COINT 2.1 Manual

Sam Ouliaris and Peter C. B. Phillips

November 2016

COINT Source Files: UNIT.SRC, CREGRS.SRC, BASE.SRC, KERNELS.SRC

Consider the linear regression model:

$$y_t = \beta' x_t + \varepsilon_t \tag{1}$$

where

$$x_t = x_{t-1} + v_t \tag{2}$$

and ε_t and v_t are stationary random variates.

Equation (1) is a cointegrated regression equation. **COINT** contains routines for testing the null hypothesis that y and X possess unit roots (i.e., follow (2)) and that ε is a stationary process. It also has routines which estimate the cointegrating vector β. Routines for testing linear hypotheses on β using Gaussian and chi-squared asymptotics are also provided.

In particular, the library contains routines for:

1. Testing the unit root hypothesis, including the:
   - (a) Phillips (1987) Z_α and Z_t statistics;
   - (b) Park–Choi (1988) $G(p, q)$ and $J(p, q)$ statistics;
   - (c) Said–Dickey (1984) Augmented Dickey–Fuller (ADF) statistic.
2. Testing the cointegration hypothesis (i.e., ε is stationary), including the:
   - (a) Phillips (1987) Z_α and Z_t statistics;
   - (b) Park (1992) $H(p, q)$ statistic;
   - (c) Said–Dickey (1984) Augmented Dickey–Fuller statistic;
   - (d) Stock–Watson (1988) common trends statistic;
   - (e) Johansen (1988) trace statistic;
   - (f) Phillips–Ouliaris (1990) P_u and P_z statistics.
3. Estimating β, the cointegrating vector:
   - (a) Park (1992) Canonical Cointegrating Regression or "CCR" procedure;
   - (b) Phillips–Hansen (1990) "fully modified" or "FM" procedure;
   - (c) Phillips (1990) spectral regression procedure;
   - (d) Phillips (1990) GIVE spectral regression procedure;
   - (e) Bandwidth versions of (3) and (4);
   - (f) Johansen (1988) "reduced rank regression" approach (i.e., maximum likelihood);
   - (g) Saikkonen (1991) asymptotically efficient least squares estimator.

4. Structural stability tests:

   (a) Hansen (1991) Lc, MeanF, and SupF statistics for testing the null hypothesis of no structural change in β.

Most of the procedures in **COINT** allow for deterministic polynomial trends (of an arbitrary order) in the fitted regression. They also support automatic bandwidth selection with autoregressive pre-filtering.

COINT Source File: ARMA.SRC

Consider the ARMA(p, q) model:

$$a(L)y(t) = b(L)e(t) \tag{3}$$

where $e(t)$ is $iid(0,\, \sigma^2)$ and $a(L)$ and $b(L)$ are polynomials in the lag operator L of degree p and q respectively. The polynomials $a(L)$ and $b(L)$ have no common factors and have zeros outside the unit circle in the complex plane. The system (3) is therefore stable and irreducible. The time series $y(t)$ is stationary.

ARMA.SRC provides procedures for estimating the coefficients of (3) and determining the orders of the polynomials $a(\cdot)$ and $b(\cdot)$. Graphics procedures which show surfaces of the model selection criteria are also included. These can be used to assess how well determined the selected orders of $a(\cdot)$ and $b(\cdot)$ are. Some illustrations of the use of these programs and graphs are given in Phillips (1994) and Phillips and Ploberger (1994).

In particular **ARMA.SRC** contains routines for:

1. Finding the order of an autoregression and the degree of a deterministic trend (with graphics) using the:
   (a) Akaike (1969) AIC criterion
   (b) Schwarz (1978) BIC criterion
   (c) Phillips–Ploberger (1994) PIC criterion
2. Finding the order of an autoregression, the degree of a deterministic trend and testing for the presence of a unit root (with graphics):
   (a) Phillips–Ploberger (1994) PIC criterion
3. Finding the lag orders of an ARMA process with a deterministic trend, estimating the ARMA coefficients by recursive least squares and testing for the presence of a unit root (with graphics) using
   (a) The Hannan–Rissanen (1982) estimation procedure (2-stage; asymptotically inefficient)
   (b) The Hannan–Rissanen (1982) estimation procedure (3-stage asymptotically efficient)
   (c) The Hannan–Rissanen (1982)–Kavalieris (1991) ARMA model selection procedure (based on the BIC criterion)
   (d) The Phillips–Ploberger (1994) PIC criterion.

COINT Source File: LRVAR.SRC

LRVAR.SRC provides procedures for calculating the spectral density and long-run variance of stationary time series. Some procedures are given for parametric ARMA models like (3), some procedures allow for data-driven kernel techniques and other procedures combine AR- and ARMA-based prefiltering algorithms with data driven kernel estimators. Some graphics procedures are given which allow the methods to be compared in applications.

The library contains routines for

1. Finding and graphing the spectrum of an ARMA(p, q) process.
2. Estimating and graphing the spectrum of a stationary time series by:
   (a) using the spectrum of an approximating ARMA model
   (b) data-driven kernel estimation
   (c) AR-prefiltered and recolored data-driven kernel estimation, see Andrews–Monahan (1992)
   (d) ARMA-prefiltered and recolored data-driven estimation, see Lee–Phillips (1993)
3. Estimating the long-run variance of a time series by the same methods as in (2) above
4. Finding the Phillips (1987) Z_α and Z_t unit root tests using automated estimates of the long-run variance as in (B) and (C) above.

COINT Source File: BAYES.SRC

BAYES.SRC contains procedures for the Bayesian analysis of nonstationary regressions and cointegrating regressions. All of the procedures are based on AR(p) and AR(p) + TR(pt) models where "TR(pt)" signifies a deterministic trend of degree pt (with $pt \geq -1$; here $pt = -1$ corresponds to no intercept, $pt = 0$ corresponds to a fitted intercept and $pt = 1$ to a linear trend in the regression, etc.). The procedures calculate the Bayesian posterior densities of the long-run autoregressive coefficient (i.e., the sum of the autoregressive coefficients) in these models under Jeffreys' prior, the uniform prior and the e-prior of Phillips (1991a) and Zivot & Phillips (1994). Laplace approximations are used in the calculation of the marginal posterior densities — see Phillips (1983, 1991b) and Tierney & Kadane (1986). Graphics procedures are supplied which graph several of the posterior densities on the same figure with user supplied titles and legends that specify the models, parameter settings and prior densities that are used.

In particular BAYES.SRC contains routines for

1. Nonstationary regression analysis:
   (a) Computing and graphing the marginal posterior density of the long-run AR coefficient in an autoregression with deterministic trend using the Jeffreys' prior and a uniform prior. Posterior probabilities of nonstationarity and near nonstationarity are calculated.
   (b) Cointegrating regression residual analysis:
   (c) Computing and graphing the marginal posterior density of the long-run AR coefficient in an autoregression fitted to the residuals of a cointegrating regression. The procedures use Jeffreys' prior, a uniform prior and the e-prior of Phillips (1991a) and Zivot & Phillips (1994). Posterior probabilities on nonstationarity and near nonstationarity are calculated. The are illustrated in Phillips (1992).

References

Akaike, H. (1969) "Fitting Autoregressive Models for Prediction," *Annals of the Institute of Statistical Mathematics*, **21**: 243–247.

Andrews, D. W. K. (1991) "Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimation," *Econometrica*, **59**: 817–858.

Andrews, D. W. K. & C. Monahan (1991) "An Improved Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimator," *Econometrica*, **60**: 953–966.

Brillinger, David R. (1981) *Time Series Data Analysis and Theory.* San Francisco: Holden–Day.

Corbae, P. D., S. Ouliaris & P. C. B. Phillips (1994) "A Reexamination of the Consumption Function Using Frequency Domain Regressions," *Journal of Empirical Economics,* **19**: 595–609.

Hannan, E. J. & J. Rissanen (1982) "Recursive Estimation of Mixed Autoregressive-Moving Average Order," *Biometrika*, **69**: 81–94.

Hansen, B. E. (1991) "Tests for Parameter Instability in Regressions with I(1) Processes," working paper, University of Rochester.

Johansen, S. J. (1988) "Statistical Analysis of Cointegration Vectors," *Journal of Economic Dynamics and Control*, **12**: 231–254.

Johansen, S. J. (1991) "Estimation and Hypothesis Testing of Cointegration Vectors in Gaussian Vector Autoregressive Models," *Econometrica,* **59**: 1551–1580.

Johansen, S. J. & K. Juselius (1990) "Maximum Likelihood Estimation and Inference on Cointegration — With Applications to the Demand for Money," *Oxford Bulletin of Economics and Statistics,* **52**: 169–210.

Kavalieris, L. (1991) "A Note on Estimating Autoregressive-Moving Average Order," Biometrika, **78**: 920–922.

Kitamura, Y. & P. C. B. Phillips (1997) "Fully Modified IV, GIVE and GMM Estimation with Possibly Nonstationary Regressors and Instruments," *Journal of Econometrics*, **80**: 85–123.

Lee, C. C. & P. C. B. Phillips (1993) "An ARMA-Prefiltered Estimator of the Long Run Variance with an Application to Unit Root Tests," mimeo, Cowles Foundation, Yale University.

Ouliaris, S., J. Y. Park & P. C. B. Phillips (1989) "Testing for a Unit Root in the Presence of a Maintained Trend," Ch. 1 in Baldev Raj (ed.), *Advances in Econometrics and Modelling.* Netherlands: Kluwer Academic Publishers.

Park, J. Y. (1992) "Canonical Cointegrating Regression," *Econometrica*, **60**: 119–144.

Park, J. Y. & B. Choi (1988) "A New Approach to Testing for a Unit Root," Working Paper #88–23, Department of Economics, Cornell University.

Phillips, P. C. B. (1983) "Marginal Densities of Instrumental Variables Estimators in the General Single Equation Case," *Advances in Econometrics*, **2**: 1–24.

Phillips, P. C. B. (1987) "Time Series Regression with a Unit Root," *Econometrica*, **55**: 277–301.

Phillips, P. C. B. (1990) "Spectral Regression for Cointegrated Time Series," in W. Barnett (ed.), *Nonparametric and Semiparametric Methods in Economics and Statistics*. Cambridge: Cambridge University Press.

Phillips, P. C. B. (1991a) "Bayesian Routes and Unit Roots: de Rebus Prioribus Semper est Disputandum," *Journal of Applied Econometrics*, **6**(4): 435–474.

Phillips, P. C. B. (1991b) "To Criticize the Critics: An Objective Bayesian Analysis of Stochastic Trends," *Journal of Applied Econometrics*, **6**(4): 333–364.

Phillips, P. C. B. (1992) "Long-Run Australian Consumption Function Reexamined: An Empirical Exercise in Bayesian Inference," Ch. 11, in C. J. Hargreaves, *Macroeconomic Modelling of the Long Run*. Aldershot: Edward Elgar, pp. 287–322.

Phillips, P. C. B. (1995a) "Forward Exchange Market Unbiasedness: The Case of the Australian Dollar Since 1984," *Asia–Pacific Economic Review*.

Phillips, P. C. B (1995b) "Fully Modified Least Squares and Vector Autoregression," *Econometrica*,

Phillips, P. C. B. & B. E. Hansen (1990) "Statistical Inference in Instrumental Variables Regression with I(1) Processes," *Review of Economic Studies*, **57**: 99–125.

Phillips, P. C. B. & M. Loretan (1991) "Estimating Long Run Economic Equilibrium," *Review of Economic Studies,* **58**: 407–436.

Phillips, P. C. B. & S. Ouliaris (1990) "Asymptotic Properties of Residual Based Tests for Cointegration," *Econometrica*, **58**: 165–193.

Phillips, P. C. B. & W. Ploberger (1994) "Posterior Odds Testing for a Unit Root with Data Based Model Selection," *Econometric Theory*, **10**(3/4): 774–808.

Phillips, P. C. B. & E. Zivot (1994) "A Bayesian Analysis of Trend Determination in Economic Time Series" *Econometric Reviews* **13**(5): 291–336.

Said, S. E. & D. A. Dickey (1984) "Testing for Unit Roots in Autoregressive Moving Average Model of Unknown Order," *Biometrika*, **71**: 599–607.

Saikkonen, P. (1991) "Asymptotically Efficient Estimation of Cointegrating Regressions," *Econometric Theory*, **7**: 1–21.

Schwarz, G. (1978) "Estimating the Dimension of a Model," *Annals of Statistics*, **6**: 461–464.

Stock, J. & M. K. Watson (1988) "Testing for Common Trends," *Journal of the American Statistical Association*, **83**: 1097–1107.

Stock, J. & M. K. Watson (1993) "A Simple MLE of Cointegrating Vectors in Higher Order Integrated Systems," *Econometrica*, **61**: 783–820.

Tierney, L. & J. B. Kadane (1986) "Accurate Approximations for Posterior Moments and Marginal Densities," *Journal of the American Statistical Association*, **81**: 82–86.

Zivot, E. & P. C. B. Phillips (1994) "A Bayesian Analysis of Trend Determination in Economic Time Series," *Econometric Reviews*, **13**(5): 291–336.

COINT 2.1 Installation Instructions — GAUSS 3.0(+) Required

The following instructions assume that you have installed **GAUSS** Version 3.0 (or higher) in `C:\GAUSS`, that you keep your `*.SRC` files in `C:\GAUSS\SRC` and your `*.LIB` files in `C:\GAUSS\LIB`. They also assume that the **COINT** distribution disk is in drive A. Please adjust the steps accordingly if you are using different subdirectory names/source drive. Steps 1–3 below will enable **GAUSS** to load the procedures automatically as they are called.

IMPORTANT: DO NOT FOLLOW THESE STEPS IF YOU ARE USING VERSION 2.2 (or earlier) OF GAUSS, OR YOUR VERSION OF GAUSS DOES NOT SUPPORT COMPLEX ARITHMETIC. YOU MUST HAVE GAUSS 3.0 OR HIGHER TO USE COINT 2.0.

1. Copy `A:\LCG\*.SRC to C:\GAUSS\SRC`.
2. Copy `A:\LCG\COINT.LCG to C:\GAUSS\LIB`.
3. Edit your STARTUP file (usually in `C:\GAUSS`) and include the line:

   ```
   LIBRARY GAUSS COINT PGRAPH;
   ```

   (i.e., add `''COINT PGRAPH''` to your existing `LIBRARY` statement).
4. **(OPTIONAL)** Copy the `A:\LCG\*.EX` and `A:\LCG.OUT` files to your hard disk. The `*.EX` files demonstrate how to call the procedures in `COINT.LIB`. The `*.OUT` files contain the output you should get from the `*.EX` files.

UNIT.SRC: Unit Root Procedures

ADF(y, p, l)

PURPOSE

- Computes the Augmented Dickey–Fuller (ADF) statistic for the null hypothesis that y has a unit root.

FORMAT

- `{alpha,tstat,c_t} = adf(y,p,l);`

INPUTS

- `y` time-series variable (nobs $\times$ 1)
- `p` order of the time polynomial in the fitted regression
- `l` number of lagged first difference terms in the fitted regression

OUTPUTS

- `alpha` estimate of the first-order autoregressive parameter
- `tstat` ADF t-statistic
- `c_t` (6$\times$1) vector containing the critical values {1%, 5%, 10%, 90%, 95%, 99%}

DECISION RULE

- Reject the null hypothesis of a unit root if the ADF statistic $\leq$ critical value.

REMARKS

1. Set $p = -1$ for no deterministic part in the fitted regression.
2. Set $p = 0$ for a constant term in the fitted regression.
3. Set $p = 1$ for a constant term and trend in the fitted regression.

4. Critical values are available for $p \in [-1, 5]$.
5. Set $l = 0$ to obtain the standard Dickey–Fuller statistic.
6. Illustration program: `adf.ex`.

EXAMPLE

```
{a,b,c} = adf(y,1,5);
''Autoregressive parameter ='' a;
''ADF t-statistic for y ='' b; ''1% critical value ='' c[1,1];
''5% critical value ='' c[2,1];
```

REFERENCE

Said, S. E. & D. A. Dickey (1984) "Testing for Unit Roots in Autoregressive Moving Average Model of Unknown Order," *Biometrika*, **71**: 599–607.

CZA(y, x, p, l)

PURPOSE

- Test the null hypothesis of no cointegration between y and x using Phillips' (1987) Z_α and Z_t statistics and Phillips and Ouliaris (1990) limit theory.

