Package ‘rFAMS’

February 10, 2026

Version 0.0.2

Date 2026-1-13

Title Fisheries Analysis and Modeling Simulator

Description
Simulates the dynamics of exploited fish populations using the Jones modification of the Beverton-Holt
equilibrium yield equation to compute yield-per-recruit and dynamic pool models (Ricker 1975) <https://publications.gc.ca/site/eng/480738/publication.html>.
Allows users to evaluate minimum, slot, and inverted length limits on exploited fisheries using specified life history parameters. Users can simulate population under a variety of conditional fishing mortality
and conditional natural mortality. Calculated quantities include number of fish harvested and dying naturally,
mean weight and length of fish harvested, number of fish that reach specified lengths of interest, total number
of fish and biomass in the population, and stock density indices.

URL https://github.com/fishR-Core-Team/rFAMS/

BugReports https://github.com/fishR-Core-Team/rFAMS/issues

License GPL (>= 2)

Depends R (>= 4.1.0)

Imports stats, purrr, FSA

Suggests dplyr, ggplot2, metR, knitr, rmarkdown, testthat (>= 3.0.0), zipfR, quarto, FSAdata, tidyr

Encoding UTF-8

RoxygenNote 7.3.3

Config/testthat/edition 3

Config/Needs/website quarto

NeedsCompilation no

Author Jason C. Doll [aut, cre],
Derek H. Ogle [aut]

Maintainer Jason C. Doll <`jcdoll20@hotmail.com`>

Repository CRAN

Date/Publication 2026-02-10 21:30:02 UTC

Contents

`dpmBH` *Main function to simulate expected yield using the Dynamic Pool model for a range of input parameters, including minimum length limits for harvest*

Description

Estimate yield using the Beverton-Holt Yield-per-Recruit (YPR) model using ranges of values for conditional fishing mortality (`cf`), conditional natural mortality (`cm`), and minimum length limits for harvest (`minLL`).

Usage

```
dpmBH(
  simyears,
  minLL,
  cf,
  cm,
  rec,
  lhparms,
  matchRicker = FALSE,
  species = NULL,
  group = NULL
)
```

Arguments

`simyears`	A single numeric for the lower limit of minimum length limit for harvest in mm.
`minLL`	A single numeric representing the minimum length limit for harvest in mm.
`cf`	A matrix of conditional fishing mortality where each row represents a year and each column represents age. Ages are age-0 through maximum age.
`cm`	A matrix of conditional natural mortality where each row represents a year and each column represents age. Ages are age-0 through maximum age.
`rec`	A single numeric representing initial recruitment abundance.
`lhparms`	A named vector or list that contains values for each `N0`, `tmax`, `Linf`, `K`, `t0`, `LWalpha`, and `LWbeta`. See `makeLH` for definitions of these life history parameters. Also see details.
`matchRicker`	A logical that indicates whether the yield function should match that in Ricker (). Defaults to `TRUE`. The only reason to changed to `FALSE` is to try to match output from FAMS. See the "YPR_FAMSvRICKER" article.
`species`	is a single character to specify the species used in the simulation and will define the length for `stock`, `quality`, `preferred`, `memorable`, and `trophy`. Length categories are obtained from the FSA package, see the `PSDlit` documentation.
`group`	is a single character to specify the sub-group name of a species used in the simulation and will define the length for `stock`, `quality`, `preferred`, `memorable`, and `trophy`. Length categories are obtained from the FSA package, see the `PSDlit` documentation.

Details

Details

Value

A list with two data.frame object. The first list item contains a data.frame with the following calculated values in a summary by age:

- `year` is the year number for the simulation
- `yc`is the year class number for the simulation
- `age` is the age of fish from the year class
- `length` is the length at age at the beginning of the year based on parameters supplied for the von Bertlanffy growth model.
- `weight` is the total weight at the beginning of the year for length at age based on the parameters supplied for the weight-length model.
- `N_start` is the number of fish alive at the start of the year for the given age and year class.
- `explitation` is the exploitation rate at age based on the supplied conditional fishing mortality rate.
- `expect_nat_death` is the expectation of natural death based on the supplied conditional natural mortality rate.
- `cf` is the supplied conditional fishing mortality rate.

- `cm` is the supplied conditional natural mortality rate.
- `F` is the instantaneous rate of fishing mortality.
- `M` is the instantaneous rate of natural mortality.
- `Z` is the instantaneous rate of total mortality.
- `S` is the (total) annual rate of survival.
- `biomass` is the total biomass of fish at age and year
- `N_harvest` is the total number of fish harvested at age and year
- `N_die` is the total number of fish that die at age and year
- `yield` is the estimated yield (in g).
- `minLL` is the minimum length limit specified in the simulation
- `N0` is the number of initial

For convenience the data.frame also contains the model input values (`minLL`, `N0`, `N0`, `Linf`, `K`, `t0`, `LWalpha`, `LWbeta`, and `tmax`).

The data.frame also contains a `notes` value which may contain abbreviations for "issues" that occurred when computing the results and were adjusted for. The possible abbreviates are as follows:

- `minLL>=Linf`: The minimum length limit (minLL) being explored was greater than the given asymptotic mean length (Linf). For the purpose (only) of computing the time at recruitment to the fishery (tr) the Linf was set to minLL+0.1.
- `tr<t0`: The age at recruitment to the fishery (tr) was less than the hypothetical time when the mean length is zero (t0). The fish can't recruit to the fishery prior to having length 0 so tr was set to t0. This also assures that the time it takes to recruit to the fishery is greater than 0.
- `Y=Infinite`: The calculated yield (Y) was inifinity, which is impossible and suggests some other propblem. Yield was set to NA.
- `Y<0`: The calculated yield (Y) was negative, which is impossible. Yield was set to 0.
- `Nharv<0`: The calculated number of fish harvested (Nharv) was negative, which is not possible. Number harvested was set to 0.
- `Ndie<0`: The calculated number of fish recruiting to the fishery that died naturally (Ndie) was negative, which is not possible. Number that died was set to 0.
- `agvglen<minLL`: The average length of harvested fish was less than the given minimum length limit being explored, which is not possible (with only legal harvest). The average length was set to the minimum length limit.

The second list item contains a data.frame with the following calculated values in a summary by year:

- `year` is the year number for the simulation
- `substock` is the number of substock sized fish at age and year at the beginning of the year.
- `stock` is the number of stock sized fish at age and year at the beginning of the year.
- `quality` is the number of quality sized fish at age and year at the beginning of the year.
- `preferred` is the number of preferred sized fish at age and year at the beginning of the year.
- `memorable` is the number of memorable sized fish at age and year at the beginning of the year.

- `trophy` is the number of trophy sized fish at age and year at the beginning of the year.
- `PSD` is the number of quality sized fish divided by the number of stock sized multiplied by 100.
- `PSD_P` is the number of preferred sized fish divided by the number of stock sized multiplied by 100.
- `PSD_M` is the number of memorable sized fish divided by the number of stock sized multiplied by 100.
- `PSD_T` is the number of trophy sized fish divided by the number of stock sized multiplied by 100.
- `Age_1plus` is the total number of fish age-1 plus per year.
- `Yield_Age_1plus` is the total year of age-1 plus fish per year.
- `Total_biomass` is the total biomass of age-1 plus fish per year.
- `N_harvest_Age_1plus` is the number of age-1 plus fish that are harvested per year.
- `N_die_Age_1plus` is the nubmer of age-1 plus fish that die per year.

