

Package ‘DEBtoolAnimal’

June 1, 2016

Type Package

Title DEB functions for an animal

Version 0.1

Date 2015-09-30

Author Goncalo M. Marques <goncalo.marques@tecnico.ulisboa.pt>

Maintainer Goncalo M. Marques <goncalo.marques@tecnico.ulisboa.pt>

Description

DEB based functions for the std (standard) and abj (with acceleration) models for animals.

License GPL

LazyData TRUE

NeedsCompilation no

R topics documented:

addchem

Sets chemical parameters and text for units and labels

Description

Sets chemical parameters and text for units and labels

Usage

```
addchem(par, units, label, free, phylum, class)
```

Arguments

par	data frame with parameter values
units	data frame with parameter units
label	data frame with parameter labels
free	data frame with information on which parameter to free or fix
phylum	string with species phylum
class	string with species class

Details

Calls `get_d_V` to set specific density of structure. For a specific density of wet mass of 1 g/cm³, a specific density of `d_E = d_V = 0.1 g/cm3` means a dry-over-wet weight ratio of 0.1

Value

list with updated par, units, label and free

See Also

Other add-my-pet auxiliary functions: [addpseudodata](#); [fieldnm_wtxt](#); [fieldnmnst](#); [get_d_V](#); [parscomp](#); [predict_pseudodata](#); [print_filterflag](#); [rmpseudodata](#); [setweights](#)

Examples

```
pars_init_my_pet(metaData)
```

`addpseudodata`

Adds pseudodata information into inputted data structures

Description

Adds the pseudodata information and weights for purposes of the regression

Usage

```
addpseudodata(data = list(), units = list(), label = list(),
              weights = list())
```

Arguments

<code>data</code>	structure with data
<code>units</code>	structure with data units
<code>label</code>	structure with data labels
<code>weights</code>	structure with weights

Value

structures with data, units, label and weights

See Also

Other add-my-pet auxiliary functions: [addchem](#); [fieldnm_wtxt](#); [fieldnmnst](#); [get_d_V](#); [parscomp](#); [predict_pseudodata](#); [print_filterflag](#); [rmpseudodata](#); [setweights](#)

Examples

```
list[data, units, label, weight] <- addpseudodata();
```

beta0	<i>Particular incomplete beta function</i>
-------	--

Description

particular incomplete beta function:

Usage

beta0(x0, x1)

Arguments

x0	scalar with lower boundary for integration
x1	scalar with upper boundary for integration

Details

Computes

$$B_{x_1} \left(\frac{4}{3}, 0 \right) - B_{x_0} \left(\frac{4}{3}, 0 \right) = \int_{x_0}^{x_1} t^{4/3-1} (1-t)^{-1} dt$$

To be used in the computation of the age at birth (or related quantities) for an egg.

Value

scalar with particular incomplete beta function

See Also

Other miscellaneous functions: [C2K](#); [K2C](#)

Examples

beta0(0.1, 0.2)

C2K	<i>Conversion of Celsius to Kelvin</i>
-----	--

Description

Converts temperature in degrees Celsius to Kelvin

Usage

C2K(C)

Arguments

C	numeric temperature in degrees Celsius
---	--

Value

temperature in Kelvin

See Also

Other miscellaneous functions: [K2C](#); [beta0](#)

Examples

```
C2K(20)
```

dget_lbarb2

Computes derivative d delta/dx

Description

Obtains the derivative d delta/dx from lbarb, xb and k.

Usage

```
dget_lbarb2(x, delta, pars)
```

Arguments

x	scalar $x = g/(g + e)$
delta	scalar $\delta = x e_H / (1 - \kappa)g$
pars	data.frame with lbarb, xb, xb3 ($xb^{1/3}$), k

Value

scalar with derivative value d delta/ dx

See Also

Other scaled get functions: [fnget_lbarb2](#); [get_lbarb2](#); [get_lbarb](#); [get_lb](#); [get_lp](#); [get_tb](#); [get_tm_s](#); [get_tp](#); [initial_scaled_reserve](#); [reprod_rate](#)

Examples

```
dget_lbarb2(10^(-6), 0, c(lbarb = 0.003, xb = 10/11, xb3 = (10/11)^(1/3), k = 1))
```

dget_l_ISO	<i>Computes derivative d l/d vH</i>
------------	-------------------------------------

Description

Obtains the derivative $d l/d vH$ from g, k, lT, f, sM.