FORMAT

- `{alpha,xza,xzt,c_za,c_zt} = cza(y,x,p,l);`

INPUTS

- `y` time-series variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `p` order of the time polynomial in the cointegrating regressio
- `l` number of autocovariance terms for computing the spectrum at frequency zero

OUTPUTS

- `alpha` estimate of the first order autoregressive parameter
- `xza` Z_α statistic
- `xzt` Z_t statistic
- `c_za` (6×1) vector containing the critical values for Z_α $\{1\%, 5\%, 10\%, 90\%, 95\%, 99\%\}$
- `c_zt` (6×1) vector containing the critical values for Z_t $\{1\%, 5\%, 10\%, 90\%, 95\%, 99\%\}$

DECISION RULE

- Reject the null hypothesis of no cointegration if the Z statistic $\leq$ critical value.

GLOBALS

1. `ker_fun; _aband; _filter`
2. Please refer to **KERNELS.SRC** for a detailed explanation of these global constants.

REMARKS

1. Set $p = -1$ to have no deterministic term in the cointegrating regression.
2. Set $p = 0$ for a constant term.
3. Set $p = 1$ for a constant term and trend.
4. Critical values are available for $p \in [-1, 5]$.
5. Illustration program: `cza.ex`.

EXAMPLE

```
_ker_fun = &parzen; /* Select Parzen kernel (see KERNELS.SRC) */
_aband = 0; /* Automatic bandwidth selection disabled */
_filter = 1; /* Enable estimation of spectrum using AR(1) filter */
{a,b,c,d,e} = cza(y,x,1,5); ``Zt_statistic for residuals
    ='' c; ``5% critical value ='' e[2,1];
```

REFERENCE

Phillips, P. C. B. & S. Ouliaris (1990) "Asymptotic Properties of Residual Based Tests for Cointegration," *Econometrica*, **58**: 165–193.

CADF(y, x, p, l)

PURPOSE

- Tests the null hypothesis of no cointegration between y and x using the Said and Dickey (1984) Augmented Dickey–Fuller statistic.

FORMAT

- `{alpha,tstat,c_t} = cadf(y,x,p,l);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `p` order of the time polynomial in the cointegrating regression
- `l` number of lagged first difference terms in the ADF regression

OUTPUTS

- `alpha` estimate of the first-order autoregressive parameter
- `tstat` ADF t-statistic
- `c_t` (6×1) vector containing the critical values {1%, 5%, 10%, 90%, 95%, 99%}

DECISION RULE

- Reject the null hypothesis of no cointegration if the ADF statistic $\leq$ critical value.

REMARKS

1. Set $p = -1$ to have no deterministic part in the cointegrating regression.
2. Set $p = 0$ for a constant term.
3. Set $p = 1$ for a constant term and trend.
4. Critical values are available for $p \in [-1, 5]$.
5. Set $l = 0$ to obtain the standard Dickey–Fuller statistic.
6. Illustration program: `cadf.ex`.

EXAMPLE

```
{a,b,c} = cadf(y,x,1,5); ‘‘ADF t-statistic =’’ b;
‘‘5% critical     value =’’ c[2,1];
```

REFERENCES

Phillips, P. C. B. & S. Ouliaris (1990) "Asymptotic Properties of Residual Based Tests for Cointegration," *Econometrica*, **58:** 165–193.

Said, S. E. & D. A. Dickey (1984) "Testing for Unit Roots in Autoregressive Moving Average Model of Unknown Order," *Biometrika*, **71:** 599–607.

GSTAT(y, p, q, v)

PURPOSE

- Computes the Park and Choi (1988) $G(p, q)$ statistic for the null hypothesis that y is stationary around a p-th order polynomial time trend.

FORMAT

- `{g,gp} = gstat(y,p,q,v);`

INPUTS

- `y` time-series variable (nobs $\times$ 1)
- `p` order of the time polynomial in the null hypothesis
- `q` order of the time polynomial in the fitted regression
- `l` number of autocovariance terms for computing the spectrum at frequency zero

OUTPUTS

- `g` computed value of the G-statistic
- `gp` p-value of the $G(p, q)$ statistic, taken from chi-squared distribution with q–p degrees of freedom

DECISION RULE

- Reject the null hypothesis of stationarity if $g \geq$ critical value.

GLOBALS

- `ker_fun; _aband; _filter`
- See **KERNELS.SRC** for a detailed explanation of these global parameters.

REMARKS

1. Set $p = -1$ to have no deterministic part in the null hypothesis.
2. Set $p = 0$ for stationarity around a constant term.
3. Set $p = 1$ for stationarity around a constant term and trend.
4. Illustration program: `gstat.ex.`

EXAMPLE

```
_ker_fun = &parzen; /* Use the Parzen kernel */
_aband = 1; /* Automatic bandwidth enabled (= 1) */
_filter = 0; /* Do not use AR(1) filter to estimate spectrum */
{g,gp} = gstat(y,1,5,20); ''G(1,5) ='' g ''p-value ='' gp;
_ker_fun = &tukham; /* Recompute using the Tukey–Hamming kernel */
{g,gp} = gstat(y,1,5,20) ''Tukey-Hamming Kernel'';
''G(1,5) ='' g ''p-value ='' gp;
```

REFERENCE

Park, J. Y. & B. Choi (1988) "A New Approach to Testing for a Unit Root," Wo rking Paper #88–23, Department of Economics, Cornell University.

JSTAT(y, p, q)

PURPOSE

- Computes the Park and Choi (1988) $J(p,q)$ statistic for the null hypothesis that y has a unit root after allowing for a p-th order polynomial time trend.

FORMAT

- `{js,cv} = jstat;(y,p,q);`

INPUTS

- `y` time-series variable (nobs $\times$ 1)
- `p` order of the time polynomial in the null hypothesis
- `q` order of the time polynomial in the fitted regression

OUTPUTS

- `js` computed value of the $J(p,q)$-statistic
- `cv` (6×1) vector of critical values {1%, 5%, 10%, 90%, 95%, 99%}

DECISION RULE

- Reject the null hypothesis of a unit root if `js` $\leq$ critical value.

REMARKS

1. Set $p = -1$ to have no deterministic part in the null hypothesis.
2. Set $p = 0$ for nonstationarity around a constant term.
3. Set $p = 1$ for nonstationarity around a constant term and trend.
4. Critical values are available for $q - p$: $0 \leq q - p \leq 11$.
5. Illustration program: `jstat.ex`.

EXAMPLE

```
{j,cv} = jstat(y,1,5); ‘‘J(1,5) =’’ j ‘‘c-value =’’ cv[2,1];
```

REFERENCE

Park, J. Y. & B. Choi (1988) “A New Approach to Testing for a Unit Root,” Working Paper #88–23, Department of Economics, Cornell University.

ZA(y, p, l)

PURPOSE

- Computes Phillips' (1987) Z_α and Z_t statistics for the null hypothesis that y has a unit root.

FORMAT

- `{alpha,xza,xzt,c_za,c_zt} = za(y,p,l);`

INPUTS

- `y` time-series variable (nobs × 1)
- `p` order of the time polynomial in the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `alpha` alpha estimate of the first order autoregressive parameter
- `xza` Z_α statistic
- `xzt` Z_t statistic
- `c_za` (6×1) vector containing the critical values for Z_α $\{1\%, 5\%, 10\%, 90\%, 95\%, 99\%\}$
- `c_zt` (6×1) vector containing the critical values for Z_t $\{1\%, 5\%, 10\%, 90\%, 95\%, 99\%\}$

DECISION RULE

- Reject the null hypothesis of a unit root if the Z statistic $\leq$ critical value.

GLOBALS

- `_ker_fun; _aband; _filter`
- See **KERNELS.SRC** for a detailed explanation of the global constants.

REMARKS

1. Set $p = -1$ to have no deterministic part in the fitted regression.
2. Set $p = 0$ to include a constant term in the fitted regression.
3. Set $p = 1$ to include a constant term and time trend in the fitted regression.
4. Critical values are available for $p \in [-1, 5]$.
5. Illustration program: `za.ex`.

EXAMPLE

```
_ker_fun = &parzen /* Use the Parzen window */;
_aband = 0; /* Automatic bandwidth disabled */
_filter = 1; /* Use AR(1) filter to estimate the spectrum */
{a,b,c,d,e} = za(y,1,5); ''Zt_statistic for y ='' c;
''5% critical value ='' e[2,1];
```

REFERENCES

Ouliaris, S., J. Y. Park & P. C. B. Phillips (1989) "Testing for a Unit Root in the Presence of a Maintained Trend," Ch. 1 in Baldev Raj (ed.), *Advances in Econometrics and Modelling.* Netherlands: Kluwer Academic Publishers.

Phillips, P. C. B. (1987) "Time Series Regression with a Unit Root," *Econometrica*, **55:** 277–301.

SW(y, p, l)

PURPOSE

- Computes Stock and Watson (1988) common trends statistic for the null hypothesis that y is a noncointegrated system (after allowing for a p-th order polynomial time trend).

FORMAT

- `{sw_stat,c_sw} = sw(y,p,l);`

INPUTS

- `y` matrix of time-series variables (nobs $\times$ k)
- `p` order of the time polynomial in the null hypothesis
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `sw_stat` $(k \times 1)$ vector containing the SW statistics, in ascending order (maximum to minimum)
- `c_sw` $(6{\times}1)$ vector containing the critical value of the smallest SW statistic $\{1\%, 5\%, 10\%, 90\%, 95\%, 99\%\}$

DECISION RULE

- Reject the null hypothesis of a unit root if the SW statistic $\leq$ critical value.

GLOBALS

- `_ker_fun; _aband; _filter`
- See **KERNELS.SRC** for an explanation of these global constants.

REMARKS

1. Set $p = -1$ to have no deterministic part in the null hypothesis.
2. Set $p = 0$ for a constant term.
3. Set $p = 1$ for a constant term and trend.
4. **SW( )** uses the **GAUSS** routine EIGRG2(x), which sets the global variable `_eigerr`. Refer to the **GAUSS** manual for more details.
5. Critical values are available for $p \in [-1, 5]$.
6. Illustration program: `sw.ex`.

EXAMPLE

```
/* Assume y is nobs × 5 in the following example */
_ker_fun = &parzen; /* Use the Parzen kernel */
_aband = 0; /* Disable automatic bandwidth selection */
_filter = 1; /* Use an AR(1) filter to estimate the spectrum */
a = sw(y,1,5); ‘‘Stock-Watson statistic for null hypothesis of
    no cointegration:’’ a[5,1];
```

REFERENCE

Stock, J. & M. K. Watson (1988) "Testing for Common Trends," *Journal of the American Statistical Association*, **83:** 1097–1107.

PU(y, x, p, l)

PURPOSE

- Computes the P_u statistic for the null-hypothesis that y and x are not cointegrated.

FORMAT

- `{p1,p2} = pu(y,x,p,l);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `p` order of the time polynomial in the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `p2` P_u statistic
- `p1` (6$\times$1) vector of critical values for the P_u statistic $\{1\%, 5\%, 10\%, 90\%, 95\%, 99\%\}$

DECISION RULE

- Reject the null hypothesis of no cointegration if the P_u statistic is $\geq$ the critical value.

GLOBALS

- `_ker_fun; _aband; _filter`
- See **KERNELS.SRC** for a detailed explanation of the global constants.

REMARKS

1. Set $p = -1$ to have no deterministic part in the cointegrating regression.
2. Set $p = 0$ to include a constant term in the cointegrating regression.
3. Set $p = 1$ to include a constant term and trend.
4. Critical values are available for $p \in [-1, 5]$.
5. Illustration program: `pupz.ex`.

EXAMPLE

```
/* Compute P_u statistic using the Parzen kernel */
_ker_fun = &parzen; /* Use Parzen kernel */
_aband = 0; /* Automatic bandwidth selection disabled */
_filter = 1; /* Use and AR(1) filter to estimate the spectrum */
{p1,p2} = pu(y,x,0,20);
''PU statistic ='' p1;
```

```
''Critical value (@ 5% level) ='' p2[5,1];
```

REFERENCE

Phillips, P. C. B. & S. Ouliaris (1990) "Asymptotic Properties of Residual Based Tests for Cointegration," *Econometrica*, **58**: 165–193.

PZ(y, x, p, l)

PURPOSE

- Computes the P_z statistic for the null-hypothesis that y and x are not cointegrated.

FORMAT

- `{p1,p2} = pz(y,x,p,l);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `p` order of the time polynomial in the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `p1` P_z statistic
- `p2` (6×1) vector of critical values for the P_z statistic $\{1\%, 5\%, 10\%, 90\%, 95\%, 99\%\}$

DECISION RULE

- Reject the null hypothesis of no cointegration if the P_z statistic is $\geq$ the critical value.

GLOBALS

- `_ker_fun; _aband; _filter`
- See **KERNELS.SRC** for a detailed explanation of the global constants.

REMARKS

1. Set $p = -1$ to have no deterministic part in the cointegrating regression.
2. Set $p = 0$ to include a constant term in the cointegrating regression.
3. Set $p = 1$ to include a constant term and trend.
4. Critical values are available for $p \in [-1, 5]$.
5. Illustration program: `pupz.ex`.

EXAMPLE

```
_ker_fun = &parzen; /* Use Parzen window */
_aband = 1; /* Automatic bandwidth selection enabled */
_filter = 0; /* AR(1) filter disabled */
{p1,p2} = pz(y,x,0,20);
''PZ statistic ='' p1;
''Critical value (5% level) ='' p2[5,1];
```

REFERENCE

Phillips, P. C. B. & S. Ouliaris (1990) "Asymptotic Properties of Residual Based Tests for Cointegration," *Econometrica*, **58:** 165–193.

CREGRS.SRC

Regression Procedures for

Cointegrated Systems

CCR(y, x, d, l)

PURPOSE

- Computes Park's (1992) Canonical Cointegrating Regression estimator for cointegrated regression models, using OLS for the first stage regression.

FORMAT

- `{beta,vc,stderr,sigma,tstats,rss,resid,dummy} = ccr(y,x,d,l);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `d` deterministic part of the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `beta` `(cols(x)+cols(d))`$\times$`1)` vector containing the parameter estimates
  - `beta[1:cols(x),1]` contains the coefficients of x; the remaining elements are the coefficients on the deterministic variables in the fitted regression
- `vc` variance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `sigma` standard error of the residuals
- `tstats` t-statistics for the parameter estimates
- `rss` residual sum of squares
- `resid` estimated residuals [nobs $\times$ 1]
- `dummy` dummy vector (3$\times$1) vector of zeros

GLOBALS

- `_ker_fun; _aband; _filter; _NoDet`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **CCR**.

EXAMPLE

```
/* Cointegrating Regression: y = a + Xb + e */
/* Automatic bandwidth disabled — _aband = 0; Use AR(1) filter when computing the
spectrum at frequency zero. */
_ker_fun = &parzen; _aband = 0; _filter = 1;
dd = ones(rows(y),1); {a,b,c,d,e,f,g,h} = ccr(y,x,dd,10);
"CCR X beta estimates, t-statistics:" a[1:cols(x),1]
e[1:cols(x),1];
"CCR estimate for constant term:" a[cols(x)+1:cols(x)+1,1]
~e[cols(x)+1:cols(x)+1,1];
/* Suppress the deterministic part (i.e., the constant term in this example)
by setting _NoDet = 1 */
_NoDet = 1; {a,b,c,d,e,f,g,h} = ccr(y,x,dd,20);
_NoDet = 0; ``CCR X beta estimates, t-statistics:''
a[1:cols(x),1] ~e[1:cols(x),1];
```

REFERENCE

Park, J. Y. (1992) "Canonical Cointegrating Regressions," *Econometrica*, **60:** 119–144.

FM(y, x, d, l)

PURPOSE

- Computes the Phillips–Hansen (1990) "Fully-Modified" estimator for cointegrated regressions, using OLS for the first stage regression.

FORMAT

- `{beta,vc,stderr,sigma,tstats,rss,resid,stests} = fm(y,x,d,l);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `d` deterministic part of the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `beta` (`cols(x)+cols(d)`)$\times$1) vector containing the parameter estimates
  - `beta[1:cols(x),1]` contains the coefficients of x; the remaining elements are the coefficients on the deterministic variables in the fitted regression}
- `vc` variance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `sigma` standard error of the residuals
- `tstats` t-statistics for the parameter estimates
- `rss` residual sum of squares
- `resid` estimated residuals [nobs $\times$ 1]
- `stests` 3$\times$1 vector containing Hansen's (1991) Lc, MeanF, and SupF (in this order) statistics for testing the null hypothesis that the cointegrating vector is stable over the sample period

GLOBALS

- `_ker_fun; _aband ; _filter ; _sbstart, _sbend; _NoDet`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **FM**.
- See **KERNELS.SRC** for a detailed explanation of the other global constants.

