0 Instructions

Homework is due Tuesday, April 30, 2024 at 23:59pm Central Time. Please refer to `https://courses.grainger.illinois.edu/cs446/sp2024/homework/hw/index.html` for course policy on homeworks and submission instructions.

1 Bellman Equation

1.1

$$Q^\pi(s,a) = R(s_0 = s, a_0 = a) + \sum_{t=0}^{\infty} \mathbb{E}_{\substack{a_{t+1}\sim\pi(a_{t+1}|s_{t+1}) \\ s_{t+1}\sim p(s_{t+1}|s_t,a_t)}} [\gamma^{t+1} R(s_{t+1}, a_{t+1})]$$

1.2

$$\begin{aligned} Q^\pi(s,a) &= R(s_0 = s, a_0 = a) + \sum_{t=0}^{\infty} \mathbb{E}_{\substack{a_{t+1}\sim\pi(a_{t+1}|s_{t+1}) \\ s_{t+1}\sim p(s_{t+1}|s_t,a_t)}} [\gamma^{t+1} R(s_{t+1}, a_{t+1})] \\ &= R(s_0, a_0) + \gamma \sum_{s_1} p(s_1|s_0,a_0) \sum_{a_1} \pi(a_1|s_1) \\ &\qquad \{R(s_1,a_1) + \sum_{t=1}^{\infty} \mathbb{E}_{\substack{a_{t+1}\sim\pi(a_{t+1}|s_{t+1}) \\ s_{t+1}\sim p(s_{t+1}|s_t,a_t)}} [\gamma^{t+1} R(s_{t+1}, a_{t+1})]\} \\ &= R(s_0, a_0) + \gamma \sum_{s_1} p(s_1|s_0,a_0) \sum_{a_1} \pi(a_1|s_1) Q^\pi(s_1,a_1) \end{aligned}$$

1.3

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[R(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$$

1.4

If for every steps, the reward is the maximum reward, then the Q-value will be the maximum reward. In this case,

$$\max Q* = R_{max} + \gamma R_{max} + \gamma^2 R_{max} + \cdots = \frac{R_{max}}{1-\gamma}$$

1.5

(a)

$$Q(s_1, a_1) = 0 + 0.5 * (-10 + 0.5 * 0 - 0) = -5$$

(b)

$$Q(s_1, a_2) = 0 + 0.5 * (-10 + 0.5 * 0 - 0) = -5$$

(c)

$$Q(s_2, a_1) = 0 + 0.5 * (18.5 + 0.5 * (-5) - 0) = 8$$

(d)

$$Q(s_1, a_2) = -5 + 0.5 * (-10 + 0.5 * 8 + 5) = -5.5$$

2 Combination Lock

2.1

Denotes N_{s_i} as the expected number of steps from state 1 to state i. Easily, we have $N_{s_1} = 0$. Then, we have following equations:

$$\begin{aligned} N_{s_2} &= 0.5 * (N_{s_1} + 1) + 0.5 * (N_{s_1} + 1 + N_{s_2}) \\ &= N_{s_1} + 1 + 0.5 N_{s_2} \\ \Rightarrow N_{s_2} &= 2N_{s_1} + 2 \end{aligned}$$

Similarly, we have,

$$\begin{aligned} N_{s_3} &= 0.5 * (N_{s_2} + 1) + 0.5 * (N_{s_2} + 1 + N_{s_3}) \\ \Rightarrow N_{s_3} &= 2N_{s_2} + 2 \\ &\vdots \\ N_{s_n} &= 2N_{s_{n-1}} + 2 \end{aligned}$$

Iteratively substitute N_{s_i} into $N_{s_{i+1}}$, we get,

$$\begin{aligned} N_{s_n} &= 2N_{s_{n-1}} + 2 \\ &= 2(2N_{s_{n-2}} + 2) + 2 \\ &= 2^2 N_{s_{n-2}} + 2^2 + 2 \\ &\vdots \\ &= 2^{n-1} N_{s_1} + 2^{n-1} + 2^{n-2} + \cdots + 2^2 + 2 \\ &= 2^{n-1} + 2^{n-2} + \cdots + 2^2 + 2 = 2^n - 2 \end{aligned}$$

