Package ‘epcc’

October 13, 2022

Type    Package
Title   Simulating Populations of Ectotherms under Global Warming
Version 1.4.7

Description Provides several functions that allow model and simulate the effects of thermal sensitivity and the exposition to different trends in environmental temperature on the abundance dynamics of ectotherms populations. It allows an easy implementation of the possible consequences of warming at global and local scales, constituting a useful tool for understanding the extinction risk of populations. (Víctor Saldaña-Núñez, Fernando Córdova-Lepe, & Felipe N. Moreno-Gómez, 2021) <doi:10.5281/zenodo.5034087>.

License GPL-3

URL https://github.com/Victor-Saldana/epcc

BugReports https://github.com/Victor-Saldana/epcc/issues

Imports deSolve (>= 1.28), ggplot2 (>= 3.3.3), httr (>= 1.4.2), cowplot (>= 1.1.1), sp (>= 1.4-5), nls2 (>= 0.2), readxl (>= 1.3.1), raster (>= 3.1-5), proto (>= 1.0.0), rlang, rgdal (>= 1.5-10), formattable (>= 0.2.1)

Encoding UTF-8

RoxygenNote 7.1.1

NeedsCompilation no

Author Víctor Saldaña-Núñez [aut, cre], Fernando Córdova-Lepe [aut], Felipe N. Moreno-Gómez [aut]

Maintainer Víctor Saldaña-Núñez <saldananunezvictor@gmail.com>

Repository CRAN

Date/Publication 2021-06-29 08:00:02 UTC
**Description**

This function allows to simulate the effect of IPCC (2014) RCP2.6 or RCP8.5 scenarios on the abundances of an ectotherm population considering three stages as age structure.

**Usage**

```r
age_structure(
  y_ini = c(N1 = 800, N1 = 800, N1 = 800, N2 = 600, N2 = 600, N2 = 600, N3 = 400, N3 = 400),
  temp_ini = rep(25 + 273.15, 3),
  temp_cmin = rep(18 + 273.15, 3),
  temp_cmax = c(25 + 273.15, 28 + 273.15, 35 + 273.15),
  ro = rep(0.7, 3),
  lambda1 = rep(4e-04, 3),
  lambda2 = rep(4e-04, 3),
  lambda3 = rep(4e-04, 3),
  alpha1 = rep(0.1, 3),
)```

Age structure under IPCC RCP2.6 or RCP8.5 scenarios
\[\alpha_2 = \text{rep}(0.7, 3),\]
\[d_2 = \text{rep}(0.005, 3),\]
\[d_3 = \text{rep}(0.5, 3),\]
\[A_{d_2} = \text{rep}(0.5, 3),\]
\[A_{d_3} = \text{rep}(0.75, 3),\]
\[T_{r_2} = \text{rep}(298, 3),\]
\[T_{r_3} = \text{rep}(298, 3),\]
\[RCP = 2.6,\]
\[\text{time}_\text{start} = 2005,\]
\[\text{time}_\text{end} = 2100,\]
\[\text{leap} = \frac{1}{50}\]

**Arguments**

- **y_ini**
  Initial population values (must be written with its name: \(N\)).
- **temp_ini**
  Initial temperature (K).
- **temp_cmin**
  Minimum critical temperature (K).
- **temp_cmax**
  Maximum critical temperature (K).
- **ro**
  Population growth rate at optimum temperature.
- **lambda1, lambda2, lambda3**
  Marginal loss by non-thermodependent intraspecific competition.
- **alpha1, alpha2**
  Stage 1 to 2 and 2 to 3 transition coefficient respectively.
- **d2, d3**
  Mortality rate at a reference temperature.
- **Ad2, Ad3**
  Arrhenius constant which quantifies the temperature sensitivity of mortality.
- **Tr2, Tr3**
  Reference temperature (K).
- **RCP**
  Representative concentration trajectories (RCP2.6 and RCP8.5 scenarios).
- **time_start**
  Start of time sequence.
- **time_end**
  End of time sequence.
- **leap**
  Time sequence step.

**Details**

Three scenarios can be evaluated for a predation interaction where the prey is an ectotherm population. The temperature trends correspond to IPCC projections under the RCP2.6 or RCP8.5 scenarios. In each input vector, the parameters for the three simulations must be specified.

**Value**

1. A data.frame with columns having the simulated trends.
2. Figure of four panels in which (a), (b) and (c) show the population abundance curves for each age stage. Panel (d) shows the temperature trend curves used for each simulation, colored brown, green and blue, respectively.
References


Examples

# Example 1: Different thermal tolerance ranges (scenario RCP2.6).

temp_cmin <- 291

# Temperature that occurs before the minimum simulation time.
temp_i <- 295

time_end <- 2100

# Temperature that occurs in the maximum time of the simulation.
temp_max <- get_RCP2.6(time_end)+temp_i

# Simulation thermal range.
RS <- temp_max-temp_cmin

temp_cmax1 <- 4/3*RS+temp_cmin

temp_cmax2 <- 2/3*RS+temp_cmin

temp_cmax3 <- 1/3*RS+temp_cmin

temp_ini <- (temp_cmin+temp_cmax3)/2

age_structure(y_ini = c(N1 = 800, N1 = 800, N1 = 800,
                        N2 = 600, N2 = 600, N2 = 600,
                        N3 = 400, N3 = 400, N3 = 400),
              temp_ini = rep(temp_ini,3),
              temp_cmin = rep(temp_cmin,3),
              temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
              ro = rep(0.7,3),
              lambda1 = c(0.00002,0,0),
              lambda2 = c(0,0.00004,0.00003),
              lambda3 = c(0,0.00003,0.00004),
              alpha1 = rep(0.3,3),
              alpha2 = rep(0.4,3),
              d2 = rep(0.004,3),
              d3 = rep(0.005,3),
              Ad2 = rep(0.5,3),
              Ad3 = rep(0.75,3),
              Tr2 = rep(298,3),
              Tr3 = rep(298,3),
              RCP = 2.6,
              time_start = 2005,
              time_end = time_end,
              leap = 1/50)
Example 2: Different thermal tolerance ranges (scenario RCP8.5).

```r
# Example 2: Different thermal tolerance ranges (scenario RCP8.5).

temp_cmin <- 291

# Temperature that occurs before the minimum simulation time.
temp_i <- 295
time_end <- 2100

# Temperature that occurs in the maximum time of the simulation.
temp_max <- get_RCP8.5(time_end)+temp_i

# Simulation thermal range.
RS <- temp_max-temp_cmin
temp_cmax1 <- 4/3*RS+temp_cmin
temp_cmax2 <- 2/3*RS+temp_cmin
temp_cmax3 <- 1/3*RS+temp_cmin
temp_ini <- (temp_cmin+temp_cmax3)/2

age_structure(y_ini = c(N1 = 800, N1 = 800, N1 = 800,
                       N2 = 600, N2 = 600, N2 = 600,
                       N3 = 400, N3 = 400, N3 = 400),
              temp_ini = rep(temp_ini,3),
              temp_cmin = rep(temp_cmin,3),
              temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
              ro = rep(0.7,3),
              lambda1 = c(0.00002,0,0),
              lambda2 = c(0,0.00004,0.00003),
              lambda3 = c(0,0.00003,0.00004),
              alpha1 = rep(0.3,3),
              alpha2 = rep(0.4,3),
              d2 = rep(0.004,3),
              d3 = rep(0.003,3),
              Ad2 = rep(0.5,3),
              Ad3 = rep(0.6,3),
              Tr2 = rep(298,3),
              Tr3 = rep(298,3),
              RCP = 8.5,
              time_start = 2005,
              time_end = time_end,
              leap = 1/50)
```

Interspecific competition under the influence of temperature trend
adapted from the IPCC projection (RCP2.6 or RCP8.5 scenarios)
competition

Description

This function allows simulating the effect of temperature trends according to RCP2.6 or RCP8.5 scenarios (2014) on the abundances of two competing species, where one of them is ectothermic.

Usage

```r
competition(
  y_ini = c(N1 = 400, N1 = 400, N1 = 400, N2 = 200, N2 = 200, N2 = 200),
  temp_ini = rep(25, 3),
  temp_cmin = rep(18, 3),
  temp_cmax = c(25, 28, 35),
  ro = rep(0.7, 3),
  r2 = rep(0.7, 3),
  lambda1 = rep(5e-05, 3),
  K2 = rep(5e-05, 3),
  alpha = rep(0.002, 3),
  beta = rep(0.03, 3),
  RCP = 2.6,
  time_start = 2005,
  time_end = 2100,
  leap = 1/50
)
```

Arguments

- `y_ini`: Initial population values (must be written with its name: N).
- `temp_ini`: Initial temperature.
- `temp_cmin`: Minimum critical temperature.
- `temp_cmax`: Maximum critical temperature.
- `ro`: Population growth rate at optimal temperature of species-1.
- `lambda1`: Marginal loss a by non-thermodependent intraspecific competition factor of species-1.
- `K2`: Carrying capacity of species-2.
- `alpha`: Competition coefficient that quantifies the per capita effect of species-2 on species-1.
- `beta`: Per capita competition coefficient that quantifies the per capita effect of species-1 on species-2.
- `RCP`: Representative concentration trajectories (RCP2.6 and RCP8.5 scenarios).
- `time_start`: Start of time sequence.
- `time_end`: End of time sequence.
- `leap`: Time sequence step.
Details

The function allows simulating simultaneously three potential outcomes for the interaction of two competing populations where one is an ectothermic species. The temperature trends that can be specified corresponds to IPCC projections under the RCP2.6 or RCP8.5 scenarios.

Value

1. A data.frame with columns having the simulated trends.
2. A four-panel figure where (a), (b), and (c) show the abundance curves of the populations for each simulation, where the brown curve corresponds to the abundance of the ectotherm species and the green curve to the species not affected by temperature. Panel (d) shows the temperature trend curves, as they may differ for each simulation, these will be displayed by the colors green, blue, and black respectively.

References


Examples

```r
# Example 1: Different thermal tolerance ranges (scenario RCP2.6).

temp_cmin <- 18
# Temperature that occurs before the minimum simulation time.
temp_i <- 22
# Temperature that occurs in the maximum time of the simulation.
time_end <- 2100
# Simulation thermal range.
RS <- temp_max-temp_cmin
temp_cmax1 <- 4/3*RS+temp_cmin
temp_cmax2 <- 2/3*RS+temp_cmin
temp_cmax3 <- 1/3*RS+temp_cmin
temp_ini <- (temp_cmin+temp_cmax3)/2

