Using bipartite to describe and plot two-mode networks in *R*

Carsten F. Dormann
Biometry & Environmental System Analysis
University of Freiburg, Germany

Jochen Fründ
Animal Network Ecology
University of Hamburg, Germany

September 26, 2025

Abstract

This vignette introduces the bipartite package to newbees. It describes it intention and, in seperate sections, the use of the most important functions. It also critically comments on some approaches and presents functions from related packages.

Contents

1 Introduction

1.1 Introduction for ecologists

Many interactions between species are between distinct groups, such as pollinators and flowers, or herbivores and plants, or parasitoids and their hosts, or predators and their prey. When no information on interactions *within* groups is known, we have the case of a *bipartite* (a.k.a. "two-mode") network. In constrast, if every species could interact with every other, and we have (necessarily incomplete) information on these interactions, we have the case of a "unipartite" or "one-mode" network (think: food web).

1.2 Introduction for non-ecologists

In mathematics, there is a field called "graph theory", which describes/analyses/formalises "graphs" and the operations among them. A graph consists of *nodes* and *edges* (which are the links between nodes). A node can be something like an airport with edges being the flights between them; a power grid of power plants and energy consumers, with the power lines being the edges; a list of industry "leaders" and the edges representing financial ties between them; or, e.g. in ecology, a list of species, with edges representing trophic relationships, i.e. who eats whom.

When the nodes are grouped into two distinct sets, with no (known or quantified) interactions *within* a set, then we call it a bipartite network ("two parts", a.k.a. "two-mode" network). In our examples these could be airports of different continents (and no intracontinental flights are in our data base); power plants and energy consumers; industry executives and companies (where an executive can sit in the board of different companies); or pollinators and plants.

1.3 Introduction continued for both, ecologists and non-ecologists

This vignette tries to illustrate and explain a package that deals almost exclusively with *bipartite* networks!

The typical tasks we want to carry out are

1. visualise the network;
2. compute network indices to summarise its structure;
3. statistically test for differences between networks;
4. statistically test for differences between observed and random networks.

bipartite aims to facilitate this steps. This vignette aims to facilitate the use of **bipartite**. We shall try to predominantly use the technical terms ("node" and "edges"), but our examples will be from the ecological realm, where nodes are (typcially) species, and edges are typically called "links", which in fact are observed (or assumed) ecological interactions (such as pollination, predation, dispersal). Apologies for shifting between these two for easier access!

Also, as ecologists we are not "graph theoreticians" or mathematicians. Often, in the ecological literature, mathematical terms are used to elevate the status of the author, at no benefit to the analysis or to the reader. We are hesitant to use pretend-mathematics, at the risk of sounding daft to a person familiar with graph theory.

Of course, we don't mind if this documented would be cited, e.g. as Dormann (2019). Alternatively, you can always cite our work on this subject, inevitably also using (some functionality of) **bipartite** (Dormann *et al.*, 2008, 2009; Dormann, 2011; Dormann & Strauß, 2014; Dormann *et al.*, 2017; Dormann & Blüthgen, 2017)!

2 *What other network packages are out there in the R-universe?*

Before going through what **bipartite** offers, we shall briefly list what other packages in this direction can do for you:[1]

There are several open-source packages devoted to the analysis of networks and food webs, but only few addressing bipartite networks explicitly. Of these, **bipartite** provides the most extensive set of features, both for plotting, computing indices, and for null models. For the sake of completeness, here are some other *R* packages (on CRAN) we are aware of for handling bipartite networks (in alphabetical order). Some of these are exemplified in section 8.

biGraph discontinued and archived;

bipartiteD3 interactive visualisation of bipartite networks;[2]

bmotif for counting "motifs" in bipartite networks;

dynsbm implements a modularity-like search through multiple time slices of a (dynamic) network ("dynamic stochastic block model" Miele & Matias, 2017);

econullnetr seems to be largely a wrapper for **bipartite**;

econetwork implements the core-periphery measures of Martín González *et al.* (2020) and allows computation of distances between networks following Ohlmann *et al.* (2019);

HiveR for fancy "hive" plots;

igraph provides functionalities largely for one-mode networks, quite a bit of plotting options, too;

networkD3 various interactive visualisations of networks (also using the "D3" JavaScript library) including a "Sankey diagram" that is somewhat similar to a bipartite graph;

NIMAA aims at exploring and imputing bipartite data in a pharmacological context;

sna provides many traditional one-mode metrics, from a "social network analysis" perspective; in a way the bigger brother of **bipartite**;

statnet a set of largely one-mode network packages, includes **ergm, networksis, statnet.common, tergm**;

tnet, which aims at re-thinking one-mode indices in a mathematically consistent way for bipartite networks; some functions of **tnet** were lifted to **bipartite** (thanks to Tore for allowing that!);

vegan for various analyses of a community matrix, including null models and nestedness computations;

WGCNA designed for gene assays, has a small set of indices and options overlapping with those of **bipartite**.

Additionally, there may well be dozens of packages on github or r-forge, which remain obscure until one stumbles over them.

bipartgraph, at `https://github.com/jgalgarra/bipartgraph`, offers polar and Ziggurat, building on his other package **kcorebip**;

ggbipart, at `https://pedroj.github.io/bipartite_plots/`, allows plotting bipartite networks in the spirit of **ggplot2**; so far (2021), there is no obvious benefit over the functions that come with **bipartite**;

[1] This is like a mini-CRAN-task view.

[2] And no, this is not a typo, it is "D3", which apparently is a JavaScript library for "data-driven documentation"

infomapecology implements the “Map Equation”, which identifies modules (Farage *et al.*, 2021, so far only on github: `https://github.com/Ecological-Complexity-Lab/infomap_ecology_package`).

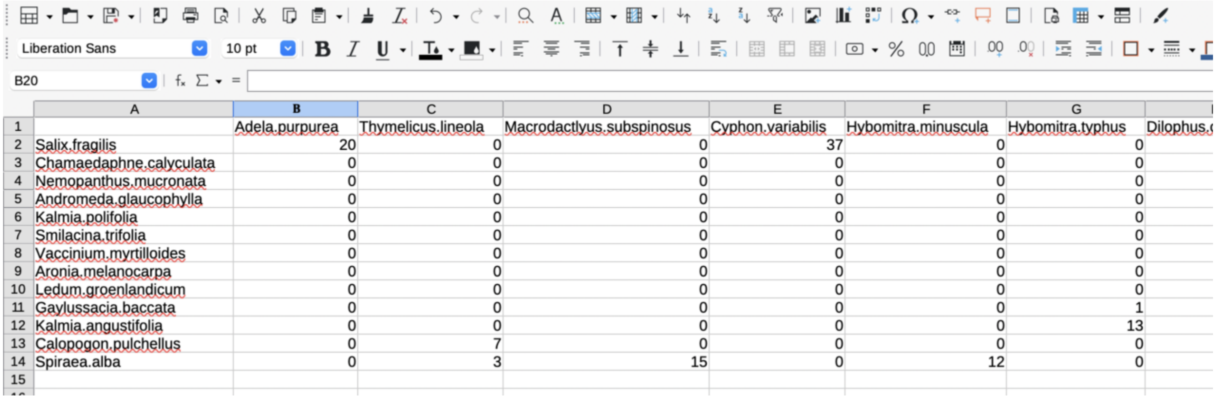
3 *Preparing data for analysis*

Most functions in **bipartite** require the same type of input structure: an adjacency matrix of what-interacts-with-what, for short called “web”.

Researchers may have organised their data in different ways, primarily as *wide* or *long* format.

3.1 *From* wide *format in a spreadsheet to ready-for-analysis for* **bipartite**

The wide format is how **bipartite** wants the data to be: lower level species as rows, higher level species as columns, with the names of each as the first row and column, respectively:



	A	B	C	D	E	F	G	
1		Adela.purpurea	Thymelicus.lineola	Macrodactlyus.subspinosus	Cyphon.variabilis	Hybomitra.minuscula	Hybomitra.typhus	Dilophus.
2	Salix.fragilis	20	0	0	37	0	0	
3	Chamaedaphne.calyculata	0	0	0	0	0	0	
4	Nemopanthus.mucronata	0	0	0	0	0	0	
5	Andromeda.glaucophylla	0	0	0	0	0	0	
6	Kalmia.polifolia	0	0	0	0	0	0	
7	Smilacina.trifolia	0	0	0	0	0	0	
8	Vaccinium.myrtilloides	0	0	0	0	0	0	
9	Aronia.melanocarpa	0	0	0	0	0	0	
10	Ledum.groenlandicum	0	0	0	0	0	0	
11	Gaylussacia.baccata	0	0	0	0	0	1	
12	Kalmia.angustifolia	0	0	0	0	0	13	
13	Calopogon.pulchellus	0	7	0	0	0	0	
14	Spiraea.alba	0	3	15	0	12	0	
15								

This screenshot is from LibreOffice Calc, and writes names with a “.” as separator between genus and epithet. That is a matter of taste (and actually, I do not like it).

After saving as .csv, we read this file in as:

```
getwd()

## [1] "/Users/carsten/Data/repositories/bipartite/bipartite/vignettes"

small1976 <- read.csv("small1976.csv", row.names=1)
```

The `row.names`-argument ensures that the names in the first column are used as (row) labels in the data set. Note that this object is a data.frame, not a matrix, even though it contains only numbers. **bipartite** internally converts it into a matrix if that is the required format for a function.

If we want to have nicer name layouts, with an empty space instead of the ugly point, we can change it like this:

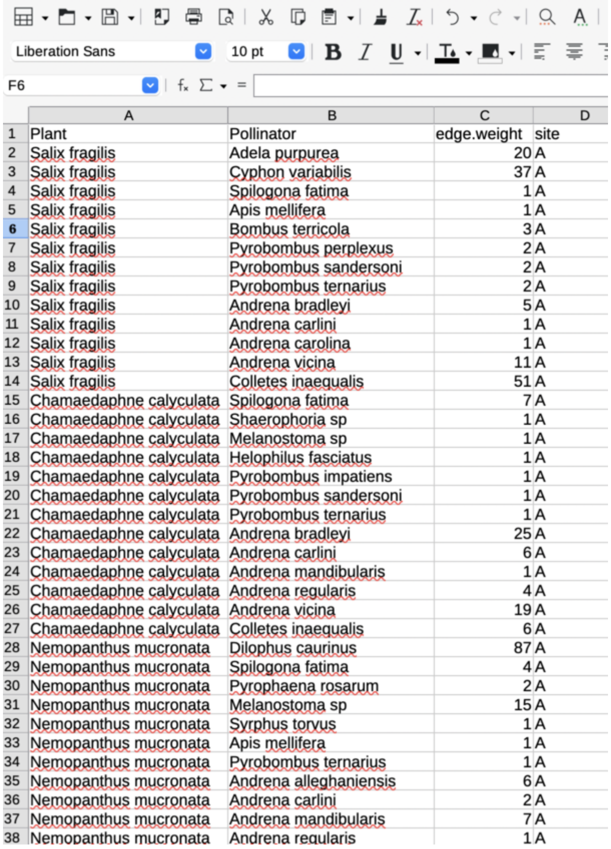
```
rownames(small1976) <- gsub(".", " ", rownames(small1976), fixed=T)
colnames(small1976) <- gsub(".", " ", colnames(small1976), fixed=T)
small1976[1:5, 1:2] # to save space

##                          Adela purpurea Thymelicus lineola
## Salix fragilis                       20                  0
## Chamaedaphne calyculata               0                  0
## Nemopanthus mucronata                 0                  0
## Andromeda glaucophylla                0                  0
## Kalmia polifolia                      0                  0
```

This object is now in the correct format for further analysis and plotting with **bipartite**.

3.2 *From* long *format in a spreadsheet to ready-for-analysis for* ***bipartite***

Raw data often start as tables of individual observations, e.g. one column containing the name of the species in the first level, another column the name of the species in the other level with which it interacts. A third column records the number of interactions for this pair, i.e. an "edge weight":



Liberation Sans 10 pt

F6

	A	B	C	D
1	Plant	Pollinator	edge.weight	site
2	Salix fragilis	Adela purpurea	20	A
3	Salix fragilis	Cyphon variabilis	37	A
4	Salix fragilis	Spilogona fatima	1	A
5	Salix fragilis	Apis mellifera	1	A
6	Salix fragilis	Bombus terricola	3	A
7	Salix fragilis	Pyrobombus perplexus	2	A
8	Salix fragilis	Pyrobombus sandersoni	2	A
9	Salix fragilis	Pyrobombus ternarius	2	A
10	Salix fragilis	Andrena bradleyi	5	A
11	Salix fragilis	Andrena carlini	1	A
12	Salix fragilis	Andrena carolina	1	A
13	Salix fragilis	Andrena vicina	11	A
14	Salix fragilis	Colletes inaequalis	51	A
15	Chamaedaphne calyculata	Spilogona fatima	7	A
16	Chamaedaphne calyculata	Shaerophoria sp	1	A
17	Chamaedaphne calyculata	Melanostoma sp	1	A
18	Chamaedaphne calyculata	Helophilus fasciatus	1	A
19	Chamaedaphne calyculata	Pyrobombus impatiens	1	A
20	Chamaedaphne calyculata	Pyrobombus sandersoni	1	A
21	Chamaedaphne calyculata	Pyrobombus ternarius	1	A
22	Chamaedaphne calyculata	Andrena bradleyi	25	A
23	Chamaedaphne calyculata	Andrena carlini	6	A
24	Chamaedaphne calyculata	Andrena mandibularis	1	A
25	Chamaedaphne calyculata	Andrena regularis	4	A
26	Chamaedaphne calyculata	Andrena vicina	19	A
27	Chamaedaphne calyculata	Colletes inaequalis	6	A
28	Nemopanthus mucronata	Dilophus caurinus	87	A
29	Nemopanthus mucronata	Spilogona fatima	4	A
30	Nemopanthus mucronata	Pyrophaena rosarum	2	A
31	Nemopanthus mucronata	Melanostoma sp	15	A
32	Nemopanthus mucronata	Syrphus torvus	1	A
33	Nemopanthus mucronata	Apis mellifera	1	A
34	Nemopanthus mucronata	Pyrobombus ternarius	1	A
35	Nemopanthus mucronata	Andrena alleghaniensis	6	A
36	Nemopanthus mucronata	Andrena carlini	2	A
37	Nemopanthus mucronata	Andrena mandibularis	7	A
38	Nemopanthus mucronata	Andrena regularis	1	A

