The ground truth about metadata and community detection in networks

Leto Peel,[1,2,*] Daniel B. Larremore,[3,†] and Aaron Clauset[4,5,3,‡]

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

Across many scientific domains, there is common need to automatically extract a simplified view or a 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 the planted 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 induces 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 no algorithm can perform better than any other across all inputs. However, node metadata still have value and 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 models. We demonstrate these techniques using both synthetic and real-world networks, and for multiple types of metadata and community structure.

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 dataset.

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 the method's ability to identify so-called *ground truth* communities, which works well in artificially generated networks, whose links are explicitly placed according to the existence of these communities, which are called a planted partition. However, for real-world networks, the true data generating process is typically unknown, which necessarily implies that there can be no ground truth communities for real networks. A common goal in searching for communities in a real network is to produce insights about this true but unknown data generating process, by reducing a large and complicated system to a simpler set of interacting units. But without access to the very thing these methods are intended to find, objectively evaluating their performance is difficult.

Instead, it has become standard practice to treat some observed data on the nodes of a network (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 argument against treating metadata as ground truth communities. We then present two novel methods that can be used effectively to explore the relationship between metadata and community structure, and we demonstrate these methods on both synthetic and real-world networks.

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

* Contributed equally; leto.peel@uclouvain.be
† Contributed equally; larremore@santafe.edu
‡ aaron.clauset@colorado.edu

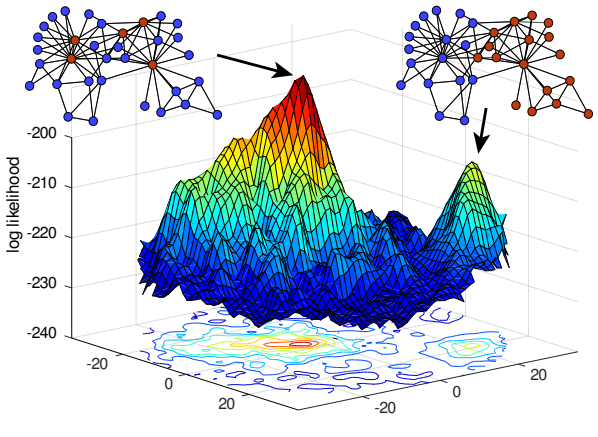

FIG. 1. The SBM likelihood surface for bipartitions of the Karate Club network [12] shows two distinct peaks that represent scientifically reasonable partitions. The lower peak corresponds to the social group partition given by the metadata—often treated as ground truth—while the higher peak corresponds to a leader-follower partition.

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.

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 [9]). Indeed, recent critiques of the general utility of community detection in networks [9–11] 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 [12] 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, these factions are not the only scientifically reasonable way to partition the network. Figure 1 shows the landscape for a large number of two-group partitions of the Karate Club, under the stochastic blockmodel (SBM) for community detection [13, 14]. 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 [15–17], suggest that the best divisions of this network have more than two communities [18, 19]. 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 [20] or do not correlate with the metadata at all [21, 22]. More generally, most real-world networks have many good partitions, there are many plausible ways to sort all partitions to find good ones, and there is no consensus on which method to use on which type of network [22, 23].

In what follows, we first rigorously develop the arguments against the use of metadata as community ground truth. We then introduce two techniques that productively explore the relationship between observed metadata and community structure, and apply both methods to a variety of synthetic and real-world networks, using 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.

GROUND TRUTH AND METADATA IN COMMUNITY DETECTION

Community detection is an inverse problem. Suppose that some generative 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 only 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 graph seeks a method f^* that minimizes the distance between the identified communities and the ground truth:

$$f^* = \arg\min_f d(\mathcal{T}, f(\mathcal{G})) \ , \tag{1}$$

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 [10, 24, 25]. 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,

$$\exists\, f^* \quad s.t. \quad f^* = \arg\min_f d\big(\mathcal{T}, f\left(g(\mathcal{T})\right)\big) \quad \forall\{g, \mathcal{T}\} \ . \tag{2}$$

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 a bijection. Any network $\mathcal{G}$ can result from 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 $(g_1, \mathcal{T}_1) \neq (g_2, \mathcal{T}_2)$ (see Theorem 1 in Supplemental Text A). Thus, no community detection algorithm method can uniquely solve the problem for all possible networks (Eq. (2)), or even a single network (Eq. (1)).

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

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

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, $d(\mathcal{M}, \mathcal{T}) = 0$. Consequently, both $d(\mathcal{C}, \mathcal{T}) = 0$ and $d(\mathcal{C}, \mathcal{T}) > 0$ 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., $d(\mathcal{M}, \mathcal{T})$ [case (ii) in the previous section] and the community detection method's ability to find true communities, i.e., $d(\mathcal{C}, \mathcal{T})$ [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. 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, there is no best overall community detection method. As in supervised learning, we prove a No Free Lunch theorem [26] for community detection (Theorem 3, Supplemental Text A). That is, no method has an *a priori* advantage over any other. 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. On the other hand, 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 be a method that outperforms others within the confines of that subset. In short, matching beliefs about the generative process g with the assumptions of the algorithm f can lead to better, more accurate, but restricted results. (See Supplemental Text A for additional discussion.)

RELATING METADATA AND STRUCTURE

Metadata labels describe the nodes, while communities describe how nodes interact. Correspondence between metadata and communities does 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 [22], learn the relationship between metadata and network topology [27, 28] and explain dependencies between metadata and network structure [29].

Here we propose two new methods to explore how metadata relates 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 is a statistical test to assess whether or not the metadata partition and network structure are related [case (i)]. The second 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., [13, 14, 30–33]) of communities in networks could be used within these methods. Here we derive results for the popular stochastic blockmodel [13, 14] and its degree-corrected equivalent [21] (alternative formulations discussed in Supplemental Texts B and C). 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 their community assignment, i.e., $p_{ij} = \omega_{\pi_i,\pi_j}$, where π_i is the community assignment for node i and ω_{π_i,π_j} is the probability that a link exists between members of groups π_i and π_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*, 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, the better the metadata describe the network.

Here, we use the SBM with maximum likelihood parameters for the partition induced by the metadata, which is given by $\hat{\omega}_{rs} = \frac{m_{rs}}{n_r n_s}$ where m_{rs} is the number of links between group r and group s and n_r is the number of nodes in group r. Then the entropy $H_{\text{SBM}}(\mathcal{G};\mathcal{M})$ can be calculated as a sum of entropies for every possible link (see Supplemental Text B). The statistical significance of this entropy value is obtained by comparing it to a null distribution of such values, derived by computing the entropies induced by random permutations $\{\tilde{\pi}\}$ of the observed metadata values $H(\mathcal{G};\tilde{\pi})$. This choice of null model preserves both the empirical network structure and the relative frequencies of metadata values, but removes the correlation between the two. The result is a standard p-value, defined as

$$p\text{-value} = \Pr\left[H(\mathcal{G};\tilde{\pi}) \leq H(\mathcal{G};\mathcal{M})\right] . \tag{4}$$

Smaller p-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. Recently, Bianconi *et al.* [34] 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. S2).

To gain some intuition as to how this p-value behaves, we first apply it to synthetic networks with known community structure (see Supplemental Text B). 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 p-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.

