# Package ‘nortsTest’

July 27, 2020

**Title** Assessing Normality of Stationary Process

**Version** 1.0.0

**Description** Despite that several tests for normality in stationary processes have been proposed
in the literature, consistent implementations of these tests in programming languages are limited.
Four normality test are implemented. The Lobato and Ve-
lasco's, Epps, Psaradakis and Vavra, and the
random projections tests for stationary process. Some other diagnostics such as, unit root test for
stationarity, seasonal tests for seasonality, and arch effect test for volatility; are also performed.
The package also offers residual diagnostic for linear time series models developed in sev-
eral packages.

**License** LGPL

**Encoding** UTF-8

**LazyData** true

**NeedsCompilation** no

**RoxygenNote** 7.1.0

**Depends** R (>= 3.5.0), methods

**Imports** forecast, nortest, ggplot2, gridExtra, tseries, uroot, MASS,
zoo

**URL** https://github.com/asael697/nortsTest

**BugReports** https://github.com/asael697/nortsTest/issues

**Author** Asael Alonzo Matamoros [aut, cre],
Alicia Nieto-Reyes [aut],
Rob Hyndman [ctb],
Mitchell O'Hara-Wild [ctb],
Trapletti A. [ctb]

**Maintainer** Asael Alonzo Matamoros <asael.alonzo@gmail.com>

**Repository** CRAN

**Date/Publication** 2020-07-27 11:50:03 UTC

# R topics documented:



---

`nortsTest-package` *'Assessing Normality of Stationary Process.'*

---

## Description

Despite that several tests for normality in stationary processes have been proposed in the literature, consistent implementations of these tests in programming languages are limited. Four normality test are implemented. The Lobato and Velasco's, Epps, Psaradakis and Vavra, and the random projections tests for stationary process. Some other diagnostics such as, unit root test for stationarity, seasonal tests for seasonality, and arch effect test for volatility; are also performed. The package also offers residual diagnostic for linear time series models developed in several packages.

## Details

We present four main functions, for testing the hypothesis of normality in stationary process, the `epps.test`, `lobato.test`, `rp.test`, and `varvra.test`. Additionally, we provide functions for unit root, seasonality and ARCH effects tests for stationary, and other additional methods for visual checks using the **ggplot2** and **forecast** packages.

### References


Epps, T.W. (1987). Testing that a stationary time series is Gaussian. *The Annals of Statistic*. 15(4), 1683-1698.`https://projecteuclid.org/euclid.aos/1176350618`.

Lobato, I., & Velasco, C. (2004). A simple test of normality in time series. *Journal of econometric theory*. 20(4), 671-689. `doi:https://doi.org/10.1017/S0266466604204030`.

Psaradakis, Z. & Vavra, M. (2017). A distance test of normality for a wide class of stationary process. *Journal of Econometrics and Statistics*. 2, 50-60. `doi:https://doi.org/10.1016/j.ecosta.2016.11.005`

Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014). A random-projection based test of Gaussianity for stationary processes. *Computational Statistics & Data Analysis, Elsevier*, vol. 75(C), pages 124-141.

Hyndman, R. & Khandakar, Y. (2008). Automatic time series forecasting: the forecast package for `R`. *Journal of Statistical Software*. 26(3), 1-22.`doi: 10.18637/jss.v027.i03`.

Wickham, H. (2008). ggplot2: Elegant Graphics for Data Analysis. *Springer-Verlag New York*.


---

`arch.test` *The ARCH effect test function.*

---

### Description

Performs the Pormanteau Q and Lagrange Multipliers test for homoscedasticity in a univariate stationary process. The null hypothesis (H0), is that the process is homoscedastic.

### Usage

```
arch.test(y,arch="box",alpha=0.05,lag.max = 2)
```

### Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `arch` | A character string naming the desired test for checking stationarity. Valid values are `"box"` for the Ljung-Box, and `"Lm"` for the Lagrange Multiplier test. The default value is `"box"` for the Augmented Ljung-Box test. |
| `alpha` | Level of the test, possible values range from 0.01 to 0.1. By default `alpha = 0.05` is used |
| `lag.max` | an integer with the number of used lags. |

### Details

Several different tests are available: Performs Portmanteau Q and Lagrange Multiplier tests for the null hypothesis that the residuals of an ARIMA model are homoscedastic. The ARCH test is based on the fact that if the residuals (defined as `e(t)`) are heteroscedastic, the squared residuals (e^2[t]) are autocorrelated. The first type of test is to examine whether the squares of residuals are a sequence of white noise, which is called the Portmanteau Q test, and similar to the Ljung-Box test on the squared residuals. By default, `alpha = 0.05` is used to select the more likely hypothesis.

