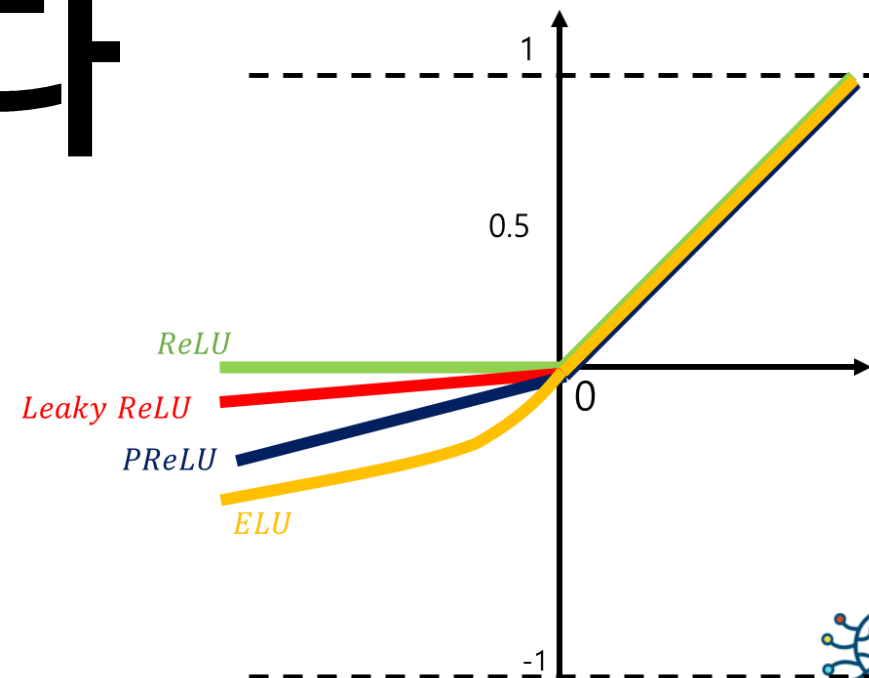
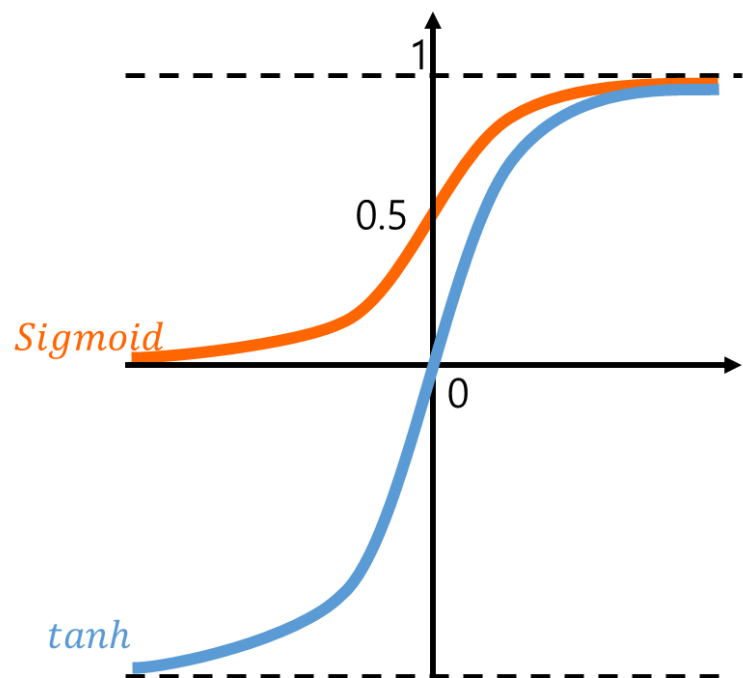


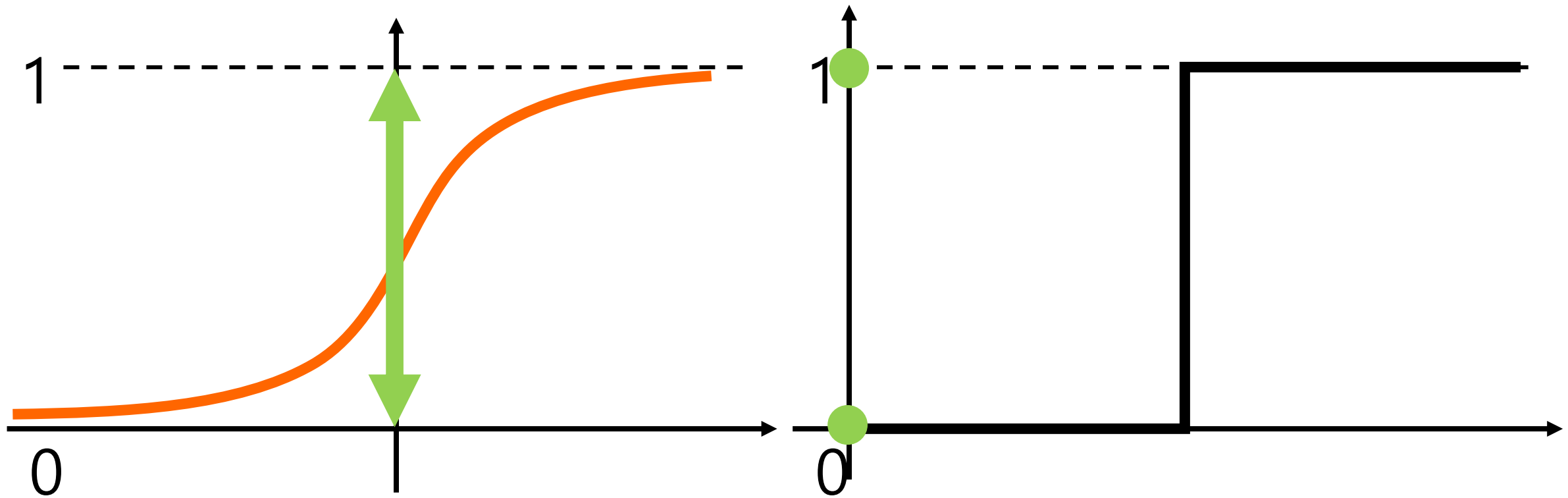
Introduction to 인공 신경망

활성화 함수 가족들을 소개합니다

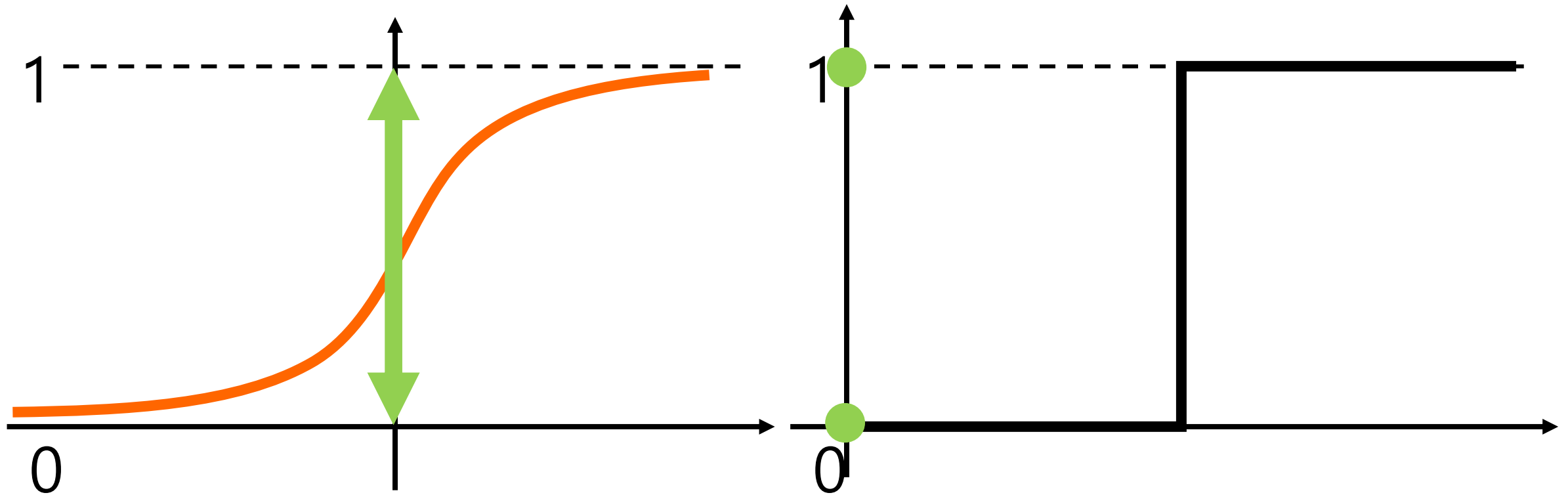


안녕하세요 신박AI입니다

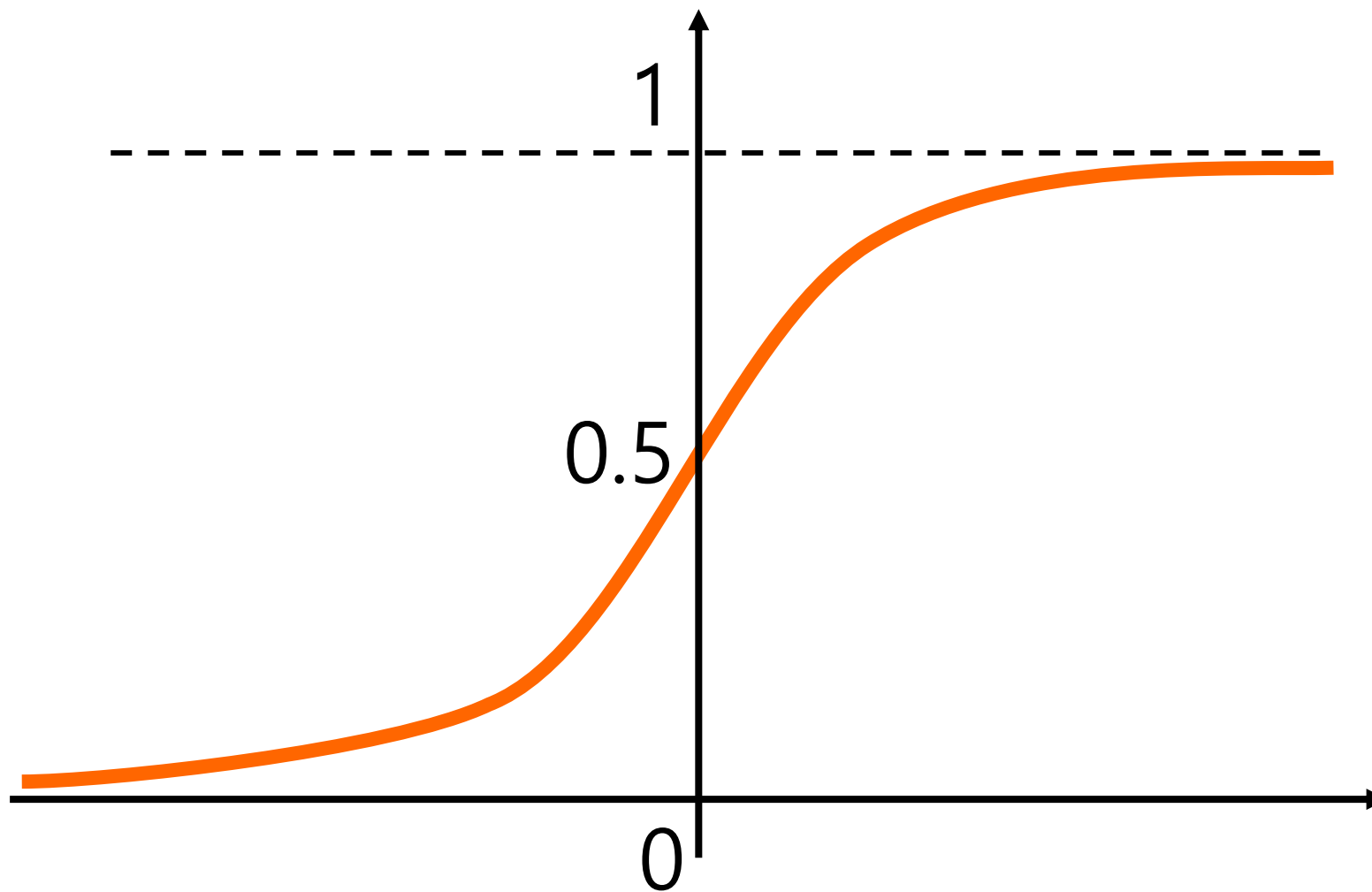
우리는 지난 영상에서 계단 함수의 약점과



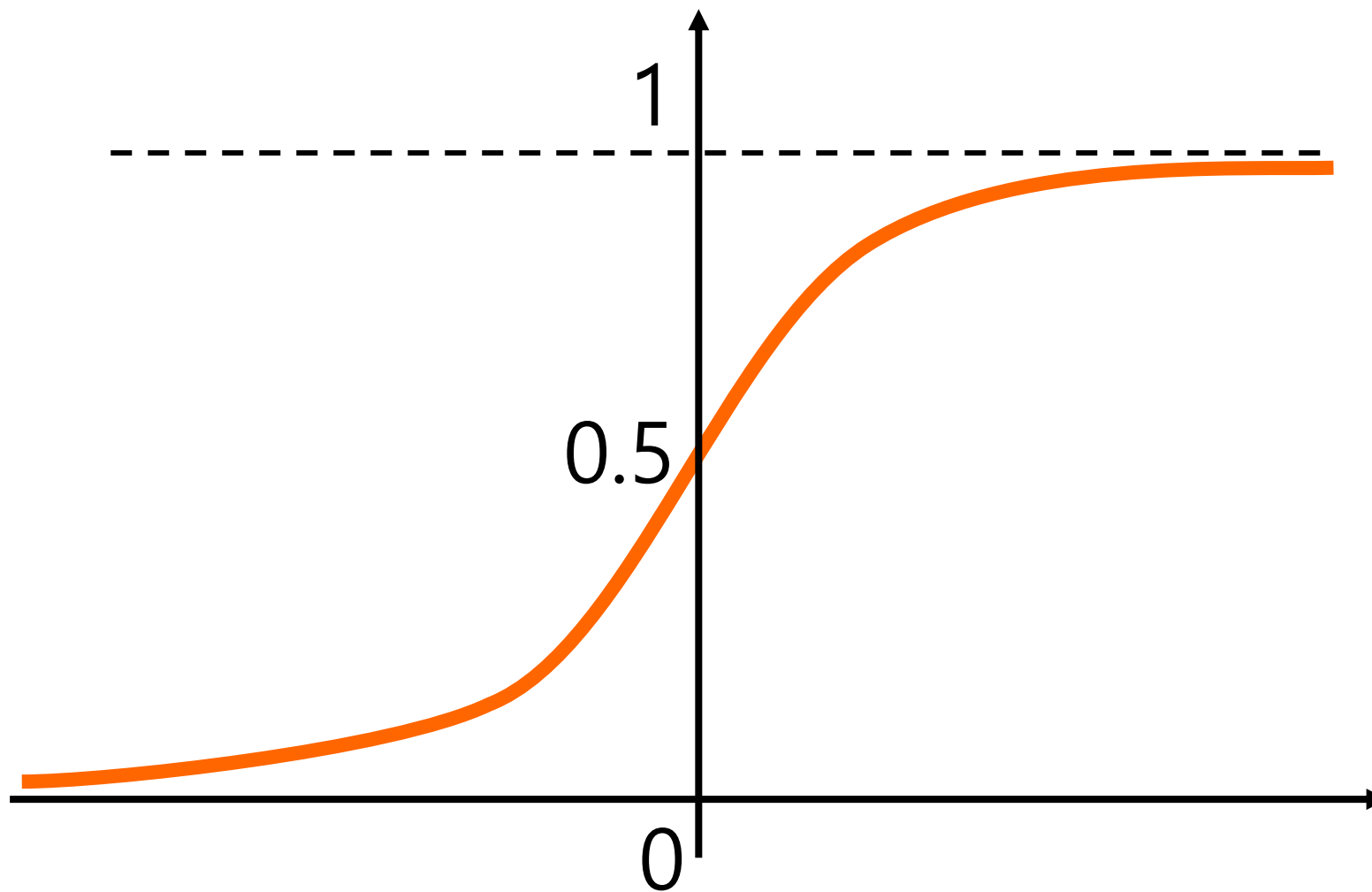
시그모이드 함수의 개념과 특징에 대해 알아보았습니다



이번 영상에서는 시그모이드 함수의 한계와



그 약점을 극복하는 다른 활성화 함수를 소개해 드리겠습니다!



이 채널은 여러분의 관심과 사랑이 필요합니다



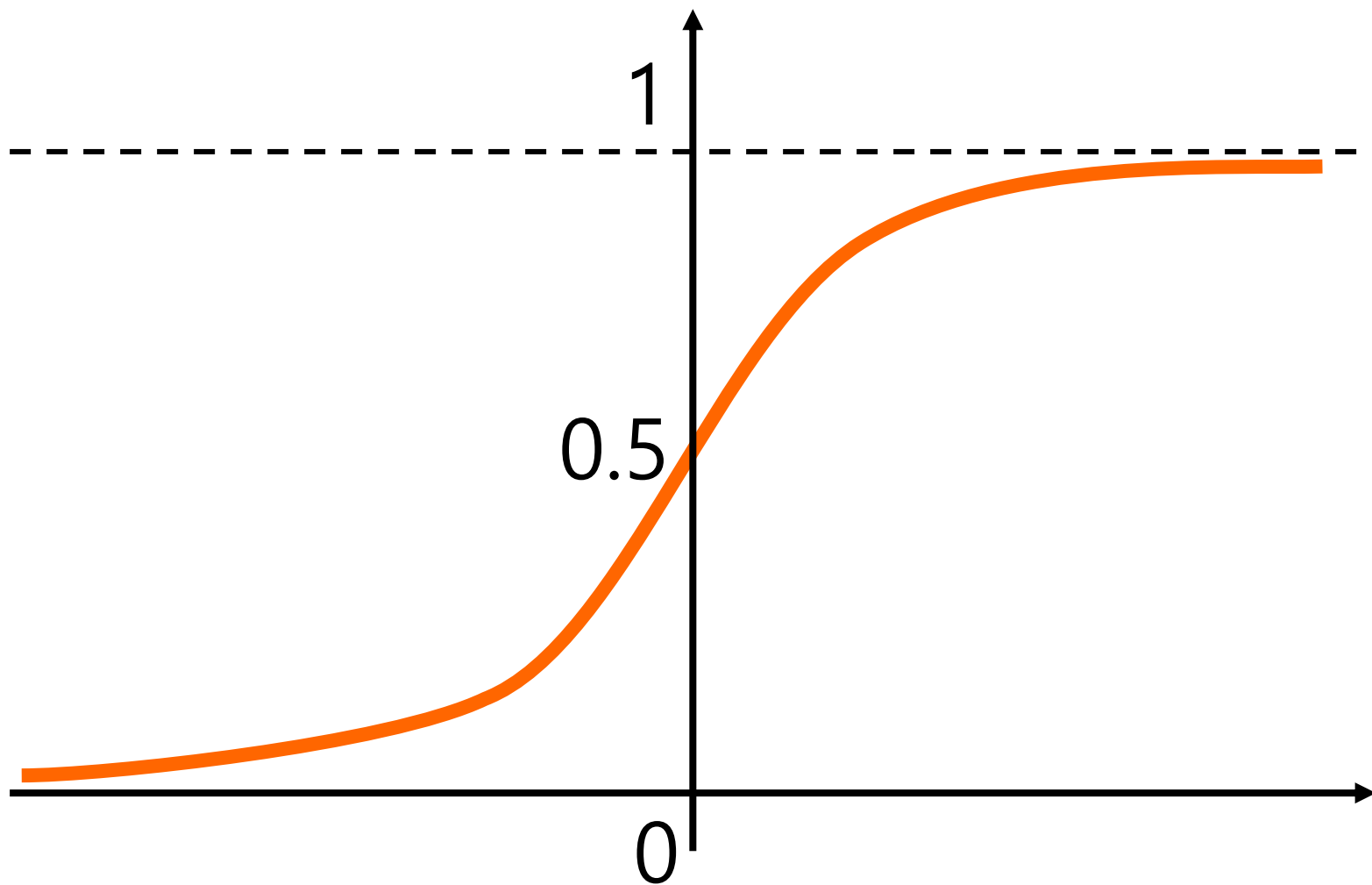
‘좋아요’와 ‘구독’버튼은 강의 준비에 큰 힘이 됩니다!



