

Package ‘MSEtool’

October 17, 2018

Type Package

Title Management Strategy Evaluation Toolkit

Version 1.1.0

Date 2018-10-01

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

Description Simulation tools for management strategy evaluation are provided for the 'DLMtool' operating model to inform data-rich fisheries. 'MSEtool' provides complementary assessment models of varying complexity with standardized reporting, diagnostic tools for evaluating assessment models within closed-loop simulation, and helper functions for building more complex operating models and management procedures.

License GPL-3

Depends R (>= 3.3.0), DLMtool (>= 5.2.0)

Imports MASS, TMB, coda, corpcor, gplots, grDevices, graphics, methods, mvtnorm, pryr, r4ss (<= 1.24.0), reshape2, snowfall, stats, utils

LinkingTo TMB, RcppEigen

LazyData yes

LazyLoad yes

RoxygenNote 6.0.1

Suggests knitr, testthat

VignetteBuilder knitr

URL <http://www.datalimitedtoolkit.org>

BugReports <https://github.com/tcarruth/MSEtool/issues>

NeedsCompilation yes

Author Quang Huynh [aut],
Tom Carruthers [aut, cre],
Adrian Hordyk [aut],
Chris Grandin [ctb] (iSCAM functions)

Repository CRAN

Date/Publication 2018-10-17 08:10:03 UTC

R topics documented:

MSEtool-package	3
Assessment-class	4
avail	6
Awatea2OM	7
calculate_TAC	7
Data-rich-MP	8
DD_TMB	10
diagnostic_AM	12
fetch.file.names	13
getGpars	14
getinds	14
getSR	15
HCRlin	16
HCR_FB	17
HCR_MSX	18
HCR_ramp	19
iSCAM2Data	21
iSCAM2OM	21
iSCAMcomps	22
iSCAMinds	23
LinInterp	23
load.iscam.files	24
mahplot	24
makemov	25
make_MP	26
movdistil	27
negcorlogspace	27
optSR	28
plot,Assessment,ANY-method	28
plot_betavar	29
plot_composition	30
plot_crosscorr	31
plot_lognormalvar	32
plot_residuals	33
plot_SR	34
plot_steepness	34
plot_timeseries	35
PRBcalc	36
prelim_AM	37
Probs	38
profile_likelihood	39
read.control.file	40
read.data.file	41
read.mcmc	41
read.par.file	42
read.projection.file	42

read.report.file	43
retrospective	43
retrospective_AM	44
SCA	45
simmov	50
SP	51
SP_production	54
SRopt	55
SS2Data	56
SS2OM	57
summary,Assessment-method	58
swordfish	59
Index	60

MSEtool-package	<i>Management Strategy Evaluation</i>
-----------------	---------------------------------------

Description

Simulation tools for management strategy evaluation are provided for the DLMtool operating model to inform data-rich fisheries. MSEtool provides complementary assessment models of varying complexity with standardized reporting, diagnostic tools for evaluating assessment models within closed-loop simulation, and helper functions for building more complex operating models and management procedures.

Additional Information

See the [DLMtool User Guide](#) for a detailed description of how to use the DLMtool package.

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)

Quang Huynh <q.huynh@oceans.ubc.ca>

Tom Carruthers <t.carruthers@oceans.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*. 73: 464-482.

Assessment-class	Class-Assessment
------------------	------------------

Description

An S4 class that contains assessment output. Created from a function of class Assess.

Slots

Model Name of the assessment model.

Name Name of Data object.

conv Logical. Whether the assessment model converged (defined by whether TMB returned a positive-definite covariance matrix for the model).

UMSY Estimate of exploitation at maximum sustainable yield.

FMSY Estimate of instantaneous fishing mortality rate at maximum sustainable yield.

MSY Estimate of maximum sustainable yield.

BMSY Biomass at maximum sustainable yield.

SSBMSY Spawning stock biomass at maximum sustainable yield.

VBMSY Vulnerable biomass at maximum sustainable yield.

B0 Biomass at virgin equilibrium.

R0 Recruitment at virgin equilibrium.

N0 Abundance at virgin equilibrium.

SSB0 Spawning stock biomass at virgin equilibrium.

VB0 Vulnerable biomass at virgin equilibrium.

h Steepness.

U Time series of exploitation.

U_UMSY Time series of relative exploitation.

FMort Time series of instantaneous fishing mortality.

F_FMSY Time series of fishing mortality relative to MSY.

B Time series of biomass.

B_BMSY Time series of biomass relative to MSY.

B_B0 Time series of depletion.

SSB Time series of spawning stock biomass.

SSB_SSBMSY Time series of spawning stock biomass relative to MSY.

SSB_SSB0 Time series of spawning stock depletion.

VB Time series of vulnerable biomass.

VB_VBMSY Time series of vulnerable biomass relative to MSY.

VB_VB0 Time series of vulnerable biomass depletion.

R Time series of recruitment.
N Time series of population abundance.
N_at_age Time series of numbers-at-age matrix.
Selectivity Selectivity-at-age matrix.
Obs_Catch Observed catch.
Obs_Index Observed index.
Obs_C_at_age Observed catch-at-age matrix.
Catch Predicted catch.
Index Predicted index.
C_at_age Predicted catch-at-age matrix.
Dev A vector of estimated deviation parameters.
Dev_type A description of the deviation parameters, e.g. "log recruitment deviations".
NLL Negative log-likelihood. A vector for the total likelihood, integrated across random effects if applicable, components, and penalty term (applied when $U > 0.975$ in any year).
SE_UMSY Standard error of UMSY estimate.
SE_FMSY Standard error of FMSY estimate.
SE_MSY Standard error of MSY estimate.
SE_U_UMSY_final Standard error of U/UMSY in the terminal year.
SE_F_FMSY_final Standard error of F/FMSY in the terminal year.
SE_B_BMSY_final Standard error of B/BMSY in the terminal year.
SE_B_B0_final Standard error of B/B0 in the terminal year.
SE_SSB_SSBMSY_final Standard error of SSB/SSBMSY in the terminal year.
SE_SSB_SSB0_final Standard error of SSB/SSB0 in the terminal year.
SE_VB_VBMSY_final Standard error of VB/VBMSY in the terminal year.
SE_VB_VB0_final Standard error of VB/VB0 in the terminal year.
SE_Dev A vector of standard errors of the deviation parameters.
info A list containing the data and starting values of estimated parameters for the assessment.
obj A list with components returned from [MakeADFun](#).
opt A list with components from calling [nlminb](#) to **obj**.
SD A list (class `sdreport`) with parameter estimates and their standard errors, obtained from [sdreport](#).
TMB_report A list of model output reported from the TMB executable, i.e. `obj$report()`, and derived quantities (e.g. MSY).
dependencies A character string of data types required for the assessment.

Author(s)

Q. Huynh

See Also

[plot,Assessment,ANY-method summary,Assessment-method retrospective profile_likelihood make_MP](#)

Examples

```
output <- DD_TMB(Data = DLMtool::Red_snapper)
class(output)
```

avail	<i>What objects of this class are available</i>
-------	---

Description

Generic class finder

Usage

```
avail(classy, all_avail = TRUE)
```

Arguments

classy	A class of object (character string, e.g. 'Fleet')
all_avail	Logical. If TRUE, function will return all objects of class classy available to user. If FALSE, returns only those objects included in MSEtool.

Details

Finds objects of the specified class in the global environment or in the MSEtool and DLMtool packages. This function is an addendum to the [avail](#) function in DLMtool.

Author(s)

Q. Huynh

Examples

```
avail("Assess")
avail("HCR")
avail("Stock")
avail("MP")
avail("MP", all_avail = FALSE)
```

Awatea2OM	<i>Reads MCMC estimates from Awatea (Paul Starr) processed r file structure into an operating model</i>
-----------	---

Description

A function that uses the file location of a fitted Awatea model post-processed into a set of rmd files

Usage

```
Awatea2OM(AwateaDir, nsim = 48, proyears = 50, Name = NULL,
  Source = "No source provided", Author = "No author provided",
  verbose = T)
```

Arguments

AwateaDir	A folder with Awatea files
nsim	The number of simulations to take for parameters with uncertainty (for OM@cpars custom parameters)
proyears	The number of projection years for MSE
Name	The name of the operating model
Source	Reference to assessment documentation e.g. a url
Author	Who did the assessment
verbose	Should the r4ss function SS_ouput return detailed messages?

Author(s)

T. Carruthers

calculate_TAC	<i>Calculate TAC from Assessment object</i>
---------------	---

Description

A function to calculate the total allowable catch TAC as the product of either UMSY or FMSY and the vulnerable biomass in terminal year.

Usage

```
calculate_TAC(Assessment, reps, MSY_frac = 1)
```

Arguments

Assessment	An Assessment object with estimates of UMSY or FMSY and terminal year vulnerable biomass.
reps	The number of stochastic draws of UMSY or FMSY.
MSY_frac	The fraction of FMSY or UMSY for calculating the TAC (e.g. MSY_frac = 0.75 fishes at 75% of FMSY).

Value

A vector of length reps of stochastic samples of TAC recommendation.

See Also

[HCR_MSY](#) [HCR40_10](#) [HCR60_20](#)

Data-rich-MP

Data-rich management procedures

Description

A suite of data-rich management procedures (MPs) included in the package. Additional MPs, with specific model configurations (e.g., stock-recruit function or fixing certain parameters) or alternative ramped harvest control rules can be created with [make_MP](#) and the available Assess and HCR objects.

Usage

SCA_MSY(x, Data, reps = 1, ...)

SCA_75MSY(x, Data, reps = 1, ...)

SCA_4010(x, Data, reps = 1, ...)

DDSS_MSY(x, Data, reps = 1, ...)

DDSS_75MSY(x, Data, reps = 1, ...)

DDSS_4010(x, Data, reps = 1, ...)

SP_MSY(x, Data, reps = 1, ...)

SP_75MSY(x, Data, reps = 1, ...)

SP_4010(x, Data, reps = 1, ...)

