Wet3 RL 046203

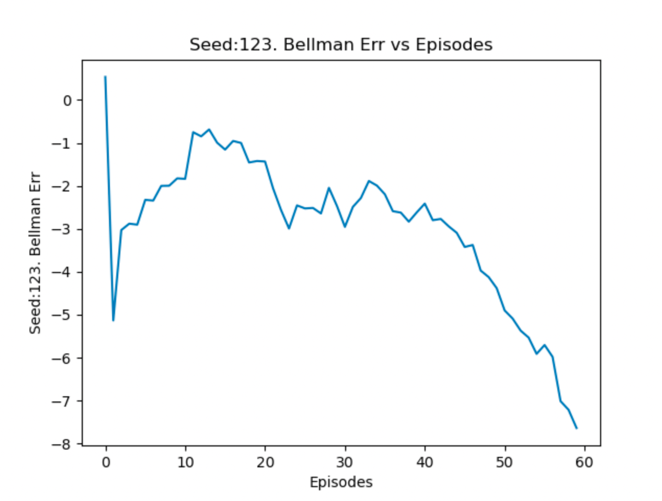
**Question 4 Q-learning**

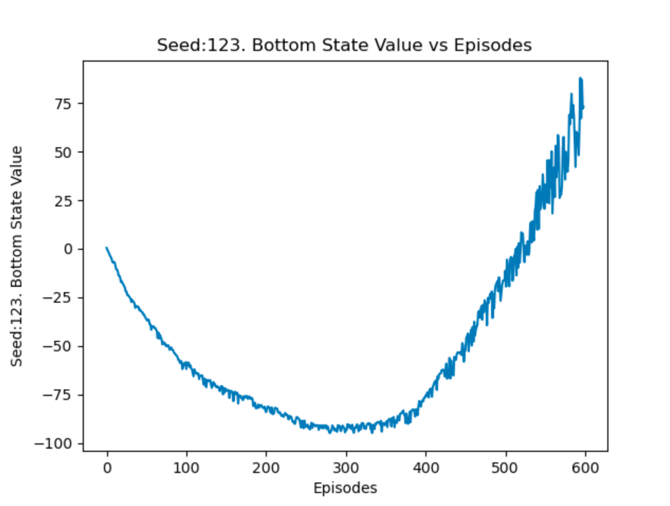
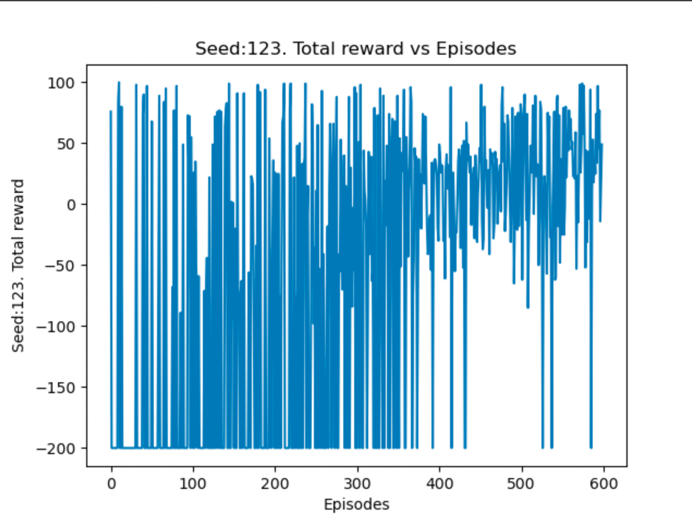
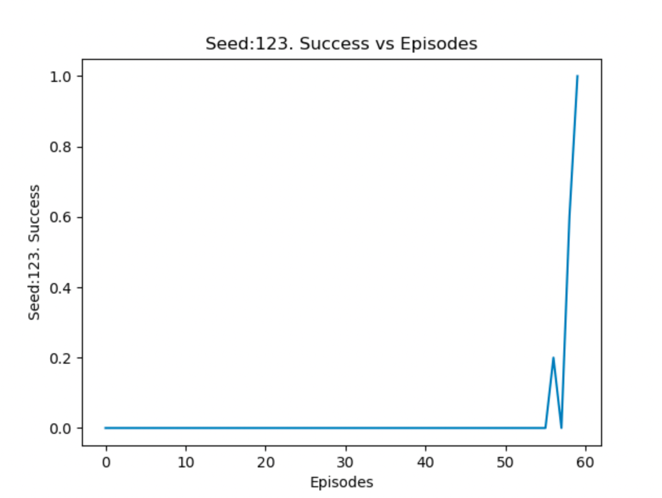
1. Each episode is limited to 200 steps and the agent gets a reward of 100 when it reaches to goal, else it gets -1 at each time-step. In order to achieve a cumulative reward of -75 in this setting, the agent must reach the peak within the first 175 steps.