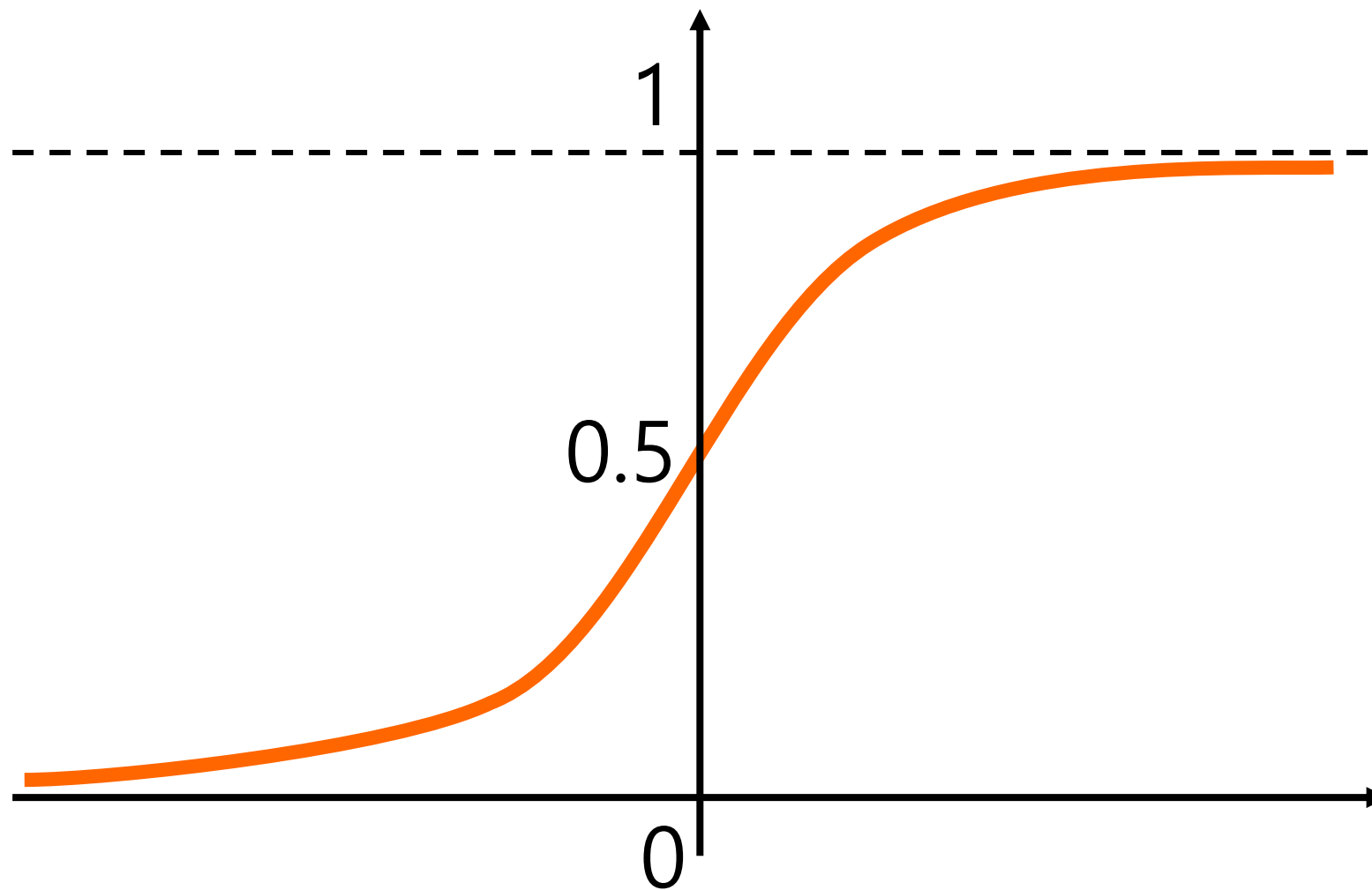
Chapter 1

시그모이드 (sigmoid) 함수의 약점

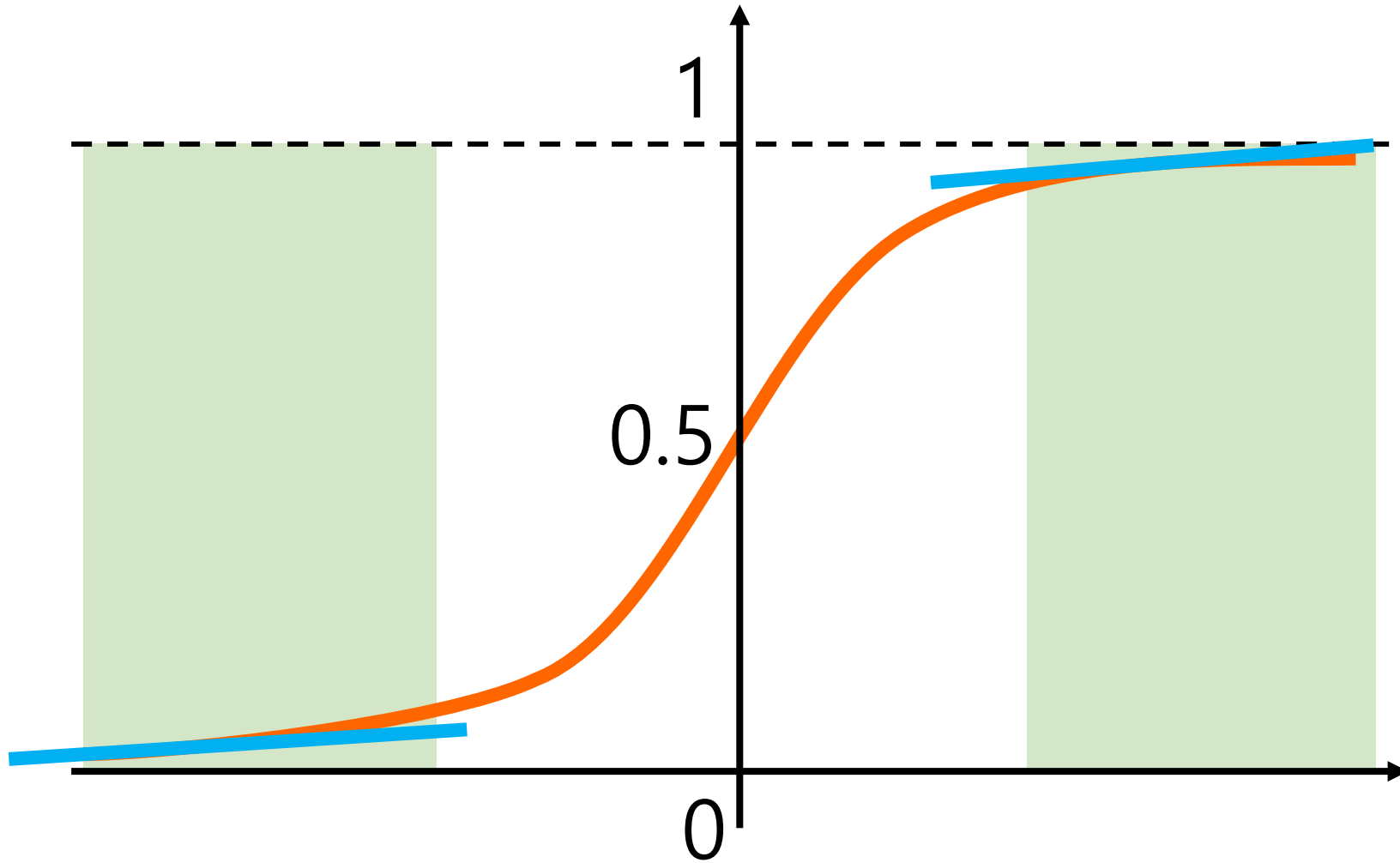
시그모이드 함수는 계단 함수에 비해 분명히 개선된 활성화 함수이지만



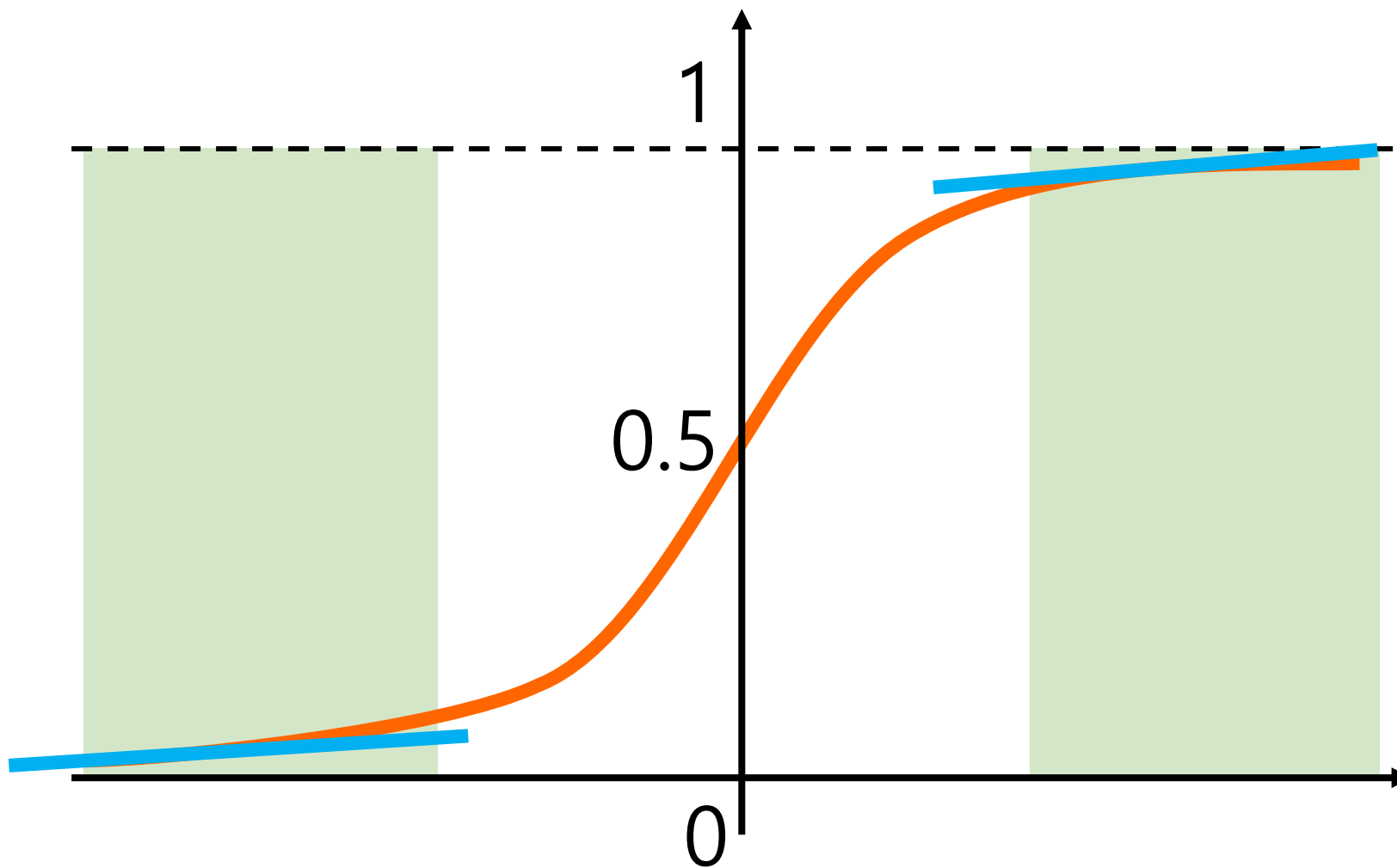
분명한 약점도 있었습니다



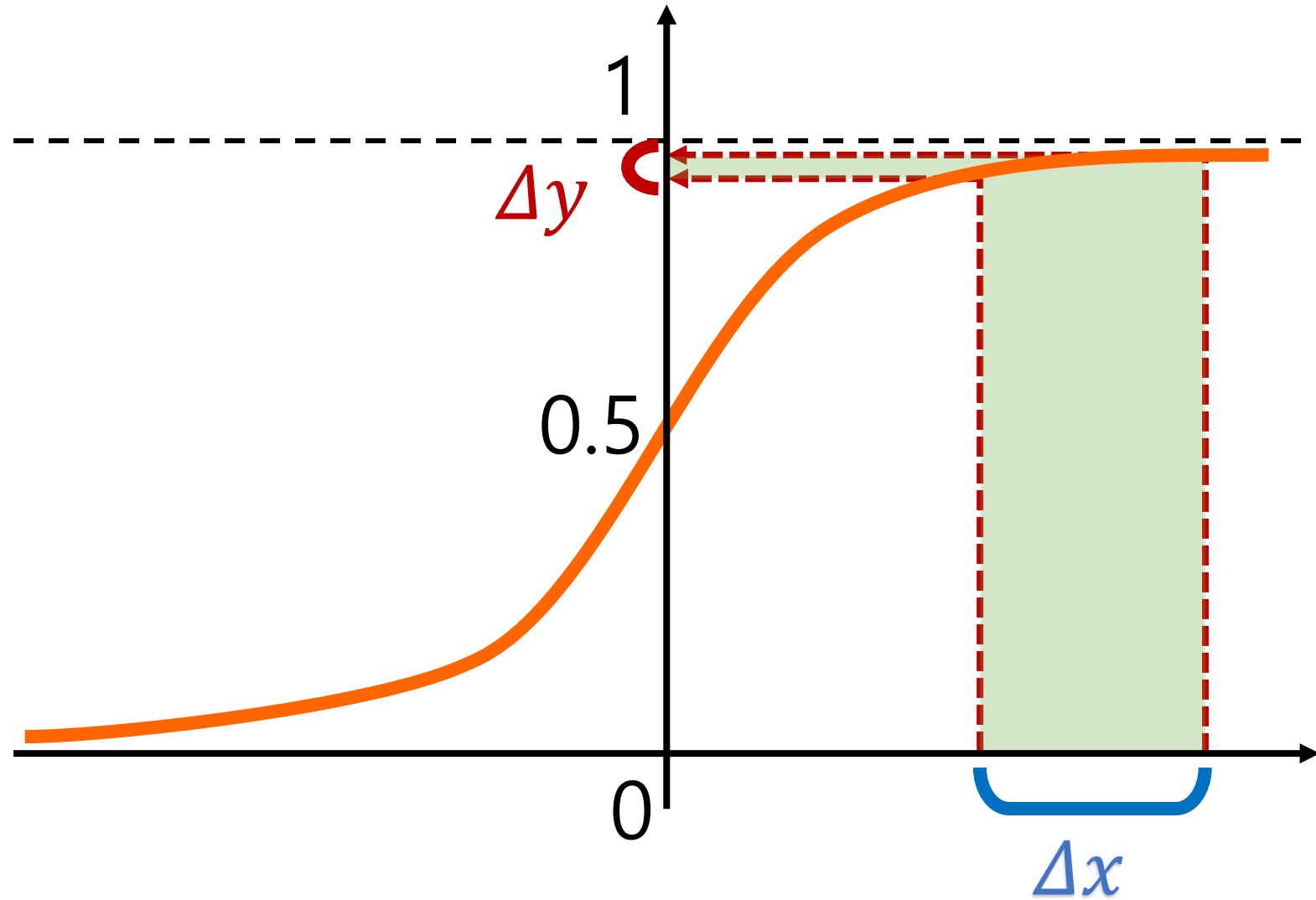
가장 중요한 약점은 양 끝 쪽의 기울기가 0에 가깝다는 것입니다



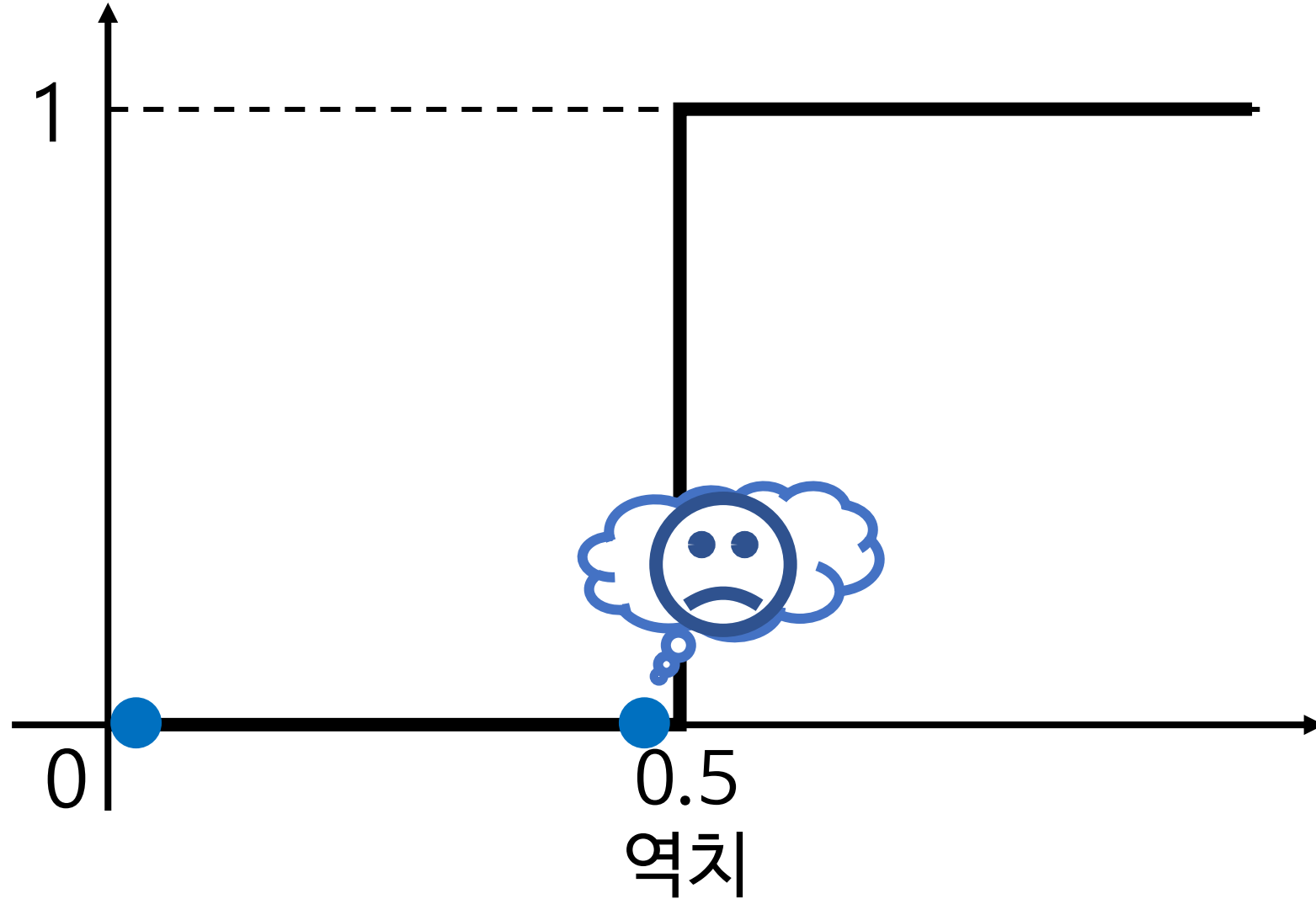
양 끝 쪽의 기울기가 0에 가깝다는 것이 무슨 문제가 되는걸까요?