EXAMPLE

```
/* Cointegrating Regression: y = a + Xb + e */
_ker_fun = &parzen; _aband = 1; _filter = 1;
dd = ones(rows(y),1); {a,b,c,d,e,f,g} = fm(y,x,dd,10);
''FM X beta estimates, t-statistics:'' a[1:cols(x),1]
e[1:cols(x),1];
''FM estimate for constant term:'' a[cols(x)+1:cols(x)+1,1]
e[cols(x)+1:cols(x)+1,1];
```

REFERENCES

Hansen, B. E. (1992) "Tests for Parameter Instability in Regressions with I(1) Processes," *Journal of Business and Economic Statistics*, **10:** 321–335.

Phillips, P. C. B. & B. E. Hansen (1990) "Statistical Inference in Instrumental Variables Regression with I(1) Processes," *Review of Economic Studies*, **57:** 99–125.

FM_OLS(y, z, d, l)

PURPOSE

- Computes the Phillips' (1993) "Fully-Modified" OLS estimator for single equation and multivariate cointegrated regression models.

FORMAT

- `{beta,vc} = fm_ols(y,x,d,l);`

INPUTS

- `y` dependent variable (nobs $\times$ n)
- `x` explanatory variables (nobs $\times$ m)
- `d` deterministic part in the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `beta` `(m+cols(d))`$\times$`n` vector containing the parameter estimates
  - `beta[1:m,.]` contains the coefficients of x; the remaining elements are the coefficients on the deterministic variables in the fitted regression}
- `vc` variance matrix for the parameter estimates

GLOBALS

- `_ker_fun; _aband ; _filter ; _NoDet`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **FM_OLS**.
- See **KERNELS.SRC** for a detailed explanation of the other global constants.

EXAMPLE

/* Cointegrating Regression: $y = a + Xb + e$ */

```
_ker_fun = &parzen; _aband = 1; _filter = 1;
dd = ones(rows(y),1); {bhat,vc} = fm_ols(y,x,dd,5);
``FM_OLS X beta estimates:'' bhat[1:cols(x),1:cols(y)];
``FM_OLS estimate for the deterministic part'' bhat[cols(x)+1:
    cols(x)+n,1:cols(y)]
``Now drop the deterministic term (dd) by setting _NoDet = 1'';
_NoDet = 1;
{bhat,vc} = fm_ols(y,x,d,10);
``FM X beta estimates:''  bhat[1:cols(x),1];
```

REFERENCE

Phillips, P. C. B (1995) "Fully Modified Least Squares and Vector Autoregression," *Econometrica,* **63:** 1023–1078.

FM_VAR(y, d, k, l)

PURPOSE

- Computes the Phillips' (1993) "Fully-Modified" VAR estimator for cointegrated regressions, using OLS for the first stage regression.

FORMAT

- `{beta,vc} = fm_var(y,d,k,l);`

INPUTS

- `y` dependent variable (nobs $\times$ n)
- `d` deterministic part in the fitted regression
- `k` number of lagged innovation terms to include in the fitted regression (Δy_{t-1}, ..., Δy_{t-k+1})
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `beta` `[(k-1)*n+n+cols(d)]`$\times$`n` vector containing the parameter estimates
  - `beta[1:(k-1)*n,.]` contains the coefficients of the lagged innovation terms;
  - `beta[(k-1)*n+1:k*n,.]` contains the coefficients on the lagged dependent variables;
  - the remaining elements are the coefficients on the deterministic variables in the fitted regression.
- `vc` variance matrix for the parameter estimates

GLOBALS

- `_ker_fun; _aband ; _filter ; _NoDet`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **FM_VAR**.
- See **KERNELS.SRC** for a detailed explanation of the other global constants.

EXAMPLE

```
_ker_fun = &parzen; _aband = 1; _filter = 1;
dd = ones(rows(y),1); {a,b} = fm_var(y,dd,5,5);
''FM_VAR lagged innovation parameter estimates:''
    a[1:4*n,1:cols(y)];
''FM_VAR lagged Y estimates:'' a[4*n+1:5*n,1:cols(y)];
''FM_VAR estimate for constant term:'' a[5*n+1:6*n,1:cols(y)];
''Drop the constant term.by setting _NoDet = 1'';
_NoDet = 1;
{a,b} = fm_var(y,d,5,5);
''FM_VAR lagged Y estimates:'' a[4*n+1:5*n,1:cols(y)];
```

REFERENCE

Phillips, P. C. B (1995) "Fully Modified Least Squares and Vector Autoregression," *Econometrica,* **63:** 1023–1078.

FM_GIVE(y, x, z, l, t)

PURPOSE

- Computes the Kitamura–Phillips (1997) "Fully-Modified" GIVE estimator for single equation and multivariate cointegrated regression models.

FORMAT

- `{beta,vc,lromega,s1,s2} = fm_give(y,x,z,l,t);`

INPUTS

- `y` dependent variables (nobs $\times$ n)
- `x` integrated explanatory variables (nobs $\times$ m)
- `z` instrumental variables (nobs $\times$ z)
- `l` number of autocovariance terms to compute spectrums at frequency zero
- `t` number of terms to use in the computation of the W_T matrix

OUTPUTS

- `beta` $m \times n$ vector containing the parameter estimates
- `vc` variance–covariance matrix of the parameter estimates
- `lromega` long-run variance–covariance matrix of residuals ($n \times n$)
- `s1` first statistic for testing validity of overidentifying restrictions
- `s2` second statistic for testing validity of overidentifying restrictions

GLOBALS

- `_ker_fun; _aband; _filter`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **FM_GMM**.
- See **KERNELS.SRC** for a detailed explanation of the other global constants.

EXAMPLE

/* Cointegrating Regression: $y = a + Xb + e$ */
```
_ker_fun = &parzen; _aband = 1; _filter = 1;
{bhat,vc,lromega,s1,s2} = fm_give(y,x,z,5,3);
``FM_GIVE X beta estimates:'' bhat;
```

REFERENCE

Kitamura, Y. & P. C. B. Phillips (1997) "Fully Modified IV, GIVE and GMM Estimation with Possibly Nonstationary Regressors and Instruments," *Journal of Econometrics,* **80:** 85 –123.

FM_GMM(y, x, z, l)

PURPOSE

- Computes the Kitamura–Phillips (1997) "Fully-Modified" GMM estimator for single equation and multivariate cointegrated regression models.

FORMAT

- `{beta,vc,lromega,s1,s2} = fm_gmm(y,x,z,l);`

INPUTS

- `y` dependent variables (nobs $\times$ n)
- `x` integrated explanatory variables (nobs $\times$ m)
- `z` instrumental variables (nobs $\times$ z)
- `l` number of autocovariance terms to compute spectrums at frequency zero

OUTPUTS

- `beta` $m \times n$ vector containing the parameter estimates
- `vc` variance–covariance matrix of the parameter estimates
- `lromega` long-run variance–covariance matrix of residuals ($n \times n$)
- `s1` first statistic for testing validity of overidentifying restrictions
- `s2` second statistic for testing validity of overidentifying restrictions

GLOBALS

- `_ker_fun; _aband; _filter`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **FM_GIVE**.
- See **KERNELS.SRC** for a detailed explanation of the other global constants.

EXAMPLE

/* Cointegrating Regression: $y = a + Xb + e$ */

```
_ker_fun = &parzen; _aband = 1; _filter = 1;
{bhat,vc,lromega,s1,s2} = fm_gmm(y,x,z,5,3);
''FM_GIVE X beta estimates:'' bhat;
```

REFERENCE

Kitamura, Y. & P. C. B. Phillips (1997) "Fully Modified IV, GIVE and GMM Estimation with Possibly Nonstationary Regressors and Instruments," *Journal of Econometrics*, **80:** 85 –123.

PCB(y, x, d, M)

PURPOSE

- Computes Phillips' (1990) spectral estimator for cointegrated regression models.

FORMAT

- `{beta,vc,stderr,tstats} = pcb(y,x,d,M);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `d` deterministic part in the fitted regression (nobs $\times$ cols(`d`))
- `M` bandwidth parameter

OUTPUTS

- `beta` ($k \times 1$) vector containing the parameter estimates
- `vc` variance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `tstats` t-statistics for the parameter estimates

GLOBALS

- Set `_NoDet = 1` to exclude the deterministic part from the cointegrating regression.

REMARK

1. The procedure calculates the spectrum by taking simple averages of the periodograms within a particular band. The number of periodogram ordinates is given by $nt(y)/(2M)$, where $nt(y)$ = number of observations augmented to the nearest power of 2.

EXAMPLE

```
/* Cointegrating Regression: y = a + Xb + e */
{a,b,c,d} = pcb(y,x,ones(rows(x),1),2);
``PCB beta estimates, t-statistics:'' a~d;
/* Now use demeaned data, and suppress the constant term */
dummy = ones(rows(x),1); _NoDet = 1;
{a,b,c,d} = pcb(detrend(y,0), detrend(x,0), dummy,2);
``PCB beta estimates, t-statistics:'' a~d;
```

REFERENCE

Phillips, P. C. B. (1990) "Spectral Regression for Cointegrated Time Series," in W. Barnett (ed.), *Nonparametric and Semiparametric Methods in Economics and Statistics.* Cambridge: Cambridge University Press.

PCBBW(y, x, d, f, M)

PURPOSE

- Computes Phillips' (1990) spectral estimator for cointegrated regression models, using the periodogram ordinates indicated by f.

FORMAT

- {`beta,vc,stderr,tstats`} `= pcbbw(y,x,d,f,M);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `d` deterministic part (nobs $\times$ cols(`d`))
- `f` $nt(y) \times 1$ indicator vector
- `M` bandwidth parameter

OUTPUTS

- `beta` $(k \times 1)$ vector containing the parameter estimates
- `vc` variance–covariance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `tstats` t-statistics for the parameter estimates.

GLOBALS

- Set `_NoDet = 1` to exclude the deterministic part from the cointegrating regression.

REMARKS

1. The procedure calculates the spectrum by taking simple averages of the periodograms within a particular band. The number of periodogram ordinates is given by $nt(y)/(2M)$, where $nt(y)$ = number of observations augmented to the nearest power of 2.

2. The f vector allows you to exclude periodogram ordinates. It must contain zeros or ones (and nothing else). Set $f[i, 1] = 0.0$ to exclude the i-th ordinate (where i is relative to $[0, 2\pi)$), and $f[j, 1] = 1.00$ to include the j-th ordinate. The f vector MUST have dimension $nt(y) \times 1$.

EXAMPLE

```
/* Cointegrating Regression: y = a + Xb + e */
/* Estimate over the interval [π/3,π] */
ZeroFreq = round(nt(y)/3); /* Number of ordinates to exclude, starting
from the zero frequency */
f = ones(nt(y),1); f[1:ZeroFreq+1,1] = zeros(ZeroFreq+1,1);
f[nt(y)-ZeroFreq+1:nt(y),1] = zeros(ZeroFreq,1);
_NoDet = 1; /* Suppress the constant term over the high frequencies */
{a,b,c,d} = pcbbw(detrend(y,0), detrend(x,0),
    ones(rows(x),1),f,2);
''PCB bandwidth beta estimates, t-statistics:'' a~d;
```

REFERENCE

Phillips, P. C. B. (1990) "Spectral Regression for Cointegrated Time Series," in W. Barnett (ed.), *Nonparametric and Semiparametric Methods in Economics and Statistics.* Cambridge: Cambridge University Press.

PCBZ(y, x, d, M)

PURPOSE

- Computes Phillips' (1990) spectral estimator for cointegrated regression models, using the low-frequency ordinates (i.e., $\beta(0)$).

FORMAT

- `{beta,vc,stderr,tstats} = pcbz(y,x,d,M);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `d` deterministic part of the cointegrating regression
- `M` bandwidth parameter

OUTPUTS

- `beta` ($k \times 1$) vector containing the parameter estimates
- `vc` variance–covariance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `tstats` t-statistics for the parameter estimates

GLOBALS

- `_NoDet`
- Set `_NoDet = 1` to exclude the deterministic part from the cointegrating regression.

REMARK

1. The procedure calculates the spectrum by taking simple averages of the periodograms within a particular band. This routine uses the zero-frequency spectrum to compute the parameters of the cointegrating regression. The number of periodogram ordinates is given by $nt(y)/(2M)$, where $nt(y) =$ number of observations augmented to the nearest power of 2.

EXAMPLE

```
/* Cointegrating Regression: y = a + Xb + e */
/* Compute using PCBZ, allowing for a constant (default) */
{a,b,c,d} = pcbz(y,x,ones(rows(y),1),2);
``PCB beta(0) estimates, t(0)-statistics:'' a~d;
```

REFERENCE

Phillips, P. C. B. (1990) "Spectral Regression for Cointegrated Time Series," in W. Barnett (ed.), *Nonparametric and Semiparametric Methods in Economics and Statistics*. Cambridge: Cambridge University Press.

GPCB(y, x, z,d, M)

PURPOSE

- Computes Phillips' (1990) GIVE spectral estimator for cointegrated regression models.

FORMAT

- `{beta,vc,stderr,tstats} = gpcb(y,x,z,d,M);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `z` instrumental variables (nobs $\times$ cols(z))
- `d` deterministic part (nobs $\times$ cols(d))
- `M` bandwidth parameter

OUTPUTS

- `beta` ($k \times 1$) vector containing the parameter estimates
- `vc` variance–covariance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `tstats` t-statistics for the parameter estimates

GLOBALS

- `_NoDet`
- Set `_NoDet = 1` to exclude the deterministic part from the cointegrating regression.

REMARK

1. The procedure calculates the spectrum by taking simple averages of the periodograms within a particular band. The number of periodogram ordinates is given by $nt(y)/(2M)$, where $nt(y)$ = number of observations augmented to the nearest power of 2.

EXAMPLE

```
/* Cointegrating Regression: y = a + Xb + e */
{a,b,c,d} = gpcb(y,x,z,ones(rows(x),1),2);
''PCB GIVE beta estimates, t-statistics:'' a~d;
```

REFERENCES

Corbae, P. D., S. Ouliaris & P. C. B. Phillips (1994) "A Reexamination of the Consumption Function Using Frequency Domain Regressions," *Journal of Empirical Economics,* **19**: 595–609.

Phillips, P. C. B. (1990) "Spectral Regression for Cointegrated Time Series," in W. Barnett (ed.), *Nonparametric and Semiparametric Methods in Economics and Statistics.* Cambridge: Cambridge University Press.

GPCBBW(y, x, z, d, f, M)

PURPOSE

- Computes Phillips' (1990) GIVE spectral estimator for cointegrated regression models, using the periodogram ordinates indicated by f.

FORMAT

- `{beta,vc,stderr,tstats} = gpcbbw(y,x,z,d,f,M);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `z` instrumental variables (nobs $\times$ cols(z))
- `d` deterministic part in the fitted regression (nobs $\times$ cols(`d`))
- `f` $nt(y) \times 1$ indicator vector
- `M` bandwidth parameter

OUTPUTS

- `beta` ($k \times 1$) vector containing the parameter estimates
- `vc` variance–covariance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `tstats` t-statistics for the parameter estimates

GLOBALS

- `_NoDet`
- Set `_NoDet = 1` to exclude the deterministic part from the cointegrating regression.

REMARKS

1. The procedure calculates the spectrum by taking simple averages of the periodograms within a particular band. The number of periodogram ordinates is given by $nt(y)/(2M)$, where $nt(y)$ = number of observations augmented to the nearest power of 2.
2. The f vector allows you to exclude periodogram ordinates. It must contain zeros or ones (nothing else).
3. Set $f[i,1] = 0.0$ to exclude the i-th ordinate (where i set is relative to $[0, \pi)$) and $f[j,1] = 1.00$ to include the j-th ordinate. The f vector **MUST** have dimension $nt(y) \times 1$.

EXAMPLE

/* Cointegrating Regression: $y = a + Xb + e$ over the interval $[\pi/3, \pi]$ */

```
ZeroFreq = round(nt(y)/3); /* Number of ordinates to exclude, starting from the zero frequency */
f = ones(nt(y),1); f[1:ZeroFreq+1,1] = zeros(ZeroFreq+1,1);
f[nt(y)-ZeroFreq+1:nt(y),1] = zeros(ZeroFreq,1); _NoDet = 1;
{a,b,c,d} = gpcbbw(detrend(y,0), detrend(x,0),
    detrend(rows(z),0), ones(rows(x),1),f,2);
```

REFERENCES

Corbae, P. D., S. Ouliaris & P. C. B. Phillips (1994) "A Reexamination of the Consumption Function Using Frequency Domain Regressions," *Journal of Empirical Economics*, **19**: 595–609.