competition(y_ini = c(N1 = 400, N1 = 400, N1 = 400, 
                  N2 = 300, N2 = 300, N2 = 300),
            temp_ini = rep(temp_ini,3),
            temp_cmin = rep(temp_cmin,3),
            temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
            ro = rep(0.7,3),
```

r2 = rep(0.7,3),
lambda1 = rep(0.0005,3),
K2 = rep(1400,3),
alpha = rep(0.02,3),
beta = rep(0.3,3),
RCP = 2.6,
time_start = 2005,
time_end = time_end,
leap = 1/50)

#Example 2: Different thermal tolerance ranges (scenario RCP8.5).

temp_cmin <- 18
# Temperature that occurs before the minimum simulation time.
temp_i <- 22
# Temperature that occurs in the maximum time of the simulation.
temp_max <- get_RCP8.5(time_end)*temp_i
# Simulation thermal range.
RS <- temp_max-temp_cmin
temp_cmax1 <- 4/3*RS+temp_cmin
temp_cmax2 <- 2/3*RS+temp_cmin
temp_cmax3 <- 1/3*RS+temp_cmin
temp_ini <- (temp_cmin+temp_cmax3)/2

competition(y_ini = c(N1 = 400, N1 = 400, N1 = 400,
                     N2 = 300, N2 = 300, N2 = 300),
            temp_ini = rep(temp_ini,3),
            temp_cmin = rep(temp_cmin,3),
            temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
            ro = rep(0.7,3),
            r2 = rep(0.7,3),
            lambda1 = rep(0.0005,3),
            K2 = rep(1400,3),
            alpha = rep(0.02,3),
            beta = rep(0.3,3),
            RCP = 8.5,
            time_start = 2005,
            time_end = time_end,
            leap = 1/50)

#Example 3: Different marginal losses by a non-thermodependent
# component of intraspecific competition for species-1
# (scenario RCP2.6).
lambda3 <- 0.002
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

competition(y_ini = c(N1 = 400, N1 = 400, N1 = 400,
N2 = 200, N2 = 200, N2 = 200),
temp_ini = rep(25,3),
temp_cmin = rep(20,3),
temp_cmax = rep(30,3),
ro = rep(0.5,3),
r2 = rep(0.4,3),
lambda1 = c(lambda1,lambda2,lambda3),
K2 = rep(1200,3),
alpha = rep(0.02,3),
beta = rep(0.3,3),
RCP = 2.6,
time_start = 2005,
time_end = 2100,
leap = 1/50)

# Example 4: Different marginal losses by a non-thermodependent
# component of intraspecific competition for species-1
# (scenario RCP8.5).

lambda3 <- 0.002
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

competition(y_ini = c(N1 = 400, N1 = 400, N1 = 400,
N2 = 200, N2 = 200, N2 = 200),
temp_ini = rep(25,3),
temp_cmin = rep(20,3),
temp_cmax = rep(30,3),
ro = rep(0.5,3),
r2 = rep(0.4,3),
lambda1 = c(lambda1,lambda2,lambda3),
K2 = rep(1200,3),
alpha = rep(0.02,3),
beta = rep(0.3,3),
RCP = 8.5,
time_start = 2005,
time_end = 2100,
leap = 1/50)

# Example 5: Different competition coefficients (scenario RCP2.6).

alpha1 <- 0.02
alpha2 <- 2*alpha1
\[
\text{alpha3} \leftarrow 2 \times \text{alpha2}
\]

\[
\text{competition}(y_{\text{ini}} = \begin{cases} N1 = 400, N1 = 400, N1 = 400, \\
N2 = 200, N2 = 200, N2 = 200 \end{cases},
\]
\[
\begin{align*}
temp_{\text{ini}} &= \text{rep}(25,3), \\
temp_{\text{cmin}} &= \text{rep}(20,3), \\
temp_{\text{cmax}} &= \text{rep}(30,3), \\
ro &= \text{rep}(0.5,3), \\
r2 &= \text{rep}(0.4,3), \\
lambda1 &= \text{rep}(0.0005,3), \\
K2 &= \text{rep}(1200,3), \\
alpha &= c(\text{alpha1}, \text{alpha2}, \text{alpha3}), \\
beta &= \text{rep}(0.3,3), \\
\text{RCP} &= 2.6, \\
\text{time\_start} &= 2005, \\
\text{time\_end} &= 2100, \\
\text{leap} &= 1/50)
\]

# Example 6: Different competition coefficients (scenario RCP8.5).

\[
\text{alpha1} \leftarrow 0.02 \\
\text{alpha2} \leftarrow 2 \times \text{alpha1} \\
\text{alpha3} \leftarrow 2 \times \text{alpha2}
\]

\[
\text{competition}(y_{\text{ini}} = \begin{cases} N1 = 400, N1 = 400, N1 = 400, \\
N2 = 200, N2 = 200, N2 = 200 \end{cases},
\]
\[
\begin{align*}
temp_{\text{ini}} &= \text{rep}(25,3), \\
temp_{\text{cmin}} &= \text{rep}(20,3), \\
temp_{\text{cmax}} &= \text{rep}(30,3), \\
ro &= \text{rep}(0.5,3), \\
r2 &= \text{rep}(0.4,3), \\
lambda1 &= \text{rep}(0.0005,3), \\
K2 &= \text{rep}(1200,3), \\
alpha &= c(\text{alpha1}, \text{alpha2}, \text{alpha3}), \\
beta &= \text{rep}(0.3,3), \\
\text{RCP} &= 8.5, \\
\text{time\_start} &= 2005, \\
\text{time\_end} &= 2100, \\
\text{leap} &= 1/50)
\]

---

**cooling\_pulse1**

**Cooling pulse-1**

### Description

This function allows simulating the effect of an environmental cooling pulse on the abundance of ectotherm populations. After the pulse, the temperature stabilizes at a specific temperature (temp_a).
Usage

cooling_pulse1(
  y_ini = c(N = 400, N = 400, N = 400),
  temp_ini = rep(35, 3),
  temp_cmin = rep(18, 3),
  temp_cmax = c(25, 28, 35),
  ro = rep(0.7, 3),
  lambda = rep(5e-05, 3),
  temp_peak = rep(25, 3),
  time_peak = rep(2060, 3),
  sd = rep(2, 3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12
)

Arguments

y_ini  Initial population values (must be written with its name: N).
temp_ini  Initial temperature.
temp_cmin  Minimum critical temperature.
temp_cmax  Maximum critical temperature.
ro  Population growth rate at optimum temperature.
lambda  Marginal loss by non-thermodependent intraspecific competition.
temp_peak  Peak pulse temperature.
time_peak  Time when temp_peak is reached.
sd  Vector of standard deviations associated with temperature trend.
time_start  Start of time sequence.
time_end  End of time sequence.
leap  Time sequence step.

Details

Three populations and/or scenarios can be simulated simultaneously. The temperature trend is determined by a Gaussian function, whose main parameters are the mean and standard deviation. In each input vector, the parameters for the three simulations must be specified (finite numbers for initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

Value

(1) A data.frame with columns having the simulated trends.
(2) A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.
References


Examples

#Example 1: Different initial population abundances.
cooling_pulse1(y_ini = c(N = 100, N = 200, N = 400),
    temp_ini = rep(26,3),
    temp_cmin = rep(18,3),
    temp_cmax = rep(30,3),
    ro = rep(0.7,3),
    lambda = rep(0.00005,3),
    temp_peak = rep(19,3),
    time_peak = rep(2060,3),
    sd = rep(10,3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12)

#Example 2: Different thermal tolerance ranges.
temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2
cooling_pulse1(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = rep(30,3),
    temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
    temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
    ro = rep(0.7,3),
    lambda = rep(0.00005,3),
    temp_peak = rep(19,3),
    time_peak = rep(2060,3),
    sd = rep(10,3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12)
# Example 3: Different relationships between initial environmental temperature and optimum temperature.

temp_cmin <- 18
temp_cmax <- 30

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-
(temp_cmax*temp_cmin)/3)
temp_ini1 <- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op+temp_cmax)/2

cooling_pulse1(y_ini = c(N = 100, N = 100, N = 100),
                 temp_ini = c(temp_ini1,temp_ini2,temp_ini3),
                 temp_cmin = rep(temp_cmin,3),
                 temp_cmax = rep(temp_cmax,3),
                 ro = rep(0.7,3),
                 lambda = rep(0.00005,3),
                 temp_peak = rep(19,3),
                 time_peak = rep(2060,3),
                 sd = rep(10,3),
                 time_start = 2005,
                 time_end = 2100,
                 leap = 1/12)

# Example 4: Different peaks of temperature.

temp_peak1 <- 16
temp_peak2 <- 5/4*temp_peak1
temp_peak3 <- 5/4*temp_peak2

cooling_pulse1(y_ini = c(N = 100, N = 100, N = 100),
                 temp_ini = rep(28,3),
                 temp_cmin = rep(18,3),
                 temp_cmax = rep(30,3),
                 ro = rep(0.7,3),
                 lambda = rep(0.00005,3),
                 temp_peak = c(temp_peak1,temp_peak2,temp_peak3),
                 time_peak = rep(2060,3),
                 sd = rep(10,3),
                 time_start = 2005,
                 time_end = 2100,
                 leap = 1/12)
#Example 5: Different marginal losses by a non-thermodependent component of intraspecific competition.

```r
lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

cooling_pulse1(y_ini = c(N = 100, N = 100, N = 100),
                 temp_ini = rep(29,3),
                 temp_cmin = rep(18,3),
                 temp_cmax = rep(30,3),
                 ro = rep(0.7,3),
                 lambda = c(lambda1,lambda2,lambda3),
                 temp_peak = rep(19,3),
                 time_peak = rep(2060,3),
                 sd = rep(10,3),
                 time_start = 2005,
                 time_end = 2100,
                 leap = 1/12)
```

---

cooling_pulse2 Cooling pulse-2

**Description**

This function allows simulating the effect of an environmental cooling pulse on the abundance of ectotherm populations. After the pulse, the temperature stabilizes at a temperature $q$ units lower than the starting temperature (temp_ini).

**Usage**

```r
cooling_pulse2(
    y_ini = c(N = 400, N = 400, N = 400),
    temp_ini = rep(35, 3),
    temp_cmin = c(15, 18, 20),
    temp_cmax = rep(40, 3),
    ro = rep(0.7, 3),
    lambda = rep(5e-05, 3),
    temp_peak = rep(25, 3),
    time_peak = rep(2060, 3),
    q = rep(5, 3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12
)
```
cooling_pulse2

Arguments

- **y_ini**: Initial population values (must be written with its name: N).
- **temp_ini**: Initial temperature.
- **temp_cmin**: Minimum critical temperature.
- **temp_cmax**: Maximum critical temperature.
- **ro**: Population growth rate at optimum temperature.
- **lambda**: Marginal loss by non-thermodependent intraspecific competition.
- **temp_peak**: Peak pulse temperature.
- **time_peak**: Time when temp_peak is reached.
- **q**: Difference between temp_ini and the stabilized temperature.
- **time_start**: Start of time sequence.
- **time_end**: End of time sequence.
- **leap**: Time sequence step.

Details

Three populations and/or scenarios can be simulated simultaneously. The temperature trend is determined by a Gaussian function, whose main parameters are the mean and standard deviation. In each input vector, the parameters for the three simulations must be specified (finite numbers for initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

Value

1. A data.frame with columns having the simulated trends.
2. A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References


Examples

```R
#Example 1: Different initial population abundances.
cooling_pulse2(y_ini = c(N = 100, N = 200, N = 400),
               temp_ini = rep(28,3),
               temp_cmin = rep(18,3),
```
cooling_pulse2

temp_cmax = rep(30,3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
temp_peak = rep(19,3),
time_peak = rep(2060,3),
q = rep(1,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)

#Example 2: Different thermal tolerance ranges.

temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
temp_cmax1 <- 32.4

cooling_pulse2(y_ini = c(N = 100, N = 100, N = 100),
temp_ini = rep(30,3),
temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
temp_peak = rep(19,3),
time_peak = rep(2060,3),
q = rep(1,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)

#Example 3: Different relationships between initial environmental
# temperature and optimum temperature

temp_cmin <- 18
temp_cmax <- 30

temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-
(temp_cmax*temp_cmin)/3)
temp_ini1<- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <-(temp_op+temp_cmax)/2

cooling_pulse2(y_ini = c(N = 100, N = 100, N = 100),
temp_ini = c(temp_ini1,temp_ini2,temp_ini3),
```r
temp_cmin = rep(temp_cmin,3),
temp_cmax = rep(temp_cmax,3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
temp_peak = rep(19,3),
time_peak = rep(2060,3),
q = rep(1,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)
```

# Example 4: Different peaks of temperature.

