Markov Decision Process

Chao Lan
Outline

Introduction

Value Iteration

Policy Iteration

Monte Carlo Methods
Recap: State, Action, Reward, Dynamics

The diagram illustrates a grid with states labeled from 1 to 3, 1 to 4. The transitions and rewards are as follows:

- From state 1, moving to state 2 results in a reward of -0.04.
- From state 1, moving to state 3 results in a reward of -0.04.
- From state 1, moving to state 4 results in a reward of -0.04.
- From state 2, moving to state 2 results in a reward of -1.
- From state 2, moving to state 3 results in a reward of -0.04.
- From state 2, moving to state 4 results in a reward of -0.04.
- From state 3, moving to state 2 results in a reward of 0.8.
- From state 3, moving to state 3 results in a reward of 0.8.
- From state 3, moving to state 4 results in a reward of 0.8.

The diagram also shows the transition probabilities as 0.1 in each direction.
Recap: Utility of Sequence

Discounted reward: $U = R(S_0) + \gamma R(S_1) + \gamma^2 R(S_2) + \gamma^3 R(S_3) + \ldots$
Recap: Policy

- $R(s) < -1.6284$
- $-0.4278 < R(s) \mu < -0.0850$
- $-0.0221 < R(s) \mu < 0$
- $R(s) \mu > 0$
Recap: Utility of a State by some Policy

\[ U^\pi(s) = E \left[ \sum_{t=0}^{\infty} \gamma^t R(S_t) \right] \]
Recap: Optimal Policy

\[ \pi^*_s = \arg\max_{\pi} U^\pi(s) \]

\[ U^\pi(s) = E \left[ \sum_{t=0}^{\infty} \gamma^t R(S_t) \right] \]
How to find an optimal solution?
Bellman Equation

It says the utility of a state is the immediate reward for that state plus the expected discounted utility of the next state, assuming the agent chooses the optimal action.

$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s')$$

- it connects utility of a state and utility of its neighbors
Bellman Equation

It says the utility of a state is the immediate reward for that state plus the expected discounted utility of the next state, assuming the agent chooses the optimal action.

\[
U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s')
\]

- it connects utility of a state and utility of its neighbors
- Q: how to derive it?  
  \[
  U^\pi(s) = E \left[ \sum_{t=0}^{\infty} \gamma^t R(S_t) \right]
  \]
Derivation of Bellman Equation

\[ U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s') \]

Recall

\[ U^\pi(s) = E \left[ \sum_{t=0}^{\infty} \gamma^t R(S_t) \right] \]

\[ U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s') \]
Example

Let’s compute the utility of state (1,1) using Bellman equation.

\[
U(1, 1) = -0.04 + \gamma \max [\begin{array}{l}
(U_p) \\
(Left) \\
(Down) \\
(Right)
\end{array}]
\]

\[
U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' | s, a) U(s')
\]
Example

Let’s compute the utility of state \((1, 1)\) using Bellman equation.

\[
U(1, 1) = -0.04 + \gamma \max [0.8U(1, 2) + 0.1U(2, 1) + 0.1U(1, 1), \quad \text{(Up)}
\]

\[
(Left) \quad (Down) \quad \text{(Right)}
\]

\[
U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' | s, a) U(s')
\]
Example

Let’s compute the utility of state \((1,1)\) using Bellman equation.

\[
U(1, 1) = -0.04 + \gamma \max\left[ 0.8U(1, 2) + 0.1U(2, 1) + 0.1U(1, 1), 0.9U(1, 1) + 0.1U(1, 2), U_p \right],
\]

\[(Left), (Down), (Right)\]

\[
U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s')
\]
Example

Let’s compute the utility of state (1,1) using Bellman equation.

\[ U(1, 1) = -0.04 + \gamma \max \left[ 0.8U(1, 2) + 0.1U(2, 1) + 0.1U(1, 1), \quad (Up) \right. \\
\left. 0.9U(1, 1) + 0.1U(1, 2), \quad (Left) \right. \\
\left. 0.9U(1, 1) + 0.1U(2, 1), \quad (Down) \right. \\
\left. \right]. \quad (Right) \\
\]

\[ U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' | s, a) U(s') \]
Example

