Homework9. Reinforcement Gym

Jae Dong Hwang

QLearningzCartPole

Run the starting point code 10 times with your implementation (learn the policy then evaluate it as in the starting point). Hand in a table of the scores you achieved on each of the 10 iterations as well as the average. [Many runs should score 200.0]

Ran 10 times the script and aggregate the total rewards and majority of average rewards were 200.0.

runNumber	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Trial 6	Trial 7	Trial 8	Trial 9	Trial 10
0	200.0	200.0	200.0	200.0	200.0	200.0	200.0	200.0	200.0	200.0
1	200.0	186.0	200.0	200.0	200.0	200.0	200.0	200.0	128.0	200.0
2	200.0	200.0	200.0	200.0	200.0	200.0	197.0	200.0	200.0	200.0
3	200.0	200.0	200.0	200.0	200.0	189.0	200.0	200.0	200.0	200.0
4	200.0	200.0	200.0	200.0	200.0	200.0	200.0	200.0	125.0	200.0
5	200.0	200.0	200.0	200.0	200.0	200.0	200.0	200.0	123.0	200.0
6	200.0	200.0	200.0	200.0	200.0	200.0	200.0	200.0	168.0	200.0
7	200.0	200.0	200.0	200.0	200.0	175.0	200.0	200.0	119.0	200.0
8	200.0	200.0	200.0	200.0	200.0	182.0	200.0	200.0	129.0	200.0
9	200.0	200.0	200.0	200.0	200.0	200.0	200.0	200.0	200.0	200.0
Average	200.0	198.6	200.0	200.0	200.0	194.6	199.7	200.0	159.2	200.0

QLearningMountainCar

Tune the following 4 parameters:

- discountRate
- actionProbabilityBase
- trainingIterations
- Assignment7Support.mountainCarBinsPerDimension

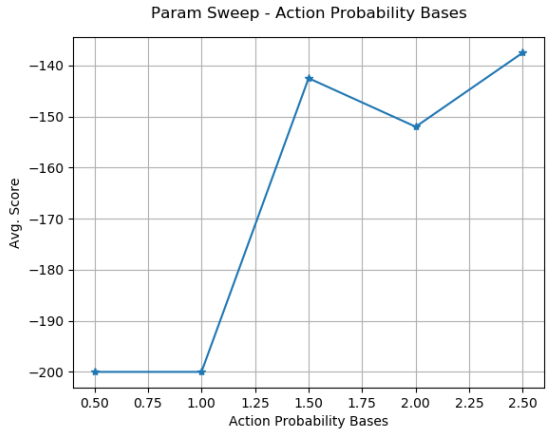
For each produce a chart with at least 5 settings of the parameter value on the x-axis and the average score across 10 policy learning runs on the y axis (10x learn a policy then evaluate it as the sample code does, get the average score). Consider the properties of this problem and use your understanding to guide which regions you explore.

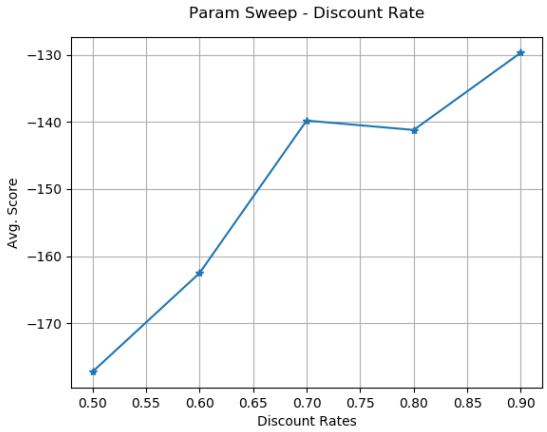
- Average Score for Parameter Sweeps

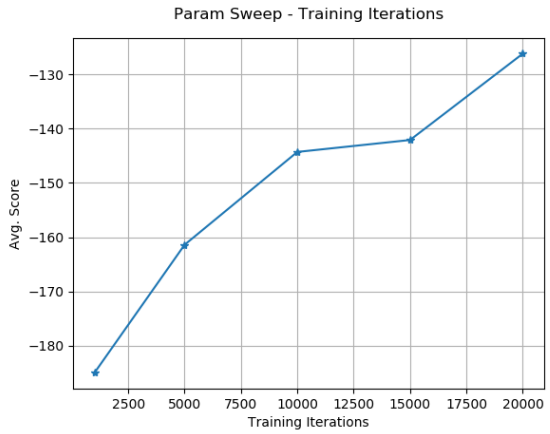
  I ran the start script with each parameter options and collected the average score.

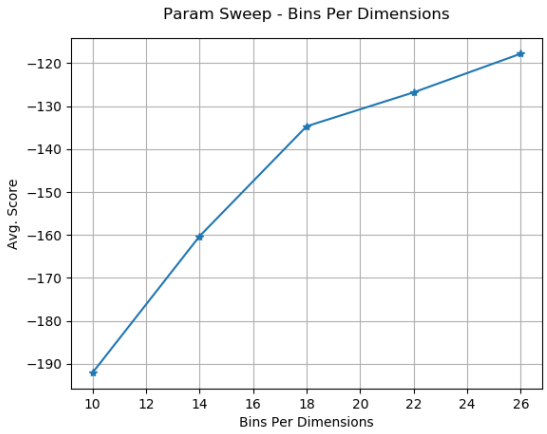
```
discountRates = [0.5, 0.6, 0.7, 0.8, 0.9]
actionProbabilityBases = [0.5, 1.0, 1.5, 2.0, 2.5]
```

```
trainingIterations = [1000, 5000, 10000, 15000, 20000]
BinsPerDimensions = [10, 14, 18, 22, 26]
```









