Package 'fqardl'

November 10, 2026

Type Package

Title Fourier ARDL Methods: Quantile, Nonlinear, Multi-Threshold & Unit Root Tests

Version 1.1.1

Date 2026-11-07

Maintainer Muhammad Alkhalaf <muhammedalkhalaf@gmail.com>

Description Implementation of ARDL methods for cointegration analysis with structural breaks and asymmetric effects. Includes: (1) Fourier Quantile ARDL (FQARDL), quantile regression with Fourier approximation following the quantile ARDL of Cho, Kim and Shin (2015) <doi:10.1016/j.jeconom.2015.05.003>; (2) Fourier Nonlinear ARDL (FNARDL), asymmetric cointegration with partial sum decomposition following Shin, Yu and Greenwood-Nimmo (2014) <doi:10.1007/978-1-4899-8008-3_9>; (3) a deprecated Multi-Threshold NARDL (MTNARDL) function; (4) Fourier unit root tests, ADF and KPSS tests with Fourier terms following Enders and Lee (2012) <doi:10.1016/j.econlet.2012.04.081> and Becker, Enders and Lee (2006) <doi:10.1111/j.1467-9892.2006.00478.x>. Features automatic lag and frequency selection, Wald bounds statistics with the tables of Pesaran, Shin and Smith (2001) <doi:10.1002/jae.616> (no verdict with Fourier terms, for which the tables are not valid), recursive bootstrap bounds tests following Bertelli, Vacca and Zoia (2022) <doi:10.1016/j.econmod.2022.105987> and McNown, Sam and Goh (2018) <doi:10.1080/00036846.2017.1366643>, Wald tests for asymmetry, multiplier computation, and visualizations.

License GPL-3

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

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

Encoding UTF-8

LazyData true

Depends R (>= 3.5.0)

Imports quantreg (>= 5.0), ggplot2 (>= 3.0.0), tidyr, gridExtra, stats

Suggests testthat (>= 3.0.0), knitr, rmarkdown, plotly, covr

RoxygenNote 7.3.3

Config/testthat/edition 3

NeedsCompilation no

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

Repository CRAN

Date/Publication 2026-11-10 07:50:02 UTC

Contents

fqardl-package . . . 3
bootstrap_bounds_test . . . 3
compute_asymmetric_multipliers . . . 6
compute_diagnostics . . . 6
compute_multipliers . . . 7
decompose_multi_threshold . . . 7
decompose_variables . . . 8
dynamic_multipliers . . . 8
estimate_qardl . . . 9
fadf_f_test . . . 10
fnardl . . . 10
fourier_adf . . . 12
fourier_adf_test . . . 12
fourier_kpss_test . . . 13
fourier_unit_root_analysis . . . 14
fqardl . . . 15
generate_fnardl_report . . . 17
generate_fourier_terms . . . 17
macro_data . . . 18
mtnardl . . . 19
oil_gdp_data . . . 20
perform_bounds_test . . . 21
plot.fnardl . . . 22
plot.fqardl . . . 23
plot.fqardl_mtnardl . . . 24
plot_cumulative_multipliers . . . 24
plot_dynamic_multipliers . . . 25
plot_persistence . . . 25
quantile_wald_test . . . 26
select_fourier_frequency . . . 26
select_optimal_lags . . . 27
test_asymmetry . . . 27

Index **29**

`fqardl-package` *Fourier ARDL Methods for R*

Description

Comprehensive implementation of advanced ARDL methodologies for cointegration analysis with structural breaks and asymmetric effects.

Main functions:

- `fqardl`: Fourier Quantile ARDL estimation
- `fnardl`: Fourier Nonlinear ARDL with asymmetry
- `mtnardl`: Multi-Threshold NARDL (deprecated, to be removed in 2.0.0; use ardlverse::mtnardl)
- `fourier_adf_test`: Fourier ADF unit root test
- `fourier_kpss_test`: Fourier KPSS stationarity test
- `perform_bounds_test`: Wald bounds statistics (no analytical verdict with Fourier terms)
- `bootstrap_bounds_test`: recursive bootstrap bounds test

Author(s)

Muhammad Alkhalaf <contact@rufyqelngeh.com>

References

Shin, Y., Yu, B., and Greenwood-Nimmo, M. (2014). Modelling Asymmetric Cointegration and Dynamic Multipliers in a Nonlinear ARDL Framework.

Enders, W., and Lee, J. (2012). The flexible Fourier form and Dickey-Fuller type unit root tests.

Pesaran, M. H., Shin, Y., and Smith, R. J. (2001). Bounds testing approaches to the analysis of level relationships.

`bootstrap_bounds_test` *Bootstrap Bounds Test*

Description

Recursive bootstrap bounds test for the Fourier ARDL model estimated by `fqardl`. Rewritten in 1.1.0.