Usage

dget_l_ISO(vH, l, pars)

Arguments

vH	scaled maturity volume
l	scaled length
pars	data.frame with g, k, lT, f, sM

Value

scalar with derivative value $d l/d vH$

estim_options	<i>Sets options for estim_pars</i>
---------------	------------------------------------

Description

Sets options for estimation one by one

Usage

estim_options(key = "inexistent", val = "")

Arguments

key	string with option to set
val	value of the option

Details

no input: print values to screen

one input:

* "default": sets options at default values * any other key (see below): print value to screen

two inputs:

* "filter": 1 - use filter (default); 0 - do not; * "pars_init_method": 0 - get initial estimates from automatized computation (default) 1 - read initial estimates from .mat file (for continuation) 2 - read initial estimates from pars_init file * "pseudodata_pets": 0 - put pseudodata together with data (default) 1 - put it apart (only for multispecies estimation) * "results_output": 0 - prints results to screen (default) 1 - prints results to screen, saves to .mat file 2 - saves data to .mat file and graphs to

.png files (prints results to screen using a customized results file when it exists) * "method": "nm" - use Nelder-Mead method; "no" - do not estimate;
for other options see corresponding options file of the method (e.g. nmregr_options)

See Also

Other regression functions: [nmregr_options](#)

Examples

```
estim_options("default")
```

estim_pars	<i>Estimates parameters</i>
------------	-----------------------------

Description

Runs the entire estimation procedures: gets the parameters, gets the data, initiates the estimation procedure and sends the results for handling

Usage

```
estim_pars()
```

See Also

Other add-my-pet functions: [mydata_pets](#); [petregr_f](#); [predict_pets](#); [printpar](#); [printprd](#); [results_pets](#)

Examples

```
estim_pars()
```

fieldnmnst	<i>Creates a list of field names of a structure</i>
------------	---

Description

Creates a list of field names of a structure

Usage

```
fieldnmnst(st)
```

Arguments

st data frame with fields

Value

list of list of strings with fields including the field name in str

See Also

Other add-my-pet auxiliary functions: [addchem](#); [addpseudodata](#); [fieldnm_wtxt](#); [get_d_V](#); [parscomp](#); [predict_pseudodata](#); [print_filterflag](#); [rmpseudodata](#); [setweights](#)

Examples

```
nst <- fieldnmnst(st)
```

fieldnm_wtxt	<i>Searches for fields with a given name in a multilevel structure</i>
--------------	--

Description

Creates a list of field names of a data frame with str

Usage

```
fieldnm_wtxt(data = list(), str = "")
```

Arguments

data	data frame with fields
str	string with field name

Value

list of strings with fields including the field name in str

See Also

Other add-my-pet auxiliary functions: [addchem](#); [addpseudodata](#); [fieldnmnst](#); [get_d_V](#); [parscomp](#); [predict_pseudodata](#); [print_filterflag](#); [rmpseudodata](#); [setweights](#)

Examples

```
nm-psd <- fieldnm_wtxt(data, "psd")
```

filter_std	<i>Filters for allowed parameters of standard DEB model without acceleration</i>
------------	--

Description

Checks if parameter values are in the allowable part of the parameter space of standard DEB model without acceleration. Meant to be run in the estimation procedure

Usage

```
filter_std(par)
```

Arguments

par data frame with parameter values

Details

The flag is an indicator of reason for not passing the filter and it means 0: parameters pass the filter 1: some parameter is negative 2: some kappa is larger than 1 3: growth efficiency is larger than 1 4: maturity levels do not increase during life cycle 5: puberty cannot be reached

Value

list with filter and flag

See Also

Other filter functions: [reach_birth](#); [warning_std](#)

Examples

```
filter_std(par)
```

fnget_lbarb2	<i>Computes f using the ode solver for delta(x), for finding lbarb</i>
--------------	--

Description

Computes f using the ode solver for delta(x), for finding lbarb.

