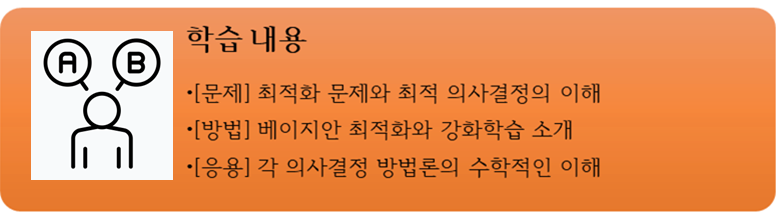
# Chapter 6. 기계학습 기반 의사결정 및 최적화



## 최적화 문제와 최적 의사결정의 이해

주어진 문제상황에서 제약 조건(constraints: )들을 만족시키면서 목적함수(objective function: 를 최대화 (혹은 최소화) 시키는 방법을 찾아내는 것을 최적화 문제라고 한다. 이는 수학적으로 식(1)과 같이 표현될 수 있다.

|  |
| --- |
| (1) |

여기서 보편적으로 는 종속변인 *y*와 조작변인 *x*의 관계를 지정하는 일종의 모델이다. 본 장에서는 의 형태를 모르는 경우, 즉 주어진 모델이 없는 경우의 최적화에 대해서 알아볼 예정이다.

화학공학에서는 신 공정조건 탐색, 신 물질 탐색, 공정 제어 등의 상황에서 최적화 문제를 풀게 된다. 특히 화학공학에서는 최적화 문제를 풀 때 효율적이고 조심스러운 접근 방법이 요구되는데, 이는 화학공학에서의 실험이 많은 시간과 비용이 소요되는 경우가 일반적이기 때문이며, 또한 특정 실험의 경우 위험한 결과를 초래할 수 있기 때문이다. 따라서 화학공학에서의 기계학습 기반 의사결정 및 최적화를 진행할 때는 trend 모델을 이용하여 경향을 분석, 즉, 상관관계(correlation) 규명에 집중하는 것이 아닌 cause-effect 모델을 사용하여 인과관계(causation)을 파악하는 것이 중요하다. 이러한 특성 때문에 화학공학에서의 의사결정에는 탐험-탐사(exploration-exploitation) 간의 최적 균형을 이룬 의사결정방법이 필요하다. 본 장에서는 이러한 의사결정 방법 2가지를 학습한다. 첫번째 방법론은 베이즈 정리를 활용하여 탐험-탐사 간의 균형을 맞춘 베이지안 최적화이고, 두번째 방법론은 무수히 많은 경험을 통한 시스템 이해를 바탕으로 탐험-탐사 간의 균형을 맞춘 강화학습이다.

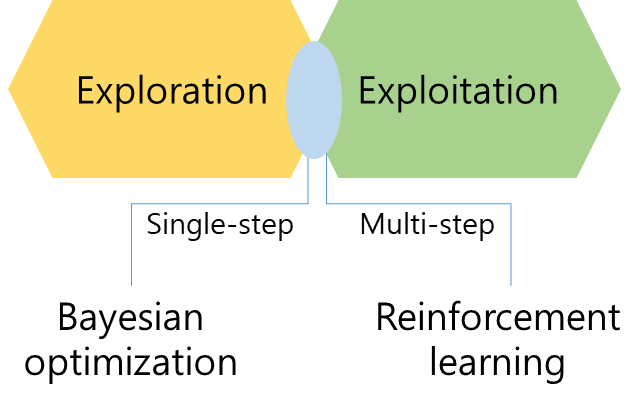


Figure 탐험-탐사 간의 균형을 맞춘 기계학습 방법론

### 베이지안 최적화

베이지안 최적화는 조작변인 과 종속변인 사이의 관계가 매우 복잡한 상황에서 모델 기반 의사결정이 불가능한 경우에 (black-box 시스템), 실험의 횟수를 최소화하면서 원하는 종속변인의 원인이 되는 조작변인 들을 규명할 때 주로 사용되는 의사결정 방법론이다. 특히 초기 데이터가 상대적으로 적거나 실험 시간이 오래 걸리는 경우, 적은 데이터에 기반한 이해를 바탕으로 탐사-탐험 간의 균형을 맞춘 후, 실험 포인트를 결정한다.

