Community detection in networks: Modularity optimization and maximum likelihood are equivalent

M. E. J. Newman
Department of Physics and Center for the Study of Complex Systems, University of Michigan, Ann Arbor, MI 48109

We demonstrate an exact equivalence between two widely used methods of community detection in networks, the method of modularity maximization in its generalized form which incorporates a resolution parameter controlling the size of the communities discovered, and the method of maximum likelihood applied to the special case of the stochastic block model known as the planted partition model, in which all communities in a network are assumed to have statistically similar properties. Among other things, this equivalence provides a mathematically principled derivation of the modularity function, clarifies the conditions and assumptions of its use, and gives an explicit formula for the optimal value of the resolution parameter.

I. INTRODUCTION

Community detection, sometimes called network clustering, is the division of the nodes of an observed network into groups such that connections are dense within groups but sparser between them [1–3]. Not all networks support such divisions, but many do, and the existence of good divisions is often taken as a hint of underlying semantic structure or possible mechanisms of network formation, making community detection a useful tool for interpreting network data.

The development of methods or algorithms to perform community detection on empirical networks has been a popular pursuit among researchers in physics, mathematics, statistics, and computer science—a tremendous number of such algorithms have been published in the last decade or so [1–4]. In this paper we study two of the most popular and widely used methods for community detection in simple undirected networks, the method of modularity maximization and the method of maximum likelihood as applied to the stochastic block model. Building on previous work by ourselves and others [5–7], we show that, different though they at first appear, these two methods are in fact exactly equivalent, for appropriate choices of models and parameters, which we specify. This sheds light in particular on the modularity maximization method, which is generally motivated with heuristic arguments [8–10] (although there are some rigorous results [11, 12]). Our results provide a rigorous derivation for the modularity and demonstrate that modularity maximization is optimal under appropriate conditions, but also highlight the method's limitations. In particular, we show that modularity maximization effectively assumes that communities in a network are statistically similar, and it is not guaranteed to give good results for networks where this is not true.

II. COMMUNITY DETECTION

We begin by describing the two methods of community detection that we study, in their most widely accepted forms, beginning with the method of modularity maximization.

A. Modularity maximization

Modularity maximization is perhaps the most widely used method for community detection for networks. It operates by defining a benefit function, called the modularity, that measures the quality of divisions of a network into communities. One optimizes this benefit function over possible divisions of the network of interest to find the one that gives the highest score, taking this to be the definitive division of the network. Since the number of possible divisions of a network is exponentially large, we normally cannot perform the optimization exhaustively, so we turn instead to approximate optimization methods, of which many have been tried, including greedy algorithms [9, 13], extremal optimization [14], spectral relaxation [15], genetic algorithms [16], simulated annealing [17, 18], and belief propagation [7]. The popular Louvain algorithm for community detection [19], which is built into a number of network analysis software packages, uses a multiscale modularity optimization scheme and is one of the fastest community detection methods in practice.

The definition of the modularity function is straightforward [8]. We desire a benefit function which, given a network and a candidate division of that network into groups, returns a score that is larger if the division is a "good" one and smaller if it is "bad." The heuristic notion used to define the modularity is that a good division is one that places most of the edges of a network within groups and only a few of them between groups.

Let us represent our network by its adjacency matrix. For an undirected network of n nodes the adjacency matrix $\mathbf{A}$ is the real symmetric $n \times n$ matrix with elements $A_{ij} = 1$ if there is an edge between nodes i and j and 0 otherwise. Further, let us consider a division of the network into q nonoverlapping groups, numbered (in any order) from 1 to q, and let us denote by g_i the number of the group to which node i is assigned. Thus the complete vector $\mathbf{g}$ of group assignments specifies the division of the network. Then the number of edges that fall within groups, for this particular division, is equal to $\frac{1}{2}\sum_{ij} A_{ij}\delta_{g_i g_j}$, where δ_{ij} is the Kronecker delta and the leading factor of a half prevents double counting of edges.

The number of in-group edges alone, however, is not a good measure of the quality of a division, since it can be trivially maximized by putting all the nodes in one of the q groups and none in any of the others. This would put 100% of edges inside groups but clearly doesn't constitute a useful division of the network. Instead, therefore, modularity measures not just the number of edges within groups but the difference between that number and the expected number of such edges, were edges placed at random within the network.

