CptS 570 Machine Learning, Fall 2021
Homework #4

Due Date: Fri, Dec 10 midnight via Canvas

NOTE 1: Please use a word processing software (e.g., Microsoft word or Latex) to write your answers and submit a printed copy to me at the beginning of class on Oct 23. The rationale is that it is sometimes hard to read and understand the hand-written answers.

NOTE 2: Please ensure that all the graphs are appropriately labeled (x-axis, y-axis, and each curve). The caption or heading of each graph should be informative and self-contained.

1. (**3 Percent**) Implementation of Q-Learning algorithm and experimentation.

   You are given a Gridworld environment that is defined as follows:

   State space: GridWorld has 10x10 = 100 distinct states. The start state is the top left cell. The gray cells are walls and cannot be moved to.

   Actions: The agent can choose from up to 4 actions (left, right, up, down) to move around.

   Environment Dynamics: GridWorld is deterministic, leading to the same new state given each state and action

   Rewards: The agent receives +1 reward when it is in the center square (the one that shows R 1.0), and -1 reward in a few states (R -1.0 is shown for these). The state with +1.0 reward is the goal state and resets the agent back to start.

   In other words, this is a deterministic, finite Markov Decision Process (MDP). Assume the discount factor β=0.9.

   Implement the Q-learning algorithm (slide 46) to learn the Q values for each state-action pair. Assume a small fixed learning rate α=0.01.

   Experiment with different explore/exploit policies:

   1) ϵ-greedy. Try ϵ values 0.1, 0.2, and 0.3.

   2) Boltzman exploration. Start with a large temperature value T and follow a fixed scheduling rate. Give these details in your report.

   How many iterations did it take to reach convergence with different exploration policies?

   Please show the converged Q values for each state-action pair.

2. (**3 percent**) Convolutional Neural Networks (CNNs) for solving image classification task. You will train a CNN on Fashion MNIST data. The network architecture contains 4 CNN layers followed by one pooling layer and a final fully connected layer. The basic architecture (in sequential order) will be as follows:

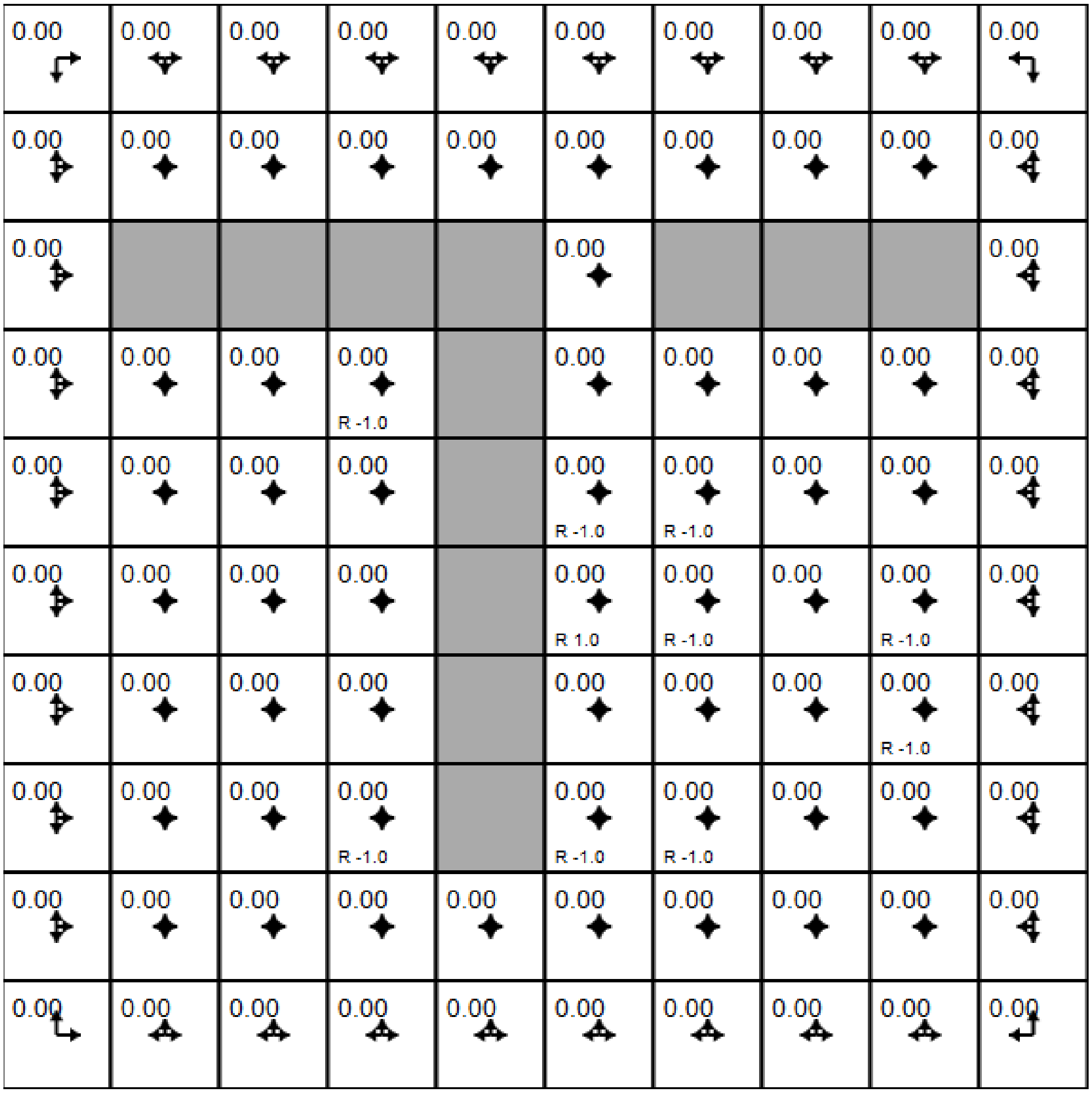
   First CNN layer: input channels - 1, output channels - 8, kernel size = 5, padding = 2, stride = 2 followed by ReLU operation