2.2

We have following equations:

$$\begin{aligned}
Q(s_n, a_1) &= 1 + \frac{\gamma}{2}(Q(s_n, a_1) + Q(s_n, a_2)) \\
&= 1 + \gamma * \frac{1}{2} + \gamma^2 * \frac{1}{2} + \cdots \\
&= 1 + \frac{\gamma}{2(1-\gamma)} \\
Q(s_n, a_2) &= \frac{\gamma}{2}(Q(s_n, a_1) + Q(s_n, a_2)) \\
&= \frac{\gamma}{2(1-\gamma)} \\
Q(s_i, a_1) &= \frac{\gamma}{2}(Q(s_{i+1}, a_1) + Q(s_{i+1}, a_2)) \\
Q(s_i, a_2) &= \frac{\gamma}{2}(Q(s_1, a_1) + Q(s_1, a_2))
\end{aligned}$$

Iteratively substitute from step n to step 1, we have,

$$\begin{aligned}
Q(s_{n-1}, a_1) &= \frac{\gamma}{2}(1 + \frac{\gamma}{2(1-\gamma)} + \frac{\gamma}{2(1-\gamma)}) \\
&= \frac{\gamma}{2}(1 + \frac{\gamma}{1-\gamma}) = \frac{\gamma}{2(1-\gamma)} \\
Q(s_{n-1}, a_2) &= \frac{\gamma}{2}(Q(s_1, a_1) + Q(s_1, a_2)) \\
Q(s_{n-2}, a_1) &= \frac{\gamma}{2}(\frac{\gamma}{2(1-\gamma)} + \frac{\gamma}{2}(Q(s_1, a_1) + Q(s_1, a_2))) \\
&= (\frac{\gamma}{2})^2 \frac{1}{(1-\gamma)} + (\frac{\gamma}{2})^2((Q(s_1, a_1) + Q(s_1, a_2))) \\
Q(s_{n-2}, a_2) &= \frac{\gamma}{2}(Q(s_1, a_1) + Q(s_1, a_2)) \\
Q(s_{n-3}, a_1) &= (\frac{\gamma}{2})^3 \frac{1}{(1-\gamma)} + (\frac{\gamma}{2})^3((Q(s_1, a_1) + Q(s_1, a_2))) + (\frac{\gamma}{2})^2((Q(s_1, a_1) + Q(s_1, a_2))) \\
Q(s_{n-3}, a_2) &= \frac{\gamma}{2}(Q(s_1, a_1) + Q(s_1, a_2)) \\
&\vdots \\
Q(s_i, a_1) &= (\frac{\gamma}{2})^{n-i} \frac{1}{(1-\gamma)} + (\frac{\gamma}{2})^{n-i}(Q(s_1, a_1) + Q(s_1, a_2)) + \cdots + (\frac{\gamma}{2})^2((Q(s_1, a_1) + Q(s_1, a_2))) \\
&= (\frac{\gamma}{2})^{n-i} \frac{1}{(1-\gamma)} + (\frac{\gamma(1 - (\frac{\gamma}{2})^{n-1-i})}{2-\gamma})(Q(s_1, a_1) + Q(s_1, a_2))
\end{aligned}$$

We can then derive the similar equation for $Q_(s_1, a_1)$:

$$Q(s_1, a_1) = (\frac{\gamma}{2})^{n-1}\frac{1}{(1-\gamma)} + (\frac{\gamma(1-(\frac{\gamma}{2})^{n-2})}{2-\gamma})(Q(s_1, a_1) + Q(s_1, a_2))$$

Since we know $Q(s_1, a_2) = \frac{\gamma}{2}(Q(s_1, a_1) + Q(s_1, a_2))$, we can solve for $Q(s_1, a_1)$:

$$\begin{aligned} Q(s_1, a_1) &= (\frac{\gamma}{2})^{n-1}\frac{1}{(1-\gamma)} + (\frac{\gamma(1-(\frac{\gamma}{2})^{n-2})}{2-\gamma})(Q(s_1, a_1) + Q(s_1, a_2)) \\ &= (\frac{\gamma}{2})^{n-1}\frac{1}{(1-\gamma)} + (\frac{\gamma(1-(\frac{\gamma}{2})^{n-2})}{2-\gamma})(\frac{2}{2-\gamma})Q(s_1, a_1) \end{aligned}$$