Phillips, P. C. B. (1990) "Spectral Regression for Cointegrated Time Series," in W. Barnett (ed.), *Nonparametric and Semiparametric Methods in Economics and Statistics*. Cambridge: Cambridge University Press.

GCBZ(y, x, z, d, M)

PURPOSE

- Computes Phillips' (1990) GIVE spectral estimator for cointegrated regression models, using the low-frequency ordinates (i.e., $\beta_Z(0)$).

FORMAT

- `{beta,vc,stderr,tstats} = gpcbz(y,x,z,d,M);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `z` instrumental variables
- `d` deterministic part in the fitted regression (nobs $\times$ cols(`d`))
- `M` bandwidth parameter

OUTPUTS

- `beta` $(k \times 1)$ vector containing the parameter estimates
- `vc` variance–covariance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `tstats` t-statistics for the parameter estimates

GLOBALS

- `_NoDet`
- Set `_NoDet = 1` to suppress the deterministic part from the cointegrating regression.

REMARK

1. The procedure calculates the spectrum by taking simple averages of the periodograms within a particular band. In contrast to **GPCB, GPCBZ** uses only the zero-frequency spectrum to compute the parameters of the cointegrating regression. The number of periodogram ordinates is given by $nt(y)/(2M)$, where $nt(y)$ = number of observations augmented to the nearest power of 2.

EXAMPLE

```
/* Cointegrating Regression: y = a + Xb + e */
{a,b,c,d} = gpcbz(y,x,z,ones(rows(x),1),2);
``PCB GIVE beta(0) estimates, t(0)-statistics:'' a~d;
```

REFERENCES

Corbae, P. D., S. Ouliaris and P. C. B. Phillips (1994) "A Reexamination of the Consumption Function Using Frequency Domain Regressions," *Journal of Empirical Economics*, **19**: 595–609.

Phillips, P. C. B. (1990) "Spectral Regression for Cointegrated Time Series," in W. Barnett (ed.), *Nonparametric and Semiparametric Methods in Economics and Statistics*. Cambridge: Cambridge University Press.

SJ(x, p, k)

PURPOSE

- Computes Johansen's (1988) ML estimator.

FORMAT

- `{ev,evec,lr1,lr2} = sj(x,p,k);`

INPUTS

- `x` data matrix (nobs $\times$ m)
- `p` order of the time polynomial in the fitted regression
- `k` number of lagged difference terms to use when computing the estimator

OUTPUTS

- `ev` ($m \times 1$) vector containing the eigen values
- `evec` ($m \times m$) matrix containing the eigen vectors. First r columns are the unnormalized cointegrating vectors
- `lr1` ($m \times 1$) vector of Johansen's likelihood ratio trace statistics for $r = 0$ to m–1 cointegrating vectors
- `lr2` Johansen's max statistic for the null hypothesis of 0 to m–1 cointegrating vectors

GLOBALS

- Set `_NoDet = 1` to suppress the constant term from the fitted regression and include it in the cointegrating regression.

REMARKS

1. Set $p = 0$ for a constant term.
2. Set $p = 1$ for a constant term and trend.
3. Refer to Johansen and Juselius (1990) for tabulations of the critical values.
4. **SJ( )** uses the **GAUSS** routine EIGRG2(x), which sets the global variable `_eigerr`. Refer to the **GAUSS** manual for more details.

EXAMPLE

/* Cointegrating Regression: $y = a + Xb + e$ */

```
{a,b,lr1,lr2} = sj(y~x,0,10); ``Johansen trace statistics for
    r=0 to r=cols(y~x)-1'' lr1;
``Estimate of the (unnormalized) cointegrating vector:'' b[.,1];
```

REFERENCES

Johansen, S. J. (1988) "Statistical Analysis of Cointegration Vectors," *Journal of Economic Dynamics and Control*, **12:** 231–254.

Johansen, S. J. (1991) "Estimation and Hypothesis Testing of Cointegration Vectors in Gaussian Vector Autoregressive Models," *Econometrica*, **59:** 1551–1580.

Johansen, S. J. & K. Juselius (1990) "Maximum Likelihood Estimation and Inference on Cointegration — with Applications to the Demand for Money," *Oxford Bulletin of Economics and Statistics*, **52:** 169–210.

PS(y, x, p, ld, lg)

PURPOSE

- Computes Saikkonen's (1991) estimator for cointegrated regressions.

FORMAT

- `{beta,vc,stderr,sigma,tstats,rss,resid} = ps(y,x,d,ld,lg);`

INPUTS

- `y` dependent variable
- `x` explanatory variables (I(1) only) (nobs $\times$ m)
- `d` deterministic part of the cointegrating regression
- `lg` number of lagged differences of x
- `ld` number of lead differences of x

OUTPUTS

- `beta` vector containing the parameter estimates, organized as follows:
  - Let `m = cols(x); p = cols(d);`
  - `beta[1:p,1]` — p coefficients of the deterministic part;
  - `beta[1+p:m+p,1]` — m coefficients of x;
  - Remaining terms are the coefficient estimates for the lag/lead differences of x.
- `vc` variance–covariance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `sigma` standard error of the residuals
- `tstats` t-statistics for the parameter estimates
- `rss` residual sum of squares
- `resid` estimated residuals (nobs $\times$ 1)

GLOBALS

- `_NoDet`
- Set `_NoDet = 1` to suppress the deterministic part from the cointegrating regression.

REMARK

1. `lg` and `ld` can be set to zero. Doing so will suppress the lead/lag variables.

REFERENCES

Phillips, P.C.B. & M. Loretan (1991) "Estimating Long Run Economic Equilibrium," *Review of Economic Studies*, **58:** 407–436.

Saikkonen, P. (1991) "Asymptotically Efficient Estimation of Cointegrating Regressions," *Econometric Theory*, **7:** 1–21.

Stock, J. H. & Watson, M. W. (1991) "A Simple MLE of Cointegrating Vectors in Higher Order Integrated Systems," mimeo, Department of Economics, Northwestern University.

HSTATC(y, x, p, q, l)

PURPOSE

- Computes Park's (1992) $H(p,q)$ statistic for the null-hypothesis that y and x are cointegrated, using **CCR( )**.

FORMAT

- `{hs,pv} = hstatc(y,x,p,q,l);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `p,q` orders of the time polynomial in the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `hs` $h(p,q)$ statistic, which possesses a chi-squared distribution (asymptotically) with $q-p$ degrees of freedom
- `pv` p-value of the $H(p,q)$ statistic

DECISION RULE

- `_ker_fun; _filter; _aband`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **HSTATC**.

EXAMPLE

```
/* Compute H(0,5) statistic using the Parzen kernel */
_ker_fun = &parzen ; _filter = 1; _aband = 0;
{hst,pvalue} = hstatc(y,x,0,5,20);
``H(0,5) statistic; p-value =’’ hst~pvalue;
```

REFERENCE

Park, J. Y. (1992) "Canonical Cointegrating Regressions," *Econometrica*, **60:** 119–144.

HSTATF(y, x, p, q, l)

PURPOSE

- Computes Park's (1988) $H(p,q)$ statistic for the null-hypothesis that y and x are cointegrated, using **FM( )**.

FORMAT

- `{hs,pv} = hstatf(y,x,p,q,l);`

INPUTS

- `y` dependent variable (nobs × 1)
- `x` explanatory variables (nobs × k)
- `p,q` orders of the time polynomial in the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `hs` $h(p,q)$ statistic, which possesses a chi-squared distribution (asymptotically) with $q-p$ degrees of freedom
- `pv` p-value of the $H(p,q)$ statistic

GLOBALS

- `_ker_fun; _filter; _aband`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **HSTATF**.

EXAMPLE

```
/* Compute H(0,5) statistic using the Parzen kernel */
_ker_fun = &parzen; _aband = 0; _filter = 1;
{hst,pvalue} = hstatf(y,x,0,5,20);
``H(0,5) statistic; p-value (Parzen kernel) ='' hst~pvalue;
```

REFERENCES

Park, J. Y. (1992) "Canonical Cointegrating Regressions," *Econometrica*, **60:** 119–144.

Phillips, P. C. B. & B. E. Hansen (1990) "Statistical Inference in Instrumental Variables Regression with I(1) Processes," *Review of Economic Studies*, **57:** 99–125.

HSTAT(y, x, p, q, l)

PURPOSE

- Driver routine for **HSTATC** and **HSTATF**.

FORMAT

- `{hs,pv} = hstat(y,x,p,q,l);`

INPUTS

- `y` dependent variable (nobs $\times$ 1)
- `x` explanatory variables (nobs $\times$ k)
- `p,q` orders of the time polynomial in the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `hs` $h(p,q)$ statistic, which possesses a chi-squared distribution (asymptotically) with
  - $q-p$ degrees of freedom
- `pv` p-value of the $H(p,q)$ statistic

GLOBALS

- `_ker_fun; _hstat; _filter; _aband`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **HSTAT**.
- Set `_hstat c` to either `&HSTATF` or `&HSTATC` before calling this routine.

EXAMPLE

```
/* Compute H(0,5) statistic using the Parzen kernel and HSTATF */
_ker_fun = &parzen;
_hstat = &hstatf; _aband = 1; _filter = 0;
{hs,pv} = hstat(y,x,0,5,20);
''FM H(0,5) statistic; p-value ='' hs~pv;
/* Now use HSTATC */
_ker_fun = &parzen;
_hstat = &hstatc;
{hs,pv} = hstat(y,x,0,5,20);
```

```
''FM H(0,5) statistic; p-value ='' hs~pv;
```

CREGR(y, x, d, l)

PURPOSE

- Driver routine for **CCR( )** and **FM( )**

FORMAT

- `{beta,vc,stderr,sigma,tstats,rss,resid,stests}=cregr(y,x,d,l);`

INPUTS

- `y` dependent variable (nobs × 1)
- `x` explanatory variables (nobs × k)
- `d` deterministic part in the fitted regression
- `l` number of autocovariance terms to compute the spectrum at frequency zero

OUTPUTS

- `beta` (`cols(x)+cols(d)`) vector containing the parameter estimates
  - `beta[1:cols(x),1]` contains the coefficients of x.}
- `vc` variance–covariance matrix for the parameter estimates
- `stderr` standard errors of the parameter estimates
- `sigma` standard error of the residuals
- `tstats` t-statistics for the parameter estimates
- `rss` residual sum of squares
- `resid` estimated residuals (nobs × 1)
- `stests` (3×1) vector containing B. Hansen's structural stability tests; namely, Lc,
  - MeanF, and SupF (only available when `_cregr = &FM`)

GLOBALS

- `_ker_fun ; _cregr; _filter; _sbstart, _sbend; _aband`
- Set `_ker_fun` to one of the available kernels (see **KERNELS.SRC**) before using **CREGR**.
- Set `_cregr` to `&CCR` or `&FM` before calling this routine.
- See **KERNELS.SRC** for a detailed explanation of the global constants.

EXAMPLE

```
/* Cointegrating Regression: y = a + Xb + e */
_ker_fun = &parzen; _cregr = &fm; _aband = 0; _filter = 1;
dd = ones(rows(y),1);{a,b,c,d,e,f,g,stests} = cregr(y,x,dd,20);
``FM X beta estimates, t-statistics:'' a[1:cols(x),1]
     ~e[1:cols(x),1];
 ``FM estimate for constant term:'' a[cols(x)+1:cols(x)+1,1]
     ~e[cols(x)+1:cols(x)+1,1];
/* Recompute using Tukey–Hamming kernel and CCR */
_ker_fun = &tukham;
_cregr = &ccr;
{a,b,c,d,e,f,g,stests} = cregr(y,x,dd,20);
``CCR X beta estimates, t-statistics:'' a[1:cols(x),1]
     ~e[1:cols(x),1];
 ``CCR estimate for constant term:'' a[cols(x)+1:cols(x)+1,1]
     ~e[cols(x)+1:cols(x)+1,1];
```

ARMA.SRC

ARMA(p,q) Recursive Least Squares Estimation and Model Selection Procedures

Introduction:

COINT 2.1 versus COINT 2.0

COINT 2.0 used throughout an OLS (degrees of freedom adjusted) error variance estimator to calculate the AIC, BIC and PIC model selection criteria. These calculations involve two separate parts: (1) the estimator for the variance used to calculate the AIC, BIC, and PIC statistics, and (2) the estimated variance used to calculate the posterior odds ratio. For many applications an MLE variance estimator, which is not adjusted for the degrees of freedom, may be preferred. This option is now available. New to COINT 2.1, the global "**_variance**" switch allows you use an MLE estimator to calculate these statistics.

COINT 2.1 recognizes three settings for the **_variance** switch:

1. **_variance** = 0, in which case the OLS estimator is used to calculate the AIC, BIC, and PIC criteria. The associated variance-covariance matrix of the parameter estimates, which is used to calculated the posterior odds ratio, is also estimated using the OLS estimator in this case. Using **_variance** = 0 replicates COINT 2.0, and remains the default.

2. **_variance** = 1, in which case the MLE estimator is used to calculate the AIC, BIC, and PIC criteria. The variance-covariance matrix of the parameter estimates (and hence the posterior odds ratio) is calculated using the OLS estimator.

3. **_variance** = 2. In this case, the MLE estimator is used to calculate all model selection criteria and the variance-covariance matrix of the parameter estimates.

The routines affected by the **_variance** switch are: **pparord(), ppadfbic(), ppadfbit(), armamay(), armamayy(), pparord()**, including any routines that depend on these routines to calculate the various model selection criteria.

ARORD(x, pmax, tmax)

PURPOSE

- Computes the Akaike (1969) AIC, Schwarz (1978) BIC, and Phillips–Ploberger (1994) PIC criterion or order selection in an autoregression with deterministic trend.

FORMAT

- `{aic,bic,pic} = arord(x,pmax,tmax);`

INPUTS

- `x` dependent variable (nobs × 1)
- `pmax` maximum AR lag (`pmax` ≥ `0`)
- `tmax` maximum degree of deterministic trend (`tmax` ≥ `-1`)

OUTPUTS

- `aic` `(pmax+1 × tmax+2)` array of AIC values
- `bic` `(pmax+1 × tmax+2)` array of BIC values
- `pic` `(pmax+1 × tmax+2)` array of PIC values

EXAMPLE

```
/* Calculate AIC, BIC and PIC criteria for an AR(p) + TR(t) model */
{aic,bic,pic} = arord(x,pmax,tmax);
"aic values'' = aic;
"bic values =" bic;
"pic values =" pic;
```

RELATED PROCEDURES

- **ARBIC, ARPIC, PPARORD**

REMARKS

1. Set `pmax` > `0` or `tmax` < `-1`.
2. Use _variance = 1 to use a maximum likelihood estimator for the variance used to calculate the model selection criteria. See page 63 for more information.

REFERENCES

Akaike H. (1969) "Fitting Autoregressive Models for Prediction," *Annals of the Institute of Statistical Mathematics*, **21**: 243–247.

Phillips, P. C. B. & W. Ploberger (1994) "Posterior Odds Testing for a Unit Root with Data Based Model Selection," *Econometric Theory,* **10**(3/4): 774–808.

Schwarz, G. (1978) "Estimating the Dimension of a Model," *Annals of Statistics*, **6**: 461–464.

ARBC(x, pmax, tmax)

PURPOSE

- Computes the Schwarz (1978) BIC criterion for an autoregression with deterministic trend and reports the selected lag order and trend degree. The BIC values are output in array form and graphed as a surface in 3-D. Estimated coefficients of the selected model are returned together with standard errors, t-ratios and variance matrix. Additional graphs are returned of the BIC values for the AR order conditional on the selected trend degree and the trend degree conditional on the selected AR lag order.

