Package ‘TropFishR’

May 1, 2025

Type Package

Title Tropical Fisheries Analysis

Version 1.6.6

Date 2025-04-06

Depends R (>= 3.0.0)

Imports Matrix, msm, reshape2, MASS, GenSA, GA, stats, utils, parallel, doParallel

LazyData true

Description A compilation of fish stock assessment methods for the analysis of length-frequency data in the context of data-poor fisheries. Includes methods and examples included in the FAO Manual by P. Sparre and S.C. Venema (1998), ``Introduction to tropical fish stock assessment'' (<https://openknowledge.fao.org/server/api/core/bitstreams/bc7c37b6-30df-49c0-b5b4-8367a872c97e/content>), as well as other more recent methods.

License GPL-3

NeedsCompilation no

BugReports https://github.com/tokami/TropFishR/issues

URL https://github.com/tokami/TropFishR

Encoding UTF-8

RoxygenNote 7.3.2

Suggests graphics, grDevices, knitr, rmarkdown, plyr

VignetteBuilder knitr

Author Tobias K. Mildenberger [aut, cre] (ORCID: <https://orcid.org/0000-0002-6631-7524>), Marc H. Taylor [aut], Matthias Wolff [aut]

Maintainer Tobias K. Mildenberger <t.k.mildenberger@gmail.com>

Repository CRAN

Date/Publication 2025-05-01 15:40:02 UTC

Contents

`alba` *Length-frequency data of the clam Abra alba*

Description

Length-frequency data of the clam Abra alba as presented by Brey et al. (1988). Includes 7 approximately bi-monthly samplings of A. alba binned into 14 length classes. The data is of class "lfq", which can be used used e.g. in functions estimating growth parameters: `ELEFAN`, `ELEFAN_SA` or `ELEFAN_GA`.

Usage

`data(alba)`

Format

A list of class lfq consisting of:

- **dates** dates of sampling times (class Date),
- **midLengths** midpoints of the length classes,
- **catch** matrix with catches/counts per length class (row) and sampling date (column).

Source

Brey, T., Soriano, M., and Pauly, D. 1988. Electronic length frequency analysis: a revised and expanded user's guide to ELEFAN 0, 1 and 2. ICLARM Contrib. 261.

Examples

```
data(alba)

# plot raw catch frequencies
plot(alba, Fname = "catch")

# plot restructured frequencies
alba <- lfqRestructure(alba, MA=5)
plot(alba, Fname = "rcounts")


# ELEFAN_SA fitting
set.seed(1)
fitSA <- ELEFAN_SA(
  alba, seasonalised = TRUE,
  init_par = list(Linf=14.5, K=1.1, t_anchor=0.4, ts=0, C=0.2),
  low_par = list(Linf=13, K=0.7, t_anchor=0, ts=0, C=0),
  up_par = list(Linf=15.5, K=1.5, t_anchor=1, ts=1, C=1),
  SA_time = 60
)
unlist(fitSA$par)
fitSA$Rn_max

# plot ELEFAN_SA results
plot(alba, Fname = "catch", draw = FALSE)
lfqFitCurves(fitSA, col=2, par=fitSA$par, draw=TRUE)$Rn_max
```

`Bhattacharya` *Bhattacharya's method*

Description

Find relative frequencies and frequency distribution of distinct cohorts in the observed length frequency distribution by resolving it into Gaussian components.

Usage

```
Bhattacharya(param, n_rnorm = 1000, savePlots = FALSE)
```

Arguments

`param`	a list consisting of following parameters: • `midLengths` midpoints of the length class as vector, • `catch` a vector with the catch per length class or a matrix with catches per length class of subsequent years;
`n_rnorm`	number of observations for the function `rnorm`. The default is 1000.
`savePlots`	logical; indicating whether the analyis graphs should be recorded

Details

This method includes the `identify` function, which allows to choose points on a graphical device manually. To stop this process please press right mouse click on the graph, and in case you are working on a windows maschine click on "Stop". An error will be caused if the graphical device is closed manually. After you have selected the points for regression analysis you will be asked if you want to redo the selection or if you want to continue. Please enter in the Console "y" for continuation if you are satisfied with your selection and the corresponding Gaussian distribution or enter "redo" if you want to repeat the selection procedure. This function allows a maximum of 12 cohorts or seperate distributions in one sample. Please find more details in the Vignette of this package or in the FAO manual by Sparre and Venema (1998) (**Section 3.4.1, p. 80**).

Value

A list with the input parameters and

- **regressionLines** dataframe with intercept, slope, start and end points of the regression lines,
- **Lmean_SD_list** dataframe with the mean length (Lmean), standard deviation (SD), and seperation index (SI) for each cohort,
- **bhat_results** dataframe with the results of the Bhattacharya method,
- **distributions** list with the x and y values of selected distributions,
- **cohort_plots** list with analysis plots (when savePlots = TRUE).

References

Bhattacharya, C.G., 1967. A simple method of resolution of a distribution into Gaussian components, *Biometrics*, 23:115-135

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
# The following example requires to choose certain values for the regression analyses:
# first cohort:   point  2 to  8
# second cohort:  point 12 to 17
# third cohort:   point 19 to 23
# fourth cohort:  point 26 to 30


 data(synLFQ1)
 Bhattacharya(param = synLFQ1)
```

`bream` *bream data*

Description

Data of a covered codend experimental catch of the species Threadfin bream (*Nemipterus japonicus*) in South China Sea. This data can be analysed with the function `select`.

Usage

```
data(bream)
```

Format

A list consiting of:

- `midLengths` the midlengths of size classes,
- `numCodend` the number of fish retained in codend,
- `numCover` the number of fish retained in cover,
- `msCodend` the meshsize of codend.

Source

Jones, R., 1976. Mesh regulation in the demersal fisheries of the South China Sea area. Manila, South China Sea Fisheries Development and Coordinating Programme, SCS/76/WP/34: 75 p.

Examples

```
data(bream)
str(bream)
summary(bream)
```

`catchCurve` *Catch curve*

Description

This function applies the (length-converted) linearised catch curve to age composition and length-frequency data, respectively. It allows to estimate the instantaneous total mortality rate (Z). Optionally, the gear selectivity can be estimated and the cumulative catch curve cna be applied.

Usage

```
catchCurve(
  param,
  catch_columns = NA,
  cumulative = FALSE,
  calc_ogive = FALSE,
  reg_int = NULL,
  reg_num = 1,
  auto = FALSE,
  plot = TRUE
)
```

Arguments

`param`	a list consisting of following parameters: • `midLengths` or `age`: midpoints of the length classes (length-frequency data) or ages (age composition data), • `Linf`: infinite length for investigated species in cm [cm], • `K`: growth coefficent for investigated species per year [1/year], • `t0`: theoretical time zero, at which individuals of this species hatch, • `catch`: catches, vector or matrix with catches of subsequent years if the catch curve with constat time intervals should be applied;
`catch_columns`	numerical; indicating the column of the catch matrix which should be used for the analysis.
`cumulative`	logical; if TRUE the cumulative catch curve is applied (Jones and van Zalinge method)
`calc_ogive`	logical; if TRUE the selection ogive is additionally calculated from the catch curve (only if `cumulative` = `FALSE`)
`reg_int`	instead of using the identity method a range can be determined, which is to be used for the regression analysis. If equal to NULL identity method is applied (default). For multiple regression lines provide list with the two points for the regression line in each element of the list.
`reg_num`	integer indicating how many separate regression lines should be applied to the data. Default 1.
`auto`	logical; no interactive functions used instead regression line is chosen automatically. Default = FALSE
`plot`	logical; should a plot be displayed? Default = TRUE

Details

This function includes the identify function, which asks you to choose two points from a graph manually. The two points which you choose by clicking on the plot in the graphical device represent the start and end of the data points, which should be used for the analysis. Based on these points the regression line is calculated. When the selection ogive is calculated by means of the catch curve the assumption is made, that Z is constant for all year classes or length groups, respectively. Accoring to Sparre and Venema (1998) this assumption might be true, because F is smaller for young fish (Selectivity) while M is higher for young fish (high natural mortality). The selectivity for not fully exploited old fish (e.g. due to gillnet fishery) can not be calculated yet by use of the catch curve. Based on the format of the list argument `catch` and whether the argument `catch_columns` is defined, the function automatically distinguishes between the catch curve with variable parameter system (if catch is a vector) and the one with constant parameter system (if catch is a matrix or a data.frame and `catch_columns = NA`). In the case of the variable parameter system the catches of one year are assumed to represent the catches during the entire life span of a so called pseudo-cohort. The cumulative catch curve does not allow for the estimation of the selectivity ogive.

Value

A list with the input parameters and following list objects:

- **classes.num**, **tplusdt_2**, **t_midL**, or **ln_Linf_L**: age, relative age or subsitute depending on input and method,
- **lnC** or **lnC_dt**: logarithm of (rearranged) catches,
- **reg_int**: the interval used for the regression analysis,
- **linear_mod**: linear model used for the regression analysis,
- **Z**: instantaneous total mortality rate, confidenceInt
- **se**: standard error of the total mortality;
- **confidenceInt**: confidence interval of the total mortality;

in case calc_ogive == TRUE, additionally:

- **intercept**: intercept of regression analysis,
- **linear_mod_sel**: linear model used for the selectivity analysis,
- **Sobs**: observed selection ogive,
- **ln_1_S_1**: dependent variable of regression analysis for selectivity parameters,
- **Sest**: estimated selection ogive,
- **t50**: age at first capture (age at which fish have a 50 probability to be caught),
- **t75**: age at which fish have a 75
- **L50**: length at first capture (length at which fish have a 50 probability to be caught),
- **L75**: length at which fish have a 75

References

Baranov, F.I., 1926. On the question of the dynamics of the fishing industry. *Nauchn. Byull. Rybn. Khoz*, 8 (1925), 7-11

Beverton, R.J.H. and S.J. Holt, 1956. A review of methods for estimating mortality rates in exploited fish populations, with special reference to sources of bias in catch sampling. *Rapports et Proces verbaux des Reunions*, Conseil Table3

Chapman, D., and D.S Robson, 1960. The analysis of a catch curve. *Biometrics*, 354-368

Edser, T., 1908. Note on the number of plaice at each length, in certain samples from the southern part of the North Sea, 1906. *Journal of the Royal Statistical Society*, 686-690

Heincke, F., 1913. Investigations on the plaice. General report. 1. The plaice fishery and protective regulations. Part I. *Rapp.P.-v.Reun.CIEM*, 17A:1-153 + Annexes

ICES, 1981. Report of the *Ad hoc* working group on the use of effort data in assessment, Copenhagen, 2-6 March 1981. *ICES C.M.* 1981/G:5 (mimeo)

Jones, R., and N.P. Van Zalinge, 1981. Estimates of mortality rate and population size for shrimp in Kuwait waters. *Kuwait Bull. Mar. Sci*, 2, 273-288

Pauly, D., 1983. Length-converted catch curves: a powerful tool for fisheries research in the tropics (part I). *ICLARM Fishbyte*, 1(2), 9-13

Pauly, D., 1984. Length-converted catch curves: a powerful tool for fisheries research in the tropics (part II). *ICLARM Fishbyte*, 2(1), 17-19

Pauly, D., 1984. Length-converted catch curves: a powerful tool for fisheries research in the tropics (III: Conclusion). *ICLARM Fishbyte*, 2(3), 9-10

Ricker, W.E., 1987. Computation and interpretation of biological statistics of fish populations. *Bull.Fish.Res.Board Can.*, (191):382 p.

Robson, D.S., and D.G. Chapman, 1961. Catch curves and mortality rates. *Trans.Am.Fish.Soc.*, 90(2):181-189

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Van Sickle, J. 1977. Mortality rates from size distributions: the application of a conservation law. *Oecologia, Berl.*, 27(4):311-318

Examples

```
#______________________________________________
# Variable paramter system (with catch vector)
# based on length frequency data
data(goatfish)
output <- catchCurve(goatfish)
summary(output$linear_mod)

# based on age composition data
data(whiting)
catchCurve(whiting, catch_columns = 1)

#______________________________________________
# Constant parameter system based on age composition data (with catch matrix)
catchCurve(whiting)

#______________________________________________
# Cumulative Catch Curve
# based on length frequency data
data(goatfish)
catchCurve(goatfish, cumulative = TRUE)

# based on age composition data
data(synCAA2)
catchCurve(synCAA2, cumulative = TRUE)
```

```
#_______________________________________________
# Catch Curve with estimation of selection ogive
data(synLFQ3)
output <- catchCurve(synLFQ3, calc_ogive = TRUE)
summary(output$linear_mod_sel)


# the same with predefined selection for regression line:
data(synLFQ3)
output <- catchCurve(synLFQ3, calc_ogive = TRUE, reg_int = c(9,21))
plot(output, plot_selec = TRUE)
```

`date2yeardec` *Date - Year conversion*

Description

Convert dates to numeric years with decimal as fraction of a year

Usage

```
date2yeardec(date)
```

Arguments

`date`	a date (class 'Date')

Value

a scalar (class 'numeric')

Examples

```
date2yeardec(Sys.Date())
```

`ELEFAN` *ELEFAN*

Description

Electronic LEngth Frequency ANalysis for estimating growth parameter.

Usage

```
ELEFAN(
  lfq,
  Linf_fix = NA,
  Linf_range = NA,
  K_range = exp(seq(log(0.1), log(10), length.out = 100)),
  C = 0,
  ts = 0,
  MA = 5,
  addl.sqrt = FALSE,
  agemax = NULL,
  flagging.out = TRUE,
  method = "optimise",
  cross.date = NULL,
  cross.midLength = NULL,
  cross.max = FALSE,
  hide.progressbar = FALSE,
  plot = FALSE,
  contour = FALSE,
  add.values = TRUE,
  rsa.colors = terrain.colors(20),
  plot_title = TRUE
)
```

Arguments

`lfq`	a list consisting of following parameters: • **midLengths** midpoints of the length classes, • **dates** dates of sampling times (class Date), • **catch** matrix with catches/counts per length class (row) and sampling date (column);
`Linf_fix`	numeric; if used the K-Scan method is applied with a fixed Linf value (i.e. varying K only).
`Linf_range`	numeric vector with potential Linf values. Default is the last length class plus/minus 5 cm
`K_range`	K values for which the score of growth functions should be calculated (by default: exp(seq(log(0.1),log(10),length.out = 100)))
`C`	growth oscillation amplitude (default: 0)
`ts`	onset of the first oscillation relative to t0 (summer point, default: 0)
`MA`	number indicating over how many length classes the moving average should be performed (default: 5, for more information see lfqRestructure).
`addl.sqrt`	Passed to lfqRestructure. Applied an additional square-root transformation of positive values according to Brey et al. (1988). (default: FALSE, for more information see lfqRestructure).
`agemax`	maximum age of species; default NULL, then estimated from Linf

`flagging.out`	logical; should positive peaks be flagged out? (Default : TRUE)
`method`	Choose between the old FiSAT option to force VBGF crossing of a pre-defined bin (method = "cross"), or the more sophisticated (but computationally expensive) option to solve for t_anchor via a maximisation of reconstructed score (default: method = "optimise").
`cross.date`	Date. For use with `method = "cross"`. In combination with `cross.midLength`, defines the date of the crossed bin.
`cross.midLength`	Numeric. For use with `method = "cross"`. In combination with `cross.date`, defines the mid-length of the crossed bin.
`cross.max`	logical. For use with `method = "cross"`. Forces growth function to cross the bin with maximum positive score.
`hide.progressbar`	logical; should the progress bar be hidden? (default: FALSE)
`plot`	logical; indicating if plot with restructured frequencies and growth curves should be displayed
`contour`	if used in combination with response surface analysis, contour lines are displayed rather than the score as text in each field of the score plot. Usage can be logical (e.g. TRUE) or by providing a numeric which indicates the number of levels (`nlevels` in `contour`). By default FALSE.
`add.values`	logical. Add values to Response Surface Analysis plot (default: TRUE). Overridden when `contour = TRUE`.
`rsa.colors`	vector of colors to be used with the Response Surface Analysis plot. (default: terrain.colors(20))
`plot_title`	logical; indicating whether title to score plots should be displayed

Details

This functions allows to perform the K-Scan and Response surface analysis to estimate growth parameters. It combines the step of restructuring length-frequency data (lfqRestructure) followed by the fitting of VBGF curves through the restructured data (lfqFitCurves). K-Scan is a method used to search for the K parameter with the best fit while keeping the Linf fixed. In contrast, with response surface analysis both parameters are estimated and the fits are displayed in a heatmap. Both methods use `optimise` to find the best `t_anchor` value for each combination of `K` and `Linf`. To find out more about `t_anchor`, please refer to the Details description of `lfqFitCurves`. The score value `Rn_max` is comparable with the score values of the other ELEFAN functions (`ELEFAN_SA` or `ELEFAN_GA`) when other settings are consistent (e.g. ‘MA‘, ‘addl.sqrt‘, ‘agemax‘, ‘flagging.out‘).

Value

A list with the input parameters and following list objects:

- **rcounts**: restructured frequencies,
- **peaks_mat**: matrix with positive peaks with distinct values,
- **ASP**: available sum of peaks, sum of posititve peaks which could be potential be hit by growth curves,

- **score_mat**: matrix with scores for each Linf (only Linf_fix) and K combination,
- **t_anchor_mat**: maximum age of species,
- **ncohort**: number of cohorts used for estimation,
- **agemax**: maximum age of species,
- **par**: a list with the parameters of the von Bertalanffy growth function:
  - **Linf**: length infinity in cm,
  - **K**: curving coefficient;
  - **t_anchor**: time point anchoring growth curves in year-length coordinate system, corrsponds to peak spawning month,
  - **C**: amplitude of growth oscillation (if `seasonalised` = TRUE),
  - **ts**: summer point of oscillation (ts = WP - 0.5) (if `seasonalised` = TRUE),
  - **phiL**: growth performance index defined as phiL = log10(K) + 2 * log10(Linf);
- **Rn_max**: highest score value

References

Brey, T., Soriano, M., and Pauly, D. 1988. Electronic length frequency analysis: a revised and expanded user's guide to ELEFAN 0, 1 and 2.

Pauly, D. 1981. The relationship between gill surface area and growth performance in fish: a generalization of von Bertalanffy's theory of growth. *Meeresforsch.* 28:205-211

Pauly, D. and N. David, 1981. ELEFAN I, a BASIC program for the objective extraction of growth parameters from length-frequency data. *Meeresforschung*, 28(4):205-211

Pauly, D., 1985. On improving operation and use of ELEFAN programs. Part I: Avoiding "drift" of K towards low values. *ICLARM Conf. Proc.*, 13-14

Pauly, D., 1987. A review of the ELEFAN system for analysis of length-frequency data in fish and aquatic invertebrates. *ICLARM Conf. Proc.*, (13):7-34

Pauly, D. and G. R. Morgan (Eds.), 1987. Length-based methods in fisheries research. (No. 13). WorldFish

Pauly, D. and G. Gaschuetz. 1979. A simple method for fitting oscillating length growth data, with a program for pocket calculators. I.C.E.S. CM 1979/6:24. Demersal Fish Cttee, 26 p.

Pauly, D. 1984. Fish population dynamics in tropical waters: a manual for use with programmable calculators (Vol. 8). WorldFish.

Quenouille, M. H., 1956. Notes on bias in estimation. *Biometrika*, 43:353-360

Somers, I. F., 1988. On a seasonally oscillating growth function. ICLARM Fishbyte 6(1): 8-11.

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2): 407 p.

Tukey, J., 1958. Bias and confidence in not quite large samples. *Annals of Mathematical Statistics*, 29: 614

Tukey, J., 1986. The future of processes of data analysis. In L. V. Jones (Eds.), The Collected Works of John W. Tukey-philosophy and principles of data analysis: 1965-1986 (Vol. 4, pp. 517-549). Monterey, CA, USA: Wadsworth & Brooks/Cole

Examples

```
data(alba)

### Response Surface Analysis (varies Linf and K) ###

# 'cross' method (used in FiSAT)
fit1 <- ELEFAN(
   lfq = alba, method = "cross",
   Linf_range = seq(from = 10, to = 20, length.out = 10),
   K_range = exp(seq(from = log(0.1), to = log(2), length.out = 20)),
   cross.date = alba$dates[3], cross.midLength = alba$midLengths[4],
   contour = TRUE
)
fit1$Rn_max; unlist(fit1$par)
plot(fit1); points(alba$dates[3], alba$midLengths[4], col=2, cex=2, lwd=2)

