AMSC808N Final Exam - Problem 2

Guilherme de Sousa (gdesousa@umd.edu)

December 19, 2020

All codes for Problems 1 2 are available on ELMS together with this report submission. The chosen language was Python for simplicity. Additionally, one can find the codes for Final exam Problema 1 and 2 at Guilherme's Github

Question 1.

On a different report.

Question 2.

The goal of this question is to model and test numerically the SIR (*Susceptible* → *Infected* → *Recovered*) model in a random network. The degree distribution of the graph is given by a power-law distribution of the form $p_k \propto k^{-\alpha},\ \alpha = 2.2$.

In the present simulation I'll be using a particular and suitable definition to test epidemic conditions. Throughout the report I'll use *# of infected per time* as the total number (or fraction) of infected people at a given time. And *# of total infections* as the total number of infected people after the disease cease to exist (equivalently, the number of recovered people).

As indicated by the question, each individual remain sick for 1 unit of time and it can transmit to its neighbors with probability T. After being infected and recovered I considered that node as *immune* to a second infection. Similarly, when considering a vaccination rate, any previously vaccinated node cannot get infected nor transmit the disease.

(a) First, let's consider a theoretical description of the problem. Following instructor's lecture notes we know that a critical transmission probability if given by $T_c G_1'(1) = 1$, where $G_1(x) = xG_0(x)/G_0(1)$. From that we can easily show that

$$T_c = \frac{\sum_{k=1} kp_k}{\sum_{k=1} k(k-1)p_k} = \frac{\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle} = \frac{1}{\kappa_0 - 1}, \quad \kappa_0 = \langle k^2 \rangle / \langle k \rangle \tag{1}$$

Where κ_0 is calculated with respect to the underlying degree distribution of the network.

Now we can model a vaccination similarly to the percolation theory on random graphs. Each vaccinated node is effectively disconnected from the network as it does not transmit the disease. From percolation theory we know what is the critical probability to destroy the giant component given a initial network

$$p_c = 1 - \frac{1}{\kappa_0 - 1}. \tag{2}$$

To apply Eq.2 to our model we need to note that κ here denotes the ratio with respect to the infected giant component.

$$\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} = \kappa - 1 = \frac{\frac{d^2}{dx^2} G_0(1 - T + Tx)|_{x=1}}{\frac{d}{dx} G_0(1 - T + Tx)|_{x=1}} = T(\kappa_0 - 1)$$

Thus we can find the critical vaccination fraction

$$v_c = 1 - \frac{1}{\kappa - 1} = 1 - \frac{T_c}{T} \tag{3}$$

note this equation is only meaningful for $v_c \geq 0 \Rightarrow T > T_c$. This makes sense because if $T < T_c$ then there is no epidemic in the first place, so no need for vaccination.

(b) Using the method described above and the self consistency equation to find the fraction in the giant component we can answer the following 3 questions:

- Fraction in the giant component S
- Fraction of infected in the pandemic $S(T)$
- Critical fraction v_c

For giant component fraction we need to solve numerically the following equation

$$u = G_1(1 - T + Tu), \quad S = 1 - G_0(1 - T + Tu) \tag{4}$$

Using Python NumPy built-in function *root* to solve for u I found the following results when $\alpha = 2.2$ and $T = 0.4$

T_c	S	$S(T)$	v_c
0.16	0.72	0.26	0.60

Table 1: Note that the fraction of infected nodes is smaller than the fraction in the giant component. This indicates the pandemic is not that strong.

Even if we only have 26% of infections, we still need to vaccinate %60 of people.

(c) When implementing this model I used graphs up to 5000 nodes because 10^4 nodes were taking the order of hours to run.

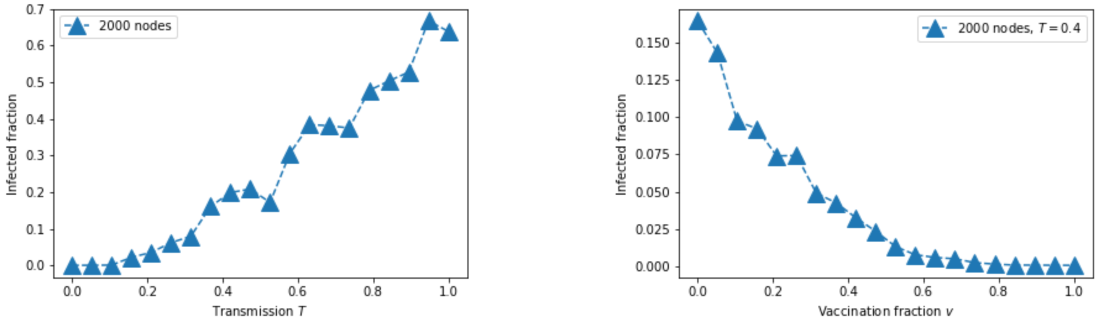
Using $\alpha = 2.2$ and $T = 0.4$ for 2000 nodes and 1000 trials I found

(a) Fraction in the giant component $S = 0.81 \pm 0.03$

(b) Fraction of infected nodes $I = 0.2 \pm 0.2$

which is close but not equivalent to the numerical value found for $S = 0.72$. One possibility is because 2000 nodes is a small number to be able to approximate a infinite size graph. For the infected ratio we got a smaller number but large uncertainty. Possibly I would need to run with a very large number of repetitions to get good statistics.

