# Introduction to Programming

## Lecture 11-12: Econometrics with R

Clément Mazet-Sonilhac
SciencesPo

29 avril 2019

# Preliminary stuff

## Install and call packages

- install.package("package name")
- Library("package name")

# Preliminary stuff

## Install and call packages

- require(tidyverse)
- require(nycflights13)
- require(gapminder)
- require(Lahman)
- require(gridExtra)
- require(ggthemes)

# Preliminary stuff

## Formulas

- Formulas : used to specify models ; encoded with ∼
- Ex : y ∼ x stands for "y is explained by x"
- Use extensively for econometric analysis, but not only.

- **Exercice 1** : Generate a vector of numbers from 0 to 10 with increment 0.3 using `x = seq(from = 0, to = 10, by = 0.3)`
  Then, `y <- 2 + 3 * x + rnorm(length(x))`
  And finally, `plot(y ∼ x)`. What do you obtain ?

# Ordinary Least Square

## Disclaimer

Will not cover :

- panel data (very similar to OLS)
- time series (excellent tools from the ‘stats‘ and ‘tseries‘ packages)
- quantile regressions

# Ordinary Least Square

## Introduction

- Interested in estimating the $\beta$s in

$$y = X\beta + \epsilon$$

- Estimates : $\hat{\beta} = (X^T X)^{-1} X^T y$
- Fitted values : $\hat{y} = X\hat{\beta}$
- Residuals : $\hat{\epsilon} = y - \hat{y}$
- Residual sum of squares : $RSS = \hat{\epsilon}^T \hat{\epsilon}$

# Ordinary Least Square

## Linear Regression in R

Models are estimated by calling a model-fitting function

- Most of them take two key arguments : the formula and the data
- For Linear Models fitted with OLS : `lm()`
- These functions return a fitted-model object, from which can extract the point estimates, compute predicted values, etc.

# Ordinary Least Square

## Linear Regression in R : example

Data from Stock & Watson (2007) on subscriptions to economics journals at US libraries for the year 2000. Write :

- `data(Journals, package = "AER")`
- `journals <- as_tibble(Journals)`
- `journals`

Goal : estimate the effect of the price per citation on the number of library subscriptions

# Ordinary Least Square

## Linear Regression in R : example

- **Exercice 2** : compute a new variable, the price per citation, compute some summary statistics, and plot the number of subscriptions againsts the price per citation. Combine the summary statistics and the plot to describe what type of model we should use.

# Ordinary Least Square

## Linear Regression in R : example

**Exercice 2 (Solution)** :

- Generate new variable price per citation : `journals <- journals %>% mutate(citeprice = price / citations)`
- Plot subscriptions againt price per citation :
  (i) `plot(subs~(price/citations), data= Journals)` or
  (ii) `ggplot(journals) + geom_point(aes(x = subs, y=citeprice))`

# Ordinary Least Square

## Linear Regression in R : example

**Exercice 2 (Solution)** :

- Generate new variable price per citation : `journals <- journals %>% mutate(citeprice = price / citations)`
- Plot subscriptions againt price per citation :
  (i) `plot(subs~(price/citations), data= Journals)` or
  (ii) `ggplot(journals) + geom_point(aes(x = subs, y=citeprice))`
- The relationship isn't clear. Maybe you want more information about the data.

# Ordinary Least Square

## Linear Regression in R : example

```
require(gridExtra)
g = ggplot(journals)
p1 = g + geom_histogram(aes(x = citeprice), fill = "tomato3")
p2 = g + geom_histogram(aes(x = subs), fill = "tomato3")
p3 = g + geom_histogram(aes(x = log(citeprice)), fill = "tomato3")
p4 = g + geom_histogram(aes(x = log(subs)), fill = "tomato3")
grid.arrange(p1,p2,p3,p4)
```

# Ordinary Least Square

## Linear Regression in R : example

```
require(gridExtra)
g = ggplot(journals)
p1 = g + geom_histogram(aes(x = citeprice), fill = "tomato3")
p2 = g + geom_histogram(aes(x = subs), fill = "tomato3")
p3 = g + geom_histogram(aes(x = log(citeprice)), fill = "tomato3")
p4 = g + geom_histogram(aes(x = log(subs)), fill = "tomato3")
grid.arrange(p1,p2,p3,p4)
```

$\Rightarrow$ Wide range of the variables + big skewness : linear log-log model !

$$\log(subs_i) = \beta_1 + \beta_2 \log(citeprice_i) + \varepsilon_i$$

# Ordinary Least Square

## Linear Regression in R : example

How to fit the log-log linear model with R ? To estimate our model and store it in `journals_lm` :

- `journal_lm = journals %>% lm(log(subs) ~ log(citeprice), data = .)`
- See results : `summary(jour_lm)`
- What is the $\beta$ estimated ?