# 'cross' method (bin with maximum score crossed)
fit2 <- ELEFAN(
   lfq = alba, method = "cross",
   Linf_range = seq(from = 10, to = 20, length.out = 20),
   K_range = exp(seq(from = log(0.1), to = log(2), length.out = 20)),
   cross.max = TRUE,
   contour = TRUE
)
fit2$Rn_max; unlist(fit2$par)
plot(fit2); points(alba$dates[7], alba$midLengths[9], col=2, cex=2, lwd=2)


# 'optimise' method (default)
fit3 <- ELEFAN(
   lfq = alba, method = "optimise",
   Linf_range = seq(from = 10, to = 20, length.out = 10),
   K_range = exp(seq(from = log(0.1), to = log(2), length.out = 20)),
   contour = TRUE
)
fit3$Rn_max; unlist(fit3$par)
plot(fit3)


### K-Scan (varies K, Linf is fixed) ###

# 'cross' method
fit4 <- ELEFAN(
   lfq = alba, method = "cross",
   Linf_fix = 10,
   K_range = round(exp(seq(from = log(0.1), to = log(2), length.out = 50)),2),
   cross.date = alba$dates[3], cross.midLength = alba$midLengths[4],
   plot = FALSE
)
fit4$Rn_max; unlist(fit4$par)
plot(fit4); points(alba$dates[3], alba$midLengths[4], col=2, cex=2, lwd=2)
```

`ELEFAN_GA` *ELEFAN_GA*

Description

Electronic LEngth Frequency ANalysis with genetic algorithm used for estimating growth parameters.

Usage

```
ELEFAN_GA(
  lfq,
  seasonalised = FALSE,
  low_par = NULL,
  up_par = NULL,
  popSize = 50,
  maxiter = 100,
  run = maxiter,
  parallel = FALSE,
  pmutation = 0.1,
  pcrossover = 0.8,
  elitism = base::max(1, round(popSize * 0.05)),
  MA = 5,
  addl.sqrt = FALSE,
  agemax = NULL,
  flagging.out = TRUE,
  seed = NULL,
  monitor = FALSE,
  plot = FALSE,
  plot.score = TRUE,
  ...
)
```

Arguments

`lfq` a list consisting of following parameters:

- **midLengths** midpoints of the length classes,
- **dates** dates of sampling times (class Date),
- **catch** matrix with catches/counts per length class (row) and sampling date (column);

`seasonalised` logical; indicating if the seasonalised von Bertalanffy growth function should be applied (default: FALSE).

`low_par`	a list providing the minimum of the search space in case of real-valued or permutation encoded optimizations. When set to NULL the following default values are used: • **Linf** length infinity in cm (default is calculated from maximum length class in the data), • **K** curving coefficient (default: 0.01), • **t_anchor** time point anchoring growth curves in year-length coordinate system, corrsponds to peak spawning month (range: 0 to 1, default: 0), • **C** amplitude of growth oscillation (range: 0 to 1, default: 0), • **ts** summer point (ts = WP - 0.5) (range: 0 to 1, default: 0);
`up_par`	a list providing the maximum of the search space in case of real-valued or permutation encoded optimizations. When set to NULL the following default values are used: • **Linf** length infinity in cm (default is calculated from maximum length class in the data), • **K** curving coefficient (default: 1), • **t_anchor** time point anchoring growth curves in year-length coordinate system, corrsponds to peak spawning month (range: 0 to 1, default: 1), • **C** amplitude of growth oscillation (range: 0 to 1, default: 1), • **ts** summer point (ts = WP - 0.5) (range: 0 to 1, default: 1);
`popSize`	the population size. Default: 50
`maxiter`	the maximum number of iterations to run before the GA search is halted. default:100
`run`	the number of consecutive generations without any improvement in the best fitness value before the GA is stopped. Default: equals maxiter
`parallel`	a logical argument specifying if parallel computing should be used (TRUE) or not (FALSE, default) for evaluating the fitness function. See `ga` for details. Default:FALSE, but setting to TRUE may substantially improve required calculation time. Use of this functionality requires the following packages: parallel, doParallel.
`pmutation`	the probability of mutation in a parent chromosome. Usually mutation occurs with a small probability, and by default is set to 0.1.
`pcrossover`	the probability of crossover between pairs of chromosomes. Typically this is a large value and by default is set to 0.8.
`elitism`	the number of best fitness individuals to survive at each generation. By default the top 5% individuals will survive at each iteration.
`MA`	number indicating over how many length classes the moving average should be performed (default: 5, for more information see lfqRestructure)
`addl.sqrt`	additional squareroot transformation of positive values according to Brey et al. (1988) (default: FALSE, for more information see lfqRestructure)
`agemax`	maximum age of species; default NULL, then estimated from Linf
`flagging.out`	logical; should positive peaks be flagged out? Original setting of ELEFAN in TRUE. Default:TRUE

`seed`	an integer value containing the random number generator state. This argument can be used to replicate the results of a GA search. Note that if parallel computing is required, the doRNG package must be installed. (Default: 'seed = NULL')
`monitor`	a logical or an R function which takes as input the current state of the 'ga-class' object and show the evolution of the search. By default, 'monitor = FALSE' so any output is suppressed. Possible also, the functions 'gaMonitor' or 'gaMonitor2' (depending on whether or not is an RStudio session) which print the average and best fitness values at each iteration. If set to 'plot' these information are plotted on a graphical device. Other functions can be written by the user and supplied as argument.
`plot`	logical; Plot restructured counts with fitted lines using `plot.lfq` and `lfqFitCurves` (default : FALSE).
`plot.score`	logical; Plot genetic algorithm fitness progression. (Default: plot.score=TRUE).
`...`	additional parameters to pass to `ga`

Details

A more detailed description of the generic algorithm (GA) can be found in Scrucca (2013). The score value `fitnessValue` is not comparable with the score value of the other ELEFAN functions (`ELEFAN` or `ELEFAN_SA`).

Value

A list with the input parameters and following list objects:

- **samplingPeriod**: length of sampling period in years,
- **samplingDays**: time of sampling times in relation to first sampling time,
- **delta_t**: array with time differences between relative sampling time set to zero and other sampling times,
- **rcounts**: restructured frequencies,
- **peaks_mat**: matrix with positive peaks with distinct values,
- **ASP**: available sum of peaks, sum of posititve peaks which could be potential be hit by growth curves,
- **ncohort**: maximum age of species,
- **agemax**: maximum age of species,
- **par**: a list with the parameters of the von Bertalanffy growth function:
  - **Linf**: length infinity in cm,
  - **K**: curving coefficient;
  - **t_anchor**: time point anchoring growth curves in year-length coordinate system, corrsponds to peak spawning month,
  - **C**: amplitude of growth oscillation (if `seasonalised` = TRUE),
  - **ts**: summer point of oscillation (ts = WP - 0.5) (if `seasonalised` = TRUE),
  - **phiL**: growth performance index defined as phiL = log10(K) + 2 * log10(Linf);
- **Rn_max**: highest value of fitness function, (comparable with ELEFAN and ELEFAN_SA).

References

Brey, T., Soriano, M., and Pauly, D. 1988. Electronic length frequency analysis: a revised and expanded user's guide to ELEFAN 0, 1 and 2.

Pauly, D. and N. David, 1981. ELEFAN I, a BASIC program for the objective extraction of growth parameters from length-frequency data. *Meeresforschung*, 28(4):205-211

Scrucca, L. (2013). GA: a package for genetic algorithms in R. Journal of Statistical Software, 53(4), 1-37.

Examples

```
# load data and view catch length frequencies
data(synLFQ4)
plot(synLFQ4, Fname="catch")

# Genetic algorithm
# (if using a multicore processor,
#   consider adding the argument 'parallel=TRUE'
#   to reduce computation time)
output <- ELEFAN_GA(synLFQ4, seasonalised = TRUE,
   low_par = list(Linf = 70, K = 0.25, t_anchor = 0, C = 0, ts= 0),
   up_par = list(Linf = 90, K = 0.7, t_anchor = 1, C = 1, ts = 1),
   popSize = 40, maxiter = 50, run = 20,
   MA = 11, plot = TRUE, seed = 1111)
output$par
output$ASP
output$Rn_max

# compare fitness score (fESP) to
# that calculated with "true" growth parameter values
plot(output, draw = FALSE)
lfqFitCurves(output, par=list(Linf=80, K=0.5, t_anchor=0.25, C=0.75, ts=0.5),
       draw = TRUE, col=1, flagging.out = FALSE)$fESP
lfqFitCurves(output, par=output$par, draw = TRUE, col=2, flagging.out = FALSE)$fESP
legend("top", legend=c("orig.", "GA"), lty=2, col=1:2, ncol=2)
```

`ELEFAN_SA` *ELEFAN_SA*

Description

Electronic LEngth Frequency ANalysis with simulated annealing for estimating growth parameters.

Usage

```
ELEFAN_SA(
  lfq,
```

```
  seasonalised = FALSE,
  init_par = list(Linf = 50, K = 0.5, t_anchor = 0.5, C = 0, ts = 0),
  low_par = NULL,
  up_par = NULL,
  SA_time = 60 * 1,
  maxit = NULL,
  nb.stop.improvement = NULL,
  SA_temp = 1e+05,
  verbose = TRUE,
  MA = 5,
  addl.sqrt = FALSE,
  agemax = NULL,
  flagging.out = TRUE,
  plot = FALSE,
  plot.score = TRUE
)
```

Arguments

`lfq` a list consisting of following parameters:

- **midLengths** midpoints of the length classes,
- **dates** dates of sampling times (class Date),
- **catch** matrix with catches/counts per length class (row) and sampling date (column);

`seasonalised` logical; indicating if the seasonalised von Bertalanffy growth function should be applied (default: FALSE).

`init_par` a list providing the Initial values for the components to be optimized. When set to NULL the following default values are used:

- **Linf** length infinity in cm (default is the maximum length class in the data),
- **K** curving coefficient (default: 0.5),
- **t_anchor** time point anchoring growth curves in year-length coordinate system, corrsponds to peak spawning month (range: 0 to 1, default: 0.5),
- **C** amplitude of growth oscillation (range: 0 to 1, default: 0),
- **ts** summer point (ts = WP - 0.5) (range: 0 to 1, default: 0);

`low_par` a list providing the lower bounds for components. When set to NULL the following default values are used:

- **Linf** length infinity in cm (default is calculated from maximum length class in the data),
- **K** curving coefficient (default: 0.01),
- **t_anchor** time point anchoring growth curves in year-length coordinate system, corrsponds to peak spawning month (range: 0 to 1, default: 0),
- **C** amplitude of growth oscillation (range: 0 to 1, default: 0),
- **ts** summer point (ts = WP - 0.5) (range: 0 to 1, default: 0);

`up_par` a list providing the upper bounds for components. When set to NULL the following default values are used:

	• **Linf** length infinity in cm (default is calculated from maximum length class in the data), • **K** curving coefficient (default: 0.01), • **t_anchor** time point anchoring growth curves in year-length coordinate system, corrsponds to peak spawning month (range: 0 to 1, default: 0), • **C** amplitude of growth oscillation (range: 0 to 1, default: 0), • **ts** summer point (ts = WP - 0.5) (range: 0 to 1, default: 0);
`SA_time`	numeric; Maximum running time in seconds (default : 60 * 1).
`maxit`	Integer. Maximum number of iterations of the algorithm. Default is NULL.
`nb.stop.improvement`	Integer. The program will stop when there is no any improvement in 'nb.stop.improvement' steps. Default is NULL
`SA_temp`	numeric; Initial value for temperature (default : 1e5).
`verbose`	logical; TRUE means that messages from the algorithm are shown (default : TRUE).
`MA`	number indicating over how many length classes the moving average should be performed (defalut: 5, for more information see lfqRestructure).
`addl.sqrt`	Passed to lfqRestructure. Applied an additional square-root transformation of positive values according to Brey et al. (1988). (default: FALSE, for more information see lfqRestructure).
`agemax`	maximum age of species; default NULL, then estimated from Linf
`flagging.out`	logical; passed to lfqFitCurves. Default is TRUE
`plot`	logical; Plot restructured counts with fitted lines using `plot.lfq` and `lfqFitCurves` (default : FALSE).
`plot.score`	logical; Plot simulated annealing score progression. (Default: plot.score=TRUE)

Details

A more detailed description of the simulated annealing (SA) can be found in Xiang et al. (2013). The score value `cost_value` is not comparable with the score value of the other ELEFAN functions (`ELEFAN` or `ELEFAN_GA`).

Value

A list with the input parameters and following list objects:

- **rcounts**: restructured frequencies,
- **peaks_mat**: matrix with positive peaks with distinct values,
- **ASP**: available sum of peaks, sum of posititve peaks which could be potential be hit by growth curves,
- **ncohort**: maximum age of species,
- **agemax**: maximum age of species,
- **par**: a list with the parameters of the von Bertalanffy growth function:

  - **Linf**: length infinity in cm,
  - **K**: curving coefficient;
  - **t_anchor**: time point anchoring growth curves in year-length coordinate system, corrsponds to peak spawning month,
  - **C**: amplitude of growth oscillation (if `seasonalised` = TRUE),
  - **ts**: summer point of oscillation (ts = WP - 0.5) (if `seasonalised` = TRUE),
  - **phiL**: growth performance index defined as phiL = log10(K) + 2 * log10(Linf);
- **Rn_max**: highest score value (absolute value of cost function, comparable with ELEFAN and ELEFAN_GA).

References

Brey, T., Soriano, M., and Pauly, D. 1988. Electronic length frequency analysis: a revised and expanded user's guide to ELEFAN 0, 1 and 2.

Pauly, D. and N. David, 1981. ELEFAN I, a BASIC program for the objective extraction of growth parameters from length-frequency data. *Meeresforschung*, 28(4):205-211

Xiang, Y., Gubian, S., Suomela, B., & Hoeng, J. (2013). Generalized simulated annealing for global optimization: the GenSA Package. R Journal, 5(1), 13-28.

Examples

```
## synthetic lfq data example
data(synLFQ4)
plot(synLFQ4, Fname="catch")

# ELEFAN_SA (takes approximately 30 seconds)
output <- ELEFAN_SA(synLFQ4, SA_time = 60*0.5, seasonalised = TRUE, MA = 11,
  init_par = list(Linf = 75, K = 0.5, t_anchor = 0.5, C = 0.5, ts = 0.5),
  low_par = list(Linf = 70, K = 0.3, t_anchor = 0, C = 0, ts = 0),
  up_par = list(Linf = 90, K = 0.7, t_anchor = 1, C = 1, ts = 1))
output$par
output$Rn_max

# view fit
plot(output)

# or
plot(output, draw = FALSE)
lfqFitCurves(output, col=1, par=output$par, draw=TRUE)$ESP

# compare to original parameters
tmp <- lfqFitCurves(output, col=4, lty=1,
   par=list(Linf=80, K=0.5, t_anchor=0.25, C=0.75, ts=0.5), draw=TRUE)
tmp$fESP
output$Rn_max
```

`emperor` *Emperor data*

Description

This dataset contains information about sky emperor (*Lethrinus mahsena*) and its fisheries of offshore Mauritius banks (Nazareth banks). It can be used for production models (`prod_mod` and `prod_mod_ts`).

Usage

```
data(emperor)
```

Format

A dataframe containing: 1. **year** years, 2. **Y** yield, 3. **CPUE** CPUE or abundance index

Source

Dharmendra, D., Solmundsson, J., 2005. Stock assessment of the offshore Mauritian banks using dynamic biomass models and analysis of length frequency of the Sky Emperor (*Lethrinus mahsena*). Fisheries Training Program The United Nations University, 61

Examples

```
data(emperor)
str(emperor)
summary(emperor)
```

`gillnet` *Gillnet data*

Description

Data of an experiment with several gillnets with different mesh sizes. Data can be analysed with function `select_Millar`.

Usage

```
data(gillnet)
```

Format

A list consiting of:

- `midLengths` the midlengths of size classes,
- `meshSizes` the meshsizes,
- `catchPerNet_mat` a matrix with the numbers in catch of the corresponding mesh sizes (same order),

Source

Millar, R. B., Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3), 471-477.

Holt, S. J. 1963. A method for determining gear selectivity and its application. *ICNAF Special Publication*, 5: 106-115.

Examples

```
data(gillnet)
str(gillnet)
summary(gillnet)
```

`gillnetfit` *Millar's original gillnet selectivity fitting function*

Description

Function to estimate selectivity parameters from experimental data. This function is applied within `select_Millar` to derive starting parameters. `select_Millar` is the recommended function for selectivity estimation.

Usage

```
gillnetfit(
  data,
  meshsizes,
  rtype = "norm.loc",
  rel.power = NULL,
  plotlens = NULL,
  details = FALSE
)
```

Arguments

`data`	matrix with the number of individuals caught with each sized mesh (`CatchPerNet_mat`).
`meshsizes`	vector with meshSizes in increasing order (`meshSizes`),
`rtype`	A character string indicating which method for estimating selection curves should be used: `"norm.loc"` for a normal curve with common spread, `"norm.sca"` for a normal curve with variable spread, `"lognorm"` for a lognormal curve, `"gamma"` for a gamma curve.
`rel.power`	A string indicating the relative power of different meshSizes, must have same length as `meshSizes` (Default: `rel.power = NULL`).
`plotlens`	lengths which should be used for graphical output, for more detailed curves. Default : NULL
`details`	logical; should details be included in the output?

Value

list of fitted parameters

Source

https://www.stat.auckland.ac.nz/~millar/selectware/

References

Millar, R. B., Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3):471-477

Examples

```
data(gillnet)

dat <- matrix(c(gillnet$midLengths, gillnet$CatchPerNet_mat),
         byrow = FALSE, ncol=(dim(gillnet$CatchPerNet_mat)[2]+1))

gillnetfit(data = dat, meshsizes = gillnet$meshSizes)
```

`goatfish` *Yellowstriped goatfish data*

Description

Data of Yellowstriped goatfish (*Upeneus vittatus*) from Manila Bay, Philippines. Can be used for the estimation of the instantaneous mortality rate (Z) by means of `catchCurve`.

Usage

```
data(goatfish)
```

Format

A list consisting of:

- `midLengths`: mid points of length classes,
- `catch`: a vector with catches in numbers,
- `Linf`: infinite length in cm [cm],
- `K`: growth coefficent per year [1/year].

Source

Ziegler, B., 1979. Growth and mortality rates of some fishes of Manila Bay, Philippines, as estimated from analysis of length-frequencies. Thesis. Kiel University, 115 p.

Examples

```
data(goatfish)
str(goatfish)
summary(goatfish)
```

`growth_length_age` *Estimation of growth parameter using length-at-age data*

Description

This function estimates growth parameters from length-at-age data. It allows to perform different methods: Gulland and Holt, Ford Walford plot, Chapman's method, Bertalanffy plot, or non linear least squares method (LSM).

Usage

