AMSC808N-Final Exam-F2

Chenyang Fang
https://github.com/Chenyang-Fang/AMSC808N

December 21, 2020

1. Consider a SIR model for propagating infectious disease over a random graph with degree distribution $\{p(k)\}$ such that there exist a giant. Let T be the probability for any given edge to transmit the infection. According to the lecture note, we can get the critical transmission is as following:

$$T_c = \frac{1}{G_1'(1)} = \frac{G_1'(1)}{G_0''(1)} = \frac{\sum_{k=1}^{\infty} kp_k}{\sum_{k=1}^{\infty} k(k-1)p_k} \tag{1}$$

For $T > T_c$, we have a giant component connected by transmitting edges(an epidemic).
For $T < T_c$, all components are small(no epidemic).
Based on the definition, we can get

$$\langle k \rangle = \sum_{k=0}^{\infty} kp_k \tag{2}$$

$$\langle k^2 \rangle = \sum_{k=0}^{\infty} k^2 p_k \tag{3}$$

Then the critical transmissibility T_c can be expressed via $\kappa := \langle k^2 \rangle / \langle k \rangle$ the ratio of the second and the first moments in the original graph:

$$T_c = \frac{\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle} = \frac{1}{\kappa - 1}. \tag{4}$$

Assume that $T > T_c$, an eqidemic is possible. Suppose that we vaccinate a fraction v of randomly selected nodes. Then the probability of a node gets infected is not only depending on the edge transmission but also depending on the vaccination. So the vaccination transmissibility T_v should be

$$T_v = T(1 - v). \tag{5}$$

When we compare the T_v with the T_c, we can get the an epidemic is possible when

$$T_v > T_c \to v < 1 - \frac{T_c}{T} \tag{6}$$

which means if the fraction of the vaccination is smaller than $1 - \frac{T_c}{T}$, then the epidemic may occur. If the fraction of vaccination is greater than the threshold $1 - \frac{T_c}{T}$, then the epidemic will no happen. So the critical fraction of nodes v_c is $v_c = 1 - \frac{T_c}{T}$.

2. (a) Consider an infinitely large graph with power-law degree distribution:

$$p(k) = \frac{k^{-\alpha}}{\zeta(\alpha)}, \quad k = 1, 2, \ldots, \quad , \tag{7}$$

where $\zeta(\alpha) := \sum_{k=1}^{\infty} k^{-\alpha}$ is the Riemann zeta-function. Based on the lecture note, we can get

$$Li_{\alpha}(x) = \sum_{k=1}^{\infty} \frac{x^k}{k^{\alpha}} = polylogarithm \tag{8}$$

$$G_0 = \frac{Li_{\alpha}(x)}{Li_{\alpha}(1)} \tag{9}$$

$$G_1 = \frac{G_0'(x)}{G_0'(1)} = \frac{Li_{\alpha-1}(x)}{xLi_{\alpha-1}(1)} \tag{10}$$

According to the lecture, we can get

$$H_0(1;T) = 1 - S(T) \tag{11}$$

$$S(T) = 1 - H_0(1;T) = 1 - G_0(u;T) \tag{12}$$

$$u = H_1(1;T) = G_1(u;T) \tag{13}$$

For this problem, we can get

$$S = 1 - G_0(u), \quad u = G_1(u). \tag{14}$$

Suppose that $\alpha = 2.2$, we can use the build-in function **fzero** in the matlab to solve the function $u = G_1(u)$. Then we can get $u = 0.1963$, if we substitute the value of u in to the formula $S = 1 - G_0(u)$, then we can get $S = 0.8622$.

(b) When an epidemic happened, it means that we have a giant component connected by transmitting edges. In this problem, calculating the fraction of nodes affected by the epidemic if it occurs means we need to calculate the fraction of the giant component. Same as previous question, we can get the formula of the S.

$$G_0(1 - T + Tu) = G_0(u;T) = 1 - S(T) \tag{15}$$

$$u = G_1(u,T) = G_0(1 - T + Tu) \tag{16}$$

Set $\alpha = 2.2$ and $T = 0.4$. Based on the procedure of part(a), then we can get the value of u, which is $u = 0.2883$. If we substitute value of u in the formula of S, then we can get $S = 0.4078$.

