Package 'fishgrowth'

April 8, 2025

Version 1.0.2

Date 2025-04-08

Title Fit Growth Curves to Fish Data

Depends R (>= 2.10), RTMB

Suggests areaplot

LazyData yes

Description Fit growth models to otoliths and/or tagging data, using the 'RTMB' package and maximum likelihood. The otoliths (or similar measurements of age) provide direct observed coordinates of age and length. The tagging data provide information about the observed length at release and length at recapture at a later time, where the age at release is unknown and estimated as a vector of parameters. The growth models provided by this package can be fitted to otoliths only, tagging data only, or a combination of the two. Growth variability can be modelled as constant or increasing with length.

License GPL-3

URL https://github.com/arni-magnusson/fishgrowth

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Arni Magnusson [aut, cre],
Mark Maunder [aut]

Maintainer Arni Magnusson <thisisarni@gmail.com>

Repository CRAN

Date/Publication 2025-04-08 04:10:02 UTC

Contents

`fishgrowth-package` *Fit Growth Curves to Fish Data*

Description

Fit growth models to otoliths and/or tagging data, using the `RTMB` package and maximum likelihood.

The otoliths (or similar measurements of age) provide direct observed coordinates of age and length. The tagging data provide information about the observed length at release and length at recapture at a later time, where the age at release is unknown and estimated as a vector of parameters.

The growth models provided by this package can be fitted to otoliths only, tagging data only, or a combination of the two. Growth variability can be modelled as constant or increasing with length.

Details

Growth models:

`gcm`	growth cessation
`gompertz`	Gompertz
`gompertzo`	Gompertz (old style)
`richards`	Richards
`richardso`	Richards (old style)
`schnute3`	Schnute Case 3
`vonbert`	von Bertalanffy
`vonberto`	von Bertalanffy (old style)

Utilities:

`pred_band`	prediction band

Example data:

`otoliths_had`	otoliths (haddock)
`otoliths_skj`	otoliths (skipjack)

`tags_skj`	tags (skipjack)

Note

The parameter estimation method follows the statistical approach of Maunder et al. (2018), which stems from Aires-da-Silva et al. (2015), Eveson et al. (2004), and Laslett et al. (2002), collectively called the Laslett-Eveson-Polacheck approach.

Author(s)

Arni Magnusson and Mark Maunder.

References

Aires-da-Silva, A.M., Maunder, M.N., Schaefer, K.M., and Fuller, D.W. (2015). Improved growth estimates from integrated analysis of direct aging and tag-recapture data: An illustration with bigeye tuna (*Thunnus obesus*) of the eastern Pacific Ocean with implications for management. *Fisheries Research*, **163**, 119–126. doi:10.1016/j.fishres.2014.04.001.

Maunder, M.N., Deriso, R.B., Schaefer, K.M., Fuller, D.W., Aires-da-Silva, A.M., Minte-Vera, C.V., and Campana, S.E. (2018). The growth cessation model: a growth model for species showing a near cessation in growth with application to bigeye tuna (*Thunnus obesus*). *Marine Biology*, **165**, 76. doi:10.1007/s0022701833369.

Eveson, J.P., Laslett, G.M., and Polacheck, T. (2004). An integrated model for growth incorporating tag-recapture, length-frequency, and direct aging data. *Canadian Journal of Fisheries and Aquatic Sciences*, **61**, 292–306. doi:10.1139/f03163.

Laslett, G.M., Eveson, J.P., and Polacheck, T. (2002). A flexible maximum likelihood approach for fitting growth curves to tag-recapture data. *Canadian Journal of Fisheries and Aquatic Sciences*, **59**, 976–986. doi:10.1139/f02069.

See Also

Useful links:

- `https://github.com/arni-magnusson/fishgrowth`

`gcm` *Growth Cessation Model*

Description

Fit a growth cessation model (GCM) to otoliths and/or tags.

Usage

```
gcm(par, data, silent = TRUE, ...)

gcm_curve(t, L0, rmax, k, t50)

gcm_objfun(par, data)
```

Arguments

`par`	a parameter list.
`data`	a data list.
`silent`	passed to `MakeADFun`.
`...`	passed to `MakeADFun`.
`t`	age (vector).
`L0`	predicted length at age 0.
`rmax`	shape parameter that determines the initial slope.
`k`	shape parameter that determines how quickly the growth curve reaches the asymptotic maximum.
`t50`	shape parameter that determines the logistic function midpoint.

Details

The main function `gcm` creates a model object, ready for parameter estimation. The auxiliary functions `gcm_curve` and `gcm_objfun` are called by the main function to calculate the regression curve and objective function value. The user can also call the auxiliary functions directly for plotting and model exploration.

The `par` list contains the following elements:

- `L0`, predicted length at age 0
- `log_rmax`, shape parameter that determines the initial slope
- `log_k`, shape parameter that determines how quickly the growth curve reaches the asymptotic maximum
- `t50`, shape parameter that determines the logistic function midpoint
- `log_sigma_min`, growth variability at the shortest observed length in the data
- `log_sigma_max` (*), growth variability at the longest observed length in the data
- `log_age` (*), age at release of tagged individuals (vector)

*: The parameter `log_sigma_max` can be omitted to estimate growth variability that does not vary with length. The parameter vector `log_age` can be omitted to fit to otoliths only.

The `data` list contains the following elements:

- `Aoto` (*), age from otoliths (vector)
- `Loto` (*), length from otoliths (vector)
- `Lrel` (*), length at release of tagged individuals (vector)
- `Lrec` (*), length at recapture of tagged individuals (vector)
- `liberty` (*), time at liberty of tagged individuals in years (vector)

*: The data vectors `Aoto` and `Loto` can be omitted to fit to tagging data only. The data vectors `Lrel`, `Lrec`, and `liberty` can be omitted to fit to otoliths only.

Value

The gcm function returns a TMB model object, produced by MakeADFun.

The gcm_curve function returns a numeric vector of predicted length at age.

The gcm_objfun function returns the negative log-likelihood as a single number, describing the goodness of fit of par to the data.

Note

The growth cessation model (Maunder et al. 2018) predicts length at age as:

$$\hat{L}_t \;=\; L_0 \;+\; r_{\max} \left[\frac{\log\left(1+e^{-kt_{50}}\right) \;-\; \log\left(1+e^{k(t-t_{50})}\right)}{k} \;+\; t \right]$$

The variability of length at age increases linearly with length,

$$\sigma_L \;=\; \alpha \;+\; \beta \hat{L}$$

where the slope is $\beta = (\sigma_{\max} - \sigma_{\min})/(L_{\max} - L_{\min})$, the intercept is $\alpha = \sigma_{\min} - \beta L_{\min}$, and $L_{\min}$ and $L_{\max}$ are the shortest and longest observed lengths in the data. Alternatively, growth variability can be modelled as a constant $\sigma_L = \sigma_{\min}$ that does not vary with length, by omitting log_sigma_max from the parameter list (see above).

The negative log-likelihood is calculated by comparing the observed and predicted lengths:

```
nll_Loto <- -dnorm(Loto, Loto_hat, sigma_Loto, TRUE)
nll_Lrel <- -dnorm(Lrel, Lrel_hat, sigma_Lrel, TRUE)
nll_Lrec <- -dnorm(Lrec, Lrec_hat, sigma_Lrec, TRUE)
nll <- sum(nll_Loto) + sum(nll_Lrel) + sum(nll_Lrec)
```

References

Maunder, M.N., Deriso, R.B., Schaefer, K.M., Fuller, D.W., Aires-da-Silva, A.M., Minte-Vera, C.V., and Campana, S.E. (2018). The growth cessation model: a growth model for species showing a near cessation in growth with application to bigeye tuna (*Thunnus obesus*). *Marine Biology*, **165**, 76. doi:10.1007/s0022701833369.

The fishgrowth-package help page includes references describing the parameter estimation method.

See Also

gcm, gompertz, gompertzo, richards, richardso, schnute3, vonbert, and vonberto are alternative growth models.

pred_band calculates a prediction band for a fitted growth model.

otoliths_had, otoliths_skj, and tags_skj are example datasets.

fishgrowth-package gives an overview of the package.