Setting a relatively higher positive reward when reaching the peak gives the agent a better motivation to shift parameters towards this high reward. It increases the step update and thus helps the optimization process. Moreover, the fact that reaching the goal gives a high reward also give high Q-value to states close to the goal. It also gives higher Q-values to states further away and helps to propagate the good signal.

We note that thanks to the reshaped reward, we can track easily the number of steps needed to achieve the goal (since each non-reaching state give -1 reward).

1. We code the evaluation process.
2. We plot the different values as required during the training process for 3 seeds: 123, 234, 345. The training process stops only when an average score of -75 is obtained for 10 test episodes.

123:



234:

Chart, line chart

Description automatically generatedChart, line chart, histogram

Description automatically generatedChart, bar chart

Description automatically generatedChart, box and whisker chart

Description automatically generated

345:

Chart, line chart, histogram

Description automatically generatedChart, line chart, histogram

Description automatically generatedChart, bar chart

Description automatically generatedChart

Description automatically generated with medium confidence

Let’s describe what’s in each plot.

The total reward plots show the episode reward for all episode during the training process. In all these plots, we see that in most episodes, the agent doesn’t reach the peak since in this case, the reward is -200 and we observe a reward of -200 for many episodes especially for the seed 234. For the other seeds, we see that the mean reward per episode rises with training as the Q function begins to be more and more accurate.

The Success vs Episodes plots show us the success rate for the 10 test episodes while a success is reaching the peak before 200 steps. In the two converging seeds, the Success rate is almost always 0 expect when it reaches the threshold of the average score.

The Bottom State Value (BS value) plots show the value of the bottom state [-0.5,0] in the Q-function approximation of the Solver. For the seeds with success (123, 345), we observe that the BS value decreases until -75/-80 and then increases. We wanted to reach at least a BS value of -75 since this is the minimal value where the cart reaches the peak. However, the Q-function approximation of that state increases significantly. We assume this is due to the learning rate which is high (0.05) but which allows to learn fast enough to pass the test threshold (lower values lead to weak and slow learning).

The Bellman Error plots show the average Bellman error in the training phase for each episode. We see that the Bellman errors decrease during the training process.

1. Using the original parameters, the Solver took a huge number of episodes to converge, or it didn’t converge. The Q function approximation didn’t converge to a satisfying value.

