Package 'FCUSUM'

November 2, 2026

Type Package

Title Fourier CUSUM Cointegration Test

Version 1.1.0

Description Implements a Fourier CUSUM test for the null hypothesis of cointegration in the presence of smooth structural change. The cointegrating regression is estimated by dynamic OLS with a Fourier pair at a fractional or integer frequency, and the residual CUSUM statistic of Xiao and Phillips (2002) <doi:10.1016/S0304-4076(01)00103-8> is scaled by a Bartlett long-run variance estimate. A sup-F pre-test decides whether the Fourier terms are needed, and bootstrap critical values that replicate the frequency search give correct size when a break is present.

License GPL-3

Encoding UTF-8

Depends R (>= 3.5.0)

Imports stats, graphics, utils

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Taha Zaghdoudi [aut, cre]

Maintainer Taha Zaghdoudi <zedtaha@gmail.com>

Repository CRAN

Date/Publication 2026-11-01 23:40:19 UTC

Contents

fcum 2
fcum_cv_sim 5
plot.fcum 6
print.fcum 7
summary.fcum 7

Index 9

`fcum` *Fourier CUSUM Test for the Null of Cointegration*

Description

Tests the null hypothesis of cointegration against the alternative of no cointegration, allowing for smooth structural change in the intercept approximated by a Fourier pair at a fractional (default) or integer frequency. The procedure extends the residual-based CUSUM test of Xiao and Phillips (2002) with the flexible Fourier form, in three steps:

1. estimate the cointegrating regression by dynamic OLS and search the Fourier frequency that minimises the residual sum of squares;
2. pre-test the Fourier terms with a sup-F statistic;
3. if the Fourier terms are significant, compute the Fourier CUSUM statistic with bootstrap critical values that replicate the frequency search; otherwise compute the Xiao-Phillips CUSUM statistic without Fourier terms.

Usage

```
fcum(
  y,
  x,
  kstar = 3,
  freq = c("fractional", "integer"),
  fourier = c("auto", "always", "never"),
  pretest_level = 0.05,
  cv = c("bootstrap", "table"),
  boot = c("iid", "sieve"),
  nboot = 499,
  q = NULL,
  qmax = NULL,
  bw = c("nw", "kurozumi", "andrews"),
  step = 0.01,
  seed = NULL
)
```

Arguments

`y`	Numeric vector. Dependent I(1) variable.
`x`	Numeric vector, matrix or data frame of I(1) regressors with the same number of observations as `y`.
`kstar`	Positive integer. Maximum Fourier frequency searched (default 3).
`freq`	`"fractional"` (default; grid `0.1`, `0.1 + step`, ..., `kstar`) or `"integer"` (grid `1`, ..., `kstar`).

`fourier`	`"auto"` (default) decides with the sup-F pre-test; `"always"` reports the Fourier CUSUM regardless; `"never"` reports the Xiao-Phillips CUSUM without Fourier terms.
`pretest_level`	Significance level of the sup-F pre-test (default 0.05).
`cv`	Critical values for the Fourier CUSUM: `"bootstrap"` (default) or `"table"` (asymptotic values at the selected frequency; fast but conservative in finite samples because they ignore the frequency search).
`boot`	Bootstrap scheme: `"iid"` (default; best power) or `"sieve"` (AR sieve on the residuals; better size with highly persistent errors, lower power).
`nboot`	Integer. Bootstrap replications (default 499). Also used for simulated critical values when a configuration is outside the built-in tables.
`q`	Integer or `NULL`. DOLS leads and lags of Δx_t; chosen by BIC between 0 and `qmax` when `NULL` (default).
`qmax`	Integer. Largest number of leads/lags considered. Default `min(4, floor(4 * (n / 100)^(1/4)))`.
`bw`	Bandwidth rule for the Bartlett long-run variance: `"nw"` (Newey-West, default), `"kurozumi"` (Andrews AR(1) plug-in bounded as in Kurozumi 2002) or `"andrews"`.
`step`	Grid step for fractional frequencies (default 0.01).
`seed`	Optional integer seed for the bootstrap.

