The ground truth about metadata and community detection in networks

Leto Peel,1,2,* Daniel B. Larremore,3,† and Aaron Clauset4,5,3,‡

1ICTEAM, Université Catholique de Louvain, Louvain-la-Neuve, Belgium
2naXys, Université de Namur, Namur, Belgium
3Santa Fe Institute, Santa Fe, NM 87501, USA
4Department of Computer Science, University of Colorado, Boulder, CO 80309, USA
5BioFrontiers Institute, University of Colorado, Boulder, CO 80309, USA

Across many scientific domains, there is a common need to automatically extract a simplified view or coarse-graining of how a complex system’s components interact. This general task is called community detection in networks and is analogous to searching for clusters in independent vector data. It is common to evaluate the performance of community detection algorithms by their ability to find so-called ground truth communities. This works well in synthetic networks with planted communities because such networks’ links are formed explicitly based on those known communities. However, there are no planted communities in real world networks. Instead, it is standard practice to treat some observed discrete-valued node attributes, or metadata, as ground truth. Here, we show that metadata are not the same as ground truth, and that treating them as such induce severe theoretical and practical problems. We prove that no algorithm can uniquely solve community detection, and we prove a general No Free Lunch theorem for community detection, which implies that there can be no algorithm that is optimal for all possible community detection tasks. However, community detection remains a powerful tool and node metadata still have value so a careful exploration of their relationship with network structure can yield insights of genuine worth. We illustrate this point by introducing two statistical techniques that can quantify the relationship between metadata and community structure for a broad class of models. We demonstrate these techniques using both synthetic and real-world networks, and for multiple types of metadata and community structure.

INTRODUCTION

Community detection is a fundamental task of network science that seeks to describe the large-scale structure of a network by dividing its nodes into communities (also called blocks or groups), based only on the pattern of links. This task is similar to that of clustering vector data, as both seek to identify meaningful groups within some data set.

Community detection has been used productively in many applications, including identifying allegiances or personal interests in social networks [1, 2], biological function in metabolic networks [3, 4], fraud in telecommunications networks [5], and homology in genetic similarity networks [6]. Many approaches to community detection exist, spanning not just different algorithms and partitioning strategies, but also fundamentally different definitions of what it means to be a “community.” This diversity is a strength, as networks generated by different processes and phenomena should not a priori be expected to be well-described by the same structural principles.

With so many different approaches to community detection available, it is natural to compare them to assess their relative strengths and weaknesses. Typically, this comparison is made by assessing a method’s ability to identify so-called ground truth communities, a single partition of the network’s nodes into groups which is considered the correct answer. This approach for evaluating community detection methods works well in artificially generated networks, whose links are explicitly placed according to those ground truth communities and a known data generating process. For this reason, the partition of nodes into ground truth communities in synthetic networks is called a planted partition. However, for real-world networks, both the correct partition and the true data generating process are typically unknown, which necessarily implies that there can be no ground truth communities for real networks. Without access to the very thing these methods are intended to find, objective evaluation of their performance is difficult.

Instead, it has become standard practice to treat some observed data on the nodes of a network, which we call node metadata, (e.g., a person’s ethnicity, gender or affiliation for a social network, or a gene’s functional class for a gene regulatory network) as if they were ground truth communities. While this widespread practice is convenient, it can lead to incorrect scientific conclusions under relatively common circumstances. In this paper, we identify these consequences and articulate the epistemological argument against treating metadata as ground truth communities. Next, we provide rigorous mathematical arguments and prove two theorems that render the search for a universally best ground-truth recovery algorithm as fundamentally flawed. We then present two novel methods that can be used to productively explore the relationship between observed metadata and community structure, and demonstrate both methods on a variety of synthetic and real-world networks, using mul-

* Contributed equally; leto.peel@uclouvain.be
† Contributed equally; larremore@santafe.edu
‡ Contributed equally; aaron.clauset@colorado.edu
multiple community detection frameworks. Through these examples, we illustrate how a careful exploration of the relationship between metadata and community structure can shed light on the role that node attributes play in generating network links in real complex systems.

RESULTS

The trouble with metadata and community detection

The use of node metadata as a proxy for ground truth stems from a reasonable need: since artificial networks may not be representative of naturally occurring networks, community detection methods must also be confronted with real-world examples to show that they work well in practice. If the detected communities correlate with the metadata, we may reasonably conclude that the metadata are involved in or depend upon the generation of the observed interactions. However, the scientific value of a method is as much defined by the way it fails as by its ability to succeed. Because metadata always have an uncertain relationship with ground truth, failure to find a good division that correlates with our metadata is a highly confounded outcome, arising for any of several reasons:

(i) these particular metadata are irrelevant to the structure of the network,
(ii) the detected communities and the metadata capture different aspects of the network’s structure,
(iii) the network contains no communities as in a simple random graph [7] or a network that is sufficiently sparse that its communities are not detectable [8], or
(iv) the community detection algorithm performed poorly.

In the above, we refer to the observed network and metadata and note that noise in either could lead to one of the reasons above. For instance, measurement error of the network structure may make our observations unreliable and in extreme cases can obscure community structure entirely, resulting in case (iii). It is also possible that human errors are introduced when handling the data, exemplified by the widely used American college football network [9] of teams that played each other in one season, whose associated metadata representing each team’s conference assignment were collected during a different season [10]. Large errors in the metadata can render them irrelevant to the network [case (i)].

Most work on community detection assumes that failure to find communities that correlate with metadata implies case (iv), algorithm failure, although some critical work has focused on case (iii), difficult or impossible to recover communities. The lack of consideration for cases (i) and (ii) suggests the possibility for selection bias in the published literature in this area (a point recently suggested by [11]). Indeed, recent critiques of the general utility of community detection in networks [11–13] can be viewed as a side effect of confusion about the role of metadata in evaluating algorithm results.

For these reasons, using metadata to assess the performance of community detection algorithms can lead to errors of interpretation, false comparisons between methods, and oversights of alternative patterns and explanations, including those that do not correlate with the known metadata.

For example, Zachary’s Karate Club [14] is a small real-world network with compelling metadata frequently used to demonstrate community detection algorithms. The network represents the observed social interactions of 34 members of a karate club. At the time of study, the club fell into a political dispute and split into two factions. These faction labels are the metadata commonly used as ground truth communities in evaluating community detection methods. However, it is worth noting at this point that Zachary’s original network and metadata differ from those commonly used for community detection [9]. Links in the original network were by the different types of social interaction that Zachary observed. Zachary also recorded two metadata attributes: the political leaning of each of the members (strong, weak, or neutral support for one of the factions) and the faction they ultimately joined after the split. However, the community detection literature uses only the metadata representing the faction each node joined, often with one of the nodes mislabeled.
This node ("Person number 9") supported the president during the dispute but joined the instructor’s faction as joining the president’s faction would have involved retraining as a novice when he was only two weeks away from taking his black belt exam.

The division of the Karate Club nodes into factions is not the only scientifically reasonable way to partition the network. Figure 1 shows the log-likelihood landscape for a large number of two-group partitions (embedded in two dimensions for visualization) of the Karate Club, under the stochastic blockmodel (SBM) for community detection [15, 16]. Partitions that are similar to each other are embedded nearby in the horizontal coordinates, meaning that the two broad peaks in the landscape represent two distinct sets of high-likelihood partitions, one centered around the faction division and one that divides the network into leaders and followers. Other common approaches to community detection [9, 17], suggest that the best divisions of this network have more than two communities [10, 18]. The multiplicity and diversity of good partitions illustrates the ambiguous status of the faction metadata as a desirable target.

The Karate Club network is among many examples for which standard community detection methods return communities that either subdivide the metadata partition [19] or do not correlate with the metadata at all [20, 21]. More generally, most real-world networks have many good partitions and there are many plausible ways to sort all partitions to find good ones, sometimes leading to a large number of reasonable results. Moreover, there is no consensus on which method to use on which type of network [21, 22].

In what follows, we explore both the theoretical origins of these problems and the practical means to address the confounding cases described above. To do so, we make use of a generative model perspective of community detection. In this perspective, we describe the relationship between community assignments \( \mathcal{C} \) and graphs \( \mathcal{G} \) via a joint distribution \( P(\mathcal{C}, \mathcal{G}) \) over all possible community assignments and graphs that we may observe. We take this perspective because it provides a precise and interpretable description of the relationship between communities and network structure. Although generative models, like the SBM, describe the relationship between networks and communities directly via a mathematically explicit expression for \( P(\mathcal{C}, \mathcal{G}) \), other methods for community detection nevertheless maintain an implicit relationship between network structure and community assignment. As such, the theorems we present, as well as their implications are more generally applicable across all methods of community detection.

In the next section we present rigorous theoretical results with direct implications for cases (i) and (iv), while the remaining sections introduce two statistical methods for addressing cases (i) and (ii), respectively. These contributions do not address case (iii), when there is no structure to be found, which has been previously explored by other authors, e.g., for the SBM [8, 23–27] and modularity [28, 29].

**Ground truth and metadata in community detection**

Community detection is an inverse problem: using only the edges of the network as data, we aim to find the grouping or partition of the nodes that relates to how the network came to be. More formally, suppose that some data generating process \( g \) embeds ground-truth communities \( \mathcal{T} \) in the patterns of links in a network \( \mathcal{G} = g(\mathcal{T}) \). Our goal is to discover those communities based on the observed links. To do so, we write down a community detection scheme \( f \) that uses the network to find communities \( \mathcal{C} = f(\mathcal{G}) \). If we have chosen \( f \) well, then the communities \( \mathcal{C} \) will be equal to the ground truth \( \mathcal{T} \) and we have solved the inverse problem. Thus, the community detection problem for a single network seeks a method \( f^* \) that minimizes the distance between the identified communities and the ground truth:

\[
    f^* = \arg \min_f d(\mathcal{T}, f(\mathcal{G})) ,
\]

where \( d \) is a measure of distance between partitions.

For a method \( f \) to be generally useful, it should be the minimizer for many different graphs, each with its own generative process and ground truth. Often in the community detection literature, several algorithms are tested on a range of networks to identify which performs best overall [12, 30, 31]. If a universally optimal community detection method exists, it must solve Eq. (1) for any type of generative process \( g \) and partition \( \mathcal{T} \), that is,

\[
    \exists f^* \text{ s.t. } f^* = \arg \min_f d(\mathcal{T}, f(\mathcal{g}(\mathcal{T}))) \quad \forall\{g, \mathcal{T}\} .
\]

In fact, no such universal \( f^* \) community detection method can exist because the mapping from generative models \( g \) and ground truth partitions \( \mathcal{T} \) to graphs \( \mathcal{G} \) is not uniquely invertible due to the fact that the map is not a bijection. In other words, any particular network \( \mathcal{G} \) can be produced by multiple, distinct generative processes, each with its own ground truth, such that \( \mathcal{G} = g_1(\mathcal{T}_1) = g_2(\mathcal{T}_2) \), with \( (g_1, \mathcal{T}_1) \neq (g_2, \mathcal{T}_2) \). Thus, no community detection algorithm method can uniquely solve the problem for all possible networks (Eq. (2)), or even a single network (Eq. (1)). This reasoning underpins the following theorem, which we state and prove in Supplementary Text C:

**Theorem 1:** For a fixed network \( \mathcal{G} \), the solution to the ground truth community detection problem—given \( \mathcal{G} \), find the \( \mathcal{T} \) such that \( \mathcal{G} = g(\mathcal{T}) \)—is not unique.

Substituting metadata \( \mathcal{M} \) for ground truth \( \mathcal{T} \) exacerbates the situation by creating additional problems. In real networks we do not know the ground truth or the generating process. Instead, it is common to seek a partition that matches some node metadata \( \mathcal{M} \). Optimizing
a community detection method to discover $\mathcal{M}$ is equivalent to finding $f^*$ such that

$$f^* = \arg \min_f d(\mathcal{M}, f(G)),$$  \hspace{1cm} (3)

yet this does not necessarily solve the community detection problem of Eq. (1) since we cannot guarantee that the metadata are equivalent to the unobserved ground truth, $d(\mathcal{M}, T) = 0$. Consequently, both $d(\mathcal{C}, T) = 0$ and $d(\mathcal{C}, T) > 0$ are possibilities. Thus, when we evaluate a community detection method by its ability to find a metadata partition, we confound the metadata’s correspondence to the true communities, i.e., $d(\mathcal{M}, T)$ [case (ii) in the previous section] and the community detection method’s ability to find true communities, i.e., $d(\mathcal{C}, T)$ [case (iv)]. In this way, treating metadata as ground truth simultaneously tests the metadata’s relevance and the algorithm’s performance, with no ability to differentiate between the two. For instance, when considering competing partitions of the Karate Club (Figure 1), the leader-follower partition is the most likely partition under the SBM, yet it correlates poorly with the known metadata. On the other hand, under the degree-corrected SBM, the optimal partition is more highly correlated with the metadata (Fig. S1). Based only on the performance of recovering metadata, one would conclude that the degree-corrected model is better. However, if Zachary had not provided the faction information, but instead some metadata that correlated with the degree (e.g. the identities of the club’s four officers) then our conclusion might change to the regular SBM being the better model. We would arrive at a different conclusion despite the fact that the network, and the underlying process that generated it, remain unchanged. A similar case of dependence on a particular choice of metadata are exemplified by divisions of high-school social networks using metadata of students’ grade level or race [21]. Past evaluations of community detection algorithms that only measure performance by metadata recovery are thus inconclusive. It is only with synthetic data, where the generative process is known, that ground truth is knowable and performance objectively measurable.