The solution to the above equation is:

$$Q(s_1, a_1) = (\frac{\gamma}{2})^{n}\frac{1}{(1-\gamma)}\frac{1}{(1-\frac{2\gamma(1-(\frac{\gamma}{2})^{n-2})}{(2-\gamma)^2})}$$

Thus, we can write out the closed form solution for all state:

$$\begin{aligned} &Q(s_1, a_1) = (\frac{\gamma}{2})^{n}\frac{1}{(1-\gamma)}\frac{1}{(1-\frac{2\gamma(1-(\frac{\gamma}{2})^{n-2})}{(2-\gamma)^2})} \\ &Q(s_1, a_2) = \frac{\gamma}{2-\gamma}Q(s_1, a_1) \\ &Q(s_i, a_1) = (\frac{\gamma}{2})^{n-i}\frac{1}{(1-\gamma)} + (\frac{\gamma(1-(\frac{\gamma}{2})^{n-1-i})}{2-\gamma})(Q(s_1, a_1) + Q(s_1, a_2)) \\ &Q(s_i, a_2) = \frac{\gamma}{2}(Q(s_1, a_1) + Q(s_1, a_2)) \\ &Q(s_n, a_1) = 1 + \frac{\gamma}{2(1-\gamma)} \\ &Q(s_n, a_2) = \frac{\gamma}{2(1-\gamma)} \end{aligned}$$

2.3

k-1 steps. From the results above, we can easily see $Q(s_n, a_1) > Q(s_n, a_2)$ and $Q(s_1, a_1) > Q(s_1, a_2)$. For states between 1 and n, let's consider the difference between $Q(s_i, a_1)$ and

$Q(s_i, a_2)$:

$$\begin{aligned} Q(s_i,a_1) - Q(s_i,a_2) &= (\frac{\gamma}{2})^{n-i}\frac{1}{(1-\gamma)} + (\frac{\gamma(1-(\frac{\gamma}{2})^{n-1-i})}{2-\gamma} - \frac{\gamma}{2})(Q(s_1,a_1)+Q(s_1,a_2)) \\ &= (\frac{\gamma}{2})^{n-i}\frac{1}{(1-\gamma)} + (\frac{\gamma(1-(\frac{\gamma}{2})^{n-1-i})}{2-\gamma} - \frac{\gamma}{2})(Q(s_1,a_1)+Q(s_1,a_2)) \\ &\geq 0 \end{aligned}$$

3 Q-value Initialization

3.1

As the same as question 2.1. Since all Q values are initialized to the same values for both policies, they will choose random actions at each state. The expected number of steps to reach the final state is $2^n - 2$.

3.2

Since all Q values are initialized to 0 for policy 1, so the Q values will remain 0 before reaching the final state, which again downgrades to random policy. The expected number of steps to reach the final state is still $2^n - 2$.

3.3

Still $2^n - 2$. Because we didn't update any Q values for states before final state.

3.4

If we have replay buffer that stores previous transitions, for example (s_{n-1}, a_1, s_n), after updating $Q(s_n, a_1)$, we can immediately update $Q(s_{n-1}, a_1)$ using the stored transition from replay buffer.

3.5

n-1 steps. At each state, the agent randomly takes a_1 to move, and the updated Q values won't affect the agent's decision on future states. In this case, the minimum steps is $n - 1$.

4 Policy Gradient

4.1

$$\begin{aligned}
\nabla J(\theta) &= \nabla \mathbb{E}_{\tau\sim\pi_\theta}[R(\tau)] \\
&= \mathbb{E}_{\tau\sim\pi_\theta}[R(\tau)\nabla \log \pi_\theta(\tau)] \\
&= \mathbb{E}_{\tau\sim\pi_\theta}[R(\tau)\nabla \log d_0(s_0)\prod_{i=0}^{T-1}\pi_\theta(a_i|s_i)p(s_{i+1}|s_i,a_i)\pi(a_T|s_T)] \\
&= \mathbb{E}_{\tau\sim\pi_\theta}[R(\tau)\nabla \sum_{i=0}^{T-1}\log \pi_\theta(a_i|s_i)]
\end{aligned}$$

