Package ‘r4ss’: r code for Stock Synthesis

Ian Taylor, Ian Stewart, Allan Hicks, Tommy Garrison,Andre Punt, John Wallace,
Chantel Wetzel, James Thorson,Yukio Takeuchi, Cole Monnahan, and other
contributors

SEDAR39-RD-12

October 2014

Package ‘r4ss’

July 12, 2014

Type Package

Title R code for Stock Synthesis

Version 1.22.1

Date 2014-07-11

Author Ian Taylor, Ian Stewart, Allan Hicks, Tommy Garrison,Andre Punt, John Wallace, Chantel Wetzel, James Thorson,Yukio Takeuchi, Cole Monnahan, and other contributors.

Maintainer Ian Taylor <`Ian.Taylor@noaa.gov`>

Depends R (>= 2.10.0)

Imports coda, corpcor, gplots, gtools, maps, pso, RCurl, tcltk

Description A collection of R functions for use with Stock Synthesis, a fisheries stock assessment modeling platform written in ADMB by Dr. Richard D. Methot at the NMFS Northwest Fisheries Science Center. The functions include tools for summarizing and plotting results, manipulating files,visualizing model parameterizations, and various other tasks.

License GPL-3

LazyLoad yes

URL `https://github.com/r4ss`

NeedsCompilation no

Repository CRAN

Date/Publication 2014-07-12 07:13:57

R topics documented:

`r4ss-package` *R tools for Stock Synthesis*

Description

A collection of R functions for use with Stock Synthesis, a fisheries stock assessment modeling platform written in ADMB by Dr. Richard D. Methot at the NMFS Northwest Fisheries Science Center. The functions include tools for summarizing and plotting results, manipulating files, visualizing model parameterizations, and various other tasks.

Details

Package:	r4ss
Type:	Package
Version:	1.22.1
Date:	2014-07-11
License:	GPL-3
LazyLoad:	yes
URL:	`https://github.com/r4ss/`

Should be compatible with Stock Synthesis versions 3.20 through 3.3.

Author(s)

Ian Taylor, Ian Stewart, Allan Hicks, Tommy Garrison, Andre Punt, John Wallace, Chantel Wetzel, James Thorson, Yukio Takeuchi, Kotaro Ono, and other contributors. Package maintainer: Ian Taylor <Ian.Taylor@noaa.gov>

References

r4ss on GitHub: `https://github.com/r4ss`
Download Stock Synthesis: `http://nft.nefsc.noaa.gov/`

Examples

```
## Not run:
# read in the report file using SS_output
myreplist <- SS_output(dir='c:/SS/simple/')

# make a collection of plots using SS_plots
SS_plots(replist=myreplist)

## End(Not run)
```

`addSSsummarize` *Add a model to the list of models to compare*

Description

Adds specified quantities from any model to the list of models returned from `SSsummarize` for further comparison.

Usage

```
addSSsummarize(origModels, newModels)
```

Arguments

`origModels`	A list of models created by `SSsummarize`.
`newModels`	A list of models to add to the originals models list. Each new model is an element of the list, and is a list itself with possible components described in the details below.

Value

Returns list as is returned from `SSsummarize`, but contains additions for the new models.

Note

This function was made to compare TINSS results and SS results, and assumed that you would always start with a list of SS models output from `SSsummarize`. It has not been tested to see how it works when starting with an empty list.

Author(s)

Allan Hicks

See Also

`SSsummarize` `SSplotComparisons`

Examples

```
## Not run:
  #######################################
  #DO NOT RUN
tinss1 <- list(npars=A$fit$npar,maxgrad=A$fit$maxgrad,nsexes=1,
               #note, there is an estimated parameter called sd_sbt,
               #      but it is a single value
               SpawnBio=data.frame(c(1964,1965,A$yrs),
                                   c(A$sbo,A$sbo,A$sbt)*1e6,0,
                                   qnorm(0.025,c(A$so,A$so,A$sbt)*1e6,0),
                                   qnorm(0.975,c(A$so,A$so,A$sbt)*1e6,0)),
               Bratio=data.frame(A$yrs,A$sbt/A$sbo,0,
                                 qnorm(0.025,A$sbt/A$sbo,0),
                                 qnorm(0.975,A$sbt/A$sbo,0)),
               SPRratio=data.frame(A$yr,A$spr,0,qnorm(0.025,A$spr,0),
                                   qnorm(0.975,A$spr,0)),
               recruits=data.frame(A$yr,A$nt[,1]*1e6,0,qnorm(0.025,A$nt[,1]*1e5,0),
                                   qnorm(0.975,A$nt[,1]*1e6,0)),
               #I'm not sure exactly what wt are,
               #   but it is important to line them up correctly
               recdevs=data.frame(A$recYrs,A$wt),
               indices = data.frame(A$iyr,1e6*A$yt,1e6*A$qbt,
                                    rep(A$q,length(A$iyr)),rep(0.4,length(A$iyr)),
                                    rep(0,length(A$iyr)),rep(1,length(A$iyr)))
               )
  tinss <- list(tinss1,tinss1)   #can add more models here
```

```
  #add TINSS model to SS models already summarized
  SSnTINSS <- addSSsummarize(models,tinss)
  mcmcInd <- seq(burnin+1,nrow(A$mc.sbt),thin)
  SSnTINSS$mcmc[[2]] <- data.frame(A$mc.sb0[mcmcInd],
                                   A$mc.sbt[mcmcInd,],
                                   A$mc.depl[mcmcInd,],
                                   A$mc.spr[mcmcInd,],
                                   A$mc.rt[mcmcInd,],
                                   log(A$mcmc[mcmcInd,"Ro"]*1e6),
                                   A$mcmc[mcmcInd,"msy"]*1e6)
  names(SSnTINSS$mcmc[[2]]) <-
    c("SPB_Virgin",paste("SPB",A$yrs,sep="_"),
      paste("Bratio",A$yrs,sep="_"),
      paste("SPRratio",A$yr,sep="_"),
      paste("Recr",A$yr,sep="_"),"SR_R0","TotYield_MSY")
  modelnames <- c("SS", "TINSS","TINSS.MLE")

  SSplotComparisons(SSnTINSS, legendlabels=modelnames,
                    subplot=2,endyr=2011,mcmcVec=c(T,T,F))
  title(main="MCMC")
  SSplotComparisons(SSnTINSS, legendlabels=modelnames,
                    subplot=4,endyr=2011,mcmcVec=c(T,T,F))
  title(main="MCMC")
  ################################################

## End(Not run)
```

`bubble3` *Create a bubble plot.*

Description

Bubble plot based on function vaguely based on `bubble` by Edzer Pebesma in gstat package. By default, positive values have closed bubbles and negative values have open bubbles.

Usage

```
bubble3(x, y, z, col = 1, cexZ1 = 5, maxsize = NULL, do.sqrt = TRUE,
  legend = TRUE, legendloc = "top", legend.z = "default",
  legend.yadj = 1.1, main = "", cex.main = 1, xlab = "", ylab = "",
  minnbubble = 8, xlim = NULL, ylim = NULL, axis1 = TRUE,
  xlimextra = 1, add = FALSE, las = 1, allopen = TRUE)
```

Arguments

`x`	Vector of x-values.
`y`	Vector of y-values.

`z`	Vector of bubble sizes.
`col`	Color for bubbles.
`cexZ1`	Character expansion (cex) value for a proportion of 1.0.
`maxsize`	Size of largest bubble. Prefered option is now an expansion factor for a bubble with z=1 (see `cexZ1` above).
`do.sqrt`	Should size be based on the area? (Diameter proportional to sqrt(z)). Default=TRUE.
`legend`	Add a legend to the plot?
`legendloc`	Location for legend (default='top')
`legend.z`	If a legend is added, what z values will be shown. Default is c(-3,-2,-1,.1,1,2,3) for Pearson-like quantities and a smaller range for proportions that are all less than 1.
`legend.yadj`	If a legend is added, how much should the y-axis be expanded to make space for it.
`main`	Title of plot. Default="".
`cex.main`	Charecter expansion for title. Default=1.
`xlab`	X-axis label.
`ylab`	Y-axis label.
`minnbubble`	Minimum number of unique x values below which extra space is added to horizontal axis (to make plot look better). Default = 8.
`xlim`	Optional limits on x-range.
`ylim`	Optional limits on y-range.
`axis1`	Show the horizontal axis on plot? Option allows turning off for use in multi-figure plots.
`xlimextra`	Extra space (see minnbubble above). Default = 1.
`add`	Add bubbles to existing plot? Default=FALSE.
`las`	Style of axis labels (see ?par for more info).
`allopen`	Should all bubbles be open (instead of just negative values)?

Author(s)

Ian Stewart and Ian Taylor

`DoProjectPlots` *Make plots from Rebuilder program*

Description

Make a set of plots based on output from Andre Punt's Rebuilder program.

Usage

```
DoProjectPlots(dirn = "C:/myfiles/", fileN = c("res.csv"), Titles = "",
  ncols = 200, Plots = list(1:25), Options = list(c(1:9)),
  LegLoc = "bottomright", yearmax = -1, Outlines = c(2, 2),
  OutlineMulti = c(2, 2), AllTraj = c(1, 2, 3, 4), AllInd = c(1, 2, 3, 4,
  5, 6, 7), BioType = "Spawning biomass", CatchUnit = "(mt)",
  BioUnit = "(mt)", BioScalar = 1, ColorsUsed = "default",
  Labels = "default", pdf = FALSE, pwidth = 7, pheight = 7, lwd = 2)
```

Arguments

`dirn`	Directory where rebuilder output files are stored.
`fileN`	Vector of filenames containing rebuilder output. Default=c("res.csv").
`Titles`	Titles for plots when using multiple filenames. Default="".
`ncols`	Number of columns to read in output file (fileN). Deafult=200.
`Plots`	List to get specific plots (currently 1 through 8). Default=list(1:25). If there are multiple files, supply a list of vectors, e.g. list(c(1,5),c(2:5))
`Options`	List to get specific strategies in the trajectory plots. Default=list(c(1:9)).If there are multiple files, supply a list of vectors, e.g. list(c(1,5),c(2:5))
`LegLoc`	Location for the legend (for plots with a legend). Default="bottomright".
`yearmax`	Maximum year to show in the plots. Set negative to show all years. Default=-1.
`Outlines`	Number of rows, columns for some of the plots. Default=c(2,2).
`OutlineMulti`	Number of rows, columns for other plots. Default=c(2,2).
`AllTraj`	Vector of trajectories to show. Default=c(1,2,3,4).
`AllInd`	Vector of individual plots to show. Default=c(1,2,3,4,5,6,7).
`BioType`	Label for biomass type. Default="Spawning biomass".
`CatchUnit`	Units of catch. Default="(mt)".
`BioUnit`	Units of biomass. Default="(mt)".
`BioScalar`	Scalar for biomass plot. Default=1.
`ColorsUsed`	Optional vector for alternative line colors. Default="default".
`Labels`	Optional vector for alternative legend labels. Default="default".
`pdf`	Option to send figures to pdf file instead of plot window in Rgui. Default=FALSE.
`pwidth`	Width of the plot window or PDF file (in inches). Default=7.
`pheight`	Height of the plot window or PDF file (in inches). Default=7.
`lwd`	Line width for many of the plot elements. Default=2.

Author(s)

Andre Punt

Examples

```
## Not run:
# example with one file
 DoProjectPlots(dirn="c:/myfiles/", Plots=1:8,
                Options=c(1,2,3,4,5,9), LegLoc="bottomleft")

# example with multiple files
 # Plots - set to get specific plots
 # Options - set to get specific strategies in the trajectory plots

 Titles <- c("Res1","Res2","Res3")
 Plots <- list(c(1:9),c(6:7))
 Options <- list(c(7:9,3),c(5,7))
 DoProjectPlots(fileN=c("res1.csv","res2.csv"),Titles=Titles,Plots=Plots,
                Options=Options,LegLoc="bottomleft",yearmax=-1,
                Outlines=c(2,2),OutlineMulti=c(3,3),AllTraj=c(1:4),
                AllInd=c(1:7),BioType="Spawning numbers",BioUnit="(lb)",
                BioScalar=1000,CatchUnit="(lb)",
                ColorsUse=rep(c("red","blue"),5),
                Labels=c("A","B","C","D","E","F"))

## End(Not run)
```

`getADMBHessian` *Read admodel.hes file*

Description

This function reads in all of the information contained in the admodel.hes file. Some of this is needed for relaxing the covariance matrix, and others just need to be recorded and rewritten to file so ADMB "sees" what it's expecting.

Usage

```
getADMBHessian(File, FileName)
```

Arguments

`File`	Directory in which .hes file is located.
`FileName`	Name of .hes file.

Value

A list with elements num.pars, hes, hybrid_bounded_flag, and scale.

Note

Also published here: http://www.admb-project.org/examples/admb-tricks/covariance-calculations

Author(s)

Cole Monnahan

`IOTCmove` *Make a map of movement for a 5-area Indian Ocean model*

Description

Run the `SSplotMovementMap` function with defaults related to a 5-area model for tunas in the Indian Ocean as discussed at the Indian Ocean Tuna Commission Working Party on Tropical Tunas in October, 2010. Obviously this is not useful for the majority of r4ss users, but it could serve as an example of how a wrapper function might be written for any other model.

Usage

```
IOTCmove(replist = NULL, moveage = 5, moveseas = 1, legend = FALSE,
  title = NULL, areanames = c("R1", "R2", "R3", "R4", "R5"), ...)
```

Arguments

`replist`	optional list created by `SS_output`
`moveage`	age for which movemement rates will be represented
`moveseas`	season for which movement rates will be represented
`legend`	add a legend to show the movement rate associated with the widest arrows
`title`	optional title to add at top of figure
`areanames`	vector of names for each area
`...`	Additional arguments can get passed to `SSplotMovementMap`

Author(s)

Ian Taylor

`make_multifig` *Create multi-figure plots.*

Description

Function created as an alternative to lattice package for multi-figure plots of composition data and fits from Stock Synthesis output.

Usage

```
make_multifig(ptsx, ptsy, yr, linesx = 0, linesy = 0, ptsSD = 0,
  sampsize = 0, effN = 0, showsampsize = TRUE, showeffN = TRUE,
  sampsizeround = 1, maxrows = 6, maxcols = 6, rows = 1, cols = 1,
  fixdims = TRUE, main = "", cex.main = 1, xlab = "", ylab = "",
  size = 1, cexZ1 = 1.5, bublegend = TRUE, maxsize = NULL,
  do.sqrt = TRUE, minnbubble = 8, allopen = TRUE, horiz_lab = "default",
  xbuffer = c(0.1, 0.1), ybuffer = c(0, 0.15), ymin0 = TRUE,
  axis1 = "default", axis2 = "default", linepos = 1, type = "o",
  bars = FALSE, barwidth = "default", ptscex = 1, ptscol = 1,
  ptscol2 = 1, linescol = 2, lty = 1, lwd = 1, pch = 1,
  nlegends = 3, legtext = list("yr", "sampsize", "effN"),
  legx = "default", legy = "default", legadjx = "default",
  legadjy = "default", legsize = c(1.2, 1), legfont = c(2, 1),
  sampsizeline = FALSE, effNline = FALSE, sampsizemean = NULL,
  effNmean = NULL, ipage = 0, scalebins = FALSE, ...)
```

Arguments

`ptsx`	vector of x values for points or bars
`ptsy`	vector of y values for points or bars of same length as ptsx
`yr`	vector of category values (years) of same length as ptsx
`linesx`	optional vector of x values for lines
`linesy`	optional vector of y values for lines
`ptsSD`	optional vector of standard deviations used to plot error bars on top of each point under the assumption of normally distributed error
`sampsize`	optional sample size vector of same length as ptsx
`effN`	optional effective sample size vector of same length as ptsx
`showsampsize`	show sample size values on plot?
`showeffN`	show effective sample size values on plot?
`sampsizeround`	rounding level for sample size values
`maxrows`	maximum (or fixed) number or rows of panels in the plot
`maxcols`	maximum (or fixed) number or columns of panels in the plot
`rows`	number or rows to return to as default for next plots to come or for single plots

`cols`	number or cols to return to as default for next plots to come or for single plots
`fixdims`	fix the dimensions at maxrows by maxcols or resize based on number of elements in `yr` input.
`main`	title of plot
`cex.main`	character expansion for title
`xlab`	x-axis label
`ylab`	y-axis label
`size`	vector of bubbles sizes if making a bubble plot
`cexZ1`	Character expansion (cex) for point associated with value of 1.
`bublegend`	Add legend with example bubble sizes to bubble plots.
`maxsize`	maximum size of bubbles
`do.sqrt`	scale bubbles based on sqrt of size vector. see ?bubble3 for more info.
`minnbubble`	number of unique x values before adding buffer. see ?bubble3 for more info.
`allopen`	should all bubbles be open? see ?bubble3 for more info.
`horiz_lab`	axis labels set horizontal all the time (TRUE), never (FALSE) or only when relatively short ("default")
`xbuffer`	extra space around points on the left and right as fraction of total width of plot
`ybuffer`	like xbuffer
`ymin0`	fix minimum y-value at 0?
`axis1`	position of bottom axis values
`axis2`	position of left size axis values
`linepos`	should lines be added on top of points (linepos=1) or behind (linepos=2)?
`type`	type of line/points used for observed values (see 'type' in ?plot for details) on top of a grey polygon. Default is "o" for overplotting points on lines.
`bars`	should the ptsx/ptsy values be bars instead of points (TRUE/FALSE)
`barwidth`	width of bars in barplot, default method chooses based on quick and dirty formula also, current method of plot(...type='h') could be replaced with better approach
`ptscex`	character expansion factor for points (default=1)
`ptscol`	color for points/bars
`ptscol2`	color for negative value points in bubble plots
`linescol`	color for lines
`lty`	line type
`lwd`	line width
`pch`	point character type
`nlegends`	number of lines of text to add as legends in each plot
`legtext`	text in legend, a list of length=nlegends. values may be any of 1. "yr", 2. "sampsize", 3. "effN", or a vector of length = ptsx.

`legx`	vector of length=nlegends of x-values of legends (default is first one on left, all after on right)
`legy`	vector of length=nlegends of y-values of legends (default is top for all plots)
`legadjx`	left/right adjustment of legends around legx
`legadjy`	left/right adjustment of legends around legy
`legsize`	font size for legends. default=c(1.2,1.0) (larger for year and normal for others)
`legfont`	font type for legends, same as "font" under ?par
`sampsizeline`	show line for input sample sizes on top of conditional age-at-length plots (TRUE/FALSE/scalar, still in development)
`effNline`	show line for effective sample sizes on top of conditional age-at-length plots (TRUE/FALSE/scalar, still in development)
`sampsizemean`	mean input sample size value (used when sampsizeline=TRUE)
`effNmean`	mean effective sample size value (used when effNline=TRUE)
`ipage`	which page of plots when covering more than will fit within maxrows by maxcols.
`scalebins`	Rescale expected and observed proportions by dividing by bin width for models where bins have different widths? Caution!: May not work correctly in all cases.
`...`	additional arguments (NOT YET IMPLEMENTED).