Usage

```
fnget_lbarb2(lbarb, pars)
```

Arguments

lbarb scalar with scaled length at birth (lbarb = lb/ g)
 pars data.frame with lbarb, xb, xb3 (xb^1/3), k

Value

scalar with function f which when zero indicates lbarb

See Also

Other scaled get functions: [dget_lbarb2](#); [get_lbarb2](#); [get_lbarb](#); [get_lb](#); [get_lp](#); [get_tb](#); [get_tm_s](#); [get_tp](#); [initial_scaled_reserve](#); [reprod_rate](#)

Examples

```
fngget_lbarb2(0.03, c(xb = 10/11, xb3 = (10/11)^(1/3), vbarHb = 0.001, k = 1))
```

get_d_V

Sets chemical parameters and text for units and labels

Description

Sets chemical parameters and text for units and labels

Usage

```
get_d_V(phylum, class)
```

Arguments

phylum	string with species phylum
class	string with species class

Details

Calls `get_d_V` to set specific density of structure. For a specific density of wet mass of 1 g/cm³, a specific density of `d_E = d_V = 0.1 g/cm3` means a dry-over-wet weight ratio of 0.1

Value

list with `d_V` and `info`

See Also

Other add-my-pet auxiliary functions: [addchem](#); [addpseudodata](#); [fieldnm_wtxt](#); [fieldnmnst](#); [parscomp](#); [predict_pseudodata](#); [print_filterflag](#); [rmpseudodata](#); [setweights](#)

Examples

```
get_d_V("Chordata", "Mammalia")
```

get_lb

*Computes scaled length at birth***Description**

Obtains scaled length at birth, given the scaled reserve density at birth.

Usage

```
get_lb(pars, eb = 1, lb0 = as.numeric(pars[3]^(1/3)))
```

Arguments

pars	3-vector with parameters: g, k, v _H ^b
eb	optional scalar with scaled reserve density at birth (default eb = 1)
lb0	optional scalar with initial estimate for scaled length at birth (default lb0: lb for k = 1)

Details

The theory behind get_lb, get_tb and get_ue0 is discussed in <http://www.bio.vu.nl/thb/research/bib/Kooy2009b.html>. Solves $y(x_b) = y_b$ for l_b with explicit solution for $y(x)$

$$y(x) = \frac{x e_H}{1 - k a p} = x g \frac{u_H}{l^3}$$

and $y_b = x_b g u_H^b / ((1 - k a p) l_b^3)$

$$\frac{d}{dx} y = r(x) - y s(x)$$

with solution $y(x) = v(x) \int_0^x r(x')/v(x') dx'$ and $v(x) = \exp(-\int_0^x s(x') dx')$. A Newton Raphson scheme is used with Euler integration, starting from an optional initial value. Shooting method: get_lb2. In case of no convergence, get_lb2 is run automatically as backup. Consider the application of get_lb_foetus for an alternative initial value.

Value

scalar with scaled length at birth (lb) and indicator equals 1 if successful convergence, 0 otherwise (info)

See Also

Other scaled get functions: [dget_lbarb2](#); [fnget_lbarb2](#); [get_lbarb2](#); [get_lbarb](#); [get_lp](#); [get_tb](#); [get_tm_s](#); [get_tp](#); [initial_scaled_reserve](#); [reprod_rate](#)

Examples

```
get_lb(c(g = 10, k = 1, vHb = 0.5), 1)
```

get_lbarb	<i>Computes scaled length at birth lbarb</i>
-----------	--

Description

Obtains scaled length at birth, given the scaled reserve density at birth.

Usage

```
get_lbarb(pars, eb = 1, lbarb0 = NA)
```

Arguments

pars	3-vector with parameters: g, k, vbar_H^b
eb	optional scalar with scaled reserve density at birth (default eb = 1)
lbarb0	optional scalar with initial estimate for scaled length at birth (default lbarb0: lbarb for k = 1)

Value

scalar with scaled length at birth (lbarb) and indicator equals 1 if successful, 0 otherwise (info)

See Also

Other scaled get functions: [dget_lbarb2](#); [fngget_lbarb2](#); [get_lbarb2](#); [get_lb](#); [get_lp](#); [get_tb](#); [get_tm_s](#); [get_tp](#); [initial_scaled_reserve](#); [reprod_rate](#)

Examples

```
get_lbarb(c(g = 10, k = 1, vbarHb = 0.0005), 1)
```

get_lbarb2	<i>Computes initial scaled reserve</i>
------------	--

Description

Obtains scaled length at birth, given the scaled reserve density at birth. Like get_lbarb, but uses a shooting method in 1 variable.

Usage

```
get_lbarb2(pars, eb = NA)
```

Arguments

pars	3-vector with parameters: g, k, vbar_H^b
eb	optional scalar with scaled reserve density at birth (default eb = 1)

Value

scalar with scaled length at birth (lbarb) and indicator equals 1 if successful, 0 otherwise (info)

See Also

Other scaled get functions: [dget_lbarb2](#); [fnget_lbarb2](#); [get_lbarb](#); [get_lb](#); [get_lp](#); [get_tb](#); [get_tm_s](#); [get_tp](#); [initial_scaled_reserve](#); [reprod_rate](#)

Examples

```
get_lbarb2(c(g = 10, k = 1, vbarHb = 0.01), 1)
```

get_lp	<i>Computes scaled length at puberty</i>
--------	--

Description

Obtains scaled length at puberty at constant food density.

