

1 Modular Networks: Structure

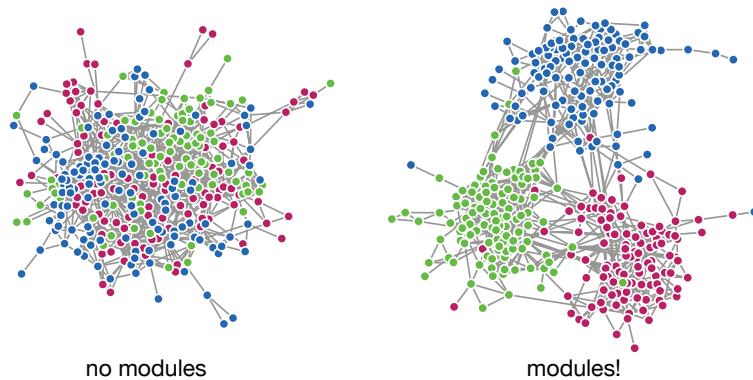
Beyond local and global measures of a network's structure, connectivity can also vary in highly non-uniform ways across a network. By focusing on individual nodes, local measures may miss these larger-scale variations, while global measures may miss them because they average across all nodes. Instead, what we want is a way of considering structure, and its variation, at a scale that falls *between* the local and global levels.

One way to do this is to consider “clusters” of nodes and edges, which may range in size from relatively small (like a small motif) to relatively large (hundreds of nodes, or more), but always above the level of individual nodes and below the level of the whole network. By decomposing a network into clusters, we can effectively “coarse grain” its structure to produce a simpler view that allows us to talk about how big pieces of a network fit together to construct a whole.

We can define the idea of a network cluster, which we will call a **module** or a **community** or a **compartment**,¹ very generally, and this broad definition will cover a variety more specific notions of large-scale structure that we will learn about below.

a community is a group of nodes that connect to other groups in similar ways.

This is an inherently statistical definition (the word “similar”): nodes in the same group will follow the same statistical rules for how they connect to other nodes, and, these rules can vary between groups. Some groups might be *assortative*, meaning their nodes connect mainly to each other; some might be *disassortative*, meaning they connect mainly to nodes in other groups; and other groups might exhibit a mixture of these behaviors, connecting preferentially to some groups but not to others. The variation in these group-level connectivity patterns allows us to capture a wide variety of notions of modular structure in a network.



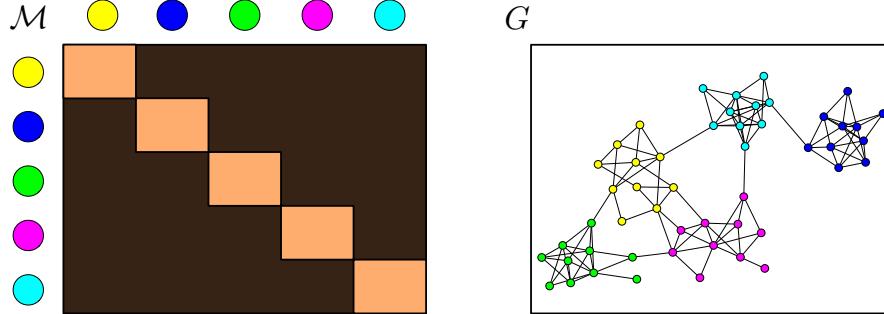
¹The term “module” is most common in molecular biology, while “community” is more common in the social and physical sciences, and “compartment” is more common in ecology. Each refers to the same underlying idea.

In biological networks, communities (modules) may represent functional clusters of genes or proteins or neurons, while in ecological networks, communities (compartments) may represent groups of species that are functionally similar in an ecosystem, such as a group of predators that consume similar prey. These communities may represent targets of natural selection above the level of individual nodes but below the level of the whole system. In social networks, communities often represent different kinds of human social groups, e.g., with common interests, histories, or behaviors.

2 The mixing matrix of communities

We can formalize the above notion of a community (module, compartment) using a simple tabulation called a **mixing matrix** \mathcal{M} . If there are c communities in the network, \mathcal{M} is a $c \times c$ matrix, and the value of the element \mathcal{M}_{rs} is related to the frequency or likelihood that a node in a group r connects to a node in a group s .

Below is a simple mixing matrix and its corresponding network. In the matrix, a lighter color indicates a greater density of edges (higher likelihood of connection), while a darker color indicates a lower density. Because the values along the diagonal $\mathcal{M}_{r=s}$ are higher than the values in the off-diagonal $\mathcal{M}_{r \neq s}$, this mixing matrix defines an *assortative* community structure: two nodes with the same group label are more likely to be connected than are two nodes with different labels. Hence, in the corresponding graph G , we see more edges within each particular group, than between them.