However, even when the generative process is known for a single network or even a set of networks, there is no best community detection method over all networks. This is because, when averaged over all possible community detection problems, every algorithm has provably identical performance, a notion that is captured in a No Free Lunch theorem for community detection which we rigorously state and prove in Supplementary Text C and paraphrase here:

**Theorem 3 (paraphrased):** For the community detection problem, with accuracy measured by adjusted mutual information, the uniform average of the accuracy of any method $f$ over all possible community detection problems is a constant which is independent of $f$.

This No Free Lunch theorem, based on the No Free Lunch theorems for supervised learning [32], implies that no method has an a priori advantage over any other across all possible community detection tasks. (In fact, Theorem 3 and its proof apply to clustering and partitioning methods in general, beyond community detection.) In other words, for a set of cases that a particular method $f_a$ outperforms $f_b$, there must exist a set of cases where $f_b$ outperforms $f_a$—on average no algorithm performs better than any other. Yet this does not render community detection pointless because the theorem also implies that if the tasks of interest correspond to a restricted subset of cases (e.g., finding communities in gene regulatory networks or certain kinds of groups in social networks), then there may indeed be a method that outperforms others within the confines of that subset. In short, matching beliefs about the data generating process $g$ with the assumptions of the algorithm $f$ can lead to better and more accurate results, at the cost of reduced generalizability. (See Supplementary Text C for additional discussion.)

The combined implications of the epistemological arguments in the previous section with Theorems 1 and 3 in this section do not render community detection impossible or useless, by any means. They do, however, imply that efforts to find a universally best community detection algorithm are in vain, and that metadata should not be used as a benchmark for evaluating or comparing the efficacy of community detection algorithms. These theorems indicate that better community detection results may stem from a better understanding of how to divide the problem space into categories of community detection tasks, eventually identifying classes of algorithms whose strengths are aligned with the requirements of a specific category.

**Relating metadata and structure**

From a scientific perspective, metadata labels have direct and genuine value in helping to understand complex systems. Metadata describe the nodes, while communities describe how nodes interact. Therefore, correspondence between metadata and communities suggests a relationship between how nodes interact and the properties of the nodes themselves. This correspondence has been used productively to assist in the inference of community structure [21], learn the relationship between metadata and network topology [33, 34] and explain dependencies between metadata and network structure [35].

Here we propose two new methods to explore how metadata relate to the structure of the network when the metadata only correlate weakly with the identified communities. Both methods utilize the powerful tools of probabilistic models, but are not restricted to any particular model of community structure. The first method is a statistical test to assess whether or not the metadata partition and network structure are related [case (i)]. The
second method explores the space of network partitions to determine if the metadata represent the same or different aspects of the network structure as the “optimal” communities inferred by a chosen model [case (ii)].

In principle, any probabilistic generative model (e.g., [15, 16, 36–39]) of communities in networks could be used within these methods. Here we derive results for the popular stochastic blockmodel [15, 16] and its degree-corrected version [20] (alternative formulations discussed in Supplementary Texts B and A). The SBM defines communities as sets of nodes that are stochastically equivalent. This means that the probability \( p_{ij} \) of a link between a pair of nodes \( i \) and \( j \) depends only on their community assignment, i.e., \( p_{ij} = \omega_{\pi_i,\pi_j} \), where \( \pi_i \) is the community assignment for node \( i \) and \( \omega_{\pi_i,\pi_j} \) is the probability that a link exists between members of groups \( \pi_i \) and \( \pi_j \). This general definition of community structure is quite flexible, and allows for both assortative and disassortative community structure, as well as arbitrary mixtures thereof.

### Testing for a relationship between metadata and structure

Our first method, called the blockmodel entropy significance test (BESTest), is a statistical test to determine if the metadata partition is relevant to the network structure [case (i)], i.e., if it provides a good description of the network under a given model. We quantify relevance using the entropy, which is a measure of how many bits of information it takes to record the network given both the network model and its parameters. The lower the entropy under this model, the better the metadata describe the network, while a higher entropy implies that the metadata and the patterns of edges in the network are relatively uncorrelated. We derive and discuss the BESTest using five different models in Supplementary Text B. Here we describe a particularly straightforward version of this test using the SBM.

The BESTest works by first dividing a network’s nodes according to the labels of the metadata and then computing the entropy of the SBM that best describes the partitioned nodes. This entropy is then compared to a standard \( \pi \)-value, which can be estimated empirically by computing \( H(G; \pi) \) for a large number of randomly permuted metadata vectors \( \pi \). Smaller \( \pi \)-values indicate that the metadata provide a better description of the network, making it relatively less plausible that a random permutation of the metadata values could describe the network as well as the observed metadata does. It is important to note that \( \pi \)-values measure statistical significance but not effect strength, meaning that a low \( \pi \)-value does not indicate a strong correlation between the metadata and the network structure. Recently, Bianconi et al. [40] proposed a related entropy test for this task, based on a Normal approximation to the null distribution under the SBM. The blockmodel entropy significance test described here is a generalization of Bianconi et al.’s test that is both more flexible, as it can be used with any number of null models, and more accurate, as the true null distribution is substantially non-Normal (Fig. S5).

The blockmodel entropy significance test is, by construction, sensitive to even low correlations between metadata and network structure. To quantify the sensitivity of this \( \pi \)-value, we first apply it to synthetic networks with known community structure (see Supplementary Text B for a complete description of synthetic network generation). For these networks, our ability to detect relevant metadata is determined jointly by the strength of the planted communities and the correlation between metadata and communities. Figure 2 shows that for networks with strong community structure we can reliably detect relevant metadata even for relatively low levels of correlation with the planted structure. In fact, our method can still identify relevant metadata when the community structure is sufficiently weak that communities are provably undetectable by any community detection algorithm that relies only on the network [8]. Statistical significance requires an increasing level of correlation with the underlying structure as community strength decreases; if there is no structure in the network (\( \epsilon = 1 \)) then any metadata partition will be correctly identified as irrelevant. Note that a low \( \pi \)-value does not mean that the metadata provide the best description of the network, nor does it imply that we should be able to recover the metadata partition using community detection.

We now apply the blockmodel entropy significance test to a social network of interactions within a law firm, and to biological networks representing similarities among genes in the human malaria parasite \( P. falciparum \) (see Supplementary Text D). The first set, the Lazega Lawyers networks, comprises three networks on the same set of nodes and five metadata attributes. The
FIG. 2. Expected \( p \)-value estimates of the blockmodel entropy significance test as the correlation \( \ell \) between metadata and planted communities increases (each metadata label correctly reflects the planted community with probability \((1 + \ell) / 2\); see Supplementary Text B). Each curve represents networks with a fixed community strength \( \epsilon = \omega_{rr} / \omega_{rr} \). Solid lines indicate strong community structure in the so-called detectable regime \((\epsilon < \lambda)\), while dashed lines represent weak undetectable communities \((\epsilon > \lambda)\) [8]. Three block density diagrams visually depict \( \epsilon \) values.

### TABLE I. BESTest \( p \)-values for Lazega Lawyers

| Network      | Metadata Attribute |
|--------------|--------------------|
|              | Status Gender Office Practice Law School |
| Friendship   | \(< 10^{-6}\) 0.034 \(< 10^{-6}\) 0.033 0.134 |
| Cowork       | \(< 10^{-3}\) 0.094 \(< 10^{-6}\) \(< 10^{-6}\) 0.922 |
| Advice       | \(< 10^{-6}\) 0.010 \(< 10^{-6}\) \(< 10^{-6}\) 0.205 |

### TABLE II. BESTest \( p \)-values for Malaria \( var \) genes

| \( var \) Gene | Network Number |
|---------------|---------------|
| Genome        | 0.566 0.064 0.536 0.588 0.382 0.275 0.020 0.464 0.115 |

multiple combinations of edge and metadata types that yield highly significant \( p \)-values (Table I; see Table S3 for results using additional models of community structure) indicate that each set of metadata provides non-trivial information about the structure of multiple networks, and vice versa, implying that all metadata sets are relevant to the edge formation process, so none should be individually treated as ground truth.

The second set, the malaria \( var \) gene networks, comprises nine networks on the same set of nodes and three sets of metadata. For each network, we find a non-significant \( p \)-value when the metadata denote the parasite genome-of-origin (Table II; see Table S4 for results using additional models of community structure and additional metadata). In contrast to the Lazega Lawyers network, these genome metadata are statistically irrelevant for explaining the observed patterns of gene recombinations. This finding substantially strengthens the conclusions of Ref. [41] which used a less sensitive test based on label assortativity. Some metadata for these networks do correlate, however (see Supplementary Text B).

### Diagnosing the structural aspects captured by metadata and communities

Our second method provides a direct means to diagnose whether some metadata and a network’s detected communities differ because they reveal different aspects of the network’s structure [case (ii)]. We accomplish this by extending the SBM to probe the local structure around and between the metadata partition and the detected structural communities. This extended model, which we call the neoSBM, performs community detection under a constraint in which each node is assigned one of two states, which we call blue or red, and a parameter \( q \) that governs the number of nodes in each state. If a node is blue, its community is fixed as its metadata label, while if it is red, its community is free to be chosen by the model. We choose \( q \) automatically within the inference step of the model by imposing a likelihood penalty in the form of a Bernoulli prior with parameter \( \theta \), which controls for the additional freedom that comes from varying \( q \). The neoSBM’s log likelihood is \( L_{\text{neoSBM}} = L_{\text{SBM}} + q \psi(\theta) \), where \( \psi(\theta) \) may be interpreted as the cost of freeing a node from its metadata label (see Supplementary Text A for exact formulation).

By varying the cost of freeing a node, we can use the neoSBM to produce a graphical diagnostic of the interior of the space between the metadata partition and the inferred community partition. In this way, the neoSBM can shed light on how the metadata and inferred community partitions are related, beyond direct comparison of the partitions via standard techniques such as normalized mutual information or the Rand index. As the cost of freeing nodes is reduced, the neoSBM creates a path through the space of partitions from metadata to the optimal community partition and, as it does so, we monitor the improvement of the partition by the increase in SBM log likelihood. A steady increase indicates that the neoSBM is incrementally refining the metadata partition until it matches the globally optimal SBM communities. This behavior implies that the metadata and community partitions represent related aspects of the network structure. On the other hand, if the SBM likelihood remains constant for a substantial range of \( \theta \), followed by a sharp increase or jump, it indicates that the neoSBM has moved from one local optimum to another. Multiple plateaus and jumps indicate that several local optima have been traversed, revealing that the partitions are capturing different aspects of the network’s structure.

To demonstrate the usage of the neoSBM, we examine the path between partitions for a synthetic network with four locally optimal partitions which correspond to the four distinct peaks in the surface plot (Fig. 3A; see Supplementary Text A for a complete description of synthetic network generation). We take the partition of the lowest of these peaks as metadata and use the neoSBM
FIG. 3. The neoSBM on synthetic data. (A) The stochastic block model (SBM) likelihood surface shows four distinct peaks corresponding to a sequence of locally optimal partitions. (B) Block density diagrams depict community structure for locally optimal partitions, where darker color indicates higher probability of interaction. (C) The neoSBM, with partition 1 as the metadata partition, interpolates between partition 1 and the globally optimal SBM partition 4. The number of free nodes $q$ and SBM log likelihood as a function of $\theta$ show three discontinuous jumps as the neoSBM traverses each of the locally optimal partitions (1–4).

to generate a path to the globally optimal partition by varying the $\theta$ parameter of the neoSBM from 0 to 1. The corresponding changes in the SBM log likelihood and the number of free nodes show three discontinuous jumps (Fig. 3C), one for each time the model encounters a new locally optimal partition.

Examining the partitions along the neoSBM’s path can provide direct insights into the relationship between metadata and network structure. Figure 3B shows the structure at each of the four traversed optima as block-wise interaction matrices $\omega$. Each partition has a different type of large-scale structure, from core-periphery to assortative patterns. In this way, when metadata do not closely match inferred communities, the neoSBM can shed light on whether and how the metadata capture similar or different aspects of network structure.

We now present an application of the neoSBM to the Lazega Lawyers data analyzed in the previous section. When initialized with the law school and office location metadata, the neoSBM produces distinct patterns of relaxation to the global optimum (Fig. 4A,C), approaching it from opposite sides of the peak in the likelihood surface. Starting at the law school metadata, the model traverses the space of partitions to the global SBM-optimal partition without encountering any local optima. In contrast, the path from the office metadata crosses one local optimum (Fig. 4A,B), which indicates that the law school metadata are more closely associated with the large-scale organization of the network than are the office metadata. Both sets of metadata labels are relevant, however, as we determined in the previous section using the BESTest. Results for other real-world networks are included in Supplementary Text A, including generalizations of the neoSBM to other community detection methods.

DISCUSSION

Treating node metadata as ground truth communities for real-world networks is commonly justified via an erroneous belief that the purpose of community detection is to recover groups that match metadata labels [11, 13, 31, 43]. Consequently, metadata recovery is often used to measure community detection performance [44] and metadata are often referred to as ground truth [21, 45]. However, the organization of real networks typically correlates with multiple sets of metadata, both observed and unobserved. Thus, labeling any particular set to be “ground truth” is an arbitrary and generally unjustified decision. Furthermore, when a community detection algorithm fails to identify communities that match known metadata, poor algorithm performance is indistinguishable from three alternative possibilities: (i) the metadata are irrelevant to the network structure, (ii) the metadata and communities capture different aspects of the network structure, or (iii) the network lacks group structure. Here, we have introduced two new statistical tools to directly investigate cases (i) and (ii), while (iii) remains well addressed by work from other authors [8, 23–29]. We have also articulated multiple mathematical arguments which conclude that treating metadata as ground truth in community detection induces both theoretical and practical problems. However, we have also shown that metadata remain useful and that a careful exploration of the relationship between node metadata and community structure can yield new insights into the network’s underlying generating process.