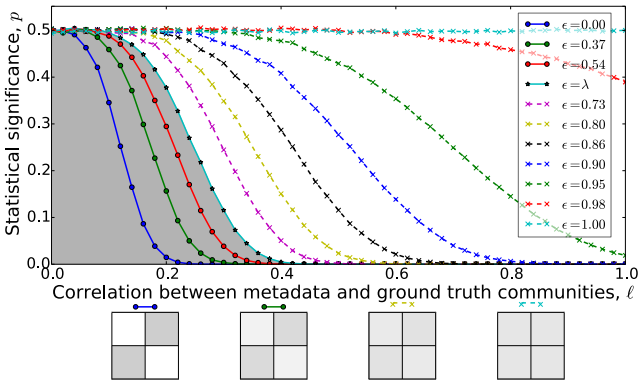

FIG. 2. Expected p-value estimates of the blockmodel entropy significance test for metadata with varying correlation ℓ with two equally sized planted communities (see Supplemental Text B). Each curve represents networks with a fixed community strength $\epsilon = \omega_{rs}/\omega_{rr}$. Solid lines indicate strong community structure in the so-called detectable regime ($\epsilon < \lambda$), while dashed lines represent weak undetectable communities ($\epsilon > \lambda$) [8]. Four block density diagrams visually depict ϵ values.

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 Supplemental Text D). The first set, the Lazega Lawyers networks, comprises three networks on the same set of nodes and five metadata attributes. The multiple combinations of edge and metadata types that yield highly significant p-values (Table S5) 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 strain-of-origin, under both the SBM and the degree-corrected SBM (Table S6). In contrast to the Lazega Lawyers network, these strain metadata are statistically irrelevant for explaining the observed patterns of gene recombinations. This finding substantially strengthens the conclusions of Ref. [35] which used a less sensitive test based on label assortativity. Some metadata for these networks do correlate, however (see Supplemental Text B).

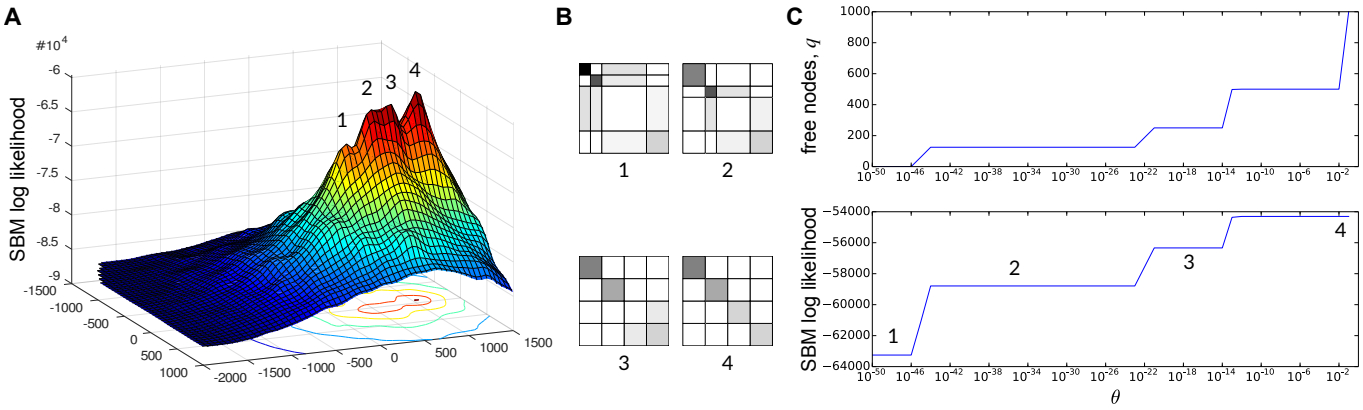

FIG. 3. The neoSBM on synthetic data. (A) The SBM likelihood surface shows four distinct peaks corresponding to a sequence of locally optimal partitions. (B) Block density diagrams depict community structure for locally optimal partitions, where darker color indicates higher probability of interaction. (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 θ show three discontinuous jumps as the neoSBM traverses each of the locally optimal partitions (1–4).

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 accomlish 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 θ, which controls for the additional freedom that comes from varying q. The neoSBM's log likelihood is $\mathcal{L}_{\text{neoSBM}} = \mathcal{L}_{\text{SBM}} + qf(\theta)$, where $f(\theta)$ may be interpreted as the cost of freeing a node from its metadata label (see Supplemental Text C for exact formulation).

By varying the cost of freeing a node, we can use the neoSBM to produce a graphical diagnostic of how the metadata and inferred community partitions are related. 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 θ, 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.

We examine the path between partitions in terms of the SBM log likelihood and the number of free nodes as a function of θ for a synthetic network with four locally optimal partitions (see Supplemental Text C), which correspond to the four distinct peaks in the surface plot (Fig. 3A). We take the partition of the lowest of these peaks as metadata and use the neoSBM to generate a path to the globally optimal partition. This produces three discontinuous jumps in log likelihood and number of free nodes (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 blockwise interaction matrices ω. 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

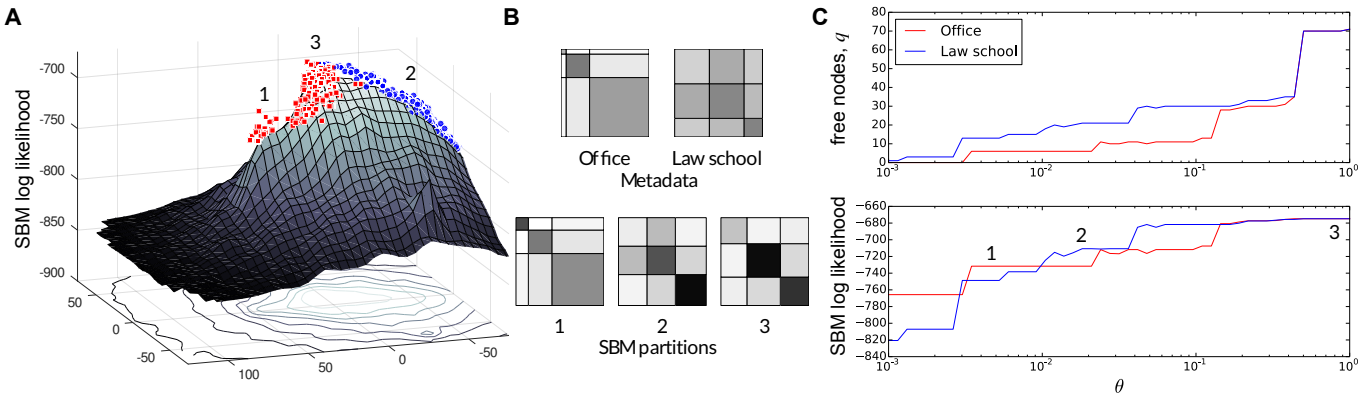

FIG. 4. The neoSBM on Lazega Lawyers friendship data [36]. (A) Points of two neoSBM paths using office (red) and law school (blue) metadata partitions are shown on the SBM likelihood surface (greyscale to emphasize paths). (B) Block density diagrams depict community structure for metadata, (1–2) intermediate optimal, and (3) globally optimal partitions, where darker color indicates higher probability of interaction. (C) The neoSBM traverses two distinct paths to the global optimum (3), but only the path beginning at the office metadata partition traverses a local optimum (1), indicated by a plateau in free nodes q and log likelihood.

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 metadata are relevant, however, as we determined in the previous section. Results for other real-world networks are included in Supplemental Text C, including generalizations of the neoSBM to degree-corrected SBMs.

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 [9, 11, 25, 37]. Consequently, metadata recovery is often used to measure community detection performance [38] and metadata are often referred to as ground truth [22, 39]. 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), and explored the mathematical and practical problems of treating metadata as ground truth.

