

Package ‘casal2’

August 9, 2017

Title casal2 extract package

Version 1.0

Date 2017-03-20

Author D. Fu and C. Marsh

Description A set of R functions for extracting and plotting from casal2 output files.

Maintainer Casal2 development team <casal2@niwa.co.nz>

License CPL v1.0. See the CASAL2 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:

check_short_hand	2
convert.to.lines	3
CV.for.CPUE	3
DoubleExponential	4
DoubleNormal	4
evalit	5
extract.csl2.file	5
extract.mcmc	6
extract.mpd	6
extract.parameters	7
get.casal2_list	7
get.line.label	8
get.line.type	8
get.lines	8
Increasing	9
InverseLogistic	9
is.all.numeric	10
is.even	10
is.in	11
is.odd	11
KnifeEdge	11
Logistic	12
LogisticProducing	13
make.complete_vector	13

make.data.frame	14
make.list	14
make.list_element	14
make.matrix	15
make.named_complete_vector	15
make.string_vector	15
make.vector	16
Method.TA1.10	16
Method.TA1.8	17
mpd_derived_quantity	18
param.profile	18
param.profile.by.cohort	19
Paste	20
plot.derived_quantities	20
pos	21
pos.match	21
pow	21
ReadSimulatedData	22
reformat.compositional.data	22
regexp.in	23
Regexpr	23
remove.first.words	23
string.to.vector.of.numbers	24
string.to.vector.of.words	24
strip	24
Sum	25
summarise_process	25
summarise_warnings_encountered	25
summary.default	26
unpaste	26
write.csl2.file	27

Index	28
--------------	-----------

check_short_hand	<i>Utility extract function</i>
------------------	---------------------------------

Description

Utility extract function

Usage

check_short_hand(x)

Author(s)

Craig Marsh

convert.to.lines	<i>Utility extract function</i>
------------------	---------------------------------

Description

Utility extract function

Usage

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

Author(s)

Dan Fu

CV.for.CPUE	<i>CV.for.CPUE</i>
-------------	--------------------

Description

This function is useful for deciding on a c.v. to be used with a CPUE series in a stock assessment model. Originally written in Chris Francis's DataWeighting Package, this has been copied over and modified so that users can use this functionality with Casal2 models/output.

Usage

```
CV.for.CPUE(year, cpue, f, plot.it = TRUE)
```

Arguments

year	vector of years with CPUE indices
cpue	CPUE indices
f	degree of lowess smoothing (0 = no smoothing, 1 = maximum smoothing)
plot.it	If TRUE, plot the index and the smoothed fit. Otherwise, return a dataframe of the year, index, smoothed fitted value, and cv)

Value

The function either plots the CPUE, together with a lowess line fitted to it, and returns the c.v. of the residuals to the fit. Or returns a dataframe of the lowess line fits and associated c.v.s for each point.

Author(s)

Chris Francis

DoubleExponential	<i>Calculate the Double-Exponential Selectivity used in Casal2</i>
-------------------	--

Description

This function calculates the Double-Exponential selectivity and should be used to plot up the form of the selectivity when estimating the parameters in the model.

Usage

```
DoubleExponential(age_or_length, x0, x1, x2, y0, y1, y2, alpha = 1)
```

Arguments

age_or_length	"vector<numeric>" A vector of ages and or lengths to calculate the selectivity over.
x0	"numeric" See manual for formal definition of these parameters
x1	"numeric"
x2	"numeric"
y0	"numeric"
y1	"numeric"
y2	"numeric"
alpha	"numeric" The capping parameter of the selectivity, to move the max away from one

Value

"vector<numeric>" selectivity values over the age/length range and parameters supplied

Author(s)

Craig Marsh

DoubleNormal	<i>Calculate the Double-Normal Selectivity used in Casal2</i>
--------------	---

Description

This function calculates the Double-Normal selectivity and should be used to plot up the form of the selectivity when estimating the parameters in the model.

