Package 'fUnitRoots'

December 19, 2025

Title Rmetrics - Modelling Trends and Unit Roots

Version 4052.82

Description Provides four addons for analyzing trends and unit roots in financial time series: (i) functions for the density and probability of the augmented Dickey-Fuller Test, (ii) functions for the density and probability of MacKinnon's unit root test statistics, (iii) reimplementations for the ADF and MacKinnon Test, and (iv) an 'urca' Unit Root Test Interface for Pfaff's unit root test suite.

Depends R (>= 2.15.1)

Imports timeSeries, fBasics, urca, graphics, methods, stats, utils

Suggests RUnit, interp

License GPL (>= 2)

URL https://geobosh.github.io/fUnitRootsDoc/ (doc),
https://CRAN.R-project.org/package=fUnitRoots,
https://www.rmetrics.org

BugReports
https://r-forge.r-project.org/tracker/?atid=633&group_id=156&func=browse

NeedsCompilation yes

Author Diethelm Wuertz [aut] (original code),
Tobias Setz [aut],
Yohan Chalabi [aut],
Georgi N. Boshnakov [cre] (ORCID:
<https://orcid.org/0000-0003-2839-346X>)

Maintainer Georgi N. Boshnakov <georgi.boshnakov@manchester.ac.uk>

Repository CRAN

Date/Publication 2025-12-19 13:50:14 UTC

Contents

`fUnitRoots-package` *Modelling Trends and Unit Roots*

Description

The Rmetrics "fUnitRoots" package is a collection of functions to model trends and to analyze unit roots.

1 Introduction

The 'fUnitroots' provides four addons for analyzing trends and unit roots in financial time series: (i) functions for the density and probability of the augmented Dickey-Fuller Test, (ii) functions for the density and probability of MacKinnon's unit root test statistics, (iii) reimplementations for the ADF and MacKinnon Test, and (iv) an 'urca' Unit Root Test Interface for Pfaff's unit root test suite.

2 Dickey-Fuller p Values

The section provides functions to compute the distribution and quantile functions for the ADF unit root test statistics.

```
padf           returns the cumulative probability for the ADF test
qadf           returns the quantiles for the ADF test
adfTable       tables p values for ADF test
```

3 Mc Kinnon p Values

The section provides functions to compute the distribution and quantile functions for MacKinnon's unit root test statistics.

```
punitroot      returns the cumulative probability
qunitroot      returns the quantiles of the unit root test statistics
unitrootTable  tables p values from MacKinnon's response surface
```

4 Unit Root Time Series Tests

This section provides two functions for unit root testing of financial time series, the ADF tests based on Banerjee's et al. tables and the unit root tests based on J.G. McKinnons' tables:

```
adfTest         augmented Dickey-Fuller test for unit roots
unitrootTest    the same based on McKinnons's test statistics
```

5 "urca"" Unit Root Test Interface

This is an interface to the unitroot tests suite implemented by Bernhard Pfaff available through the R package "**urca**"

```
urdfTest        Augmented Dickey-Fuller test for unit roots
urersTest       Elliott--Rothenberg-Stock test for unit roots
urkpssTest      KPSS unit root test for stationarity
urppTest        Phillips-Perron test for unit roots
urspTest        Schmidt-Phillips test for unit roots
urzaTest        Zivot-Andrews test for unit roots
```

About Rmetrics

The `fUnitroots` Rmetrics package is written for educational support in teaching "Computational Finance and Financial Engineering" and licensed under the GPL.

`DickeyFullerPValues` *Dickey-Fuller p Values*

Description

A collection and description of functions to compute the distribution and quantile function for the ADF unit root test statistics.