By focusing on detecting communities that are highly correlated with metadata, we risk overlooking other scientifically relevant organizational patterns. Disagreements between metadata labels and community detection results may in fact point to interesting or unexpected generative processes. For instance, in the Karate Club network, there is one node whose metadata label is not recovered by most algorithms. In this case, although the student had more social ties to the president's group, he chose to join the instructor's group so as not to lose his progress toward his black belt [12]. In other cases, metadata may provide a narrative that blinds us to additional structure, exemplified by a network of political blogs [1] in which liberal and conservative blogs formed two highly assortative groups. Consequently, recovery of these two groups has been used as a signal that a method produces "good" results [21]. A deeper analysis, however, suggests that this network is better described by subdividing these two groups, a step that reveals substantial substructure within the dominant patterns of political connectivity [20, 33]. These subgroups remained overlooked in part because the metadata labels aligned closely with an attractively simple narrative.

The task of community detection is the network analogy of data clustering. Whereas clustering divides a set of vectors into groups with similar attribute patterns, community detection divides a network into groups of nodes with similar connectivity patterns. The general problem of clustering, however, is notoriously slippery [40] and cannot be solved universally [41]. Essentially, which clustering is optimal depends on its subsequent uses, and similar constraints apply to community detection [42].

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 are no free lunches in community detection.

Instead, algorithmic biases that improve performance on one class of networks must reduce performance on others. 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 [43, 44].

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 [22] and identify relevant communities to predict missing network or metadata information [27, 28, 45]. 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

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). The authors thank Cris Moore, Tiago Peixoto, Michael Schaub and David Wolpert for insightful conversations.

[1] L. A. Adamic and N. Glance, in *Proc. of the 3rd Int. Workshop on Link Discovery* (ACM, 2005) pp. 36–43.
[2] S. Fortunato, Physics Reports, **486**, 75 (2010).
[3] P. Holme, M. Huss, and H. Jeong, Bioinformatics, **19**, 532 (2003).
[4] R. Guimera and L. A. N. Amaral, Nature, **433**, 895 (2005).
[5] C. Cortes, D. Pregibon, and C. Volinsky, in *Advances in Intelligent Data Analysis*, Lecture Notes in Computer Science, Vol. 2189, edited by F. Hoffmann, D. Hand, N. Adams, D. Fisher, and G. Guimaraes (Springer Berlin / Heidelberg, 2001) pp. 105–114.
[6] L. S. Haggerty, P.-A. Jachiet, W. P. Hanage, D. A. Fitzpatrick, P. Lopez, M. J. OConnell, D. Pisani, M. Wilkinson, E. Bapteste, and J. O. McInerney, Mol. Biol. Evol., **31**, 501 (2014).
[7] P. Erdős and A. Rényi, Publ. Math. Debrecen, **6**, 290 (1959).
[8] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová, Phys. Rev. Lett., **107**, 065701 (2011).
[9] D. Hric, R. K. Darst, and S. Fortunato, Phys. Rev. E, **90**, 062805 (2014).
[10] J. Leskovec, K. J. Lang, and M. Mahoney, in *Proc. of the 19th Int. World Wide Web Conf.* (ACM, 2010) pp. 631–640.
[11] J. Yang and J. Leskovec, in *Proc. of the 12th Int. Conf. on Data Mining* (IEEE, 2012) pp. 1170–1175.
[12] W. W. Zachary, J. Anthropol. Res., 452 (1977).
[13] P. W. Holland, K. B. Laskey, and S. Leinhardt, Social networks, **5**, 109 (1983).
[14] K. Nowicki and T. A. B. Snijders, J. American Statistical Association, **96**, 1077 (2001).
[15] M. Girvan and M. E. Newman, Proc. Natl. Acad. Sci. USA, **99**, 7821 (2002).
[16] A. Sinclair and M. Jerrum, Information and Computation, **82**, 93 (1989).
[17] J. Shi and J. Malik, IEEE Transactions on Pattern Analysis and Machine Intelligence, **22**, 888 (2000).
[18] X.-Q. Cheng and H.-W. Shen, J. Stat. Mech.: Theory Exp., **2010**, P04024 (2010).
[19] T. S. Evans, J. Stat. Mech.: Theory Exp., **2010**, P12037 (2010).
[20] F. Krzakala, C. Moore, E. Mossel, J. Neeman, A. Sly, L. Zdeborov, and P. Zhang, Proc. Natl. Acad. Sci. USA, **110**, 20935 (2013).
[21] B. Karrer and M. E. Newman, Phys. Rev. E, **83**, 016107 (2011).
[22] M. E. Newman and A. Clauset, Nature Communications, **7**, 11863 (2016).
[23] B. H. Good, Y.-A. de Montjoye, and A. Clauset, Phys. Rev. E, **81**, 046106 (2010).
[24] A. Lancichinetti and S. Fortunato, Phys. Rev. E, **80**, 056117 (2009).
[25] J. Yang and J. Leskovec, Knowledge and Information Systems, **42**, 181 (2015).
[26] D. H. Wolpert, Neural computation, **8**, 1341 (1996).
[27] L. Peel, in *Proc. of the 14th Int. Conf. on Information Fusion* (IEEE, 2011) pp. 1–8.
[28] L. Peel, in *ECML/PKDD Workshop on Collective Learning and Inference on Structured Data (CoLISD)* (2012) arXiv 1209.5561.
[29] B. K. Fosdick and P. D. Hoff, J. Am. Stat. Assoc., **110**, 1047 (2015).
[30] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing, in *Advances in Neural Information Processing Systems* (2009) pp. 33–40.
[31] B. Ball, B. Karrer, and M. Newman, Phys. Rev. E, **84**, 036103 (2011).
[32] D. B. Larremore, A. Clauset, and A. Z. Jacobs, Phys. Rev. E, **90**, 012805 (2014).
[33] T. P. Peixoto, Phys. Rev. X, **4**, 011047 (2014).
[34] G. Bianconi, P. Pin, and M. Marsili, Proc. Natl. Acad. Sci. USA, **106**, 11433 (2009).
[35] D. B. Larremore, A. Clauset, and C. O. Buckee, PLoS Comput Biol, **9**, e1003268 (2013).
[36] E. Lazega, *The collegial phenomenon: The social mechanisms of cooperation among peers in a corporate law partnership* (Oxford University Press on Demand, 2001).
[37] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, Nature, **466**, 761 (2010).

[38] S. Soundarajan and J. Hopcroft, in *Proc. of the 21st Int. World Wide Web Conf.* (ACM, 2012) pp. 607–608.
[39] T. Chakraborty, S. Sikdar, V. Tammana, N. Ganguly, and A. Mukherjee, in *Proc. of the Int. Conf. on Advances in Social Networks Analysis and Mining* (IEEE/ACM, 2013) pp. 426–433.
[40] U. von Luxburg, B. Williamson, I. Guyon, *et al.*, J. of Machine Learning Research, **27**, 65 (2012).
[41] J. Kleinberg, Advances in neural information processing systems, 463 (2003).
[42] A. Browet, J. M. Hendrickx, and A. Sarlette, arXiv preprint arXiv:1603.00621 (2016).
[43] L. Peel, J. of Advances of Information Fusion, **6**, 119 (2011).
[44] Z. Yang, R. Algesheimer, and C. J. Tessone, Scientific Reports, **6**, 30750 (2016).
[45] D. Hric, T. P. Peixoto, and S. Fortunato, arXiv preprint arXiv:1604.00255 (2016).
[46] N. X. Vinh, J. Epps, and J. Bailey, in *Proc. 26th Int. Conf. on Machine Learning* (ACM, 2009) pp. 1073–1080.
[47] T. P. Peixoto, Phys. Rev. E, **85**, 056122 (2012).
[48] T. P. Peixoto, Phys. Rev. E, **89**, 012804 (2014).