Usage

```
get_lp(pars, f = 1, lb0 = NA)
```

Arguments

pars	5-vector with parameters: g , k , l_T , v_H^b , v_H^p
lb0	optional scalar with initial estimate for scaled length at birth (default lb0: lb for $k = 1$)
eb	optional scalar with scaled reserve density at birth (default eb = 1)

Details

If scaled length at birth (second input) is not specified, it is computed (using automatic initial estimate). If it is specified, however, is it just copied to the (second) output. Food density is assumed to be constant.

Value

scaled length at puberty (lp), scalar length at birth (lb) and indicator equals 1 if successful convergence, 0 otherwise (info)

See Also

Other scaled get functions: [dget_lbarb2](#); [fnget_lbarb2](#); [get_lbarb2](#); [get_lbarb](#); [get_lb](#); [get_tb](#); [get_tm_s](#); [get_tp](#); [initial_scaled_reserve](#); [reprod_rate](#)

Examples

```
get_lp(c(g = 10, k = 1, lT = 0, vHb = 0.5, vHp = 10), 1)
```

get_tb	<i>Gets scaled age at birth</i>
--------	---------------------------------

Description

Obtains scaled age at birth, given the scaled reserve density at birth.

Usage

```
get_tb(pars, eb = 1, lb = NA)
```

Arguments

pars	3-vector with parameters: g , k , v_H^b
eb	optional scalar with scaled reserve density at birth (default $eb = 1$)
lb0	optional scalar with initial estimate for scaled length at birth (default lb0: lb for $k = 1$)

Details

Multiply the result with the somatic maintenance rate coefficient to arrive at age at puberty.

Value

list with scaled age at birth $\tau_b = a_b k_M(t_b)$, scaled length at birth (lb) and indicator equals 1 if successful convergence, 0 otherwise (info)

See Also

Other scaled get functions: [dget_lbarb2](#); [fnget_lbarb2](#); [get_lbarb2](#); [get_lbarb](#); [get_lb](#); [get_lp](#); [get_tm_s](#); [get_tp](#); [initial_scaled_reserve](#); [reprod_rate](#)

Examples

```
get_tb(c(g = 10, k = 1, vHb = 0.5), 1)
```

get_tm_s	<i>Obtains scaled mean age at death for short growth periods</i>
----------	--

Description

Obtains scaled mean age at death assuming a short growth period relative to the life span

Usage

```
get_tm_s(pars, f = 1, lb = NA, lp = NA)
```

Arguments

pars	4 or 7-vector with parameters: [g lT ha sG] or [g k lT vHb vHp ha SG]
f	optional scalar with scaled reserve density at birth (default f = 1)
lb	optional scalar with scaled length at birth (default: lb is obtained from get_lb)
lp	optional scalar with scaled length at puberty

Details

Divide the result by the somatic maintenance rate coefficient to arrive at the mean age at death. The variant get_tm_foetus does the same in case of foetal development. If the input parameter vector has only 4 elements (for [g, lT, ha/ km2, sG]), it skips the calculation of the survival probability at birth and puberty.

Value

list with scalar with scaled mean life span (tm), scalar with survival probability at birth (if length p = 7) (Sb), scalar with survival probability at puberty (if length p = 7) (Sp) and indicator equals 1 if successful convergence, 0 otherwise (info)

See Also

Other scaled get functions: [dget_lbarb2](#); [fnget_lbarb2](#); [get_lbarb2](#); [get_lbarb](#); [get_lb](#); [get_lp](#); [get_tb](#); [get_tp](#); [initial_scaled_reserve](#); [reprod_rate](#)

get_tp

Gets scaled age at puberty

Description

Obtains scaled age at puberty.

Usage

```
get_tp(pars, f = 1, lb0 = as.numeric(pars[4]^(1/3)))
```

Arguments

pars	5-vector with parameters: g, k, l_T, v_H^b, v_H^p
f	optional scalar with functional response (default f = 1)
lb0	optional scalar with scaled length at birth (default lb0: lb for k = 1)

Details

Food density is assumed to be constant. Multiply the result with the somatic maintenance rate coefficient to arrive at age at puberty.

