Efficient Programming in Stata and Mata II: Obtaining Non-Standard Distributions for a Cointegration Test via Simulation

Sebastian Kripfganz
University of Exeter Business School

Daniel C. Schneider
Max Planck Institute for Demographic Research

German Stata Users Group Meeting, June 22, 2018, Konstanz
Last Year’s Talk

• efficient coding strategies:
  • use common sense
  • use your knowledge of your software (Stata, of course!)
  • use your knowledge of matrix algebra

• case study: the -ardl- estimation command
  • last year: optimal lag selection
  • this talk: simulation of finite sample distributions
Stationarity vs. Non-Stationarity

- fundamental distinction in time series analysis (TSA)
- mostly about time series with a unit root: I(0) vs. I(1)
- non-stationary TS behave fundamentally different
Multiple Time Series Analysis

Long-run relationship: Some time series are bound together due to equilibrium forces even though the individual time series might move considerably.
The ARDL Model and the Bounds Test

- ARDL\((p, q, \ldots, q)\) model:

\[
y_t = c_0 + c_1 t + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{i=0}^{q} \beta_i' x_{t-i} + \varepsilon_t,
\]

with \(x_t\) a \(K \times 1\) vector.

- Reparameterization in error-correction (EC) form:

\[
\Delta y_t = c_0 + c_1 t - \alpha(y_{t-1} - \theta x_{t-1}) + \sum_{i=1}^{p-1} \psi_{yi} \Delta y_{t-i} + \omega' \Delta x_t + \sum_{i=1}^{q-1} \psi_{xi}' \Delta x_{t-i} + \varepsilon_t,
\]

- Pesaran / Shin / Smith (2001) (PSS) derive the asymptotic coefficient distributions under the opposing assumptions of stationary vs. non-stationary regressors, the basis for their bounds test for a levels relationship.

- They provide critical values (CV) tables obtained via simulation.
### ARDL Toy Model Estimation

```stata
. ardl w prod union ur, ec maxlag(6) dots trend(qtime) restricted vsquish
```

**Optimal lag selection, % complete:**

<table>
<thead>
<tr>
<th>20%</th>
<th>40%</th>
<th>60%</th>
<th>80%</th>
<th>100%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BIC optimized over 2058 lag combinations

ARDL(2,0,2,0) regression

Sample: 1971q3 - 1997q4

<table>
<thead>
<tr>
<th></th>
<th>Number of obs</th>
<th>R-squared</th>
<th>Adj R-squared</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log likelihood</td>
<td>330.70424</td>
<td>0.2637</td>
<td>0.2029</td>
</tr>
</tbody>
</table>

|          | Coef. | Std. Err. | t     | P>|t| | [95% Conf. Interval] |
|----------|-------|-----------|-------|-------|----------------------|
| D.w      |       |           |       |       |                      |
| ADJ      |       |           |       |       |                      |
| W        | -0.240 | 0.063     | -3.827 | 0.000 | -0.365 to -0.116    |
| L1.      |       |           |       |       |                      |
| LR       | prod  | 0.416     | 0.208  | 1.998 | 0.049 to 0.829      |
|          | union | -0.210    | 0.235  | -0.893| 0.374 to 0.256      |
|          | ur    | 0.039     | 0.017  | 2.382 | 0.019 to 0.072      |
|          | qtime | 0.003     | 0.001  | 2.962 | 0.004 to 0.005      |
| SR       |       |           |       |       |                      |
| W        | -0.203 | 0.094     | -2.161 | 0.033 | -0.389 to -0.017    |
| LD.      | union | 0.058     | 0.597  | 0.097 | 0.923 to 1.128      |
|          | D1.   | -1.535    | 0.596  | -2.574| 0.012 to -2.719     |
|          | LD.   | 0.527     | 0.153  | 3.454 | 0.001 to 0.830      |
|          | _cons|           |       |       |                      |
ARDL Toy Model Estimation

```
. estat btest

note: estat btest has been superseded by estat ectest
as the prime procedure to test for a levels relationship.
(click to run)
```

Pesaran/Shin/Smith (2001) ARDL Bounds Test
H0: no levels relationship

<table>
<thead>
<tr>
<th>[I_0]</th>
<th>[I_1]</th>
<th>[I_0]</th>
<th>[I_1]</th>
<th>[I_0]</th>
<th>[I_1]</th>
<th>[I_0]</th>
<th>[I_1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>L_1</td>
<td>L_1</td>
<td>L_05</td>
<td>L_05</td>
<td>L_025</td>
<td>L_025</td>
<td>L_01</td>
<td>L_01</td>
</tr>
</tbody>
</table>

