

# Package ‘monitor’

January 5, 2006

**Title** Dynamic Systems Estimation - monitoring extensions

**Description** Multivariate Time Series - monitoring extensions

**Depends** R (>= 2.0.0), tframe (>= 2006.1-1), dse1 (>= 2006.1-1), dse2 (>= 2006.1-1), dsepadi (>= 2006.1-1), syskern (>= 2006.1-1)

**Suggests** setRNG (>= 2004.4-1)

**Version** 2006.1-1

**LazyLoad** yes

**License** Free. See the LICENCE file for details.

**Author** Paul Gilbert <pgilbert@bank-banque-canada.ca>

**Maintainer** Paul Gilbert <pgilbert@bank-banque-canada.ca>

**URL** <http://www.bank-banque-canada.ca/pgilbert>

## R topics documented:

checkForValueChanges . . . . .	2
combinationMonitoring . . . . .	2
combineAndForecast . . . . .	4
fprint . . . . .	5
simpleMonitoring . . . . .	6
tags . . . . .	7
testEqual.tagged . . . . .	8
tfplot.combinedForecast . . . . .	9
tfwindow.tagged . . . . .	10

<b>Index</b>	<b>11</b>
--------------	-----------

---

checkForValueChanges

*Simple Monitoring Utility Functions*

---

## Description

Utility functions for simple monitoring.

## Usage

```
checkForValueChanges(data.names, verification.data,
    discard.current=FALSE, ignore.before= NULL, fuzz=1e-10)
checkForFileDateChanges(data.names, verification.dates)
watch.data(data.names, previous.data=NULL, mail.list="gilp", error.mail.list=NULL,
    message.title="Data Monitor\n", message.subject="Data Monitor", message.footer="")
```

## Details

Internal functions, not to be called by user.

## Value

depends

## See Also

[simpleMonitoring](#)

---

combinationMonitoring

*Combination Monitoring*

---

## Description

Automatic monitoring with e-mail of results

## Usage

```
combinationMonitoring(model, data.names,
    previous.data=NULL,
    overriding.data.names=NULL,
    restrict.overriding.data=TRUE, overriding.horizon=0,
    mail.list=NULL,
    error.mail.list=NULL,
    message.title="Combination Monitoring",
    message.subject="Combination Monitoring",
    message.footer=NULL,
    show.start= c(0,-3),
    show.end   = c(0,12),
    report.variables=seriesNames(data.names),
```

```

data.sub.heading=NULL,
data.tag=" ",
future.inputData.tag="p",
overriding.data.tag="m",
overlapping.period.forecast.tag="g",
forecast.tag="f",
run.again=FALSE,
save.as=NULL)

```

## Arguments

```

model          see simpleMonitoring.
data.names     see simpleMonitoring.
previous.data  see simpleMonitoring.
overriding.data.names
               a TSdata (names) object.
restrict.overriding.data
               a logical indicating if restrict.overriding.data should be used to
               truncate the resticion.
overriding.horizon
               an integer indicating the horizon of the restriction.
mail.list      see simpleMonitoring.
error.mail.list
               see simpleMonitoring.
message.title  see simpleMonitoring.
message.subject
               see simpleMonitoring.
message.footnote
               see simpleMonitoring.
show.end       see simpleMonitoring.
show.start     see simpleMonitoring.
report.variables
               see simpleMonitoring.
data.sub.heading
               see simpleMonitoring.
data.tag       see simpleMonitoring.
future.inputData.tag
               .
overriding.data.tag
               .
overlapping.period.forecast.tag
               .
forecast.tag   see simpleMonitoring.
run.again     see simpleMonitoring.
save.as       .

```

**Details**

This function allows for -combining forecasts (ie. monitoring or other forecast data) -input (policy) projections. If these features are not needed, see `simpleMonitoring`. `mail.list` and `error.mail.list` should be single strings (not vectors) but the string can contain multiple user ids for mail. `overriding.data.names` indicates a source for data which should be used in place of model forecasts (e.g. preliminary data from a source or data from another forecast). If `overriding.data.names=NULL` then no overriding data is used. `report.variables` indicates output variables which are reported. If `NULL`, then all outputs are reported. `show.end` is min of this and `overriding.data$input` if needed.

This function allows for the use of over-riding data which may come from other forecasts or monitoring and can be used to augment (and replace) actual data. Overriding data is used in place of data and model forecasts to the horizon for which it is available. Also, input (policy) variable forecasts can be used. NB. The combination is not in the sense of averaging together forecasts.

**Value**

Invisibly return latest data for next comparison. This function is run mainly for its side effects.

**See Also**

[simpleMonitoring](#)

---

combineAndForecast *Simple Monitoring Utility Functions*

---

**Description**

Utility functions for simple monitoring.

