`maxnodf`: an `R` package for fair and fast comparisons of nestedness between networks

Christoph Hoeppke[1,2,3] and Benno I. Simmons[1,4]

1 Conservation Science Group, Department of Zoology, University of Cambridge, The David Attenborough Building, Pembroke Street, Cambridge, CB2 3QZ, UK

2 Faculty of Mathematics, Wilberforce Road, Cambridge CB3 0WA, UK

3 Mathematical Institute, Andrew Wiles Building, Woodstock Road, Oxford, OX2 6GG, UK

4 Centre for Ecology and Conservation, College of Life and Environmental Sciences, University of Exeter, Cornwall Campus, Penryn, UK

Corresponding authors:

Benno I. Simmons. Address: Centre for Ecology and Conservation, College of Life and Environmental Sciences, University of Exeter, Cornwall Campus, Penryn, UK. Email: benno.simmons@gmail.com

Christoph Hoeppke. Address: Mathematical Institute, Woodstock Road, Oxford, OX2 6GG Email: christoph.hoeppke@maths.ox.ac.uk

Running head: maxnodf: fast nestedness comparison in R

Abstract

1. Nestedness is a widespread pattern in mutualistic networks that has high ecological and evolutionary importance due to its role in enhancing species persistence and community stability. Nestedness measures tend to be correlated with fundamental properties of networks, such as size and connectance, and so nestedness values must be normalised to enable fair comparisons between different ecological communities. Current approaches, such as using null-corrected nestedness val-

ues and z-scores, suffer from extensive statistical issues. Thus a new approach called $NODF_c$ was recently proposed, where nestedness is expressed relative to network size, connectance and the maximum nestedness that could be achieved in a particular network. While this approach is demonstrably effective in overcoming the issues of collinearity with basic network properties, it is computationally intensive to calculate, and current approaches are too slow to be practical for many types of analysis, or for analysing large networks.

2. We developed three highly-optimised algorithms, based on greedy, hillclimbing and simulated annealing approaches, for calculation of $NODF_c$, spread along a speed-quality continuum. Users thus have the choice between a fast algorithm with a less accurate estimate, a slower algorithm with a more accurate estimate, and an intermediate option.

3. We outline the package, and its implementation, as well as provide comparative performance benchmarking and two example analyses. We show that `maxnodf` enables speed increases of hundreds of times faster than existing approaches, with large networks seeing the biggest improvements. For example, for a large network with 3000 links, computation time was reduced from 50 minutes using the fastest existing algorithm to 11 seconds using `maxnodf`.

4. `maxnodf` makes correctly-normalised nestedness measures feasible for complex analyses of even large networks. Analyses that would previously take weeks to complete can now be finished in hours or even seconds. Given evidence that correctly normalising nestedness values can significantly change the conclusions of ecological studies, we believe this package will usher in necessary widespread use of appropriate comparative nestedness statistics.

Keywords: bipartite networks, comparison, nestedness, NODF, pollination, R, seed dispersal

Introduction

Nestedness is a widespread and important feature of species interaction networks (Bascompte et al. 2003). Nestedness refers to the tendency for specialist species to interact with subsets of the

species that more generalist species interact with. The prevalence of nested architectures, coupled with their high ecological and evolutionary importance, has given nestedness research a high profile, particularly for networks representing mutualistic interactions between species (Bastolla et al. 2009, Thébault & Fontaine 2010, James et al. 2012, Saavedra & Stouffer 2013, Suweis et al. 2013).

Like many indices of network structure, however, nestedness is correlated with other network properties, like connectance and the number of species in the network, which are themselves also highly correlated (Song et al. 2017, Ulrich et al. 2009). Additionally, many nestedness measures have bounds that are unconstrained to other network properties. For example, NODF, a popular measure of nestedness, is bounded between 0 and 1. This is problematic because fundamental constraints resulting from the size of the network and the number of links mean that, for many networks, maximum NODF values may be substantially less than 1 (Song et al. 2017). Therefore, comparing the nestedness of different networks using raw nestedness values should be avoided. Instead, it is essential to use nestedness metrics which are independent from network size, connectance and maximum nestedness (Song et al. 2017).

