Package 'DLMtool'

November 26, 2017

Type Package

Title Data-Limited Methods Toolkit

Version 5.0.2

Date 2017-11-25

Maintainer Tom Carruthers <t.carruthers@fisheries.ubc.ca>

Description Development, simulation testing, and implementation of management procedures for data-limited fisheries
(see Carruthers et al (2014) <doi:10.1016/j.fishres.2013.12.014>).

License GPL-2

Depends R (>= 2.10.0), snowfall

Imports abind, coda, devtools, methods, LBSPR, grDevices, ggplot2, gridExtra, ggrepel, knitr, MASS, mvtnorm, openxlsx, parallel, r4ss, readxl, rmarkdown, stats, utils

LazyData yes

LazyLoad yes

Suggests testthat

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 6.0.1

BugReports https://github.com/DLMtool/DLMtool/issues

NeedsCompilation yes

Author Tom Carruthers [aut, cre],
Adrian Hordyk [aut]

Repository CRAN

Date/Publication 2017-11-25 23:38:30 UTC

R topics documented:

DLMtool-package *Data-Limited Methods Toolkit*

Description

Simulation testing and implementation of data-limited fishery stock assessment methods

Additional Information

See the DLMtool User Guide for a detailed description of how to use the DLMtool package.

The help documentation for the DLMtool package can also be accessed here.

See the Data-Limited Toolkit Website for more information on the DLMtool, including an interactive demo of the main features of the toolkit, information on case studies where the toolkit has been applied, and more about the history and development of the DLMtool.

Author(s)

Tom Carruthers <t.carruthers@fisheries.ubc.ca>

Adrian Hordyk <a.hordyk@oceans.ubc.ca>

References

Carruthers, T.R., Punt, A.E., Walters, C.J., MacCall, A., McAllister, M.K., Dick, E.J., Cope, J. 2014. Evaluating methods for setting catch limits in data-limited fisheries. Fisheries Research. 153: 48-68.

Carruthers, T.R., Kell, L.T., Butterworth, D.S., Maunder, M.N., Geromont, H.F., Walters, C., McAllister, M.K., Hillary, R., Levontin, P., Kitakado, T., Davies, C.R. Performance review of simple management procedures. ICES Journal of Marine Science.

AAVY *Performance Metric: Probability AAVY < 0.2*

Description

Performance Metric: Probability AAVY < 0.2

Usage

```
AAVY(MSEobj = NULL)
```

Arguments

MSEobj	An object of class MSE

Value

An object of class PMobj

Examples

```
## Not run:
AAVY(myMSE)

## End(Not run)
```

`Albacore` *Albacore Stock*

Description

An object of class Stock

Usage

```
Albacore
```

Format

An object of class `Stock` of length 1.

`alphaconv` *Calculate alpha parameter for beta distribution from mean and standard deviation*

Description

Calculate alpha parameter for beta distribution from mean and standard deviation

Usage

```
alphaconv(m, sd)
```

Arguments

`m`	mean
`sd`	standard deviation

Value

numeric

Author(s)

T. Carruthers

Atlantic_mackerel *Atlantic_mackerel Data*

Description

An object of class Data

Usage

```
Atlantic_mackerel
```

Format

An object of class `Data` of length 1.

avail *What objects of this class are available*

Description

Generic class finder

Usage

```
avail(classy)
```

Arguments

`classy`	A class of object (character string, e.g. 'Fleet')

Details

Finds objects of the specified class in the global environment or the package:DLMtool

Author(s)

T. Carruthers

AvC *Average Catch*

Description

A simple average catch MP that is included to demonstrate a 'status quo' management option

Usage

```
AvC(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Author(s)

T. Carruthers

barplot.MSE *Plot a barplot of MSE results*

Description

Plot a barplot of MSE results

Usage

```
## S3 method for class 'MSE'
barplot(height, MSEobj = NULL, PMs = list(B_BMSY = 0.5,
  SSB_SSB0 = 0.2), PLim = 0.8, lastYrs = 10, maxMP = 14, MPs = NA,
  Title = NULL, sims = NULL, msg = TRUE, cex.names = 1.3,
  incRef = FALSE, ...)
```

Arguments

height	An object of class MSE. Generic function must have argument height. But note that this must be an MSE object.
MSEobj	Optional. An object of class MSE. Overides height
PMs	List of performance metrics. Options are c('SSB_SSB0','B_BMSY', 'F_FMSY', 'AAVE', 'AAVY')
PLim	Probability threshold
lastYrs	Last number of years in projection to calculate statistics

maxMP	Maximum number of MPs to include in each plot
MPs	Optional subset MSE object by MP
Title	Optional title for plot
sims	Optional subset MSE object by simulation
msg	Logical. Print out messages?
cex.names	Size of names
incRef	Logical. Include the reference methods?
...	Optional additional arguments passed to barplot

Author(s)

A. Hordyk

`betaconv` *Calculate beta parameter for beta distribution from mean and standard deviation*

Description

Calculate beta parameter for beta distribution from mean and standard deviation

Usage

```
betaconv(m, sd)
```

Arguments

m	mean
sd	standard deviation

Value

numeric

Author(s)

T. Carruthers

BK *Beddington and Kirkwood life-history MP (simple version)*

Description

Sets an OFL according to current abundance and an approximation of Fmax based on length at first capture.

Usage

```
BK(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object.
Data	A data-limited methods data object.
reps	The number of stochastic samples of the TAC recommendation

Note

This is the simple version of the BK MP. The paper has a more complex approach that might work better.

Author(s)

T. Carruthers.

References

Beddington, J.R., Kirkwood, G.P., 2005. The estimation of potential yield and stock status using life history parameters. Philos. Trans. R. Soc. Lond. B Biol. Sci. 360, 163-170.

BK_CC *Beddington and Kirkwood life-history method combined with catch curve analysis*

Description

Calculates an OFL using a catch curve estimate of current F and an approximation of FMSY based on length at first capture.

Usage

```
BK_CC(x, Data, reps = 100, Fmin=0.005)
```

Arguments

x	Position in a data-limited methods data object
Data	A data-limited methods data object (class Data)
reps	The number of samples of the TAC recommendation
Fmin	The minimum fishing mortality rate that is derived from the catch-curve (interval censor)

Author(s)

T. Carruthers

References

Beddington, J.R., Kirkwood, G.P., 2005. The estimation of potential yield and stock status using life history parameters. Philos. Trans. R. Soc. Lond. B Biol. Sci. 360, 163-170.

BK_ML — *Beddington and Kirkwood life-history analysis with mean-length estimator of current abundance*

Description

Uses an approximation to FMSY based on length at first capture and an estimate of current abundance based on a mean-length estimator.

Usage

```
BK_ML(x, Data, reps = 100)
```

Arguments

x	Position in a data-limited methods data object
Data	A data-limited methods data object (class Data)
reps	The number of samples of the TAC recommendation

Note

The mean length extension was programmed by Gary Nelson as part of his excellent R package 'fishmethods'

Author(s)

T. Carruthers

References

Beddington, J.R., Kirkwood, G.P., 2005. The estimation of potential yield and stock status using life history parameters. Philos. Trans. R. Soc. Lond. B Biol. Sci. 360, 163-170.

Blow_opt *Blow internal parallel optimization function*

Description

Find the current biomass at which it would take HZN mean generation times to reach Bfrac x SSBMSY biomass level given zero catches

Usage

```
Blow_opt(lnq, SSBMSYc, MGThorizon, Fc, Perrc, Mc, hc, Mac, Wac, R0c, Vc, nyears,
  maxage, movc, Spat_targc, SRrelc, aRc, bRc, Bfrac, mode = 1)
```

Arguments

lnq	number: estimate of log catchability
SSBMSYc	number: spawning biomass at MSY
MGThorizon	number: MGT x HZN
Fc	vector nyears long of fishing mortality rate
Perrc	vector nyears+maxage-1 long of recruitment devitions
Mc	matrix maxage by nyears+proyears of natural mortality rate
hc	number: steepness values
Mac	matrix nages by nyears+proyears of maturity at age
Wac	vector nages long of weight at age
R0c	number: unfished recruitment
Vc	matrix of vulnerability maxage x nyears
nyears	integer: number of historical years
maxage	integer: maximum age
movc	matrix of movement 2 x 2
Spat_targc	number: spatial targetting parameters
SRrelc	integer representing recruitmetn dynamics type 1: Bev Holt 2: Ricker
aRc	number: recruitment parameter
bRc	number: recruitment parameter
Bfrac	fraction of SSBMSY that is the target
mode	1: find Blow 2:report blow 3:plot results

Author(s)

T. Carruthers

`Bluefin_tuna` *Bluefin_tuna Stock*

Description

An object of class Stock

Usage

```
Bluefin_tuna
```

Format

An object of class `Stock` of length 1.

`Bluefin_tuna_WAtl` *Bluefin_tuna_WAtl Stock*

Description

An object of class Stock

Usage

```
Bluefin_tuna_WAtl
```

Format

An object of class `Stock` of length 1.

`Blue_shark` *Blue_shark Stock*

Description

An object of class Stock

Usage

```
Blue_shark
```

Format

An object of class `Stock` of length 1.

boxplot.Data *Boxplot of TAC recommendations*

Description

Boxplot of TAC recommendations

Usage

```
## S3 method for class 'Data'
boxplot(x, upq = 0.9, lwq = 0.1, outline = FALSE, ...)
```

Arguments

x	An object of class MSE
upq	Upper quantile of TACs for max ylim
lwq	Lower quantile of TACs for min ylim
outline	Logical. Include outliers in plot?
...	Optional additional arguments passed to boxplot

Value

Returns a data frame containing the information shown in the plot

Author(s)

A. Hordyk

boxplot.MSE *Boxplot of MP performance from MSE object*

Description

Boxplot of MP performance from MSE object

Usage

```
## S3 method for class 'MSE'
boxplot(x, MPs = NA, maxMP = 8, PMRefs = list(B_BMSY = 1,
  SSB_SSB0 = 0.2, F_FMSY = 1, AAVY = 30, AAVE = 30), lastYrs = 10,
  cex.lab = 1.2, cex.PM = 0.75, canMPs = NULL, cols = TRUE,
  outline = FALSE, CexName = 1.25, incLine = TRUE, incref = FALSE,
  Names = TRUE, ...)
```

Arguments

`x`	An object of class MSE
`MPs`	Optional subset MSE object by MP
`maxMP`	Maximum number of MPs to plot
`PMRefs`	List containing the Performance Metrics reference points. Options are `'SSB_SSB0', 'B_BMSY', 'F_FMSY`
`lastYrs`	Last number of years in projection to calculate statistics
`cex.lab`	Size of axis label text
`cex.PM`	Size of performacne metric text
`canMPs`	Optional character vector of MPs that can be applied (plotted in different colour)
`cols`	Optional vector of colours
`outline`	Logical. Include outliers in boxplot?
`CexName`	Size of the names
`incLine`	Logical. Include vertical line?
`incref`	Logical. Include reference methods?
`Names`	Logical. Include MP names in plot?
`...`	Additional arguments to be passed to plotting functions

Author(s)

A. Hordyk

`Butterfish` *Butterfish Stock*

Description

An object of class Stock

Usage

```
Butterfish
```

Format

An object of class `Stock` of length 1.

calcMean *Calculate Mean Probabilty*

Description

Calculate Mean Probabilty

Usage

```
calcMean(Prob, MSEobj)
```

Arguments

Prob	Prob slot from an object of class PMobj
MSEobj	An object of class MSE

CalcOutput *Apply output control recommendations and calculate population dynamics*

Description

Apply output control recommendations and calculate population dynamics

Usage

```
CalcOutput(y, Asize, TACused, TAC_f, lastCatch, availB, maxF, Biomass_P,
  VBiomass_P, CB_P, CB_Pret, FM_P, Z_P, Spat_targ, V_P, retA_P, M_ageArray, qs,
  nyears, nsim, maxage, nareas)
```

Arguments

y	Projection year
Asize	relative size of areas (matrix nsim by nareas)
TACused	TAC recommendation
TAC_f	Implementation error on TAC
lastCatch	Catch from last year
availB	Total available biomass
maxF	Maximum fishing mortality
Biomass_P	Numeric array (nsim, maxage, proyears, nareas) with Biomass at age
VBiomass_P	Numeric array (nsim, maxage, proyears, nareas) with Vulnerable Biomass at age
CB_P	Numeric array (nsim, maxage, proyears, nareas) with Catch Biomass at age

CB_Pret	Numeric array (nsim, maxage, proyears, nareas) with Retained catch biomass at age
FM_P	Numeric array (nsim, maxage, proyears, nareas) with fishing mortality at age
Z_P	Numeric array (nsim, maxage, proyears, nareas) with total mortality at age
Spat_targ	Spatial targetting
V_P	Numeric array(nsim, maxage, nyears+proyears) with vulnerability at age
retA_P	Numeric array(nsim, maxage, nyears+proyears) with retention at age
M_ageArray	Numeric array (nsim, maxage, nyears+proyears) Natural mortality at age
qs	Catchability coefficient
nyears	Number of historical years
nsim	Number of simulations
maxage	Maximum age
nareas	Number of areas

Author(s)

A. Hordyk

`calcProb` *Calculate Probabilty*

Description

Calculate Probabilty

Usage

```
calcProb(PM, MSEobj)
```

Arguments

PM	A PM method
MSEobj	An object of class MSE

Can — *What data-limited methods can be applied to this Data object?*

Description

An diagnostic tool that looks up the slot requirements of each method and compares this to the data available to limit the analysis to methods that have the correct data, do not produce errors and run within a time limit. Time limit is the maximum time taken to carry out five reps (stochastic samples) of a given method and is in units of seconds.

Usage

```
Can(Data, timelimit = 1)
```

Arguments

Data	A data-limited methods data object (class Data)
timelimit	The maximum time (seconds) taken for a method to undertake 10 reps (this filters out methods that are too slow)

Cant — *What methods can't be applied to this DLM data object*

Description

The methods that don't have sufficient data, lead to errors or don't run in time along with a list of their data requirments.

Usage

```
Cant(Data, timelimit = 1)
```

Arguments

Data	A data-limited methods data object (class Data)
timelimit	The maximum time (seconds) taken for a method to undertake 10 reps (this filters out methods that are too slow)

CC1 *Constant catch management procedure of Geromont and Butterworth (2014)*

Description

The TAC is the average catch over last yrsmth years.

Usage

```
CC1(x, Data, reps = 100, yrsmth = 5, xx=0)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrsmth	Years over which to calculate mean catches
xx	Parameter controlling the TAC. Mean catches are multiplied by (1-xx)

Details

This is one of four constant catch rules of Geromont and Butterworth 2014.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

CC4 *Constant catch management procedure of Geromont and Butterworth (2014)*

Description

The TAC is the average catch over last yrsmth years reduced by 30

Usage

```
CC4(x, Data, reps = 100, yrsmth = 5, xx=0.3)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrsmth	Years over which to average catches
xx	Parameter controlling the TAC. Mean catches are multiplied by (1-xx)

Details

This is one of four constant catch MPs of Geromont and Butterworth 2014.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

CheckConverg *MSE convergence diagnostic*

Description

Have I undertaken enough simulations (nsim)? Has my MSE converged on stable (reliable) peformance metrics?

Usage

```
CheckConverg(MSEobj,thresh=2, Plot=TRUE)
```

Arguments

MSEobj	An object of class 'MSE'
thresh	The convergence threshold (percentage). If mean perforamnce metrics are within thresh percent of the second to last interation, the MSE can be considered to have converged.
Plot	Should figures be plotted?

Author(s)

A. Hordyk

checkMSE *Check that MSE object includes all slots*

Description

Check that an MSE object includes all slots in the latest version of DLMtool Use 'updateMSE' to update the MSE object

Usage

```
checkMSE(MSEobj)
```

Arguments

MSEobj	A MSE object.

Author(s)

A. Hordyk

China_rockfish *China_rockfish Data*

Description

An object of class Data

Usage

```
China_rockfish
```

Format

An object of class `Data` of length 1.

ChkDatNA *Check Data object is valid for a MP*

Description

Checks that all slots in Data object required by the MP contain finite values

Usage

```
ChkDatNA(Data, dependencies)
```

Arguments

`Data`	An object of class Data
`dependencies`	A string of slots in the Data object required for the MP

Author(s)

A. Hordyk

ChkObj *Check that a DLM object is valid*

Description

Check that all slots in Object are valid and contain values

Usage

```
ChkObj(OM)
```

Arguments

OM	An object of class OM, Stock, Fleet, Obs, or Imp

ChooseEffort *Manually map the historical relative fishing effort trajectory.*

Description

Interactive plot which allows users to specify the relative trajectory and variability in the historical fishing effort and populates Fleet object.

Usage

```
ChooseEffort(FleetObj, Years=NULL)
```

Arguments

FleetObj	A fleet object.
Years	An optional vector of years. Should be nyears long.

Author(s)

A. Hordyk

ChooseM *Manually map natural mortality at age or size.*

Description

Interactive plot which allows users to specify M by age or size class

Usage

```
ChooseM(OM, type = c("age", "length"), x = NULL, y = NULL)
```

Arguments

OM	An object of class 'OM'
type	A character string - is M to be mapped by 'age' or 'length'?
x	Optional vector for x-axis
y	Optional vector for y-axis

Author(s)

A. Hordyk

ChooseSelect *Manually choose the historical selectivity pattern*

Description

Input the first historical year, and all years where selectivity pattern changed (separated by comma). Interactive plot which allows users to specify a range for the length at 5% and full selection (LFS), as well as selectivity at maximum length for each year. Produces a simple plot which shows the range in selectivity pattern for each break-point year. Selectivity-at-length is fixed in between break-point years. Note that this function replaces 'nyears' in the Fleet object with the value defined here (FstYr:current year).

Usage

```
ChooseSelect(Fleet, Stock, FstYr = NULL, SelYears = NULL)
```

Arguments

Fleet	A fleet object.
Stock	Optional Stock object. If provided, average length-at-maturity is included on plot for reference.
FstYr	Optional value for first historical year. If empty, user must specify the year in console.
SelYears	Optional vector of values for each year where selectivity pattern changed. If empty, user must specify the years in console (comma separated).

Author(s)

A. Hordyk

`Cobia` *Cobia Data*

Description

An object of class Data

Usage

```
Cobia
```

Format

An object of class `Data` of length 1.

`compplot` *Generic comparison plot for simulation testing of Stochastic SRA method*

Description

Plots simulation variables versus estimation variables for Stochastic SRA methods of conditioning operating models.

Usage

```
compplot(simy, samy, xlab = "", ylab = "", maxplot = 10, type = "l")
```

Arguments

`simy`	The simulated time series
`samy`	The matrix of estimated time series from of StochasticSRA() function.
`xlab`	The x axis label for the plot
`ylab`	The y axis label for the plot
`maxplot`	The total number of individual simulations to be plotted in the first plot
`type`	Should a line 'l' or points 'p' be plotted?

Value

A plot

Author(s)

T. Carruthers (Canadian DFO grant)

Examples

```
nyears<-100
nsims<-200
simy<-sin(seq(0,2,length.out=nyears))
samy<-array(rep(simy,each=nsims)*rnorm(nsims,1,0.2)*rnorm(nsims*nyears,1,0.1),c(nsims,nyears))
par(mfrow=c(1,2))
compplot(simy,samy,xlab="Year",ylab="Some time varying parameter")
```

`CompSRA` *Age-composition-based estimate of current stock depletion given constant Z linked to an FMSY estimate to provide OFL*

Description

Estimates an OFL based on a Stock Reduction analysis fitted to current age-composition data. Knife-edge vulnerability at age at maturity allows for an FMSY estimate. OFL=FMSY*F/C

Usage

```
CompSRA(x, Data, reps = 100)
```

Arguments

`x`	A position in a data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of stochastic samples of the TAC.

Note

Given a fixed historical F, What level of depletion gives you this length composition?

Author(s)

T. Carruthers

CompSRA4010 *Age-composition-based estimate of current stock depletion given constant Z linked to an FMSY estimate to provide OFL (with a 40-10 rule)*

Description

Estimates an OFL based on a Stock Reduction analysis fitted to current age-composition data. Knife-edge vulnerability at age at maturity allows for an FMSY estimate. OFL=FMSY*F/C

Usage

```
CompSRA4010(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC.

Note

Given a fixed historical F, What level of depletion gives you this length composition?

Author(s)

T. Carruthers

condmet *Condition met?*

Description

Condition met?

Usage

```
condmet(vec)
```

Arguments

vec	vector of logical values

Converge *Check Convergence*

Description

Have I undertaken enough simulations (nsim)? Has my MSE converged on stable (reliable) peformance metrics?

Usage

```
Converge(MSEobj, thresh = 2, Plot = TRUE)
```

Arguments

MSEobj	An MSE object of class 'MSE'
thresh	The convergence threshold (percentage). If mean performance metrics are within thresh percent of the second to last interation, the MSE can be considered to have converged.
Plot	Should figures be plotted?

Author(s)

A. Hordyk

COSEWIC_plot *COSEWIC forward projection plot*

Description

Projection of biomass under three scenarios: no catch, FMSY fishing and status quo fishing This plot if for an MSE object created from runMSE with the argument MPs=c("NFref","FMSYref","curE")

Usage

```
COSEWIC_plot(MSEobj, syear = 2015)
```

Arguments

MSEobj	An object of class MSE created from runMSE() with the argument MPs=c("NFref","FMSYref","curE")
syear	Starting year of the projection for graphing purposes

Value

A plot

Author(s)

T. Carruthers

cparscheck — *Internal function of runMSE for checking that the OM slot cpars slot is formatted correctly*

Description

Internal function of runMSE for checking that the OM slot cpars slot is formatted correctly

Usage

```
cparscheck(cpars)
```

Arguments

cpars	a list of model parameters to be sampled (single parameters are a vector nsim long, time series are matrices nsim x nyears)

Value

either an error and the length of the first dimension of the various cpars list items or passes and returns the number of simulations

Author(s)

T. Carruthers

Cplot — *Plot the median biomass and yield relative to last historical year*

Description

Compare median biomass and yield in first year and last 5 years of projection

Usage

```
Cplot(MSEobj, MPs = NA, lastYrs = 5, XMin = NULL, YMin = NULL,
ShowLabs = FALSE)
```

Arguments

MSEobj	An object of class MSE
MPs	Optional subset by MP
lastYrs	Last number of years of projection to calculate median
XMin	Optional minimum for the x-axis
YMin	Optional minimum for the y-axis
ShowLabs	Logical. Show the MP labels? Otherwise only plot points

Value

Invisibly returns a data frame containing information shown in the plot

Author(s)

A. Hordyk

CSRA *Catch at size reduction analysis*

Description

What depletion level and corresponding equlibrium F arise from data regarding mean length of current catches, natural mortality rate, steepness of the stock recruitment curve, maximum length, maximum growth rate, age at maturity, age based vulnerability, maturity at age, maximum age and number of historical years of fishing.

Usage

```
CSRA(M,h,Linf,K,t0,AM,a,b,vuln,mat,ML,CAL,CAA,maxage,nyears)
```

Arguments

M	A vector of natural mortality rate estimates
h	A vector of sampled steepness (Beverton-Holt stock recruitment)
Linf	A vector of maximum length (von Bertalanffy growth)
K	A vector of maximum growth rate (von Bertalanffy growth)
t0	A vector of theoretical age at length zero (von Bertalanffy growth)
AM	A vector of age at maturity
a	Length-weight conversion parameter a (W=aL^b)
b	Length-weight conversion parameter b (W=aL^b)
vuln	A matrix nsim x nage of the vulnerabilty at age (max 1) to fishing.
mat	A matrix nsim x nage of the maturity at age (max 1)
ML	A vector of current mean length estimates
CAL	A catch-at-length matrix nyears x (1 Linf unit) length bins
CAA	A catch-at-age matrix nyears x maximum age
maxage	Maximum age
nyears	Number of historical years of fishing

Author(s)

T. Carruthers

CSRAfunc *Optimization function for CSRA*

Description

What depletion level and corresponding equlibrium F arise from data regarding mean length of current catches, natural mortality rate, steepness of the stock recruitment curve, maximum length, maximum growth rate, age at maturity, age based vulnerability, maturity at age, maximum age and number of historical years of fishing.

Usage

```
CSRAfunc(lnF,Mc,hc,maxage,nyears,AFSc,AFCc,Linfc,
Kc,t0c,AMc,ac,bc,vulnc,matc,MLc,CAL,CAA, opt=T,meth='ML')
```

Arguments

lnF	A proposed value of current instantaneous fishing mortality rate
Mc	Natural mortality rate estimates
hc	Steepness (Beverton-Holt stock recruitment)
maxage	Maximum age
nyears	Number of historical years of fishing
AFSc	Age at full selection
AFCc	Age at first capture
Linfc	Maximum length (von Bertalanffy growth)
Kc	Maximum growth rate (von Bertalanffy growth)
t0c	Theoretical age at length zero (von Bertalanffy growth)
AMc	Age at maturity
ac	Length-weight conversion parameter a (W=aL^b)
bc	Length-weight conversion parameter b (W=aL^b)
vulnc	A vector (nage long) of the vulnerabilty at age (max 1) to fishing.
matc	A vector (nage long) of the maturity at age (max 1)
MLc	A current mean length estimates
CAL	A catch-at-length matrix nyears x (1 Linf unit) length bins
CAA	A catch-at-age matrix nyears x maximum age
opt	Should the measure of fit be returned?
meth	Are we fitting to mean length or catch composition?

Author(s)

T. Carruthers

curE *Fishing at current effort levels*

Description

Constant fishing effort set at final year of historical simulations subject to changes in catchability determined by OM@qinc and interannual variability in catchability determined by OM@qcv. This MP is intended to represent a 'status quo' management approach.

Usage

```
curE(x, Data, ...)
```

Arguments

x	A position in a data-limited methods data object.
Data	A data-limited methods data object.
...	Optional additional arguments that are ignored. Note arguments reps or ... are required for all input controls

Note

Made up for this package.

Author(s)

T. Carruthers.

curE75 *Fishing at 75 per cent of current effort levels*

Description

Constant fishing effort set at 75 per cent of final year of historical simulations subject to changes in catchability determined by OM@qinc and interannual variability in catchability determined by OM@qcv. This MP is intended to represent a 'status quo' management approach.

Usage

```
curE75(x, Data, ...)
```

Arguments

x	A position in a data-limited methods data object.
Data	A data-limited methods data object.
...	Optional additional arguments that are ignored. Note arguments reps or ... are required for all input controls

Note

Made up for this package.

Author(s)

T. Carruthers.

---

cv *Calculate CV from vector of values*

Description

Calculate CV from vector of values

Usage

```
cv(x)
```

Arguments

x	vector of numeric values

Value

numeric

Author(s)

T. Carruthers

DAAC *Depletion Adjusted Average Catch*

Description

Essentially DCAC multiplied by 2*depletion and divided by BMSY/B0 (Bpeak)

Usage

```
DAAC(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Author(s)

W. Harford and T. Carruthers

References

MacCall, A.D., 2009. Depletion-corrected average catch: a simple formula for estimating sustainable yields in data-poor situations. ICES J. Mar. Sci. 66, 2267-2271. Harford W. and Carruthers, T. 2016. Simulation testing novel catch-based fisheries management. In draft, intended for Fish. Bull.

Data-class *Class 'Data'*

Description

An object for storing data for analysis using data-limited methods

Slots

Name The name of the Data object. Single value. Character string

Year Years that corresponding to catch and relative abundance data. Vector nyears long. Positive integer

Cat Total annual catches. Matrix of nsim rows and nyears columns. Non-negative real numbers

Ind Relative abundance index. Matrix of nsim rows and nyears columns. Non-negative real numbers

t The number of years corresponding to AvC and Dt. Single value. Positive integer

AvC Average catch over time t. Vector nsim long. Positive real numbers

`Dt` Depletion over time t SSB(now)/SSB(now-t+1). Vector nsim long. Fraction

`ML` Mean length time series. Matrix of nsim rows and nyears columns. Non-negative real numbers

`Mort` Natural mortality rate. Vector nsim long. Positive real numbers

`FMSY_M` An assumed ratio of FMSY to M. Vector nsim long. Positive real numbers

`BMSY_B0` The most productive stock size relative to unfished. Vector nsim long. Fraction

`L50` Length at 50 percent maturity. Vector nsim long. Positive real numbers

`L95` Length at 95 percent maturity. Vector nsim long. Positive real numbers

`Lbar` Mean length of catches over Lc. Matrix of nsim rows and nyears columns. Positive real numbers

`Lc` Modal length of catches. Matrix of nsim rows and nyears columns. Positive real numbers

`LFC` Length at first capture. Matrix of nsim rows and nyears columns. Positive real numbers

`LFS` Shortest length at full selection. Matrix of nsim rows and nyears columns. Positive real numbers

`CAA` Catch at Age data. Array of dimensions nsim x nyears x MaxAge. Non-negative integers

`Dep` Stock depletion SSB(current)/SSB(unfished). Vector nsim long. Fraction.

