Mini-project 1: Deep Q-learning for Epidemic Mitigation

Mekhron Bobokhonov
Tikhon Parshikov

1 Naive approach

Question 1.a) study the behavior of the model when epidemics are unmitigated

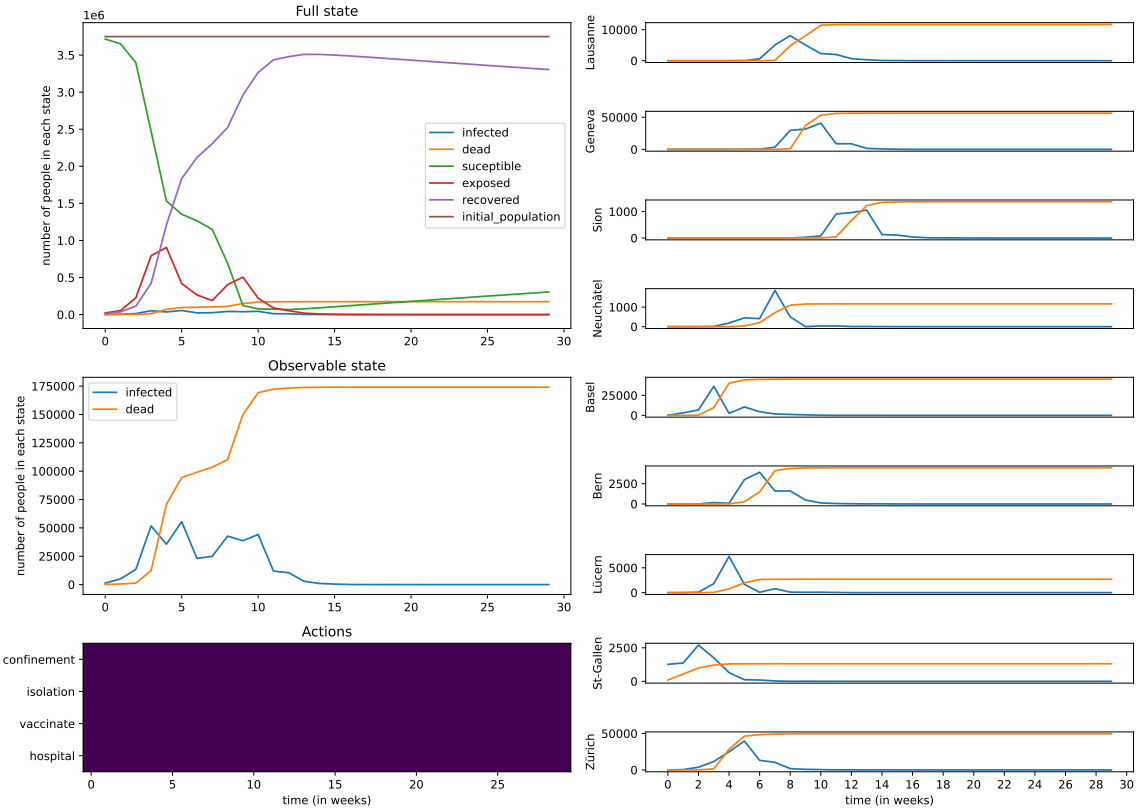

Figure 1: Plots for question 1

Discuss the evolution of the variables over time.

Answer: *Here we used a NullAgent policy when no action performs. With time number of dead people increases (sharply from week 3 to 5 and from week 8 to 10), while the number of infected people is about* 50′000 *from week 3 to week 10 and then decreases.*
Overall, it looks like a pandemic needs several weeks to pick up stream and then it rages for some time causing deaths growing. Starting from week 10-12 less people are susceptible to the virus and therefore almost no new deaths and infections happen. But the amount of susceptible people increases from around week 14 with decrease in the number of recovered people.
Looking into cities level, we can notice that the behaviour of two main characteristics is pretty similar to what we see on the general plot. Hence, in each city there is a peak of infection lasting for couple of weeks and then number of dead and infected stables. The closer city to the source of infection, the earlier this peak happens.
When we do not mitigate the infection, it finishes anyway some time later with the total number of deaths about 175′000.

