Math 2174 Extra Credit Project

Due by noon on Monday, April 8, by email

This (optional) extra credit project consists of 2 parts. Part 1 has 6 steps; Part 2 has 4 steps. Each step is worth 1 point, and your total score (a max of 10 points) will be added to your final grade as extra credit. Part 1 should be solved using Matlab (I suggest checking the intuition behind your solutions by hand as appropriate). Part 2 can be solved by hand. Your solutions must be clearly presented and well-explained to receive full credit. I have included a few suggested websites as resources on the last page. You are encouraged to email me or come to office hours if you have questions (if office hours conflict with your class or work schedule, let me know).

To complete Part 1, please fill in the Matlab files part1.m and part1_distance.m. For Part 2, scanned images of handwritten notes are fine; if you would like to learn LaTeX, send me an email and I can provide you with a template document for typing solutions. A complete project consists of 3 files:

1. part1.m
2. part1_distance.m
3. a pdf file titled part2_YourName.pdf providing your solution to Part 2

Please submit these files (or the subset of files pertaining to the exercises you choose to do) to me by email as a zipped folder with your name in the title by April 8 at noon.

1 Linear algebra applied to networks

Take-away: In this class, you learn how to perform various algebraic operations with matrices and compute their eigenvalues and eigenvectors. Part 1 illustrates a few ways these Math 2174 tools are being applied in the very active and fairly new field of network science.

A network is a collection of points (called *nodes*) and lines (called *edges*) that connect different pairs of points (see Fig. 1a). Examples of networks include the internet, Facebook friendship, Twitter followership, highways, flight paths between airports, and connections between neurons in the brain. In the Facebook setting, nodes can be used to denote different accounts; an edge between two nodes denotes friendship between those two individuals. While friendship is not directed (if α and β are Facebook friends, so are β and α), this is not the case on Twitter: α may follow β, but β might not follow α. We call Twitter followership a *directed network*; for directed networks, we add an arrow to each edge to denote the edge direction (see Fig. 1b–1c).

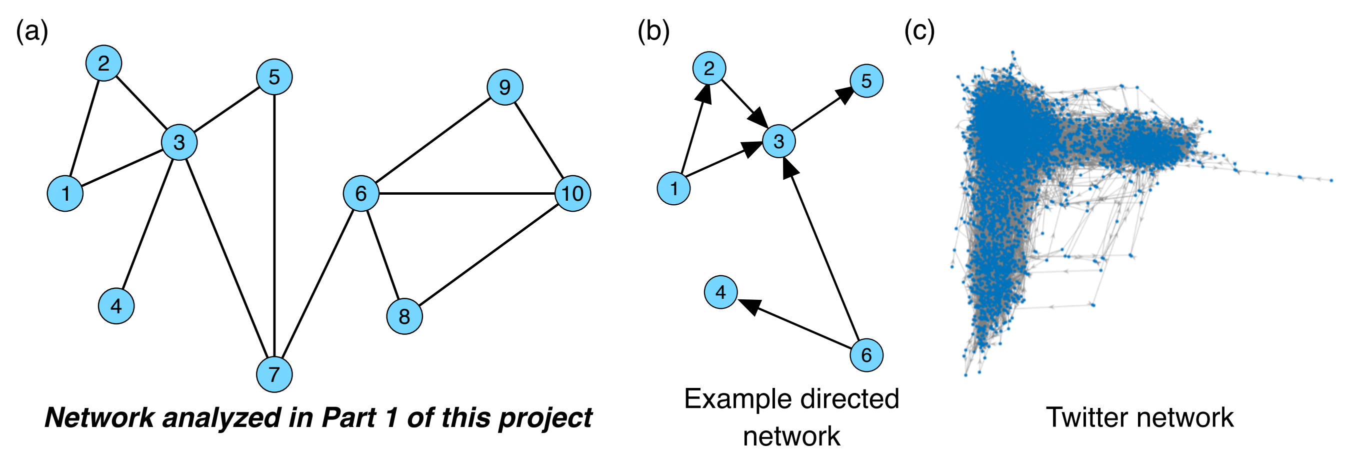

Fig. 1: (a) Network that you are asked to analyze in this project. (b) Example directed network. (c) As an example of a current research area, a real-world network obtained from Twitter data related to the NFL anthem protests (blue points are different Twitter accounts; grey lines denote followership).

