RESEARCH ARTICLE

Axioms for clustering simple unweighted graphs: No impossibility result

James Willson, Tandy Warnow *

Siebel School of Computing and Data Science, University of Illinois Urbana-Champaign, Urbana, Illinois, United States of America

* warnow@illinois.edu

Abstract

In 2002, Kleinberg proposed three axioms for distance-based clustering, and proved that it was impossible for a clustering method to satisfy all three. While there has been much subsequent work examining and modifying these axioms for distance-based clustering, little work has been done to explore axioms relevant to the graph partitioning problem when the graph is unweighted and given without a distance matrix. Here, we propose and explore axioms for graph partitioning for this case, including modifications of Kleinberg's axioms and three others: two axioms relevant to the "Resolution Limit" and one addressing well-connectedness. We prove that clustering under the Constant Potts Model satisfies all the axioms, while Modularity clustering and iterative k-core both fail many axioms we pose. These theoretical properties of the clustering methods are relevant both for theoretical investigation as well as to practitioners considering which methods to use for their domain science studies.

Author summary

In 2002, Kleinberg proposed three axioms for distance-based clustering and proved that it was not possible for any clustering method to simultaneously satisfy all three axioms. Here, we examine these axioms in the context where the input network is given without any pairwise distance matrix and is instead a simple unweighted graph. For this case we propose corresponding axioms, and we include three additional axioms, two related to the resolution limit and the other related to well-connectedness. We establish that some methods, such as optimizing under the Constant Potts Model, satisfy all the axioms we pose, but that others (notably clustering under the Modularity optimization problem) fail to satisfy some of these axioms. This study sheds light on limitations of existing clustering methods.

Introduction

Graph clustering, also known as community detection or graph partitioning, is the problem of taking a graph $G = (V, E)$ as input and returning a partition of the vertex set into disjoint subsets, referred to equally as clusters or communities. In some contexts, the graph is given as a distance matrix D so that $D[i, j]$ is the distance between vertices i and j.

In 2002, Kleinberg [1] defined three axioms (Richness, Consistency, and Scale-Invariance) for clustering based on distances, and proved that it was impossible for any clustering method to satisfy all three axioms. The Refinement-Consistency axiom is a relaxation of Consistency, but Kleinberg [1] also proved an impossibility result with this substitution.

The apparent impossibility of distance-based clustering to satisfy all stated desirable axioms drove research in several directions. For example, [2] addresses the Consistency axiom, pointing out cases where it might not be desirable. Furthermore, there has been work in sidestepping axioms by defining the number of clusters in advance [3, 4]. For example, [3] does this by replacing Richness with k-Richness, which is a version of Richness restricted only to consider clusterings with k clusters, and [4] argue that Consistency should not hold if the "correct" number of clusters changes. Additional work has also been done applying the principles of Kleinberg's distance-based axioms to quality measures instead of directly to the clustering function. For example, [5] formulates such a set of axioms and shows that these new axioms do not lead to an impossibility result.

Here we consider axiomatic properties of clustering when the input is an unweighted simple graph (i.e., neither the edges nor the vertices are weighted) and where the graph is given without a distance matrix. The motivation for considering these simple unweighted graphs is that many real-world graphs are of this form (e.g., citation graphs). In addition, while it is certainly *possible* to define a pairwise distance matrix relating the vertices (e.g., the length of the shortest path between each pair of vertices), such approaches lose information about the input graph (see discussion in [6, 7]). Finally, graph clustering when the input does not include a distance matrix is very common (e.g., see the DIMACS report [8]).

Very little has been done to discuss axiomatic approaches for graph clustering when the input is a graph without any distance matrix. However, three studies [6, 7, 9] provide overviews of the literature related to axiomatic properties of clustering methods, with Kleinberg's axioms reformulated for the distanceless case. Of these, [9] provides theoretical advances in axiomatic properties of clustering methods when the input graph has non-negative edge-weights, and established that Modularity-optimization satisfies Richness.

Another property that has been discussed in the literature is the "resolution limit" [10], which roughly speaking indicates that a clustering method may fail to return obvious communities if they are too small. This resolution limit was established for clustering based on Modularity [11] in [10], using a ring-of-cliques as an example of how Modularity can fail to find the obvious communities (i.e., the cliques) as the number of cliques grows but not their size. This observation led to the development of other methods, including an approach to clustering based on optimizing under the Constant Potts Model [12], for which the failure on the ring-of-cliques example does not hold.

We expand on the prior work by formulating seven axioms suitable for clustering methods that operate on unweighted simple networks. Four of these axioms are reformulations of Kleinberg's original Richness and Consistency axioms, following on [6] and [7] for the distanceless case. The final three axioms include one that addresses how well-connected the clusters are (i.e., considering the size of the minimum edge cut of each cluster) and two others that are related to the resolution limit, one of which was formulated in [12]. Each of these axioms can be seen as "desirable" properties of a clustering method, and is motivated by real-world analyses; for example, [13] documented that standard clustering methods applied to real-world datasets often produced poorly connected clusters, and [14] documented that the Louvain [15] algorithm could produce clusters that were not even connected.

We find that optimizing under the Constant Potts Model (CPM) (i.e., CPM-optimization) satisfies all the axioms we pose, but all other clustering methods we study, including

optimizing under the Modularity criterion (i.e., Modularity-optimization), fail to satisfy most of the axioms we pose.

Our study provides new evidence that CPM-optimization has superior theoretical properties compared to Modularity-optimization. It also sheds light on the tricky question of which methods suffer from the "resolution limit", as the original formulation in [10] and the response from [12] do not fully overlap. In addition to proposing new research questions for theoreticians, the insights from this study provide useful insight for domain scientists in selecting methods for use in their empirical work.

Background

Notation and definitions

We use N to denote a network, V to denote its node set, and E to denote its edge set. We assume the network is simple (i.e., no parallel edges or self-loops) and undirected, so that if an edge exists between two nodes, it is not oriented from one node to the other. Therefore, an individual edge between nodes v and w is denoted by the pair (v, w). We also assume that the network is unweighted, which means all edges have the same (unit) weight. The degree of a node v is the number of its neighbors (i.e., the number of elements in the set $\{w \in V: (v, w) \in E\}$). An isolated node in N is one that has degree 0 (i.e., it has no neighbors).

A connected component of a network N is a maximal subset S of the nodes in N that is connected, so that for all pairs of nodes in S there is a path connecting them through nodes in S. These are more simply just referred to as the "components" of the network. We do not assume that the input network is connected, and so we allow for the possibility that there are two or more components.

We use $\mathcal{C}$ to denote a clustering of the network (i.e., a partition of the nodes of the network into disjoint sets). We allow for clusters to have only one node, and refer to these as "singleton clusters". Individual clusters within the clustering $\mathcal{C}$ will be written variably, often as c or c_i, but sometimes using A, B, C, etc. An edge cut for a cluster C is a set of edges between nodes in C so that if the edges in the set (but not their endpoints) are removed, the cluster will disconnect into two or more parts. The min cut size of a cluster is the number of edges in a smallest edge cut for the cluster. If a cluster is not connected, then its min cut size is 0, while if a cluster is connected, then the min cut size is at least 1. If a connected cluster can be disconnected by the removal of a single edge, then that single edge is called a "cut edge" for the cluster. Finally, a "tree" is a connected acyclic graph; therefore, a "tree cluster" is a connected cluster where every edge is a cut edge.

Clustering methods

We discuss theoretical properties of Modularity, CPM (constant Potts model), and IKC (iterative k-core) clustering. We also consider two additional "toy" clustering methods:

- Components-are-Clusters: the clustering method that returns the connected components of the network as the clusters

- Nodes-are-Clusters: the clustering method that returns every node as a singleton cluster

Modularity.   Given a clustering $\mathcal{C}$ of N, we define the Modularity score of $\mathcal{C}$ as follows. Let e_c denote the number of edges internal to cluster c and d_c denote the sum of the degrees of nodes found in cluster c (noting that the degree of a node v in a cluster c is the total number of

neighbors of v, whether or not in the cluster). The Modularity score of $\mathcal{C}$ is

$$\mathcal{H}_{mod} = \sum_{c \in \mathcal{C}} \left[\frac{e_c}{|E|} - \left(\frac{d_c}{2|E|} \right)^2 \right]. \tag{1}$$

The Modularity optimization problem, which takes as input a network and seeks a clustering with the largest modularity score, was proven NP-hard in [16]. We make a minor modification to the Modularity optimization problem by requiring that the clusters be connected. Note that the only consequence of this modification is that isolated nodes in the network will be placed in singleton clusters.

The Constant Potts Model (CPM) clustering problem. Optimizing under the Constant Potts Model (CPM) [12] was developed as a way of addressing the weakness in Modularity optimization that it is subject to the resolution limit [10]. The CPM optimization criterion takes a parameter γ (the resolution value). Letting e_c denote the number of edges and n_c the number of nodes in cluster c, the CPM score of $\mathcal{C}$ is

$$\mathcal{H}_{CPM} = \sum_{c \in \mathcal{C}} \left[e_c - \gamma \binom{n_c}{2} \right]. \tag{2}$$

Note that the optimization problem depends on the resolution parameter γ; in this study, we will constrain γ to be in the open interval $(0, 1)$. When not clear by context, we refer to the usage of CPM with a fixed value for parameter γ as CPM(γ).

IKC and IKC(no-mod). The iterative k-core [17] algorithm (also known as IKC) is a deterministic clustering algorithm based on finding k-cores, which are maximal connected subgraphs where every vertex is adjacent to at least k other vertices in the subgraph. A k-core can be found by iteratively pruning all nodes with degree smaller than k from the graph until no more remain. IKC operates by determining the largest k for which a k-core exists, removes that k-core, and then recurses. IKC takes a parameter k_0 and only returns those clusters that satisfy two properties: the minimum degree within the cluster is at least k_0 and every non-singleton cluster has positive Modularity score. In this study, we consider two versions of IKC: both have $k_0 = 0$ and one drops the requirement of positive Modularity for each non-singleton cluster. We refer to the version that drops Modularity as IKC(no-mod) and the other as simply IKC.

Kleinberg's axioms

In distance-based clustering, a clustering function f takes set S with n elements and an $n \times n$ distance matrix d and returns $\mathcal{C}$, a partition of S. With this notation, [1] proposed the following three axioms:

- **Scale Invariance**: Given some constant $\alpha > 0$, $f(d) = f(\alpha \cdot d)$. In other words, if all the distance between points in the data are multiplied by a constant amount this should not affect the output of the clustering method.

- **Richness**: The clustering function f satisfies, for all networks and clusterings $\mathcal{C}$, that there is some distance matrix d on the network such that $f(d) = \mathcal{C}$. In other words, there should not be any clustering that is impossible to obtain.