**Usage**

```
combineAndForecast(model, new.data,
  overlapping.period.forecast.tag="g", forecast.tag="f")
reconstruct.combinedForecast(combinedForecast)
construct.data.to.override.horizon(new.data, model, plot=TRUE, forecast.tag="f")
get.overriding.data(file="overriding.data",
  first.input="", first.output="", second.output="", m=1, p=10)
restrict.overriding.data(data, overriding.horizon=0)
```

**Details**

Internal functions, not to be called by user.

**Value**

depends

**See Also**

[combinationMonitoring](#)

---

fprint*Formatted Printing of Time Series*

---

## Description

Generate a formatted character matrix of time series data.

## Usage

```
fprint(x, super.title=NULL, sub.title=NULL,  
      digits=options()$digits, space=" ", file=NULL, append=FALSE)  
## S3 method for class 'tagged':  
fprint(x, super.title=NULL, sub.title=NULL,  
      digits=options()$digits, space=" ", file=NULL, append=FALSE)
```

## Arguments

<code>x</code>	a matrix of time series data.
<code>super.title</code>	a string to be used as top heading.
<code>sub.title</code>	a string to be used as second level of headings.
<code>digits</code>	number of digits to print.
<code>space</code>	a string to be used between columns of data.
<code>file</code>	name of a file to generate.
<code>append</code>	If TRUE output is appended to the file, otherwise the file is overwritten.

## Details

This is a generic method for print a table of time series data with formatting control. Currently the only specific method is `fprint.tagged`.

## Value

A character matrix.

## See Also

[print tfprint](#)

---

simpleMonitoring      *Simple Monitoring*

---

## Description

Automatic monitoring with e-mail of results.

## Usage

```
simpleMonitoring(model, data.names,
previous.data=NULL,
mail.list=NULL,
error.mail.list=Sys.info()[["user"]],
message.title="Simple Monitoring",
message.subject="Simple Monitoring",
message.footnote=NULL,
show.start= c(0,-3),
show.end   = c(0,12),
report.variables= seriesNames(data.names),
data.sub.heading=NULL,
data.tag=" ",
forecast.tag="f",
run.again=FALSE,
save.as=NULL)
```

## Arguments

model	a TSmodel.
data.names	a TSdata (names) object.
previous.data	a TSdata object similar to that returned by freeze(data.names). Used to check if the data has been updated.
mail.list	string containing user ids for mail
error.mail.list	string contain user ids for mail
message.title	string or vectors of strings placed at the top of the message.
message.subject	string used as the message subject.
message.footnote	string or vectors of strings placed at the bottom of the message.
show.end	integer indicating the number of periods after the end of data which should be displayed.(i.e. number of forecast periods)
show.start	negative integer indicating the number of periods before the end of data (i.e. history) which should be displayed. It is added to the end (so it should be negative).
report.variables	indicates output variables which are reported. It should be a vector of strings corresponding to a subset of names returned by seriesNames.

<code>data.sub.heading</code>	string or vectors of strings placed at the top of the data.
<code>data.tag</code>	matrix of strings placed beside data points in the report.
<code>forecast.tag</code>	matrix of strings placed beside data points in the report.
<code>run.again</code>	logical indicating that the monitoring should be run regardless of data updates.
<code>save.as</code>	optional string giving file name in which to save details of the data and model (useful for debugging).

### Details

`mail.list` and `error.mail.list` should be single strings (not vectors) but the string can contain multiple user ids for mail. If `mail.list` is NULL (default) then mail is not sent (useful for testing). If `error.mail.list` is NULL then mail is not sent (useful for testing). The default for `error.mail.list` is the result of `Sys.info()[["user"]]`. This version does not allow for -combining forecasts (ie. `monitoring.data` or overriding data) -input (policy) projections See `combinationMonitoring` for these features.

### Value

Invisibly return latest data for next comparison. This function is run mainly for its side effects.

### See Also

[combinationMonitoring Sys.mail](#)

---

tags

*Tagged Matrices*

---

### Description

Matrices with an decription for each point.

### Usage

```
tags(x)
tags(x) <- value
tagged(x, tags)
## Default S3 method:
tagged(x, tags)
## S3 method for class 'TSdata':
tagged(x, tags)
is.tagged(obj)
```

### Arguments

<code>x</code>	a matrix or TSdata object.
<code>value</code>	same as tags below.
<code>tags</code>	if x is a matrix then tags should be a matrix of strings of the same dimension or a scalar string which is expanded to the dimension of x. If x is TSdata then tags should be a list with elements input and output, each with tags as for a matrix.
<code>obj</code>	any object.

## Details

The matrix of descriptive information is assigned as an attribute of the matrix. Most testing of the methods for this class has been with a single character tag which can be used as a flag, for example, to indicate the different sources for the data points.

