ardl: Estimating autoregressive distributed lag and equilibrium correction models

Sebastian Kripfganz[1] Daniel C. Schneider[2]

[1]University of Exeter Business School, Department of Economics, Exeter, UK

[2]Max Planck Institute for Demographic Research, Rostock, Germany

London Stata Conference

September 7, 2018

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

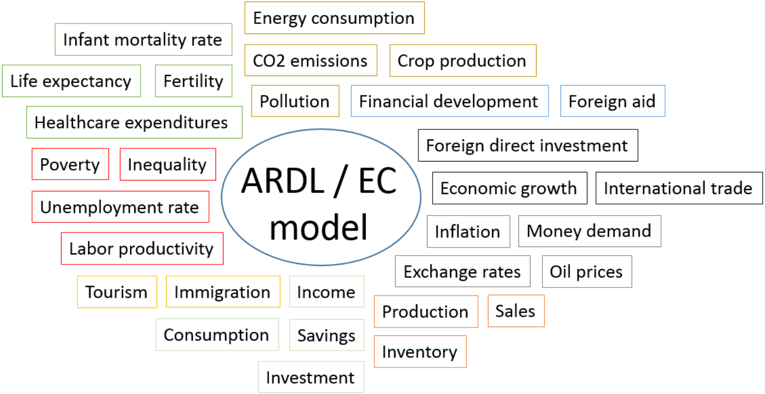
ARDL: autoregressive distributed lag model

- The autoregressive distributed lag (ARDL)[1] model is being used for decades to model the relationship between (economic) variables in a single-equation time series setup.
- Its popularity also stems from the fact that cointegration of nonstationary variables is equivalent to an error correction (EC) process, and the ARDL model has a reparameterization in EC form (Engle and Granger, 1987; Hassler and Wolters, 2006).
- The existence of a long-run / cointegrating relationship can be tested based on the EC representation. A bounds testing procedure is available to draw conclusive inference without knowing whether the variables are integrated of order zero or one, $I(0)$ or $I(1)$, respectively (Pesaran, Shin, and Smith, 2001).

[1] Another commonly used abbreviation is ADL.

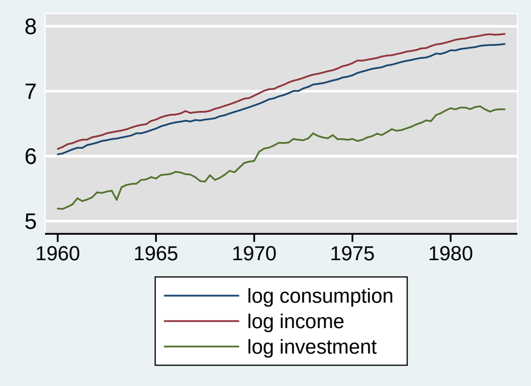
Analyzing long-run relationships

- The ARDL / EC model is useful for forecasting and to disentangle long-run relationships from short-run dynamics.



Analyzing long-run relationships

- Long-run relationship: Some time series are bound together due to equilibrium forces even though the individual time series might move considerably.



Data: National accounts, West Germany, seasonally adjusted, quarterly, billion DM, Lütkepohl (1993, Table E.1).

ARDL model

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

$$y_t = c_0 + c_1 t + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{i=0}^{q} \beta_i' \mathbf{x}_{t-i} + u_t,$$

$p \geq 1$, $q \geq 0$, for simplicity assuming that the lag order q is the same for all variables in the $K \times 1$ vector $\mathbf{x}_t$.

  - **ardl** *depvar* [*indepvars*] [*if*] [*in*] [, *options*]
- `ardl` options for the lag order selection:
  - Fixed lag order for some or all variables: **lags(*numlist*)**
  - Optimally with the Akaike information criterion: **aic**
  - Optimally with the Bayesian information criterion:[2] **bic**
  - Maximum lag order for selection criteria: **maxlags(*numlist*)**
  - Store information criteria in a matrix: **matcrit(*name*)**
  - Default: **lags(.)** **bic maxlags(4)**

[2] The BIC is also known as the Schwarz or Schwarz-Bayesian information criterion.

Reproducible example: ARDL lag specification

```
. webuse lutkepohl2
(Quarterly SA West German macro data, Bil DM, from Lutkepohl 1993 Table E.1)

. ardl ln_consump ln_inc ln_inv, lags(. . 0) aic maxlags(. 2 .) matcrit(lagcombs)

ARDL(4,1,0) regression

Sample: 1961q1 - 1982q4                         Number of obs     =         88
                                                F(   7,      80)  =   49993.34
                                                Prob > F          =     0.0000
                                                R-squared         =     0.9998
                                                Adj R-squared     =     0.9998
Log likelihood =  304.37474                     Root MSE          =     0.0080

------------------------------------------------------------------------------
  ln_consump |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
  ln_consump |
         L1. |   .4568483   .1064085     4.29   0.000     .2450887    .6686079
         L2. |   .3250994   .1127767     2.88   0.005     .1006666    .5495322
         L3. |   .1048324   .1092992     0.96   0.340      -.11268    .3223449
         L4. |  -.1632413   .0853844    -1.91   0.059    -.3331616    .0066791
             |
      ln_inc |
         --. |   .4629184    .078421     5.90   0.000     .3068557    .6189812
         L1. |   -.202756   .0965775    -2.10   0.039    -.3949513   -.0105607
             |
      ln_inv |   .0080284   .0118391     0.68   0.500    -.0155322    .0315889
       _cons |   .0373585   .0143755     2.60   0.011     .0087504    .0659667
------------------------------------------------------------------------------
```

