

Community detection in network neuroscience

Richard F. Betzel^{1–4*}

¹Department of Psychological and Brain Sciences,

²Cognitive Science Program, ³Program in Neuroscience,

⁴Network Science Institute, Indiana University, Bloomington, IN 47405

(Dated: November 16, 2020)

Many real-world networks, including nervous systems, exhibit meso-scale structure. This means that their elements can be grouped into meaningful sub-networks. In general, these sub-networks are unknown ahead of time and must be “discovered” algorithmically using community detection methods. In this article, we review evidence that nervous systems exhibit meso-scale structure in the form of communities, clusters, and modules. We also provide a set of guidelines to assist users in applying community detection methods to their own network data. These guidelines focus on the method of modularity maximization but, in many cases, are general and applicable to other techniques.

INTRODUCTION

The human brain is a complex network, comprised of neurons, neuronal populations, and brain areas linked to one another by structural and functional connections [1, 2]. This network can be expressed as a graph of nodes and edges and analyzed using methods from network science [3, 4] (Fig. 1a). Analyses of human brain networks have begun to reveal the key organizational principles of brain networks, including small-world architecture [5, 6], presence of hubs and rich clubs [7, 8], and cost-efficient wiring diagrams [9].

Many network analyses use graph-theoretic measures to describe features of a brain network [10]. Some measures focus on properties of the brain at a *local* scale (Fig. 1b), characterizing the number of connections a node makes, its centrality to the network, or the extent to which its neighbors are also connected to one another. Other measures describe *global* properties of the network as a whole (Fig. 1b), characterizing its paths and cycles to reveal how information flows through the system.

Both local and global scales represent extremes. Situated between the two is a rich topological *mesoscale* that focuses on groups of nodes that are sometimes called sub-networks, sub-graphs, communities, or modules [11, 12] (Fig. 1b). A network’s community structure is of broad interest scientifically (Fig. 2). It represents a condensed, coarse-grained view of a network by shifting focus onto clusters rather than individual elements (Fig. 2e). It can even be used to further characterize the functional roles of individual nodes (Fig. 2f), for example as connector hubs whose connections form bridges between two communities, or to discover groups of nodes and edges that likely participate in similar network functions (Fig. 2g).

Importantly, in the context of brain networks, the methods used to study a network’s community structure can be applied profitably to both structural [13] and functional connectivity [14] (and even used to explicitly link

the two modalities [15]). Brain network communities also represent a potentially powerful and flexible substrate for biomarker generation [16, 17] and to help discovering links between brain, behavior, and disease [18].

While a network’s community structure can offer insight into its organization and function, it is usually unknown ahead of time and cannot be inferred from visual inspection alone. Rather, a network’s communities must be detected algorithmically through a process known as *community detection* [19, 20].

This article focuses on the topic of community detection from a practical, user-oriented perspective. However, its aim is not to provide a definitive roadmap for community detection. Rather, it is to convince the reader that community detection is a fundamentally difficult process that is guided by the user, who, even for the “simplest” algorithms, must make a series of non-trivial decisions. On one hand, these decisions present challenges to the user. This is especially true for researchers new to network neuroscience who might be running community detection for the first time. On the other hand, these decisions afford the user some agency, making community detection a flexible framework that can be directed toward a wide range of research questions.

BACKGROUND

Communities refer to meaningful sub-networks – also referred as modules or clusters – embedded within a larger global network [21]. Although sub-networks can form core-periphery and multi-partite structures, they are usually assumed to be “assortative,” meaning that the nodes in a given community tend to be strongly connected to one another but weakly connected to nodes in other communities. This type of structure helps support autonomous, specialized function [22], buffer the impact of damage away from the rest of the network [23], reduce the material and metabolic cost of wiring [9], and support separation of dynamic timescales [24]. Assortative communities also promote a network’s evolvability [25], allowing individual communities to explore phenotypic

* rbetzel @ indiana.edu

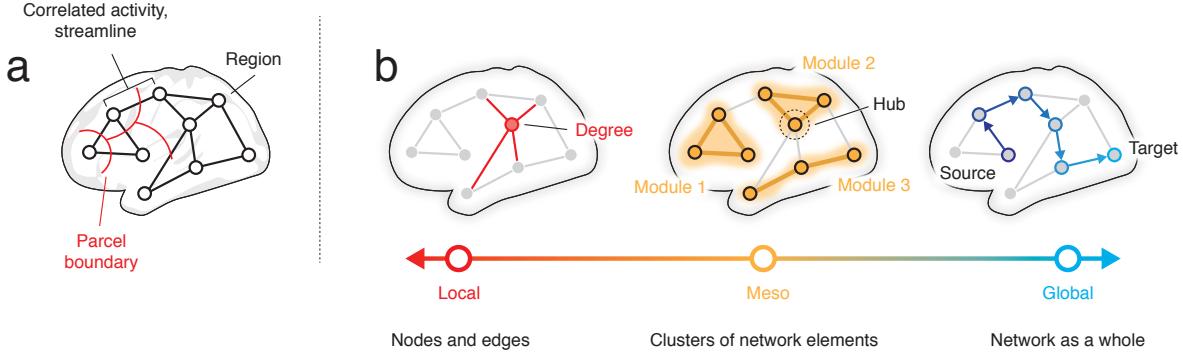


FIG. 1. Scales of network analysis. (a) Brain networks are usually defined as collections of neural elements (cells, populations, regions) linked to one another *via* structural or functional connections. (b) Analysis of brain networks can be carried out at the local and global scales, which focus on properties of individual network elements (nodes and edges) or the network as a whole, respectively. Situated between these extremes is a *meso-scale* that focuses on sub-networks, i.e. groups and clusters of nodes or edges.

variation without compromising the broader functionality of the network as a whole.

Indeed, modular structure has been observed in empirical brain network data, both in structural [7] and functional [14] brain networks reconstructed from virtually all imaging and recording modalities and across every spatial scale [26–30]. From the whole-brain neuronal connectomes of *C. elegans* [31], zebrafish [32, 33], and drosophila [29], to populations of neurons [34–36], to fast electrophysiological recordings [27, 37], to large-scale human brain networks [38] – wherever neuroscientists search for communities, communities tend to appear.

The analysis of community structure has had its greatest impact in the neuroscientific community at the macroscale, where brain networks are reconstructed from functional and diffusion MRI [39]. Early highly influential studies applied community detection algorithms to whole-brain resting-state functional networks to reveal non-overlapping modules whose boundaries, surprisingly, were closely aligned with previously delineated brain systems and task-evoked patterns of activity [14, 38]. These modules, which bore a striking resemblance to maps of correlated activity recovered from independent component analyses [40], mapped to distinct cognitive domains [41], and are recapitulated in meta-analytic networks of task co-activations [42]. These observations suggest that, even in the absence of explicit task instructions, the brain’s intrinsic architecture is modular and subtends cognitive function.

Similar findings have been reported in networks of anatomical connectivity, although the correspondence of functional and anatomical modules is inexact and not one-to-one. For instance, in Hagmann et al [7], the authors reported six modules: two located along the midline and four distributed in either hemisphere. In contrast to functional modules, which are spatially distributed and

oftentimes comprised of multiple extended sub-modules, anatomical modules tend to be spatially contiguous, reflecting the effect of material and metabolic constraints on the brain to reduce its wiring cost [43], a tendency that has been replicated in recent studies [13, 44].

Collectively, evidence abounds that nervous systems are modular. These modules span multiple spatial scales and network modalities, revealing the brain’s rich multi-scale and hierarchical communities.

COMMUNITY DETECTION: METHODS DESCRIPTION

In most applications, a network’s community structure is unknown ahead of time and cannot be inferred based on visual inspection alone. Instead, we use algorithms and heuristics to partition nodes and edges into sub-networks, called communities, modules, or clusters. This enterprise is referred to as “community detection” and involves many steps [19, 20]. Because the space of community detection methods is paralyzingly massive, it is impossible to discuss every method and contingency as part of this article. In the following sections we break down some of the common steps for performing community detection and decisions that a user has to make, focused through the lens of the *modularity heuristic*.

