BipartGraph: An interactive application to plot bipartite ecological networks

J. Garcia-Algarra[a,b,*], J. M. Pastor[b,c], M. L. Mouronte[d], J. Galeano[b,c]

[a] *ICADE, Universidad Pontificia Comillas, Madrid, Spain*
[b] *Complex Systems Group, Universidad Politecnica de Madrid, Spain*
[c] *ETSIAAB, Dept. Ingenieria Agroforestal, Universidad Politecnica de Madrid, Spain*
[d] *Computer Science Department, Universidad Francisco de Vitoria, Madrid, Spain*

Abstract

Biotic interactions among two different guilds of species are very common in nature and are modelled as bipartite networks. The usual ways to visualize them, the bipartite graph and the interaction matrix, become rather confusing when working with real ecological networks. We have developed two new types of visualization, using the observed structural properties of these networks. A technique called *k-core decomposition* identifies groups of species by their connectivity. With the results of this analysis we build a plot based on information reduction (Polar Plot) and another which takes the groups as elementary blocks for spatial distribution (Ziggurat plot). We describe the basic `kcorebip` package for analysis and static plotting and *BipartGraph*, the interactive tool to explore and visualize networks.

Key words: Ecological bipartite networks, visualization, *k-core* decomposition

Software Availability

Name of software: BipartGraph Programming language: R Operating system: Windows, Linux and MacOS Availability: Source code and documentation can be accessed at https://github.com/jgalgarra/bipartgraph User interface: Web browser License: Free, under MIT License

*Corresponding author
Email addresses: `fjgalgarra@comillas.edu` (J. Garcia-Algarra)

1. Introduction

Network Science is a powerful tool for ecology. Species and their interactions, define the global architecture of communities [1]. Interactions like predation or parasitism are positive for one species and negative for another. Commensalism is positive for one partner and neutral for another while mutualism is benefitial for both sides [2].

The network model of an ecological community represents species as nodes and interactions as links. In mutualism, species fall into two disjoint guilds, such as plants and pollinators or hosts and parasites. Ties among species of the same guild are forbidden. This configuration defines a bipartite network, a common structure in other walks of network science.

Visualization plays an important role in network analysis. A scientific visualization is a graphical interface between data and people [3, 4, 5]. For instance, a field ecologist could identify central species and those most endangered with a good network plot. The range of possible applications is wide [6], but tools designed for ecological networks are scarce [7, 8, 9, 10].

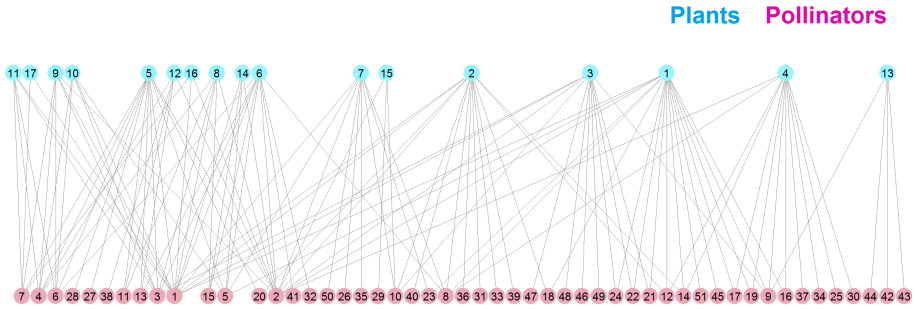

Figure 1: Plant-pollinator community in Tenerife, Canary Islands (Spain), with 68 species and 129 links [11].

Pervasive plots in literature are the bipartite graph and the interaction matrix. In the former, nodes are plotted along two parallel lines. Interactions appear as links among them (fig. 1). It is quite simple, as it makes clear the separation of guilds. It is not easy to follow indirect interactions. If pollinator

A feeds on plant C and plant D, the fate of these species is indirectly tied by the shared pollinator through positive feedback.

When the number of nodes of a bipartite plot is in the range of a few tens it becomes extremely confusing. It is hard to distinguish individual links and impossible to follow indirect interactions. The accumulation of links in the space between guilds creates what is known as the *hairball effect*.

The second common type of bipartite visualization is the interaction matrix. Species of one guild are arranged along rows, and species of the opposite guild along columns. A filled cell marks the interaction between two species.

