**CS285 Assignment 1**

Cuiqianhe Du

1.2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Common parameters:  n\_iter: 1,  eval\_batch\_size: 5000,  ep\_len: 1000 | **Ant** | Train\_AverageReturn: 4713.65 Train\_StdReturn: 12.20 | **Hopper** | Train\_AverageReturn: 3772.67 Train\_StdReturn: 1.95 | Comment |
|  | Eval\_AverageReturn | Eval\_StdReturn | Eval\_AverageReturn | Eval\_StdReturn |  |
| Train 1: No extra parameter | 814.95 | 153.86 | 442.28 | 254.17 |  |
| Train 2:  num\_agent\_train\_steps\_per\_iter: 2000 | 4103.46 | 224.28 | 835.09 | 216.8 | changed number of steps per train iteration from 1k to 2k, which mean it has double training effort that leads to better performance |
| Train 3:  num\_agent\_train\_steps\_per\_iter: 2000,  learning\_rate: 1e-2 | 4211.88 | 323.93 | 1056.1 | 215.51 | increased learning rate from 5e-3 to 1e-2, thus increasing the speed of learning |

1.3 I changed the number of training steps per iteration as it will provide more training and have a positive correlation with the policy performance. How Ant’s performance varies with the change of number of agent training steps per iter:

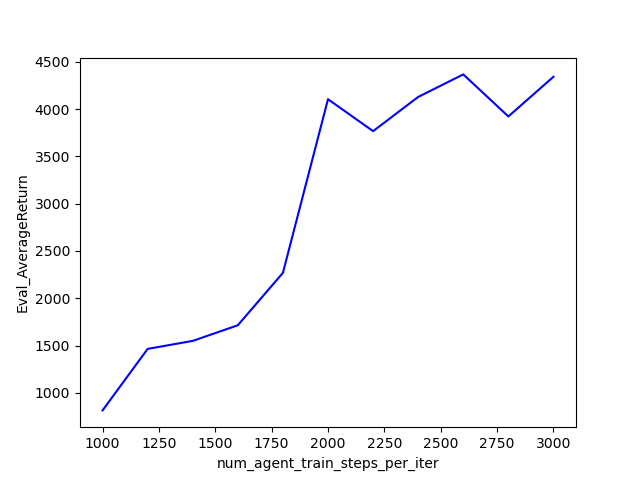
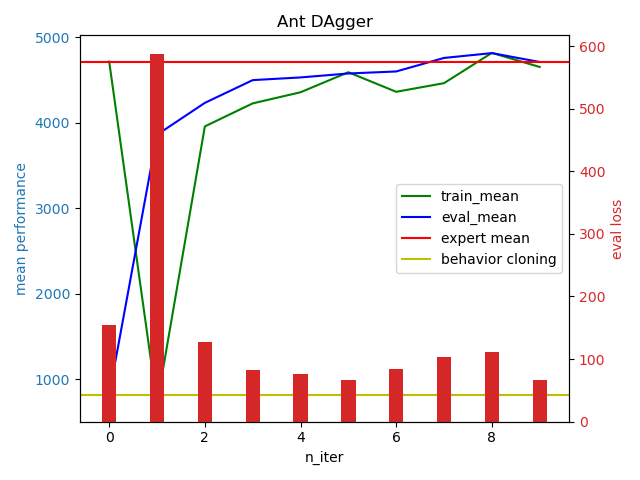


Figure : Ant behavior cloning

2.2 Dagger performance based on number of iterations with following parameter:

n\_iter: 10, eval\_batch\_size: 5000, ep\_len: 1000, all other parameters are default

**Ant:**



**Hopper:**

