

R documentation

of all in ‘PredictImportance/man/’

November 18, 2015

R topics documented:

braycurtis	2
BuildCascade	2
BuildMPN	3
BuildNiche	4
caricaie	4
discreteLV	5
flensburg	5
GetCentralities	6
getTL	6
loadEmpiricalScript	7
loadModelScript	7
LognormalParam	8
normalizeM	8
OneRun	9
otago	9
predictimportance	10
reef	10
runScripts	11
serengeti	12
Step1_Empirical_Parameterization	12
Step1_Generate_Networks	13
Step2_Discrete_LV	14
Step3_Hierarchical_Model	15
stmarks	15
StoufferZTransform	16
sylt	16
tatoosh	17
ythan	17
Index	18

braycurtis	<i>Calculate Bray-Curtis similarity and Jaccard distance between two ecological communities</i>
------------	---

Description

Calculate Bray-Curtis similarity and Jaccard distance between two real-valued vectors of abundances. Bray-Curtis similarity is calculated as $\text{sum}(|x-y|/|x+y|)$, where x and y are real-valued vectors. and Jaccard similarity is calculated as $2BC/(1+BC)$, where BC is the Bray-Curtis similarity.

Usage

```
braycurtis(x, y)
```

Arguments

x	A vector of species abundances.
y	A second vector of abundances, with the same length as x.

Value

The similarity or distance between the two abundance vectors.

BuildCascade	<i>Build a Cascade Model Network</i>
--------------	--------------------------------------

Description

Generates a random network of a given size and approximate connectance according to the cascade model

Usage

```
BuildCascade(S = 50, C = 0.1)
```

Arguments

S	Number of species in network; defaults to 50
C	Network connectance, defined as $2L/(S(S-1))$, where L is the number of links. Defaults to .1. The resulting matrix will have a connectance of $C \pm 2\sqrt{.5C(1-C)S(S-1)}$.

Value

A weakly connected, unsigned adjacency matrix of size (S,S).

References

Cohen, J.E. and C.M. Newman. 1985. A Stochastic Theory of Community Food Webs: I. Models and Aggregated Data. Proc. R. Soc. Lond. B. 224: 421-448; DOI: 10.1098/rspb.1985.0042.

Examples

```
BuildCascade()
```

BuildMPN

Build a Minimum Potential Niche Model Network

Description

Generates a random network of a given size and approximate connectance according to the minimum potential niche model.

Usage

```
BuildMPN(S = 50, C = 0.1, GapProb = 0.25)
```

Arguments

S	Number of species in network; defaults to 50
C	Network connectance, defined as $2L/(S(S-1))$, where L is the number of links. Defaults to .1. The resulting matrix will have a connectance of $C \pm \sqrt{.5C(1-C)S(S-1)}$.
GapProb	Probability of a gap in a feeding niche; defaults to .25.

Value

M A weakly connected, unsigned adjacency matrix of size (S,S).

References

Allesina, S., D. Alonso, and M. Pascual. 2008. A General Model for Food Web Structure. Science 320(5876): 658-661.

Examples

```
BuildMPN()
```

BuildNiche

Build a Niche Model Network

Description

Generates a random network of a given size and approximate connectance according to the niche model

Usage

```
BuildNiche(S = 50, C = 0.1)
```

Arguments

S	Number of species in network; defaults to 50
C	Network connectance, defined as $2L/(S(S-1))$, where L is the number of links. Defaults to .1. The resulting matrix will have a connectance of $C \pm 2\sqrt{.5C(1-C)S(S-1)}$.

Value

M A weakly connected, unsigned adjacency matrix of size (S,S).

References

Williams, R.J., and N.D. Martinez. 2000. Simple rules yield complex food webs. *Nature* 404(6774): 180-3.

Examples

```
BuildNiche()
```

caricaie

The food web of the marsh at Grande Caricaie at Lake Neuchâtel, Switzerland.

Description

The food web of the marsh at Grande Caricaie at Lake Neuchâtel, Switzerland.

Format

An adjacency matrix with 163 rows and 163 columns.