Appendix A: No optimal community detection algorithm

Spoon boy: *Do not try and bend the spoonthat'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.*

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 Supplemental Text, we describe and expound those claims using technical arguments.

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

TABLE S1. Notation used in this Supplemental Text

Variable	Definition
$\mathcal{G}$	a network, $\mathcal{G} = (V, E)$
N	the number of nodes $\lvert V\rvert$
$\mathcal{T}$	ground truth (planted) partition
$\mathcal{C}$	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 $\mathcal{C} = f(\mathcal{G})$
$\ell(\cdot,\cdot)$	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
γ	the true relationship between X and Y
h	the hypothesis about the relationship between X and Y
σ_X	probability density over X
$\Lambda(\ell)$	total loss across all possible inputs for loss function ℓ
u, v	two partitions of N objects
Ω	the set of all possible partitions of N objects.
$\mathcal{B}_N$	the Nth Bell number, the number of possible ways that N objects can be partitioned. $\mathcal{B} = \lvert\Omega\rvert$

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 $g_{\mathcal{G}^*}$ 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 practictioners, yet nontrivial 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 [21].

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. Theo-

rem 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 [26] for machine learning states that for supervised learning problems, the expected misclassification rate, summed over all possible datasets, 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. 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 ℓ, the performance of any community detection method f, summed over all problems $\{g, \mathcal{T}\}$, is identical

$$\sum_{g,\mathcal{T}} \ell\big(\mathcal{T}, f\left(g(\mathcal{T})\right)\big) = \Lambda(\ell) \quad \forall f \ , \tag{A1}$$

where $\Lambda(\ell)$ depends on the particular error function ℓ 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 ℓ. 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. [26] 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. [26]. 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 γ be the true or target relationship between X and Y, i.e., γ is the conditional distribution of Y, given X. The points in the space X need not be distributed uniformly either, so we also specify σ, 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 $\mathcal{G}$, and the output $y \in Y$ is the true partition into communities $\mathcal{T}$ 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 γ 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 γ, i.e., $P(h|d, f) = P(h|d)$. If the algorithm performs well the hypothesis h will have high correspondence with the true relationship γ. 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 γ and σ is important to establish, but for unsupervised tasks, the set d is empty so it is trivially independent of γ and σ.

To better understand the EBF for community detection, an example is helpful. Consider the problem of finding two planted communities in a network $\mathcal{G}$. The true relationship γ between the network and its partition is hidden. Given only $\mathcal{G}$—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 γ. (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 γ 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 γ with possibly limited accuracy, and therefore its community assignments may or may not be highly correlated with the true assignments $\mathcal{T} \in Y$. 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 γ, 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 [15, 31] or disassortative [32] community structures, while other are more flexible and can find mixtures of assortative, disassortative, and core-periphery structures [13, 14, 21, 33] and allow for nodes to belong to multiple communities [30, 31].

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 γ 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

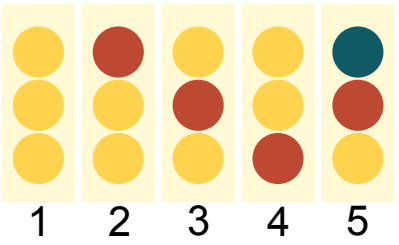

FIG. S1. The five distinct ways to partition three nodes. Normalized mutual information and adjusted mutual information between each pair of partitions are presented in Tables S2 and S3, respectively.

example, in the context of supervised learning, choosing the loss function ℓ 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 $p_i = |u_i|/N$, $i = 1 \dots K_u$. 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, $j = 1 \dots K_v$ 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

$$\mathrm{NMI}(u, v) = \frac{I(u, v)}{\sqrt{H(u)H(v)}} \ . \tag{A2}$$

Other normalizing factors in the denominator are possible, including $\frac{1}{2}[H(u)+H(v)]$ and $\max\{H(u), H(v)\}$; see [46]. 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 $Y = \{y \mid \|y\| \leq 1,\ y \in \mathcal{R}^2\}$. 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. S1 shows all five possible partitions of three vertices, and Table S2 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 [46]. Table S3 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 $\mathcal{B}_N$—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, $1/\mathcal{B}_N \approx 0$. 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.

TABLE S2. Normalized mutual information for partitions in Fig. S1

	Partition 2				
Partition 1	**1**	**2**	**3**	**4**	**5**
1	1	0	0	0	0
2	0	1	0.27	0.27	0.76
3	0	0.27	1	0.27	0.76
4	0	0.27	0.27	1	0.76
5	0	0.76	0.76	0.76	1
$\mathbb{E}$[NMI]	0.20	0.46	0.46	0.46	0.66

TABLE S3. Adjusted mutual information for partitions in Fig. S1

	Partition 2				
Partition 1	**1**	**2**	**3**	**4**	**5**
1	1	0	0	0	0
2	0	1	-0.5	-0.5	0
3	0	-0.5	1	-0.5	0
4	0	-0.5	-0.5	1	0
5	0	0	0	0	1
$\mathbb{E}$[AMI]	0.20	0	0	0	0.20

c. *Theorem and corollary*

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 corollary 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 $\mathcal{B}_N^{-1}$.

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

$$L(u) = \sum_{v \in \Omega} \mathrm{AMI}(u, v) \tag{A3}$$

is independent of u, where Ω 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:

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

where I is mutual information and H is entropy [46]. 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 every 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 ϕ be a subset of all partitions Ω 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 Ω may be subdivided into non-overlapping subsets $\{\phi_i\}$, such that $\cup_i \phi_i = \Omega$ and $\phi_i \cap \phi_j = \varnothing$ for any $i \neq j$. (For example, in Fig. S1, partition 1 belongs to ϕ_1, partitions 2, 3, and 4 belong to ϕ_2, and partition 5 belongs to ϕ_3.) Let the particular subset ϕ_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 are in a single group and the N-partition in which each object is in its own group. These will be denoted by $\bar{1}$ and $\bar{N}$ respectively. Note that $\bar{1} = \phi(\bar{1})$ so that $|\phi(\bar{1})| = 1$, and that $\phi(\bar{N})$ is equivalent to the set of all possible relabelings of the N objects, so that $|\phi(\bar{N})| = N!$. Because there is only one element of $\phi(\bar{1})$, it is necessarily true that $I(\bar{1},\bar{1}) = E[I(\bar{1},\bar{1})] = H(\bar{1})$. Thus, for this special case, the numerator and denominator of AMI are identical, and $\mathrm{AMI}(\bar{1},\bar{1}) = 1$. Similarly, because the set $\phi(\bar{N})$ contains every possible permutation of the labels of the objects, yet all partitions are identical, $I(\bar{N},\bar{N}) = E[I(\bar{N},\bar{N})] = H(\bar{N})$, and so $\mathrm{AMI}(\bar{N},\bar{N}) = 1$.