PSD-X are calculated based on the number of fish in each category (`stock`, `quality`, `preferred`, `memorable`, and `trophy`) at the beginning of the year. That is, the length at age during the start of the year is used to assign PSD-X categories at age. For example, if Quality size is 300mm, an age-1 fish at 275mm at the start of the year would not be counted as a quality sized fish, but an age-2 fish at 325mm at the start of the year would be counted as a quality sized fish.

Author(s)

Jason C. Doll, `<jason.doll@fmarion.edu>`

See Also

`yprBH_func` for estimating yield from single values of `cf`, `cm`, and `minLL`, and `yprBH_minLL_fixed` for simulating yield with multiple values of `cf` and `cm` but a fixed value for `minLL`.

Examples

```
#load required library
library(dplyr)
library(ggplot2)

# Setting a custom theme for plots (to make look nice)
# Optional for plotting
theme_FAMS <- function(...) {
  theme_bw() +
    theme(
      panel.grid.major=element_blank(),panel.grid.minor=element_blank(),
      axis.text=element_text(size=14,color="black"),
      axis.title=element_text(size=16,color="black"),
      axis.title.y=element_text(angle=90),
      axis.line=element_line(color="black"),
      panel.border=element_blank()
    )
}
```

```
# Example of simulating yield with the dynamic pool model,

lhparms <- makeLH(N0=100,tmax=30,Linf=1349.5,K=0.111,t0=0.065,
            LWalpha=-5.2147,LWbeta=3.153)
simyears <- 50
minLL <- 400
rec <- genRecruits(method = "fixed", Nrec = 100, simyears = simyears)
cm <- matrix(rep(c(rep(0,1), rep(0.18,(lhparms$tmax))), simyears),nrow=simyears,byrow=TRUE)
cf <- matrix(rep(c(rep(0,1), rep(0.33,(lhparms$tmax))), simyears),nrow=simyears,byrow=TRUE)

out<-dpmBH(simyears = simyears, minLL = minLL, cf = cf, cm = cm, rec = rec, lhparms = lhparms,
           matchRicker=FALSE,species="Striped Bass",group="landlocked")

#Use summary by year data frame to plot PSD vs year
ggplot(data=out[[2]],mapping=aes(x=year,y=PSD)) +
  geom_point() +
  geom_line() +
  labs(y="PSD",x="Year") +
  theme_FAMS()

#Use summary by year data frame to plot yield vs year
ggplot(data=out[[2]],mapping=aes(x=year,y=Yield_age_1plus)) +
  geom_point() +
  geom_line() +
  labs(y="Total yield (g)",x="Year") +
  theme_FAMS()

#Plot date using summary by age
#filter for year class = 1
plotdat<- out[[1]] |> filter(yc==1)
#Plot yield vs age
ggplot(data=plotdat,mapping=aes(x=age,y=yield)) +
  geom_point() +
  geom_line() +
  labs(y="Total yield (g)",x="Age") +
  theme_FAMS()

#Plot Number harvested vs age
ggplot(data=plotdat,mapping=aes(x=age,y=N_harvest)) +
  geom_point() +
  geom_line() +
  labs(y="Number harvested",x="Age") +
  theme_FAMS()


#Recruitment based on a normal distribution
rec <- genRecruits(method = "normal", simyears = simyears,
                   meanR = 1000, sdR = 500, MinR = 100, MaxR =2500)
cm <- matrix(rep(c(rep(0,1), rep(0.18,(lhparms$tmax))), simyears),nrow=simyears,byrow=TRUE)
cf <- matrix(rep(c(rep(0,1), rep(0.33,(lhparms$tmax))), simyears),nrow=simyears,byrow=TRUE)

out_2<-dpmBH(simyears = simyears, minLL = minLL, cf = cf, cm = cm, rec = rec, lhparms = lhparms,
```

```
             matchRicker=FALSE,species="Striped Bass",group="landlocked")

#Use summary by year data frame to plot PSD vs year
ggplot(data=out_2[[2]],mapping=aes(x=year,y=PSD)) +
  geom_point() +
  geom_line() +
  labs(y="PSD",x="Year") +
  theme_FAMS()

#Use summary by year data frame to plot yield vs year
ggplot(data=out_2[[2]],mapping=aes(x=year,y=Yield_age_1plus)) +
  geom_point() +
  geom_line() +
  labs(y="Total yield (g)",x="Year") +
  theme_FAMS()

#Plot date using summary by age
#Plot yield vs age for each year class
ggplot(data=out_2[[1]],mapping=aes(x=age,y=yield,group=yc,color=yc)) +
  geom_point() +
  geom_line() +
  labs(y="Total yield (g)",x="Age") +
  theme_FAMS()

#Plot Number harvested vs age
ggplot(data=out_2[[1]],mapping=aes(x=age,y=N_harvest,group=yc,color=yc)) +
  geom_point() +
  geom_line() +
  labs(y="Number harvested",x="Age") +
  theme_FAMS()
```

`dpmBH_func` *Simulate yield using the dynamic pool model.*

Description

Estimate yield-at-age using the Beverton-Holt Yield-per-Recruit (YPR) model for a single year-class. This main function accepts a minimum length limit for harvest (`minLL`), a vector for conditional fishing mortality (`cf`), a vector of conditional natural mortality (`cm`), a vector of recruitment abundance (`rec`), and life history parameters (`lhparams`).

Usage

```
dpmBH_func(minLL, cf, cm, rec, lhparms, matchRicker = FALSE)
```

Arguments

`minLL`	A single numeric representing the minimum length limit for harvest in mm.
`cf`	A matrix of conditional fishing mortality where each row represents a year and each column represents age. Ages are age-0 through maximum age.
`cm`	A matrix of conditional natural mortality where each row represents a year and each column represents age. Ages are age-0 through maximum age.
`rec`	A single numeric representing initial recruitment abundance.
`lhparms`	A named vector or list that contains values for each `N0`, `tmax`, `Linf`, `K`, `t0`, `LWalpha`, and `LWbeta`. See `makeLH` for definitions of these life history parameters. Also see details.
`matchRicker`	A logical that indicates whether the yield function should match that in Ricker (). Defaults to `TRUE`. The only reason to changed to `FALSE` is to try to match output from FAMS. See the "YPR_FAMSvRICKER" article.

Details

Details will be filled out later

Value

A data.frame with the following calculated values:

- `age` is the age of the year class
- `length` is the mean length at age calculated using the von Bertalanffy growth model and provided parameters
- `weight` is the mean weight at age calculated using the log10 length-weight regression using the provided parameters
- `N_start` is the number of individuals at age at the start of the year.
- `exploitation` is the exploitation rate.
- `expect_nat_death` is the expectation of natural death.
- `cf` is the conditional fishing mortality at age.
- `cm` is the conditional natural mortality at age
- `F` is the instantaneous rate of fishing mortality.
- `M` is the instantaneous rate of natural mortality.
- `Z` is the instantaneous rate of total mortality.
- `S` is the (total) annual rate of survival
- `tr` is the time for a fish to recruit to a minimum length limit (i.e., time to enter fishery).
- `Nt` is the number of fish at time tr (time they become harvestable size).
- `biomass` is the total biomass at age (g)
- `N_harvest` is the total number harvested at age
- `N_die` is the total number that die at age
- `yield` is the estimated yield (in g).