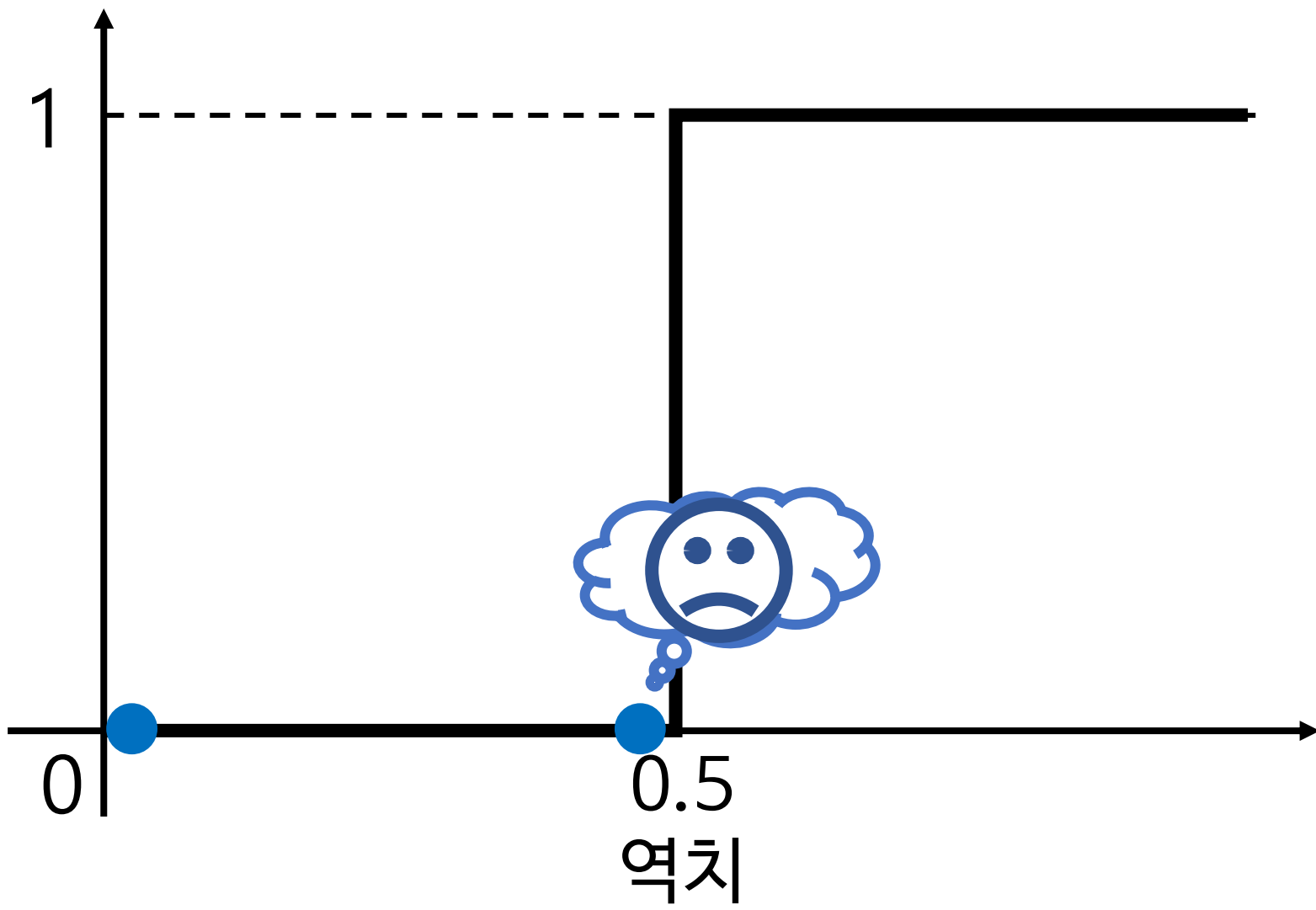
입력값의 차이가 커져도, 출력값의 차이는 미미하기 때문입니다



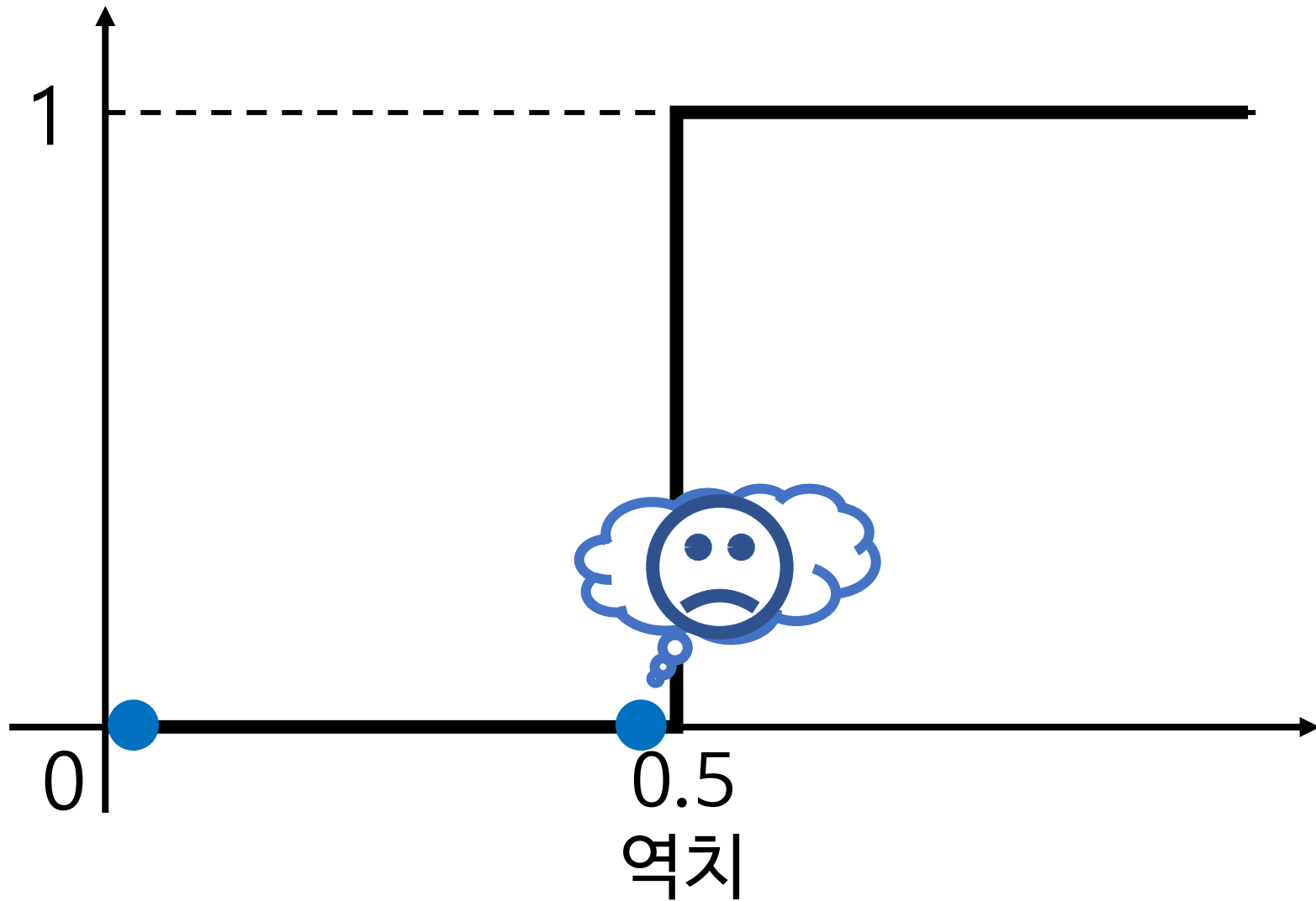
이것은 앞에서 본 계단함수의 문제점과 거의 유사한 문제입니다



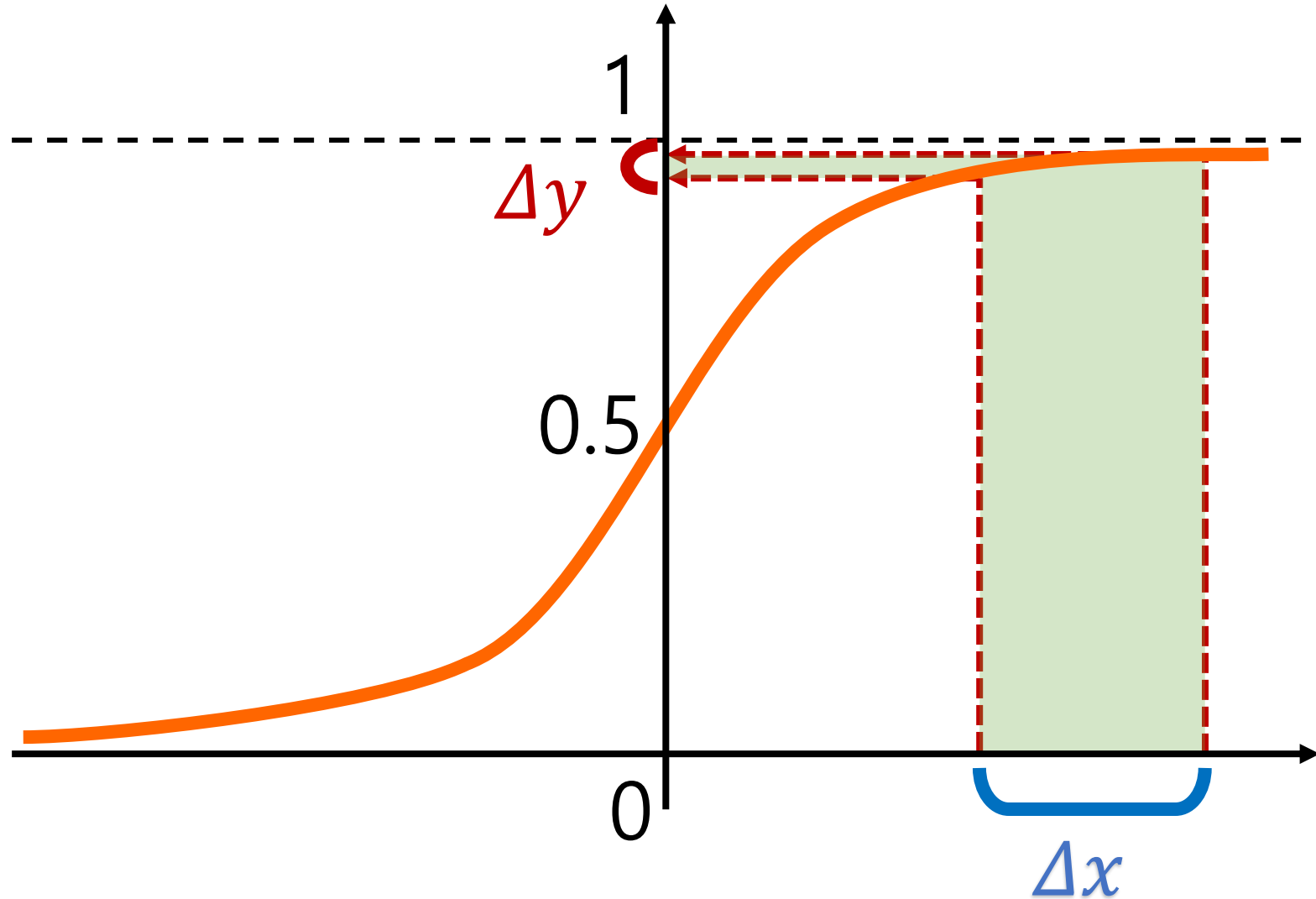
계단함수에서는 입력값이 0.499든 0.0001이든 출력값이 0이었습니다



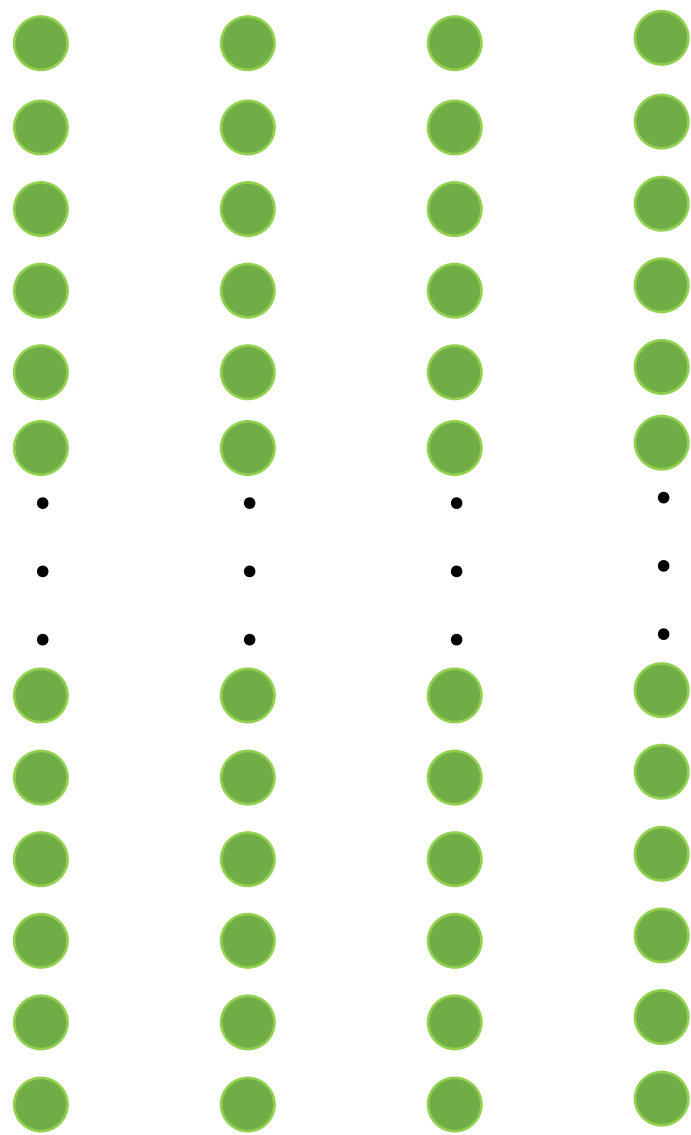
즉, 계단함수의 문제는 입력값의 차이가 출력값에 효과적으로 반영되지 않는다는 점이었습니다



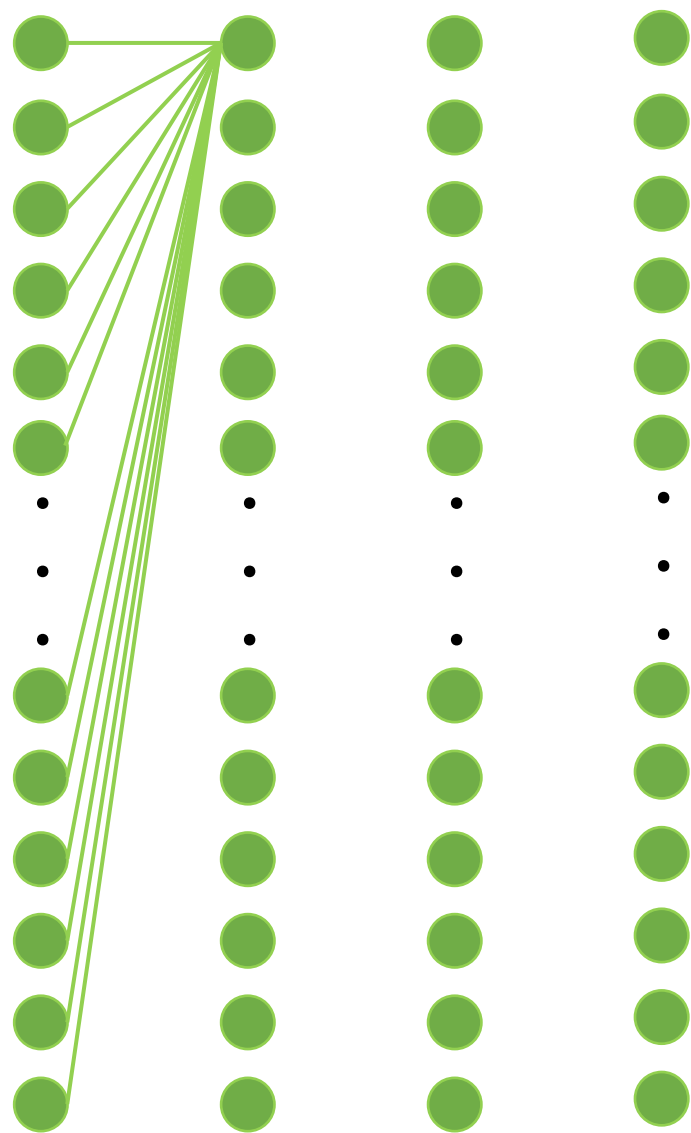
지금 안타깝게도 시그모이드 함수에서도 함수의 양 극단에서 그 문제가 다시 재현되고 있습니다



