

REVIEW

Pattern detection in bipartite networks: A review of terminology, applications, and methods

Zachary P. Neal¹, Annabell Cadieux², Diego Garlaschelli^{3,4,5}, Nicholas J. Gotelli⁶, Fabio Saracco⁷, Tiziano Squartini^{4,5}, Shade T. Shutters⁸, Werner Ulrich⁹, Guanyang Wang¹⁰, Giovanni Strona^{11*}

1 Department of Psychology, Michigan State University, East Lansing, Michigan, United States of America, **2** Mathedoc, <https://mathedoc.de>, Halle, Germany, **3** Lorentz Institute for Theoretical Physics, Leiden University, Leiden, the Netherlands, **4** IMT School for Advanced Studies, Lucca, Italy, **5** The National Institute for Advanced Mathematics, Rome, Italy, **6** Department of Biology, University of Vermont, Burlington, Vermont, United States of America, **7** Enrico Fermi Research Center, Rome, Italy, **8** School of Complex Adaptive Systems, Arizona State University, Tempe, Arizona, United States of America, **9** Faculty of Biological and Veterinary Sciences, Nicolaus Copernicus University, Torun, Poland, **10** Department of Statistics, Rutgers University, Piscataway, New Jersey, United States of America, **11** European Commission, Joint Research Centre, Ispra, Italy

* giovanni.strona@ec.europa.eu



OPEN ACCESS

Citation: Neal ZP, Cadieux A, Garlaschelli D, Gotelli NJ, Saracco F, Squartini T, et al. (2024) Pattern detection in bipartite networks: A review of terminology, applications, and methods. PLOS Complex Syst 1(2): e0000010. <https://doi.org/10.1371/journal.pcsy.0000010>

Editor: Hocine Cherifi, Université de Bourgogne: Université de Bourgogne, FRANCE

Published: October 3, 2024

Copyright: © 2024 Neal et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Funding: ZPN was supported by USA National Science Foundation grants 2016320 and 2211744. DG and TS were supported by EU/NextGenerationEU/PNRR grant IR0000013 'SoBigData.it'. GW was supported by USA National Science Foundation grant 2210849. NJG was supported by USA National Science Foundation grant 2019470. STS was supported by a USA Air Force Office of Scientific Research grant FA9550-21-1-0140. WU was supported by an NCU institutional grant. FS was partially supported by the project "CODE – Coupling Opinion Dynamics with Epidemics", funded under PNRR Mission 4 "Education and Research" - Component C2 - Investment 1.1 – Next Generation EU "Fund for

Abstract

Rectangular association matrices with binary (0/1) entries are a common data structure in many research fields. Examples include ecology, economics, mathematics, physics, psychometrics, and others. Because their columns and rows are associated to distinct entities, these matrices can be equivalently expressed as bipartite networks that, in turn, can be projected onto pairs of unipartite networks. A variety of diversity statistics and network metrics can be used to quantify patterns in these matrices and networks. But, to be defined as such, what should these patterns be compared to? In all of these disciplines, researchers have recognized the necessity of comparing an empirical matrix to a benchmark ensemble of 'null' matrices created by randomizing certain elements of the original data. This common need has nevertheless promoted the independent development of methodologies by researchers who come from different backgrounds and use different terminology. Here, we provide a multidisciplinary review of randomization techniques and null models for matrices representing binary, bipartite networks. We aim at translating concepts from different technical domains to a common language that is accessible to a broad scientific audience. Specifically, after briefly reviewing examples of binary matrix structures encountered across different fields, we introduce the major approaches and strategies for randomizing these matrices. We then explore the details of and performance of specific techniques and discuss their limitations and computational challenges. In particular, we focus on the conceptual importance and implementation of structural constraints on the randomization, such as preserving row and/or columns sums of the original matrix in each of the randomized matrices. Our review serves both as a guide for empiricists in different disciplines, as well as a reference point for researchers working on theoretical and methodological developments in matrix randomization methods.

National Research Program and Projects of “Significant National Interest” PRIN 2022 PNRR, grant code P2022AKRZ9. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Introduction

What do ecological metacommunities, biotic interactions, gene mutations, international trade, public transportation, musical preferences, and organized group events have in common? They all represent systems that we can conveniently synthesize and investigate as bipartite networks [1–7], i.e., networks with nodes of two types, and connections only appearing between nodes of different type. These graphs provide information about the presence or absence of relationships between two entities and/or of the strength of these relationships. Accordingly, numerous examples of application of bipartite networks can be found across multiple research fields such as anthropology, biology, ecology, economics, engineering, finance, logistics, management, mathematics, physics, and social sciences (see Fig 1 and Table 1). Indeed, some have argued that “any complex network [i.e., system] may be viewed as a bipartite graph” [8]. Moreover, various “nominally” unipartite networks (i.e., networks whose nodes are *per se* not separated into two or more distinct sets) have been recently found to display a close-to-bipartite organization, a situation that is understood to arise when most links in the network connect complementary, rather than similar, nodes [9,10]. Indeed, the principle of homophily, which is the one explaining the increased abundance of links between similar nodes in unipartite graphs (and implying that nodes of the same type are tightly interconnected), is not the only possible determinant of connections. The principle of complementarity, stating that complementary nodes connect synergistically when they “need” each other in order to carry out some functionality, predicts that nodes of the same type are not mutually connected in complementary-driven networks, therefore giving rise to bipartite- or multipartite-like structural features. Examples of networks strongly shaped by complementarity are protein–protein interaction networks [10], production networks [11], and some semantic networks [12]. Indeed, it makes

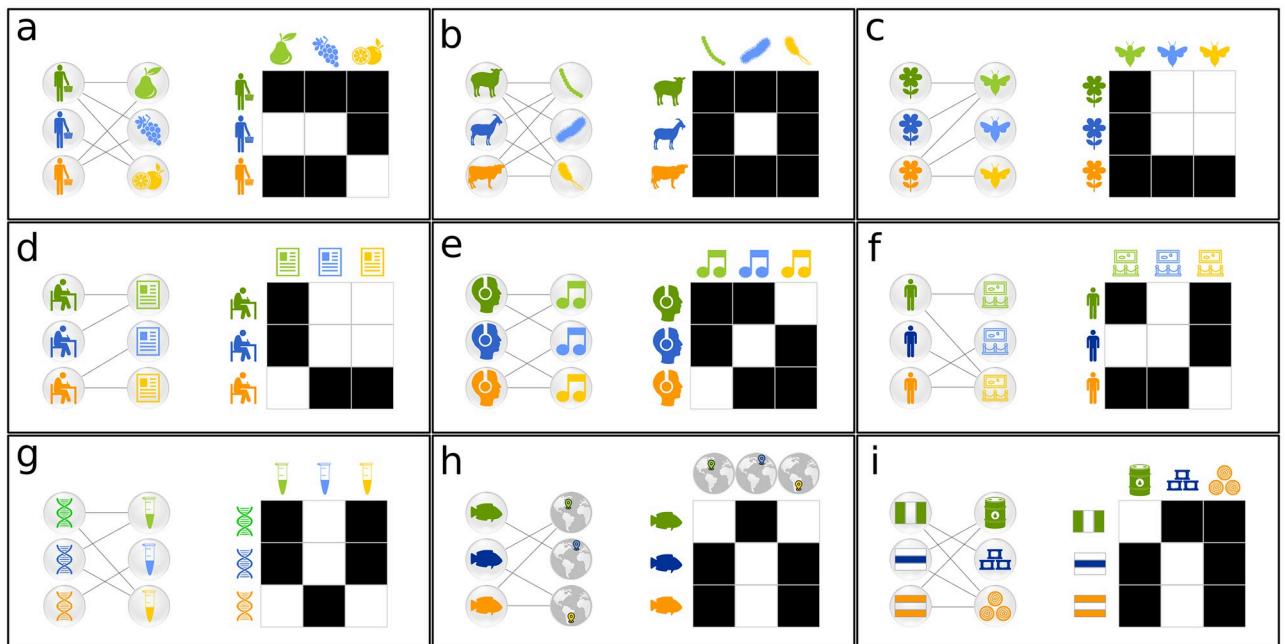


Fig 1. Bipartite networks are ubiquitous in the real world. Examples of systems that can be represented by bipartite networks (left) and their corresponding binary matrix representation (right). Black/gray cells in each matrix indicate presence of links (i.e., 1s) between the items in rows and the items in columns, while white cells indicate the absence of links (i.e., 0s). The networks link, respectively: (a) buyers to purchases; (b) ruminants to associated microbiota; (c) plants to pollinators; (d) authors to articles; (e) listeners to songs; (f) visitors to exhibitions; (g) genes to samples; (h) species occurrences to localities; and (i) countries to exported commodities.

<https://doi.org/10.1371/journal.pcsy.0000010.g001>

Table 1. Examples of bipartite networks and their applications in the real world.

Field	Subfield	System/application	Example Reference (DOI)
Biology	Biomedical bipartite networks	Drug–target interaction networks	10.1093/bioinformatics/btaa157
Biology	Genetics	Gene–sample binary mutation matrices	10.1038/ng.3557 ; 10.1038/ng.3168
Biology	Genetics	Gene–tissue networks	10.1101/517565
Biology	Molecular biology	Transcription factors	10.1126/science.aam8940
Data Science	Innovation	Patents–technological codes networks	10.1371/journal.pone.0230107
Ecology	Archeozoology	Species–locality matrices	10.1016/j.jasrep.2015.02.008
Ecology	Ecological networks	Cleaning symbiosis	10.1098/rsbl.2006.0562
Ecology	Ecological networks	Dung beetle–mammal networks	10.1111/ele.13095
Ecology	Ecological networks	Host–microbiota networks	10.1111/1365-2656.13297
Ecology	Ecological networks	Host–parasite networks	10.1038/ncomms12462
Ecology	Ecological networks	Plant–herbivore networks	10.1371/journal.pone.0052967
Ecology	Ecological networks	Mutualistic networks	10.1111/j.1461-0248.2007.01061.x
Ecology	Infectious disease ecology	Dbl α types vs. isolates	10.1002/ece3.3803
Ecology	Meta-community ecology	Species–locality matrices	10.1007/BF00317508
Economics	E-commerce	Buyers–products purchase networks	20.500/12469/2899
Economics	World trade web	Country–commodity networks	10.1038/srep10595
Finance	Cryptocurrency	Bitcoin lightning network	10.1088/1367-2630/aba062
Finance	Interbank networks	Exposure (borrower–lender) networks	10.1038/srep03357
Mathematics	Combinatorics/graph theory/matrix theory	Network enumeration	10.1016/0890-5401(89)90067-9
Physics	Social/ecological/economic/financial networks	Network reconstruction	10.1088/1367-2630/16/4/043022
Physics	Transportation networks	Bus route–bus stop networks	10.1016/j.physa.2006.10.071
Physics	Transportation networks	Ferry vehicle–flight networks	10.1016/j.omega.2019.102178
Social Sciences	Blogger communities	Users–posts networks	10.1016/j.physa.2012.06.004
Social Sciences	Human behavior	Listeners–music groups	10.1103/PhysRevE.72.066107
Social Sciences	Contact networks	Event–participation networks	10.1371/journal.pone.0171565
Social Sciences	Political science	Bill cosponsorship	10.2478/connections-2019.026

<https://doi.org/10.1371/journal.pcsy.0000010.t001>

sense to conjecture that companies and proteins connect to each other more likely if they need each other, hence because they are different/complementary, rather than similar.

Many contexts benefit from directly analyzing bipartite networks. Additionally, when a bipartite network's two disjoint sets of nodes represent distinct types of entities (as they do in Fig 1), the system is also known as a two-mode network, and it can be used to generate two one-mode networks, each containing only nodes of the same type, via the following projection: pairs of same-type nodes that, in the original two-mode network, are connected to common node(s) in the other layer become connected to each other in the projected one-mode network. As with bipartite networks, analyses of one-mode projections are commonly used in multiple fields; e.g., in bibliometrics, a bipartite network connecting scientific publications to their authors can be projected onto a coauthorship network synthesizing scientific collaborations [13]. Other common examples of projections of bipartite network include those mapping legislative collaboration through bill cosponsorship in political science [14], gene interactions through sample coexpression in biology [15], and bacterial interactions through sample co-occurrence in microbial ecology [16] (see Fig 2). Actually, any matrix/network mapping the simultaneous presence (i.e., co-occurrence) of items, organisms, or events in space and/or time can be considered a bipartite network projection of the ideal bipartite network linking items/organisms/events to different localities and/or times [17–19].

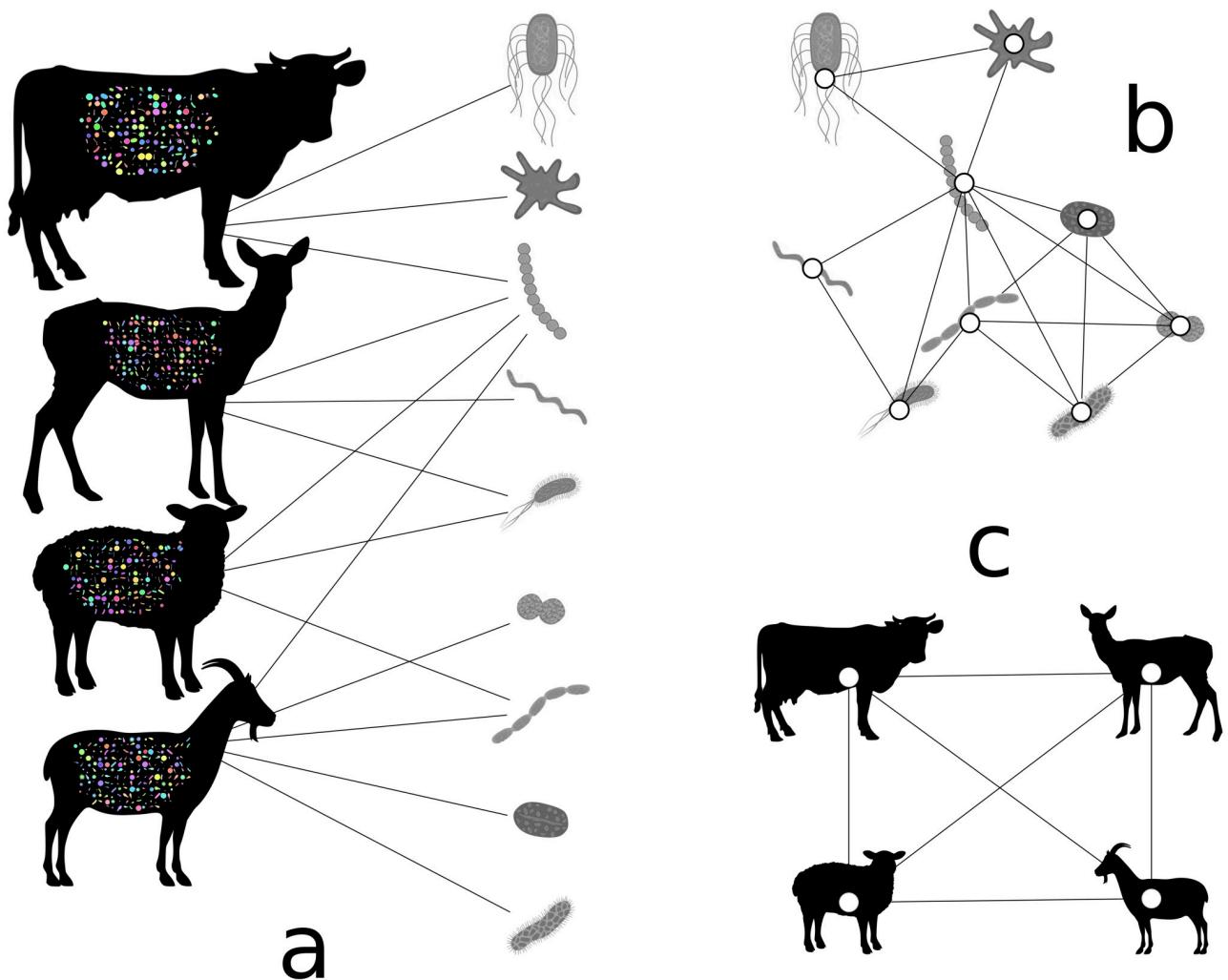


Fig 2. Bipartite networks and their projections. A hypothetical bipartite network connecting different ruminants to associated microorganisms (a), and its two one-mode projections. One projection (b) connects all the microorganisms that are found together in at least one host. The other projection (c) connects all the ruminants sharing at least one microorganism, generating a fully connected network in this example.