4.2

$$\begin{aligned}
\nabla J(\theta) &= \nabla \mathbb{E}_{s\sim d_0}[V^{\pi_\theta}(s)] \\
&= \mathbb{E}_{s\sim d_0}[\nabla V^{\pi_\theta}(s)] \\
&= \mathbb{E}_{s\sim d_0}\nabla \sum_a \pi_\theta(a|s)Q^{\pi_\theta}(s,a) \\
&= \mathbb{E}_{s\sim d_0}\sum_a(\nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a)+\pi_\theta(a|s)\nabla Q^{\pi_\theta}(s,a))
\end{aligned}$$

where we konw that $Q^{\pi_\theta}(s,a) = R(s,a) + \gamma\sum_{s'} p(s'|s,a)V^{\pi_\theta}(s')$ and $R(s,a)$ is independent of θ, we have

$$\begin{aligned}
\nabla J(\theta) &= \mathbb{E}_{s\sim d_0}\sum_a(\nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a)+\pi_\theta(a|s)\gamma\sum_{s'}p(s'|s,a)\nabla V^{\pi_\theta}(s')) \\
&= \mathbb{E}_{\substack{s\sim d_0\\ a\sim\pi_\theta}}\nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a)+\mathbb{E}_{s\sim d_0}\pi_\theta(a|s)\gamma\sum_{s'}p(s'|s,a)\nabla V^{\pi_\theta}(s') \\
&= \mathbb{E}_{\substack{s\sim d_0\\ a\sim\pi_\theta}}\nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a)+\mathbb{E}_{\substack{s'\sim d_1\\ a\sim\pi_\theta}}\gamma\nabla V^{\pi_\theta}(s')
\end{aligned}$$

Then, we iteratively substitute

$$\mathbb{E}_{\substack{s'\sim d_i\\ a\sim\pi_\theta}}\nabla V^{\pi_\theta}(s') = \mathbb{E}_{\substack{s'\sim d_i\\ a\sim\pi_\theta}}\nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a)+\mathbb{E}_{\substack{s'\sim d_{1+1}\\ a\sim\pi_\theta}}\gamma\nabla V^{\pi_\theta}(s') \tag{1}$$

into the equation. We get,

$$\begin{aligned}
\nabla J(\theta) &= \mathbb{E}_{\substack{s\sim d_0\\ a\sim\pi_\theta}} \nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a) + \mathbb{E}_{\substack{s\sim d_1\\ a\sim\pi_\theta}} \gamma\nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a) + \cdots \\
&\qquad + \mathbb{E}_{\substack{s\sim d_i\\ a\sim\pi_\theta}} \gamma^i\nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a) + \cdots \\
&= \sum_s \sum_{i=0}^{\infty} \gamma^i d_i \sum_a \pi_\theta(a|s)\nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a) \\
&= \mathbb{E}_{\substack{a\sim\pi_\theta\\ s\sim d}} \nabla\pi_\theta(a|s)Q^{\pi_\theta}(s,a)
\end{aligned}$$

4.3

We only need to show that $\mathbb{E}_{\substack{s\sim d\\ a\sim\pi_\theta}}[f(s)\nabla_\theta \log\pi_\theta(a|s)] = 0$.

$$\begin{aligned}
\mathbb{E}_{\substack{s\sim d\\ a\sim\pi_\theta}}[f(s)\nabla_\theta \log\pi_\theta(a|s)] &= \sum_s d^\pi(s)f(s)\mathbb{E}_{a\sim\pi_\theta}\frac{1}{\pi_\theta(a|s)}\nabla\pi_\theta(a|s) \\
&= \sum_s d^\pi(s)f(s)\sum_a \pi_\theta(a|s) * \frac{1}{\pi_\theta(a|s)}\nabla\pi_\theta(a|s) \\
&= \sum_s d^\pi(s)f(s)\nabla\sum_a \pi_\theta(a|s) \\
&= \sum_s d^\pi(s)f(s)\nabla 1 \\
&= 0
\end{aligned}$$

5 Coding: Tabular Q-learning

5.1

1. States correspond to observations in gym environments. In the case of "Taxi-v3", there are 500 discrete states. The action space is the set of actions that the agent can take for each state. Here, the action space is a discrete set of 6 actions.