Usage

```
bootstrap_bounds_test(
  y,
  X,
  fourier,
  p,
  q,
  tau = NULL,
  case = 3,
  n_boot = 1000,
  verbose = FALSE,
  reselect = NULL,
  scheme = c("bvz", "mcnown"),
  level = 0.05,
  seed = NULL
)
```

Arguments

y	Dependent variable (no missing values).
X	Matrix of regressors in levels.
fourier	Matrix of Fourier terms (from generate_fourier_terms) used for the observed data and as fixed deterministic terms of the bootstrap data generating process.
p	Lag order for y, as in build_ardl_design.
q	Lag order for X, as in build_ardl_design.
tau	Not used since 1.1.0 (the statistics are OLS statistics); kept for compatibility.
case	Model case; only 3 is supported.
n_boot	Number of bootstrap replications.
verbose	Print a progress message.
reselect	NULL (hold the frequency and lags fixed) or a list with elements max_k, max_p, max_q, criterion and optionally k (a frequency fixed by the user in advance) to re-run the selection on every bootstrap sample.
scheme	"bvz" or "mcnown"; see Details.
level	Significance level of the decision.
seed	Optional seed for the bootstrap; the caller's random number state is restored afterwards.

Details

The pseudo-samples are generated by the shared recursive bootstrap engine (file ardl_boot_engine.R, engine version 1.1.0, copied verbatim from the package 'ardlverse'). Under each null hypothesis the restricted conditional error correction model (ECM) for Δy is estimated by OLS together with a marginal model for Δx; y^* and x^* are then generated recursively from resampled, recentred residuals, with the Fourier terms held fixed as deterministic regressors.

`scheme = "bvz"` **(default)** Bertelli, Vacca and Zoia (2022): a separate restricted model for each of the three statistics, a marginal VECM for Δx, residuals recentred after each draw, initial values from a random block of the data (the levels of the regressors and partial sums continue from the block).

`scheme = "mcnown"` The Fov null for all statistics (McNown, Sam and Goh 2018, Steps 1-8) applied to the conditional ECM (package choice; MSG write the y equation in unconditional form): the null of the overall F test generates the samples for all statistics, the Δx equation is unrestricted (it includes y_{t-1}), residuals mean-centred once, the first observations as initial values.

The three statistics are computed from fqardl's own design (`build_ardl_design`) estimated by OLS: the overall F test on the lagged levels of y and x (Fov), the t ratio on the lagged level of y (t) and the F test on the lagged levels of x (Find). The same function is applied to the data and to every bootstrap sample, so the bootstrap distribution refers to the reported statistics. When `reselect` is given, the Fourier frequency (unless fixed) and the lag orders are chosen again on every bootstrap sample with `select_fourier_frequency` and `select_optimal_lags`. Re-selection is a package choice: Bertelli, Vacca and Zoia (2022, step 5(c)) re-estimate the unrestricted model on each bootstrap sample but do not discuss re-selection. In the package's Monte Carlo, holding a data-selected frequency fixed made the test oversized.

The decision combines the three tests with the AND rule: cointegration only if Fov, t and Find all reject at level `level`; Fov rejecting without t, or Fov and t without Find, are reported as the degenerate cases of the first and second type. Critical values are order statistics of the bootstrap distributions.

The statistics are OLS (conditional mean) statistics. Cho, Kim and Shin (2015) give no bounds test and no bootstrap for the null of no cointegration in the quantile ARDL model, and no quantile-specific bootstrap statistic is computed. Versions up to 1.0.6 regressed a new Gaussian random walk on fixed regressors at the first quantile and declared cointegration when F or t rejected; that procedure was oversized and has been removed.

Value

A list with the observed statistics (`orig_F`, `orig_t`, `orig_Find`), the bootstrap distributions (`boot_F`, `boot_t`, `boot_Find`; failed replications are `NA`), the p-values (`p_value_F`, `p_value_t`, `p_value_Find`), the critical values `cv` at 10, 5, 2.5 and 1 percent, the decision code `decision` and its text `label`, and the engine output `engine`.

References

Bertelli, S., Vacca, G. and Zoia, M. (2022). Bootstrap cointegration tests in ARDL models. *Economic Modelling*, 116, 105987. doi:10.1016/j.econmod.2022.105987

McNown, R., Sam, C.Y. and Goh, S.K. (2018). Bootstrapping the autoregressive distributed lag test for cointegration. *Applied Economics*, 50(13), 1509-1521. doi:10.1080/00036846.2017.1366643

Cho, J.S., Kim, T. and Shin, Y. (2015). Quantile cointegration in the autoregressive distributed-lag modeling framework. *Journal of Econometrics*, 188(1), 281-300. doi:10.1016/j.jeconom.2015.05.003

compute_asymmetric_multipliers *Compute Asymmetric Multipliers*

Description

Long-run multipliers L(+) = -theta(+)/rho and L(-) = -theta(-)/rho of the NARDL error correction model (Shin, Yu and Greenwood-Nimmo 2014, eq. 9.7), and the coefficients of the current differences of the partial sums.

Usage

```
compute_asymmetric_multipliers(nardl_result, decompose, x_names)
```

Arguments

nardl_result	NARDL estimation results
decompose	Decomposed variable names
x_names	Original variable names

Value

List with long_run_positive, long_run_negative (named by the decomposed variables), short_run_positive and short_run_negative (the coefficients of the current differences of the partial sums, the impact effects), and ect (rho).

compute_diagnostics *Compute Model Diagnostics*

Description

Computes various diagnostic statistics for the QARDL models.

