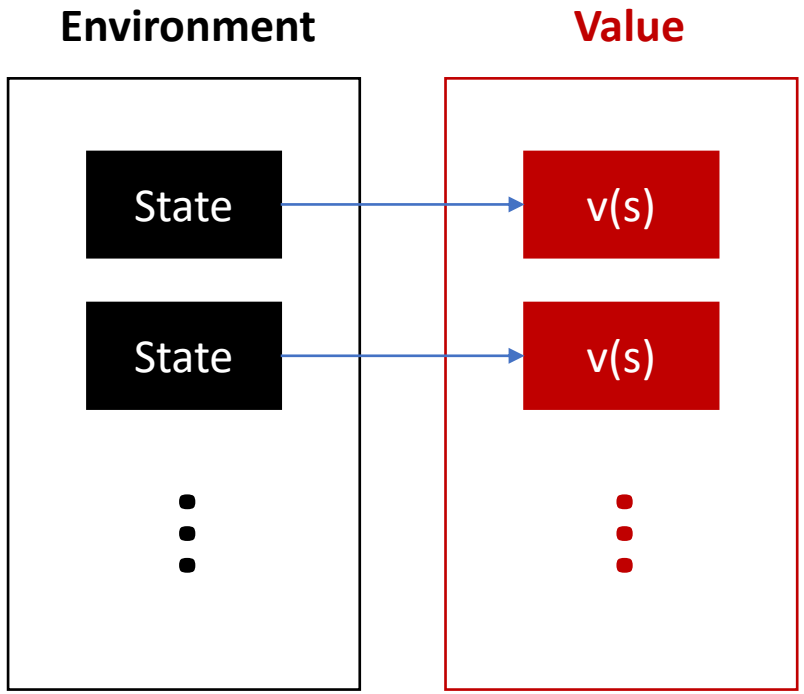
Chap.7
Deep RL

Wonseo.Choi

상태 개수가 무수히 많은 커다란 MDP : 문제 공간이 매우 큼



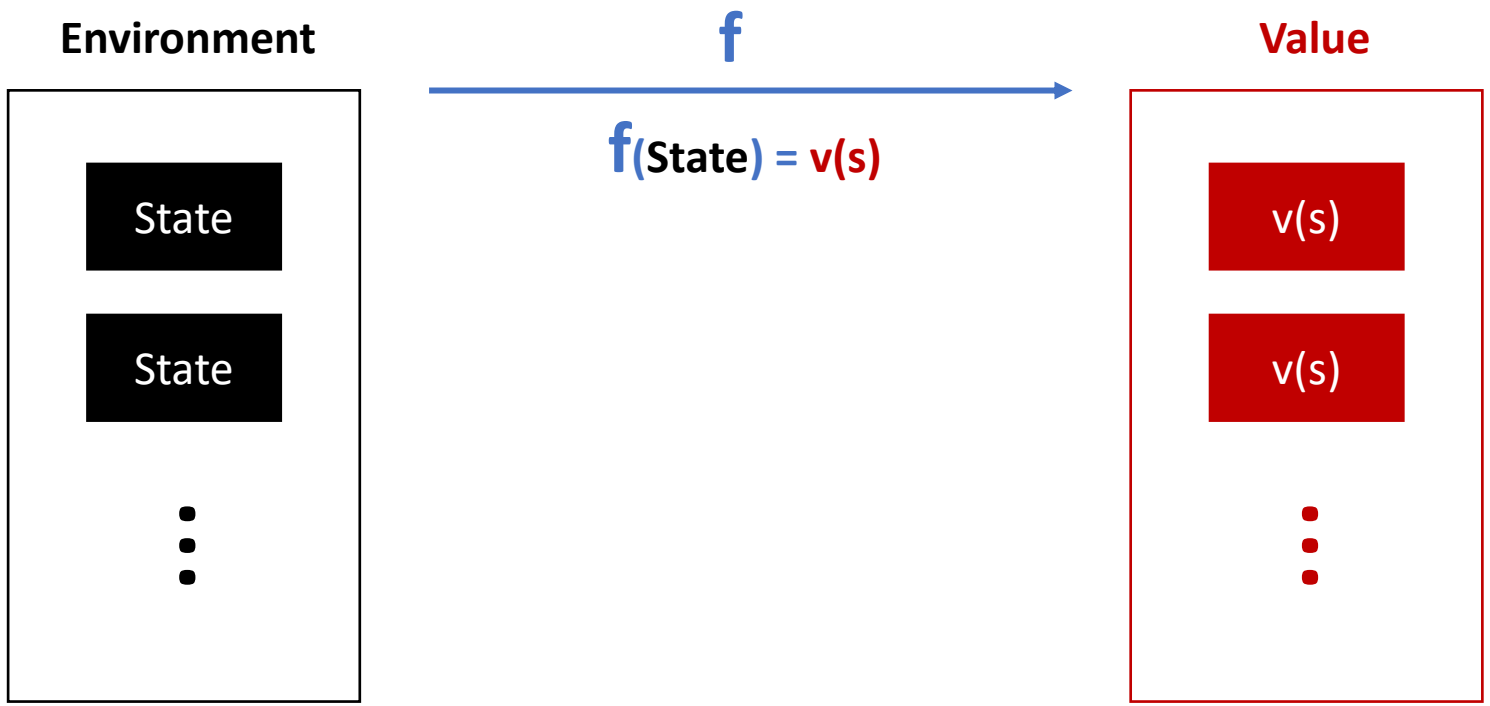
환경 변수가 매우 많은 경우 또는 셀 수 없는 경우

Discrete State Space
체스 : 10^{47}
바둑 : 10^{170}

Continuous State Space
속도 ($\mathbb{R}^3$)

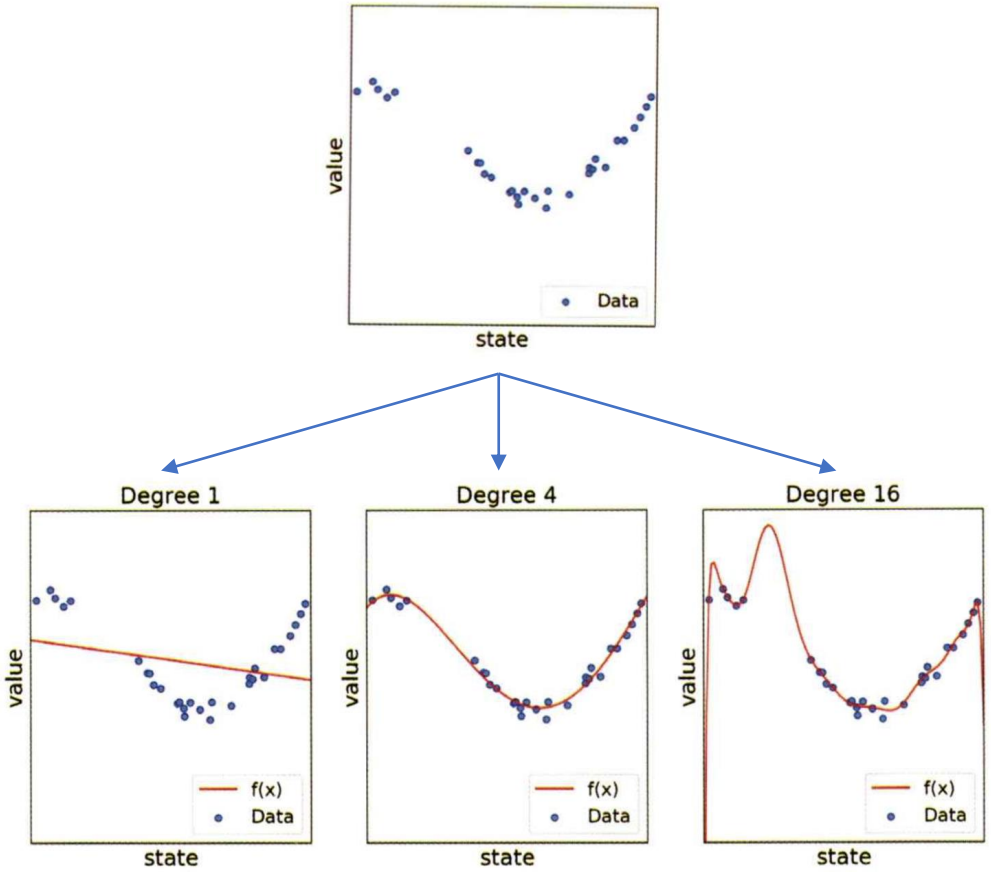
이트하려면 모든 칸을 다 방문해야 합니다. 10^{170}칸에 있는 숫자들을 의미 있는 값으로 업데이트하려면 10^{170}의 상태 각각을 여러 번 방문해야 합니다. 정리하면 필요한 경험의 숫자가 너무나 큽니다. 컴퓨터의 성능이 좋아서 1초에 1억 개 $=10^8$개의 상태를 방문한다고 해도 10^{162}초가 필요합니다.
1년이 약 $3*10^7$초(=넉넉하게 10^8초)이기 때문에 총 10^{154}년에 해당하는 시간입니다. 이는 우주가 10^{144}번 없어졌다가 다시 탄생하는 데 필요한 시간입니다. 요컨대 이런 방식으로는 모든 상태의 값을 추정할 수 없으므로 다른 접근법이 필요합니다.

