Package 'NlinTS'

March 16, 2018

Type Package

Title Non Linear Time Series Analysis

Version 1.3.2

Date 2018-03-15

Maintainer Youssef Hmamouche <hmamoucheyussef@gmail.com>

Description Models for time series forecasting and causality detection. The main functionalities of this package consist of a neural network Vector Auto-Regressive model, the classical Granger causality test C.W.J.Granger (1980) <doi:10.1016/0165-1889(80)90069-X>, and a non-linear version of it based on feedforward neural networks.

License GNU General Public License

Depends Rcpp

Imports methods, timeSeries, Rdpack

RdMacros Rdpack

LinkingTo Rcpp

SystemRequirements C++11

NeedsCompilation yes

RoxygenNote 6.0.1

Author Youssef Hmamouche [aut, cre],
Sylvain Barthelemy [cph]

Repository CRAN

Date/Publication 2018-03-16 12:59:55 UTC

R topics documented:

causality.test *The Granger causality test*

Description

The Granger causality test

Usage

```
causality.test(ts1, ts2, lag, diff = FALSE)
```

Arguments

ts1	Numerical dataframe containing one variable
ts2	Numerical dataframe containing one variable
lag	The lag parameter
diff	Logical argument for the option of making data stationary

Details

The test evaluates if the second time series causes the first one using the Granger test of causality.

Value

summary (): shows the test results

F-test (): returns the value of the test

References

Granger CWJ (1980). "Testing for Causality." *Journal of Economic Dynamics and Control*, **2**, pp. 329–352. ISSN 0165-1889, doi: 10.1016/01651889(80)90069X.

Examples

```
library (timeSeries) # to extract time series
library (NlinTS)
data = LPP2005REC
model = causality.test (data[,1], data[,2], 2)
model$summary ()
```

`df.test` *Augmented Dickey_Fuller test*

Description

Augmented Dickey_Fuller test

Usage

```
df.test(ts, lag)
```

Arguments

`ts`	Numerical dataframe
`lag`	The lag parameter

Details

Computes the stationarity test for a given univariate time series.

Value

summary (): shows the test results

df (): returns the value of the test

References

Elliott G, Rothenberg TJ and Stock JH (1992). "Efficient tests for an autoregressive unit root."

Examples

```
library (timeSeries)
library (NlinTS)
#load data
data = LPP2005REC
model = df.test (data[,1], 1)
model$summary ()
```

`nlin_causality.test` *A non linear Granger causality test*

Description

A non linear Granger causality test

Usage

```
nlin_causality.test(ts1, ts2, lag, LayersUniv, LayersBiv, iters, bias = TRUE)
```

Arguments

`ts1`	Numerical series
`ts2`	Numerical series
`lag`	The lag parameter
`LayersUniv`	Integer vector of the size of hidden layers of the univariate model
`LayersBiv`	Integer vector of the size of hidden layers of the bivariate model
`iters`	The number of iterations
`bias`	Logical argument for the option of using the bias in the networks

Details

The test evaluates if the second time series causes the first one. Two MLP artificial neural networks are evaluated to perform the test, one using just the target time series (ts1), and the second using both time series.

Value

pvalue: the p-value of the test

Ftest: the statistic of the test

summary (): shows the test results

F-test (): returns the value of the test

Examples

```
library (timeSeries) # to extract time series
library (NlinTS)
data = LPP2005REC
model = nlin_causality.test (data[,1], data[,2], 2, c(2), c(4), 500, TRUE)
model$summary ()
```

varmlp *Artificial Neural Network VAR (Vector Auto-Regressive) model using a MultiLayer Perceptron.*

Description

Artificial Neural Network VAR (Vector Auto-Regressive) model using a MultiLayer Perceptron.

Usage

```
varmlp(df, lag, sizeOfHLayers, iters, bias = TRUE)
```

Arguments

df	A numerical dataframe
lag	The lag parameter
sizeOfHLayers	Integer vector that contains the size of hidden layers (the number of hidden layers is the size of this vector)
iters	The number of iterations
bias	Logical, true if the bias have to be used in the network

Details

This function constructs the model.

Value

train (df): updates the model using the input dataframe df

forecast (df): returns the next row forecasts of an given dataframe df

Examples

```
library (timeSeries) # to extract time series
library (NlinTS)
#load data
data = LPP2005REC
# Prepare data to make one forecasts
train_data = head (data, nrow (data) - 1)
test_data = tail (data, 1)
model = varmlp (train_data, 1, c(10,5), 200, TRUE)
predictions = model$forecast (train_data)
print (tail (predictions,1))
# Update the model (learning from new data)
model$train (test_data)
```

Index