(c) Based on the problem 1, we can get the $v_c = 1 - \frac{T_c}{T}$. According to the lecture note, we can get the formula of the critical transmissibility T_c is

$$T_c = \frac{\sum_{k=1}^{\infty} kp_k}{\sum_{k=1}^{\infty} k(k-1)p_k} = \frac{\zeta(\alpha-1)}{\zeta(\alpha-2) - \zeta(\alpha-1)} \tag{17}$$

If $\alpha \leq 3$, then $T_c = 0$. Here $T = 0.4$, then we can get the critical fraction v_c as

$$v_c = 1 - \frac{T_c}{T} = 1. \tag{18}$$

which means the epidemic will happen unless all the nodes get vaccinated.

3. (a) In this problem, for the generated finite graphs, find the average fraction of vertices in the giant component. The given routine **MakePowerLawRandomGraph** is used to generate the random graph. The giant component is a connected component of a given random graph that contains a finite fraction of the entire graph's vertices. In the matlab code, I used the build-in function **conncomp** to calculate the size of all the components in a random graph. Then I chose the largest component as the giant component and calculate the fraction of the giant component in a random graph. Then I run the code 100 times and calculate the average fraction of vertices in the giant component. Based on the matlab, we can get the final result:

$$\bar{S} = 0.8214. \tag{19}$$

(b) For a given $T = 0.4$, the average fraction of nodes affected by an epidemic is the fraction of giant component. Because the generating function of transmitting edge follows the Bernoulli distribution:

$$f(x|T) = \begin{cases} 1-T & x = 0 \\ T & x = 1 \end{cases} \tag{20}$$

So in the matlab code, I used build-in function **binornd** to generate a random Bernoulli matrix which has the same size with the egdes. For each edge, if the Bernoulli random variable is 1, it means the edge is connected to the giant component. If the random variable is 0, it means that the edge is not connected to the giant component. After that, I calculated the fraction of the giant component which is the fraction of nodes affected by an epidemic. Then I repeat this procedure 100 times and calculate the average fraction of nodes. The final result is

$$\bar{S} = 0.3698. \tag{21}$$

(c) In the problem 1, we have compared the relationship between T and T_c. If $T > T_c$, an epidemic is possible. In this problem, to find out the critical transmissibility T_c, I tried different value of T and plotted the average fraction of nodes affected by an epidemic in the figure 1. According to the figure 1, we can find the development of the size of the giant component with the increasing of transmissibility. In the matlab, the basic procedure is same as part (b). For the parameter T, T is from 0.01 to 0.4 at a stepsize of 0.01.
In the figure, the blue line is the actual fraction of giant component with increasing of T. From the figure, we can find that $T = 0.03$ is a inflection point. Before $T = 0.03$, the increment of the fraction is very slow, after $T = 0.03$, we can find that fraction will increase quickly. To clearly note the difference, I draw a line in the figure which is the orange line. The orange line shows the development of fraction of the infected nodes in the rate of $T < 0.03$. So, from the figure, it is shown that critical value $T_c = 0.03$.

