

Package ‘nos’

May 31, 2017

Type Package

Title Compute Node Overlap and Segregation in Ecological Networks

Version 1.0.0

Author Thomas J. Matthews and Giovanni Strona

Maintainer Thomas J. Matthews <txm676@gmail.com>

Description Calculate NOS (node overlap and segregation) and the associated metrics described in Strona and Veech (2015) <DOI:10.1111/2041-210X.12395> and Strona et al. (2017; In Press, DOI to be provided in subsequent package version). The functions provided in the package enable assessment of structural patterns ranging from complete node segregation to perfect nestedness in a variety of network types. In addition, they provide a measure of network modularity.

License GPL-3

URL <https://github.com/txm676/nos>

BugReports <https://github.com/txm676/nos/issues>

Imports dplyr, gmp, stats

Depends R(>= 3.3.0)

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2017-05-31 19:28:24 UTC

R topics documented:

nos-package	2
boreal	3
NOSM_bip	4

NOSM_dir	5
NOSM_POT_dir	6
NOSM_POT_undir	7
NOSM_undir	8
summary.NOSM	9

Index	11
--------------	-----------

nos-package	<i>nos: A package for calculating node overlap and segregation in ecological networks</i>
-------------	---

Description

A package to calculate NOS (node overlap and segregation) and the associated metrics described in Strona and Veech (2015) and Strona et al. (2017). The functions provided in the package enable assessment of structural patterns ranging from complete node segregation to perfect nestedness in a variety of network types. In addition, they provide a measure of network modularity.

Does not currently work on Macs (OS X El Capitan) as the dependent gmp package is not yet available in binary form for OS X El Capitan.

Details

There are five main S3 generics that the user can choose depending on whether a directed, bipartite or undirected network is provided as input, and whether a network of potential interactions (`pot_net`) is available from the user. These five functions are: 1) `NOSM_dir` - for directed unimode networks (e.g. food webs) when `pot_net` is not provided, 2) `NOSM_undir` - for undirected networks (e.g. co-occurrence networks) when `pot_net` is not provided, 3) `NOSM_bip` - for bipartite networks (e.g. plant-pollinator networks), 4) `NOSM_pot_dir` - for directed unimode (e.g. food webs) and bipartite networks when `pot_net` is provided, and 5) `NOSM_pot_undir` - for undirected networks (e.g. co-occurrence networks) when `pot_net` is provided.

Each of these five main functions produces an output with class 'NOSM'. An S3 method (`summary.NOSM`) provides the summary statistics of interest (e.g.. NOS, Mod/network modularity, effect size Z, and p-value).

If a network of potential interactions is not provided, the computation will be made according to the following criteria: in an undirected network, all nodes will be considered as potential interacting partners; in a directed, unimode network, all nodes having at least an in-coming link will be considered as potential partners for nodes having at least an out-going link and vice-versa; in a bipartite network, where nodes can be formally categorized into two distinct categories (e.g. plant-pollinators) all nodes in one category will be considered as potential partners for the nodes in the other category (and vice-versa).

The internal function `comb` calculates factorials. If the user provided `n` or `k` are too large (> roughly 170) `factorial(n || k)` produces Inf. In these cases `gmp::factorialZ` is used instead. See the help documentation for the 'gmp' package for further information.

It should be noted that due to differences in the way R and Python calculate the standard deviation, the modularity values and z values calculated using this R package differ very slightly (less than 0.2) from the values calculated using the Python scrip from Strona et al. (2017).

Author(s)

Thomas J. Matthews and Giovanni Strona

References

Strona, G., Matthews, T.J., Kortsch, S. and Veech, J.A. (2017) NOS: A software suite to compute node overlap and segregation in ecological networks. *Ecography*. In review.

boreal

A sample food-web network from the boreal region of the Barents Sea

Description

The sample network is a marine food web from the boreal region of the Barents Sea. The Barents Sea is a large, open sub-arctic shelf sea bordering the Arctic Ocean. The sample food web contains 180 trophospecies, consisting of detritus, 8 basal taxa, 32 zooplankton, 66 benthic, 56 fish, 8 sea birds and 9 marine mammal, and 1546 feeding interactions.

Usage

```
data(boreal)
```

Format

A data frame with 2 columns and 1546 rows. Each row is a node pair where the first element of each node pair is 'consumed' (or pollinated etc) by the second element.

Source

Kortsch, S. et al. 2015. Climate change alters the structure of arctic marine food webs due to poleward shifts of boreal generalists. *Proc. R. Soc. B* 282: 20151546.

Examples

```
data(boreal)
```

 NOSM_bip

Compute NOS using a bipartite network

Description

Computation of NOS using a bipartite network (e.g. plant-pollinator network), where nodes can be formally categorized into two distinct categories (e.g. plant-pollinators). All nodes in one category will be considered as potential partners for the nodes in the other category (and vice-versa).