### 강화학습

강화학습은 마르코프 결정 과정(MDP, Markov Decision Process)로 표현된 순차적 의사 결정 문제를 풀 때 사용되는 방법론이다. 어떠한 순차적 의사 결정 과정이 그 당시 상태에만 영향을 받고, 그 이전까지의 상태에 무관 하다면, 이러한 의사결정과정은 ‘마르코프 하다’ 라고 정의된다. 수학적으로 MDP는 상태(), 행동(), 상태 변환 확률(), 보상(), 감가상각률()의 5중쌍으로 표현되며, 강화학습은 에이전트가 MDP로 정의된 환경과의 상호작용을 통해 최적의 의사 결정 정책을 찾아 나가는 방법론이다.

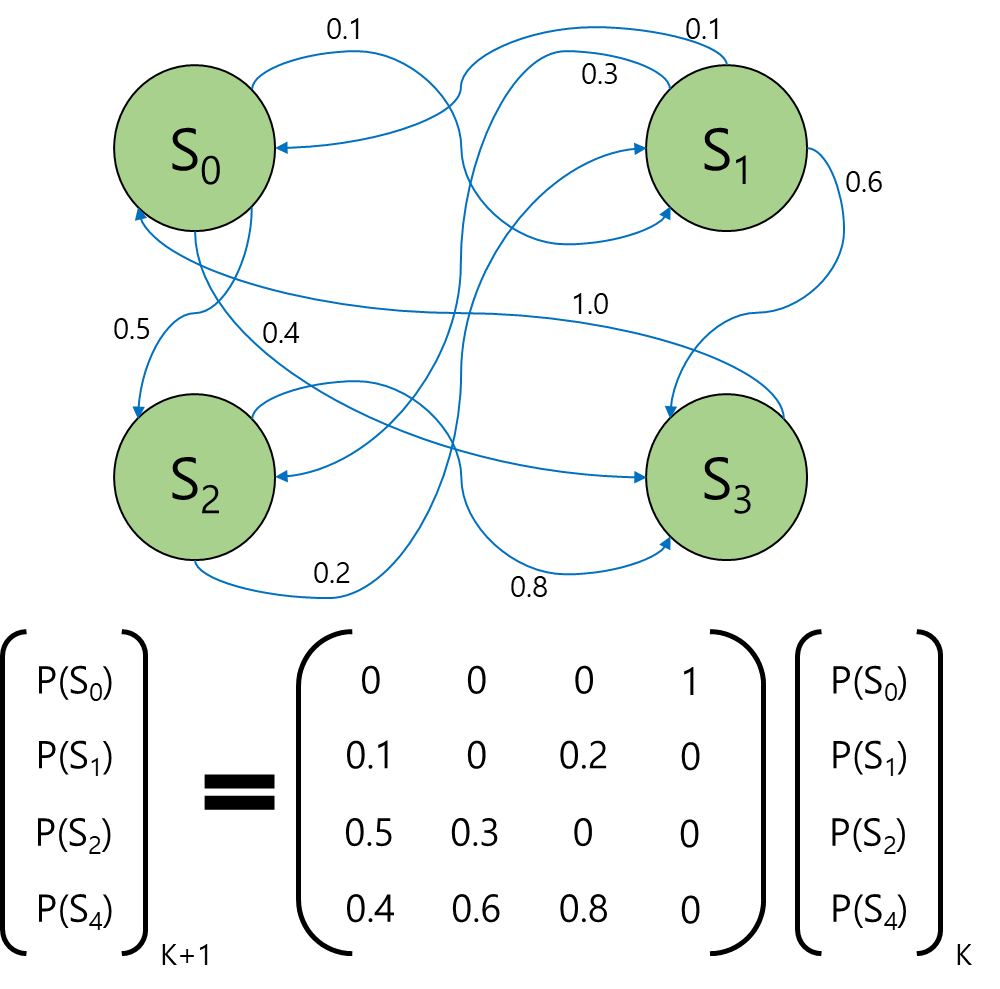


Figure 간단한 마르코프 결정 과정의 예시 ((P(Si))k here is the probability of the state being Si at time step k.

## 알고리즘과 작동 원리

2장에서는 베이지안 최적화와 강화학습의 알고리즘과 작동 원리를 학습한다. 이를 통해 두 최적화 방법론에 대한 이해의 깊이를 넓힌다.