- **Consistency**: Given two distance functions d and d', $f(d) = f(d')$ if d' transforms d in the following way: If i and j are from the same cluster then $d'(i, j) \leq d(i, j)$; otherwise, if they are from different clusters $d'(i, j) \geq d(i, j)$. The motivation for this axiom is that if the clusters are made tighter, or if the clusters are made more distinct from one another (by being

moved further away from each other), then these changes should reinforce the existing clustering rather than change it.

- **Refinement-Consistency**: This is the same as Consistency except for the following change: instead of requiring that $f(d) = f(d')$, it is sufficient that every cluster in $f(d')$ be a subset of a cluster in $f(d)$. In other words, $f(d')$ is a refinement of $f(d)$, when each is considered as a set of sets. Kleinberg's study showed that his impossibility result held even with this relaxation.

The resolution limit

As shown by Fortunato and Barthélemy in [10], Modularity optimization can fail to return what are obvious true communities if they are too small. Specifically, Fortunato and Barthélemy described an infinite family of networks formed of rings of cliques, each clique connected to each of its two neighbors by a single edge, where the cliques are a constant size but the number of the cliques increases. Fortunato and Barthélemy prove that if the number the cliques is large enough, then Modularity will stop returning the cliques as communities and will instead return sets of cliques as communities. They described this by saying that Modularity suffers from the resolution limit.

Traag *et al.* [12] proposed the following definition of what it means for an optimization problem (or method that solves the optimization problem exactly) to be "resolution-limit free":

Let $\mathcal{C} = \{C_1, C_2, \ldots, C_q\}$ be a $\mathcal{H}$-optimal partition of a graph G. Then the objective function $\mathcal{H}$ is called resolution-limit-free if for each subgraph H induced by $\mathcal{D} \subset \mathcal{C}$, the partition $\mathcal{D}$ is also $\mathcal{H}$-optimal.

Traag *et al.* prove that optimizing under the Constant Potts Model (CPM) is resolution-limit-free but optimizing under the Modularity criterion is not resolution-limit-free.

Of concern to us is that this definition of resolution-limit-free does not address in full the issue raised by [7]. For example, a method that returns each component in the network as a cluster satisfies the definition of "resolution-limit-free" as provided by [12] but fails to return the cliques inside the ring-of-cliques component as communities and will instead return the entire component.

Well-connectedness

A natural expectation of a community (i.e., cluster) is that it should be both dense (i.e., have more edges inside the cluster than would be expected by chance) and well-connected (i.e., not have a small edge cut).

However, definitions for "well-connected" vary by study. For example, [14] established a lower bound on the cut size for any cluster in a clustering that is optimal for the CPM criterion as a function of the resolution parameter γ, so that if an edge cut splits a cluster into two sets A and B then the edge cut has size at least $\gamma \times |A| \times |B|$, and used this as the definition for "well-connected" clusters. [13] showed empirically that many clustering methods, including optimizing the CPM criterion using the Leiden [14] software, often produced clusters with small edge cuts, and even produced clusters that were trees. Based on this observation, [13] proposed instead that a cluster be considered well-connected if the size of a min cut in a cluster with n nodes is greater than $f(n)$, where $f(n)$ is a non-decreasing function provided by the user that increases to infinity.

Our distanceless axioms

In the distanceless context, our input is a simple unweighted undirected graph $N = (V, E)$, where V is the vertex set and E is the edge set. We propose seven axioms, where the first four are obtained by modifying Kleinberg's axioms for the distanceless context, one is designed to address well-connectedness, and a final two relate to the resolution limit (one introduced earlier in [12]).

- **Richness**: A clustering method M satisfies richness if, for any clustering $\mathcal{C}$ of a set V, there exists an edge set E so that $M(N) = \mathcal{C}$ when $N = (V, E)$. Note that we allow for the trivial clusterings, i.e., when all the nodes are in the same cluster, or when they are each in separate clusters.

- **Standard Consistency**: A clustering method M satisfies standard consistency if, for every graph $N = (V, E)$ and output clustering $M(N)$, when E' differs from E by the removal of edges between clusters in $M(N)$ or the addition of edges within clusters in $M(N)$, then $M(N') = M(N)$ where $N' = (V, E')$.

- **Refinement Consistency**: This is a relaxation of Standard Consistency where adding internal edges to a cluster is allowed to split the cluster apart but no other changes are allowed. More formally, a clustering method M satisfies refinement consistency if $M(N')$ is identical to $M(N)$ or is a refinement of $M(N)$ for all networks N and N', where N' differs from N only by the addition of edges between nodes in the same cluster in $M(N)$.

- **Inter-edge Consistency**: This is a relaxation of Standard Consistency, where the clustering must remain unchanged when edges between clusters are removed. More formally, a clustering method M satisfies inter-edge consistency if $M(N')$ is identical to $M(N)$ for all networks N and N', where N' differs from N only by the deletion of edges between nodes in different clusters in $M(N)$.

- **Connectivity**: We extend [13] to define this axiom. Let $f : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ be a non-negative and non-decreasing function that approaches infinity. We say that a cluster is well-connected with respect to f if the size of the minimum edge cut exceeds $f(n)$, where n is the number of nodes in the cluster. We say that a clustering method M satisfies connectivity if and only $\exists$ $f : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ that is non-negative, non-decreasing, and that approaches infinity, so that for all networks N and all non-singleton clusters c in the clustering produced by M, c is well-connected with respect to f.

- **Pair-of-Cliques**: This axiom is a small start towards a more thorough evaluation of robustness to the resolution-limit, since the characterization in [12] does not adequately address the concerns raised in [10]. Recall that [10] presented the resolution limit problem with an example of a network containing a ring of n-cliques, and established that as the number of cliques increased Modularity optimization would fail to return the cliques as communities, returning instead clusters containing two or more of these cliques. Since a ring of cliques is not the only condition where methods can fail to detect small or meso-scale communities, we consider a simple case where one component in the network contains a pair of n-cliques, connected by an edge, and we refer to this as a Pair-of-Cliques component. We say a graph partitioning method satisfies the Pair-of-Cliques axiom if there is a constant n_0 such that if the network N has a Pair-of-Cliques component of size at least n_0 then the clustering method would return A and B as separate clusters, where the Pair-of-Cliques component has cliques A and B.

- **Fixed-Point**: We consider the property proposed in [12], whereby a method is said to be "resolution-limit-free" if iteratively applying the clustering method to subnetworks induced by nodes in a subset of the clusters will not change the clustering. More formally, if the method M applied to a network N produces a set of clusters $C_1, C_2, \ldots, C_k$, then for any $I \subset \{1, 2, \ldots, k\}$, applying M to the subnetwork of N induced by $\{v \in C_i, i \in I\}$ would return clustering $\{C_i, i \in I\}$. Note that this induced subgraph does not remove edges between clusters, and that according to this property, reclustering any single cluster will return that cluster. We refer to this by saying that M satisfies the Fixed-Point axiom.

The choice of axioms is motivated by generally desirable properties of clustering methods, rather than properties that have been observed in some clusterings. For example, as argued in several studies, such as [9], clustering methods should preferentially be *local*, and so changes in one part of the network should not impact clusters in another part of the network. This property informs our selection of axioms. The Richness, Standard Consistency, Refinement Consistency, and Fixed-Point axioms have been proposed before, and while the Pair-of-Cliques axiom may be new, it is closely related to the prior literature, and in particular to the Ring-of-Cliques example given in [10]. Inter-edge Consistency follows naturally from Standard Consistency. Connectivity (i.e., having the minimum cut size for each cluster grow with the number of nodes) is also natural and has been discussed in the prior literature, where it is also referred to as "set conductance" [18, 19]. Here we provide some additional motivation for the Refinement Consistency axiom.

Consider a ring lattice, i.e., a cycle of length n with vertices numbered sequentially in the cycle by 1, 2, ..., n, in which every node v is adjacent to the $2k$ nearest nodes, and so k nodes on each side of v. For example, if $n = 100$ and $k = 2$, then the ring lattice has 100 nodes and each node is adjacent to 4 nodes in the ring lattice. Now make a path of L of these ring lattices, each with the same values for n and k (with $n >> k$), with the first ring lattice connected to the second by a single edge, the second connected to the third by a single edge, etc. It is not hard to see that IKC, as well as many other clustering methods, will return a clustering with L clusters in which each of these ring lattices is a cluster. Now suppose that we add just enough edges within each ring lattice so that vertices 1, 2, ..., $n/2$ form a clique and vertices $n/2 + 1, n/2 + 2, \ldots, n$ also form a clique. On this modified network, IKC and many other clustering methods will now return $2L$ clusters, by breaking each of original clusters into two cliques. This is *desirable* behavior, and is why we include this Refinement Consistency axiom.

Results

In some cases we provide sketches of proofs, leaving full proofs to the end of the paper. We begin with a lemma.

Lemma 1. *If M is a clustering method that satisfies Connectivity, then for some $n_0 \geq 1$, no clusters returned by M of size at least n_0 have cut edges.*

Proof. Suppose M satisfies Connectivity. By definition, there is some function f that is non-decreasing and satisfies $f(x) \to \infty$ as $x \to \infty$, such that for all networks N and all clusters C returned by M on N, the min cut size of C is strictly greater than $f(n)$ where n is the size of C. Since $f(x)$ is non-decreasing and converges to infinity, there is some n_0 so that for all $n \geq n_0$, $f(n) \geq 1$. Hence, for all networks N and clusters of size at least n_0 returned by M on N, the min-cut size for the cluster will be strictly greater than 1, and so no found cluster of size at least n_0 can have a cut edge.

Theory for Components-are-Clusters

Recall that the Components-are-Clusters method returns every connected component as a cluster.

Theorem 1. *Components-are-Clusters satisfies the Richness, Standard Consistency, and Fixed Point axioms, but fails Connectivity and the Pair-of-Cliques axioms.*

Proof. First we establish Richness. Suppose we are given clustering $\mathcal{C}$ of a set V of nodes. For every cluster in $\mathcal{C}$, we make all the nodes in the cluster pairwise-adjacent, i.e., each cluster now becomes a clique. No other edges are added, so that every cluster is a connected component in the network. Components-are-Clusters will return each connected component as a cluster, and thus satisfies Richness.

For Standard Consistency, note that adding edges between nodes in a connected component can never connect two disconnected components, nor can it split a component. The same is true for removing edges between two connected components. Thus, Components-are-Clusters satisfies Standard Consistency.

For Connectivity, the proof is by contradiction. If Components-are-Clusters satisfied Connectivity, then by Lemma 1, there is some n_0 such that no cluster of size at least n_0 returned by Components-are-Clusters can have a cut edge. Now consider a network N that has a component of size n_0 that is a tree. Components-are-Clusters would return that component, thus failing Connectivity.