<https://doi.org/10.1371/journal.pcsy.0000010.g002>

The ubiquity of bipartite networks and their projections has resulted in a considerable amount of theoretical and applied knowledge. On the other hand, their vast interdisciplinary span has prevented the convergence of knowledge into an organic corpus. One critical task where this lack of convergence presents significant barriers is the detection of patterns in bipartite networks [20]. While there is widespread consensus that a *pattern* should be understood as a statistically significant structural feature that, to be identified, requires a comparison with a *null model* or randomized benchmark, different research fields have developed their own (sometimes duplicate) methods for this task, using field-specific terminology and summarizing their findings in specialized reviews (e.g., in ecology [1], social [21] and computer science [22], and complex systems [23]). For example, one widely used method is known as “stochastic degree sequence model” in social science [21], but as a “canonical configuration model” in statistical physics [24–26]. Despite the widespread applicability of these methods, their strong intradisciplinary focus has hindered progress, often forcing individual fields to rediscover methods without fully benefiting from innovations in other areas.

To confront and overcome these challenges, this review provides a comprehensive, multidisciplinary synthesis of available knowledge on bipartite network null-model definition and randomization techniques, with the ultimate aim to enhance progress and prevent the wheel from being reinvented multiple times. It is important to note that the techniques described here serve as null models and/or randomization methods for both bipartite networks and their unipartite projections, for the following reason. In order to identify patterns in an empirical bipartite network, one should obviously randomize the network while preserving its bipartite nature, and this is what the null models described here can do. Much in the same way, in order to identify patterns in one-mode projections of a bipartite network, one should first randomize the original bipartite graph (again preserving its bipartiteness), then generate the unipartite projection for each of the randomized bipartite variants, and finally compare the ensemble of these unipartite projections with the empirical unipartite projection. Indeed, randomizing a one-mode projection directly (as if it were a genuinely unipartite network) would in general violate its compatibility with a “parent” bipartite network. Therefore, patterns in the unipartite projection can be effectively regarded as patterns of the original bipartite graph, because they are ultimately (possibly complicated) functions of the topology of the bipartite graph and, moreover, should “survive” the same bipartite randomization procedure that all inherently bipartite properties undergo in order to be identified as patterns.

As a useful preliminary clarification, we note that, given the multidisciplinary character of research on bipartite networks and their projections, there are many different patterns that researchers may seek to detect, as well as many potential uses they may want to make of those patterns once identified. Examples of bipartite patterns include the following: the so-called ν -motifs [3,21] (quantifying co-occurrence or common interactions for nodes in one layer, based on their connections to the other layer), communities [27,28] (representing pairs of densely connected sets of nodes across the two layers), nestedness [29,30] (quantifying the degree of “triangularity” of a bipartite adjacency matrix), and the abundance of 4-cycles or quadrangles [10,31] (while homophily in unipartite networks gives rise to many triangles, complementarity in bipartite graphs gives rise to many quads). Examples of possible uses of such patterns are the following: purely descriptive (characterizing and possibly classifying different bipartite networks in terms of their empirical properties); predictive (e.g., does nestedness affect network stability under species removal, and can it be used to rank nodes in terms of the impact they would have if removed? [32]); and inferential (e.g., can 4-cycles enhance the imputation of missing links, based on the hypothesis that two nodes are more likely to be connected if their neighbors are mutually connected? [31]). In this paper, we are not focused on any specific type of pattern or potential use of it, but instead review the methods available for detecting any pattern in bipartite networks or their projections through comparison of an observed network to a null model obtained via randomization.

To tackle these aims in a way that is hopefully accessible to a broad audience, we provide conceptual illustrations and concrete examples to identify overlaps and differences between the approaches developed in the various fields of science, limiting the use of formal notation to specific technical sections dedicated to readers interested in the formal specifications of the models we discuss. To keep focus and simplicity, we deal with presence/absence (0/1) Boolean matrices only.

What are bipartite networks?

As with any network, a bipartite network is composed of a set of *nodes*, pairs of which may be connected by *edges*. The essential feature of a bipartite network is that these nodes can be partitioned into two sets such that edges exist only between these sets. This feature is clear in each

panel of [Fig 1](#), where edges connect nodes in the left row (e.g., panel 1, buyers) to nodes in the right row (e.g., panel 1, purchases), but never to nodes in the same row.

A bipartite network can be represented as a graph, where nodes are drawn as dots and edges are drawn as lines connecting them (left half of each panel in [Fig 1](#)). It can also be represented as a binary matrix where rows correspond to one set of nodes, columns correspond to the other set of nodes, and entries contain a 1 if the row-node is connected to the column-node (right half of each panel in [Fig 1](#)). Graph representations are often useful for visualization, while matrix representations are more useful for formal analysis.

Whether represented as a graph or matrix, bipartite networks can be characterized by several properties. Here, we focus on two properties that play a particularly important role in the pattern detection methods we discuss below. First, a bipartite network can be characterized by its *density*, which is the fraction of possible edges that are present. In its matrix representation, the density is simply the proportion of filled cells (i.e., cells with entry 1). For example, the density of the bipartite network shown in [Fig 1a](#) is .66 because 6 of a possible 9 edges are present. Second, a bipartite network can be characterized by its nodes' degree sequences, which capture each node's number of connections. In its matrix representation, the degree sequences are given by the matrix's row and column sums. For example, the degree sequence for the row nodes in [Fig 1a](#) is {3,1,2}, while the degree sequence for its column nodes is {2,2,2}.

The defining feature of a bipartite network compared to a general network is its partitionability into two sets of nodes, which are only connected in between, but not inside the sets. However, in real-world bipartite networks, those two sets of nodes often represent distinctly different types of entities, in which case, as we mentioned, the bipartite network is also called a two-mode network. For example, the nodes in the bipartite network shown in [Fig 2a](#) represent distinctly different types of entities: ruminants on the left, and microorganisms on the right. A two-mode network can be analyzed as a bipartite network per se; however, it can also be transformed into two one-mode (unipartite) networks via projection, each consisting of the nodes of only one mode." An edge between two nodes in a projection exists if and only if these nodes are connected to at least one common node of the other mode in the two-mode network. For example, [Fig 2c](#) illustrates that the cow and deer are connected in a one-mode projection because they are both connected to the same long worm-shaped microorganism in the two-mode network. The square matrix representation of each one-mode projection (e.g., the one linking microorganisms occurring in the same ruminant, [Fig 2b](#); and the one linking ruminants sharing microorganisms, [Fig 2c](#)) is the product of the two-mode network's rectangular matrix representation with its transpose and vice versa.

Where are bipartite networks found?

One primary focus in the analysis of bipartite networks and one-mode projections is pattern detection. Before turning to the different methods, for the sake of concreteness, we briefly review the range of contexts where such networks and patterns are observed.

Ecology and biogeography. Employing null models of binary matrices for pattern detection has a long history in ecology [33] and became particularly important in the context of the ongoing debate on how species interactions, particularly competition, determine the spatial distribution of species [34–37]. In this context, the distribution of species across a set of localities is represented as a bipartite network where the species (one set of nodes) are connected to the localities (the other set of nodes) where they are found at a given point in time. Comparing an observed bipartite network of species location to a set of randomized versions of the same network has allowed researchers to investigate structural patterns in both species–locality matrices and bipartite plant–pollinator ecological networks [30,38–41]. Additionally, a

bipartite species location network can be projected into a unipartite species colocation network, where similar methods allow researchers to evaluate whether pairs of species are consistently found in the same locations [1]. This topic has received a lot of interest in recent years. In fact, it was at the center of a lively scientific debate on the possibility of inferring biotic interactions (and, hence, deriving ecological networks of interacting species) from species co-occurrence data [42,43], based on the idea that ecological interactions might play a fundamental role in determining overlapping (or segregated) species distribution patterns.

Social sciences. In the social sciences, bipartite networks are frequently used to represent individuals' (one set of nodes) affiliations or preferences with objects (the other set of nodes). For example, in a sociological context, they can represent individuals' membership in clubs [44], while in a political science context, they can represent legislators sponsorship of bills [21]. In rating systems, individuals express preferences toward items in the other layer [45]. Limited extensions of classical network analytic techniques make it possible to describe and analyze patterns in social bipartite networks [46,47]; however, it is more common for social scientists to study one-mode projections. The one-mode projection of a person–club bipartite network yields a network of individuals connected by their club comemberships, while the one-mode projection of a legislator–bill bipartite network yields a network of legislators connected by their bill cosponsorships. Pattern detection methods can be employed to determine when a dyad's number of comemberships or cosponsorships exceeds what would be expected by chance and, therefore, can be treated as a proxy for an unobserved relationship of interest such as friendship or collaboration [48].

Psychology. In psychology, data capturing individuals' responses to psychological survey items can be represented as a bipartite network. In such a network, the respondents serve as one set of nodes, while the items serve as the other set of nodes, with each respondent nodes connected to each item nodes by an edge weighted with the response, often arbitrary ordinal (Likert) values. In psychometrics, such data are frequently analyzed using a dichotomous Rasch model to construct and score educational and psychological tests [49,50]. Estimation of a Rasch model involves identifying patterns in the bipartite network by comparing it to a series of randomized alternatives [51]. More recently, psychologists have also explored the use of one-mode projections of bipartite clinical data, generating networks of symptom co-occurrence or comorbidity [52], which requires determining when such co-occurrence patterns exceed what would be expected by chance and relies on methods that remain subject to debate [53].

Economics. Bipartite networks have found broad application in economics where they can represent products produced by countries [54–60], financial entities exposed to specific assets [61,62], skills required by occupations [63,64], location of industries in cities [65,66], occupations [67,68], or patent technologies [69,70]. Analysis of such data often focuses on the one-mode projection of these networks. For example, a growing branch of research known as “Economic Complexity” has recently focused on identifying the productive capabilities of the various countries through the analysis of their exported products [56,57]. As in other cases, such analysis rests on determining when patterns in countries' exports of a goods exceed random levels, taking into account such factors as the good's rarity.

How are patterns detected in bipartite networks?

Detecting nonrandom patterns in bipartite networks (i.e., patterns that are not likely to be seen by chance in a set of networks with certain properties) follows a procedure that, at least conceptually, is fairly simple (Fig 3). However, as we will discuss extensively below, several difficulties can arise in the rigorous definition and/or implementation of the simple idea.

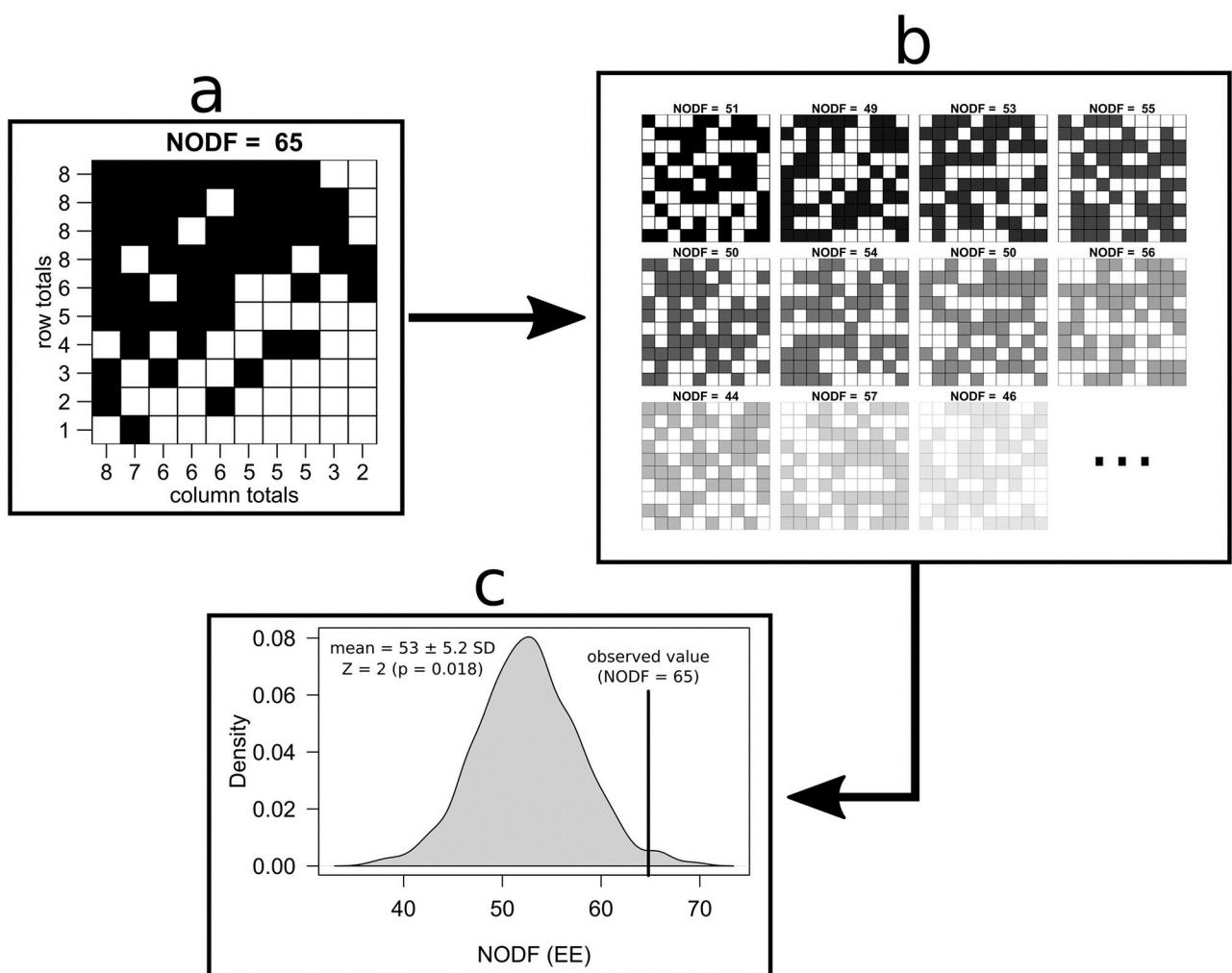


Fig 3. Pattern detection in bipartite networks. Schematic example of how matrix randomization is used to detect nonrandom patterns in bipartite networks/rectangular matrices. Black/gray cells in each matrix indicate presence of links (i.e., 1s) between the items in rows and the items in columns, while white cells indicate the absence of links (i.e., 0s). First, the structural measure of interest (in this case, a nestedness metric, NODF [71]) is computed on the target matrix (a). Then, a large set (ideally some hundreds or thousands) of randomized versions of the starting matrix are generated, and the target metric is computed for each of them (b). The possible rules to be applied in the generation of the random matrices, the reasoning behind, and the implications of choosing a particular set of rules over another, as well as the practical implementation of randomization procedures will be described in detail in the sections. The target metric computed on the original matrix will be then compared with the distribution of the metric values computed on the random matrices. Such comparison would permit to obtain an estimated *p*-value computed as the frequency of random matrices for which the target metric is equal or higher than that of the original matrix. In some fields, and particularly in ecology [29], it is also common practice to compute a standardized effect size (Z) as $(\mu - x) / \sigma$, where μ and σ are the average and standard deviation of the target metric across the randomized matrices, and x is the value of the metric in the original matrix. It should be noted that the use of Z values is based on the underlying assumption that the distribution of the target metric values in the set of randomized matrices follows a normal distribution, which might not be always the case.

<https://doi.org/10.1371/journal.pcsy.0000010.g003>

First, a statistic of interest is computed from an observed bipartite network. The specific statistic depends entirely on the substantive research question. For example, in ecology, the compositional change among communities (β -diversity) has been quantified by a dozen of different measures, each focusing on different aspects of change [72]. In political science, one may explore the structure of cosponsorship networks in terms of number of bills cosponsored by two legislators [21].