Author(s)

Ian Taylor

See Also

`SS_plots`,`SSplotComps`

`mcmc.nuisance` *Summarize nuisance MCMC output*

Description

Summarize nuisance MCMC output (used in combination with `mcmc.out` for key parameters).

Usage

```
mcmc.nuisance(directory = "c:/mydirectory/", run = "mymodel/",
  file = "posteriors.sso", file2 = "derived_posteriors.sso",
  bothfiles = FALSE, printstats = FALSE, burn = 0, header = TRUE,
  thin = 1, trace = 0, labelstrings = "all", columnnumbers = "all",
  sep = "")
```

Arguments

`directory`	Directory where all results are located, one level above directory for particular run.
`run`	Directory with files from a particular run.
`file`	File containing posterior samples for nuisance parameters. This could be posteriors.sso or something written by the function `SSgetMCMC`.
`file2`	Optional second file containing posterior samples for nuisance parameters. This could be derived_posteriors.sso.
`bothfiles`	TRUE/FALSE indicator on whether to read `file2` in addition to `file1`.
`printstats`	Return all the statistics for a closer look.
`burn`	Optional burn-in value to apply on top of the option in the starter file and `SSgetMCMC`.
`header`	Data file with header?
`thin`	Optional thinning value to apply on top of the option in the starter file, in the `mcsave` runtime command, and in `SSgetMCMC`.
`trace`	Plot trace for param # (to help sort out problem parameters).
`labelstrings`	Vector of strings that partially match the labels of the parameters you want to consider.
`columnnumbers`	Vector of column numbers indicating the columns you want to consider.
`sep`	Separator for data file passed to the `read.table` function.

Author(s)

Ian Stewart

See Also

`mcmc.out`, `SSgetMCMC`

`mcmc.out` *Summarize, analyze and plot key MCMC output.*

Description

Makes four panel plot showing trace plots, moving average, autocorrelations, and densities for chosen parameters from MCMC output.

Usage

```
mcmc.out(directory = "c:/mydirectory/", run = "mymodel/",
  file = "keyposteriors.csv", namefile = "postplotnames.sso",
  names = FALSE, headernames = TRUE, numparams = 1, closeall = TRUE,
  burn = 0, thin = 1, scatter = FALSE, surface = FALSE, surf1 = 1,
  surf2 = 2, stats = FALSE, plots = TRUE, header = TRUE, sep = ",",
  print = FALSE)
```

Arguments

`directory`	Directory where all results are located, one level above directory for particular run.
`run`	Directory with files from a particular run.
`file`	File containing posterior samples for key parameters. This could be written by the function `SSgetMCMC`.
`namefile`	The (optional) file name of the dimension and names of posteriors.
`names`	Read in names file (T) or use generic naming (F).
`headernames`	Use the names in the header of `file`?
`numparams`	The number of parameters to analyze.
`closeall`	By default close all open devices.
`burn`	Optional burn-in value to apply on top of the option in the starter file and `SSgetMCMC`.
`thin`	Optional thinning value to apply on top of the option in the starter file, in the `-mcsave` runtime command, and in `SSgetMCMC`.
`scatter`	Can add a scatter-plot of all params at end, default is none.
`surface`	Add a surface plot of 2-way correlations.
`surf1`	The first parameter for the surface plot.
`surf2`	The second parameter for the surface plot.
`stats`	Print stats if desired.
`plots`	Show plots or not.
`header`	Data file with header?
`sep`	Separator for data file passed to the `read.table` function.
`print`	Send to screen unless asked to print.

Author(s)

Ian Stewart

See Also

`mcmc.nuisance`, `SSgetMCMC`

`mountains` *Make shaded polygons with a mountain-like appearance*

Description

Designed to replicate like the cool-looking Figure 7 in Butterworth et al. (2003).

Usage

```
mountains(zmat, xvec = NULL, yvec = NULL, zscale = 3, rev = TRUE,
  nshades = 100, axes = TRUE, xaxs = "i", yaxs = "i", xlab = "",
  ylab = "", las = 1, addbox = FALSE, ...)
```

Arguments

`zmat`	A matrix where the rows represent the heights of each mountain range
`xvec`	Optional input for the x variable
`yvec`	Optional input for the y variable
`zscale`	Controls the height of the mountains relative to the y-axis and max(zmat)
`rev`	Reverse the order of the display of yvec values.
`nshades`	Number of levels of shading
`axes`	Add axes to the plot?
`xaxs`	X-axis as internal or regular (see ?par for details)
`yaxs`	Y-axis as internal or regular (see ?par for details)
`xlab`	Optional label for x-axis
`ylab`	Optional label for y-axis
`las`	Xxis label style (see ?par for details). Default = 1 = horizontal axis labels.
`addbox`	Puts a box around the whole plot
`...`	Extra inputs passed to the plot command

Author(s)

Ian Taylor

References

Butterworth D.S., Ianelli J.N., Hilborn R. (2003) A statistical model for stock assessment of southern bluefin tuna with temporal changes in selectivity. South African Journal of Marine Science 25:331-362.

`movepars` *Explore movement parameterizations in a GUI*

Description

A function to visualize parameterization of movement in Stock Synthesis. It creates a GUI interface for movement exploration. Based on selectivity GUI by Tommy Garrison

Usage

```
movepars(nareas = 4, accuage = 40, season.duration = 1,
  min.move.age = 0.5)
```

Arguments

`nareas`	Number of areas
`accuage`	Accumulator age
`season.duration`	Length of season (annual rates are scaled by this value in SS).
`min.move.age`	Minimum age of movement.

Author(s)

Ian Taylor

`NegLogInt_Fn` *Perform SS implementation of Laplace Approximation*

Description

(Attempt to) perform the SS implementation of the Laplace Approximation from Thorson, Hicks and Methot (2014) ICES J. Mar. Sci.

Usage

```
NegLogInt_Fn(File = NA, Input_SD_Group_Vec, CTL_linenum_List, ESTPAR_num_List,
  PAR_num_Vec, Int_Group_List, Version = 5, StartFromPar = TRUE,
  Intern = TRUE, ReDoBiasRamp = FALSE, BiasRamp_linenum_Vec = NULL,
  CTL_linenum_Type = NULL)
```

Arguments

`File`	Directory containing Stock Synthesis files (e.g., "C:/Users/James Thorson/Desktop/")
`Input_SD_Group_Vec`	Vector where each element is the standard deviation for a group of random effects (e.g., a model with a single group of random effects will have Input_SD_Group_Vec be a vector of length one)
`CTL_linenum_List`	List (same length as `Input_SD_Group_Vec`), where each element is a vector giving the line number(s) for the random effect standard deviation parameter or penalty in the CTL file (and where each line will correspond to a 7-parameter or 14-parameter line).
`ESTPAR_num_List`	List (same length as `Input_SD_Group_Vec`), where each element is a vector giving the parameter number for the random effect coefficients in that group of random effects. These "parameter numbers" correspond to the number of these parameters in the list of parameters in the "ss3.std" output file.
`PAR_num_Vec`	Vector giving the number in the "ss3.par" vector for each random effect coefficient.
`Int_Group_List`	List where each element is a vector, providing a way of grouping different random effect groups into a single category. This is not used when `Version=1`.
`Version`	Integer (options are 1, 5, and 6) giving the type of Laplace Approximation. I recommend 1.
`StartFromPar`	Logical flag (TRUE or FALSE) saying whether to start each round of optimization from a "ss3.par" file (I recommend TRUE)
`Intern`	Logical flag saying whether to display all ss3 runtime output in the R terminal
`ReDoBiasRamp`	Logical flag saying whether to re-do the bias ramp (using `SS_fitbiasramp`) each time Stock Synthesis is run.
`BiasRamp_linenum_Vec`	Vector giving the line numbers from the CTL file that contain the information about the bias ramp.
`CTL_linenum_Type`	Character vector (same length as `Input_SD_Group_Vec`), where each element is either "Short_Param", "Long_Penalty", "Long_Penalty". Default is NULL, and if not explicitly specified the program will attempt to detect these automatically based on the length of relevant lines from the CTL file.

Author(s)

James Thorson

References

Thorson, J.T., Hicks, A.C., and Methot, R.D. 2014. Random effect estimation of time-varying factors in Stock Synthesis. ICES J. Mar. Sci.

`PinerPlot` *Make plot of likelihood contributions by fleet*

Description

This style of plot was officially named a "Piner Plot" at the CAPAM Selectivity Workshop, La Jolla March 2013. This is in honor of Kevin Piner's contributions to interpreting likelihood profiles. He's surely not the first person to make such a plot but the name seems to have stuck.

Usage

```
PinerPlot(summaryoutput, plot = TRUE, print = FALSE,
  component = "Length_like",
  main = "Changes in length-composition likelihoods by fleet",
  models = "all", fleets = "all", fleetnames = "default",
  profile.string = "R0", profile.label = expression(log(italic(R)[0])),
  ylab = "Change in -log-likelihood", col = "default", pch = "default",
  lty = 1, lty.total = 1, lwd = 2, lwd.total = 3, cex = 1,
  cex.total = 1.5, xlim = "default", ymax = "default", xaxs = "r",
  yaxs = "r", type = "o", legend = TRUE, legendloc = "topright",
  pwidth = 7, pheight = 7, punits = "in", res = 300, ptsize = 12,
  cex.main = 1, plotdir = NULL, verbose = TRUE)
```

Arguments

`summaryoutput`	List created by the function `SSsummarize`.
`plot`	Plot to active plot device?
`print`	Print to PNG files?
`component`	Which likelihood component to plot. Default is "Length_like".
`main`	Title for plot. Should match component.
`models`	Optional subset of the models described in `summaryoutput`. Either "all" or a vector of numbers indicating columns in summary tables.
`fleets`	Optional vector of fleet numbers to include.
`fleetnames`	Optional character vector of names for each fleet.
`profile.string`	Character string used to find parameter over which the profile was conducted. Needs to match substring of one of the SS parameter labels found in the Report.sso file. For instance, the default input 'steep' matches the parameter 'SR_BH_steep'.
`profile.label`	Label for x-axis describing the parameter over which the profile was conducted.
`ylab`	Label for y-axis. Default is "Change in -log-likelihood".
`col`	Optional vector of colors for each line.
`pch`	Optional vector of plot characters for the points.
`lty`	Line total for the liklihood components.
`lty.total`	Line type for the total likelihood.

`lwd`	Line width for the liklihood components.
`lwd.total`	Line width for the total likelihood.
`cex`	Character expansion for the points representing the likelihood components.
`cex.total`	Character expansion for the points representing the total likelihood.
`xlim`	Range for x-axis. Change in likelihood is calculated relative to values within this range.
`ymax`	Maximum y-value. Default is 10% greater than largest value plotted.
`xaxs`	The style of axis interval calculation to be used for the x-axis (see ?par for more info)
`yaxs`	The style of axis interval calculation to be used for the y-axis (see ?par for more info).
`type`	Line type (see ?plot for more info).
`legend`	Include legend?
`legendloc`	Location of legend (see ?legend for more info).
`pwidth`	Width of plot written to PNG file
`pheight`	Height of plot written to PNG file
`punits`	Units for PNG file
`res`	Resolution for PNG file
`ptsize`	Point size for PNG file
`cex.main`	Character expansion for plot titles
`plotdir`	Directory where PNG files will be written. by default it will be the directory where the model was run.
`verbose`	Return updates of function progress to the R GUI? (Doesn't do anything yet.)

Author(s)

Ian Taylor, Kevin Piner

`plotCI` *Plot points with confidence intervals.*

Description

Given a set of x and y values and upper and lower bounds, this function plots the points with error bars. This was Written by Venables and modified to add access to ylim and contents.

Usage

```
plotCI(x, y = NULL, uiw, liw = uiw, ylo = NULL, yhi = NULL, ...,
  sfrac = 0.01, ymax = NULL, add = FALSE, col = "black")
```

Arguments

`x`	The x coordinates of points in the plot
`y`	The y coordinates of the points in the plot.
`uiw`	The width of the upper portion of the confidence region.
`liw`	The width of the lower portion of the confidence region.
`ylo`	Lower limit of y range.
`yhi`	Upper limit of y range.
`...`	Additional inputs that will be passed to the function `plot(x,y,ylim=ylim,...)`
`sfrac`	Fraction of width of plot to be used for bar ends.
`ymax`	Additional input for Upper limit of y range.
`add`	Add points and intervals to existing plot? Default=FALSE.
`col`	Color for the points and lines.

Author(s)

Bill Venables, Ian Stewart, Ian Taylor, John Wallace

`r4ss_logo` *Make a simple logo for r4ss organization on GitHub*

Description

I was tired of the automatically generated symbol that appeared by default.

Usage

```
r4ss_logo()
```

Author(s)

Ian Taylor

`read.admbFit` *Read ADMB .par and .cor files.*

Description

This function will parse the .par and .cor files to provide things like parameter estimates, standard deviations, and correlations. Required for Jim Thorson's Laplace Approximation but likely useful for other purposes.

Usage

```
read.admbFit(file)
```

Arguments

`file`	Name of ADMB executable such that files to read will have format file.par and file.cor.

Value

List of various things from these files.

Author(s)

James Thorson

`rich.colors.short` *Make a vector of colors.*

Description

A subset of rich.colors by Arni Magnusson from the gplots package, with the addition of alpha transparency (which is now available in the gplots version as well)

Usage

```
rich.colors.short(n, alpha = 1)
```

Arguments

`n`	Number of colors to generate.
`alpha`	Alpha transparency value for all colors in vector. Value is passed to rgb function.

Author(s)

Arni Magnusson, Ian Taylor

`sel.line` *a function for drawing selecitivity curves*

Description

This function is primarily inteded for use by the selfit function.

Usage

```
sel.line(x, model, sp, min.dist, max.dist)
```

Arguments

`x`	vector of x values (age or length)
`model`	selectivity model "Double_Normal" or "Double_Logistic"
`sp`	vector of parameters
`min.dist`	minimum value for selectivity
`max.dist`	maximum value for selectivity

Author(s)

Tommy Garrison

See Also

`selfit`

Examples

```
## Not run:
plot(0, xlim = c(0, 50), ylim = c(0, 1),
xlab = 'Length', ylab = 'Selectivity', type = 'n',
xaxs = 'i', yaxs = 'i')
sel.line(model = 'Double_Normal', min.dist = 10, max.dist = 50,
sp = c(25, -0.5, 3, 3, -5, 0))

## End(Not run)
```

---

selfit *A function to visual parameterization of double normal and double logistic selectivity in Stock Synthesis*

Description

A GUI interface for exploring selectivity.

Usage

```
selfit(minLength = 10, maxLength = 65, silent = FALSE, init = NULL)
```

Arguments

`minLength`	Minimum size to show
`maxLength`	Maximum size to show
`silent`	T/F switch to return fit at the end
`init`	Optional initial values for the parameters

Author(s)

Tommy Garrison

See Also

`sel.line`

Examples

```
## Not run:
selfit()

## End(Not run)
```

---

selfit_spline *visualize parameterization of cubic spline selectivity in SS*

Description

A GUI interface for exploring spline selectivity.

Usage

```
selfit_spline(n = 4, minBin = 10, maxBin = 65, knots = NULL,
  slopevec = c(0.01, -0.01), params = NULL, dir = getwd(),
  silent = FALSE)
```

Arguments

`n`	Number of knots.
`minBin`	Minimum length or age to show.
`maxBin`	Maximum length or age to show.
`knots`	Vector giving location of each knot.
`slopevec`	Optional initial values parameters controlling slope at first and last knot.
`params`	Optional initial values for the parameters controlling selectivity at each knot.
`dir`	Directory in which the spline_selex executable is located (default = working directory).
`silent`	TRUE/FALSE switch to return fit at the end.

Author(s)

Ian Taylor

See Also

`selfit`

Examples

```
## Not run:
selfit_spline()

## End(Not run)
```

`SSbootstrap` *Fit models to parametric bootstraps*

Description

Run a series of models fit to parametric bootstrap data taken from data.ss_new. This is not yet a generalized function, just some example code for how to do a parametric bootstrap such as was done for the Pacific hake model in 2006.

Usage

```
SSbootstrap()
```

Note

Thanks to Nancie Cummings for inspiration.

Author(s)

Ian Taylor

References

http://www.pcouncil.org/wp-content/uploads/2006_hake_assessment_FINAL_ENTIRE.pdf (A description is on page 41 and Figures 55-56 (pg 139-140) show some results.)

SSFishGraph *A function for converting Stock Synthesis output to the format used by FishGraph*

Description

Only skeleton of a function right now, needs work. Intended as a translator to convert the output from object created by `SS_output` to the format used by FishGraph.

Usage

```
SSFishGraph(replist, title = "SSv3 output", species = "some kind of fish")
```

Arguments

`replist`	Object created by SS_output
`title`	Title of output
`species`	Species name

Author(s)

Ian Taylor

References

A website related to FishGraph is http://r-forge.r-project.org/projects/fishgraph/

SSgetMCMC *Read MCMC output.*

Description

Reads the MCMC output (in the posteriors.sso and derived_posteriors.sso files) from one or more models.

Usage

```
SSgetMCMC(dir = NULL, verbose = TRUE, writecsv = FALSE,
  csv1 = "keyposteriors.csv", csv2 = "nuisanceposteriors.csv",
  keystrings = c("NatM", "R0", "steep", "RecrDev_2008", "Q_extraSD"),
  nuisancestrings = c("Objective_function", "SPB_", "InitAge", "RecrDev"),
  modelnames = "default", burnin = 0, thin = 1)
```

Arguments

`dir`	A string (or vector of strings) of the directory (or directories) with MCMC output.
`verbose`	TRUE/FALSE switch to get more or less information about the progress of the function.
`writecsv`	Write key parameters and certainty nuisance quantities to a CSV file.
`csv1`	First CSV file for key parameters.
`csv2`	Second CSV file for nuisance quantities.
`keystrings`	Vector of strings that partially match parameter names to write to the file csv1. This file intended to feed into `mcmc.out`.
`nuisancestrings`	Vector of strings that partially match derived quantity names to write to the file csv2. This file intended to feed into `mcmc.nuisance`.
`modelnames`	Either "default" or a vector of names to use in naming elements of list that is output by the function. Default is "model1", "model2", etc.
`burnin`	Optional burn-in value to apply on top of the option in the starter file.
`thin`	Optional thinning value to apply on top of the option in the starter file and in the `-mcsave` runtime command.

Author(s)

Ian Taylor

See Also

`mcmc.out`, `mcmc.nuisance`, `SSplotPars`

`SSgetoutput` *Get output from multiple Stock Synthesis models.*

Description

Apply the function `SS_output` multiple times and save output as individual objects or a list of lists.

