RL: MDP

The Markov Reward Process

Marius Lindauer

tnt

Leibniz Universität Hannover

Automated Machine Learning Hannover

AutoML.org

Markov Decision Process (MDP)

- Markov Decision Process is Markov Reward Process + actions
- Definition of MDP
  - S is a (finite) set of Markov states $s \in S$
  - A is a (finite) set of actions $a \in A$
  - P is dynamics/transition model for each action, that specifies $P(s_{t+1} = s' \mid s_t = s, a_t = a)$
  - R is a reward function $R(s_t = s, a_t = a) = \mathbb{E}[r_r \mid s_t = s, a_t = a]$
    - Sometimes R is also defined based on (s) or on (s, a, s')
  - Discount factor $\gamma \in [0, 1]$
- MDP is tuple (S, A, P, R, γ)

Automated Machine Learning Hannover

Mars Rover as MRP

s_1	s_2	s_3	s_4	s_5	s_6	s_7
						

- 2 deterministic Actions: TryLeft and TryRight

MDP Policies

- Policy specifies what action to take in each state
  - Can be deterministic or stochastic
- For generality, consider as a conditional distribution
  - Given a state, specifies a distribution over actions
- Policy: $\pi(a \mid s) = P(a_t = a | s_t = s)$

Automated Machine Learning Hannover

MDP + Policy

- MDP + Policy $\pi(a \mid s)$ = Markov Reward Process
- Precisely, it is the MRP (S, R^pi, P^π, γ) where

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

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

- Implies we can use same techniques to evaluate the value of a policy for a MDP as we could to compute the value of a MRP, by defining a MRP with R^π and P^π

Automated Machine Learning Hannover

MDP Policy Evaluation, Iterative Algorithm

- Goal: For a given π, determine V^π
- iterative approach:
  - Initialize $V_0(s) = 0$ for all s
  - For $k = 1$ until convergence
    - For all s in S:

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

- This is a Bellmann backup for a particular policy

Practice: MDP 1 Iteration of Policy Evaluation, Mars Rover Example

- Dynamics: $p(s_6|s_6, a_1) = 0.5, p(s_7|s_6, a_1) = 0.5, \dots$
- Reward: for all actions, $+1$ in state s_1 , $+10$ in state s_7 , 0 otherwise
- Let $\pi(s) = a_1.\forall s$, assume $V_k^\pi = [1, 0, 0, 0, 0, 0, 10]$ and $k = 1$, $\gamma = 0.5$

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

Automated Machine Learning Hannover

Practice: MDP 1 Iteration of Policy Evaluation, Mars Rover Example

- Dynamics: $p(s_6|s_6, a_1) = 0.5, p(s_7|s_6, a_1) = 0.5, \ldots$
- Reward: for all actions, $+1$ in state s_1 , $+10$ in state s_7 , 0 otherwise
- Let $\pi(s) = a_1.\forall s$, assume $V_k^\pi = [1, 0, 0, 0, 0, 0, 10]$ and $k = 1$, $\gamma = 0.5$

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

$$V_{k+1}^\pi(s_6) = 0 + \gamma[p(s_6 \mid s_6, a_1) \cdot V_k^\pi(s_6) + p(s_7 \mid s_6, a_1) \cdot V_k^\pi(s_7)]$$

Automated Machine Learning Hannover

Practice: MDP 1 Iteration of Policy Evaluation, Mars Rover Example

- Dynamics: $p(s_6|s_6, a_1) = 0.5, p(s_7|s_6, a_1) = 0.5, \ldots$
- Reward: for all actions, $+1$ in state s_1 , $+10$ in state s_7 , 0 otherwise
- Let $\pi(s) = a_1.\forall s$, assume $V_k^\pi = [1, 0, 0, 0, 0, 0, 10]$ and $k = 1$, $\gamma = 0.5$

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

$$\begin{aligned}
V_{k+1}^\pi(s_6) &= 0 + \gamma[p(s_6 \mid s_6, a_1) \cdot V_k^\pi(s_6) + p(s_7 \mid s_6, a_1) \cdot V_k^\pi(s_7)] \\
&= \gamma[0.5 \cdot 0 + 0.5 \cdot 10]
\end{aligned}$$

Automated Machine Learning Hannover

AutoML.org

Practice: MDP 1 Iteration of Policy Evaluation, Mars Rover Example

- Dynamics: $p(s_6|s_6,a_1) = 0.5, p(s_7|s_6,a_1) = 0.5, \ldots$
- Reward: for all actions, $+1$ in state s_1 , $+10$ in state s_7 , 0 otherwise
- Let $\pi(s) = a_1 . \forall s$, assume $V_k^\pi = [1,0,0,0,0,0,10]$ and $k = 1$, $\gamma = 0.5$

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

$$\begin{aligned} V_{k+1}^\pi(s_6) &= 0 + \gamma[p(s_6 \mid s_6, a_1) \cdot V_k^\pi(s_6) + p(s_7 \mid s_6, a_1) \cdot V_k^\pi(s_7)] \\ &= \gamma[0.5 \cdot 0 + 0.5 \cdot 10] \\ &= \gamma \cdot 5 \end{aligned}$$

Automated Machine Learning Hannover

Practice: MDP 1 Iteration of Policy Evaluation, Mars Rover Example

- Dynamics: $p(s_6|s_6, a_1) = 0.5, p(s_7|s_6, a_1) = 0.5, \ldots$
- Reward: for all actions, $+1$ in state s_1 , $+10$ in state s_7 , 0 otherwise
- Let $\pi(s) = a_1.\forall s$, assume $V_k^\pi = [1, 0, 0, 0, 0, 0, 10]$ and $k = 1$, $\gamma = 0.5$

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

$$\begin{aligned}
V_{k+1}^\pi(s_6) &= 0 + \gamma[p(s_6 \mid s_6, a_1) \cdot V_k^\pi(s_6) + p(s_7 \mid s_6, a_1) \cdot V_k^\pi(s_7)] \\
&= \gamma[0.5 \cdot 0 + 0.5 \cdot 10] \\
&= \gamma \cdot 5 \\
&= 2.5
\end{aligned} \tag{1}$$

MDP Control

- Compute the optimal policy

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

- There exists a unique optimal value function
- Optimal policy for an MDP in an infinite horizon problem is (i.e. .agents acts forever is)
  - deterministic
  - stationary (does not depend on time step)
  - Unique? Not necessarily, may have state-actions with identical optimal values

Automated Machine Learning Hannover