FORMAT

- `{bic,p,t,b,st,trat,vmat} = arbc(x,pmax,tmax);`

INPUTS

- `x` dependent variable (nobs $\times$ 1)
- `pmax` maximum AR lag (`pmax` $\geq$`0`)
- `tmax` maximum degree of deterministic trend (`tmax` $\geq$`-1`)

OUTPUTS

- `bic` (`pmax+1` $\times$`tmax+2`) array of BIC values
- `p` selected AR order
- `t` selected degree of deterministic trend
- `b` (p+t+1)-vector of estimated coefficients of selected model
- `st` standard errors of the parameter estimates
- `trat` t-ratios of the parameter estimates
- `vmat` variance matrix of the parameter estimates

EXAMPLE

```
/* Model selection for an AR(l) + TR(t) by BIC */
{bic,p,t,b,st,trat,vmat} = arbc(x,pmax,tmax);
``bic values ='' bic; ``selected lag =`` p;
``selected trend ='' t;``coefficients'' = b';
``st. errors ='' st'; ``t-ratios ='' trat';
```

REMARK

1. Use _variance = 1 to use a maximum likelihood estimator for the variance when calculating the BIC criterion. See page 63 for more information.

RELATED PROCEDURES

- **ARBIC, ARSTAT, GRFBIC, GRFBICL, GRFBICT**

REFERENCE

Schwarz, G. (1978) "Estimating the Dimension of a Model," *Annals of Statistics*, **6**: 461–464.

ARPC(x, pmax, tmax)

PURPOSE

- Computes the Phillips–Ploberger (1994) PIC criterion for an autoregression with deterministic trend and reports the selected lag order and trend degree. The PIC values are output in array form and graphed as a surface in 3-D. Estimated coefficients of the selected model are returned together with standard errors, t-ratios and variance matrix. Additional graphs are returned of: (i) PIC values for the AR order conditional on the selected trend degree; (ii) the trend degree conditional on the selected AR lag order; and (iii) the PIC value of odds in favor of a unit root against the PIC values for different AR orders conditional on the selected trend degree.

FORMAT

- `{bic,picu,p,t,b,st,trat,vmat} = arpc(x,pmax,tmax);`

INPUTS

- `x` dependent variable (nobs × 1)
- `pmax` maximum AR lag (`pmax` ≥ `0`)
- `tmax` maximum degree of deterministic trend (`tmax` ≥ `-1`)

OUTPUTS

- `pic` (`pmax+1` × `tmax+2`) array of PIC values
- `picu` odds in favor of unit root
- `p` selected AR order
- `t` selected degree of deterministic trend
- `b` (p+t+1)-vector of estimated coefficients of selected model
- st standard errors of the parameter estimates
- `trat` t-ratios of the parameter estimates
- `vmat` variance matrix of the parameter estimates

EXAMPLE

```
/* Model selection for an AR(l) + TR(t) by PIC */
{pic,picu,p,t,b,st,trat,vmat} = arpc(x,pmax,tmax);
``pic values ='' pic; ``odds in favor of a unit root ='' picu;
``selected lag ='' p; ``selected trend ='' t;
``coefficients ='' b';``st. errors =''st';
``t-ratios ='' trat';
```

RELATED PROCEDURES

- **ARPIC, ARSTAT, GRFPIC, GRFPICL, GRFPICT**

REMARKS

1. The graphics routines are interactive and allow the user to alter the perspective of viewing the PIC surface.

2. PIC tests for a unit root in the series are incorporated in the graphs and show the gain/loss in odds from including a unit root in the specification.

3. Use _variance = 1 or _variance = 2 to use a maximum likelihood estimator for the variance when calculating the pic/picu. See page 63 for more information.

REFERENCE

Phillips, P. C. B. & W. Ploberger (1994) "Posterior Odds Testing for a Unit Root with Data Based Model Selection," *Econometric Theory,* **10**(3/4): 774–808.

ADFTR(x, p, r)

PURPOSE

- Estimates coefficients (using OLS) of an AR(p) + TR(r) model formulated in levels and differences as:

$$\begin{aligned} x(t) &= b1 * x(t-1) + b2 * del(x(t-1)) + \cdots \\ &\quad + bp * del(x(t-p+1) + a0 + \cdots + ar * t^r \end{aligned} \tag{4}$$

  Standard errors and t-ratios and variance matrix are returned. When $r = -1$ in equation (1) there is no intercept in the regression.

FORMAT

- `{b,st,trat,vmat} = adftr(x,p,r);`

INPUTS

- `x` dependent variable (nobs $\times$ 1)
- `p` AR lag ($p \geq 0$)
- `r` degree of deterministic trend ($r \geq -1$)

OUTPUTS

- `b` (p+r+1)-vector of estimated coefficients of selected model
- `st` standard errors of the parameter estimates
- `trat` t-ratios of the parameter estimates
- `vmat` variance matrix of the parameter estimates

EXAMPLE

```
/* Estimation of an AR(p) + TR(r) by OLS */
{b,st,trat,vmat} = adftr(x,p,r);
''coefficients ='' b'; ''st. errors =`` st'; ''t-ratios =`` trat';
```

RELATED PROCEDURES

- **PPADFBIC, PPADFBIT, ARSTAT**

ARMATR(x, p, q, pt, pmax)

PURPOSE

- Computes recursive least squares estimates of an ARMA(p, q) + TR(pt) model and reports the estimated coefficients, standard errors and t-ratios. The recursive least squares estimates are asymptotically equivalent to the Gaussian maximum likelihood estimates and employ the three stages of the Hannan–Rissanen recursion. The third stage is iterated until stability is achieved.

FORMAT

- `{d,st,trat,vmat} = armatr(x,p,q,pt,pmax);`

INPUTS

- `x` dependent variable (nobs $\times$ 1)
- `p` specified AR lag (`p` $\geq$`0`)
- `q` specified MA lag (`q` $\geq$`0`)
- `pt` specified TR degree (`pt` $\geq$`-1`)
- `pmax` AR lag for long AR in first stage (`pmax` ≥ 0)

OUTPUTS

- `d` $(p{+}q{+}pt{+}1)$-vector of estimated coefficients
- `st` standard errors of the parameter estimates
- `trat` t-ratios of the parameter estimates
- `vmat` variance matrix of the parameter estimates

EXAMPLE

/* Estimation of an ARMA(p, q) + TR(pt) by recursive least squares */
```
{d,st,trat,vmat} = armatr(x,p,q,pt, pmax);
``coefficients ='' d'; ``st. errors ='' st'; ``t-ratios ='' trat';
```

RELATED PROCEDURES

- **ARMATRA, ARMATR2, ARMAMAYY, ARMAMAY, ARMASTAT, ARMASTA3, ARMAORD**

REMARKS

1. The AR lag length pmax in the first stage of the recursion could be set using the AIC criterion (see **ARMATRA.PRG**).

2. When the roots of the characteristic polynomial of the estimated coefficients are unstable or close to unstable the third stage of the Hannan–Rissanen recursion is unstable. When the roots $|\lambda| > 0.95$ the third stage of the recursion is not followed and the second stage estimates are reported instead. A message is sent to the screen in this event.

REFERENCE

Hannan, E. J. & J. Rissanen (1982) "Recursive Estimation of Mixed Autoregressive Moving Average Order," *Biometrika*, **69**: 81-94.

ARMATRA(x, p, q, pt, pmax)

PURPOSE

- Computes recursive least squares estimates of an ARMA(p, q) + TR(pt) model and reports the estimated coefficients, standard errors and t-ratios. The recursive least squares estimates are asymptotically equivalent to the Gaussian maximum likelihood estimates and employ the three stages of the Hannan–Rissanen recursion. The third stage is iterated until stability is achieved. The AIC criterion is used to set the AR lag length and TR trend degree in the first stage of the recursion.

FORMAT

- `{d,st,trat,vmat} = armatra(x,p,q,pt,pmax);`

INPUTS

- `x` dependent variable (nobs $\times$ 1)
- `p` specified AR lag (`p` $\geq$ `0`)
- `q` specified MA lag (`q` $\geq$ `0`)
- `pt` specified TR degree (`pt` $\geq$ `-1`)
- `pmax` maximum AR lag in long AR in first stage (`pmax` $>$ `0`)

OUTPUTS

- `d` $(p{+}q{+}pt{+}1)$-vector of estimated coefficients
- `st` standard errors of the parameter estimates
- `trat` t-ratios of the parameter estimates
- `vmat` variance matrix of the parameter estimates

EXAMPLE

/* Estimation of an ARMA(p, q) + TR(pt) by recursive least squares */

```
{d,st,trat,vmat} = armatra(x,p,q,pt,pmax);
``coefficients =`` d'; ``st. errors ='' st'; ``t-ratios ='' trat';
```

RELATED PROCEDURES

- **ARMATR, ARMATR2, ARMAMAYY, ARMAMAY, ARMASTAT, ARMASTA3, ARMAORD**

REMARKS

1. The AR lag length in the first stage of the recursion is determined by the AIC criterion.

2. When the roots of the characteristic polynomial of the estimated coefficients are unstable or close to unstable the third stage of the Hannan–Rissanen recursion is unstable. When the roots $|\lambda| > 0.95$ the third stage of the recursion is not followed and the second stage estimates are reported instead. A message is sent to the screen in this event.

REFERENCE

Hannan, E. J. & J. Rissanen (1982) "Recursive Estimation of Mixed Autoregressive Moving Average Order," *Biometrika*, **69**: 81–94.

ARMATR2(x, p, q, pt, pmax)

PURPOSE

- Computes two stage recursive least squares estimates of an ARMA(p, q) + TR(pt) model and reports the estimated coefficients, standard errors and t-ratios. The two stage recursive estimates are consistent but not asymptotically equivalent to the Gaussian maximum likelihood estimates.

FORMAT

- `{d,t,st,trat,vmat} = armatr2(x,p,q,pt,pmax);`

INPUTS

- `x` dependent variable (nobs × 1)
- `p` specified AR lag (`p` $\geq$ `0`)
- `q` specified MA lag (`q` $\geq$ `0`)
- `pt` specified TR degree (`pt` $\geq$ `-1`)
- `pmax` maximum AR lag in long AR in first stage (`pmax` $\geq$ 0)

OUTPUTS

- `d` (p+q+pt+1)-vector of estimated coefficients
- `st` standard errors of the parameter estimates
- `trat` t-ratios of the parameter estimates
- `vmat` variance matrix of the parameter estimates

EXAMPLE

/* ARMA(p, q) + TR(pt) using two stage recursive least squares */
```
{d,st,trat,vmat} = armatr2(x,p,q,pt,pmax);
``coefficients ='' d'; ``st. errors ='' st'; ``t-ratios ='' trat';
```

RELATED PROCEDURES

- **ARMATR, ARMAMAYY, ARMAMAY, ARMASTAT, ARMASTA3, ARMAORD**

REMARK

1. The AR lag length pmax in the first stage of the recursion is set using the AIC criterion.

REFERENCE

Hannan, E. J. & J. Rissanen (1982) "Recursive Estimation of Mixed Autoregressive Moving Average Order," *Biometrika*, **69**: 81–94.

ARMABIC3(x, pmax, tmax)

PURPOSE

- Performs model selection in the ARMA(p, q) + TR(pt) class using the BIC criterion in conjunction with recursive least squares estimation by means of the Hannan–Rissanen (1982) recursion with the Kavalieris (1991) modification for the residual variance estimate. BIC(p, q) values are returned as well as the selected AR, MA and TR orders, and estimated coefficients, standard errors, t-ratios and covariance matrix. The third stage of the Hannan–Rissanen recursion is iterated until stability is achieved. If the third stage of the recursion is unstable then the second stage estimates are reported.

FORMAT

- `{bic,p,q,t,d,st,trat,vmat,sig2} = armabic3(x,pmax,qmax,tmax);`

INPUTS

- `x` dependent variable (nobs × 1)
- `pmax` AR lag for long AR in first stage (`pmax` $\geq$ `0`)
- `qmax` maximum MA lag (`qmax` $\geq$ `0`)
- `tmax` maximum TR degree (`tmax` $\geq$ `-1`)

OUTPUTS

- `bic` (pmax+1) × (qmax+1) matrix of bic values
- `p` selected AR order
- `q` selected MA order
- `t` selected TR degree ($t \geq -1$)
- `d` $(p{+}q{+}pt{+}1)$-vector of estimated coefficients
- `st` standard errors of the parameter estimates
- `trat` t-ratios of the parameter estimates
- `vmat` variance matrix of the parameter estimates
- `sig2` estimated equation error variance

EXAMPLE

/* Model search in ARMA(p, q) + TR(pt) class by recursive least squares estimation */

```
{bic,p,q,t,d,st,trat,vmat,sig2} = armabic3(x,pmax,qmax,tmax);
``bic values ='' bic; ``selected AR lag ='' p;
    ``selected MA lag ='' q;
``coefficients =`` d'; ``st. errors ='' st'; ``t-ratios ='' trat';
```

REMARK

1. Use _variance = 1 or _variance = 2 to use a maximum likelihood estimator for the variance when calculating the pic/picu. See page 63 for more information.

RELATED PROCEDURES

- **ARMABIC2, ARMATR2, ARMAMAYY, ARMAMAY, ARMASTAT, ARMASTA3, ARMAORD**

REMARKS

1. The AR lag length pmax in the first stage of the recursion could be set using the AIC criterion (use **ARORD.PRG**).

2. When the roots of the characteristic polynomial of the estimated coefficients are unstable or close to unstable the third stage of the Hannan–Rissanen recursion is unstable. When the roots $|\lambda| > 0.95$ the third stage of the recursion is not followed and the second stage estimates are reported instead. A message is sent to the screen in this event.

REFERENCES

Hannan, E. J. & J. Rissanen (1982) "Recursive Estimation of Mixed Autoregressive Moving Average Order," *Biometrika*, **69**: 81–94.

Kavalieris, L. (1991) "A Note on Estimating Autoregressive-Moving Average Order," *Biometrika*, **78**: 920–922.

ARMABIC2(x, pmax, qmax, tmax)

PURPOSE

- Performs model selection in the ARMA(p, q) + TR(pt) class using the BIC criterion in conjunction with recursive least squares estimation by means of the Hannan–Rissanen (1982) recursion with the Kavalieris (1991) modification for the residual variance estimate. BIC(p, q) values are returned as well as the selected AR, MA and TR orders, and estimated coefficients, standard errors, t-ratios and covariance matrix.

FORMAT

- `{bic,p,q,t,d,st,trat,vmat,sig2} = armabic2(x,pmax,qmax,tmax);`

INPUTS

- `x` dependent variable(nobs × 1)
- `pmax` AR lag for long AR in first stage (`pmax` ≥ `0`)
- `qmax` maximum MA lag (`qmax` ≥ `0`)
- `tmax` maximum TR degree (`qmax` ≥ `-1`)

OUTPUTS

- `bic` `(pmax+1)`×`(qmax+1)` matrix of bic values
- `p` selected AR order
- `q` selected MA order
- `t` selected TR degree ($t \geq -1$)
- `d` $(p+q+pt+1)$-vector of estimated coefficients
- `st` standard errors of the parameter estimates
- `trat` t-ratios of the parameter estimates
- `vmat` variance matrix of the parameter estimates
- `sig2` estimated equation error variance

EXAMPLE

/* Model search in ARMA(p, q) + TR(pt) class by recursive least squares estimation */

```
{bic,p,q,t,d,st,trat,vmat,sig2} = armabic2(x,pmax,qmax,tmax);
``bic values ='' bic; ``selected AR lag ='' p;
    ``selected MA lag ='' q;
``coefficients ='' d'; ``st. errors ='' st'; ``t-ratios ='' trat';
```

RELATED PROCEDURES

- **ARMABIC3, ARMATR2, ARMAMAYY, ARMAMAY, ARMASTAT, ARMASTA3, ARMAORD**

REMARK

1. The AR lag length pmax in the first stage of the recursion could be set using the AIC criterion (use **ARORD.PRG**).

2. Use _variance = 1 or _variance = 2 to use a maximum likelihood estimator for the variance when calculating the pic/picu. See page 63 for more information.

REFERENCES

Hannan, E. J. & J. Rissanen (1982) "Recursive Estimation of Mixed Autoregressive Moving Average Order," *Biometrika*, **69**: 81–94.

Kavalieris, L. (1991) "A Note on Estimating Autoregressive-Moving Average Order," *Biometrika*, **78**: 920–922.

ARMABC(x, p, q, pt)

PURPOSE

- Performs model selection in the ARMA(p, q) + TR(pt) class using the BIC criterion in conjunction with recursive least squares estimation by means of the Hannan–Rissanen (1982) recursion with the Kavalieris (1991) modification for the residual variance estimate. BIC(p, q) values are returned as well as the selected AR, MA and TR orders, and estimated coefficients, standard errors, t-ratios and covariance matrix. The third stage of the Hannan–Rissanen recursion is iterated until stability is achieved. If the third stage of the recursion is unstable then the second stage estimates are reported. Also graphs BIC surface for an ARMA(p, q) model with maximum lag orders `pmax` > 0 and `qmax` > 0. Additional model selection graphics are returned of the BIC values for the AR order conditional on the selected MA lag and the MA lag conditional on the selected AR lag order.