In order to prove Eq. (A3), we will show that $L(u) = 0$ for all u except $\bar{1}$ and $\bar{N}$ by demonstrating that the numerator of the definition of AMI is 0, specifically,

$$\sum_{v\in\Omega} [I(u,v) - E[I(u,v)]] = 0 \quad \forall\ u \neq \bar{1} \text{ or } \bar{N} \ . \tag{A4}$$

In fact, we will show that Eq. (A4) holds by breaking the entire sum over all partitions Ω into sums over each of its disjoint subsets $\{\phi_i\}$, and proving that

$$\sum_{v'\in\phi(v)} [I(u,v') - E[I(u,v')]] = 0$$
$$\forall\ u \text{ and } \forall\ v \text{ except } u = v = \bar{1} \text{ or } u = v = \bar{N} \ . \tag{A5}$$

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. (A5). Recall that the expectation is taken over all pairs of members of the subsets $\phi(u)$ and $\phi(v)$, respectively,

$$E[I(u,v)] = \frac{1}{|\phi(u)||\phi(v)|} \sum_{u'\in\phi(u)} \sum_{v'\in\phi(v)} I(u',v') \ . \tag{A6}$$

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 ζ for any pair of partitions drawn from $\phi(u)$ and $\phi(v)$,

$$E[I(u,v)] = \zeta \quad \forall\ u \in \phi(u) \text{ and } \forall\ v \in \phi(v) \ . \tag{A7}$$

Note then that we may rewrite the sum over expectations in Eq. (A5) 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. (A5) is also equal to $|\phi(v)|\ \zeta$,

$$\sum_{v'\in\phi(v)} I(u,v') = |\phi(v)|\ \zeta \ . \tag{A8}$$

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. (A8) 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 reindexings) of the N objects being partitioned, it must be true that,

$$\sum_{v'\in\phi(v)} I(u_1,v') = \sum_{v'\in\phi(v)} I(u_2,v') \text{ whenever } u_1 \in \phi(u_2) \ . \tag{A9}$$

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. (A8) is true, meaning that the sum over the two terms in Eq. (A5) is zero, independent of u. This first implies that the AMI between any boundary partition and any interior partition is 0, $\mathrm{AMI}(u,\bar{1}) = 0$ for any $u \neq \bar{1}$ and $\mathrm{AMI}(u,\bar{N}) = 0$ for any $u \neq \bar{N}$. This, in turn, implies Eq. (A4) is true. This completes the proof of the first statement, that Eq. (A3) 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 $\mathrm{AMI}(\bar{1},\bar{1}) = 1$, $\mathrm{AMI}(\bar{N},\bar{N}) = 1$, and $\mathrm{AMI}(u,\bar{1}) = 0$ for any $u \neq \bar{1}$ and $\mathrm{AMI}(u,\bar{N}) = 0$ for any $u \neq \bar{N}$. Therefore,

$$L(\bar{1}) = \sum_{v\in\Omega} \mathrm{AMI}(\bar{1},v) = \mathcal{B}_N^{-1} \ ,$$
$$L(\bar{N}) = \sum_{v\in\Omega} \mathrm{AMI}(\bar{N},v) = \mathcal{B}_N^{-1} \ , \tag{A10}$$

completing the proof of the second statement: including the boundary points, AMI is homogenous within an additive constant $\mathcal{B}_N^{-1}$. □

Theorem 2 (Wolpert 1996): For homogeneous loss ℓ, the uniform average over all γ of $P(c|\gamma,d)$ equals $\Lambda(c)/r$.

Proof: See Ref. [26].

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 γ of $P(c|\gamma)$ equals $\Lambda(c)/r$.

Proof: Lemma 1 proves that adjusted mutual information is homogenous and applying Theorem 2 with $d = \varnothing$ 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 [32]—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 correct; in the other cases, the same model assumptions that improved performance on some problems will diminish it for others. 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 [43, 44].

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.*

Appendix B: Blockmodel Entropy Significance Test

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?* Cypher: *Ignorance is bliss.*

This Supplemental Text is divided into four subsections providing additional details on the Blockmodel Entropy Significance Test.

- Subsection I describes maximum likelihood parameter estimation for the SBM (I.a) and degree-corrected SBM (I.b).
- Subsection II describes rapid computation of the entropy $H(G;\mathcal{M})$ for the SBM.
- Subsection III demonstrates the mathematical link between our formulation of the SBM entropy and the SBM log likelihood which has been derived elsewhere [21, 47].
- Subsection IV provides additional examples of results of the Blockmodel Entropy Significance Test using multiple different network data and metadata sets (see Supplemental 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 Supplemental Text.

TABLE S4. Notation used in this Supplemental Text

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

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 π_i.

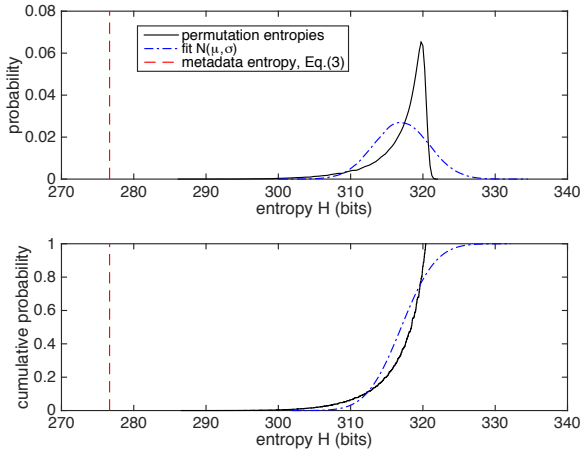

FIG. S2. **Distributions of permuted partition entropies are negatively skewed.** Probability density functions (top) and cumulative distribution functions (bottom) are shown for the entropies of partitions of the Karate Club network and its faction metadata. The red broken line indicates the point entropy of the metadata partition while the black solid line shows the distribution of entropies for 10^4 independent permutations of the metadata partition. Note that these permutation entropies are far from normal; a normal distribution with equivalent mean μ and variance σ^2 is shown in blue for contrast.

In the SBM, the probability of a link existing between any two nodes i and j depends only on the group assignments π_i and π_j. This means that the entire model can be parameterized by a $K\times K$ matrix of block-to-block edge probabilities, ω. Accordingly, let ω 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_rn_s possible links, each of which has the same probability of existence, ω_{rs}. This implies that the existence of the n_rn_s edges between groups r and s will be determined by n_rn_s independent Bernoulli trials, each with parameter ω_{rs}.

We must now estimate the value of ω_{rs} for a network G whose nodes have been divided according to their assignments in partition π. Of course, any ω 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 ω to be those that maximize the likelihood of observing G. Specifically, observe that of the n_rn_s Bernoulli trials, there are m_{rs} actual edges in the graph, i.e., m_{rs} trial successes. Therefore, the maximum likelihood estimate of ω_{rs} is simply $\hat{\omega}_{rs}=m_{rs}/n_rn_s$. Thus, $\hat{p}_{ij}=\hat{\omega}_{\pi_i\pi_j}$.

b. *Poisson degree-corrected SBM parameters*

In the degree-corrected Poisson SBM [21], it is still assumed that each link exists independently of the others,

with some specified probability given by a block connectivity matrix ω. 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 θ_i so that $q_{ij} = \theta_i\theta_j e_{\pi_i\pi_j}$, where e_{rs} is the $K \times K$ block structure matrix, controlling the numbers of links between groups, similar in principle to ω_{rs} above. The new parameters, θ_i, properly chosen [21], can be used to specify the expected degree of each node.