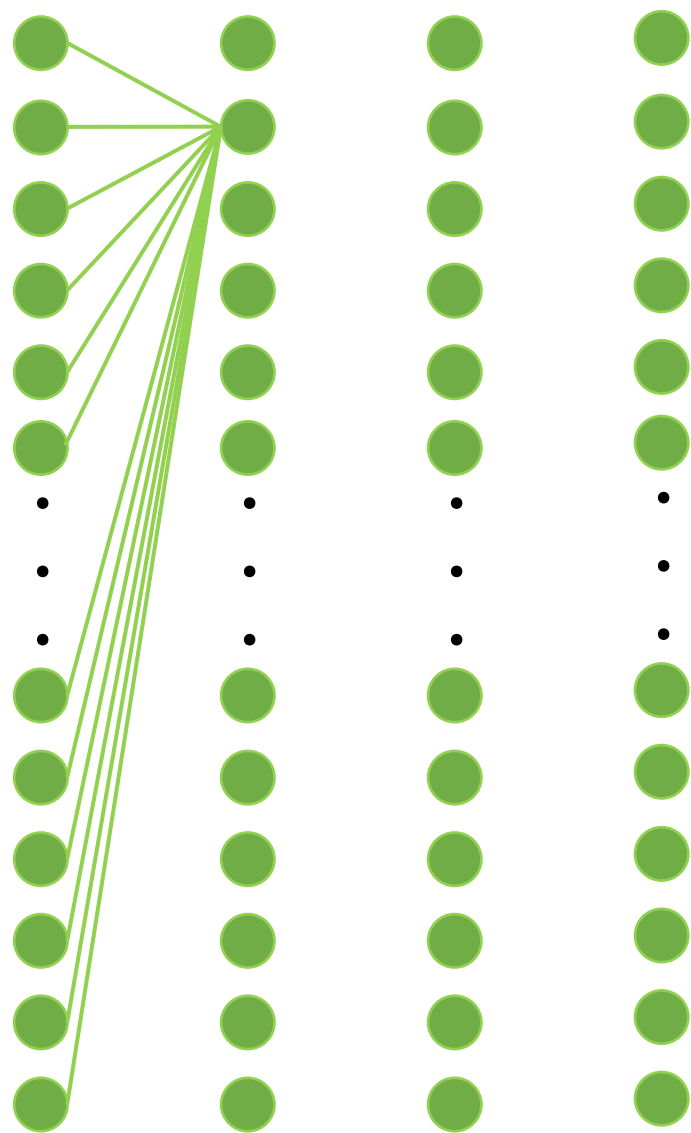
다음과 같은 신경망 deep neural network을 가정해봅시다



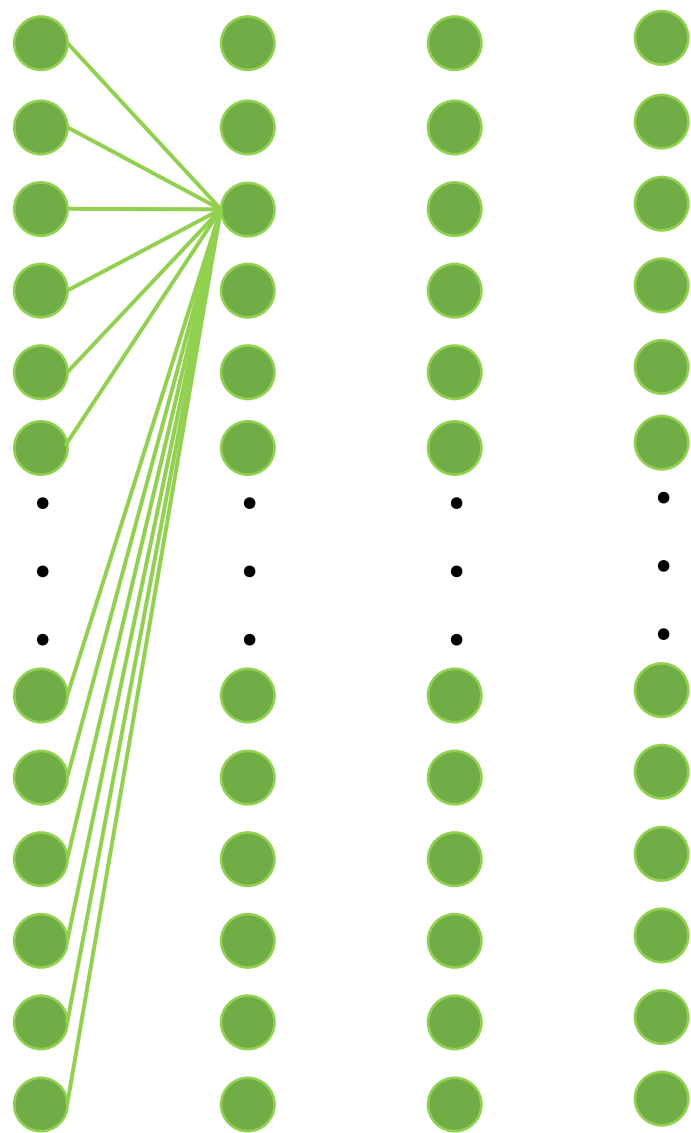
상위노드의 입력값은 많은 하위 노드들의 출력값들의 합이 됩니다



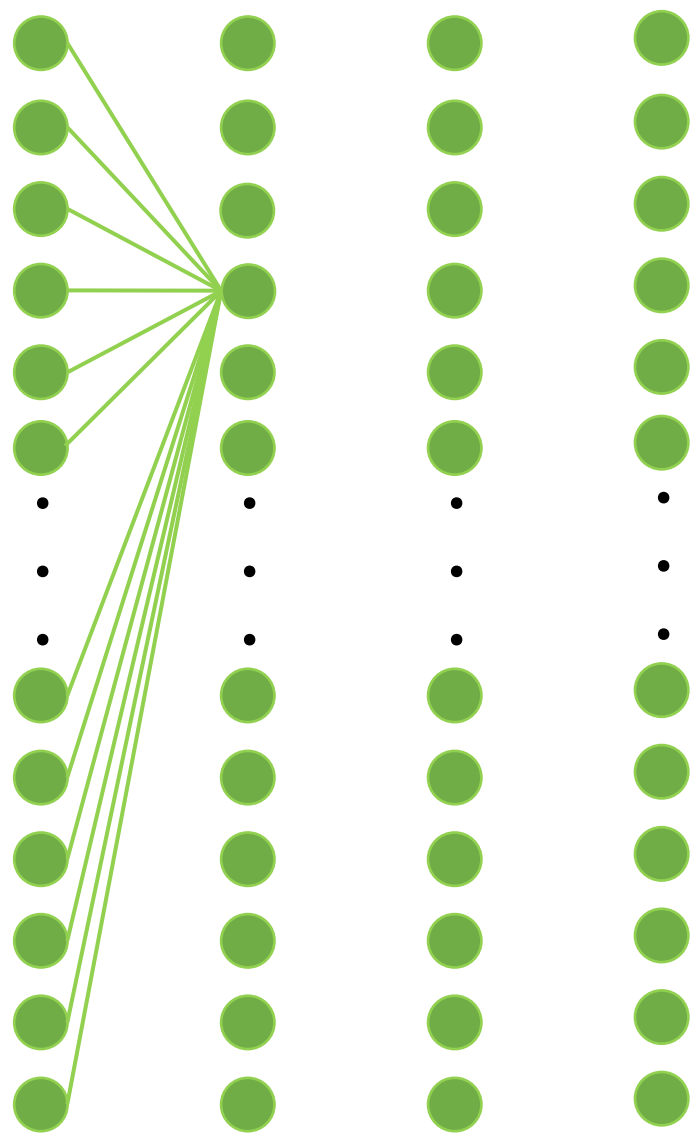
상위노드의 입력값은 많은 하위 노드들의 출력값들의 합이 됩니다



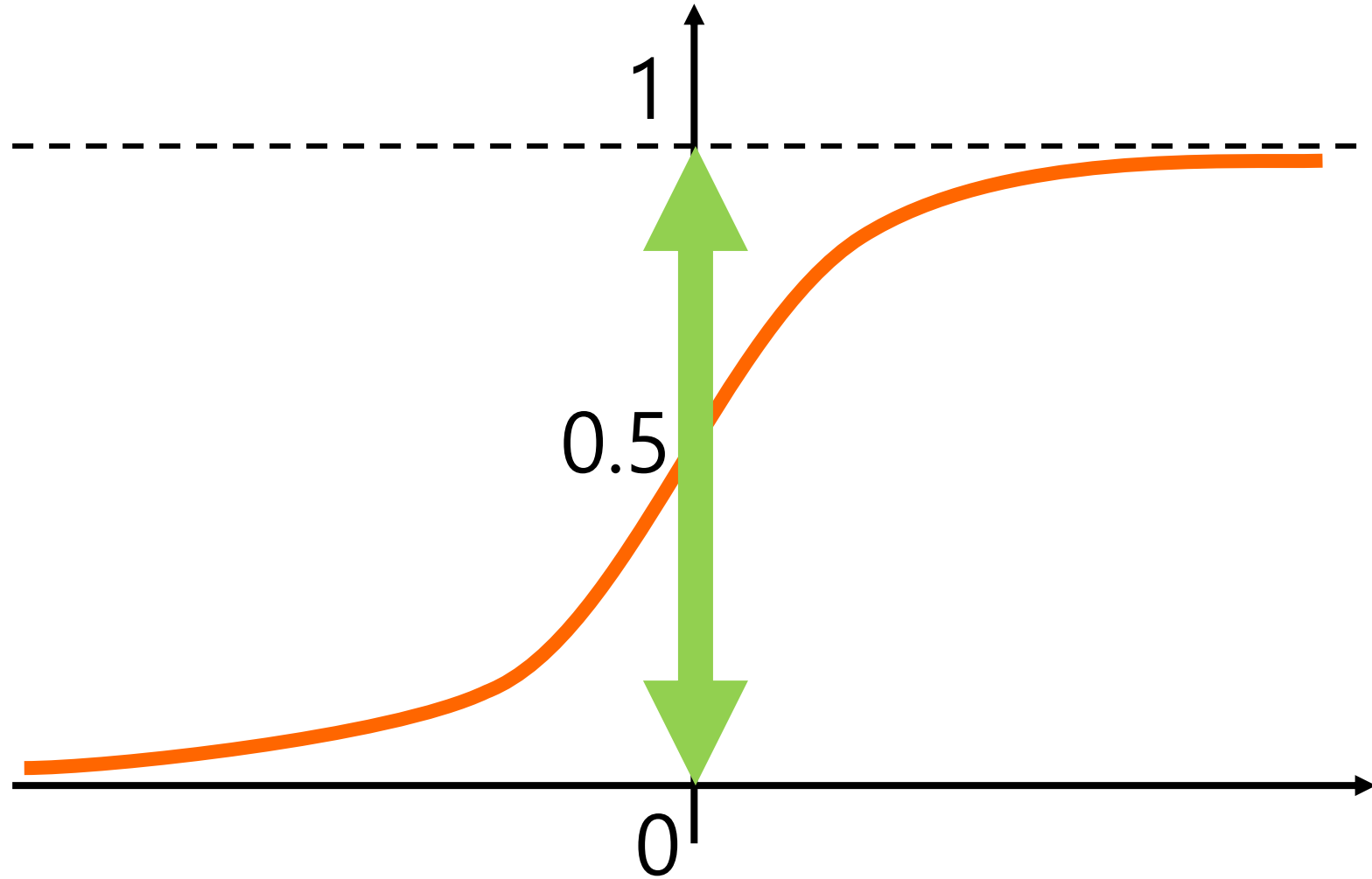
상위노드의 입력값은 많은 하위 노드들의 출력값들의 합이 됩니다



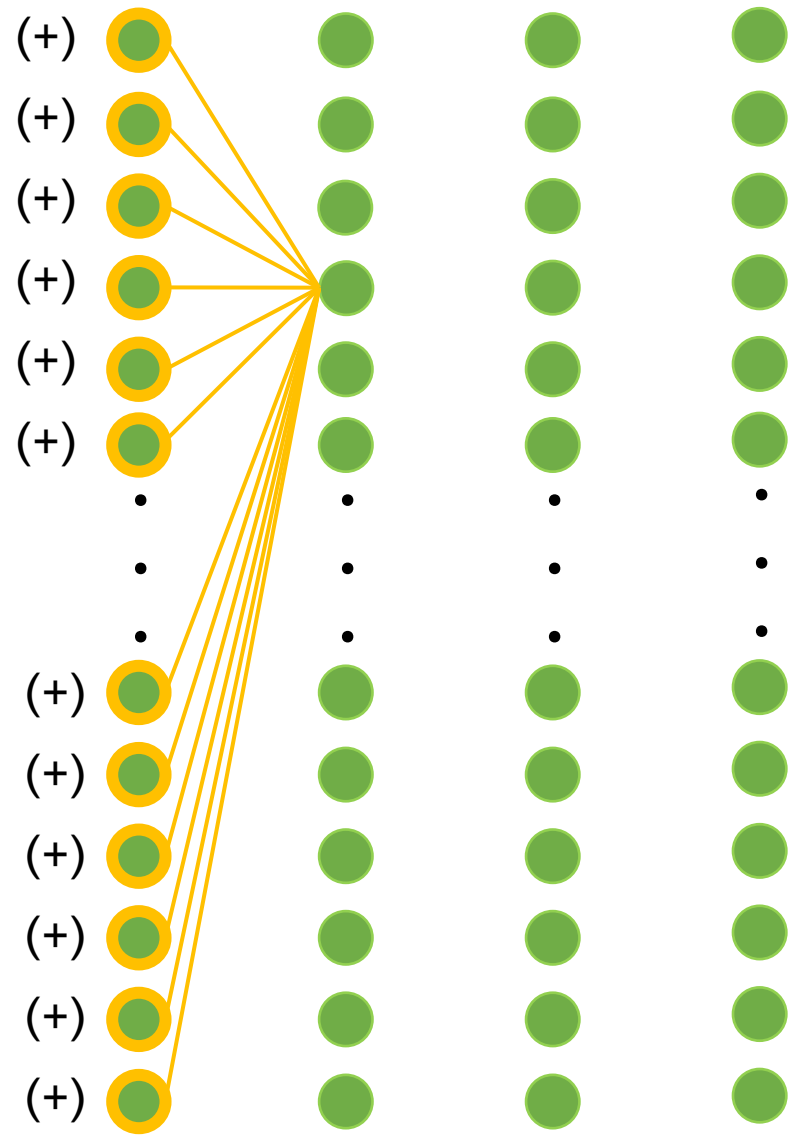
상위노드의 입력값은 많은 하위 노드들의 출력값들의 합이 됩니다



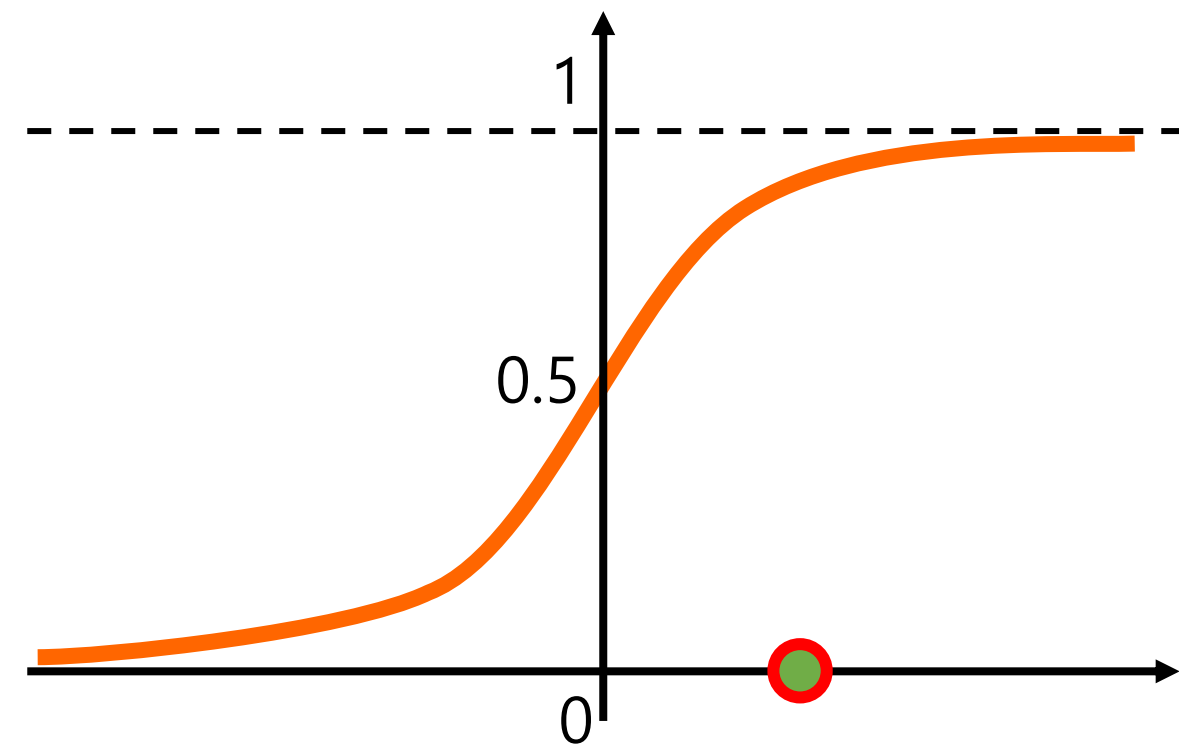
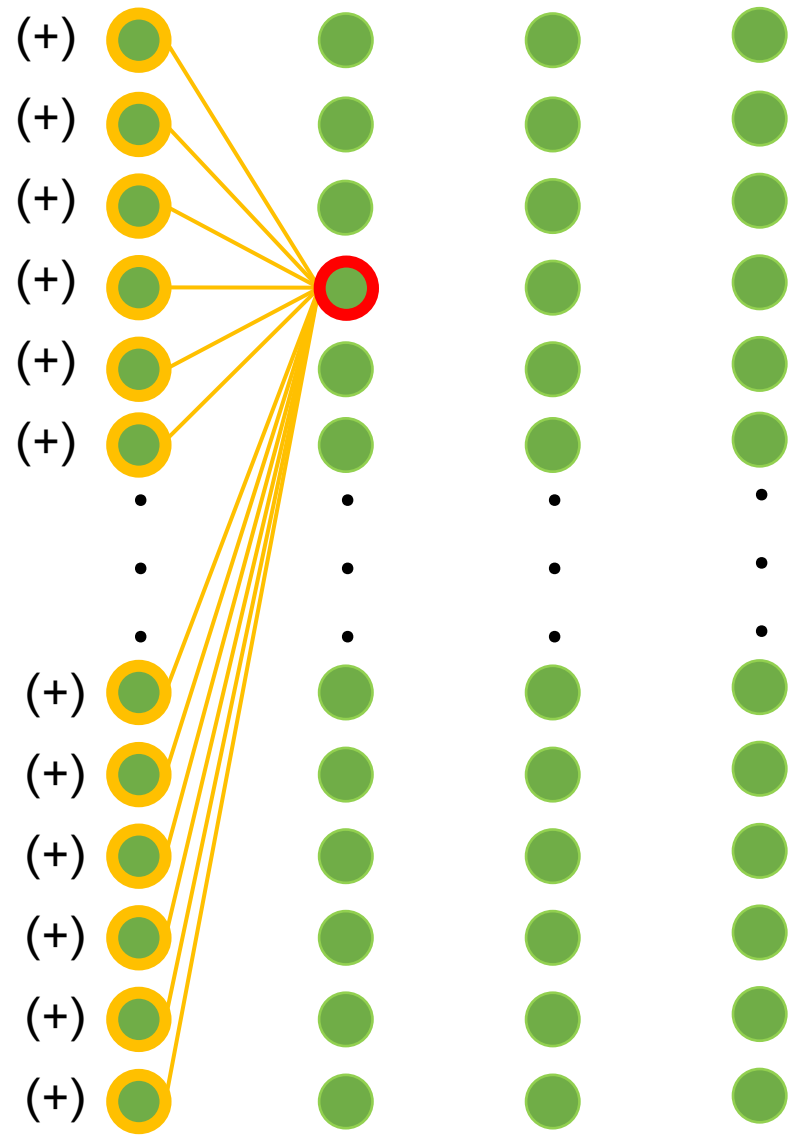
그런데 시그모이드 출력값은 항상 양수이기 때문에..