Usage

```
NOSM_bip(net, perc = 1, sl = 1)
```

Arguments

net	A network, in the form of an edge list. This should be a matrix or dataframe with two columns. Each value in a column is a node (e.g. a food item in a trophic-web). Nodes can be identified using numbers or characters. For each row (i.e. node pair), the value in the first column is 'consumed' (or pollinated, parasitized etc) by the value in the second column.
perc	(default to 1) - the fraction of node pair comparisons to be performed to compute NOS. We recommend performing all possible pair comparisons (perc = 1). However, for exploratory analyses on large sets of networks (or for very large networks), the possibility of using a lower fraction of pair comparisons is a useful option.
sl	(default is 1) Specifies whether cannibalistic interactions should be considered as possible and therefore taken into account and removed during computation ('1') or not ('0').

Value

A list (two elements) of class 'NOSM' with a 'Type' attribute 'bip'. The first element in the list is a vector of overlap values for the "in nodes" and the second element is a vector of overlap values for the "out nodes".

The [summary.NOSM](#) methods provides more useful summary statistics.

Examples

```
data(boreal)
y <- boreal[sample(rownames(boreal), 100, FALSE),] #subset 100 rows for speed
x <- NOSM_bip(y, perc = 1, sl = 1)
summary(x)
```

NOSM_dir	<i>Compute NOS using a directed network without a user provided network of potential interactions</i>
----------	---

Description

Computation of NOS using a directed network (e.g. food web) and without a user provided network of potential interactions. In a directed, unimode network, all nodes having at least an in-coming link will be considered as potential partners for nodes having at least an out-going link and vice-versa.

Usage

```
NOSM_dir(net, perc = 1, sl = 1)
```

Arguments

net	A network, in the form of an edge list. This should be a matrix or dataframe with two columns. Each value in a column is a node (e.g. a food item in a trophic-web). Nodes can be identified using numbers or characters. For each row (i.e. node pair), the value in the first column is 'consumed' (or pollinated, parasitized etc) by the value in the second column.
perc	(default to 1) - the fraction of node pair comparisons to be performed to compute NOS. We recommend performing all possible pair comparisons (perc = 1). However, for exploratory analyses on large sets of networks (or for very large networks), the possibility of using a lower fraction of pair comparisons is a useful option.
sl	(default is 1) Specifies whether cannibalistic interactions should be considered as possible and therefore taken into account and removed during computation ('1') or not ('0').

Value

A list (two elements) of class 'NOSM' with a 'Type' attribute 'Dir'. The first element in the list is a vector of overlap values for the "in nodes" and the second element is a vector of overlap values for the "out nodes".

The [summary.NOSM](#) methods provides more useful summary statistics.

Examples

```
data(boreal)
y <- boreal[sample(rownames(boreal), 100, FALSE),] #subset 100 rows for speed
x <- NOSM_dir(y, perc = 1, sl = 1)
summary(x)
```

NOSM_POT_dir	<i>Compute NOS using a directed network and with a user provided network of potential interactions</i>
--------------	--

Description

Computation of NOS using an directed network (e.g. a food web) and with a user provided network of potential interactions.

Usage

```
NOSM_POT_dir(net, pot_net, perc = 1, sl = 1)
```

Arguments

net	A network, in the form of an edge list. This should be a matrix or dataframe with two columns. Each value in a column is a node. Nodes can be identified using numbers or characters.
pot_net	A network of all potential interactions. These should include, as a minimum, all the observed interactions (i.e. all links in net), plus any other possible interaction (such as all those permitted by a certain trophic rule). pot_net should have the same structure as net (e.g. it should be a data frame or matrix).
perc	(default to 1) - the fraction of node pair comparisons to be performed to compute NOS. We recommend performing all possible pair comparisons (perc = 1). However, for exploratory analyses on large sets of networks (or for very large networks), the possibility of using a lower fraction of pair comparisons is a useful option.
sl	(default is 1) Specifies whether cannibalistic interactions should be considered as possible and therefore taken into account and removed during computation ('1') or not ('0').

Value

A list (two elements) of class 'NOSM' with a 'Type' attribute 'Pot_dir'. The first element in the list is a vector of overlap values for the "in nodes" and the second element is a vector of overlap values for the "out nodes".

The [summary.NOSM](#) methods provides more useful summary statistics.

Examples

```
data(boreal)
y <- boreal[1:300,] #subset 300 rows for speed
d <- sample(nrow(y), 200, replace = FALSE) #create a random pot_net
pot_net <- y[d,] #by randomly sampling 200 rows from boreal
x <- NOSM_POT_dir(y, pot_net, perc = 1, sl = 1)
summary(x)
```

NOSM_POT_undir	<i>Compute NOS using an undirected network and with a user provided network of potential interactions</i>
----------------	---

Description

Computation of NOS using an undirected network (e.g. a social co-occurrence network) and with a user provided network of potential interactions. In an undirected network, all nodes are considered as potential interacting partners.