The connections between nodes in a network (or graph) are described using matrices. In particular, the adjacency matrix $A = (a_{ij})$ of a graph with n nodes is an $n \times n$ matrix of $0's$ and $1's$ that represent connections between different nodes. We will assume in this project that nodes are not connected to themselves, so note that $a_{ii} = 0$. For a graph that is not directed (e.g., Facebook), the adjacency matrix A is given by:

$$
\begin{aligned}
a_{ij} &= 1 \qquad \text{if node } i \text{ and node } j \text{ are connected by an edge (e.g., Facebook friends)} \\
a_{ij} &= 0 \qquad \text{if node } i \text{ and node } j \text{ are not connected by an edge (e.g., not Facebook friends).}
\end{aligned}
$$

For a directed graph (e.g., Twitter followership), the adjacency matrix A is given by

$$
\begin{aligned}
a_{ij} &= 1 \qquad \text{if there is a directed edge from } i \text{ to } j \text{ (e.g., node } i \text{ follows node } j) \\
a_{ij} &= 0 \qquad \text{if } i \text{ and } j \text{ are not connected.}
\end{aligned}
$$

In Part 1, you are asked to compute some properties of the network in Fig. 1a using Matlab. Please fill in the script "part1.m" to completes steps 1.1-1.5 below and the function "part1_distance.m" to complete step 1.6:

(1.1) Create the 10×10 adjacency matrix A for the network in Fig. 1a. (Carefully check your solution, as the rest of Part 1 is based on this matrix).

(1.2) The degree of a node is the number of edges emanating from that node (on Facebook, this would be the number of friends a person has). You can find the degree of each node in Fig. 1a by computing $\mathbf{d} = \mathbf{A}_1 + \mathbf{A}_2 + ... + \mathbf{A}_{10}$, where $A = [\mathbf{A}_1, \mathbf{A}_2, ..., \mathbf{A}_{10}]$ is the adjacency matrix. Then d_i is the degree of node i. By computing $\mathbf{d}$, find the maximum and average degree of the graph in Fig. 1a; please label these values maxDegree and avgDegree in your code.

(1.3) Verify that A is symmetric by computing A^T and showing that $A - A^T = 0$.

(1.4) In fact, all networks that are not directed have symmetric adjacency matrices. Moreover, because A is real, we know (by Theorem 17 on pg. 319 of the linear algebra textbook) that all the eigenvalues of A are real. To verify this, use Matlab to find the eigenvalues and eigenvectors of A; please label the eigenvalues lambda1, lambda2, lambda3,... and the eigenvectors u1, u2, u3... in your code. Notice that the eigenvalues of A are all less than or equal to the max degree of the graph; this is always the case.

(1.5) From an applied perspective, it is important to understand what nodes are most important or central in a network; for example, if you are a civil engineer responsible for prioritizing what road intersections should be cleared of snow first in the winter, what intersections are most important for traffic flow? Or, if you are in charge of promoting a product on Facebook by selecting accounts to be influencers, which accounts will have the widest reach? There are many measures of centrality, but eigenvector centrality is one of the main ones. (Google relies on a slight variant of eigenvector centrality to specify the order you see webpage results after a search). Eigenvector centrality is computed by first finding the maximum eigenvalue $\lambda_{\max}$ of A and its associated eigenvector $\mathbf{x}$. Using the previous step, find these objects and label them lambdaMax and x in your code. We know eigenvectors are only defined up to a common factor (e.g., if $\mathbf{x}$ is an eigenvector of A, so is $a\mathbf{x}$ for $a \neq 0$), so normalize $\mathbf{x}$ by dividing by the sum over all its elements; in particular, compute $\mathbf{x} = \frac{1}{\sum_{j=1}^{10} x_i}\mathbf{x}$ in your code. The ith component of this normalized $\mathbf{x}$ is the eigenvector centrality of the ith node in the network. Find the eigenvector centrality of each node in the network in Fig. 1a. As a comment in your code, write down which node is the most central (has the highest eigenvector centrality) in the network and explain why this makes sense.

