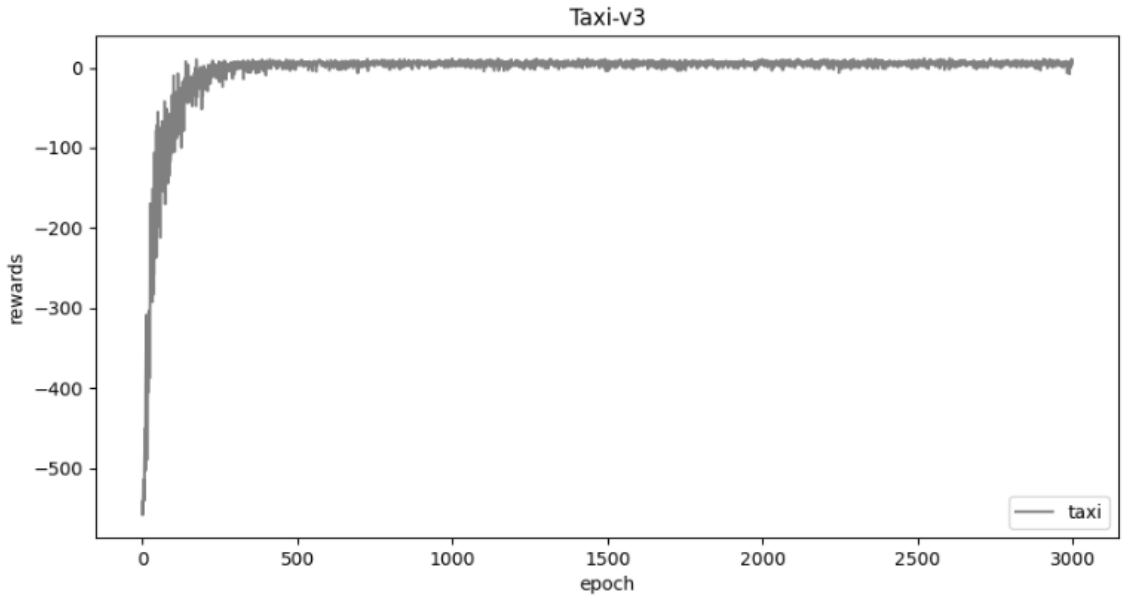
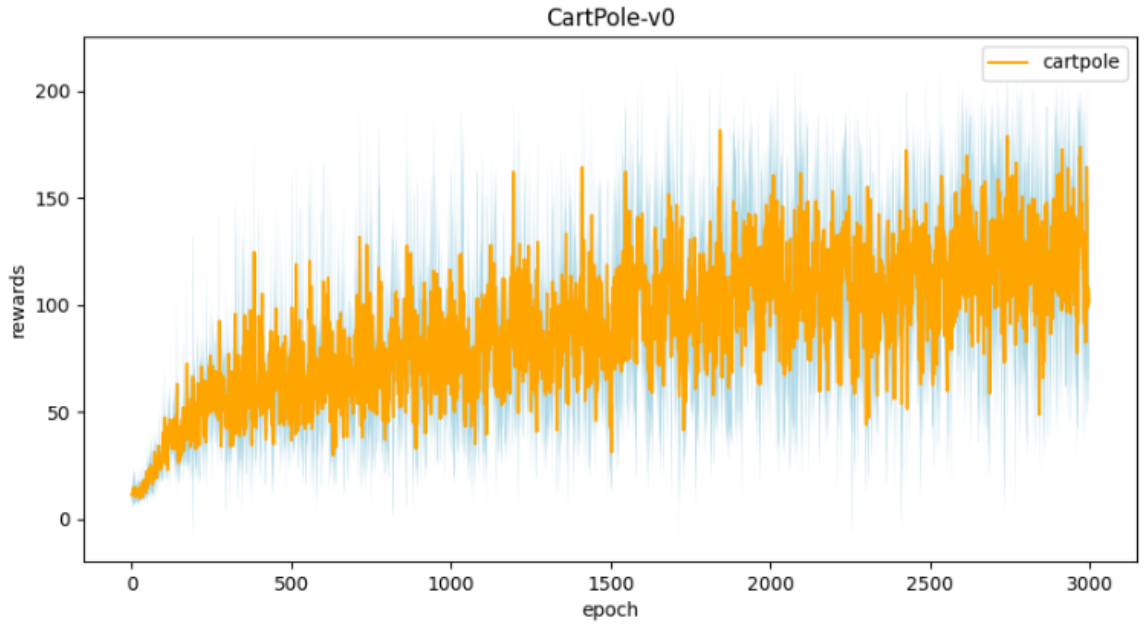
# Homework 4: Reinforcement Learning Report

