Tutorials

Table of Contents

Setup

1. [OPTIONAL] Install Anaconda. Download from anaconda.com. Try watch this tutorial if you are getting issues in installation.
   - [OPTIONAL] After successful installation of anaconda, create a new anaconda environment. In a terminal/command prompt enter
     ```
     conda create --name pansim python=3.8
     ```
   - `conda activate pansim`
2.
   You can replace pansim with a name of your choice. **Continue using this terminal/command prompt window for the following steps.**
   - Clone the repository this is done by entering the following in a terminal/command prompt window.
     ```
     git clone -b tst
     https://github.com/alfred100p/PandemicSimulator
     ```
3.
   - Continue in the same terminal/command prompt. Change current directory and run setup.py
     ```
     cd PandemicSimulator
     ```
   - `python -m pip install -e .`
4.

Congratulations you finished the setup. Time to start the first Tutorial.

Tutorial 1

Welcome to the first tutorial.

After this tutorial, you should get an idea of what this repository is about. You will manually control the stage of response to a simulated pandemic. After that by studying the observations try to create a response policy using if-else statements.

1. **Read the Original Paper.**
   This will help you understand the background of this problem and the environment.
2. **Run scripts/tutorials/7_run_pandemic_gym_env.py**.
   Do this to understand how to use the environment.
3. **Run scripts/tutorials/manual_control.py**
   This allows you to manually set the stage at each point. Use this to understand which stage to apply at a given moment by choosing stage based on the observation.

4. **Run scripts/tutorials/example_policy_if_else.py**
   This is an example policy implemented using an if else statement based on the observation.
5. **Read Explaining the Observation and Action**
   This will help you understand the observation and make a custom policy for the next part of the tutorial.

Explaining the Observation and Action

Observation Table

Observation	Explanation	How to Access
Critical Population (Testing)	Number of people in Critical condition according to current testing policy	obs.global_testing_summary[...,0]
Dead Population (Testing)	Number of people in dead according to current testing policy	obs.global_testing_summary[...,1]
Infected Population (Testing)	Number of people in Infected condition according to current testing policy	obs.global_testing_summary[...,2]
None Population (Testing)	Number of people in not in any other condition according to current testing policy	obs.global_testing_summary[...,3]
Recovered Population (Testing)	Number of people in Recovered condition according to current testing policy	obs.global_testing_summary[...,4]
Critical Population (Actual)	Actual number of people in Critical condition	obs.global_infection_summary[...,0]
Dead Population (Actual)	Actual number of people in dead	obs.global_infection_summary[...,1]

Infected Population (Actual)	Actual number of people in Infected condition	obs.global_infection_summary[...,2]
None Population (Actual)	Actual number of people in not in any other condition	obs.global_infection_summary[...,3]
Recovered Population (Actual)	Actual number of people in Recovered condition	obs.global_infection_summary[...,4]
Current Stage	Stage of Response at Current timestep	obs.stage[...,0]
Infection Flag	Whether Number of Infected People according to current testing policy exceeds threshold specified according to current simulator configuration (10 by default)	obs.infection_above_threshold[...,0]
Current Day	Current Day of simulation	obs.time_day[...,0]
Current Unlocked Non Essential Business Locations	List of ids of businesses that are Unlocked (By default additional businesses are not used in the simulator, so this is unused)	obs.unlocked_non_essential_business_locations[...,0]

Action Table

Stages	Stay home if sick, Practice good hygiene	Wear facial coverings	Social distancing	Avoid gathering size (Risk: number)	Locked locations
Stage 0	False	False	None	None	None
Stage 1	True	False	None	Low: 50, High: 25	None
Stage 2	True	True	0.3	Low: 25, High: 10	School, Hair Salon
Stage 3	True	True	0.5	Low: 0, High: 0	School, Hair Salon
Stage 4	True	True	0.7	Low: 0, High: 0	School, Hair Salon, Office, Retail Store

6. **Run scripts/tutorials/custom_policy_if_else.py**
   This is another example of a policy with an if else statement. Try replace the statements and create your own policy.
7. **Run scripts/tutorials/custom_policy_test.py**
   Take your policy from scripts/tutorials/custom_policy_if_else.py and run this to evaluate your policy over multiple episodes. This will be evaluated in class with a random seed.