함수를 도입



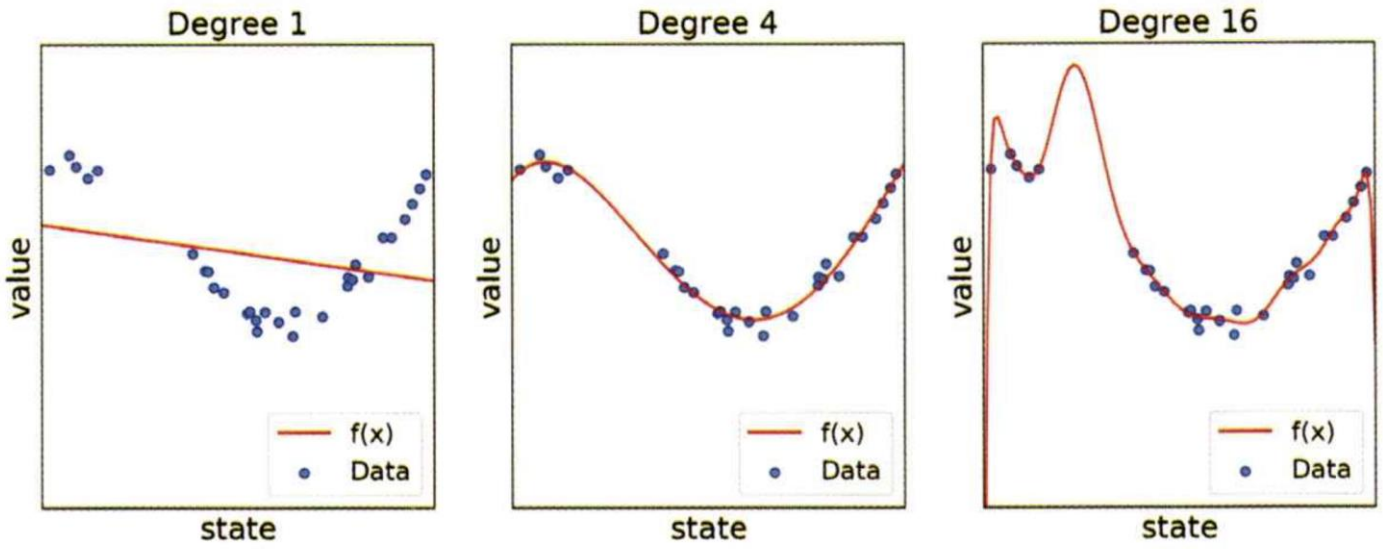
함수 구하기

Fitting



0. 함수의 종류를 정하기 (polynomial function)
1. 함수에 데이터를 기록하기
2. 함수의 파라미터 값을 찾기
   - MSE (Mean Squared Error)