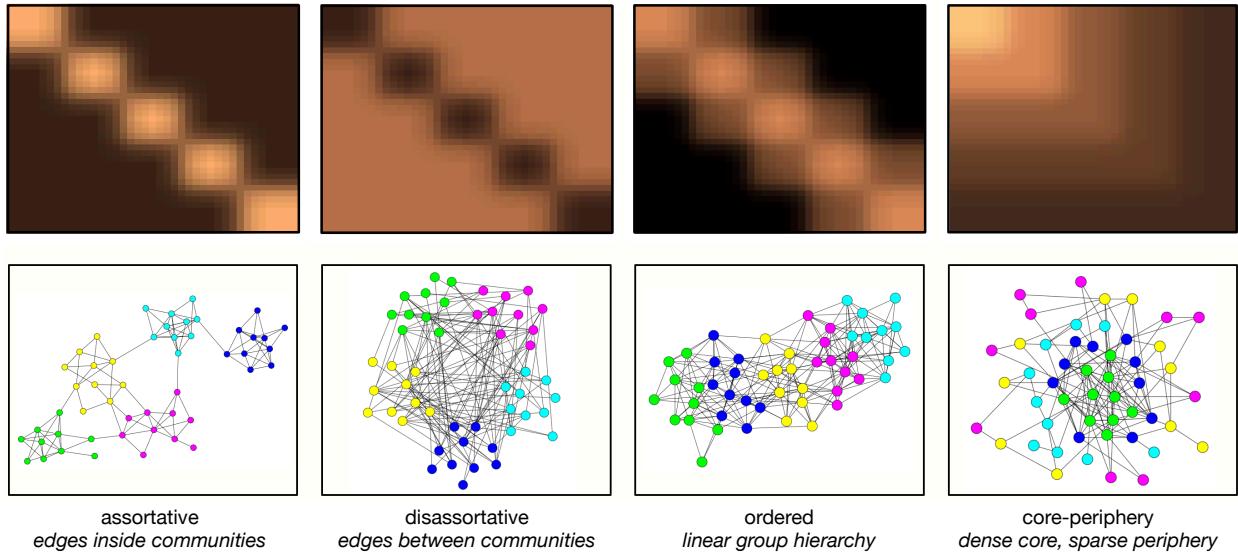


The mixing matrix can naturally represent a wide variety of group-level connectivity patterns, depending on how we organize the values of \mathcal{M}_{rs} . If we specify a particular pattern of group-level interactions, we can use the mixing matrix to *generate* a random graph with that structure. Or, we could *infer* a mixing matrix for a real-world network that represents a reasonable clustering of its nodes and describes how these clusters interact. In both settings, the mixing matrix provides a simple view of how a network is organized at one particular scale, showing how its parts at this scale connect to each other. If we vary the number of groups, we vary the scale of analysis—fewer groups means larger groups, and more groups means each is a smaller piece of the network.

If groups all interact with each other in the same way, then it doesn't matter how many groups we define, because $\forall_{rs} \mathcal{M}_{rs} = \text{const}$. In this case, we might as well just say we have $c = 1$ group, and collapse the mixing matrix into a single number that represents the uniform (homogeneous) density of edges in the entire network. This situation should sound familiar: it is exactly the setting of an Erdős-Rényí random graph!

With $c > 1$ groups, we can arrange the $O(c^2)$ values of \mathcal{M} to represent a wide variety of “stylized” modular patterns, such as:

- **assortative communities**: edges are mainly *within* the groups, and most interactions occur along the mixing matrix's diagonal, so that $\mathcal{M}_{rr} > \mathcal{M}_{r \neq s}$.
- **disassortative communities**: where edges are mainly *between* the groups, and most interactions occur in the matrix's off-diagonal, so that $\mathcal{M}_{rr} < \mathcal{M}_{r \neq s}$.
- **ordered communities**: where edges are mainly within groups or run between “consecutive” groups in a linear ordering; this pattern is similar to assortative communities, except that the values in the first off-diagonal fall between those of the main diagonal and the off-diagonal bulk, i.e., $\mathcal{M}_{rr} > \mathcal{M}_{r,r+1} > \mathcal{M}_{r,r+\ell}$ for $\ell \geq 2$.
- **core-periphery structure**: where edges tend to occur mainly within a relatively dense “core” community, which is surrounded by successively more sparse peripheral layers (kind of like an onion).

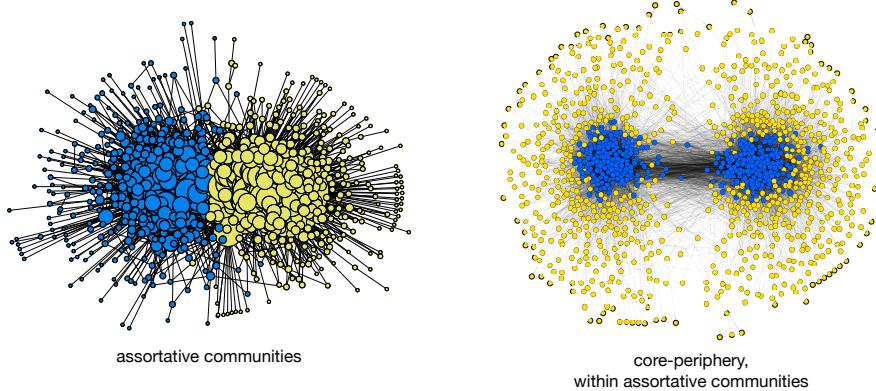


In the above examples, we assumed that the mixing matrix is undirected, but these matrices can be naturally generalized to directed interactions, so that $\mathcal{M}_{rs} \neq \mathcal{M}_{sr}$, which would capture the notion that edges tend to run more often in one direction than the other, e.g., in a food web, where predation links point down, toward basal or primary-produce species, or in a social network, where links represent some notion of social status.

2.1 Real-world examples

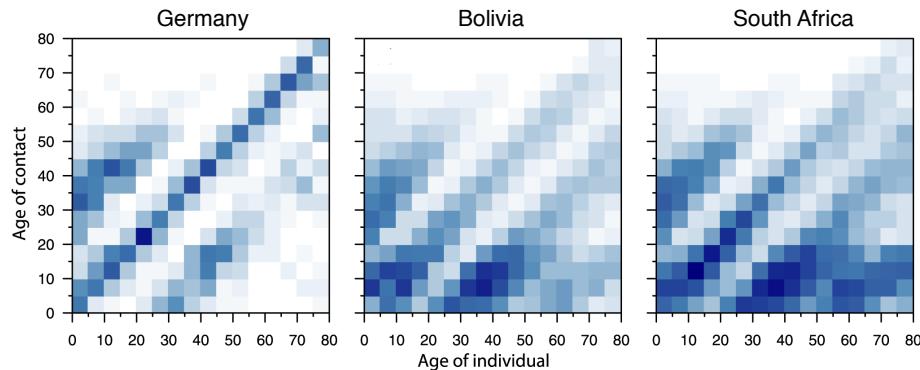
Most real-world networks exhibit a mixture of these four stylized patterns, and which ones we see can vary depending on the scale of analysis, i.e., what value of c we choose. For larger choices of c , we might observe that some communities in a network are assortative, while within them, they exhibit either an ordered or a core-periphery structure. A community might also be assortative (many edges within it), but exhibit different levels of connectivity with other communities, being densely connected to some, and sparsely connected to others. All of these variations are possible, and as c increases, the range of patterns and combinations of patterns also increases. (What happens to the mixing matrix \mathcal{M} in the limit of $c \rightarrow n$?) Theoretical and empirical analyses suggest that for most networks $c \approx \sqrt{m}$, meaning that as a network grows in size (more edges m), we need a sublinear additional number of groups to account for the additional connectivity patterns.