To resolve some of these issues, studies typically express nestedness values relative to a null expectation (for example, Welti & Joern 2015). Specifically, nestedness is expressed as a z-score: $z = (\text{Nestedness} - \mu)/\sigma$, where μ and σ are the mean and standard deviation, respectively, of the nestedness values across an ensemble of networks generated using a particular null model. Problematically, it was recently shown that this method suffers from irrevocable statistical and inconsistency issues (Johnson 1999, Gelman & Stern 2006, Chagnon 2015, Song et al. 2017), prompting the search for an appropriate way to compare the nestedness of different networks. Song et al. (2017) proposed a new normalised nestedness metric, NODF_c, based on the NODF measure: $\text{NODF}_c = \text{NODF}_n /(C \cdot log(S))$, where $\text{NODF}_n = \text{NODF} /max(\text{NODF})$, C is connectance, S is the geometric mean of the number of species in each level of the network (such as plants and pollinators or plants and seed dispersers), NODF is the raw NODF value for the network and $max(\text{NODF})$ is the maximum nestedness of a network with the same number of species and links

75 as the focal network, subject to the constraint that every species has at least one link (Song et al.

76 2017). This new metric does not suffer from the statistical issues associated with z-scores and

77 is thus robust for nestedness comparisons between networks (Song et al. 2017). To demonstrate

78 this, Song et al. (2017) considered the long-standing prediction that networks are more ordered

79 in less predictable environments (Levins 1968, May 1975). Previous studies using raw or z-score

80 normalised nestedness values failed to find unified answers to this question, but by employing the

81 NODF_c metric, Song et al. (2017) were able to confirm a positive associated between nestedness

82 and temperature seasonality. This ability of the NODF_c metric to uncover patterns that previously

83 could not be found is one of the strongest arguments for why it should be widely adopted.

84 While NODF_c is demonstrably a good statistic for nestedness comparisons, more technically,

85 calculating the max(NODF) term in its formula is a non-convex optimisation problem. Non-convex

86 optimisation problems are generally NP-hard; as such there is no known polynomial time algorithm

87 for computing the true maximum (see Appendix S1 where we show that NODF maximisation can

88 be formulated into the class of mixed integer problems, which are also generally NP-hard). For

89 these kinds of problems, heuristic algorithms can be used. While heuristic algorithms are not

90 able to guarantee finding the true optimum, they should at least find solutions close to the true

91 optimum. Widespread adoption of the NODF_c approach is therefore highly dependent on the

92 availability of fast algorithms that can find good solutions for the maximum nestedness of a

93 network.

94 To date, two algorithms for this problem have been proposed. The first was a greedy algorithm

95 by Song et al. (2017) (the 'Song algorithm') that was intuitive and achieved good optima, but was

96 slow when run on large networks (33 minutes for a network with 797 species and 2933 interactions)

97 (Simmons, Hoeppke & Sutherland 2019). This algorithm was refined by Simmons, Hoeppke &

98 Sutherland (2019), who combined the greedy approach with simulated annealing (the 'Simmons

99 old' algorithm). This new algorithm found higher levels of maximum nestedness than the Song

100 algorithm, while reducing computation time by two thirds for large networks (11 minutes for the

101 797 species, 2933 interaction network). However, it was not available in R, which might limit its

use among ecologists, and it was still too slow to be viable for many common types of analysis. For example, analyses with many iterations are often used in network ecology, such as testing the effect of network structure on thousands of extinction simulations, or quantifying the structure of thousands of simulated networks (Thébault & Fontaine 2010, Simmons, Cirtwill, Baker, Wauchope, Dicks, Stouffer & Sutherland 2019). Such analyses would require calculating $NODF_c$ hundreds or thousands of times. If the Song algorithm was used 1000 times on the large network mentioned above, the $NODF_c$ calculations would take 22.9 days. While the 'Simmons old' algorithm reduces this to 7.6 days, this remains impractical. Even if permutational approaches are not necessary, run times of 33 minutes and 11 minutes, for the original and refined algorithms respectively, for a single large network are likely to deter users. Furthermore, if even larger networks are considered, such as the largest network in the Web of Life (www.web-of-life.es) database which has 1500 species and 15255 interactions, neither of these algorithms are likely to be practical for even a single calculation.

