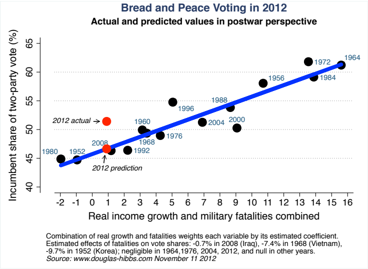
Introduction à la modélisation

1. Pourquoi modéliser ?
2. Les moindres carrés ordinaires
3. La fonction `lm` sous R



Pourquoi modéliser ?

- Pour analyser
- Pour comprendre

Modéliser pour analyser

- Un modèle réduit de la réalité
- Isoler le rôle de chaque variable
- Raisonner « toutes choses égales par ailleurs »

Modéliser pour analyser

Modéliser, c'est mettre en relation une variable expliquée (dépendante / prédite) et une ou plusieurs variables explicatives (indépendantes / prédicteurs).

$$Y = f(X_1, X_2, X_3, \ldots, X_n)$$

L'estimation du modèle consiste à estimer la valeur des paramètres (ou coefficients).

$$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots + \beta_n X_n + \epsilon$$

Modéliser pour analyser

Exemple : on s'intéresse au vote FN à Marseille, par bureau de vote, lors des élections municipales de 2014, en fonction de la sociologie des bureaux de vote.

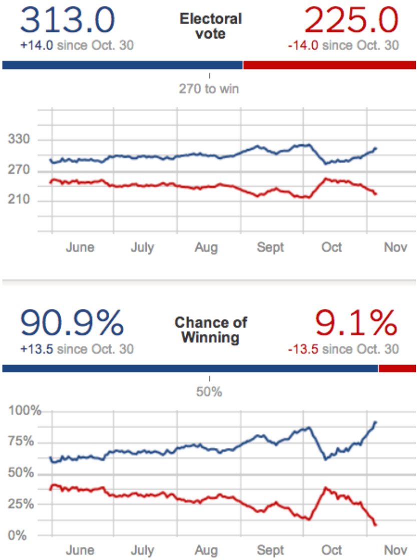
$$
\begin{aligned}
\text{Vote FN} = f(&\text{Composition socioprofessionnelle},\\
&\text{population étrangère},\\
&\text{taux de chômage},\\
&\text{locataires HLM})
\end{aligned}
$$

Modéliser pour analyser

Hypothèses :

- Classes populaires : positivement associées au vote FN
- Population étrangère : positivement associée au vote FN
- Taux de chômage : positivement associé au vote FN
- Locataires HLM : positivement associé au vote FN

Modéliser pour prédire



Les moindres carrés ordinaires (MCO)

Le terme d'erreur

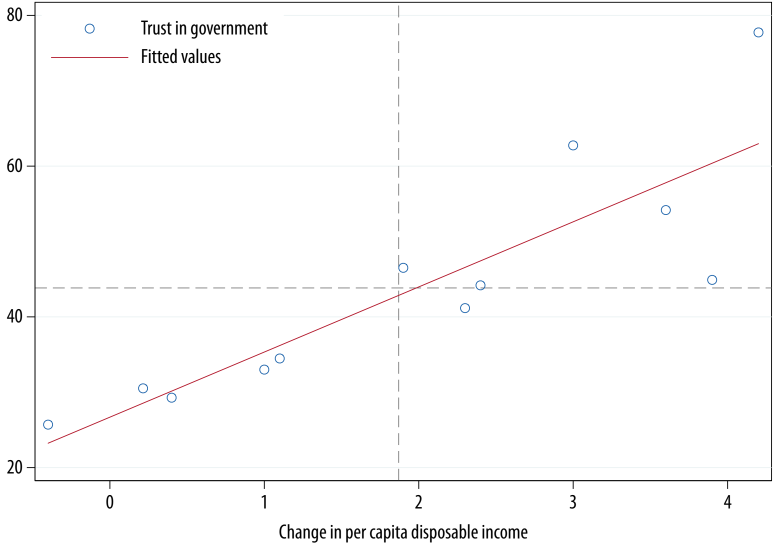
Dans un modèle linéaire simple $Y = \alpha + \beta X + \epsilon$, le coefficient de régression β est calculé de telle sorte que la somme des carrés des écarts soit minimisée.

