

## APPLICATION

# BiMat: a MATLAB package to facilitate the analysis of bipartite networks

Cesar O. Flores<sup>1,\*</sup>, Timothée Poisot<sup>2,3,4</sup>, Sergi Valverde<sup>5,6</sup> and Joshua S. Weitz<sup>7,1,\*</sup>

<sup>1</sup>School of Physics, Georgia Institute of Technology, Atlanta, GA, 30332, USA; <sup>2</sup>School of Biological Sciences, University of Canterbury, Private Bag 4800, Christchurch, 8140, New Zealand; <sup>3</sup>Département de Sciences Biologiques, Université de Montréal, 90 Avenue Vincent d'Indy, Montréal, QC, H2V3S9, Canada; <sup>4</sup>Québec Centre for Biodiversity Sciences, 1205 Dr. Penfield Avenue, Montréal, QC, H3A 1B1, Canada; <sup>5</sup>Complex Systems Lab, Universitat Pompeu Fabra, Dr Aiguader 80, 08003 Barcelona, Spain; <sup>6</sup>Institute of Evolutionary Biology (CSIC-UPF), Passeig Marítim de la Barceloneta 37-49, E-08003 Barcelona, Spain; and <sup>7</sup>School of Biology, Georgia Institute of Technology, Atlanta, GA, 30332, USA

## Summary

1. Bipartite networks are ubiquitous in community ecology, including examples of facilitative interactions, such as plant–pollinator networks, and antagonistic interactions, such as virus–host infection networks. Statistical network analysis is increasingly used to identify emergent, nonrandom patterns of interaction and the effect of interaction patterns on ecological and evolutionary dynamics. Two recurring patterns are that of modularity and nestedness. Modularity is a feature of networks in which there are densely interacting subgroups. Nestedness is a feature of networks in which the interactions form ordered subsets.

2. Here we describe BiMat, an open-source MATLAB package for the study of the structure of bipartite ecological networks. Unlike alternative tools, BiMat enables both multiscale analysis of the structure of a bipartite ecological network – spanning global (i.e. entire network) to local (i.e. module-level) scales – and meta-analyses of many bipartite networks simultaneously. In common with other tools, BiMat calculates the degree to which a bipartite network is modular and/or nested, the statistical significance of these patterns, and enables visualization of latent structures in the network.

3. BiMat is available as an open-source MATLAB package, with a quick-start guide and worked examples at: <http://bimat.github.io/>.

**Key-words:** community ecology, complex networks, ecological networks, host–parasite interactions, microbial ecology

## Introduction

Interactions amongst distinct sets of individuals in an ecological community can be represented as bipartite networks. Prominent examples of bipartite networks include plant–pollinator networks (Bascompte *et al.* 2003; Dunne 2006; Bascompte & Jordano 2007; Olesen *et al.* 2007; Bastolla *et al.* 2009) and phage–bacteria infection networks (Flores *et al.* 2011; Poisot *et al.* 2011; Flores, Valverde & Weitz 2013; Poisot, Lounnas & Hochberg 2013; Weitz *et al.* 2013). The nodes in the network are members of two distinct sets: plants and pollinators in the first example and phage and bacteria in the second example. An edge connecting two nodes denotes that an interaction takes place and the absence of an edge denotes that an interaction does not take place. In a bipartite network, interactions, for example, pollination or infection, are considered only between nodes of distinct sets.

Analysis of plant–pollinator and host–parasite systems has identified emergent patterns of modularity and nestedness. A

modular network is one in which subsets of nodes often preferentially connect to each other, rather than to other nodes (Olesen *et al.* 2007). For bipartite networks, we assume that the nodes within a module must necessarily connect to nodes from the other set (Barber 2007) [but see (Guimerà, Sales-Pardo & Nunes Amaral 2007; Larremore, Clauset & Jacobs 2014)]. A nested network is one in which the interaction between nodes form (partially) ordered subsets of each other (Bascompte *et al.* 2003; Flores *et al.* 2011). For bipartite networks, this ordering applies to both sets (e.g. plants and pollinators or phage and bacteria). Understanding the structure of ecological networks can provide insights into the generation and maintenance of diversity (Bascompte *et al.* 2003; Bascompte & Jordano 2007; Bastolla *et al.* 2009; Joppa *et al.* 2009; Stouffer & Bascompte 2011). Similarly, characterizing an interaction network may also help predict which species are most likely to go extinct (Harfoot *et al.* 2014).