F = 3.863  
t = -3.827

Critical Values (0.1-0.01), F-statistic, Case 4

| k_3   | 2.97  | 3.74  | 3.38  | 4.23  | 3.80  | 4.68  | 4.30  | 5.23  |

accept if F < critical value for I(0) regressors
reject if F > critical value for I(1) regressors

Critical Values (0.1-0.01), t-statistic, Case 4

| k_3   | -3.13 | -3.84 | -3.41 | -4.16 | -3.65 | -4.42 | -3.96 | -4.73 |

accept if t > critical value for I(0) regressors
reject if t < critical value for I(1) regressors

k: # of non-deterministic regressors in long-run relationship
Critical values from Pesaran/Shin/Smith (2001)
Simulation Project Outline

- PSS bounds test very popular, but CV tables only cover a limited number of cases

⇒ computational / simulation project:
1. simulate distributions for all combinations of c, l, k, q, T
2. store calculated statistics / distributions
3. run response surface regressions (RSR), where the depvars are distributional quantiles
4. implement and distribute an ARDL postestimation feature that displays RSR-based CVs / p-values
Response Surface Regressions (RSR)

• idea:
  for each c, l, k: regress quantile of distr \sim g(T,q)
  We implement variations thereof.

• use predicted values for a particular T, q as CVs in applied work


• Other Stata commands, e.g.
  • ersur (Baum/Otero 2017)
  • kssur, ksur (Otero/Smith 2017)
The Computational Task

Similar to PSS, the DGP is

\[ y_t = y_{t-1} + \epsilon_{yt} \]
\[ x_t = Px_{t-1} + \epsilon_{xt} \]

for \( t = 1, 2, \ldots, T + 50 \) (including 50 burn-in periods), and where

\[ (y_0, x'_0)' = 0, \varepsilon_t \sim N(0, I_{k+1}) \]

and

\[ P = 0 \ (I(0) \text{ regressors}) \]
\[ P = I_k \ (I(1) \text{ regressors}) \]
The Computational Task

Project size:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
<th>Values</th>
<th># values</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>deterministics cases</td>
<td>1, 2, …, 5 (F); 1, 3, 5 (t)</td>
<td>8</td>
</tr>
<tr>
<td>l</td>
<td>integration order</td>
<td>0, 1</td>
<td>2</td>
</tr>
<tr>
<td>k</td>
<td># of regressors</td>
<td>0, 1, …, 10</td>
<td>11</td>
</tr>
<tr>
<td>q</td>
<td># of lags</td>
<td>0, 1, …, 4, 6, 8, 12</td>
<td>8</td>
</tr>
<tr>
<td>T</td>
<td>sample size</td>
<td>20, 22, …, 400, 500, 1000</td>
<td>18</td>
</tr>
<tr>
<td>r</td>
<td># replications</td>
<td></td>
<td>100,000</td>
</tr>
<tr>
<td>m</td>
<td># meta replications</td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>

Results in ~160,000,000,000 stats
Implies several months of computation (“Oh my!”)
Implies ~600GB disk space (“Oh dear!”)
Reducing Data Size

Idea, omitting details: i) round to 3 decimal places, ii) store tabulation

<table>
<thead>
<tr>
<th>cIkr_group</th>
<th>stat</th>
<th>stat3</th>
<th>tmpdif</th>
<th>statdif</th>
<th>mult</th>
</tr>
</thead>
<tbody>
<tr>
<td>2310</td>
<td>2.345145</td>
<td>2345</td>
<td>2345</td>
<td>-28655</td>
<td>1</td>
</tr>
<tr>
<td>2310</td>
<td>2.761234</td>
<td>2761</td>
<td>416</td>
<td>-30584</td>
<td>2</td>
</tr>
<tr>
<td>2310</td>
<td>2.761411</td>
<td>2761</td>
<td>0</td>
<td>-31000</td>
<td>2</td>
</tr>
<tr>
<td>2310</td>
<td>2.761932</td>
<td>2762</td>
<td>1</td>
<td>-30999</td>
<td>4</td>
</tr>
<tr>
<td>2310</td>
<td>2.761944</td>
<td>2762</td>
<td>0</td>
<td>-31000</td>
<td>4</td>
</tr>
<tr>
<td>2310</td>
<td>2.761948</td>
<td>2762</td>
<td>0</td>
<td>-31000</td>
<td>4</td>
</tr>
<tr>
<td>2310</td>
<td>2.762331</td>
<td>2762</td>
<td>0</td>
<td>-31000</td>
<td>4</td>
</tr>
<tr>
<td>2310</td>
<td>10.85794</td>
<td>10858</td>
<td>100</td>
<td>-20142</td>
<td>1</td>
</tr>
<tr>
<td>2310</td>
<td>10.99043</td>
<td>10990</td>
<td>132</td>
<td>-20010</td>
<td>1</td>
</tr>
<tr>
<td>2311</td>
<td>2.118192</td>
<td>2118</td>
<td>2118</td>
<td>-28882</td>
<td>1</td>
</tr>
<tr>
<td>2311</td>
<td>2.239101</td>
<td>2239</td>
<td>121</td>
<td>-30879</td>
<td>1</td>
</tr>
<tr>
<td>2311</td>
<td>2.241233</td>
<td>2241</td>
<td>2</td>
<td>-30998</td>
<td>1</td>
</tr>
<tr>
<td>2311</td>
<td>2.241708</td>
<td>2242</td>
<td>1</td>
<td>-30999</td>
<td>2</td>
</tr>
<tr>
<td>2311</td>
<td>2.241744</td>
<td>2242</td>
<td>0</td>
<td>-31000</td>
<td>2</td>
</tr>
</tbody>
</table>
Reducing Data Size