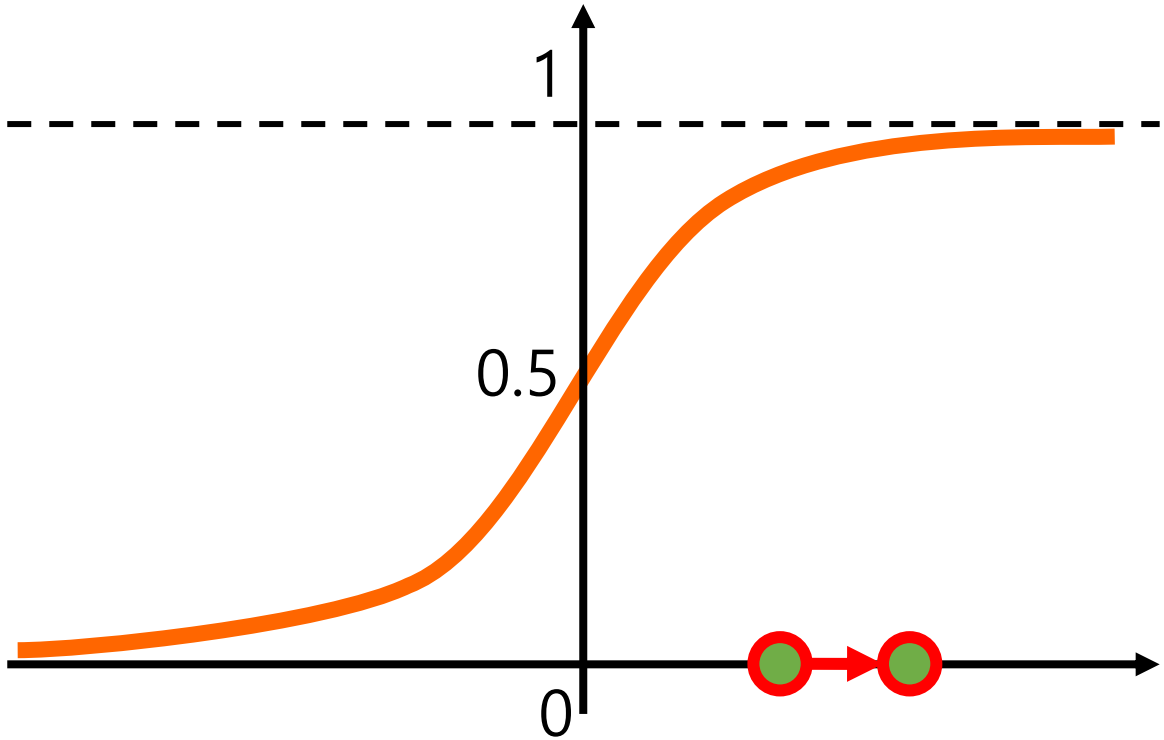
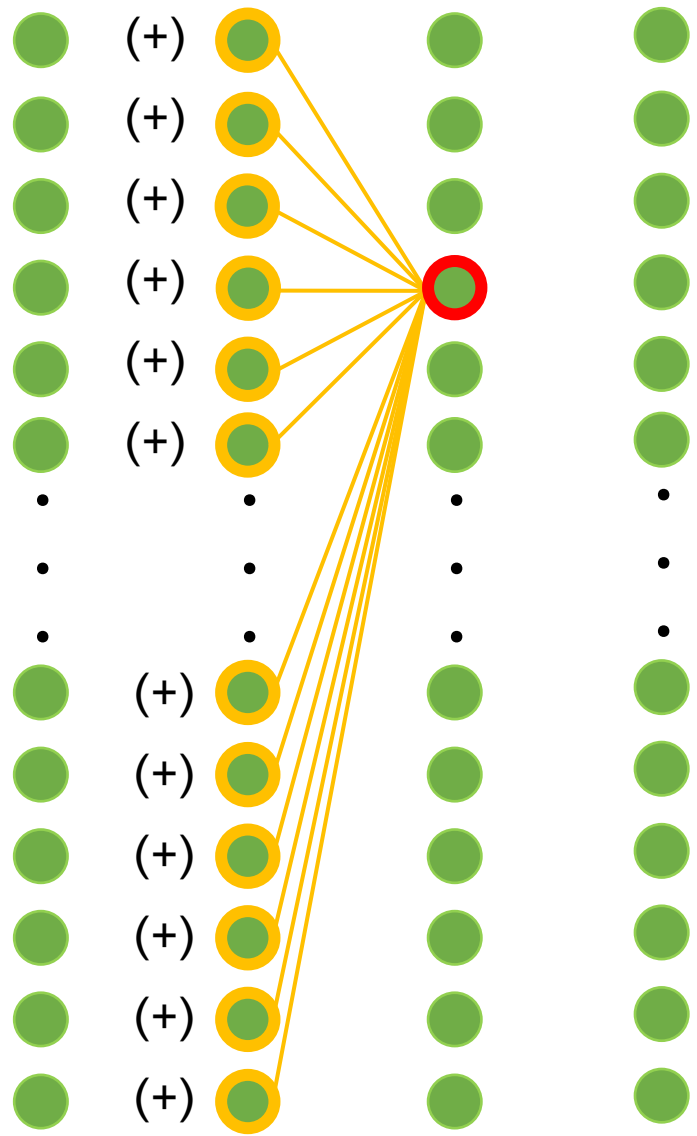
만약 연결강도 (가중치)의 값 또한 0과 1 사이의 양수라고 가정한다면,



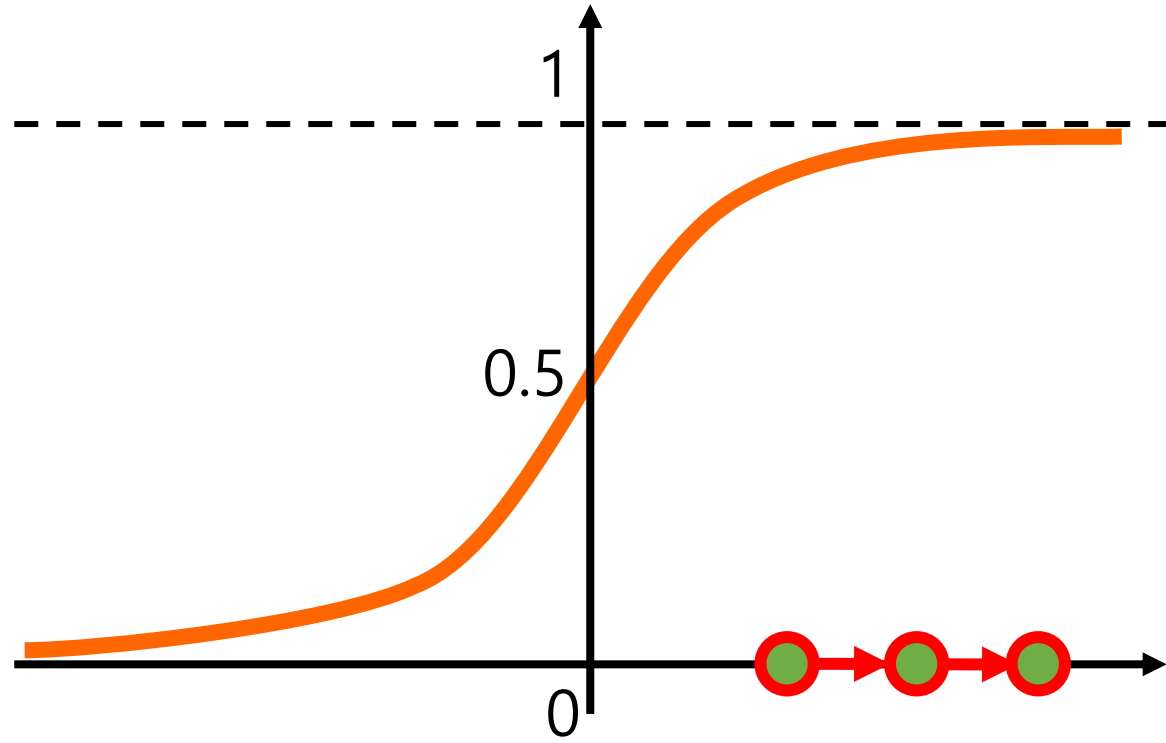
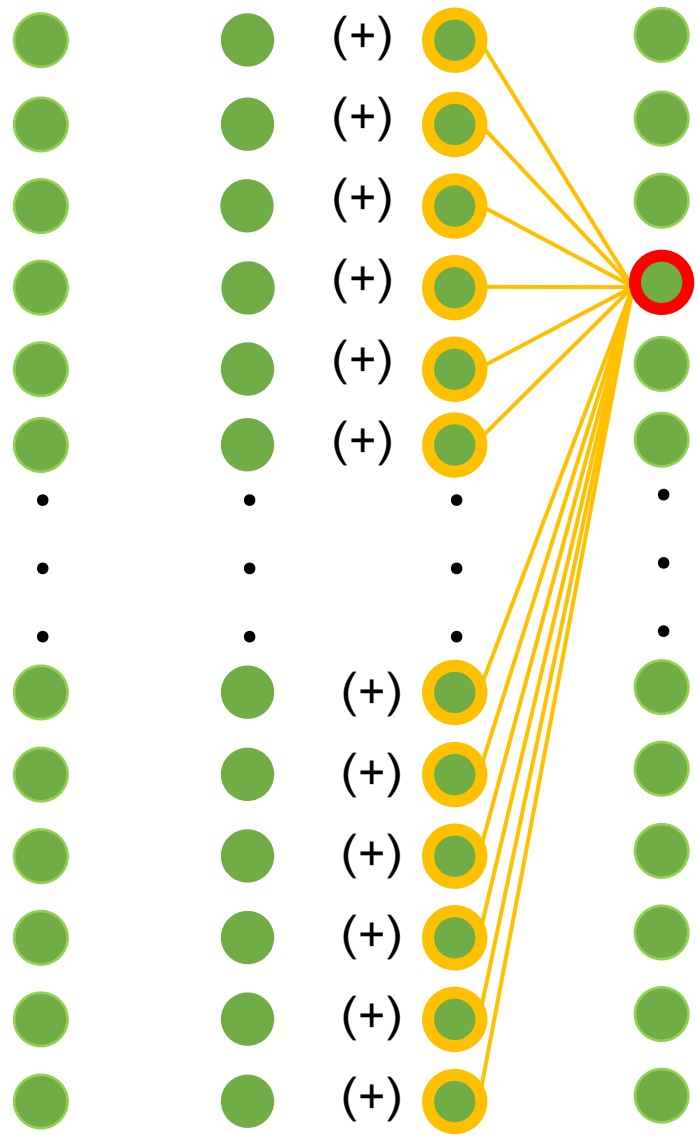
상위 노드의 입력값은 당연히 양수가 되고 (양수의 합은 양수),



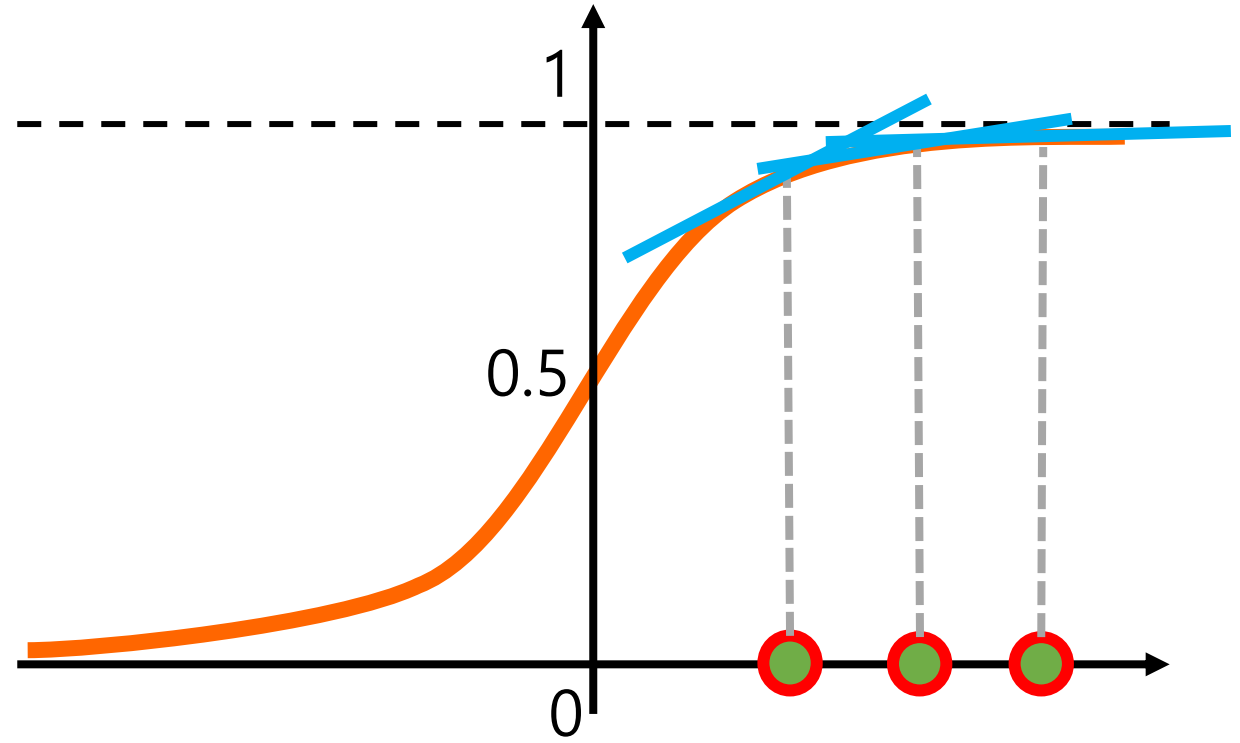
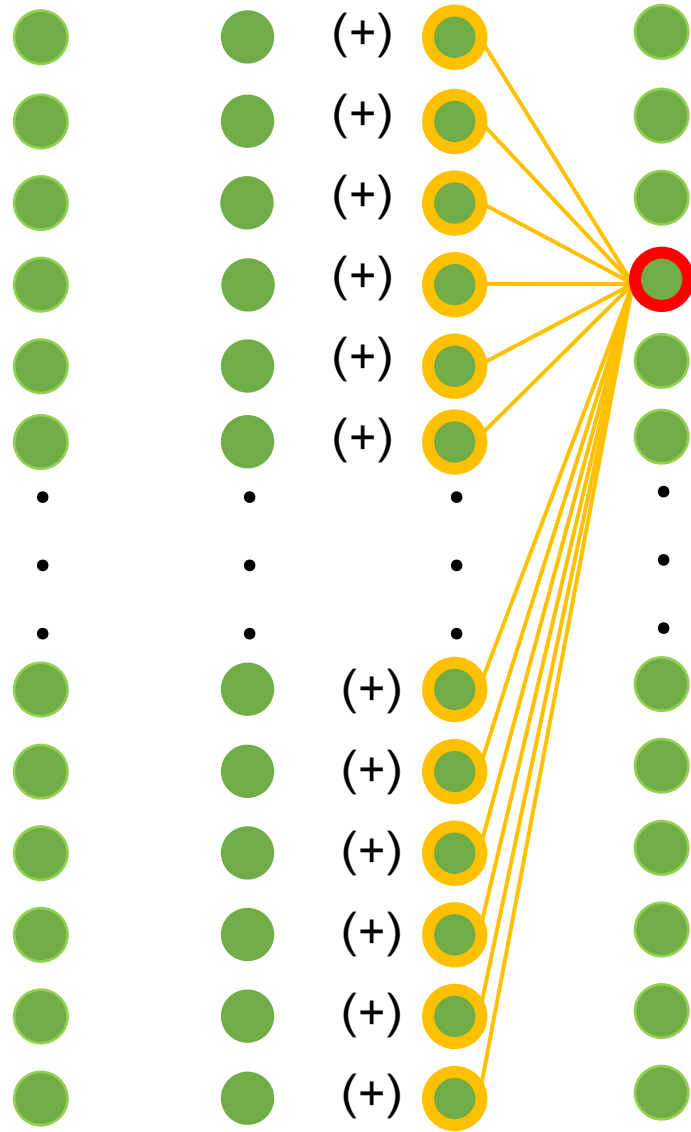
상위 계층으로 올라갈 수록 입력값은 점점 커져가는 경향성이 생깁니다



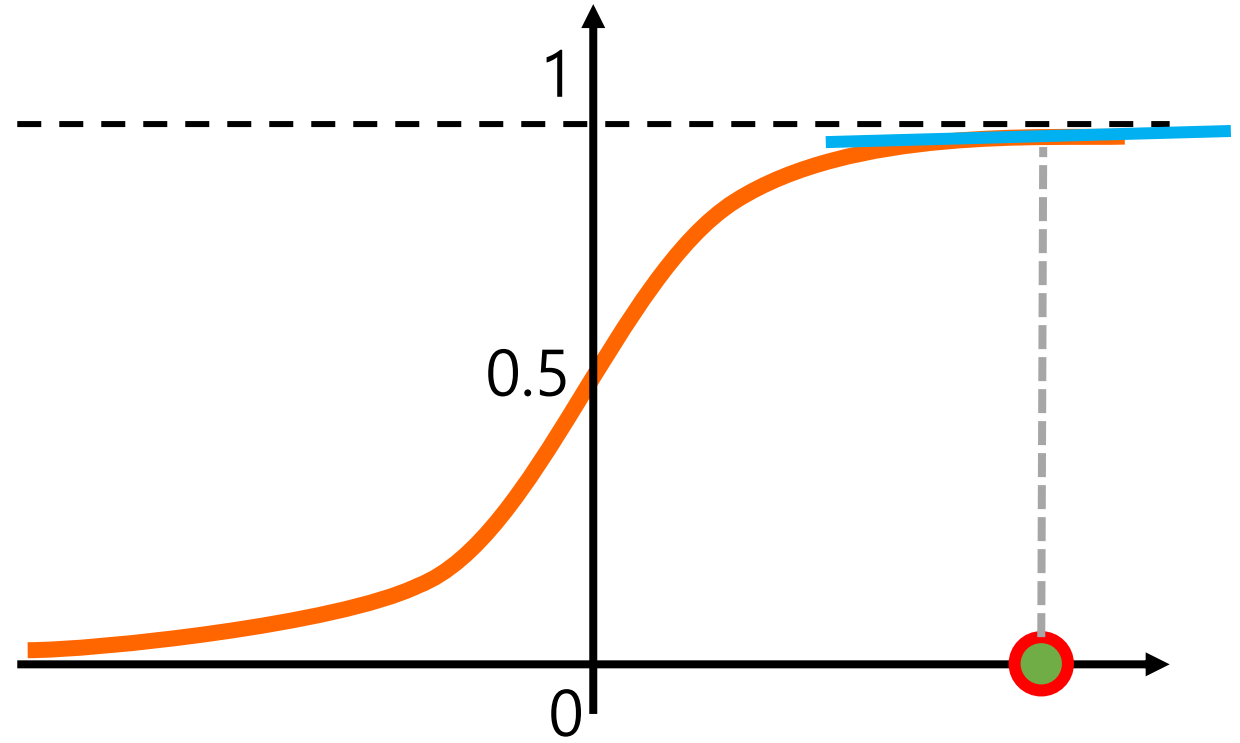
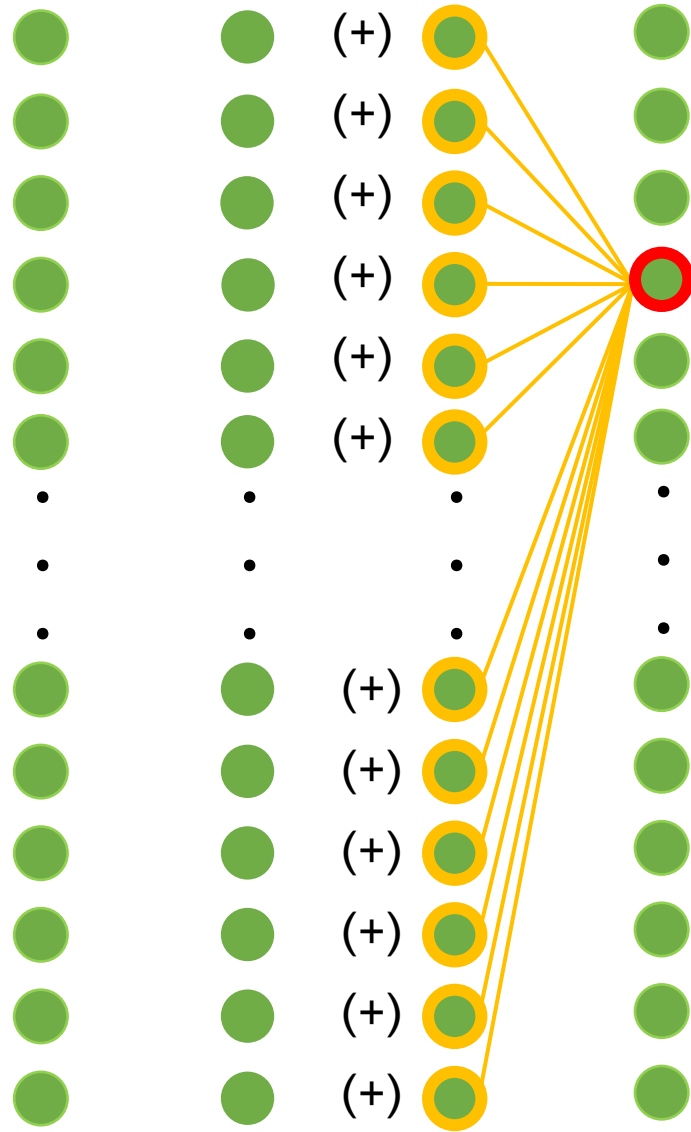
상위 계층으로 올라갈 수록 입력값은 점점 커져가는 경향성이 생깁니다