References

Cattin Blandenier, M.-F. Food web ecology: models and application to conservation. PhD thesis, (2004).

discreteLV	<i>Simulate a discrete-time Lotka-Volterra time series</i>
------------	--

Description

Simulate discrete-time Lotka-Volterra population dynamics with perturbed growth rates

Usage

```
discreteLV(rmat, alphas, n0s, deltat, simtime)
```

Arguments

rmat	a matrix with S (number of species) rows and simtime/deltat cols containing the growth rates at each time step (potentially perturbed at each time step)
alphas	an S by S matrix of per capita interaction strengths
n0s	a vector of length S containing species abundances
deltat	the size of the time step
simtime	the total number of time steps to simulate

Value

an S by simtime matrix of species abundances at each timestep

flensburg	<i>The food web of Flensburg Fjord in the Baltic Sea, between Germany and Denmark.</i>
-----------	--

Description

The food web of Flensburg Fjord in the Baltic Sea, between Germany and Denmark.

Format

An adjacency matrix with 180 rows and 180 columns.

References

Zander, C. D., Josten, N., Detloff, K. C., Poulin, R., McLaughlin, J. P., and Thieltges, D. W. Food web including metazoan parasites for a brackish shallow water ecosystem in Germany and Denmark: Ecological Archives E092-174. Ecology 92(10), 2007-2007 (2011).

GetCentralities	<i>Calculate network metrics for a food web</i>
-----------------	---

Description

Read in a matrix and calculate log closeness centrality, log eigenvector centrality, and trophic level for each species. Note that this function is used internally and is not likely to be useful on its own. Use igraph functionality and getTL() for centrality calculations beyond replicating simulations from Wootton et al.

Usage

```
GetCentralities(matfile)
```

Arguments

matfile	path to parameterized network
---------	-------------------------------

Value

a data frame containing log(closeness centrality), log(eigenvector centrality), and trophic level for each species in the network.

getTL	<i>Calculate prey-averaged trophic levels</i>
-------	---

Description

Calculate prey-averaged trophic levels for each species in a matrix

Usage

```
getTL(M)
```

Arguments

M	An S by S adjacency matrix, where S is the number of species
---	--

Value

A vector of S trophic levels

References

Williams, R.J. and N.D. Martinez. 2004. Limits to Trophic Levels and Omnivory in Complex Food Webs: Theory and Data. Am. Nat.

Examples

```
mat <- BuildCascade()
getTL(mat)
```

loadEmpiricalScript	<i>Parameterize empirical networks, run models, and plot results as in Wootton et al.</i>
---------------------	---

Description

Parameterizes empirical networks 30 times, simulates population dynamics, runs hierarchical model, and generates violin plots for the following food webs: caricaie, flensburg, otago, reef, serengeti, stmarks, sylt, tatoosh, and ythan. This replicates main text results in Wootton et al. (in prep). Note that this simulation and analysis will likely days to complete!

Usage

```
loadEmpiricalScript()
```

loadModelScript	<i>Generate simulated networks, run models, and plot results as in Wootton et al SI</i>
-----------------	---

Description

Generates 30 simulated networks, parameterizes each network 30 times, simulates population dynamics, runs hierarchical model, and generates violin plots for the following food web models: Cascade, Niche, and Minimum Potential Niche (with gap probabilities of 25, 35, and 45 in Wootton et al. (in prep). Note that this simulation and analysis will likely days to complete!

Usage

```
loadModelScript()
```

LognormalParam	<i>Parameterize a Lotka-Volterra system given an adjacency matrix</i>
----------------	---

Description

Parameterize an adjacency matrix with interaction strengths and equilibrium abundances. Interaction strengths and abundances are drawn from a lognormal distribution truncated 5 standard deviations above the mean.

Usage

```
LognormalParam(Adj)
```

Arguments

Adj	An adjacency matrix
-----	---------------------

Value

a list with elements Mat (a matrix of interaction strengths) and Pop (a vector of equilibrium population sizes).

Examples

```
mat <- BuildCascade()
LognormalParam(mat)
```

normalizeM	<i>Normalize matrix columns</i>
------------	---------------------------------

Description

Normalize the columns of the adjacency matrix M by its column sums. This is a utility function for predictimportance which will likely not be useful except internally.