Examples

```
# Model 1: Fit to haddock otoliths

# Explore initial parameter values
plot(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,105), pch=16,
     col=”#0080a010”)
x <- seq(1, 18, 0.1)
lines(x, gcm_curve(x, L0=5, rmax=20, k=0.15, t50=0), lty=3)

# Prepare parameters and data
init <- list(L0=5, log_rmax=log(20), log_k=log(0.15), t50=-1,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_had$age, Loto=otoliths_had$len)
gcm_objfun(init, dat)

# Fit model
model <- gcm(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
Lhat <- with(report, gcm_curve(x, L0, rmax, k, t50))
lines(x, Lhat, lwd=2, col=2)
legend(”bottomright”, c(”initial curve”,”model fit”), col=c(1,2), lty=c(3,1),
       lwd=c(1,2), bty=”n”, inset=0.02, y.intersp=1.25)

# Model summary
report[c(”L0”, ”rmax”, ”k”, ”t50”, ”sigma_min”, ”sigma_max”)]
fit[-1]
summary(sdreport)

# Plot 95% prediction band
band <- pred_band(x, model)
areaplot::confplot(cbind(lower,upper)~age, band, xlim=c(0,18), ylim=c(0,100),
         ylab=”len”, col=”mistyrose”)
points(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,100),
       pch=16, col=”#0080a010”)
lines(x, Lhat, lwd=2, col=2)
lines(lower~age, band, lty=1, lwd=0.5, col=2)
lines(upper~age, band, lty=1, lwd=0.5, col=2)

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

# Model 2: Fit to skipjack otoliths and tags

# Explore initial parameter values
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
x <- seq(0, 4, 0.1)
points(lenRel~I(lenRel/60), tags_skj, col=4)
points(lenRec~I(lenRel/60+liberty), tags_skj, col=3)
```

```
lines(x, gcm_curve(x, L0=20, rmax=120, k=2, t50=0), lty=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "initial curve"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,2),
       lwd=c(1.2,1.2,1.2,1), bty="n", inset=0.02, y.intersp=1.25)

# Prepare parameters and data
init <- list(L0=20, log_rmax=log(120), log_k=log(4), t50=0,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len,
            Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty)
gcm_objfun(init, dat)

# Fit model
model <- gcm(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
points(report$age, report$Lrel, col=4)
points(report$age+report$liberty, report$Lrec, col=3)
Lhat <- with(report, gcm_curve(x, L0, rmax, k, t50))
lines(x, Lhat, lwd=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "model fit"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,1),
       lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L0", "rmax", "k", "t50", "sigma_min", "sigma_max")]
fit[-1]
head(summary(sdreport), 6)

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

# Model 3: Stepwise estimation procedure, described by Maunder et al. (2018)
# - estimate L0 and rmax using linear regression on younger fish
# - estimate k and t50 using GCM and all data, keeping L0 and rmax fixed

# Estimate L0 and rmax
plot(otoliths_skj, xlim=c(0,4), ylim=c(0,100))
fm <- lm(len~age, otoliths_skj)
abline(fm)
L0 <- coef(fm)[[1]]
rmax <- coef(fm)[[2]]

# Explore initial parameter values (k, t50, age)
t <- seq(0, 4, by=0.1)
points(t, gcm_curve(t, L0, rmax, k=3, t50=2), col="gray")
points(lenRel~I(lenRel/50), tags_skj, col=4)
points(lenRec~I(lenRel/50+liberty), tags_skj, col=3)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "linear regression (otoliths)"), col=c(1,4,3,1), pch=c(1,1,1,NA),
       lty=c(0,0,0,1), lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02,
       y.intersp=1.25)

# Prepare parameters
init <- list(L0=L0, log_rmax=log(rmax), log_k=log(3), t50=2,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/50))

# Fit model
map <- list(L0=factor(NA), log_rmax=factor(NA))  # fix L0 and rmax
model <- gcm(init, dat, map=map)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4,iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
points(report$age, report$Lrel, col=4)
points(report$age+report$liberty, report$Lrec, col=3)
Lhat <- with(report, gcm_curve(x, L0, rmax, k, t50))
lines(x, Lhat, lwd=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "model fit"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,1),
       lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L0", "rmax", "k", "t50", "sigma_min", "sigma_max")]
fit[-1]
head(summary(sdreport), 6)
```

gompertz *Gompertz Growth Model*

Description

Fit a Gompertz growth model to otoliths and/or tags, using the Schnute parametrization.

Usage

```
gompertz(par, data, silent = TRUE, ...)

gompertz_curve(t, L1, L2, k, t1, t2)

gompertz_objfun(par, data)
```

Arguments

`par`	a parameter list.
`data`	a data list.
`silent`	passed to `MakeADFun`.
`...`	passed to `MakeADFun`.
`t`	age (vector).
`L1`	predicted length at age `t1`.
`L2`	predicted length at age `t2`.
`k`	growth coefficient.
`t1`	age where predicted length is `L1`.
`t2`	age where predicted length is `L2`.

Details

The main function `gompertz` creates a model object, ready for parameter estimation. The auxiliary functions `gompertz_curve` and `gompertz_objfun` are called by the main function to calculate the regression curve and objective function value. The user can also call the auxiliary functions directly for plotting and model exploration.

The `par` list contains the following elements:

- `log_L1`, predicted length at age `t1`
- `log_L2`, predicted length at age `t2`
- `log_k`, growth coefficient
- `log_sigma_min`, growth variability at the shortest observed length in the data
- `log_sigma_max` (*), growth variability at the longest observed length in the data
- `log_age` (*), age at release of tagged individuals (vector)

*: The parameter `log_sigma_max` can be omitted to estimate growth variability that does not vary with length. The parameter vector `log_age` can be omitted to fit to otoliths only.

The `data` list contains the following elements:

- `Aoto` (*), age from otoliths (vector)
- `Loto` (*), length from otoliths (vector)
- `Lrel` (*), length at release of tagged individuals (vector)
- `Lrec` (*), length at recapture of tagged individuals (vector)
- `liberty` (*), time at liberty of tagged individuals in years (vector)
- `t1`, age where predicted length is `L1`
- `t2`, age where predicted length is `L2`

*: The data vectors `Aoto` and `Loto` can be omitted to fit to tagging data only. The data vectors `Lrel`, `Lrec`, and `liberty` can be omitted to fit to otoliths only.

Value

The gompertz function returns a TMB model object, produced by MakeADFun.

The gompertz_curve function returns a numeric vector of predicted length at age.

The gompertz_objfun function returns the negative log-likelihood as a single number, describing the goodness of fit of par to the data.

Note

The Schnute parametrization used in gompertz reduces parameter correlation and improves convergence reliability compared to the traditional parametrization used in gompertzo. Therefore, the gompertz parametrization can be recommended for general usage, as both parametrizations produce the same growth curve. However, there can be some use cases where the traditional parametrization (Linf, k, tau) is preferred over the Schnute parametrization (L1, L2, k).

Gompertz is a special case of the Richards (1959) model, where $b = 0$. If the best model fit of a richards model to a particular dataset involves a very small estimated value of b, then the gompertz model offers a preferable parametrization, as it produces the same curve using fewer parameters.

The Gompertz (1825) growth model, as parametrized by Schnute (1981, Eq. 16) predicts length at age as:

$$\hat{L}_t \;=\; L_1 \exp\left[\log(L_2/L_1)\,\frac{1-e^{-k(t-t_1)}}{1-e^{-k(t_2-t_1)}}\right]$$

The variability of length at age increases linearly with length,

$$\sigma_L \;=\; \alpha \,+\, \beta\hat{L}$$

where the slope is $\beta = (\sigma_{\max} - \sigma_{\min})/(L_{\max} - L_{\min})$, the intercept is $\alpha = \sigma_{\min} - \beta L_{\min}$, and $L_{\min}$ and $L_{\max}$ are the shortest and longest observed lengths in the data. Alternatively, growth variability can be modelled as a constant $\sigma_L = \sigma_{\min}$ that does not vary with length, by omitting log_sigma_max from the parameter list (see above).

The negative log-likelihood is calculated by comparing the observed and predicted lengths:

```
nll_Loto <- -dnorm(Loto, Loto_hat, sigma_Loto, TRUE)
nll_Lrel <- -dnorm(Lrel, Lrel_hat, sigma_Lrel, TRUE)
nll_Lrec <- -dnorm(Lrec, Lrec_hat, sigma_Lrec, TRUE)
nll <- sum(nll_Loto) + sum(nll_Lrel) + sum(nll_Lrec)
```

References

Gompertz, B. (1825). On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life contingencies. *Philosophical Transactions of the Royal Society*, **115**, 513-583.

Schnute, J. (1981). A versatile growth model with statistically stable parameters. *Canadian Journal of Fisheries and Aquatic Science*, **38**, 1128-1140. doi:10.1139/f81153.

The fishgrowth-package help page includes references describing the parameter estimation method.

See Also

gcm, gompertz, gompertzo, richards, richards, schnute3, vonbert, and vonberto are alternative growth models.

pred_band calculates a prediction band for a fitted growth model.

otoliths_had, otoliths_skj, and tags_skj are example datasets.

fishgrowth-package gives an overview of the package.

