Mini-Course 1, Module 3
Value Functions & Bellman Equations

CMPUT 365
Fall 2021

Reminders: Sept 20, 2020

- **First mini-essay is due October 20**
    - We will release a list of topics soon & in lecture I will give some tips on effective writing
- Keep asking questions on Discord: its ok to ask about the quizzes, they are there to help ensure you absorbed the key ideas
- If you are having weird Coursera issues, email the TAs
- **If you want to appeal your Peer Review grade, please email the TAs**
- No late submissions accepted, so please ensure you’ve hit submit. You get auto-graded for Quizzes/Notebooks, so its easy to check if you’ve submit
- **Please do not post any solutions online**

Review of Mini-Course 1, Module 3

Video 1: Policies

- All about policies. All about how our agents **select actions**
- Goals:
    - recognize that a policy is a **distribution** over actions for each state
    - describe the similarities and differences between **stochastic** and **deterministic policies**
    - generate valid policies for a given MDP, or Markov Decision Process.
- *In plain words what does a policy describe?*

Video 2: Value Functions

- All about value functions, the key data structure of RL
- Goals:
    - describe the roles of the **state-value** and **action-value** functions in reinforcement learning
    - describe the relationship between **value functions** and **policies**
    - create examples of value functions for a given MDP.
- *What is the difference between the value functions we saw in Bandits compared with CH3?*

Video 3: Bellman Equation Derivation

- Bellman equations: the foundation of many RL algorithms
- Goals:
  - derive the Bellman equation for **state value functions**
  - derive the Bellman equation for **action-value functions**
  - understand how Bellman equations relate **current and future values**.
- *This is one of the key concepts in RL and is the basis for Q-learning!*
  - *It says we can reason about values (which are infinite sums) by just looking at the value in one state and the value in the next state plus the reward along the way!*

Video 4: Why Bellman Equations?

- Why are Bellman equations so important in RL
- Goals:
  - use the Bellman equations to **compute** value functions
  - understand how Bellman Equations will allow our algorithms to make updates now, to take into account the future
- *Some of you have wondered “returns include future rewards”, what aspect of value functions gives a hint of how we deal with this?*

Video 5: Optimal Policies

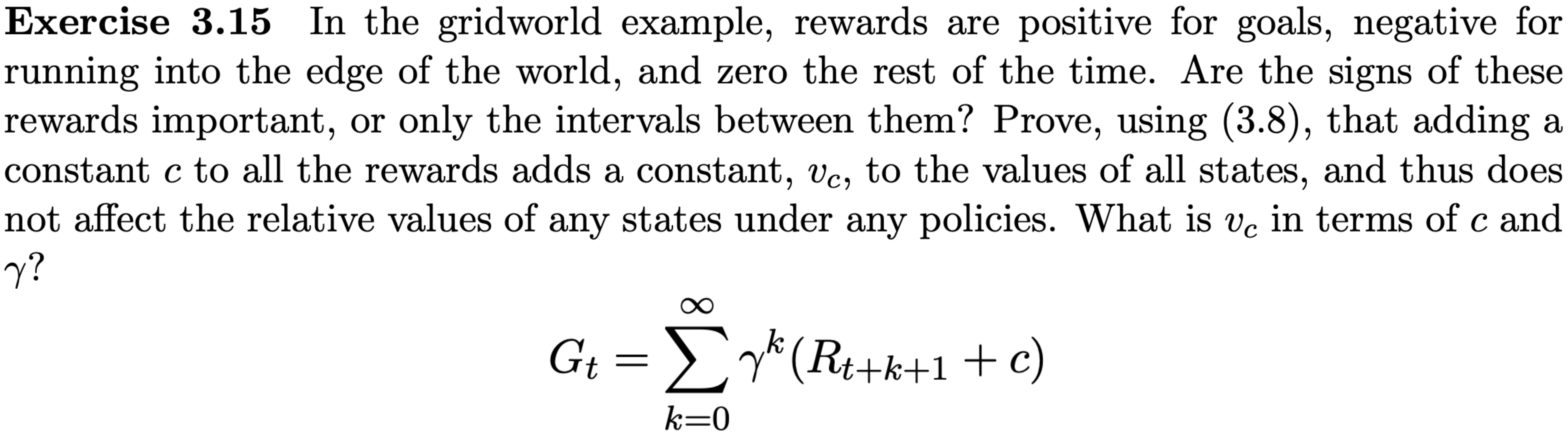
- Formalizing our goals: policy that obtains as much reward as possible in the long run
- **Goals**:
  - define an **optimal** policy
  - understand how a policy can be **at least as good** as every other policy in every state
  - Identify an optimal policy for a given MDP.
- *Why can there be several optimal policies for an MDP?*

