Value and Policy Iteration

DOROZHKO Anton

Novosibirsk State University

May 20, 2019

Outline

Novosibirsk State University
*THE REAL SCIENCE

Intorduction to MDPs

- MDP descrives environment
- **Fully observable** - state completely characterises the process
- Almost all RL problems can be formalised as MDPs

Definition

A state S_t is Markov iff

$$\mathbb{P}[S_{t+1}|S_t] = \mathbb{P}[S_{t+1}|S_1, ..., S_t]$$

- captures all relevant information
- can throw away the history if we know state

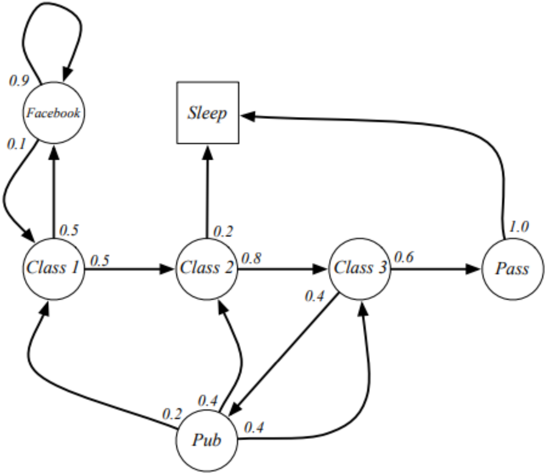
Definition

A Markov Process (or Markov Chain) is a tuple $(\mathcal{S}, \mathcal{P})$

- $\mathcal{S}$ is a (finite) set of states
- $\mathcal{P}$ is a transition probability matix

$$\mathcal{P}_{ss'} = \mathbb{P}[S_{t+1} = s' | S_t = s]$$

Student's MRP [1]



[1] from David Silver course

Markov Reward Process

Definition

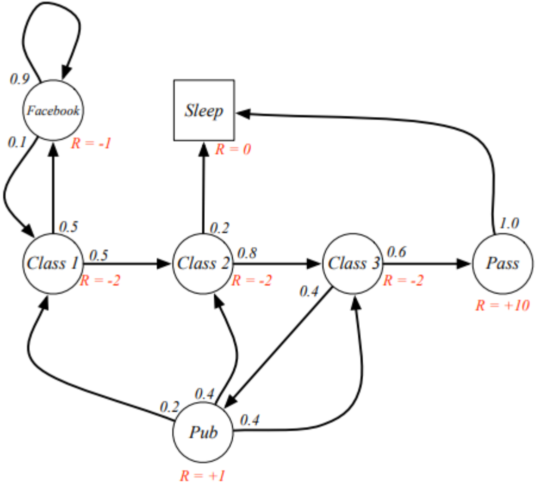
A Markov Reward Process is a tuple $(\mathcal{S}, \mathcal{P}, \mathcal{R}, \gamma)$

- $\mathcal{S}$ is a (finite) set of states
- $\mathcal{P}$ is a transition probability matix

$$\mathcal{P}_{ss'} = \mathbb{P}[S_{t+1} = s' | S_t = s]$$

- $\mathcal{R}$ is a reward function : $\mathcal{R}_s = \mathbb{E}[R_{t+1} | S_t = s]$
- γ is a discount factor, $\gamma \in [0, 1]$

Student's MRP [2]

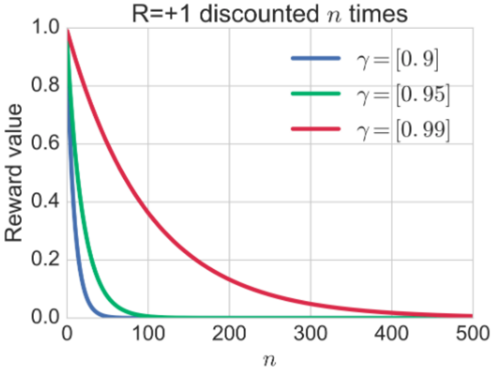


[2] from David Silver course

Discounted Reward

$$G_t = R_t + \gamma R_{t+1} \ldots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