Details

The Fourier cointegrating regression is

$$y_t = \alpha_0 + \alpha_1 \sin(2\pi k t/T) + \alpha_2 \cos(2\pi k t/T) + \beta' x_t + \sum_{j=-q}^{q} \gamma_j' \Delta x_{t-j} + u_t,$$

estimated by dynamic OLS so that the limit distribution of the statistic is free of nuisance parameters arising from endogeneity and serial correlation (Xiao and Phillips 2002). The CUSUM statistic is

$$CS = \max_t \frac{1}{\hat{\omega}\sqrt{T}} \left| \sum_{s=1}^{t} \hat{u}_s \right|,$$

with $\hat{\omega}^2$ a Bartlett long-run variance estimate. Large values reject cointegration.

Why a pre-test and a bootstrap. With a fractional frequency the null distribution of the Fourier CUSUM depends on the true frequency, and the search over k shrinks the statistic. When there is no break the selected frequency is spurious; when there is one, it is estimated with error. The sup-F pre-test (Becker, Enders and Lee 2006; Tsong et al. 2016) sends samples without a break to the standard Xiao-Phillips test, and the parametric bootstrap reproduces the frequency search around the estimated break so that the Fourier CUSUM has correct size. In Monte Carlo experiments (see `data-raw/size_power.R`) this keeps size close to the nominal level both with and without a break, whereas critical values conditional on the selected frequency alone are very conservative and have little power.

Value

An object of class `fcum`, a list with

test `"Fourier CUSUM"` or `"Xiao-Phillips CUSUM"`: the statistic on which the decision is based.

statistic, p.value, critical_values, decision, significance For the reported test.

pretest List with the sup-F statistic, its critical values, p-value and whether the Fourier terms were retained.

stat_fourier, stat_plain Both CUSUM statistics.

best_frequency Selected Fourier frequency.

coefficients, residuals, lrv, bw_lags, nobs Of the model behind the reported test.

best_model That model as an `lm` object (its default standard errors are not valid for DOLS inference).

settings kstar, freq, q, bw, cv, boot, nboot, p.

call The matched call.

Changes from version 1.0.0

The 1.0.0 implementation (i) used critical values that were lower-tail quantiles, so a true null was rejected in about 90% of samples at the nominal 5% level; (ii) used OLS residuals scaled by their short-run standard deviation, so the test was not robust to serial correlation or endogeneity; (iii) used critical values indexed by `kstar` that ignore the dependence on the frequency; and (iv) flattened a multivariate `x` into one column. All are corrected here.

References

Xiao, Z., & Phillips, P. C. B. (2002). A CUSUM test for cointegration using regression residuals. *Journal of Econometrics*, 108(1), 43-61.

Tsong, C.-C., Lee, C.-F., Tsai, L.-J., & Hu, T.-C. (2016). The Fourier approximation and testing for the null of cointegration. *Empirical Economics*, 51(3), 1085-1113.

Becker, R., Enders, W., & Lee, J. (2006). A stationarity test in the presence of an unknown number of smooth breaks. *Journal of Time Series Analysis*, 27(3), 381-409.

Stock, J. H., & Watson, M. W. (1993). A simple estimator of cointegrating vectors in higher order integrated systems. *Econometrica*, 61(4), 783-820.

Kurozumi, E. (2002). Testing for stationarity with a break. *Journal of Econometrics*, 108(1), 63-99.