Usage

```
SSgetoutput(keyvec = NULL, dirvec = NULL, getcovar = TRUE,
  getcomp = TRUE, forecast = FALSE, verbose = TRUE, ncols = 210,
  listlists = TRUE, underscore = FALSE, save.lists = FALSE)
```

Arguments

`keyvec`	A vector of strings that are appended to the output files from each model if models are all in one directory. Default=NULL.
`dirvec`	A vector of directories (full path or relative to working directory) in which model output is located. Default=NULL.
`getcovar`	Choice to read or not read covar.sso output (saves time and memory). Default=TRUE.
`getcomp`	Choice to read or not read CompReport.sso output (saves time and memory). Default=TRUE.
`forecast`	Choice to read or not read forecast quantities. Default=FALSE.
`verbose`	Print various messages to the command line as the function runs? Default=TRUE.
`ncols`	Maximum number of columns in Report.sso (same input as for `SS_output`). Default=210.
`listlists`	Save output from each model as a element of a list (i.e. make a list of lists). Default = TRUE.
`underscore`	Add an underscore '_' between any file names and any keys in keyvec. Default=FALSE.
`save.lists`	Save each list of parsed output as a .Rdata file (with default filenaming convention based on iteration and date stamp.

Author(s)

Ian Taylor

See Also

`SS_output` `SSsummarize`

`SSmakeMmatrix` *Convert a matrix of natural mortality values into inputs for Stock Synthesis*

Description

Inspired by Valerio Bartolino and North Sea herring

Usage

```
SSmakeMmatrix(mat, startyr, outfile = NULL, overwrite = FALSE,
  yrs.in.columns = TRUE)
```

Arguments

`mat`	a matrix of natural mortality by year and age, starting with age 0
`startyr`	the first year of the natural mortality values (no missing years)
`outfile`	optional file to which the results will be written
`overwrite`	if 'outfile' is provided and exists, option to overwrite or not
`yrs.in.columns`	an indicator of whether the matrix has years in columns or rows

Value

Prints inputs with option to write to chosen file

Author(s)

Ian Taylor

`SSplotBiology` *Plot biology related quantities.*

Description

Plot biology related quantities from Stock Synthesis model output, including mean weight, maturity, fecundity, and spawning output.

Usage

```
SSplotBiology(replist, plot = TRUE, print = FALSE, add = FALSE,
  subplots = 1:11, seas = 1, col1 = "red", col2 = "blue",
  legendloc = "topleft", plotdir = "default", labels = c("Length (cm)",
  "Age (yr)", "Maturity", "Mean weight (kg) in last year", "Spawning output",
  "Length (cm, middle of the year)", "Natural mortality", "Female weight (kg)",
  "Female length (cm)", "Fecundity", "Default fecundity label", "Year"),
  pwidth = 7, pheight = 7, punits = "in", res = 300, ptsize = 12,
  cex.main = 1, verbose = TRUE)
```

Arguments

`replist`	List created by `SS_output`
`plot`	Plot to active plot device?
`print`	Print to PNG files?
`add`	add to existing plot
`subplots`	vector controlling which subplots to create
`seas`	which season to plot (obviously only works in seasonal models, but maybe not fully implemented even then)
`col1`	color of some points/lines

`col2`	color of other points/lines
`legendloc`	Location of legend (see ?legend for more info)
`plotdir`	Directory where PNG files will be written. by default it will be the directory where the model was run.
`labels`	Vector of labels for plots (titles and axis labels)
`pwidth`	Width of plot written to PNG file
`pheight`	Height of plot written to PNG file
`punits`	Units for PNG file
`res`	Resolution for PNG file
`ptsize`	Point size for PNG file
`cex.main`	Character expansion for plot titles
`verbose`	Return updates of function progress to the R GUI?

Author(s)

Ian Stewart, Ian Taylor

See Also

`SS_plots`, `SS_output`

`SSplotCatch` *Plot catch related quantities.*

Description

Plot catch related quantities from Stock Synthesis output. Plots include harvest rate, continuous F, landings, and discard fraction.

Usage

```
SSplotCatch(replist, subplots = 1:15, add = FALSE, areas = 1,
  plot = TRUE, print = FALSE, type = "l", fleetlty = 1, fleetpch = 1,
  fleetcols = "default", fleetnames = "default", lwd = 3,
  areacols = "default", areanames = "default", minyr = NULL,
  maxyr = NULL, annualcatch = TRUE, forecastplot = TRUE,
  plotdir = "default", showlegend = TRUE, legendloc = "topleft",
  order = "default", xlab = "Year", labels = c("Harvest rate/Year",
  "Continuous F", "Landings", "Total catch", "Predicted Discards",
  "Discard fraction", "(mt)", "(numbers x1000)", "Observed and expected",
  "aggregated across seasons"), catchasnumbers = NULL, catchbars = TRUE,
  addmax = TRUE, ymax = NULL, pwidth = 7, pheight = 7, punits = "in",
  res = 300, ptsize = 12, cex.main = 1, verbose = TRUE)
```

Arguments

`replist`	List created by `SS_output`
`subplots`	Vector controlling which subplots to create
`add`	Add to existing plot? (not yet implemented)
`areas`	Optional subset of areas to plot for spatial models
`plot`	Plot to active plot device?
`print`	Print to PNG files?
`type`	Type parameter passed to plot function. Default "l" is lines only. Other options include "o" for overplotting points on lines.
`fleetlty`	Vector of line type by fleet
`fleetpch`	Vector of plot character by fleet
`fleetcols`	Vector of colors by fleet
`fleetnames`	Optional replacement for fleenames used in data file
`lwd`	Line width
`areacols`	Vector of colors by area. Default uses rich.colors by Arni Magnusson
`areanames`	Names for areas. Default is to use Area1, Area2,...
`minyr`	Optional input for minimum year to show in plots
`maxyr`	Optional input for maximum year to show in plots
`annualcatch`	Include plot of catch aggregated across seasons within each year
`forecastplot`	Add points from forecast years
`plotdir`	Directory where PNG or PDF files will be written. By default it will be the directory where the model was run.
`showlegend`	Put legend on plot
`legendloc`	Location of legend (see ?legend for more info)
`order`	Optional input to change the order of fleets in stacked plots.
`xlab`	x-label for all plots
`labels`	Vector of labels for plots (titles and axis labels)
`catchasnumbers`	Is catch in numbers instead of biomass? Should be set automatically if set to NULL. If fleets include a mix of biomass and numbers, then catch plots should be interpreted carefully.
`catchbars`	Show catch by fleet as barplot instead of stacked polygons? (default=TRUE)
`addmax`	Add a point on the y-axis for the maximum catch (default=TRUE)
`ymax`	Optional input for ymax value (can be used to add or subtract white space at the top of the figure)
`pwidth`	Width of plot written to PNG file
`pheight`	Height of plot written to PNG file
`punits`	Units for PNG file
`res`	Resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	Character expansion for plot titles
`verbose`	Report progress to R console?

Author(s)

Ian Taylor, Ian Stewart

See Also

`SS_plots`, `SS_output`

`SSplotCohorts` *Plot cumulative catch by cohort.*

Description

Cumulative catch contributions for each cohort are plotted based on estimated catch-at-age matrix and weight-at-age values by fleet. Curves are shown in units of both numbers and biomass.

Usage

```
SSplotCohorts(replist, subplots = 1:2, add = FALSE, plot = TRUE,
  print = FALSE, cohortcols = "default", cohortfrac = 1,
  cohortvec = NULL, cohortlabfrac = 0.1, cohortlabvec = NULL, lwd = 3,
  plotdir = "default", xlab = "Year", labels = c("Age",
  "Cumulative catch by cohort (in numbers x1000)",
  "Cumulative catch by cohort (x1000 mt)"), pwidth = 7, pheight = 7,
  punits = "in", res = 300, ptsize = 12, cex.main = 1, verbose = TRUE)
```

Arguments

`replist`	List created by `SS_output`
`subplots`	Vector controlling which subplots to create
`add`	Add to existing plot? (not yet implemented)
`plot`	Plot to active plot device?
`print`	Print to PNG files?
`cohortcols`	Vector of colors to show for each cohort. Default is range of colors shade indicating time period.
`cohortfrac`	What fraction of the cohorts to include in plot. If value < 1 is used, then cohorts are filtered to only include those with the highest maximum cumulative catch. Value will be overridden by `cohortvec`.
`cohortvec`	Optional vector of birth years for cohorts to include in plot. Value overrides `cohortfrac`.
`cohortlabfrac`	What fraction of the cohorts to label in plot. By default, top 10% of cohorts are labeled. Value will be overridden by `cohortlabvec`.
`cohortlabvec`	Optional vector of birth years for cohorts to label in plot. Value overrides `cohortlabfrac`.
`lwd`	Line width

`plotdir`	Directory where PNG or PDF files will be written. By default it will be the directory where the model was run.
`xlab`	x-label for all plots
`labels`	Vector of labels for plots (titles and axis labels)
`pwidth`	Width of plot written to PNG file
`pheight`	Height of plot written to PNG file
`punits`	Units for PNG file
`res`	Resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	Character expansion for plot titles (no titles in this function yet)
`verbose`	Report progress to R console?

Author(s)

Ian Taylor

See Also

`SS_plots`, `SS_output`

`SSplotComparisons` *plot model comparisons*

Description

Creates a user-chosen set of plots comparing model output from a summary of multiple models, where the collection was created using the `SSsummarize` function.

Usage

```
SSplotComparisons(summaryoutput, subplots = 1:20, plot = TRUE,
  print = FALSE, png = print, pdf = FALSE, models = "all",
  endyrvec = "default", indexfleets = NULL, indexUncertainty = FALSE,
  indexQlabel = TRUE, indexQdigits = 4, indexSEvec = "default",
  indexPlotEach = FALSE, labels = c("Year", "Spawning biomass (mt)",
  "Spawning depletion", "Age-0 recruits (1,000s)", "Recruitment deviations",
  "Index", "Log index", "SPR ratio", "Density", "Management target",
  "Minimum stock size threshold", "Spawning output (eggs)", "Harvest rate"),
  col = "default", shadecol = "default", pch = "default", lty = 1,
  lwd = 2, spacepoints = 10, staggerpoints = 1, xlim = "default",
  ylimAdj = 1, xaxs = "r", yaxs = "r", type = "o", uncertainty = TRUE,
  shadealpha = 0.1, legend = TRUE, legendlabels = "default",
  legendloc = "topright", legendorder = "default", legendncol = 1,
  sprtarg = NULL, btarg = NULL, minbthresh = NULL, pwidth = 7,
  pheight = 7, punits = "in", res = 300, ptsize = 12, cex.main = 1,
```

```
plotdir = NULL, densitynames = c("SPB_Virgin", "R0"),
densityxlabs = "default", densityscalex = 1, densityscaley = 1,
densityadjust = 1, densitysymbols = TRUE, densitytails = TRUE,
densitylwd = 1, fix0 = TRUE, new = TRUE, add = FALSE, par = list(mar
= c(5, 4, 1, 1) + 0.1), verbose = TRUE, mcmcVec = "default")
```

Arguments

`summaryoutput`	List created by `SSsummarize`
`subplots`	Vector of subplots to be created.
`plot`	Plot to active plot device?
`print`	Send plots to PNG files in directory specified by `plotdir`?
`png`	Has same result as `print`, included for consistency with `SS_plots`.
`pdf`	Write output to PDF file? Can't be used in conjunction with `png` or `print`.
`models`	Optional subset of the models described in `summaryoutput`. Either "all" or a vector of numbers indicating columns in summary tables.
`endyrvec`	Optional single year or vector of years representing the final year of values to show for each model. By default it is set to the ending year specified in each model.
`indexfleets`	Vector of fleets for each model for which to compare indices of abundance. Only necessary if any model has more than one index.
`indexUncertainty`	Show uncertainty intervals on index data? Default=FALSE because if models have any extra standard deviations added, these intervals may differ across models.
`indexQlabel`	Add catchability to legend in plot of index fits (TRUE/FALSE)?
`indexQdigits`	Number of significant digits for catchability in legend (if `indexQlabel=TRUE`)
`indexSEvec`	Optional replacement for the SE values in summaryoutput$indices to deal with the issue of differing uncertainty by models described above.
`indexPlotEach`	TRUE plots the observed index for each model with colors, or FALSE just plots observed once in black dots.
`labels`	Vector of labels for plots (titles and axis labels)
`col`	Optional vector of colors to be used for lines. Input 'default' makes use of `rich.colors.short` function.
`shadecol`	Optional vector of colors to be used for shading uncertainty intervals. Input 'default' makes use of `rich.colors.short` function with alpha transparency.
`pch`	Optional vector of plot character values
`lty`	Optional vector of line types
`lwd`	Optional vector of line widths
`spacepoints`	Number of years between points shown on top of lines (for long timeseries, points every year get mashed together)
`staggerpoints`	Number of years to stagger the first point (if `spacepoints > 1`) for each line (so that adjacent lines have points in different years)

`xlim`	Optional x limits
`ylimAdj`	Multiplier for ylim parameter. Allows additional white space to fit legend if necessary. Default=1.
`xaxs`	Choice of xaxs parameter (see ?par for more info)
`yaxs`	Choice of yaxs parameter (see ?par for more info)
`type`	Type parameter passed to points (default 'o' overplots points on top of lines)
`uncertainty`	Show plots with uncertainty intervals
`shadealpha`	Transparency parameter used to make default shadecol values (see ?rgb for more info)
`legend`	Add a legend?
`legendlabels`	Optional vector of labels to include in legend. Default is 'model1','model2',etc.
`legendloc`	Location of legend. See ?legend for more info.
`legendorder`	Optional vector of model numbers that can be used to have the legend display the model names in an order that is different than that which is represented in the summary input object.
`legendncol`	Number of columns for the legend.
`btarg`	Target biomass value at which to show a line (set to 0 to remove)
`minbthresh`	Minimum biomass threshhold at which to show a line (set to 0 to remove)
`sprtarg`	Target value for SPR-ratio where line is drawn in the SPR plots and phase plot.
`pwidth`	Width of plot written to PNG file
`pheight`	Height of plot written to PNG file
`punits`	Units for PNG file
`res`	Resolution for PNG file
`ptsize`	Point size for PNG file
`cex.main`	Character expansion for plot titles
`plotdir`	Directory where PNG or PDF files will be written. By default it will be the directory where the model was run.
`densitynames`	Vector of names (or subset of names) of parameters or derived quantities contained in summaryoutput$pars$Label or summaryoutput$quants$Label for which to make density plots
`densityxlabs`	Optional vector of x-axis labels to use in the density plots (must be equal in length to the printed vector of quantities that match the `densitynames` input)
`densityscalex`	Scalar for upper x-limit in density plots (values below 1 will cut off the right tail to provide better contrast among narrower distributions
`densityscaley`	Scalar for upper y-limit in density plots (values below 1 will cut off top of highest peaks to provide better contrast among broader distributions
`densityadjust`	Multiplier on bandwidth of kernel in density function used for smoothing MCMC posteriors. See 'adjust' in ?density for details.
`densitysymbols`	Add symbols along lines in density plots. Quantiles are `c(0.025,0.1,0.25,0.5,0.75,0.9,0.975)`.
`densitytails`	Shade tails outside of 95% interval in density plots.

`densitylwd`	Line width for density plots
`fix0`	Always include 0 in the density plots?
`new`	Create new empty plot window
`add`	Allows single plot to be added to existing figure. This needs to be combined with specific 'subplots' input to make sure only one thing gets added.
`par`	list of graphics parameter values passed to the `par` function
`verbose`	Report progress to R GUI?
`mcmcVec`	Vector of TRUE/FALSE values (or single value) indicating whether input values are from MCMC or to use normal distribution around MLE

Author(s)

Ian Taylor

See Also

`SS_plots`, `SSsummarize`, `SS_output`, `SSgetoutput`

`SSplotComps` *Plot composition data and fits.*

Description

Plot composition data and fits from Stock Synthesis output. Mult-figure plots depend on `make_multifig`.

Usage

```
SSplotComps(replist, subplots = 1:12, kind = "LEN", sizemethod = 1,
  aalyear = -1, aalbin = -1, plot = TRUE, print = FALSE,
  fleets = "all", fleetnames = "default", sexes = "all",
  datonly = FALSE, samplesizeplots = TRUE, compresidplots = TRUE,
  bub = FALSE, showyears = TRUE, showsampsize = TRUE, showeffN = TRUE,
  sampsizeline = FALSE, effNline = FALSE, minnbubble = 8,
  pntscalar = NULL, scalebubbles = FALSE, cexZ1 = 1.5, bublegend = TRUE,
  blue = rgb(0, 0, 1, 0.7), pwidth = 7, pheight = 7, punits = "in",
  ptsize = 12, res = 300, plotdir = "default", cex.main = 1,
  linepos = 1, fitbar = FALSE, do.sqrt = TRUE, smooth = TRUE,
  cohortlines = c(), labels = c("Length (cm)", "Age (yr)", "Year",
  "Observed sample size", "Effective sample size", "Proportion", "cm",
  "Frequency", "Weight", "Length", "(mt)", "(numbers x1000)",
  "Stdev (Age) (yr)", "Andre's conditional AAL plot, "), printmkt = TRUE,
  printsex = TRUE, maxrows = 6, maxcols = 6, maxrows2 = 2,
  maxcols2 = 4, rows = 1, cols = 1, andrerows = 3, fixdims = TRUE,
  fixdims2 = FALSE, maxneff = 5000, verbose = TRUE, scalebins = FALSE,
  addMeans = TRUE, ...)
```

Arguments

`replist`	list created by `SSoutput`
`subplots`	vector controlling which subplots to create
`kind`	indicator of type of plot can be "LEN", "SIZE", "AGE", "cond", "GSTAGE", "L[at]A", or "W[at]A".
`sizemethod`	if kind = "SIZE" then this switch chooses which of the generalized size bin methods will be plotted.
`aalyear`	Years to plot multi-panel conditional age-at-length fits for all length bins; must be in a "c(YYYY,YYYY)" format. Useful for checking the fit of a dominant year class, critical time period, etc. Default=-1.
`aalbin`	The length bin for which multi-panel plots of the fit to conditional age-at-length data will be produced for all years. Useful to see if growth curves are ok, or to see the information on year classes move through the conditional data. Default=-1.
`plot`	plot to active plot device?
`print`	print to PNG files?
`fleets`	optional vector to subset fleets for which plots will be made
`fleetnames`	optional vector of fleet names to put in the labels
`sexes`	which sexes to show plots for. Default="all" which will include males, females, and unsexed. This options is not fully implemented for all plots.
`datonly`	make plots of data without fits as well as data with fits?
`samplesizeplots`	make sample size plots?
`compresidplots`	make plots of residuals for fit to composition data?
`bub`	make bubble plot for numbers at age or size?
`showyears`	Add labels for years to sample size plots?
`showsampsize`	add sample sizes to plot
`showeffN`	add effective sample sizes to plot
`sampsizeline`	show line for input sample sizes on top of conditional age-at-length plots (TRUE/FALSE, still in development)
`effNline`	show line for effective sample sizes on top of conditional age-at-length plots (TRUE/FALSE, still in development)
`minnbubble`	number of unique x values before adding buffer. see ?bubble3 for more info.
`pntscalar`	This scalar defines the maximum bubble size for bubble plots. This option is still available but a better choice is to use cexZ1 which allow the same scaling throughout all plots.
`scalebubbles`	scale data-only bubbles by sample size, not just proportion within sample? Default=FALSE.
`cexZ1`	Character expansion (cex) for point associated with value of 1.
`bublegend`	Add legend with example bubble sizes to bubble plots.

