Package: GMDH (via r-universe)

August 25, 2024

Type Package

Title Short Term Forecasting via GMDH-Type Neural Network Algorithms

Version 1.6

Date 2016-09-20

Depends R (>= 3.2.5)

Imports MASS, stats, utils

Author Osman Dag, Ceylan Yozgatligil

Maintainer Osman Dag <osman.dag@hacettepe.edu.tr>

Description Group method of data handling (GMDH) - type neural network algorithm is the heuristic self-organization method for modelling the complex systems. In this package, GMDH-type neural network algorithms are applied to make short term forecasting for a univariate time series.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2016-09-20 15:38:59

Contents

GMDH-package *Short Term Forecasting via GMDH-Type Neural Network Algorithms*

Description

Package GMDH includes a function for short term forecasting of a univariate time series by using GMDH-type neural network algorithms, and a dataset for implementation.

Details

Package:	GMDH
Type:	Package
Version:	1.6
Date:	2016-09-20
License:	GPL (>=2)

cancer *Cancer Data*

Description

Yearly cancer death rate (per 100,000 population) of Pennsylvania between 1930 and 2000.

Usage

```
data(cancer)
```

Format

A time series with 71 observations on the following variable.

`cancer` a time series for yearly cancer death rate

References

Wei, W. W.S. (2006). *Time Series Analysis: Univariate and Multivariate Methods (2nd ed.)* Boston: Addison-Wesley.

Examples

```
data(cancer)
plot(cancer)
out = fcast(cancer, f.number = 2)
out$mean
```

`fcast` *A Function to Make Short Term Forecasting via GMDH-Type Neural Network Algorithms*

Description

`fcast` forecasts time series via GMDH-type neural network algorithms.

Usage

```
fcast(data, method = "GMDH", input = 4, layer = 3, f.number = 5, level = 95,
tf = "all", weight = 0.7,lambda = c(0,0.01,0.02,0.04,0.08,0.16,0.32,0.64,
1.28,2.56,5.12,10.24))
```

Arguments

`data`	is a univariate time series of class ts
`method`	expects a character string to choose the desired method to forecast time series. To utilize GMDH-type neural network in forecasting, method is set to "GMDH". One should set method to "RGMDH" for forecasting via Revised GMDH-type neural network. Default is set to "GMDH"
`input`	is the number of inputs. Defaults input = 4
`layer`	is the number of layers. Default is set to layer = 3
`f.number`	is the number of observations to be forecasted. Defaults f.number = 5
`level`	confidence level for prediction interval. Default is set to 95
`tf`	expects a character string to choose the desired transfer function to be used in forecasting. To use polynomial function, tf should be set to "polynomial". Similarly, tf should be set to "sigmoid", "RBF", "tangent" to utilize sigmoid function, radial basis function and tangent function, respectively. To use all functions simultaneously, default is set to "all"
`weight`	is the percent of the data set to be utilized as learning set to estimate regularization parameter via validation. Default is set to weight = 0.70
`lambda`	is a vector which includes the sequence of feasible regularization parameters. Defaults lambda=c(0,0.01,0.02,0.04,0.08,0.16,0.32,0.64,1.28,2.56,5.12,10.24)

Value

Returns a list containing following elements:

`method`	the forecasting method as a character string
`mean`	point forecasts as a time series
`lower`	lower limits for prediction interval
`upper`	upper limits for prediction interval
`level`	confidence level for prediction interval

x	the original time series
fitted	the fitted values
residuals	the residuals of the model. The residuals are x minus the fitted values

Author(s)

Osman Dag, Ceylan Yozgatligil

References

Dag, O., Yozgatligil, C. (2016). GMDH: An R Package for Short Term Forecasting via GMDH-Type Neural Network Algorithms. *The R Journal*, **8:1**, 379-386.

Ivakhnenko, A. G. (1966). Group Method of Data Handling - A Rival of the Method of Stochastic Approximation. *Soviet Automatic Control*, **13**, 43-71.

Kondo, T., Ueno, J. (2006). Revised GMDH-Type Neural Network Algorithm With A Feedback Loop Identifying Sigmoid Function Neural Network. *International Journal of Innovative Computing, Information and Control*, **2:5**, 985-996.

Examples

```
data = ts(rnorm(100, 10, 1))
out = fcast(data)
out

data = ts(rnorm(100, 10, 1))
out = fcast(data, input = 6, layer = 2, f.number = 1)
out$mean
out$fitted
out$residuals
plot(out$residuals)
hist(out$residuals)
```

Index