Examples

```
set.seed(123)
n <- 150
x <- cumsum(rnorm(n))
brk <- 2 * sin(2 * pi * 1.5 * seq_len(n) / n)
y <- 2 + brk + 1.5 * x + rnorm(n)

res <- fcum(y, x, kstar = 3, nboot = 99, seed = 1)
res
summary(res)
```

```
# Two regressors, integer frequencies, fast table critical values
x2 <- cbind(x1 = x, x2 = cumsum(rnorm(n)))
y2 <- 1 + brk + x2 %*% c(1, -0.5) + rnorm(n)
fcum(y2, x2, kstar = 2, freq = "integer", cv = "table")
```

`fcum_cv_sim` *Simulate Null Distributions for the Fourier CUSUM Test*

Description

Simulates, under the null of cointegration without structural change, the distribution of one of the statistics used by `fcum`, with exactly the same estimation steps (DOLS, Fourier terms, Bartlett long-run variance). It reproduces the built-in tables and gives finite-sample critical values for any configuration.

Usage

```
fcum_cv_sim(
  n,
  p = 1,
  type = c("plain", "supF", "fixed"),
  k = 1,
  kstar = 3,
  freq = c("fractional", "integer"),
  step = 0.01,
  q = 1,
  bw = c("nw", "kurozumi", "andrews"),
  nsim = 2000,
  probs = c(0.5, 0.75, 0.8, 0.85, 0.9, 0.925, 0.95, 0.975, 0.99),
  seed = NULL
)
```

Arguments

`n`	Integer. Sample size (before DOLS trimming).
`p`	Integer. Number of I(1) regressors.
`type`	Statistic to simulate:
	`"plain"` Xiao-Phillips CUSUM without Fourier terms.
	`"supF"` sup-F pre-test statistic for the Fourier terms, maximised over the frequency grid defined by `kstar` and `freq`.
	`"fixed"` Fourier CUSUM at the fixed frequency `k` (no search).
`k`	Numeric. Fourier frequency for `type = "fixed"`.
`kstar`, `freq`, `step`	Frequency grid for `type = "supF"`; see `fcum`.

q	Integer. Number of DOLS leads and lags.
bw	Bandwidth rule for the long-run variance (see fcum).
nsim	Integer. Number of Monte Carlo replications.
probs	Probabilities for the returned quantiles.
seed	Optional integer seed.

Details

Under the null, $y_t = u_t$ with u_t iid N(0, 1) and x_t a p-dimensional Gaussian random walk independent of u_t. The statistics are invariant to the cointegrating vector, the intercept and the scale of u_t.

Value

A list with the simulated quantiles and all simulated statistics (draws).

Examples

```
fcum_cv_sim(n = 100, p = 1, type = "plain", nsim = 200, seed = 1)$quantiles
fcum_cv_sim(n = 100, p = 1, type = "supF", kstar = 2, step = 0.05,
            nsim = 100, seed = 1)$quantiles
```

---

plot.fcum *Plot Method for fcum Objects*

Description

Plots the scaled cumulative sum of residuals of the reported model with the critical band.

Usage

```
## S3 method for class 'fcum'
plot(x, level = c("5%", "10%", "1%"), ...)
```

Arguments

x	An object of class fcum.
level	Significance level for the band: "5%" (default), "10%" or "1%".
...	Further graphical parameters passed to plot.

Value

Invisibly returns the input object.

print.fcum *Print Method for fcum Objects*

Description

Print Method for fcum Objects

Usage

```
## S3 method for class 'fcum'
print(x, digits = 4, ...)
```

Arguments

x	An object of class fcum.
digits	Number of digits to print.
...	Additional arguments (not used).

Value

Invisibly returns the input object.

summary.fcum *Summary Method for fcum Objects*

Description

Summary Method for fcum Objects

Usage

```
## S3 method for class 'fcum'
summary(object, ...)

## S3 method for class 'fcum'
coef(object, ...)

## S3 method for class 'fcum'
residuals(object, ...)
```

Arguments

object	An object of class fcum.
...	Additional arguments (not used).

Value

Invisibly returns the input object.

Index

coef.fcum *(*summary.fcum*)*, 7

fcum, 2, *5*, *6*
fcum_cv_sim, 5

plot.fcum, 6
print.fcum, 7

residuals.fcum *(*summary.fcum*)*, 7

summary.fcum, 7