VRDI 2019 Networks Breakout
Day 2: Null Models

Daryl DeFord

June 25, 2019

1 Introduction

Today we are looking at the concept of null models, which are random graph processes that seek to represent some of the fundamental properties of a network. This builds on our discussion yesterday, since the metrics that we evaluated are going to tell us which fundamental properties you are trying to preserve. The random networks that have been studied from a mathematical perspective have nice theoretical properties that make it convenient to prove results about them but tend to be poor matches for the types of social networks that we encounter in practice. On the other hand, these models tend to not be particularly theoreticlaly tractable.

The idea behind null model analysis is to compare in a "principled" fashion a particular observed network with an ensemble of related random networks in order to determine the features of the observed network that are not likely to be caused by random behaviors. This is very intrinsically related to the idea that in most cases of interest the actual network that we are operating on is only an approximation of a snapshot of the underlying physical system. This document contains descriptions of the basic random network models and a discussion of how to use these models to discover significance from observed networks.

2 Standard Random Network Constructions

Here are a few of the most common network models. You can experiment with their properties using this tool. One thing that is worth considering is how you might modify these models to better reflect our census data.

1. (Erdös–Renyi) Inputs: The number of desired nodes n and a probability parameter p. Construct a network on n nodes where each of the $\binom{n}{2}$ edges independently occur with probability p. Another standard version of this model selects an arbitrary graph from the collection of all graphs on n vertices with m edges uniformly, although this formulation makes it more difficult to calculate some standard network parameters since there is no assumption of independence on the edges.

2. (Barabsi–Albert) Inputs: An initial network, a final number of nodes, and a fixed number c of edges to add for each new node. This is an iterative process. At each step, until the final number of nodes is reached, add a new node to the network. Connect this new node to c nodes already in the graph with probability $\frac{\deg(i)}{\sum_j \deg(j)}$. This preferential attachment process generates scale free networks.

3. (Watts–Strogattz) Inputs: The number of desired nodes n, a probability parameter p, and the desired mean degree d. The construction begins with a ring lattice on n nodes where each node is connected to its $\frac{k}{2}$ nearest neighbors. Visit the nodes sequentially and reattach each edge at that node with probability p. Select the new target for the reattached edges uniformly. This method produces small world graphs.

4. (configuration models) Configuration models are a generalization of the Erdös–Renyi model that preserves the degree distribution of a network of interest. The idea is to cut each edge in the original network in half and reattach these "edge ends" at random. Since the number of ends at each node doesn't change the degree distribution is preserved. This is the null model used in the definition of modularity.

Network Type	Average Degree	Average Path Length	Diameter	Clustering Coefficient	Degree Distribution
Erdös–Renyi	np	$\log(n)$	$\log(n)$	p	Binomial
Barabsi–Albert	c	$\frac{\log(n)}{\log(\log(n))}$	$\frac{\log(n)}{\log(\log(n))}$	$n^{-\frac{3}{4}}$	Scale Free
Watts–Strogattz	k	$\log(n)$	$\log(n)$	$\frac{3}{4}$	Poisson

2.1 Why these models?

While the ER model has the big advantage of independence of edges, the other models are less tractable theoretically. The main two metrics that lead to these models are the notion of average path length and clustering coefficient - two of our metrics from yesterday! Together, these properties characterize what are colloquially known as "small-world" networks, those that resemble empirical social networks by having small average path length and lots of clustering.

Trying to understand the properties of these networks as a class is what led to the formulation of the BA and WS models. We would like to do a similar thing with census data - formulate models that output examples that match the properties of census dual graphs in order to better understand their underlying structure. This is useful both for practical reasons, as test cases for new algorithms, and for theoretical purposes, as it lets us refine the hypotheses that we use to prove theorems.

I should mention that there are other well behaved graph models that are appealing to combinatorialists. For example, there are methods for generating asymptotically uniform regular graphs. There are also some nice connections to algebra here. One reasonable way to draw a nice looking graph is to pick your favorite group and form the Cayley graph of a randomly chosen generating set. This procedure can lead to some quite interesting mathematics as the spectral structure of the corresponding adjacency matrices is connected to the Fourier transform on the group.

