Package ‘AnalyzeTS’

November 24, 2016

Type Package

Title Analyze Fuzzy Time Series

Version 2.2

Date 2016-11-21

Author
Tran Thi Ngoc Han, Doan Hai Nghi, Mai Thi Hong Diem, Nguyen Thi Diem My, Hong Viet Minh, Vo Van Tai, Pham Minh T

Maintainer Hong Viet Minh <hongvietminh@gmail.com>

Description Analyze fuzzy time series by Chen, Singh, Heuristic and Chen-Hsu models. The Abbasov-Mamedova and NFTS models is included as well.

License GPL (>= 2.0)

LazyLoad yes

Depends MASS,TSA,TTR,tseries, urca

NeedsCompilation no

Repository CRAN

Date/Publication 2016-11-24 11:31:00

R topics documented:

`AnalyzeTS-package` *Analyze Fuzzy Time Series*

Description

Analyze fuzzy time series by Chen, Singh, Heuristic and Chen-Hsu models. The Abbasov-Mamedova and NFTS models is included as well.

Author(s)

Tran Thi Ngoc Han, Doan Hai Nghi, Mai Thi Hong Diem, Nguyen Thi Diem My, Hong Viet Minh, Vo Van Tai, Pham Minh Truc.

Maintainer: Hong Viet Minh <hongvietminh@gmail.com>

References

Chen, S.M., 1996. Forecasting enrollments based on fuzzy time series. Fuzzy Sets and Systems. 81: 311-319.

Chen, S.M. and Hsu, C.C., 2004. A New method to forecast enrollments using fuzzy time series. International Journal of Applied Science and Engineering, 12: 234-244.

Huarng, H., 2001. Huarng models of fuzzy time series for forecasting. Fuzzy Sets and Systems. 123: 369-386.

Singh, S.R., 2008. A computational method of forecasting based on fuzzy time series. Mathematics and Computers in Simulation. 79: 539-554

Abbasov, A.M. and Mamedova, M.H., 2003. Application of fuzzy time series to population forecasting, Proceedings of 8th Symposion on Information Technology in Urban and Spatial Planning, Vienna University of Technology, February 25-March1, 545-552.

Examples

```
library(AnalyzeTS)
data(enrollment)
#Sing model
fuzzy.ts1(lh,n=5,type="Singh",plot=TRUE)

#Abbasov Mamedova model
fuzzy.ts2(enrollment,n=5,w=5,C=0.01,forecast=5,plot=TRUE,type="Abbasov-Mamedova")

#NFTS model
fuzzy.ts2(enrollment,n=5,w=5,C=0.01,forecast=5,plot=TRUE,type="NFTS")
```

---

av.res *The criterion to evaluate forecasting model*

Description

Calculating to return answer which are 7 criterion to evaluate forecasting models, which are ME (Mean error), MAE (Mean absolute error), MPE (Mean percentage error), MAPE (Mean absolute percentage error), MSE (Mean squared error), RMSE (Root of mean square error), and U (Theil's U statistic).

Usage

```
av.res(Y = NULL, F = NULL, E = NULL, r = 3)
```

Arguments

`Y`	A data frame of univariate time series.
`F`	A data frame of interpolat time series.
`E`	A data frame of residual time series.
`r`	Display results returned to the specified number of decimal places (default 3). (See `round2str` for details of r paramicter.)

Details

The Yt is 'observation series'. The Ft is 'Forecasting series'. The et is 'residual series'. The n is size of sample. The accuracies are calculated by theory:

ME = sum(et)/n

MAE = sum(|et|)/n

MPE = sum((et/Yt)*100)/n

MAPE = sum((|et|/Yt)*100)/n

MSE = sum(et*et)/n

RMSE = sqrt(sum(et*et)/n)

U = RMSE of the forecast/RMSE of the naive forecast.

Value

ME	Mean Error.
MAE	Mean Absolute Error.
MPE	Mean Percent Error (unit: %).
MAPE	Mean Absolute Percent Error (unit: %).
MSE	Mean Square Error.
RMSE	Root of Mean Square Error.
U	Number Theil U.
min.model	The best model follow a criterion to evaluate forecasting model.

Note

This function just receive data frame. You must translate your series to data frames before send to function.

Author(s)

Nguyen Thi Diem My <myntdhg@gmail.com>

Hong Viet Minh <hongvietminh@gmail.com>

References

http://www.tailieu.tv/tai-lieu/bai-giang-quy-trinh-du-bao-khao-sat-du-lieu-va-lua-chon-mo-hinh-224

Examples

```
#----------The moving average models-------------------------
library(TTR)
data(enrollment)
sma5<-ts(SMA(enrollment,5),start=1971)
dsma5.5<-ts(SMA(SMA(enrollment)),start=1971)
cma5<-as.ts(CMA(enrollment,5))
ses.002<-as.ts(SES(enrollment,0.002))

#Translate series to data frame
actual<-data.frame(enrollment)
forecasted<-data.frame(sma5,dsma5.5,cma5,ses.002)

#Comparing forecasting models
av.res(Y=actual,F=forecasted,r=5)


#----------The liner and arima models-------------------------
#Loading data
#data(enrollment)

#Liner model
#t<-1:length(enrollment)
```

```
#lm.model<-lm(enrollment~t)

#Arima modle
#arima.model<-arima(enrollment,order=c(1,1,0))

#Translate residual series to data frame
#actual<-data.frame(enrollment)
#residual.models<-data.frame(lm=lm.model$resid,arima=arima.model$resid)

#Comparing forecasting models
#av.res(Y=actual,E=residual.models)


#----------The fuzzy time series models-------------------------
#Following example(fuzzy.ts1)
#Comparing fuzzy time series models for lh time series
#from example of fuzzy.ts1 function
#av.res(Y=data.frame(lh),F=data.frame(chen10,singh10,heuristic10,chenhsu6))

#Following example(Gfuzzy.ts1)
#Comparing fuzzy time series models for lh time series
#from example of Gfuzzy.ts1 function
#av.res(Y=data.frame(lh),F=KQ3)

#Following example(Gfuzzy.ts2)
#Comparing fuzzy time series models for enrollment time series
#from example of Gfuzzy.ts2 function
#av.res(Y=data.frame(enrollment),F=g.fuzzy1$interpolate)
```