Arguments

x	A position in the Data object.
Data	An object of class Data
reps	Numeric, the number of stochastic replicates for the management advice.
...	Additional arguments passed to the Assessment model.

Value

An object of class Rec which contains the management recommendation.

Functions

- SCA_MSY: A statistical catch-at-age model with a TAC recommendation based on fishing at UMSY, and default arguments for configuring [SCA](#).
- SCA_75MSY: An SCA with a TAC recommendation based on fishing at 75% of UMSY.
- SCA_4010: An SCA with a 40-10 control rule.
- DDSS_MSY: A state-space delay difference model with a TAC recommendation based on fishing at UMSY, and default arguments for configuring [DD_SS](#).
- DDSS_75MSY: A state-space delay difference model with a TAC recommendation based on fishing at 75% of UMSY.
- DDSS_4010: A state-space delay difference model with a 40-10 control rule.
- SP_MSY: A surplus production model with a TAC recommendation based on fishing at UMSY, and default arguments for configuring [SP](#).
- SP_75MSY: A surplus production model with a TAC recommendation based on fishing at 75% of UMSY.
- SP_4010: A surplus production model with a 40-10 control rule.

Examples

```
avail("MP", all_avail = FALSE)

## Not run:
myMSE <- DLMtool::runMSE(DLMtool::testOM, MPs = c("FMSYref", "SCA_MSY", "SCA_4010"))

## End(Not run)
```

Description

A simple delay-difference assessment model using a time-series of catches and a relative abundance index and coded in TMB. The model is conditioned on effort and estimates predicted catch. In the state-space version, recruitment deviations from the stock-recruit relationship are estimated.

Usage

```
DD_TMB(x = 1, Data, SR = c("BH", "Ricker"), rescale = "mean1",
       start = NULL, fix_h = FALSE, fix_U_equilibrium = TRUE, silent = TRUE,
       opt_hess = FALSE, n_restart = ifelse(opt_hess, 0, 1),
       control = list(iter.max = 5000, eval.max = 10000), ...)
```

```
DD_SS(x = 1, Data, SR = c("BH", "Ricker"), rescale = "mean1",
      start = NULL, fix_h = FALSE, fix_U_equilibrium = TRUE,
      fix_sigma = FALSE, fix_tau = TRUE, integrate = FALSE, silent = TRUE,
      opt_hess = FALSE, n_restart = ifelse(opt_hess, 0, 1),
      control = list(iter.max = 5000, eval.max = 10000), inner.control = list(),
      ...)
```

Arguments

x	An index for the objects in Data when running in closed loop simulation. Otherwise, equals to 1 when running an assessment.
Data	An object of class Data .
SR	Stock-recruit function (either "BH" for Beverton-Holt or "Ricker").
rescale	A multiplicative factor that rescales the catch in the assessment model, which can improve convergence. By default, "mean1" scales the catch so that time series mean is 1, otherwise a numeric. Output is re-converted back to original units.
start	Optional list of starting values. See details.
fix_h	Logical, whether to fix steepness to value in Data@steep in the assessment model.
fix_U_equilibrium	Logical, whether the equilibrium harvest rate prior to the first year of the model is estimated. If TRUE, U_equilibrium is fixed to value provided in start (if provided), otherwise, equal to zero (assumes virgin conditions).
silent	Logical, passed to MakeADFun , whether TMB will print trace information during optimization. Used for diagnostics for model convergence.
opt_hess	Logical, whether the hessian function will be passed to nlminb during optimization (this generally reduces the number of iterations to convergence, but is memory and time intensive and does not guarantee an increase in convergence rate). Ignored if integrate = TRUE.

<code>n_restart</code>	The number of restarts (calls to <code>nlminb</code>) in the optimization procedure, so long as the model hasn't converged. The optimization continues from the parameters from the previous (re)start.
<code>control</code>	A named list of parameters regarding optimization to be passed to <code>nlminb</code> .
<code>...</code>	Additional arguments (not currently used).
<code>fix_sigma</code>	Logical, whether the standard deviation of the catch is fixed. If TRUE, sigma is fixed to value provided in <code>start</code> (if provided), otherwise, value based on <code>Data@CV_Cat</code> .
<code>fix_tau</code>	Logical, the standard deviation of the recruitment deviations is fixed. If TRUE, tau is fixed to value provided in <code>start</code> (if provided), otherwise, equal to 1.
<code>integrate</code>	Logical, whether the likelihood of the model integrates over the likelihood of the recruitment deviations (thus, treating it as a state-space variable).
<code>inner.control</code>	A named list of arguments for optimization of the random effects, which is passed on to <code>newton</code> via <code>MakeADFun</code> .

Details

To provide starting values for DD_TMB, a named list can be provided for R_0 (virgin recruitment), h (steepness), and q (catchability coefficient) via the `start` argument (see example).

For DD_SS, additional start values can be provided for `sigma` and `tau`, the standard deviation of the catch and recruitment variability, respectively.

Value

An object of `Assessment` containing objects and output from TMB.

Functions

- DD_TMB: Observation-error only model
- DD_SS: State-Space version of Delay-Difference model

Required Data

- DD_TMB: `Cat`, `Ind`, `Mort`, `L50`, `vbK`, `vbLinf`, `vbt0`, `wla`, `wlb`, `MaxAge`
- DD_SS: `Cat`, `Ind`, `Mort`, `L50`, `vbK`, `vbLinf`, `vbt0`, `wla`, `wlb`, `MaxAge`

Optional Data

- DD_TMB: `steep`
- DD_SS: `steep`, `CV_Cat`

Note

Similar to many other assessment models, the model depends on assumptions such as 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 & Z. Siders. Zach Siders coded the TMB function.

References

Carruthers, T, Walters, C.J., and McAllister, M.K. 2012. Evaluating methods that classify fisheries stock status using only fisheries catch data. Fisheries Research 119-120:66-79.

Hilborn, R., and Walters, C., 1992. Quantitative Fisheries Stock Assessment: Choice, Dynamics and Uncertainty. Chapman and Hall, New York.

See Also

[plot,Assessment,ANY-method summary,Assessment-method retrospective profile_likelihood make_MP](#)

Examples

```
#### Observation-error delay difference model
res <- DD_TMB(Data = DLMtool::Red_snapper)

# Provide starting values
start <- list(R0 = 1, h = 0.95)
res <- DD_TMB(Data = DLMtool::Red_snapper, start = start)

summary(res@SD) # Parameter estimates

### State-space version
### Set recruitment variability SD = 0.3 (since fix_tau = TRUE)
res <- DD_SS(Data = Red_snapper, start = list(tau = 0.3))
```

diagnostic_AM

diagnostic_AM (diagnostic of Assessments in MSE): Did Assess models converge during MSE?

Description

Diagnostic check for convergence of Assess models during MSE. Assess models write output to the DLMenv environment if the MP was created with [make_MP](#) with argument `diagnostic = TRUE`. This function summarizes and plots the diagnostic information.

Usage

```
diagnostic_AM(MSE, MP = NULL, gradient_threshold = 0.1, figure = TRUE)
```

Arguments

MSE	An object of class MSE created by <code>runMSE</code> . If no MSE object is available, use argument MP instead.
MP	A character vector of MPs with assessment models.
gradient_threshold	The maximum magnitude (absolute value) desired for the gradient of the likelihood.
figure	Logical, whether a figure will be drawn.

Value

A matrix with diagnostic performance of assessment models in the MSE. If `figure = TRUE`, a set of figures: traffic light (red/green) plots indicating whether the model converged (defined if a positive-definite Hessian matrix was obtained), the optimizer reached pre-specified iteration limits (as passed to `nlminb`), and the maximum gradient of the likelihood in each assessment run. Also includes the number of optimization iterations function evaluations reported by `nlminb` for each application of the assessment model.

Author(s)

Q. Huynh

See Also

[retrospective_AM](#)

Examples

```
## Not run:
DD_MSY <- make_MP(DD_TMB, HCR_MSY, diagnostic = "min")
show(DD_MSY)

# Set PPD = TRUE in runMSE function
myMSE <- runMSE(DLMtool::testOM, MPs = "DD_MSY", PPD = TRUE)
diagnostic_AM(myMSE)

## End(Not run)
```

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)

getGpars	<i>Extracts growth parameters from a SS3 r4ss replist</i>
----------	---

Description

Extracts growth parameters from a SS3 r4ss replist

Usage

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

Arguments

repllist	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

getinds	<i>Characterize posterior predictive data</i>
---------	---

Description

Characterize posterior predictive data

Usage

```
getinds(PPD, styr, res = 6, tsd = c("Cat", "Cat", "Cat", "Ind", "ML"),
  stat = c("slp", "AAV", "mu", "slp", "slp"))
```

Arguments

PPD	An object of class Data stored in the Misc slot of an MSE object following a call of runMSE(PPD = TRUE).
styr	Positive integer, the starting year for calculation of quantities
res	Positive integer, the temporal resolution (chunks - normally years) over which to calculate quantities
tsd	Character vector of names of types of data: Cat = catch, Ind = relative abundance index, ML = mean length in catches
stat	Character vector of types of quantity to be calculated: slp = slope(log(x)), AAV = average annual variability, mu = mean(log(x))

Value

A 3D array of results (type of data/stat (e.g. mean catches), time period (chunk), simulation)

Author(s)

T. Carruthers

References

Carruthers and Hordyk 2018

getSR

Predict recruitment and return fit to S-R observations

Description

Internal function to [optSR](#)

Usage

```
getSR(pars, SSB, rec, SSBpR, mode = 1, plot = FALSE, type = c("BH",
  "Ricker"))
```

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 (= 1) or recruitment deviations (not 1) be returned
plot	should a plot of the model fit be produced?#
type	what type of stock recruitment curve is being fitted ("BH" = Beverton-Holt or "Ricker")

Author(s)

T. Carruthers

HCRlin*Generic linear harvest control rule based on biomass*

Description

A general function used by HCR_ramp that adjusts the TAC by a linear ramp based on estimated biomass.

