Package 'fqardl'

March 17, 2026

Type Package

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

Version 1.0.2

Date 2026-02-25

Author Muhammad Alkhalaf [aut, cre, cph] (ORCID: <https://orcid.org/0009-0002-2677-9246>), Merwan Roudane [ctb] (Original Stata/Python implementation)

Maintainer Muhammad Alkhalaf <muhammedalkhalaf@gmail.com>

Description Comprehensive implementation of advanced ARDL methodologies for cointegration analysis with structural breaks and asymmetric effects. Includes: (1) Fourier Quantile ARDL (FQARDL) - quantile regression with Fourier approximation for analyzing relationships across the conditional distribution; (2) Fourier Nonlinear ARDL (FNARDL) - asymmetric cointegration with partial sum decomposition following Shin, Yu & Greenwood-Nimmo (2014) <doi:10.1007/978-1-4899-8008-3_9>; (3) Multi-Threshold NARDL (MTNARDL) - multiple regime asymmetry analysis; (4) Fourier Unit Root Tests - ADF and KPSS tests with Fourier terms following Enders & Lee (2012) <doi:10.1016/j.econlet.2012.05.019> and Becker, Enders & Lee (2006) <doi:10.1111/j.1467-9892.2006.00490.x>. Features automatic lag and frequency selection, PSS bounds testing following Pesaran, Shin & Smith (2001) <doi:10.1002/jae.616>, bootstrap cointegration tests, Wald tests for asymmetry, dynamic multiplier computation, and publication-ready visualizations. Ported from Stata/Python by Dr. Merwan Roudane.

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

Repository CRAN

Date/Publication 2026-03-17 19:10:14 UTC

Contents

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
- `fourier_adf_test`: Fourier ADF unit root test
- `fourier_kpss_test`: Fourier KPSS stationarity test
- `perform_bounds_test`: PSS bounds testing

Author(s)

Muhammad Alkhalaf <contact@rufyqelngeh.com>

Contributor: Merwan Roudane (Original Stata/Python implementation)

References

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

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

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

bootstrap_bounds_test *Bootstrap Bounds Test*

Description

Performs bootstrap-based bounds test for cointegration, following McNown et al. (2018) methodology.

Usage

```
bootstrap_bounds_test(
  y,
  X,
  fourier,
  p,
  q,
  tau,
  case,
  n_boot = 1000,
  verbose = FALSE
)
```

Arguments

y	Dependent variable
X	Independent variables
fourier	Fourier terms
p	Lag for y
q	Lag for X
tau	Quantiles
case	Model case
n_boot	Number of bootstrap replications
verbose	Logical. Print progress messages (default: FALSE)

Value

List with bootstrap p-values

---

compute_asymmetric_multipliers

Compute Asymmetric Multipliers

Description

Compute Asymmetric Multipliers

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 and short-run asymmetric multipliers

`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 multiple regimes based on thresholds.

Usage

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

Arguments

x	Numeric vector
thresholds	Threshold values (must include 0)

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

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 F-statistic and p-value

`fnardl` *Fourier Nonlinear ARDL Estimation*

Description

Estimates Fourier Nonlinear ARDL models with asymmetric cointegration following Shin, Yu & Greenwood-Nimmo (2014).

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)
```

Arguments

`formula`	A formula specifying the model.
`data`	A data frame containing the variables.
`decompose`	Character vector of variables to decompose into positive/negative. Default NULL.
`max_p`	Maximum lag order for dependent variable.
`max_q`	Maximum lag order for independent variables.
`max_k`	Maximum Fourier frequency.
`criterion`	Information criterion for selection ("BIC", "AIC", or "HQ").
`case`	PSS case (1-5), default is 3.
`bootstrap`	Logical. Whether to perform bootstrap bounds test.
`n_boot`	Number of bootstrap replications.
`verbose`	Logical. Print progress messages (default: TRUE).

Value

An object of class "fnardl" containing estimation results.

Author(s)

Muhammad Alkhalaf

References

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

Examples