Second, a random network is generated. For that, the observed bipartite network is randomized (we discuss how in section “Randomization algorithms and procedures”) in a way

that preserves certain features of the original network (we discuss which ones in section “Bipartite null models—What characteristics should be preserved”). This new, random bipartite network arises from a “null model,” so called because it implements a pattern-free null hypothesis, i.e., the pattern of interest, if present in the original network, should be nullified by the randomization process leading to the null model. The feature being preserved while randomizing the network (technically, the constraints of the null model) in fact define how a random pattern should statistically look like in the given context. For a given situation, prior work and experience provide an indication of the pattern(s) of interest and, consequently, also about the corresponding null expectation. This expectation restricts the null space and the null model has to account for these constraints.

Third, the statistic of interest is computed in the random bipartite network. The second and third steps are performed repeatedly, yielding a distribution of the statistic of interest observed in a set of random bipartite networks (i.e., under the null model). The set of random bipartite networks is known as an ensemble, and each randomly generated network can be viewed as a random sample from this ensemble.

Finally, the statistic of interest from the observed network is compared to its distribution under the null model. Of particular interest is the proportion of times the statistic of interest under the null model is greater than or equal to the statistic of interest from the observed network. For example, observing a proportion of 0.02 would indicate that only 2% of the random networks produced a statistic of interest that is larger than that from the observed network. This proportion is known as a *p*-value and can be used in hypothesis testing concerning the randomness of the pattern captured by the statistic. In this example, using a conventional threshold of statistical significance, such as $p < 0.05$, one would reject the null hypothesis that the pattern measured by the statistic of interest is random and would instead conclude that the pattern is nonrandom.

Bipartite null models

As we mentioned, detecting patterns in bipartite networks involves comparing an observed network to an ensemble of random networks. However, there are multiple ways to conceptualize and realize a “random” ensemble, depending on which constraints are chosen and how they are implemented. Consequently, one can end up in several ensembles and null models. In this section, we focus on null model choice and describe two features of the null model that are particularly important: which characteristics of the observed network are preserved in the ensemble of random networks, and how this ensemble can be generated.

What characteristics should be preserved? Bipartite null models are primarily defined by which characteristics of the original bipartite network are preserved in the randomly generated bipartite ensemble. Note that, in principle, it is possible to imagine a null model that does not contain any of the original network’s characteristic. In this case, one generates an ensemble of bipartite networks, uniformly drawn from the set of all possible bipartite networks. However, for practical but also scientific reasons, it makes more sense to generate an ensemble as a subset of bipartite networks, which contain some characteristics of the original one, e.g., the dimension (number of nodes), the fill (density of edges), and/or one or both of the matrix margins (node degrees). In section “Choosing a null model”, we discuss in detail what sort of theoretical and practical considerations can guide the choice of the constraints to be enforced. Here, we keep this discussion to a minimum, only in order to arrive at the description of the main methods that, for a given choice of constraints, have been proposed.

First, null models can differ in the way constraints are imposed. The constraints can be “hard,” i.e., such that, on each of the matrices of the ensemble, the values of the constrained

quantities match the ones measured on the original network *exactly*. We will denote hard constraints also as “fixed” (and label them as F). Otherwise, constraints can be “soft,” i.e., such that the constrained quantities in each matrix of the ensemble do not necessarily match the values observed in the original network, but their *averages over the ensemble of matrices* do. For reasons that will be clearer later, we may denote soft constraints also as “proportional” (and label them as P). In statistical physics, ensembles with hard constraints are called “microcanonical,” while ensembles with soft constraints are called “canonical” [23,25]. Importantly, canonical and microcanonical ensembles enforcing a given constraint in a hard and soft way, respectively, can asymptotically be either equivalent or inequivalent to each other, depending on the nature of the constraint itself [73–76]. This has implications for the choice of the method, as we discuss later on in section “Choosing a null model”. In general, all null models considered in this review have hard constraints on the network’s dimensions, i.e., they require that each random bipartite network in the ensemble has the same dimensions (i.e., the same numbers of nodes of each type) as the original network. For example, if the original network’s matrix representation has 5 rows and 10 columns, then all random networks generated under a null model will also have 5 rows and 10 columns. On the other hand, all other constrained properties can be either hard or soft; e.g., depending on the models (see below), the network’s fill is replicated either *exactly*, as a hard constraint, or *on average*, as a soft constraint.

Second, null models can vary in the constraints they impose on the network’s row and column marginals (i.e., the degrees of the row and column nodes). Marginals can be unconstrained (such that the marginals in the randomly generated networks do not necessarily match those in the original network) or constrained, and, in the latter case, they can be enforced either softly or hardly. Note that constraining *exactly* the marginal totals results in constraining *exactly* also the matrix fill; analogously, constraining *on average* the marginal totals results in constraining *on average* also the matrix fill. In section “Choosing a null model”, we discuss how to decide which margins to preserve, given the scientific question at hand.

Fig 4 illustrates how the combination of the two ingredients discussed above generates different null models, which we will denote with specific names. For example, the highly constrained null model described by lower-left cell in the right panel of **Fig 4** (which we can denote as “Fixed-Fixed,” or FF) requires that every random network in the ensemble has both row and column marginals that exactly match those in the original. This null model is sometimes called the Fixed Degree Sequence Model [22] or the microcanonical version of the Bipartite Configuration Model [3,30,75]. The somewhat less constrained null model in the central matrix of the right panel of **Fig 4** (which we can denote as “Proportional-Proportional,” or PP) requires that the row and column marginals of the random networks match those in the original only *on average*. This null model is sometimes called the Stochastic Degree Sequence Model [21] or the canonical Bipartite Configuration Model [3,30,75] and relies on an ensemble that is canonical. While many other null models are possible, these two have become the most widely used. Other combinations, discussed in more detail below, are obtained depending on how the column and row margins are treated: as hard constraints (“fixed,” or “microcanonical”), soft constraints (“proportional,” or “canonical”), or as unconstrained (“equiprobable”).

How are random bipartite networks generated? Two broad approaches exist for generating random bipartite networks: fill methods and swap methods. Fill methods begin with an empty matrix with a fixed number of rows and columns (fixed number of nodes for two modes) and incrementally add 0s and 1s as entries (edges between nodes). For example, one microcanonical implementation of the configuration model [77] begins with an empty matrix and has the condition to fill a certain fixed number of 1s in each row and column. This version of the configuration model aims at providing a way to generate a network that satisfies the constraints of FF .

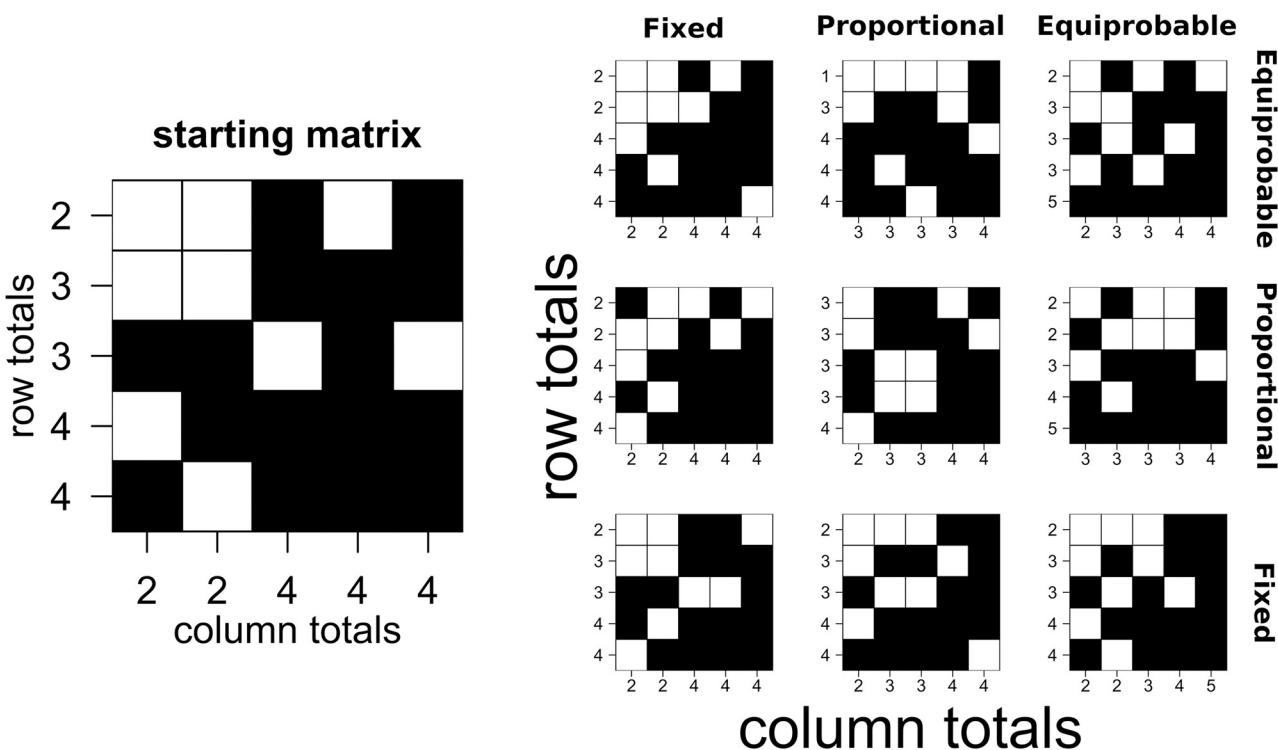


Fig 4. A classification scheme of bipartite randomization algorithms, based on whether the matrix row and columns sums are preserved exactly (Fixed, F), preserved on average (Proportional, P), or unconstrained (Equiprobable, E). Black cells in each matrix indicate presence of links (i.e., 1s) between the items in rows and the items in columns, while white cells indicate the absence of links (i.e., 0s).

<https://doi.org/10.1371/journal.pcsy.0000010.g004>

uniformly at random. We will describe various implementations of the *FF* model, along with their complications in the next section. In contrast, the matrices that satisfy the constraints described by the *PP* model, where the column and row sums are fixed on average, can be generated by assigning each entry M_{ij}^* an appropriate Bernoulli probability p_{ij} . A Bernoulli trial for each entry gives then the decision if the entry is 1 (with probability p_{ij}) or 0 (with probability $1 - p_{ij}$). We will describe various implementations of the *PP* model in the next section.

Swap methods begin with a network (often an observed bipartite network) and swap the nodes (of one mode) from two randomly chosen edges, but only when the new possible edges do not already exist. In this case, a swap is not possible. In the matrix perspective, the algorithm starts with the existing matrix and swaps so-called “checkerboards” (e.g., swapping $\begin{matrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{matrix}$ with $\begin{matrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{matrix}$). This approach was invented by Ryser in 1957 [78]. More advanced algorithms, known as Curveball algorithms, perform multiple swaps from two nodes (two rows) simultaneously [79,80]. They were proven to be at least as efficient as the Ryser version [81]. In practice, they perform often much more efficiently than the classical version of Ryser [82].

In the next section, we will show in detail how binary matrices can be randomized maintaining the constraints summarized in Fig 4. That section is intended for readers interested in getting a better understanding of the technical aspects behind matrix randomization and could serve as a “cookbook” for a coding implementation of the various algorithms (or as a roadmap to help navigating the many implementations that are already available in several programming languages). Noninterested readers can comfortably skip to section 4.

Randomization algorithms and procedures

Basic definitions and notation. We denote a binary (r, c) -matrix as M , where r is the number of rows and c the number of columns. The size, or dimension, S of M is defined as $S = r \times c$. In two-mode networks, rows and columns correspond to distinct sets of real-world entities: for example, each row can be thought as representing an insect species, while each column can be thought as representing a plant species.

The entry (or cell) M_{ij} of matrix M can have either value 0 or 1: $M_{ij} = 1$ indicates that the entity in the i -th row r_i has some kind of association with the entity in the j -th column c_j (for example, the i -th insect pollinates the j -th plant); $M_{ij} = 0$ indicates that no association exists (or has been observed) between the entities in the i -th row and j -th column. We will often refer to the 1s as “presences,” “occurrences,” or “filled cells” (with identical meaning) and to the 0s as “absences” or “empty cells.” The total number of occurrences (i.e., the number of 1s) in the i -th row and in the j -th column are denoted as $r_i = \sum_{j=1}^c M_{ij}$ and $c_j = \sum_{i=1}^r M_{ij}$, respectively. We will denote instances where $r_i = 0$ as “empty rows” and instances where $c_j = 0$ as “empty columns.” Similarly, we will denote a matrix where all M_{ij} entries are equal to 0 an “empty matrix.” We refer to the two sets of row and column totals as, respectively, $R = \{r_1, \dots, r_r\}$ and $C = \{c_1, \dots, c_c\}$. We denote the total number of occurrences in the matrix as $N = \sum_{i=1}^r r_i = \sum_{j=1}^c c_j$ and matrix fill as $f = N/S$.

A binary matrix M defined as above is equivalent to a bipartite network. In a bipartite network, we can identify two distinct sets of nodes, which correspond to the two sets of real-world entities (e.g., plants and pollinators) identified by M 's rows and columns. Thus, the number of elements in the first set of nodes (e.g., plants) is equal to r and the number of elements in the latter set of nodes (e.g., pollinators) is equal to c . The marginal totals of M correspond to the so-called *degrees* of the nodes in the bipartite network; e.g., r_i indicates the degree of the i -th node in the first set of nodes (e.g., the number of pollinators associated to the i -th plant), while c_j corresponds to the degree of the j -th node in the second set of nodes (e.g., the number of plants associated to the j -th pollinator). Each entry for which $M_{ij} = 1$ corresponds to an edge (or “link”) in the bipartite network connecting the i -th node in the first set of nodes (e.g., a plant species) to the j -th node in the latter set (i.e., a pollinator species). Thus, N will also correspond to the total number of edges in the bipartite network. In some scientific fields, the matrix M is called the biadjacency matrix of the network.

We will refer to a single, randomized version of M (i.e., a new matrix obtained as the output of a given sampling algorithm) as M^* (note that, in the statistical physics literature [23–26,48], the notation is usually the opposite, since the asterisk is used to denote the single empirical matrix, while the randomized matrices are left without an asterisk). Each null model produces a set $\{M^*\}$ of possible randomized versions of M , each of which is one possible outcome of the method. A given null model will then define a probability distribution $P(M^*)$ (which may be computable or not) over the set $\{M^*\}$. In practice, one will use the model either to explicitly sample a sufficiently large subset of all the possible randomized matrices (and then compute expected matrix properties as sample averages over this subset) or to compute expectation values analytically over the entire ensemble, if $P(M^*)$ is known and sufficiently simple to work with.

The randomization algorithms we consider can preserve, in different ways, the row and/or column sums of the original matrix M . In the procedure that we denote as F , the row and/or column sums of the real observed matrix are preserved exactly under the randomization. In the procedure that we denote as P , the row and/or column sums in the randomized matrices match only on average (i.e., as an average over the generated set of randomized matrices) those of the original matrix. Finally, in the procedure that we denote as E , the row and/or column sums of the randomized matrices are unconstrained and, hence, to a large extent independent of those of the focal matrix.

To add more to the zoology of possible randomization algorithms, we note that some of them apply to the row and column sums different choices of the procedures F , P , and E . For this reason, we will describe the nature of the constraints of a given algorithm using the notation XY , where X (respectively, Y) indicates the procedure applied to the row (respectively, column) sums of the original matrix. Both X and Y can take any of the three values F , P , and E , hence producing the 9 possible cases illustrated in Fig 4. These algorithms have been implemented in multiple packages across different programming languages. To help readers navigate the options, we have compiled a nonexhaustive table (that we plan to update dynamically) listing R and $Python$ packages and scripts implementing specific randomization procedures. The table can be accessed at https://github.com/giovannistrona/bipartite_randomization_review.

Unconstrained rows, Unconstrained columns (EE). Method EE is the most trivial method out of our 9 possible ones, which requires to leave both margins of the matrix unconstrained. If this were the only prescription, technically the resulting ensemble would be entirely uniform, i.e., each of the 2^S Boolean matrices of dimension S would be assigned the same probability 2^{-S} , irrespective of any property of the empirical matrix. A more informative, yet still quite unstructured, alternative is that of leaving the two margins unconstrained (so we can still denote the model as EE), while specifying only the overall fill f as a constraint. As for all other properties, the total fill can be in principle enforced either as a hard constraint or as a soft one.