Tutorial 2

In this tutorial we will work on observations. An observation is what an agent (your program) can view of the state. Lets consider our current problem of Pandemic Simulation, here we are trying to make an RL agent which will provide us an optimal response to a pandemic. In real life, the authorities who provide such response do not have complete knowledge of the state of the environment which would be an accurate number of how many people have the disease. This data is not accurately obtainable because of testing limitations so the data available is the testing data. We can also augment the data we get from the observation and add it to the observation to "point out" important information. For example, if you consider chess we can make the observation include the position of each piece but you could also include a flag for whether you are in check to point out your king is in danger. This allows us to explicitly convey important information. In our current problem, we convey using flags that the current population has more critical cases than the threshold using a flag. In this tutorial we will create another such flag and add it to the observation.

PLEASE NOTE LINE NUMBER VALUES MAY CHANGE WHEN CODING LOOK IN THE AREA OF THE SPECIFIED LINE NUMBER

Steps

1. **Read Explaining the Observation and Action**
2. **Replace contents of following files with files from replacements folder**
   ○ python/pandemic_simulator/environment/simulator_opts.py
   ○ python/pandemic_simulator/environment/pandemic_sim.py
   ○ python/pandemic_simulator/environment/pandemic_env.py
   ○ python/pandemic_simulator/environment/interfaces/sim_state.py
   ○ python/pandemic_simulator/environment/interfaces/pandemic_observation.py
   ○ python/pandemic_simulator/viz/mplib_viz.py
3. **Add Critical Flag**
   To modify the observation we can change the code at two levels. We can either change the structure of the simulator state or we can modify the final observation. For this tutorial we will modify the simulator state. The code for simulator state is in python/pandemic_simulator/environment/interfaces/sim_state.py
   For demonstration, we will be creating a new flag variable for the environment. This flag will check if the no. of patients who are critical exceeds a threshold. The name of the variable should be critical_above_threhsold.
   ○ **Modify Simulator State Template to add new flag.**
      Open python/pandemic_simulator/environment/interfaces/sim_state.py
      i. (Line 52) Add code for the new flag in the specified area. This change will allow the sim_state to hold the new flag.
   ○ **Modify Simulator Config to add threshold, as it is a characteristic of the simulator**
      Open python/pandemic_simulator/environment/simulator_opts.py
      i. (Line 49) Add code for a threshold to use for the flag and assign a default value of 10. The threshold is not a variable in the PandemicSimulator class as it is constant throughout the simulation and is a characteristic of the simulator.

- ○ **Write Code to incorporate the flag in the simulator**
  Open python/pandemic_simulator/environment/pandemic_sim.py
    - i. (Line 45,63,93,121) Here we will have to modify the init function. This is to enable users to add a value for our new flag when initializing the Pandemic Simulator State.

    - ii. (Line 168) Change from_config function

    - iii. (Line 325) Next we will have to modify the step function in pandemic_sim.py. This is to add code to update the simulator state after each timestep(hour).

    - iv. (Line 416) Then set a default value for flag in the reset function.
- ○ **Modify the observation for flag**

  Open python/pandemic_simulator/environment/interfaces/pandemic_observation.py

    - i. (Line 25) Add code to initialize flag in observation.
    - ii. (Line 44) Add code to attribute for flag in function.
    - iii. (Line 78) Add code to update flag in observation.
4. Add beta Using Testing data into the observation, considering a time period of 1 day.

   - ○ Use the formula:

     beta(t)=[None(t-1)-None(t)]/[None(t)*Infected(t)].

   - ○ The variable should be accessible via obs.beta
   - ○ Test if it works using test_beta.py

Tutorial 3