Usage

```
normalizeM(M)
```

Arguments

M	a matrix to be normalized
---	---------------------------

Value

the matrix M with columns normalized by column sum

OneRun	<i>Simulate Lotka-Volterra population dynamics and get summary stats</i>
--------	--

Description

Read in a matrix and population file, simulate discrete-time population dynamics with perturbed growth rates, and calculate species importance, variability, and degree. Note that this function is used internally, and many parameters (simtime, perturbation size) are hardcoded. Use discreteLV() for your own simulations or Step2_Discrete_LV() to replicate simulations from Wootton et al.

Usage

OneRun(matname, popname)

Arguments

- | | |
|---------|--|
| matname | path to parameterized network to be simulated |
| popname | path to vector of equilibrium abundances for the network |

Value

a data frame containing Jaccard distance, perturbation importance, mean abundance, standard deviation of abundance over the simulation, and network degree for each species in the network.

otago	<i>The food web of the intertidal mudflat in Otago Harbour, New Zealand.</i>
-------	--

Description

The food web of the intertidal mudflat in Otago Harbour, New Zealand.

Format

An adjacency matrix with 180 rows and 180 columns.

References

Mouritsen, K. N., Poulin, R., McLaughlin, J. P., and Thieltges, D. W. Food web including metazoan parasites for an intertidal ecosystem in New Zealand: Ecological Archives E092-713. Ecology 92(10), 2006-2006 (2011).

predictimportance	<i>predictimportance: a package containing code and data necessary to replicate results in Wootton et al. (in prep.).</i>
-------------------	---

Description

The predictimportance package contains functions to randomly parameterize empirical networks, generate and parameterize model food webs, simulate community dynamics, and predict species importance using several covariates using a hierarchical modelling approach. The highest-level and most user-friendly functions are runScriptsEmpirical() and runScriptsModel(), which can be used to replicate all or part of the data simulation and analysis in a single function call. For users who wish to replicate all results, the simplest approach is to run the scripts in inst/scripts, FullSimulationEmpirical.R (for main text results) and FullSimulationModels.R (for supplemental results). These scripts run perform all data simulation and analysis, and plot violin plots to summarize the results. For users who wish to analyze their own empirical dataset, it is recommended to use the Step1_Empirical_Parameterization(), Step2_Discrete_LV(), and Step3_Hierarchical_Model() functions in order. Note that it is worthwhile to examine output from Step 2 before determining if the specific hierarchical model in Step 3 is appropriate for your data. Note that Step 2 (simulation of community dynamics) may produce non-finite values for large networks, due to precision issues.

References

Wootton, J.T., Sander, E.L., Barab\'as, Gy\'orgy, and A. Henry. In prep.

reef	<i>The food web for the Caribbean reef on the American Virgin Islands.</i>
------	--

Description

The food web for the Caribbean reef on the American Virgin Islands.

Format

An adjacency matrix with 249 rows and 249 columns.

References

Opitz, S. Trophic interactions in Caribbean coral reefs. Number 1085. WorldFish, (1996).

runScripts	<i>Run all analyses for a food web model or network Run all data simulation and analysis steps for a given model or empirical web(s). This may take a while!</i>
------------	--

Description

Run all analyses for a food web model or network

Run all data simulation and analysis steps for a given model or empirical web(s). This may take a while!

Usage

```
runScriptsModel(model = "Cascade", path = "./")
```

```
runScriptsEmpirical(web = "All", path = "./")
```

Arguments

model	food web model used to construct the network: options are 'Cascade', 'Niche', 'MPN25', 'MPN35', and 'MPN45'.
path	path to where Data and Results folders are or should be created
web	empirical network to be parameterized and simulated: options are "caricaie", "otago", "serengeti", "sylv", "ythan", "flensburg", "reef", "stmarks", and "tatoosh". A vector of network names can be provided to simulate more than one network, or web = "All" can be used to simulate all networks in the package.

Examples

```
## Not run:
runScriptsModel('Niche')

## End(Not run)
## Not run:
runScriptsEmpirical('tatoosh')

## End(Not run)
```

serengeti

The terrestrial food web of the Serengeti grassland in Tanzania.

Description

The terrestrial food web of the Serengeti grassland in Tanzania.

Format

An adjacency matrix with 170 rows and 170 columns.