`Abun` An estimate of absolute current vulnerable abundance. Vector nsim long. Positive real numbers

`SpAbun` An estimate of absolute current spawning stock abundance. Vector nsim long. Positive real numbers

`vbK` The von Bertalanffy growth coefficient K. Vector nsim long. Positive real numbers

`vbLinf` Maximum length. Vector nsim long. Positive real numbers

`vbt0` Theoretical age at length zero. Vector nsim long. Non-positive real numbers

`LenCV` Coefficient of variation of length-at-age (assumed constant for all age classes). Vector nsim long. Positive real numbers

`wla` Weight-Length parameter alpha. Vector nsim long. Positive real numbers

`wlb` Weight-Length parameter beta. Vector nsim long. Positive real numbers

`steep` Steepness of stock-recruitment relationship. Vector nsim long. Value in the range of one-fifth to 1

`CV_Cat` Coefficient of variation in annual catches. Vector nsim long. Positive real numbers

`CV_Dt` Coefficient of variation in depletion over time t. Vector nsim long. Positive real numbers

`CV_AvC` Coefficient of variation in average catches over time t. Vector nsim long. Positive real numbers

`CV_Ind` Coefficient of variation in the relative abundance index. Vector nsim long. Positive real numbers

`CV_Mort` Coefficient of variation in natural mortality rate. Vector nsim long. Positive real numbers

`CV_FMSY_M` Coefficient of variation in the ratio in FMSY/M. Vector nsim long. Positive real numbers

`CV_BMSY_B0` Coefficient of variation in the position of the most productive stock size relative to unfished. Vector nsim long. Positive real numbers

`CV_Dep` Coefficient of variation in current stock depletion. Vector nsim long. Positive real numbers

`CV_Abun` Coefficient of variation in estimate of absolute current stock size. Vector nsim long. Positive real numbers

`CV_vbK` Coefficient of variation in the von Bertalanffy K parameter. Vector nsim long. Positive real numbers

`CV_vbLinf` Coefficient of variation in maximum length. Vector nsim long. Positive real numbers

`CV_vbt0` Coefficient of variation in age at length zero. Vector nsim long. Positive real numbers

`CV_L50` Coefficient of variation in length at 50 per cent maturity. Vector nsim long. Positive real numbers

`CV_LFC` Coefficient of variation in length at first capture. Vector nsim long. Positive real numbers

`CV_LFS` Coefficient of variation in length at full selection. Vector nsim long. Positive real numbers

`CV_wla` Coefficient of variation in weight-length parameter a. Vector nsim long. Positive real numbers

`CV_wlb` Coefficient of variation in weight-length parameter b. Vector nsim long. Positive real numbers

`CV_steep` Coefficient of variation in steepness. Vector nsim long. Positive real numbers

`sigmaL` Assumed observaton error of the length composition data. Vector nsim long. Positive real numbers

`MaxAge` Maximum age. Vector nsim long. Positive integer

`Units` Units of the catch/absolute abundance estimates. Single value. Character string

`Ref` A reference management level (eg a catch limit). Single value. Positive real number

`Ref_type` Type of reference management level (eg 2009 catch limit). Single value. Character string

`Log` A record of events. Single value. Character string

`params` A place to store estimated parameters. An object. R list

`PosMPs` The methods that can be applied to these data. Vector. Character strings

`MPs` The methods that were applied to these data. Vector. Character strings

`OM` A table of operating model conditions. R table object of nsim rows. Real numbers

`Obs` A table of observation model conditions. R table object of nsim rows. Real numbers

`TAC` The calculated catch limits (function TAC). An array with dimensions PosMPs x replicate TAC samples x nsim. Positive real numbers

`Sense` The results of the sensitivity analysis (function Sense). An array with dimensions PosMPs x sensitivity increments. Positive real numbers

`CAL_bins` The values delimiting the length bins for the catch-at-length data. Vector. Non-negative real numbers

`CAL` Catch-at-length data. An array with dimensions nsim x nyears x length(CAL_bins). Non-negative integers

`Cref` Reference or target catch level (eg MSY). Vector of length nsim. Positive real numbers

`Iref` Reference or target relative abundance index level (eg BMSY / B0). Vector of length nsim. Positive real numbers

`Bref` Reference or target biomass level (eg BMSY). Vector of length nsim. Positive real numbers

CV_Cref Log-normal CV for reference or target catch level. Vector of length nsim. Positive real numbers

CV_Iref Log-normalCV for reference or target relative abundance index level. Vector of length nsim. Positive real numbers

CV_Bref Log-normal CV for reference or target biomass level. Vector of length nsim. Positive real numbers

CV_Rec Log-normal CV for recent recruitment strength. Vector of length nsim. Positive real numbers

Rec Recent recruitment strength. Vector of length nsim. Positive real numbers

MPrec The previous recommendation of a management procedure. Vector of length nsim. Positive real numbers

MPeff The current level of effort. Vector of length nsim. Positive real numbers

LHYear The last historical year of the simulation (before projection). Single value. Positive integer

nareas Number of fishing areas. Vector of length nsim. Non-negative integer

Misc Other information for MPs. An object. R list

Objects from the Class

Objects can be created by calls of the form `new('Data', stock)`

Author(s)

T. Carruthers and A. Hordyk

Examples

```
newdata<-new('Data')
```

DataDescription *DataDescription*

Description

A data.frame with description of slots for class Data

Usage

```
DataDescription
```

Format

An object of class `data.frame` with 75 rows and 2 columns.

Data_xl *Read in Data object from Excel spreadsheet*

Description

A function to read in Data object from an Excel spreadsheet with tabs named following specific convention.

Usage

```
Data_xl(fname, stkname, fpath = "", saveCSV = FALSE)
```

Arguments

fname	Name of the Excel spreadsheet file. Must include file extension.
stkname	Name of the Stock.
fpath	Full file path, if file is not in current working directory
saveCSV	Do you also want to the Data parameters to a CSV file?

Details

The Excel spreadsheet must have tabs named with the following convention. For example if stkname is 'myFish', the Data parameters are in a tab named 'myFishData'.

Value

A object of class Data

Author(s)

A. Hordyk

Examples

```
## Not run:
OM <- OM_xl(fname='OMTables.xlsx', stkname='myFish')

## End(Not run)
```

DBSRA *Depletion-Based Stock Reduction Analysis*

Description

User prescribed BMSY/B0, M, FMSY/M are used to find B0 and therefore the OFL by back-constructing the stock to match a user specified level of stock depletion (OFL = M * FMSY/M * depletion* B0).

Usage

```
DBSRA(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods object.
Data	A data-limited methods object.
reps	The number of samples of the TAC (OFL) recommendation.

Details

You specify a range of stock depletion and, given historical catches DB-SRA calculates what unfished biomass must have been to get you here given samples for M, FMSY relative to M and also BMSY relative to Bunfished.

Value

A vector of TAC (OFL) values.

Note

This is set up to return the OFL (FMSY * current biomass).

You may have noticed that you -the user- specify three of the factors that make the quota recommendation. So this can be quite a subjective method.

Also the DB-SRA method of this package isn't exactly the same as the original method of Dick and MacCall (2011) because it has to work for simulated depletions above BMSY/B0 and even on occasion over B0. Also it doesn't have the modification for flatfish life histories that has previously been applied by Dick and MacCall.

Author(s)

T. Carruthers

References

Dick, E.J., MacCall, A.D., 2011. Depletion-Based Stock Reduction Analysis: A catch-based method for determining sustainable yields for data-poor fish stocks. Fish. Res. 110, 331-341.

DBSRA4010 — *Depletion-Based Stock Reduction Analysis paired with 40-10 harvest control rule*

Description

User prescribed BMSY/B0, M, FMSY/M are used to find B0 and therefore the OFL by back-constructing the stock to match a user specified level of stock depletion (OFL = M * FMSY/M * depletion* B0). In this method DBSRA is paried with the 40-10 rule that throttles back the OFL to zero at 10 percent of unfished biomass.

Usage

```
DBSRA4010(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Author(s)

T. Carruthers

References

Dick, E.J., MacCall, A.D., 2011. Depletion-Based Stock Reduction Analysis: A catch-based method for determining sustainable yields for data-poor fish stocks. Fish. Res. 110, 331-341.

DBSRA_40 — *Depletion-Based Stock Reduction Analysis assuming 40 per cent stock depletion*

Description

DBSRA assuming that current stock depletion is exactly 40 per cent of unfished stock levels.

Usage

```
DBSRA_40(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Note

A 40 percent assumption for current depletion is more or less the most optimistic state for a stock (ie very close to BMSY/B0 for many stocks).

Author(s)

T. Carruthers.

References

Dick, E.J., MacCall, A.D., 2010. Estimates of sustainable yield for 50 data-poor stocks in the Pacific Coast groundfish fishery management plan. Technical memorandum. Southwest fisheries Science Centre, Santa Cruz, CA. National Marine Fisheries Service, National Oceanic and Atmospheric Administration of the U.S. Department of Commerce. NOAA-TM-NMFS-SWFSC-460.

DBSRA_ML *Depletion-Based Stock Reduction Analysis using mean length estimator of stock depletion*

Description

DBSRA using the mean length estimator to calculate current stock depletion.

Usage

```
DBSRA_ML(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the quota recommendation

Note

The mean length extension was programmed by Gary Nelson as part of his excellent R package 'fishmethods'

Author(s)

T. Carruthers

References

Dick, E.J., MacCall, A.D., 2011. Depletion-Based Stock Reduction Analysis: A catch-based method for determining sustainable yields for data-poor fish stocks. Fish. Res. 110, 331-341.

DCAC *Depletion Corrected Average Catch*

Description

A method of calculating an MSY proxy (FMSY * BMSY and therefore the OFL at most productive stock size) based on average catches accounting for the windfall catch that got the stock down to BMSY levels.

Usage

```
DCAC(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Note

It's probably worth noting that DCAC TAC recommendations do not tend to zero as depletion tends to zero. It adjusts for depletion only in calculating historical average catch. It follows that at stock levels much below BMSY, DCAC tends to chronically overfish.

Author(s)

T. Carruthers

References

MacCall, A.D., 2009. Depletion-corrected average catch: a simple formula for estimating sustainable yields in data-poor situations. ICES J. Mar. Sci. 66, 2267-2271.

DCAC4010 *Depletion Corrected Average Catch paired with the 40-10 rule*

Description

A method of calculating an MSY proxy (FMSY * BMSY and therefore the OFL at most productive stock size) based on average catches accounting for the windfall catch that got the stock down to BMSY levels. In this method DCAC is paired with the 40-10 rule that throttles back the OFL to zero at 10 percent of unfished stock size (the OFL is not subject to downward adjustment above 40 percent unfished)

Usage

```
DCAC4010(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Note

DCAC can overfish below BMSY levels. The 40-10 harvest control rule largely resolves this problem providing an MP with surprisingly good performance even at low stock levels.

Author(s)

T. Carruthers

References

MacCall, A.D., 2009. Depletion-corrected average catch: a simple formula for estimating sustainable yields in data-poor situations. ICES J. Mar. Sci. 66, 2267-2271.

DCAC_40 *Depletion Corrected Average Catch assuming 40 per cent stock depletion*

Description

DCAC assuming that current stock biomass is exactly 40 per cent of unfished levels.

Usage

```
DCAC_40(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Note

The 40 percent depletion assumption doesn't really affect DCAC that much as it already makes TAC recommendations that are quite MSY-like.

Author(s)

T. Carruthers

References

MacCall, A.D., 2009. Depletion-corrected average catch: a simple formula for estimating sustainable yields in data-poor situations. ICES J. Mar. Sci. 66, 2267-2271.

DCAC_ML *Depletion-Based Stock Reduction Analysis using mean-length estimator of current depletion*

Description

DCAC that uses the mean length estimator to calculate current stock depletion.

Usage

```
DCAC_ML(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Note

The mean length extension was programmed by Gary Nelson as part of his excellent R package 'fishmethods'

Author(s)

T. Carruthers

References

MacCall, A.D., 2009. Depletion-corrected average catch: a simple formula for estimating sustainable yields in data-poor situations. ICES J. Mar. Sci. 66, 2267-2271.

DD *Delay - Difference Stock Assessment with UMSY and MSY leading*

Description

A simple delay-difference assessment that estimates the TAC using a time-series of catches and a relative abundance index.

Usage

```
DD(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Value

A numeric vector of TAC recommendations

Note

This DD model is observation error only and has does not estimate process error (recruitment deviations). Similar to many other assessment models it depends on a whole host of dubious assumptions such as temporally stationary productivity and proportionality between the abundance index and real abundance. Unsurprisingly the extent to which these assumptions are violated tends to be the biggest driver of performance for this method.

Author(s)

T. Carruthers

References

Method based on equations of Carl Walters (bug him with questions and expect colourful responses)

DD4010 — *Delay - Difference Stock Assessment with UMSY and MSY leading coupled with a 40-10 harvest control rule*

Description

A simple delay-difference assessment that estimates the OFL using a time-series of catches and a relative abundance index. In this version of the DD MP a 40-10 rule is imposed over the OFL recommendation.

Usage

```
DD4010(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Method based on equations of Carl Walters

DDe — *Effort control version of DD - Delay - Difference Stock Assessment with UMSY and MSY leading*

Description

A simple delay-difference assessment that estimates and recommends FMSY using a time-series of catches and a relative abundance index.

Usage

```
DDe(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Note

This DD model is observation error only and has does not estimate process error (recruitment deviations). Similar to many other assessment models it depends on a whole host of dubious assumptions such as temporally stationary productivity and proportionality between the abundance index and real abundance. Unsurprisingly the extent to which these assumptions are violated tends to be the biggest driver of performance for this method.

Author(s)

T. Carruthers

References

Method based on equations of Carl Walters (bug him with questions and expect colourful responses)

DDe75 — *Effort control version of DD - Delay - Difference Stock Assessment with UMSY and MSY leading that fishes at 75 per cent of FMSY*

Description

A simple delay-difference assessment that estimates and recommends 75 per cent FMSY using a time-series of catches and a relative abundance index.

Usage

```
DDe75(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Note

This DD model is observation error only and has does not estimate process error (recruitment deviations). Similar to many other assessment models it depends on a whole host of dubious assumptions such as temporally stationary productivity and proportionality between the abundance index and real abundance. Unsurprisingly the extent to which these assumptions are violated tends to be the biggest driver of performance for this method.

Author(s)

T. Carruthers

References

Method based on equations of Carl Walters (bug him with questions and expect colourful responses)

`DDes` *Effort searching version of DD - Delay - Difference Stock Assessment with UMSY and MSY leading that fishes at 75 per cent of FMSY*

Description

A simple delay-difference assessment that estimates FMSY using a time-series of catches and a relative abundance index. The MP provides a change in effort in the direction of FMSY up to a maximum change of 10 percent.

Usage

```
DDes(x, Data, reps = 100, LB=0.9, UB=1.1)
```

Arguments

`x`	A position in a data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of stochastic samples of the TAC recommendation
`LB`	The lowest permitted factor of previous fishing effort
`UB`	The highest permitted factor of previous fishing effort

Note

This DD model is observation error only and has does not estimate process error (recruitment deviations). Similar to many other assessment models it depends on a whole host of dubious assumptions such as temporally stationary productivity and proportionality between the abundance index and real abundance. Unsurprisingly the extent to which these assumptions are violated tends to be the biggest driver of performance for this method.

Author(s)

T. Carruthers

References

Method based on equations of Carl Walters (bug him with questions and expect colourful responses)

DecE_Dom *DecE_Dom Fleet*

Description

An object of class Fleet

Usage

```
DecE_Dom
```

Format

An object of class `Fleet` of length 1.

DecE_HDom *DecE_HDom Fleet*

Description

An object of class Fleet

Usage

```
DecE_HDom
```

Format

An object of class `Fleet` of length 1.

DecE_NDom *DecE_NDom Fleet*

Description

An object of class Fleet

Usage

```
DecE_NDom
```

Format

An object of class `Fleet` of length 1.

DepF *Depletion Corrected Fratio*

Description

The Fratio MP with a harvest control rule that reduces F according to the production curve given an estimate of current stock depletion.

Usage

```
DepF(x, Data, reps = 100)
```

Arguments

x	A position in data-limited methods data object DLM
Data	A data-limited methods data object
reps	The number of TAC samples

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Made-up for this package.

derive_beta_par *This function reduces the CV by 5 per cent until steepness values can be sampled without error*

Description

This function reduces the CV by 5 per cent until steepness values can be sampled without error

Usage

```
derive_beta_par(mu, sigma)
```

Arguments

mu	mean h
sigma	sd of h

Author(s)

Q. Huynh

`DFO_hist` *Deparment of Fisheries and Oceans historical plot*

Description

A plot of current and historical stock status by simulation according to the stock status zones and reference points of DFO. http://www.dfo-mpo.gc.ca/reports-rapports/regs/sff-cpd/precaution-eng.htm

Usage

```
DFO_hist(OM, panel = T, nsim = 48)
```

Arguments

`OM`	An operating model object of class OM
`panel`	should the plots be seperate or in two panels?
`nsim`	how many simulations should be plotted (over-ridden by OM@nsim where cpars is specified)

Author(s)

T. Carruthers

`DFO_plot` *Deparment of Fisheries and Oceans trade-off plot*

Description

A plot of mean biomass relative to BMSY and fishing mortality rate relative to FMSY over the final 5 years of the projection http://www.dfo-mpo.gc.ca/reports-rapports/regs/sff-cpd/precaution-eng.htm

Usage

```
DFO_plot(MSEobj)
```

Arguments

`MSEobj`	An MSE object of class MSE produced by DLMtool function runMSE

Author(s)

T. Carruthers

DFO_plot2 *Deparment of Fisheries and Oceans default plot 2*

Description

A preliminary plot for returning trade-offs plots and performance table for probability of obtaining half reference (FMSY) yield and probability of biomass dropping below 50 per cent BMSY

Usage

```
DFO_plot2(MSEobj, nam = NA, panel = T, Bcut = 50, Ycut = 50)
```

Arguments

MSEobj	An object of class MSE
nam	Title of plot
panel	Should the plots be organized in many panels in a single figure
Bcut	The cutoff biomass for satisficing (relative to BMSY)
Ycut	the cutoff yield for satisficing (relative to reference yield)

Value

A table of performance metrics.

Author(s)

T. Carruthers

DFO_proj *Deparment of Fisheries and Oceans projection plot*

Description

A projection plot of MP performance by simulation according to the stock status zones and reference points of DFO. http://www.dfo-mpo.gc.ca/reports-rapports/regs/sff-cpd/precaution-eng.htm

Usage

```
DFO_proj(MSEobj, maxplot = 3)
```

Arguments

MSEobj	An operating model object of class MSE
maxplot	The maximum number of MPs to be plotted per figure

Author(s)

T. Carruthers

`DLMDataDir` *Directory of the installed package on your computer*

Description

A way of locating where the package was installed so you can find example data files and code etc.

Usage

```
DLMDataDir(stock=NA)
```

Arguments

`stock`	Character string representing the name of a .csv file e.g. 'Snapper', 'Rockfish'

Author(s)

T. Carruthers

`DLMextra` *Load more data from DLMextra package*

Description

Downloads the DLMextra package from GitHub

Usage

```
DLMextra(silent = FALSE)
```

Arguments

`silent`	Logical. Should messages to printed?

dnormal *Double-normal selectivity curve*

Description

Double-normal selectivity curve

Usage

```
dnormal(lens, lfs, sl, sr)
```

Arguments

lens	Vector of lengths
lfs	Length at full selection
sl	Sigma of ascending limb
sr	Sigma of descending limb

DOM *How dominant is an MP?*

Description

The DOM function examines how consistently an MP outperforms another. For example DCAC might provide higher yield than AvC on average but outperforms AvC in less than half of simulations.

Usage

```
DOM(MSEobj, MPtg=NA)
```

Arguments

MSEobj	An object of class 'MSE'
MPtg	A character vector of management procedures for cross examination

Value

A matrix of performance comparisons length(MPtg) rows by MSE@nMPs columns

Author(s)

A. Hordyk

DTe40 *Effort searching MP aiming for 40 per cent stock depletion*

Description

A very simple MP that modifies effort to reach 40 percent stock depletion

Usage

```
DTe40(x, Data, reps = 100, alpha=0.4, LB=0.9, UB=1.1)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation
alpha	The target level of depletion
LB	The lowest permitted factor of previous fishing effort
UB	The highest permitted factor of previous fishing effort

Author(s)

T. Carruthers

DTe50 *Effort searching MP aiming for 50 per cent stock depletion*

Description

A very simple MP that modifies effort to reach 50 percent stock depletion

Usage

```
DTe50(x, Data, reps = 100, alpha=0.5, LB=0.9, UB=1.1)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation
alpha	The target level of depletion
LB	The lowest permitted factor of previous fishing effort
UB	The highest permitted factor of previous fishing effort

Author(s)

T. Carruthers

DynF *Dynamic Fratio MP*

Description

The Fratio MP with a controller that changes the level of F according to the relationship between Surplus production and biomass. Ie lower F when dSP/dB is positive and higher F when dSP/dB is negative.

Usage

```
DynF(x, Data, yrsmth=10, gg=2, reps = 100)
```

Arguments

x	A position in a data-limited methods object
Data	A data-limited methods object
yrsmth	The number of historical recent years used for smoothing catch and biomass data
gg	A gain parameter that modifies F according to the gradient in surplus production with biomass
reps	The number samples of the TAC

Details

The method smoothes historical catches and biomass and then infers the relationship between surplus production and biomass (as suggested by Mark Maunder and Carl Walters). The approach then regulates a F based policy according to this gradient in which F may range between two different fractions of natural mortality rate.

The core advantage is the TAC(t) is not strongly determined by TAC(t-1) and therefore errors are not as readily propagated. The result is method that tends to perform alarmingly well and therefore requires debunking ASAP.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Made-up for this package.

EtargetLopt *Effort MP: adjust effort up/down if mean length above/below Ltarget*

Description

Effort MP: adjust effort up/down if mean length above/below Ltarget

Usage

```
EtargetLopt(x, Data, reps = 100, yrsmth=3, buffer=0.1)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrsmth	Number of years to calculate average length
buffer	Parameter controlling the fraction of mean catch to set the reference (or target) TAC level - acts as a precautionary buffer

Value

An adjustment for fishing effort

Author(s)

HF Geromont

Example_datafile *Example_datafile Data*

Description

An object of class Data

Usage

```
Example_datafile
```

Format

An object of class Data of length 1.

`Fadapt` *An adaptive MP that uses trajectory in inferred suplus production and fishing mortality rate to update a TAC*

Description

Fishing rate is modified each year according to the gradient of surplus production with biomass (aims for zero). F is bounded by FMSY/2 and 2FMSY and walks in the logit space according to dSP/dB. This is derived from the theory of Maunder 2014.

Usage

```
Fadapt(x, Data, reps = 100, yrsmth = 7, gg=1)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples
`yrsmth`	Years over which to smooth recent estimates of surplus production
`gg`	A gain parameter controlling the speed in update in TAC.

Details

Tested in Carruthers et al. 2015.

Value

A numeric vector of quota recommendations

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press. Maunder. 2014. http://www.iattc.org/Meetings/Meetings2014/MAYSAC/PDFs/SAC-05-10b-Management-Strategy-Evaluation.pdf

Fdem — *Demographic FMSY method*

Description

FMSY is calculated as r/2 where r is calculated from a demographic approach (inc steepness). Coupled with an estimate of current abundance that gives you the OFL.

Usage

```
Fdem(x, Data, reps = 100)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples

Details

Made up for this package. This uses Murdoch McAllister's demographic r method to derive FMSY (r/2) and then makes the quota r*current biomass / 2. Easy.

Author(s)

T. Carruthers

References

McAllister, M.K., Pikitch, E.K., and Babcock, E.A. 2001. Using demographic methods to construct Bayesian priors for the intrinsic rate of increase in the Schaefer model and implications for stock rebuilding. Can. J. Fish. Aquat. Sci. 58: 1871-1890.

Fdem_CC — *Demographic FMSY method using catch-curve analysis to estimate recent Z*

Description

FMSY is calculated as r/2 from a demographic r prior method, current abudnance is estimated from naive catch curve analysis.

Usage

```
Fdem_CC(x, Data, reps = 100, Fmin=0.005)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
Fmin	The minimum fishing mortality rate derived from the catch-curve analysis

Author(s)

T. Carruthers

References

McAllister, M.K., Pikitch, E.K., and Babcock, E.A. 2001. Using demographic methods to construct Bayesian priors for the intrinsic rate of increase in the Schaefer model and implications for stock rebuilding. Can. J. Fish. Aquat. Sci. 58: 1871-1890.

Fdem_ML — *Demographic FMSY method that uses mean length data to estimate recent Z*

Description

Demographic F (r/2) method using the mean length estimator to calculate current abundance.

