ECON 4160: Econometrics–Modelling and Systems Estimation:

Computer Class # 2

André K. Anundsen

Norges Bank

September 10, 2014

Practical information

Who am I? → André K. Anundsen (Researcher at Norges Bank)
Email: *andre-kallak.anundsen@norges-bank.no*
Responsible for the rest of the CPU classes

Outline

Data sets

Review of some elementary econometrics

The “test battery”

Some first Monte Carlos

Testing parameter stability

Though the teaching plan also says that we will cover NLS, encompassing tests and vector autoregressive (VAR) models today, we won’t have time!!! But, don’t despair; we shall (try to) find time for all these topics in another CC (specification and estimation of VAR models will be covered in detail already in the next class)!!

Data sets for today (posted on the web page):

- KonsDataSim.zip (same as you used last time)

Regression with experimental and non-experimental data

Consider the modelling task with experimental/lab data:

$$\underset{\text{result}}{Y_i} = \underset{\text{input}}{g(X_i)} + \underset{\text{shock}}{v_i} \tag{1}$$

and compare with the situation with non-experimental, real-world data:

$$\underset{\text{observed}}{Y_i} = \underset{\text{explained}}{f(X_i)} + \underset{\text{remainder}}{\varepsilon_i} \tag{2}$$

Clearly, we know much less about the match between $f(X_i)$ and Y_i in the non-experimental case: We simply can't control the input and then study the output. Often, we may not even know all variables that are included in the vector X_i! All our choices of functional form, $f(\cdot)$, and explanatory variables, X_i, will be reflected in the remainder ε_i:

$$\varepsilon_i = Y_i - f(X_i) \tag{3}$$

However, we will follow custom and refer to ε_i as the *disturbance*, and the estimated counterpart, $\hat{\varepsilon}_i$, as the *residual*.

When we choose regression, the $f(X_i)$ function is the conditional expectation:

$$f(X_i) = E(Y_i|X_i) \tag{4}$$

and it follows that

$$E(\varepsilon_i|X_i) = E((Y_i - f(X_i))|X_i) = 0 \tag{5}$$

- But if $E(Y_i|X_i)$ is incomplete or wrong relative to the DGP, ε_i will in general be correlated with omitted variables
- The assumptions that we make about the disturbances , e.g., the "classical assumptions" in applied modelling, are only tentative, and we need to validate them after estimation
- This is called residual mis-specification testing

Residual mis-specification overview

	Disturbances ε_i are:			
X_i	heteroscedastic		autocorrelated	
	$\hat{\beta}_1$	$\widehat{Var}(\hat{\beta}_1)$	$\hat{\beta}_1$	$\widehat{Var}(\hat{\beta}_1)$
exogenous	unbiased consistent	wrong	unbiased consistent	wrong
predetermined	biased consistent	wrong	biased inconsistent	wrong

Test battery in PcGive I: Non-normality

Normality, Jarque and Bera (1980): (Note: Small sample correction in PcGive)
Test the joint hypothesis of no skewness and no excess kurtosis (3^{rd} and 4^{th} moment of the normal distr.):

$$JB = \frac{n}{6}\left(\text{Skewness}^2 + \frac{1}{4}Excess\ kurtosis^2\right) \tag{6}$$

The sample skewness and excess kurtosis (skewness $=0$ and kurtosis $=3$ for normal distr.) are defined as follows

$$\text{Skewness} = \frac{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{\varepsilon}_i - \bar{\hat{\varepsilon}}\right)^3}{\left(\frac{1}{n}\sum_{i=1}^{n}\left(\hat{\varepsilon}_i - \bar{\hat{\varepsilon}}\right)^2\right)^{\frac{3}{2}}} = \frac{\frac{1}{n}\sum_{i=1}^{n}\hat{\varepsilon}_i^3}{\left(\frac{1}{n}\sum_{i=1}^{n}\hat{\varepsilon}_i^2\right)^{\frac{3}{2}}} \tag{7}$$

$$\text{Excess kurtosis} = \frac{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{\varepsilon}_i - \bar{\hat{\varepsilon}}\right)^4}{\left(\frac{1}{n}\sum_{i=1}^{n}\left(\hat{\varepsilon}_i - \bar{\hat{\varepsilon}}\right)^2\right)^{\frac{4}{2}}} = \frac{\frac{1}{n}\sum_{i=1}^{n}\hat{\varepsilon}_i^4}{\left(\frac{1}{n}\sum_{i=1}^{n}\hat{\varepsilon}_i^2\right)^{2}} \tag{8}$$

Null is normality.