Examples

```
# Model 1: Fit to haddock otoliths

# Explore initial parameter values
plot(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,105), pch=16,
     col="#0080a010")
x <- seq(1, 18, 0.1)
lines(x, gompertz_curve(x, L1=15, L2=70, k=0.4, t1=1, t2=10), lty=3)

# Prepare parameters and data
init <- list(log_L1=log(20), log_L2=log(70), log_k=log(0.1),
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_had$age, Loto=otoliths_had$len, t1=1, t2=10)
gompertz_objfun(init, dat)

# Fit model
model <- gompertz(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
Lhat <- with(report, gompertz_curve(x, L1, L2, k, t1, t2))
lines(x, Lhat, lwd=2, col=2)
legend("bottomright", c("initial curve","model fit"), col=c(1,2), lty=c(3,1),
       lwd=c(1,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L1", "L2", "k", "sigma_min", "sigma_max")]
fit[-1]
summary(sdreport)

# Plot 95% prediction band
band <- pred_band(x, model)
areaplot::confplot(cbind(lower,upper)~age, band, xlim=c(0,18), ylim=c(0,100),
         ylab="len", col="mistyrose")
points(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,100),
       pch=16, col="#0080a010")
lines(x, Lhat, lwd=2, col=2)
lines(lower~age, band, lty=1, lwd=0.5, col=2)
lines(upper~age, band, lty=1, lwd=0.5, col=2)
```

```
##############################################################################

# Model 2: Fit to skipjack otoliths and tags

# Explore initial parameter values
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
x <- seq(0, 4, 0.1)
points(lenRel~I(lenRel/60), tags_skj, col=4)
points(lenRec~I(lenRel/60+liberty), tags_skj, col=3)
lines(x, gompertz_curve(x, L1=28, L2=74, k=1, t1=0, t2=4), lty=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "initial curve"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,2),
       lwd=c(1.2,1.2,1.2,1), bty="n", inset=0.02, y.intersp=1.25)

# Prepare parameters and data
init <- list(log_L1=log(28), log_L2=log(74), log_k=log(1),
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len,
            Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty, t1=0, t2=4)
gompertz_objfun(init, dat)

# Fit model
model <- gompertz(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
points(report$age, report$Lrel, col=4)
points(report$age+report$liberty, report$Lrec, col=3)
Lhat <- with(report, gompertz_curve(x, L1, L2, k, t1, t2))
lines(x, Lhat, lwd=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "model fit"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,1),
       lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L1", "L2", "k", "sigma_min", "sigma_max")]
fit[-1]
head(summary(sdreport), 5)

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

# Model 3: Fit to skipjack otoliths only

init <- list(log_L1=log(28), log_L2=log(74), log_k=log(1),
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len, t1=0, t2=4)
model <- gompertz(init, dat)
```

```
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "k", "sigma_min", "sigma_max")]

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

# Model 4: Fit to skipjack otoliths only,
# but now estimating constant sigma instead of sigma varying by length

# We do this by omitting log_sigma_max
init <- list(log_L1=log(28), log_L2=log(74), log_k=log(1),
             log_sigma_min=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len, t1=0, t2=4)
model <- gompertz(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
                    control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "k", "sigma_min")]

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

# Model 5: Fit to skipjack tags only

init <- list(log_L1=log(28), log_L2=log(74), log_k=log(1),
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty, t1=0, t2=4)
model <- gompertz(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
                   control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "k", "sigma_min", "sigma_max")]
```

gompertzo *Gompertz Growth Model (Old Style)*

Description

Fit a Gompertz growth model to otoliths and/or tags, using a traditional parametrization.

Usage

```
gompertzo(par, data, silent = TRUE, ...)

gompertzo_curve(t, Linf, k, tau)

gompertzo_objfun(par, data)
```

Arguments

par	a parameter list.
data	a data list.
silent	passed to MakeADFun.
...	passed to MakeADFun.
t	age (vector).
Linf	asymptotic maximum length.
k	growth coefficient.
tau	location parameter.

Details

The main function gompertzo creates a model object, ready for parameter estimation. The auxiliary functions gompertzo_curve and gompertzo_objfun are called by the main function to calculate the regression curve and objective function value. The user can also call the auxiliary functions directly for plotting and model exploration.

The par list contains the following elements:

- log_Linf, asymptotic maximum length
- log_k, growth coefficient
- tau, location parameter
- log_sigma_min, growth variability at the shortest observed length in the data
- log_sigma_max (*), growth variability at the longest observed length in the data
- log_age (*), age at release of tagged individuals (vector)

*: The parameter log_sigma_max can be omitted to estimate growth variability that does not vary with length. The parameter vector log_age can be omitted to fit to otoliths only.

The data list contains the following elements:

- Aoto (*), age from otoliths (vector)
- Loto (*), length from otoliths (vector)
- Lrel (*), length at release of tagged individuals (vector)
- Lrec (*), length at recapture of tagged individuals (vector)
- liberty (*), time at liberty of tagged individuals in years (vector)

*: The data vectors Aoto and Loto can be omitted to fit to tagging data only. The data vectors Lrel, Lrec, and liberty can be omitted to fit to otoliths only.

Value

The gompertzo function returns a TMB model object, produced by MakeADFun.

The gompertzo_curve function returns a numeric vector of predicted length at age.

The gompertzo_objfun function returns the negative log-likelihood as a single number, describing the goodness of fit of par to the data.

Note

The Schnute parametrization used in gompertz reduces parameter correlation and improves convergence reliability compared to the traditional parametrization used in gompertzo. Therefore, the gompertz parametrization can be recommended for general usage, as both parametrizations produce the same growth curve. However, there can be some use cases where the traditional parametrization (Linf, k, tau) is preferred over the Schnute parametrization (L1, L2, k).

Gompertz is a special case of the Richards (1959) model, where $b = 0$. If the best model fit of a richards model to a particular dataset involves a very small estimated value of b, then the gompertz model offers a preferable parametrization, as it produces the same curve using fewer parameters.

The Gompertz (1825) growth model, as parametrized by Ricker (1979, Eq. 23) predicts length at age as:

$$\hat{L}_t \;=\; L_\infty \exp\!\big(-e^{-k(t-\tau)}\big)$$

The variability of length at age increases linearly with length,

$$\sigma_L \;=\; \alpha \,+\, \beta \hat{L}$$

where the slope is $\beta = (\sigma_{\max} - \sigma_{\min})/(L_{\max} - L_{\min})$, the intercept is $\alpha = \sigma_{\min} - \beta L_{\min}$, and $L_{\min}$ and $L_{\max}$ are the shortest and longest observed lengths in the data. Alternatively, growth variability can be modelled as a constant $\sigma_L = \sigma_{\min}$ that does not vary with length, by omitting log_sigma_max from the parameter list (see above).

The negative log-likelihood is calculated by comparing the observed and predicted lengths:

```
nll_Loto <- -dnorm(Loto, Loto_hat, sigma_Loto, TRUE)
nll_Lrel <- -dnorm(Lrel, Lrel_hat, sigma_Lrel, TRUE)
nll_Lrec <- -dnorm(Lrec, Lrec_hat, sigma_Lrec, TRUE)
nll <- sum(nll_Loto) + sum(nll_Lrel) + sum(nll_Lrec)
```

References

Gompertz, B. (1825). On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life contingencies. *Philosophical Transactions of the Royal Society*, **115**, 513-583.

Ricker, W.E. (1979). Growth rates and models. In: W.S. Hoar et al. (eds.) *Fish physiology 8: Bioenergetics and growth*. New York: Academic Press, pp. 677-743.

The fishgrowth-package help page includes references describing the parameter estimation method.

See Also

gcm, gompertz, gompertzo, richards, richards, schnute3, vonbert, and vonberto are alternative growth models.

pred_band calculates a prediction band for a fitted growth model.

otoliths_had, otoliths_skj, and tags_skj are example datasets.

fishgrowth-package gives an overview of the package.

Examples

```
# Model 1: Fit to haddock otoliths

# Explore initial parameter values
plot(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,105), pch=16,
     col="#0080a010")
x <- seq(1, 18, 0.1)
lines(x, gompertzo_curve(x, Linf=73, k=0.4, tau=2), lty=3)

# Prepare parameters and data
init <- list(log_Linf=log(73), log_k=log(0.4), tau=2,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_had$age, Loto=otoliths_had$len)
gompertzo_objfun(init, dat)

# Fit model
model <- gompertzo(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
Lhat <- with(report, gompertzo_curve(x, Linf, k, tau))
lines(x, Lhat, lwd=2, col=2)
legend("bottomright", c("initial curve","model fit"), col=c(1,2), lty=c(3,1),
       lwd=c(1,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("Linf", "k", "tau", "sigma_min", "sigma_max")]
fit[-1]
summary(sdreport)

# Plot 95% prediction band
band <- pred_band(x, model)
areaplot::confplot(cbind(lower,upper)~age, band, xlim=c(0,18), ylim=c(0,100),
         ylab="len", col="mistyrose")
points(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,100),
       pch=16, col="#0080a010")
lines(x, Lhat, lwd=2, col=2)
lines(lower~age, band, lty=1, lwd=0.5, col=2)
lines(upper~age, band, lty=1, lwd=0.5, col=2)

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

# Model 2: Fit to skipjack otoliths and tags

# Explore initial parameter values
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
x <- seq(0, 4, 0.1)
points(lenRel~I(lenRel/60), tags_skj, col=4)
points(lenRec~I(lenRel/60+liberty), tags_skj, col=3)
```

```
lines(x, gompertzo_curve(x, Linf=75, k=1, tau=0), lty=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "initial curve"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,2),
       lwd=c(1.2,1.2,1.2,1), bty="n", inset=0.02, y.intersp=1.25)

# Prepare parameters and data
init <- list(log_Linf=log(75), log_k=log(1), tau=0,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len,
            Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty)
gompertzo_objfun(init, dat)

# Fit model
model <- gompertzo(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
points(report$age, report$Lrel, col=4)
points(report$age+report$liberty, report$Lrec, col=3)
Lhat <- with(report, gompertzo_curve(x, Linf, k, tau))
lines(x, Lhat, lwd=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "model fit"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,1),
       lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("Linf", "k", "tau", "sigma_min", "sigma_max")]
fit[-1]
head(summary(sdreport), 5)

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

# Model 3: Fit to skipjack otoliths only

init <- list(log_Linf=log(75), log_k=log(1), tau=0,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len)
model <- gompertzo(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
                  control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("Linf", "k", "tau", "sigma_min", "sigma_max")]

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

# Model 4: Fit to skipjack otoliths only,
# but now estimating constant sigma instead of sigma varying by length
# We do this by omitting log_sigma_max
init <- list(log_Linf=log(75), log_k=log(1), tau=0,
             log_sigma_min=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len)
model <- gompertzo(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("Linf", "k", "tau", "sigma_min")]

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

# Model 5: Fit to skipjack tags only

init <- list(log_Linf=log(75), log_k=log(1), tau=0,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty)
model <- gompertzo(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("Linf", "k", "tau", "sigma_min", "sigma_max")]
```

otoliths_had *Otolith Data (Haddock)*

Description

Otolith data for Icelandic haddock from the Icelandic spring bottom trawl survey 2011-2020.