`ChenHsu.bin` *Devide point-bin values in Chen-Hsu model*

Description

Calculating point-bin values, which devece divide fuzzy sets in Chen-Hsu model.

Usage

```
ChenHsu.bin(table, n.subset)
```

Arguments

`table`	Object table1 from answer of fuzzy.ts1 fuction.
`n.subset`	A vector contain fuzzy subset number in every old fuzzy set.

Value

A vector contain point-bin values.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

See Also

GChenHsu.bin for more models at the same time.

Examples

```
#For examples see example(fuzzy.ts1)
```

CMA *Center Moving Average*

Description

Calculating moving average by mothod of centered moving average.

Usage

```
CMA(x, n = 5)
```

Arguments

x	Univariate time series.
n	Number of periods to average over.

Value

Observation series after centered moving.

Note

Before, name of this function is TTT.

Author(s)

Doan Hai Nghi <Hainghi1426262609121094@gmail.com>

See Also

SES

Examples

```
CMA(lh,n=5)
CMA(lh,n=8)
```

Compare.Cs *Comparing and sort Abbasov-Mamedova models or NFTS models*

Description

Comparing and sort Abbasov-Mamedova models or NFTS models according a criterion to evaluate forecasting model (ME, MAE, MPE, MAPE, MSE or RMSE) for C values in Cs.

Usage

```
Compare.Cs(ts, n = 7, w = 7, D1 = 0, D2 = 0, Cs = NULL,
type = "Abbasov-Mamedova", complete = NULL)
```

Arguments

ts	Univariate time series.
n	Number of fuzzy set.
w	The w parameter.
D1,D2	Two proper positive numbers.
Cs	A vector contain C values.
type	Model is choosed to predicts time series by fuzziness, type = "Abbasov-Manedova" or type = "NFTS".
complete	A parameter help connecting from DOC and GDOC functions to Compare.Cs function. User are not allowed to use parameter 'complete'.

Details

Now, this documen are updating.

Value

Table comparing and sort Abbasov-Mamedova models or FVD models.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Examples

```
#Compering Abbasov-Mamedova models
Compare.Cs(lh,n=5,w=7,Cs=seq(0,1,0.01),type="Abbasov-Mamedova")

#Compering NFTS models
Compare.Cs(lh,n=5,w=7,Cs=seq(0,1,0.01),type="NFTS")
```

Descriptives *Descriptives*

Description

Calculating to return answer which are descriptive statistics values for a continuously variable or continuously variables in data frame.

Usage

```
Descriptives(x, plot = FALSE, r = 2, answer = 1, statistic = "ALL")
```

Arguments

x	A continuously variable or a data frame contain continuously variables.
plot	Parameter 'Plot' are used by 2 form: Let plot=TRUE to paint description graph when x is time series. Let plot=list(a1,a2,...) (in a1,a2,...are descriptive statistic values as 'Mean', 'Max',...) to paint comparing graph between variables.
r	Rounds the answer to the specified number of decimal places (default 2).(See round2str for details of r paramicter.)
answer	Form of answers are returned. Let answer=1 or answer=2 (default 1)
statistic	A list contain descriptive statistic values that user want R print screen (default ALL).

Details

Statistic descriptive values are calculated by theory of base statistic.

Value

N	Length sample
NaN	Number NA values
Min	Min value
1sq QU	Value in 25% of interval probabilities
Median	Median value
Mean	Mean value
3rd QU	Value in 75% of interval probabilities
Max	Max value
VAR	Variance value
SD	Standard Deviation
SE	Standard Deviation of the Estimated Means

Note

You must not withdraw discrete variables from data frame. When you let a data frame in to this function which will auto withdraw discrete variables and calculate descriptive statistic to continuously variables.

Author(s)

Mai Thi Hong Diem <maidiemks@gmail.com>

Hong Viet Minh <hongvietminh@gmail.com>

References

Theory of base statistic.

See Also

Frequencies, Dgroup

Examples

```
#Load data
library(MASS)
data(crabs)

#Calculate descriptive statistic to a continuously variable
Descriptives(crabs$FL)

#Calculate descriptive statistic to continuously variables
Descriptives(crabs)
Descriptives(crabs,answer=2)
Descriptives(crabs,answer=2,r=6)

#To just see some descriptive statistic variables
Descriptives(crabs,statistic=list("Min","Mean","Median","Max"))

#Combined paint graph to compare
Descriptives(crabs,plot=list("Mean","SD"))

#Descriptives for time series
Descriptives(lh,plot=TRUE)
```

Dgroup *Discriptives Follow Groups*

Description

Descriptive statistics in group for a continous variable. Usual using to statistic a time series following dates in week or months in year.