Suppose we take our observed network and randomize the positions of its edges. We keep the total number of edges the same but we reposition them between the nodes at random, in a manner to be determined shortly. And suppose that, following this randomization, the probability that nodes i and j are connected by an edge is P_{ij}. Then the expected number of edges within groups, post-randomization, is $\frac{1}{2}\sum_{ij} P_{ij}\delta_{g_i g_j}$. The modularity is then proportional to the actual minus expected number of edges thus:

$$\begin{aligned} Q &= \frac{1}{m}\biggl(\tfrac{1}{2}\sum_{ij} A_{ij}\delta_{g_i g_j} - \tfrac{1}{2}\sum_{ij} P_{ij}\delta_{g_i g_j}\biggr) \\ &= \frac{1}{2m}\sum_{ij}\bigl(A_{ij} - P_{ij}\bigr)\delta_{g_i g_j} \end{aligned} \quad (1)$$

where m is the total number of edges in the network and is included here by convention only—it makes Q equal to a *fraction* of edges rather than an absolute number, which makes modularity values more easily comparable between networks of different sizes. For the purposes of maximizing the modularity, which is our main concern here, the factor of m makes no difference at all. The position of the maximum does not depend on overall constant factors.

Note that if we now put all nodes in the same group, then $\delta_{g_i g_j} = 1$ for all i, j and

$$Q = \frac{1}{2m}\sum_{ij}\bigl(A_{ij} - P_{ij}\bigr) = 0, \quad (2)$$

since, as we have said, the number of edges in the network is held constant during randomization, and hence $\sum_{ij} P_{ij} = \sum_{ij} A_{ij} = 2m$. Thus we no longer get a high modularity score for putting all nodes in a single group together. The maximum of modularity occurs for some other (nontrivial) division of the nodes, which we take to be the best division of the network. This is the method of modularity maximization.

It remains to determine what P_{ij} is. The value depends on the particular scheme we use to randomize the positions of the edges. The simplest scheme would be just to reposition the edges uniformly at random, every position having the same probability as every other. For a network of n nodes there are $\binom{n}{2}$ places to put an edge, and hence the probability of filling any of them with one of the m edges is

$$P_{ij} = \frac{m}{\binom{n}{2}}. \quad (3)$$

(Technically this is the expected number of edges not the probability, but normally $m \ll \binom{n}{2}$ so that probability and expected number are essentially the same.)

In practice, however, this choice does not work very well because it fails to respect the degrees of the nodes in the network. The probabilities of connections between nodes depend strongly on the total number of connections nodes have—their degrees—with nodes of high degree being much more likely to be connected than nodes of low degree. For reasons that will become clear in this paper, it is important to include this effect in the definition of modularity if things are to work correctly.

Instead, therefore, we consider a constrained randomization of the edges in the network in which we preserve the degree of every node, but otherwise position the edges at random. This kind of randomization is well known in the study of networks: it gives rise to the random graph ensemble known as the configuration model [20, 21]. After randomization, the probability of connection between two nodes is equal to

$$P_{ij} = \frac{k_i k_j}{2m}, \quad (4)$$

where $k_i = \sum_j A_{ij}$ is the degree of node i and m is once again the number of edges in the network. (Again, this is technically the expected number of edges, but the probability and expected number are essentially the same.)

This is the choice that is most commonly used in the definition of the modularity. With this choice the modularity is given by

$$Q = \frac{1}{2m}\sum_{ij}\biggl(A_{ij} - \frac{k_i k_j}{2m}\biggr)\delta_{g_i g_j}, \quad (5)$$

which is the form in which it is most commonly written.

There is a further twist, however, because even this definition does not always work well. As shown by Fortunato and Barthélémy [22], community detection by modularity maximization using the definition of (5), while it works in many situations, has one specific shortcoming: it is unable to find community structure in large networks with many small communities. In particular, if the number of communities in a network is greater than about $\sqrt{2m}$, then the maximum modularity will not correspond to the correct division. The maximum will instead tend to combine communities into larger groups and fail to resolve the smallest divisions in the network.

To address this problem, Reichardt and Bornholdt [10] proposed a generalized modularity function

$$Q(\gamma) = \frac{1}{2m}\sum_{ij}\biggl(A_{ij} - \gamma\frac{k_i k_j}{2m}\biggr)\delta_{g_i g_j}. \quad (6)$$

When the parameter $\gamma = 1$, this is the same as the traditional modularity of Eq. (5), but other choices allow us to vary the relative weight given to the observed and randomized edge terms. If one places more weight on the observed edge term (by making γ smaller), the maximum modularity division favors, and the method therefore tends to find, larger communities. If one places more weight on the randomized edge term (larger γ), the method finds smaller communities. There has not previously been any fundamental theory dictating what value of γ one should use, but this is one of the questions on which we will shed light in this paper.

B. Statistical inference

The other method of community detection we consider is the method of statistical inference, as applied to the stochastic block model and its variants. With this method, one fits a generative model of a network to observed network data, and the parameters of the fit tell us about the structure of the network in much the same way that fitting a straight line through a set of data points tells us about their slope.

