Package 'CWERNIrevamped'

December 11, 2025

Type Package

Title Community Wide Evolutionary Rescue Neutral Implementation revamped

Version 0.1.0

Description The original CWERNI package was developed to simulate community wide evolutionary rescue in a neutral community. It accompanied the publication Van Eldijk, Bisschop & Etienne (2020), Uniting Community Ecology and Evolutionary Rescue Theory: Community-Wide Rescue Leads to a Rapid Loss of Rare Species, Frontiers in Ecology and Evolution, Volume 8 - 2020 (https://doi.org/10.3389/fevo.2020.552268). The goal of CWERNIrevamped is to re-make the the original package, making it compatible with R 4.5 and using best development pratices. This includes writing good documentation, proper use of git & github and adding some automated testing.

License GPL (>= 3)

Encoding UTF-8

LazyData true

Imports SADISA, EntropyEstimation

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

RoxygenNote 7.3.3

Contents

CWRsim — *Community Wide evolutionary Rescue simulation (CWR)*

Description

Community Wide evolutionary Rescue simulation (CWR)

Usage

```
CWRsim(tmax, b1, d1, b2, d2, m12, m21, k1, abun_original, interval)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Birthrate of the mutants
d1	Deathrate of the mutants
b2	Birthrate of the residents
d2	Death rate of the residents
m12	Mutation rate of mutants to residents
m21	Mutation rate of residents to mutants
k1	Carrying capacity
abun_original	A vector of initial abundances for each species such as those created by generate_spat_abund
interval	After how many evenst should the population state be saved to the output matrix

Value

A matrix denoting the abundances of all the species in the community over time

Examples

```
CWRsim(10,0.6 , 0.1, 0.05, 0.1, 0, 0.0005, 16000,
generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000)), 500)
```

CWRsimSPEC — *Community Wide evolutionary Rescue simulation (CWR), with certain precise timepoints saved*

Description

Community Wide evolutionary Rescue simulation (CWR), with certain precise timepoints saved

Usage

```
CWRsimSPEC(
  tmax,
  b1,
  d1,
  b2,
  d2,
  m12,
  m21,
  k1,
  abun_original,
  interval,
  wantedtimes
)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Birthrate of the mutants
d1	Deathrate of the mutants
b2	Birthrate of the residents
d2	Death rate of the residents
m12	Mutation rate of mutants to residents
m21	Mutation rate of residents to mutants
k1	Carrying capacity
abun_original	A vector of initial abundances for each species such as those created by generate_spat_abund
interval	After how many evenst should the population state be saved to the output matrix
wantedtimes	The preciese timepoints that should be saved

Value

A matrix denoting the abundances of all the species in the community over time, with certain precise timpoints saved

Examples

```
CWRsimSPEC(10,0.6 , 0.1, 0.05, 0.1, 0, 0.0005, 16000,
generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000)),
 200,c(15,30,50,75,100))
```

generate_spat_abund *Generate a neutral community*

Description

Uses sampling function for neutral community from Etienne et al. 2007 (Ecology Letters), used to initialize communities.

Usage

```
generate_spat_abund(theta, Ivec, Jvec)
```

Arguments

theta	an integer that gives a value to Theta
Ivec	a vector of I values
Jvec	a vector of J values

Value

a vector of species abundances

Examples

```
generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000))
```

loglikCWR *Perform CWR simulation, estimate paramters using SADISA, re-simulate neutral communities to generate distribution of likelihoods*

Description

Perform CWR simulation, estimate paramters using SADISA, re-simulate neutral communities to generate distribution of likelihoods

Usage

```
loglikCWR(
  tmax,
  b1,
  d1,
  k1,
  interval,
  b2,
  d2,
  m12,
  m21,
  orgsimmaxcount,
  totalresamp,
  tolsadisa = c(1e-06, 1e-06, 1e-06)
)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Birthrate of the mutants
d1	Deathrate of the mutants
k1	Carrying capacity
interval	After how many evenst should the population state be saved to the output matrix
b2	Birthrate of the residents
d2	Death rate of the residents
m12	Mutation rate of mutants to residents
m21	Mutation rate of residents to mutants
orgsimmaxcount	how many CWR simulations should be performed
totalresamp	how many neutral communities should be simulated for each single CWR simulation
tolsadisa	tolerances for SADISA estimation vector of 3 values, default c(1e-06, 1e-06, 1e-06)

Value

This function saves a .pdf plot for each CWR simulation performed, showing the distribution of likelihoods

Examples