Also, several networks may be included in this table, so another column would provide a web-ID (e.g. a name of the site or the web itself). The function `frame2webs` is a convenience function to convert such tables into one or more webs. Its output is a list or an array of webs in the appropriate format for further analysis.[3]

```
library(bipartite)
small.long <- read.csv("small1976long.csv")
small.wide <- frame2webs(small.long, varnames=c("Plant", "Pollinator", "site", "edge.weight"))
str(small.wide)

## List of 1
##  $ A: num [1:13, 1:34] 0 0 0 0 0 0 0 0 0 20 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ Plant      : chr [1:13] "Andromeda glaucophylla" "Aronia melanocarpa""..
##   .. ..$ Pollinator: chr [1:34] "Adela purpurea" "Andrena alleghaniensis" "An"..

small.wide[[1]][, 1:2] # to save space

##                             Pollinator
## Plant                        Adela purpurea Andrena alleghaniensis
##   Andromeda glaucophylla                  0                      0
##   Aronia melanocarpa                      0                      3
##   Calopogon pulchellus                    0                      0
##   Chamaedaphne calyculata                 0                      0
##   Gaylussacia baccata                     0                      0
```

[3] `webs2array` can also make an array of previously separate webs, padding non-overlapping species with zeros.

```
##   Kalmia angustifolia                  0                      0
##   Kalmia polifolia                     0                      0
##   Ledum groenlandicum                  0                      2
##   Nemopanthus mucronata                0                      6
##   Salix fragilis                      20                      0
##   Smilacina trifolia                   0                      0
##   Spiraea alba                         0                      0
##   Vaccinium myrtilloides               0                      0
```

As we see, this returns a list of networks, even if only site was provided. We can access this element using double brackets "`[[1]]`". We now are back to our wide format, as required for most **bipartite** analyses.

3.3 *From web to edge list*

Many other packages (in *R* or otherwise) are based on one-mode representations of networks. The typical input file would be the edge list of the previous section. The function `web2edges` transforms bipartite "webs" into edge lists to be used e.g. in Pajek or by **sna** or **tnet**.

```
head(web2edges(small1976)) # to save space

##                      row col edge.weights
## Salix.fragilis         1  14           20
## Calopogon.pulchellus  12  15            7
## Spiraea.alba          13  15            3
## Spiraea.alba          13  16           15
## Salix.fragilis         1  17           37
## Spiraea.alba          13  18           12
```

Note that this format is different to the edge list used as input above. That is because often not the names but the positions are used as reference. That is, the numbers refers to the position in a vector {lower level species, higher level species}. In this specific case, the first pollinator species is number 14, as there are 13 plant species. Note that the row names only list the plant species: it is *this* plant that is visited by pollinator 14. To get the full list of names, we can simply append row and column names:

```
head(c(rownames(small1976), colnames(small1976)), 16)

##  [1] "Salix.fragilis"            "Chamaedaphne.calyculata"
##  [3] "Nemopanthus.mucronata"     "Andromeda.glaucophylla"
##  [5] "Kalmia.polifolia"          "Smilacina.trifolia"
##  [7] "Vaccinium.myrtilloides"    "Aronia.melanocarpa"
##  [9] "Ledum.groenlandicum"       "Gaylussacia.baccata"
## [11] "Kalmia.angustifolia"       "Calopogon.pulchellus"
## [13] "Spiraea.alba"              "Adela.purpurea"
## [15] "Thymelicus.lineola"        "Macrodactlyus.subspinosus"
```

4 *Visualising bipartite networks*

A useful first step in the analysis of bipartite networks is to plot it. To do so, **bipartite** offers two main functions, `plotweb` and `visweb`. They produce a bipartite graph (e.g. Fig. 1 top) and a prettified version of the adjacency matrix (Fig. 1 bottom), respectively (Motten, 1982). Both offer a wide range of options to change sizes, colours and placement of labels. Please run example code for these functions to get a feeling for their abilities (and hopefully you're

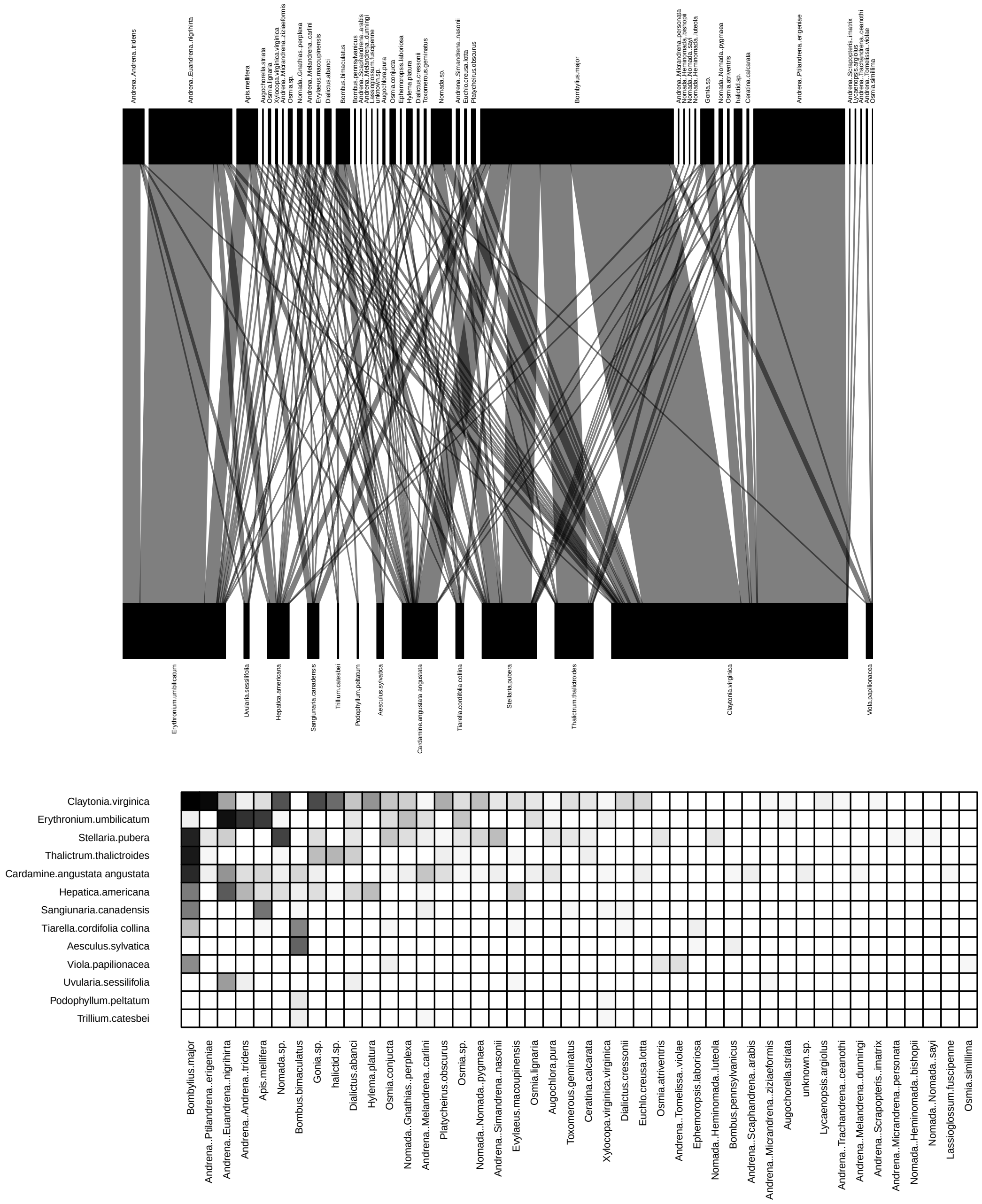


Figure 1: A bipartite graph of Motten's (1982) pollination network (*top*) and a visualisation of the adjacency matrix (*bottom*). The darker a cell is represented, the more interactions have been observed. By default, `plotweb` minimises overlap of lines and `visweb` sorts by marginal totals.

impressed).[4] For a more in-depth guide on how to use the `ploweb` function also have a look at the `PlottingWithBipartite` vignette.

```
par(xpd=T)
plotweb(motten1982, srt= 90, sorting = "ca")
visweb(motten1982)
```

If these options are still not sufficient, and you want more customisation, such as side-by-side plotting of several networks, please check out the **bipartiteD3**, which provides the power

[4] The additional function `plotweb2` is still a bit experimental but allows to combine several bipartite plots into a stack of tripartite, quadripartite, pentapartite etc. graphs. See the examples in the function's help page for illustration.

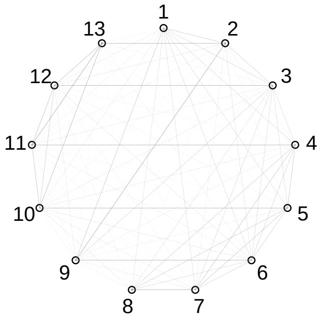


Figure 2: A PAC-plot of Motten's 1982 pollination network. Lines connect plants potentially visited next by a pollinator. Line width is proportional to probability, which is very similar in this example. See help of *plotPAC* for details on plotting and meaning.

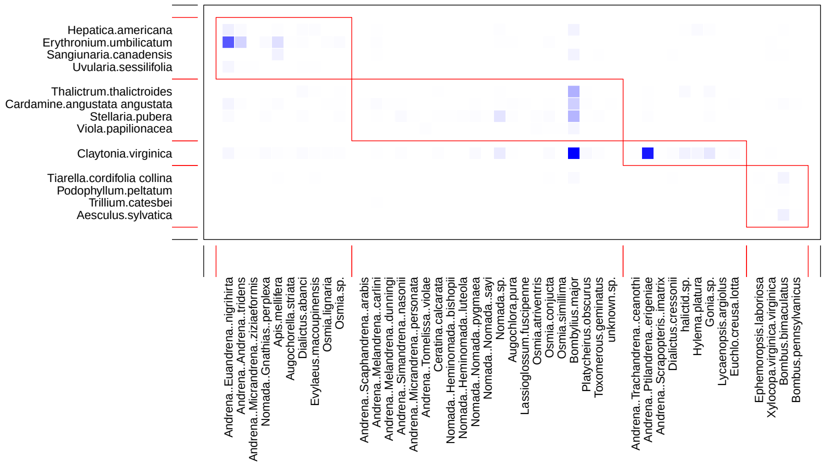


Figure 3: The adjacency matrix of Motten's (1982) pollination network, organised into modules.

of the "D3" library of JavaScript.[5] Note that D3 makes *interactive* figures, so on mouse-over the figures change. While not a feature useful for publication in the current standard, it adds 21st century appeal to such graphs.

The function `plotPAC` visualises (Fig. 2) a very special idea: the potential for apparent competition (Morris *et al.*, 2005). The idea is that every parasitoid emerging from a host can attack another species. Thus, the more individuals emerge from a host, the larger is the potential for these to reduce the competition between hosts, as long as they choose their prey equiprobable. Pretty as it may be, its current value is highest for host-parasitoid networks, so the plot employing a pollination network is for illustration only.

```r
plotPAC(PAC(motten1982), outby=0.9)
```

We can also visualise communities within the network by computing its modularity and then plotting the result, using the functions `computeModules` and `plotModuleWeb`, respectively (Fig. 3).

```r
mod <- computeModules(motten1982)
plotModuleWeb(mod)
```

To visualise either level separately projected into one-mode mode, we have to employ a different package, for example **sna** (Butts, 2013). We can do the projection on the fly, though.

[5] The package explicitly links to "our" bipartite, although I think it is largely independent of it, and actually very nice!

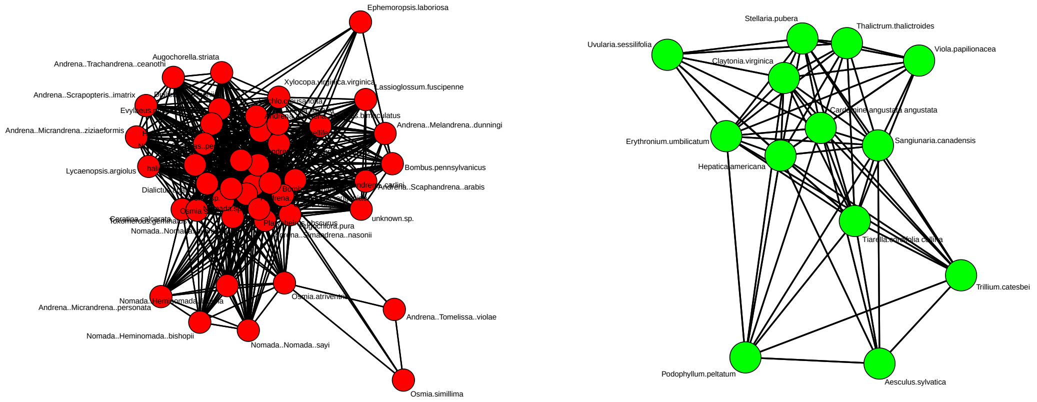


Figure 4: Plots of one-mode projections of Motten's (1982) pollination network. Pollinators in red (*top*), plants in green (*bottom*). Note that these plots are point-symmetrically variable, hence every plot will look somewhat different.

