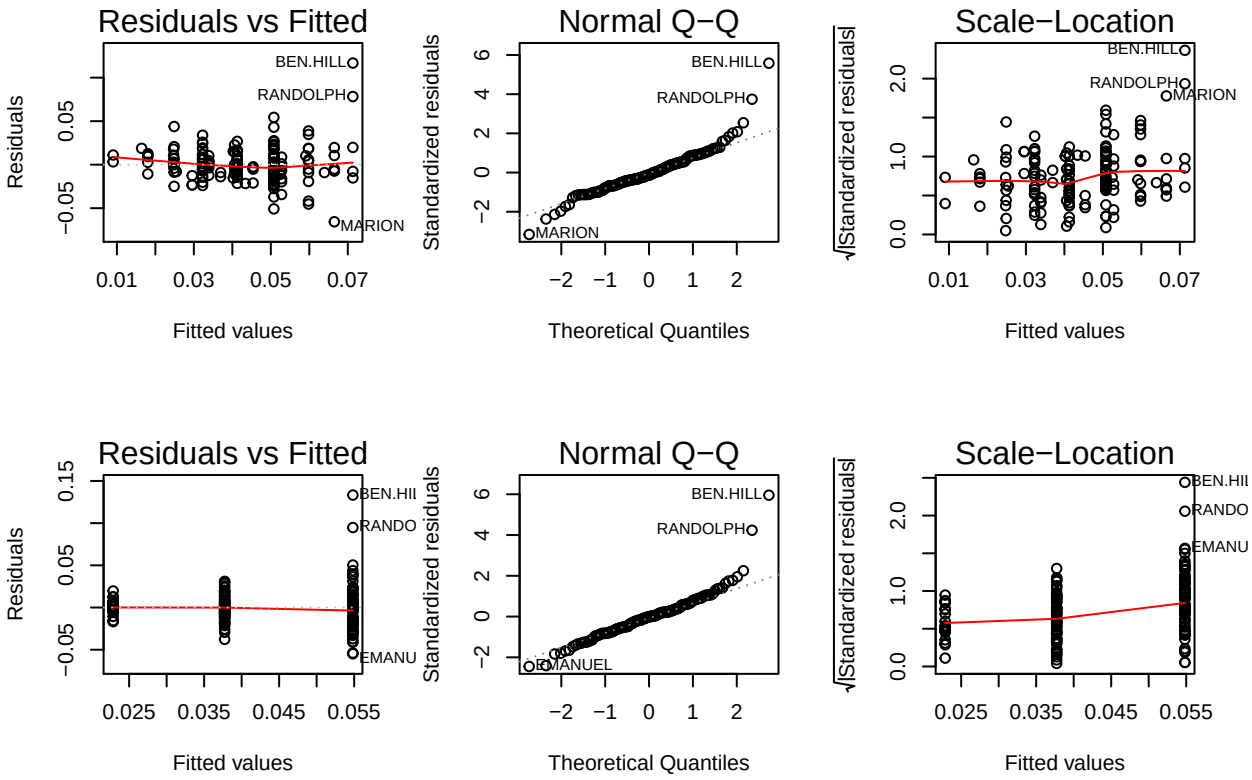
```
## Start:  AIC=-1168.62
## undercount ~ equip + econ + perAA + rural + atlanta + pergore +
##     perother
##
##            Df Sum of Sq      RSS     AIC
## - atlanta   1 0.0000034 0.069708 -1173.7
## - pergore   1 0.0000868 0.069792 -1173.5
## - perAA     1 0.0001935 0.069899 -1173.2
## - perother  1 0.0006455 0.070351 -1172.2
## - rural     1 0.0007085 0.070414 -1172.1
## - equip     4 0.0084239 0.078129 -1170.8
```

```
## <none>                  0.069705 -1168.6
## - econ      2 0.0078679 0.077573 -1161.8
##
## Step:  AIC=-1173.68
## undercount ~ equip + econ + perAA + rural + pergore + perother
##
##            Df Sum of Sq      RSS     AIC
## - pergore   1 0.0000878 0.069796 -1178.5
## - perAA     1 0.0001910 0.069899 -1178.3
## - perother  1 0.0006611 0.070369 -1177.2
## - rural     1 0.0007126 0.070421 -1177.1
## - equip     4 0.0086883 0.078397 -1175.3
## <none>                  0.069708 -1173.7
## - econ      2 0.0082120 0.077920 -1166.1
##
## Step:  AIC=-1178.55
## undercount ~ equip + econ + perAA + rural + perother
##
##            Df Sum of Sq      RSS     AIC
## - perAA     1 0.0001493 0.069946 -1183.3
## - perother  1 0.0005815 0.070378 -1182.3
## - rural     1 0.0007368 0.070533 -1182.0
## - equip     4 0.0087722 0.078568 -1180.0
## <none>                  0.069796 -1178.5
## - econ      2 0.0083559 0.078152 -1170.7
##
## Step:  AIC=-1183.28
## undercount ~ equip + econ + rural + perother
##
##            Df Sum of Sq      RSS     AIC
## - perother  1 0.0004889 0.070434 -1187.2
## - rural     1 0.0008833 0.070829 -1186.4
## - equip     4 0.0086956 0.078641 -1184.9
## <none>                  0.069946 -1183.3
## - econ      2 0.0090961 0.079042 -1174.0
##
## Step:  AIC=-1187.24
## undercount ~ equip + econ + rural
##
##         Df Sum of Sq      RSS     AIC
## - rural  1 0.0011213 0.071556 -1189.8
## - equip  4 0.0084709 0.078905 -1189.5
## <none>               0.070434 -1187.2
## - econ   2 0.0150415 0.085476 -1166.6
##
## Step:  AIC=-1189.8
## undercount ~ equip + econ
##
##         Df Sum of Sq      RSS     AIC
## - equip  4 0.0077405 0.079296 -1193.7
## <none>               0.071556 -1189.8
## - econ   2 0.0249718 0.096528 -1152.3
##
## Step:  AIC=-1193.74
```

```
## undercount ~ econ
##
##        Df Sum of Sq      RSS     AIC
## <none>              0.079296 -1193.7
## - econ  2  0.019181 0.098477 -1169.4
```