Usage

```
compute_diagnostics(qardl_results)
```

Arguments

qardl_results	List of QARDL estimation results

Value

List of diagnostics for each quantile

compute_multipliers *Compute Long-run and Short-run Multipliers*

Description

Calculates the long-run and short-run multipliers from QARDL estimates.

Usage

```
compute_multipliers(qardl_results, x_names, tau)
```

Arguments

qardl_results	List of QARDL results for each quantile
x_names	Names of independent variables
tau	Vector of quantiles

Value

List with long-run and short-run multiplier matrices

decompose_multi_threshold *Multi-Threshold Decomposition*

Description

Decomposes a variable into partial sums of its positive and negative changes (threshold 0), starting at zero in the first observation.

Since 1.1.0 thresholds other than 0 give an error: the multi-threshold allocation of earlier versions was wrong (for example, with thresholds -2, 0 and 2 a change of -1 was split into pieces that did not sum to -1). Use `ardlverse::mtnardl()` for several thresholds.

Usage

```
decompose_multi_threshold(x, thresholds)
```

Arguments

x	Numeric vector
thresholds	Threshold values; only 0 is supported.

Value

List with regime components and names

decompose_variables *Decompose Variables into Positive and Negative Changes*

Description

Decomposes time series into cumulative positive and negative partial sums.

Usage

```
decompose_variables(data, variables)
```

Arguments

data	Data frame
variables	Variables to decompose

Value

List with positive and negative components

dynamic_multipliers *Cumulative Dynamic Multipliers*

Description

Cumulative dynamic multipliers of Shin, Yu and Greenwood-Nimmo (2014, eq. 9.13), m(+)_h and m(-)_h, h = 0, 1, ..., horizon: the effect on y at t + h of a unit change in the positive (negative) partial sum at t. They are computed from the estimated error correction model (current and lagged differences of y and of the partial sums, lagged levels), which is equivalent to the recursion of footnote 12 of the paper, and tend to the long-run multipliers L(+) and L(-) as h grows when the model is dynamically stable. The intercept and the Fourier terms do not enter the multipliers.

Usage

```
dynamic_multipliers(object, horizon = 20, decompose = NULL)
```

Arguments

object	An object returned by `fnardl()`, or its element `nardl_result`.
horizon	Largest horizon h.
decompose	Names of the decomposed variables. Defaults to those of the `fnardl()` object.

Details

A unit change in the negative partial sum is a unit change in x through a negative change; the response of y to a fall of x by one unit is `-m_negative`.

Value

A list with one data frame per decomposed variable, with columns `h`, `m_positive`, `m_negative` and `asymmetry` (`m_positive - m_negative`), and the attributes `lr_positive` and `lr_negative`.

References

Shin, Y., Yu, B. and Greenwood-Nimmo, M. (2014). doi:10.1007/9781489980083_9

`estimate_qardl` *Estimate Quantile ARDL Model*

Description

Estimates the Quantile ARDL model for a given quantile tau.

Usage

```
estimate_qardl(y, X, fourier, p, q, tau, case = 3)
```

Arguments

`y`	Dependent variable
`X`	Independent variables
`fourier`	Fourier terms
`p`	Lag for dependent variable
`q`	Lag for independent variables
`tau`	Quantile (0 < tau < 1)
`case`	Model case (1-5)

Value

List with estimation results

fadf_f_test *F-test for Linearity in Fourier ADF*

Description

Tests H0: gamma1 = gamma2 = 0 (no Fourier terms needed)

Usage

```
fadf_f_test(y, model, k, p)
```

Arguments

y	Time series
model	Model specification
k	Fourier frequency
p	Number of lags

Value

List with the F statistic, its critical values (Enders and Lee, 2012, Table 1b) and p_value = NA (the distribution is non-standard)

fnardl *Fourier Nonlinear ARDL Estimation*

Description

Estimates the Fourier Nonlinear ARDL (FNARDL) model that combines: - Nonlinear ARDL for asymmetric effects (Shin et al., 2014) - Fourier approximation for smooth structural breaks

Usage

```
fnardl(
  formula,
  data,
  decompose = NULL,
  max_p = 4,
  max_q = 4,
  max_k = 3,
  criterion = c("BIC", "AIC", "HQ"),
  case = 3,
  bootstrap = FALSE,
  n_boot = 1000,
  verbose = TRUE,
```

```
  fourier_k = NULL,
  boot_scheme = c("bvz", "mcnown"),
  seed = NULL
)
```

Arguments

formula	A formula of the form y ~ x1 + x2 + ...
data	A data frame containing the time series variables
decompose	Vector of variable names to decompose into positive/negative
max_p	Maximum lag for dependent variable
max_q	Maximum lag for independent variables
max_k	Maximum Fourier frequency
criterion	Information criterion ("AIC", "BIC", "HQ")
case	Model case; only 3 (unrestricted intercept, no trend) is supported.
bootstrap	Perform the recursive bootstrap bounds test (see Details).
n_boot	Number of bootstrap replications
verbose	Logical. Print progress messages (default: TRUE)
fourier_k	Optional Fourier frequency fixed by the user in advance; NULL (default) selects it from 1:max_k and the bootstrap re-selects it in every replication.
boot_scheme	"bvz" (default) or "mcnown"; see bootstrap_bounds_test.
seed	Optional seed for the bootstrap; the caller's random number state is restored afterwards.