For networks with even a moderate number o species, such plots can become rather obtuse and uninformative (Fig. 4 top).

```
par(mfrow=c(1,2), xpd=T)
sna::gplot(as.one.mode(motten1982, project="higher"),
           label=colnames(motten1982), gmode="graph",
           label.cex=0.6, vertex.cex=2)
sna::gplot(as.one.mode(motten1982, project="lower"),
           label=rownames(motten1982), gmode="graph",
           label.cex=0.6, vertex.cex=2, vertex.col="green")
```

5 Computing indices

Network indices are primarily that: metrics to quantify an aspect of the network. Each may have a justification, a purpose, but the information in a matrix is only finite. As a consequence, network indices are highly correlated. There is thus no point in computing all indices in the book; rather should we consider very carefully what we want to test with the data and select one (or a few) indices to do so.

The **bipartite**-package offers itself to data dredging: with one function one can compute dozens of indices, certainly some of them will be significantly correlated with whatever we want to test. While analyses are commonly carried out in this way, this is obviously statistically inappropriate. Two issues are important here. One is that of multiple testing, and there are dozens of publications covering this issue. The other is an attempt to formalise one's expectation. Null models are one way to do so, in particular one's expectation of what a pattern is like in the *absence* of a process. This technical details of this topic are covered further below (section 6.3).

Indices (currently) come at the following hierarchical levels:

1. network-level (a single value, per index, for the entire network);

2. group-level (a value, per index, for each of the two groups);

3. link-level (a value, per index, for each edge = link); and

4. species-level (= node-level) (a value, per index, for each species).

5.1 Network-level indices

This set of indices is computed for the entire network. They are called using the function `networklevel`, which additionally (and conveniently) makes available all indices from the group level, too. Those restricted to the network-level are:

1. `connectance`, also called 'standardised number of species combinations' in biogeographic analyses (Gotelli & Graves, 1996; Dunne *et al.*, 2002);

2. `web asymmetry`, the balance of the number of species in the two levels (Blüthgen *et al.*, 2007);

3. `links per species`;

4. `number of compartments`;

5. `compartment diversity` (Tylianakis *et al.*, 2007);

6. `cluster coefficient`, which will compute both the network-wide binary, one-mode-based cluster coefficient as well as those for each level,

7. `nestedness`, with the returned value indicating the 'temperature' of the matrix, which is low (0) for a perfectly nested matrix (Rodríguez-Gironés & Santamaría, 2006);

8. `NODF`, as an alternative measure of quantitative nestedness, automatically correcting for matrix filling and also indicating nestedness by values towards 1 (Almeida-Neto *et al.*, 2008);

9. `weighted nestedness`, or more specifically 'weighted interaction nestedness estimator' (WINE), with 1 indicating perfected nestedness (thus exactly opposite to the previous index) (Galeano *et al.*, 2009);

10. `weighted NODF`, as an alternative measure of quantitative nestedness, automatically correcting for matrix filling and also indicating nestedness by values towards 1 (Almeida-Neto & Ulrich, 2011);

11. `interaction strength asymmetry` (Blüthgen *et al.*, 2007) (or alternatively `ISA` or `dependence asymmetry`: Bascompte *et al.*, 2003), which quantify whether specialised species interact with generalised ones in the other level (or vice versa);

12. `specialisation asymmetry` (or alternatively `SA`), following the same idea as the previous, but this being based on species specialisation in d';

13. `linkage density`, marginal totals-weighted diversity of interactions per species (Bersier *et al.*, 2002);

14. `Fisher alpha`, diversity index based on fitting a Fisher-series (Fisher *et al.*, 1943);

15. `interaction evenness` quantifies how balanced the distribution of interactions is across species, based on Shannon's diversity (Tylianakis *et al.*, 2007);

16. `Alatalo interaction evenness`, as an alternative measure of interaction evenness, attempting to overcome some of the shortcomings of the previous, Shannon's, version (Alatalo, 1981; Müller *et al.*, 1999);

17. `Shannon diversity`;

18. `H2`, network-wide specialisation index H_2' (Blüthgen *et al.*, 2006).

Similarly to the previous sets of indices, several options are available to fine-tune those in `networklevel`:

```
networklevel(bezerra2009, index=c("ISA", "weighted NODF", "Fisher alpha"),
             SAmethod="log")

##                  weighted NODF interaction strength asymmetry
##                   5.717160e+01                   8.842823e-05
##                   Fisher alpha
##                   8.793401e+00
```

5.2 *Group-level indices*

Indices that can be computed for each group separately are collected in the function `grouplevel`.[6] Quite a few of these indices are aggregates from the species level (e.g. by averaging species' degrees), in which case the mean is weighted by the number of observations of each species, thus giving more weight to species for which information is more reliable.[7] Others are biogeographic indices which change their value when the matrix is inverted (i.e. when a group changes from "species" to "islands"). The following indices are implemented:

1. `number of species` in the respective trophic level;

2. `mean number of links`;

3. `mean number of shared partners` (Roberts & Stone, 1990; Stone & Roberts, 1992);

4. `cluster coefficient`, which supposedly informs us whether a network has 'small world' properties (Watts & Strogatz, 1998);

5. `weighted cluster coefficient`, which takes into account the number of interactions and is a properly derived bipartite version of the previous one-mode cluster coefficent above (Opsahl, 2010);

[6] If network-level indices are also desired, all indices in `grouplevel` are also accessible through `networklevel`, see below.

[7] The weighting can be switched off by stating `weighted=FALSE`.

6. togetherness as the mean number of co-occurrences across all pairwise species combinations (Stone & Roberts, 1992);

7. C score, or ‘checkerboardness’, which averages the number of instances of 01/10-patterns (i.e. exclusive occurrences) for all pairwise species combinations (Stone & Roberts, 1990);

8. V ratio, the variance ratio of species numbers to interaction numbers within species of a level (Schluter, 1984);

9. discrepancy, the number of links one would have to move to achieve perfect nestedness (Brualdi & Sanderson, 1999);

10. degree distribution, fitting an exponential function and (truncated) power laws(Jordano *et al.*, 2003);

11. extinction slope, with different options for the extinction sequence (Memmott *et al.*, 2004);

12. robustness, the area under the extinction curve (Burgos *et al.*, 2007);

13. niche overlap, with options for distance metrics (Pielou, 1972; Hurlbert, 1978);

14. generality (for the higher trophic level) or vulnerability (for the lower one) (Bersier *et al.*, 2002);

15. partner diversity, simply the mean Shannon diversity of interactions of each species in a level;

16. effective partners as before, but as the exponent of the base to which Shannon was computed (typically *e* or 2) (Jost, 2006);

17. fd (or alternatively functional diversity), which is similar to niche overlap but computed as branch length of a cluster diagram of dissimilarity of resource use (Devoto *et al.*, 2012).

As in the case of species-level indices, some allow for normalisation (between 0 and 1), others for different basis to the logarithm or similar tuning. A typical call to grouplevel will thus comprise the list of desired indices[8] and their options, as well as the level(s) for which the computation shall be carried out:

```
grouplevel(bezerra2009, level="both", index=c("mean number of links", "weighted
     cluster coefficient", "effective partners", "niche overlap"), dist="bray")

## mean.number.of.links.HL mean.number.of.links.LL        niche.overlap.HL
##               9.9502551               7.0975057               0.2824538
##        niche.overlap.LL           generality.HL        vulnerability.LL
##               0.4795595               8.0590841               5.6963672
```

The LL and HL behind each index indicate that it was computed for the lower and higher level, respectively. A relatively comprehensive comparison of network-level indices (Dormann *et al.*, 2009), including some at the group level, has shown substantial redundancy among these indices, and few of them have a sound theoretical basis.

5.3 *Link-level indices*

At the moment, only few indices have been investigated at the level of the individual link (i.e. the cell of a network matrix):

1. dependence (i.e. the relevance of each species for the other level: Bascompte *et al.*, 2006) one matrix for each level;

[8] Or alternatively the argument index="ALL".

2. endpoint degree (i.e. product of degrees of species linked by this cell: Barrat *et al.*, 2004).

It is employed by simply applying it to the network under consideration and selecting the index desired:

```
str(linklevel(bezerra2009, index=c("dependence", "endpoint")))

## List of 3
##  $ HL dependence: num [1:13, 1:13] 0.193 0.131 0.056 0.108 0.105 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:13] "Diplopterys.pubipetala" "Byrsonima.gardnerana" "Banist"..
##   .. ..$ : chr [1:13] "Centris.aenea" "Centris.fuscata" "Centris.caxiensis" ""..
##  $ LL dependence: num [1:13, 1:13] 0.2415 0.2238 0.0997 0.2474 0.2632 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:13] "Diplopterys.pubipetala" "Byrsonima.gardnerana" "Banist"..
##   .. ..$ : chr [1:13] "Centris.aenea" "Centris.fuscata" "Centris.caxiensis" ""..
##  $ endpoint     : num [1:13, 1:13] 117 78 156 104 91 65 52 26 52 52 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:13] "Diplopterys.pubipetala" "Byrsonima.gardnerana" "Banist"..
##   .. ..$ : chr [1:13] "Centris.aenea" "Centris.fuscata" "Centris.caxiensis" ""..
```

This function is so rudimentary, and its output so disproportionally voluminous, that it is not worth going into further details here and use `str` to only show the structure of the output.

5.4 *Species-level (= node-level) indices*

The (growing) list of indices computable for each species in the network comprises (the index name to be used in the call is given in `typewriter` font):

1. `degree`, i.e. the number of links of each species,
2. `normalised degree` for a normalised version of degree (Martín González *et al.*, 2010),
3. `species strength` as sum of dependencies for each species (Bascompte *et al.*, 2006),
4. `nestedrank` as rank in a nested matrix (Alarcón *et al.*, 2008),
5. `interaction` for interaction push/pull (a version of dependence asymmetry), (Vázquez *et al.*, 2007),
6. `PDI` for Paired Differences Index (Poisot *et al.*, 2011b,a),
7. `resource range` for Schoener's index of unused resources (Schoener, 1989),
8. `species specificity` (or coefficient of variation of interactions) (Julliard *et al.*, 2006; Poisot *et al.*, 2012),
9. `PSI` for pollination service index (or pollinator support index, depending on the trophic level),[9]
10. `NS` for node specialisation index (Dalsgaard *et al.*, 2008),
11. `betweenness` for betweenness (Borgatti & Everett, 1997),
12. `closeness` (both automatically also return their weighted counterparts proposed by Tore Opsahl in package **tnet**),
13. `Fisher` for Fisher's alpha as a measure of diversity (Fisher *et al.*, 1943),

[9] devised by the author with Nico Blüthgen and Bernd Gruber

14. `diversity` for Shannon diversity of interactions of that species,

15. `effective partners` for the effective number of interacting partners (Bersier *et al.*, 2002),

16. `proportional generality` for a quantitative version of normalised degree,[10]

17. `proportional similarity` as specialisation measured as similarity between use and availability (Feinsinger *et al.*, 1981),

18. `d` for Blüthgen's discrimination/selectivity index d' (Blüthgen *et al.*, 2006).

Some of them allow for normalisation (between 0 and 1), others for different basis to the logarithm or similar tuning. A typical call to `specieslevel` will thus comprise the list of desired indices[11] and their options, as well as the level(s) for which the computation shall be carried out:

```
specieslevel(bezerra2009, level="lower", index=c("normalised degree", "PDI",
      "effective partners"), PDI.normalise=F)

##                            normalised.degree       PDI effective.partners
## Diplopterys.pubipetala             0.6923077 1010.0000           7.437631
## Byrsonima.gardnerana               0.4615385 1939.6667           3.819660
## Banisteriopsis.muricata            0.9230769  657.0000           9.191488
## Heteropterys.sp1                   0.6153846  570.3333           6.278285
## Heteropterys.sp2                   0.5384615  567.3333           5.865968
## Dicella.bracteosa                  0.3846154  429.0000           4.582040
## Carolus.chasei                     0.3076923  513.6667           3.751208
## Stigmaphyllon.paralias             0.1538462  774.0000           1.944388
## Banisteriopsis.stellaris           0.3076923  245.6667           3.787887
## Banisteriopsis.schizoptera         0.3076923  189.0000           3.822392
## Stigmaphyllon.auriculatum          0.3076923  204.3333           3.687447
## Stigmaphyllon.ciliatum             0.2307692  238.0000           2.937197
## Janusia.anisandra                  0.2307692  220.3333           2.813322
```

A relatively comprehensive comparison of species-level indices of specialisation, Dormann (2011) has shown substantial redundancy among these indices, and only few of them have a sound theoretical basis (Poisot *et al.*, 2012).

5.5 *Which index to choose?*

Different indices were invented ("developed") for different purposes. However, for most indices, it has not been demonstrated that they actually achieve what their inventor had in mind. Most commonly indices quantify a specific pattern which may have come about by a whole variety of different causes. Take, for example, connectance, i.e. the proportion of possible links actually recorded. Low connectance may be caused by high specialisation or by low sampling intensity. The same holds true for a species' degree, which (typically) increases with sampling effort as well as generalisation. For the choice of an index this means that we cannot solely rely on what the inventor has proposed this index to be good for, but we have to keep in mind that it may actually quantify several other things, too.

Apart from the different levels at which an index can quantify a network pattern (from the individual link to the entire network, as covered in sections 5.3–5.1), indices can be divided into those for binary and those for weighted data (called here binary and weighted indices, respectively). Binary indices only use the information of whether a link exists, while weighted indices additionally account for how strong a link is, as judged from the actual value in a network matrix.

[10] devised by Jochen Fründ

[11] Or alternatively the argument `index="ALL"`.

Table 1: Binary-quantitative pairs of network indices. Note: Most quantitative indices can also be computed on binary networks. That does not make them a quantitative measure!

	binary	quantitative
network level	connectance	H_2'
	links per species	linkage density
	nestedness	weighted NODF, wine
group level	mean number of partners	effective number of partners
species level	degree	species strength, effective number of partners

Binary indices, by definition, use less information. It does not follow that they are thus more robust! In fact, Blüthgen (2010) argues the opposite, since only a weighted index can tell whether an observed value indicates specialisation or not.[12]

This, if two indices are available to quantify the same idea, one binary and one quantitative, one should generally choose the quantitative one. Rather than species degree, we should use species strength or linkage density. Table 5.5 depicts some such binary-quantitative index pairs.