2 Pr.Russo's policy

Question 2.a) Implement Pr. Russo's Policy

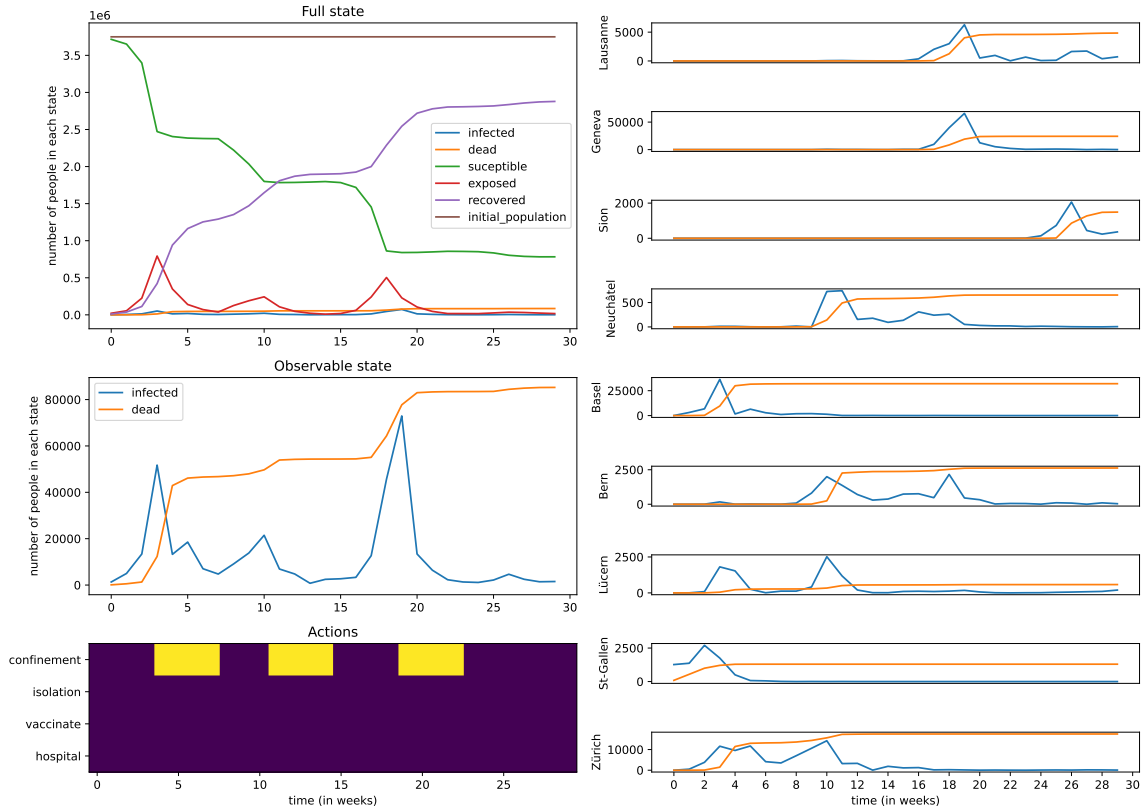

Figure 2: Plots for question 2.a)

Discuss how the epidemic simulation responds to Pr. Russo's Policy (focus on how it differs from the unmitigated scenario).

Answer: *Here we used a policy by Pr. Russo that consists of confining the whole country after the total number of infected people reaches* 20000 *by the end of the week. There were 3 peaks of infections (weeks 3, 10 and 19) when this threshold was held out. It should be mentioned that those peaks happen at the same time when there are peaks in number of exposed people.*
After the confinement action is performed, we can see that the total number of infected people decreases and the number of deaths increases less sharply.
From the level of cities, there are also 1-2 peaks (comparing to the only one in previous question) of infections happening at the same time as peaks in the whole country that is quite obvious.
Finally, using this policy the total number of deaths is around $85'000$ **that is about twice less than with *NullAgent*.**

Question 2.b) Evaluate Pr. Russo's Policy

Answer: *As we can see from the histograms, an average number of deaths is about* $58'000$*, an average cumulative reward is about* -70*. Therefore, graphs from question 2b are obtained using "unlucky" seed, because number of deaths there is much bigger than the mean one.*
From the plot with number of confinement days we can observe that there are 2 main peaks corresponding to confinements for 3 and 4 full weeks, respectively. Other observations relate to the cases when either there were only 2 weeks of confinement or this action for the last time was performed less than 4 weeks before the end of the 30 weeks period.

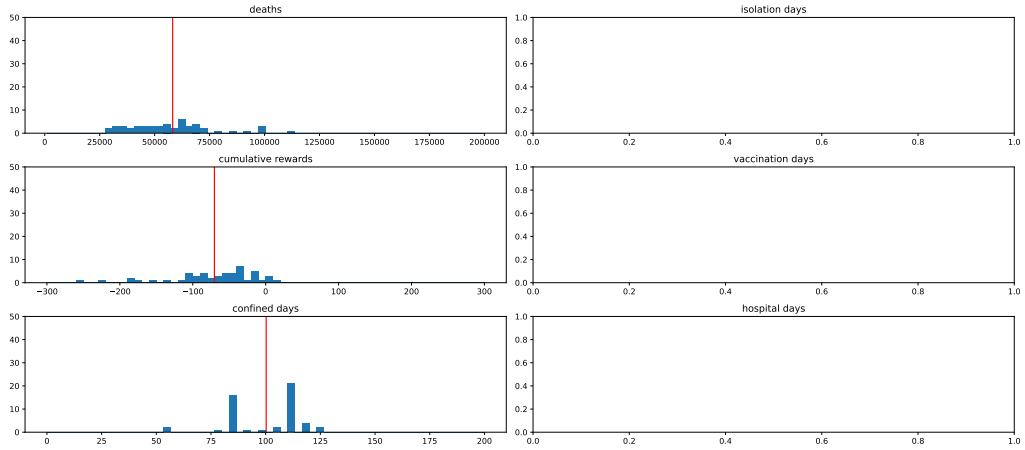

Figure 3: Plots for question 2.b)

3 Deep Q-learning approach

Binary confinement action space

Here we assume that an agent can perform one of two actions: confine or not. To train this agent we used a neural network that has a one dimensional vector consisting of number of infected and dead people in different cities as an input and outputs two Q-values for actions related performing confinement or not, respectively.