Usage

```
Fdem_ML(x, Data, reps = 100)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples

Note

The mean length extension was programmed by Gary Nelson as part of his excellent R package 'fishmethods'

Author(s)

T. Carruthers

References

McAllister, M.K., Pikitch, E.K., and Babcock, E.A. 2001. Using demographic methods to construct Bayesian priors for the intrinsic rate of increase in the Schaefer model and implications for stock rebuilding. Can. J. Fish. Aquat. Sci. 58: 1871-1890.

Fease *MP feasibility diagnostic*

Description

What MPs may be run (best case scenario) for various data-availability scenarios?

Usage

```
Fease(feaseobj,outy='table')
```

Arguments

feaseobj	An object of class 'Fease'
outy	Determines whether you would like a full table or some column of the table for a specific case of the feasibility object. When set equal to table, the full table is produced. When set equal to an integer number the names of MPs that are feasible for that case are returned.

Author(s)

T. Carruthers

Fease-class *Class* 'Fease'

Description

An object for storing information about what data are available or might be available

Slots

Name The name of the data feasibility object

Case The names of the data feasibility cases

Catch Total annual catches

Index An index of relative abundance, catch per unit effort data or of fishing mortality rate (effort)

Natural_mortality_rate From Maximum age, Tagging data, early fishery catch composition data

Maturity_at_length From gonadal analysis, growth and natural mortality rate estimates

Growth Paired length and age observations, maximum length and an estimate of natural mortality rate

Length_weight_conversion Paired weight and length observations, equivalent data from a similar species

`Fleet_selectivity` Length composition of catches with growth curve and natural mortality rate, estimates from a similar fleet type targetting a similar species

`Catch_at_length` Length composition of catches (length samples)

`Catch_at_age` Age composition of catches (age samples)

`Recruitment_index` Spawn survey, estimates from a stock assessment, VPA analysis of catch composition data

`Stock_recruitment_relationship` Stock assessment, a stock assessment of a similar species

`Target_catch` An agreed annual catch target, MSY proxy

`Target_biomass` An agreed absolute biomass target, mean historical biomass estimate

`Target_index` An agreed catch rate target

`Abundance` Fishery independent survey, current fishing mortality rate from recent length composition, natural mortality rate, maturity at age, growth and stock recruitment relationship, habitat and relative density extrapolation

Objects from the Class

Objects can be created by calls of the form `new('Fease', stock)`

Author(s)

T. Carruthers and A. Hordyk

Examples

```
newdata<-new('Fease')
```

`Fease_xl` *Read in feasibility parameters from Excel spreadsheet*

Description

A function to read in feasibility parameters from an Excel spreadsheet with tabs named following specific convention

Usage

```
Fease_xl(fname, stkname, fpath = '', saveCSV = FALSE)
```

Arguments

`fname`	Name of the Excel spreadsheet file. Must include file extension.
`stkname`	Name of the Stock.
`fpath`	Full file path, if file is not in current working directory
`saveCSV`	Do you also want to save the Stock, Fleet and Observation parameters to CSV files?

Details

The Excel spreadsheet must have tabs named with the following convention. For example if `stkname` is 'myFish', the tab must be named 'myFishFease,

Value

A object of class Fease

Author(s)

A. Hordyk

Examples

```
 ## Not run:
 myFease <- Fease_xl(fname='FeaseTables.xlsx', stkname='myFish')

## End(Not run)
```

`Feasibility` *Feasibility Fease*

Description

An object of class Fease

Usage

```
Feasibility
```

Format

An object of class `Fease` of length 1.

Feasibility2 *Feasibility2 Fease*

Description

An object of class Fease

Usage

```
Feasibility2
```

Format

An object of class `Fease` of length 1.

fetch.file.names *Reads iSCAM Data, Control and Projection files*

Description

A function for returning the three types of iSCAM input and output files

Usage

```
fetch.file.names(path, filename)
```

Arguments

`path`	File path
`filename`	The filename

Author(s)

Chris Grandin (DFO PBS)

FlatE_Dom *FlatE_Dom Fleet*

Description

An object of class Fleet

Usage

```
FlatE_Dom
```

Format

An object of class `Fleet` of length 1.

FlatE_HDom *FlatE_HDom Fleet*

Description

An object of class Fleet

Usage

```
FlatE_HDom
```

Format

An object of class `Fleet` of length 1.

FlatE_NDom *FlatE_NDom Fleet*

Description

An object of class Fleet

Usage

```
FlatE_NDom
```

Format

An object of class `Fleet` of length 1.

Fleet-class *Class* 'Fleet'

Description

The component of the operating model that controls fishing dynamics

Slots

Name Name of the Fleet object. Single value. Character string.

nyears The number of years for the historical 'spool-up' simulation. Single value. Positive integer

Spat_targ Distribution of fishing in relation to spatial biomass: fishing distribution is proportional to B^Spat_targ. Uniform distribution lower and upper bounds. Real numbers

EffYears Years representing join-points (vertices) of time-varying effort. Vector. Non-negative real numbers

EffLower Lower bound on relative effort corresponding to EffYears. Vector. Non-negative real numbers

EffUpper Upper bound on relative effort corresponding to EffYears. Vector. Non-negative real numbers

Esd Additional inter-annual variability in fishing mortality rate. Uniform distribution lower and upper bounds. Non-negative real numbers

qinc Average percentage change in fishing efficiency (applicable only to forward projection and input controls). Uniform distribution lower and upper bounds. Non-negative real numbers

qcv Inter-annual variability in fishing efficiency (applicable only to forward projection and input controls). Uniform distribution lower and upper bounds. Non-negative real numbers

L5 Shortest length corresponding to 5 percent vulnerability. Uniform distribution lower and upper bounds. Positive real numbers

LFS Shortest length that is fully vulnerable to fishing. Uniform distribution lower and upper bounds. Positive real numbers

Vmaxlen The vulnerability of fish at Stock@Linf. Uniform distribution lower and upper bounds. Fraction

isRel Selectivity parameters in units of size-of-maturity (or absolute eg cm). Single value. Boolean.

LR5 Shortest length corresponding ot 5 percent retention. Uniform distribution lower and upper bounds. Non-negative real numbers

LFR Shortest length that is fully retained. Uniform distribution lower and upper bounds. Non-negative real numbers

Rmaxlen The retention of fish at Stock@Linf. Uniform distribution lower and upper bounds. Non-negative real numbers

DR Discard rate - the fraction of caught fish that are discarded. Uniform distribution lower and upper bounds. Fraction

SelYears (Optional) Years representing join-points (vertices) at which historical selectivity pattern changes. Vector. Positive real numbers

AbsSelYears (Optional) Calendar years corresponding with SelYears (eg 1951, rather than 1), used for plotting only. Vector (of same length as SelYears). Positive real numbers

L5Lower (Optional) Lower bound of L5 (use ChooseSelect function to set these). Vector. Non-negative real numbers

L5Upper (Optional) Upper bound of L5 (use ChooseSelect function to set these). Vector. Non-negative real numbers

LFSLower (Optional) Lower bound of LFS (use ChooseSelect function to set these). Vector. Non-negative real numbers

LFSUpper (Optional) Upper bound of LFS (use ChooseSelect function to set these). Vector. Non-negative real numbers

VmaxLower (Optional) Lower bound of Vmaxlen (use ChooseSelect function to set these). Vector. Fraction

VmaxUpper (Optional) Upper bound of Vmaxlen (use ChooseSelect function to set these). Vector. Fraction

CurrentYr The current calendar year (final year) of the historical simulations (eg 2011). Single value. Positive integer.

Objects from the Class

Objects can be created by calls of the form new('Fleet')

Author(s)

T. Carruthers and A. Hordyk

Examples

```
showClass('Fleet')
```

FleetDescription *FleetDescription*

Description

A data.frame with description of slots for class Fleet

Usage

```
FleetDescription
```

Format

An object of class data.frame with 26 rows and 2 columns.

FMSYref — *A reference FMSY method (uses perfect information about FMSY)*

Description

FMSY is taken from the operating model stored at DLM@OM$FMSY

Usage

```
FMSYref(x, Data, reps = 100)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples

Details

Note that you can out-perform this MP even though it has perfect information of FMSY and current abundance. The requirment for fixed F is actually quite strict and is by no means the upper limit in terms of yield. Don't panic if your method beats this one for yield, especially for short-lived species of high temporal variability in productivity!

Author(s)

T. Carruthers

FMSYref50 — *A reference FMSY method that fishes at half of FMSY (uses perfect information about FMSY)*

Description

FMSY is taken from the operating model stored at DLM@OM$FMSY

Usage

```
FMSYref50(x, Data, reps = 100)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC (OFL) samples

Details

Note that you can out-performm this method easily. The requirement for fixed F is actually quite strict and is by no means the upper limit in terms of yield. Don't panic if your method beats this one for yield!

Interesting that the reduction in yield is no way near commensurate with the reduction in F - as predicted by a yield curve and expressed in the pretty good yield theory.

Author(s)

T. Carruthers

`FMSYref75` *A reference FMSY method that fishes at three quarters of FMSY (uses perfect information about FMSY)*

Description

FMSY is taken from the operating model stored at DLM@OM$FMSY

Usage

```
FMSYref75(x, Data, reps = 100)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples

Details

Note that you can out-performm this method easily. The requirement for fixed F is actually quite strict and is by no means the upper limit in terms of yield. Don't panic if your method beats this one for yield!

Interesting that the reduction in yield is no way near commensurate with the reduction in F as predicted by a yield curve and expressed in the pretty good yield theory.

Author(s)

T. Carruthers

ForceCor *Forces correlation among operating model parameters for M, K, Linf and L50*

Description

Uses typical correlations among estimated parameters to generate realistic samples for natural mortality rate (M), growth rate (K), maximum length (Linf) and length at 50

Usage

```
ForceCor(OM, nsim = 48, plot = T)
```

Arguments

OM	An operating model object with M, growth, stock-recruitment and maturity parameters specified.
nsim	The number of simulated values to create (note that OM@nsim will be used preferentially).
plot	Should the sampled parameters and distributions be plotted?

Value

An object of class OM with a populated (or appended) cpars slot

Author(s)

T. Carruthers (Canadian DFO grant)

Examples

```
testOM<-ForceCor(testOM)
```

Fratio *An FMSY/M ratio method*

Description

Calculates the OFL based on a fixed ratio of FMSY to M multiplied by a current estimate of abundance.

Usage

```
Fratio(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of samples of the TAC recommendation

Details

A simple method that tends to outperform many other approaches alarmingly often even when current biomass is relatively poorly known. The low stock crash potential is largely due to the quite large difference between Fmax and FMSY for most stocks.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Gulland, J.A., 1971. The fish resources of the ocean. Fishing News Books, West Byfleet, UK.

Martell, S., Froese, R., 2012. A simple method for estimating MSY from catch and resilience. Fish Fish. doi: 10.1111/j.1467-2979.2012.00485.x.

Fratio4010 *An FMSY/M ratio method paired with the 40-10 rule*

Description

Calculates the OFL based on a fixed ratio of FMSY to M multiplied by a current estimate of abundance. In this method DBSRA is paired with the 40-10 rule that throttles back the OFL to zero at 10 percent of unfished biomass.

Usage

```
Fratio4010(x, Data, reps = 100)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples

Author(s)

T. Carruthers

References

Gulland, J.A., 1971. The fish resources of the ocean. Fishing News Books, West Byfleet, UK.

Martell, S., Froese, R., 2012. A simple method for estimating MSY from catch and resilience. Fish Fish. doi: 10.1111/j.1467-2979.2012.00485.x.

`Fratio_CC` *A data-limited method that uses FMSY/M ratio and a naive catch-curve estimate of recent Z*

Description

Calculates the OFL based on a fixed ratio of FMSY to M and a catch curve estimate of current stock size.

Usage

```
Fratio_CC(x, Data, reps = 100, Fmin = 0.005)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples
`Fmin`	Minimum current fishing mortality rate for the catch-curve analysis

Author(s)

T. Carruthers

References

Gulland, J.A., 1971. The fish resources of the ocean. Fishing News Books, West Byfleet, UK.

Martell, S., Froese, R., 2012. A simple method for estimating MSY from catch and resilience. Fish Fish. doi: 10.1111/j.1467-2979.2012.00485.x.

Fratio_ML *An FMSY/M ratio MP that uses a mean length estimator of recent Z*

Description

Calculates the OFL based on a fixed ratio of FMSY/M and an estimate of current stock size from a mean-length estimator.

Usage

```
Fratio_ML(x, Data, reps = 100)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples

Note

The mean length extension was programmed by Gary Nelson as part of his excellent R package 'fishmethods'

Author(s)

T. Carruthers

References

Gulland, J.A., 1971. The fish resources of the ocean. Fishing News Books, West Byfleet, UK.

Martell, S., Froese, R., 2012. A simple method for estimating MSY from catch and resilience. Fish Fish. doi: 10.1111/j.1467-2979.2012.00485.x.

GB_CC *Geromont and Butterworth Constant Catch Harvest Control Rule*

Description

A simple MP that aims for average historical catches (as a proxy for MSY) subject to imperfect information.

Usage

```
GB_CC(x, Data, reps = 100)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples

Details

Note that this is my interpretation of their MP and is now stochastic. Currently it is generalized and is not 'tuned' to more detailed assessment data which might explain why in some cases it leads to stock declines.

Author(s)

T. Carruthers

References

Geromont, H.F. and Butterworth, D.S. 2014. Complex assessment or simple management procedures for efficient fisheries management: a comparative study. ICES J. Mar. Sci. doi:10.1093/icesjms/fsu017

GB_slope *Geromont and Butterworth index slope Harvest Control Rule*

Description

An MP similar to SBT1 that modifies a time-series of catch recommendations and aims for a stable catch rates.

Usage

```
GB_slope(x, Data, reps = 100, yrsmth = 5, lambda = 1)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrsmth	Number of years for evaluating slope in relative abundance index
lambda	A gain parameter

Details

Note that this is my interpretation of their approach and is now stochastic. Currently it is generalized and is not 'tuned' to more detailed assessment data which might explain why in some cases it leads to stock declines.

Author(s)

T. Carruthers

References

Geromont, H.F. and Butterworth, D.S. 2014. Complex assessment or simple management procedures for efficient fisheries management: a comparative study. ICES J. Mar. Sci. doi:10.1093/icesjms/fsu017

GB_target *Geromont and Butterworth target CPUE and catch MP*

Description

An MP similar to SBT2 that modifies a time-series of catch recommendations and aims for target catch rate and catch level based on BMSY/B0 and MSY, respectively.

Usage

```
GB_target(x, Data, reps = 100, w = 0.5)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of quota samples
w	A gain parameter

Details

Note that this is my interpretation of their MP and is now stochastic. Currently it is generalized and is not 'tuned' to more detailed assessment data which might explain why in some cases it leads to stock declines.

Author(s)

T. Carruthers

References

Geromont, H.F. and Butterworth, D.S. 2014. Complex assessment or simple management procedures for efficient fisheries management: a comparative study. ICES J. Mar. Sci. doi:10.1093/icesjms/fsu017

Gcontrol *G-control MP*

Description

A harvest control rule proposed by Carl Walters that uses trajectory in inferred surplus production to make upward/downward adjustments to TAC recommendations

Usage

```
Gcontrol(x, Data, reps = 100, yrsmth = 10, gg = 2, glim = c(0.5,
2))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of quota samples
yrsmth	The number of years over which to smooth catch and biomass data
gg	A gain parameter
glim	A constraint limiting the maximum level of change in quota recommendations

Author(s)

C. Walters and T. Carruthers

References

Made-up for this package. Carruthers et al. 2015. Performance of Simple Management Procedures.

Generic_DecE *Generic_DecE Fleet*

Description

An object of class Fleet

Usage

```
Generic_DecE
```

Format

An object of class `Fleet` of length 1.

Generic_FlatE *Generic_FlatE Fleet*

Description

An object of class Fleet

Usage

```
Generic_FlatE
```

Format

An object of class `Fleet` of length 1.

Generic_Fleet *Generic_Fleet Fleet*

Description

An object of class Fleet

Usage

```
Generic_Fleet
```

Format

An object of class `Fleet` of length 1.

Generic_IncE *Generic_IncE Fleet*

Description

An object of class Fleet

Usage

```
Generic_IncE
```

Format

An object of class `Fleet` of length 1.

Generic_Obs *Generic_Obs Obs*

Description

An object of class Obs

Usage

```
Generic_Obs
```

Format

An object of class `Obs` of length 1.

genLenComp *Generate length composition of catch*

Description

Generate size composition of catch given sample of catch-at-age, expected length-at-age, and standard deviation of length-at-age. Model assumes length-at-age is normally distributed, and that selectivity is size-dependant

Usage

```
genLenComp(CAL_bins, CAL_binsmid, SL, CAL_ESS, CAL_nsamp, CN, LaA, LaASD,
  truncSD)
```

Arguments

`CAL_bins`	vector of catch-at-length size bins
`CAL_binsmid`	vector (nbins = length(CAL_bins) - 1) of mid-points for catch-at-length size bins
`SL`	matrix (nbins, nyears) of selectivity-at-length class for each year
`CAL_ESS`	effective sample size of catch-at-length data
`CAL_nsamp`	sample size of catch-at-length data
`CN`	matrix (nyears, maxage) of catch-at-age for each year
`LaA`	matrix (maxage, nyears) of expected length-at-age for each year
`LaASD`	matrix (maxage, nyears) of standard deviation of length-at-age for each year
`truncSD`	optional argument to truncate the length-at-age distribution at 'truncSD' standard deviations e.g., a value of 2 truncates the length-at-age distribution at two standard deviations (set to 0 to ignore (default))

getAFC *Calculate age at first capture from length at first capture and growth*

Description

As title.

Usage

```
getAFC(t0c,Linfc,Kc,LFC,maxage)
```

Arguments

t0c	A vector of theoretical age at length zero (von Bertalanffy growth)
Linfc	A vector of maximum length (von Bertalanffy growth)
Kc	A vector of maximum growth rate (von Bertalanffy growth)
LFC	A vector of length at first capture
maxage	Maximum age

Author(s)

T. Carruthers

getBH *Predict Beverton-Holt recruitment and return fit to S-R observations*

Description

Internal function to optBH

Usage

```
getBH(pars, SSB, rec, SSBpR, mode = 1, plot = F)
```

Arguments

pars	an initial guess at model parameters steepness and R0
SSB	'observations' of spawning biomass
rec	'observations' (model predictions) of recruitment
SSBpR	spawning stock biomass per recruit at unfished conditions
mode	should fit or recruitment deviations be returned
plot	should a plot of the model fit be produced?

Author(s)

T. Carruthers

getBlow *Blow parallel optimization function*

Description

Find the current biomass at which it would take HZN mean generation times to reach Bfrac x SSBMSY biomass level given zero catches

Usage

```
getBlow(x, SSBMSY, MGThorizon, Find, Perr, M_ageArray, hs, Mat_age, Wt_age, R0,
  V, nyears, maxage, mov, Spat_targ, SRrel, aR, bR, Bfrac = 0.5, ploty = F)
```

Arguments

x	position in a vector
SSBMSY	vector nsim long of spawning biomass at MSY
MGThorizon	vector nsim long of MGT x HZN
Find	matrix of fishing mortality rate nsim x nyears
Perr	matrix of recruitment devitions nsim x nyears + maxage -1
M_ageArray	array of natural mortality rate nsim x maxage x nyears + proyears
hs	vector nsim long of steepness values
Mat_age	array nsim x nages x nyears+proyears of maturity at age
Wt_age	matrix nsim x nages of weight at age
R0	vector nsim long of unfished recruitment
V	array of vulnerability nsim x maxage x nyears
nyears	integer: number of historical years
maxage	integer: maximum age
mov	array of movement nsim x 2 x 2
Spat_targ	vector of spatial targetting parameters
SRrel	integer representing recruitmetn dynamics type 1: Bev Holt 2: Ricker
aR	vector of recruitment parameters
bR	vector of recruitment parameters
Bfrac	fraction of SSBMSY that is the target
ploty	logical: should a plot be produced

Author(s)

T. Carruthers

getclass *get object class*

Description

Internal function for determining if object is of classy

Usage

```
getclass(x, classy)
```

Arguments

x	Character string object name
classy	A class of object (character string, e.g. 'Fleet')

Value

TRUE or FALSE

Author(s)

T. Carruthers

getFMSY3 *Calculate FMSY and related metrics using Rcpp code*

Description

Calculate FMSY and related metrics using Rcpp code

Usage

```
getFMSY3(x, Asize, nareas, maxage, N, pyears, M_ageArray, Mat_age, Wt_age, V,
  retA, Perr, mov, SRrel, Find, Spat_targ, hs, R0a, SSBpR, aR, bR, SSB0, B0,
  maxF, useCPP = TRUE)
```

Arguments

x	Integer, the simulation number
Asize	A matrix (nsim by nareas) with size of areas
nareas	The number of spatial areas
maxage	The maximum age

N	Array of the numbers-at-age in population. Dimensions are nsim, maxage, nyears, nareas. Only values from the first year (i.e N[,,1,]) are used, which is the current N-at-age.
pyears	The number of years to project forward. Equal to 'nyears' for optimizing for q.
M_ageArray	An array (dimensions nsim, maxage, nyears+proyears) with the natural mortality-at-age and year
Mat_age	A matrix (dimensions nsim, maxage) with the proportion mature for each age-class
Wt_age	An array (dimensions nsim, maxage, nyears+proyears) with the weight-at-age and year
V	An array (dimensions nsim, maxage, nyears+proyears) with the vulnerability-at-age and year
retA	An array (dimensions nsim, maxage, nyears+proyears) with the probability retained-at-age and year
Perr	A matrix (dimensions nsim, nyears+proyears) with the recruitment deviations
mov	An array (dimensions nsim, nareas, nareas) with the movement matrix
SRrel	A numeric vector nsim long specifying the recruitment curve to use
Find	A matrix (dimensions nsim, nyears) with the historical fishing effort
Spat_targ	A numeric vector nsim long with the spatial targeting
hs	A numeric vector nsim long with the steepness values for each simulation
R0a	A matrix (dimensions nsim, nareas) with the unfished recruitment by area
SSBpR	A matrix (dimensions nsim, nareas) with the unfished spawning-per-recruit by area
aR	A numeric vector nsim long with the Ricker SRR a values
bR	A numeric vector nsim long with the Ricker SRR b values
SSB0	Unfished spawning biomass
B0	Unfished total biomass
maxF	A numeric value specifying the maximum fishing mortality for any single age class
useCPP	logical - use the CPP code? For testing purposes only

Author(s)

A. Hordyk

getFref3 *Calculate Reference Yield*

Description

Calculate Reference Yield

Usage

```
getFref3(x, Asize, nareas, maxage, N, pyears, M_ageArray, Mat_age, Wt_age, V,
  retA, Perr, mov, SRrel, Find, Spat_targ, hs, R0a, SSBpR, aR, bR, maxF,
  useCPP = TRUE)
```

Arguments

x	Integer, the simulation number
Asize	A matrix (dimensions nsim by nareas) with relative size of areas
nareas	The number of spatial areas
maxage	The maximum age
N	Array of the numbers-at-age in population. Dimensions are nsim, maxage, nyears, nareas. Only values from the first year (i.e N[,,1,]) are used, which is the current N-at-age.
pyears	The number of years to project forward. Equal to 'nyears' for optimizing for q.
M_ageArray	An array (dimensions nsim, maxage, nyears+proyears) with the natural mortality-at-age and year
Mat_age	An array (dimensions nsim, maxage, nyears+proyears) with the proportion mature for each age-class
Wt_age	An array (dimensions nsim, maxage, nyears+proyears) with the weight-at-age and year
V	An array (dimensions nsim, maxage, nyears+proyears) with the vulnerability-at-age and year
retA	An array (dimensions nsim, maxage, nyears+proyears) with the probability retained-at-age and year
Perr	A matrix (dimensions nsim, nyears+proyears) with the recruitment deviations
mov	An array (dimensions nsim, nareas, nareas) with the movement matrix
SRrel	A numeric vector nsim long specifying the recruitment curve to use
Find	A matrix (dimensions nsim, nyears) with the historical fishing effort
Spat_targ	A numeric vector nsim long with the spatial targeting
hs	A numeric vector nsim long with the steepness values for each simulation
R0a	A matrix (dimensions nsim, nareas) with the unfished recruitment by area
SSBpR	A matrix (dimensions nsim, nareas) with the unfished spawning-per-recruit by area

aR	A numeric vector nareas long with the Ricker SRR a values
bR	A numeric vector nareas long with the Ricker SRR b values
maxF	A numeric value specifying the maximum fishing mortality for any single age class
useCPP	logical - use the CPP code? For testing purposes only

Author(s)

A. Hordyk

A. Hordyk

`getGpars` *Extracts growth parameters from a SS3 r4ss replist*

Description

Extracts growth parameters from a SS3 r4ss replist

Usage

```
getGpars(replist, seas = 1)
```

Arguments

replist	the list output of the r4ss SS_output function (a list of assessment inputs / outputs)
seas	The reference season for the growth (not actually sure what this does yet)

Author(s)

T. Carruthers

`getmov2` *Optimization function to find a movement model that matches user specified movement characteristics modified for Rcpp.*

Description

The user specifies the probability of staying in the same area and spatial heterogeneity (both in the unfished state).

Usage

```
getmov2(x, Prob_staying, Frac_area_1)
```

Arguments

x	A position in vectors Prob_staying and Frac_area_1
Prob_staying	User specified probability that individuals in area 1 remain in that area (unfished conditions)
Frac_area_1	User specified fraction of individuals found in area 1 (unfished conditions)

Details

This is paired with movfit to find the correct movement model.

Value

A markov movement matrix

Author(s)

T. Carruthers

Examples

```
Prob_staying<-0.8 # probability  that individuals remain in area 1 between time-steps
Frac_area_1<-0.35 # the fraction of the stock found in area 1 under equilibrium conditions
markovmat<-getmov2(1,Prob_staying, Frac_area_1)
vec<-c(0.5,0.5) # initial guess at equilibrium distribution (2 areas)
for(i in 1:300)vec<-apply(vec*markovmat,2,sum) # numerical approximation to stable distribution
c(markovmat[1,1],vec[1]) # pretty close right?
```

---

getq3 *optimize for catchability (q)*

Description

Function optimizes catchability (q, where F=qE) required to get to user-specified stock depletion

Usage

```
getq3(x, dep, SSB0, nareas, maxage, N, pyears, M_ageArray, Mat_age, Asize,
  Wt_age, V, retA, Perr, mov, SRrel, Find, Spat_targ, hs, R0a, SSBpR, aR, bR,
  bounds = c(1e-05, 15), maxF, useCPP = TRUE)
```

Arguments

x	Integer, the simulation number
dep	A numeric vector nsim long of sampled depletion
SSB0	A numeric vector nsim long of total unfished spawning biomass
nareas	The number of spatial areas
maxage	The maximum age
N	Array of the numbers-at-age in population. Dimensions are nsim, maxage, nyears, nareas. Only values from the first year (i.e N[,,1,]) are used, which is the current N-at-age.
pyears	The number of years to project forward. Equal to 'nyears' for optimizing for q.
M_ageArray	An array (dimensions nsim, maxage, nyears+proyears) with the natural mortality-at-age and year
Mat_age	An array (dimensions nsim, maxage, proyears+nyears) with the proportion mature for each age-class
Asize	A matrix (dimensions nsim, nareas) with size of each area
Wt_age	An array (dimensions nsim, maxage, nyears+proyears) with the weight-at-age and year
V	An array (dimensions nsim, maxage, nyears+proyears) with the vulnerability-at-age and year
retA	An array (dimensions nsim, maxage, nyears+proyears) with the probability retained-at-age and year
Perr	A matrix (dimensions nsim, nyears+proyears) with the recruitment deviations
mov	An array (dimensions nsim, nareas, nareas) with the movement matrix
SRrel	A numeric vector nsim long specifying the recruitment curve to use
Find	A matrix (dimensions nsim, nyears) with the historical fishing effort
Spat_targ	A numeric vector nsim long with the spatial targeting
hs	A numeric vector nsim long with the steepness values for each simulation
R0a	A matrix (dimensions nsim, nareas) with the unfished recruitment by area
SSBpR	A matrix (dimensions nsim, nareas) with the unfished spawning-per-recruit by area
aR	A numeric vector nareas long with the Ricker SRR a values
bR	A numeric vector nareas long with the Ricker SRR b values
bounds	A numeric vector of length 2 with bounds for the optimizer
maxF	A numeric value specifying the maximum fishing mortality for any single age class
useCPP	logical - use the CPP code? For testing purposes only

Author(s)

A. Hordyk

getsel *Calculate selectivity curve*

Description

Calculate selectivity curve

Usage

```
getsel(x, lens, lfs, sls, srs)
```

Arguments

x	Simulation number
lens	Matrix of lengths (nsim by nlengths)
lfs	Vector of length at full selection (nsim long)
sls	Vector of sigmas of ascending limb (nsim long)
srs	Vector of sigmas of descending limb (nsim long)

Gulf_blue_tilefish *Gulf_blue_tilefish Data*

Description

An object of class Data

Usage

```
Gulf_blue_tilefish
```

Format

An object of class Data of length 1.

HDAAC *Hybrid Depletion Adjusted Average Catch*

Description

Essentially DCAC multiplied by 2*depletion and divided by BMSY/B0 (Bpeak) when below BMSY, and DCAC above BMSY

Usage

```
HDAAC(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the TAC recommendation

Author(s)

W. Harford and T. Carruthers

References

MacCall, A.D., 2009. Depletion-corrected average catch: a simple formula for estimating sustainable yields in data-poor situations. ICES J. Mar. Sci. 66, 2267-2271. Harford W. and Carruthers, T. 2016. Testing novel catch-based fisheries management procedures.

Herring *Herring Stock*

Description

An object of class Stock

Usage

```
Herring
```

Format

An object of class Stock of length 1.

hist2 *Wrapper for histogram function*

Description

Produces a blank plot if all values in x are equal

Usage

```
hist2(x, col, axes = FALSE, main = "", breaks = 10, cex.main = 1)
```

Arguments

x	A vector of values
col	Colour of the histogram
axes	Logical - should axes be included?
main	Character - main title
breaks	Number of breaks. See ?hist for more details
cex.main	Text size of the main title

ICI *Index Confidence Interval (ICI) MP by Jardim et al. (2015)*

Description

The MP adjusts catch based on the value of the index in the current year relative to the time series mean and standard error.

The mean and standard error of the index time series is calculated. There are two thresholds which delineates whether catch is reduced, held constant, or increased. The catch is reduced by 0.75 if the Z-score of the current year's index is less than -0.44. The catch is increased by 1.05 if the Z-score of the current year's index is greater than 1.96. Otherwise, the catch is held constant.

Usage

```
ICI(x, Data, reps)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples

Author(s)

Coded by Q. Huynh. Developed by Jardim et al. (2015)

References

Ernesto Jardim, Manuela Azevedo, Nuno M. Brites, Harvest control rules for data limited stocks using length-based reference points and survey biomass indices, Fisheries Research, Volume 171, November 2015, Pages 12-19, ISSN 0165-7836, https://doi.org/10.1016/j.fishres.2014.11.013

ICI2 *Less Precautionary Index Confidence Interval (ICI) MP by Jardim et al. (2015)*

Description

The MP adjusts catch based on the value of the index in the current year relative to the time series mean and standard error. This method is less precautionary of the two ICI MPs by allowing for a larger increase in TAC and a lower threshold of the index to decrease the TAC (see Jardim et al. 2015).