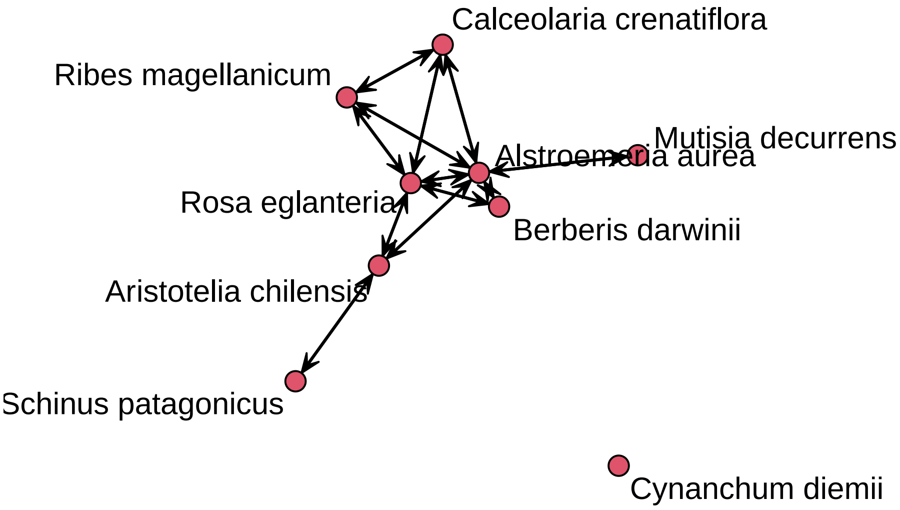
In a nutshell: Quantitative indices make use of more information in the data. They are generally more advisable, although most of them also are strongly affected by sampling issues.

5.5.1 *One index, many values: A cautionary tale told by the betweenness mess*

Betweenness is a graph index which quantifies the importance of a node for fluxes between nodes. It does so by counting how many of all shortest paths between two nodes go through the target node. That means, a peripheral node will have a betweenness score of 0, while a central node can have values up to n (if all shortest paths go through it).

```r
data(Safariland)

# plot the one-mode projection for the lower level:
set.seed(4) # don't ask me why gplot is stochastic ...
par(xpd=T, mar=c(0,6,0,6))
sna::gplot(as.one.mode(Safariland, project="lower"), label=rownames(Safariland))
```



So, what we see is for the main module a rather central position for, e.g., *Rosa eglanteria*. For 8 species (we can neglect *Cynanchum diemii*), we have $8(8-1)/2 = 28$ paths. For a simple example, *Aristotelia chilensis* (plant number 1 in the edge list below) should have only the shortest path to *Schinus patagonicus* running through it, to all other 6 plant species.

[12] Imagine two species, one with entries (1, 1), the other with entries (1, 100). A binary index would not be able to tell between them, while a quantitative would expose the second as far more specialised.

Let's see what its betweenness score is, using several different functions across R-packages:

```
# convert matrix into one-mode edgelist:
SafPlantsEL <- as.tnet(as.one.mode(Safariland, project="lower"))

## Warning in as.tnet(as.one.mode(Safariland, project = "lower")): Data assumed to be weighted
one-mode tnet (if this is not correct, specify type)
```

```
# compute betweenness:
tnet::betweenness_w(SafPlantsEL) # 6
bipartite::BC(Safariland, rescale=F)$lower # 6
DiagrammeR::get_betweenness(DiagrammeR::from_igraph(tnet_igraph(SafPlantsEL))) # 6
igraph::betweenness(tnet_igraph(SafPlantsEL), cutoff=9) # 6
influenceR::betweenness(tnet_igraph(SafPlantsEL)) # 12
sna::betweenness(SafPlantsEL) # length 36!!
sna::betweenness(as.matrix(SafPlantsEL)) # length 27!!
```

Let's put them all next to each other, into a named matrix:

```
btws <- cbind("t:betw"=tnet::betweenness_w(SafPlantsEL)[,2],
    "b:BC"=bipartite::BC(Safariland, rescale=F)$lower,
    "DR:betw"=DiagrammeR::get_betweenness(DiagrammeR::from_igraph(tnet_igraph(SafPlantsEL)))[,2],
    "i:estbetw"=igraph::betweenness(tnet_igraph(SafPlantsEL), cutoff=9),
    "inf:betw"=influenceR::betweenness(tnet_igraph(SafPlantsEL))/2 )
rownames(btws) <- rownames(Safariland)
btws
```

```
##                           t:betw b:BC DR:betw i:estbetw inf:betw
## Aristotelia chilensis          6    6       6         6        6
## Alstroemeria aurea            11   10      14        14       10
## Schinus patagonicus            0    0       0         0        0
## Berberis darwinii              0    0       0         0        0
## Rosa eglanteria               13    4       6         6        4
## Cynanchum diemii               0    0       0         0        0
## Ribes magellanicum             0    0       0         0        0
## Mutisia decurrens              0    0       0         0        0
## Calceolaria crenatiflora       0    0       9         9        0
```

So they all agree on the betweenness-value for *Aristotelia*, but not for *Alstroemeria* and *Rosa*, and not at all for *Calceolaria*.

The case of *Calceolaria* is instructive. Here we can see that actually none of the shortest paths goes through its node, making a value of 9 incorrect. (Try finding a single path that cannot be short-cutted through a link between *Alstroemeria*.)

Also *Rosa* is instructive. Here we encounter the problem of ties: several shortest paths are on equal footing, e.g. *Aristotelia* - *Rosa* - *Ribes* is as short as *Aristotelia* - *Alstroemeria* - *Ribes*. In fact, there are 4 **unique** shortest paths through *Rosa*, but 6 shortest paths in total. So it seems that **DiagrammeR** and **igraph** simply handle ties differently to **bipartite**'s BC and **influenceR**. **Tnet**'s `betweenness_w`[13] remains a mystery. Even if 13 may be the number of all paths, then that would not explain the many 0s for the other species.

Note that all functions used here have options and arguments that affect the outcome. Setting them differently will yield, for example, standardised values. We only investigated out-of-the-box behaviour and are not suggesting something is actually wrong with the functions,

[13] Available in **bipartite** under the same name.

but you may have to tweak them to yield an intuitive result. A case in point is **influenceR**'s `betweenness`, which will return **igraph**'s value if the argument `snap` is set to FALSE, but 2*BC's values for the default (TRUE).

To summarize: `bipartite::BC` and `influenceR::betweenness/2` yield values as one would expect for *unique* shortest paths; `DiagrammeR::get_betweenness` and `igraph::betweeness` yield number of non-unique shortest paths, which we also regard as an intuitive measure.

As always in data analysis: it is the responsibility of the analyst to make sure suitable-for-purpose indices are used!

5.5.2 *Index redundancy: do different indices tell the same thing?*

To visualise the similarity and possible redundancy of the different indices, they are computed for the networks in **bipartite** and summarised through a principal component analysis (see Fig. 5).

```
web.names <- data(package="bipartite")$results[,3]
data(list=web.names) #loads all webs
# the next step takes around 10 minutes:
netw.indic.webs <- t(sapply(web.names, function(x) networklevel(get(x),
              index="ALLBUTDD")))
```

```
PCA.out <- prcomp(netw.indic.webs[,-5], scale.=T)
biplot(PCA.out, xpd=T, las=1)
```

```
summary(PCA.out)

## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6    PC7
## Standard deviation     4.2780 3.6461 2.09732 1.84163 1.32612 1.09350 1.0375
## Proportion of Variance 0.3979 0.2890 0.09562 0.07373 0.03823 0.02599 0.0234
## Cumulative Proportion  0.3979 0.6869 0.78248 0.85621 0.89444 0.92044 0.9438
##                            PC8     PC9    PC10    PC11    PC12    PC13    PC14
## Standard deviation     0.81058 0.64158 0.62560 0.56960 0.49217 0.43179 0.36230
## Proportion of Variance 0.01428 0.00895 0.00851 0.00705 0.00527 0.00405 0.00285
## Cumulative Proportion  0.95812 0.96707 0.97558 0.98263 0.98789 0.99195 0.99480
##                           PC15    PC16   PC17    PC18    PC19    PC20      PC21
## Standard deviation     0.26591 0.25249 0.2254 0.18080 0.11401 0.09049 7.038e-16
## Proportion of Variance 0.00154 0.00139 0.0011 0.00071 0.00028 0.00018 0.000e+00
## Cumulative Proportion  0.99634 0.99772 0.9988 0.99954 0.99982 1.00000 1.000e+00
```

Apparently 40, 30 and 10% of the variation in indices can be partitioned to the first three axes. We can now investigate which indices load on the few axes:

```
round(PCA.out$rotation[, 1:4], 3)

##                                         PC1    PC2    PC3    PC4
## connectance                           0.177  0.147 -0.044  0.025
## web asymmetry                        -0.137 -0.098 -0.298  0.063
## links per species                     0.189 -0.087 -0.020  0.183
## number of compartments               -0.147  0.005  0.167 -0.051
## cluster coefficient                   0.169  0.128  0.005  0.119
## nestedness                           -0.020  0.196 -0.004  0.305
## weighted nestedness                   0.066 -0.108 -0.170 -0.399
## weighted NODF                         0.193  0.068 -0.138 -0.182
## interaction strength asymmetry       -0.172  0.080 -0.178 -0.028
## specialisation asymmetry              0.021  0.109  0.337 -0.096
## linkage density                       0.049 -0.258 -0.058 -0.038
## Fisher alpha                         -0.041 -0.246  0.045 -0.053
## Shannon diversity                     0.073 -0.251  0.011  0.082
## interaction evenness                  0.179 -0.078  0.012  0.102
## Alatalo interaction evenness          0.076  0.159  0.143 -0.045
## H2                                   -0.191  0.089  0.091  0.037
## number.of.species.HL                 -0.048 -0.249  0.026 -0.062
## number.of.species.LL                 -0.020 -0.256  0.121 -0.038
## mean.number.of.partners.shared.in.HL  0.191 -0.032 -0.177  0.189
## mean.number.of.partners.shared.in.LL  0.218  0.042 -0.049 -0.042
## cluster.coefficient.HL                0.189  0.093 -0.001 -0.036
## cluster.coefficient.LL                0.155  0.085 -0.285 -0.009
## weighted.cluster.coefficient.HL       0.150 -0.122 -0.189  0.162
## weighted.cluster.coefficient.LL       0.212 -0.040  0.035  0.145
## niche.overlap.HL                      0.045  0.142 -0.287 -0.272
## niche.overlap.LL                      0.202  0.035  0.062 -0.195
## togetherness.HL                       0.023  0.181 -0.283 -0.032
## togetherness.LL                       0.216  0.052  0.087 -0.041
## C.score.HL                           -0.162 -0.123  0.217  0.147
## C.score.LL                           -0.216 -0.020  0.038  0.026
## V.ratio.HL                            0.177 -0.121  0.121 -0.101
## V.ratio.LL                           -0.048 -0.238 -0.151 -0.113
## discrepancy.HL                       -0.041 -0.251  0.059 -0.042
## discrepancy.LL                       -0.039 -0.252  0.062 -0.037
## extinction.slope.HL                   0.216 -0.006  0.063  0.131
## extinction.slope.LL                   0.120 -0.061  0.035  0.326
## robustness.HL                         0.214 -0.020  0.059  0.148
## robustness.LL                         0.150 -0.084 -0.114  0.254
## functional.diversity.HL               0.162  0.003  0.027 -0.257
## functional.diversity.LL               0.163 -0.003  0.029 -0.259
## partner.diversity.HL                  0.198 -0.078  0.183 -0.028
## partner.diversity.LL                  0.061 -0.222 -0.212  0.082
## effective.partners.HL                 0.187 -0.079  0.206 -0.101
## effective.partners.LL                -0.003 -0.256 -0.125 -0.011
## generality.HL                         0.187 -0.079  0.206 -0.101
## vulnerability.LL                     -0.003 -0.256 -0.125 -0.011
```

As we can see, there are quite a few indices on the first axis with a loading of around 0.2 (links per species, weighted NODF, H2, mean number of partners shared; weighted cluster coefficient, niche overlap, togetherness, C-score (all lower level); extinction slope, robustness, partner diversity (all higher level)).

Also the PCA biplot (Fig. 5) is not very helpful, since the many indices pointing "southwards" are unreadably overlapping.