2. env.step() returns observation, reward, terminated, info and done. env.reset() returns observation and info. Each state is represented by a tuple: taxi_row, taxi_col, passenger_location, destination, where each element is an integer.

3. There are 4 render modes: 'human', 'rgb_array', 'rgb_array_list' and 'ansi'. 'human' mode returns None and is only used for human display. 'rgb_array' mode returns an numpy array representing the frame of the current state. 'rgb_array_list' mode returns a list of numpy arrays representing the frames since last reset. 'ansi' mode returns a terminal-style text representation of the current state.

4. They use self.decode(self.s) to decode the current state into terminal texts.

5.2

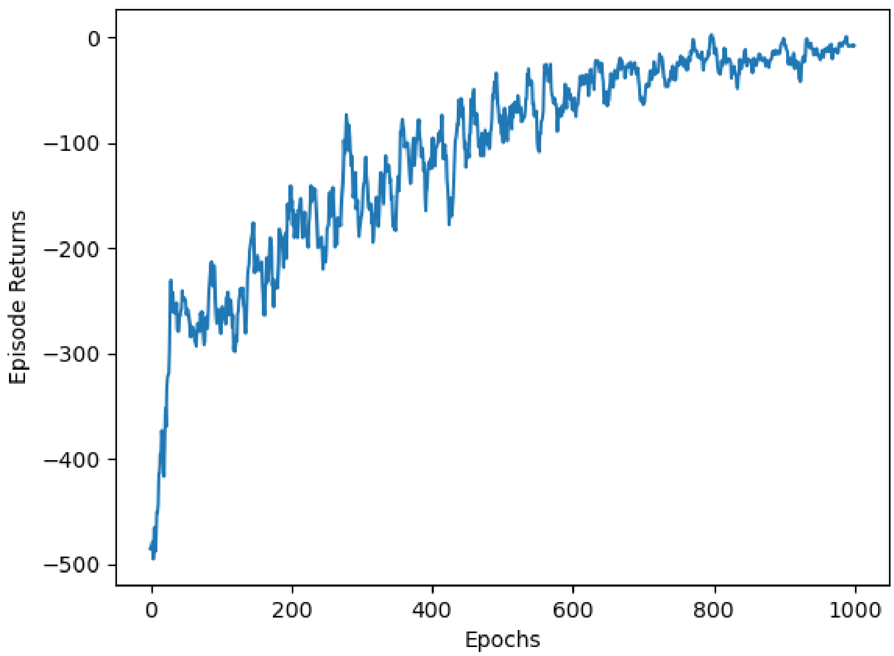

Figure 1: 5.2 train rewards

5.3

The success rate became very low (only 0.0427). Because the agent now is hard to learn the optimal Q values given the episodes. Thus, it is also hard to find the optimal policy.

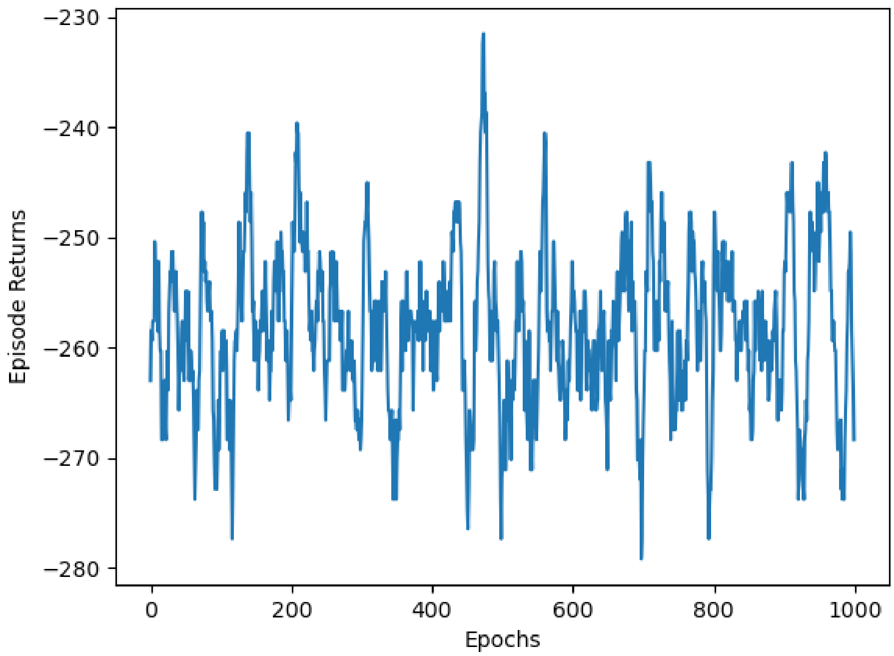

Figure 2: 5.3 train rewards

5.4

0.0427.