If the fill f is treated as a hard constraint, the resulting ensemble contains all possible $\binom{S}{N}$ matrices with fixed size S and fixed fill $f = N/S$ each taken with the same probability. This is sometimes called the microcanonical Bipartite Random Graph Model [3,75]. Sampling uniformly from this ensemble can be achieved by a variety of different approaches. In the class of swap methods, an efficient recipe is of that of exchanging entries 0s and 1s in the initial matrix in the following way. All entries M_{ij} of matrix M get a different number, starting from 1 to S (the matrix size). These numbers can be permuted uniformly by a classical random permutation algorithm to get a new ordering of the numbers, and, hence, their corresponding entries M_{ij}^* . Let, for example, be a (2,2)-matrix with entries $M_{11} = 0$, $M_{12} = 1$, $M_{21} = 1$, $M_{22} = 1$. M_{11} , M_{12} , M_{21} , M_{22} get the numbers 1, 2, 3, 4 in this order. We randomly permute the numbers and get the new order 2, 4, 3, 1. Then, the new matrix M^* has the following entries: $M_{11}^* = 1$, $M_{12}^* = 1$, $M_{21}^* = 1$, $M_{22}^* = 0$. Classical random permutation algorithms come from mixing card decks and are known as random shuffles [83]. They all have efficient running times. Note, that the number of all possible Boolean matrices of this class is $\binom{S}{N}$. EE sampling can be also performed using filling approaches. There, we consider an empty (r, c) -matrix M^* , with $M_{ij}^* = 0$ for all i and j . Our goal is to fill that matrix with N 1s and $(S-N)$ 0s. These values are taken from the initial matrix M . We give all entries M_{ij}^* a different number from 1 to S . N times we use one after another a classical random number generator to first choose a number uniformly at random and then delete it from the set of numbers. The corresponding entries M_{ij}^* of all chosen numbers are set to 1. The remaining entries to 0. Classical random number generators are efficient and can be found, for example, in [84].

If the fill is enforced as a soft constraint, the resulting ensemble contains again all the possible 2^S Boolean matrices of dimension S , however, with the requirement that the expected fill equals the desired (empirical) value $f = N/S$. This is sometimes called the canonical Bipartite Random Graph Model [3,75]. The probability distribution over matrices in the ensemble is obtained by maximizing the Shannon–Gibbs entropy [25] under the average constraint on the overall fill, and the result is a model where all the entries of the matrix are i.i.d. and take value $M_{ij}^* = 1$ with probability $p = f$ and value $M_{ij}^* = 0$ with probability $1 - p$.

One important aspect to take into account is that all the above procedures might generate empty rows and/or columns. This might or might not be desirable/acceptable. If not, there are ad hoc potential solutions. In the filling approach that starts from an empty matrix, assuming $N \geq (r + c)$ one might first assign a 1 to each entry $M_{ix}^* \forall i \in [1, r]$, with x being a random integer in $[1 \dots c]$, and then to each entry $M_{yy}^* \forall j \in [1, c]$, with y being a random integer in $[1 \dots r]$. In sparse matrices where $N < (r + c)$, then a different approach would be needed. Assuming that, for example, $r > c$ and $N \geq r$, one might first convert to 1 all the M_{ij}^* entries where $i = j$ and then attribute one presence in the each entry $M_{ix}^* \forall i \in [c, r]$ and x being a random integer in $[1 \dots c]$. We note, however, that these approaches are superseded by the following models, which, by placing soft (resp. hard) constraints on the row and/or column sums, largely (resp. completely) reduce the probability of having zero margins in the randomized matrices, given that the empirical margins are typically nonzero (unless there are isolated nodes in the data).

Unconstrained rows and Fixed columns, or vice versa (EF or FE). In these models, either the r row sums $R = \{r_1, \dots, r_r\}$ (for *FE*) or the c column sums $C = \{c_1, \dots, c_c\}$ (for *EF*) are treated as hard constraints, while the other margin is left unconstrained. For *EF*, this means that each randomized matrix M^* generated by the null model is such that $\sum_{i=1}^r M_{ij}^* = c_j$ (for $j = 1, c$), where c_j denotes the empirical value of the column sum measured in the data. For *FE*, each randomized matrix M^* is such that $\sum_{j=1}^c M_{ij}^* = r_i$ (for $i = 1, r$), where r_i denotes the empirical row sum. In the jargon of physics, when the rectangular matrix represents the adjacency matrix of a bipartite network, both models are examples of the microcanonical Bipartite Partial Configuration Model (BiPCM) [48,75], because the constraint is the degree of each node (“configuration model”) but it is enforced “partially,” i.e., on only one of the two layers.

As for the *EE* case, the *FE* (and the analogous *EF*) can be easily and efficiently implemented using different approaches. Conceptually, *FE* requires sampling uniformly all the $(\prod_{i=1}^r r_i^c)$ matrices with given R (similarly, *EF* requires sampling uniformly all the $(\prod_{i=1}^c c_i^r)$ matrices with given C). Practically, this can be achieved by randomizing the positions of the 0s and 1s in each individual row (column) of M . If only the position of presences and absences within a row (column) is randomized but not their respective numbers, it is intuitive that R in the randomized matrix M^* will remain the same as in M . A simple algorithmic implementation of *FE* might consist of generating r random lists each including r_i 1s and $c - r_i$ 0s and then combining those lists into a matrix M^* . Similarly, for *EF*, one might generate c random lists each including c_j 1s and $r - c_j$ 0s and then combine those lists into a matrix M^* .

As in the case of *EE*, such approaches might result in generating empty columns or rows. If this is not desirable, one can implement additional steps/constraints in the randomization algorithms. For example, for *FE*, one might preassign a presence to the j -th position to a randomly selected row $\forall j \in [1, c]$. Then, the algorithm will be implemented as above, but presences and absences will be randomly placed in each row conditionally to the preassignments (with the probability distribution of edge weights in MM' being given by the hypergeometric distribution [85]).

Constrained rows and columns (FF). This null model constrains both margins R and C in a hard fashion (usually taking their values from the empirical matrix): each randomized matrix M^* is such that $\sum_{i=1}^r M_{ij}^* = c_j$ (for $j = 1, c$) and $\sum_{j=1}^c M_{ij}^* = r_i$ (for $i = 1, r$), where c_j and r_i denote the empirical values of the j th column sum and i th row sum observed in the data, respectively. Let us denote any such matrix by $M^*(R, C)$. Ideally, this model assigns the same probability to all such matrices, for fixed R and C . In the jargon of physics, this model is known under the name of microcanonical Bipartite Configuration Model (BiCM) [48,75] as it samples uniformly all bipartite networks with the same hard degree sequences R and C . This is one of the null models that has received most of the attention from different fields, as it is

relevant not only for a variety of practical applications but also for important theoretical questions in mathematics. Indeed, in this case, even enumerating how many matrices with given margins R and C is an open problem, and only asymptotic expressions are known in certain regimes [75,86,87]. For this reason, it has been the object of a large number of studies that have produced a large corpus of methods that is constantly growing. As anticipated, such methods can be roughly classified into those that obtain a random matrix from scratch, i.e., by filling up at random an empty matrix, and those that randomize an existing matrix. We will first cover filling strategies and then move to randomization algorithms.

In principle, in the filling approach, any **extant** matrix $M^*(R, C)$ of size $S = r \times c$ with row and column totals equal, respectively, to R and C can be obtained by starting with an empty $r \times c$ matrix where each entry is initially $M_{ij}^* = 0$ and then converting progressively entries to 1 until the marginal totals matches exactly the expected R and C . We emphasized the term “extant” as it is not for granted that an $r \times c$ matrix with margins matching arbitrary integer numbers $R = \{r_1, \dots, r_r\}$ and $C = \{c_1, \dots, c_c\}$ exists. In other words, in general, a matrix $M^*(R, C)$ does not exist for all values of R and C . In the language of graph theory, the bipartite degree sequences R and C must be **graphic**, i.e., realizable by at least one bipartite graph. Intuitively, an obvious necessary condition for the existence of the matrix is that $\sum_{i=1}^c c_i = \sum_{j=1}^r r_j$ (the total number N of 1s in the matrix is the same if computed by first summing over columns, and then over rows, or the other way around). However, such necessary condition does not ensure the existence of $M^*(R, C)$, i.e., it is not sufficient. Indeed, a necessary and sufficient condition for the existence of $M^*(R, C)$ is provided by the classic Gale–Ryser theorem [78,88]. If we follow the simple example provided by Gale [88], we can imagine that our matrix maps the placement of r families going to a picnic across c buses. There, r_j is the total number of members in the j -th family, and c_i is the total number of places available in the i -th bus. The theorem answers the question, “When is it possible to seat all passengers in such a way that no two members of the same family are in the same bus?” Such a question is equivalent to asking whether it is theoretically possible to generate at least one $M^*(R, C)$ matrix. The theorem provides the following necessary and sufficient condition for the existence of a solution to the problem:

$$\sum_{j=1}^k r_j \leq \sum_{j=1}^k s_j$$

for all integers k , where $s_j = \{c_i | c_i \geq j\}$, $c_i = 0$ for $i > c$, and $r_i = 0$ for $i > r$ and with c_i and r_j being listed in decreasing order [78,88].

The existence of the $M^*(R, C)$ matrix does not imply that generating it is an easy task; e.g., if we start from an empty matrix M^* and then progressively modify randomly selected entries to 1 while checking at each step that the observed marginal totals do not exceed the desired R and C , we will most likely end up in a situation where any further addition of a 1 to M_{ij}^* will lead to exceed either r_j or c_i . However, the sufficient condition provided by the Gale–Ryser problem offers also an efficient way to generate a matrix $M^*(R, C)$. In the example of the families and buses, if a solution exists, it will be always possible to succeed in placing all members of the different families in different buses (i.e., avoiding that two members of the same family are in the same bus) by allocating first all the members of the largest family to the buses having the largest number of available seats, then all the members of the second largest family to the buses having most free seats after the allocation of the first family, then all the members of the third family to the buses having most free seats after the allocation of the first and second family, and so on. This procedure will always end with all persons seated, all members of each family seated in a different bus, and no empty seats left in any bus. It is clear, however, that

although this procedure permits to generate one $M^*(R, C)$ matrix, it will always generate the same matrix, while it is clear that we need to succeed in generating different matrices (without getting stuck in the allocation of 1s to the matrix entries before reaching the target R and C) and, crucially, to sample them with uniform probability from the universe of all possible $M^*(R, C)$ matrices (whose number, as we mentioned, is not even known in full generality).

Various approaches have been proposed for this purpose, but most of them either have problems in terms of computational efficiency or sample $M^*(R, C)$ matrices with biased (i.e., nonuniform) probability (or both) [89–91]. The “knight tour” algorithm proposed by Sanderson [92] and its variations [93] try to fill progressively the matrix choosing cells randomly one at a time and “backtracking,” i.e., returning to a previous state, when the procedure gets stuck, that is when it is no longer possible to fill a cell without exceeding R or C . Besides being prone to biases [93], these methods are impractical for even moderately sized matrices as the algorithm might spend a considerable (and hardly predictable) amount of time for backtracking [93].

More recently, Chen and colleagues [94] have proposed an approach based on “sequential importance sampling,” which generates the matrix by sampling columns progressively. As noted by the authors, if the position of the c_j 1s of the j th column is determined uniformly at random, it becomes extremely difficult to sample a valid column, which makes the process exceedingly computationally intensive. To overcome this issue, they proposed to generate the columns using the conditional-Poisson sampling method [95,96], which, in a simplification, increases the chances to allocate a 1 in the i th position of the target j th column if r_i is large. This choice dramatically improves the computational efficiency of the method but prevents it from sampling matrices exactly from the uniform distribution, with the extent of the bias depending on both the actual setup of the conditional-Poisson sampling (i.e., the degree to which r_i affects the probability of the i th element in the j th column to be a 1) and the distribution of values in R and C .

The alternative approach is that of using Markov Chain procedures where small incremental changes are applied to the target matrix. Those changes progressively bring the matrix far from its initial status. Ideally, if enough small changes are applied to the starting matrix, the probability of sampling any of all $M^*(R, C)$ matrices will converge to a uniform distribution. Clearly, the changes will need to ensure that the marginal totals of the initial matrix, R and C , remain unaltered. The easiest—and most classical—way to achieve this goal consists in progressively selecting “checkerboards” and swapping their diagonal elements [93,97]. A checkerboard is a specific pattern in the matrix, involving two row nodes (say, i and z) and two column nodes (say, j and k), where $M_{ij}^* = 0$, $M_{ik}^* = 1$, $M_{zj}^* = 1$, and $M_{zk}^* = 0$. It is intuitive that if we modify the matrix by “swapping” the diagonal elements of the checkerboard, i.e., by setting $M_{ij}^* = 1$, $M_{ik}^* = 0$, $M_{zj}^* = 0$, and $M_{zk}^* = 1$, then the row (r_i and r_z) and column (c_j and c_k) totals will not change, leaving R and C unaltered. Note that the rows and columns forming the checkerboard do not need to be contiguous in the matrix. The move that is iteratively applied to the initial configuration in order to generate a family of randomized variants has been popularized with the name of local rewiring algorithm (LRA) in the literature concerning unipartite networks [98–101].

One obvious drawback of this procedure is that each swap will produce a small modification in the matrix so that a very large number of swaps will be required to generate “sufficiently random” matrices (i.e., matrices sampled uniformly from all possible ones). How many swaps ensure that each randomized matrix is sampled uniformly from the universe of possible $M^*(R, C)$ matrices is not clear. In practical implementations in the ecological literature, the number of swaps used has been one or more orders of magnitude larger than the number of cells in

the matrix; e.g., a common choice has been that of using 30 to 50k swaps for matrices having size smaller than 100×100 cells [102–104]. However, performing many swaps does not ensure unbiasedness of the algorithm; e.g., a rule of thumb for the LRA on unipartite networks (hence, in a setting different from the one considered here) recommends the number of swaps to be larger than $4N$, i.e., four times the total number of 1s in the network [99,100]. Yet, when the margins of the matrix (i.e., the node degrees) are very heterogeneous (i.e., when the second moment of the empirical degree distribution is larger than a certain threshold), it has been shown rigorously that, irrespective of the number of swaps being executed, the LRA remains biased as it fails to sample the desired matrices uniformly [89–91], and no computationally feasible corrections to this bias have been proposed. In particular, uniformity holds (at least approximately) only when the degrees are such that $k_{\max} \cdot \bar{k}^2 / (\bar{k})^2$ is much smaller than the total number of nodes—with k_{\max} being the largest degree in the network, \bar{k} being the average degree, and \bar{k}^2 being the second moment [90]. In order to restore uniformity, at each iteration, the attempted “rewiring move” must be accepted with a probability that depends on some complicated property of the current network configuration. Since this property must be recalculated at each step, the resulting algorithm is extremely time consuming. Unfortunately, on real-world networks, the matrix margins (node degrees) are typically very heterogeneously distributed, and their second moment exceeds the aforementioned threshold, which implies that the LRA is prone to bias in most practical situations [26,90].

Even in the “weakly heterogeneous” regime for which uniformity can be in principle ensured, the computational costs involved in the execution of many swaps and checks to produce a single “sufficiently randomized” matrix should be multiplied by the (large) number of random matrices that are needed to perform robust tests when comparing the empirical network with the randomized ones. To partially reduce these costs, two modified approaches have proposed, i.e., the so-called “sequential” and “independent” swap algorithms [93,105,106]. In the former class of algorithms, a predefined randomly chosen number of swaps (e.g., 30k) is applied to the original matrix to generate a single random matrix. In the latter class, an initial, large number of swaps is applied to the starting matrix to generate the first random matrix, while each subsequent random matrix is generated by applying a smaller number of swaps to the last generated matrix in the sequence. Clearly, the second approach is less computationally intensive when there is a need for generating a large set of random matrices. Still, biases in hypothesis testing might emerge from the potential non-independence of the random matrices in the sequence.