Usage

```
HCRlin(Brel, LRP, TRP)
```

Arguments

Brel	Improper fraction: An estimate of biomass (either absolute or relative, e.g. B/BMSY or B/B0).
LRP	Improper fraction: the Limit Reference Point, the biomass below which the adjustment is zero (no fishing). Same units as Brel.
TRP	Improper fraction: the Target Reference Point, the biomass above which the adjustment is 1 (no adjustment). Same units as Brel.

Value

a TAC or TAE adjustment factor.

Author(s)

T. Carruthers

Examples

```
#40-10 linear ramp
Brel <- seq(0, 1, length.out = 200)
plot(Brel, HCRlin(Brel, 0.1, 0.4), xlab = "Estimated B/B0", ylab = "TAC adjustment factor",
     main = "A 40-10 harvest control rule", type = 'l', col = 'blue')
abline(v = c(0.1,0.4), col = 'red', lty = 2)
```

HCR_FB	<i>A Harvest Control Rule using B/BMSY and F/FMSY to adjust TAC or TAE.</i>
--------	---

Description

A Harvest Control Rule using B/BMSY and F/FMSY to adjust TAC or TAE.

Usage

```
HCR_FB(Brel, Frel, Bpow = 2, Bgrad = 1, Fpow = 1, Fgrad = 1)
```

Arguments

Brel	improper fraction: an estimate of Biomass relative to BMSY
Frel	improper fraction: an estimate of Fishing mortality rate relative to FMSY
Bpow	non-negative real number: controls the shape of the biomass adjustment, when zero there is no adjustment
Bgrad	non-negative real number: controls the gradient of the biomass adjustment
Fpow	non-negative real number: controls the adjustment speed relative to F/FMSY. When set to 1, next recommendation is FMSY. When less than 1 next recommendation is between current F and FMSY.
Fgrad	improper fraction: target Fishing rate relative to FMSY

Value

a TAC or TAE adjustment factor.

Author(s)

T. Carruthers

References

Made up for this package

Examples

```
res <- 100
Frel <- seq(1/2, 2, length.out = res)
Brel <- seq(0.05, 2, length.out=res)
adj <- array(HCR_FB(Brel[rep(1:res, res)], Frel[rep(1:res, each = res)]),
             Bpow = 2, Bgrad = 1, Fpow = 1, Fgrad = 0.75), c(res, res))
contour(Brel, Frel, adj, nlevels = 20, xlab = "B/BMSY", ylab = "F/FMSY",
        main = "FBsurface TAC adjustment factor")
abline(h = 1, col = 'red', lty = 2)
abline(v = 1, col = 'red', lty = 2)
legend('topright', c("Bpow = 2", "Bgrad = 1", "Fpow = 1", "Fgrad = 0.75"), text.col = 'blue')
```

HCR_MSY	<i>Harvest control rule to fish at some fraction of maximum sustainable yield</i>
---------	---

Description

A simple control rule that specifies the total allowable catch (TAC) to be the product of current vulnerable biomass and UMSY.

Usage

```
HCR_MSY(Assessment, reps = 1, MSY_frac = 1, ...)
```

Arguments

Assessment	An object of class Assessment with estimates of FMSY or UMSY and vulnerable biomass in terminal year.
reps	The number of stochastic samples of the TAC recommendation.
MSY_frac	The fraction of FMSY or UMSY for calculating the TAC (e.g. MSY_frac = 0.75 fishes at 75% of FMSY).
...	Miscellaneous arguments.

Value

An object of class [Rec](#) with the TAC recommendation.

Author(s)

Q. Huynh

References

Punt, A. E, Dorn, M. W., and Haltuch, M. A. 2008. Evaluation of threshold management strategies for groundfish off the U.S. West Coast. *Fisheries Research* 94:251-266.

See Also

[make_MP](#)

Examples

```
# create an MP to run in closed-loop MSE (fishes at UMSY)
DD_MSY <- make_MP(DD_TMB, HCR_MSY)
class(DD_MSY)

# The same MP which fishes at 75% of UMSY
DD_75MSY <- make_MP(DD_TMB, HCR_MSY, MSY_frac = 0.75)
class(DD_75MSY)
```

```
## Not run:
myOM <- DLMtool::runMSE(DLMtool::testOM, MPs = c("FMSYref", "DD_MSY"))

## End(Not run)
```

HCR_ramp

*Ramped harvest control rules***Description**

An output control rule with a ramp that reduces the TAC recommendation ($UMSY * \text{Vulnerable biomass}$) linearly when the relative biomass (i.e., spawning depletion or spawning biomass relative to that at MSY) is less than the target reference point (TRP). The TAC reduction is linearly reduced to 0 when the relative biomass is less than the limit reference point (LRP). For example, the TRP and LRP for spawning depletion is 0.4 and 0.1, respectively, in the 40-10 control rule. Class HCR objects are typically used with function [make_MP](#).

Usage

```
HCR_ramp(Assessment, reps = 1, LRP, TRP, RP_type = c("SSB_SSB0",
  "SSB_SSBMSY"), ...)

HCR40_10(Assessment, reps = 1, ...)

HCR60_20(Assessment, reps = 1, ...)
```

Arguments

Assessment	An object of class Assessment with estimates of FMSY or UMSY, vulnerable biomass, and spawning biomass depletion in terminal year.
reps	The number of stochastic samples of the TAC recommendation.
LRP	Numeric, the limit reference point.
TRP	Numeric, the target reference point.
RP_type	The reference point metric for TRP and LRP ("SSB_SSB0" for spawning depletion by default, or "SSB_SSBMSY" for spawning biomass relative to MSY).
...	Miscellaneous arguments.

Value

An object of class [Rec](#) with the TAC recommendation.

Functions

- HCR_ramp: Generic ramped-HCR function where user specifies LRP, TRP, and relative biomass metric.
- HCR40_10: Common U.S. west coast control rule (LRP and TRP of 0.1 and 0.4 spawning depletion, respectively).
- HCR60_20: More conservative than 40-10, with LRP and TRP of 0.2 and 0.6 spawning depletion, respectively).

Note

Function will use slot B_B0 of the Assessment object (e.g. for surplus production models [SP](#) and [SP_SS](#)) if SSB_SSB0 is unavailable.

Author(s)

Q. Huynh & T. Carruthers

References

- Deroba, J.J. and Bence, J.R. 2008. A review of harvest policies: Understanding relative performance of control rules. *Fisheries Research* 94:210-223.
- Edwards, C.T.T. and Dankel, D.J. (eds.). 2016. *Management Science in Fisheries: an introduction to simulation methods*. Routledge, New York, NY. 460 pp.
- Punt, A. E, Dorn, M. W., and Haltuch, M. A. 2008. Evaluation of threshold management strategies for groundfish off the U.S. West Coast. *Fisheries Research* 94:251-266.
- Restrepo, V.R. and Power, J.E. 1999. Precautionary control rules in US fisheries management: specification and performance. *ICES Journal of Marine Science* 56:846-852.

See Also

[HCRlin](#) [make_MP](#)

Examples

```
# 40-10 linear ramp
Brel <- seq(0, 1, length.out = 200)
plot(Brel, HCRlin(Brel, 0.1, 0.4), xlab = "Estimated SSB/SSB0",
     ylab = "TAC adjustment factor", main = "40-10 harvest control rule",
     type = "l", col = "blue")
abline(v = c(0.1, 0.4), col = "red", lty = 2)

# create a 40-10 MP to run in closed-loop MSE
DD_40_10 <- make_MP(DD_TMB, HCR40_10)

# Alternatively,
DD_40_10 <- make_MP(DD_TMB, HCR_ramp, LRP = 0.1, TRP = 0.4)

# An SCA with LRP and TRP at 0.4 and 0.8, respectively, of SSB/SSBMSY
SCA_80_40 <- make_MP(SCA, HCR_ramp, LRP = 0.4, TRP = 0.8, RP_type = "SSB_SSBMSY")
```

```
## Not run:
myOM <- DLMtool::runMSE(DLMtool::testOM, MPs = c("FMSYref", "DD_40_10"))

## End(Not run)
```

iSCAM2Data	<i>Reads data from iSCAM file structure into a DLMtool Data object</i>
------------	--

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. iSCAM2OM 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

iSCAM2OM	<i>Reads MLE estimates from iSCAM file structure into an operating model</i>
----------	--

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. iSCAM2OM relies on several functions written by Chris Grandin (DFO PBS).

Usage

```
iSCAM2OM(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@cpar custom parameters)
proyears	The number of MSE projection years
mcmc	Whether to use mcmc samples to create custom parameters cpar
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	<i>Combines indices into a single index using linear modelling</i>
-----------	--

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

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

Author(s)

T. Carruthers

<code>LinInterp</code>	<i>Linear interpolation of a y value at level xlev based on a vector x and y</i>
------------------------	--

Description

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

Usage

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

Arguments

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

Author(s)

T. Carruthers

load.iscam.files	<i>Reads iSCAM files into a hierarchical R list object</i>
------------------	--

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)

mahplot	<i>Plot statistical power of the indicator with increasing time blocks</i>
---------	--

Description

Plot statistical power of the indicator with increasing time blocks

Usage

```
mahplot(outlist, res = 6, maxups = 5, MPs)
```

Arguments

outlist	A list object produced by the function PRBcalc
res	Integer, the resolution (time blocking) for the calculation of PPD
maxups	Integer, the maximum number of update time blocks to plot
MPs	Character vector of MP names

Author(s)

T. Carruthers

References

Carruthers and Hordyk 2018

makemov	<i>Calculates movement matrices from user inputs for fraction in each area (fracs) and probability of staying in areas (prob)</i>
---------	---

Description

A function for calculating a movement matrix from user specified unfished stock biomass fraction in each area. Used by [simmov](#) to generate movement matrices for a DLMtool operating model.

Usage