```r
temp_peak1 <- 16
temp_peak2 <- 5/4*temp_peak1
temp_peak3 <- 5/4*temp_peak2
cooling_pulse2(y_ini = c(N = 100, N = 100, N = 100),
               temp_ini = rep(28,3),
               temp_cmin = rep(18,3),
               temp_cmax = rep(30,3),
               ro = rep(0.7,3),
               lambda = rep(0.00005,3),
               temp_peak = c(temp_peak1,temp_peak2,temp_peak3),
               time_peak = rep(2060,3),
               q = rep(1,3),
               time_start = 2005,
               time_end = 2100,
               leap = 1/12)
```

# Example 5: Different marginal losses by a non-thermodependent component of intraspecific competition.

```r
lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2
cooling_pulse2(y_ini = c(N = 100, N = 100, N = 100),
               temp_ini = rep(28,3),
               temp_cmin = rep(18,3),
               temp_cmax = rep(30,3),
               ro = rep(0.7,3),
               lambda = c(lambda1,lambda2,lambda3),
               temp_peak = rep(25,3),
               time_peak = rep(2060,3),
               q = rep(1,3),
               time_start = 2005,
               time_end = 2100,
```

decreasing_linear

leap = 1/12)

decreasing_linear  Projection of decreasing linear temperature

Description
This function simulates the effect of a linear decreasing trend in environmental temperature on the abundance of ectotherm populations.

Usage
decreasing_linear(
  y_ini = c(N = 400, N = 400, N = 400),
  temp_ini = rep(35, 3),
  temp_cmin = c(18, 19, 20),
  temp_cmax = rep(40, 3),
  ro = rep(0.7, 3),
  m = rep(1/5, 3),
  lambda = rep(5e-05, 3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)

Arguments
  y_ini  Initial population values (must be written with its name: N).
  temp_ini  Initial temperature.
  temp_cmin  Minimum critical temperature.
  temp_cmax  Maximum critical temperature.
  ro  Population growth rate at optimum temperature.
  m  Slope of the temperature decreasing trend.
  lambda  Marginal loss by non-thermodependent intraspecific competition.
  time_start  Start of time sequence.
  time_end  End of time sequence.
  leap  Time sequence step.

Details
Three populations and/or scenarios can be simulated simultaneously. The temperature trend is determined by decreasing a linear function. The slope can be modified. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).
**decreasing_linear**

**Value**

1. A data.frame with columns having the simulated trends.
2. A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

**References**


**Examples**

```r
#Example 1: Different initial population abundances.
decreasing_linear(y_ini = c(N = 100, N = 200, N = 400),
                  temp_ini = rep(30, 3),
                  temp_cmin = rep(18, 3),
                  temp_cmax = rep(35, 3),
                  ro = rep(0.7, 3),
                  m = rep(1/5, 3),
                  lambda = rep(0.00005, 3),
                  time_start = 2005,
                  time_end = 2100,
                  leap = 1/12)
```

```r
#Example 2: Different thermal tolerance ranges.

temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2
decreasing_linear(y_ini = c(N=100,N=100,N=100),
                  temp_ini = rep(32, 3),
                  temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
                  temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
                  ro = rep(0.7, 3),
                  m = rep(1/5, 3),
                  lambda = rep(0.00005, 3),
                  time_start = 2005,
                  time_end = 2100,
```
decreasing_periodicity

leap = 1/12)

#############################################################################
#Example 3: Different relationships between initial environmental
#temperature and optimum temperature
#############################################################################
temp_cmin <- 18
temp_cmax <- 35

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2- 
(temp_cmax*temp_cmin)/3)
temp_ini1 <- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op+temp_cmax)/2
decreasing_linear(y_ini = c(N = 100, N = 100, N = 100),
 temp_ini = c(temp_ini1,temp_ini2,temp_ini3),
 temp_cmin = rep(temp_cmin,3),
 temp_cmax = rep(temp_cmax,3),
 ro = rep(0.7,3),
 m = rep(1/5,3),
 lambda = rep(0.00005,3),
 time_start = 2005,
 time_end = 2100,
 leap = 1/12)

#############################################################################
#Example 4: Different marginal losses by a non-thermodependent
#component of intraspecific competition.
#############################################################################
lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2
decreasing_linear(y_ini = c(N = 100, N = 100, N = 100),
 temp_ini = rep(30,3),
 temp_cmin = rep(18,3),
 temp_cmax = rep(35,3),
 ro = rep(0.7,3),
 m = rep(1/5,3),
 lambda = c(lambda1,lambda2,lambda3),
 time_start = 2005,
 time_end = 2100,
 leap = 1/12)


decreasing_periodicity

Periodic decreasing temperature trend

Description

This function simulates the effect of a decrease temperature trend with periodic variability on the abundance of ectotherm populations.

Usage

```r
decreasing_periodicity(
    y_ini = c(N = 400, N = 400, N = 400),
    temp_ini = rep(20, 3),
    temp_cmin = rep(18, 3),
    temp_cmax = c(25, 28, 35),
    ro = rep(0.7, 3),
    lambda = rep(5e-05, 3),
    A = rep(5, 3),
    B = rep(0.06, 3),
    m = rep(1/5, 3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12
)
```

Arguments

- **y_ini**: Initial population values (must be written with its name: N).
- **temp_ini**: Initial temperature.
- **temp_cmin**: Minimum critical temperature.
- **temp_cmax**: Maximum critical temperature.
- **ro**: Population growth rate at optimum temperature.
- **lambda**: Marginal loss by non-thermodependent intraspecific competition.
- **A**: Temperature wave amplitude.
- **B**: Parameter specifying the period of the trend (period is (2 pi)/|B|).
- **m**: Decreasing slope.
- **time_start**: Start of time sequence.
- **time_end**: End of time sequence.
- **leap**: Time sequence step.
decreasing_periodicity

Details

Three populations and/or scenarios can be simulated simultaneously. The temperature trend is determined by an expression that shows a linear decrease with periodic variability. The amplitude, amplitude, period and downward speed of change can be specified. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

Value

(1) A data.frame with columns having the simulated trends.
(2) A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References


Examples

```r
#Example 1: Different initial population abundances.
decreasing_periodicity(y_ini = c(N = 100, N = 200, N = 400),
  temp_ini = rep(30,3),
  temp_cmin = rep(18,3),
  temp_cmax = rep(35,3),
  ro = rep(0.7,3),
  lambda = rep(0.00005,3),
  A = rep(3,3),
  B = rep(0.6,3),
  m = rep(1/5,3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)
```

```r
#Example 2: Different thermal tolerance ranges.
temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
```
temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2
decreasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = rep(26,3),
  temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
  temp_cmax = rep(temp_cmax1,temp_cmax2,temp_cmax3),
  ro = rep(0.7,3),
  lambda = rep(0.00005,3),
  A = rep(2,3),
  B = rep(0.6,3),
  m = rep(1/5,3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)

# Example 3: Different relationships between initial environmental
# temperature and optimum temperature.

temp_cmin <- 18
temp_cmax <- 35

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-
  (temp_cmax*temp_cmin)/3)
temp_ini1 <- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op+temp_cmax)/2

decreasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = c(temp_ini1,temp_ini2,temp_ini3),
  temp_cmin = rep(temp_cmin,3),
  temp_cmax = rep(temp_cmax,3),
  ro = rep(0.7,3),
  lambda = rep(0.00005,3),
  A = rep(2,3),
  B = rep(0.6,3),
  m = rep(1/5,3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)

# Example 4: Different marginal losses by a non-thermodependent
# component of intraspecific competition.

lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

decreasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = rep(30,3),
  temp_cmin = rep(18,3),
  temp_cmax = rep(35,3),
  ro = rep(0.7,3),
  lambda = c(lambda1,lambda2,lambda3),
  A = rep(2,3),
  B = rep(0.6,3),
  m = rep(1/5,3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)

#Example 5: Different wave amplitude.

A3 <- 2
A2 <- 1/2 * A3
A1 <- 1/2 * A2

decreasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = rep(30,3),
  temp_cmin = rep(18,3),
  temp_cmax = rep(35,3),
  ro = rep(0.7,3),
  lambda = rep(0.00005,3),
  A = c(A1,A2,A3),
  B = rep(0.6,3),
  m = rep(1/5,3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)

#Example 6: Different period.

B3 <- pi/5
B2 <- 1/2 * B3
B1 <- 1/2 * B2

decreasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = rep(30,3),
  temp_cmin = rep(18,3),
  temp_cmax = rep(35,3),
  ro = rep(0.7,3),
  lambda = rep(0.00005,3),
  A = rep(2,3),
  B = c(B1,B2,B3),
  m = rep(1/5,3),
decreasing_stabilization

*Decreasing temperature and stabilization*

**Description**

This function allows simulating the effect of a decrease in environmental temperature, which stabilizes at a specific temperature (temp_stabilization), on the abundance of ectotherm populations.

**Usage**

```r
decreasing_stabilization(
  y_ini = c(N = 400, N = 400, N = 400),
  temp_ini = rep(35, 3),
  temp_cmin = rep(18, 3),
  temp_cmax = c(25, 28, 32),
  ro = rep(0.7, 3),
  lambda = rep(5e-05, 3),
  temp_stabilization = rep(25, 3),
  q = rep(0.03, 3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12
)
```

**Arguments**

- **y_ini**: Initial population values (must be written with its name: N).
- **temp_ini**: Initial temperature.
- **temp_cmin**: Minimum critical temperature.
- **temp_cmax**: Maximum critical temperature.
- **ro**: Population growth rate at optimum temperature.
- **lambda**: Marginal loss by non-thermodependent intraspecific competition.
- **temp_stabilization**: Stabilization temperature.
- **q**: Temperature increase factor.
- **time_start**: Start of time sequence.
- **time_end**: End of time sequence.
- **leap**: Time sequence step.
**decreasing_stabilization**

**Details**

Three populations and/or scenarios can be simulated simultaneously. A logistic type function determines the temperature trend. The temperature decreases and then stabilizes a given value. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

**Value**

1. A data.frame with columns having the simulated trends.
2. A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

**References**


**Examples**

```r
#######################################################################
#Example 1: Different initial population abundances.
#######################################################################

decreasing_stabilization(y_ini = c(N = 100, N = 200, N = 400), 
  temp_ini = rep(26,3), 
  temp_cmin = rep(18,3), 
  temp_cmax = rep(30,3), 
  ro = rep(0.7,3), 
  lambda = rep(0.00005,3), 
  temp_stabilization = rep(19,3), 
  q = rep(0.03,3), 
  time_start = 2005, 
  time_end = 2100, 
  leap = 1/12)