If max $R = 1$ then $G_0 = \sum \gamma^k = \frac{1}{1-\gamma}$



Discounted Reward

$$G_t = R_t + \gamma R_{t+1} \ldots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

$$G_t = R_t + \gamma G_{t+1}$$

Novosibirsk State University
*THE REAL SCIENCE

Value Function

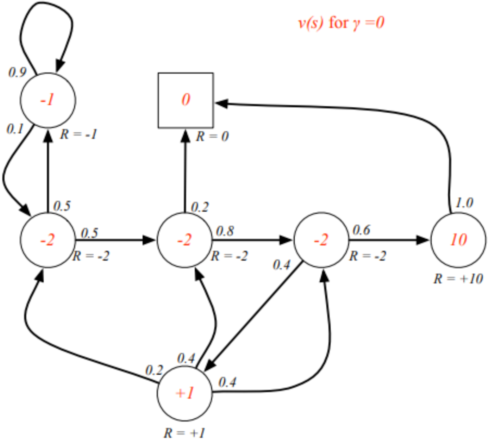
Definition

The state value function $V(s)$ of an MRP is the expected return starting from state s

$$V(s) = \mathbb{E}[G_t | S_t = s]$$

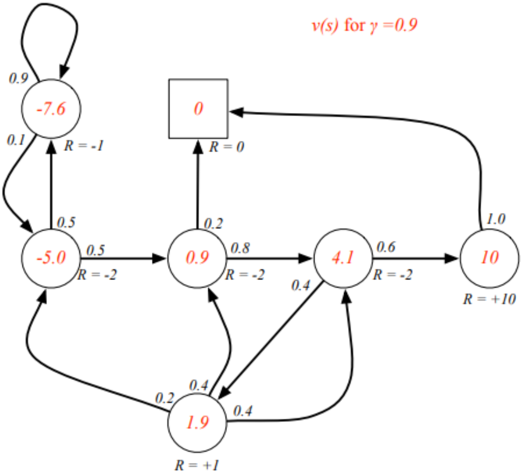
$V(s)$ gives the long-term value of state s

State value function's for Student MRP [3]



[3] from David Silver course

State value function's for Student MRP [4]



[4] from David Silver course

Bellman Equation for Value Function

Decomposition:

- immediate reward R_{t+1}
- discounted value of the next state $\gamma V(S_{t+1})$

$$
\begin{aligned}
V(s) &= \mathbb{E}[G_t|S_t = s] \\
&= \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + ...|S_t = s] \\
&= \mathbb{E}[R_{t+1} + \gamma(R_{t+2} + \gamma R_{t+3} + ...)|S_t = s] \\
&= \mathbb{E}[R_{t+1} + \gamma G_{t+1}|S_t = s] \\
&= \mathbb{E}[R_{t+1} + \gamma V_{t+1}|S_t = s]
\end{aligned}
$$

Novosibirsk State University
*THE REAL SCIENCE

Bellman equation MRP

$$V(s) = \mathbb{E}[R_{t+1} + \gamma V(S_{t+1}|S_t = s]$$

$$V(s) = r + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'} V(s'))$$

$$\begin{aligned} V &= \mathcal{R} + \gamma \mathcal{P} V \\ (I - \gamma \mathcal{P}) V &= \mathcal{R} \\ V &= (I - \gamma \mathcal{P})^{-1} \mathcal{R} \end{aligned}$$

Novosibirsk State University
*THE REAL SCIENCE

$$
\begin{aligned}
V &= \mathcal{R} + \gamma \mathcal{P} V \\
(I - \gamma \mathcal{P}) V &= \mathcal{R} \\
V &= (I - \gamma \mathcal{P})^{-1} \mathcal{R}
\end{aligned}
$$

- $\mathcal{O}(n^3)$ for n states
- small MDPs
- Other options:
  - Dynamic programming (DP)
  - Monte-Carlo evaluation (MC)
  - Temporal-Difference learning (TD)