```
loglikCWR(5, 0.6, 0.1, 16000,200, 0.05,0.1, 0, 0.0005, 1,1,c(1e-1, 1e-1, 1e-1))
```

---

multiCWRsim *Multiple simulations of the Community Wide Rescue model (CWR)*

Description

Multiple simulations of the Community Wide Rescue model (CWR)

Usage

```
multiCWRsim(
  tmax,
  b1,
  d1,
  b2,
  d2,
  m12,
  m21,
  k1,
  abun_original,
```

```
  nsim,
  filenomen,
  interval
)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Birthrate of the mutants
d1	Deathrate of the mutants
b2	Birthrate of the residents
d2	Death rate of the residents
m12	Mutation rate of mutants to residents
m21	Mutation rate of residents to mutants
k1	Carrying capacity
abun_original	A vector of initial abundances for each species such as those created by generate_spat_abund
nsim	The number of simulations to be performed
filenomen	The name of the plot that is saved
interval	After how many evenst should the population state be saved to the output matrix

Value

this function plots all the results of the simulations in a .tiff file in the current working directory

Examples

```
multiCWRsim(10,0.6 , 0.1, 0.05, 0.1, 0, 0.0005, 16000,
generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000)),
2, "CWR.tiff", 200)
```

---

multiCWRsimSPEC *Multiple simulations of the Community Wide Rescue model (CWR) saving specific timepoints and also saving the output matrix for every simulation.*

Description

Multiple simulations of the Community Wide Rescue model (CWR) saving specific timepoints and also saving the output matrix for every simulation.

Usage

```
multiCWRsimSPEC(
  tmax,
  b1,
  d1,
  b2,
  d2,
  m12,
  m21,
  k1,
  abun_original,
  interval,
  nsim,
  wantedtimes
)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Birthrate of the mutants
d1	Deathrate of the mutants
b2	Birthrate of the residents
d2	Death rate of the residents
m12	Mutation rate of mutants to residents
m21	Mutation rate of residents to mutants
k1	Carrying capacity
abun_original	A vector of initial abundances for each species such as those created by generate_spat_abund
interval	After how many evenst should the population state be saved to the output matrix
nsim	The number of simulations to be performed
wantedtimes	Specific timepoints for which the RAC is to be plotted

Value

this function plots the RAC of the simulations for the times specified in Wantedtimes as .tiff files in the current working directory also writes a .csv file outputting the poplation matrix for every simulation performed

this function plots all the results of the simulations in a .tiff file in the current working directory

Examples

```
multiCWRsimSPEC(10,0.6 , 0.1, 0.05, 0.1, 0, 0.0005, 16000,
generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000)),
200, 2, c(15,30,50,75,100))
```

`multinulsim` *Multiple Simple Neutral community (SN) simulations*

Description

Multiple Simple Neutral community (SN) simulations

Usage

```
multinulsim(tmax, b1, d1, k1, abun_original, nsim, filenomen, interval)
```

Arguments

`tmax`	Arbitrary units of time the simulation should be run
`b1`	Birthrate
`d1`	Deathrate
`k1`	Carrying capacity
`abun_original`	A vector of initial abundances for each species such as those created by generate_spat_abund
`nsim`	number of simulations that should be performed (integer)
`filenomen`	the name of the plot that is saved
`interval`	After how many evenst should the population state be saved to the output matrix

Value

this function plots all the results of the simulations in a .tiff file in the current working directory

Examples

```
multinulsim(10, 0.6, 0.1, 16000,
generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000)),
2,"nulsim.tiff",  200)
```

`multinulsimSPEC` *Multiple Simple Neutral community (SN) simulations, plotting the RAC at specific timepoints and also saving the output matrix for every simulation*

Description

Multiple Simple Neutral community (SN) simulations, plotting the RAC at specific timepoints and also saving the output matrix for every simulation

Usage