- Étudier les résidus de ces modèles.

```
par(mfrow=c(2,3))
plot(model.AIC, which=1:3)
plot(model.BIC, which=1:3)
```



Recherche exhaustive

Lorsque le nombre de prédicteurs reste raisonable (< 30), il est possible de tester tous les modèles et de choisir le meilleur au sens d'un critère.

- En utilisant la procédure `regsubsets` du package `leaps`, déterminer le meilleur modèle au sens du R^2 ajusté. Commentez.

```
model.all <- regsubsets(undercount~.,gavote)
summary(model.all)$outmat[which.max(summary(model.all)$adjr2), ]
```

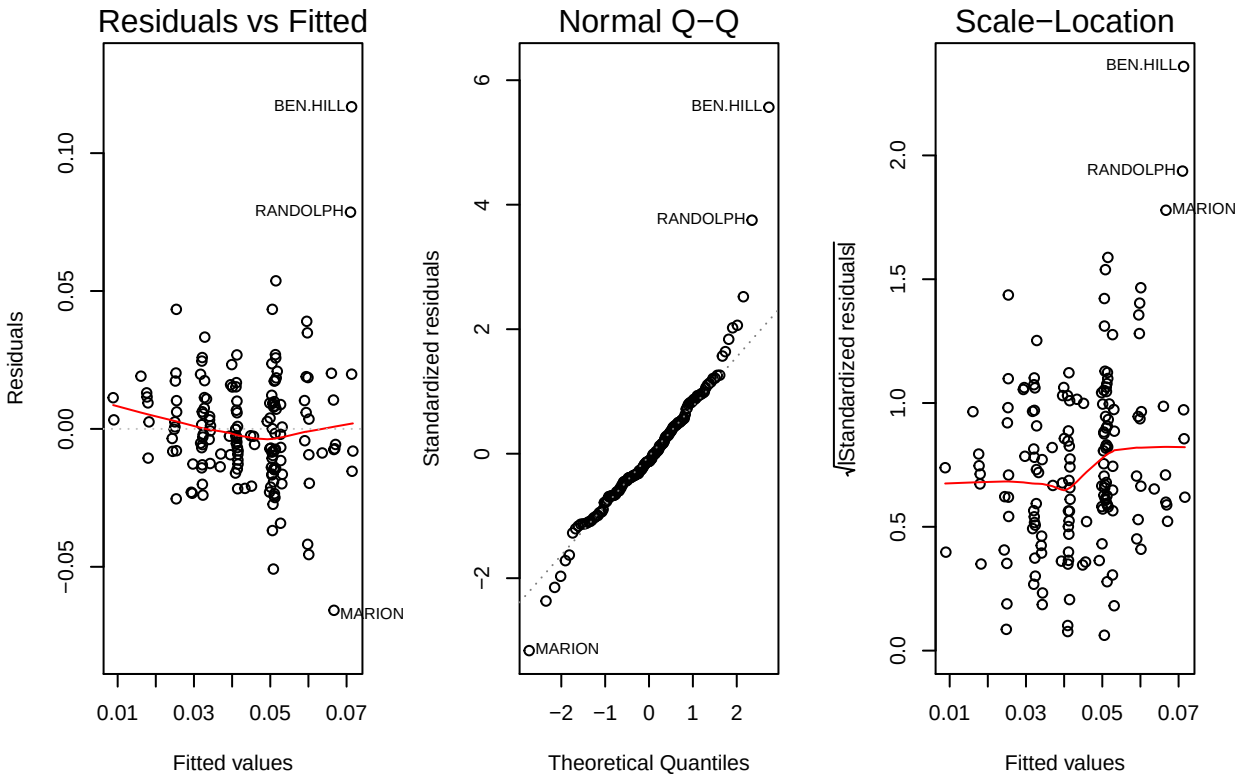
```
##        equipOS-CC        equipOS-PC        equipPAPER        equipPUNCH
##               "*"               "*"               " "               "*"
##        econmiddle          econrich             perAA        ruralurban
```

```
##               "*"              "*"              " "              "*"
## atlantanotAtlanta          pergore         perother
##               " "              " "              "*"
```

```
model.r2 <- lm(undercount~equip+econ+rural+pergore, gavote)
```

- Étudier les résidus de `model.r2`.

```
par(mfrow=c(1,3))
plot(model.r2, which=1:3)
```



5. Étude du modèle final

- Choisissez un modèle entre `model.test`, `model.AIC`, `model.BIC` et `model.r2`. Faites un diagnostic complet (résidus, test des hypothèses associées au modèle linéaire, distance de Cook, leviers). On pourra utiliser les fonctions `plot.lm`, `cook.distance`, `rstudent`, ...

Les 3 modèles sont emboîtés ! Ils tournent autour des même variables que nous avions déjà identifiés.

```
anova(model.BIC,model.AIC,model.r2)
```

