Experimenting with Spreading Processes

Several libraries exist to do diffusion on networks. I propose to use `ndlib`, that can be installed using `pip`.
Documentation and examples: https://ndlib.readthedocs.io/en/latest/.

1. Spreading on a real network
   (a) Initialize a SIR model on the airport network using `model = ep.SIRModel(g)`
   (b) Create and initialize a custom configuration for your model, following https://ndlib.readthedocs.io/en/latest/tutorial.html#configure-the-simulation. Start with 1% of nodes infected, and a same value for β and γ.
   (c) Still following the tutorial, plot the evolution of the fraction of nodes that are Infected and Removed.
   (d) Run the simulation a few times and observe how much the results differ from one run to the next.

   To properly study the results of simulations, you should analyze results averaged over many simulations. If you see how to do such tests with python, you can extract the results of a simulation using for instance `trends[0]["trends"]["node_count"][1]`, and create averaged plots.

2. Spreading compared with synthetic networks
   (a) Create an ER random graph with the same number of nodes and edges than the airport network. Run the same SIR as before and compare the results.
   (b) Let's say that we identified a virus with $\gamma = 0.1$, and that we can use policies to reduce β. What is the value of β above which there will be an outbreak, in theory, on this random network? Test values slightly above and below this value and check the results
   (c) Test the same model with the same parameters on the original network. What do you observe?
   (d) Do the same with a configuration model based on the airport dataset.

3. Dependence on initial conditions
   (a) Using `cfg.add_model_initial_configuration("Infected", infected_nodes)`, you can choose which nodes are infected in the beginning. Start by infected the 5 nodes of highest degrees, then the 5 nodes of lowest degree. Observe the difference.
   (b) Start by infecting 5 nodes in the same country, and then 5 random nodes. Observe the differences.

4. Optimal node removal
   (a) Let's assume that we can vaccinate some limited fraction of nodes. In practice, vaccinated nodes are removed from the network before running the SIR model. Starting with parameters leading to a large diffusion of the virus, starting from 5 nodes in the same country, what fraction of nodes to you need to vaccinate so that the virus infect less than 10% of nodes?
   (b) Same question but removing nodes of largest degree
   (c) Same question but removing nodes of largest betweenness

5. Going further
   (a) Choose 1 or a few nodes in the same country, and some parameters that will ensure a large spread of the virus, but not higher than 80%. Run many simulations, and memorize the infected nodes.

(b) Create plots in which the color of nodes correspond to their probability to be infected after 5 steps, 50 steps, 300 steps.

(c) Can you find good predictors of the infection or not of a node? You could check for instance their centralities, distance to originally infected nodes, belonging to the same community or not, geographical distance, etc.