Mutualistic networks show a special configuration. There is one group of very interconnected nodes, the `generalists`, that provide stability and resilience [12]. Species with a low number of links (`degree`) are tied to those specialists. This property is called nestedness and there are different indexes to measure the strength of this core-periphery organization. Modularity accounts for the existence of small groups of nodes of high degree inside a network sparsely connected [13]. With the interaction matrix it is easy to visually discover patterns of nestedness and modularity. On the other hand, indirect interactions are less apparent than in the bipartite plot and the matrix also becomes difficult to interpret when the number of species grows.

To overcome these limitations, there are two possible attack strategies: information reduction or taking advantage of known network traits to order nodes and links in space. In this paper, we explain how structural properties of bipartite ecological networks are the basis of two new types of visualization. We also describe an interactive application to plot them. Both rely on a classical technique called *k-core decomposition*.

2. Plots

The rationale behind this research is that, as ecological networks are not random, this fact should provide a natural way to group species using their connectivity properties. The *k-core decomposition* is a fast and efficient way to

identify those groups [14, 15].

The *k-core* of a graph G is a maximal connected subgraph of degree k. Each node of the core of order m (called *m-shell*) has links with at least m other nodes that belong to that same core. The practical implication of this definition is that species are classified according to their connectivity. The innermost shell is the set of highest k index nodes. Species with higher degrees are the *generalists*. As k index decreases species become more specialist. We refer to [16] for further details on the `k-core` analysis.

As a result of the analysis we define two magnitudes. The first one is k_{radius}. The $k^A_{radius}(m)$ of node m of guild A is the average distance to all species of the innermost shell of guild B (set C^B).

$$k^A_{radius}(m) = \frac{1}{\mid C^B \mid} \sum_{j \in C^B} dist_{mj} \qquad m \in A \tag{1}$$

where $dist_{mj}$ is the shortest path from species m of guild A to species j of guild B. In an intuitive way, k_{radius} measures how far the species is from the most connected shell, the group that is the corner stone of the community.

The second magnitude is k_{degree}. It is defined as the sum of the inverses of $k_{kradius}$ of neighbor species:

$$k^A_{degree}(m) = \sum_j \frac{a_{mj}}{k^B_{radius}(j)} \quad m \in A, \forall j \in B \tag{2}$$

where a_{mj} is the element of the interaction matrix that represents the link, considered as binary (1 if $a_{mj} > 0$, 0 if $a_{mj} = 0$). k_{degree} is a measure of centrality.

2.1. The Polar plot

The *k-core* decomposition helps to understand the structure and to visualize very large systems and networks [17, 18]. In particular, the *fingerprint plot* [19], uses a polar coordinate system. Nodes are depicted at a distance proportional to the shell they belong to and with areas proportional to their `degree`.

Taking this idea as the starting point, we build the Polar Plot. Differences are noteworthy. The first one is the bipartite nature of the networks, so space is divided in two half planes, one for each guild. Node shapes are also different for each guild. The center of each node is plotted at a distance that is its k_{radius}. Angle does not convey information, the algorithm computes it to reduce node overlapping. The area is proportional to k_{degree}. Color is assigned according to *k-shell*. Optionally, the user may choose to display the histograms of k_{degree}, k_{radius} and *k-shell*.

