

Package ‘embryogrowth’

October 16, 2015

Type Package

Title Tools to Analyze the Thermal Reaction Norm of Embryo Growth

Version 6.1.1

Date 2015-10-16

Author Marc Girondot <marc.girondot@u-psud.fr>

Maintainer Marc Girondot <marc.girondot@u-psud.fr>

Description Tools to analyze the embryo growth and the sexualisation thermal reaction norms.

Depends deSolve, polynom, parallel, HelpersMG (>= 1.2),R (>= 2.14.2)

Suggests entropy, shiny, coda, optimx, numDeriv

License GPL-2

LazyData yes

LazyLoad yes

Encoding UTF-8

NeedsCompilation no

Repository CRAN

Date/Publication 2015-10-16 20:30:28

R topics documented:

embryogrowth-package	2
ChangeSSM	5
dydt.exponential	6
dydt.Gompertz	8
dydt.linear	9
FormatNests	10
GenerateAnchor	11
GenerateConstInc	12
GenerateTest	13
GRTRN_MHmcmc	14
hist.Nests	16
hist.NestsResult	17

info.nests	18
likelihoodR	21
logLik.NestsResult	23
logLik.tsd	24
MovingIncubation	25
nest	26
plot.NestsResult	27
plot.tsd	30
plotR	32
plotR_hist	34
plot_transition	35
predict.tsd	36
resultNest_4p	37
resultNest_4p_transition	38
resultNest_4p_weight	39
resultNest_6p	40
resultNest_newp	41
result_mcmc_4p	42
result_mcmc_4p_weight	43
result_mcmc_6p	44
result_mcmc_newp	46
searchR	47
STRN	49
STRN_MHmcmc	51
STSRE_NestingArea	53
STSRE_TSD	54
summary.Nests	57
switch.transition	58
tempConst	59
TestParallel	60
TRN_MHmcmc_p	61
tsd	62
tsd_MHmcmc	65
tsd_MHmcmc_p	67
TSP.list	68
weightmaxentropy	69
Index	71

embryogrowth-package *The package embryogrowth*

Description

Tools to analyze the embryo growth and the sexualisation thermal reaction norms.

Details

Fit a parametric function that describes dependency of embryo growth to temperature

```

Package:    embryogrowth
Type:      Package
Version:    6.1.1 - build 467
Date:      2015-10-16
License:    GPL (>= 2)
LazyLoad:  yes

```

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M. & Kaska, Y. 2014. A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*. 45, 96-102.

See Also

Delmas, V., Prevot-Julliard, A.-C., Pieau, C. & Girondot, M. 2008. A mechanistic model of temperature-dependent sex determination in a Chelonian, the European pond turtle. *Functional Ecology*, 22, 84-93.

Girondot, M., Ben Hassine, S., Sellos, C., Godfrey, M. & Guillon, J.-M. 2010. Modeling thermal influence on animal growth and sex determination in Reptiles: being closer of the target gives new views. *Sexual Development*, 4, 29-38.

Girondot, M. 1999. Statistical description of temperature-dependent sex determination using maximum likelihood. *Evolutionary Ecology Research*, 1, 479-486.

Girondot, M., & Kaska, Y. (2014). Nest temperatures in a loggerhead- nesting beach in Turkey is more determined by sea surface temperature than air temperature. *Journal of Thermal Biology*, 47, 13-18.

Examples

```

## Not run:
library("embryogrowth")
packageVersion("embryogrowth")
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(115.758929130522, 428.649022170996, 503.687251738993,
12.2621455821612, 306.308841227278, 116.35048615105), .Names = c("DHA",
"DHH", "DHL", "DT", "T12L", "Rho25"))
# or
x <- structure(c(118.431040984352, 498.205702157603, 306.056280989839,
118.189669472381), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)

```

```

resultNest_4p <- searchR(parameters=x, fixed.parameters=pfixed,
  temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
  test=c(Mean=39.33, SD=1.92))
data(resultNest_4p)
pMCMC <- TRN_MHmcmc_p(resultNest_4p, accept=TRUE)
# Take care, it can be very long, sometimes several days
result_mcmc_4p <- GRTRN_MHmcmc(result=resultNest_4p,
  parametersMCMC=pMCMC, n.iter=10000, n.chains = 1, n.adapt = 0,
  thin=1, trace=TRUE)
data(result_mcmc_4p)
out <- as.mcmc(result_mcmc_4p)
# This out obtained after as.mcmc can be used with coda package
# plot() can use the direct output of GRTRN_MHmcmc() function.
plot(result_mcmc_4p, parameters=1, xlim=c(0,550))
plot(result_mcmc_4p, parameters=3, xlim=c(290,320))
# summary() permits to get rapidly the standard errors for parameters
summary(result_mcmc_4p)
se <- result_mcmc_4p$SD

## End(Not run)

```

ChangeSSM

Generate set of parameters for Schoolfield-Sharpe-Magnuson model

Description

Generate a set of parameters for Schoolfield-Sharpe-Magnuson model

Usage

```

ChangeSSM(temperatures = (200:350)/10,
  parameters = stop("A set of parameters must be supplied"),
  initial.parameters = stop("A set of parameters for new model must be supplied"),
  ...)

```

Arguments

temperatures	A vector with incubation temperatures in degrees Celsius
parameters	A vector of parameters for model to be converted (4 or 6 parameters)
initial.parameters	A vector of parameters for initial model model to be fitted (4 or 6 parameters)
...	A control list to be used with optim, see ?optim

Details

ChangeSSM convert Schoolfield-Sharpe-Magnuson model from 4 to 6 parameters or reverse

Value

A vector with parameters

Author(s)

Marc Girondot

Examples

```
## Not run:
data(resultNest_6p)
x1 <- resultNest_6p$par
data(resultNest_4p)
x2 <- resultNest_4p$par
temperaturesC <- (200:350)/10
s <- ChangeSSM(temperatures=temperaturesC, parameters=x1, initial.parameters=x2)
plotR(list(resultNest_6p, resultNest_4p, s), ylim=c(0,0.3),
col=list("black", "red", "green"), lty=list(1,1,1),
legend=list("R function to mimic", "Initial new R function",
"Fitted new R function"), show.box=FALSE)
# Other example to fit anchored parameters
data(resultNest_4p)
x0 <- resultNest_4p$par
t <- hist(resultNest_4p, plot=FALSE)
x <- c(3.4, 3.6, 5.4, 5.6, 7.6, 7.5, 3.2)
names(x) <- seq(from=range(t$temperatures)[1], to=range(t$temperatures)[2],
length.out=7)
newx <- ChangeSSM(temperatures = (200:350)/10, parameters = x0,
initial.parameters = x,
control=list(maxit=5000))
# Example on how to generate a set of SSM parameters from anchored parameters
xanchor <- GenerateAnchor(nests=resultNest_4p)
x <- resultNest_4p$par
xanchor["294"] <- 0
xanchor["308"] <- 2.3291035
xprime <- ChangeSSM(parameters = xanchor,
initial.parameters = x, control=list(maxit=5000))
plotR(result=resultNest_4p, parameters=list(resultNest_4p$par, xprime$par),
ylim=c(0,0.3), col=c("black", "red"),
legend=list("Fitted parameters", "Constrained parameters"))

## End(Not run)
```

dydt.exponential

Return the derivative of the exponential function

Description

Return the derivative of the exponential function
dydt.exponential(t, size, parms)

Usage

```
dydt.exponential(t, size, parms)
```

Arguments

t	The time in any unit
size	The current size
parms	A vector with alpha and K values being c(alpha=x1, K=x2). K is not used.

Details

dydt.exponential returns the derivative of the exponential function.

Value

A list with the derivative

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(306.174998729436, 333.708348843241,
299.856306141849, 149.046870203155),
.Names = c("DHA", "DHH", "T12H", "Rho25"))
# K or rK are not used for dydt.linear or dydt.exponential
resultNest_4p_exponential <- searchR(parameters=x, fixed.parameters=NULL,
temperatures=formatted, derivate=dydt.exponential, M0=1.7,
test=c(Mean=39.33, SD=1.92))

## End(Not run)
```

dydt.Gompertz *Return the derivative of the Gompertz function*

Description

Return the derivative of the Gompertz function
 dydt.Gompertz(t, size, parms)

Usage

```
dydt.Gompertz(t, size, parms)
```

Arguments

t	The time in any unit
size	The current size
parms	A vector with alpha and K values being c(alpha=x1, K=x2)

Details

dydt.Gompertz returns the derivative of the Gompertz function.

Value

A list with the derivative

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(118.768297442004, 475.750095909406, 306.243694918151,
116.055824800264), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
# K or rK are not used for dydt.linear or dydt.exponential
resultNest_4p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_4p)
```



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

```
dydt.linear          Return the derivative of the linear function
```

Description

Return the derivative of the linear function
 dydt.linear(t, size, parms)

Usage

```
dydt.linear(t, size, parms)
```

Arguments

t	The time in any unit
size	The current size
parms	A vector with alpha being c(alpha=x1, K=x2). Only alpha is used.

Details

dydt.Linear returns the derivative of the linear function.

Value

A list with the derivative

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(306.174998729436, 333.708348843241, 299.856306141849,
149.046870203155), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# K or rK are not used for dydt.linear or dydt.exponential
resultNest_4p_linear <- searchR(parameters=x, fixed.parameters=NULL,
temperatures=formatted, derivate=dydt.linear, M0=1.7,
test=c(Mean=39.33, SD=1.92))
```

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

 FormatNests

Create a dataset of class Nests to be used with searchR

Description

Will create a dataset of class Nests to be used with searchR

FormatNests(nest, previous=x) with x being a previously formatted data.

The raw data must be organized being:

First column is the time in minutes since the beginning of incubation

Each column next is the trace of temperatures, one column for each nest.

For example, for two nests:

Time Nest1 Nest2

0 29.8 27.6

90 30.2 28.8

120 30.4 30.7

180 31.2 32.6

...

65800 30.8 32.6

65890 30.2

65950 30.4

The Nest1 ends incubation at 65800 minutes whereas Nest2 ends incubation at 65950 (last row with temperature for each).

The parameter Weight is a vector: weight=c(Nest1=1, Nest2=1.2)

It can be used to format database already formatted with old format; in this case, just use data=xxx with xxx being the old format database.

Usage

```
FormatNests(data = stop("A dataset must be provided !"), previous = NULL,
            simplify = TRUE, weight = NULL)
```

Arguments

data	Data to be newly formatted
previous	Data already formatted
simplify	If TRUE, simply the time series by removing identical time series of temperatures
weight	Named vector with weight for likelihood

Details

FormatNests creates a dataset of class "Nests" to be used with searchR

Value

A list with all the nests formatted to be used with searchR.

Author(s)

Marc Girondot

Examples

```
## Not run:  
library(embryogrowth)  
data(nest)  
formatted <- FormatNests(nest, previous=NULL)  
formatted <- FormatNests(nest)  
  
## End(Not run)
```

GenerateAnchor

Generate a set of anchored parameters

Description

Generate a set of anchored parameters.

It is important that the anchors (i.e. the temperatures used as anchors) encompass the highest and lowest temperatures that are present in nests.

Usage

```
GenerateAnchor(temperatures = NULL, nests = NULL, parameters = NULL,  
              number.anchors = 7)
```

Arguments

temperatures	A vector with temperatures to serve as anchors
nests	Formatted nest data or result object obtained from searchR()
parameters	A set of parameters value
number.anchors	Number of anchors

Details

GenerateAnchor Generate a set of anchored parameters

Value

A vector with parameters

Author(s)

Marc Girondot

Examples

```
## Not run:
# Example to generate anchored parameters
newp <- GenerateAnchor()
newp <- GenerateAnchor(temperatures=seq(from=20,
  to=35, length.out=7))
newp <- GenerateAnchor(number.anchors=7)
data(nest)
formatted <- FormatNests(nest, previous=NULL)
newp <- GenerateAnchor(nests=formatted)
newp <- GenerateAnchor(nests=formatted, number.anchors=10)
data(resultNest_4p)
newp <- GenerateAnchor(nests=resultNest_4p, number.anchors=7)
newp <- GenerateAnchor(nests=resultNest_4p, temperatures=seq(from=20,
  to=35, length.out=10))
newp <- GenerateAnchor(nests=resultNest_4p, number.anchors=7)
newp <- c(newp, Scale=1)

## End(Not run)
```

GenerateConstInc

Generate a data.frame with constant incubation temperature and incubation duration

Description

Generate a data.frame from constant incubation temperature and incubation duration

Usage

```
GenerateConstInc(durations = stop("At least one incubation length must be provided"),
  temperatures = stop("At least one incubation temperature must be provided"),
  names = NULL)
```

Arguments

durations	A vector with incubation durations
temperatures	A vector with incubation temperatures
names	A vector of column names

Details

GenerateConstInc generates a data.frame with constant incubation temperature and incubation duration

Value

A data.frame that can be used with FormatNests()

Author(s)

Marc Girondot

Examples

```
## Not run:
temp_cst <- GenerateConstInc(durations=c(150000, 100100, 100000),
  temperatures=c(28, 30.5, 30.6),
  names=c("T28", "T30.5", "T30.6"))

## End(Not run)
```

GenerateTest

Generate a data.frame that can be used as test value for searchR()

Description

Generate a data.frame that can be used as test value for searchR()

Usage

```
GenerateTest(series = stop("A result object or names of series must be provided"),
  size = NULL, previous = NULL)
```

Arguments

series	Name of series or object from searchR()
size	Size or mass at hatching. Will be recycled if necessary
previous	Previous formatted test data

Details

GenerateTest Generate a data.frame that can be used as test value for searchR()

Value

A data.frame with size or mass at hatching for each nest

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(resultNest_4p)
testsize1 <- GenerateTest(resultNest_4p)
testsize2 <- GenerateTest(series=resultNest_4p,
size=c(Mean=39.3, SD=1.92))

## End(Not run)
```

GRTRN_MHmcmc	<i>Metropolis-Hastings algorithm for Embryo Growth Rate Thermal Reaction Norm</i>
--------------	---

Description

Run the Metropolis-Hastings algorithm for data.