Usage

```
padf(q, N = Inf, trend = c("nc", "c", "ct"), statistic = c("t", "n"))
qadf(p, N = Inf, trend = c("nc", "c", "ct"), statistic = c("t", "n"))

adfTable(trend = c("nc", "c", "ct"), statistic = c("t", "n"),
    includeInf = TRUE)
```

Arguments

q	vector of quantiles or test statistics. Missing values are allowed.
p	vector of probabilities. Missing values are allowed.
N	the number of observations in the sample from which the quantiles are to be computed.
trend	a character string describing the regression from which the quantiles are to be computed. Valid choices are: "nc" for a regression with no intercept (constant) nor time trend, "c" for a regression with an intercept (constant) but no time trend, and "ct" for a regression with an intercept (constant) and a time trend. The default is "c".
statistic	a character string describing the type of test statistic. Valid choices are "t" for t-statistic and "n" for normalized statistic, sometimes referred to as the rho-statistic. The default is "t".
includeInf	a logical flag. Should the asymptotic value be included into the table?

Details

padf computes cumulative probabilities for the ADF test.

qadf computes quantiles for the ADF test.

With sufficiently fine grid for the first argument, padf and qadf compute, respectively, cumulative distribution functions and quantile functions of ADF test statistics.

adfTable produces tables of p-values for ADF tests.

Value

for padf and qadf, a named numeric vector with attribute "control" holding N,

for adfTable, an object from class "gridData", which is a list with the following components:

x	the values of N (length of the time series) for which the statistics are provided,
y	quantiles for which the statistics are provided,
Table	a matrix with one row for each N in x and one column for each quantile in y.

The "gridData" object contains also attribute "control" with information about the requested test.

Note

padf and qadf is based on the tables from A. Banerjee et al. (1993). Interpolation is used For value of N not in the tables. For small N (N < 25) the result is NA.

Author(s)

Diethelm Wuertz for the Rmetrics R-port.

References

Banerjee A., Dolado J.J., Galbraith J.W., Hendry D.F. (1993); *Cointegration, Error Correction, and the Econometric Analysis of Non-Stationary Data*, Oxford University Press, Oxford.

Dickey, D.A., Fuller, W.A. (1979); *Distribution of the estimators for autoregressive time series with a unit root*, Journal of the American Statistical Association 74, 427–431.

Examples

```
padf(q = -2:2, N = 25)
qadf(p = (1:9)/10, N = 25)
adfTable()
```

MacKinnonPValues *MacKinnon's Unit Root p Values*

Description

A collection and description of functions to compute the distribution and quantile function for MacKinnon's unit root test statistics.

Usage

```
punitroot(q, N = Inf, trend = c("c", "nc", "ct", "ctt"),
    statistic = c("t", "n"), na.rm = FALSE)
qunitroot(p, N = Inf, trend = c("c", "nc", "ct", "ctt"),
    statistic = c("t", "n"), na.rm = FALSE)

unitrootTable(trend = c("c", "nc", "ct", "ctt"), statistic = c("t", "n"))
```

Arguments

`q`	vector of quantiles or test statistics. Missing values are allowed.
`p`	a numeric vector of probabilities. Missing values are allowed.
`N`	the number of observations in the sample from which the quantiles are to be computed.
`na.rm`	a logical value. If set to `TRUE`, missing values will be removed, otherwise not. The default is `FALSE`.
`statistic`	a character string describing the type of test statistic. Valid choices are `"t"` for t-statistic, and `"n"` for normalized statistic, sometimes referred to as the rho-statistic. The default is `"t"`.
`trend`	a character string describing the regression from which the quantiles are to be computed. Valid choices are: `"nc"` for a regression with no intercept (constant) nor time trend, `"c"` for a regression with an intercept (constant) but no time trend, and `"ct"` for a regression with an intercept (constant) and a time trend. The default is `"c"`.

Details

punitroot computes the cumulative probability of the asymptotic or finite sample distribution of the unit root test statistics.

qunitroot computes the quantiles of the asymptotic or finite sample distribution of the unit root test statistics, given the probabilities.

unitrootTable produces tables of p-values from MacKinnon's response surface.

Value

for punitroot and qunitroot, a numeric vector,

for unitrootTable, a matrix with attribute "control" containing information about the type of test.

Note

The function punitroot and qunitroot use Fortran routines and the response surface approach from J.G. MacKinnon (1988). Many thanks to J.G. MacKinnon putting his code and tables under the GPL license, which made this implementation possible.