- Achieved size reduction: over 90%
- After `-zipfile-`, data occupy 10GB
- Solving this was crucial as now computational steps can be separated.
- But: Takes up 20% computation time
- `. help data types, . help compress`
- Data transformations and data types
  - Years, age in years
- Wish list item: if Mata supported all numeric types of Stata
  - Could implement more complex storage ideas in Mata and its mmat files
  - Could write (de-)compression in terms of a class
// ------------------ beg dosim.do -------------
args inputarg
if "`inputarg'"!=""
    confirm integer number `inputarg'
// (...) potentially some setup statements here
// like startup scripts that set matsize, maxvar, etc.
}

set rng mt64s
local laglist 1 2 3 4 6 8 12
if "`inputarg'"!=""
local laglist `inputarg'

foreach lag of local laglist {
    set seed 123456
    set rngstream `lag'
mata : dosim(`lag')
}

// ------------------ end dosim.do -------------
Windows / DOS batch file to fire up Stata instances

rem -------- beg multiinstance.bat --------
for %%c in (1 2 3 4 6 8 12) do {
    copy dosim.do dosim_multiinst_%%c.do /Y
    start "sim%%c" /D "PROJECTPATH" "STATAPATH\StataMP-64.exe" ^
    /e do dosim_multiinst_%%c.do  %%c
}
rem -------- end multiinstance.bat --------
Simulation & Multiple Stata Instances

- Multiple instances
  - help entry: [GSW] B.5 Stata batch mode
  - careful with any kind of file saving operations, e.g. logs
  - batch file to kill processes?

- RNG streams
  - new in Stata 15
  - . help set rngstream
Mata Code Optimization

- necessary to examine each expression for speed improvements
- examples of smaller improvements
  - row extraction instead of column extraction
  - inner vector product: sum of squares vs. cross() vs. multiplication
- most important code features
  - pre-calculation of cross-products, accessing through indexing
  - use pointer variables to facilitate storing numbers
  - experiment with inverters / solvers
- not pursued: C/C++
  - Stata/Mata has a MUCH better convenience-speed trade-off
  - Stata/Mata great in other respects too: version control
Usage of pointer variables

/*

Structure of returned results:

<table>
<thead>
<tr>
<th>return matrix</th>
<th>p point to</th>
<th>return matrix</th>
<th>p point to</th>
<th>return matrix</th>
<th>p point to</th>
</tr>
</thead>
<tbody>
<tr>
<td>pstatldx</td>
<td>lag-idx</td>
<td>pFkI, ptkI</td>
<td>I</td>
<td>unnamed</td>
<td>c</td>
</tr>
<tr>
<td>(returned</td>
<td>0 1 ...</td>
<td></td>
<td>0 1</td>
<td>but referenced</td>
<td>1 (2) 3</td>
</tr>
<tr>
<td>matrix)</td>
<td></td>
<td></td>
<td></td>
<td>matrices</td>
<td>4 5</td>
</tr>
<tr>
<td>stat=F</td>
<td>p p ...</td>
<td>k 0</td>
<td>p p</td>
<td>statdata</td>
<td># # # #</td>
</tr>
<tr>
<td>stat=t</td>
<td>p p ...</td>
<td>1</td>
<td>p p</td>
<td>2</td>
<td># # # #</td>
</tr>
<tr>
<td></td>
<td></td>
<td>...</td>
<td></td>
<td>...</td>
<td># # # #</td>
</tr>
<tr>
<td></td>
<td></td>
<td>kmax</td>
<td>p p</td>
<td>reps</td>
<td># # # #</td>
</tr>
</tbody>
</table>