Thus while the normalised nestedness metric proposed by Song et al. (2017) is conceptually very robust, to date either slow implementations or suboptimal maxima has made the method impractical for all but simple analyses. However, it is essential that nestedness values are normalised correctly in order to ensure studies make accurate inferences. Here we fill this gap by introducing, `maxnodf`, an `R` package that enables rapid evaluation of $NODF_c$, for the first time making correctly-normalised nestedness values accessible for even complex analyses of large networks. Below we describe the package and its implementation, alongside comparative performance benchmarking and two examples of how it can be applied to empirical data.

We note that while our focus here is on species interaction networks, nestedness is also found in patterns of species occurrence. `maxnodf` is equally applicable to these data.

Description

maxnodf contains three functions: `maxnodf()`, `NODFc()` and `nodf_cpp()`. `maxnodf()` and `NODFc()` are the main two functions and are closely related: `maxnodf()` calculates the maximum NODF of a network, while `NODFc()` calculates the full $NODF_c$ metric as defined in the Introduction. `nodf_cpp()` simply calculates the raw NODF of a network; calculation is fast because the code for this function is implemented in C++.

`maxnodf()` and `NODFc()` have the same two arguments. The first is `web`, which requires a numeric matrix describing a bipartite network (a bipartite incidence matrix where rows represent one group of nodes, columns represent the other group of nodes, and elements are positive numbers if nodes interact, and 0 otherwise). `maxnodf()` alternatively accepts a numeric vector of length 3 of the form `c(nRows, nColumns, nLinks)` for the `web` argument, where each element of the vector gives the number of rows, columns and links in the network for which you want to find the maximum nestedness.

The second argument is `quality`, which is a parameter to control the tradeoff between computation time and result quality; it can be 0 (the default), 1 or 2. Lower-quality settings are faster, but find worse optima; higher-quality settings are slower, but find better optima. `quality = 0` (the fastest) uses a greedy algorithm; `quality = 1`, uses a greedy algorithm plus hillclimbing; and `quality = 2` (the slowest) uses a simulated annealing algorithm. Each algorithm is guaranteed to find an equal- or higher-quality solution than the quality below it. Full details of each of these algorithms are given in Appendix S2. We believe the Quality 0 algorithm is the best choice for most questions, and the sacrifice in quality will rarely change conclusions qualitatively (see below and Analysis S1).

As output, `NODFc()` returns the value of $NODF_c$ as a single number. `maxnodf()` returns a list of length 2, where the first element ('max_nodf') is the maximum nestedness of the network and the second element ('max_nodf_mtx') is the incidence matrix corresponding to this maximum nestedness.

Below is a simple example of calculating $NODFc$ for a small network, at the quality 2 setting:

```
m <- matrix(c(1,1,1,1,0,1,0,1,1), 3, 3) # make a small matrix
NODFc(web = m, quality = 2) # calculate NODFc
```

or calculating the maximum nestedness of a network with 14 rows, 13 columns and 52 links using the default quality:

```
maxnodf(web = c(14, 13, 52))
```

Performance

We recorded the time to compute, and maximum NODF achieved, when calculating max(NODF) using five algorithms: the three algorithms in the `maxnodf` package (quality 0, 1 and 2), the original greedy algorithm proposed by Song et al. (2017) ('Song'), and the refinement of this algorithm proposed by Simmons, Hoeppke & Sutherland (2019) ('Simmons old'). Algorithms were run on all pollination networks from the Web of Life dataset (http://www.web-of-life.es), excluding 13 networks with either more than 3000 links or which were not a single, connected component. These inclusion criteria ensured all algorithms could be run on the entire dataset: the Song algorithm was impractically slow on networks with more than 3000 links, and the algorithms cannot be run on networks which comprise multiple, disconnected subnetworks. This resulted in a dataset of 135 networks. Timings were carried out on a computer with an i7-8550U (1.8 GHz) processor with 16 GB RAM (2133 MHz).

