

Package ‘casal2’

August 11, 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.

RoxygenNote 6.0.1

R topics documented:

convert.to.lines	2
CV.for.CPUE	3
evalit	3
extract.csl2.file	4
extract.mcmc	4
extract.mpd	5
extract.parameters	5
extract.tabular	6
get.casal2_list	6
get.line.label	7
get.line.type	7
get.lines	7
is.all.numeric	8
is.even	8
is.in	8
is.odd	9
make.complete_vector	9
make.data.frame	9
make.list	10
make.list_element	10
make.matrix	10
make.named_complete_vector	11

make.string_vector	11
make.vector	11
Method.TA1.8	12
mpd_derived_quantity	13
Paste	13
plot.derived_quantities	13
plot.fits	14
plot.pressure	15
plot.yes	16
pos	17
pos.match	18
pow	18
ReadSimulatedData	19
reformat.compositional.data	19
regexp.in	20
Regexpr	20
remove.first.words	20
string.to.vector.of.numbers	21
string.to.vector.of.words	21
strip	21
Sum	22
summarise_process	22
summarise_warnings_encountered	22
summary.default	23
transform.comparisons	23
unpaste	24
write.csl2.file	24
Index	25

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

Description

Utility extract function

Usage

`convert.to.lines(filename)`

Author(s)

Dan Fu

CV.for.CPUE

CV.for.CPUE

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

evalit

Utility plot function

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	<i>extract.mcmc function for casal2 output</i>
--------------	--

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	<i>extract MPD function for readin in Casal2 output that has been generated from a -r, -e, -f, -p run mode.</i>
-------------	---

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

Examples

```
library(casal2)
data <- extract.mpd(file = system.file("extdata", "MPD.log", package="casal2"))
class(data)
```

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

extract.tabular	<i>extract Tabular function for readin in Casal2 output that has been generated from a -r, -e, -f, -p run mode with the -tabular.</i>
-----------------	---

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 'casal2TAB' class which can be used in plotting and summary functions. See the casal2 manual for more information.

Usage

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

Arguments

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

Value

a 'casal2TAB' object which is essentially a list, that can be integrated using the str() function.

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

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

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
```

Utility extract function

Description

Utility extract function

Usage

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

Author(s)

Dan Fu

```
make.string_vector
```

Utility extract function

Description

Utility extract function

Usage

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

Author(s)

C Marsh

```
make.vector
```

Utility extract function

Description

Utility extract function

Usage

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

Author(s)

Dan Fu

Method.TA1.8

*Method.TA1.8***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 $N_{2y} = w \times N_{1y}$, where N_{1y} and N_{2y} 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

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 Derived Quantities for the 'casal2TAB' and 'casal2MPD' objects.

Usage

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

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

## S3 method for class 'casal2TAB'
plot.derived_quantities(model, report_label = "",
  type = "number", xlim, ylim, xlab, ylab, main, col, plot.it = T, ...)
```

Arguments

model	<casal2MPD, casal2TAB> object that are generated from one of the extract() functions.
report_label	<string>
type	<string> whether numbers or scaled by B0.
plot.it	Whether to generate a default plot or return the values as a matrix.
...	remaining plotting options

Value

A plot of derived quantities over time if plot.it = T, if plot.it = F it will return a matrix of derived quantities.

NULL

NULL

Author(s)

Craig Marsh

Examples

```
library(casal2)
# plotting Standard Output
data <- extract.mpd(file = system.file("extdata", "estimate.log", package="casal2"))
names(data)
plot.derived_quantities(model = data, report_label = "biomass")
# if you are unhappy with the default plotting you can use plot.it = FALSE and create a plot of your own.
SSB = plot.derived_quantities(model = data, report_label = "biomass", plot.it = FALSE)
plot(rownames(SSB),SSB, main = "My SSB", type = "l", ylim = c(0,90000))
# plotting Tabular Output
tab <- extract.tabular(file = system.file("extdata", "tabular_report.out", package="casal2"))
names(tab)
plot.derived_quantities(model = tab, report_label = "derived_quant")
```

plot.fits

plot.fits default

Description

A plotting function to fits to Casal2 observations from a model run.

Usage

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

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

```
## S3 method for class 'casal2TAB'
plot.fits(model, report_label = "", type = "resid",
  plot.it = T, xlim, ylim, xlab, ylab, main, col, ...)
```

Arguments

model	<casal2MPD, casal2TAB> object that are generated from one of the extract() functions.
report_label	<string>
plot.it	Whether to generate a default plot or return the values as a matrix.
...	remaining plotting options
type	Whether to plot an observed vs expected (fit) or plot the residuals (resid)

Value

A plot of derived quantities over time if plot.it = T, if plot.it = F it will return a matrix of derived quantities.