### 베이지안 최적화 알고리즘

베이지안 최적화는 어떠한 조작 변인 가 주어졌을 때, 어떠한 종속 변인 (목적 함수) 가 얻어지는 시스템에서, 시스템이 복잡하여 모델을 만들기가 어렵고, 매 실험에 소요되는 시간과 비용이 많이들 때 (black-box 시스템) 사용되는 최적화 방법론이다. 이때, 최적화라 함은 목적 함수 를 극대화시킬 수 있는 특정 조작 변인 를 찾는 것이다. 이를 달성하기 위해선 기존에 관찰된 조작 변인과 목적 함수 사이의 데이터를 바탕으로 대체 모델 (surrogate model)을 생성하여 의 형태를 확률적으로 추정한 후, 대체 모델을 바탕으로 획득 함수 (acquisition function)를 정의하여 최적 조작 변인 을 찾을 가능성이 가장 높은 다음 실험 후보를 얻는다. 이러한 추천을 통해 얻어진 점을 실험하고, 추가된 데이터를 기반으로 다시 대체 모델을 업데이트 하는 과정을 반복하는 것이 베이지안 최적화의 구성이다.

#### 대체 모델 (surrogate modeling)

베이지안 최적화의 첫 단계는 대체 모델을 만드는 것이다. 베이지안 최적화는 조작 변인 와 목적 함수 사이의 관계를 모를 때 사용되는 방법론이기에, 와 사이의 관계를 확률적으로 추정하는 대체 모델을 구축하여 실험되지 않은 구간의 예측치와 불확실성을 표현한다. 실험되지 않은 구간의 예측치와 불확실성을 표현하기 위해서 가장 자주 쓰이는 데이터 기반 모델링 방법은 가우시안 프로세스 (Gaussian process) 이다. 가우시안 프로세스는 기존에 관찰된 점들을 관통하는 모든 그래프의 가능성을 인지하고, 이를 바탕으로 실험되지 않은 지점에서의 예측값은 정규분포의 평균, 예측값의 불확실성은 정규분포의 표준편차로 나타내는 방법이다. 가우시안 프로세스는 우리에게 친숙한 확률분포인 정규분포를 사용하기에 직관적으로 시스템을 이해할 수 있고, 추후 획득함수를 정의하기도 용이하여서 자주 사용된다.

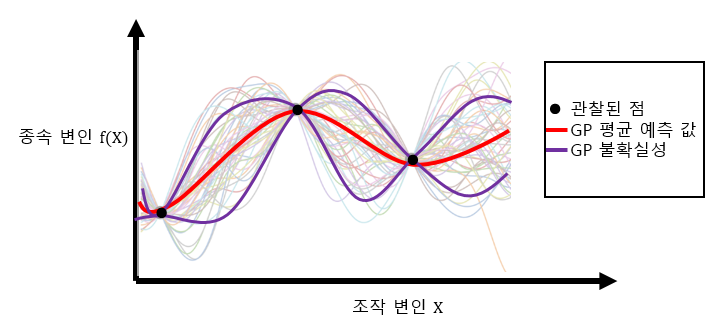


Figure 3 가우시안 프로세스 예시

#### 획득 함수 (acquisition function)

기존에 관찰된 데이터를 바탕으로 대체 모델을 생성하여 전체적인 시스템에 대한 이해가 완료되었다면, 그러한 이해를 바탕으로 어떠한 점을 다음 실험 포인트로 잡을지를 결정해야 한다. 다음 실험 포인트는 단순히 예측치의 최댓값을 실험하는 것이 아닌, 예측치와 그 예측치가 얼마나 신뢰할 만한 지에 기반하여 종합적인 의사결정이 요구된다. 목적 함수를 최대화시키는 문제에서, 예측치가 높기 때문에 개선을 기대할 수 있는 점을 실험하는 것도 중요하고(exploitation), 혹은, 예측치는 낮아도 불확실성이 큰 지점에서 새로운 개선(exploration)을 기대하는 것이 중요하기 때문이다. 베이지안 최적화는 이러한 종합적인 사고를 이뤄낼 수 있는 획득함수를 정의하고, 획득함수의 값이 가장 높은 지점을 그 다음 실험 포인트로 설정한다. 자주 쓰이는 획득 함수로는 PI (probability of improvement), EI (expected improvement), 그리고 UCB (upper confidence bound) 등이 있다. 각각의 획득 함수는 Table 1과 같이 수학적으로 표현할 수 있다. 직관적인 이해가 편한 UCB의 경우, 그림 4와 같이 예측치와 표준편차의 합이 최대가 되는 지점을 다음 실험에 추천한다. 주로 초기 데이터가 많은 경우에는 exploitation을 잘 수행하는 PI 혹은 UCB 기반 베이지안 최적화를 자주 사용하고, 초기 데이터가 많지 않은 경우에는 exploration을 잘 수행하는 EI 기반 베이지안 최적화를 자주 사용한다. 이에 더해, 최근에는 미래의 의사결정까지 고려하여 현재의 의사결정을 가이드 하는 획득함수가 새롭게 개발되고 있다. 이러한 기법으로는 롤 아웃 기반 혹은 강화학습 기반 베이지안 최적화가 존재하며 기존의 PI, EI, UCB 기반의 베이지안 최적화보다 좋은 성능을 보이고 있다.

