CS285 Homework 1 Imitation Learning

Chi Zhang

February 9th, 2025

1 1.1

Show that $\sum_{s_t} |p_{\pi_\theta}(s_t) - p_{\pi^*}(st)| \leq 2T\epsilon$
Answer:

$$E_{p_{\pi^*(s)}} \pi_\theta(a \neq \pi^*(s)|s) = \frac{1}{T} \sum_{t=1}^{T} \sum_{s_t} p_{\pi_\theta}(s_t) - p_{\pi^*}(s_t) \leq \epsilon \tag{1}$$

$$\sum_{t=1}^{T} \sum_{s_t} p_{\pi_\theta}(s_t) - p_{\pi^*}(s_t) \leq T\epsilon \tag{2}$$

$$\sum_{s_t} |p_{\pi_\theta}(s_t) - p_{\pi^*}(s_t)| \leq 2 \sum_{s_t} p_{\pi_\theta}(s_t) - p_{\pi^*}(s_t) \leq 2T\epsilon \tag{3}$$

2 1.2

$J(\pi) = \sum_{t=1}^{T} E_{p_\pi(s_t)} r(s_t)$
(a) Show that $J(\pi^*) - J(\pi_\theta) = O(T\epsilon)$ when the reward only depends on the last state

$$J(\pi^*) - J(\pi_\theta) = \sum_{t=1}^{T} E_{p^*_\pi(s_t)} r(s_t) - \sum_{t=1}^{T} E_{p_{\pi_\theta}(s_t)} r(s_t) \tag{4}$$

$$= E_{p^*_\pi(s_T)} r(s_T) - E_{p_{\pi_\theta}(s_T)} r(s_T) \tag{5}$$

(b) Show that $J(\pi^*) - J(\pi_\theta) = O(T^2\epsilon)$ for an arbitrary reward

3 Behavior Cloning

Number	ep len	num agent	n iter	batch size	eval batch size	train batch size	lr
1	1000	1000	1	1000	1000	100	5e-3
2	1000	1000	1	1000	5000	100	5e-3
3	1000	5000	1	1000	5000	100	5e-3
4	1000	10000	1	1000	5000	100	5e-3
5	1000	1000	1	5000	5000	100	5e-3
6	1000	1000	1	1000	5000	100	1e-3
7	1000	1000	1	1000	5000	100	5e-4
8	1000	10000	1	1000	5000	100	5e-4

Table 1: BC Experiment Setup

Takeaways
1. The num gradient steps per iter bigger, the training loss is lower, but the average return is lower and std is bigger.
2. The batch size doesn't affect the final eval return.
3. If we only use less learning rate the average episode length will be small and the return will be bad. But if we give the agent more steps to learn the policy, which means using bigger number of gradient steps for training policy, the agent can learn the policy better and get higher return.

Number	section	avg return	std return	max return	min return	avg ep len	loss	time
1	eval	384.130	0	384.130	384.130	1000.0	NA	NA
1	train	4681.891	30.708	4712.600	4651.183	1000.0	0.0366	
2	eval	371.347	9.033	384.130	356.267	1000	NA	NA
2	train	4681.891	30.708	4712.600	4651.183	1000.0	0.036	4.134
3	eval	318.537	7.386	328.486	309.464	1000.0	NA	NA
3	train	4681.891	30.708	4712.600	4651.183	1000.0	0.00136	6.6885
4	eval	257.724	54.693	366.503	225.117	1000.0	NA	NA
4	train	4681.891	30.708	4712.600	4651.183	1000.0	0.000138	9.953
5	eval	371.347	9.033	384.130	356.267	1000.0	NA	NA
5	train	4681.891	30.708	4712.600	4651.183	1000.0	0.0366	3.901
6	eval	-154.174	188.305	6.604	-583.376	333.0	NA	NA
6	train	4681.891	30.708	4712.600	4651.183	1000.0	0.2862	4.1165
7	eval	-230.911	348.572	-14.636	-1350.314	195.3	NA	NA
7	train	4681.891	30.708	4712.600	4651.183	1000.0	0.5081	4.675
8	eval	575.628	5.646	585.100	568.259	1000.0	NA	NA
8	train	4681.891	30.708	4712.600	4651.183	1000.0	0.0070	21.2002

Table 2: BC Experiment Setup

4 DAGGER

Number	ep len	num agent	n iter	batch size	eval batch size	train batch size	lr
1	1000	1000	10	1000	1000	100	5e-3
2	1000	1000	10	1000	5000	100	5e-3
3	1000	1000	2	1000	5000	100	5e-3
4	1000	1000	3	1000	5000	100	5e-3
5	1000	1000	5	1000	5000	100	5e-3
6	1000	1000	15	1000	5000	100	5e-3
7	1000	1000	20	1000	5000	100	5e-3