By searching only for communities that are highly correlated with metadata, we risk focusing only on positive correlations while overlooking other scientifically relevant organizational patterns. In some cases, disagreements between metadata labels and community detection re-
There is no universally accepted definition of community structure, nor should there be. Networks represent a wide variety of complex systems, from biological to social to artificial systems, and their large-scale structure may be generated by fundamentally different processes. Good community detection methods like the SBM can be powerful exploratory tools, able to uncover a wide variety of such patterns in real networks. But, as we have shown here, there is no free lunch in community detection. Instead, algorithmic biases that improve performance on one class of networks must reduce performance on others. This is a natural trade off, and suggests that good community detection algorithms come in two flavors: general algorithms that perform fairly well on a wide variety of tasks and inputs, and specialized algorithms that perform very well on a more narrow set of tasks, outperforming any general algorithm, but which perform more poorly when applied outside their preferred domain [an insight foreshadowed in past work [49]]. In fact, in some cases it may be advantageous to use a set of carefully chosen metadata and a narrow set of corresponding networks to train specialized algorithms. Historically, most work on community detection algorithms has focused on developing general approaches. A deeper consideration of how the outputs of community detection algorithms will be subsequently used, e.g., in testing scientific hypotheses, predicting missing information, or simply coarse-graining the network, may shed new light on how to design better algorithms for those specific tasks. An important direction of future work is thus to better understand both these trade offs and the errors that can occur in domain-agnostic applications [50, 51].

A complementary approach is to incorporate the metadata into inference process itself, which can help guide a method toward producing more useful results. The neoSBM introduced here is one such method. Others include methods that use metadata as a prior for community assignment [21] and identify relevant communities to predict missing network or metadata information [33, 34, 52]. However, there is potential to go...
further than these domain-agnostic methods can take us. Tools that incorporate correct domain-specific knowledge about the systems they represent will provide the best lens for revealing patterns beyond what is already known and ultimately lead to important scientific breakthroughs. By rigorously probing these relationships we can move past the false notion of metadata as ground truth, and instead uncover the particular organizing principles underlying real world networks and their metadata.

ACKNOWLEDGEMENTS

The authors thank Cris Moore, Tiago Peixoto, Michael Schaub, David Wolpert, and Johan Ugander for insightful conversations. This work was supported by IAP (Belgian Scientific Policy Office) and ARC (Federation Wallonia-Brussels) (LP), the SFI Omidyar Fellowship (DBL), and NSF Grant IIS-1452718 (AC).

DATA AND CODE

Computer code implementing the analysis methods described in this paper and other information can be found online at https://piratepeel.github.io/code.html and http://danlarremore.com/metadata.

[1] Lada A Adamic and Natalie Glance, “The political blogosphere and the 2004 US election: divided they blog,” in Proc. of the 3rd Int. Workshop on Link Discovery (ACM, 2005) pp. 36–43.
[2] Santo Fortunato, “Community detection in graphs,” Physics Reports, 486, 75–174 (2010).
[3] Petter Holme, Mikael Huss, and Hawoong Jeong, “Subnetwork hierarchies of biochemical pathways,” Bioinformatics, 19, 532–538 (2003).
[4] Roger Guimera and Luís A Nunes Amaral, “Functional cartography of complex metabolic networks,” Nature, 433, 895–900 (2005).
[5] Corinna Cortes, Daryl Pregibon, and Chris Volinsky, “Communities of interest,” in Advances in Intelligent Data Analysis, Lecture Notes in Computer Science, Vol. 2189, edited by Frank Hofmann, David Hand, Niall Adams, Douglas Fisher, and Gabriela Guimaraes (Springer Berlin / Heidelberg, 2001) pp. 105–114.
[6] Leanne S Haggerty, Pierre-Alain Jachiet, William P Hanage, David A Fitzpatrick, Philippe Lopez, Mary J OConnell, Davide Pisani, Mark Wilkinson, Eric Bapteste, and James O McNerney, “A pluralistic account of homology: adapting the models to the data,” Mol. Biol. Evol., 31, 501–516 (2014).
[7] Paul Erdős and Alfréd Rényi, “On random graphs,” Publ. Math. Debrecen, 6, 290–297 (1959).
[8] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová, “Inference and phase transitions in the detection of modules in sparse networks,” Phys. Rev. Lett., 107, 065701 (2011).
[9] Michelle Girvan and Mark EJ Newman, “Community structure in social and biological networks,” Proc. Natl. Acad. Sci. USA, 99, 7821–7826 (2002).
[10] T S Evans, “Clique graphs and overlapping communities,” J. Stat. Mech.: Theory Exp., 2010, P12037 (2010).
[11] Darko Hric, Richard K Darst, and Santo Fortunato, “Community detection in networks: Structural communities versus ground truth.” Phys. Rev. E, 90, 062805 (2014).
[12] Jure Leskovec, Kevin J Lang, and Michael Mahoney, “Empirical comparison of algorithms for network community detection,” in Proc. of the 19th Int. World Wide Web Conf. (ACM, 2010) pp. 631–640.
[13] Jaewon Yang and Jure Leskovec, “Community-affiliation graph model for overlapping network community detection,” in Proc. of the 13th Int. Conf. on Data Mining (IEEE, 2012) pp. 1170–1175.
[14] Wayne W Zachary, “An information flow model for conflict and fission in small groups,” J. Anthropol. Res., 452–473 (1977).
[15] Paul W Holland, Kathryn Blackmon Laskey, and Samuel Leinhardt, “Stochastic blockmodels: First steps,” Social networks, 5, 109–137 (1983).
[16] Krzysztof Nowicki and Tom A B Snijders, “Estimation and prediction for stochastic blockstructures,” J. American Statistical Association, 96, 1077–1087 (2001).
[17] Jianbo Shi and Jitendra Malik, “Normalized cuts and image segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 22, 888–905 (2000).
[18] Xue-Qi Cheng and Hua-Wei Shen, “Uncovering the community structure associated with the diffusion dynamics on networks,” J. Stat. Mech.: Theory Exp., 2010, P04024 (2010).
[19] Florent Krzakala, Cristopher Moore, Elchanan Mossel, Joe Neeman, Allan Sly, Lenka Zdeborov, and Pan Zhang, “Spectral redemption in clustering sparse networks,” Proc. Natl. Acad. Sci. USA, 110, 20935–20940 (2013).
[20] Brian Karrer and Mark EJ Newman, “Stochastic blockmodels and community structure in networks,” Phys. Rev. E, 83, 016107 (2011).
[21] Mark EJ Newman and Aaron Clauset, “Structure and inference in annotated networks,” Nature Communications, 7, 11863 (2016).
[22] Benjamin H Good, Yves-Alexandre de Montjoye, and Aaron Clauset, “Performance of modularity maximization in practical contexts,” Phys. Rev. E, 81, 046106 (2010).
[23] Charles Bordenave, Marc Lelarge, and Laurent Massoulié, “Non-backtracking spectrum of random graphs: community detection and non-regular ramanujan graphs,” in 56th Annual Symposium on Foundations of Computer Science (IEEE, 2015) pp. 1347–1357.
[24] Amir Ghasemian, Pan Zhang, Aaron Clauset, Cristopher Moore, and Leto Peel, “Detectability thresholds and optimal algorithms for community structure in dynamic networks,” Phys. Rev. X, 6, 031005 (2016).
[25] Laurent Massoulié, “Community detection thresholds and the weak ramanujan property,” in Proc. of the 46th Annual ACM Symposium on Theory of Computing (ACM, 2014) pp. 694–703.
[26] Elchanan Mossel, Joe Neeman, and Allan Sly, “Belief propagation, robust reconstruction and optimal recovery of block models,” in Proc. of the 57th Conf. on Learning Theory, Vol. 35 (2014) pp. 356–370.
[27] Elchanan Mossel, Joe Neeman, and Allan Sly, “Reconstruction and estimation in the planted partition model,” Probability Theory and Related Fields, 162, 431–461 (2015).
[28] Roger Guimera, Marta Sales-Pardo, and Luís A Nunes Amaral, “Modularity from fluctuations in random graphs and complex networks,” Phys. Rev. E, 70, 025101 (2004).
[29] Dane Taylor, Marysia Zbigniew, and Cristopher Moore, “Detectability of small communities in multilayer and temporal networks: Eigenvector localization, layer aggregation, and time series discretization,” arXiv preprint arXiv:1609.04376 (2016).
[30] Andrea Lancichinetti and Santo Fortunato, “Community detection algorithms: a comparative analysis,” Phys. Rev. E, 80, 056117 (2009).
[31] Jaewon Yang and Jure Leskovec, “Defining and evaluating network communities based on ground-truth,” Knowledge and Information Systems, 42, 181–213 (2015).
[32] David H Wolpert, “The lack of a priori distinctions between learning algorithms,” Neural computation, 8, 1341–1390 (1996).
[33] Leto Peel, “Topological feature based classification,” in Proc. of the 14th Int. Conf. on Information Fusion (IEEE, 2011) pp. 1–8.
[34] Leto Peel, “Supervised blockmodeling,” in ECML/PKDD Workshop on Collective Learning and Inference on Structured Data (CoLISD) (2012) arXiv 1209.5561.
[35] Bailey K Fosdick and Peter D Hoff, “Testing and modeling dependencies between a network and nodal attributes,” J. Am. Stat. Assoc., 110, 1047–1056 (2015).
[36] Edo M Airoldi, David M Blei, Stephen E Fienberg, and Eric P Xing, “Mixed membership stochastic blockmodels,” in Advances in Neural Information Processing Systems (2009) pp. 33–40.
[37] Brian Ball, Brian Karrer, and MEJ Newman, “Efficient and principled method for detecting communities in networks,” Phys. Rev. E, 84, 036103 (2011).
[38] Daniel B Larremore, Aaron Clauset, and Abigail Z. Jacobs, “Efficiently inferring community structure in bipartite networks,” Phys. Rev. E, 90, 012805 (2014).
[39] Tiago P Peixoto, “Hierarchical block structures and high-resolution model selection in large networks,” Phys. Rev. X, 4, 011047 (2014).
[40] Cinzia Bianconi, Paolo Pin, and Matteo Marsili, “Assessing the relevance of node features for network structure,” Proc. Natl. Acad. Sci. USA, 106, 11433–11438 (2009).
[41] Daniel B Larremore, Aaron Clauset, and Caroline O Buckee, “A network approach to analyzing highly recombinant malaria parasite genes,” PLoS Comput Biol, 9, e1003268 (2013).
[42] Emmanuel Lazega, The collegial phenomenon: The social mechanisms of cooperation among peers in a corporate law partnership (Oxford University Press on Demand, 2001).
[43] Yong-Yeol Ahn, James P Bagrow, and Sune Lehmann, “Link communities reveal multiscale complexity in networks,” Nature, 466, 761–764 (2010).
[44] Sucheta Soundarajan and John Hopcroft, “Using community information to improve the precision of link prediction methods,” in Proc. of the 21st Int. World Wide Web Conf. (ACM, 2012) pp. 607–608.
[45] Tammoj Chakraborty, Sandipan Sikdar, Vihar Tammana, Niloy Ganguly, and Animesh Mukherjee, “Computer science fields as ground-truth communities: Their impact, rise and fall,” in Proc. of the Int. Conf. on Advances in Social Networks Analysis and Mining (IEEE/ACM, 2013) pp. 426–433.
[46] Ulrike von Luxburg, Bob Williamson, Isabelle Guyon, et al., “Clustering: Science or art?” J. of Machine Learning Research, 27, 65–80 (2012).
[47] Jon Kleinberg, “An impossibility theorem for clustering,” Advances in neural information processing systems, 453–470 (2003).
[48] Arnaud Browet, Julien M Hendrickx, and Alain Sarlette, “Incompatibility boundaries for properties of community partitions,” arXiv preprint arXiv:1603.00621 (2016).
[49] Aaron Clauset, Cristopher Moore, and Mark EJ Newman, “Hierarchical structure and the prediction of missing links in networks,” Nature, 453, 98–101 (2008).
[50] Leto Peel, “Estimating network parameters for selecting community detection algorithms,” J. of Advances in Information Fusion, 6, 119–130 (2011).
[51] Zhao Yang, René Algesheimer, and Claudio J Tesseone, “A comparative analysis of community detection algorithms on artificial networks,” Scientific Reports, 6, 30750 (2016).
[52] Darko Hric, Tiago P Peixoto, and Santo Fortunato, “Network structure, metadata and the prediction of missing nodes,” arXiv preprint arXiv:1604.00255 (2016).
[53] Tiago P Peixoto, “Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models,” Phys. Rev. E, 89, 012804 (2014).
[54] Tiago P Peixoto, “Entropy of stochastic blockmodel ensembles,” Phys. Rev. E, 85, 056122 (2012).
[55] Juuso Parkkinen, Janne Sinkkonen, Adam Gyenge, and Samuel Kaski, “A block model suitable for sparse graphs,” in Proceedings of the 7th International Workshop on Mining and Learning with Graphs (MLG 2009), Leuven, Vol. 5 (2009).
[56] Jacek Cichoń and Zbigniew Gołębiewski, “On bernoulli sums and bernstein polynomials,” in 23rd International Meeting on Probabilistic, Combinatorial, and Asymptotic Methods in the Analysis of Algorithms (AofA’12) (Discrete Mathematics and Theoretical Computer Science, 2012) pp. 179–190.
[57] Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Görke, Martin Heloer, Zoran Nikoloski, and Dorothea Wagner, “Maximizing modularity is hard,” arXiv preprint physics/0608255 (2006).
[58] Nguyen Xuan Vinh, Julien Epps, and James Bailey, “Information theoretic measures for clusterings comparison: is a correction for chance necessary?” in Proc. 26th Int. Conf. on Machine Learning (ACM, 2009) pp. 1073–1080.
[59] Ingwer Borg and Patrick J F Groenen, *Modern multidimensional scaling: Theory and applications* (Springer Science & Business Media, 2005).

