1 The stochastic block model

Another popular and well-known choice for a network partition scoring function is the likelihood function of the *stochastic block model* (SBM), which was first studied in mathematical sociology by Holland, Laskey and Leinhardt in 1983 and by Wang and Wong in 1987. Conventionally, this model is defined for simple unweighted networks, and for non-overlapping community assignments. It generalizes easily to directed networks. Versions with weights, overlapping ("mixed") memberships, arbitrary degree distributions, and other features have also been introduced.

1.1 Generative models and statistical inference

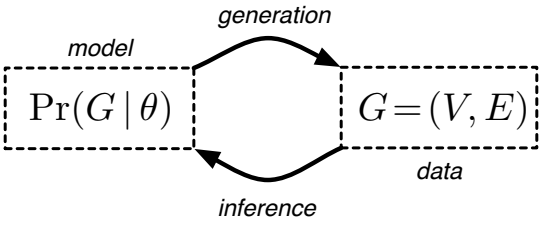
Unlike the modularity function, the stochastic block model is a probabilistic or *generative model*, which assigns a probability value to each pair i, j in the network. Generative models are a powerful way of encoding specific assumptions about the way "latent" or unknown parameters interact to create edges, and offer many advantageous features. For example,

- they make our assumptions about the world explicit (rather than encoding them within a procedure or algorithm),
- their parameters can (often) be directly interpreted with respect to certain hypotheses about network structure,
- they allow us to use likelihood scores, which are based on fundamental principles in statistics and probability theory, to compare different parameterizations or even different models,
- they make probabilistic statements about the observation of (or lack-thereof) specific network features, and
- they allow us to estimate *missing* or *future* structures, based on a partial or past observations of network structure.

These benefits come with some costs, however, the largest of which is that the fitting of the model to the data can seem more complicated than with simple heuristic approaches or vertex-/network-level measures.

Like other generative models, the stochastic block model defines a probability distribution over networks $\Pr(G \,|\, \theta)$, where θ is the set of parameters that govern the edge probabilities under the model.[1] Given a choice of θ, we can then draw or *generate* a network instance G from the distribution by flipping a set of appropriately biased coins. *Inference* is the reverse of this process: we are given some network G, whether generated synthetically from the model or obtained empirically, and we aim to identify the model, or rather the choice of θ, that produced it.

[1] Most generative models for networks are what you might call "edge generative," meaning that they do not consider networks with different numbers of vertices, only networks of fixed size with different patterns of edges.



1.2 Model definition

In its most basic version, the SBM is defined by a scalar value and two simple data structures:

- k : a scalar value denoting the number of groups or modules in the network,
- $\vec{z}$: a $n \times 1$ vector where z_ℓ [or $z(\ell)$] gives the group index of vertex ℓ,
- M : a $k \times k$ stochastic block matrix, where M_{ij} gives the probability that a vertex of type i is connected to a vertex of type j.

Note that k must be chosen before either z or M can be written down. Because each vertex in a given group connects to all other vertices in the same way, vertices with the same label are sometimes called *stochastically equivalent*.

Given these choices, every pair u, v is assigned a probability of forming an edge because every vertex has a type assignment, given by z, and knowing z_u and z_v allows us to index into the matrix M to find the probability that such an edge exists.

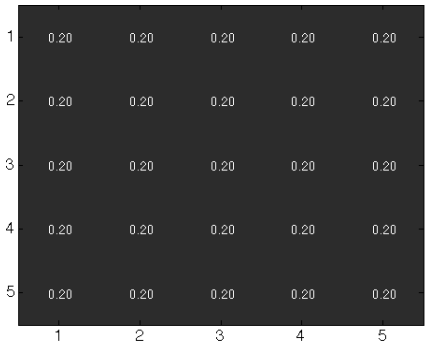
1.3 Generating networks

Given choices for k, z and M, we can draw a network instance from the SBM model by flipping a coin for each pair of vertices u, v where that edge exists with probability $M_{z(u),z(v)}$. In this way, edges are independent but not identically distributed. Instead, they are conditionally independent, i.e., conditioned on their types, all edges independent, and for a given pair of types i, j, edges are iid.

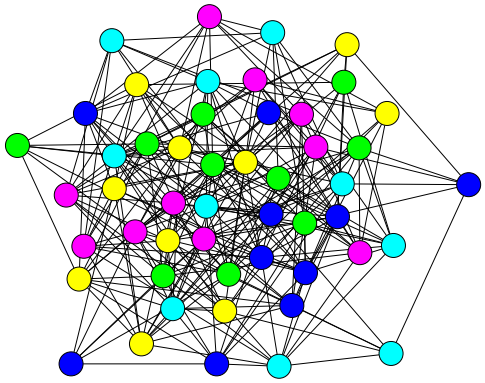
Note that the SBM has a large number of parameters. In the undirected case, it has $\binom{k}{2}$ values in M that need to be specified before we can generate any edges, even if we have already chosen the labeling on the vertices. This flexibility allows the SBM to produce a wide variety of large-scale structures. Before discussing how to infer structure from data, we will explore some examples of how different choices of parameters produce different types of large-scale structure.