```
makemov(frac = c(0.1, 0.2, 0.3, 0.4), prob = c(0.5, 0.8, 0.9, 0.95))
```

Arguments

frac	A vector nareas long of fractions of unfished stock biomass in each area
prob	A vector of the probability of individuals staying in each area or a single value for the mean probability of staying among all areas

Author(s)

T. Carruthers

See Also

[simmov](#)

make_MP	<i>Make a custom management procedure (MP)</i>
---------	--

Description

Function operator that combines a function of class `Assess` and a function of class `HCR` to create a management procedure (MP). The resulting function can then be tested in closed-loop simulation via [runMSE](#).

Usage

```
make_MP(.Assess, .HCR, diagnostic = c("none", "min", "full"), ...)
```

Arguments

<code>.Assess</code>	A function of class <code>Assess</code> .
<code>.HCR</code>	A function of class <code>HCR</code> .
<code>diagnostic</code>	A character string describing if any additional diagnostic information from the assessment models will be collected during a call with runMSE ("none" is the default). "min" (minimal) will collect information on convergence and "full" will also collect the Assessment object generated by the <code>.Assess</code> . This information will be written to DLMenv . See example.
<code>...</code>	Additional arguments to be passed to <code>.Assess</code> and <code>.HCR</code> .

Value

A function of class `MP`.

See Also

[diagnostic_AM](#) [retrospective_AM](#)

Examples

```
# A delay-difference model with a 40-10 control rule
DD_40_10 <- make_MP(DD_TMB, HCR40_10)

# A delay difference model that will produce convergence diagnostics
DD_40_10 <- make_MP(DD_TMB, HCR40_10, diagnostic = "min")

# MP that uses a Delay-Difference which assumes a Ricker stock-recruit function.
DD_Ricker <- make_MP(DD_TMB, HCR_MSY, SR = "Ricker")

## Not run:
myMSE <- DLMtool::runMSE(DLMtool::testOM, MPs = c("FMSYref", "DD_40_10"))

ls(DLMenv) # Model output during MSE is assigned to this environment.
diagnostic_AM(myMSE)
```

```
## End(Not run)
```

movdistil	<i>Simplified a multi-area transition matrix into the best 2 x 2 representation</i>
-----------	---

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

negcorlogspace	<i>A function that samples multivariate normal (logspace) variables</i>
----------------	---

Description

A function that samples multivariate normal (logspace) variables

Usage

```
negcorlogspace(xmu, ymu, xcv = 0.1, nsim, cor = -0.9, ploty = FALSE)
```

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

optSR	<i>Wrapper for estimating stock recruitment parameters from resampled stock-recruitment data</i>
-------	--

Description

Wrapper for estimating stock recruitment parameters from resampled stock-recruitment data

Usage

```
optSR(x, SSB, rec, SSBpR, pars, frac = 0.5, plot = FALSE, type = c("BH",
  "Ricker"))
```

Arguments

x	position to accommodate lapply-type functions
SSB	'observations' of spawning biomass
rec	'observations' (model predictions) of recruitment
SSBpR	spawning stock biomass per recruit at unfished conditions
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?
type	what type of stock recruitment curve is being fitted ("BH" = Beverton-Holt or "Ricker")

Value

Estimated value of steepness.

Author(s)

T. Carruthers

plot, Assessment, ANY-method	<i>Plot Assessment object</i>
------------------------------	-------------------------------

Description

Plot Assessment object

Usage

```
## S4 method for signature 'Assessment,ANY'
plot(x, save_figure = TRUE, save_dir = tempdir())
```

Arguments

x	An object of class Assessment
save_figure	Indicates whether figures will be saved to disk. A corresponding html report will be produced.
save_dir	The directory (by default, the current working directory) in which a sub-directory will be created to save figures.

Examples

```
output <- DD_TMB(Data = Simulation_1)
plot(output, save_figure = FALSE, save_dir = tempdir())
```

plot_betavar	<i>Plots a beta variable</i>
--------------	------------------------------

Description

Plots the probability distribution function of a beta variable from the mean and standard deviation in either transformed (logit) or untransformed space.

Usage

```
plot_betavar(m, sd, label = NULL, is_logit = FALSE, color = "black")
```

Arguments

m	A vector of means of the distribution.
sd	A vector of standard deviations of the distribution.
label	Name of the variable to be used as x-axis label.
is_logit	Logical that indicates whether the means and standard deviations are in transformed (logit) or untransformed space.
color	A vector of colors.

Value

A plot of the probability distribution function. Vertical dotted line indicates mean of distribution. This function can plot multiple curves when multiple means and standard deviations are provided.

Author(s)

Q. Huynh

See Also

[plot_lognormalvar](#) [plot_steepness](#)

Examples

```

mu <- 0.5
stddev <- 0.1
plot_betavar(mu, stddev) # mean of plot should be 0.5

#logit parameters
mu <- 0
stddev <- 0.1
plot_betavar(mu, stddev, is_logit = TRUE) # mean of plot should be 0.5

```

plot_composition	<i>Plot composition data</i>
------------------	------------------------------

Description

Plots annual length or age composition data.

Usage

```

plot_composition(Year = 1:nrow(obs), obs, fit = NULL,
  plot_type = c("annual", "bubble_data", "bubble_residuals", "mean"),
  N = rowSums(obs), CAL_bins = NULL, ind = 1:nrow(obs), bubble_adj = 5,
  fit_linewidth = 3, fit_color = "red")

```

Arguments

Year	A vector of years.
obs	A matrix of either length or age composition data. For lengths, rows and columns should index years and length bin, respectively. For ages, rows and columns should index years and age, respectively.
fit	A matrix of predicted length or age composition from an assessment model. Same dimensions as obs.
plot_type	Indicates which plots to create. Options include annual distributions, bubble plot of the data, and bubble plot of the residuals, and annual means.
N	Annual sample sizes. Vector of length nrow(obs).
CAL_bins	A vector of lengths corresponding to the columns in obs. and fit. Ignored for age data.
ind	A numeric vector for plotting a subset of rows (which indexes year) of obs and fit.
bubble_adj	Numeric, for adjusting the relative size of bubbles in bubble plots (larger number = larger bubbles).
fit_linewidth	Argument lwd for fitted line.
fit_color	Color of fitted line.

Value

Plots depending on plot_type.

Author(s)

Q. Huynh

Examples

```
data(Red_snapper)
plot_composition(obs = Red_snapper@CAA[1, , ], plot_type = "annual")
plot_composition(obs = Red_snapper@CAA[1, , ], plot_type = "bubble_data")

plot_composition(obs = Red_snapper@CAL[1, , ], plot_type = "annual", Red_snapper@CAL_bins[1:43])
plot_composition(obs = Red_snapper@CAL[1, , ], plot_type = "bubble_data",
CAL_bins = Red_snapper@CAL_bins[1:43])
```

plot_crosscorr	<i>Produce a cross-correlation plot of the derived data arising from getinds(MSE_object)</i>
----------------	--

Description

Produce a cross-correlation plot of the derived data arising from getinds(MSE_object)

Usage

```
plot_crosscorr(indPPD, indData, pp = 1, dnam = c("CS", "CV", "CM", "IS",
"MLS"), res = 1)
```

Arguments

indPPD	A 3D array of results arising from running getind on an MSE of the Null operating model (type of data/stat (e.g. mean catches),time period (chunk), simulation)
indData	A 3D array of results arising from running getind on an MSE of the Alternative operating model (type of data/stat (e.g. mean catches),time period (chunk), simulation)
pp	Positive integer, the number of time chunks (blocks of years normally, second dimension of indPPD and indData) to produce the plot for.
dnam	A character vector of names of the data for plotting purposes (as long as dimension 1 of indPPD and indData).
res	The size of the temporal blocking that created indPPD and indData - this is just used for labelling purposes

Value

A cross-correlation plot (ndata-1) x (ndata-1)

Author(s)

T. Carruthers

References

Carruthers and Hordyk 2018

plot_lognormalvar *Plots a lognormal variable*

Description

Plots the probability distribution function of a lognormal variable from the mean and standard deviation in either transformed (normal) or untransformed space.

Usage

```
plot_lognormalvar(m, sd, label = NULL, logtransform = FALSE,
  color = "black")
```

Arguments

m	A vector of means of the distribution.
sd	A vector of standard deviations of the distribution.
label	Name of the variable to be used as x-axis label.
logtransform	Indicates whether the mean and standard deviation are in transformed (normal) or untransformed space.
color	A vector of colors.

Value

A plot of the probability distribution function. Vertical dotted line indicates mean of distribution. This function can plot multiple curves when multiple means and standard deviations are provided.

Author(s)

Q. Huynh

See Also

[plot_betavar](#) [plot_steepness](#)

Examples

```

mu <- 0.5
stddev <- 0.1
plot_lognormalvar(mu, stddev) # mean of plot should be 0.5

#logtransformed parameters
mu <- 0
stddev <- 0.1
plot_lognormalvar(mu, stddev, logtransform = TRUE) # mean of plot should be 1

```

plot_residuals	<i>Plot residuals</i>
----------------	-----------------------

Description

Plots figure of residuals (or any time series with predicted mean of zero).

Usage

```

plot_residuals(Year, res, res_sd = NULL, res_sd_CI = 0.95,
  res_upper = NULL, res_lower = NULL, res_ind_blue = NULL,
  draw_zero = TRUE, zero_linetype = 2, label = "Residual")

```

Arguments

Year	A vector of years for the data.
res	A vector of residuals.
res_sd	A vector of year specific standard deviation for res.
res_sd_CI	The confidence interval for the error bars based for res_sd.
res_upper	A vector of year-specific upper bounds for the error bars of the residual (in lieu of argument res_CV).
res_lower	A vector of year-specific lower bounds for the error bars of the residual (in lieu of argument res_CV).
res_ind_blue	Indices of obs for which the plotted residuals and error bars will be blue.
draw_zero	Indicates whether a horizontal line should be drawn at zero.
zero_linetype	Passes argument lty (e.g. solid line = 1, dotted = 2) to draw_zero.
label	Character string that describes the data to label the y-axis.

Author(s)

Q. Huynh

See Also

[plot_timeseries](#)

plot_SR *Plot stock-recruitment function*

Description

Plot stock-recruitment (with recruitment deviations if estimated).

Usage