A number of tools have been developed to analyse the structure of complex networks. These tools enable researchers to interactively query and analyse a network [e.g. Gephi (Bastian, Heymann & Jacomy 2009), Cytoscape (Shannon *et al.* 2003)] and to embed network analysis into workflows via a library

\*Correspondence author: E-mail: cesar7@gmail.com and jsweitz@gatech.edu

framework [e.g. NetworkX (Hagberg, Swart & Chult 2008), iGraph (Csardi & Nepusz 2006)]. Available tools focus largely on the analysis of unipartite networks, that is, bipartite networks are represented as a special case of unipartite networks. In contrast, `bipartite` is a general package for the analysis of bipartite networks (Dormann *et al.* 2009). `bipartite` nestedness and modularity, within a R language framework (Dormann, Gruber & Fründ 2008; Dormann & Strauss 2014). Additional tools focusing on bipartite networks have been developed specifically for the analysis of one kind of network pattern, for example, nestedness estimation as in ANINHADO (Guimaraes & Guimarães 2006), WINE (Galeano, Pastor & Iriando 2009), and recently FALCON (Beckett, Boulton & Williams 2014), or modularity estimation as in MODULAR (Marquitti *et al.* 2014).

Here we present `BiMat`, a MATLAB library for the analysis and visualization of bipartite networks, that introduces new features not available in other bipartite network analysis tools and replicates some of the features of the `bipartite` R package in a different language and framework. Consistent with standards developed in established packages, `BiMat` enables the analysis of bipartite network structure, including modularity and nestedness, and the significance of identified patterns. In addition, `BiMat` includes meta-analysis tools for the analysis of many related networks simultaneously. Crucially, the structure of a network may be different at different scales, as was recently demonstrated in the case of phage–bacteria infection networks that were found to be modular at the entire network scale, yet the identified modules had internal nested structure (Flores, Valverde & Weitz 2013). In response, we have included a suite of methods to analyse the multiscale structure of bipartite networks. The integration of all of these features into a single package is not available in other tools. Finally, `BiMat` provides a range of visualization tools for exploring bipartite network structure in either matrix or graph layouts.

We recognize that there is not a single language used and preferred by researchers interested in bipartite networks. The use of MATLAB may represent a feature for some and an obstacle for others. In order to facilitate the use of the tool and the replication of its methods in other languages, we have released the `BiMat` package as an open-source library, with a quick-start guide and worked examples at <http://bimat.github.io/>. We have also released a port of `BiMat` for use in the open-access GNU Octave language, on the same repository.

## Bipartite network analysis – theory and implementation

### TERMINOLOGY

A bipartite network,  $\mathbf{B} = (R, C, E)$ , denotes two nonoverlapping sets of nodes  $u \in R$  (or row nodes) and  $v \in C$  (or column nodes). In a phage–bacteria infection network, one set of nodes corresponds to host bacteria and the other to viruses. In the case of phages and hosts, a link  $e = (u, v) \in E$

denotes infection. In the pollinator example, there will be a link between an insect/bird and a plant if the former can pollinate the latter. Links between nodes of the same kind are forbidden.

The adjacency matrix  $\mathbf{B}$  of size  $m \times n$  is an alternative definition of a bipartite network, where  $m = |R|$  and  $n = |C|$  is the size of the sets  $R$  and  $C$ , respectively. Network links and entries of the adjacency matrix only indicate the presence (1) or absence (0) of pairwise interactions, that is, we do not consider weighted bipartite networks. The total number of links in the network is  $E = \sum_{ij} B_{ij}$ . The degree  $k_i = \sum_j B_{ij}$  is the number of interactions of row node  $i$ , and  $d_j = \sum_i B_{ij}$  is the degree of a column node  $j$ .

### NESTEDNESS

Nestedness describes the extent to which interactions form ordered subsets of each other. There are several indices for quantifying nestedness [see (Ulrich, Almeida-Neto & Gotelli 2009) for details]. `BiMat` implements two of the most commonly used methods, namely NODF (for nestedness metric based on overlap and decreasing fill) (Almeida-Neto *et al.* 2008) and temperature (Atmar & Patterson 1993; Rodríguez-Gironés & Santamaría 2006).