|  |  |
| --- | --- |
| **Acquisition function** | **Formula** |
| Probability of improvement |  |
| Expected improvement |  |
| Upper confidence bound |  |

**Table 1 베이지안 최적화의 획득힘수의 예**

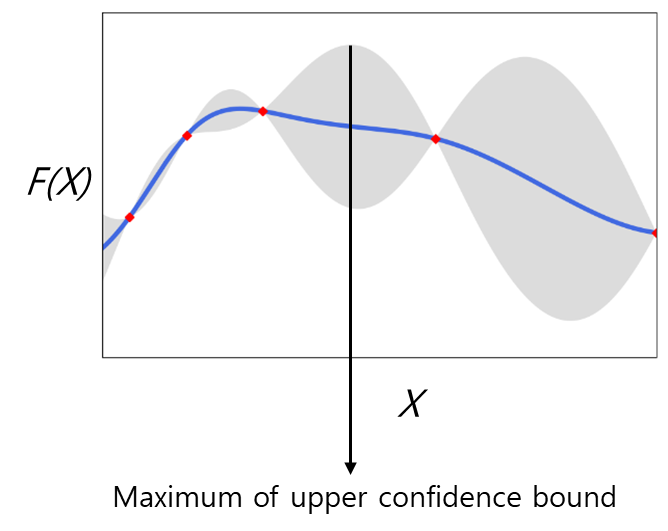
****

Figure 4 Upper confidence의 시각적 예시

#### 주요 문제 해결 과정

베이지안 모델을 이용한 최적화 수행 과정은 Figure 5 와 같다. 초기에 보유하고 있는 조작변인 와 종속변인 (목적함수) 사이의 관계 (데이터)를 활용하여 대체 모델을 형성을 한 후, 생성된 대체 모델을 바탕으로 획득함수를 정의한다. 이때, 어떠한 획득 함수를 사용할지는 해당 시스템에 대한 이해도에 따라서 결정한다. 획득함수를 시스템에 맞게 정의를 하였다면, 획득함수를 최대화시키는 조작변인을 찾아낸 후, 이를 다음 실험 포인트로 설정하여 실제 실험을 진행한다. 이러한 실험을 통해서 얻어진 결과값이 사용자의 기대(요구조건)를 만족하였다면 실험이 종료되는 것이고, 결과값이 사용자의 기대를 만족하지 못하였다면, 얻어진 새로운 데이터를 기반으로 다시 대체 모델을 생성을 하고 처음부터 모든 과정을 순차적으로 반복한다.

밤하늘이(가) 표시된 사진

자동 생성된 설명

Figure 5 베이지안 최적화의 알고리즘

### 강화학습 알고리즘

강화학습은 행동심리학에서 사용되는 용어인 ‘강화’를 기계학습에 접목한 방법론이다. 몇 년 전에 큰 화제가 되었던 알파고에서 사용된 기계학습이 바로 강화학습이다. 강화란 행동의 주체 (이를 강화학습에서는 ‘에이전트’ 라고 부른다)가 주어진 환경에서 여러 행동을 시행한 뒤, 행동에서 나온 보상을 통해 학습을 하여, 최적의 의사결정 방식을 찾아 나가는 방법론이다. 어릴 적 스케이트 보드 타는 방법을 배우는 과정을 연상하면 ‘강화’의 개념을 이해하기가 더 쉽다. 현재의 속도, 도로의 모양, 바람의 세기, 중력의 세기 등이 주어진 환경에서, 스케이트 보드를 처음 타 보는 아이는 무게 중심의 위치, 발을 구르는 세기 등을 바꾸어 본다. 특정 행위는 보드가 잘 나아가게끔 하고, 특정 행위는 보드가 쓰러지게끔 유도를 한다. 이렇게 환경과의 상호작용을 통해서 학습을 하여 결국 최적의 행동 정책을 찾아가는 것이 ‘강화’이다. 이러한 과정과 같이 에이전트가 특정 환경에서의 최적 행동 정책을 찾아내는 것이 강화학습이다.