### Value

a h.test class with the main results of unit root hypothesis test.

### Author(s)

Asael Alonzo Matamoros

### References

Engle, R. F. (1982). Auto-regressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica*. 50(4), 987-1007.

McLeod, A. I. & W. K. Li. (1984). Diagnostic Checking ARMA Time Series Models Using Squared-Residual Auto-correlations. *Journal of Time Series Analysis.* 4, 269-273.

### See Also

normal.test,seasonal.test,uroot.test

### Examples

```
#  stationary  ar process
y = arima.sim(100,model = list(ar = 0.3))
arch.test(y)
```

---

## autoplot.ts — *Automatically create a ggplot for time series objects*

---

### Description

autoplot takes an object of type ts or mts and creates a ggplot object suitable for usage with stat_forecast.

### Usage

```
## S3 method for class 'ts'
autoplot(
  object,
  series = NULL,
  xlab = "Time",
  ylab = deparse(substitute(object)),
  main = NULL,
  facets = FALSE,
  colour = TRUE,
  ...
)
```

```
## S3 method for class 'numeric'
autoplot(
  object,
  series = NULL,
  xlab = "Time",
  ylab = deparse(substitute(object)),
  main = NULL,
  ...
)

## S3 method for class 'ts'
fortify(model, data, ...)
```

## Arguments

| | |
|---|---|
| `object` | Object of class "ts" or "mts". |
| `series` | Identifies the time series with a colour, which integrates well with the functionality of geom_forecast. |
| `xlab` | a string with the plot's x axis label. By default a NUll value. |
| `ylab` | a string with the plot's y axis label. By default a counts" value. |
| `main` | a string with the plot's title. |
| `facets` | If TRUE, multiple time series will be faceted (and unless specified, colour is set to FALSE). If FALSE, each series will be assigned a colour. |
| `colour` | If TRUE, the time series will be assigned a colour aesthetic |
| `...` | Other plotting parameters to affect the plot. |
| `model` | Object of class "ts" to be converted to "data.frame". |
| `data` | Not used (required for fortify method) |

## Details

`fortify.ts` takes a `ts` object and converts it into a data frame (for usage with ggplot2).

## Value

None. Function produces a ggplot2 graph.

## Author(s)

Mitchell O'Hara-Wild

## See Also

`plot.ts`, `fortify`

## Examples

```
library(ggplot2)
autoplot(USAccDeaths)

lungDeaths <- cbind(mdeaths, fdeaths)
autoplot(lungDeaths)
autoplot(lungDeaths, facets=TRUE)
```

---

check_plot.ts *Generic function for a visual check of residuals in time series models*

---

## Description

Generic function for a visual check of residuals in time series models, these methods are inspired in the `check.residuals` function provided by the `forecast` package.

## Usage

```
## S3 method for class 'ts'
check_plot(y, model = " ", ...)
```

## Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `model` | A string with the model name. |
| `...` | Other plotting parameters to affect the plot. |

## Value

A graph object from ggplot2

## Author(s)

Asael Alonzo Matamoros

## See Also

`check_residuals`

## Examples

```
y = arima.sim(100,model = list(ar = 0.3))
check_plot(y)
```

check_residuals.ts *Generic functions for checking residuals in time series models*

## Description

Generic function for residuals check analysis, these methods are inspired in the `check.residuals` function provided by the `forecast` package.

## Usage

```
## S3 method for class 'ts'
check_residuals(
  y,
  normality = "epps",
  unit_root = NULL,
  seasonal = NULL,
  arch = NULL,
  alpha = 0.05,
  plot = FALSE,
  ...
)
```

