ROB537: HW3

Ammar Kothari

1 Introduction

Reinforcement learning is the ability to learn from the enviornment. With a reward signal, an agent can figure out what actions are best to achieve the reward. The key insight is to evaluate the value of a given conditions. In this assignment, associating values with only an action and associating values with a state action pair are examined. The agent trained with Q-learning which uses state action pairs achieves near optimal performance on the grid world domain. The action value learner which only uses actions achieves good peformance on the N-armed bandit problem.

2 Bandit Problem

A 10 step and 100 step version of the Bandit Problem were tested. The same total number of steps was taken, 10,000 steps, but the episode resets occured depnding on the step length of an episode. The solution quality for the 100 step agent is similar to the 10 step problem.

2.1 Problem Description

The goal for the bandit problem is to maximize the expected return given a set of slot machines that give a reward. In this test, the rewards have a gaussian distrubution and is different for each slot machine. The number of steps is the number of steps allowed before the situation is reset. For the Bandit Problem, the initial state is with the accumulated reward as zero.

2.2 Results

An action value learning agent is used in both cases. Figure 1 and Figure 2 shows the expected values as estimated by the learning for both episode lengths with two different exploration amounts along with the actual distribution of the learners. Table 1 shows statistics of the reward obtained by the agents.

2.3 Analysis

All version struggle to approximate the true distribution. The solutions that they find are within variation of each other indicating that there is no improvement by increasing the length of the episodes. The solution will closer approximate the true distribution as the number of samples increases. With increased exploration, the true values are generally approximated better. For action 2 and no exploration, the agent appears to be close to the true value while the many of the other estimates are not close to approximating the true distribution.

3 Grid World

For grid world, two different types of learners are tested. The first only estimates the expected reward of actions. The second estimates the rewards based on state action pairs. The performance of the state action pair is better.

Steps	Exploration	Reward	StDev
10	0.2	0.914	1.24
100	0.2	1.49	0.58
10	0.0	1.14	0.94
100	0.0	0.93	1.87

Table 1: Results for different tests for Bandit Problem with an Action Value learner

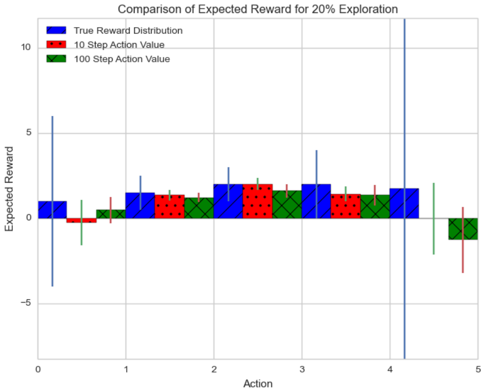

Figure 1: Comparison of Action Value tables between selection for 10 steps and 100 steps for an action-value learner with 20% epsilon greedy exploration on a multiarmed bandit problem.

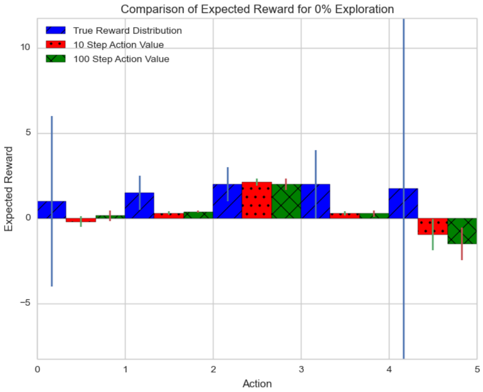

Figure 2: Comparison of Action Value tables between selection for 10 steps and 100 steps for an action-value learner with 0% epsilon greedy exploration on a multiarmed bandit problem.

3.1 Problem Description

In Grid World, the agent is spawned at a random location in the world. The goal is a location on the map. Every location that is not the goal gives a reward of -1 and the goal gives a reward of 100. The length of an episode is 100 steps. At the end of an episode, the agent is randomly spawned on the map.

3.2 Results

The learning progression is shown in Figure 3. The Q-Learning agent performs better than the Action Value agent. An example trajector after training is completed is shown in Figure 4 and Figure 5. In both situations, the episode length is 20 steps. A random action is choosen randomly 20% of the time. The learning rate is 0.2, and the agents were stopped after 10,000 iterations. However, convergence occurs within the first hundred iterations. For both scenarios, ten trials were conducted with results shown including results from all ten trial runs.

3.3 Analysis

Action Value learning basically just walks around randomly. Although moving down and the right seems to be the best move if the agent does not know where it is relative to the reward, the agent does not learn this. This may be because the sparseness of the reward is not enough to overcome all of the small negative rewards. The result is a random walk through the space, often staying in the same spot. This is shown by the example trajectories in Figure 5. All of the action values are almost -1 which corresponds to the reward for all squares that do not hold the reward.

Q learning performs much better. The trade off is that significantly more information must be stored compared to the Action Value learner. State action pairs are saved which requires an array 50 times larger than the action value table. However, the Q-value agent is able to navigate to the goal optimally because of associating state actions with rewards. This requires more iterations to converge, but allows the agent to understand the layout of the rewards as well as the best actions to achieve that reward.