Usage

```
otoliths_had
```

Format

Data frame containing two columns:

age	age (years)
len	length (cm)

Note

The data were contributed by the Icelandic Marine and Freshwater Research Institute. The otoliths were collected following the sampling protocol described in the survey manual (Sólmundsson et al. 2020).

Source

Sólmundsson, J., Karlsson, H., Björnsson, H., Jónsdóttir, I.G., Jakobsdóttir, K.B., and Bogason, V. (2020). *A manual for the Icelandic groundfish survey in spring 2020*. Marine and Freshwater Research in Iceland HV 2020-08.

See Also

`gcm`, `gompertz`, `gompertzo`, `richards`, `richardso`, `schnute3`, `vonbert`, and `vonberto` are alternative growth models.

`otoliths_had`, `otoliths_skj`, and `tags_skj` are example datasets.

`fishgrowth-package` gives an overview of the package.

Examples

```
head(otoliths_had)
```

---

otoliths_skj *Otolith Data (Skipjack)*

Description

Simulated otolith data, loosely based on a skipjack tuna dataset analyzed by Macdonald et al. (2022).

Usage

```
otoliths_skj
```

Format

Data frame containing two columns:

`age`	age (years)
`len`	length (cm)

Details

The simulation code that was used to produce this dataset is included in the package:

```
file.show(system.file(package="fishgrowth", "sim/simulate.R"))
```

Source

Macdonald, J., Day, J., Magnusson, A., Maunder, M., Aoki, Y., Matsubara, N., Tsuda, Y., McKechnie, S., Teears, T., Leroy, B., Castillo-Jordán, C., Hampton, J., and Hamer, P. (2022). *Review and new analyses of skipjack growth in the Western and Central Pacific Ocean*. Western and Central Pacific Fisheries Commission Report WCPFC-SC18-2022/SA-IP-06. https://meetings.wcpfc.int/node/16254.

See Also

gcm, gompertz, gompertzo, richards, richardso, schnute3, vonbert, and vonberto are alternative growth models.

otoliths_had, otoliths_skj, and tags_skj are example datasets.

fishgrowth-package gives an overview of the package.

Examples

```
otoliths_skj
```

pred_band *Prediction Band*

Description

Calculate a prediction band for a fitted growth curve.

Usage

```
pred_band(age, model, level = 0.95)
```

Arguments

age	a vector of ages to calculate the prediction band.
model	a fitted growth model.
level	significance level.

Value

A data frame containing five columns:

age	age
Lhat	predicted length
sigma	growth variability
lower	lower limit of prediction band
upper	upper limit of prediction band

Note

The variability of length at age (sigma) increases linearly with length:

$$\sigma_L \;=\; \alpha \,+\, \beta\hat{L}$$

This calculation of sigma is demonstrated in the example below.

The lower and upper limits of the prediction band are calculated as $\hat{L} \pm 1.96\sigma_L$ at the 95% significance level.

See Also

gcm, gompertz, gompertzo, richards, richardso, schnute3, vonbert, and vonberto are alternative growth models.

fishgrowth-package gives an overview of the package.

Examples

```
# Fit a model
init <- list(log_L1=log(20), log_L2=log(70), log_k=log(0.1),
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_had$age, Loto=otoliths_had$len, t1=1, t2=10)
model <- vonbert(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))

# Calculate 95% prediction band
x <- seq(1, 18, 0.5)
band <- pred_band(x, model)

# Plot 95% prediction band
areaplot::confplot(cbind(lower,upper)~age, band, xlim=c(0,18), ylim=c(0,100),
         ylab="len", col="mistyrose")
points(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,100),
       pch=16, col="#0080a010")
lines(Lhat~age, band, lwd=2, col=2)
lines(lower~age, band, lty=1, lwd=0.5, col=2)
lines(upper~age, band, lty=1, lwd=0.5, col=2)

# Calculate sigma by hand
report <- model$report()
alpha <- report$sigma_intercept
beta <- report$sigma_slope
Lhat <- band$Lhat
alpha + beta * Lhat  # same values as band$sigma calculated by pred_band()
```

richards *Richards Growth Model*

Description

Fit a Richards growth model to otoliths and/or tags, using the Schnute parametrization.

Usage

```
richards(par, data, silent = TRUE, ...)

richards_curve(t, L1, L2, k, b, t1, t2)

richards_objfun(par, data)
```

Arguments

`par`	a parameter list.
`data`	a data list.
`silent`	passed to `MakeADFun`.
`...`	passed to `MakeADFun`.
`t`	age (vector).
`L1`	predicted length at age `t1`.
`L2`	predicted length at age `t2`.
`k`	growth coefficient.
`b`	shape parameter.
`t1`	age where predicted length is `L1`.
`t2`	age where predicted length is `L2`.

Details

The main function `richards` creates a model object, ready for parameter estimation. The auxiliary functions `richards_curve` and `richards_objfun` are called by the main function to calculate the regression curve and objective function value. The user can also call the auxiliary functions directly for plotting and model exploration.

The `par` list contains the following elements:

- `log_L1`, predicted length at age `t1`
- `log_L2`, predicted length at age `t2`
- `log_k`, growth coefficient
- `b`, shape parameter
- `log_sigma_min`, growth variability at the shortest observed length in the data
- `log_sigma_max` (*), growth variability at the longest observed length in the data
- `log_age` (*), age at release of tagged individuals (vector)

*: The parameter `log_sigma_max` can be omitted to estimate growth variability that does not vary with length. The parameter vector `log_age` can be omitted to fit to otoliths only.

The `data` list contains the following elements:

- `Aoto` (*), age from otoliths (vector)
- `Loto` (*), length from otoliths (vector)
- `Lrel` (*), length at release of tagged individuals (vector)
- `Lrec` (*), length at recapture of tagged individuals (vector)
- `liberty` (*), time at liberty of tagged individuals in years (vector)
- `t1`, age where predicted length is `L1`
- `t2`, age where predicted length is `L2`

*: The data vectors `Aoto` and `Loto` can be omitted to fit to tagging data only. The data vectors `Lrel`, `Lrec`, and `liberty` can be omitted to fit to otoliths only.

Value

The richards function returns a TMB model object, produced by MakeADFun.

The richards_curve function returns a numeric vector of predicted length at age.

The richards_objfun function returns the negative log-likelihood as a single number, describing the goodness of fit of par to the data.

Note

The Schnute parametrization used in richards reduces parameter correlation and improves convergence reliability compared to the traditional parametrization used in richardso. Therefore, the richards parametrization can be recommended for general usage, as both parametrizations produce the same growth curve. However, there can be some use cases where the traditional parametrization (Linf, k, tau, b) is preferred over the Schnute parametrization (L1, L2, k, b).

The Richards (1959) growth model, as parametrized by Schnute (1981, Eq. 15), predicts length at age as:

$$\hat{L}_t \;=\; \left[\,L_1^b \;+\; (L_2^b - L_1^b)\,\frac{1-e^{-k(t-t_1)}}{1-e^{-k(t_2-t_1)}}\,\right]^{1/b}$$

The variability of length at age increases linearly with length,

$$\sigma_L \;=\; \alpha \,+\, \beta\hat{L}$$

where the slope is $\beta = (\sigma_{\max} - \sigma_{\min})/(L_{\max} - L_{\min})$, the intercept is $\alpha = \sigma_{\min} - \beta L_{\min}$, and $L_{\min}$ and $L_{\max}$ are the shortest and longest observed lengths in the data. Alternatively, growth variability can be modelled as a constant $\sigma_L = \sigma_{\min}$ that does not vary with length, by omitting log_sigma_max from the parameter list (see above).

The negative log-likelihood is calculated by comparing the observed and predicted lengths:

```
nll_Loto <- -dnorm(Loto, Loto_hat, sigma_Loto, TRUE)
nll_Lrel <- -dnorm(Lrel, Lrel_hat, sigma_Lrel, TRUE)
nll_Lrec <- -dnorm(Lrec, Lrec_hat, sigma_Lrec, TRUE)
nll <- sum(nll_Loto) + sum(nll_Lrel) + sum(nll_Lrec)
```

References

Richards, F.J. (1959). A flexible growth function for empirical use. *Journal of Experimental Botany*, **10**, 290-300. https://www.jstor.org/stable/23686557.

Schnute, J. (1981). A versatile growth model with statistically stable parameters. *Canadian Journal of Fisheries and Aquatic Science*, **38**, 1128-1140. doi:10.1139/f81153.

The fishgrowth-package help page includes references describing the parameter estimation method.

See Also

gcm, gompertz, gompertzo, richards, richardso, schnute3, vonbert, and vonberto are alternative growth models.

pred_band calculates a prediction band for a fitted growth model.

otoliths_had, otoliths_skj, and tags_skj are example datasets.

fishgrowth-package gives an overview of the package.