Example (continued): Information criteria

```
. matrix list lagcombs

lagcombs[12,4]
     ln_consump      ln_inc      ln_inv         aic
 r1           1           0           0  -585.22447
 r2           1           1           0  -585.39189
 r3           1           2           0  -583.88179
 r4           2           0           0  -590.66282
 r5           2           1           0   -592.6904
 r6           2           2           0  -591.62792
 r7           3           0           0  -588.69069
 r8           3           1           0  -590.83183
 r9           3           2           0  -589.67101
r10           4           0           0  -590.03466
r11           4           1           0  -592.73282
r12           4           2           0  -592.15636

. estat ic

Akaike's information criterion and Bayesian information criterion

-----------------------------------------------------------------------------
       Model |        Obs  ll(null)  ll(model)      df         AIC        BIC
-------------+---------------------------------------------------------------
           . |         88 -64.51057   304.3747       8   -592.7495  -572.9308
-----------------------------------------------------------------------------
               Note: N=Obs used in calculating BIC; see [R] BIC note.
```

Example (continued): Fast automatic lag selection

```
. timer on 1
. ardl ln_consump ln_inc ln_inv, aic dots noheader

Optimal lag selection, % complete:
----+---20%---+---40%---+---60%---+---80%---+-100%
..................................................
AIC optimized over 100 lag combinations

------------------------------------------------------------------------------
  ln_consump |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
  ln_consump |
         L1. |   .3068554   .0958427     3.20   0.002     .1160853    .4976255
         L2. |    .325385   .0789039     4.12   0.000     .1683307    .4824393
             |
      ln_inc |   .3682844    .041534     8.87   0.000      .285613    .4509558
             |
      ln_inv |
         --. |   .0656722   .0180596     3.64   0.000     .0297255    .1016189
         L1. |  -.0375288   .0225036    -1.67   0.099    -.0823212    .0072636
         L2. |   .0228142   .0228968     1.00   0.322    -.0227607    .0683892
         L3. |  -.0129321   .0226411    -0.57   0.569    -.0579981    .0321339
         L4. |  -.0528173   .0184696    -2.86   0.005    -.0895801   -.0160544
             |
       _cons |   .0469399   .0110639     4.24   0.000     .0249178     .068962
------------------------------------------------------------------------------

. timer off 1
. timer list 1
1:      0.01 /        1 =       0.0150
```

Example (continued): Slow automatic lag selection

```
. timer on 2
. ardl ln_consump ln_inc ln_inv, aic dots noheader nofast

Optimal lag selection, % complete:
----+---20%---+---40%---+---60%---+---80%---+-100%
..................................................
AIC optimized over 100 lag combinations

------------------------------------------------------------------------------
  ln_consump |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
  ln_consump |
         L1. |   .3068554   .0958427     3.20   0.002     .1160853    .4976255
         L2. |    .325385   .0789039     4.12   0.000     .1683307    .4824393
             |
      ln_inc |   .3682844    .041534     8.87   0.000      .285613    .4509558
             |
      ln_inv |
         --. |   .0656722   .0180596     3.64   0.000     .0297255    .1016189
         L1. |  -.0375288   .0225036    -1.67   0.099    -.0823212    .0072636
         L2. |   .0228142   .0228968     1.00   0.322    -.0227607    .0683892
         L3. |  -.0129321   .0226411    -0.57   0.569    -.0579981    .0321339
         L4. |  -.0528173   .0184696    -2.86   0.005    -.0895801   -.0160544
             |
       _cons |   .0469399   .0110639     4.24   0.000     .0249178     .068962
------------------------------------------------------------------------------

. timer off 2
. timer list 2
2:       0.75 /        1 =       0.7520
```

Example (continued): Sample depends on lag selection

```
. ardl ln_consump ln_inc ln_inv, aic maxlags(8 8 4)

ARDL(2,0,4) regression

Sample: 1962q1 - 1982q4                         Number of obs     =         84
                                                F(   8,      75)  =   56976.90
                                                Prob > F          =     0.0000
                                                R-squared         =     0.9998
                                                Adj R-squared     =     0.9998
Log likelihood =   307.9708                     Root MSE          =     0.0065

------------------------------------------------------------------------------
  ln_consump |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
  ln_consump |
         L1. |     .30383   .0942165     3.22   0.002     .1161411     .491519
         L2. |   .3195318   .0776321     4.12   0.000     .1648808    .4741828
             |
      ln_inc |   .3767587   .0389267     9.68   0.000     .2992128    .4543046
             |
      ln_inv |
         --. |   .0581759   .0170736     3.41   0.001     .0241635    .0921884
         L1. |  -.0185484   .0214624    -0.86   0.390    -.0613036    .0242068
         L2. |     .01012    .021505     0.47   0.639    -.0327202    .0529602
         L3. |  -.0146641   .0213098    -0.69   0.493    -.0571154    .0277872
         L4. |  -.0488136   .0174121    -2.80   0.006    -.0835003   -.0141269
             |
       _cons |   .0416317   .0107782     3.86   0.000     .0201603     .063103
------------------------------------------------------------------------------
```

ARDL model: Optimal lag selection

- The optimal model is the one with the smallest value (most negative value) of the AIC or BIC. The BIC tends to select more parsimonious models.
- The information criteria are only comparable when the sample is held constant. This can lead to different estimates even with the same lag orders if the maximum lag order is varied.
- `ardl` uses a fast Mata-based algorithm to obtain the optimal lag order. This comes at the cost of minor numerical differences in the values of the criteria compared to `estat ic` but the ranking of the models is unaffected. The option **`nofast`** avoids this problem but it uses a substantially slower algorithm based on Stata's `regress` command.
- For very large models, it might be necessary to increase the admissible maximum number of lag combinations with the option `maxcombs(#)`.