#######################################################################
#Example 2: Different thermal tolerance ranges.
#######################################################################

temp_cmin3 <- 18 
temp_cmin2 <- 10/9*temp_cmin3 
temp_cmin1 <- 10/9*temp_cmin2 

temp_cmax1 <- 32.4 
temp_cmax2 <- 10/9*temp_cmax1 
temp_cmax3 <- 10/9*temp_cmax2 
```
decreasing_stabilization(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = rep(32, 3),
    temp_cmin = c(temp_cmin1, temp_cmin2, temp_cmin3),
    temp_cmax = c(temp_cmax1, temp_cmax2, temp_cmax3),
    ro = rep(0.7, 3),
    lambda = rep(0.00005, 3),
    temp_stabilization = rep(19, 3),
    q = rep(0.03, 3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12)

#Example 3: Different relationships between initial environmental
# temperature and optimum temperature.

temp_cmin <- 18
temp_cmax <- 30

# Temperature at which performance is at its maximum value.
    temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-
    (temp_cmax*temp_cmin)/3)

temp_ini1 <- (temp_cmin+temp_op)/2
    temp_ini2 <- temp_op
    temp_ini3 <- (temp_op+temp_cmax)/2

decreasing_stabilization(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = c(temp_ini1, temp_ini2, temp_ini3),
    temp_cmin = rep(temp_cmin, 3),
    temp_cmax = rep(temp_cmax, 3),
    ro = rep(0.7, 3),
    lambda = rep(0.00005, 3),
    temp_stabilization = rep(19, 3),
    q = rep(0.03, 3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12)

#Example 4: Different stabilizing temperature.

temp_stabilization1 <- 18
    temp_stabilization2 <- 10/9*temp_stabilization1
    temp_stabilization3 <- 10/9*temp_stabilization2

decreasing_stabilization(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = rep(26, 3),
    temp_cmin = rep(18, 3),
    temp_cmax = rep(30, 3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
temp_stabilization = c(temp_stabilization1,
                      temp_stabilization2,
                      temp_stabilization3),
q = rep(0.03,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)

#######################################################################
#Example 5: Different marginal losses by a non-thermodependent
# component of intraspecific competition.
#######################################################################

lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

decreasing_stabilization(y_ini = c(N = 100, N = 100, N = 100),
                        temp_ini = rep(26,3),
                        temp_cmin = rep(18,3),
                        temp_cmax = rep(30,3),
                        ro = rep(0.7,3),
                        lambda = c(lambda1,lambda2,lambda3),
                        temp_stabilization = rep(19,3),
                        q = rep(0.03,3),
                        time_start = 2005,
                        time_end = 2100,
                        leap = 1/12)

---

get_RCP2.6

Projected values under IPCC RCP2.6 scenario

Description

This function allows obtaining the projected increment in environmental temperature according to
the IPCC RCP2.6 scenario (2014).

Usage

get_RCP2.6(date)

Arguments

date A specific year or a vector of years.
Details

The temperature increment projection of the change in global mean surface temperature according to the IPCC RCP2.6 scenario. It is possible to get the value for one or various years.

Value

No return value, called for side effects.

References


Examples

########################################################################
#Example 1: Projection of the temperature increase for a given year.
########################################################################
date <- 2050
temp <- get_RCP2.6(date)
temp

########################################################################
#Example 2: Projection of the temperature increase for a vector of years.
########################################################################
date <- seq(2005,2100,1/12)
temp <- get_RCP2.6(date)
plot(date,temp, type="l")

get_RCP8.5

Projected values under IPCC RCP8.5 scenario

Description

This function allows obtaining the projected increment in environmental temperature according to the IPCC RCP8.5 scenario.

Usage

get_RCP8.5(date)

Arguments

date A specific year or a vector of years.
Details

The temperature increment projection of the change in global mean surface temperature according to the IPCC RCP8.5 scenario. It is possible to get the value for one or various years.

Value

No return value, called for side effects.

References


Examples

########################################################################
#Example 1: Projection of the temperature increase for a given year.
########################################################################

date <- 2050
temp <- get_RCP8.5(date)
temp

########################################################################
#Example 2: Projection of the temperature increase for a vector of years.
########################################################################

date <- seq(2005,2100,1/12)
temp <- get_RCP8.5(date)
plot(date,temp,type="l")

heating_pulse1

Description

This function allows simulating the effect of an environmental warming pulse on the abundance of ectotherm populations. After the pulse, the temperature stabilizes at a specific value (temp_a).

Usage

heating_pulse1(  
y_ini = c(N = 400, N = 400, N = 400),
    temp_ini = rep(20, 3),
    temp_cmin = rep(18, 3),
)
```r

temp_cmax = c(25, 28, 32),
ro = rep(0.7, 3),
lambda = rep(5e-05, 3),
temp_peak = rep(25, 3),
time_peak = rep(2060, 3),
sd = rep(2, 3),
time_start = 2005,
time_end = 2100,
leap = 1/12
)
```

**Arguments**

- **y_ini**: Initial population values (must be written with its name: N).
- **temp_ini**: Initial temperatures.
- **temp_cmin**: Minimum critical temperature.
- **temp_cmax**: Maximum critical temperature.
- **ro**: Population growth rate at optimum temperature.
- **lambda**: Marginal loss by non-thermodependent intraspecific competition.
- **temp_peak**: Peak pulse temperature.
- **time_peak**: Time when temp_peak is reached.
- **sd**: Vector of standard deviations associated with temperature trend.
- **time_start**: Start of time sequence.
- **time_end**: End of time sequence.
- **leap**: Time sequence step.

**Details**

Three populations and/or scenarios can be simulated simultaneously. The temperature trend corresponds to a heating pulse determined by a Gaussian function, and the characteristics of the pulse are determined by the mean and the standard deviation. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

**Value**

1. A data.frame with columns having the simulated trends.
2. A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.
References


Examples

(Resources deleted)
heating_pulse1

temp_cmin <- 18
temp_cmax <- 30

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-(temp_cmax*temp_cmin)/3)

temp_init1 <- (temp_cmin+temp_op)/2
temp_init2 <- temp_op

temp_init3 <- (temp_op+temp_cmax)/2

heating_pulse1(y_ini = c(N = 100, N = 100, N = 100),
                temp_init = c(temp_init1,temp_init2,temp_init3),
                temp_cmin = rep(temp_cmin,3),
                temp_cmax = rep(temp_cmax,3),
                ro = rep(0.7,3),
                lambda = rep(0.00005,3),
                temp_peak = rep(29,3),
                time_peak = rep(2060,3),
                sd = rep(10,3),
                time_start = 2005,
                time_end = 2100,
                leap = 1/12)

# Example 4: Different peaks temperature.

temp_peak3 <- 30

temp_peak2 <- 9/10*temp_peak3

temp_peak1 <- 9/10*temp_peak2

heating_pulse1(y_ini = c(N = 100, N = 100, N = 100),
                temp_init = rep(22,3),
                temp_cmin = rep(18,3),
                temp_cmax = rep(30,3),
                ro = rep(0.7,3),
                lambda = rep(0.00005,3),
                temp_peak = c(temp_peak1,temp_peak2,temp_peak3),
                time_peak = rep(2060,3),
                sd = rep(10,3),
                time_start = 2005,
                time_end = 2100,
                leap = 1/12)

# Example 5: Different marginal losses by a non-thermodependent
# component of intraspecific competition.
heating_pulse2

Description

This function allows simulating the effect of an environmental warming pulse on the abundance of ectotherm populations. After the pulse, the temperature stabilizes at a temperature \( q \) units greater than the initial value (temp_ini).

Usage

heating_pulse2(
  y_ini = c(N = 400, N = 400, N = 400),
  temp_ini = rep(20, 3),
  temp_cmin = rep(18, 3),
  temp_cmax = c(25, 28, 32),
  ro = rep(0.7, 3),
  lambda = rep(5e-05, 3),
  temp_peak = rep(25, 3),
  time_peak = rep(2060, 3),
  q = rep(5, 3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12
)

Arguments

- **y_ini**: Initial population values (must be written with its name: N).
- **temp_ini**: Initial temperature.
heating_pulse2

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>temp_cmin</td>
<td>Minimum critical temperature.</td>
</tr>
<tr>
<td>temp_cmax</td>
<td>Maximum critical temperature.</td>
</tr>
<tr>
<td>ro</td>
<td>Population growth rate at optimum temperature.</td>
</tr>
<tr>
<td>lambda</td>
<td>Marginal loss by non-thermodependent intraspecific competition.</td>
</tr>
<tr>
<td>temp_peak</td>
<td>Peak pulse temperature.</td>
</tr>
<tr>
<td>time_peak</td>
<td>Time when temp_peak is reached.</td>
</tr>
<tr>
<td>q</td>
<td>Difference between initial and stabilization temperature.</td>
</tr>
<tr>
<td>time_start</td>
<td>Start of time sequence.</td>
</tr>
<tr>
<td>time_end</td>
<td>End of time sequence.</td>
</tr>
<tr>
<td>leap</td>
<td>Time sequence step.</td>
</tr>
</tbody>
</table>

Details

Three populations and/or scenarios can be simulated simultaneously. The temperature trend is determined by a rational function in which the temperature stabilizes at a different value after the pulse (that is, the final temperature differs from the initial temperature by q units). In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

Value

1. A data.frame with columns having the simulated trends.
2. A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References


Examples

```r
#Example 1: Different initial population abundances.

heating_pulse2(y_ini = c(N = 100, N = 200, N = 400),
               temp_ini = rep(20,3),
               temp_cmin = rep(18,3),
               temp_cmax = rep(30,3),
               ro = rep(0.7,3),
               lambda = rep(0.00005,3),
               # Example 2: Different initial population abundances.

heating_pulse2(y_ini = c(N = 10, N = 20, N = 40),
               temp_ini = rep(20,3),
               temp_cmin = rep(18,3),
               temp_cmax = rep(30,3),
               ro = rep(0.7,3),
               lambda = rep(0.00005,3),
               # Example 3: Different initial population abundances.

heating_pulse2(y_ini = c(N = 1, N = 2, N = 4),
               temp_ini = rep(20,3),
               temp_cmin = rep(18,3),
               temp_cmax = rep(30,3),
               ro = rep(0.7,3),
               lambda = rep(0.00005,3),
```

heating_pulse2

```r
temp_peak = rep(29,3),
time_peak = rep(2060,3),
q = rep(3,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)
```

Example 2: Different thermal tolerance ranges.

```r
temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2
```

```r
heating_pulse2(y_ini = c(N=100,N=100,N=100),
temp_ini = rep(23,3),
temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
temp_peak = rep(35,3),
time_peak = rep(2060,3),
q = rep(1,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)
```

Example 3: Different relationships between initial environmental temperature and optimum temperature.

```r
temp_cmin <- 18
temp_cmax <- 30

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-(temp_cmax*temp_cmin)/3)
temp_ini1 <- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op+temp_cmax)/2
```

```r
heating_pulse2(y_ini = c(N = 100, N = 100, N = 100),
temp_ini = c(temp_ini1,temp_ini2,temp_ini3),
temp_cmin = rep(temp_cmin,3),
temp_cmax = rep(temp_cmax,3),
```
heating_pulse2

ro = rep(0.7, 3),
lambda = rep(0.00005, 3),
temp_peak = rep(29, 3),
time_peak = rep(2000, 3),
q = rep(1, 3),
time_start = 2005,
time_end = 2100,
leap = 1/12)

#######################################################################
#Example 4: Different peaks of temperature.
#######################################################################

temp_peak3 <- 30
temp_peak2 <- 9/10*temp_peak3
temp_peak1 <- 9/10*temp_peak2

heating_pulse2(y_ini = c(N = 100, N = 100, N = 100),
temp_ini = rep(19, 3),
temp_cmin = rep(18, 3),
temp_cmax = rep(30, 3),
ro = rep(0.7, 3),
lambda = rep(0.00005, 3),
temp_peak = c(temp_peak1, temp_peak2, temp_peak3),
time_peak = rep(2000, 3),
q = rep(1, 3),
time_start = 2005,
time_end = 2100,
leap = 1/12)

#######################################################################
#Example 5: Different marginal losses by a non-thermodependent
#component of intraspecific competition.
#######################################################################

lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

heating_pulse2(y_ini = c(N = 100, N = 100, N = 100),
temp_ini = rep(26, 3),
temp_cmin = rep(18, 3),
temp_cmax = rep(30, 3),
ro = rep(0.7, 3),
lambda = c(lambda1, lambda2, lambda3),
temp_peak = rep(29, 3),
time_peak = rep(2075, 3),
q = rep(1, 3),
time_start = 2005,
time_end = 2100,
leap = 1/12)
increasing_linear  \hspace{1cm} \textit{Projection of increasing linear temperature}

\section*{Description}

This function allows simulating the effect of a linear increase in environmental temperature on the abundance of ectotherm populations.

\section*{Usage}

\begin{verbatim}
increasing_linear(
    y_ini = c(N = 100, N = 200, N = 400),
    temp_ini = rep(26, 3),
    temp_cmin = rep(18, 3),
    temp_cmax = rep(40, 3),
    ro = rep(0.7, 3),
    m = rep(0.15, 3),
    lambda = rep(5e-05, 3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12
)
\end{verbatim}

\section*{Arguments}

\begin{itemize}
\item \textbf{y_ini} \hspace{1cm} Initial population values (must be written with its name: N).
\item \textbf{temp_ini} \hspace{1cm} Initial temperature.
\item \textbf{temp_cmin} \hspace{1cm} Minimum critical temperature.
\item \textbf{temp_cmax} \hspace{1cm} Maximum critical temperature.
\item \textbf{ro} \hspace{1cm} Population growth rate at optimum temperature.
\item \textbf{m} \hspace{1cm} Temperature trend growth slope.
\item \textbf{lambda} \hspace{1cm} Marginal loss by non-thermodpendent intraspecific competition.
\item \textbf{time_start} \hspace{1cm} Start of time sequence.
\item \textbf{time_end} \hspace{1cm} End of time sequence.
\item \textbf{leap} \hspace{1cm} Time sequence step.
\end{itemize}

\section*{Details}

Three populations and/or scenarios can be simulated simultaneously. The temperature trend is determined by a linear function. The slope can be specified. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).
Value

(1) A data.frame with columns having the simulated trends.

(2) A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References


Examples