Question 3.a) implementing Deep Q-Learning

Figure 4: Plots for question 3.a)

Answer: *We trained Deep Q-network 3 times for* 500 *training episodes, with* $\epsilon = 0.7$, $\gamma = 0.9$ *and Adam optimizer with* $learning\_rate = 5 * 10^{-3}$. *On the plots above (fig. 4) we can see the distribution of reward value during each of the trainings. In all the cases most of rewards were less than zero and only approximately* $12 - 15\%$ *of rewards were positive. Moreover, with time the agent does not prefer actions that lead to higher train reward. It can be explained by the high value of* $\epsilon = 0.7$ *that pushes agent to explore with high probability. Distribution of rewards differs among three training runs: sparse on the first and third and highly dense on the second, but overall there are no relations with time.*

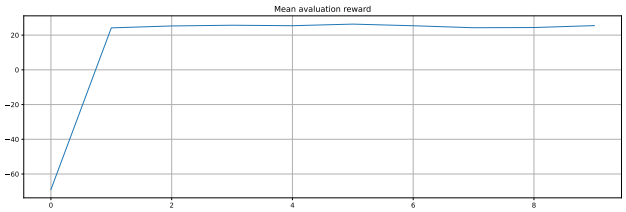

Figure 5: Plot for question 3.a)

Mean evaluation reward has almost optimal value starting from the beginning (fig. 5), then it does not

fluctuate significantly. After all episodes, we can see that the model has the mean reward about $22 - 23$*, as it was during the whole training.*

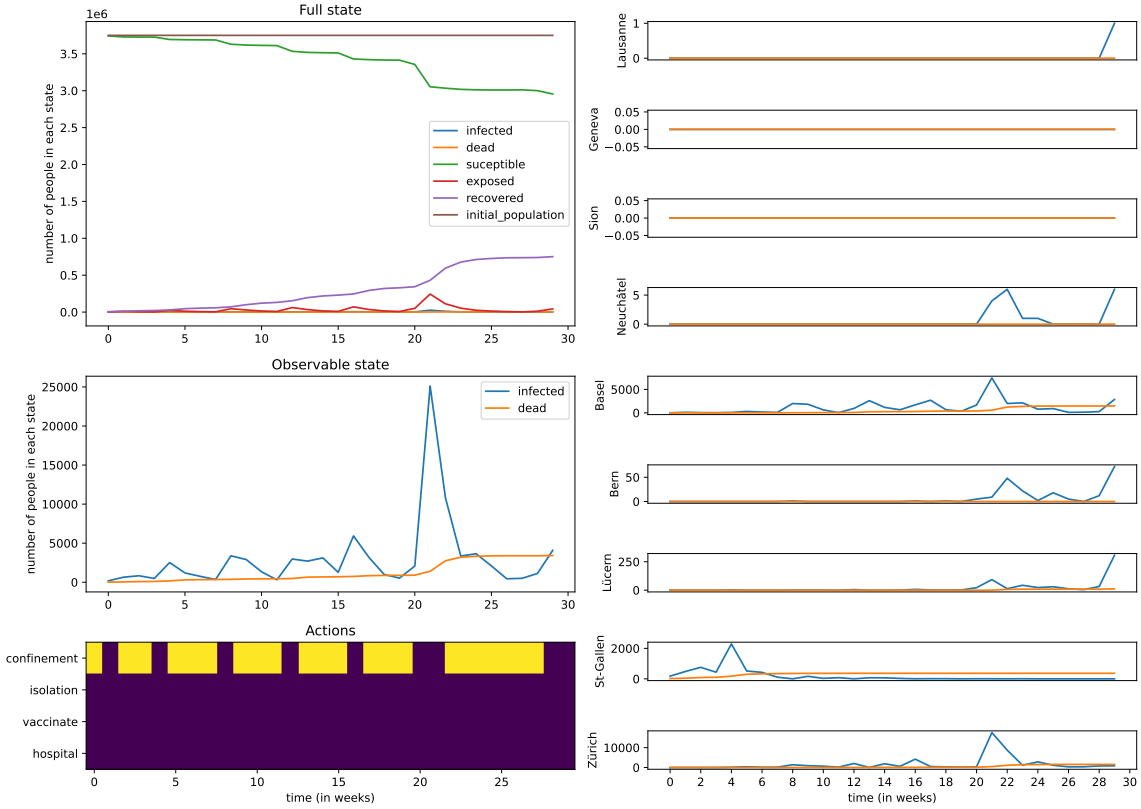

Figure 6: Plots for question 3.a)

From the graph with the π^*_{DQN} *performance (fig. 6) we can see that the* ***agent's behaviour looks very meaningful****. It performs confinement action when the amount of infected people increases sharply or it is too high. Comparing to* π_{Russo}*, confinements might last for 1-2 weeks that does not allow to suppress infection level totally, but at the same time it does not wait until number of infected people reaches* $20'000$*. Overall, there are many peaks of number of infected (and exposed, as it was in question 2) over 30 weeks, but these peaks are much lower than before.*
From the cities point of view, there are several outbreaks in most of them, but each of them is quite low. Also for the first time we can see that there are 2 cities that were not affected by the pandemic at all. It proves effectiveness of the policy used.
Overall, using DQN agent with a binary action space we got total number of deaths of about $4'000$ **that is much better than in all previous policies.**