## Arguments

| | |
|---|---|
| y | Either a time series model,the supported classes are `arima0`, `Arima`, `sarima`, `fGarch`, or a time series (assumed to be residuals). |
| normality | A character string naming the desired test for checking gaussian distribution. Valid values are `"epps"` for the Epps, `"lobato"` for Lobato and Velasco's,`"vavras"` for the Psaradakis and Vavra, `"rp"` for the random projections, `"jb"` for the Jarque and Beras, `"ad"` for Anderson Darling test, and `"shapiro"` for the Shapiro-Wilk's test. The default value is `"epps"` test. |
| unit_root | A character string naming the desired unit root test for checking stationarity. Valid values are `"adf"` for the Augmented Dickey-Fuller, `"pp"` for the Phillips-Perron, and `"kpss"` for Kwiatkowski, Phillips, Schmidt, and Shin. The default value is `"adf"` for the Augmented Dickey-Fuller test. |
| seasonal | A character string naming the desired unit root test for checking seasonality. Valid values are `"ocsb"` for the Osborn, Chui, Smith, and Birchenhall, `"ch"` for the Canova and Hansen, and `"hegy"` for Hylleberg, Engle, Granger, and Yoo. The default value is `"ocsb"` for the Osborn, Chui, Smith, and Birchenhall test. |
| arch | A character string naming the desired test for checking stationarity. Valid values are `"box"` for the Ljung-Box, and `"Lm"` for the Lagrange Multiplier test. The default value is `"box"` for the Augmented Ljung-Box test. |
| alpha | Level of the test, possible values range from 0.01 to 0.1. By default `alpha = 0.05` is used |
| plot | A boolean value. If `TRUE`, will produce produces a time plot of the residuals, the corresponding ACF, and a histogram. |

`...` Other testing parameters

### Details

The function performs a residuals analysis, it prints a unit root and seasonal test to check stationarity, and a normality test for checking Gaussian distribution. In addition, if the plot option is `TRUE` a time plot, ACF, and histogram of the series are presented.

### Value

The function does not return any value

### Author(s)

Asael Alonzo Matamoros

### References

Dickey, D. & Fuller, W. (1979). Distribution of the Estimators for Autoregressive Time Series with a Unit Root. *Journal of the American Statistical Association*. 74, 427-431.

Epps, T.W. (1987). Testing that a stationary time series is Gaussian. *The Annals of Statistic*. 15(4), 1683-1698.http://www.jstor.org/stable/2336512. doi:10.1214/aos/1176350618

Osborn, D., Chui, A., Smith, J., & Birchenhall, C. (1988). Seasonality and the order of integration for consumption. *Oxford Bulletin of Economics and Statistics*. 50(4), 361-377.

### Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
check_residuals(y,unit_root = "adf")
```

---

## `epps.statistic` *Estimates the Epps statistic*

---

### Description

Estimates the Epps statistic minimizing the quadratic loss of the process' characteristic function in terms of the first two moments.

### Usage

```
epps.statistic(y)
```

### Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |

### Details

This function is the equivalent of `Sub` in *Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014)*. This function uses a quadratic optimization solver implemented by *Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (2007)*.

### Value

a real value with the Epps test's statistic.

### Author(s)

Alicia Nieto-Reyes and Asael Alonzo Matamoros.

### References

Epps, T.W. (1987). Testing that a stationary time series is Gaussian. *The Annals of Statistic*. 15(4), 1683-1698.

Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014). A random-projection based test of Gaussianity for stationary processes. *Computational Statistics & Data Analysis, Elsevier*, vol. 75(C), pages 124-141.

Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (2007). Numerical Recipes. The Art of Scientific Computing. *Cambridge University Press*.

### See Also

lobato.statistic

### Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
epps.statistic(y)
```

---

## `epps.test` *The Epps and Pulley Test for normality.*

---

### Description

Performs the Epps test for normality. The null hypothesis (H0) is that the given data follows a stationary Gaussian process.

### Usage

```
epps.test(y)
```

### Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |

### Details

The Epps test minimize the process' empirical characteristic function using a quadratic loss in terms of the process two first moments. The test was proposed by *Epps, T.W. (1987)* and implemented by *Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014)* using the `amoebam()` function of *Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (2007)*.

### Value

a h.test class with the main results of the Epps hypothesis test. The h.test class have the following values:

- "epps"The Epps statistic
- "df"The test degrees freedoms
- "p.value"The p value
- "alternative"The alternative hypothesis
- "method"The used method
- "data.name"The data name.

### Author(s)

Asael Alonzo Matamoros and Alicia Nieto-Reyes.

### References

Epps, T.W. (1987). Testing that a stationary time series is Gaussian. *The Annals of Statistic*. 15(4), 1683-1698.

Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014). A random-projection based test of Gaussianity for stationary processes. *Computational Statistics & Data Analysis, Elsevier*, vol. 75(C), pages 124-141.

Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. (2007). Numerical Recipes. The Art of Scientific Computing. *Cambridge University Press*.