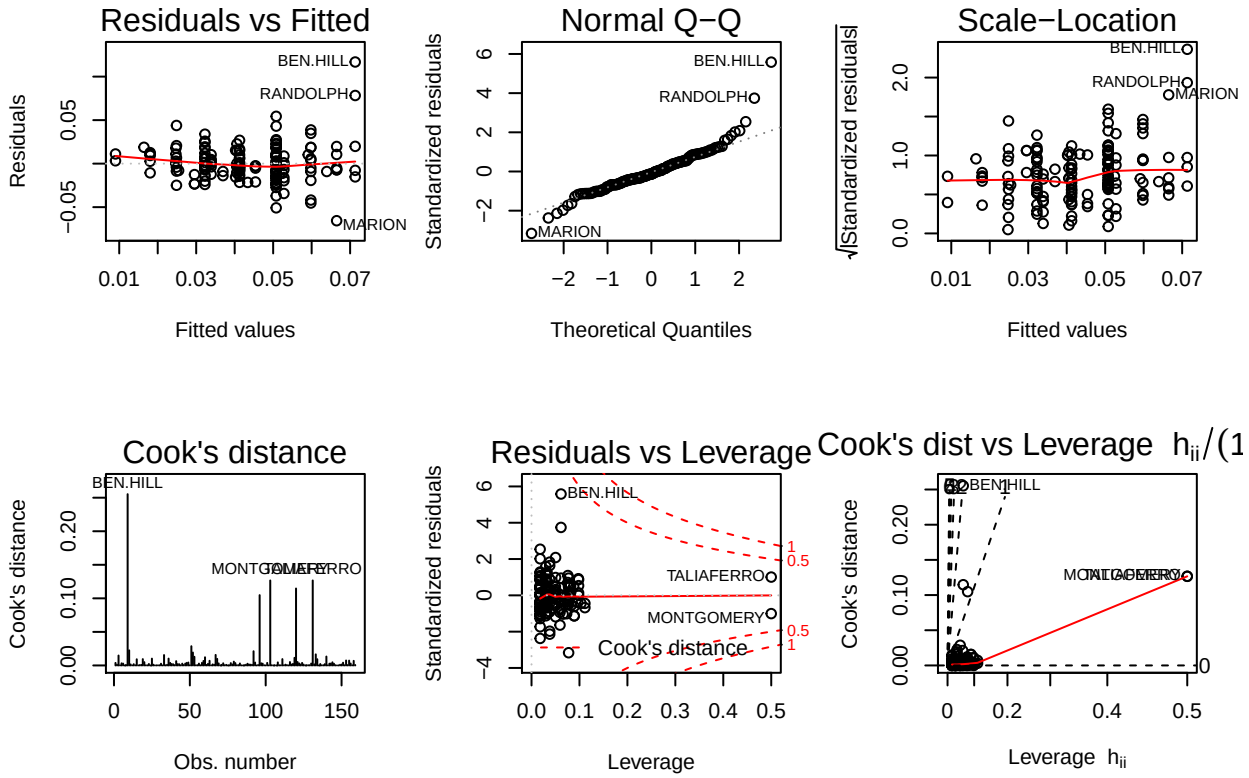
```
## Analysis of Variance Table
##
## Model 1: undercount ~ econ
## Model 2: undercount ~ equip + econ + rural
## Model 3: undercount ~ equip + econ + rural + pergore
```

```
##   Res.Df      RSS Df Sum of Sq      F   Pr(>F)
## 1    156 0.079296
## 2    151 0.070434  5 0.0088618 3.7760 0.002999 **
## 3    150 0.070405  1 0.0000289 0.0616 