For the Pair-of-Cliques axiom, any component consisting of a pair of cliques connected by a single edge would be returned by Components-are-Clusters. Since we can make such a component arbitrarily large, Components-are-Clusters fails the Pair-of-Cliques axiom.

Finally, Components-are-Clusters is easily seen to satisfy the Fixed Point axiom.

Theory for Nodes-are-Clusters

Recall that the Nodes-are-Clusters method returns every node as a singleton cluster.

Theorem 2. *Nodes-are-Clusters fails Richness and Pair-of-Cliques and satisfies Connectivity, Standard and Refinement Consistency, and the Fixed Point axioms.*

Proof. Nodes are clusters will return n clusters given any network on n nodes, and so fails the Richness axiom. Similarly, it fails Pair-of-Cliques, as it cannot return any clique of size greater than 1 as a cluster. The connectivity axiom is satisfied, since letting the axiom is only applied to non-singleton clusters. Standard consistency follows, since adding or deleting edges from a network does not change the clustering. Similarly, Nodes-are-Clusters trivially satisfies the Fixed Point axiom.

Theory for CPM

Theorem 3. *For all values $\gamma > 0$, CPM(γ) follows all axioms.*

That CPM(γ) satisfies the Fixed Point axiom was established in [12]. We now provide proofs that CPM(γ) follows the remaining axioms, assuming in each case that $0 < \gamma < 1$ is fixed but arbitrary.

Lemma 2. *CPM(γ) is Rich.*

Proof. Let V be a set of nodes and $\mathcal{C}$ a partition of V. For each set in the partition, form a clique. For any $\gamma > 0$, the clustering that puts every clique into a cluster attains the largest possible score, and all other clusterings have lower scores. Thus, CPM(γ) satisfies the Richness axiom.

Lemma 3. *CPM(γ) follows Inter-Edge Consistency.*

Proof. Let $\gamma > 0$ be fixed, and let $G = (V, E)$ be a network. Let $\mathcal{C}$ be a clustering $\{c_1, c_2, \cdots, c_m\}$ of G that is optimal for the CPM criterion with resolution parameter γ. Let E' be a subset of E produced by removing some edges whose endpoints are in different clusters in $\mathcal{C}$. We let

$CPM(c, E)$ denote the CPM score for cluster c given edge set E and e'_c denote the number of edges from E' in c. From Eq 2, we see that

$$\forall c_i \in \mathcal{C}, \; CPM(c_i, E) = CPM(c_i, E').$$

Additionally,

$$\forall c \subseteq V, \quad CPM(c, E) \geq CPM(c, E'), \, as$$
$$\forall c \subseteq V, \quad e_c \geq e'_c$$

Therefore, $\mathcal{C}$ remains optimal.

Lemma 4. *CPM(γ) follows Standard (and therefore Refinement) Consistency.*

Proof. Let $\gamma > 0$ be fixed, and let $G = (V, E)$ be a network. Consider an optimal clustering $\mathcal{C}$ and imagine adding a single edge into one of the clusters. The score of $\mathcal{C}$ will go up by 1, since the edge was added to a cluster within $\mathcal{C}$. As per Eq 2, the most that the CPM score of any other clustering can increase by is exactly 1; hence $\mathcal{C}$ remains optimal after adding that edge. Therefore, inductively, a clustering that is optimal for a network given edge set E remains optimal if we add edges within the clusters. We also note that removing edges does not need to be considered, as CPM was shown to satisfy inter-edge consistency in Lemma 3.

Lemma 5. *CPM(γ) is Connective.*

Proof. A proof of this theorem also follows from Eq D1 in the Supplementary Information in [14]; here we provide a simple proof.

Given $\gamma > 0$, we let function f_γ be defined by $f_\gamma(n) = \lceil \gamma(n - 1) \rceil$. Note that f_γ maps positive integers to integers, is non-decreasing, and grows unboundedly (i.e., $f_\gamma(n) \to \infty$ as $n \to \infty$). We will show that for every γ, every network N, and every $CPM(\gamma)$-optimal clustering of N, the minimum edge cut of any cluster c in the clustering is at least size $f_\gamma(n)$, where n is the number of nodes in the cluster c. Therefore, this will establish that CPM(γ) is Connective.

Suppose C is a cluster with n nodes in a CPM-optimal clustering of a network N for some fixed γ. We consider an edge cut E_0 for C. Since C is a cluster in a CPM-optimal clustering, dividing C into two clusters cannot improve the CPM-score. Hence, whatever division of C into two sets is produced by deleting E_0, the best that can happen is that the CPM-score is not reduced.

Let n' denote the number of nodes on one side of the edge cut, A denote the number of edges connecting those nodes, and B denote the number of edges connecting the nodes on the other side of the edge cut. The CPM score of cluster C is $A + B + |E_0| - \gamma \binom{n}{2}$. Therefore, we obtain

$$A + B + |E_0| - \gamma \binom{n}{2} \geq A - \gamma \binom{n'}{2} + B - \gamma \binom{n - n'}{2},$$

because the score of the cluster C is at least the sum of the scores of the subclusters produced by deleting E_0. Therefore,

$$|E_0| \geq \gamma \binom{n}{2} - \gamma \left[\binom{n'}{2} + \binom{n - n'}{2} \right].$$

We then note

$$
\begin{aligned}
|E_0| \quad &\geq \gamma \binom{n}{2} - \gamma \max_{n'}\left[\binom{n'}{2} + \binom{n-n'}{2}\right] \\
&= \gamma\left[\binom{n}{2} - \binom{n-1}{2} - \binom{1}{2}\right] \\
&= \gamma\left[\binom{n}{2} - \binom{n-1}{2}\right] \\
&= \gamma(n-1).
\end{aligned}
$$

Hence, $|E_0| \geq f_\gamma(n)$ for any edge cut E_0 separating a CPM(γ)-optimal cluster C with n nodes.

Note that Lemma 10 establishes that the connectivity guarantee provided for CPM(γ) depends on γ, and that small values for γ allow for large clusters with cut edges being returned.

Lemma 6. *CPM(γ) satisfies the Pairs-of-Cliques axiom.*

Proof. To show that CPM(γ) satisfies the Pairs-of-Cliques axiom, we must show that for a fixed $\gamma > 0$, there is value for n where all cliques with n vertices or more will be clustered in separate clusters if connected by a single edge. Since CPM(γ) is connective, we can pick N large enough so that the mincut size for any cluster of size at least N is at least two. Hence, if C_1 is a component in N that has two n-cliques A and B connected by an edge and $2n \geq N$, then no clustering of C_1 in a CPM-optimal clustering can have a cut edge. Therefore, each of the clusters of C_1 in an optimal CPM clustering must be subsets of A or B. It is easy to see that the CPM score is maximized by returning A and B as clusters, and so CPM(γ) follows the Pair-of-Cliques axiom.

Theory for Modularity

Theorem 4. *Modularity follows Richness but violates all the other axioms, i.e., the Standard and Refinement Consistency, Inter-edge Consistency, Connectivity, Pair-of-Cliques, and Fixed Point axioms.*

Proof. Modularity was shown to satisfy Richness in Theorem 1 of [9], and was shown to fail the Fixed Point axiom in [12].

We now sketch the proof that Modularity violates Refinement Consistency and hence Standard Consistency (see Lemma 8 for full details). In Lemma 8, we consider a network N that has a component G_1 that is a pair-of-cliques (i.e., it has two node-disjoint n-cliques A and B that are connected by an edge, with $n \geq 5$). Lemma 7 establishes that a Modularity-optimal clustering of N will either return G_1 as a cluster or will return the two n-cliques A and B as clusters. In Lemma 8, we then consider a network N with G_1 as one component and with a second component G_0 that is a p-star (i.e., the graph with a single node adjacent to p other nodes, and no other edges). Lemma 9 shows that for $n \geq 5$ and p large enough, the Modularity-optimal clustering of the network will produce A and B as two clusters, and that when G_0 is a $(p + 1)$-clique then a Modularity-optimal clustering will return G_1 as a cluster. Thus, adding edges within a cluster can change the clustering, but in a way that does not satisfy Refinement Consistency. Hence, Modularity violates Refinement Consistency, which in turn establishes that it violates Standard Consistency. Note that this argument also establishes that Modularity violates the Pair-of-Cliques axiom.

The proof that Modularity violates Inter-edge Consistency is provided in Lemma 9 and uses a similar argument to Lemma 8. We construct a graph with two components, where the first component is a pair-of-cliques component. In Lemma 9, we show that this network has the following two properties: The optimal modality clustering of the network containing both

components returns the pair-of-cliques component as a single cluster and splits the other component into multiple clusters, and if any edge is removed from the second component, then the first property is no longer satisfied. Hence, Modularity violates Inter-edge Consistency.

We now prove that Modularity is not Connective. If it were, then by Lemma 1, there would be a value n_0 so that all clusters of size at least n_0 have min cut size greater than 1, and so do not have any cut edge. Let n be picked so that $2n \geq n_0$, and consider the network given in Lemma 9 where G_1 is a component with $2n$ vertices containing two n-cliques connected by an edge and G_0 is a sufficiently large p-star, so that the optimal Modularity clustering returns G_1 as a single cluster. Note that G_1 has a cut-edge, so that its minimum cut size is 1. This contradicts our assumption, proving that Modularity violates connectivity.

Finally, we prove that Modularity fails the Pair-of-Cliques component. As shown in Corollary 1, for any $k \geq 5$, a network that two components, with one a Pair-of-Cliques component where the cliques are of size $\binom{k}{2}$ and the other a clique of size $2\binom{k}{2} + 1$, the optimal modularity clustering will return the two components as the two clusters. Thus, Modularity fails the Pair-of-Cliques axiom.

Theory for IKC

Recall that we examine two versions of IKC: the "default" setting that enforces a positive modularity score on all its non-singleton clusters, and the other, which we refer to as IKC(no-mod), that does not. Here we present the theory specifically for the default usage of IKC.

Theorem 5. *IKC violates the Richness, Standard Consistency, Refinement Consistency, Inter-edge Consistency, Connectivity, Pair-of-Cliques, and Fixed Point axioms.*

Proof. To see that Richness is violated, note that a clustering containing a single cluster that includes every vertex in a network has a Modularity score of zero, and thus can never be considered a valid cluster for any edge set. The proofs for IKC violating Standard Consistency, Refinement Consistency, and Inter-edge Consistency are based on networks shown in Fig 1. In each subfigure, the shown graph is one component of a two-component network, where the other component is a single edge. The edge colors in each subfigure indicate edges that are present (blue), present but deleted (red), or not present but will be added (green).