[60] Marina Meilă, “Comparing clusterings by the variation of information,” in *Learning theory and kernel machines* (Springer, 2003) pp. 173–187.

[61] The Wachowskis, “The Matrix.” (1999).
Appendix A: The neoSBM

Morpheus: Unfortunately, no one can be told what the Matrix is. You have to see it for yourself... This is your last chance. After this, there is no turning back. You take the blue pill, the story ends, you wake up in your bed and believe whatever you want to believe. You take the red pill, you stay in Wonderland, and I show you how deep the rabbit hole goes. Remember: all I’m offering is the truth. Nothing more. [61]

This Supplementary Text is divided into four subsections providing additional details on the neoSBM.

- Subsection I describes the neoSBM (I.a) and the inference methods used in this paper (I.b).
- Subsection II describes the generation of the synthetic network used in the main text, Fig. 3.
- Subsection III describes how the neoSBM can be extended to other models including the degree corrected neoSBM.
- Subsection IV provides additional examples of results of the neoSBM applied to the Lazega Lawyers networks (IV.a) and the Malaria networks (IV.b).

For convenience, we provide a reference table of notation used in derivations in this Supplementary Text.

| Variable | Definition |
|----------|------------|
| $\mathcal{G}$ | a network, $\mathcal{G} = (V, E)$ |
| $N$ | the number of nodes $|V|$ |
| $A_{ij}$ | the number of edges between nodes $i$ and $j$, $A_{ij} \in \{0, 1\}$ |
| $k_i$ | the degree of node $i$. |
| $\omega_{rs}$ | the probability of an edge between nodes in groups $r$ and $s$ |
| $\pi$ | a partition of nodes into groups |
| $M$ | a set of metadata labels |
| $C$ | an inferred optimal community assignment |
| $z$ | neo-state indicator variable, $z_i \in \{b, r\}$ |
| $\theta$ | Bernoulli prior probability parameter |
| $L_X$ | log likelihood $L$ of model $X$ |
| $q$ | the number of free nodes, $q = \sum z_i$ |
| $\delta_{a,b}$ | the Kronecker delta: $\delta_{a,b} = 1$ for $a = b$; $\delta_{a,b} = 0$ for $a \neq b$ |

1. neoSBM model description and inference

   a. Model description

   The neoSBM extends the SBM, allowing metadata to influence the inferred partitions by controlling the number of nodes that are assigned to groups according to their metadata labels. The task of the neoSBM is to perform community detection under a constraint in which each node is assigned a latent state variable $z_i$, which can take one of two states, which we call blue or red. If a node is blue $z_i = b$, its community is fixed as its metadata label $\pi_i = M_i$. However, if it is red $z_i = r$, its community is free to be chosen by the model. We adjust the number of free nodes $q$ by varying the Bernoulli prior probability $\theta$ that a node will be free (red state). We can then write down the likelihood $L_{\text{neo}}$ of a network $\mathcal{G}$ given a community assignment $\pi$ under the neoSBM as:

   $$ L_{\text{neo}}(\mathcal{G}; \pi, z) = \prod_{ij} A_{ij} \omega_{\pi_i \pi_j} (1 - \omega_{\pi_i \pi_j}) \prod_i \theta^{\delta_{z_i, r}} (1 - \theta)^{\delta_{z_i, b}}. $$  (A1)

   The first product in Eq. (A1) corresponds to the standard SBM likelihood $L_{\text{sbm}}$, while the second product corresponds to the probability of the states $P(z = r|\theta)$ and acts as a penalty function to control the number of free nodes. While it is possible to find communities by optimizing Eq. (A1) directly, instead we work with the more practical log likelihood,

   $$ L_{\text{neo}}(\mathcal{G}; \pi, z) = \sum_{ij} A_{ij} \log \omega_{\pi_i \pi_j} + (1 - A_{ij}) \log(1 - \omega_{\pi_i \pi_j}) $$

   $$ + \sum_i \delta_{z_i, r} \log \theta + \delta_{z_i, b} \log(1 - \theta), $$  (A2)

   since maximizing Eq. (A1) is equivalent to maximizing Eq. (A2). We can then rearrange the second sum $\log P(z = r|\theta)$, to give:

   $$ \log P(z = r|\theta) = \sum_i \delta_{z_i, r} \left( \log \frac{\theta}{1 - \theta} \right) + N \log(1 - \theta) $$

   $$ = q \psi(\theta) + N \log(1 - \theta), $$  (A3)

   dropping the constant term, we can rewrite the neoSBM log likelihood in terms of the SBM log likelihood and a function of the number of free nodes $q$,

   $$ L_{\text{neo}}(\mathcal{G}; \pi, z) = L_{\text{sbm}}(\mathcal{G}; \pi) + q \psi(\theta). $$  (A4)

   We emphasize that in the equation above, $\theta$ is a fixed parameter, and $q$ is selected automatically during inference as part of the likelihood maximization. Optimization of $L_{\text{sbm}}$ yields the SBM optimal communities $C$,

   $$ C = \arg \max_{\pi} L_{\text{sbm}}(\mathcal{G}; \pi), $$  (A5)

   and so the SBM likelihood given the metadata partition $M$ will always be less than or equal to the likelihood of the inferred partition $C$. That is $L_{\text{sbm}}(\mathcal{G}; M) \leq L_{\text{sbm}}(\mathcal{G}; C)$, where the inequality is saturated if and only if the metadata is equal to the optimal SBM partition. So the minimum number of free nodes $\hat{q}$ required to maximize the SBM likelihood is

   $$ \hat{q} = \sum_i 1 - \delta_{M_i, C_i}. $$  (A6)
for which the label permutations of $M$ and $C$ are maximally aligned. Whenever $q > \hat{q}$ there will be no further improvement in $L_{\text{sbm}}$. To interpolate between $M$ and $C$ we vary the prior probability of each node to take the red state $P(z = r | \theta)$. For values of $\theta < 0.5$ we can interpret the log probability, or $\psi(\theta)$, as the cost of freeing a node because the log likelihood $L_{\text{neo}}$ will incur a penalty for setting each $z_i = r$. Maximizing $L_{\text{neo}}$ is therefore a trade-off between freeing nodes to maximize $L_{\text{sbm}}$ and fixing nodes to metadata labels to maximize log $P(z | \theta)$. When the SBM likelihood of both partitions is equal (i.e., $M = C$) then $L_{\text{neo}}(G; \pi, z)$ will be maximized when $q = 0$ unless $\theta \geq 0.5$. However, when $L_{\text{sbm}}(G; M) < L_{\text{sbm}}(G; C)$, $q$ can be greater than 0 if the resulting partition $\pi$ provides a sufficient increase in log likelihood. Specifically, if

$$L_{\text{sbm}}(G; \pi) - L_{\text{sbm}}(G; M) > q \psi(\theta),$$

then it indicates that the cost of freeing $q$ nodes is outweighed by its contribution to improving the likelihood.

Here we have discussed the extension of the SBM to the neoSBM, but this extension can be easily generalized to any probabilistic generative network model that specifies the likelihood of a graph given a partition of the network. We present one such generalization, the degree-corrected neoSBM, in subsection III of this Supplementary Text.

b. Inference

Inference of the parameters of the neoSBM was performed using a Markov chain Monte Carlo (MCMC) approach. The community labels of the free nodes were inferred in the same way as the standard SBM [53]. However, to infer the values of $z_i$ that determined whether or not each node was free, we used a uniform Bernoulli (i.e., a fair coin) as a proposal distribution. Since this distribution is symmetric we can simply accept each proposal with probability $a$:

$$a = \min \left\{ \Delta L_{\text{neo}}, 1 \right\}. \quad (A8)$$

To avoid getting trapped in local optima of the likelihood, we initialize the neoSBM with the labels set to the inferred SBM partition, $\pi = C$, and all nodes initialized to be free, $z_i = r$ for all $i$.

2. Extensions

The neoSBM can easily be extended to any probabilistic model for which we identify communities by maximizing the model likelihood. As an example, consider the degree-corrected SBM, which allows for nodes with heterogenous degrees to belong to the same community (see Supplementary Text B for more details). We can create a degree-corrected neoSBM in much the same way as we created the neoSBM, by penalizing the likelihood according to the number of free nodes using a Bernoulli prior. This treatment gives the log likelihood:

$$L_{\text{dcsbm}}(G; \pi, z) = L_{\text{dcsbm}}(G; \pi) + q \psi(\theta), \quad (A9)$$

where $q \psi(\theta) = q \log P(z = r | \theta) + N \log(1 - \theta)$ as before. We present results from this model in subsection IV of this Supplementary Text.

We can also easily extend the neoSBM to other, non-probabilistic, community detection methods provided they explicitly optimize a global objective function. Then we can similarly create a penalized version of this objective function. That is, for some community detection model $X$, we can create a neo-objective function $U_{\text{neo}X}$

$$U_{\text{neo}X} = U_X + q \psi(\theta), \quad (A10)$$

where $\psi(\theta)$ could either represent the Bernoulli prior as before or any other cost function, e.g., $\psi(\theta) = \theta$, for $\theta \leq 0$.

3. IV. Results on real-world networks

In order to further demonstrate the neoSBM and the neoDCSBM described above, we present and discuss the application of the neoSBM to malaria var gene networks and the application of the neoDCSBM to the Karate Club network. Full details about these data sets are presented in Supplementary Text D.

a. neoDCSBM and the Karate Club network

The likelihood surface for both models contains two local optima that correspond two the same two partitions, each being globally optimal for one of the models. Using the faction each member joined after the club split as metadata Fig. S1 compares the output from the neoSBM and the neoDCSBM. Both models initially change just a single node to reach a local optimum. For the DCSBM this is the global optimum and so we see no further change. However, for the neoSBM this is not the global optimum (see Fig. 1) and so once $\theta$ is large enough we see a discontinuous jump as it switches to the globally optimal high-degree/low-degree partition.

b. neoSBM and the Malaria var gene networks

The metadata corresponding to upstream promoter sequence (UPS) are known to correlate with community structure in the malaria var gene networks, particularly at loci one and six [21, 41]. We provided the neoSBM with UPS metadata ($K = 4$) and investigated the path of partitions between the metadata partition and the globally optimal partitions for each of the two networks. Figures S2 (locus one) and S3 (locus six) show likelihood
Comparison of the neoSBM results for the same metadata on two different network layers reveals not only that the intermediate paths of locally optimal partitions differ but that the UPS metadata are more locally stable for the locus six network. This is indicated by the substantially larger value of $\theta$ at which the neoSBM switches from the metadata partition to the first intermediate local optimum. These transitions $1 \rightarrow 2$ involve different numbers of free nodes, however, indicating that the switch from optimum 1 to optimum 2 was accompanied by a much larger change in node mobility for the locus six network. Note that the neoSBM provides a more nuanced view of the relationship between UPS metadata and malaria layers one and six than the BESTest did, which found that UPS metadata were significantly correlated with the structures of both networks.

4. Synthetic network generation for the neoSBM

The test that demonstrated the function of the neoSBM on synthetic data, depicted in Fig. 3 of the main text, required networks with multiple local optima under the SBM: one corresponding to the inferred partition (global optimum) and at least one other to represent a relevant metadata partition. To create such a network, we divided vertices into $2K$ groups to create $K$ assortative communities, each of which was subdivided to contain a core and a periphery group. For $K = 4$, Figure S4 shows the 8-block interaction matrix used to create the synthetic networks. By subsequently varying the mean degree within each block, we obtained two uncorrelated partitions when $K = 4$, both of which are relevant to the network structure. Finally, we assigned as metadata the core-periphery structure containing one periphery group ($\{2, 4, 5, 7\}$ in Fig. S4) and three core groups ($\{1, 3\}, \{6\}, \{8\}$ in Fig. S4). The partition inferred by the SBM in the absence of the neoSBM’s likelihood penalty corresponds to the assortative group structure.
FIG. S2. Results of the neoSBM on the malaria var gene network at locus one ("malaria 1") using UPS metadata. (A) The SBM likelihood surface shows two peaks, one subtle 2 and one prominent 3, corresponding to a locally optimal partition near the metadata and the globally optimal partition, respectively. There is no peak at the metadata partition 1, however. (B) Block density diagrams depict community structure for metadata and locally optimal partitions, where darker color indicates higher probability of interaction. (C) The neoSBM, beginning from UPS metadata, interpolates between metadata 1 and the globally optimal SBM partition 3. The number of free nodes q and SBM log likelihood as a function of θ shows two discontinuous jumps as the neoSBM traverses from the metadata to the locally optimal partition (1 → 2) and then from that partition to the global optimum (2 → 3).

FIG. S3. Results of the neoSBM on the malaria var gene network at locus six ("malaria 6") using UPS metadata. (A) The SBM likelihood surface shows one prominent peak at the globally optimal partition. (B) Block density diagrams depict community structure for metadata and locally optimal partitions where darker color indicates higher probability of interaction. (C) The neoSBM, beginning from UPS metadata, interpolates between metadata 1 and the globally optimal SBM partition, traversing a local optimum during its path. The number of free nodes q and SBM log likelihood as a function of θ shows two discontinuous jumps as the neoSBM traverses from the metadata to the locally optimal partition (1 → 2), from that partition to another the global optimum (2 → 3).