2.2 Example: Ramsey Theory

The Ramsey number $R(x, y)$ is the smallest number n such that any graph on n nodes has either x nodes that are all connected to each other or y nodes that have no edges between them. This is usually stated as the minimum number of people that you can invite to a part so that there must either exists x people who all know each other or y people, none of whom have met before. Consider $R(3, 3)$ on graphs with five and six nodes. Is it possible to make a graph on five nodes without this property? How about 6?[1]

We can prove a lower bound on $R(k, k)$ for some fixed k using the ER model as follows. Consider the set of graphs generated by $ER(n, \frac{1}{2})$. The probability that any particular set S of k nodes in the graph is all connected is $\frac{1}{2}^{\binom{k}{2}}$ and the same is true for the probability that there are no edges between them. Thus, the probability that an arbitrarily chosen k subset is either completely connected or completely disconnected is $2^{1-\binom{k}{2}}$. There are $\binom{n}{k}$ such subsets in the graph and so the probability that at least one of them is completely connected or completely disconnected is at most $\binom{n}{k}2^{1-\binom{k}{2}}$ by the union bound. When $n < 2^{\frac{k}{2}}$ this probability is less than 1 and hence there must exist at least one graph that violates the condition. The success of this type of example led to an explosion of interest in these methods among graph theorists and probabilists. Note that the bound we get for $R(3, 3)$ is $2^{1.5} \sim 2.83$, which is definitely lower than the actual value we computed above.

3 Karate & Dolphin Examples

Think about your ego network from Week 1 (luckily it is probably small enough to easily compute statistics on by hand ☺). Is it possible that your network could have been generated by one of these models? Using the Sage widget, generate an example network for each type of null model - do these seem like a good fit for your ego network? Can you come up with a model that is more likely to generate your network? Which of these is the best[2] fit for our census data.

[1] Pick an arbitrary node and use the pidgeon hole principle on edges.

[2] note: best doesn't necessarily mean good ☺

Using the dolphin and karate club networks from yesterday, generate examples of these null models (in python, th documentation for the graph generators is here, scroll down to the random section) attempting to match the parameters of the input network. For each example, find a metric that distinguishes the actual network from the models. These slides have some possible solutions in Section 3.

4 Models with Community Structure

As in the case of the standard models, models that attempt to encode community structure range from those with very nice theoretical properties to those that more closely match observed data. In this case the theoretical model is called the Stochastic Block Model (SBM). This model generalizes the ER model by assigning each node to a group. Then, we select a different binomial parameter for each pair of groups and for each pair of nodes, draw an edge independently using the parameter associated to their groups. Usually, the parameters are chosen so that edges between members of the same group are much more likely than edges between groups. This is an example of an assortative preference[3]. These SBMs have nice statistical properties and are frequently used as a background model for testing the existence of community structure.

Another model that I personally fond appealing is the dot product model, which allows us to incorportate degree heterogeneity into our community structure. This family grew out of the study of intersection graphs and their generalizations, like interval graphs, circular arc graphs, and the other 20 examples listed on Wikipedia. The basic construction is that the nodes are represented by sets and two nodes are connected with an edge if they have non-empty intersection. Here is a quick proof that all graphs are intersection graphs: let the elements of the sets be the edges of your graph and let each node be represented by the set of edges incident to it. As with all of the combinatorial questions we addressed yesterday, once we have settled the issue of existence we want to ask the extremal questions: what is the smallest number of elements needed to represent a given graph as an intersection graph[4].

Our interest is in a specific generalization to dot product graphs. Here, we represent each node with a vector of length k and connect two nodes if the dot product of the corresponding vectors is at least[5] 1. As with standard intersection graphs, every graph is representable as a dot product graph: stealing the same proof technique, use $k = \binom{n}{2}$ dimensions and represent each node with a vector that has a 2 in the position that corresponds to each incident edge. Now the extremal question becomes - what is the smallest dimension necessary to represent a given graph?

So far, we haven't introduced any randomness, so let's do that now. Instead of trying to represent a graph with vectors we will instead draw vectors according to some probability distribution over $\mathbb{R}^k$ and consider the induced distribution over graphs formed by taking the respective dot products[6]. The dot product is particularly convenient because $\langle x, y \rangle = ||x|| \cdot ||y|| \cdot \cos(\theta_{x,y})$. We can think of the magnitude of each vector representing its likelihood to form connections and the angle representing the similarity of com unity assignment.

Consider a distribution centered around the coordinate axes, with a small amount of noise that pushes the vectors into the first quadrant. If we pick one axis per community we can encode the community structure and if we place a long tailed distribution over each axis, we can have structure within the communities. Similarly, if we are given a graph, we can estimate the assignment of nodes to vectors and generate other similar graphs - exactly the purpose of a null model. Many colorful figures demonstrating these examples are available in this paper.