Examples

```
# Model 1: Fit to haddock otoliths

# Explore initial parameter values
plot(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,105), pch=16,
     col="#0080a010")
x <- seq(1, 18, 0.1)
lines(x, richards_curve(x, L1=18, L2=67, k=0.1, b=1, t1=1, t2=10), lty=3)

# Prepare parameters and data
init <- list(log_L1=log(18), log_L2=log(67), log_k=log(0.1), b=1,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_had$age, Loto=otoliths_had$len, t1=1, t2=10)
richards_objfun(init, dat)

# Fit model
model <- richards(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
Lhat <- with(report, richards_curve(x, L1, L2, k, b, t1, t2))
lines(x, Lhat, lwd=2, col=2)
legend("bottomright", c("initial curve","model fit"), col=c(1,2), lty=c(3,1),
       lwd=c(1,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L1", "L2", "k", "b", "sigma_min", "sigma_max")]
fit[-1]
summary(sdreport)

# Plot 95% prediction band
band <- pred_band(x, model)
areaplot::confplot(cbind(lower,upper)~age, band, xlim=c(0,18), ylim=c(0,100),
         ylab="len", col="mistyrose")
points(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,100),
       pch=16, col="#0080a010")
lines(x, Lhat, lwd=2, col=2)
lines(lower~age, band, lty=1, lwd=0.5, col=2)
lines(upper~age, band, lty=1, lwd=0.5, col=2)
```

```
#############################################################################

# Model 2: Fit to skipjack otoliths and tags

# Explore initial parameter values
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
x <- seq(0, 4, 0.1)
points(lenRel~I(lenRel/60), tags_skj, col=4)
points(lenRec~I(lenRel/60+liberty), tags_skj, col=3)
lines(x, richards_curve(x, L1=25, L2=75, k=0.8, b=1, t1=0, t2=4), lty=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "initial curve"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,2),
       lwd=c(1.2,1.2,1.2,1), bty="n", inset=0.02, y.intersp=1.25)

# Prepare parameters and data
init <- list(log_L1=log(25), log_L2=log(75), log_k=log(0.8), b=1,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len,
            Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty, t1=0, t2=4)
richards_objfun(init, dat)

# Fit model
model <- richards(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
points(report$age, report$Lrel, col=4)
points(report$age+report$liberty, report$Lrec, col=3)
Lhat <- with(report, richards_curve(x, L1, L2, k, b, t1, t2))
lines(x, Lhat, lwd=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "model fit"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,1),
       lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L1", "L2", "k", "b", "sigma_min", "sigma_max")]
fit[-1]
head(summary(sdreport), 6)

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

# Model 3: Fit to skipjack otoliths only

init <- list(log_L1=log(25), log_L2=log(75), log_k=log(0.8), b=1,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len, t1=0, t2=4)
model <- richards(init, dat)
```

```
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "k", "b", "sigma_min", "sigma_max")]

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

# Model 4: Fit to skipjack otoliths only,
# but now estimating constant sigma instead of sigma varying by length

# We do this by omitting log_sigma_max
init <- list(log_L1=log(25), log_L2=log(75), log_k=log(0.8), b=1,
             log_sigma_min=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len, t1=0, t2=4)
model <- richards(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "k", "b", "sigma_min")]

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

# Model 5: Fit to skipjack tags only

init <- list(log_L1=log(25), log_L2=log(75), log_k=log(0.8), b=1,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty, t1=0, t2=4)
model <- richards(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "k", "b", "sigma_min", "sigma_max")]
```

richardso *Richards Growth Model (Old Style)*

Description

Fit a Richards growth model to otoliths and/or tags, using a traditional parametrization.

Usage

```
richardso(par, data, silent = TRUE, ...)

richardso_curve(t, Linf, k, tau, b)

richardso_objfun(par, data)
```

Arguments

`par`	a parameter list.
`data`	a data list.
`silent`	passed to `MakeADFun`.
`...`	passed to `MakeADFun`.
`t`	age (vector).
`Linf`	asymptotic maximum length.
`k`	growth coefficient.
`tau`	location parameter.
`b`	shape parameter.

Details

The main function `richardso` creates a model object, ready for parameter estimation. The auxiliary functions `richardso_curve` and `richardso_objfun` are called by the main function to calculate the regression curve and objective function value. The user can also call the auxiliary functions directly for plotting and model exploration.

The `par` list contains the following elements:

- `log_Linf`, asymptotic maximum length
- `log_k`, growth coefficient
- `tau`, location parameter
- `b`, shape parameter
- `log_sigma_min`, growth variability at the shortest observed length in the data
- `log_sigma_max` (*), growth variability at the longest observed length in the data
- `log_age` (*), age at release of tagged individuals (vector)

*: The parameter `log_sigma_max` can be omitted to estimate growth variability that does not vary with length. The parameter vector `log_age` can be omitted to fit to otoliths only.

The `data` list contains the following elements:

- `Aoto` (*), age from otoliths (vector)
- `Loto` (*), length from otoliths (vector)
- `Lrel` (*), length at release of tagged individuals (vector)
- `Lrec` (*), length at recapture of tagged individuals (vector)
- `liberty` (*), time at liberty of tagged individuals in years (vector)

*: The data vectors `Aoto` and `Loto` can be omitted to fit to tagging data only. The data vectors `Lrel`, `Lrec`, and `liberty` can be omitted to fit to otoliths only.

Value

The richardso function returns a TMB model object, produced by MakeADFun.

The richardso_curve function returns a numeric vector of predicted length at age.

The richardso_objfun function returns the negative log-likelihood as a single number, describing the goodness of fit of par to the data.

Note

The Schnute parametrization used in richards reduces parameter correlation and improves convergence reliability compared to the traditional parametrization used in richardso. Therefore, the richards parametrization can be recommended for general usage, as both parametrizations produce the same growth curve. However, there can be some use cases where the traditional parametrization (Linf, k, tau, b) is preferred over the Schnute parametrization (L1, L2, k, b).

The Richards (1959) growth model, as parametrized by Tjørve and Tjørve (2010, Eq. 4), predicts length at age as:

$$\hat{L}_t \;=\; L_\infty\left(1\,-\,\frac{1}{b}\,e^{-k(t-\tau)}\right)^b$$

The variability of length at age increases linearly with length,

$$\sigma_L \;=\; \alpha \,+\, \beta\hat{L}$$

where the slope is $\beta = (\sigma_{\max} - \sigma_{\min})/(L_{\max} - L_{\min})$, the intercept is $\alpha = \sigma_{\min} - \beta L_{\min}$, and $L_{\min}$ and $L_{\max}$ are the shortest and longest observed lengths in the data. Alternatively, growth variability can be modelled as a constant $\sigma_L = \sigma_{\min}$ that does not vary with length, by omitting log_sigma_max from the parameter list (see above).

The negative log-likelihood is calculated by comparing the observed and predicted lengths:

```
nll_Loto <- -dnorm(Loto, Loto_hat, sigma_Loto, TRUE)
nll_Lrel <- -dnorm(Lrel, Lrel_hat, sigma_Lrel, TRUE)
nll_Lrec <- -dnorm(Lrec, Lrec_hat, sigma_Lrec, TRUE)
nll <- sum(nll_Loto) + sum(nll_Lrel) + sum(nll_Lrec)
```

References

Richards, F.J. (1959). A flexible growth function for empirical use. *Journal of Experimental Botany*, **10**, 290-300. https://www.jstor.org/stable/23686557.

Tjørve, E. and Tjørve, K.M.C. (2010). A unified approach to the Richards-model family for use in growth analyses: Why we need only two model forms. *Journal of Theoretical Biology*, **267**, 417-425.

The fishgrowth-package help page includes references describing the parameter estimation method.

See Also

gcm, gompertz, gompertzo, richards, richardso, schnute3, vonbert, and vonberto are alternative growth models.

pred_band calculates a prediction band for a fitted growth model.

otoliths_had, otoliths_skj, and tags_skj are example datasets.

fishgrowth-package gives an overview of the package.