The number of iterations is $n.iter+n.adapt+1$ because the initial likelihood is also displayed.

I recommend that $thin=1$ because the method to estimate SE uses resampling.

If initial point is maximum likelihood, $n.adapt = 0$ is a good solution.

To get the SE of the point estimates from `result_mcmc <- GRTRN_MHmcmc(result=try)`, use:
`result_mcmc$SD`

coda package is necessary for this function.

The parameters `intermediate` and `filename` are used to save intermediate results every 'intermediate' iterations (for example 1000). Results are saved in a file named `filename`.

The parameter `previous` is used to indicate the list that has been save using the parameters `intermediate` and `filename`. It permits to continue a mcmc search.

These options are used to prevent the consequences of computer crash or if the run is very very long and processes with user limited time.

Usage

```
GRTRN_MHmcmc(result = NULL, n.iter = 10000, parametersMCMC = NULL,
n.chains = 1, n.adapt = 0, thin = 1, trace = FALSE,
batchSize = sqrt(n.iter), parallel = TRUE, intermediate = NULL,
filename = "intermediate.Rdata", previous = NULL)
```

Arguments

<code>result</code>	An object obtained after a SearchR fit
<code>n.iter</code>	Number of iterations for each step
<code>parametersMCMC</code>	A set of parameters used as initial point for searching with information on priors
<code>n.chains</code>	Number of replicates
<code>n.adapt</code>	Number of iterations before to store outputs
<code>thin</code>	Number of iterations between each stored output

trace	True or False, shows progress
batchSize	Number of observations to include in each batch fo SE estimation
parallel	If true, try to use several cores using parallel computing
intermediate	Period for saving intermediate result, NULL for no save
filename	If intermediate is not NULL, save intermediate result in this file
previous	Previous result to be continued. Can be the filename in which intermediate results are saved.

Details

GRTRN_MHmcmc runs the Metropolis-Hastings algorithm for data (Bayesian MCMC)

Value

A list with resultMCMC being mcmc.list object, resultLnL being likelihoods and parametersMCMC being the parameters used

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "T12H", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(118.768297442004, 475.750095909406, 306.243694918151,
116.055824800264), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
resultNest_4p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_4p)
pMCMC <- TRN_MHmcmc_p(resultNest_4p, accept=TRUE)
# Take care, it can be very long; several days
result_mcmc_4p <- GRTRN_MHmcmc(result=resultNest_4p,
parametersMCMC=pMCMC, n.iter=10000, n.chains = 1,
n.adapt = 0, thin=1, trace=TRUE)
data(result_mcmc_4p)
out <- as.mcmc(result_mcmc_4p)
# This out can be used with coda package
# Test for stationarity and length of chain
require(coda)
heidel.diag(out)
```

```

raftery.diag(out)
# plot() can use the direct output of GRTRN_MHmcmc() function.
plot(result_mcmc_4p, parameters=1, xlim=c(0,550))
plot(result_mcmc_4p, parameters=3, xlim=c(290,320))
# summary() permits to get rapidly the standard errors for parameters
# They are store in the result also.
se <- result_mcmc_4p$SD
# the confidence interval is better estimated by:
apply(out[[1]], 2, quantile, probs=c(0.025, 0.975))
# The use of the intermediate method is as followed;
# Here the total mcmc iteration is 10000, but every 1000, intermediate
# results are saved in file intermediate1000.Rdata:
result_mcmc_4p <- GRTRN_MHmcmc(result=resultNest_4p,
parametersMCMC=pMCMC, n.iter=10000, n.chains = 1,
n.adapt = 0, thin=1, trace=TRUE,
intermediate=1000, filename="intermediate1000.Rdata")
# If run has been stopped for any reason, it can be resumed with:
result_mcmc_4p <- GRTRN_MHmcmc(previous="intermediate1000.Rdata")

## End(Not run)

```

hist.Nests

Show the histogram of temperatures with set of nests

Description

Show the histogram of temperatures with set of nests hist(data)

Usage

```

## S3 method for class 'Nests'
hist(x, series = "all", ...)

```

Arguments

x	Data formatted using formatdata.
series	Series to be used, logical (TRUE ou FALSE), numbers or names. If "all", all series are used.
...	Parameters used by hist function

Details

hist.Nests shows the histogram of temperatures with set of nests

Value

A list with an histogram object with information on histogram or NULL if no series was selected and the complete set of temperatures used.

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
h <- hist(formatted, series="all")

## End(Not run)
```

hist.NestsResult	<i>Show the histogram of temperatures with set of nests</i>
------------------	---

Description

Show the histogram of temperatures with set of nests hist(data)

Usage

```
## S3 method for class 'NestsResult'
hist(x, series = "all", ...)
```

Arguments

x	Results obtained after searchR
series	Series to be used, logical (TRUE ou FALSE), numbers or names. If "all", all series are used.
...	Parameters used by hist function (example main="Title")

Details

hist.NestsResult shows the histogram of temperatures with set of nests

Value

A list with an histogram object with information on histogram or NULL if no series was selected and the complete set of temperatures used.

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

Examples

```
## Not run:
library(embryogrowth)
data(resultNest_4p)
h <- hist(resultNest_4p, series=c(1:5))

## End(Not run)
```

info.nests

Calulte statistics about nests

Description

Calculate statistics about nests

The `embryo.stages` is a list with stage numbers and relative size as compared to final size at the beginning of the stage.

For example `embryo.stages=list(number=21:30, size=c(8.4, 9.4, 13.6, 13.8, 18.9, 23.5, 32.2, 35.2, 35.5, 38.5)/39.33)` indicates that the stages 21 begins at the relative size of 8.4/39.33.

series can be indicated as the name of the series, their numbers or series of TRUE or FALSE. "all" indicates that all series must be analyzed.

The likelihood object is just the total likelihood of the data in the model.

The summary object is a data.frame composed of these elements:

- `TimeWeighted.temperature.mean` Average temperature weighed by the time at each temperature
- `TimeWeighted.temperature.se` Standard error for the average temperature weighed by the time at each temperature
- `TSP.TimeWeighted.temperature.mean` Average temperature during the TSP weighed by the time at each temperature
- `TSP.TimeWeighted.temperature.se` Standard error for the average temperature during the TSP weighed by the time at each temperature
- `TSP.MassWeighted.temperature.mean` Average temperature during the TSP weighed by the mass increase at each temperature
- `TSP.MassWeighted.temperature.se` Standard error for the average temperature during the TSP weighed by the mass increase at each temperature
- `TSP.STRNWeighted.temperature.mean` Averaged temperature during the TSP weighed by sexualisation thermal reaction norm
- `TSP.STRNWeighted.temperature.se` Standard error for temperature during the TSP weighed by sexualisation thermal reaction norm
- `TSP.MassWeighted.STRNWeighted.temperature.mean` Averaged temperature during the TSP weighed by sexualisation thermal reaction norm and embryo growth
- `TSP.MassWeighted.STRNWeighted.temperature.se` Standard error for temperature during the TSP weighed by sexualisation thermal reaction norm and embryo growth
- `TSP.length.mean` Average length of the TSP

- TSP.length.se Standard error for length of the TSP
- TSP.begin.mean Average time of the beginning of the TSP
- TSP.begin.se Standard error for the beginning of the TSP
- TSP.end.mean Average time of the end of the TSP
- TSP.end.se Standard error for the end of the TSP
- Incubation.length.mean Average number of days for the incubation length
- Incubation.length.se Standard error for number of days for the incubation length
- Middlethird.begin.mean Average day at which the middle third incubation begins
- Middlethird.begin.se Standard error for day at which the middle third incubation begins
- Middlethird.end.mean Average day at which the middle third incubation ends
- Middlethird.end.se Standard error at which the middle third incubation ends
- Middlethird.length.mean Average days for second third of incubation
- Middlethird.length.se Standard error for second third of incubation
- MiddleThird.TimeWeighted.temperature.mean Average temperature during the middle third of incubation
- MiddleThird.TimeWeighted.temperature.se Standard error for the temperature during the middle third of incubation
- MiddleThird.MassWeighted.temperature.mean Average temperature during the middle third of incubation weight by embryo growth
- MiddleThird.MassWeighted.temperature.se Standard error for temperature during the middle third of incubation weight by embryo growth

The metric object is a list composed of data.frames which have the following attributes for each element of the list:

- time.begin.tsp Average time at the beginning of TSP
- time.end.tsp Average time at the end of TSP
- time.begin.middlethird Average time at the beginning of the middle third of incubation
- time.end.middlethird Average time at the end of the middle third of incubation
- time.begin.tsp.se Standard error for the time at the beginning of TSP
- time.end.tsp.se Standard error for the time at the end of TSP
- time.begin.middlethird.se Standard error for the time at the beginning of the middle third of incubation
- time.end.middlethird.se Standard error for the time at the end of the middle third of incubation
- stopattest NA if stopattest was false, TRUE if at least one incubation series was longer than hatchling size and FALSE at contrary

If you indicate new set of temperatures, you must probably also indicate new test values.

Note: two species have predefined embryo stages. embryo.stages parameter can take the values:

- *Caretta caretta*.SCL
- *Emys orbicularis*.SCL
- *Emys orbicularis*.mass

The progress bar is based on replicates.

Usage

```
info.nests(NestsResult = NULL, parameters = NULL, fixed.parameters = NULL,
  SE = NULL, temperatures = NULL, derivate = NULL, test = NULL,
  stopattest = FALSE, M0 = NULL, series = "all", TSP.borders = NULL,
  embryo.stages = "Caretta caretta.SCL", replicate.CI = 1, weight = NULL,
  out = "Likelihood", fill = NULL, SexualisationTRN = NULL,
  metric.end.incubation = NULL, progress = TRUE, warnings = TRUE)
```

Arguments

NestsResult	A NestsResult file generated by searchR
parameters	A set of parameters if result is not provided.
fixed.parameters	Another set of parameters if result is not provided.
SE	Standard error for each parameter if result is not provided, or replace the one in NestsResult. Use SE=NA to remove SE from NestResult
temperatures	Timeseries of temperatures formatted using formatNests(). Will replace the one in result.
derivate	Function used to fit embryo growth: dydt.Gompertz, dydt.exponential or dydt.linear
test	Mean and SD of size of hatchlings
stopattest	TRUE or FALSE. If TRUE, the model stops when proxy of size reached the mean test size.
M0	Measure of hatchling size proxi at laying date
series	The name or number of the series to be displayed. Only one series can be displayed at a time.
TSP.borders	The limits of TSP in stages. See embryo.stages parameter.
embryo.stages	The embryo stages. At least TSP.borders stages must be provided to estimate TSP borders. See note.
replicate.CI	Number of replicates to estimate CI. If 1, no CI is estimated.
weight	Weights of the different nests to estimate likelihood
out	Can take the values of likelihood or metric or summary. metric and summary can be combined: c("metric", "summary")
fill	Number of minutes between two records. Create new one if they do not exist. NULL does not change the time of temperature recordings.
SexualisationTRN	A set of parameters used to model thermal reaction norm during TSP

<code>metric.end.incubation</code>	The expected metric at the end of incubation. Used to calibrate TSP size. If NULL, take the maximum Mean of the test parameter. If NA, use the actual final size. Can be a vector and is recycled if necessary.
<code>progress</code>	If FALSE, the progress bar is not shown (useful for using with sweave or knitr)
<code>warnings</code>	if FALSE, does not show warnings

Details

Calculate statistics about nests

Value

Return or the total likelihood or a list with `$metric` and `$summary` depending on `out` parameter