Alternatively, we might want to depict the correlation between different indices by means of a cluster analysis (Fig. 6).

```
library(Hmisc)
plot(varclus(netw.indic.webs), cex=0.8)
abline(h=0.5, lty=2, col="grey")
```

It picks out a slightly different structure, with the PC1 being distributed over more than one cluster (towards the left, with extinction.slope.HL but also towards the centre, with weighted NODF). In either case it becomes clear that many indices are of limited originality. However,

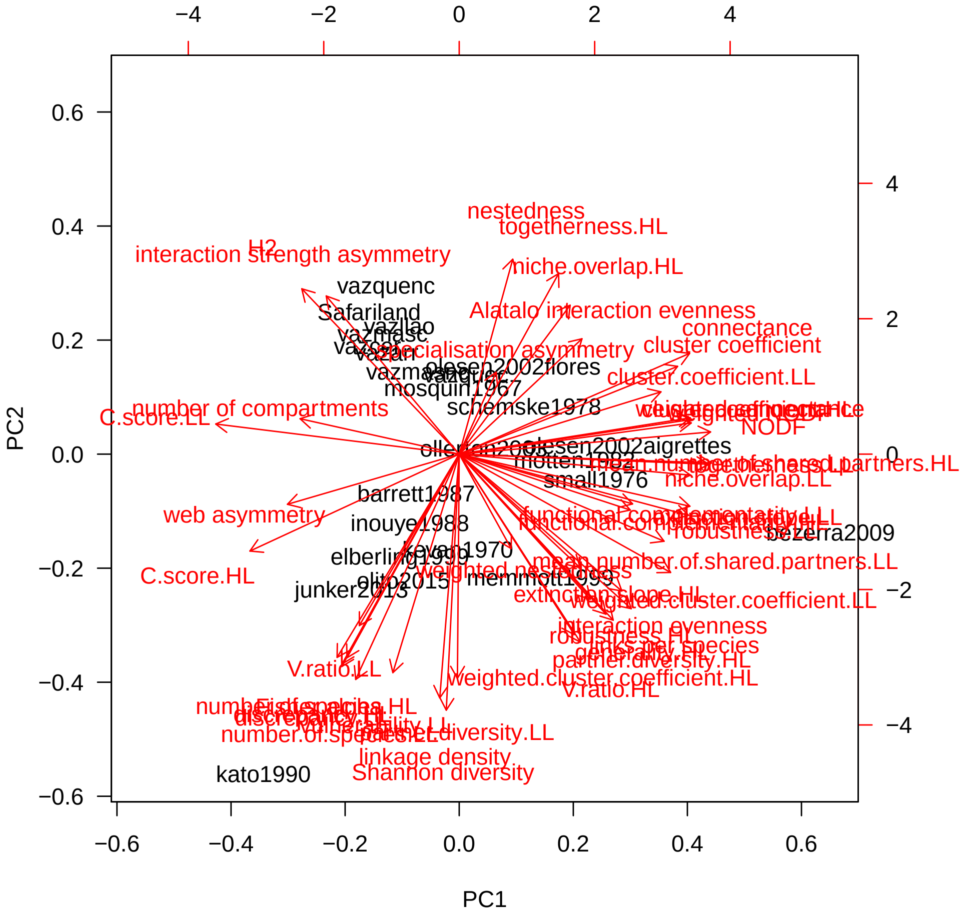


Figure 5: Principal component analysis biplot of indices at the network level, based on 22 quantitative pollination networks. Notice that network "kato1990" is by far the largest in the set, defining on its own the second principal component. Indices pointing "southwards" may thus simply be more sensitive to network size and related properties (such as density of interactions).

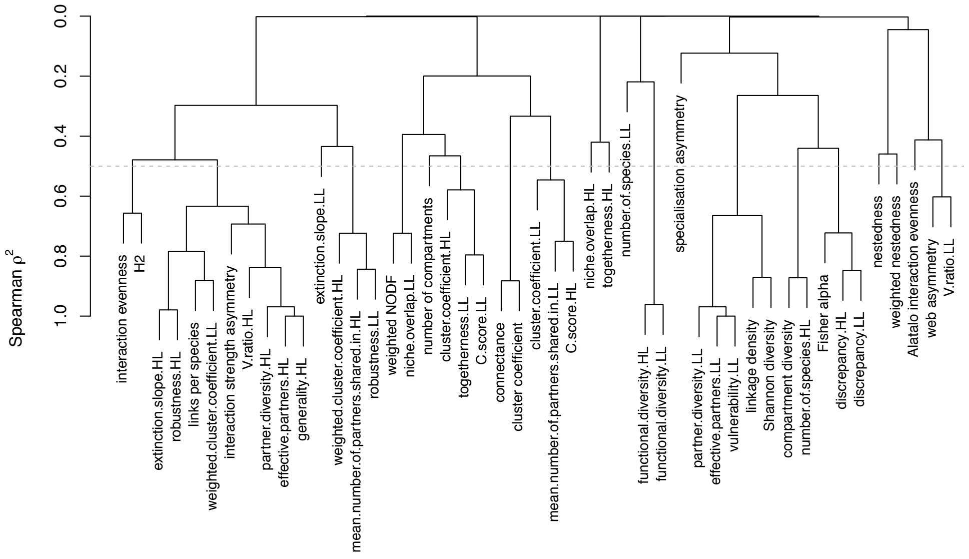


Figure 6: Cluster analysis of indices at the network level, based on 22 quantitative pollination networks. Grey dotted line indicates the level below which clustering could be considered relevant ($|r| > 0.7$).

there are some branches ending above the dashed line, indicating that they add a new element to the description of the network. Some are trivial, such as the number of species in the lower level, others less so (specialisation asymmetry, extinction slope LL, niche overlap HL, (weighted) nestedness).

It should become clear that there is no need to compute all indices a software can offer, and that some clusters seem to measure the same latent property, even it is unclear what that property is. Curiously, higher and lower level indices are always intermixed. There seems to be always an effect onto the other level, so that some indices pick up a signal in one level, but others the same signal in the other.[14]

The 22 pollination networks, on which both PCA and cluster are based, are most certainly not representative of bipartite networks in general. The simulations of Dormann *et al.* (2009) for a wider range of sizes and linkage densities show a qualitatively similar picture, though: indices are overlapping into clusters.[15]

5.5.3 *Variations on on the same theme: which index of a group of similar ones should I choose?*

Different indices were developed for different aims. Some are very reasonable, logically sound, thoughtful and even productive (in the sense of "shown to be doing what they are supposed to do"), others aren't. Each index should be understood when being used, based on the literature introducing it and any later study refining, comparing or criticising/appraising it. There is no easy way to decide the question "Which is the right index for me?"

That said, there are a few guiding considerations that seem plausible to me:

1. Quantitative indices use more information than binary ones; preferably use the former.

2. Indices developed for bipartite networks are likely to be closer, theoretically, to the structure of the data than one-mode indices working on projections of bipartite networks.

3. Indices that came out positively in comparisons have proven their worth more than fresh-off-the-press indices.

4. If two or more indices seem barely distinguishable, do **not** choose the one easiest to interpret! This would be a logical fault, since their similarity indicates clearly that they are affected by "something" and we don't know what that something is, yet. But interpreting an effect as being caused by what is "most convenient to interpret" is most certainly wrong (Shermer, 2012).[16]

The last point in particular seems rather little helpful. Imagine the two index values "functional diversity" for the higher and lower trophic level, which are near-identical in Fig. 6. This indicates that the *network*, rather than each level, contains the information which these two indices describe. In this particular case, we may want to consider using neither index, or using any of them but interpreting it *as if it represented* both *level*.

Thus, in a nutshell, there is no easy answer and there is no way around knowing the index and the literature about it and its friends.

6 *Null models*

6.1 *Why null models?*

When a network is structured according to some measure (e.g. nestedness), what does that mean? It could mean that ecological processes leading to nestedness are at play. Or it could

[14] That should be the case for extinction slopes and robustness, which measures the response of one level to extinctions in the other. But is difficult to see why the cluster coefficient for the higher level should be correlated with togetherness in the lower. *Post hoc*, it is always possible to find a reason, but *a priori* this is not expected.

[15] We could similarly say that there is a substantial redundancy in network indices, evident from the analysis where we condense 92% of variation in 40 odd indices onto six axes.

[16] Of the many, many biases that can creep into scientific explorations, this is one. We could call it the "parsimony bias" as a special case of the "selection bias" (Shermer, 2012).

mean that a sampling effect makes nestedness an inevitable consequence. Or, finally, it could be a mixture of both. Furthering our understanding of nature is not aided by the reporting of spurious relationships resulting from sampling effects, hence a formal description of what the network would look like without the ecological process would be a useful comparison. That is what a null model aims to provide (Gotelli & Graves, 1996).

In ecology, null models are common only in some fields, particularly in biogeography (Gotelli & Graves, 1996; Hausdorf & Hennig, 2007), but also in network analyses (Dormann *et al.*, 2009; Joppa *et al.*, 2009; Blüthgen *et al.*, 2008; Vázquez & Aizen, 2006, 2003; Vázquez *et al.*, 2009). With respect to particular bipartite indices, some studies rejected the current nestedness paradigm using abundance-based null models, for observed (Kallimanis *et al.*, 2009; Moore & Swihart, 2007; Santamaría & Rodríguez-Gironés, 2007), as well as for simulated networks (Krishna *et al.*, 2008).

Null models extend much beyond ecology and were probably first used in physics. Latapy *et al.* (2008) use them to investigate degree distributions in non-ecological bipartite networks. Using another network example, the module detection algorithm proposed by Newman & Girvan (2004) also uses a null model to define a "group" (or module or community) within a network: there are more links within than between groups. The null model then is a random network with as many links between as within groups (Guimerà *et al.*, 2005). In its quantitative version, the null model is constructed from proportionally probable interactions by differently abundant participants (Barber *et al.*, 2004). The specific null model used above is underlying most current implementations of contingency table tests such as χ^2 or Fisher's signed rank test (Patefield, 1981).

In conclusion, null models are established approaches in physics as well as ecology to correct for statistical artefacts and to establish an expectation in the absence of the hypothesised structuring mechanism.

6.2 *Justifying conservative null models*

The interactions observed in a network are the result of a dynamic interaction between species at two levels. If we use null models to represent the outcome of these interactions (the abundances of the participants), we seem to be over-correcting. How can we possibly find a pattern, if we use this pattern as a null model? Is an abundance-based null model not a circular argument? Here I argue when it is not, and if circularity would sneak in, why it would still be better to over-correct. I shall focus on pollination networks, and every network analyst has to decide in how far this reasoning can be applied to his/her system.

Plant persistence in a site depends to some extend on them being pollinated, and pollinator population dynamics is affected by the amount of nectar and pollen they collect from the flowers they visit. However, plant growth and population dynamics are also affected by nutrient and light availability, save sites for seed germination, herbivory and fungal pathogens, mycorrhisation and environmental disturbances; and similar constraints exist for pollinators. It is thus difficult, if not impossible, to say how much of the observed abundance of a species (plant or pollinator) is caused by the specific plant-pollinator interaction and hence the structure of the network. To the extent that abundances are determined by the pollination interaction we may indeed introduce collinearity. To date, there are no data to estimate the proportion of population growth controlled by mutualistic interactions of either plants or pollinators. We can only guess based on observations such as strong nest-site limitation in solitary bees (Steffan-Dewenter & Schiele, 2008), relatively high parasitism rates of pollinators (Klein *et al.*, 2002), strong nutrient limitations of plant population dynamics (Ghazoul, 2005) and grazing pressure (Crawley, 1987; de Mazancourt & Loreau, 2000) that they are probably not dominating population dynamics. On the other hand, clear (but rare) examples of co-evolution indicate that such a strong mutual dependence may indeed occur (Waser *et al.*, 1996; Waser & Ollerton, 2006).

Secondly, the observed interactions are not necessarily reflecting abundance of flowers or pollinators, but rather their attractiveness (e.g. due to scent, colouration or nectar production) and activity (e.g. daily period of foraging, phenology through the season, temperature-dependence of foraging activity), respectively. Hence by using observed sums of observed interactions, one does not correct for abundance, but for attractiveness and activity, respectively.

Third, the null model uses marginal totals, i.e. summed across all links of a species. If a pollinator was specialised on a single plant species, the null model would scatter these interactions across all and hence correctly identify a miss-match between observed and null model. If, on the other hand, a pollinator was generalised, then the scattering of interactions across all plant species would also yield a similar picture and generalisation would correctly be inferred.

Finally, even if the null model used would introduce some amount of circularity and thus "over-correction", we have to balance this against "over-detection". Finding specialisation in an uncorrected approach will incur high levels of type I error (i.e. detecting a pattern that is actually not there). A null model insures against this type of error. We can thus be sure that a pattern that "survives" the null model test is no statistical artefact. Since it has been shown repeatedly that sampling intensity and "natural" differences in abundance will cause such artefacts (Dormann *et al.*, 2009; Blüthgen, 2010; Joppa *et al.*, 2009; Blüthgen *et al.*, 2008), they must be addressed. Ignoring them will lead to misinterpretation and over-reporting of spurious network structure. After correction for expectation based on abundances, we can start understanding ecology (Vázquez *et al.*, 2007).

In conclusion, circularity is unlikely to substantially impair null models for pollination networks. The risk of reporting non-existent network structure and further speculation of network stability due to this spurious network structure is a problem that requires a correction, even if circularity was present.

6.3 *Using null models to test for significant tests in data*

Once you have settled on a null model, there are fundamentally two different approaches to use such null models in statistical significance testing: corrected indices and null modelling the entire analysis. Let us look at them in turn, starting with the seemingly easier option.

Corrected indices and *z*-scores Imagine we computed the index I for a network. We now want to know whether this value is anything special, given the abundances of the species.[17] So we build, say, 1000 null models[18] and compute our index I for each of them. We can visualise the distribution of null-model I-values relative to our observed index value (Dormann *et al.*, 2009).

```
data(Safariland)
Iobs <- nestednodf(Safariland)$statistic[3]
nulls <- nullmodel(web=Safariland, N=1000, method='r2d') # takes a while!
Inulls <- sapply(nulls, function(x) nestednodf(x)$statistic[3])
plot(density(Inulls), xlim=c(0, 100), lwd=2, main="NODF")
abline(v=Iobs, col="red", lwd=2)
```

All that is fine for a single network, but no option if we want to analyse dozens or hundreds of networks. We thus have to summarise this figure in a value. The simplest, most obvious but generally not best solution would be to subtract the null model mean from the observed, so that $I_{\text{corrected}} = \Delta I = I_{\text{observed}} - \bar{I}_{\text{nulls}}$. The reason why this ΔI is not ideal is that the null model distribution is likely to vary from network to network. Thus, a difference of, say, 20 may be a lot for pointy distributed null models such as the one in Fig. 7, but only little for null model values spreading over the entire x-axis.

The better, and more established approach, would be to compute z-scores (= standard scores or normal scores) for each network, which express the difference in terms of standard

[17] Thereby we justify the use of the Patefield algorithm as null model. But the approach is independent of which null model we use.