Value

list with scaled age at puberty $\tau_p = a_p k_M (tp)$, scaled age at birth $\tau_b = a_b k_M (tb)$, scaled length at puberty (lp), scaled length at birth (lb) and indicator equals 1 if successful convergence, 0 otherwise (info)

See Also

Other scaled get functions: [dget_lbarb2](#); [fnget_lbarb2](#); [get_lbarb2](#); [get_lbarb](#); [get_lb](#); [get_lp](#); [get_tb](#); [get_tm_s](#); [initial_scaled_reserve](#); [reprod_rate](#)

Examples

```
get_tp(c(g = 10, k = 1, lT = 0, vHb = 0.5, vHp = 10), 1)
```

get_ubarE0

Computes initial scaled reserve density at birth

Description

Obtains the initial scaled reserve given the scaled reserve density at birth. Function `get_ue0` does so for eggs, `get_ue0_foetus` for foetuses. Specification of length at birth as third input by-passes its computation, so if you want to specify an initial value for this quantity, you should use `get_lb` directly.

Usage

```
get_ubarE0(g = NA, k = NA, vbarHb = NA, eb = 1, lbarb = NA)
```

Arguments

<code>g</code>	energy investment ratio
<code>k</code>	maintenance ratio
<code>vbarHb</code>	rescaled maturity volume at birth
<code>eb</code>	optional scalar with scaled reserbe density at birth
<code>lbarb</code>	optional scalar with scaled length at birth

Value

scalar with particular incomple beta function

See Also

Other get functions: [get_ue0](#)

Examples

```
get_ubarE0(g = 10, lbarb = 0.01)
get_ubarE0(g = 10, k = 0.7, vbarHb = 5e-4)
```

get_ue0	<i>Computes initial scaled reserve</i>
---------	--

Description

Obtains the initial scaled reserve given the scaled reserve density at birth. Function get_ue0 does so for eggs, get_ue0_foetus for foetuses. Specification of length at birth as third input by-passes its computation, so if you want to specify an initial value for this quantity, you should use get_lb directly.

Usage

```
get_ue0(pars, eb = 1, lb0 = NA)
```

Arguments

pars	1 or 3 -vector with parameters g , k_J/k_M , v_H^b , see get_lb
eb	optional scalar with scaled reserbe density at birth (default: $eb = 1$)
lb0	optional scalar with scaled length at birth (default: lb is optained from get_lb)

Value

ue0 scalar with scaled reserve at $t=0$: $U_E^0 g^2 k_M^3 / v^2$ with $U_E^0 = M_E^0 / \{J_{EAm}\}$,
lb scalar with scaled length at birth and info indicator equals 1 if successful, 0 otherwise

See Also

Other get functions: [get_ubarE0](#)

Examples

```
get_ue0(pars = c(0.42, 1, 0.066), eb = 1, lb0 = 0.4042)
```

initial_scaled_reserve	<i>Gets initial scaled reserve</i>
------------------------	------------------------------------

Description

Gets initial scaled reserve.

Usage

```
initial_scaled_reserve(f, pars, Lb0 = NA)
```

Arguments

f	n-vector with scaled functional responses
pars	5-vector with parameters: VHb , g , k_J , k_M , v
Lb0	optional n-vector with lengths at birth

Value

n-vector with initial scaled reserve: $M_E^0 / J_{EAm}(U_0)$, n-vector with length at birth (Lb) and n-vector with 1's if successful, 0's otherwise (info)

See Also

Other scaled get functions: [dget_lbarb2](#); [fnget_lbarb2](#); [get_lbarb2](#); [get_lbarb](#); [get_lb](#); [get_lp](#); [get_tb](#); [get_tm_s](#); [get_tp](#); [reprod_rate](#)

Examples

`initial_scaled_reserve(f = c(1, 0.9), pars = c(VHb = .8, g = .42, kJ = 1.7, kM = 1.7, v = 3.24))`

K2C	<i>Conversion of Kelvin to Celsius</i>
-----	--

Description

Converts temperature in Kelvin to degrees Celsius

Usage

`K2C(K)`

Arguments

K numeric temperature in degrees Kelvin

Value

temperature in Kelvin

See Also

Other miscellaneous functions: [C2K](#); [beta0](#)

Examples

`K2C(293.15)`

mre_st	<i>Computes mean relative error</i>
--------	-------------------------------------

Description

Computes relative errors and mean relative error for using data and predictions

Usage

```
mre_st(func, par, data, auxData, weights)
```

Arguments

func	string with predict file name
par	data frame with parameter values
data	data frame with data values
auxData	data frame with auxiliary data values
weights	data frame with values of weights