# Ordinary Least Square

## Linear Regression in R : example

ggplot2 also has a linear fitting function that directly plots the output !

- `ggplot(journals, aes(x = log(citeprice), y = log(subs))) + geom_point() + stat_smooth(method = "lm", col = "tomato3")`

# Ordinary Least Square

## Linear Regression in R : example

- coef() : extracts the regression coefficients
- confint() : returns confidence intervals on the estimates
- residuals() : extracts the residuals
- fitted() : returns the fitted values
- predict() : computes predictions for new data
- plot() : produces diagnostic plots

# Ordinary Least Square

## Linear Regression in R : example

Basic R plot functions offer some diagnostic plots

1. residuals vs. fitted values
2. QQ plot for normality
3. Scale-location plot

# Ordinary Least Square

## Linear Regression in R : example

Basic R plot functions offer some diagnostic plots

1. residuals vs. fitted values (Useful to test $\mathbb{E}(\varepsilon \mid X) = 0$)

# Ordinary Least Square

## Linear Regression in R : example

Basic R plot functions offer some diagnostic plots

1. residuals vs. fitted values (Useful to test $\mathbb{E}(\varepsilon \mid X) = 0$)

   plot(journal_lm, which = 1)

# Ordinary Least Square

## Linear Regression in R : example

Basic R plot functions offer some diagnostic plots

1. residuals vs. fitted values (Useful to test $\mathbb{E}(\varepsilon \mid X) = 0$)

   plot(journal_lm, which = 1)

2. QQ plot for normality (error terms are i.i.d. and N(0,x) ?)

# Ordinary Least Square

## Linear Regression in R : example

Basic R plot functions offer some diagnostic plots

1. residuals vs. fitted values (Useful to test $\mathbb{E}(\varepsilon \mid X) = 0$)

   plot(journal_lm, which = 1)

2. QQ plot for normality (error terms are i.i.d. and N(0,x) ?)

   plot(journal_lm, which = 2)

# Ordinary Least Square

## Linear Regression in R : example

Basic R plot functions offer some diagnostic plots

1. residuals vs. fitted values (Useful to test $\mathbb{E}(\varepsilon \mid X) = 0$)

   plot(journal_lm, which = 1)

2. QQ plot for normality (error terms are i.i.d. and N(0,x) ?)

   plot(journal_lm, which = 2)

3. Scale-location plot (homoskedasticity ? )

# Ordinary Least Square

## Linear Regression in R : example

Basic R plot functions offer some diagnostic plots

1. residuals vs. fitted values (Useful to test $\mathbb{E}(\varepsilon \mid X) = 0$)

   plot(journal_lm, which = 1)

2. QQ plot for normality (error terms are i.i.d. and N(0,x) ?)

   plot(journal_lm, which = 2)

3. Scale-location plot (homoskedasticity ? )

   plot(journal_lm, which = 3)

# Ordinary Least Square

## Linear Regression in R : example

Testing hypothesis ? Test the hypothesis that the elasticity of the number of library subscriptions with respect to the price per citation equals -0.5, i.e. $H_0 : \beta_2 = -0.5$

- linearHypothesis(journal_lm, "log(citeprice) = -0.5")

# Multivariate Linear Regression

## Introduction

Illustration with CPS of 1988

```
data("CPS1988", package = "AER")
cps <- as_tibble(CPS1988)
cps
```

3 continuous variables : a) wage, b) education, c) experience and 4 categorical variables : a) ethnicity, b) smsa : live in an urban region, c) region, d) parttime.

# Multivariate Linear Regression

## Introduction

Interested in the model :

$$\log(wage_i) = \beta_1 + \beta_2\, exp_i + \beta_3\, exp_i^2 + \beta_4\, education_i + \beta_5\, ethnicity_i + \varepsilon_i$$

# Multivariate Linear Regression

## Introduction

Interested in the model :

$$\log(wage_i) = \beta_1 + \beta_2\, exp_i + \beta_3\, exp_i^2 + \beta_4\, education_i + \beta_5\, ethnicity_i + \varepsilon_i$$

In R, try :

```
cps_lm = cps %>% lm(log(wage) ~ experience + I(experience^2) +
education + ethnicity, data = .)
```

# Multivariate Linear Regression

## Dummy variables

- In the above summary, have "ethnicityafam" because "cauc" is taken as the reference category
- Un-ordered factors are always handled like this by R : R always creates a reference category
- Can modify the reference category with "relevel(<factor>, ref = <new_reference>)"
- Alternatively, can remove the intercept to avoid the multicolinearity with "-1"