Recently, the computational challenges associated with classical swap algorithms have been partially overcome by more efficient approaches where the swaps are replaced by trades of elements between adjacency lists representing the set of neighbors of a focal node in the network representation of M^* [51,79,107]. As in the original example [79], we can consider a matrix M^* where the r rows correspond to a set of kids, and the c columns correspond to a set of different baseball cards. Each cell M_{ij}^* in the matrix indicates whether (1) or not (0) the i -th kid owns the j -th card. Then, we can imagine that the kids meet during class break to trade cards and that trades happen according to the following two rules: (i) cards have identical value, meaning that one card is traded with exactly one card; and (ii) kids are not interested in owning duplicated cards, thus a trade cannot take place if leading to such a situation. Now, a situation where, in compliance with the above rules, a kid trades a Babe Ruth with a Willie Mays will correspond to a typical swap in the matrix. The number of cards owned by the two kids will remain the same, as well as the number of owners for the two cards. However, nothing prevents the two kids from trading more than one card. If we call $\{a\}$ and $\{b\}$ the sets of cards owned, respectively, by the first (k_a) and the second kid (k_b), we can identify the set of cards

that k_a can potentially trade with k_b as $a_b = \{b\} - \{a\}$, and the set of cards that k_b can potentially trade with k_a as $b_a = \{a\} - \{b\}$. The two kids will be in a position to make a trade where k_a gives n cards sampled from a_b to k_b while receiving from k_b an identical number of cards sampled from b_a , with n being an integer varying between 0 and the minimum of a_b and b_a sizes. Intuitively, as in the case of the single card trade, this exchange will result in no changes to the total number of cards, respectively, owned by k_a and k_b , nor in the number of kids owning any of the traded cards.

The algorithmic implementation of such multiple card trades consist in first converting the matrix in a set of adjacency lists mapping the position of 1s in each column for each row (or the position of 1s in each row for each column). In the example above, such lists will include the set of cards owned by each kid (or the set of kids owning a certain card). Then, at each step, two lists will be drawn at random, and a trade of size n (with n being an integer randomly sampled with uniform probability between 0 and the maximum number of tradable cards) will be performed. There are two distinct cases where a step will result in no changes in the underlying M^* , namely, when the maximum number of tradable cards is 0 (i.e., when a_b or b_a or both are empty) or when n is randomly assigned a value of 0. A formal proof that the Curveball algorithm is unbiased, i.e., it samples uniformly from the universe of all possible $M^*(R, C)$ matrices, has been provided [81]. However, it has also been shown that, while the algorithm remains unbiased even if “no-trade shuffles” are excluded from the Markov Chain (i.e., if n is sampled between 1 and the maximum number of tradable cards when the latter is ≥ 1), the sampling is no longer guaranteed to be uniform if “no-trade row pairs,” i.e., all the list comparisons where there are no tradable cards, are excluded.

By modifying larger portions of M^* at each step, the Curveball and other similar algorithms [51,107] speeds up dramatically the Markov Chain convergence with respect to older swap algorithms, i.e., they reach a virtually uniform sampling of $M^*(R, C)$ matrices in a much smaller number of steps [79–82]. However, both for the “classical” and the more recent approaches, how fast (i.e., in how many steps) the selected algorithm converges toward the uniform sampling of random matrices for a given $M^*(R, C)$ is an open question.

Unconstrained rows and Proportionally constrained columns, or vice versa (EP, PE). We now come to models where one margin of the matrix is left unconstrained (E), while the other margin is fixed in a soft/proportional way (P). In statistical physics, these models are known as *canonical* Bipartite Partial Configuration Models (BiPCMs) [48,75]. These models are naturally sampled in the filling approach, starting from an empty matrix and sampling each of its S entries as independent Bernoulli trials, each entry M_{ij}^* being given value $M_{ij}^* = 1$ with an appropriate success probability p_{ij} , and value $M_{ij}^* = 0$ with the complementary probability $1 - p_{ij}$. This means that the distribution $P(M^*)$ over randomized matrices factorizes in this case as independent trials over different pair of nodes. If the angular brackets $\langle \cdot \rangle$ denote expected values over this distribution, then clearly $\langle M_{ij}^* \rangle = p_{ij}$.

In the *EP* case, the probability p_{ij} should be chosen in such a way that the expected column sums $\langle C \rangle = \{\langle c_1 \rangle, \langle c_2 \rangle, \dots, \langle c_c \rangle\}$ equal the corresponding empirical values $C = \{c_1, c_2, \dots, c_c\}$. This means $c_j = \sum_{i=1}^r \langle M_{ij}^* \rangle = \sum_{i=1}^r p_{ij}$ for all $j = 1, c$. It is straightforward to show that a solution for p_{ij} , which realizes the requirement, is $p_{ij} = c_j / r$; this solution coincides with the one producing the maximum-entropy probability $P(M^*)$, given the observed margin C [3,45,48,75].

Similarly, in the *PE* case, the probability p_{ij} is chosen in such a way that the expected row sums $\langle R \rangle = \{\langle r_1 \rangle, \langle r_2 \rangle, \dots, \langle r_r \rangle\}$ equal the corresponding empirical values $R = \{r_1, r_2, \dots, r_r\}$. This means $r_i = \sum_{j=1}^c \langle M_{ij}^* \rangle = \sum_{j=1}^c p_{ij}$ for all $i = 1, r$. The solution, which again coincides with the maximum-entropy one given the observed margin R , is $p_{ij} = r_i / c$ [3,45,48,75].

So, both the *EP* and the *PE* models can be easily sampled in the filling approach via independent Bernoulli trials, with success probabilities immediately calculated from the empirical margins (C or R , respectively). In each case, the success probability is proportional to the value of the margin in the corresponding row or column (hence the term “proportional”).

Proportionally constrained rows and columns (PP). This model samples in principle each possible binary matrix M^* with the same dimensions (r and c) as the empirical matrix but assigns different probabilities to different matrices in such a way that the mean values of both row and column sums over the ensemble match the empirical ones. In other words, the probabilities given by different matrices are such that $\langle c_j \rangle = \sum_{i=1}^r \langle M_{ij}^* \rangle = c_j$ (for $j = 1, c$) and $\langle r_i \rangle = \sum_{j=1}^c \langle M_{ij}^* \rangle = r_i$ (for $i = 1, r$), where c_j and r_i denote the empirical values of the j th column sum and i th row sum observed in the data, respectively, while the angular brackets denote average values of the ensemble probability $P(M^*)$, as above. In the jargon of physics, this model is known under the name of canonical Bipartite Configuration Model [3,75].

As for the *EP* or *PE* models, in the *PP* case, most randomization methods generate an instance M^* via a filling approach, by looking for an appropriate probability p_{ij} for the event $M_{ij}^* = 1$ (so that M_{ij}^* with probability $1 - p_{ij}$; note also that the probability distribution of edge weights in the projected one-mode matrix MM' is given by the Poisson-binomial distribution, where the parameters are derived from p_{ij} [85]). This means that the probability $P(M^*)$ is again assumed to be factorizable into independent Bernoulli trials, each with appropriate success probability p_{ij} , over distinct pairs of nodes. This assumption is correct, as we discuss below; however, in the *PP* model, finding the explicit expression for p_{ij} given the observed values of R and C is not as easy as in the simpler *EP* and *PE* cases. Indeed, different methods differ in how they define p_{ij} . Note that valid values of p_{ij} are subject to *at least* three constraints. First, because they must be well-defined probabilities, they need to take values in $0 \leq p_{ij} \leq 1$ for all i, j . Second, since the expected value of M_{ij}^* is $\langle M_{ij}^* \rangle = p_{ij}$, enforcing the average row constraints requires that $\sum_{j=1}^c p_{ij} = r_i$. Third, enforcing the average column constraints similarly requires $\sum_{i=1}^r p_{ij} = c_j$. In principle, within these minimum constraints, many different choices for p_{ij} are still possible. Three prototypical choices are discussed below.

First, defining $p_{ij} = r_i c_j / N$ directly matches the last two constraints (whence, again, the term “proportional”) but does not necessarily ensure the first one. Indeed, it is possible to show that, for empirical values of r_i and c_j that are too broadly distributed over rows and/or columns, respectively, one gets $p_{ij} > 1$. This creates a situation akin to the one we discussed in section “Randomization algorithms and procedures-Constrained rows and columns (FF)” in the *FF* case: If the second moment of the degree distribution of row and column nodes is too large, then it becomes much harder to impose double constraints on the margins of the matrix. Unfortunately, real-world bipartite networks are typically so heterogeneous that this problem cannot be avoided. If out-of-bound values of p_{ij} are truncated, so that the values are forced to remain between 0 and 1 [1], then the first constraint is respected, but one loses the last two constraints and the average margins no longer match the empirical values.

Second, fitted linear models can be used to define p_{ij} as the value of M_{ij}^* predicted as a function of r_i and c_j [21]. We note that these Bernoulli trial approaches generate random matrices that do not retain the original marginal total distribution. This is because these trials follow a Poisson distribution that is nearly symmetrical at larger marginal values while being positively skewed at small values, by this causing higher simulated node numbers for comparatively low row or column total values.

Third, entropy maximization can be carried out explicitly to find the exact values of p_{ij} realizing the joint row and column constraints [48,75]. The result is given in implicit form,

through the parametric expression $p_{ij} = x_i y_j / (1 + x_i y_j)$, where the $r + c$ parameters (x_1, \dots, x_r) and (y_1, \dots, y_c) are the (provably unique [87,108]) nonnegative solution to the following set of $r + c$ coupled nonlinear equations:

$$r_i = \sum_{j=1}^c \frac{x_i y_j}{1 + x_i y_j}, \quad i = 1, r \quad (1)$$

$$c_j = \sum_{i=1}^r \frac{x_i y_j}{1 + x_i y_j}, \quad j = 1, c. \quad (2)$$

Efficient algorithms to solve the above equations exist [108]. It is possible to show that, for sufficiently narrow distributions of r_i and c_j over rows and columns, respectively, the solution to the above equation becomes approximately $p_{ij} \approx x_i y_j \approx r_i c_j / N$, consistently with the expression discussed above. Unfortunately, as we mentioned, typical real-world distributions of the matrix margins are instead too broad for this approximation to hold. Therefore, the correct procedure to implement the *PP* model remains the one of solving Eqs 1 and 2 and using the resulting, exact values of p_{ij} . A recent review [85] found that the Bipartite Configuration Model is the fastest and most accurate method for computing p_{ij} .

Recently, [109] proposed another *PP* randomization model that does not rely on probability-based cell filling. Rather than randomizing whether a cell is filled or not, this approach first randomly sets the target values for r_i and c_i in M^* and then fills the cells of M^* to achieve these target values.

Constrained rows and proportionally constrained columns, or vice versa (*FP* and *PF*). Algorithms where the proportional constraints are applied only to rows (or columns) while the marginal totals of columns (or rows) are kept fixed to C (or R), i.e., the *PF* (or *FP*) case, are a theoretical possibility but have not received much attention and have been rarely used in real-world analyses. Some straightforward implementations have, however, been proposed [1]. For the *FP* model, one might start with an empty matrix and then add presences to rows one row at a time. For each i -th row, the algorithm reiterates the procedure of sampling a random j with probability c_j / N and setting M_{ij}^* to 1 until the total number of presences in the target row matches the desired value (r_i). For *PF*, for each j -th column, the algorithm reiterates the procedure of sampling a random i with probability r_i / N and setting M_{ij}^* to 1 until the total number of presences in the target column matches the desired value (c_j). The procedure can sometimes end up in matrices with either empty rows or columns [1]. Adjustments similar to those discussed for the *EF* and *FE* cases can be used to tackle this potential issue.

Choosing a null model

In the previous sections, we have listed the most common constraints/rules that can be taken into account to randomize a bipartite matrix, and we have described how such rules can be implemented into dedicated algorithms. However, we have not discussed a fundamental question that, even if not in itself central to the technical details of the randomization procedures, constitutes the main reason for which these are developed. Why—or under which circumstances—one should choose one specific set of constraints over another? This is an important question because, as Fig 5 illustrates, the choice of randomization constraints can impact the patterns that are detected.

Each choice could be equally valid and useful to answer specific questions. In fact, identifying such questions within a specific research context is an essential and dire challenge in itself, which should be regarded by researchers as a first—possibly the most important—step to be

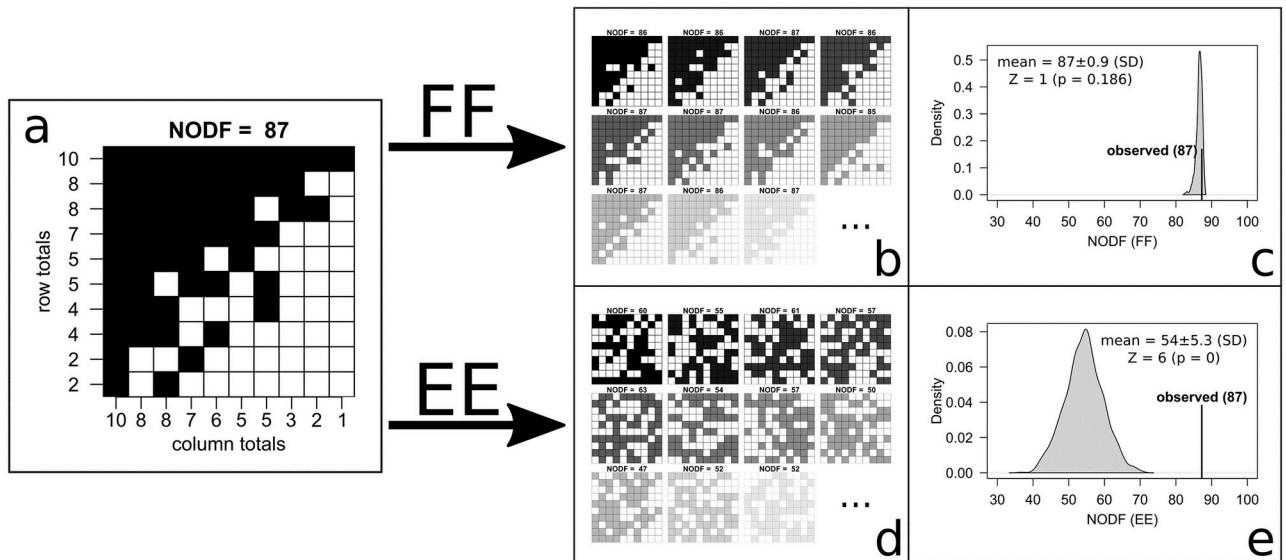


Fig 5. Effect of different randomization constraints on pattern detection. Example of how applying different constraints to matrix randomization can lead to contrasting results in pattern detection. In this example, we apply the same pattern detection workflow as described in Fig 3. Black/gray cells in each matrix indicate presence of links (i.e., 1s) between the items in rows and the items in columns, while white cells indicate the absence of links (i.e., 0s). First, the structural measure of interest (in this case, a nestedness metric, *NODF* [71]) is computed on the target matrix (a). Then, two sets of 1,000 randomized versions of the starting matrix are generated using, alternatively, an algorithm that generates random matrices with the same exact row and column totals of the starting matrix (FF), and an algorithm that generates random matrices having the same size, shape, and fraction of occupied cells of the starting matrix, but with varying (equiprobable) row and column totals (EE). The target metric is computed for each random matrix in the two sets (b, d). Then, the starting *NODF* value is compared against the two distributions of “null” values in the two sets of randomized matrices. In this example, the starting *NODF* does not depart significantly from the null expectation from the set of matrices generated with the FF algorithm ($Z = 1; p = 0.186$). Conversely, the pattern is identified as particularly strong when compared with the metrics measured in the random matrices generated with the EE algorithm ($Z = 6; p = 0$).

<https://doi.org/10.1371/journal.pcsy.0000010.g005>

completed before even starting to think about the actual implementation of randomization routines. Unfortunately, this is not often the case, and sometimes researchers make a fairly blind use of null models, without a clear reasoning behind the choice of the enforced constraints. Overseeing the importance of linking solid questions about processes to null model pattern analysis might lead to a difficult or biased interpretation of the results. In the worst possible situation, one could even take advantage of the fact that different, sometimes contrasting, results can arise when processing the same data using different randomization strategies (Fig 5) and adjust procedural choices to steer the results in the desired direction.