Usage

```
Dgroup(x, follow = NULL, r = 2, answer = 1, statistic = "ALL")
```

Arguments

x	A continous variable or a time series.
follow	A factor or a list factor which contain not too two factors.
r	Rounds the answer to the specified number of decimal places (default 2).
answer	Form of answers are returned. Let answer=1 or answer=2 (default 1).
statistic	A list contain descriptive statistic values that user want R print screen (default ALL).

Details

Data is divided into groups by follow and then every group are calculated by Descriptives function.

Value

N	Length sample
NaN	Number NA values
Min	Min value
1sq QU	Value in 25% of interval probabilities
Median	Median value
Mean	Mean value
3rd QU	Value in 75% of interval probabilities
Max	Max value
VAR	Variance value
SD	Standard Deviation
SE	Standard Deviation of the Estimated Means

Note

The function just maximum calculated for two factors.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

References

Theory of base statistic.

See Also

Descriptives,Frequencies

Examples

```
#Factor date
date<-as.factor(c("Tue","Wed","Thu","Fri","Mon","Tue","Wed","Thu","Fri","Mon",
"Tue","Wed","Thu","Fri","Mon","Tue","Wed","Thu","Fri","Mon","Tue","Wed","Thu",
"Fri","Mon","Tue","Wed","Thu","Fri","Mon","Tue","Wed","Thu"))

#Factor hk
hk<-as.factor(c("hk1","hk2","hk1","hk3","hk3","hk1","hk1","hk1","hk2","hk2","hk2",
"hk1","hk2","hk1","hk1","hk1","hk2","hk1","hk1","hk1","hk1","hk2","hk1",
"hk1","hk1","hk1","hk3","hk1","hk3","hk3","hk2","hk3","hk1"))

#A continous variable
coffee<-c(5,6,8,4,3,7,6,0,3,2,3,4,9,1,3,8,7,8,2,3,8,6,4,4,6,7,6,5,2,3,8,4,4)

#Descriptive statistics in group
Dgroup(coffee,r=4,answer=2)
Dgroup(coffee,follow=list(date),r=4)
Dgroup(coffee,follow=date,r=4,answer=2)
Dgroup(coffee,follow=date,r=4,statistic=list("Mean","Max"))
Dgroup(coffee,r=4,follow=list(date,hk),answer=1)
Dgroup(coffee,r=4,follow=list(date,hk),answer=2)
Dgroup(coffee,r=4,follow=list(hk,date),answer=1)
Dgroup(coffee,r=4,follow=list(hk,date),answer=2)
```

DOC *Finding the best C value*

Description

Finding the best C value for Abbasov Mamedova and NFTS models according to DOC algorithm.

Usage

```
DOC(ts, n = 7, w = 7, D1 = 0, D2 = 0, error = 1e-06, k=500,r=13,
CEF = "MSE",type="Abbasov-Mamedova", show.complete = TRUE, keyword)
```

Arguments

ts	Univariate time series.
n	Number of fuzzy set.
w	The w parameter.
D1,D2	Two proper positive numbers.
error	Error of C value is finded by DOC algorithm, which compare the best C value really. Default error = 0.000001.
k	In each iteration of the algorithm, k+1 (or k or k-1) values of C will be considered. The k must be a integer and greater than 499, default k = 500.

r	Display results returned to the specified number of decimal places (default 13). (See round2str for details of r paramicter.)
CEF	One of the criterion to evaluate forecasting model, must be one of "ME","MAE" , "MPE", "MAPE", "MSE" (default), or "RMSE".
type	Model is choosed to predicts time series by fuzziness, type = "Abbasov-Manedova" (default) or type = "NFTS".
show.complete	If TRUE, a graph will appear showing the percentage completed.
keyword	A keyword help connecting from GDOC function to Compare.Cs function. User are not allowed to use parameter 'keyword'.

Value

The best C value and CEF corresponding value.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Vo Van Tai <vvtai@ctu.edu.vn>

See Also

GDOC

Examples

```
#data(enrollment)
#DOC(enrollment,n=7,w=7,error=0.00001,CEF="MSE",type="Abbasov-Mamedova")
#DOC(enrollment,n=7,w=7,error=0.00001,CEF="MAPE",type="NFTS")
```

---

enrollment *Time Series of Enrollment*

Description

A time series of enrollment from 1971 to 1992.

Usage

```
data("enrollment")
```

Source

Chen, S.M., 1996. Forecasting enrollments based on fuzzy time series. Fuzzy Sets and Systems. 81: 311-319.

Chen, S.M. and Hsu, C.C., 2004. A New method to forecast enrollments using fuzzy time series. International Journal of Applied Science and Engineering, 12: 234-244.

Huarng, H., 2001. Huarng models of fuzzy time series for forecasting. Fuzzy Sets and Systems. 123: 369-386.

Singh, S.R., 2008. A computational method of forecasting based on fuzzy time series. Mathematics and Computers in Simulation. 79: 539-554

Examples

```
data(enrollment)
ts.plot(enrollment,col="red")
grid.on()
```

`Fitted.arima` *Fitted for ARIMA model*

Description

Computes the fitted values of an ARIMA model.

Usage

```
Fitted.arima(object)
```

Arguments

`object`	A fitted model from the arima function.

Value

Fitted values.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Pham Minh Truc <trucm1813014@gstudent.ctu.edu.vn>

See Also

`fitted.Arima` in TSA package.