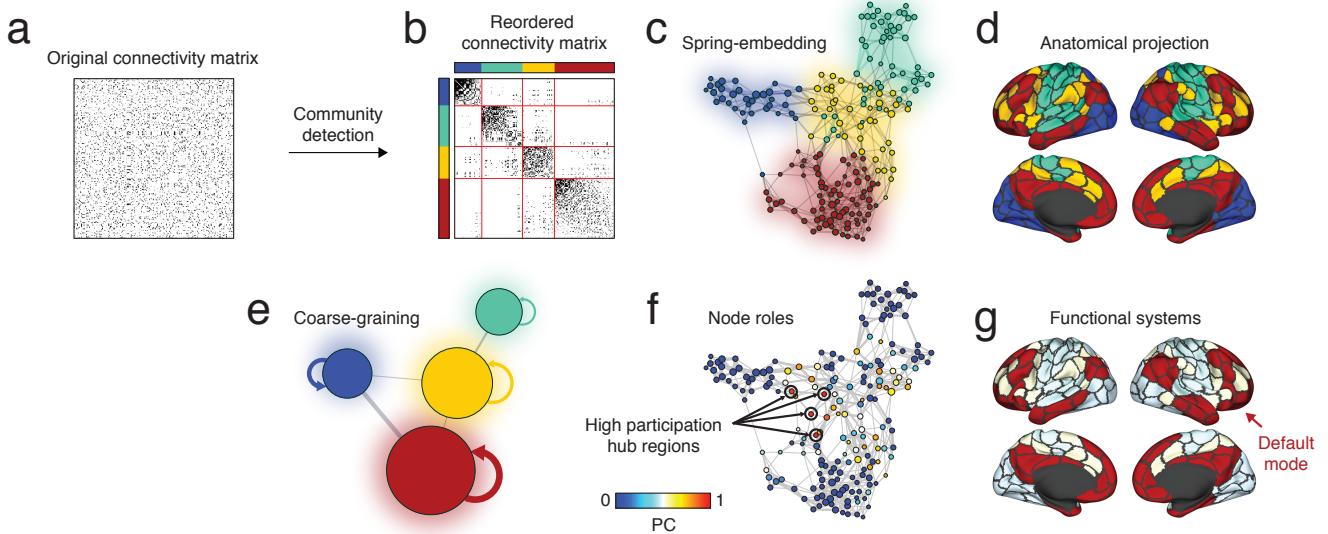


FIG. 2. Schematic illustrating why community detection is necessary and what can be gained from it. (a) Community detection usually starts with a network's connectivity matrix. Initially, no community labels are known and its rows and columns are in an arbitrary order, making it difficult to establish whether it has communities. (b) Community detection algorithms can be applied to this matrix to reveal its meso-scale structure. Here, we show the original matrix with its rows and columns reordered to highlight the “block diagonal” structure of assortative communities. We can visualize these communities through a spring-embedding (c) or by projecting node's community labels onto brain anatomy (d). (e) The advantage of this approach is that it allows us effectively reduce the dimensionality and complexity of a network by treating its communities rather than its nodes as the units of interest. (f) With community labels, we can classify nodes' roles by describing how each node's links are concentrated within or distributed across communities. (g) In the case of brain networks, community labels can be used to identify functionally-related groups of nodes, e.g. communities that recapitulate known intrinsic connectivity networks like the default mode.

Choosing an algorithm

Box 1. Different algorithms define communities in different ways and can yield dissimilar estimates of your network's community structure. For instance, modularity maximization is designed to detect internally dense and external sparse assortative communities. Stochastic blockmodels can detect more general classes of communities. Care should be taken in selecting the algorithm.

The first step in community detection is to choose an algorithm. This step is critical – every algorithm operates according to a different definition of what it means for a group of nodes or edges to form a community. Additionally, some algorithms require first transforming your network in some way, e.g. by discarding or rectifying negative connection weights. More generally, algorithms encode a set of assumptions about how your network was generated and what its communities look like. Consequently, the choice of algorithm has profound impact on the character of the detected communities.

Methods like Infomap [45], for instance, define communities to be collections of nodes that constrain the

probabilistic flow of a random walker moving over the network. As a result, Infomap detects communities that are “assortative,” with connection density concentrated within communities compared to between. However, because Infomap is based on random walks which are only defined for networks with positively-weighted edges, all negative weights must be first thresholded.

In contrast, stochastic blockmodels [46, 47] use statistical inference to recover the parameters (including community labels) of the generative model that gave rise to an observed network. In the classical blockmodel, every node belongs to a community and edges are generated stochastically based on some underlying distribution that depends only on the community to which an edge's stub nodes belong. Consequently, blockmodels can detect more general classes of communities; including core-periphery and disassortative configurations (where between-community connection density exceeds the within-community density) [48, 49].

Both Infomap and the stochastic blockmodel assign nodes to non-overlapping communities, meaning that each node is assigned to one community and one community only. However, in many real-world networks nodes belong to multiple communities. Methods like link communities [50], line graph clustering [51], and clique percolation [52] are capable to detecting overlapping commu-

nities although, as before, the precise definition of what constitutes a community varies between these methods.

The modularity heuristic

Box 2. Modularity maximization is a widely used community detection method. It defines communities as groups of nodes that are more densely connected in your observed network than you would expect by chance. The Q value associated with a partition – its modularity – is a measure of its quality. Larger values usually imply higher quality partitions.

Among the most widely used community detection algorithms is modularity maximization [53]. Modularity maximization operates according to an eminently simple principle: compare what you see with what you expect. More specifically, modularity defines communities to be groups of nodes more densely connected to one another than would be expected had the network been generated by a random null model. This intuition is formalized by the modularity heuristic, Q , which serves as an objective function for quantifying the quality of any partition and is calculated as:

$$Q = \sum_{ij} B_{ij} \delta(\sigma_i, \sigma_j). \quad (1)$$

In this expression, B_{ij} is an element taken from the *modularity matrix*, which is computed as the difference in each connection's observed weight, W_{ij} (Fig. 3a), with the weight we would expect it to have under the null model, P_{ij} (Fig. 3b), such that $B_{ij} = W_{ij} - P_{ij}$ (Fig. 3c). The variable $\sigma_i \in \{1, \dots, K\}$ encodes the community to which node i is assigned, and $\delta(\sigma_i, \sigma_j)$ is the Kronecker delta function and is equal to 1 when its arguments are identical and zero otherwise. Although the summation is carried out over all pairs of nodes, the delta function ensures that the only node pairs that increase or decrease Q are those that fall within the same community. In general, partitions corresponding to larger values of Q are considered “better” than those with smaller values.

Null models

Box 3. Modularity, Q , is calculated by comparing your real network with the network you would expect given some a null connectivity model. The choice of null model is critical and should be motivated by research question and properties of your network, e.g. structural *versus* functional connectivity or signed *versus* positive-only connection weights. In general, different null connectivity models will yield dissimilar estimates of community structure.

The modularity, Q , of a partition depends on the structure of our observed network, W , as well as the network we would expect by chance – the matrix P in the modularity expression. This presents a challenge, and in most applications, one of the first instances where the user needs to make a decision. What is the appropriate null model for a network? What is the appropriate null model for a *brain* network? Are there special considerations that need to be taken given how we define our observed network?

Before we address those questions, it is important to explain exactly what the null model does and (does not) represent. In network analyses it is common to compute a summary statistic on a network, e.g. its global efficiency [54], and compare that value with the distribution of values obtained under some null model, oftentimes one that preserves a set of features from the observed network while randomizing others. The aim of this comparison is to estimate the likelihood that we would obtain the observed efficiency value amongst an ensemble of similar but structureless networks. In the context of modularity maximization, however, the null model serves as an internal comparison and is baked into the modularity function itself. Note, however, that statistical testing can still be carried out after the fact and Q values compared between the observed and random networks.

So what is the appropriate null model for modularity maximization? In practice, most applications use a model that preserves nodes' degrees but otherwise randomizes the placement of links [55] (Fig. 3b). In fact, in most software packages, this model is treated as the “default,” oftentimes leaving the user no other options. Part of why this model is used so widely is because it is useful – by preserving nodes' degrees, it ensures that high-degree nodes cannot spuriously inflate the value of Q . However, it also has several limitations that make it less appropriate for brain networks.

Notably, the degree-preserving null model – also known as the Newman-Girvan model or configuration model – creates random networks through unconstrained edge swaps, resulting in networks with many more long-distance connections. In contrast, real-world anatomical brain networks are constrained by cranial volume and connectional cost (material and metabolic) such that the

