Package: mvardlurt (via r-universe)

November 2, 2026

Type Package

Title Multivariate ARDL Unit Root Test

Version 1.1.1

Date 2026-10-03

Description Implements the multivariate autoregressive distributed lag (ARDL) unit root test of Sam, McNown, Goh and Goh (2025) <doi:10.1080/03796205.2024.2439101>. The test augments the ADF regression with the lagged level, the current difference and lagged differences of one or more covariates so that cointegration between the series under test and the covariates is taken into account. The t statistic on the lagged level of the series and the joint F statistic on the lagged levels of the covariates are bootstrapped with the respective null imposed (residual bootstrap), giving critical values and p-values. Provides automatic lag selection via AIC or BIC, diagnostic plots, and the four-case classification of the order of integration of the series.

License GPL-3

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

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

Encoding UTF-8

RoxygenNote 7.3.2

Depends R (>= 4.0.0)

Imports grDevices, graphics, stats, utils

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Repository https://muhammedalkhalaf.r-universe.dev

Date/Publication 2026-10-03 17:04:32 UTC

RemoteUrl https://github.com/muhammedalkhalaf/mvardlurt

RemoteRef HEAD

RemoteSha 4b9fb2911d6748020e55e121650f635f26f239f0

Contents

mvardlurt-package 2
autoplot.mvardlurt 3
mvardlurt 4
plot.mvardlurt 7
print.mvardlurt 9

Index **11**

`mvardlurt-package` *Multivariate ARDL Unit Root Test*

Description

Implements the multivariate autoregressive distributed lag (ARDL) unit root test of Sam, McNown, Goh and Goh (2025). The test augments the ADF regression with the lagged level, the current difference and lagged differences of one or more covariates, so that a possible cointegrating relationship between the series under test and the covariates is taken into account. Critical values and p-values come from the residual bootstrap described in the paper.

Details

The main function is `mvardlurt`, which performs the multivariate ARDL unit root test. The package provides:

- Automatic lag selection via AIC or BIC on a common sample
- Residual bootstrap critical values and p-values with the null imposed (Section 4.2 of the paper)
- Three deterministic specifications (none, intercept, intercept + trend)
- One or several covariates (joint F test on their lagged levels)
- Four-case classification of the order of integration of y
- Comprehensive print and summary output
- Diagnostic plots

The test produces two statistics:

- **t-statistic**: Tests $H_0 : b_1 = 0$ on the lagged level of y
- **F-statistic**: Tests $H_0 : \beta_2 = 0$ jointly on the lagged levels of the covariates

The four cases of Section 3.2 of the paper are:

- Case I: neither rejected; y is I(1), no cointegration
- Case II: t rejected, F not rejected; y is I(0)
- Case III: t not rejected, F rejected; y is I(2)
- Case IV: both rejected; y is I(1) and cointegrated with x

Author(s)

Muhammad Alkhalaf

References

Sam, C. Y., McNown, R., Goh, S. K. and Goh, K. L. (2025). A multivariate autoregressive distributed lag unit root test. *Studies in Economics and Econometrics*, 49(1), 17-33. doi:10.1080/03796205.2024.2439101

See Also

mvardlurt

Examples

```
# Generate cointegrated data
set.seed(123)
n <- 200
x <- cumsum(rnorm(n))
y <- 0.5 * x + rnorm(n, sd = 0.5)

# Run the test
result <- mvardlurt(y, x, case = 3, fixlag = c(2, 2), nboot = 199)
print(result)
```

---

autoplot.mvardlurt *Create a Combined Diagnostic Plot for mvardlurt Objects*

Description

Creates a 2x2 panel of diagnostic plots for the multivariate ARDL unit root test including residuals vs fitted, Q-Q plot, residuals over time, and ACF.

Usage

```
## S3 method for class 'mvardlurt'
autoplot(x, ...)
```

Arguments

x	An object of class "mvardlurt".
...	Additional arguments passed to plotting functions.

Value

Invisibly returns x.