```
data(macro_data, package = "fqardl")
result <- fnardl(gdp ~ oil_price, data = macro_data, decompose = "oil_price")
summary(result)
```

`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

A list with test results

`fourier_adf_test` ==========
Fourier Unit Root Tests Based on Enders & Lee (2012) and Becker, Enders & Lee (2006) Ported from Python: Dr. Merwan Roudane R implementation: Muhammad Alkhalaf (Rufyq Elngeh)
==========
Fourier ADF Test

Description

Tests for unit roots allowing for smooth structural breaks using Fourier approximation. Implements Enders & 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., & 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 & 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., & 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)
```

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.
max_p	Maximum lag for dependent variable.
max_q	Maximum lag for independent variables.
max_k	Maximum Fourier frequency to test.
criterion	Information criterion for lag selection ("BIC", "AIC", "HQ").
case	Model case (1-5) following Pesaran et al. (2001).
bootstrap	Logical, perform bootstrap cointegration test.
n_boot	Number of bootstrap replications.
seed	Random seed for reproducibility.
verbose	Logical. Print progress messages (default: TRUE).

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	Results of bounds test for cointegration
diagnostics	Model diagnostics

Author(s)

Muhammad Alkhalaf

References

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

Examples

```
data(macro_data)
result <- fqardl(gdp ~ inflation + interest_rate,
                 data = macro_data,
                 tau = c(0.25, 0.5, 0.75))
summary(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` ========== *Fourier Approximation Functions For capturing smooth structural breaks* ==========

Generate Fourier Trigonometric Terms

Description

Creates sine and cosine terms for Fourier approximation of structural breaks. Based on Enders & 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 Nonlinear ARDL*

Description

Estimates Multi-Threshold NARDL models with multiple regime asymmetry.

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 specifying the model.
data	A data frame containing the variables.
decompose	Variable to decompose (default NULL).
thresholds	List of threshold values (default NULL).
max_p	Maximum lag for dependent variable.
max_q	Maximum lag for independent variables.
criterion	Information criterion for selection.
case	PSS bounds test case.
verbose	Logical. Print progress messages (default: TRUE).

Value

An object of class "mtnardl".

Author(s)

Muhammad Alkhalaf

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` ==========
Bounds Test for Cointegration Based on Pesaran, Shin & Smith (2001) With extensions for Quantile ARDL
==========
Perform Bounds Test for Cointegration

Description

Performs the PSS (2001) bounds test for cointegration in the ARDL framework. Tests the joint significance of the lagged level variables.

Usage

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

Arguments

`qardl_results`	List of QARDL estimation results
`n`	Sample size
`k`	Number of regressors
`case`	Model case (1-5)

Details

The five cases are:

- Case 1: No intercept, no trend
- Case 2: Restricted intercept, no trend
- Case 3: Unrestricted intercept, no trend (most common)
- Case 4: Unrestricted intercept, restricted trend
- Case 5: Unrestricted intercept, unrestricted trend

Value

List with bounds test results

plot.fnardl ==========
Visualization Functions for FNARDL Dynamic Multiplier Plots and Asymmetry Analysis
==========
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 ==========
Visualization Functions for FQARDL Publication-ready plots
==========
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_cumulative_multipliers *Plot Cumulative Multipliers*

Description

Plot Cumulative Multipliers

Usage

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

Arguments

obj	FNARDL object
variable	Variable to plot
horizon	Number of periods

Value

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

plot_dynamic_multipliers *Plot Dynamic Multipliers*

Description

Plot Dynamic Multipliers

Usage

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

Arguments

obj	FNARDL object
variable	Variable to plot
horizon	Number of periods

Value

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

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 *Quantile ARDL Estimation Functions Based on Cho et al. (2015) and extensions*

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 Test)*

Description

Test for Asymmetry (Wald Test)

Usage

```
test_asymmetry(nardl_result, decompose)
```

Arguments

`nardl_result`	NARDL estimation results
`decompose`	Decomposed variables

Value

List of Wald test results for each variable

Index