EC representation

- Reparameterization in conditional EC form (`ardl` option `ec`):

$$\Delta y_t = c_0 + c_1 t - \alpha(y_{t-1} - \boldsymbol{\theta}\mathbf{x}_t) + \sum_{i=1}^{p-1} \psi_{yi} \Delta y_{t-i} + \sum_{i=0}^{q-1} \boldsymbol{\psi}'_{xi} \Delta \mathbf{x}_{t-i} + u_t.$$

  with the speed-of-adjustment coefficient $\alpha = 1 - \sum_{j=1}^{p} \phi_i$ and the long-run coefficients $\boldsymbol{\theta} = \frac{\sum_{j=0}^{q} \boldsymbol{\beta}_j}{\alpha}$.

- Alternative EC parameterization (`ardl` option `ec1`):

$$\Delta y_t = c_0 + c_1 t - \alpha(y_{t-1} - \boldsymbol{\theta}\mathbf{x}_{t-1}) + \sum_{i=1}^{p-1} \psi_{yi} \Delta y_{t-i} + \boldsymbol{\omega}' \Delta \mathbf{x}_t + \sum_{i=1}^{q-1} \boldsymbol{\psi}'_{xi} \Delta \mathbf{x}_{t-i} + u_t,$$

Example (continued): EC representation

```
. ardl ln_consump ln_inc ln_inv, aic ec noheader

------------------------------------------------------------------------------
D.ln_consump |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
ADJ          |
  ln_consump |
         L1. |  -.3677596   .0406085    -9.06   0.000    -.4485888   -.2869304
-------------+----------------------------------------------------------------
LR           |
      ln_inc |   1.001427   .0265233    37.76   0.000     .9486337     1.05422
      ln_inv |  -.0402213   .0309082    -1.30   0.197    -.1017424    .0212999
-------------+----------------------------------------------------------------
SR           |
  ln_consump |
         LD. |   -.325385   .0789039    -4.12   0.000    -.4824393   -.1683307
             |
      ln_inv |
         D1. |    .080464   .0187106     4.30   0.000     .0432214    .1177066
         LD. |   .0429352   .0193931     2.21   0.030     .0043342    .0815361
        L2D. |   .0657494   .0181592     3.62   0.001     .0296045    .1018943
        L3D. |   .0528173   .0184696     2.86   0.005     .0160544    .0895801
             |
       _cons |   .0469399   .0110639     4.24   0.000     .0249178     .068962
------------------------------------------------------------------------------
```

Example (continued): Alternative EC representation

```
. ardl ln_consump ln_inc ln_inv, aic ec1 noheader

------------------------------------------------------------------------------
D.ln_consump |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
ADJ          |
  ln_consump |
         L1. |  -.3677596   .0406085    -9.06   0.000    -.4485888   -.2869304
-------------+----------------------------------------------------------------
LR           |
      ln_inc |
         L1. |   1.001427   .0265233    37.76   0.000     .9486337     1.05422
             |
      ln_inv |
         L1. |  -.0402213   .0309082    -1.30   0.197    -.1017424    .0212999
-------------+----------------------------------------------------------------
SR           |
  ln_consump |
         LD. |   -.325385   .0789039    -4.12   0.000    -.4824393   -.1683307
             |
      ln_inc |
         D1. |   .3682844    .041534     8.87   0.000      .285613    .4509558
             |
      ln_inv |
         D1. |   .0656722   .0180596     3.64   0.000     .0297255    .1016189
         LD. |   .0429352   .0193931     2.21   0.030     .0043342    .0815361
        L2D. |   .0657494   .0181592     3.62   0.001     .0296045    .1018943
        L3D. |   .0528173   .0184696     2.86   0.005     .0160544    .0895801
             |
       _cons |   .0469399   .0110639     4.24   0.000     .0249178     .068962
------------------------------------------------------------------------------
```

Example (continued): Attaching exogenous variables

```
. ardl ln_consump ln_inc, exog(L(0/3)D.ln_inv) aic ec noheader

------------------------------------------------------------------------------
D.ln_consump |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
ADJ          |
  ln_consump |
         L1. |  -.3788728   .0420886    -9.00   0.000    -.4626481   -.2950975
-------------+----------------------------------------------------------------
LR           |
      ln_inc |   .9669152   .0039557   244.44   0.000     .9590416    .9747889
-------------+----------------------------------------------------------------
SR           |
  ln_consump |
         LD. |   -.346926   .0806726    -4.30   0.000    -.5075007   -.1863512
        L2D. |  -.1074193   .0790118    -1.36   0.178    -.2646883    .0498497
             |
      ln_inv |
         D1. |   .0758713   .0176989     4.29   0.000     .0406425    .1111002
         LD. |   .0422224   .0191523     2.20   0.030     .0041008     .080344
        L2D. |   .0678568   .0185208     3.66   0.000      .030992    .1047216
        L3D. |   .0485441   .0179609     2.70   0.008     .0127938    .0842944
             |
       _cons |   .0504873   .0114518     4.41   0.000      .027693    .0732816
------------------------------------------------------------------------------
```