Examples

```
results_pets(par, metaPar, txtPar, data, auxData, metaData, txtData, weights)
```

mydata_my_pet	<i>Sets referenced data</i>
---------------	-----------------------------

Description

Sets data, pseudodata, metadata, auxdata, explanatory text, weights coefficients. Meant to be a template in add-my-pet

Usage

```
mydata_my_pet()
```

Value

list with data, auxData, metaData, txtData and weights

See Also

Other add-my-pet template functions: [pars_init_my_pet](#); [predict_my_pet](#)

Examples

```
mydata_my_pet()
```

mydata_pets	<i>Concatenates mydata files for several species</i>
-------------	--

Description

Concatenates mydata files for several species

Usage

```
mydata_pets()
```

Value

structure with data, auxData, metaData, txtData and weights for several pets

See Also

Other add-my-pet functions: [estim_pars](#); [petregr_f](#); [predict_pets](#); [printpar](#); [printprd](#); [results_pets](#)

Examples

```
mydata_pets()
```

nmregr_options	<i>Sets options for function nmregr</i>
----------------	---

Description

Sets options for estimation one by one

Usage

```
nmregr_options(key = "inexistent", val = "")
```

Arguments

key	string with option to set
val	value of the option

Details

no input: print values to screen

one input:

* "default": sets options at default values * any other key (see below): print value to screen

two inputs:

* "report": 1 - to report steps to screen; 0 - not to; * "max_step_number": maximum number of steps * "max_fun_evals": maximum number of function evaluations * "tol_simplex": tolerance for how close the simplex points must be together to call them the same * "tol_tun": tolerance for how close the loss-function values must be together to call them the same * "simplex_size": fraction added (subtracted if negative) to the free parameters when building the simplex

Value

1 if input is valid key, 0 if input is unknown key

See Also

Other regression functions: [estim_options](#)

Examples

```
nmregr_options("default")
```

parscomp	<i>Computes compound parameters from primary parameters</i>
----------	---

Description

Computes compound parameters from primary parameters that are frequently used

Usage

```
parscomp(par)
```

Arguments

par	data frame with parameter values
-----	----------------------------------

Value

list with compound parameters

See Also

Other add-my-pet auxiliary functions: [addchem](#); [addpseudodata](#); [fieldnm_wtxt](#); [fieldnmnst](#); [get_d_V](#); [predict_pseudodata](#); [print_filterflag](#); [rmpseudodata](#); [setweights](#)

Examples

```
parscomp(par)
```

pars_init_my_pet	<i>Sets (initial values for) parameters</i>
------------------	---

Description

Sets (initial values for) parameters\$ Meant to be a template in add-my-pet

Usage

```
pars_init_my_pet(metaData)
```

Arguments

metaData	data frame with info about this entry (needed for names of phylum and class to get d_V)
----------	---

Value

list with par (with values of parameters), metaPar (with information on metaparameters) and txtPar (with information on parameters)

See Also

Other add-my-pet template functions: [mydata_my_pet](#); [predict_my_pet](#)

Examples

```
pars_init_my_pet(metaData)
```

petregr_f	<i>Calculates least squares estimates using Nelder Mead's simplex method using a filter</i>
-----------	---

Description

Calculates least squares estimates using Nelder Mead's simplex method using a filter

Usage

```
petregr_f(func, par, data, auxData, weights, filternm)
```

Arguments

func	character string with name of user-defined function
par	list with parameters
data	list with data
auxData	list with auxiliary data
weights	list with weights
filternm	character string with name of user-defined filter function

Value

list with list with parameters resulting from estimation procedure (par) and indicator 1 if convergence has been successful or 0 otherwise (info)

See Also

Other add-my-pet functions: [estim_pars](#); [mydata_pets](#); [predict_pets](#); [printpar](#); [printprd](#); [results_pets](#)

predict_my_pet	<i>Obtains predictions, using parameters and data</i>
----------------	---

Description

Obtains predictions, using parameters and data

Usage

```
predict_my_pet(par, data, auxData = list())
```

Arguments

par	data frame with parameter values
data	data frame with data values
auxData	data frame with auxiliary data values

Value

list with prdData (data frame with values of predictions) and info (indicator for customized filters)

See Also

Other add-my-pet template functions: [mydata_my_pet](#); [pars_init_my_pet](#)

Examples

```
predict_my_pet(par, data, auxData)
```

predict_pets	<i>Concatenates predict files for several species</i>
--------------	---