NULL

NULL

Author(s)

Craig Marsh

Examples

```
library(casal2)
# plotting Standard Output
data <- extract.mpd(file = system.file("extdata", "estimate.log", package="casal2"))
names(data)
plot.fits(model = data, report_label = "Tangaroa_propn_at_age_Aug")
plot.fits(model = data, report_label = "wcsiTRLcpue")
# if you are unhappy with the default plotting you can use plot.it = FALSE and create a plot of your own.
Tangaroa_fits = plot.fits(model = data, report_label = "Tangaroa_propn_at_age_Aug", plot.it = FALSE)
# plotting Tabular Output
tab <- extract.tabular(file = system.file("extdata", "tabular_report.out", package="casal2"))
names(tab)
plot.fits(model = tab, report_label = "Tangaroa_propn_at_age_Aug")
plot.fits(model = tab, report_label = "wcsiTRLcpue")
```

plot.pressure	<i>plot.pressure plot fishing pressure if there has been an exploitation process reported.</i>
---------------	--

Description

A plotting function to plot fishing pressure (U's) for the 'casal2TAB' and 'casal2MPD' objects.

Usage

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

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

## S3 method for class 'casal2TAB'
plot.pressure(model, report_label = "", xlim, ylim, xlab,
  ylab, main, col, plot.it = T, ...)
```

Arguments

model	<casal2MPD, casal2TAB> object that are generated from one of the extract.mpd() and extract.tabular() functions.
report_label	<string>
plot.it	Whether to generate a default plot or return the values as a matrix.
...	remaining plotting options

Value

generate a plot over time if plot.it = T, if plot.it = F it will return a matrix of values.

NULL

NULL

Author(s)

Craig Marsh

Examples

```
library(casal2)
# plotting Standard Output
data <- extract.mpd(file = system.file("extdata", "estimate.log", package="casal2"))
names(data)
plot.pressure(model = data, report_label = "exploit")
# if you are unhappy with the default plotting you can use plot.it = FALSE and create a plot of your own.
Fishing_pressures = plot.pressure(model = data, report_label = "exploit", plot.it = FALSE)
```

plot.ycs

plot.ycs plot true Year Class Strengths from a Casal2 model.

Description

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

Usage

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

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

```
## S3 method for class 'casal2TAB'
plot.ycs(model, report_label = "", xlim, ylim, xlab, ylab,
main, col, plot.it = T, ...)
```

Arguments

model	<casal2MPD, casal2TAB> object that are generated from one of the extract.mpd() and extract.tabular() functions.
report_label	<string>
plot.it	Whether to generate a default plot or return the values as a matrix.
...	remaining plotting options

Value

generate a plot over time if plot.it = T, if plot.it = F it will return a matrix of values.

NULL

NULL

Author(s)

Craig Marsh

Examples

```
library(casal2)
# plotting Standard Output
data <- extract.mpd(file = system.file("extdata", "estimate.log", package="casal2"))
names(data)
plot.ycs(model = data, report_label = "Rec")
# if you are unhappy with the default plotting you can use plot.it = FALSE and create a plot of your own.
true_YCS = plot.ycs(model = data, report_label = "Rec", plot.it = FALSE)
```

pos

Utility extract function

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
```

Reformat Casal2 compositional observations so they are in the same format as the legacy Casal observations.

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_encountered	<i>Utility summarise_warnings_encountered function</i>
--------------------------------	--

Description

used in the summarise function for casal2MPD

Usage

```
summarise_warnings_encountered(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

transform.comparisons	<i>Utility extract function</i>
-----------------------	---------------------------------

Description

Utility extract function

Usage

```
## S3 method for class 'comparisons'
transform(Data)
```

Author(s)

Craig 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

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

`evalit`, [3](#)
`extract.csl2.file`, [4](#)
`extract.mcmc`, [4](#)
`extract.mpd`, [5](#)
`extract.parameters`, [5](#)
`extract.tabular`, [6](#)

`get.casal2_list`, [6](#)
`get.line.label`, [7](#)
`get.line.type`, [7](#)
`get.lines`, [7](#)

`is.all.numeric`, [8](#)
`is.even`, [8](#)
`is.in`, [8](#)
`is.odd`, [9](#)

`make.complete_vector`, [9](#)
`make.data.frame`, [9](#)
`make.list`, [10](#)
`make.list_element`, [10](#)
`make.matrix`, [10](#)
`make.named_complete_vector`, [11](#)
`make.string_vector`, [11](#)
`make.vector`, [11](#)
`Method.TA1.8`, [12](#)
`mpd_derived_quantity`, [13](#)

`Paste`, [13](#)
`plot.derived_quantities`, [13](#)
`plot.fits`, [14](#)
`plot.pressure`, [15](#)
`plot.ycs`, [16](#)
`pos`, [17](#)
`pos.match`, [18](#)
`pow`, [18](#)

`ReadSimulatedData`, [19](#)
`reformat.compositional.data`, [19](#)
`regexp.in`, [20](#)
`Regexpr`, [20](#)
`remove.first.words`, [20](#)

`string.to.vector.of.numbers`, [21](#)
`string.to.vector.of.words`, [21](#)
`strip`, [21](#)
`Sum`, [22](#)
`summarise_process`, [22](#)
`summarise_warnings_encounted`, [22](#)
`summary.casal2MPD (summary.default)`, [23](#)
`summary.default`, [23](#)

`transform.comparisons`, [23](#)

`unpaste`, [24](#)

`write.csl2.file`, [24](#)