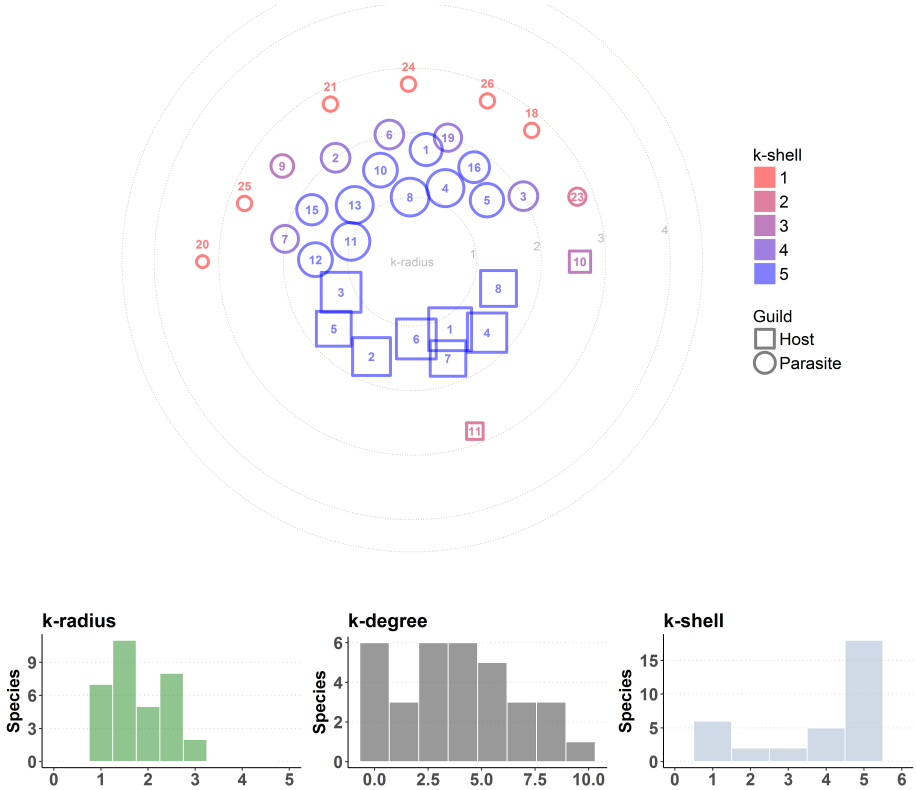

Figure 2: Host-parasite assembly in Tyva (Russian Federation) [20].

2.2. The Ziggurat plot

The polar plot does not show network links, as it works on the information reduction strategy. The ultimate goal of the research was the creation of a new

kind of diagram with as many details as possible.

The initial idea was grouping species by their *k-shell*. If we stick species with the same *k-index* in a reduced area, links among the same shell nodes will not spread across the whole space. Only ties with their edges in different shells would have long paths. This simple principle was not so easy to implement. The bipartite nature of networks means that links go from one guild to the opposite.

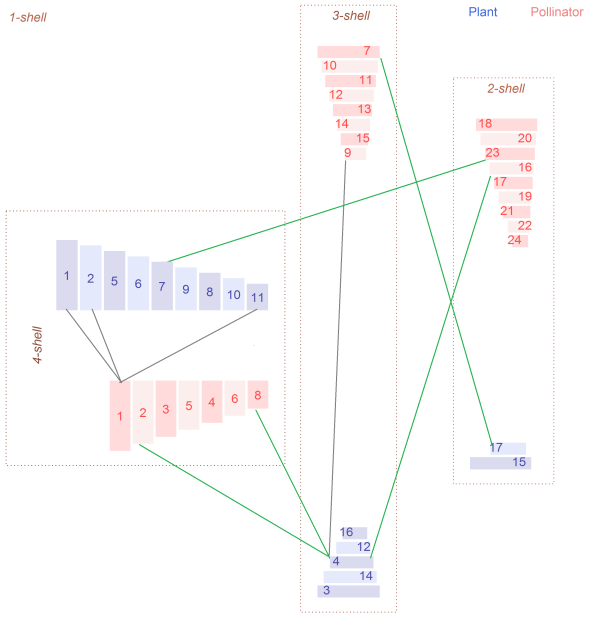

Figure 3: Building blocks of the ziggurat plot.

The typical configuration of generalists and specialists implies that there are many ties from 1-*shell* species towards upper *k-index* groups. Nodes with high degree are very prone to be visually suffocated by surrounding links in the bipartite graph. See plant species numbers $1, 2, 3, 4$ in figure 1. This danger is a formidable obstacle.

Figure 3 shows a ziggurat plot under construction. It is the same network of fig. 1. The *k-core* decomposition puts each species inside one shell, we do not show nodes of 1-*shell* at this moment. The maximum *k-index* is 4 for this community.

The innermost shell is found on the center of the plot, slightly leftwards. Nodes are rectangular-shaped, and are ordered by k_{degree}. Heights decrease just for plotting convenience. The specular position of both guilds leaves space to draw the links among them. In fig. 3 we have plotted just three connections from *pollinator*1 towards plants of 4-*shell*.