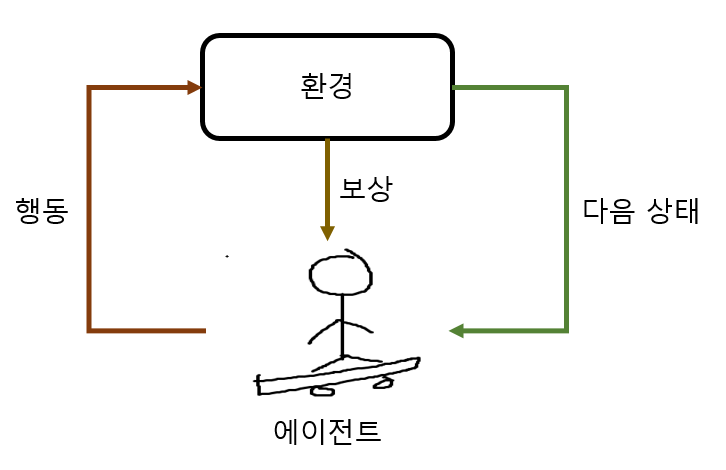


Figure 6 스케이트 타는 아이와 강화학습

이러한 기계학습을 진행하기 위해선 첫번째로 환경이 제시하는 ‘상태’를 수학적으로 정의하여 에이전트가 이해를 할 수 있게끔 해야 한다. 두번째로는 상태를 인식한 에이전트가 스스로의 정책에 따라 행동을 해야 한다. 세번째로는 행동을 바탕으로 환경이 에이전트에게 수학적으로 정의된 보상과 다음 상태를 전달하여야 한다. 이러한 과정을 반복하는 것이 강화학습의 전체적인 알고리즘이다.

#### 순자적 의사결정 문제와 MDP

2.2장에서 논의된 바를 생각해본다면, 강화학습을 사용할 수 있는 문제의 두가지 특징을 생각해낼 수 있다. 첫번째로는 순차적 행동 결정 문제에서 사용된다는 점이다. 에이전트가 환경과 꾸준히 상호작용을 해 가면서 최적의 행동 양식(혹은 최적 정책 )을 찾아가는 방식이기 때문이다. 또한 두번째 특징으로는 현재 ‘상태’만 행동에 영향을 미친다는 것이다. 이러한 성질을 ‘마르코프 하다’라고 표현하는데, 수학적으로는 조건부 확률을 사용하여 다음과 같이 나타낼 수 있다. . 즉, 과거가 어떠 하였든, 현재의 상태만 미래에 영향을 주는 것이다. 마르코프한 순차적인 의사결정 문제는 마르코프 결정 과정 (MDP)으로 표현할 수 있다. 따라서, 어떠한 문제가 MDP로 정의가 될 수 있어야 강화학습으로 풀 수 있는 문제인 것이다.