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(resultNest_4p)
summary.nests <- info.nests(resultNest_4p, out="summary", replicate.CI=20,
  SE=c(DHA=17.7357362231266, DHH=55.2002231419364,
  T12H=0.850237694629938, Rho25=8.47868153136681))
# Result is in summary.nests$summary
infoall <- info.nests(resultNest_4p)
# Result is a value
infoall.df <- info.nests(resultNest_4p, out="metric", replicate.CI=20,
  SE=c(DHA=17.7357362231266, DHH=55.2002231419364,
  T12H=0.850237694629938, Rho25=8.47868153136681)))
# Result is in summary.nests$metric
infoall.both <- info.nests(resultNest_4p, out=c("metric", "summary"), replicate.CI=20,
  SE=c(DHA=17.7357362231266, DHH=55.2002231419364,
  T12H=0.850237694629938, Rho25=8.47868153136681)))
# Result is in summary.nests$summary and in summary.nests$metric

## End(Not run)
```

likelihoodR

Estimate the likelihood of a set of parameters for nest incubation data

Description

Estimate the likelihood of a set of parameters for nest incubation data

Usage

```
likelihoodR(result = NULL, parameters = NULL, fixed.parameters = NULL,
            temperatures = NULL, derivate = NULL, test = NULL, M0 = NULL,
            hessian = FALSE, weight = NULL, parallel = TRUE, echo = TRUE)
```

Arguments

result	A object obtained after searchR or likelihoodR
parameters	A set of parameters
fixed.parameters	A set of parameters that will not be changed
temperatures	Timeseries of temperatures
derivate	Function used to fit embryo growth: dydt.Gompertz, dydt.exponential or dydt.linear
test	Mean and SD of size of hatchlings
M0	Measure of hatchling size or mass proxi at laying date
hessian	If TRUE, the hessian matrix is estimated and the SE of parameters estimated.
weight	A named vector of the weight for each nest for likelihood estimation
parallel	If true, try to use several cores using parallel computing.
echo	If FALSE, does not display the result.

Details

likelihoodR estimates the likelihood of a set of parameters for nest incubation data

Value

A result object

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
# K for Gompertz must be set as fixed parameter or being a constant K
# or relative to the hatchling size rK
x <- structure(c(118.768297442004, 475.750095909406, 306.243694918151,
116.055824800264), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
```

```
pfixed <- c(rK=2.093313)
# K or rK are not used for dydt.linear or dydt.exponential
LresultNest_4p <- likelihoodR(parameters=x, fixed.parameters=pfixed,
  temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
  test=c(Mean=39.33, SD=1.92))
data(resultNest_4p)
LresultNest_4p <- likelihoodR(result=resultNest_4p)

## End(Not run)
```

logLik.NestsResult *Return Log Likelihood of a fit generated by searchR*

Description

Return Log Likelihood of a fit generated by searchR

Usage

```
## S3 method for class 'NestsResult'
logLik(object, ...)
```

Arguments

object	A result file generated by searchR
...	Not used

Details

logLik.NestsResult Return Log Likelihood of a fit

Value

The Log Likelihood value of the fitted model and data

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(resultNest_4p)
logLik(resultNest_4p)
AIC(resultNest_4p)

## End(Not run)
```

`logLik.tsd`*Return Log Likelihood of a fit generated by tsd*

Description

Return Log Likelihood of a fit generated by tsd

Usage

```
## S3 method for class 'tsd'  
logLik(object, ...)
```

Arguments

<code>object</code>	A result file generated by tsd
<code>...</code>	Not used

Details

logLik.tsd Return Log Likelihood of a fit

Value

The Log Likelihood value of the fitted model and data

Author(s)

Marc Girondot

Examples

```
## Not run:  
library(embryogrowth)  
m <- c(10, 14, 7, 4, 3, 0, 0)  
f <- c(0, 1, 2, 4, 15, 10, 13)  
t <- c(25, 26, 27, 28, 29, 30, 31)  
result <- tsd(males=m, females=f, temperatures=t)  
logLik(result)  
AIC(result)  
  
## End(Not run)
```

MovingIncubation *Simulate incubation of a nest with the beginning of incubation varying*

Description

Simulate incubation of a nest with the beginning varying day by day
 Temperatures must be in a data.frame with one column (Time) being the time and the second the temperatures (Temperature). A third column can indicate the temperature at the end of incubation (Temperature.end.incubation). Do not use FormatNests() for this dataframe.

Usage

```
MovingIncubation(NestsResult = NULL,
  temperatures.df = stop("A data.frame with timeseries of temperatures must be provided"),
  average.incubation.duration = 60 * 1440, skip = 1, parameters = NULL,
  fixed.parameters = NULL, SE = NULL, derivate = NULL, test = NULL,
  M0 = NULL, TSP.borders = c(21, 26),
  embryo.stages = "Caretta caretta.SCL", SexualisationTRN = NULL,
  replicate.CI = 1, progress = TRUE)
```

Arguments

NestsResult	A result file generated by searchR
temperatures.df	A data.frame with 2 or 3 columns: Times, Temperatures and Temperatures.end.incubation (facultative)
average.incubation.duration	The average time to complete incubation
skip	Number of data to skip between two runs
parameters	A set of parameters if result is not provided.
fixed.parameters	Another set of parameters if result is not provided.
SE	Standard error for each parameter if not present in result is not provided (not used still)
derivate	Function used to fit embryo growth: dydt.Gompertz, dydt.exponential or dydt.linear
test	Mean and SD of size of hatchlings as a vector ie test=c(Mean=xx, SD=yy)
M0	Measure of hatchling size proxi at laying date
TSP.borders	The limits of TSP
embryo.stages	The embryo stages. At least TSP.borders stages must be provided to estimate TSP length
SexualisationTRN	A set of parameters used to model thermal reaction norm during TSP
replicate.CI	Number of randomizations to estimate CI
progress	If TRUE, progression bar is shown (deactivate for sweave or knitr)

Details

MovingIncubation simulate incubation of a nest with the beginning varying day by day

Value

A dataframe with informations about thermosensitive period length and incubation length day by day of incubation

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(resultNest_4p)
ti <- seq(from=0, to=(60*24*100), by=60)
temp <- rnorm(length(ti), 29, 5)
temp <- temp+ti/(60*24*100)/2
layout(mat=c(1, 2))
parpre <- par(mar=c(4, 4, 1, 1)+0.4)
plot(ti/(60*24), temp, type="l", xlab="Days", ylab="Nest temperature", bty="n", las=1)
out <- MovingIncubation(NestsResult=resultNest_4p,
  temperatures.df=data.frame(Time=ti, Temperature=temp))
with(out, plot(Time/(60*24), Incubation.length.mean/(60*24), xlab="Days along the season",
  ylab="Incubation duration", type="l", bty="n", las=1, ylim=c(74, 78)))
par(mar=parpre)
layout(mat=c(1))

## End(Not run)
```

nest

Timeseries of temperatures for nests

Description

Timeseries of temperatures for nests

Usage

nest

Format

A dataframe with raw data

Details

Timeseries of temperatures for nests

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M. & Kaska, Y. 2014. A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*. 45, 96-102.

Examples

```
## Not run:  
library(embryogrowth)  
data(nest)  
  
## End(Not run)
```

plot.NestsResult *Plot the embryo growth*

Description

Plot the embryo growth from one or several nests.

The embryo.stages is a list with stage numbers and relative size as compared to final size at the beginning of the stage.

For example for *Caretta caretta*, embryo.stages=list(number=21:30, size=c(8.4, 9.4, 13.6, 13.8, 18.9, 23.5, 32.2, 35.2, 35.5, 38.5)/39.33) indicates that the stages 21 begins at the relative size of 8.4/39.33.

The default is for the turtle "*Caretta caretta*".

Series can be indicated as the name of the series, its number or succession of TRUE or FALSE. "all" indicates that all series must be printed.

show.floritures parameter does not affect show.test option.

Note: two species have predefined embryo stages. embryo.stages parameter can take the values:

- *Caretta caretta*.SCL
- *Emys orbicularis*.SCL
- *Emys orbicularis*.mass

Usage

```
## S3 method for class 'NestsResult'
plot(x, ..., parameters = NULL,
     fixed.parameters = NULL, SE = NULL, temperatures = NULL,
     derivate = NULL, test = NULL, stopattest = FALSE, M0 = NULL,
     series = "all", TSP.borders = NULL,
     embryo.stages = "Caretta caretta.SCL", replicate.CI = 100,
     weight = NULL, metric.end.incubation = NULL, col.stages = "blue",
     col.PT = "red", col.TSP = "gray", col.temperatures = "green",
     col.S = "black", lty.temperatures = 1, lwd.temperatures = 2,
     ylimT = NULL, ylimS = NULL, xlim = NULL, show.stages = TRUE,
     show.TSP = TRUE, show.third = TRUE, show.CI = TRUE,
     show.metric = TRUE, show.fioritures = TRUE, show.temperatures = TRUE,
     show.PT = TRUE, PT = c(mean = NA, SE = NA), show.test = TRUE,
     add = FALSE, lab.third = "2nd third of incubation", at.lab.third = 4,
     lab.PT = "PT", lab.stages = "Stages", mar = c(4, 5, 4, 5) + 0.3,
     xlab = "Days of incubation", ylabT = expression("Temperatures in " *
     degree * "C"), ylabS = "Embryo metric", progress = TRUE)
```

Arguments

x	A result file generated by searchR
...	Parameters for plot()
parameters	A set of parameters if result is not provided.
fixed.parameters	Another set of parameters if result is not provided.
SE	Standard error for each parameter if result is not provided, or replace the one in NestsResult. Use SE=NA to remove SE from NestResult
temperatures	Timeseries of temperatures formatted using formatNests(). Will replace the one in result.
derivate	Function used to fit embryo growth: dydt.Gompertz, dydt.exponential or dydt.linear
test	Mean and SD of size of hatchlings
stopattest	TRUE or FALSE. If TRUE, the model stops when proxy of size reached the mean test size.
M0	Measure of hatchling size proxi at laying date
series	The name or number of the series to be displayed. Only one series can be displayed at a time.
TSP.borders	The limits of TSP in stages. See embryo.stages parameter.
embryo.stages	The embryo stages. At least TSP.borders stages must be provided to estimate TSP borders. See note.
replicate.CI	Number of replicates to estimate CI. If 1, no CI is estimated.
weight	Weights of the different nests to estimate likelihood

metric.end.incubation	The expected metric at the end of incubation. Used to calibrate TSP size. If NULL, take the maximum Mean of the test parameter. If NA, use the actual final size. Can be a vector and is recycled if necessary.
col.stages	The color of the stages
col.PT	The color of the pivotal temperature
col.TSP	The color of the TSP
col.temperatures	The color of the temperatures
col.S	The color of the size or mass. Can be a vector (useful when series="all" option).
lty.temperatures	Type of line for temperatures
lwd.temperatures	Width of line for temperatures
ylimT	Range of temperatures to be displayed
ylimS	Range of size to be displayed
xlim	Range of incubation days to be displayed
show.stages	TRUE or FALSE, does the embryo stages should be displayed?
show.TSP	TRUE or FALSE, does the TSP borders should be displayed?
show.third	TRUE or FALSE, does the first and second third borders should be displayed?
show.CI	TRUE or FALSE, do the confidence intervals should be displayed?
show.metric	TRUE or FALSE, does the plot of embryo metric is shown?
show.fioritures	If FALSE, set show.PT, show.temperatures, show.stages, show.TSP, show.third, show.CI to FALSE
show.temperatures	TRUE or FALSE, does the temperatures should be displayed?
show.PT	TRUE or FALSE, does the pivotal temperature should be displayed?
PT	Value for pivotal temperature, mean and SE
show.test	TRUE or FALSE, does the hatchling size should be displayed
add	If TRUE, all the curves are shown on the same graph
lab.third	Label for 2nd third of incubation
at.lab.third	Position of Label for 2nd third of incubation [default=4]
lab.PT	Label for Pivotal Temperature
lab.stages	Label for Stages
mar	Parameter mar used for plot
xlab	Label for axis
ylabT	Label for temperature axis
ylabS	Label for size axis
progress	If FALSE, the progress bar is not shown (useful for use with sweave or knitr)
NestsResult	A NestsResult file generated by searchR

Details

plot.NestsResult Plot the embryo growth

Value

Nothing

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(resultNest_4p)
plot(resultNest_4p, xlim=c(0,70), ylimT=c(22, 32), ylimS=c(0,45), series=1,
SE=c(DHA=1.396525, DHH=4.101217, T12H=0.04330405, Rho25=1.00479))
# to plot all the nest at the same time, use
plot(resultNest_4p, xlim=c(0,70), ylimT=c(22, 32), ylimS=c(0,45),
series="all", show.fioritures=FALSE, add=TRUE)
# to use color different for series
plot(resultNest_4p, xlim=c(0,70), ylimT=c(22, 32), ylimS=c(0,45), add=TRUE,
series="all", show.fioritures=FALSE, col.S=c(rep("black", 5), rep("red", 6)))

## End(Not run)
```

plot.tsd

*Plot results of tsd() that best describe temperature-dependent sex de-
termination*

Description

Plot the estimates that best describe temperature-dependent sex determination.

Usage