Usage

```
ICI2(x, Data, reps)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples

Details

The mean and standard error of the index time series is calculated. There are two thresholds which delineates whether catch is reduced, held constant, or increased. The catch is reduced by 0.75 if the Z-score of the current year's index is less than -1.96. The catch is increased by 1.25 if the Z-score of the current year's index is greater than 1.96. Otherwise, the catch is held constant.

Author(s)

Coded by Q. Huynh. Developed by Jardim et al. (2015)

References

Ernesto Jardim, Manuela Azevedo, Nuno M. Brites, Harvest control rules for data limited stocks using length-based reference points and survey biomass indices, Fisheries Research, Volume 171, November 2015, Pages 12-19, ISSN 0165-7836, https://doi.org/10.1016/j.fishres.2014.11.013

Imp-class *Class* 'Imp'

Description

An operating model component that specifies the degree of adherence to management recommendations (Implementation error)

Slots

`Name` The name of the Implementation error object. Single value. Character string.

`TACFrac` Mean fraction of TAC taken. Uniform distribution lower and upper bounds. Positive real number.

`TACSD` Log-normal coefficient of variation in the fraction of Total Allowable Catch (TAC) taken. Uniform distribution lower and upper bounds. Non-negative real numbers.

`TAEFrac` Mean fraction of TAE taken. Uniform distribution lower and upper bounds. Positive real number.

`TAESD` Log-normal coefficient of variation in the fraction of Total Allowable Effort (TAE) taken. Uniform distribution lower and upper bounds. Non-negative real numbers.

`SizeLimFrac` The real minimum size that is retained expressed as a fraction of the size. Uniform distribution lower and upper bounds. Positive real number.

`SizeLimSD` Log-normal coefficient of variation controlling mismatch between a minimum size limit and the real minimum size retained. Uniform distribution lower and upper bounds. Non-negative real numbers.

`Source` A reference to a website or article form which parameters were taken to define the object. Single value. Character string.

Objects from the Class

Objects can be created by calls of the form `new('Imp')`

Author(s)

T. Carruthers and A. Hordyk

Examples

```
showClass('Imp')
```

ImpDescription *ImpDescription*

Description

A data.frame with description of slots for class Imp

Usage

```
ImpDescription
```

Format

An object of class `data.frame` with 8 rows and 2 columns.

Imprecise_Biased *Imprecise_Biased Obs*

Description

An object of class Obs

Usage

```
Imprecise_Biased
```

Format

An object of class `Obs` of length 1.

Imprecise_Unbiased *Imprecise_Unbiased Obs*

Description

An object of class Obs

Usage

```
Imprecise_Unbiased
```

Format

An object of class `Obs` of length 1.

IncE_HDom *IncE_HDom Fleet*

Description

An object of class Fleet

Usage

```
IncE_HDom
```

Format

An object of class `Fleet` of length 1.

IncE_NDom *IncE_NDom Fleet*

Description

An object of class Fleet

Usage

```
IncE_NDom
```

Format

An object of class `Fleet` of length 1.

initialize-methods *~~ Methods for Function* `initialize` *~~*

Description

~~ Methods for function `initialize` ~~

Methods

list('signature(.Object = \'DLM\')') here~~

list('signature(.Object = \'Fleet\')') here~~

list('signature(.Object = \'MSE\')') here~~

list('signature(.Object = \'Obs\')') method here~~

list('signature(.Object = \'OM\')')

list('signature(.Object = \'Stock\')') here~~

list('signature(.Object = \'Fease\')') here~~

list('signature(.Object = \'DLM_general\')') this method here~~

`Input` *Function to run a set of input control methods*

Description

Runs a set of input control methods are returns the output in a single table

Usage

```
Input(Data, MPs = NA, reps = 100, timelimit = 10, CheckMPs = TRUE,
  msg = TRUE)
```

Arguments

`Data`	A Data object
`MPs`	A list of input MPs, if NA all available input MPs are run
`reps`	Number of repetitions (for those methods that use them)
`timelimit`	Maximum timelimit to run MP (in seconds)
`CheckMPs`	Logical, the Can function is run if this is TRUE
`msg`	Logical. Should messages be printed?

Author(s)

A. Hordyk

InputRec-class *Class* 'InputRec'

Description

An object for storing the recommendation for an input control MP

Slots

`Effort` A numeric value with the effort recommendation as a fraction of current (nyear) fishing effort

`Spatial` A boolean vector of length 'nareas' specifying if area is open (1) or closed (0) to fishing

`Allocate` A boolean value describing if effort should be re-allocated from close to open areas

`LR5` smallest length at 5 per cent retention

`LFR` smallest length at full retention

`HS` upper harvest slot (no retention above this)

`Rmaxlen` retention of the largest size class

`Misc` An empty list that can be used to store information and pass on to MPs in future

Objects from the Class

Objects can be created by calls of the form `new('InputRec')`

Author(s)

A. Hordyk

IOTC_plot *Indian Ocean Tuna Commission trade-off plot*

Description

A one-panel trade-off plot showing the probability of exceeding a biomass reference level and a yield reference level

Usage

```
IOTC_plot(MSEobj, Bref = 0.75, Yref = 0.75, Bsat = 0.8, Ysat = 0.8,
  xlim = c(0, 1.1), ylim = c(0, 1.1))
```

Arguments

MSEobj	An object of class MSE created by the function runMSE()
Bref	A biomass reference level (an improper fraction of BMSY)
Yref	A yield reference level (an improper fraction of yield given FMSY management)
Bsat	The satisficing level for biomass (required fraction of simulations exceeding Bref)
Ysat	The satisficing level for yield (required fraction of simulations exceeding Yref)
xlim	The limits of the x axis plotting
ylim	The limits of the y axis plotting

Author(s)

T. Carruthers

Iratio *Mean index ratio MP from Jardim et al. 2015*

Description

The TAC is adjusted by the ratio alpha, where the numerator being the mean index in the most recent two years of the time series and the denominator being the mean index in the three years prior to those in the numerator.

Usage

```
Iratio(x, Data, reps, yrs = c(2, 5))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrs	Vector of length 2 specifying the reference years

Details

This MP is the stochastic version of Method 3.2 used by ICES for Data-Limited Stocks (ICES 2012).

Author(s)

Coded by Q. Huynh. Developed by Jardim et al. (2015)

References

Ernesto Jardim, Manuela Azevedo, Nuno M. Brites, Harvest control rules for data limited stocks using length-based reference points and survey biomass indices, Fisheries Research, Volume 171, November 2015, Pages 12-19, ISSN 0165-7836, https://doi.org/10.1016/j.fishres.2014.11.013

ICES. 2012. ICES Implementation of Advice for Data-limited Stocks in 2012 in its 2012 Advice. ICES CM 2012/ACOM 68. 42 pp.

`iSCAM2Data` *Reads data from iSCAM file structure into a DLMtool Data object*

Description

A function that uses the file location of a fitted iSCAM model including input files to population the various slots of an data object. iSCAM2DLM relies on several functions written by Chris Grandin (DFO PBS).

Usage

```
iSCAM2Data(iSCAMdir, Name = NULL, Source = "No source provided",
  length_timestep = 1, Author = "No author provided")
```

Arguments

`iSCAMdir`	A folder with iSCAM input and output files in it
`Name`	The name of the operating model
`Source`	Reference to assessment documentation e.g. a url
`length_timestep`	How long is a model time step in years (e.g. a quarterly model is 0.25, a monthly model 1/12)
`Author`	Who did the assessment

Author(s)

T. Carruthers

iSCAM2DLM *Reads MLE estimates from iSCAM file structure into an operating model*

Description

A function that uses the file location of a fitted iSCAM model including input files to population the various slots of an operating model parameter estimates. iSCAM2DLM relies on several functions written by Chris Grandin (DFO PBS).

Usage

```
iSCAM2DLM(iSCAMdir, nsim = 48, proyears = 50, mcmc = F, Name = NULL,
  Source = "No source provided", length_timestep = 1,
  Author = "No author provided")
```

Arguments

iSCAMdir	A folder with iSCAM input and output files in it
nsim	The number of simulations to take for parameters with uncertainty (for OM@cpars custom parameters)
proyears	The number of MSE projection years
mcmc	Whether to use mcmc samples to create custom parameters cpars
Name	The name of the operating model
Source	Reference to assessment documentation e.g. a url
length_timestep	How long is a model time step in years (e.g. a quarterly model is 0.25, a monthly model 1/12)
Author	Who did the assessment

Author(s)

T. Carruthers

iSCAMcomps *Combines all iSCAM age composition data across fleets*

Description

iSCAM assessments are often fitted to numerous fleets that have differing age selectivities. iSCAMcomps is a simple way of providing the aggregate catch at age data. It should be noted that this process is important and in a real application would require due diligence (ie peer reviewed data workshop).

Usage

```
iSCAMcomps(replist, Year)
```

Arguments

replist	S3 class object: the output from a read from an iSCAM data folder
Year	Integer vector: the years of the DLMtool data object ie Data@Year

Author(s)

T. Carruthers

---

iSCAMinds *Combines indices into a single index using linear modelling*

Description

iSCAM assessments often make use of multiple indices of abundance. The DLMtool data object and MPs currently only make use of a single index. combiSCAMinds is a function that creates a single index from many using linear modelling. It is a simple way of providing initial calculations of management recommendations and it should be noted that this process is important and in a real application would require due diligence (ie peer reviewed data workshop).

Usage

```
iSCAMinds(idata, Year, fleeteffect = T)
```

Arguments

idata	List: the indices recorded in a read from an iSCAM data folder, e.g. replist$data$indices
Year	Integer vector: the years of the DLMtool data object ie Data@Year
fleeteffect	Logical: should a fleet effect be added to the linear model?

Author(s)

T. Carruthers

`Islope1` *A management procedure that incrementally adjusts the TAC to maintain a constant CPUE or relative abundance index*

Description

The least biologically precautionary of two constant index / CPUE methods proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
Islope1(x, Data, reps = 100, yrsmth = 5, lambda=0.4,xx=0.2)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples
`yrsmth`	Years over which to smooth recent estimates of surplus production
`lambda`	A gain parameter controlling the speed in update in TAC.
`xx`	Parameter controlling the fraction of mean catch to start using in first year

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of quota recommendations

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance review of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

`Islope4` *A management procedure that incrementally adjusts the TAC to maintain a constant CPUE or relative abundance index*

Description

The most biologically precautionary of two constant index / CPUE methods proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
Islope4(x, Data, reps = 100, yrsmth = 5, lambda=0.2,xx=0.4)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples
`yrsmth`	Years over which to smooth recent estimates of surplus production
`lambda`	A gain parameter controlling the speed in update in TAC.
`xx`	Parameter controlling the fraction of mean catch to start using in first year

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of quota recommendations

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

IT10 *Index Target 10*

Description

An index target MP where the TAC is modified according to current index levels (mean index over last 5 years) relative to a target level. Maximum annual changes are 10 per cent.

Usage

```
IT10(x, Data, reps = 100,yrsmth=5,mc=0.1)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the quota recommendation
yrsmth	The number of historical years over which to average the index
mc	The maximum fractional change in the TAC among years.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

IT5 *Index Target 5*

Description

An index target MP where the TAC is modified according to current index levels (mean index over last 5 years) relative to a target level. Maximum annual changes are 5 per cent.

Usage

```
IT5(x, Data, reps = 100,yrsmth=5,mc=0.05)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the quota recommendation
yrsmth	The number of historical years over which to average the index
mc	The maximum fractional change in the TAC among years.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

Itarget1 *A management procedure that incrementally adjusts the TAC (starting from reference level that is a fraction of mean recent catches) to reach a target CPUE / relative abundance index*

Description

The least biologically precautionary of two index/CPUE target MPs proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
Itarget1(x, Data, reps = 100, yrsmth = 5, xx=0, Imulti=1.5)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrsmth	Years over which to smooth recent estimates of surplus production
xx	Parameter controlling the fraction of mean catch to start using in first year
Imulti	Parameter controlling how much larger target CPUE / index is compared with recent levels.

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

`Itarget4` *A management procedure that incrementally adjusts the TAC (starting from reference level that is a fraction of mean recent catches) to reach a target CPUE / relative abundance index*

Description

The most biologically precautionary of two index/CPUE target MPs proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
Itarget4(x, Data, reps = 100, yrsmth = 5, xx=0.3, Imulti=2.5)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples
`yrsmth`	Years over which to smooth recent estimates of surplus production
`xx`	Parameter controlling the fraction of mean catch to start using in first year
`Imulti`	Parameter controlling how much larger target CPUE / index is compared with recent levels.

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

`ItargetE1` *A management procedure that incrementally adjusts the effort to reach a target CPUE / relative abundance index*

Description

An effort-based version of the least biologically precautionary of two index/CPUE target MPs proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
ItargetE1(x, Data, reps = 100, yrsmth = 5, xx = 0, Imulti = 1.5)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of samples
`yrsmth`	Years over which to smooth recent estimates of surplus production
`xx`	Parameter controlling the fraction of mean catch to start using in first year
`Imulti`	Parameter controlling how much larger target CPUE / index is compared with recent levels.

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of input controls

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

`ItargetE4` *A management procedure that incrementally adjusts the Effort to reach a target CPUE / relative abundance index*

Description

An effort-based version of the most biologically precautionary of two index/CPUE target MPs proposed by Geromont and Butterworth 2014.

Usage

```
ItargetE4(x, Data, reps = 100, yrsmth = 5, xx = 0, Imulti = 2.5)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of samples
`yrsmth`	Years over which to smooth recent estimates of surplus production
`xx`	Parameter controlling the fraction of mean catch to start using in first year
`Imulti`	Parameter controlling how much larger target CPUE / index is compared with recent levels.

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of input controls

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

ITe10 *Index Target Effort-Based 10*

Description

An index target MP where the Effort is modified according to current index levels (mean index over last 5 years) relative to a target level. Maximum annual changes are 10 per cent.

Usage

```
ITe10(x, Data, reps = 100, yrsmth = 5, mc = 0.1)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the quota recommendation
yrsmth	The number of historical years over which to average the index
mc	The maximum fractional change in the Effort among years.

Value

A numeric vector of input controls

Author(s)

T. Carruthers

ITe5 *Index Target Effort-Based 5*

Description

An index target MP where the Effort is modified according to current index levels (mean index over last 5 years) relative to a target level. Maximum annual changes are 5 per cent.

Usage

```
ITe5(x, Data, reps = 100, yrsmth = 5, mc = 0.05)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the quota recommendation
yrsmth	The number of historical years over which to average the index
mc	The maximum fractional change in the effort among years.

Value

A numeric vector of input controls

Author(s)

T. Carruthers

ITM *Index Target based on natural mortality rate*

Description

An index target MP where the TAC is modified according to current index levels (mean index over last yrsmth years) relative to a target level. Maximum fractional annual changes are mc. mc=(5+M*25)/100 yrsmth=4*(1/M)^(0.25)

Usage

```
ITM(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the quota recommendation

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

joinMSE *Join multiple MSE objects together*

Description

Joins two or more MSE objects together. MSE objects must have identical number of historical years, and projection years.

Usage

```
joinMSE(MSEobjs = NULL)
```

Arguments

MSEobjs	A list of MSE objects. Must all have identical operating model and MPs. MPs which don't appear in all MSE objects will be dropped.

Value

An object of class MSE

Author(s)

A. Hordyk

Jplot *Joint probability plot*

Description

Calculates and plots the joint probability of meeting all performance metrics simultaneously

Usage

```
Jplot(MSEobj, PLim = 0.8, YVar = c('LTY', 'STY', 'avgSSB_SSB0',
'avgB_BMSY'), PMRefs = list(B_BMSY = 0.5, SSB_SSB0 = 0.2), UseMean = TRUE,
lastYrs = 10, AvailMPs = NULL, XLim = NULL, ShowCols = TRUE, ShowLabs =
FALSE, All = TRUE)
```

Arguments

MSEobj	An object of class MSE
PLim	Probability limit (acceptable risk threshold; e.g., 0.8 for 80 percent)
YVar	What to plot of the y-axis: choose from c('LTY', 'STY','avgSSB_SSB0', 'avgB_BMSY')
PMRefs	List containing the reference limits for each metric
UseMean	Logical. Calculate mean (TRUE) or median (FALSE)
lastYrs	Last number of years in projection period to calculate summary statistics
AvailMPs	Optional character vector of available MPs (plotted in a different colour)
XLim	Optional limits for the x-axis
ShowCols	Logical. Show the background colours?
ShowLabs	Logical. Show the MP labels?
All	Logical. Plot all MPs (TRUE) or only those above the probability limit (PLim))?

Value

Invisibly returns data frame containing statistics shown in the plot

Author(s)

A. Hordyk

Kplot *KOBE plot: a projection by projection plot of F/FMSY and B/BMSY*

Description

A standard KOBE plot by each method that also shows the percentage of methods that ended up in each quadrant.

Usage

```
Kplot(MSEobj,maxsim=60, MPs=NA, sims=NULL, maxMP=9, nam=NA,
cex.leg=1.5)
```

Arguments

MSEobj	An object of class MSE
maxsim	Maximum number of simulations (lines) to plot on each panel.
MPs	Optional subset MSE object by MP
sims	Optional subset MSE object by simulation
maxMP	Maximum number of MPs to include in plot
nam	The name of the plot
cex.leg	Size of legend

Note

Apologies for the nauseating shading.

Author(s)

T. Carruthers with some additions from A. Hordyk

`L2A` *Length to age conversion*

Description

Simple deterministic length to age conversion given inverse von Bertalanffy growth.

Usage

```
L2A(t0c,Linfc,Kc,Len,maxage)
```

Arguments

`t0c`	Theoretical age at length zero
`Linfc`	Maximum length
`Kc`	Maximum growth rate
`Len`	Length
`maxage`	Maximum age

Value

An age (vector of ages, matrix of ages) corresponding with Len

Author(s)

T. Carruthers

L95target — *A management procedure that adjusts the TAC up/down from reference (target) level (that is a fraction of mean recent premanagement catches) to reach a target mean length of fish caught.*

Description

This MP is based on Ltarget1 proposed by Geromont and Butterworth 2014, but here the target and limit mean lengths are based on the length at maturity distribution rather than an arbitrary multiplicative of the mean length.

Usage

```
L95target(x, Data, reps = 100, yrsmth = 5, buffer=0)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrsmth	Years over which to calculate the mean historical catch
buffer	Parameter controlling the fraction of mean catch to set the reference (or target) TAC level - acts as a precautionary buffer

Value

A numeric vector of TAC recommendations

Author(s)

HF Geromont

References

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

label.class-class — *Label class union for performance metric objects*

Description

Used internally. Nothing to see here!

LBSPR *Apply the Length-based SPR model to DLMtool Data Object*

Description

Apply the Length-based SPR model to DLMtool Data Object

Usage

```
LBSPR(x, Data, yrsmth = 1, reps = 1, lstyrs = 10)
```

Arguments

x	Simulation number
Data	Data object
yrsmth	Number of years to smooth length data - not currently used
reps	Number of repetitions
lstyrs	Last number of years to run model

Author(s)

A. Hordyk

LBSPR_ItEff *Length-based SPR model with HCR that iteratively adjusts Effort*

Description

Iteratively adjusts Effort based on distance between estimated and target SPR (40%), and slope of recent SPR estimates.

Usage

```
LBSPR_ItEff(x, Data, yrsmth=1,reps=5, ...)
```

Arguments

x	Simulation number
Data	Data object
yrsmth	Number of years to smooth length data - not currently used
reps	Number of repetitions. Not currently used
...	ignored

Author(s)

A. Hordyk

LBSPR_ItSel *Length-based SPR model with HCR that iteratively adjusts Selectivity*

Description

Management Procedure which adjusts size-at-selection based on estimated SPR. Entirely untested, and included at to demonstrate MPs of this type.

Usage

```
LBSPR_ItSel(x, Data, yrsmth=1,reps=5, ...)
```

Arguments

x	Simulation number
Data	Data object
yrsmth	Number of years to smooth length data - not currently used
reps	Number of repetitions. Not currently used
...	ignored

Author(s)

A. Hordyk

LBSPR_ItTAC *Length-based SPR model with HCR that iteratively adjusts TAC*

Description

Iteratively adjusts TAC based on distance between estimated and target SPR (40%), and slope of recent SPR estimates.

Usage

```
LBSPR_ItTAC(x, Data, yrsmth=1,reps=5, ...)
```

Arguments

x	Simulation number
Data	Data object
yrsmth	Number of years to smooth length data - not currently used
reps	Number of repetitions
...	ignored

Author(s)

A. Hordyk

`LinInterp` *Linear interpolation of a y value at level xlev based on a vector x and y*

Description

Linear interpolation of a y value at level xlev based on a vector x and y

Usage

```
LinInterp(x, y, xlev, ascending = F, zeroint = F)
```

Arguments

`x`	A vector of x values
`y`	A vector of y values (identical length to x)
`xlev`	A the target level of x from which to guess y
`ascending`	Are the the x values supposed to be ordered before interpolation
`zeroint`	is there a zero-zero x-y intercept?

Author(s)

T. Carruthers

`load.iscam.files` *Reads iSCAM files into a hierarchical R list object*

Description

A function for reading iSCAM input and output files into R

Usage

```
load.iscam.files(model.dir, burnin = 1000, thin = 1, verbose = FALSE)
```

Arguments

`model.dir`	An iSCAM directory
`burnin`	The initial mcmc samples to be discarded
`thin`	The degree of chain thinning 1 in every thin iterations is kept
`verbose`	Should detailed outputs be provided.

Author(s)

Chris Grandin (DFO PBS)

Low_Effort_Non_Target *Low_Effort_Non_Target Fleet*

Description

An object of class Fleet

Usage

```
Low_Effort_Non_Target
```

Format

An object of class `Fleet` of length 1.

Lratio_BHI *Mean length-based indicator MP of Jardim et al. 2015 using Beverton-Holt invariant M/K ratio = 1.5 and assumes FMSY = M.*

Description

The TAC is adjusted by the ratio alpha, where the numerator is the mean length of the catch (of lengths larger than Lc) and the denominator is the mean length expected when FMSY = M and M/K = 1.5. Natural mortality M and von Bertalanffy K are not used in this MP (see Appendix A of Jardim et al. 2015). Here, Lc is the length at full selection (LFS).

Usage

```
Lratio_BHI(x, Data, reps, yrsmth = 3)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples
`yrsmth`	The most recent years of data to smooth the calculation of the mean length

Details

Argument yrsmth currently takes the mean length of the most recent 3 years of data as a smoother.

Author(s)

Coded by Q. Huynh. Developed by Jardim et al. (2015)

References

Ernesto Jardim, Manuela Azevedo, Nuno M. Brites, Harvest control rules for data limited stocks using length-based reference points and survey biomass indices, Fisheries Research, Volume 171, November 2015, Pages 12-19, ISSN 0165-7836, https://doi.org/10.1016/j.fishres.2014.11.013

`Lratio_BHI2` *The more general version of the mean length-based indicator MP of Jardim et al. 2015.*

Description

The TAC is adjusted by the ratio alpha, where the numerator is the mean length of the catch (of lengths larger than Lc) and the denominator is the mean length as a function of Linf, FMSY/M, and M/K (see Appendix A of Jardim et al. 2015). Here, Lc is the length at full selection (LFS).

Usage

```
Lratio_BHI2(x, Data, reps, yrsmth = 3)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples
`yrsmth`	The most recent years of data to smooth the calculation of the mean length

Details

Argument yrsmth currently takes the mean length of the most recent 3 years of data as a smoother.

Author(s)

Coded by Q. Huynh. Developed by Jardim et al. (2015)

References

Ernesto Jardim, Manuela Azevedo, Nuno M. Brites, Harvest control rules for data limited stocks using length-based reference points and survey biomass indices, Fisheries Research, Volume 171, November 2015, Pages 12-19, ISSN 0165-7836, https://doi.org/10.1016/j.fishres.2014.11.013

LSRA *Estimates R0 using SRA to match current F estimates and avoid penalities for low stock sizes*

Description

Estimates R0 using SRA to match current F estimates and avoid penalities for low stock sizes

Usage

```
LSRA(x, FF, Chist_arr, M, Mat_age, Wt_age, sel, Recdevs, h)
```

Arguments

x	a position in the various arrays and vectors that corresponds with a simulation (for use with sapply)
FF	a vector of recent fishign mortality rates (apical Fs)
Chist_arr	a vector of historical catch observations [nyears]
M	a vector of natural mortality rates [nsim]
Mat_age	a matrix of maturity at age [nsim x nage]
Wt_age	a matrix of weight at age [nsim x nage]
sel	a matrix of selectivity at age [nsim x nage]
Recdevs	a matrix of recruitment deviations [nsim x nyears]
h	a vector of steepness values of the Bev-Holt Stock-Recruitment relationship

Value

all package data objects are placed in the global namespace dir

Author(s)

T. Carruthers

LSRA2 *Alternative version of LSRA that's a wrapper for LSRA_opt to return the right type of output (mode) using sapply*

Description

Alternative version of LSRA that's a wrapper for LSRA_opt to return the right type of output (mode) using sapply

Usage

```
LSRA2(x, lnR0s, FF, Chist, M, Mat_age, Wt_age, sel, Recdevs, h, mode = 2)
```

Arguments

x	a position in the various arrays and vectors that corresponds with a simulation (for use with sapply)
lnR0s	a vector nsim long that are estimated R0 values
FF	a vector of recent fishign mortality rates (apical Fs)
Chist	a vector of historical catch observations [nyears]
M	a vector of natural mortality rates [nsim]
Mat_age	a matrix of maturity at age [nsim x nage]
Wt_age	a matrix of weight at age [nsim x nage]
sel	a matrix of selectivity at age [nsim x nage]
Recdevs	a matrix of recruitment deviations [nsim x nyears]
h	a vector of steepness values of the Bev-Holt Stock-Recruitment relationship
mode	optimization or plotting

Value

all package data objects are placed in the global namespace dir

Author(s)

T. Carruthers

LSRA_MCMC_sim *Internal SRA MCMC CPP code*

Description

Rcpp version of R code

Usage

```
LSRA_MCMC_sim(nits, pars, JumpCV, adapt, parLB, parUB, R0ind, inflind, slpind,
  RDind, nyears, maxage, M, Mat_age, Wt_age, Chist_a, Umax, h, CAA, CAAadj,
  sigmaR)
```

Arguments

nits	number of iterations
pars	vector of parameters
JumpCV	jump cv vector
adapt	adapt vector
parLB	lower bounds
parUB	upper bounds
R0ind	index for R0
inflind	index for inflection
slpind	index for slope
RDind	index for recruitment deviations
nyears	number of projection years
maxage	maximum age
M	Natural mortality
Mat_age	A vector of maturity at age
Wt_age	A vector of weight at age
Chist_a	A vector of historical catch observations (nyears long) going back to unfished conditions
Umax	A numeric value representing the maximum harvest rate for any age class (rejection of sims where this occurs)
h	steepness of SRR
CAA	A matrix nyears (rows) by nages (columns) of catch at age (age 1 to maxage in length)
CAAadj	internal parameter
sigmaR	A numeric value representing the prior standard deviation of log space recruitment deviations

Author(s)

A. Hordyk

LSRA_opt *Internal estimation function for LSRA and LSRA2 functions*

Description

Internal estimation function for LSRA and LSRA2 functions

Usage

```
LSRA_opt(param, FF_a, Chist, M_a, Mat_age_a, Wt_age_a, sel_a, Recdevs_a, h_a,
  Umax = 0.5, mode = 1)
```

Arguments

param	a numeric value representing log(R0)
FF_a	numeric value, recent fishign mortality rate (apical F)
Chist	a vector of historical catch observations [nyears]
M_a	numeric value, natural mortality rate
Mat_age_a	a vector of maturity at age [nage]
Wt_age_a	a vector of weight at age [nage]
sel_a	a vector of selectivity at age [nage]
Recdevs_a	a vector of recruitment deviations [nyears]
h_a	a numeric value of steepness values of the Bev-Holt Stock-Recruitment relationship
Umax	maximum harvest rate per year
mode	1-5 see below

Value

depends on mode but could be 1:objective function 2:trajectory of Fs 3: SSB depletion 4:log(R0) 5:diagnostic plots

Author(s)

T. Carruthers

LstepCC1 *A management procedure that incrementally adjusts the TAC according to the mean length of recent catches.*

Description

The least biologically precautionary of four adaptive length-based MPs proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
LstepCC1(x, Data, reps = 100, yrsmth = 5, xx=0, stepsz=0.05,
llim=c(0.96,0.98,1.05))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrsmth	Years over which to smooth recent estimates of surplus production
xx	Parameter controlling the fraction of mean catch to start using in first year
stepsz	Parameter controlling the size of the TAC update increment.
llim	A vector of length reference points that determine the conditions for increasing, maintaining or reducing the TAC.

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

LstepCC4 *A management procedure that incrementally adjusts the TAC according to the mean length of recent catches.*

Description

The most biologically precautionary of four adaptive length-based MPs proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
LstepCC4(x, Data, reps = 100, yrsmth = 5, xx=0.3, stepsz=0.05,
llim=c(0.96,0.98,1.05))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrsmth	Years over which to smooth recent estimates of surplus production
xx	Parameter controlling the fraction of mean catch to start using in first year
stepsz	Parameter controlling the size of the TAC update increment.
llim	A vector of length reference points that determine the conditions for increasing, maintaining or reducing the TAC.

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

LstepCE1 — *A management procedure that incrementally adjusts the TAC according to the mean length of recent catches.*

Description

A effort-based version of least biologically precautionary of four adaptive length-based MPs proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
LstepCE1(x, Data, reps = 100, yrsmth = 5, xx = 0, stepsz = 0.05,
llim = c(0.96, 0.98, 1.05))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of effort samples
yrsmth	Years over which to smooth recent estimates of surplus production
xx	Parameter controlling the fraction of mean catch to start using in first year
stepsz	Parameter controlling the size of the effort update increment.
llim	A vector of length reference points that determine the conditions for increasing, maintaining or reducing the effort.

Value

A numeric vector of input controls

Author(s)

T. Carruthers

LstepCE2 *A management procedure that incrementally adjusts the Effort according to the mean length of recent catches.*

Description

A effort-based version of one of the four adaptive length-based MPs proposed by Geromont and Butterworth 2014.

Usage

```
LstepCE2(x, Data, reps = 100, yrsmth = 5, xx = 0, stepsz = 0.1,
llim = c(0.96, 0.98, 1.05))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of samples
yrsmth	Years over which to smooth recent estimates of surplus production
xx	Parameter controlling the fraction of mean catch to start using in first year
stepsz	Parameter controlling the size of the effort update increment.
llim	A vector of length reference points that determine the conditions for increasing, maintaining or reducing the effort.

