Overall Instructions: Due by Monday, Dec. 8th, 11:00pm EST. Please submit a .pdf of your answers via Canvas, including a url to your .ipynb notebook in a public GitHub repository. Use the provided LaTeX template for your submission. Please name your Canvas submission as Lastname.pdf, (*e.g.* Trujillo.pdf). If you need help with LaTeX or GitHub, please visit during office hours or post questions to the Canvas. Include your code in a (well-documented, commented, clear) `assignment03.ipynb` file. Scores for the code-based questions below will be based on your written responses *and* code.

Problem 1 (40 points)

In this question we will work with the Lotka-Volterra *predator-prey* model, which we have discussed some in class. In this model, we represent the populations of a prey species (such as little bunny rabbits) and a predator species (such as foxes) and how they interact with one another. We will begin with a mean-field compartmental model, where we assume that all animals mingle and interact with all others, without considering environmental factors. This simple version of the model can be represented as follows:

$$\underset{\text{Change in prey population over time}}{\frac{dx}{dt}} = \overset{\text{Prey reproduction rate}}{\alpha}\, x - \underset{\text{Predation rate}}{\beta}\, xy$$

$$\underset{\text{Change in predator population over time}}{\frac{dy}{dt}} = -\overset{\text{Predator death rate}}{\gamma}\, y + \underset{\text{Predator growth rate given food}}{\delta}\, xy$$

Prey are assumed to have an unlimited food supply (plenty of grass for the bunny rabbits) and only die as a result of predation. Predators are assumed to only feed on prey and have no other factors influencing their reproduction.

(a) Closed form (20 points).

First, implement the Lotka-Volterra model in Python as a continuous deterministic model. Using the parameters $\{\alpha, \beta, \gamma, \delta\} = \{0.1, 0.002, 0.2, 0.0025\}$, plot the populations of both species over time from $t = 0$ to $t = 100$, as we did in lecture for the SIR model. You may use starting populations of 80 prey and 20 predators.

NOTE: You should evaluate at very small increments of dt, perhaps $dt = 0.0001$, to maintain numeric stability.

(b) Sensitivity Analysis (10 points).

Describe the possible stable states of the Lotka-Volterra model. For example, one outcome is "all the bunnies are eaten and all foxes starve, resulting in x=0, y=0." What are the other possible outcomes? Find parameters that yield each outcome, make an accompanying population plot, and describe the result.

(c) Discussion (10 points).

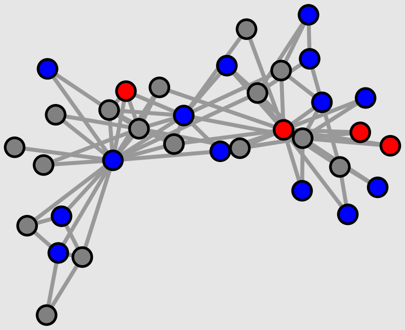
The Lotke-Volterra model as outlined above is very similar to the SIS epidemic model. However, we can observe population *cycles* in the predator prey model, and we cannot in SIS. Why do we observe cycles in one and not the other? Do we see cycles in SIR or SIRS models?

Solution:

Your solution here...

Problem 2 (40 points)

We will now adapt Lotke-Volterra to a network setting. We will take a graph G as an argument, and assign ratio x nodes to start as prey, ratio y nodes to start as predators, and all other nodes to begin as "empty." For example, here is our favorite Karate Club graph with blue prey (0.4), red predators (0.1), and gray empty vertices:



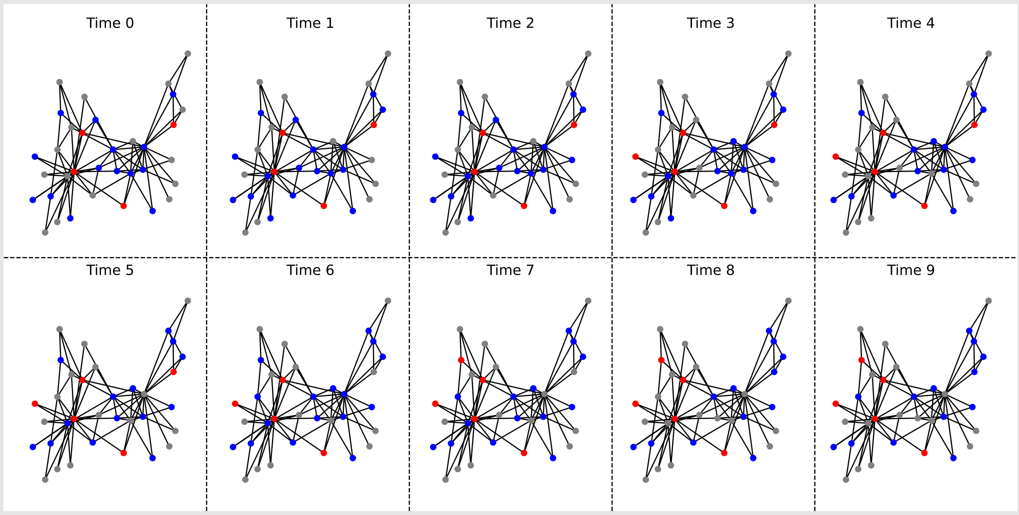
At each timestep, each prey animal may, with probability α, "reproduce" by turning a randomly-chosen empty neighboring vertex into a prey vertex. Each prey animal may be eaten by each neighboring predator (and replaced with an empty vertex) with probability β. If the predator eats a prey animal there is a δ chance of that vertex becoming another predator vertex instead of empty. Finally, each predator may die and become an empty cell with probability γ.

(a) Implement the simulation (20 points).

Implement the above simulation in Python, creating a function with the signature:

```
def graphLotkeVolterra(G, x, y, alpha, beta, gamma, delta, T)
```

This simulation runs for T time steps, and returns a list of the simulation state at each time step. You may return a list of networkx graphs, or a list of tuples representing predator and prey populations. Here is one example of species progression over time in the Karate club graph:



If you need to adjust starting populations or your $\alpha, \beta, \gamma, \delta$ parameters to produce interesting results, do so. For example, if you find that all the predators quickly die without reproducing, you may need to increase β or γ or decrease γ a little.

You do not need to produce a plot like the one above, but it may help your debugging.

(b) Influence of Network Structure on Population (10 points).

Create the following graphs: 2d-grid, Erdös-Rényi, Barabasi, and Watts-Strogatz small-world using the following parameters:

```
G1 = nx.grid_2d_graph(100, 100)
G2 = nx.gnp_random_graph(1000, 0.05)
G3 = nx.barabasi_albert_graph(1000, 2)
G4 = nx.watts_strogatz_graph(1000, 2, 0.05)
```

Run your simulation on each for at least 100 time steps. For each simulation, produce a plot of predator and prey populations over time.

(c) Discussion (10 points).

What effect does network topology have on predator-prey dynamics? Are some topologies more advantageous for one species or the other, are some more favorable to stable equilibrium? Why?

Solution:

Your solution here...

Problem 5 OPTIONAL (5 bonus points)
We are nearing the end of the course. What topics do you wish we had covered, what topics do you wish we had gone into more detail on, and would you have preferred we spent less time on any topic? This will help calibrate future versions of this course, but may also influence what we cover in CNET 5052.

Solution:

Your (optional) solution here...