```
## S3 method for class 'tsd'
plot(x, ..., males.freq = TRUE, las.x = 1, las.y = 1,
lab.PT = "Pivotal temperature",
lab.TRT = paste0("Transitional range of temperatures l=", l * 100, "%"),
col.TRT = "gray", col.TRT.CI = rgb(0.8, 0.8, 0.8, 0.5),
col.PT.CI = rgb(0.8, 0.8, 0.8, 0.5), show.CI = TRUE)
```

Arguments

x	A result file generated by tsd()
...	Parameters for plot()
males.freq	Should the graph uses males frequency [TRUE] or females [FALSE]
las.x	las parameter for x axis
las.y	las parameter for y axis
lab.PT	Label to describe pivotal temperature
lab.TRT	Label to describe transitional range of temperature
col.TRT	The color of TRT
col.TRT.CI	The color of CI of TRT based on range.CI
col.PT.CI	The color of CI of PT based on range.CI
show.CI	Do the CI for the curve should be shown

Details

plot.tsd plot result of tsd() that best describe temperature-dependent sex determination

Value

Nothing

Author(s)

Marc Girondot

References

Girondot, M. 1999. Statistical description of temperature-dependent sex determination using maximum likelihood. *Evolutionary Ecology Research*, 1, 479-486.

Godfrey, M.H., Delmas, V., Girondot, M., 2003. Assessment of patterns of temperature-dependent sex determination using maximum likelihood model selection. *Ecoscience* 10, 265-272.

Hulin, V., Delmas, V., Girondot, M., Godfrey, M.H., Guillon, J.-M., 2009. Temperature-dependent sex determination and global change: are some species at greater risk? *Oecologia* 160, 493-506.

Girondot M., Submitted. On the concept of embryological thermosensitive period for sex determination in reptiles.

Examples

```
## Not run:
CC_AtlanticSW <- subset(STSRE_TSD, RMU=="Atlantic, SW" &
                       Species=="Caretta caretta" & Sexed!=0)
tsdL <- with (CC_AtlanticSW, tsd(males=Males, females=Females,
                               temperatures=Incubation.temperature-Correction.factor,
                               equation="logistic"))

plot(tsdL)

## End(Not run)
```

plotR

Show the fitted growth rate dependent on temperature

Description

To show the growth rate, the syntax is:
`plotR(result=res)`

Usage

```
plotR(result = NULL, parameters = NULL, fixed.parameters = NULL,
      col = "black", legend = NA, SE = NULL, set.par = 1, size = NA,
      xlim = c(20, 35), scaleY = "auto", lty = 1, ltyCI = 3, lwd = 1,
      lwdCI = 1, xlimR = xlim, replicate.CI = 100, show.box = TRUE,
      local.box = "topleft", ...)
```

Arguments

<code>result</code>	A result object or a list of result objects
<code>parameters</code>	Indicate some parameters if the result object is not supplied
<code>fixed.parameters</code>	Indicate some parameters if the result object is not supplied
<code>col</code>	The color to use for a list of colors if several results
<code>legend</code>	Text to show in bottom right legend or a list of text if several results
<code>SE</code>	The standard error for the parameters or a list of SE if several results. Use NA to force not use SE
<code>set.par</code>	1 or 2 or a list of 1 or 2 to designate with set of parameters to show
<code>size</code>	If indicated, will show the growth rate for this size
<code>xlim</code>	Range of values for x-axis
<code>scaleY</code>	Scaling factor for y axis or "auto"
<code>lty</code>	The type of line to use if several results as a list
<code>ltyCI</code>	The type of line to use for confidence interval as a list
<code>lwd</code>	The type of line to use if several results as a list
<code>lwdCI</code>	The type of line to use for confidence interval as a list
<code>xlimR</code>	Range of values to be displayed for R curve; can be a list if a list of results is used
<code>replicate.CI</code>	Number of randomizations to estimate CI
<code>show.box</code>	If TRUE show a box with "mean" and "confidence interval"
<code>local.box</code>	Position of the box with "mean" and "confidence interval", default="topleft"
<code>...</code>	Parameters for <code>plot()</code> such as <code>main=</code> or <code>ylim=</code>

Details

plotR shows the fitted growth rate dependent on temperature

Value

Nothing

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(118.768297442004, 475.750095909406, 306.243694918151,
116.055824800264), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
resultNest_4p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_4p)
pMCMC <- TRN_MHmcmc_p(resultNest_4p, accept=TRUE)
# Take care, it can be very long, sometimes several days
result_mcmc_4p_80 <- GRTRN_MHmcmc(result=resultNest_4p,
parametersMCMC=pMCMC, n.iter=10000, n.chains = 1, n.adapt = 0,
thin=1, trace=TRUE)
data(result_mcmc_4p)
plotR(result=resultNest_4p, SE=result_mcmc_4p$TimeSeriesSE,
ylim=c(0,0.3))
x <- structure(c(115.758929130522, 428.649022170996, 503.687251738993,
12.2621455821612, 306.308841227278, 116.35048615105), .Names = c("DHA",
"DHH", "DHL", "DT", "T12L", "Rho25"))
plotR(parameters=x, xlim=c(20,35))
pfixed <- c(rK=2.093313)
resultNest_6p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_6p)
plotR(list(resultNest_4p, resultNest_6p),
col=c("black", "red"), legend=c("4 parameters", "6 parameters"))
#####
# new formulation of parameters
data(resultNest_newp)
plotR(resultNest_newp)
```

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

plotR_hist	<i>Shows the histogram of temperatures with set of nests and the R function superimpose</i>
------------	---

Description

Shows the histogram of temperatures with set of nests and the R function superimpose `plotR_hist(data)`

Usage

```
plotR_hist(x, ..., ylimH = NULL, ylabH = "Frequency of temperatures")
```

Arguments

x	Result data
...	Parameters used by hist or plotR functions
ylimH	Scale of histogram using <code>ylimH=c(min, max)</code>
ylabH	Label for histogram scale

Details

`plotR_hist` shows the histogram of temperatures with set of nests and the R function superimpose

Value

Nothing

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

Examples

```
## Not run:  
library(embryogrowth)  
data(resultNest_4p)  
plotR_hist(resultNest_4p)  
  
## End(Not run)
```

plot_transition	<i>Show fonction used for transition</i>
-----------------	--

Description

Plot the transition function

Usage

```
plot_transition(result = NULL, parameters = NULL, sizes = c(0, 40), ...)
```

Arguments

result	A result object
parameters	Set of parameters. If both result and parameters are indicated, parameters have priority.
sizes	The range of possible sizes
...	Parameters for plot() such as main= or ylim=

Details

plot_transition show fonction used for transition

Value

Nothing

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
data(resultNest_4p)
# Get a set of parameters without transition
x1 <- resultNest_4p$par
# Generate a set of parameters with transition
x2 <- switch.transition(x1)
x2 <- x2[names(x2)!="transition_P"]
x2["transition_S"] <- 4
pfixed <- c(rK=2.093313, transition_P=20)
resultNest_4p_transition <- searchR(parameters=x2, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
```

```

data(resultNest_4p_transition)
# show the model for smallest size
plotR(resultNest_4p_transition, ylim=c(0,0.3))
# show the model for larger sizes
plotR(resultNest_4p_transition, set.par=2, ylim=c(0,0.3))
# plot model for both together
plotR(resultNest_4p_transition, set.par=c(1,2), ylim=c(0,0.3),
      col=c("red", "black"), legend=list("Initial", "End"))
plot_transition(result=resultNest_4p_transition, las=1, sizes=c(0,40))
compare_AIC(one.model=list(resultNest_4p), two.models=list(resultNest_4p_transition))
# Note that the model with fitted transition_P is trivial. Embryos grow fast until
# they reach hatchling size and then growth rate becomes null!

## End(Not run)

```

predict.tsd

Estimate sex ratio according to constant incubation temperature

Description

Estimate sex ratio according to constant incubation temperature

The data.frame has the following components:

temperatures, SE, sexratio, CI.minus.sexratio, CI.plus.sexratio, range.CI

Note that incubation duration is a very bad proxy for sex ratio. See Georges, A., Limpus, C. J. & Stoutjesdijk, R. 1994. Hatchling sex in the marine turtle *Caretta caretta* is determined by proportion of development at a temperature, not daily duration of exposure. *J. Exp. Zool.*, 270, 432-444.

Usage

```

## S3 method for class 'tsd'
predict(object, temperatures = NULL, durations = NULL,
      SE = NULL, range.CI = 0.95, replicates = 1000, progressbar = FALSE,
      ...)

```

Arguments

object	A result file generated by tsd
temperatures	A vector of temperatures
durations	A vector of durations
SE	The standard error for temperatures or durations
range.CI	The range of confidence interval for estimation, default=0.95
replicates	Number of replicates to estimate CI
progressbar	Logical. Does a progression bar must be shown
...	Not used

Details

predict.tsd Estimate sex ratio according to constant incubation temperature

Value

A data.frame with informations about sex-ratio

Author(s)

Marc Girondot

See Also

Other Functions.for.temperature.dependent.sex.determination: [STSRE_TSD](#); [TSP.list](#); [tsd_MHmcmc_p](#); [tsd_MHmcmc](#); [tsd](#)

Examples

```
## Not run:
library(embryogrowth)
m <- c(10, 14, 7, 4, 3, 0, 0)
f <- c(0, 1, 2, 4, 15, 10, 13)
t <- c(25, 26, 27, 28, 29, 30, 31)
result <- tsd(males=m, females=f, temperatures=t)
plot(result)
predict(result, temperatures=c(25, 31))
predict(result, temperatures=c(25, 31), SE=c(1, 2))
d <- c(72, 70, 65, 63, 62, 60, 59)
result <- tsd(males=m, females=f, durations=d)
predict(result, durations=c(67, 68))

## End(Not run)
```

resultNest_4p

Fit using the nest database

Description

Fit using the nest database

Usage

```
resultNest_4p
```

Format

A list with fitted information about data(nest)

Details

Result of the fit using the nest database

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M., & Kaska, Y. (2014). A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*, 45, 96-102. doi: 10.1016/j.jtherbio.2014.08.005

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(118.431040984352, 498.205702157603, 306.056280989839,
118.189669472381), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
resultNest_4p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))

## End(Not run)
```

resultNest_4p_transition

Result of the fit using the nest database using transition

Description

Fit using the nest database using transition

Usage

```
resultNest_4p_transition
```

Format

A list with fitted information about data(nest)

Details

Result of the fit using the nest database using transition

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M., & Kaska, Y. (2014). A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*, 45, 96-102. doi: 10.1016/j.jtherbio.2014.08.005

Examples

```
## Not run:  
library(embryogrowth)  
data(nest)  
formatted <- FormatNests(nest)  
data(resultNest_4p_transition)  
  
## End(Not run)
```

resultNest_4p_weight *Fit using the nest database with weight*

Description

Fit using the nest database with weight

Usage

```
resultNest_4p_weight
```

Format

A list with fitted information about data(nest)

Details

Result of the fit using the nest database with weight

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M., & Kaska, Y. (2014). A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*, 45, 96-102. doi: 10.1016/j.jtherbio.2014.08.005

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
w <- weightmaxentropy(formatted, control_plot=list(xlim=c(20,36)))
x <- structure(c(118.768297442004, 475.750095909406, 306.243694918151,
116.055824800264), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
# K or rK are not used for dydt.linear or dydt.exponential
resultNest_4p_weight <- searchR(parameters=x,
fixed.parameters=pfixed, temperatures=formatted,
derivate=dydt.Gompertz, M0=1.7, test=c(Mean=39.33, SD=1.92),
method = "BFGS", weight=w)
data(resultNest_4p_weight)
plotR(resultNest_4p_weight, ylim=c(0,0.50), xlim=c(15, 35))

## End(Not run)
```

resultNest_6p

Fit using the nest database

Description

Fit using the nest database

Usage

```
resultNest_6p
```

Format

A list with fitted information about data(nest)

Details

Result of the fit using the nest database

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M., & Kaska, Y. (2014). A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*, 45, 96-102. doi: 10.1016/j.jtherbio.2014.08.005

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(115.770032186805, 428.649144244891, 503.716700005443,
12.2641939339981, 306.305498098871, 116.380086027401),
.Names = c("DHA", "DHH", "DHL", "DT", "T12L", "Rho25"))
pfixed <- c(rK=2.093313)
resultNest_6p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))

## End(Not run)
```

resultNest_newp

Fit using the nest database with anchored parameters

Description

Fit using the nest database with anchored parameters

Usage

```
resultNest_newp
```

Format

A list with fitted information from data(nest) with anchored parameters

Details

Result of the fit using the nest database with anchored parameters

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M., & Kaska, Y. (2014). A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*, 45, 96-102. doi: 10.1016/j.jtherbio.2014.08.005

