# Policies and Value Functions

CMPUT 366: Intelligent Systems

S&B §3.5

# Lecture Outline

1. Recap
2. Policies & Value Functions
3. Bellman Equations

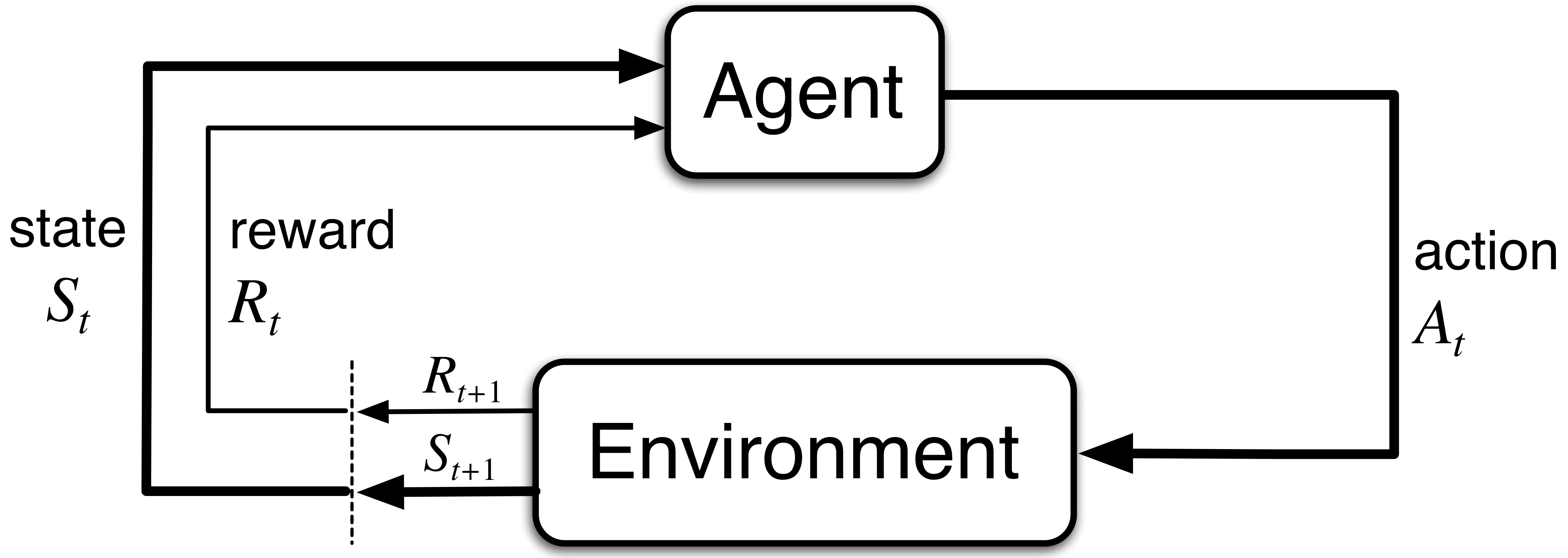
# Recap:
# Interacting with the Environment

At each time $t$ = 1, 2, 3, ...

1. Agent receives input denoting **current state** $S_t$

2. Agent chooses **action** $A_t$

3. Next time step, agent receives **reward** $R_{t+1}$ and **new state** $S_{t+1}$, chosen according to a distribution $p(s',r|s,a)$




This interaction between agent and environment produces a **trajectory**:
$S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, R_3, \ldots$

# Policies

**Question:** How should an agent in a Markov decision process choose its **actions**?

- **Markov assumption**: The state incorporates all of the necessary information about the history up until this point
  - i.e., Probabilities of future rewards & transitions are the same from state $S_t$ **regardless of how you got there**
- So the agent can choose its actions based **only** on $S_t$
- This is called a **policy**: $\pi(a|s) \in [0,1]$ is

  the probability of taking **action** $a$ given that the **current state** is $s$

# State-Value Function

- Once you know the **policy** $\pi$ and the **dynamics** ***p***, you can compute the probability of every possible state transition starting from any given state

- It is often valuable to know the **expected return** starting from a given state *s* under a given policy $\pi$ (**why?**)

- The **state-value function** $v_\pi$ estimates this quantity:

$$v_\pi(s) \doteq \mathbb{E}_\pi[G_t \mid S_t = s]$$
$$= \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \,\middle|\, S_t = s\right]$$

# Action-Value Function

The **action-value function** $q_\pi(s,a)$ estimates the expected return $G_t$ starting from state $s$ if we

1. Take action $a$ in state $S_t = s$, **and then**

2. Follow policy $\pi$ for every state $S_{t+1}$ afterward

$$
\begin{aligned}
q_\pi(s,a) &\doteq \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a] \\
&= \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \,\middle|\, S_t = s, A_t = a\right]
\end{aligned}
$$