EC representation: Interpretation

- The long-run coefficients $\boldsymbol{\theta}$ are reported in the output section `LR`. They represent the equilibrium effects of the independent variables on the dependent variable. In the presence of cointegration, they correspond to the negative cointegration coefficients after normalizing the coefficient of the dependent variable to unity. The latter is not explicitly displayed.
- The negative speed-of-adjustment coefficient $-\alpha$ is reported in the output section `ADJ`. It measures how strongly the dependent variable reacts to a deviation from the equilibrium relationship in one period or, in other words, how quickly such an equilibrium distortion is corrected.
- The short-run coefficients ψ_{yi}, $\boldsymbol{\psi}_{xi}$ (and $\boldsymbol{\omega}$) are reported in the output section `SR`. They account for short-run fluctuations not due to deviations from the long-run equilibrium.

EC representation: Integration order

- The independent variables are allowed to be individually $I(0)$ or $I(1)$.
- The independent variables must be long-run forcing (weakly exogenous) for the dependent variable, i.e. there can be at most one cointegrating relationship involving the dependent variable. (There might be further cointegrating relationships among the independent variables themselves.)
- By default, each independent variable is included in the long-run relationship. $I(0)$ variables that shall only affect the short-run dynamics can be specified with the option `exog(`*varlist*`)`. An automatic lag selection or first-difference transformation is not performed for the latter.

Testing the existence of a long-run relationship

- Pesaran, Shin, and Smith (2001) bounds test:
  1. Use the F-statistic to test the joint null hypothesis $H_0^F : (\alpha = 0) \cap \left(\sum_{j=0}^{q} \boldsymbol{\beta}_j = \mathbf{0}\right)$ versus the alternative hypothesis $H_1^F : (\alpha \neq 0) \cup \left(\sum_{j=0}^{q} \boldsymbol{\beta}_j \neq \mathbf{0}\right)$.[3]
  2. If H_0^F is rejected, use the t-statistic to test the single hypothesis $H_0^t : \alpha = 0$ versus $H_1^t : \alpha \neq 0$.
  3. If H_1^F is rejected, use conventional z-tests (or Wald tests) to test whether the elements of $\boldsymbol{\theta}$ are individually (or jointly) statistically significantly different from zero.
- There is statistical evidence for the existence of a long-run / cointegrating relationship if the null hypothesis is rejected in all three steps.

[3] The test is not directly performed on the long-run coefficients $\boldsymbol{\theta} = \left(\sum_{j=0}^{q} \boldsymbol{\beta}_j\right) / \alpha$.

Testing the existence of a long-run relationship

- The distributions of the test statistics in steps 1 and 2 are nonstandard and depend on the integration order of the independent variables.
- Kripfganz and Schneider (2018) use response surface regressions to obtain finite-sample and asymptotic critical values, as well as approximate p-values, for the lower and upper bound of all independent variables being purely $I(0)$ or purely $I(1)$ (and not mutually cointegrated), respectively.
- These critical values supersede the near-asymptotic critical values provided by Pesaran, Shin, and Smith (2001) and the finite-sample critical values by Narayan (2005), among others.

Testing the existence of a long-run relationship

- The critical values depend on the number of independent variables, their integration order, the number of short-run coefficients,[4] and the inclusion of an intercept or time trend.
- `ardl` options for the deterministic model components:
  1. No intercept, no trend: **noconstant**
  2. Restricted intercept, no trend: **restricted**
  3. Unrestricted intercept, no trend: the default
  4. Unrestricted intercept, restricted trend: **trend(*varname*)** and **restricted**
  5. Unrestricted intercept, unrestricted trend: **trend(*varname*)**

[4] The number of short-run coefficients only affects the finite-sample but not the asymptotic critical values (Cheung and Lai, 1995; Kripfganz and Schneider, 2018). The elements of $\boldsymbol{\omega}$ in the `ec1` parameterization for variables that have 0 lags in the ARDL model do not count towards this number.

Testing the existence of a long-run relationship

- Test decisions:
  - Do not reject H_0^F or H_0^t, respectively, if the test statistic is closer to zero than the lower bound of the critical values.
  - Reject the H_0^F or H_0^t, respectively, if the test statistic is more extreme than the upper bound of the critical values.
- The first two steps of the bounds test are implemented in the `ardl` postestimation command **`estat ectest`**.
  - By default, finite-sample critical values for the 1%, 5%, and 10% significance levels are provided. Asymptotic critical values are displayed with option **`asymptotic`**. Alternative significance levels can be specified with option **`siglevels(`***`numlist`***`)`**.
- The test statistics in step 3 have the usual asymptotic standard normal (or χ^2) distributions irrespective of the integration order of the independent variables.[5]

[5] The OLS estimator for the long-run coefficients $\boldsymbol{\theta}$ of $I(1)$ independent variables is "super-consistent" with convergence rate T instead of $\sqrt{T}$ (Pesaran and Shin, 1998; Hassler and Wolters, 2006).