The model most commonly used in this context is the so-called stochastic block model, which is a random graph model of a network with community structure [12, 23, 24]. One takes some number n of nodes, initially without any edges, and divides them into q groups in some way, with g_i being the group to which node i is assigned, as previously. Then one places edges between nodes independently at random, with the probability, which we denote ω_{rs}, of an edge between a particular pair of nodes depending on the groups r and s to which the nodes belong. Thus there is a symmetric $q \times q$ matrix of parameters ω_{rs} which determine the probabilities of edges within and between every pair of groups. If the diagonal elements ω_{rr} of this matrix are larger than the off-diagonal elements, then networks generated by the model have a higher probability of edges within groups than between them and hence have traditional community structure.

In fact, the stochastic block model is often studied in a slightly different formulation in which one places not just a single edge between any pair of nodes but a Poisson distributed number of edges with mean ω_{rs}. Thus ω_{rs} is the expected number, rather than the probability, of edges between nodes in groups r and s, and the networks generated by the model can in principle have multiedges, meaning there can be more than one edge between the same pair of nodes. Moreover, one typically also allows the network to contain self-edges, edges that connect a node to itself, which are also Poisson distributed in number, with mean $\frac{1}{2}\omega_{rr}$ for a node in group r. (The factor of half is included solely because it makes the algebra simpler.) The inclusion of multiedges and self-edges in the model is unrealistic in the sense that most network data encountered in the real world contain neither. However, most real-world networks are also very sparse, meaning that only a tiny fraction of all possible edges that could exist actually do, and hence the values of the edge probabilities ω_{rs} are very small. In this situation, the density of multiedges and self-edges in the network will itself be small and the Poisson version of the model is virtually indistinguishable from the first (Bernoulli) version defined above. At the same time, the Poisson version is technically easier to handle than the Bernoulli version. In this paper we use the Poisson version. Similar results can be derived for the Bernoulli version, but the formulas are more complicated and the end result is little different.

The definition of the model above is in terms of its use to generate networks. As applied to community detection, however, the model is used in the "reverse" direction to infer structure by fitting it to data. In this context, one hypothesizes that an observed network, with adjacency matrix $\mathbf{A}$, was generated from the stochastic block model, and attempts to work out what values of the model parameters must have been used in the generation. The parameters in this case are the edge probabilities ω_{rs} and the group memberships g_i.

Given particular values of the parameters we can write down the probability, or likelihood, that the observed network was generated from the block model thus:

$$P(\mathbf{A}|\mathbf{\Omega},\mathbf{g}) = \prod_i \frac{(\frac{1}{2}\omega_{g_i g_i})^{A_{ii}/2}}{(\frac{1}{2}A_{ii})!} \mathrm{e}^{-\omega_{g_i g_i}/2} \prod_{i<j} \frac{\omega_{g_i g_j}^{A_{ij}}}{A_{ij}!} \mathrm{e}^{-\omega_{g_i g_j}}, \tag{7}$$

where $\mathbf{\Omega}$ denotes the complete matrix of values ω_{rs} and we have adopted the standard convention that a self-edge is represented by a diagonal adjacency matrix element $A_{ii} = 2$ (and not 1 as one might at first imagine).

The position of the maximum of this quantity with respect to $\mathbf{\Omega}$ and $\mathbf{g}$ tells us the values of the parameters most likely to have generated the observed network. Typically we are interested only in the group assignments $\mathbf{g}$, which tell us how the network divides into groups. The probabilities $\mathbf{\Omega}$ are usually uninteresting and can be discarded once the likelihood has been maximized.

Alternatively (and usually more conveniently), we can maximize the logarithm of the likelihood:

$$\begin{aligned}\log P(\mathbf{A}|\mathbf{\Omega},\mathbf{g}) &= \sum_i \bigl[\tfrac{1}{2}A_{ii}\log\bigl(\tfrac{1}{2}\omega_{g_i g_j}\bigr) - \tfrac{1}{2}\omega_{g_i g_j} - \log\bigl(\tfrac{1}{2}A_{ij}!\bigr)\bigr] \\ &\quad + \sum_{i<j}\bigl(A_{ij}\log\omega_{g_i g_j} - \omega_{g_i g_j} - \log A_{ij}!\bigr). \end{aligned} \tag{8}$$

The terms $\frac{1}{2}A_{ii}\log\frac{1}{2}$, $\log(\frac{1}{2}A_{ij}!)$, and $\log A_{ij}!$ are all independent of the parameters and do not affect the position of the maximum, so they can be ignored, and the log-likelihood then simplifies to

$$\log P(\mathbf{A}|\mathbf{\Omega},\mathbf{g}) = \tfrac{1}{2}\sum_{ij}\bigl(A_{ij}\log\omega_{g_i g_j} - \omega_{g_i g_j}\bigr). \tag{9}$$

The optimal division of the network into communities is then given by maximizing this quantity with respect to both $\mathbf{g}$ and $\mathbf{\Omega}$.