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
newp <- GenerateAnchor(nests=formatted, number.anchors=7)
pfixed <- c(rK=2.093313)
resultNest_newp <- searchR(parameters=newp, fixed.parameters=pfixed,
  temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
  test=c(Mean=39.33, SD=1.92))
data(resultNest_newp)
plotR(resultNest_newp)

## End(Not run)
```

result_mcmc_4p

Result of the mcmc using the nest database

Description

Fit using the nest database

Usage

```
result_mcmc_4p
```

Format

A list of class `mcmcComposite` with mcmc result for `data(nest)` with 4 parameters and Gompertz model of growth

Details

Result of the mcmc using the nest database

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M., & Kaska, Y. (2014). A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*, 45, 96-102. doi: 10.1016/j.jtherbio.2014.08.005

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(118.431040984352, 498.205702157603, 306.056280989839,
118.189669472381), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
resultNest_4p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_4p)
pMCMC <- TRN_MHmcmc_p(resultNest_4p, accept=TRUE)
# Take care, it can be very long, sometimes several days
result_mcmc_4p <- GRTRN_MHmcmc(result=resultNest_4p,
parametersMCMC=pMCMC, n.iter=10000, n.chains = 1, n.adapt = 0,
thin=1, trace=TRUE)
data(result_mcmc_4p)
plot(result_mcmc_4p, parameters="T12H", main="", xlim=c(290, 320), bty="n")
plotR(resultNest_4p, SE=result_mcmc_4p$SD, ylim=c(0,0.4), las=1)

## End(Not run)
```

result_mcmc_4p_weight *Result of the mcmc using the nest database*

Description

Fit using the nest database

Usage

```
result_mcmc_4p_weight
```

Format

A list of class mcmcComposite with mcmc result for data(nest) with 4 parameters and Gompertz model of growth weighted to maximized entropy

Details

Result of the mcmc using the nest database with weight

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M., & Kaska, Y. (2014). A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*, 45, 96-102. doi: 10.1016/j.jtherbio.2014.08.005

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
w <- weightmaxentropy(formatted, control_plot=list(xlim=c(20,36)))
x <- structure(c(118.768297442004, 475.750095909406, 306.243694918151,
116.055824800264), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
# K or rK are not used for dydt.linear or dydt.exponential
resultNest_4p_weight <- searchR(parameters=x,
fixed.parameters=pfixed, temperatures=formatted,
derivate=dydt.Gompertz, M0=1.7, test=c(Mean=39.33, SD=1.92),
method = "BFGS", weight=w)
data(resultNest_4p_weight)
pMCMC <- TRN_MHmcmc_p(resultNest_4p_weight, accept=TRUE)
# Take care, it can be very long, sometimes several days
result_mcmc_4p_weight <- GRTRN_MHmcmc(result=resultNest_4p_weight,
parametersMCMC=pMCMC, n.iter=10000, n.chains = 1, n.adapt = 0,
thin=1, trace=TRUE)
data(result_mcmc_4p_weight)
plot(result_mcmc_4p_weight, parameter="T12H", main="", xlim=c(290, 320), bty="n")
plotR(resultNest_4p_weight, SE=result_mcmc_4p_weight$SD,
ylim=c(0,0.3), las=1)
data(resultNest_4p)
data(result_mcmc_4p)
par(xpd=TRUE)
plotR(list(resultNest_4p_weight, resultNest_4p),
SE=list(result_mcmc_4p_weight$SD, result_mcmc_4p$SD),
ylim=c(0,0.4), las=1, col=list("red", "black"),
legend=list("Maximum entropy weighted", "Not weighted"))

## End(Not run)
```

result_mcmc_6p

Result of the mcmc using the nest database

Description

Fit using the nest database

Usage

```
result_mcmc_6p
```

Format

A list of class `mcmcComposite` with `mcmc` result for `data(nest)` with 6 parameters and Gompertz model of growth

Details

Result of the `mcmc` using the `nest` database

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M., & Kaska, Y. (2014). A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*, 45, 96-102. doi: 10.1016/j.jtherbio.2014.08.005

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(115.758929130522, 428.649022170996, 503.687251738993,
12.2621455821612, 306.308841227278, 116.35048615105), .Names = c("DHA",
"DHH", "DHL", "DT", "T12L", "Rho25"))
pfixed <- c(rK=2.093313)
resultNest_6p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_6p)
pMCMC <- TRN_MHmcmc_p(resultNest_6p, accept=TRUE)
# Take care, it can be very long, sometimes several days
result_mcmc_6p <- GRTRN_MHmcmc(result=resultNest_6p,
parametersMCMC=pMCMC, n.iter=10000, n.chains = 1, n.adapt = 0,
thin=1, trace=TRUE)
data(result_mcmc_6p)
plot(result_mcmc_6p, parameters="T12L", main="", xlim=c(290, 320), bty="n")

## End(Not run)
```

result_mcmc_newp	<i>Result of the mcmc using the nest database with anchored parameters</i>
------------------	--

Description

Fit using the nest database with anchored parameters

Usage

```
result_mcmc_newp
```

Format

A list of class mcmcComposite with mcmc result for data(nest) with anchored parameters

Details

Result of the mcmc using the nest database with anchored parameters

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M., & Kaska, Y. (2014). A model to predict the thermal reaction norm for the embryo growth rate from field data. *Journal of Thermal Biology*, 45, 96-102. doi: 10.1016/j.jtherbio.2014.08.005

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
newp <- GenerateAnchor(nests=formatted, number.anchors=7)
pfixed <- c(rK=2.093313)
resultNest_newp <- searchR(parameters=newp, fixed.parameters=pfixed,
  temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
  test=c(Mean=39.33, SD=1.92))
data(resultNest_newp)
pMCMC <- TRN_MHmcmc_p(resultNest_newp, accept=TRUE)
# Take care, it can be very long, sometimes several days
result_mcmc_newp <- GRTRN_MHmcmc(result=resultNest_newp,
  parametersMCMC=pMCMC, n.iter=1000, n.chains = 1, n.adapt = 0,
  thin=1, trace=TRUE)
data(result_mcmc_newp)
data(resultNest_4p)
newp <- GenerateAnchor(nests=resultNest_4p, number.anchors=7)
# Here the confidence interval is built based on anchored parameters
```

```

plotR_hist(resultNest_4p, parameters=newp, SE=result_mcmc_newp$SD,
  ylim=c(0,0.4), ylimH=c(0,0.4))
# Here the confidence interval is built based on parametric SSM equation
data(result_mcmc_4p)
plotR_hist(resultNest_4p, SE=result_mcmc_4p$SD,
  ylim=c(0,0.4), ylimH=c(0,0.4))
plot(result_mcmc_newp, las=1, xlim=c(0,30), parameters="294",
breaks=c(0, 1.00095, 2.0009, 3.00085, 4.0008, 5.00075, 6.0007, 7.00065, 8.0006, 9.00055,
10.0005, 11.00045, 12.0004, 13.00035, 14.0003, 15.00025, 16.0002, 17.00015, 18.0001,
19.00005, 20))
plot(result_mcmc_newp, las=1, xlim=c(0,30), parameters="296.333333333333")
plot(result_mcmc_newp, las=1, xlim=c(0,30), parameters=3)
# Confidence interval based on quantiles
plotR_hist(resultNest_4p, parameters=newp, SE=NULL,
  ylim=c(0,0.4), ylimH=c(0,0.4))
CI <- apply(result_mcmc_newp$resultMCMC[[1]], 2, quantile, probs=c(0.025, 0.5, 0.975))
plot_add(as.numeric(colnames(CI))-273.15, CI[1,], lty=2, type="l")
plot_add(as.numeric(colnames(CI))-273.15, CI[2,], lty=2, type="l")

## End(Not run)

```

searchR

Fit the parameters that best represent nest incubation data.

Description

Fit the parameters that best represent data.

test can be a list with two elements Mean and SD and each element is a named vector with the nest name.

Usage

```

searchR(parameters = stop("Initial set of parameters must be provided"),
  fixed.parameters = NULL,
  temperatures = stop("Formatted temperature must be provided !"),
  derivate = dydt.Gompertz, test = c(Mean = 39.33, SD = 1.92), M0 = 1.7,
  saveAtMaxiter = FALSE, fileName = "intermediate", weight = NULL,
  hessian = TRUE, control = list(trace = 1, REPORT = 100, maxit = 500))

```

Arguments

parameters	A set of parameters used as initial point for searching
fixed.parameters	A set of parameters that will not be changed
temperatures	Timeseries of temperatures after formatted using FormatNests()
derivate	Function used to fit embryo growth: dydt.Gompertz, dydt.exponential or dydt.linear
test	A vector with Mean and SD of size of hatchlings, ex. test=c(Mean=39, SD=3)

M0	Measure of hatchling size or mass proxy at laying date
saveAtMaxiter	If True, each time number of iteration reach maxiter, current data are saved in file with filename name
fileName	The intermediate results are saved in file with fileName.Rdata name
weight	A named vector of the weight for each nest for likelihood estimation
hessian	If TRUE, the hessian matrix is estimated and the SE of parameters estimated.
control	List for control parameters for optimx

Details

searchR fits the parameters that best represent nest incubation data.

Value

A result object

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "DT", "DHA", "DHH", "DHL", "Rho25"
# K for Gompertz must be set as fixed parameter or being a constant K
# or relative to the hatchling size rK
x <- structure(c(106.59891311201, 614.181133951497, 306.267053513175,
120.327257089974), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
resultNest_4p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_4p)
plot(resultNest_4p, xlim=c(0,70), ylimT=c(22, 32), ylimS=c(0,45), series=1)
x <- structure(c(106.567809092008, 527.359011254683, 614.208632495199,
2720.94506457237, 306.268259715624, 120.336791245212), .Names = c("DHA",
"DHH", "DHL", "DT", "T12L", "Rho25"))
pfixed <- c(rK=2.093313)
resultNest_6p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_6p)
pMCMC <- TRN_MHmcmc_p(resultNest_6p, accept=TRUE)
```



```

# Take care, it can be very long, sometimes several days
result_mcmc_6p <- GRTRN_MHmcmc(result=resultNest_6p,
parametersMCMC=pMCMC, n.iter=10000, n.chains = 1, n.adapt = 0,
thin=1, trace=TRUE)
data(result_mcmc_6p)
# compare_AIC() is a function from the package "HelpersMG"
compare_AIC(test1=resultNest_4p, test2=resultNest_6p)
##### with new parametrization
data(resultNest_4p)
x0 <- resultNest_4p$par
t <- hist(resultNest_4p, plot=FALSE)
x <- c(3.4, 3.6, 5.4, 5.6, 7.6, 7.5, 3.2)
names(x) <- seq(from=range(t$temperatures)[1], to=range(t$temperatures)[2],
length.out=7)
newx <- ChangeSSM(temperatures = (200:350)/10, parameters = x0,
initial.parameters = x,
control=list(maxit=5000))
pfixed <- c(rK=2.093313)
resultNest_newp <- searchR(parameters=newx, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
plotR_hist(resultNest_newp, ylim=c(0,0.3), xlimR=c(23, 34), ylimH=c(0, 0.3))
compare_AIC(test4p=resultNest_4p,
test6p=resultNest_6p,
testAnchor=resultNest_newp)

## End(Not run)

```

STRN

Estimate the parameters that best describe the sexualisation thermal reaction norm within the TSP

Description

Estimate the parameters that best describe the sexualisation thermal reaction norm within the TSP.