Usage

```
DoubleNormal(age_or_length, alpha = 1, mu, sigma_l, sigma_r)
```

Arguments

age_or_length	"vector<numeric>" A vector of ages and or lengths to calculate the selectivity over.
alpha	"numeric" The capping parameter of the selectivity, to move the max away from one
mu	"numeric" Mean of the selectivity
sigma_l	"numeric" left hand standard deviation of the selectivity
sigma_r	"numeric" Right hand standard deviation of the selectivity

Value

"vector<numeric>" selectivity values over the age/length range and parameters supplied

Author(s)

Craig Marsh

evalit	<i>Utility plot function</i>
--------	------------------------------

Description

Utility plot function

Usage

```
evalit(x)
```

Author(s)

Craig Marsh

extract.csl2.file	<i>Model configuration write function</i>
-------------------	---

Description

This function reads a Casal2 configuration file and returns a list object in R. Where each element is a command and subcommand from the configuration file

Usage

```
extract.csl2.file(file, path = "")
```

Arguments

file	the name of the input file containing model configuration
path	Optionally, the path to the file

Author(s)

Craig Marsh

extract.mcmc

*extract.mcmc function for casal2 output***Description**

An extract function that reads objective and sample output that are produced from a 'casal2 -m' model run. This function also create a 'casal2.mcmc' class which can be used in plotting and summary functions.

Usage

```
extract.mcmc(samples.file = "mcmc_samples.out.0",
             objectives.file = "mcmc_objectives.out.0", path = "",
             return_covariance = F)
```

Arguments

samples.file <string> the name of the input file containing the samples.file output by casal2
 objectives.file <string> the name of the input file containing the objectives.file output by casal2
 path Optional<string>, the path to the file
 return_covariance Optional<bool>, Whether you want to extract the covariance matrix with the mcmc object?

Value

a 'casal2MCMC' that can be integrated using the str() function.

Author(s)

C. Marsh

extract.mpd

*extract MPD function for readin in Casal2 output that has been generated from a -r, -e, -f, -p run mode.***Description**

An extract function that reads Casal2 output that are produced from a '-r' or '-e' or '-f' or '-p' model run. This function also create a 'casal2.mpd' class which can be used in plotting and summary functions. See the casal2 manual for more information.

Usage

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

Arguments

file	the name of the input file containing model output to extract
path	Optionally, the path to the file

Value

a 'casal2MPD' object which is essentially a list, that can be integrated using the str() function.

Author(s)

Dan Fu

extract.parameters	<i>Utility extract.parameters function</i>
--------------------	--

Description

This function reads in a parameter file that would be generated using the -o syntax.

Usage

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

Arguments

file	the name of the input file containing model output to extract
path	Optionally, the path to the file

Value

Data <"data.frame"> of parameters that are from a -i format.

Author(s)

Craig Marsh

get.casal2_list	<i>Utility function</i>
-----------------	-------------------------

Description

Utility function

Usage

```
get.casal2_list()
```

Author(s)

Craig Marsh

get.line.label	<i>Utility extract function</i>
----------------	---------------------------------

Description

Utility extract function

Usage

```
get.line.label(line)
```

Author(s)

Dan Fu

get.line.type	<i>Utility extract function</i>
---------------	---------------------------------

Description

Utility extract function

Usage

```
get.line.type(line)
```

Author(s)

Dan Fu

get.lines	<i>Utility extract function</i>
-----------	---------------------------------

Description

Utility extract function

Usage

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

Author(s)

Dan Fu

Increasing

Calculate the Increasing Selectivity used in Casal2

Description

This function calculates the Increasing selectivity and should be used to plot up the form of the selectivity when estimating the parameters in the model.

Usage

```
Increasing(age_or_length, alpha = 1, v, low, high)
```

Arguments

age_or_length	"vector<numeric>" A vector of ages and or lengths to calculate the selectivity over.
alpha	"numeric" The capping parameter of the selectivity, to move the max away from one
v	"vector<numeric>" A vector of selectivities, must have a one for one relationship with age_or_length.
low	"numeric" age or length at which selectivity is set = 0 below
high	"numeric" age or length at which selectivity is set = alpha above

Value

"vector<numeric>" selectivity values over the age/length range and parameters supplied

Author(s)

Craig Marsh

InverseLogistic

Calculate the Inverse-Logistic Selectivity used in Casal2

Description

This function calculates the Inverse-Logistic selectivity and should be used to plot up the form of the selectivity when estimating the parameters in the model.

