

Package ‘fast’

August 2, 2015

Type Package

Title Implementation of the Fourier Amplitude Sensitivity Test (FAST)

Version 0.64

Date 2015-08-01

Author Dominik Reusser

Depends zoo

Maintainer Dominik Reusser <reusser@pik-potsdam.de>

Description The Fourier Amplitude Sensitivity Test (FAST) is a method to determine global sensitivities of a model on parameter changes with relatively few model runs. This package implements this sensitivity analysis method.

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2015-08-02 16:02:45

R topics documented:

fast-package	2
double_serie	3
example_model2	4
fast_parameters	5
freq_cukier	6
na2mean	7
p.sensitivity	8
rerange	9
S	10
s	11
sa	12
sensitivity	13
sensitivity_rep	14

Index	16
--------------	-----------

fast-package

Implementation of the Fourier Amplitude Sensitivity Test (FAST)

Description

The Fourier Amplitude Sensitivity Test (FAST) is a method to determine global sensitivities of a model on parameter changes with relatively few model runs. This package implements this sensitivity analysis method.

Details

Package: fast
Type: Package
Version: 0.5
Date: 2007-12-15
License: GPL2

Generate a set of parameter sets with the function `fast_parameters`. Run your model with each parameter set. `sensitivity` then evaluates the sensitivities of the model results on each of the parameters.

Author(s)

Dominik Reusser Maintainer: Dominik Reusser <dreusser@uni-potsdam.de>

References

Reusser, Dominik E., Wouter Buytaert, and Erwin Zehe. "Temporal dynamics of model parameter sensitivity for computationally expensive models with FAST (Fourier Amplitude Sensitivity Test)." *Water Resources Research* 47 (2011): W07551.

CUKIER, R. I.; LEVINE, H. B. & SHULER, K. E. Non-Linear Sensitivity Analysis Of Multi-Parameter Model Systems *Journal Of Computational Physics*, 1978 , 26 , 1-42

CUKIER, R. I.; FORTUIN, C. M.; SHULER, K. E.; PETSCHKE, A. G. & SCHAIBLY, J. H. Study Of Sensitivity Of Coupled Reaction Systems To Uncertainties In Rate Coefficients .1. Theory *Journal Of Chemical Physics*, 1973 , 59 , 3873-3878

SCHAIBLY, J. H. & SHULER, K. E. Study Of Sensitivity Of Coupled Reaction Systems To Uncertainties In Rate Coefficients .2. Applications *Journal Of Chemical Physics*, 1973 , 59 , 3879-3888

CUKIER, R. I.; SCHAIBLY, J. H. & SHULER, K. E. Study Of Sensitivity Of Coupled Reaction Systems To Uncertainties In Rate Coefficients .3. Analysis Of Approximations *Journal Of Chemical Physics*, 1975 , 63 , 1140-1149

Examples

```
#A simple model depending on two
```

```

#parameters and an additional
#"hyperparameter" x. Depending on
#x the model is sensitive to p[1] only (x=1)
#or p[2] only (x=0) or both (0<x<1)

example_model1<-function(p,x){
  return(p[1]*x+p[2]*(1-x))
}
paras<-fast_parameters(minimum=c(0,0,0),maximum=c(1,1,1))
paras
model_results <- apply(paras, 1, example_model1, x=0.5)
plot(model_results)
sensitivity <- sensitivity(x=model_results, numberf=3, make.plot=TRUE)
sensitivity

#In the second example, sensitivities are calculated for
#200 model results (which might be a time series).
#
#The model depends on 4 parameters
#
#It produces a weighted sum of the 4 parameters and returns this sum
#
#The weights depend on an additional parameter x=1:200

example_model2(p=c(1,3,1,1),fig=TRUE)
example_model2(p=c(1,2,2,3),fig=TRUE)
paras<-fast_parameters(min=c(0,0,0,0),max=c(1,2,2,3))
paras
model_results <- apply(paras, 1, example_model2)
plot(model_results)
dev.new()
sensitivity <- sensitivity_rep(data = model_results, xval=1:200, direction = 1, order=4 , numberf=4)
p.sensitivity(sen=sensitivity, xval=1:200, legend=names(paras))

```

double_serie

Double the length of a data series for the FAST algorithm

Description

This function is used internally for the FAST-algorithm. It duplicates the length of a data series assuming that the second part is a mirror image of the first part

Usage

```
double_serie(x)
```

Arguments

x x is the data series to make longer

Details

This function reverses the model output series from a number of model runs for the FAST analysis and appends it to the original series. The last element of the existing series is not duplicated during this process.