### NODF

The NODF nestedness metric is based on the extent to which a network exhibits decreasing fill and paired overlap. NODF measures nestedness across rows by assigning a value  $M_{ij}^{\text{row}}$  to each pair  $i, j$  of rows in the interaction matrix (Almeida-Neto *et al.* 2008):

$$M_{ij}^{\text{row}} = \begin{cases} 0 & \text{if } k_i \leq k_j \\ n_{ij} / \min(k_i, k_j) & \text{otherwise} \end{cases} \quad (1)$$

where  $k_i$  is the number of ones in row  $i$ ,  $k_j$  is the number of ones in row  $j$ , and  $n_{ij}$  is the number of shared interactions between rows  $i$  and  $j$  (so-called paired overlap). Formally,  $n_{ij}$  is equal to the number of entries in which  $\mathbf{B}(i,:) = \mathbf{B}(j,:)$ . Notice that positive contributions to NODF require pairs of columns to exhibit decreasing fill. The term ‘fill’ denotes the degree of a node, and hence, decreasing fill is satisfied when  $k_i > k_j$ , such that the degree of node  $j$  is less than that of node  $i$ . A similar term  $M_{ij}^{\text{col}}$  is used to compute column contributions. The total nestedness is the sum of columns and row contributions:

$$N_{\text{NODF}} = \frac{\sum_{ij} M_{ij}^{\text{row}} + \sum_{ij} M_{ij}^{\text{col}}}{\left[ \frac{m(m-1)}{2} \right] + \left[ \frac{n(n-1)}{2} \right]}. \quad (2)$$

The NODF metric normalizes for matrix size and thus allows matrices of different sizes to be compared. The nestedness range is  $0 \leq N_{\text{NODF}} \leq 1$ , where 0 indicates the lack of any decreasing fill and paired overlap (e.g. a block-like matrix or checkerboard matrix) and 1 corresponds to a perfectly nested structure.

In practice, MATLAB vectorization capabilities allow for the efficient calculation of (1) and its column version:

$$M_{ij}^{\text{row}} = \frac{(\mathbf{r}_i \cdot \mathbf{r}_j)(1 - \delta(k_i, k_j))}{\min(k_i, k_j)}, \quad (3)$$

where  $\mathbf{r}_i$  is a vector that represents the row  $i$  of the adjacency matrix, and  $\delta(k_i, k_j) = 1$  if and only if  $k_i = k_j$ . Eqn (2) can be expressed in terms of matrix multiplications (while normalizing for double-counting) that exploit the implicit parallelism provided by MATLAB (see code for details).

### Temperature

The temperature metric for nestedness is estimated by sorting rows and columns such that the largest quantity of interactions falls above the isocline. An isocline is a curve that divides the interaction from the noninteraction zone of a perfectly nested matrix of the same size and connectance. In doing so, the value of  $T$  quantifies the extent to which interactions only take place in the upper left ( $T \approx 0$ ) or are equally distributed between the upper left and the lower right ( $T \approx 100$ ). The interactions within perfectly nested interaction matrices can be sorted to lie exclusively in the upper left portion and hence have a temperature of 0. The value of temperature depends on the size, connectance and structure of the network. We define the nestedness,  $N_{\text{NTC}} = (100 - T)/100$ , such that  $N_{\text{NTC}} = 1$  when  $T = 0$  (corresponding to perfect nestedness) and  $N_{\text{NTC}} = 0$  when  $T = 100$  (corresponding to ‘checkerboard’ matrices). We use the phrase ‘Nestedness Temperature Calculator’ (NTC) to denote the value of nestedness calculated using this approach (Atmar & Patterson 1993; Rodríguez-Gironés & Santamaría 2006).