Author(s)

J.G. MacKinnon for the underlying Fortran routine and the tables,
Diethelm Wuertz for the Rmetrics R-port.

References

Dickey, D.A., Fuller, W.A. (1979); *Distribution of the estimators for autoregressive time series with a unit root*, Journal of the American Statistical Association 74, 427–431.

MacKinnon, J.G. (1996); *Numerical distribution functions for unit root and cointegration tests*, Journal of Applied Econometrics 11, 601–618.

Phillips, P.C.B., Perron, P. (1988); *Testing for a unit root in time series regression*, Biometrika 75, 335–346.

Examples

```
## Asymptotic quantile of t-statistic
qunitroot(0.95, trend = "nc", statistic = "t")
## 1st argument a vector
qunitroot(c(0.90, 0.95), trend = "nc", statistic = "t")

## Finite sample quantile of n-statistic
qunitroot(0.95, N = 100, trend = "nc", statistic = "n")


## Asymptotic cumulative probability of t-statistic
punitroot(1.2836, trend = "nc", statistic = "t")

## Finite sample cumulative probability of n-statistic
punitroot(1.2836, N = 100, trend = "nc", statistic = "n")
```

```
## Mac Kinnon's unitrootTable
unitrootTable(trend = "nc")
```

`UnitrootTests` *Unit root time series tests*

Description

Carry out augmented Dickey-Fuller tests for unit roots based on Banerjee's et al. tables and on J.G. McKinnons' numerical distribution functions.

Usage

```
unitrootTest(x, lags = 1, type = c("nc", "c", "ct"), title = NULL,
    description = NULL)

adfTest(x, lags = 1, type = c("nc", "c", "ct"), title = NULL,
    description = NULL)
```

Arguments

`x`	a numeric vector or time series object.
`lags`	the maximum number of lags used for error term correction.
`type`	a character string describing the type of the unit root regression. Valid choices are `"nc"` for a regression with no intercept (constant) nor time trend, `"c"` for a regression with an intercept (constant) but no time trend, `"ct"` for a regression with an intercept (constant) and a time trend. The default is `"c"`.
`title`	a character string which allows for a project title.
`description`	a character string which allows for a brief description.

Details

`adfTest()` computes test statistics and p-values along the implementation from Trapletti's augmented Dickey-Fuller test for unit roots. In contrast to Trapletti's function, three kind of test types can be selected.

`unitrootTest()` computes test statistics and p-values using McKinnon's response surface approach.

Value

an object from S4 class `"fHTEST"` with the following slots:

`@call`	the function call.
`@data`	a data frame with the input data.

`@data.name`	a character string giving the name of the data frame.
`@test`	a list object which holds the output of the underlying test function.
`@title`	a character string with the name of the test.
`@description`	a character string with a brief description of the test.

The entries of the `@test` slot include the following components:

`$statistic`	the value of the test statistic.
`$parameter`	the lag order.
`$p.value`	the p-value of the test.
`$method`	a character string indicating what type of test was performed.
`$data.name`	a character string giving the name of the data.
`$alternative`	a character string describing the alternative hypothesis.
`$name`	the name of the underlying function, which may be wrapped.
`$output`	additional test results to be printed.

Author(s)

Adrian Trapletti for the tests adapted from R's "tseries" package,
Diethelm Wuertz for the Rmetrics R-port.

References

Banerjee A., Dolado J.J., Galbraith J.W., Hendry D.F. (1993); *Cointegration, Error Correction, and the Econometric Analysis of Non-Stationary Data*, Oxford University Press, Oxford.

Dickey, D.A., Fuller, W.A. (1979); *Distribution of the estimators for autoregressive time series with a unit root*, Journal of the American Statistical Association 74, 427–431.

MacKinnon, J.G. (1996); *Numerical distribution functions for unit root and cointegration tests*, Journal of Applied Econometrics 11, 601–618.

Said S.E., Dickey D.A. (1984); *Testing for Unit Roots in Autoregressive-Moving Average Models of Unknown Order*, Biometrika 71, 599–607.