Examples

```
# Model 1: Fit to haddock otoliths

# Explore initial parameter values
plot(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,105), pch=16,
     col="#0080a010")
x <- seq(1, 18, 0.1)
lines(x, richardso_curve(x, Linf=100, k=0.1, tau=-1, b=1), lty=3)

# Prepare parameters and data
init <- list(log_Linf=log(100), log_k=log(0.1), tau=-1, b=1,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_had$age, Loto=otoliths_had$len)
richardso_objfun(init, dat)

# Fit model
model <- richardso(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
Lhat <- with(report, richardso_curve(x, Linf, k, tau, b))
lines(x, Lhat, lwd=2, col=2)
legend("bottomright", c("initial curve","model fit"), col=c(1,2), lty=c(3,1),
       lwd=c(1,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("Linf", "k", "tau", "b", "sigma_min", "sigma_max")]
fit[-1]
summary(sdreport)

# Plot 95% prediction band
band <- pred_band(x, model)
areaplot::confplot(cbind(lower,upper)~age, band, xlim=c(0,18), ylim=c(0,100),
         ylab="len", col="mistyrose")
points(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,100),
       pch=16, col="#0080a010")
lines(x, Lhat, lwd=2, col=2)
lines(lower~age, band, lty=1, lwd=0.5, col=2)
lines(upper~age, band, lty=1, lwd=0.5, col=2)
```

```
###############################################################################

# Model 2: Fit to skipjack otoliths and tags

# Explore initial parameter values
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
x <- seq(0, 4, 0.1)
points(lenRel~I(lenRel/60), tags_skj, col=4)
points(lenRec~I(lenRel/60+liberty), tags_skj, col=3)
lines(x, richardso_curve(x, Linf=80, k=0.8, tau=-0.5, b=1), lty=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "initial curve"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,2),
       lwd=c(1.2,1.2,1.2,1), bty="n", inset=0.02, y.intersp=1.25)

# Prepare parameters and data
init <- list(log_Linf=log(80), log_k=log(0.8), tau=-0.5, b=1,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len,
            Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty)
richardso_objfun(init, dat)

# Fit model
model <- richardso(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
points(report$age, report$Lrel, col=4)
points(report$age+report$liberty, report$Lrec, col=3)
Lhat <- with(report, richardso_curve(x, Linf, k, tau, b))
lines(x, Lhat, lwd=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "model fit"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,1),
       lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("Linf", "k", "tau", "b", "sigma_min", "sigma_max")]
fit[-1]
head(summary(sdreport), 6)

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

# Model 3: Fit to skipjack otoliths only

init <- list(log_Linf=log(80), log_k=log(0.8), tau=-0.5, b=1,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len)
model <- richardso(init, dat)
```

```
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("Linf", "k", "tau", "b", "sigma_min", "sigma_max")]

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

# Model 4: Fit to skipjack otoliths only,
# but now estimating constant sigma instead of sigma varying by length

# We do this by omitting log_sigma_max
init <- list(log_Linf=log(80), log_k=log(0.8), tau=-0.5, b=1,
             log_sigma_min=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len)
model <- richardso(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("Linf", "k", "tau", "b", "sigma_min")]

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

# Model 5: Fit to skipjack tags only

init <- list(log_Linf=log(80), log_k=log(0.8), tau=-0.5, b=1,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty)
model <- richardso(init, dat)           # using 1e3 to keep CRAN checks fast,
fit <- nlminb(model$par, model$fn, model$gr,             # but try 1e4 to get
              control=list(eval.max=1e3, iter.max=1e3))  # better convergence
model$report()[c("Linf", "k", "tau", "b", "sigma_min", "sigma_max")]
```

`schnute3` *Schnute Case 3 Model*

Description

Fit a Schnute Case 3 model to otoliths and/or tags.

Usage

```
schnute3(par, data, silent = TRUE, ...)

schnute3_curve(t, L1, L2, b, t1, t2)

schnute3_objfun(par, data)
```

Arguments

`par`	a parameter list.
`data`	a data list.
`silent`	passed to `MakeADFun`.
`...`	passed to `MakeADFun`.
`t`	age (vector).
`L1`	predicted length at age `t1`.
`L2`	predicted length at age `t2`.
`b`	shape parameter.
`t1`	age where predicted length is `L1`.
`t2`	age where predicted length is `L2`.

Details

The main function `schnute3` creates a model object, ready for parameter estimation. The auxiliary functions `schnute3_curve` and `schnute3_objfun` are called by the main function to calculate the regression curve and objective function value. The user can also call the auxiliary functions directly for plotting and model exploration.

The `par` list contains the following elements:

- `log_L1`, predicted length at age `t1`
- `log_L2`, predicted length at age `t2`
- `b`, shape parameter
- `log_sigma_min`, growth variability at the shortest observed length in the data
- `log_sigma_max` (*), growth variability at the longest observed length in the data
- `log_age` (*), age at release of tagged individuals (vector)

*: The parameter `log_sigma_max` can be omitted to estimate growth variability that does not vary with length. The parameter vector `log_age` can be omitted to fit to otoliths only.

The `data` list contains the following elements:

- `Aoto` (*), age from otoliths (vector)
- `Loto` (*), length from otoliths (vector)
- `Lrel` (*), length at release of tagged individuals (vector)
- `Lrec` (*), length at recapture of tagged individuals (vector)
- `liberty` (*), time at liberty of tagged individuals in years (vector)
- `t1`, age where predicted length is `L1`
- `t2`, age where predicted length is `L2`

*: The data vectors `Aoto` and `Loto` can be omitted to fit to tagging data only. The data vectors `Lrel`, `Lrec`, and `liberty` can be omitted to fit to otoliths only.

Value

The schnute3 function returns a TMB model object, produced by MakeADFun.

The schnute3_curve function returns a numeric vector of predicted length at age.

The schnute3_objfun function returns the negative log-likelihood as a single number, describing the goodness of fit of par to the data.

Note

The Schnute Case 3 model is a special case of the Richards (1959) model, where $k = 0$. If the best model fit of a richards model to a particular dataset involves a very small estimated value of k, then the schnute3 model offers a preferable parametrization, as it produces the same curve using fewer parameters.

The Schnute Case 3 model (Schnute 1981, Eq. 17) predicts length at age as:

$$\hat{L}_t \;=\; \left[\, L_1^b \;+\; (L_2^b - L_1^b)\,\frac{t - t_1}{t_2 - t_1} \right]^{1/b}$$

The variability of length at age increases linearly with length,

$$\sigma_L \;=\; \alpha \,+\, \beta \hat{L}$$

where the slope is $\beta = (\sigma_{\max} - \sigma_{\min})/(L_{\max} - L_{\min})$, the intercept is $\alpha = \sigma_{\min} - \beta L_{\min}$, and $L_{\min}$ and $L_{\max}$ are the shortest and longest observed lengths in the data. Alternatively, growth variability can be modelled as a constant $\sigma_L = \sigma_{\min}$ that does not vary with length, by omitting log_sigma_max from the parameter list (see above).

The negative log-likelihood is calculated by comparing the observed and predicted lengths:

```
nll_Loto <- -dnorm(Loto, Loto_hat, sigma_Loto, TRUE)
nll_Lrel <- -dnorm(Lrel, Lrel_hat, sigma_Lrel, TRUE)
nll_Lrec <- -dnorm(Lrec, Lrec_hat, sigma_Lrec, TRUE)
nll <- sum(nll_Loto) + sum(nll_Lrel) + sum(nll_Lrec)
```

References

Richards, F.J. (1959). A flexible growth function for empirical use. *Journal of Experimental Botany*, **10**, 290-300. https://www.jstor.org/stable/23686557.

Schnute, J. (1981). A versatile growth model with statistically stable parameters. *Canadian Journal of Fisheries and Aquatic Science*, **38**, 1128-1140. doi:10.1139/f81153.

The fishgrowth-package help page includes references describing the parameter estimation method.

See Also

gcm, gompertz, gompertzo, richards, richardso, schnute3, vonbert, and vonberto are alternative growth models.

pred_band calculates a prediction band for a fitted growth model.

otoliths_had, otoliths_skj, and tags_skj are example datasets.

fishgrowth-package gives an overview of the package.

Examples

```
# Model 1: Fit to haddock otoliths

# Explore initial parameter values
plot(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,105), pch=16,
     col="#0080a010")
x <- seq(1, 18, 0.1)
lines(x, schnute3_curve(x, L1=15, L2=70, b=2, t1=1, t2=10), lty=3)

# Prepare parameters and data
init <- list(log_L1=log(15), log_L2=log(70), b=2,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_had$age, Loto=otoliths_had$len, t1=1, t2=10)
schnute3_objfun(init, dat)

# Fit model
model <- schnute3(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
Lhat <- with(report, schnute3_curve(x, L1, L2, b, t1, t2))
lines(x, Lhat, lwd=2, col=2)
legend("bottomright", c("initial curve","model fit"), col=c(1,2), lty=c(3,1),
       lwd=c(1,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L1", "L2", "b", "sigma_min", "sigma_max")]
fit[-1]
summary(sdreport)

# Plot 95% prediction band
band <- pred_band(x, model)
areaplot::confplot(cbind(lower,upper)~age, band, xlim=c(0,18), ylim=c(0,100),
         ylab="len", col="mistyrose")
points(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,100),
       pch=16, col="#0080a010")
lines(x, Lhat, lwd=2, col=2)
lines(lower~age, band, lty=1, lwd=0.5, col=2)
lines(upper~age, band, lty=1, lwd=0.5, col=2)

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

# Model 2: Fit to skipjack otoliths and tags

# Explore initial parameter values
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
x <- seq(0, 4, 0.1)
points(lenRel~I(lenRel/60), tags_skj, col=4)
points(lenRec~I(lenRel/60+liberty), tags_skj, col=3)
```

```
lines(x, schnute3_curve(x, L1=25, L2=75, b=3, t1=0, t2=4), lty=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "initial curve"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,2),
       lwd=c(1.2,1.2,1.2,1), bty="n", inset=0.02, y.intersp=1.25)

# Prepare parameters and data
init <- list(log_L1=log(25), log_L2=log(75), b=3,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len,
            Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty, t1=0, t2=4)
schnute3_objfun(init, dat)

# Fit model
model <- schnute3(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
points(report$age, report$Lrel, col=4)
points(report$age+report$liberty, report$Lrec, col=3)
Lhat <- with(report, schnute3_curve(x, L1, L2, b, t1, t2))
lines(x, Lhat, lwd=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "model fit"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,1),
       lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L1", "L2", "b", "sigma_min", "sigma_max")]
fit[-1]
head(summary(sdreport), 5)

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

# Model 3: Fit to skipjack otoliths only

init <- list(log_L1=log(25), log_L2=log(75), b=3,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len, t1=0, t2=4)
model <- schnute3(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "b", "sigma_min", "sigma_max")]

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

# Model 4: Fit to skipjack otoliths only,
# but now estimating constant sigma instead of sigma varying by length
# We do this by omitting log_sigma_max
init <- list(log_L1=log(25), log_L2=log(75), b=3,
             log_sigma_min=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len, t1=0, t2=4)
model <- schnute3(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "b", "sigma_min")]

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

# Model 5: Fit to skipjack tags only

init <- list(log_L1=log(25), log_L2=log(75), b=3,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty, t1=0, t2=4)
model <- schnute3(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "b", "sigma_min", "sigma_max")]
```

tags_skj *Tagging Data (Skipjack)*

Description

Simulated tagging data, loosely based on a skipjack tuna dataset analyzed by Macdonald et al. (2022).