### MODULARITY

Modularity indicates the presence of dense clusters of nodes with many overlapping interactions embedded within the network (Fortunato 2010). Clusters are considered dense when they have high internal edge density relative to the expected edge density in the null model. These dense clusters are termed modules. Identifying modules and estimating the associated modularity require a partitioning of the network. The modularity of a bipartite network is defined as:

$$Q = \frac{1}{|E|} \sum_{ij} \left( B_{ij} - \frac{k_i d_j}{|E|} \right) \delta(g_i, h_j), \quad (4)$$

where  $g_i$  and  $h_j$  are the module indices of nodes  $i$  (that belongs to set  $R$ ) and  $j$  (that belongs to set  $C$ ),  $k_i$  is the degree of node  $i$ ,  $d_j$  is the degree of node  $j$ , and  $|E|$  is the number of links in the network. The expression for calculating modularity can be written in matrix notation, thus allowing an efficient MATLAB implementation:

$$Q = \frac{1}{|E|} \text{Tr} \mathbf{R}^T \tilde{\mathbf{B}} \mathbf{T} \quad (5)$$

where  $\tilde{B}_{ij} = B_{ij} - \frac{k_i d_j}{|E|}$  is the so-called modularity matrix and  $c$  is the number of modules. In this case,  $\mathbf{g}$  and  $\mathbf{h}$  vectors are replaced by the  $m \times c$  index matrix  $\mathbf{R} = [\mathbf{r}_1 | \mathbf{r}_2 | \dots | \mathbf{r}_m]^T$  and the  $n \times c$  index matrix  $\mathbf{T} = [\mathbf{t}_1 | \mathbf{t}_2 | \dots | \mathbf{t}_n]^T$ .

The modularity algorithms within BiMat attempt to maximize eqn (4) by partitioning the network. The standard maximization approach is the Bipartite Recursively Induced Modules (BRIM) algorithm (Barber 2007). The BRIM algorithm computes the optimal modularity by assigning nodes into modules successively to maximize the per-node contribution of modularity given prior assignments. In that way, each set of nodes (e.g. vector  $\mathbf{T}$ ) recursively induces the other set of nodes (e.g. vector  $\mathbf{R}$ ). BRIM assigns nodes of each type to modules until a local maximum is reached. Maximizing eqn 4 is NP-hard (Miyachi & Sukegawa 2014), and so BiMat implements two different heuristic approaches, adaptive BRIM as well as LP & BRIM:

- **AdaptiveBRIM:** The algorithm uses the BRIM algorithm with a heuristic for identifying the optimal network partition. The heuristic involves doubling the number of modules by a factor of two until  $Q$  decreases and then using a bisection method to identify the maximal value of  $Q$  between the current and previous partition.
- **LP&BRIM:** This method combines label propagation (LP) with the BRIM algorithm to partition the community (Liu & Murata 2010). The algorithm consists of two stages. First during the LP phase, neighbouring nodes (i.e. those which share links) exchange their labels representing the community they belong to, with each node receiving the most common label amongst its neighbours. We iterate this process until densely connected groups of nodes reach a consensus of what is the most representative label, as indicated by the fact that the modularity is not increased by additional exchanges. Secondly, the BRIM algorithm refines the partitions found with label propagation.

### STATISTICAL SIGNIFICANCE

The values of modularity and nestedness depend on network size and the density of links (Dormann *et al.* 2009). Null models can be used to correct for size and density effects. We calculate the statistical significance  $p$  as the likelihood that the measured value of nestedness and modularity of the original network is greater than or equal to that measured in an ensemble of suitably chosen random networks. The choice of the random network depends on the system under observation. BiMat provides several null models used to generate ensembles of random networks that make different assumptions about the expected probability of interaction  $P_{ij}$  between two species. The null models include ‘equiprobable’, ‘average’, ‘columns’, ‘rows’ and ‘fixed’. Each class of null models preserves some feature of the marginal distribution of the original network, either exactly or on average (see <http://bimat.github.io/> for visual illustrations of each of these null models):

**Equiprobable:** Random networks have, on average, the same overall connectance as the original network.

**Average:** Random networks have, on average, the same overall connectance as the original network, and each node has the same expected number of interactions as in the original.

**Columns:** Random networks have, on average, the same overall connectance as the original network, and the expected num-

ber of interactions of row nodes is preserved, that is, preserving the sum across columns.

**Rows:** Random networks have, on average, the same overall connectance as the original network, and the expected number of interactions of column nodes is preserved, that is, preserving the sum across rows.

**Fixed:** The exact marginal distributions are preserved (i.e. row and column sums). A swapping algorithm is utilized to generate random networks.