```r
#Example 1: Different initial population abundances.
increasing_linear(y_ini = c(N = 100, N = 200, N = 400),
   temp_ini = rep(26,3),
   temp_cmin = rep(18,3),
   temp_cmax = rep(40,3),
   ro = rep(0.7,3),
   m = rep(0.15,3),
   lambda = rep(0.00005,3),
   time_start = 2005,
   time_end = 2100,
   leap = 1/12)
```

```r
#Example 2: Different thermal tolerance ranges.
temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2
increasing_linear(y_ini = c(N = 100, N = 100, N = 100),
   temp_ini = rep(26,3),
   temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
   temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
   ro = rep(0.7,3),
   m = rep(0.15,3),
   lambda = rep(0.00005,3),
   time_start = 2005,
   time_end = 2100,
   leap = 1/12)
```
increasing_periodicity

leap = 1/12)

#######################################################################
#Example 3: Different relationships between initial environmental
# temperature and optimum temperature.
#######################################################################

temp_cmin <- 18
temp_cmax <- 40

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-
(temp_cmax*temp_cmin)/3)
temp_ini1 <- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op+temp_cmax)/2

increasing_linear(y_ini = c(N = 100, N = 100, N = 100),
 temp_ini = c(temp_ini1,temp_ini2,temp_ini3),
 temp_cmin = rep(temp_cmin,3),
 temp_cmax = rep(temp_cmax,3),
 ro = rep(0.7,3),
 m = rep(0.15,3),
 lambda = rep(0.00005,3),
 time_start = 2005,
 time_end = 2100,
 leap = 1/12)

#######################################################################
#Example 4: Different marginal losses by a non-thermodependent
# component of intraspecific competition.
#######################################################################

lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

increasing_linear(y_ini = c(N = 100, N = 100, N = 100),
 temp_ini = rep(26,3),
 temp_cmin = rep(18,3),
 temp_cmax = rep(36,3),
 ro = rep(0.7,3),
 m = rep(0.15,3),
 lambda = c(lambda1,lambda2,lambda3),
 time_start = 2005,
 time_end = 2100,
 leap = 1/12)
increasing_periodicity

*Periodic increasing temperature trend*

**Description**

This function allows simulating the effect of increasing temperature with periodic variability on the abundance of ectotherm populations.

**Usage**

```r
increasing_periodicity(
  y_ini = c(N = 400, N = 400, N = 400),
  temp_ini = rep(25, 3),
  temp_cmin = rep(18, 3),
  temp_cmax = c(25, 28, 32),
  ro = rep(0.7, 3),
  lambda = rep(5e-05, 3),
  A = rep(5, 3),
  B = rep(0.06, 3),
  m = rep(1/5, 3),
  time_start = 2005,
  time_end = 2300,
  leap = 1/12
)
```

**Arguments**

- `y_ini`: Initial population values (must be written with its name: N).
- `temp_ini`: Initial temperature.
- `temp_cmin`: Minimum critical temperature.
- `temp_cmax`: Maximum critical temperature.
- `ro`: Population growth rate at optimal temperature.
- `lambda`: Marginal loss by non-thermodependent intraspecific competition.
- `A`: Temperature wave amplitude.
- `B`: Parameter affecting the period of the trend (period is (2 pi)/|B|).
- `m`: Temperature increase slope.
- `time_start`: Start of time sequence.
- `time_end`: End of time sequence.
- `leap`: Time sequence step.
Details
Three populations and/or scenarios can be simulated simultaneously. The temperature trend is determined by an expression that shows a linear increment with periodic variability. The amplitude, period, and downward speed of change can be specified. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

Value
(1) A data.frame with columns having the simulated trends.
(2) A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References

Examples

```r
#Example 1: Different initial population abundances.
increasing_periodicity(y_ini = c(N = 100, N = 200, N = 400),
  temp_ini = rep(19,3),
  temp_cmin = rep(18,3),
  temp_cmax = rep(35,3),
  ro = rep(0.7,3),
  lambda = rep(0.00005,3),
  A = rep(2,3),
  B = rep(0.6,3),
  m = rep(1/5,3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)
```

```r
#Example 2: Different thermal tolerance ranges.
temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
```
temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2

increasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
                       temp_ini = rep(23,3),
                       temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
                       temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
                       ro = rep(0.7,3),
                       lambda = rep(0.00005,3),
                       A = rep(2,3),
                       B = rep(0.6,3),
                       m = rep(1/5,3),
                       time_start = 2005,
                       time_end = 2300,
                       leap = 1/12)

#Example 3: Different relationships between initial environmental temperature and optimum temperature.

temp_cmin <- 18
temp_cmax <- 40

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-
                                         (temp_cmax*temp_cmin)/3)

temp_ini1 <- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op+temp_cmax)/2

increasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
                       temp_ini = c(temp_ini1,temp_ini2,temp_ini3),
                       temp_cmin = rep(temp_cmin,3),
                       temp_cmax = rep(temp_cmax,3),
                       ro = rep(0.7,3),
                       lambda = rep(0.00005,3),
                       A = rep(2,3),
                       B = rep(0.6,3),
                       m = rep(1/5,3),
                       time_start = 2005,
                       time_end = 2300,
                       leap = 1/12)

#Example 4: Different marginal losses by a non-thermodependent component of intraspecific competition.

lambda3 <- 0.01
\begin{verbatim}
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

increasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = rep(19,3),
    temp_cmin = rep(18,3),
    temp_cmax = rep(35,3),
    ro = rep(0.7,3),
    lambda = c(lambda1,lambda2,lambda3),
    A = rep(2,3),
    B = rep(0.6,3),
    m = rep(1/5,3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12)

#Example 5: Different wave amplitude.
A3 <- 2
A2 <- 1/2 * A3
A1 <- 1/2 * A2

increasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = rep(22,3),
    temp_cmin = rep(18,3),
    temp_cmax = rep(35,3),
    ro = rep(0.7,3),
    lambda = rep(0.00005,3),
    A = c(A1,A2,A3),
    B = rep(0.6,3),
    m = rep(1/5,3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12)

#Example 6: Different period.
B3 <- pi/5
B2 <- 1/2 * B3
B1 <- 1/2 * B2

increasing_periodicity(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = rep(22,3),
    temp_cmin = rep(18,3),
    temp_cmax = rep(35,3),
    ro = rep(0.7,3),
    lambda = rep(0.00005,3),
    A = rep(2,3),
    B = c(B1,B2,B3),
\end{verbatim}
increasing_stabilization

Increasing temperature and stabilization

Description

This function allows simulating the effect of an increase in environmental temperature, which stabilizes at a specific value, on the abundance of ectotherm populations.

Usage

```r
increasing_stabilization(
    y_ini = c(N = 400, N = 400, N = 400),
    temp_ini = rep(20, 3),
    temp_cmin = rep(18, 3),
    temp_cmax = c(25, 28, 32),
    ro = rep(0.7, 3),
    lambda = rep(5e-05, 3),
    temp_stabilization = rep(25, 3),
    q = rep(0.03, 3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12
)
```

Arguments

- `y_ini`     Initial population values (must be written with its name: N).
- `temp_ini`  Initial temperature.
- `temp_cmin` Minimum critical temperature.
- `temp_cmax` Maximum critical temperature.
- `ro`        Population growth rate at optimum temperature.
- `lambda`    Marginal loss by non-thermodependent intraspecific competition.
- `temp_stabilization`  Stabilization temperature.
- `q`         Temperature increase factor.
- `time_start` Start of time sequence.
- `time_end`  End of time sequence.
- `leap`      sequence increase.
Details

Three populations and/or scenarios can be simulated simultaneously. A logistic type function determines the temperature trend. The temperature increases and then stabilizes at a specific value. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

Value

(1) A data.frame with columns having the simulated trends.
(2) A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References


Examples

#########################################################################
#Example 1: Different initial population abundances.
#########################################################################

increasing_stabilization(y_ini = c(N = 100, N = 200, N = 400),
                          temp_ini = rep(26,3),
                          temp_cmin = rep(18,3),
                          temp_cmax = rep(30,3),
                          ro = rep(0.7,3),
                          lambda = rep(0.00005,3),
                          temp_stabilization = rep(30,3),
                          q = rep(0.03,3),
                          time_start = 2005,
                          time_end = 2100,
                          leap = 1/12)

#########################################################################
#Example 2: Different thermal tolerance ranges.
#########################################################################

temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2

temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2
```r
increasing_stabilization(y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = rep(30, 3),
  temp_cmin = c(temp_cmin1, temp_cmin2, temp_cmin3),
  temp_cmax = c(temp_cmax1, temp_cmax2, temp_cmax3),
  ro = rep(0.7, 3),
  lambda = rep(0.00005, 3),
  temp_stabilization = rep(33, 3),
  q = rep(0.03, 3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)
```

#Example 3: Different relationships between initial environmental temperature and optimum temperature.

```r
temp_cmin <- 18
temp_cmax <- 30

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax + temp_cmin)/3 + sqrt(((temp_cmax + temp_cmin)/3)^2 - (temp_cmax*temp_cmin)/3)
temp_ini1 <- (temp_cmin + temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op + temp_cmax)/2

increasing_stabilization(y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = c(temp_ini1, temp_ini2, temp_ini3),
  temp_cmin = rep(temp_cmin, 3),
  temp_cmax = rep(temp_cmax, 3),
  ro = rep(0.7, 3),
  lambda = rep(0.00005, 3),
  temp_stabilization = rep(32, 3),
  q = rep(0.03, 3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)
```

#Example 4: Different stabilization temperature.

```r
temp_stabilization3 <- 42
temp_stabilization2 <- 14/13*temp_stabilization3
temp_stabilization1 <- 14/13*temp_stabilization2

increasing_stabilization(y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = rep(30, 3),
  temp_cmin = rep(18, 3),
  temp_cmax = rep(40, 3),
```

# Example 5: Different marginal losses by a non-thermodependent component of intraspecific competition.