Description

Concatenates predict files for several species

Usage

```
predict_pets(parGrp, data, auxData)
```

Arguments

parGrp	data frame with parameter values of the group
data	data frame with data values
auxData	data frame with auxiliary data values

Value

structure with prdData and prdInfo for several pets

See Also

Other add-my-pet functions: [estim_pars](#); [mydata_pets](#); [petregr_f](#); [printpar](#); [printprd](#); [results_pets](#)

predict_pseudodata	<i>Predicts pseudodata values</i>
--------------------	-----------------------------------

Description

Adds pseudodata predictions into predictions structure

Usage

```
predict_pseudodata(par, data, prdData)
```

Arguments

par	data frame with parameter values
data	data frame with data values
prdData	data frame with prediction values

Value

structure with pseudodata predictions

See Also

Other add-my-pet auxiliary functions: [addchem](#); [addpseudodata](#); [fieldnm_wtxt](#); [fieldnmnst](#); [get_d_V](#); [parscomp](#); [print_filterflag](#); [rmpseudodata](#); [setweights](#)

printpar	<i>Prints parameters of a species to screen</i>
----------	---

Description

Prints parameters of a species to screen

Usage

```
printpar(par, txtPar)
```

Arguments

par	list with parameter values
txtPar	list with text info on parameters

See Also

Other add-my-pet functions: [estim_pars](#); [mydata_pets](#); [petregr_f](#); [predict_pets](#); [printpar](#); [results_pets](#)

printprd	<i>Prints data of a species to screen</i>
----------	---

Description

Prints data of a species to screen

Usage

```
printprd(data, txtData, prdData, RE)
```

Arguments

data	list with data values
txtData	list with text info on data
prdData	list with prediction values
RE	list with relative errors

See Also

Other add-my-pet functions: [estim_pars](#); [mydata_pets](#); [petregr_f](#); [predict_pets](#); [printpar](#); [results_pets](#)

print_filterflag	<i>Prints an explanation of the filter flag onto the screen</i>
------------------	---

Description

Prints an explanation to the screen according to the flag produced by a filter. Meant to be run in the estimation procedure for the seed parameter set

Usage

```
print_filterflag(flag)
```

Arguments

flag	integer with code from filter
------	-------------------------------

See Also

Other add-my-pet auxiliary functions: [addchem](#); [addpseudodata](#); [fieldnm_wtxt](#); [fieldnmnst](#); [get_d_V](#); [parscomp](#); [predict_pseudodata](#); [rmpseudodata](#); [setweights](#)

Examples

```
print_filterflag(3)
```

reach_birth	<i>Checks if parameters allow for reaching birth in the standard DEB model</i>
-------------	--

Description

Checks if parameters allow for reaching birth in the standard DEB model

Usage

```
reach_birth(g, k, vHb, f = 1)
```

Arguments

g	energy investment ratio
k	ratio of maturity and somatic maintenance rate coeff
vHb	scaled maturity volume at birth
f	functional response (default 1)

Value

info, indicator equals 1 if reaches birth, 0 otherwise

See Also

Other filter functions: [filter_std](#); [warning_std](#)

Examples

```
reach_birth(g = 10, k = 1, vHb = 0.5)
```

reprod_rate	<i>Gets reproduction rate</i>
-------------	-------------------------------

Description

Calculates the reproduction rate in number of eggs per time for an individual of length L and scaled reserve density f.

Usage

```
reprod_rate(L, f = 1, pars, Lf = NA)
```

Arguments

L	n-vector with length
f	scalar with functional response
pars	9-vector with parameters: kap, kapR, g, kJ, kM, LT, v, UHb, UHp
Lf	optional scalar with length at birth (initial value only) or optional 2-vector with length, L, and scaled functional response f0 for a juvenile that is now exposed to f, but previously at another f

Value

list with n-vector with reproduction rates (R), scalar with scaled initial reserve (UE0), scalar with (volumetric) length at birth (Lb), scalar with (volumetric) length at puberty (Lp) and indicator with 1 for success, 0 otherwise (info)

See Also

Other scaled get functions: [dget_lbarb2](#); [fnget_lbarb2](#); [get_lbarb2](#); [get_lbarb](#); [get_lb](#); [get_lp](#); [get_tb](#); [get_tm_s](#); [get_tp](#); [initial_scaled_reserve](#)

results_pets	<i>Prints results of estimation</i>
--------------	-------------------------------------