## IMPLEMENTATION

**BiMat** is developed in MATLAB using an Object-Oriented Programming (OOP) paradigm. The use of OOP is meant to facilitate maintainability and extensibility of the codebase; prior experience with OOP is not required for use of **BiMat**. Access to **BiMat** functions is granted (with the exception of some static classes) using instances of the class that implements the functions. The main package class is the `bipartite` class, which works as a common interface to the available statistical, algorithmic, plotting, input and output classes. Because of this OOP design pattern, most of the functionality is accessible using the following syntax:

```
bip.class_instance_in_bip.method_name(arguments)
```

where `bip` is a bipartite instance created by the user, `class_instance_in_bip` is a property of the `bipartite` class which represents an instance of the class which has access to the method `method_name`. The method that is called will frequently have direct read and writable access to other properties inside `bip`. Documentation on the use of the OOP framework for estimation of network structure is available in the quick-start guide released in the **BiMat** package. In addition to conventional estimation of network patterns, **BiMat** also enables researchers to perform a meta-analysis of many networks and a multiscale analysis of a single network. The meta-analysis feature is described on the project home page, in the quick-start guide, and an example of a multiscale analysis is described next.

## Multiscale analysis

The multiscale analysis within **BiMat** consists of the following steps:

**Module detection:** The modular structure of the entire network is identified.

**Submodule structural analysis:** Network analysis is performed at the intermodule scale. **BiMat** evaluates the extent to which modules exhibit internal structure, including modularity and/or nestedness.

**Submodule label analysis:** Additional node information may be available, for example, the size of individuals of a node or the location of collection. This information is termed a node 'label'. As an optional step, **BiMat** can evaluate whether or not the partitioning of nodes within modules is related to properties of the node labels.

As an example, consider the phage–bacteria infection network of Moebius and Nattkemper (Moebius & Nattkemper 1981). The individual phage and bacteria in this study were isolated from different locations across the Atlantic Ocean. In a previous study, we developed a multiscale analysis of network structure in this data set, including 286 phage and 215 bacteria types (Flores, Valverde & Weitz 2013). **BiMat** can be used to first identify the global-scale structure of a bipartite network. Assuming that the original interaction matrix is stored in the variable `moebius.weight_matrix`, then the left panel of Fig. 1 shows a visual representation of this data in matrix layout after the following sequence of commands:

```
bp=Bipartite(moebius.weight_matrix);
bp.community.Detect();
bp.plotter.use_module_format=false;
bp.plotter.isocline_color='black';
bp.plotter.division_color='black';
bp.plotter.use_labels=false;
bp.plotter.PlotModularMatrix();
```