For the proof that IKC violates Standard Consistency, we refer to Fig 1(a). This figure describes a network N_1 with blue edges with two components, where one of these components is a simple 6-cycle and the other component is a single edge; the green edges are added to define a modified network N_1'. Running IKC on N_1 would return the shown 6-cycle component as a cluster, since it is a 2-core and has positive Modularity. In N_1', the vertex set $\{1, 2, 3, 4\}$ forms a 3-core that has positive Modularity, and there is no 4-core in N_1'. Hence, when IKC is applied to N_1', the 3-core $\{1, 2, 3, 4\}$ would be returned as the cluster found in the first iteration. Therefore, the IKC output clustering has been changed by the addition of edges within a cluster, and so IKC violates Standard Consistency.

To see that IKC violates Refinement Consistency, see Fig 1(b). The initial network N_2 contains only the blue edges and the final network N_2' also contains the green edges. In N_2, the round vertices form a 3-core that has positive Modularity, and there is no 4-core in N_2; therefore, the round vertices would be returned as a cluster by IKC when applied to N_2. After removing the 3-core of round vertices, the square vertices form a 2-core. and since they have positive Modularity, they would be returned as the second cluster by IKC when clustering N_2. However, the network N_2' has the green edges added. In N_2', the component shown constitutes a 3-core that has positive Modularity and so would be returned as a cluster by IKC when applied to N_2'. Thus, IKC fails Refinement Consistency.

(a) N_1, green edges are added to create N_1'. **(b)** N_2, green edges are added to create N_2'

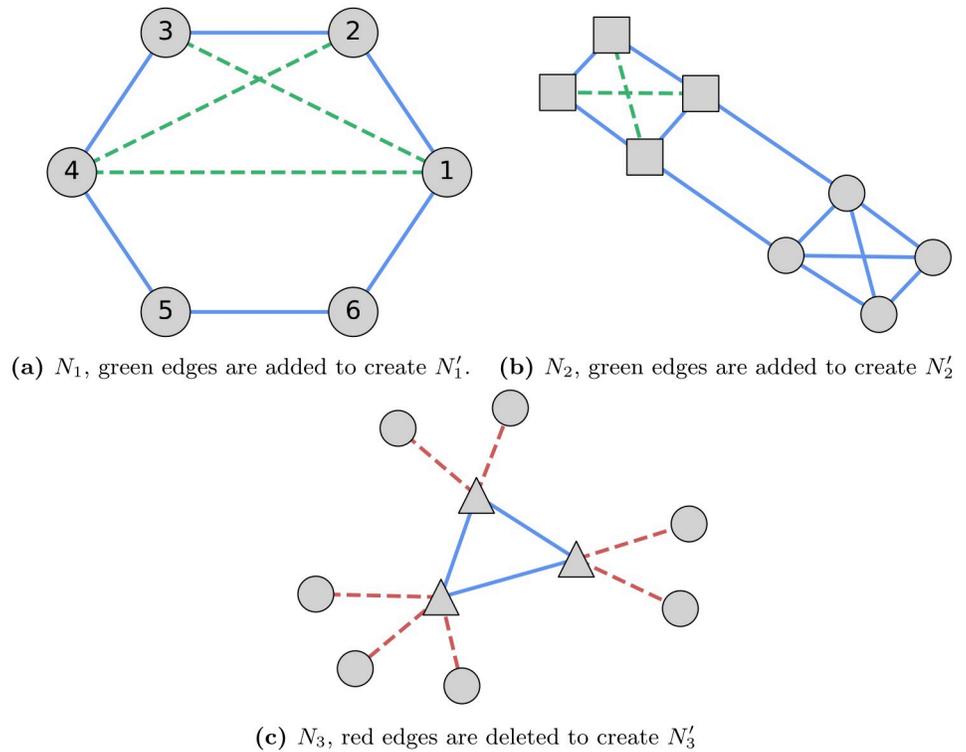
(c) N_3, red edges are deleted to create N_3'

Fig 1. Theoretical properties of IKC and IKC(no-mod). In each case, the network shown is one connected component of a network with two connected components, where the second component is a single edge (hence the shown component always has positive modularity). Green edges are added to a starting network, red edges are deleted from a starting network, and blue edges represent edges that are in the starting and final network. Subfigure (a) gives one component in a network N_1 where IKC and IKC(no-mod) both fail Standard Consistency. Subfigure (b) gives one component in a network N_2 where IKC and IKC(no-mod) both fail Refinement Consistency. Subfigure (c) gives an example of one component in a network N_3 where IKC will return only singleton clusters (due to its check for positive modularity). However, if the red edges were deleted, then IKC would return the 3-clique, establishing IKC fails Inter-Edge Consistency.

We now show that IKC violates Inter-Edge Consistency. The network N_3 described in Fig 1 (c) has two components, a component with a single edge, which we will refer to as C_1, and the displayed 10-node component with both blue and red edges, which we will refer to as C_2. In the first iteration of IKC applied to N_3, the blue edge 3-clique is detected as a 2-core, but since its modularity score is not positive (specifically, its modularity score is $3/11 - (13/22)^2 < 0$), it would not be returned as a cluster, and its three nodes would be turned into singleton clusters. Hence, on network N_3, IKC will return only one non-singleton cluster, and that is the two nodes in C_1, and will not return any non-singleton clusters for component C_2. This establishes that the red edges shown in this figure go between different clusters obtained by IKC on N_3.

Recall that the network N_3' is formed by deleting the red edges from N_3, and so has the same vertex set, with one component a 3-clique, one component a single edge, and then seven isolated nodes. When IKC is applied to N_3', it would find the 3-clique, and since it has positive modularity (specifically, its modularity score is $3/7 - (3/7)^2$), it would return the 3-clique as a cluster. In other words, we have shown that deleting edges between different clusters changed what is returned by IKC, which contradicts the Inter-Edge Consistency Axiom.

We provide a proof by contradiction that IKC violates Connectivity. Suppose it did; then for some function f that is increasing unboundedly and for all n and all clusters of size n

returned by IKC, the edge cut for the cluster will be of size at least $f(n)$. Since $f(x) \to \infty$ as $x \to \infty$, this means that for some $n_0 \geq 2$, no clusters of size at least n_0 returned by IKC have cut-edges. Now consider a network with two components, where one component C has two n_0-cliques connected by an edge and the other component contains a single edge. This component C is a $n_0 - 1$-core and has positive modularity, and the network does not contain any n_0-core. Hence, IKC would return the component C as the first cluster, and then the single edge as the second cluster. However, C has a cut edge, violating our assumption, and establishing that IKC fails Connectivity.

We next consider whether IKC satisfies the Fixed Point axiom. Note that IKC requires that a returned cluster have positive modularity. Therefore, if C is a k-clique returned by IKC it has positive modularity within its network. However, when IKC is reapplied to the cluster C, it calculates the modularity score with respect to C as the entire network. Thus, C will now have zero modularity score, and so will not be returned by IKC. Therefore, IKC fails the Fixed Point axiom.

We now show that IKC fails the Pair-of-Cliques axiom. Consider a network that has at least two components, where the first has n-cliques A and B connected by a single edge, and there is at least one edge not in the first component. IKC would return this first component as a cluster since it has positive modularity, and so would fail to return the cliques A and B as clusters. The other case is where the network has only one component, which has the two n-cliques connected by edges. Since the modularity score of an entire network with a single component is 0, IKC will return only singletons. Thus, in both cases, IKC will fail to return the cliques A and B and will return only singletons. Since this outcome holds for all values of n, this establishes that IKC fails the Pair-of-Cliques axiom.

Theory for IKC(no-mod)

Theorem 6. *IKC(no-mod) satisfies the Richness, Inter-Edge Consistency, and Fixed Point axioms, but violates the Standard Consistency, Refinement Consistency, Connectivity, and Pair-of-Cliques axioms.*

Proof. To establish Richness, we consider the same network as used in the proof of richness for Modularity (Theorem 4), where every component is a clique. It is easy to see that when running IKC(no-mod), each component of the network is returned as a cluster, since every non-singleton component has positive Modularity and is a k-core for some value of k. Notably, IKC(no-mod) can return all vertices in a single cluster, as the positive modularity restriction has been removed.

The proofs for IKC violating Standard Consistency, Refinement Consistency, and Connectivity do not rely on checking for positive Modularity, and so apply to IKC(no-mod). It is trivial to see that IKC(no-mod) returns the Pair-of-Cliques component as a cluster, since it is an $(n - 1)$-core (where each clique has n nodes). Hence, IKC(no-mod) fails the Pair-of-Cliques axiom. Finally, it is easy to see that because IKC(no-mod) does not check for positive modularity, IKC(no-mod) satisfies the Fixed Point axiom.

We now establish that IKC(no-mod) satisfies inter-edge consistency. Consider two clusters c_1 and c_2 returned by IKC(no-mod), with at least one edge between them, and assume c_1 a k-core and c_2 a k'-core, with $k \leq k'$. Note that $k \neq k'$, as otherwise the connected subgraph on $c_1 \cup c_2$ would be a k-core and would be returned. Removing an edge e connecting these two clusters would only affect the degree of nodes in these two clusters, so all other clusters would remain unaffected by any edge deletion. Furthermore, after removing e, c_1 would still be a k-core and c_2 would still be a k'-core. Therefore, in running IKC on the network obtained by deleting edge e between the clusters c_1 and c_2, these sets would still be considered for being

clusters, and since modularity is not evaluated, c_1 and c_2 would still be returned as clusters by IKC(no-mod). Moreover, since no other cluster is affected, IKC(no-mod) would return the same clustering on the resultant graph. Hence, IKC(no-mod) follows inter-edge consistency.

Some additional observations

We now show some examples of clusterings that satisfy some axioms but not others, to help elucidate the meaning of the axioms.

Connectivity vs. Pair-of-Cliques. Here we show that satisfying Connectivity does not imply satisfying Pair-of-Cliques. Note that there are multiple ways that a method can fail the Pair-of-Cliques axiom. One way is that the method can return the Pair-of-Cliques component as a cluster (and this is true for IKC, IKC(no-mod), Modularity, and Components-as-Clusters), in which case the method fails Connectivity. Another way the method can fail the Pair-of-Cliques axiom is if it breaks the Pair-of-Clique component into two or more clusters where at least one of the clusters is not one of the two cliques of the component. The Nodes-as-Clusters method is an example of such a method. Here we present yet another clustering method that satisfies Connectivity but fails Pair-of-Cliques. Consider a clustering method that breaks a network into the smallest number of subsets, so that each subset is a clique of size at most k, where k is a user-provided parameter. Since the minimum edge cut size in a clique of size n is $n - 1$, this clustering method satisfies Connectivity with respect to $f(n) = n - 1$. However, this clustering method will not satisfy Pair-of-Cliques, since it cannot return the cliques of size n when $n > k$.