Examples

```
## a time series which contains no unit-root:
x <- rnorm(1000)
## a time series which contains a unit-root:
y <- cumsum(c(0, x))

adfTest(x)
adfTest(y)

unitrootTest(x)
unitrootTest(y)
```

`UnitrootUrcaInterface` *Unit Root Time Series Tests*

Description

A collection and description of functions for unit root testing. This is an interface to the unitroot tests implemented by B. Pfaff available through the R package **urca** which is required here.

Added functions based on the **urca** package include:

`urdfTest`	Augmented Dickey-Fuller test for unit roots,
`urersTest`	Elliott-Rothenberg-Stock test for unit roots,
`urkpssTest`	KPSS unit root test for stationarity,
`urppTest`	Phillips-Perron test for unit roots,
`urspTest`	Schmidt-Phillips test for unit roots,
`urzaTest`	Zivot-Andrews test for unit roots.

Usage

```
urdfTest(x, lags = 1, type = c("nc", "c", "ct"), doplot = TRUE)
urersTest(x, type = c("DF-GLS", "P-test"), model = c("constant", "trend"),
          lag.max = 4, doplot = TRUE)
urkpssTest(x, type = c("mu", "tau"), lags = c("short", "long", "nil"),
           use.lag = NULL, doplot = TRUE)
urppTest(x, type = c("Z-alpha", "Z-tau"), model = c("constant", "trend"),
         lags = c("short", "long"), use.lag = NULL, doplot = TRUE)
urspTest(x, type = c("tau", "rho"), pol.deg = c(1, 2, 3, 4),
         signif = c(0.01, 0.05, 0.1), doplot = TRUE)
urzaTest(x, model = c("intercept", "trend", "both"), lag, doplot = TRUE)
```

Arguments

`x`	a numeric vector or time series object.
`lags`	[urkpssTest][urppTest] - the maximum number of lags used for error term correction.
`type`	[urkpssTest] - a character string which denotes the type of deterministic part, either `"mu"`, the default, or `"tau"`. [urppTest] - a character string which specifies the test type, either `"Z-alpha"`, the default, or `"Z-tau"`. [urspTest] - a character string which specifies the test type, either `"tau"`, the default, or `"rho"`.
`doplot`	[ur*Test] - a logical flag, by default `TRUE`. Should a diagnostical plot be displayed?

`model` [urersTest] -
a character string dennoting the deterministic model used for detrending, either `"constant"`, the default, or `"trend"`.
[urppTest] -
a character string which determines the deterministic part in the test regression, either `"constant"`, the default, or `"trend"`.
[urzaTest] -
a character string specifying if the potential break occured in either the `"intercept"`, the linear `"trend"` or in `"both"`.

`lag.max` [urersTest] -
the maximum numbers of lags used for testing of a decent lag truncation for the `"P-test"`, BIC used, or the maximum number of lagged differences to be included in the test regression for `"DF-GLS"`.

`use.lag` [urkpssTest] -
a character string specifying the number of lags. Allowed arguments are `lags=c("short", "long", "nil")`, for more information see the details section.
[urppTest] -
Use of a different lag number, specified by the user.

`pol.deg` [urspTest] -
the polynomial degree in the test regression.

`signif` [urspTest] -
the significance level for the critical value of the test statistic.

`lag` [urzaTest] -
the highest number of lagged endogenous differenced variables to be included in the test regression.

Details

Unit Root Tests from Berhard Pfaff's "urca" Package:

Elliott-Rothenberg-Stock Test for Unit Roots:
To improve the power of the unit root test, Elliot, Rothenberg and Stock proposed a local to unity detrending of the time series. ERS developed a feasible point optimal test, `"P-test"`, which takes serial correlation of the error term into account. The second test type is the `"DF-GLS"` test, which is an ADF-type test applied to the detrended data without intercept. Critical values for this test are taken from MacKinnon in case of `model="constant"` and else from Table 1 of Elliot, Rothenberg and Stock.
`[urca:ur.ers]`