Underfitting Overfitting



Underfitting

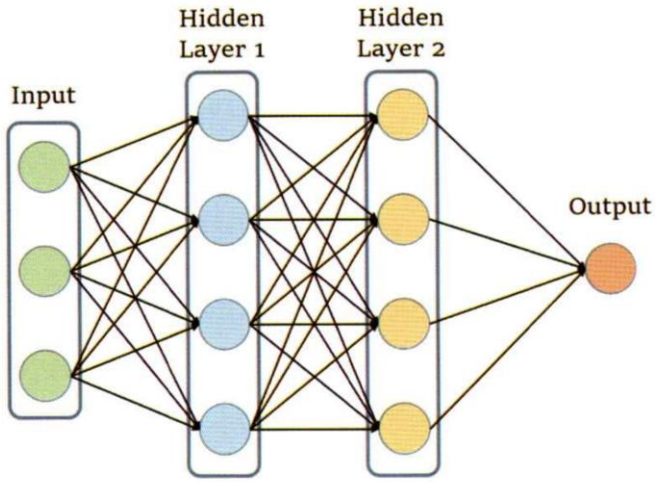
free-parameter 많이 없음

Overfitting

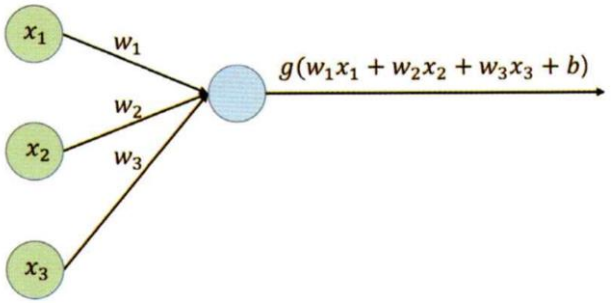
free-parameter 너무 많음

function의 자유도를 적절히

인공 신경망



| 그림 7-7 | 인공 신경망 개요도



| 그림 7-8 | 하나의 노드에 대한 그림

1) 선형결합 (Linear Combination)
   - 새로운 Feature 생성

2) 비선형 함수 (Non-Linear Activation)
   - 추상화 된 Feature 생성
   - Sigmoid, ReLu