For completion, we also provide an example of a method that satisfies Pair-of-Cliques but not Connectivity. Consider the method that explicitly examines each component to see if it has a cut edge. If the component does not have a cut edge, the method returns the component as a cluster, but if the component does have a cut edge, the method repeatedly removes cut edges until there are no more remaining, at which point all the components are returned as clusters. Such a method will satisfy Pair-of-Cliques (since every Pair-of-Cliques component has a cut edge), but will fail Connectivity, since if the network has arbitrarily large cycles, the set of returned clusters will contain arbitrarily these large cycles. Note that a cycle has min cut size of 2. Since satisfying Connectivity requires a function f that grows unboundedly, there must be a value n so that all clusters of size at least n have min cut size at least 3. This contradicts the method satisfying Connectivity.

Inter-edge consistency vs. Fixed-Point and Pair-of-Cliques axioms. We present a clustering method that satisfies inter-edge consistency but not the Fixed Point Axiom nor the Pair-of-Cliques axiom. Consider the clustering method that takes an integer parameter $k > 0$ and partitions each component of the network into exactly k clusters, so as to maximize the number of edges that are contained within clusters. It is easy to see that such a clustering method will satisfy inter-edge consistency for every $k \geq 1$. Now consider the Fixed Point Axiom. If $k \geq 2$ and we take the output of such a clustering and apply the algorithm to one single cluster, it will break that cluster into $k \geq 2$ parts, thus failing the Fixed Point Axiom. Note now if $k \geq 3$, that this method will fail the Pair-of-Cliques axiom, since it will never break a component into two pieces.

Inter-edge consistency vs. Refinement Consistency. We show two clustering methods, where the first satisfies Inter-Edge Consistency but not Refinement Consistency, and the second satisfies Refinement Consistency but not Inter-Edge Consistency. The IKC(no-mod) method satisfies inter-edge consistency but not refinement consistency, as we establish in Theorem 6. Now consider the clustering method that returns a clustering with the minimum number of clusters such that each non-singleton cluster has positive modularity and is a clique of

size at least 3. Now consider the network N_3 described in Fig 1(c). Because of the requirement for positive modularity score, the method would return all singleton clusters when given N_3 as input. The removal of red edges (that all go between different clusters, since the singleton nodes are clusters) would produces a network that has a 3-clique component, and the method would return that component as a cluster, as it has positive modularity score. Hence the method fails Inter-Edge Consistency. Furthermore, by design the method automatically satisfies Refinement Consistency, since it is not possible to add edges to any cluster that is returned.

Additional theory

Additional proofs for modularity

We will use the following notation. (i) If X is a subset of the nodes in a network N, then Q_X denotes the modularity score of the cluster X within a clustering. (ii) If $\mathcal{C}$ is a clustering of a network N, then $Score(\mathcal{C})$ denotes the total modularity scores of the clusters in the clustering. (iii) The largest modularity score across all clusterings of a network N is written as $Modularity(N)$. (iv) If $G \subset N$ is a subgraph of network N, then the largest modularity score of G across all clusterings of N that make G either into a single cluster or a collection of clusters is denoted by $Modularity_N(G)$.

Lemma 7. *Let G_1 be a component in network N, where G_1 consists of two node-disjoint cliques A and B, each with n > 5 nodes, and a single edge connecting nodes in the two cliques. There are only two options for how G_1 is clustered in a modularity-optimal clustering of N: either G_1 is returned as a single cluster, or G_1 is split into two clusters, A and B.*

Proof. To demonstrate that splitting the nodes in these two cliques apart is never optimal, we take a look at [20]. In Theorem 1, [20] proves that two endpoints of an edge will be in the same cluster if these endpoints are identically connected to every other node in the network. With this theorem, most of possible partitions splitting the cliques can be discarded; however, there is a single exception. Say edge (a_0, b_0) is the edge connecting the two n-cliques A and B; a partition separating a_0 from the clique A is contained in might still be valid, when only considering this Theorem, as a_0 is connected to b_0, while this is not true for any other node in A. This leaves us with several cases we still need to consider. Let $A_0 = A\backslash\{a_0\}$ and $B_0 = B\backslash\{b_0\}$. Then the options for clustering that we must consider are: Option 1: A_0, $\{a_0\}$, B_0, and $\{b_0\}$. Option 2: A_0, $\{a_0\}$, and B. Option 3: A, B_0, and $\{b_0\}$. Option 4: A_0, B_0, and $\{a_0, b_0\}$. Option 5: $A \cup \{b_0\}$ and B_0. Option 6: A_0 and $B \cup \{a_0\}$.

Ruling out Clustering options 1–3. First we show that $Q_A > Q_{A_0} + Q_{\{a_0\}}$; this will establish also that $Q_B > Q_{B_0} + Q_{\{b_0\}}$. Hence, we will be able to rule out clustering options 1–3.

Let E denote the total number of edges in the network. Then the modularity of clusters A_0 and $\{a_0\}$ (and hence also of B_0 and $\{b_0\}$) can be written as:

$$Q_{A_0} = \frac{\binom{n-1}{2}}{|E|} - \frac{(n-1)^4}{4|E|^2}$$

$$Q_{\{a_0\}} = -\frac{(n+1)^2}{4|E|^2}$$

We then write the modularity of A (and B) as:

$$Q_A = \frac{\binom{n-1}{2} + n - 1}{|E|} - \frac{[(n-1)^2 + n]^2}{4|E|^2}$$

and with some arithmetic we get:

$$
\begin{aligned}
Q_A \quad &= \frac{\binom{n-1}{2} + n - 1}{|E|} - \frac{(n-1)^4 + 2n(n-1)^2 + n^2}{4|E|^2} \\
&= \frac{\binom{n-1}{2} + n - 1}{|E|} - \frac{(n-1)^4}{4|E|^2} - \frac{2n(n-1)^2 + n^2}{4|E|^2} \\
&= \frac{\binom{n-1}{2}}{|E|} - \frac{(n-1)^4}{4|E|^2} + \frac{n-1}{|E|} - \frac{2n(n-1)^2 + n^2}{4|E|^2}
\end{aligned}
$$

which is equivalent to:

$$
Q_A = Q_{A_0} + \frac{n-1}{|E|} - \frac{2n(n-1)^2 + n^2}{4|E|^2}
$$

Thus to determine whether a clustering in which A appears as a cluster has a better modularity score than the clustering obtained by splitting A into two clusters, A_0 and $\{a_0\}$, we evaluate the conditions under which $Q_A > Q_{A_0} + Q_{\{a_0\}}$. This is equivalent to showing

$$
Q_{A_0} + \frac{n-1}{|E|} - \frac{2n(n-1)^2 + n^2}{4|E|^2} > Q_{A_0} - \frac{(n+1)^2}{4|E|^2}
$$

which is equivalent to:

$$
\frac{n-1}{|E|} - \frac{2n(n-1)^2 - (n+1)^2 + n^2}{4|E|^2} > 0
$$

which is equivalent to:

$$
4|E|(n-1) > 2n(n-1)^2 - (n+1)^2 + n^2
$$

This is always true, as we now argue. Note that E is the set of edges in the network and so $|E| \geq n^2 - n + 1$ (the number of edges in component G_1) and so $4|E|(n-1) \geq 2n(n-1)^2$. Note also that $n^2 - (n+1)^2$ is always negative. Therefore $Q_A > Q_{A_0} + Q_{\{a_0\}}$ and $Q_B > Q_{B_0} + Q_{\{b_0\}}$. As a result, clustering options 1–3 can be eliminated.

Ruling out clustering option 4. The modularity of $\{a_0, b_0\}$ can be written as:

$$
Q_{\{a_0, b_0\}} = \frac{1}{|E|} - \frac{n^2}{|E|^2}
$$

We see that $Q_A > Q_{A_0} + Q_{\{a_0, b_0\}}$ if and only if

$$
\frac{n-1}{|E|} - \frac{2n(n-1)^2 + n^2}{4|E|^2} + Q_{A_0} > \frac{1}{|E|} - \frac{n^2}{|E|^2} + Q_{A_0}
$$

if and only if

$$
\frac{n-1}{|E|} - \frac{1}{|E|} > \frac{2n(n-1)^2}{4|E|^2} - \frac{3n^2}{4|E|^2}
$$

if and only if

$$
4|E|(n-2) > 2n(n-1)^2 - 3n^2
$$

And since $-3(n-1)^2 > -3n^2$, it follows that $Q_A > Q_{A_0} + Q_{\{a_0,b_0\}}$ if

$$
\begin{aligned}
4|E|(n-2) \quad &> (2n-3)(n-1)^2 \\
&= (2n-3)(n-2)^2 + (2n-3)^2 \\
&= (2n-3)(n-2)^2 + (4n^2 - 12n + 8) + 1 \\
&= (2n-3)(n-2)^2 + (4n-4)(n-2) + 1
\end{aligned}
$$

If both sides are divided by $(n-2)$ we get

$$
\begin{aligned}
4|E| \quad &> (2n-3)(n-2) + 4n - 4 + \frac{1}{n-2} \\
&= 2n^2 - n + \frac{1}{n-2} + 6
\end{aligned}
$$

Note that this is always true. Hence, we have established $Q_A > Q_{A_0} + Q_{\{a_0,b_0\}}$. Therefore, we can rule out option 4.