KPSS Test for Unit Roots:
Performs the KPSS unit root test, where the Null hypothesis is stationarity. The test types specify as deterministic component either a constant `"mu"` or a constant with linear trend `"tau"`. `lags="short"` sets the number of lags to *root 4 of [4 times (n/100)*, whereas `lags="long"` sets the number of lags to *root 4 of [12 times (n/100)]*. If `lags="nil"` is choosen, then no error correction is made. Furthermore, one can specify a different number of maximum lags by setting use.lag accordingly.
`[urca:ur.kpss]`

Phillips-Perron Test for Unit Roots:
Performs the Phillips and Perron unit root test. Beside the Z statistics Z-alpha and Z-tau, the Z statistics for the deterministic part of the test regression are computed, too. For correction of the error term a Bartlett window is used.
`[urca:ur.pp]`

Schmidt-Phillips Test for Unit Roots:
Performs the Schmidt and Phillips unit root test, where under the Null and Alternative Hypothesis the coefficients of the deterministic variables are included. Two test types are available: the `"rho-test"` and the `"tau-test"`. Both tests are extracted from the LM principle.
`[urca:ur.sp]`

Zivot-Andrews Test for Unit Roots:
Performs the Zivot and Andrews unit root test, which allows a break at an unknown point in either the intercept, the linear trend or in both. This test is based upon the recursive estimation of a test regression. The test statistic is defined as the minimum t-statistic of the coeffcient of the lagged endogenous variable.
`[urca:ur.za]`

Value

All tests return an object of class `"fHTEST"` with the following slots:

`@call`	the function call.
`@data`	a data frame with the input data.
`@data.name`	a character string giving the name of the data frame.
`@test`	a list object which holds the output of the underlying test function.
`@title`	a character string with the name of the test.
`@description`	a character string with a brief description of the test.

The entries of the `@test` slot include the following components:

`$statistic`	the value of the test statistic.
`$parameter`	the lag order.
`$p.value`	the p-value of the test.
`$method`	a character string indicating what type of test was performed.
`$data.name`	a character string giving the name of the data.
`$alternative`	a character string describing the alternative hypothesis.
`$name`	the name of the underlying function, which may be wrapped.
`$output`	additional test results to be printed.

Note

The functions `ur*Test()` fullfill the naming conventions of Rmetrics, return an S4 object named `fHTEST` as any other hypothesis test from Rmetrics, and allow for `timeSeries` objects as input. These are the only differences to the original implementation of the functions.

Fur further details we refer to the manual pages of the **urca** package which is required for all these.

Author(s)

Bernhard Pfaff for the tests implemented in R's "**urca**" package,
Diethelm Wuertz for the Rmetrics R-port.

References

Banerjee A., Dolado J.J., Galbraith J.W., Hendry D.F. (1993); *Cointegration, Error Correction, and the Econometric Analysis of Non-Stationary Data*, Oxford University Press, Oxford.

Dickey, D.A., Fuller, W.A. (1979); *Distribution of the estimators for autoregressive time series with a unit root*, Journal of the American Statistical Association 74, 427–431.

Kwiatkowski D., Phillips P.C.B, Schmidt P., Shin Y. (1992); *Testing the Null Hypothesis of Stationarity against the Alternative of a Unit Root*, Journal of Econometrics 54, 159–178.

Perron P. (1988); *Trends and Random Walks in Macroeconomic Time Series*, Journal of Economic Dynamics and Control 12, 297–332.

Phillips P.C.B., Perron P. (1988); *Testing for a unit root in time series regression*, Biometrika 75, 335–346.

Said S.E., Dickey D.A. (1984); *Testing for Unit Roots in Autoregressive-Moving Average Models of Unknown Order*, Biometrika 71, 599–607.

Schwert G.W. (1989); *Tests for Unit Roots: A Monte Carlo Investigation*, Journal of Business and Economic Statistics 2, 147–159.

Examples

```
## a time series which contains no unit-root:
x <- rnorm(1000)
## a time series which contains a unit-root:
y <- cumsum(c(0, x))

## ERS Test:
if(require("urca")) {
   urersTest(x)
   urersTest(y)
}
```

Index