A few real-world networks do exhibit relatively “clean” versions of the stylized patterns. For instance, the political blogs network we saw in Lecture 2 exhibits a clear *assortative* community structure at the largest scale ($c = 2$; left-hand figure below). But, if we drill down a little more within each of those communities, we observe that each exhibits a *core-periphery* pattern, of a relatively dense core surrounded by a diffuse, weakly interconnected periphery ($c = 4$; right-hand figure).² (What would the mixing matrices look like for each of these networks?)



²Figures from Karrer & Newman, *Phys. Rev. E* **83**, 016107 (2011) <https://arxiv.org/abs/1008.3926>, and from Zhang, Martin, & Newman, *Phys. Rev. E* **91**, 032803 (2015) <https://arxiv.org/abs/1409.4813>.

Ordered patterns are more prevalent in social networks, where linear hierarchies are common. Empirical studies of social connections show that social interactions tend to be ordered by age: most people interact with others who are within 5 years of their own age, and they interact at slightly lower rates for people within 5–10 years of their age.³ However, a general exception exists: children interact at high rates with individuals about 25–35 years (and about 60 years) older than them, which creates strong but specific off-diagonal patterns in the mixing matrices.



2.2 Hierarchies, and multiple scales

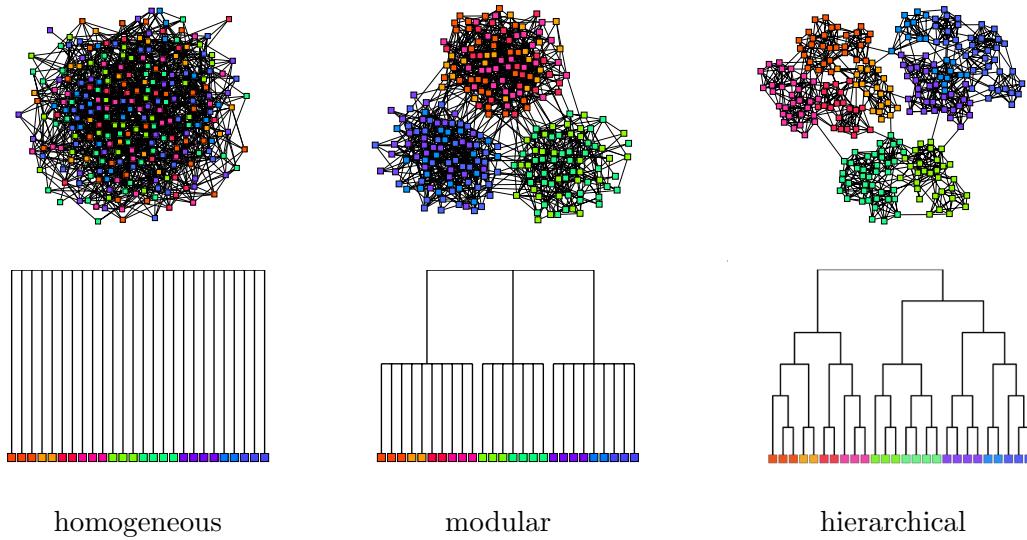
When we talk about modules or communities in a network, and use the mixing matrix to describe how their connectivity, we often assume or imply that there is otherwise little additional structure either within a community, or in the connections between a pair of communities. That is, we're implying that there's one main level of organization between the local and global. But why stop at one?

A natural generalization of community structure is called *hierarchical* community structure, in which vertices divide into groups that further subdivide into groups of groups, and so forth over multiple scales. In this way, a hierarchy spans all the scales between the local and global levels. This kind of nested structure is often represented using a *dendrogram* or a tree that shows how the smallest groups are nested within larger groups, and they in turn are nested within still larger groups, etc.⁴ Taking a “slice” across the dendrogram induces a single decomposition of the network into groups, and sliding that line up or down in the dendrogram provides a natural way to “zoom” in or out across scales.

³Figures from Prem, Cook & Jit, *PLoS Comp. Bio.* **13**, e1005697 (2017)
<http://dx.doi.org/10.1371/journal.pcbi.1005697>.

⁴Figure below from Clauset, Moore & Newman, *Nature* **453**, 98–101 (2008) <https://arxiv.org/abs/0811.0484>.

A common assumption for this kind of large-scale organization is that two nodes that are “closely related”, that is, are separated by a short distance on the tree, are more likely to be connected. This is typical, for instance, of human social structures, for instance: our class sits within the larger Computer Science department, which itself sits inside the larger College of Engineering, which sits inside CU, which sits inside Colorado schools, etc.



3 Random graphs with modular structure

In this section, we’ll describe how to define modular random graphs, in which we define a distribution over graphs $\Pr(G)$ that is conditioned on having a specified modular or community structure, and also potentially a specified degree distribution. This idea continues the theme introduced in Lecture 3, in which we defined the Erdős-Rényi random graph, with a specified edge density p , and the configuration model, in which a random graph has a specified degree sequence \vec{k} .