# Bellman Equations

Value functions satisfy a **recursive consistency condition** called the **Bellman equation**:

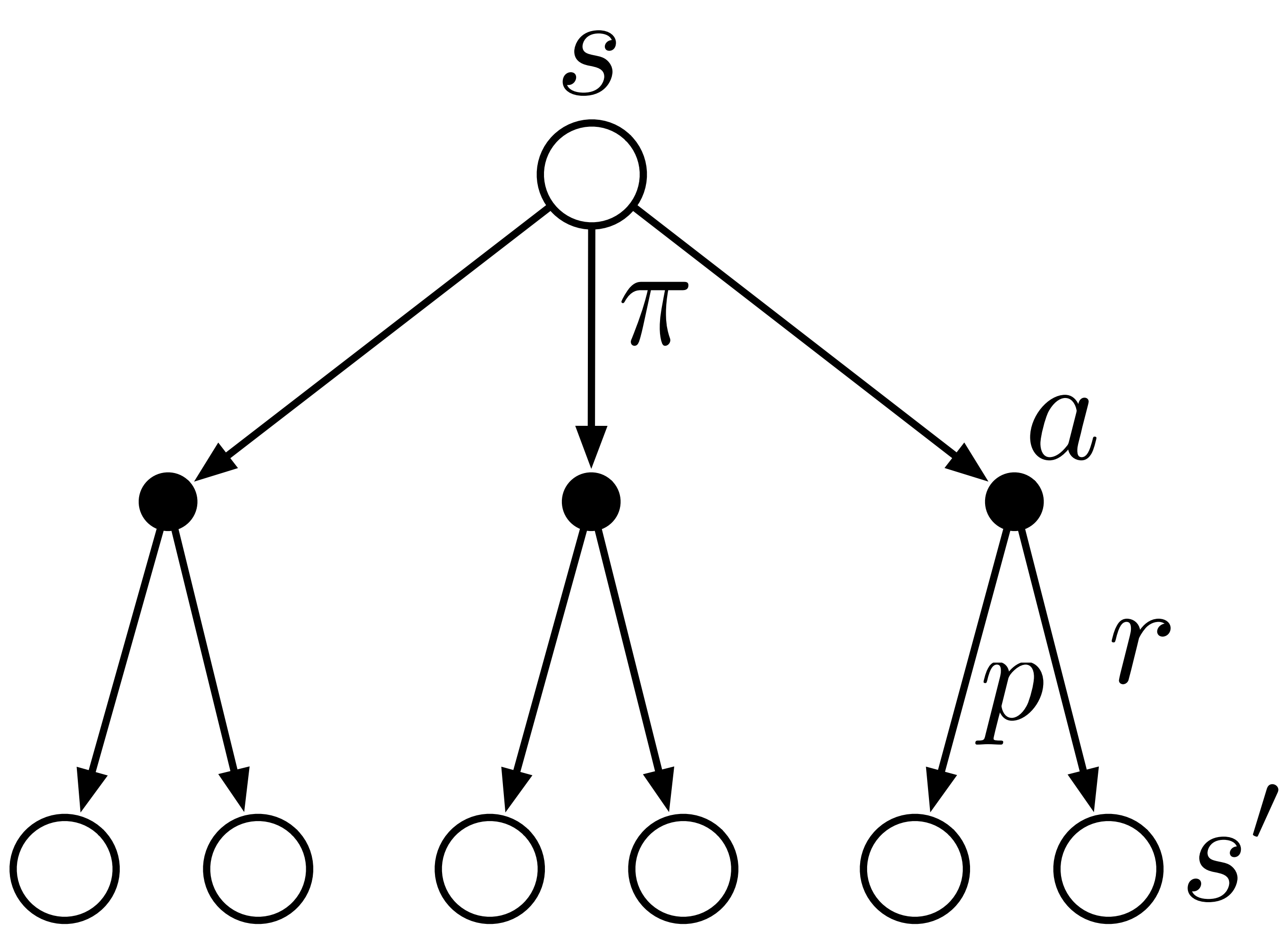
$$
\begin{aligned}
v_\pi(s) &\doteq \mathbb{E}_\pi[G_t \mid S_t = s] \\
&= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1} \mid S_t = s] \\
&= \sum_a \pi(a \mid s) \sum_{s'} \sum_r p(s', r \mid s, a) \Big[ r + \gamma \mathbb{E}_\pi[G_{t+1} \mid S_{t+1} = s'] \Big] \\
&= \sum_a \pi(a \mid s) \sum_{s', r} p(s', r \mid s, a) \Big[ r + \gamma v_\pi(s') \Big]
\end{aligned}
$$