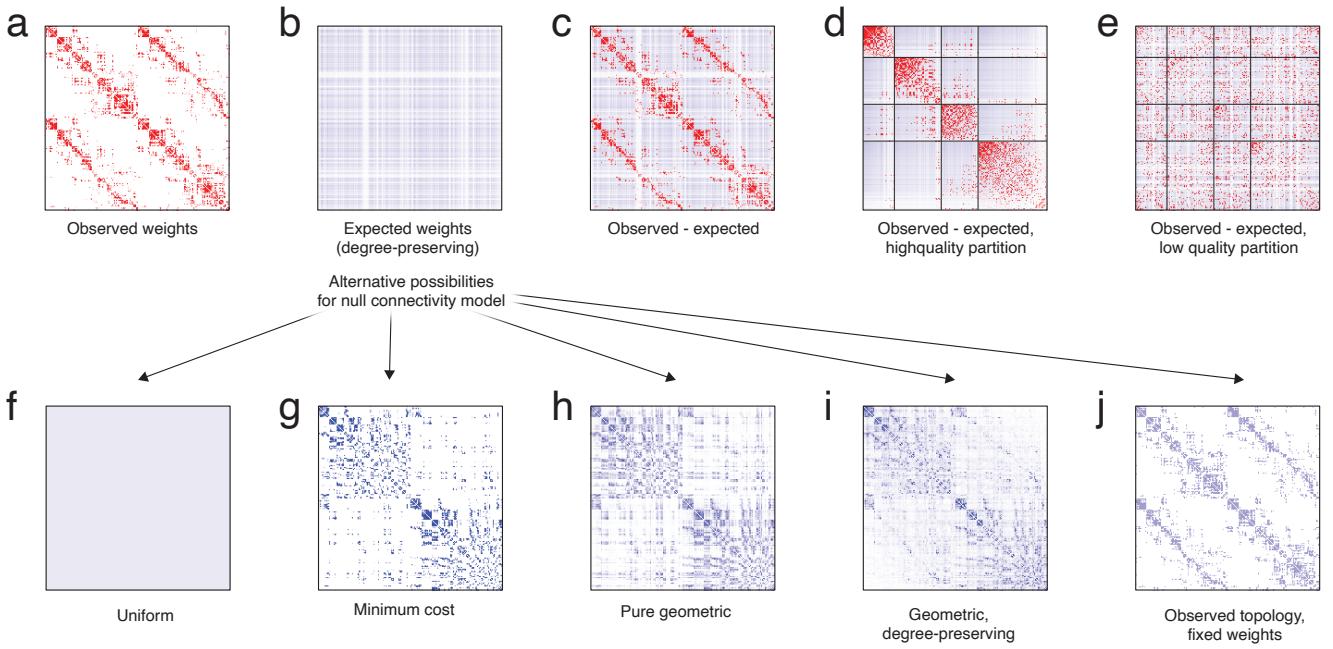


FIG. 3. Modularity maximization and choice of null model. Modularity maximization works by comparing weights in an observed network (*a*) with the weights expected under a null connectivity model (*b*). A modularity matrix is constructed by literally subtracting each edge’s expected weight from its observed weight (*c*). Modularity maximization algorithms aim to select clusters such that the stronger-than-expected edges fall within communities. In panels *d* and *e* we show example partitions, one that concentrates strong connections within communities and would be considered of high quality, and another that does not. The detected communities depend on the user’s choice of null connectivity model. In *b* we show expected connection weights under a configuration (degree-preserving) null model, which is common in the network neuroscience literature. However, other null models may be more appropriate under certain circumstances. For instance, one can consider a *uniform* model (*f*) in which all connections are assigned an equal weight, a *minimum cost* model (*g*) in which the m shortest connections are formed and their weights assigned inversely proportional distance, a *geometric* model (*h*) where connections and their weights are assigned probabilistically based on distances from one another, a *geometric and degree-preserving* model (*i*) in which both the wiring cost and degree sequence of the original network are preserved, and a *fixed topology* model (*j*) that preserves the exactly configuration of connections as in the original network but where weights are assigned uniformly.

overwhelming majority of their connections are short-range [43, 56–58]. In violating this fundamental constraint, degree-preserving null models can be viewed as excessively lenient.

A second related issue concerns dependencies between connections. The connection weights in functional brain networks, because they are calculated using statistical measures like bivariate correlations, are not independent of one another. That is, if we have three nodes $\{A, B, C\}$ and know the weights of edges w_{AB} and w_{BC} , we can place a bound on the weight of edge w_{AC} [59]. However, the networks generated by the edge-swapping model shuffle the placement of edges uniformly and randomly, resulting in configurations that violate statistical limits and are not realizable.

Despite these limitations, the configuration model (and versions of the model made compatible to networks with signed edges [60]) remains the “default” null model for modularity maximization. There exists many alternative options that help resolve these issues (Fig. 3g-j). For in-

stance, it is straightforward to condition “edge swaps” in the configuration model to occur among edges of similar length, which results in a network with approximately the same wiring cost as an observed structural network [61, 62]. In the case of functional connectivity, several approaches exist that address the issue of edge interdependencies. For instance, the uniform null model can be used [63, 64], which tests the null hypothesis that all time series are mutually correlated with one another to some arbitrary extent. Of course, more complicated versions of null models can be imagined, including some based on random matrix theory or [65, 66] others that merge both spatial and correlative information [67] or preserve topology (the binary pattern of connections) while randomly assigning weights to each edge [68].

In general, the choice of null model can impact a partition’s quality (larger or smaller values of Q) (Fig. 3d,e). Historically, however, the network neuroscience literature has focused on a single null model and the landscape of alternative models has not been fully explored. Of course,

exploring the space of null connectivity models falls beyond the scope of most studies. In those cases, selection of a null model should balance continuity with past work, so as to facilitate clear comparisons, but strive to incorporate more appropriate null models whenever possible.

Algorithms for optimizing Q

In general, Q measures the quality of a modular partition – partitions corresponding to larger values of Q are generally considered better. While this makes it possible to perform comparative analyses of existing, previously-defined partitions, it also opens up the possibility of detecting even better partitions by optimizing Q directly, a procedure referred to as *modularity maximization*.

Although many optimization heuristics have been considered in the past, including simulated annealing [69], spectral clustering [70], and min-cut algorithms [71], each of which varies both in terms of runtime as well as ability to recover planted communities in synthetic networks, the most popular method is the Louvain algorithm [72].

The Louvain algorithm is a greedy, two-stage algorithm that scores well on benchmark tests and, in most practical contexts, runs exceptionally fast [73]. The Louvain algorithm is initialized with each node assigned to its own community (although can also begin with a “seed” partition); if a network has N nodes, then the number of communities at this stage is equal to N . The algorithm then loops over nodes in random order and considers how Q would change if the current node was moved to all of the other communities. The algorithm then accepts the move that results in the greatest ΔQ , subject to the constraint that $\Delta Q > 0$. The loop continues over all nodes and repeats until no node-level moves exist that can improve Q . At this point, the algorithm moves to the aggregation stage, in which a meta-network is constructed where each meta-node represents a community from the previous stage and each meta-edge represents that total weight of connections from one meta-node to another. Then, the algorithm returns to the move/merge stage with the meta-network, and attempts to merge meta-nodes into larger communities. In the end, the Louvain algorithm returns community labels for each node, along with the corresponding modularity score, Q .

The Louvain algorithm does have limitations, some of which can be addressed with recent extensions. For instance, the Louvain algorithm can lead to poorly-connected communities, meaning that the nodes comprising a community are not directly connected, but only linked to one another by extra-community paths. The recently proposed “Leiden algorithm” [74] addresses this and guarantees well-connected communities while preserving the speed and parsimony of the original Louvain algorithm. Nonetheless, the Leiden algorithm was introduced only recently and has yet to be applied in network neuroscience.

Stochasticity and consensus clustering

Box 4. Many community detection algorithms, including the Louvain algorithm for optimizing Q , are non-deterministic such that repeated runs can yield dissimilar results. This variability can be resolved using consensus clustering methods.

While the Louvain algorithm performs well in practical contexts, it is a greedy and stochastic algorithm, which means that successive runs of the algorithm can (and in general will) result in different estimates of the optimal partition [75] (Fig. 4a). Some studies have embraced this variability, generating “soft” partitions of a network by creating a community co-assignment matrix from multiple runs of the algorithm whose elements represent the fraction of those runs in which pairs of nodes were assigned to the same community [76, 77].

On the other hand, it is often advantageous to summarize a network’s community structure in terms of a definitive, representative partition. There are several strategies for selecting a single partition. One possibility is to simply select the partition with the biggest Q – after all, Q is a measure of partition quality. While this strategy seems appealing, the near-degeneracy of solutions means that there may exist dissimilar partitions but whose Q values differ by some arbitrarily small amount [75]. Additionally, given the variability inherent in brain network construction and the noise associated with individual edges and their weights, adopting one partition over another due to a small advantage in Q should be avoided.

A better strategy is to create a representative partition by synthesizing data from multiple partitions, a procedure called consensus clustering. Although there are many methods for performing consensus clustering [78], the most popular in network neuroscience involves iteratively clustering a co-assignment matrix until convergence [79]. The procedure is as follows. Starting with an observed network, a user runs the modularity maximization some fixed number of times, N_{iter} , creating an ensemble of high-quality but dissimilar partitions (Fig. 4b). From these partitions, one can generate a co-assignment matrix, calculating for nodes i and j the fraction of N_{iter} partitions in which they appeared together in the same community, D_{ij} (Fig. 4c).

Then, we set up a new modularity maximization problem by comparing the values in the observed co-assignment matrix with the values we would expect by chance. As before, the chance or null model must be chosen carefully, but one possibility is to compute the expected co-assignment had the community labels in each of the N_{iter} partitions been uniformly and randomly permuted [32, 57, 80] (note that instead of uniform permutations, one can use “spin” models to permute nodes’ while approximately preserving their relative distances in space [81, 82]). This expected level of co-assignment can be cal-

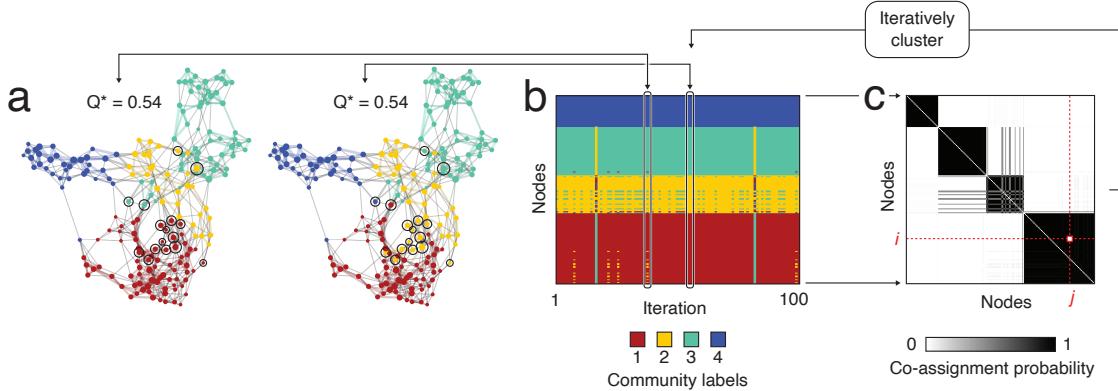


FIG. 4. Stochasticity and consensus clustering. The Louvain algorithm is stochastic, such that two runs can produce dissimilar results. Modularity maximization also exhibits a degeneracy of high-quality partitions, meaning that there are likely to be many different partitions of near optimal and equal quality (similar Q value). Consensus clustering is a strategy for combining the outputs of dissimilar partitions into a single, representative partition. (a) Two partitions of approximately equal quality ($Q = 0.54$) but different from one another. The dissimilar community assignments are highlighted in with black circles. (b) If repeated the Louvain algorithm 100 times, in general we find several different solutions. As part of consensus clustering, we generate a co-assignment or agreement matrix (c) from the outputs of multiple Louvain runs. The $\{i, j\}$ element of this matrix counts the fraction of partitions in which nodes i and j were assigned to the same community. This matrix can be clustered using modularity maximization by estimating the expected co-assignment of two nodes to a community under some null model (usually one in which nodes' assignments are uniformly and randomly permuted). Consensus clustering iterates between these two steps: constructing a co-assignment matrix from multiple partitions and clustering it. After a few steps, this algorithm converges so that every run of the optimization algorithm returns an identical partition.