Table 3: DAGGER Experiment Setup

Number	section	avg return	std return	max return	min return	avg ep len	loss	time
1	eval	4773.100	0	4773.100	4773.100	1000.0	NA	NA
1	train	4776.946	0	4776.946	4776.946	1000.0	-2.436	
2	eval	4549.032	84.578	4678.968	4420.138	1000.0	NA	NA
2	train	4674.956	0	4674.956	4674.956	1000.0	-2.5253	23.698
3	eval	4685.809	166.117	4887.419	4455.141	1000.0	NA	NA
3	train	4604.381	0	4604.381	4604.381	1000.0	-2.302	4.444
4	eval	4669.874	115.880	4874.916	4522.125	1000.0	NA	NA
4	train	4850.563	0	4850.563	4850.563	1000.0	-2.5386	6.757
5	eval	4668.167	96.770	4845.206	4563.481	1000.0	NA	NA
5	train	4782.043	0	4782.043	4782.043	1000.0	-2.497	11.696
6	eval	4755.949	68.399	4862.198	4678.494	1000.0	NA	NA
6	train	4656.931	0	4656.931	4656.931	1000.0	-2.653	36.729
7	eval	4452.156	343.086	4646.300	3766.879	1000.0	NA	NA
7	train	4954.585	0	4954.585	4954.585	1000.0	-2.6589	49.136