Examples

```
mod<-arima(lh,order=c(0,1,1))
Fitted.arima(mod)
```

forecast.Greg.ts *Forcast group regression models Time-Series*

Description

Forecast from models fitted by Greg.ts.

Usage

```
forecast.Greg.ts(object, model = "ALL", n.ahead = 5, plot = FALSE)
```

Arguments

object	Resul returned from Greg.ts function.
model	Names of models want to forecast.
n.ahead	The number of steps ahead for which prediction is required.
plot	Let plot=TRUE to paint graph of obsevation series and forecast series in future. Let plot=FLASE (default) to do not paint graph.

Value

A data frame where each column is a time series forecast in future from regression model corresponding.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Pham Minh Truc <trucm1813014@gstudent.ctu.edu.vn>

See Also

Greg.ts

Examples

```
data(enrollment)
mod<-Greg.ts(enrollment)
forecast.Greg.ts(mod)
```

`forecastGARCH` *Forecast GARCH Model*

Description

The function get two object from 'Arima' class and 'garch' class, and then calculate to return forecasting answer of mean and variance of next day.

Usage

```
forecastGARCH(fitARMA, fitGARCH, r = 3, trace = FALSE, newxreg = NULL)
```

Arguments

`fitARMA`	A object from 'Arima' class.
`fitGARCH`	A object from 'garch' class.
`r`	Rounds the answer to the specified number of decimal places (default 3). (See `round2str` for details of r paramicter.)
`trace`	Logical. Trace optimizer output?
`newxreg`	A covariates value of next day for ARMAX-GARCH mdels.

Value

`ARCH`	GARCH coefficients.
`ARMA`	ARMA coefficients.
`forecast`	Forecasting answer: Point: forecasting time. res: forecasting residual. res^2: res square. SSL.forecast: forecating mean value. VAR.forecast: forecasting variance value.

Author(s)

Mai Thi Hong Diem <maidiemks@gmail.com>

Hong Viet Minh <hongvietminh@gmail.com>

Examples

```
#Load data
library(TTR)
data(ttrc)

#Calculate SSL series
t<-ts(ttrc[,"Close"],start=1,frequency=5)
```

```
ln.t<-log(t)
r<-diff(ln.t)

#Find a ARIMA model
fit1<-arima(r,order=c(4,0,0))

#Find a GARCH model
res1<-resid(fit1)
library(tseries)
fit2<-garch(res1,order=c(2,1),trace=0)

#Forecasting
forecastGARCH(fit1,fit2,r=6,trace=TRUE)
forecastGARCH(fit1,fit2,r=6)
```

`Frequencies` *Frequencies*

Description

Calculating to return answer which are descriptive statistics values for a discrete variable or discrete variables in data frame.

Usage

```
Frequencies(x, plot = FALSE, r = 2, answer = 1)
```

Arguments

`x`	A discrete variable or a data frame contain discrete variable.
`plot`	Let plot=TRUE to paint pie graph.
`r`	Rounds the answer to the specified number of decimal places (default 2).
`answer`	Form of answers are returned. Let answer=1 or answer=2 (default 1)

Details

Statistic descriptive values are calculated by theory of base statistic.

Value

`N`	Length sample
`NaN`	Number NA values
`xi`	Length of xi

Note

You must not withdraw continuously variables from data frame. When you let a data frame in to this function which will auto withdraw continuously variables and calculate descriptive statistic to discrete variables.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

References

Theory of base statistic.

See Also

Descriptives, Dgroup

Examples

```
#Loading data
library(MASS)
attach(quine)

#Descriptive for a discrete variable
Frequencies(Age,plot=TRUE)

#Descriptive for discrete more variables
Frequencies(quine,answer=2,plot=TRUE)
```

---

fuzzy.ts1 *Chen, Sing, Heuristic and Chen-Hsu models*

Description

Calculating fuzziness of time series with Chen (1996), Singh (2008), Heuristic (Huarng 2001) and Chen-Hsu (2004) models.

Usage

```
fuzzy.ts1(ts, n = 5, D1 = 0, D2 = 0, type = c("Chen", "Singh",
"Heuristic", "Chen-Hsu"), bin = NULL, trace = FALSE, plot = FALSE, grid = FALSE)
```

Arguments

ts	Univariate time series.
n	Number of fuzzy set.
D1,D2	Two proper positive numbers.
type	Type of model.
bin	Point-bin values use to divide fuzzy stes for Chen-Hsu model. If bin=NULL (default) then function just inform information about fuzzy sets.
trace	Let trace=TRUE to print all of calculation results out to creen. Let trace=FALSE (default) to only print fuzzy series out to creen.

plot	Let plot=TRUE to paint graph of obsevation series and fuzzy series. Let plot=FLASE (default) to do not paint graph.
grid	If TRUE, a gray background grid is put on the graph.

Value

When trace = TRUE, results are returned as a list containing the following components.

type	Name of fuzzy model.
table1	Information about fuzzy sets.
table2	Information about fuzzy series of Chen, Sing, Heuristic and Chen-Hsu models (in bin!=NUL).
accuracy	Information about the criterion to evaluate forecasting model.

When trace = FALSE, results is a time series fitted by fuzzy time series model.

Author(s)

Doan Hai Nghi <Hainghi142626260912 1094@gmail.com>

Tran Thi Ngoc Han <tranthingochan01011994@gmail.com>

Hong Viet Minh <hongvietminh@gmail.com>

References

Chen, S.M., 1996. Forecasting enrollments based on fuzzy time series. Fuzzy Sets and Systems. 81: 311-319.