   Second CNN layer: input channels - 8, output channels - 16, kernel size = 3, padding = 1, stride = 2 followed by ReLU operation

   Third CNN layer: input channels - 16, output channels - 32, kernel size = 3, padding = 1, stride = 2 followed by ReLU operation

   Fourth CNN layer: input channels - 32, output channels - 32, kernel size = 3, padding = 1, stride = 2 followed by ReLU operation

   one "Average" pooling layer (nn.AdaptiveAvgPool2d(1) would work in PyTorch)



0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00					0.00				0.00
0.00	0.00	0.00	0.00 R -1.0		0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00		0.00 R -1.0	0.00 R -1.0	0.00	0.00	0.00
0.00	0.00	0.00	0.00		0.00 R 1.0	0.00 R -1.0	0.00	0.00 R -1.0	0.00
0.00	0.00	0.00	0.00		0.00	0.00	0.00	0.00 R -1.0	0.00
0.00	0.00	0.00	0.00 R -1.0		0.00 R -1.0	0.00 R -1.0	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Figure 1: Grid world domain with states and rewards.

Fully connected layer (nn.Linear in PyTorch) - determine the number of input features from previous CNN layers. This can be done easily by hand. The number of output features will be equal to number of classes, i.e., 10. If you want help, you can use the direct formula given on this page: `http://cs231n.github.io/convolutional-networks/`.

This will be a straightforward extension from the code discussed in the demo session. Plot the training and testing accuracy as a function of atleast 10 epochs. You could use a smaller sized dataset if compute power is a hurdle. A good choice would be 50 percent of the training set and 10 percent of the testing set. Please make sure you have equal ratio of all classes in the dataset. You can try all tips mentioned in the demo session for solving this task. Optionally, it will be a good idea to try adding other training techniques to see the maximum accuracy possible. Some of them include batch normalization, data augmentation, using other optimizers like ADAM etc.

3. **(1 Percent)** Please read the following paper and write a brief summary (at most one page) of the main points.

   D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-Franois Crespo, Dan Dennison: Hidden Technical Debt in Machine Learning Systems. NIPS 2015: 2503-2511

4. **(1 Percent)** Please read the following paper and write a brief summary (at most one page) of the main points.

   Eric Breck, Shanqing Cai, Eric Nielsen, Michael Salib, D. Sculley: The ML test score: A rubric for ML production readiness and technical debt reduction. BigData 2017: 1123-1132

Grading Rubric

Each question in the students work will be assigned a letter grade of either A,B,C,D, or F by the Instructor and TAs. This five-point (discrete) scale is described as follows:

- **A) Exemplary (=100%)**.
  Solution presented solves the problem stated correctly and meets all requirements of the problem.
  Solution is clearly presented.
  Assumptions made are reasonable and are explicitly stated in the solution.
  Solution represents an elegant and effective way to solve the problem and is not overly complicated than is necessary.

- **B) Capable (=75%)**.
  Solution is mostly correct, satisfying most of the above criteria under the exemplary category, but contains some minor pitfalls, errors/flaws or limitations.

- **C) Needs Improvement (=50%)**.
  Solution demonstrates a viable approach toward solving the problem but contains some major pitfalls, errors/flaws or limitations.

- **D) Unsatisfactory (=25%)**
  Critical elements of the solution are missing or significantly flawed.
  Solution does not demonstrate sufficient understanding of the problem and/or any reasonable directions to solve the problem.

- **F) Not attempted (=0%)**
  No solution provided.

The points on a given homework question will be equal to the percentage assigned (given by the letter grades shown above) multiplied by the maximum number of possible points worth for that question. For example, if a question is worth 6 points and the answer is awarded a B grade, then that implies 4.5 points out of 6.