The Temperatures parameter is a character string which can be:

- TimeWeighted.temperature.mean
- TSP.TimeWeighted.temperature.mean
- TSP.MassWeighted.temperature.mean
- TSP.STRNWeighted.temperature.mean
- TSP.MassWeighted.STRNWeighted.temperature.mean
- MiddleThird.TimeWeighted.temperature.mean

Usage

```
STRN(Initial_STRN = NULL,
     EmbryoGrowthTRN = stop("Embryo Growth Thermal Reaction Norm must be provided"),
     tsd = stop("A result from the function tsd() must be provided"),
     Sexed = NULL, Males = NULL, Females = NULL,
     Temperatures = "TSP.MassWeighted.STRNWeighted.temperature.mean", ...)
```

Arguments

Initial_STRN	Values for initial model of Sexualisation Thermal Reaction Norm
EmbryoGrowthTRN	The Embryo growth Thermal Reaction Normal obtained with searchR()
tsd	The model used to predict sex ratio, obtained from tsd()
Sexed	The number of sexed embryos
Males	The number of males embryos
Females	The number of females embryos
Temperatures	The temperature from out of info.nests to be used
...	Parameters used for optim()

Details

STRN estimates the parameters that best describe the sexualisation thermal reaction norm within the TSP

Value

The list with object return by optim()

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
MedIncubation_Cc <- subset(STSRE_TSD, Species=="Caretta caretta" &
  RMU=="Mediterranean" & Sexed!=0)
Med_Cc <- with(MedIncubation_Cc, tsd(males=Males, females=Females,
  temperatures=Incubation.temperature, par=c(P=29, S=-0.01)))
plot(Med_Cc, xlim=c(25, 35))
# Initial_STRN <- rep(1, 7)
# names(Initial_STRN) <- as.character(seq(from=20, to=35, length=7))
Initial_STRN <- structure(c(1, 143.248982215757, -25.7029976477549, -0.00489843027318209,
-8.94560833594928, 135.781961273868, 71.2176230826628),
.Names = c("20", "22.5", "25", "27.5", "30", "32.5", "35"))
males <- c(7, 0, 0, 0, 0, 5, 6, 3, 5, 3, 2, 3, 0, 0, 0, 0, 0, 0, 0)
sexed <- rep(10, length(males))
```

```

fitSTRN <- STRN(Initial_STRN, EmbryoGrowthTRN=resultNest_4p, tsd=Med_Cc,
Sexed=sexed, Males=males,
Temperatures="TSP.MassWeighted.STRNWeighted.temperature.mean")
CTE <- info.nests(NestsResult=resultNest_4p,
SexualisationTRN=fitSTRN$par, out="summary")
plot_add(x=CTE$TSP.MassWeighted.STRNWeighted.temperature.mean, y=males/sexed,
col="red", pch=19)
legend("topright", legend=c("CTE with Sexualisation TRN"),
pch=19, col=c("red"))
plotR(parameters=fitSTRN$par, main="Sexualisation TRN")
# Initial_STRN <- resultNest_4p$par
Initial_STRN <- structure(c(4230.10750319997, 510.543319171189, 1015.78663983953,
118.189709917707), .Names = c("DHA", "DHH", "T12H", "Rho25"))
males <- c(7, 0, 0, 0, 0, 5, 6, 3, 5, 3, 2, 3, 0, 0, 0, 0, 0, 0, 0, 0)
sexed <- rep(10, length(males))
fitSTRN <- STRN(Initial_STRN, EmbryoGrowthTRN=resultNest_4p, tsd=Med_Cc,
Sexed=sexed, Males=males,
Temperatures="TSP.MassWeighted.STRNWeighted.temperature.mean")
CTE <- info.nests(NestsResult=resultNest_4p,
SexualisationTRN=fitSTRN$par, out="summary")
plot(Med_Cc, xlim=c(25, 35))
plot_add(x=CTE$TSP.MassWeighted.STRNWeighted.temperature.mean, y=males/sexed,
col="red", pch=19)
legend("topright", legend=c("CTE with Sexualisation TRN"),
pch=19, col=c("red"))
plotR(parameters=fitSTRN$par, main="Sexualisation TRN")

## End(Not run)

```

STRN_MHmcmc

Metropolis-Hastings algorithm for Sexualisation Thermal Reaction Norm

Description

Run the Metropolis-Hastings algorithm for Sexualisation Thermal Reaction Norm.

The number of iterations is $n.iter+n.adapt+1$ because the initial likelihood is also displayed.

I recommend that $thin=1$ because the method to estimate SE uses resampling.

If initial point is maximum likelihood, $n.adapt = 0$ is a good solution.

To get the SE of the point estimates from `result_mcmc <- STRN_MHmcmc(result=try)`, use:
`result_mcmc$SD`

`coda` package is necessary for this function.

The `dataSTRN` is a named list with the following objects:

- EmbryoGrowthTRN= result of [searchR](#)
- tsd= result of [tsd](#)
- sexed= vector with number of sexed embryos
- males= vector with number of males (could be also females=)

- Temperatures= a text of the temperatures name used as CTE

The Temperatures text for CTE can be:

- TimeWeighted.temperature.mean
- TSP.TimeWeighted.temperature.mean
- TSP.MassWeighted.temperature.mean
- TSP.STRNWeighted.temperature.mean
- TSP.MassWeighted.STRNWeighted.temperature.mean
- MiddleThird.TimeWeighted.temperature.mean

They are explained in the `info.nests` function.

This function is not still fully described as it has not been published still.

The parameters `intermediate` and `filename` are used to save intermediate results every 'intermediate' iterations (for example 1000). Results are saved in a file of name `filename`.

The parameter `previous` is used to indicate the list that has been save using the parameters `intermediate` and `filename`. It permits to continue a mcmc search.

These options are used to prevent the consequences of computer crash or if the run is very very long and processes at time limited.

Usage

```
STRN_MHmcmc(result = NULL, n.iter = 10000, parametersMCMC = NULL,
  n.chains = 1, n.adapt = 0, thin = 1, trace = FALSE,
  batchSize = sqrt(n.iter), dataSTRN = NULL, intermediate = NULL,
  filename = "intermediate.Rdata", previous = NULL)
```

Arguments

<code>result</code>	An object obtained after a STRN fit
<code>n.iter</code>	Number of iterations for each step
<code>parametersMCMC</code>	A set of parameters used as initial point for searching with information on priors
<code>n.chains</code>	Number of replicates
<code>n.adapt</code>	Number of iterations before to store outputs
<code>thin</code>	Number of iterations between each stored output
<code>trace</code>	True or False, shows progress
<code>batchSize</code>	Number of observations to include in each batch fo SE estimation
<code>dataSTRN</code>	A named list data used to estimate likelihoods (see further in description)
<code>intermediate</code>	Period for saving intermediate result, NULL for no save
<code>filename</code>	If <code>intermediate</code> is not NULL, save intermediate result in this file
<code>previous</code>	Previous result to be continued. Can be the filename in which intermediate results are saved.

Details

STRN_MHmcmc runs the Metropolis-Hastings algorithm for STRN (Bayesian MCMC)

Value

A list with resultMCMC being mcmc.list object, resultLnL being likelihoods and parametersMCMC being the parameters used

Author(s)

Marc Girondot

Examples

```
## Not run:  
library(embryogrowth)  
  
## End(Not run)
```

STSRE_NestingArea *Database of RMU for marine turtles*

Description

Database of RMU for marine turtles

Usage

STSRE_NestingArea

Format

A dataframe with raw data.

Details

Database of RMU for marine turtles

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

Maria Sousa Martins <maria.esmartins@gmail.com>

References

Wallace, B. P., DiMatteo, A. D., Hurley, B. J., Finkbeiner, E. M., Bolten, A. B., Chaloupka, M. Y., Hutchinson, B. J., Abreu-Grobois, F. A., Amorocho, D., Bjorndal, K. A., Bourjea, J., Bowen, B. W., Duenas, R. B., Casale, P., Choudhury, B. C., Costa, A., Dutton, P. H., Fallabrino, A., Girard, A., Girondot, M., Godfrey, M. H., Hamann, M., Lopez-Mendilaharsu, M., Marcovaldi, M. A., Mortimer, J. A., Musick, J. A., Nel, R., Seminoff, J. A., Troeng, S., Witherington, B. & Mast, R. B. 2010. Regional management units for marine turtles: a novel framework for prioritizing conservation and research across multiple scales. *Plos One*, 5, e15465.

Examples

```
## Not run:
library(embryogrowth)
data(STSRE_NestingArea)

## End(Not run)
```

STSRE_TSD

Database of TSD information for marine turtles

Description

Database of TSD information for turtles

The columns are:

- * Species: Name de the species in binominal nomenclature
- * Country: From which country the eggs come from
- * Area: Name of the beach or region the eggs come from
- * RMU: For marine turtles, name of the RMU for this population; see Wallace, B.P., DiMatteo, A.D., Hurley, B.J., Finkbeiner, E.M., Bolten, A.B., Chaloupka, M.Y., Hutchinson, B.J., Abreu-Grobois, F.A., Amorocho, D., Bjorndal, K.A., Bourjea, J., Bowen, B.W., Duenas, R.B., Casale, P., Choudhury, B.C., Costa, A., Dutton, P.H., Fallabrino, A., Girard, A., Girondot, M., Godfrey, M.H., Hamann, M., Lopez-Mendilaharsu, M., Marcovaldi, M.A., Mortimer, J.A., Musick, J.A., Nel, R., Seminoff, J.A., Troeng, S., Witherington, B., Mast, R.B., 2010. Regional management units for marine turtles: a novel framework for prioritizing conservation and research across multiple scales. *Plos One* 5, e15465.
- * Incubation.temperature: Nominal incubation temperature
- * Fluctuation: How much the temperature could fluctuate around nominal temperature
- * Precision: What is the precision of the measure of temperature
- * Correction.factor: Difference between the incubator temperature and the eggs temperature
- * IP.min: Shorter incubation period
- * IP.max: Longer incubation period
- * IP.amplitude: IP.max-IP.min if available
- * IP.mean: Mean incubation periods
- * IP.mode: Mode for incubation periods
- * IP.SE: Standard error for incubation periods
- * IP.SD: Standard deviation for incubation periods
- * IP.pm: If incubation period are indicated with a plus-minus term with no more precision

- * Total: Total number of eggs incubated
- * Hatched: Number of hatchlings
- * Intersexes: Number of individuals intersexes or ambiguous for sex phenotype
- * Males: Number of individuals identified as males
- * Females: Number of individuals identified as females
- * Sexed: Number of sexed individuals
- * Clutch: Identity of clutch
- * Reference: Bibliographic reference
- * Note: Diverse information for this incubation
- * Version: Date of the version of this database

Usage

STSRE_TSD

Format

A dataframe with raw data.

Details

Database of TSD information for marine turtles

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

Maria Sousa Martins <maria.esmartins@gmail.com>

References

- Binckley C.A., Spotila J.R., Wilson K.S. and Paladino F.V. (1998) Sex determination and sex ratios of Pacific Leatherback Turtles, *Dermochelys coriacea*. *Copeia*, 1998(2), 291-300.
- Crastz F. (1982) Embryological stages of the marine turtle *Lepidochelys olivacea* (Eschscholtz). *Rev. Biol. Trop.*, 30, 113-120.
- De Souza, R.R., Vogt, R.C., 1994. Incubation temperature influences sex and hatchling size in the neotropical turtle *Podocnemis unifilis*. *Journal of Herpetology* 28, 453-464.
- Girondot M. 1999 Statistical description of temperature-dependent sex determination using maximum likelihood. *Evolutionary Ecology Research*, 1(3), 479-486.
- Gutzke, W. H. N., & Paukstis, G. L. (1983). Influence of the hydric environment on sexual differentiation in turtles. *J. Exp. Zool.*, 226, 467-469. [*Chrysemys picta*]
- Hewavisenthi, S. (1999) Influence of incubation environment on the development of the flatback turtle (*Natator depressus*). Ph.D. Thesis, Central Queensland University.
- Hewavisenthi, S. & Parmenter, C.J. (2000) Hydric environment and sex determination in the flatback turtle (*Natator depressus* Garman) (*Chelonia* : *Cheloniidae*). *Australian Journal of Zoology*, 48, 653-659.

- Hulin, V., Delmas, V., Girondot, M., Godfrey, M. H. and Guillon, J.-M. 2009. Temperature-dependent sex determination and global change: are some species at greater risk? *Oecologia*, 160, 493-506.
- Limpus C.J., Reed P.C. and Miller J.D. (1985) Temperature dependent sex determination in Queensland sea turtles: Intraspecific variation in *Caretta caretta*. In Grigg G., Shine R. and Ehmann H. (eds) *Biology of Australian frogs and reptiles*. New South Wales, Australia: Royal Zoological Society, pp 343-351.
- Lopez Correa, J.Y., 2010. Diferenciación gonádica en crías de *Lepidochelys olivacea* (Eschscholtz, 1829) (Testudinata: Cheloniidae), Instituto Politécnico Nacional, Centro Interdisciplinario de Ciencias Marinas, La Paz, BCS, p. 108.
- Marcovaldi M.A., Godfrey M.H. and Mrosovsky N. (1997) Estimating sex ratios of loggerhead turtles in Brazil from pivotal incubation temperatures. *Canadian Journal of Zoology-Revue Canadienne De Zoologie*, 75, 755-770.
- McCoy C.J., Vogt R.C. and Censky E.J. (1983) Temperature-controlled sex determination in the sea turtle *Lepidochelys olivacea*. *J. Herpetol.*, 17(4), 404-406.
- Merchant-Larios, H., Diaz-Hernandez, V., Marmolejo-Valencia A. 2010. Gonadal morphogenesis and gene expression in reptiles with temperature-dependent sex determination. *Sexual Development* 4:50-61.
- Merchant-Larios, H., Villalpando-Fierro, I. & Centeno-Urruiza, B. 1989. Gonadal morphogenesis under controlled temperature in the sea turtle *Lepidochelys olivacea*. *Herpetol. Monographs*, 3, 43-61.
- Merchant-Larios, H., Ruiz-Ramirez, S., Moreno-Mendoza, N. & Marmolejo-Valencia, A. 1997. Correlation among thermosensitive period, estradiol response, and gonad differentiation in the sea turtle *Lepidochelys olivacea*. *General and Comparative Endocrinology*, 107, 373-385.
- Michel-Morfinu J.E., Gomez Munoz V.M. and Navarro Rodriguez C. (2001) Morphometric model for sex assessment in hatchling olive ridley sea turtles. *Chelonian Conservation and Biology*, 40(1), 53-58.
- Mrosovsky N., Kamel S., Rees A.F. and Margaritoulis D. (2002) Pivotal temperature for loggerhead turtles (*Caretta caretta*) from Kyparissia Bay, Greece. *Canadian Journal of Zoology-Revue Canadienne De Zoologie*, 80(12), 2118-2124.
- Mrosovsky N., Dutton P.H. and Whitmore C.P. (1984) Sex ratios of two species of sea turtle nesting in Suriname. *Can. J. Zool.*, 62, 2227-2239.
- Mrosovsky N., Bass A., Corliss L.A., Richardson J.I. and Richardson T.H. (1992) Pivotal and beach temperature for hawksbill turtles nesting in Antigua. *Can. J. Zool.*, 70, 1920-1925.
- Rimblot F., Fretey J., Mrosovsky N., Lescure J. and Pieau C. (1985) Sexual differentiation as a function of the incubation temperature of eggs in the sea-turtle *Dermochelys coriacea* (Vandelli, 1761). *Amphibia-Reptilia*, 85(6), 83-92.
- Rimblot-Baly F., Lescure J., Fretey J. and Pieau C. (1986-1987) Sensibilité à la température de la différenciation sexuelle chez la tortue Luth, *Dermochelys coriacea* (Vandelli, 1761); application des données de l'incubation artificielle à l'étude de la sex-ratio dans la nature. *Annales des Sciences Naturelles, Zoologie*, 8, 277-290.
- Ruiz Garcia N.A. (2014) Efectos de la temperatura sobre el desarrollo embrionario y el desempeño de crías de la tortuga golfina, *Lepidochelys olivacea*. Tesis Nivel Maestría.

Valenzuela, N., 2001. Constant, shift, and natural temperature effects on sex determination in *Podocnemis expansa*. *Ecology* 82, 3010-3024.

Tokunaga, S., Iwakiri, Y., Nakajima, Y., (1999). Temperature-dependent sex determination of a sea turtle, *Caretta caretta*, from Miyazaki, Japan. *Bull. Kitakyushu Mus. Nat. Hist.* 18, 147-156.

Wibbels T., Rostal D.C. and Byles R. (1998) High pivotal temperature in the sex determination of the olive ridley sea turtle, *Lepidochelys olivacea*, from Playa Nancite, Costa Rica. *Copeia*, 1998(4), 1086-1088.

Yntema C.L. and Mrosovsky N. (1980) Sexual differentiation in hatchling loggerheads (*Caretta caretta*) incubated at different controlled temperatures. *Herpetologica*, 36(1), 33-36.

Yntema C.L. and Mrosovsky N. (1982) Critical periods and pivotal temperatures for sexual differentiation in loggerhead sea turtles. *Canadian Journal of Zoology-Revue Canadienne De Zoologie*, 60(5), 1012-1016.

See Also

Other Functions.for.temperature.dependent.sex.determination: [TSP.list](#); [predict.tsd](#); [tsd_MHmcmc_p](#); [tsd_MHmcmc](#); [tsd](#)

Examples