`blue`	What color to use for bubble plots (default is slightly transparent blue)
`pwidth`	default width of plots printed to files in units of `punits`. Default=7.
`pheight`	default height width of plots printed to files in units of `punits`. Default=7.
`punits`	units for `pwidth` and `pheight`. Can be "px" (pixels), "in" (inches), "cm" or "mm". Default="in".
`ptsize`	point size for plotted text in plots printed to files (see help("png") in R for details). Default=12.
`res`	resolution of plots printed to files. Default=300
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`cex.main`	character expansion parameter for plot titles
`linepos`	should lines be added before points (linepos=1) or after (linepos=2)?
`fitbar`	show fit to bars instead of points
`do.sqrt`	scale bubbles based on sqrt of size vector. see ?bubble3 for more info.
`smooth`	add loess smoother to observed vs. expected index plots and input vs. effective sample size?
`cohortlines`	optional vector of birth years for cohorts for which to add growth curves to numbers at length bubble plots
`labels`	vector of labels for plots (titles and axis labels)
`printmkt`	show market categories in plot titles?
`printsex`	show gender in plot titles?
`maxrows`	maximum (or fixed) number or rows of panels in the plot
`maxcols`	maximum (or fixed) number or columns of panels in the plot
`maxrows2`	maximum number of rows for conditional age at length plots
`maxcols2`	maximum number of columns for conditional age at length plots
`rows`	number or rows to return to as default for next plots to come or for single plots
`cols`	number or cols to return to as default for next plots to come or for single plots
`andrerows`	Number of rows of Andre's conditional age-at-length plots within each page. Default=3.
`fixdims`	fix the dimensions at maxrows by maxcols or resize based on number of years of data
`fixdims2`	fix the dimensions at maxrows by maxcols in aggregate plots or resize based on number of fleets
`maxneff`	the maximum value to include on plots of input and effective sample size. Occasionally a calculation of effective N blows up to very large numbers, rendering it impossible to observe the relationship for other data. Default=5000.
`verbose`	return updates of function progress to the R GUI?
`scalebins`	Rescale expected and observed proportions by dividing by bin width for models where bins have different widths? Caution!: May not work correctly in all cases.
`addMeans`	Add parameter means in addition to medians for MCMC posterior distributions in which the median and mean differ.
`...`	additional arguments that will be passed to the plotting.

Author(s)

Ian Taylor

See Also

`SS_plots`, `make_multifig`

`SSplotData` *Timeline of presence/absence of data by type, year, and fleet.*

Description

Plot shows graphical display of what data is being used in the model. Some data types may not yet be included. Note, this is based on output from the model, not the input data file.

Usage

```
SSplotData(replist, plot = TRUE, print = FALSE, plotdir = "default",
  fleetcol = "default", datatypes = "all", fleets = "all",
  fleetnames = "default", ghost = FALSE, pwidth = 7, pheight = 7,
  punits = "in", res = 300, ptsize = 12, cex.main = 1,
  margins = c(5.1, 2.1, 2.1, 8.1), cex = 2, lwd = 12, verbose = TRUE)
```

Arguments

`replist`	list created by `SS_output`
`plot`	plot to active plot device?
`print`	print to PNG files?
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`fleetcol`	Either the string "default", or a vector of colors to use for each fleet.
`datatypes`	Either the string "all", or a vector including some subset of the following: "catch", "cpue", "lendbase", "sizedbase", "agedbase", "condbase", "ghostagedbase", "ghostcondbase", "ghostlendbase", "ladbase", "wadbase", "mnwgt", "discard", "tagdbase1", "tagdbase2".
`fleets`	Either the string "all", or a vector of numerical values, like c(1,3), listing fleets or surveys to be included in the plot.
`fleetnames`	A vector of alternative names to use in the plot. By default the parameter names in the data file are used.
`ghost`	TRUE/FALSE indicator for whether to show presence of composition data from ghost fleets (data for which the fit is shown, but is not included in the likelihood calculations).
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file

`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles
`margins`	margins of plot (passed to par() function), which may need to be increased if fleet names run off right-hand margin
`cex`	Character expansion for points showing isolated years of data
`lwd`	Line width for lines showing ranges of years of data
`verbose`	report progress to R GUI?

Author(s)

Ian Taylor, Chantel Wetzel

See Also

`SS_plots`, `SS_output`, `SS_readdat`

`SSplotDiscard` *Plot fit to discard fraction.*

Description

Plot fit to discard fraction from Stock Synthesis output file.

Usage

```
SSplotDiscard(replist, subplots = 1:2, plot = TRUE, print = FALSE,
  plotdir = "default", fleets = "all", fleetnames = "default",
  datplot = FALSE, labels = c("Year", "Discard fraction", "Total discards",
  "for"), yhi = 1, col1 = "blue", col2 = "black", pwidth = 7,
  pheight = 7, punits = "in", res = 300, ptsize = 12, cex.main = 1,
  verbose = TRUE)
```

Arguments

`replist`	List created by `SS_output`
`subplots`	Vector of which plots to make (1 = data only, 2 = with fit). If `plotdat = FALSE` then subplot 1 is not created, regardless of choice of `subplots`.
`plot`	Plot to active plot device?
`print`	Print to PNG files?
`plotdir`	Directory where PNG files will be written. by default it will be the directory where the model was run.
`fleets`	Optional vector to subset fleets for which plots will be made

`fleetnames`	Optional replacement for fleenames used in data file
`datplot`	Make data-only plot of discards? This can override the choice of `subplots`.
`labels`	Vector of labels for plots (titles and axis labels)
`yhi`	Maximum y-value to include in plot (all data included regardless). Default = 1.
`col1`	First color to use in plot (for expected values)
`col2`	Second color to use in plot (for observations and intervals)
`pwidth`	Width of plot written to PNG file
`pheight`	Height of plot written to PNG file
`punits`	Units for PNG file
`res`	Resolution for PNG file
`ptsize`	Point size for PNG file
`cex.main`	Character expansion for plot titles
`verbose`	Report progress to R GUI?

Author(s)

Ian Taylor, Ian Stewart

See Also

`SS_plots`

`SSplotIndices` *Plot indices of abundance and associated quantities.*

Description

Plot indices of abundance and associated quantities.

Usage

```
SSplotIndices(replist, subplots = 1:9, plot = TRUE, print = FALSE,
  fleets = "all", fleetnames = "default", smooth = TRUE, add = FALSE,
  datplot = FALSE, labels = c("Year", "Index", "Observed index",
  "Expected index", "Log index", "Log observed index", "Log expected index",
  "Standardized index", "Catchability (Q)", "Time-varying catchability",
  "Vulnerable biomass", "Catchability vs. vulnerable biomass"),
  col1 = "default", col2 = "default", col3 = "blue", col4 = "red",
  pch1 = 21, pch2 = 16, cex = 1, bg = "white", legend = TRUE,
  legendloc = "topright", seasnames = NULL, pwidth = 7, pheight = 7,
  punits = "in", res = 300, ptsize = 12, cex.main = 1, addmain = TRUE,
  plotdir = "default", minyr = NULL, maxyr = NULL, verbose = TRUE)
```

Arguments

`replist`	list created by `SS_output`
`subplots`	vector controlling which subplots to create
`plot`	plot to active plot device?
`print`	print to PNG files?
`fleets`	optional vector to subset fleets for which plots will be made
`fleetnames`	optional replacement for fleenames used in data file
`smooth`	add smoothed line to plots of observed vs. expected sample sizes
`add`	add to existing plot (not yet implemented)
`datplot`	make plot of data only?
`labels`	vector of labels for plots (titles and axis labels)
`col1`	vector of colors for points in each season for time series plot. Default is red for single season models and a rainbow using the rich.colors.short function for multiple seasons.
`col2`	vector of colors for points in each season for obs. vs. exp. plot. Default is blue for single season models and a rainbow using the rich.colors.short function for multiple seasons.
`col3`	color of line showing expected index in time series plot. Default is blue.
`col4`	color of smoother shown in obs. vs. exp. plots. Default is red.
`pch1`	single value or vector of plotting characters (pch parameter) for time-series plots of index fit. Default=21.
`pch2`	single value or vector of plotting characters (pch parameter) for sample size plots of index fit. Default=16.
`cex`	character expansion factor for points showing observed values. Default=1.
`bg`	Background color for points with pch=21.
`legend`	add a legend to seasonal colors (only for seasonal models)
`legendloc`	add a legend to seasonal colors (default is "topright")
`seasnames`	optional vector of names for each season to replace defaults if a legend is used
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles
`addmain`	switch which allows the plot title to be left off
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`minyr`	First year to show in plot (for zooming in on a subset of values)
`maxyr`	Last year to show in plot (for zooming in on a subset of values)
`verbose`	report progress to R GUI?

Author(s)

Ian Stewart, Ian Taylor

See Also

`SS_plots`, `SS_output`

`SSplotMCMC_ExtraSelex` *Plot uncertainty around chosen selectivity ogive from MCMC.*

Description

Plot uncertainty in selectivity from an MCMC output for whichever fleet/year was chosen in the optional extra "more stddev reporting"

Usage

```
SSplotMCMC_ExtraSelex(post, add = FALSE, nsexes = 1, shift = 0,
  fleetname = "default", col = "blue")
```

Arguments

`post`	A data frame containing either derived_posteriors.sso or a good subset of it. This can be an element of the list created by the the `SSgetMCMC` function.
`add`	TRUE/FALSE option to add results to an existing plot.
`nsexes`	Number of sexes in the model (should match model values but is only used in the title).
`shift`	Optional adjustment to the x values to avoid overlap of intervals when overplotting on an existing plot.
`fleetname`	Optional input to make the title better. Default will be something like "Fleet 1", using the numbering from the model.
`col`	Color for points and lines.

Author(s)

Ian Taylor

`SSplotMnwt` *Plot mean weight data and fits.*

Description

Plot mean weight data and fits from Stock Synthesis output. Intervals are based on T-distributions as specified in model.

Usage

```
SSplotMnwt(replist, subplots = 1:2, ymax = NULL, plot = TRUE,
  print = FALSE, fleets = "all", fleetnames = "default",
  datplot = FALSE, labels = c("Year", "discard", "retained catch",
  "whole catch", "Mean individual body weight (kg)", "Mean weight in", "for"),
  col1 = "blue", col2 = "black", pwidth = 7, pheight = 7,
  punits = "in", res = 300, ptsize = 12, cex.main = 1,
  plotdir = "default", verbose = TRUE)
```

Arguments

`replist`	list created by `SS_output`
`ymax`	Optional input to override default ymax value.
`subplots`	Vector of which plots to make (1 = data only, 2 = with fit). If `plotdat = FALSE` then subplot 1 is not created, regardless of choice of `subplots`.
`plot`	plot to active plot device?
`print`	print to PNG files?
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`fleets`	optional vector to subset fleets for which plots will be made
`fleetnames`	optional replacement for fleenames used in data file
`datplot`	Make data-only plot of discards? This can override the choice of `subplots`.
`labels`	vector of labels for plots (titles and axis labels)
`col1`	first color to use in plot (for expected values)
`col2`	second color to use in plot (for observations and intervals)
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles
`verbose`	report progress to R GUI?

Author(s)

Ian Taylor, Ian Stewart

See Also

`SS_plots`, `SS_output`

`SSplotMovementMap` *Show movement rates on a map.*

Description

Make a map with colored spatial cells and add arrows representing movement rates between cells.

Usage

```
SSplotMovementMap(replist = NULL, xlim, ylim, polygonlist, colvec,
  land = "grey", xytable = NULL, moveage = 5, moveseas = 1,
  lwdscale = 5, legend = TRUE, title = NULL, areanames = NULL,
  cex = 1)
```

Arguments

`replist`	optional list created by `SS_output`
`xlim`	range of longitude values in the map
`ylim`	range of latitude values in the map
`polygonlist`	a list of data frames, each with two columns representing the longitude and latitude values of the colored polygons. The order of elements in the list should match the numbering of areas in the SS model.
`colvec`	vector of colors for each polygon (if `replist` is provided)
`land`	color of landmasses in the map
`xytable`	data frame of latitude and longitude values which will be connected by the arrows representing movement rates. The order should match the order of areas in `polygonlist` and in the SS model. Not necessary if no arrows are shown on the map.
`moveage`	age for which movemement rates will be represented
`moveseas`	season for which movement rates will be represented
`lwdscale`	scaling factor for arrows in the plot. The largest rate of movement shown will be scaled to have a line width equal to this value.
`legend`	add a legend to show the movement rate associated with the widest arrows
`title`	optional title to be added above map
`areanames`	optional vector of names to be shown on map at coordinates matching xytable values
`cex`	character expansion to apply to text shown by areanames (if used)

Note

Inspired by plots of MULTIFAN-CL movement patterns presented by Adam Langley

Author(s)

Ian Taylor

See Also

`SS_output`, `SSplotMovementRates`, `IOTCmove`

`SSplotMovementRates` *Plot movement rates from model output*

Description

Plots estimated movement rates in final year for each area/seaon with movement as reported in Report.sso. If movement is time-varying, an additional figure shows pattern across years.

Usage

```
SSplotMovementRates(replist, plot = TRUE, print = FALSE, subplots = 1:2,
  plotdir = "default", colvec = "default", ylim = "default",
  legend = TRUE, legendloc = "topleft", moveseas = "all",
  min.move.age = 0.5, pwidth = 7, pheight = 7, punits = "in",
  res = 300, ptsize = 12, cex.main = 1, verbose = TRUE)
```

Arguments

`replist`	list created by `SS_output`
`plot`	plot to active plot device?
`print`	print to PNG files?
`subplots`	which subplots to create
`plotdir`	where to put the plots (uses model directory by default)
`colvec`	vector of colors for each movement rate in the plot
`ylim`	optional input for y range of the plot. By default plot ranges from 0 to 10% above highest movement rate (not including fish staying in an area).
`legend`	add a legend designating which color goes with which pair of areas?
`legendloc`	location passed to legend function (if used)
`moveseas`	choice of season for which movemement rates are shown
`min.move.age`	Minimum age of movement (in future will come from Report file)
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file

`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	Character expansion parameter for plot titles
`verbose`	Print information on function progress.

Author(s)

Ian Taylor

See Also

`SS_output`, `SSplotMovementRates`, `IOTCmove`

Examples

```
## Not run:
    SSplotMovementRates(myreplist)

## End(Not run)
```

`SSplotNumbers` *Plot numbers-at-age related data and fits.*

Description

Plot numbers-at-age related data and fits from Stock Synthesis output. Plots include bubble plots, mean age, equilibrium age composition, sex-ratio, and ageing imprecision patterns.

Usage

```
SSplotNumbers(replist, subplots = 1:9, plot = TRUE, print = FALSE,
  areas = "all", areanames = "default", areacols = "default",
  pntscalar = 2.6, bublegend = TRUE, period = c("B", "M"), add = FALSE,
  labels = c("Year", "Age", "True age (yr)", "SD of observed age (yr)",
  "Mean observed age (yr)", "Mean age (yr)", "mean age in the population",
  "Ageing imprecision", "Numbers at age at equilibrium",
  "Equilibrium age distribution", "Sex ratio of numbers at age (males/females)",
  "Length", "Mean length (cm)", "mean length (cm) in the population",
  "expected numbers at age", "Beginning of year", "Middle of year",
  "expected numbers at length",
  "Sex ratio of numbers at length (males/females)",
  "Sex ratio of numbers at length (females/males)"), pwidth = 7,
  pheight = 7, punits = "in", res = 300, ptsize = 12, cex.main = 1,
  plotdir = "default", verbose = TRUE)
```

Arguments

`replist`	list created by `SSoutput`
`subplots`	vector controlling which subplots to create
`plot`	plot to active plot device?
`print`	print to PNG files?
`areas`	optional subset of areas to plot for spatial models
`areanames`	names for areas. Default is to use Area1, Area2,...
`areacols`	vector of colors by area
`pntscalar`	maximum bubble size for bubble plots; each plot scaled independently based on this maximum size and the values plotted. Often some plots look better with one value and others with a larger or smaller value. Default=2.6
`bublegend`	Add legend with example bubble sizes?
`period`	indicator of whether to make plots using numbers at age just from the beginning ("B") or middle of the year ("M") (new option starting with SSv3.11)
`add`	add to existing plot? (not yet implemented)
`labels`	vector of labels for plots (titles and axis labels)
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`verbose`	report progress to R GUI?

Author(s)

Ian Stewart, Ian Taylor

See Also

`SS_output`, `SS_plots`

SSplotPars *Plot distributions of priors, posteriors, and estimates.*

Description

Make multi-figure plots of prior, posterior, and estimated asymptotic parameter distributions. MCMC not required to make function work.

Usage

```
SSplotPars(dir = "c:/path/", repfile = "Report.sso",
  xlab = "Parameter value", ylab = "Density", postfile = "posteriors.sso",
  showpost = TRUE, showprior = TRUE, showmle = TRUE, showinit = TRUE,
  showrecdev = TRUE, priorinit = TRUE, priorfinal = TRUE,
  showlegend = TRUE, fitrange = FALSE, xaxs = "i", xlim = NULL,
  ylim = NULL, verbose = TRUE, nrows = 3, ncols = 3, new = TRUE,
  pdf = FALSE, pwidth = 7, pheight = 7, punits = "in", ptsize = 12,
  returntable = FALSE, strings = c(), exact = FALSE, newheaders = NULL,
  burn = 0, thin = 1, ctlfile = "control.ss_new")
```

Arguments

`dir`	Directory where all files are located.
`repfile`	Name of report file. Default="Report.sso".
`xlab`	Label on horizontal axis.
`ylab`	Label on vertical axis.
`postfile`	Name of MCMC posteriors file (not required). Default="posteriors.sso".
`showpost`	Show posterior distribution as bar graph? Default=TRUE.
`showprior`	Show prior distribution as black line? Default=TRUE.
`showmle`	Show MLE estimate and asymptotic variance estimate with blue lines? Default=TRUE.
`showinit`	Show initial value as red triangle? Default=TRUE.
`showrecdev`	Include recdevs in the plot? Default=TRUE.
`priorinit`	TRUE/FALSE for prior probability at initial value (not implemented).
`priorfinal`	TRUE/FALSE for prior probability at final value (not implemented).
`showlegend`	Show the legend? Default=TRUE.
`fitrange`	Fit range tightly around MLE & posterior distributions, instead of full parameter range? Default=FALSE.
`xaxs`	Parameter input for x-axis. See `?par` for more info. Default="i".
`xlim`	Optional x-axis limits to be applied to all plots. Otherwise, limits are based on the model results. Default=NULL.