```
plot_SR(Spawners, expectedR, R0, S0, rec_dev = NULL, trajectory = FALSE,
        y_zoom = NULL)
```

Arguments

Spawners	A vector of the number of the spawners (x-axis).
expectedR	A vector of the expected recruitment (from the stock-recruit function) corresponding to values of Spawners.
R0	Virgin recruitment.
S0	Virgin spawners.
rec_dev	If recruitment deviations are estimated, a vector of estimated recruitment (in normal space) corresponding to values of Spawners.
trajectory	Indicates whether arrows will be drawn showing the trajectory of spawners and recruitment deviations over time.
y_zoom	If recruitment deviations are plotted, the y-axis limit relative to maximum expected recruitment expectedR. If NULL, all recruitments are plotted.

Value

A stock-recruit plot

Author(s)

Q. Huynh

plot_steepness *Plots probability distribution function of stock-recruit steepness*

Description

Plots the probability distribution function of steepness from the mean and standard deviation.

Usage

```
plot_steepness(m, sd, is_transform = FALSE, SR = c("BH", "Ricker"),
               color = "black")
```

Arguments

m	The mean of the distribution (vectorized).
sd	The standard deviation of the distribution (vectorized).
is_transform	Logical, whether the mean and standard deviation are in normal space (FALSE) or transformed space.
SR	The stock recruitment relationship (determines the range and, if relevant, transformation of steepness).
color	A vector of colors.

Value

A plot of the probability distribution function. Vertical dotted line indicates mean of distribution.

Note

The function samples from a beta distribution with parameters alpha and beta that are converted from the mean and standard deviation. Then, the distribution is transformed from 0 - 1 to 0.2 - 1.

Author(s)

Q. Huynh

See Also

[plot_lognormalvar](#) [plot_betavar](#)

Examples

```
mu <- DLMtool::Simulation_1@steep
stddev <- DLMtool::Simulation_1@steep * DLMtool::Simulation_1@CV_steep
plot_steepness(mu, stddev)
```

plot_timeseries	<i>Plot time series of data</i>
-----------------	---------------------------------

Description

Plot time series of observed (with lognormally-distributed error bars) vs. predicted data.

Usage

```
plot_timeseries(Year, obs, fit = NULL, obs_CV = NULL, obs_CV_CI = 0.95,
  obs_upper = NULL, obs_lower = NULL, obs_ind_blue = NULL,
  fit_linewidth = 3, fit_color = "red", label = "Observed data")
```

Arguments

Year	A vector of years for the data.
obs	A vector of observed data.
fit	A vector of predicted data (e.g., from an assessment model).
obs_CV	A vector of year-specific coefficient of variation in the observed data.
obs_CV_CI	The confidence interval for the error bars based for obs_CV.
obs_upper	A vector of year-specific upper bounds for the error bars of the observed data (in lieu of argument obs_CV).
obs_lower	A vector of year-specific lower bounds for the error bars of the observed data (in lieu of argument obs_CV).
obs_ind_blue	Indices of obs for which the plotted points and error bars will be blue.
fit_linewidth	Argument lwd for fitted line.
fit_color	Color of fitted line.
label	Character string that describes the data to label the y-axis.

Author(s)

Q. Huynh

See Also

[plot_residuals](#)

Examples

```
data(Red_snapper)
plot_timeseries(Red_snapper@Year, Red_snapper@Cat[1, ],
obs_CV = Red_snapper@CV_Cat, label = "Catch")
```

PRBcalc

Calculate mahalanobis distance (null and alternative MSEs) and statistical power for all MPs in an MSE

Description

Calculate mahalanobis distance (null and alternative MSEs) and statistical power for all MPs in an MSE

Usage

```
PRBcalc(MSE_null, MSE_alt, tsd = c("Cat", "Cat", "Cat", "Ind", "ML"),
stat = c("slp", "AAV", "mu", "slp", "slp"), dnam = c("C_S", "C_V", "C_M",
"I_S", "ML_S"), res = 6, alpha = 0.05, plotCC = FALSE,
removedat = FALSE, removethresh = 0.025)
```

Arguments

MSE_null	An object of class MSE representing the null hypothesis
MSE_alt	An object of class MSE representing the alternative hypothesis
tsd	Character string of data types: Cat = catch, Ind = relative abundance index, ML = mean length in catches
stat	Character string defining the quantity to be calculated for each data type, slp = slope(log(x)), AAV = average annual variability, mu = mean(log(x))
dnam	Character string of names for the quantities calculated
res	Integer, the resolution (time blocking) for the calculation of PPD
alpha	Probability of incorrectly rejecting the null operating model when it is valid
plotCC	Logical, should the PPD cross correlations be plotted?
removedat	Logical, should data not contributing to the mahalanobis distance be removed?
removethresh	Positive fraction: the cumulative percentage of removed data (removedat=TRUE) that contribute to the mahalanobis distance

Value

A list object with two hierarchies of indexing, first by MP, second has two positions as described in [Probs](#): (1) mahalanobis distance, (2) a matrix of type 1 error (first row) and statistical power (second row), by time block.

Author(s)

T. Carruthers

References

Carruthers, T.R, and Hordyk, A.R. In press. Using management strategy evaluation to establish indicators of changing fisheries. Canadian Journal of Fisheries and Aquatic Science.

```
prelim_AM
```

Preliminary Assessments in MSE

Description

Evaluates the likely performance of Assessment models in the operating model. This function will apply the assessment model for Data generated during the historical period of the MSE, and report the convergence rate for the model and total time elapsed in running the assessments.

Usage

```
prelim_AM(x, Assess, ncpus = 1, ...)
```

Arguments

x	Either a Hist, Data or OM object.
Assess	An Assess function of class Assess.
ncpus	Numeric, the number of CPUs to run the Assessment model (will run in parallel if greater than 1).
...	Arguments to be passed to Assess, e.g., model configurations.

Value

Returns invisibly a list of [Assessment](#) objects of length `OM@nsim`. Messages via console.

Author(s)

Q. Huynh

Examples

```
## Not run:
prelim_AM(DLMtool::testOM, DD_TMB)

## End(Not run)
```

Probs	<i>Calculates mahalanobis distance and rejection of the Null operating model</i>
-------	--

Description

Calculates mahalanobis distance and rejection of the Null operating model, used by wrapping function [PRBcalc](#).

Usage

```
Probs(indPPD, indData, alpha = 0.05, removedat = FALSE,
      removethresh = 0.05)
```

Arguments

indPPD	A 3D array of results arising from running <code>getind</code> on an MSE of the Null operating model (type of data/stat (e.g. mean catches), time period (chunk), simulation)
indData	A 3D array of results arising from running <code>getind</code> on an MSE of the Alternative operating model (type of data/stat (e.g. mean catches), time period (chunk), simulation)
alpha	Positive fraction: rate of type I error, alpha
removedat	Logical, should data not contributing to the mahalanobis distance be removed?
removethresh	Positive fraction: the cumulative percentage of removed data (<code>removedat=TRUE</code>) that contribute to the mahalanobis distance

Value

A list object. Position 1 is an array of the mahalanobis distances. Dimension 1 is length 2 for the Null OM (indPPD) and the alternative OM (indData). Dimension 2 is the time block (same length as indPPD dim 2). Dimension 3 is the simulation number (same length as indPPD dim 3.), Position 2 is a matrix (2 rows, ntimeblock columns) which is (row 1) alpha: the rate of false positives, and row 2 the power (1-beta) the rate of true positives

Author(s)

T. Carruthers

References

Carruthers and Hordyk 2018

profile_likelihood *Profile likelihood of assessment models*

Description

Profile the likelihood for leading parameters of assessment models.

Usage

```
profile_likelihood(Assessment, figure = TRUE, save_figure = TRUE,
  save_dir = tempdir(), ...)
```

Arguments

Assessment	An S4 object of class Assessment .
figure	Indicates whether a figure will be plotted.
save_figure	Indicates whether figures will be saved to directory.
save_dir	The directory to which figures will be saved. By default: getwd()
...	A sequence of values of the parameter(s) for the profile. See details for name of arguments to be passed on.

Details

For the following assessment models, the required sequence of values are:

- DD_TMB and DD_SS: R0 and h
- SP and SP_SS: UMSY and MSY
- SCA: R0 and h
- SCA2: meanR

Value

A data frame of negative log-likelihood values from the profile and, optionally, a figure of the likelihood surface.

Author(s)

Q. Huynh

Examples

```
output <- DD_TMB(Data = DLMtool::Red_snapper)
pro <- profile_likelihood(output, R0 = seq(0.75, 1.25, 0.025), h = seq(0.9, 0.99, 0.01),
save_figure = FALSE)

# Ensure your grid is of proper resolution. A grid that is too coarse will distort the shape of
# the likelihood surface.
```

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	<i>Reads iSCAM dat file</i>
----------------	-----------------------------

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	<i>Reads iSCAM mcmc output files</i>
-----------	--------------------------------------

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	<i>Reads iSCAM Rep file</i>
------------------	-----------------------------

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)

retrospective	<i>Retrospective analysis of assessment models</i>
---------------	--

Description

Perform a retrospective analysis, successive removals of most recent years of data to evaluate resulting parameter estimates.

Usage

```
retrospective(Assessment, nyr = 5, figure = TRUE, save_figure = TRUE,
  save_dir = tempdir())
```

Arguments

Assessment	An S4 object of class Assessment .
nyr	The maximum number of years to remove for the retrospective analysis.
figure	Indicates whether plots will be drawn.
save_figure	Indicates whether figures will be saved to directory.
save_dir	The directory to which figures will be saved. By default: getwd()

Value

A list with an array of model output and of model estimates from the retrospective analysis. Figures showing the time series of biomass and exploitation and parameter estimates with successive number of years removed. Returns invisibly a list of model output and model estimates.

Author(s)

Q. Huynh

Examples