Question 3.b) decreasing exploration

Answer: *There are a lot of differences (fig.7) between models' performance when we use constant* $\epsilon = 0.7$ *and when we decrease it with time. The evaluation reward is much higher in the latter case and it increases with time that is expected as a result of model's training. Moreover, with decreasing exploration rate it is shown that train reward becomes much bigger with time (training rewards are located from bottom left to top right). It can be explained by the fact that exploration rate does not affect choice of the agent too much and it chooses more greedy action.*

Question 3.c) evaluate the best performing policy against Pr. Russo's policy

Answer: *On the plot (fig. 8) we can see histograms showing how the best policy* π^*_{DQN} *performs on multiple episodes. There is almost no spreading on these histograms, all the episodes have very similar results. It means that this policy is very consistent.*
Mean number of deaths is about $4'400$ *that is more than 15 times better than in Pr.Russo policy. Cumulative reward is always positive with average number of about* 25*. Finally, total amount of confinement days is signif-*

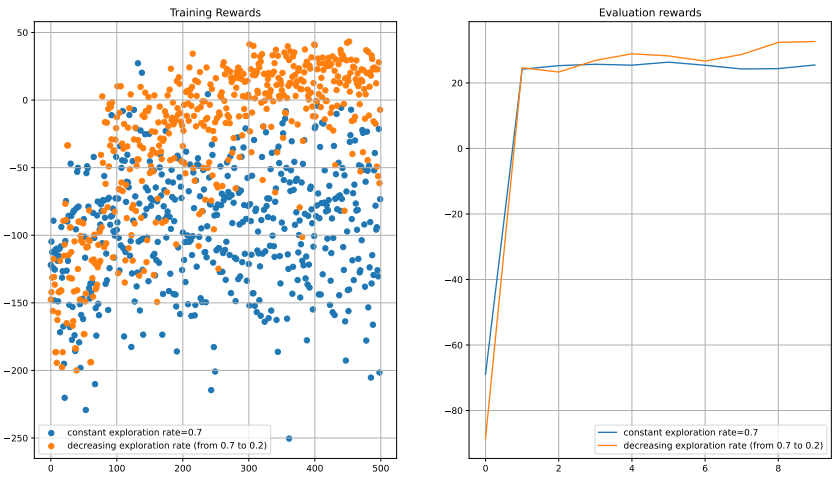

Figure 7: Plots for question 3.b)

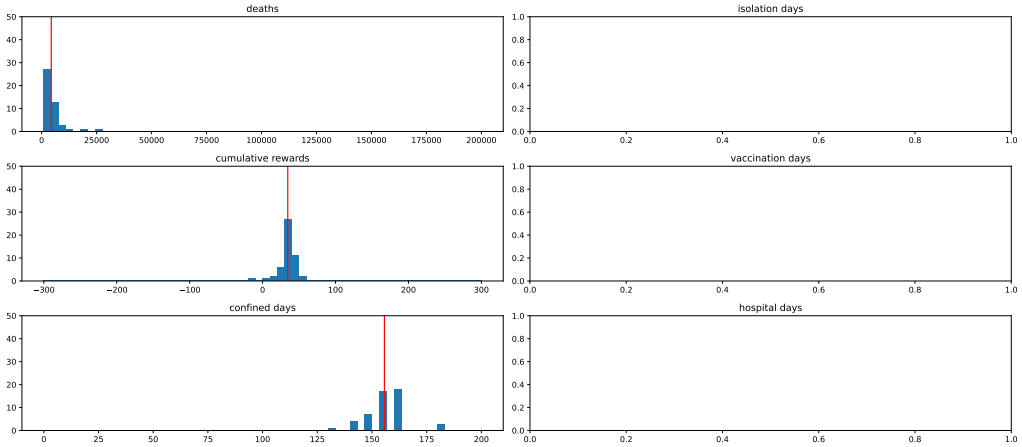

Figure 8: Plots for question 3.c)

icantly bigger than in Pr.Russo policy (with mean around 155*), but as far as it allows us to achieve much less deaths, it does not penalize reward too much.*

Complex action space

Question 4.1.a) (Theory) Action space design

Answer: *There could be many reasons in using one action-observation state instead of another one. One of the possible ideas in favor of this action space is that in reality the number of resources is limited and it is difficult to turn on or turn off any activity, so it can take some time (for example, one week, as it is implemented in this action-space) to start any new mitigation or to stop one that is already in use. For example, creating new hospitals or deploying some restrictions can take some time and demand a lot of resources.*
In this part we will use a neural network again, but need to change input and output of the model. As an input we will use a one dimensional vector with number of infected and dead people in each city, but we will also add 4 more binary elements to this vector: current states of all the mitigations (for example, if confinement action is active now or not etc.). As an output we have a vector containing 5 Q-values using Discrete space (one for each action: 4 "toggles" and 1 "do nothing").