## Part I. Experiment Results (the score here is included in your implementation):

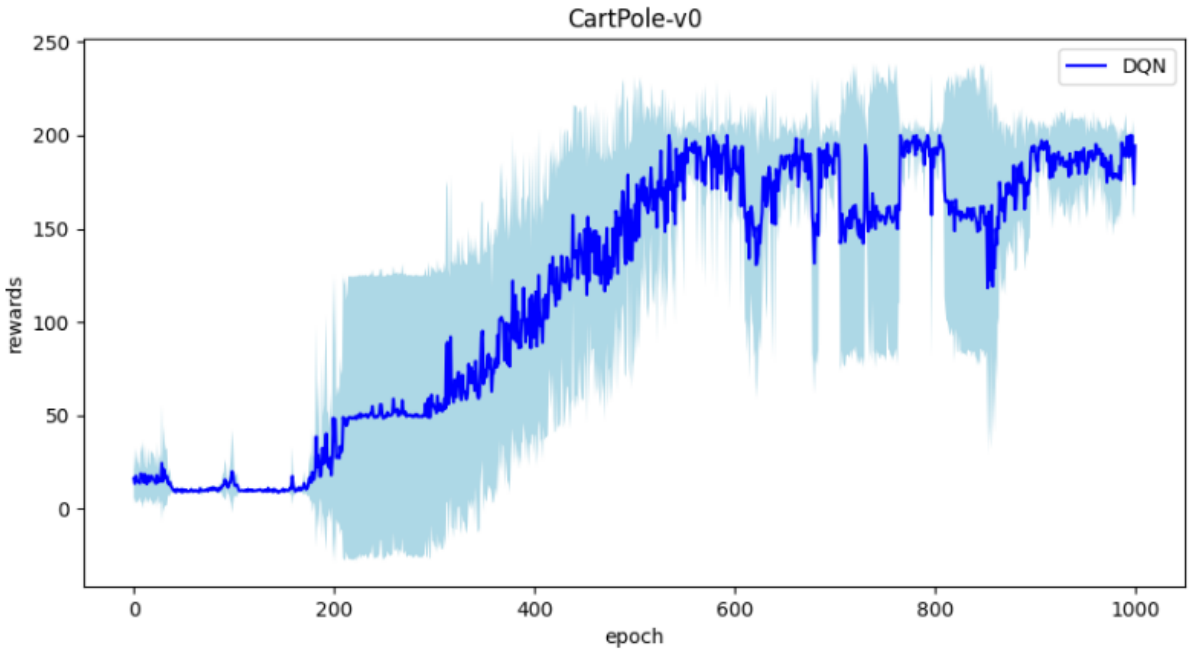
### 1. taxi.png:



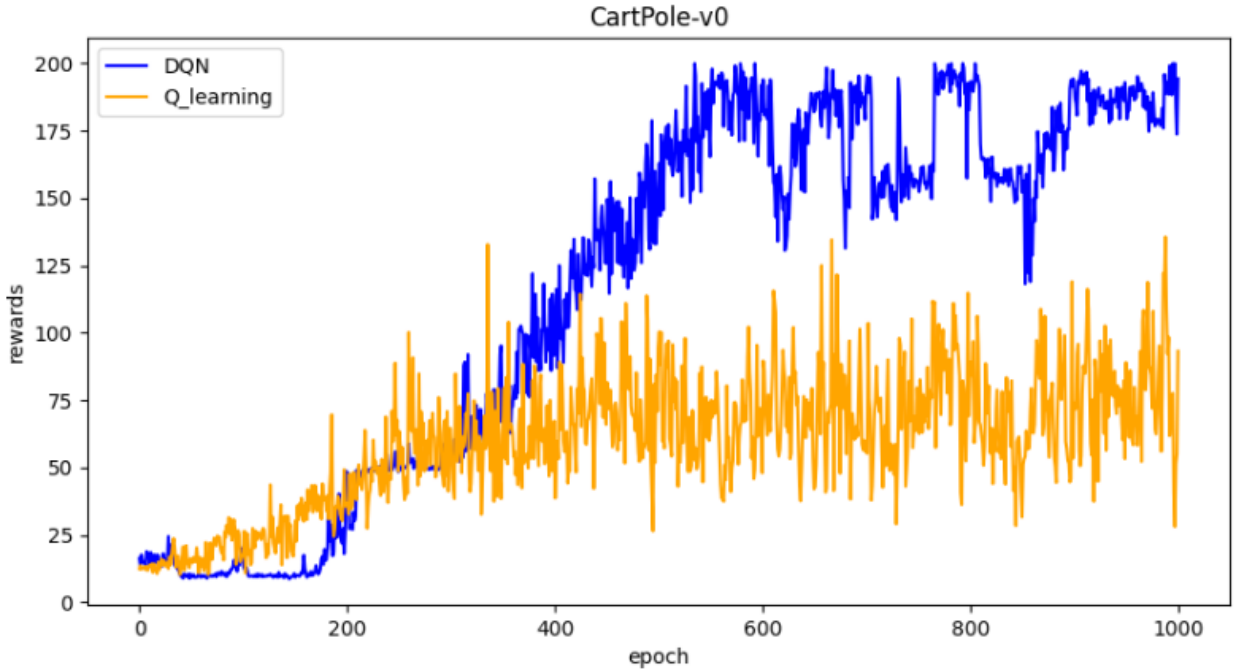

### 2. cartpole.png



## 3. DQN.png




## 4. compare.png



# Part II. Question Answering (50%):

1. **Calculate the optimal Q-value of a given state in Taxi-v3 (the state is assigned in google sheet), and compare with the Q-value you learned** (Please screenshot the result of the “check_max_Q” function to show the Q-value you learned). (4%)

```
# Begin your code
# calculate max_q with state
max_q = np.max(self.qtable[state])
# calculate optimal_q:(2, 2)-> B -> G = 11steps
power = np.power(self.gamma, 11)
optimal = (-1)*(1-power)/(1-self.gamma)+power*20
print("optimal Q:", optimal)
return max_q
# End your code
```

```
average reward: 8.41
Initail state:
taxi at (2, 2), passenger at B, destination at R
optimal Q: -0.5856821172999993
max Q:-0.5856821172999982
```

2. **Calculate the max Q-value of the initial state in CartPole-v0, and compare with the Q-value you learned.** (Please screenshot the result of the “check_max_Q” function to show the Q-value you learned) (4%)

```
# Begin your code
# calculate optimal_q: highest score is 200
power = np.power(self.gamma, 200)
optimal = (1-power)/(1-self.gamma)
print("optimal Q =", optimal)
# calculate max_q with discretize state
return np.max(self.qtable[self.discretize_observation(self.env.reset())])
# End your code
```

```
average reward: 198.7
optimal Q = 33.25795863300011
max Q:30.4686174811151
```

3. 
   a. **Why do we need to discretize the observation in Part 2?** (2%)
      Because the value of state is continuous, which may cause infinite state-action pairs.Therefore, we need to discretize it.
   b. **How do you expect the performance will be if we increase “num_bins”?** (2%)
      I think it will. because the increase of num_bins represents higher precision when we discretize the state.
   c. **Is there any concern if we increase “num_bins”?** (2%)
      It increases the time of learning since more states are created.

4. **Which model (DQN, discretized Q learning) performs better in Cartpole-v0, and what are the reasons?** (3%)
   DQN performed better compared to Q-learning, DQN requires less memory and time under numerous amounts of states.

5.
   a. **What is the purpose of using the epsilon greedy algorithm while choosing an action?** (2%)
      For exploring/enlarging more possible states that can use for training the model.
   b. **What will happen, if we don't use the epsilon greedy algorithm in the CartPole-v0 environment?** (3%)
      The model will only find the optimal within the explored space.
   c. **Is it possible to achieve the same performance without the epsilon greedy algorithm in the CartPole-v0 environment? Why or Why not?** (3%)
      It's possible to achieve, but it's a very low chance since the epsilon is very small which implies that most of the time, the program takes the max q-value. It's a matter of probability.

   d. **Why don't we need the epsilon greedy algorithm during the testing section?** (2%)
      Because we do not need the randomness to enlarge the exploring space in the test phase.
6. **Why is there "with torch.no_grad():" in the "choose_action" function in DQN?** (3%)
   It disabled gradient calculation, which will reduce memory consumption for computations if we don't need tensor.backward()

7.
   a. **Is it necessary to have two networks when implementing DQN?** (1%)
      Yes. It's crucial.
   b. **What are the advantages of having two networks?** (3%)
      It increases the stability of function approximation, giving the learning network time to converge and lose more of its initial bias.
   c. **What are the disadvantages?** (2%)
      It learns very slowly, since it needs time to converge/learn the target network.
8.
   a. **What is a replay buffer(memory)? Is it necessary to implement a replay buffer? What are the advantages of implementing a replay buffer?** (5%)
      Replay buffer is required for experience replay for storing past

experiences and then using a random subset of these experiences to update the Q-network.

b. **Why do we need batch size?** (3%)
It controls the accuracy of the estimate of the error gradient when training a neural network.

c. **Is there any effect if we adjust the size of the replay buffer(memory) or batch size? Please list some advantages and disadvantages.** (2%)
Replay buffer:
Larger buffer will lead to a more stable training process. However, it cannot be too large or too small or it will cause a significant performance drop.
Batch size:
Larger batch size will give us a smoother gradient during training. On the other hand, it also has critical value that will cause degradation.

**9.**

a. **What is the condition that you save your neural network?** (1%)
I always save my neural network on every iteration.

b. **What are the reasons?** (2%)
My computer can handle it and the program running time is not too long.

**10. What have you learned in the homework?** (2%)
Follow my heart:))
To be honest, homework 4 is much harder than any of the homework before. It takes me a lot of time and effort to understand the code, not to mention the time writing and debugging the code. However, the effect of the seed is beyond my imagination, almost like playing the lottery (I tried my best in part2 since my friend easily passed 150:)))). I hope it will not affect that much in the future's homeworks.