[18] E.g. using `nulls <- nullmodel(web=Safariland, N=1000, method=’r2d’)`

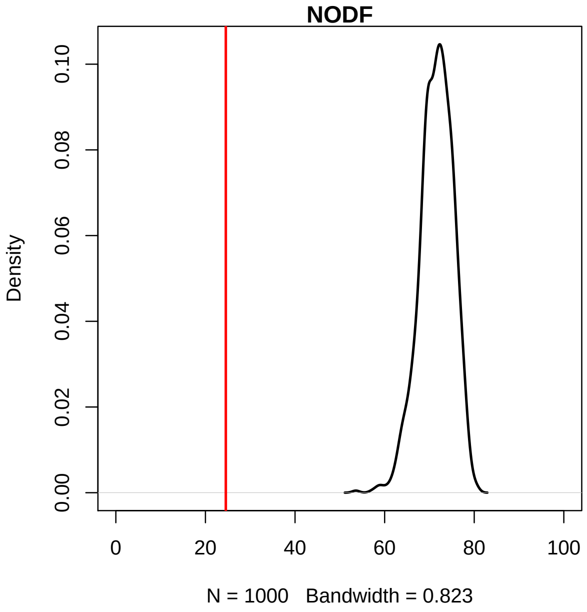


Figure 7: Observed and null-modelled nestedness (computed as NODF) for the network 'Safariland". Note that values of 100 indicate perfect nestedness. Thus, null models are much *less* nested than the observed network. (That is typically the case (Dormann et al. 2009), although often no null models are computed for comparison, making interpretation challenging.)

deviations of the null distribution:

$$z_I = \frac{I_{\text{observed}} - \bar{I}_{\text{nulls}}}{\sigma_{I_{\text{nulls}}}}.$$

We can now analyse the z-scores instead of the original I_{obs}-values.

Z-scores have one major disadvantage for several of the network indices presented earlier: the standard deviation becomes an unsuitable measure of spread as the values approach an upper or lower bound. Imagine our network index I is bound between 0 and 1 (as many are). If the null models have values above, say, 0.9, the upper bound is so close that the distribution will be skewed and the standard deviation biased too low. This may lead to inflated z-scores. So whenever you observe values close to either bound, beware of distorted z-scores.

Null-model the entire analysis The alternative approach is to carry out the intended analysis on the observed index values I and then, in a second step, repeat it with one realisation of a null model per network. Next, repeat this second step several thousand times (for good measure) and store the statistics you are interested in (say F-values in an ANOVA or slopes in a regression). Instead of using the p-values from the analysis of the observed values, compare observed and null-modelled statistics and count the proportion of values exceeding the observed and use those as p-value. To anyone with any experience in resampling analysis or bootstrapping this is standard procedure (Efron & Tibshirani, 1993; Manly, 1997).

As an example, we can use a question addressed in Vázquez & Simberloff (2003), where the authors investigate whether cattle grazing affects the network structure of pollination networks in four replicated sites in Argentina. As response we haphazardly choose linkage density.

```
weblist <- lapply(c("Safariland", "vazarr", "vazllao", "vazcer", "vazmasc",
                        "vazmasnc", "vazquec", "vazquenc"), get)
# Write a function to compute the desired statistic, e.g. the difference
# between grazed and ungrazed:
meandiff <- function(webs){
   obs <- sapply(webs, networklevel, index="linkage density")
```

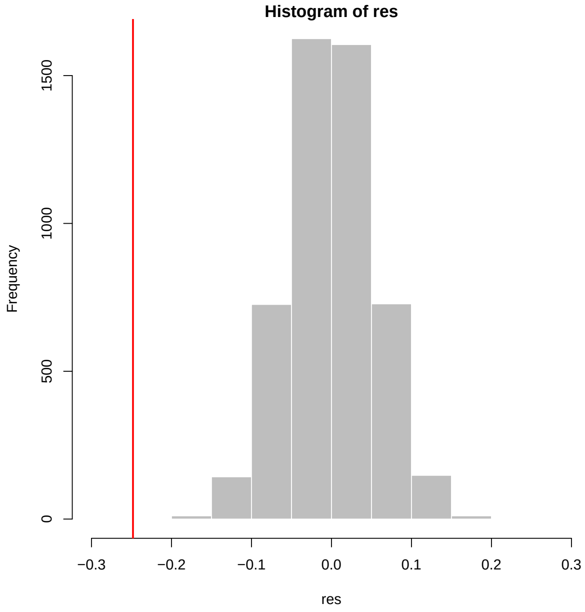


Figure 8: Distribution of mean differences in linkage density of pollination networks in grazed and ungrazed meadows, correcting for differences in species numbers and observation intensity through the use of null models ($N = 5000$). Observed difference is shown as vertical line. The observed difference is significantly ($P < 0.001$) different from null model expectations.

```
   mean(obs[1:4]) - mean(obs[5:8])
}
(observed <- meandiff(weblist))

## [1] -0.2481145
```

Now generate a null model and repeat:

```
nulllist <- lapply(weblist, nullmodel, N=1, method="r2d")
meandiff(weblist)

## [1] -0.2481145
```

Repeat this procedure, say, 5000 times:[19]

```
res <- 1:5000
for (i in 1:5000){ # takes a few minutes !!
   nulllist <- sapply(weblist, nullmodel, N=1, method="r2d")
   res[i] <- meandiff(nulllist)
}
```

Finally, we can depict the results in a histogram (Fig. 8), with observed as vertical line, and compute the *p*-value based on the 5000 null models:

```
hist(res, xlim=c(-0.3, 0.3), border="white", col="grey")
abline(v=observed, col="red", lwd=2)
# compute p-value as proportion smaller or than observed
sum(res < observed)/length(res) * 2 # *2 for two-tailed test
```

[19] For didactic purposes, we use an inelegant but easy to understand `for`-loop here. In productive mode we would instead use `sapply` or `replicate`, which would make the code more compact, but in this case not faster.

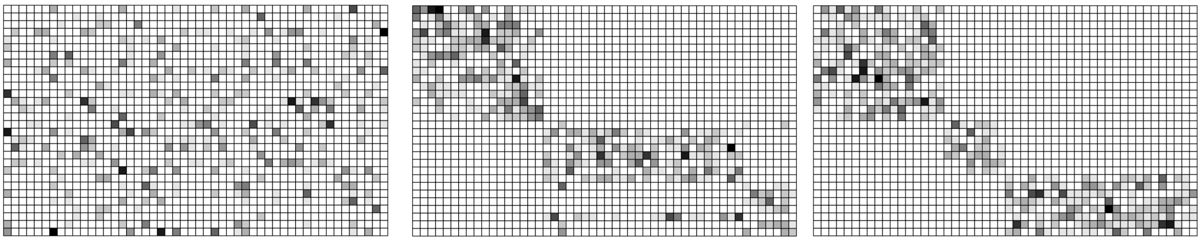


Figure 9: A simulated 3-compartment network in random sequence (left), as sorted by a correspondence analysis (centre) and by the modularity algorithm with default settings (right).

The idea here is to compute a statistic of interest, in this case the mean difference between some network index for the two treatments, and then use the null model to compute expectations. In the above example, we would apparently expect no difference (which is intuitive); additionally we also get an idea of how much variability we can expect (the distribution of the null model-based mean differences in linkage density). Thus, in this example, the difference in linkage density of pollination networks from grazed *vs* ungrazed pastures is highly significant ($p < 0.001$).

7 *Modularity*

Aggregated sets of interacting species are called 'modules'. Their defining feature is that *within*-module interactions are more prevalent than *between*-module interactions (Newman, 2003; Newman & Girvan, 2004; Fortunato, 2010). In other words, modules are link-rich clusters of species in a community. What causes such modules and whether they actually teach us something is discussed at length by Dormann *et al.* (2017).

There are several algorithms for *one-mode* networks, particularly for binary one-mode networks, such as the Louvain algorithm, the infomap or Clauset *et al.* (2004)'s "fast greedy" algorithm. These algorithms vary substantially in their robustness, in the sense that small changes in the network may yield very different modules, as compared by Policastro *et al.* (2021). That paper also expores ways to test for significance of modularity, based on null model simulations.

The package **bipartite** offers an algorithm (QuaBiMo, described in technical detail in Dormann & Strauß, 2014) to detect modules in bipartite networks, taking into account the quantitative nature of links. The algorithm has become outdated after a vastly improved approach was published by Beckett (2016), which is also now the default algorithm used in **bipartite**'s `computeModules` function.

Old-school modularity was identified using correspondence analysis, which, however, will not be able to identify modules sufficiently well, even if modules are actually compartments (i.e. perfectly separated: Fig. 9 left, centre). The algorithm used in **bipartite** can do so, at least in principle (Fig. 9 right).[20]

QuaBiMo can be invoked recursively (we are working on doing the same for Beckett), searching for modules within modules (Fig. 10, bottom). While such nested modules become ever smaller and are thus ever faster to detect, there are plenty of them and hence nesting will typically dramatically prolong computing time during the search for patterns (Memmott, 1999).

Modularity Q is likely to be correlated with other network metric, as specialisation of module members is the prime reason for the existence of modules. Across the (then) 22 quantitative pollination networks of the NCEAS "interaction webs" data base, Q was evidently highly positively correlated with complementary specialisation H_2' (Fig. 11). Ecologically, the

[20] As a note on terminology: If modules are perfectly separated, with no species interacting with species in another module, they are called "compartments" and will be visible as clearly separated blocks of interacting species. It is relatively straightforward to implement a recursive compartment detection function, but compartments are too much black-and-white compared to modules and often only occur in ecological networks because sampling was rather incomplete or very different habitats were sampled.

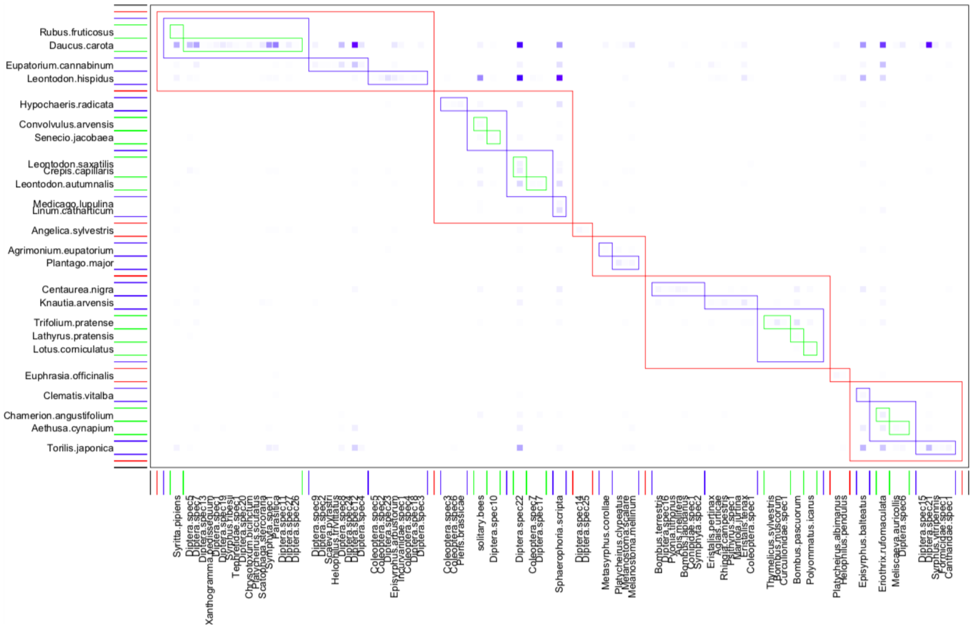


Figure 10: Interaction matrix featuring nested modules for the data of Memmott (1999). Darker squares indicate more observed interactions. Coloured boxes delineate the seven modules. (Note that results may vary between runs.) In the central module yellow Asteraceae feature heavily, while a possible ecological cause pattern for the other modules is less apparent.

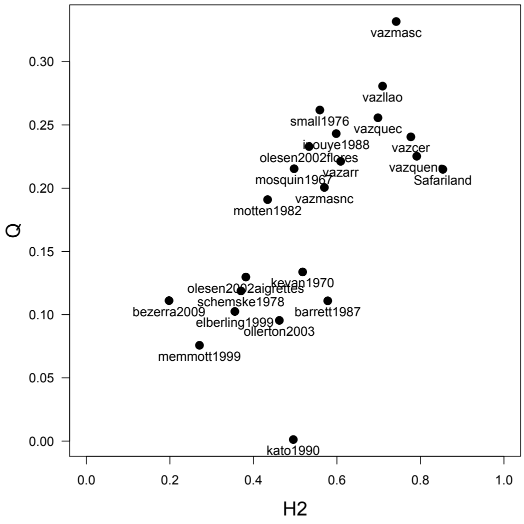


Figure 11: Modularity (Q) is highly correlated with specialisation H_2' across 22 pollination networks. Names refer to network data sets in **bipartite**.

correlation with specialisation makes good sense. Modules only exist because some species do not interact with some others, i.e. because they are specialised. An overall low degree of specialisation is equivalent to random interactions, which will yield no modules.