Let’s compute the utility of state (1,1) using Bellman equation.

\[
U(1, 1) = -0.04 + \gamma \max \left[ 0.8U(1, 2) + 0.1U(2, 1) + 0.1U(1, 1), \quad (U_p) \\
0.9U(1, 1) + 0.1U(1, 2), \quad (Left) \\
0.9U(1, 1) + 0.1U(2, 1), \quad (Down) \\
0.8U(2, 1) + 0.1U(1, 2) + 0.1U(1, 1) \right]. \quad (Right)
\]

\[
U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s')
\]
Example

Now we identify “Up” as the optimal policy in state (1,1), i.e., \( \pi^*(1,1) = \text{Up} \).

\[
U(1,1) = -0.04 + \gamma \max\left[ 0.8U(1,2) + 0.1U(2,1) + 0.1U(1,1), \quad 0.7456 \quad (\text{Up}) \\
0.9U(1,1) + 0.1U(1,2), \quad 0.7107 \quad (\text{Left}) \\
0.9U(1,1) + 0.1U(2,1), \quad 0.7 \quad (\text{Down}) \\
0.8U(2,1) + 0.1U(1,2) + 0.1U(1,1) \right] 0.6707 \quad (\text{Right})
\]

\[
U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' | s, a) U(s')
\]
Example

Now we identify “Up” as the optimal policy in state (1,1), i.e., \( \pi^*(1,1) = \text{Up} \).

But how to get state utilities?

\[
U(1, 1) = -0.04 + \gamma \max \left[ 0.8U(1, 2) + 0.1U(2, 1) + 0.1U(1, 1), \ 0.7456 \ (\text{Up}) \right. \\
0.9U(1, 1) + 0.1U(1, 2), \ 0.7107 \ (\text{Left}) \\
0.9U(1, 1) + 0.1U(2, 1), \ 0.7 \ (\text{Down}) \\
0.8U(2, 1) + 0.1U(1, 2) + 0.1U(1, 1) \left. \right] 0.6707 \ (\text{Right})
\]

\[
U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' | s, a) U(s')
\]
Assume there are $n$ states.

If we treat $U(s)$ as an unknown variable, we have $n$ variables.

$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s')$$
Observations

Assume there are $n$ states.

If we treat $U(s)$ as an unknown variable, we have $n$ variables.

If we build one Bellman equation for each $U(s)$, we have $n$ equations.

$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' | s, a) U(s')$$
Observations

Assume there are n states.

If we treat $U(s)$ as an unknown variable, we have n variables.

If we build one Bellman equation for each $U(s)$, we have n equations.

Just solve n variables from n equations!

$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' | s, a) U(s')$$
Value Iteration Algorithm

1. initialize all $U(s)$ randomly.

2. update all $U(s)$ by B.E. *simultaneously* (Bellman update)

   $$U_{i+1}(s) \leftarrow R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' | s, a) U_i(s')$$

3. repeat until converge
   - convergence is guaranteed
   - converge to unique solutions to B.E.

Refer to [AIMA, Chapter 17.2.3] for proof of convergence.
Demo

VALUES AFTER 1 ITERATIONS

This example is from the lecture slides of “Markov Decision Process and Exact Solution Methods” by Pieter Abbeel.
VALUES AFTER 4 ITERATIONS
VALUES AFTER 5 ITERATIONS

0.51  0.72  0.84  1.00

0.27

0.55  -1.00

0.00  0.22  0.37  0.13
VALUES AFTER 100 ITERATIONS

0.64  0.74  0.85  1.00

0.57  0.57  -1.00

0.49  0.43  0.48  0.28

Demo
VALUES AFTER 1000 ITERATIONS

0.64  0.74  0.85  1.00
0.57  0.57 -1.00
0.49  0.43  0.48  0.28
Convergence of State Utilities
Effects of Discount Factor and Threshold

Observations

Algorithm converges faster with
- smaller discount factor
- larger c

Discount factor has (almost) exponential impact on convergence rate.

Smaller c means stricter convergence criterion.