```r
lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

increasing_stabilization(y_ini = c(N = 100, N = 100, N = 100),
                          temp_ini = rep(30, 3),
                          temp_cmin = rep(18, 3),
                          temp_cmax = rep(40, 3),
                          ro = rep(0.7, 3),
                          lambda = c(lambda1, lambda2, lambda3),
                          temp_stabilization = rep(35, 3),
                          q = rep(0.03, 3),
                          time_start = 2005,
                          time_end = 2100,
                          leap = 1/12)
```

---

**IPCC_RCP2_6**  
*IPCC RCP2.6 scenario*

**Description**

This function allows simulating the effect of an increase in environmental temperature according to the IPCC RCP2.6 scenario (2014) on the abundance of ectotherm populations.

**Usage**

```r
IPCC_RCP2_6(
    y_ini = c(N = 400, N = 400, N = 400),
    temp_ini = rep(20, 3),
    temp_cmin = rep(18, 3),
    temp_cmax = c(20, 25, 36),
    ro = rep(0.7, 3),
    lambda = rep(5e-05, 3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12)
```
Arguments

- `y_ini`  
  Initial population values (must be written with its name: N).
- `temp_ini`  
  Initial temperature.
- `temp_cmin`  
  Minimum critical temperature.
- `temp_cmax`  
  Maximum critical temperature.
- `ro`  
  Population growth rate at optimum temperature.
- `lambda`  
  Marginal loss by non-thermodependent intraspecific competition.
- `time_start`  
  Start of time sequence.
- `time_end`  
  End of time sequence.
- `leap`  
  Time sequence step.

Details

Three populations can be simulated simultaneously. The temperature trend is determined by a projection of the change in global mean surface temperature according to the IPCC RCP2.6 scenario. In each input vector, the parameters for the three simulations must be specified (finite numbers for initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

Value

1. A data.frame with columns having the simulated trends.
2. A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References


Examples

```r
# Example 1: Different initial population abundances.
```
IPCC_RCP2_6

(y ini = c(N = 100, N = 200, N = 400),
temp ini = rep(26,3),
temp cmin = rep(18,3),
temp cmax = rep(40,3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
time start = 2005,
time end = 2100,
leap = 1/12)

#######################################################################
#Example 2: Different thermal tolerance ranges.
#######################################################################

temp cmin3 <- 18
temp cmin2 <- 10/9*temp cmin3
temp cmin1 <- 10/9*temp cmin2
temp cmax1 <- 32.4
temp cmax2 <- 10/9*temp cmax1
temp cmax3 <- 10/9*temp cmax2

IPCC_RCP2_6(y ini = c(N = 100, N = 100, N = 100),
temp ini = rep(26,3),
temp cmin = c(temp cmin1,temp cmin2,temp cmin3),
temp cmax = c(temp cmax1,temp cmax2,temp cmax3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
time start = 2005,
time end = 2100,
leap = 1/12)

#######################################################################
#Example 3: Different relationships between initial environmental
# temperature and optimum temperature.
#######################################################################

temp cmin <- 18
temp cmax <- 40

# Temperature at which performance is at its maximum value.
temp op <- (temp cmax+temp cmin)/3+sqrt(((temp cmax+temp cmin)/3)^2-
(temp cmax*temp cmin)/3)
temp ini1 <- (temp cmin+temp op)/2
temp ini2 <- temp op
temp ini3 <- (temp op+temp cmax)/2

IPCC_RCP2_6(y ini = c(N = 100, N = 100, N = 100),
temp ini = c(temp ini1,temp ini2,temp ini3),
temp cmin = rep(temp cmin,3),
temp cmax = rep(temp cmax,3),
Description

This function allows simulating the effect of an increase in environmental temperature according to the IPCC RCP8.5 scenario (2014) on the abundance of ectotherm populations.

Usage

```R
IPCC_RCP8_5(
  y_ini = c(N = 400, N = 400, N = 400),
  temp_ini = rep(20, 3),
  temp_cmin = rep(18, 3),
  temp_cmax = c(25, 28, 30),
  ro = rep(0.7, 3),
  lambda = rep(5e-05, 3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12
)
```
Arguments

- **y_ini**: Initial population values (must be written with its name: N).
- **temp_ini**: Initial temperature.
- **temp_cmin**: Minimum critical temperature.
- **temp_cmax**: Maximum critical temperature.
- **ro**: Population growth rate at optimum temperature.
- **lambda**: Marginal loss by non-thermodependent intraspecific competition.
- **time_start**: Start of time sequence.
- **time_end**: End of time sequence.
- **leap**: Time sequence step.

Details

Three populations can be simulated simultaneously. The temperature trend is determined by a projection of the change in global mean surface temperature according to the IPCC RCP8.5 scenario. In each input vector, the parameters for the three simulations must be specified (finite numbers for initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

Value

1. A data.frame with columns having the simulated trends.
2. A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References


Examples

```r
#############################
#Example 1: Different initial population abundances.
#############################
IPCC_RCP8_5(y_ini = c(N = 100, N = 200, N = 400),
             temp_ini = rep(26,3),
             temp_cmin = rep(18,3),
```
temp_cmax = rep(30,3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)

#Example 2: Different thermal tolerance ranges.

temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2

IPCC_RCP8_5(y_ini = c(N = 100, N = 100, N = 100),
temp_ini = rep(30,3),
temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)

#Example 3: Different relationships between initial environmental
# temperature and optimum temperature.

temp_cmin <- 18
temp_cmax <- 30

temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-
(temp_cmax*temp_cmin)/3)
temp_ini1 <- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op+temp_cmax)/2

IPCC_RCP8_5(y_ini = c(N = 100, N = 100, N = 100),
temp_ini = c(temp_ini1,temp_ini2,temp_ini3),
temp_cmin = rep(temp_cmin,3),
temp_cmax = rep(temp_cmax,3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
time_start = 2005,
time_end = 2100,
predation

predation

Description

This function allows simulating the effect of the IPCC RCP2.6 or RCP8.5 scenarios (2014) on the abundances of two species interacting through predation. The prey is an ectotherm population, and the predator is not affected by temperature.

Usage

predation(
  y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = rep(25, 3),
  temp_cmin = rep(18, 3),
  temp_cmax = rep(30, 3),
  ro = rep(0.7, 3),
  lambda = c(lambda1, lambda2, lambda3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)

Example 4: Different marginal losses by a non-thermodependent component of intraspecific competition.

lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

IPCC_RCP8_5(y_ini = c(N = 100, N = 100, N = 100),
            temp_cmin = rep(18, 3),
            temp_ini = rep(25, 3),
            temp_cmax = rep(30, 3),
            ro = rep(0.7, 3),
            lambda = c(lambda1, lambda2, lambda3),
            time_start = 2005,
            time_end = 2100,
            leap = 1/12)
Arguments

- **y_ini**: Initial population values (must be written with its name: N).
- **temp_ini**: Initial temperature.
- **temp_cmin**: Minimum critical temperature (prey).
- **temp_cmax**: Maximum critical temperature (prey).
- **ro**: Population growth rate at optimum temperature (prey).
- **lambda**: Marginal loss by non-thermodependent intraspecific competition (prey).
- **e**: Efficiency with which food is converted into population growth (predator).
- **mp**: Mortality rate (predator).
- **q**: Maximum per capita consumption rate (predator).
- **a**: Mean saturation rate (predator).
- **RCP**: Representative concentration trajectories (RCP2.6 and RCP8.5 scenarios).
- **time_start**: Start of time sequence.
- **time_end**: End of time sequence.
- **leap**: Time sequence step.

Details

Three scenarios can be evaluated for a predation interaction where the prey is an ectotherm population. The temperature trends correspond to IPCC projections under the RCP2.6 or RCP8.5 scenarios. In each input vector, the parameters for the three simulations must be specified.

Value

1. A data.frame with columns having the simulated trends.
2. A four-panel figure where (a), (b), and (c) show the population abundance curves for each simulation. The brown curve corresponds to the abundance of prey and the green curve to predators. Panel (d) shows the temperature trend curves used for each simulation, green, blue, and black, respectively.

References


Examples

#Example 1: Different thermal tolerance ranges (scenario RCP2.6).

temp_cmin <- 18
# Temperature that occurs before the minimum simulation time.
\[
\text{temp}_i \leftarrow 22
\]

\[
\text{time}_\text{end} \leftarrow 2100
\]

# Temperature that occurs in the maximum time of the simulation.
\[
\text{temp}_\text{max} \leftarrow \text{get}_\text{RCP2.6}(\text{time}_\text{end})+\text{temp}_i
\]

# Simulation thermal range.
\[
\text{RS} \leftarrow \text{temp}_\text{max}-\text{temp}_\text{cmin}
\]

\[
\text{temp}_\text{cmax}1 \leftarrow \frac{4}{3}\times\text{RS}+\text{temp}_\text{cmin}
\]
\[
\text{temp}_\text{cmax}2 \leftarrow \frac{2}{3}\times\text{RS}+\text{temp}_\text{cmin}
\]
\[
\text{temp}_\text{cmax}3 \leftarrow \frac{1}{3}\times\text{RS}+\text{temp}_\text{cmin}
\]
\[
\text{temp}_\text{ini} \leftarrow \frac{\text{temp}_\text{cmin}+\text{temp}_\text{cmax}3}{2}
\]

\[
\text{predation}(y_{\_\text{ini}}=c(V=800, V=800, V=800, \\
\hspace{1cm} P=600, P=600, P=600), \\
\hspace{1cm} \text{temp}_\text{ini} = \text{rep}(\text{temp}_\text{ini},3), \\
\hspace{1cm} \text{temp}_\text{cmin} = \text{rep}(\text{temp}_\text{cmin},3), \\
\hspace{1cm} \text{temp}_\text{cmax} = \text{c}(\text{temp}_\text{cmax}1, \text{temp}_\text{cmax}2, \text{temp}_\text{cmax}3), \\
\hspace{1cm} \text{ro} = \text{rep}(0.7,3), \\
\hspace{1cm} \text{lambda} = \text{rep}(0.0004,3), \\
\hspace{1cm} \text{e} = \text{rep}(0.9,3), \\
\hspace{1cm} \text{mp} = \text{rep}(0.1,3), \\
\hspace{1cm} \text{q} = \text{rep}(0.7,3), \\
\hspace{1cm} \text{a} = \text{rep}(1000,3), \\
\hspace{1cm} \text{RCP} = 2.6, \\
\hspace{1cm} \text{time}_\text{start} = 2005, \\
\hspace{1cm} \text{time}_\text{end} = \text{time}_\text{end}, \\
\hspace{1cm} \text{leap} = 1/50)
\]

# Example 2: Different thermal tolerance ranges (scenario RCP8.5).

# Temperature that occurs before the minimum simulation time.
\[
\text{temp}_i \leftarrow 22
\]

\[
\text{time}_\text{end} \leftarrow 2100
\]

# Temperature that occurs in the maximum time of the simulation.
\[
\text{temp}_\text{max} \leftarrow \text{get}_\text{RCP8.5}(\text{time}_\text{end})+\text{temp}_i
\]

# Simulation thermal range.
\[
\text{RS} \leftarrow \text{temp}_\text{max}-\text{temp}_\text{cmin}
\]

\[
\text{temp}_\text{cmax}1 \leftarrow \frac{4}{3}\times\text{RS}+\text{temp}_\text{cmin}
\]
\[
\text{temp}_\text{cmax}2 \leftarrow \frac{2}{3}\times\text{RS}+\text{temp}_\text{cmin}
\]
\[
\text{temp}_\text{cmax}3 \leftarrow \frac{1}{3}\times\text{RS}+\text{temp}_\text{cmin}
\]
\[
\text{temp}_\text{ini} \leftarrow \frac{\text{temp}_\text{cmin}+\text{temp}_\text{cmax}3}{2}
\]
predation(y_ini = c(V = 800, V = 800, V = 800,
    P = 600, P = 600, P = 600),
    temp_ini = rep(temp_ini,3),
    temp_cmin = rep(temp_cmin,3),
    temp_cmax = c(temp_cmax1, temp_cmax2, temp_cmax3),
    ro = rep(0.7,3),
    lambda = rep(0.0004,3),
    e = rep(0.9,3),
    mp = rep(0.1,3),
    q = rep(0.7,3),
    a = rep(1000,3),
    RCP = 8.5,
    time_start = 2005,
    time_end = time_end,
    leap = 1/50)

#######################################################################
#Example 3: Different conversion efficiencies (scenario RCP2.6).
#######################################################################

e1 <- 0.2
e2 <- 2*e1
e3 <- 2*e2

predation(y_ini = c(V = 800, V = 800, V = 800,
    P = 400, P = 400, P = 400),
    temp_ini = rep(22,3),
    temp_cmin = rep(20,3),
    temp_cmax = rep(35,3),
    ro = rep(0.9,3),
    lambda = rep(0.0006,3),
    e = c(e1,e2,e3),
    mp = rep(0.1,3),
    q = rep(0.7,3),
    a = rep(800,3),
    RCP = 2.6,
    time_start = 2005,
    time_end = 2100,
    leap = 1/50)

#######################################################################
#Example 4: Different conversion efficiencies (scenario RCP8.5).
#######################################################################

e1 <- 0.2
e2 <- 2*e1
e3 <- 2*e2

predation(y_ini = c(V = 800, V = 800, V = 800,
    P = 400, P = 400, P = 400),
    temp_ini = rep(22,3),
    temp_cmin = rep(20,3),
    temp_cmax = rep(35,3),
    ro = rep(0.9,3),
    lambda = rep(0.0006,3),
    e = c(e1,e2,e3),
    mp = rep(0.1,3),
    q = rep(0.7,3),
    a = rep(800,3),
    RCP = 8.5,
rate_adjustment

**Description**

This function allows you to adjust the intrinsic growth rate of an ectothermic population using temperature and growth rate data obtained empirically, using a cubic TPC (Saldaña et al., 2019).

**Usage**