```
multinulsimSPEC(tmax, b1, d1, k1, abun_original, nsim, interval, wantedtimes)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Birthrate
d1	Deathrate
k1	Carrying capacity
abun_original	A vector of initial abundances for each species such as those created by generate_spat_abund
nsim	number of simulations that should be performed (integer)
interval	After how many evenst should the population state be saved to the output matrix
wantedtimes	Specific timepoints for which the RAC is to be plotted

Value

this function plots the RAC of the simulations for the times specified in Wantedtimes as .tiff files in the current working directory also writes a .csv file outputting the poplation matrix for every simulation performed

Examples

```
multinulsimSPEC(10, 0.6, 0.1, 1600,
generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000)),
2, 200, c(15,30,50,75,100))
```

---

multisimbvar *Simulate multiple neutral communities with a variable birthrate (VBN)*

Description

Simulate multiple neutral communities with a variable birthrate (VBN)

Usage

```
multisimbvar(
  tmax,
  b1,
  d1,
  k1,
  bneck,
  kneckstart,
  kneckend,
  abun_original,
  nsim,
  filenomen,
  interval
)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Normal birthrate
d1	Deathrate
k1	Carrying capacity
bneck	Birthrate during the bottleneck
kneckstart	Timepoint at which the bottleneck starts
kneckend	Timepoint at which the bottleneck ends
abun_original	A vector of initial abundances for each species such as those created by generate_spat_abund
nsim	number of simulations that should be performed
filenomen	the name of the plot that is saved
interval	After how many evenst should the population state be saved to the output matrix

Value

this function plots all the results of the simulations in a .tiff file in the current working directory

Examples

```
multisimbvar( 10, 0.6, 0.1, 16000, 0.05,0, 5,
generate_spat_abund (theta = 200,Ivec = rep(40,1),Jvec = c(16000)),
2, "bvar.tiff", 200)
```

---

multisimbvarSPEC — *Simulate multiple neutral communities with a variable birthrate (VBN) saving specific timepoints and also saving the output matrix for every simulation*

Description

Simulate multiple neutral communities with a variable birthrate (VBN) saving specific timepoints and also saving the output matrix for every simulation

Usage

```
multisimbvarSPEC(
  tmax,
  b1,
  d1,
  k1,
  bneck,
  kneckstart,
  kneckend,
  abun_original,
  nsim,
  interval,
  wantedtimes
)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Normal birthrate
d1	Deathrate
k1	Carrying capacity
bneck	Birthrate during the bottleneck
kneckstart	Timepoint at which the bottleneck starts
kneckend	Timepoint at which the bottleneck ends
abun_original	A vector of initial abundances for each species such as those created by generate_spat_abund
nsim	number of simulations that should be performed
interval	After how many evenst should the population state be saved to the output matrix
wantedtimes	Specific timepoints for which the RAC is to be plotted

Value

this function plots the RAC of the simulations for the times specified in Wantedtimes as .tiff files in the current working directory also writes a .csv file outputting the poplation matrix for every simulation performed

Examples

```
multisimbvarSPEC(10, 0.6, 0.1, 16000, 0.05, 0, 5,
generate_spat_abund (theta = 200,Ivec = rep(40,1),Jvec = c(16000)),2,
200, c(15,30,50,75,100))
```

---

nulsim *Simple Neutral community (SN)*

Description

Simple Neutral community (SN)

Usage

```
nulsim(tmax, b1, d1, k1, abun_original, interval)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Birthrate
d1	Deathrate
k1	Carrying capacity
abun_original	A vector of initial abundances for each species such as those created by generate_spat_abund
interval	After how many evenst should the population state be saved to the output matrix

Value

A matrix denoting the abundances of all the species in the community over time

Examples

```
nulsim(10, 0.6, 0.1, 16000, generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000)), 500)
```

`nulsimSPEC` *Simple Neutral community (SN) saving certain precise timepoints*

Description

Simple Neutral community (SN) saving certain precise timepoints

Usage

```
nulsimSPEC(tmax, b1, d1, k1, abun_original, interval, wantedtimes)
```

Arguments

`tmax`	Arbitrary units of time the simulation should be run
`b1`	Birthrate
`d1`	Deathrate
`k1`	Carrying capacity
`abun_original`	A vector of initial abundances for each species such as those created by generate_spat_abund
`interval`	After how many evenst should the population state be saved to the output matrix
`wantedtimes`	The specific timepoints that should be included in the final matrix

Value

A matrix denoting the abundances of all the species in the community over time, with certain precise timepoints saved (wantedtimes)

Examples