culated directly from the observed co-assignment matrix as the mean of its upper triangle elements. This allows us to write the following modularity expression:

$$Q_{consensus} = \sum_{ij} [D_{ij} - \langle D \rangle] \delta(\sigma_i, \sigma_j). \quad (2)$$

As before, we can use modularity maximization to try and detect the partition that optimizes $Q_{consensus}$. Due to how it is constructed, the co-assignment matrix tends to be “blockier” than the original connectivity matrix and detecting its communities is an easier task. This leads to reduced variability in the partitions detected by optimizing $Q_{consensus}$, sometimes to the point that the clustering algorithm consistently returns an identical partition each time. If this occurs, then this partition is considered the consensus partition and the consensus clustering algorithm can stop. If there remains variability in the solutions, then a new co-assignment matrix can be created from the estimated consensus partitions and the algorithm repeats. Note that this procedure diverges slightly from the one proposed in [79], where the authors thresholded away elements of the co-assignment matrix below some critical value prior to reclustering.

In general, consensus clustering is a useful tool for resolving variability across multiple estimates of a network’s community structure that do not necessarily have to come from the same community detection algorithm. However, it entails some assumptions that a user should

be aware of. First, it assumes that there is a single representative set of communities for a network. While this a convenient assumption, recent studies have suggested that the landscape of solutions can be highly degenerate, with multiple near-optimal solutions. Similarly, in extracting consensus clusters, there is an assumption that every node must be assigned to a cluster – i.e. it is possible to arrive at a consensus about the community to which every node in the network belongs. However, it could be the case (especially if multiple “good” partitions exist) that some nodes fundamentally have no clear consensus community assignment [83]. In such cases, describing a single optimal partition is not appropriate; an alternative strategy is to describe, statistically, properties of the ensemble of *all* good partitions [84].

Multi-scale extensions

Box 5. Modularity maximization can be used to detect communities of different size and number by incorporating a resolution parameter into the modularity expression.

At this point, we have defined the modularity quality function, described some guidelines for selecting its internal null model, reviewed some of the algorithms for optimizing Q , and provided a solution for resolving its inherent stochastic nature, yielding (possibly) a single

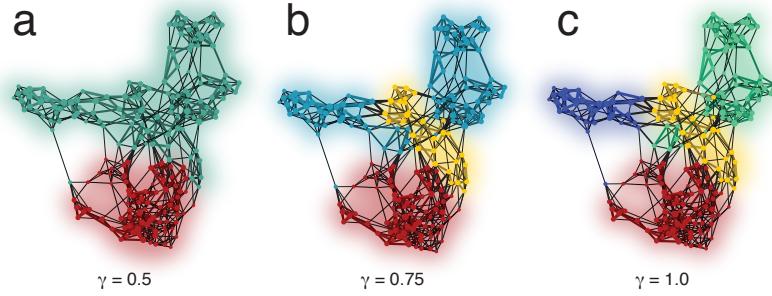


FIG. 5. Detecting multi-scale community structure using multi-resolution modularity maximization. Modularity maximization includes a tunable resolution parameter, γ , to detect communities of different size. In panels *a*, *b*, and *c*, we show three examples of communities in the same network detected at $\gamma = 0.5, 0.75, 1.0$, resulting in 2, 3, and 4 communities .

representative consensus partition. At this point, one might expect that the community detection process is over. Unfortunately, this is not the case.

Modularity maximization (and other related methods) can suffer from a limitation known as a “resolution limit” [85]. The resolution limit means that, for a network of given density, modularity maximization will be unable to detect communities below a certain scale, even if they are unambiguously defined, i.e. perfect cliques. This is because, for certain combinations of null models and networks, especially those that are sparse, the presence of even a single edge between two small communities is considered unexpected. Consequently, it becomes advantageous, in terms of maximizing Q , to merge those communities such that unexpectedly strong connection falls within communities, rather than between.

The implication of the resolution limit for modularity maximization is that the communities may not reflect the scale of your network’s true and intrinsic community structure and can therefore be misleading (the worst case) or obscure interesting community structure at other scales (a slightly less worse case). Luckily, this issue can be at least partially mitigated through the introduction of a resolution parameter, γ to the modularity function [86], so that it reads:

$$Q(\gamma) = \sum_{ij} [W_{ij} - \gamma \cdot P_{ij}] \delta(\sigma_i, \sigma_j). \quad (3)$$

Effectively, γ scales the importance of the null connectivity model relative to the weights of the observed network. Consequently, when γ is small, many connections exceed their expected weights. From the perspective of optimizing Q , this makes it advantageous to form large extended communities comprised of many nodes. Conversely, when γ is large, only the strongest connections exceed their expected weights, yielding much smaller communities and in greater number.

Incorporating the resolution parameter into modularity maximization makes it possible to change its value

and detect communities that might have been undetectable otherwise. It is important to note that, even if one chooses not to incorporate γ into their modularity function, it is nonetheless implicitly there, set to its default value of $\gamma = 1$. Additionally, it is worth noting that the particular value of 1 is not privileged or special. In general, it is a good idea to explore the effect of varying γ on your detected communities (Fig. 5a-c).

Additionally, the resolution parameter enables user to detect multi-scale or hierarchical communities – meaningful groupings of nodes and edges into possibly nested communities of different sizes. This is critical for the study of brain networks where we anticipate ahead of time that the such hierarchical divisions exist, e.g. a bipartition of the brain into task-positive and -negative communities, each of which can be further divided into increasingly functionally specialized sub-systems. Indeed, recent methods have been developed to explicitly address this; given a set of communities sampled at different scales, methods like multi-resolution consensus clustering aim to assemble communities into a natural hierarchy [87].

Multi-scale community detection, while useful for circumventing issues related to the resolution limit, nonetheless introduces a free parameter, γ . While embracing the multi-scale nature of communities may be appealing in some cases [15], in other applications it is more useful to establish meaningful communities at a single scale corresponding to a specific value of γ . How might one go about doing this? One possibility is to develop an additional quality function and report communities at the γ value where this quality function is optimized [88]. For instance, one could impose communities estimated using one subject’s data onto other subjects, and select the γ where the modularity induced by the one subject’s partitions is maximized. Another possibility is to select the γ value where repeated runs of the optimization algorithm yields highly similar partitions, suggesting that at that particular scale, there exists a single well-defined solution [89].

Node roles

Box 6. Once a network’s communities have been estimated, measures like participation coefficient and module-degree-zscore can be used to assess node roles within the network, e.g. hub *versus* non-hub.

The procedures described above can serve as guideposts for detecting a network’s community structure using modularity maximization. However, they are far from definitive; like other data clustering tools, modularity maximization is at times more art than science. Nonetheless, community detection, when done well, can reduce dimensionality and highlight patterns and structure in a network not otherwise evident. Another way that it can be useful is by providing insight into nodes’ functional roles within a network.

Two of the best-known measures for ascertaining a node’s function role are participation coefficient and module degree-z-score [69]. Each requires that the user know a network’s topology and have previously estimated its communities. The participation coefficient, in particular, has proven especially useful in network neuroscience, as it quantifies whether a node’s connections are distributed uniformly across many modules (high participation) or concentrated within a relatively small number (low participation) [90–92]. Intuitively, nodes with high participation serve as bridges, binding functionally-specialized modules to one another and regulating the flow of information across their boundaries.

Within-module degree-z-score is a complementary method, and describes how strongly a node is connected to its own module relative the other nodes within its module. Nodes with larger values are generally considered more influential among their neighbors. Together with participation coefficient, within-module degree z-score has been used for identifying hub regions and further classifying them based on whether they are “connector” hubs, linking communities to one another, or “provincial” hubs that exert local influence over their own module but little influence elsewhere.

Due to its widespread use, participation coefficient has been extended within the network neuroscience community in several ways. These include extending its definition so that it can be applied to networks with signed edge weights, but also modifying the definition to combat some peculiar biases. For instance, [93] demonstrated that nodes’ participation coefficients are influenced by the connectivity and density of the module to which that node was assigned and introduced a normalization factor to reduce this particular bias.

Limitations

Community detection can reveal a network’s mesoscale structure and, in the case of brains, their system-level architecture. Nonetheless, it has important limitations that users should be wary of. Modularity maximization, specifically, has a number of well-known issues that can limit its applicability or hinder interpretation.

One of the most common misunderstandings concerns the modularity function and how to interpret Q . In most applications, larger values of Q are usually treated as evidence that a network has segregated modules. In general, however, this is not true. Erdos-Renyi networks, where links are formed probabilistically and uniformly, can exhibit high levels of Q due to random fluctuations [94]. Relatedly, geometric networks, where links are formed based on the proximity of nodes to one another, can also exhibit artificially high levels of modularity, leading users to spuriously detect modules [58, 61]. In the case of geometric networks, the spuriously high levels of Q are a consequence of using a null model that does not depend on the network’s spatial embedding, e.g. a degree-preserving model, as geometric networks exhibit high levels of transitivity due to the underlying metric space in which they are embedded. A simple solution to combat this problem is to incorporate distances into the null model so connections that exist due to spatial effects can be anticipated [95].