Chen, S.M. and Hsu, C.C., 2004. A New method to forecast enrollments using fuzzy time series. International Journal of Applied Science and Engineering, 12: 234-244.

Huarng, H., 2001. Huarng models of fuzzy time series for forecasting. Fuzzy Sets and Systems. 123: 369-386.

Singh, S.R., 2008. A computational method of forecasting based on fuzzy time series. Mathematics and Computers in Simulation. 79: 539-554

See Also

Gfuzzy.ts1

Examples

```
#Print all of calculation results out to creen.
#par(mfrow=c(2,2))
#data(enrollment)
#fuzzy.ts1(enrollment,n=7,type="Chen",plot=TRUE,grid=TRUE)
#fuzzy.ts1(enrollment,n=7,type="Singh",plot=TRUE)
#fuzzy.ts1(enrollment,n=7,type="Heuristic",plot=TRUE,grid=TRUE)

#Only print fuzzy series out to creen.
#chen10<-fuzzy.ts1(lh,n=10,type="Chen")
```

```
#singh10<-fuzzy.ts1(lh,n=10,type="Singh")
#heuristic10<-fuzzy.ts1(lh,n=10,type="Heuristic")

#Useing ChenHsu.bin function to find divide point fuzzy set values.
#a<-fuzzy.ts1(lh,n=5,type="Chen-Hsu",plot=1)
#b<-ChenHsu.bin(a$table1,n.subset=c(1,2,1,1,1))
#chenhsu6<-fuzzy.ts1(lh,type="Chen-Hsu",bin=b,plot=1,trace=1)
```

---

fuzzy.ts2 *Abbasov Mamedova model and FVD model*

Description

Calculating fuzziness and forecast time series by fuzziness method according to Abbasov - Manedova (2010) and NFTS models.

Usage

```
fuzzy.ts2(ts, n = 7, w = 7, D1 = 0, D2 = 0, C = NULL, forecast = 5,
r = 12, trace = FALSE, plot = FALSE, grid = FALSE, type = "Abbasov-Mamedova")
```

Arguments

ts	Univariate time series.
n	Number of fuzzy set.
w	The w parameter.
D1, D2	Two proper positive numbers.
C	A optional constant.
forecast	Number of points to forecast in future.
r	Display results returned to the specified number of decimal places (default 12). (See round2str for details of r paramicter.)
trace	Let trace=TRUE to print all of calculation results out to creen. Let trace=FALSE (default) to only print forecasting series out to creen.
plot	Let plot=TRUE to paint graph of obsevation series and fuzzy series. Let plot=FLASE (default) to do not paint graph.
grid	If TRUE, a gray background grid is put on the graph.
type	Model is choosed to predicts time series by fuzziness, type = "Abbasov-Manedova" or type = "NFTS".

Value

When trace = TRUE, results are returned as a list containing the following components.

- `type` The value of type.
- `table1` Information about changing fuzzy sets consit four column: set is name of the fuzzy sets, low and up are upper and lower bounds of the fuzzy sets, and mid is middle values corresponding every fuzzy set.
- `table2` Series - observation consit three column: point is time of observation, ts is the original series, and diff.ts is changing series from original series.
- `table3` The change fuzzy of original series.
- `table4` Series - interpolation consit three column: point is time of interpolation, interpolate is the series - interpolation, and diff.interpolate is changing series from series - interpolation.
- `table5` Forecasting series consit three column: point is time of forecast, forecast is the forecasting series, and diff.forecast is changing series from forecasting series.
- `table6` The change fuzzy of forecasting series.
- `accuracy` Information about the criterion to evaluate forecasting model.

When trace = FALSE, results are returned as a list containing two components.

- `interpolate` Series - interpolation.
- `forecast` Forecasting series.

Author(s)

Doan Hai Nghi <Hainghi1426262609121094@gmail.com>

Tran Thi Ngoc Han <tranthingochan01011994@gmail.com>

Hong Viet Minh <hongvietminh@gmail.com>

References

Abbasov, A.M. and Mamedova, M.H., 2003. Application of fuzzy time series to population forecasting, Proceedings of 8th Symposion on Information Technology in Urban and Spatial Planning, Vienna University of Technology, February 25-March1, 545-552.

See Also

Gfuzzy.ts2

Examples

```
#data(enrollment)
#layout(1:2)
#NF.mod<-fuzzy.ts2(enrollment,n=7,w=7,C=0.0001,forecast=11,trace=TRUE,plot=TRUE,type="NFTS")
#AM.mod<-fuzzy.ts2(enrollment,n=5,w=5,C=0.01,forecast=5,plot=TRUE,type="Abbasov-Mamedova")
#NF.mod
#AM.mod
```

```
#Finding the best C value by DOC function
#Abbasov-Mamedova model
#str.C1<-DOC(enrollment,n=7,w=7,D1=0,D2=0,CEF="MAPE",type="Abbasov-Mamedova")
#C1<-as.numeric(str.C1[1])
#fuzzy.ts2(enrollment,n=7,w=7,D1=0,D2=0,C=C1,forecast=5,type="Abbasov-Mamedova")

#NFTS model
#str.C2<-DOC(enrollment,n=7,w=7,D1=0,D2=0,CEF="MAPE",type="NFTS")
#C2<-as.numeric(str.C2[1])
#fuzzy.ts2(enrollment,n=7,w=7,D1=0,D2=0,C=C1,forecast=5,type="NFTS")
```

`GChenHsu.bin` *Devide point-bin values in more Chen-Hsu models at the same time*

Description

Calculating point-bin values, which devece divide fuzzy sets in more Chen-Hsu models (the result returned form Gfuzzy.ts1(...,type=c(...,"Chen-Hsu"),bin=NULL) function).