Example (continued): Bounds test

```
. estat ectest

Pesaran, Shin, and Smith (2001) bounds test

H0: no level relationship                                  F =     40.952
Case 3                                                     t =     -9.002

Finite sample (1 variables, 88 observations, 6 short-run coefficients)

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

   | 10%                 | 5%                  | 1%                  | p-value
   |    I(0)      I(1)   |    I(0)      I(1)   |    I(0)      I(1)   |    I(0)      I(1)
---+---------------------+---------------------+---------------------+-------------------
 F |   4.032     4.831   |   4.958     5.843   |   7.070     8.119   |   0.000     0.000
 t |  -2.550    -2.899   |  -2.861    -3.225   |  -3.470    -3.854   |   0.000     0.000

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)
```

Example (continued): EC model with restricted trend

```
. ardl ln_consump ln_inc, exog(L(0/3)D.ln_inv) trend(qtr) aic ec restricted noheader

------------------------------------------------------------------------------
D.ln_consump |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
ADJ          |
  ln_consump |
         L1. |   -.341178   .0431316    -7.91   0.000    -.4270464   -.2553096
-------------+----------------------------------------------------------------
LR           |
      ln_inc |    1.14358   .0782318    14.62   0.000     .9878321    1.299327
         qtr |  -.0036516   .0016171    -2.26   0.027     -.006871   -.0004322
-------------+----------------------------------------------------------------
SR           |
  ln_consump |
         LD. |  -.4362663      .0851    -5.13   0.000    -.6056874   -.2668452
        L2D. |  -.1899566   .0825977    -2.30   0.024     -.354396   -.0255172
             |
      ln_inv |
         D1. |   .0842961   .0173889     4.85   0.000     .0496775    .1189146
         LD. |   .0517241   .0188448     2.74   0.008     .0142069    .0892412
        L2D. |   .0726232    .017972     4.04   0.000     .0368437    .1084027
        L3D. |   .0482872   .0173383     2.79   0.007     .0137693    .0828051
             |
       _cons |  -.3188651   .1422961    -2.24   0.028     -.602155   -.0355753
------------------------------------------------------------------------------
```

Example (continued): Bounds test with restricted trend

```
. estat ectest

Pesaran, Shin, and Smith (2001) bounds test

H0: no level relationship                                     F =     31.557
Case 4                                                        t =     -7.910

Finite sample (1 variables, 88 observations, 6 short-run coefficients)

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

   | 10%               | 5%                | 1%                | p-value
   |    I(0)     I(1)  |    I(0)     I(1)  |    I(0)     I(1)  |    I(0)     I(1)
---+-------------------+-------------------+-------------------+------------------
 F |   4.066    4.582  |   4.784    5.351  |   6.396    7.057  |   0.000    0.000
 t |  -3.107   -3.384  |  -3.412   -3.704  |  -4.014   -4.327  |   0.000    0.000

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)
```

Further information on the bounds test

- The validity of the bounds test relies on normally distributed error terms that are homoskedastic and serially uncorrelated, as well as stability of the coefficients over time.
- If in doubt about remaining serial error correlation, increase the lag order for testing purposes (e.g. use the AIC instead of the BIC to obtain the optimal lag order).
- A more parsimonious model for interpretation and forecasting purposes can be estimated after the testing procedure.
    - If the bounds test does not reject the null hypothesis of no long-run relationship, an ARDL model purely in first differences (without an equilibrium correction term) might be estimated.

Postestimation commands

- Besides **estat ectest**, the **ardl** command supports standard Stata postestimation commands such as **estat ic**, **estimates**, **lincom**, **nlcom**, **test**, **testnl**, and **lrtest**.
- **predict** allows to obtain fitted values (option **xb**) and residuals (option **residuals**) in the usual way. In addition, the option **ec** generates the equilibrium correction term:
  - $\widehat{ec}_t = y_{t-1} - \hat{\boldsymbol{\theta}} \mathbf{x}_t$ after ardl, ec
  - $\widehat{ec}_t = y_{t-1} - \hat{\boldsymbol{\theta}} \mathbf{x}_{t-1}$ after ardl, ec1
- The diagnostic commands **sktest**, **qnorm**, and **pnorm** are helpful as well to detect nonnormality of the residuals.

Postestimation commands

- The final `ardl` estimation results are internally obtained with the `regress` command. These underlying `regress` estimates can be stored with the `ardl` option `regstore(`*name*`)` and restored with `estimates restore` *name*.
- Subsequently, all the familiar `regress` postestimation commands are available, in particular:
  - `estat hettest` and `estat imtest` for heteroskedasticity and normality testing,
  - `estat bgodfrey` and `estat durbinalt` for serial-correlation testing,[6]
  - `estat sbcusum`, `estat sbknown`, and `estat sbsingle` for structural-breaks testing.

[6] `estat dwatson` is not valid for ARDL / EC models because the lagged dependent variable is not strictly exogenous by construction.