0.804291
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Essayez d'améliorer l'ajustement en transformant la réponse, certains prédicteurs, en excluant les points aberrants, etc. On pourra aussi fusionner les niveaux de certains facteurs dont les effets sont similaires.

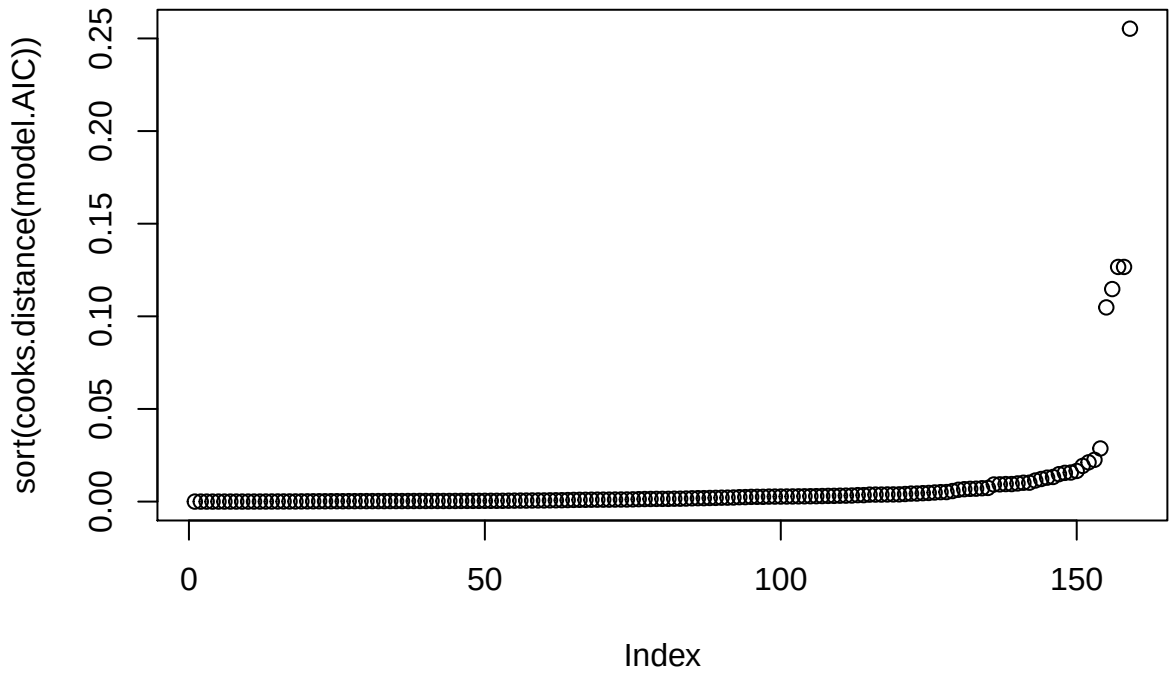
Je choisis par exemple le modèle de l'AIC, un bon compromis.

