1 Analysis

텍스트, 친필, 문서, 종이이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

텍스트, 친필, 종이, 종이 제품이(가) 표시된 사진

AI 생성 콘텐츠는 정확하지 않을 수 있습니다.

3 Behavioral Cloning

1. Run behavioral cloning (BC) and report results on two tasks: one where a behavioral cloning agent should achieve at least 30% of the performance of the expert, and one environment of your choosing where it does not.



|  |
| --- |
| python cs285/scripts/run\_hw1.py --expert\_policy\_file cs285/policies/experts/Ant.pkl --env\_name Ant-v4 --exp\_name bc\_ant\_video --n\_iter 1 --expert\_data cs285/expert\_data/expert\_data\_Ant-v4.pkl --ep\_len 1000 --eval\_batch\_size 5000 --video\_log\_freq 1 --n\_layers 2 --size 64 |

* Eval\_AverageReturn : 1552.6072998046875
* Eval\_StdReturn : 292.062255859375
* Eval\_MaxReturn : 2147.45068359375
* Eval\_MinReturn : 1278.1318359375
* Eval\_AverageEpLen : 896.1666666666666
* Train\_AverageReturn : 4681.891673935816
* Train\_StdReturn : 30.70862278765526
* Train\_MaxReturn : 4712.600296723471
* Train\_MinReturn : 4651.18305114816
* Train\_AverageEpLen : 1000.0
* Training Loss : 0.03868953138589859
* Train\_EnvstepsSoFar : 0
* TimeSinceStart : 30.320274114608765
* Initial\_DataCollection\_AverageReturn : 4681.891673935816



|  |
| --- |
| python cs285/scripts/run\_hw1.py --expert\_policy\_file cs285/policies/experts/Hopper.pkl --env\_name Hopper-v4 --exp\_name bc\_hopper\_video --n\_iter 1 --expert\_data cs285/expert\_data/expert\_data\_Hopper-v4.pkl --ep\_len 1000 --eval\_batch\_size 5000 --video\_log\_freq 1 --n\_layers 2 --size 64 |

* Eval\_AverageReturn : 1006.6049194335938
* Eval\_StdReturn : 315.4906005859375
* Eval\_MaxReturn : 1974.660400390625
* Eval\_MinReturn : 405.4320068359375
* Eval\_AverageEpLen : 302.2352941176471
* Train\_AverageReturn : 3717.5129936182307
* Train\_StdReturn : 0.3530361779417035
* Train\_MaxReturn : 3717.8660297961724
* Train\_MinReturn : 3717.159957440289
* Train\_AverageEpLen : 1000.0
* Training Loss : 0.03620936721563339
* Train\_EnvstepsSoFar : 0
* TimeSinceStart : 11.099523305892944
* Initial\_DataCollection\_AverageReturn : 3717.5129936182307

|  |
| --- |
| Eval\_AverageReturn |
| 텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  AI 생성 콘텐츠는 정확하지 않을 수 있습니다. |
| Eval\_StdReturn |
| 텍스트, 스크린샷, 멀티미디어 소프트웨어, 소프트웨어이(가) 표시된 사진  AI 생성 콘텐츠는 정확하지 않을 수 있습니다. |

2. Experiment with one set of hyperparameters that affects the performance of the behavioral cloning agent, such as the amount of training steps, the amount of expert data provided, or something that you come up with yourself. For one of the tasks used in the previous question, show a graph of how the BC agent’s performance varies with the value of this hyperparameter. In the caption for the graph, state the hyperparameter and a brief rationale for why you chose it.



|  |
| --- |
| python cs285/scripts/run\_hw1.py --expert\_policy\_file cs285/policies/experts/Ant.pkl --env\_name Ant-v4 --exp\_name bc\_ant\_video --n\_iter 1 --expert\_data cs285/expert\_data/expert\_data\_Ant-v4.pkl --ep\_len 1000 --eval\_batch\_size 5000 --video\_log\_freq 1 --n\_layers 2 --size 64 |