Another issue associated with community detection, more generally, concerns the lack of a unique solution. Networks are generated according to some process that is a function of its ground truth, “true” community structure. Community detection, then, can be thought of as an attempt to invert this process to get at those communities. However, there is not a unique solution to this inversion [96]. That is, we can always define some arbitrary function that maps a different set of communities onto our observed network. Accordingly, unless we know the true function (and can therefore invert it), we can never be certain that the communities we detect represent the ground truth. These observations suggest that applications of community detection to real-world networks where the (community-to-network mapping is unknown), should test multiple community detection algorithms and focus on consensus, rather than emphasizing the results of any single approach.

Other concerns include lack of correspondence between “ground truth” community structure, e.g. affiliations obtained base on individual’s association with an online group, and the community structure detected based on network topology [97, 98]. One potential strategy for mitigating this particular issue is to incorporate metadata and vertex annotations directly into the community detection algorithm, which can be done easily with stochastic blockmodels [99], an approach that has also been applied to functional brain networks [100].

Yet other concerns are specific to the modularity heuristic. For instance, modularity maximization forces

every node to have a community assignment, a requirement that, in general, real-world networks are not subjected to (although some methods can reject the hypothesis that a node should be assigned to a community [101]). Relatedly, modularity maximization also forces communities to be non-overlapping, making it impossible for nodes to have multiple community affiliations. In the next section, we introduce some alternative methods that address these limitations.

Future directions and alternative approaches

Throughout this article we focused on community detection using modularity maximization and highlighted some of the decisions a user must make along the way. Modularity maximization, however, is but one method among a pantheon of methods for detecting a network’s community structure and returns communities of a specific character – non-overlapping, assortative subnetworks. Other methods, however, break free from these assumptions (although they bring along their own sets of assumptions), and can detect communities with vastly different properties. In this section, we briefly review some of the common alternatives to modularity maximization, focusing on those that have been applied to brain network data.

One of the simplest alternatives to “vanilla” modularity maximization is to extend it to multi-layer networks [103, 104]. Whereas single-layer modularity maximization can be used to detect communities in one connectivity matrix at a time, multi-layer modularity maximization can detect the community structure of systems composed of multiple connectivity matrices. These matrices could represent estimates of the same network made at different points in time [105, 106], networks from different individuals [107], or network structure under different task conditions [108]. The principle advantage of the multi-layer approach is two-fold. First, it resolves the practical issue of how to map communities from one network to another. In general, establishing a correspondence between communities in two networks is not straightforward. There are instances where the correspondence is fundamentally ambiguous, and resolving this ambiguity requires introducing additional heuristics or parameters. With multi-layer modularity maximization, communities are estimated for all layers (networks) simultaneously, such that community labels are preserved across layers, making it trivially easy to assess whether a community in one layer is the same as a community in another. Second, the preservation of community labels makes it possible to assess which communities change their network assignment from one layer to another, a measure referred to as “flexibility” [88], which has been linked to motor and reinforcement learning [105, 109], day-to-day variation in affective state [110], executive function [111], language [112], and clinical status [113]. In general, multi-layer modularity maximization serves as

a powerful extension of the traditional single-layer modularity and can be modified and made applicable in any context where community structure for multiple networks (layers) is desired.

Beyond modularity maximization, other popular methods for community include Infomap [45] and stochastic blockmodels [47]. Infomap detects communities based on the trajectory of a random walker moving over a network. Specifically, it aims to reduce an objective function known as the map equation [114], by assigning each node in the network as short of a “name” as possible (in information theoretic terms). If a network has communities that traps the probabilistic flow of a random walker, then instead of assigning each node a unique name, we can assign communities names and reuse the *very* short names for nodes, similar to the way ZIP codes are assigned to cities and street names are reused across different cities. Like modularity maximization, Infomap generates communities that are assortative, internally dense, and externally sparse and has been used widely within the network neuroscience community [14].

Yet another method is the stochastic blockmodel. Unlike modularity maximization and Infomap, blockmodels can detect more general classes of community structure, including core-periphery and disassortative (bipartite or multipartite) organization. Blockmodels operate under the assumption of structural equivalence – whether or not two nodes are connected (and the weight of that connection) depends only on the communities to which those nodes are assigned [47] – essentially grouping nodes together if they have similar connectivity profiles. Although blockmodeling is relatively new in network neuroscience [48, 49, 115, 116], it has a long history in the statistical and social sciences [46, 117], and continues to be the subject of intense research within network science [84, 118].

Modularity maximization, Infomap, and blockmodels generally assume that a network’s nodes form non-overlapping communities (although blockmodels can be extended to accommodate overlapping communities [119]). This assumption, however, disagrees with the observation that some brains regions flexibly reconfigure their community assignments over time, maintaining affiliations with multiple overlapping communities. There are, however, many methods that allow for overlap between nodes’ community affiliations, although they are relatively uncommon in network neuroscience. For instance, methods like clique percolation [52] and fuzzy k-means graph clustering [120] generate overlapping estimates of community structure. Others even extend well-established methods like modularity maximization to allow for overlap [121, 122].

Yet another approach for studying community overlap is to shift focus away from a node-centric methods that attempt to cluster a network’s nodes and onto edge-centric methods that partition connections into clusters. Because edges are assigned to different communities, the

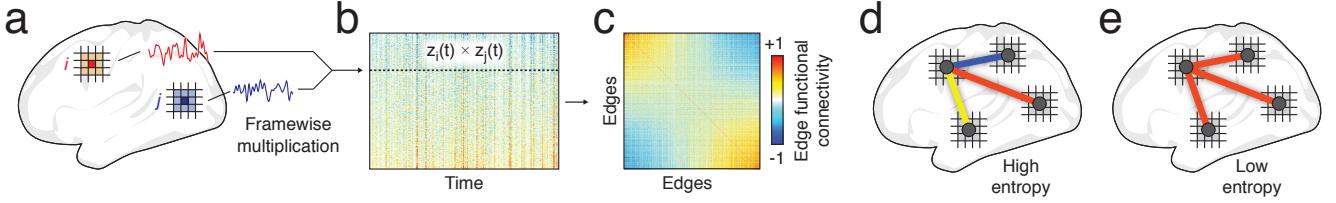


FIG. 6. **Edge functional connectivity and pervasively overlapping community structure.** (a) Detecting overlapping communities with edge functional connectivity starts by first generating “edge time series” as the element-wise product of z-scored regional time series. (b) This procedure is repeated for all pairs of regions. (c) Then, the pairwise similarity of edge time series is computed. The result edge-by-edge graph can be clustered. Because edges can take on different community assignments, the nodes upon which they are incident can be associated (*via* their edges) with multiple communities, leading to higher or lower levels of overlap from a node perspective. In panels d and e, we show examples of a high- and low-overlap scenario from the perspective of the node in the upper left. Following [102], we can quantify the level of community overlap at each node using an entropy measure.

nodes upon which those edges are incident can be associated with multiple communities at the same time. Methods like link similarity [50] and line graph clustering [51] operate this way, transforming node-to-node networks into edge-to-edge networks, which can be clustered using any of the traditional methods. Recently, a novel edge-centric method was proposed for generating overlapping estimates of communities in functional brain networks [102, 123]. This method works by generating co-fluctuation time series for every pair of nodes as the element-wise product of their z-scored activity [124] (Fig. 6a,b). We can calculate *edge functional connectivity* for every pair of time series as their similarity across time, generating a fully weighted and signed edge-by-edge matrix (Fig. 6c), which can be clustered directly to assign edges to communities, yielding overlapping communities at the level of nodes (Fig. 6d,e). Whereas other methods for detecting overlapping communities allow for a small number of nodes to participate in multiple communities [52, 122, 125, 126], edge communities exhibit pervasive overlap, meaning that every node participates in multiple communities by design.

OUTLOOK AND CONCLUSION

Community detection can be used to reveal a network’s community structure and provide useful insight into its organization and function. However, community detection also prove to be a complicated process that forces a user to define precisely what it means for a network’s elements to form a community and make critical decisions about parameters, optimization algorithm, and scale. These decisions make community detection challenging, yet endow users with the freedom to tailor the process to their specific research question.

SOFTWARE

1. Brain Connectivity Toolbox (<https://sites.google.com/site/bctnet/>):
 - `community_louvain.m`: Implements Louvain algorithm for optimizing Q .
 - `consensus_und.m`: Runs threshold-based consensus clustering.
 - `participation_coef.m` and `participation_coef_sign.m`: calculates nodal participation coefficient for networks with positive and signed connections, respectively.
 - `module_degree_zscore.m`: calculates nodal module-degree-zscore.
2. NetWiki (<http://netwiki.amath.unc.edu/GenLouvain/GenLouvain>): Includes function for running a generalized Louvain algorithm (`genlouvain.m`). This function allows the user to flexibly define their own modularity functions by changing null models and resolution parameters. Includes tutorials for implementing multi-layer and multi-resolution network models.
3. Network Community Toolbox (requires `genlouvain.m` function; <http://commdetect.weebly.com/>): Suite of functions for optimizing modularity, null models for significance testing, and for further characterizing single communities and entire partitions. Also includes flexibility coefficient.

ACKNOWLEDGEMENTS

Thanks to Haily Merritt, Jacob Tanner, and Joshua Faskowitz for reading an early version of this manuscript and providing feedback.