Description

Prints the results of the estimation procedure in the screen, .mat file and makes figures of graphs

Usage

```
results_pets(par, metaPar, txtPar, data, auxData, metaData, txtData, weights)
```

Arguments

par	data frame with parameter values
metaPar	data frame with metainformation on models
txtPar	data frame with information on parameters
data	data frame with data values
auxData	data frame with auxiliary data values
metaData	data frame with metainformation on the entry
txtData	data frame with information on data
weights	data frame with values of weights

See Also

Other add-my-pet functions: [estim_pars](#); [mydata_pets](#); [petreg_f](#); [predict_pets](#); [printpar](#); [printprd](#)

Examples

```
results_pets(par, metaPar, txtPar, data, auxData, metaData, txtData, weights)
```

rmpseudodata	<i>Removes pseudodata information from inputted data structures</i>
--------------	---

Description

Removes pseudodata information from inputted data structures

Usage

```
rmpseudodata(data = list())
```

Arguments

data	structure with "psd" field to be removed
------	--

Value

structure with "psd" field removed

See Also

Other add-my-pet auxiliary functions: [addchem](#); [addpseudodata](#); [fieldnm_wtxt](#); [fieldnmnst](#); [get_d_V](#); [parscomp](#); [predict_pseudodata](#); [print_filterflag](#); [setweights](#)

Examples

```
data <- rmpseudodata(data)
```

setweights	<i>Sets automatically the weights for the data (to be used in a regression)</i>
------------	---

Description

computes weights for given data and adds it to the weight structure

Usage

```
setweights(data, weights = list())
```

Arguments

data	structure with data
weights	structure with weights

Details

computes weights for given data and adds it to the weight structure for the zero-variate data y, the weight will be

$$\min(100, 1/\max(10^{-6}, y)^2)$$

for the uni-variate data y, the weight will be

$$1/N\bar{y}^2$$

Value

structure with weights

See Also

Other add-my-pet auxiliary functions: [addchem](#); [addpseudodata](#); [fieldnm_wtxt](#); [fieldnmnst](#); [get_d_V](#); [parscomp](#); [predict_pseudodata](#); [print_filterflag](#); [rmpseudodata](#)

Examples

```
setweights(data)
```

tempcorr	<i>Temperature correction</i>
----------	-------------------------------

Description

Calculates the factor with which physiological rates should be multiplied to go from a reference temperature to a given temperature

Usage

```
tempcorr(Temp, T_1, T_A, T_L = NA, T_AL = NA, T_H = NA, T_AH = NA)
```

Arguments

Temp	vector with temperatures (in Kelvin)
T_1	scalar with reference temperature (in Kelvin)
T_A	scalar with Arrhenius temperature (in Kelvin)
T_L	optional scalar with lower boundary of temperature range (in Kelvin)
T_AL	optional scalar with Arrhenius temperature for lower boundary of temperature range (in Kelvin)
T_H	optional scalar with upper boundary of temperature range (in Kelvin)
T_AH	optional scalar with Arrhenius temperature for upper boundary of temperature range (in Kelvin)

Details

Temperature impacts metabolic rates. This impact, in its most simplest way (1 parameter), is modeled by multiplying all the time-dependent parameters by a correction factor:

$$\exp\left(\frac{T_A}{T_1} - \frac{T_A}{T}\right)$$

For a more detailed modeling one can multiply with an extra fraction $s(T_1)/s(T)$ with (3 parameters):

$$s(T) = 1 + \exp\left(\frac{T_{AL}}{T} - \frac{T_{AL}}{T_L}\right)$$

or (5 parameters)

$$s(T) = 1 + \exp\left(\frac{T_{AL}}{T} - \frac{T_{AL}}{T_L}\right) + \exp\left(\frac{T_{AH}}{T_H} - \frac{T_{AH}}{T}\right)$$

Value

vector with temperature correction factors that affect all rates

Examples

```
tempcorr(c(330, 331, 332), 320, T_A = 12000, T_L = 277, T_H = 331, T_AL = 20000, T_AH = 190000)
```

warning_std	<i>Warns of unreasonable parameters for the standard DEB model without acceleration</i>
-------------	---

Description

Checks if parameter values are in the reasonable part of the parameter space of standard DEB model without acceleration, produces warnings. Meant to be run after the estimation procedure

Usage

```
warning_std(par)
```

Arguments

par	data frame with parameter values
-----	----------------------------------

See Also

Other filter functions: [filter_std](#); [reach_birth](#)

Examples

```
warning_std(par)
```