Video 6: Using Optimal Value Functions to get Optimal Policies

- A hint of how our agents might use value functions to select actions
- **Goals**:
  - understand the connection between the optimal value function and optimal policies
  - **verify** the optimal value function for given MDPs.
- *Can you think of an expensive algorithm for finding the optimal policy of an MDP?*

Quiz review

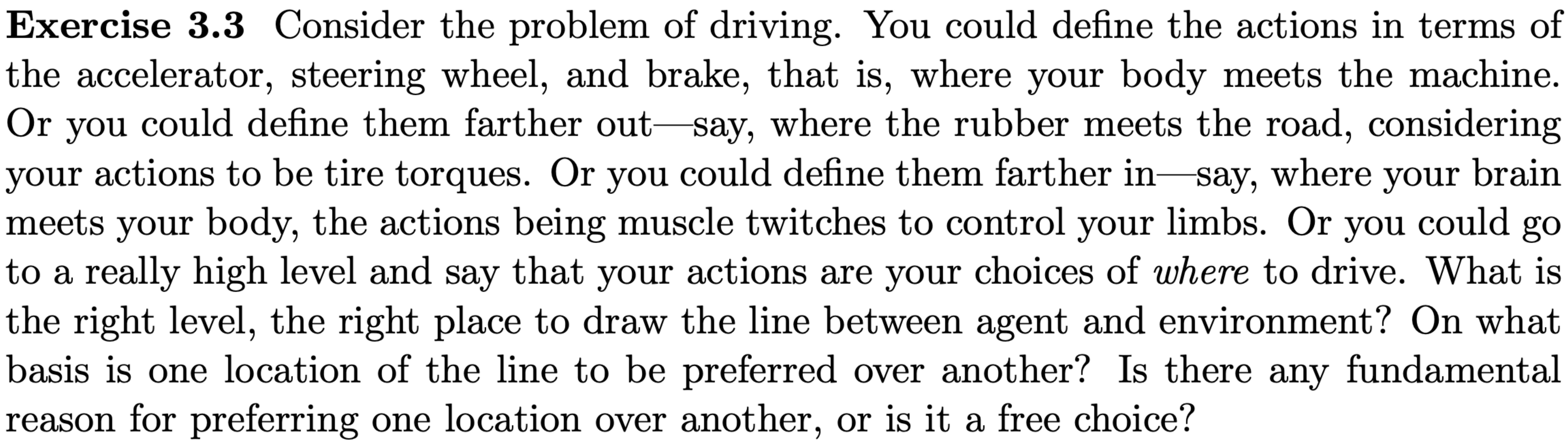
- https://www.coursera.org/learn/fundamentals-of-reinforcement-learning/quiz/AxJgj/practice-value-functions-and-bellman-equations



Exercise 3.15 In the gridworld example, rewards are positive for goals, negative for running into the edge of the world, and zero the rest of the time. Are the signs of these rewards important, or only the intervals between them? Prove, using (3.8), that adding a constant c to all the rewards adds a constant, v_c, to the values of all states, and thus does not affect the relative values of any states under any policies. What is v_c in terms of c and γ?

$$G_t = \sum_{k=0}^{\infty} \gamma^k (R_{t+k+1} + c)$$

Your questions!