`ylim`	Optional y-axis limits to be applied to all plots. Otherwise, limits are based on the model results. Default=NULL.
`verbose`	Controls amount of text output (maybe). Default=TRUE.
`nrows`	How many rows in multi-figure plot. Default=3.
`ncols`	How many columns in multi-figure plot. Default=3.
`new`	Open new window for plotting? Default=TRUE.
`pdf`	Write to PDF file instead of R GUI? Default=FALSE.
`pwidth`	Default width of plots printed to files in units of `punits`. Default=7.
`pheight`	Default height width of plots printed to files in units of `punits`. Default=7.
`punits`	Units for `pwidth` and `pheight`. Can be "px" (pixels), "in" (inches), "cm" or "mm". Default="in".
`ptsize`	Point size for plotted text in plots printed to files (see help("png") in R for details). Default=12.
`returntable`	Return table of parameter info? Default=FALSE.
`strings`	Subset parameters included in the plot using substring from parameter names (i.e. "SR" will get "SR_R0" and "SR_steep" if they are both estimated quantities in this model). Default=c().
`exact`	Should strings input match parameter names exactly? Otherwise substrings are allowed. Default=FALSE.
`newheaders`	Optional vector of headers for each panel to replace the parameter names. Default=NULL.
`burn`	Additional burn-in applied to MCMC posteriors. Default=0.
`thin`	Additional thinning applied to MCMC posteriors. Default=1.
`ctlfile`	Specify control file to get min and max recdev values (otherwise assumed to be -5 and 5). Default="control.ss_new".

Author(s)

Ian Taylor

Examples

```
## Not run:
pars <- SSplotPars(dir='c:/SS/Simple/')

# strings can be partial match
pars <- SSplotPars(dir='c:/SS/Simple/',strings=c("steep"))

## End(Not run)
```

`SSplotProfile` *Plot likelihood profile results*

Description

Makes a plot of change in negative-log-likelihood for each likelihood component that contributes more than some minimum fraction of change in total.

Usage

```
SSplotProfile(summaryoutput, plot = TRUE, print = FALSE, models = "all",
  profile.string = "steep", profile.label = "Spawner-recruit steepness (h)",
  ylab = "Change in -log-likelihood", components = c("TOTAL", "Catch",
  "Equil_catch", "Survey", "Discard", "Mean_body_wt", "Length_comp", "Age_comp",
  "Size_at_age", "SizeFreq", "Morphcomp", "Tag_comp", "Tag_negbin",
  "Recruitment", "Forecast_Recruitment", "Parm_priors", "Parm_softbounds",
  "Parm_devs", "F_Ballpark", "Crash_Pen"), component.labels = c("Total",
  "Catch", "Equilibrium catch", "Index data", "Discard", "Mean body weight",
  "Length data", "Age data", "Size-at-age data", "Generalized size data",
  "Morph composition data", "Tag recapture distribution", "Tag recapture total",
  "Recruitment", "Forecast recruitment", "Priors", "Soft bounds",
  "Parameter deviations", "F Ballpark", "Crash penalty"), minfraction = 0.01,
  sort.by.max.change = TRUE, col = "default", pch = "default", lty = 1,
  lty.total = 1, lwd = 2, lwd.total = 3, cex = 1, cex.total = 1.5,
  xlim = "default", ymax = "default", xaxs = "r", yaxs = "r",
  type = "o", legend = TRUE, legendloc = "topright", pwidth = 7,
  pheight = 7, punits = "in", res = 300, ptsize = 12, cex.main = 1,
  plotdir = NULL, verbose = TRUE)
```

Arguments

`summaryoutput`	List created by the function `SSsummarize`.
`plot`	Plot to active plot device?
`print`	Print to PNG files?
`models`	Optional subset of the models described in `summaryoutput`. Either "all" or a vector of numbers indicating columns in summary tables.
`profile.string`	Character string used to find parameter over which the profile was conducted. Needs to match substring of one of the SS parameter labels found in the Report.sso file. For instance, the default input 'steep' matches the parameter 'SR_BH_steep'.
`profile.label`	Label for x-axis describing the parameter over which the profile was conducted.
`ylab`	Label for y-axis. Default is "Change in -log-likelihood".
`components`	Vector of likelihood components that may be included in plot. List is further refined by any components that are not present in model or have little change over range of profile (based on limit `minfraction`). Hopefully this doesn't need to be changed.

`component.labels`	Vector of labels for use in the legend that matches the vector in `components`.
`minfraction`	Minimum change in likelihood (over range considered) as a fraction of change in total likelihood for a component to be included in the figure.
`sort.by.max.change`	Switch giving option to sort components in legend in order of maximum amount of change in likelihood (over range considered). Default=TRUE.
`col`	Optional vector of colors for each line.
`pch`	Optional vector of plot characters for the points.
`lty`	Line total for the liklihood components.
`lty.total`	Line type for the total likelihood.
`lwd`	Line width for the liklihood components.
`lwd.total`	Line width for the total likelihood.
`cex`	Character expansion for the points representing the likelihood components.
`cex.total`	Character expansion for the points representing the total likelihood.
`xlim`	Range for x-axis. Change in likelihood is calculated relative to values within this range.
`ymax`	Maximum y-value. Default is 10% greater than largest value plotted.
`xaxs`	The style of axis interval calculation to be used for the x-axis (see ?par for more info)
`yaxs`	The style of axis interval calculation to be used for the y-axis (see ?par for more info).
`type`	Line type (see ?plot for more info).
`legend`	Include legend?
`legendloc`	Location of legend (see ?legend for more info).
`pwidth`	Width of plot written to PNG file
`pheight`	Height of plot written to PNG file
`punits`	Units for PNG file
`res`	Resolution for PNG file
`ptsize`	Point size for PNG file
`cex.main`	Character expansion for plot titles
`plotdir`	Directory where PNG files will be written. by default it will be the directory where the model was run.
`verbose`	Return updates of function progress to the R GUI? (Doesn't do anything yet.)

Note

Someday the function `SS_profile` will be improved and made to work directly with this plotting function, but they don't yet work well together. Thus, even if `SS_profile` is used, the output should be read using `SSgetoutput` or by multiple calls to `SS_output`.

Author(s)

Ian Taylor, Ian Stewart

See Also

`SSsummarize`, `SS_profile`, `SS_output`, `SSgetoutput`

`SSplotRecdevs` *Plot recruitment deviations*

Description

Plot recruitment deviations and associated quantities including derived measures related to bias adjustment.

Usage

```
SSplotRecdevs(replist, subplots = 1:3, plot = TRUE, print = FALSE,
  add = FALSE, uncertainty = TRUE, forecastplot = FALSE, col1 = "black",
  col2 = "blue", col3 = "green3", col4 = "red", legendloc = "topleft",
  labels = c("Year", "Asymptotic standard error estimate",
  "Log recruitment deviation",
  "Bias adjustment fraction, 1 - stddev^2 / sigmaR^2"), pwidth = 7,
  pheight = 7, punits = "in", res = 300, ptsize = 12, cex.main = 1,
  plotdir = "default", verbose = TRUE)
```

Arguments

`replist`	list created by `SSoutput`
`subplots`	vector controlling which subplots to create
`plot`	plot to active plot device?
`print`	print to PNG files?
`add`	add to existing plot (not yet implemented)
`uncertainty`	include plots showing uncertainty?
`forecastplot`	include points from forecast years?
`col1`	first color used
`col2`	second color used
`col3`	third color used
`col4`	fourth color used
`legendloc`	location of legend. see ?legend for more info
`labels`	vector of labels for plots (titles and axis labels)
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file

`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`verbose`	report progress to R GUI?

Author(s)

Ian Taylor, Ian Stewart

See Also

`SS_plots`, `SS_fitbiasramp`

`SSplotRecdist` *Plot of recruitment distribution among areas and seasons*

Description

Image plot shows fraction of recruitment in each combination of area and season. This is based on the RECRUITMENT_DIST section of the Report.sso file.

Usage

```
SSplotRecdist(replist, plot = TRUE, print = FALSE, areanames = NULL,
  seasnames = NULL, xlab = "", ylab = "",
  main = "Distribution of recruitment by area and season",
  plotdir = "default", pwidth = 7, pheight = 7, punits = "in",
  res = 300, ptsize = 12, cex.main = 1, verbose = TRUE)
```

Arguments

`replist`	list created by `SS_output`
`plot`	plot to active plot device?
`print`	print to PNG files?
`areanames`	optional vector to replace c("Area1","Area2",...)
`seasnames`	optional vector to replace c("Season1","Season2",...)
`xlab`	optional x-axis label (if the area names aren't informative enough)
`ylab`	optional y-axis label (if the season names aren't informative enough)
`main`	title for plot
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.

`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles
`verbose`	report progress to R GUI?

Author(s)

Ian Taylor

See Also

`SS_plots`, `SSplotRecdevs`

`SSplotRetroRecruits` *Make squid plot of retrospectives of recruitment deviations.*

Description

Inspired by Jim Ianelli and named by Sean Cox, the squid plot is a way to examine retrospective patterns in estimation of recruitment deviations.

Usage

```
SSplotRetroRecruits(retroSummary, endyrvec, cohorts, ylim = NULL,
  uncertainty = FALSE, labels = c("Recruitment deviation",
  "Recruitment (billions)", "relative to recent estimate", "Age"),
  main = "Retrospective analysis of recruitment deviations",
  mcmcVec = FALSE, devs = TRUE, relative = FALSE, labelyears = TRUE,
  legend = FALSE, leg.ncols = 4)
```

Arguments

`retroSummary`	List object created by `SSsummarize` that summarizes the results of a set of retrospective analysis models.ss
`endyrvec`	Vector of years representing the final year of values to show for each model.
`cohorts`	Which cohorts to show in plot.
`ylim`	Limits of y-axis.
`uncertainty`	Show uncertainty intervals around lines? (This can get a bit busy.)
`labels`	Vector of plot labels.
`main`	Title for plot.

`mcmcVec`	Either vector of TRUE/FALSE values indicating which models use MCMC. Or single value applied to all models.
`devs`	Plot deviations instead of absolute recruitment values?
`relative`	Show deviations relative to most recent estimate or relative to 0.
`labelyears`	Label cohorts with text at the end of each line?
`legend`	Add a legend showing which color goes with which line (as alternative to `labelyears`).
`leg.ncols`	Number of columns for the legend.

Author(s)

Ian Taylor

References

Ianelli et al. (2011) Assessment of the walleye pollock stock in the Eastern Bering Sea. `http://www.afsc.noaa.gov/REFM/docs/2011/EBSpollock.pdf`. (Figure 1.31, which is on an absolute, rather than log scale.)

See Also

`SSsummarize`

Examples

```
## Not run:
 # run retrospective analysis
 SS_doRetro(olddir='2013hake_12',years=0:-10)
 # read in output
 retroModels <- SSgetoutput(dirvec=paste('retrospectives/retro',-10:0,sep=''))
 # summarize output
 retroSummary <- SSsummarize(retroModels)

 # set the ending year of each model in the set
 endyrvec <- retroModels[[1]]$endyr-10:0
 # make comparison plot
 pdf('retrospectives/retrospective_comparison_plots.pdf')
 SSplotComparisons(retroSummary,endyrvec=endyrvec,new=FALSE)
 dev.off()

 # make Squid Plot of recdev retrospectives
 pdf('retrospectives/retrospective_dev_plots.pdf',width=7,height=10)
 par(mfrow=c(2,1))
 # first scaled relative to most recent estimate
 SSplotRetroRecruits(retroSummary, endyrvec=endyrvec, cohorts=1999:2012,
                     relative=TRUE, legend=FALSE)
 # second without scaling
 SSplotRetroDevs(retroSummary, endyrvec=endyrvec, cohorts=1999:2012,
                 relative=FALSE, legend=FALSE)
 dev.off()

## End(Not run)
```

`SSplotSelex` *Plot selectivity*

Description

Plot selectivity, including retention and other quantities, with additional plots for time-varying selectivity.

Usage

```
SSplotSelex(replist, infotable = NULL, fleets = "all",
  fleetnames = "default", sizefactors = c("Lsel"), agefactors = c("Asel",
  "Asel2"), years = "endyr", season = 1, sexes = "all",
  selexlines = 1:6, subplot = 1:25, skipAgeSelex10 = TRUE, plot = TRUE,
  print = FALSE, add = FALSE, labels = c("Length (cm)", "Age (yr)",
  "Year", "Selectivity", "Retention", "Discard mortality"), col1 = "red",
  col2 = "blue", lwd = 2, fleetcols = "default", fleetpch = "default",
  fleetlty = "default", spacepoints = 5, staggerpoints = 1,
  legendloc = "bottomright", pwidth = 7, pheight = 7, punits = "in",
  res = 300, ptsize = 12, cex.main = 1, showmain = TRUE,
  plotdir = "default", verbose = TRUE)
```

Arguments

`replist`	List created by `SS_output`
`fleets`	Optional vector to subset fleets for which to make plots
`infotable`	Optional table of information controlling appearance of plot and legend. Is produced as output and can be modified and entered as input.
`fleetnames`	Optional replacement for fleenames used in data file
`sizefactors`	Which elements of the factors column of SIZE_SELEX should be included in plot of selectivity across multiple fleets?
`agefactors`	Which elements of the factors column of AGE_SELEX should be included in plot of selectivity across multiple fleets?
`years`	Which years for selectivity are shown in multi-line plot (default = last year of model).
`season`	Which season (if seasonal model) for selectivity shown in multi-line plot (default = 1).
`sexes`	Optional vector to subset genders for which to make plots (1=females, 2=males)
`selexlines`	Vector to select which lines get plotted. values are 1. Selectivity, 2. Retention, 3. Discard mortality, 4. Keep = Sel*Ret, 5. Dead = Sel*(Ret+(1-Ret)*Mort).
`subplot`	Vector controlling which subplots to create
`skipAgeSelex10`	Exclude plots for age selectivity type 10 (selectivity = 1.0 for all ages beginning at age 1)?

`lwd`	Line widths for plots
`fleetcols`	Optional vector of colors for each fleet (in multi-fleet plots)
`fleetpch`	Optional vector of plot characters for each fleet (in multi-fleet plots)
`fleetlty`	Optional vector of line types for each fleet (in multi-fleet plots)
`spacepoints`	number of years between points shown on top of lines (for long timeseries, points every year get mashed together)
`staggerpoints`	number of years to stagger the first point (if `spacepoints > 1`) for each line (so that adjacent lines have points in different years)
`legendloc`	location of legend. See ?legend for more info.
`plot`	Plot to active plot device?
`print`	Print to PNG files?
`add`	Add to existing plot (not yet implemented)
`labels`	vector of labels for plots (titles and axis labels)
`col1`	color for female growth curve
`col2`	color for male growth curve
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles
`showmain`	Include main title at top of plot?
`plotdir`	Directory where PNG files will be written. By default it will be the directory where the model was run.
`verbose`	report progress to R GUI?

Author(s)

Ian Stewart, Ian Taylor

See Also

`SS_plots`, `SS_output`

`SSplotSpawnrecruit` *Plot spawner-recruit curve.*

Description

Plot spawner-recruit curve based on output from Stock Synthesis model.

Usage

```
SSplotSpawnrecruit(replist, subplot = 1:2, add = FALSE, plot = TRUE,
  print = FALSE, xlim = NULL, ylim = NULL,
  xlab = "Spawning biomass (mt)", ylab = "Recruitment (1,000s)",
  bioscale = "default", plotdir = "default", pwidth = 7, pheight = 7,
  punits = "in", res = 300, ptsize = 12, cex.main = 1, verbose = TRUE,
  line1 = "blue", line2 = "green3", line3 = "black", minyr = "default",
  textmindev = 0.5, ptcol = "red", virg = TRUE, init = FALSE,
  forecast = FALSE)
```

Arguments

`replist`	list created by `SS_output`
`subplot`	vector of which subplots to show. 1=plot without labels, 2=plot with year labels.
`add`	add to existing plot?
`plot`	plot to active plot device?
`print`	print to PNG files?
`xlim`	optional control of x range
`ylim`	optional control of y range
`xlab`	x-axis label
`ylab`	y-axis label
`bioscale`	multiplier on spawning biomass, set to 0.5 for single-sex models
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles
`verbose`	report progress to R GUI?
`line1`	first line color
`line2`	second line color

`line3`	third line color
`minyr`	minimum year of recruitment deviation to show in plot
`textmindev`	minimum recruitment deviation for label to be added so only extreme devs are labeled (labels are added to first and last years as well). Default=0.7.
`ptcol`	point color
`virg`	add point for equilibrium conditions (x=B0,y=R0)
`init`	add point for initial conditions (x=B1,y=R1)
`forecast`	include forecast years in the curve?

Author(s)

Ian Stewart, Ian Taylor

See Also

`SS_plots`, `SS_output`

`SSplotSPR` *Plot SPR quantities.*

Description

Plot SPR quantities, including 1-SPR and phase plot.

Usage

```
SSplotSPR(replist, add = FALSE, plot = TRUE, print = FALSE,
  uncertainty = TRUE, subplots = 1:4, forecastplot = FALSE,
  col1 = "black", col2 = "blue", col3 = "green3", col4 = "red",
  sprtarg = "default", btarg = "default", labels = c("Year", "SPR",
  "1-SPR"), pwidth = 7, pheight = 7, punits = "in", res = 300,
  ptsize = 12, cex.main = 1, plotdir = "default", verbose = TRUE)
```

Arguments

`replist`	list created by `SSoutput`
`add`	add to existing plot (not yet implemented)
`plot`	plot to active plot device?
`print`	print to PNG files?
`uncertainty`	include plots showing uncertainty?
`subplots`	vector controlling which subplots to create
`forecastplot`	Include forecast years in plot?
`col1`	first color used

`col2`	second color used
`col3`	third color used
`col4`	fourth color used
`sprtarg`	F/SPR proxy target. "default" chooses based on model output.
`btarg`	target depletion to be used in plots showing depletion. May be omitted by setting to NA. "default" chooses based on model output.
`labels`	vector of labels for plots (titles and axis labels)
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`verbose`	report progress to R GUI?

Author(s)

Ian Stewart, Ian Taylor

See Also

`SS_plots`, `SS_output`

`SSplotSummaryF` *Plot the summary F (or harvest rate).*

Description

Plots the summary F (or harvest rate) as set up in the starter file Needs a lot of work to be generalized

Usage

```
SSplotSummaryF(replist, yrs = "all", Ftgt = NA,
  ylab = "Summary Fishing Mortality", plot = TRUE, print = FALSE,
  plotdir = "default", verbose = TRUE, uncertainty = TRUE, pwidth = 7,
  pheight = 7, punits = "in", res = 300, ptsize = 12)
```

Arguments

`replist`	List created by `SS_output`
`yrs`	Which years to include.
`Ftgt`	Target F where horizontal line is shown.
`ylab`	Y-axis label.
`plot`	Plot to active plot device?
`print`	Print to PNG files?
`plotdir`	Directory where PNG files will be written. By default it will be the directory where the model was run.
`verbose`	Verbose output to R console?
`uncertainty`	Show 95% uncertainty intervals around point estimates?
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file

Author(s)

Allan Hicks

See Also

`SSplotTimeseries`, ~~~

`SSplotTags` *Plot tagging data and fits*

Description

Plot observed and expected tag recaptures in aggregate and by tag group.