### See Also

`lobato.test`

### Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
epps.test(y)
```

ggacf *acf plot*

## Description

Plot of the auto-correlation function for a univariate time series.

## Usage

```
ggacf(y, title = NULL)
```

## Arguments

| | |
|---|---|
| y | a numeric vector or an object of the `ts` class containing a stationary time series. |
| title | a string with the plot's title. |

## Value

None.

## Author(s)

Asael Alonzo Matamoros

## Examples

```
x = rnorm(100)
ggacf(x)
```

gghist *Histogram with optional normal density functions*

## Description

Plots a histogram and density estimates using ggplot.

## Usage

```
gghist(y, title = NULL, xlab = NULL, ylab = "counts", bins, add.normal = TRUE)
```

### Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `title` | a string with the plot's title. |
| `xlab` | a string with the plot's x axis label. By default a NUll value |
| `ylab` | a string with the plot's y axis label. By default a "counts" value |
| `bins` | The number of bins to use for the histogram. Selected by default using the Friedman-Diaconis rule. |
| `add.normal` | A boolean value. Add a normal density function for comparison, by default `add.normal = TRUE`. |

### Value

None.

### Author(s)

Rob J Hyndman

### Examples

```
x = rnorm(100)
gghist(x,add.normal = TRUE)
```

---

## `ggnorm` — `qqplot` *with normal* `qqline`

---

### Description

Plot the quantile-quantile plot and quantile-quantile line using ggplot.

### Usage

```
ggnorm(y, title = NULL, add.normal = TRUE)
```

### Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `title` | a string with the plot's title. |
| `add.normal` | Add a normal density function for comparison. |

### Value

None.

### Author(s)

Asael Alonzo Matamoros

### Examples

```
x = rnorm(100)
ggnorm(x)
```

---

## ggpacf — pacf *plot.*

---

### Description

Plot of the partial autocorrelation function for a univariate time series.

### Usage

```
ggpacf(y, title = NULL)
```

### Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `title` | a string with the plot's title. |

### Value

None.

### Author(s)

Mitchell O'Hara-Wild and Asael Alonzo Matamoros

### Examples

```
x = rnorm(100)
ggpacf(x)
```

---

`Lm.test` *The Lagrange Multiplier test for arch effect.*

---

## Description

Performs the Lagrange Multipliers test for homoscedasticity in a stationary process. The null hypothesis (H0), is that the process is homoscedastic.

## Usage

```
Lm.test(y,lag.max = 2,alpha = 0.05)
```

## Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `lag.max` | an integer with the number of used lags. |
| `alpha` | Level of the test, possible values range from 0.01 to 0.1. By default `alpha = 0.05` is used. |

## Details

The Lagrange Multiplier test proposed by *Engle (1982)* fits a linear regression model for the squared residuals and examines whether the fitted model is significant. So the null hypothesis is that the squared residuals are a sequence of white noise, namely, the residuals are homoscedastic.

## Value

a h.test class with the main results of the Lagrage multiplier hypothesis test. The h.test class have the following values:

- "Lm"The lagrange multiplier statistic
- "df"The test degrees freedoms
- "p.value"The p value
- "alternative"The alternative hypothesis
- "method"The used method
- "data.name"The data name.

## Author(s)

A. Trapletti and Asael Alonzo Matamoros

## References

Engle, R. F. (1982). Auto-regressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica*. 50(4), 987-1007.

McLeod, A. I. and W. K. Li. (1984). Diagnostic Checking ARMA Time Series Models Using Squared-Residual Auto-correlations. *Journal of Time Series Analysis.* 4, 269-273.

### See Also

arch.test

### Examples

```
# generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
Lm.test(y)
```

---

## lobato.statistic *Computes the Lobato and Velasco statistic*

---

### Description

Computes the Lobato and Velasco's statistic. This test proves a normality assumption in correlated data employing the skewness-kurtosis test statistic, but studentized by standard error estimates that are consistent under serial dependence of the observations.

### Usage

```
lobato.statistic(y,c = 1)
```

### Arguments

| | |
|---|---|
| y | a numeric vector or an object of the `ts` class containing a stationary time series. |
| c | a positive real value that identifies the total amount of values used in the cumulative sum. |

### Details

This function is the equivalent of `GestadisticoVn` of *Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014)*.

### Value

A real value with the Lobato and Velasco test's statistic.

### Author(s)

Alicia Nieto-Reyes and Asael Alonzo Matamoros.

### References

Lobato, I., & Velasco, C. (2004). A simple test of normality in time series. *Journal of econometric theory*. 20(4), 671-689.

Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014). A random-projection based test of Gaussianity for stationary processes. *Computational Statistics & Data Analysis, Elsevier*, vol. 75(C), pages 124-141.

## See Also

`epps.statistic`

## Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
lobato.statistic(y,3)
```

---

`lobato.test` *The Lobato and Velasco's Test for normality*

---

## Description

Performs the Lobato and Velasco's test for normality. The null hypothesis (H0), is that the given data follows a Gaussian process.

