Package 'sdcn'

February 23, 2015

Type Package

Title Structures and Dynamics on (of) Complex Networks (sdcn)

Version 0.0.0.9000

Date 2015-02-10

Author Wenfeng Feng, Richard Bailey, Kirsty McGregor

Maintainer Wenfeng Feng <fengwenfeng@gmail.com>

Description The package intends to implement general simulation of dynamics on (of) networks which have different structural features. The current goal is to simuate ecological interactions among species in ecological networks, as the first instance of complex networks.
Modules should include:
1) Dynamic models. Holling Type I, II dynamic models should be implemented for mutualistic networks, food webs, competitive networks, and mixed networks.
2) Environmental Perturbations. Two types of perturbations: continuously pressed env. and repeated pulsed env. (stochastics). The perturbations can effect not only on (all or part of) species(nodes) but also on (all or part of) interactions(links).
3) Null models of different structural features such as degree heterogeneity and modularity.
4) Analysis of simulation results.
5) Fit of empirical data?

Imports deSolve (>= 1.10-8),
simecol (>= 0.8-4),
rootSolve (>= 1.6.5)

License What license is it under?

Suggests knitr,
testthat

VignetteBuilder knitr

LazyData true

R topics documented:

`inc_to_adj` *transfer an incidence matrix to an adjacency matrix*

Description

transfer an incidence matrix to an adjacency matrix

Usage

```
inc_to_adj(inc)
```

Arguments

`inc,`	an incidence matrix

Value

adj, an adiacency matrix

`init_lv2` *initial values of state variables, i.e., abundances of species*

Description

Assign initial values according to two criteria: 1. using the equilibrium values of LV1 model as initial values. 2. If any of the initial values is less than 0, using the intrinsic growth rates as initial values.

Usage

```
init_lv2(parms)
```

Arguments

`parms,`	the parameters assigned to LV2 model

model_lv2 — *Lotka-Volterra (LV) Equations of Holling type II for mutualistic communities provided by Bastolla et al.*

Description

Lotka-Volterra (LV) Equations of Holling type II for mutualistic communities provided by Bastolla et al.

Usage

```
model_lv2(time, init, parms, ...)
```

Arguments

time,	time step of simulation
init,	the initial state of the LV system, a vector
parms,	parameters passed to LV model, a list of:
	r a vector of the intrinsic growth rates of species
	C the competitive matrix inside plants and animals
	M the mutualistic matrix between plants and animals
	h the saturate coefficient, handling time of species feed

Value

the derivation

M_PL_001 — *web-of-life data sets from Bascompte et atl.*

Description

web-of-life data sets from Bascompte et atl.

Usage

```
M_PL_001
```

Format

data frames with rows and cols represent two different species groups

Source

http://www.web-of-life.es/map.php

parms_lv2 *parmaters for mutualistic LV2 model according to the network and the coefficients*

Description

parmaters for mutualistic LV2 model according to the network and the coefficients

Usage

```
parms_lv2(graph, coeff)
```

Arguments

graph,	the interaction topology of mutualistic communities, which is the incidence matrix of a bipartite network
coeff,	a list of coefficients:

alpha.mu, alpha.sd coefficients of the intrinsic growth rates of species

beta0.mu, beta0.sd the intra-species competition coefficients which determin a uniform distribution in [beta0.mu - beta0.sd, beta0.mu + beta0.sd]

beta1.mu, beta1.sd the inter-species competition coefficients

gamma.mu, gamma.sd the inter-species mutualism coefficients

delta trade-off coefficients of mutualistic interaction strengths

h.mu, h.sd coefficients of the handling time of species

Value

a list of parameters for ode model:

r a vector of the intrinsic growth rates of species

C the competitive matrix inside plants and animals

M the mutualistic matrix between plants and animals

h the saturate coefficient, handling time of species feed

perturb_growthrate *perturbation that effect on species by increasing/decreasing the intrinsic growth rates of species*

Description

perturbation that effect on species by increasing/decreasing the intrinsic growth rates of species

Usage

```
perturb_growthrate(parms, nstar, r.delta = 0.01)
```

Arguments

parms	parameters assigned to the ODE model
nstar	state values at equilibrium
r.delta	difference of intrinsic growth rates at each iterating step

---

sdcn *sdcn: Structures and Dynamics on (of) Complex Networks.*

Description

The sdcn package provides three categories of functions:

1. Structures
2. Dynamics
3. Analysis

Structures functions

swaplinks

Dynamics functions

model_lv2, parms_lv2

---

sim_ode_auto *Simulate ODE dynamics of autonomous systems.The dynamic starts at initialized state variables, and ends in equilibrium (or error where some values of state variables approach infinity?)*

Description

Simulate ODE dynamics of autonomous systems.The dynamic starts at initialized state variables, and ends in equilibrium (or error where some values of state variables approach infinity?)

Usage

```
sim_ode_auto(model, parms, init, steps = 1000, stepwise = 1,
  extinct_threshold)
```

Arguments

model	model of ODE dynamics
parms	parameters assigned to the model
init	initial values of the model according to the parameters
steps	steps of simulation
stepwise	step length
extinct_threshold	abundance threshold, species with abundance less than that is considered to be exintct

Value

a list of:

`sim_ode_press` *Simulate ODE dynamics of non-autonomous systems.A example is ecosystems under "press" perturbations. The dynamic is iteration of successive ODE dynamics of automous sytems (`sim_ode_auto`), while at each iterating step, the parameters and/or state values of systems are changed to reflect "press" perturbations.*

Description

Simulate ODE dynamics of non-autonomous systems.A example is ecosystems under "press" perturbations. The dynamic is iteration of successive ODE dynamics of automous sytems (`sim_ode_auto`), while at each iterating step, the parameters and/or state values of systems are changed to reflect "press" perturbations.

Usage

```
sim_ode_press(model, parms, init, steps = 1000, stepwise = 1,
  extinct_threshold, perturb, iter_steps = 10, isout = TRUE, ...)
```

Arguments

`model`	model of ODE dynamics
`parms`	parameters assigned to the model
`init`	initial values of the model according to the parameters
`steps`	steps of simulation
`stepwise`	step length
`extinct_threshold`	abundance threshold, species with abundance less than that is considered to be exintct
`perturb`	a function that change the parameters and state values after each iteration step
`iter_steps`	iteration steps
`isout`	if output the transiting trajectory of each ODE iterate step
`...`	any arguments which are transfered to perturbation function

swaplinks *Swapping links Algorithm for null model of bipartite networks, that generates random network (ensembles) which keep the node degree distribution of a real network.*

Description

Swapping links Algorithm for null model of bipartite networks, that generates random network (ensembles) which keep the node degree distribution of a real network.

Usage

```
swaplinks(bigraph, ntry = 5000)
```

Arguments

bigraph,	incidence matrix of a bipartite network, rows and cols represent two groups of nodes/species
ntry,	the maximum possible times of swapping links to try

Value

an incidence matrix of bipartite network whose links being randomly swapped.

Examples

```
## Not run:
require(bipartite) # for plot
data(M_PL_003)
# M_PL_003 <- as.matrix(M_PL_003)
bipartite::visweb(M_PL_003)
M_PL_003.rand = swaplinks(M_PL_003)
bipartite::visweb(M_PL_003.rand)

## End(Not run)
```

Index