Usage

```
GChenHsu.bin(list, n.subset)
```

Arguments

`list`	The information of fuzzy sets in Chen-Hsu models form Gfuzzy.ts1(...,type=c(...,"Chen-Hsu"),bin=NULL) function.
`n.subset`	A list where each component is a vector containing fuzzy subset number in every old fuzzy set. The each components of n.subset must be compatible with each componets of list.

Details

GChenHsu.bin function use `ChenHsu.bin` function to calculate point-bin values in each Chen-Hsu model compatible each components of list.

Value

A list where each components is a vector contain point-bin values for Chen-Hsu models.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Vo Van Tai <vvtai@ctu.edu.vn>

See Also

Using ChenHsu.bin function in case only a Chen-Hsu model.

Examples

```
#For examples see example(Gfuzzy.ts1)
```

GDOC *Finding the best C values*

Description

Finding the best C values for more Abbasov Mamedova and NFTS models according to DOC algorithm at the same time.

Usage

```
GDOC(ts, n = 7, w = 7, D1 = 0, D2 = 0, error = 1e-06, k = 500, r = 13,
CEF = "MSE", type = "Abbasov-Mamedova", show.complete = TRUE)
```

Arguments

ts	Univariate time series.
n	A numeric vector where each element is number of fuzzy set.
w	A numeric vector where each element is w parameter.
D1,D2	Two proper positive numbers.
error	Error of C value is finded by DOC algorithm, which compare the best C value really. Default error = 0.000001.
k	In each iteration of the algorithm, k+1 (or k or k-1) values of C will be considered. The k must be a integer and greater than 499, default k = 500.
r	Display results returned to the specified number of decimal places (default 13). (See round2str for details of r paramicter.)
CEF	One of the criterion to evaluate forecasting model, must be one of "ME","MAE" , "MPE", "MAPE", "MSE" (default), or "RMSE".
type	A character vector where each element is choosing model to predicts time series by fuzziness, type = "Abbasov-Manedova" (default) or type = "NFTS" or both.
show.complete	If TRUE, a graph will appear showing the percentage completed.

Details

GDOC function consider length(n)*length(w)*length(type) models combining from three parameter n, w and type, and then using DOC function finding the best C values forone by models.

Value

A list contain two components where the first component is the best C values of Abbasov-Mamedova models and the second component is the best C values of NFTS models.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Vo Van Tai <vvtai@ctu.edu.vn>

See Also

Using DOC function in case only a Abbasov-Mamedova or NFTS model.

Examples

```
#For examples see example(Gfuzzy.ts2)
```

Gfuzzy.ts1 *Chen, Sing, Heuristic and Chen-Hsu models at the same time*

Description

Calculating fuzziness of time series with Chen (1996), Singh (2008), Heuristic (Huarng 2001) and Chen-Hsu (2004) models at the same time.

Usage

```
Gfuzzy.ts1(ts, n = 5, D1 = 0, D2 = 0, type = "Chen", bin = NULL,
plot = FALSE, grid = FALSE)
```

Arguments

ts	Univariate time series.
n	A numeric vector where each element is number of fuzzy set.
D1,D2	Two proper positive numbers.
type	A character vector where each element is type of models.
bin	A list where each component is point-bin values use to divide fuzzy stes for Chen-Hsu models. If bin=NULL (default) then function just inform information about fuzzy sets compatible with each Chen-Hsu model.
plot	Let plot=TRUE to paint graph of obsevation series and fuzzy series. Let plot=FLASE (default) to do not paint graph.
grid	If TRUE, a gray background grid is put on the graph.

Details

Gfuzzy.ts1 function consider length(n)*length(type) models combining from two parameter n and type.

Value

A data frame where each column is a time series fitted by fuzzy time series model corresponding.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Vo Van Tai <vvtai@ctu.edu.vn>

References

Chen, S.M., 1996. Forecasting enrollments based on fuzzy time series. Fuzzy Sets and Systems. 81: 311-319.

Chen, S.M. and Hsu, C.C., 2004. A New method to forecast enrollments using fuzzy time series. International Journal of Applied Science and Engineering, 12: 234-244.

Huarng, H., 2001. Huarng models of fuzzy time series for forecasting. Fuzzy Sets and Systems. 123: 369-386.

Singh, S.R., 2008. A computational method of forecasting based on fuzzy time series. Mathematics and Computers in Simulation. 79: 539-554

See Also

Using fuzzy.ts1 function in case only a fuzzy time series model.