Details

The positive and negative partial sums start at zero in the first observation (decompose_variables). The regression always contains Fourier terms, for which the Pesaran, Shin and Smith (2001) bounds are not valid: since 1.1.0 bounds_test reports the F, t and Find statistics with the bounds labelled as the bounds for the model without Fourier terms, not valid with Fourier terms, and no analytical verdict. With bootstrap = TRUE the recursive bootstrap generates x* and y*, rebuilds the partial sums from x*, holds the Fourier terms fixed, re-selects the Fourier frequency (unless fourier_k is given) and the lag orders in every replication (a package choice; Bertelli, Vacca and Zoia 2022 do not discuss re-selection), and combines the three tests with the AND rule (cointegration only if Fov, t and Find all reject).

Value

An object of class "fnardl"

References

Shin, Y., Yu, B. and Greenwood-Nimmo, M. (2014). Modelling asymmetric cointegration and dynamic multipliers in a nonlinear ARDL framework. In *Festschrift in Honor of Peter Schmidt*, 281-314. Springer, New York. doi:10.1007/9781489980083_9

Bertelli, S., Vacca, G. and Zoia, M. (2022). Bootstrap cointegration tests in ARDL models. *Economic Modelling*, 116, 105987. doi:10.1016/j.econmod.2022.105987

Examples

```
result <- fnardl(
  formula = gdp ~ oil_price + exchange_rate,
  data = macro_data,
  decompose = c("oil_price"),  # Decompose oil price into + and -
  max_k = 3,
  verbose = FALSE
)
summary(result)
plot(result, type = "asymmetry")
```

`fourier_adf` *Fourier ADF Unit Root Test*

Description

Performs the Fourier Augmented Dickey-Fuller test for unit roots with smooth structural breaks.

Usage

```
fourier_adf(y, max_k = 3, max_lag = 8, criterion = c("BIC", "AIC"))
```

Arguments

`y`	Time series vector
`max_k`	Maximum Fourier frequency
`max_lag`	Maximum number of lags for ADF
`criterion`	Lag selection criterion

Value

The result of `fourier_adf_test` with a constant, with the additional fields `optimal_k` and `t_statistic`.

`fourier_adf_test` *Fourier ADF Test*

Description

Tests for unit roots allowing for smooth structural breaks using Fourier approximation. Implements Enders and Lee (2012) methodology.

Usage

```
fourier_adf_test(
  y,
  model = c("c", "ct"),
  max_freq = 3,
  max_lag = NULL,
  criterion = c("AIC", "BIC", "t-sig"),
  verbose = TRUE
)
```

Arguments

y	Numeric vector of time series data
model	Model specification: "c" (constant), "ct" (constant + trend)
max_freq	Maximum Fourier frequency to test (default: 3)
max_lag	Maximum lag for ADF (default: NULL, auto-select)
criterion	Lag selection criterion ("AIC", "BIC", "t-sig")
verbose	Logical. Print progress messages (default: TRUE)

Value

Object of class "fadf" with test results

References

Enders, W., and Lee, J. (2012). The flexible Fourier form and Dickey-Fuller type unit root tests. Economics Letters, 117(1), 196-199.

Examples

```
set.seed(123)
y <- cumsum(rnorm(200))  # Random walk
result <- fourier_adf_test(y, model = "c", max_freq = 3)
print(result)
```

---

fourier_kpss_test *Fourier KPSS Test*

Description

Tests for stationarity allowing for smooth structural breaks. Implements Becker, Enders and Lee (2006) methodology.

Usage

```
fourier_kpss_test(y, model = c("c", "ct"), max_freq = 3, verbose = TRUE)
```

Arguments

y	Numeric vector of time series data
model	Model specification: "c" (constant), "ct" (constant + trend)
max_freq	Maximum Fourier frequency (default: 3)
verbose	Logical. Print progress messages (default: TRUE)

Value

Object of class "fkpss" with test results

References

Becker, R., Enders, W., and 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.

---

fourier_unit_root_analysis

Complete Unit Root Analysis

Description

Performs both Fourier ADF and Fourier KPSS tests for comprehensive unit root analysis.

Usage

```
fourier_unit_root_analysis(y, name = "Series", max_freq = 3, verbose = TRUE)
```

Arguments

y	Time series
name	Optional name for the series
max_freq	Maximum Fourier frequency
verbose	Logical. Print progress messages (default: TRUE)

Value

List with results from both tests and joint conclusion

`fqardl` *Fourier Quantile ARDL Estimation*

Description

Estimates the Fourier Quantile Autoregressive Distributed Lag (FQARDL) model. This methodology extends QARDL by incorporating Fourier trigonometric terms to capture smooth structural breaks without prior knowledge of break timing.