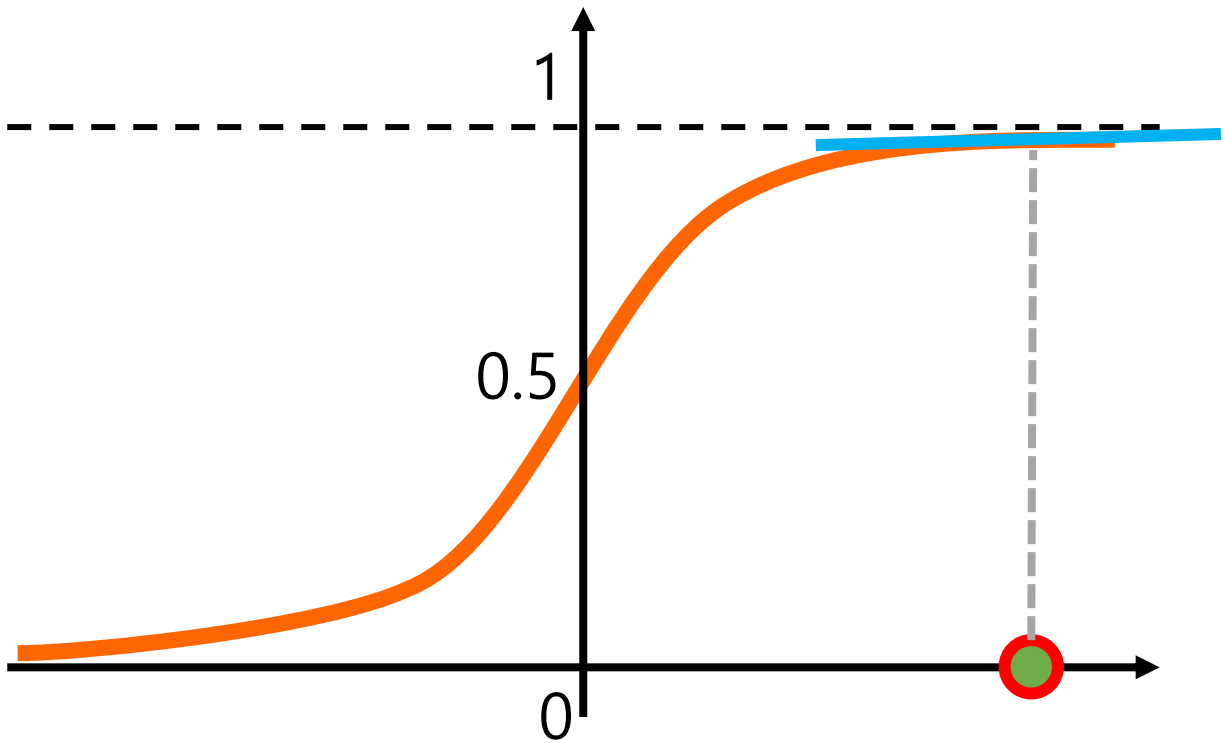
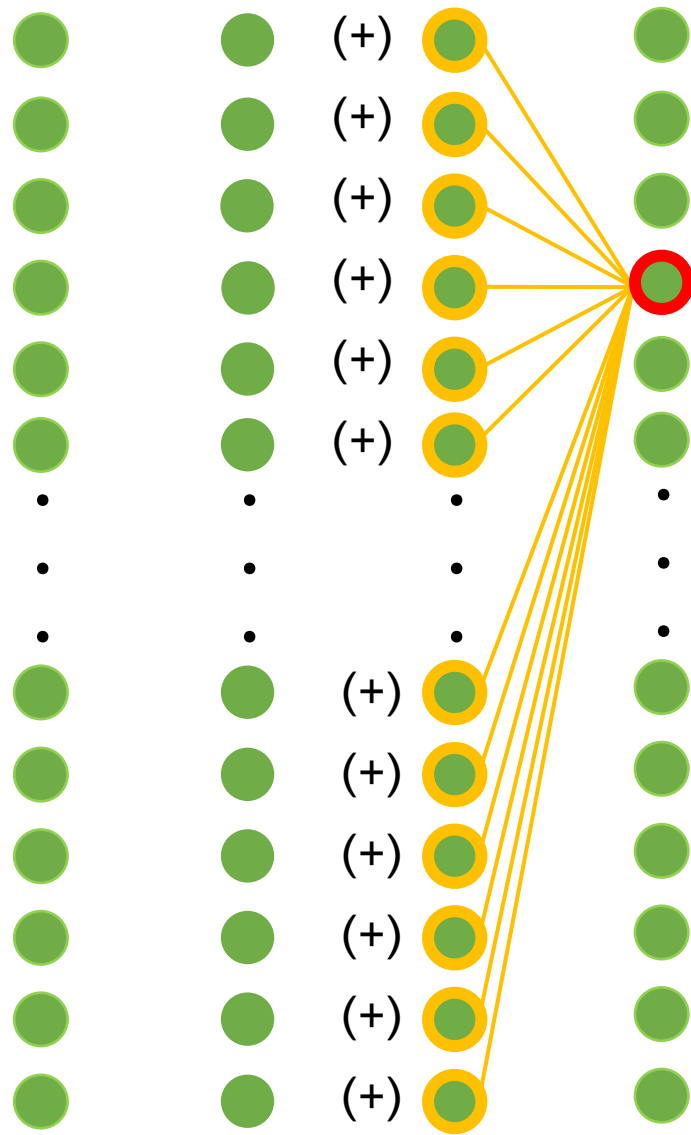
입력값이 커지면 그에 해당하는 기울기는 0에 가까워집니다



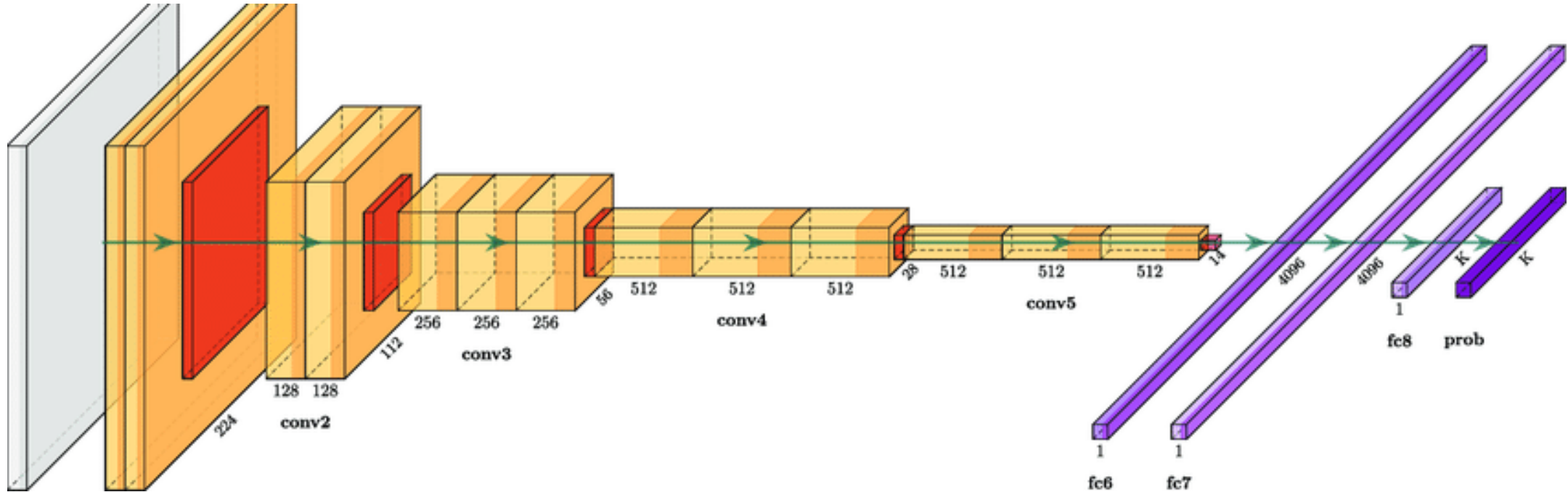
나중 영상에서 더 자세히 나누겠지만, 기울기가 0이 되면



오류역전파backpropagation의 학습효과가 0에 가까워집니다

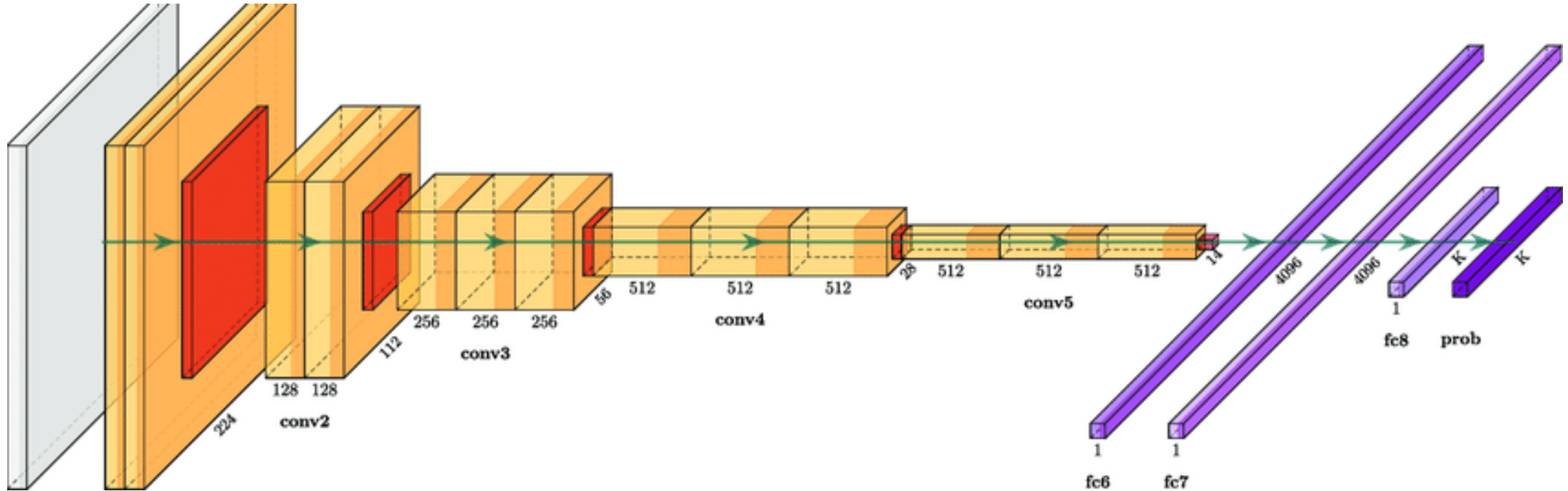


이렇게 신경망의 층이 깊어지면 깊어질 수록 학습효과가 떨어지는 것을



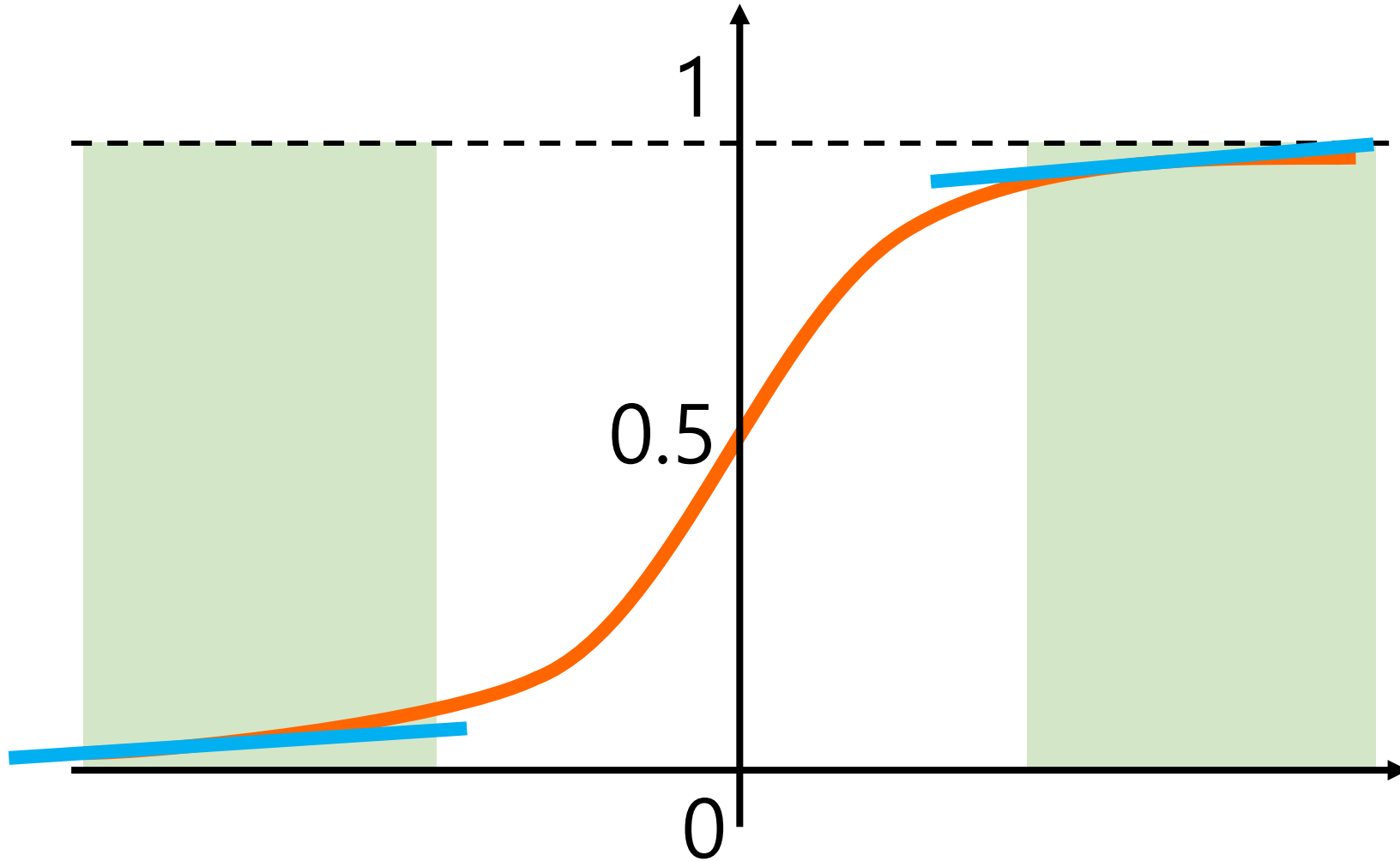
Blauch, Nicholas & Behrmann, Marlene & Plaut, David. (2020). Computational insights into human perceptual expertise for familiar and unfamiliar face recognition. Cognition. 208. 104341. 10.1016/j.cognition.2020.104341.