```
par(mfrow=c(2,3))
plot(model.AIC, which=1:6)
```



Visiblement, un certains nombres de points se démarquent du reste de l'échantillon. Ils correspondent aux 5 plus grande distance de Cook.

```
plot(sort(cooks.distance(model.AIC)))
```




```
remove <- order(cooks.distance(model.AIC), decreasing = TRUE)[1:5]
```

Si on ôte ces points, le R^2 est sensiblement meilleur et la variable rural devient significative à 5%.

```
gavote2 <- gavote[-remove, ]
summary(lm(undercount~equip+econ+rural,gavote2))
```

```
##
## Call:
## lm(formula = undercount ~ equip + econ + rural, data = gavote2)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.049471 -0.009938 -0.001223  0.009715  0.055697
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.049471   0.002315  21.369  < 2e-16 ***
## equipOS-CC   0.007834   0.003423   2.289 0.023513 *
## equipOS-PC   0.009288   0.004546   2.043 0.042828 *
## equipPUNCH   0.019072   0.005092   3.746 0.000258 ***
## econmiddle  -0.015076   0.003260  -4.625 8.15e-06 ***
## econrich    -0.029872   0.005410  -5.522 1.48e-07 ***
## ruralurban  -0.007916   0.003807  -2.079 0.039319 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01716 on 147 degrees of freedom
## Multiple R-squared:  0.3218, Adjusted R-squared:  0.2941
## F-statistic: 11.63 on 6 and 147 DF,  p-value: 1.24e-10
```

En fait, on a enlevé une modalité de la variable `equip` !

```r
table(gavote2$equip)
```

```
##
## LEVER OS-CC OS-PC PAPER PUNCH
##    74    44    20     0    16
```

Tentons de recoder cette variable en fusionnant en plus les modalité OS-CC et OS-PC qui ont des coefficients de régression sensiblement identiques.

