

Deep Q-Network (100 points)

The purpose of this homework is to learn how to play games using Deep Q-Network. In each of the games mentioned below, the objective for the agent is to solve the game (a) or maximize the total score achieved during the game (b):

(a) **CartPole-v0**. You are familiar with this game from the last assignment. Use discount factor = 0.95 for this game. The solved requirements are specified [here](#).

(b) **MsPacman-v0**. Take a look [here](#) to learn how the game is played. Train a neural net that uses images of the game as the states and models the action-value function. The discount factor is 0.99. This game has 9 actions.

You may pre-process all the images from the game with the following function before feeding them into the neural network.

```
mspacman_color = 210 + 164 + 74
def preprocess_observation(obs):
    img = obs[1:176:2, ::2] # crop and downsize
    img = img.sum(axis=2) # to greyscale
    img[img==mspacman_color] = 0 # Improve contrast
    img = (img // 3 - 128).astype(np.int8) # normalize from -128 to 127
    return img.reshape(88, 80, 1)
```

For both questions:

- (i) Plot the maximum Q-values versus the number of training episodes.
- (ii) Plot the episode rewards versus the number training episodes, overlay with the moving average of last 100 episode rewards.
- (iii) After you finish training, roll out 500 episodes using your trained neural network. Plot the histogram of the episode rewards, report the mean and standard deviation.

Your submission should include:

- (i) The code to train the network and create the plots.
 - (ii) A file with the instructions to run your program.
 - (iii) A report with the three required plots for each question.
-

Notes

- You are responsible for training your network and creating the plots by the due time. The same rule applies to the entire class with no excuses allowed. The late homework policy still applies (see the use of Joker in the syllabus).
 - You should use the specified version of both games (CartPole-v0 and MsPacman-v0).
 - Your DQN implementation must use a neural network, a replay buffer, and the notion of the target network.
 - Your implementation must be done exclusively by you without looking at code from others. Please refer to the syllabus for the complete homework policy.
 - You should use a single GPU card to run your program on deepdish.
 - Your assignment will be graded on the performance of your algorithm, including how fast your algorithm converges and the episode rewards that your trained model gains. If your DQN implementation is correct and your solution works correctly, i.e., the solved requirements for game (a) are met and your plots show that the network is learning how to play for game (b), you will get the majority of the points.
-