Sigmoid saturate and killing gradients 현상이라고 하는데요

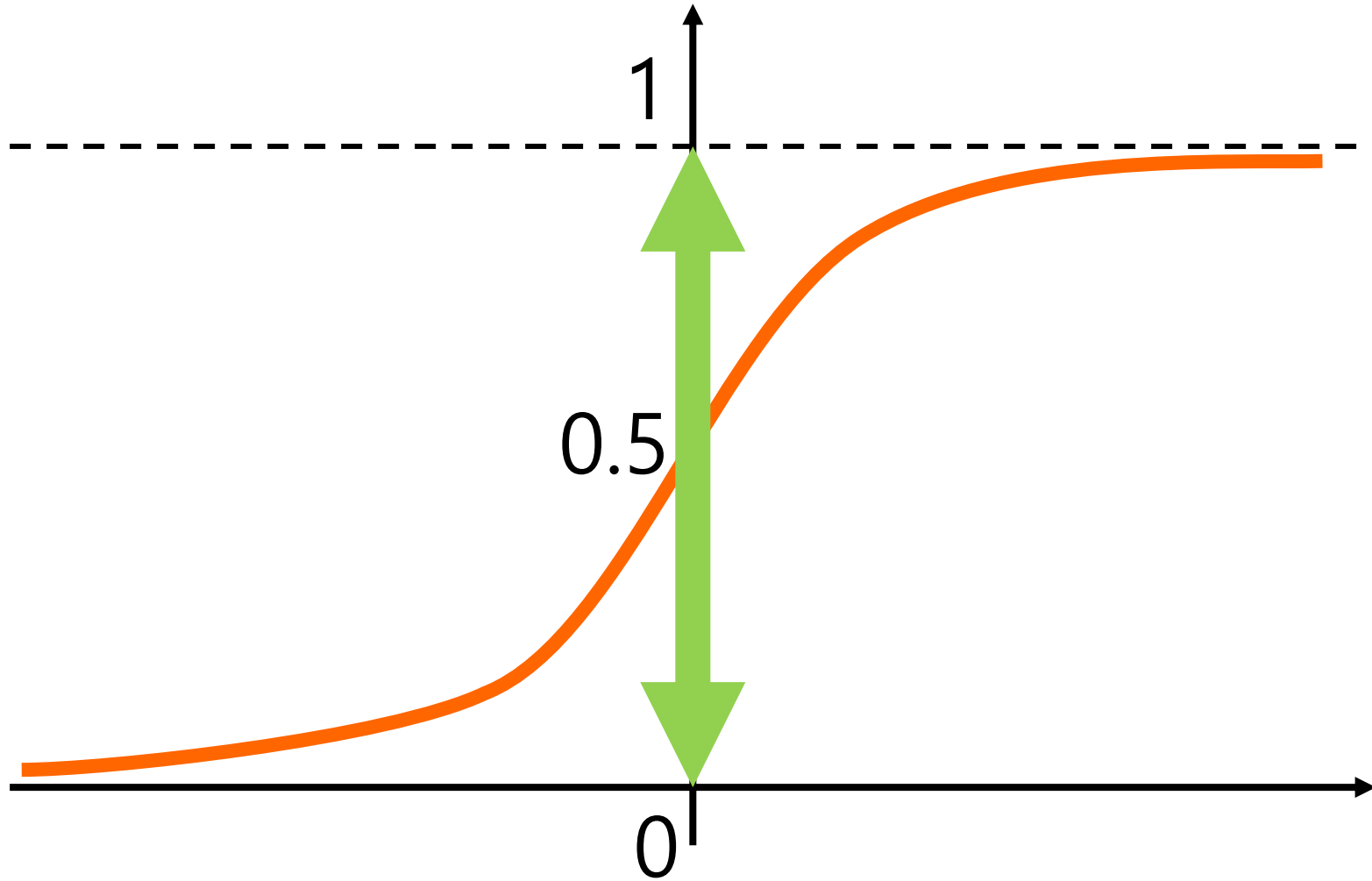


Blauch, Nicholas & Behrmann, Marlene & Plaut, David. (2020). Computational insights into human perceptual expertise for familiar and unfamiliar face recognition. *Cognition*. 208. 104341. 10.1016/j.cognition.2020.104341.

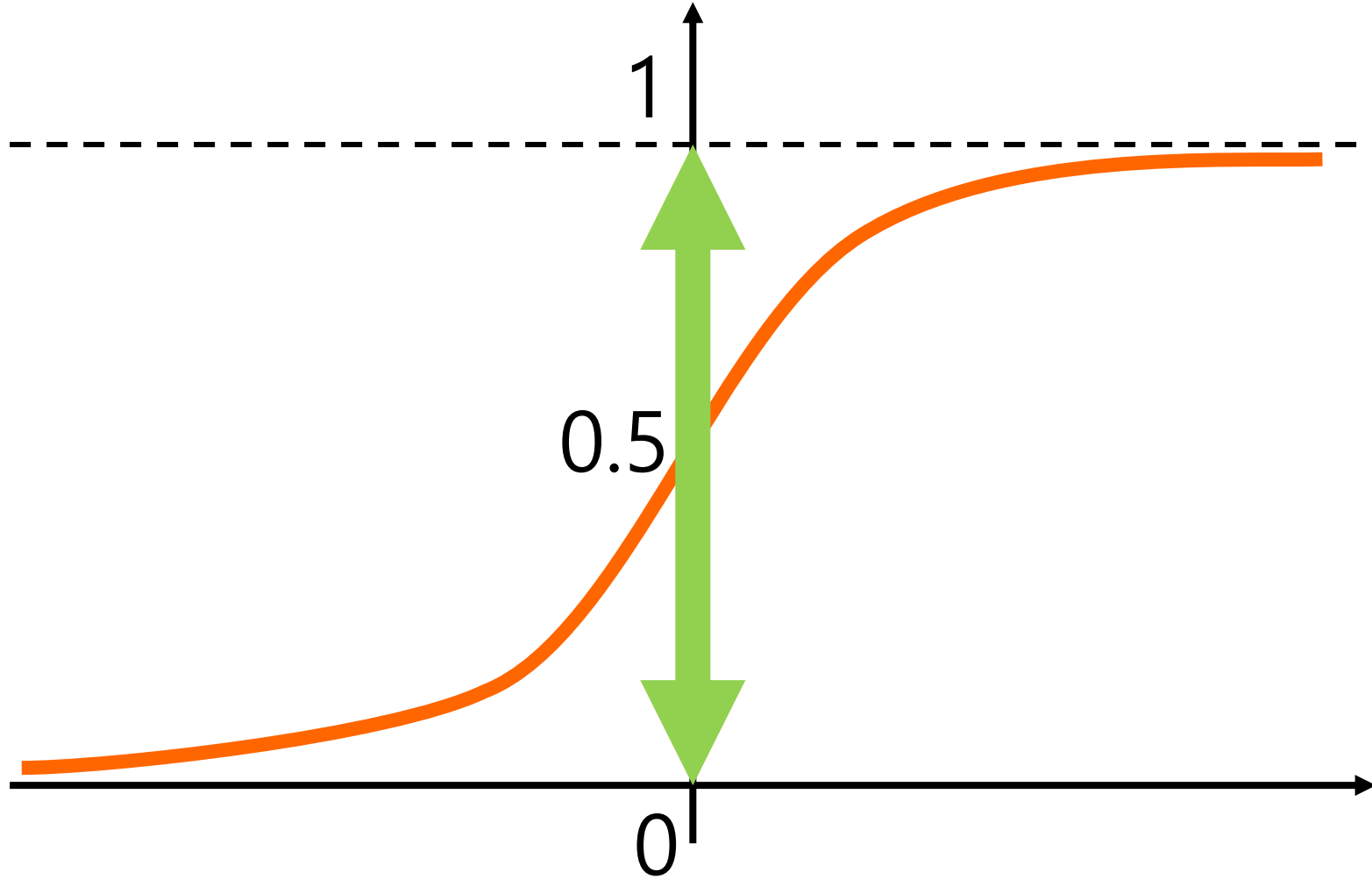
이 현상이 나중 영상에서 다룰 기울기 소실 문제 (vanishing gradient problem)의 근본적인 이유가 됩니다



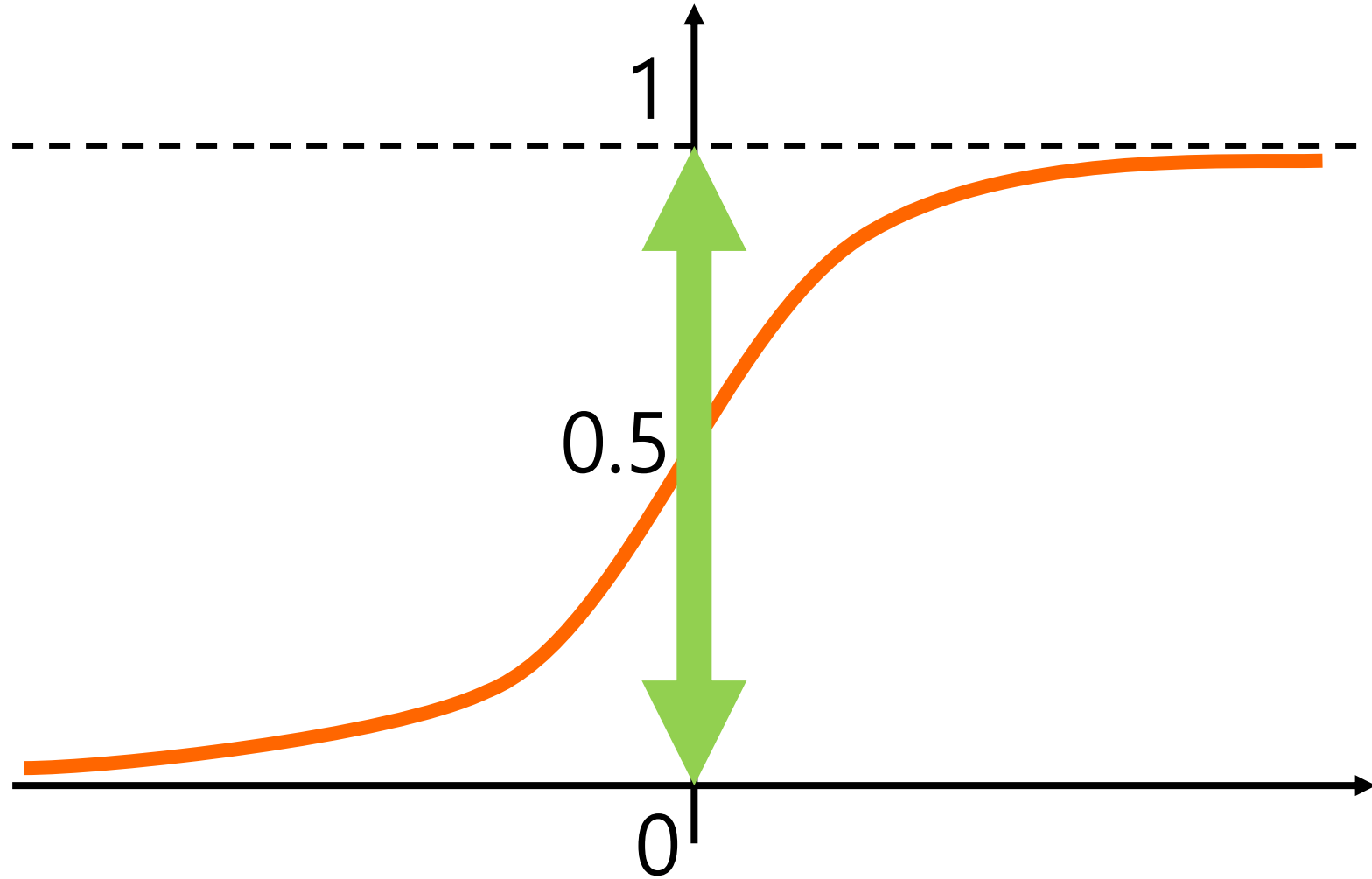
그리고 두번째 약점은 시그모이드 출력값이 항상 양수(+) 라는 것입니다



이것은 non zero-centered 문제라고 불리는 것인데요



시그모이드 출력값이 항상 양수(+) 라는 것이 왜 문제가 될까요?

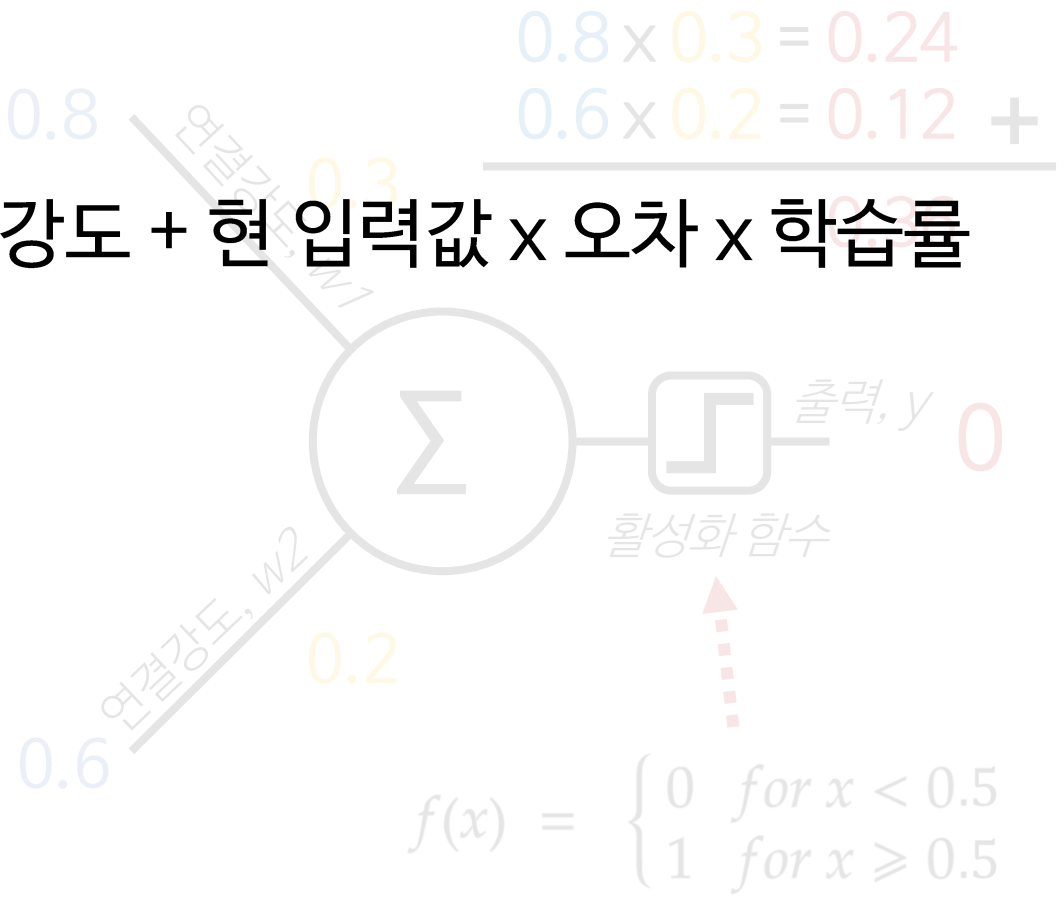


이해를 돕기 위해 두번째 강의 영상의 슬라이드를 잠시 빌려오겠습니다

x1	x2
0.1	0.2
0.3	0.1
0.6	0.6
0.4	0.3
0.1	0.2

퍼셉트론의 학습방법:

새 연결강도 = 현 연결강도 + 현 입력값 x 오차 x 학습률



y
1
1
0
0
1
0
0

y'
0

여기서 퍼셉트론의 학습방법은 다음과 같습니다

퍼셉트론의 학습방법:

새 연결강도 = 현 연결강도 + 현 입력값 x 오차 x 학습률

물론 다층 신경망의 학습방법은 퍼셉트론 보다 훨씬 복잡하지만

퍼셉트론의 학습방법:

새 연결강도 = 현 연결강도 + 현 입력값 x 오차 x 학습률

개념적으로는 퍼셉트론이나 다층신경망이나 동일하게

다층신경망의 학습방법:

새 연결강도의 변화량 \propto [현 입력값 \times 오차]

연결강도의 변화는 입력값 (혹은 입력값과 관련된 기울기)과 오차의 곱에 비례한다고 볼수 있습니다

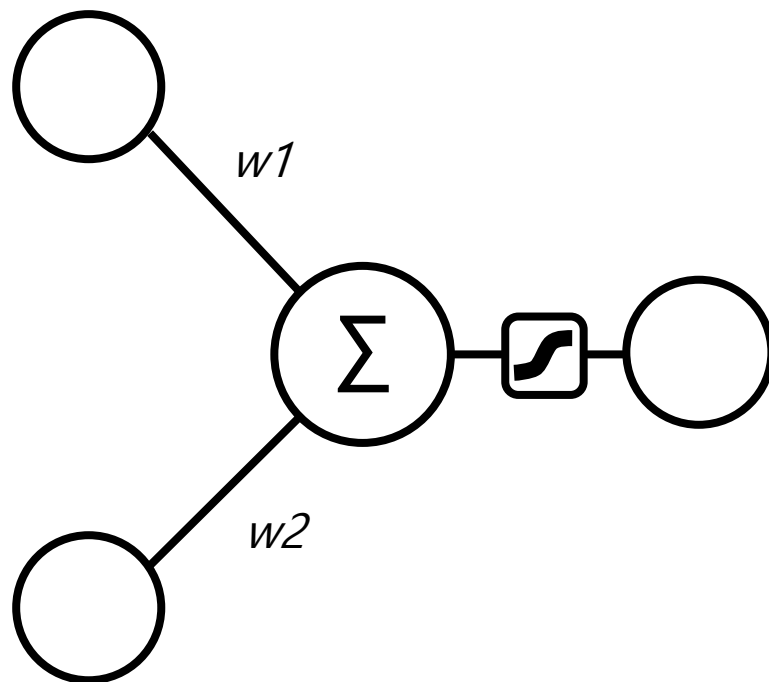
다층신경망의 학습방법:

새 연결강도의 변화량 \propto [현 입력값 \times 오차]

그래서 간단한 신경망을 통해서 살펴보자면..

다층신경망의 학습방법:

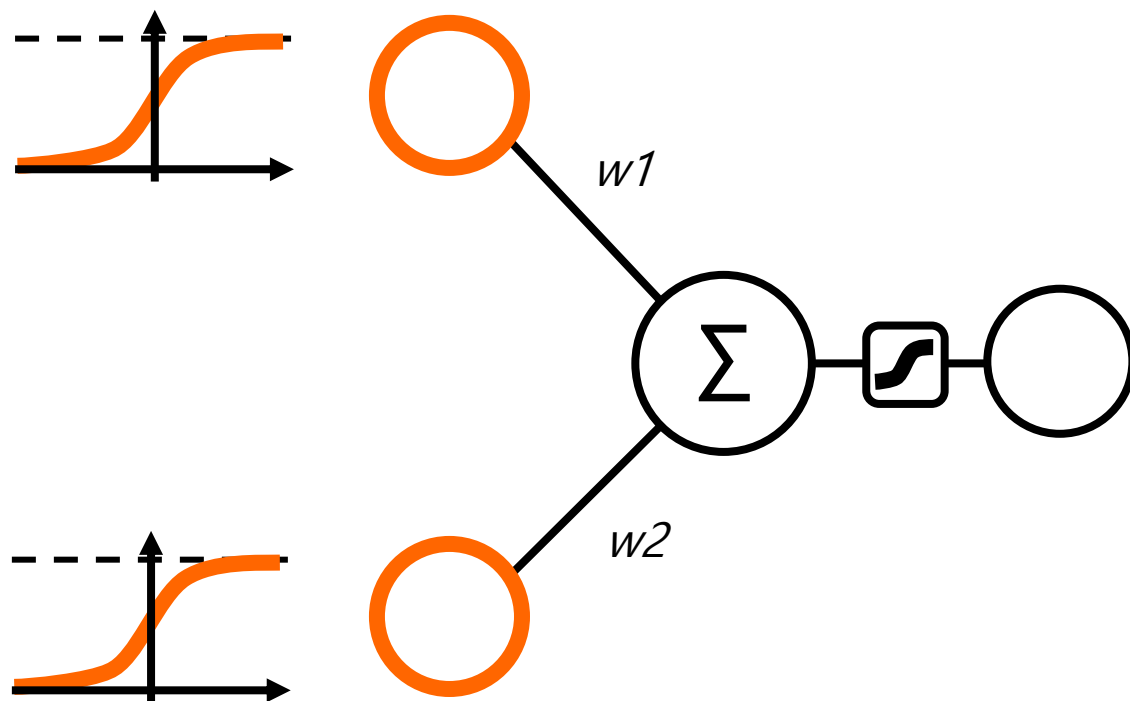
새 연결강도의 변화량 \propto [현 입력값 \times 오차]



시그모이드 함수에 의해 입력값이 양수가 되고..