## Value

A matrix of class "tagged" or a TSdata object with specific class "tagged"

## See Also

[simpleMonitoring TSdata](#)

## Examples

```
x <- tagged(matrix(rnorm(100), 50,2), "r")
is.tagged(x)
```

---

testEqual.tagged	<i>Specific Methods for Testing Equality</i>
------------------	--

---

## Description

See the generic function description.

## Usage

```
## S3 method for class 'tagged':
testEqual(obj1, obj2, fuzz= 1e-16)
```

## Arguments

obj1	object to be compared with obj2.
obj2	object to be compared with obj1.
fuzz	tolerance for numerical comparisons.

## See Also

[testEqual](#)



---

tfplot.combinedForecast

*Specific Methods for tfplot*


---

## Description

See the generic function description.

## Usage

```
## S3 method for class 'combinedForecast':
tfplot(x,
       start=tfstart(x$data$output), end=tfend(x$data$output),
       select.inputs=NULL, select.outputs=NULL,
       Title="Projection", xlab=NULL, ylab=NULL,
       graphs.per.page=5, mar=par()$mar, verbose=FALSE, ...)
```

## Arguments

x	object to be plotted.
start	start of plot. (passed to tfwindow)
end	end of plot. (passed to tfwindow)
select.inputs	vector of integers or strings indicating inputs to be plotted.
select.outputs	vector of integers or strings indicating outputs to be plotted.
Title	title for plot.)
xlab	xlab for plot.)
ylab	ylab for plot.)
mar	margins passed to plot. See par.
graphs.per.page	integer indicating number of graphs to place on a page.
verbose	logical indicating if additional information is provided.
...	arguments passed to other methods.

## See Also

[tfplot](#)

---

tfwindow.tagged	<i>Specific Methods for tframed Data</i>
-----------------	--

---

**Description**

See the generic function description.

**Usage**

```
## S3 method for class 'tagged':
tfwindow(x, tf=NULL, start=tfstart(tf), end=tfend(tf), warn=TRUE)
## S3 method for class 'tagged':
tbind(x, mat2, ..., pad.start=TRUE, pad.end=TRUE, warn=TRUE)
## S3 method for class 'tagged':
splice(mat1, mat2, tag1=tags(mat1), tag2=tags(mat2), ...)
## S3 method for class 'tagged':
selectSeries(x, series=seq(ncol(x)))
## S3 method for class 'tagged':
tframe(x) <- value
```

**Arguments**

x	see the generic function.
start	see the generic function.
end	see the generic function.
tf	see the generic function.
pad.start	see the generic function.
pad.end	see the generic function.
warn	see the generic function.
mat1	see the generic function.
mat2	see the generic function.
tag1	tags for the first matrix. See tags.
tag2	tags for the second matrix. See tags.
series	see the generic function.
value	see the generic function.
...	(further arguments, currently disregarded)

**See Also**

[tfwindow](#), [tbind](#), [trimNA](#) [splice](#) [selectSeries](#)

# Index

**\*Topic internal**  
  checkForValueChanges, 1  
  combineAndForecast, 4

**\*Topic programming**  
  fprintf, 4

**\*Topic ts**  
  combinationMonitoring, 2  
  fprintf, 4  
  simpleMonitoring, 5  
  tags, 7  
  testEqual.tagged, 8  
  tfplot.combinedForecast, 8  
  tfwindow.tagged, 9

**\*Topic utilities**  
  fprintf, 4

checkForFileDateChanges  
  (*checkForValueChanges*), 1  
checkForValueChanges, 1  
combinationMonitoring, 2, 4, 6  
combineAndForecast, 4  
construct.data.to.override.horizon  
  (*combineAndForecast*), 4  
  
fprintf, 4  
  
get.overriding.data  
  (*combineAndForecast*), 4  
  
is.tagged(tags), 7  
  
print, 5  
  
reconstruct.combinedForecast  
  (*combineAndForecast*), 4  
restrict.overriding.data  
  (*combineAndForecast*), 4  
  
selectSeries, 10  
selectSeries.tagged  
  (*tfwindow.tagged*), 9  
simpleMonitoring, 2, 3, 5, 7  
splice, 10  
splice.tagged(*tfwindow.tagged*), 9  
Sys.mail, 6  
  
tagged(tags), 7  
tags, 7  
tags<- (tags), 7  
tbind, 10  
tbind.tagged(*tfwindow.tagged*), 9  
testEqual, 8  
testEqual.tagged, 8  
tfplot, 9  
tfplot.combinedForecast, 8  
tfprint, 5  
tframe<- .tagged  
  (*tfwindow.tagged*), 9  
tfwindow, 10  
tfwindow.tagged, 9  
trimNA, 10  
TSdata, 7  
  
watch.data  
  (*checkForValueChanges*), 1