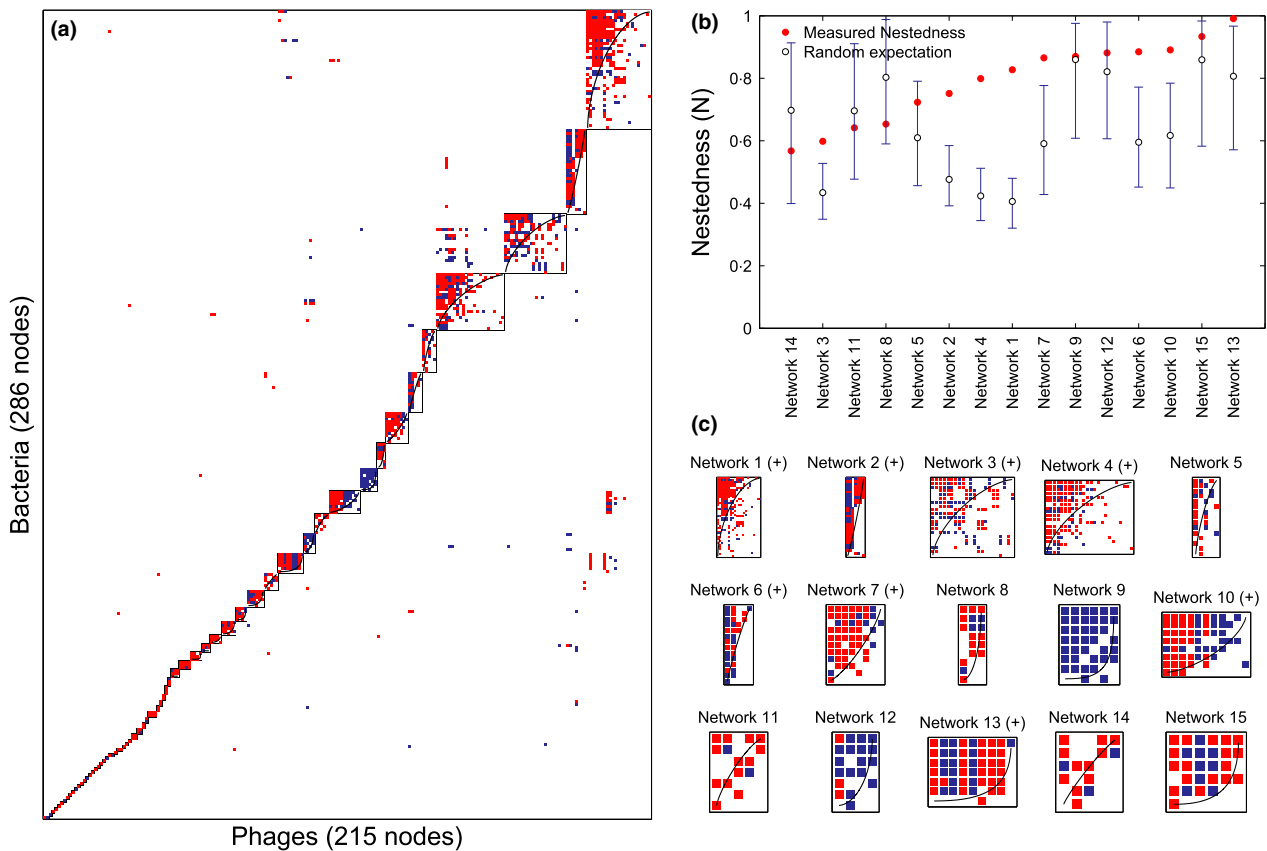
The modularity significance detection suite confirms that the network is modular. Yet, the modules seem to have nested structure. For example, there is a triangular pattern of interactions with most of the links above the temperature isocline. To measure nestedness within modules, **BiMat** makes use of the `InternalStatistics` class by treating each of the modules as an independent network:

```
bp.internal_statistics.idx_to_focus_on=1:15;
bp.internal_statistics.TestInternalModules();
bp.internal_statistics.meta_statistics...
.plotter.PlotNestednessValues();
bp.internal_statistics.meta_statistics...
.plotter.PlotNestedMatrices();
```

The results of the last two plot commands can be seen in the right panels of Fig. 1. The `meta_statistics` property is an instance of the class `MetaStatistics`. As such, it can use any of methods inside `MetaStatistics` including its property `plotter` in the internal modules. This feature is a consequence of the use of OOP in developing **BiMat**.

Finally, **BiMat** can evaluate whether there is a relationship between node labels and module distribution. This feature is of particular use when the node information is available, for example, with respect to their study origin or other categorical (i.e. nonmetric) feature. If there is a strong relationship between label and module, then every node inside the same module will share the same (or similar) label. **BiMat** makes use of both Shannon's and Simpson's indices to analyse the label variation inside and between modules (Schloss & Handelsman 2005). The heterogeneity of label indices is measured within each module. Then, node labels are randomly swapped, generating an ensemble from which to compare the measured relationship.





**Fig. 1.** Examples of visualizations using the multiscale analysis capabilities of BiMat. The analysis here recapitulates results in Flores, Valverde & Weitz (2013) using the BiMat software library to examine the phage–bacteria interaction data from Moebius & Nattkemper (1981) (a) A modular matrix layout of the interaction data, suggestive of multiscale structure. (b) Comparisons of  $N_{NTC}$  values within identified modules with respect to random expectations. Error bars represent 95% of the random replicate values. (c) Visualization of the interactions within modules where red labels indicate statistically significant  $N_{NTC}$  values. The plus signs denote instances where modules have elevated nestedness values.