To find the simulation values for T_c and v_c I used a graphic method. So I plotted the values of infected people for different transmission rates (and no vaccine) to spot the moment where an epidemic surges. Also, fixing $T = 0.4$ changed the values of vaccination rate to find the moment when epidemic dies. Here are the results

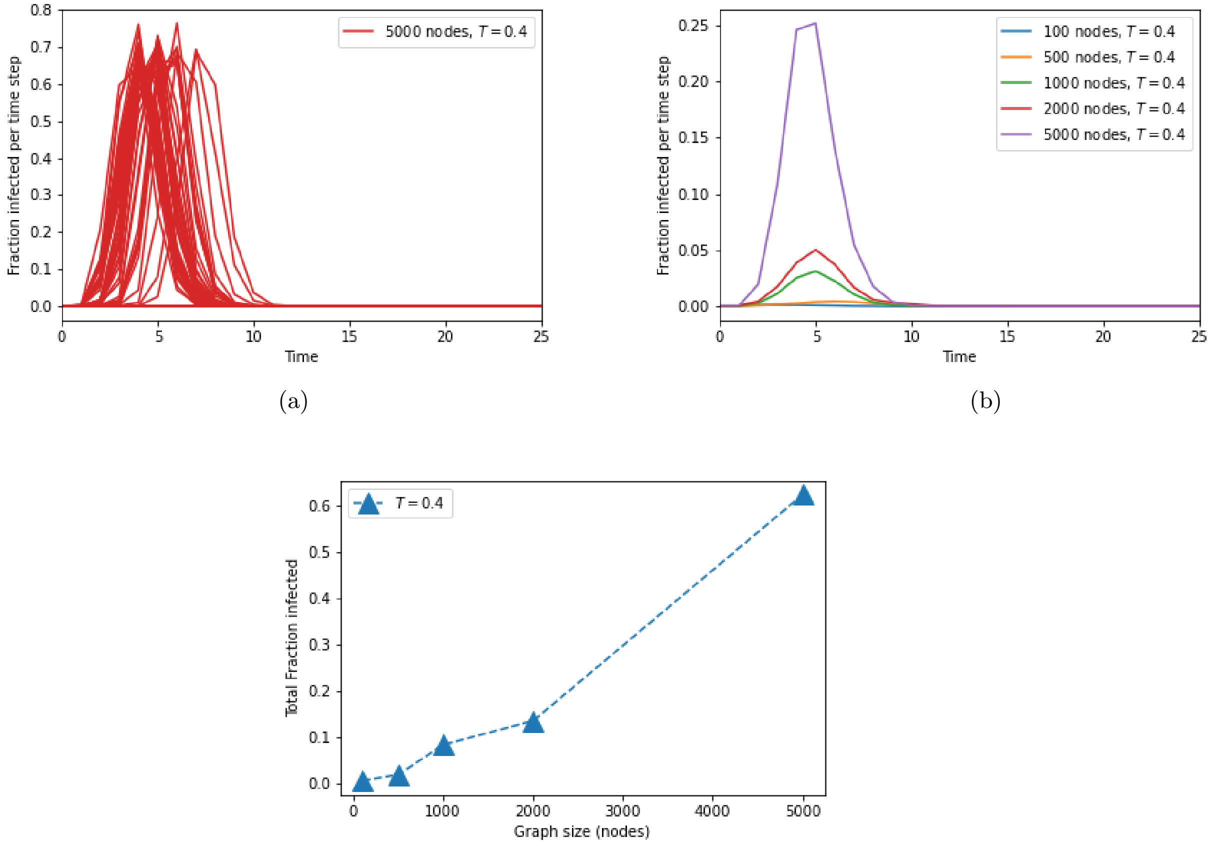


(a) (b)

Figure 1: 1a From a naive analysis we can conclude that $T_c \approx 0.2$ is the threshold for beginning of large epidemic outbreak. 1b To get a outbreak smaller than 2% we need a vaccination fraction larger than 60%.

From Fig.2 we can get a result that is close to the predicted values $T_c = 0.16$ and $v_c = 0.6$. For a better agreement we would need to collect more statistics and use a larger graph, but as mentioned this increases too much the computational cost.

(d) Again I used the method described above, where a recovered person cannot be infected anymore. The plots below indicate the number of people infected at any given time.



(a)

(b)

(c)

Figure 2: **2a** Collection of 100 realizations of epidemic outbreak on 5000 nodes, $T = 0.4$. **2b** Infection peak for networks on different sizes, the fraction of infected people grow with graph size. **2c** The total fraction of infected (and recovered) people seems to grow linearly for this range of graph sizes.)

The first plot shows all 100 realizations of the epidemic simulation with 2000 nodes $\alpha = 2.2$ and $T = 0.4$. The second shows the average infection per time for different graph sizes. Note the larger the graph the larger the fraction of infections as we are simulating situations where exists giant components.

Finally I plotted the total number of infected people (after the cease of epidemic) as a function of the graph size to find that we have a linear relation for the fraction of infected people to the size. Which means a quadratic relation to the number of infections to the size of the graph. In any case, the duration of the epidemic seems to be similar for all sizes tested, we get larger fraction of people infected but the pandemic dies similarly. We may get other behaviors if I could tested using larger networks ($\sim 10^4$) but this was outside the scope of this report.

This particular epidemic model is similar to the BFS in the sense that we start “spreading" the disease (or the discovered nodes in the case of the BFS) and start changing the color of the nodes as we progress. The “colors" change from susceptible (white) to infected (gray) to recovered (black).