

Package ‘EXPARMA’

July 19, 2023

Type Package

Title Fitting of Exponential Autoregressive Moving Average (EXPARMA) Model

Version 0.1.0

Author Bishal Gurung [aut, cre],
Saikat Das [aut],
Achal Lama [aut],
Kn Singh [aut]

Maintainer Bishal Gurung <Bishal.Gurung@icar.gov.in>

Description The amplitude-dependent autoregressive time series model (EXPAR) proposed by Haggan and Ozaki (1981) <[doi:10.2307/2335819](https://doi.org/10.2307/2335819)> was improved by incorporating the moving average (MA) framework for capturing the variability efficiently. Parameters of the EXPARMA model can be estimated using this package. The user is provided with the best fitted EXPARMA model for the data set under consideration.

Encoding UTF-8

License GPL-3

Imports forecast, stats

NeedsCompilation no

Repository CRAN

Date/Publication 2023-07-19 10:50:11 UTC

R topics documented:

BestExp	2
EXPARMAfit	3
EXPARMA_optim	4
init_val	5
Index	6

BestExp	<i>Fitting of Best Exponential Autoregressive Moving Average (EX-PARMA) Model</i>
---------	---

Description

This function will fit best EXPARMA model.

Usage

```
BestExp(ts_data, opt_method, max.p, max.q)
```

Arguments

ts_data	a univariate time series data
opt_method	opt_method represents the optimization methods that can be used. All the methods available at optim() function can be used here. By default the method used is "BFGS".
max.p	max.p is the order of autoregressive parameter of the model. By default the value is 5.
max.q	max.q is the order of moving average parameter of the model. By default the value is 5.

Details

Takes only one input, the data. Checks the AIC values of all EXPARMA models from order (1,1) up to (max.p, max.q) by finding the optimal parameters of those models and returns a fit with the model having the least AIC. The maximum order up to which checks are set to 5 by default.

Value

It returns the best fitted EXPARMA model.

Author(s)

Saikat Das, Bishal Gurung, Achal Lama and Kn Singh

References

Haggan and Ozaki (1981). Modelling nonlinear random vibrations using an amplitude-dependent autoregressive time series model. *Biometrika*,68(1):189-199.<doi:10.1093/biomet/68.1.189>. Gurung (2015). An exponential autoregressive (EXPAR) (EXPAR) model for the forecasting of all India annual rainfall. *Mausam*,66(4):847-849. <doi:10.54302/mausam.v66i4.594>.

Examples

```

datats=c(17597,14074,11425,11691,11298,12351,14311,
12349,10537,11755,13154,11989,13022,12107,11172,10667,
10091,12204,12274,22343)
BestExp(datats,opt_method="BFGS",max.p=1, max.q=1)

```

EXPARMAfit	<i>Fitting of Exponential Autoregressive Moving Average (EXPARMA) Model</i>
------------	---

Description

This function will fit EXPARMA model for given parameters.

Usage

```
EXPARMAfit(ts_data, order, par)
```

Arguments

ts_data	a univariate time series data
order	order represents the values of autoregressive (p) and moving average (q) parameters of EXPARMA model. p and q will take integer values of 1 or more. By default these values are set as 1.
par	par is the parametric space of the EXPARMA model. The total number of parameters are $2*(p+q+1)$.

Details

This function takes 3 inputs, all compulsory, i.e., data, order and parameters of the model to be fitted and returns the fitted values, residuals, RSS and AIC for the fitted model. No optimisation is done. This function is useful for simulation of data with given order and parameters.

Value

It returns the fitted EXPARMA model.

Examples

```

datats=c(17597,14074,11425,11691,11298,12351,14311,
12349,10537,11755,13154,11989,13022,12107,11172,10667,
10091,12204,12274,22343)
EXPARMAfit(datats,order=c(1,1), par=c(1,0.5,0.1,0.5,0.5,0.5))

```

EXPARMA_optim	<i>Fitting of Exponential Autoregressive Moving Average (EXPARMA) Model</i>
---------------	---

Description

This function will fit EXPARMA model for given parameters.

Usage

```
EXPARMA_optim(ts_data, order, init, opt_method)
```

Arguments

ts_data	a univariate time series data
order	order represents the values of autoregressive (p) and moving average (q) parameters of EXPARMA model. p and q will take integer values of 1 or more. By default these values are set as 1.
init	init represents the initial values of the model EXPARMA. These values can be user specified or the output from init_val() can be used.
opt_method	opt_method represents the optimization method used for estimating the model parameters. All methods available under optim() can be used here. By default we have made use of "BFGS".

Details

Takes 2 compulsory inputs, i.e., the data and order of model to be fitted. Optimizes the parameters of the model of the given order (p,q) by minimizing. RSS and returns a fit with the optimized parameters. The fit is returned using the previous function EXPARMAfit(), with the parameters inputted being the optimized ones. All other inputs to this function are arguments for the optim() part.

Value

It returns the model parameters, RSS, AIC, fitted values and residuals.

Examples

```
datats=c(17597,14074,11425,11691,11298,12351,14311,
12349,10537,11755,13154,11989,13022,12107,11172,10667,
10091,12204,12274,22343)
EXPARMA_optim(datats,order=c(1,1))
```

init_val	<i>Fitting of Exponential Autoregressive Moving Average (EXPARMA) Model</i>
----------	---

Description

This function will fit EXPARMA model for given parameters.

Usage

```
init_val(ts_data, order)
```

Arguments

ts_data	a univariate time series data
order	order represents the values of autoregressive (p) and moving average (q) parameters of EXPARMA model. p and q will take integer values of 1 or more. By default these values are set as 1.

Details

This function takes two inputs, i.e., the data and order of the model to be fitted and returns the initial values over which optimization is to be done for fitting the model. The initial values are taken from an ARIMA model with the same order and appropriate differencing. These will act as default initial values in the EXPARMA function, unless specified by user. Without these values, the optimization process may fail.

Value

It returns the initial values of the EXAPRMA model over which optimization is to be done for fitting the model.

Examples

```
datats=c(17597,14074,11425,11691,11298,12351,14311,  
12349,10537,11755,13154,11989,13022,12107,11172,10667,  
10091,12204,12274,22343)  
init_val(datats,order=c(1,1))
```

Index

BestExp, [2](#)

EXPARMA_optim, [4](#)

EXPARMAfit, [3](#)

init_val, [5](#)