Example (continued): Serial-correlation testing

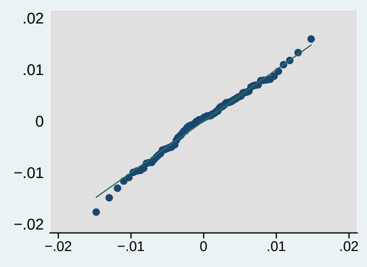
```
. quietly ardl ln_consump ln_inc, exog(L(0/3)D.ln_inv) trend(qtr) aic ec regstore(ardlreg)
. estimates restore ardlreg
(results ardlreg are active now)

. estat bgodfrey, lags(1/4) small

Breusch-Godfrey LM test for autocorrelation
---------------------------------------------------------------------------
    lags(p)  |          F                   df                 Prob > F
-------------+-------------------------------------------------------------
       1     |        0.116             (  1,    77 )            0.7341
       2     |        0.068             (  2,    76 )            0.9340
       3     |        0.364             (  3,    75 )            0.7791
       4     |        0.453             (  4,    74 )            0.7702
---------------------------------------------------------------------------
                        H0: no serial correlation

. estat durbinalt, lags(1/4) small

Durbin's alternative test for autocorrelation
---------------------------------------------------------------------------
    lags(p)  |          F                   df                 Prob > F
-------------+-------------------------------------------------------------
       1     |        0.102             (  1,    77 )            0.7505
       2     |        0.059             (  2,    76 )            0.9426
       3     |        0.314             (  3,    75 )            0.8150
       4     |        0.389             (  4,    74 )            0.8162
---------------------------------------------------------------------------
                        H0: no serial correlation
```

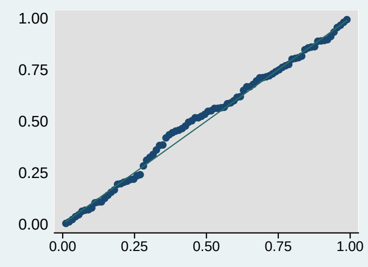
Example (continued): Heteroskedasticity testing

```
. estat hettest

Breusch-Pagan / Cook-Weisberg test for heteroskedasticity
         Ho: Constant variance
         Variables: fitted values of D.ln_consump

         chi2(1)      =     0.26
         Prob > chi2  =   0.6067

. estat imtest, white

White's test for Ho: homoskedasticity
         against Ha: unrestricted heteroskedasticity

         chi2(54)     =     52.03
         Prob > chi2  =    0.5508

Cameron & Trivedi's decomposition of IM-test

---------------------------------------------------
              Source |       chi2     df      p
---------------------+-----------------------------
  Heteroskedasticity |      52.03     54    0.5508
            Skewness |      12.24      9    0.2000
            Kurtosis |       0.02      1    0.8967
---------------------+-----------------------------
               Total |      64.29     64    0.4664
---------------------------------------------------
```

Example (continued): Normality testing

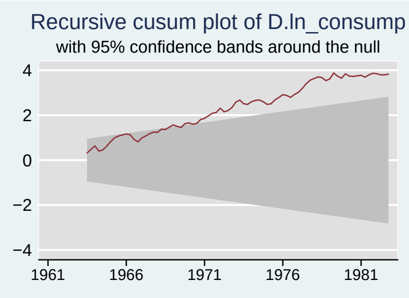
```
. predict resid, residuals
(4 missing values generated)

. sktest resid

                    Skewness/Kurtosis tests for Normality
                                                        ------ joint ------
    Variable |        Obs  Pr(Skewness)  Pr(Kurtosis) adj chi2(2)   Prob>chi2
-------------+---------------------------------------------------------------
       resid |         88     0.3270        0.8107        1.04         0.5939

. qnorm resid
. pnorm resid
```





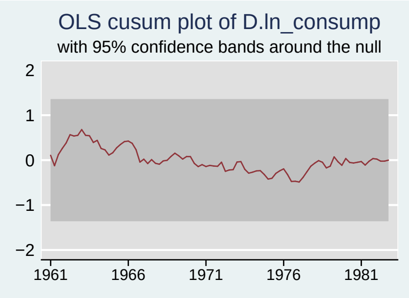
Example (continued): Structural-breaks testing

```
. estat sbcusum

Cumulative sum test for parameter stability

Sample: 1961q1 - 1982q4                              Number of obs =        88
Ho: No structural break

                                  1% Critical      5% Critical      10% Critical
 Statistic    Test Statistic         Value            Value             Value
------------------------------------------------------------------------------
 recursive        1.4690            1.1430           0.9479            0.850
------------------------------------------------------------------------------
```



Example (continued): Structural-breaks testing

```
. estat sbcusum, ols

Cumulative sum test for parameter stability

Sample: 1961q1 - 1982q4                          Number of obs =        88
Ho: No structural break

                                1% Critical      5% Critical     10% Critical
 Statistic   Test Statistic       Value            Value            Value
------------------------------------------------------------------------------
 ols             0.6793           1.6276           1.3581           1.224
------------------------------------------------------------------------------
```



Example (continued): Structural-breaks testing

```
. estat sbsingle, all
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
..................................................    50
..........

Test for a structural break: Unknown break date

                             Number of obs =        88

Full sample:                 1961q1 - 1982q4
Trimmed sample:              1964q3 - 1979q3
Ho: No structural break

     Test            Statistic          p-value
-----------------------------------------------
     swald            20.1088           0.3040
     awald            13.9245           0.1019
     ewald             7.9897           0.1939
       slr            22.7977           0.1605
       alr            16.3306           0.0330
       elr             9.3047           0.0886
-----------------------------------------------
Exogenous variables:          L.ln_consump ln_inc LD.ln_consump L2D.ln_consump D.ln_inv LD.ln_inv
                                  L2D.ln_inv L3D.ln_inv qtr
Coefficients included in test: L.ln_consump ln_inc LD.ln_consump L2D.ln_consump D.ln_inv LD.ln_inv
                                  L2D.ln_inv L3D.ln_inv qtr _cons
```