As above, since we are given a network G and a fixed partition π, we must estimate the entries of e, as well as the values of θ. The parameters can again be chosen to maximize the likelihood of observing G, which are derived in [21] but we do not derive here. First, $\hat{e}_{rs} = m_{rs}$, where $m_{rs} = \sum_{ij} A_{ij}\delta_{r,\pi_i}\delta_{s,\pi_j}$ is the number of links between groups r and s. Then, $\hat{\theta}_i = k_i/\kappa_{\pi_i}$, where κ_r is the number of degrees connecting to group r, $\kappa_r = \sum_s m_{rs}$. Thus, $\hat{q}_{ij} = k_i k_j m_{\pi_i\pi_j}/\kappa_{\pi_i}\kappa_{\pi_j}$. We note that this maximum likelihood estimate is only valid in the regime that $k_i k_j m_{\pi_i\pi_j} \ll \kappa_{\pi_i}\kappa_{\pi_j}$.

2. Rapidly computing entropy for Bernoulli SBMs

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

$$h(p_{ij}) \equiv -p_{ij}\log_2 p_{ij} - (1-p_{ij})\log_2(1-p_{ij}). \quad \text{(B1)}$$

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

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

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

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

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

$$H_{\text{SBM}}(G) = \dots - \frac{1}{2}\left[\sum_{rs} m_{rs}\log\hat{\omega}_{rs} + (n_r n_s - m_{rs})\log(1-\hat{\omega}_{rs})\right] + \mathcal{O}(n^{-1}). \quad \text{(B4)}$$

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 $\mathcal{L}$.

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, mixed-membership models, and other generative models with computable log likelihoods.

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

$$(1-x)\ln(1-x) = -x + \sum_{\ell=2}^{\infty}\frac{x^\ell}{\ell(\ell-1)}, \quad \text{(B5)}$$

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

$$H_{\text{SBM}}(G) \approx -\frac{1}{2}\sum_{rs}\left[m_{rs}\ln\left(\frac{m_{rs}}{n_r n_s}\right) - m_{rs}\dots + n_r n_s\sum_{\ell=2}^{\infty}\frac{1}{\ell(\ell-1)}\left(\frac{m_{rs}}{n_r n_s}\right)^\ell\right]. \quad \text{(B6)}$$

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

$$H(G) \approx |E| - \frac{1}{2}\sum_{rs}\left[m_{rs}\ln\left(\frac{m_{rs}}{n_r n_s}\right)\dots + n_r n_s\sum_{\ell=2}^{\infty}\frac{1}{\ell(\ell-1)}\left(\frac{m_{rs}}{n_r n_s}\right)^\ell\right]. \quad \text{(B7)}$$

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

$$H_{\rm SBM}(G) \approx |E| - \frac{1}{2}\sum_{rs} m_{rs} \ln\left(\frac{m_{rs}}{n_r n_s}\right). \tag{B8}$$

Here we derived Eq. (B7) and Eq. (B8) 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 Ω 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 $\mathcal{L} = -H$ [47].

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,

$$p\text{-value} = \Pr\left[\mathcal{L}(G;\tilde{\pi}) \geq \mathcal{L}(G;\mathcal{M})\right]. \tag{B9}$$

The Bernoulli SBM entropy Eq. (B4) or its approximation for sparse networks Eq. (B8) are convenient because they are fast to compute—one need only to count links between groups, sizes of groups, and compute $\mathcal{O}(K^2)$ terms. By contrast, Eq. (B2), which is exact, requires $\mathcal{O}(N^2)$ 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 Supplemental Text, we employ the equations above to apply the BESTest using Bernoulli SBM and degree-corrected SBM, as well as Poisson SBM and degree-corrected SBM.

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. [34].

4. Generation of synthetic networks for blockmodel entropy significance test

The tests described in the main text, and detailed in this Supplemental 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 ($\mathcal{T}_i = r$ or $\mathcal{T}_i = s$) with equal probability. We then generated a network with a given community strength $\epsilon = \omega_{rs}/\omega_{rr}$ such that low values of ϵ generate strongly assortative communities with few connecting edges between them and as ϵ grows, the generated communities become weaker, producing a random graph with no communities when $\epsilon = 1$. For each node i, with probability ℓ we assigned its metadata label to be its community label ($\mathcal{M}_i = \mathcal{T}_i$), otherwise we assigned it a uniformly random label. Thus, as ℓ increases from 0 to 1 the metadata labels correlate more with the planted communities.

TABLE S5. Lazega Lawyers: BESTest p-values

	Attribute				
Network	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	$< 10^{-6}$	$< 10^{-3}$	$< 10^{-3}$	$< 10^{-3}$	0.034
Cowork	$< 10^{-6}$	0.982	0.396	$< 10^{-6}$	0.805
Advice	$< 10^{-6}$	0.033	0.147	$< 10^{-6}$	0.115
	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

5. V. Additional applications of the BESTest

We now present and discuss the results of applying the BESTest to the Lazega Lawyers and Malaria data sets (see Supplemental 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 S5.

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 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 Supplemental Text D.

The *parasite origin* results are shown in Table S6, 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 [35]. 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 S7 and S8 the BESTest indeed finds that this is the case, with a handful of exceptions, again confirming previous results that used less sophisticated techniques [35].

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.*

TABLE S6. Malaria: BESTest p-values for parasite origin metadata

	Model			
Network	SBM	DCSBM	Poi. SBM	Poi. DCSBM
Malaria 1	0.566	0.066	0.606	0.086
Malaria 2	0.064	0.126	0.066	0.143
Malaria 3	0.536	0.415	0.532	0.501
Malaria 4	0.588	0.570	0.604	0.644
Malaria 5	0.382	0.097	0.369	0.087
Malaria 6	0.275	0.817	0.293	0.751
Malaria 7	0.020	0.437	0.019	0.501
Malaria 8	0.464	0.143	0.468	0.172
Malaria 9	0.115	0.104	0.108	0.200

TABLE S7. Malaria: BESTest p-values for CP group metadata

	Model			
Network	SBM	DCSBM	Poi. SBM	Poi. DCSBM
Malaria 1	$< 10^{-5}$	0.002	$< 10^{-5}$	$< 10^{-5}$
Malaria 2	$< 10^{-5}$	0.042	$< 10^{-5}$	$< 10^{-5}$
Malaria 3	$< 10^{-5}$	0.237	$< 10^{-5}$	$< 10^{-5}$
Malaria 4	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$
Malaria 5	$< 10^{-5}$	0.005	$< 10^{-5}$	$< 10^{-5}$
Malaria 6	$< 10^{-5}$	0.002	$< 10^{-5}$	$< 10^{-5}$
Malaria 7	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$
Malaria 8	$< 10^{-5}$	0.002	$< 10^{-5}$	$< 10^{-5}$
Malaria 9	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$

TABLE S8. Malaria: BESTest p-values for UPS metadata

	Model			
Network	SBM	DCSBM	Poi. SBM	Poi. DCSBM
Malaria 1	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$	$< 10^{-5}$
Malaria 2	$< 10^{-5}$	0.100	$< 10^{-5}$	$< 10^{-5}$
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	$< 10^{-5}$	0.007	$< 10^{-5}$	$< 10^{-5}$
Malaria 9	$< 10^{-5}$	$< 10^{-4}$	$< 10^{-5}$	$< 10^{-5}$

Appendix C: 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.*

This Supplemental 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 Supplemental Text.

