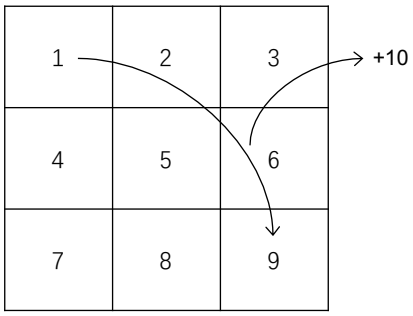
SI252 Reinforcement Learning 2021/05/11

Homework 6

Professor: Ziyu Shao *Due: 2021/06/05 11:59am*

1. **3x3 Grid World**:



The agent can be in one of the nine cells at any starting time. It can then move in one of four directions:{E,S,W,N}. If the agent hits a wall, it remains in its current cell and gets a reward -1. When the agent moves to cell 1, it then immediately moves to cell 9 and gets a reward of 10. The discount factor $\gamma = 0.9$.

   (a) Under the uniform policy (equal probabilities for each possible actions), compute the value of each state(cell).

   (b) For subproblem (a), show numerical results obtained by policy-evaluation algorithm and TD algorithm. Discuss the pros and cons of each algorithm.

   (c) Find the optimal value of each state and corresponding optimal policy.

   (d) For subproblem (c), show numerical results obtained by policy-iteration algorithm and Q-learning algorithm. Discuss the pros and cons of each algorithm.

2. **Python Implementation of REINFORCEjs**. Written by JavaScript language, REINFORCEjs is a Reinforcement Learning library that implements several common RL algorithms supported with fun web demos. The web address is: here. The source code is maintained in GitHub.

   (a) Reproduce the "GridWorld: Dynamic Programming Demo" by Python.

   (b) Reproduce the "GridWorld: Temporal Difference Learning Demo" by Python.

   (c) (**Bonus Problem**) Reproduce the "PuckWorld: DQN Demo" by Python.

   (d) (**Bonus Problem**) Reproduce the "WaterWorld: DQN Demo" by Python.

3. **Paper Summary**. To understand state-of-the-art results, it is important to read and parse research papers. If this is your first time reading a research paper, this link may be helpful. This resource may be even useful for veterans in research. It describes a multi-pass approach to reading papers. The papers that you can choose to read may vary in difficulty, and we do not expect anyone to understand all of the content in the papers. Nonetheless, it is important to start building your ability to read research papers, especially research papers out of your comfort zone. The notation, mathematics, and jargon of the paper matter less than the big ideas and context of the paper. Read papers, take notes, and answer the questions for each paper individually (see below).

   Questions for each paper:

   - What are the papers' main contributions? Describe the core idea.
   - What was surprising, difficult, or confusing in the paper?
   - Were the experiments (if there are any) convincing?
   - How can these methods be applied in ways not described in the paper? Have you seen these ideas in other papers? Feel free to cite relevant work.

   Paper list (Pick 2 out of 6 and write corresponding summaries):

   - Human-level control through deep reinforcement learning, by Mnih et al, Nature, 2015.
   - Mastering the game of Go with deep neural networks and tree search, by Silver et al, Nature 2016.
   - Mastering the game of Go without human knowledge, by Silver et al, Nature 2017.
   - A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play, by Silver et al, Science 2018.
   - DeepStack: Expert-level artificial intelligence in heads-up no-limit poker, by Moravcik et al, Science, 2017
   - Safe and Nested Subgame Solving for Imperfect-Information Games, by Brown et al, NeurIPS, 2017

4. **Bonus Problem: OpenAI Spinning Up in Deep RL**. Welcome to Spinning Up in Deep RL! This is an educational resource produced by OpenAI that makes it easier to learn about deep reinforcement learning (deep RL). Please study the documents and install the environment. Either PyTorch or TensorFlow are allowed. In your report, please provide detailed figures and analysis to show many aspects of performances of various DRL algorithms.

   (a) Finish the problem set 1: "Basics of Implementation" . It includes three exercises: Gaussian Log-Likelihood, Policy for PPO, and Computation Graph for TD3.

(b) Finish the problem set 2: "Algorithm Failure Modes" . It includes two exercises: Value Function Fitting in TRPO, and Silent Bug in DDPG.