Usage

```
SSplotTags(replist = replist, subplots = 1:8, latency = 0, rows = 1,
  cols = 1, tagrows = 3, tagcols = 3, plot = TRUE, print = FALSE,
  pntscalar = 2.6, minnbubble = 8, pwidth = 7, pheight = 7,
  punits = "in", ptsize = 12, res = 300, cex.main = 1, col1 = rgb(0,
  0, 1, 0.7), col2 = "red", col3 = "grey95", col4 = "grey70",
  labels = c("Year", "Frequency", "Tag Group",
  "Fit to tag recaptures by tag group",
  "Tag recaptures aggregated across tag groups",
  "Observed tag recaptures by year and tag group",
  "Residuals for tag recaptures: (obs-exp)/sqrt(exp)",
  "Observed and expected tag recaptures by year and tag group"),
  plotdir = "default", verbose = TRUE)
```

Arguments

`replist`	list created by `SS_output`
`subplots`	vector controlling which subplots to create
`latency`	period of tag mixing to exclude from plots (in future could be included in SS output)
`rows`	number or rows of panels for regular plots
`cols`	number or columns of panels for regular plots
`tagrows`	number or rows of panels for multi-panel plots
`tagcols`	number or columns of panels for multi-panel plots
`plot`	plot to active plot device?
`print`	print to PNG files?
`pntscalar`	maximum bubble size for balloon plots; each plot scaled independently based on this maximum size and the values plotted. Often some plots look better with one value and others with a larger or smaller value. Default=2.6
`minnbubble`	minimum number of years below which blank years will be added to bubble plots to avoid cropping
`pwidth`	default width of plots printed to files in units of `punits`. Default=7.
`pheight`	default height width of plots printed to files in units of `punits`. Default=7.
`punits`	units for `pwidth` and `pheight`. Can be "px" (pixels), "in" (inches), "cm" or "mm". Default="in".
`ptsize`	point size for plotted text in plots printed to files (see help("png") in R for details). Default=12.
`res`	resolution of plots printed to files. Default=300
`cex.main`	character expansion parameter for plot titles
`col1`	color for bubbles
`col2`	color for lines with expected values
`col3`	shading color for observations within latency period
`col4`	shading color for observations after latency period
`labels`	vector of labels for plots (titles and axis labels)
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`verbose`	return updates of function progress to the R GUI?

Author(s)

Andre Punt, Ian Taylor

See Also

`SS_plots`, `SS_output`

`SSplotTimeseries` *Plot timeseries data*

Description

Plot timeseries data contained in TIME_SERIES output from Stock Synthesis report file. Some values have optional uncertainty intervals.

Usage

```
SSplotTimeseries(replist, subplot, add = FALSE, areas = "all",
  areacols = "default", areanames = "default", forecastplot = TRUE,
  uncertainty = TRUE, bioscale = "default", minyr = NULL, maxyr = NULL,
  plot = TRUE, print = FALSE, plotdir = "default", verbose = TRUE,
  btarg = "default", minbthresh = "default", xlab = "Year",
  labels = NULL, pwidth = 7, pheight = 7, punits = "in", res = 300,
  ptsize = 12, cex.main = 1)
```

Arguments

`replist`	list created by `SS_output`
`subplot`	number controlling which subplot to create
`add`	add to existing plot? (not yet implemented)
`areas`	optional subset of areas to plot for spatial models
`areacols`	vector of colors by area. Default uses rich.colors by Arni Magnusson
`areanames`	names for areas. Default is to use Area1, Area2,...
`forecastplot`	add points from forecast years
`uncertainty`	add intervals around quantities for which uncertainty is available
`bioscale`	scaling for spawning biomass by default it will be set to 0.5 for single-sex models, and 1.0 for all others
`minyr`	optional input for minimum year to show in plots
`maxyr`	optional input for maximum year to show in plots
`plot`	plot to active plot device?
`print`	print to PNG files?
`plotdir`	directory where PNG or PDF files will be written. by default it will be the directory where the model was run.
`verbose`	report progress to R GUI?
`btarg`	Target depletion to be used in plots showing depletion. May be omitted by setting to 0. "default" chooses value based on modeloutput.
`minbthresh`	Threshold depletion to be used in plots showing depletion. May be omitted by setting to 0. "default" assumes 0.25 unless btarg in model output is 0.25 in which case minbthresh = 0.125 (U.S. west coast flatfish).

`xlab`	x axis label for all plots
`labels`	vector of labels for plots (titles and axis labels)
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`cex.main`	character expansion for plot titles

Author(s)

Ian Taylor, Ian Stewart

See Also

`SS_plots`, `SS_output`

`SSplotYield` *Plot yield and surplus production.*

Description

Plot yield and surplus production from Stock Synthesis output. Surplus production is based on Walters et al. (2008).

Usage

```
SSplotYield(replist, subplots = 1:2, add = FALSE, plot = TRUE,
  print = FALSE, labels = c("Relative depletion", "Equilibrium yield (mt)",
  "Total biomass (mt)", "Surplus production (mt)"), col = "blue", lty = 1,
  lwd = 2, cex.main = 1, pwidth = 7, pheight = 7, punits = "in",
  res = 300, ptsize = 12, plotdir = "default", verbose = TRUE)
```

Arguments

`replist`	list created by `SS_output`
`subplots`	vector controlling which subplots to create
`add`	add to existing plot? (not yet implemented)
`plot`	plot to active plot device?
`print`	print to PNG files?
`labels`	vector of labels for plots (titles and axis labels)
`col`	line color (only applied to equilbrium yield plot at this time)
`lty`	line type (only applied to equilbrium yield plot at this time)

`lwd`	line width (only applied to equilbrium yield plot at this time)
`cex.main`	character expansion for plot titles
`pwidth`	width of plot written to PNG file
`pheight`	height of plot written to PNG file
`punits`	units for PNG file
`res`	resolution for PNG file
`ptsize`	ptsize for PNG file
`plotdir`	directory where PNG files will be written. by default it will be the directory where the model was run.
`verbose`	report progress to R GUI?

Author(s)

Ian Stewart, Ian Taylor

References

Walters, Hilborn, and Christensen, 2008, Surplus production dynamics in declining and recovering fish populations. Can. J. Fish. Aquat. Sci. 65: 2536-2551

See Also

`SS_plots`, `SS_output`

`SSsummarize` *Summarize the output from multiple Stock Synthesis models.*

Description

Summarize various quantities from the model output collected by `SSgetoutput` and return them in a list of tables and vectors.

Usage

```
SSsummarize(biglist, keyvec = NULL, numvec = NULL, sizeselfactor = "Lsel",
  ageselfactor = "Asel", selfleet = NULL, selyr = "startyr",
  selgender = 1, SpawnOutputUnits = NULL, lowerCI = 0.025,
  upperCI = 0.975)
```

Arguments

`biglist`	A list of lists created by `SSgetoutput`.
`keyvec`	Optional list of strings matching names of elements of biglist to subset. Default=NULL.
`numvec`	Optional list of numbers of elements from biglist to subset. Default=NULL.
`sizeselfactor`	A string or vector of strings indicating which elements of the selectivity at length output to summarize. Default=c("Lsel").
`ageselfactor`	A string or vector of strings indicating which elements of the selectivity at age output to summarize. Default=c("Asel").
`selfleet`	Vector of fleets for which selectivity will be summarized. NULL=all fleets. Default=NULL.
`selyr`	String or vector of years for which selectivity will be summarized. NOTE: NOT CURRENTLY WORKING. Options: NULL=all years, "startyr" = first year.
`selgender`	Vector of genders (1 and/or 2) for which selectivity will be summarized. NULL=all genders. Default=NULL.
`SpawnOutputUnits`	Optional single value or vector of "biomass" or "numbers" giving units of spawning for each model.
`lowerCI`	Quantile for lower bound on calculated intervals. Default = 0.025 for 95% intervals.
`upperCI`	Quantile for upper bound on calculated intervals. Default = 0.975 for 95% intervals.

Author(s)

Ian Taylor

See Also

`SSgetoutput`

`SStableComparisons` *make table comparing quantities across models*

Description

Creates a table comparing key quantities from multiple models, which is a reduction of the full information in various parts of the list created using the `SSsummarize` function.

Usage

```
SStableComparisons(summaryoutput, models = "all", likenames = c("TOTAL",
  "Survey", "Length_comp", "Age_comp", "priors", "Size_at_age"),
  names = c("R0", "steep", "NatM", "Q_calc", "L_at_Amax", "VonBert_K",
  "RecrDev_2008", "SPB_Virg", "Bratio_2012", "SPRratio_2011"), digits = NULL,
  modelnames = "default", csv = FALSE, csvdir = "workingdirectory",
  csvfile = "parameter_comparison_table.csv", verbose = TRUE,
  mcmc = FALSE)
```

Arguments

`summaryoutput`	list created by `SSsummarize`
`models`	optional subset of the models described in `summaryoutput`. Either "all" or a vector of numbers indicating columns in summary tables.
`likenames`	Labels for likelihood values to include, should match substring of labels in `summaryoutput$likelihoods`.
`names`	Labels for parameters or derived quantities to include, should match substring of labels in `summaryoutput$pars` or `summaryoutput$quants`.
`digits`	Optional vector of the number of decimal digits to use in reporting each quantity.
`modelnames`	optional vector of labels to use as column names. Default is 'model1','model2',etc.
`csv`	write resulting table to CSV file?
`csvdir`	directory for optional CSV file
`csvfile`	filename for CSV file
`verbose`	report progress to R GUI?
`mcmc`	summarize MCMC output in table?

Author(s)

Ian Taylor

See Also

`SSsummarize`, `SSplotComparisons`, `SS_output`

`SS_changepars` *Change parameters in the control file.*

Description

A function to take advantage of `SS_parlines` that could be used to create a series of control files with different parameter values. This is used by `SS_profile`, but may also be useful for simulation work.

Usage

```
SS_changepars(dir = "C:/myfiles/mymodels/myrun/",
  ctlfile = "control.ss_new", newctlfile = "control_modified.ss",
  linenums = NULL, strings = NULL, newvals = NULL, repeat.vals = FALSE,
  estimate = FALSE, verbose = TRUE)
```

Arguments

`dir`	Directory with control file to change.
`ctlfile`	Control file name. Default="control.ss_new".
`newctlfile`	Name of new control file to be written. Default="control_modified.ss".
`linenums`	Line numbers of control file to be modified. Either this or the Strings input are needed. Default=NULL.
`strings`	Strings (with optional partial matching) indicating which parameters to be modified. This is an alternative to linenums. Strings correspond to the commented parameter names included in control.ss_new, or whatever is written as comment at the end of the 14 number parameter lines. Default=NULL.
`newvals`	Vector of new parameter values. Default=NULL.
`repeat.vals`	If multiple parameter lines match criteria, repeat the `newvals` input for each line
`estimate`	Vector of TRUE/FALSE for which changed parameters are to be estimated. Default=FALSE.
`verbose`	More detailed output to command line. Default=TRUE.

Author(s)

Ian Taylor

See Also

`SS_parlines`, `SS_profile`

Examples

```
## Not run:
SS_changepars(dir='Y:/ss/SSv3.03a/Simple/',ctlfile='Control.SS_New',
              strings=c('SR_steep','SR_sigmaR'),newvals=c(.35,.6))
# [1] wrote new file to Control_Modified.SS
#    oldvals newvals oldphase newphase     comment
# 1 0.609048    0.35        4       -4  # SR_steep
# 2 0.600000    0.60       -4       -4 # SR_sigmaR

## End(Not run)
```

`SS_doRetro` *Run retrospective analyses*

Description

Do retrospective analyses by creating new directories, copying model files, and iteratively changing the starter file to set the number of years of data to exclude.

Usage

```
SS_doRetro(masterdir, oldsubdir, newsubdir = "retrospectives",
  subdirstart = "retro", years = 0:-5, overwrite = TRUE,
  extras = "-nox", intern = FALSE, CallType = "system")
```

Arguments

`masterdir`	Directory where everything takes place.
`oldsubdir`	Subdirectory within `masterdir` with existing model files.
`newsubdir`	Subdirectory within `masterdir` where retrospectives will be run. Default is 'retrospectives'.
`subdirstart`	First part of the pattern of names for the directories in which the models will actually be run.
`years`	Vector of values to iteratively enter into the starter file for retrospective year. Should be zero or negative values.
`overwrite`	Overwrite any input files with matching names in the subdirectories where models will be run.
`extras`	Additional commands to use when running SS. Default = "-nox" will reduce the amound of command-line output.
`intern`	Display runtime information from SS in the R console (vs. saving to a file).
`CallType`	Either "system" or "shell" (choice depends on how you're running R. Default is "system".

Author(s)

Ian Taylor

See Also

`SSgetoutput`

Examples

```
## Not run:
   # note: don't run this in your main directory--make a copy in case something goes wrong
    mydir <- "C:/Simple"

    ## retrospective analyses
    SS_doRetro(masterdir=mydir, oldsubdir="", newsubdir="retrospectives", years=0:-5)

   retroModels <- SSgetoutput(dirvec=file.path(mydir, "retrospectives",paste("retro",0:-5,sep="")))
    retroSummary <- SSsummarize(retroModels)
    endyrvec <- retroSummary$endyrs + 0:-5
   SSplotComparisons(retroSummary, endyrvec=endyrvec, legendlabels=paste("Data",0:-5,"years"))

## End(Not run)
```

`SS_fitbiasramp` *Estimate bias adjustment for recruitment deviates*

Description

Uses standard error of estimated recruitment deviates to estimate the 5 controls for the bias adjustment in Stock Synthesis

Usage

```
SS_fitbiasramp(replist, verbose = FALSE, startvalues = NULL,
  method = "BFGS", twoplots = TRUE, transform = FALSE, plot = TRUE,
  print = FALSE, plotdir = "default", shownew = TRUE, oldctl = NULL,
  newctl = NULL, altmethod = "nlminb", pwidth = 7, pheight = 7,
  punits = "in", ptsize = 12, res = 300, cex.main = 1)
```

Arguments

`replist`	Object created using `SS_output`
`verbose`	Controls the amount of output to the screen. Default=FALSE.
`startvalues`	A vector of 5 values for the starting points in the minimization. Default=NULL.
`method`	A method to apply to the 'optim' function. See ?optim for options. Default="BFGS". By default, optim is not used, and the optimization is based on the input `altmethod`.
`twoplots`	Make a two-panel plot showing devs as well as transformed uncertainty, or just the second panel in the set? Default=TRUE.
`transform`	An experimental option to treat the transform the 5 quantities to improve minimization. Doesn't work well. Default=FALSE.
`plot`	Plot to active plot device?
`print`	Print to PNG files?
`plotdir`	Directory where PNG files will be written. By default it will be the directory where the model was run.

`shownew`	Include new estimated bias adjustment values on top of values used in the model? (TRUE/FALSE)
`oldctl`	Optional name of existing control file to modify. Default=NULL.
`newctl`	Optional name of new control file to create from old file with estimated bias adjustment values. Default=NULL.
`altmethod`	Optimization tool to use in place of optim, either "nlminb" or "psoptim". If not equal to either of these, then optim is used.
`pwidth`	Default width of plots printed to files in units of `punits`. Default=7.
`pheight`	Default height width of plots printed to files in units of `punits`. Default=7.
`punits`	Units for `pwidth` and `pheight`. Can be "px" (pixels), "in" (inches), "cm" or "mm". Default="in".
`ptsize`	Point size for plotted text in plots printed to files (see help("png") in R for details). Default=12.
`res`	Resolution of plots printed to files. Default=300.
`cex.main`	Character expansion for plot titles.

Author(s)

Ian Taylor

References

Methot, R.D. and Taylor, I.G., 2011. Adjusting for bias due to variability of estimated recruitments in fishery assessment models. Can. J. Fish. Aquat. Sci., 68:1744-1760.

See Also

`SS_output`

`SS_html` *Create HTML files to view figures in browser.*

Description

Writes a set of HTML files with tabbed navigation between them. Depends on `SS_plots` with settings in place to write figures to PNG files. Should open main file in default browser automatically.

Usage

```
SS_html(replist = NULL, plotdir = "plots", plotInfoTable = NULL,
  title = "SS Output", width = 500, openfile = TRUE, multimodel = FALSE,
  filenotes = NULL, verbose = TRUE)
```

Arguments

`replist`	Object created by `SS_output`
`plotdir`	Directory where PNG files are located.
`plotInfoTable`	CSV file with info on PNG files. By default, the `plotdir` directory will be searched for files with name beginning 'plotInfoTable*'
`title`	Title for HTML page.
`width`	Width of plots (in pixels).
`openfile`	Automatically open index.html in default browser?
`multimodel`	Override errors associated with plots from multiple model runs. Only do this if you know what you're doing.
`filenotes`	Add additional notes to home page.
`verbose`	Display more info while running this function?

Note

By default, this function will look in the directory where PNG files were created for CSV files with the name 'plotInfoTable...' written by 'SS_plots. HTML files are written to link to these plots and put in the same directory. Please provide feedback on any bugs, annoyances, or suggestions for improvement.

Author(s)

Ian Taylor

See Also

`SS_plots`, `SS_output`

`SS_makedatlist` *make a list for SS data*

Description

create a list similar to those built by `SS_readdat` which can be written to a Stock Synthesis data file using `SS_writedat`. In hindsight, this function doesn't seem very useful and I haven't taken time to describe the arguments below.

Usage

```
SS_makedatlist(styr = 1971, endyr = 2001, nseas = 1,
  months_per_seas = 12, spawn_seas = 1, Nfleet = 1, Nsurveys = 1,
  N_areas = 1, fleetnames = c("fishery1", "survey1"), surveytiming = 0.5,
  areas = 1, units_of_catch = 1, se_log_catch = 0.01, Ngenders = 2,
  Nages = 40, init_equil = 0, catch = NULL, CPUE = NULL,
  N_discard_fleets = 0, discard_data = NULL, meanbodywt = NULL,
  DF_for_meanbodywt = 30, lbin_method = 2, binwidth = 2,
  minimum_size = 2, maximum_size = 90, comp_tail_compression = -1e-04,
  add_to_comp = 1e-04, max_combined_lbin = 0, lbin_vector = seq(22, 90,
  2), lencomp = NULL, agebin_vector = 1:25,
  ageerror = data.frame(rbind(0:40 + 0.5, 0.001, 0:40 + 0.5, seq(0.525, 2.525,
  0.05))), agecomp = NULL, Lbin_method = 3, max_combined_age = 1,
  MeanSize_at_Age_obs = NULL, N_environ_variables = 0, N_environ_obs = 0,
  N_sizefreq_methods = 0, do_tags = 0, morphcomp_data = 0)
```

Arguments

`styr`	start year of the model
`endyr`	end year of the model
`nseas`	number of seasons
`months_per_seas`	months per seasons
`spawn_seas`	spawning season
`Nfleet`	number of fishing fleets
`Nsurveys`	number of surveys
`N_areas`	number of areas
`fleetnames`	names of fleets
`surveytiming`	vector of survey timings
`areas`	area definitions for each fleet or survey
`units_of_catch`	units of catch for each fleet
`se_log_catch`	Uncertainty in catch (standard error in log space).
`Ngenders`	Number of genders.
`Nages`	Number of ages.
`init_equil`	Initial equilibrium catch.
`catch`	Catch data
`CPUE`	Indices of abundance (if present).
`N_discard_fleets`	Number of fleets with discard data.
`discard_data`	Discard data (if exists).
`meanbodywt`	Mean body weight data (if exists)

`DF_for_meanbodywt`	Degrees of freedom for mean body weight t-distribution.
`lbin_method`	Method for entering length bins. (1=use databins; 2=generate from binwidth,min,max below; 3=read vector). Not sure if all options implemented.
`binwidth`	Bin width for length bins.
`minimum_size`	Lower bound of length bins.
`maximum_size`	Upper bound of length bins.
`comp_tail_compression`	Value below which tails of composition data will be compressed (negative to turn off).
`add_to_comp`	Robustifying constant added to multinomial composition likelihoods.
`max_combined_lbin`	Maximum length bin below which length composition data will have genders combined.
`lbin_vector`	Vector of length bins.
`lencomp`	Length composition data (if exists).
`agebin_vector`	Vector of age bins.
`ageerror`	Ageing error matrices.
`agecomp`	Age composition data (if exists).
`Lbin_method`	Method of specifying length bins in conditional age-at-length data.
`max_combined_age`	Maximum age below which age composition data will have genders combined.
`MeanSize_at_Age_obs`	Data on mean size at age (if exists).
`N_environ_variables`	Number of environmental variables.
`N_environ_obs`	Number of environmental observations.
`N_sizefreq_methods`	Number of size frequency methods.
`do_tags`	Include tag data? NOT IMPLEMENTED YET.
`morphcomp_data`	Morph composition data. NOT IMPLEMENTED YET.