Question 4.1.b) Toggle-action-space multi-action policy training

Answer: *It should be mentioned that in this approach it was very difficult to find a learning rate that allows this model to train. By trial and error method we found finally* $learning\_rate = 0.07$.

Figure 9: Plots for question 4.1.a)

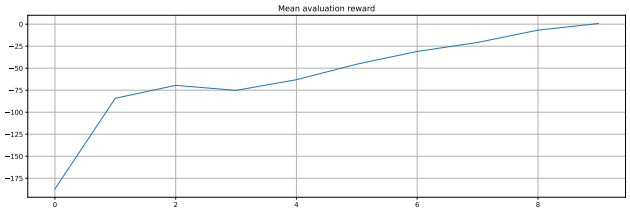

Figure 10: Plots for question 4.1.a)

As in question 3, training reward (fig. 9) increases with time as exploration rate decreases.
Also we can see that the model learns (fig. 10): mean evaluation reward increases with number of episodes trained. But this learning is quite slow, therefore the mean evaluation rewards reaches a final value around 0 after 500 episodes.
It is quite difficult to describe how the model works (fig. 11). Hence, we will try to describe some common patterns. First of all, it should be mentioned that this agent learnt to perform only three actions (confinement, hospital and vaccinate) out of four possible. Moreover, the latter one (vaccination) is taken very rare, therefore the agent mostly works with other two. Most likely this action space is not appropriate for all the existing actions and pushes the agent to be skewed in decision on confinement and hospital sides.
When there is either increase in number of infected people or if this number reaches some level, then confinement action is activated, while when the amount of infected people falls, this action is toggled to inactive. Regarding the hospital mitigation, it is quite difficult to find any correlation in general between number of infected or dead people and decision to perform hospital action.

Question 4.1.c) Toggle-action-space multi-action policy evaluation

Answer: *From the plot (fig. 12) we can observe that this trained agent is less consistent than one in question 3.c and distributions are more skewed, though fluctuations around mean value are not very big. From the average number of deaths (* $13'300$*) and mean cumulative reward (* 14*) sides, this "toggle" agent is between two and three times worse than the binary action policy agent, whereas it has almost the same mean number of confinement days (*155*). Also this agent sometimes takes hospital action with an average number of about* 56 *and seldom vaccinate mitigation with a mean value around* 12 *days.*

Question 4.1.d) (Theory) question about toggled-action-space policy, what assumption does it make?

Answer: *As it was already mentioned in the architecture part, we assume that information about previous states of mitigations is known and therefore could be used as an input. Moreover, we use an action space where all the actions have only two states (active and inactive). For example, for the space where there are more states than two (small, middle, large), this toggling technique would be highly inefficient.*

Factorized Q-values, multi-action agent

In this question we need to change our previous architecture a little bit. First of all, now we have MultiBinary space of dimension 4 instead of the Discrete one. As an input our network uses again a vector containing number of dead and infected people in all the cities. As an output we have 4 pairs of Q-values that define if each action should be taken or not.