3.1 The stochastic block model

The simplest random graph with modular structure is a generalization of the Erdős-Rényi model called the **stochastic block model** or SBM.⁵ This model was first studied in mathematical sociology in the mid-1980s. Today, it’s common in all branches of network analysis, from computational biology, to machine learning and statistical physics.

⁵Sometimes also called the “stochastic blockmodel.” Introduced in Holland, Laskey & Leinhardt, *Social Networks* **5**(2), 109–137 (1983) <https://bit.ly/2legJU0>, and then further developed in Wang & Wong, *J. American Statistical Association* **82**, 8–19 (1987) <https://bit.ly/2mKrJK1>.

An SBM graph is conventionally a simple graph, and is defined by a tuple of parameters $\theta = (c, \vec{z}, \mathcal{M})$, where

1. c is the number of communities,
2. \vec{z} is a labeling or a **partition** of nodes by community membership so that $z_i \in \{1, 2, \dots, c\}$, and
3. \mathcal{M} is a $c \times c$ “stochastic block matrix” where \mathcal{M}_{rs} gives the probability that a node in community r connects to a node in community s . (In other words, the mixing matrix, defined more precisely.)

Given these parameters, the probability of an undirected edge is

$$\forall_{i>j} \quad A_{ij} = A_{ji} = \begin{cases} 1 & \text{with probability } \mathcal{M}_{z_i, z_j} \\ 0 & \text{otherwise} . \end{cases}$$

That is, given the labels $r = z_i$ and $s = z_j$ for a pair of nodes i, j , we simply look up in the mixing matrix \mathcal{M} the corresponding density parameter, and flip a coin with that bias. The variation in these edge density parameters represents the block or community structure of the graph, and all of the conceptual ideas introduced in Section 2 carry over naturally. For the subgraph defined by a single community, i.e., when $r = s$, the SBM produces a simple Erdős-Rényi random graph with density \mathcal{M}_{rr} , while for the subgraph defined by a pair of communities, i.e., when $r \neq s$, the SBM produces a simple random bipartite graph with density \mathcal{M}_{rs} .

3.1.1 Properties of SBM graphs

At a high level, SBM graphs have the following properties:

- Traditionally a *simple graph*, but also easily generalized to directed edges, with or without self-loops (do you see how?).
- Connectivity is *modular*, and is governed by the sizes of the c groups \vec{n} and their groupwise edge densities \mathcal{M} ; edges are independent, conditioned on their nodes’ labels.
- The degree distribution $\Pr(k)$ a combination of Poisson distributions, each with mean \mathcal{M}_{rs} , which can produce high-variance distributions only if node degree correlates with group label (i.e., high degree nodes are grouped together).
- The diameter and mean geodesic distance are typically $O(\log n)$, unless \mathcal{M} has unusual structure.
- The clustering coefficient tends toward $C = O(1/n)$, but depends strongly on group sizes and their internal density specified by \mathcal{M} .
- The largest connected component (LCC) tends to be $O(n)$ when G is not too sparse.

Because edges are conditionally independent (conditioned on z_i, z_j only), we sometimes say that nodes in the same group are *stochastically equivalent*, because they have equivalent connectivity patterns to other vertices. That is, every vertex in group r has the same set of probability values that govern the connections to other vertices, and this set is given by the r th row (or column) of the matrix \mathcal{M} .

In that sense, the SBM is the most literal translation of our the definition of a community we saw in Section 1: a community in the SBM is, by definition, a group of nodes with the same rules for connecting to nodes in other groups. The effective definition of some group r , then, is given by the r th row vector of groupwise densities \mathcal{M}_{r*} —if a node follows those rules, then it is “stochastically equivalent” to all the other nodes that follow the same rules, and these nodes define the r th group.

3.1.2 Generating a SBM network

Once the parameters $\theta = (c, \vec{z}, \mathcal{M})$ have been specified, drawing a graph from the ensemble is straightforward:

1. initialize an empty graph G with n nodes
2. for each of the $\binom{n}{2}$ pairs i, j , draw a uniformly random number r
3. if $r \leq \mathcal{M}_{z_i, z_j}$, then add the undirected edge (i, j) to G .

Basically, this procedure flips $\binom{n}{2}$ coins, and for each pair i, j , looks up the corresponding bias \mathcal{M}_{z_i, z_j} , and checks whether the ij th coin was a “heads” under that bias. As a result, generating a network in this way takes $\Theta(n^2)$ time, and is computationally expensive when $n > 10^4$ or so.

All-in-all, the SBM has a large number of parameters, and this greater flexibility is what allows it to produce such a broad variety of large-scale network patterns. In total, there are $1 + n + \Theta(c^2)$ parameters, which correspond to the number of values required to specify the number of groups c , a group label for each node \vec{z} and the groupwise edge densities \mathcal{M} , respectively.

3.1.3 Planted partitions

When we generate a synthetic network using the SBM, we have to specify both the **partition** of nodes \vec{z} , and the mixing matrix \mathcal{M} . Once specified, we sometimes call these a *planted partition*, since we’re “planting” a particular kind of structure inside the random graph. Although the full flexibility of the model can be useful, often it is convenient to make the planted partition very simple, in order to induce one of the stylized patterns we saw in Section 2 in a way that we can vary using only a small number of parameters. Such simplified models are often very useful as a substrate for numerical experiments.

The most common form of a planted partition is a simplified assortative pattern, where we couple together all the off-diagonal (between group) densities using a single parameter. This reduces the $\Theta(c^2)$ free parameters in \mathcal{M} to only $c + 1$ parameters: one for the between-group densities q , and c parameters p_1, p_2, \dots, p_c for each of the within-group densities, like so:

$$\mathcal{M}_{\text{assort}}^{\text{planted-}c} = \begin{pmatrix} p_1 & q & \cdots & q \\ & p_2 & \ddots & \vdots \\ & & \ddots & q \\ & & & p_c \end{pmatrix}.$$

We can then vary the group sizes and the densities within each community to produce a variety of assortative patterns.