Furthermore, the absolute value of Q is, like all network indices (Dormann *et al.*, 2009), dependent on network size (i.e. the number of species) as well as the number of links and the total number of interactions observed (Thébault, 2013). We would thus recommend a null model comparison (Vázquez & Aizen, 2003; Blüthgen *et al.*, 2008; Dormann *et al.*, 2009) to correct the observed value of Q by null model expectation (e.g. by standardising them to z-scores: $z_Q = \frac{Q_{\text{observed}} - \overline{Q}_{\text{null}}}{\sigma_{Q_{\text{null}}}}$). Modularity Q is in itself not an index of an ecological feature. It is merely a measure of how well links and interactions can be separated into different modules. Large networks, with many species and links, allow for more combinations of species-in-modules, leading to higher values of Q, as Allesina & Pascual (2009) pointed out for any

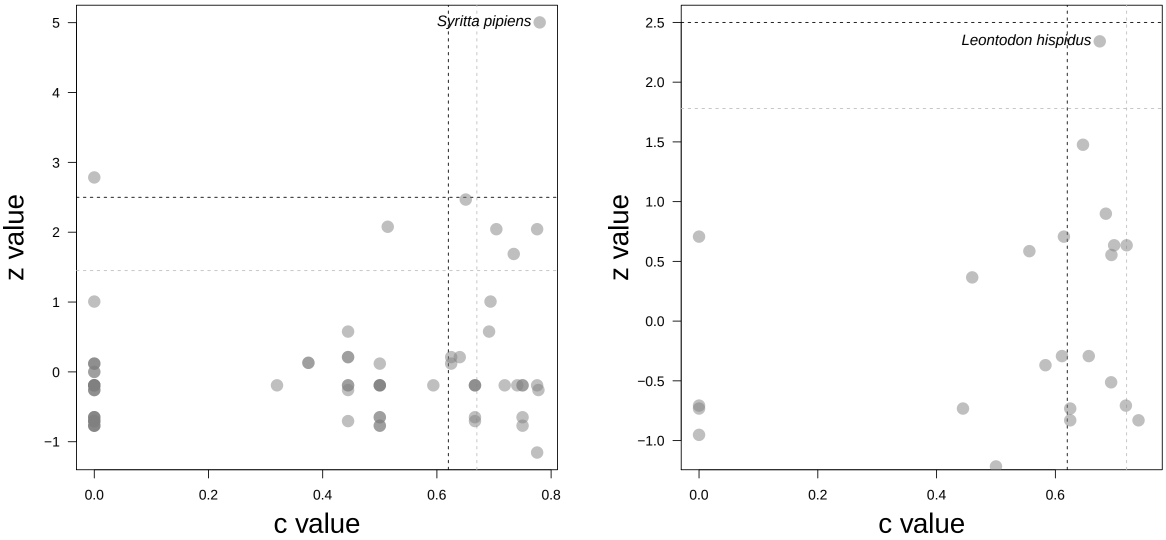


Figure 12: Connection (c) and participation (z) values for pollinators (left) and plants (right) in the network of Memmott (1999). Dashed black lines indicate critical values according to Olesen et al. (2007), those in grey 95% quantiles from 100 null models (see text).

grouping algorithm.

7.1 *Using modularity to identifying species with important roles in the network*

Guimerà *et al.* (2005) and Olesen *et al.* (2007) propose to compute standardised connection and participation values, called c and z, for each species to describe their role in networks, where c refers to the between-modules connectivity (called "participation coefficient" P: Guimerà *et al.*, 2005) and z refers to within-module degrees. Originally, both are computed based on the number of links, but a weighted version based on species strength is implemented, too. Guimerà *et al.* (2005) suggest critical values (only for the binary approach) of c and z of 0.625 and 2.5, respectively. Species exceeding both of these values are called "hubs" because they link different modules, combining high between- with high within-module connectivity.

In the case of the pollination network of Fig. 10, c-values range between 0 and 0.78 (with 23 of 79 pollinators and 13 of 25 plant species exceeding the threshold of 0.625); z-values range between -1.21 and 5.00 (with two pollinators but no plant species exceeding the value of 2.5: Fig. 12). Put together, only the syrphid *Syritta pipiens* (and hawkbit *Leontodon hispidus* almost) exceeded both thresholds and would thus be called a "hub species". As can be seen in Fig. 10, this syrphid is relatively rare but clearly not randomly distributed over the six modules, thus linking three highest-level modules. In contrast, *Leontodon hispidus* is a common plant species, visited by many different pollinators, and it actually links almost all modules.

To objectively define this threshold one could run null models of the original network and employ 95% quantiles as critical c- and z-values. For the pollinators in the network of Fig. 10 these would be 0.67 ($\pm$0.039) and 1.45 ($\pm$0.220), respectively, based on 100 null models (for plants: $c_{\text{critical}} = 0.72 \pm 0.036$ and $z_{\text{critical}} = 1.78 \pm 0.297$; Fig. 12 left). While for plant species this has little effect (except for moving *Leontodon hispidus* across the threshold), three more pollinators would become hub species (the common hoverfly *Episyrphus balteatus*, the tachinid fly *Eriothrix rufomaculata* and undetermined fly "Diptera spec.22").

8 *Analyses beyond* ***bipartite***

Of course, there are many different packages that analyse one or the other aspect of networks, and some even of bipartite networks (listed in the introduction of this vignette). Some of them we find particularly interesting or useful or innovative, and here we quickly show some of them at work.

8.1 *Plotting pretty bipartites*

The plotting functions provided by **bipartite** are somewhat rudimentary. We're happy with them, and they are surprisingly flexible already, but our graphical skills (and intentions) are limited. Other people have developed plotting functions/solutions, which we shall briefly (and probably incompletely) show here.[21]

8.1.1 *bipartiteD3*

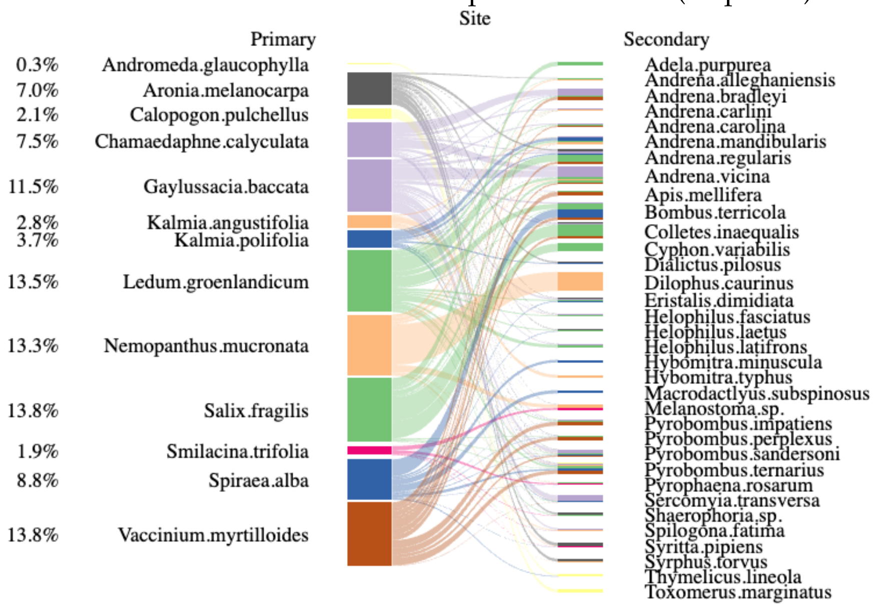
This package allows for very customised plotting. The coolest thing about **bipartiteD3** is that it can be used interactively. By moving the mouse over the plot, the link colours change and so forth.

However, it is a bit awkward as it produces a plot in the viewer (or wherever), which then has to be saved and linked back into the document. Quite often, the plot is then incomplete, as in the example below (where the percentages are cut off despite being visible in the export-window of RStudio).

Here is a very short and simple default plot:

```
library(bipartiteD3)
bipartite_D3(small1976, colouroption = 'brewer', ColourBy = 1,
             PercentageDecimals=1, filename = 'small1976Plot')
```

After saving the resulting image, it looks like this, wth the percentages representing the proportion of links in the network of the particular node (= species).



The main interest of D3 is the interactiveness, not publication-ready graphics. For this aim, it is a very nice tool and worth exploring.

8.1.2 *igraph*

https://kateto.net/netscix2016.html; https://rpubs.com/pjmurphy/317838; https://igraph.org/r/doc/layout_as_bipartite.html

[21] Partly based on this question on SO: https://stackoverflow.com/questions/31366066/how-to-plot-a-bipartite-graph-in-r

8.1.3 *ggbipart*

There is the package **ggbipart**, on github.[22]

```
library(devtools)
install_github(rep="pedroj/bipartite_plots")
```

8.1.4 *multigraph*

Plot several bipartite layers into one graph: see `https://stackoverflow.com/questions/31366066/how-to-plot-a-bipartite-graph-in-r`

8.2 *Analysing the structure of bipartite networks*

There are a range of packages that aim at identifying structure within networks, be it clusters/modules/compartments of a larger subset of interactions, or specific motifs of only few links. It is almost impossible to keep up with these developments, so here are just a few pointers to such packages:

bmotif A "motif" is a pattern in a network, e.g. the v-shaped link of one plant to two pollinators. This packages (Simmons *et al.*, 2019) counts how often each of 44 such motifs is encountered in a bipartite network. This frequency distribution can then be compared to those of a null model. The majority of these motifs have no obvious ecological interpretation. For example, motif 5 is an N-shape, connecting in that N-way 2 lower with 2 higher nodes. Meaning what exactly? The brief hype following the release of this package and its main publications (Monteiro & Faria, 2016; Simmons *et al.*, 2019; Jácome-Flores *et al.*, 2020) has died down a bit, and the packages has been archived on CRAN in 2022 due to unresolved code issues. Its github version can be found here: `https://github.com/SimmonsBI/bmotif`.

dynsbm

infomapecology

8.3 *Multi-layer networks*

This somewhat cryptic term refers to networks that are defined by different types of links, for example because links change over time (dynamic links) or because they depend on other networks (Pilosof *et al.*, 2017). The mathematical description and analysis of such multilayer networks can be rather complex (Aleta & Moreno, 2019), and in ecology typically only relatively simple types of multilayer networks are investigated; probably the most common being multi-level networks are temporally or spatially resolved networks.

While statistically and mathematically a lot of things are conceivable, few ideas actually resonate with ecologists (i.e. in application). One idea is that networks evolve and hence change dynamically. Another is that any observed bipartite network is actually embedded in many other networks, e.g. a plant-pollinator-network may continue "into the ground" as plant-mycorrhiza networks, "into the sky" as pollinator-parasite network, and "sideways" as plant-herbivore network (Pocock *et al.*, 2012).

Currently, **bipartite** does not support any such analysis (apart from a pathetic function `npartite` and `plotweb2` for plotting stacked bipartite networks).

Packages such as **robber** address some of the challenges of intersecting multiple bipartite networks.

[22] See also `https://pedroj.github.io/bipartite_plots/` for more code examples.

Conclusions

The analysis of bipartite networks has seen a recent explosion of applications in ecology. The *R*-package **bipartite** aims at providing a user-friendly set of function to aid the visualisation and analysis of such networks. This vignette hopefully illustrated some of the challenges and solutions to such analyses. The main take-home-message it aims to convey is that there are considerable interpretational strings attached, and that the analysis of bipartite networks is not as straightforward as one might have initially hoped.

Feedback, improvements, corrections and additions are welcomed!

The Rnw source of this document is at `https://github.com/biometry/bipartite/bipartite/vignettes/Intro2bipartite.Rnw`.

Bibliography

Alarcón, R., Waser, N.M. & Ollerton, J. (2008) Year-to-year variation in the topology of a plant-pollinator interaction network. *Oikos*, **117**, 1796–1807. 10.1111/j.0030-1299.2008.16987.x. 14

Alatalo, R.V. (1981) Problems in the measurement of evenness in ecology. *Oikos*, **37**, 199–204. 12

Aleta, A. & Moreno, Y. (2019) Multilayer networks in a nutshell. *Annual Review of Condensed Matter Physics*, **10**, 45–62. ArXiv: 1804.03488. 30

Allesina, S. & Pascual, M. (2009) Food web models: a plea for groups. *Ecology Letters*, **12**, 652–62. 27

Almeida-Neto, M., Guimarães, P., Guimarães Jr, P.R., Loyola, R.D. & Ulrich, W. (2008) A consistent metric for nestedness analysis in ecological systems: reconciling concept and measurement. *Oikos*, **117**, 1227–1239. 11

Almeida-Neto, M. & Ulrich, W. (2011) A straightforward computational approach for measuring nestedness using quantitative matrices. *Environmental Modelling and Software*, **26**, 173–178. 11

Barber, K.R., Leeds-Harrison, P.B., Lawson, C.S. & Gowing, D.J.G. (2004) Soil aeration status in a lowland wet grassland. *Hydrological Processes*, **18**, 329–341. 22

Barrat, A., Barthélemy, M., Pastor-Satorras, R. & Vespignani, A. (2004) The architecture of complex weighted networks. *Proceedings of the National Academy of Sciences of the USA*, **101**, 3747–52. 14

Bascompte, J., Jordano, P., Melián, C.J. & Olesen, J.M. (2003) The nested assembly of plant–animal mutualistic networks. *Proceedings of the National Academy of Sciences of the USA*, **100**, 9383–9387. 12

Bascompte, J., Jordano, P. & Olesen, J.M. (2006) Asymmetric coevolutionary networks facilitate biodiversity maintenance. *Science*, **312**, 431–433. 13, 14

Beckett, S.J. (2016) Improved community detection in weighted bipartite networks. *Royal Society open science*, **3**, 140536–140536. 26

Bersier, L.F., Banasek-Richter, C. & Cattin, M.F. (2002) Quantitative descriptors of food-web matrices. *Ecology*, **83**, 2394–2407. 12, 13, 15

Blüthgen, N. (2010) Why network analysis is often disconnected from community ecology: A critique and an ecologist's guide. *Basic and Applied Ecology*, **11**, 185–195. 16, 23

Blüthgen, N., Fründ, J., Vázquez, D.P. & Menzel, F. (2008) What do interaction network metrics tell us about specialization and biological traits? *Ecology*, **89**, 3387–99. 22, 23, 27

Blüthgen, N., Menzel, F. & Blüthgen, N. (2006) Measuring specialization in species interaction networks. *BMC Ecology*, **6**, 9. 12, 15

Blüthgen, N., Menzel, F., Hovestadt, T., Fiala, B. & Blüthgen, N. (2007) Specialization, constraints, and conflicting interests in mutualistic networks. *Current Biology*, **17**, 341–346. 11, 12

Borgatti, S.P. & Everett, M.G. (1997) Network analysis of 2-mode data. *Social Networks*, **19**, 243–269. 14

Brualdi, R.A. & Sanderson, J.G. (1999) Nested species subsets, gaps, and discrepancy. *Oecologia*, **119**, 256–264. 13

Burgos, E., Ceva, H., Perazzo, R.P.J., Devoto, M., Medan, D., Zimmermann, M. & Delbue, A.M. (2007) Why nestedness in mutualistic networks? *Journal of Theoretical Biology*, **249**, 307–313. 13

Butts, C.T. (2013) *sna: Tools for Social Network Analysis*. R package version 2.2-2. 9