MDP는 5가지의 요소로 구성이 되는데, 이는 다음과 같다: . 첫번째로 는 에이전트가 현재 스스로의 상태를 인식할 수 있는 상태(state)의 집합이다. 강화학습 에이전트는 상태를 기반으로 의사결정을 내리기 때문에, 수학적으로 정의된 상태는 에이전트가 의사결정을 내리기 위한 충분한 정보를 갖고 있어야 한다. 따라서, 상태를 의사결정에 모자란 부분 없이 정의해서 에이전트가 학습할 수 있도록 하는 것이 매우 중요하다. 두번째로 는 행동(action)의 집합이다. 에이전트가 할 수 있는 모든 행동들의 집합이다. 강화학습의 최종 목표는, 임의의 상태에서 현재와 미래의 보상을 고려한 가장 현명한 행동을 할 수 있는 최적 정책을 찾는 것이다. 세번째로 는 상태 변환 확률이다. 특정 상태에서 특정한 행동을 했을 때, 어떠한 상태로 넘어가게 될 지를 정하는 기반이 이다. 네번째로 은 보상(reward)이다. 강화학습은 현재와 미래의 보상을 최대화하는 최적의 행동 정책을 찾는 것이기에, 보상이 엄밀하게 정의되어야 어떤 행동이 옳고 그른 지를 에이전트가 배울 수 있다. 마지막으로 는 할인율(discount rate)이다. 현재의 행동에서도 보상은 나오고, 미래의 행동에서도 보상은 나온다. 이때, 현재 혹은 현재와 가까운 시점에서 나오는 보상에 더욱 중점을 둔 의사결정을 할 것인지, 아니면 미래에 궁극적으로 받게 될 보상에 더욱 중점을 둘 것인지 여부가 할인율을 통해 결정된다. 이는 은행에서 말하는 이자율과도 같은 개념이다. 할인율이 높으면 미래에 얻을 보상보단 근래에 얻게 될 보상에 더 중점을 둔 의사결정을 의미한다.

#### 가치 함수와 벨만 방정식

강화학습 에이전트가 최적의 정책을 찾는다는 것은, 시작할 때의 상태부터 한 에피소드가 끝날 때까지의 보상합을 최대화시키는 행동을 하게 된 다는 것이다. 이때 에피소드는 시작 상태로부터 종결상태(terminal state)까지의 에이전트와 환경의 상호작용을 의미한다. 현재 시점(T=*t*)에서의 상태를 , 행동을 , 그때 받게 되는 보상을 로 표현했을 때, 하나의 에피소드가 끝날 때까지 받게 되는 모든 보상의 합은 할인율을 적용하여 으로 표현할 수 있다. 이렇게 에피소드를 진행함에 따라 미래에 받게 될 모든 보상의 합을 반환값(return)이라고 표현한다. 어떠한 상태에 진입했을 때 어떤 반환값을 받게 되는지를 알 수 있다면, 에이전트는 최적의 의사결정을 내리도록 학습될 수 있다. 따라서, 어떠한 상태가 얼마만큼의 가치가 있는지를 가치함수(value function)을 정의하여 계산한다. 수식적으로는 식 (2)과 같이 표현할 수 있다

|  |  |
| --- | --- |
|  | (2) |

막연할 수 있는 ‘미래의 보상’을 조금 더 정확히 표현하기 위하여 가치함수를 재귀 방정식 형태로 나타낼 수 있다. 여기서 에이전트는 특정 상태에서 특정 행동을 취하는 정책를 기반으로 행동을 하기에, 최종적으로는 가치함수를 식 (3)와 같이 나타낼 수 있다. 이 식이 강화학습에서 가장 중요한 개념인 벨만 방정식이다. 이 방정식을 풀어서, 이 가치함수를 최대화할 수 있는 최적 정책 를 찾아내는 것이 바로 강화학습 에이전트를 트레이닝 하는 것이다.

|  |  |
| --- | --- |
|  | (3) |

#### MC(몬테카를로) 학습과 TD(시간차) 학습

이전 챕터에서 특정 상태()에서 정책()을 따라 행동했을 때의 가치를 나타내는 벨만 방정식에 대해서 이야기해 보았는데, 구체적으로 가치 함수를 추정하는 방식에 따라서 MC 학습법과 TD 학습법으로 나눌 수 있다. MC 학습은 몬테카를로 방법을 사용하여 가치함수를 추정한다. 몬테카를로 방법 혹은 몬테카를로 알고리즘은 특정한 상태에서 에피소드를 따라 종결상태까지 의사결정을 진행해본다. 종결상태까지 진행하였기에, 각 상태에서의 반환값(return)을 얻을 수 있다. 이러한 과정을 여러 번 반복하다 보면, 특정 상태에서의 반환값이 여러 번 구해진다. 만약 무수히 많은 반복을 통하여 여러 번 구해진 반환값들의 평균을 낸다면, 그 평균값은 가치함수의 참값에 한없이 가까워진다. 이러한 방식을 통해서 가치 함수를 업데이트 하는 것을 수식으로 표현하면 식 (4)과 같다.

|  |  |
| --- | --- |
|  | (4) |

MC 학습법은 직관적이고 효용이 높은 학습법이지만, 몇가지의 단점이 있다. 크게 두가지의 문제를 꼽자면, 첫째로 학습을 위해서는 한 에피소드의 끝까지 기다려야 하기에 실시간 학습이 불가능 하다는 것이고, 둘째로 에피소드가 끝나지 않고 계속 지속되는 경우 학습이 불가능하다는 점이다. 따라서 이러한 문제점을 보완한 정책평가 방식이 바로 TD(시간차) 학습이다.