Usage

```
fqardl(
  formula,
  data,
  tau = c(0.25, 0.5, 0.75),
  max_p = 4,
  max_q = 4,
  max_k = 3,
  criterion = c("BIC", "AIC", "HQ"),
  case = 3,
  bootstrap = FALSE,
  n_boot = 1000,
  seed = NULL,
  verbose = TRUE,
  fourier_k = NULL,
  boot_scheme = c("bvz", "mcnown")
)
```

Arguments

`formula`	A formula of the form y ~ x1 + x2 + ...
`data`	A data frame containing the time series variables
`tau`	Numeric vector of quantiles to estimate (default: c(0.25, 0.5, 0.75))
`max_p`	Maximum lag for dependent variable (default: 4)
`max_q`	Maximum lag for independent variables (default: 4)
`max_k`	Maximum Fourier frequency to test (default: 3)
`criterion`	Information criterion for lag selection ("AIC", "BIC", "HQ")
`case`	Model case; only 3 (unrestricted intercept, no trend) is supported.
`bootstrap`	Logical, perform the recursive bootstrap bounds test (`bootstrap_bounds_test`).
`n_boot`	Number of bootstrap replications (default: 1000)
`seed`	Random seed, set with `set.seed()` at the start (the quantile regression standard errors and the bootstrap use random numbers).
`verbose`	Logical. Print progress messages (default: TRUE).

`fourier_k`	Optional Fourier frequency fixed by the user in advance. `NULL` (default) selects it from `1:max_k`; the bootstrap then re-selects it in every replication.
`boot_scheme`	Bootstrap scheme, `"bvz"` (Bertelli, Vacca and Zoia 2022, default) or `"mcnown"` (the Fov null for all statistics, McNown, Sam and Goh 2018, applied to the conditional ECM); see `bootstrap_bounds_test`.

Details

Inference on cointegration. The regression always contains Fourier terms, for which the Pesaran, Shin and Smith (2001) bounds are not valid. Since 1.1.0 `bounds_test` reports the Wald F and t statistics at each quantile with the PSS bounds labelled as the bounds for the model without Fourier terms, and gives no analytical verdict. Use `bootstrap = TRUE`: the recursive bootstrap of `bootstrap_bounds_test` with OLS conditional ECM statistics, the Fourier frequency (unless `fourier_k` is given) and the lag orders re-selected in every replication (a package choice), and the AND decision rule. The quantile-specific statistics have no published null distribution (Cho, Kim and Shin 2015) and are descriptive.

Value

An object of class "fqardl" containing:

`coefficients`	Estimated coefficients for each quantile
`long_run`	Long-run multipliers
`short_run`	Short-run multipliers
`optimal_k`	Optimal Fourier frequency
`optimal_lags`	Optimal lag structure
`bounds_test`	Wald bounds statistics by quantile (no verdict with Fourier terms; see Details)
`bootstrap`	Result of `bootstrap_bounds_test` when `bootstrap = TRUE`
`diagnostics`	Model diagnostics

References

Pesaran, M.H., Shin, Y. and Smith, R.J. (2001). Bounds testing approaches to the analysis of level relationships. *Journal of Applied Econometrics*, 16(3), 289-326. doi:10.1002/jae.616

Cho, J.S., Kim, T. and Shin, Y. (2015). Quantile cointegration in the autoregressive distributed-lag modeling framework. *Journal of Econometrics*, 188(1), 281-300. doi:10.1016/j.jeconom.2015.05.003

Bertelli, S., Vacca, G. and Zoia, M. (2022). Bootstrap cointegration tests in ARDL models. *Economic Modelling*, 116, 105987. doi:10.1016/j.econmod.2022.105987

Examples

```
data(macro_data)
result <- fqardl(gdp ~ inflation + interest_rate,
                 data = macro_data,
                 tau = c(0.25, 0.5, 0.75),
                 max_k = 3)
summary(result)
```

```
plot(result)
```

`generate_fnardl_report` *Generate FNARDL Report*

Description

Generates a comprehensive report with all plots and tables.

Usage

```
generate_fnardl_report(
  obj,
  file = "fnardl_report.html",
  horizon = 20,
  verbose = TRUE
)
```

Arguments

`obj`	FNARDL object
`file`	Output file path (HTML or PDF)
`horizon`	Horizon for dynamic multipliers
`verbose`	Logical. Print completion message (default: TRUE)

Value

No return value, called for side effects (generates output or plots)

`generate_fourier_terms` *Generate Fourier Trigonometric Terms*

Description

Creates sine and cosine terms for Fourier approximation of structural breaks. Based on Enders and Lee (2012) methodology.

Usage

```
generate_fourier_terms(n, k, cumulative = FALSE)
```

Arguments

n	Sample size (number of observations)
k	Fourier frequency (integer >= 1)
cumulative	If TRUE, includes all frequencies from 1 to k

Details

The Fourier terms are computed as:

$$sin(2\pi kt/T)$$

$$cos(2\pi kt/T)$$

where t is the time index and T is the sample size.