Novosibirsk State University
*THE REAL SCIENCE

Markov Decision Process

Definition

A Markov Decision Process is a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$

- $\mathcal{S}$ is a (finite) set of states
- $\mathcal{A}$ is a finite set of actions
- $\mathcal{P}$ is a transition probability matix

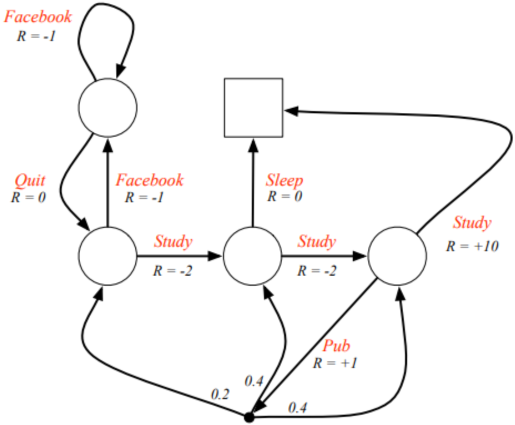
$$\mathcal{P}^{a}{}_{ss'} = \mathbb{P}[S_{t+1} = s' | S_t = s, A_t = a]$$

- $\mathcal{R}$ is a reward function :

$$\mathcal{R}^{a}{}_{s} = \mathbb{E}[R_{t+1} | S_t = s, A_t = a]$$

- γ is a discount factor, $\gamma \in [0, 1]$

Student's MDP [5]



[5] from David Silver course

Policy

Definition

A policy π is a distribution over actions given states

$$\pi(a|s) = \mathbb{P}[A_t = a|S_t = s]$$

Value Function

State-value function

The state value function $V_\pi(s)$ of an MDP is the expected return starting from state s, and then following policy π

$$V_\pi(s) = \mathbb{E}_\pi[G_t | S_t = s]$$

Action-value function

The action-value function $Q_\pi(s, a)$ is expected return starting from state s, taking action a, and then following policy *pi*

$$Q_\pi(s, a) = \mathbb{E}_\pi[G_t | S_t = s, A_t = a]$$

Notation variants

$$
\begin{aligned}
\mathbb{E}[G_0] &= \mathbb{E}[R_0 + \gamma R_1 + \ldots + \gamma^T R_T] \\
&= \mathbb{E}[G_0 | \pi_\theta] \\
&= \mathbb{E}_{\pi_\theta}[G_0] \\
&= \sum_{t=0}^{T} \mathbb{E}_{(s_t, a_t) \sim p_\theta}[\gamma^t R_t] \\
&= \mathbb{E}_{\tau \sim p_\theta(\tau)}[G(\tau)]
\end{aligned}
$$

- $\tau = (s_0, a_0, s_1, a_1, \ldots, a_{T-1}, s_T)$
- $p_\theta(\tau) = p(s_0) \prod_{t=0}^{T-1} \pi_\theta(a_t | s_t) p(s_{t+1} | s_t, a_t)$

Novosibirsk State University
*THE REAL SCIENCE

Bellman Expectation Equation

Decomposition into immediate reward plus discounted value in next state

$$V_\pi(s) = \mathbb{E}_\pi[R_{t+1} + \gamma V_\pi(S_{t+1}) | S_t = s]$$

$$Q_\pi(s, a) = \mathbb{E}_\pi[R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1}) | S_t = s, A_t = a]$$

Novosibirsk State University
*THE REAL SCIENCE

Optimal Value Functions

Definition

The optimal state-value function $V_*(s)$ is the maximum value function over all policies

$$V_*(s) = \max_\pi V_\pi(s)$$

The optimal action-value function $Q_*(s, a)$ is the maximum action-value function over all policies

$$Q_*(s, a) = \max_\pi Q_\pi(s, a)$$

Optimal Policy

Partial ordering over policies

$$\pi \geq \pi' \quad \text{if} \quad V_\pi(s) \geq V_{\pi'}(s) \quad \forall s$$