References

Baskerville, E. B., Dobson, A. P., Bedford, T., Allesina, S., Anderson, T. M., and Pascual, M. Spatial guilds in the Serengeti food web revealed by a Bayesian group model. PLoS Computational Biology 7(12), e1002321 (2011).

Step1_Empirical_Parameterization

Step 1: Generate and/or parameterize Lotka-Volterra communities

Description

Generates parameterized networks and abundance data that can be simulated in Step 2.

Usage

```
Step1_Empirical_Parameterization(web, foldname = web, path = "Data",
                                nruns = 30, Exponential = FALSE, seed = NULL)
```

Arguments

web	empirical network name ("caricaie", "otago", "serengeti", "sylv", "ythan", "flensburg", "reef", "stmarks", or "tatoosh")
foldname	Folder name where matrices and abundance vectors should be written
path	Path where data are kept. 'foldname' will be created as a folder here if it does not already exist
nruns	Number of parameterizations to be generated.
Exponential	TEMPORARY
seed	Random seed for reproducibility

Value

This function does not return an object, but it writes parameterized matrices (with file names ending in '-mat.txt') and vectors of equilibrium abundances (with file names ending in '-pop.txt') to the folder path/foldname.

Step1_Generate_Networks

Step 1: Generate and/or parameterize Lotka-Volterra communities

Description

Generates parameterized networks and abundance data that can be simulated in Step 2.

Generates parameterized networks and abundance data that can be simulated in Step 2.

Usage

```
Step1_Generate_Networks(web = "Cascade", foldname = web, path = "Data",
  S = 50, C = 0.1, GapProb = 0.25, nwebs = 30, nruns = 30,
  seed = NULL)
```

```
Step1_Generate_Networks(web = "Cascade", foldname = web, path = "Data",
  S = 50, C = 0.1, GapProb = 0.25, nwebs = 30, nruns = 30,
  seed = NULL)
```

Arguments

web	Model name ('Cascade', 'Niche', or 'MPN') or empirical network name ("caricaie", "otago", "serengeti", "sylv", "ythan", "flensburg", "reef", "stmarks", or "tatoosh")
foldname	Folder name where matrices and abundance vectors should be written
path	Path where data are kept. 'foldname' will be created as a folder here if it does not already exist
S	Number of species in the generated networks
C	Network connectance, defined as $2L/(S(S-1))$, where L is the number of links.
GapProb	Probability of a gap in a niche. Only used if web == 'MPN'.
nwebs	Number of random web structures to be generated.
nruns	Number of parameterizations to be generated.
seed	Random seed for reproducibility
web	Model name ('Cascade', 'Niche', or 'MPN') or empirical network name ("caricaie", "otago", "serengeti", "sylv", "ythan", "flensburg", "reef", "stmarks", or "tatoosh")
foldname	Folder name where matrices and abundance vectors should be written
path	Path where data are kept. 'foldname' will be created as a folder here if it does not already exist
S	Number of species in the generated networks
C	Network connectance, defined as $2L/(S(S-1))$, where L is the number of links.
GapProb	Probability of a gap in a niche. Only used if web == 'MPN'.

nwebs	Number of random web structures to be generated.
nruns	Number of parameterizations to be generated.
seed	Random seed for reproducibility

Value

This function does not return an object, but it writes parameterized matrices (with file names ending in '-mat.txt') and vectors of equilibrium abundances (with file names ending in '-pop.txt') to the folder path/foldname.

This function does not return an object, but it writes parameterized matrices (with file names ending in '-mat.txt') and vectors of equilibrium abundances (with file names ending in '-pop.txt') to the folder path/foldname.

Step2_Discrete_LV	<i>Step 2: Simulate dynamics using discrete-time Lotka-Volterra</i>
-------------------	---

Description

Takes network parameterizations created in Step 1 and simulates community dynamics using a discrete-time Lotka-Volterra. Transforms variables for use in hierarchical model (Step 3).

Usage

```
Step2_Discrete_LV(path, seed = NULL)
```

Arguments

path	path to folder where '-mat.txt' and '-pop.txt' files were generated in Step 1.
seed	random seed, for reproducibility

Value

Returns the csv file name where results are stored. This file name can then be passed to Step 3. This results file is appended to file as the simulations are completed.