C. Degree-corrected block model

As with the modularity, however, this is not the whole story. This approach fares poorly when applied to most real-world networks because it doesn't respect the node degrees in the network. The stochastic block model as described here (in either Bernoulli or Poisson versions) generates networks that have a Poisson degree distribution, which is very different from the broad distributions seen in empirical networks. This means that, typically, the model does not fit observed networks well for any choice of parameter values. It's as if one were trying to fit a straight line through an inherently curved set of data points. Even the best fit of such a line will not be a good fit. There are no good fits when the model you are fitting is simply wrong.

The conventional solution to this problem, in the present situation, is to use a slightly different model, the degree-corrected block model [25], which can fit networks with any degree distribution. In this model the nodes are again assigned to groups g_i and edges placed independently at random between them, but now the expected number of edges between nodes i and j is $(k_i k_j/2m)\omega_{rs}$ (where as before r, s are the groups to which the nodes belong and k_i, k_j are the degrees) or a half that number for self-edges. The factor of $2m$ in the denominator is optional, but convenient since, as we have said, $k_i k_j/2m$ is the probability of an edge in the configuration model and hence, with this definition, ω_{rs} quantifies the probability of edges relative to the configuration model.

Following the same line of reasoning as before, and again neglecting constants that have no effect on the position of the likelihood maximum, the log-likelihood for this model is

$$\log P(\mathbf{A}|\mathbf{\Omega},\mathbf{g}) = \tfrac{1}{2}\sum_{ij}\biggl(A_{ij}\log\omega_{g_i g_j} - \frac{k_i k_j}{2m}\omega_{g_i g_j}\biggr). \quad (10)$$

Community detection now involves the maximization of this quantity with respect to the parameters $\mathbf{\Omega}, \mathbf{g}$ to find the best fit of the model to the observed network. This maximization can be achieved in a number of ways. As with the modularity, there are too many possible group assignments $\mathbf{g}$ to maximize exhaustively on any but the smallest of networks, but researchers have successfully applied a variety of approximate methods, including label switching algorithms [12], Kernighan–Lin style greedy algorithms [25, 26], spectral methods [6], Monte Carlo [24, 27], and belief propagation [28, 29].

III. THE PLANTED PARTITION MODEL AND MODULARITY MAXIMIZATION

We now come to the central result of this paper, the equivalence of modularity maximization to a particular case of the maximum likelihood method described above. We previously discussed a version of this equivalence in the context of work on spectral algorithms [5, 6] and it has also been discussed by Zhang and Moore [7] in the context of work on finite-temperature ensembles of graph partitions. Building on these works, our purpose in this paper is to make explicit the exact equivalence of the two approaches and investigate some of its consequences.

The planted partition model [30, 31] is a special case of the stochastic block model in which the parameters ω_{rs} describing the community structure take only two different values:

$$\omega_{rs} = \begin{cases} \omega_{\text{in}} & \text{if } r = s, \\ \omega_{\text{out}} & \text{if } r \neq s. \end{cases} \quad (11)$$

This is a less flexible model than the full stochastic block model. It effectively assumes that all communities in the network are similar in the sense of having the same in-group and between-group connection rates. Nonetheless, for networks that do have this property, fits to the model should recover the community structure accurately, and indeed it has been proved that such fits are optimal in that case [28, 32, 33].

In practice, if one wanted to apply the planted partition model, one should in almost all cases use a degree-corrected version of the kind described in Section II C. Let us explore the form of the log-likelihood, Eq. (10), for such a model. Following [5, 6] we note that Eq. (11) implies that

$$\omega_{rs} = (\omega_{\text{in}} - \omega_{\text{out}})\delta_{rs} + \omega_{\text{out}}, \quad (12)$$

$$\log\omega_{rs} = (\log\omega_{\text{in}} - \log\omega_{\text{out}})\delta_{rs} + \log\omega_{\text{out}}, \quad (13)$$

where δ_{rs} is the Kronecker delta, as previously. Substituting these forms into Eq. (10), we find the log-likelihood for the degree-corrected planted partition model to be

$$\begin{aligned}\log P(\mathbf{A}|\mathbf{\Omega},\mathbf{g}) &= \tfrac{1}{2}\sum_{ij} A_{ij}\biggl[\delta_{g_i g_j}\log\frac{\omega_{\text{in}}}{\omega_{\text{out}}} + \log\omega_{\text{out}}\biggr] \\ &\qquad - \tfrac{1}{2}\sum_{ij}\frac{k_i k_j}{2m}\bigl[(\omega_{\text{in}} - \omega_{\text{out}})\delta_{g_i g_j} + \omega_{\text{out}}\bigr] \\ &= \tfrac{1}{2}\log\frac{\omega_{\text{in}}}{\omega_{\text{out}}}\sum_{ij}\biggl(A_{ij} - \frac{(\omega_{\text{in}} - \omega_{\text{out}})}{(\log\omega_{\text{in}} - \log\omega_{\text{out}})}\frac{k_i k_j}{2m}\biggr)\delta_{g_i g_j} \\ &\qquad + m\bigl(\omega_{\text{out}} + \log\omega_{\text{out}}\bigr) \\ &= B\,\frac{1}{2m}\sum_{ij}\biggl(A_{ij} - \gamma\frac{k_i k_j}{2m}\biggr)\delta_{g_i g_j} + C, \end{aligned} \quad (14)$$