Examples

```
#Step 1: Analyze fuzzy time series actual series of n fuzzy set (n is
#number fuzzy set choosed in the first times, in our case n = 5, 7 and 9)
#to has information about fuzzy sets.
#KQ1<-Gfuzzy.ts1(lh,n=c(5,7,9),type=c("Chen","Singh","Heuristic","Chen-Hsu"))


#Step 2: Finding bin-point values divide fuzzy sets second times.
#v1<-c(1,1,1,1,1)
#v2<-c(1,2,2,1,2,1,1)
#v3<-c(1,1,2,4,1,2,1,1,1)
#KQ2<-GChenHsu.bin(KQ1,n.subset=list(v1,v2,v3))


#Step 3 Analyze fuzzy time series by Chen-Hsu more times with new fuzzy
#sets from step 2.
#KQ3<-Gfuzzy.ts1(lh,n=c(5,7,9),type=c("Chen","Singh","Heuristic",
#"Chen-Hsu"),bin=KQ2,plot=1,grid=1)
#KQ3
```

`Gfuzzy.ts2` *Updating*

Description

Updating

Usage

```
Gfuzzy.ts2(ts, n = 7, w = 7, D1 = 0, D2 = 0, C = list(C1 = NULL, C2 = NULL),
forecast = 5, plot = FALSE, grid = FALSE, type = "Abbasov-Mamedova")
```

Arguments

`ts`	Univariate time series.
`n`	A numeric vector where each element is number of fuzzy set.
`w`	A numeric vector where each element is w parameter.
`D1, D2`	Two proper positive numbers.
`C`	A list consiting 2 component C1 and C2 or a rusult object from GDOC function.
`forecast`	Number of points to forecast in future.
`plot`	Let plot=TRUE to paint graph of obsevation series and fuzzy series. Let plot=FLASE (default) to do not paint graph.
`grid`	If TRUE, a gray background grid is put on the graph.
`type`	Model is choosed to predicts time series by fuzziness, type = "Abbasov-Manedova" or type = "NFTS" or both.

Details

Gfuzzy.ts2 function consider length(n)*length(w)*length(type) models combining from three parameter n, w and type, and then using `fuzzy.ts2` function analyze for each submodel.

Value

A list with three component.

`information`	Explaining for lables in annotation on the graph also as name of columns in interpolate and forecast components.
`interpolate, forecast`	two data frame where each column is a time series interpolate and forecasted in future by fuzzy time series model corresponding.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Vo Van Tai <vvtai@ctu.edu.vn>

References

Abbasov, A.M. and Mamedova, M.H., 2003. Application of fuzzy time series to population forecasting, Proceedings of 8th Symposion on Information Technology in Urban and Spatial Planning, Vienna University of Technology, February 25-March1, 545-552.

See Also

Using `Gfuzzy.ts1` function in case only a fuzzy time series model.

Examples

```
#data(enrollment)
#g.C<-GDOC(enrollment,n=c(5,7,9),w=c(7,9),D1=0,D2=0,
#CEF="MSE",type=c("Abbasov-Mamedova","NFTS"))
#g.fuzzy1<-Gfuzzy.ts2(enrollment,n=c(5,7,9),w=c(7,9),D1=0,D2=0,C=g.C,forecast=5,
#plot=1,grid=0,type=c("Abbasov-Mamedova","NFTS"))
```

Greg.ts *Group regression models for Time-Series*

Description

Building some regression models for time series.

Usage

```
Greg.ts(ts, p.max = 3, r = 4, plot = FALSE)
```

Arguments

`ts`	Univariate time series.
`p.max`	Greatest level of polynomial models.
`r`	Display results returned to the specified number of decimal places (default 3). (See `round2str` for details of r paramicter.)
`plot`	Let plot=TRUE to paint graph of obsevation series and interpolate series. Let plot=FLASE (default) to do not paint graph.

Details

luy thua: Y = a*t^b

luong giac: Y = a + b*sin(t) + c*cos(t)

hyperbol: Y = a + b/t

CS cong: ...

CS nhan

Value

Results are returned as a list containing the following components.

`Models` Formula of regression models.

`Interpolate, Error`
: Two data frame where each column is a time series interpolate and error from regression model corresponding.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Pham Minh Truc <trucm1813014@gstudent.ctu.edu.vn>

References

Vo Van Tai, 2012. Du bao san luong lua Viet Nam bang cac mo hinh toan hoc, Tap chi Khoa hoc Dai hoc Can Tho. 23b125-134.

See Also

The `forecast.Greg.ts` function using to predict for models fitted by Greg.ts function.

Examples

```
data(enrollment)
Greg.ts(enrollment)
```

---

grid.on *Grid on*

Description

Painting to line graph.

Usage

```
grid.on(h = TRUE, v = TRUE, col = "gray", nv = 1, nh = 1, lty = 3)
```

Arguments

`h`	Let h=FALSE to do not paint grid in horizontal.
`v`	Let v=FALSE to do not paint grid in vertical.
`col`	The color for grid lines.
`nv`	The number grid lines between two point in x axis.
`nh`	The number grid lines between two point in y axis.
`lty`	The grid line type.

Value

A grid is painted in graph.

Author(s)

Hong Vet Minh <hongvietminh@gmail.com>

Examples

```
plot(lh)
grid.on(h=TRUE,v=FALSE)
plot(co2)
grid.on(col="red",nv=3)
```

`pmax` *The lag maximum value*

Description

Choose number of lags for endogenous variable to be included for augmented Dickey-Fuller unit root test according Schwert (1989).

Usage

```
pmax(ts)
```

Arguments

`ts`	Univariate time series.

Details

p_max = [12*(T/4)^1/4]

where T is number of observations.

Note: [1.6] = 1

Value

Number of lags for endogenous variable to be included for augmented Dickey-Fuller unit root test according Schwert (1989).

Author(s)

Hong Viet Minh

Examples

```
data(enrollment)
p<-pmax(enrollment)
library(urca)
summary(ur.df(enrollment,type="trend",lag=p,selectlag="BIC"))
```

PrintAIC *Print AIC Values*

Description

Calculates and outputs AIC value for some models including ARMA, ARIMA, SARIMA, ARMAX, ARIMAX, SARIMAX, ARCH and GARCH. To classify and extracts the best model by AIC values.