## Conclusion and future work

We have developed the BiMat package to enable structural analysis of bipartite ecological networks within a MATLAB environment. We have also ported BiMat for use in the GNU Octave environment. BiMat enables the identification of key patterns, including modularity and nestedness, and the evaluation of their statistical significance. In that sense, BiMat is similar to existing tools, including bipartite (Dormann *et al.* 2009), a software library written in R. However, unlike other tools, BiMat also includes capabilities to perform a meta-analysis of the structure of multiple networks and to analyse the multiscale structure of networks. In future releases, BiMat will incorporate methods to analyse the structure of bipartite networks where interactions are weighted, rather than strictly Boolean.

## Acknowledgements

We are indebted to Yongmin Cho, Abhiram Das, Agnes Huettman, Luis Jover, Julio Cesar Ignacio Espinoza, Megan Larsen, Joey Leung, Gabriel Mitchell, Pablo Sánchez and Bradford Taylor for testing and providing feedback on BiMat. We are especially grateful to Charles Wigington for testing and providing feedback on BiMat and BiMat-oct. We thank two anonymous reviewers and Fernanda Valdovinos for their feedback on the manuscript. COF acknowl-

edges the support of the CONACyT Foundation. TP thanks the FQRNT-MELS for funding through the PBEE post-doctoral program. SV thanks the Spanish Ministry of Economy and Competitiveness (Grant FIS2013-44674-P) and the Botin Foundation for funding support. JSW acknowledges NSF grant OCE-1233760 and support from a Career Award at the Scientific Interface from the Burroughs Wellcome Fund.

## Data accessibility

The data analysed in this paper are compiled in (Flores *et al.* 2011 and Flores, Valverde & Weitz 2013), available at: <http://www.pnas.org/content/108/28/E288.abstract?tab=ds> and <http://www.nature.com/ismej/journal/v7/n3/supinfo/ismej2012135s1.html>, respectively.

## References

- Almeida-Neto, M., Guimaraes, P., Guimaraes, 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.
- Atmar, W. & Patterson, B.D. (1993) The measure of order and disorder in the distribution of species in fragmented habitat. *Oecologia*, **96**, 373–382.
- Barber, M. (2007) Modularity and community detection in bipartite networks. *Physical Review E*, **76**, 066102.
- Bascompte, J. & Jordano, P. (2007) Plant-animal mutualistic networks: the architecture of biodiversity. *Annual Review of Ecology and Systematics*, **38**, 567–593.
- 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 USA*, **100**, 9383–9387.