```
output <- DD_TMB(Data = DLMtool::Red_snapper)
get_retro <- retrospective(output, nyr = 5, figure = FALSE)
```

retrospective_AM *retrospective_AM (retrospective of Assessment model in MSE)*

Description

Plots the true retrospective of an assessment model during the MSE. A series of time series estimates of SSB, F, and VB are plotted over the course of the MSE are plotted against the operating model (true) values (in black).

Usage

```
retrospective_AM(MSE, sim = 1, MP, MSE_Hist = NULL, plot_legend = FALSE)
```

Arguments

MSE	An object of class MSE created by runMSE with PPD = TRUE.
sim	Integer between 1 and MSE@nsim. The simulation number for which the retrospectives will be plotted.
MP	Character. The name of the management procedure created by make_MP containing the assessment model.
MSE_Hist	Optional. The list containing historical data for the MSE, created by runMSE with argument Hist = TRUE. Currently only used to plot operating model vulnerable biomass in historical period.
plot_legend	Logical. Whether to plot legend to reference year of assessment in the MSE.

Details

For assessment models that utilize annual harvest rates (u), the instantaneous fishing mortality rates are obtained as $F = -\log(1 - u)$.

Value

A series of figures for spawning stock biomass (SSB, including absolute magnitude and relative to MSY and virgin), fishing mortality (F, including absolute magnitude and relative to MSY), and vulnerable biomass (VB) estimates over the course of the MSE are plotted against the operating model (true) values (in black).

Note

This function only plots retrospectives from a single simulation in the MSE. Results from one figure may not be indicative of general assessment behavior and performance overall.

For [SP](#) and [SP_SS](#) assessment models don't model SSB. Instead, the estimated vulnerable biomass is plotted.

Author(s)

Q. Huynh

See Also

[diagnostic_AM](#)

Examples

```
## Not run:
DD_MSY <- makeMP(DD_TMB, HCR_MSY, diagnostic = "full")
myMSE_hist <- DLMtool::runMSE(DLMtool::testOM, Hist = TRUE)
myMSE <- DLMtool::runMSE(DLMtool::testOM, MPs = "DD_MSY", PPD = TRUE)
retrospective_AM(myMSE, sim = 1, MP = "DD_MSY")
retrospective_AM(myMSE, sim = 1, MP = "DD_MSY", Hist = myMSE_hist)

## End(Not run)
```

Description

A generic statistical catch-at-age model (single fleet, single season) that uses catch, index, and catch-at-age composition data. An annual harvest rate is calculated (assuming a pulse fishery) as described in Forrest et al. (2008). There are two parameterizations for estimation of recruitment deviations, the stock-recruit relationship, and reference points (see functions section below).

Usage

```
SCA(x = 1, Data, SR = c("BH", "Ricker"), vulnerability = c("logistic",
  "dome"), CAA_dist = c("multinomial", "lognormal"), CAA_multiplier = 50,
  I_type = c("B", "VB", "SSB"), rescale = "mean1", start = NULL,
  fix_h = TRUE, fix_U_equilibrium = TRUE, fix_sigma = FALSE,
  fix_tau = TRUE, early_dev = c("comp_onegen", "comp", "all"),
  late_dev = "comp50", integrate = FALSE, silent = TRUE,
  opt_hess = FALSE, n_restart = ifelse(opt_hess, 0, 1),
  control = list(iter.max = 2e+05, eval.max = 4e+05),
  inner.control = list(), ...)
```

```
SCA2(x = 1, Data, SR = c("BH", "Ricker"), vulnerability = c("logistic",
  "dome"), CAA_dist = c("multinomial", "lognormal"), CAA_multiplier = 50,
  I_type = c("B", "VB", "SSB"), rescale = "mean1", start = NULL,
  fix_h = TRUE, fix_U_equilibrium = TRUE, fix_sigma = FALSE,
  fix_tau = TRUE, common_dev = "comp50", integrate = FALSE,
  silent = TRUE, opt_hess = FALSE, n_restart = ifelse(opt_hess, 0, 1),
  control = list(iter.max = 2e+05, eval.max = 4e+05),
  inner.control = list(), ...)
```

Arguments

<code>x</code>	A position in the Data object (by default, equal to one for assessments).
<code>Data</code>	An object of class Data
<code>SR</code>	Stock-recruit function (either "BH" for Beverton-Holt or "Ricker").
<code>vulnerability</code>	Whether estimated vulnerability is "logistic" or "dome" (double-normal). See details for parameterization.
<code>CAA_dist</code>	Whether a multinomial or lognormal distribution is used for likelihood of the catch-at-age matrix. See details.
<code>CAA_multiplier</code>	Numeric for data weighting of catch-at-age matrix if <code>CAA_hist = "multinomial"</code> . Otherwise ignored. See details.
<code>I_type</code>	Whether the index surveys population biomass (B; this is the default in the DLMtool operating model), vulnerable biomass (VB), or spawning stock biomass (SSB).
<code>rescale</code>	A multiplicative factor that rescales the catch in the assessment model, which can improve convergence. By default, "mean1" scales the catch so that time series mean is 1, otherwise a numeric. Output is re-converted back to original units.
<code>start</code>	Optional list of starting values. See details.
<code>fix_h</code>	Logical, whether to fix steepness to value in <code>Data@steep</code> in the model for SCA. This only affects calculation of reference points for SCA2.
<code>fix_U_equilibrium</code>	Logical, whether the equilibrium harvest rate prior to the first year of the model is estimated. If TRUE, <code>U_equilibrium</code> is fixed to value provided in <code>start</code> (if provided), otherwise, equal to zero (assumes virgin conditions).
<code>fix_sigma</code>	Logical, whether the standard deviation of the index is fixed. If TRUE, <code>sigma</code> is fixed to value provided in <code>start</code> (if provided), otherwise, value based on <code>Data@CV_Ind</code> .
<code>fix_tau</code>	Logical, the standard deviation of the recruitment deviations is fixed. If TRUE, <code>tau</code> is fixed to value provided in <code>start</code> (if provided), otherwise, value based on <code>Data@sigmaR</code> .
<code>early_dev</code>	Character string describing the years for which recruitment deviations are estimated in SCA. By default, "comp_onegen" rec devs are estimated one full generation prior to the first year when catch-at-age (CAA) data are available. With "comp", rec devs are estimated starting in the first year with CAA. With "all", rec devs start at the beginning of the model.

late_dev	Typically, a numeric for the number of most recent years in which recruitment deviations will not be estimated in SCA (recruitment in these years will be based on the mean predicted by stock-recruit relationship). By default, "comp50" uses the number of ages (smaller than the mode) for which the catch-at-age matrix has less than half the abundance than that at the mode.
integrate	Logical, whether the likelihood of the model integrates over the likelihood of the recruitment deviations (thus, treating it as a state-space variable).
silent	Logical, passed to <code>MakeADFun</code> , whether TMB will print trace information during optimization. Used for diagnostics for model convergence.
opt_hess	Logical, whether the hessian function will be passed to <code>nlminb</code> during optimization (this generally reduces the number of iterations to convergence, but is memory and time intensive and does not guarantee an increase in convergence rate). Ignored if <code>integrate = TRUE</code> .
n_restart	The number of restarts (calls to <code>nlminb</code>) in the optimization procedure, so long as the model hasn't converged. The optimization continues from the parameters from the previous (re)start.
control	A named list of arguments for optimization to be passed to <code>nlminb</code> .
inner.control	A named list of arguments for optimization of the random effects, which is passed on to <code>newton</code> .
...	Other arguments to be passed.
common_dev	Typically, a numeric for the number of most recent years in which a common recruitment deviation will be estimated (in SCA2, uninformative years will have a recruitment closer to the mean, which can be very misleading, especially near the end of the time series). By default, "comp50" uses the number of ages (smaller than the mode) for which the catch-at-age matrix has less than half the abundance than that at the mode.

Details

For the statistical catch-at-age model, the basic data inputs are catch (by weight), index (by weight/biomass), and catch-at-age matrix (by numbers). Catches are assumed to be known perfectly (the harvest rate in a given year is the ratio of the observed catch to the vulnerable biomass at the beginning of the year). The maximum age in the model is a plus-group.

By default, steepness is fixed in the model to the value in `Data@steep`.

The annual sample sizes of the catch-at-age matrix is provided to the model (used in the likelihood for catch-at-age, assuming a multinomial distribution), and is manipulated via argument `CAA_multiplier`. This argument is interpreted in two different ways depending on the value provided. If `CAA_multiplier > 1`, then this value will cap the annual sample sizes to that number. If `CAA_multiplier <= 1`, then all the annual samples sizes will be re-scaled by that number. By default, sample sizes are capped at 50.

Alternatively, a lognormal distribution with inverse proportion variance can be used for the catch at age (Punt and Kennedy, 1994, as cited by Maunder 2011).

For `start` (optional), a named list of starting values of estimates can be provided for:

- R_0 Virgin recruitment, only for SCA.

- `h` Steepness, only for SCA. If not provided, the value in `Data@steep` is used.
- `meanR` Mean recruitment, only for SCA2.
- `U_equilibrium` Harvest rate prior to the first year of model, e.g. zero means unfished conditions. Defaults to zero.
- `vul_par` Vulnerability parameters (length 2 vector for logistic or length 4 for dome, see below). Users should provide estimates of the parameters in normal space, e.g. `vul_max` between 0-1, and the function will perform the appropriate transformations for the model.
- `sigma` Standard deviation of index. If not provided, the value based on `Data@CV_Ind` is used.
- `tau` Standard deviation of recruitment deviations. If not provided, the value in `Data@sigmaR` is used.

Vulnerability can be specified to be either logistic or dome. If logistic, then the parameter vector `vul_par` is of length 2:

- `vul_par[1]`: `a_95`, the age of 95% vulnerability, via logit transformation to constrain `a_95` to less than 75% of the maximum age: $a_{95} = 0.75 * \text{max_age} * \text{plogis}(vul_par[1])$.
- `vul_par[2]`: `a_50`, the age of 50% vulnerability as an offset, i.e., $a_{50} = a_{95} - \exp(vul_par[2])$.

A vague prior for `vul_par[2]` $\sim N(0, sd = 3)$ is used to aid convergence, for example, when vulnerability $\gg 0.5$ for the youngest age class.

With dome vulnerability, a double Gaussian parameterization is used, where `vul_par` is an estimated vector of length 4:

- `vul_par[1]`: `a_asc`, the first age of full vulnerability for the ascending limb, via logit transformation to constrain `a_95` to less than 75% of the maximum age: $a_{asc} = 0.75 * \text{maxage} * \text{plogis}(vul_par[1])$.
- `vul_par[2]`: `a_50`, the age of 50% vulnerability for the ascending limb as an offset, i.e., $a_{50} = a_{asc} - \exp(vul_par[2])$.
- `vul_par[3]`: `a_des`, the last age of full vulnerability (where the descending limb starts) via logit transformation to constrain between `a_asc` and `max_age`, i.e., $a_{des} = (\text{max_age} - a_{asc}) * \text{plogis}(vul_par[3])$.
- `vul_par[4]`: `vul_max`, the vulnerability (in logit space) at the maximum age.

Vague priors of `vul_par[2]` $\sim N(0, sd = 3)$ and `vul_par[3]` $\sim N(0, 3)$ are used to aid convergence, for example, when vulnerability $\gg 0.5$ for the youngest age class.

Value

An object of class [Assessment](#).

Functions

- SCA: The parameterization with `R0` and steepness as leading parameters. Recruitment is estimated as deviations from the resulting stock-recruit relationship.
- SCA2: The mean recruitment in the time series is estimated and recruitment deviations around this mean are estimated as penalized parameters (similar to Cadigan 2016). This version is generally very fast and robust. Virgin and MSY reference points are estimated after the assessment run.

Required Data

- SCA: Cat, Ind, Mort, L50, L95, CAA, vbK, vbLinf, vbt0, wla, wlb, MaxAge
- SCA2: Cat, Ind, Mort, L50, L95, CAA, vbK, vbLinf, vbt0, wla, wlb, MaxAge

Optional Data

- SCA: Rec, steep, sigmaR, CV_Ind
- SC2: Rec, steep, CV_Ind

Note

For a given catch history, the starting value of R_0 or meanR may be too low and the population crashes in the middle of the time series. If the assessment function detects this pattern is occurring, it will increase R_0 or meanR before running the model.

Author(s)

Q. Huynh

References

Cadigan, N.G. 2016. A state-space stock assessment model for northern cod, including under-reported catches and variable natural mortality rates. *Canadian Journal of Fisheries and Aquatic Science* 72:296-308.

Forrest, R.E., Martell, S.J.D., Melnychuk, M.C., and Walters, C.J. 2008. An age-structured model with leading management parameters, incorporating age-specific selectivity and maturity. *Canadian Journal of Fisheries and Aquatic Science* 65:286-296.

Maunder, M.N. 2011. Review and evaluation of likelihood functions for composition data in stock-assessment models: Estimating the effective sample size. *Fisheries Research* 209:311-319.

Punt, A.E. and Kennedy, R.B. 1997. Population modelling of Tasmanian rock lobster, *Jasus edwardsii*, resources. *Marine and Freshwater Research* 48:967-980.

See Also

[plot,Assessment,ANY-method summary,Assessment-method retrospective profile_likelihood make_MP](#)

Examples