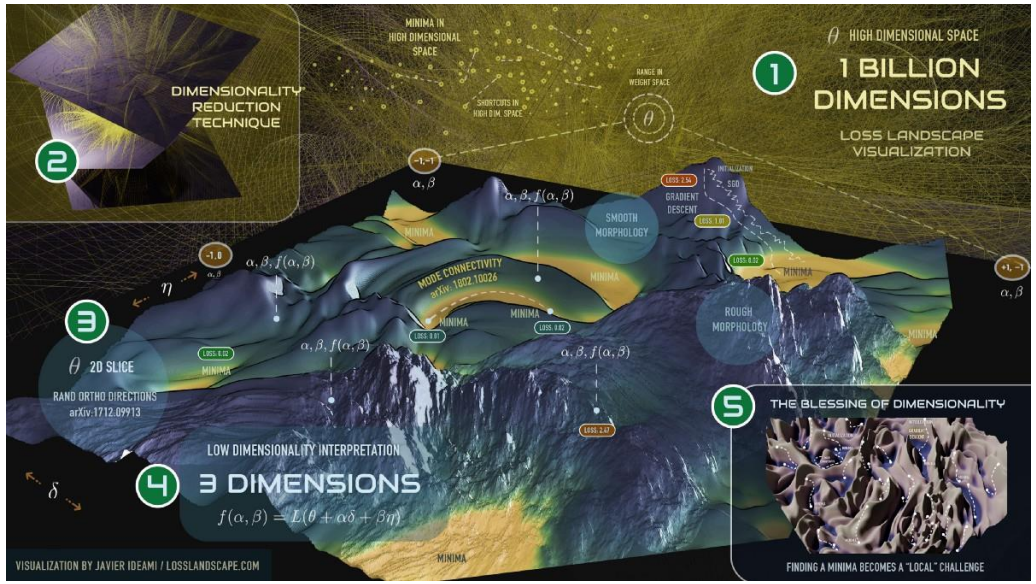
Optimizer

Gradient Descent

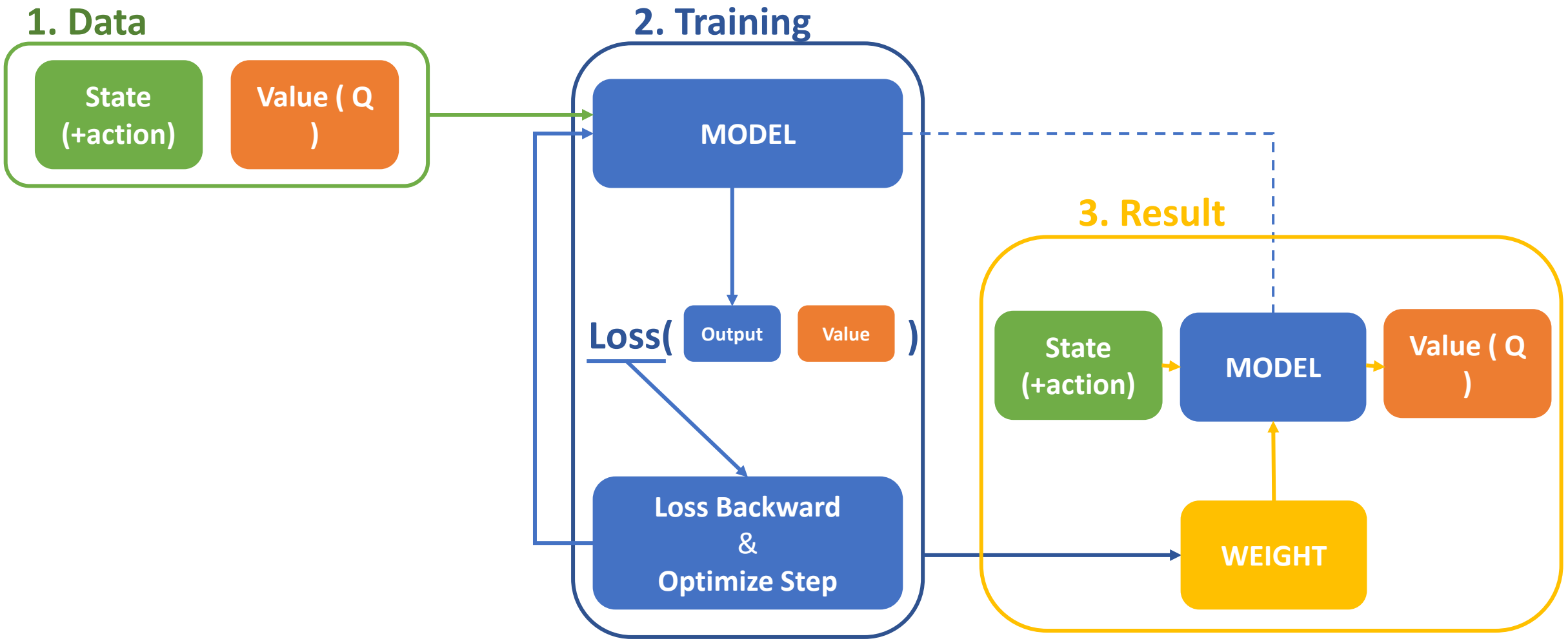
$$\mathbf{w}' = \mathbf{w} - \alpha * \nabla_w \mathrm{L}(\mathrm{w})$$

SGD (Stochastic Gradient Descent)

Adam



정리


1. Data
State (+action)
Value (Q)
2. Training
MODEL
Loss(
Output
Value
)
Loss Backward & Optimize Step
3. Result
State (+action)
MODEL
Value (Q)
WEIGHT

Reference

- 바닥부터 배우는 강화학습 (노승은 저)