Package 'spm'

December 11, 2019

Title SPM extract package
Version 1.1
Date 2019-12-09
Author A. Dunn
Description A set of R functions for extracting from SPM output files.
Maintainer A Dunn <alistair.dunn@oceanenvironmental.co.nz></alistair.dunn@oceanenvironmental.co.nz>
License CPL v1.0. See the SPM User Manual for license details.

URL http://www.niwa.co.nz

Copyright National Institute of Water & Atmospheric Research (NIWA), New Zealand Ministry for Primary Industries.

R topics documented:

spm-package	2
calc.abundance	3
extract	3
extract.ageingerror	4
extract.agesize	4
extract.ageweight	5
extract.covariance	5
extract.derivedquantity	6
extract.derivedquantitybycell	
extract.estimatesummary	7
extract.estimatevalue	
extract.initialisationphase	
extract.layer	
extract.layerderivedview	
extract.MCMC	
extract.objectivefunction	
extract.observation	
extract.partition	11
extract.partitionbiomass	
extract.preferencefunction	12
extract.process	
extract.randomnumberseed	
extract.selectivity	
extract.simulations	14

2 spm-package

Index		27
	spm.zeroFun	26
	spm.unpaste	26
	spm.string.to.vector.of.words	25
	spm.string.to.vector.of.numbers	25
	spm.report.types	25
	spm.remove.last.words	24
	spm.remove.first.words	24
	spm.recodevector	24
	spm.pos.match	23
	spm.pos	23
	spm.make.table	23
	spm.make.list	22
	spm.make.filename	22
	spm.isin	22
	spm.is.whole.number	21
	spm.get.lines	21
	spm.extract.simulated.observation	21
	spm.dplot	20
	spm.convert.to.lines	20
	spm.area	20
	PFthreshold	19
	PFnormal	19
	PFlogistic	18
	PFinverselogistic	18
	PFexponential	17
	PFconstant	17
	PF	16 16
	generate.MVU	15
	extract.spatialmap	15

Description

A set of R functions for extracting and processing SPM output.

Details

Package: spm Date: 2013-02-12

License: See the SPM manual, or use 'spm -1' at the command line to view the SPM license.

URL: http://www.niwa.co.nz

Copyright: National Institute of Water & Atmospheric Research (NIWA).

Index:

calc.abundance 3

extract elements from an SPM output file

spm-package SPM package

Author(s)

A. Dunn

Maintainer: A Dunn <a.dunn@niwa.co.nz>

calc.abundance Generate a multivariate uniform distribution based on the bounds for

the extimated parameters

Description

Generate a multivariate uniform distribution based on the bounds for the extimated parameters

Usage

```
calc.abundance(data, categories = NULL, ages = NULL, total = T)
```

Arguments

data data representing a partition categories

Optional subset of categories

Optional subset of ages

total If true, then sum over selected categories and ages. Default = TRUE

Author(s)

Alistair Dunn

extract SPM output into R

Description

Extract SPM output into R

Usage

```
extract(file, path = "", ignore.unknown = FALSE)
```

Arguments

file the name of the input file containing model output to extract

path Optionally, the path to the file

ignore.unknown Ignore unknown objects in the output file

Author(s)

4 extract.agesize

```
extract.ageingerror Helper function for 'extract'
```

Description

Helper function for 'extract'

Usage

```
extract.ageingerror(lines)
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.agesize

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.agesize(lines)
```

Arguments

lines

list of lines to process

Author(s)

extract.ageweight 5

extract.ageweight

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.ageweight(lines)
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.covariance

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.covariance(lines)
```

Arguments

lines

list of lines to process

Author(s)

extract.derivedquantity

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

extract.derivedquantity(lines)

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.derivedquantitybycell

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

extract.derivedquantitybycell(lines)

Arguments

lines

list of lines to process

Author(s)

extract.estimatesummary 7

```
extract.estimatesummary
```

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.estimatesummary(lines)
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.estimatevalue Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.estimatevalue(lines)
```

Arguments

lines

list of lines to process

Author(s)

8 extract.layer

```
{\tt extract.initialisation phase}
```

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.initialisationphase(lines)
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.layer

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.layer(lines)
```

Arguments

lines

list of lines to process

Author(s)

extract.layerderivedview

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.layerderivedview(lines)
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.MCMC

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.MCMC(lines)
```

Arguments

lines

list of lines to process

Author(s)

10 extract.observation

```
extract.objectivefunction
```

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
\verb|extract.objectivefunction(lines)|\\
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.observation

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.observation(lines)
```

Arguments

lines

list of lines to process

Author(s)

extract.partition 11

extract.partition

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.partition(lines)
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.partitionbiomass

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.partitionbiomass(lines)
```

Arguments

lines

list of lines to process

Author(s)

12 extract.process

```
extract.preferencefunction
```

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.preferencefunction(lines)
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.process

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.process(lines)
```

Arguments

lines

list of lines to process

Author(s)

extract.randomnumberseed 13

extract.randomnumberseed

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.randomnumberseed(lines)
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.selectivity Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.selectivity(lines)
```

