RL: Basics

Policy Iteration

Marius Lindauer

tnt

Leibniz Universität Hannover

AutoML.org
Automated Machine Learning Hannover

Policy Search (PS)

- One option is searching to compute best policy
- Number of deterministic policies is $|A|^{|S|}$
- Policy iteration is generally more efficient than enumeration

Automated
Machine Learning
Hannover

MDP Policy Iteration (PI)

- Set $i = 0$
- Initialize $\pi_0(s)$ randomly for all states s
- While $i == 0$ or $||\pi_i - \pi_{i-1}||_1 > 0$ (L1-norm, measures if the policy changed for any state)
  - $V^{\pi_i} \leftarrow$ MDP V-function policy evaluation of π
  - $\pi_{i+1} \leftarrow$ Policy improvement
  - $i \leftarrow i + 1$

Definition: State-Action Value Q

- State-action value of a policy

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

$\rightsquigarrow$ Take action a, then follow the policy π

Automated Machine Learning Hannover

AutoML.org

Policy Improvement

- Compute state-action value of a policy π_i
  - For s in S and a in A:

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

Automated Machine Learning Hannover

Policy Improvement

- Compute state-action value of a policy π_i
  - For s in S and a in A:

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

- Compute new policy π_{i+1} for all $s \in S$

$$\pi_{i+1}(s) \in \arg\max_{a \in A} Q^{\pi_i}(s,a). \forall s \in S$$

MDP Policy Iteration (PI)

- Set $i = 0$
- Initialize $\pi_0(s)$ randomly for all states s
- While $i == 0$ or $||\pi_i - \pi_{i-1}||_1 > 0$ (L1-norm, measures if the policy changed for any state)
  - $V^{\pi_i} \leftarrow$ MDP V function policy evaluation of π $\quad \rightsquigarrow$ use Q
  - $\pi_{i+1} \leftarrow$ Policy improvement
  - $i \leftarrow i + 1$

Automated Machine Learning Hannover

Delving Deeper Into Policy Improvement Step

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

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

$$\pi_{i+1}(s) \in \arg\max_{a \in A} Q^{\pi_i}(s,a)$$

- Suppose we take $\pi_{i+1}(s)$ for one action, then follow π_i forever
  - Our expected sum of rewards is at least as good as if we had always followed π_i
- But new proposed policy is to always follow π_{i+1}

Automated Machine Learning Hannover

Monotonic Improvement in Policy

- Definition

$$V^{\pi_2} \geq V^{\pi_1} : V^{\pi_2}(s) \geq V^{\pi_1} . \forall s \in S$$

- Proposition: $V^{\pi_{i+1}} \geq V^{\pi_i}$
  - where π_{i+1} is the new policy we get from policy improvement on π_i
  - with strict inequality if π_i is suboptima

MDP Policy Iteration (PI): Check your Understanding

- Set $i = 0$
- Initialize $\pi_0(s)$ randomly for all states s
- While $i == 0$ or $||\pi_i - \pi_{i-1}||_1 > 0$ (L1-norm, measures if the policy changed for any state)
  - $V^{\pi_i} \leftarrow$ MDP V function policy evaluation of π $\rightsquigarrow$ use Q
  - $\pi_{i+1} \leftarrow$ Policy improvement
  - $i \leftarrow i + 1$
- If policy doesn't change, can it ever change?
- Is there a maximum of iterations of policy iteration?

Automated Machine Learning Hannover