## Usage

```
lobato.test(y,c = 1)
```

## Arguments

| | |
|---|---|
| y | a numeric vector or an object of the `ts` class containing a stationary time series. |
| c | a positive real value that identifies the total amount of values used in the cumulative sum. |

## Details

This test proves a normality assumption in correlated data employing the skewness-kurtosis test statistic, but studentized by standard error estimates that are consistent under serial dependence of the observations. The test was proposed by *Lobato, I., & Velasco, C. (2004)* and implemented by *Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014)*.

## Value

A h.test class with the main results of the Lobato and Velasco's hypothesis test. The h.test class have the following values:

- "lobato"The Lobato and Velasco's statistic
- "df"The test degrees freedoms
- "p.value"The p value
- "alternative"The alternative hypothesis
- "method"The used method
- "data.name"The data name.

### Author(s)

Asael Alonzo Matamoros and Alicia Nieto-Reyes.

### References

Lobato, I., & Velasco, C. (2004). A simple test of normality in time series. *Journal of econometric theory*. 20(4), 671-689.

Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014). A random-projection based test of Gaussianity for stationary processes. *Computational Statistics & Data Analysis, Elsevier*, vol. 75(C), pages 124-141.

### See Also

lobato.statistic,epps.test

### Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
lobato.test(y)
```

---

## normal.test — *The normality test for stationary process*

---

### Description

Perform a normality test. The null hypothesis (H0) is that the given data follows a stationary Gaussian process.

### Usage

```
normal.test(y,normality="epps",alpha=0.05)
```

### Arguments

| | |
|---|---|
| y | a numeric vector or an object of the ts class containing a stationary time series. |
| normality | A character string naming the desired test for checking normality. Valid values are "epps" for the Epps, "lobato" for Lobato and Velasco's,"vavra" for the Psaradakis and Vavra, "rp" for the random projections, "jb" for the Jarque and Beras, "ad" for Anderson Darling test, and "shapiro" for the Shapiro-Wilk's test. The default value is "epps" test. |
| alpha | Level of the test, possible values range from 0.01 to 0.1. By default alpha = 0.05 is used. |

### Details

Several different tests are available: `"lobato"`, `"epps"`, `"vavras"` and `"rp"` test are for testing normality in stationary process. `"jb"`, `"ad"`, and `"shapiro"` tests are for numeric data. In all cases, the alternative hypothesis is that `y` follows a Gaussian process. By default, `alpha = 0.05` is used to select the more likely hypothesis.

### Value

An h.test class with the main results of normal hypothesis test.

### Author(s)

Asael Alonzo Matamoros

### References


Epps, T.W. (1987). Testing that a stationary time series is Gaussian. *The Annals of Statistic*. 15(4), 1683-1698.

Lobato, I., & Velasco, C. (2004). A simple test of normality in time series. *Journal of econometric theory*. 20(4), 671-689.

Psaradakis, Z. & Vavra, M. (2017). A distance test of normality for a wide class of stationary process. *Journal of Econometrics and Statistics*. 2, 50-60.

Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014). A random-projection based test of Gaussianity for stationary processes. *Computational Statistics & Data Analysis, Elsevier*, vol. 75(C), pages 124-141.

Patrick, R. (1982). An extension of Shapiro and Wilk's W test for normality to large samples. *Journal of Applied Statistics*. 31, 115-124.

Cromwell, J. B., Labys, W. C. & Terraza, M. (1994). Univariate Tests for Time Series Models. *Sage, Thousand Oaks, CA*. 20-22.


### See Also

`uroot.test`,`seasonal.test`

### Examples

```
#  stationary  ar process
y = arima.sim(100,model = list(ar = 0.3))
normal.test(y) # epps test

# normal random sample
y = rnorm(100)
normal.test(y,normality = "shapiro")

# exponential random sample
y = rexp(100)
normal.test(y,normality = "ad")
```

`random.projection` *Generate a random projection*

## Description

Generates a random projection of a univariate stationary stochastic process. Using a beta(shape1,shape2) distribution.

## Usage