Usage

```
tags_skj
```

Format

Data frame containing three columns:

lenRel	length at release (cm)
lenRec	length at recapture (cm)
liberty	time at liberty (years)

Details

The simulation code that was used to produce this dataset is included in the package:

```
file.show(system.file(package="fishgrowth", "sim/simulate.R"))
```

Source

Macdonald, J., Day, J., Magnusson, A., Maunder, M., Aoki, Y., Matsubara, N., Tsuda, Y., McKechnie, S., Teears, T., Leroy, B., Castillo-Jordán, C., Hampton, J., and Hamer, P. (2022). *Review and new analyses of skipjack growth in the Western and Central Pacific Ocean*. Western and Central Pacific Fisheries Commission Report WCPFC-SC18-2022/SA-IP-06. https://meetings.wcpfc.int/node/16254.

See Also

gcm, gompertz, gompertzo, richards, richardso, schnute3, vonbert, and vonberto are alternative growth models.

otoliths_had, otoliths_skj, and tags_skj are example datasets.

fishgrowth-package gives an overview of the package.

Examples

```
head(tags_skj)
```

vonbert *Von Bertalanffy Growth Model*

Description

Fit a von Bertalanffy growth model to otoliths and/or tags, using the Schnute-Fournier parametrization.

Usage

```
vonbert(par, data, silent = TRUE, ...)

vonbert_curve(t, L1, L2, k, t1, t2)

vonbert_objfun(par, data)
```

Arguments

par	a parameter list.
data	a data list.
silent	passed to MakeADFun.
...	passed to MakeADFun.
t	age (vector).
L1	predicted length at age t1.
L2	predicted length at age t2.
k	growth coefficient.
t1	age where predicted length is L1.
t2	age where predicted length is L2.

Details

The main function vonbert creates a model object, ready for parameter estimation. The auxiliary functions vonbert_curve and vonbert_objfun are called by the main function to calculate the regression curve and objective function value. The user can also call the auxiliary functions directly for plotting and model exploration.

The par list contains the following elements:

- log_L1, predicted length at age t1
- log_L2, predicted length at age t2
- log_k, growth coefficient
- log_sigma_min, growth variability at the shortest observed length in the data
- log_sigma_max (*), growth variability at the longest observed length in the data
- log_age (*), age at release of tagged individuals (vector)

*: The parameter log_sigma_max can be omitted to estimate growth variability that does not vary with length. The parameter vector log_age can be omitted to fit to otoliths only.

The data list contains the following elements:

- Aoto (*), age from otoliths (vector)
- Loto (*), length from otoliths (vector)
- Lrel (*), length at release of tagged individuals (vector)
- Lrec (*), length at recapture of tagged individuals (vector)
- liberty (*), time at liberty of tagged individuals in years (vector)
- t1, age where predicted length is L1
- t2, age where predicted length is L2

*: The data vectors Aoto and Loto can be omitted to fit to tagging data only. The data vectors Lrel, Lrec, and liberty can be omitted to fit to otoliths only.

Value

The vonbert function returns a TMB model object, produced by MakeADFun.

The vonbert_curve function returns a numeric vector of predicted length at age.

The vonbert_objfun function returns the negative log-likelihood as a single number, describing the goodness of fit of par to the data.

Note

The Schnute-Fournier parametrization used in vonbert reduces parameter correlation and improves convergence reliability compared to the traditional parametrization used in vonberto. Therefore, the vonbert parametrization can be recommended for general usage, as both parametrizations produce the same growth curve. However, there can be some use cases where the traditional parametrization (Linf, k, t0) is preferred over the Schnute-Fournier parametrization (L1, L2, k).

The von Bertalanffy (1938) growth model, as parametrized by Schnute and Fournier (1980), predicts length at age as:

$$\hat{L}_t \;=\; L_1 \;+\; (L_2 - L_1)\,\frac{1 \;-\; e^{-k(t-t_1)}}{1 \;-\; e^{-k(t_2-t_1)}}$$

The variability of length at age increases linearly with length,

$$\sigma_L \;=\; \alpha \;+\; \beta \hat{L}$$

where the slope is $\beta = (\sigma_{\max} - \sigma_{\min})/(L_{\max} - L_{\min})$, the intercept is $\alpha = \sigma_{\min} - \beta L_{\min}$, and $L_{\min}$ and $L_{\max}$ are the shortest and longest observed lengths in the data. Alternatively, growth variability can be modelled as a constant $\sigma_L \;=\; \sigma_{\min}$ that does not vary with length, by omitting `log_sigma_max` from the parameter list (see above).

The negative log-likelihood is calculated by comparing the observed and predicted lengths:

```
nll_Loto <- -dnorm(Loto, Loto_hat, sigma_Loto, TRUE)
nll_Lrel <- -dnorm(Lrel, Lrel_hat, sigma_Lrel, TRUE)
nll_Lrec <- -dnorm(Lrec, Lrec_hat, sigma_Lrec, TRUE)
nll <- sum(nll_Loto) + sum(nll_Lrel) + sum(nll_Lrec)
```

References

von Bertalanffy, L. (1938). A quantitative theory of organic growth. *Human Biology*, **10**, 181-213. https://www.jstor.org/stable/41447359.

Schnute, J. and Fournier, D. (1980). A new approach to length-frequency analysis: Growth structure. *Canadian Journal of Fisheries and Aquatic Science*, **37**, 1337-1351. doi:10.1139/f80172.

The `fishgrowth-package` help page includes references describing the parameter estimation method.

See Also

`gcm`, `gompertz`, `gompertzo`, `richards`, `richardso`, `schnute3`, `vonbert`, and `vonberto` are alternative growth models.

`pred_band` calculates a prediction band for a fitted growth model.

`otoliths_had`, `otoliths_skj`, and `tags_skj` are example datasets.

`fishgrowth-package` gives an overview of the package.

Examples

```
# Model 1: Fit to haddock otoliths

# Explore initial parameter values
plot(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,105), pch=16,
     col="#0080a010")
x <- seq(1, 18, 0.1)
lines(x, vonbert_curve(x, L1=18, L2=67, k=0.1, t1=1, t2=10), lty=3)

# Prepare parameters and data
init <- list(log_L1=log(18), log_L2=log(67), log_k=log(0.1),
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_had$age, Loto=otoliths_had$len, t1=1, t2=10)
```

```
vonbert_objfun(init, dat)

# Fit model
model <- vonbert(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
Lhat <- with(report, vonbert_curve(x, L1, L2, k, t1, t2))
lines(x, Lhat, lwd=2, col=2)
legend("bottomright", c("initial curve","model fit"), col=c(1,2), lty=c(3,1),
       lwd=c(1,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L1", "L2", "k", "sigma_min", "sigma_max")]
fit[-1]
summary(sdreport)

# Plot 95% prediction band
band <- pred_band(x, model)
areaplot::confplot(cbind(lower,upper)~age, band, xlim=c(0,18), ylim=c(0,100),
         ylab="len", col="mistyrose")
points(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,100),
       pch=16, col="#0080a010")
lines(x, Lhat, lwd=2, col=2)
lines(lower~age, band, lty=1, lwd=0.5, col=2)
lines(upper~age, band, lty=1, lwd=0.5, col=2)

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

# Model 2: Fit to skipjack otoliths and tags

# Explore initial parameter values
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
x <- seq(0, 4, 0.1)
points(lenRel~I(lenRel/60), tags_skj, col=4)
points(lenRec~I(lenRel/60+liberty), tags_skj, col=3)
lines(x, vonbert_curve(x, L1=25, L2=75, k=0.8, t1=0, t2=4), lty=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "initial curve"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,2),
       lwd=c(1.2,1.2,1.2,1), bty="n", inset=0.02, y.intersp=1.25)

# Prepare parameters and data
init <- list(log_L1=log(25), log_L2=log(75), log_k=log(0.8),
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len,
            Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty, t1=0, t2=4)
vonbert_objfun(init, dat)
```

```
# Fit model
model <- vonbert(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
points(report$age, report$Lrel, col=4)
points(report$age+report$liberty, report$Lrec, col=3)
Lhat <- with(report, vonbert_curve(x, L1, L2, k, t1, t2))
lines(x, Lhat, lwd=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "model fit"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,1),
       lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("L1", "L2", "k", "sigma_min", "sigma_max")]
fit[-1]
head(summary(sdreport), 5)

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

# Model 3: Fit to skipjack otoliths only

init <- list(log_L1=log(25), log_L2=log(75), log_k=log(0.8),
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len, t1=0, t2=4)
model <- vonbert(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "k", "sigma_min", "sigma_max")]

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

# Model 4: Fit to skipjack otoliths only,
# but now estimating constant sigma instead of sigma varying by length

# We do this by omitting log_sigma_max
init <- list(log_L1=log(25), log_L2=log(75), log_k=log(0.8),
             log_sigma_min=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len, t1=0, t2=4)
model <- vonbert(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "k", "sigma_min")]

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

# Model 5: Fit to skipjack tags only

init <- list(log_L1=log(25), log_L2=log(75), log_k=log(0.8),
              log_sigma_min=log(3), log_sigma_max=log(3),
              log_age=log(tags_skj$lenRel/60))
dat <- list(Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty, t1=0, t2=4)
model <- vonbert(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
               control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("L1", "L2", "k", "sigma_min", "sigma_max")]
```

vonberto *Von Bertalanffy Growth Model (Old Style)*

Description

Fit a von Bertalanffy growth model to otoliths and/or tags, using a traditional parametrization.