-
- [1] Olaf Sporns, Giulio Tononi, and Rolf Kötter, “The human connectome: a structural description of the human brain,” *PLoS Comput Biol* **1**, e42 (2005).
- [2] Hae-Jeong Park and Karl Friston, “Structural and functional brain networks: from connections to cognition,” *Science* **342** (2013).
- [3] Ed Bullmore and Olaf Sporns, “Complex brain networks: graph theoretical analysis of structural and functional systems,” *Nature reviews neuroscience* **10**, 186–198 (2009).
- [4] Danielle S Bassett and Olaf Sporns, “Network neuroscience,” *Nature neuroscience* **20**, 353–364 (2017).
- [5] Sophie Achard, Raymond Salvador, Brandon Whitcher, John Suckling, and ED Bullmore, “A resilient, low-frequency, small-world human brain functional network with highly connected association cortical hubs,” *Journal of Neuroscience* **26**, 63–72 (2006).
- [6] Olaf Sporns and Jonathan D Zwi, “The small world of the cerebral cortex,” *Neuroinformatics* **2**, 145–162 (2004).
- [7] Patric Hagmann, Leila Cammoun, Xavier Gigandet, Reto Meuli, Christopher J Honey, Van J Wedeen, and Olaf Sporns, “Mapping the structural core of human cerebral cortex,” *PLoS Biol* **6**, e159 (2008).
- [8] Martijn P Van Den Heuvel and Olaf Sporns, “Rich-club organization of the human connectome,” *Journal of Neuroscience* **31**, 15775–15786 (2011).
- [9] Danielle S Bassett, Daniel L Greenfield, Andreas Meyer-Lindenberg, Daniel R Weinberger, Simon W Moore, and Edward T Bullmore, “Efficient physical embedding of topologically complex information processing networks in brains and computer circuits,” *PLoS comput biol* **6**, e1000748 (2010).
- [10] Mikail Rubinov and Olaf Sporns, “Complex network measures of brain connectivity: uses and interpretations,” *Neuroimage* **52**, 1059–1069 (2010).
- [11] Olaf Sporns and Richard F Betzel, “Modular brain networks,” *Annual review of psychology* **67**, 613–640 (2016).
- [12] David Meunier, Renaud Lambiotte, and Edward T Bullmore, “Modular and hierarchically modular organization of brain networks,” *Frontiers in neuroscience* **4**, 200 (2010).
- [13] Graham L Baum, Rastko Ceric, David R Roalf, Richard F Betzel, Tyler M Moore, Russell T Shinohara, Ari E Kahn, Simon N Vandekar, Petra E Rupert, Megan Quarmley, *et al.*, “Modular segregation of structural brain networks supports the development of executive function in youth,” *Current Biology* **27**, 1561–1572 (2017).
- [14] Jonathan D Power, Alexander L Cohen, Steven M Nelson, Gagan S Wig, Kelly Anne Barnes, Jessica A Church, Alecia C Vogel, Timothy O Laumann, Fran M Miezin, Bradley L Schlaggar, *et al.*, “Functional network organization of the human brain,” *Neuron* **72**, 665–678 (2011).
- [15] Richard F Betzel, Alessandra Griffa, Andrea Avena-Koenigsberger, Joaquín Goñi, Jean-Philippe Thiran, Patric Hagmann, and Olaf Sporns, “Multi-scale community organization of the human structural connectome and its relationship with resting-state functional connectivity,” *Network Science* **1**, 353–373 (2013).
- [16] Cedric Huchuan Xia, Zongming Ma, Zaixu Cui, Danilo Bzdok, Bertrand Thirion, Danielle S Bassett, Theodore D Satterthwaite, Russell T Shinohara, and Daniela M Witten, “Multi-scale network regression for brain-phenotype associations,” *Human Brain Mapping* (2020).
- [17] Aaron F Alexander-Bloch, Nitin Gogtay, David Meunier, Rasmus Birn, Liv Clasen, Francois Lalonde, Rhoshel Lenroot, Jay Giedd, and Edward T Bullmore, “Disrupted modularity and local connectivity of brain functional networks in childhood-onset schizophrenia,” *Frontiers in systems neuroscience* **4**, 147 (2010).
- [18] Maxwell A Bertolero, BT Thomas Yeo, Danielle S Bassett, and Mark D’Esposito, “A mechanistic model of connector hubs, modularity and cognition,” *Nature human behaviour* **2**, 765–777 (2018).
- [19] Santo Fortunato, “Community detection in graphs,” *Physics reports* **486**, 75–174 (2010).
- [20] Santo Fortunato and Darko Hric, “Community detection in networks: A user guide,” *Physics reports* **659**, 1–44 (2016).
- [21] Mark EJ Newman, “Communities, modules and large-scale structure in networks,” *Nature physics* **8**, 25–31 (2012).
- [22] Carlos Espinosa-Soto and Andreas Wagner, “Specialization can drive the evolution of modularity,” *PLoS Comput Biol* **6**, e1000719 (2010).
- [23] Azadeh Nematizadeh, Emilio Ferrara, Alessandro Flammini, and Yong-Yeol Ahn, “Optimal network modularity for information diffusion,” *Physical review letters* **113**, 088701 (2014).
- [24] Raj Kumar Pan and Sitabhra Sinha, “Modularity produces small-world networks with dynamical time-scale separation,” *EPL (Europhysics Letters)* **85**, 68006 (2009).
- [25] Marc Kirschner and John Gerhart, “Evolvability,” *Proceedings of the National Academy of Sciences* **95**, 8420–8427 (1998).
- [26] Katharina Glomb, Emeline Mullier, Margherita Carbone, Maria Rubega, Giannarita Iannotti, Sébastien Tourbier, Martin Seeber, Serge Vulliemoz, and Patric Hagmann, “Using structural connectivity to augment community structure in eeg functional connectivity,” *Network Neuroscience* , 1–59 (2019).
- [27] L-E Martinet, MA Kramer, W Viles, LN Perkins, E Spencer, CJ Chu, SS Cash, and ED Kolaczyk, “Robust dynamic community detection with applications to human brain functional networks,” *Nature Communications* **11**, 1–13 (2020).
- [28] Chi-Tin Shih, Olaf Sporns, Shou-Li Yuan, Ta-Shun Su, Yen-Jen Lin, Chao-Chun Chuang, Ting-Yuan Wang, Chung-Chuang Lo, Ralph J Greenspan, and Ann-Shyn Chiang, “Connectomics-based analysis of information flow in the drosophila brain,” *Current Biology* **25**, 1249–1258 (2015).
- [29] Kevin Mann, Courtney L Gallen, and Thomas R Clandinin, “Whole-brain calcium imaging reveals an intrinsic functional network in drosophila,” *Current Biology* **27**, 2389–2396 (2017).