FORMAT

- `{bic,p,q,t,d,st,trat,vmat} = armabc(x,pmax,qmax,tmax);`

INPUTS

- `x` dependent variable(nobs $\times$ 1)
- `pmax` AR lag for long AR in first stage (`pmax` $\geq$ `0`)
- `qmax` maximum MA lag (`qmax` $\geq$ `0`)
- `tmax` maximum TR degree (`tmax` $\geq$ `-1`)

OUTPUTS

- `bic` (`pmax`+1) $\times$ (`qmax`+1) matrix of bic values
- `p` selected AR order
- `q` selected MA order
- `t` selected TR degree ($t \geq -1$)
- `d` $(p{+}q{+}pt{+}1)$-vector of estimated coefficients
- `st` standard errors of the parameter estimates
- `trat` t-ratios of the parameter estimates
- `vmat` variance matrix of the parameter estimates

EXAMPLE

/* Model search in ARMA(p, q) + TR(pt) class by recursive least squares estimation*/

```
{bic,p,q,t,d,st,trat,vmat} = armabc(x,pmax,qmax,tmax);
``bic values ='' bic; ``selected AR lag ='' p;
    ``selected MA lag =''q;  ``selected TR degree ='' t;
``coefficients ='' d'; ``st. errors ='' st'; ``t-ratios ='' trat';
```

RELATED PROCEDURES

- **ARMABIC2, ARMABIC3, ARMATR2, ARMAMAYY, ARMAMAY, ARMASTAT, ARMASTA3, ARMAORD**

REMARKS

1. The AR lag length pmax in the first stage of the recursion could be set using the AIC criterion (use **ARORD.PRG**).
2. When the roots of the characteristic polynomial of the estimated coefficients are unstable or close to unstable the third stage of the Hannan–Rissanen recursion is unstable. When the roots $|\lambda| > 0.95$ the third stage of the recursion is not followed and the second stage estimates are reported instead. A message is sent to the screen in this event.
3. Use _variance = 1 or _variance = 2 to use a maximum likelihood estimator for the variance when calculating the pic/picu. See page 63 for more information.

REFERENCES

Hannan, E. J. & J. Rissanen (1982) "Recursive Estimation of Mixed Autoregressive Moving Average Order," *Biometrika*, **69**: 81–94.

Kavalieris, L. (1991) "A Note on Estimating Autoregressive-Moving Average Order," *Biometrika*, **78**: 920–922.

GRFBICPQ(pmax, qmax, bic)

PURPOSE

- Graphs BIC surface for an ARMA(p, q) model with maximum lag orders `pmax` > 0 and `qmax` > 0. Requires input matrix of BIC values for ARMA models which can be obtained from the procedure **ARMABIC3.PRG**.

FORMAT

- `grfbicpq(pmax,qmax,bic);`

INPUTS

- `pmax` maximum AR lag (`pmax` $\geq$ `0`)
- `qmax` maximum MA lag (`qmax` $\geq$ `0`)

OUTPUTS

- graphical surface of BIC values

EXAMPLE

/* Search in ARMA(p, q) + TR(pt) class and graphical display of BIC(p, q) values */

```
{bic,p,q,b,st,trat,vmat} = armabic3(x,pmax,qmax,tmax);
    grfbicpq(pmax,qmax,bic);
```

REMARK

1. Use _variance = 1 or _variance = 2 to use a maximum likelihood estimator for the variance when calculating the pic/picu. See page 63 for more information.

RELATED PROCEDURES

- **ARBC, ARPC, GRFBICP, GRFBICQ, GRFBICL, GRFBICT**

LRVAR.SRC

Long-Run Variance and Spectral Density

Estimation Procedures and Graphics

SPECARMA(a, b, sig2, x)

PURPOSE

- Computes the spectral density of an ARMA(p, q) process with AR coefficients carried in the vector a and MA coefficients carried in the vector b. The spectrum is computed at the points specified in the input vector x.

FORMAT

- `spectrum = specarma(a,b,sig2,x);`

INPUTS

- `a(1xp)` vector of autoregressive coefficients in $a[1] + a[2]L + \cdots + a[p]L^p$
- `b(1xq)` vector of moving-average coefficients in $b[1] + b[2]L + \cdots + b[q]L^q$
- `sig2` error variance
- `x` $(1 \times n)$ vector of frequencies to evaluate spectrum

OUTPUTS

- spectrum = $(n \times 1)$ vector of values of spectral density at x

EXAMPLE

/* Computation of spectrum of an ARMA(p, q) process */
```
spectrum = specarma(a,b,sig2,x);
``spectral density at x ='' spectrum;
```

RELATED PROCEDURES

- **SARMAGRF, LRVARWX, LRVARO, DSPECTRA**

SARMAGRF(a, b, sig2, x)

PURPOSE

- Computes the spectral density of an ARMA(p, q) process with AR coefficients carried in the vector a and MA coefficients carried in the vector b. The spectrum is computed at the points specified in the input vector x and then graphed.

FORMAT

- `spectrum = sarmagrf(a,b,sig2,x);`

INPUTS

- `a(1xp)` vector of autoregressive coefficients in $a[1] + a[2]L + \cdots + a[p]L^p$
- `b(1xq)` vector of moving-average coefficients in $b[1] + b[2]L + \cdots + b[q]L^q$
- `sig2` error variance
- `x` $(1 \times n)$ vector of frequencies to evaluate spectrum

OUTPUTS

- spectrum = $(n \times 1)$ vector of values of spectral density at x graph of spectral density

EXAMPLE

/* Computation of spectrum of an ARMA(p, q) process */
```
spectrum = sarmagrf(a,b,sig2,x);
``spectral density at x ='' spectrum;
```

RELATED PROCEDURES

- **SPECARMA, LRVARWX, LRVARO, DSPECTRA**

SPECWX(x, pmax, qmax, _kernel, wx)

PURPOSE

- Computes the spectral density of a time series by several different methods.
- Uses recursive ARMA model selection methods (Hannan–Rissanen, 1982) to find the most suited parametric model in the ARMA class and uses the 3-stage Hannan–Rissanen recursion to estimate the coefficients of this model. A parametric spectral density estimate is then constructed from these coefficient estimates. Both 2-stage and 3-stage estimates from this procedure are returned.
- Uses a nonparametric kernel estimate constructed with a data-based bandwidth.
- Uses the Lee–Phillips (1993) procedure to find the best model for prefiltering the data — as in (1) above — and then employs a kernel procedure with a data-based bandwidth to nonparametrically estimate the spectral density of the residuals. The nonparametric kernel estimate is then recolored using the inverse of the model chosen to prefilter the data.
- Uses the Andrews–Monahan (1992) AR prefiltered and recolored spectrum estimate.

FORMAT

- `{g1,g2,g3,g4,g5} = specwx(x,pmax,qmax,_kernel,wx);`

INPUTS

- `x` $(T \times 1)$ vector of times series data
- `pmax` maximum AR lag used in model selection
- `qmax` maximum MA lag used in model selection
- `_kernel` 1 for quadratic spectral kernel estimate; 2 for Parzen kernel estimate
- `wx` $(1 \times n)$ vector of frequencies to evaluate spectrum

OUTPUTS

Five $(n \times 1)$ vectors of spectral density estimates at x:

- `g1` parametric ARMA$(\hat{p}, \hat{q})$ estimate obtained from 2-stage Hannan–Rissanen recursion
- `g2` parametric ARMA$(\hat{p}, \hat{q})$ estimate obtained from 3-stage Hannan–Rissanen recursion

- `g3` nonparametric kernel estimate using a data-based bandwidth choice
- `g4` Lee–Phillips (1993) model-selected ARMA prefiltered and recolored kernel estimate
- `g5` Andrews–Monahan (1992) AR prefiltered kernel estimate

EXAMPLE

```
/* Estimation of spectrum of an ARMA(p,q) process */
n = 4;
_kernel = 1;
wx = seqa(0,pi/n,n+1)';
pmax = 5;
qmax = 1;
{g1,g2,g3,g4,g5} = specwx(x,pmax,qmax,_kernel,wx);
``ARMA 2-stage estimate ='' g1';
``ARMA 3-stage estimate ='' g2';
``QS kernel estimate ='' g3';
``Lee-Phillips estimate ='' g4';
``Andrews-Monahan estimate ='' g5';
```

RELATED PROCEDURES

- **LRVARWX, LRVRO, DSPECTRA, ANDREWSM**

REMARKS

1. This program returns spectrum over $[0, \pi]$ with the $(n \times 1)$ vector $wx =$ frequencies evaluated.
2. The data-based bandwidth choice is based on obtaining the optimal estimate of the spectral density at the origin. This choice is not optimal at other frequencies and will therefore affect estimates g3, g4 and g5 in this regard.

REFERENCES

Andrews, D. W. K. & C. Monahan (1991) "An Improved Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimator," *Econometrica*, **60**: 953–966.

Lee, C. C. & P. C. B. Phillips (1993) "An ARMA-Prefiltered Estimator of the Long Run Variance with an Application to Unit Root Tests," Cowles Foundation, Yale University, mimeographed.

SPWXGRF(x, pmax, qmax, _kernel, wx)

PURPOSE

- Graphs estimates of the spectral density of a time series obtained by the different methods used in the procedure **LRVRWX.PRC**.
  - Parametric estimate based on the ARMA model selected and the coefficient estimates obtained in a 2-stage Hannan–Rissanen recursion.
  - Parametric estimate based on the ARMA model selected and the coefficient estimates obtained in a 3-stage Hannan–Rissanen recursion.
  - A nonparametric kernel estimate constructed with a data-based bandwidth.
  - The Lee–Phillips (1993) procedure of finding the best ARMA model for prefiltering the data, employing a nonparametric kernel estimate of the spectrum of the residuals, and subsequently recoloring the spectral estimate.
  - The Andrews–Monahan (1992) AR prefiltered and recolored spectrum estimate.

FORMAT

- `{g1,g2,g3,g4,g5} = spwxgrf(x,pmax,qmax,_kernel,wx);`

INPUTS

- `x` $(T \times 1)$ vector of times series data
- `pmax` maximum AR lag used in model selection
- `qmax` maximum MA lag used in model selection
- `_kernel` 1 for quadratic spectral kernel estimate;2 for Parzen kernel estimate
- `wx` $(1 \times n)$ vector of frequencies to evaluate spectrum

OUTPUTS

Five $(n \times 1)$ vectors of spectral density estimates at x:

- `g1` parametric ARMA$(\hat{p}, \hat{q})$ estimate obtained from 2-stage Hannan–Rissanen recursion
- `g2` parametric ARMA$(\hat{p}, \hat{q})$ estimate obtained from 3-stage Hannan–Rissanen recursion
- `g3` nonparametric kernel estimate using a data-based bandwidth choice

- g4 Lee–Phillips (1993) model-selected ARMA prefiltered and
- g5 Andrews–Monahan(1992) AR prefiltered kernel estimate

and sequential graphs of these spectral estimates and a composite graph of all the estimates together.

EXAMPLE

```
/* Estimation of spectrum of an ARMA(p,q) process */
n = 40;
_kernel = 1;
wx = seqa(0,pi/n,n+1)';
pmax = 5; qmax = 1;
{g1,g2,g3,g4,g5} = spwxgrf(x,pmax,qmax,_kernel,wx);
''ARMA 2-stage estimate ='' g1';
''ARMA 3-stage estimate ='' g2';
''QS kernel estimate ='' g3';
''Lee-Phillips estimate ='' g4';
''Andrews-Monahan estimate ='' g5';
```

RELATED PROCEDURES

- **SPECWX, LRVRWX, LRVARWX, LRVRO, DSPECTRA, ANDREWSM**

REMARKS

1. This program returns spectrum and supplies graphs over $[0, \pi]$ with the $(n \times 1)$ vector $wx =$ frequencies evaluated.
2. The data-based bandwidth choice is based on obtaining the optimal estimate of the spectral density at the origin. This choice is not optimal at other frequencies and will therefore affect estimates g3, g4 and g5 in this regard.

REFERENCES

Andrews, D. W. K. & C. Monahan (1991) "An Improved Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimator," *Econometrica*, **60**: 953–966.

Lee, C. C. & P. C. B. Phillips (1993) "An ARMA-Prefiltered Estimator of the Long Run Variance with an Application to Unit Root Tests," Cowles Foundation, Yale University, mimeographed.

PPLEE(x, pmax, qmax, _kernel)

PURPOSE

- Computes the long-run variance of a time series by the Lee–Phillips (1993) method: uses recursive ARMA model selection methods (Hannan–Rissanen, 1982) to find the most suited parametric model in the ARMA class for prefiltering the data and then employs a kernel procedure with a data-based bandwidth to nonparametrically estimate the spectrum of the residuals at the origin. The nonparametric kernel estimate is then recolored using the inverse of the model chosen to prefilter the data.

FORMAT

- `lrvar = pplee(x,pmax,qmax,_kernel);`

INPUTS

- `x` $(T \times 1)$ vector of times series data
- `pmax` maximum AR lag used in model selection
- `qmax` maximum MA lag used in model selection
- `_kernel` 1 for quadratic spectral kernel estimate
  - 2 for Parzen kernel estimate

OUTPUTS

- `lrvar` Lee–Phillips model-selected ARMA prefiltered and recolored kernel estimate

EXAMPLE

/* Estimation of the long-run variance of an ARMA(p, q) process */

```
_kernel = 1; pmax = 5; qmax = 1;
lrvar = pplee(x,pmax,qmax,_kernel)
``Lee-Phillips estimate ='' lrvar;
```

RELATED PROCEDURES

- **SPECWX, LRVRWX, LRVARWX, LRVRO, DSPECTRA, ANDREWSM, AMLRVR**

REFERENCE

Lee, C. C. & P. C. B. Phillips (1993) "An ARMA-Prefiltered Estimator of the Long Run Variance with an Application to Unit Root Tests," Cowles Foundation, Yale University, mimeographed.

AMLRVR(x, _kernel)

PURPOSE

- Computes the long-run variance of a time series by the Andrews–Monahan (1992) method of AR prefiltering and recoloring combined with a data-based kernel procedure.

FORMAT

- `lrvr = amlrvr(x,_kernel);`

INPUTS

- `x` $(T \times 1)$ vector of times series data
- `_kernel` 1 for quadratic spectral kernel estimate 2 for Parzen kernel estimate

OUTPUTS

- `lrvar` Andrews–Monahan AR prefiltered and recolored kernel estimate

EXAMPLE

/* Estimation of the long-run variance of an ARMA(p, q) process */

```
_kernel = 1;
lrvr = amlrvr(x, _kernel)
''Andrews-Monahan estimate ='' lrvr;
```

RELATED PROCEDURES

- **SPECWX, LRVRWX, LRVARWX, LRVRO, DSPECTRA, PPLEE**

REFERENCE

Andrews, D. W. K. & C. Monahan (1991) "An Improved Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimator," *Econometrica*, **60**: 953–966.

LRVRO(x, pmax, qmax, _kernel)

PURPOSE

- Computes the long-run variance of a time series by several different methods:
  - Uses recursive ARMA model selection methods (Hannan–Rissanen, 1982) to find the most suited parametric model in the ARMA class and uses the 3-stage Hannan–Rissanen recursion to estimate the coefficients of this model. A parametric spectral density estimate is then constructed from these coefficient estimates. Both 2-stage and 3-stage estimates from this procedure are returned.
  - Uses a nonparametric kernel estimate constructed with a data-based bandwidth.
  - Uses the Lee–Phillips (1993) procedure to find the best model for prefiltering the data — as in (1) above — and then employs a kernel procedure with a data-based bandwidth to nonparametrically estimate the spectral density of the residuals. The nonparametric kernel estimate is then recolored using the inverse of the model chosen to prefilter the data.
  - Uses the Andrews–Monahan (1992) AR prefiltered and recolored spectrum estimate.