Author(s)

Muhammad Alkhalaf

See Also

mvardlurt, plot.mvardlurt

Examples

```
set.seed(123)
n <- 100
x <- cumsum(rnorm(n))
y <- 0.5 * x + rnorm(n, sd = 0.5)

result <- mvardlurt(y, x, fixlag = c(2, 2), nboot = 99)
autoplot.mvardlurt(result)
```

---

mvardlurt *Multivariate ARDL Unit Root Test*

Description

Implements the multivariate autoregressive distributed lag (ARDL) unit root test of Sam, McNown, Goh and Goh (2025). The test augments the ADF regression with the lagged level, the current difference and lagged differences of one or more covariates, so that a possible cointegrating relationship between the series under test and the covariates is taken into account. Critical values and p-values are obtained by the residual bootstrap of Section 4.2 of the paper.

Usage

```
mvardlurt(y, x, case = 3L, maxlag = 10L, ic = "aic",
          fixlag = NULL, nboot = 999L, level = 0.05,
          seed = NULL, boot = TRUE, reps = NULL)
```

Arguments

y	A numeric vector or time series. The series under test.
x	A numeric vector, time series or numeric matrix with one column per covariate (the I(1) forcing variables of the paper).
case	Integer. Deterministic specification: • 1: No deterministic terms • 3: Intercept only (default) • 5: Intercept and linear trend
maxlag	Integer. Maximum ARDL order for AIC/BIC selection. Default is 10. Must be between 1 and 12.
ic	Character. Information criterion for lag selection: "aic" (default) or "bic".

`fixlag`	Optional numeric vector of length 2, the fixed orders `c(p, q)` of the ARDL(p, q) model (see Details). Both must be at least 1. If provided, overrides automatic lag selection.
`nboot`	Integer. Number of bootstrap replications. Default is 999. Minimum is 99.
`level`	Numeric. Significance level of the two tests used for the case classification (0 to 1). Default is 0.05.
`seed`	Optional integer. If supplied, the random number generator is seeded with it for the bootstrap and the caller's RNG state is restored on exit. Default `NULL` leaves the RNG untouched.
`boot`	Logical. Whether to compute bootstrap critical values and p-values. Default is `TRUE`.
`reps`	Deprecated name for `nboot`, kept for backward compatibility.

Details

The test regression is equation (1) of the paper,

$$\Delta y_t = c_1 + c_2 t + b_1 y_{t-1} + \beta_2' x_{t-1} + \sum_{i=1}^{p-1} \phi_i \Delta y_{t-i} + \sum_{j=1}^{q-1} \Phi_j' \Delta x_{t-j} + \omega' \Delta x_t + u_t,$$

estimated by OLS on $t = \max(p, q) + 1, \ldots, n$. The orders follow equation (1) of the paper: an ARDL(p, q) model contains $p - 1$ lagged differences of y and $q - 1$ lagged differences of x in addition to the contemporaneous difference of x, so p and q are at least 1 and ARDL(1, 1) contains no lagged differences at all. The paper is not consistent on this point: equations (1) and (24) sum to $p - 1$ and $q - 1$, whereas Section 4.1 describes the ARDL(1, 1) data generating process as having one lagged difference of each variable and Table 4 reports an ARDL(0, 2) model with no lagged difference of y. This package follows equation (1); the paper's ARDL(0, 2) corresponds to `fixlag = c(1, 3)` here, and `fixlag = c(p, q)` in versions before 1.1.0 corresponds to `fixlag = c(p + 1, q + 1)` in 1.1.0 (apart from the contemporaneous difference of x, which earlier versions omitted).

Two statistics are computed (Section 3.3, equations (11) and (12) of the paper):

- the t statistic on y_{t-1} for $H_0 : b_1 = 0$ against $H_1 : b_1 < 0$ (left tailed);
- the F (Wald) statistic for the joint hypothesis $H_0 : \beta_2 = 0$ on the lagged levels of all covariates (right tailed). With one covariate this equals the squared t statistic on x_{t-1}.

