

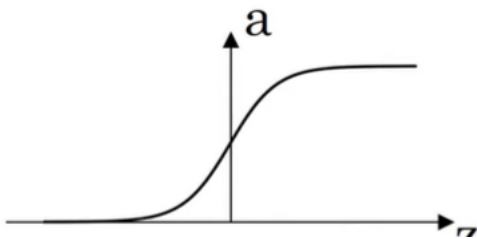
[Week4]_이원주

신경망 학습

활성화 함수

- 비선형이어야 함.
- 종류 (4가지)

sigmoid



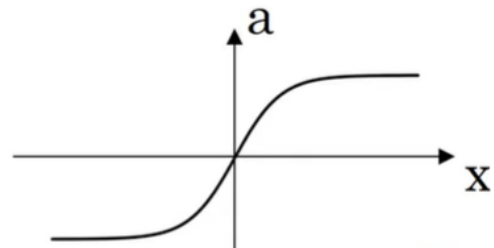
▼ 장점

출력값 : 0~1

→ binary classification의 출력층에 적합.

왜냐하면 그 출력층은 $y=1$ 일 확률을 나타내는데, 확률은 0~1 사이 값이니까.

tanh



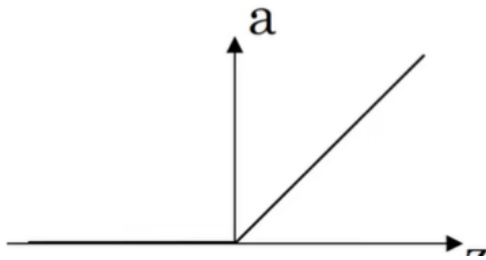
▼ 장점

평균이 0

→ 데이터를 원점으로 이동하는 효과 有.

→ 평균이 0.5인 Sigmoid보다 효율적.

ReLU

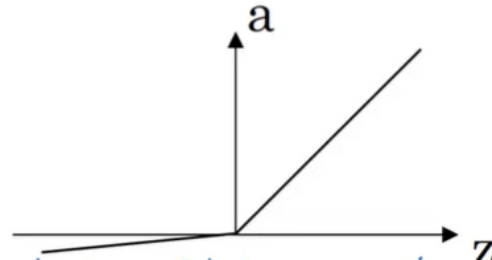


▼ 장점

sigmoid, tanh는 x 가 너무 크거나 작을 때 도함수가 거의 0
→ 학습 속도 저하

반면 ReLU는 x 절반 ($x > 0$) 은 도함수가 1으로, 0에 가깝지 X
→ 학습 빠름.

leaky ReLU



▼ 장점

남은 x 절반($x < 0$)도 도함수가 0이 아님. → 학습 더 빠름.



[요약]

- 활성화 함수 정하기
 - 뭘 쓸지 모르겠다면 → **ReLU**
 - Binary classification의 출력층 → **sigmoid**

파라미터 초기화



[요약]

- 학습 전 파라미터를 초기화한다.
 - `w = np.random.rand(shape) * 0.01`
→ 랜덤 초기화.
 - `b = np.zeros(shape)`
→ 0으로 초기화

▼ 이유

- w
 - w를 모두 0으로 초기화하면 안 됨. 같은 층의 모든 유닛이 언제나 같은 값을 계산하기 때문에. → 랜덤하게 초기화.
 - 0.01을 곱해서 아주 작은 수로 만들어줌. (0.01 말고 다른 수여도 괜찮음.)
← w가 크면
 $z = w^T x + b$ 가 너무 크거나 작고, 그러면 sigmoid, tanh 활성화함수를 거쳤을 때 도함수가 거의 0이라 학습 속도가 너무 느려지기 때문.
- b
 - b는 모두 0으로 초기화해도 이런 문제가 생기지 않음. → 그냥 0으로 초기화.