Test battery in PcGive II: Heteroskedasticity

Heteroskedasticity, White (1980):
Auxiliary regression:

$$\hat{\varepsilon}_i^2 = \beta_0 + \beta_1 X_i + \beta_2 X_i^2 + u_i \tag{9}$$

Test that $\beta_1 = \beta_2 = 0$ against non-zero using an F-test.
Hetero X (two or more regressors), White (1980)
Auxiliary regression:

$$\hat{\varepsilon}_i^2 = \beta_0 + \beta_1 X_i + \beta_2 Z_i + \beta_3 X_i Z_i + \beta_4 X_i^2 + \beta_5 Z_i^2 + u_i \tag{10}$$

Test that $\beta_1 = \beta_2 = \beta_3 = \beta_4 = \beta_5 = 0$ against non-zero using an F-test.

In both cases the null is homoskedasticity.

Test battery in PcGive III: Autocorrelation

Autocorrelation, Godfrey (1978):
Auxiliary regression (note, *i* is now time unit! Call it *t*):

$$\hat{\varepsilon}_t = \beta_0 + \sum_{i=1}^{p} \beta_i \hat{\varepsilon}_{t-i} + \beta_{p+1} X_t + u_t \tag{11}$$

A test for p^{th} order autocorrelation is then to test that $\beta_1 = \beta_2 = \cdots = \beta_p = 0$ against non-zero using an F-test. Null is no autocorrelation.

Test battery in PcGive IV: Autoregressive conditional heteroskedasticity

ARCH, Engle (1982):
Auxiliary regression:

$$\hat{\varepsilon}_t = \beta_0 + \sum_{i=1}^{p} \beta_i \hat{\varepsilon}_{t-i}^2 + u_t \tag{12}$$

A test for p^{th} order ARCH is then to test that $\beta_1 = \beta_2 = \cdots = \beta_p = 0$ against non-zero using an F-test. Null is no ARCH.

Test battery in PcGive V: Regression Specification Test

RESET, Ramsey (1969):
Auxiliary regression (Note 2,3 in PcGive means squares and cubes!):

$$\hat{\varepsilon}_t = \beta_0 + \beta_1 X_t + \beta_2 \hat{Y}_t^2 + \beta_3 \hat{Y}_t^3 + u_t \tag{13}$$

A test for correct specification is then to test that $\beta_2 = \beta_3 = 0$ against non-zero using an F-test.
Null is no specification error.

Open *KonsDataSim.in7* and use "Single-equation dynamic modelling" under " Models for time series data" in PcGive to estimate a simple static model between consumption (C) and income (I). Do the residuals appear to be well behaved?

What is Monte Carlo simulation? I

Say that the process that has generated the data (the data generating process, or the DGP) takes the following form:

$$y_t = \beta_0 + \beta_1 x_t + \varepsilon_t \tag{14}$$

where ε_t is WN. Now, assume that we had some data for the period $t = 1, \ldots, T$ on y_t and x_t, and that we want to pin down β_0 and β_1 (our parameters of interest)

As long as $Cov(x_t, \varepsilon_t) = 0$, you know that the OLS estimator is BLUE! How could we confirm this by simulation?

What is Monte Carlo simulation? II

1. Fix β_0 and β_1 in (14) at some values, e.g. $\beta_0 = 2$ and $\beta_1 = 1.5$
2. Generate some numbers for the time series x_t on a sample $t = 1\ldots, T$
3. Assume e.g. that $\varepsilon_t \sim N(0,1)$, and draw T numbers from the standard normal distribution
4. Then, the time series y_t for the sample $t = 1,\ldots T$ will follow by definition from the DGP!
5. Estimate an equation of the form (14) by OLS and collect your β_0 and β_1 estimates; call them $\hat{\beta}_0^1$ and $\hat{\beta}_1^1$
6. Now, repeat the steps 1–5 M times, and calculate $\beta_0^{MC} = \frac{\sum_{m=1}^{M}\hat{\beta}_0^m}{M}$ and $\beta_1^{MC} = \frac{\sum_{m=1}^{M}\hat{\beta}_1^m}{M}$!