- Bastian, M., Heymann, S. & Jacomy, M. (2009) Gephi: an open source software for exploring and manipulating networks. *ICWSM*, **8**, 361–362.
- Bastolla, U., Fortuna, M.A., Pascual-García, A., Ferrera, A., Luque, B. & Bascompte, J. (2009) The architecture of mutualistic networks minimizes competition and increases biodiversity. *Nature*, **458**, 1018–1020.
- Beckett, S.J., Boulton, C.A. & Williams, H.T.P. (2014) FALCON: a software package for analysis of nestedness in bipartite networks (v1; ref status: approved 1, <http://f1000r.es/3z8>). *F1000 Research*, **3**, 185.
- Csardi, G. & Nepusz, T. (2006) The iGraph software package for complex network research. *Inter Journal, Complex Systems*, **1695**, 1–9.
- Dormann, C.F. & Strauss, R. (2014) A method for detecting modules in quantitative bipartite networks. *Methods in Ecology and Evolution*, **5**, 90–98.
- Dormann, C.F., Gruber, B. & Fründ, J. (2008) Introducing the bipartite package: Analysing ecological networks. *R News*, **8**, 8–11.
- Dormann, C.F., Fründ, J., Blüthgen, N. & Gruber, B. (2009) Indices, graphs and null models: analyzing bipartite ecological networks. *The Open Ecology Journal*, **2**, 7–24.
- Dunne, J.A. (2006) The network structure of food webs. *Ecological Networks: Linking Structure to Dynamics in Food Webs*, pp. 27–86. Oxford University Press, Oxford.
- Flores, C.O., Meyer, J.R., Valverde, S., Farr, L. & Weitz, J.S. (2011) Statistical structure of host–phage interactions. *Proceedings of the National Academy of Sciences USA*, **108**, E288–E297.
- Flores, C.O., Valverde, S. & Weitz, J.S. (2013) Multi-scale structure and geographic drivers of cross-infection within marine bacteria and phages. *ISME Journal*, **7**, 520–532.
- Fortunato, S. (2010) Community detection in graphs. *Physics Reports*, **486**, 75–174.
- Galeano, J., Pastor, J.M. & Iriando, J.M. (2009) Weighted-interaction nestedness estimator (WINE): A new estimator to calculate over frequency matrices. *Environmental Modelling & Software*, **24**, 1342–1346.
- Guimaraes, P.R. Jr. & Guimarães, P. (2006) Improving the analyses of nestedness for large sets of matrices. *Environmental Modelling & Software*, **21**, 1512–1513.
- Guimerà, R., Sales-Pardo, M. & Nunes Amaral, L.A. (2007) Module identification in bipartite and directed networks. *Physical Review E*, **76**, 036102.
- Hagberg, A., Swart, P. & Chult, D.S. (2008) Exploring network structure, dynamics, and function using NetworkX. Technical report, Los Alamos National Laboratory (LANL).
- Harfoot, M.B.J., Newbold, T., Tittensor, D.P., Emmott, S., Hutton, J., Lyutsarev, V., Smith, M.J., Scharlemann, J.P.W. & Purves, D.W. (2014) Emergent global patterns of ecosystem structure and function from a mechanistic general ecosystem model. *PLoS Biology*, **12**, e1001841.
- Joppa, L.N., Bascompte, J., Montoya, J.M., Sole, R.V., Sanderson, J. & Pimm, S.L. (2009) Reciprocal specialization in ecological networks. *Ecology Letters*, **12**, 961–969.
- Larremore, D.B., Clauset, A. & Jacobs, A.Z. (2014) Efficiently inferring community structure in bipartite networks. *Physical Review E*, **90**, 012805.
- Liu, X. & Murata, T. (2010) Community detection in large-scale bipartite networks. *Information and Media Technologies*, **5**, 184–192.
- Marquitti, F.M.D., Guimarães, P.R., Pires, M.M. & Bittencourt, L.F. (2014) MODULAR: software for the autonomous computation of modularity in large network sets. *Ecography*, **37**, 221–224.
- Miyauchi, A. & Sukegawa, N. (2014) Maximizing Barber's bipartite modularity is also hard. *Optimization Letters*, **9**, 897–913.
- Moebus, K. & Nattkemper, H. (1981) Bacteriophage sensitivity patterns among bacteria isolated from marine waters. *Helgoländer Meeresuntersuchungen*, **34**, 375–385.
- Olesen, J.M., Bascompte, J., Dupont, Y.L. & Jordano, P. (2007) The modularity of pollination networks. *Proceedings of the National Academy of Sciences USA*, **104**, 19891–19896.
- Poisot, T., Lepennetier, G., Martinez, E., Ramsayer, J. & Hochberg, M.E. (2011) Resource availability affects the structure of a natural bacteria–bacteriophage community. *Biology Letters*, **7**, 201–204.
- Poisot, T., Lounnas, M. & Hochberg, M.E. (2013) The structure of natural microbial enemy-victim networks. *Ecological Processes*, **2**, 1–9.
- 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.
- Schloss, P.D. & Handelsman, J. (2005) Introducing DOTUR, a computer program for defining operational taxonomic units and estimating species richness. *Applied and Environmental Microbiology*, **71**, 1501–1506.
- Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N., Schwikowski, B. & Ideker, T. (2003) Cytoscape, a software environment for integrated models of biomolecular interaction networks. *Genome Research*, **13**, 2498–2504.
- Stouffer, D.B. & Bascompte, J. (2011) Compartmentalization increases food-web persistence. *Proceedings of the National Academy of Sciences USA*, **108**, 3648–3652.
- Ulrich, W., Almeida-Neto, M. & Gotelli, N.J. (2009) A consumer's guide to nestedness analysis. *Oikos*, **118**, 3–17.
- Weitz, J.S., Poisot, T., Meyer, J.R., Flores, C.O., Valverde, S., Sullivan, M.B. & Hochberg, M.E. (2013) Phage–bacteria infection networks. *Trends in Microbiology*, **21**, 82–91.

Received 22 April 2015; accepted 28 July 2015

Handling Editor: Greg McInerney

## Supporting Information

Additional Supporting Information may be found in the online version of this article.

**Appendix S1.** BiMat: Start Guide.