```
nulsimSPEC(10, 0.6, 0.1, 16000, generate_spat_abund(theta = 200,Ivec = rep(40,1),
Jvec = c(16000)), 20000000, c(1,3,5))
```

renCWR *Perform CWR simulation, Calculate Reny entropy (alfa=1, alfa=2), re-simulate neutral communities using SADISA parameters estimated on CWR simulation, simulation neutral communities, calculate their Reny entropy (alfa=1, alfa=2) to generate a distribution.*

Description

Perform CWR simulation, Calculate Reny entropy (alfa=1, alfa=2), re-simulate neutral communities using SADISA parameters estimated on CWR simulation, simulation neutral communities, calculate their Reny entropy (alfa=1, alfa=2) to generate a distribution.

Usage

```
renCWR(
  tmax,
  b1,
  d1,
  k1,
  interval,
  b2,
  d2,
  m12,
  m21,
  orgsimmaxcount,
  totalresamp
)
```

Arguments

tmax	Arbitrary units of time the simulation should be run
b1	Birthrate of the mutants
d1	Deathrate of the mutants
k1	Carrying capacity
interval	After how many evenst should the population state be saved to the output matrix
b2	Birthrate of the residents
d2	Death rate of the residents
m12	Mutation rate of mutants to residents
m21	Mutation rate of residents to mutants
orgsimmaxcount	how many CWR simulations should be performed
totalresamp	how many neutral communities should be simulated for each single CWR simulation

Value

This function saves two .pdf plots for each CWR simulation performed, showing the Reny entropy of each CWR simulation and the distribution of Reny entorpy generated from the neutral re-simulations

Examples

```
renCWR(10, 0.6, 0.1, 16000,200, 0.05,0.1, 0, 0.0005, 1,1)
```

`simbvar` *Simulate a neutral community with a variable birthrate (VBN)*

Description

Simulate a neutral community with a variable birthrate (VBN)

Usage

```
simbvar(tmax, b1, d1, k1, bneck, kneckstart, kneckend, abun_original, interval)
```

Arguments

`tmax`	Arbitrary units of time the simulation should be run
`b1`	Normal birthrate
`d1`	Deathrate
`k1`	Carrying capacity
`bneck`	Birthrate during the bottleneck
`kneckstart`	Timepoint at which the bottleneck starts
`kneckend`	Timepoint at which the bottleneck ends
`abun_original`	A vector of initial abundances for each species such as those created by generate_spat_abund
`interval`	After how many evenst should the population state be saved to the output matrix

Value

A matrix denoting the abundances of all the species in the community over time

Examples

```
simbvar( 10, 0.6, 0.1, 16000, 0.05,0, 5,
generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000)), 200)
```

`simbvarSPEC` *Simulate a neutral community with a variable birthrate (VBN), saving certain precise timepoints*

Description

Simulate a neutral community with a variable birthrate (VBN), saving certain precise timepoints

Usage

```
simbvarSPEC(
  tmax,
  b1,
  d1,
  k1,
  bneck,
  kneckstart,
  kneckend,
  abun_original,
  interval,
  wantedtimes
)
```

Arguments

`tmax`	Arbitrary units of time the simulation should be run
`b1`	Normal birthrate
`d1`	Deathrate
`k1`	Carrying capacity
`bneck`	Birthrate during the bottleneck
`kneckstart`	Timepoint at which the bottleneck starts
`kneckend`	Timepoint at which the bottleneck ends
`abun_original`	A vector of initial abundances for each species such as those created by generate_spat_abund
`interval`	After how many evenst should the population state be saved to the output matrix
`wantedtimes`	The specific timepoints that should be included in the final matrix

Value

A matrix denoting the abundances of all the species in the community over time, with certain specific timepoints saved (wantedtimes)

Examples

```
simbvarSPEC( 10, 0.6, 0.1, 16000, 0.05,0, 5,
generate_spat_abund(theta = 200,Ivec = rep(40,1),Jvec = c(16000)),
200,c(15,30,50,75,100))
```

Index