Usage

```
PrintAIC(DataTimeSeries, order = c(p, d = NULL, q = NULL),
seas = list(order = c(P = NULL, D = NULL, Q = NULL), frequency = NULL),
type = NULL,xreg=NULL)
```

Arguments

`DataTimeSeries`	Univariate time series.
`order`	If type="ARMA" (or ARMAX) then 'order' is a vector contain two positive integer which are order of ARMA model (or ARMAX model). If type="ARIMA" (or ARIMAX) then 'order' is a vector contain three positive integer which are order of ARIMA model (or ARIMAX model). If type="SARIMA" (or SARIMAX) then 'order' is a vector contain three positive integer which are order of ARIMA model (or ARIMAX model) of the non-seasonal part of the SARIMA model (or SARIMAX model). If type="ARCH" then 'order' is a positive integer which are order of ARCH model. If type="GARCH" then 'order' is a vector contain two positive integer which are order of GARCH model.
`seas`	A list contain two part which are 'order' and 'frequency'. 'order' part is a vector contain three positive integer which are order of ARIMA model of the seasonal part of the SARIMA model (or SARIMAX model). 'frequency' part is frequency of observation series.
`type`	A character string specifying the type of models, must be one of "ARMA", "ARIMA", "SARIMA", "ARCH", "GARCH", "ARMAX", "ARIMAX" or "SARIMAX".
`xreg`	Optionally, a vector or matrix of external regressors, which must have the same number of rows as x.

Details

The first, function identify type of models according to 'type' parameter. The next, test other parameters. All of parameters are reasonable, function will combine orders of models. And then, to calculate AIC value of each model. The last step, performing ranked and extracting the best model.

Value

`mohinh`	Calculation results.
`best`	The best model following AIC value.

Note

You must be careful with 'order' and 'type' parameter of models.

Author(s)

Tran Thi Ngoc Han <tranthingochan01011994@gmail.com>

Mai Thi Hong Diem <maidiemks@gmail.com>

Hong Viet Minh <hongvietminh@gmail.com>

References

Nguyen Thi Diem My va Hong Viet Minh, Phan tich chuoi thoi gian voi su ho tro cua package AnalyzeTS.

Hong Viet Minh, Luan van tot nghiep dai hoc: Phan tich so lieu thong ke voi ngon ngu R.

Examples

```
#----A time series--------------------------------
sl<-ts(c(180,165,110,126,125,134,163,153,122,171,171,155
,175,248,99,187,173,147,184,108,171,195,192,163))

#----The ARMA models--------------------------------
PrintAIC(sl,order=c(1,4),type="ARMA")

#----The ARIMA models--------------------------------
PrintAIC(sl,order=c(1,1,4),type="ARIMA")

#----The SARIMA models--------------------------------
PrintAIC(sl,order=c(1,1,4),seas=list(order=c(0,0,1),
frequency=4),type="SARIMA")

#----The ARCH models--------------------------------
PrintAIC(sl,order=c(4),type="ARCH")

#----The GARCH models--------------------------------
PrintAIC(sl,order=c(1,4),type="GARCH")

#----The ARIMAX models--------------------------------
#A factor
```

```
date<-as.factor(c("Tue","Wed","Thu","Fri","Mon","Tue","Wed",
"Thu","Fri","Mon","Tue","Wed","Thu","Fri","Mon","Tue","Wed",
"Thu","Fri","Mon","Tue","Wed","Thu","Fri","Mon","Tue","Wed",
"Thu","Fri","Mon","Tue","Wed","Thu"))
#Observation series.
coffee<-c(5,6,8,4,3,7,6,0,3,2,3,4,9,1,3,8,7,8,2,3,8,6,4,
4,6,7,6,5,2,3,8,4,4)
coffee<-ts(coffee,start=c(1,2),frequency=5)
Mon<-1*(date=="Mon")
event<-data.frame(Mon)
PrintAIC(coffee,order=c(2,2),xreg=event,type="ARMAX")
```

---

round2str *rounded and converted to string of numbers*

Description

Rounded and converted to string of numbers to the specified number of decimal places.

Usage

```
round2str(x, r = 12)
```

Arguments

x	A numeric, vector, matrix or data frame of numeric.
r	A integer indicating the number of decimal places.

Details

Unlike `round` function, round2str function cross-section numbers to the specified number of decimal places and then converted result to string.

Author(s)

Hong Viet Minh <hongvietminh@gmail.com>

Examples

```
a<-rnorm(3)
a
round2str(a,r=2)
round2str(a,r=5)
```

`SES` *Simple Exponential Smoothing*

Description

Calculating simple exponential smoothing for a time series.

Usage

```
SES(ts, alpha = 0.5, s0 = NULL)
```

Arguments

`ts`	Univariate time series.
`alpha`	The smoothing parameter, 0 < alpha < 1 (default 0.5).
`s0`	Original estimate of s0 value.

Details

alpha = 2 / (n + 1) in n is moving periodic.

Value

Observation series after.

Note

Before, name of this function is TM.

Author(s)

Doan Hai Nghi <Hainghi1426262609121094@gmail.com>

Hong Viet Minh <hongvietminh@gmail.com>

References

https://www.otexts.org/fpp/7/1

See Also

`CMA`

Examples

```
SES(lh,alpha=0.5)
SES(lh,alpha=0.8)
```

Index