For convenience the data.frame also contains the model input values (`minLL`, `N0`, `N0`, `Linf`, `K`, `t0`, `LWalpha`, `LWbeta`, and `tmax`).

The data.frame also contains a `notes` value which may contain abbreviations for "issues" that occurred when computing the results and were adjusted for. The possible abbreviates are as follows:

- `minLL>=Linf`: The minimum length limit (minLL) being explored was greater than the given asymptotic mean length (Linf). For the purpose (only) of computing the time at recruitment to the fishery (tr) the Linf was set to minLL+0.1.
- `tr<t0`: The age at recruitment to the fishery (tr) was less than the hypothetical time when the mean length is zero (t0). The fish can't recruit to the fishery prior to having length 0 so tr was set to t0. This also assures that the time it takes to recruit to the fishery is greater than 0.
- `Y=Infinite`: The calculated yield (Y) was inifinity, which is impossible and suggests some other propblem. Yield was set to NA.
- `Y<0`: The calculated yield (Y) was negative, which is impossible. Yield was set to 0.
- `Nharv<0`: The calculated number of fish harvested (Nharv) was negative, which is not possible. Number harvested was set to 0.
- `Ndie<0`: The calculated number of fish recruiting to the fishery that died naturally (Ndie) was negative, which is not possible. Number that died was set to 0.
- `agvglen<minLL`: The average length of harvested fish was less than the given minimum length limit being explored, which is not possible (with only legal harvest). The average length was set to the minimum length limit.

Author(s)

Jason C. Doll, `<jason.doll@fmarion.edu>`

References

Ricker, W.E. 1975. Computation and interpretation of biological statistics of fish populations. Technical Report Bulletin 191, Bulletin of the Fisheries Research Board of Canada. Was (is?) from `https://waves-vagues.dfo-mpo.gc.ca/library-bibliotheque/1485.pdf`.

Slipke, J.W., and M.J. Maceina. 2014. Fishery analysis and modeling simulator. v1.64. American Fisheries Society, Bethesda, MD.

Examples

```
lhparms <- makeLH(N0=100,tmax=30,Linf=1349.5,K=0.111,t0=0.065,LWalpha=-5.2147,LWbeta=3.153)

# simulate yield from a single year-class
cm <- rep(0.18,(lhparms$tmax+1))
cf <- c(rep(0,3), rep(0.33,(lhparms$tmax+1) - 3))

Res_1 <- dpmBH_func(minLL=400,cm=cm,cf=cf,rec=1000,lhparms=lhparms,matchRicker=FALSE)

Res_1
```

`est_natmort` *Compute meta-analytic estimates of instantaneous and conditional natural mortality*

Description

Several methods may be used to estimate instantaneous (M) and conditional natural mortality (cm) from other types of data, especially those saved in the life history parameters vector/list from `makeLH`.

Usage

```
est_natmort(lhparms = NULL, method = "rFAMS", incl.avg = FALSE, ...)
```

Arguments

`lhparms`	A named vector or string returned by `lhparms`.
`method`	A string that indicates what methods to use to estimate M (see `metaM`).
`incl.avg`	A logical that indicates whether the average cm should be computed from the estimated M of all methods.
`...`	Option arguments for parameter values required by methods using parameters other than those in `lhparms`. See examples.

Details

The default `methods` to use are all of those listed in `Mmethods` that use some of the life history parameters required by `makeLH`. These methods are not all equally useful or robust, so the user may want to select a subset of them for use after learning more about them. See references in `metaM`.

Other methods that require parameters other than those required by `makeLH` can be used by providing the name of the method in `method` and the required parameters as arguments, as defined in `metaM`. See `metaM` for more details and the examples below for an example.

Value

A data.frame with the following items:

- `method`: The name for the method within the function (as given in `method`).
- `M`: The estimated instantaneous natural mortality rate (from `metaM`)
- `cm`: The estimated conditional natural mortality rate (computed directly from `M`).
- `givens`: A string that contains the input values required by the method to estimate M.

Author(s)

Derek Ogle

Examples

```
# An example lhparm as would be returned from makeLH
tmp <- list(N0=100,tmax=15,Linf=500,K=0.3,t0=-0.5,LWalpha=-5.16,LWbeta=3.1)

# All methods in metaM() that use those life history parameters
est_natmort(tmp)

# Same but including the average in the last row
est_natmort(tmp,incl.avg=TRUE)

# Selecting just one method
est_natmort(tmp,method="HoenigNLS")

# Selecting several methods
est_natmort(tmp,method=c("HoenigNLS","HoenigO","HoenigO2","HoenigLM"))

# A method that uses a parameter not usually in lhparms
est_natmort(tmp,method="QuinnDeriso",PS=0.05)

# Selecting all Hoenig methods using Mmethods from FSA
est_natmort(tmp,method=FSA::Mmethods("Hoenig"))

# Over-riding the Linf param in parameters list, but others from tmp
est_natmort(tmp,method="PaulyLNoT")             # Linf from tmp
est_natmort(tmp,Linf=1000/10,method="PaulyLNoT") # Linf from Linf= arg
```

`genRecruits` *Generate a vector of recruitment abundance for the dynamic pool model.*

Description

These function is used to generate recruitment abundances across multiple years using different random function.

Usage

```
genRecruits(
  method = c("fixed", "uniform", "normal", "StrYC_Nth", "StrYC_randInt"),
  simyears = 50,
  Nrec = NULL,
  MinR = NULL,
  MaxR = NULL,
  meanR = NULL,
  sdR = NULL,
  Nthyr = NULL,
  sizeStr = NULL,
  avgFreq = NULL
)
```

Arguments

`method`	A single string to call the method of generating a vector of recruits. `fixed` generate recruitment based on a fixed value for each year of simyears, `uniform` generates recruitment based on random values from a unifrom distribution for each year of simyears, `normal` generates recruitment based on random values from a unifrom distribution for each year of simyears, `StrYC_Nth` generates recruitment based on a strong year class every Nth year, and `StrYC_randInt` generates recruitment based on a strong year classes at random intervals.
`simyears`	A single numeric that sets the number of years to simulate recruitment
`Nrec`	A single numeric that sets the fixed number of recruitment
`MinR`	A single numeric that sets the minimum recruitment abundance during simulations.
`MaxR`	A single numeric that sets the maximum recruitment abundance during simulations.
`meanR`	A single numeric that sets the mean recruitment abundance.
`sdR`	A single numeric that sets the standard deviation of recruitment abundance
`Nthyr`	A single numeric that sets the Nth year that a strong year class will occur
`sizeStr`	A single numeric that sets the multiplier for the strong year class relative to meanR
`avgFreq`	A single numeric that sets the average frequency of a strong year class.

Details

This function is used internally and not generally used interactively

Value

A vector that contains the given recruitment options that can be used directly in the dynamic pool model (e.g., `dpmBH`).