- [30] Angela M Bruno, William N Frost, and Mark D Humphries, "Modular deconstruction reveals the dynamical and physical building blocks of a locomotion motor program," *Neuron* **86**, 304–318 (2015).
- [31] Travis A Jarrell, Yi Wang, Adam E Bloniarz, Christopher A Brittin, Meng Xu, J Nichol Thomson, Donna G Albertson, David H Hall, and Scott W Emmons, "The connectome of a decision-making neural network," *science* **337**, 437–444 (2012).
- [32] Richard F Betzel, "Organizing principles of whole-brain functional connectivity in zebrafish larvae," *Network Neuroscience* **4**, 234–256 (2020).
- [33] Darian H Hadjibabadi, Matthew Lovett-Barron, Ivan Raikov, Fraser Sparks, Zhenrui Liao, Scott C Baraban, Jure Leskovec, Attila Losonczy, Karl Deisseroth, and Ivan Soltesz, "Higher-order hub cells involved in feed-forward motifs as critical factors in epileptic network instability," *bioRxiv* (2020).
- [34] Masanori Shimono and John M Beggs, "Functional clusters, hubs, and communities in the cortical microconnectome," *Cerebral Cortex* **25**, 3743–3757 (2015).
- [35] Richard F Betzel, Katherine C Wood, Christopher Angeloni, Maria Neimark Geffen, and Danielle S Bassett, "Stability of spontaneous, correlated activity in mouse auditory cortex," *PLOS Computational Biology* **15**, e1007360 (2019).
- [36] Jesús Pérez-Ortega, Tzitzitlini Alejandre-García, and Rafael Yuste, "Long-term stability of neuronal ensembles in mouse visual cortex," *bioRxiv* (2020).
- [37] Richard F Betzel, John D Medaglia, Ari E Kahn, Jonathan Soffer, Daniel R Schonhaut, and Danielle S Bassett, "Structural, geometric and genetic factors predict interregional brain connectivity patterns probed by electrocorticography," *Nature biomedical engineering* **3**, 902–916 (2019).
- [38] BT Thomas Yeo, Fenna M Krienen, Jorge Sepulcre, Mert R Sabuncu, Danial Lashkari, Marisa Hollinshead, Joshua L Roffman, Jordan W Smoller, Lilla Zöllei, Jonathan R Polimeni, *et al.*, "The organization of the human cerebral cortex estimated by intrinsic functional connectivity," *Journal of neurophysiology* (2011).
- [39] R Cameron Craddock, Saad Jbabdi, Chao-Gan Yan, Joshua T Vogelstein, F Xavier Castellanos, Adriana Di Martino, Clare Kelly, Keith Heberlein, Stan Colcombe, and Michael P Milham, "Imaging human connectomes at the macroscale," *Nature methods* **10**, 524 (2013).
- [40] Jessica S Damoiseaux, SARB Rombouts, Frederik Barkhof, Philip Scheltens, Cornelis J Stam, Stephen M Smith, and Christian F Beckmann, "Consistent resting-state networks across healthy subjects," *Proceedings of the national academy of sciences* **103**, 13848–13853 (2006).
- [41] Stephen M Smith, Peter T Fox, Karla L Miller, David C Glahn, P Mickle Fox, Clare E Mackay, Nicola Filippini, Kate E Watkins, Roberto Toro, Angela R Laird, *et al.*, "Correspondence of the brain's functional architecture during activation and rest," *Proceedings of the national academy of sciences* **106**, 13040–13045 (2009).
- [42] Nicolas A Crossley, Andrea Mechelli, Petra E Vértes, Toby T Winton-Brown, Ameera X Patel, Cedric E Ginestet, Philip McGuire, and Edward T Bullmore, "Cognitive relevance of the community structure of the human brain functional coactivation network," *Proceedings of the National Academy of Sciences* **110**, 11583–11588 (2013).
- [43] Jennifer Stiso and Danielle S Bassett, "Spatial embedding imposes constraints on neuronal network architectures," *Trends in cognitive sciences* **22**, 1127–1142 (2018).
- [44] Bratislav Mišić, Richard F Betzel, Azadeh Nematzadeh, Joaquin Goni, Alessandra Griffa, Patric Hagmann, Alessandro Flammini, Yong-Yeol Ahn, and Olaf Sporns, "Cooperative and competitive spreading dynamics on the human connectome," *Neuron* **86**, 1518–1529 (2015).
- [45] Martin Rosvall and Carl T Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences* **105**, 1118–1123 (2008).
- [46] Stanley Wasserman and Joseph Galaskiewicz, *Advances in social network analysis: Research in the social and behavioral sciences* (Sage, 1994).
- [47] Brian Karrer and Mark EJ Newman, "Stochastic block-models and community structure in networks," *Physical review E* **83**, 016107 (2011).
- [48] Richard F Betzel, John D Medaglia, and Danielle S Bassett, "Diversity of meso-scale architecture in human and non-human connectomes," *Nature communications* **9**, 1–14 (2018).
- [49] Joshua Faskowitz, Xiaoran Yan, Xi-Nian Zuo, and Olaf Sporns, "Weighted stochastic block models of the human connectome across the life span," *Scientific reports* **8**, 1–16 (2018).
- [50] Yong-Yeol Ahn, James P Bagrow, and Sune Lehmann, "Link communities reveal multiscale complexity in networks," *nature* **466**, 761–764 (2010).
- [51] TS Evans and Renaud Lambiotte, "Line graphs, link partitions, and overlapping communities," *Physical Review E* **80**, 016105 (2009).
- [52] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *nature* **435**, 814–818 (2005).
- [53] Mark EJ Newman and Michelle Girvan, "Finding and evaluating community structure in networks," *Physical review E* **69**, 026113 (2004).
- [54] Vito Latora and Massimo Marchiori, "Efficient behavior of small-world networks," *Physical review letters* **87**, 198701 (2001).
- [55] Sergei Maslov and Kim Sneppen, "Specificity and stability in topology of protein networks," *Science* **296**, 910–913 (2002).
- [56] Marcus Kaiser and Claus C Hilgetag, "Nonoptimal component placement, but short processing paths, due to long-distance projections in neural systems," *PLoS Comput Biol* **2**, e95 (2006).
- [57] Richard F Betzel and Danielle S Bassett, "Specificity and robustness of long-distance connections in weighted, interareal connectomes," *Proceedings of the National Academy of Sciences* **115**, E4880–E4889 (2018).
- [58] David Samu, Anil K Seth, and Thomas Nowotny, "Influence of wiring cost on the large-scale architecture of human cortical connectivity," *PLoS Comput Biol* **10**, e1003557 (2014).
- [59] Andrew Zalesky, Alex Fornito, and Ed Bullmore, "On the use of correlation as a measure of network connectivity," *Neuroimage* **60**, 2096–2106 (2012).

- [60] Sergio Gómez, Pablo Jensen, and Alex Arenas, “Analysis of community structure in networks of correlated data,” *Physical Review E* **80**, 016114 (2009).
- [61] Richard F Betzel, John D Medaglia, Lia Papadopoulos, Graham L Baum, Ruben Gur, Raquel Gur, David Roalf, Theodore D Satterthwaite, and Danielle S Bassett, “The modular organization of human anatomical brain networks: Accounting for the cost of wiring,” *Network Neuroscience* **1**, 42–68 (2017).
- [62] Leonardo L Gollo, James A Roberts, Vanessa L Cropley, Maria A Di Biase, Christos Pantelis, Andrew Zalesky, and Michael Breakspear, “Fragility and volatility of structural hubs in the human connectome,” *Nature neuroscience* **21**, 1107–1116 (2018).
- [63] Marya Bazzi, Mason A Porter, Stacy Williams, Mark McDonald, Daniel J Fenn, and Sam D Howison, “Community detection in temporal multilayer networks, with an application to correlation networks,” *Multiscale Modeling & Simulation* **14**, 1–41 (2016).
- [64] Vincent A Traag, Paul Van Dooren, and Yurii Nesterov, “Narrow scope for resolution-limit-free community detection,” *Physical Review E* **84**, 016114 (2011).
- [65] Teddy J Akiki and Chadi G Abdallah, “Determining the hierarchical architecture of the human brain using subject-level clustering of functional networks,” *Scientific reports* **9**, 1–15 (2019).
- [66] Assaf Almog, M Renate Buijink, Ori Roethler, Stephan Michel, Johanna H Meijer, Jos HT Rohling, and Diego Garlaschelli, “Uncovering functional signature in neural systems via random matrix theory,” *PLoS computational biology* **15**, e1006934 (2019).
- [67] Farnaz Zamani Esfahlani, Maxwell A Bertolero, Danielle S Bassett, and Richard F Betzel, “Space-independent community and hub structure of functional brain networks,” *NeuroImage* **211**, 116612 (2020).
- [68] Danielle S Bassett, Eli T Owens, Mason A Porter, M Lisa Manning, and Karen E Daniels, “Extraction of force-chain network architecture in granular materials using community detection,” *Soft Matter* **11**, 2731–2744 (2015).
- [69] Roger Guimera and Luis A Nunes Amaral, “Functional cartography of complex metabolic networks,” *nature* **433**, 895–900 (2005).
- [70] Mark EJ Newman, “Finding community structure in networks using the eigenvectors of matrices,” *Physical review E* **74**, 036104 (2006).
- [71] Mark EJ Newman, “Fast algorithm for detecting community structure in networks,” *Physical review E* **69**, 066133 (2004).
- [72] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre, “Fast unfolding of communities in large networks,” *Journal of statistical mechanics: theory and experiment* **2008**, P10008 (2008).
- [73] Andrea Lancichinetti and Santo Fortunato, “Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities,” *Physical Review E* **80**, 016118 (2009).
- [74] Vincent A Traag, Ludo Waltman, and Nees Jan van Eck, “From louvain to leiden: guaranteeing well-connected communities,” *Scientific reports* **9**, 1–12 (2019).
- [75] Benjamin H Good, Yves-Alexandre De Montjoye, and Aaron Clauset, “Performance of modularity maximization in practical contexts,” *Physical Review E* **81**, 046106 (2010).
- [76] Max Hinne, Matthias Ekman, Ronald J Janssen, Tom Heskes, and Marcel AJ van Gerven, “Probabilistic clustering of the human connectome identifies communities and hubs,” *PloS one* **10**, e0117179 (2015).
- [77] Yoed N Kenett, Richard F Betzel, and Roger E Beaty, “Community structure of the creative brain at rest,” *NeuroImage* **210**, 116578 (2020).
- [78] Alexander Strehl and Joydeep Ghosh, “Cluster ensembles—a knowledge reuse framework for combining multiple partitions,” *Journal of machine learning research* **3**, 583–617 (2002).
- [79] Andrea Lancichinetti and Santo Fortunato, “Consensus clustering in complex networks,” *Scientific reports* **2**, 336 (2012).
- [80] Richard F Betzel and Danielle S Bassett, “Multi-scale brain networks,” *Neuroimage* **160**, 73–83 (2017).
- [81] Aaron F Alexander-Bloch, Haochang Shou, Siyuan Liu, Theodore D Satterthwaite, David C Glahn, Russell T Shinohara, Simon N Vandekar, and Armin Raznahan, “On testing for spatial correspondence between maps of human brain structure and function,” *Neuroimage* **178**, 540–551 (2018).
- [82] Ross Markello and Bratislav Misic, “Comparing spatially-constrained null models for parcellated brain maps,” *BioRxiv* (2020).
- [83] Tiago P Peixoto, “Revealing consensus and disensus between network partitions,” *arXiv preprint arXiv:2005.13977* (2020).
- [84] Tiago P Peixoto, “Nonparametric bayesian inference of the microcanonical stochastic block model,” *Physical Review E* **95**, 012317 (2017).
- [85] Santo Fortunato and Marc Barthelemy, “Resolution limit in community detection,” *Proceedings of the national academy of sciences* **104**, 36–41 (2007).
- [86] Jörg Reichardt and Stefan Bornholdt, “Statistical mechanics of community detection,” *Physical review E* **74**, 016110 (2006).
- [87] Lucas GS Jeub, Olaf Sporns, and Santo Fortunato, “Multiresolution consensus clustering in networks,” *Scientific reports* **8**, 1–16 (2018).
- [88] Danielle S Bassett, Mason A Porter, Nicholas F Wymbs, Scott T Grafton, Jean M Carlson, and Peter J Mucha, “Robust detection of dynamic community structure in networks,” *Chaos: An Interdisciplinary Journal of Nonlinear Science* **23**, 013142 (2013).
- [89] Ye He, Sol Lim, Santo Fortunato, Olaf Sporns, Lei Zhang, Jiang Qiu, Peng Xie, and Xi-Nian Zuo, “Reconfiguration of cortical networks in mdd uncovered by multiscale community detection with fmri,” *Cerebral Cortex* **28**, 1383–1395 (2018).
- [90] James M Shine, Patrick G Bissett, Peter T Bell, Oluwasanmi Koyejo, Joshua H Balsters, Krzysztof J Gorgolewski, Craig A Moodie, and Russell A Poldrack, “The dynamics of functional brain networks: integrated network states during cognitive task performance,” *Neuron* **92**, 544–554 (2016).
- [91] Maxwell A Bertolero, BT Thomas Yeo, and Mark D’Esposito, “The modular and integrative functional architecture of the human brain,” *Proceedings of the National Academy of Sciences* **112**, E6798–E6807 (2015).
- [92] Jonathan D Power, Bradley L Schlaggar, Christina N Lessov-Schlaggar, and Steven E Petersen, “Evidence