However, the most well-studied planted partition model further simplifies this form in four ways, which produces a *one* parameter model:

1. Equalize the within-group densities: $\forall_i p_i = p$. This choice reduces the mixing matrix to just two parameters, p for within-group edges, and q for between-group edges
2. Equalize the group sizes: $\forall_i n_i = n/c$.
3. Have only $c = 2$ groups.
4. Fix the mean degree to be $\langle k \rangle = d$.

Hence, the model becomes

$$\mathcal{M}_{\text{assort}}^{\text{planted-}2} = \begin{pmatrix} p & q \\ & p \end{pmatrix} \quad \text{and} \quad n_1 = n_2 = n/2 , \quad (1)$$

and the fourth constraint implies that p and q are no longer independent. That is, in order for the mean degree to remain fixed at d , if we increase the within-group density p , then we must decrease the between-group density q by the same amount, and vice versa.

Let's define $p = d_{\text{in}}/n$ and $q = d_{\text{out}}/n$, where d_{in} and d_{out} are twice the within- and between-group degrees of each node (do you see why?). A little algebra can then show that $2d = d_{\text{in}} + d_{\text{out}}$. Defining $\epsilon = d_{\text{in}} - d_{\text{out}}$, we can then solve the $2d$ expression to obtain $d_{\text{out}} = d - \epsilon/2$ and $d_{\text{in}} = d + \epsilon/2$. Plugging these into our original expressions for p and q yields

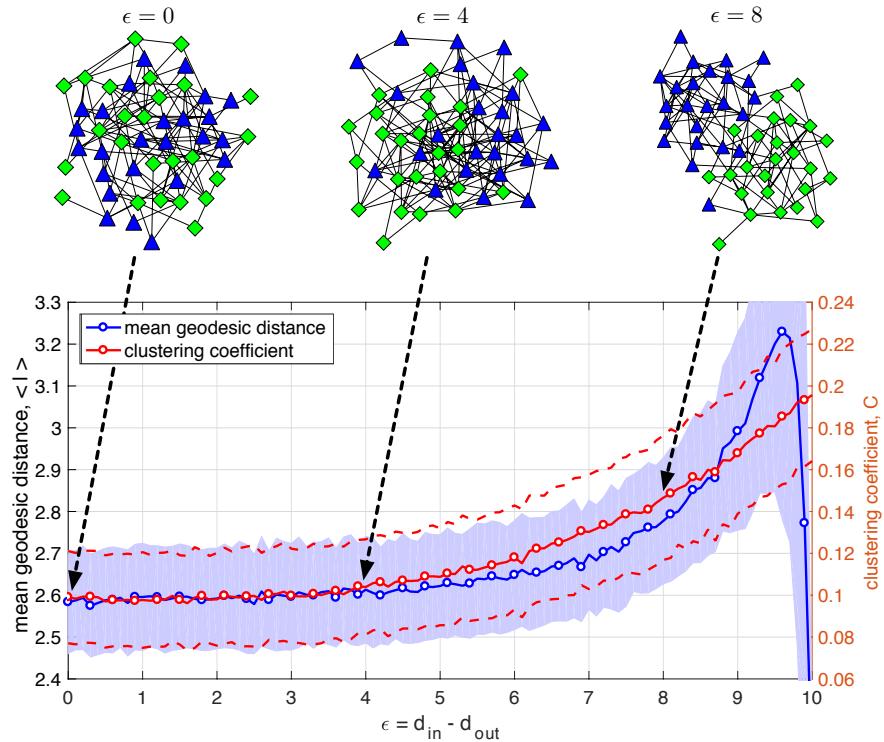
$$p = (d + \epsilon/2)/n \quad q = (d - \epsilon/2)/n .$$

Now, if we fix n and d , we have a simple, one-parameter model in which varying ϵ changes the relative balance of within- vs. between-group edges for each node. When $\epsilon \gg 0$, within-group edges

far outnumber between-group edges, and the two groups are very distinct. As $\epsilon \rightarrow 0$, the relative difference goes away, and the groups become indistinguishable (as in an ER graph).

Planting partitions for simulation experiments. The value of ϵ naturally varies from $[0, 2d]$ for assortative communities (do you see what happens if $\epsilon < 0$?). Given a choice of ϵ , we can derive the values of p and q , and then generate a synthetic network. Hence, this simple planted partition model provides a way to characterize how different measures of network structure vary with how “crisp” the division is between a pair of communities, as captured by ϵ .

For example, setting $n = 50$ nodes and the mean degree $d = 5$, the following figure shows how the mean geodesic distance $\langle \ell \rangle$ (lefthand y -axis) and the clustering coefficient C (righthand y -axis) vary as a function of ϵ . Each point on either line is the average measure over 250 repetitions, and the shaded (dashed) region represents one standard deviation above and below this mean behavior. For three choices of ϵ , a visualization provides some visual intuition as to the structure of the corresponding graph.



When $\epsilon < 4$ or so, both network measures are relatively stable, even though the relative share of within- to between-group degrees, given by ϵ , is changing. Above $\epsilon = 4$, we see that the clustering coefficient responds more quickly to the increasing division between the communities, while the

$\langle \ell \rangle$ rises only slightly. It's not until around $\epsilon = 6$ that the $\langle \ell \rangle$ starts to increase more noticeably, and then it does so very quickly, especially for $\epsilon > 8$. This difference in the responsiveness of $\langle \ell \rangle$ and C to increasingly strong community divisions ϵ illustrates a key insight: $\langle \ell \rangle$ has much more of non-linear response to structural changes than C does. In fact, the $\langle \ell \rangle$ peaks just before and then collapses at $\epsilon = 2d$ (do you see why?), while C exhibits only a smooth increase over the entire range.