Theorem

For any Markov Decision Process

- *There exists an optimal policy π_* that is better than or equal to all other policies $\pi_* \geq \pi, \forall \pi$*
- *All optimal policies achieve the optimal values function $V_{\pi_*}(s) = V_*(s)$*
- *All optimal policies achieve the optimal action-value function $Q_{\pi_*}(s, a) = Q_*(s, a)$*

THE REAL SCIENCE

Iterative algorithm

- Initialize $V_0(s) = 0$ for all s
- for $k = 1$ until convergence
    - for all $s \in \mathcal{S}$

$$V_k(s) = R(s) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s) V_{k-1}(s')$$

- $\mathcal{O}(|S|^2)$ for each iteration

Novosibirsk State University
*THE REAL SCIENCE

MDP + Policy

- MDP $+ \pi(a|s) =$ Markov Reward Process
- MRP($\mathcal{S}, \mathcal{R}^{\pi}, \mathcal{P}^{\pi}, \gamma$), where

$$R^{\pi}(s) = \sum_{a \in A} \pi(a|s) R(s, a)$$

$$P^{\pi}(s'|s) = \sum_{a \in A} \pi(a|s) P(s'|s, a)$$

- We can reuse iterative algorithm

Novosibirsk State University
THE REAL SCIENCE

Iterative algorithm

- Initialize $V_0(s) = 0$ for all s
- for $k = 1$ until convergence
  - for all $s \in \mathcal{S}$

$$V_k^\pi(s) = r(s, \pi(s)) + \gamma \sum_{s' \in \mathcal{S}} p(s'|s, \pi(s)) V_{k-1}^\pi(s')$$

- **Bellman backup** for particular policy

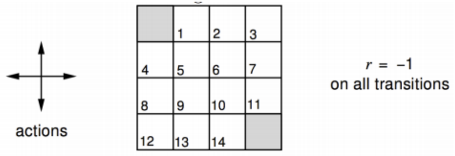
MDP Control

- Compute optimal policy

$$\pi^*(s) = arg\max_{\pi} V^{\pi}(s)$$

- There **exists a unique value function**

Gridworld example



- Undiscounted episodic MDP ($\gamma = 1$)
- Nonterminal states $1, \dots, 14$
- One terminal state (shown twice as shaded squares)
- Actions leading out of the grid leave state unchanged
- Reward is -1 until the terminal state is reached
- Agent follows uniform random policy

$$\pi(n|\cdot) = \pi(e|\cdot) = \pi(s|\cdot) = \pi(w|\cdot) = 0.25$$

Novosibirsk State University
*THE REAL SCIENCE

Gridworld example

v_k for the Random Policy | Greedy Policy w.r.t. v_k

$k = 0$

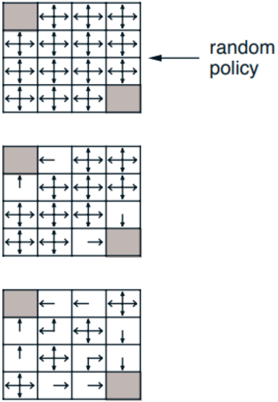
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0

$k = 1$

0.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	0.0

$k = 2$

0.0	-1.7	-2.0	-2.0
-1.7	-2.0	-2.0	-2.0
-2.0	-2.0	-2.0	-1.7
-2.0	-2.0	-1.7	0.0



Policy Iteration(PI)

- $i = 0$
- Initialize $\pi_0(s)$ randomly for all s
- While $i == 0$ or $||\pi_i - \pi_{i-1}|| > 0$
    - $V^\pi \leftarrow$ MDP policy evaluation of π_i
    - $\pi_{i+1} \leftarrow$ Policy improvement
    - $i = i + 1$

Q function

Action value or State-Action value or Q-function

$$Q_\pi(s, a) = \mathbb{E}_\pi[R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1}) | S_t = s, A_t = a]$$

$$Q^\pi(s, a) = R(s, a) + \gamma \sum_{s' \in S} P(s'|s, a) V^\pi(s')$$

Take action a, then follow policy π

Policy Improvement

- Compute Q function of π_i
  - For $s \in S$ and $a \in A$:

$$Q^{\pi_i}(s,a) = R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V^{\pi_i}(s')$$