Random graphs.
Suppose $M_{ij} = p$ constant for all pairs i, j. In this case, the SBM reduces to the Erdős-Rényi random graph model $G(n, p)$. In this case, all the results from ER graphs, from the calculations of the component sizes to the appearance of the giant component, would hold. For $k = 5$, below is an example of a stochastic block matrix and a corresponding network instance drawn from it.

If the values of M are not all the same, then the SBM generates Erdős-Rényi random graphs within each community i, with an internal density given by M_{ii}, and random bipartite graphs between pairs of communities i and j.



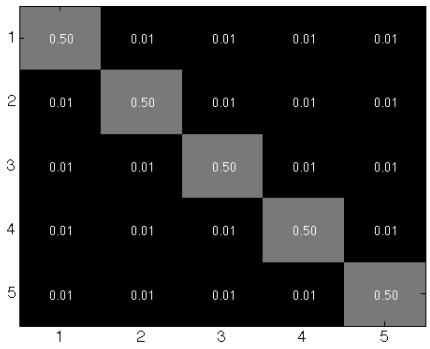
stochastic block matrix



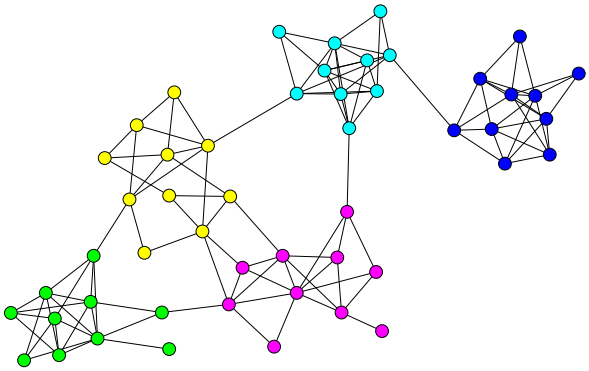
random graph

Assortative and disassortative communities.
When communities are assortative, then vertices tend to connect to vertices that are like them, i.e., there are relatively more edges within communities. Under the SBM, assortative community structure appears as a pattern on M in which the values on the diagonal are greater than the values off the diagonal. That is, $M_{ii} > M_{ij}$ for $i \neq j$. Similarly, disassortative structure implies that unlike vertices are more likely to connect than like vertices, i.e., $M_{ii} < M_{ij}$ for $i \neq j$.

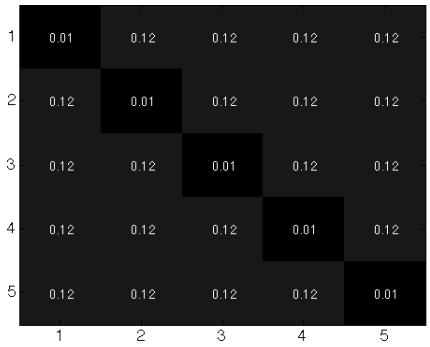
The figures on the next page illustrate these patterns, where each network has the same mean degree. Notably, the disassortative network looks visually similar to the ER network above, but this hides the fact that vertices with similar colors are not connecting with each other. In contrast, the assortative network shows nicely what we normally expect from communities, and what the modularity function Q prefers.



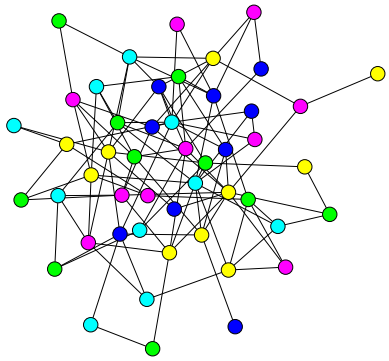
stochastic block matrix



assortative communities



stochastic block matrix

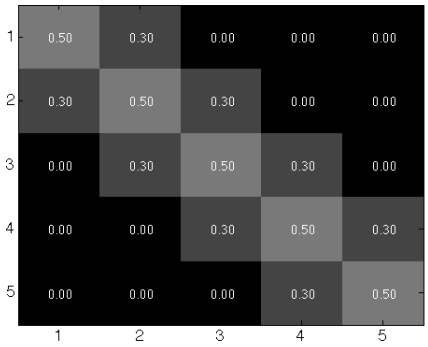


disassortative communities

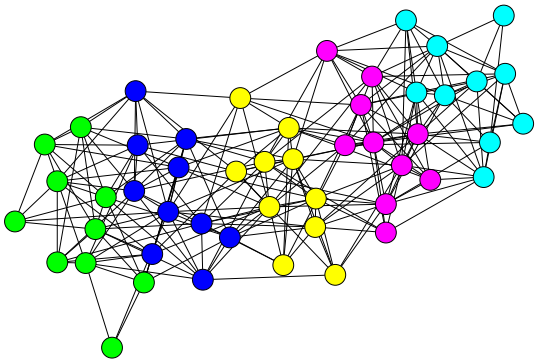
Core-periphery and ordered communities.
In an ordered network, communities connect to each other according to a latent sequence.