Usage

```
NOSM_POT_undir(net, pot_net, perc = 1, sl = 1)
```

Arguments

net	A network, in the form of an edge list. This should be a matrix or dataframe with two columns. Each value in a column is a node. Nodes can be identified using numbers or characters.
pot_net	A network of all potential interactions. These should include, as a minimum, all the observed interactions (i.e. all links in net), plus any other possible interaction (such as all those permitted by a certain trophic rule). pot_net should have the same structure as net (e.g. it should be a data frame or matrix).
perc	(default to 1) - the fraction of node pair comparisons to be performed to compute NOS. We recommend performing all possible pair comparisons (perc = 1). However, for exploratory analyses on large sets of networks (or for very large networks), the possibility of using a lower fraction of pair comparisons is a useful option.
sl	(default is 1) Specifies whether cannibalistic interactions should be considered as possible and therefore taken into account and removed during computation ('1') or not ('0').

Value

A list of class 'NOSM' with a 'Type' attribute 'Pot_undir', containing a vector of overlap values. The [summary.NOSM](#) methods provides more useful summary statistics.

Examples

```
data(boreal)
y <- boreal[1:300,] #subset 300 rows for speed
d <- sample(nrow(y), 200, replace = FALSE) #create a random pot_net
pot_net <- y[d,] #by randomly sampling 200 rows from boreal
x <- NOSM_POT_undir(y, pot_net, perc = 1, sl = 1)
summary(x)
```

NOSM_undir	<i>Compute NOS using an undirected network without a user provided network of potential interactions</i>
------------	--

Description

Computation of NOS using an undirected network (e.g. a social co-occurrence network) and without a user provided network of potential interactions. In an undirected network, all nodes are considered as potential interacting partners.

Usage

```
NOSM_undir(net, perc = 1, sl = 1)
```

Arguments

net	A network, in the form of an edge list. This should be a matrix or dataframe with two columns. Each value in a column is a node. Nodes can be identified using numbers or characters.
perc	(default to 1) - the fraction of node pair comparisons to be performed to compute NOS. We recommend performing all possible pair comparisons (perc = 1). However, for exploratory analyses on large sets of networks (or for very large networks), the possibility of using a lower fraction of pair comparisons is a useful option.
sl	(default is 1) Specifies whether cannibalistic interactions should be considered as possible and therefore taken into account and removed during computation ('1') or not ('0').

Value

A list of class 'NOSM' with a 'Type' attribute 'Undir', containing a vector of overlap values. The [summary.NOSM](#) methods provides more useful summary statistics.

Examples

```
data(boreal)
y <- boreal[sample(rownames(boreal), 100, FALSE),] #subset 100 rows for speed
x <- NOSM_undir(y, perc = 1, sl = 1)
summary(x)
```


summary.NOSM

*Summarising the results of the five main NOSM functions***Description**

S3 method for class 'NOSM'. summary.NOSM creates summary statistics for objects of class NOSM. The exact summary statistics computed depends on the 'Type' attribute (e.g. 'bip') of the NOSM object (see below). The summary method generates more useful information for the user than the standard NOSM functions. Another S3 method (print.summary.NOSM; not documented) is used to print the output.

Usage

```
## S3 method for class 'NOSM'
summary(object, ..., y = 3)
```

Arguments

object	An object of class 'NOSM'.
...	further arguments passed to or from other methods.
y	(default of 3) The adjustment value for the computation of the z value (see Strona & Veech, 2015).

Value

Returns object of class 'summary.NOSM' with a Type attribute (e.g. 'bip') which is inherited. For NOSM objects of Type 'Pot_dir', 'bip' or 'Dir', the summary.NOSM method returns the mean of the overlap values for the "in nodes" (NOS_In), the mean of the overlap values for the "out nodes" (NOS_Out), the mean of Nos In and Nos Out (NOS), the standard deviation of the overlap values for the "in nodes" (MOD_In), the SD of the overlap values for the "out nodes" (MOD_Out), and the SD of the combined set of overlap values (MOD; network modularity).

For NOSM objects of Type 'Dir' and 'Undir', the summary.NOSM method returns just the NOS and MOD values (network modularity).

For all types of NOSM object, the z value and associated p value are also provided (see Strona & Veech, 2015).

References

Strona, G. & Veech, J. A. (2015). A new measure of ecological network structure based on node overlap and segregation. *Methods in Ecology and Evolution*, 6(8), 907-915.

See Also

[NOSM_bip](#), [NOSM_POT_dir](#), [NOSM_POT_undir](#), [NOSM_dir](#), [NOSM_undir](#)

Examples

```
data(boreal)
z <- boreal[sample(rownames(boreal), 200, FALSE),] #subset for speed
x <- NOSM_bip(z, perc = 1, sl = 1)
summary(x, y = 3)
```

Index

*Topic **datasets**

boreal, [3](#)

boreal, [3](#)

nos-package, [2](#)

NOSM_bip, [4](#), [9](#)

NOSM_dir, [5](#), [9](#)

NOSM_POT_dir, [6](#), [9](#)

NOSM_POT_undir, [7](#), [9](#)

NOSM_undir, [8](#), [9](#)

summary.NOSM, [4–8](#), [9](#)