(1.6) In many applications, it is useful to know how far apart different nodes in a network are (for example, some research has suggested that all people in the world are separated by at most 6 social connections), and linear algebra gives us a way to do this. For an adjacency matrix A, it turns out that the (i, j) element of the matrix A^n (the product of A with itself n times) is the number of walks of length n from node i to node j. A walk between node i and node j is a way of getting from one node to the other by passing along edges of the graph; for example, in Fig. 1a, the shortest way of getting from node 4 to node

5 is to first walk along the edge between nodes 3 and 4 and then walk along the edge between nodes 3 and 5 (because we travelled along two edges in this example, we say this walk has length 2). Finding the smallest integer n such that the (i, j) element of A^n is positive amounts to finding the distance n between nodes i and j. Write a function titled "part1_distance.m" in Matlab that takes the indices of two nodes i and j as its inputs and returns the shortest distance n between those nodes for the network in Fig. 1a.

2 Linear algebra & differential equations applied to contagion spread

Take-away: In this class, you learn how to determine if equilibrium points are stable or unstable for linear differential equations of the form $\mathbf{x}' = A\mathbf{x}$, but differential equations often have nonlinear terms in applications. Part 2 of this project illustrates how to extend ideas based in Math 2174 to nonlinear differential equations.

The Susceptible-Infected-Recovered (SIR) model is a widely-studied system of differential equations that has been used to explore the spread of biological diseases, the evolution of opinions, the spread of memes, and the adaption of techniques by individuals in a community. In this model (first proposed by Kermack and McKendrick in 1927), individuals are described as either *susceptible* (e.g., having the potential to be infected in the future), *infected* (e.g., currently infected with the contagion), or *recovered* (e.g., immune to the contagion). Infected individuals can infect susceptible people, and infected individuals can randomly recover; once recovered, individuals remain recovered. We will let $S(t), I(t)$, and $R(t)$ denote the number of individuals in a population that are susceptible, infected, and recovered at time t; the SIR model is then given by 3 differential equations:

$$\frac{dS}{dt} = \underbrace{-\frac{\beta}{N}IS}_{\text{loss of susceptible individuals due to infection}}$$

$$\frac{dI}{dt} = \underbrace{\frac{\beta}{N}IS}_{\text{growth of } I(t) \text{ due to susceptibles becoming infected}} - \underbrace{\gamma I}_{\text{loss of infected individuals due to recovery}}$$

$$\frac{dR}{dt} = \underbrace{\gamma I}_{\text{growth of } R(t) \text{ due to infected individuals recovering}}$$

where β and γ are positive numbers (these parameters would depend on the specific disease considered and are usually fit to data), and N is the number of people in your population. Notice that $S(t) + I(t) + R(t) = N$; using this, we can reduce the system above to 2 differential equations:

$$\frac{dS}{dt} = -\frac{\beta}{N}IS \tag{1}$$

$$\frac{dI}{dt} = \frac{\beta}{N}IS - \gamma I. \tag{2}$$

Once $S(t)$ and $I(t)$ are known, $R(t)$ can be computed as $R(t) = N - S(t) - I(t)$. In Part 2, you are asked to study the dynamics and stability properties of differential equations (1)–(2); please submit your solution to Part 2 as a pdf document.

(2.1) Find the equilibrium (S^*, I^*, R^*) of Equations (1)–(2) such that $R(0) = 0$. Verify that

$$0 = -\frac{\beta}{N}I^*S^* \tag{3}$$

$$0 = \frac{\beta}{N}I^*S^* - \gamma I^*. \tag{4}$$

Explain why the equilibrium you found makes sense from an intuitive perspective.