TD 학습법은 에피소드가 끝날 때까지 기다린 후에 가치함수를 업데이트 하는 것이 아닌, 에피소드가 진행되는 와중에 매 의사결정 스텝마다 가치함수를 업데이트 한다. 매 스텝마다 업데이트를 하기 때문에 한 번에 하나의 가치함수만을 업데이트 하는 것이 MC 방법과의 가장 큰 차이이다. 실시간으로 업데이트가 이루어지기 때문에, 가치함수를 업데이트 할 때, 항이 아닌, 항을 사용하여 식 (5)와 같이 업데이트를 하게 된다.

|  |  |
| --- | --- |
|  | (5) |

대표적인 TD학습의 알고리즘으로는 살사 알고리즘과 큐러닝이 존재한다. 이 둘의 차이는 학습하는 정책과 실제로 행동하는 정책이 같은 지의 여부이며, 그 두개가 같으면 살사 알고리즘, 다르면 큐러닝이다.

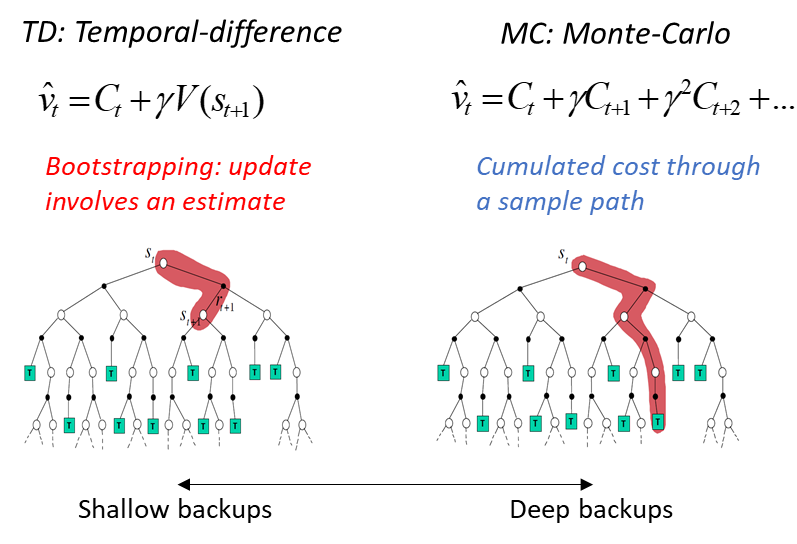


Figure 7 TD 러닝과 MC 러닝

## 알고리즘 선택

본 장에서 소개한 베이지안 최적화와 강화학습은 각 방법론이 어떤 원리를 가지고 있고, 어떠한 문제 상황에 사용될 수 있는지에 대한 감을 잡을 수 있도록 소개되었다. 화학공학 산업에서 디지털 전환(digital transformation)이 이루어짐에 따라, 효율적인 의사결정을 해 낼 수 있는 베이지안 최적화와 강화학습의 효용이 더 커지고 있다.

베이지안 최적화의 경우, 회분 베이지안 최적화, Rollout기반 베이지안 최적화, 강화학습 기반 베이지안 최적화 등의 기법이 최근에 연구가 되고 있다. 강화학습은 크게 가치함수(value function)를 학습하는 방법, 정책(policy)을 학습하는 방법, 또 두가지를 같이 하는 방법 (actor-critic)으로 구분할 수 있는, PPO. DDPG, 베이지안 강화학습 등의 기법이 새로 등장하면서 효율적인 의사결정을 돕고 있다 (Figure 8).

기계학습의 핵심은 주어진 문제의 상태 (복잡성, 데이터 확보 여부, 문제 해결 수준 등)를 명확하게 인식하고, 그러한 문제상황에 적합한 알고리즘을 선정하는 과정이 가장 중요하다. 즉, 문헌 및 인터넷 상의 다양한 라이브러리 혹은 기본 모델 등을 이용하여 대상 문제에 적합한 최적의 알고리즘을 개발하는 능력이 미래의 화학공학자들에게 가장 중요한 요소이다. 그림 8을 통해 모델의 유무, 학습의 방법에 따른 알고리즘들의 종류를 이해할 수 있다.

