Reinforcement Learning2

출처: http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html

Index

Model-based RL Methods - Dynamic Programming

Optimal policy를 구하는 방법

Policy Evalution and Improvement steps

1. 먼저 임의로 초기화된 policy π_0가 있고, state value function v_0를 모든 state s에 대해 0의 값을 갖게 초기화하자.
2. 그렇다면 π_0에 대해서 state value function을 업데이트할 수 있다.
    - MRP에서 구했던 것처럼 Bellman equation을 이용해 구할 수 있다.
    - 이를 state value function v_1이라 하자(formally v_{π_0})
3. v_1을 이용하여 greedy하게 policy π_1를 구할 수 있다.
    - $\pi_1 = greedy(v_1)$
    - $\pi_1(s) = \text{argmax}_{a \in A}\, q_\pi(s,a)$
4. π_k와 v_k가 수렴할 때까지 이를 반복한다.

이에 대한 π_k와 v_k수렴성은 보장되어 있고 이는 π_*, v_*로 수렴한다.

Value Iteration

1. state value function v_1을 모든 state에 대해 0으로 초기화한다.
2. 수렴할 때까지
    - $v_{new}(s) \leftarrow \max_{a \in A}(R_s^a + \gamma \sum_{s' \in S} P_{ss'}^a v_{old}(s'))$

      를 반복한다.

Policy iteration과는 다르게 명확한 policy가 정의되지 않기 때문에, 중간에 있는 state value function에 해당하는 policy iteration은 존재하지 않을 수도 있다.

Model-free RL Methods

Model-based RL methods에서는 환경에 대해 모든 것을 알고 있고 이를 이용하여 policy와 state-value-function을 구했다. 하지만 많은 경우 환경에 대한 정보가 충분하게 주어지지 못한다. 이러한 상태를 model-free라고 하며 MDP transition이나 reward에 대해 전혀 아는 것이 없는 상태이다. 하지만 여러 실험을 통해 $v_\pi(s), q_\pi(s,a)$ 혹은 $\pi(s,a)$를 추정할 수 있다면, optimal state value function과 policy를 추정할 수 있을 것이다.

$v_\pi(s), q_\pi(s,a)$를 추정하여 optimal policy를 추정하는 방법에는 대표적으로 Monte-Carlo Learning와 Temporal-Difference이 있다.

$\pi(s,a)$를 직접 추정할 수도 있는데 이를 policy gradient라고 하며, NLP에서는 시퀀스 생성 작업에 자주 쓰이는 방법이다.