Morpheus: Have you ever had a dream, Neo, that you seemed so sure it was real? But if were unable to wake up from that dream, how would you tell the difference between the dream world and the real world? [61]
FIG. S4. The block interaction matrix used to generate synthetic networks. The external colored rows and columns indicate the partition used as metadata (M) and the maximum likelihood partition under the SBM (C).
Appendix B: Blockmodel Entropy Significance Test

**Morpheus:** I’m trying to free your mind, Neo. But I can only show you the door. You’re the one that has to walk through it. [61]

This Supplementary Text is divided into six subsections providing additional details on the blockmodel entropy significance test.

- Subsection B 1 describes maximum likelihood parameter estimation for the SBM (I.a) and degree-corrected SBM (I.b).
- Subsection B 2 describes rapid computation of the entropy $H(G; \mathcal{M})$ for the Bernoulli SBM and Multinomial degree-corrected SBM (DCSBM).
- Subsection B 3 demonstrates the mathematical link between our formulation of the SBM entropy and the SBM log likelihood which has been derived elsewhere [20, 54].
- Subsection B 4 discusses the use of non-generative models like modularity.
- Subsection B 5 provides details on the generation of synthetic networks for the tests shown in Fig. 2.
- Subsection B 6 provides additional examples of results of the blockmodel entropy significance test using multiple different network data and metadata sets (see Supplementary Text D) as well as three additional generative network models beyond the SBM.

For convenience, we provide a reference table of notation used in derivations in this Supplementary Text.

**TABLE S2. Notation used in this Supplementary Text**

| Variable | Definition |
|----------|------------|
| $G$ | a network, $G = (V, E)$ |
| $N$ | the number of nodes $|V|$ |
| $\pi$ | a partition of nodes into groups |
| $K$ | the total number of groups |
| $\pi_i$ | the group assignment of node $i$ |
| $n_r$ | the number of nodes in group $r$ |
| $m_{rs}$ | the number of edges between groups $r$ and $s$ |
| $\kappa_r$ | the total degrees of group $r$, $\kappa_r = \sum_s m_{rs}$ |
| $k_i$ | the degree of node $i$ |
| $H_X(G|\pi)$ | entropy $H$ of model $X$ estimated for graph $G$ using partition $\pi$ |
| $\hat{a}$ | maximum likelihood estimate of model parameter $a$ |
| $p_{ij}$ | the probability that an edge exists between nodes $i$ and $j$ |

**FIG. S5. Distributions of permuted partition entropies are negatively skewed.** Probability density functions (top) and cumulative distribution functions (bottom) are shown for $10^4$ independent permutations of the metadata partition. Note that these permutation entropies are far from normal; a normal distribution with equivalent mean $\mu$ and variance $\sigma^2$ is shown in blue for contrast.

1. **Estimation of SBM parameters**

   a. **Bernoulli SBM parameters**

   Let the $N$ nodes of a network $G$ be partitioned into $K$ groups, with the group assignment of node $i$ given by $\pi_i$. In the SBM, the probability of a link existing between any two nodes $i$ and $j$ depends only on the group assignments $\pi_i$ and $\pi_j$. This means that the entire model can be parameterized by a $K \times K$ matrix of block-to-block edge probabilities, $\omega$. Accordingly, let $\omega$ be a matrix such that $p_{ij} = \omega_{\pi_i\pi_j}$ is the probability of a link existing between $i$ and $j$. Letting the number of nodes in group $r$ be $n_r$, then between two groups $r$ and $s$ there are $n_r n_s$ possible links, each of which has the same probability of existence, $\omega_{rs}$. This implies that the existence of the $n_r n_s$ edges between groups $r$ and $s$ will be determined by $n_r n_s$ independent Bernoulli trials, each with parameter $\omega_{rs}$.

   We must now estimate the value of $\omega_{rs}$ for a network $G$ whose nodes have been divided according to their assignments in partition $\pi$. Of course, any $\omega$ whose entries are positive will have some non-zero probability of having generated the observed links in $G$. However, here we choose the values of $\omega$ to be those that maximize the likelihood of observing $G$. Specifically, observe that of the $n_r n_s$ Bernoulli trials, there are $m_{rs}$ actual edges in the graph, i.e., $m_{rs}$ trial successes. Therefore, the maximum likelihood estimate of $\omega_{rs}$ is simply $\hat{\omega}_{rs} = m_{rs}/n_r n_s$. Thus, $\hat{p}_{ij} = \hat{\omega}_{\pi_i\pi_j}$.
b. Poisson degree-corrected SBM parameters

In the degree-corrected Poisson SBM [20], it is still assumed that each link exists independently of the others, with some specified probability given by a block connectivity matrix \( \omega \). However, this model differs in two key ways from the Bernoulli SBM. First, rather than each edge existing with probability \( p_{ij} \), Poisson SBMs state that the expected number of edges between nodes \( i \) and \( j \) is given by a parameter \( q_{ij} \), with the actual number of edges drawn from a Poisson distribution with identical mean. For very small values of \( q \), the probability of an edge existing is approximately \( q \), and thus if the graph is sufficiently sparse, Poisson SBMs behave similarly to Bernoulli SBMs, despite the fact that they could, in principle, generate multigraphs.

The second way in which this degree-corrected Poisson SBM differs from the Bernoulli SBM is that the parameters \( q_{ij} \) are no longer identical across the set of all \( i \) in group \( r \) and all \( j \) in group \( s \), as they are in the uncorrected SBM. Now, each node has a degree affinity \( \omega \) with some specified probability given by a block connectivity matrix, controlling the numbers of links between groups, similar in principle to \( \omega_{rs} \) above. The new parameters, \( \theta_{rs} \), properly chosen [20], can be used to specify the expected degree of each node.

As above, since we are given a network \( G \) and a fixed partition \( \pi \), we must estimate the entries of \( e \), as well as the values of \( \theta \). The parameters can again be chosen to maximize the likelihood of observing \( G \), which are derived in [20] but we do not derive here. First, \( e_{rs} = m_{rs} \), where \( m_{rs} = \sum_{ij} A_{ij} \delta_{r,i} \delta_{s,j} \), is the number of links between groups \( r \) and \( s \) (or twice the number of links if \( r = s \)). Then, \( \hat{\theta}_{rs} = k_i/\kappa_{rs} \), where \( \kappa_r = \sum_s m_{rs} \). Thus, \( q_{ij} = k_i k_j m_{\pi_i, \pi_j} / \kappa_r \kappa_{ij} \). We note that this maximum likelihood estimate is only valid in the regime that \( k_i k_j m_{\pi_i, \pi_j} \ll \kappa_r \kappa_{ij} \).

2. Rapidly computing entropy

a. Rapid Bernoulli SBM entropy

Under either a Bernoulli-type SBM, a link exists between nodes \( i \) and \( j \) with probability \( p_{ij} \), independently of all other links. This amounts to a Bernoulli trial or flip of a biased coin, and the entropy of this Bernoulli trial with parameter \( p_{ij} \) is simply

\[
h(p_{ij}) = -p_{ij} \log_2 p_{ij} - (1 - p_{ij}) \log_2 (1 - p_{ij}). \tag{B1}
\]

Hereafter, we will write simply log in place of \( \log_2 \). Because the Bernoulli trial on each link is conditionally independent of other links, the entropy of the network is the sum of all valid \( h(p_{ij}) \). For an undirected network this is

\[
H_{\text{SBM}}(G) = \sum_{i \leq j} h(p_{ij}) = \frac{1}{2} \left[ \sum_{ij} h(p_{ij}) + \sum_i h(p_{ii}) \right]. \tag{B2}
\]

Under the SBM, the probabilities within each block are identical so we may group them and change to an index over groups, rewriting Eq. (B2) as

\[
H_{\text{SBM}}(G) = \frac{1}{2} \left[ \sum_{rs} n_r n_s h(\omega_{rs}) + \sum_r n_r h(\omega_{rr}) \right]. \tag{B3}
\]

which may be simplified by plugging in the maximum likelihood estimate of \( \omega_{rs} \) and the definition of Bernoulli entropy \( h \) Eq. (B1), yielding

\[
H_{\text{SBM}}(G) = \cdots - \frac{1}{2} \sum_{rs} m_{rs} \log \omega_{rs} + (n_r n_s - m_{rs}) \log(1 - \omega_{rs}) + \mathcal{O}(n^{-1}). \tag{B4}
\]

where we have noted that the diagonal terms are \( \mathcal{O}(n^{-1}) \) whenever \( n_r = cn \) for some constant \( c \).

Eq. (B4) allows for a \( \mathcal{O}(K^2) \) computation, rather than \( \mathcal{O}(N^2) \) of Eq. (B2). For degree-corrected Bernoulli SBMs, entropies may be summed as in Eq. (B2), even though the rapid computation of Eq. (B4) will not be valid. However, in what follows, we show the connection between model entropy \( H \) and model log likelihood \( L \).

b. Rapid Multinomial DCSBM entropy

The degree-corrected SBM, introduced as a Poisson DCSBM by Karrer and Newman [20], can also be written in a “Multinomial” form in which each of the \( m \) edges is placed sequentially, according to the multinomial probabilities \( p_{ij} \) [55]. The values of \( p_{ij} \) are defined as

\[
p_{ij} = \theta_{ir} \omega_{rs} \theta_{js} = \frac{k_i k_j e_{rs}}{2 m e_r e_s} \tag{B5}
\]

where \( \theta_{ir} = k_i / e_r \) if node \( i \) is in group \( r \), and 0 otherwise, and \( \omega_{rs} = e_{rs} / 2m \). Note that by definition, \( \sum_{ij} p_{ij} = 1 \). When constructing a network, \( m \) edges are placed among the possible edge locations, with each one independently according to a categorical distribution with probabilities \( p_{ij} \) [55].

Since it is possible that multiple edges are formed between pairs of vertices, the entropy of this ensemble is not the entropy of \( m \) categorical distributions with parameters \( p \), but rather the entropy of the multinomial distribution with \( m \) draws and \( b \) “bins” with parameters \( p \). Note that if there is a nonzero possibility of an edge between each pair of vertices, then \( b = \binom{N}{2} \) [or \( b = N^2 \) in the directed case]. (There may be fewer than \( \binom{N}{2} \) bins in the undirected case if some values of \( e_{rs} \) are equal to
the parameters of the multinomial DSCBM, yielding

\[ H = \frac{1}{2} \log \left( \frac{2\pi me}{\pi} \right)^{b-1} \prod_{ij : p_{ij} \neq 0} p_{ij} \cdots \]

+ \frac{1}{12m} \left[ 3b - 2 - \sum_{ij:p_{ij} \neq 0} \frac{1}{p_{ij}} \right] + O \left( \frac{1}{m^2} \right). \quad (B6)

Thus, computing the entropy of this degree-corrected model \[55\] amounts to the rapid estimation of the parameters of the entropy from Eq. (B6).

3. Connecting entropy and log likelihood

The connection between model entropy \(H\) and model log likelihood \(\mathcal{L}\) enables the blockmodel entropy significance test to be expanded beyond the simple Bernoulli SBM to degree-corrected SBMs, Poisson SBMs, membership models, and other generative models with computable log likelihoods.

We begin from Eq. (B4) and use the Taylor series

\[ (1 - x) \ln(1 - x) = -x + \sum_{\ell=2}^{\infty} \frac{x^\ell}{\ell(\ell - 1)}, \quad (B7) \]

in which we substitute \(x = \hat{\omega}_{rs} = m_{rs}/n_r n_s\) to write Eq. (B4) to leading order as

\[ H_{SBM}(G) \approx -\frac{1}{2} \sum_{rs} \left[ m_{rs} \ln \left( \frac{m_{rs}}{n_r n_s} \right) - m_{rs} \ldots \right.\]

\[ + n_r n_s \sum_{\ell=2}^{\infty} \frac{1}{\ell(\ell - 1)} \left( \frac{m_{rs}}{n_r n_s} \right)^{\ell} \right]. \quad (B8) \]

Finally, we note that \(\frac{1}{2} \sum_{rs} m_{rs}\) is simply \(|E|\), the total number of links in the network and therefore

\[ H_{SBM}(G) \approx |E| - \frac{1}{2} \sum_{rs} \left[ m_{rs} \ln \left( \frac{m_{rs}}{n_r n_s} \right) \ldots \right.\]

\[ + n_r n_s \sum_{\ell=2}^{\infty} \frac{1}{\ell(\ell - 1)} \left( \frac{m_{rs}}{n_r n_s} \right)^{\ell} \right]. \quad (B9) \]

If all blocks of links are sparse, then \(m_{rs} \ll n_r n_s\) and the terms in the infinite sum decay rapidly, leading to the first order approximation

\[ H_{SBM}(G) \approx |E| - \frac{1}{2} \sum_{rs} m_{rs} \ln \left( \frac{m_{rs}}{n_r n_s} \right). \quad (B10) \]

Here we derived Eq. (B9) and Eq. (B10) by considering the conditionally independent entropies associated with every link of the network. However, the same equations can also be derived by calculating the size \(\Omega\) of the ensemble of networks associated with the same SBM, and then taking a logarithm, \(H = \log \Omega\). The log likelihood is the logarithm of the probability of observing an individual network realization from the ensemble, \(\mathcal{L} = \log P\), and under the assumption that each graph in the ensemble occurs with the same probability, \(P = 1/\Omega\). Therefore, the entropy \(H\) and the log likelihood \(\mathcal{L}\) are related simply by \(\mathcal{L} = -H [54]\).