```
## Not run:
library(embryogrowth)
data(STSRE_TSD)
STSRE_TSD$Version[1]
totalIncubation_Lo <- subset(STSRE_TSD, Species=="Lepidochelys olivacea" & Sexed!=0)
tot_Lo <- with(totalIncubation_Lo, tsd(males=Males, females=Females,
  temperatures=Incubation.temperature, par=c(P=29, S=-0.01), xlim=c(25, 35)))
predict(tot_Lo)

## End(Not run)
```

summary.Nests

Summarize the information from a Nests object.

Description

Summarize the information from a Nests object:
 - Name of the nests, total incubation length and average temperature

Usage

```
## S3 method for class 'Nests'
summary(object, ...)
```

Arguments

object	A object obtained after FormatNests()
...	Not used

Details

summary.Nests Summarize the information from a Nests object

Value

None

Author(s)

Marc Girondot

Examples

```
## Not run:  
library(embryogrowth)  
data(nest)  
formatted <- FormatNests(nest, previous=NULL)  
summary(formatted)  
  
## End(Not run)
```

switch.transition *Add a transition parameter on a set of parameters or remove it*

Description

Add a transition parameter on a set of parameters or remove it

Usage

```
switch.transition(parameters = stop("A set of parameters must be supplied"))
```

Arguments

parameters A vector with parameters

Details

switch.transition Add a transition parameter on a set of parameters or remove it

Value

A vector with parameters

Author(s)

Marc Girondot

Examples

```
## Not run:
data(resultNest_6p)
# Get a set of parameters without transition
x1 <- resultNest_6p$par
# Generate a set of parameters with transition
x2 <- switch.transition(x1)
# Generate a set of parameters without transition
x3 <- switch.transition(x3)

## End(Not run)
```

tempConst

Timeseries of constant temperatures for nests

Description

Timeseries of temperatures for nests

Usage

```
tempConst
```

Format

A dataframe with raw data.

Details

Timeseries of constant temperatures for nests

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Girondot, M. & Kaska, Y. Submitted. A model to predict temperature dependency on embryo growth rate and incubation duration from field data.

Examples

```
## Not run:
library(embryogrowth)
# Same as:
# GenerateConstInc(durations = rep(104*60*24, 11),
# temperatures = 25:35,
# names = paste0("T",25:35))
data(tempConst)
```

```

tempConst_f <- FormatNests(tempConst)
x <- structure(c(118.768297442004, 475.750095909406, 306.243694918151,
116.055824800264), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
resultNest_4p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.linear, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_4p)
# Use the fitted parameters from resultNest_4p with
# the constant incubation temperatures:
plot(resultNest_4p, temperatures=tempConst_f,
stopattest=TRUE, series="all", xlim=c(0,120),
ylimT=c(22, 32), show.stages=FALSE, show.PT=FALSE,
show.temperatures=FALSE, show.TSP=FALSE)

## End(Not run)

```

TestParallel

Estimate the likelihood of a set of parameters for nest incubation data with or without parallel computing option

Description

Estimate the likelihood of a set of parameters for nest incubation data with or without parallel computing option. It uses the user time from the print result of `system.time()` function.

Usage

```
TestParallel(result = stop("A ResultNest object must be provided"))
```

Arguments

`result` A object obtained after `searchR` or `likelihoodR`

Details

TestParallel estimates the likelihood of a set of parameters for nest incubation data with or without parallel computing option

Value

The gain or loss of computing time using parallel version

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(resultNest_4p)
TestParallel(resultNest_4p)

## End(Not run)
```

TRN_MHmcmc_p	<i>Generates set of parameters to be used with GRTRN_MHmcmc() or STRN_MHmcmc()</i>
--------------	--

Description

Interactive script used to generate set of parameters to be used with GRTRN_MHmcmc() or STRN_MHmcmc().

Usage

```
TRN_MHmcmc_p(result = NULL, parameters = NULL, fixed.parameters = NULL,
             accept = FALSE)
```

Arguments

result	An object obtained after a SearchR fit
parameters	A set of parameters. Replace the one from result
fixed.parameters	A set of fixed parameters. Replace the one from result
accept	If TRUE, the script does not wait user information

Details

TRN_MHmcmc_p generates set of parameters to be used with GRTRN_MHmcmc() or STRN_MHmcmc()

Value

A matrix with the parameters

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
# The initial parameters value can be:
# "T12H", "DHA", "DHH", "Rho25"
# Or
# "T12L", "T12H", "DHA", "DHH", "DHL", "Rho25"
x <- structure(c(118.768297442004, 475.750095909406, 306.243694918151,
116.055824800264), .Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
resultNest_4p <- searchR(parameters=x, fixed.parameters=pfixed,
temperatures=formatted, derivate=dydt.Gompertz, M0=1.7,
test=c(Mean=39.33, SD=1.92))
data(resultNest_4p)
pMCMC <- TRN_MHmcmc_p(resultNest_4p, accept=TRUE)

## End(Not run)
```

tsd

Estimate the parameters that best describe temperature-dependent sex determination

Description

Estimate the parameters that best describe temperature-dependent sex determination

Usage

```
tsd(df = NULL, males = NULL, females = NULL, N = NULL,
temperatures = NULL, durations = NULL, l = 0.05,
parameters.initial = c(P = NA, S = -0.5, K = 0, K1 = 1, K2 = 0),
males.freq = TRUE, fixed.parameters = NULL, SE = NULL,
equation = "logistic", replicates = 1000, range.CI = 0.95,
limit.low.TRT.minimum = 5, limit.high.TRT.maximum = 90, print = TRUE,
temperatures.plot = seq(from = 20, to = 40, by = 0.1),
durations.plot = seq(from = 15, to = 100, by = 0.1))
```

Arguments

df	A dataframe with at least two columns named males, females or N and temperatures, Incubation.temperature or durations column
males	A vector with male numbers
females	A vector with female numbers
N	A vector with total numbers

temperatures	The constant incubation temperatures used to fit sex ratio
durations	The duration of incubation or TSP used to fit sex ratio
l	The limit to define TRT (see Girondot, 1999)
parameters.initial	Initial values for P, S or K search as a vector, ex. c(P=29, S=-0.3)
males.freq	If TRUE data are shown as males frequency
fixed.parameters	Parameters that will not be changed
SE	Standard errors for parameters
equation	Could be "logistic", "Hill", "Richards", "Hulin", "Double-Richards" or "GSD"
replicates	Number of replicates to estimate confidence intervals
range.CI	The range of confidence interval for estimation, default=0.95
limit.low.TRT.minimum	Minimum lower limit for TRT
limit.high.TRT.maximum	Maximum higher limit for TRT
print	Do the results must be printed at screen? TRUE (default) or FALSE
temperatures.plot	Sequences of temperatures that will be used for plotting. If NULL, does not estimate them
durations.plot	Sequences of durations that will be used for plotting. If NULL, does not estimate them

Details

tsd estimates the parameters that best describe temperature-dependent sex determination

Value

A list the pivotal temperature, transitional range of temperatures and their SE

Author(s)

Marc Girondot

References

- Girondot, M. 1999. Statistical description of temperature-dependent sex determination using maximum likelihood. *Evolutionary Ecology Research*, 1, 479-486.
- Godfrey, M.H., Delmas, V., Girondot, M., 2003. Assessment of patterns of temperature-dependent sex determination using maximum likelihood model selection. *Ecoscience* 10, 265-272.
- Hulin, V., Delmas, V., Girondot, M., Godfrey, M.H., Guillon, J.-M., 2009. Temperature-dependent sex determination and global change: are some species at greater risk? *Oecologia* 160, 493-506.

See Also

Other Functions.for.temperature.dependent.sex.determination: [STSRE_TSD](#); [TSP.list](#); [predict.tsd](#); [tsd_MHmcmc_p](#); [tsd_MHmcmc](#)

Examples

```
## Not run:
CC_AtlanticSW <- subset(STSRE_TSD, RMU=="Atlantic, SW" &
                        Species=="Caretta caretta" & Sexed!=0)
tsdL <- with (CC_AtlanticSW, tsd(males=Males, females=Females,
                                temperatures=Incubation.temperature-Correction.factor,
                                equation="logistic"))
tsdH <- with (CC_AtlanticSW, tsd(males=Males, females=Females,
                                temperatures=Incubation.temperature-Correction.factor,
                                equation="Hill"))
tsdR <- with (CC_AtlanticSW, tsd(males=Males, females=Females,
                                temperatures=Incubation.temperature-Correction.factor,
                                equation="Richards"))
tsdDR <- with (CC_AtlanticSW, tsd(males=Males, females=Females,
                                temperatures=Incubation.temperature-Correction.factor,
                                equation="Double-Richards"))
gsd <- with (CC_AtlanticSW, tsd(males=Males, females=Females,
                                temperatures=Incubation.temperature-Correction.factor,
                                equation="GSD"))
compare_AIC(Logistic_Model=tsdL, Hill_model=tsdH, Richards_model=tsdR,
            DoubleRichards_model=tsdDR, GSD_model=gsd)
#####
eo <- subset(STSRE_TSD, Species=="Emys orbicularis", c("Males", "Females",
            "Incubation.temperature"))

eo_Hill <- with(eo, tsd(males=Males, females=Females,
                        temperatures=Incubation.temperature,
                        equation="Hill"))

eo_Hill <- tsd(df=eo, equation="Hill")
eo_logistic <- tsd(eo)
eo_Richards <- with(eo, tsd(males=Males, females=Females,
                            temperatures=Incubation.temperature,
                            equation="Richards"))

### The Hulin model is a modification of Richards (See Hulin et al. 2009)
### limit.low.TRT and limit.high.TRT must be setup for Hulin equation
par <- eo_Richards$par
names(par)[which(names(par)=="K")] <- "K2"
par <- c(par, K1=0)
eo_Hulin <- with(eo, tsd(males=Males, females=Females,
                        parameters.initial=par,
                        temperatures=Incubation.temperature,
                        equation="Hulin",
                        limit.low.TRT.minimum=25,
                        limit.high.TRT.maximum=35))