Author(s)

Jason C. Doll, `<jason.doll@fmarion.edu>`

Examples

```
# To be filled out later
```

`makeLH` *Make a list or vector of life history parameters for yield-per-recruit analyses.*

Description

Efficiently construct either a vector or list that contains the seven life history parameters required for Beverton-Holt yield-per-recruit analyses. The parameters can be given by the user through function arguments. Alternativvely, the von Bertalanffy parameters (`Linf`, `K`, and `t0`) may be extracted from an `nls` object created from fitting the von Bertalanffy equation to length-at-age data (object created outside this function). Similarly the log10-transformed weight-length model coefficients may be extracted from an `lm` object created from fitting the model to transformed weight-length data (object created outside this function). All parameter values are checked for sanity (e.g., Linf>0).

Usage

```
makeLH(N0, tmax, Linf, K, t0, LWalpha, LWbeta, restype = c("list", "vector"))
```

Arguments

`N0`	A single numeric that represents the number of fish in the population at the hypothetical age of `t0`.
`tmax`	A single whole number that represents maximum age in the population in years.
`Linf`	A single numeric that represents the point estimate of asymptotic mean length from the von Bertalanffy growth model OR an `nls` object created from fitting the von Bertalanffy equation to length-at-age data.
`K`	A single numeric that represents the point estimate of the Brody growth coefficient from the von Bertalanffy growth model.
`t0`	A single numeric that represents the point estimate of the x-intercept (i.e., theoretical age at a mean length of 0) from the von Bertalanffy growth model.
`LWalpha`	A single numeric that represents the point estimate of alpha from the length-weight regression on the log10 scale OR an `lm` object created from fitting the model to log10-transformed weight-length data.
`LWbeta`	A single numeric that represents the point estimate of beta from the length-weight regression on the log10 scale.
`restype`	A character that indicates the type of output (list or vector) returned by the function.

Details

Use of this function for putting life history parameters into a list or vector is recommended as (i) values for `Linf`, `K`, `t0`, `LWalpha`, and `LWbeta` can be extracted from objects from appropriate model fitting and (ii) checks for impossible or improbable values for each parameter are performed; i.e.,

Value

A named list or vector (depending on `restype`) that contains the given (or extracted) life history parameters values that can be used directly in the yield-per-recruit calculation functions (e.g., `yprBH_SlotLL`).

Author(s)

Derek Ogle

```
# Best practice for entering life history parameter values
LH <- makeLH(N0=100,tmax=15,Linf=600,K=0.30,t0=-0.6,
             LWalpha=-5.453,LWbeta=3.10)

# Works but no checks on the values
LH <- list(N0=100,tmax=15,Linf=600,K=0.30,t0=-0.6,
           LWalpha=-5.453,LWbeta=3.10)
```

If a list is returned then values will be displayed with the number of decimals provided by the user. If a vector is returned then the number of decimals displayed will be the same for each value and will match the value supplied by the user with the most decimals. Thus, a list is preferred as it will be easier to match what was given to what was expected to be given.

Examples

```
library(FSA)
library(FSAdata)

# ----- Simple examples with explicity arguments for each -------------------
makeLH(N0=100,tmax=15,Linf=500,K=0.3,t0=-0.5,LWalpha=-5.613,LWbeta=3.1)
makeLH(N0=100,tmax=15,Linf=500,K=0.3,t0=-0.5,LWalpha=-5.613,LWbeta=3.1,
       restype="vector")

# ----- Example of extracting values from model fits ------------------------
# N0 and tmax provided as arguments ... Linf, K, and t0 extracted from nls
#   output and LWalpha and LWbeta extracted from lm output. Note that nls
#   and lm output here are just examples of the function, they should be
#   calculated for the same species from the same waterbody, etc.

## get some LVB results (as an example)
data(SpotVA1,package="FSA")
SpotVA1 <- SpotVA1 |>
  dplyr::mutate(tl=tl*25.4)
vb1 <- FSA::vbFuns()
fit1 <- nls(tl~vb1(age,Linf,K,t0),data=SpotVA1,
            start=FSA::vbStarts(tl~age,data=SpotVA1))

## get some LW results (as an example)
data(BluegillLM,package="FSAdata")
BluegillLM <- BluegillLM |>
  dplyr::mutate(logW=log10(wght),
```

```
                logL=log10(tl))
fit2 <- lm(logW~logL,data=BluegillLM)

makeLH(N0=100,tmax=15,Linf=fit1,LWalpha=fit2)
```

`yprBH_func` *Simulate expected yield using the Beverton-Holt Yield-per-Recruit model for single input parameters*

Description

Estimate yield using the Beverton-Holt Yield-per-Recruit (YPR) model. This main function accepts only single values for conditional fishing mortalitiy (`cf`), conditional natural mortality (`cm`), and a minimum length limit for harvest (`minLL`).

Usage

```
yprBH_func(minLL, cf, cm, loi = NA, lhparms, matchRicker = FALSE)
```

Arguments

`minLL`	A single numeric representing the minimum length limit for harvest in mm.
`cf`	A single numeric representing conditional fishing mortality.
`cm`	A single numeric representing conditional natural mortality.
`loi`	A numeric vector for lengths of interest. Used to determine number of fish that reach desired lengths.
`lhparms`	A named vector or list that contains values for each `N0`, `tmax`, `Linf`, `K`, `t0`, `LWalpha`, and `LWbeta`. See `makeLH` for definitions of these life history parameters. Also see details.
`matchRicker`	A logical that indicates whether the yield function should match that in Ricker (). Defaults to `TRUE`. The only reason to changed to `FALSE` is to try to match output from FAMS. See the "YPR_FAMSvRICKER" article.

Details

Details will be filled out later

Value

A data.frame with the following calculated values:

- `yield` is the estimated yield (in g).
- `exploitation` is the exploitation rate.
- `Nharvest` is the number of harvested fish.
- `Ndie` is the number of fish that die of natural deaths.

- `Nt` is the number of fish at time tr (time they become harvestable size).
- `avgwt` is the average weight of fish harvested.
- `avglen` is the average length of fish harvested.
- `tr` is the time for a fish to recruit to a minimum length limit (i.e., time to enter fishery).
- `F` is the instantaneous rate of fishing mortality.
- `M` is the instantaneous rate of natural mortality.
- `Z` is the instantaneous rate of total mortality.
- `S` is the (total) annual rate of survival.
- `N at xxx mm` is the number that reach the length of interest supplied. There will be one column for each length of interest.

For convenience the data.frame also contains the model input values (`minLL`, `cf`, `cm`, `N0`, `Linf`, `K`, `t0`, `LWalpha`, `LWbeta`, and `tmax`).