FORMAT

- `{g1,g2,g3,g4,g5} = lrvro(x,pmax,qmax,_kernel);`

INPUTS

- `x` $(T \times 1)$ vector of times series data
- `pmax` maximum MA lag used in model selection
- `_kernel` 1 for quadratic spectral kernel estimate; 2 for Parzen kernel estimate

OUTPUTS

Five long-run variance estimates:

- `g1` parametric ARMA$(\hat{p}, \hat{q})$ estimate obtained from 2-stage Hannan–Rissanen recursion
- `g2` parametric ARMA$(\hat{p}, \hat{q})$ estimate obtained from 3-stage Hannan–Rissanen recursion
- `g3` nonparametric kernel estimate using a data-based bandwidth choice

- `g4` Lee–Phillips (1993) model-selected ARMA prefiltered and recolored kernel estimate
- `g5` Andrews–Monahan (1992) AR prefiltered kernel estimate

EXAMPLE

/* Estimation of spectrum of an ARMA(p, q) process */
```
_kernel = 1; pmax = 5; qmax = 1;
{g1,g2,g3,g4,g5} = lrvro(x,pmax,qmax,_kernel);
''ARMA 2-stage estimate ='' g1; ''ARMA 3-stage estimate ='' g2;
''QS kernel estimate ='' g3; ''Lee-Phillips estimate ='' g4;
''Andrews-Monahan estimate ='' g5;
```

RELATED PROCEDURES

- **LRVRWX, DSPECTRA, ANDREWSM, AMLRV, PPLEE**

REFERENCES

Andrews, D. W. K. & C. Monahan (1991) "An Improved Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimator," *Econometrica*, **60**: 953–966.

Lee, C. C. & P. C. B. Phillips (1993) "An ARMA-Prefiltered Estimator of the Long Run Variance with an Application to Unit Root Tests," Cowles Foundation, Yale University, mimeographed.

PPZAZT(x, pmax, qmax, pt, _kernel, sel)

PURPOSE

- This procedure calculates the Phillips (1987) & Phillips–Perron (1988) test statistics $Z(a)$ and $Z(t)$ using automatic data-based estimates of the long-run variance. The lrvar estimates that can be employed are the Lee–Phillips (1993) ARMA model selected prefiltered and recolored kernel estimate and the Andrews–Monahan (1992) AR prefiltered and recolored kernel estimate.

FORMAT

- `{alpha,za,zt} = ppzazt(x,pmax,qmax,pt,_kernel,sel);`

INPUTS

- `x` $(T \times 1)$ vector of times series data
- `pmax` maximum AR lag used in model selection
- `qmax` maximum MA lag used in model selection
- `pt` trend degree in regression ($pt \geq -1$)
- `_kernel` 1 for quadratic spectral kernel estimate; 2 for Parzen kernel estimate
- `sel` 1 for Lee–Phillips lrvar estimate; 2 for Andrews–Monahan lrvar estimate

OUTPUTS

- `alpha` estimated AR coefficient
- `za` $Z(a)$ statistic
- `zt` $Z(t)$ statistic

EXAMPLE

/* $Z(a)$ and $Z(t)$ unit root tests for an ARIMA(0,1,1) process */

```
_kernel = 1; pmax = 5; qmax = 1; pt = -1; sel = 1;
{alpha,za,zt} = ppzazt(x,pmax,qmax,pt,_kernel,sel);
``AR coefficient ='' alpha;
``Z(a) statistic ='' za;
``Z(t) statistic ='' zt;
```

RELATED PROCEDURES

- **AMLRV, PPLEE**

REMARK

1. The model selection principles associated with the estimation of the Lee–Phillips estimate of the long-run variance are explained in Phillips–Ploberger (1994) where a posterior odds test for a unit root is developed.

REFERENCES

Andrews, D. W. K. & C. Monahan (1991) "An Improved Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimator," *Econometrica*, **60**: 953–966.

Lee, C. C. & P. C. B. Phillips (1993) "An ARMA-Prefiltered Estimator of the Long Run Variance with an Application to Unit Root Tests," Cowles Foundation, Yale University, mimeographed.

Phillips P. C. B. and W. Ploberger (1994) "Posterior Odds Testing for a Unit Root with Data Based Model Selection," *Econometric Theory,* **10**(3/4): 744–808.

BAYES.SRC

Procedures for Bayesian Posterior Analysis

of an AR(k) + TR(pt) Model and

for the Bayesian Analysis of

Cointegrating Regression Residuals

BARTR(x, nstd, npoints, klag, pt)

PURPOSE

- Computes posterior distributions for the long-run autoregressive parameter (i.e., the sum of the coefficients) in an AR(k) + TR(pt) model. The procedure uses an invariant Jeffreys' prior as well as a uniform prior on the coefficients in the regression (see Phillips, 1991a, 1991b). Posterior probabilities for the nonstationary region $[1.0, \infty)$ and the near nonstationary region $[0.975, \infty)$ are calculated and printed out, together with the least squares estimates and their standard errors. Graphics programs are called to graph the posterior distributions and the figures are output with user supplied titles and labels. The Laplace method of approximation (see Phillips, 1983, 1991b) and Tierney & Kadane, 1986) is used to calculate the marginal posterior distributions of the long-run autoregressive coefficient. Numerical integration is performed to renormalize the Bayesian posteriors.

FORMAT

- `{pdf1,pdf2,points} = bartr(x,nstd,npoints,klag,pt);`

INPUTS

- `x` time series data (nobs $\times$ 1)
- `npoints` number of points at which densities are evaluated (usually `npoints` = 500 is sufficient)
- `nstd` number of asymptotic standard deviations on either side of the least squares estimate: defines the support of the posterior distribution (usually $3.5 \leq$ `nstd` ≤ 5.0)
- `klag` number of lags in the autoregression (`klag` ≥ 1)
- `pt` degree of deterministic trend (`pt` ≥ -1 ; -1 is case of no intercept)

OUTPUTS

- `pdf1` posterior density for a uniform prior
- `pdf2` posterior density for a Jeffreys' prior
- `points` vector of points where the densities are evaluated

EXAMPLE

/* Computation of posterior densities for AR(k) + TR(pt) model */

```
{pdf1,pdf2,points} = bartr(x,nstd,npoints,klag,pt);
    ``posterior density ordinates at `points'`` = pdf1~df2;
```

RELATED PROCEDURES

- **BARTR2, BEC0, BEC3, DGRAPH**

REMARKS

1. Need to set `klag`≥ 1 and `pt` > -1.
2. If the **GAUSS** procedure "INTSIMP" fails try reducing the input parameter nstd. This will contract the support over which the posterior density is being computed and facilitate the use of "INTSIMP."

REFERENCES

Phillips, P. C. B. (1983) "Marginal Densities of Instrumental Variables Estimators in the General Single Equation Case," *Advances in Econometrics,* **2**: 1–24.

Phillips, P. C. B. (1991a) "Bayesian Routes and Unit Roots: de Rebus Prioribus Semper est Disputandum," *Journal of Applied Econometrics*, **6**(4): 435–474.

Phillips, P. C. B. (1991b) "To Criticize the Critics: an Objective Bayesian Analysis of Stochastic Trends," *Journal of Applied Econometrics*, **6**(4): 333–364.

Tierney, L. & J. B. Kadane (1986) "Accurate Approximations for Posterior Moments and Marginal Densities," *Journal of the American Statistical Association*, **81:** 82–86.

BARTR2(x, nstd, npoints, klag, pt)

PURPOSE

- Computes posterior distributions for the long-run autoregressive parameter (i.e., the sum of the coefficients) in an AR(k) + TR(pt) model using an invariant Jeffreys' prior and a uniform prior on the coefficients in the regression (see Phillips, 1991a, 1991b). The procedure returns results for AR(1) + TR(pt) and AR(k) + TR(pt) models. The posterior densities for these two models can then be graphed on the same figure — see the procedure **DGRAPH**.
- Laplace approximations are used as in the procedure **BARTR**.

FORMAT

- `{pdf1,pdf2,pdf3,points} = bartr(x,nstd,npoints,klag,pt);`

INPUTS

- `x` time series data (nobs $\times$ 1)
- `npoints` number of points at which densities are evaluated (usually `npoints` = 500 is sufficient)
- `nstd` number of asymptotic standard deviations on either side of the least squares estimate: defines the support of the posterior distribution (usually $3.5 \leq$ `nstd` ≤ 5.0)
- `klag` number of lags in the autoregression (`klag` ≥ 1)
- `pt` degree of deterministic trend (`pt` ≥ -1 ; -1 is case of no intercept)

OUTPUTS

- `pdf1` posterior density: model = `AR(1) + TR(pt)` with a Jeffreys' prior
- `pdf2` posterior density model = `AR(klag) + TR(pt)` for a Jeffreys' prior
- `pdf3` posterior density model = `AR(klag) + TR(pt)` for a uniform prior
- `points` vector of points where the densities are evaluated

EXAMPLE

```
/* Computation of posterior densities for AR(k) + TR(pt) model */
{pdf1,pdf2,pdf3,points} = bartr(x,nstd,npoints,klag,pt);
''posterior density ordinates'' = pdf1~pdf2~pdf3;
```

RELATED PROCEDURES

- **BARTR, BEC0, BEC3, DGRAPH**

REMARKS

1. Need to set `klag` $\geq$ `1` and `pt` $>$ `-1`.

2. If the **GAUSS** procedure "INTSIMP" fails try reducing the input parameter nstd. This will contract the support over which the posterior density is being computed and facilitate the use of "INTSIMP."

3. Some empirical illustrations with this procedure are reported in Phillips (1991a,b, 1992).

REFERENCES

Phillips, P. C. B. (1991a) "Bayesian Routes and Unit Roots: de Rebus Prioribus Semper est Disputandum," *Journal of Applied Econometrics*, **6**(4): 435–474.

Phillips, P. C. B. (1991b) "To Criticize the Critics: an Objective Bayesian Analysis of Stochastic Trends," *Journal of Applied Econometrics*, **6**(4): 333–364.

Phillips, P. C. B. (1992) "The Long-Run Australian Consumption Function Reexamined: an Empirical Exercise in Bayesian Inference," Ch. 11 in C. P. Hargreaves (ed.), *Macroeconomic Modelling of the Long Run.* Aldershot: Edward Elgar, pp. 287–322.

DGRAPH(zx, zy, name, dates, klag, nobs, ngraph)

PURPOSE

- Graphics procedure for computing the posterior densities for the long-run autoregressive parameter in an AR(k) + TR(pt) model. The procedure has options to compute the posterior densities for the same model with different priors (a Jeffreys' prior and a uniform prior on the coefficients in the regression) or for AR models with different lag lengths. This facilitates comparisons of posteriors across priors and across models.

FORMAT

- `dgraph(zx,zy,name,dates,klag,nobs,ngraph);`

INPUTS

- `zx` vector of points where the densities are to be evaluated
- `zy` matrix of data of density ordinates (`zx` $\times$ `cols(zy)`)
- `name` title for figure giving name of time series (string)
- `dates` dates of data (string)
- `klag` number of lags in the longest autoregression (`klag` ≥ 1)
- `nobs` number of observations in the time series
- `ngraph` select number of posterior densities on figure:
  - $= 0$ for just AR(k) results
  - $= 1$ for both AR(1) and AR(k) results

OUTPUTS

- Graphs with user supplied labels and legends that record the parameters of the models being used

EXAMPLE

/* Graphing the posterior densities for an AR(k) + TR(pt) model with $k = 1$ and $k > 1$ */

```
name = ``Test Data from an AR(2) + trend model''; @ data @
dates = ``1:n simulated'';
seed = 43125;
nstd = 4.0;
npoints = 500; @ number of points to evaluate densities @
klag = 4; @ set AR lag parameter :  use klag ge 1 @
pt = 2; @ set trend degree ge -1 @
```

```
/* input parameters are now set */
/* now generate data */
a = 0.85; b = 0.45; c = 0.5; d = 0.025; n = 150;
x = rndns(n,1,seed); x = recserar(x,0,b); x = recserar(x,0,a)
    + d*seqa(1,1,n) + c*ones(n,1);
{pdf1, pdf2, pdf3, points} = bartr2(x, nstd, npoints,
     klag, pt);
zy = pdf1~pdf2~pdf3; zx = points;
dgraph(zx,zy,name,dates,klag,n,1);
```

RELATED PROCEDURES

- **BARTR, BARTR2, BEC0, BEC3**

REMARK

1. Need to set `klag` ≥ 1 and `pt` > -1.

DATGRAPH(x, scr, name)

PURPOSE

- Graphs time series data.

FORMAT

- `datgraph(x,scr,name);`

INPUTS

- `x` vector of time series data
- `scr` screen selector: scr
  - = 1 sends graphs to screen
  - = 0 otherwise
- `name` title for figure giving name of time series (string)

OUTPUTS

- Data graph with user supplied title

EXAMPLE

```
/* Graphing simulated data */
name = ``Test Data:  iid N(0,1)'';
seed = 43125;
n = 150;
x = rndns(n,1,seed);
datgraph(x,1,name);
```

RELATED PROCEDURE

- **GRAPH**

CBARE(x, nstd, npoints, klag, pt)

PURPOSE

- This program is designed to be used with the residuals from a cointegrating regression. It computes posterior distributions for the long-run autoregressive parameter (i.e., the sum of the coefficients) in an AR(k) model for $k = 1$ and $k > 1$ using a Jeffreys' prior (for $k = 1$), and e-priors (see Phillips, 1992, and Zivot & Phillips, 1994) and uniform prior (for $k > 1$) on the coefficients in the autoregression. The procedure returns results for both AR(1) and AR(k) models. The posterior densities for these two models can then be graphed on the same figure — see the procedure **DGRAPHE**.
- Laplace approximations are used to compute the marginal posterior densities. Numerical integration is used to renormalize the Bayesian posteriors.

FORMAT

- `{pdf1,pdf2,pdf3,pts} = cbare(x,nstd,npoints,klag);`

INPUTS

- `x` residuals from a cointegrating regression (nobs $\times$ 1)
- `npoints` pointsnumber of points at which densities are evaluated (usually `npoints` = 500 is sufficient)
- `nstd` number of asymptotic standard deviations on either side of the least squares estimate: defines the support of the posterior distribution (usually $3.5 \leq$ `nstd` ≤ 7.0)
- `klag` number of lags in the autoregression (`klag` ≥ 1)

OUTPUTS

- `pfd1` posterior density: model = AR(1) with a Jeffreys' prior
- `pdf2` posterior density: model = AR(klag) with an e-prior as in Phillips (1992) and Zivot & Phillips (1993)
- `pdf3` posterior density: model = AR(klag) with a uniform prior
- `pts` vector of points where the densities are evaluated

EXAMPLE

/* Computation of posterior densities for cointegrating regression residuals using AR(1) and AR(klag) models for the residuals /*

```
{pdf1,pdf2,pdf3,points} = cbare(sdat,nstd,npoints,klag);
‘‘posterior density ordinates’’ = pdf1~pdf2~pdf3;
```

RELATED PROCEDURES

- **BARTR, BMAT3, DGRAPHE**

REMARKS

1. Need to set `klag` ≥ 1.
2. If the **GAUSS** procedure "INTSIMP" fails try reducing the input parameter nstd. This will contract the support over which the posterior density is being computed and facilitate the use of "INTSIMP."
3. The procedure uses a product (of diagonal elements) formulation for the e-prior with an exponential weighting to ensure integrability and sets the exponent in the e-prior by formula which determines the point $(= 1 + e)$ at which the e-prior attains its maximum: see Phillips (1992) and Zivot and Phillips (1994) for details.
4. Some empirical illustrations with this procedure are reported in Phillips (1992).

REFERENCES

Phillips, P. C. B. (1992) "The Long-Run Australian Consumption Function Reexamined: An Empirical Exercise in Bayesian Inference," Ch. 11 in C. P. Hargreaves (ed.), *Macroeconomic Modelling of the Long Run.* Aldershot: Edward Elgar, pp. 287–322.

Zivot, E. & P. C. B. Phillips (1994) "A Bayesian Analysis of Trend determination in Economic Time Series," *Econometric Reviews*, **13:** 291–336.

DGRAPHE(zx, zy, eqnname, filegr, yname, xname)

PURPOSE

- This is a graphics procedure for use in Bayesian cointegrating regression analysis. The procedure graphs the posterior densities for the long-run autoregressive parameter in AR(1) and AR(k) models fitted to the residuals from a cointegrating regression. Labels and equation specifics are input as arguments of the procedure. This procedure is generally used in conjunction with the procedure **CBARE**.