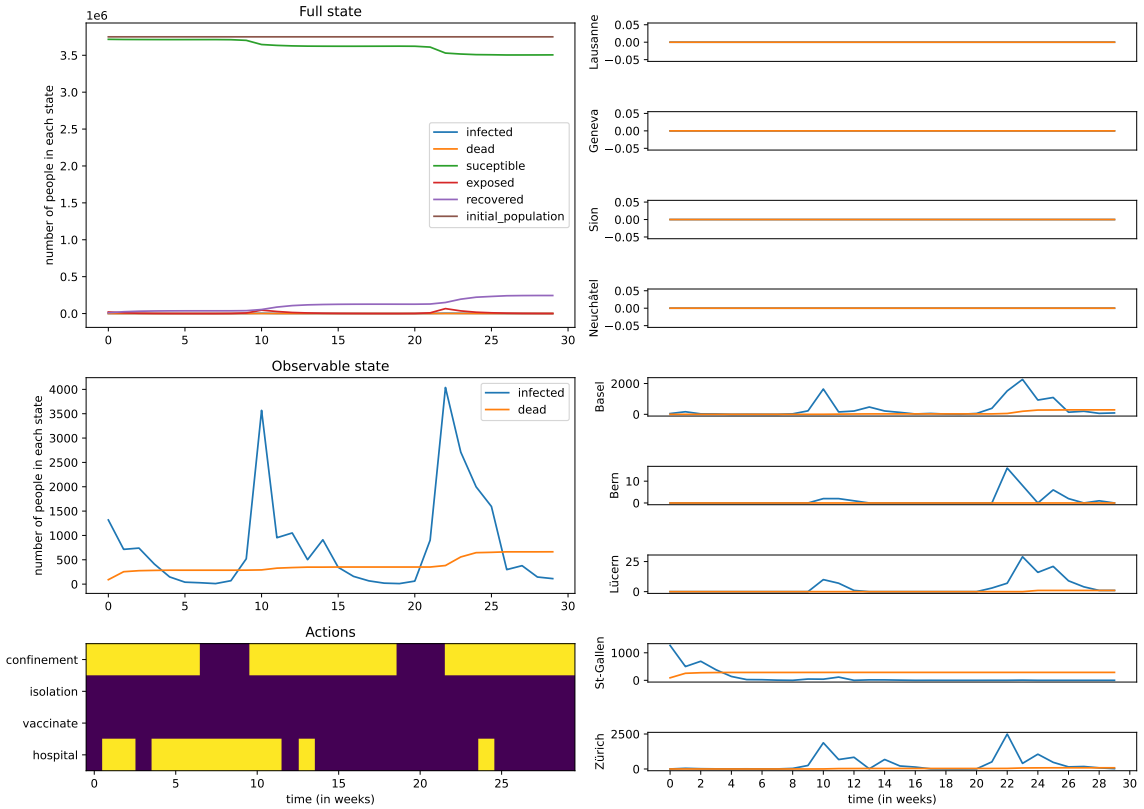

Figure 11: Plots for question 4.1.a)

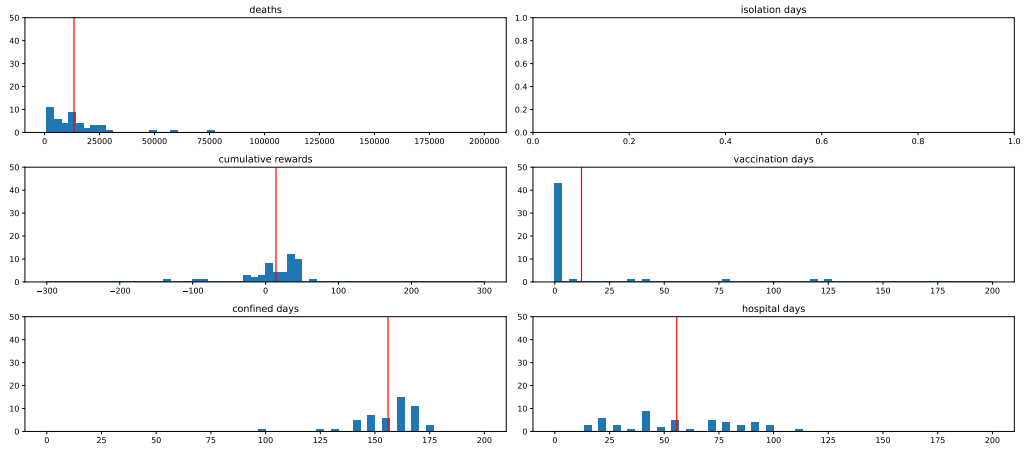

Figure 12: Plot for question 4.1.c)

Question 4.2.a) multi-action factorized Q-values policy training

Answer: *As we can see from the plots with train rewards (fig. 13), model successfully learns with time and tends to take actions that lead to the bigger reward. Mean evaluation reward increases with time as well (fig. 14).*
In order to try to understand model behaviour, we can look at fig.15. Most probably the agent performs confinement action when there are some peaks and increases in number of infected people. While when these two peaks happen close to each other, it also takes hospital mitigation.
There are 4 cities that were not affected by the pandemic. In other cities the situation looks in worst case the same as in general and in best cases there are much less peaks of infected people. ***Using this agent the number of dead people is extremely low!***

Figure 13: Plots for question 4.2.a)

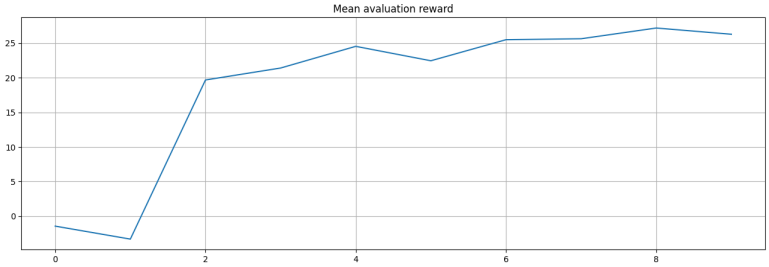

Figure 14: Plots for question 4.2.a)