Overall, the variation in runtime was substantial, spanning five orders of magnitude. We find that, on average, the slowest algorithm is the 'Simmons old' algorithm, followed by the Quality 2 algorithm, the original Song algorithm, the Quality 1 algorithm and finally the Quality 0 algorithm, which is the fastest (Figure 1a). This ordering is broadly expected: simulated annealing algorithms, like 'Simmons old' and Quality 2, achieve better optima but are slower, while greedy algorithms,

like Song and Quality 0, achieve worse optima but are faster. The `maxnodf` Quality 2 algorithm is the fastest simulated annealing algorithm, and the Quality 0 algorithm is the fastest greedy algorithm. Notably, the Quality 1 algorithm, which uses greedy and hillclimbing components, would be expected to take an intermediate time, slower than greedy algorithms, but faster than simulated annealing. However, instead we find that Quality 1 is actually faster on average than the Song greedy algorithm, while also achieving better optima (Figure 1a).

The above discussion focuses on the average performance of each algorithm, but average values can mask important patterns. Given that the Quality 0 algorithm will be the most widely used, in Figure 1b, we compare its performance to that of the original Song algorithm. Note that these two algorithms are equivalent: the `maxnodf` version is simply a faster implementation of the Song algorithm. The speed improvement offered by our implementation is substantial, and is greatest for networks with larger numbers of links: for the largest network in our dataset, our algorithm offers a 274 times speed improvement, reducing the computation time from 50 minutes to 11 seconds (Figure 1b). For networks with fewer links, the improvement is still large, becoming increasing less important for the networks with the fewest links. These performance improvements enable complex analyses to use the max(NODF) approach, even for large networks, while this was unlikely to be possible previously.

In terms of maximum NODF achieved, as expected the 'Simmons old' and Quality 2 simulated annealing algorithms perform best, while the Song and Quality 0 greedy algorithms perform worst (Figure 1a). The magnitude of the improvement afforded by the slower algorithms is, however, generally small: on average, the Quality 2 algorithm produces maximum NODF values that are 2.3% higher than those produced by Quality 0. However, again, this average masks some variation, with improvements of up to 17% (Figure 1c).

In Analysis S1, we find that the Quality 2 simulated annealing algorithm and the Quality 0 greedy algorithm produce almost identical results when the maximum nestedness is high, but the simulated annealing algorithm produces increasingly better maxima than the greedy algorithm

when the maximum nestedness is low (Figure S4). Therefore, best practice might involve running all networks in a dataset through the greedy algorithm, and if any of these networks have a maximum nestedness below 0.6 (the value of maximum nestedness where the simulated annealing algorithm improves on the greedy algorithm by more than 5%), these networks should be run through the simulated annealing algorithm.