- for hubs in human functional brain networks," *Neuron* **79**, 798–813 (2013).
- [93] Mangor Pedersen, Amir Omidvarnia, James M Shine, Graeme D Jackson, and Andrew Zalesky, "Reducing the influence of intramodular connectivity in participation coefficient," *Network Neuroscience* **4**, 416–431 (2020).
- [94] Roger Guimera, Marta Sales-Pardo, and Luís A Nunes Amaral, "Modularity from fluctuations in random graphs and complex networks," *Physical Review E* **70**, 025101 (2004).
- [95] Paul Expert, Tim S Evans, Vincent D Blondel, and Renaud Lambiotte, "Uncovering space-independent communities in spatial networks," *Proceedings of the National Academy of Sciences* **108**, 7663–7668 (2011).
- [96] Leto Peel, Daniel B Larremore, and Aaron Clauset, "The ground truth about metadata and community detection in networks," *Science advances* **3**, e1602548 (2017).
- [97] Darko Hric, Richard K Darst, and Santo Fortunato, "Community detection in networks: Structural communities versus ground truth," *Physical Review E* **90**, 062805 (2014).
- [98] Jaewon Yang and Jure Leskovec, "Defining and evaluating network communities based on ground-truth," *Knowledge and Information Systems* **42**, 181–213 (2015).
- [99] Mark EJ Newman and Aaron Clauset, "Structure and inference in annotated networks," *Nature communications* **7**, 1–11 (2016).
- [100] Andrew C Murphy, Shi Gu, Ankit N Khambhati, Nicholas F Wymbs, Scott T Grafton, Theodore D Satterthwaite, and Danielle S Bassett, "Explicitly linking regional activation and function connectivity: community structure of weighted networks with continuous annotation," *arXiv preprint arXiv:1611.07962* (2016).
- [101] Andrea Lancichinetti, Filippo Radicchi, José J Ramasco, and Santo Fortunato, "Finding statistically significant communities in networks," *PloS one* **6**, e18961 (2011).
- [102] Joshua Farkowitz, Farnaz Zamani Esfahlani, Youngheun Jo, Olaf Sporns, and Richard F Betzel, *Edge-centric functional network representations of human cerebral cortex reveal overlapping system-level architecture*, Tech. Rep. (Nature Publishing Group, 2020).
- [103] Peter J Mucha, Thomas Richardson, Kevin Macon, Mason A Porter, and Jukka-Pekka Onnela, "Community structure in time-dependent, multiscale, and multiplex networks," *science* **328**, 876–878 (2010).
- [104] Mikko Kivelä, Alex Arenas, Marc Barthelemy, James P Gleeson, Yamir Moreno, and Mason A Porter, "Multilayer networks," *Journal of complex networks* **2**, 203–271 (2014).
- [105] Danielle S Bassett, Nicholas F Wymbs, Mason A Porter, Peter J Mucha, Jean M Carlson, and Scott T Grafton, "Dynamic reconfiguration of human brain networks during learning," *Proceedings of the National Academy of Sciences* **108**, 7641–7646 (2011).
- [106] Karolina Finc, Kamil Bonna, Xiaosong He, David M Lydon-Staley, Simone Kühn, Włodzisław Duch, and Danielle S Bassett, "Dynamic reconfiguration of functional brain networks during working memory training," *Nature communications* **11**, 1–15 (2020).
- [107] Richard F Betzel, Maxwell A Bertolero, Evan M Gordon, Caterina Gratton, Nico UF Dosenbach, and Danielle S Bassett, "The community structure of functional brain networks exhibits scale-specific patterns of inter-and intra-subject variability," *Neuroimage* **202**, 115990 (2019).
- [108] Michael W Cole, Danielle S Bassett, Jonathan D Power, Todd S Braver, and Steven E Petersen, "Intrinsic and task-evoked network architectures of the human brain," *Neuron* **83**, 238–251 (2014).
- [109] Raphael T Gerraty, Juliet Y Davidow, Karin Foerde, Adriana Galvan, Danielle S Bassett, and Daphna Shohamy, "Dynamic flexibility in striatal-cortical circuits supports reinforcement learning," *Journal of Neuroscience* **38**, 2442–2453 (2018).
- [110] Richard F Betzel, Theodore D Satterthwaite, Joshua I Gold, and Danielle S Bassett, "Positive affect, surprise, and fatigue are correlates of network flexibility," *Scientific Reports* **7**, 1–10 (2017).
- [111] Urs Braun, Axel Schäfer, Henrik Walter, Susanne Erk, Nina Romanczuk-Seiferth, Leila Haddad, Janina I Schweiger, Oliver Grimm, Andreas Heinz, Heike Tost, et al., "Dynamic reconfiguration of frontal brain networks during executive cognition in humans," *Proceedings of the National Academy of Sciences* **112**, 11678–11683 (2015).
- [112] Lucy R Chai, Marcelo G Mattar, Idan Asher Blank, Evelina Fedorenko, and Danielle S Bassett, "Functional network dynamics of the language system," *Cerebral Cortex* **26**, 4148–4159 (2016).
- [113] Urs Braun, Axel Schäfer, Danielle S Bassett, Franziska Rausch, Janina I Schweiger, Edda Bilek, Susanne Erk, Nina Romanczuk-Seiferth, Oliver Grimm, Lena S Geiger, et al., "Dynamic brain network reconfiguration as a potential schizophrenia genetic risk mechanism modulated by nmda receptor function," *Proceedings of the National Academy of Sciences* **113**, 12568–12573 (2016).
- [114] Martin Rosvall, Daniel Axelsson, and Carl T Bergstrom, "The map equation," *The European Physical Journal Special Topics* **178**, 13–23 (2009).
- [115] Joshua Farkowitz and Olaf Sporns, "Mapping the community structure of the rat cerebral cortex with weighted stochastic block modeling," *Brain Structure and Function* **225**, 71–84 (2020).
- [116] Daniel Moyer, Boris Gutman, Gautam Prasad, Joshua Farkowitz, Greg Ver Steeg, and Paul Thompson, "Blockmodels for connectome analysis," in *11th International Symposium on Medical Information Processing and Analysis*, Vol. 9681 (International Society for Optics and Photonics, 2015) p. 96810A.
- [117] Tom AB Snijders and Krzysztof Nowicki, "Estimation and prediction for stochastic blockmodels for graphs with latent block structure," *Journal of classification* **14**, 75–100 (1997).
- [118] Tiago P Peixoto, "Hierarchical block structures and high-resolution model selection in large networks," *Physical Review X* **4**, 011047 (2014).
- [119] Mark EJ Newman and Elizabeth A Leicht, "Mixture models and exploratory analysis in networks," *Proceedings of the National Academy of Sciences* **104**, 9564–9569 (2007).
- [120] Shihua Zhang, Rui-Sheng Wang, and Xiang-Sun Zhang, "Identification of overlapping community structure in

- complex networks using fuzzy c-means clustering,” *Physica A: Statistical Mechanics and its Applications* **374**, 483–490 (2007).
- [121] Vincenzo Nicosia, Giuseppe Mangioni, Vincenza Carchiolo, and Michele Malgeri, “Extending the definition of modularity to directed graphs with overlapping communities,” *Journal of Statistical Mechanics: Theory and Experiment* **2009**, P03024 (2009).
- [122] Mahshid Najafi, Brenton W McMenamin, Jonathan Z Simon, and Luiz Pessoa, “Overlapping communities reveal rich structure in large-scale brain networks during rest and task conditions,” *Neuroimage* **135**, 92–106 (2016).
- [123] Youngheun Jo, Farnaz Zamani Esfahlani, Joshua Faskowitz, Evgeny Chumin, Olaf Sporns, and Richard Betzel, “The diversity and multiplexity of edge communities within and between brain systems,” *bioRxiv* (2020).
- [124] Farnaz Zamani Esfahlani, Youngheun Jo, Joshua Faskowitz, Lisa Byrge, Daniel Kennedy, Olaf Sporns, and Richard Betzel, “High-amplitude co-fluctuations in cortical activity drive functional connectivity,” *bioRxiv* , 800045 (2020).
- [125] Steve Gregory, “Finding overlapping communities in networks by label propagation,” *New journal of Physics* **12**, 103018 (2010).
- [126] Xuan Li, John Q Gan, and Haixian Wang, “Collective sparse symmetric non-negative matrix factorization for identifying overlapping communities in resting-state brain functional networks,” *NeuroImage* **166**, 259–275 (2018).