In this section, we discuss different methodological considerations, both theoretical and practical in nature, that are relevant to choosing how to generate the random networks and which characteristics to preserve.

Analytical versus numerical simplicity. We start with some practical considerations about the choice of null models based on their simplicity. Before delving into this, we should, however, make a general warning: Choosing a null model based *solely* on simplicity, e.g., computational efficiency or mathematical convenience, is of course not recommended and not scientifically acceptable in general. This is because the choice of the null model should rely primarily on the soundness of the underlying null hypothesis, leading to the identification of the margins to be preserved and of whether they are enforced as soft or hard constraints.

Generally, one expects that the statistical power of matrix randomization techniques increases as more matrix features are controlled [85]. This implies that the most constrained

models (*FF* or *PP*), which are also the most complex ones, may represent the most statistically robust choice (provided there is no concurrent risk of overfitting). In practice, implementing these models is feasible when a sufficient number of bipartite networks can be generated in a practical amount of time. When the *FF/PP* models are computationally impractical, the next less complicated choices are the *FE/PE* or the *EF/EP* models, because these at least fully control for one dimension of the network and can be computed one row (or one column) at a time. The choice between the *FE/PE* and *EF/EP* models depends on a substantive and context-dependent judgment concerning whether it is more important to control the effects of the rows or columns. An alternative approach would be that of exploring simultaneously a wide range of possible combinations of constraints and then placing and discussing the results within the multidimensional null modeling space identified by such constraints [20]. On the one hand, this might provide a more comprehensive information on network structure, but at the cost of a more challenging interpretation and increased computational demand.

As another dimension along which simplicity considerations apply, generating and/or handling a null model entails different levels of numerical and mathematical complexity, depending on whether constraints are enforced in a hard or soft manner. Basically, hard constraints are easier to work with via numerical sampling of randomized matrices (using one of the various algorithms discussed in the previous sections), while they are very difficult to be implemented in an analytically tractable way. This is because the hardness of the constraints makes the entries of the randomized matrices dependent on each other, since such entries must always sum up to the same value. For this reason, filling algorithms generally do not lead to unbiased (uniform) sampling, and the reliable schemes are indeed based on iterative randomizations of the original matrix. As an extreme example of the unfeasibility of analytical approaches under hard constraints, we recall that in the doubly constrained (*FF*) case, the (uniform) probability distribution over configurations cannot even be calculated in the general case, because it requires the solution of an unsolved combinatorial enumeration problem.

By contrast, soft constraints lead to independent entries in the randomized matrices, even in the doubly constrained (*PP*) case [75]. This makes the probability distribution for the entire matrix factorize as independent (but not identically distributed) Bernoulli trials over distinct matrix entries, so that filling algorithms becomes exact, once the correct success probability p_{ij} is determined for each entry i, j . Moreover, these probabilities coincide with the expected values $\langle M_{ij}^* \rangle$ of each matrix entry. Therefore, using these (exact) expectation values, the averages of many quantities of interest across randomized versions of the orginal network can be calculated analytically, without even having to sample matrices from the distribution [24,26]. This is a considerable speed-up compared with the generation of several random matrices and averaging of the quantities of interest across the sample. However, the soft and hard enforcements of the same constraint(s) lead to nonequivalent null models as soon as (at least) one of the two margins is constrained [75,87], which implies that one cannot use the two implementations interchangeably.

As a final consideration, we note that, once the above choices (which constraints and whether hard or soft) are made, it is of course reasonable to implement the selected null model in the simplest and/or more efficient way. Indeed, for a given randomization goal, there might be tradeoffs between the computational demand of a given algorithm, its reliability (e.g., in terms of sampling uniformly from the universe of possible matrix configurations), and its easiness of implementation and integration in different analytical workflows; e.g., the recently introduced *fastball* algorithm [107] has a theoretical time complexity $O(n)$ time, while the earlier *curveball* algorithm [79] has a theoretical time complexity of $O(n \log n)$, but the practical

running times of both algorithms depend on the programming language used to implement them. Since both algorithms permit sampling random matrices without biases, the choice of using one or another method depends on practical considerations related, among the others, to the actual amount of data to be processed and the coding integration with other analyses. If one has to randomize a few, small matrices, then using a simpler but less efficient code implementation might be a reasonable choice, while for larger analyses, the performance advantage might outweigh the potential additional effort in coding integration.

Constrained versus unconstrained margins. Conceptually, the most important choice is realizing which constraints to apply to the randomized matrices, or, in other words, which margins to constrain. This consideration revolves around the nature of the features the researcher wishes to control, and how those features translate into specific matrix properties. A typical example is provided by the analysis of rectangular matrices representing the presence or absence of a set of plant or animal species across a set of localities (often islands or, in any case, isolated habitat patches). In that context, the marginal totals could be linked to different kind of ecological information. Specifically, for a matrix where rows correspond to species and columns correspond to islands, the column totals, representing species richness across localities, might be linked to various features affecting local species diversity. Some of these might be obvious and/or known, such as island size, while others, such as habitat heterogeneity or resource availability, or particular biogeographical features, might be less intuitive or not known. Still, one might assume that the effect of all of these features combined is actually reflected in column totals. Similarly, one could consider row totals, i.e., the prevalence of species across the islands, as a proxy for various features of the species, such as their ability to disperse and colonize islands and the generalism or specialization in their needs for resources.

Based on these considerations, one should then decide whether or not to preserve the marginal totals in the randomized matrices. To understand the choice, we need to make a clear distinction between “patterns,” i.e., the different forms of organization of the various entities, which are represented in the matrix and that are captured by ad hoc metrics (such as nestedness [38,110]), and the “processes” that led to the emergence of such patterns. Research questions usually target both patterns and processes, i.e., one could be interested in measuring whether and to what extent species are distributed across islands in a certain, nonrandom pattern, and what are the causes (i.e., processes) that led to such a pattern. However, these two objectives are not independent of one another. On the contrary, they are two sides of the same coin.

To assess the relevance of a given process, one should ideally identify some way to isolate the effect of that process on the observed pattern from all the other processes that might be also involved in the emergence of the pattern. The null model approach, which is central to this review, offers one straightforward way to achieve this objective. In principle, one could explore the importance of a given process by comparing the target pattern in the original matrix with the same pattern in a large set of randomized matrices obtained by preserving all the features that might affect the emergence of the the observed pattern, with the exclusion of those features potentially emerging from the process of interest. But this also means that the assessed magnitude of the pattern could vary depending on the identity of the target process. Thus, a matrix might show a strong structural pattern when examined with a focus on a given process, but no structure in a different context [20].

Patterns in a matrix can be often described and measured by single values; e.g., one could measure the “temperature” (the original metric used to describe nestedness [38]) of a given matrix and then use that only information to assess whether or not the matrix is structured, by placing the observed temperature within the theoretical range of possible values (0 to 100). However, as already emphasized multiple times within this review, such an approach might

not be particularly enlightening. Specifically, most metrics of matrix structural patterns are not independent from matrix structural properties such as matrix size, shape, fill, and marginal totals. Therefore, the metrics' raw values are a simultaneous result of the processes that determined the matrix properties and of *other* processes. Such *other* processes are usually central to interesting and meaningful questions, and standardizing the target metrics by controlling for matrix structural patterns is an obvious way to try isolating them. For example, by comparing the structure of a target species–island matrix with that of randomized versions having the same marginal totals, one might be able to assess the structuring importance of some ecological processes other than those which determine local species richness and species prevalence across islands. Similarly, one could constrain selected structural properties to explore specific hypotheses or to answer specific questions; e.g., one could test the importance of local species richness in determining nestedness by comparing the target matrix with randomized versions obtained by constraining row marginal totals only (i.e., species prevalence across sites in our species–island matrix).

To make a different example, we might consider a matrix mapping the authorship of scientific publications (with authors in rows and publications in columns). We can imagine a situation where one would be interested in quantifying the overall tendency for collaboration between authors. It is obvious that the frequency of coauthorship would naturally increase with the overall productivity of the scientific community represented by the matrix. However, it might be also reasonable to assume that the overall productivity is both a driver and a result of collaborations. Thus, it might be meaningful to assess the degree of coauthorship both taking or not taking into account the overall community productivity. One could also advance hypotheses on how the individual productivity of the different authors might affect the overall intensity of coauthorships (quantified by row totals in the matrix); i.e., we might expect that a situation where all the authors have similar productivity would lead to different coauthorship patterns/levels compared to a situation where a few authors are highly productive while most authors are associated with few publications. A similar reasoning applies to the number of authors per publication (quantified by column totals in the matrix). Intuitively, we might expect different coauthorship patterns in a situation where most publications tend to have a similar number of coauthors compared to a situation where we have a few articles signed by many authors and most papers authored by few scientists. Again, depending on their actual goals, the investigators might decide to either constrain or not row and/or column totals when generating the randomized matrices to be used as a frame of reference to assess the target community's tendency for scientific collaboration.

Another example could be that of exploring the determinants of modularity in a network mapping listeners' musical preferences. There, one could be interested in exploring how generalism in listeners' tastes affects network modularity. In that context, the observed degree of modularity should be compared to that in randomized versions of the initial network having identical number of listeners per musical genre, but unconstrained number of genres per listener.

In general, enforcing one margin of the matrix implements the idea that, in order to characterize the null behavior of the system's properties, it is essential to control for the empirically observed heterogeneity of the nodes of the corresponding layer of the bipartite network. So, in short, choosing which margins to constrain boils down to choosing whether it is appropriate to control for the heterogeneity of nodes in one layer, in the other layer, or in both layers.

Soft versus hard constraints. Once the formal choice of the constraints is made, the next important choice is whether these constraints should be enforced in a hard (*F*) or soft (*P*) way. In the literature, there is a tendency of regarding the two alternatives as two basically equivalent routes to the implementation of the same null model. This is true “asymptotically” (i.e.,

when the size of the matrix becomes larger and larger), but only for certain constraints: e.g., for a single constraint that is global in nature (e.g., the total number of 1s in the matrix, or equivalently the matrix fill f as in the EE case), the probability distribution of matrices in the soft (canonical) ensemble concentrates around the smaller set of matrices that realize the constraints in the hard (microcanonical) ensemble [75]. In information theory, this property is known as *asymptotic equipartition property* (AEP) [111], while in statistical physics, it is known as *asymptotic equivalence between canonical and microcanonical ensembles*, or, more compactly, *ensemble equivalence* [25,75].

While attractive and convenient, the property of ensemble equivalence breaks down as soon as the enforced constraints are local, i.e., node-specific in nature [25,73–75]. This deep and somewhat surprising result means that, for the cases of interest here, the FF and PP null models are not asymptotically equivalent to each other, and the same goes for FE versus PE and for EF versus EP . To some researchers, the breakdown of ensemble equivalence might seem primarily a theoretical curiosity with abstract significance and no operational implications. However, this is not the case: It is possible to prove rigorously that if the probability of configurations with soft constraints does not concentrate around those with hard constraints (a notion known as “measure nonequivalence”), then there must necessarily exist properties of the system that have different expected values under the two ensembles (which is known as “macrostate nonequivalence”) [112]. This means that the two null models will produce different reference values for certain properties and may, therefore, single out different patterns when such reference values are compared with the empirical ones observed in the data. As a notable example for bipartite plant–pollinator networks in ecology, it has been found that (i) certain definitions of the nestedness property mentioned above indeed display different expected values in the FF and PP null models and (ii) even more surprisingly, alternative definitions of nestedness turn out to be mutually positively correlated in the canonical ensemble and negatively correlated in the microcanonical one [113]. This means that the empirical network might appear as “positively nested” under one null model and “negatively nested” under the other null model. This seemingly puzzling conclusion is actually a direct consequence of the fact that, since the nestedness is strongly influenced by the values of the margins of the matrix (as we have already noted above), it can obey very different statistical distributions when the margins are fixed exactly and when they are allowed to fluctuate.

An important consequence of ensemble nonequivalence for models with constrained margins is that the choice between PP and FF null models (as similarly between PE and FE , or between EP and EF) should come from a guiding principle, i.e., an assumption about what specification is theoretically more appropriate, and cannot be left to mathematical or numerical convenience [25,26,74]. Clearly, such guiding principle should focus primarily on the following question: In its “null behavior,” i.e., in absence of the higher-order patterns we are looking for in the data, do we expect the system to be equally represented by alternative configurations where the margins chosen as constraints are kept fixed *exactly*, or only *on average*? In other words, do we expect the constraints to *fluctuate around their mean* under the null hypothesis? As we exemplify below, there are arguments in favor of choosing hard constraints, and other arguments in favor of choosing soft constraints. Finally, it is possible to resort to model selection to make a choice in absence of a prior guiding principle.

A theoretical guiding principle in favor of enforcing constraints in a hard manner is the 2-fold confidence that (i) the experimentally measured values of the constraints themselves are error-free (so that the empirical values coincide with the true values), and (ii) the same system, under the null hypothesis of absence of higher-order structural patterns, would retain exactly the same values for the constraints. Indeed, if one is confident that both hypotheses are true for the specific system at hand, it would make no sense to let the values of the constraints

themselves fluctuate, because in that way, the null model would explore matrix configurations that neither the real network nor its randomized variants are expected to exhibit [74,75]. Since the statistical power of matrix randomization techniques generally increases as the constraints are enforced in a stricter way [85], one might argue that, under the two hypotheses above, it is important to enforce the constraints in a hard way.

In the other side of the same coin, one finds a principled argument in favor of enforcing constraints in a soft manner. Indeed, if one has reasons to assume that, as in virtually all experimental sciences, the observed network data are affected by error or noise, one has to conclude that the empirical values of the constraints are different from (albeit hopefully close to) the corresponding true values that one would observe in absence of noise; e.g., if some associations between plants and pollinators in an ecological networks are not observed because of poor sampling, or if spurious associations are incorrectly interpreted as observed, the number of 1s along the rows or columns of the empirical rectangular matrix might be smaller or larger than the true value. In such a situation, enforcing constraints in a hard manner would paradoxically give, in the null model, zero probability to the “true” configuration and to all configurations with the same margins. By contrast, enforcing constraints in a soft way would instead let the null model capture the true configuration and give it only a slightly smaller probability (if the errors in the data are small) with respect to the configurations sampled under hard constraints [26,74,75]. The same would happen for all other configurations where the values of the constraints are close to, even if not necessarily equal to, the empirically observed values. In the ecological context, this observation has led to the consideration that, due to their intrinsic variability, real-world bipartite networks might be inherently understood as realizations of a process described by soft constraints [41].

Finally, if there is no theoretical prior expectation available about the presence or absence of noise in the empirical values of the constraints, one might argue that the decision between hard and soft constraints should be based on posterior evidence, i.e., on which of the two null models achieves the best fit to the data. The state of the art in statistical model selection, which is based on (variations of) the *minimum description length* principle [114], identifies the best model as the one that achieves the best combination of accuracy and parsimony: Among models with the same complexity, it should be the model with maximum likelihood; however, for models with different complexity (e.g., different numbers of parameters, or different uses of the same number of parameters), it should be the model with maximum difference between likelihood and complexity. Model complexity is, therefore, a penalizing term to be subtracted from the likelihood, to reduce the risk of *overfitting*, i.e., greedily achieving higher likelihood via the introduction of too many parameters that, however, may end up fitting a particular, contingent realization of the randomness, rather than the structural information behind it [114]. Recent results have indicated that, given the same constraints, “hard” (microcanonical) null models have always higher likelihood, but also higher complexity, than the corresponding “soft” (canonical) null models [76]. The net result, i.e., the best-scoring model in terms of realized difference between likelihood and complexity, is surprisingly found to depend crucially on the numerical values of the constraints: In particular, for bipartite matrices with given row and/or column margins, this means that whether the best fitting model is the “fixed” (FE, EF, or FF) or the corresponding “proportional” (PE, EP, or PP, respectively) variant of the null model depends on the specific observed values of the marginals defining the model themselves [76]. In this perspective, one should therefore simply input the empirical margins into a formula that calculates the model score (likelihood minus complexity) and identify the best-scoring null model accordingly.