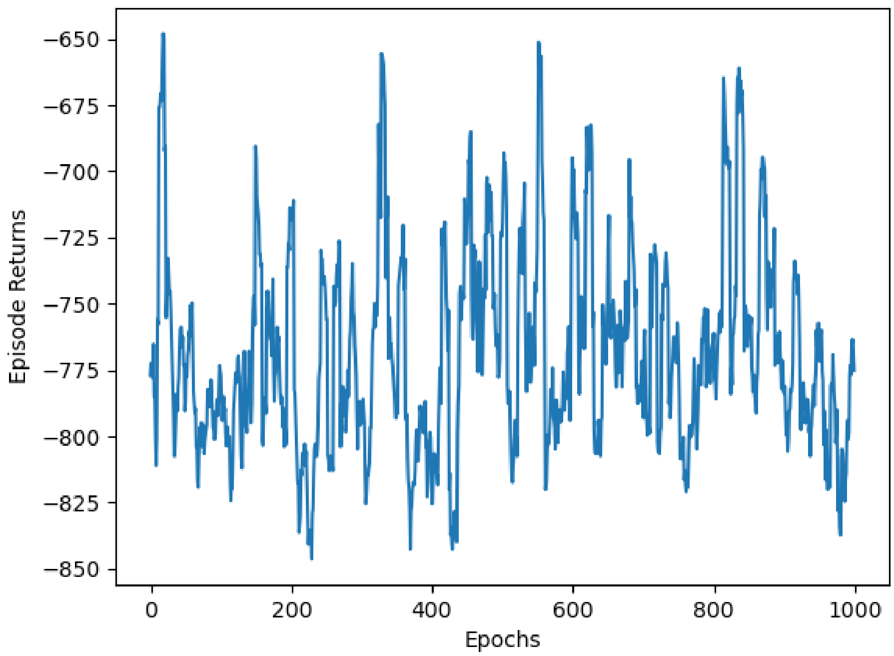

Figure 3: 5.4 train rewards

5.5

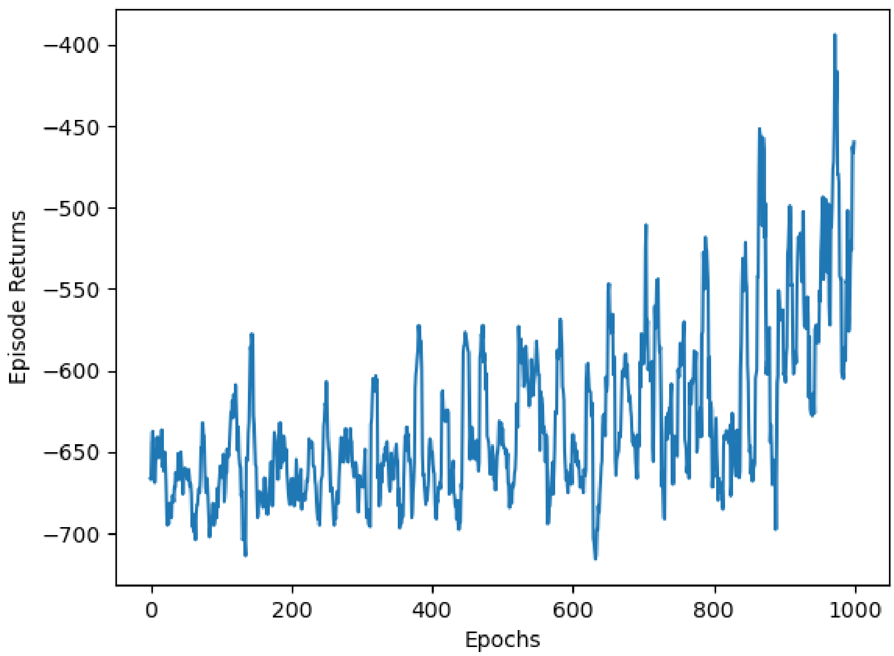

Figure 4: 5.5 train rewards

5.6

0.139.

