CS 4/5740 Networks, Crowds, and Markets
Project #3: Misinformation and Epidemiology

Submission requirements:

- A .zip file containing your source code. You may use any language you would like.
- A PDF (**submitted separately** to the Canvas assignment) containing each item below that is listed as a **Deliverable.** For each item contained in your PDF, clearly mark which deliverable it is associated with. Plots should be clearly labeled and have descriptive captions.

5740 Students:

- In Deliverables 4 and 5, you will implement an epidemic simulation and try to understand which nodes are the most important places to implement mitigation techniques. You'll learn a bit about how an adjacency matrix's eigenvalues impact the relative importance of each node.

Tips:

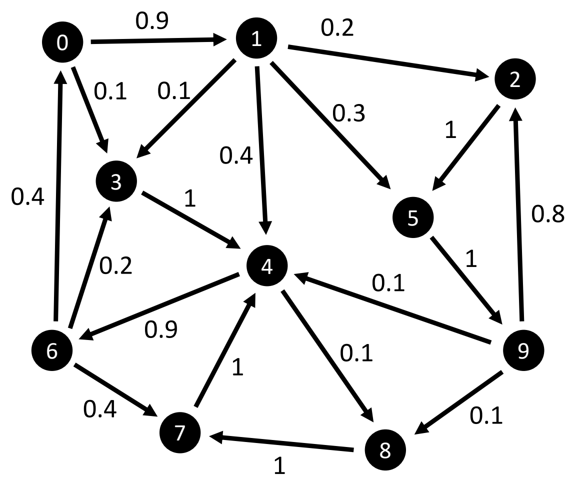
- The technical purpose of this project is to practice using these linear-algebra-based opinion/epidemic models and draw broader conclusions.
- The extra-technical purpose of this project is to start you on a path to curiosity: as you run these simulations and consider these concepts, you owe it to yourself to take the opportunity to *describe what is happening in your own words.* Any time you come up with a result of some kind, try to explain what the result means in non-technical language.

Misinformation in the Friedkin-Johnsen model. In class, we discussed the Friedkin-Johnsen model of opinion dynamics, where opinions spread on a network but people always remember their initial opinions. To refresh your memory, the model works like this: You're given a society with n individuals connected on a social influence network described by an $n \times n$ row-stochastic, directed, weighted adjacency matrix A. An edge (i, j) (arrow pointing from i to j with weight a_{ij} means that node i gives a fraction a_{ij} of its attention to node j. Each node i has a *susceptibility* $\lambda_i \in [0, 1]$, and we collect these susceptibilities into an $n \times n$ diagonal matrix $\Lambda = \mathrm{diag}(\lambda_0, \lambda_1, \ldots, \lambda_{n-1})$. Given an initial vector of opinions $x(0) = (x_0(0), x_1(0), \ldots, x_{n-1}(0))^T$, the opinion at time t is given recursively by

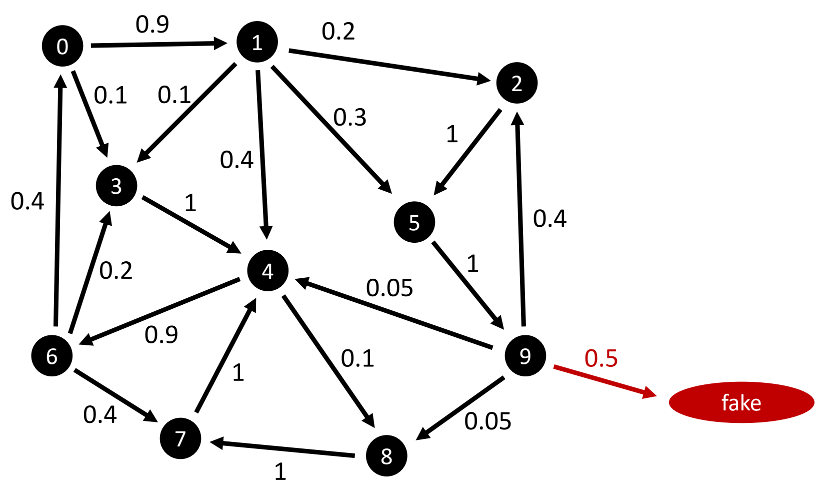
$$x(t) = \Lambda A x(t-1) + (I - \Lambda)x(0),$$

where I is the $n \times n$ identity matrix.

In this project, all questions consider the following weighted directed graph, with susceptibilities of $\lambda = (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.95)$:



A simple model of misinformation. Consider the following model of misinformation: let $x(0) = (0, 0, \ldots, 0)^T$ so that everybody starts out at 0. Then, choose a node to influence (let's say you're influencing node i), and connect i to a new "fake" node whose initial "opinion" is 1. To make this connect, cut the weights of each of i's out-edges in half, and give the new edge a weight of 0.5. So the math works, the new fake node should have a single self-loop with weight 1. See the following figure for a misinformation node attached to node 9:



Because everybody starts out at 0, you can measure the effect of your misinformation simply by adding up everybody's opinion; we'll call this sum the *propaganda value* of your misinformation scheme, and denote it by $P(t) := \sum_{i=0}^{n-1} x_i(t)$. An ineffective misinformation scheme would have $P(t)$ close to 0 for all time, but a very effective one would have $P(t)$ close to n.

