Epidemic Control with Reinforcement Learning Project Report

AccountAdmin VectorInstitute

1 Introduction

Modelling the spread of an epidemic in a population has been a long-standing problem. Numerous modelling methods emerged from this field of research, ranging from simple analytical models to agent-based modelling (de Menezes et al. [2004]). Among them, the most extensively-studied is a category named compartmental models (Nowzari et al. [2016], Eames et al. [2012], Klepac et al. [2018], Gog et al. [2014]). All compartmental models trace back to a fundamental "susceptible and infected" ideology, or SI. However, SI systems are limited in their ability to model recovery in an epidemic, where the recovered is no longer susceptible or infected. Therefore, the most popular building block for compartmental models takes into account the portion of the population that is "recovered" from the disease, and the system is named SIR for the same reason. Below we list the SIR model, represented as ordinary differential equations (ODEs). We use s to denote the portion of susceptible population, i to denote the portion of infected population, and r to denote the portion of recovered population over time. α and β are tunable parameters of the model, representing respectively the recovery and spread rate of infected individuals. Under the overarching assumption that only susceptible individuals can be infected (i.e. the only two state transitions are from susceptible to infected, and infected to recovered), these equations intuitively tells us that we assume a fixed rate of change within the infected population, and we assume no deaths occur from infected individuals (equations 1). See (Brauer et al. [2012]) for a more detailed explanation.

$$\begin{aligned} \frac{ds}{dt} &= -\beta si \\ \frac{di}{dt} &= \beta si - \alpha i \\ \frac{dr}{dt} &= \alpha i \end{aligned} \tag{1}$$

In recent years, reinforcement learning (RL) has been a large part of artificial intelligence research. The algorithmic breakthrough came in 1992 with REINFORCE (Williams [1992]), and more excitement came when RL was able to master the game of Go and beat the world champion Lee Sedol by 4-1 (Wang et al. [2016]). RL has also demonstrated its versatility when playing seven different Atari Games at a superhuman level (Mnih et al. [2013]). Most recently, deep reinforcement learning has enjoyed much attention in robotics, video segmentation, and beyond (Wang et al. [2020], Libin et al. [2020]). In section 2 we dive into detail on numerous reasons to use RL for epidemic control. Most important of all, RL does not impose any constraints on which model we use. In the long run, that allows us to apply our solution to population networks in a data-driven way rather than have a predefined model such as SIR.

Our agent interacts with the environment formulated as a Markov Decision Process (MDP). An MDP is a tuple ($\mathcal{S}$, $\mathcal{A}$, p, r) represented by a state space $\mathcal{S}$, an action space $\mathcal{A}$, a state transition p, and a reward function r. At each timestep t, the agent receives reward $r_t = r(s_t, a_t, s_{t+1})$ and the overall goal of the agent is to maximize the expected return, also named discounted sum of rewards

Preprint. Under review.

$R = \sum_{t=0}^{T-1} \gamma^t r_t$, where $\gamma \in [0,1]$ is a discount factor.

2 Background

There have been a wide variety of efforts to analyze the SIR model. While the SI model consists of only one first-order ODE and is thus analytically solvable, the SIR model in general does not have an analytical solution, and efforts to do so stop at parameterized solutions (Harko et al. [2014]). Nevertheless, the SIR model consists of only two first-order ODE equations, and therefore the most natural way to solve SIR is to use an ODE-solver.

In order to achieve epidemic control, however, one must not be limited to a simple ODE-solver, since that would imply a traversal of all possible parameters for control. As such, optimization-based techniques under the category "optimal control" have also been employed to solve an SIR model with control parameters. The shortcomings of optimal control is also very clear—it does not allow for robustness in the sense that the solver must re-run every time a change occurs in the environment dynamics. It also get prohibitively time-consuming when we run the solver on an entire population.

Therefore, we choose to solve the SIR model using reinforcement learning. Training an agent to learn the model dynamics and to come up with optimal solutions allows for faster problem-solving time than optimal control, since performing inference from the agent can be very fast.