- Compute new policy π_{i+1}

$$\pi_{i+1}(s) = arg \max_a Q^{\pi_i}(s,a) \quad \forall s \in S$$

Monotonic Improvement in Policy

$$
\begin{aligned}
V^{\pi_i}(s) &\leq \max_a Q^{\pi_i}(s,a) \\
&= \max_a R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V^{\pi_i}(s') \\
&= R(s,\pi_{i+1}(s)) + \gamma \sum_{s' \in S} P(s'|s,\pi_{i+1}(s)) V^{\pi_i}(s') \\
&\leq R(s,\pi_{i+1}(s)) + \gamma \sum_{s' \in S} P(s'|s,\pi_{i+1}(s)) \max_{a'} Q^{\pi_i}(s',a') \\
&\text{continue to expand } a' = \pi_{i+1}(s') \\
&= V^{\pi_{i+1}}(s)
\end{aligned}
$$

Value Iteration

- Policy iteration : computes optimal value and policy
- Value Iteration :
    - Optimal value for state s if k episodes left
    - Iterate to consider longer episodes

Novosibirsk State University
*THE REAL SCIENCE

Gridworld example

Example: Shortest Path

g			

Problem

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

V_1

0	-1	-1	-1
-1	-1	-1	-1
-1	-1	-1	-1
-1	-1	-1	-1

V_2

0	-1	-2	-2
-1	-2	-2	-2
-2	-2	-2	-2
-2	-2	-2	-2

V_3

0	-1	-2	-3
-1	-2	-3	-3
-2	-3	-3	-3
-3	-3	-3	-3

V_4

0	-1	-2	-3
-1	-2	-3	-4
-2	-3	-4	-4
-3	-4	-4	-4

V_5

0	-1	-2	-3
-1	-2	-3	-4
-2	-3	-4	-5
-3	-4	-5	-5

V_6

0	-1	-2	-3
-1	-2	-3	-4
-2	-3	-4	-5
-3	-4	-5	-6

V_7

Novosibirsk State University
*THE REAL SCIENCE

Value Iteration

- $k = 1$
- Init $V_0(s) = 0$
- Look until convergence / T
    - $$V_{k+1}(s) = \max_a R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V_k(s')$$
    - Operator view
    $$V_{k+1} = BV_k$$
    - $\pi_{k+1}(s) = arg\max_a R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V_k(s')$

Novosibirsk State University
*THE REAL SCIENCE

Contraction Operator

- Let O be an operator, and $||.||$ denote any norm of x
- if $||OV - OV'|| \leq ||V - V'||$ then O is a contraction operator
- O has fixed point x
- $Ox = x$

Proof: Bellman Backup is a Contraction on V for $\gamma < 1$

$$||V - V'|| = max_s ||V(s) - V'(s)||$$

Exercise

POMDP

Figure: Doom Classic

A Partially Observable Markov Decision Process is an MDP with hidden states. It is HMM with actions.

Novosibirsk State University
*THE REAL SCIENCE

POMDP

Definition

A POMDP is a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{P}, \mathcal{R}, \mathcal{Z}, \gamma)$

- $\mathcal{S}$ is a (finite) set of states
- $\mathcal{A}$ is a finite set of actions
- $\mathcal{O}$ is a finite set of observations
- $\mathcal{P}$ is a transition probability matix
  $\mathcal{P}^a_{ss'} = \mathbb{P}[S_{t+1} = s' | S_t = s, A_t = a]$
- $\mathcal{R}$ is a reward function : $\mathcal{R}^a_s = \mathbb{E}[R_{t+1} | S_t = s, A_t = a]$
- $\mathcal{Z}$ is an observation function $\mathcal{Z}^a_{s'o} = \mathbb{P}[O_{t+1} = o | S_t = s', A_t = a]$
- γ is a discount factor, $\gamma \in [0, 1]$

University
*THE REAL SCIENCE

Questions ?

The only stupid question is the one you were afraid to ask but never did

- Rich Sutton