There are many other ways we could use the planted partition model, or more elaborate variations. For instance, imagine defining a simple planted partition model that represents core-periphery structure, to investigate how a measure like each node's mean geodesic distance $\langle \ell \rangle_i$ varies as a function of how sparse the periphery is. Etc. The utility of the planted partition model comes primarily from its ability to be a substrate for other analyses, and any useful specification of \vec{z} and \mathcal{M} can provide great insight about how structure varies, and how it shapes dynamics that run on top of the network.

3.2 The degree-corrected stochastic block model

Just as the stochastic block model is a modular generalization of Erdős-Rényi random graphs, the **degree-corrected stochastic block model** or DC-SBM is a modular generalization of a random graph with specified degree structure.⁶ Combining a specified group structure and a specified degree structure makes the DC-SBM the most flexible random graph model in our toolbox, and also the most realistic.⁷ The DC-SBM was introduced in 2010, and is now widely used in many branches of network analysis.

There are notable differences between the DC-SBM and the SBM. First, the DC-SBM is an undirected *multigraph model*. In the DC-SBM, each adjacency value A_{ij} is a Poisson random variable with a mean that depends both on the degrees k_i, k_j and on their groupwise mixing parameter \mathcal{M}_{z_i, z_j} . This makes A_{ij} a counting variable rather than a binary 0 or 1, as in the SBM. To output a simple graph, we can just collapse the multi-edges. Second, because we need to balance the effect of the mixing matrix with the effect of the degree sequence, the DC-SBM requires a little more work to parameterize properly.

Concretely, beyond the number of nodes n , a DC-SBM is specified by a tuple $\theta = (c, \vec{z}, \vec{k}, \mathcal{M})$:

1. c is the number of communities,
2. \vec{z} is a partition of the nodes, by their community membership, where $z_i \in \{1, 2, \dots, c\}$,
3. \vec{k} is the expected degree sequence (like the Chung-Lu model), and

⁶Introduced in Karrer & Newman, *Phys. Rev. E* **83**, 016107 (2011) <https://arxiv.org/abs/1008.3926>.

⁷By using explicit parameters for degrees \vec{k} , the DC-SBM can use the mixing matrix \mathcal{M} as intended, to represent a network's group-level structure, rather than simply grouping together vertices with similar degrees as the SBM tends to do in practice to create a heavy-tailed degree distribution.

4. \mathcal{M} is a $c \times c$ mixing matrix, where \mathcal{M}_{rs} counts the number of edges between groups r and s , or, if $r = s$, twice that number, implying $\mathcal{M}_{rs} = \sum_{ij} A_{ij} \delta_{z_i, r} \delta_{z_j, s}$,

where $\delta_{z_i, r} = 1$ if $z_i = r$ and is 0 otherwise.⁸ Finally, given the partition \vec{z} , we define the r th group's size $n_r = \sum_{i=1}^n \delta_{z_i, r}$, i.e., the number of nodes with label r .

Crucially, the definition of \mathcal{M} in the DC-SBM is different than in the SBM: in the DC-SBM, \mathcal{M}_{rs} is an edge count, not an edge density. This fact implies that the matrix sum $\sum_{rs} \mathcal{M} = 2m$, because every edge must run between some pair of groups r, s .⁹

It's useful think of the DC-SBM as a two-level model. In the first level, the groups themselves act as nodes in a group-level network, and the mixing matrix \mathcal{M} gives the connectivity among these groups (like an adjacency matrix, but for a multigraph). In the second level, each group node gets expanded into a collection of individual nodes. At this level, a connection between a pair of nodes i, j in the same group $r = z_i = z_j$ will be equivalent to a self-loop for the r th group in the first level (counted by \mathcal{M}_{rr}). And, much like the planted partition model for the SBM, each node's degree k_i can be divided into a within-group portion, which contributes to \mathcal{M}_{rr} and a between-group portion, which contributes to $\mathcal{M}_{r \neq s}$. This ideas will be useful when we describe below how to generate a graph from a DC-SBM.

3.2.1 Properties of DC-SBM graphs

At a high level, DC-SBM graphs have the following properties:

- By default, an *undirected multigraph*, generalizable to directed edges; obtain a simple network by collapsing multi-edges.
- *Modular* connectivity, governed by the sizes of the c groups and the mixing matrix \mathcal{M} .
- The *degree structure* is specified by \vec{k} .
- Diameter and mean geodesic distance are typically $O(\log n)$, unless \mathcal{M} has unusual structure.
- The clustering coefficient tends toward $C = O(1/n)$, but depends strongly on group sizes, their internal density specified by \mathcal{M} , and the variance of \vec{k} .
- The largest connected component (LCC) tends to be $O(n)$ when G is not too sparse.

⁸The term $\delta_{a,b}$ is the Kronecker delta function, which behaves like an indicator variable:
https://en.wikipedia.org/wiki/Kronecker_delta.

⁹Note: being an edge count does not necessarily imply that $\mathcal{M}_{rs} \leq n_r n_s$, since the DC-SBM generates multigraphs.

3.2.2 Generating a DC-SBM network

In the first-level of the DC-SBM, groups act like “super” nodes, and so we can define the “degree” of the r th group as

$$\kappa_r = \sum_s \mathcal{M}_{rs} = \sum_i k_i \delta_{z_i, r} , \quad (2)$$

that is, the sum of the degrees of all the nodes in group r . The network at this level is then very much like a (loopy multigraph) configuration model, with a degree sequence $\vec{\kappa}$.

In the model’s second level, where we expand the r th group to be a collection of n_r nodes, we now need to allocate the κ_r connections of the group “node” to specific group members, in a way that respects their degrees. Consider one such connection. The probability that it lands on the i th node is given by what we can call its *propensity*

$$\gamma_i = \frac{k_i}{\kappa_{z_i}} , \quad (3)$$

which is just the fraction of the group’s total degree that belongs to node i .