Usage

```
InverseLogistic(age_or_length, alpha = 1, a50, ato95)
```

Arguments

age_or_length	"vector<numeric>" A vector of ages and or lengths to calculate the selectivity over.
alpha	"numeric" The capping parameter of the selectivity, to move the max away from one
a50	"numeric" age or length where selectivity 50%
ato95	"numeric" age or length difference where selectivity goes from 50% - 95% selective

Value

"vector<numeric>" selectivity values over the age/length range and parameters supplied

Author(s)

Craig Marsh

is.all.numeric	<i>Utility extract function</i>
----------------	---------------------------------

Description

Utility extract function

Usage

```
is.all.numeric(x, what = c("test", "vector"), extras = c(".", "NA", "na",
  "null", "NULL"))
```

Author(s)

Dan Fu (not really)

is.even	<i>Utility extract function</i>
---------	---------------------------------

Description

Utility extract function

Usage

```
is.even(x)
```

Author(s)

Dan Fu

is.in	<i>Utility extract function</i>
-------	---------------------------------

Description

Utility extract function

Usage

```
is.in(x, y)
```

Author(s)

Dan Fu

is.odd	<i>Utility extract function</i>
--------	---------------------------------

Description

Utility extract function

Usage

```
is.odd(x)
```

Author(s)

Dan Fu

KnifeEdge	<i>Calculate the KnifeEdge Selectivity used in Casal2</i>
-----------	---

Description

This function calculates the KnifeEdge selectivity and should be used to plot up the form of the selectivity when estimating the parameters in the model.

Usage

```
KnifeEdge(age_or_length, alpha = 1, Edge)
```

Arguments

age_or_length	"vector<numeric>" A vector of ages and or lengths to calculate the selectivity over.
alpha	"numeric" The capping parameter of the selectivity, to move the max away from one
edge	"numeric" age or length at which selectivity is 0 to the left or alpha to the right

Value

"vector<numeric>" selectivity values over the age/length range and parameters supplied

Author(s)

Craig Marsh

Logistic

Calculate the Logistic Selectivity used in Casal2

Description

This function calculates the Logistic selectivity and should be used to plot up the form of the selectivity when estimating the parameters in the model.

Usage

```
Logistic(age_or_length, alpha = 1, a50, ato95)
```

Arguments

age_or_length	"vector<numeric>" A vector of ages and or lengths to calculate the selectivity over.
alpha	"numeric" The capping parameter of the selectivity, to move the max away from one
a50	"numeric" age or length where selectivity 50%
ato95	"numeric" age or length difference where selectivity goes from 50% - 95% selective

Value

"vector<numeric>" selectivity values over the age/length range and parameters supplied

Author(s)

Craig Marsh

LogisticProducing	<i>Calculate the Logistic-Producing Selectivity used in Casal2</i>
-------------------	--

Description

This function calculates the Logistic-Producing selectivity and should be used to plot up the form of the selectivity when estimating the parameters in the model.

Usage

```
LogisticProducing(age_or_length, alpha = 1, a50, ato95, high, low)
```

Arguments

age_or_length	"vector<numeric>" A vector of ages and or lengths to calculate the selectivity over.
alpha	"numeric" The capping parameter of the selectivity, to move the max away from one
a50	"numeric" age or length where selectivity 50%
ato95	"numeric" age or length difference where selectivity goes from 50% - 95% selective
high	"numeric" age or length at which selectivity is set = alpha above
low	"numeric" age or length at which selectivity is set = 0 below

Value

"vector<numeric>" selectivity values over the age/length range and parameters supplied

Author(s)

Craig Marsh

make.complete_vector	<i>Utility extract function</i>
----------------------	---------------------------------