$$RSS = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{n} \epsilon^2$$

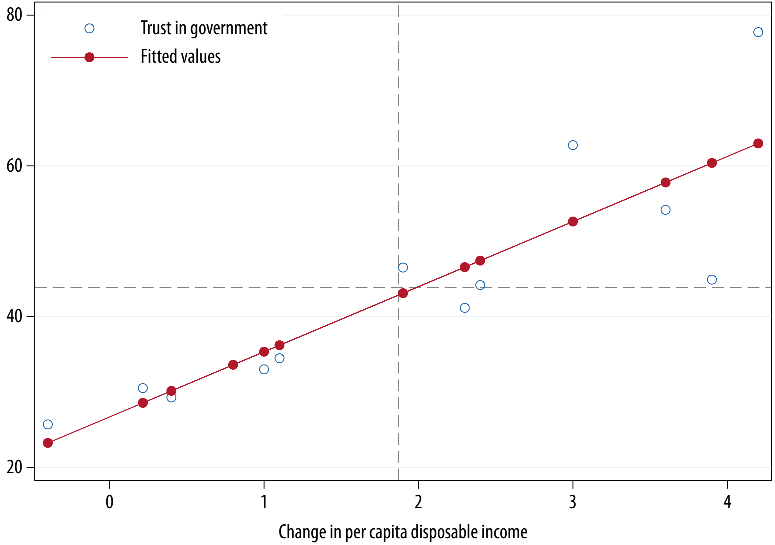
où $Y_i - \hat{Y}_i$ est le résidu (ou terme d'erreur) de chaque observation.

Estimation des paramètres

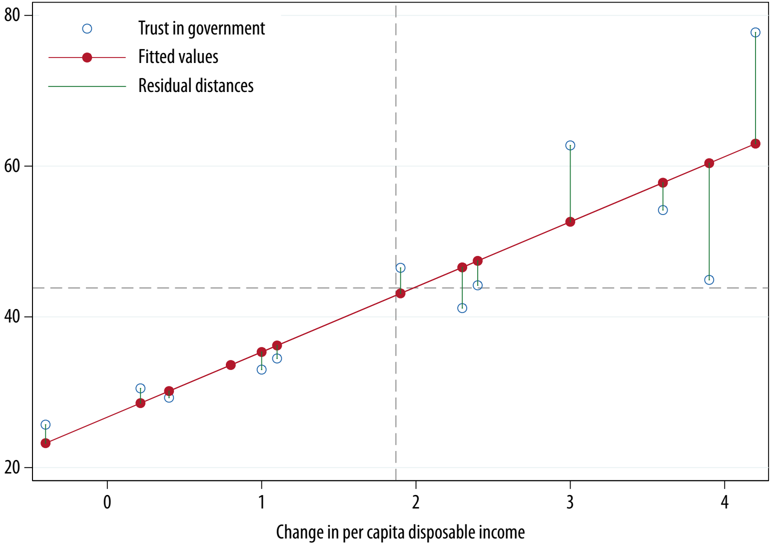
$$\beta = \frac{\mathrm{Cov}(X, Y)}{\mathrm{Var}_X} = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n} (X_i - \bar{X})^2} \quad \alpha = \bar{Y} - \beta \bar{X}$$



Trust in government
Fitted values
80
60
40
20
0
1
2
3
4
Change in per capita disposable income



Trust in government
Fitted values
80
60
40
20
0
1
2
3
4
Change in per capita disposable income



80
60
40
20
0
1
2
3
4
Trust in government
Fitted values
Residual distances
Change in per capita disposable income

Les moindres carrés ordinaires (MCO)

Attention

- Les modèles de régression linéaire supposent que les relations sont linéaires et additives.
- Les résidus sont supposés être normalement distribués.
- Les coefficients ne sont pas standardisés (on ne peut les comparer entre eux).
- Les coefficients s'interprètent relativement à l'unité de la variable dépendante.