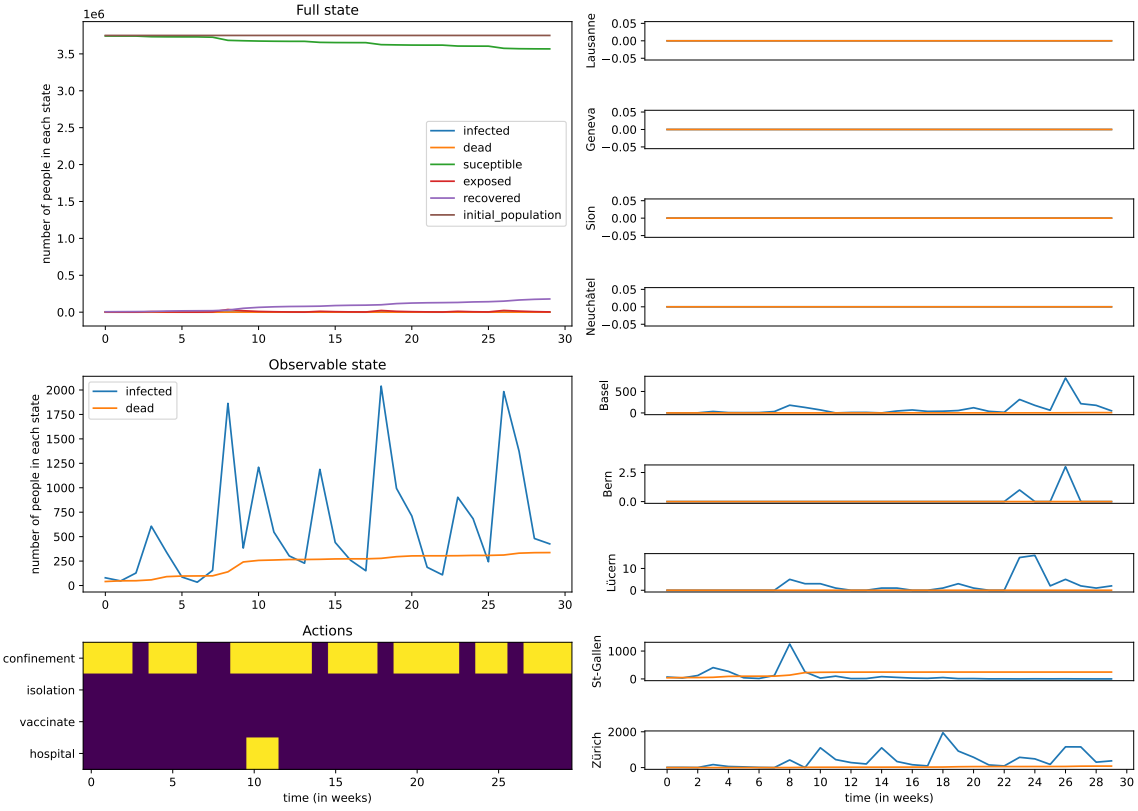

Figure 15: Plots for question 4.2.a)

Question 4.2.b) multi-action factorized Q-values policy evaluation

Answer: *According to the plot (fig. 16),* π^*_{Factor} *allows to achieve on the average just about* 520 *deaths that is significantly better than in the* π^*_{toggle}*. From the side of confinement and hospital days, these agents have almost the same mean values about* 162 *and* 13*. respectively. The average cumulative reward of about* 28 *is lower than for the previous "toggle" agent. Also it should be mentioned that this agent learnt to perform only*

two actions out of four possible and distributions for different metrics are very dense (especially comparing to ones in question 4.1)).

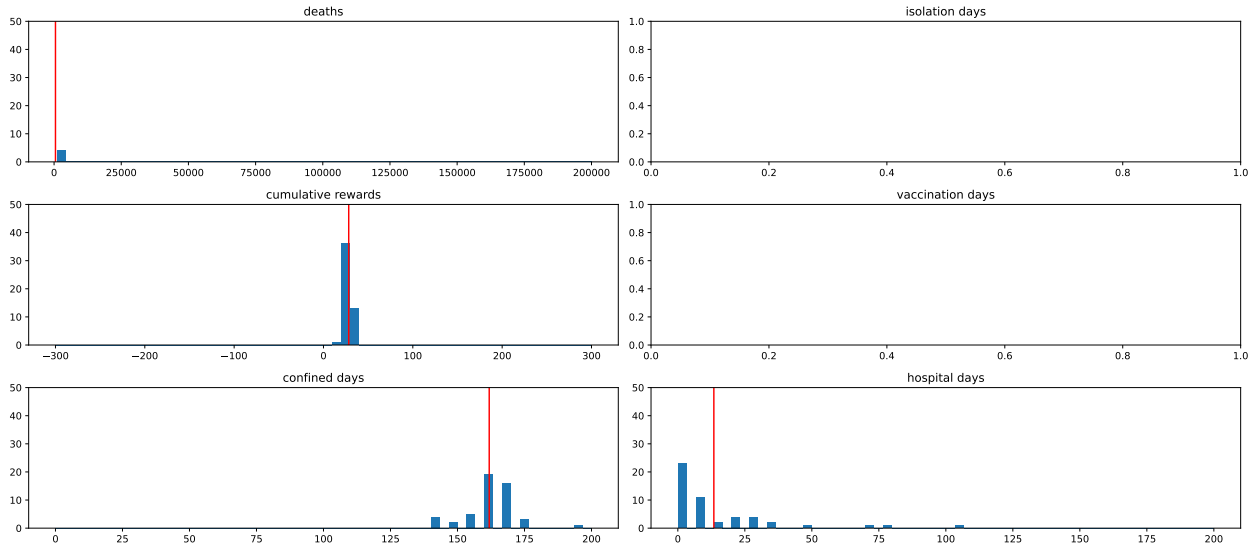

Figure 16: Plots for question 4.2.b)

Question 4.2.c) (Theory) Factorized-Q-values, what assumption does it make?