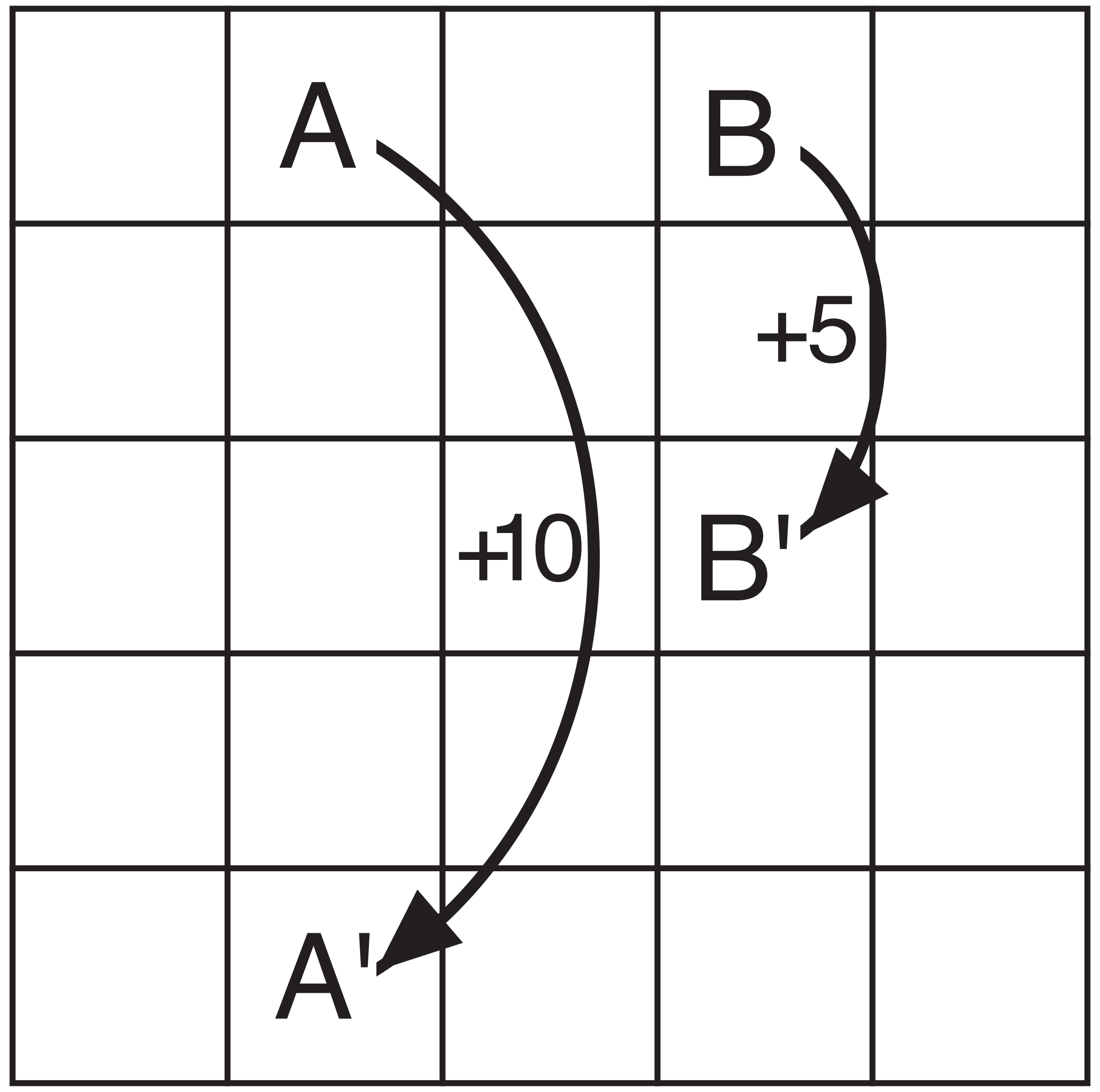
- How do we determine the state in an MDP?
  - Are you the problem designer or the solution (agent) designer?
- 

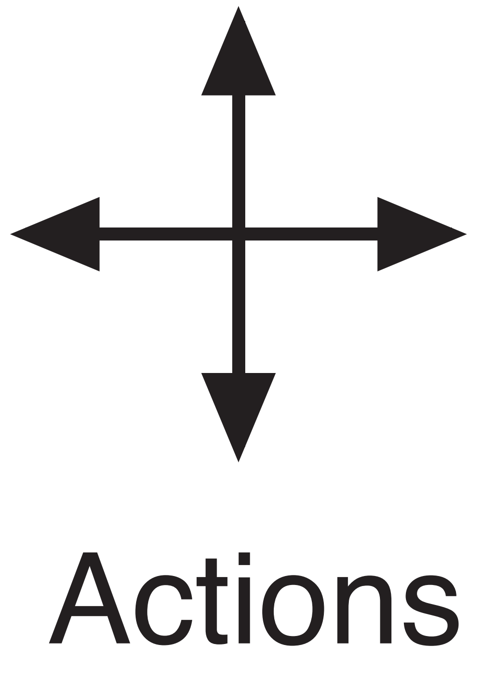
  Exercise 3.3 Consider the problem of driving. You could define the actions in terms of the accelerator, steering wheel, and brake, that is, where your body meets the machine. Or you could define them farther out—say, where the rubber meets the road, considering your actions to be tire torques. Or you could define them farther in—say, where your brain meets your body, the actions being muscle twitches to control your limbs. Or you could go to a really high level and say that your actions are your choices of *where* to drive. What is the right level, the right place to draw the line between agent and environment? On what basis is one location of the line to be preferred over another? Is there any fundamental reason for preferring one location over another, or is it a free choice?