Les moindres carrés ordinaires (MCO)

Attention

- Les coefficients estiment l'effet d'une variable indépendante sur la variable dépendante toutes choses égales par ailleurs, c'est-à-dire en neutralisant l'effet des autres variables.
- La qualité globale du modèle peut être quantifié au travers du R^2, qui représente la part de variance (de la variable dépendante) expliquée.
- Pour les variables indépendantes catégoriques, on estime un coefficient par modalité, à l'exception de la première (baseline).

La fonction `lm` sous R

La fonction `lm` permet d'estimer des linear models.
Elle nécessite simplement le modèle, sous forme d'une formule, et un `dataframe`.

```
modele1 <- lm(y ~ x1 + x2, data = data)
```

`lm` permet également d'estimer des modèles pondérés (argument `weights`) ou portant sur un sous-ensemble du jeu de données (argument `subset`).

Les objets de classe `lm`

Il faut stocker le résultat d’une régression dans un objet.
On peut ensuite appliquer des méthodes à cet objet (exemple : `summary`), ou accéder à ses composantes.

```
names(modele1)

##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "na.action"     "xlevels"       "call"          "terms"
## [13] "model"
```

Exemple d’utilisation de la commande `lm`

Reprenons l’exemple du vote FN à Marseille. Ouvrir le script `regMarseille.R`.

```r
modele1 <- lm(Ravier ~ CS2 + CS3 + CS4 + CS5 + CS6 + etrangers + chomage + HLM,
    data = marseille)

summary(modele1)
```

```
##
## Call:
## lm(formula = Ravier ~ CS2 + CS3 + CS4 + CS5 + CS6 + etrangers +
##     chomage + HLM, data = marseille)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.791 -2.142 -0.269  1.956 11.382
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  16.9749     2.2179    7.65  1.1e-13 ***
## CS2           0.0772     0.1497    0.52  0.60641
## CS3          -0.3401     0.0568   -5.99  4.2e-09 ***
## CS4           0.1086     0.0563    1.93  0.05461 .
## CS5           0.0251     0.0498    0.50  0.61466
## CS6           0.3253     0.0844    3.85  0.00013 ***
## etrangers    -0.2934     0.0556   -5.28  2.0e-07 ***
## chomage      -0.4647     0.0710   -6.54  1.6e-10 ***
## HLM          -0.0311     0.0111   -2.81  0.00510 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.32 on 468 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.519, Adjusted R-squared:  0.51
## F-statistic:   63 on 8 and 468 DF,  p-value: <2e-16
```

```r
library(texreg)
screenreg(modele1, single.row = TRUE)
```

```
##
## ==============================
##                  Model 1
## ------------------------------
## (Intercept)   16.97 (2.22) ***
## CS2            0.08 (0.15)
## CS3           -0.34 (0.06) ***
## CS4            0.11 (0.06)
## CS5            0.03 (0.05)
## CS6            0.33 (0.08) ***
## etrangers     -0.29 (0.06) ***
## chomage       -0.46 (0.07) ***
## HLM           -0.03 (0.01) **
## ------------------------------
## R^2            0.52
## Adj. R^2       0.51
## Num. obs.    477
## ==============================
## *** p < 0.001, ** p < 0.01, * p < 0.05
```

Les objets de classe `lm`

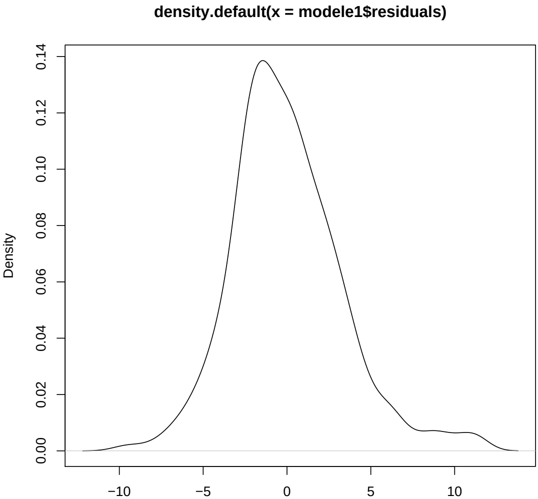
On peut ensuite analyser un élément donné du modèle, par exemple les résidus.

```
summary(modele1$residuals)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -9.790  -2.140  -0.269   0.000   1.960  11.400
```