Value

A numeric vector of input controls

Author(s)

T. Carruthers

`Ltarget1` *A management procedure that incrementally adjusts the TAC to reach a target mean length in catches.*

Description

The least biologically precautionary of four target length MPs proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
Ltarget1(x, Data, reps = 100, yrsmth = 5, xx=0, xL=1.05)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples
`yrsmth`	Years over which to smooth recent estimates of surplus production
`xx`	Parameter controlling the fraction of mean catch to start using in first year
`xL`	Parameter controlling the magnitude of the target mean length of catches relative to average length in catches.

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

`Ltarget4` *A management procedure that incrementally adjusts the TAC to reach a target mean length in catches.*

Description

The most biologically precautionary of four target length MPs proposed by Geromont and Butterworth 2014. Tested by Carruthers et al. 2015

Usage

```
Ltarget4(x, Data, reps = 100, yrsmth = 5, xx=0.2, xL=1.15)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of TAC samples
`yrsmth`	Years over which to smooth recent estimates of surplus production
`xx`	Parameter controlling the fraction of mean catch to start using in first year
`xL`	Parameter controlling the magnitude of the target mean length of catches relative to average length in catches.

Details

Tested by Carruthers et al. 2015.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

Carruthers et al. 2015. Performance evaluation of simple management procedures. Fish and Fisheries. In press.

Geromont, H.F., Butterworth, D.S. 2014. Generic management procedures for data-poor fisheries; forecasting with few data. ICES J. Mar. Sci. doi:10.1093/icesjms/fst232

`LtargetE1` *A management procedure that incrementally adjusts the Effort to reach a target mean length in catches.*

Description

A effort based version of the least biologically precautionary of four target length MPs proposed by Geromont and Butterworth 2014.

Usage

```
LtargetE1(x, Data, reps = 100, yrsmth = 5, xx = 0, xL = 1.05)
```

Arguments

`x`	A position in data-limited methods data object
`Data`	A data-limited methods data object
`reps`	The number of samples
`yrsmth`	Years over which to smooth recent estimates of surplus production
`xx`	Parameter controlling the fraction of mean catch to start using in first year
`xL`	Parameter controlling the magnitude of the target mean length of catches relative to average length in catches.

Value

A numeric vector of input controls

Author(s)

T. Carruthers

`LtargetE4` *A management procedure that incrementally adjusts the Effort to reach a target mean length in catches.*

Description

A effort based version of the most biologically precautionary of four target length MPs proposed by Geromont and Butterworth 2014.

Usage

```
LtargetE4(x, Data, reps = 100, yrsmth = 5, xx = 0, xL = 1.15)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of samples
yrsmth	Years over which to smooth recent estimates of surplus production
xx	Parameter controlling the fraction of mean catch to start using in first year
xL	Parameter controlling the magnitude of the target mean length of catches relative to average length in catches.

Value

A numeric vector of input controls

Author(s)

T. Carruthers

LTY *Performance Metric: Probability Long-Term Yield > 0.5 Relative Yield*

Description

Performance Metric: Probability Long-Term Yield > 0.5 Relative Yield

Usage

```
LTY(MSEobj = NULL)
```

Arguments

MSEobj	An object of class MSE

Value

An object of class PMobj

Examples

```
## Not run:
LTY(myMSE)

## End(Not run)
```

Mackerel *Mackerel Stock*

Description

An object of class Stock

Usage

```
Mackerel
```

Format

An object of class Stock of length 1.

makePerf *Convert a OM object to one without observation or process error*

Description

Takes an existing OM object and converts it to one without any observation error, and very little process error. Used for debugging and testing that MPs perform as expected under perfect conditions.

Usage

```
makePerf(OMin, except = NULL)
```

Arguments

OMin	An object of class OM
except	An optional vector of slot names in the OM that will not be changed (not tested perfectly so watch out!)

Value

A new OM object

Author(s)

A. Hordyk

makeTransparent *Make colors transparent*

Description

Make colors transparent

Usage

```
makeTransparent(someColor, alpha = 100)
```

Arguments

someColor	Character string describing color
alpha	transparency

Author(s)

T. Carruthers

matlenlim *A data-limited method in which fishing retention is set according to the maturity curve*

Description

An example of the implementation of input controls in the DLM toolkit, where retention-at-length is set equivalent to maturity-at-length

Usage

```
matlenlim(x, Data, ...)
```

Arguments

x	A position in a data-limited methods object
Data	A data-limited methods object
...	Optional additional arguments that are ignored. Note arguments reps or ... are required for all input controls

Value

A input control recommendation object

Author(s)

T. Carruthers

References

Made-up for this package

`matlenlim2` *A data-limited method in which fishing vulnerability is set slightly higher than the maturity curve*

Description

An example of the implementation of input controls in the DLM toolkit, where selectivity-at-length is set slightly higher than the maturity-at-length

Usage

```
matlenlim2(x, Data, ...)
```

Arguments

`x`	A position in a data-limited methods object
`Data`	A data-limited methods object
`...`	Optional additional arguments that are ignored. Note arguments `reps` or `...` are required for all input controls

Value

A vector of input control recommendations, with values for length at first capture and full selection

Author(s)

A. Hordyk

References

Made-up for this package

MCD *Mean Catch Depletion*

Description

A simple average catch-depletion MP that was included to demonstrate just how informative an estimate of current stock depletion can be. TAC=2*D*AvC

Usage

```
MCD(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the quota recommendation

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

MCD4010 *Mean Catch Depletion*

Description

A simple average catch-depletion MP linked to a 40-10 harvest controle rule that was included to demonstrate just how informative an estimate of current stock depletion can be. TAC=d(1-d)AvC

Usage

```
MCD4010(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the quota recommendation

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

`mconv` *Get log normal mean from transformed space mean and standard deviation*

Description

Get log normal mean from transformed space mean and standard deviation

Usage

```
mconv(m, sd)
```

Arguments

`m`	mean
`sd`	standard deviation

Value

numeric

Author(s)

T. Carruthers

`minlenLopt1` *This input control sets the minimum length of fish caught to a fraction of the length that maximises the biomass, Lopt.*

Description

This aim of this simple MP is restrict the catch of small fish to rebuild the stock biomass towards the optimal length, Lopt, expressed in terms of the growth parameters Lopt=b/(M/k+b) (Hordyk et al. (2014)

Usage

```
minlenLopt1(x, Data, reps = 100, buffer = 0.1)
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
buffer	Parameter controlling the fraction of Lopt to set the minimum length of fish caught: minlen=Lopt*(0.7+buffer).

Value

The length at first caprture, LFC, and length at full selectivity

Author(s)

HF Geromont

References

Hordyk, A., Ono, K., Sainsbury, K., Loneragan, N., and J. Prince. 2014. Some explorations of the life history ratios to describe length composition, spawning-per-recruit, and the spawning potential ratio ICES Journal of Marine Science, doi:10.1093/icesjms/fst235.

ML2D *Depletion and F estimation from mean length of catches*

Description

A highly dubious means of getting very uncertain estimates of current stock biomass and (equilibrium) fishing mortality rate from growth, natural mortality rate, recruitment and fishing selectivity.

Usage

```
ML2D(OM,ML,nsim=100,ploty=T,Dlim=c(0.05,0.6))
```

Arguments

OM	An object of class 'OM'
ML	A estimate of current mean length of catches
nsim	Number of simulations
ploty	Produce a plot of depletion and F
Dlim	Limits on the depletion that is returned as a fraction of unfished biomass.

Value

An object of class 'OM' with 'D' slot populated

Author(s)

T. Carruthers

`movdistil` *Simplified a multi-area transition matrix into the best 2 x 2 representation*

Description

A Function that takes a larger movement matrix, identifies the most parsimonious representation of 2 non-mixed areas and returns the final unfished movement matrix

Usage

```
movdistil(movtab)
```

Arguments

`movtab`	a table of estimated movements

Author(s)

T. Carruthers

`movfit` *Optimization function that returns the squared difference between user specified and calculated movement parameters. (deprecated: now in Rcpp)*

Description

The user specifies the probability of staying in the same area and spatial heterogeneity (both in the unfished state). This function returns the squared difference between these values and those produced by the three logit movement model.

Usage

```
movfit(par,prb,frac)
```

Arguments

`par`	Three parameters in the logit space that control the four probabilities of moving between 2 areas
`prb`	User specified probability that individuals in area 1 remain in that area (unfished conditions)
`frac`	User specified fraction of individuals found in area 1 (unfished conditions)

Details

This is paired with getmov2 to find the correct movement model.

Author(s)

T. Carruthers

`movfit_Rcpp` *Rcpp version of the Optimization function that returns the squared difference between user specified and calculated movement parameters.*

Description

The user specifies the probability of staying in the same area and spatial heterogeneity (both in the unfished state). This function returns the squared difference between these values and those produced by the three logit movement model.

Usage

```
movfit_Rcpp(par, prb, frac)
```

Arguments

`par`	Three parameters in the logit space that control the four probabilities of moving between 2 areas
`prb`	User specified probability that individuals in area 1 remain in that area (unfished conditions)
`frac`	User specified fraction of individuals found in area 1 (unfished conditions)

Details

This is paired with getmov to find the correct movement model.

Author(s)

T. Carruthers with an amateur attempt at converting to Rcpp by A. Hordyk (but it works!)

MPStats *Calculate Statistics for MP Performance*

Description

Function calculates probabilities and other statistics for a range of performance metrics

Usage

```
MPStats(MSEobj, PMRefs = list(B_BMSY = 0.5, SSB_SSB0 = 0.2, F_FMSY = 1,
AAVY = 30, AAVE = 30), lastYrs = 10, UseMean = TRUE, msg = TRUE)
```

Arguments

MSEobj	An object of class MSE
PMRefs	A list of reference points for the performance metrics (must be named)
lastYrs	The last number of years in the projection to calculate the statistics
UseMean	Logical. Calculate mean (TRUE) or median (FALSE)?
msg	Logical. Print out messages?

Author(s)

A. Hordyk

MRnoreal *An marine reserve in area 1 with no spatial reallocation of fishing effort*

Description

A spatial control that prevents fishing in area 1 and does not reallocate this fishing effort to area 2.

Usage

```
MRnoreal(x, Data, ...)
```

Arguments

x	A position in data / simulation object DLM
Data	A data limited methods data object
...	Optional additional arguments that are ignored. Note arguments reps or ... are required for all input controls

Author(s)

T. Carruthers

MRreal *An marine reserve in area 1 with full reallocation of fishing effort*

Description

A spatial control that prevents fishing in area 1 and reallocates this fishing effort to area 2.

Usage

```
MRreal(x, Data, ...)
```

Arguments

x	A position in data / simulation object DLM
Data	A data limited methods data object
...	Optional additional arguments that are ignored. Note arguments reps or ... are required for all input controls

Author(s)

T. Carruthers

MSE-class *Class* 'MSE'

Description

A Management Strategy Evaluation object that contains information about simulation conditions and performance of data-limited methods

Slots

Name Name of the MSE object. Single value. Character string

nyears The number of years for the historical simulation. Single value. Positive integer

proyears The number of years for the projections - closed loop simulations. Single value. Positive integer

nMPs Number of management procedures simulation tested. Single value. Positive integer.

MPs The names of the MPs that were tested. Vector of length nMPs. Character strings.

nsim Number of simulations. Single value. Positive integer

OM A table of sampled parameter of the operating model. Table object of nsim rows. Real numbers

- RefY: reference yield, the highest long-term yield (mean over last five years of projection) obtained from a fixed F strategy. This is a useful reference point for framing performance of MPs because it standardizes for starting point and future productivity.
- M: instantaneous natural mortality rate
- Depletion: stock depletion (biomass / unfished biomass) in the final historical year (prior to projection)
- A: abundance (biomass) updated in each management update of projection
- BMSY_B0: most productive stock size relative to unfished
- FMSY_M: fishing mortality rate divided by natural mortality rate
- Mgrad: mean average percentage gradient in natural mortality rate (percentage per time step)
- Msd: interannual variability in natural mortality rate (lognormal CV)
- procsd: process error - CV in log-normal recruitment deviations
- Esd: interannual variability in historical effort (fishing mortality rate)
- dFfinal: gradient in fishing mortality rate over final five years of the historical simulation
- MSY: Maximum Sustainable Yield
- qinc: mean percentage increase in fishing efficiency (catchability) in projected years (input controls only)
- qcv: interannual variability in future fishing efficiency (catchability) in projected years (input controls only)
- FMSY: Fishing mortality rate at Maximum Sustainable Yield
- Linf: maximum length (von Bertalanffy Linf parameter)
- K: maximum growth rate (von Bertalanffy K parameter)
- t0: theoretical length at age zero (von Bertalanffy t0 parameter)
- hs: steepness of the stock recruitment relationship (the fraction of unfished recruitment at a fifth of unfished stock levels)
- Linfgrad: mean gradient in maximum length (per cent per time step)
- Kgrad: mean gradient in maximum growth rate (per cent per time step)
- Linfsd: interannual variability in maximum length (log normal CV)
- Ksd: interannual variability in maximum growth rate (log normal CV)
- ageM: age at 50 per cent maturity
- LFS: length at full selection (the shortest length class where fishery selectivity is 100 per cent)
- age05: the age at 5 percent selectivity (ascending limb of selectivity curve)
- Vmaxage: the selectivity of the oldest age class (controls dome shape of selectivity curve)
- LFC: length at first capture, the smallest length that can be caught by the gear
- OFLreal: the true simulated Over Fishing Limit (FMSY x biomass) updated in each management update of the projection
- Spat_targ: spatial targetting parameter, fishing mortality rate across areas is proportional to vulnerable biomass raised to the power of this number.
- Size_area_1: The size of area 1 relative to area 2
- Frac_area_1: the fraction of unfished biomass inhabiting area 1 (can be seen as fraction of habitat in area 1 or relative size of area 1)
- Prob_staying: the probability that individuals in area 1 remain there between time-steps

- AC: autocorrelation in recruitment

`Obs` A table of sampled parameters of the observation model. Table of nsim rows. Real numbers

- Cbias: bias in observed catches
- Csd: observation error in observed catches (lognormal CV)
- CAA_nsamp: the number of catch-at-age observations per time step
- CAA_ESS: the effective sample size of multinomial catch-at-age observation model (number of independent draws)
- CAL_nsamp: the number of catch-at-length observations per time step
- CAL_ESS: the effective sample size of multinomial catch-at-length observation model (number of independent draws)
- Isd: observation error in relative abundance index (lognormal CV)
- Dbias: bias in observed stock depletion (also applies to depletion Dt for DCAC)
- Mbias: bias in observed natural mortality rate
- FMSY_Mbias: bias in ratio of FMSY to natural mortality rate
- BMSY_B0bias: bias in ratio of most productive stock size relative to unfished
- AMbias: bias in age at 50 per cent maturity
- LFCbias: bias in length at first capture
- LFSbias: bias in length at full selection
- Abias: bias in observed current absolute stock biomass
- Kbias: bias in maximum growth rate (von Bertalanffy K parameter)
- t0bias: bias in theoretical length at age zero (von Bertalanffy t0 parameter)
- Linfbias: bias in maximum length (von Bertalanffy Linf parameter)
- hbias: bias in observed steepness of the stock recruitment relationship
- Irefbias: bias in abundance index corresponding to BMSY stock levels
- Crefbias: bias in MSY prediction (target or reference catch)
- Brefbias: bias in BMSY stock levels (target or reference biomass levels)

`B_BMSY` Simulated biomass relative to BMSY over the projection. An array with dimensions: nsim, nMPs, proyears. Non-negative real numbers

`F_FMSY` Simulated fishing mortality rate relative to FMSY over the projection. An array with dimensions: nsim, nMPs, proyears. Non-negative real numbers

`B` Simulated stock biomass over the projection. An array with dimensions: nsim, nMPs, proyears. Non-negative real numbers

`SSB` Simulated spawning stock biomass over the projection. An array with dimensions: nsim, nMPs, proyears. Non-negative real numbers

`VB` Simulated vulnerable biomass over the projection. An array with dimensions: nsim, nMPs, proyears. Non-negative real numbers

`FM` Simulated fishing mortality rate over the projection. An array with dimensions: nsim, nMPs, proyears. Non-negative real numbers

`C` Simulated catches (taken) over the projection. An array with dimensions: nsim, nMPs, proyears. Non-negative real numbers

`TAC` Simulated Total Allowable Catch (prescribed) over the projection (this is NA for input controls). An array with dimensions: nsim, nMPs, proyears. Non-negative real numbers

SSB_hist Simulated historical spawning stock biomass. An array with dimensions: nsim, nages, nMPs, proyears. Non-negative real numbers

CB_hist Simulated historical catches in weight. An array with dimensions: nsim, nages, nMPs, proyears. Non-negative real numbers

FM_hist Simulated historical fishing mortality rate. An array with dimensions: nsim, nages, nMPs, proyears. Non-negative real numbers

Effort Simulated relative fishing effort in the projection years. An array with dimensions: nsim, nMPs, proyears. Non-negative real numbers

PAA Population at age in last projection year. An array with dimensions: nsim, nMPs, nages. Non-negative real numbers

CAA Catch at age in last projection year. An array with dimensions: nsim, nMPs, nages. Non-negative real numbers

CAL Catch at length in last projection year. An array with dimensions: nsim, nMPs, nCALbins. Non-negative real numbers

CALbins Mid-points of the catch-at-length bins. Vector of length nCALbins. Positive real numbers.

Objects from the Class

Objects can be created by calls of the form `new('MSE', Name, nyears, proyears, nMPs, MPs, nsim, OMtable, Obs,B_`

Author(s)

T. Carruthers

NAor0 *Is a value NA or zero.*

Description

As title

Usage

```
NAor0(x)
```

Arguments

x	A numeric value.

Value

TRUE or FALSE

Author(s)

T. Carruthers

Needed *Data needed to get MPs running*

Description

Wrapper function for DLMdiag that lists what data are needed to run data-limited methods that are current not able to run given a DLM_cdata object

Usage

```
Needed(Data, timelimit=1)
```

Arguments

Data	A data-limited methods data object
timelimit	The maximum time (seconds) taken to complete 10 reps

Author(s)

T. Carruthers

negcorlogspace *A function that samples multivariate normal (logspace) variables*

Description

A function that samples multivariate normal (logspace) variables

Usage

```
negcorlogspace(xmu, ymu, xcv = 0.1, nsim, cor = -0.9, ploty = F)
```

Arguments

xmu	The mean (normal space) of the first (x) variable
ymu	The mean (normal space) of the second (y) variable
xcv	The coefficient of variation (normal space, log normal sd) of the x variable
nsim	The number of random draws
cor	The off-diagonal (symmetrical) correlation among x and y
ploty	Whether a plot of the sampled variables should be produced

Author(s)

T. Carruthers

NFref *No Fishing Reference MP*

Description

A reference MP that sets annual catch to zero (or very close to it). Used for looking at variability in stock with no fishing.

Usage

```
NFref(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of stochastic samples of the quota recommendation

Value

A TAC of 0.01

Author(s)

A. Hordyk

NOAA_plot *National Oceanographic and Atmospheric Administration default plot 1*

Description

A preliminary plot for returning trade-offs plots and performance table for total yield, variability in yield, probability of overfishing and likelihood of biomass dropping below 50 per cent BMSY

Usage

```
NOAA_plot(MSEobj,nam=NA,type=NA,panel=T)
```

Arguments

MSEobj	An object of class MSE
nam	Title of plot
type	Plots full range of data if NA. Plots a subset that meet thresholds if not NA.
panel	Should a two panel plot be made or should plots be made in sequence.

Value

A table of performance metrics.

Author(s)

T. Carruthers

`Obs-class` *Class* `'Obs'`

Description

An operating model component that controls the observation model

Slots

`Name` The name of the observation model object. Single value. Character string.

`Cobs` Log-normal catch observation error expressed as a coefficient of variation. Uniform distribution lower and upper bounds. Non-negative real numbers

`Cbiascv` Log-normal coefficient of variation controlling the sampling of bias in catch observations for each simulation. Uniform distribution lower and upper bounds. Non-negative real numbers

`CAA_nsamp` Number of catch-at-age observation per time step. Uniform distribution lower and upper bounds. Positive real numbers

`CAA_ESS` Effective sample size (independent age draws) of the multinomial catch-at-age observation error model. Uniform distribution lower and upper bounds. Positive integers

`CAL_nsamp` Number of catch-at-length observation per time step. Uniform distribution lower and upper bounds. Positive integers

`CAL_ESS` Effective sample size (independent length draws) of the multinomial catch-at-length observation error model. Uniform distribution lower and upper bounds. Positive integers

`Iobs` Observation error in the relative abundance indices expressed as a coefficient of variation. Uniform distribution lower and upper bounds. Positive real numbers

`Ibiascv` Log-normal coefficient of variation controlling error in observations of relative abundance index. Uniform distribution lower and upper bounds. Positive real numbers

`Btobs` Log-normal coefficient of variation controlling error in observations of current stock biomass among years. Uniform distribution lower and upper bounds. Positive real numbers

`Btbiascv` Uniform-log bounds for sampling persistent bias in current stock biomass. Uniform-log distribution lower and upper bounds. Positive real numbers

`beta` A parameter controlling hyperstability/hyperdepletion where values below 1 lead to hyperstability (an index that decreases slower than true abundance) and values above 1 lead to hyperdepletion (an index that decreases more rapidly than true abundance). Uniform distribution lower and upper bounds. Positive real numbers

`LenMbiascv` Log-normal coefficient of variation for sampling persistent bias in length at 50 percent maturity. Uniform distribution lower and upper bounds. Positive real numbers

`Mbiascv` Log-normal coefficient of variation for sampling persistent bias in observed natural mortality rate. Uniform distribution lower and upper bounds. Positive real numbers

`Kbiascv` Log-normal coefficient of variation for sampling persistent bias in observed growth parameter K. Uniform distribution lower and upper bounds. Positive real numbers

`t0biascv` Log-normal coefficient of variation for sampling persistent bias in observed t0. Uniform distribution lower and upper bounds. Positive real numbers

`Linfbiascv` Log-normal coefficient of variation for sampling persistent bias in observed maximum length. Uniform distribution lower and upper bounds. Positive real numbers

`LFCbiascv` Log-normal coefficient of variation for sampling persistent bias in observed length at first capture. Uniform distribution lower and upper bounds. Positive real numbers

`LFSbiascv` Log-normal coefficient of variation for sampling persistent bias in length-at-full selection. Uniform distribution lower and upper bounds. Positive real numbers

`FMSYbiascv` Log-normal coefficient of variation for sampling persistent bias in FMSY. Uniform distribution lower and upper bounds. Positive real numbers

`FMSY_Mbiascv` Log-normal coefficient of variation for sampling persistent bias in FMSY/M. Uniform distribution lower and upper bounds. Positive real numbers

`BMSY_B0biascv` Log-normal coefficient of variation for sampling persistent bias in BMSY relative to unfished. Uniform distribution lower and upper bounds. Positive real numbers

`Irefbiascv` Log-normal coefficient of variation for sampling persistent bias in relative abundance index at BMSY. Uniform distribution lower and upper bounds. Positive real numbers

`Brefbiascv` Log-normal coefficient of variation for sampling persistent bias in BMSY. Uniform distribution lower and upper bounds. Positive real numbers

`Crefbiascv` Log-normal coefficient of variation for sampling persistent bias in MSY. Uniform distribution lower and upper bounds. Positive real numbers

`Dbiascv` Log-normal coefficient of variation for sampling persistent bias in stock depletion. Uniform distribution lower and upper bounds. Positive real numbers

`Dobs` Log-normal coefficient of variation controlling error in observations of stock depletion among years. Uniform distribution lower and upper bounds. Positive real numbers

`hbiascv` Log-normal coefficient of variation for sampling persistent bias in steepness. Uniform distribution lower and upper bounds. Positive real numbers

`Recbiascv` Log-normal coefficient of variation for sampling persistent bias in recent recruitment strength. Uniform distribution lower and upper bounds. Positive real numbers

Objects from the Class

Objects can be created by calls of the form `new('Obs')`

Note

Its questionable whether the hyperstability/hyperdepletion should be categorised as an observation model characteristic as it is most often driven by fleet dynamics (and therefore should be in the fleet object). Oh well its here and you might want to make it hyperstable beta < 1 or hyperdeplete beta > 1, only.

Author(s)

T. Carruthers and A. Hordyk

Examples

```
showClass('Obs')
```

ObsDescription *ObsDescription*

Description

A data.frame with description of slots for class Obs

Usage

```
ObsDescription
```

Format

An object of class `data.frame` with 29 rows and 2 columns.

OM-class *Class* 'OM'

Description

An object containing all the parameters needed to control the MSE which can be build from component Stock, Fleet and Obs objects. Almost all of these inputs are a vector of length 2 which describes the upper and lower bounds of a uniform distribution from which to sample the parameter.

Slots

`Name` Name of the operating model

`Agency` Name of the agency responsible for the management of the fishery. Character string

`Region` Name of the general geographic region of the fishery. Character string

`Latitude` Latitude (decimal degrees). Negative values represent the South of the Equator. Numeric. Single value

`Longitude` Longitude (decimal degrees). Negative values represent the West of the Prime Meridian. Numeric. Single value

`nsim` The number of simulations

`proyears` The number of projected years

`interval` The assessment interval - how often would you like to update the management system?

`pstar` The percentile of the sample of the management recommendation for each method

`maxF` Maximum instantaneous fishing mortality rate that may be simulated for any given age class

`reps` Number of samples of the management recommendation for each method. Note that when this is set to 1, the mean value of the data inputs is used.

`Species` Scientific name of the species. Genus and species name. Character string

`maxage` The maximum age of individuals that is simulated (there is no 'plus group'). Single value. Positive integer

`R0` The magnitude of unfished recruitment. Single value. Positive real number

`M` Natural mortality rate. Uniform distribution lower and upper bounds. Positive real number

`M2` (Optional) Natural mortality rate at age. Vector of length 'maxage'. Positive real number

`Mexp` Exponent of the Lorenzen function assuming an inverse relationship between M and weight. Uniform distribution lower and upper bounds. Real numbers <= 0.

`Msd` Inter-annual variability in natural mortality rate expressed as a coefficient of variation. Uniform distribution lower and upper bounds. Non-negative real numbers

`Mgrad` Mean temporal trend in natural mortality rate, expressed as a percentage change in M per year. Uniform distribution lower and upper bounds. Real numbers

`h` Steepness of the stock recruit relationship. Uniform distribution lower and upper bounds. Values from 1/5 to 1

`SRrel` Type of stock-recruit relationship. Single value, switch (1) Beverton-Holt (2) Ricker. Integer

`Perr` Process error, the CV of lognormal recruitment deviations. Uniform distribution lower and upper bounds. Non-negative real numbers

`AC` Autocorrelation in recruitment deviations rec(t)=AC*rec(t-1)+(1-AC)*sigma(t). Uniform distribution lower and upper bounds. Non-negative real numbers

`Period` (Optional) Period for cyclical recruitment pattern in years. Uniform distribution lower and upper bounds. Non-negative real numbers

`Amplitude` (Optional) Amplitude in deviation from long-term average recruitment during recruitment cycle (eg a range from 0 to 1 means recruitment decreases or increases by up to 100% each cycle). Uniform distribution lower and upper bounds. 0 < Amplitude < 1

`Linf` Maximum length. Uniform distribution lower and upper bounds. Positive real numbers

`K` von Bertalanffy growth parameter k. Uniform distribution lower and upper bounds. Positive real numbers

`t0` von Bertalanffy theoretical age at length zero. Uniform distribution lower and upper bounds. Non-positive real numbers

`LenCV` Coefficient of variation of length-at-age (assumed constant for all age classes). Uniform distribution lower and upper bounds. Positive real numbers

`Ksd` Inter-annual variability in growth parameter k. Uniform distribution lower and upper bounds. Non-negative real numbers

`Kgrad` Mean temporal trend in growth parameter k, expressed as a percentage change in k per year. Uniform distribution lower and upper bounds. Real numbers

`Linfsd` Inter-annual variability in maximum length. Uniform distribution lower and upper bounds. Non-negative real numbers

`Linfgrad` Mean temporal trend in maximum length, expressed as a percentage change in Linf per year. Uniform distribution lower and upper bounds. Real numbers

`L50` Length at 50 percent maturity. Uniform distribution lower and upper bounds. Positive real numbers

`L50_95` Length increment from 50 percent to 95 percent maturity. Uniform distribution lower and upper bounds. Positive real numbers

`D` Current level of stock depletion SSB(current)/SSB(unfished). Uniform distribution lower and upper bounds. Fraction

`a` Length-weight parameter alpha. Single value. Positive real number

`b` Length-weight parameter beta. Single value. Positive real number

`Size_area_1` The size of area 1 relative to area 2. Uniform distribution lower and upper bounds. Positive real numbers

`Frac_area_1` The fraction of the unfished biomass in stock 1. Uniform distribution lower and upper bounds. Positive real numbers

`Prob_staying` The probability of inviduals in area 1 remaining in area 1 over the course of one year. Uniform distribution lower and upper bounds. Positive fraction.