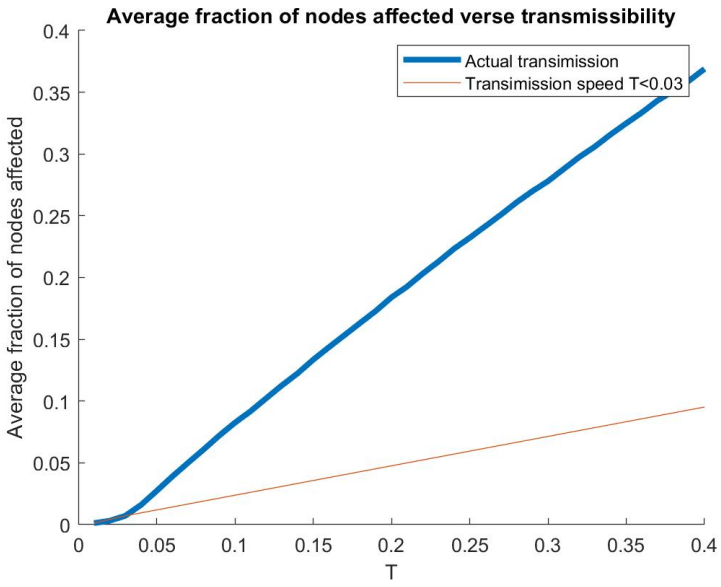

Figure 1: Average fraction of infected nodes for different transmissibilities.

(d) Based on the problem 1, we can get the critical fraction of vaccination v_c is $v_c = 1 - \frac{T_c}{T}$. If we substitute the value of T and T_c from the problem 3 (c), we can get

$$v_c = 1 - \frac{T_c}{T} = 1 - \frac{0.03}{0.4} = 0.925 \tag{22}$$

4. In this problem, run the discrete-time SIR model on a graph from the previous item starting from a single infection node. The simple epidemic model uses three states to describe the spread of an infection: the susceptible (S), the infected (I), and the recovered (R). In the problem, assume that each infecting node remains infecting for one time step. So the basic procedure of discrete-time SIR model consists of these steps:

   - **Susceptible**. In the random figure, start from a single infecting node. According to the lecture note, we can label all the uninfected nodes as white. To distinguish the infected nodes in different steps, I label the new infected nodes as black and label the old infected nodes as gray. The probability for any given edge to transmit the infection $T = 0.4$. Same as before, we can use the build-in function **binornd** to generate a Bernoulli random variable for the edges.
   - **Infected** Starting from a single infecting node, for the new infected nodes, we change the color of new infected nodes from white to black. In the next step, these new infected nodes will infect other nodes.
   - **Recovered** Because that each infecting node remains infecting for one step. After the infection, the gray node will change to white in the next step which is uninfected. The black node will change to gray.

   Based on these steps, we can simulate the development of epidemic in the matlab. To test the performance of the matlab code, first, I set $n = 100$, and time step $t = 100$. After we run the code once, we can get the evolution of the fraction of infection nodes with time steps. Then I plotted the results in the figure 2 and 3.

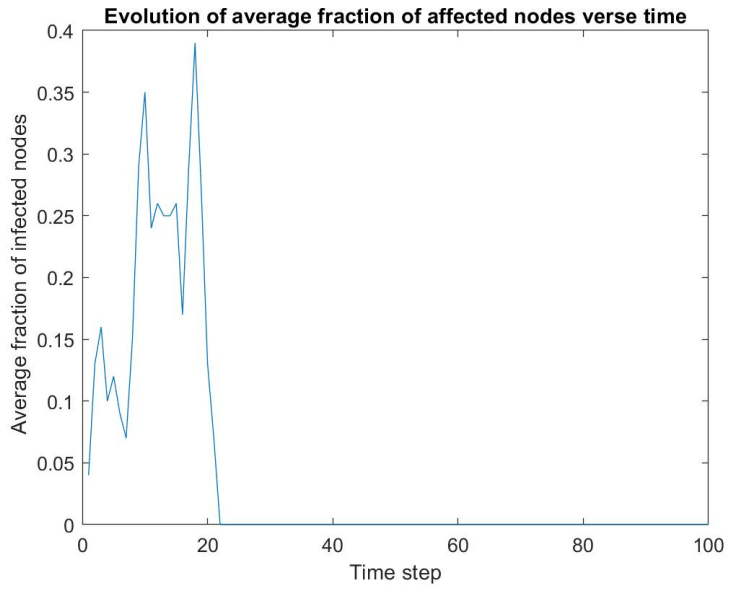

Figure 2: Evolution of average fraction of infected nodes with time steps.