where B and C are constants that depend on ω_{in} and ω_{out} but not on $\mathbf{g}$, and

$$\gamma = \frac{\omega_{\text{in}} - \omega_{\text{out}}}{\log\omega_{\text{in}} - \log\omega_{\text{out}}}. \quad (15)$$

We have also made use of $\sum_{ij} A_{ij} = \sum_i k_i = 2m$ in the second equality of (14).

To perform community detection, one would now maximize this expression with respect to both the group assignments g_i and the parameters $\omega_{\rm in}$ and $\omega_{\rm out}$. But suppose for a moment that we already know the correct values of $\omega_{\rm in}$ and $\omega_{\rm out}$, leaving us only to maximize with respect to the group assignments. Comparing Eq. (14) with Eq. (6), we see that, apart from overall constants, (14) is precisely the generalized modularity $Q(\gamma)$, and hence the likelihood and the modularity have their maxima with respect to the g_i in the same place. Thus community detection by maximization of the likelihood for the planted partition model with known values of $\omega_{\rm in}$ and $\omega_{\rm out}$ is equivalent to maximizing the generalized modularity for the appropriate value of γ, given by Eq. (15). (We leave it as a exercise for the reader to show that a similar equivalence applies—with the same value of γ—between maximizing the likelihood for the non-degree-corrected stochastic block model and the modularity when one makes the choice (3) for P_{ij}.)

Among other things, this result tells us what the correct value of the resolution parameter γ is for the generalized modularity, an issue that has hitherto been undecided. The correct value is given by Eq. (15). An immediate corollary is that in most cases the conventional choice $\gamma = 1$, corresponding to the original, non-generalized modularity function of Eq. (5), is not correct.

Unfortunately, however, we cannot normally employ Eq. (15) directly to calculate γ, since we do not know the values of the parameters $\omega_{\rm in}$ and $\omega_{\rm out}$. We present one possible solution to this problem in Section III A, but for the moment let us proceed under the assumption that we know the correct value of γ.

The equivalence between modularity maximization and maximum-likelihood methods has a number of immediate implications. First of all, it provides a derivation of the modularity that is more rigorous and principled than the usual heuristic arguments: modularity maximization (with the correct choice of γ) is equivalent to fitting a network to a degree-corrected version of the planted partition model using the method of maximum likelihood. It also explains why the standard degree-dependent choice, Eq. (4), for the definition of the modularity is better than the uniform choice of Eq. (3). It is for the same reason that the degree-corrected block model is the correct choice for the analysis of most real-world networks: the uniform choice effectively assumes a network with a Poisson degree distribution, which is a poor approximation to most empirical networks. The degree-dependent choice, by contrast, fits networks of any degree distribution.

The equivalence of modularity and maximum-likelihood methods also implies that modularity maximization is a consistent method of community detection, i.e., that under suitable conditions it will correctly and without bias find community structure where present. This follows because maximum-likelihood fits to stochastic block models are also known to be consistent: when the maximum-likelihood method is applied to networks that were themselves generated from the same block model (either the traditional or degree-corrected model), it will return correct assignments of nodes to groups in the limit of large node degrees [12]. The consistency of modularity maximization has been demonstrated previously by other means [7, 34], but the equivalence with likelihood maximization makes the physical intuition behind it particularly clear.

A further point of interest is that while the value of γ in Eq. (15) is always positive, regardless of the values of $\omega_{\rm in}$ and $\omega_{\rm out}$, the value of the constant $B = m \log(\omega_{\rm in}/\omega_{\rm out})$ in Eq. (14) changes sign depending on which of $\omega_{\rm in}$ and $\omega_{\rm out}$ is larger. This means that maximization of the likelihood becomes equivalent to *minimization* of the modularity when $\omega_{\rm out} > \omega_{\rm in}$, i.e., when the network has so-called disassortative structure, in which connections are more common between groups than within them. The minimization of modularity to find such structure has been proposed previously on heuristic grounds [35], but the derivation here gives a rigorous foundation for the procedure.

On the other hand, the equivalence of maximum likelihood and maximum modularity methods also reveals some hidden assumptions and limitations of the modularity. The planted partition model, with its assumption, Eq. (11), that the edge parameters ω_{rs} take the same values for every community, is less powerful than the full stochastic block model and modularity maximization is similarly less powerful as a result. In effect, modularity maximization assumes all communities in a network to be statistically similar. This may be a good assumption in some networks, but there are certainly examples where it is not, and we would expect the modularity maximization method to perform less well in such cases than more general methods.