`Fdisc` Fraction of discarded fish that die. Uniform distribution lower and upper bounds. Non-negative real numbers

`nyears` The number of years for the historical 'spool-up' simulation. Single value. Positive integer

`Spat_targ` Distribution of fishing in relation to spatial biomass: fishing distribution is proportional to B^Spat_targ. Uniform distribution lower and upper bounds. Real numbers

`EffYears` Years representing join-points (vertices) of time-varying effort. Vector. Non-negative real numbers

`EffLower` Lower bound on relative effort corresponding to EffYears. Vector. Non-negative real numbers

`EffUpper` Upper bound on relative effort corresponding to EffYears. Vector. Non-negative real numbers

`Esd` Additional inter-annual variability in fishing mortality rate. Uniform distribution lower and upper bounds. Non-negative real numbers

`qinc` Average percentage change in fishing efficiency (applicable only to forward projection and input controls). Uniform distribution lower and upper bounds. Non-negative real numbers

`qcv` Inter-annual variability in fishing efficiency (applicable only to forward projection and input controls). Uniform distribution lower and upper bounds. Non-negative real numbers

`L5` Shortest length corresponding to 5 percent vulnerability. Uniform distribution lower and upper bounds. Positive real numbers

`LFS` Shortest length that is fully vulnerable to fishing. Uniform distribution lower and upper bounds. Positive real numbers

`Vmaxlen` The vulnerability of fish at Stock@Linf. Uniform distribution lower and upper bounds. Fraction

`isRel` Selectivity parameters in units of size-of-maturity (or absolute eg cm). Single value. Boolean.

`LR5` Shortest length corresponding ot 5 percent retention. Uniform distribution lower and upper bounds. Non-negative real numbers

LFR Shortest length that is fully retained. Uniform distribution lower and upper bounds. Non-negative real numbers

Rmaxlen The retention of fish at Stock@Linf. Uniform distribution lower and upper bounds. Non-negative real numbers

DR Discard rate - the fraction of caught fish that are discarded. Uniform distribution lower and upper bounds. Fraction

SelYears (Optional) Years representing join-points (vertices) at which historical selectivity pattern changes. Vector. Positive real numbers

AbsSelYears (Optional) Calendar years corresponding with SelYears (eg 1951, rather than 1), used for plotting only. Vector (of same length as SelYears). Positive real numbers

L5Lower (Optional) Lower bound of L5 (use ChooseSelect function to set these). Vector. Non-negative real numbers

L5Upper (Optional) Upper bound of L5 (use ChooseSelect function to set these). Vector. Non-negative real numbers

LFSLower (Optional) Lower bound of LFS (use ChooseSelect function to set these). Vector. Non-negative real numbers

LFSUpper (Optional) Upper bound of LFS (use ChooseSelect function to set these). Vector. Non-negative real numbers

VmaxLower (Optional) Lower bound of Vmaxlen (use ChooseSelect function to set these). Vector. Fraction

VmaxUpper (Optional) Upper bound of Vmaxlen (use ChooseSelect function to set these). Vector. Fraction

CurrentYr The current calendar year (final year) of the historical simulations (eg 2011). Single value. Positive integer.

Cobs Log-normal catch observation error expressed as a coefficient of variation. Uniform distribution lower and upper bounds. Non-negative real numbers

Cbiascv Log-normal coefficient of variation controlling the sampling of bias in catch observations for each simulation. Uniform distribution lower and upper bounds. Non-negative real numbers

CAA_nsamp Number of catch-at-age observation per time step. Uniform distribution lower and upper bounds. Positive real numbers

CAA_ESS Effective sample size (independent age draws) of the multinomial catch-at-age observation error model. Uniform distribution lower and upper bounds. Positive integers

CAL_nsamp Number of catch-at-length observation per time step. Uniform distribution lower and upper bounds. Positive integers

CAL_ESS Effective sample size (independent length draws) of the multinomial catch-at-length observation error model. Uniform distribution lower and upper bounds. Positive integers

Iobs Observation error in the relative abundance indices expressed as a coefficient of variation. Uniform distribution lower and upper bounds. Positive real numbers

Ibiascv Log-normal coefficient of variation controlling error in observations of relative abundance index. Uniform distribution lower and upper bounds. Positive real numbers

Btobs Log-normal coefficient of variation controlling error in observations of current stock biomass among years. Uniform distribution lower and upper bounds. Positive real numbers

`Btbiascv` Uniform-log bounds for sampling persistent bias in current stock biomass. Uniform-log distribution lower and upper bounds. Positive real numbers

`beta` A parameter controlling hyperstability/hyperdepletion where values below 1 lead to hyperstability (an index that decreases slower than true abundance) and values above 1 lead to hyperdepletion (an index that decreases more rapidly than true abundance). Uniform distribution lower and upper bounds. Positive real numbers

`LenMbiascv` Log-normal coefficient of variation for sampling persistent bias in length at 50 percent maturity. Uniform distribution lower and upper bounds. Positive real numbers

`Mbiascv` Log-normal coefficient of variation for sampling persistent bias in observed natural mortality rate. Uniform distribution lower and upper bounds. Positive real numbers

`Kbiascv` Log-normal coefficient of variation for sampling persistent bias in observed growth parameter K. Uniform distribution lower and upper bounds. Positive real numbers

`t0biascv` Log-normal coefficient of variation for sampling persistent bias in observed t0. Uniform distribution lower and upper bounds. Positive real numbers

`Linfbiascv` Log-normal coefficient of variation for sampling persistent bias in observed maximum length. Uniform distribution lower and upper bounds. Positive real numbers

`LFCbiascv` Log-normal coefficient of variation for sampling persistent bias in observed length at first capture. Uniform distribution lower and upper bounds. Positive real numbers

`LFSbiascv` Log-normal coefficient of variation for sampling persistent bias in length-at-full selection. Uniform distribution lower and upper bounds. Positive real numbers

`FMSYbiascv` Log-normal coefficient of variation for sampling persistent bias in FMSY. Uniform distribution lower and upper bounds. Positive real numbers

`FMSY_Mbiascv` Log-normal coefficient of variation for sampling persistent bias in FMSY/M. Uniform distribution lower and upper bounds. Positive real numbers

`BMSY_B0biascv` Log-normal coefficient of variation for sampling persistent bias in BMSY relative to unfished. Uniform distribution lower and upper bounds. Positive real numbers

`Irefbiascv` Log-normal coefficient of variation for sampling persistent bias in relative abundance index at BMSY. Uniform distribution lower and upper bounds. Positive real numbers

`Brefbiascv` Log-normal coefficient of variation for sampling persistent bias in BMSY. Uniform distribution lower and upper bounds. Positive real numbers

`Crefbiascv` Log-normal coefficient of variation for sampling persistent bias in MSY. Uniform distribution lower and upper bounds. Positive real numbers

`Dbiascv` Log-normal coefficient of variation for sampling persistent bias in stock depletion. Uniform distribution lower and upper bounds. Positive real numbers

`Dobs` Log-normal coefficient of variation controlling error in observations of stock depletion among years. Uniform distribution lower and upper bounds. Positive real numbers

`hbiascv` Log-normal coefficient of variation for sampling persistent bias in steepness. Uniform distribution lower and upper bounds. Positive real numbers

`Recbiascv` Log-normal coefficient of variation for sampling persistent bias in recent recruitment strength. Uniform distribution lower and upper bounds. Positive real numbers

`TACFrac` Mean fraction of TAC taken. Uniform distribution lower and upper bounds. Positive real number.

TACSD Log-normal coefficient of variation in the fraction of Total Allowable Catch (TAC) taken. Uniform distribution lower and upper bounds. Non-negative real numbers.

TAEFrac Mean fraction of TAE taken. Uniform distribution lower and upper bounds. Positive real number.

TAESD Log-normal coefficient of variation in the fraction of Total Allowable Effort (TAE) taken. Uniform distribution lower and upper bounds. Non-negative real numbers.

SizeLimFrac The real minimum size that is retained expressed as a fraction of the size. Uniform distribution lower and upper bounds. Positive real number.

SizeLimSD Log-normal coefficient of variation controlling mismatch between a minimum size limit and the real minimum size retained. Uniform distribution lower and upper bounds. Non-negative real numbers.

cpars A list of custom parameters (single parameters are a vector nsim long, time series are a matrix nsim rows by nyears columns)

seed A random seed to ensure users can reproduce results exactly

Source A reference to a website or article from which parameters were taken to define the operating model

Objects from the Class

Objects can be created by calls of the form `new('OM', Stock, Fleet, Obs, Imp)`.

Author(s)

T. Carruthers and A. Hordyk

`OMDescription` *OMDescription*

Description

A data.frame with description of slots for class OM

Usage

```
OMDescription
```

Format

An object of class `data.frame` with 104 rows and 2 columns.

OMdoc *Generate OM Documentation Report*

Description

Generate OM Documentation Report

Usage

```
OMdoc(OM = NULL, rmd.source = NULL, overwrite = FALSE, out.file = NULL,
  inc.plot = TRUE, render = TRUE, output = "html_document")
```

Arguments

OM	An object of class 'OM' or the name of an OM xlsx file
rmd.source	Optional. Name of the source.rmd file corresponding to the 'OM'. Default assumption is that the file is 'OM@Name.Rmd'
overwrite	Logical. Should existing files be overwritten?
out.file	Optional. Character. Name of the output file. Default is the same as the text file.
inc.plot	Logical. Should the plots be included?
render	Logical. Should the document be compiled? May be useful to turn off if there are problems with compililing the Rmd file.
output	Character. Output file type. Default is 'html_document'. 'pdf_document' is available but may require additional software and have some formatting issues.

Value

Creates a Rmarkdown file and compiles a HTML report file in the working directory.

Author(s)

A. Hordyk

Examples

```
## Not run:
OMinit('myOM', templates=list(Stock='Herring', Fleet='Generic_Fleet', Obs='Generic_Obs',
Imp='Perfect_Imp'), overwrite=TRUE)
myOM <- XL2OM('myOM.xlsx')
OMdoc(myOM)

## End(Not run)
```

OMexample *Copy example OM XL and OM Documentation*

Description

Copy example OM XL and OM Documentation

Usage

```
OMexample()
```

Examples

```
## Not run:
OMexample()

## End(Not run)
```

OMinit *Initialize Operating Model*

Description

Generates an Excel spreadsheet and a source.rmd file in the current working directory for specifying and documenting a DLMtool Operating Model.

Usage

```
OMinit(name = NULL, ..., files = c("xlsx", "rmd"), overwrite = FALSE)
```

Arguments

name	The name of the Excel and source.rmd file to be created in the working directory (character). Use 'example' for a populated example OM XL and documentation file.
...	Optional DLMtool objects to use as templates: OM, Stock, Fleet, Obs, or Imp objects
files	What files should be created: 'xlsx', 'rmd', or c('xlsx', 'rmd') (default: both) to use as templates for the Operating Model.
overwrite	Logical. Should files be overwritten if they already exist?

Value

name.xlsx and name.rmd files are created in the working directory.

Author(s)

A. Hordyk

Examples

```
## Not run:
# Create an Excel OM template and rmd file called 'myOM.xlsx' and 'myOM.rmd':
OMinit('myOM')

# Create an Excel OM template and text file called 'myOM.rmd' and 'myOM.rmd', using
another OM as a template:
OMinit('myOM', myOM)

# Create an Excel OM template and text file called 'myOM.rmd' and 'myOM.rmd', using
the Stock object 'Herring' as a template:
OMinit('myOM', Herring)

# Create an Excel OM template and text file called 'myOM.rmd' and 'myOM.rmd', using
the Stock object 'Herring', and Obs object 'Generic_obs' as templates:
OMinit('myOM', Herring, Generic_obs)

## End(Not run)
```

OM_xl *Read in operating model parameters from Excel spreadsheet*

Description

A function to read in operating model parameters from an Excel spreadsheet with tabs named following specific convention. Since DLMtool 4.5 this function is no longer recommended. Use 'OMinit' instead.

Usage

```
OM_xl(fname, stkname, fpath = '', saveCSV = FALSE)
```

Arguments

fname	Name of the Excel spreadsheet file. Must include file extension.
stkname	Name of the Stock. Only required if more than one Stock in the Excel file.
fpath	Full file path, if file is not in current working directory
saveCSV	Do you also want to save the Stock, Fleet and Observation parameters to CSV files?

Details

The Excel spreadsheet must have tabs named with the following convention. For example if `stkname` is 'myFish', the Stock parameters are in a tab named 'myFishStock', Fleet parameters in a tab named 'myFishFleet', Observation parameters in a tab named 'myFishObs', and Implementation in 'myFishImp'. All tabs (Stock, Fleet, Obs, and Imp) must be present for a single stock. You can have multiple stocks in a single spreadsheet, provided that the stock names are different.

Value

A object of class OM

Author(s)

A. Hordyk

Examples

```
## Not run:
OM <- OM_xl(fname='OMTables.xlsx', stkname='myFish')

## End(Not run)
```

---

optBH *Wrapper for estimating stock recruitment parameters from resampled stock-recruitment data*

Description

Wrapper for estimating stock recruitment parameters from resampled stock-recruitment data

Usage

```
optBH(x, SSB, rec, SSBpR, R0temp, pars, frac = 0.5, plot = F)
```

Arguments

`x`	position (currently redundant)
`SSB`	'observations' of spawning biomass
`rec`	'observations' (model predictions) of recruitment
`SSBpR`	spawning stock biomass per recruit at unfished conditions
`R0temp`	an initial guess at the level of unfished recruitment
`pars`	an initial guess at model parameters steepness and R0
`frac`	the fraction of observations for resampling
`plot`	should a plot of model fit be produced?

Author(s)

T. Carruthers

`optMSY` *Optimize yield for a single simulation*

Description

Optimize yield for a single simulation

Usage

```
optMSY(logFa, Asize_c, nareas, maxage, Ncurr, pyears, M_age, MatAge, WtAge,
  Vuln, Retc, Prec, movc, SRrelc, Effind, Spat_targc, hc, R0c, SSBpRc, aRc, bRc,
  Qc, maxF, useCPP = TRUE)
```

Arguments

`logFa`	log apical fishing mortality
`Asize_c`	A vector of length areas with relative size of areas
`nareas`	Number of area
`maxage`	Maximum age
`Ncurr`	Current N-at-age
`pyears`	Number of projection years
`M_age`	M-at-age
`MatAge`	Maturity-at-age
`WtAge`	Weight-at-age
`Vuln`	Vulnerablity-at-age
`Retc`	Retention-at-age
`Prec`	Recruitment error
`movc`	Movement matrix
`SRrelc`	SR Relationship
`Effind`	Historical effort
`Spat_targc`	Spatial targeting
`hc`	Steepness
`R0c`	Unfished recruitment by area
`SSBpRc`	Unfished spawning stock per recruit by area
`aRc`	Ricker aR
`bRc`	Ricker bR
`Qc`	Catchability
`maxF`	A numeric value specifying the maximum fishing mortality for any single age class
`useCPP`	logical - use the CPP code? For testing purposes only

Author(s)

A. Hordyk

`optQ` *Optimize q for a single simulation*

Description

Optimize q for a single simulation

Usage

```
optQ(logQ, depc, SSB0c, nareas, maxage, Ncurr, pyears, M_age, Asize_c, MatAge,
  WtAge, Vuln, Retc, Prec, movc, SRrelc, Effind, Spat_targc, hc, R0c, SSBpRc,
  aRc, bRc, maxF, useCPP)
```

Arguments

`logQ`	log q
`depc`	Depletion value
`SSB0c`	Unfished spawning biomass
`nareas`	Number of areas
`maxage`	Maximum age
`Ncurr`	Current N-at-age
`pyears`	Number of years to project population dynamics
`M_age`	M-at-age
`Asize_c`	Numeric vector (length nareas) with size of each area
`MatAge`	Maturity-at-age
`WtAge`	Weight-at-age
`Vuln`	Vulnerability-at-age
`Retc`	Retention-at-age
`Prec`	Recruitment error by year
`movc`	movement matrix
`SRrelc`	SR parameter
`Effind`	Historical fishing effort
`Spat_targc`	Spatial targetting
`hc`	Steepness
`R0c`	Unfished recruitment by area
`SSBpRc`	Unfished spawning biomass per recruit by area
`aRc`	Ricker aR
`bRc`	Ricker bR
`maxF`	maximum F
`useCPP`	Logical. Use the CPP code?

Author(s)

A. Hordyk

---

ourReefFish *Example data object*

Description

Example data object with a number of output control MPs run on it, and includes resulting distributions of TACs

An object of class Data

Usage

```
data('ourReefFish')

ourReefFish
```

Format

An object of class `Data` of length 1.

Examples

```
## Not run:
data(ourReefFish)
str(ourReefFish)
plot(ourReefFish)

## End(Not run)
```

Overages *Overages Imp*

Description

An object of class Imp

Usage

```
Overages
```

Format

An object of class `Imp` of length 1.

P10 *Performance Metric: Probability B > 0.1 BMSY*

Description

Performance Metric: Probability B > 0.1 BMSY

Usage

```
P10(MSEobj = NULL)
```

Arguments

MSEobj	An object of class MSE

Value

An object of class PMobj

Examples

```
## Not run:
P10(myMSE)

## End(Not run)
```

P100 *Performance Metric: Probability B > BMSY*

Description

Performance Metric: Probability B > BMSY

Usage

```
P100(MSEobj = NULL)
```

Arguments

MSEobj	An object of class MSE

Value

An object of class PMobj

Examples

```
## Not run:
P100(myMSE)

## End(Not run)
```

P50 *Performance Metric: Probability B > 0.5 BMSY*

Description

Performance Metric: Probability B > 0.5 BMSY

Usage

```
P50(MSEobj = NULL)
```

Arguments

MSEobj	An object of class MSE

Value

An object of class PMobj

Examples

```
## Not run:
P50(myMSE)

## End(Not run)
```

Perfect_Imp *Perfect_Imp Imp*

Description

An object of class Imp

Usage

```
Perfect_Imp
```

Format

An object of class Imp of length 1.

Perfect_Info *Perfect_Info Obs*

Description

An object of class Obs

Usage

```
Perfect_Info
```

Format

An object of class `Obs` of length 1.

plot,Data,ANY-method *Plot Data object*

Description

Plot Data object

Usage

```
## S4 method for signature 'Data,ANY'
plot(x, funcs = NA, maxlines = 6, perc = 0.5,
  xlims = NA)
```

Arguments

`x`	object of class Data
`funcs`	MPs
`maxlines`	maximum number of lines
`perc`	percentile of TAC recommendation
`xlims`	limits of x-axis

plot,MSE,ANY-method *Plot MSE object*

Description

Plot MSE object

Usage

```
## S4 method for signature 'MSE,ANY'
plot(x)
```

Arguments

x	object of class MSE

plot.OM *Plot the operating model (OM) object parameters*

Description

A function that plots the parameters and resulting time series of an operating model.

Usage

```
## S3 method for class 'OM'
plot(x, ...)
```

Arguments

x	An object of class OM or a list with historical simulation information (ie runMSE(OM, Hist=TRUE))
...	Optional additional arguments passed to plot

Author(s)

T. Carruthers

plotFleet *Plot the Fleet object parameters*

Description

Plot the Fleet object parameters

Usage

```
plotFleet(x, Stock = NULL, nsamp = 3, nsim = 500, proyears = 28,
  col = "darkgray", breaks = 10, lwd = 2, ...)
```

Arguments

x	An object of class Fleet (or of class OM)
Stock	An object of class Stock
nsamp	Number of random samples for time-series plots
nsim	Number of iterations for histograms
proyears	Number of projection years
col	Color of histograms
breaks	Number of breaks for histograms
lwd	line width
...	Optional additional arguments passed to plot

Author(s)

A. Hordyk

plotFun *Print out plotting functions*

Description

This function prints out the available plotting functions for objects of class MSE or Data

Usage

```
plotFun(class = c('MSE', 'Data'), msg=TRUE)
```

Arguments

class	Character string. Prints out the plotting functions for objects of this class.
msg	Logical. Should the functions be printed to screen?

Note

Basically the function looks for any functions in the DLMtool that have the word ‘plot‘ in them. There is a chance that some plotting functions are missed. Let us know if you find any and we will add them.

Author(s)

A. Hordyk

`plotImp` *Plot the Implementation object parameters*

Description

A function that plots histograms of samples from the implementation object parameters, and time-series plots of ‘nsamp‘ samples of time-series examples. Used to visually examine the parameter values and ranges entered into the Obs object.

Usage

```
plotImp(x, nsim = 500, nyears = 50, col = "darkgray", breaks = 10, ...)
```

Arguments

`x`	An object of class Imp (or of class OM)
`nsim`	Number of iterations for histograms
`nyears`	Number of historical years
`col`	Color of histograms
`breaks`	Number of breaks for histograms
`...`	Optional additional arguments passed to `plot`

Author(s)

T. Carruthers and A. Hordyk

plotM *Plot M-at-Age and Size*

Description

Plot M-at-Age and Size

Usage

```
plotM(Stock, nsim = 5)
```

Arguments

Stock	An object of class 'Stock' or 'OM'
nsim	The number of simulations to plot

Author(s)

A. Hordyk

Examples

```
plotM(Albacore)
```

plotObs *Plot the Observation object parameters*

Description

A function that plots histograms of samples from the observation object parameters, and time-series plots of 'nsamp' samples of time-series examples. Used to visually examine the parameter values and ranges entered into the Obs object.

Usage

```
plotObs(x, nsim = 500, nyears = 50, col = "darkgray", breaks = 10, ...)
```

Arguments

x	An object of class Obs (or of class OM)
nsim	Number of iterations for histograms
nyears	Number of historical years
col	Color of histograms
breaks	Number of breaks for histograms
...	Optional additional arguments passed to plot

Author(s)

T. Carruthers and A. Hordyk

---

plotOFL *A generic OFL plot for NOAA use*

Description

As title.

Usage

```
plotOFL(Data,xlims=NA,perc=0.5)
```

Arguments

Data	An object of class Data that has been run though TAC()
xlims	x axis limits
perc	The percentile of the OFL distribution to be plotted

Value

A table of performance metrics.

Author(s)

T. Carruthers

---

plotSelect *Plot the vulnerability and retention curves*

Description

Plot the vulnerability and retention curves

Usage

```
plotSelect(OM, Pars = NULL, pyears = 4, sim = NA, type = "l")
```

Arguments

OM	An object of class 'OM'
Pars	Named list of sampled parameters
pyears	number of years to plot
sim	the simulation to plot. default is NA to plot a random simulation. Set to 1 for reproducible plot
type	plot type - line "l", point "p", or both "b"

Author(s)

A. Hordyk

`plotStock` *Plot the Stock object parameters*

Description

A function that plots histograms of samples from the Stock object parameters, and time-series plots of 'nsamp' samples of time-varying parameters. Used to visually examine the parameter values and ranges entered into the Stock object.

Usage

```
plotStock(x, nsamp = 3, nsim = 500, nyears = 50, proyears = 28,
  col = "darkgray", breaks = 10, lwd = 2, ask = FALSE, incVB = TRUE,
  ...)
```

Arguments

`x`	An object of class Stock (or of class OM)
`nsamp`	Number of random samples for time-series plots
`nsim`	Number of iterations for histograms. Ignored if x is class 'OM'
`nyears`	Number of historical years. Ignored if x is class 'OM'
`proyears`	Number of projection years. Ignored if x is class 'OM'
`col`	Color of histograms
`breaks`	Number of breaks for histograms
`lwd`	line width
`ask`	Ask before displaying next page?
`incVB`	Show the sampled von Bertalanffy growth curves on second page?
`...`	Optional additional arguments passed to `plot`

Author(s)

A. Hordyk

PMobj-class *An object for storing data for analysis using data-limited methods*

Description

Used interally

Slots

`name` Name of the Performance Metric. Character

`caption` A caption to be used in plots. Character, call, or function.

`Stat` Statistic of interest for the PM. Dimensions: nsim, nMP, yrs. Array

`Prob` Probability (mean over years) Dimensions: nsim by MP. Matrix, numeric or data.frame

`Mean` Mean probability (mean over years and simulations). Numeric. Length nMPs

`MPs` Name of MPs. Single value. Character string

Objects from the Class

Objects can be created by calls of the form `new('PMobj')`

Author(s)

A. Hordyk

POF *Performance Metric: Probability F < FMSY*

Description

Performance Metric: Probability F < FMSY

Usage

```
POF(MSEobj = NULL)
```

Arguments

`MSEobj` An object of class MSE

Value

An object of class PMobj

Examples

```
## Not run:
POF(myMSE)

## End(Not run)
```

---

popdyn *Population dynamics model*

Description

Population dynamics model

Usage

```
popdyn(nareas, maxage, Ncurr, pyears, M_age, Asize_c, MatAge, WtAge, Vuln, Retc,
  Prec, movc, SRrelc, Effind, Spat_targc, hc, R0c, SSBpRc, aRc, bRc, Qc,
  Fapic = NULL, maxF, control = 1)
```

Arguments

nareas	Integer. The number of spatial areas
maxage	Integer. The maximum age
Ncurr	Numeric matrix (dimensions maxage, nareas) with the current N-at-age
pyears	Integer. Number of years to project the model forward
M_age	Numeric matrix (dimensions maxage, pyears) with natural mortality at age
Asize_c	Numeric vector (length nareas) with size of each area
MatAge	Numeric matrix (dimensions maxage, nyears+proyears) with proportion mature for each age-class
WtAge	Numeric matrix (dimensions maxage, pyears) with weight-at-age
Vuln	Numeric matrix (dimensions maxage, pyears) with proportion vulnerable-at-age
Retc	Numeric matrix (dimensions maxage, pyears) with proportion retained-at-age
Prec	Numeric vector (length pyears) with recruitment error
movc	Numeric matrix (dimensions nareas, nareas) with movement matrix
SRrelc	Integer. Stock-recruitment curve
Effind	Numeric vector (length pyears) with the fishing effort by year
Spat_targc	Integer. Value of spatial targetting
hc	Numeric. Steepness of stock-recruit relationship
R0c	Numeric vector of length nareas with unfished recruitment by area
SSBpRc	Numeric vector of length nareas with unfished spawning per recruit by area
aRc	Numeric. Ricker SRR a value

bRc	Numeric. Ricker SRR b value
Qc	Numeric. Catchability coefficient
Fapic	Numeric. Apical F value
maxF	A numeric value specifying the maximum fishing mortality for any single age class
control	Integer. 1 to use q and effort to calculate F, 2 to use Fapic (apical F) and vulnerablity to calculate F.

Value

A named list of length 8 containing with arrays (dimensions: maxage, pyears, nareas) containing numbers-at-age, biomass-at-age, spawning stock numbers, spawning biomass, vulnerable biomass, fishing mortality, retained fishing mortality, and total mortality

Author(s)

A. Hordyk

Porgy *Porgy Stock*

Description

An object of class Stock

Usage

```
Porgy
```

Format

An object of class Stock of length 1.