Lower *k-shells* have ziggurat shape, with species ordered by ascending k_{radius}, so *pollinator*7 is the closest to the innermost shell in 3-*shell*. Links inside the shell (gray color) connect the left sides of rectangles (*plant*4-*pollinator*9). Links between two different shells (green) connect the right side of the highest *k-index* species to the left side of the lowest one (*plant*17-*pollinator*7).

3-*shell* ziggurats are more distant from the horizontal axis than 2-*shell* ziggurats. Moving them up or down, it is possible to change the area of the internal almond-shaped space defined by the ziggurats and the innermost shell triangles. This area is key because links from 4-*shell* lay here, and do not cross the inner ziggurats.

The outer space is the 1-*shell* species home. We divide them into three groups: outsiders, tails and chains of specialists. Outsiders are species disconnected from the giant component. They are unusual in recorded ecological networks because by definition they do not interact with the community. This network lacks outsiders. Tails are species directly connected to higher *k-indexnodes*. They are very common, and to reduce the number of lines we apply a simple grouping rule. If n tails are tied to the same species of a ziggurat, we plot them in a unique box with just one link. Chains of specialists are less frequent. They are built with species of 1-*shell* linked among them, the edge that has a link with a higher *k-index* shell is the root node (*plant*13).

Links are drawn as straight lines or splines that make the diagram more appealing to the eye. It the interaction matrix is weighted, setting the width of each link to be proportional to a function of the interaction strength is optional.

Bringing everything together, we obtain the ziggurat plot of fig. 4. This may be compared with the bipartite graph of the same network (fig. 1).

Ziggurat plots show the beauty and diversity of ecological communities and

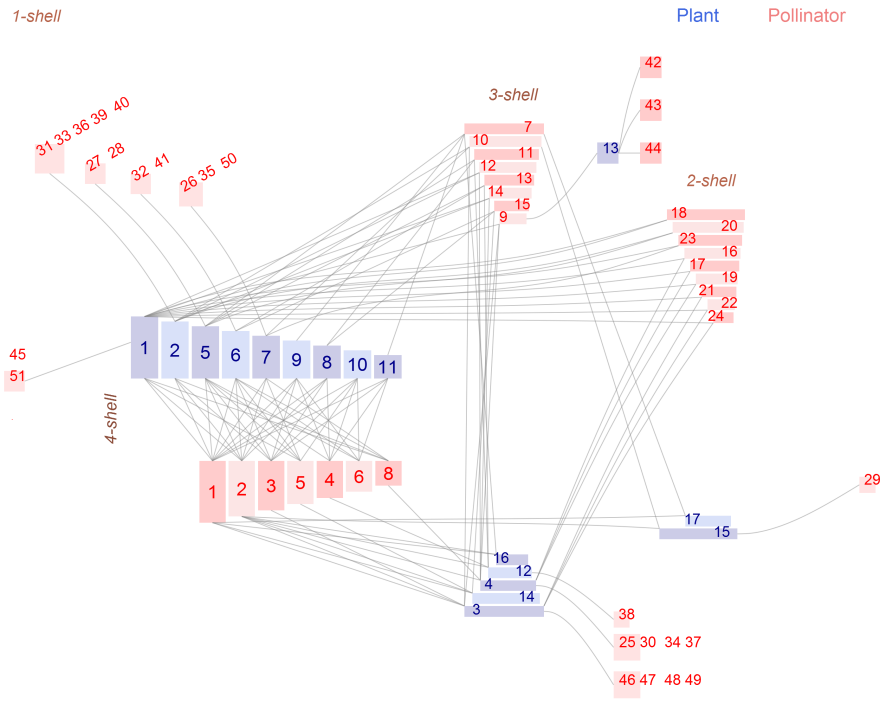

Figure 4: Ziggurat plot of the same plant-pollinator community in Tenerife, Canary Islands (Spain), that appears in fig. 1

provide a quick visual insight of network properties. Figure 5 is a large pollinator community with 185 species and 361 interactions. There are many tails of both guilds, just one chain of specialists and a small number of outsiders. It has a complex and strong structure.