Thus, while the Quality 0 algorithm will be suitable for most purposes, the higher-quality algorithms are available if only a small number of networks are being studied, or if the most accurate NODF values are needed. We believe that the greedy algorithm is the best choice for most questions and think that the sacrifice in quality would rarely change conclusions qualitatively (Song et al. 2019).
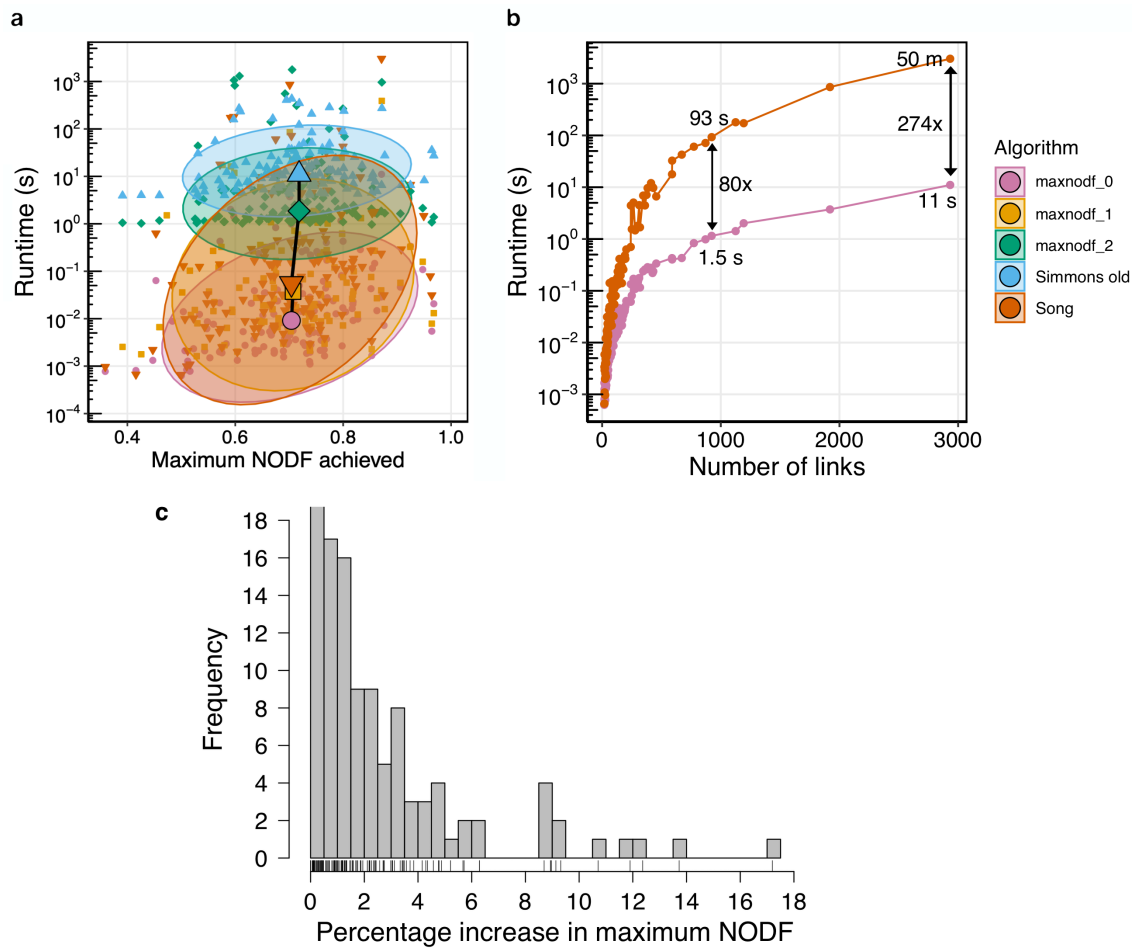
Figure 1: Comparative benchmarking of nestedness maximisation algorithms. **(a)** Comparison of the time taken to run, and maximum NODF achieved, for: (i) the three algorithms in the `maxnodf` package (quality 0, 1 and 2); (ii) the algorithm proposed by Simmons, Hoeppke & Sutherland (2019); and (iii) the original algorithm proposed by Song et al. (2017). All algorithms were run on an identical set of 135 empirical pollination networks from the Web of Life dataset. Small points represent individual networks; large points represent medians. Ellipses represent 95% confidence intervals. **(b)** Comparison of how long the quality 0 greedy algorithm and the original greedy algorithm proposed by Song et al. (2017) take to run on networks with different numbers of links. Network data were the same 135 networks as in (a). Arrows show the time difference between the two algorithms for particular networks. The label of the arrow shows how many times faster the quality 0 algorithm was (e.g. '80x' is 80 times faster), while the numbers at the ends of the arrow show the time each algorithm took to complete for a particular network (e.g. '50m' = 50 minutes, '11s' = 11 seconds.) **(c)** Percentage increase in maximum NODF achieved by the Quality 2 algorithm compared to the Quality 0 algorithm for all 135 networks in the dataset.