Value

A matrix with sine and cosine columns

---

macro_data *Simulated Macroeconomic Data with Structural Break*

Description

A simulated quarterly dataset containing GDP, inflation, and interest rate with a structural break.

Usage

```
macro_data
```

Format

A data frame with 100 rows and 5 variables: gdp, inflation, interest_rate, oil_price, exchange_rate.

Source

Simulated data for package demonstration

Examples

```
data(macro_data)
head(macro_data)
```

`mtnardl` *Multi-Threshold NARDL Estimation (deprecated)*

Description

Deprecated. `fqardl::mtnardl()` is deprecated and will be removed in fqardl 2.0.0; use `ardlverse::mtnardl()`. The first call in a session gives a warning. The conventions differ: the lag order p of `ardlverse::mtnardl()` equals p - 1 here (here p counts the lagged level of y together with the p - 1 lagged differences), and the thresholds are a single numeric vector there (a named list here).

Estimates the NARDL model with the changes of each decomposed regressor split at threshold 0 into positive and negative partial sums.

Usage

```
mtnardl(
  formula,
  data,
  decompose = NULL,
  thresholds = NULL,
  max_p = 4,
  max_q = 4,
  criterion = c("BIC", "AIC", "HQ"),
  case = 3,
  verbose = TRUE
)
```

Arguments

`formula`	A formula of the form y ~ x1 + x2 + ...
`data`	Data frame with time series
`decompose`	Variables to decompose
`thresholds`	Named list of threshold values for each variable; only 0 is supported since 1.1.0.
`max_p`	Maximum lag for dependent variable
`max_q`	Maximum lag for independent variables
`criterion`	Information criterion ("AIC", "BIC", "HQ")
`case`	Model case; only 3 (unrestricted intercept, no trend) is supported.
`verbose`	Logical. Print progress messages (default: TRUE)

Details

Changes in 1.1.0.

- The bounds F statistic uses only the lagged level of y and the lagged levels of the regressors (partial sums and undecomposed variables), with k equal to the number of level regressors. Up to 1.0.6 every coefficient whose name ended in `_lag1` entered the test, including `dy_lag1` and the lagged differences, and the table was read at the wrong k. When the selected p or q exceeds 1 the F statistic, the bounds and the verdict therefore differ from 1.0.6.
- Thresholds other than 0 give an error: the multi-threshold decomposition of earlier versions did not allocate the changes correctly (the regime pieces did not sum to the change of the variable).
- Only `case = 3` is supported; other cases give an error instead of being silently replaced by case 3.
- The returned object has class `c("fqardl_mtnardl", "mtnardl")` and the print and summary methods are registered for `"fqardl_mtnardl"` only, so that they no longer mask the methods of ardlverse.

The bounds are those of Pesaran, Shin and Smith (2001), Table CI(iii), at k equal to the number of level regressors. This choice of k for partial sums is a package choice, pending verification against Shin, Yu and Greenwood-Nimmo (2014) and Pal and Mitra (2016).

Value

Object of class `c("fqardl_mtnardl", "mtnardl")`

References

Shin, Y., Yu, B. and Greenwood-Nimmo, M. (2014). Modelling asymmetric cointegration and dynamic multipliers in a nonlinear ARDL framework. In *Festschrift in Honor of Peter Schmidt*, 281-314. Springer, New York. doi:10.1007/9781489980083_9

Examples

```
result <- suppressWarnings(mtnardl(
  formula = gdp ~ oil_price,
  data = oil_gdp_data,
  decompose = "oil_price",
  max_p = 2, max_q = 2,
  verbose = FALSE
))
summary(result)
```

`oil_gdp_data` *Simulated Oil Price and GDP Data with Asymmetric Effects*

Description

A simulated quarterly dataset where GDP responds asymmetrically to oil price changes.

Usage

```
oil_gdp_data
```

Format

A data frame with 200 rows and 3 variables: date, gdp, oil_price.

Source

Simulated data for package demonstration

Examples

```
data(oil_gdp_data)
head(oil_gdp_data)
```

---

perform_bounds_test *Wald Bounds Test for Cointegration*

Description

Computes the Pesaran, Shin and Smith (2001) bounds test from an estimated conditional error correction model, at every quantile supplied.

The F statistic is the Wald statistic for the joint null that all lagged level coefficients are zero, divided by the number of restrictions:

$$W = (Rb)'(RVR')^{-1}(Rb), \qquad F = W/m$$

with $m = k + 1$ in Cases I, III and V and $m = k + 2$ in Cases II and IV. The t statistic is the ratio on the lagged level of the dependent variable, which is negative under cointegration.

Usage

```
perform_bounds_test(qardl_results, n, k, case = 3)
```

Arguments

qardl_results	List of quantile estimation results from `estimate_qardl`. The test is computed for every element.
n	Sample size.
k	Number of regressors (excluding the dependent variable).
case	Model case. Only `case = 3` is supported; see Details.