Usage

```
vonberto(par, data, silent = TRUE, ...)

vonberto_curve(t, Linf, k, t0)

vonberto_objfun(par, data)
```

Arguments

par	a parameter list.
data	a data list.
silent	passed to MakeADFun.
...	passed to MakeADFun.
t	age (vector).
Linf	asymptotic maximum length.
k	growth coefficient.
t0	age where the predicted length is zero, the x-intercept.

Details

The main function vonberto creates a model object, ready for parameter estimation. The auxiliary functions vonberto_curve and vonberto_objfun are called by the main function to calculate the regression curve and objective function value. The user can also call the auxiliary functions directly for plotting and model exploration.

The par list contains the following elements:

- log_Linf, asymptotic maximum length
- log_k, growth coefficient

- to, age where the predicted length is zero, the x-intercept
- log_sigma_min, growth variability at the shortest observed length in the data
- log_sigma_max (*), growth variability at the longest observed length in the data
- log_age (*), age at release of tagged individuals (vector)

*: The parameter log_sigma_max can be omitted to estimate growth variability that does not vary with length. The parameter vector log_age can be omitted to fit to otoliths only.

The data list contains the following elements:

- Aoto (*), age from otoliths (vector)
- Loto (*), length from otoliths (vector)
- Lrel (*), length at release of tagged individuals (vector)
- Lrec (*), length at recapture of tagged individuals (vector)
- liberty (*), time at liberty of tagged individuals in years (vector)

*: The data vectors Aoto and Loto can be omitted to fit to tagging data only. The data vectors Lrel, Lrec, and liberty can be omitted to fit to otoliths only.

Value

The vonberto function returns a TMB model object, produced by MakeADFun.

The vonberto_curve function returns a numeric vector of predicted length at age.

The vonberto_objfun function returns the negative log-likelihood as a single number, describing the goodness of fit of par to the data.

Note

The Schnute-Fournier parametrization used in vonbert reduces parameter correlation and improves convergence reliability compared to the traditional parametrization used in vonberto. Therefore, the vonbert parametrization can be recommended for general usage, as both parametrizations produce the same growth curve. However, there can be some use cases where the traditional parametrization (Linf, k, t0) is preferred over the Schnute-Fournier parametrization (L1, L2, k).

The von Bertalanffy (1938) growth model, as parametrized by Beverton and Holt (1957), predicts length at age as:

$$\hat{L}_t \;=\; L_\infty\left(1\,-\,e^{-k(t-t_0)}\right)$$

The variability of length at age increases linearly with length,

$$\sigma_L \;=\; \alpha\,+\,\beta\hat{L}$$

where the slope is $\beta = (\sigma_{\max} - \sigma_{\min})/(L_{\max} - L_{\min})$, the intercept is $\alpha = \sigma_{\min} - \beta L_{\min}$, and $L_{\min}$ and $L_{\max}$ are the shortest and longest observed lengths in the data. Alternatively, growth variability can be modelled as a constant $\sigma_L = \sigma_{\min}$ that does not vary with length, by omitting log_sigma_max from the parameter list (see above).

The negative log-likelihood is calculated by comparing the observed and predicted lengths:

```
  nll_Loto <- -dnorm(Loto, Loto_hat, sigma_Loto, TRUE)
  nll_Lrel <- -dnorm(Lrel, Lrel_hat, sigma_Lrel, TRUE)
  nll_Lrec <- -dnorm(Lrec, Lrec_hat, sigma_Lrec, TRUE)
  nll <- sum(nll_Loto) + sum(nll_Lrel) + sum(nll_Lrec)
```

References

von Bertalanffy, L. (1938). A quantitative theory of organic growth. *Human Biology*, **10**, 181-213. https://www.jstor.org/stable/41447359.

Beverton, R.J.H. and Holt, S.J. (1957). *On the dynamics of exploited fish populations*. London: Her Majesty's Stationery Office.

The fishgrowth-package help page includes references describing the parameter estimation method.

See Also

gcm, gompertz, gompertzo, richards, richards, schnute3, vonbert, and vonberto are alternative growth models.

pred_band calculates a prediction band for a fitted growth model.

otoliths_had, otoliths_skj, and tags_skj are example datasets.

fishgrowth-package gives an overview of the package.

Examples

```
# Model 1: Fit to haddock otoliths

# Explore initial parameter values
plot(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,105), pch=16,
     col="#0080a010")
x <- seq(1, 18, 0.1)
lines(x, vonberto_curve(x, Linf=100, k=0.1, t0=-1), lty=3)

# Prepare parameters and data
init <- list(log_Linf=log(100), log_k=log(0.1), t0=-1,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_had$age, Loto=otoliths_had$len)
vonberto_objfun(init, dat)

# Fit model
model <- vonberto(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
Lhat <- with(report, vonberto_curve(x, Linf, k, t0))
lines(x, Lhat, lwd=2, col=2)
legend("bottomright", c("initial curve","model fit"), col=c(1,2), lty=c(3,1),
       lwd=c(1,2), bty="n", inset=0.02, y.intersp=1.25)
```

```
# Model summary
report[c("Linf", "k", "t0", "sigma_min", "sigma_max")]
fit[-1]
summary(sdreport)

# Plot 95% prediction band
band <- pred_band(x, model)
areaplot::confplot(cbind(lower,upper)~age, band, xlim=c(0,18), ylim=c(0,100),
         ylab="len", col="mistyrose")
points(len~age, otoliths_had, xlim=c(0,18), ylim=c(0,100),
       pch=16, col="#0080a010")
lines(x, Lhat, lwd=2, col=2)
lines(lower~age, band, lty=1, lwd=0.5, col=2)
lines(upper~age, band, lty=1, lwd=0.5, col=2)

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

# Model 2: Fit to skipjack otoliths and tags

# Explore initial parameter values
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
x <- seq(0, 4, 0.1)
points(lenRel~I(lenRel/60), tags_skj, col=4)
points(lenRec~I(lenRel/60+liberty), tags_skj, col=3)
lines(x, vonberto_curve(x, Linf=80, k=0.8, t0=-0.5), lty=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "initial curve"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,2),
       lwd=c(1.2,1.2,1.2,1), bty="n", inset=0.02, y.intersp=1.25)

# Prepare parameters and data
init <- list(log_Linf=log(80), log_k=log(0.8), t0=-0.5,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len,
            Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty)
vonberto_objfun(init, dat)

# Fit model
model <- vonberto(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
report <- model$report()
sdreport <- sdreport(model)

# Plot results
plot(len~age, otoliths_skj, xlim=c(0,4), ylim=c(0,100))
points(report$age, report$Lrel, col=4)
points(report$age+report$liberty, report$Lrec, col=3)
Lhat <- with(report, vonberto_curve(x, Linf, k, t0))
lines(x, Lhat, lwd=2)
legend("bottomright", c("otoliths","tag releases","tac recaptures",
       "model fit"), col=c(1,4,3,1), pch=c(1,1,1,NA), lty=c(0,0,0,1),
       lwd=c(1.2,1.2,1.2,2), bty="n", inset=0.02, y.intersp=1.25)

# Model summary
report[c("Linf", "k", "t0", "sigma_min", "sigma_max")]
fit[-1]
head(summary(sdreport), 5)

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

# Model 3: Fit to skipjack otoliths only

init <- list(log_Linf=log(80), log_k=log(0.8), t0=-0.5,
             log_sigma_min=log(3), log_sigma_max=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len)
model <- vonberto(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("Linf", "k", "t0", "sigma_min", "sigma_max")]

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

# Model 4: Fit to skipjack otoliths only,
# but now estimating constant sigma instead of sigma varying by length

# We do this by omitting log_sigma_max
init <- list(log_Linf=log(80), log_k=log(0.8), t0=-0.5,
             log_sigma_min=log(3))
dat <- list(Aoto=otoliths_skj$age, Loto=otoliths_skj$len)
model <- vonberto(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("Linf", "k", "t0", "sigma_min")]

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

# Model 5: Fit to skipjack tags only

init <- list(log_Linf=log(80), log_k=log(0.8), t0=-0.5,
             log_sigma_min=log(3), log_sigma_max=log(3),
             log_age=log(tags_skj$lenRel/60))
dat <- list(Lrel=tags_skj$lenRel, Lrec=tags_skj$lenRec,
            liberty=tags_skj$liberty)
model <- vonberto(init, dat)
fit <- nlminb(model$par, model$fn, model$gr,
              control=list(eval.max=1e4, iter.max=1e4))
model$report()[c("Linf", "k", "t0", "sigma_min", "sigma_max")]
```

Index