Arguments

lines

list of lines to process

Author(s)

14 extract.sizeweight

extract.simulations

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.simulations(file, path = "")
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

extract.sizeweight

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.sizeweight(lines)
```

Arguments

lines

list of lines to process

Author(s)

extract.spatialmap 15

extract.spatialmap

Helper function for 'extract'

Description

Helper function for 'extract'

Usage

```
extract.spatialmap(lines)
```

Arguments

lines

list of lines to process

Author(s)

Alistair Dunn

generate.MVU

Generate a multivariate uniform distribution based on the bounds for the extimated parameters

Description

Generate a multivariate uniform distribution based on the bounds for the extimated parameters

Usage

```
generate.MVU(file, path = "", output.file, sample.size = 1)
```

Arguments

file the name of the input file containing the estimated fits

path Optionally, the path to the file

output.file The name of the output file to write randomly generated values

sample.size The number f samples to generate

Author(s)

Sophie Mormede

16 PFconstant

PF

Evaluate a preference function

Description

Evaluate a preference function

Usage

```
PF(type = "none", x, alpha, ..., rescale = T)
```

Arguments

type A type of preference function. Valid values include constant", "doublenormal",

"exponential", "inverselogistic", "logistic", "normal", and "threshold"

x x-values over which to evaluate. Default = 1
 alpha The value of the alpha parameter. Default = 1
 the parameters of the preference functions

rescale Rescale the function to have value 1. Default = TRUE

Author(s)

Alistair Dunn

PFconstant

Evaluate the constant preference function

Description

Evaluate the constant preference function

Usage

```
PFconstant(x, alpha, rescale = T)
```

Arguments

x x-values over which to evaluate. Default = 1alpha The value of the alpha parameter. Default = 1

rescale Rescale the function to have value 1?. Default = TRUE

Author(s)

PFdoublenormal 17

PFdoublenormal	Evaluate the double normal preference function

Description

Evaluate the double normal preference function

Usage

```
PFdoublenormal(x, alpha, mu, sigmaL, sigmaR, rescale = T)
```

Arguments

X	x-values over which to evaluate. Default = 1
alpha	The value of the alpha parameter. Default = 1

mu the mean of the double normal

sigmaL the standard deviation of the left-hand side of the double normal sigmaR the standard deviation of the right-hand side of the double normal

rescale Rescale the function to have value 1?. Default = TRUE

Author(s)

Alistair Dunn

PFexponential	Evaluate the exponential preference function
	,

Description

Evaluate the exponential preference function

Usage

```
PFexponential(x, alpha, lambda, rescale = T)
```

Arguments

X	x-values over which to evaluate. Default = 1
alpha	The value of the alpha parameter. Default = 1

lambda the rate of the exponential

rescale Rescale the function to have value 1?. Default = TRUE

Author(s)

PFlogistic PFlogistic

Р	Εi	nv	orc	elo	noi	ct.	ic	
ГΙ	1 Т	HV	-12	$e^{\pm i}$	フとエ	SL.	L C.	

Evaluate the inverse-logistic preference function

Description

Evaluate the inverse-logistic preference function

Usage

```
PFinverselogistic(x, alpha, a50, ato95, rescale = T)
```

Arguments

Χ	x-values over which to evaluate. Default = 1
alpha	The value of the alpha parameter. Default = 1
a50	the a50 value of the inverse logisitic
ato95	the ato95 value of the inverse logisitic
rescale	Rescale the function to have value 1?. Default = TRUE

Author(s)

Alistair Dunn

PFlogistic	Evaluate the	logistic	nreference	function
LI TOSTOCIC	Evaluate the	weish	prejerence	junction

Description

Evaluate the logistic preference function

Usage

```
PFlogistic(x, alpha, a50, ato95, rescale = T)
```

Arguments

X	x-values over which to evaluate. Default = 1
alpha	The value of the alpha parameter. Default = 1

a50 the a50 value of the logisitic ato95 the ato95 value of the logisitic

rescale Rescale the function to have value 1?. Default = TRUE

Author(s)

PFnormal 19

PFnormal E	valuate the normal preference function
------------	--

Description

Evaluate the normal preference function

Usage

```
PFnormal(x, alpha, mu, sigma, rescale = T)
```

Arguments

x x-values over which to evaluate. Default = 1 alpha The value of the alpha parameter. Default = 1

mu the mean of the normal

sigma the standard deviation of the normal

rescale Rescale the function to have value 1?. Default = TRUE

Author(s)

Alistair Dunn

PFthreshold	Evaluate the threshold preference function

Description

Evaluate the threshold preference function

Usage

```
PFthreshold(x, alpha, N, lambda, rescale = T)
```

Arguments

x x-values over which to evaluate. Default = 1 alpha The value of the alpha parameter. Default = 1

N the threshhold value

rescale Rescale the function to have value 1?. Default = TRUE

Author(s)

20 spm.dplot

spm.area

utility function

Description

utility function

Usage