Pplot *A projection by projection plot of F/FMSY and B/BMSY*

Description

A shorter version of the plot method for MSEs that just shows the projected trends in stock status and over exploitation

Usage

```
Pplot(MSEobj, nam=NA)
```

Arguments

MSEobj	An object of class MSE
nam	Title of plot

Author(s)

T. Carruthers

Pplot2 *A projection by projection plot of F/FMSY, B/BMSY, B/B0, and yield*

Description

A projection by projection plot of F/FMSY, B/BMSY, B/B0, and yield

Usage

```
Pplot2(MSEobj, YVar = c("SSB_SSBMSY", "F_FMSY"), MPs = NA, sims = NULL,
  traj = c("all", "quant"), quants = c(0.1, 0.9), incquant = TRUE,
  quantcol = "lightgray", RefYield = c("lto", "curr"), LastYr = TRUE,
  maxMP = 6, alpha = 60, cex.axis = 1.35, cex.lab = 1.4, YLab = NULL,
  incMP = TRUE, MPcex = 1.4, incLeg = TRUE, cex.leg = 1.5,
  legPos = "topleft", yline = NULL, parOR = FALSE, xaxis = TRUE,
  yaxis = TRUE, oneIt = TRUE, ...)
```

Arguments

MSEobj	An object of class MSE
YVar	What to plot on the y-axis? Options are: c('SSB_SSB0','SSB_SSBMSY', 'F_FMSY', 'Yield')
MPs	Optional subset by MP
sims	Optional subset by simulation
traj	Plot all projections (all) or only quantiles (quant)
quants	Numeric vector of length 2 specifying the quantiles (e.g., 10th and 90th. Median is always included)
incquant	Logical. Include the quantiles or only plot median?
quantcol	Colour of the quantile polygon
RefYield	Should yield be relative to long-term optimum (lto) or last historical year (curr)
LastYr	Logical. Include the last historical year in the yield projections?
maxMP	Maximum number of MPs to plot
alpha	Alpha for transparency of lines
cex.axis	Size of axis text
cex.lab	Size of axis label

YLab	Optional label for y-axis
incMP	Logical. Include name of MP?
MPcex	Size of MP label
incLeg	Logical. Include a legend?
cex.leg	Size of legend text
legPos	Legend position
yline	Optional horizontal line
parOR	Logical to over-ride the par parameters
xaxis	Logical. Should x-axis labels be displayed?
yaxis	Logical. Should y-axis labels be displayed?
oneIt	Logical. Should one iteration be plotted on the quantile plot?
...	Additional arguments to be passed to plotting functions

Author(s)

T. Carruthers & A.Hordyk

Precise_Biased *Precise_Biased Obs*

Description

An object of class Obs

Usage

```
Precise_Biased
```

Format

An object of class Obs of length 1.

Precise_Unbiased *Precise_Unbiased Obs*

Description

An object of class Obs

Usage

```
Precise_Unbiased
```

Format

An object of class Obs of length 1.

prob.class-class *Prob class union for performance metric objects*

Description

Used internally. Nothing to see here!

PWhisker *Performance Whisker Plot*

Description

A NAFO / ICCAT / SSB style MSE performance whisker plot

Usage

```
PWhisker(MSEobj)
```

Arguments

MSEobj	An object of class MSE

Value

A box plot of performance

Author(s)

T. Carruthers

Range *Standardize values*

Description

Function to standardize to value relative to minimum and maximum values

Usage

```
Range(x, Max, Min)

range01(x)
```

Arguments

x	vector of values
Max	Maximum value
Min	Minimum value

Rcontrol *Harvest Control Rule using prior for intrinsic rate of increase*

Description

An MP proposed by Carl Walters that modifies TACs according to trends in apparent surplus production that includes information from a demographically derived prior for intrinsic rate of increase

Usage

```
Rcontrol(x, Data, reps = 100, yrsmth = 10, gg = 2, glim = c(0.5,
2))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of quota samples
yrsmth	The number of years for smoothing catch and biomass data
gg	A gain parameters
glim	Limits for the change in TAC among years

Author(s)

C. Walters and T. Carruthers

References

Made-up for this package.

Rcontrol2 — *MP using prior for intrinsic rate of increase with a quadratic approximation to surplus production*

Description

An MP proposed by Carl Walters that modifies quotas according to trends in apparent surplus production that includes information from a demographically derived prior for intrinsic rate of increase. This is different from Rcontrol because it includes a quadratic approximation of recent trend in surplus production given biomass

Usage

```
Rcontrol2(x, Data, reps = 100, yrsmth = 10, gg = 2, glim = c(0.5,
2))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of TAC samples
yrsmth	The number of years for smoothing catch and biomass data
gg	A gain parameters
glim	Limits for the change in TAC among years

Author(s)

C. Walters and T. Carruthers

References

Made-up for this package.

read.control.file — *Reads iSCAM control file*

Description

A function for returning the results of the iscam control file

Usage

```
read.control.file(file = NULL, num.gears = NULL, num.age.gears = NULL,
  verbose = FALSE)
```

Arguments

file	File location
num.gears	The number of gears
num.age.gears	The number age-gears
verbose	should detailed results be printed to console

Author(s)

Chris Grandin (DFO PBS)

---

read.data.file *Reads iSCAM dat file*

Description

A function for returning the results of the .dat iscam file

Usage

```
read.data.file(file = NULL, verbose = FALSE)
```

Arguments

file	File location
verbose	should detailed results be printed to console

Author(s)

Chris Grandin (DFO PBS)

---

read.mcmc *Reads iSCAM mcmc output files*

Description

A function for returning the results of the iscam mcmc files

Usage

```
read.mcmc(model.dir = NULL, verbose = TRUE)
```

Arguments

model.dir	Folder name
verbose	should detailed results be printed to console

Author(s)

Chris Grandin (DFO PBS)

---

read.par.file *Reads iSCAM parameter file*

Description

A function for returning the results of the iscam .par file

Usage

```
read.par.file(file = NULL, verbose = FALSE)
```

Arguments

file	File location
verbose	should detailed results be printed to console

Author(s)

Chris Grandin (DFO PBS)

---

read.projection.file *Reads iSCAM projection file*

Description

A function for returning the results of the iscam projection file

Usage

```
read.projection.file(file = NULL, verbose = FALSE)
```

Arguments

file	File location
verbose	should detailed results be printed to console

Author(s)

Chris Grandin (DFO PBS)

read.report.file *Reads iSCAM Rep file*

Description

A function for returning the results of the .rep iscam file

Usage

```
read.report.file(fn)
```

Arguments

fn	File location

Author(s)

Chris Grandin (DFO PBS)

Red_snapper *Red_snapper Data*

Description

An object of class Data

Usage

```
Red_snapper
```

Format

An object of class `Data` of length 1.

Replace — *Replace an existing Stock, Fleet, Obs, or Imp object*

Description

A function that replaces a Stock, Fleet, Obs, or Imp object from an OM with one from another OM. Mainly used for internal functions.

Usage

```
Replace(OM, from, Sub = c("Stock", "Fleet", "Obs", "Imp"))
```

Arguments

OM	An operating model object (class OM) which will be updated with a sub-model from another OM
from	The OM object from which the sub-model is being taken
Sub	A character string specifying what object type to replace "Stock", "Fleet", "Obs" or "Imp" (default is all four which is probably not what you want to do)

Value

An object of class OM

Author(s)

A. Hordyk

replic8 — *Enlarge (replicate) a DLM data object to create an additional dimension for simulation / sensitivity testing*

Description

Replicates position 1 data to multiple positions for sensitivity testing etc

Usage

```
replic8(Data, nrep)
```

Arguments

Data	A data-limited methods data object
nrep	The number of positions to expand the DLM object to

Author(s)

T. Carruthers

Required *What methods need what data*

Description

A function that finds all methods in the environment and searches the function text for slots in the DLM data object

Usage

```
Required(funcs = NA)
```

Arguments

funcs	A character vector of possible methods of class DLM quota, DLM space or DLM size

Author(s)

T. Carruthers

Rockfish *Rockfish Stock*

Description

An object of class Stock

Usage

```
Rockfish
```

Format

An object of class `Stock` of length 1.

runInMP *Runs input control MPs on a Data object.*

Description

Function runs a MP (or MPs) of class 'Input' and returns a list: input control recommendation(s) in element 1 and Data object in element 2.

Usage

```
runInMP(Data, MPs = NA, reps = 100)
```

Arguments

Data	A object of class Data
MPs	A vector of MPs of class 'Input'
reps	Number of stochastic repititions - often not used in input control MPs.

Author(s)

A. Hordyk

runMSE *Run a Management Strategy Evaluation*

Description

A function that runs a Management Strategy Evaluation (closed-loop simulation) for a specified operating model

Usage

```
runMSE(OM = DLMtool::testOM, MPs = c("AvC", "DCAC", "FMSYref", "curE",
  "matlenlim", "MRreal"), nsim = 48, proyears = 50, interval = 4,
  pstar = 0.5, maxF = 0.8, reps = 1, CheckMPs = FALSE, timelimit = 1,
  Hist = FALSE, ntrials = 50, fracD = 0.05, CalcBlow = FALSE, HZN = 2,
  Bfrac = 0.5)
```

Arguments

OM	An operating model object (class 'OM')
MPs	A vector of methods (character string) of class Output or Input.
nsim	Number of simulations. Note that in DLMtool V4.1+ 'nsim is ignored if OM object contains the slot 'nsim'.
proyears	Number of projected years. Note that in DLMtool V4.1+ 'proyears is ignored if OM object contains the slot 'proyears'.
interval	The assessment interval - how often would you like to update the management system? NOTE: since DLMtool V4.5 this slot is included in the OM object which will override the value used here. This slot to be deprecated in the future.
pstar	The percentile of the sample of the management recommendation for each method. NOTE: since DLMtool V4.5 this slot is included in the OM object which will override the value used here. This slot to be deprecated in the future.
maxF	Maximum instantaneous fishing mortality rate that may be simulated for any given age class. NOTE: since DLMtool V4.5 this slot is included in the OM object which will override the value used here. This slot to be deprecated in the future.
reps	Number of samples of the management recommendation for each method. Note that when this is set to 1, the mean value of the data inputs is used. NOTE: since DLMtool V4.5 this slot is included in the OM object which will override the value used here. This slot to be deprecated in the future.
CheckMPs	Logical to indicate if Can function should be used to check if MPs can be run.
timelimit	Maximum time taken for a method to carry out 10 reps (methods are ignored that take longer)
Hist	Should model stop after historical simulations? Returns a list containing all historical data
ntrials	Maximum of times depletion and recruitment deviations are resampled to optimize for depletion. After this the model stops if more than percent of simulations are not close to the required depletion
fracD	maximum allowed proportion of simulations where depletion is not close to sampled depletion from OM before model stops with error
CalcBlow	Should low biomass be calculated where this is the spawning biomass at which it takes HZN mean generation times of zero fishing to reach Bfrac fraction of SSBMSY
HZN	The number of mean generation times required to reach Bfrac SSBMSY in the Blow calculation
Bfrac	The target fraction of SSBMSY for calculating Blow

Value

An object of class MSE

Author(s)

T. Carruthers and A. Hordyk

runMSErobust *Run a Management Strategy Evaluation*

Description

Run a Management Strategy Evaluation and save out the results to a Rdata file. To increase speed and efficiency, particulary for runs with a large number simulations (nsim), the simulations are split into a number of packets. The functions loops over the packets and combines the output into a single MSE object. If the MSE model crashes during a run, the MSE is run again until it is successfully completed. The MSE is stopped if the number of consecutive crashes exceeds maxCrash. There is an ption to save the packets as Rdata files to the current working directory (default is FALSE). By default, the functions saves the completed MSE object as a Rdata file (to the current working directory).

Usage

```
runMSErobust(OM = DLMtool::testOM, MPs = c("AvC", "DCAC", "FMSYref", "curE",
  "matlenlim", "MRreal"), nsim = 256, proyears = 50, interval = 4,
  pstar = 0.5, maxF = 0.8, timelimit = 1, reps = 1, CheckMPs = FALSE,
  Hist = FALSE, ntrials = 50, fracD = 0.05, CalcBlow = FALSE, HZN = 2,
  Bfrac = 0.5, maxsims = 64, name = NULL, unique = FALSE,
  maxCrash = 10, saveMSE = TRUE, savePack = FALSE)
```

Arguments

OM	An operating model object (class OM)
MPs	A vector of methods (character string) of class Output or Input.
nsim	Number of simulations
proyears	Number of projected years
interval	The assessment interval - how often would you like to update the management system?
pstar	The percentile of the sample of the management recommendation for each method
maxF	Maximum instantaneous fishing mortality rate that may be simulated for any given age class
timelimit	Maximum time taken for a method to carry out 10 reps (methods are ignored that take longer)
reps	Number of samples of the management recommendation for each method. Note that when this is set to 1, the mean value of the data inputs is used.
CheckMPs	Logical to indicate if Can function should be used to check if MPs can be run.
Hist	Should model stop after historical simulations? Returns a list containing all historical data
ntrials	Maximum of times depletion and recruitment deviations are resampled to optimize for depletion. After this the model stops if more than percent of simulations are not close to the required depletion

fracD	maximum allowed proportion of simulations where depletion is not close to sampled depletion from OM before model stops with error
CalcBlow	Should low biomass be calculated where this is the spawning biomass at which it takes HZN mean generation times of zero fishing to reach
HZN	The number of mean generation times required to reach Bfrac SSBMSY in the Blow calculation
Bfrac	fraction of SSBMSY
maxsims	Maximum number of simulations per packet
name	Character string for name of saved MSE packets (if savePack=TRUE) and final MSE object. If none provided, it uses the first five letters from the OM name
unique	Logical. Should the name be unique? Current date and time appended to name.
maxCrash	Maximum number of consecutive crashes before the MSE stops
saveMSE	Logical to indicate if final MSE object should be saved to current working directory (this is probably a good idea)
savePack	Logical to indicate if packets should be save to current working directory

Value

An object of class MSE

Author(s)

A. Hordyk and T. Carruthers

Sam *Conduct stock assessment*

Description

A wrapper function that gets the OFL recommendation in cases where a method of DLM quota has been specified

Usage

```
Sam(Data, MPs = NA, reps = 100, perc = 0.5)
```

Arguments

Data	A data-limited methods data object
MPs	A character vector of methods of DLM quota, DLM space or DLM size
reps	The number of samples of quota recommendations by method
perc	quantile of the recommendation to use

Author(s)

T. Carruthers

SampleCpars *Sample custom pars*

Description

Sample custom pars

Usage

```
SampleCpars(cpars, nsim = 48, msg = TRUE)
```

Arguments

cpars	A named list containing custom parameters for the OM
nsim	number of simulations
msg	logical - print the names of the cpars? Turn off when using the function in a loop

Value

A named list of sampled custom parameters

SampleFleetPars *Sample Fleet Parameters*

Description

Sample Fleet Parameters

Usage

```
SampleFleetPars(Fleet, Stock = NULL, nsim = NULL, nyears = NULL,
  proyears = NULL, cpars = NULL)
```

Arguments

Fleet	An object of class 'Fleet' or class 'OM'
Stock	An object of class 'Stock' or a list of sampled Stock parameters. Ignored if 'Fleet' is class 'OM'
nsim	Number of simulations. Ignored if 'Fleet' is class 'OM'
nyears	Number of historical years. Ignored if 'Fleet' is class 'OM'
proyears	Number of projection years. Ignored if 'Fleet' is class 'OM'
cpars	Optional named list of custom parameters. Ignored if 'Fleet' is class 'OM'

Value

A named list of sampled Fleet parameters

SampleImpPars *Sample Implementation Error Parameters*

Description

Sample Implementation Error Parameters

Usage

```
SampleImpPars(Imp, nsim = NULL)
```

Arguments

Imp	An object of class 'Imp' or class 'OM'
nsim	Number of simulations. Ignored if 'Stock' is class 'OM'

Value

A named list of sampled Implementation Error parameters

SampleObsPars *Sample Observation Parameters*

Description

Sample Observation Parameters

Usage

```
SampleObsPars(Obs, nsim = NULL)
```

Arguments

Obs	An object of class 'Obs' or class 'OM'
nsim	Number of simulations. Ignored if 'Obs' is class 'OM'

Value

A named list of sampled Observation parameters

SampleStockPars *Sample Stock parameters*

Description

Sample Stock parameters

Usage

```
SampleStockPars(Stock, nsim = 48, nyears = 80, proyears = 50,
  cpars = NULL, Msg = TRUE)
```

Arguments

Stock	An object of class 'Stock' or class 'OM'
nsim	Number of simulations. Ignored if 'Stock' is class 'OM'
nyears	Number of historical years. Ignored if 'Stock' is class 'OM'
proyears	Number of projection years. Ignored if 'Stock' is class 'OM'
cpars	Optional named list of custom parameters. Ignored if 'Stock' is class 'OM'
Msg	logical. Warning message for M values?

Value

A named list of sampled Stock parameters

sample_steepness2 *Sample steepness given mean and cv*

Description

Sample steepness given mean and cv

Usage

```
sample_steepness2(n, mu, cv)
```

Arguments

n	number of samples
mu	mean h
cv	cv of h

Author(s)

Q. Huynh

sampy *Sample vector*

Description

Sample vector

Usage

```
sampy(x)
```

Arguments

x	vector of values

SBT1 *SBT simple MP*

Description

An MP that makes incremental adjustments to TAC recommendations based on the apparent trend in CPUE

Usage

```
SBT1(x, Data, reps = 100, yrsmth=10, k1=1.5, k2=3, gamma=1)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of samples of the TAC
yrsmth	The number of years for evaluating trend in relative abundance indices
k1	Control parameter
k2	Control parameter
gamma	Control parameter

Details

This isn't exactly the same as the proposed methods and is stochastic in this implementation. The method doesn't tend to work too well under many circumstances possibly due to the lack of 'tuning' that occurs in the real SBT assessment environment. You could try asking Rich Hillary at CSIRO about this approach.

Author(s)

T. Carruthers

References

http://www.ccsbt.org/site/recent_assessment.php

SBT2 *SBT complex MP*

Description

An MP that makes incremental adjustments to TAC recommendations based on index levels relative to target levels (BMSY/B0) and catch levels relative to target levels (MSY)

Usage

```
SBT2(x, Data, reps = 100,
epsB=0.25,epsR=0.75,tauR=5,tauB=7,gamma=1)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of samples of the TAC
epsB	Control parameter
epsR	Control parameter
tauR	Control parameter
tauB	Control parameter
gamma	Control parameter

Details

This isn't exactly the same as the proposed methods and is stochastic in this implementation. The method doesn't tend to work too well under many circumstances possibly due to the lack of 'tuning' that occurs in the real SBT assessment environment. You could try asking Rich Hillary at CSIRO about this approach.

Author(s)

T. Carruthers

References

http://www.ccsbt.org/site/recent_assessment.php

sdconv — *Get log normal standard deviation from transformed space mean and standard deviation*

Description

Get log normal standard deviation from transformed space mean and standard deviation

Usage

```
sdconv(m, sd)
```

Arguments

m	mean
sd	standard deviation

Value

numeric

Author(s)

T. Carruthers

Sense — *Sensitivity analysis*

Description

A function that determines the inputs for a given data-limited method of class Output and then analyses the sensitivity of TAC estimates to marginal differences in each input. The range used for sensitivity is based on the user-specified CV for that input (e.g. CV_Mort, Mort)

Usage

```
Sense(Data, MP, nsense = 6, reps = 100, perc = c(0.05, 0.5, 0.95), ploty = T)
```

Arguments

Data	A data-limited methods data object
MP	A character string representing an MP applied in calculating the TAC recommendations in the DLM object
nsense	The number of points over which to calculate the TAC (resolution)
reps	The number of samples of the quota taken for the calculation of the TAC
perc	The percentile of the sample TAC
ploty	A logical switch, (T/F, should a plot be drawn?)

Author(s)

T. Carruthers

`setup` *Setup parallel processing*

Description

Sets up parallel processing using the snowfall package

Usage

```
setup(cpus = parallel::detectCores())
```

Arguments

`cpus`	number of CPUs

`show,PMobj-method` *Show the output of a PM*

Description

Show the output of a PM

Usage

```
## S4 method for signature 'PMobj'
show(object)
```

Arguments

`object`	object of class MSE

`Simulation_1` *Simulation_1 Data*

Description

An object of class Data

Usage

```
Simulation_1
```

Format

An object of class `Data` of length 1.

simYears *Simulate population dynamics for historical years*

Description

Simulate population dynamics for historical years

Usage

```
simYears(x, nareas, maxage, N, pyears, M_ageArray, Asize, Mat_age, Wt_age, V,
  retA, Perr, mov, SRrel, Find, Spat_targ, hs, R0a, SSBpR, aR, bR, qs, maxF,
  useCPP = TRUE)
```

Arguments

x	Integer, the simulation number
nareas	The number of spatial areas
maxage	The maximum age
N	Array of the numbers-at-age in population. Dimensions are nsim, maxage, nyears, nareas. Only values from the first year (i.e N[,,1,]) are used, which is the current N-at-age.
pyears	The number of years to project forward. Equal to 'nyears' for optimizing for q.
M_ageArray	An array (dimensions nsim, maxage, nyears+proyears) with the natural mortality-at-age and year
Asize	A matrix (dimensions nsim, nareas) of size of areas
Mat_age	A matrix (dimensions nsim, maxage) with the proportion mature for each age-class
Wt_age	An array (dimensions nsim, maxage, nyears+proyears) with the weight-at-age and year
V	An array (dimensions nsim, maxage, nyears+proyears) with the vulnerability-at-age and year
retA	An array (dimensions nsim, maxage, nyears+proyears) with the probability retained-at-age and year
Perr	A matrix (dimensions nsim, nyears+proyears) with the recruitment deviations
mov	An array (dimensions nsim, nareas, nareas) with the movement matrix
SRrel	A numeric vector nsim long specifying the recruitment curve to use
Find	A matrix (dimensions nsim, nyears) with the historical fishing effort
Spat_targ	A numeric vector nsim long with the spatial targeting
hs	A numeric vector nsim long with the steepness values for each simulation
R0a	A matrix (dimensions nsim, nareas) with the unfished recruitment by area
SSBpR	A matrix (dimensions nsim, nareas) with the unfished spawning-per-recruit by area

aR	A numeric vector nsim long with the Ricker SRR a values
bR	A numeric vector nsim long with the Ricker SRR b values
qs	A numeric vector nsim long with catchability coefficients
maxF	A numeric value specifying the maximum fishing mortality for any single age class
useCPP	logical - use the CPP code? For testing purposes only

Author(s)

A. Hordyk

SketchFun *Manually map the historical relative fishing effort trajectory.*

Description

Internal function for innteractive plot which allows users to specify the relative trajectory and variability in the historical fishing effort.

Usage

```
SketchFun(nyears, Years=NULL)
```

Arguments

nyears	Number of years
Years	An optional vector of years. Should be nyears long.

Author(s)

A. Hordyk

---

slotlim *An data-limited method which sets a slot limit*

Description

An example of the implementation of input controls in the DLM toolkit, where selectivity-at-length is set using a slot limit; that is, a minimum and maximum legal length. The maximum limit is set here, quite arbitrarily, as the 75th percentile between the new minimum legal length and the estimated asymptotic length.

Usage

```
slotlim(x, Data, ...)
```

Arguments

x	A position in a data-limited methods object
Data	A data-limited methods object
...	Optional additional arguments that are ignored. Note arguments reps or ... are required for all input controls

Value

An object of class 'InputRec'

Author(s)

A. Hordyk

References

Made-up for this package

Snapper *Snapper Stock*

Description

An object of class Stock

Usage

```
Snapper
```

Format

An object of class Stock of length 1.

Sole *Sole Stock*

Description

An object of class Stock

Usage

```
Sole
```

Format

An object of class Stock of length 1.

Splot *Scatter plot of B/BMSY or B/B0 and F/FMSY for lastYrs*

Description

Scatter plot of B/BMSY or B/B0 and F/FMSY for lastYrs

Usage

```
Splot(MSEobj = NULL, MPs = NA, All = TRUE, Var = c('B_BMSY', 'SSB_SSB0'),
lastYrs = 10, Fref = 1, BMSYref = 1, B0ref = 0.4, cex.MP = 1, Fbg = FALSE,
Bbg = FALSE, Props = FALSE, TP = FALSE)
```

Arguments

MSEobj	An object of class MSE
MPs	Optional subset by MP
All	Logical. Plot all points or just the mean?
Var	What to plot on the y-axis: B_BMSY or SSB_SSB0
lastYrs	Last number of years in projection to calculate statistics
Fref	Location of F statistic reference line
BMSYref	Location of B_MSY statistic reference line
B0ref	Location of B_0 statistic reference line
cex.MP	size of MP label
Fbg	Logical. Include background colours for F-statistic?
Bbg	Logical. Include background colours for B-statistic?
Props	Logical. Display the proportion of points in each quadrant?
TP	Logical. Use transparent colours?

Author(s)

A. Hordyk

SPmod *Surplus production based catch-limit modifier*

Description

An MP that makes incremental adjustments to TAC recommendations based on the apparent trend in surplus production. Based on the theory of Mark Maunder (IATTC)

Usage

```
SPmod(x, Data, reps = 100, alp = c(0.8, 1.2), bet = c(0.8, 1.2))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of quota samples
alp	Condition for modifying the TAC (bounds on change in abundance)
bet	Limits for how much the TAC can change among years

Details

Note that this isn't exactly what Mark has previously suggested and is stochastic in this implementation.

Value

A numeric vector of TAC recommendations

Author(s)

T. Carruthers

References

http://www.iattc.org/Meetings/Meetings2014/MAYSAC/PDFs/SAC-05-10b-Management-Strategy-Evaluation.pdf

SPMSY *Catch trend Surplus Production MSY MP*

Description

An MP that uses Martell and Froese (2012) method for estimating MSY to determine the OFL. Since their approach estimates stock trajectories based on catches and a rule for intrinsic rate of increase it also returns depletion. Given their surplus production model predicts K, r and depletion it is straighforward to calculate the OFL based on the Schaefer productivity curve. OFL = dep x (1-dep) x r x K x 2

Usage

```
SPMSY(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of samples of the TAC

Details

Requires the assumption that catch is proportional to abundance. Occasionally the rule that limits r and K ranges does not allow r-K pairs to be found that lead to the depletion inferred by the catch trajectories. In this case this method widens the search.

Author(s)

T. Carruthers

References

Martell, S. and Froese, R. 2012. A simple method for estimating MSY from catch and resilience. Fish and Fisheries. DOI: 10.1111/j.1467-2979.2012.00485.x

SPslope *Slope in surplus production MP*

Description

A management procedure that makes incremental adjustments to TAC recommendations based on the apparent trend in recent surplus production. Based on the theory of Mark Maunder (IATTC)

Usage

```
SPslope(x, Data, reps = 100, yrsmth = 4, alp = c(0.9, 1.1), bet =
c(1.5, 0.9))
```

Arguments

x	A position in data-limited methods data object
Data	A data-limited methods data object
reps	The number of quota samples
yrsmth	Years over which to smooth recent estimates of surplus production
alp	Condition for modifying the Data (bounds on change in abundance)
bet	Limits for how much the Data can change among years

Details

Note that this isn't exactly what Mark has previously suggested and is stochastic in this implementation.

Value

A numeric vector of Data recommendations

Author(s)

T. Carruthers

References

http://www.iattc.org/Meetings/Meetings2014/MAYSAC/PDFs/SAC-05-10b-Management-Strategy-Evaluation.pdf

SPSRA *Surplus Production Stock Reduction Analysis*

Description

A surplus production equivalent of DB-SRA that uses a demographically derived prior for intrinsic rate of increase (McAllister method, below)

Usage

```
SPSRA(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object (class DLM)
reps	The number of samples of the TAC taken for the calculation of the quota

Author(s)

T. Carruthers

References

McAllister, M.K., Pikitch, E.K., and Babcock, E.A. 2001. Using demographic methods to construct Bayesian priors for the intrinsic rate of increase in the Schaefer model and implications for stock rebuilding. Can. J. Fish. Aquat. Sci. 58: 1871-1890.

SPSRA_ML *Surplus Production Stock Reduction Analysis using a mean-length estimate of current stock depletion*

Description

A surplus production equivalent of DB-SRA that uses a demographically derived prior for intrinsic rate of increase. A prior for depletion is calculated from a mean-length estimator

Usage

```
SPSRA_ML(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object (class DLM)
reps	The number of samples of the TAC taken

Note

The mean length extension was programmed by Gary Nelson as part of his excellent R package 'fishmethods'

Author(s)

T. Carruthers

References

McAllister, M.K., Pikitch, E.K., and Babcock, E.A. 2001. Using demographic methods to construct Bayesian priors for the intrinsic rate of increase in the Schaefer model and implications for stock rebuilding. Can. J. Fish. Aquat. Sci. 58: 1871-1890.

SRAcomp *Plot simulation test of Stochastic SRA method*

Description

Plots simulation variables versus estimation variables for Stochastic SRA methods of conditioning operating models.

Usage

```
SRAcomp(sim, OM, outfile = NA, maxplot = 10)
```

Arguments

sim	The output list object of SRAsim() function.
OM	The output object of StochasticSRA() function.
outfile	The name of the figure (something.jpg) you wish to make using SRAcomp
maxplot	The maximum number of simulations to plot

Author(s)

T. Carruthers (Canadian DFO grant)

Examples

```
## Not run:
sim<-SRAsim(testOM,qmult=1,patchy=0.8)
CAA<-sim$CAA
Chist<-sim$Chist
testOM<-StochasticSRA(testOM,CAA,Chist,nsim=30,nits=500)
SRAcomp(sim,testOM)

## End(Not run)
```

SRAsim *Simulates catch at age and catch history data for testing SRA methods*

Description

Catch at age and catch simulator.