3 Problem formulation

In order to solve ODEs with RL, we first rephrase an epidemic control problem with the SIR model in a discretized MDP formulation. In our definition, the discrete finite state space $\mathcal{S}$ represents the population dynamics, which can be categorized into "susceptible, infected, recovered", sticking to the SIR model. The action space $\mathcal{A}$ represents the control measures the agent will take, symbolizing controls taken in real life to prevent the spread of the disease, such as quarantine. Since we parameterize the agent with a simple neural network, it is more natural to describe the actions using continuous distributions. In our case, we have chosen to let our neural network output parameters of the beta distribution, which allows us to control the actions in the interval [0, 1], eliminating the unlikely event of having negative control actions (Chou et al. [2017]). In reality, one could also choose to have simple discrete actions representing quarantine actions being on and off. Our learned policy, represented by p, maps the state input s to an action output a. Finally, r represents undiscounted rewards along sampled trajectories. The continuous and discretized formulations of the SIR model are shown below.

$$\begin{aligned} \frac{ds}{dt} &= \frac{-\beta}{(1+v)} si - us \\ \frac{di}{dt} &= \frac{\beta}{(1+v)} si - \alpha i - ri \\ \frac{dn}{dt} &= -(1-f)\alpha i \\ r &= n - s - i \end{aligned} \tag{2}$$

$$\begin{aligned} s_{t+1} &= s_t + \int_t^{t+1} \left(\frac{-\beta}{(1+v)} si - us \right) dt \\ i_{t+1} &= i_t + \int_t^{t+1} \left(\frac{\beta}{(1+v)} si - \alpha i - ri \right) dt \\ n_{t+1} &= n_t + \int_t^{t+1} (-(1-f)\alpha i) dt \end{aligned} \tag{3}$$

In our problem formulation, we define $(r, u, v) \in \mathcal{A}$ as our control actions. Realistically, one could think of r as quarantine on the infected, u as vaccinating the susceptible, and v as a general control measure (e.g. informing the public). We expand slightly upon the SIR formulation by accounting for a death factor f in equation 2, while maintaining the definition of α and β the same. As our MDP reward definition, we add together two penalties, representing the penalty for the number of infected individuals dying, as well as the magnitude of control actions.

$$R = -(1-f)\alpha i - B \left\| r, u, v \right\|_2 \tag{4}$$

In this rewards definition, B is a hyperparameter we can tune depending on how expensive our control actions compared to the death of infected individuals is. In the results section, plots are for $B = 2e - 4$ unless otherwise mentioned. Empirically, we find that this value of B puts the control penalty and death penalty on the same scale. Realistically, the consequence of one person dying might be much more severe than the cost on control measures. In that case, we could simply lower the value of B.

4 Methods and Results

For our actor network and critic network, we use 3-layer fully-connected neural nets, with an input dimension equal to the output dimension equal to the state dimension, and a hidden layer of 64 units. The hyperparameters for the SIR model is $\alpha = 0.1$, $\beta = 0.5$, $f = 0.5$, $s_0 = 0.9$, $i_0 = 0.1$, $r_0 = 0$, and $B = 2e - 4$. We train our policy for 5 independent runs, each with 40000 timesteps and the result is averaged over 5 runs.

We first compare results trained with A2C and PPO, assuming that actions come from a Beta distribution. For A2C training, we use separate learning rates for the actor optimizer and the critic optimizer, at 1e-3 and 1e-4 respectively. Similarly for PPO, the actor optimizer learning rate is 5e-4, and the critic optimizer learning rate is 7e-5. Figure 1 shows the comparison between A2C and PPO returns, plotted with the death penalty, the control penalty, and the total penalty. Figure 2 shows the change of infected states at the start, middle, and end of training.

We then consider various baseline methods. Specifically, we look at A2C-trained Beta policy, PPO-trained Beta policy, as well as max-action policy (all actions are at maximum value of one), infected-conditioned policy (all actions are 0.9 when the infected proportion is greater than 0.05 of the total population), susceptible-conditioned policy (all actions are 0.5 when the susceptible proportion is lower than 0.9 of the total population), A2C-trained Gaussian policy, do-nothing policy (all actions are zero), and GEKKO-solved optimal control policy.

Figure 3 shows the infected states of all methods mentioned above, and Figure 4 shows the sample rewards from these methods. Note that Figure 4 does not include the do-nothing policy, which surpasses the current y-axis scale. We also negated the rewards for ease of plotting, but the actual rewards should have a negative sign, since it is defined as the penalty in our SIR equations.

5 Conclusion

With the rapid development of COVID-19, the need for more accurate mathematical models to depict disease spread in a population is more essential than ever. We have demonstrated in this work that reinforcement learning is a promising direction of research in this area, given its representation and learning power. RL significantly outperforms baseline methods in accuracy, and optimal control methods in flexibility. For future work, we will consider expanding this simple SIR model to networks of SIR-modelled nodes.