3. The kcorebip package

The `R` package `kcorebip` contains the functions to perform the *k-core* analysis and to plot static ziggurat and polar graphs of a network. The package provides a detailed user manual and a set of networks for testing purposes. These data were downloaded from the *web of life* database [22]. As the format of the *web of life* files has became a standard *de facto* by its simplicity, `kcorebip`

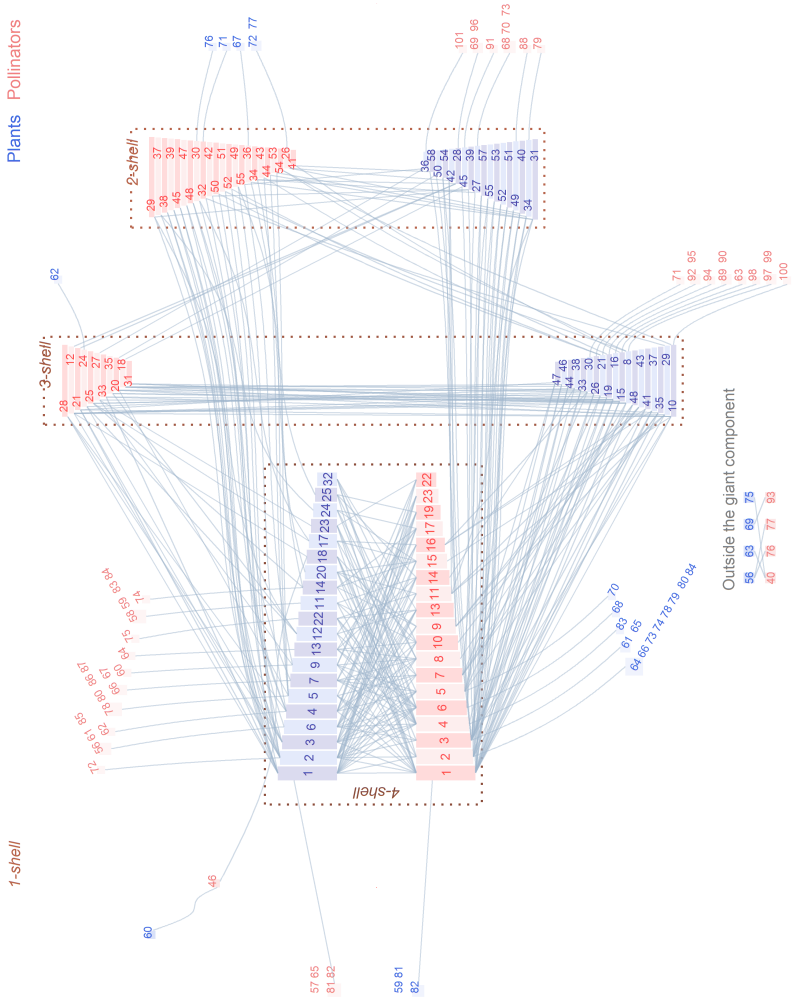

Figure 5: Plant-pollinator community in Chile [21].

follows the same convention for input files.

The function `network_k_analysis` computes the *k-magnitudes* and other useful indexes, using the functions that provide packages as `bipartite` and `igraph` [23, 24]. We refer to the user manual for details.

Ziggurat and polar graphs use basic calls to the `ggplot2` graphics package [25]. We compute from scratch coordinates and sizes, not relying on other

network plotting libraries.

4. Interactive application

The `kcorebip` package is a powerful solution for researchers with programming skills that need high quality plots for scientific publications. When we requested feedback of potential users in the ecology field, a major concern was usability. `Bipartgraph` was designed with this need in mind.

The technological choice was straightforward once `kcorebip` was ready. `Shiny`, the `R` reactive programming environment, has the advantage of a native backend and a `Javascript`-based user interface that may be easily extended. This combination of technologies ensures a wide compatibility with most common operating systems.

The `Interactive Ziggurat` is the main feature of `Bipartgraph`. The first implementation of the `kcorebip` package only provided the static `ggplot2` object to display or save. To create an interactive version we faced two main choices, replicating the code with a dynamic technology or extending `kcorebip`. We found a fast and almost non intrusive solution creating a new `SVG` object. The ziggurat is a set of rectangles, lines and texts. The most time consuming tasks are network analysis and spatial distribution. These computations are performed just once, and besides each `ggplot2` element the function plots, it creates the `SVG` equivalent.

The browser displays the `SVG` ziggurat with multiple options for the user: tooltips, select a node or a link, highlight connections, zoom in and zoom out. In addition, a second panel shows information of highlighted species and the available information on Wikipedia (fig. 6).