```
spm.area(corners)
```

Author(s)

Alistair Dunn

```
spm.convert.to.lines utility function
```

Description

utility function

Usage

```
spm.convert.to.lines(filename)
```

Author(s)

Alistair Dunn

spm.dplot

utility function

Description

utility function

Usage

```
spm.dplot(..., name = T, quantiles = c(0.5), plot.mean = F, main = "", xlab = "", ylab = "", ylim, srtx = 0, bw = "nrd0", adjust = 1/3, adj = 0, fill = F, gap = 0.2)
```

Author(s)

```
{\it spm.extract.simulated.observation} \\ {\it utility function}
```

Description

utility function

Usage

```
spm.extract.simulated.observation(lines)
```

Author(s)

Alistair Dunn

spm.get.lines

utility function

Description

utility function

Usage

```
spm.get.lines(lines, from = -1, to = -1, contains = "",
    starts.with = "", clip.to = "", clip.from = "",
    clip.to.match = "", clip.from.match = "", ...)
```

Author(s)

Alistair Dunn

```
spm.is.whole.number utility function
```

Description

utility function

Usage

```
spm.is.whole.number(string)
```

Author(s)

22 spm.make.list

 ${\tt spm.isin}$

utility function

Description

utility function

Usage

```
spm.isin(x, y)
```

Author(s)

Alistair Dunn

 ${\tt spm.make.filename}$

utility function

Description

utility function

Usage

```
spm.make.filename(file, path = "")
```

Author(s)

Alistair Dunn

 ${\tt spm.make.list}$

utility function

Description

utility function

Usage

```
spm.make.list(lines)
```

Author(s)

spm.make.table 23

spm.make.table

utility function

Description

utility function

Usage

```
spm.make.table(lines)
```

Author(s)

Alistair Dunn

spm.pos

utility function

Description

utility function

Usage

```
spm.pos(vector, x)
```

Author(s)

Alistair Dunn

spm.pos.match

utility function

Description

utility function

Usage

```
spm.pos.match(vector, regexp)
```

Author(s)

```
spm.recodevector
```

utility function

Description

utility function

Usage

```
spm.recodevector(in.data, from.vals, to.vals)
```

Author(s)

Alistair Dunn

```
spm.remove.first.words
```

utility function

Description

utility function

Usage

```
spm.remove.first.words(string, words = 1)
```

Author(s)

Alistair Dunn

```
spm.remove.last.words utility function
```

Description

utility function

Usage

```
spm.remove.last.words(string, words = 1)
```

Author(s)

spm.report.types 25

spm.report.types

utility function

Description

utility function

Usage

```
spm.report.types()
```

Author(s)

Alistair Dunn

Description

utility function

Usage

```
spm.string.to.vector.of.numbers(string, sep = " ")
```

Author(s)

Alistair Dunn

```
{\it spm.string.to.vector.of.words} \\ {\it utility function}
```

Description

utility function

Usage

```
spm.string.to.vector.of.words(string, sep = " ")
```

Author(s)

26 spm.zeroFun

spm.unpaste

utility function

Description

utility function

Usage

```
spm.unpaste(string, sep = " ")
```

Author(s)

Alistair Dunn

spm.zeroFun

utility function

Description

utility function

Usage

```
spm.zeroFun(x, delta = 1e-11)
```

Author(s)

Index

```
*Topic package
    spm-package, 2
calc.abundance, 3
extract, 3
extract.ageingerror, 4
extract.agesize, 4
extract.ageweight, 5
extract.covariance, 5
extract.derivedquantity, 6
extract.derivedquantitybycell, 6
extract.estimatesummary, 7
extract.estimatevalue, 7
extract.initialisationphase, 8
extract.layer, 8
extract.layerderivedview, 9
extract.MCMC, 9
extract.objectivefunction, 10
extract.observation, 10
extract.partition, 11
extract.partitionbiomass, 11
extract.preferencefunction, 12
extract.process, 12
extract.randomnumberseed, 13
extract.selectivity, 13
extract.simulations, 14
extract.sizeweight, 14
extract.spatialmap, 15
generate. MVU, 15
PF, 16
PFconstant, 16
PFdoublenormal, 17
PFexponential, 17
PFinverselogistic, 18
PFlogistic, 18
PFnormal, 19
PFthreshold, 19
spm (spm-package), 2
spm-package, 2
spm.area, 20
spm.convert.to.lines, 20
```

```
spm.dplot, 20
spm.extract.simulated.observation, 21
spm.get.lines, 21
spm.is.whole.number, 21
spm.isin, 22
spm.make.filename, 22
spm.make.list, 22
spm.make.table, 23
spm.pos, 23
spm.pos.match, 23
spm.recodevector, 24
spm.remove.first.words, 24
spm.remove.last.words, 24
spm.report.types, 25
spm.string.to.vector.of.numbers, 25
spm.string.to.vector.of.words, 25
spm.unpaste, 26
spm.zeroFun, 26
```