Report Individual Assignment
- *Flappy Bird* -

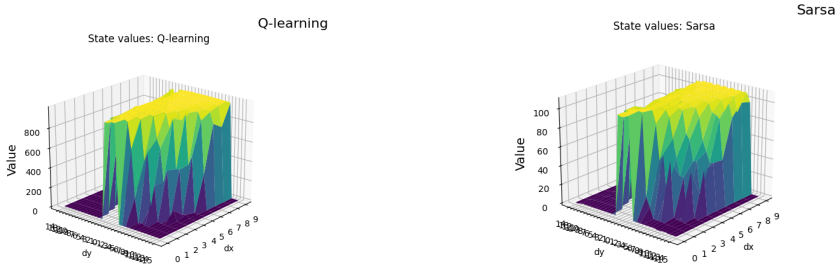
Sophia Chirrane

CentraleSupélec, Université Paris-Saclay
surname.name@student-cs.fr

1 Agents choice

The TextFlappyBird-v0 environment is not a MDP : therefore we do not have access to the transition matrix state action that is necessary to build directly the optimal policy. Thus, we need to adopt a model free control approach in order to estimate a good policy. I implemented two such approach : a Q learning approach and a Sarsa approach (see: https://github.com/sopchi/FlappyBir_RL.git). Both agents uses an epsilon greedy policy to estimate the optimal policy. They have several parameters such as the discount, the step size, the epsilon and the number of episodes needed for the estimation. Both agent can do 2 actions (go up or down), and have the same state space i.e all the possible couples dx and dy.

2 Agents comparison

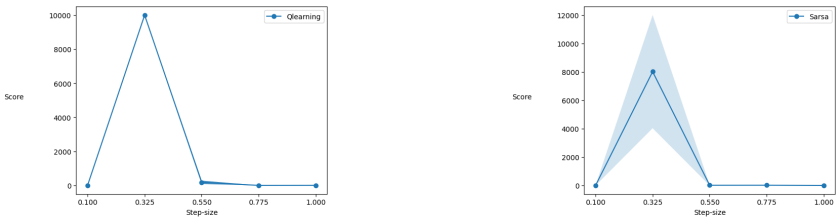


QLearning Sarsa

Fig. 1. The State Value function plot for each agent after 50K episodes : $\gamma = 1$, $\epsilon = 0.1$, $\alpha = 0.1$

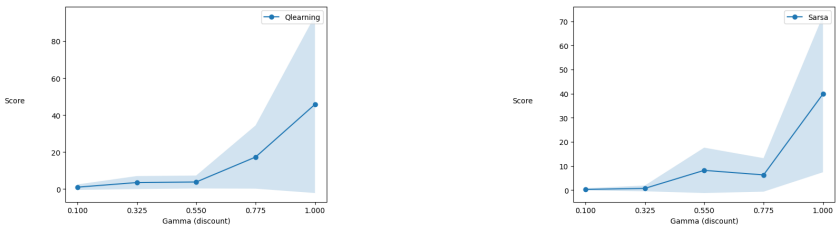
We can compare the two approaches that are quite similar and share the same parameters. The Figure 1 presented the state value function of each agent, we can see that they are really close.

We can also perfom a sentitivity analysis of both agents to parameter change.



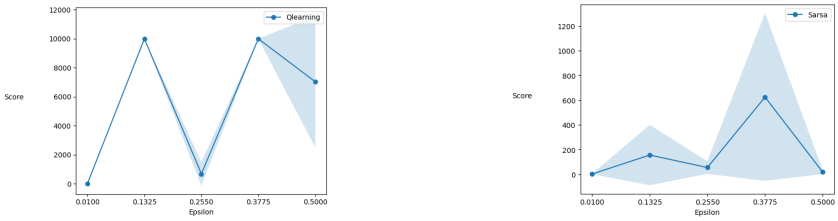
QLearning Sarsa

Fig. 2. Step size sensitivity for each agent after 3K episodes for the same parameters



QLearning Sarsa

Fig. 3. Discount sensitivity for each agent after 3K episodes



QLearning Sarsa

Fig. 4. Epsilon sensitivity for each agent after 3K episodes

Figures 2, 3, 4 present the mean score and standard deviation as uncertainty for 10 repetitions after trained each agent with 3K episodes. We can see that for the step size parameter (α) for both agents the optimal value seem to be around 0.325. Moreover the Sarsa agent seems to have more variability. For the discount γ, this time the QLearning agent seems more variable, the optimal value is around 1. For the epsilon parameter both agents seems to behave the same with two optimal values 0.1325 and 0.3775. Finally, a chanage in α or on ϵ seems to have bigger impact than a change in γ.

We can finally look at the convergence of each agent (Figure 5). Apparently the QLearning agent seems to reach the fixed maximal score (10K) faster than the Sarsa agent.

QLearning Sarsa

Fig. 5. Scores with respect to the number of episode of training

3 Discussions

3.1 Environment

The main differences between the two implemented environment is that the TextFlappyBird-screen-v0 environment provide much more information to the agent such as the position not only of the closer pipe but also the following ones that are already on the screen.

Due to this difference the optimal policy of an agent evolving in the TextFlappyBird-screen-v0 environment might be different from the one estimated with the two implemented agent. For instance, the optimal policy might take into account the information of the following pipes position in order to better positioned the agent and optimised the distance for the one after.

Therefore the main limitations of using the two implemented agents is the fact that the estimated policy is probably not optimal.

3.2 Real Flappy Bird

Contrary to the environments TextFlappyBird-screen-v0 and TextFlappyBird-v0 in the real flappy bird environment when the agent chooses to go up the following state given by the environment is just $dy + 1$. In other word we know exactly in which state the agent will be after choosing one of the two possible action. We can therefore just go up or down according to the sign of dy.

The estimate policies of the implemented agents take into account the fact that the environment react differently according to how far we are to the hole (the value of dx). These policies are therefore not optimal for the real flappy bird environment.