Their null distributions depend on nuisance parameters, so both are bootstrapped separately with the respective null imposed: the restricted regression (without y_{t-1} for the t test, without x_{t-1} for the F test) is estimated, its recentred residuals are resampled with replacement, a bootstrap series y^* is generated recursively from the restricted estimated equation with the observed x held fixed, and the unrestricted regression is re-estimated on (y^*, x). The contemporaneous difference of x, which equation (24) of the paper omits although equation (1) has it, is included in the restricted regressions and in the bootstrap, following equation (1). The initial values $y_1^*, \ldots, y_{\max(p,q)}^*$ are set to the observed y, so the recursion starts at $t = \max(p, q) + 1$ with the observed history. Critical values are the percentiles of the `nboot` bootstrap statistics, and the bootstrap p-values are the proportions of bootstrap statistics at least as extreme as the observed ones.

The lag orders p and q of the bootstrap are those of the test regression, either fixed through `fixlag` or selected by AIC or BIC on the observed data. They are held fixed in every bootstrap replication:

the restricted regression is estimated with these orders and the unrestricted regression is re-estimated on (y^*, x) with the same orders, and the lag selection is not repeated on the bootstrap samples. This follows Section 4.2 of the paper, where the restricted regression (24) is estimated with the lag length of the data generating process and equation (1) is re-estimated on y_t^* and x_t. When the orders are selected from the data, the bootstrap distribution therefore does not reflect the selection step. In a Monte Carlo experiment by the package author (n = 100, y and x independent random walks, 200 replications, 199 bootstrap replications, 5 percent level) the rejection frequencies of the bootstrap tests were 0.055 to 0.070 with fixed lag orders and 0.080 to 0.100 with AIC selection (Monte Carlo standard error about 0.02). Fixing the orders through `fixlag` avoids this source of size distortion.

The decision at significance level `level` classifies the series according to the four cases of Section 3.2 of the paper:

- **Case I** (t not rejected, F not rejected): nonstationary process, no cointegration; y is I(1).
- **Case II** (t rejected, F not rejected): stationary process; y is I(0).
- **Case III** (t not rejected, F rejected): degenerate lagged dependent variable; y is I(2).
- **Case IV** (both rejected): nonstationary process with cointegration between y and x; y is I(1).

The Monte Carlo experiments of the paper use a 5 percent level for both tests, which is the default here. In the empirical application of the paper the F statistic is reported as significant at the 10 percent level and cointegration is concluded (Section 6); set `level = 0.10` to reproduce that convention. Before version 1.1.0 `level` was a confidence level (default 0.95); values of 0.5 or more are now rejected.

Lag orders are either fixed through `fixlag` or selected by AIC or BIC over p = 1, ..., `maxlag` and q = 1, ..., `maxlag`. All candidate models are estimated on the same observations (t = `maxlag` + 1, ..., n) so that the criteria are comparable; the selected model is then re-estimated on its own full sample.

Value

An object of class `"mvardlurt"` containing:

`tstat`	t statistic for $H_0 : b_1 = 0$
`fstat`	F statistic for $H_0 : \beta_2 = 0$
`t_pval`, `f_pval`	Bootstrap p-values (NA when `boot = FALSE`)
`t_cv`, `f_cv`	Bootstrap critical values at the 10, 5, 2.5 and 1 percent levels
`t_boot`, `f_boot`	The bootstrap distributions
`b1`, `b1_se`	Estimate and standard error of b_1
`beta2`, `beta2_se`	Estimates and standard errors of β_2 (one per covariate)
`lr_mult`	Long-run multipliers $-\beta_2/b_1$
`opt_p`, `opt_q`	Orders of the ARDL(p, q) model used
`case`, `casename`	Deterministic case and its description
`nboot`	Number of bootstrap replications
`nobs`	Number of observations in the regression
`aic`, `bic`, `r_squared`, `adj_r_squared`	Fit statistics of the model

`ic_table`	Matrix of information criteria for all (p, q) candidates (NULL with `fixlag`)
`decision`	List with the decision at `level`: `reject_t`, `reject_f`, `case_num`, `case_result`, `integration` (the implied order of integration of y) and the significance codes
`model`	The fitted `lm` object, for inspection (its call refers to an internal data frame; to refit use `lm(formula(object$model), data = object$data)`)
`data`	The data frame of the test regression
`y`, `x`	The data used (x as a matrix)
`residuals`	Residuals from the fitted model