Details

Only Case III is supported. In versions up to 1.0.2 the `case` argument was accepted and never used: the design matrix was always that of Case III (unrestricted intercept, no trend), while the critical values were switched. Cases 4 and 5 therefore compared a Case III model against Case IV and Case V tables. Rather than continue to report that, this release raises an error for any case other than 3. Proper case handling requires changing the design matrix, not the table, and is scheduled for 2.0.0.

Fourier caveat. The PSS critical values are tabulated for the model without Fourier terms. With Fourier terms in the regression the null distributions change and the PSS (2001) bounds do not apply with Fourier terms (Monte Carlo evidence; in a Monte Carlo with n = 100 and independent random walks the 5 percent decision of fqardl 1.0.6 rejected in 10 percent of samples, the fnardl decision in 47.5 percent). Since 1.1.0, whenever the design contains Fourier terms (columns named `sin_*` or `cos_*`, which is always the case for models estimated by `fqardl` and `fnardl`), no analytical verdict is given: `decision` says so, `bounds_valid` is `FALSE`, and the bounds are returned only for reference, labelled as the bounds for the model without Fourier terms, not valid with Fourier terms. Use the recursive bootstrap (`bootstrap = TRUE` in `fqardl`, see `bootstrap_bounds_test`); when `fqardl` or `fnardl` is called without the bootstrap, their decision text adds this advice.

Quantile caveat. The PSS critical values were simulated for the conditional mean. Applying them quantile by quantile has no distribution theory behind it (Cho, Kim and Shin 2015 give no bounds test for the null of no cointegration), so the quantile statistics are descriptive.

Value

A list with one element per quantile plus a `summary` data frame. Each per-quantile element contains `F_stat`, `t_stat`, the critical value bounds, `bounds_valid`, `bounds_note` and `decision`. Without Fourier terms the decision is three-way and reports the inconclusive region; with Fourier terms it states that no analytical verdict is available.

References

Pesaran, M.H., Shin, Y. and Smith, R.J. (2001). Bounds testing approaches to the analysis of level relationships. *Journal of Applied Econometrics*, 16(3), 289-326. doi:10.1002/jae.616

Cho, J.S., Kim, T. and Shin, Y. (2015). Quantile cointegration in the autoregressive distributed-lag modeling framework. *Journal of Econometrics*, 188(1), 281-300. doi:10.1016/j.jeconom.2015.05.003

`plot.fnardl` *Plot FNARDL Results*

Description

==========
Visualization Functions for FNARDL Dynamic Multiplier Plots and Asymmetry Analysis ==========
Plot FNARDL Results

Usage

```
## S3 method for class 'fnardl'
plot(
  x,
  type = c("asymmetry", "dynamic", "cumulative", "comparison"),
  variable = NULL,
  horizon = 20,
  ...
)
```

Arguments

x	An object of class "fnardl"
type	Type of plot
variable	Variable to plot
horizon	Horizon for dynamic multipliers
...	Additional arguments

Value

No return value, called for side effects (generates output or plots)

---

plot.fqardl *Plot FQARDL Results*

Description

Creates various diagnostic and result plots for FQARDL models.

Usage

```
## S3 method for class 'fqardl'
plot(
  x,
  type = c("coefficients", "multipliers", "3d", "heatmap", "residuals"),
  variable = NULL,
  ...
)
```

Arguments

x	An object of class "fqardl"
type	Type of plot: "coefficients", "multipliers", "3d", "heatmap", "residuals"
variable	Variable name for coefficient plots
...	Additional arguments passed to plotting functions

Value

A ggplot object or plotly object for 3D plots

`plot.fqardl_mtnardl` *Plot method for fqardl mtnardl objects*

Description

fqardl provides no plot for its (deprecated) `mtnardl` objects. This method stops with an informative error instead of letting `plot()` dispatch, through the inherited class `"mtnardl"`, to a method of another package that expects a different object.

Usage

```
## S3 method for class 'fqardl_mtnardl'
plot(x, ...)
```

Arguments

`x`	An object of class `"fqardl_mtnardl"`.
`...`	Not used.

Value

Does not return; always an error.

`plot_cumulative_multipliers`
Plot Cumulative Multipliers and Asymmetry

Description

Cumulative dynamic multipliers m(+)_h and m(-)_h of Shin, Yu and Greenwood-Nimmo (2014, eq. 9.13) and their difference m(+)_h - m(-)_h.

Usage

```
plot_cumulative_multipliers(obj, variable, horizon = 20)
```

Arguments

`obj`	FNARDL object
`variable`	Variable to plot
`horizon`	Number of periods

Value

A ggplot object.

`plot_dynamic_multipliers` *Plot Dynamic Multipliers*

Description

Cumulative dynamic multipliers m(+)_h and m(-)_h of Shin, Yu and Greenwood-Nimmo (2014, eq. 9.13), computed by `dynamic_multipliers`, with the long-run multipliers as dotted lines.

Usage

```
plot_dynamic_multipliers(obj, variable, horizon = 20)
```

Arguments

`obj`	FNARDL object
`variable`	Variable to plot
`horizon`	Number of periods

Value

A ggplot object.