* Eval\_AverageReturn : 1552.6072998046875
* Eval\_StdReturn : 292.062255859375
* Eval\_MaxReturn : 2147.45068359375
* Eval\_MinReturn : 1278.1318359375
* Eval\_AverageEpLen : 896.1666666666666
* Train\_AverageReturn : 4681.891673935816
* Train\_StdReturn : 30.70862278765526
* Train\_MaxReturn : 4712.600296723471
* Train\_MinReturn : 4651.18305114816
* Train\_AverageEpLen : 1000.0
* Training Loss : 0.03868953138589859
* Train\_EnvstepsSoFar : 0
* TimeSinceStart : 30.320274114608765
* Initial\_DataCollection\_AverageReturn : 4681.891673935816



|  |
| --- |
| … --n\_layers 4 --size 128 |

* Eval\_AverageReturn : 1277.498291015625
* Eval\_StdReturn : 362.19354248046875
* Eval\_MaxReturn : 1998.316650390625
* Eval\_MinReturn : 870.1470336914062
* Eval\_AverageEpLen : 998.5
* Train\_AverageReturn : 4681.891673935816
* Train\_StdReturn : 30.70862278765526
* Train\_MaxReturn : 4712.600296723471
* Train\_MinReturn : 4651.18305114816
* Train\_AverageEpLen : 1000.0
* Training Loss : 0.03484756872057915
* Train\_EnvstepsSoFar : 0
* TimeSinceStart : 33.58611559867859
* Initial\_DataCollection\_AverageReturn : 4681.891673935816



|  |
| --- |
| … --n\_layers 1 --size 32 |

* Eval\_AverageReturn : 612.7852783203125
* Eval\_StdReturn : 90.59205627441406
* Eval\_MaxReturn : 790.5887451171875
* Eval\_MinReturn : 546.1014404296875
* Eval\_AverageEpLen : 1000.0
* Train\_AverageReturn : 4681.891673935816
* Train\_StdReturn : 30.70862278765526
* Train\_MaxReturn : 4712.600296723471
* Train\_MinReturn : 4651.18305114816
* Train\_AverageEpLen : 1000.0
* Training Loss : 0.03733008727431297
* Train\_EnvstepsSoFar : 0
* TimeSinceStart : 28.81749987602234
* Initial\_DataCollection\_AverageReturn : 4681.891673935816