```
random.projection(y,shape1,shape2,seed = NULL)
```

## Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `shape1` | an optional real value with the first shape parameters of the beta distribution. |
| `shape2` | an optional real value with the second shape parameters of the beta distribution. |
| `seed` | An optional `seed` to use. |

## Details

Generates one random projection of a stochastic process using a beta distribution. For more details, see: *Nieto-Reyes, A.,Cuesta-Albertos, J. & Gamboa, F. (2014)*.

## Value

a real vector with the projected stochastic process.

## Author(s)

Alicia Nieto-Reyes and Asael Alonzo Matamoros

## References

Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014). A random-projection based test of Gaussianity for stationary processes. *Computational Statistics & Data Analysis, Elsevier*, vol. 75(C), pages 124-141.Result

Epps, T.W. (1987). Testing that a stationary time series is Gaussian. *The Annals of Statistic*. 15(4), 1683-1698.

Lobato, I., & Velasco, C. (2004). A simple test of normality in time series. *Journal of econometric theory*. 20(4), 671-689.

## See Also

`lobato.test` `epps.test`

## Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
rp.test(y)
```

---

`rp.sample` *Generates a test statistics sample of random projections*

---

## Description

Generates a sample of test statistics using k independent random projections of a stationary process. The first half values of the sample, are estimated using a Lobato and Velasco's statistic test. The last half values with an Epps statistic test.

## Usage

```
rp.sample(y,k = 2,pars1 = c(100,1),pars2 = c(2,7),seed = NULL)
```

## Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `k` | an integer with the number of random projections to be used, by default `k = 2`. |
| `pars1` | an optional real vector with the shape parameters of the beta distribution used for the odd number random projection. By default, `pars1 = c(100,1)` where, `shape1 = 100` and `shape2 = 1`. |
| `pars2` | an optional real vector with the shape parameters of the beta distribution used for the even number random projection. By default, `pars2 = c(2,7)` where, `shape1 = 2` and `shape2 = 7`. |
| `seed` | An optional `seed` to use. |

## Details

The `rp.sample` function generates k independent random projections of the process. A Lobatos and Velasco's test is applied to the first half of the projections. And an Epps test for the other half.

For generating the k random projections a beta distribution is used. By default a `beta(shape1 = 100,shape = 1)` and a `beta(shape1 = 2,shape = 7)` are used to generate the odd and even projections respectively. For using a different parameter set, change `pars1` or `pars2` values.

The test was proposed by *Nieto-Reyes, A.,Cuesta-Albertos, J. & Gamboa, F. (2014)*.

## Value

A list with 2 real value vectors:

- "lobato"A vector with the Lobato and Velasco's statistics sample
- "epps"A vector with the Epps statistics sample.

### Author(s)

Alicia Nieto-Reyes and Asael Alonzo Matamoros

### References

Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014). A random-projection based test of Gaussianity for stationary processes. *Computational Statistics & Data Analysis, Elsevier*, vol. 75(C), pages 124-141.

Epps, T.W. (1987). Testing that a stationary time series is Gaussian. *The Annals of Statistic*. 15(4), 1683-1698.

Lobato, I., & Velasco, C. (2004). A simple test of normality in time series. *Journal of econometric theory*. 20(4), 671-689.

### See Also

`lobato.test` `epps.test`

### Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
rp.test(y)
```

---

`rp.test` *The k random projections test for normality*

---

### Description

Performs the random projection test for normality. The null hypothesis (H0) is that the given data follows a stationary Gaussian process, and k is the number of used random projections.

### Usage

```
rp.test(y,k = 64,FDR = FALSE,pars1 = c(100,1),pars2  = c(2,7),seed = NULL)
```

### Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `k` | an integer with the number of random projections to be used, by default `k = 2`. |
| `FDR` | a logical value for mixing the p-values using a dependent False discovery rate method. By default `FDR = FALSE`. |
| `pars1` | an optional real vector with the shape parameters of the beta distribution used for the odd number random projection. By default, `pars1 = c(100,1)` where, `shape1 = 100` and `shape2 = 1`. |

pars2 an optional real vector with the shape parameters of the beta distribution used for the even number random projection. By default, pars2 = c(2,7) where, shape1 = 2 and shape2 = 7.

seed An optional seed to use.

### Details

The random projection test generates k independent random projections of the process. A Lobato and Velasco's test are applied to the first half of the projections, and an Epps test for the other half. By default, a Monte Carlo p-value estimate is used for mixing the tests. A False discovery rate can be used for mixing by setting FDR = TRUE.

For generating the k random projections a beta distribution is used. By default a beta(shape1 = 100,shape = 1) and a beta(shape1 = 2,shape = 7) are used to generate the odd and even projections respectively. For using a different parameter set, change pars1 or pars2.

The test was proposed by *Nieto-Reyes, A.,Cuesta-Albertos, J. & Gamboa, F. (2014)*.

### Value

a h.test class with the main results of the Epps hypothesis test. The h.test class have the following values:

- "k"The number of used projections
- "lobato"The average Lobato and Velasco's test statistics of the k projected samples
- "epps"The average Epps test statistics of the k projected samples
- "p.value"The mixed p value
- "alternative"The alternative hypothesis
- "method"The used method: rp.test
- "data.name"The data name.

### Author(s)

Asael Alonzo Matamoros and Alicia Nieto-Reyes.

### References

Nieto-Reyes, A., Cuesta-Albertos, J. & Gamboa, F. (2014). A random-projection based test of Gaussianity for stationary processes. *Computational Statistics & Data Analysis, Elsevier*, vol. 75(C), pages 124-141.

Epps, T.W. (1987). Testing that a stationary time series is Gaussian. *The Annals of Statistic*. 15(4), 1683-1698.

Lobato, I., & Velasco, C. (2004). A simple test of normality in time series. *Journal of econometric theory*. 20(4), 671-689.

### See Also

lobato.test epps.test

## Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
rp.test(y)
```