```
growth_length_age(
  param,
  method,
  Linf_est = NA,
  Linf_init = 10,
  K_init = 0.1,
  t0_init = 0,
  CI = FALSE,
  ci.level = 0.95,
  age_plot = NULL,
  do.sim = FALSE
)
```

Arguments

`param`	a list consisting of following parameters: • **age**: age measurements, • **length**: corresponding lengths in cm.
`method`	indicating which of following methods should be applied: `"GullandHolt"`, `"FordWalford"`, `"Chapman"`, `"BertalanffyPlot"`, or `"LSM"`
`Linf_est`	BertalanffyPlot requires an estimate for Linf to derive K and t0 (for more information see Details).
`Linf_init`	initital parameter of Linf for non-linear sqaures fitting (default 10)
`K_init`	initital parameter of K for non-linear sqaures fitting (default 0.1)
`t0_init`	initital parameter of t0 for non-linear sqaures fitting (default 0)
`CI`	logical; Should confidence intervals be calculated? This option only works for the LSM method. Default is FALSE.
`ci.level`	required confidence level (for LSM method only)
`age_plot`	sequence with ages used for plotting (LSM method only). By default age_plot = seq(min(param$age),max(param$age),0.1)
`do.sim`	Deprecated.

Details

Gulland and Holt plot assumes infinitestimal delta t (only reasonable approximation of growth parameters if delta t is small). Ford Walford plot and Chapman assume constant time intervals between ages (delta t). The Bertalanffy plot is a robust method, however it requires an estimate of Linf. As long as this estimate is reasonable the resulting estimate of K is reasonable. For a first estimate of Linf the Powell Wetherall method powell_wetherall can be used. Otherwise, the largest fish or the average of the ten largest fish can be used for a small or large sample, respectively. All lengths have to be smaller than Linf as otherwise the logarithm is not defined. Oldest fish (if larger than Linf) have to be omitted. Non-linear least squares fitting is the preferred method to estimate growth parameters according to Sparre and Venema (1998). If `CI = TRUE` the confidence interval of parameters is calculated and plotted.

Value

A list with the input parameters and following parameters:

- **x**: independent variable used for regression analysis,
- **y**: dependent variable used for regression analysis,
- **mod**: (non) linear model,
- **Linf**: infinite length for investigated species in cm [cm],
- **K**: growth coefficent for investigated species per year [1/year],
- **t0**: theoretical time zero, at which individuals of this species hatch (only for Bertalanffy plot and LSM method).
- **estimates**: dataframe with growth parameters and confidence intervals (only if LSM method was applied).

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
# synthetical length at age data
dat <- list(age = rep(1:7,each = 5),
   length = c(rnorm(5,25.7,0.9),rnorm(5,36,1.2),rnorm(5,42.9,1.5),rnorm(5,47.5,2),
   rnorm(5,50.7,0.4),rnorm(5,52.8,0.5),rnorm(5,54.2,0.7)))
growth_length_age(dat, method = "GullandHolt")

# Bertalaffy plot
growth_length_age(dat, method = "BertalanffyPlot", Linf_est = 50)

# non linear least squares method

output <- growth_length_age(param = dat, method = "LSM",
     Linf_init = 30, CI = TRUE, age_plot=NULL)
summary(output$mod)
```

`growth_tagging` *Growth from tagging data*

Description

This function estimates growth parameters from tagging data. Munro plot is applied

Usage

```
growth_tagging(param, method, Linf_range = c(5, 600), time_unit = "year")
```

Arguments

`param`	a list consisting of following parameters: • **L1**: length at tagging [cm], • **L2**: length at recapture [cm], • **delta_t**: time interval between tagging ang recapture (instead two vectors with **t1** (age at tagging) and **t2** (age at recapture) can be provided.
`method`	indicating which of following methods should be applied: "GullandHolt" or "Munro".
`Linf_range`	two values indicating the lower and upper limits of the range, in which the optimise searches for the Linf value with the best fit (lowest CV value),
`time_unit`	indicating the unit of the time interval, either "year", "month", "week", or "day"

Details

If Munro plot is applied the optimal Linf value is found by minimizing the coefficient of variation (CV = sd(K)/mean(K)). For this iterative method the optimise function is applied. The histogram of the individual K values allows to distinguish potential differences in growth performance between individuals. t0 can not be estimated by Munro plot, neither by the Gulland Holt method.

Value

A list with the input parameters and following parameters:

- **x**: independent variable used for regression analysis,
- **y**: dependent variable used for regression analysis,
- **reg_coeffs**: regression coefficients,
- **r2**: r squared of regression analysis,
- **Linf**: infinite length for investigated species in cm [cm],
- **K**: growth coefficent for investigated species per year [1/year],
- **conf_int_K**: confidence intervals of K (only if Gulland Holt method was applied).

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Sparre, P., Venema, S.C., 1999. Introduction to tropical fish stock assessment. Part 2. Excercises. FAO Fisheries Technical Paper, (306.2, Rev. 2). 94 p.

Wolff, M., 1984. Early setback for scallop culture in Peru.

Examples

```
# from Wolff (1984)
dat <- list(L1 = c(40,46,29,30,18,31,48,49,59,58,61,65,57,55),
   L2 = c(85,53,55,56,25,43,70,59,62,80,72,83,65,56),
   delta_t = c(289,26,84,77,14,38,89,38,28,149,89,74,38,21))
growth_tagging(param = dat, "Munro", time_unit = "day", Linf_range=c(80,120))
growth_tagging(param = dat, "GullandHolt", time_unit = "day")


# from Sparre and Venema (1999)
dat <- list(L1 = c(9.7,10.5,10.9,11.1,12.4,12.8,14.0,16.1,16.3,17.0,17.7),
   L2 = c(10.2,10.9,11.8,12.0,15.5,13.6,14.3,16.4,16.5,17.2,18.0),
   delta_t = c(53,33,108,102,272,48,53,73,63,106,111))
growth_tagging(param = dat, "Munro", time_unit = "day", Linf_range = c(10,40))
growth_tagging(param = dat, "GullandHolt", time_unit = "day")
```

`haddock` *Haddock data*

Description

Data of a covered codend experimental catch of the haddock (*Melanogrammus aeglefinus*). Can be used for function `select_Millar`.

Usage

```
data(haddock)
```

Format

A list consisting of:

- `midLengths` the midlengths of size classes,
- `numCodend` the number of fish retained in codend,
- `numCover` the number of fish retained in cover,

Source

Millar, R. B., Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3), 471-477.

Examples

```
data(haddock)
str(haddock)
summary(haddock)
```

`hake` *Hake data*

Description

This dataset contains length-frequency data and biological characteristics about hake (*Merluccius merluccius*) and its fisheries off Senegal. It can be used for `VPA` or `predict_mod`.

Usage

```
data(hake)
```

Format

A list consisting of: 1. a vector with midlengths of size classes, 2. a vector with catch in numbers, 3. K value, 4. Linf value, 5. M value, 6. a value, 7. b value, 8. a vector with fishing mortalities, and 9. a vector with average value of fish per kg

Source

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
data(hake)
str(hake)
summary(hake)
```

`lfqCreate` *Create lfq data from length measurements*

Description

Convert raw length measurements to length frequency data (lfq class).

Usage

```
lfqCreate(
  data,
  Lname,
  Dname,
  Fname = NA,
  bin_size = 1,
  species = NA,
  stock = NA,
  comment = "",
  Lmin = 0,
  length_unit = "cm",
  plus_group = FALSE,
  aggregate_dates = FALSE,
  plot = FALSE
)
```

Arguments

`data`	data with at least two columns, one with the length measurements, one with the sampling date
`Lname`	name of the length column
`Dname`	name of the date column
`Fname`	optional; name of column with frequency, in case each length was measured more than one time
`bin_size`	size of the bins in cm (Default: 2)
`species`	character; to store species name in lfq list
`stock`	character; to store stock ID or name in lfq list
`comment`	optional character; to store comments conerning the lfq list
`Lmin`	minimum length for the midLengths vector (default: 0)
`length_unit`	unit of length measurements, either "cm" (default), "mm" or "m"
`plus_group`	logical; should a plus group be created? If yes you will be asked to insert the length for the plus group in the console (default: FALSE). Instead of inserting the length of the plus group via the console, the value can be incorporated in a vector, e.g. plus_group = c(TRUE, 30).
`aggregate_dates`	logical; indicating whether dates should be lumped in monthly sampling times (assuming sampling always aound the 15th of each month; default is FALSE). More exact lumping can only done manually and then sampling dates provided in data.
`plot`	logical; should a graph of lfq data be displayed? (Default: FALSE)

Value

A list of "lfq" class with

- **dates** dates of sampling times (class Date),
- **midLengths** midpoints of the length classes,
- **catch** matrix with catches/counts per length class (row) and sampling date (column).

Examples

```
# create random data
set.seed(1)
data <- data.frame(length.mm. = sample(c(rpois(300, lambda = 60),
           rpois(200, lambda = 100), rpois(100, lambda = 150)),
           size = 1000, replace = TRUE),
           dates = seq.Date(as.Date("2015-10-02"),as.Date("2016-08-28"),
           length.out = 1000))
# create lfq data
lfq_dat <- lfqCreate(data,Lname = "length.mm.", Dname = "dates", aggregate_dates = TRUE,
   length_unit = "mm", bin_size = 0.5, plot=TRUE, plus_group=c(TRUE,15.75))
```

`lfqFitCurves` *Fitting VBGF growth curves through lfq data*

Description

Thsi function estimates von Bertalanffy growth function (VBGF) curves for a set of growth parameters.

Usage

```
lfqFitCurves(
  lfq,
  par = list(Linf = 100, K = 0.1, t_anchor = 0.25, C = 0, ts = 0),
  agemax = NULL,
  flagging.out = TRUE,
  lty = 2,
  lwd = 1,
  col = 1,
  draw = FALSE,
  tincr = 0.05
)
```

Arguments

`lfq` a list of the class "lfq" consisting of following parameters:

- **midLengths** midpoints of the length classes,
- **dates** dates of sampling times (class Date),
- **catch** matrix with catches/counts per length class (row) and sampling date (column),
- **rcounts** restructured frequencies,
- **peaks_mat** matrix with positive peaks with distinct values,
- **ASP** available sum of peaks, sum of posititve peaks which could be potential be hit by growth curves;

`par` a list with following growth parameters:

- **Linf** length infinity in cm (default: 100),
- **K** curving coefficient (default: 0.1),
- **t_anchor** time point anchoring growth curves in year-length coordinate system, corrsponds to peak spawning month (range: 0 to 1, default: 0.25),
- **C** amplitude of growth oscillation (range: 0 to 1, default: 0),
- **ts** summer point (ts = WP - 0.5) (range: 0 to 1, default: 0);

`agemax` maximum age of species; default NULL, then estimated from Linf

`flagging.out` logical; should positive peaks be flagged out? (Default : TRUE)

`lty`	The line type. Line types can either be specified as an integer (0=blank, 1=solid, 2=dashed (default), 3=dotted, 4=dotdash, 5=longdash, 6=twodash) or as one of the character strings "blank", "solid", "dashed", "dotted", "dotdash", "longdash", or "twodash", where "blank" uses 'invisible lines' (i.e., does not draw them).
`lwd`	The line width, a positive number, defaulting to 2. The interpretation is device-specific, and some devices do not implement line widths less than one. (See the help on the device for details of the interpretation.)
`col`	A specification for the default plotting color. See section 'Color Specification'.
`draw`	logical; indicating whether growth curves should be added to existing lfq plot
`tincr`	step for plotting

Details

`t_anchor` subsitutes the starting point from known from Fisat 2. This parameter is necessary for anchoring the growth curves on the time axis. It does not subsitute `t0`. However, it corresponds to the peak spawning of the species (x intercept of growth curve) and has values between 0 and 1, where 0 corresponds to spawning at the 1st of January and 0.999 corresponds to the 31st of December. The default value of 0.25 or 3/12 corresponds the third month of the year, March.

Value

A list with the input parameters and following list objects:

- **Lt**: dataframe with ages and lengths of the cohorts,
- **agemax**: maximum age of species.
- **ncohort**: number of cohorts,
- **ASP**: available sum of peaks, sum of posititve peaks which could be potential be hit by growth curves. This is calculated as the sum of maximum values from each run of posive restructured scores.
- **ESP**: available sum of peaks,
- **fASP**: available sum of peaks,
- **fESP**: available sum of peaks,

References

Brey, T., Soriano, M., and Pauly, D. 1988. Electronic length frequency analysis: a revised and expanded user's guide to ELEFAN 0, 1 and 2.

Pauly, D. 1981. The relationship between gill surface area and growth performance in fish: a generalization of von Bertalanffy's theory of growth. *Meeresforsch.* 28:205-211

Pauly, D. and N. David, 1981. ELEFAN I, a BASIC program for the objective extraction of growth parameters from length-frequency data. *Meeresforschung*, 28(4):205-211

Pauly, D., 1985. On improving operation and use of ELEFAN programs. Part I: Avoiding "drift" of K towards low values. *ICLARM Conf. Proc.*, 13-14

Pauly, D., 1987. A review of the ELEFAN system for analysis of length-frequency data in fish and aquatic invertebrates. *ICLARM Conf. Proc.*, (13):7-34

Pauly, D. and G. R. Morgan (Eds.), 1987. Length-based methods in fisheries research. (No. 13). WorldFish

Pauly, D. and G. Gaschuetz. 1979. A simple method for fitting oscillating length growth data, with a program for pocket calculators. I.C.E.S. CM 1979/6:24. Demersal Fish Cttee, 26 p.

Pauly, D. 1984. Fish population dynamics in tropical waters: a manual for use with programmable calculators (Vol. 8). WorldFish.

Quenouille, M. H., 1956. Notes on bias in estimation. *Biometrika*, 43:353-360

Somers, I. F., 1988. On a seasonally oscillating growth function. ICLARM Fishbyte 6(1): 8-11.

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2): 407 p.

Tukey, J., 1958. Bias and confidence in not quite large samples. *Annals of Mathematical Statistics*, 29: 614

Tukey, J., 1986. The future of processes of data analysis. In L. V. Jones (Eds.), The Collected Works of John W. Tukey-philosophy and principles of data analysis: 1965-1986 (Vol. 4, pp. 517-549). Monterey, CA, USA: Wadsworth & Brooks/Cole

Examples

```
data(synLFQ5)
res <- lfqRestructure(synLFQ5, MA=11)
plot(res)
tmp <- lfqFitCurves(res, par=list(Linf=80,K=0.5,t_anchor=0.25), draw=TRUE)
```

`lfqModify` *Modify lfq data for further analysis*

Description

Modify length-freqeuncy (LFQ) data. Allows to summarise catch matrix of LFQ data to one column per year. This is required for e.g. `catchCurve`. Allows to change bin size of LFQ data. Allows to ad plus group to catch matrix.

Usage

```
lfqModify(
  lfq,
  par = NULL,
  bin_size = NA,
  aggregate = NA,
  vectorise_catch = FALSE,
  plus_group = FALSE,
  minDate = NA,
  maxDate = NA,
  years = NA,
```

```
  Lmin = NA,
  Lmax = NA,
  lfq2 = NA
)
```

Arguments

`lfq`	lfq object with dates, midLengths, and catch
`par`	growth parameters as resulting from e.g. `ELEFAN`
`bin_size`	Bin size for length frequencies (in cm)
`aggregate`	Factor to aggregate catch per year (`"year"`), per quarter (`"quarter"`), or per month (`"month"`). By default data is not aggregated (`NA`).
`vectorise_catch`	logical; indicating if the catch matrix should be summarised to yearly vectors (default: FALSE).
`plus_group`	logical or numeric; should a plus group be created? If yes you will be asked to insert the length for the plus group in the console (default: FALSE). Instead of inserting the length of the plus group via the console, the value can be inserted, e.g. plus_group = 85.5.
`minDate`	minimum date to subset lfq data
`maxDate`	maximum date to subset lfq data
`years`	numeric with year(s) to subset lfq data
`Lmin`	minimum length to subset lfq data
`Lmax`	maximum length to subset lfq data
`lfq2`	optional second lfq object which will be merged with lfq. This might be interesting for fleet specific lfq objects. Default: NA. Be aware that catches are combined without weighting!

Value

lfq object with rearranged catch matrix (yearly sums) and growth parameters if provided.

Examples

```
data(synLFQ4)

## summarise catch matrix per year
lfq_sum <- lfqModify(synLFQ4, vectorise_catch = TRUE)

## change bin size
lfq_bin <- lfqModify(synLFQ4, bin_size = 4)

## add plus_group
lfq_plus <- lfqModify(synLFQ4, plus_group = 85.5)
```

`lfqRestructure` *Restructuring of length frequency data*

Description

First step of the Electronic LEngth Frequency ANalysis (ELEFAN), which is restructuring length-frequency data (lfq). This is done according to a certain protocol, described by many authors (see Details or References for more information).

Usage

```
lfqRestructure(param, MA = 5, addl.sqrt = FALSE)
```

Arguments

`param`	a list consisting of following parameters: • **midLengths** midpoints of the length classes • **dates** dates of sampling times (class Date) • **catch** matrix with catches/counts per length class (row) and sampling date (column)
`MA`	number indicating over how many length classes the moving average should be performed (default: 5)
`addl.sqrt`	additional squareroot transformation of positive values according to Brey et al. (1988) (default: FALSE). Particularly useful if many observations have a low frequency (<10)

Details

This function is used prior to fitting of growth curves (e.g. in `ELEFAN`, `ELEFAN_SA` functions). It restructures a length frequency data set according to a list of steps to emphasise cohorts in the data. The steps can be found in various publications, see e.g. Brey et al. (1988) or Pauly and David (1981). Here, the most recent steps documented in Gayanilo (1997) are followed.

Value

A list with the input parameters and following list objects:

- **rcounts**: restructured frequencies,
- **peaks_mat**: matrix with uniquely numbered positive peaks,
- **ASP**: available sum of peaks, sum of posititve peaks which could be potential be hit by growth curves. This is calculated as the sum of maximum values from each run of posive restructured scores,
- **MA**: moving average used for restructuring.

References

Brey, T., Soriano, M., and Pauly, D. 1988. Electronic length frequency analysis: a revised and expanded user's guide to ELEFAN 0, 1 and 2.

Gayanilo, Felimon C. FAO-ICLARM stock assessment tools: reference manual. No. 8. Food & Agriculture Org., 1997.

Pauly, D. 1981. The relationship between gill surface area and growth performance in fish: a generalization of von Bertalanffy's theory of growth. *Meeresforsch.* 28:205-211

Pauly, D. and N. David, 1981. ELEFAN I, a BASIC program for the objective extraction of growth parameters from length-frequency data. *Meeresforschung*, 28(4):205-211

Pauly, D., 1985. On improving operation and use of ELEFAN programs. Part I: Avoiding "drift" of K towards low values. *ICLARM Conf. Proc.*, 13-14

Pauly, D., 1987. A review of the ELEFAN system for analysis of length-frequency data in fish and aquatic invertebrates. *ICLARM Conf. Proc.*, (13):7-34

Pauly, D. and G. R. Morgan (Eds.), 1987. Length-based methods in fisheries research. (No. 13). WorldFish

Pauly, D. and G. Gaschuetz. 1979. A simple method for fitting oscillating length growth data, with a program for pocket calculators. I.C.E.S. CM 1979/6:24. Demersal Fish Cttee, 26 p.

Pauly, D. 1984. Fish population dynamics in tropical waters: a manual for use with programmable calculators (Vol. 8). WorldFish.

Quenouille, M. H., 1956. Notes on bias in estimation. *Biometrika*, 43:353-360

Somers, I. F., 1988. On a seasonally oscillating growth function. ICLARM Fishbyte 6(1): 8-11.

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2): 407 p.

Tukey, J., 1958. Bias and confidence in not quite large samples. *Annals of Mathematical Statistics*, 29: 614

Tukey, J., 1986. The future of processes of data analysis. In L. V. Jones (Eds.), The Collected Works of John W. Tukey-philosophy and principles of data analysis: 1965-1986 (Vol. 4, pp. 517-549). Monterey, CA, USA: Wadsworth & Brooks/Cole

Examples

```
# data and plot of catch frequencies
data(synLFQ4)
plot(synLFQ4, Fname="catch")

# restructuring and calculation of ASP
synLFQ4 <- lfqRestructure(synLFQ4, MA=11)
synLFQ4$ASP

# plot of restructured scores and fit of soVBGF growth curves
plot(synLFQ4)
lfqFitCurves(synLFQ4,
 par=list(Linf=80, K=0.5, t_anchor=0.25, C=0.75, ts=0),
 draw=TRUE
)$fASP
```

`M_empirical` *Empirical formulas for the estimation of natural mortality*

Description

Functions to calculate the instantaneous natural mortality rate (M) according to 12 different empirical formulas.

Usage