Some variants of the maximum-likelihood method also include additional parameters that allow for heterogeneous group sizes, but the version used here, to which modularity maximization is equivalent, includes no such embellishments, meaning that modularity maximization implicitly favors groups of uniform size, which could also hurt performance.

A. Value of the resolution parameter

A drawback of the equivalence we have demonstrated is that it applies only when we use the correct value of the resolution parameter γ, which normally we do not know. One can, however, make an empirical estimate of the value of γ using an iterative scheme as follows.

First, one makes an initial guess about the value of γ. This guess need not be particularly accurate: $\gamma = 1$ usually works fine. Then one finds the communities in the network by modularity maximization using this choice. This gives us some set of assignments g_i of nodes to groups—likely not optimal—from which we can then make an estimate of the parameters $\omega_{\rm in}$ and $\omega_{\rm out}$ by noting that the expected total number of in-group edges $m_{\rm in}$

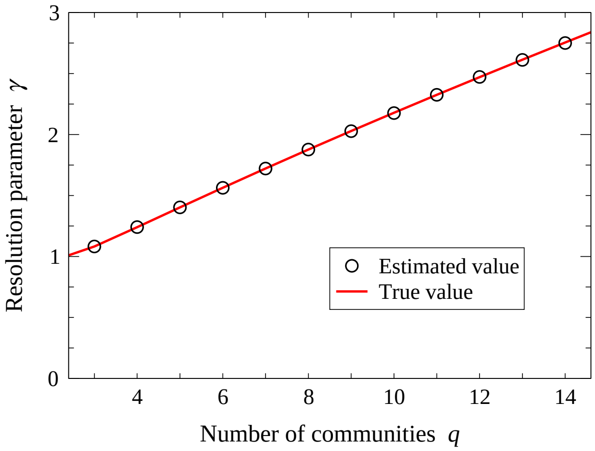

FIG. 1: Resolution parameter γ, estimated using the method described here, for a set of synthetic networks with varying numbers of communities. The networks were generated using the standard (non-degree-corrected) planted partition model with q equally sized groups of 250 nodes each and parameters ω_{in} and ω_{out} chosen so that each node has an average of 16 connections within its own group and 8 to every other group. Modularity was maximized using simulated annealing. The circles represent the estimated values of γ and the solid line represents the true values calculated from Eq. (15).

in the (degree-corrected) planted partition model is

$$m_{\text{in}} = \tfrac{1}{2}\sum_r \sum_{ij} \frac{k_i k_j}{2m}\omega_{rr}\delta_{g_i,r}\delta_{g_j,r} = \frac{\omega_{\text{in}}}{4m}\sum_r \kappa_r^2, \qquad (16)$$

where $\kappa_r = \sum_i k_i \delta_{g_i,r}$ is the sum of the degrees of the nodes in group r. Hence we can estimate ω_{in} from

$$\omega_{\text{in}} = \frac{2m_{\text{in}}}{\sum_r \kappa_r^2/2m}, \qquad (17)$$

using the observed value of m_{in} as an estimate of the expected value. Similarly for ω_{out} we have

$$\omega_{\text{out}} = \frac{2m_{\text{out}}}{\sum_{r\neq s}\kappa_r\kappa_s/2m} = \frac{2m - 2m_{\text{in}}}{2m - \sum_r \kappa_r^2/2m}, \qquad (18)$$

where m_{out} is the number of edges running between distinct groups.

Using these estimates of ω_{in} and ω_{out} we can now calculate a new value of γ from Eq. (15). Then we repeat the process, maximizing the modularity and recalculating ω_{in}, ω_{out}, and γ until we achieve convergence. The consistency of modularity maximization, mentioned earlier, implies that this procedure should converge to the correct value of γ (and the correct community structure) for sufficiently dense networks that actually are generated from the planted partition model. For all other networks (meaning, in practice, for all real-world applications of the method) we have no formal guarantees of correctness or convergence, though the same is also true of all other methods of community detection, including, but not limited to, community detection by statistical inference.

Network	n	m	q	γ
Karate club	34	78	2	0.78
Dolphin social network	62	159	2	0.59
Political blogs	1225	16780	2	0.67
Books about politics	105	441	2	0.59
Characters from *Les Miserables*	77	254	6	1.36
American college football	115	614	11	2.27
Jazz collaborations	198	2742	16	1.19
Email messages	1133	5451	26	3.63

TABLE I: Number of nodes n, number of edges m, number of communities q, and estimated value of the resolution parameter γ for a range of networks studied in the previous literature. The networks are the karate club network of Zachary [36], the dolphin social network of Lusseau *et al.* [37], the network of political weblogs studied by Adamic and Glance [38], the network of books about politics studied by Krebs (unpublished, but see for instance [15]), the network of interactions between fictional characters in the novel *Les Miserables* by Victor Hugo [8], the network of regular season games between Division I-A college football teams in the year 2000 [8], the network of collaborations between jazz musicians studied by Gleiser and Danon [39], and the network of email messages between university students of Ebel *et al.* [40]. The value of q used for each network is the generally accepted one, except for the last two networks, for which there does not appear to be a consensus. For these two networks we estimate the number of communities using the inference method of [41].