This is required in order to process the model run results for the FAST analysis with the `fft` function.

Value

If `x=c(1,2,3,4)` the returned vector is `c(1,2,3,4,3,2,1)`

Author(s)

Dominik Reusser

References

cukier 1978

Examples

```
x=c(1,2,3,4)
double_serie(x)
```

example_model2

Functions to demonstrate FAST

Description

Model 1 depends on a flexible number of parameters. It is documented in Saltelli and Sobol 1995 and Davis and Rabinowith 1984.

Model 2 depends on 4 parameters. It produces a weighted sum of the 4 parameters and returns this sum. The weights depend on an additional parameter `x=1:200`.

Usage

```
example_model1(par,a, output=c("model", "analytical sensitivities"))
example_model2(p, fig=FALSE)
```

Arguments

<code>par</code>	A vector with parameters
<code>a</code>	A vector with a values of the same length as <code>par</code> . <code>a</code> is a measure for the importance of each parameter, with highest importance for <code>a=0</code> , non important parameters for <code>a=9</code> and negligible parameters for <code>a=99</code>
<code>output</code>	A character indicating whether to return the model results or the analytical parameter sensitivities

p A vector of 4 parameters
fig boolean: Plot the model(x)

Value

A vector of the weighted sum of parameters.

Author(s)

Dominik Reusser

References

Saltelli, Andrea, and Ilya M Sobol. "About the use of rank transformation in sensitivity analysis of model output." *Reliability Engineering & System Safety* 50, no. 3 (1995): 225-239.

Davis, P. J. & Rabinowitz, P., *Methods of Numerical Integration*, 2nd edition, Academic Press, New York, 1984.

See Also

[fast, sensitivity_rep](#)

Examples

```
example_model1(par=c(0.5,0.5,0.5),a=c(1,1,1))  
  
#The model depends on 4 parameters  
#  
#It produces a weighted sum of the 4 parameters and returns this sum  
#  
#The weights depend on an additional parameter x=1:200  
example_model2(p=c(1,3,1,1),fig=TRUE)  
example_model2(p=c(1,2,2,3),fig=TRUE)
```

fast_parameters *Generate a parameter set for the FAST method*

Description

This function generates an array of parameters for the FAST method.

Usage

```
fast_parameters(minimum, maximum, names=paste(sep=" ", "P", 1:n),  
factor=1,logscale=rep(FALSE, n), cukier=TRUE, reorder=1:n)
```

Arguments

minimum	Vector of lower boundaries for parameters
maximum	Vector of upper boundaries for parameters
names	A vector of parameter names.
factor	Create more values then the minimum requires. Passed to s
logscale	A vector of booleans indicating whether a parameter is varied on a logarithmic scale. In this case, minimum and maximum are exponents
cukier	boolean. Indicates wheter to use freq_cukier or freq_mcrae82
reorder	A vector of indices that allows to use a different order for the parameters. This is important to check effects of the sampling scheme on results

Value

An array of dimension `c(min_number_of_runs, n)`

Author(s)

Dominik Reusser

See Also

[fast_parameters](#)

Examples

```
paras<-fast_parameters(minimum=c(0,0,0),maximum=c(1,2,2))
```

freq_cukier	<i>Calculate independent frequencies according to CUKIER1975 or McRae1982</i>
-------------	---

Description

This function returns a vector of independent frequencies for usage in the [fast](#) method.

Usage

```
freq_cukier(m, i = 1, omega_before = 0)
freq_mcrae82(m, i = 1, omega_before = 0)
```

Arguments

m	Number of frequencies (parameters) needed.
i	Used internally, recursion counter
omega_before	Used internally, previous frequency.

Value

A vector of independent frequencies to the order of 4.