Clauset, A., Newman, M. & Moore, C. (2004) Finding community structure in very large networks. *Physical Review E*, **70**, 066111–066111. 26

Crawley, M.J. (1987) *Herbivory: The Dynamics of Animal Plant Interactions*. University of California Press, San Diego, USA. 22

Dalsgaard, B., Martín González, A.M., Olesen, J.M., Timmermann, A., Andersen, L.H. & Ollerton, J. (2008) Pollination networks and functional specialization: a test using lesser antillean plant-hummingbird assemblages. *Oikos*, **117**, 789–793. 14

de Mazancourt, C. & Loreau, M. (2000) Effect of herbivory and plant species replacement on primary production. *The American Naturalist*, **155**, 735–754. 22

Devoto, M., Bailey, S., Craze, P. & Memmott, J. (2012) Understanding and planning ecological restoration of plant-pollinator networks. *Ecology Letters*, **15**, 319–328. 13

Dormann, C.F. (2011) How to be a specialist? quantifying specialisation in pollination networks. *Network Biology*, **1**, 1–20. 3, 15

Dormann, C.F. (2019) Bipartite vignette. *github*. 3

Dormann, C.F. & Blüthgen, N. (2017) *Food webs versus interaction networks: principles, pitfalls and perspectives*, pp. 1–11. Cambridge University Press. 3

Dormann, C.F., Blüthgen, N., Fründ, J. & Gruber, B. (2009) Indices, graphs and null models: Analyzing bipartite ecological networks. *The Open Ecology Journal*, **2**, 7–24. 3, 13, 21, 22, 23, 27

Dormann, C.F., Fründ, J. & Gruber, B. (2008) Introducing the bipartite package: Analysing ecological networks. *R News*, **8**, 8–11. 3

Dormann, C.F., Fründ, J. & Schaefer, H.M. (2017) Identifying causes of patterns in ecological networks: Opportunities and limitations. *Annual Reviews in Ecology, Evolution, and Systematics*, **48**, 559–84. 3, 26

Dormann, C.F. & Strauß, R. (2014) A method for detecting modules in quantitative bipartite networks. *Methods in Ecology and Evolution*, **5**, 90–98. 3, 26

Dunne, J.A., Williams, R.J. & Martinez, N.D. (2002) Food-web structure and network theory: The role of connectance and size. *Proceedings of the National Academy of Sciences of the USA*, **99**, 12917–22. 11

Efron, B. & Tibshirani, R.J. (1993) *An Introduction to the Bootstrap*. Chapman & Hall, London. 24

Farage, C., Edler, D., Eklöf, A., Rosvall, M. & Pilosof, S. (2021) Identifying flow modules in ecological networks using Infomap. *Methods in Ecology and Evolution*, **12**, 778–786. 5

Feinsinger, P., Spears, E.E. & Poole, R.W. (1981) A simple measure of niche breadth. *Ecology*, **62**, 27–32. 15

Fisher, R., Corbet, A. & Williams, C. (1943) The relation between the number of species and the number of individuals in a random sample of animal population. 12, 42–58. *Journal of Animal Ecology*, **12**, 42–58. 12, 14

Fortunato, S. (2010) Community detection in graphs. *Physics Reports*, **486**, 75–174. 26

Galeano, J., Pastor, J.M. & Iriondo, J.M. (2009) Weighted-interaction nestedness estimator (wine): A new estimator to calculate over frequency matrices. *Environmental Modelling and Software*, **24**, 1342–1346. 11

Ghazoul, J. (2005) Buzziness as usual? questioning the global pollination crisis. *Trends in Ecology and Evolution*, **20**, 367–73. 22

Gotelli, N.J. & Graves, G.R. (1996) *Null Models in Ecology*. Smithsonian Institution Press. 11, 22

Guimerà, R., Mossa, S., Turtschi, A. & Amaral, L.A.N. (2005) The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles. *Proceedings of the National Academy of Sciences of the United States of America*, **102**, 7794–9. 22, 28

Hausdorf, B. & Hennig, C. (2007) Null model tests of clustering of species, negative co-occurrence patterns and nestedness in meta-communities. *Oikos*, **116**, 818–828. 22

Hurlbert, S.H. (1978) Measurement of niche overlap and some relatives. *Ecology*, **59**, 67–77. 13

Joppa, L.N., Bascompte, J., Montoya, J.M., Solé, R.V., Sanderson, J. & Pimm, S.L. (2009) Reciprocal specialization in ecological networks. *Ecology Letters*, **12**, 961–969. 22, 23

Jordano, P., Bascompte, J. & Olesen, J.M. (2003) Invariant properties in coevolutionary networks of plant-animal interactions. *Ecology Letters*, **6**, 69–81. 13

Jost, L. (2006) Entropy and diversity. *Oikos*, **113**, 363–375. 13

Julliard, R., Clavel, J., Devictor, V., Jiguet, F. & Couvet, D. (2006) Spatial segregation of specialists and generalists in bird communities. *Ecology Letters*, **9**, 1237–1244. 14

Jácome-Flores, M.E., Jordano, P., Delibes, M. & Fedriani, J.M. (2020) Interaction motifs variability in a Mediterranean palm under environmental disturbances: the mutualism–antagonism continuum. *Oikos*, **129**, 367–379. 30

Kallimanis, A., Petanidou, T., Tzanopoulos, J., Pantis, J. & Sgardelis, S. (2009) Do plant–pollinator interaction networks result from stochastic processes? *Ecological Modelling*, **220**, 684–693. 22

Klein, A.M., Steffan-Dewenter, I. & Tscharntke, T. (2002) Predator–prey ratios on cocoa along a land-use gradient in indonesia. *Biodiversity and Conservation*, **11**, 683–693. 22

Krishna, A., Guimarães Jr, P.R., Jordano, P. & Bascompte, J. (2008) A neutral-niche theory of nestedness in mutualistic networks. *Oikos*, **117**, 1609–1618. 22

Latapy, M., Magnien, C. & Vecchio, N.D. (2008) Basic notions for the analysis of large two-mode networks. *Social Networks*, **30**, 31–48. 22

Manly, B.F. (1997) *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Chapman & Hall/CRC, New York, 2nd edition. 24

Martín González, A.M., Dalsgaard, B. & Olesen, J.M. (2010) Centrality measures and the importance of generalist species in pollination networks. *Ecological Complexity*, **7**, 36–47. 14

Martín González, A.M., Vázquez, D.P., Ramos-Jiliberto, R., Lee, S.H. & Miele, V. (2020) Core-periphery structure in mutualistic networks: an epitaph for nestedness? *bioRxiv*, p. 2020.04.02.021691. 4

Memmott, J. (1999) The structure of a plant-pollinator food web. *Ecology Letters*, **2**, 276–280. 26

Memmott, J., Waser, N.M. & Price, M.V. (2004) Tolerance of pollination networks to species extinctions. *Proceedings of the Royal Society London B*, **271**, 2605–11. 13

Miele, V. & Matias, C. (2017) Revealing the hidden structure of dynamic ecological networks. *Royal Society Open Science*, **4**, 170251. 4

Monteiro, A.B. & Faria, L.D.B. (2016) The interplay between population stability and food-web topology predicts the occurrence of motifs in complex food-webs. *Journal of Theoretical Biology*, **409**, 165–171. 30

Moore, J.E. & Swihart, R.K. (2007) Toward ecologically explicit null models of nestedness. *Oecologia*, **152**, 763–77. 22

Morris, R.J., Lewis, O.T. & Godfray, H.C.J. (2005) Apparent competition and insect community structure: towards a spatial perspective. *Annales Zoologici Fennici*, **42**, 449–462. 9

Motten, A.F. (1982) *Pollination Ecology of the Spring Wildflower Community in the Deciduous Forests of Piedmont North Carolina*. PhD thesis, Duke University. 7, 9

Müller, C.B., Adriaanse, I.C.T., Belshaw, R. & Godfray, H.C.J. (1999) The structure of an aphid–parasitoid community. *Journal of Animal Ecology*, **68**, 346–370. 12

Newman, M.E.J. (2003) The structure and function of complex networks. *SIAM Review*, **45**, 167–256. 26

Newman, M.E.J. & Girvan, M. (2004) Finding and evaluating community structure in networks. *Physical Review E*, **69**, 1–15. 22, 26

Ohlmann, M., Miele, V., Dray, S., Chalmandrier, L., O'Connor, L. & Thuiller, W. (2019) Diversity indices for ecological networks: a unifying framework using hill numbers. *Ecology Letters*, **22**, 737–747. 4

Olesen, J.M., Bascompte, J., Dupont, Y.L. & Jordano, P. (2007) The modularity of pollination networks. *Proceedings of the National Academy of Sciences of the USA*, **104**, 19891–19896. 28

Opsahl, T. (2010) Triadic closure in two-mode networks: redefining the global and local cluster coefficient. *arXiv*, p. 1006.0887v2. 12

Patefield, W.M. (1981) Algorithm as159. an efficient method of generating r x c tables with given row and column totals. *Applied Statistics*, **30**, 91–97. 22

Pielou, E.C. (1972) 2^k contingency tables in ecology. *Journal of Theoretical Biology*, **34**, 337–352. 13

Pilosof, S., Porter, M.A., Pascual, M. & Kéfi, S. (2017) The multilayer nature of ecological networks. *Nature Ecology & Evolution*, **1**, 0101–0101. 30

Pocock, M.J.O., Evans, D.M. & Memmott, J. (2012) The robustness and restoration of a network of ecological networks. *Science*, **335**, 973–977. 30

Poisot, T., Bever, J.D., Nemri, A., Thrall, P.H. & Hochberg, M.E. (2011a) A conceptual framework for the evolution of ecological specialisation. *Ecology Letters*, **14**, 841–851. 14

Poisot, T., Canard, E., Mouquet, N. & Hochberg, M.E. (2012) A comparative study of ecological specialization estimators. *Methods in Ecology and Evolution*, **3**, 537–544. 14, 15

Poisot, T., Lepennetier, G., Martinez, E., Ramsayer, J. & Hochberg, M.E. (2011b) Resource availability affects the structure of a natural bacteria-bacteriophage community. *Biology Letters*, **7**, 201–4. 14

Policastro, V., Righelli, D., Carissimo, A., Cutillo, L. & De Feis, I. (2021) ROBustness In Network (robin): an R package for comparison and validation of communities. *The R Journal*, **13**, 292–309. 26

Roberts, A. & Stone, L. (1990) Island-sharing by archipelago species. *Oecologia*, **83**, 560–567. 12

Rodríguez-Gironés, M.A. & Santamaría, L. (2006) A new algorithm to calculate the nestedness temperature of presence-absence matrices. *Journal of Biogeography*, **33**, 924–935. 11

Santamaría, L. & Rodríguez-Gironés, M.A. (2007) Linkage rules for plant–pollinator networks: trait complementarity or exploitation barriers ? *PLoS Biology*, **5**, e31. 22

Schluter, D. (1984) A variance test for detecting species associations, with some example applications. *Ecology*, **65**, 998–1005. 13

Schoener, T. (1989) Food webs from the small to the large. *Ecology*, **70**, 1559–1589. 14

Shermer, M. (2012) *The Believing Brain: From Ghosts and Gods to Politics and Conspiracies – How We Construct Beliefs and Reinforce Them as Truths*. St. Martin's Griffin, New York. 21

Simmons, B.I., Sweering, M.J.M., Schillinger, M., Dicks, L.V., Sutherland, W.J. & Di Clemente, R. (2019) bmotif: A package for motif analyses of bipartite networks. *Methods in Ecology and Evolution*, **0**. 30

Steffan-Dewenter, I. & Schiele, S. (2008) Do resources or natural enemies drive bee population dynamics in fragmented habitats? *Ecology*, **89**, 1375–1387. 22

Stone, L. & Roberts, A. (1990) The checkerboard score and species distributions. *Oecologia*, **85**, 74–79. 13

Stone, L. & Roberts, A. (1992) Competitive exclusion, or species aggregation? an aid in deciding. *Oecologia*, **91**, 419–424. 12, 13

Thébault, E. (2013) Identifying compartments in presence-absence matrices and bipartite networks: insights into modularity measures. *Journal of Biogeography*, **40**, 759–768. 27

Tylianakis, J.M., Tscharntke, T. & Lewis, O.T. (2007) Habitat modification alters the structure of tropical host-parasitoid food webs. *Nature*, **445**, 202–5. 11, 12

Vázquez, D.P. & Aizen, M.A. (2003) Null model analyses of specialization in plant–pollinator interactions. *Ecology*, **84**, 2493–2501. 22, 27

Vázquez, D.P. & Aizen, M.A. (2006) Community-wide patterns of specialization in plant-pollinator interactions revealed by null models. N.M. Waser & J. Ollerton, eds., *Plant-Pollinator Interactions: From Specialization to Generalization*, pp. 200–219. University of Chicago Press, Chicago. 22

Vázquez, D.P., Blüthgen, N., Cagnolo, L. & Chacoff, N.P. (2009) Uniting pattern and process in plant-animal mutualistic networks: a review. *Annals of Botany*, **103**, 1445–57. 22

Vázquez, D.P., Melián, C.J., Williams, N.M., Blüthgen, N., Krasnov, B.R. & Poulin, R. (2007) Species abundance and asymmetric interaction strength in ecological networks. *Oikos*, **116**, 1120–1127. 14, 23

Vázquez, D.P. & Simberloff, D. (2003) Changes in interaction biodiversity induced by an introduced ungulate. *Ecology Letters*, **6**, 1077–1083. 24

Waser, N.M., Chittka, L., Price, M.V., Williams, N.M. & Ollerton, J. (1996) Generalization in pollination systems, and why it matters. *Ecology*, **77**, 1043–1060. 22

Waser, N.M. & Ollerton, J. (2006) *Plant-Pollinator Interactions: From Specialization to Generalization*. University of Chicago Press, Chicago. 22

Watts, D.J. & Strogatz, S.H. (1998) Collective dynamics of ’small-world’ networks. *Nature*, **393**, 440–442. 12

Index