Author(s)

Ian Taylor

See Also

`SS_readdat`, `SS_writedat`

`SS_output` *A function to create a list object for the output from Stock Synthesis*

Description

Reads the Report.sso and (optionally) the covar.sso, CompReport.sso and other files files produced by Stock Synthesis and formats the important content of these files into a list in the R workspace. A few statistics unavailable elsewhere are taken from the .par and .cor files. Summary information and statistics can be returned to the R console or just contained within the list produced by this function.

Usage

```
SS_output(dir = "C:/myfiles/mymodels/myrun/", model = "ss3",
  repfile = "Report.sso", compfile = "CompReport.sso",
  covarfile = "covar.sso", forefile = "Forecast-report.sso",
  wtfile = "wtatage.ss_new", ncols = 200, forecast = TRUE, warn = TRUE,
  covar = TRUE, readwt = TRUE, checkcor = TRUE, cormax = 0.95,
  cormin = 0.01, printhighcor = 10, printlowcor = 10, verbose = TRUE,
  printstats = TRUE, hidewarn = FALSE, NoCompOK = FALSE,
  aalmaxbinrange = 4)
```

Arguments

`dir`	Locates the directory of the files to be read in, double backslashes (or forwardslashes) and quotes necessary.
`model`	Name of the executable (leaving off the .exe). Deafult="ss3"
`repfile`	Name of the big report file (could be renamed by user). Default="Report.sso".
`compfile`	Name of the composition report file. Default="CompReport.sso".
`covarfile`	Name of the covariance output file. Default="covar.sso".
`forefile`	Name of the forecast file. Default="Forecast-report.sso".
`wtfile`	Name of the file containing weight at age data. Default="wtatage.ss_new".
`ncols`	The maximum number of columns in files being read in. If this value is too big the function runs more slowly, too small and errors will occur. A warning will be output to the R command line if the value is too small. It should be bigger than the maximum age + 10 and the number of years + 10. Default=200.
`forecast`	Read the forecast-report file? Default=TRUE.
`warn`	Read the Warning.sso file? Default=TRUE.
`covar`	Read covar.sso to get variance information and identify bad correlations? Default=TRUE.
`readwt`	Read the weight-at-age file? Default=TRUE.
`checkcor`	Check for bad correlations? Default=TRUE.

`cormax`	The specified threshold for defining high correlations. A quantity with any correlation above this value is identified. Default=0.95.
`cormin`	The specified threshold for defining low correlations. Only quantities with all correlations below this value are identified (to find variables that appear too independent from the model results). Default=0.01.
`printhighcor`	The maximum number of high correlations to print to the R GUI. Default=10.
`printlowcor`	The maximum number of low correlations to print to the R GUI. Default=10.
`verbose`	Return updates of function progress to the R GUI? Default=TRUE.
`printstats`	Print summary statistics about the output to the R GUI? Default=TRUE.
`hidewarn`	Hides some warnings output from the R GUI. Default=FALSE.
`NoCompOK`	Allow the function to work without a CompReport file. Default=FALSE.
`aalmaxbinrange`	The largest length bin range allowed for composition data to be considered as conditional age-at-length data. Default=4.

Value

Many values are returned. Complete list would be quite long, but should probably be created at some point in the future.

Author(s)

Ian Stewart, Ian Taylor

See Also

`SS_plots`

Examples

```
## Not run:
    myreplist <- SS_output(dir='c:/SS/SSv3.10b/Simple/')

## End(Not run)
```

`SS_parlines` *Get parameter lines from Stock Synthesis control file*

Description

A simple function which takes as input the full path and filename of a control file for input to Stock Synthesis. Ideally, a Control.SS_New file will be used, so that it represents what SS thinks the inputs are, and not what the user thinks the inputs are.

Usage

```
SS_parlines(ctlfile = "C:/myfiles/mymodels/myrun/control.ss_new",
  dir = NULL, verbose = TRUE, active = FALSE, print14 = FALSE)
```

Arguments

`ctlfile`	File name of control file including path.
`dir`	Alternative input of path, where file is assumed to be "control.ss_new". Default=NULL.
`verbose`	TRUE/FALSE switch for amount of detail produced by function. Default=TRUE.
`active`	Should only active parameters (those with positive phase) be output? Default=FALSE.
`print14`	Print 14 columns or just 7 (TRUE/FALSE)

Details

It returns a table which should contain one line for each parameter in the model. Currently, only the first 7 values are returned, because all parameters have those values. In the future, extended parameter lines could be returned.

Parameter lines are identified as those which have 7 or 14 numeric elements followed by a non-numeric element. It's possible that this system could break down under certain circumstances

Author(s)

Ian Taylor

See Also

`SS_changepars`

Examples

```
## Not run:
x <- SS_parlines(ctlfile='Y:/ss/SSv3.03a/Simple/Control.SS_New')
head(x)
#       LO    HI     INIT PRIOR PR_type   SD PHASE              Label Line_num
# 42  0.05  0.15  0.10000  0.10       0  0.8    -3  NatM_p_1_Fem_GP_1       42
# 43  0.05  0.15  0.10000  0.10       0  0.8    -3  NatM_p_2_Fem_GP_1       43
# 44  1.00 45.00 32.28100 36.00       0 10.0     2 L_at_Amin_Fem_GP_1       44
# 45 40.00 90.00 71.34260 70.00       0 10.0     4 L_at_Amax_Fem_GP_1       45
# 46  0.05  0.25  0.15199  0.15       0  0.8     4 VonBert_K_Fem_GP_1       46
# 47  0.05  0.25  0.10000  0.10       0  0.8    -3  CV_young_Fem_GP_1       47

## End(Not run)
```

`SS_plots` *plot many quantities related to output from Stock Synthesis*

Description

Creates a user-chosen set of plots, including biological quantities, time series, and fits to data. Plots are sent to R GUI, single PDF file, or multiple PNG files. This is now just a wrapper which calls on separate functions to make all the plots.

Usage

```
SS_plots(replist = NULL, plot = 1:24, print = NULL, pdf = FALSE,
  png = FALSE, html = png, printfolder = "plots", dir = "default",
  fleets = "all", areas = "all", fleetnames = "default",
  fleetcols = "default", fleetlty = 1, fleetpch = 1, lwd = 1,
  areacols = "default", areanames = "default", verbose = TRUE,
  uncertainty = TRUE, forecastplot = FALSE, datplot = FALSE,
  Natageplot = TRUE, samplesizeplots = TRUE, compresidplots = TRUE,
  sprtarg = "default", btarg = "default", minbthresh = "default",
  pntscalar = NULL, bub.scale.pearson = 1.5, bub.scale.dat = 3,
  pntscalar.nums = 2.6, minnbubble = 8, aalyear = -1, aalbin = -1,
  aalresids = FALSE, maxneff = 5000, cohortlines = c(), smooth = TRUE,
  showsampsize = TRUE, showeffN = TRUE, sampsizeline = FALSE,
  effNline = FALSE, showlegend = TRUE, pwidth = 7, pheight = 7,
  punits = "in", ptsize = 12, res = 300, cex.main = 1,
  selexlines = 1:6, rows = 1, cols = 1, maxrows = 6, maxcols = 6,
  maxrows2 = 2, maxcols2 = 4, andrerows = 3, tagrows = 3, tagcols = 3,
  fixdims = TRUE, new = TRUE, SSplotDatMargin = 8, filenotes = NULL,
  catchasnumbers = NULL, catchbars = TRUE, legendloc = "topleft",
  minyr = NULL, maxyr = NULL, scalebins = FALSE, scalebubbles = FALSE,
  tslabels = NULL, catlabels = NULL, ...)
```

Arguments

`replist`	List created by `SS_output`
`plot`	Plot sets to be created, see list of plots below. Use to specify only those plot sets of interest, e.g., c(1,2,5,10). Plots for data not available in the model run will automatically be skipped, whether called or not.
`print`	Deprecated input for backward compatability, now replaced by `png = TRUE/FALSE`.
`pdf`	Send plots to PDF file instead of R GUI?
`png`	Send plots to PNG files instead of R GUI?
`html`	Run `SS_html` on completion? By default has same value as png.
`printfolder`	Name of subfolder to create within the working directory into which any PNG files specified by `print` will be saved. By default the working directory is used with no subfolder. Default="".
`dir`	The directory in which any PNG files requested by `print` are created. By default it will be the same directory that the report file was read from by the `SS_output` function. Default="default".
`fleets`	Either the string "all", or a vector of numerical values, like c(1,3), listing fleets or surveys for which plots should be made. By default, plots will be made for all fleets and surveys. Default="all".
`areas`	Either the string "all", or a vector of numerical values, like c(1,3), listing areas for which plots should be made in a multi-area model. By default, plots will be made for all areas (excepting cases where the function has not yet been updated for multi-area models). Default="all".

`fleetnames`	Either the string "default", or a vector of characters strings to use for each fleet name. Default="default".
`fleetcols`	Either the string "default", or a vector of colors to use for each fleet. Default="default".
`fleetlty`	Vector of line types used for each fleet in some plots. Default=1.
`fleetpch`	Vector of point types used for each fleet in some plots. Default=1.
`lwd`	Line width for some plots. Default=1.
`areacols`	Either the string "default", or a vector of colors to use for each area. Default="default".
`areanames`	Optional vector of names for each area used in titles. Default="default".
`verbose`	Return updates of function progress to the R GUI? Default=T.
`uncertainty`	Include values in plots showing estimates of uncertainty (requires positive definite hessian in model and `covar`=T in `SS_output`)? Default=T.
`forecastplot`	Include forecast years in the plots? Obviously requires forecast options to have been used in the model. Default=T.
`datplot`	Plot the data by itself? This is useful in document preparation. Setting datplot=F is equivalent to leaving off plots 15 and 16. Default=F.
`Natageplot`	Plot the expected numbers at age bubble plots and mean-age time series? Default=T.
`samplesizeplots`	Show sample size plots? Default=T.
`compresidplots`	Show residuals for composition plots?
`sprtarg`	Specify the F/SPR proxy target. Default=0.4.
`btarg`	Target depletion to be used in plots showing depletion. May be omitted by setting to NA. Default=0.4.
`minbthresh`	Threshold depletion to be used in plots showing depletion. May be omitted by setting to NA. Default=0.25.
`pntscalar`	This scalar defines the maximum bubble size for bubble plots. This option is still available but a better choice is to use bub.scale.pearson and bub.scale.dat, which are allow the same scaling throughout all plots.
`pntscalar.nums`	This scalar defines the maximum bubble size for numbers-at-age and numbers-at-length plots.
`bub.scale.pearson`	Character expansion (cex) value for a proportion of 1.0 in bubble plot of Pearson residuals. Default=1.5.
`bub.scale.dat`	Character expansion (cex) value for a proportion of 1.0 in bubble plot of composition data. Default=3.
`minnbubble`	This defines the minimum number of years below which blank years will be added to bubble plots to avoid cropping. Default=8.
`aalyear`	Years to plot multi-panel conditional age-at-length fits for all length bins; must be in a "c(YYYY,YYYY)" format. Useful for checking the fit of a dominant year class, critical time period, etc. Default=-1.

`aalbin`	The length bin for which multi-panel plots of the fit to conditional age-at-length data will be produced for all years. Useful to see if growth curves are ok, or to see the information on year classes move through the conditional data. Default=-1.
`aalresids`	Plot the full set of conditional age-at-length Pearson residuals? Default=F.
`maxneff`	The maximum value to include on plots of input and effective sample size. Occasionally a calculation of effective N blows up to very large numbers, rendering it impossible to observe the relationship for other data. Default=5000.
`cohortlines`	Optional vector of birth years for cohorts for which to add growth curves to numbers at length bubble plots. Default=c().
`smooth`	Add loess smoother to observed vs. expected index plots and input vs. effective sample size? Default=T.
`showsampsize`	Display sample sizes on composition plots? Default=T.
`showeffN`	Display effective sample sizes on composition plots? Default=T.
`sampsizeline`	show line for input sample sizes on top of conditional age-at-length plots (TRUE/FALSE, still in development)
`effNline`	show line for effective sample sizes on top of conditional age-at-length plots (TRUE/FALSE, still in development)
`showlegend`	Display legends in various plots? Default=T.
`pwidth`	Default width of plots printed to files in units of `punits`. Default=7.
`pheight`	Default height width of plots printed to files in units of `punits`. Default=7.
`punits`	Units for `pwidth` and `pheight`. Can be "px" (pixels), "in" (inches), "cm" or "mm". Default="in".
`ptsize`	Point size for plotted text in plots printed to files (see help("png") in R for details). Default=12.
`res`	Resolution of plots printed to files. Default=300.
`cex.main`	Character expansion parameter for plot titles (not yet implemented for all plots). Default=1.
`selexlines`	Vector controling which lines should be shown on selectivity plots if the model includes retention. Default=1:5.
`rows`	Number of rows to use for single panel plots. Default=1.
`cols`	Number of columns to use for single panel plots. Default=1.
`maxrows`	Maximum number of rows to for multi-panel plots. Default=6.
`maxcols`	Maximum number of columns for multi-panel plots. Default=6.
`maxrows2`	Maximum number of rows for conditional age-at-length multi-panel plots. Default=2.
`maxcols2`	Maximum number of rows for conditional age-at-length multi-panel plots. Default=4.
`andrerows`	Number of rows of Andre's conditional age-at-length plots within each page. Default=3.
`tagrows`	Number of rows for tagging-related plots. Default=3.

`tagcols`	Number of columns for tagging-related plots. Default=3.
`fixdims`	Control whether multi-panel plots all have dimensions equal to maxrows by maxcols, or resized within those limits to fit number of plots. Default=T.
`new`	Open a new window or add to existing plot windows. Default=T.
`SSplotDatMargin`	Size of right-hand margin in data plot (may be too small if fleet names are long)
`filenotes`	Optional vector of character strings to be added to intro HTML page (if created) with notes about the model.
`catchasnumbers`	Is catch input in numbers instead of biomass? Default=F.
`catchbars`	show catch by fleet as barplot instead of stacked polygons (default=TRUE)
`legendloc`	Location for all legends. Default="topleft".
`minyr`	First year to show in time-series plots (changes xlim parameters).
`maxyr`	Last year to show in time-series plots (changes xlim parameters).
`scalebins`	Rescale expected and observed proportions in composition plots by dividing by bin width for models where bins have different widths? Caution!: May not work correctly in all cases.
`scalebubbles`	scale data-only bubbles by sample size, not just proportion within sample? Default=FALSE.
`tslabels`	Either NULL to have default labels for timeseries plots or a vector of appropriate length (currently 11) with labels for each figure
`catlabels`	Either NULL to have default labels for catch plots or a vector of appropriate length (currently 10) with labels for each figure
`...`	Additional arguments that will be passed to some subfunctions.

Author(s)

Ian Stewart, Ian Taylor

References

Walters, Hilborn, and Christensen, 2008, Surplus production dynamics in declining and recovering fish populations. Can. J. Fish. Aquat. Sci. 65: 2536-2551.

See Also

`SS_output`, `SSplotBiology`, `SSplotCatch`, `SSplotComps`, `SSplotDiscard`, `SSplotIndices`, `SSplotMnwt`, `SSplotNumbers`, `SSplotRecdevs`, `SSplotSelex`, `SSplotSpawnrecruit`, `SSplotSPR`, `SSplotTags`, `SSplotTimeseries`, `SSplotYield`

`SS_profile` *Run a likelihood profile in Stock Synthesis.*

Description

Iteratively changes the control file using SS_changepars.

Usage

```
SS_profile(dir = "C:/myfiles/mymodels/myrun/",
  masterctlfile = "control.ss_new", newctlfile = "control_modified.ss",
  linenum = NULL, string = NULL, profilevec = NULL, usepar = FALSE,
  globalpar = FALSE, parfile = NULL, parlinenum = NULL,
  parstring = NULL, dircopy = TRUE, exe.delete = FALSE, model = "ss3",
  extras = "-nox", systemcmd = FALSE, saveoutput = TRUE,
  overwrite = TRUE, whichruns = NULL, verbose = TRUE)
```

Arguments

`dir`	Directory where input files and executable are located.
`masterctlfile`	Source control file. Default = "control.ss_new"
`newctlfile`	Destination for new control files (must match entry in starter file). Default = "control_modified.ss".
`linenum`	Line number of parameter to be changed. Can be used instead of `string` or left as NULL.
`string`	String partially matching name of parameter to be changed. Can be used instead of `linenum` or left as NULL.
`usepar`	Use PAR file from previous profile step for starting values?
`globalpar`	Use global par file for all runs instead of the par file from each successive run
`parfile`	Name of par file to use (Ian says "I don't remember how this interacts with the globalpar input")
`parlinenum`	Line number in par file to change.
`parstring`	String in par file preceding line number to change.
`dircopy`	Copy directories for each run? NOT IMPLEMENTED YET.
`exe.delete`	Delete exe files in each directory? NOT IMPLEMENTED YET.
`profilevec`	Vector of values to profile over. Default = NULL.
`model`	Name of executable. Default = "ss3".
`extras`	Additional commands to use when running SS. Default = "-nox" will reduce the amound of command-line output.
`systemcmd`	Should R call SS using "system" function intead of "shell". This may be required when running R in Emacs. Default = FALSE.
`saveoutput`	Copy output .SSO files to unique names. Default = TRUE.

`overwrite`	Overwrite any existing .SSO files. Default = TRUE. If FALSE, then some runs may be skipped.
`whichruns`	Optional vector of run indices to do. This can be used to re-run a subset of the cases in situations where the function was interupted or some runs fail to converge. Must be a subset of 1:n, where n is the length of profilevec.
`verbose`	Controls amount of info output to command line. Default = TRUE.

Note

The starting values used in this profile are not ideal and some models may not converge. Care should be taken in using an automated tool like this, and some models are likely to require rerunning with alternate starting values.

Also, someday this function will be improved to work directly with the plotting function `SSplotProfile`, but they don't yet work well together. Thus, even if `SS_profile` is used, the output should be read using `SSgetoutput` or by multiple calls to `SS_output` before sending to `SSplotProfile`.