Author(s)

Muhammad Alkhalaf

References

Sam, C. Y., McNown, R., Goh, S. K. and Goh, K. L. (2025). A multivariate autoregressive distributed lag unit root test. *Studies in Economics and Econometrics*, 49(1), 17-33. doi:10.1080/03796205.2024.2439101

Examples

```
# Cointegrated data: y is I(1) through x (Case IV)
set.seed(123)
n <- 200
x <- cumsum(rnorm(n))
y <- 0.5 * x + rnorm(n, sd = 0.5)

result <- mvardlurt(y, x, case = 3, fixlag = c(2, 2), nboot = 199)
print(result)
summary(result)

# Automatic lag selection with two covariates
X <- cbind(x, z = cumsum(rnorm(n)))
result2 <- mvardlurt(y, X, maxlag = 3, nboot = 199)
result2$fstat
```

---

plot.mvardlurt *Plot Method for mvardlurt Objects*

Description

Creates diagnostic plots for the multivariate ARDL unit root test.

Usage

```
## S3 method for class 'mvardlurt'
plot(x, which = c(1, 2, 3, 4),
     ask = (length(which) > 1 && dev.interactive()), ...)
```

Arguments

x	An object of class "mvardlurt".
which	Integer vector indicating which plots to produce: • 1: Residuals vs Fitted • 2: Q-Q plot of residuals • 3: Residuals over time • 4: ACF of residuals • 5: Time series of y and the covariates • 6: Information criterion surface Default is c(1, 2, 3, 4).
ask	Logical. If TRUE, prompt before each plot. Default is TRUE when multiple plots are requested in an interactive session.
...	Additional arguments passed to plotting functions.

Value

Invisibly returns x.

Author(s)

Muhammad Alkhalaf

See Also

mvardlurt, autoplot.mvardlurt

Examples

```
set.seed(123)
n <- 100
x <- cumsum(rnorm(n))
y <- 0.5 * x + rnorm(n, sd = 0.5)

result <- mvardlurt(y, x, maxlag = 3, nboot = 99)

# Default diagnostic plots
plot(result, ask = FALSE)

# IC surface plot
plot(result, which = 6)
```

`print.mvardlurt` *Methods for mvardlurt Objects*

Description

Print, summary, and accessor methods for objects of class `"mvardlurt"`.

Usage

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

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

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

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

## S3 method for class 'mvardlurt'
fitted(object, ...)
```

Arguments

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

Value

`print` and `summary` invisibly return `x`/`object`.

`coef` returns a named numeric vector with `b1`, the `beta2` coefficient of each covariate and the corresponding long-run multipliers `lr_mult`.

`residuals` returns the numeric vector of residuals from the fitted model.

`fitted` returns the numeric vector of fitted values.

Author(s)

Muhammad Alkhalaf

See Also

`mvardlurt`

Examples

```
set.seed(123)
n <- 100
x <- cumsum(rnorm(n))
y <- 0.5 * x + rnorm(n, sd = 0.5)

result <- mvardlurt(y, x, fixlag = c(2, 2), nboot = 99)

# Print method
print(result)

# Summary method
summary(result)

# Extract coefficients
coef(result)

# Extract residuals
head(residuals(result))

# Extract fitted values
head(fitted(result))
```

Index

∗ **package**
mvardlurt-package, 2

autoplot.mvardlurt, 3, *8*

coef.mvardlurt *(*print.mvardlurt*)*, 9

fitted.mvardlurt *(*print.mvardlurt*)*, 9

mvardlurt, *2*, *3*, 4, *4*, *8*, *9*
mvardlurt-package, 2

plot.mvardlurt, *4*, 7
print.mvardlurt, 9

residuals.mvardlurt *(*print.mvardlurt*)*, 9

summary.mvardlurt *(*print.mvardlurt*)*, 9