Example (continued): Structural-breaks testing

```
. estat sbsingle, breakvars(L.ln_consump ln_inc) all
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
..................................................    50
..........

Test for a structural break: Unknown break date

                             Number of obs =        88

Full sample:                 1961q1 - 1982q4
Trimmed sample:              1964q3 - 1979q3
Ho: No structural break

     Test            Statistic         p-value
-----------------------------------------------
     swald             8.9039          0.1457
     awald             2.5060          0.2608
     ewald             2.0321          0.1738
       slr             9.7492          0.1046
       alr             2.8269          0.2027
       elr             2.3571          0.1225
-----------------------------------------------
Exogenous variables:           L.ln_consump ln_inc LD.ln_consump L2D.ln_consump D.ln_inv LD.ln_inv
                                   L2D.ln_inv L3D.ln_inv qtr
Coefficients included in test: L.ln_consump ln_inc
```

Note: This is a test for a structural break in the speed-of-adjustment and long-run coefficients.

Further topics

- The ardl command can estimate autoregressive models without independent variables. In this case, the bounds test collapses to the familiar augmented Dickey-Fuller unit root test. The Kripfganz and Schneider (2018) critical values cover this special case, too.
- The **forecast** command suite can be used for model forecasting after ardl.
- ardl does not compute robust standard errors. Yet, once the optimal lag order is obtained, the final model can be reestimated with the **newey** command to obtain Newey-West standard errors.

Example (continued): Augmented Dickey-Fuller regression

```
. ardl dln_inv, aic ec restricted

ARDL(4) regression

Sample: 1961q2 - 1982q4                         Number of obs     =         87
                                                R-squared         =     0.6462
                                                Adj R-squared     =     0.6289
Log likelihood =  154.12285                     Root MSE          =     0.0424

------------------------------------------------------------------------------
   D.dln_inv |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
ADJ          |
     dln_inv |
         L1. |   -.755277   .2295731    -3.29   0.001    -1.211971   -.2985831
-------------+----------------------------------------------------------------
LR           |
       _cons |    .015006   .0060544     2.48   0.015     .0029618    .0270501
-------------+----------------------------------------------------------------
SR           |
     dln_inv |
         LD. |  -.4633003   .2005284    -2.31   0.023    -.8622152   -.0643855
        L2D. |  -.4938993   .1577325    -3.13   0.002    -.8076796    -.180119
        L3D. |  -.3133117   .1029967    -3.04   0.003    -.5182049   -.1084184
------------------------------------------------------------------------------
```

Note: The aim is to test whether `dln_inv`, the first difference of `ln_inv`, is nonstationary.

Example (continued): Augmented Dickey-Fuller test

```
. estat ectest

Pesaran, Shin, and Smith (2001) bounds test

H0: no level relationship                                  F =     5.478
Case 2                                                     t =    -3.290

Finite sample (0 variables, 87 observations, 3 short-run coefficients)

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

   | 10%                | 5%                 | 1%                 | p-value
   |     I(0)      I(1) |     I(0)      I(1) |     I(0)      I(1) |     I(0)      I(1)
---+--------------------+--------------------+--------------------+------------------
 F |    3.823     3.812 |    4.677     4.659 |    6.644     6.601 |    0.026     0.025
 t |   -2.565    -2.569 |   -2.869    -2.874 |   -3.463    -3.472 |    0.017     0.017

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)
```

Note: The null hypothesis is that dln_inv follows a unit root process (without drift).

Example (continued): Augmented Dickey-Fuller test

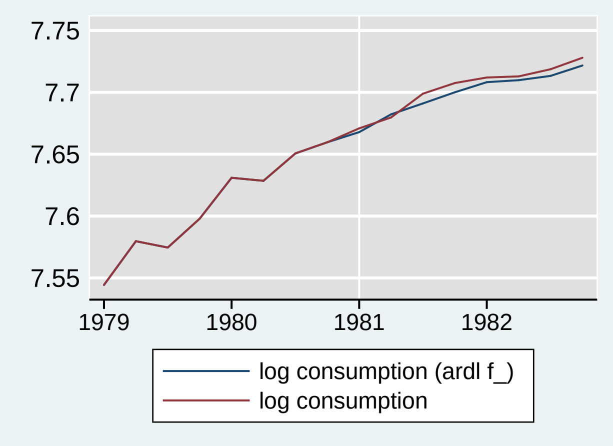
```
. dfuller dln_inv if e(sample), lags(3) regress

Augmented Dickey-Fuller test for unit root         Number of obs   =        87

                               ---------- Interpolated Dickey-Fuller ---------
                  Test         1% Critical       5% Critical      10% Critical
               Statistic           Value             Value             Value
------------------------------------------------------------------------------
 Z(t)             -3.290            -3.528            -2.900            -2.585
------------------------------------------------------------------------------
MacKinnon approximate p-value for Z(t) = 0.0153

------------------------------------------------------------------------------
   D.dln_inv |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     dln_inv |
         L1. |   -.755277   .2295731    -3.29   0.001    -1.211971   -.2985831
         LD. |  -.4633003   .2005284    -2.31   0.023    -.8622152   -.0643855
        L2D. |  -.4938993   .1577325    -3.13   0.002    -.8076796    -.180119
        L3D. |  -.3133117   .1029967    -3.04   0.003    -.5182049   -.1084184
             |
       _cons |   .0113337   .0060208     1.88   0.063    -.0006437     .023311
------------------------------------------------------------------------------
```