Applications

Here we conduct two short analyses. First, we test the importance of normalising nestedness values by comparing normalised values to their raw counterparts. We use pollination networks from the Web of Life dataset (www.web-of-life.es) to determine the level of collinearity between raw NODF and $NODF_c$ values. Specifically, we calculate NODF and $NODF_c$ for all networks and test for a correlation in their ranks. We find no correlation in ranks (Spearman's: ρ = -0.11, S = 831880, P = 0.16; Figure 2a), indicating that the most nested network as measured by raw NODF is not the most nested as measured by $NODF_c$ (and so on). This demonstrates the importance of normalising nestedness correctly — the relative nestedness of networks when normalised is very different to when they are not normalised. Note that an extensive analysis comparing the normalised nestedness values derived with z-scores to those derived using the $NODF_c$ method, for two different null models and networks of different sizes and connectances, is provided in Song et al. (2017). Second, we test how normalising nestedness values could change ecological understanding. We compare how nestedness varies over time when measured using NODF and $NODF_c$. Data was from four plant-pollinator communities in the Seychelles, sampled for eight consecutive months from September 2012 to April 2013 (Kaiser-Bunbury et al. 2017). To measure the stability of nestedness over time, we calculated the coefficient of variation in raw and normalised nestedness for each of the four communities. We found that nestedness appears more stable over time when measured using $NODF_c$ (t = 4.80, df = 3, P = 0.02), suggesting that the macro structure of these communities may be more stable than would be recognised from using raw NODF values.
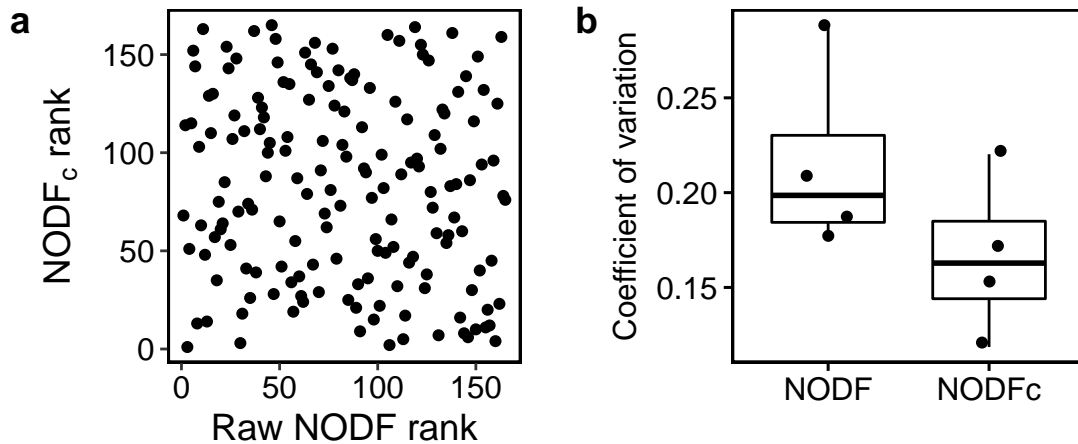
Figure 2: (a) The relationship between the ranks of nestedness values in raw and $NODF_c$ forms for pollination networks from the Web of Life dataset. No correlation in ranks was found. (b) Coefficient of variation of nestedness values in four pollination networks over time, measured using raw NODF and $NODF_c$. Networks were measured over eight months in the Seychelles. $NODF_c$ values indicate that nestedness values are significantly more stable over time than is shown with the raw nestedness values.

Implementation and availability

The `maxnodf` package is available for the `R` programming language. To install the package, run `install.packages ("maxnodf")`. This paper describes version 1.0.0 of the software. The source code of the package is available at https://github.com/christophhoeppke/maxnodf. Any problems can be reported using the Issues system. The code is version controlled with continuous integration and has code coverage of approximately 95%. All code is released under the MIT license.