One might imagine that this would not be a very efficient method for calculating γ: it requires repeated maximization of different modularity functions until the correct value of γ is reached. In practice, however, we have found that it converges very quickly. In most cases we have examined, γ is calculated to within a few percent after just one iteration, and in no case have we found a need for more than ten iterations, so the method may in fact be quite serviceable. Figure 1 shows an example application to a set of artificially generated ("synthetic") networks for which the true value of γ is known and, as the figure shows, the algorithm is able to determine that value accurately in every case. Table I gives values of γ calculated using the algorithm for a number of real-world networks that have been used as test cases in previous community detection studies.

The values of γ vary in size, but there is an overall trend towards larger values in networks with larger numbers of communities, both among the synthetic networks and the real ones. This is perhaps not unexpected given that the resolution parameter γ was originally introduced precisely in order to deal with networks with larger numbers of communities. Recall that larger values of γ, and specifically values larger than the traditional value of 1, are needed in networks where the number of communities exceeds the resolution limit at $\sqrt{2m}$. None of the networks studied here approach this limit, but nonetheless we should not find it surprising that the larger values of q in both Fig. 1 and Table I are best treated using values $\gamma > 1$.

Whether the algorithm given here is in fact a useful one in practice is a debatable point. As we have shown, it does no more than the likelihood maximization method, and the latter in principle gives better results, since it does not assume that all groups are statistically similar. Modularity maximization does have the advantage of being less nonlinear than maximum-likelihood methods, which allows for some faster algorithms such as spectral [6, 15] and multiscale [19] algorithms. Still, the results derived here are primarily of interest not because of the algorithms they suggest, but because of the light they shed on the strengths and weaknesses of modularity maximization.

IV. CONCLUSIONS

We have shown that modularity maximization is a special case of the maximum likelihood method of community detection, as applied to the degree-corrected planted partition model. The equivalence between the two approaches highlights some weaknesses of the modularity maximization method: the method assumes all communities to have statistically similar properties, which may not be the case, and it also contains an undetermined parameter γ. Most often this parameter is assumed to take the value 1, but we have shown that this assumption is normally not correct and given an explicit formula for the correct value, along with a simple algorithm for computing it on observed networks.

Acknowledgments

The author thanks Aaron Clauset, Travis Martin, Cris Moore, and Cosma Shalizi for useful comments. This research was supported in part by the US National Science Foundation under grants DMS–1107796 and DMS–1407207.

[1] M. A. Porter, J.-P. Onnela, and P. J. Mucha, Communities in networks. *Notices of the American Mathematical Society* **56**, 1082–1097 (2009).

[2] S. Fortunato, Community detection in graphs. *Phys. Rep.* **486**, 75–174 (2010).

[3] M. E. J. Newman, Communities, modules and large-scale structure in networks. *Nature Physics* **8**, 25–31 (2012).

[4] M. Coscia, F. Giannotti, and D. Pedreschi, A classification for community discovery methods in complex networks. *Statistical Analysis and Data Mining* **4**, 512–546 (2011).

[5] M. E. J. Newman, Community detection and graph partitioning. *Europhys. Lett.* **103**, 28003 (2013).

[6] M. E. J. Newman, Spectral methods for network community detection and graph partitioning. *Phys. Rev. E* **88**, 042822 (2013).

[7] P. Zhang and C. Moore, Scalable detection of statistically significant communities and hierarchies, using message passing for modularity. *Proc. Natl. Acad. Sci. USA* **111**, 18144–18149 (2014).

[8] M. E. J. Newman and M. Girvan, Finding and evaluating community structure in networks. *Phys. Rev. E* **69**, 026113 (2004).

[9] M. E. J. Newman, Fast algorithm for detecting community structure in networks. *Phys. Rev. E* **69**, 066133 (2004).

[10] J. Reichardt and S. Bornholdt, Statistical mechanics of community detection. *Phys. Rev. E* **74**, 016110 (2006).

[11] U. Brandes, D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski, and D. Wagner, On finding graph clusterings with maximum modularity. In *Proceedings of the 33rd International Workshop on Graph-Theoretic Concepts in Computer Science*, number 4769 in Lecture Notes in Computer Science, Springer, Berlin (2007).