The configuration panels make plot properties easy to modify. Visual and intuitive Shiny controls, as sliders or checkboxes, hide the complexity of the input parameters of the `ziggurat_graph` function.

At any moment, the user may download the high quality, high resolution static plot with the `Printable Ziggurat` option. In order to reproduce the

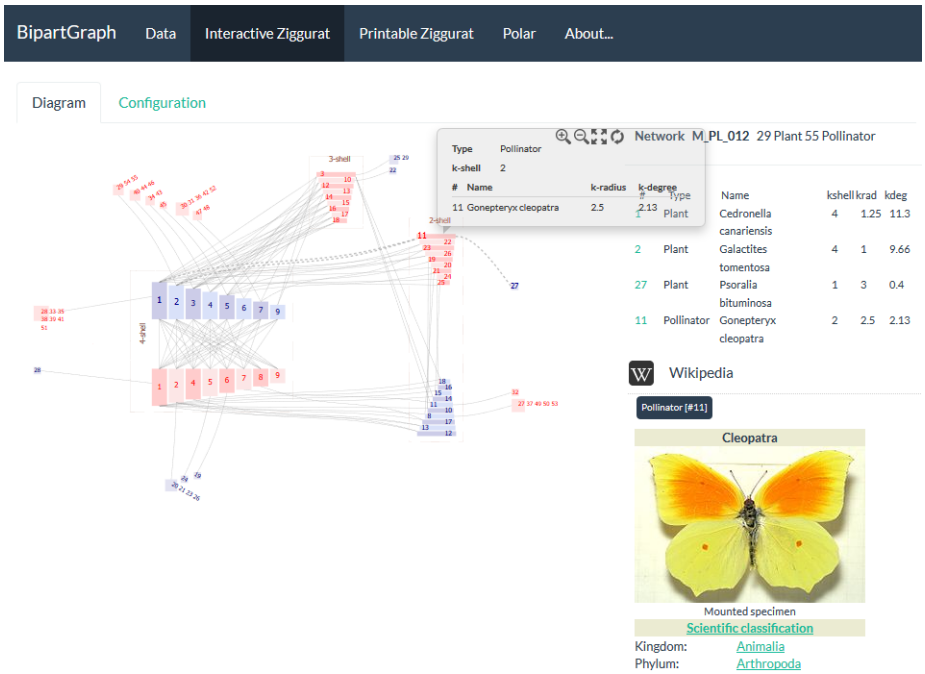

Figure 6: Interactive ziggurat user interface. Plant-pollinator assembly in Garajonay, Canary Islands (Spain), recorded by Olesen. Highlighted the pollinator species number 11, *Gonepteryx cleopatra*

results or to include the graph in other environments, such as `R Markdown` or `Jupyter` notebooks, we added the `Download generating code` button. When clicked upon, `Bipartgraph` writes a file with the last `ziggurat_graph` call, ready to use in any `R` script.

There is not interactive version of the `Polar` plot, as we think that network exploration is much easier with the ziggurat. The user may produce the static polar plot, the high quality downloadable `PNG` file and the generating code, in the same way that we have explained with the ziggurat.

5. Conclusions

Visualization of bipartite ecological networks is a useful task for researchers. However, the available toolset is rather limited. Using the *k-core* decomposition

we have designed and developed two new graphs that work by information reduction (Polar plot) and spatial grouping by connectivity (Ziggurat plot). They provide two complementary views of internal network structure and behave properly for the size of typical ecological communities.

Software is provided as Open Source, under a very loose MIT license, and comes in two versions. The package `kcorebip` provides the full functionality for researchers with a minimum of `R` programming skills. The application `Bipartgraph` is the fully fledged interactive environment to build both kind of graphs for this public. Its user centric design makes it very easy to master, provides some additional features and is open to new fields of application such as education.

Acknowledgements

This work was supported by Ministry of Economy and Competitiveness of Spain (MTM2015-63914-9).

Data availability

Interaction matrixes were downloaded from the web of life database http://www.web-of-life.es/. A subset of these matrixes is installed by default with *BipartGraph*, including all networks used in this paper.

References

[1] K. McCann, Protecting biostructure, Nature 446 (7131) (2007) 29.

[2] P. Jordano, Patterns of mutualistic interactions in pollination and seed dispersal: connectance, dependence asymmetries, and coevolution, The American Naturalist 129 (5) (1987) 657–677.