Conclusions

Nestedness is a pervasive pattern in ecological systems. In particular, nestedness measures have been widely used in studies of mutualistic species interaction networks, but a lack of proper normalisation has limited our ability to make inferences. `maxnodf` is the first package to implement

rapid calculation of the $NODF_c$ metric, a nestedness measure that is demonstrably fair to compare between networks of different sizes and connectances. The package contains three optimised algorithms that allow the user to choose their own trade off between speed and quality. We anticipate that, by making $NODF_c$ calculations feasible for complex analyses and for use with large networks, `maxnodf` will usher in widespread adoption of correctly-normalised nestedness values.

Acknowledgements

Author contributions

BIS conceived the study; CH and BIS developed the algorithms, conducted analyses and wrote the first manuscript draft.

Data availability

All networks used in this study are available from the Web of Life repository (www.web-of-life.es)

References

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* **100**(16), 9383–

262 9387.

263 Bastolla, U., Fortuna, M. A., Pascual-García, A., Ferrera, A., Luque, B. & Bascompte, J. (2009),

264 'The architecture of mutualistic networks minimizes competition and increases biodiversity',

265 *Nature* **458**(7241), 1018–1020.

266 Chagnon, P.-L. (2015), 'Characterizing topology of ecological networks along gradients: the limits

267 of metrics' standardization', *Ecological Complexity* **22**, 36–39.

268 Gelman, A. & Stern, H. (2006), 'The difference between "significant" and "not significant" is not

269 itself statistically significant', *The American Statistician* **60**(4), 328–331.

270 James, A., Pitchford, J. W. & Plank, M. J. (2012), 'Disentangling nestedness from models of

271 ecological complexity', *Nature* **487**(7406), 227–230.

272 Johnson, D. H. (1999), 'The insignificance of statistical significance testing', *The journal of wildlife*

273 *management* pp. 763–772.

274 Kaiser-Bunbury, C. N., Mougal, J., Whittington, A. E., Valentin, T., Gabriel, R., Olesen, J. M.

275 & Blüthgen, N. (2017), 'Ecosystem restoration strengthens pollination network resilience and

276 function', *Nature* **542**(7640), 223–227.

277 Levins, R. (1968), *Evolution in changing environments: some theoretical explorations*, number 2,

278 Princeton University Press.

279 May, R. M. (1975), Stability in ecosystems: some comments, *in* 'Unifying concepts in ecology',

280 Springer, pp. 161–168.

281 Saavedra, S. & Stouffer, D. B. (2013), '"disentangling nestedness" disentangled', *Nature*

282 **500**(7463), E1–E2.

283 Simmons, B. I., Cirtwill, A. R., Baker, N. J., Wauchope, H. S., Dicks, L. V., Stouffer, D. B. &

284 Sutherland, W. J. (2019), 'Motifs in bipartite ecological networks: uncovering indirect interac-

285 tions', *Oikos* **128**(2), 154–170.

Simmons, B. I., Hoeppke, C. & Sutherland, W. J. (2019), 'Beware greedy algorithms', *Journal of Animal Ecology* **88**, 804–807.

Song, C., Rohr, R. P. & Saavedra, S. (2017), 'Why are some plant–pollinator networks more nested than others?', *Journal of Animal Ecology* **86**(6), 1417–1424.

Song, C., Rohr, R. P. & Saavedra, S. (2019), 'Beware z-scores', *Journal of Animal Ecology* **88**(5), 808–809.

Suweis, S., Simini, F., Banavar, J. R. & Maritan, A. (2013), 'Emergence of structural and dynamical properties of ecological mutualistic networks', *Nature* **500**(7463), 449–452.

Thébault, E. & Fontaine, C. (2010), 'Stability of ecological communities and the architecture of mutualistic and trophic networks', *Science* **329**(5993), 853–856.

Ulrich, W., Almeida-Neto, M. & Gotelli, N. J. (2009), 'A consumer's guide to nestedness analysis', *Oikos* **118**(1), 3–17.

Welti, E. A. & Joern, A. (2015), 'Structure of trophic and mutualistic networks across broad environmental gradients', *Ecology and Evolution* **5**(2), 326–334.