```
M_empirical(
  Linf = NULL,
  Winf = NULL,
  K_l = NULL,
  K_w = NULL,
  temp = NULL,
  tmax = NULL,
  tm50 = NULL,
  GSI = NULL,
  Wdry = NULL,
  Wwet = NULL,
  Bl = NULL,
  schooling = FALSE,
  method
)
```

Arguments

`Linf`	infinite total length (TL) from a von Bertalanffy growth curve in cm.
`Winf`	infinite weight form a von Bertalanffy growth curve in wet weight-grams.
`K_l`	is the growth coefficient (per year) from a von Bertalanffy growth curve for length.
`K_w`	is the growth coefficient (per year) from a von Bertalanffy growth curve for weight.
`temp`	average annual temperature at the surface in degrees centigrade.
`tmax`	the oldest age observed for the species.
`tm50`	age when 50% of the population is mature [year] ("age of massive maturation").
`GSI`	gonadosomatic index (wet ovary weight over wet body weight).
`Wdry`	total dry weight in grams.
`Wwet`	total wet weight at mean length in grams.
`Bl`	vector with body lengths in cm for size dependent mortality estimates (method = "Gislason")

`schooling`	logical; if TRUE it is accounted for the schooling behaviour of the species, only for Pauly's methods. Default is FALSE.
`method`	vector of method names. Any combination of following methods can be employed: "AlversonCarney", "Gislason" (size dependent mortality estimates), "GundersonDygert", "Hoenig", "Lorenzen", "Pauly_Linf", "Pauly_Winf", "PetersonWroblewski", "RikhterEfanov", "Roff", "Then_growth", or "Then_tmax". Please refer to Details to see which input parameters are required by each method.

Details

Function adapted from the mortality function of the fishmethods package by Gary A. Nelson (https://cran.r-project.org/web/packages/fishmethods/index.html).

Depending on the method different input parameters are required:

- `"AlversonCarney"` requires `K_l` and `tmax`,
- `"Gislason"` requires `Linf`, `K_l` and `Bl`,
- `"GundersonDygert"` requires `GSI`,
- `"Hoenig"` requires `tmax`,
- `"Lorenzen"` requires `Wwet`,
- `"Pauly_Linf"` requires `Linf`, `K_l` and `temp`,
- `"Pauly_Winf"` requires `Winf`, `K_w` and `temp`,
- `"PetersonWroblewski"` requires `Wdry`,
- `"RikhterEfanov"` requires `tm50`,
- `"Roff"` requires `K_l` and `tm50`,
- `"Then_tmax"` requires `tmax`,
- `"Then_growth"` requires `Linf` and `K_l`.

If accounting for schooling behaviour M is multiplied by 0.8 according to Pauly (1983).

Value

A matrix of M estimates.

Source

https://cran.r-project.org/web/packages/fishmethods/index.html

References

Alverson, D. L. and M. J. Carney. 1975. A graphic review of the growth and decay of population cohorts. J. Cons. Int. Explor. Mer 36: 133-143.

Gislason, H., N. Daan, J. C. Rice, and J. G. Pope. 2010. Size, growth, temperature and the natural mortality of marine fish. Fish and Fisheries 11: 149-158.

Gunderson, D. R. and P. H. Dygert. 1988. Reproductive effort as a predictor of natural mortality rate. J. Cons. Int. Explor. Mer 44: 200-209.

Hoenig, J. M. 1983. Empirical use of longevity data to estimate mortality rates. Fish. Bull. 82: 898-903.

Lorenzen, K. 1996. The relationship between body weight and natural mortality in juvenile and adult fish: a comparison of natural ecosystems and aquaculture. J. Fish. Biol. 49: 627-647.

Pauly, D. 1980. On the interrelationships between natural mortality, growth parameters, and mean environmental temperature in 175 fish stocks. J. Cons. Int. Explor. Mer: 175-192.

Pauly, D., 1983. Some simple methods for the assessment of tropical fish stocks. *FAO Fish.Tech.Pap.*, (234): 52p. Issued also in French and Spanish

Peterson, I. and J. S. Wroblewski. 1984. Mortality rate of fishes in the pelagic ecosystem. Can. J. Fish. Aquat. Sci. 41: 1117-1120.

Rikhter, V.A., and V.N. Efanov, 1976. On one of the approaches to estimation of natural mortality of fish populations. *ICNAF Res.Doc.*, 76/VI/8: 12p.

Roff, D. A. 1984. The evolution of life history parameters in teleosts. Can. J. Fish. Aquat. Sci. 41: 989-1000.

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Then, A. Y., J. M. Hoenig, N. G. Hall, D. A. Hewitt. 2015. Evaluating the predictive performance of empirical estimators of natural mortality rate using information on over 200 fish species. ICES J. Mar. Sci. 72: 82-92.

Examples

```
M_empirical(Linf = 80, K_l = 0.5, temp = 25, tmax = 30,
     method = c("Pauly_Linf","Hoenig"))
```

`plot.Bhattacharya` *Bhattacharya plot*

Description

This function plots the seperated frequency distributions and selected regression lines of Bhattacharya method.

Usage

```
## S3 method for class 'Bhattacharya'
plot(x, analysisPlot = TRUE, ...)
```

Arguments

`x`	a list of the class `"Bhattacharya"` containing the results of Bhattacharya's method.
`analysisPlot`	logical; indicating wheter the anaylsis graph with the regression lines should be created
`...`	additional options of the `plot` function

Details

This function plots the results of the Bhattacharya method.

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
## Not run:
 data(synLFQ1)
 output <- Bhattacharya(param = synLFQ1)
 plot(output)

## End(Not run)
```

`plot.catchCurve` *Plotting catch curve*

Description

This function plots the results from the `catchCurve` model.

Usage

```
## S3 method for class 'catchCurve'
plot(
  x,
  xaxis = "age",
  plot_selec = FALSE,
  col = c("blue", "darkgreen", "orange", "darkred"),
  cex = 1.5,
  xlim = NULL,
  ylim = NULL,
  xlab = "default",
  ylab = "default",
  ...
)
```

Arguments

`x`	A list of the class `"catchCurve"` containing the results of the catchCurve model.
`xaxis`	Character defining if x axis should represent length or age (default: 'age')
`plot_selec`	logical; if TRUE the regression line is plotted for not fully exploited length groups and the probability of capture is plotted. This only works if the catchCurve was applied with `calc_ogive` == TRUE.

`col`	a specification for colour of regression points, line and annotation
`cex`	a numerical value giving the amount by which plotting text and symbols should be magnified relative to the default.
`xlim`	limits of x axis
`ylim`	limits of y axis
`xlab`	label of x axis. Default display by setting to "default".
`ylab`	label of y axis. Default display by setting to "default".
`...`	standard parameters of plot function

Details

A function to plot the results of the catchCurve model.

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. FAO Fisheries Technical Paper, (306.1, Rev. 2). 407 p.

Examples

```
data(synLFQ3)
output <- catchCurve(synLFQ3, calc_ogive = TRUE, reg_int = c(9,21))
plot(output, plot_selec = TRUE)
```

`plot.lfq` *Plotting of length frequency data (with VBGF curves)*

Description

This function plots length frequency (lfq) samples sequentially arranged in time. An object of "lfq" class is obatined by applying the `lfqRestructure` function. In case growth parameters are known, von Bertalanffy growth curves can be plotted through the lfq samples.

Usage

```
## S3 method for class 'lfq'
plot(
  x,
  Fname = "rcounts",
  par = NULL,
  agemax = NULL,
  rel = FALSE,
  y = NA,
  curve.col = 1,
```

```
  hist.sc = 0.5,
  hist.col = c("white", "black", "orange", "darkgreen"),
  image.col = NULL,
  region.col = NULL,
  zlim = NULL,
  zlimtype = "balanced",
  date.axis = "traditional",
  date.at = seq(as.Date("1500-01-01"), as.Date("2500-01-01"), by = "months"),
  date.format = "'%y-%b",
  xlab = "",
  ylab = "Length classes",
  draw = TRUE,
  ...
)
```

Arguments

`x`	a list of the class "lfq" consisting of following parameters: - **midLengths** midpoints of the length classes, - **dates** dates of sampling times (class Date), - **catch** matrix with catches/counts per length class (row) and sampling date (column);
`Fname`	indicating whether restructured ("rcounts") or original frequencies ("catch") should be displayed (default: "rcounts")
`par`	a list with following growth parameters (default NULL): - **Linf** asymptotic length, - **K** growth coefficient, - **t_anchor** time at length zero, - **C** amplitude of growth oscillation (optional), - **ts** summer point (optional);
`agemax`	maximum age of species; default NULL, then estimated from Linf
`rel`	logical; defines if relative numbers per length class should be plotted (relative to the sample size per sampling time, e.g. month). Default: FALSE.
`y`	an optional second list of class "lfq" consisting of same parameters as x. This allows to plot samples from different sources (e.g. different fleets) on top of eachother, classes and dates have to correspond at least partially. Default is NA.
`curve.col`	colour of growth curves (default: 1)
`hist.sc`	defines the scaling factor to use for maximum histogram extent (x-axis direction). The default setting of hist.sc=0.5 will result in a maximum distance equal to half the distance between closest sample dates (i.e. ensures no overlap and full plotting within the plot region).
`hist.col`	vector of 2 values defining coloring to use on negative and positive histogram bars (default: hist.col=c("white", "black", "orange", "darkgreen"))
`image.col`	colour of image, by default (NULL) red and blue colours are used. To remove image coloring, set image.col=NA.

`region.col`	colour of plotting region. Will overwrite image.col (default: region.col=NULL)
`zlim`	the minimum and maximum z values for which colors should be plotted (default : NULL).
`zlimtype`	indicating if zlim should be based on the range of the catch ("range") or based on the maximum absolute value in Fname ("balanced", default). This parameter is only considered if zlim is NULL.
`date.axis`	the style of the x axis. By default the "traditional" approach is used with years under the months. Alternatively, by using "modern" the date is plotted in one line according to the chosen format `date.format`.
`date.at`	the points at which tick-marks are to be drawn. Non-finite (infinite, NaN or NA) values are omitted. By default it is seq(as.Date("1500-01-01"), as.Date("2500-01-01"), by="months")
`date.format`	format of date (default : "%y-%b")
`xlab`	label of x axis (default : "")
`ylab`	label of y axis (default : "Length classes")
`draw`	logical; indicating whether growth curves should be added to lfq plot if parameters are provided (default : TRUE)
`...`	additional options of the plot function

Details

This function uses `lfqFitCurves` when growth parameters are provided to plot growth curves, this can be turned off with `draw` = FALSE.

References

Brey, T., Soriano, M., and Pauly, D. 1988. Electronic length frequency analysis: a revised and expanded user's guide to ELEFAN 0, 1 and 2.

Pauly, D. 1981. The relationship between gill surface area and growth performance in fish: a generalization of von Bertalanffy's theory of growth. *Meeresforschung*. 28:205-211

Pauly, D. and N. David, 1981. ELEFAN I, a BASIC program for the objective extraction of growth parameters from length-frequency data. *Meeresforschung*, 28(4):205-211

Pauly, D., 1985. On improving operation and use of ELEFAN programs. Part I: Avoiding "drift" of K towards low values. *ICLARM Conf. Proc.*, 13-14

Pauly, D., 1987. A review of the ELEFAN system for analysis of length-frequency data in fish and aquatic invertebrates. *ICLARM Conf. Proc.*, (13):7-34

Pauly, D. and G. R. Morgan (Eds.), 1987. Length-based methods in fisheries research. (No. 13). WorldFish

Pauly, D. and G. Gaschuetz. 1979. A simple method for fitting oscillating length growth data, with a program for pocket calculators. I.C.E.S. CM 1979/6:24. Demersal Fish Comittee, 26 p.

Pauly, D. 1984. Fish population dynamics in tropical waters: a manual for use with programmable calculators (Vol. 8). WorldFish.

Quenouille, M. H., 1956. Notes on bias in estimation. *Biometrika*, 43:353-360

Somers, I. F., 1988. On a seasonally oscillating growth function. ICLARM Fishbyte 6(1): 8-11.

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2): 407 p.

Tukey, J., 1958. Bias and confidence in not quite large samples. *Annals of Mathematical Statistics*, 29: 614

Tukey, J., 1986. The future of processes of data analysis. In L. V. Jones (Eds.), The Collected Works of John W. Tukey philosophy and principles of data analysis: 1965-1986 (Vol. 4, pp. 517-549). Monterey, CA, USA: Wadsworth & Brooks/Cole

Examples

```
data(alba)
res <- lfqRestructure(alba)

# simple plot or reconstructed frequencies
plot(x = res, Fname = "rcounts")

# add VBGF curves
plot(res, Fname = "rcounts", par = list(Linf = 14, K = 1.1, t_anchor = 0.3))

# add soVBGF curves, adjust hist.sc and xlim
plot(res, Fname = "catch", curve.col=4,
  par = list(Linf = 14, K = 1.1, t_anchor = 0.3, C = 0.2, ts = 0.75),
  hist.sc = 0.9,
  xlim=range(res$dates)+c(-30, 0)
)

# adjust image colors
plot(res, Fname = "rcounts", image.col = NA )
plot(res, Fname = "rcounts", image.col = rev(cm.colors(21)) )
plot(res, Fname = "rcounts", image.col = colorRampPalette(c("red","grey90","green"))(21))

# solid plot region color
plot(res, xlim=range(res$dates)+c(-60, 60),
  hist.sc=0.75, image.col="grey90") # leaves gaps
plot(res, xlim=range(res$dates)+c(-60, 60),
  hist.sc=0.75, region.col="grey90") # full coverage

# low-level plot additions
plot(res)
abline(h=4, lty=2)
mtext("Restructured frequencies (MA=5)", line=0.25, side=3)
```

`plot.predict_mod` *Plotting prediction models yield per recruit and Thompson & Bell*

Description

This function plots objects of the class "predict_mod", which are results of the function `predict_mod`.

Usage

```
## S3 method for class 'predict_mod'
plot(
  x,
  type = "ypr",
  xaxis1 = "FM",
  yaxis1 = "Y_R.rel",
  yaxis2 = "B_R.rel",
  yaxis_iso = "Lc",
  identify = FALSE,
  mark = FALSE,
  contour = TRUE,
  xlab = NA,
  ylab1 = NA,
  ylab2 = NA,
  ylab3 = NA,
  plot_refs = c("F01", "Fmax", "F05", "F04"),
  cols_refs = c("goldenrod2", "darkred", "darkgreen", "darkorange"),
  ...
)
```

Arguments

`x`	a object of the class 'predict_mod'
`type`	a character indicating, which type of plot should be displayed in case of Beverton and Holt's yield per recurit model. Options are either "ypr" (default) for line plot or "Isopleth" for isopleth plot.
`xaxis1`	which x-axis should be plotted? Either "FM" (fishing mortality; default) or "E" (exploitation rate).
`yaxis1`	which (first) y-axis should be plotted? "Y_R" (yield per recruit; default) or "Y_R.rel" (relative yield per recruit) for type = "ypr". For "Isopleth" in addition: "B_R" (biomass per recruit) and "B_R.rel" (relative yield per recruit). For Thompson and Bell model in addition also "value" or "catch" possible.
`yaxis2`	which second y-axis should be plotted for type = "ypr"? Either "B_R" (biomass per recruit; default), "B_R.rel" (relative biomass per recruit), or "B_R.percent" (percentage biomass per recruit)
`yaxis_iso`	determines label and scale of y axis of Isopleth graph. Either "Lc" (default) for length at first capture or "Lc/Linf" for the relation of length at first capture to the infinite length
`identify`	logical; indicating whether points in the graph are supposed to be identified by clicking on them (uses `locator` function). To stop press right mouse click. (default: TRUE).

`mark`	logical; if value of choosen points should be displayed in graph (default: TRUE)
`contour`	used in combination with the Isopleth graph. Usage can be logical (e.g. TRUE) or providing a numeric which indicates the number of levels (`nlevels` in `contour`). By default TRUE.
`xlab`	Label of x-axis. If set to NA, then default "Fishing mortality", or "Exploitation rate" is used.
`ylab1`	Label of y-axis. If set to NA, then default "Yield" is used for the Thompson and Bell model, "Lc" is used for the Isopleth graph, and "Y/R" is used for ypr.
`ylab2`	Label of second y-axis. If set to NA, then default "Biomass" is used for the Thompson and Bell model and "B/R" for ypr.
`ylab3`	Label of third y-axis. If set to NA, then default "Value" is used for the Thompson and Bell model.
`plot_refs`	Define reference points to be plotted. Default: c("F01","Fmax","F05","F04")
`cols_refs`	Define colours for reference points. Default: c("goldenrod2","darkred","darkgreen","darkorange")
`...`	optional parameters of plot function

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
## Not run:
# Nemipterus marginatus - age structured data
threadfin <- list(Winf = 286,K = 0.37, t0 = -0.2, M = 1.1, tr = 0.4)

output <- predict_mod(threadfin, FM_change = seq(0,6,0.1),
   tc_change = seq(0.2,1,0.2), type = 'ypr')
plot(output)

# hake - length structured data
data(hake)
hake$Lr <- 35
select.list <- list(selecType = 'trawl_ogive', L50 = 20, L75 = 24)
output <- predict_mod(param = hake, FM_change = seq(0,0.4,0.05),
                      curr.E = 0.4, curr.Lc = 40,
                      Lc_change = seq(5,80,1), s_list = select.list,
                      type = 'ypr', plot = FALSE)
plot(output, type = "Isopleth", xaxis1 = "FM", yaxis1 = "Y_R.rel",
   identify = FALSE)


## End(Not run)
```

`plot.prod_mod` *Plotting production models*

Description

This function plots CPUE and yield values against fishing effort resulting from the production models (prod_mod).

Usage

```
## S3 method for class 'prod_mod'
plot(x, display_MSY = TRUE, ...)
```

Arguments

`x`	a object of the class `"prod_mod"`
`display_MSY`	logical; should MSY be displayed in the graph?
`...`	optional parameters of plot function

References

Fox, W. W. Jr., 1970. An exponential surplus-yield model for optimizing exploited fish populations. *Trans.Am.Fish.Soc.*, 99:80-88

Graham, M., 1935. Modern theory of exploiting a fishery and application to North Sea trawling. *J.Cons.CIEM*, 10(3):264-274

Schaefer, M., 1954. Some aspects of the dynamics of populations important to the management of the commercial marine fisheries. *Bull.I-ATTC/Bol. CIAT*, 1(2):27-56

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. FAO Fisheries Technical Paper, (306.1, Rev. 2). 407 p.

Examples

```
data(trawl_fishery_Java)
output <-  prod_mod(data = trawl_fishery_Java)
plot(output, display_years = TRUE)
```

`plot.prod_mod_ts` *Plotting time series production models*

Description

This function plots objects of the class "prod_mod_ts".

Usage

```
## S3 method for class 'prod_mod_ts'
plot(x, correlation_plots = FALSE, ...)
```

Arguments

`x`	a object of the class "prod_mod_ts",
`correlation_plots`	logical; indicating if correlation plots should be displayed
`...`	additional parameters of the plot function

References

Dharmendra, D., Solmundsson, J., 2005. Stock assessment of the offshore Mauritian banks using dynamic biomass models and analysis of length frequency of the Sky Emperor (*Lethrinus mahsena*). Fisheries Training Program The United Nations University, 61

Examples

```
data(emperor)
output <-  prod_mod_ts(emperor, method  = "Fox")
plot(output, correlation_plots = TRUE)
```

`plot.recruitment` *Plot of recruitment patterns*

Description

This function plots the recruitment patterns from the recruitment model.

Usage

```
## S3 method for class 'recruitment'
plot(
  x,
  percent = TRUE,
  col = "darkgreen",
  xtitle = "default",
  ytitle = "default",
  ...
)
```

Arguments

`x`	list of the class `"recruitment"`
`percent`	logical; should number of recruits be relative (percentage)?
`col`	colour of bars (default is "darkgreen")
`xtitle`	title of x axis (default "rel. months" or no title, respectively)
`ytitle`	title of y axis (default "# Recruits" or "Recruits [%]", respectively)
`...`	standard parameters of `barplot`

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. FAO Fisheries Technical Paper, (306.1, Rev. 2). 407 p.

Examples

```
dat <- list(midLengths = seq(2,98,4),
              catch = c(0.6,17.6,93,83.2,12.6,0.3,0,0,0,1,17.1,51.4,
              26.1,2.2,0.2,4.5,21.6,17.6,3.7,8.7,10.6,6.3,5.6,2.9,0.8),
              Linf = 100,
              K = 0.5)
output <- recruitment(param = dat, tsample = 0.25)
plot(output, percent = FALSE)
```

`plot.select` *Selectivity plot*

Description

This function plots the selectivity estimates of the function `select`.

Usage

```
## S3 method for class 'select'
plot(x, regression_fit = TRUE, cols = c("darkgreen", "orange"), ...)
```

Arguments

`x`	a list of the class `"select"` containing the results of the gillnet selectivity function.
`regression_fit`	logical; indicating if a plot with the fit of the regression line should be displayed
`cols`	a specification for the two colours of the two selection curves. Default is c("darkgreen","orange").
`...`	additional parameters of the plot function

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
data(tilapia)
output <- select(tilapia, plot = FALSE)
plot(output, regression_fit = TRUE)

data(bream)
output <- select(bream, plot = FALSE)
plot(output, regression_fit = TRUE)
```

`plot.select_Millar` *Millar's selectivity plot*

Description

This function plots the selectivity estimates of Millar's selectivity model (`select_Millar`).