Concluding remarks

In this review, we have provided readers from different fields (such as, but not limited to, mathematics, physics, social sciences, and ecology) with the conceptual tools needed to properly embark in matrix randomization exercises. Our hope is that it could also help unifying future theoretical and applied research and avoid the accumulation of further confusion due to duplicated efforts from different disciplines. However, there are various, additional outstanding issues and open questions that we could not discuss here but which we deem essential mentioning.

In most practical situations, and especially when referring to the natural world, detecting the existence of a link between two nodes (e.g., by observing a pollinator's visit to a flower or by detecting the presence of a parasite on a host) is much easier than quantifying the strength of the underlying association (e.g., the actual importance of the pollinator for the target plant, or the prevalence of the parasite species in the target host's population). Furthermore, different quantities could often be attributed to the same 0/1 link, depending on the specific process that the target interaction represents; e.g., binary links connecting pollinators to plants might be associated to quantitative measures of pollinator preference but also of pollination efficiency [115]. As a consequence, there is a disproportion of studies—and tools—on binary matrices compared to quantitative ones.

Nevertheless, although presence-absence matrices can be used to represent many, different entities, 0s and 1s cannot capture all the nuances and complexity, which permeate the real world. Now, despite the challenges mentioned above, also thanks to novel tools and technologies, quantitative matrices are becoming increasingly available in many fields, and there is a growing recognition of the fact that weighting interactions might reveal different structural patterns from those identified after translating the same data into 0/1 links [116]. The analysis of structural patterns in quantitative matrices might also require the use of randomization techniques. However, identifying a well-defined set of criteria and constraints to be applied to the randomization procedures for quantitative matrices is not straightforward and present many more possible cases than those identified for binary matrices (as in the classification scheme proposed in Fig 4). Since quantitative matrices underlie a binary structure (since each cell can be identified as either occupied or not), one could ideally perform randomizations by applying the same principles and techniques developed for 0/1 matrices. However, in doing that, one should also decide whether to apply random changes to the individual values within each cell or to preserve the original values while randomizing their position within the matrix, or to combine the two approaches. This opens up an extremely wide spectrum of possibilities, which becomes even wider when one starts thinking at possible alternative criteria and rules to modify (or not) the cell values [117]. Remarkably, some of the approaches described here have already been straightforwardly extended to weighted bipartite networks, both in abstract models [75] and in applications to, e.g., financial [61,62], economic [118,119], rating [45], and ecological [41] systems.

Another obvious limitation of presence-absence or, more in general, of rectangular matrices is that they can only capture a single feature of the system they are representing. That is to say, a rectangular matrix representing species occurrences across localities cannot provide any information on species and localities going beyond those we can directly derive from matrix structure; e.g., the matrix can tell us whether a given location has a high species richness, or whether a species is rare (clearly with specific reference to the set of localities included in the matrix). But it cannot tell us anything more about other “exogenous” features of species and localities. However, there are many possible contexts where such features not captured by the matrix itself could be relevant in the context of pattern detection. Such features might be used

to define additional, external constraints. In addition to “endogenous” criteria for randomization looking at specific network/matrix properties, we might consider also “exogenous” criteria based on properties of the entities represented in the matrix, which cannot be inferred by the matrix itself, but which can be made available by additional datasets. These can be, for example, additional data matrices with information relating to the rows or columns of the target matrix. This kind of information might be absolute, which is a simple covariate or attribute of an individual component, or relative, which is a measure relating the component to other components. Considering again the species per locality matrix example, one could associate covariate vectors representing individual traits (body size, geographic range) to species, as well as relative measures such as the phylogenetic relationship of each species to all others. For sites, there might be physical covariates (soil nutrients, island area), but also relative measures such as the pairwise distances between all possible pairs of locations. The problem is particularly compelling in ecology, where it has led to the development of a few statistical approaches trying to improve pattern detection in species per site matrices by pairing them to additional information, incorporating, e.g., species functional traits and/or environmental characteristics of the sites [120]. Although the implementation of exogenous constraints into “typical” randomization strategies might appear more as a conceptual challenge that one could tackle in practice by simply adding a few more lines to extant algorithms, the issue might be much more delicate. In fact, the additional constraints could modify fundamental properties of robust randomization techniques and lead to unexpected (and hardly detectable) biases. This calls for a more in-depth and formally structured investigation of the problem to identify potential extensions of extant, robust algorithms ideally capable of preserving the desired qualities of their original counterparts while accommodating extra rules dictated by features external to the target matrix. In this review, we focused on a set of different approaches to generate matrices under different constraints, such as that of ensuring that the randomized matrices generated by the selected approach have some predefined marginal totals. Such constraints, in principle, should permit a user’s need to replicate the potential effects of real-world or hypothetical processes on matrix structure; e.g., by constraining column totals when applying a randomization algorithm to a species \times locality matrix, one would generate random matrices where species richness in each locality matches that of the initial matrix. In turn, that would ideally make it possible to compare the structure of the starting matrix with that of the randomized matrices while controlling for all the processes affecting local species richness. However, depending on the research questions, the actual study setting and the nature of the data under investigation, the choice of which structural properties of the matrix to be controlled for in the randomization process might be not so obvious and, possibly, biased by subjectivity. Additionally, one might be interested in exploring a gradient of assumptions and multiple scenarios. In such a context, a potential solution is that of comparing the target matrix not just with a specific set of null matrices obtained using a specific algorithm but, instead, with multiple sets of null matrices covering a larger—ideally continuous—portion of the null space entailed by different, specific combinations of constraint; e.g., in the ecological context, an algorithm has been proposed [20] to explore thoroughly the null space delimited by the 9 different combinations of constraints we focused on in this review (see Fig 4). Such an algorithm can produce a “bidimensional landscape” of significance and effect size, in contrast with the typical single significance and effect size values provided by the standard null model analysis focusing on a specific set of constraints. The landscapes of significance/effect size offer a more comprehensive and less subjective representation of matrix structural patterns, by showing how the intensity and significance of such patterns vary under a continuous range of different hypotheses. Similarly, we have mentioned various theoretical guiding principles aiding the choice of whether constraints should be enforced in a hard or soft manner (the two choices

leading in general to different statistical conclusions [26,30]). In absence of a clear preference for any of these guiding principles, the choice can be left to posterior evidence by selecting the best-scoring null model in terms of the optimal combination of accuracy and parsimony [76].

While many open questions remain regarding the randomization of matrices and the detection of patterns in bipartite networks, we hope this review lays the groundwork for future research on these topics. Two directions for future research are particularly pressing. First, to facilitate researchers' use of these methods, it will be important to develop a standardized library of efficient implementations of each of these randomization methods. Second, to inform the selection of randomization methods, it will be important to systematically compare the circumstances under which each method is most appropriate, and specifically when different methods may be more or less able to detect patterns.

Acknowledgments

We thank Filippo Radicchi for his valuable suggestions on this work. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funders.

Author Contributions

Conceptualization: Zachary P. Neal, Annabell Cadieux, Diego Garlaschelli, Nicholas J. Gotelli, Fabio Saracco, Tiziano Squartini, Shade T. Shutters, Werner Ulrich, Guanyang Wang, Giovanni Strona.

Data curation: Zachary P. Neal, Giovanni Strona.

Supervision: Zachary P. Neal, Diego Garlaschelli, Giovanni Strona.

Visualization: Giovanni Strona.

Writing – original draft: Zachary P. Neal, Annabell Cadieux, Diego Garlaschelli, Nicholas J. Gotelli, Fabio Saracco, Tiziano Squartini, Shade T. Shutters, Werner Ulrich, Guanyang Wang, Giovanni Strona.

Writing – review & editing: Zachary P. Neal, Annabell Cadieux, Diego Garlaschelli, Nicholas J. Gotelli, Fabio Saracco, Tiziano Squartini, Shade T. Shutters, Werner Ulrich, Guanyang Wang, Giovanni Strona.

References

1. Gotelli NJ. Null model analysis of species co-occurrence patterns. *Ecology*. 2000; 81(9):2606–2621.
2. Kim J, Mouw KW, Polak P, Braunstein LZ, Kamburov A, Tiao G, et al. Somatic ERCC2 mutations are associated with a distinct genomic signature in urothelial tumors. *Nat Genet*. 2016; 48(6):600–606. <https://doi.org/10.1038/ng.3557> PMID: 2711103
3. Saracco F, Di Clemente R, Gabrielli A, Squartini T. Randomizing bipartite networks: the case of the World Trade Web. *Sci Rep*. 2015; 5(1):1–18. <https://doi.org/10.1038/srep10595> PMID: 26029820
4. Chen YZ, Li N, He DR. A study on some urban bus transport networks. *Physica A Stat Mech Appl*. 2007; 376:747–754.
5. Lambiotte R, Ausloos M. Uncovering collective listening habits and music genres in bipartite networks. *Phys Rev E*. 2005; 72(6):066107. <https://doi.org/10.1103/PhysRevE.72.066107> PMID: 16486010
6. Smiljanic J, Mitrović DM. Associative nature of event participation dynamics: A network theory approach. *PLoS ONE*. 2017; 12(2):e0171565. <https://doi.org/10.1371/journal.pone.0171565> PMID: 28166305
7. Straka MJ, Caldarelli G, Squartini T, Saracco F. From ecology to finance (and back?): A review on entropy-based null models for the analysis of bipartite networks. *J Stat Phys*. 2018; 173:1252–1285.

8. Guillaume JL, Latapy M. Bipartite graphs as models of complex networks. *Physica A Stat Mech Appl.* 2006; 371(2):795–813. <https://doi.org/10.1016/j.physa.2006.04.047>
9. Budel G, Kitsak M. Complementarity in complex networks. *arXiv [Preprint].* 2020;arXiv:200306665.
10. Talaga S, Nowak A. Structural measures of similarity and complementarity in complex networks. *Sci Rep.* 2022; 12(1):16580. <https://doi.org/10.1038/s41598-022-20710-w> PMID: 36195736
11. Mattsson CE, Takes FW, Heemskerk EM, Diks C, Buiten G, Faber A, et al. Functional structure in production networks. *Front Big Data.* 2021; 4:666712. <https://doi.org/10.3389/fdata.2021.666712> PMID: 34095822
12. Budel G, Jin Y, Van Mieghem P, Kitsak M. Topological properties and organizing principles of semantic networks. *Sci Rep.* 2023; 13(1):11728. <https://doi.org/10.1038/s41598-023-37294-8> PMID: 37474614
13. Newman ME. Coauthorship networks and patterns of scientific collaboration. *Proc Natl Acad Sci.* 2004; 101(suppl 1):5200–5205. <https://doi.org/10.1073/pnas.0307545100> PMID: 14745042
14. Neal ZP. A sign of the times? Weak and strong polarization in the US Congress, 1973–2016. *Soc Netw.* 2020; 60:103–112. <https://doi.org/10.1016/j.socnet.2018.07.007>
15. Zhang B, Horvath S. A general framework for weighted gene co-expression network analysis. *Stat Appl Genet Mol Biol.* 2005; 4(1). <https://doi.org/10.2202/1544-6115.1128> PMID: 16646834
16. Freilich S, Kreimer A, Meilijson I, Gophna U, Sharan R, Ruppin E. The large-scale organization of the bacterial network of ecological co-occurrence interactions. *Nucleic Acids Res.* 2010; 38(12):3857–3868. <https://doi.org/10.1093/nar/gkq118> PMID: 20194113
17. Vasques Filho D, O’Neale DRJ. Transitivity and degree assortativity explained: The bipartite structure of social networks. *Phys Rev E.* 2020; 101:052305. <https://doi.org/10.1103/PhysRevE.101.052305> PMID: 32575287
18. Guillaume JL, Latapy M. Bipartite structure of all complex networks. *Inf Process Lett.* 2004; 90(5):215–221.
19. Newman ME, Park J. Why social networks are different from other types of networks. *Phys Rev E.* 2003; 68(3):036122. <https://doi.org/10.1103/PhysRevE.68.036122> PMID: 14524847
20. Strona G, Ulrich W, Gotelli NJ. Bi-dimensional null model analysis of presence-absence binary matrices. *Ecology.* 2018; 99(1):103–115. <https://doi.org/10.1002/ecy.2043> PMID: 29023670
21. Neal Z. The backbone of bipartite projections: Inferring relationships from co-authorship, co-sponsorship, co-attendance and other co-behaviors. *Soc Netw.* 2014; 39:84–97.
22. Zweig KA, Kaufmann M. A systematic approach to the one-mode projection of bipartite graphs. *Soc Netw Anal Min.* 2011; 1(3):187–218.
23. Cimini G, Squartini T, Saracco F, Garlaschelli D, Gabrielli A, Caldarelli G. The statistical physics of real-world networks. *Nat Rev Phys.* 2019; 1(1):58–71.
24. Squartini T, Garlaschelli D. Analytical maximum-likelihood method to detect patterns in real networks. *New J Phys.* 2011; 13. <https://doi.org/10.1088/1367-2630/13/8/083001>
25. Squartini T, Garlaschelli D. Maximum-Entropy Networks: Pattern Detection, Network Reconstruction and Graph Combinatorics. Springer; 2017.
26. Squartini T, Mastrandrea R, Garlaschelli D. Unbiased sampling of network ensembles. *New J Phys.* 2015; 17(2):023052.
27. Zhang P, Wang J, Li X, Li M, Di Z, Fan Y. Clustering coefficient and community structure of bipartite networks. *Physica A Stat Mech Appl.* 2008; 387(27):6869–6875. <https://doi.org/10.1016/j.physa.2008.09.006>
28. Barber MJ. Modularity and community detection in bipartite networks. *Phys Rev E.* 2007; 76(6):066102. <https://doi.org/10.1103/PhysRevE.76.066102> PMID: 18233893
29. Strona G, Fattorini S. On the methods to assess significance in nestedness analyses. *Theory Biosci.* 2014; 133:179–186. <https://doi.org/10.1007/s12064-014-0203-1> PMID: 24974139
30. Bruno M, Saracco F, Garlaschelli D, Tessone CJ, Caldarelli G. The ambiguity of nestedness under soft and hard constraints. *Sci Rep.* 2020; 10:1–13. <https://doi.org/10.1038/s41598-020-76300-1> PMID: 33199720
31. Daminelli S, Thomas JM, Durán C, Cannistraci CV. Common neighbours and the local-community-paradigm for topological link prediction in bipartite networks. *New J Phys.* 2015; 17(11):113037. <https://doi.org/10.1088/1367-2630/17/11/113037>
32. Domínguez-García V, Muñoz MA. Ranking species in mutualistic networks. *Sci Rep.* 2015; 5(1):8182. <https://doi.org/10.1038/srep08182> PMID: 25640575
33. Gotelli NJ, Ulrich W. Statistical challenges in null model analysis. *Oikos.* 2012; 121(2):171–180.