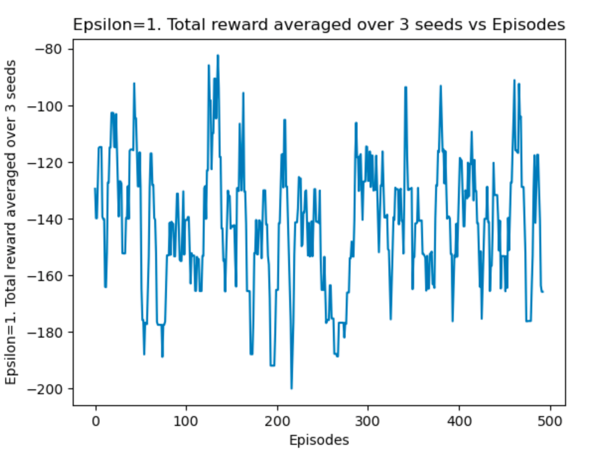
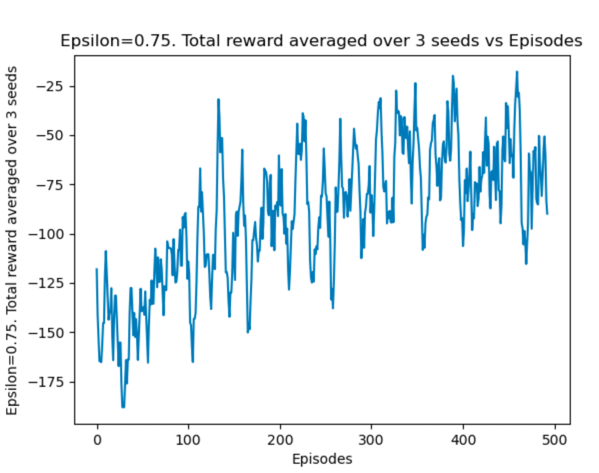
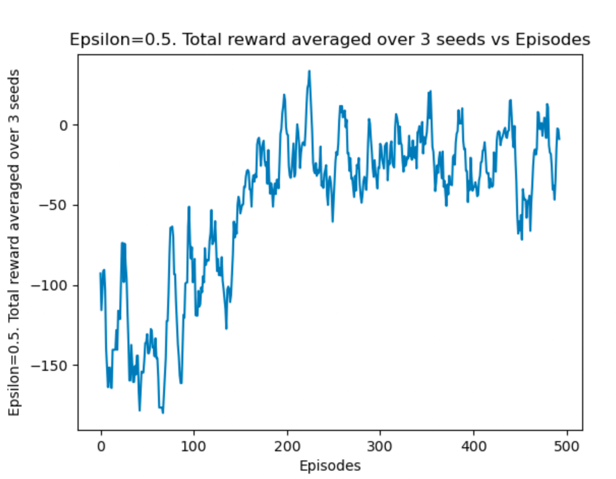
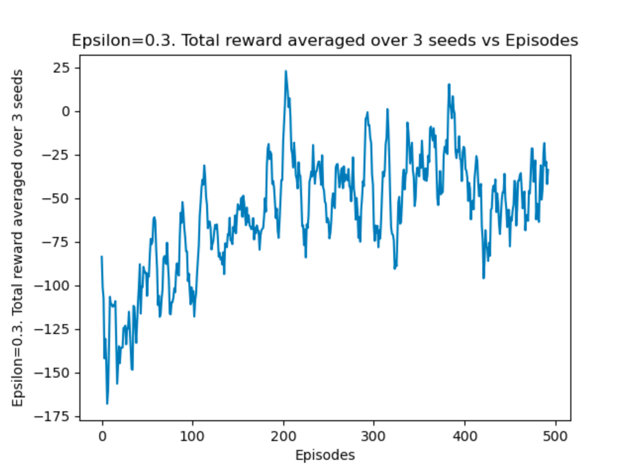
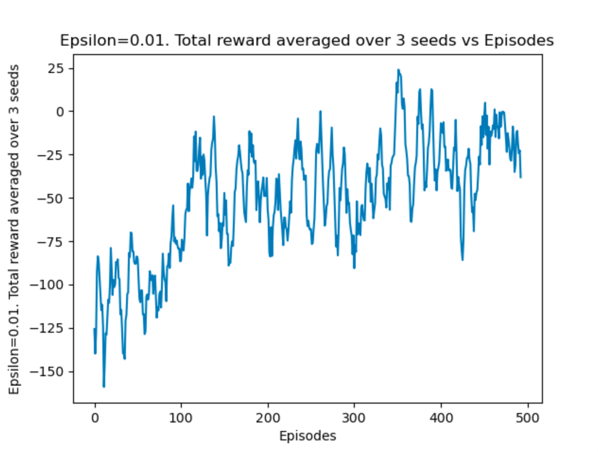
However, using the suggested parameters, we obtained the plots above and saw a better convergence. We can assume that for the two seeds 123 and 345, the Q-function approximation was reasonable since the Bottom State value reached higher value than the minimal value needed to reach the peak -75. It seems that to obtain the required threshold for 10 episodes, the solver had to continue improving the Q-function (above -75) such that averaging 10 test episodes gives at least -75.

1. In order to find what is the best value, we run the solver for each one for 500 episodes and average over the 3 seeds. Then, we smooth slightly the plots in order to get readable plots as shown below.

We observe that for , the average episode reward is the highest and gets values around 0. For , the episode rewards are close to 0 too. However, for larger , we observe a strong dropout in the episode rewards.

Thus, the best is 0.5.

For larger , the exploration takes too much importance and doesn’t allow the learned policy to be expressed in the environment.