The data.frame also contains a `notes` value which may contain abbreviations for "issues" that occurred when computing the results and were adjusted for. The possible abbreviates are as follows:

- `minLL>=Linf`: The minimum length limit (minLL) being explored was greater than the given asymptotic mean length (Linf). For the purpose (only) of computing the time at recruitment to the fishery (tr) the Linf was set to minLL+0.1.
- `tr<t0`: The age at recruitment to the fishery (tr) was less than the hypothetical time when the mean length is zero (t0). The fish can't recruit to the fishery prior to having length 0 so tr was set to t0. This also assures that the time it takes to recruit to the fishery is greater than 0.
- `Nt<0`: The number of fish recruiting to the fishery was less than 0. There cannot be negative fish, so Nt was then set to 0.
- `Nt>N0`: The number of fish recruiting to the fishery was more than the number of fish recruited to the populations. Fish cannot be added to the cohort, so Nt was set to N0.
- `Y=Infinite`: The calculated yield (Y) was inifinity, which is impossible and suggests some other propblem. Yield was set to NA.
- `Y<0`: The calculated yield (Y) was negative, which is impossible. Yield was set to 0.
- `Nharv<0`: The calculated number of fish harvested (Nharv) was negative, which is not possible. Number harvested was set to 0.
- `Nharv>Nt`: The calculated number of fish harvested (Nharv) was greater than the number of fish recruiting to the fishery, which is impossible. The number harvested was set to the number recruiting to the fishery.
- `Ndie<0`: The calculated number of fish recruiting to the fishery that died naturally (Ndie) was negative, which is not possible. Number that died was set to 0.
- `Ndie>Nt`: The calculated number of fish recruiting to the fishery that died naturally (Ndie) was greater than the number of fish recruiting to the fishery, which is impossible. The number that died was set to the number recruiting to the fishery.
- `agvglen<minLL`: The average length of harvested fish was less than the given minimum length limit being explored, which is not possible (with only legal harvest). The average length was set to the minimum length limit.

Author(s)

Jason C. Doll, <`jason.doll@fmarion.edu`>

References

Ricker, W.E. 1975. Computation and interpretation of biological statistics of fish populations. Technical Report Bulletin 191, Bulletin of the Fisheries Research Board of Canada. Was (is?) from `https://waves-vagues.dfo-mpo.gc.ca/library-bibliotheque/1485.pdf`.

Slipke, J.W., and M.J. Maceina. 2014. Fishery analysis and modeling simulator. v1.64. American Fisheries Society, Bethesda, MD.

See Also

`yprBH_minLL_fixed` and `yprBH_minLL_var` for simulating yield with multiple values of `cf`, `cm`, and `minLL`.

Examples

```
#' # Life history parameters to be used below
LH <- makeLH(N0=100,tmax=15,Linf=592,K=0.20,t0=-0.3,LWalpha=-5.528,LWbeta=3.273)

# Estimate yield with fixed parameters
Res_1 <- yprBH_func(minLL=355,cf=0.45,cm=0.25,
                    loi=c(200,250,300,325,350),lhparms=LH)
Res_1
```

`yprBH_minLL_fixed` *Main function to simulate expected yield using the Beverton-Holt Yield-per-Recruit model for a range of input parameters*

Description

Estimate yield using the Beverton-Holt Yield-per-Recruit (YPR) model using a range of values for conditional fishing (`cf`) and natural (`cm`) mortality and a single fixed minimum length limit for harvest (`minLL`).

Usage

```
yprBH_minLL_fixed(
  minLL,
  cfmin,
  cfmax,
  cfinc,
  cmmin,
  cmmax,
  cminc,
```

```
  loi = NA,
  lhparms,
  matchRicker = FALSE
)
```

Arguments

`minLL`	The minimum length limit for harvest in mm
`cfmin`	A single numeric for minimum conditional fishing mortality.
`cfmax`	A single numeric for maximum conditional fishing mortality.
`cfinc`	A single numeric for increment to cycle from minimum to maximum conditional fishing mortality.
`cmmin`	A single numeric for minimum conditional natural mortality.
`cmmax`	A single numeric for maximum conditional natural mortality.
`cminc`	A single numeric for increment to cycle from minimum to maximum conditional natural mortality.
`loi`	A numeric vector for lengths of interest. Used to determine number of fish that reach desired lengths.
`lhparms`	A named vector or list that contains values for each `N0`, `tmax`, `Linf`, `K`, `t0`, `LWalpha`, and `LWbeta`. See `makeLH` for definitions of these life history parameters. Also see details.
`matchRicker`	A logical that indicates whether the yield function should match that in Ricker (). Defaults to `TRUE`. The only reason to changed to `FALSE` is to try to match output from FAMS. See the "YPR_FAMSvRICKER" article.

Details

Details will be filled out later

Value

A data.frame with the following calculated values:

- `yield` is the estimated yield (in g).
- `exploitation` is the exploitation rate.
- `Nharvest` is the number of harvested fish.
- `Ndie` is the number of fish that die of natural deaths.
- `Nt` is the number of fish at time tr (time they become harvestable size).
- `avgwt` is the average weight of fish harvested.
- `avglen` is the average length of fish harvested.
- `tr` is the time for a fish to recruit to a minimum length limit (i.e., time to enter fishery).
- `F` is the instantaneous rate of fishing mortality.
- `M` is the instantaneous rate of natural mortality.
- `Z` is the instantaneous rate of total mortality.

- `S` is the (total) annual rate of survival.
- `N at xxx mm` is the number that reach the length of interest supplied. There will be one column for each length of interest.

For convenience the data.frame also contains the model input values (`minLL`; `cf` derived from `cfmin`, `cfmax`, and `cfinc`; `cm` derived from `cmmin`, `cmmax`, and `cminc`; `N0`; `Linf`; `K`; `t0`; `LWalpha`; `LWbeta`; and `tmax`).

The data.frame also contains a `notes` value which may contain abbreviations for "issues" that occurred when computing the results and were adjusted for. The possible abbreviates are defined under "values" in the documentation for `yprBH_func`.

Author(s)

Jason C. Doll, <`jason.doll@fmarion.edu`>

See Also

`yprBH_func` for estimating yield from single values of `cf`, `cm`, and `minLL`, and `yprBH_minLL_var` for simulating yield with multiple values of `cf`, `cm`, and `minLL`.

Examples

```
# Life history parameters to be used below
LH <- makeLH(N0=100,tmax=15,Linf=592,K=0.20,t0=-0.3,LWalpha=-5.528,LWbeta=3.273)

# Estimate yield for multiple values of minLL, cf, and cm
# # This is a minimal example, lengthinc, cfinc, cminc would likely be smaller
# #   to produce finer-scaled results
Res_1 <- yprBH_minLL_fixed(minLL=200,
                         cfmin=0.1,cfmax=0.9,cfinc=0.1,
                         cmmin=0.1,cmmax=0.9,cminc=0.1,
                         loi=c(200,250,300,350),lhparms=LH)

# Load other required packages for organizing output and plotting
library(dplyr)    ## for filter
library(ggplot2)  ## for ggplot et al.
library(tidyr)    ## for pivot_longer

# Custom theme for plots (to make look nice)
theme_FAMS <- function(...) {
  theme_bw() +
  theme(
    panel.grid.major=element_blank(),panel.grid.minor=element_blank(),
    axis.text=element_text(size=14,color="black"),
    axis.title=element_text(size=16,color="black"),
    axis.title.y=element_text(angle=90),
    axis.line=element_line(color="black"),
    panel.border=element_blank()
  )
}
```

```
# Yield curve (yield vs exploitation)
# Extract results for cm=0.40
plot_dat <- Res_1 |> dplyr::filter(cm==0.40)

ggplot(data=plot_dat,mapping=aes(x=u,y=yield)) +
  geom_point() +
  geom_line() +
  labs(y="Yield (g)",x="Exploitation (u)") +
  theme_FAMS()

#Plot number of fish reaching 300 mm as a function of exploitation with cm = 0.40
ggplot(data=plot_dat,mapping=aes(x=u,y=`N at 300 mm`)) +
  geom_point() +
  geom_line() +
  labs(y="Number of fish at 300 mm",x="Exploitation (u)") +
  theme_FAMS()
# Plot number of fish reaching 300 mm as a function of exploitation with cm = 0.40
# Select columns for plotting and convert to long
plot_data_long <- plot_dat %>%
  select(u,`N at 200 mm`, `N at 250 mm`, `N at 300 mm`, `N at 350 mm`) %>%
  pivot_longer(!u, names_to="loi",values_to="number")

# Generate plot
ggplot(data=plot_data_long,mapping=aes(x=u,y=number,group=loi,color=loi)) +
  geom_point() +
  scale_color_discrete(name="Yield",labels=c("N at 200 mm",
                     "N at 250 mm", "N at 300 mm", "N at 350 mm"))+
  geom_line() +
  labs(y="Number of fish",x="Exploitation (u)") +
  theme_FAMS() +
  theme(legend.position = "top")+
  guides(color=guide_legend(title="Length of interest"))
```

