Readings

Newman, 12.1-12.6, (Optional): 13.1-13.2.

Problem 1.

Suppose that we'd like to generate an Erdős-Rényi random graph $G(n, p)$. Assume that we incur a small constant cost each time we want to generate a random number. You can think of this cost as computation time. So, a computation that requires $10,000$ random numbers takes twice the time of a computation that requires $5,000$ random numbers. You can assume that the cost of a random number is the same whether we need to flip a weighted coin, sample from a Poisson distribution, or choose a random element from a discrete and finite set (perhaps surprisingly, this is approximately true).

Part (a)

What is the cost, in terms of quantity of random numbers needed, to generate $G(n, p)$ using the method specified in its definition? That is, we check each pair of edges and flipping a weighted coin with success probability p.

Part (b)

Suppose we are generating a sparse Erdős-Rényi model, in which $p = c/(n-1)$. Consider now the following alternative algorithm.

  i. Sample a random number $M \sim \text{Binomial}(t, q)$, where t is the number of trials and q is the success probability of the binomial distribution.

  ii. Then, choose M pairs of nodes **without replacement**, and place an edge between each of those pairs.

Prove that the graph generated by this algorithm is, asymptotically, $G(n, c/(n-1))$. To do this, you should:

- Determine the correct value of the binomial parameters t and q.

- Prove that the presence of an edge on each pair of nodes is independent of the presence of an edge on any other pair.

- Prove that the probability of an edge being present on a pair of nodes is $c/(n-1)$, as needed.

Part (c)

What is the cost (in random numbers) of the alternative algorithm? How does that compare to the cost you calculated in Part (a)?

Part (d)

Suppose now that we wish to sample from a *Chung-Lu* model (as described in lecture and by Newman on 12.1.2). Recall that the Chung-Lu model has parameters c_i for each $i \in N$ giving the expected degree of each node. Recall also that the Chung-Lu model allow self-loops and multi-edges, unlike the Erdős-Rényi model.

   Propose a modification to the alternative algorithm in Part (b) for sampling from a Chung-Lu model. You should explain each of your choices, and show that the expected degree of node i in your proposed modification is equal to c_i. However, a formal proof is not required.

Problem 2.

Newman, 12.16

Problem 3. (3 points)

Yes, this problem is worth the equivalent of **3** regular problems.

Submit your code and outputs from Short Essay 3. Your submission should include, in a single **PDF** document,

- All code used in your solution.

- A histogram or scatterplot of the degree distribution.

- Output clearly showing the clustering coefficient for your network, and a comparison to the clustering coefficient of an Erdős-Rényi random graph with the same degree distribution.

- A visualization of the network you chose, in which nodes are colored according to a community-detection algorithm and sized according to their betweenness centrality.

If you are using a Jupyter Notebook to work through the computational components of Short Essay 3, a PDF rendering of this notebook that contains both code and output will be sufficient for full credit on this problem. Other approaches that still result in a PDF containing legible code and outputs are also fine.

Problem 4.

Please share the **names of your project group members** and your fun **group name**.

In one brief paragraph, please describe your scientific question and some of the ideas you intend to use to address it. Extreme detail is not required. It is fine for you to write the paragraph as a group and submit the same paragraph in each of your assignments.

If your proposed group contains fewer than 3 or more than 4 people, please explain why.

Problem 5.

Newman 13.7

    Note: This is a challenge problem. It involves doing some reading in Chapter 13 that we won't cover directly in lecture, and has many parts. I recommend only attempting this problem after you've completed the others, and then only if you are trying to grab some extra points.