Description

Utility extract function

Usage

```
make.complete_vector(lines)
```

Author(s)

Dan Fu

make.data.frame	<i>Utility extract function</i>
-----------------	---------------------------------

Description

Utility extract function

Usage

```
make.data.frame(lines)
```

Author(s)

Dan Fu

make.list	<i>Utility extract function</i>
-----------	---------------------------------

Description

Utility extract function

Usage

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

Author(s)

Dan Fu

make.list_element	<i>Utility extract function</i>
-------------------	---------------------------------

Description

Utility extract function

Usage

```
make.list_element(lines)
```

Author(s)

Dan Fu

make.matrix	<i>Utility extract function</i>
-------------	---------------------------------

Description

create a matrix, does not expect header values.

Usage

```
make.matrix(lines)
```

Author(s)

Dan Fu

make.named_complete_vector	<i>Utility extract function</i>
----------------------------	---------------------------------

Description

Utility extract function

Usage

```
make.named_complete_vector(lines)
```

Author(s)

Dan Fu

make.string_vector	<i>Utility extract function</i>
--------------------	---------------------------------

Description

Utility extract function

Usage

```
make.string_vector(lines)
```

Author(s)

C Marsh

make.vector	<i>Utility extract function</i>
-------------	---------------------------------

Description

Utility extract function

Usage

```
make.vector(lines)
```

Author(s)

Dan Fu

Method.TA1.10	<i>Method.TA1.10</i>
---------------	----------------------

Description

This function is useful for deciding on the data weights of one or more at-age or at-length data sets with assumed lognormal error structure in a stock assessment. Originally written in Chris Francis's DataWeighting Package, this has been copied over and modified so that users can use this functionality with Casal2 models/output.

Usage

```
Method.TA1.10(model, Observation_label, pe.cv.stg1, plotit = F, xlim = NULL,
  ylim = NULL, trial.pe.cvs = seq(0, 2, 0.05))
```

Arguments

model	Casal2 output that is the result of a -r, -e run.
Observation_label	Label of the observation you want to apply the weighting too.
pe.cv.stg1	process-error c.v. assumed or estimated in stage 1 ("This can be extracted from the report")
xlim	x-axis limits for the illustrative plot
ylim	y-axis limits for the illustrative plot
trial.pe.cvs	vector of trial stage-2 process-error c.v.s (the default value will usually be OK; if not, an error message is output)
plot.it	If TRUE, plot the index and the smoothed fit. Otherwise, return a dataframe of the year, index, smoothed fitted value, and cv)

Value

Outputs a stage-2 process-error c.v. for the data set. Returns a 2-vector containing the process-error c.v. assumed or estimated in stage 1, and that estimated using TA1.10 for stage 2

Note

Method TA1.10 is described in Appendix A of the following paper Francis, R.I.C.C. (2011). Data weighting in statistical fisheries stock assessment models. Canadian Journal of Fisheries and Aquatic Sciences 68: 1124-1138. (With corrections to the equation in Francis R.I.C.C. (2011) Corrigendum: Data weighting in statistical fisheries stock assessment models. Haven't exported this function because I don't think it works even in Chris's DataWeighting package.

Author(s)

Chris Francis

Method.TA1.8	<i>Method.TA1.8</i>
--------------	---------------------

Description

This function is useful for deciding on the data weights of one or more at-age or at-length data sets with assumed multinomial error structure in a stock assessment. Can produce a diagnostic plot if the analysis is for a single data set

Usage

```
Method.TA1.8(model, observation_labels, plot.it = F, xlim = NULL,
             ylim = NULL)
```

Arguments

<code>model</code>	Casal2 output that is the result of a -r, -e run.
<code>observation_labels</code>	vector<string> Labels of the observations you want to apply the iterative weighting too, can be multiple datasets as in in Chris's original package <code>multiple = T</code> .
<code>plot.it</code>	If TRUE, plot the index and the smoothed fit. Otherwise, return a dataframe of the year, index, smoothed fitted value, and cv)
<code>xlim</code>	x-axis limits for the illustrative plot
<code>ylim</code>	y-axis limits for the illustrative plot

Value

Outputs a multiplier, w , so that $N2y = w \times N1y$, where $N1y$ and $N2y$ are the stage-1 and stage-2 multinomial sample sizes for the data set in year y .