`yprBH_minLL_var` *Main function to simulate expected yield using the Beverton-Holt Yield-per-Recruit model for a range of input parameters, including minimum length limits for harvest*

Description

Estimate yield using the Beverton-Holt Yield-per-Recruit (YPR) model using ranges of values for conditional fishing mortality (`cf`), conditional natural mortality (`cm`), and minimum length limits for harvest (`minLL`).

Usage

```
yprBH_minLL_var(
  lengthmin,
  lengthmax,
```

```
  lengthinc,
  cfmin,
  cfmax,
  cfinc,
  cmmin,
  cmmax,
  cminc,
  loi = NA,
  lhparms,
  matchRicker = FALSE
)
```

Arguments

`lengthmin`	A single numeric for the lower limit of minimum length limit for harvest in mm.
`lengthmax`	A single numeric for the upper limit of minimum length limit for harvest in mm.
`lengthinc`	A single numeric for the increment to cycle from lower to upper minimum length limit for harvest in mm.
`cfmin`	A single numeric for minimum conditional fishing mortality.
`cfmax`	A single numeric for maximum conditional fishing mortality.
`cfinc`	A single numeric for increment to cycle from minimum to maximum conditional fishing mortality.
`cmmin`	A single numeric for minimum conditional natural mortality.
`cmmax`	A single numeric for maximum conditional natural mortality.
`cminc`	A single numeric for increment to cycle from minimum to maximum conditional natural mortality.
`loi`	A numeric vector for lengths of interest. Used to determine number of fish that reach desired lengths.
`lhparms`	A named vector or list that contains values for each `N0`, `tmax`, `Linf`, `K`, `t0`, `LWalpha`, and `LWbeta`. See `makeLH` for definitions of these life history parameters. Also see details.
`matchRicker`	A logical that indicates whether the yield function should match that in Ricker (). Defaults to `TRUE`. The only reason to changed to `FALSE` is to try to match output from FAMS. See the "YPR_FAMSvRICKER" article.

Details

Details will be filled out later

Value

A data.frame with the following calculated values:

- `yield` is the estimated yield (in g).
- `exploitation` is the exploitation rate.
- `Nharvest` is the number of harvested fish.

- `Ndie` is the number of fish that die of natural deaths.
- `Nt` is the number of fish at time tr (time they become harvestable size).
- `avgwt` is the average weight of fish harvested.
- `avglen` is the average length of fish harvested.
- `tr` is the time for a fish to recruit to a minimum length limit (i.e., time to enter fishery).
- `F` is the instantaneous rate of fishing mortality.
- `M` is the instantaneous rate of natural mortality.
- `Z` is the instantaneous rate of total mortality.
- `S` is the (total) annual rate of survival.
- `N at xxx mm` is the number that reach the length of interest supplied. There will be one column for each length of interest.

For convenience the data.frame also contains the model input values (`minLL` derived from `lengthmin`, `lengthmax`, and `lengthinc`; `cf` derived from `cfmin`, `cfmax`, and `cfinc`; `cm` derived from `cmmin`, `cmmax`, and `cminc`; `N0`; `Linf`; `K`; `t0`; `LWalpha`; `LWbeta`; and `tmax`).

The data.frame also contains a `notes` value which may contain abbreviations for "issues" that occurred when computing the results and were adjusted for. The possible abbreviates are defined under "values" in the documentation for `yprBH_func`.

Author(s)

Jason C. Doll, `<jason.doll@fmarion.edu>`

See Also

`yprBH_func` for estimating yield from single values of `cf`, `cm`, and `minLL`, and `yprBH_minLL_fixed` for simulating yield with multiple values of `cf` and `cm` but a fixed value for `minLL`.

Examples

```
# Life history parameters to be used below
LH <- makeLH(N0=100,tmax=15,Linf=592,K=0.20,t0=-0.3,LWalpha=-5.528,LWbeta=3.273)

# Estimate yield for multiple values of minLL, cf, and cm
# # This is a minimal example, lengthinc, cfinc, cminc would likely be smaller
# #   to produce finer-scaled results.
Res_1 <- yprBH_minLL_var(lengthmin=200,lengthinc=50,lengthmax=550,
                       cfmin=0.1,cfmax=0.9,cfinc=0.1,
                       cmmin=0.1,cmmax=0.9,cminc=0.1,
                       loi=c(400,450,500,550),lhparms=LH)

# Load other required packages for organizing output and plotting
library(dplyr)    ## for filter
library(ggplot2)  ## for ggplot et al.
library(metR)     ## geom_text_contour

# Custom theme for plots (to make look nice)
theme_FAMS <- function(...) {
```

```
    theme_bw() +
    theme(
      panel.grid.major=element_blank(),panel.grid.minor=element_blank(),
      axis.text=element_text(size=14,color="black"),
      axis.title=element_text(size=16,color="black"),
      axis.title.y=element_text(angle=90),
      axis.line=element_line(color="black"),
      panel.border=element_blank()
    )
  }

  # Yield curve (yield vs exploitation)
  # Extract results for cm=0.40 and minimum length limit=400
  plot_dat <- Res_1 |> dplyr::filter(cm==0.40,minLL==400)

  ggplot(data=plot_dat,mapping=aes(x=u,y=yield)) +
    geom_point() +
    geom_line() +
    labs(y="Yield (g)",x="Exploitation (u)") +
    theme_FAMS()

  # Yield curves by varying minimum lengths, using cm=40
  plot_dat <- Res_1 |> filter(cm==0.40)

  ggplot(data=plot_dat,mapping=aes(y=yield,x=u,
                                   group=minLL,color=minLL)) +
    geom_line(linewidth=1) +
    scale_color_gradient2(high="black") +
    labs(y="Yield (g)",x="Exploitation (u)",color="Min Length Limit") +
    theme_FAMS()

  # Yield isopleths for varying minLL and exploitation with cm=0.40
  # # Using same data as previous example
  ggplot(data=plot_dat,mapping=aes(x=u,y=minLL,z=yield)) +
    geom_contour2(aes(label = after_stat(level))) +
    xlab("Exploitation (u)") +
    ylab("Minimum length limit (mm)") +
    theme_FAMS()

  # Same as previous but using number harvested isopleths
  ggplot(data=plot_dat,mapping=aes(x=u,y=minLL,z=Nharvest)) +
    geom_contour2(aes(label = after_stat(level))) +
    xlab("Exploitation (u)")+
    ylab("Minimum length limit (mm)")+
    theme_FAMS()
```

`yprBH_SlotLL` *Main function to simulate expected yield using the Beverton-Holt Yield Per Recruit model for a slot limit*

Description

Main wrapper function to estimate yield using the Beverton-Holt YPR model. This main function accepts a range of values for cf, cm, recruitment length, lower slot limit length, and upper slot limit length.