Author(s)

Dominik Reusser

References

CUKIER, R. I.; SCHAIBLY, J. H. & SHULER, K. E. Study Of Sensitivity Of Coupled Reaction Systems To Uncertainties In Rate Coefficients .3. Analysis Of Approximations Journal Of Chemical Physics, 1975 , 63 , 1140-1149

McRae, G.; Tilden, J. & Seinfeld, J. Global sensitivity analysis - a computational implementation of the Fourier amplitude sensitivity test (FAST) Comput. Chem. Eng., 1982 , 6 , 15-25

See Also

[fast](#)

Examples

```
freq_cukier(5)
```

na2mean

Replace NA in a vector by mean of neighboring values.

Description

This method replaces NA-Values in a vector by the mean of the neighboring values. Example: `c(1,NA,3)` is converted to `c(1,2,3)`.

Usage

```
na2mean(x)
```

Arguments

x Vector to fill

Details

The method does not work if multiple NA are in sequence.

Value

Vector with NA replaced by mean.

Author(s)

Dominik Reusser

Examples

```
na2mean(c(1,NA,2))
na2mean(c(1,NA,NA,2))
```

p.sensitivity

Plot a vector (e.g. time series) of sensitivities

Description

This function plots the results from the [sensitivity_rep](#) function.

Usage

```
p.sensitivity(sen, xval, legend, legend.cex = 0.5, range =
              1:NROW(sen), col = 1:NROW(sen), lty = rep(1,
              NROW(sen)), smooth = rep(FALSE, NROW(range)), x.range
              = 1:length(xval), m.max = max(sen[range, ], na.rm =
              TRUE), limits = rep(FALSE, NROW(range)), xlab =
              "time", ylab = "Sensitivity", ...)
```

Arguments

sen	returned object from sensitivity_rep
xval	Data defining the position on the x-axis. Usually a vector of POSIX-Dates
legend	Entries for the legend text
legend.cex	Size of the legend
range	List of parameter-sensitivities to plot
col	Vector of colors for parameters
smooth	Vector of TRUE or FALSE values. Sensitivity series are smoothed using a LOWES smoother if TRUE.
lty	see par
xlab	see par
ylab	see par
x.range	Range of x-values to plot.
m.max	Maximum value for the y-axis
limits	Vector of limits for sensitivities. Plotted as ... If FALSE, the limit for the corresponding series is ignored
...	Parameters passed to the plot-function

Value

This value is used for its plot-function.

Author(s)

Dominik Reusser

See Also

[sensitivity_rep](#)

rerange	<i>Linear transformation of data</i>
---------	--------------------------------------

Description

The function performs a linear transformation of the data, such that afterwards $\text{range}(\text{data}) = c(\text{theMin}, \text{theMax})$.

Usage

```
rerange(data, min.goal = 0, max.goal = 1, min.data =  
        min(data, na.rm=na.rm), max.data = max(data, na.rm=na.rm),  
        center = NA, na.rm=FALSE)
```

Arguments

data	vector with the data to transform
min.goal	new minimum value
max.goal	new maximum value
min.data	old minimum value
max.data	old maximum value
center	which old value should become the new center ((max.goal + min.goal) / 2)
na.rm	boolean, passed to min and max

Value

vector with the transformed data

Author(s)

Dominik Reusser

Examples

```
rerange(data=1:20)  
rerange(data=1:30, center=5)
```

S*Generate S-set for FAST method*

Description

This function generates an array of values which provide the base for parameters for the FAST method. It is usually not used directly but called from [fast_parameters](#)

Usage

```
S(m, factor = 1, cukier = TRUE, par.names = NULL, reorder = 1:m)
```

Arguments

<code>m</code>	Number of parameters/frequencies
<code>factor</code>	Create more values than the minimum requires. Passed to s
<code>cukier</code>	boolean. Indicates whether to use freq_cukier or freq_mcrae82
<code>par.names</code>	A vector of parameter names.
<code>reorder</code>	A vector of indices that allows to use a different order for the parameters. This is important to check effects of the sampling scheme on results

Value

An array of dimension `c(min_number_of_runs, n)`

Author(s)

Dominik Reusser

See Also

[fast_parameters](#)

Examples

```
S(3)
```

s*Generate an s-set for FAST-method*

Description

Generates a number of equally spaced values between $-\pi/2$ and $\pi/2$. The number is determined by the number of runs required for the FAST method for a number of runs.