---

`seasonal.test` *The Seasonal unit root tests function*

---

## Description

Perform a seasonal unit root test to check seasonality in a linear stochastic process

## Usage

```
seasonal.test(y,seasonal="ocsb",alpha=0.05)
```

## Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `seasonal` | A character string naming the desired seasonal unit root test for checking seasonality. Valid values are `"ocsb"` for the Osborn, Chui, Smith, and Birchenhall, `"ch"` for the Canova and Hansen, and `"hegy"` for Hylleberg, Engle, Granger, and Yoo. The default value is `"ocsb"` for the Osborn, Chui, Smith, and Birchenhall test. |
| `alpha` | Level of the test, possible values range from 0.01 to 0.1. By default `alpha = 0.05` is used |

## Details

Several different tests are available: In the `kpss` test, the null hypothesis that `y` has a stationary root against a unit-root alternative. In the two remaining tests, the null hypothesis is that `y` has a unit root against a stationary root alternative. By default, `alpha = 0.05` is used to select the more likely hypothesis.

## Value

a h.test class with the main results of unit root hypothesis test.

## Author(s)

Asael Alonzo Matamoros

### References

Osborn, D., Chui, A., Smith, J., & Birchenhall, C. (1988). Seasonality and the order of integration for consumption. *Oxford Bulletin of Economics and Statistics*. 50(4), 361-377.

Canova, F. & Hansen, B. (1995). Are Seasonal Patterns Constant over Time? A Test for Seasonal Stability. *Journal of Business and Economic Statistics*. 13(3), 237-252.

Hylleberg, S., Engle, R., Granger, C. & Yoo, B. (1990). Seasonal integration and cointegration. *Journal of Econometrics* 44(1), 215-238.

### See Also

`normal.test`,`uroot.test`

### Examples

```
#  stationary  ar process
y = ts(rnorm(100),frequency = 6)
seasonal.test(y)
```

---

## `sieve.bootstrap` *Generates a sieve bootstrap sample*

---

### Description

The function generates a sieve bootstrap sample for a univariate stochastic process.

### Usage

```
sieve.bootstrap(y,reps = 1000,pmax = NULL,h = 100,seed = NULL)
```

### Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `reps` | an integer with the total bootstrap repetitions. |
| `pmax` | an integer with the max considered lags for the generated `ar(p)` process. By default, `pmax = NULL`. |
| `h` | an integer with the first `burn-in` sieve bootstrap replicates. |
| `seed` | An optional `seed` to use. |

### Details

simulates bootstrap samples for the stochastic process y, using a stationary auto-regressive model of order `"pmax"`, `AR(pmax)`. If `pmax = NULL` (*default*), the function estimates the process maximum lags using an `AIC` as a model selection criteria.

## Value

A matrix or `reps` row and `n` columns, with the sieve bootstrap sample and `n` the time series length.

## Author(s)

Asael Alonzo Matamoros

## References

Bulmann, P. (1997). Sieve Bootstrap for time series. *Bernoulli*. 3(2), 123 -148.