Usage

```
yprBH_SlotLL(
  recruitmentTL,
  lowerSL,
  upperSL,
  cfunder,
  cfin,
  cfabove,
  cmmin,
  cmmax,
  cminc,
  loi = NA,
  lhparms,
  matchRicker = FALSE
)
```

Arguments

`recruitmentTL`	A numeric representing the minimum length limit for recruiting to the fishery in mm.
`lowerSL`	A numeric representing the length of the lower slot limit in mm.
`upperSL`	A numeric representing the length of the upper slot limit in mm.
`cfunder`	Single value, conditional fishing mortality under the lower slot limit.
`cfin`	Single value, conditional fishing mortality within the lower and upper slot limit.
`cfabove`	Single value, conditional fishing mortality over the upper slot limit.
`cmmin`	Single value, minimum conditional natural mortality
`cmmax`	Single value, maximum conditional natural mortality
`cminc`	Single value, increment to cycle from minimum to maximum conditional natural mortality
`loi`	A numeric vector for lengths of interest. Used to determine number of fish that reach desired lengths.
`lhparms`	A named vector or list that contains values for each `N0`, `tmax`, `Linf`, `K`, `t0`, `LWalpha`, and `LWbeta`. See `makeLH` for definitions of these life history parameters. Also see details.
`matchRicker`	A logical that indicates whether the yield function should match that in Ricker (). Defaults to `TRUE`. The only reason to changed to `FALSE` is to try to match output from FAMS. See the "YPR_FAMSvRICKER" article.

Details

Details will be filled out later

Value

A data.frame with the following calculated values:

- cm A numeric representing conditional natural mortality
- TotalYield is the calculated total yield
- TotalHarvest is the calculated total number of harvested fish
- TotalNdie is the calculated total number of fish that die of natural death
- yieldUnder is the calculated yield under the slot limit
- yieldIn is the calculated yied within the slot limit
- yieldAbove is the calculated yield above the slot limit
- exploitationUnder is the exploitation rate under the slot limit
- exploitationIn is the exploitation rate within the slot limit
- exploitationAbove is the exploitation rate above the slot limit
- NharvestUnder is the number of harvested fish under the slot limit
- NharvestIn is the number of harvested fish within the slot limit
- NharvestAbove is the number of harvested fish above the slot limit
- NdieUnder is the number of fish that die of natural death under the slot limit
- NdieIn is the number of fish that die of natural deaths within the slot limit
- NdieAbove is the number of fish that die of natural deaths above the slot limit
- avglenUnder is the average length of fish harvested under the slot limit
- avglenIn is the average length of fish harvested within the slot limit
- avglenAbove is the average length of fish harvested above the slot limit
- avgwtUnder is the average weight of fish harvested under the slot limit
- avgwtIn is the average weight of fish harvested within the slot limit
- avgwtAbove is the average weight of fish harvested above the slot limit
- trUnder is the time for a fish to recruit to a minimum length limit (i.e., time to enter fishery)
- trIn is the time for a fish to recruit to a lower length limit of the slot limit
- trOver is the time for a fish to recruit to a upper length limit of the slot limit
- NtUnder is the number of fish at time trUnder (time they become harvestable size under the slot limit)
- NtIn is the number of fish at time trIn (time they reach the lower slot limit size)
- NtAbove is the number of fish at time trAbove (time they reach the upper slot limit size)
- FUnder is the estimated instantaneous rate of fishing mortality under the slot limit
- FIn is the estimated instantaneous rate of fishing mortality within the slot limit
- FAbove is the estimated instantaneous rate of fishing mortality above the slot limit

- MUnder is the estimated instantaneous rate of natural mortality under the slot limit
- MIn is the estimated instantaneous rate of natural mortality within the slot limit
- MAbove is the estimated instantaneous rate of natural mortality above the slot limit
- ZUnder is the estimated instantaneous rate of total mortality under the slot limit
- ZIn is the estimated instantaneous rate of total mortality within the slot limit
- ZAbove is the estimated instantaneous rate of total mortality above the slot limit
- SUnder is the estimated total survival under the slot limit
- SIn is the estimated total survival within the slot limit
- SAbove is the estimated total survival above the slot limit
- cfUnder A numeric representing conditional fishing mortality
- cfIn A numeric representing conditional fishing mortality
- cfOver A numeric representing conditional fishing mortality
- recruitmentTL A numeric representing the minimum length limit for recruiting to the fishery in mm.
- lowerSL A numeric representing the length of the lower slot limit in mm.
- upperSL A numeric representing the length of the upper slot limit in mm.
- N0 A numeric representing the initial number of new recruits entering the fishery OR a vector or list that contains named values for each `N0`, `Linf`, `K`, `t0`, `LWalpha`, `LWbeta`, and `tmax`
- Linf A numeric representing the point estimate of the asymptotic mean length (L-infinity) from the von Bertalanffy growth model in mm
- K A numeric representing the point estimate of the Brody growth coefficient from the von Bertalanffy growth model
- t0 A numeric representing the point estimate of the x-intercept (i.e., theoretical age at a mean length of 0) from the von Bertalanffy growth model
- LWalpha A numeric representing the point estimate of alpha from the length-weight regression on the log10 scale.
- LWbeta A numeric representing the point estimate of beta from the length-weight regression on the log10 scale.
- tmax An integer representing maximum age in the population in years
- `N at xxx mm` is the number that reach the length of interest supplied. There will be one column for each length of interest. #'

Author(s)

Jason C. Doll, `<jason.doll@fmarion.edu>`

Examples

```
#Load other required packages for organizing output and plotting
library(ggplot2)
library(dplyr)
library(tidyr)
```

```
# Custom theme for plots (to make look nice)
theme_FAMS <- function(...) {
  theme_bw() +
  theme(
    panel.grid.major=element_blank(),panel.grid.minor=element_blank(),
    axis.text=element_text(size=14,color="black"),
    axis.title=element_text(size=16,color="black"),
    axis.title.y=element_text(angle=90),
    axis.line=element_line(color="black"),
    panel.border=element_blank()
  )
}

# Life history parameters to be used below
LH <- makeLH(N0=100,tmax=15,Linf=592,K=0.20,t0=-0.3,LWalpha=-5.528,LWbeta=3.273)

#Estimate yield
 Res_1 <- yprBH_SlotLL(recruitmentTL=200,lowerSL=250,upperSL=325,
                      cfunder=0.25,cfin=0.6,cfabove=0.15,cmmin=0.3,cmmax=0.6,cminc=0.05,
                       loi=c(200,250,300,325,350),lhparms=LH)

 Res_1

# Plot results
# Total Yield vs Conditional Natural Mortality (cm)
ggplot(data=Res_1,mapping=aes(x=cm,y=TotalYield)) +
  geom_point() +
  geom_line() +
  labs(y="Total Yield (g)",x="Conditional Natural Mortality (cm)") +
  theme_FAMS()


# Yield under, in, and above the slot limit vs Conditional Natural Mortality (cm)
# Select columns for plotting
plot_data <- Res_1 %>%
  select(cm, yieldUnder, yieldIn, yieldAbove) %>%
  pivot_longer(!cm, names_to="YieldCat",values_to="Yield")

# Generate plot
ggplot(data=plot_data,mapping=aes(x=cm,y=Yield,group=YieldCat,color=YieldCat)) +
  geom_point() +
  scale_color_discrete(name="Yield",labels=c("Above SL","In SL","Under SL"))+
  geom_line() +
  labs(y="Total Yield (g)",x="Conditional Natural Mortality (cm)") +
  theme_FAMS() +
  theme(legend.position = "top")+
  guides(color=guide_legend(title="Yield"))
```