Usage

```
s(m, factor = 1, cukier = TRUE)
```

Arguments

m	number of frequencies/parameters
factor	the length of the returned vector is the minimum number required for FAST time factor
cukier	boolean. Indicates wheter to use freq_cukier or freq_mcrae82

Value

A vector of equally spaced values between $-\pi/2$ and $\pi/2$

Author(s)

Dominik Reusser

See Also

[fast](#)

Examples

```
s(4)
diff(s(3))
```

sa *Calculate sensitivities*

Description

sa provides a general interface to sensitivity analysis, similar to [optim](#).

Usage

```
sa(par, fn, method = c("FAST"), ..., xval = NULL)
```

Arguments

par	A matrix with two columns. The first giving the lower, the second giving the upper bound to the parameters of fn.
fn	Function for which to analyse sensitivities, with first argument the vector of parameters for which to analyze sensitivities.
method	Currently, only fast is available.
...	Additional arguments passed to fn
xval	Values for the x-axis if fn returns a vector.

Details

Let me know what details you would like to know (email to the maintainer)!

Value

A vector of sensitivities

Author(s)

Dominik Reusser

References

Reusser, Dominik E., Wouter Buytaert, and Erwin Zehe. "Temporal dynamics of model parameter sensitivity for computationally expensive models with FAST (Fourier Amplitude Sensitivity Test)." *Water Resources Research* 47 (2011): W07551.

See Also

See Also as [sensitivity](#), [sensitivity_rep](#) for the internally used functions.

Examples

```
#Does the same as the example in sensitivity_rep but with less code  
sa(par=matrix(c(0,0,0,0, 1,2,2,3), ncol=2), fn=example_model2)
```

sensitivity	<i>Calculate sensitivity according to the FAST algorithm</i>
-------------	--

Description

sensitivity calculates the sensitivity from a series of model outputs (x) according to the FAST algorithm.

Usage

```
sensitivity(x, numberf, order = 4, make.plot = FALSE, show.legend
           = TRUE, plot.max = max(ff[-1]), include.total.variance
           = FALSE, cukier = TRUE, names = paste(sep = "", "P",
           1:numberf), main = "", xlab = "frequency", ylab =
           "Fourier Coef", pch = rep(0, numberf), col =
           (1:numberf) + 1, reorder = 1:numberf, ...)
```

Arguments

x	A vector of model outputs where parameters vary between runs according to the fast algorithm.
numberf	Number of parameters varied.
order	Order of parameter frequency independence (see Cukier)
make.plot	plot the Fourier spectrum?
plot.max	xmax in the spectrum
include.total.variance	include the sum of all variances in the result list.
pch	see par
col	see par
...	Additional parameters passed to plot
xlab	see par
ylab	see par
main	Title for the plot
show.legend	Boolean indicating whether to plot the legend
names	A vector of parameter names.
cukier	boolean: Calculate FAST-parameters according to Cukier 1975 or McRae 1982
reorder	A vector of indices that allows to use a different order for the parameters. This is important to check effects of the sampling scheme on results. Use the same as for fast_parameters

Value

A list of the partial variance accounted for by each parameter.

Author(s)

Dominik Reusser

References

Reusser, Dominik E., Wouter Buytaert, and Erwin Zehe. "Temporal dynamics of model parameter sensitivity for computationally expensive models with FAST (Fourier Amplitude Sensitivity Test)." *Water Resources Research* 47 (2011): W07551.

CUKIER, R. I.; SCHAIBLY, J. H. & SHULER, K. E. Study Of Sensitivity Of Coupled Reaction Systems To Uncertainties In Rate Coefficients .3. Analysis Of Approximations *Journal Of Chemical Physics*, 1975 , 63 , 1140-1149

McRae, G.; Tilden, J. & Seinfeld, J. Global sensitivity analysis - a computational implementation of the Fourier amplitude sensitivity test (FAST) *Comput. Chem. Eng.*, 1982 , 6 , 15-25

See Also

[S](#), [fast](#)

Examples

```
example_model1<-function(p,x){
  return(p[1]*x+p[2]*(1-x))
}
paras<-fast_parameters(min=c(0,0,0),max=c(1,2,2))
paras
model_results <- apply(paras, 1, example_model1, x=0.5)
model_results
sensitivity <- sensitivity(x=model_results, numberf=3, make.plot=TRUE)
sensitivity
```

sensitivity_rep

Repeat calculation of sensitivities for lots of data

Description

This function calculates the sensitivity for a series of data, e.g. a time series.