Usage

```
## S3 method for class 'select_Millar'
plot(
  x,
  plotlens = NULL,
  standardise = TRUE,
  deviance_plot = TRUE,
  selectivity_plot = TRUE,
  xlab_dev = "Length [cm]",
  xlab_sel = "Length [cm]",
  ylab_dev = "Mesh size [cm]",
  ylab_sel = "Relative retention",
  title_dev = "Deviance residuals",
  title_sel = NULL,
  ...
)
```

Arguments

`x`	a list of the class `"select_Millar"` containing the results of Millar's selectivity model
`plotlens`	A vector with lengths which should be used for drawing the selection curves
`standardise`	A parameter indicating if the retention should be realtive to the maximum value (Default: TRUE).
`deviance_plot`	logical (Default: TRUE); indicating whether a plot of deviance residuals should be displayed
`selectivity_plot`	logical (Default: TRUE); indicating whether a plot of relative retention selectivities should be displayed
`xlab_dev`	character string. Label for x axis of deviance plot. Default: "Length [cm]"
`xlab_sel`	character string. Label for x axis of selectivity plot. Default: "Length [cm]"
`ylab_dev`	character string. Label for y axis of deviance plot. Default: "Mesh size [cm]"
`ylab_sel`	character string. Label for y axis of selectivity plot. Default: "Relative retention".
`title_dev`	character string. Label for main title of deviance plot. Default: "Deviance residuals".
`title_sel`	character string. Label for main title of selectivity plot. Default is taken from the results of the select_Millar (e.g. res$rtype).
`...`	additional parameter options from plot function

Details

This function draws a selectivity plot for the object class `"select_Millar"`, which is created by applying Millar's selectivity model `select_Millar`.

References

Millar, R. B., Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3), 471-477.

Examples

```
data(gillnet)

output <- select_Millar(gillnet, x0 = c(60,4), rel.power = rep(1,8),
   rtype = "norm.loc", plot = FALSE)

plot(output, plotlens = seq(40,90,0.1), deviance_plot = FALSE)
```

`plot.VPA` *VPA plot*

Description

This function plots the survivors, catches, natural losses, and fishing mortality resulting from the VPA model.

Usage

```
## S3 method for class 'VPA'
plot(
  x,
  yaxis = "numbers",
  display_last_class = TRUE,
  xlabel = NA,
  ylabel1 = "Population",
  ylabel2 = "Fishing mortality",
  ylim = NA,
  ylim_FM = NA,
  plot.bars = TRUE,
  plot.FM = TRUE,
  plot.legend = TRUE,
  ...
)
```

Arguments

`x`	list of the class `"VPA"` containing the results of the VPA model.
`yaxis`	indicating which variable should be displayed on the y axis, either "numbers" or "biomass".
`display_last_class`	logical; should last age/length class be displayed in graph?
`xlabel`	Label of the x axis
`ylabel1`	Label of the first y axis
`ylabel2`	Label of the second y axis
`ylim`	limits of y axis
`ylim_FM`	limits of y axis of fishing mortality plot
`plot.bars`	logical; should the barplot of survivors, nat.losses and catch be displayed? (Default: TRUE)
`plot.FM`	logical; should the fishing mortality be displayed in the graph? (Default: TRUE)
`plot.legend`	logical; should a legend be displayed in the graph? (Default: TRUE)
`...`	standard parameters of `barplot`

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. FAO Fisheries Technical Paper, (306.1, Rev. 2). 407 p.

Examples

```
data(whiting)
output <- VPA(whiting, terminalF = 0.5)
plot(output, display_last_class = FALSE)

data(hake)
output <- VPA(hake, terminalE = 0.5, catch_unit = "'000")
plot_mat <- output$plot_mat[,-c(1,2)]  # remove first two length classes
class(plot_mat) <- "VPA"
plot(plot_mat, xlabel = "Midlengths [cm]")
```

`powell_wetherall` *Powell-Wetherall method*

Description

A method to estimate the instantaneous total mortality rate (Z) and the infinite length of the von Bertalanffy growth equation (Powell, 1979; Wetherall et al., 1987).

Usage

```
powell_wetherall(
  param,
  catch_columns = NA,
  savePlots = FALSE,
  reg_int = NULL,
  main = "Powell-Wetherall plot"
)
```

Arguments

`param`	a list consisting of following parameters: • `midLengths`: midpoints of the length groups, • `Linf`: infinite length for investigated species [cm], • `K`: growth coefficent for investigated species [1/year], • `t0`: theoretical time zero, at which individuals of this species hatch, • `catch`: catch as vector, or a matrix with catches of subsequent years;
`catch_columns`	optional; in case catch is a matrix or data.frame, a number indicating which column of the matrix should be analysed (Default: `NA`).
`savePlots`	logical; if TRUE the plot is recorded. Default is FALSE.

`reg_int`	instead of using the identity method a range can be determined, which is to be used for the regression analysis. If equal to NULL identity method is applied (default).
`main`	title of plot (Default is "Powell-Wetherall plot")

Details

The first length group or age class within the list object `midLengths` or `age` will be used as the Lprim or tprime (length of recruitment to fishery). This function includes the `identify` function, which asks you to choose two points from a graph manually. The two points which you choose by clicking on the plot in the graphical device represent the start and end of the data points, which should be used for the analysis. Based on these points the regression line is calculated. The Powell and Wetherall method only works with length-frequency data.

Value

A list with the input parameters and follwing objects:

- **tmean** or **Lmean**: mean age or length of fish,
- **Z**: total mortality;

and/or following objects when applying the Powell and Wetherall method:

- **Lmean_Lprime**: dependent variable for regression analysis,
- **Lprime**: some length for which all fish of that length and longer are under full exploitation,
- **Linf_est**: infinite length in [cm] (Linf),
- **se_Linf**: standard error of Linf,
- **confidenceInt_Linf**: confidence interval for Linf,
- **ZK**: total mortality divided by K (Z/K),
- **se_ZK**: standard error of Z/K,
- **confidenceInt_ZK**: confidence interval of Z/K;

References

Powell, D.G., 1979. Estimation of mortality and growth parameters from the length- frequency of a catch. *Rapp.P.-v.Reun.CIEM*, 175:167-169

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Wetherall, J.A., J.J. Polovina and S. Ralston, 1987. Estimating growth and mortality in steady-state fish stocks from length-frequency data. *ICLARM Conf.Proc.*, (13):53-74

Examples

```
data(synLFQ3)
powell_wetherall(synLFQ3)

data(synLFQ5)
```

```
powell_wetherall(synLFQ5, catch_columns = 1:12)
```

`predict_mod` *Prediction models*

Description

This function applies Beverton & Holt's yield per recruit model as well as the Thompson & Bell model. These models predict catch, yield, biomass and economic values for different fishing mortality scenarions (in combination with gear changes).

Usage

```
predict_mod(
  param,
  type,
  FM_change = NA,
  E_change = NA,
  FM_relative = FALSE,
  Lc_change = NULL,
  tc_change = NULL,
  s_list = NA,
  stock_size_1 = NA,
  age_unit = "year",
  curr.E = NA,
  curr.Lc = NA,
  plus_group = NA,
  Lmin = NA,
  Lincr = NA,
  plot = FALSE,
  mark = TRUE,
  hide.progressbar = FALSE
)
```

Arguments

`param` a list consisting of following parameters (not all are required):

- **Linf** or **Winf**: infinite length or weight, respectively, for investigated species in cm [cm],
- **K**: growth coefficent for investigated species per year [1/year],
- **t0**: theoretical time zero, at which individuals of this species hatch,
- **M**: natural mortality or
- **Z**: total mortality,
- **FM**: fishing mortality,

	• **a**: length-weight relationship coefficent (W = a * L^b), • **b**: length-weight relationship coefficent (W = a * L^b), • **Lr** or **tr**: length or age of recruitment; additional list objects for the Thompson and Bell model: • **midLengths** or **age**: midpoints of the length classes (length-frequency data) or ages (age composition data), • **meanWeight**: vector with mean weight per length group or age class, • **meanValue**: vector with mean value per length group or age class,
`type`	indicating which model should be applied: `"ypr"` for Beverton and Holt's yield per recruit model and `"ThompBell"` for the Thompson and Bell model
`FM_change`	vector with ascending fishing mortalities (if `FM_relative` is set to TRUE, values can also be relative (only for Thompson and Bell model). Default are absolute values from 0 to 10). Or
`E_change`	vector with ascending absolute exploitation rates;
`FM_relative`	logical; indicating whether `FM_change` is relative or absolute. Default is FALSE (absolute fishing mortalities in `FM_change`).
`Lc_change`	vector with ascending lengths at first capture (Lc), or
`tc_change`	vector with ascending ages at first capture (tc)
`s_list`	list with selectivity parameters
`stock_size_1`	stock size of smallest size class, if NA values are calculated relative to a stock size of 1000 individuals
`age_unit`	in which time unit the data is provided? "month" or "year"
`curr.E`	current exploitation rate if available
`curr.Lc`	current Lc (length at first capture) if available
`plus_group`	if a value is provided, a plus group is created comprising this size class and all above
`Lmin`	smallest length group where to start with selection ogive. Not required for "knife_edge" selection type
`Lincr`	arbitrary length increment between length groups for estimation of selection ogive. The smaller the higher the resolution but the slower the model run. Not required for "knife_edge" selection type
`plot`	logical; if TRUE results are displayed graphically
`mark`	logical; if value of choosen points should be displayed in graph (default: TRUE)
`hide.progressbar`	logical; should progressbar be displayed or hidden? (Default: FALSE)

Details

The Thompson and Bell model incorporates an iteration step simulating the stock by means of the `stock_sim` function. In case changes in gear characteristics - here measured in terms of Lc or tc, the length or age at first capture, respectively - should be explored, a list with selectivity information about the gear has to be provided and the prediction models make use of the selectivity `select_ogive` function. Sparre and Venema (1998) recommend to treat the last length class always as plus group. This model is very sensitive to zero observations in the ultimate length classes. If unrealistic results are returned, it is recommended to cut length classes with zero observations, group them in a plus group or to change the interval between length classes. Equations which are used in this function assume isometric growth, an assumption often not met. Further, the assumption that there is no relationship between the parental stock size and progeny over a wide range of fishing mortalities or exploitation values, respectively, is also said to be untrue. By default, the functions assume knife-edge recruitment and selection of gears (Sparre and Venema, 1998). If E_change instead of FM_change is used the range is cut at E=0.9, because higher values of E correspond to unrealistic high values of fishing mortality. If no selectivity information is given (by use of s_list), knife edge selectivity with L50 equal to the first argument of Lc_change is assumed.

Value

A list with the input parameters and dependent on the model type following list objects:

- `type = 'ypr'`
  - **FM**: fishing mortalities,
  - **Lc** or **tc**: lengths or ages at first capture,
  - **list_Lc_runs**: a list with dataframes for each Lc value:
    - **FM_change**: fishing mortalities
    - **E**: expoitation rates
    - **Ty**: mean age in annual yield
    - **LY**: mean length in annual yield
    - **Wy**: mean weight in annual yield
    - **Y_R.rel**: relative yield per recruit (change in catch in weigth per recruit relative to initial Y/R value)
    - **B_R.rel**: relative biomass per recruit
    - **Y_R**: yield per recurit (catch in weight per recruit)
    - **B_R**: biomass per recruit
    - **B_R.percent**: percentage biomass per recurit in relation to virgin biomass per recruit
  - **df_Es**: a dataframe with references points (columns) for different Lc values (rows)
  - **df_current**: a dataframe with the exploitation status, yield and biomass values of current exploitation or selectivity (if E_curr or Lc_tc_curr provided).
- `type = 'ThompBell'`
  - **dt**: delta t,
  - **N**: population number,
  - **dead**: deaths due to natural reasons,
  - **C**: catch,
  - **Y**: yield,
  - **B**: biomass,
  - **V**: value,
  - **totals**: summed up values (total catch, total yield, total value, average biomass),
  - **totC**: total catches for different x factors,
  - **totY**: total yield values for different x factors,

  - **totV**: total values for different x factors,
  - **meanB**: average biomasses for different x factors,
  - **F_change**: fishing mortality changes;
- `type = 'ThompBell'` and `Lc_change` provided
  - **FM_change**: fishing mortality changes,
  - **Lc_change**: changes in length at first capture,
  - **Lt**: lengths at age,
  - **sel**: probability of capture,
  - **mat_FM_Lc_com.C**: catch matrix for all fishing mortality and Lc/tc combinations,
  - **mat_FM_Lc_com.Y**: yield matrix for all fishing mortality and Lc/tc combinations,
  - **mat_FM_Lc_com.V**: value matrix for all fishing mortality and Lc/tc combinations,
  - **mat_FM_Lc_com.B**: biomass matrix for all fishing mortality and Lc/tc combinations;

References

Berkeley, S.A., and Houde, E.D., 1980. Swordfish, Xiphias gladius, dynamics in the Straits of Florida. *ICES C.M.*, 11.

Beverton, R.J.H., and Holt, S.J., 1964. Table of yield functions for fishery management. *FAO Fish. Tech. Pap.* 38, 49 p.

Beverton, R.J.H., and Holt, S.J., 1966. Manual of methods for fish stock assessment. Pt. 2: Tables of yield functions. *FAO Fisheries Technical Paper*, (38)Rev.1:67 p.

Boerema, L.K., and J.A. Gulland, 1973. Stock assessment of the Peruvian anchovy (Engraulis ringens) and management of the fishery. *Journal of the Fisheries Board of Canada*, 30(12):2226-2235

Garcia, S. and N.P. van Zalinge, 1982. Shrimp fishing in Kuwait: methodology for a joint analysis of the artisanal and industrial fisheries. pp. 119-142 In: Report on the Workshop on assessment of the shrimp stocks of the west coast of the Gulf between Iran and the Arabian Peninsula. Fisheries development in the Gulf. Rome, FAO, FI:DP/RAB/80/015/1, 163 p.

Gulland, J.A., 1983. Fish stock assessment: a manual of basic methods. *FAO/Wiley*, New York.

Gulland, J.A. and Boerema, L., 1973. Scientific advice on catch levels. *Fish. Bull. (US)* 71:325-335.

Jones, R.E. 1957. A much simplified version of the fish yield equation. Doc. No. P. 21. Paper presented at the Lisbon joint meeting of International Commission Northwest Atlantic-Fisheries, International Council for the Exploration of the Sea, and Food and Agriculture Organization of the United Nations. 8 p. [Mimeo].

Millar, R.B., and Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3):471-477

Pauly, D., 1980. A selection of simple methods for the assessment of tropical fish stocks. *FAO Fisheries Circulars (FAO).* no. 729.

Pauly, D., 1984. Fish population dynamics in tropical waters: a manual for use with programmable calculators. *ICLARM* Stud. Rev. 8, 325 p.

Pauly, D. and M. Soriano. 1986. Some practical extensions to Beverton and Holt's relative yield-per-recruit model, p. 491-495. In J.L. Maclean, L.B. Dizon and L.V. Hosillos (eds.) The First Asian Fisheries Forum. Asian Fisheries Society, Manila.

Schaefer, M.B., 1954. Some aspects of the dynamics of populations important to the management of the commercial marine fisheries. *Inter-Am. Trop. Tuna Comm., Bull.* 1(2):27-56.

Schaefer, M.B., 1957. A study of the dynamics of the fishery for yellowfin tuna in the eastern tropical Pacific Ocean. *Inter-Am. Trop. Tuna Comm., Bull.* 2:247-268.

Sparre, P., and Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
#______________________________________
# Yiel Per Recruit (YPR) / Beverton and Holt's model
#______________________________________
# age structured data
# Nemipterus marginatus
threadfin <- list(Winf = 286, K = 0.37, t0 = -0.2, M = 1.1, tr = 0.4)

predict_mod(threadfin, FM_change = seq(0,6,0.1),
   tc_change = seq(0.2,1,0.2), type = 'ypr')

# Leiognathus spendens (Pauly, 1980)
ponyfish <- list(Winf = 64, K = 1, t0 = -0.2, M = 1.8, tr = 0.2)

predict_mod(ponyfish, tc_change = c(0.2,0.3,1.0), type = 'ypr', plot=TRUE)

#______________________________________
# length structured data
# Xiphias gladius (Berkeley and Houde, 1980)
swordfish <- list(Linf = 309, K = 0.0949, M = 0.18,
                  a = 0.0003, b = 3, Lr = 90)

select.list <- list(selecType = 'trawl_ogive', L50 = 120, L75 = 132)
#swordfish$midLengths <- seq(60,300,5)

output <- predict_mod(param = swordfish, Lc_change = c(100,118,150,180),
            s_list = select.list, type = 'ypr', Lmin = 90, Lincr = 8)
plot(output)

data(hake)
hake$Lr <- 35
select.list <- list(selecType = 'trawl_ogive', L50 = 50, L75 = 54)
output <- predict_mod(param = hake, FM_change = seq(0,3,0.05),
                      Lc_change = seq(30,70,1), s_list = select.list,
                      type = 'ypr', plot = FALSE, curr.Lc = 50, curr.E = 0.73)
plot(output, type = "Isopleth", xaxis1 = "FM", yaxis1 = "Y_R.rel", mark = TRUE)

output <- predict_mod(param = hake, E_change = seq(0,1,0.1),
                      Lc_change = seq(2,120,2), #s_list = select.list,
                      type = 'ypr', plot = FALSE)
plot(output, type = "Isopleth", xaxis1 = "E", yaxis1 = "B_R")

#______________________________________
#      Thompson and Bell model
```

```
#______________________________________
# with age structured data
data(shrimps)

output <- predict_mod(param = shrimps, FM_change = seq(0.1,20,0.1),
     type = "ThompBell", age_unit = "month", plot = TRUE)

#______________________________________
# with length structured data
data(hake)
par(mar = c(5, 4, 4, 7))
predict_mod(param = hake,FM_change = seq(0.1,3,0.05),
     type = 'ThompBell', plot = TRUE)

# create list with selectivity information
select.list <- list(selecType = 'trawl_ogive', L50 = 50, L75 = 55)

output <- predict_mod(param = hake, FM_change = seq(0,2,0.1),
     Lc_change = seq(20,70,1),
     curr.E = 0.4, curr.Lc = 50,
     type = 'ThompBell', s_list = select.list)
plot(output, xaxis1 = "FM", yaxis_iso = "Lc", yaxis1 = "B_R", mark = TRUE)
```

`prod_mod` *Production models*

Description

Production models are holistic models, which can be used to estimate maximum sustainable yield (MSY) and virgin biomass. This function uses the equilibrium approach to estimate parameters (Schaefer model and Fox model).

Usage