What is Monte Carlo simulation? III

As we have seen, β_0^{MC} and β_1^{MC} are nothing but the mean estimates obtained by repeating 1–5 M times. From the law of large numbers, we know that β_0^{MC} and β_1^{MC} converge to $E(\hat{\beta}_0)$ and $E(\hat{\beta}_1)$ as $M \to \infty$.

But then, we know that a measure of the biases of these estimates would be: $\beta_0^{MC} - \beta_0$ and $\beta_1^{MC} - \beta_1$. What we can also do is to consider a recursive Monte Carlo, where we e.g. vary the sample size from say $T = 20$ to $T = 500$ in increments of 20 to check the bias of the OLS estimator for different samples!

Experiment 1: The properties of the OLS estimator when we estimate the true DGP I

Suppose the DGP is given by:

$$y_t = 2 + 1.5x_t + \varepsilon_t$$

We want to explore the properties of the OLS estimator for different sample sizes when the classical assumptions are satisfied (we keep the number of MC replications at $M = 1000$ for all samples). Do we expect the OLS estimator to be unbiased? Let's first derive this theoretically! In matrix notation, the equation of interest takes the following form:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

where in our example $\mathbf{y}$ is $T \times 1$, $\mathbf{X}$ is $T \times 2$ (with ones in the first column), $\boldsymbol{\beta}$ is 2×1 and $\boldsymbol{\varepsilon}$ is $T \times 1$. The OLS estimator is given as:

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}$$

Experiment 1: The properties of the OLS estimator when we estimate the true DGP II

By substituting in for **y**, we get:

$$\begin{aligned}\hat{\beta} &= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\left(\mathbf{X}\beta+\varepsilon\right)\\ &= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\left(\mathbf{X}'\mathbf{X}\right)\beta+\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\varepsilon\\ &= \beta+\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\varepsilon\end{aligned}$$

Taking expectations, we get:

$$\begin{aligned}E\left(\hat{\beta}\right) &= \beta+E\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\varepsilon\right)\\ &= \beta+\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'E\left(\varepsilon\right)=\beta\end{aligned}$$

i.e. we expect the OLS estimator to be unbiased! Let's check this out by MC!

Experiment 2: The properties of the OLS estimator when we have an omitted variable I

Suppose the DGP is given by:

$$y_t = 2 + 1.5x_t - 2z_t + u_t$$

This means that, if we were to estimate the true DGP, we would consider a model of the following form:

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}\delta + \mathbf{u}_t$$

where the notation is as previously, $\mathbf{Z}$ is a $T \times 1$ matrix and δ is a scalar. Suppose instead that the model we consider is the same model as considered previously:

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon$$

with the OLS estimator given by:

$$\hat{\beta} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}$$

Experiment 2: The properties of the OLS estimator when we have an omitted variable II

Would this be an unbiased estimator? Let's do exactly the same as previously (substitute in for $\mathbf{y}$ from the DGP). We get:

$$\begin{aligned}\hat{\beta} &= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\left(\mathbf{X}\beta + \mathbf{Z}\delta + \mathbf{u}\right)\\ &= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\left(\mathbf{X}'\mathbf{X}\right)\beta + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Z}\delta + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{u}\\ &= \beta + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Z}\delta + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{u}\end{aligned}$$

Taking expectations, we get:

$$\begin{aligned}E\left(\hat{\beta}\right) &= \beta + E\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Z}\delta\right) + E\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\varepsilon\right)\\ &= \beta + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Z}\delta + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'E\left(\varepsilon\right)\\ &= \beta + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Z}\delta \neq \beta\end{aligned}$$

i.e. we expect the OLS estimator to be **biased**! Again, let's check this out by some simulations!