`plot_persistence` *Plot Persistence Profile*

Description

Plots the persistence profile showing the adjustment path to long-run equilibrium after a shock.

Usage

```
plot_persistence(obj, horizons = 20)
```

Arguments

`obj`	FQARDL object
`horizons`	Number of periods for persistence profile

Value

ggplot object

quantile_wald_test *Quantile Wald Test for Coefficient Constancy*

Description

Tests whether coefficients are constant across quantiles.

Usage

```
quantile_wald_test(qardl_results, coef_name)
```

Arguments

qardl_results	List of QARDL results
coef_name	Name of coefficient to test

Value

List with test results

select_fourier_frequency *Select Optimal Fourier Frequency*

Description

Selects the optimal Fourier frequency k based on information criteria. Tests all frequencies from 1 to max_k and selects the one minimizing the chosen criterion.

Usage

```
select_fourier_frequency(y, X, max_k = 3, criterion = c("BIC", "AIC", "HQ"))
```

Arguments

y	Dependent variable vector
X	Matrix of independent variables
max_k	Maximum Fourier frequency to test
criterion	Information criterion ("AIC", "BIC", "HQ")

Value

A list containing:

optimal_k	The optimal Fourier frequency
ic_values	Information criterion values for each k
criterion	The criterion used

select_optimal_lags *Select Optimal Lag Structure*

Description

Selects optimal lag orders for ARDL model using grid search over all combinations of p and q.

Usage

```
select_optimal_lags(
  y,
  X,
  fourier,
  max_p,
  max_q,
  criterion = c("BIC", "AIC", "HQ")
)
```

Arguments

y	Dependent variable
X	Matrix of independent variables
fourier	Fourier terms matrix
max_p	Maximum lag for dependent variable
max_q	Maximum lag for independent variables
criterion	Information criterion

Value

List with optimal lags

test_asymmetry *Test for Asymmetry (Wald Tests)*

Description

Wald tests of Shin, Yu and Greenwood-Nimmo (2014, sections 9.2.2 and 9.2.4) for each decomposed variable of the FNARDL model:

- long-run symmetry, H0: L(+) = L(-), with L(+) = -theta(+)/rho and L(-) = -theta(-)/rho; the variance of L(+) - L(-) is obtained by the delta method; chi-squared with 1 degree of freedom;
- short-run symmetry in the pairwise form, H0: pi(+)_j = pi(-)_j for j = 0, ..., q - 1 (the coefficients of the current and lagged differences of the partial sums); chi-squared with q degrees of freedom;

- short-run symmetry in the additive form, H0: sum_j pi(+)_j = sum_j pi(-)_j; chi-squared with 1 degree of freedom.

With q = 1 the two short-run forms coincide.

Usage

```
test_asymmetry(nardl_result, decompose = NULL)
```

Arguments

`nardl_result`	An object returned by `fnardl()`, or its element `nardl_result`.
`decompose`	Names of the decomposed variables. Defaults to those of the `fnardl()` object.

Value

A list with one element per decomposed variable, containing `wald_stat`, `p_value` and `decision` (5 percent level) of the long-run test; `lr_positive` and `lr_negative` (L(+) and L(-)); `difference` (L(+) - L(-)); `coef_positive` and `coef_negative` (theta(+) and theta(-)); and `short_run_pairwise` and `short_run_additive`, each a list with `wald_stat`, `df`, `p_value` and `decision`.

References

Shin, Y., Yu, B. and Greenwood-Nimmo, M. (2014). Modelling asymmetric cointegration and dynamic multipliers in a nonlinear ARDL framework. In R. C. Sickles and W. C. Horrace (eds), Festschrift in Honor of Peter Schmidt, Springer, pp. 281-314. doi:10.1007/9781489980083_9

Index

∗ **datasets**
    macro_data, 18
    oil_gdp_data, 20
∗ **package**
    fqardl-package, 3

bootstrap_bounds_test, *3*, 3, *11*, *15, 16*, *22*

compute_asymmetric_multipliers, 6
compute_diagnostics, 6
compute_multipliers, 7

decompose_multi_threshold, 7
decompose_variables, 8
dynamic_multipliers, 8, *25*

estimate_qardl, 9

fadf_f_test, 10
fnardl, *3*, 10
fourier_adf, 12
fourier_adf_test, *3*, *12*, 12
fourier_kpss_test, *3*, 13
fourier_unit_root_analysis, 14
fqardl, *3*, 15, *22*
fqardl-package, 3

generate_fnardl_report, 17
generate_fourier_terms, 17

macro_data, 18
mtnardl, *3*, 19

oil_gdp_data, 20

perform_bounds_test, *3*, 21
plot.fnardl, 22
plot.fqardl, 23
plot.fqardl_mtnardl, 24
plot_cumulative_multipliers, 24
plot_dynamic_multipliers, 25
plot_persistence, 25

quantile_wald_test, 26

select_fourier_frequency, 26
select_optimal_lags, 27

test_asymmetry, 27