```
prod_mod(data, plot = FALSE)
```

Arguments

`data`	a dataframe consisting of: • `year` year vector, • `Y` catch in weight per year, and • `f` fishing effort per year, or • `CPUE` catch per unit of effort per year (optional).
`plot`	logical; if TRUE, a graph is displayed

Details

Production models are also called surplus production models or biomass dynamic models. They can be applied if sufficient data are available: effort and yield parameters have to be expended over a certain number of years. Furthermore, the fishing effort must have undergone substantial changes over the period covered (Sparre and Venema, 1998). Either the catch per unit of effort (CPUE) is inserted into the model directly (objectname: `CPUE`) or the CPUE is calculated from the catch and effort, then these two vectors should have required units. There are three ways of estimating paramaters of production models, (i) assuming equlibrium conditions, (ii) transforming equation to linear form, or (iii) time-series fitting (Hilborn and Walters, 1992). The first approach corresponds to the Schaefer and Fox model and thus the methodology of this function. The authors recommend to use dynamic fitting methods when possible rather than the equilibrium approach. For dynamic production models please refer to prod_mod_ts.

Value

A list with the input parameters and following list objects:

- **Schaefer_lm**: intercept and slope of linear model following the Schaefer model,
- **Fox_lm**: intercept and slope of linear model following the Fox model,
- **Schaefer_MSY**: MSY according to Schaefer model,
- **Schaefer_fMSY**: fishing effort yielding in MSY according to Schaefer model,
- **Schaefer_Bv**: virgin biomass according to Schaefer model,
- **ln_CPUE**: natural logarithm of CPUE values,
- **Fox_MSY**: MSY according to Fox model,
- **Fox_fMSY**: fishing effort yielding in MSY according to Fox model,
- **Fox_Bv**: virgin biomass according to Fox model.

References

Fox, W. W. Jr., 1970. An exponential surplus-yield model for optimizing exploited fish populations. *Trans.Am.Fish.Soc.*, 99:80-88

Graham, M., 1935. Modern theory of exploiting a fishery and application to North Sea trawling. *J.Cons.CIEM*, 10(3):264-274

Hilborn, R., Walters, C. J. (1992). Quantitative fisheries stock assessment: choice, dynamics and uncertainty. *Reviews in Fish Biology and Fisheries*, 2(2), 177-178.

Schaefer, M., 1954. Some aspects of the dynamics of populations important to the management of the commercial marine fisheries. *Bull.I-ATTC/Bol. CIAT*, 1(2):27-56

Schaefer, M., 1957. A study of the dynamics of the fishery for yellowfin tuna of the eastern tropical Pacific Ocean [in English and Spanish]. *Ibid.*, 2(6): 245-285

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. FAO Fisheries Technical Paper, (306.1, Rev. 2). 407 p.

Examples

```
data(trawl_fishery_Java)
prod_mod(data = trawl_fishery_Java, plot = TRUE)
```

`prod_mod_ts` *Production models with time series fitting*

Description

This function applies the production models under non-equilibrium conditions by applying time series fitting using non-linear least squares minimisation.

Usage

```
prod_mod_ts(
  data,
  method = "Schaefer",
  B0_init = NA,
  B0_est = NA,
  effort_unit = 1,
  plot = TRUE
)
```

Arguments

`data`	a dataframe of parameters • `year` years, • `yield` catch in weight of fishery per year, • `effort` fishing effort per year, • `CPUE` catch per unit of effort per year (optional).
`method`	indicating if Schaefer or Fox model should be applied. First assumes a logistic relationship between growth rate and biomass, whereas second assumes it to foolow the Gompertz distribution (Richards 1959). Default is the dynamic Schaefer model.
`B0_init`	numeric; if realistic initial estimate for virgin biomass is available. If NA initial estimate for virgin biomass is set to two times average yield of all or part of yield values (see `B0_est`).
`B0_est`	intital value of virgin biomass estimating using all yield values (NA) or first years of time series, then provide numerical representing number of years
`effort_unit`	multiplication factor for the unit of effort. Default is 1.
`plot`	logical; if TRUE (default) a graph is displayed

Details

Either catch per unit of effort (CPUE) is inserted into the model directly (by a column `CPUE`) or CPUE is calculated from the catch and effort, then these two vectors should have required units. Whenever a good estimate for the virigin biomass is available, this estimate should be inserted for `B_init`. The default approach for the initial estimate of the virgin biomass is to multiply the average yield by 2 (Dharmendra and Solmundsson, 2005). Alternatively, just a part of the time series of yield values can be choosen to represent the virgin biomass. The minimisation procedure is based on least error sum of squares (SSE). For the logistic (Schaefer) method the standard calculation of SSE is applied (`sum((CPUE - predicted CPUE)^2)`), for the method with Gompertz distribution (Fox) SSE is calculated according to the Thiel's U statistic `sqrt(sum(CPUE - predicted CPUE)/sum(CPUE(t) - CPUE(t-1)))` (Wittink, 1988).

Value

A list with the input parameters and following list objects:

- **Bvec**: biomass vector,
- **CPUE_hat**: predicted CPUE,
- **K**: carrying capacity,
- **r**: population growth rate,
- **q**: catchability coefficient,
- **MSY**: maximum sustainabale yield (MSY),
- **Bmsy**: biomass at MSY,
- **Emsy**: fishing effort at MSY
- **Fmsy**: fishing mortality at MSY,

References

Dharmendra, D., Solmundsson, J., 2005. Stock assessment of the offshore Mauritian banks using dynamic biomass models and analysis of length frequency of the Sky Emperor (*Lethrinus mahsena*). Fisheries Training Program The United Nations University, 61

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

Prager, M. H., 1994. A suite of extensions to a non-equilibrium surplus production model. *Fishery Bulletin* 92: 374-389

Richards, F. J., 1959. A flexible growth function for empirical use. *Journal of experimental Botany*, 10(2), 290-301.

Wittink, D. R., 1988. The application of regression analysis. Allyn and Bacon. Inc. Boston. MA. 324p.

Examples

```
data(emperor)
prod_mod_ts(emperor, method = "Schaefer")
prod_mod_ts(emperor, method = "Fox")
```

`rcurves_Millar` *Predict gillnet selectivity (old Millar method)*

Description

Predict gillnet selectivity (old Millar method)

Usage

```
rcurves_Millar(type, meshsizes, rel, pars, plotlens)
```

Arguments

`type`	("norm.loc", "norm.sca", "gamma", "lognorm")
`meshsizes`	mesh sizes
`rel`	relative powers (one for each mesh size)
`pars`	selection curve parameters
`plotlens`	vector of new mesh sizes for selectivity prediction

Value

selectivities

Source

https://www.stat.auckland.ac.nz/~millar/selectware/

References

Millar, R. B., Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3):471-477

`recruitment` *Recruitment patterns*

Description

This function estimates recrutiment patterns from length-frequency data.

Usage

```
recruitment(param, tsample, catch_column = NA, plot = FALSE)
```

Arguments

`param`	a list consisting of following parameters: • `midLengths`: midpoints of the length classes (length-frequency data), • `Linf`: infinite length for investigated species in cm [cm], • `K`: growth coefficent for investigated species per year [1/year], • `t0`: theoretical time zero, when growth curve crosses length equalling zero, • `D`: optional; for generalised vBGF • `C`: optional; intensity of the (sinusoid) growth oscillations of the model, • `ts`: optional; onset of the positive phase of the growth oscillation (fraction of year; i.e. Jan 1 = 0, July 1 = 0.5, etc.), • `catch`: catches, vector or matrix with catches of subsequent sampling times
`tsample`	sampling time corresponding to time when catch was sampled as fraction of year (e.g. 0.25 for 1st of April). If catch was sampled several times (catch as matrix) vector has to be provided with sampling times in equal order.
`catch_column`	numeric; if catch in param is a matrix, this number indicates the column of the catch matrix which should be used for the analysis.
`plot`	logical; indicating whether a plot should be printed

Details

This function calculates recruitment patterns of a stock by backward projection onto the length axis of a set of length frequency data using the (special, generalised or seasonalised) von Bertallanfy growth curve (vBGF, Pauly 1982). The method assumes that (i) all fish in a data set grow as described by a single set of growth parameters and (ii) one month out of twelve always has zero recruitment. The second assumption is probably not met, since temperate species may contain more than one month with zero recruitment, while tropical species may have more constant recruitment without months of no recruitment. If t0 is not provided, a relative recruitment pattern will be estimated without specific month values returned in the results. However, an estimate of t0 can be obtained by the time lag between peak spawning and recruitment. Several length-frequency data sets can be used to estimate the recrutiment pattern by providing catch as a matrix and setting catch_column to NA (default). Then the fraction per time is calculated for each size class in each sample and then pooled together. For the generalised vBGF, D is required, for the seasonalised vBGF C, ts and D.

Value

A list with the input parameters and following list objects:

- **ti**: actual age,
- **tS_frac**: age at which the length was zero expressed as fraction of the year,
- **cor_months**: corresponding months,
- **months**: numeric months (relative if no t0 is not given),
- **months_abb**: months (only presented if t0 is given),
- **all_recruits**: number of recruits per month as matrix if several length-frequency data sets are provided,

- **mean_recruits**: (mean) number of recruits per month,
- **per_recruits**: precentage number of recruits per month.

References

Brey, T., Soriano, M., Pauly, D., 1988. Electronic length frequency analysis. A revised and expanded user's guide to ELEFAN 0, 1 and 2. (Second edition). Berichte aus dem Institut f??r Meereskunde Kiel, No 177, 31p.

Moreau, J., & Cuende, F. X., 1991. On improving the resolution of the recruitment patterns of fishes. *Fishbyte*, 9(1), 45-46.

Pauly, D., 1982. Studying single-species dynamics in a tropical multispecies context. In Theory and management of tropical fisheries. *ICLARM Conference Proceedings* (Vol. 9, No. 360, pp. 33-70).

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
# one sample
dat <- list(midLengths = seq(2,98,4),
               catch = c(0.6,17.6,93,83.2,12.6,0.3,0,0,0,1,17.1,51.4,
               26.1,2.2,0.2,4.5,21.6,17.6,3.7,8.7,10.6,6.3,5.6,2.9,0.8),
               Linf = 100,
               K = 0.5,
               t0 = 0)
recruitment(param = dat, tsample = 0.25)


# several samples
data(synLFQ4)

# add growth parameters
synLFQ4$Linf <- 80
synLFQ4$K <- 0.5
synLFQ4$t0 <- 0.25

# retrieve sampling times from catch matrix
s_dates <- as.POSIXlt(synLFQ4$dates, format="%d.%m.%Y")

recruitment(param = synLFQ4, tsample = s_dates$yday/365, plot = TRUE)


plot(synLFQ4, Fname = "catch",
   par = list(Linf = 80, K = 0.5, t_anchor = 0.25, C = 0.75, ts = 0),
   ylim = c(0,80))
```

`rtypes_Millar` *Millar's selectivity types*

Description

This function returns a function corresponding to the type of curve which was selected to represent the selectivity of nets or hooks.

Usage

```
rtypes_Millar(rtype)
```

Arguments

`rtype`	a character string indicating which method for the estimation of selection curves should be used: `"norm.loc"` for normal with common spread method, `"norm.sca"` for normal with variable spread method, `"lognorm"` for lognormal method, `"binorm.sca"` for bi-normal method, `"bilognorm"` for bi-lognormal method, `"tt.logistic"` for control and logistic method, `"gamma"` for gamma method.

Details

Function adapted from the selectivity functions provided by Prof. Dr. Russell Millar (https://www.stat.auckland.ac.nz/~millar
Until now following curves are incorporated: `"norm.loc"` for a normal curve with common spread, `"norm.sca"` for a normal curve with variable spread, `"lognorm"` for a lognormal curve, `"binorm.sca"` for a bi-normal curve, `"bilognorm"` for a bi-lognormal curve, `"tt.logistic"` for a control and logistic curve.

Source

https://www.stat.auckland.ac.nz/~millar/selectware/

References

Millar, R. B., Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3), 471-477.

`select` *Selectivity model*

Description

This function estimates the selecitvity of gillnets and trawl nets from experimental catches.

Usage

```
select(param, plot = FALSE)
```

Arguments

`param`	a list with following parameters: • `type` a string indicating which type of gear was used (options: "gillnet" or "trawl_net"), • `midLengths` midlengths of size classes • `meshSizes` a vector with mesh sizes in increasing order, • `CatchPerNet_mat` a matrix with the catches per net in corresponding order of mesh sizes
`plot`	logical; should the results be displayed graphically?

Details

This function estimates the fractions retained by each net, the optimum lengths for each net, the selection factor (SF), and the standard deviation of the factor (stand.dev). Calculations are based on a normal distribution with common spread. Assumptions of this method are, that (i) the optimum length Lm is proportional to the mesh size (Lm = SF * m), (ii) the selection curves are normally distributed with a common standard deviation, (iii) the nets have the same fishing power (same dimensions and material). Requirements for the experimental set-up are: selection curves corresponding to the two mesh sizes have to overlap, and the nets have to be set in the same area, during the same time. To calculate selection factor (SF), L25, L50 and L75 for trawl nets /fisheries.

Value

A list with the input parameters and following list objects for type = "gillnet": #'

- **classes.num**: numeric vector with length classes without a plus group,
- **SNet1**: selection ogive net 1,
- **SNet2**: selection ogive net 2,
- **lnNet2_Net1**: logarithm ratio between nets,
- **linear_mod**: linear model,
- **LmNet1**: optimum length net 1,
- **LmNet2**: optimum length net 2,
- **SF**: selection factor,
- **stand.dev**: standard deviation of selection factor.

and following objects for type = "trawl_net":

- **classes.num**: numeric vector with length classes,
- **SLobs**: observed selection ogive,
- **SLest**: estimated selection ogive,
- **lnSL**: logarithm of observed selection,
- **linear_mod**: linear model,
- **S1**: constant of selection curve,
- **S2**: another constant of selection curve,

- **L25**: length at which 25
- **L50**: length at which 50
- **L75**: length at which 75
- **SF**: selection factor.

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
# Gillnet selectivity
data(tilapia)
out <- select(param = tilapia)
plot(out)

# Trawl selectivity
data(bream)
select(bream)
```

`select_Millar` *Millar's selectivity model*

Description

This model estimates the selecitvity of different gears from experimental catches.

Usage

```
select_Millar(
  param,
  x0 = NULL,
  rtype = "norm.loc",
  rel.power = NULL,
  plot = TRUE
)
```

Arguments

`param`	A list with following parameters: vector with midlengths of size classes (`midLengths`), vector with meshSizes in increasing order (`meshSizes`), and a matrix with the number of individuals caught with each sized mesh (`CatchPerNet_mat`).
`x0`	A string of initial values for the parameters to be optimized over when applying the function `optim`. When rtype = "norm.loc", "norm.sca", or "lognorm", initial values may be ommitted (`x0 = NULL`), and starting vallues will be estimated using the earlier approach of Millar and Holst (1997).

`rtype`	A character string indicating which method for estimating selection curves should be used: `"norm.loc"` for a normal curve with common spread, `"norm.sca"` for a normal curve with variable spread, `"lognorm"` for a lognormal curve, `"binorm.sca"` for a bi-normal curve, `"bilognorm"` for a bi-lognormal curve, `"tt.logistic"` for a control and logistic curve
`rel.power`	A string indicating the relative power of different meshSizes, must have same length as `meshSizes` (Default: `rel.power = NULL`).
`plot`	logical; should a plot be printed?

Details

Model adapted from the selectivity functions provided by Prof. Dr. Russell Millar (https://www.stat.auckland.ac.nz/~millar/). In the deviance plot open circles correspond to negative, closed to positive residuals. The size of the circles is proportional to the square of the residuals. To assess the model fit by the deviance plot it requires some experience, in general the pattern should be random and the sizes not too big. Please refer to Millar's publications and other publications for comparison. The model can produce errors if the starting values (`x0`) for the optim function are not realistic. Please be aware that if the method is changed the outcoming parameters can greatly vary. Simliarly the starting values have to be adapted when changing the method (`rtype`).

Source

https://www.stat.auckland.ac.nz/~millar/selectware/

References

Millar, R. B., Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3):471-477

Holt, S. J. 1963. A method for determining gear selectivity and its application. *ICNAF Special Publication*, 5: 106-115.

Examples

```
data(haddock)

output <- select_Millar(haddock, x0 = c(-10,0.3,0),
   rtype = "tt.logistic")

plot(output, plotlens=seq(25,35,0.1), deviance_plot = FALSE)
legend("topleft",c("Control","Experimental"), lty=1:2, col=1:2)


# Gillnet
data(gillnet)

# Using inital estimates from old method
select_Millar(gillnet, x0 = NULL, rtype = "norm.loc")$value
select_Millar(gillnet, x0 = NULL, rtype = "norm.sca")$value
select_Millar(gillnet, x0 = NULL, rtype = "lognorm")$value
```

```
# Two rtypes which require starting values
select_Millar(gillnet, x0 = c(55,4,65,4,3), rtype="binorm.sca")
select_Millar(gillnet, x0 = c(4,0.2,4.2,0.1,2), rtype="bilognorm")

# Calculation with finer length resolution
output <- select_Millar(gillnet, x0 = NULL, rtype = "lognorm")
plot(output, plotlens=seq(40,90,0.1))

# Use alternate plot settings
output <- select_Millar(gillnet, x0 = NULL, rtype = "lognorm")
ncolor <- length(output$meshSizes)
plot(output, plotlens=seq(40,90,0.1), deviance_plot = FALSE,
 lty=1, col=rainbow(ncolor))
legend("topleft", col=rainbow(ncolor), legend=output$meshSizes,
 lty=1, title="Mesh size [cm]")

 # deviance plot only
 plot(output, plotlens=seq(40,90,0.1), selectivity_plot = FALSE)


# Stacked trammel net
# The data come from two experiments using different mesh sizes
# This analysis assumes common retention curve in both experiments.
# Note that summary function does not produce residual plot
# since lengths are not unique
data(trammelnet)

output <- select_Millar(trammelnet, x0 = c(25,4),
   rtype="norm.loc", rel.power = rep(1,6))

plot(output,plotlens=seq(10,40,0.1))
```

`select_ogive` *Selectivity patterns*

Description

Based on a few parameters, this function estimates the fraction per length group retained in the net. Different selection curves can be used for the estimation.

Usage

```
select_ogive(s_list, Lt, Lc = NA)
```

Arguments

`s_list` a list with selectivity parameters dependent on the type of selection curve:

- `selecType` type of selection curve used for estimation (options: "knife_edge", "trawl_ogive", "lognormal", "normal_fixed"),

- `Lc` length-at-first-capture (also called L50),
- `meshSizes` a vector with mesh sizes in increasing order,
- `select_p1` selectivity parameter 1 (see Millar and Holst (1997)),
- `select_p2` selectivity parameter 2 (see Millar and Holst (1997)),
- `L75` length at which individuals are caught with a probability of 75

`Lt`	a vector with lengths corresponding to age classes
`Lc`	length-at-first-capture (Default: NA)

Details

This function is embedded within `predict_mod`. `selecType` "knife_edge" only requires a Lc value. "trawl_ogive" requires a Lc (L50) and a L75 value. "lognormal" requires two mesh sizes, an estimate of mu and of sigma. "normal_fixed" requires two mesh sizes with an estimate of the selection factor (SF) and an estimate of sigma.

References

Millar, R. B., Holst, R. (1997). Estimation of gillnet and hook selectivity using log-linear models. ICES Journal of Marine Science: Journal du Conseil, 54(3), 471-477.

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
# create list with selectivity information
select.list <- list(selecType = 'knife_edge',
   Lc = 34, L75 = 37, tc = 5, meshSizes = c(60,80),
   select_p1 = 2.7977, select_p2 = 0.1175)


# create vector with mid lengths
Lt <- seq(5, 50, 0.01)

# knife edge selectivity
sel_ke <- select_ogive(select.list, Lt)

# trawl ogive selectivity
select.list$selecType = "trawl_ogive"
sel_to <- select_ogive(select.list, Lt)

plot(Lt, sel_ke, type = 'l')
lines(Lt, sel_to, col = 'blue')


# Gillnet selectivity ("lognormal" and "normal_fixed")
select.list$selecType <- "lognormal"
sel_log <- select_ogive(select.list, Lt)

select.list$selecType <- "normal_fixed"
select.list$select_p1 <- 0.2
```

```
select.list$select_p2 <- 1.5
sel_nf <- select_ogive(select.list, Lt)

plot(Lt, sel_log, type = 'l')
lines(Lt, sel_nf, col = 'blue')
```

`shrimps` *Shrimp data*

Description

Data from the Kuwait shrimp (*Penaeus semisulcatus*) fishery. They can be used to simulate different management scenarios by applying prediction models (`predict_mod`). Fishing mortalities were estimated from catch data and biomass estimates were obtained from the swept area method. Recuritment to fishery at the age of one month (Tr = 1).