- $v_\pi$ is the **unique solution** to $\pi$'s Bellman equation
- There is also a Bellman equation for $\pi$'s **action-value function**

# Backup Diagrams

Backup diagrams help to visualize the flow of **information back to a state** from its successor states or action-state pairs:

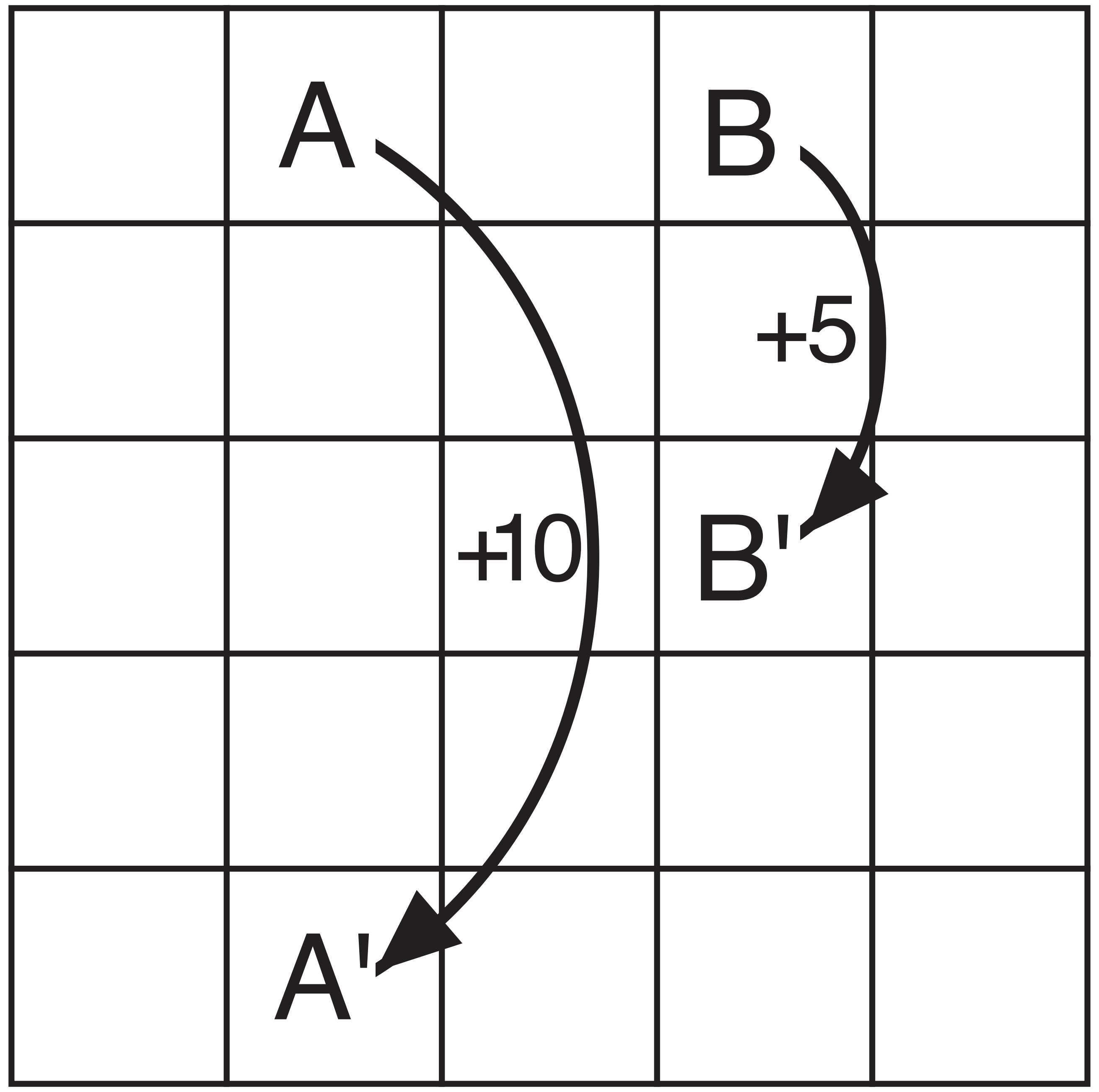
$$
\begin{aligned}
v_\pi(s) &\doteq \mathbb{E}_\pi[G_t \mid S_t = s] \\
&= \sum_a \pi(a \mid s) \sum_{s',r} p(s', r \mid s, a)\big[r + \gamma v_\pi(s')\big]
\end{aligned}
$$



