Package ‘hatemicoint’

May 8, 2026

Type Package

Title Hatemi-J Cointegration Test with Two Unknown Regime Shifts

Version 1.0.1

Description Implements the Hatemi-J (2008) cointegration test which allows for two unknown structural breaks (regime shifts) in the cointegrating relationship. The test provides three test statistics: ADF* (Augmented Dickey-Fuller), Zt* (Phillips-Perron Z_t), and Za* (Phillips-Perron Z_alpha), along with endogenously determined break dates. Critical values are based on simulations from Hatemi-J (2008) <doi:10.1007/s00181-007-0175-9>.

License GPL-3

URL https://github.com/muhammedalkhalaf/hatemicoint

BugReports https://github.com/muhammedalkhalaf/hatemicoint/issues

Encoding UTF-8

Depends R (>= 3.5.0)

Imports stats

Suggests testthat (>= 3.0.0), knitr, rmarkdown

RoxygenNote 7.3.3

Config/testthat/edition 3

NeedsCompilation no

Author Muhammad Alkhalaf [aut, cre, cph] (ORCID: <https://orcid.org/0009-0002-2677-9246>)

Maintainer Muhammad Alkhalaf <muhammedalkhalaf@gmail.com>

Repository CRAN

Date/Publication 2026-03-13 13:30:08 UTC

Contents

hatemicoint-package *hatemicoint: Hatemi-J Cointegration Test with Two Unknown Regime Shifts*

Description

Implements the Hatemi-J (2008) cointegration test which allows for two unknown structural breaks (regime shifts) in the cointegrating relationship. The test provides three test statistics: ADF* (Augmented Dickey-Fuller), Zt* (Phillips-Perron Z_t), and Za* (Phillips-Perron Z_alpha), along with endogenously determined break dates.

Details

The main function in this package is `hatemicoint`, which performs the cointegration test with two structural breaks.

The test is particularly useful when:

- Standard cointegration tests fail to reject the null of no cointegration
- There is reason to believe the relationship has changed over time
- Two structural breaks are suspected in the data

Test Statistics

ADF* Augmented Dickey-Fuller test on residuals with optimal lag selection

Zt* Phillips-Perron Z_t test with kernel-based long-run variance

Za* Phillips-Perron Z_alpha test with kernel-based long-run variance

Critical Values

Critical values depend on the number of regressors (k = 1, 2, 3, or 4) and are taken from Table 1 of Hatemi-J (2008). The null hypothesis of no cointegration is rejected when the test statistic is smaller (more negative) than the critical value.

Author(s)

Maintainer:

References

Hatemi-J, A. (2008). Tests for cointegration with two unknown regime shifts with an application to financial market integration. *Empirical Economics*, 35, 497-505. doi:10.1007/s0018100701759

Gregory, A.W., & Hansen, B.E. (1996). Residual-based tests for cointegration in models with regime shifts. *Journal of Econometrics*, 70(1), 99-126. doi:10.1016/03044076(69)416857

hatemicoint *Hatemi-J Cointegration Test with Two Unknown Regime Shifts*

Description

Performs the Hatemi-J (2008) cointegration test which allows for two unknown structural breaks (regime shifts) in the cointegrating relationship. The test searches over all possible break date combinations and returns the minimum test statistics along with the endogenously determined break dates.

Usage

```
hatemicoint(
  y,
  x,
  maxlags = 8,
  lag_selection = c("tstat", "aic", "sic"),
  kernel = c("iid", "bartlett", "qs"),
  bwl = NULL,
  trimming = 0.15
)
```

Arguments

`y`	Numeric vector. The dependent variable (must be I(1)).
`x`	Numeric matrix or vector. The independent variable(s) (must be I(1)). Maximum of 4 regressors allowed (k <= 4).
`maxlags`	Integer. Maximum number of lags for ADF test. Default is 8.
`lag_selection`	Character. Lag selection criterion: `"tstat"` (default), `"aic"`, or `"sic"`.
`kernel`	Character. Kernel for long-run variance estimation in PP tests: `"iid"` (default), `"bartlett"`, or `"qs"` (quadratic spectral).
`bwl`	Integer. Bandwidth for kernel estimation. If NULL (default), computed as `round(4 * (n/100)^(2/9))`.
`trimming`	Numeric. Trimming parameter for break point search. Must be between 0 and 0.5 (exclusive). Default is 0.15.

Details

The Hatemi-J (2008) test extends the Gregory and Hansen (1996) cointegration test by allowing for two structural breaks instead of one. The test is based on the residuals from the cointegrating regression with regime shift dummies:

$$y_t = \alpha_0 + \alpha_1 D_{1t} + \alpha_2 D_{2t} + \beta_0' x_t + \beta_1' D_{1t} x_t + \beta_2' D_{2t} x_t + u_t$$

where D_{1t} and D_{2t} are dummy variables for the two regime shifts.

Three test statistics are computed:

- **ADF***: Modified Augmented Dickey-Fuller test on residuals
- **Zt***: Phillips-Perron Z_t test on residuals
- **Za***: Phillips-Perron Z_alpha test on residuals

The null hypothesis is no cointegration. Rejection occurs when the test statistic is smaller (more negative) than the critical value.

Value

An object of class `"hatemicoint"` containing:

adf_min Minimum ADF* test statistic

tb1_adf First break location (observation number) for ADF*

tb2_adf Second break location (observation number) for ADF*

zt_min Minimum Zt* test statistic

tb1_zt First break location for Zt*

tb2_zt Second break location for Zt*

za_min Minimum Za* test statistic

tb1_za First break location for Za*

tb2_za Second break location for Za*

cv_adfzt Critical values for ADF* and Zt* tests (1%, 5%, 10%)

cv_za Critical values for Za* test (1%, 5%, 10%)

nobs Number of observations

k Number of regressors

maxlags Maximum lags used

lag_selection Lag selection method used

kernel Kernel used

bwl Bandwidth used

trimming Trimming parameter

References

Hatemi-J, A. (2008). Tests for cointegration with two unknown regime shifts with an application to financial market integration. *Empirical Economics*, 35, 497-505. doi:10.1007/s0018100701759

Gregory, A.W., & Hansen, B.E. (1996). Residual-based tests for cointegration in models with regime shifts. *Journal of Econometrics*, 70(1), 99-126. doi:10.1016/03044076(69)416857

Examples

```
# Generate example data with structural breaks
set.seed(123)
n <- 200
x <- cumsum(rnorm(n))

# Create cointegrated series with two breaks
y <- numeric(n)
y[1:70] <- 1 + 0.8 * x[1:70] + rnorm(70, sd = 0.5)
y[71:140] <- 3 + 1.2 * x[71:140] + rnorm(70, sd = 0.5)
y[141:200] <- 2 + 0.6 * x[141:200] + rnorm(60, sd = 0.5)

# Run the test
result <- hatemicoint(y, x)
print(result)
summary(result)
```

`print.hatemicoint` *Print Method for hatemicoint Objects*

Description

Print Method for hatemicoint Objects

Usage

```
## S3 method for class 'hatemicoint'
print(x, ...)
```

Arguments

`x`	An object of class "`hatemicoint`".
`...`	Additional arguments (ignored).

Value

Invisibly returns the input object.

`summary.hatemicoint` *Summary Method for hatemicoint Objects*

Description

Summary Method for hatemicoint Objects

Usage

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

Arguments

`object`	An object of class "`hatemicoint`".
`...`	Additional arguments (ignored).

Value

Invisibly returns a summary list with inference at various significance levels.

Index