*/
Mata Code Optimization

Loop structure

```mata
for [T] {
    for [lags] {
        // - calculate deterministics for all cases (X1)
        // - cross products thereof (XX11)
        for [reps] {
            // - random draws
            // - calculation of levels variables (X2)
            // - cross products thereof (XX22)
            // - calculation of first-difference variables (X3))
            // - cross products thereof (XX33)
            // - also calculate cross products among y, X1, X2, X3 variables (XX12, ...)
            for [cases] {
                for [k] {
                    // - check degree-of-freedom requirement
                    for [I-order] {
                        // - select / assemble matrices from parts for (un-)restricted models (F-test)
                        // - calculate (un-)restricted SSR (solver: lusolve())
                    }
                }
            }
        }
    }
}
```
Project Results: ARDL Toy Example

. quietly ardl w prod union ur, ec maxlag(6) dots trend(qtime) restricted vsquish

. estat ectest

Pesaran, Shin, and Smith (2001) bounds test

H0: no level relationship
Case 4

F = 3.863
t = -3.827

Finite sample (3 variables, 106 observations, 3 short-run coefficients)

Kripfganz and Schneider (2018) critical values and approximate p-values

<table>
<thead>
<tr>
<th></th>
<th>10%</th>
<th>5%</th>
<th>1%</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I(0)</td>
<td>I(1)</td>
<td>I(0)</td>
<td>I(1)</td>
</tr>
<tr>
<td>F</td>
<td>3.011</td>
<td>3.829</td>
<td>3.486</td>
<td>4.373</td>
</tr>
<tr>
<td>t</td>
<td>-3.116</td>
<td>-3.829</td>
<td>-3.419</td>
<td>-4.162</td>
</tr>
</tbody>
</table>

do not reject H0 if
both F and t are closer to zero than critical values for I(0) variables
(if p-values > desired level for I(0) variables)

reject H0 if
both F and t are more extreme than critical values for I(1) variables
(if p-values < desired level for I(1) variables)
Project Results: ARDL Toy Example

PSS values

<table>
<thead>
<tr>
<th></th>
<th>([I_0]_{L_1})</th>
<th>([I_1]_{L_1})</th>
<th>([I_0]_{L_05})</th>
<th>([I_1]_{L_05})</th>
<th>([I_0]<em>{L</em>{025}})</th>
<th>([I_1]<em>{L</em>{025}})</th>
<th>([I_0]_{L_01})</th>
<th>([I_1]_{L_01})</th>
</tr>
</thead>
<tbody>
<tr>
<td>(k_3)</td>
<td>2.97</td>
<td>3.74</td>
<td>3.38</td>
<td>4.23</td>
<td></td>
<td></td>
<td>4.30</td>
<td>5.23</td>
</tr>
<tr>
<td>(k)</td>
<td>-3.13</td>
<td>-3.84</td>
<td>-3.41</td>
<td>-4.16</td>
<td></td>
<td></td>
<td>-3.96</td>
<td>-4.73</td>
</tr>
</tbody>
</table>

Response surface regression based values

<table>
<thead>
<tr>
<th></th>
<th>10%</th>
<th>5%</th>
<th>1%</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I(0)</td>
<td>I(1)</td>
<td>I(0)</td>
<td>I(1)</td>
</tr>
<tr>
<td>(F)</td>
<td>3.011</td>
<td>3.829</td>
<td>3.486</td>
<td>4.373</td>
</tr>
<tr>
<td>(t)</td>
<td>-3.116</td>
<td>-3.829</td>
<td>-3.419</td>
<td>-4.162</td>
</tr>
</tbody>
</table>
Besides Cheung and Lai (1995), the existing literature largely neglects the lag-order dependence of the finite-sample critical values (t-statistic, k=0, case (iii), \( \alpha = 5\% \))
Recap

• Non-stationary time series and cointegration, ardl and the PSS bounds test
• Simulation project: Improve CV tables for bounds test
  • Storing large quantity of numbers
  • Computation time
    • Multiple Stata instances
    • Code improvements within Mata
Thank you!

Questions? Comments?
schneider@demogr.mpg.de

See also: the ardl discussion thread on the Stata Forum

. net install ardl, from(http://www.kripfganz.de/stata/)