Physical proximity networks exhibit this kind of structure with age acting as a latent ordering variable. That is, individuals tend to associate physically with others who are close to themselves in age, so that children tend to be physically proximate to other children, teenagers with teenagers, 20-somethings with 20-somethings, etc. This induces a strong diagonal component in the stochastic block matrix, as in assortative communities, plus a strong first-off-diagonal component, i.e., communities connect to those just above and below themselves in the latent ordering $M_{ii} \approx M_{i,i+1} \approx M_{i,i-1}$. In social networks, an exception to this pattern occurs during the childbearing years, so that individuals split their time between their peers and their children (who are generally 20-30 years younger).[2]

[2]This fact was demonstrated nicely in a longitudinal study in Scandinavia, in which individuals were asked to

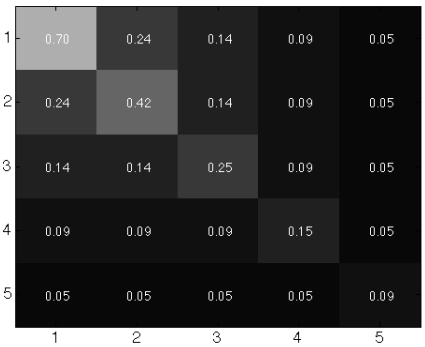


stochastic block matrix

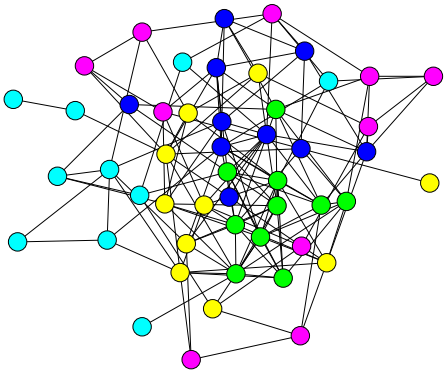


ordered communities

Core-periphery structure is a form of ordering on communities, but where we place the additional constraint that the density of connections decreases with the community index. The following instance shows only one way to specify this structure, in which each layer of the network connects to all other layers, but with exponentially decreasing probability. In the stochastic block matrix, you can see evidence in the upper left corner of the nested structure of this core-periphery network. In the network instance, the green vertices are the inner core, while the magenta and cyan vertices are the outer periphery.



stochastic block matrix

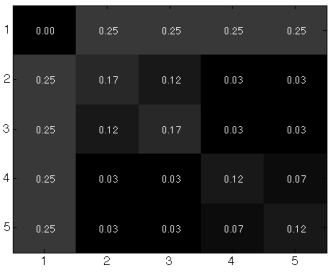


core-periphery structure

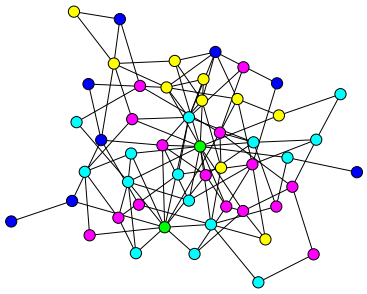
record in a journal the characteristics of the people they associated with at different times of the day. I don't have the reference handy, but will try to find it.

Degree heterogeneity.
The networks the SBM generates are Erdős-Rényi random graphs within the groups, and random bipartite networks between the groups. As such, the degree distribution of the generated networks are always mixtures of Poisson degree distributions. Each bundle of edges contributes to the degrees of the vertices it runs between, and so if its density is large, it will contribute many more edges to the degrees of its end points. We can use this flexibility to create more heavy-tailed degree distributions than we would normally expect from an ER graph by placing a small number of vertices in a group with large densities to other, larger groups.

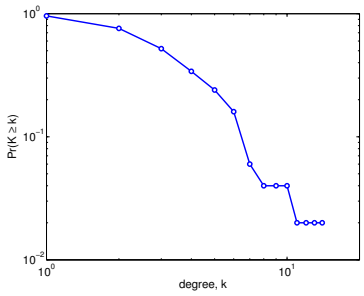
The following example illustrates this idea, where we now modify the number of vertices in each group to be $\{2, 8, 10, 15, 15\}$. In the stochastic block matrix, the smallest group, with 2 vertices (green, in the network image), connects to 0.25 of the other vertices, and thus each of these vertices has expected degree $\mathrm{E}[k] = 12$, which is about twice as large as the expected degree of the other vertices. (Do you see how to calculate $\mathrm{E}[k]$?)



stochastic block matrix



heterogeneous degrees



degree distribution

Directed or undirected.
As a final comment, the SBM can naturally handle directed networks, by relaxing the previous assumption that the stochastic block matrix be symmetric. In this way, the probability of an edge running from $u \to v$ can be different from the probability of an edge running in the opposite direction, from $v \to u$.

2 At home

1. Read Chapter 8 (pages 359–418) in *Pattern Recognition*
2. Next time: fitting block models to data