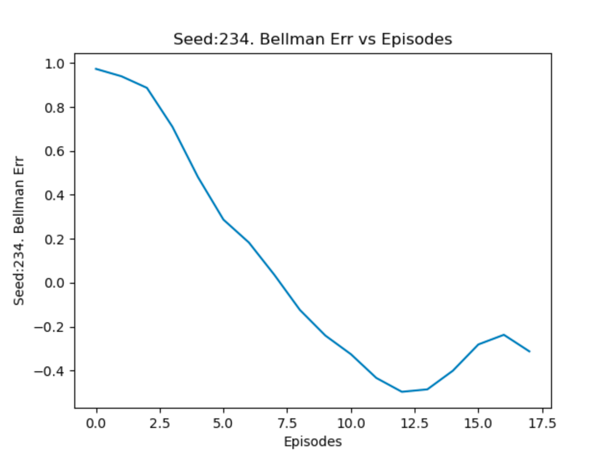
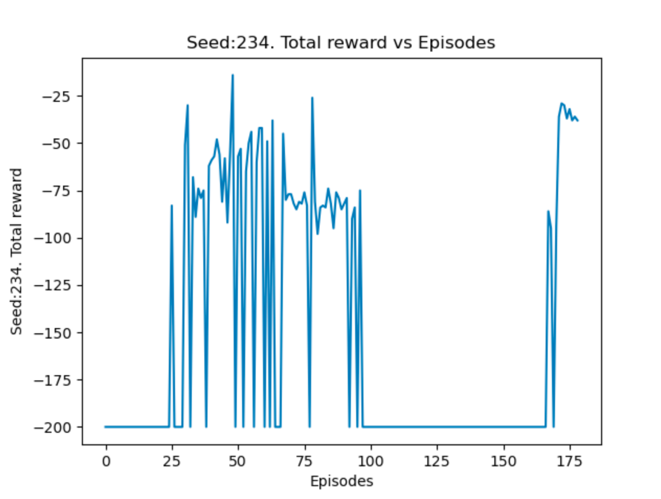
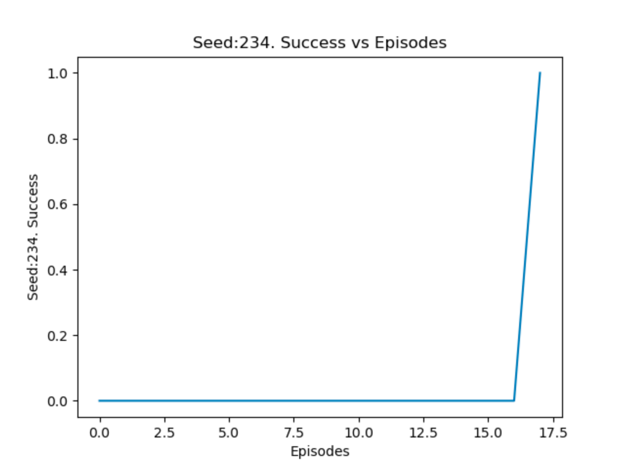
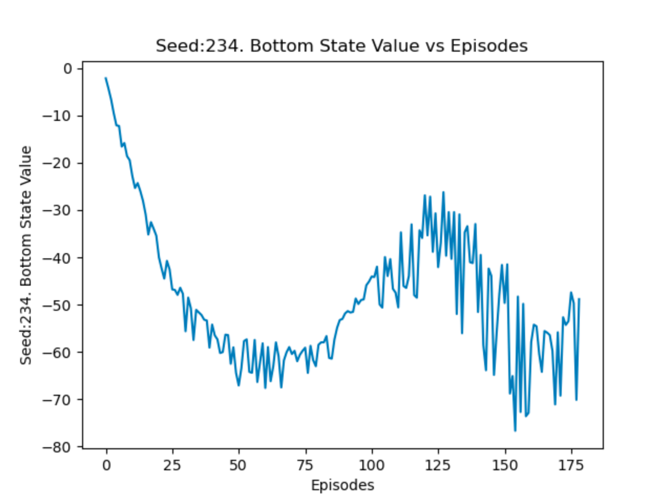


**Question 5 Bonus Q-learning**

We run the Mountain Car where the car always starts from the bottom position with 0 velocity.

With all the tested seeds (123, 234, 345), the success was very fast in the new setting. This was surprising since at a first glance, the exploration would not be sufficient (we used , which added to the astonishment). I made sure that the interactions with the environment always start from the bottom state.

For example, we show the plots for seed 234 which succeeded in 180 episodes:



As we can see, the bottom state value decreases fast in these settings, it shows that when starting always from the same bottom state, the solver learns quickly the value of that state. We expected to see failure in this setting, however we saw convergence even more quickly.

Our guess is that learning from that bottom state gives these states negative values (thanks to our custom reward function) and the solver chooses to escape from these states when it’s possible. Moreover, states with velocity capable of escaping very negative values to 0 value states or to positive value state (success) will be encouraged and thus, the solver will tend to go to these states.

Note that for other seeds, the convergence may be even more fast. For the two other seeds, the learning was even faster with 40 and 70 episodes needed to succeed according to the 10 episodes criteria.

We still tried to enhance the performance, since for seed 234, the algorithm needed 180 episodes. So, we increased the value to a larger value and make it decrease at some points like this . This really helps to pass the evaluation criteria and helps learning. Here we see the success rate and the Bottom state value for 30 episodes until success.