Note

Method TA1.8 is described in Appendix A of the following paper Francis, R.I.C.C. (2011). Data weighting in statistical fisheries stock assessment models. Canadian Journal of Fisheries and Aquatic Sciences 68: 1124-1138. (With corrections to the equation in Francis R.I.C.C. (2011) Corrigendum: Data weighting in statistical fisheries stock assessment models.

Author(s)

Chris Francis

mpd_derived_quantity	<i>Utility function for summary</i>
----------------------	-------------------------------------

Description

Utility function for summary

Usage

```
mpd_derived_quantity(report_list)
```

Author(s)

C Marsh This is a utility function that will summarise a derived quantity report for a Casal2MPD class

param.profile	<i>Generate an objective score profile plot for a particular parameter as a consequence from a casal2 -p run</i>
---------------	--

Description

This function reads an extracted list from Casal2 and plots the likelihood/objective profile for one or many observations over the specified parameters range. This range is defined in the profile block of the Casal2 configuration file. Casal2 calculates the profile steps as even intervals between the upper and lower bound by $= (\text{upper_bound} - \text{lower_bound}) / (\text{steps} + 1)$

Usage

```
param.profile(obs, param, extract_list, Rescale = F,
  ylab = "Objective contribution", xlab = "Parameter", y_min = -0.5)
```

Arguments

obs	"vector<string>" A vector of report labels that correspond to observations that you would like to plot the profile for. Can be many observations
param	"string" The parameter label that the profile was run for. It should follow the syntax that is used in Casal2 that is, "block[label].parameter" e.g. process[Recruitment].b0
extract_list	"string" The name of the R object that was extracted into R using the extract() function.
Rescale	"bool/logical" If true it will rescale the likelihoods to have a minimum on 0. This is done by subtracting off the minimum value of the likelihood series.
ylab	"string" optional, y-axis label
xlab	"string" optional, x-axis label
ymin	"numeric" optional, minimum value parsed to the ylim parameter in plot()

Author(s)

Craig Marsh

```
param.profile.by.cohort
```

Generate an objective score profile plot for a particular parameter and age compositional dataset as a consequence from a casal2 -p run.

Description

This function reads an extracted list from Casal2 and plots the likelihood/objective profile for an age compositional observation over the specified parameters range. This range is defined in the profile block of the Casal2 configuration file. Casal2 calculates the profile steps as even intervals between the upper and lower bound by $= (\text{upper_bound} - \text{lower_bound}) / (\text{steps} + 1)$. The plot will generate a line for each cohort and category. The reason I made this function was for the specific investigation of the natural mortality parameter M. Sometimes running a profile of aggregated compositional data on a parameter is not that informative. The aim of this plot is to show which cohorts are suggestive of certain parameters values, this will either create more confusion (most likely) or at least give the user something interesting to think about (perhaps cohort based processes).

Usage

```
param.profile.by.cohort(obs, param, extract_list, Rescale = F,
  ylab = "Objective contribution", xlab = "Parameter", y_min = -0.5,
  threshold = 10)
```

Arguments

obs	"string" A report label that correspond to the age compositional observation that you would like to plot the profile for. Can be many observations
param	"string" The parameter label that the profile was run for. It should follow the syntax that is used in Casal2 that is, "block[label].parameter" e.g. process[Recruitment].b0
extract_list	"string" The name of the R object that was extracted into R using the extract() function.
Rescale	"bool/logical" If true it will rescale the likelihoods to have a minimum on 0. This is done by subtracting off the minimum value of the likelihood series.
ylab	"string" optional, y-axis label
xlab	"string" optional, x-axis label
threshold	"numeric" the minimum number of years a cohort is seen in the observation, for being used in the plot. if threshold = 5 then any cohort that has been sampled for a minimum of 5 years will be automatically plotted.
ymin	"numeric" optional, minimum value parsed to the ylim parameter in plot()

Author(s)

Craig Marsh

Paste	<i>Utility plot function</i>
-------	------------------------------

Description

Utility plot function

Usage

```
Paste(..., sep = "")
```

Author(s)

Craig Marsh

plot.derived_quantities	<i>plot.derived_quantities default</i>
-------------------------	--

Description

A plotting function to plot SSB's for the 'casal2TAB' and 'casal2MPD' objects.