Assignment:

1. Implement the Friedkin-Johnsen model on the given graph. As always, I will check for copy-pasted code from the internet so if you use someone else's code, you must cite them.

   Deliverable 1: Either the computer code for your implementation, or a complete description of the math you used to prove the other deliverables. CS 5740 students, this deliverable includes your epidemic code as well.

2. If you want to influence opinions over the short-term, which node should you influence?

   Deliverable 2: The graph has 10 nodes; create a table which shows $P(1)$ (the total opinion of society after 1 time step) for each node to be influenced. That is, compute $P(1)$ in the case that you influence node 0; then compute $P(1)$ in the case that you influence node 1, and so on. If you care most about $P(1)$ (society's opinions in the very short term), which node is best to influence? Which is worst? Write a few sentences explaining why you think this is the case.

3. If you want to influence opinions over the long-term, which node should you influence?

   Deliverable 3: The graph has 10 nodes; create a table which shows $P^\infty := \lim_{t\to\infty} P(t)$ (the opinion that society converges to after a long time) for each node to be influenced. That is, compute P^∞ in the case that you influence node 0; then compute P^∞ in the case that you influence node 1, and so on. If you care most about P^∞ (society's opinions in the very long term), which node is best to influence? Which is worst? Write a few sentences explaining why you think this is the case. If the answer is different from what you got in Deliverable 2, try to explain why.

4. **For the graduate section only**: Using the same graph as above, implement the epidemic model we discussed in class. For simplicity, assume that the population of each node is $N_i = 1$, so that $S_i(t)$, $I_i(t)$, and $R_i(t)$ denote the fraction of the population at node i that is *susceptible, infectious,* and *recovered* at time t. Write $S(t)$, $I(t)$, and $R(t)$ to denote the column vectors containing the susceptible, infectious, and recovered fractions for each node. If A denotes the graph's weighted row-substochastic adjacency matrix, the infection rate is $\beta \in [0,1]$, and the curing rate is $\gamma \in [0,1]$, then the epidemic evolves over time according to these dynamics:

$$\begin{aligned}
S(t+1) &= S(t) - \beta \ \mathrm{diag}(S(t))AI(t) \\
I(t+1) &= I(t) + \beta \ \mathrm{diag}(S(t))AI(t) - \gamma I(t) \\
R(t+1) &= R(t) + \gamma I(t).
\end{aligned}$$

   Now, suppose you're a public health authority and you can choose to implement mitigation efforts on one node. We'll model mitigation efforts as a 50% reduction of the contact frequencies of the mitigated node; this is equivalent to multiplying the corresponding row in A by 50% (note: the matrix won't be row-stochastic once you do this, but it will still be row-substochastic since the mitigated row will add to 0.5).

   Which node is the most important to mitigate?

Deliverable 4 (CS 5740 only):

Let $\beta = 0.8$ and $\gamma = 0.75$. Let $I_i(0) = 0.0001$, $S_i(0) = 0.9999$, and $R_i(0) = 0$ for every node (that is, every location starts out with 0.01% of its population infected). Simulate the epidemic with no mitigation by running the simulation long enough that the infection dies out. Then, measure the total number of people who were ever sickened (do this by summing the $R(t)$ vector over all nodes, for a large enough t).

Then, repeat this experiment 10 times; in the ith experiment, enforce mitigation efforts on node i. Report the results of each of the 11 experiments in a table or plot.

On which node are mitigation efforts the most effective? On which node are they the least effective?

Repeat the experiments with a much more virulent disease, with $\beta = 0.8$ and $\gamma = 0.3$ (people cure more slowly, so they stay infectious longer). Compare these results with the first set of experiments. Are the important nodes the same in both sets of experiments?

5. **For the graduate section only**: It turns out that the eigenvalues of A tell us a lot about whether a major outbreak will happen in this epidemic model. For reference, we call λ an *eigenvalue* of A if there is some vector x such that $Ax = \lambda x$. In general, an $n \times n$ matrix has n eigenvalues (though sometimes some of them are repeated). In graph theory, the largest eigenvalue is very important; its magnitude is called the *spectral radius* of A.

   Sometimes the spectral radius (the maximum magnitude of any eigenvalue of A) is denoted $\rho(A)$.

   Deliverable 5 (CS 5740 only):

   Using the results of the mitigation experiment you performed in Deliverable 4, plot the relationship between the spectral radius of the mitigated matrix and the total number of people sickened when that node is mitigated. Let A_i be the matrix with the ith node mitigated, and let R^i be the total number of people who eventually got sick when the ith node was mitigated.

   Your deliverable is a scatter plot with pairs $(\rho(A_i), R^i)$. That is, the horizontal axis should be the spectral radius $\rho(A_i)$, and the vertical axis should be the total number of people sickened R^i. If your experiments all worked correctly, the result should be nearly a straight line.

   Comment on the relationship that your plot shows. How might this relationship be used in an actual pandemic scenario?

   Code hints: If you're using Python, you might find the `numpy.linalg.eig` function useful to find the eigenvalues of a matrix. Also, note that since the matrices we're using in this project are not symmetric, many of the eigenvalues you find will be complex numbers so you'll need to use a function like `numpy.absolute` to compute the magnitude of each eigenvalue.