Experiment III: The Hurwitz bias in an autoregressive model

In Lecture # 3, RNy introduced the AR(1) model as an example of a simple time series model:

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \varepsilon_t, \quad |\phi_1| < 1 \tag{15}$$

He also demonstrated that the MLE of ϕ_1 is given by:

$$\hat{\phi}_1 = \phi_1 + \frac{\sum_{t=1}^{T} \varepsilon_t \left(Y_{t-1} - \bar{Y}\right)}{\sum_{t=1}^{T} \left(Y_{t-1} - \bar{Y}\right)^2} \tag{16}$$

And that we cannot prove that $E(\hat{\phi}_1) = \phi_1$, i.e. that the MLE of ϕ_1 is unbiased

Consider the following DGP:

$$Y_t = 2 + 0.6Y_{t-1} + \varepsilon_t, \ \ \varepsilon_t \sim N(0,1) \tag{17}$$

How will the bias of the OLS estimator be in a finite sample? Let's check this out by a Monte Carlo!!

Recursive graphs of parameters I

Consider the model

$$y_t = \beta_1 + \beta_2 x_t + \varepsilon_t, \;\; t = 1, 2, \ldots, T, \tag{18}$$

A simple and intuitive method for assessing the hypothesis that the parameters β_1, β_2, $\sigma = \sqrt{Var(\varepsilon_t)}$ are stable, is to graph the output from a recursive estimation.

Recursive graphs of parameters II

Recursive estimation works in the following way:

1. Estimate on a short sample: $t = 1, 2, ..., T_1,\ T_1 < T$
2. Extend the sample with a single observation, and estimate the model on the sample $t = 1, 2, ..., T_1 + 1,\ T_1 + 1 < T$
3. Continue until all observations are part of the sample, i.e. $t = 1, 2, ..., T$
4. The sequence of estimates, $\hat{\beta}_1(j)$, $\hat{\beta}_2(j)$, and $\hat{\sigma}(j)$ $(j = T_1, T_1 + 1, \ldots, T)$, can be plotted against time

Pc-Give plots e.g., $\hat{\beta}_2(j)$ together with $\pm 2\sqrt{Var(\hat{\beta}_1(j))}$, and $\hat{\sigma}(j)$ together with the *1-step residuals*, which are

$$y_j - \hat{\beta}_1(j) - \hat{\beta}_2(j)x_j, \quad j = T_1, T_1 + 1, ..., T$$

Let us look at the recursive estimates when we estimate the static Keynesian consumption function using *KonsDataSim.in7*

Chow-tests of parameter stability

- If we have a hypothesis about when a structural break occurs, we can test that hypothesis
- Let T_1 denote the last period with the "old" regime and let $T_1 + 1$ denote the first period of the "new" regime:

  $$y_t = \beta_1 + \beta_2 x_t + \varepsilon_t, \; t = 1, 2, \ldots, T_1 \text{ and}$$
  $$y_s = \gamma_1 + \gamma_2 x_s + \epsilon_s, \; s = T_1 + 1, 2, \ldots, T.$$

  then

  $$H_0\text{: } \beta_1 = \gamma_1, \, \beta_2 = \gamma_2 \text{ vs } H_1\text{: } \beta_1 \neq \gamma_1, \, \beta_2 \neq \gamma_2.$$

- There are two well known statistics for these cases, both due to Chow (1960) and referred to as *Chow tests*.

2-sample Chow-test

Let SSE be the Sum of Squared Errors (which of course corresponds to RSS or SSR):

- ◇ SSE_1 is for the first sample $(t = 1, 2, .., T_1)$
- ◇ SSE_2 is for the second sample $(t = T_1 + 1, 2, .., T_1)$
- ◇ $SSE_U = SSE_1 + SSE_2$
- ◇ SSE_R is the SSE when the whole sample is used, i.e. under H_0

The test statistic in our example is:

$$F_{Chow2} = \frac{SSE_R - SSE_U}{SSE_U} \cdot \frac{(T-4)}{2} \sim F(2, T-4).$$