Given these parameters, the value of each element of the adjacency matrix is

$$\forall i > j \quad A_{ij} = A_{ji} = \text{Poisson}(\gamma_i \gamma_j \mathcal{M}_{z_i, z_j}) , \quad (4)$$

which implies that the expected number of edges running between i and j is $\gamma_i \gamma_j \mathcal{M}_{z_i, z_j}$. Intuitively, the product $\gamma_i \gamma_j$ plays a similar role here to the degree product $k_i k_j$ in the configuration model. A key difference, however, is that γ_i is a fraction rather than a count, and so the product $\gamma_i \gamma_j$ serves to pick out of the edge count \mathcal{M}_{z_i, z_j} the number of edges we allocate to i, j .

Given Eq. (4) and a choice of parameters $\theta = (c, \vec{z}, \vec{k}, \mathcal{M})$, we draw a simple graph from the ensemble as follows, in $\Theta(n^2)$ time:

1. compute the group-level degrees κ_r by Eq. (2)
2. compute the node-level propensities γ_i by Eq. (3)
3. initialize an empty graph G with n nodes
4. for each pair $i > j$, draw a Poisson random variable r with mean $\gamma_i \gamma_j \mathcal{M}_{z_i, z_j}$, as in Eq. (4)
5. if $r > 0$, add the undirected, unweighted edge (i, j) to G .

The last step collapses the multi-edges to create a simple network. If a multigraph is desired, replace this step with assigning $A_{ij} = A_{ji} = r$. This simplification step will induce small differences in the generated degree sequence and the specified one, but usually, these differences are negligible.

The DC-SBM has more parameters than the SBM, which gives it even greater flexibility in producing a broad variety of large-scale network patterns. In total, there are $1 + 2n + \Theta(c^2)$ parameters, which correspond to the number of values required to specify the number of groups c , a group label for each node \vec{z} , the node degrees \vec{k} , and the groupwise edge counts \mathcal{M} , respectively.

3.2.3 Planted partitions in the DC-SBM

Like the SBM, we can define a planted partition model for the DC-SBM and use it to explore how degree and modular structures interact. As with the SBM, our goal is to develop a one-parameter model that interpolates between more or less strongly modular structure.

To do this, we'll use a slightly different strategy than we did for the SBM. This time, we will define two mixing matrices, one for perfectly assortative communities and one for a random mixing (as in a configuration model), and then to combine these matrices using a parameter λ , like this

$$\mathcal{M}_{rs} = \lambda \mathcal{M}_{rs}^{\text{assort.}} + (1 - \lambda) \mathcal{M}_{rs}^{\text{random}} . \quad (5)$$

When $\lambda = 1$, we use only the assortative planted partition, and as $\lambda \rightarrow 0$, we mix in increasing amounts of configuration model randomness, until, when $\lambda = 0$, all of the structure is entirely that of a random graph with specified degrees. The random mixing matrix should have the form of a configuration model for each pair r, s , and hence we say

$$\forall_{rs} \mathcal{M}_{rs}^{\text{random}} = \kappa_r \kappa_s / 2m .$$

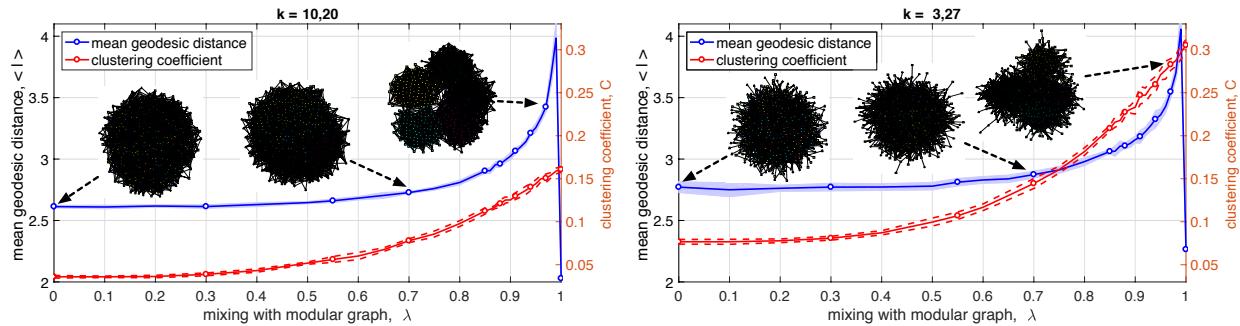
While for the perfectly assortative communities, we set the r th community's degree to be entirely within-group:

$$\mathcal{M}^{\text{assort.}} = \begin{pmatrix} \kappa_1 & 0 & \cdots & 0 \\ & \kappa_2 & \ddots & \vdots \\ & & \ddots & 0 \\ & & & \kappa_c \end{pmatrix} , \quad (6)$$

and set the group sizes to be all equal, $n_r = n/c$. Now, we need only to specify the degree structure \vec{k} . The values k_i can be drawn from any integer probability distribution we like, e.g., a discrete exponential distribution or a discrete power-law distribution (or anything in between). The one constraint on every degree value is that the maximum degree cannot exceed the size of its group, meaning $k_{\max} \leq n/c$, as otherwise, its propensity score would exceed 1, and we would have the impossible situation of $k_i > \kappa_{z_i}$.

Planting partitions for simulation experiments. The value of λ naturally varies from $[0, 1]$, and across this range, the strength of the modular structure will vary considerably. In this version

of our experiment, we will explore how the same network measures vary as a function of λ , for two different choices of a degree distribution: one with low variance, and one with very high variance. For example, consider a $c = 5$ group network on $n = 500$ nodes, where we either (i) set each degree $k_i = 10$ or 20 with equal probability (lower variance), or (ii) set each degree $k_i = 3$ or 27 with equal probability (higher variance). In either case, the mean degree $\langle k \rangle = 15$ is the same, so all that's different between the two settings is the variance, and the structure it induces. Now, as in the SBM experiment, we measure the mean geodesic distance $\langle \ell \rangle$ (lefthand y -axis) and the clustering coefficient C (righthand y -axis) as a function of how strong the community structure is λ , for each distribution.