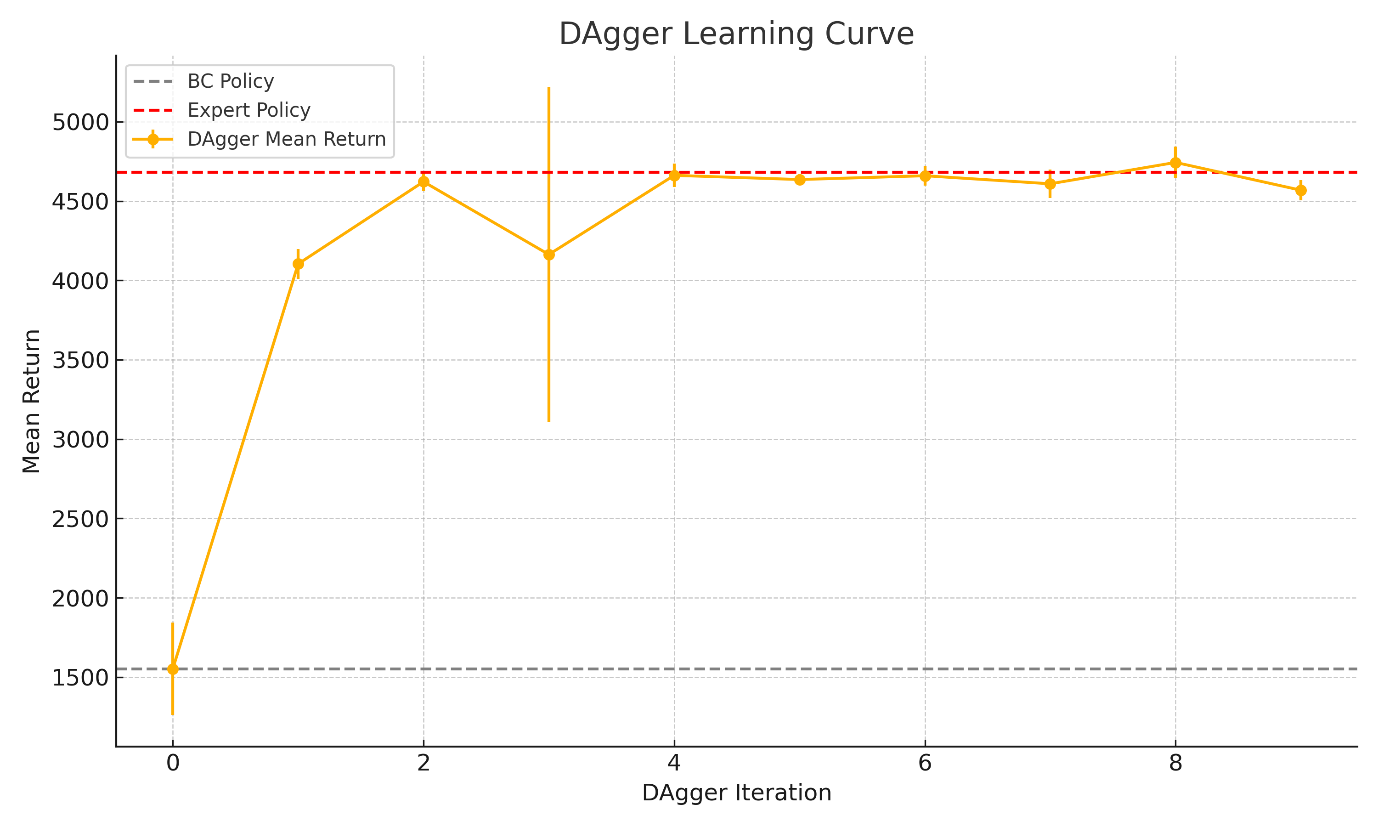
|  |
| --- |
| Eval\_AverageReturn |
| 텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진  AI 생성 콘텐츠는 정확하지 않을 수 있습니다. |
| Eval\_StdReturn |
| 텍스트, 스크린샷, 멀티미디어 소프트웨어, 소프트웨어이(가) 표시된 사진  AI 생성 콘텐츠는 정확하지 않을 수 있습니다. |

4 Dagger

1. Using the same code, you should be able to run DAgger by modifying the runtime parameters as follows:

|  |
| --- |
| python cs285/scripts/run\_hw1.py --expert\_policy\_file cs285/policies/experts/Ant.pkl --env\_name Ant-v4 --exp\_name dagger\_ant\_video --n\_iter 10 --do\_dagger --expert\_data cs285/expert\_data/expert\_data\_Ant-v4.pkl --ep\_len 1000 --eval\_batch\_size 5000 --video\_log\_freq 1 --n\_layers 2 --size 64 |

2. Run DAgger and report results on the two tasks you tested previously with behavioral cloning. Report your results in the form of a learning curve, plotting the number of DAgger iterations vs. the policy’s mean return, with error bars to show the standard deviation. Include the performance of the expert policy and the behavioral cloning agent on the same plot (as horizontal lines that go across the plot). In the caption, state which task you used, and any details regarding network architecture, amount of data, etc. (as in the previous section).



**Environment (Task)**: Ant-v4

**Expert Policy File**: Ant.pkl (used to imitate expert behavior)

**Learning Method**: DAgger (Dataset Aggregation)

* Executed with the --do\_dagger flag

**Number of Iterations**: --n\_iter 10 → DAgger was run for 10 iterations

**Episode Length**: --ep\_len 1000 → Each episode consists of 1000 steps

**Policy Network Architecture**:

* Number of hidden layers: --n\_layers 2
* Size of each hidden layer: --size 64

**Evaluation Batch Size**: --eval\_batch\_size 5000 → Each evaluation is performed using 5000 steps

5 Discussion

1. How much time did you spend on each part of this assignment.

|  |  |
| --- | --- |
| Analysis | 1h |
| Editing Code | 2h |
| Behavioral Cloning | 2h |
| DAgger | 2h |

2. Any additional feedback?