Answer: *Our action space consists of sets of 4 binary decisions. In factorized Q-values policy, we assumed that in each action these decisions are made independently. Such an assumption is not always correct, for example, if we consider an agent who is a robot whose goal is to go 100 m forward, and whose actions consist of paired values (direction, speed), then these two variables strongly depend on each other, because in the forward direction, the speed must be high in the reverse direction, close to 0.*

4 Wrapping Up

Question 5.a) (Result analysis) Comparing the training behaviors

Answer: *Pr. Russo's policy performs the worst as the most simple and basic one with a great number of deaths and negative reward. But even this policy significantly outperforms the Null Agent policy when no action is taken.*
Toggled-action-policy is much better than Russo's since it achieves positive reward, but not so good comparing to the other two DQN policies we implemented. Moreover, even though it is allowed to use any of four possible actions, it never uses isolate mitigation and very rare vaccinate one.
Despite of the fact that logically factorized policy should achieve the best results, it is only on the second place. This abruptness happens due to the very heavy assumption of actions' independence. In this case the agent learns only to perform confine and hospital actions out of possible four. We suppose that if there had not been such an assumption, this action-space would have perform the best among all the explored. Single-action policy is the best one we achieved in this project even though it just decides if we need to take a confinement or not. Such a good performance can be explained by the fact that in the current environment confinement action is the most important one that is also confirmed by the fact that other policies strongly prefer to take it.

Question 5.b) (Result analysis) Comparing policies

Answer: *In the table 1 metrics for different policies are shown. The best value among all the policies for each metric is bold. It does not make a lot of sense to observe all the metrics individually, as far as for example π_{Russo} has the lowest amount of confinement days, it's cumulative reward is very small and number of deaths is extremely big. Overall, according to the final reward, DQN policy is the best one, but for the total number of deaths factorized policy is much better. All the three DQN policies have almost the same number of confinement days that is can be considered as an optimal value.*

	$avg[N_{confinement}]$	$avg[N_{isolation}]$	$avg[N_{vaccination}]$	$avg[N_{hospital}]$	$avg[N_{deaths}]$	$avg[N_{cumulative}]$
π_{Russo}	**100.24**	-	-	-	58202	-70.29
π_{DQN}	155.54	-	-	-	4405	**34.63**
π_{toggle}	157.08	0	12.18	54.6	12033	17.21
π_{factor}	161.84	0	**0**	**13.16**	**520**	28.29

Table 1: Comparing policies

Question 5.c) (Interpretability) Q-values

Answer: *For the π_{DQN} shown in fig. 17 with time Q-values for both actions are between 13 and 18. In the moments and there are many moments when the difference between two actions is very small. Confinement action has almost never negative Q-value that is more or less logical, while there are many cases when non-confinement action are highly negative. It can be explained that in this moment peaks of infection happens and therefore performing non-confinement action will lead to worsening.*
For the π_{factor} (fig. 18) confinement action with time converges to some value around 2.8, whereas non-confinement fluctuates a lot. For the vaccination and isolation mitigations there is always better not to perform it as far as the difference between taken and non-taken them is very significant, but it decreases a little bit with time. For the hospital action there is only one moment (week 11) when it is better to take it. This week contains most of the extreme values for the actions.

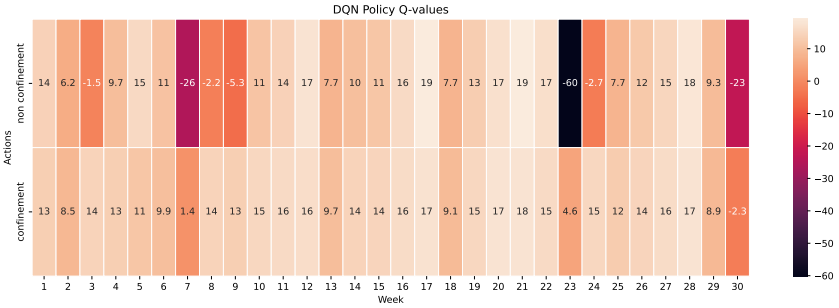

Figure 17: DQN heatmap

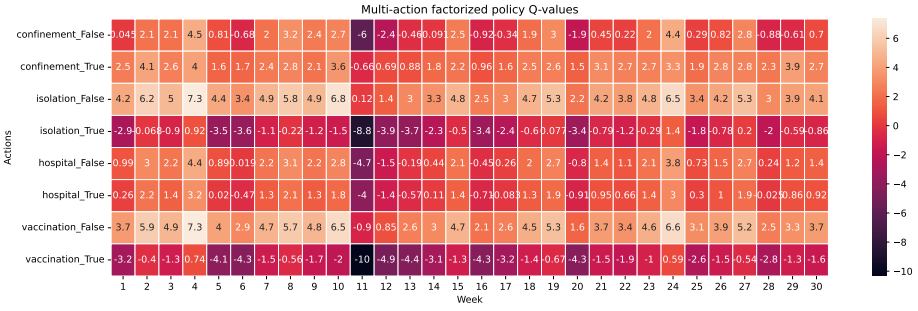

Figure 18: Factorized heatmap

Question 5.d) (Theory), Is cumulative reward an increasing function of the number of actions?

Answer: *No. For example, our implementation of a DQN agent with a binary action space confinement, non-confinement reached a reward value exceeding 20 when performing an evaluation, while a multi-action DQN agent with a toggle-action-space dimension of 5 reached a maximum reward value of 0. In general, a huge number of possible actions for a simple environment can make it difficult for an agent to learn a good policy.*