Ruling out clustering options 5 and 6. To eliminate the final options, 5 and 6, we show that $Q_A + Q_B > Q_{A \cup \{b_0\}} + Q_{B_0}$. We write Q_A as

$$
Q_A \quad = \frac{\binom{n}{2}}{|E|} - \frac{\left[2\binom{n}{2} + 1\right]^2}{4|E|^2}
$$

and additionally,

$$
Q_{A \cup \{b_0\}} \quad = \frac{\binom{n}{2} + 1}{|E|} - \frac{\left[2\binom{n}{2} + n + 1\right]^2}{4|E|^2}
$$

Since $Q_B = Q_A$ and $Q_{B_0} = Q_{A_0}$, we know

$$
Q_B - Q_{B_0} = \frac{n-1}{|E|} - \frac{2n(n-1)^2 + n^2}{4|E|^2}
$$

Hence, $Q_A + Q_B > Q_{A \cup \{b_0\}} + Q_{B_0}$ if and only if

$$
\begin{aligned}
\left(\frac{\binom{n}{2}}{|E|} - \frac{\left[2\binom{n}{2} + 1\right]^2}{4|E|^2} \right) &- \left(\frac{\binom{n}{2} + 1}{|E|} - \frac{\left[2\binom{n}{2} + n + 1\right]^2}{4|E|^2} \right) \\
&+ \left(\frac{n-1}{|E|} - \frac{2n(n-1)^2 + n^2}{4|E|^2} \right) > 0
\end{aligned}
\tag{3}
$$

Simplifying, $Q_A + Q_B > Q_{A \cup \{b_0\}} + Q_{B_0}$ if

$$
\begin{aligned}
0 < {} & \frac{4\binom{n}{2}^2 + 4\binom{n}{2} + 1}{4|E|^2} + \frac{4\binom{n}{2}^2 + 4\binom{n}{2}(n+1) + (n+1)^2}{4|E|^2} \\
& - \frac{1}{|E|} + \frac{n-1}{|E|} - \frac{2n(n-1)^2 + n^2}{4|E|^2} \\
= {} & \frac{4n\binom{n}{2} + (n+1)^2 + 1}{4|E|^2} + \frac{n-2}{|E|} - \frac{2n(n-1)^2 + n^2}{4|E|^2} \\
= {} & \frac{3n^2 + 2}{4|E|^2} + \frac{n-2}{|E|} - \frac{2n^3 + 5n^2 - 2n}{4|E|^2} \\
= {} & \frac{n-2}{|E|} - \frac{2n^3 + 2n^2 - 2n - 2}{4|E|^2}
\end{aligned}
\tag{4}
$$

if and only if

$$
\begin{aligned}
4|E|(n-2) \quad & > 2n^3 + 2n^2 - 2n - 2 \\
& = (n-2)(2n^2 + 6n + 10) + 18
\end{aligned}
$$

Dividing both sides by $(n-2)$ gives us that $Q_A + Q_B > Q_{A \cup \{b_0\}} + Q_{B_0}$ if

$$
4|E| > 2n^2 + 6n + 10 + \frac{18}{n-2}
$$

Since $|E| \geq n^2 - n + 1$, we see that $Q_A + Q_B > Q_{A \cup \{b_0\}} + Q_{B_0}$ if

$$
4n^2 - 4n + 4 > 2n^2 + 6n + 10 + \frac{18}{n-2}
$$

which is equivalent to

$$
2n^2 - 10n - 6 - \frac{18}{n-2} \quad > 0
$$

This is true for $n > 5$. Thus we eliminate the final options, 5 and 6.

Therefore, for any network with this structure, optimizing modularity does not separate the nodes within the cliques A and B. The lemma follows.

Lemma 8. *Modularity violates Standard and Refinement Consistency.*

Proof. Consider a network $G = (V, E)$ with two components, G_0 and G_1, with G_1 as in Lemma 7; thus, G_1 contains two cliques A and B, each with e edges, connected by a single edge. Let E_{other} denote the edge set for the other component G_0. By Lemma 7, in a modularity-optimal clustering of this network, there are only two options for how G_1 is clustered: either as a single cluster (containing all the nodes in G_1) or as two clusters, A and B.

We define Q_2 to be the modularity score of G_1 when the clustering produces two clusters (i.e., each clique is considered a single cluster) and Q_1 is the modularity score of G_1 when the entire component is considered a single cluster (thus, the index indicates how many clusters G_1 is split into). Equivalently, $Q_2 = Q_A + Q_B$ and $Q_1 = Q_{G_1}$. We are interested in understanding when $Q_1 > Q_2$, so that returning a single cluster for G_1 is preferable to returning A and B as single clusters. We find $\Delta Q = Q_1 - Q_2$, by referring to Eq 14 from [10]. (Using the notation from [10], in our network, $l_1 = l_2 = e$, $b_1 = b_2 = 0$ and $a_1 = a_2 = \frac{1}{e}$, since a single edge connects

the two cliques.) Hence we obtain:

$$\Delta Q = \frac{2|E| - 4e^2 - 4e - 1}{2|E|^2}$$

Note that $Q_1 > Q_2$ if and only if:

$$|E| > 2e^2 + 2e + \frac{1}{2} \tag{5}$$

This inequality can be rewritten as (by subtracting $2e + 1$):

$$|E_{other}| > 2e^2 - \frac{1}{2} \tag{6}$$

Thus, the modularity score of the clustering where G_1 is one component is larger than the modularity score of the clustering where G_1 is two clusters if and only if (6) holds.

Now consider *Modularity*(*N*), the score of the best achievable modularity clustering of *N*. We write this as *Modularity*(*N*) = *Modularity*(G_1) + *Modularity*(G_0), as we require that output clusters be connected. Recall *Modularity*(G_1) = max(Q_1, Q_2) is ensured by Lemma 7. Hence *Modularity*(*N*) = max(Q_1, Q_2) + *Modularity*(G_0).

Next we consider the component G_0. We will let G_0 be a *p*-star (i.e., a graph with a center node adjacent to *p* other nodes that all have degree 1). Consider an optimal modularity clustering of G_0 within this network. If this clustering breaks G_0 into two or more clusters, then exactly one cluster contains the center node and all the other clusters are singletons (since we require that the clusters be connected). Let *x* be the number of singleton clusters (that do not include the center node), and assume the total number of nodes is *p* + 1 (so there are *p* nodes adjacent to the center node). Then the modularity score of this clustering is given by:

$$\begin{aligned}
Q_{p-star} \quad &= \frac{p-x}{|E|} - \left(\frac{2p-x}{2|E|}\right)^2 - x\left(\frac{1}{2|E|}\right)^2 \\
&= \frac{p}{|E|} - \frac{x}{|E|} - \frac{4p^2 - 4px + x^2}{4|E|^2} - \frac{x}{4|E|^2} \\
&= \frac{4px - 4|E|x - x^2 - x - 4p^2 + 4|E|p}{4|E|^2}
\end{aligned}$$

Note that this equation is maximized at *x* = 0, since $x \geq 0$ and $|E| \geq p$, so clustering the entire star into a single cluster has the optimal modularity score.

We set up *G* so that G_0 is a *p*-star in our original network (so that G_0 is returned as a cluster) and then we add edges until G_0 is a clique, creating a new network. We can select values for *e* (the number of edges in the cliques in G_1) and *p* (where G_0 is a *p*-star) that will cause Inequality (6) to be violated (and so indicate $Q_2 > Q_1$) in the case where G_0 is a *p*-star and not violated (and so indicate $Q_1 > Q_2$) in the case where G_0 is a (*p* + 1)-clique. This will prove that Modularity violates refinement consistency, and so also violates standard consistency.

For instance, if $p = 2e$ (and recalling that $e \geq \binom{5}{2}$), then when G_0 is a *p*-star:

$$|E_{other}| = 2e < 2e^2 - \frac{1}{2},$$

which violates Inequality (6), and hence means that G_1 will be split into two clusters, *A* and *B*,

in an optimal modularity clustering. However, when G_0 is a clique:

$$|E_{other}| = \binom{2e+1}{2} = \frac{2e(2e+1)}{2} = 2e^2 + e > 2e^2 - \frac{1}{2}$$

which obeys Inequality (6). Note that this argument applied to all $e \geq \binom{5}{2}$.

To summarize, we see that returning A and B as separate clusters is modularity-optimal in the case where G_0 is a p-star, whereas returning G_1 as a single cluster is modularity-optimal when G_0 is a $(p + 1)$-clique. This means that Modularity violates Standard Consistency.

The proof of Lemma 9 yields this helpful lemma as an immediate corollary:

Corollary 1. *Let $e = \binom{k}{2}$ where $k \geq 5$ is a positive integer and let N_1 be a network with two components, C_1 and C_2, where C_1 is a Pair-of-Cliques component with two cliques A and B connected by an edge, with A and B each having exactly e edges. Let the other component C_2 of N_1 be a p-star, with $p = 2e$. Let N_2 also be a network with two components, C_1 and C_3 (i.e., C_1 is the same Pair-of-Cliques component as for N_1) and where C_3 is a $(p + 1)$-clique. Then the optimal modularity clustering of N_1 will return A and B as separate clusters, and an optimal modularity clustering of N_2 will return C_1 as a cluster.*

Lemma 9. *Modularity fails the inter-edge consistency axiom.*

Proof. We form a network N where G_1 (a pair-of-cliques component) is one component, and then we add a network N' that is not connected to G_1, and that has the following properties:

- Property (1): The optimal modularity clustering of $N = G_1 \cup N'$ returns G_1 as a single cluster, and splits N' into at least one more cluster than the number of components in N'.

- Property (2): N' is minimal subject to Property (1), which means that if we delete *any* edge of N', then N longer satisfies Property (1).

Now suppose such a network N' exists (and note that N' depends on the value of n, where G_1 has two n-cliques). Since N' satisfies Property (2), if we delete any edge in N' at all, then Property (1) does not hold. Let $\mathcal{C}$ be an optimal modularity clustering of N. Now consider the network N^* produced by the deletion of an edge e_0 that goes between two different clusters in $\mathcal{C}$ (such an edge exists since the optimal clustering produces more clusters than there are components), and then running modularity on N^* to produce clustering $\mathcal{C}'$. Since N' was minimal subject to Property (1), it follows that Property (1) does not hold for $N' \setminus \{e_0\}$ (the network produced by deleting the edge e_0 but not its endpoints from N'). Hence, in the clustering $\mathcal{C}'$, either G_1 is not returned as a cluster or N^* does not splits into at least two clusters. Therefore, no matter how $\mathcal{C}'$ differs from $\mathcal{C}$, it follows that Modularity violates inter-edge consistency.

Therefore, all we need to do to complete the proof is to establish that such a network N' exists that satisfies Properties (1) and (2), above. Consider a network N with two components. G_1 is made of two cliques of equal size (containing e edges), connected by a single edge. We let $N' = G_0$, which is made of two sets of vertices. The first set contains e^2 edges and the second contains $e^2 - 1$ edges; these two sets are connected by a single edge e^*; therefore, $|E_{G_0}| = 2e^2$. Specifically, we need to show Property (1), i.e., that the optimal modularity clustering of $N = G_1 \cup G_0$ returns G_1 as a single cluster and splits G_0 into at least two clusters, and Property (2), i.e., that the removal of any edge in G_0 will not satisfy Property (1). This will complete the proof.

Given the fact that G_1 and G_0 are components and we require that the clusters be connected, the modularity score for the entire network N satisfies $Modularity(N) = Modularity_N(G_1) + Modularity_N(G_0)$.

By the proof of Lemma 8, G_1 will be clustered as a single cluster if and only if $|E(G_0)| > 2e^2 - \frac{1}{2}$. Given how we have defined G_0 and G_1, this is equivalent to saying that G_1 will be returned as a single cluster if and only if $2e^2 > 2e^2 - \frac{1}{2}$. Hence for how we have defined the network, G_1 is returned as a single cluster in any optimal modularity clustering.