```
res <- SCA(Data = DLMtool::SimulatedData)
res2 <- SCA2(Data = DLMtool::Simulation_1)
```

 simmov

Calculates movement matrices from user inputs

Description

A wrapper function for [makemov](#) used to generate movement matrices for a DLMtool operating model. Calculates a movement matrix from user-specified unfished stock biomass fraction in each area and probability of staying in the area in each time step.

Usage

```
simmov(OM, dist = c(0.1, 0.2, 0.3, 0.4), prob = 0.5, distE = 0.1,
       probE = 0.1, prob2 = NA, figure = TRUE)
```

```
plot_mov(mov, age = 1, type = c("matrix", "all"))
```

Arguments

OM	Operating model, an object of class OM .
dist	A vector of fractions of unfished stock in each area. The length of this vector will determine the number of areas (nareas) in the OM.
prob	Mean probability of staying across all areas (single value) or a vector of the probability of individuals staying in each area (same length as dist)
distE	Logit (normal) St.Dev error for sampling stock fractions from the frags vector
probE	Logit (normal) St.Dev error for sampling desired probability of staying either by area (prob is same length as dist) or the mean probability of staying (prob is a single number)
prob2	Optional vector as long as prob and dist. Upper bounds on uniform sampling of probability of staying, lower bound is prob.
figure	Logical to indicate if the movement matrix will be plotted (mean values and range across OM@nsim simulations.)
mov	A four-dimensional array of dimension c(nsim, maxage, nareas, nareas) specifying movement in the operating model.
age	An age from 1 to maxage for the movement-at-age matrix figure when type = "matrix".
type	Whether to plot a movement matrix for a single age ("matrix") or the full movement versus age figure ("all")

Value

The operating model OM with movement parameters in slot cpars. The mov array is of dimension nsim, maxage, nareas, nareas.

Functions

- `simmov`: Estimation function for creating movement matrix.
- `plot_mov`: Plotting function.

Note

Array `mov` is age-specific, but currently the movement generated by `simmov` is independent of age.

Author(s)

T. Carruthers and Q. Huynh

Examples

```
movOM_5areas <- simmov(testOM, dist = c(0.01,0.1,0.2,0.3,0.39), prob = c(0.1,0.6,0.6,0.7,0.9))
movOM_5areas@cpars$mov[1, 1, , ] # sim 1, age 1, movement from areas in column i to areas in row j
plot_mov(movOM_5areas@cpars$mov)
plot_mov(movOM_5areas@cpars$mov, type = "all")
```

Description

A surplus production model that estimates the TAC using a time-series of catches and a relative abundance index and coded in TMB. The model is conditioned on catch and estimates a predicted index. The state-space version estimates annual deviates in biomass.

Usage

```
SP(x = 1, Data, rescale = "mean1", start = NULL, fix_dep = TRUE,
  fix_n = TRUE, silent = TRUE, opt_hess = FALSE,
  n_restart = ifelse(opt_hess, 0, 1), control = list(iter.max = 5000,
  eval.max = 10000), ...)
```

```
SP_SS(x = 1, Data, rescale = "mean1", start = NULL, fix_dep = TRUE,
  fix_n = TRUE, fix_sigma = TRUE, fix_tau = TRUE, early_dev = c("all",
  "index"), integrate = FALSE, silent = TRUE, opt_hess = FALSE,
  n_restart = ifelse(opt_hess, 0, 1), control = list(iter.max = 5000,
  eval.max = 10000), inner.control = list(), ...)
```

Arguments

<code>x</code>	An index for the objects in <code>Data</code> when running in closed loop simulation. Otherwise, equals to 1 When running an assessment interactively.
<code>Data</code>	An object of class <code>Data</code> .
<code>rescale</code>	A multiplicative factor that rescales the catch in the assessment model, which can improve convergence. By default, "mean1" scales the catch so that time series mean is 1, otherwise a numeric. Output is re-converted back to original units.
<code>start</code>	Optional list of starting values. See details.
<code>fix_dep</code>	Logical, whether to fix the initial depletion (ratio of biomass to carrying capacity in the first year of the model). If TRUE, uses the value in <code>start</code> , otherwise equal to 1 (assumes virgin conditions).
<code>fix_n</code>	Logical, whether to fix the exponent of the production function. If TRUE, uses the value in <code>start</code> , otherwise equal to $n = 2$, where the biomass at MSY is half of carrying capacity.
<code>silent</code>	Logical, passed to <code>MakeADFun</code> , whether TMB will print trace information during optimization. Used for diagnostics for model convergence.
<code>opt_hess</code>	Logical, whether the hessian function will be passed to <code>nlminb</code> during optimization (this generally reduces the number of iterations to convergence, but is memory and time intensive and does not guarantee an increase in convergence rate). Ignored if <code>integrate = TRUE</code> .
<code>n_restart</code>	The number of restarts (calls to <code>nlminb</code>) in the optimization procedure, so long as the model hasn't converged. The optimization continues from the parameters from the previous (re)start.
<code>control</code>	A named list of parameters regarding optimization to be passed to <code>nlminb</code> .
<code>...</code>	Additional arguments (not currently used).
<code>fix_sigma</code>	Logical, whether the standard deviation of the index is fixed. If TRUE, sigma is fixed to value provided in <code>start</code> (if provided), otherwise, value based on <code>Data@CV_Ind</code> .
<code>fix_tau</code>	Logical, the standard deviation of the biomass deviations is fixed. If TRUE, tau is fixed to value provided in <code>start</code> (if provided), otherwise, equal to 1.
<code>early_dev</code>	Character string describing the years for which biomass deviations are estimated in <code>SP_SS</code> . By default, deviations are estimated in each year of the model ("all"), while deviations could also be estimated once index data are available ("index").
<code>integrate</code>	Logical, whether the likelihood of the model integrates over the likelihood of the biomass deviations (thus, treating it as a state-space variable).
<code>inner.control</code>	A named list of arguments for optimization of the random effects, which is passed on to <code>newton</code> via <code>MakeADFun</code> .

Details

To provide starting values for the SP, a named list can be provided for UMSY, MSY, dep, and n via the `start` argument (see example).

For SP_SS, a start value can also be provided for sigma and tau, the standard deviation of the index and log-biomass deviates, respectively. Deviations are estimated beginning in the year when index data are available.

Value

An object of [Assessment](#) containing objects and output from TMB.

Functions

- SP: Fixed effects model
- SP_SS: State-space version

Required Data

- SP: Cat, Ind
- SP_SS: Cat, Ind

Optional Data

SP_SS: CV_Ind

Note

The model uses the Fletcher (1978) formulation and is parameterized with UMSY and MSY as leading parameters. The default conditions assume virgin conditions in the first year of the time series and a symmetric production function.

Author(s)

Q. Huynh

References

- Fletcher, R. I. 1978. On the restructuring of the Pella-Tomlinson system. Fishery Bulletin 76:515:521.
- Pella, J. J. and Tomlinson, P. K. 1969. A generalized stock production model. Inter-Am. Trop. Tuna Comm., Bull. 13:419-496.

See Also

[SP_production](#)

[plot,Assessment,ANY-method summary,Assessment-method retrospective profile_likelihood make_MP](#)

Examples

