OLS Estimation in gretl

Applied Economics

Department of Economics
Universidad Carlos III de Madrid

Outline

Basic commands and functions

Basic commands and functions for OLS Estimation

- `ols`: computes ordinary least squares
- `$coeff`: returns a column vector containing the estimated coefficients for the last model
- `$yhat`: a function which computes predicted values in running sample
- `$uhat`: a function which computes residuals in running sample
- `$sample`: a function which identifies the observations used in estimation
- `omit/add`: tests joint significance
- `restrict`: tests restrictions using the Wald test

ols *depvar indepvars* ––robust ––simple-print ––print-final

- you must include the constant as *const*
- ––robust: correct standard errors under our assumptions
- otherwise, only correct if variance of error term is constant (homoskedasticity)
- ––print-final: (only with loops) output shown only at final iteration
- various variables may be retrieved using genr after ols

- ols wages const educ exp expsq ––robust ––simple-print
- genr uhat = $uhat

Using the estimated coefficients after OLS

After ols, we can work in our script with the estimated coefficients

- $coeff: returns a column vector containing the estimated coefficients for the last model
- $coeff(x1): the estimated parameter for x1
- if we store these values in a vector, then we can access each of them using standard vector notation

a simple example

- ols wage const educ age agesq --robust
- b = $coeff
- genr yhat = b[1]+b[2]*educ+b[3]*age+b[4]*agesq
- genr uhat = wages - yhat

Other accessors after OLS

other variables may be retrieved using genr after ols

- $yhat: returns the fitted values from the last regression
- $uhat: returns the residuals from the last model
- $sample: 1: used , 0: in sample but not used, NA: outside the sample
- see "Accessors" in functions reference

a complete example using a “list”

- list indepvar = educ age agesq
- ols wage const indepvar −−robust
- genr olssmpl = $sample
- genr uhat = $uhat

omit *varlist* −−wald −−quiet

- must follow an estimation command like ols
- test joint significance of *varlist* using, by default, an F-test
- the restricted model replaces the original as the "current model"
- −−wald: asymptotic Wald chi-square test
- −−quiet: only the result of the test is printed
- results can be retrieved using $test and $pvalue

ols wage const educ age agesq −−robust

- omit age agesq −−wald

add *varlist* –quiet

- must be invoked after an estimation command
- *varlist* are added to previous model and the new model is estimated
- computes an F-test (and its p-value) for significance of *varlist*
- ––quiet: only prints the test and its p-value

restrict −−quiet −−bootstrap

restrict −−quiet −−bootstrap
equations-describing-restrictions
end restrict

- after ols, computes a Wald test (unless −−bootstrap)
- each linear restriction is given as an equation
- parameters are referenced by b[*i*] or b[*varname*]
- for nonlinear restrictions, see help restrict for details

- restrict
- b[2] - 2*b[3] = 0
- end restrict