The relationship between the “microcanonical” entropy and log likelihood allows for the Blockmodel Entropy Significance test to be expanded easily to any generative model for networks for which a likelihood is easily computed,

\[ p\text{-value} = \Pr \left[ \mathcal{L}(\tilde{G}; \pi) \geq \mathcal{L}(\tilde{G}; \mathcal{M}) \right]. \quad (B11) \]

The Bernoulli SBM entropy Eq. (B4) or its approximation for sparse networks Eq. (B10) are convenient because they are fast to compute—one need only to count links between groups, sizes of groups, and compute \(O(K^2)\) terms. By contrast, Eq. (B2), which is exact, requires \(O(N^2)\) computations. Depending on the assumptions involved, computing a log likelihood \(\mathcal{L}\) may be more or less rapid, or more or less exact.

In the additional tests in this Supplementary Text, we employ the Likelihood equations to apply the BESTTest using the Poisson SBM and degree-corrected SBM, and use the rapid entropy equations for the Bernoulli SBM and Multinomial DCSBM.

Finally, we note that an alternative version of entropy that is not based on the blockmodel but instead by the size of the ensemble of networks with identical degree sequence and communities is discussed in Ref. [40].

4. Application of the significance test approach to non-generative models for community structure

The blockmodel entropy significance test provides an estimate of how often a given partition provides a lower-entropy explanation of the data, as viewed through a particular model. While we have, so far, derived expressions for this test in terms of the entropy of a model Eq. (4) or its likelihood Eq. (B11), there exist many other approaches to community detection that are not generative, and therefore have neither a likelihood or an entropy. These models rely on a quality function or Hamiltonian which is optimized over partitions. Supposing that optimization of the Hamiltonian \(Q\) involves maximization, the test statistic is

\[ p\text{-value} = \Pr \left[ Q(\tilde{G}; \pi) \geq Q(\tilde{G}; \mathcal{M}) \right]. \quad (B12) \]

If optimization involves minimization of \(Q\), the direction of the inequality above should be reversed.
Modularity [9], one of the most popular quality functions used for community detection, serves as an instructive example of the blockmodel entropy significance test in two ways. First, it is a measure of the strength of the assortment of links into communities, but has no generative model. Indeed, sampling from the space of networks with a particular modularity NP-hard [57]. Second, the modularity score itself defines community structure narrowly as assortative, and therefore networks with disassortative structures, which have significant \( p \)-values using any SBM as the test model, are likely to be found to have non-significant \( p \)-values when Eq. \((B12)\) is used. This emphasizes both the versatility of the test statistic, as well as the differences between definitions of community structure—spanning generative and non-generative models alike.

It is worth noting that the value of modularity is asymptotically zero whenever assortative communities are uncorrelated with the partition at which it is being evaluated—indeed, the premise of modularity maximization is to find communities whose internal edges defy expectation based on this uncorrelated null model. Thus, if metadata provide a partition of the network, and modularity is found to be exactly zero, then from the perspective of the particular type of assortative structure defined by modularity, there is not a significant relationship between metadata and community structure. On the other hand, simply finding that modularity under a particular metadata partition \( Q(G, M) \) is zero-need not imply that the relationship is or is not statistically significant in the sense of Eq. \((B12)\); the test must be performed.

5. Generation of synthetic networks for blockmodel entropy significance test

The tests described in the main text, and detailed in this Supplementary Text, will yield a \( p \)-value which indicates the extent to which a set of metadata (and a generative model) describes a network better than a random partition. In order to understand the sensitivity of the BESTest, we generated sets of synthetic networks and synthetic metadata, applied the BESTest to them, and produced Fig. 2. Here we describe the process used to generated those synthetic networks.

We generated networks of \( N = 1000 \) nodes and two planted communities \( r \) and \( s \) using the (Bernoulli) SBM. Each node was assigned to one of the communities \( (T_i = r \text{ or } T_i = s) \) with equal probability. We then generated a network with a given community strength \( \epsilon = \omega_{rs}/\omega_{rr} \) such that low values of \( \epsilon \) generate strongly assortative communities with few connecting edges between them and as \( \epsilon \) grows, the generated communities become weaker, producing a random graph with no communities when \( \epsilon = 1 \). For each node \( i \), with probability \( \ell \) we assigned its metadata label to be its community label \( (M_i = T_i) \), otherwise we assigned it a uniformly random label. Thus, as \( \ell \) increases from 0 to 1 the metadata labels correlate more with the planted communities, and the probability that any individual node’s metadata label matches its community label is \( \ell (1) + (1 - \ell)(1/2) = (1 + \ell)/2 \).

6. Additional applications of the BESTest to real data

We now present and discuss the results of applying the BESTest to the Lazega Lawyers and Malaria data sets (see Supplementary Text D).

a. Lazega Lawyers

We applied the BESTest to all three Lazega Lawyers networks (Friendship, Cowork, Advice) which share the same set of nodes but have different sets of edges, representing different relationships between individuals. There were five sets of node metadata (Status, Gender, Office, Practice, and Law School). We applied the BESTest to each combination of network and metadata, using four generative models (SBM, degree-corrected SBM, Poisson SBM, and Poisson degree-corrected SBM). These results are shown in Table S3.

First, note that values between Bernoulli and Poisson models are not identical, though they are similar, implying that the models are not entirely interchangeable. More importantly, however, the results for degree-corrected and degree-uncorrected models are substantially more different, with relationships varying from sig-

| Network | Attribute | Status | Gender | Office | Practice | Law School |
|---------|-----------|--------|--------|--------|----------|------------|
| SBM     | Friendship| $< 10^{-6}$ | 0.034  | $< 10^{-6}$ | 0.033 | 0.134  |
|         | Cowork    | $< 10^{-3}$ | 0.094  | $< 10^{-6}$ | $< 10^{-6}$ | 0.922  |
|         | Advice    | $< 10^{-6}$ | 0.010  | $< 10^{-6}$ | $< 10^{-6}$ | 0.205  |
| DCSBM   | Friendship| $< 10^{-6}$ | 0.001  | $< 10^{-6}$ | 0.002 | 0.094  |
|         | Cowork    | $< 10^{-3}$ | 0.842  | $< 10^{-6}$ | $< 10^{-6}$ | 0.938  |
|         | Advice    | $< 10^{-6}$ | 0.205  | $< 10^{-6}$ | $< 10^{-6}$ | 0.328  |
| Poisson SBM | Friendship| $< 10^{-6}$ | 0.046  | $< 10^{-6}$ | 0.044 | 0.167  |
|         | Cowork    | $< 10^{-3}$ | 0.099  | $< 10^{-6}$ | $< 10^{-6}$ | 0.977  |
|         | Advice    | $< 10^{-6}$ | 0.013  | $< 10^{-6}$ | $< 10^{-6}$ | 0.316  |
| Poisson DCSBM | Friendship| $< 10^{-6}$ | $< 10^{-3}$ | $< 10^{-6}$ | $< 10^{-3}$ | 0.014  |
|         | Cowork    | $< 10^{-4}$ | 0.969  | $< 10^{-6}$ | $< 10^{-6}$ | 0.781  |
|         | Advice    | $< 10^{-5}$ | 0.018  | $< 10^{-6}$ | $< 10^{-6}$ | 0.046  |
TABLE S4. Malaria: BESTest p-values for parasite origin metadata

| Network | Model | SBM | DCSBM | Poi. SBM | Poi. DCSBM |
|---------|-------|-----|-------|---------|-----------|
| Malaria 1 | 0.566 | 0.096 | 0.606 | 0.086 |
| Malaria 2 | 0.064 | 0.148 | 0.066 | 0.143 |
| Malaria 3 | 0.536 | 0.389 | 0.532 | 0.501 |
| Malaria 4 | 0.588 | 0.617 | 0.604 | 0.644 |
| Malaria 5 | 0.382 | 0.077 | 0.369 | 0.087 |
| Malaria 6 | 0.275 | 0.923 | 0.293 | 0.751 |
| Malaria 7 | 0.020 | 0.388 | 0.019 | 0.501 |
| Malaria 8 | 0.464 | 0.176 | 0.468 | 0.172 |
| Malaria 9 | 0.115 | 0.067 | 0.108 | 0.200 |

Significant under one model to insignificant under another. This highlights the fact that metadata can explain patterns of group structure in a network only through the lens of a particular network generative model; a change in the model may impact the metadata’s ability to explain patterns in network community structure.

Second, note that under all models, for each network there exist multiple sets of metadata that are significant. Similarly, there exist multiple networks for which any individual set of metadata is significant. This fundamentally undermines the notion that one should expect a single set of metadata to function as ground truth, given that multiple sets of metadata explain multiple networks.

b. Malaria

We applied the BESTest to nine layers of a network of malaria parasite genes (Malaria 1-9) using four generative models (SBM, degree-corrected SBM, Poisson SBM, and Poisson degree-corrected SBM). Three sets of metadata exist for these networks, (parasite origin, CP group, and UPS), described in detail in Supplementary Text D.

The parasite origin results are shown in Table S4, and none of the p-values listed is significant. This result indicates that when the nodes of each layer are divided into groups based on parasite origin, the entropy of the resulting model is no better than assigning the nodes to groups at random. This implies, in turn, that the malaria parasite antigen genes do not group by the parasite from which they came, confirming previous observations [41]. However, as shown in Fig. 2 the BESTest is sensitive to even small levels of explanatory power provided by metadata, indicating that parasite origin has truly no bearing on the community structure of malaria parasite antigen genes, for all four generative models tested.

On the other hand, it is known that the genes represented by the nodes of the malaria parasite networks are correlated with CP group and UPS metadata. As shown in Tables S5 and S6 the BESTest indeed finds that this is the case, with a handful of exceptions, again confirming previous results that used less sophisticated techniques [41].

TABLE S5. Malaria: BESTest p-values for CP group metadata

| Network | Model | SBM | DCSBM | Poi. SBM | Poi. DCSBM |
|---------|-------|-----|-------|---------|-----------|
| Malaria 1 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 2 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 3 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 4 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 5 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 6 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 7 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 8 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 9 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |

TABLE S6. Malaria: BESTest p-values for UPS metadata

| Network | Model | SBM | DCSBM | Poi. SBM | Poi. DCSBM |
|---------|-------|-----|-------|---------|-----------|
| Malaria 1 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 2 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 3 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 4 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 5 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 6 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 7 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 8 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |
| Malaria 9 | <10^{-5} | <10^{-5} | <10^{-5} | <10^{-5} |

Morpheus: The Matrix is a system, Neo. That system is our enemy. But when you’re inside, you look around, what do you see? ... The very minds of the people we are trying to save. But until we do, these people are still a part of that system and that makes them our enemy. You have to understand, most of these people are not ready to be unplugged. And many of them are so inured, so hopelessly dependent on the system, that they will fight to protect it. [61]
Appendix C: No optimal community detection algorithm

Spoon boy: Do not try and bend the spoon—that’s impossible. Instead, only try to realize the truth.
Neo: What truth?
Spoon boy: There is no spoon.
Neo: There is no spoon?
Spoon boy: Then you will see that it is not the spoon that bends, it is only yourself. [61]

In the main text we argue that that the goal of recovering ground truth communities is ill posed because it lacks a unique solution and we also claim a “No Free Lunch” theorem for community detection. In this Supplementary Text, we describe and expound those claims using technical arguments.

For convenience, we provide a reference table of notation used in derivations in this Supplementary Text.

| Variable | Definition |
|----------|------------|
| $\mathcal{G}$ | a network, $\mathcal{G} = (V, E)$ |
| $N$ | the number of nodes $|V|$ |
| $\mathcal{T}$ | ground truth (planted) partition |
| $C$ | detected communities partition |
| $g$ | generative model, maps a partition to a network. $\mathcal{G} = g(\mathcal{T})$ |
| $f$ | comm. detection method, maps $\mathcal{G}$ to a partition $C = f(\mathcal{G})$ |
| $\ell(\cdot)$ | an error or loss function, returns a scalar |
| $X$ | the space of possible inputs, i.e., all possible graphs $\mathcal{G}$ |
| $Y$ | the space of possible outputs, i.e., all possible partitions |
| $\gamma$ | the true relationship between $X$ and $Y$ |
| $h$ | the hypothesis about the relationship between $X$ and $Y$ |
| $\sigma_{X}$ | probability density over $X$ |
| $\Lambda(\ell)$ | total loss across all possible inputs for loss function $\ell$ |
| $\nu, \nu$ | two partitions of $N$ objects |
| $\Omega$ | the set of all possible partitions of $N$ objects |
| $B_{N}$ | the $N$th Bell number, the number of possible ways that $N$ objects can be partitioned. $B = |\Omega|$ |

1. Ground-truth community detection is an ill-posed inverse problem

A problem that is well posed satisfies three properties: (i) a solution exists, (ii) the solution is unique, and (iii) the solution’s behavior changes continuously with the problem’s initial conditions. The classic example of an ill-posed problem is the inverse heat equation, which violates condition (iii) because its solution (the distribution of temperature in the past) is highly sensitive to changes in the distribution of temperatures at the present. The problem of reproducing ground truth communities $\mathcal{T}$ from a network $\mathcal{G}$ by formulating the correct community detection algorithm $f^{*}$ is ill posed because it fails condition (ii), i.e., community detection has no unique solution.

**Definition:** The ground truth community detection problem: given a fixed network $\mathcal{G}$ in which there has been hidden some ground truth partition $\mathcal{T}$, find the true communities that were planted in, embedded in, or used to generate the network. In other words, given $\mathcal{G}$, find the $\mathcal{T}$ such that $\mathcal{G} = g(\mathcal{T})$.

We now argue that the ground truth community detection problem is ill posed because its solution is not unique. The intuition behind this argument is that any network $\mathcal{G}$ could have been created using many different planted partitions via different generative processes. Therefore, searching for the ground truth partition without knowing the exact generative mechanism is an impossible task; there is no ground truth.

**Theorem 1:** For a fixed network $\mathcal{G}$, the solution to the ground truth community detection problem is not unique.