Step3_Hierarchical_Model

Step 3: Analyze data using a hierarchical model

Description

Takes data file created in Step 2 and analyzes the data using a hierarchical model

Usage

```
Step3_Hierarchical_Model(datafname, PathToResults = "Results",
  empirical = TRUE)
```

Arguments

datafname	path to file generated in Step 2 (file name will end in '-AllData.csv')
PathToResults	path to location where results should be stored.
empirical	flags if the simulated data are based on an empirical network structure. This will affect the hierarchical model structure

Value

This function does not return an object, but it writes hierarchical model output to text files, and saves the entire lmer object to a .RData file

stmarks	<i>The food web for the seagrass community at Goose Creek By in St. Marks National Wildlife Refuge, Florida.</i>
---------	--

Description

The food web for the seagrass community at Goose Creek By in St. Marks National Wildlife Refuge, Florida.

Format

An adjacency matrix with 143 rows and 143 columns.

References

Christian, R. R. and Luczkovich, J. J. Organizing and understanding a winter's sea-grass foodweb network through effective trophic levels. *Ecological Modelling* 117(1), 99-124 (1999).

StoufferZTransform	<i>Calculates a combined p-value using the weighted Z-method</i>
--------------------	--

Description

Combines a vector of one-sided p-values from independent tests into a single combined two-sided p-value

Usage

```
StoufferZTransform(ps)
```

Arguments

ps	A vector of independent one-sided p-values to be combined. Note that these p-values must be one-sided to account for the fact that different tests could suggest a difference from the null in both directions, which should result in less overall evidence against the null when combined.
----	--

Value

A two-sided combined p-value.

References

Whitlock, M.C. 2005. Combining probability from independent tests: the weighted Z-method is superior to Fisher's approach. J. evol. biol 18(5): 1368-1373.

Examples

```
ps <- c(.05, .007, .4, .2)
StoufferZTransform(ps)
```

sylv	<i>The food web for the North Sea Sylv tidal basin between Germany and Denmark.</i>
------	---

Description

The food web for the North Sea Sylv tidal basin between Germany and Denmark.

Format

An adjacency matrix with 230 rows and 230 columns.

References

Thieltges, D. W., Reise, K., Mouritsen, K. N., McLaughlin, J. P., and Poulin, R. Food web including metazoan parasites for a tidal basin in Germany and Denmark: Ecological Archives E092-172. Ecology 92(10), 2005-2005 (2011).

tatoosh	<i>Food web for the mussel bed at the Tatoosh intertidal in Tatoosh Island, Washington.</i>
---------	---

Description

Food web for the mussel bed at the Tatoosh intertidal in Tatoosh Island, Washington.

Format

An adjacency matrix with 110 rows and 110 columns.

Source

<http://datadryad.org/review?doi=doi:10.5061/dryad.39jv1>

ythan	<i>The food web for Ythan Estuary in Scotland.</i>
-------	--

Description

The food web for Ythan Estuary in Scotland.

Format

An adjacency matrix with 92 rows and 92 columns.

References

Cohen, J. E., Schittler, D. N., Raffaelli, D. G., and Reuman, D. C. Food webs are more than the sum of their tritrophic parts. Proceedings of the National Academy of Sciences 106(52), 22335-22340 (2009).

Index

braycurtis, [2](#)
BuildCascade, [2](#)
BuildMPN, [3](#)
BuildNiche, [4](#)

caricaie, [4](#)

discreteLV, [5](#)

flensburg, [5](#)

GetCentralities, [6](#)
getTL, [6](#)

loadEmpiricalScript, [7](#)
loadModelScript, [7](#)
LognormalParam, [8](#)

normalizeM, [8](#)

OneRun, [9](#)
otago, [9](#)

predictimportance, [10](#)
predictimportance-package
 (predictimportance), [10](#)

reef, [10](#)
runScripts, [11](#)
runScriptsEmpirical(runScripts), [11](#)
runScriptsModel(runScripts), [11](#)

serengeti, [12](#)
Step1_Empirical_Parameterization, [12](#)
Step1_Generate_Networks, [13](#)
Step2_Discrete_LV, [14](#)
Step3_Hierarchical_Model, [15](#)
stmarks, [15](#)
StoufferZTransform, [16](#)
sylv, [16](#)

tatoosh, [17](#)

ythan, [17](#)