TABLE S9. Notation used in this Supplemental Text

Variable	Definition
$\mathcal{G}$	a network, $\mathcal{G} = (V, E)$
N	the number of nodes $\lvert V\rvert$
e_{ij}	the number of edges between nodes i and j, $e_{ij} \in \{0, 1\}$
k_i	the degree of node i.
ω_{rs}	the probability of an edge between nodes in groups r and s
π	a partition of nodes into groups
M	a set of metadata labels
C	an inferred optimal community assignment
z	neo-state indicator variable, $z_i \in \{b, r\}$
$\mathcal{L}_X$	log likelihood L of model X
q	the number of free nodes, $q = \sum_i \delta_{z_i,r}$
$\delta_{a,b}$	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 θ that a node will be free (red state). We can then write down the likelihood L_{neo} of a network $\mathcal{G}$ given a community assignment π under the neoSBM as:

$$L_{\text{neo}}(\mathcal{G}; \pi, z) = \prod_{ij} \omega_{\pi_i\pi_j}^{e_{ij}} (1 - \omega_{\pi_i\pi_j})^{(1-e_{ij})} \prod_i \theta^{\delta_{z_i,r}} (1-\theta)^{\delta z_i,b} \, . \tag{C1}$$

The first product in Eq. (C1) corresponds to the standard SBM likelihood L_{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. (C1) directly, instead we work with the more practical log likelihood,

$$\begin{aligned}\mathcal{L}_{\text{neo}}(\mathcal{G}; \pi, z) = &\sum_{ij} e_{ij} \log \omega_{\pi_i\pi_j} + (1 - e_{ij}) \log(1 - \omega_{\pi_i\pi_j}) \\ &+ \sum_i \delta_{z_i,r} \log\theta + \delta_{z_i,b} \log(1-\theta) \ , \end{aligned} \tag{C2}$$

since maximizing Eq. (C1) is equivalent to maximizing Eq. (C2). We can then rearrange the second sum $\log P(z = r|\theta)$, to give:

$$\begin{aligned}\log P(z = r|\theta) &= \sum_i \delta_{z_i,r} \left(\log \frac{\theta}{1-\theta} \right) + N \log(1-\theta) \\ &= q f(\theta) + N \log(1-\theta) \ , \end{aligned} \tag{C3}$$

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,

$$\mathcal{L}_{\text{neo}}(\mathcal{G}; \pi, z) = \mathcal{L}_{\text{sbm}}(\mathcal{G}; \pi) + q f(\theta) \ . \tag{C4}$$

Unconstrained optmization of $\mathcal{L}_{\text{sbm}}$ yields the SBM optimal communities C,

$$C = \arg\max_{\pi} \mathcal{L}_{\text{sbm}}(\mathcal{G}; \pi) \ , \tag{C5}$$

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 $\mathcal{L}_{\text{sbm}}(\mathcal{G}; M) \leq \mathcal{L}_{\text{sbm}}(\mathcal{G}; C)$, 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

$$\hat{q} = \sum_i 1 - \delta_{M_i,C_i} \ , \tag{C6}$$

for which the label permutations of M and C are maximally aligned. Whenever $q > \hat{q}$ there will be no further

improvement in $\mathcal{L}_{\text{sbm}}$. 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 $f(\theta)$, as the cost of freeing a node because the log likelihood $\mathcal{L}_{\text{neo}}$ will incur a penalty for setting each $z_i = r$. Maximizing $\mathcal{L}_{\text{neo}}$ is therefore a trade-off between freeing nodes to maximize $\mathcal{L}_{\text{sbm}}$ 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 $\mathcal{L}_{\text{neo}}(\mathcal{G}; \pi, z)$ will be maximized when $q = 0$ unless $\theta \geq 0.5$. However, when $\mathcal{L}_{\text{sbm}}(\mathcal{G}; M) < \mathcal{L}_{\text{sbm}}(\mathcal{G}; C)$, q can be greater than 0 if the resulting partition π provides a sufficient increase in log likelihood. Specifically, if

$$\mathcal{L}_{\text{sbm}}(\mathcal{G}; \pi) - \mathcal{L}_{\text{sbm}}(\mathcal{G}; M) > qf(\theta) \ , \tag{C7}$$

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 Supplemental 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 [48]. However, to infer the values of b_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:

$$a = \min\{\Delta L_{\text{neo}}, 1\} \ . \tag{C8}$$

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, $b_i = r$ for all i.

2. 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 S3 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. S3) and three core groups ({1,3},{6},{8} in Fig. S3). The partition inferred by the SBM in the absence of the neoSBM's likelihood penalty corresponds to the assortative group structure.

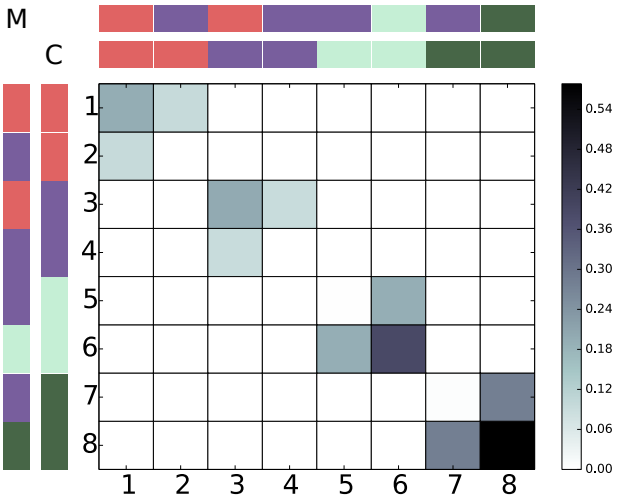

FIG. S3. The block interaction matrix used to generate synthetic networks. The external colored rows and columns indicate the partition used as metadata (M) and the maximum likelihood partition under the SBM (C).

3. 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 Supplemental 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:

$$\mathcal{L}_{\text{dcneo}}(\mathcal{G}; \pi, z) = \mathcal{L}_{\text{dcsbm}}(\mathcal{G}; \pi) + qf(\theta) \ , \tag{C9}$$

where $qf(\theta) = q \log P(z = r|\theta) + N \log(1 - \theta)$ as before. We present results from this model in subsection IV of this Supplemental 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 $\mathcal{U}_{neoX}$

$$\mathcal{U}_{neoX} = \mathcal{U}_X + qf(\theta) \ , \tag{C10}$$

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

4. 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 datasets are presented in Supplementary Text D.

a. 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 [22, 35]. 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 S4 (locus one) and S5 (locus six) show likelihood surfaces, block density diagrams, and the neoSBM's outputs q (free nodes) and SBM log 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 θ at which the neoSBM switches from the metadata partition to the first intermediate local optimum. These transitions $1 \to 2$ 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.

b. 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. S6 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 θ is large enough we see a discontinuous jump as it switches to the globally optimal high-degree/low-degree partition.