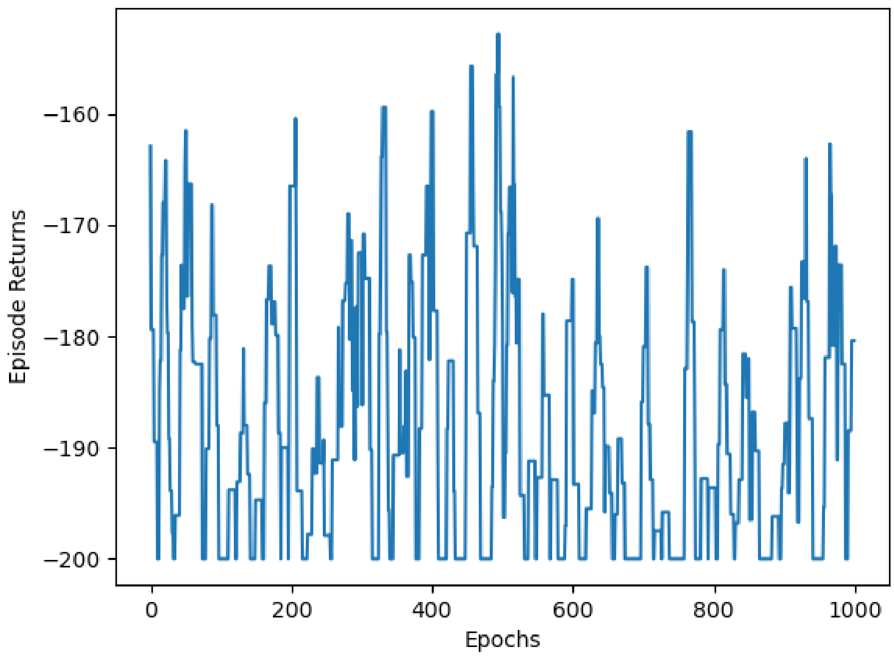

Figure 5: 5.6 train rewards

5.7

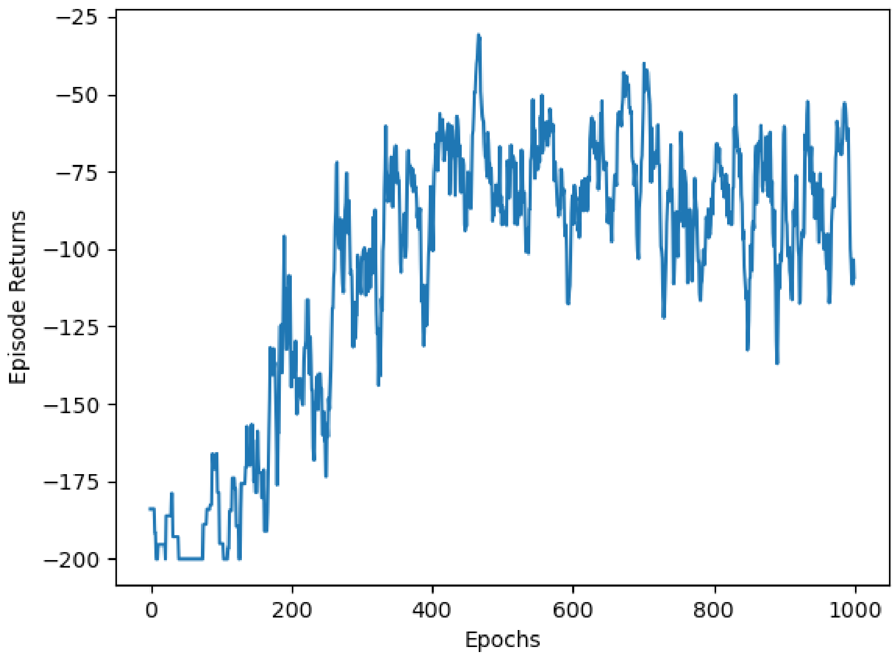

Figure 6: 5.7 train rewards