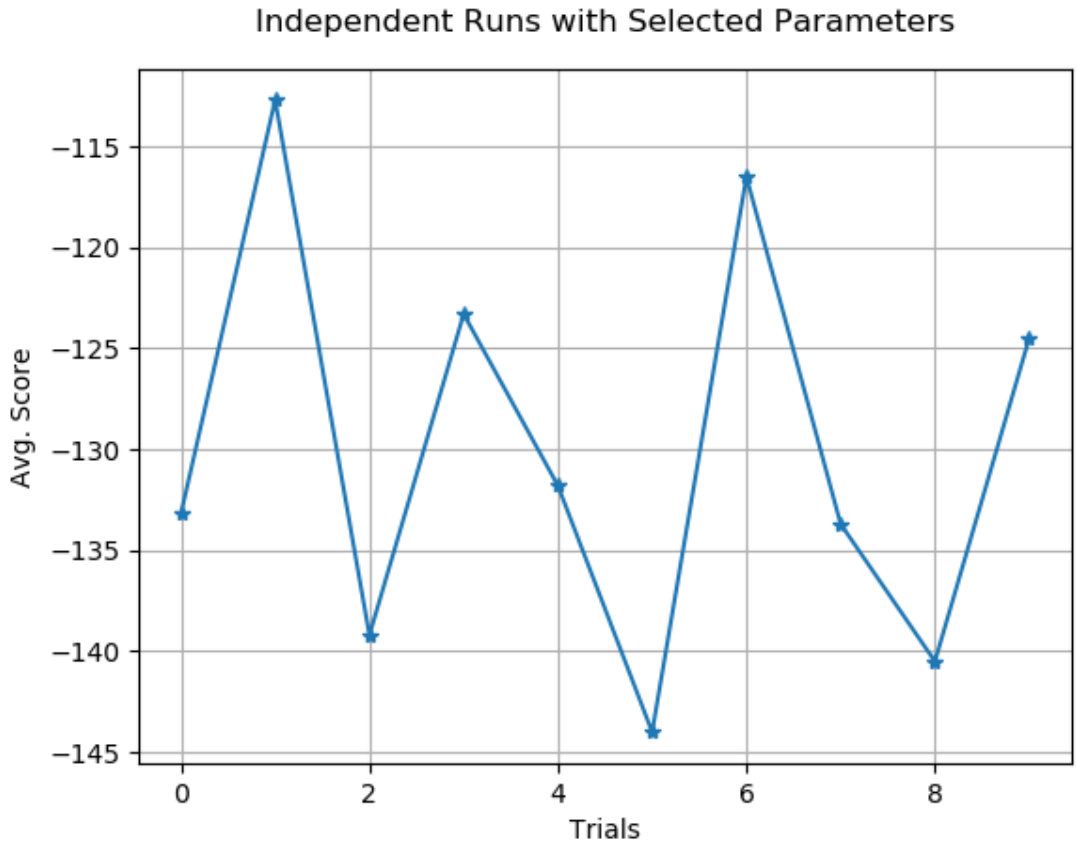
- Action Probability Base (k) value less than and equals to 1.0 doesn't improve policy to control the carts (-200). It means it doesn't have high Q values enough to make better the range. Increasing discount rates within the chosen range, 0.5 - 0.9, get closer to the optimal action. And training iteration also helped to converge to the optimal control space. Similarly, more state space helped to reach the best results.

Produce an improved parameter setting. You may change the 4 you tuned and you may change any other that you think matters (and do additional tuning).

- Improved parameter setting: Given the results of paramter sweep data, I picked following configuration.

```
discountRate = 0.9
actionProbabilityBase = 2.5
trainingIteration = 20000
BinsPerDimension = 26
```

And here is the results from running 10 times policy learning.



runNumber	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Trial 6	Trial 7	Trial 8	Trial 9	Trial 10
0	-150.0	-85.0	-147.0	-89.0	-155.0	-163.0	-138.0	-151.0	-136.0	-145.0
1	-137.0	-86.0	-146.0	-134.0	-138.0	-147.0	-97.0	-143.0	-153.0	-138.0
2	-152.0	-105.0	-93.0	-142.0	-138.0	-155.0	-147.0	-140.0	-136.0	-147.0
3	-104.0	-148.0	-144.0	-108.0	-135.0	-163.0	-94.0	-151.0	-149.0	-89.0
4	-104.0	-90.0	-143.0	-107.0	-130.0	-111.0	-106.0	-142.0	-142.0	-139.0
5	-104.0	-146.0	-145.0	-110.0	-136.0	-144.0	-137.0	-88.0	-152.0	-91.0
6	-142.0	-106.0	-145.0	-107.0	-136.0	-129.0	-139.0	-149.0	-109.0	-137.0
7	-150.0	-104.0	-142.0	-138.0	-105.0	-143.0	-94.0	-143.0	-138.0	-88.0
8	-152.0	-106.0	-143.0	-161.0	-106.0	-139.0	-107.0	-85.0	-154.0	-135.0
9	-137.0	-151.0	-144.0	-137.0	-139.0	-146.0	-106.0	-145.0	-136.0	-136.0
Average	-133.2	-112.7	-139.2	-123.3	-131.8	-144.0	-116.5	-133.7	-140.5	-124.5

Total average: = -129.94