4 Conclusion

In general, Q learning is the more effective agent. If the problem has only one state, an Action Value learning, which is a single state case of Q learning, will be able to solve the problem. For simple grid world problems, Q-learning is able to find the optimal path from any location on the map to the reward.

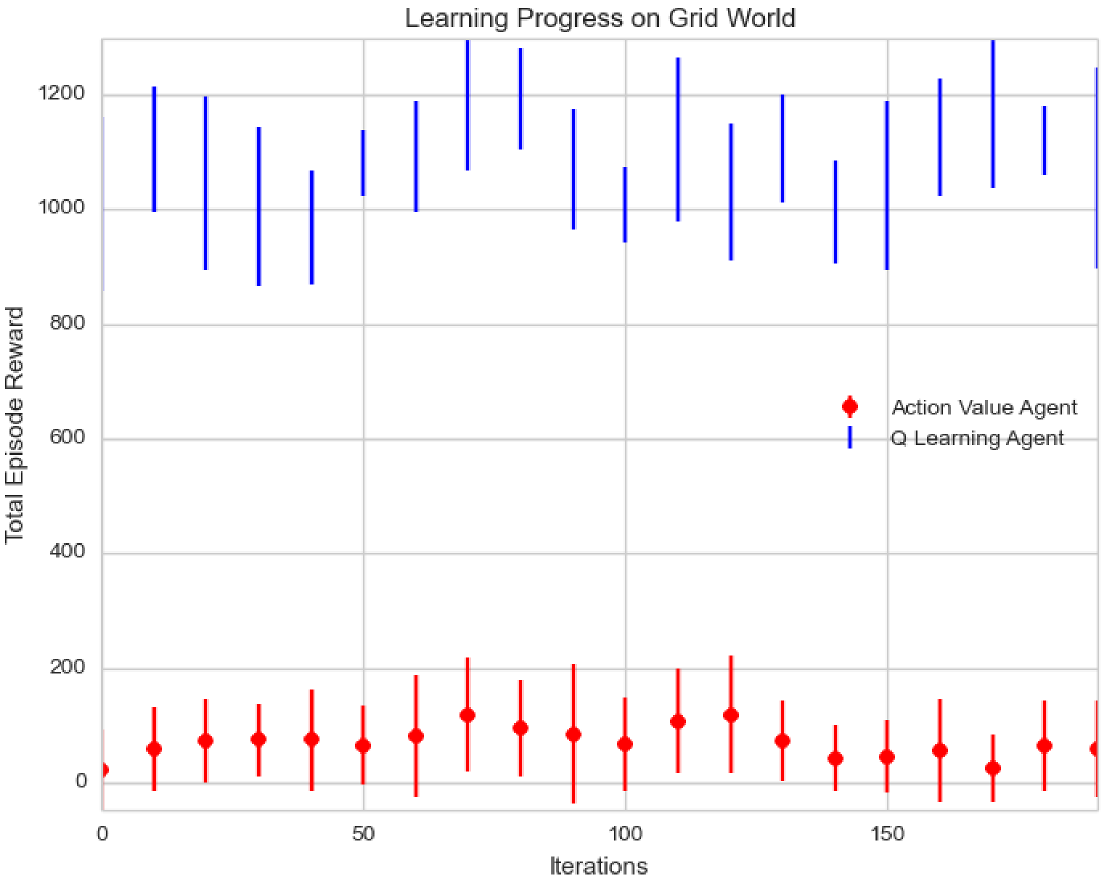

Figure 3: Learning progress of both agents on grid world. Q learning agent is able to improve over the course of training, where as the Action Value agent never improves significantly.

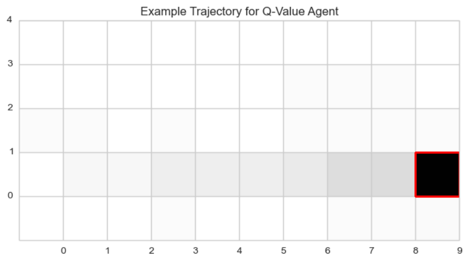

Figure 4: Converged Q-Value learning agent trajectories. Darker colors indicate a square that is visited most often. The test episodes only last 10 steps. The agent learns the most direct route to the highest reward.

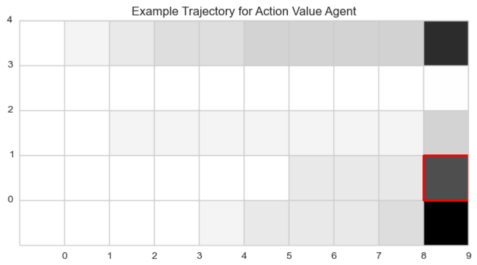

Figure 5: Converged Action Value learning agent trajectories. Darker colors indicate a square that is visited most often. The test episodes only last 10 steps. The agent learns to go down and right regardless of location.