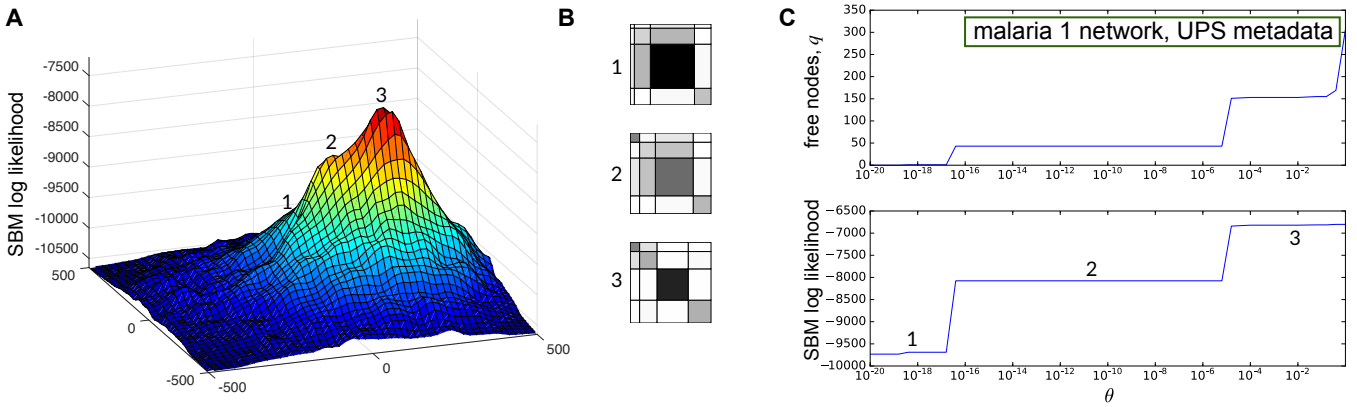

FIG. S4. Results of the neoSBM on the malaria *var* gene network at locus one ("malaria 1") using UPS metadata. (A) The SBM likelihood surface shows two peaks, one subtle 2 and one prominent 3, corresponding to a locally optimal partition near the metadata and the globally optimal partition, respectively. 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 ($1 \to 2$) and then from that partition to the global optimum ($2 \to 3$).

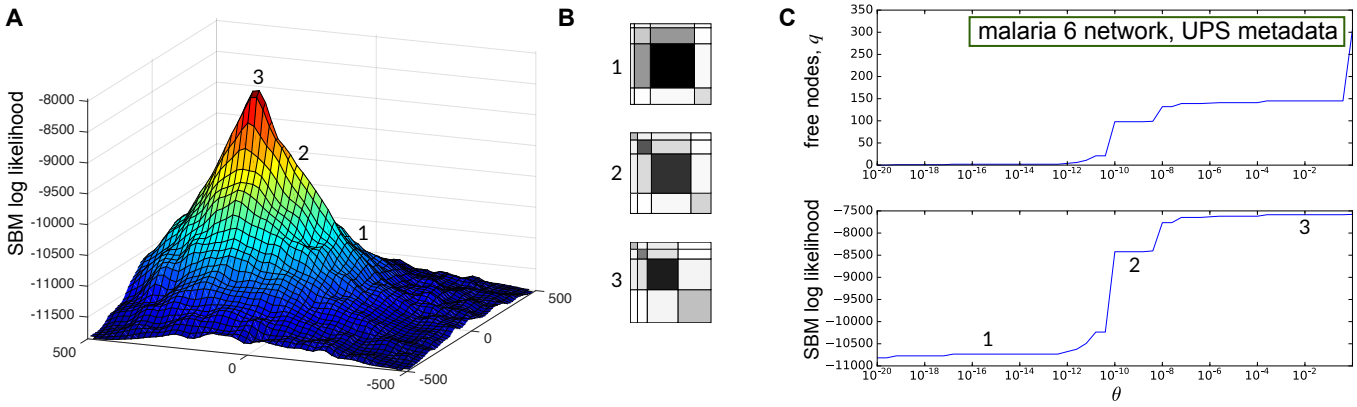

FIG. S5. Results of the neoSBM on the malaria *var* gene network at locus six ("malaria 6") using UPS metadata. (A) The SBM likelihood surface shows one prominent peak at the globally optimal partition. (B) Block density diagrams depict community structure for metadata and locally optimal partitions 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 ($1 \to 2$), from that partition to another the global optimum ($2 \to 3$).

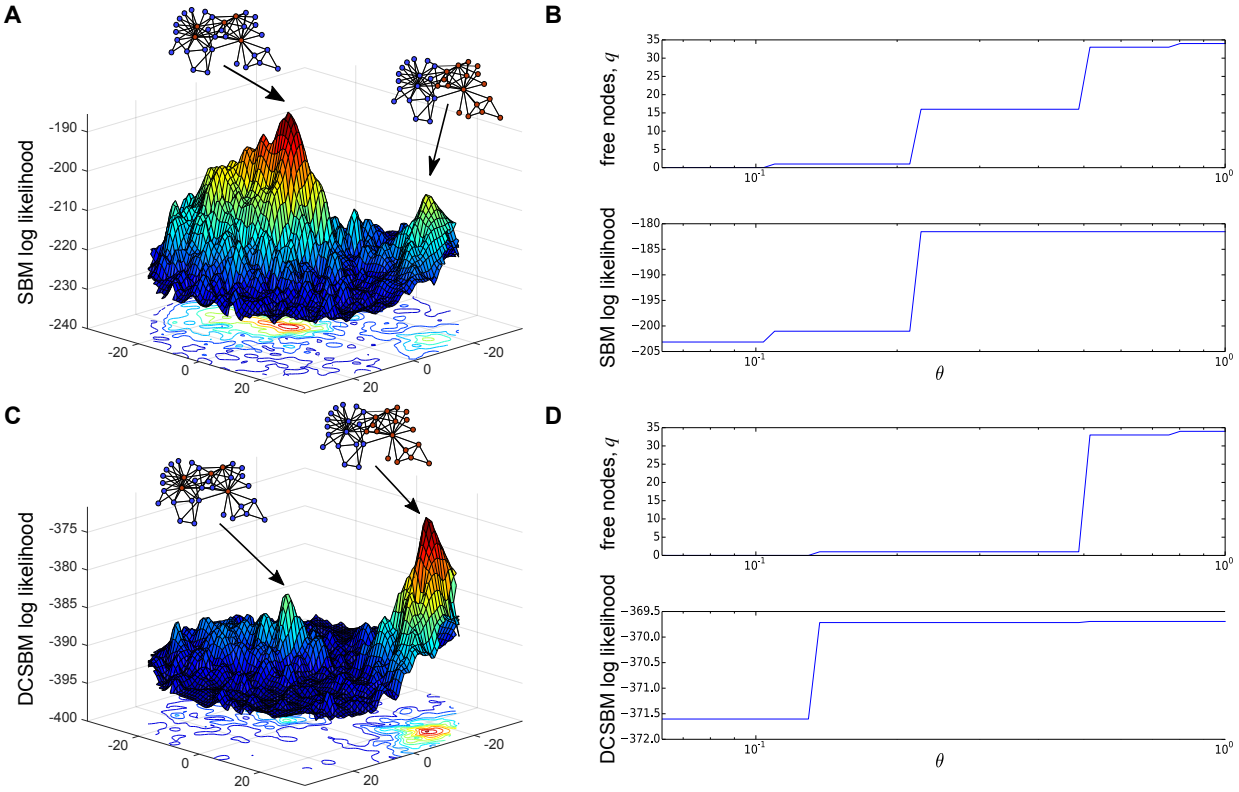

FIG. S6. The results of the neoSBM and the degree-corrected neoSBM on the karate club network. The SBM and DCSBM log likelihood surfaces (A and C respectively) show distinct two peaks that correspond to the same two partitions of the network: the two social factions and the leader-follower partition. When we use the faction partition as metadata, we from the output (B and D) that both models change a single node in order to reach the locally optimal partition. For the neoDCSBM (D), this is the global optimum and no further change is observed. For the neoSBM, the leader-follower partition is globally optimal, so once theta is large enough we see the model jump to this partition.

Appendix D: Datasets

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 [36]. 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 datasets, but they were not primarily analyzed as complex networks.

2. Malaria *var* gene networks

The Malaria dataset consists of 307 *var* gene sequences from the malaria parasite *P. falciparum* [35]. 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 strain 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 [12]. 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.

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.*