```r
rate_adjustment(data = data)
```

**Arguments**

- `data` database where the first column shows the ambient temperature (TA) and the second column contains the intrinsic growth rate (R) associated with them.

**Details**

This function allows you to adjust the intrinsic growth rate of an ectotherm population by providing as input the values of environmental or body temperature together with the growth rate, using the `nls2` function you can find the parameters `temp_cmax`, `temp_cmax` and `ro`, which are necessary to adjust the curve through a cubic polynomial which fulfills the essential conditions of a TPC.

**Value**

A figure showing the fitting curve corresponding to the intrinsic growth rate of an ectothermic population, the empirically obtained temperature and growth rate data points, in addition, called for side effects.
References


Examples

# Example 1: We consider a population of Macrolophus pygmaeus whose intrinsic growth rate is adjusted to the data obtained from Rezende and Bozinovic (2019).

```r
github_link <- "https://github.com/Victor-Saldana/epcc/raw/main/M_pygmaeus.xlsx"
library(httr)
temp_file <- tempfile(fileext = ".xlsx")
req <- GET(github_link,
           authenticate(Sys.getenv("GITHUB_PAT"), ""),
           write_disk(path = temp_file))
M_pygmaeus <- readxl::read_excel(temp_file)
TPC <- rate_adjustment(data = M_pygmaeus)
```

# Example 2: We consider a population of Eretmocerus furuhashii whose intrinsic growth rate is adjusted to the data obtained from Rezende and Bozinovic (2019).

```r
library(httr)
temp_file <- tempfile(fileext = ".xlsx")
req <- GET(github_link,
           authenticate(Sys.getenv("GITHUB_PAT"), ""),
           write_disk(path = temp_file))
E_furuhashii <- readxl::read_excel(temp_file)
TPC <- rate_adjustment(data = E_furuhashii)
```

# Example 3: We consider a population of Trichogramma pretoisum whose intrinsic growth rate is adjusted to the data obtained from Rezende and Bozinovic (2019).

```r
github_link <- "https://github.com/Victor-Saldana/epcc/raw/main/T_pretoisum.xlsx"
library(httr)
temp_file <- tempfile(fileext = ".xlsx")
req <- GET(github_link,
```
rate_TPC

Intrinsic growth rate dependent on temperature

Description

This function allows to obtain the intrinsic growth rate following a thermal performance curve (TPC). A cubic polynomial is used considering Saldaña et al (2019).

Usage

rate_TPC(T, ro, temp_cmin, temp_cmax, temp_op)

Arguments

T
Temperature trend at which the population dynamics are studied.

ro
Population growth rate at optimal temperature.

temp_cmin
Minimum critical temperature.

temp_cmax
Maximum critical temperature.

temp_op
Optimum performance temperature.

Details

The intrinsic growth rate is represented by a thermal performance curve (TPC). These curves associate the performance of ectothermic organisms as a function of body temperature. These curves have a characteristic unimodal asymmetric shape skewed to the left. Its main descriptors are minimum and maximum critical temperatures, which indicate the thermal tolerance range, and the optimum temperature, which indicates the temperature at which performance is at its maximum value (Anguilleta, 2006; Huey et al., 2012). The function implements a cubic expression that follows the characteristic shape of TPCs as described in Saldaña et al. (2019).

Value

No return value, called for side effects.
trend_periodic

References


Examples

times<- seq(2005, 2100, 1/12)
temp_cmin <- 18
temp_cmax <- 26

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-(temp_cmax*temp_cmin)/3)

ro <- 0.8

# Temperature that occurs in the minimum time of the simulation.
temp_i <- 20

temp <- get_RCP8.5(times)+temp_i

rate <- rate_TPC(temp,ro,temp_cmin,temp_cmax,temp_op)

plot(times,rate, type="l")

trend_periodic

Description

This function allows simulating the effect of a periodic temperature trend on the abundance of ectotherm populations.

Usage

trend_periodic(  
y_ini = c(N = 400, N = 400, N = 400),  
temp_ini = rep(20, 3),  
temp_cmin = rep(18, 3),
)
temp_cmax = c(25, 28, 32),  
ro = rep(0.7, 3),  
lambda = rep(5e-05, 3),  
A = rep(0.5, 3),  
B = rep(0.35, 3),  
time_start = 2005,  
time_end = 2100,  
leap = 1/12  
)

Arguments

y_ini Initial population values (must be written with its name: N).
temp_ini Initial temperature.
temp_cmin Minimum critical temperature.
temp_cmax Maximum critical temperature.
ro Population growth rate at optimum temperature.
lambda Marginal loss by non-thermodependent intraspecific competition.
A Temperature wave amplitude.
B Parameter affecting the period of the trend (period is (2 pi)/|B|).
time_start Start of time sequence.
time_end End of time sequence.
leap Time sequence step.

Details

Three populations and/or scenarios can be simulated simultaneously. The temperature trend is determined by a trigonometric function characterized by amplitude and period. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

Value

(1) A data.frame with columns having the simulated trends.
(2) A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References

Examples

trend_periodic(y_ini = c(N = 100, N = 200, N = 400),
temp_ini = rep(22,3),
temp_cmin = rep(18,3),
temp_cmax = rep(35,3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
A = rep(0.2,3),
B = rep(0.6,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)

temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2

trend_periodic(y_ini = c(N = 100, N = 100, N = 100),
temp_ini = rep(30,3),
temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
ro = rep(0.7,3),
lambda = rep(0.00005,3),
A = rep(2,3),
B = rep(0.6,3),
time_start = 2005,
time_end = 2100,
leap = 1/12)

temp_cmin <- 18
temp_cmax <- 35

temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-
64

trend_periodic

(temp_cmax*temp_cmin)/3)

temp_ini1 <- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op+temp_cmax)/2

trend_periodic(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = c(temp_ini1,temp_ini2,temp_ini3),
    temp_cmin = rep(temp_cmin,3),
    temp_cmax = rep(temp_cmax,3),
    ro = rep(0.7,3),
    lambda = rep(0.00005,3),
    A = rep(2,3),
    B = rep(0.6,3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12)

#######################################################################
#Example 4: Different marginal losses by a non-thermodependent
# component of intraspecific competition.
#######################################################################

lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

trend_periodic(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = rep(22,3),
    temp_cmin = rep(18,3),
    temp_cmax = rep(35,3),
    ro = rep(0.7,3),
    lambda = c(lambda1,lambda2,lambda3),
    A = rep(2,3),
    B = rep(0.6,3),
    time_start = 2005,
    time_end = 2100,
    leap = 1/12)

#######################################################################
#Example 5: Different wave amplitudes.
#######################################################################

A3 <- 4
A2 <- 1/2*A3
A1 <- 1/2*A2

trend_periodic(y_ini = c(N = 100, N = 100, N = 100),
    temp_ini = rep(25,3),
    temp_cmin = rep(18,3),
    temp_cmax = rep(35,3),
    ro = rep(0.7,3),
variability

\[
\begin{align*}
\lambda &= \text{rep}(0.00005, 3), \\
A &= \text{c}(A1, A2, A3), \\
B &= \text{rep}(0.6, 3), \\
time\text{\_start} &= 2005, \\
time\text{\_end} &= 2100, \\
\text{leap} &= 1/12)
\end{align*}
\]

#Example 6: Different periods.

\[
\begin{align*}
B3 &= \pi/5 \\
B2 &= 1/2 \times B3 \\
B1 &= 1/2 \times B2
\end{align*}
\]

\[
\begin{align*}
\text{trend\_periodic}(y\_ini = \text{c}(N = 100, N = 100, N = 100), \\
\text{temp\_cmin} &= \text{rep}(18, 3), \\
\text{temp\_ini} &= \text{rep}(22, 3), \\
\text{temp\_cmax} &= \text{rep}(35, 3), \\
\text{ro} &= \text{rep}(0.7, 3), \\
\lambda &= \text{rep}(0.00005, 3), \\
A &= \text{rep}(2, 3), \\
B &= \text{c}(B1, B2, B3), \\
time\text{\_start} &= 2005, \\
time\text{\_end} &= 2100, \\
\text{leap} &= 1/12)
\end{align*}
\]

variability

Description

This function allows simulating the effect of increasing variable temperature trend on the abundance of ectotherm populations.

Usage

variability(
y\_ini = \text{c}(N = 400, N = 400, N = 400), \\
\text{temp\_ini} = \text{rep}(20, 3), \\
\text{temp\_cmin} = \text{rep}(18, 3), \\
\text{temp\_cmax} = \text{c}(25, 28, 32), \\
\text{ro} = \text{rep}(0.7, 3), \\
\lambda &= \text{rep}(5e-05, 3), \\
\text{mean} &= \text{rep}(5, 3), \\
\text{sd} &= \text{rep}(2, 3), \\
time\text{\_start} &= 2005, \\
time\text{\_end} &= 2100,
\[
\text{leap} = 1/12
\]

**Arguments**

- **y_ini**  
  Initial population values (must be written with its name: N).
- **temp_ini**  
  Initial temperature.
- **temp_cmin**  
  Minimum critical temperature.
- **temp_cmax**  
  Maximum critical temperature.
- **ro**  
  Population growth rate at optimum temperature.
- **lambda**  
  Marginal loss by non-thermodependent intraspecific competition.
- **mean**  
  Average noise.
- **sd**  
  Standard deviation of noise.
- **time_start**  
  Start of time sequence.
- **time_end**  
  End of time sequence.
- **leap**  
  Time sequence step.

**Details**

Three populations and/or scenarios can be simulated simultaneously. The temperature trend considers the IPCC RCP8.5 but adding random noise following a normal distribution, which is characterized by the mean and standard deviation. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010).

**Value**

1. A data.frame with columns having the simulated trends.
2. A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

**References**


**Examples**

```r
# Example 1: Different initial population abundances.
```
variability(y_ini = c(N = 100, N = 200, N = 400),
  temp_ini = rep(22,3),
  temp_cmin = rep(18,3),
  temp_cmax = rep(35,3),
  ro = rep(0.7,3),
  lambda = rep(0.00005,3),
  mean = rep(2,3),
  sd = rep(3,3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)

#######################################################################
#Example 2: Different thermal tolerance ranges.
#######################################################################

temp_cmin3 <- 18
temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
temp_cmax1 <- 32.4
temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2

variability(y_ini = c(N = 100, N = 100, N = 100),
  temp_ini = rep(26,3),
  temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
  temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
  ro = rep(0.7,3),
  lambda = rep(0.00005,3),
  mean = rep(2,3),
  sd = rep(3,3),
  time_start = 2005,
  time_end = 2100,
  leap = 1/12)

#######################################################################
#Example 3: Different relationships between initial environmental
# temperature and optimum temperature.
#######################################################################

temp_cmin <- 18
temp_cmax <- 40

# Temperature at which performance is at its maximum value.
temp_op <- (temp_cmax+temp_cmin)/3+sqrt(((temp_cmax+temp_cmin)/3)^2-
  (temp_cmax*temp_cmin)/3)
temp_ini1 <- (temp_cmin+temp_op)/2
temp_ini2 <- temp_op
temp_ini3 <- (temp_op+temp_cmax)/2

variability(y_ini = c(N = 100, N = 100, N = 100),
```r
temp_init = c(temp_init1, temp_init2, temp_init3),
temp_cmin = rep(temp_cmin, 3),
temp_cmax = rep(temp_cmax, 3),
ro = rep(0.7, 3),
lambda = rep(0.00005, 3),
mean = rep(1, 3),
sd = rep(0.5, 3),
time_start = 2005,
time_end = 2100,
leap = 1/12)
```

# Example 4: Different marginal losses by a non-thermodependent component of intraspecific competition.