## See Also

`lobato.test`,`epps.test`

## Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
M = sieve.bootstrap(y)
```

---

`uroot.test` *The Unit root tests function*

---

## Description

Perform a unit root test to check stationarity in a linear stochastic process.

## Usage

```
uroot.test(y,unit_root="adf",alpha=0.05)
```

## Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `unit_root` | A character string naming the desired unit root test for checking stationarity. Valid values are `"adf"` for the Augmented Dickey-Fuller, `"pp"` for the Phillips-Perron, `"kpss"` for Kwiatkowski, Phillips, Schmidt, and Shin, and `"box"` for the Ljung-Box. The default value is `"adf"` for the Augmented Dickey-Fuller test. |
| `alpha` | Level of the test, possible values range from 0.01 to 0.1. By default `alpha = 0.05` is used |

## Details

Several different tests are available: In the `kpss` test, the null hypothesis that `y` has a stationary root against a unit-root alternative. In the two remaining tests, the null hypothesis is that `y` has a unit root against a stationary root alternative. By default, `alpha = 0.05` is used to select the more likely hypothesis.

### Value

a h.test class with the main results of unit root hypothesis test.

### Author(s)

Asael Alonzo Matamoros and A. Trapletti

### References

Dickey, D. & Fuller, W. (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., Schmidt, P. & Shin, Y. (1992). Testing the Null Hypothesis of Stationarity against the Alternative of a Unit Root, *Journal of Econometrics*. 54, 159-178.

Phillips, P. & Perron, P. (1988). Testing for a unit root in time series regression, *Biometrika*. 72(2), 335-346.

Ljung, G. M. & Box, G. E. P. (1978). On a measure of lack of fit in time series models. *Biometrika*. 65, 297-303.

### See Also

normal.test,seasonal.test

### Examples

```
#  stationary  ar process
y = arima.sim(100,model = list(ar = 0.3))
uroot.test(y)

# a random walk process
y = cumsum(y)
uroot.test(y,unit_root = "pp")
```

---

## vavra.sample *vavra test's sieve Bootstrap sample for Anderson Darling statistic*

---

### Description

Generates a sieve bootstrap sample of the Anderson-Darling statistic test.

### Usage

```
vavra.sample(y,reps = 1000,h = 100,seed = NULL)
```

## Arguments

- `y` a numeric vector or an object of the `ts` class containing a stationary time series.
- `reps` an integer with the total bootstrap repetitions.
- `h` an integer with the first `burn-in` sieve bootstrap replicates.
- `seed` An optional `seed` to use.

## Details

The Vavra test approximates the empirical distribution function of the Anderson-Darlings statistic, using a sieve bootstrap approximation. The test was proposed by *Psaradakis, Z. & Vavra, M (20.17)*.

This function is the equivalent of `xarsieve` of *Psaradakis, Z. & Vavra, M (20.17)*.

## Value

A numeric array with the Anderson Darling sieve bootstrap sample

## Author(s)

Asael Alonzo Matamoros.

## References

Psaradakis, Z. & Vavra, M. (2017). A distance test of normality for a wide class of stationary process. *Journal of Econometrics and Statistics*. 2, 50-60.

Bulmann, P. (1997). Sieve Bootstrap for time series. *Bernoulli*. 3(2), 123 -148.

## See Also

`epps.statistic` `lobato.statistic`

## Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
adbs = vavra.sample(y)
mean(adbs)
```

`vavra.test` *The Psaradakis and Vavra test for normality*

## Description

Performs the Psaradakis and Vavra distance test for normality. The null hypothesis (H0), is that the given data follows a Gaussian process.

## Usage

```
vavra.test(y,reps = 1000,h = 100,seed = NULL)
```

## Arguments

| | |
|---|---|
| `y` | a numeric vector or an object of the `ts` class containing a stationary time series. |
| `reps` | an integer with the total bootstrap repetitions. |
| `h` | an integer with the first `burn-in` sieve bootstrap replicates. |
| `seed` | An optional `seed` to use. |

## Details

The Psaradakis and Vavra test approximates the empirical distribution function of the Anderson Darling's statistic, using a sieve bootstrap approximation. The test was proposed by *Psaradakis, Z. & Vavra, M (20.17).*

## Value

a h.test class with the main results of the Epps hypothesis test. The h.test class have the following values:

- "bootstrap A"The sieve bootstrap A statistic
- "p.value"The p value
- "alternative"The alternative hypothesis
- "method"The used method
- "data.name"The data name.

## Author(s)

Asael Alonzo Matamoros.

## References

Psaradakis, Z. & Vavra, M. (2017). A distance test of normality for a wide class of stationary process. *Journal of Econometrics and Statistics*. 2, 50-60.

Bulmann, P. (1997). Sieve Bootstrap for time series. *Bernoulli*. 3(2), 123 -148.

## See Also

lobato.test,epps.test

## Examples

```
# Generating an stationary arma process
y = arima.sim(100,model = list(ar = 0.3))
vavra.test(y)
```

# Index