34. Gotelli NJ, McCabe DJ. Species co-occurrence: a meta-analysis of JM Diamond's assembly rules model. *Ecology*. 2002; 83(8):2091–2096.
35. Ulrich W. Species co-occurrences and neutral models: reassessing JM Diamond's assembly rules. *Oikos*. 2004; 107(3):603–609.
36. Stone L, Roberts A. The checkerboard score and species distributions. *Oecologia*. 1990; 85(1):74–79. <https://doi.org/10.1007/BF00317345> PMID: 28310957
37. Gilpin ME, Diamond JM. Factors contributing to non-randomness in species co-occurrences on islands. *Oecologia*. 1982; 52(1):75–84. <https://doi.org/10.1007/BF00349014> PMID: 28310111
38. Patterson BD, Atmar W. Nested subsets and the structure of insular mammalian faunas and archipelagos. *Biol J Linn Soc*. 1986; 28(1–2):65–82.
39. Bascompte J, Jordano P, Melián CJ, Olesen JM. The nested assembly of plant–animal mutualistic networks. *Proc Natl Acad Sci*. 2003; 100(16):9383–9387. <https://doi.org/10.1073/pnas.1633576100> PMID: 12881488
40. Payrát-Borràs C, Hernández L, Moreno Y. Breaking the Spell of Nestedness: The Entropic Origin of Nestedness in Mutualistic Systems. *Phys Rev X*. 2019; 9:031024. <https://doi.org/10.1103/PhysRevX.9.031024>
41. Caruso T, Rillig MC, Garlaschelli D. Fluctuating ecological networks: A synthesis of maximum-entropy approaches for pattern detection and process inference. *Methods Ecol Evol*. 2022; 13(11):2306–2317.
42. Morales-Castilla I, Matias MG, Gravel D, Araújo MB. Inferring biotic interactions from proxies. *Trends Ecol Evol*. 2015; 30(6):347–356. <https://doi.org/10.1016/j.tree.2015.03.014> PMID: 25922148
43. Blanchet FG, Cazelles K, Gravel D. Co-occurrence is not evidence of ecological interactions. *Ecol Lett*. 2020; 23(7):1050–1063. <https://doi.org/10.1111/ele.13525> PMID: 32429003
44. Breiger RL. The duality of persons and groups. *Soc Forces*. 1974; 53(2):181–190. <https://doi.org/10.1093/sf/53.2.181>
45. Becatti C, Caldarelli G, Saracco F. Entropy-based randomization of rating networks. *Phys Rev E*. 2019; 99(2):022306. <https://doi.org/10.1103/PhysRevE.99.022306> PMID: 30934284
46. Faust K. Centrality in affiliation networks. *Soc Netw*. 1997; 19(2):157–191. [https://doi.org/10.1016/S0378-8733\(96\)00300-0](https://doi.org/10.1016/S0378-8733(96)00300-0)
47. Wang P, Pattison P, Robins G. Exponential random graph model specifications for bipartite networks—A dependence hierarchy. *Soc Netw*. 2013; 35(2):211–222. <https://doi.org/10.1016/j.socnet.2011.12.004>
48. Saracco F, Straka MJ, Clemente RD, Gabrielli A, Caldarelli G, Squartini T. Inferring monopartite projections of bipartite networks: An entropy-based approach. *New J Phys*. 2017. <https://doi.org/10.1088/1367-2630/aa6b38>
49. Rasch G. Studies in mathematical psychology: I. Probabilistic models for some intelligence and attainment tests. Nielsen & Lydiche; 1960.
50. Rasch G. Probabilistic models for some intelligence and attainment tests. ERIC; 1993.
51. Verhelst ND. An efficient MCMC algorithm to sample binary matrices with fixed marginals. *Psychometrika*. 2008; 73(4):705–728. <https://doi.org/10.1007/s11336-008-9062-3>
52. Borsboom D, Deserno MK, Rheinländer M, Epskamp S, Fried EI, McNally RJ, et al. Network analysis of multivariate data in psychological science. *Nat Rev Methods Primers*. 2021; 1(1):1–18. <https://doi.org/10.1038/s43586-021-00055-w>
53. Neal ZP, Forbes MK, Neal JW, Brusco MJ, Krueger R, Markon K, et al. Critiques of network analysis of multivariate data in psychological science. *Nat Rev Methods Primers*. 2022; 2(1):1–2. <https://doi.org/10.1038/s43586-022-00177-9>
54. Hidalgo CA, Klinger B, Barabasi AL, Hausmann R. The product space conditions the development of nations. *Science*. 2007; 317(5837):482–487. <https://doi.org/10.1126/science.1144581> PMID: 17656717
55. Hidalgo CA, Hausmann R. The building blocks of economic complexity. *Proc Natl Acad Sci U S A*. 2009; 106:10570–10575. <https://doi.org/10.1073/pnas.0900943106> PMID: 19549871
56. Hausmann R, Hidalgo CA. The network structure of economic output. *J Econ Growth*. 2011; 16:309–342.
57. Tacchella A, Cristelli M, Caldarelli G, Gabrielli A, Pietronero L. A New Metrics for Countries' Fitness and Products' Complexity. *Sci Rep*. 2012; 2:1–4. <https://doi.org/10.1038/srep00723> PMID: 23056915
58. Caldarelli G, Cristelli M, Gabrielli A, Pietronero L, Scala A, Tacchella A. A Network Analysis of Countries' Export Flows: Firm Grounds for the Building Blocks of the Economy. *PLoS ONE*. 2012; 7:1–17. <https://doi.org/10.1371/journal.pone.0047278> PMID: 23094044

59. Cristelli M, Gabrielli A, Tacchella A, Caldarelli G, Pietronero L. Measuring the Intangibles: A Metrics for the Economic Complexity of Countries and Products. *PLoS ONE*. 2013; 8(8):e70726. <https://doi.org/10.1371/journal.pone.0070726> PMID: 23940633
60. Cristelli M, Tacchella A, Pietronero L. The heterogeneous dynamics of economic complexity. *PLoS ONE*. 2015; 10(2):e0117174. <https://doi.org/10.1371/journal.pone.0117174> PMID: 25671312
61. Di Gangi D, Lillo F, Pirino D. Assessing systemic risk due to fire sales spillover through maximum entropy network reconstruction. *J Econ Dyn Control*. 2018; 94:117–141.
62. Squartini T, Almog A, Caldarelli G, Van Lelyveld I, Garlaschelli D, Cimini G. Enhanced capital-asset pricing model for the reconstruction of bipartite financial networks. *Phys Rev E*. 2017; 96(3):032315. <https://doi.org/10.1103/PhysRevE.96.032315> PMID: 29347051
63. Alabdulkareem A, Frank MR, Sun L, AlShebli B, Hidalgo C, Rahwan I. Unpacking the polarization of workplace skills. *Sci Adv*. 2018; 4(7):eaao6030. <https://doi.org/10.1126/sciadv.aao6030> PMID: 30035214
64. Kok S, Bt W. Cities, tasks, and skills. *J Reg Sci*. 2014; 54(5):856–892.
65. Neffke F, Henning M, Boschma R. How do regions diversify over time? Industry relatedness and the development industry relatedness and the development. *Econ Geogr*. 2011; 87(3):237–265.
66. O’Clery N, Heroy S, Hulot F, Beguerisse-Díaz M. Unravelling the forces underlying urban industrial agglomeration. *arXiv [Preprint]*. 2019;1903.09279v2.
67. Galetti JR, Tessarin MS, Morceiro PC. Types of occupational relatedness and branching processes across Brazilian regions. *Area Dev Policy*. 2022;1–23.
68. Muneepakul R, Lobo J, Shutters ST, Goméz-Liévano A, Qubbaj MR. Urban economies and occupation space: Can they get “there from “here? *PLoS ONE*. 2013; 8(9):e73676. <https://doi.org/10.1371/journal.pone.0073676> PMID: 24040021
69. Tóth G, Elekes Z, Whittle A, Lee C, Kogler DF. Technology network structure conditions the economic resilience of regions. *Econ Geogr*. 2022; 98(4):355–378.
70. O’Neale DR, Hendy SC, Vasques FD. Structure of the Region-Technology Network as a Driver for Technological Innovation. *Front Big Data*. 2021; 4:689310. <https://doi.org/10.3389/fdata.2021.689310> PMID: 34337398
71. Almeida-Neto M, Guimaraes P, Guimaraes PR Jr, Loyola RD, Ulrich W. A consistent metric for nestedness analysis in ecological systems: reconciling concept and measurement. *Oikos*. 2008; 117(8):1227–1239.
72. Tuomisto H. A diversity of beta diversities: straightening up a concept gone awry. Part 2. Quantifying beta diversity and related phenomena. *Ecography*. 2010; 33(1):23–45.
73. Squartini T, de Mol J, den Hollander F, Garlaschelli D. Breaking of ensemble equivalence in networks. *Phys Rev Lett*. 2015; 115(26):268701. <https://doi.org/10.1103/PhysRevLett.115.268701> PMID: 26765034
74. Garlaschelli D, den Hollander F, Roccaverde A. Ensemble nonequivalence in random graphs with modular structure. *J Phys A Math Theor*. 2016; 50(1):015001. <https://doi.org/10.1088/1751-8113/50/1/015001>
75. Zhang Q, Garlaschelli D. Strong ensemble nonequivalence in systems with local constraints. *New J Phys*. 2022; 24(4):043011.
76. Giuffrida F, Squartini T, Grünwald P, Garlaschelli D. Description length of canonical and microcanonical models. *arXiv:2307.05645v2 [Preprint]*. 2023.
77. Blanchet J, Stauffer A. Characterizing optimal sampling of binary contingency tables via the configuration model. *Random Struct Algorithms*. 2013; 42(2):159–184. <https://doi.org/10.1002/rsa.20403>
78. Ryser HJ. Combinatorial properties of matrices of zeros and ones. *Can J Math*. 1957; 9:371–377.
79. Strona G, Nappo D, Boccacci F, Fattorini S, San-Miguel-Ayanz J. A fast and unbiased procedure to randomize ecological binary matrices with fixed row and column totals. *Nat Commun*. 2014; 5(1):1–9. <https://doi.org/10.1038/ncomms5114> PMID: 24916345
80. Carstens CJ, Berger A, Strona G. A unifying framework for fast randomization of ecological networks with fixed (node) degrees. *MethodsX*. 2018; 5:773–780. <https://doi.org/10.1016/j.mex.2018.06.018> PMID: 30094204
81. Carstens CJ. Proof of uniform sampling of binary matrices with fixed row sums and column sums for the fast curveball algorithm. *Phys Rev E*. 2015; 91(4):042812. <https://doi.org/10.1103/PhysRevE.91.042812> PMID: 25974552
82. Carstens CJ, Kleer P. Comparing the switch and curveball Markov chains for sampling binary matrices with fixed marginals. *arXiv:170907290 [Preprint]*. 2017.
83. Aldous DJ, Diaconis P. Shuffling cards and stopping-times. *Am Math Mon*. 1986; 93:333–348.

84. Knuth DE. *The art of computer programming: Volume 3: Sorting and Searching*. Addison-Wesley Professional; 1998.
85. Neal ZP, Domagalski R, Sagan B. Comparing alternatives to the fixed degree sequence model for extracting the backbone of bipartite projections. *Sci Rep*. 2021; 11(1):1–13.
86. Barvinok A. On the number of matrices and a random matrix with prescribed row and column sums and 0–1 entries. *Adv Math*. 2010; 224(1):316–339.
87. Squartini T, Garlaschelli D. Reconnecting statistical physics and combinatorics beyond ensemble equivalence. *arXiv:171011422 [Preprint]*. 2017.
88. Gale D. A theorem on flows in networks. *Pacific J Math*. 1957; 7(2):1073–1082.
89. Coolen AC, De Martino A, Annibale A. Constrained Markovian dynamics of random graphs. *J Stat Phys*. 2009; 136:1035–1067.
90. Roberts E, Coolen A. Unbiased degree-preserving randomization of directed binary networks. *Phys Rev E*. 2012; 85(4):046103. <https://doi.org/10.1103/PhysRevE.85.046103> PMID: 22680534
91. Artzy-Randrup Y, Stone L. Generating uniformly distributed random networks. *Phys Rev E*. 2005; 72(5):056708. <https://doi.org/10.1103/PhysRevE.72.056708> PMID: 16383786
92. Sanderson JG, Moulton MP, Selfridge RG. Null matrices and the analysis of species co-occurrences. *Oecologia*. 1998; 116(1–2):275–283. <https://doi.org/10.1007/s004420050589> PMID: 28308537
93. Gotelli NJ, Entsminger GL. Swap and fill algorithms in null model analysis: rethinking the knight's tour. *Oecologia*. 2001; 129(2):281–291. <https://doi.org/10.1007/s004420100717> PMID: 28547607
94. Chen Y, Diaconis P, Holmes SP, Liu JS. Sequential Monte Carlo methods for statistical analysis of tables. *J Am Stat Assoc*. 2005; 100(469):109–120.
95. Chen XH, Dempster AP, Liu JS. Weighted finite population sampling to maximize entropy. *Biometrika*. 1994; 81(3):457–469.
96. Brewer KR, Hanif M. Sampling with unequal probabilities. Springer Science & Business Media; 2013, vol. 15.
97. Roberts A, Stone L. Island-sharing by archipelago species. *Oecologia*. 1990; 83(4):560–567. <https://doi.org/10.1007/BF00317210> PMID: 28313193
98. Maslov S, Sneppen K, Zaliznyak A. Detection of topological patterns in complex networks: correlation profile of the internet. *Physica A Stat Mech Appl*. 2004; 333:529–540.
99. Maslov S, Sneppen K. Specificity and stability in topology of protein networks. *Science*. 2002; 296(5569):910–913. <https://doi.org/10.1126/science.1065103> PMID: 11988575
100. Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, Chklovskii D, Alon U. Network motifs: simple building blocks of complex networks. *Science*. 2002; 298(5594):824–827. <https://doi.org/10.1126/science.298.5594.824> PMID: 12399590
101. Stouffer DB, Camacho J, Jiang W, Nunes Amaral LA. Evidence for the existence of a robust pattern of prey selection in food webs. *Proc R Soc Lond B Biol Sci*. 2007; 274(1621):1931–1940. <https://doi.org/10.1098/rspb.2007.0571> PMID: 17567558
102. Fayle TM, Manica A. Reducing over-reporting of deterministic co-occurrence patterns in biotic communities. *Ecol Model*. 2010; 221(19):2237–2242.
103. Gotelli NJ, Ulrich W, et al. Over-reporting bias in null model analysis: a response to Fayle and Manica (2010). *Ecol Model*. 2011; 222(7):1337–1339.
104. Fayle TM, Manica A. Bias in null model analyses of species co-occurrence: a response to Gotelli and Ulrich (2011). *Ecol Model*. 2011; 222(7):1340–1341.
105. Besag J, Clifford P. Generalized monte carlo significance tests. *Biometrika*. 1989; 76(4):633–642.
106. Manly BF. A note on the analysis of species co-occurrences. *Ecology*. 1995; 76(4):1109–1115.
107. Godard K, Neal ZP. fastball: A fast algorithm to sample binary matrices with fixed marginals. *J Complex Netw*. 2022. <https://doi.org/10.1093/comnet/cnac047>
108. Vallarano N, Bruno M, Marchese E, Trapani G, Saracco F, Cimini G, et al. Fast and scalable likelihood maximization for exponential random graph models with local constraints. *Sci Rep*. 2021; 11(1):1–33.
109. Ulrich W, Gotelli NJ. A null model algorithm for presence–absence matrices based on proportional resampling. *Ecol Model*. 2012; 244:20–27.
110. Patterson B, Atmar W. Analyzing species composition in fragments. *Bonn Zool Monogr*. 2000; 46:9–24.
111. Cover TM, Thomas JA. *Elements of Information Theory* 2nd Edition (Wiley Series in Telecommunications and Signal Processing). Wiley-Interscience; 2006.

112. Touchette H. Equivalence and nonequivalence of ensembles: Thermodynamic, macrostate, and measure levels. *J Stat Phys.* 2015; 159:987–1016.
113. Bruno M, Lambiotte R, Saracco F. Brexit and bots: characterizing the behaviour of automated accounts on Twitter during the UK election. *EPJ Data Sci.* 2022; 11:1–24. <https://doi.org/10.1140/epjds/s13688-022-00330-0> PMID: 35340571
114. Grünwald PD. The minimum description length principle. MIT press; 2007.
115. Strona G. Ecological Networks. Hidden Pathways to Extinction. Springer; 2022, pp. 41–55.
116. Staniczenko P, Kopp JC, Allesina S. The ghost of nestedness in ecological networks. *Nat Commun.* 2013; 4(1):1–6. <https://doi.org/10.1038/ncomms2422> PMID: 23340431
117. Ulrich W, Gotelli NJ. Null model analysis of species associations using abundance data. *Ecology.* 2010; 91(11):3384–3397. <https://doi.org/10.1890/09-2157.1> PMID: 21141199
118. Krantz R, Gemmetto V, Garlaschelli D. Maximum-entropy tools for economic fitness and complexity. *Entropy.* 2018; 20(10):743. <https://doi.org/10.3390/e20100743> PMID: 33265832
119. Bruno M, Mazzilli D, Patelli A, Squartini T, Saracco F. Inferring comparative advantage via entropy maximization. *J Phys Complex.* 2023; 4(4):045011.
120. Ulrich W, Kryszewski W, Sewerniak P, Puchałka R, Strona G, Gotelli NJ. A comprehensive framework for the study of species co-occurrences, nestedness and turnover. *Oikos.* 2017; 126(11):1607–1616.