Usage

```
data(shrimps)
```

Format

A list consisting of: 1. age groups [months], 2. meanWeight, 3. meanValue, 4.FM, and 5.M

Source

Garcia, S. and N.P. van Zalinge, 1982. Shrimp fishing in Kuwait: methodology for a joint analysis of the artisanal and industrial fisheries. pp. 119-142 In: Report on the Workshop on assessment of the shrimp stocks of the west coast of the Gulf between Iran and the Arabian Peninsula. Fisheries development in the Gulf. Rome, FAO, FI:DP/RAB/80/015/1, 163 p.

Examples

```
data(shrimps)
str(shrimps)
summary(shrimps)
```

`startingPoint2tanchor` *Convert FiSAT's starting point to t_anchor value*

Description

Starting points returned or chosen within FiSAT are not supported in TropFishR. Instead `t_anchor` takes on the job of anchoring VBGF growth curves on a temporal axis. This function allows to convert FiSAT's starting points to `t_anchor` values

Usage

```
startingPoint2tanchor(param, par, startingLength, startingSample)
```

Arguments

`param`	list with dates, midLengths, and catch
`par`	list with growth parameters 'Linf' and 'K' of VBGF
`startingLength`	starting length as returned by FiSAT, indicating the length within the starting sample cut by a growth curve
`startingSample`	starting sample as returned by FiSAT, indicating the sample which is cut by a growth curve

Value

list with input elements and estimated t_anchor value

Examples

```
data(synLFQ5)
lfqNEW <- startingPoint2tanchor(synLFQ5, par = list(Linf = 92, K = 0.37),
   startingLength = 31, startingSample = 4)
lfqRest <- lfqRestructure(lfqNEW, MA = 11)
plot(lfqRest,par=list(Linf=lfqRest$Linf,K=lfqRest$K,t_anchor=lfqRest$t_anchor))
```

`stock_sim` *Stock simulation*

Description

This function estimates stock size, biomass and yield of a stock from fishing mortality per age class or length group. This function is embedded in the Thompson and Bell model (prediction model: `predict_mod`).

Usage

```
stock_sim(param, age_unit = "year", stock_size_1 = NA, plus_group = NA)
```

Arguments

`param`	a list consisting of following parameters: • `age` or `midLengths`: midpoints of length classes (length-frequency data) or ages (age composition data), • `meanWeight`: mean weight in kg per age class or length group, • `meanValue`: mean value per kg fish per age class or length group, • `FM`: fishing mortality rates per age class or length group, • `M` or `Z`: natural or total instantaneous mortality rate.
`age_unit`	indicates if the age groups are per month (`"month"`) or per year (`"year"`). Default: `"year"`
`stock_size_1`	stock size of smallest age/length group
`plus_group`	indicates age/length group, which should be turned into a plus group (i.e. all groups above are comprised in one group)

Details

better to treat last group always as a plus group... if stock size 1 not provided assumes 1000 as intital population size make sure that FM is also in same unit as the classes, e.g. when classes in months than also FM has to be provided in 1/months

Value

A list with the input parameters and following list objects:

- **dt**: delta t,
- **N**: population numbers,
- **dead**: number of deaths due to natural mortality,
- **C**: catch,
- **Y**: yield,
- **B**: biomass,
- **V**: value,
- **totals**: summarised output:
  - **totC** total catch,
  - **totY** total yield,
  - **totV** total value,
  - **meanB** mean biomass.

,

References

Garcia, S. and N.P. van Zalinge, 1982. Shrimp fishing in Kuwait: methodology for a joint analysis of the artisanal and industrial fisheries. pp. 119-142 In: Report on the Workshop on assessment of the shrimp stocks of the west coast of the Gulf between Iran and the Arabian Peninsula. Fisheries development in the Gulf. Rome, FAO, FI:DP/RAB/80/015/1, 163 p.

Millar, R. B., & Holst, R. (1997). Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3):471-477

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
# age-based stock simulation
data(shrimps)

# option 1: without plus group
stock_sim(shrimps, age_unit = "month")

# option 2: with plus group
stock_sim(param = shrimps, age_unit = "month", plus_group = 11)

# length-based stock simulation
data(hake)

stock_sim(param = hake, stock_size_1 = 98919.3)
```

`synCAA1` *Synthetic Catch-at-age data I*

Description

Synthetic Catch-at-age data from Sparre & Venema (1998). Can be used for the estimation of the instantaneous total mortality rate (Z) by means of `Z_BevertonHolt`.

Usage

```
data(synCAA1)
```

Format

A list consisting of:

- `midAge`: a vector of the mid ages of the age groups,
- `catch`: a matrix with the catches for different years.

Source

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
data(synCAA1)
str(synCAA1)
summary(synCAA1)
```

`synCAA2` *Synthetic Catch-at-age data II*

Description

Synthetic Catch-at-age data from Sparre & Venema (1998). Can be used for the estimation of the instantaneous mortality rate (Z) by means of the cumulative catch curve (`catchCurve`).

Usage

```
data(synCAA2)
```

Format

A list consisting of:

- `midAge` a vector of the mid ages of the age groups,
- `catch` a matrix with the catches for different years.

Source

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
data(synCAA2)
str(synCAA2)
summary(synCAA2)
```

---

synCPUE *Synthetical catch per unit of effort (CPUE) dataset*

Description

Synthetical CPUE dataset from Exercise 4.3 in Sparre & Venema (1999). Can be used to estimate the instantaneous total mortality rate (Z) by means of the function `Z_CPUE`.

Usage

```
data(synCPUE)
```

Format

A dataframe consisting of following columns:

- `cohort`: name of cohort, e.g. 1982 S, meaning summer cohort of 1982,
- `age`: age of cohorts,
- `CPUE`: catch per unit of effort of cohorts.

Source

Sparre, P., Venema, S.C., 1999. Introduction to tropical fish stock assessment. Part 2. Excercises. *FAO Fisheries Technical Paper*, (306.2, Rev. 2). 94 p.

Examples

```
data(synCPUE)
head(synCPUE)
str(synCPUE)
```

---

synLFQ1 *Synthetic length-frequency data I*

Description

Synthetic length-frequency data as provided in Sparre & Venema (1998). Can be used to apply the function `Bhattacharya` or `ELEFAN`.

Usage

```
data(synLFQ1)
```

Format

A list consisting of:

- **dates** dates of sampling times (class Date),
- **midLengths** midpoints of the length classes,
- **catch** matrix with catches/counts per length class (row) and sampling date (column).

Source

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
data(synLFQ1)
str(synLFQ1)
summary(synLFQ1)
```

`synLFQ2` *Synthetic length frequency data II*

Description

Synthetic length frequency data from Sparre & Venema (1998). Can be used for the estimation of the instantaneous total mortality rate (Z) by means of `Z_BevertonHolt`.

Usage

```
data(synLFQ2)
```

Format

A list consisting of:

- **dates** dates of sampling times (class Date),
- `midLengths`: midpoints of the length classes,
- `Linf`: infinite length for investigated species in cm [cm],
- `K`: growth coefficent for investigated species per year [1/year],
- `catch`: matrix with catches/counts per length class (row) and sampling date (column).

Source

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. FAO Fisheries Technical Paper, (306.1, Rev. 2). 407 p.

Examples

```
data(synLFQ2)
str(synLFQ2)
summary(synLFQ2)
```

`synLFQ3` *Synthetic length frequency data III*

Description

Synthetic length frequency data from Sparre & Venema (1998). Can be used for the estimation of the instantaneous total mortality rate (Z) by means of `Z_BevertonHolt`.

Usage

```
data(synLFQ3)
```

Format

A list consisting of:

- `midLengths`: midpoints of the length classes,
- `Linf`: infinite length for investigated species in cm [cm],
- `K`: growth coefficent for investigated species per year [1/year],
- `t0`: theoretical time zero, at which individuals of this species hatch,
- `catch`: a vector with catches.

Source

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
data(synLFQ3)
str(synLFQ3)
summary(synLFQ3)
```

`synLFQ4` *Synthetic length-frequency data IV (with seasonal oscillation)*

Description

Synthetic length-frequency data as generated by the function `lfqGen` from the *fishdynr* package (Taylor 2016). Can be used by `ELEFAN` or `ELEFAN_SA`. The data is generated with the following von Bertalanffy growth parameters:

- K = 0.5
- Linf = 80
- C = 0.75
- ts = 0.5
- t_anchor = 0.25 (Time when yearly recruitment pulse occurs; e.g. 0 = Jan 1, 0.25 = Apr 1, 0.5 = Jul 1st, 0.75 = Oct 1)

Further settings include:

- Linf.cv = 0.05
- K.cv = 0.05
- L50 = 20 (length at 50% prob. of capture via trawl net)
- wqs = 0 (width between 25% and 75% prob. of capture quantiles; i.e. wqs = 0 is knife-edge selection)
- M = 0.5 (Natural mortality rate)
- harvest_rate = 0.5 (Fishing mortality rate)

Usage

```
data(synLFQ4)
```

Format

A list of class lfq consisting of:

- **dates** dates of sampling times (class Date),
- **midLengths** midpoints of the length classes,
- **catch** matrix with catches/counts per length class (row) and sampling date (column).

Source

Taylor, M (2016). fishdynr: Fisheries science related population dynamics models. R package version 0.2. https://github.com/marchtaylor/fishdynr

Examples

```
data(synLFQ4)

# plot of length frequencies
plot(synLFQ4, Fname = "catch")

# plot of restructured length frequencies
synLFQ4 <- lfqRestructure(synLFQ4, MA = 15)
plot(synLFQ4, Fname = "rcounts")
```

`synLFQ5` *Synthetic length-frequency data V (without seasonal oscillation)*

Description

Synthetic length-frequency data as generated by the function `lfqGen` from the *fishdynr* package (Taylor 2016). Can be used by `ELEFAN` or `ELEFAN_SA`. The data is generated with the following von Bertalanffy growth parameters:

- K = 0.5
- Linf = 80
- C = 0
- ts = 0
- t_anchor = 0.25 (Time when yearly recruitment pulse occurs; e.g. 0 = Jan 1, 0.25 = Apr 1, 0.5 = Jul 1st, 0.75 = Oct 1)

Further settings include:

- Linf.cv = 0.05
- K.cv = 0.05
- L50 = 20 (length at 50% prob. of capture via trawl net)
- wqs = 0 (width between 25% and 75% prob. of capture quantiles; i.e. wqs = 0 is knife-edge selection)
- M = 0.5 (Natural mortality rate)
- harvest_rate = 0.5 (Fishing mortality rate)

Usage

```
data(synLFQ5)
```

Format

A list of class lfq consisting of:

- **dates** dates of sampling times (class Date),
- **midLengths** midpoints of the length classes,
- **catch** matrix with catches/counts per length class (row) and sampling date (column).

Source

Taylor, M (2016). fishdynr: Fisheries science related population dynamics models. R package version 0.2. https://github.com/marchtaylor/fishdynr

Examples

```
data(synLFQ5)

# plot of length frequencies
plot(synLFQ5, Fname = "catch")

# plot of restructured length frequencies
synLFQ5 <- lfqRestructure(synLFQ5, MA = 15)
plot(synLFQ5, Fname = "rcounts")
```

`synLFQ6` *Synthetic length-frequency data VI (without seasonal oscillation)*

Description

Synthetic length-frequency data as generated by the function `lfqGen` from the *fishdynr* package (Taylor 2016). Can be used by `ELEFAN` or `ELEFAN_SA`. The data is generated with the following von Bertalanffy growth parameters:

- K = 0.47 +/- 0.2 (CV)
- Linf = 33.7 +/- 0.05 (CV)
- C = 0
- ts = 0
- t_anchor = 0.25 (Time when yearly recruitment pulse occurs; e.g. 0 = Jan 1, 0.25 = Apr 1, 0.5 = Jul 1st, 0.75 = Oct 1)

Further settings include:

- L50 = 10 (length at 50% prob. of capture via trawl net)
- wqs = 3 (width between 25% and 75% prob. of capture quantiles; i.e. wqs = 0 is knife-edge selection)
- M = 0.8 (Natural mortality rate)
- harvest_rate = 1.3 (Fishing mortality rate)

Usage

```
data(synLFQ6)
```

Format

A list of class lfq consisting of:

- **dates** dates of sampling times (class Date),
- **midLengths** midpoints of the length classes,
- **catch** matrix with catches/counts per length class (row) and sampling date (column).

Source

Taylor, M (2016). fishdynr: Fisheries science related population dynamics models. R package version 0.2. https://github.com/marchtaylor/fishdynr

Examples

```
data(synLFQ6)

# plot of length frequencies
plot(synLFQ6, Fname = "catch")

# plot of restructured length frequencies
synLFQ6 <- lfqRestructure(synLFQ6, MA = 7)
plot(synLFQ6, Fname = "rcounts")
```

`synLFQ7` *Synthetic length-frequency data VII with seasonal oscillation*

Description

Synthetic length-frequency data as generated by the function `lfqGen` from the *fishdynr* package (Taylor 2016). Can be used by `ELEFAN`, `ELEFAN_SA`, or `ELEFAN_GA`. The data is generated with the following von Bertalanffy growth parameters:

- K = 0.2 +/- 0.1 (CV)
- Linf = 123 +/- 0.05 (CV)
- C = 0.3
- ts = 0
- t_anchor between 0.16 and 0.34 (Time when yearly recruitment pulse occurs; e.g. 0 = Jan 1, 0.25 = Apr 1, 0.5 = Jul 1, 0.75 = Oct 1; repro_wt = c(0, 0, 0.2, 1, 0.6, 0, 0, 0, 0, 0, 0, 0))

Further settings include:

- Lmat = 30.8 (length at maturity)
- L50 = 35 (length at 50% prob. of capture via trawl net)
- wqs = 5 (width between 25% and 75% prob. of capture quantiles; i.e. wqs = 0 is knife-edge selection)
- M = 0.2 (Natural mortality rate)
- harvest_rate = 0.3 (Fishing mortality rate)

Usage

```
data(synLFQ7)
```

Format

A list of class lfq consisting of:

- **dates** dates of sampling times (class Date),
- **midLengths** midpoints of the length classes,
- **catch** matrix with catches/counts per length class (row) and sampling date (column).

Source

Taylor, M (2016). fishdynr: Fisheries science related population dynamics models. R package version 0.2. https://github.com/marchtaylor/fishdynr

Examples

```
data(synLFQ7)

# plot of length frequencies
plot(synLFQ7, Fname = "catch")

# plot of restructured length frequencies
synLFQ7 <- lfqRestructure(synLFQ7, MA = 11)
plot(synLFQ7, Fname = "rcounts")
```

`synLFQ8` *Synthetic length-frequency data VIII with variable harvest rate*

Description

Synthetic length-frequency data as generated by the function `virtualPop2` from the *fishdynr* package (Taylor and Mildenberger, 2018). Can be used by `ELEFAN`, `ELEFAN_SA`, or `ELEFAN_GA`. The data is generated with the following von Bertalanffy growth parameters:

- K = 0.5 +/- 0.05 (CV)
- Linf = 80 +/- 0.05 (CV)
- C = 0.0
- ts = 0.0
- t_anchor 0.25 (Time when yearly recruitment pulse occurs; e.g. 0 = Jan 1, 0.25 = Apr 1, 0.5 = Jul 1, 0.75 = Oct 1; repro_wt = c(0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0))

Further settings include:

- Lmat.f = 40 (length at maturity)
- Lmat.m = 36 (length at maturity)
- L50 = 20 (length at 50% prob. of capture with trawl net (Lc))
- wqs = 4 (width between 25% and 75% prob. of capture quantiles; i.e. wqs = 0 is knife-edge selection)
- M = 0.7 (Natural mortality rate)
- harvest_rate = 1, 0.8, 0.4, 0.1 for years 2014, 2015, 2016, 2017, repsectively (fishing mortality rate)

The average biomass relative to the unfished biomass and the yield per recurit for the years, 2014, 2015, 2016, 2017 is 0.16, 0.21, 0.41, 0.75, and 0.4, 0.43, 0.24, 0.07, respectively. Biomass over Bmsy as estimated with a surplus production model is 0.51, 0.67, 1.27, 2.18. Lopt is 53.33 cm.

Usage

```
data(synLFQ8)
```

Format

A list of class lfq consisting of:

- **stock** Stock ID or name,
- **dates** dates of sampling times (class Date),
- **midLengths** midpoints of the length classes,
- **catch** matrix with catches/counts per length class (row) and sampling date (column),
- **comment** additional comments to this data set.

Source

Taylor, M & T. K. Mildenberger (2018). fishdynr: Fisheries science related population dynamics models. R package version 0.5. https://github.com/tokami/fishdynr

Examples

```
data(synLFQ8)

# plot of length frequencies
plot(synLFQ8, Fname = "catch")

# plot of restructured length frequencies
synLFQ8 <- lfqRestructure(synLFQ8, MA = 11)
plot(synLFQ8, Fname = "rcounts")
```

`tilapia` *Tilapia data*

Description

A list with characteristics of experimental catches of tilapia *Tilapia esculenta* with gillnets of two mesh sizes. The Results of the experiment are the midlengths of size classes, the number of fish caught with net 1 & 2, and the meshsizes of both nets. The data can be analysed with the function `select`.

Usage

```
data(tilapia)
```

Format

A list consiting of:

- `midLengths` the midlengths of size classes,
- `numNet1` the number of fish caught with net 1,
- `numNet2` the number of fish caught with net 2,
- `msNet1` the meshsize of net 1,
- `msNet2` and the meshsize of net 2.

Source

Garrod, D.J., 1961. The selection characteristics of nylon gill nets for *Tilapia esculenta* Graham. J.Cons.CIEM, 26(2):191-203

Examples

```
data(tilapia)
str(tilapia)
summary(tilapia)
```

`trammelnet` *Trammel net data*

Description

Data of an experiment with several trammel nets with different mesh sizes. Can be used for function `select_Millar`.

Usage

```
data(trammelnet)
```

Format

A list consiting of:

- `$midLengths` the midlengths of size classes,
- `$meshSizes` the meshsizes,
- `$catchPerNet_mat` a matrix with the numbers in catch of the corresponding mesh sizes (same order),

Source

Millar, R. B., Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3), 471-477.

Examples

```
data(trammelnet)
str(trammelnet)
summary(trammelnet)
```

`trawl_fishery_Java` *Data from the trawl fishery off the North coast of Java*

Description

Times series of catch and effort data from the trawl fishery off the North coast of Java. This dataset can be used for the estimation of maximum sustainable yield by means of the production models (`prod_mod` and `prod_mod_ts`).

Usage

```
data(trawl_fishery_Java)
```

Format

A dataframe consisting of: 1. **year** a vector with years, 2. **Y** yield [1000 tons], and 3. **f** fishing effort [number of standard vessels].

Source

Dwiponggo, A., 1979. Review of the demersal resources and fisheries in the Java Sea. IPFC:RRD/II/79/Inf.12. Paper presented at the SCORRAD Meeting, 1979, Hong Kong.

Examples

```
data(trawl_fishery_Java)
str(trawl_fishery_Java)
summary(trawl_fishery_Java)
```

`VBGF` *Von Bertalanffy Growth function (VBGF)*

Description

This function applies the von Bertalanffy growth function (VBGF). It allows to calculate ages from lengths or lengths from ages based on the special, generalised or seasonalised VBGF.

Usage

```
VBGF(param, t = NA, L = NA, na.rm = FALSE)
```

Arguments

`param`	a list with following potential objects: • `Linf`: infinite length for investigated species in cm, or • `Winf`: infinite weight for investigated species in gramm • `K`: growth coefficent for investigated species per year • `t0`: theoretical time zero, at which individuals of this species hatch (default: 0) • `b`: exponent of weight length relationship (default: 3) • `D`: surface factor (default: 1) • `L0`: length at hatching for VBGF with L0 • `ts`: onset of the first oscillation relative to t0 • `C`: intensity of (sinusoid) growth oscillations. Default is no oscillation (C = 0)
`t`	ages for which to calculate corresponding lengths, or
`L`	lengths for which to calculate corresponding ages
`na.rm`	logical; should NA in input length or age vector be omitted? (default: FALSE)

Details

Based upon which input parameters are given one of the following VBGF types is applied: "special", "generalised", or "seasonalised" VBGF.

Value

A vector with estimated lengths corresponding to provided ages.

References

Somers, I. F. (1988). On a seasonally oscillating growth function. Fishbyte, 6(1), 8-11

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
# calculation of lengths
# with t0
t <- seq(0,6,0.1)
Lt <- VBGF(list(Linf=80, K=0.6, t0=-0.1),t=t)
plot(t, Lt, t="l")

# with L0
t <- seq(0,6,0.1)
Lt <- VBGF(list(Linf=80, K=0.6, L0=2),t=t)
plot(t, Lt, t="l")

# with Winf
t <- seq(0,6,0.1)
Wt <- VBGF(list(Winf=4000, K=0.8), t=t)
plot(t, Wt, t="l")

# seasonalised VBGF
t <- seq(0,6,0.1)
Lt <- VBGF(list(Linf=80, K=0.6, t0=-0.1, ts=0.5, C=0.75),t=t)
plot(t, Lt, t="l")


# calculation of ages
L <- seq(2,200,0.1)
t <- VBGF(L = L, list(Linf=210, K=0.8, C= 0.75))
plot(t, L, t="l")
```

`VPA` *Virtual Population Analysis (VPA)*

Description

This function applies the Virtual Population Analysis (VPA) or Cohort analysis (CA). Methods used to estimate stock biomass and fishing mortality per age/length group.

