Simulation of Complex Systems

Homework 5: Community structure and dynamics on networks

Assessment date: 12th of December

In this exercise we take a look at some effects of network structure on disease spreading as well as the spectral method for community detection. Note: parts of this exercise requires implementing the models done in the previous Networks homework.

The existence or absence of communities is one of the most interesting structural features of networks and has been studied extensively for social networks, citation networks, the web, and metabolic networks, among others. In this homework we follow Newman [1] in defining the modularity

$$Q = \frac{1}{4m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) s_i s_j.$$

The problem of finding communities then becomes a problem of assigning values $s_i \in \{-1, 1\}$ for each node in such a way as to minimize Q.

In Homework 2 we studied the SIR-model on a lattice, a large world where communication only takes place between close neighbors. This is not how the real world looks like, not even on the global scale any more; real social networks, upon which diseases spread, have small world structure with long-ranged connections. This has implication both for the direct dynamics of diseases and strategies we use to combat them.

The simplest implementation of the SIR-model on a network is a straightforward translation of the lattice version, with infected agents infecting susceptibles they are connected to with a rate β and recovering with rate γ.

However, the late time properties of this model can be more easily analyzed through a corresponding bond percolation model. Consider an infected

agent A connected to a susceptible agent B. Depending on the rates β and γ, there is some probability that A recovers before infecting B. By letting the time step go to zero (holding the rates constant in some units of time) both processes become exponential and we can calculate the aforementioned probability by integrating over all time. The result is

$$\phi \equiv \Pr[\text{A infects B}] = \frac{\beta}{\beta+\gamma} = \frac{1}{1+1/R}$$

where $R = \beta/\gamma$ as usual.[1]

Thus, in terms of the late time properties such as the fraction of ultimately recovered agents R_∞, the original model is equivalent to removing each edge with probability $1-\phi$ (no spreading before recovery) and then infecting every agent still connected to the initially infected agent(s) through any path, see Fig. 1.[2] But this is exactly finding clusters after a bond percolation process.

Examination: Work in your assigned groups. During lab hour, either 12/12 or the extra lab 17/12, you should together demonstrate your results to a tutor in the way indicated at the respective exercise. (We might be a bit flexible here, in the sense that you can have your work assessed also in the other lab sessions if you have a compelling reason for doing so. But sticking to the schedule is the preferred options as that will make things run more smoothly). Also, when you have had your work assessed, email your code to `kolbjorn@chalmers.se` with "SoCS HP5" and your name in the subject.

Make sure you go through your demonstration by yourself before so that everything works. Everyone involved will appreciate the reduced queue times. Feel free to show just a subset of the exercises if you haven't done them all (whether you plan to do so later or not).

Exercises:

1. Use the spectral method of Newman [1] to identify the clusters of the network in the file communityExample.txt and calculate the resulting

[1] Note that we once again have a dependence only through R, as we in constructing the network assume the agents have time to meet all their neighbors during the relevant time scale.

[2] This is not strictly true as the probabilities are not independent in the original model. However, the agreement is quite good for reasonable parameters. Further, the percolation model corresponds to the infected recovering after approximately the same time span rather than an exponentially distributed one, which is more plausible biologically for most diseases.

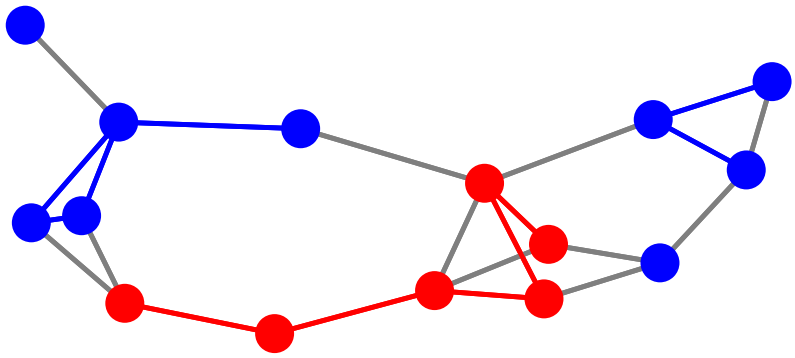

Figure 1: SIR on a network through bond percolation. Gray edges are those randomly removed to signify possible infections that failed, red nodes and edges illustrate actual infections while blue are susceptibles and their remaining connections. The infected are all members of one cluster.

modularity. Do a color-coded graph plot of the results. Hint: the optimal modularity reached with the method is $0.xxx678$, while the first split gives $Q = 0.xxx372$.
To demonstrate: The graph plot and the value of the modularity. **(6p)**

2. Implement the percolation version of SIR on a network. Choose one of your models from homework 3. For varying R, remove each edge with probability $1 - \phi(R)$ and calculate the size of the largest cluster S (its number of nodes) in the resulting networks. You can probably reuse parts of your average path length calculation from homework 3 for this calculation. Plot the size as a fraction of network size versus R. This corresponds to the R_∞-R plots demonstrating the epidemic threshold we drew for the regular SIR model.
To demonstrate: The S-R plot. **(7p)**

3. Investigate the effects of vaccination in your model. For varying f, remove a fraction f of the *nodes* from a network and calculate the size of the largest cluster S. Compare the effects of two schemes, removing nodes at random and removing the nodes with the highest degree, for random graphs and preferential growth networks. Is the difference

larger for either of the types? Make sure you do the calculations for different realizations of the networks (but with the same parameters) to reduce noise.
To demonstrate: Plots of largest cluster size S versus f for the two network types in both vaccination schemes (draw the two schemes as different lines in the same graph, one graph per network type). **(7p)**

4. In the real world we do not know the whole social graph and vaccinating high degree nodes is not possible. A proposed strategy is to pick nodes at random and vaccinate one each of their neighbors, the point being that the expected degree of a neighbor so chosen is higher than the original node.[3] The question is how effective this is. Calculate and compare the degree and excess degree distributions for one random and one scale-free network. What does this say about the expected effects of neighbor vaccination? Implement the vaccination scheme and compare with the results from exercise 3.
To demonstrate: Histogram of the two degree distribution for the networks and the S-f plots comparing the three schemes (draw them as different lines in the same graph). **(5p)**

References

[1] M. E. J Newman. Modularity and community structure in networks. *arXiv:physics/0602124*, February 2006. Proc. Natl. Acad. Sci. USA 103, 8577-8582 (2006).

[3] If the degree distribution of the network is $\{p_k\}$, the excess degree distribution (the distribution the degrees of neighbors of nodes) is $q_k = (k+1)p_{k+1}/\langle k \rangle$.