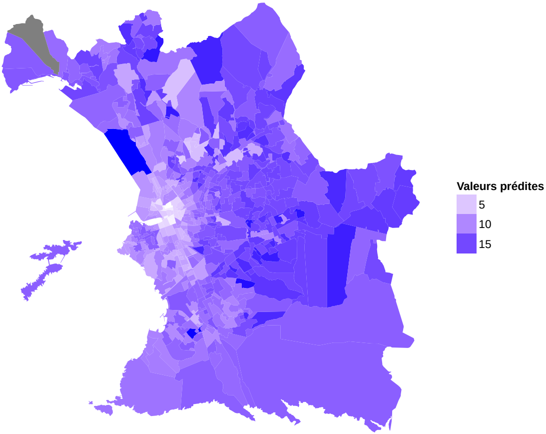
Les objets de classe `lm`

```r
plot(density(modele1$residuals))
```



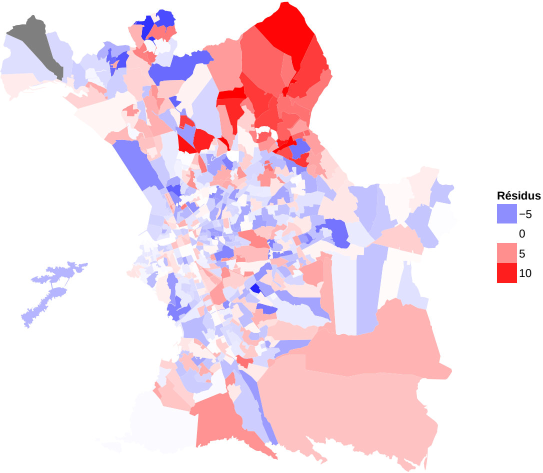
Les objets de classe `lm`

On peut cartographier les valeurs prédites (`modele1$fitted.values`) :



Les objets de classe lm

Ou les résidus (`modele1$residuals`) :



Affiner la spécification d’un modèle

Pour améliorer un modèle, on peut :

- inclure un effet d’interaction entre deux variables (avec l’opérateur `:` ou l’opérateur `*`).
- ne pas inclure de constante (avec l’opérateur `-1`).
- dans le cas d’une variable indépendante catégorique, changer la catégorie de référence (en utilisant la fonction `relevel`).
- centrer, voire centrer-réduire, les variables.
- transformer une variable dépendante (par exemple, transformation logarithmique).

```r
modele2 <- lm(Ravier ~ CS2 + CS3 + CS4 + CS5 + CS6 * etrangers + CS6 * chomage +
    HLM, data = marseille)
screenreg(list(modele1, modele2))
```

```
##
## =====================================
##                 Model 1     Model 2
## -------------------------------------
## (Intercept)     16.97 ***   16.94 ***
##                 (2.22)      (2.31)
## CS2              0.08        0.10
##                 (0.15)      (0.15)
## CS3             -0.34 ***   -0.33 ***
##                 (0.06)      (0.06)
## CS4              0.11        0.12 *
##                 (0.06)      (0.06)
## CS5              0.03        0.02
##                 (0.05)      (0.05)
## CS6              0.33 ***    0.32 **
##                 (0.08)      (0.11)
## etrangers       -0.29 ***    0.07
##                 (0.06)      (0.15)
## chomage         -0.46 ***   -0.75 ***
##                 (0.07)      (0.16)
## HLM             -0.03 **    -0.03 **
##                 (0.01)      (0.01)
## CS6:etrangers               -0.02 *
##                             (0.01)
## CS6:chomage                  0.02
##                             (0.01)
## -------------------------------------
## R^2              0.52        0.53
## Adj. R^2         0.51        0.52
## Num. obs.      477         477
## =====================================
## *** p < 0.001, ** p < 0.01, * p < 0.05
```