```r
lambda3 <- 0.01
lambda2 <- 1/2 * lambda3
lambda1 <- 1/2 * lambda2

variability(y_init = c(N = 100, N = 100, N = 100),
temp_init = rep(26, 3),
temp_cmin = rep(18, 3),
temp_cmax = rep(35, 3),
ro = rep(0.7, 3),
lambda = c(lambda1, lambda2, lambda3),
mean = rep(1, 3),
sd = rep(0.5, 3),
time_start = 2005,
time_end = 2100,
leap = 1/12)
```

# Example 5: Different noise means.

```r
mean3 <- 2
mean2 <- 1/2 * mean3
mean1 <- 1/2 * mean2

variability(y_init = c(N = 100, N = 100, N = 100),
temp_init = rep(22, 3),
temp_cmin = rep(18, 3),
temp_cmax = rep(35, 3),
ro = rep(0.7, 3),
lambda = rep(0.00005, 3),
mean = c(mean1, mean2, mean3),
sd = rep(0.5, 3),
time_start = 2005,
time_end = 2100,
leap = 1/12)
```
# Example 6: Different noise standard deviations.

```
sd3 <- 4
sd2 <- 1/2*sd3
sd1 <- 1/2*sd2

variability(y_ini = c(N = 100, N = 100, N = 100),
            temp_ini = rep(22,3),
            temp_cmin = rep(18,3),
            temp_cmax = rep(35,3),
            ro = rep(0.7,3),
            lambda = rep(0.00005,3),
            mean = rep(3,3),
            sd = c(sd1,sd2,sd3),
            time_start = 2005,
            time_end = 2100,
            leap = 1/12)
```

---

**Description**

This function allows simulating the effect of temperature trends on the abundance of ectotherm populations in different geographic locations. Temperature data is obtained from WorldClim.

**Usage**

```
w_clim(y_ini = c(N = 400, N = 400, N = 400),
        temp_cmin = rep(18,3),
        temp_cmax = c(25, 28, 32),
        ro = rep(0.7, 3),
        lambda = rep(5e-05, 3),
        lat = rep(-33, 3),
        lon = rep(-71, 3),
        s = 10,
        res = 5,
        time_start = 2000,
        time_end = 2070,
        leap = 1/12)
```

**Arguments**

- **y_ini**: Initial population values (must be written with its name: N).
Details

Three populations and/or scenarios can be simulated simultaneously. The temperature trends are obtained by data extracted from WorldClim for the years 2000, 2050 and 2070 at a specific location. The function internally calls the function getData of the raster package (Hijmans, 2020) to obtain the bioclimatic variable of interest given a spatial resolution. An exponential expression is fitted using the nls function. In each input vector, the parameters for the three simulations must be specified (finite numbers for the initial population abundance). The simulations are obtained by a model that incorporates the effects of temperature over time, which leads to a non-autonomous ODE approach. This is function uses the ODE solver implemented in the package deSolve (Soetaert et al., 2010). In the first three examples, three geographic locations are considered for Macrolophus pygmaeus as reported in Sánchez et al. (2012).

Value

(1) A data.frame with columns having the simulated trends.
(2) A two-panel figure in which (a) shows the population abundance curves represented by solid lines and the corresponding carrying capacities are represented by shaded areas. In (b) the temperature trend is shown. The three simultaneous simulations are depicted by different colors, i.e. 1st brown, 2nd green and 3rd blue.

References

Examples

```r
w_clim(y_ini = c(N = 100, N = 200, N = 400),
    temp_cmin = rep(18,3),
    temp_cmax = rep(30,3),
    ro = rep(0.7,3),
    lambda = rep(0.00005,3),
    lat = rep(-33,3),
    lon = rep(-71,3),
    s = 5,
    res = 5,
    time_start = 2000,
    time_end = 2070,
    leap = 1/12)
```

```r
temp_cmin3 <- 18
    temp_cmin2 <- 10/9*temp_cmin3
temp_cmin1 <- 10/9*temp_cmin2
temp_cmax1 <- 32.4
    temp_cmax2 <- 10/9*temp_cmax1
temp_cmax3 <- 10/9*temp_cmax2
w_clim(y_ini = c(N = 100, N = 100, N = 100),
    temp_cmin = c(temp_cmin1,temp_cmin2,temp_cmin3),
    temp_cmax = c(temp_cmax1,temp_cmax2,temp_cmax3),
    ro = rep(0.7,3),
    lambda = rep(0.00005,3),
    lat = rep(-33,3),
    lon = rep(-71,3),
    s = 5,
    res = 5,
    time_start = 2000,
    time_end = 2070,
    leap = 1/12)
```

```r
lat1 <- -10
lat2 <- -33
lat3 <- -42
```
w_clim(y_ini = c(N = 100, N = 100, N = 100),
    temp_cmin = rep(18, 3),
    temp_cmax = rep(40, 3),
    ro = rep(0.7, 3),
    lambda = rep(0.00005, 3),
    lat = c(lat1, lat2, lat3),
    lon = rep(-71, 3),
    s = 5,
    res = 5,
    time_start = 2000,
    time_end = 2070,
    leap = 1/12)

#########################################################################
# Example 4: Different marginal losses by a non-thermodependent
# component of intraspecific competition.
#########################################################################

lambda3 <- 0.01
lambda2 <- 1/2*lambda3
lambda1 <- 1/2*lambda2

w_clim(y_ini = c(N = 100, N = 100, N = 100),
    temp_cmin = rep(18, 3),
    temp_cmax = rep(30, 3),
    ro = rep(0.7, 3),
    lambda = c(lambda1, lambda2, lambda3),
    lat = rep(-33, 3),
    lon = rep(-71, 3),
    s = 5,
    res = 5,
    time_start = 2000,
    time_end = 2070,
    leap = 1/12)

#########################################################################
# Application example I: Bioclimatic variable
# (Annual Mean Temperature).
#########################################################################

# We consider a population of Macrolophus pygmaeus in three different
# locations, and its intrinsic growth rate is adjusted to data obtained
# from Rezende and Bozinovic (2019).

github_link <- "https://github.com/Victor-Saldana/epcc/raw/main/M_pygmaeus.xlsx"
library(httr)
temp_file <- tempfile(fileext = ".xlsx")
req <- GET(github_link,
    authenticate(Sys.getenv("GITHUB_PAT"), ""),
    write_disk(path = temp_file))
w_clim

M_pygmaeus <- readxl::read_excel(temp_file)

TPC <- rate_adjustment(data = M_pygmaeus)

#locality 1
lat1 <- 38.1827778
lon1 <- -1.7380555

#locality 2
lat2 <- 41.01384
lon2 <- 28.94966

#locality 3
lat3 <- 39.7213889
lon3 <- 21.63416638888889

w_clim(y_ini = c(N = 100, N = 100, N = 100),
        temp_cmin = rep(TPC$temp_cmin,3),
        temp_cmax = rep(TPC$temp_cmax,3),
        ro = rep(TPC$ro,3),
        lambda = rep(0.00005,3),
        lat = c(lat1,lat2,lat3),
        lon = c(lon1,lon2,lon3),
        s = 1,
        res = 5,
        time_start = 2000,
        time_end = 2070,
        leap = 1/12)

#######################################################################
#Application example II: Bioclimatic variable
# (Max Temperature of Warmest Month).
#######################################################################

#We consider a population of Macrolophus pygmaeus in three different
#locations, and its intrinsic growth rate is adjusted to data obtained
#from Rezende and Bozinovic (2019).

github_link <- "https://github.com/Victor-Saldana/epcc/raw/main/M_pygmaeus.xlsx"
library(httr)
temp_file <- tempfile(fileext = ".xlsx")
req <- GET(github_link,
          authenticate(Sys.getenv("GITHUB_PAT"), ""),
          write_disk(path = temp_file))
M_pygmaeus <- readxl::read_excel(temp_file)

TPC <- rate_adjustment(data = M_pygmaeus)

#locality 1
lat1 <- 38.1827778
lon1 <- -1.7380555
#locality 2
lat2 <- 41.01384
lon2 <- 28.94966

#locality 3
lat3 <- 39.7213889
lon3 <- 21.63416638888889

w_clim(y_ini = c(100, 100, 100),
temp_cmin = rep(TPC$temp_cmin,3),
temp_cmax = rep(TPC$temp_cmax,3),
ro = rep(TPC$ro,3),
lambda = rep(0.00005,3),
lat = c(lat1,lat2,lat3),
lon = c(lon1,lon2,lon3),
s = 5,
res = 5,
time_start = 2000,
time_end = 2070,
leap = 1/12)

#######################################################################
#Application example III: Bioclimatic variable
# (Mean Temperature of Warmest Quarter).
#######################################################################

#We consider a population of Macrolophus pygmaeus in three different
#locations, and its intrinsic growth rate is adjusted to data obtained
#from Rezende and Bozinovic (2019).

github_link <- "https://github.com/Victor-Saldana/epcc/raw/main/M_pygmaeus.xlsx"
library(httr)
temp_file <- tempfile(fileext = "xlsx")
req <- GET(github_link,
authenticate(Sys.getenv("GITHUB_PAT"),""),
write_disk(path = temp_file))
M_pygmaeus <- readxl::read_excel(temp_file)
TPC <- rate_adjustment(data = M_pygmaeus)

#locality 1
lat1 <- 38.1827778
lon1 <- -1.7380555

#locality 2
lat2 <- 41.01384
lon2 <- 28.94966

#locality 3
lat3 <- 39.7213889
lon3 <- 21.63416638888889
w_clim(y_ini = c(N = 100, N = 100, N = 100),
    temp_cmin = rep(TPC$temp_cmin, 3),
    temp_cmax = rep(TPC$temp_cmax, 3),
    ro = rep(TPC$ro, 3),
    lambda = rep(0.00005, 3),
    lat = c(lat1, lat2, lat3),
    lon = c(lon1, lon2, lon3),
    s = 10,
    res = 5,
    time_start = 2000,
    time_end = 2070,
    leap = 1/12)

## End(Not run)
## Index

- age_structure, 2
- competition, 5
- cooling_pulse1, 10
- cooling_pulse2, 14
- decreasing_linear, 18
- decreasing_periodicity, 20
- decreasing_stabilization, 25
- get_RCP2.6, 28
- get_RCP8.5, 29
- heating_pulse1, 30
- heating_pulse2, 34
- increasing_linear, 38
- increasing_periodicity, 40
- increasing_stabilization, 45
- IPCC_RCP2_6, 48
- IPCC_RCP8_5, 51
- predation, 54
- rate_adjustment, 58
- rate_TPC, 60
- trend_periodic, 61
- variability, 65
- w_clim, 69