Table 4: DAGGER Experiment Result

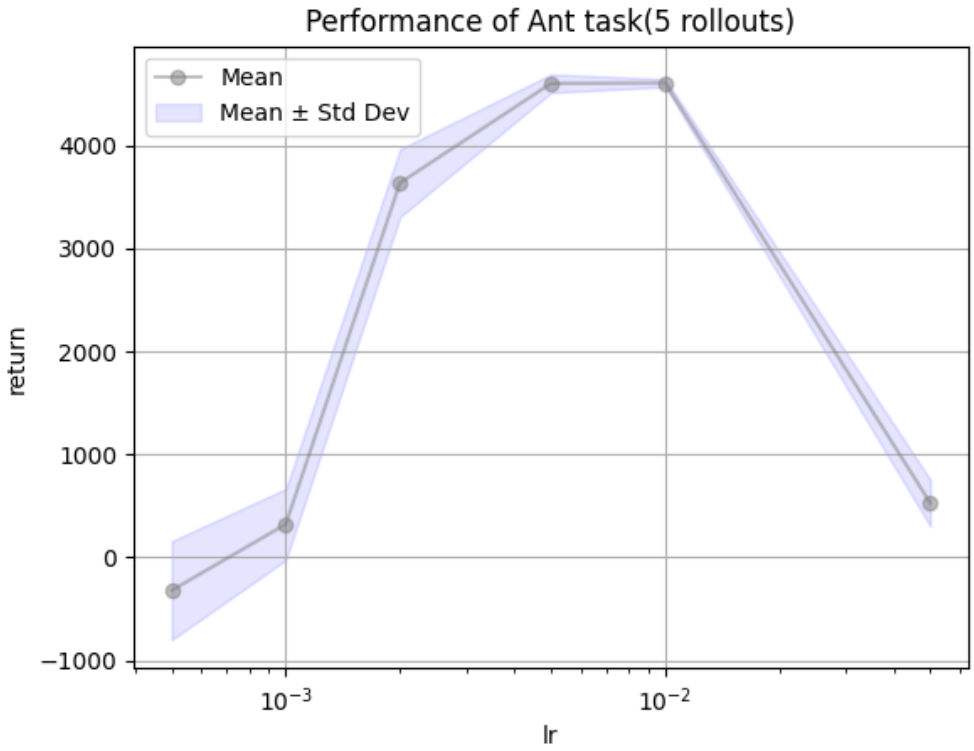

Figure 1: BC: Return vs lr with 5 rollouts in ant task

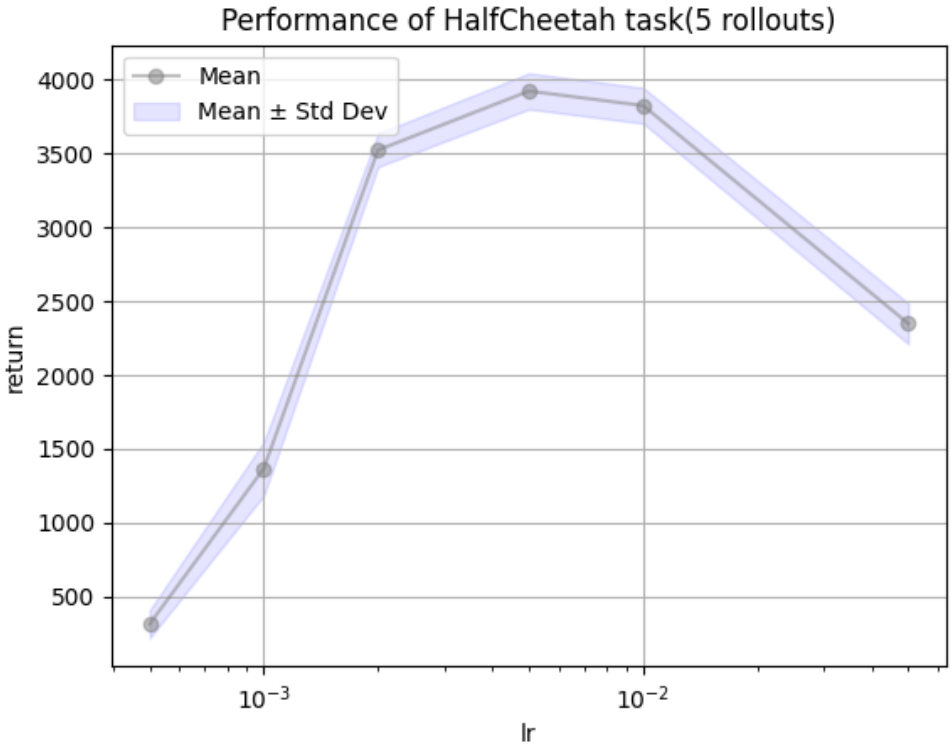

Figure 2: BC: Return vs lr with 5 rollouts in halfcheetah task

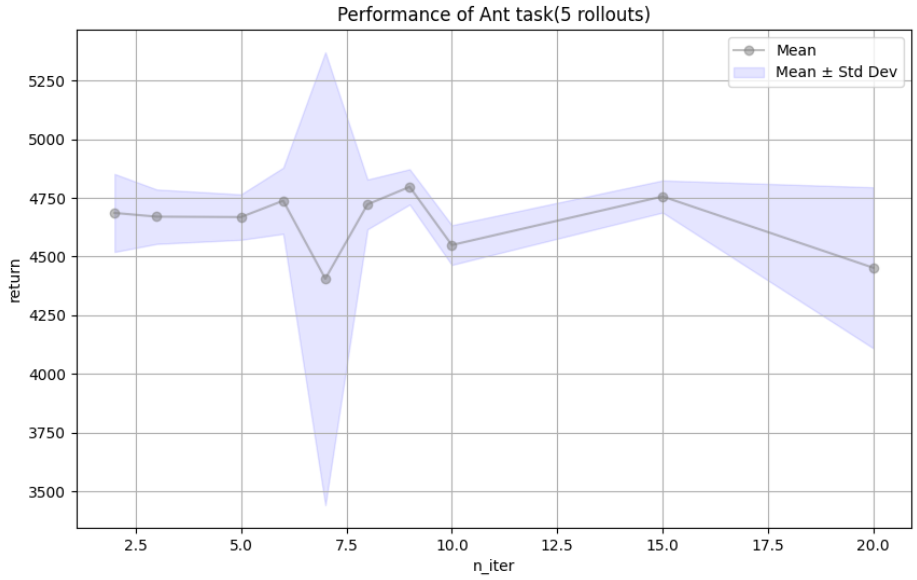

Figure 3: DAGGER: Return vs n iter with 5 rollouts in ant task

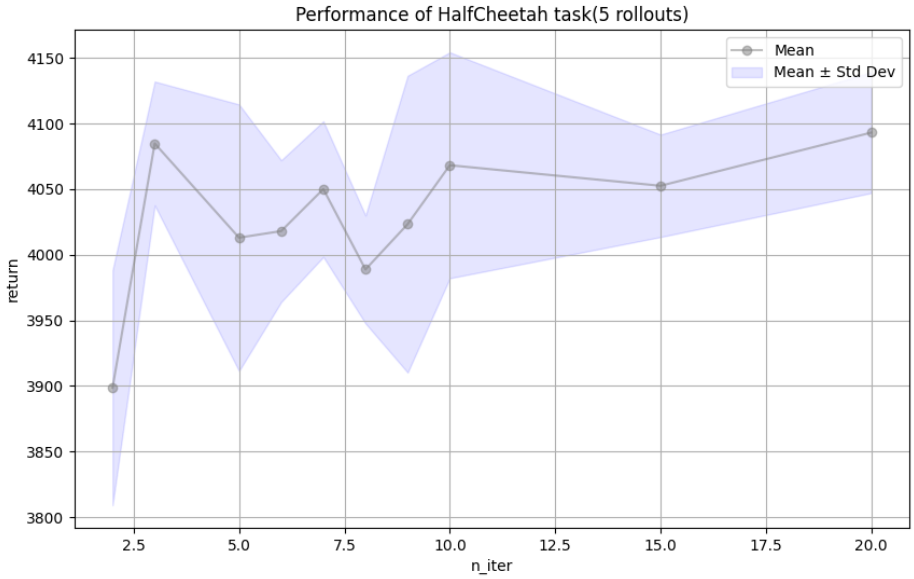

Figure 4: DAGGER: Return vs n iter with 5 rollouts in halfcheetah task