Exercise 8

Kai Schneider

June 22, 2021

Task 1 n-step and eligibility traces

a.) n-step bootstrapping or Dyna-Q

Which algorithm performs better depends on the number of steps in the first episode.

- If the *first episode has **less** than n* steps both would perform the same. Before reaching the goal, the algorithms can't modify their behaviour since all rewards are still equal. If a path is found, both have to do the same number of updates, which results in the same performance. Starting with the second episode, Dyna-Q should perform better due to using the information from the previous episode.
- If the *first episode has **more** than n* steps Dyna-Q can directly update all n steps after reaching the goal. n-bootstrapping wouldn't be able to do this, so Dyna-Q would clearly perform better in this case.

b.)

Given:

$$G_t^\lambda = (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}G_{t:t+n} \tag{1}$$

$$G_{t:t+n} = R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{n-1}R_{t+n} + \gamma^n V(S_{t+n}) \tag{2}$$

Plug (2) ind (1):

$$\begin{aligned}
G_t^\lambda &= (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}\big(R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{n-1}R_{t+n} + \gamma^n V(S_{t+n})\big) \\
&= (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}R_{t+1} + (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}\gamma\underbrace{\big(R_{t+2} + \gamma R_{t+3}\cdots + \gamma^{n-2}R_{t+n} + \gamma^{n-1}V(S_{t+n})\big)}_{G_{t+1:t+n}} \\
&= (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}R_{t+1} + (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}\gamma G_{t+1:t+n} \\
&= (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}R_{t+1} + \underbrace{\gamma(1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}G_{t+1:t+n}}_{G_{t+1}^\lambda} \\
&= (1-\lambda)\sum_{n=1}^{\infty}\lambda^{n-1}R_{t+1} + \gamma G_{t+1}^\lambda
\end{aligned}$$

Task 2 n-step sarsa

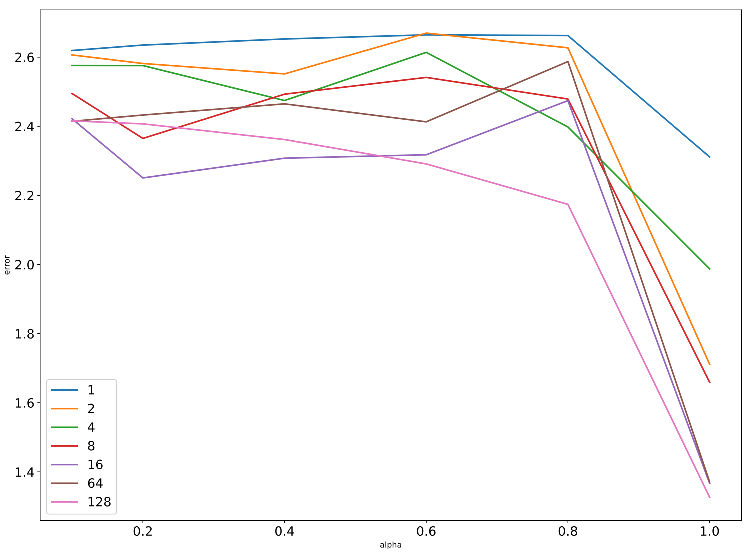

Figure 1: plot of n-sarsa performance

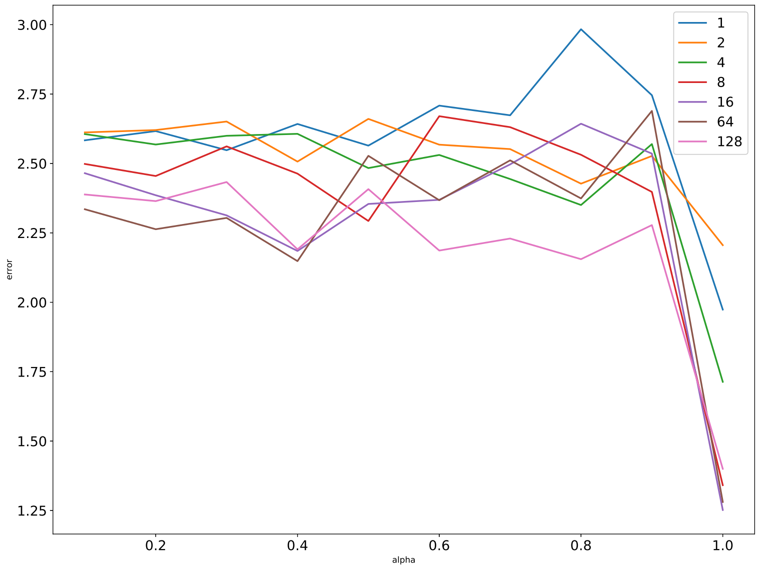

Figure 2: plot of n-sarsa performance