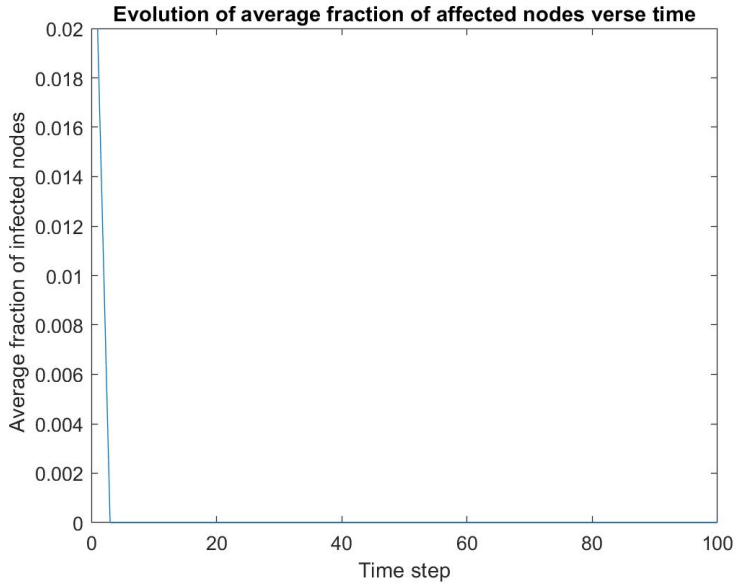

Figure 3: Evolution of average fraction of infected nodes with time steps.

Due to the randomness of the generated graph and the initial node we choose, there are two situations may happened. First, the epidemic may happen and the fraction of infection nodes will increase. Then because the transmissibility T is not large, the epidemic will disappear finally, which is shown in the figure 2. The second condition is the epidemic will not happen, the fraction of infected node decrease directly. Then a large scale experiment is conducted. To eliminate the effect of randomness, I run the code 100 times. Each time, set the number of node $n = 1000$ and time step $t = 1000$. Then the final results are plotted in the figure 4.

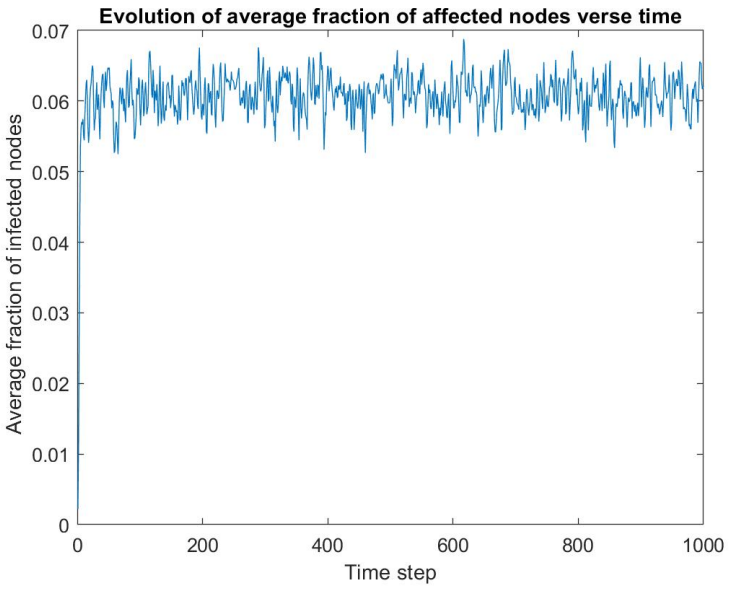

Figure 4: Evolution of average fraction of infected nodes with time steps.

Based on the result, we can get that with the long run, the fraction of infected nodes will keep at a certain level and $\bar{S} \approx 0.06$. The epidemic will not happen, while the virus will no disappear.
Here are some comments about the relationship between the SIR model and the BFS:

- The SIR model is a kind of BFS model. Because in the SIR model, each time, we will explore the neighborhood of the given nodes. This is similar to the BFS method.
- The SIR model is different form BFS method. First: for some given nodes, when we use the SIR model to search the adjacent nodes connected to them, we need to consider the transmissibility T. It means that the adjacent nodes may not be searched.
- Second, In the BFS method, each node can only be searched once. While in the SIR model, the nodes can be searched multiple times.