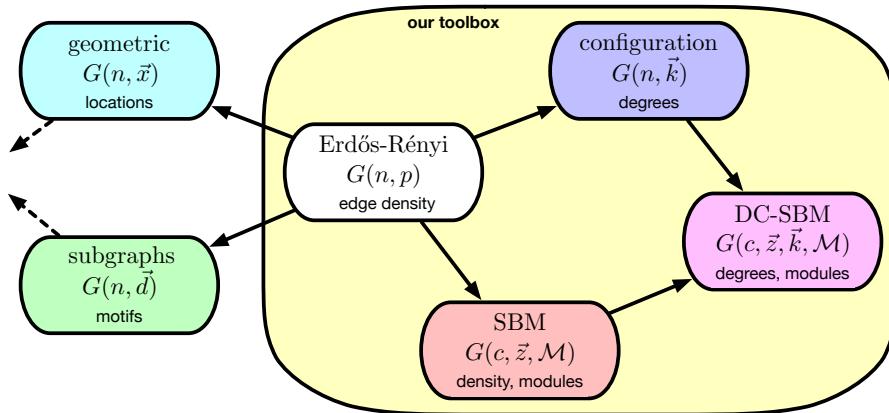


The overall pattern for both degree distributions is similar to what we saw in the SBM: when connections are mostly random (modular structure is weak $\lambda < 0.3$), both network measures are fairly stable, and both increase as the modular structure strengthens beyond this point. A clear difference, however, is that the clustering coefficient C is far more sensitive to the change in variance than is the average path length $\langle \ell \rangle$: it starts off a little higher, even in the purely random graph extreme ($\lambda = 0$), and increases faster and farther as the network approaches the purely assortative community extreme ($\lambda = 1$). Intuitively, the reason is that the higher degree nodes in the higher-variance case will tend to group together, by virtue of their larger degree giving them many more ways they could group together, and this naturally generates more triangles than when edges are distributed more equitably.

One last thing is worth noting about these experiments: even when a large majority of a graph's edges are drawn from the perfectly assortative planted partition $\mathcal{M}^{\text{assort.}}$, the path-length structure of the network is largely driven by the presence of the $\mathcal{M}^{\text{random}}$ connections. It's not until truly extreme values of $\lambda > 0.9$ do we really see the assortativity dominant the randomness. This pattern is a powerful reminder that it does not take much randomness to make a network look like a random graph, at least in terms of its path lengths.

4 Taking stock of random graph models

With the SBM and DC-SBM, and including the Erdős-Rényi and configuration (or Chung-Lu) models, we now have four random graph models in our toolbox. Together, these models specify ensembles of random graphs based on assumptions of randomness plus parameterized (i) edge density, (ii) degree structure, or (iii) modules. And, these models are generalizations of each other. Both the configuration model and the SBM can be viewed as different kinds of generalizations of the Erdős-Rényi model, one to a setting with degree structure and the other to a setting with modules. Combining these two models is what gives us the DC-SBM. We can visualize these relationships as shown below, where the large bubble covers the models we have learned about so far.



It's easy to imagine adding new models to this picture, where model i connects to model j if the latter is a generalization of the former. For example, in a *random geometric graph*, nodes have locations in a metric space \vec{x} and the probability that a pair i, j is connected depends on their distance $d(x_i, x_j)$.¹⁰ This model also generalizes ER graphs, but does so in a different direction than either the configuration model or the SBM. Further elaborations of this model might add assumptions about degree structure or modules, thereby generalizing the configuration model and the SBM, respectively. Or, we could design a random subgraph model in which we specify the motifs of different types that each node is part of \vec{d} .¹¹ This model would generalize ER graphs in still another direction. The point here is to illustrate that there are many ways to build on top of the basic random graph models in our toolbox, by adding new structural assumptions. Doing so is an active area of research today.

¹⁰See Penrose, *Random Geometric Graphs*, Oxford University Press (2003).
<https://books.google.com/books?hl=en&lr=&id=RHvnCwAAQBAJ>

¹¹As in Karrer and Newman, *Phys. Rev. E* **82**, 066118 (2010). <https://arxiv.org/abs/1005.1659>

Returning to our four random graph models, we can now take stock of the degree to which each is able to capture or reproduce the kinds of structural patterns often found in real-world networks. The table below presents the story so far for each of the models, for measures like the edge density ρ , the degree distribution $\Pr(k)$, the diameter ℓ_{\max} or mean geodesic distance $\langle \ell \rangle$, the clustering coefficient C , reciprocity r (directed networks only), and the existence of a very large connected component. Clearly, random graphs with modular structure, and especially the DC-SBM, do a pretty good job of capturing many of the observed network patterns.

measure	pattern	edge density	degrees	modules, density	modules, degrees
		<i>Erdős-Rényi</i>	<i>configuration</i>	<i>SBM</i>	<i>DC-SBM</i>
edge density ρ	sparse	specified	specified	specified	specified
degrees $\Pr(k)$	heavy tailed	Poisson	specified	Poisson mixture	specified
diameter ℓ_{\max}	$O(\log n)$	$O(\log n)$	$O(\log n)$	typically $O(\log n)$	typically $O(\log n)$
clustering C	social: many non-social: few	$O(1/n)$	$O(1/n)$	depends on \mathcal{M}	depends on \mathcal{M}
reciprocity r	high	$O(1/n)$	$O(1/n)$	depends on \mathcal{M}	depends on \mathcal{M}
large component	$\Theta(n)$	if $\langle k \rangle > 1$	if $\langle k^2 \rangle - 2\langle k \rangle > 0$	depends on \mathcal{M}	depends on \mathcal{M}

5 At home

1. Read Chapters 14.1 and 14.4 in *Networks*