```

data(swordfish)

#### Observation-error surplus production model
res <- SP(Data = swordfish)

# Provide starting values, assume B/K = 0.95 in first year of model
# and symmetrical production curve (n = 2)
start <- list(UMSY = 0.1, MSY = 1e5, dep = 0.95, n = 2)
res <- SP(Data = swordfish, start = start)

#### State-space version
res <- SP_SS(Data = swordfish, start = list(dep = 0.95, tau = 0.3),
fix_sigma = TRUE)

```

SP_production

Find the production parameter based on depletion that produces MSY

Description

For surplus production models, this function returns the production exponent n corresponding to $BMSY/K$ (Fletcher 1978).

Usage

```
SP_production(depletion, figure = TRUE)
```

Arguments

`depletion` The hypothesized depletion that produces MSY.
`figure` Local, plots figure of production function as a function of depletion (B/K)

Value

The production function exponent n (numeric).

Note

May be useful for parameterizing n in [SP](#) and [SP_SS](#).

Author(s)

Q. Huynh

References

Fletcher, R. I. 1978. On the restructuring of the Pella-Tomlinson system. *Fishery Bulletin* 76:515:521.

See Also[SP SP_SS](#)**Examples**

```
SP_production(0.5)
SP_production(0.5)
```

SRopt	<i>Function that returns a stochastic estimate of steepness given observed stock recruitment data</i>
-------	---

Description

Function that returns a stochastic estimate of steepness given observed stock recruitment data

Usage

```
SRopt(nsim, SSB, rec, SSBpR, plot = FALSE, type = c("BH", "Ricker"))
```

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 or "Ricker")

Value

Vector of length nsim with steepness values.

Author(s)

T. Carruthers

SS2Data	<i>Reads data Stock Synthesis file structure into a Data object using package r4ss</i>
---------	--

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, Name = NULL, Common_Name = "", Species = "", Region = "",
        min_age_M = 1, comp_fleet = "all", comp_season = "sum",
        comp_partition = "all", comp_gender = "all", index_fleet = "SSB",
        index_season = "mean", ...)
```

Arguments

SSdir	A folder with Stock Synthesis input and output files in it
Name	The name for the Data object
Common_Name	Character string for the common name of the stock.
Species	Scientific name of the species
Region	Geographic region of the stock or fishery.
min_age_M	Currently, the Data object supports a single value of M for all ages. The argument selects the minimum age for calculating the mean of age-dependent M from the SS assessment.
comp_fleet	A vector of indices corresponding to fleets in the assessment over which to aggregate the composition (catch-at-length and catch-at-age) data. By default, character string "all" will aggregate across all fleets.
comp_season	Integer, for seasonal models, the season for which the value of the index will be used. By default, "mean" will take the average across seasons.
comp_partition	Integer vector for selecting length/age observations that are retained (2), discarded (1), or both (0). By default, "all" sums over all available partitions.
comp_gender	Integer vector for selecting length/age observations that are female (1), male (2), or both (0), or both scaled to sum to one (3). By default, "all" sums over all gender codes.
index_fleet	Integer for selecting the fleet of the index to put in the Data object. By default, "SSB" will use the relative trend in spawning stock biomass as estimated in the model as the index.
index_season	Integer, for seasonal models, the season for which the value of the index will be used. By default, "mean" will take the average across seasons.
...	Arguments to pass to SS_output

Value

An object of class Data.

Note

Currently supports the latest version of r4ss on CRAN (v.1.24). Function may be incompatible with newer versions of r4ss on Github.

Author(s)

T. Carruthers

See Also

[SS2OM](#)

SS2OM	<i>Reads MLE estimates from Stock Synthesis file structure into an operating model using package r4ss.</i>
-------	--

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. The function mainly populates the Stock and Fleet portions of the operating model; the user still needs to parameterize most of the observation and implementation portions of the operating model.

Usage

```
SS2OM(SSdir, nsim = 48, proyears = 50, reps = 1, maxF = 3, seed = 1,
      Obs = DLMtool::Generic_Obs, Imp = DLMtool::Perfect_Imp, Name = NULL,
      Source = "No source provided", Author = "No author provided", ...)
```

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@cparams custom parameters).
proyears	The number of projection years for MSE
reps	The number of stochastic replicates within each simulation in the operating model.
maxF	The maximum allowable F in the operating model.
seed	The random seed for the operating model.
Obs	The observation model (class Obs).
Imp	The implementation model (class Imp).

Name	The name of the operating model
Source	Reference to assessment documentation e.g. a url
Author	Who did the assessment
...	Arguments to pass to SS_output .

Details

Currently, the function uses values from the terminal year of the assessment for most life history parameters (growth, maturity, M, etc). Steepness values for the operating model are obtained by bootstrapping from spawning biomass and recruitment estimates in the assessment.

Value

An object of class OM.

Note

Currently supports the latest version of r4ss on CRAN (v.1.24). Function may be incompatible with newer versions of r4ss on Github.

Author(s)

T. Carruthers

See Also

[SS2Data](#)

summary,Assessment-method

Summary of Assessment object

Description

Summary of Assessment object

Usage

```
## S4 method for signature 'Assessment'
summary(object)
```

Arguments

object An object of class [Assessment](#)

Value

A list of parameters

Examples

```
output <- DD_TMB(Data = DLMtool::Simulation_1)
summary(output)
```

swordfish

North Atlantic Swordfish dataset

Description

An S4 object containing catch and index time series for North Atlantic swordfish.

Usage

```
swordfish
```

Format

An object of class Data.

Source

ASPIC Software at <https://www.mhprager.com/aspic.html>

Examples

```
data(swordfish)
```

Index

- *Topic **datasets**
 - swordfish, 59
- *Topic **evaluation**
 - MSEtool-package, 3
- *Topic **fisheries**
 - MSEtool-package, 3
- *Topic **management**
 - MSEtool-package, 3
- *Topic **strategy**
 - MSEtool-package, 3
- Assessment, 11, 18, 19, 29, 38, 39, 43, 48, 53, 58
- Assessment (Assessment-class), 4
- Assessment-class, 4
- avail, 6, 6
- Awatea20M, 7
- calculate_TAC, 7
- Data, 10
- Data-rich-MP, 8
- DD_SS, 9
- DD_SS (DD_TMB), 10
- DD_TMB, 10
- DDSS_4010 (Data-rich-MP), 8
- DDSS_75MSY (Data-rich-MP), 8
- DDSS_MSYS (Data-rich-MP), 8
- diagnostic_AM, 12, 26, 45
- DLMenv, 26
- fetch.file.names, 13
- getGpars, 14
- getinds, 14
- getSR, 15
- HCR40_10, 8
- HCR40_10 (HCR_ramp), 19
- HCR60_20, 8
- HCR60_20 (HCR_ramp), 19
- HCR_FB, 17
- HCR_MSYS, 8, 18
- HCR_ramp, 19
- HCRlin, 16, 20
- iSCAM2Data, 21
- iSCAM20M, 21
- iSCAMcomps, 22
- iSCAMinds, 23
- LinInterp, 23
- load.iscam.files, 24
- mahplot, 24
- make_MP, 6, 8, 12, 18–20, 26, 44, 49, 53
- MakeADFun, 5, 10, 11, 47, 52
- makemov, 25, 50
- movdistil, 27
- MSEtool (MSEtool-package), 3
- MSEtool-package, 3
- negcorlogspace, 27
- newton, 11, 47, 52
- nllminb, 5, 10, 11, 13, 47, 52
- OM, 50
- optSR, 15, 28
- plot, Assessment, ANY-method, 6, 12, 28, 49, 53
- plot_betavar, 29, 32, 35
- plot_composition, 30
- plot_crosscorr, 31
- plot_lognormalvar, 29, 32, 35
- plot_mov (simmov), 50
- plot_residuals, 33, 36
- plot_SR, 34
- plot_steepness, 29, 32, 34
- plot_timeseries, 33, 35
- PRBcalc, 24, 36, 38
- prelim_AM, 37

Probs, [37](#), [38](#)
profile_likelihood, [6](#), [12](#), [39](#), [49](#), [53](#)

read.control.file, [40](#)
read.data.file, [41](#)
read.mcmc, [41](#)
read.par.file, [42](#)
read.projection.file, [42](#)
read.report.file, [43](#)
Rec, [18](#), [19](#)
retrospective, [6](#), [12](#), [43](#), [49](#), [53](#)
retrospective_AM, [13](#), [26](#), [44](#)
runMSE, [13](#), [26](#), [44](#)

SCA, [9](#), [45](#)
SCA2 (SCA), [45](#)
SCA_4010 (Data-rich-MP), [8](#)
SCA_75MSY (Data-rich-MP), [8](#)
SCA_MSY (Data-rich-MP), [8](#)
sdreport, [5](#)
simmov, [25](#), [50](#)
SP, [9](#), [20](#), [45](#), [51](#), [54](#), [55](#)
SP_4010 (Data-rich-MP), [8](#)
SP_75MSY (Data-rich-MP), [8](#)
SP_MSY (Data-rich-MP), [8](#)
SP_production, [53](#), [54](#)
SP_SS, [20](#), [45](#), [54](#), [55](#)
SP_SS (SP), [51](#)
SRopt, [55](#)
SS2Data, [56](#), [58](#)
SS20M, [57](#), [57](#)
SS_output, [56](#), [58](#)
summary, Assessment-method, [6](#), [12](#), [49](#), [53](#),
[58](#)
swordfish, [59](#)