`yprBH_slot_func` *Function to simulate expected yield using the Beverton-Holt Yield Per Recruit model for single input parameters*

Description

Function to estimate yield using the Beverton-Holt YPR model. This main function accepts only single values for cf, cm, and minlength. Use the wrapper ypr() function for specifying range of cf, cm, and minlength

Usage

```
yprBH_slot_func(
  recruitmentTL,
  lowerSL,
  upperSL,
  cfunder,
  cfin,
  cfabove,
  cm,
  loi = NA,
  lhparms,
  matchRicker = FALSE
)
```

Arguments

`recruitmentTL`	A numeric representing the minimum length limit for recruiting to the fishery in mm.
`lowerSL`	A numeric representing the length of the lower slot limit in mm.
`upperSL`	A numeric representing the length of the upper slot limit in mm.
`cfunder`	Single value, conditional fishing mortality under the lower slot limit.
`cfin`	Single value, conditional fishing mortality within the lower and upper slot limit.
`cfabove`	Single value, conditional fishing mortality over the upper slot limit.
`cm`	A numeric representing conditional natural mortality
`loi`	A numeric vector for lengths of interest. Used to determine number of fish that reach desired lengths.
`lhparms`	A named vector or list that contains values for each `N0`, `tmax`, `Linf`, `K`, `t0`, `LWalpha`, and `LWbeta`. See `makeLH` for definitions of these life history parameters. Also see details.
`matchRicker`	A logical that indicates whether the yield function should match that in Ricker (). Defaults to `TRUE`. The only reason to changed to `FALSE` is to try to match output from FAMS. See the "YPR_FAMSvRICKER" article.

Details

Details will be filled out later

Value

the following calculated and input values in a data.frame

- cm A numeric representing conditional natural mortality
- TotalYield is the calculated total yield
- TotalHarvest is the calculated total number of harvested fish
- TotalNdie is the calculated total number of fish that die of natural death
- yieldUnder is the calculated yield under the slot limit
- yieldIn is the calculated yield within the slot limit
- yieldAbove is the calculated yield above the slot limit
- uUnder is the exploitation rate under the slot limit
- uIn is the exploitation rate within the slot limit
- uAbove is the exploitation rate above the slot limit
- NharvestUnder is the number of harvested fish under the slot limit
- NharvestIn is the number of harvested fish within the slot limit
- NharvestAbove is the number of harvested fish above the slot limit
- N0die is the number of fish that die of natural death before entering the fishery at a minimum length
- NdieUnder is the number of fish that die of natural death between entering the fishery and the lower slot limit
- NdieIn is the number of fish that die of natural deaths within the slot limit
- NdieAbove is the number of fish that die of natural deaths above the slot limit
- avglenUnder is the average length of fish harvested under the slot limit
- avglenIn is the average length of fish harvested within the slot limit
- avglenAbove is the average length of fish harvested above the slot limit
- avgwtUnder is the average weight of fish harvested under the slot limit
- avgwtIn is the average weight of fish harvested within the slot limit
- avgwtAbove is the average weight of fish harvested above the slot limit
- trUnder is the time for a fish to recruit to a minimum length limit (i.e., time to enter fishery)
- trIn is the time for a fish to recruit to a lower length limit of the slot limit
- trOver is the time for a fish to recruit to a upper length limit of the slot limit
- NrUnder is the number of fish at time trUnder (time they become harvestable size under the slot limit)
- NrIn is the number of fish at time trIn (time they reach the lower slot limit size)
- NrAbove is the number of fish at time trAbove (time they reach the upper slot limit size)
- FUnder is the estimated instantaneous rate of fishing mortality under the slot limit
- FIn is the estimated instantaneous rate of fishing mortality within the slot limit
- FAbove is the estimated instantaneous rate of fishing mortality above the slot limit

- MUnder is the estimated instantaneous rate of natural mortality under the slot limit
- MIn is the estimated instantaneous rate of natural mortality within the slot limit
- MAbove is the estimated instantaneous rate of natural mortality above the slot limit
- ZUnder is the estimated instantaneous rate of total mortality under the slot limit
- ZIn is the estimated instantaneous rate of total mortality within the slot limit
- ZAbove is the estimated instantaneous rate of total mortality above the slot limit
- SUnder is the estimated total survival under the slot limit
- SIn is the estimated total survival within the slot limit
- SAbove is the estimated total survival above the slot limit
- cfUnder A numeric representing conditional fishing mortality
- cfIn A numeric representing conditional fishing mortality
- cfOver A numeric representing conditional fishing mortality
- recruitmentTL A numeric representing the minimum length limit for recruiting to the fishery in mm.
- lowerSL A numeric representing the length of the lower slot limit in mm.
- upperSL A numeric representing the length of the upper slot limit in mm.
- N0 A numeric representing the initial number of new recruits entering the fishery OR a vector or list that contains named values for each `N0`, `Linf`, `K`, `t0`, `LWalpha`, `LWbeta`, and `tmax`
- Linf A numeric representing the point estimate of the asymptotic mean length (L-infinity) from the von Bertalanffy growth model in mm
- K A numeric representing the point estimate of the Brody growth coefficient from the von Bertalanffy growth model
- t0 A numeric representing the point estimate of the x-intercept (i.e., theoretical age at a mean length of 0) from the von Bertalanffy growth model
- LWalpha A numeric representing the point estimate of alpha from the length-weight regression on the log10 scale.
- LWbeta A numeric representing the point estimate of beta from the length-weight regression on the log10 scale.
- tmax An integer representing maximum age in the population in years
- `N at xxx mm` is the number that reach the length of interest supplied. There will be one column for each length of interest. #'

Author(s)

Jason C. Doll, `<jason.doll@fmarion.edu>`

Examples

```
# Life history parameters to be used below
LH <- makeLH(N0=100,tmax=15,Linf=592,K=0.20,t0=-0.3,LWalpha=-5.528,LWbeta=3.273)

# Estimate yield with fixed parameters
Res_1 <- yprBH_slot_func(recruitmentTL=200,lowerSL=250,upperSL=325,
```

```
                       cfunder=0.25,cfin=0.6,cfabove=0.15,cm=0.4,
                       loi=c(200,250,300,325,350),lhparms=LH)
Res_1
```

Index