[3] N. P. Rougier, M. Droettboom, P. E. Bourne, Ten simple rules for better figures, PLoS computational biology 10 (9) (2014) e1003833.

[4] E. J. Mulrow, The visual display of quantitative information (2002).

[5] A. Cairo, The Functional Art: An introduction to information graphics and visualization, New Riders, 2012.

[6] M. J. Pocock, D. M. Evans, C. Fontaine, M. Harvey, R. Julliard, Ó. McLaughlin, J. Silvertown, A. Tamaddoni-Nezhad, P. C. White, D. A. Bohan, Chapter two-the visualisation of ecological networks, and their use as a tool for engagement, advocacy and management, Advances in Ecological Research 54 (2016) 41–85.

[7] S. Yoon, I. Yoon, R. Williams, N. Martinez, J. Dunne, 3d visualization and analysis of ecological networks on www, in: International Conference on Computer Graphics and Imaging, MH Hamza, ISBN 0-88986-001-7, 2004, pp. 224–229.

[8] C. Kazancı, Econet: A new software for ecological modeling, simulation and network analysis, Ecological Modelling 208 (1) (2007) 3–8.

[9] P. W. Lane, D. B. Lindenmayer, P. S. Barton, W. Blanchard, M. J. Westgate, Visualization of species pairwise associations: a case study of surrogacy in bird assemblages, Ecology and evolution 4 (16) (2014) 3279–3289.

[10] C. O. Flores, T. Poisot, S. Valverde, J. S. Weitz, Bimat: a matlab package to facilitate the analysis of bipartite networks, Methods in Ecology and Evolution 7 (1) (2016) 127–132.

[11] L. Stald, A. Valido, J. Olesen, Struktur og dynamik i rum og tid at et bestøvningsnetværk på tenerife, de kanariske øer, Master's thesis, University of Aarhus, Denmark (2003).

[12] J. Bascompte, P. Jordano, C. J. Melián, J. M. Olesen, The nested assembly of plant–animal mutualistic networks, Proceedings of the National Academy of Sciences 100 (16) (2003) 9383–9387.

[13] M. E. Newman, Modularity and community structure in networks, Proceedings of the National Academy of Sciences 103 (23) (2006) 8577–8582.

[14] S. B. Seidman, Network structure and minimum degree, Social Networks 5 (3) (1983) 269–287.

[15] V. Batagelj, M. Zaversnik, An o (m) algorithm for cores decomposition of networks, arXiv preprint cs/0310049.

[16] J. García-Algarra, J. M. Pastor, J. M. Iriondo, J. Galeano, Ranking of critical species to preserve the functionality of mutualistic networks using the k-core decomposition, PeerJ 5 (2017) e3321.

[17] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley, H. A. Makse, Identification of influential spreaders in complex networks, Nature Physics 6 (11) (2010) 888–893.

[18] H. Zhang, H. Zhao, W. Cai, J. Liu, W. Zhou, Using the k-core decomposition to analyze the static structure of large-scale software systems, The Journal of Supercomputing 53 (2) (2010) 352–369.

[19] J. I. Alvarez-Hamelin, L. Dall'Asta, A. Barrat, A. Vespignani, k-core decomposition: A tool for the visualization of large scale networks, arXiv preprint cs/0504107.

[20] J. D. Hadfield, B. R. Krasnov, R. Poulin, S. Nakagawa, A tale of two phylogenies: comparative analyses of ecological interactions, The American Naturalist 183 (2) (2013) 174–187.

[21] M. T. K. Arroyo, R. Primack, J. Armesto, Community studies in pollination ecology in the high temperate andes of central chile. i. pollination mechanisms and altitudinal variation, American journal of botany (1982) 82–97.

[22] M. A. Fortuna, R. Ortega, J. Bascompte, The web of life, arXiv preprint abs/1403.2575.

[23] C. F. Dormann, B. Gruber, J. Fründ, Introducing the bipartite package: analysing ecological networks, interaction 1 (2008) 0–2413793.

[24] G. Csardi, T. Nepusz, The igraph software package for complex network research, InterJournal, Complex Systems 1695 (5) (2006) 1–9.

[25] H. Wickham, ggplot2: elegant graphics for data analysis, Springer, 2016.