(2.2) We would like to understand whether the equilibrium (S^*, I^*, R^*) found above is stable or unstable. However, while we know how to determine if equilibrium points are stable for systems of the form $\mathbf{x}' = A\mathbf{x}$

using Math 2174 tools[1], this system of differential equations is not linear because of the IS terms. To address this, we can perform what is called *linear stability analysis*. Linear stability analysis is a method for getting a nonlinear system of differential equations like Equations (1)–(2) into a linear form that we can study. A linear stability analysis of (S^*, I^*, R^*) tells us, if we perturb slightly away from this equilibrium, whether the perturbation will grow or decay. If the perturbation decays, the equilibrium is stable; otherwise it is unstable. To perform a linear stability analysis, let

$$\begin{aligned} S(t) &= S^* + \varepsilon \tilde{S}(t) \\ I(t) &= I^* + \varepsilon \tilde{I}(t). \end{aligned}$$

Plug these expressions into differential equations (1)–(2), simplify using (3)–(4), and solve for $\frac{d\tilde{S}}{dt}$ and $\frac{d\tilde{I}}{dt}$ on the lefthand side. Hint: notice, for example, that $\frac{d}{dt}\left(S^* + \varepsilon\tilde{S}\right) = \varepsilon\frac{d\tilde{S}}{dt}$ since S^* is constant.

(2.3) From the previous step, you should now have two differential equations of the general form:

$$\begin{aligned} \frac{d\tilde{S}}{dt} &= f(S^*, I^*, \tilde{S}, \tilde{I}) + \varepsilon g(\tilde{S}, \tilde{I}) \\ \frac{d\tilde{I}}{dt} &= h(S^*, I^*, \tilde{S}, \tilde{I}) + \varepsilon k(\tilde{S}, \tilde{I}), \end{aligned}$$

where $f, h, h,$ and k are functions. These differential equations describe the growth of the perturbation from the equilibrium. Because we are interested in small perturbations, we assume ε is small. In fact, we will assume that ε is so small that we can set $\varepsilon = 0$ and instead study a simplified problem of the form:

$$\frac{d\tilde{S}}{dt} = f(S^*, I^*, \tilde{S}, \tilde{I}) \tag{5}$$

$$\frac{d\tilde{I}}{dt} = h(S^*, I^*, \tilde{S}, \tilde{I}). \tag{6}$$

Write down these differential equations explicitly (e.g., specify what the righthand sides are) and plug in the values of S^* and I^* you found in the first part of the problem to arrive at a system of 2 linear differential equations for $\tilde{S}$ and $\tilde{I}$. In particular, find the 2×2 matrix A so that your system has the form:

$$\begin{bmatrix} \frac{d\tilde{S}}{dt} \\ \frac{d\tilde{I}}{dt} \end{bmatrix} = A \begin{bmatrix} \tilde{S} \\ \tilde{I} \end{bmatrix}.$$

(Note: your A may depend on the population size N and the parameters β and γ, which we assume are both positive numbers).

(2.4) Let

$$\mathbf{x} = \begin{bmatrix} \tilde{S} \\ \tilde{I} \end{bmatrix}.$$

Notice that we now have a linear system of the form $\mathbf{x}' = A\mathbf{x}$, which we can analyze using tools from Ch. 7. First verify that $\mathbf{x} = \mathbf{0}$ is an equilibrium of this system. By finding the eigenvalues of A, determine if this equilibrium given by

$$\mathbf{x} = \begin{bmatrix} \tilde{S} \\ \tilde{I} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

is stable or not. Discuss how your findings about the stability of this equilibrium relate back to the original problem (Equations (1)–(2)). Hint: The signs of the eigenvalues you found tell you if small nonzero perturbations $\tilde{S}(t)$ and $\tilde{I}(t)$ will grow or decay back to zero. What does this mean in terms of when you expect individuals in a population to become susceptible or infected to a contagion? If only one person in your population is infected (so $I(0) = 1$, $S(0) = N - 1$, and $R(0) = 0$), do you expect the disease to spread or not? Does this depend on the parameters α and β?

[1] See Ch. 7 of the second textbook for this class. Note that we will start covering Ch. 7 on March 8 and continue after spring break

3 Suggested resources

- Nykamp DQ, "An introduction to networks." From Math Insight. `http://mathinsight.org/network_introduction`
- Mark Newman, *Networks: An Introduction*. DOI: 10.1093/acprof:oso/9780199206650.001.0001. This textbook should be available for free online through the OSU library – let me know if you have any trouble obtaining it.
- David Smith and Lang Moore. "The SIR Model for Spread of Disease - The Differential Equation Model". Mathematics Association of America. `https://www.maa.org/press/periodicals/loci/joma/the-sir-model-for-spread-of-disease-the-differential-equation-model`.