Usage

```
SRAsim(OM, qmult = 0.5, CAApatchy = 0.4, Cpatchy = 1, Ipatchy = 0.4,
  MLpatchy = 0.4, nCAA = 100, nL = 200, sigmaE = 0.25, sigmaI = 0.1)
```

Arguments

OM	An operating model object with M, growth, stock-recruitment and maturity parameters specified.
qmult	Fraction of natural mortality rate that is mean fishing mortality (Fishing catchability multiplier)
CAApatchy	The fraction of years that have catch at age data
Cpatchy	The fraction of years that have catch data
Ipatchy	The fraction of years that have index data
MLpatchy	The fraction of years that have mean length data
nCAA	The number of independent annual catch at age observations (same among all years)
nL	The number of independent annual catch at length observations (same among all years) for calculating mean length
sigmaE	Level of simulated interannual variability in effort (F) expressed as a lognormal SD
sigmaI	Observation error in relative abundance indices expressed as a lognormal SD

Value

A list: Chist = historical catch series,Recdevs = historical recruitment deviations (mean = 1), CAA = catch at age matrix, N = numbers at age matrix, SSB = annual spawning biomass, FM = Fishing mortality rate at age matrix, M = natural mortality rate classy

Author(s)

T. Carruthers (Canadian DFO grant)

Examples

```
out<-SRAsim(testOM)
```

SRopt — *Function that returns a stochastic estimate of steepness given observed stock recruitment data*

Description

Function that returns a stochastic estimate of steepness given observed stock recruitment data

Usage

```
SRopt(nsim, SSB, rec, SSBpR, plot = F, type = "BH")
```

Arguments

nsim	number of samples of steepness to generate
SSB	'observations' of spawning biomass
rec	'observations' (model predictions) of recruitment
SSBpR	spawning stock biomass per recruit at unfished conditions
plot	should plots of model fit be produced?
type	what type of stock recruitment curve is being fitted BH = Beverton-Holt

Author(s)

T. Carruthers

SS2Data — *Reads data Stock Synthesis file structure into an data object using package r4ss*

Description

A function that uses the file location of a fitted SS3 model including input files to population the various slots of an data object

Usage

```
SS2Data(SSdir, Source = "No source provided", length_timestep = NA,
  Name = "", Author = "No author provided", printstats = F, verbose = T)
```

Arguments

SSdir	A folder with Stock Synthesis input and output files in it
Source	Reference to assessment documentation e.g. a url
length_timestep	The duration (in years) of each timestep in the model (if an quarterly model is used this is 0.25)
Name	The name of the operating model
Author	Who did the assessment
printstats	Should the r4ss function SS_output return info on data that was read in?
verbose	Should the r4ss function SS_ouput return detailed messages?

Author(s)

T. Carruthers

SS2DLM *Reads MLE estimates from Stock Synthesis file structure into an operating model using package r4ss*

Description

A function that uses the file location of a fitted SS3 model including input files to population the various slots of an operating model with MLE parameter estimates

Usage

```
SS2DLM(SSdir, nsim = 48, proyears = 50, length_timestep = NA,
  Name = NULL, Source = "No source provided",
  Author = "No author provided", printstats = F, verbose = T)
```

Arguments

SSdir	A folder with Stock Synthesis input and output files in it
nsim	The number of simulations to take for parameters with uncertainty (for OM@cpars custom parameters)
proyears	The number of projection years for MSE
length_timestep	The duration (in years) of each timestep in the model (if an quarterly model is used this is 0.25)
Name	The name of the operating model
Source	Reference to assessment documentation e.g. a url
Author	Who did the assessment
printstats	Should the r4ss function SS_output return info on data that was read in?
verbose	Should the r4ss function SS_ouput return detailed messages?

Author(s)

T. Carruthers

`StochasticSRA` *Stochastic SRA construction of operating models*

Description

Specify an operating model, using catch composition data and a historical catch series. Returns and operating model with depletion (D), selectivity parameters (L5, LFS) and effort trajectory (Effyears, EffLower, EffUpper) filled.

Usage

```
StochasticSRA(OM, CAA, Chist, Ind = NA, ML = NA, CAL = NA, mulen = NA,
  wts = c(1, 1, 0.5, 0.1, 1), Jump_fac = 1, nits = 4000, burnin = 500,
  thin = 10, ESS = 300, MLsd = 0.1, ploty = T, nplot = 6,
  SRAdir = NA)
```

Arguments

`OM`	An operating model object with M, growth, stock-recruitment and maturity parameters specified.
`CAA`	A matrix nyears (rows) by nages (columns) of catch at age (age 1 to maxage in length)
`Chist`	A vector of historical catch observations (nyears long) going back to unfished conditions
`Ind`	A vector of historical abundance index observations (assumed proportional to SSB)
`ML`	A vector of historical mean length (in catch) observations
`CAL`	A matrix of nyears (row) by n length bins (columns) of catch at length samples
`mulen`	A vector mean length by length bin, a vector the same as the number of columns of CAL
`wts`	A vector of relative weights for the likelihood functions of CAA, Chist, Ind, ML and CAL
`Jump_fac`	A multiplier of the jumping distribution variance to increase acceptance (lower Jump_fac) or decrease acceptance rate (higher Jump_fac)
`nits`	The number of MCMC iterations
`burnin`	The number of initial MCMC iterations to discard
`thin`	The interval over which MCMC samples are extracted for use in graphing / statistics
`ESS`	Effective sample size - the weighting of the catch at age data
`MLsd`	The lognormal sd of the mean length observations

ploty	Do you want to see diagnostics plotted?
nplot	how many MCMC samples should be plotted in convergence plots?
SRAdir	A directory where the SRA diagnostics / fit are stored

Value

A list with three positions. Position 1 is the filled OM object, position 2 is the custompars data.frame that may be submitted as an argument to runMSE() and position 3 is the matrix of effort histories [nyears x nsim] vector of objects of classclassy

Author(s)

T. Carruthers (Canadian DFO grant)

References

Walters, C.J., Martell, S.J.D., Korman, J. 2006. A stochastic approach to stock reduction analysis. Can. J. Fish. Aqua. Sci. 63:212-213.

Examples

```
## Not run:
setup()
sim<-SRAsim(testOM,patchy=0.8)
CAA<-sim$CAA
Chist<-sim$Chist
testOM<-StochasticSRA(testOM,CAA,Chist,nsim=30,nits=1000)
runMSE(testOM)

## End(Not run)
```

StochasticSRAcpp *Stochastic SRA construction of operating models*

Description

Specify an operating model, using catch composition data and a historical catch series. Returns and operating model with depletion (D), selectivity parameters (L5, LFS) and effort trajectory (Effyears, EffLower, EffUpper) filled. Modified version using cpp code.

Usage

```
StochasticSRAcpp(OM, CAA, Chist, Ind, Cobs = 0.1, sigmaR = 0.5,
  Umax = 0.9, nsim = 48, proyears = 50, Jump_fac = 1, nits = 20000,
  burnin = 1000, thin = 50, ESS = 300, ploty = T, nplot = 6,
  SRAdir = NA)
```

Arguments

OM	An operating model object with M, growth, stock-recruitment and maturity parameters specified.
CAA	A matrix nyears (rows) by nages (columns) of catch at age (age 1 to maxage in length)
Chist	A vector of historical catch observations (nyears long) going back to unfished conditions
Ind	A vector of historical index observations (nyears long, may be patchy with NAs) going back to unfished conditions.
Cobs	A numeric value representing catch observation error as a log normal sd
sigmaR	A numeric value representing the prior standard deviation of log space recruitment deviations
Umax	A numeric value representing the maximum harvest rate for any age class (rejection of sims where this occurs)
nsim	The number desired draws of parameters / effort trajectories
proyears	The number of projected MSE years
Jump_fac	A multiplier of the jumping distribution variance to increase acceptance (lower Jump_fac) or decrease acceptance rate (higher Jump_fac)
nits	The number of MCMC iterations
burnin	The number of initial MCMC iterations to discard
thin	The interval over which MCMC samples are extracted for use in graphing / statistics
ESS	Effective sample size - the weighting of the catch at age data
ploty	Do you want to see diagnostics plotted?
nplot	how many MCMC samples should be plotted in convergence plots?
SRAdir	A directory where the SRA diagnostics / fit are stored

Value

A list with three positions. Position 1 is the filled OM object, position 2 is the custompars data.frame that may be submitted as an argument to runMSE() and position 3 is the matrix of effort histories [nyears x nsim] vector of objects of classclassy

Author(s)

T. Carruthers (Canadian DFO grant)

References

Walters, C.J., Martell, S.J.D., Korman, J. 2006. A stochastic approach to stock reduction analysis. Can. J. Fish. Aqua. Sci. 63:212-213.

Examples

```
## Not run:
setup()
sim<-SRAsim(testOM,patchy=0.8)
CAA<-sim$CAA
Chist<-sim$Chist
testOM<-StochasticSRA(testOM,CAA,Chist,nsim=30,nits=1000)
runMSE(testOM)

## End(Not run)
```

Stock-class *Class* 'Stock'

Description

An operating model component that specifies the parameters of the population dynamics model

Slots

`Name` The name of the Stock object. Single value. Character string

`Species` Scientific name of the species. Genus and species name. Character string

`maxage` The maximum age of individuals that is simulated (there is no 'plus group'). Single value. Positive integer

`R0` The magnitude of unfished recruitment. Single value. Positive real number

`M` Natural mortality rate. Uniform distribution lower and upper bounds. Positive real number

`M2` (Optional) Natural mortality rate at age. Vector of length 'maxage'. Positive real number

`Mexp` Exponent of the Lorenzen function assuming an inverse relationship between M and weight. Uniform distribution lower and upper bounds. Real numbers <= 0.

`Msd` Inter-annual variability in natural mortality rate expressed as a coefficient of variation. Uniform distribution lower and upper bounds. Non-negative real numbers

`Mgrad` Mean temporal trend in natural mortality rate, expressed as a percentage change in M per year. Uniform distribution lower and upper bounds. Real numbers

`h` Steepness of the stock recruit relationship. Uniform distribution lower and upper bounds. Values from 1/5 to 1

`SRrel` Type of stock-recruit relationship. Single value, switch (1) Beverton-Holt (2) Ricker. Integer

`Perr` Process error, the CV of lognormal recruitment deviations. Uniform distribution lower and upper bounds. Non-negative real numbers

`AC` Autocorrelation in recruitment deviations rec(t)=AC*rec(t-1)+(1-AC)*sigma(t). Uniform distribution lower and upper bounds. Non-negative real numbers

`Period` (Optional) Period for cyclical recruitment pattern in years. Uniform distribution lower and upper bounds. Non-negative real numbers

`Amplitude` (Optional) Amplitude in deviation from long-term average recruitment during recruitment cycle (eg a range from 0 to 1 means recruitment decreases or increases by up to 100% each cycle). Uniform distribution lower and upper bounds. 0 < Amplitude < 1

`Linf` Maximum length. Uniform distribution lower and upper bounds. Positive real numbers

`K` von Bertalanffy growth parameter k. Uniform distribution lower and upper bounds. Positive real numbers

`t0` von Bertalanffy theoretical age at length zero. Uniform distribution lower and upper bounds. Non-positive real numbers

`LenCV` Coefficient of variation of length-at-age (assumed constant for all age classes). Uniform distribution lower and upper bounds. Positive real numbers

`Ksd` Inter-annual variability in growth parameter k. Uniform distribution lower and upper bounds. Non-negative real numbers

`Kgrad` Mean temporal trend in growth parameter k, expressed as a percentage change in k per year. Uniform distribution lower and upper bounds. Real numbers

`Linfsd` Inter-annual variability in maximum length. Uniform distribution lower and upper bounds. Non-negative real numbers

`Linfgrad` Mean temporal trend in maximum length, expressed as a percentage change in Linf per year. Uniform distribution lower and upper bounds. Real numbers

`L50` Length at 50 percent maturity. Uniform distribution lower and upper bounds. Positive real numbers

`L50_95` Length increment from 50 percent to 95 percent maturity. Uniform distribution lower and upper bounds. Positive real numbers

`D` Current level of stock depletion SSB(current)/SSB(unfished). Uniform distribution lower and upper bounds. Fraction

`a` Length-weight parameter alpha. Single value. Positive real number

`b` Length-weight parameter beta. Single value. Positive real number

`Size_area_1` The size of area 1 relative to area 2. Uniform distribution lower and upper bounds. Positive real numbers

`Frac_area_1` The fraction of the unfished biomass in stock 1. Uniform distribution lower and upper bounds. Positive real numbers

`Prob_staying` The probability of inviduals in area 1 remaining in area 1 over the course of one year. Uniform distribution lower and upper bounds. Positive fraction.

`Fdisc` Fraction of discarded fish that die. Uniform distribution lower and upper bounds. Non-negative real numbers

`Source` A reference to a website or article from which parameters were taken to define the stock object. Single value. Character string.

Objects from the Class

Objects can be created by calls of the form `new('Stock')`

Author(s)

T. Carruthers and A. Hordyk

Examples

```
showClass('Stock')
```

StockDescription *StockDescription*

Description

A data.frame with description of slots for class Stock

Usage

```
StockDescription
```

Format

An object of class `data.frame` with 33 rows and 2 columns.

STY *Performance Metric: Probability Short-Term Yield > 0.5 Relative Yield*

Description

Performance Metric: Probability Short-Term Yield > 0.5 Relative Yield

Usage

```
STY(MSEobj = NULL)
```

Arguments

MSEobj	An object of class MSE

Value

An object of class PMobj

Examples

```
## Not run:
STY(myMSE)

## End(Not run)
```

Sub *Subset MSE object by management procedure (MP) or simulation.*

Description

Subset the MSE object by particular MPs (either MP number or name), or particular simulations, or a subset of the projection years (e.g., 1: < projection years).

Usage

```
Sub(MSEobj, MPs=NULL, sims=NULL, years=NULL)
```

Arguments

MSEobj	A MSE object.
MPs	A vector MPs names or MP numbers to subset the MSE object. Defaults to all MPs.
sims	A vector of simulation numbers to subset the MSE object. Can also be a logical vector. Defaults to all simulations.
years	A numeric vector of projection years. Should start at 1 and increase by one to some value equal or less than the total number of projection years.

Author(s)

A. Hordyk

SubCpars *Subset an OM cpars slot*

Description

Subset the custom parameters of an operating model

Usage

```
SubCpars(OM, sims)
```

Arguments

OM	An object of class OM
sims	A logical vector OM@nsim long of simulations to either retain (TRUE) or remove (FALSE)

Value

An object of class OM

Author(s)

T. Carruthers

`SubOM` *Subset a Stock, Fleet, Obs, or Imp object from an OM object*

Description

A function that strips out a Stock, Fleet, Obs, or Imp object from a complete OM object. Mainly used for internal functions.

Usage

```
SubOM(OM, Sub = c("Stock", "Fleet", "Obs", "Imp"))
```

Arguments

`OM`	An operating model object (class OM)
`Sub`	A character string specifying what object type to strip out "Stock", "Fleet", "Obs", or "Imp"

Value

An object of class Stock, Fleet, Obs, or Imp

Author(s)

A. Hordyk

`summary,Data-method` *Summary of Data object*

Description

Summary of Data object

Usage

```
## S4 method for signature 'Data'
summary(object)
```

Arguments

`object`	object of class Data

summary,MSE-method *Summary of MSE object*

Description

Summary of MSE object

Usage

```
## S4 method for signature 'MSE'
summary(object, ..., silent = FALSE)
```

Arguments

object	object of class MSE
...	a list of names of PM methods
silent	Should summary be printed to console? Logical.

TAC *Calculate TAC recommendations for more than one MP*

Description

A function that returns the stochastic TAC recommendations from a vector of data-limited MPs (Output) given a data-limited data object Data

Usage

```
TAC(Data, MPs = NA, reps = 100, timelimit = 1)
```

Arguments

Data	A data-limited methods data object
MPs	optional vector of MP names
reps	Number of repititions
timelimit	The maximum time (seconds) taken to complete 10 reps

Author(s)

T. Carruthers

TACfilter *TAC Filter*

Description

Filters vector of TAC recommendations by replacing negatives with NA and and values beyond five standard deviations from the mean as NA

Usage

```
TACfilter(TAC)
```

Arguments

TAC	A numeric vector of TAC recommendations

Author(s)

T. Carruthers

Targeting_Small_Fish *Targeting_Small_Fish Fleet*

Description

An object of class Fleet

Usage

```
Targeting_Small_Fish
```

Format

An object of class `Fleet` of length 1.

Target_All_Fish *Target_All_Fish Fleet*

Description

An object of class Fleet

Usage

```
Target_All_Fish
```

Format

An object of class `Fleet` of length 1.

tdlnorm *Calculate density of log-normally distributed random numbers*

Description

Calculate density of log-normally distributed random numbers

Usage

```
tdlnorm(x, mu, cv)
```

Arguments

x	vector
mu	mean
cv	coefficient of variation

Value

numeric

Author(s)

T. Carruthers

testOM *testOM OM*

Description

An object of class OM

Usage

```
testOM
```

Format

An object of class `OM` of length 1.

Toothfish *Toothfish Stock*

Description

An object of class Stock

Usage

```
Toothfish
```

Format

An object of class `Stock` of length 1.

Tplot *A trade-off plot for an MSE object*

Description

A shorter version of the plot method for MSEs that just shows the overall trade-offs

Usage

```
Tplot(MSEobj,nam=NA)
```

Arguments

`MSEobj`	An object of class 'MSE'
`nam`	Name of the plot

Author(s)

T. Carruthers

`Tplot2` *A shorter version of the plot method for MSEs that just shows the overall trade-offs*

Description

A trade-off plot for an MSE object that compares long-term yield (LTY: fraction of simulations getting over half FMSY yield in the last ten years of the projection), short-term yield (STY: fraction of simulations getting over half FMSY yield in the first ten years of the projection), variability in yield (VY: fraction of simulations where average annual variability in yield is less than 10 per cent) and biomass level (B10: the fraction of simulations in which biomass stays above 10 percent of BMSY).

Usage

```
Tplot2(MSEobj,nam=NA)
```

Arguments

`MSEobj`	An object of class 'MSE'
`nam`	Name of the plot

Author(s)

T. Carruthers

`Tplot3` *Test Trade-Off Plot*

Description

Test Trade-Off Plot

Usage

```
Tplot3(MSEobj, ..., lims = c(0.2, 0.2, 0.8, 0.8))
```

Arguments

`MSEobj`	An object of class MSE
`...`	Names of PM methods to plot
`lims`	Numeric vector of satisficing limits. Recycled to number of PM methods

Value

produces a plot

Author(s)

A. Hordyk

Examples

```
## Not run:
 Tplot3{myMSE}

## End(Not run)
```

TradePlot *Generic Trade-off Plot*

Description

Creates a trade-off plot (up to four panels) of built-in performance metrics.

Usage

```
TradePlot(MSEobj, XAxis=c('Overfishing', 'Biomass:BMSY'),
YAxis=c('Long-term Yield', 'AnnualVar'), XThresh=c(30, 80), YThresh=c(0,50),
maxVar=15, BmsyRef=0.5, B0Ref=0.2, AvailMPs=NULL, ShowLabs=FALSE,
ShowCols=TRUE)
```

Arguments

MSEobj	Object of class MSE, output of the runMSE function
XAxis	Character string describing the performance metrics for the x-axis (or x-axes if vector; max 4). Must be chosen for list of existing PMs and same length as YAxis. See PMs
YAxis	Character string describing the performance metrics for the y-axis (or y-axes if vector; max 4). Must be chosen for list of existing PMs and same length as XAxis. See PMs
XThresh	Minimum threshold values in percent (i.e., 50 = 50%) for the x-axes (must be same length as XAxis)
YThresh	Minimum threshold values in percent (i.e., 50 = 50%) for the y-axes (must be same length as YAxis)
maxVar	Reference for average annual variability in yield in percent
BmsyRef	Reference level of BMSY, in proportion, i.e., 0.5 = 0.5BMSY
B0Ref	Reference level of B0, in proportion, i.e., 0.2 = 0.2B0

AvailMPs	vector of MPs that *could* be applied to the fishery, i.e., sufficient data exists. These a plotted with different symbol
ShowLabs	Logical to specify if MP labels are shown
ShowCols	Logical to specify if background colors are shown

Details

Returns a list containing the names of performance metrics that meet the minimum performance metrics for each trade-off, and ranks the MPs by increasing distance from the top-right corner.

Author(s)

A. Hordyk

---

trlnorm *Generate log-normally distributed random numbers*

Description

Generate log-normally distributed random numbers

Usage

```
trlnorm(reps, mu, cv)
```

Arguments

reps	number of random numbers
mu	mean
cv	coefficient of variation

Value

numeric

Author(s)

T. Carruthers

updateMSE *Update an MSE object with new slots*

Description

Updates an existing MSE object (class MSE) from a previous version of the DLMtool to include the new slots. The slots will be empty, but avoids the 'slot doesn't exist' error that sometimes occurs. Also works with Stock, Fleet, Obs, Imp, and Data objects.

Usage

```
updateMSE(MSEobj)
```

Arguments

MSEobj	A MSE object from a previous version of the DLMtool. Also works with Stock, Fleet, Obs, Imp, and Data objects.

Value

An object of class matching class(MSEobj)

Author(s)

A. Hordyk

userguide *Open the DLMtool User Guide*

Description

Opens the DLMtool User Guide website (requires internet connection)

Usage

```
userguide()
```

Examples

```
## Not run:
userguide()

## End(Not run)
```

validcpars *Valid custom parameters (cpars)*

Description

Valid custom parameters (cpars)

Usage

```
validcpars(print = TRUE)
```

Arguments

print	Print the valid names for cpars?

Value

invisibly returns vector of valid cpars names

VOI *Calculate Value Of Information*

Description

A function that relates operating model parameters and parameters of the observation model to yield (by default). A user can also specific their own utility values (Ut) which is arranged in a matrix of nsim rows and nMP columns.

Usage

```
VOI(MSEobj, ncomp = 6, nbins = 8, maxrow = 8, Ut = NA, Utnam =
'Utility')
```

Arguments

MSEobj	An object of class MSE
ncomp	Maximum number of variables to examine per MP
nbins	Number of percentile bins for sampled parameters of the operating model or observation model, which is used for calculating variability in utility across the sampled range of each parameter
maxrow	maximum number of MPs per plot
Ut	A matrix of user-specified utility values of nsim rows and nMPs columns
Utnam	The name of the utility measure for plotting

Author(s)

T. Carruthers

VOI2 *Calculate Value Of Information 2*

Description

A function that relates operating model parameters and parameters of the observation model to relative yield (yield over last 5 years of projection relative to a 'best F' scenario that maximizes yield).

Usage

```
VOI2(MSEobj, ncomp = 6, nbins = 4, Ut = NA, Utnam = 'yield', lay = F)
```

Arguments

MSEobj	An object of class MSE
ncomp	Maximum number of observation variables to examine per MP
nbins	Number of bins for sampled observation variables used for calculating variability in utility across the sampled range of each parameter
Ut	A matrix of user-specified utility values of nsim rows and nMPs columns
Utnam	The name of the utility measure for plotting
lay	Controls whether labels are in lay terms or not

Note

VOI2 assumes that relative cost for each type of improvement in data is linearly related to the number of samples (e.g. nCAAobs) or square function of improved precision and bias e.g.: relative cost= 1/(newCV/oldCV)^2

Author(s)

T. Carruthers

VOIplot *Yet another Value of Information Plot*

Description

A function that relates parameters of the observation model and the operating model parameters to yield.

Usage

```
VOIplot(MSEobj, MPs=NA, nvars=5, nMP=4, Par=c('Obs', 'OM'),
YVar=c('Y', 'B'), doPlot=TRUE, incStat=FALSE, availMP=NULL, acceptMP=NULL,
incNames=TRUE, labcex=0.8, quants=c(0.05, 0.95))
```

Arguments

MSEobj	An object of class MSE
MPs	The MPs to plot. If NA it will plot the first nMP from MSEobj
nvars	The number of observation or operating model parameters to plot (number of columns)
nMP	The maximum number of MPs to plot (number of rows)
Par	Plot Operating Model (OM) or Observation (Obs) parameters?
YVar	Variable for Y-Axis: Yield (Y) or Biomass (B) (relative to BMSY)
doPlot	Output the plot?
incStat	Include a print out of statistic describing the curviness of the line?
availMP	Optional character string of MPs that are available. These names are colored black
acceptMP	Optional character string of MPs that are acceptable. These names are colored green if they are also in availMP
incNames	Include the names?
labcex	Character size of the label
quants	Quantiles to calculate

Value

A list of all the information included in the plot

Author(s)

A. Hordyk

---

wormplot *Biomass wormplot*

Description

A worm plot for plotting the likelihood of meeting biomass targets in future years.

Usage

```
wormplot(MSEobj, Bref=0.5, LB=0.25, UB=0.75)
```

Arguments

MSEobj	Object of class MSE, output of the runMSE function
Bref	The reference fraction of BMSY (to evaluate the probability of exceeding this level)
LB	The lower bound probability that seperates red (bad) and yellow (O.K.) colored segments
UB	The upper bound probability that seperates yellow (O.K.) and green (good) colored segments

Details

Returns a matrix of nMPs rows and proyears columns which is the fraction of simulations for which biomass was above Bref.

Author(s)

T. Carruthers

---

writeCSV *Internal function to write CSVs for objects*

Description

Used internally in the DLMtool package to write CSV files from an existing DLMtool object

Usage

```
writeCSV(inobj, tmpfile = NULL, objtype = c("Stock", "Fleet", "Obs", "Imp",
  "Data", "OM", "Fease"))
```

Arguments

inobj	A object of class Stock, Fleet, Obs, Imp, Data, OM, or Fease
tmpfile	The full file path and name for the saved CSV file
objtype	The class corresonding to the inobj

Author(s)

A. Hordyk

XL2OM *Load OM from Excel file*

Description

Imports an OM from a correctly formatted Excel file. Create the Excel spreadsheet template using 'OMinit' and document each slot in the corresponding text file.

Usage

```
XL2OM(name = NULL, cpars = NULL, msg = TRUE)
```

Arguments

name	Name of the OM Excel file in the current working directory.
cpars	An optional list of custom parameters (single parameters are a vector nsim long, time series are a matrix nsim rows by nyears columns)
msg	Should messages be printed?

Details

An error message will alert if any slots are missing values, or if the Excel file is missing the required tabs.

Value

An object of class OM

Author(s)

A. Hordyk

Examples

```
## Not run:
OMinit('myOM', templates=list(Stock='Herring', Fleet='Generic_Fleet', Obs='Generic_Obs',
Imp='Perfect_Imp'), overwrite=TRUE)
myOM <- XL2OM('myOM.xlsx')


## End(Not run)
```

Yield *Performance Metric: Average Yield*

Description

Performance Metric: Average Yield

Usage

```
Yield(MSEobj = NULL)
```

Arguments

MSEobj	An object of class MSE

Value

An object of class PMobj

Examples

```
## Not run:
Yield(myMSE)

## End(Not run)
```

YPR *Yield Per Recruit analysis to get FMSY proxy F01*

Description

A simple yield per recruit approximation to FMSY (F01) which is the position of the ascending YPR curve for which dYPR/dF = 0.1(dYPR/d0)

Usage

```
YPR(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of samples of the TAC

Value

A numeric vector of TAC samples

Note

Based on the code of Meaghan Bryan

Author(s)

Meaghan Bryan and Tom Carruthers

References

Beverton and Holt. 1954.

`YPR_CC` *Yield Per Recruit analysis to get FMSY proxy F01 paired to a naive catch curve estimate of recent Z*

Description

A simple yield per recruit approximation to FMSY (F01) which is the position of the ascending YPR curve for which dYPR/dF = 0.1(dYPR/d0) A naive catch-curve analysis is used to determine recent Z which given M (Mort) gives F and thus abundance = Ct/(1-exp(-F))

Usage

```
YPR_CC(x, Data, reps = 100, Fmin=0.005)
```

Arguments

`x`	A position in a data-limited methods data object
`Data`	A data-limited methods data object (class DLM)
`reps`	The number of samples of the TAC
`Fmin`	The minimum fishing mortality rate inferred from the catch-curve analysis

Author(s)

Meaghan Bryan and T. Carruthers

YPR_ML *Yield Per Recruit analysis to get FMSY proxy F01 paired with a mean-length estimate of current stock size*

Description

A simple yield per recruit approximation to FMSY (F01) which is the position of the ascending YPR curve for which dYPR/dF = 0.1(dYPR/d0) A mean-length estimate of recent Z is used to infer current abundance

Usage

```
YPR_ML(x, Data, reps = 100)
```

Arguments

x	A position in a data-limited methods data object
Data	A data-limited methods data object
reps	The number of samples of the TAC

Note

The mean length extension was programmed by Gary Nelson as part of his excellent R package 'fishmethods'

Author(s)

Meaghan Bryan and T. Carruthers

Index