Usage

```
plot.derived_quantities(model, type = "number", report_label = "", xlim,
  ylim, xlab, ylab, main, col, plot.it = T, ...)

## S3 method for class 'casal2MPD'
plot.derived_quantities(model, type = "number",
  report_label = "", xlim, ylim, xlab, ylab, main, col, plot.it = T, ...)
```

Arguments

model	<casal2MPD, casal2TAB> object that are generated from one of the extract() functions.
type	<string> whether numbers or scaled by B0.
report_label	<string>
...	remaining plotting functions.

Value

NULL

Author(s)

C. Marsh

pos	<i>Utility extract function</i>
-----	---------------------------------

Description

Utility extract function

Usage

```
pos(vector, x)
```

Author(s)

Dan Fu

pos.match	<i>Utility extract function</i>
-----------	---------------------------------

Description

Utility extract function

Usage

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

Author(s)

Dan Fu

pow	<i>Utility extract function</i>
-----	---------------------------------

Description

Utility extract function

Usage

```
pow(x, exponent)
```

Author(s)

Craig Marsh

ReadSimulatedData	<i>Read in multiple sets of Simualted data for a single observation</i>
-------------------	---

Description

This function reads in a set of simulated observations generated from Casal2 in simulation mode. These functions read in all the simulated obs as a list, for visualising and summarising in R

Usage

```
ReadSimulatedData(filename, path = "")
```

Arguments

filename	the name of simulated obs for an observation. For example if you generated 100 sets of simulated observations named "SubAntarticObs". Casal2 will generate 100 of these with the following extensions SubAntarticObs.001, SubAntarticObs.002, SubAntarticObs.003,,, SubAntarticObs.100. filename = SubAntarticObs.
path	Optionally, the path to the file, default is current working directory.

Author(s)

Craig Marsh

reformat.compositional.data	<i>Reformat Casal2 compositional observations so they are in the same format as the legacy Casal observations.</i>
-----------------------------	--

Description

This function will take a compositional observation that has been generated by Casal2 and re-format it so that it has the same structure as a CASAL reported compositional observation. The purpose for this function is to reformat the Casal2 observations so we can then feed them into packages that have been tailored for Casal observations, such as Chris Francis's DataWeighting library.

Usage

```
reformat.compositional.data(extract_list, comp_label)
```

Arguments

extract_list	the r object that has been extracted using the extract() function.
comp_label	<string> the label of the report for the observation you want converted

Author(s)

Craig Marsh

regex.in	<i>Utility extract function</i>
----------	---------------------------------

Description

Utility extract function

Usage

```
regex.in(vector, regex)
```

Author(s)

Dan Fu

Regexpr	<i>Utility extract function</i>
---------	---------------------------------

Description

Utility extract function

Usage

```
Regexpr(x, y, fixed = T)
```

Author(s)

Dan Fu

remove.first.words	<i>Utility extract function</i>
--------------------	---------------------------------

Description

Utility extract function

Usage

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

Author(s)

Dan Fu

`string.to.vector.of.numbers`
Utility extract function

Description

Utility extract function

Usage

`string.to.vector.of.numbers(string)`

Author(s)

Dan Fu

`string.to.vector.of.words`
Utility extract function

Description

Utility extract function

Usage

`string.to.vector.of.words(string)`

Author(s)

Dan Fu

`strip` *Utility for extract function*

Description

Utility for extract function

Usage

`strip(x)`

Author(s)