### The Double-Richards model is a Richards model with K1 and K2 using the two values
### below and above P
par <- eo_Richards$par
```



```

names(par)[which(names(par)=="K")] <- "K2"
par <- c(par, K1=as.numeric(par["K2"])-0.1)
par["K1"] <- par["K1"]-0.1
eo_Double_Richards <- with(eo, tsd(males=Males, females=Females,
                                parameters.initial=par,
                                temperatures=Incubation.temperature,
                                equation="Double-Richards"))
compare_AIC(Logistic=eo_logistic, Hill=eo_Hill, Richards=eo_Richards,
            Hulin=eo_Hulin, Double_Richards=eo_Double_Richards)
### Note the asymmetry of the Double-Richards model
predict(eo_Double_Richards,
        temperatures=c(eo_Double_Richards$par["P"]-0.2, eo_Double_Richards$par["P"]+0.2))
predict(eo_Double_Richards)
### It can be used also for incubation duration
CC_AtlanticSW <- subset(STSRE_TSD, RMU=="Atlantic, SW" &
                       Species=="Caretta caretta" & Sexed!=0)
tsdL_IP <- with (CC_AtlanticSW, tsd(males=Males, females=Females,
                                   durations=IP.mean,
                                   equation="logistic"))
plot(tsdL_IP, xlab="Incubation durations in days")

## End(Not run)

```

tsd_MHmcmc

*Metropolis-Hastings algorithm for Sex ratio***Description**

Run the Metropolis-Hastings algorithm for tsd.

Deeply modified from a MCMC script by Olivier Martin (INRA, Paris-Grignon).

The number of iterations is $n.iter+n.adapt+1$ because the initial likelihood is also displayed.

I recommend that $thin=1$ because the method to estimate SE uses resampling.

If initial point is maximum likelihood, $n.adapt = 0$ is a good solution.

To get the SE from `result_mcmc <- tsd_MHmcmc(result=try)`, use:

`result_mcmc$BatchSE` or `result_mcmc$TimeSeriesSE`

The batch standard error procedure is usually thought to be not as accurate as the time series methods.

Based on Jones, Haran, Caffo and Neath (2005), the batch size should be equal to $\sqrt{n.iter}$.

Jones, G.L., Haran, M., Caffo, B.S. and Neath, R. (2006) Fixed Width Output Analysis for Markov chain Monte Carlo, *Journal of the American Statistical Association*, 101:1537-1547.

coda package is necessary for this function.

#' The parameters `intermediate` and `filename` are used to save intermediate results every 'intermediate' iterations (for example 1000). Results are saved in a file of name `filename`.

The parameter `previous` is used to indicate the list that has been save using the parameters `intermediate` and `filename`. It permits to continue a mcmc search.

These options are used to prevent the consequences of computer crash or if the run is very very long and processes at time limited.

Usage

```
tsd_MHmcmc(result = NULL, n.iter = 10000, parametersMCMC = NULL,
  n.chains = 1, n.adapt = 0, thin = 1, trace = FALSE,
  batchSize = sqrt(n.iter), intermediate = NULL,
  filename = "intermediate.Rdata", previous = NULL)
```

Arguments

result	An object obtained after a SearchR fit
n.iter	Number of iterations for each step
parametersMCMC	A set of parameters used as initial point for searching with information on priors
n.chains	Number of replicates
n.adapt	Number of iterations before to store outputs
thin	Number of iterations between each stored output
trace	True or False, shows progress
batchSize	Number of observations to include in each batch fo SE estimation
intermediate	Period for saving intermediate result, NULL for no save
filename	If intermediate is not NULL, save intermediate result in this file
previous	Previous result to be continued. Can be the filename in which intermediate results are saved.

Details

tsd_MHmcmc runs the Metropolis-Hastings algorithm for tsd (Bayesian MCMC)

Value

A list with resultMCMC being mcmc.list object, resultLnL being likelihoods and parametersMCMC being the parameters used

Author(s)

Marc Girondot

See Also

Other Functions.for.temperature.dependent.sex.determination: [STSRE_TSD](#); [TSP.list](#); [predict.tsd](#); [tsd_MHmcmc_p](#); [tsd](#)

Examples

```
## Not run:
library(embryogrowth)
eo <- subset(STSRE_TSD, Species=="Emys orbicularis", c("Males", "Females",
  "Incubation.temperature"))
eo_logistic <- tsd(eo)
pMCMC <- tsd_MHmcmc_p(eo_logistic, accept=TRUE)
```

```

# Take care, it can be very long; several days
result_mcmc_tsd <- tsd_MHmcmc(result=eo_logistic,
parametersMCMC=pMCMC, n.iter=10000, n.chains = 1,
n.adapt = 0, thin=1, trace=TRUE)
# summary() permits to get rapidly the standard errors for parameters
summary(result_mcmc_tsd)
# They are store in the result also. Two SE are estimated using or
# batch method or time-series SE:
# The batch standard error procedure is usually thought to be not
# as accurate as the time series methods.
se1 <- result_mcmc_tsd$BatchSE
se2 <- result_mcmc_tsd$TimeSeriesSE
plot(result_mcmc_tsd, parameters="S", scale.prior=TRUE, xlim=c(-3, 3), las=1)
plot(result_mcmc_tsd, parameters="P", scale.prior=TRUE, xlim=c(25, 35), las=1)
plot(eo_logistic, se=se2)

## End(Not run)

```

tsd_MHmcmc_p

Generates set of parameters to be used with tsd_MHmcmc()

Description

Interactive script used to generate set of parameters to be used with tsd_MHmcmc().

Usage

```
tsd_MHmcmc_p(result = stop("An output from tsd() must be provided"),
accept = FALSE)
```

Arguments

result	An object obtained after a tsd fit
accept	If TRUE, the script does not wait user information

Details

tsd_MHmcmc_p generates set of parameters to be used with tsd_MHmcmc()

Value

A matrix with the parameters

Author(s)

Marc Girondot

See Also

Other Functions.for.temperature.dependent.sex.determination: [STSRE_TSD](#); [TSP.list](#); [predict.tsd](#); [tsd_MHmcmc](#); [tsd](#)

Examples

```
## Not run:
library(embryogrowth)
eo <- subset(STSRE_TSD, Species=="Emys orbicularis", c("Males", "Females",
"Incubation.temperature"))
eo_logistic <- with(eo, tsd(males=Males, females=Females,
temperatures=Incubation.temperature))
pMCMC <- tsd_MHmcmc_p(eo_logistic, accept=TRUE)

## End(Not run)
```

TSP.list	<i>Database of thermosensitive period of development for sex determination</i>
----------	--

Description

Database of thermosensitive period of development for sex determination

Usage

```
TSP.list
```

Format

A list with dataframes including attributes

Details

Database of thermosensitive period of development for sex determination

Author(s)

Marc Girondot <marc.girondot@u-psud.fr>

References

Pieau, C., Dorizzi, M., 1981. Determination of temperature sensitive stages for sexual differentiation of the gonads in embryos of the turtle, *Emys orbicularis*. *Journal of Morphology* 170, 373-382.
 Yntema, C.L., Mrosovsky, N., 1982. Critical periods and pivotal temperatures for sexual differentiation in loggerhead sea turtles. *Canadian Journal of Zoology-Revue Canadienne de Zoologie* 60, 1012-1016.

See Also

Other Functions.for.temperature.dependent.sex.determination: [STSRE_TSD](#); [predict.tsd](#); [tsd_MHmcmc_p](#); [tsd_MHmcmc](#); [tsd](#)

Examples

```
## Not run:
library(embryogrowth)
data(TSP.list)
names(TSP.list)
TSP.list[["Emys_orbicularis.mass"]]
attributes(TSP.list[["Emys_orbicularis.mass"]])$TSP.begin.stages
attributes(TSP.list[["Emys_orbicularis.mass"]])$TSP.end.stages

## End(Not run)
```

weightmaxentropy	<i>Search for the weights of the nests which maximize the entropy of nest temperatures distribution</i>
------------------	---

Description

Search for the weights of the nests which maximize the entropy of nest temperatures distribution. Entropy is measured by Shanon index.
Entropy method must be entropy.empirical because it is the only method insensitive to scaling.
If no weight is given, the initial weight is uniformly distributed.
Use control_optim=list(trace=0) for not show progress of search report.

Usage

```
weightmaxentropy(temperatures = stop("Temperature data must be provided !"),
  weight = NULL, entropy.method = entropy::entropy.empirical, plot = TRUE,
  control_optim = list(trace = 0, maxit = 500), control_plot = NULL,
  control_entropy = NULL, col = c("black", "red"))
```

Arguments

temperatures	Timeseries of temperatures formatted using FormatNests()
weight	A named vector of the initial weight search for each nest for likelihood estimation
entropy.method	Entropy function, for example entropy::entropy.empirical. See package entropy for description
plot	Do the plot of temperatures before and after weight must be shown ? TRUE or FALSE
control_optim	A list with control paramaters for optim function
control_plot	A list with control paramaters for plot function

control_entropy A list with control parameters for entropy function
 col Colors for unweighted and weighted distributions

Details

Search for the weights of the nests which maximize the entropy of nest temperatures distribution

Value

A named vector of weights

Author(s)

Marc Girondot

Examples

```
## Not run:
library(embryogrowth)
data(nest)
formatted <- FormatNests(nest)
w <- weightmaxentropy(formatted, control_plot=list(xlim=c(20,36)))
x <- structure(c(120.940334922916, 467.467455887442,
306.176613681557, 117.857995419495),
.Names = c("DHA", "DHH", "T12H", "Rho25"))
# pfixed <- c(K=82.33) or rK=82.33/39.33
pfixed <- c(rK=2.093313)
# K or rK are not used for dydt.linear or dydt.exponential
resultNest_4p_weight <- searchR(parameters=x,
fixed.parameters=pfixed, temperatures=formatted,
derivate=dydt.Gompertz, M0=1.7, test=c(Mean=39.33, SD=1.92),
method = "BFGS", weight=w)
data(resultNest_4p_weight)
plotR(resultNest_4p_weight, ylim=c(0,0.50), xlim=c(15, 35))
# Standard error of parameters can use the GRTRN_MHmcmc() function

## End(Not run)
```

Index

- *Topic **Ecology**
 - embryogrowth-package, 2
- *Topic **Embryo**
 - embryogrowth-package, 2
- *Topic **Gompertz**
 - embryogrowth-package, 2
- *Topic **Growth**
 - embryogrowth-package, 2
- *Topic **Sex-determination**
 - embryogrowth-package, 2
- *Topic **Temperature**
 - embryogrowth-package, 2
- *Topic **datasets**
 - nest, 26
 - result_mcmc_4p, 42
 - result_mcmc_4p_weight, 43
 - result_mcmc_6p, 44
 - result_mcmc_newp, 46
 - resultNest_4p, 37
 - resultNest_4p_transition, 38
 - resultNest_4p_weight, 39
 - resultNest_6p, 40
 - resultNest_newp, 41
 - STSRE_NestingArea, 53
 - STSRE_TSD, 54
 - tempConst, 59
 - TSP.list, 68
- ChangeSSM, 5
- dydt.exponential, 6
- dydt.Gompertz, 8
- dydt.linear, 9
- embryogrowth-package, 2
- FormatNests, 10
- GenerateAnchor, 11
- GenerateConstInc, 12
- GenerateTest, 13
- GRTRN_MHmcmc, 14
- hist.Nests, 16
- hist.NestsResult, 17
- info.nests, 18, 52
- likelihoodR, 21
- logLik.NestsResult, 23
- logLik.tsd, 24
- MovingIncubation, 25
- nest, 26
- plot.NestsResult, 27
- plot.tsd, 30
- plot_transition, 35
- plotR, 32
- plotR_hist, 34
- predict.tsd, 36, 57, 64, 66, 68, 69
- result_mcmc_4p, 42
- result_mcmc_4p_weight, 43
- result_mcmc_6p, 44
- result_mcmc_newp, 46
- resultNest_4p, 37
- resultNest_4p_transition, 38
- resultNest_4p_weight, 39
- resultNest_6p, 40
- resultNest_newp, 41
- searchR, 47, 51
- STRN, 49
- STRN_MHmcmc, 51
- STSRE_NestingArea, 53
- STSRE_TSD, 37, 54, 64, 66, 68, 69
- summary.Nests, 57
- switch.transition, 58
- tempConst, 59

TestParallel, 60
TRN_MHmcmc_p, 61
tsd, 37, 51, 57, 62, 66, 68, 69
tsd_MHmcmc, 37, 57, 64, 65, 68, 69
tsd_MHmcmc_p, 37, 57, 64, 66, 67, 69
TSP.list, 37, 57, 64, 66, 68, 68

weightmaxentropy, 69