```r
equip.new <- as.character(gavote2$equip)
equip.new[equip.new %in% c("OS-CC","OS-PC")] <- "OS"
gavote2$equip.new <- as.factor(equip.new )
```

On améliore un tout petit peu le R^2 et on rend le modèle plus interprétable (moins de modalité)

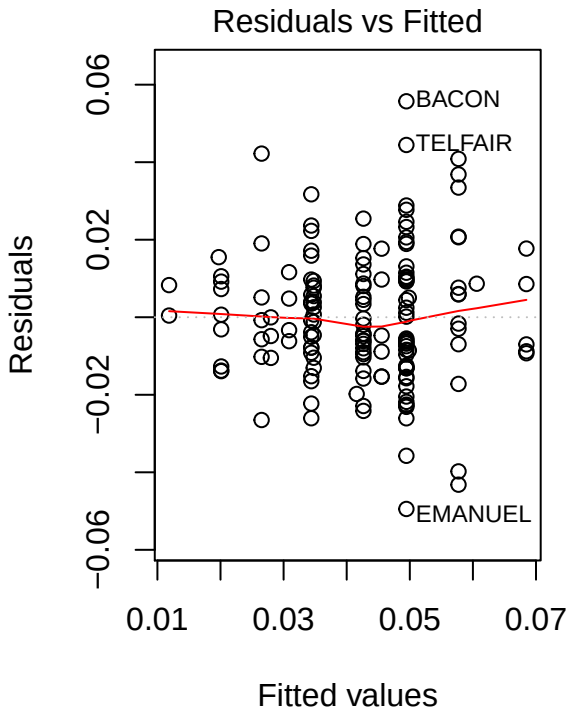
```r
model.final <- lm(undercount~equip.new+econ+rural,gavote2)
summary(model.final)
```

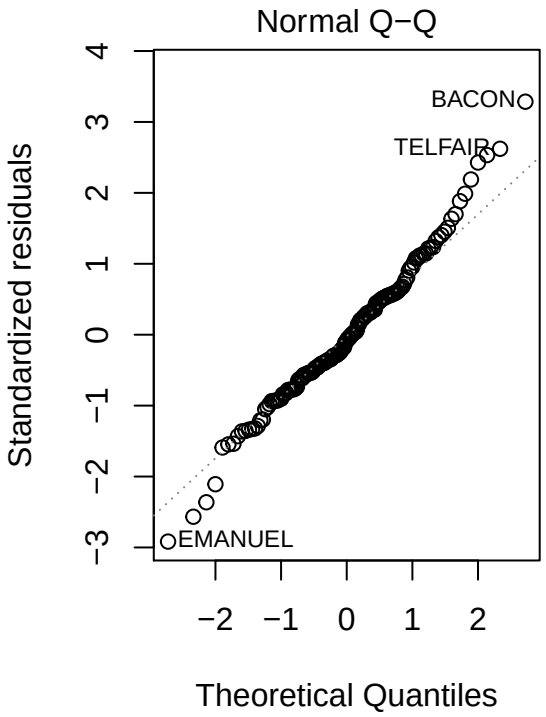
```
##
## Call:
## lm(formula = undercount ~ equip.new + econ + rural, data = gavote2)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.049458 -0.010067 -0.001435  0.009371  0.055711
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     0.049458   0.002308  21.432  < 2e-16 ***
## equip.newOS     0.008261   0.003127   2.642 0.009133 **
## equip.newPUNCH  0.019021   0.005074   3.749 0.000254 ***
## econmiddle     -0.015071   0.003250  -4.637 7.69e-06 ***
## econrich       -0.029737   0.005376  -5.532 1.40e-07 ***
## ruralurban     -0.007874   0.003793  -2.076 0.039617 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01711 on 148 degrees of freedom
## Multiple R-squared:  0.3214, Adjusted R-squared:  0.2984
## F-statistic: 14.02 on 5 and 148 DF,  p-value: 3.246e-11
```

- Interpréter le modèle final que vous retenez.

Le graphe des résidus est à peu près satisfaisant, et le la distribution symétrique.

```r
par(mfrow=c(1,2))
plot(model.final, which=1:2)
```





Ceux-ci ne semblent pas trop corrélés

```
durbinWatsonTest(model.final)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1      0.07511671      1.829459   0.276
##  Alternative hypothesis: rho != 0
```

La distribution n'est pas tout à fait gaussienne, mais ça passe... (on standardise avant si on veut)

```
shapiro.test(rstandard(model.final))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(model.final)
## W = 0.98517, p-value = 0.09868
```