This example highlights the back and forth that occurs between generative models and inferential procedures. Once we have built a generative model, we can study the expected properties of objects drawn from that model. Conversely, we can also take a given object and try to discover the parameters that were most likely to have given rise to that object, assuming that it came from the model. As we have seen, for networks both of these questions are relevant and important.

[3] often called homophily in the social science literature

[4] This is another fun problem you could spend the rest of your life studying.

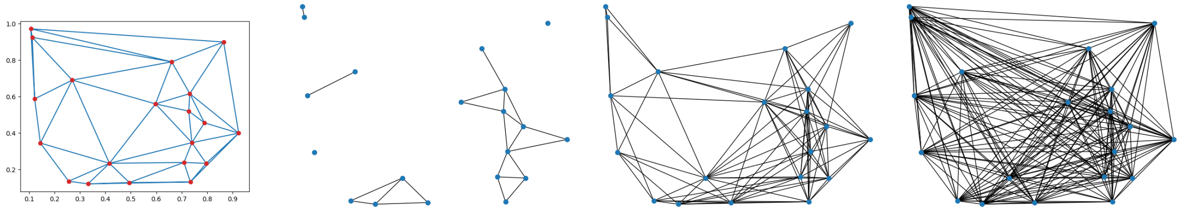
[5] sometimes exactly

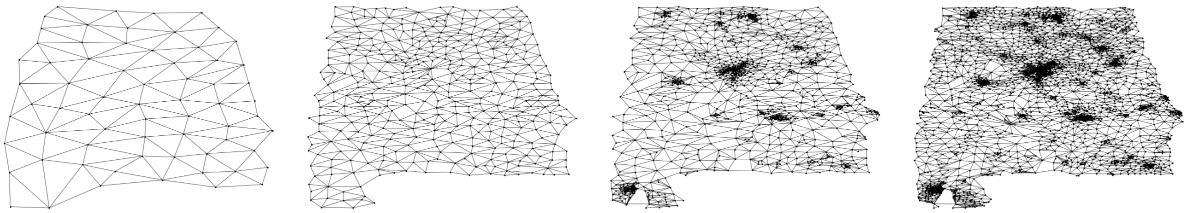
[6] We may also draw a binomial variable proportional to the dot product to further smooth things out

5 Geometric Models

Another popular class of models that may be of use to us is the family of geometric, or latent space models. The dot product modes introduced above are an example of this approach, although most latent space models use distances between points instead of dot products. One common method is to simply distribute a collection of random points in the plane representing the nodes and then connect every pair of nodes that lies within some fixed radius. As in the dot product example, we could instead relax this notion of adjacency where we draw from a binomial distribution proportional to inverse distance and instead generate a whole family of graphs from a single assignment of points. Similarly, given a specific graph, we might try to find an embedding of the nodes into the plane that would likely give rise to that graph. Try this out with your ego graph, using the length of a pen cap as your connection radius. Is it possible to embed your graph in the plane? What could you imagine going wrong? How many dimensions does your friendship need?

From a more practical standpoint, we could simply compute the triangulation of any collection of points in the plane. Some code for doing this is included in the repo for this breakout. Can you come up with better ways to distribute points in the plane to get dual graphs that look like our census examples? These graphs probably have a few too many triangles to match our census data perfectly. The triangulation function compares the triangulation to the distance version. Try varying the radius - is it ever possible to get the same graph with both models? Can you construct a set of points and a radius that gives the same graph?





6 Census Data

Take the (up to) four dual graphs for your state and compute their statistics. How well do they match up with any of the models that we have discussed? Just like with the dolphins and karate club, try to use the models we have generated to construct null models that should be similar to the state graphs. Are any of them reasonably close? For most of them you should be able to find examples of metrics that aren't particularly close to the actual values. This gap between existing models and our oberved data is one of the things that is motivating our week 4 project - can we come up with a better model for these graphs?

One approach that seems promising is to use very dense grids as proxies for the underlying geography, and try to figure out ways to form "census blocks" that could then be aggregated into larger units. The Networks Breakout Repo has some examples of dual graphs that can be constructed on grids as a proxy for partitioning the underlying geometry. How well do these graphs compare to yours? Can you come up with a better model?