Machine Learning Compendium

ML-KA Karlsruhe

August 29, 2018

Abstract

Looking up things take time. Let's just look into one single document

1 Reinforcement Learning

1.1 Bellman Equations

TODO backup diagramms
State Value Function

$$v_\pi(s) = \sum_{a \in A} \pi(s|a) Q_\pi(s,a) \tag{1}$$

Action Value Function

$$Q_\pi(s,a) = r_s^a + \gamma \sum_{s' \in S} P_{ss'}^a v_\pi(s') \tag{2}$$

State Value Function recursive

$$v_\pi(s) = \sum_{a \in A} \pi(s|a)(r_s^a + \gamma \sum_{s' \in S} P_{ss'}^a v_\pi(s')) \tag{3}$$

Action Value Function recursive

$$Q_\pi(s,a) = r_s^a + \gamma \sum_{s' \in S} P_{ss'}^a \sum_{a \in A} \pi(a'|s') Q_\pi(s',a') \tag{4}$$

Optimal State Value Function

$$v_*(s) = \max_a Q_*(s,a) \tag{5}$$

Optimal Action State Value Function

$$Q_*(s,a) = r_s^a + \gamma \sum_{s' \in S} P_{ss'}^a v_*(s') \tag{6}$$

Optimal State Value Function recursive

$$v_*(s) = \max_a r_s^a + \gamma \sum_{s' \in S} P_{ss'}^a v_*(s') \tag{7}$$

Optimal Action State Value Function recursive

$$Q_*(a,s) = r_s^a + \gamma \sum_{s' \in S} P_{ss'}^a \max_{a'} Q_*(s',a') \tag{8}$$

1.2 Advantage Function

TODO

1.3 Policy, Policy Gradient

Policy: Distribution over actions given states

$$\pi_\theta(a|s) = P(a|s) \tag{9}$$

Policy Gradient

$$\nabla_\theta \pi_\theta(s|a) = \pi_\theta(s|a) \nabla_\theta \log \pi_\theta(s|a) \tag{10}$$

Note: this is valid for all probability distributions (the policy is a distribution over actions given states). The gradient term on the right hand side is called score function. The derivation basically uses the "log-trick".