Usage

```
sensitivity_rep(data.zoo, xval = index(data.zoo), direction=1,
  data = coredata(data.zoo), numberf,
  order=4, legend = paste("P", 1:order, sep = "" ),
  cukier=TRUE, reorder = 1:numberf, ...)
```

Arguments

data	Array of data to use of the form todo
data.zoo	Zoo object containing data with one row per parameter set
xval	Vector to use as x data for plotting
direction	Dimension which contains the todo
numberf	Number of parameters used
order	Order of parameter frequency independence (see Cukier1975)
legend	legend text to plot
cukier	boolean. Indicates wheter to use freq_cukier
reorder	A vector of indices that allows to use a different order for the parameters. This is important to check effects of the sampling scheme on results. Use the same as for fast_parameters
...	Arguments passed to p.sensitivity

Value

An array of sensitivities of the form ...

Author(s)

Dominik Reusser

See Also

[fast](#), [sensitivity](#)

Examples

```
#The model depends on 4 parameters
#
#It produces a weighted sum of the 4 parameters and returns this sum
#
#The weights depend on an additional parameter x=1:200
example_model2(p=c(1,3,1,1),fig=TRUE)
example_model2(p=c(1,2,2,3),fig=TRUE)
paras<-fast_parameters(min=c(0,0,0,0),max=c(1,2,2,3))
paras
model_results <- apply(paras, 1, example_model2)
model_results
sensitivity <- sensitivity_rep(data = model_results, xval=1:200, direction = 1, order=4 , numberf=4)
p.sensitivity(sen=sensitivity, xval=1:200, legend=names(paras))

#Alternatively with zoo object
result.zoo <- zoo(model_results, order.by=1:200)
sensitivity <- sensitivity_rep(data.zoo = result.zoo, numberf=4)
```

Index

*Topic **\textasciitildesensitivity**

sa, [12](#)

*Topic **datagen**

example_model2, [4](#)

fast_parameters, [5](#)

freq_cukier, [6](#)

*Topic **hplot**

p.sensitivity, [8](#)

*Topic **misc**

fast-package, [2](#)

*Topic **multivariate**

fast-package, [2](#)

*Topic **package**

fast-package, [2](#)

*Topic **utilities**

double_serie, [3](#)

example_model2, [4](#)

fast-package, [2](#)

fast_parameters, [5](#)

freq_cukier, [6](#)

na2mean, [7](#)

p.sensitivity, [8](#)

rerange, [9](#)

S, [10](#)

s, [11](#)

sensitivity, [13](#)

sensitivity_rep, [14](#)

*Topic **util**

sa, [12](#)

d_m_mcrae82 (freq_cukier), [6](#)

double_serie, [3](#)

example_model1 (example_model2), [4](#)

example_model2, [4](#)

fast, [5-7](#), [11](#), [14](#), [15](#)

fast (fast-package), [2](#)

fast-package, [2](#)

fast_parameters, [2](#), [5](#), [6](#), [10](#), [13](#), [15](#)

fft, [4](#)

freq_cukier, [6](#), [6](#), [10](#), [11](#), [15](#)

freq_mcrae82, [6](#), [10](#), [11](#)

freq_mcrae82 (freq_cukier), [6](#)

max, [9](#)

min, [9](#)

min_runs_cukier75 (freq_cukier), [6](#)

min_runs_mcrae82 (freq_cukier), [6](#)

na2mean, [7](#)

omega_m_cukier75 (freq_cukier), [6](#)

omega_m_mcrae82 (freq_cukier), [6](#)

optim, [12](#)

p.sensitivity, [8](#)

par, [8](#), [13](#)

rerange, [9](#)

S, [10](#), [14](#)

s, [6](#), [10](#), [11](#)

sa, [12](#)

sensitivity, [2](#), [12](#), [13](#), [15](#)

sensitivity_rep, [5](#), [8](#), [9](#), [12](#), [14](#)