# Multivariate Linear Regression

## Dummy variables

**Exercise 3** : use afam as reference category

# Multivariate Linear Regression

## Dummy variables

**Exercise 3** : use afam as reference category

`lm(log(wage)` $\sim$ `experience + I(`$experience^2$`) + education + relevel(ethnicity, ref = "afam"), data = cps)`

# Multivariate Linear Regression

## Dummy variables

**Exercise 3** : use afam as reference category

`lm(log(wage)` $\sim$ `experience + I(`$experience^2$`) + education + relevel(ethnicity, ref = "afam"), data = cps)`

**Exercise 3bis** : remove the intercept

# Multivariate Linear Regression

## Dummy variables

**Exercise 3** : use afam as reference category

`lm(log(wage)` $\sim$ `experience + I(`$experience^2$`) + education + relevel(ethnicity, ref = "afam"), data = cps)`

**Exercise 3bis** : remove the intercept

`lm(log(wage)` $\sim$ `experience + I(`$experience^2$`) + education + ethnicity - 1, data = cps)`

# Multivariate Linear Regression

## Interactions

Within formulas, the mathematical operators have different meaning. For x and y two variables that are respectively continuous and discrete :

- ‘x + y‘ : add the two variables in the formula
- ‘x : y‘ : add the interaction between ‘x‘ and ‘y‘
- ‘x * y‘ : add the two variables and their interaction, i.e. ‘x + y + x : y‘
- ‘y / x‘ : compute an explicit slope estimate for each category of ‘y‘

# Multivariate Linear Regression

## Interactions

**Exercise 4** : With the same model, add an interaction term in order to study the interaction between education and ethnicity

# Multivariate Linear Regression

## Interactions

**Exercise 4** : With the same model, add an interaction term in order to study the interaction between education and ethnicity

cps_int = cps %>% lm(log(wage) ~ experience + I($experience^2$) + education * ethnicity, data = .)

# Multivariate Linear Regression

## Interactions

**Exercise 4bis** : could also fit separate regressions for African-Americans and Caucasians using "/"as in lm(y $\sim$ category / (x + y) , data = .)

# Multivariate Linear Regression

## Interactions

**Exercise 4bis** : could also fit separate regressions for African-Americans and Caucasians using "/"as in lm(y $\sim$ category / (x + y) , data = .)

cps_sep = cps %>% lm(log(wage) $\sim$ ethnicity / (experience + I($experience^2$) + education) , data = .)

# Weighted least square and co.

## What and why ?

- In our work on the effect of price on journals' subscriptions, our data featured heteroskedasticity (plot(journals_lm, which = 3)).
- What should we do ?

# Weighted least square and co.

## What and why ?

- Standard remedy : weighted least square
- Test 1 : use the inverse of the square of the price per citation as a weight

  jour_wls1 <- journals %>% lm(log(subs) ~ log(citeprice), data = ., weights = 1 / $citeprice^2$)

  plot(jour_wls1, which = 3)

# Weighted least square and co.

## What and why ?

- Standard remedy : weighted least square
- Test 1 : use the inverse of the square of the price per citation as a weight

  jour_wls1 <- journals %>% lm(log(subs) ∼ log(citeprice), data = .,
  weights = 1 / $citeprice^2$)

  plot(jour_wls1, which = 3)

- What is the result ?

# Weighted least square and co.
## FGLS

Very frequently : no clue what weight to use, leading to the feasible generalized least square (FGLS). Solution :

1. fit the model as if were homoskedastic

$$\log(subs_i) = \beta_1 + \beta_2 citeprice_i + \varepsilon_i$$

2. fit a linear model on the squared residuals

$$\log((subs_i - \hat{\beta}_1 - \hat{\beta}_2 citeprice_i)^2) = \alpha_1 + \alpha_2 citeprice_i$$

3. fit the model, using as weights the predicted values of residuals

**Exercise 5** : Implement this !(hint : use the residuals() and fitted() functions)

# Other models

## Probit, Logit and co.

- Probit, logit and similar models are referred to as generalized linear models (GLMs)
- Most of the time, closed-form solutions for the estimator do not exist, and the estimation occurs via some numerical method
- In R, most of these estimation procedures are already coded for you in the function glm()

# Other models

## Probit, Logit and co.

- Logit and probit regressions take the form :

$$\mathbb{E}(y_i \mid x_i) = p_i = F(x_i^T \beta)$$

- where $F$ is the standard normal cdf in the probit case, and the logistic CDF in the logit case
- glm() has two key arguments, family (here = binomial) and link (here = probit or logit)