Author(s)

Ian Taylor

See Also

`SSplotProfile`, `SSgetoutput`, `SS_changepars`, `SS_parlines`

Examples

```
## Not run:
# note: don't run this in your main directory
# make a copy in case something goes wrong
mydir <- "C:/ss/Simple - Copy"

# the following commands related to starter.ss could be done by hand
# read starter file
starter <- SS_readstarter(file.path(mydir, 'starter.ss'))
# change control file name in the starter file
starter$ctlfile <- "control_modified.ss"
# make sure the prior likelihood is calculated
# for non-estimated quantities
starter$prior_like <- 1
# write modified starter file
SS_writestarter(starter, dir=mydir, overwrite=TRUE)

# vector of values to profile over
h.vec <- seq(0.3,0.9,.1)
Nprofile <- length(h.vec)

# run SS_profile command
profile <- SS_profile(dir=mydir, # directory
                      # "NatM" is a subset of one of the
                      # parameter labels in control.ss_new
```

```
                          model="ss3_safe",
                          masterctlfile="control.ss_new",
                          newctlfile="control_modified.ss",
                          string="steep",
                          profilevec=h.vec)


# read the output files (with names like Report1.sso, Report2.sso, etc.)
profilemodels <- SSgetoutput(dirvec=mydir, keyvec=1:Nprofile)
# summarize output
profilesummary <- SSsummarize(profilemodels)

# OPTIONAL COMMANDS TO ADD MODEL WITH PROFILE PARAMETER ESTIMATED
MLEmodel <- SS_output("C:/ss/SSv3.24l_Dec5/Simple")
profilemodels$MLE <- MLEmodel
profilesummary <- SSsummarize(profilemodels)
# END OPTIONAL COMMANDS

# plot profile using summary created above
SSplotProfile(profilesummary,           # summary object
              profile.string = "steep", # substring of profile parameter
              profile.label="Stock-recruit steepness (h)") # axis label

# make timeseries plots comparing models in profile
SSplotComparisons(profilesummary,legendlabels=paste("h =",h.vec))


## End(Not run)
```

`SS_readctl` *read control file*

Description

read Stock Synthesis control file into list object in R

Usage

```
SS_readctl(file)
```

Arguments

`file`	Filename either with full path or relative to working directory.

Details

This function is not fully implemented. The logic to figure out all the details of a Stock Synthesis control file is very complex, so this function may be completed in a way that is not totally consistent with the other similar files. Or it may never be completed at all. The functions `SS_changepars` and `SS_parlines` offer alternatives for working with SS control files.

Author(s)

Ian Taylor

See Also

`SS_changepars`, `SS_parlines`, `SS_readstarter`, `SS_readforecast`, `SS_readdat`, `SS_writestarter`, `SS_writeforecast`, `SS_writedat`, `SS_writectl`

`SS_readdat` *read data file*

Description

read Stock Synthesis data file into list object in R

Usage

```
SS_readdat(file, verbose = TRUE, echoall = FALSE, section = NULL)
```

Arguments

`file`	Filename either with full path or relative to working directory.
`verbose`	Should there be verbose output while running the file? Default=TRUE.
`echoall`	Debugging tool (not fully implemented) of echoing blocks of data as it is being read.
`section`	Which data set to read. Only applies for a data.ss_new file created by Stock Synthesis. Allows the choice of either expected values (section=2) or bootstrap data (section=3+). Leaving default of section=NULL will read input data, (equivalent to section=1).

Author(s)

Ian Taylor

See Also

`SS_readstarter`, `SS_readforecast`, `SS_readctl`, `SS_writestarter`, `SS_writeforecast`, `SS_writedat`, `SS_writectl`

`SS_readforecast` *read forecast file*

Description

read Stock Synthesis forecast file into list object in R

Usage

```
SS_readforecast(file = "forecast.ss", Nfleets, Nareas, verbose = TRUE)
```

Arguments

`file`	Filename either with full path or relative to working directory.
`Nfleets`	Number of fleets.
`Nareas`	Number of areas.
`verbose`	Should there be verbose output while running the file?

Author(s)

Ian Taylor

See Also

`SS_readstarter`, `SS_readdat`, `SS_readctl`, `SS_writestarter`, `SS_writeforecast`, `SS_writedat`, `SS_writectl`

`SS_readstarter` *read starter file*

Description

read Stock Synthesis starter file into list object in R

Usage

```
SS_readstarter(file = "starter.ss", verbose = TRUE)
```

Arguments

`file`	Filename either with full path or relative to working directory.
`verbose`	Should there be verbose output while running the file?

Author(s)

Ian Taylor

See Also

`SS_readforecast`, `SS_readdat`, `SS_readctl`, `SS_writestarter`, `SS_writeforecast`, `SS_writedat`, `SS_writectl`

`SS_recdevs` *Insert a vector of recruitment deviations into the control file.*

Description

A function to insert a vector of recruitment deviations into the control file for simulation studies. This can also be achieved by using the .par file, but Ian Taylor prefers this approach for no good reason.

Usage

```
SS_recdevs(fyr, lyr, ctl = NULL, recdevs = NULL, rescale = TRUE,
  scaleyrs = NULL, dir = "working_directory", ctlfile = "control.ss_new",
  newctlfile = "control_modified.ss", verbose = TRUE, writectl = TRUE,
  returnctl = FALSE, newmaxbias = NULL)
```

Arguments

`fyr`	First year of the recdev vector.
`lyr`	Last year of the recdev vector.
`ctl`	Either NULL to read anew or an already read control file. Default=NULL.
`recdevs`	Either NULL to generate anew or an already generated vector of recdevs. Default=NULL.
`rescale`	Should the recdevs be rescaled to have mean = 0 and std. deviation = sigmaR? Default=TRUE.
`scaleyrs`	Vector of years over which rescaling (if chosen) should occur.
`dir`	Directory where files are located. Default is to use the working directory in use by R. Default="working_directory".
`ctlfile`	Name of control file to modify. Default="control.ss_new".
`newctlfile`	Name of new file to output modified control file. Default="control_modified.ss".
`verbose`	Verbose output to R command line? Default=TRUE.
`writectl`	Write new file? Default=TRUE.
`returnctl`	Return contents ctl file as an object in the R workspace. Default=FALSE.
`newmaxbias`	Replace the maximum bias adjustment fraction with any non-NULL value. Default=NULL.

Author(s)

Ian Taylor

SS_RunJitter *Iteratively apply the jitter option in SS*

Description

Iteratively runs SS model with different jittered starting parameter values (jitter value must be manually set in starter.ss). Output files are renamed in the format Report1.sso, Report2.sso, etc.

Usage

```
SS_RunJitter(mydir, model = "ss3",
  extras = "-nohess -cbs 500000000 -gbs 500000000", Njitter, Intern = TRUE,
  systemcmd = FALSE, printlikes = TRUE)
```

Arguments

`mydir`	Directory where model files are located
`model`	Executable name
`extras`	Additional command line arguments passed to executable
`Njitter`	Number of jitters
`Intern`	Show command line info in R console or keep hidden (Internal=TRUE)
`systemcmd`	Option to switch between 'shell' and 'system'
`printlikes`	Print likelihood values to console

Author(s)

Jim Thorson

SS_splitdat *Split apart bootstrap data to make input file.*

Description

A function to split apart bootstrap data files created in data.ss_new. To get bootstraps, the input "N bootstrap file to produce" in starter.ss needs to be 3 or greater.

Usage

```
SS_splitdat(inpath = "working_directory", outpath = "working_directory",
  inname = "data.ss_new", outpattern = "BootData", number = FALSE,
  verbose = TRUE, fillblank = TRUE, MLE = TRUE, inputs = FALSE,
  notes = "")
```

Arguments

`inpath`	Directory containing the input file. By default the working directory given by getwd() is used. Default="working_directory".
`outpath`	Directory into which the output file will be written. Default="working_directory".
`inname`	File name of input data file to be split. Default="Data.SS_New".
`outpattern`	File name of output data file. Default="BootData".
`number`	Append bootstrap number to the file name chosen in `outpattern`? Default=F.
`verbose`	Provide richer command line info of function progress? Default=TRUE.
`fillblank`	Replace blank lines with "#". Helps with running on linux. Default=TRUE.
`MLE`	Grab the maximum likelihood values from the second block in Data.SS_New (instead of bootstrap values or copies of inputs)? Default=TRUE.
`inputs`	Grab the copy of the input values values from the first block in Data.SS_New (instead of MLE or bootstrap values)? Default=F.
`notes`	Notes to the top of the new file (comment indicator "#C" will be added). Default="".

Author(s)

Ian Taylor

`SS_writectl` *write control file*

Description

Write Stock Synthesis control file. Like `SS_readctl`, this function is not fully developed.

Usage

```
SS_writectl(ctllist, outfile, overwrite = F, verbose = T)
```

Arguments

`ctllist`	List object created by `SS_readctl`.
`outfile`	Filename for where to write new control file.
`overwrite`	Should existing files be overwritten? Default=F.
`verbose`	Should there be verbose output while running the file? Default=T.

Author(s)

Ian Taylor

See Also

`SS_readstarter`, `SS_readforecast`, `SS_readdat`, `SS_readctl`, `SS_writestarter`, `SS_writeforecast`, `SS_writedat`, `SS_writectl`

`SS_writedat` *write data file*

Description

write Stock Synthesis data file from list object in R which was probably created using `SS_readdat`

Usage

```
SS_writedat(datlist, outfile, overwrite = FALSE, verbose = TRUE)
```

Arguments

`datlist`	List object created by `SS_readdat`.
`outfile`	Filename for where to write new data file.
`overwrite`	Should existing files be overwritten? Default=FALSE.
`verbose`	Should there be verbose output while running the file?

Author(s)

Ian Taylor

See Also

`SS_makedatlist`, `SS_readstarter`, `SS_readforecast`, `SS_readctl`, `SS_writestarter`, `SS_writeforecast`, `SS_writedat`, `SS_writectl`

`SS_writeforecast` *write forecast file*

Description

write Stock Synthesis forecast file from list object in R which was probably created using `SS_readforecast`

Usage

```
SS_writeforecast(mylist, dir = NULL, file = "forecast.ss",
  overwrite = FALSE, verbose = TRUE)
```

Arguments

`mylist`	List object created by `SS_readforecast`.
`dir`	Directory for new forecast file. Default=NULL (working directory).
`file`	Filename for new forecast file. Default="forecast.ss".
`overwrite`	Should existing files be overwritten? Default=FALSE.
`verbose`	Should there be verbose output while running the file? Default=TRUE.

Author(s)

Ian Taylor

See Also

`SS_readstarter`, `SS_readforecast`, `SS_readdat`, `SS_readctl`, `SS_writestarter`, `SS_writedat`, `SS_writectl`

`SS_writestarter` *write starter file*

Description

write Stock Synthesis starter file from list object in R which was probably created using `SS_readstarter`

Usage

```
SS_writestarter(mylist, dir = NULL, file = "starter.ss",
  overwrite = FALSE, verbose = TRUE, warn = TRUE)
```

Arguments

`mylist`	List object created by `SS_readstarter`.
`dir`	Directory for new starter file. Default=NULL (working directory).
`file`	Filename for new starter file. Default="starter.ss".
`overwrite`	Should existing files be overwritten? Default=FALSE.
`verbose`	Should there be verbose output while running the file? Default=TRUE.
`warn`	Print warning if overwriting file?

Author(s)

Ian Taylor

See Also

`SS_readstarter`, `SS_readforecast`, `SS_readctl`, `SS_writestarter`, `SS_writeforecast`, `SS_writedat`, `SS_writectl`

`SS_write_length.fit` *Write length.fit file to be used by the MFCL length-comp viewer.*

Description

Writes files in the format used by the MFCL length-composition viewer. Inspired by Simon Hoyle's demonstration. Still needs work.

Usage

```
SS_write_length.fit(replist = NULL, outfile = "length.fit",
  compfile = "CompReport.sso", dir = "default", overwrite = FALSE,
  verbose = TRUE)
```

Arguments

`replist`	List created by `SS_output`
`outfile`	Name of file to create.
`compfile`	SS output file with composition data info.
`dir`	Directory where stuff happens. Defaults to directory where model was run.
`overwrite`	Overwrite existing file?
`verbose`	More verbose info on progress of the function?

Author(s)

Ian Taylor

References

`http://www.multifan-cl.org/`, `http://www.spc.int/OceanFish/en/ofpsection/sam/research/272-mfcl-viewer`

`stackpoly` *function "stackpoly" by Jim Lemon from "plotrix" package*

Description

Plot one or more columns of numeric values as the top edges of polygons instead of lines.

Usage

```
stackpoly(x, y, main = "", xlab = "", ylab = "", xat = NA,
  xaxlab = NA, xlim = NA, ylim = NA, lty = 1, border = NA, col = NA,
  axis4 = F, ...)
```

Arguments

`x`	A numeric data frame or matrix with the 'x' values. If 'y' is NULL, these will become the 'y' values and the 'x' positions will be the integers from 1 to dim(x)[1].
`y`	The 'y' values.
`main`	The title for the plot.
`xlab`	x axis labels for the plot.
`ylab`	y axis labels for the plot.
`xat`	Where to put the optional xaxlabs.
`xaxlab`	Optional labels for the x positions.
`xlim`	Optional x limits.
`ylim`	Optional y limits.
`lty`	Line type for the polygon borders.
`border`	Color for the polygon borders.
`col`	Color to fill the polygons. If NULL, 'rainbow' will be called to generate the colors. If NA, the polygons will not be filled.
`axis4`	option to add an axis on the right hand side
`...`	Additional arguments passed to 'plot'.

Author(s)

Jim Lemon

References

http://cran.r-project.org/web/packages/plotrix/index.html

`TSCplot` *Create a plot for the TSC report*

Description

Creates a plot of catch and spawning biomass from the output of `SS_output` for the NOAA TSC report.

Usage

```
TSCplot(SSout, yrs = "default", ylimBar = "default", ylimDepl = c(0,
  1.025), colBar = "yellow", cexBarLabels = 1.1, cex.axis = 1.1,
  space = 0, pchDepl = 19, colDepl = "red", lwdDepl = 3,
  shiftDepl = 0.25, pchSpace = 5, ht = 4, wd = 7, labelLines = 2.8,
  makePDF = NULL, makePNG = NULL, MCMC = F)
```

Arguments

`SSout`	The output from `SS_output`
`yrs`	The vector of years to plot
`ylimBar`	y-axis limits for catch barplot
`ylimDepl`	y-axis limits for depletion line
`colBar`	colors of the bars
`cexBarLabels`	character expansion for the labels underneath the bars (years)
`cex.axis`	character expansion for the axis labels
`space`	space between bars (see space argument of `barplot`)
`pchDepl`	character type for points on the depletion line
`colDepl`	color of the points on the depletion line
`lwdDepl`	width of the depletion line
`shiftDepl`	shift from beginning of the year for the points on the depletion line. Helps to guide the eye for exactly which year it corresponds to.
`pchSpace`	number of years between points on the depletion line. Higher numbers help tidy up the plot when plotting many years.
`ht`	Height of the plot in inches
`wd`	Width of the plot in inches
`labelLines`	line argument for `mtext` to move the axis labels
`makePDF`	filename for a pdf file. If NULL it does not make a pdf. Can specify a pdf filename or a png filename. Not both at the same time.
`makePNG`	filename for a png image. If NULL it does not make a png. Can specify a pdf filename or a png filename. Not both at the same time.
`MCMC`	If TRUE, will use mcmc results. It needs a list element called 'mcmc' on SSout.

Details

It creates a plot on the current graphics device, in a pdf file, or as a png image of the figure used in the TSC report produced by the NWFSC. It expects the SS results read in by `SS_output`. If MCMC results are to be plotted, a 'mcmc' list element should be added using the `SSgetMCMC` function. See the examples below.

Value

Returns a data frame with the years, spawning biomass, depletion, and total dead catch.

Author(s)

Allan Hicks

See Also

`SS_output` `SSgetMCMC`

Examples

```
## Not run:
  ######################################
  #DO NOT RUN
    library(r4ss)
    update_r4ss_files()

    # ** CHANGE TO THE BASE DIRECTORY
    directory <- "C:\NOAA2011\Dover\Models\base_20110701"

    base <- SS_output(dir=directory,covar=F,verbose=F)

    #show the plot in R
    TSCplot(base)
    TSCplot(base,yrs=2000:2011,pchSpace = 1)
    #Create the plot as a PNG file
    TSCplot(base,makePNG="C:\NOAA2012\Assessments\TSCdover.png")
    #Create the plot as a PDF file
    TSCplot(base,makePDF="C:\NOAA2012\Assessment\TSCdover.pdf")

    # ** Hake model with MCMC results
    SSdir <- "C:/NOAA2012/Hake/Models"
    base <- SS_output(dir=paste(SSdir,"81_base_MCMC",sep="/"),covar=F)
    tmp <- SSgetMCMC(dir=paste(SSdir,"81_base_MCMC",sep="/"),writecsv=F)
    base$mcmc <- data.frame(tmp$model1)
    TSCplot(base,ylimDepl = c(0,1.25),pchSpace=1,MCMC=T)


  ###############################################

## End(Not run)
```

`update_r4ss_files` *Updates r4ss files to newest versions on web.*

Description

Sources files containing R functions r4ss package from the online code repository. These may often be newer than those available form CRAN mirrors. With a switch to GitHub from Google Code, it might be better to use this option from the devtools package: devtools::install_github("r4ss/r4ss").

Usage

```
update_r4ss_files(local = NULL, save = FALSE, revision = "newest",
  GitHub = TRUE, override = FALSE)
```

Arguments

`local`	A local directory from which to source the files instead of getting them from the web.
`save`	If TRUE, then copy files from web to local directory, then source from this same local directory
`revision`	Either "newest" (the default), or an optional revision number of the files to source. NOTE: revision number option doesn't yet work from GitHub.
`GitHub`	Get files from GitHub (as opposed to Google Code)?
`override`	Override the message about how you should get code from GitHub using function in devtools package?

Author(s)

Ian Taylor

Examples

```
## Not run:
# run with no arguments
update_r4ss_files()
r4ss is moving to GitHub from Google Code. You should no longer run
 update_r4ss_files and instead install the "devtools" package to get
 updated code by running the following command:
   devtools::install_github("r4ss/r4ss")
 to override this message, use the argument "override=TRUE"

# update anyway
update_r4ss_files(override=TRUE)
most recent change: July 03, 2014
77 files found
  sourcing...
   IOTCmove.R, NegLogInt_Fn.R, PinerPlot.R, RebuildPlot.R,
   SSFishGraph.R, SS_RunJitter.R, SS_changepars.R, SS_doRetro.R,
   ...
   sel.line.R, selfit.R, selfit_spline.R, stackpoly.R,
   update_r4ss_files.R,

  r4ss update complete.


# copy files from web to local directory and then source them
update_r4ss_files(local='c:/SS/R/r4ss_files/',save=TRUE, override=TRUE)

# source files from a local directory (i.e. if no network available)
update_r4ss_files(local='c:/SS/R/r4ss_files/',save=FALSE)

## End(Not run)
```

Index