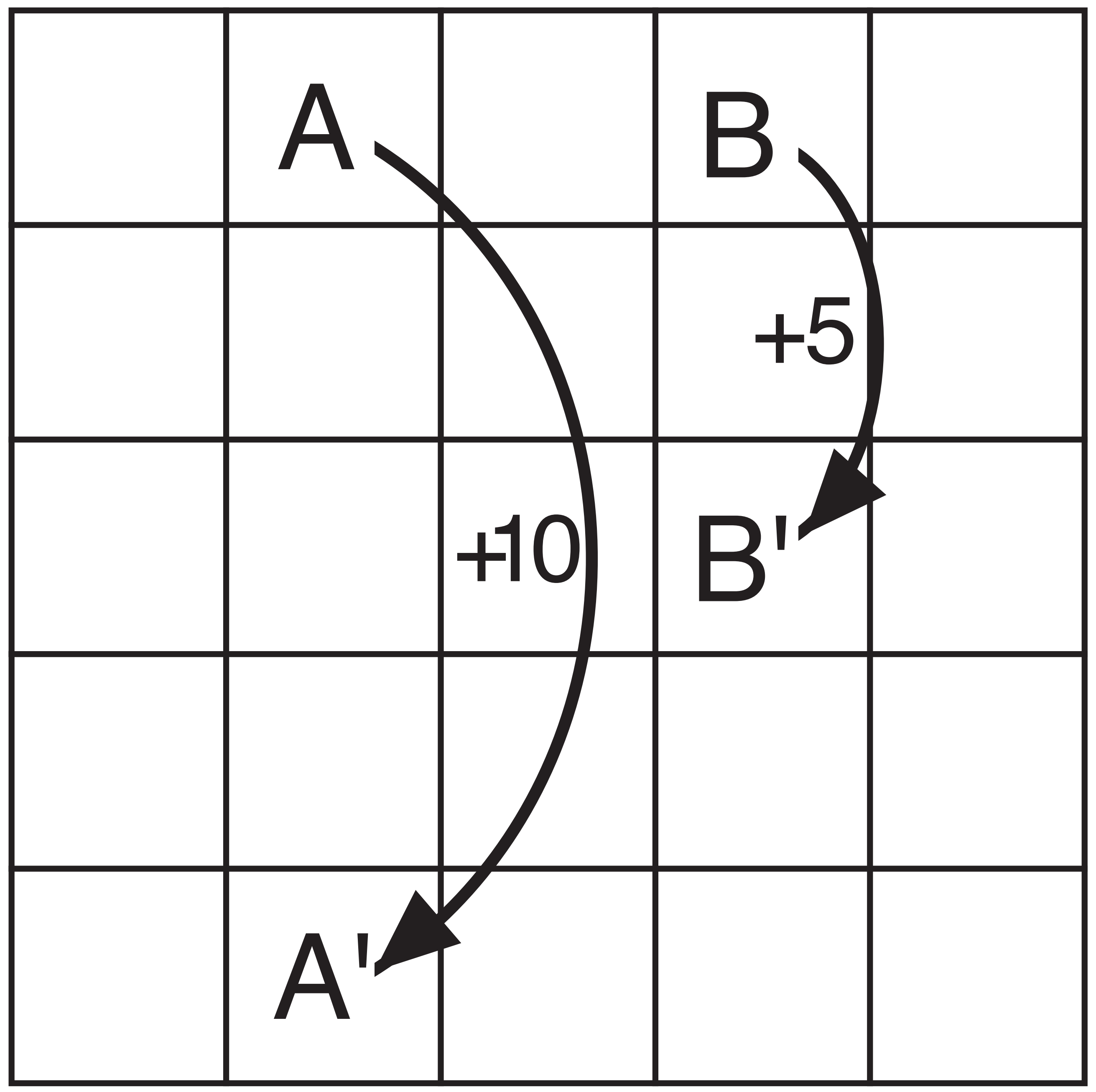

Backup diagram for $v_\pi$

# Return to GridWorld

- At each cell, can go `north`, `south`, `east`, `west`
- Try to go off the **edge**: reward of **-1**
- Leaving state **A**: takes you to state **A′**, reward of **+10**
- Leaving state **B**: takes you to state **B′**, reward of **+5**



# Return to GridWorld




Reward dynamics

| | | | | |
|---|---|---|---|---|
| 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 |

State-value function $v_\pi$ for random policy $\pi(a|s) = 0.25$

# Summary

- **Policies** map states to (distribution over) actions
- Given a **policy** $\pi$, every state $s$ has an **expected value** $v_\pi(s)$
  - and every action $a$ from state $s$ has value $q_\pi(s,a)$
  - These are the **state-value** and **action-value** functions
- State-value and action-value functions satisfy the **Bellman equations**