**Proof:** We first show that the graph $\mathcal{G}$ can be produced by using two different planted partitions, $\mathcal{T}_1$ and $\mathcal{T}_2$ with $\mathcal{T}_1 \neq \mathcal{T}_2$. Let $\mathcal{T}_1$ be the trivial 1-partition in which all vertices are in the same group, and let $g_1$ be the generative model of Erdős–Rényi random graphs with probability $p \in (0, 1)$. Then the model and partition $g_1(\mathcal{T}_1)$ create $\mathcal{G}$ with non-zero probability. Let $\mathcal{T}_2$ be the trivial $N$-partition in which each vertex is in its own group, and let $g_2$ be a generative model that specifies the exact number of edges between all groups, such that $g_2(\mathcal{T}_2)$ produces $\mathcal{G}$ with probability one. We therefore have two partitions $\mathcal{T}_1 \neq \mathcal{T}_2$ and both $g_1(\mathcal{T}_1)$ and $g_2(\mathcal{T}_2)$ can create $\mathcal{G}$. Since two different planted partitions may be responsible for $\mathcal{G}$, both are potential solutions of the community detection problem. Therefore, the solution to the community detection problem is not unique for the network $\mathcal{G}$. To complete the proof, note that the 1-partition and $N$-partition argument above applies equally well to any network $\mathcal{G}$.

The theorem above relies on two trivial partitions, the 1-partition and the $N$-partition in its proof, but other examples exist as well: consider the generative model $g_{D^*}$ which maps any partition that it is given to some fixed network $\mathcal{G}^{*}$, essentially ignoring the information provided by the partition [similar to case (i) in the main text]. These models, while sufficient for the proof, are not particularly interesting for practitioners, yet non-trivial models and partitions also exist for any network. For instance, the Karate Club network may have plausibly been produced by the SBM with a core-periphery partition or by the degree-corrected SBM with a social faction partition [20].

Belief in ground truth $\mathcal{T}$ necessitates a belief in a specific generative mechanism $g$ which together produced the network $\mathcal{G}$. For real-world networks, which may arise
through more complex processes than those described here, we do not know the generative mechanism. Theorem 1 means that, in these cases, it is impossible to recover the true partition because any partition may plausibly have been used to generate the network. Therefore the ground truth community detection problem is ill-posed for any network for which the generative process is unknown because there is no unique solution. Put differently, it is impossible to uniquely solve an inverse problem when the function to be inverted is not a bijection.

2. No Free Lunch for community detection

The “no free lunch” (NFL) theorem [32] for machine learning states that for supervised learning problems, the expected misclassification rate, summed over all possible data sets, is independent of the algorithm used. In other words, averaged over all problems, every algorithm has the same performance. Therefore, if algorithm \( f_1 \) outperforms algorithm \( f_2 \) for one set of problems, then there exists some other set of problems for which algorithm \( f_2 \) outperforms algorithm \( f_1 \). In other words, it is impossible to get overall better performance without some cost; there is no free lunch.

The NFL theorem holds for community detection, and clustering problems in general. Demonstrating this requires that we first translate the community detection problem into the language and notation of the Extended Bayesian Framework (EBF) used in the NFL theorems for supervised learning. Then, under an appropriate choice of error (or “loss”) function \( \ell \), the performance of any community detection method \( f \), summed over all problems \( \{g, T\} \), is identical

\[
\sum_{g, T} \ell(g(T), f(g(T))) = \Lambda(\ell) \quad \forall f ,
\]

(C1)

where \( \Lambda(\ell) \) depends on the particular error function \( \ell \) but is otherwise a constant, representing the total error.

In the following, we map community detection notation to EBF notation, provide a guiding example, and then resolve a subtle issue related to the loss function \( \ell \). We then discuss the implications of this result for future studies of community detection. The proofs of the NFL theorems are not recapitulated here, but are fully detailed in Ref. [32] and discussed extensively elsewhere.

a. Community detection in the Extended Bayesian Framework

The Extended Bayesian Framework (EBF) is a framework—a set of variables, definitions, and assumptions—for supervised learning that provides a clear and precise description of the problem. It is important in both the proof and implications of the NFL theorem, and was formalized at length in Ref. [32]. In what follows, random variables will be denoted by capital letters, e.g. \( X \), while instances of random variables will be denoted by the corresponding lowercase letters, e.g. \( x \). In the EBF, we suppose that there exists an input space \( X \), an output space \( Y \), and that each of these has a countable (but possibly infinite) number of elements, \(|X| = n \) and \(|Y| = r \). The fundamental relationship to be learned is how \( X \) and \( Y \) are related, and to that end, let \( \gamma \) be the true or target relationship between \( X \) and \( Y \), i.e., \( \gamma \) is the conditional distribution of \( Y \), given \( X \). The points in the space \( X \) need not be distributed uniformly either, so we also specify \( \sigma \), the probability density function of points \( x \) in the input space \( X \), i.e., \( P(x|\sigma) = \sigma X \). In the nomenclature of community detection, the input \( x \in X \) is simply the observed graph \( G \), and the output \( y \in Y \) is the true partition into communities \( T \) for the nodes described by \( x \). To solve a community detection problem, we hope to predict the true communities \( y \) from the input graph \( x \); a community detection method will be successful when its hypothesized relationship \( h \) is an accurate representation of the true relationship \( \gamma \) between \( X \) and \( Y \).

In supervised learning, for which the NFL theorems were originally proved, we aim to learn the relationship between \( X \) and \( Y \) from a training set \( d \) which consists of \( m \) ordered pairs of samples from \( X \) and \( Y \), \( \{d_X(i), d_Y(i)\}_{i=1}^m \). In response to the training data, the learning algorithm produces a hypothesis \( h \) in the form of an \( x \)-conditioned probability distribution over values \( y \). The way in which the learning algorithm produces a hypothesis from training sets is described by \( P(h|d) \), the distribution over hypotheses conditioned on the observed data. Note that the algorithm learns from the data alone and is independent of \( \gamma \), i.e., \( P(h|d, \sigma) = P(h|d) \). If the algorithm performs well the hypothesis \( h \) will have high correspondence with the true relationship \( \gamma \). Therefore, in supervised learning, algorithms are evaluated by their ability to make sufficient use of a limited training set to provide good predictions of \( y \) given \( x \) not in the training set. On the other hand, in unsupervised learning—a category which includes clustering and community detection—the training set \( d \) is empty \( (m=0) \), so the prediction \( h \) is based solely on the prior beliefs encoded in the model \( P(h) \). We note that in the NFL theorems for supervised learning, the independence of training data \( d \) from \( \gamma \) and \( \sigma \) is important to establish, but for unsupervised tasks, the set \( d \) is empty so it is trivially independent of \( \gamma \) and \( \sigma \).

To better understand the EBF for community detection, an example is helpful. Consider the problem of finding two planted communities in a network \( G \). The true relationship \( \gamma \) between the network and its partition is hidden. Given only \( G \)—which is a point in the space of graphs \( X \)—fitting the parameters of an SBM, maximizing modularity, or using another method of our choice, produces a hypothesis \( h \), which is a prediction...
about which nodes belong to which groups. If these communities are found correctly by the algorithm, then \( h \) will be highly correlated with the true communities mapped by \( \gamma \). (This is equally true for both hard partitions, where each node belongs to only one group, and soft partitions, where each node may be distributed over multiple groups.) In other words, \( h \) estimates \( \gamma \) based on a point in \( X \) called \( \mathcal{G} \). Because the estimate \( h \) is based only on \( \mathcal{G} \) and the assumptions of the algorithm \( P(h) \), it reproduces \( \gamma \) with possibly limited accuracy, and therefore its community assignments may or may not be highly correlated with the true assignments \( T \in Y \). Increasing the size of the input data set may help with accuracy as well: by generating a larger graph using the same generative model, \( \mathcal{G} \) supplies a different point in \( X \) providing more information to the community detection method. This may allow the estimate \( h \) to produce better predictions of \( \gamma \), thereby producing a more accurate partitioning of nodes into their true communities, but only if the model \( P(h) \) is sufficiently aligned to reality \( P(\gamma) \).

All learning algorithms make some prior assumptions, in the form of \( P(h) \), about the possible relationships between inputs and outputs. For unsupervised methods such as community detection, there is a much greater importance associated with these assumptions because they do not have access to training data. For instance, a supervised algorithm could supposedly start from a uniformly ignorant prior \( P(h) \) and rely on having a sufficiently large training set that \( P(h|d) \) is informative. When there is no training data it is necessary that \( P(h) \) is informative of the possible input-output relationship. Thus, community detection algorithms encode beliefs or definitions of community structure, and these beliefs constitute a prior over the kinds of problems that we expect to see. Some methods, for example, search only for assortative [9, 37] or disassortative [38] community structures, while others are more flexible and can find mixtures of assortative, disassortative, and core-periphery structures [15, 16, 20, 39] and allow for nodes to belong to multiple communities [36, 37].

b. Loss functions and a priori superiority

So far, we have discussed the phrasing of community detection in the language of EBF but have not described the way in which error (also called loss or cost) is measured. The error function quantifies the accuracy of predictions, and the EBF introduces a random variable \( C \) which represents the error associated with a particular \( \gamma \) and \( h \), i.e., the error associated with using a particular algorithm for a particular problem. Conceptually, this is what the community detection literature attempts to estimate when algorithms are compared based on their ability to recover planted communities in synthetic data. More formally, \( C \) is measured by the distribution \( P(c|h, \gamma, d) \), which incorporates the relationships between the test set and the generating process, as well as the way in which the hypothesis is related to the training data. Therefore, the quantity of interest to those developing algorithms is the expected error, \( E(C|h, \gamma, d) \). For example, in the context of supervised learning, choosing the loss function \( \ell \) to be the average misclassification rate is common. For the purposes of community detection, misclassification rate is not of interest for a pedantic but important reason: for community detection and other related unsupervised tasks such as clustering, permutations of the group labels are inconsequential because the partition is the desired outcome; labeling two groups \( a \) and \( b \) is equivalent to labeling them \( b \) and \( a \). As a result, many of the loss functions typically used to compare partitions have a “geometric” structure that implies an \( a \) priori superiority of some algorithms, which would appear to contradict the NFL theorem. We now discuss one such loss function frequently used to evaluate community detection algorithms, the normalized mutual information, and the structure that it imposes on the space of partitions.

Normalized mutual information is an information-theoretic measurement of similarity between two partitions that treats both partitions as statistical objects. For a partition \( u \) of \( N \) objects into \( K_u \) groups, the probability that an object chosen uniformly at random falls into group \( u_i \) is \( p_i = |u_i|/N \), \( i = 1 \ldots K_u \). The entropy associate with a partition \( u \) is then the entropy of its corresponding distribution \( p \),

\[
H(u) = - \sum_{i=1}^{K_u} p_i \log (p_i) .
\]

When comparing two partitions \( u \) and \( v \) of the same set of objects, each object belongs to some group \( u_i \) in the first partitions and some other group \( v_j \), \( j = 1 \ldots K_v \) in the second partition, with the corresponding probability \( p_{ij} \). The mutual information between the two partitions
is therefore

\[ I(u, v) = \sum_{i=1}^{K_u} \sum_{j=1}^{K_v} p_{ij} \log \left( \frac{p_{ij}}{p_i p_j} \right), \]

which can be normalized to define normalized mutual information as

\[ \text{NMI}(u, v) = \frac{I(u, v)}{\sqrt{H(u)H(v)}}. \quad (C2) \]

Other normalizing factors in the denominator are possible, including \( \frac{1}{2} \left[ H(u) + H(v) \right] \) and \( \max \{ H(u), H(v) \} \); see [58]. NMI maps partitions to the unit interval, with 0 indicating that two partitions are uncorrelated and 1 indicating that they are identical (even if the groups labels differ).

To understand how an error function imposes a geometric structure, consider a simple problem (unrelated to community detection) of predicting, based on some inputs \( X \), a point in the unit circle in \( Y = \{ y \mid \| y \| \leq 1, \ y \in \mathbb{R}^2 \} \). If all points in \( Y \) are equally likely, then an algorithm that guesses the center of the circle \( h = 0 \) will outperform an algorithm that guesses a point on the boundary \( h \in \partial Y \), simply due to the fact that the center of the circle is, on average, closer to the other points of the circle than any boundary point. Normalized mutual information imposes a geometric structure on the space of partitions in a similar way.

Consider a loss function based on normalized mutual information (NMI) and imagine a community detection algorithm that entirely ignores the network and simply returns a fixed partition of the vertices. As in the example above, NMI provides a geometrical structure on the space of partitions, an algorithm that always returns a partition toward the middle of the space of partitions will outperform an algorithm that always returns a partition on the boundary of that space. To demonstrate this point, Fig. S6 shows all five possible partitions of three vertices, and Table S8 shows their NMI for all pairwise comparisons. Averaged over all possible correct answers, an algorithm that consistently predicts partition 5 will outperform all others, and an algorithm that consistently predicts partition 1 will underperform all others. However, this structure is a known issue of NMI, and so other error functions and corrections have been proposed such as the adjusted mutual information (AMI), which accounts for the geometry of the space [58]. Table S9 shows the AMI for the same set of partitions, and the expected AMI is zero except for the partition that contains only a single group and the partition of each node into separate groups. In the case of these partitions, the 1-partition and the \( N \)-partition, the expected AMI is the reciprocal of the Bell number \( B_N \)—the Bell number is the total number of distinct ways that \( N \) objects can be partitioned, and it grows superexponentially with \( N \)—so as the number of vertices \( N \) increases, so AMI approaches 0 superexponentially; for even small networks, \( 1/B_N \approx 0 \). In this way, AMI provides a “geometry-free” space in which no one partition is \( a \ priori \) closer to all others. This key property of AMI, called homogeneity, is proved in a Lemma in the next section.

c. Lemma and theorems

We now prove a lemma about adjusted mutual information, and then formally state the NFL theorem for supervised learning and prove the no free lunch theorem for community detection.