Craig Marsh

Sum	<i>Utility plot function</i>
-----	------------------------------

Description

Utility plot function

Usage

```
Sum(..., na.rm = T)
```

Author(s)

Craig Marsh

summarise_process	<i>Utility summarise_estimate_values function</i>
-------------------	---

Description

used in the summarise function for casal2MPD
used in the summarise function for casal2MPD

Usage

```
summarise_process(report_object)  
  
summarise_process(report_object)
```

Author(s)

Craig Marsh
Craig Marsh

summarise_warnings_encounted	<i>Utility summarise_warnings_encounted function</i>
------------------------------	--

Description

used in the summarise function for casal2MPD

Usage

```
summarise_warnings_encounted(report_object)
```

Author(s)

Craig Marsh

summary.default	<i>summary default</i>
-----------------	------------------------

Description

A summary function for 'casal2MCMC' 'casal2TAB' and 'casal2MPD' objects.

Usage

```
summary.default(model)

## S3 method for class 'casal2MPD'
summary(model)
```

Arguments

model	<casal2MPD, casal2TAB, casal2MCMC> object that are generated from one of the extract() functions.
-------	---

Value

NULL

Author(s)

C. Marsh

unpaste	<i>Utility extract function</i>
---------	---------------------------------

Description

Utility extract function

Usage

```
unpaste(string, sep)
```

Author(s)

Dan Fu

write.csl2.file	<i>Model configuration write function</i>
-----------------	---

Description

This function will write a Casal2 configuration file based on a list object in R. Ususally this function will be used once a model has been read into R using `extract.csl2.file` and modified. This function will then print our the configuration to a new file where it can be re run into Casal2

Usage

```
write.csl2.file(object, file, path = "")
```

Arguments

object	An R list object that follows the same structure that <code>extract.csl2.file</code> would produce
file	Optionally, the file name
path	Optionally, the path to ouput the file

Author(s)

Craig Marsh

Index

`check_short_hand`, [2](#)
`convert.to.lines`, [3](#)
`CV.for.CPUE`, [3](#)

`DoubleExponential`, [4](#)
`DoubleNormal`, [4](#)

`evalit`, [5](#)
`extract.csl2.file`, [5](#)
`extract.mcmc`, [6](#)
`extract.mpd`, [6](#)
`extract.parameters`, [7](#)

`get.casal2_list`, [7](#)
`get.line.label`, [8](#)
`get.line.type`, [8](#)
`get.lines`, [8](#)

`Increasing`, [9](#)
`InverseLogistic`, [9](#)
`is.all.numeric`, [10](#)
`is.even`, [10](#)
`is.in`, [11](#)
`is.odd`, [11](#)

`KnifeEdge`, [11](#)

`Logistic`, [12](#)
`LogisticProducing`, [13](#)

`make.complete_vector`, [13](#)
`make.data.frame`, [14](#)
`make.list`, [14](#)
`make.list_element`, [14](#)
`make.matrix`, [15](#)
`make.named_complete_vector`, [15](#)
`make.string_vector`, [15](#)
`make.vector`, [16](#)
`Method.TA1.10`, [16](#)
`Method.TA1.8`, [17](#)
`mpd_derived_quantity`, [18](#)

`param.profile`, [18](#)
`param.profile.by.cohort`, [19](#)
`Paste`, [20](#)

`plot.derived_quantities`, [20](#)
`pos`, [21](#)
`pos.match`, [21](#)
`pow`, [21](#)

`ReadSimulatedData`, [22](#)
`reformat.compositional.data`, [22](#)
`regexp.in`, [23](#)
`Regexpr`, [23](#)
`remove.first.words`, [23](#)

`string.to.vector.of.numbers`, [24](#)
`string.to.vector.of.words`, [24](#)
`strip`, [24](#)
`Sum`, [25](#)
`summarise_process`, [25](#)
`summarise_warnings_encountered`, [25](#)
`summary.casal2MPD (summary.default)`, [26](#)
`summary.default`, [26](#)

`unpaste`, [26](#)

`write.csl2.file`, [27](#)