We now show that if we do not remove edge e^*, then G_0 is clustered into at least two clusters in any modularity-optimal clustering of the network N, which will establish Property (1). We refer to Eq 14 from [10], where we define $\Delta Q = Q_4 - Q_3$ and Q_3 as the modularity score of the case where G_0 is split into two clusters across the single cut edge e^* and Q_4 is the score for the case where G_0 is considered a single cluster. Note that if $\Delta Q < 0$ then returning a single cluster for G_0 is not modularity-optimal.

$$\Delta Q = \frac{2|E| - \left(2 + \frac{1}{e^2}\right)\left(\frac{1}{e^2 - 1} + 2\right)e^2(e^2 - 1)}{2|E|^2}$$

$$= \frac{2|E| - \left(\frac{2}{e^2 - 1} + 4 + \frac{1}{e^4 - e^2} + \frac{2}{e^2}\right)(e^4 - e^2)}{2|E|^2}$$

$$= \frac{2|E| - \left(\frac{4e^8 - 8e^6 + 3e^4 + e^2}{e^4 - e^2}\right)}{2|E|^2}$$

$$= \frac{2|E| - 4e^4 + 4e^2 + 1}{2|E|^2}$$

Hence,

$$\Delta Q < 0 \text{ if and only if } |E| < 2e^4 - 2e^2 - \frac{1}{2}$$

We know that $|E| = 2e^2 + 2e + 1$. Hence, for all $e \geq 2$, $\Delta Q < 0$. Therefore, for the network N we have constructed, the modularity-optimal clustering of G_0 has at least two clusters, and we have established that $N' = G_0$ satisfies Property (1).

We now establish that $N' = G_0$ satisfies (2). Imagine our network N^*, where the edge e^* contained in G_0 is removed. According to the proof of Lemma 8, G_1 will be returned as a single cluster if and only if $|E(G_0)| - 1 > 2e^2 - \frac{1}{2}$, which is the same as $2e^2 - 1 > 2e^2 - \frac{1}{2}$, which is never true. Hence if we remove the edge e^* in G_0, then G_1 will not be returned as a single cluster.

Hence, $N' = G_0$ satisfies Properties (1) and (2) above, and the lemma is proven.

Additional theory for CPM

The following lemma is not directly relevant to understanding the properties of CPM-optimization with respect to the axioms we stated, but sheds some light on the behavior of CPM(γ) and how this is impacted by γ.

Lemma 10. *If N is a network and C is a component in the network, then for all sufficiently small γ, every optimal CPM(γ) clustering returns C as a cluster. Specifically, if $\gamma \binom{n}{2} < 1$ and C is a component of size n in a network N, then C will be returned as a cluster in every CPM(γ)-optimal clustering.*

Proof. We begin by calculating the CPM(γ) score of the cluster C. Letting $E(C)$ denote the edge set of C and n denote the number of nodes in C, we obtain:

$$CPM(C) = |E(C)| - \gamma \binom{n}{2}$$

Since the CPM function is continuous in γ, as $\gamma \to 0$, this will become arbitrarily close to $|E(C)|$ (but is always smaller). Hence in particular, we can pick γ small enough to produce

$$CPM(C) \geq |E(C)| - 1$$

Specifically, if $\gamma \binom{n}{2} < 1$, the above equation holds.

Let γ_0 be such a value, and consider a clustering of N that is optimal under CPM(γ_0). Suppose that the optimal clustering of N splits C into $k \geq 2$ clusters, $C_1, C_2, \ldots, C_k$. Since C is connected, there is at least one edge in $E(C)$ that is not in any cluster. Letting m_i denote the number of edges in cluster C_i, the CPM score of this optimal clustering (for C) is given by

$$\sum_{i=1}^{k} CPM(C_i) < \sum_{i=1}^{k} m_i \leq |E(C)| - 1$$

Note that the first inequality follows since $\gamma_0 > 0$ is required, and the second inequality follows since at least one edge is not in any cluster. However, this is strictly less than the CPM score of the cluster containing the entire component C, contradicting its optimality. Hence, for small enough γ, the optimal CPM(γ) clustering returns the entire component as a cluster.

While CPM(γ) is provably connective, the function f that provides the guarantee depends on γ. Now, suppose we ask instead: Is there a function $f : \mathbb{R}^+ \to \mathbb{R}^+$ that works for all γ, i.e., so that for all γ, the mincut size for every CPM-optimal cluster of size n is greater than $f(n)$? The answer is unfortunately *no*, as we now argue.

Suppose such a function f were to exist. In this case, we could pick a value for n so that $f(n) = 2$. For that value of n, we would then pick γ small enough so that $\gamma \binom{n}{2} < 1$, with the consequence that every component of size n would be returned as a cluster (Lemma 10). Since a component can contain a cut edge, this would contradict the assumption that $f(n) = 2$, so that the min cut size is at least 2.

The consequence of this observation is that the connectivity guarantee provided for CPM(γ) depends on γ, and that small values for γ allow for large clusters with cut edges being returned.

Discussion

Relationship to other work

The closest related paper is [9] by van Laarhoven and Marchiori, which addressed axiomatic properties of graph clustering methods based on optimization problems when the graph has non-negative edge weights. This study considers several axioms, including Locality, Continuity, Monotonicity, Relative Monotonicity, Richness, Permutation Invariance, and Scale Invariance, most of which are axioms proposed in [1, 5]. When edge weights are always either 0 or 1 and the network is an undirected simple graph, then satisfying the Relative Monotonicity axiom implies satisfying our Standard Consistency axiom.

They study seven clustering methods, including Components-are-Clusters, CPM, Modularity, and two variants of Modularity (fixed scale and adaptive scale modularity). They establish that Modularity- and CPM-optimization satisfy Richness and Continuity, but CPM-optimization satisfies Locality and Modularity does not, thus showing an advantage to CPM-

optimization. However, adaptive scale modularity satisfies all the axioms they present, while CPM-optimization and Components-are-Clusters each fail one axiom (with Components-are-Clusters failing Continuity and CPM-optimization failing Scale Invariance).

Now consider Theorems 3 and 4 in [9], which establish that modularity violates monotonicity and relative monotonicity, respectively. To prove these theorems, they present directed graphs with edge weights on which Modularity violates these two axioms. Because the edge weights are not restricted to 0 and 1, these counterexamples are not directly relevant to our questions, which are focused on axioms for graph clustering given unweighted graphs. In addition, their graphs have self-loops and are directed, which also differs from our restricted attention to simple undirected graphs. Thus, the two studies address somewhat different questions, and each provides a unique and important perspective on Modularity.

Summary of theoretical results

Our evaluation of graph clustering methods with respect to the different axioms we posed provides insight into differences between the clustering methods. In particular, one noteworthy outcome of this study is that CPM(γ), i.e., optimizing under the Constant Potts Model, satisfies all seven axioms we study, for all $0 < \gamma < 1$. Hence, unlike Kleinberg's axioms (which were designed for distance-based clustering), there is *no impossibility theorem for clustering of simple unweighted graphs in the distanceless context* on our set of axioms.

On the other hand, every other method we studied fails at least two of the axioms. We also see that Modularity—one of the most well known clustering methods—fails every axiom other than Richness, IKC run in default mode fails every axiom, and IKC(no-mod) fails four axioms. In contrast to these properties of existing clustering methods, each of the toy clustering methods we studied satisfies at least five of the seven axioms. Thus, these axioms reveal differences between clustering methods, and provide potentially helpful guidance to users of clustering methods.

Our study also provides some insight into which axioms are very easy to meet, and which ones are more likely to distinguish between methods. For example, Table 1 shows that Richness is in general extremely easy to achieve, with only Nodes-as-Clusters and IKC run in default mode failing to meet this criterion. The axioms based on consistency (i.e., Standard Consistency and its two relaxations) distinguish between methods, with Modularity and IKC in its default setting failing, but CPM, IKC(no-mod), and the two "toy" clustering methods succeeding. Given that even the toy clustering methods satisfy this axiom, failure to achieve

Table 1. Overview of theoretical results.

Method	Richness	Std Consist.	Ref Consist.	Inter-Edge Consist.	Connect	Pair-of-Cliques	Fixed Point
Components	✓ [9]	✓	✓	✓	-	-	✓
Nodes	-	✓	✓	✓	✓	-	✓
CPM(γ)	✓	✓	✓	✓	✓	✓	✓ [12]
Modularity	✓ [9]	-	-	-	-	- [10]	- [12]
IKC	-	-	-	-	-	-	-
IKC(no-mod)	✓	-	-	✓	-	-	✓

For each clustering method, we show which axioms are satisfied, with a reference to the paper where the result was first established if not in this study. A ✓ indicates that the method follows the axiom and "-" indicates the method fails to follow the axiom. In CPM(γ), we assume γ (the resolution parameter) is arbitrary but fixed. IKC (no-mod) is the variant of IKC where the requirement that non-singleton clusters have positive Modularity is dropped. The first four axioms are modifications of Kleinberg's axioms for the distanceless case, the next two are new axioms we introduce, and the final one is our name for the property referred to as "resolution-limit-free" from [12].

consistency can be seen as a clear indication of a weakness for Modularity and IKC in its default setting. The results for Connectivity on the other hand show that only CPM and Nodes-as-Clusters satisfy the axiom, revealing a basic weakness for all the other methods.

The resolution limit

The resolution limit, first established by Fortunato and Barthélemy in [10] for Modularity optimization, was described in terms of having an optimal clustering failing to find communities (i.e., sets of nodes that had high modularity scores) that were contained in larger sets of nodes. The example that was given was a ring-of-cliques, i.e., a graph consisting of a set of n-cliques, each adjacent to two other cliques by single edges, so they formed a ring. Fortunato and Barthélemy showed that as the number of cliques increased, an optimal clustering using Modularity Optimization would return two or more of the cliques for a given cluster, rather than the single cliques. That Modularity would fail to return the cliques as the communities was clearly interpreted as a strong limitation of the method.

There are two somewhat separable aspects of the Resolution Limit as described by Fortunato and Barthélemy in [10]: one is that under some conditions, the output set of clusters will not contain any clusters below some size (which may depend on the method), and the other is that there can be obvious communities that ought to be returned by the method, that fail to be returned.

Traag et al. [12] posed a property, which we refer to as the Fixed Point Axiom, to address the Resolution Limit. To satisfy this property, a clustering method will not change the output when applied to a single cluster or to the subnetwork induced by the vertices in any set of clusters that are returned. In [12], any method that satisfied this property was said to be "resolution limit free".