In general, it is:

$$F_{Chow2} = \frac{SSE_R - SSE_U}{SSE_U} \cdot \frac{T-2K}{K} \sim F(K, T-2K).$$

where K is the number of parameters

Insufficient observations Chow-test

- Consider $T - T_1 < K$ (the sample for the second period), where we remember that K is the number of parameters in the model. In this case we have:
  - The same SSE_R as for the 2-sample Chow-test (i.e. based on the full sample)
  - But SSE_U is only based on the first T_1 observations

This "predictive" Chow-test is given as

$$F_{ChowP} = \frac{SSE_R - SSE_U}{SSE_U} \cdot \frac{T_1 - K}{T - T_1} \sim F(T - T_1, T_1 - K)$$

Insufficient observations Chow-test

- If we have no clear idea about the dating of a regime shift, a graph with the whole sequence of predictive Chow tests is useful
- Chow tests rely on constant and equal variances of the disturbances. Hence, it is good practice to plot the recursively estimated $\hat{\sigma}$

Recursive Chow-test: 1-step test

The Test-Menu Graphics in PcGive contains three version of the F_{ChowP} test!

- *1-step test*: Both T_1 and T is increased by one observation sequentially, so that $N = T - T_1$ is always 1. Naturally, the test statistic is:
    -

$$
\begin{aligned}
F_{ChowP} &= \frac{SSE_R - SSE_U}{SSE_U} \cdot \frac{T_1 - K}{T - T_1} \\
&= \frac{SSE_R - SSE_U}{SSE_U} \cdot \frac{T_1 - K}{1} \sim F(1, T_1 - K)
\end{aligned}
$$

Example:

1. Estimate (18) on the sample $t = 1, ..., 10$ and $t = 1, ..., 11$ and calculate F_{ChowP}
2. Estimate (18) on the sample $t = 1, ..., 11$ and $t = 1, ..., 12$ and calculate F_{ChowP}
3. Continue like this! The graph plots the sequence of F_{ChowP}

Recursive Chow-test: Break-point Chow test

- *Break-point Chow test*, called *N-dn CHOWS* (N down Chow Statistic) in PcGive Here T_1 is increased by one observation, then by 2 and so on until $T_1 = T$ (T is always fixed!). Hence, $N = T - T_1$ is decreasing (N down) through the sequence of tests. The test statistic is:

$$F_{ChowP} = \frac{SSE_R - SSE_U}{SSE_U} \cdot \frac{T_1 - K}{T - T_1} \sim F(T - T_1, T_1 - K)$$

Example

1. Estimate (18) on the sample $t = 1, ..., 10$ and $t = 1, ..., 200$ and calculate F_{ChowP}
2. Estimate (18) on the sample $t = 1, ..., 11$ and $t = 1, ..., 200$ and calculate F_{ChowP}
3. Continue like this! The graph plots the sequence of F_{ChowP}

Recursive Chow-test: Forecast Chow test

The final version is the:

- *Forecast Chow test*, called *N-up CHOWS* (N up Statistic): Here T_1 is kept fixed and $T - T_1$ is first 1, then 2, and so on. Hence, $N = T - T_1$ is increasing through the sequence of tests (N up). The test statistic is:

$$F_{ChowP} = \frac{SSE_R - SSE_U}{SSE_U} \cdot \frac{T_1 - K}{T - T_1} \sim F(T - T_1, T_1 - K)$$

Example

1. Estimate (18) on the sample $t = 1, ..., 10$ and $t = 1, ..., 199$ and calculate F_{ChowP}
2. Estimate (18) on the sample $t = 1, ..., 10$ and $t = 1, ..., 200$ and calculate F_{ChowP}
3. Continue like this! The graph plots the sequence of F_{ChowP}

- All the Recursive Chows are scaled by (1-off) critical values. A 1% significance level is the default, so that the critical values becomes a straight line at unity

You can easily calculate the different recursive Chow test statistics yourself "by hand" by using OLS, and then compare what you get to what the program tells you (choose *Write results instead of graphing*). This is a nice way to become familiar with the tests!

For now, you can open *KonsDataSim.in*7, and we shall see how to do these tests in PcGive!