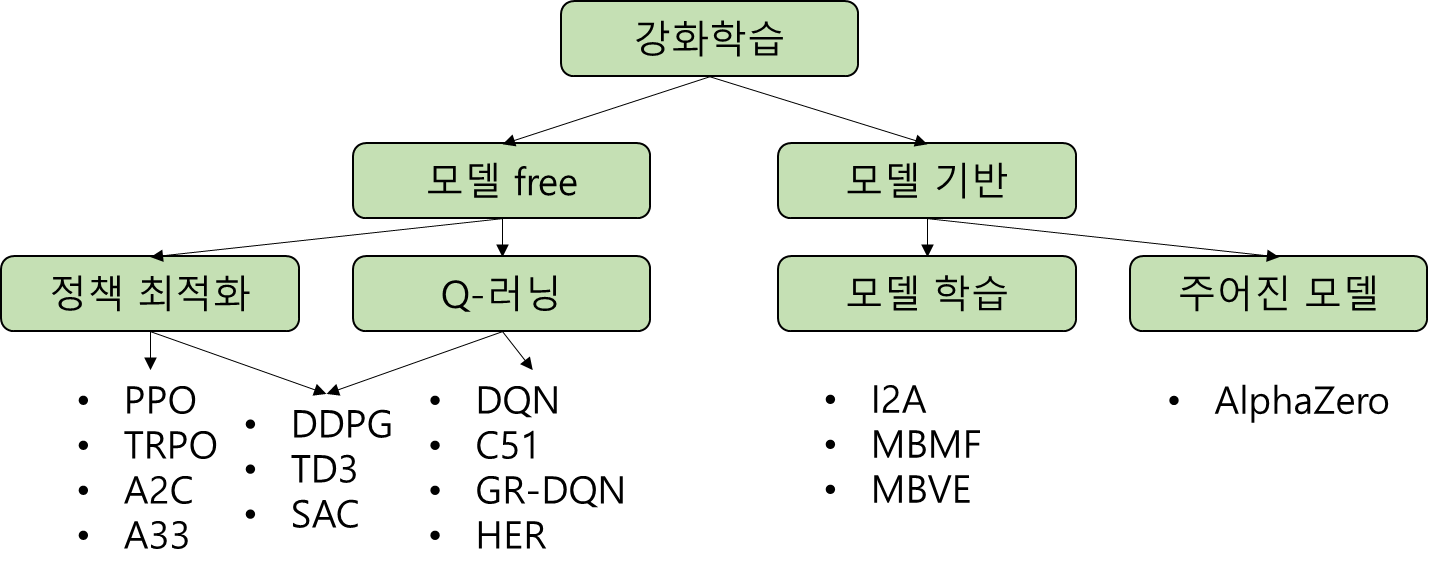


Figure 8 강화학습 알고리즘의 종류

## 결론

본 장에서는 독립 변수와 종속 변의 관계가 명확치 않는 복잡계 문제에서 적용가능한 두 가지 (베이지안 최적화, 강화기계학습) 최적화 방밥론에 관하여 학습하였다. 베이지안 최적화는 실험 혹은 샘플링 한번 한번에 비용과 시간이 많이 드는 black-box 최적화에 사용되는 의사결정 방법론이고, 강화학습은 에이전트가 환경과 상호작용을 통하여 현재와 미래를 모두 고려한 의사결정을 해야 할 때 사용하는 의사결정 방법론이다. 본인이 풀고자 하는 문제의 특성을 파악한 후, 정확한 기계학습 기법을 사용하여 효율적인 의사결정을 수행하는 것이 중요하다.

## 학습 결과

* 학습 내용

최적화 문제에 대한 이해와 화학공학에서 이루어지는 의사결정 문제의 형태에 대한 이해하기.

* 학습 결과 확인하기

베이지안 최적화와 강화학습이 어떠한 종류의 의사결정에 사용되는지에 대한 이해하기.

* 학습 결과 응용하기

베이지안 최적화와 강화학습의 수학적인 이해와 어떠한 방식의 알고리즘이 있는지를 이해함으로써 다양한 환경의 의사결정에서 상황에 맞는 알고리즘을 선택하여 본인의 환경에 맞게끔 응용.