Our study shows that the Fixed Point Axiom was often satisfied by the clustering methods we examined, and even by the two toy methods Components-are-Clusters and Nodes-are-Clusters. Thus, clustering methods that produce clusters that are too small (i.e., Nodes-are-Clusters) or too large (i.e., Components-are-Clusters) can both satisfy the Fixed Point Axiom, indicating that this axiom does not address the first of the two aspects of the Resolution Limit we identified. Furthermore, neither of the two toy methods is able to detect cliques as the true clusters when they are properly contained in components within the network; hence, the Fixed Point Axiom does not address the second aspect we identified. In other words, the Fixed Point Axiom does not adequately characterize methods that satisfy the two objectives of being resolution-limit-free, according to our interpretation of the findings in [10].

Given how the Fixed Point Axiom does not adequately address the Resolution Limit issues as identified in [10], we formulated a simple test, called the "Pair-of-Cliques" axiom. We say that a method satisfies the Pair-of-Cliques axiom if any time the network contains a component that has two sufficiently large cliques connected by an edge (where the minimum size depends on the method), it will return the individual cliques. We found that of the clustering methods we examined, only CPM-optimization satisfied the Pair-of-Cliques axiom. Moreover, as shown in Table 1, four methods satisfy the Fixed Point Axiom but fail the Pair-of-Cliques axiom. This shows that the two axioms—Pair-of-Cliques and Fixed Point—are very different from each other, although both aim to address the Resolution Limit.

Part of the focus of the study [9] by van Laarhoven and Marchiori is the resolution limit, and in particular the Fixed Point Axiom proposed by Traag et al. [12]. They propose a new axiom, Locality, and discuss its relationship to the Fixed Point Axiom (showing it is both stronger in some ways and weaker in others). They define adaptive scale modularity as a modification to Modularity and prove that it satisfies Locality. However, they prove that Locality is

also satisfied by Components-are-Clusters, and it is easy to see that it is satisfied by Nodes-are-Clusters, each of which fails the Pair-of-Cliques axiom. Thus, like the Fixed Point axiom, their Locality axiom does not fully address the issues raised in [10] about the Resolution Limit.

Conclusion

Motivated by [1], which established impossibility theorems for clustering when the input is an $n \times n$ distance matrix, we examined the question of axiomatic clustering when the input is a simple unweighted graph without a corresponding distance matrix. We introduced seven axioms for distanceless graph partitioning, with four based on Kleinberg's axioms. We established that unlike Kleinberg's axioms, there is no impossibility theorem for our axioms. Moreover, we showed that optimizing under the constant Potts model (CPM), the default criterion for the Leiden software [14], one of the most popular methods for large-scale graph partitioning, has stronger theoretical guarantees than the other clustering methods we examined.

The results here are focused on theoretical properties of methods, but they also shed light on empirical performance. For example, satisfying connectivity depends only on presenting *some* function *f* so that all clusters of *n* nodes have min cuts greater than size *f(n)*. In our proof that CPM(γ) satisfies connectivity, the function *f* we provided depended on γ, with the consequence that it provides a very weak bound when γ is small. The dependence on γ is investigated in greater depth in Lemma 10, where we showed that for a given network *N*, γ can be chosen small enough so that the clusters are the components of *N*. This theoretical weakness is also reflected in empirical studies, as observed by [13], which demonstrated that using Leiden for CPM-optimization with very small values for γ resulted in relatively sparse clusters that can be poorly connected, and can even be trees. [12] also presents a discussion of this issue for its impact on CPM-optimal clustering. Given that in practice small values for γ are often used in order to achieve high node coverage, this a non-trivial issue (see discussion in [13]). In other words, while we do prove that Leiden-CPM satisfies Connectivity for all settings of the resolution parameter, the theoretical analysis shows that the guarantee is *weaker* for small values of the resolution parameter than for large values of the resolution parameter, and so provides some insight into performance in practice.

Our study also revealed that the concerns raised in [10] regarding the resolution limit are not fully addressed by the definition of "resolution-limit-free" given in [12]. Our simple "pair-of-cliques" axiom is an initial step towards investigating the resolution limit for clustering methods, but only gives one simple case that should be checked. A more complete analysis is needed, but this is challenging since at the heart of the resolution-limit is the concept that *some communities are clear*, so that recovering them must be achieved by a good clustering method. Unfortunately, characterizing what constitutes an obvious community is difficult, since defining these based on (say) having a positive modularity score is clearly insufficient. Thus, this is another direction for future work.

We leave several questions for future research. Other graph partitioning methods beyond Modularity, CPM, and IKC, should be evaluated for their axiomatic properties. For example, variants of Modularity optimization that take a resolution parameter have been developed [21], and these should be studied for their axiomatic properties. In addition, variants of graph partitioning methods that enforce edge-connectivity, as studied in [13], should also be considered. Specifically, [13] presented the *Connectivity Modifier*, an approach for modifying an existing output clustering to ensure that all clusters are well-connected, according to a user-specified lower-bound on the minimum edge cut size for a given cluster. Such a modification, paired with (say) CPM-optimization (in which γ is not fixed in advance) might lead to new clustering methods that have strong theoretical properties. It is easy to see that this

modification would ensure that the clustering algorithm satisfies the Connectivity axiom and would not change whether the method satisfied Richness, but it is less clear that the modified method would still satisfy Standard Consistency, Refinement Consistency, or Inter-edge Consistency. Other directions for future research include considering clustering methods that do not require that all nodes be in clusters, or methods that allow for overlapping clusters, including but not limited to nested clustering methods. Axioms suitable for such methods may be very different from the ones we have proposed here.

This study has focused on theoretical properties of clustering methods, and yet performance on real-world datasets is of the utmost importance. To what extent do the theoretical results in this study provide insight into possible empirical performance? As noted above, although Leiden-CPM satisfies connectivity for all values of the resolution parameter, the guarantee provided when the resolution parameter is small is a very weak one, and the guarantee only becomes a strong one when the resolution parameter is large. That theoretical property motivated an empirical study [13] into the edge connectivity of clustering methods on real-world datasets, which confirmed that Leiden-CPM produced poorly connected clusters. The study also showed that the frequency of poorly connected clusters was very low for very large resolution parameters and increased as the resolution parameter decreased. Another finding in that study is that Leiden optimizing modularity produced many poorly connected clusters, even though there is no theory yet to explain this. Thus, theoretical properties can suggest performance differences that need to be evaluated on real-world datasets, and performance on real-world datasets in turn can suggest directions for theoretical research. Clearly, much work is needed to evaluate methods on real-world networks with respect to empirical properties, such as edge-connectivity and the impact of deleting or adding edges, and that is an important direction for future work.

Acknowledgments

Author Contributions

Conceptualization: James Willson, Tandy Warnow.

Formal analysis: James Willson, Tandy Warnow.

Supervision: Tandy Warnow.

Writing – original draft: James Willson, Tandy Warnow.

Writing – review & editing: James Willson, Tandy Warnow.

References

1. Kleinberg J. An impossibility theorem for clustering. Advances in Neural Information Processing Systems. 2002; 15.

2. Ackerman M. Towards theoretical foundations of clustering. University of Waterloo; 2012.

3. Zadeh RB, Ben-David S. A uniqueness theorem for clustering. arXiv preprint. 2012; p. 1205.2600.

4. Cohen-Addad V, Kanade V, Mallmann-Trenn F. Clustering redemption—beyond the impossibility of Kleinberg's axioms. Advances in Neural Information Processing Systems. 2018; 31.

5. Ben-David S, Ackerman M. Measures of clustering quality: a working set of axioms for clustering. Advances in Neural Information Processing Systems. 2008; 21.

6. Schaeffer SE. Graph clustering. Computer Science Review. 2007; 1(1):27–64. https://doi.org/10.1016/j.cosrev.2007.05.001

7. Fortunato S. Community detection in graphs. Physics Reports. 2010; 486(3-5):75–174. https://doi.org/10.1016/j.physrep.2009.11.002

8. Bader DA, Meyerhenke H, Sanders P, Wagner D, editors. Graph partitioning and graph clustering, 10th DIMACS implementation challenge workshop. vol. 588 of Contemporary Mathematics. Providence, RI: American Mathematical Society; 2013.

9. Van Laarhoven T, Marchiori E. Axioms for graph clustering quality functions. The Journal of Machine Learning Research. 2014; 15(1):193–215.

10. Fortunato S, Barthelemy M. Resolution limit in community detection. Proceedings of the National Academy of Sciences. 2007; 104(1):36–41. https://doi.org/10.1073/pnas.0605965104 PMID: 17190818

11. Newman ME, Girvan M. Finding and evaluating community structure in networks. Physical Review E. 2004; 69(2):026113. https://doi.org/10.1103/PhysRevE.69.026113 PMID: 14995526

12. Traag VA, Van Dooren P, Nesterov Y. Narrow scope for resolution-limit-free community detection. Physical Review E. 2011; 84(1):016114. https://doi.org/10.1103/PhysRevE.84.016114 PMID: 21867264

13. Park M, Tabatabaee Y, Liu B, Pailodi VK, Ramavarapu V, Ramachandran R, et al. Well-connectedness and community detection. PLOS Complex Systems. 2024. In Press.

14. Traag VA, Waltman L, Van Eck NJ. From Louvain to Leiden: guaranteeing well-connected communities. Scientific Reports. 2019; 9(1):1–12. https://doi.org/10.1038/s41598-019-41695-z PMID: 30914743

15. Blondel VD, Guillaume JL, Lambiotte R, Lefebvre E. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment. 2008; 2008(10):P10008. https://doi.org/10.1088/1742-5468/2008/10/P10008

16. Brandes U, Delling D, Gaertler M, Görke R, Hoefer M, Nikoloski Z, et al. On modularity clustering. IEEE Transactions on Knowledge and Data Engineering. 2007; 20(2):172–188. https://doi.org/10.1109/TKDE.2007.190689

17. Wedell E, Park M, Korobskiy D, Warnow T, Chacko G. Center–periphery structure in research communities. Quantitative Science Studies. 2022; 3(1):289–314. https://doi.org/10.1162/qss_a_00184

18. Kannan R, Vempala S, Vetta A. On clusterings: Good, bad and spectral. Journal of the ACM (JACM). 2004; 51(3):497–515. https://doi.org/10.1145/990308.990313

19. Zhu ZA, Lattanzi S, Mirrokni V. A local algorithm for finding well-connected clusters. In: International Conference on Machine Learning. PMLR; 2013. p. 396–404.

20. Belyi A, Sobolevsky S. Network size reduction preserving optimal modularity and clique partition. In: Computational Science and Its Applications–ICCSA 2022: 22nd International Conference, Malaga, Spain, July 4–7, 2022, Proceedings, Part I. Springer; 2022. p. 19–33.

21. Lancichinetti A, Fortunato S. Limits of modularity maximization in community detection. Physical Review E—Statistical, Nonlinear, and Soft Matter Physics. 2011; 84(6):066122. https://doi.org/10.1103/PhysRevE.84.066122 PMID: 22304170