Figure 1: Comparison between Beta policy trained with A2C (left) and PPO (right), averaged over 5 independent runs

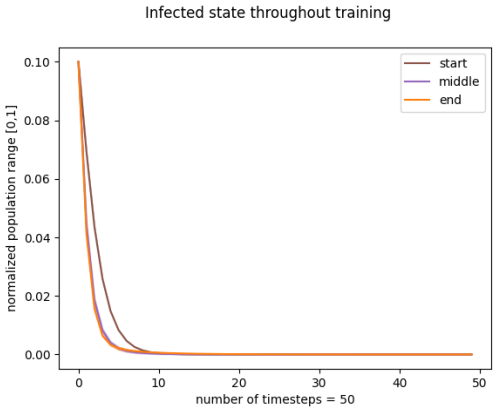

Figure 2: State change as the Beta policy is trained with A2C, averaged over 5 independent runs

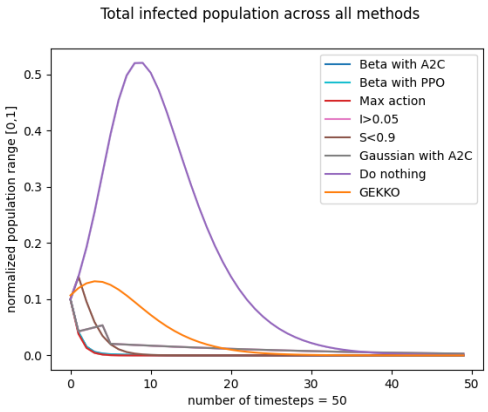

Figure 3: The infected states of all baseline methods, averaged over 5 independent runs

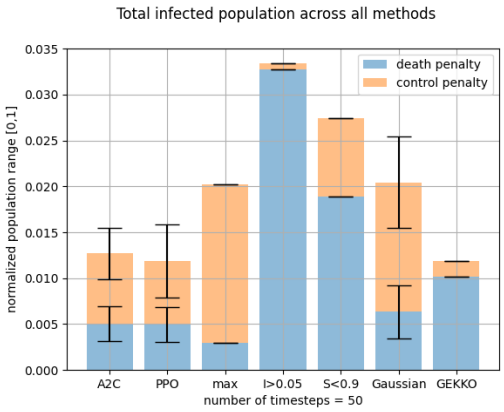

Figure 4: Sampled rewards of all baseline methods

References

Fred Brauer, Carlos Castillo-Chavez, and Carlos Castillo-Chavez. *Mathematical models in population biology and epidemiology*, volume 2. Springer, 2012.

Po-Wei Chou, Daniel Maturana, and Sebastian Scherer. Improving stochastic policy gradients in continuous control with deep reinforcement learning using the beta distribution. In *International conference on machine learning*, pages 834–843, 2017.

Renée X de Menezes, Neli RS Ortega, and Eduardo Massad. A reed-frost model taking into account uncertainties in the diagnostic of the infection. *Bulletin of mathematical biology*, 66(4):689–706, 2004.

Ken TD Eames, Natasha L Tilston, Ellen Brooks-Pollock, and W John Edmunds. Measured dynamic social contact patterns explain the spread of h1n1v influenza. *PLoS Comput Biol*, 8(3):e1002425, 2012.

Julia R Gog, Sébastien Ballesteros, Cécile Viboud, Lone Simonsen, Ottar N Bjornstad, Jeffrey Shaman, Dennis L Chao, Farid Khan, and Bryan T Grenfell. Spatial transmission of 2009 pandemic influenza in the us. *PLoS Comput Biol*, 10(6):e1003635, 2014.

Tiberiu Harko, Francisco SN Lobo, and MK Mak. Exact analytical solutions of the susceptible-infected-recovered (sir) epidemic model and of the sir model with equal death and birth rates. *Applied Mathematics and Computation*, 236:184–194, 2014.

Petra Klepac, Stephen Kissler, and Julia Gog. Contagion! the bbc four pandemic–the model behind the documentary. *Epidemics*, 24:49–59, 2018.

Pieter Libin, Arno Moonens, Timothy Verstraeten, Fabian Perez-Sanjines, Niel Hens, Philippe Lemey, and Ann Nowé. Deep reinforcement learning for large-scale epidemic control. *arXiv preprint arXiv:2003.13676*, 2020.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Cameron Nowzari, Victor M Preciado, and George J Pappas. Analysis and control of epidemics: A survey of spreading processes on complex networks. *IEEE Control Systems Magazine*, 36(1): 26–46, 2016.

Fei-Yue Wang, Jun Jason Zhang, Xinhu Zheng, Xiao Wang, Yong Yuan, Xiaoxiao Dai, Jie Zhang, and Liuqing Yang. Where does alphago go: From church-turing thesis to alphago thesis and beyond. *IEEE/CAA Journal of Automatica Sinica*, 3(2):113–120, 2016.

Yujiang Wang, Mingzhi Dong, Jie Shen, Yang Wu, Shiyang Cheng, and Maja Pantic. Dynamic face video segmentation via reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6959–6969, 2020.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.