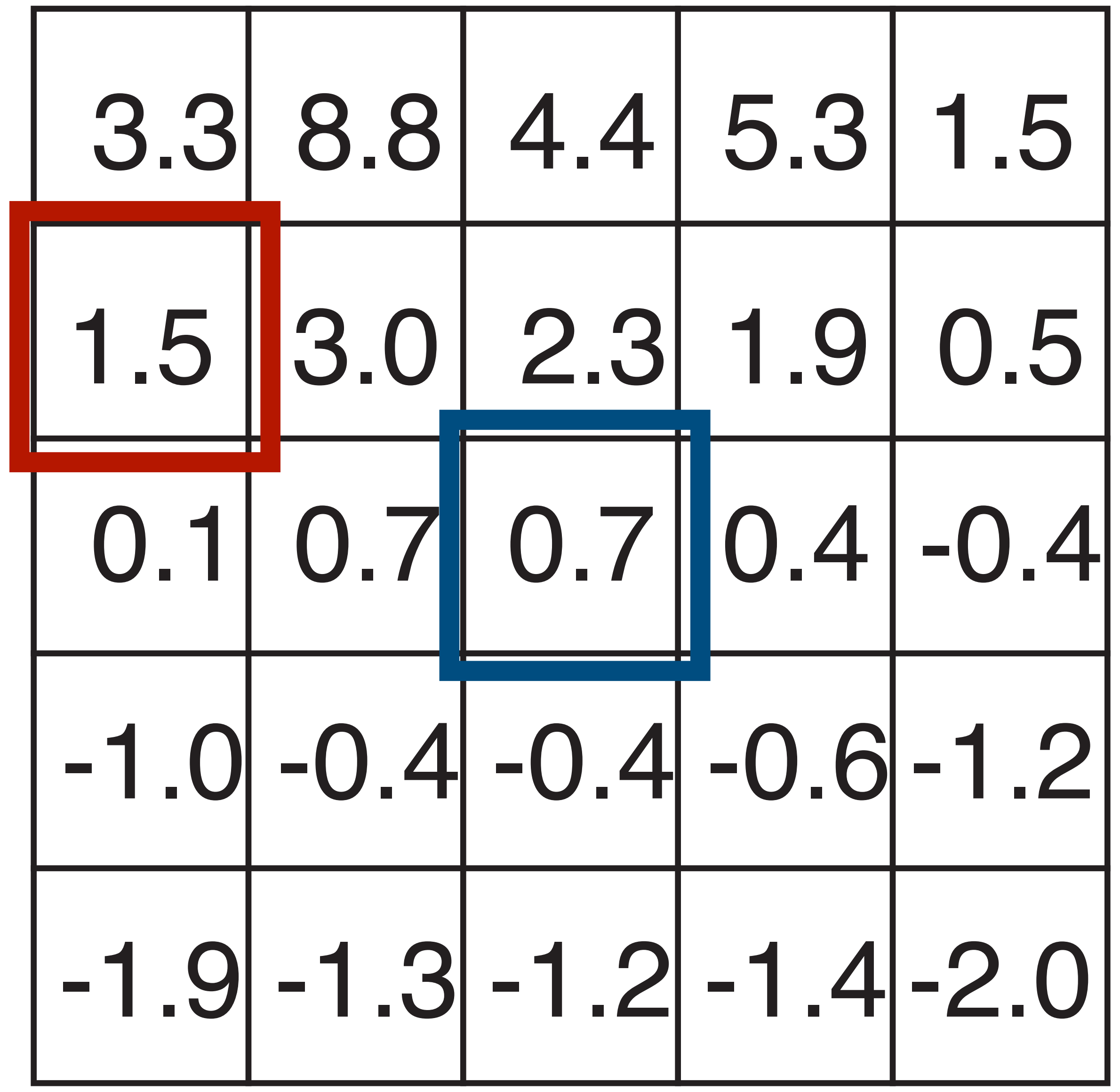
Practice Question

The Bellman equation (3.10) must hold for each state for the value function v_\pi shown in Figure 3.2. As an example, show numerically that this equation holds for the center state, valued at +0.7, with respect to its four neighboring states, valued at +2.3, +0.4, -0.4, and +0.7. (These numbers are accurate only to one decimal place.). **Harder one:** verify the red state.

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r \,|\, s, a)\Big[r + \gamma v_\pi(s')\Big], \quad \text{for all } s \in \mathcal{S},$$







3.3	8.8	4.4	5.3	1.5
1.5	3.0	2.3	1.9	0.5
0.1	0.7	0.7	0.4	-0.4
-1.0	-0.4	-0.4	-0.6	-1.2
-1.9	-1.3	-1.2	-1.4	-2.0

γ = 0.9
π = random
-1 reward on bump

Worksheet Question 1

Express the action-value function q_π in terms of v_π. The formula will also include p and π.

$$q_\pi(s,a) \quad = \quad \mathbb{E}[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s, A_t = a]$$