**Lemma 1:** Adjusted mutual information (AMI) is a homogenous loss function over the interior of the space of partitions of \( N \) objects. Including the boundary partitions, i.e., the 1-partition and the \( N \)-partition, AMI is homogenous within \( \mathcal{B}_N \).

**Proof:** Showing that AMI is a homogenous loss function requires that we show

\[ L(u) = \sum_{v \in \Omega} \text{AMI}(u, v) \quad (C3) \]

is independent of \( u \), where \( \Omega \) is the space of all partitions of \( N \) objects. Stated plainly, if \( L(u) \) is independent of \( u \), it means that the total AMI between partition \( u \) and all possible partitions will be the same, no matter which
partition $u$ is chosen. The definition of AMI is:

$$\text{AMI}(u, v) = \frac{I(u, v) - E[I(u, v)]}{\sqrt{H(u)H(v)} - E[I(u, v)]}$$

where $I$ is mutual information and $H$ is entropy [58]. The AMI takes on a value of 1 when two partitions are identical and a value of 0 when they are only correlated to the extent that one would expect by chance. In particular, the expectation $E$ is taken over all possible pairs of partitions $u'$ and $v'$ such that $u'$ has the same number of groups and the same number of objects belonging to each group as does $u$, and likewise for $v'$ and $v$. In this way, the expectation $E$ is taken over all pairs of divisions that preserve the group sizes of the two partitions being compared. For convenience of notation, let $N$ be subdivided into non-overlapping subsets $I$ such that every partition $v \in \phi$ has the same number of groups and same number of objects in each group. The set of all partitions $\Omega$ may be subdivided into non-overlapping subsets $\{\phi_i\}$, such that $\cup_i \phi_i = \Omega$ and $\phi_i \cap \phi_j = \emptyset$ for any $i \neq j$. (For example, in Fig. 86, partition 1 belongs to $\phi_1$, partitions 2, 3, and 4 belong to $\phi_2$, and partition 5 belongs to $\phi_3$.) Let the particular subset $\phi_i$ to which a partition $u$ belongs be denoted by $\phi(u)$.

Prior to proceeding, we note that there are two special boundary partitions, the 1-partition in which all objects, so that $|\phi(N)| = N!$. Because there is only one element of $\phi(1)$, it is necessarily true that $I(1, 1) = E[I(1, 1)] = H(1)$. Thus, for this special case, the numerator and denominator of AMI are identical, and AMI($1, 1$) = 1. Similarly, because the set $\phi(N)$ contains every possible permutation of the labels of the objects, yet all partitions are identical, $I(N, N) = E[I(N, N)] = H(N)$, and so AMI($N, N$) = 1.

In order to prove Eq. (C3), we will show that $L(u) = 0$ for all $u$ except 1 and $\bar{N}$ by demonstrating that the numerator of the definition of AMI is 0, specifically,

$$\sum_{v \in \Omega} [I(u, v) - E[I(u, v)]] = 0 \quad \forall \ u \neq \bar{1} \text{ or } \bar{N} . \quad (C4)$$

In fact, we will show that Eq. (C4) holds by breaking the entire sum over all partitions $\Omega$ into sums over each of its disjoint subsets $\{\phi_i\}$, and proving that

$$\sum_{v' \in \phi(v)} [I(u, v') - E[I(u, v')]] = 0$$

$\forall u$ and $\forall v$ except $u = \bar{v} = \bar{1}$ or $u = \bar{v} = \bar{N} . \quad (C5)$

In other words, we will show that the numerator of the definition of AMI is equal to zero when summed over any subset $\phi(v)$ for any fixed partition $u$, except the boundary cases that both $u$ and $v$ are equal to $\bar{1}$ or both are equal to $\bar{N}$. We first examine the expectation term in Eq. (C5). Recall that the expectation is taken over all pairs of members of the subsets $\phi(u)$ and $\phi(v)$, respectively,

$$E[I(u, v)] = \frac{1}{|\phi(u)||\phi(v)|} \sum_{u' \in \phi(u)} \sum_{v' \in \phi(v)} I(u', v') . \quad (C6)$$

In fact, because the sums above are taken over the subsets $\phi(u)$ and $\phi(v)$ that contain $u$ and $v$, the expected mutual information is equal to a constant $\zeta$ for any pair of partitions drawn from $\phi(u)$ and $\phi(v)$,

$$E[I(u, v)] = \zeta \quad \forall \ u \in \phi(u) \text{ and } \forall \ v \in \phi(v) . \quad (C7)$$

Note then that we may rewrite the sum over expectations in Eq. (C5) as $\sum_{v' \in \phi(v)} E[I(u, v')] = |\phi(v)| \zeta$. Therefore, it remains to be shown that the sum over mutual informations in Eq. (C5) is also equal to $|\phi(v)| \zeta$,

$$\sum_{v' \in \phi(v)} I(u, v') = |\phi(v)| \zeta . \quad (C8)$$

To see that this is true, despite the fact that $u$ is fixed (and not averaged over all $u' \in \phi(u)$ as in $E[I(u, v)]$), note that Eq. (C8) nevertheless sums over every $v' \in \phi(v)$ which is the set of every randomization of the partition $v$, provided group sizes are held constant. Because this includes all relabelings (or reinexdings) of the $N$ objects being partitioned, it must be true that,

$$\sum_{v' \in \phi(v)} I(u_1, v') = \sum_{v' \in \phi(v)} I(u_2, v') \text{ whenever } u_1 \in \phi(u_2) . \quad (C9)$$

In other words, the sum of mutual information between a fixed partition $u_1$ and all members of a subset $\phi(v)$ must be equal to the sum of mutual information between a different fixed partition $u_2$ and the same subset $\phi(v)$, but only if $u_1$ and $u_2$ both belong to the same subset as each other. Therefore, Eq. (C8) is true, meaning that the sum over the two terms in Eq. (C5) is zero, independent of $u$. This first implies that the AMI between any boundary partition and any interior partition is 0, AMI($u, 1$) = 0 for any $u \neq 1$ and AMI($u, N$) = 0 for any $u \neq N$. This, in turn, implies Eq. (C4) is true. This completes the proof of the first statement, that Eq. (C3) is true, and in particular, $L(u) = 0$, for any $u \neq \bar{1}, \bar{N}$ and AMI is homogeneous over all non-boundary partitions.

In the special cases of $u = v = \bar{1}$ and $u = v = \bar{N}$, note that we have already shown that AMI($1, 1$) = 1, AMI($N, N$) = 1, and AMI($u, \bar{1}$) = 0 for any $u \neq \bar{1}$ and AMI($u, \bar{N}$) = 0 for any $u \neq \bar{N}$. Therefore,

$$L(\bar{1}) = \sum_{v \in \bar{1}} \text{AMI}(\bar{1}, v) = B_N^{-1} ,$$

$$L(\bar{N}) = \sum_{v \in \bar{N}} \text{AMI}(\bar{N}, v) = B_N^{-1} , \quad (C10)$$

completing the proof of the second statement: including the boundary points, AMI is homogenous within an additive constant $B_N^{-1}$.
Theorem 2 (Wolpert 1996): For homogeneous loss $\ell$, the uniform average over all $\gamma$ of $P(c|\gamma, d)$ equals $\Lambda(c)/r$.

Proof: See Ref. [32].

Theorem 3 (No free lunch for community detection): For the community detection problem with a loss function of adjusted mutual information, the uniform average over all $\gamma$ of $P(c|\gamma)$ equals $\Lambda(c)/r$.

Proof: Lemma 1 proves that adjusted mutual information is homogeneous and applying Theorem 2 with $d = \emptyset$ completes the proof.

$d$. Implications

No free lunch for community detection means that, uniformly averaged over all community detection problems, and evaluated by AMI, all algorithms have equivalent performance. Phrased more usefully, it means that any subset of problems for which an algorithm outperforms others is balanced by another subset for which the algorithm underperforms others. Thus, there is no single community detection algorithm that is best overall.

On the other hand, if the set of problems of interest is a non-uniform subset of all problems, then one algorithm may outperform another on this subset. In other words, the bias of an algorithm to solving a particular type of community detection problem may be its strength, accepting the fact that such an advantage must be balanced by disadvantages elsewhere. For instance, algorithms like the unconstrained SBM (which can find both assortative and disassortative communities and mixtures and gradations thereof) are not universally superior to versions of the SBM constrained to find only assortative or disassortative communities [38]—if the particular subset of problems is believed to contain only disassortative communities, then the unconstrained SBM will not perform as well as a constrained one. In other words, no free lunch for community detection means that matching the assumptions in the model to the underlying generative process can lead to better, more accurate results, but only in the cases when the beliefs about the underlying generative process are accurate; in the other cases, the same model assumptions that improved performance on some problems will diminish it for others. To some extent we expect the distribution of problems to be non-uniform in general. Out of all the possible ways of constructing a graph there may be some types of graph we are less likely to observe. For each graph we can also expect that of all the possible partitions, many will correspond to random assignments of nodes that are not useful in any application. Put differently, there may be some problems we do not wish to solve—but, unless we know which problems they are, it offers us little or no benefit in practice. We note that relatively little is known about which algorithms perform better than others within particular domains or on particular classes of networks. A valuable line of future research on community detection will be developing such an understanding [50, 51].

Cypher: You know, I know this steak doesn’t exist. I know that when I put it in my mouth, the Matrix is telling my brain that it is juicy and delicious. After nine years, you know what I realize? ... Ignorance is bliss. [61]
Appendix D: Datasets and additional methodology

Neo: Whoa. [61]

1. Lazega Lawyers networks

The Lazega Lawyers network is a multilayer network consisting of 71 attorneys of a law firm with three different sets of links, corresponding to friendships, exchange of professional advice, and shared cases [42]. The original study also collected five sets of categorical node metadata, corresponding to status (partner or associate), gender, office location, type of practice (corporate or litigation), and law school (Harvard, Yale, UConn, other). The relationships and dynamics within the law firm were studied extensively in the initial publication of these data sets, but they were not primarily analyzed as complex networks.

2. Malaria var gene networks

The Malaria data set consists of 307 var gene sequences from the malaria parasite P. falciparum [41]. Each var gene encodes a protein that the parasite uses to evade the human immune system, and therefore this family of genes is under intense evolutionary pressures from the human host. The original study focused on uncovering the functional and evolutionary constraints on var gene evolution by identifying community structure in var gene networks.

These sequences were independently analyzed at 9 loci (locations within the genes), producing 9 different genetic-substring-sharing networks with the same node set. In other words, there are 9 layers in this multilayer network. Each parasite genome contains around 60 var genes, and the 307 genes in this data set represent seven parasite genomes. The original study included three sets of categorical node metadata, corresponding to the upstream promoter sequence classification (UPS, \( K = 3 \)), CysPoLV groups (CP \( K = 6 \)), and the parasite genome from which sequence was generated (parasite origin \( K = 7 \)).

3. Karate Club network

The Zachary Karate Club represents the observed social interactions of 34 members of a karate club [14]. At the time of study, the club fell into a political dispute and split into two factions, which are treated as metadata. The Karate Club has been analyzed exhaustively in studies of community detection, and its faction metadata have often been used as ground truth for community detection, due to the network’s small size and easily interpretable social narrative.

4. Generation of log-likelihood surface plots

The log-likelihood surface plots in Figs. 1, 3, 4, S1, S2, and S3 illustrate the changes in log likelihood as the partition of network nodes is varied. In the figures, we show surfaces that appear to be continuous over that two dimensional space, in spite of the fact that the true space of partitions is high dimensional and discretized, and so here we explain the methods used to produce visually meaningful plots.

Plots were generated in three steps: partition sampling, data projection and surface interpolation. For most networks it is infeasible to calculate the log likelihood of all possible partitions, so we instead sampled a subset of partitions. We began with the set of partitions along the path of the neoSBM (e.g., Fig. 3) and sampled partitions around the local neighborhood of this initial set. Specifically, we did so by selecting two partitions uniformly at random from the initial set and created each new partition by assigning \( q \) nodes (chosen randomly and uniformly) to the group assignment of the first partition and the remaining \( N - q \) nodes to that of the second partition.

Next, we projected the \( K^N \)-dimensional partition data down to two dimensions using Multi-dimensional Scaling (MDS) [59] and variation of information [60] as a similarity measure. The outcome of this projection was a two-dimensional representation of the partition space that preserves the variation of information between partitions.

Finally, we used MATLAB’s scatteredInterpolant function with natural interpolation to fit an interpolated surface to the data, which we evaluated over a grid of domain points and smoothed using a Gaussian kernel to improve legibility. The processes of embedding, interpolating, and smoothing are not particularly sensitive to changes in parameters or grid resolutions.

In the special case of Fig. 4, we also plotted the partitions of the neoSBM in addition to the interpolated log-likelihood surface to illustrate the neoSBM’s path in the broader context of the surface. There were no modifications or smoothing of the points of the neoSBM’s path beyond the embedding process described above.

Neo: I know you’re out there. I can feel you now. I know that you’re afraid ... you’re afraid of us. You’re afraid of change. I don’t know the future. I didn’t come here to tell you how this is going to end. I came here to tell you how it’s going to begin. I’m going to hang up this phone, and then I’m going to show these people what you don’t want them to see. I’m going to show them a world without you. A world without rules and controls, without borders or boundaries. A world where anything is possible. Where we go from there is a choice I leave to you. [61]