Usage

```
VPA(
  param,
  catch_columns = NA,
  catch_unit = NA,
  catch_corFac = NA,
  terminalF = NA,
  terminalE = NA,
  analysis_type = "VPA",
  algorithm = "new",
  plus_group = TRUE,
  plot = FALSE
)
```

Arguments

`param`	a list consisting of following parameters: • `midLengths` or `age`: midpoints of the length class (length-frequency data) or ages (age composition data), • `Linf`: infinite length for investigated species in cm [cm], • `K`: growth coefficent for investigated species per year [1/year], • `t0`: theoretical time zero, at which individuals of this species hatch, • `M`: natural mortality [1/year] (numeric value or vector of identical length than midLengths), • `a`: length-weight relationship coefficent (W = a * L^b; for kg/cm3), • `b`: length-weight relationship coefficent (W = a * L^b), • `catch`: catch as vector for pseudo cohort analysis, or a matrix with catches of subsequent years to follow a real cohort. For age-based VPA/CA catch has to be provided in numbers, e.g. '000 individuals for length-based VPA/CA catch can also be provided in weight, e.g. kg (use argument `catch_unit`).
`catch_columns`	numerical; indicating the column of the catch matrix which should be used for the analysis.
`catch_unit`	optional; a character indicating if the catch is provided in weight ("tons" or "kg") or in thousand individuals ("'000")
`catch_corFac`	optional; correction factor for catch, in case provided catch does spatially or temporarily not reflect catch for fishing ground of a whole year.
`terminalF`	the fishing mortality rate of the last age/length group.
`terminalE`	the exploitation rate of the last age/length group.
`analysis_type`	determines which type of assessment should be done, options: "VPA" for age or length-based Virtual Population Analysis, "CA" for age- or length-based Cohort Analysis. Default is "VPA".

`algorithm`	an Algorithm to use to solve for fishing mortality. The default setting `"new"` uses `optimise`, while `"old"` uses the algorithm described by Sparre and Venema (1998).
`plus_group`	logical; indicating if the last length group is a plus group (default: TRUE).
`plot`	logical; indicating whether a plot should be printed

Details

The main difference between virtual population analysis (VPA) and cohort analysis (CA) is the step of calculating the fishing mortality per age class or length group. While CA works with an approximation by assuming that all fish are caught during a single day, which makes the calcualtion easier, VPA assumes that the fish are caught continuously, which has to be solved by the trial and error method (Sparre and Venema, 1998). For the age-based VPA/CA the catch has to be provided in numbers (or ’000 numbers), while for the length-based VPA/CA the catch can also be provided in weight (tons or kg) by using the argument `catch_unit`. The catch has to be representative for fished species, that means there should not be other fisheries fishing the same stock. If this is the case `catch_corFac` can be used as a raising factor to account for the proportion of fish caught by other fisheries. When the model should follow a real cohort instead of a pseudo cohort, `catch` has to be provided as matrix. The model then starts to follow the first age class in the first column. If `catch` matrix is shorter than the number of age classes, the age or length classes without catch information are omitted. It is recommended to only follow a real cohort if there is enough information for all age classes (test with: `dim(catch)[1] <= dim(catch)[2]`). If `plus_group` is TRUE a different calculation for the survivors of the last length group is used (for more details please refer to Sparre & Venema (1998)).

Value

A list with the input parameters and following list objects:

- **classes.num**: numeric age classes or length groups (without plus sign),
- **catch.cohort**: a vector with the catch values which were used for the analysis (exists only if catch was a matrix),
- **FM_calc**: a vector with the ifshing mortality (M),
- **Z**: a vector with the total mortality (Z),
- **survivors**: a vector with the number of fish surviving to the next age class or length group (same unit than input catch vector),
- **annualMeanNr**: ta vector with the mean number of fish per year (same unit than input catch vector),
- **meanBodyWeight**: a vector with the mean body weight in kg,
- **meanBiomassTon**: a vector with the mean biomass in tons,
- **YieldTon**: a vector with the yield in tons,
- **natLoss**: a vector with the number of fish died due to natural mortality,
- **plot_mat**: matrix with rearranged survivors, nat losses and catches for plotting;

References

Jones, R., 1984. Assessing the effects of changes in exploitation pattern using length composition data (with notes on VPA and cohort analysis). *FAO Fish.Tech.Pap.*, (256): 118p.

Jones, R., 1990. Length-cohort analysis: the importance of choosing the correct growth parameters. *Journal du Conseil: ICES Journal of Marine Science*, 46(2), 133-139

Pope, J.G., 1972. An investigation of the accuracy of virtual population analysis using cohort analysis. *Res.Bull.ICNAF*, (9):65-74

Pope, J.G., 1979. A modified cohort analysis in which constant natural mortality is replaced by estimates of predation levels. *ICES C.M.* 1979/H:16:7p. (mimeo)

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

References for weight-length relationship parameters (a & b): Dorel, D., 1986. Poissons del'Atlantique nord-est relations taille-poids. Institut Francais de Recherche pour l'Exploitation de la Mer. Nantes, France. 165 p.

Examples

```
#_______________________________________________
# Virtual Popuation Analysis with age-composition data
data(whiting)
output <- VPA(param = whiting, catch_columns = 1, terminalE = 0.5, analysis_type = "VPA")
plot(output)
#_______________________________________________
# Pope's Cohort Analysis with age-composition data
data(whiting)
VPA(whiting, terminalE = 0.5, catch_columns = 3, analysis_type = "CA",
   plot= TRUE, plus_group = TRUE)

#_______________________________________________
# Virtual population analysis with length-composition data
data(hake)
VPA(hake, terminalE = 0.5, analysis_type = "VPA", plot = TRUE,
    catch_unit = "'000", plus_group = TRUE)
#_______________________________________________
# Jones's Cohort Analysis with length-composition data
data(hake)
VPA(hake, terminalE = 0.5, analysis_type = "CA", plot = TRUE,
   catch_unit = "'000", plus_group = TRUE)
```

`whiting` *Whiting data*

Description

A dataset of North Sea whiting *Merlangius merlangus* caught during the period 1974-1980. Can be used for `catchCurve` and `VPA`.

Usage

```
data(whiting)
```

Format

A list consisting of:

- `age`: a vector with age groups,
- `M`: natural mortality rate,
- `a`: length-weight relationship coefficent (W = a * L^b),
- `b`: length-weight relationship coefficent (W = a * L^b),
- `catch`:a matrix with catches from 1974 to 1980.

Source

ICES, 1981. Report of the *Ad hoc* working group on the use of effort data in assessment, Copenhagen, 2-6 March 1981. *ICES C.M.* 1981/G:5 (mimeo)

Examples

```
data(whiting)
str(whiting)
summary(whiting)
```

`yeardec2date` *Year - Date conversion*

Description

Convert numeric years to dates

Usage

```
yeardec2date(yeardec)
```

Arguments

`yeardec`	numeric year

Value

date in format "%Y-%m-%d" (class 'Date').

Examples

```
 yeardec2date(2014.14)
```

`ypr` *Yield per recruit*

Description

Estimates the absolute and relative yield and biomass per recruit and the first order derivative.

Usage

```
ypr(param, FM_change, t = NA)
```

Arguments

`param`	a list consisting of following parameters (not all are required): • **Linf**: infinite length in cm, • **Winf**: infinite weight, • **K**: growth coefficent for investigated species per year [1/year], • **t0**: theoretical time zero, at which individuals of this species hatch, • **M**: natural mortality, • **tr**: age of recruitment, • **tc**: age of first capture;
`FM_change`	vector with ascending fishing mortalities,
`t`	default NA

Details

The Thompson and Bell model incorporates an iteration step simulating the stock by means of the `stock_sim` function. In case changes in gear characteristics - here measured in terms of Lc or tc, the length or age at first capture, respectively - should be explored, a list with selectivity information about the gear has to be provided and the prediction models make use of the selectivity `select_ogive` function. Sparre and Venema (1998) recommend to treat the last length class always as plus group. This model is very sensitive to zero observations in the ultimate length classes. If unrealistic results are returned, it is recommended to cut length classes with zero observations, group them in a plus group or to change the interval between length classes. Equations which are used in this function assume isometric growth, an assumption often not met. Further, the assumption that there is no relationship between the parental stock size and progeny over a wide range of fishing mortalities or exploitation values, respectively, is also said to be untrue. By default, the functions assume knife-edge recruitment and selection of gears (Sparre and Venema, 1998).

Value

A list with the input parameters and dependent on the model type following list objects:

- `type = 'ypr'`

References

Berkeley, S.A., and Houde, E.D., 1980. Swordfish, Xiphias gladius, dynamics in the Straits of Florida. *ICES C.M.*, 11.

Beverton, R.J.H., and Holt, S.J., 1964. Table of yield functions for fishery management. *FAO Fish. Tech. Pap.* 38, 49 p.

Beverton, R.J.H., and Holt, S.J., 1966. Manual of methods for fish stock assessment. Pt. 2: Tables of yield functions. *FAO Fisheries Technical Paper*, (38)Rev.1:67 p.

Boerema, L.K., and J.A. Gulland, 1973. Stock assessment of the Peruvian anchovy (Engraulis ringens) and management of the fishery. *Journal of the Fisheries Board of Canada*, 30(12):2226-2235

Garcia, S. and N.P. van Zalinge, 1982. Shrimp fishing in Kuwait: methodology for a joint analysis of the artisanal and industrial fisheries. pp. 119-142 In: Report on the Workshop on assessment of the shrimp stocks of the west coast of the Gulf between Iran and the Arabian Peninsula. Fisheries development in the Gulf. Rome, FAO, FI:DP/RAB/80/015/1, 163 p.

Gulland, J.A., 1983. Fish stock assessment: a manual of basic methods. *FAO/Wiley*, New York.

Gulland, J.A. and Boerema, L., 1973. Scientific advice on catch levels. *Fish. Bull. (US)* 71:325-335.

Jones, R.E. 1957. A much simplified version of the fish yield equation. Doc. No. P. 21. Paper presented at the Lisbon joint meeting of International Commission Northwest Atlantic-Fisheries, International Council for the Exploration of the Sea, and Food and Agriculture Organization of the United Nations. 8 p. [Mimeo].

Millar, R.B., and Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3):471-477

Pauly, D., 1980. A selection of simple methods for the assessment of tropical fish stocks. *FAO Fisheries Circulars (FAO).* no. 729.

Pauly, D., 1984. Fish population dynamics in tropical waters: a manual for use with programmable calculators. *ICLARM* Stud. Rev. 8, 325 p.

Pauly, D. and M. Soriano. 1986. Some practical extensions to Beverton and Holt's relative yield-per-recruit model, p. 491-495. In J.L. Maclean, L.B. Dizon and L.V. Hosillos (eds.) The First Asian Fisheries Forum. Asian Fisheries Society, Manila.

Schaefer, M.B., 1954. Some aspects of the dynamics of populations important to the management of the commercial marine fisheries. *Inter-Am. Trop. Tuna Comm., Bull.* 1(2):27-56.

Schaefer, M.B., 1957. A study of the dynamics of the fishery for yellowfin tuna in the eastern tropical Pacific Ocean. *Inter-Am. Trop. Tuna Comm., Bull.* 2:247-268.

Sparre, P., and Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

`ypr_sel` *Yield per recruit with selection ogive*

Description

Estimates relative yield and biomass, and the first order derivative.

Usage

```
ypr_sel(param, FM_change, Lt, P)
```

Arguments

`param`	a list consisting of following parameters (not all are required): • **Linf**: infinite length in cm • **Winf**: infinite weight • **K**: growth coefficent for investigated species per year [1/year], • **t0**: theoretical time zero, at which individuals of this species hatch • **M**: natural mortality • **tr**: age of recruitment • **tc**: age of first capture
`FM_change`	vector with ascending fishing mortalities
`Lt`	length at time
`P`	population size

Details

The Thompson and Bell model incorporates an iteration step simulating the stock by means of the `stock_sim` function. In case changes in gear characteristics - here measured in terms of Lc or tc, the length or age at first capture, respectively - should be explored, a list with selectivity information about the gear has to be provided and the prediction models make use of the selectivity `select_ogive` function. Sparre and Venema (1998) recommend to treat the last length class always as plus group. This model is very sensitive to zero observations in the ultimate length classes. If unrealistic results are returned, it is recommended to cut length classes with zero observations, group them in a plus group or to change the interval between length classes. Equations which are used in this function assume isometric growth, an assumption often not met. Further, the assumption that there is no relationship between the parental stock size and progeny over a wide range of fishing mortalities or exploitation values, respectively, is also said to be untrue. By default, the functions assume knife-edge recruitment and selection of gears (Sparre and Venema, 1998).

Value

A list with the input parameters and dependent on the model type.

References

Berkeley, S.A., and Houde, E.D., 1980. Swordfish, Xiphias gladius, dynamics in the Straits of Florida. *ICES C.M.*, 11.

Beverton, R.J.H., and Holt, S.J., 1964. Table of yield functions for fishery management. *FAO Fish. Tech. Pap.* 38, 49 p.

Beverton, R.J.H., and Holt, S.J., 1966. Manual of methods for fish stock assessment. Pt. 2: Tables of yield functions. *FAO Fisheries Technical Paper*, (38)Rev.1:67 p.

Boerema, L.K., and J.A. Gulland, 1973. Stock assessment of the Peruvian anchovy (Engraulis ringens) and management of the fishery. *Journal of the Fisheries Board of Canada*, 30(12):2226-2235

Garcia, S. and N.P. van Zalinge, 1982. Shrimp fishing in Kuwait: methodology for a joint analysis of the artisanal and industrial fisheries. pp. 119-142 In: Report on the Workshop on assessment of the shrimp stocks of the west coast of the Gulf between Iran and the Arabian Peninsula. Fisheries development in the Gulf. Rome, FAO, FI:DP/RAB/80/015/1, 163 p.

Gulland, J.A., 1983. Fish stock assessment: a manual of basic methods. *FAO/Wiley*, New York.

Gulland, J.A. and Boerema, L., 1973. Scientific advice on catch levels. *Fish. Bull. (US)* 71:325-335.

Jones, R.E. 1957. A much simplified version of the fish yield equation. Doc. No. P. 21. Paper presented at the Lisbon joint meeting of International Commission Northwest Atlantic-Fisheries, International Council for the Exploration of the Sea, and Food and Agriculture Organization of the United Nations. 8 p. [Mimeo].

Millar, R.B., and Holst, R., 1997. Estimation of gillnet and hook selectivity using log-linear models. *ICES Journal of Marine Science: Journal du Conseil*, 54(3):471-477

Pauly, D., 1980. A selection of simple methods for the assessment of tropical fish stocks. *FAO Fisheries Circulars (FAO)*. no. 729.

Pauly, D., 1984. Fish population dynamics in tropical waters: a manual for use with programmable calculators. *ICLARM* Stud. Rev. 8, 325 p.

Pauly, D. and M. Soriano. 1986. Some practical extensions to Beverton and Holt's relative yield-per-recruit model, p. 491-495. In J.L. Maclean, L.B. Dizon and L.V. Hosillos (eds.) The First Asian Fisheries Forum. Asian Fisheries Society, Manila.

Schaefer, M.B., 1954. Some aspects of the dynamics of populations important to the management of the commercial marine fisheries. *Inter-Am. Trop. Tuna Comm., Bull.* 1(2):27-56.

Schaefer, M.B., 1957. A study of the dynamics of the fishery for yellowfin tuna in the eastern tropical Pacific Ocean. *Inter-Am. Trop. Tuna Comm., Bull.* 2:247-268.

Sparre, P., and Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

`Z_BevertonHolt` *Beverton & Holt's Z-Equations*

Description

A method to estimate the instantaneous total mortality rate (Z) based on a method derived by Beverton and Holt (1956).

Usage

```
Z_BevertonHolt(param, catch_columns = NA, Lprime_tprime)
```

Arguments

`param`	a list consisting of following parameters: • `midLengths` or `age`: midpoints of length groups (length-frequency data) or ages (age composition data), • `Linf`: infinite length for investigated species in cm [cm], • `K`: growth coefficent for investigated species per year [1/year], • `t0`: theoretical time zero, at which individuals of this species hatch, • `catch`: catch as vector, or a matrix with catches of subsequent years;
`catch_columns`	optional; in case catch is a matrix or data.frame, a number or vector indicating which column(s) of the matrix should be analysed (Default: `NA`).
`Lprime_tprime`	length or age prime, above which all fish are under full exploitation as mid length or age class.

Details

The first length group or age class within the list object `midLengths` or `age` will be used as the Lprim or tprime (length of recruitment to fishery).

Value

A list with the input parameters and following objects:

- **tmean** or **Lmean**: mean age or length of fish,
- **tprime** or **Lprime**: some age or length for which all fish of that length and longer are under full exploitation,
- **Z**: total mortality.

References

Beverton R.J.H and S.J. Holt, 1956. A review of methods of estimating mortality rates in exploited fish populations, with special reference to sources of bias in catch sampling. *Rapp.P.-v.Reun.CIEM*, 140:67-83

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Examples

```
# based on length-frequency data
data(synLFQ2)
Z_BevertonHolt(synLFQ2, catch_columns = 2, Lprime_tprime = 47.5)

# based on age composition data
data(synCAA1)
Z_BevertonHolt(synCAA1, catch_columns = 3, Lprime_tprime = 2.5)
```

`Z_CPUE` *Estimate Z from CPUE data*

Description

Method to estimate the instantaneous total mortality rate (Z) from catch per unit of effort (CPUE) data according to standard, Heincke's, or Robson & Chapman's method.

Usage

```
Z_CPUE(param, method = "standard", omit_age1 = FALSE)
```

Arguments

`param`	a list consisting of following parameters: • `cohort`: a vector with with a cohort label, • `age`: a vector with ages, • `CPUE`: a vector with CPUE values;
`method`	a character string indicating which assessment method should be used: `"standard"`, `"Heincke"`, or `"RobsonChapman"`.
`omit_age1`	logical; if `TRUE` the first age group is omitted (Default `FALSE`).

Details

In Heincke's and RobsonChapman's method age groups older than 4 are lumped, because age groups older than 3 or 4 are said to be hard to seperate (Ricker, 1975). Sparre and Venema (1998) recommend to omit the first age group in case it is not fully exploited by the fishery.

Value

A list with input parameters and a Z value or matrix depending on the method.

References

Sparre, P., Venema, S.C., 1998. Introduction to tropical fish stock assessment. Part 1. Manual. *FAO Fisheries Technical Paper*, (306.1, Rev. 2). 407 p.

Sparre, P., Venema, S.C., 1999. Introduction to tropical fish stock assessment. Part 2. Excercises. *FAO Fisheries Technical Paper*, (306.2, Rev. 2). 94 p.

Ricker, W.E., 1975. Computation and interpretation of biological statistics of fish populations. *Bull.Fish.Res.Board Can.*, (191):382 p.

Examples

```
# load data
data(synCPUE)

# run model with standard method
Z_CPUE(synCPUE, method = "standard")

# run model with Heincke's method
Z_CPUE(synCPUE, method = "Heincke")

# run model with Robson and Chapman's method
Z_CPUE(synCPUE, method = "RobsonChapman", omit_age1 = TRUE)
```

Index