FORMAT

- `dgraphe(zx,zy,eqnname,filegr,yname,xname);`

INPUTS

- `zx` vector of points where the densities are to be evaluated
- `zy` matrix of data of density ordinates (`zx` $\times$ `cols(zy)`)
- `eqnname` title for figure giving the cointegrating regression (string)
- `filegr` graphics tkf file name
- `yname` y-axis label
- `xname` x-axis label

OUTPUTS

- Graphs with user supplied labels and legends that record the details of the fitted models

GLOBALS

- `_eps` = epsilon parameter for e-prior (see Phillips, 1992 and Zivot & Phillips, 1994).
- Usually $0.01 \leq$ `eps` ≤ 0.05.

EXAMPLE

/* Graphing the posterior densities for an AR(k) + TR(t) model with $k = 1$ and $k > 1$ */

```
seed = 44125; npoints = 300;@ number of points to evaluate
    densities@
klag = 2;@ set AR lag parameter:  use klag ge 1@
_eps = 0.03;@ set eps = epsilon in e-prior of Zivot &
    Phillips(1994)@
eqnname = ‘‘y = a + b’x + u’’;
filegr = ‘‘myfile.tkf’’;@ set tkf file@
yname = ‘‘density’’;
xname = “\202\114\201 = l_r AR coeff.";
nstd = 7.5; @ defines posterior support:  +/-nstd*asym. st.dev.@
a = 1.0; b = 0.50; c = 2.0; n = 150; x = rndns(n,1,seed);
    u = rndns(n,1,seed);
x = recserar(x,0,a); y = b*x + c*ones(n,1) + u; sdat = y~x;
{a1,a2,a3,a4,a5} = quickols(sdat);@ ols residuals from
    regression@
    @ of first column on other cols of sdat@
sdat = a5; @ reset to residuals from cointegrating regression@
{pdf1,pdf2,pdf3,zb} = cbare(sdat,nstd,npoints,klag);
zy = pdf1~pdf2~pdf3; zx = zb; dgraphe(zx,zy,eqnname,filegr,
    yname,xname);
```

RELATED PROCEDURES

- **CBARE, BMAT3, QUICKOLS, DGRAPH**

REMARKS

1. Need to set `klag` ≥ 1.
2. Need to define global `_eps`.

KERNELS.SRC

Procedures for Computing Kernels

KERNELS(u, l)

PURPOSE

- Driver routine for the CAUCHY( ), FEJER( ), TUKHAM( ), BOHMAN( ), REISZ( ), DIRIC( ), MDIRIC( ), PARZEN( ), GW( ), QS( ) kernels.

FORMAT

- `amat = kernel(u,l);`

INPUTS

- `u` stationary variable $(n \times k)$
- `l` number of autocovariance terms in the kernel

OUTPUTS

- `amat` $(k \times k)$ matrix given by $\left(\frac{1}{n}\right) \sum_{j=1}^{\ell} w_\ell(j) \sum_{t=i+1}^{n} u_t u'_{t-j}$, where $w_\ell(j)$ is a kernel weight

GLOBALS

- `_ker_fun`
- Set `_ker_fun` to one of the available kernels before calling this routine.

AVAILABLE KERNELS

1. FEJER( ) Fejer, Bartlett kernel
2. DIRIC( ) Dirichlet kernel
3. MDIRIC( ) Modified Dirichlet kernel
4. PARZEN( ) Parzen kernel
5. TUKHAM( ) Tukey–Hamming kernel
6. TUKHAN( ) Tukey–Hanning kernel
7. CAUCHY( ) Cauchy kernel
8. BOHMAN( ) Bohman kernel
9. REISZ( ) Riesz, Bochner kernel
10. GW( ) Gauss–Weierstrass kernel
11. QS( ) Andrews (1991) Quadratic–Spectral

EXAMPLE

```
_ker_fun = &parzen;
a = kernel(detrend(y,0),5);
``Autocovariance matrix using Parzen window:'' a;
``Spectrum of y at frequency zero:'' (y'y/rows(y)) + a + a';
```

REFERENCES

Andrews, D. W. K. (1991) "Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimation," *Econometrica*, **59**: 817–858.

Brillinger, David R. (1981) *Time Series Data Analysis and Theory.* San Francisco, CA: Holden–Day.

COINT Global Constants

Please note that once these global variables are set to 1 they remain active for the remainder of your **GAUSS** program.

_filter

1. **COINT** supports the use of an AR(1) filter to compute the spectrum at frequency zero. Consider a stationary series ut such that

$$u_t = Au_{t-1} + z_t$$

   The spectrum of ut is $f_{uu}(0) = \text{inv}(I - A) f_{zz}(0) \text{inv}(I - A')$.

2. The aim of the AR(1) filter is to flatten the spectrum of u around the zero frequency, thereby making it easier to estimate the true spectrum by simple averaging of the periodogram.

3. To use the AR(1) feature of **COINT**, set `_filter = 1`; i.e., in terms of **GAUSS** code:

4. `_filter = 1;`

5. Setting _filter to any other value switches the AR(1) filter off.

6. DEFAULT VALUE: `_filter = 0`.

_aband

1. Andrews (1991) has developed data based (or automatic) bandwidth procedures for computing the spectrum. **COINT** implements these procedures for the Parzen, Fejer, Tukey–Hamming, and the Quadratic–Spectral kernels. When `_aband` is active, **COINT** ignores the value you specify for the band-width parameter and automatically substitutes the data-based value.

2. Automatic bandwidth selection is activated by setting `_aband` to unity before you invoke a procedure; i.e., in terms of **GAUSS** code,

3. `_aband = 1;`

4. You can switch off automatic bandwidth feature by setting _aband to any other value (zero is a good choice).

5. DEFAULT VALUE: `_aband = 0`.

_sbstart, _sbend

1. Hansen's (1991) MeanF and SupF statistics for testing the stability of the cointegrating vector rely on estimates of the F statistic for a range of break-points in your sample. `_sbstart` and `_sbend` define the beginning and end points of the sample. They must be expressed as a proportion of the sample period.

2. If you choose to alter these parameters, please ensure that `_sbstart` < `_sbend`.

3. DEFAULT VALUES: `_sbstart` = 0.15; `_sbend` = 0.85.

4. Thus, if you have 100 observations, the F-statistic for structural change will be computed for observations 15 to 85 inclusive.

_variance (COINT 2.1 and above)

COINT 2.0 uses an OLS (degrees of freedom adjusted estimator) to calculate the AIC, BIC and PIC model selection criteria, including the posterio odds ratio. These calculations involve two components: (1) the estimator for the variance used to calculate the AIC, BIC, and PIC statistics, and (2) the estimated variance used to calculate the posterior odds ratio. For some applications, however, an MLE variance estimator, which is not adjusted for degrees of freedom, may be preferred.
New to COINT 2.1, the **_variance** switch allows you use an MLE estimator to calculate these statistics. COINT 2.1 recognizes three settings for **_variance**:

1. **_variance = 0**, in which case the OLS estimator is used to calculate the AIC, BIC, and PIC criteria. The associated variance-covariance matrix of the parameter estimates, which is used to calculated the posterior odds ratio, is also estimated using the OLS estimator in this case. Using **_variance = 0** replicates COINT 2.0, and remains the default.

2. **_variance = 1**, in which case the MLE estimator is used to calculate the AIC, BIC, and PIC criteria. The variance-covariance matrix of the parameter estimates (and hence the posterior odds ratio) is calculated using the OLS estimator.

3. **_variance = 2**. In this case, the MLE estimator is used to calculate all model selection criteria and the variance-covariance matrix of the parameter estimates.

The routines directly affected by **_variance** parameter are: **pparord(), ppadfbic(), ppadfbit(), armamay(), armamayy(), pparord()**.

Useful Supporting Procedures

1. **JCRIT**(p, q): returns a (6×1)vector of critical values for the Park and Choi (1988) $\mathbf{J}(p, q)$ statistic {1%, 5%, 10%, 90%, 96%, 99%}. NB:$-1 \leq p \leq 5$; $q > p$; $0 \leq q \leq 11$.

2. **ZACRIT**(nobs,p): returns a (6×1) vector of critical values for Phillips' (1987) Z_α statistic {1%, 5%, 10%, 90%, 95%, 99%}. Set "nobs" to the number of observations used in the unit root regression. Set p to the order of the time polynomial in the fitted regression. NB: $-1 \leq p \leq 5$; critical values change for nobs: $1 \leq$ nobs ≤ 500; change points: nobs = 51, 100, 150, 200, 250, 300, 350, 400, 450, 500.

3. **ZTCRIT**(nobs,p): returns a (6×1)vector of critical values for Phillips' (1987) Z_t statistic {1%, 5%, 10%, 90%, 95%, 99%}. Set "nobs" to the number of observations used in the unit root regression. Set p to the order of the time polynomial in the fitted regression. NB: $-1 \leq p \leq 5$; critical values change for nobs: $1 \leq$ nobs ≤ 500; change points:nobs = 50, 100, 150, 200, 250, 300, 350, 400, 450, 500.

4. **RZACRIT**(nobs,n, p): returns a (6×1)vector of critical values for Phillips' (1987) Z_α statistic when applied to the residuals of a cointegrating regression. Set "nobs" to the number of observations used in the cointegrating regression. Set p to the order of the time polynomial in the cointegrating regression. Set n to the number of explanatory (integrated) variables in the cointegrating regression ($1 \leq n \leq 5$). NB: $-1 \leq p \leq 5$; critical values change for nobs: $1 \leq$ nobs ≤ 500; change points: nobs = 100, 200, 300, 400, 500.

5. **RZTCRIT**(nobs,n, p): returns a (6×1)vector of critical values for Phillips' (1987) Z_t statistic when applied to the residuals of a cointegrating regression. Set "nobs" to the number of observations used in the cointegrating regression. Set p to the order of the time polynomial in the cointegrating regression. Set n to the number of explanatory (integrated) variables in the cointegrating regression $1 \leq n \leq 5$). NB: $-1 \leq P \leq 5$; critical values change for nobs: $1 \leq$ nobs ≤ 500; change points: nobs = 100, 200, 300, 400, 500.

6. **C_SW**(n, p): returns a (6×1) vector of critical values for the Stock and Watson (1988) test for cointegration. Set n to the dimension of the cointegrating system; p to the order of the time polynomial in the data.

7. **C_PU**(n, p): returns a (6×1) vector of critical values for the P_u statistic. Set n to the dimension of the cointegrating system. Set p to the order of the deterministic part.

8. **C_PZ**(n, p): returns a (6×1) vector of critical values for the P_z statistic. Set n to the dimension of the cointegrating system. Set p to the order of the deterministic part.

9. **DETREND**(y, p): regresses y against a polynomial time trend of order p and returns the residuals. If $p = -1$, procedure returns y. Use $p = 0$ for demeaning; $p = 1$ for regression against a constant term and trend; $p > 1$ for higher order polynomial time trend.

10. **DIFF**(y, p): if $p > 0$, returns $y_t - y_{t-n}$ (dimension: $(\text{rows}(y) - p) \times 1$). **NOTE: DIFF**$(x, 0) = x$.

NOTE: for procedures returning critical values, if p is not in $[-1, 5]$, the procedure returns a (6×1) zero vector (i.e., zeros(6,0)).

COINT 1.0 versus COINT 2.0: New Features

1. The Quadratic-Spectral kernel estimator has been added to the list of kernels in **COINT**. Also, **COINT** now supports automatic bandwidth selection for estimating the long-run variance (i.e., spectrum at frequency zero). Automatic bandwidth is controlled by the global constant _aband. Set `_aband = 1` to enable automatic bandwidth selection. See page 113 for more information.

2. You can now estimate the long-run variance using an AR(1) filter. This is controlled by the global constant _filter. See page 113 for more information.

3. Introduction of the global parameter `_NoDet`. This parameter controls whether the regression procedures include a deterministic part in the cointegrating regression (typically denoted by "d"' in the argument list of the regression procedures). The `_NoDet` parameter eliminates the need for the "s" parameter as an argument to regression procedure. **IMPLICATION:** you will need to adjust your code to use the regression procedures in **COINT 2.0**. Simply remove the "s" parameter from your calls to the regression procedures in **COINT**. If you set $s = 1$ to suppress the deterministic part, set `_NoDet = 1` just before the procedure call. Remember to set `_NoDet = 0` after the procedure call to ensure that subsequent procedures are not affected. See **CREGRS.SRC** for a description of the regression procedures in **COINT**.

4. **FM** now computes B. E. Hansen's structural break tests. This feature resulted in a extra return argument to the calls for **FM** and **CCR**. **IMPLICATION:** if you used **FM** or **CCR** in your code, you will need to add one extra return argument. The extra argument is a 3×1 column vector containing Hansen's structural break tests.

5. The **SJ**( ) procedure has one less argument. The "r" parameter for the number of cointegrating vectors has been removed. It now computes the Johansen statistic for all values of r (i.e., $r = 0, 1$, ..., cols(x)). The procedure also allows you to put the deterministic part in the cointegrating regression, rather than the fitted VAR. Set `_NoDet = 1` to do so. Lastly, the procedure does not return any critical values. The procedure now returns only 4 arguments. **IMPLICATION:** if you used **SJ**( ) in your code, you will need to change the calls to run **SJ**( ).

New libraries of procedures have been added for: ARMA model selection and estimation; long-run variance and spectral estimation by data driven ARMA pre-filtering and kernel estimation; Bayes posterior analysis of deterministic trends and unit roots.

COINT 2.1 versus COINT 2.0: New Features

1. COINT 2.1, which was released November 2016, introduces a new global setting called "_variance" that can be used to control the type of variance estimator used when calculating the AIC, BIC, and PIC model selection criteria. The user can also control the variance estimator used to calculate the posterior odds ratio. See page 63 for more details.

Index

A

ADF(y, p, l), 12
ADFTR(x, p, r), 69
AMLRVR(x, _kernel), 93
ARBC(x, pmax, tmax), 65
ARMABC(x, p, q, pt), 80
ARMABIC2(x, pmax, qmax, tmax), 78
ARMABIC3(x, pmax, tmax), 75
ARMATR(x, p, q, pt, pmax), 70
ARMATR2(x, p, q, pt, pmax), 74
ARMATRA(x, p, q, pt, pmax), 72
ARORD(x, pmax, tmax), 64
ARPC(x, pmax, tmax), 67

B

BARTR(x, nstd, npoints, klag, pt), 99
BARTR2(x, nstd, npoints, klag, pt), 101

C

CADF(y, x, p, l), 16
CBARE(x, nstd, npoints, klag, pt), 106
CCR(y, x, d, l), 31
COINT 1.0 versus COINT 2.0: New Features, 117
COINT 2.1 versus COINT 2.0: New Features, 118
COINT Source File: ARMA.SRC, 4
COINT Source File: BAYES.SRC, 6
COINT Source File: LRVAR.SRC, 5
COINT Source Files: UNIT.SRC, CREGRS.SRC, BASE.SRC, KERNELS.SRC, 2
CREGR(y, x, d, l), 59
CZA(y, x, p, l), 14

D

DATGRAPH(x, scr, name), 105
DGRAPH(zx, zy, name, dates, klag, nobs, ngraph), 103
DGRAPHE(zx, zy, eqnname, filegr, yname, xname), 108

F

FM(y, x, d, l), 33
FM_GIVE(y, x, z, l, t), 39
FM_GMM(y, x, z, l), 40
FM_OLS(y, z, d, l), 35
FM_VAR(y, d, k, l), 37

G

GCBZ(y, x, z, d, M), 50
GPCB(y, x, z,d, M), 46
GPCBBW(y, x, z, d, f, M), 48
GRFBICPQ(pmax, qmax, bic), 83
GSTAT(y, p, q, v), 18

H

HSTAT(y, x, p, q, l), 58
HSTATC(y, x, p, q, l), 56
HSTATF(y, x, p, q, l), 57

J

JSTAT(y, p, q), 20

K

KERNELS(u, l), 111

L

LRVRO(x, pmax, qmax, _kernel), 94

P

PCB(y, x, d, M), 41
PCBBW(y, x, d, f, M), 43
PCBZ(y, x, d, M), 45
PPLEE(x, pmax, qmax, _kernel), 92
PPZAZT(x, pmax, qmax, pt, _kernel, sel), 96
PS(y, x, p, ld, lg), 54
PU(y, x, p, l), 25
PZ(y, x, p, l), 27

R

References, 7

S

SARMAGRF(a, b, sig2, x), 86

SJ(x, p, k), 52
SPECARMA(a, b, sig2, x), 85
SPECWX(x, pmax, qmax, _kernel, wx), 87
SPWXGRF(x, pmax, qmax, _kernel, wx), 90
SW(y, p, l), 23

U

Useful Supporting Procedures, 115

Z

ZA(y, p, l), 21