Example (continued): Forecasting

```
. quietly ardl ln_consump ln_inc ln_inv if qtr < tq(1981q1), trend(qtr)
. estimates store ardl
. forecast create ardl
  Forecast model ardl started.

. forecast estimates ardl, predict(xb)
  Added estimation results from ardl.
  Forecast model ardl now contains 1 endogenous variable.

. forecast exogenous ln_inc ln_inv qtr
  Forecast model ardl now contains 3 declared exogenous variables.

. forecast solve, begin(tq(1981q1))

Computing dynamic forecasts for model ardl.
-------------------------------------------
Starting period:  1981q1
Ending period:    1982q4
Forecast prefix:  f_

1981q1:  ...........
1981q2:  ...........
1981q3:  ...........
1981q4:  ...........
1982q1:  ...........
1982q2:  ..........
1982q3:  ..........
1982q4:  ...........

Forecast 1 variable spanning 8 periods.
----------------------------------------
```

Example (continued): Forecast versus actual data

```
. twoway (tsline f_ln_consump if qtr>=tq(1979q1)) (tsline ln_consump if qtr>=tq(1979q1)), tline(1981q1)
```



Note: The forecast period (1981q1 – 1982q4) is excluded from the estimation period (1961q1 – 1980q4).

Example (continued): Newey-West standard errors

```
. quietly ardl ln_consump ln_inc, exog(L(0/3)D.ln_inv) trend(qtr) aic regstore(ardlreg)
. quietly estimates restore ardlreg
. local cmdline ‘"‘e(cmdline)’"’
. gettoken cmd cmdline : cmdline
. newey ‘cmdline’ lag(4)

Regression with Newey-West standard errors      Number of obs     =         88
maximum lag: 4                                  F(  9,        78) =   62645.21
                                                Prob > F          =     0.0000

------------------------------------------------------------------------------
             |             Newey-West
  ln_consump |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
  ln_consump |
         L1. |   .2225557   .0931767     2.39   0.019     .0370552    .4080562
         L2. |   .2463097   .1003579     2.45   0.016     .0465125    .4461068
         L3. |   .1899566   .1013927     1.87   0.065    -.0119008    .3918141
             |
      ln_inc |   .3901642   .0400174     9.75   0.000     .3104956    .4698327
             |
      ln_inv |
         D1. |   .0842961   .0258047     3.27   0.002     .0329229    .1356693
         LD. |   .0517241   .0158053     3.27   0.002     .0202582      .08319
        L2D. |   .0726232   .0156803     4.63   0.000     .0414061    .1038404
        L3D. |   .0482872    .017342     2.78   0.007      .013762    .0828124
             |
         qtr |  -.0012458    .000383    -3.25   0.002    -.0020083   -.0004833
       _cons |  -.3188651   .1104624    -2.89   0.005    -.5387789   -.0989513
------------------------------------------------------------------------------
```

Example (continued): Long-run coefficient

```
. nlcom _b[ln_inc] / (1 - _b[L.ln_consump] - _b[L2.ln_consump] - _b[L3.ln_consump])

       _nl_1:  _b[ln_inc] / (1 - _b[L.ln_consump] - _b[L2.ln_consump] - _b[L3.ln_consump])

------------------------------------------------------------------------------
  ln_consump |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _nl_1 |    1.14358   .0691576    16.54   0.000     1.008033    1.279126
------------------------------------------------------------------------------
```

Note: This is the same long-run coefficient as earlier but with Newey-West standard errors.

Summary: The ardl package for Stata

- The `ardl` command estimates an ARDL model with optimal or prespecified lag orders, possibly reparameterized in EC form.
- The bounds test for the existence of a long-run / cointegrating relationship is implemented as the postestimation command `estat ectest`.
  - Asymptotic and finite-sample critical value bounds are available (Kripfganz and Schneider, 2018).
  - The augmented Dickey-Fuller unit root test is a special case in the absence of independent variables.
- The usual `regress` postestimation commands can be applied.

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

help ardl
help ardl postestimation
```

References

- Cheung, Y.-W., and K. S. Lai (1995). Lag order and critical values of the augmented Dickey-Fuller test. *Journal of Business & Economic Statistics 13(3)*: 277–280.
- Engle, R. F., and C. W. J. Granger (1987). Co-integration and error correction: representation, estimation, and testing. *Econometrica 55(2)*: 251–276.
- Hassler, U., and J. Wolters (2006). Autoregressive distributed lag models and cointegration. *Allgemeines Statistisches Archiv 90(1)*: 59–74.
- Kripfganz, S., and D. C. Schneider (2018). Response surface regressions for critical value bounds and approximate p-values in equilibrium correction models. *Manuscript*, University of Exeter and Max Planck Institute for Demographic Research, www.kripfganz.de.
- Lütkepohl, H. (1993). Introduction to Multiple Time Series Analysis (2^{nd} edition), Berlin, New York: Springer.
- Narayan, P. K (2005). The saving and investment nexus for China: evidence from cointegration tests. *Applied Economics 37(17)*: 1979–1990.
- Pesaran, M. H., and Y. Shin (1998). An autoregressive distributed-lag modelling approach to cointegration analysis. In *Econometrics and Economic Theory in the 20th Century. The Ragnar Frisch Centennial Symposium*, ed. S. Strøm, chap. 11, 371–413. Cambridge: Cambridge University Press.
- Pesaran, M. H., Y. Shin, and R. Smith (2001). Bounds testing approaches to the analysis of level relationships. *Journal of Applied Econometrics 16(3)*: 289–326.