[12] P. J. Bickel and A. Chen, A nonparametric view of network models and Newman–Girvan and other modularities. *Proc. Natl. Acad. Sci. USA* **106**, 21068–21073 (2009).

[13] A. Clauset, M. E. J. Newman, and C. Moore, Finding community structure in very large networks. *Phys. Rev. E* **70**, 066111 (2004).

[14] J. Duch and A. Arenas, Community detection in complex networks using extremal optimization. *Phys. Rev. E* **72**, 027104 (2005).

[15] M. E. J. Newman, Modularity and community structure in networks. *Proc. Natl. Acad. Sci. USA* **103**, 8577–8582 (2006).

[16] L. Shuzhuo, C. Yinghui, D. Haifeng, and M. W. Feldman, A genetic algorithm with local search strategy for improved detection of community structure. *Complexity* **15**(4), 53–60 (2010).

[17] R. Guimerà, M. Sales-Pardo, and L. A. N. Amaral, Modularity from fluctuations in random graphs and complex networks. *Phys. Rev. E* **70**, 025101 (2004).

[18] A. Medus, G. Acuña, and C. O. Dorso, Detection of community structures in networks via global optimization. *Physica A* **358**, 593–604 (2005).

[19] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, Fast unfolding of communities in large networks. *J. Stat. Mech.* **2008**, P10008 (2008).

[20] M. Molloy and B. Reed, A critical point for random graphs with a given degree sequence. *Random Structures and Algorithms* **6**, 161–179 (1995).

[21] M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E* **64**, 026118 (2001).

[22] S. Fortunato and M. Barthélemy, Resolution limit in community detection. *Proc. Natl. Acad. Sci. USA* **104**,

36–41 (2007).

[23] P. W. Holland, K. B. Laskey, and S. Leinhardt, Stochastic blockmodels: Some first steps. *Social Networks* **5**, 109–137 (1983).

[24] K. Nowicki and T. A. B. Snijders, Estimation and prediction for stochastic blockstructures. *J. Amer. Stat. Assoc.* **96**, 1077–1087 (2001).

[25] B. Karrer and M. E. J. Newman, Stochastic blockmodels and community structure in networks. *Phys. Rev. E* **83**, 016107 (2011).

[26] B. W. Kernighan and S. Lin, An efficient heuristic procedure for partitioning graphs. *Bell System Technical Journal* **49**, 291–307 (1970).

[27] T. P. Peixoto, Efficient monte carlo and greedy heuristic for the inference of stochastic block models. *Phys. Rev. E* **89**, 012804 (2014).

[28] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová, Inference and phase transitions in the detection of modules in sparse networks. *Phys. Rev. Lett.* **107**, 065701 (2011).

[29] X. Yan, C. R. Shalizi, J. E. Jensen, F. Krzakala, C. Moore, L. Zdeborova, P. Zhang, and Y. Zhu, Model selection for degree-corrected block models. *J. Stat. Mech.* **2014**, P05007 (2014).

[30] A. Condon and R. M. Karp, Algorithms for graph partitioning on the planted partition model. *Random Structures and Algorithms* **18**, 116–140 (2001).

[31] F. McSherry, Spectral partitioning of random graphs. In *Proceedings of the 42nd IEEE Symposium on the Foundations of Computer Science*, pp. 529–537, Institute of Electrical and Electronics Engineers, New York (2001).

[32] L. Massoulié, Community detection thresholds and the weak ramanujan property. In *Proceedings of the 46th Annual ACM Symposium on the Theory of Computing*, pp. 529–694–703, Associate of Computing Machinery, New York (2014).

[33] E. Mossel, J. Neeman, and A. Sly, Reconstruction and estimation in the planted partition model. *Probability Theory and Related Fields* **162**, 431–461 (2015).

[34] Y. Zhao, E. Levina, and J. Zhu, Consistency of community detection in networks under degree-corrected stochastic block models. *Annals of Statistics* **40**, 2266–2292 (2011).

[35] M. E. J. Newman, Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E* **74**, 036104 (2006).

[36] W. W. Zachary, An information flow model for conflict and fission in small groups. *Journal of Anthropological Research* **33**, 452–473 (1977).

[37] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations. Can geographic isolation explain this unique trait? *Behavioral Ecology and Sociobiology* **54**, 396–405 (2003).

[38] L. A. Adamic and N. Glance, The political blogosphere and the 2004 US election. In *Proceedings of the WWW-2005 Workshop on the Weblogging Ecosystem* (2005).

[39] P. Gleiser and L. Danon, Community structure in jazz. *Advances in Complex Systems* **6**, 565–573 (2003).

[40] H. Ebel, L.-I. Mielsch, and S. Bornholdt, Scale-free topology of e-mail networks. *Phys. Rev. E* **66**, 035103 (2002).

[41] M. E. J. Newman and G. Reinert, Estimating the number of communities in a network. Preprint arXiv:1605.02753 (2016).