다층신경망의 학습방법:

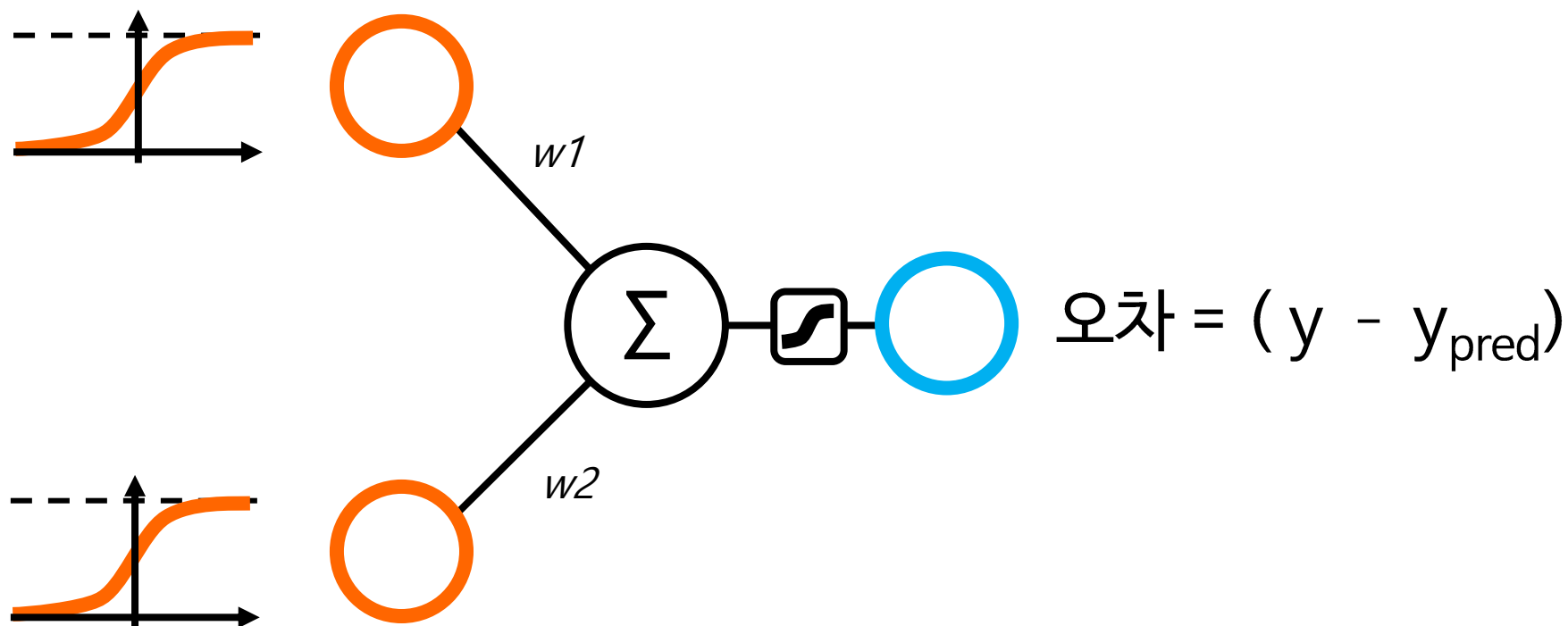
새 연결강도의 변화량 \propto [현 입력값 x 오차]



만약에 출력값의 오차가 음수가 된다면..

다층신경망의 학습방법:

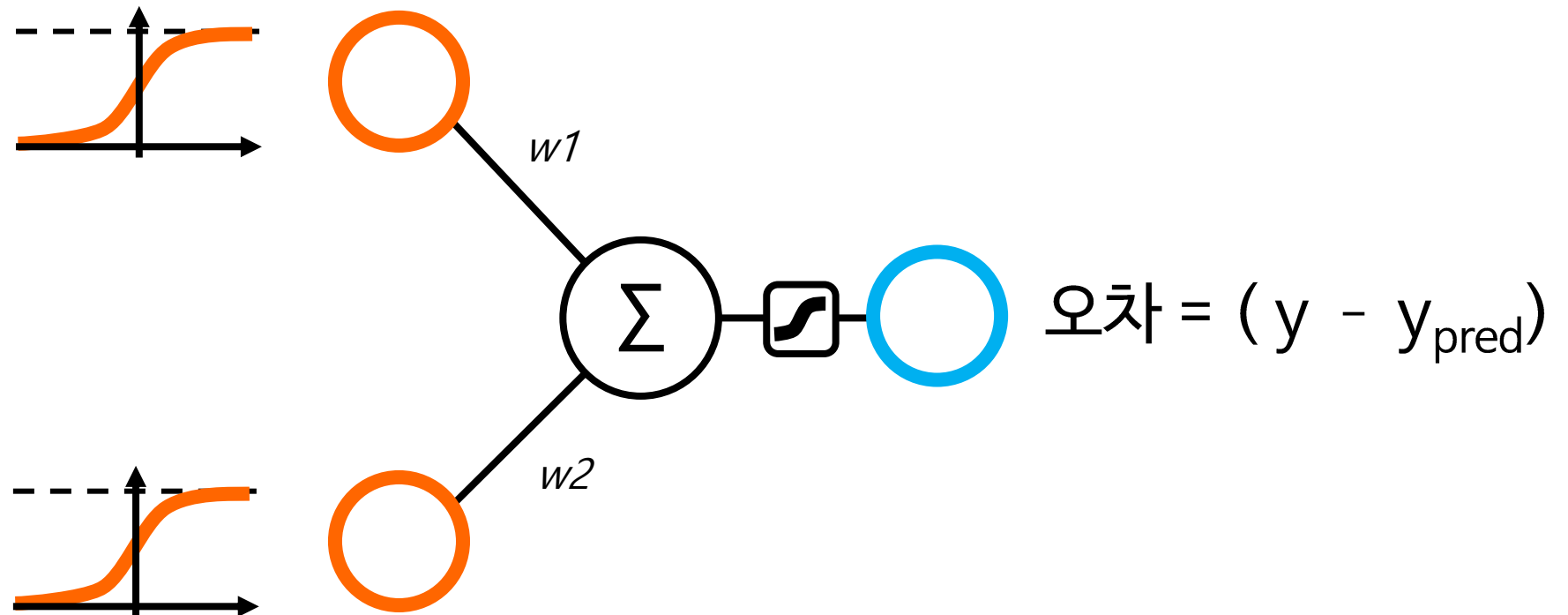
새 연결강도의 변화량 \propto [현 입력값 \times 오차]



현 입력값과 오차의 곱에 의해서..

다층신경망의 학습방법:

새 연결강도의 변화량 \propto $\begin{matrix} (+) & (-) \\ \text{현 입력값} \times \text{오차} \end{matrix}$



두 연결강도의 변화량의 부호가 '동일하게' 마이너스로 바뀌게 되고

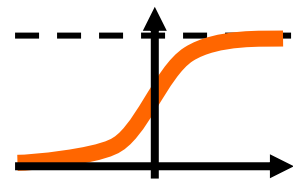
다층신경망의 학습방법:

(-)

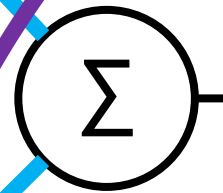
(+)

(-)

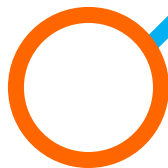
새 연결강도의 변화량 \propto [현 입력값 \times 오차]



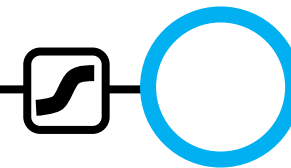
$w1$



$w2$



두 연결강도의 변화량이 동일하게 음수가 되고

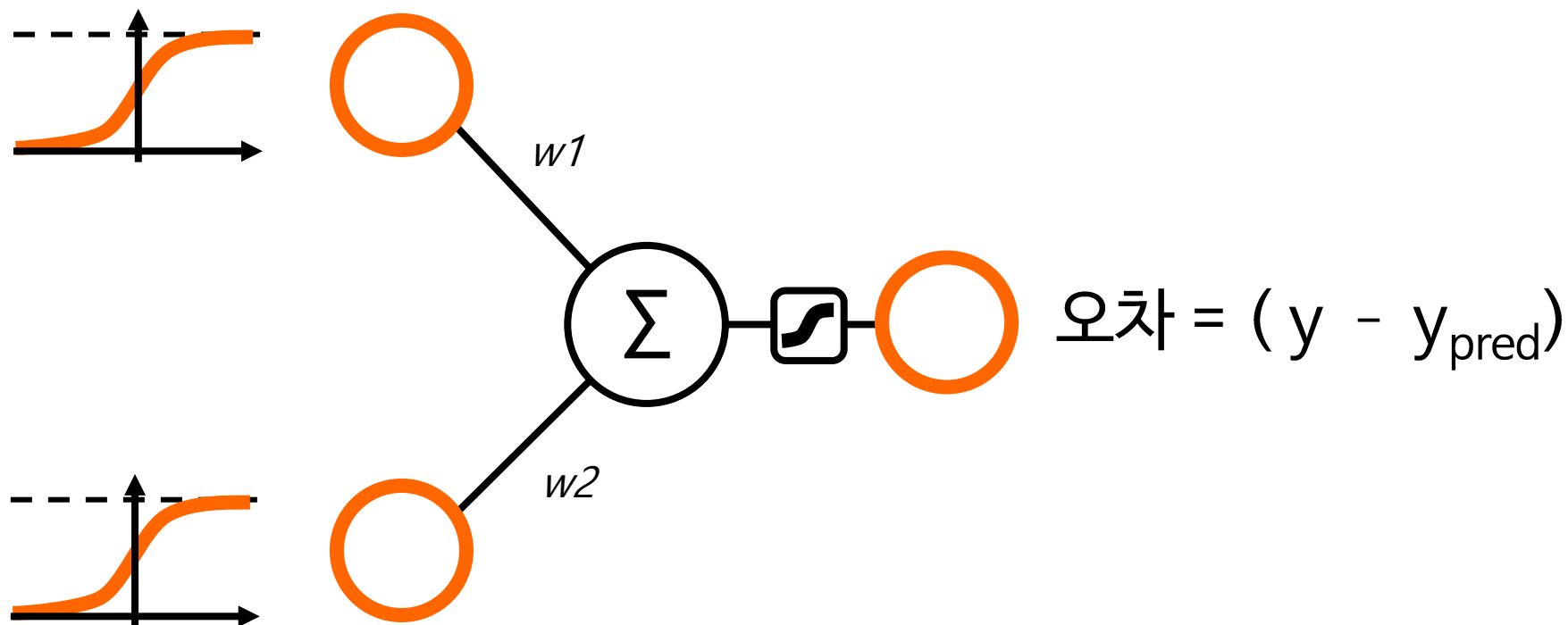


오차 = $(y - y_{\text{pred}})$

만약에 출력값의 오차가 양수가 된다면..

다층신경망의 학습방법:

새 연결강도의 변화량 \propto $\overset{(+)}{\text{[현 입력값]}} \times \overset{(+)}{\text{오차}}$



두 연결강도의 변화량의 부호가 '동일하게' 플러스로 바뀌게 됩니다

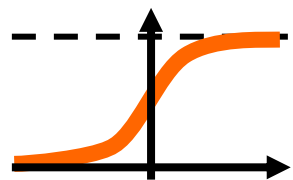
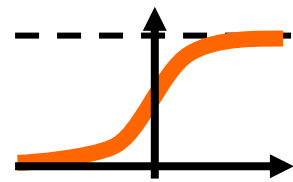
다층신경망의 학습방법:

(+)

(+)

(+)

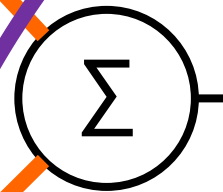
새 연결강도의 변화량 \propto [현 입력값 \times 오차]



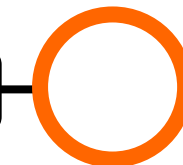
$w1$



$w2$



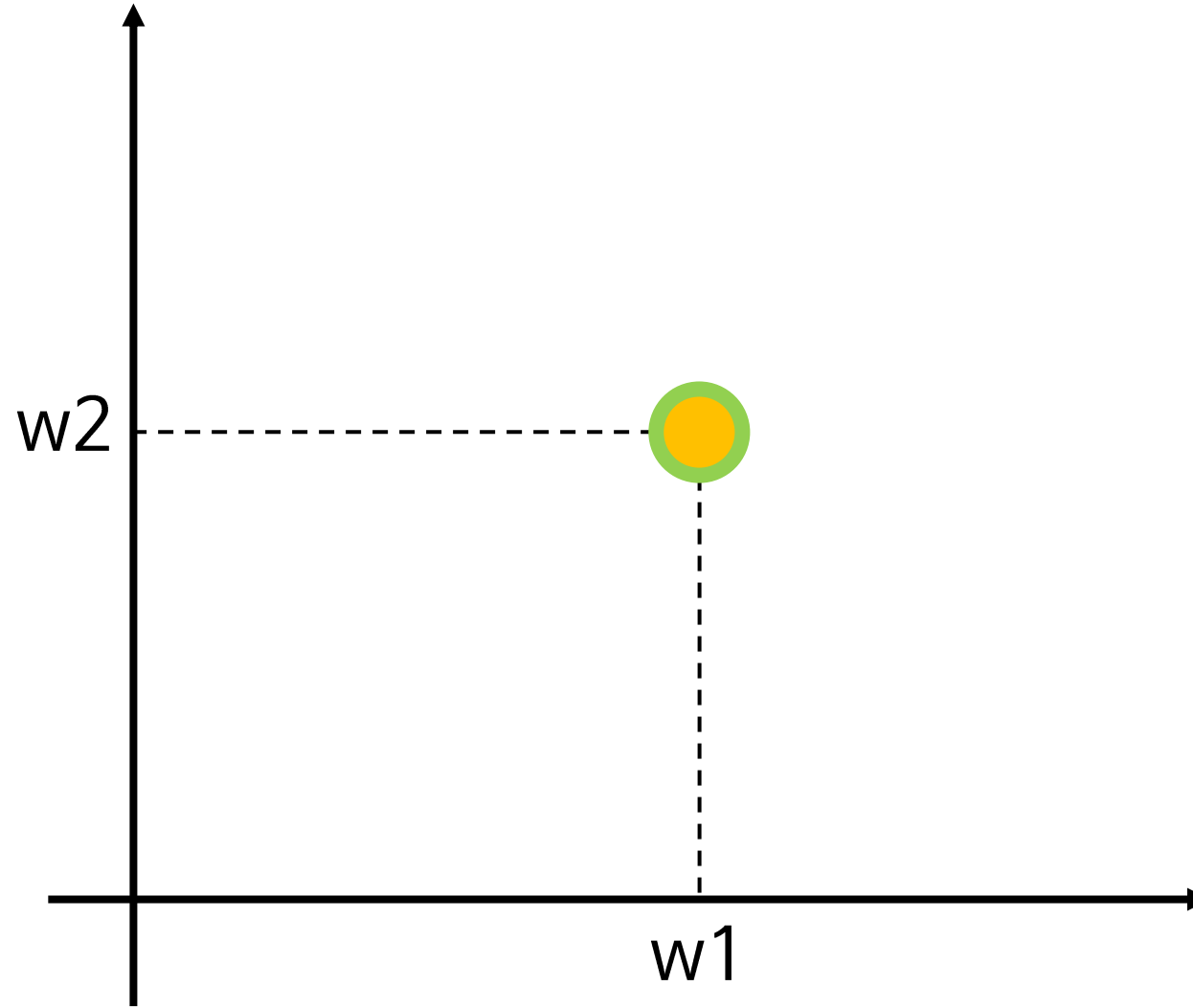
두 연결강도의 변화량이 동일하게 양수가 됩니다



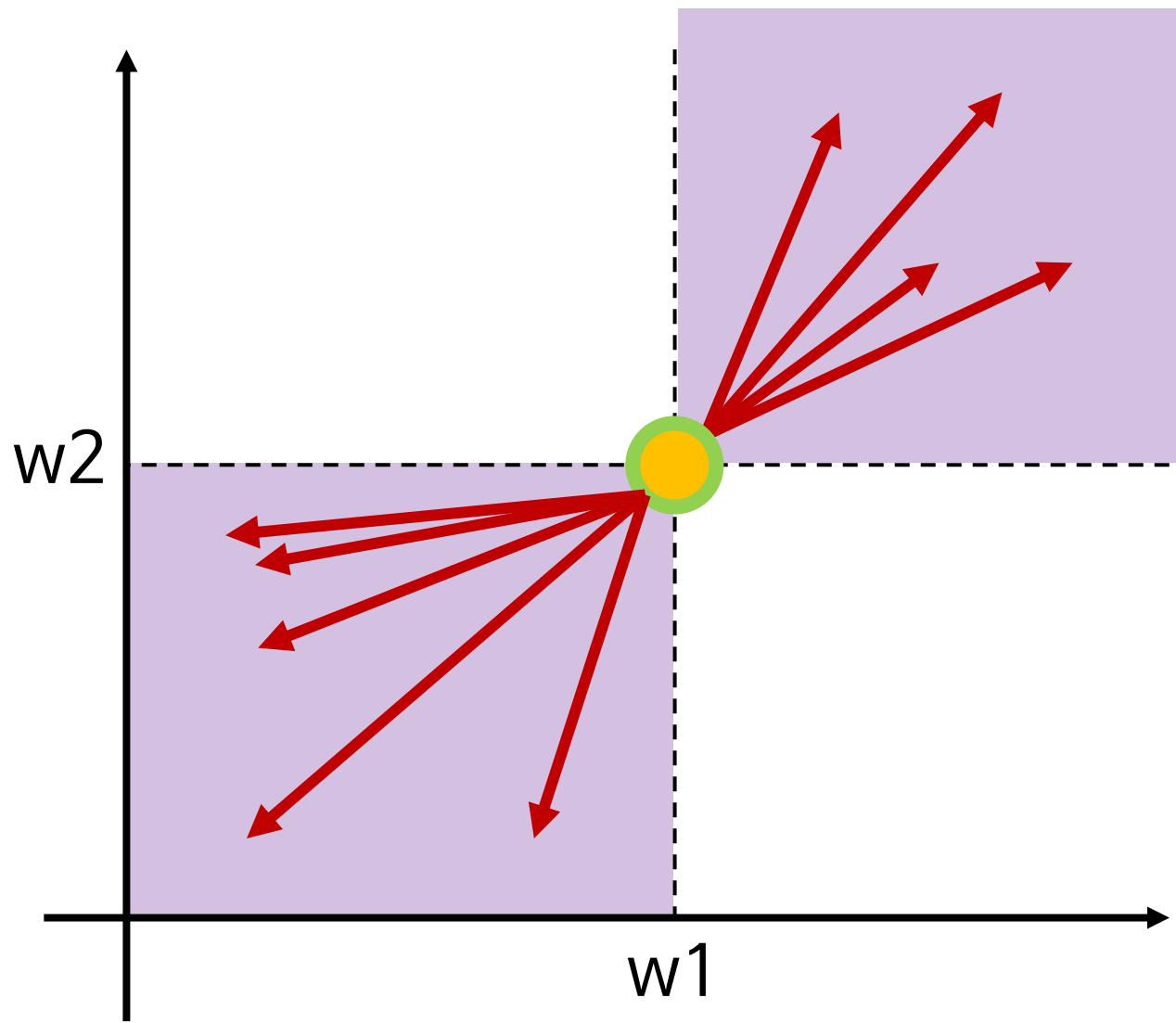
$$\text{오차} = (y - y_{\text{pred}})$$

두 연결강도의 부호 (+ 혹은 -)가 동일하게 변한다는 말은,

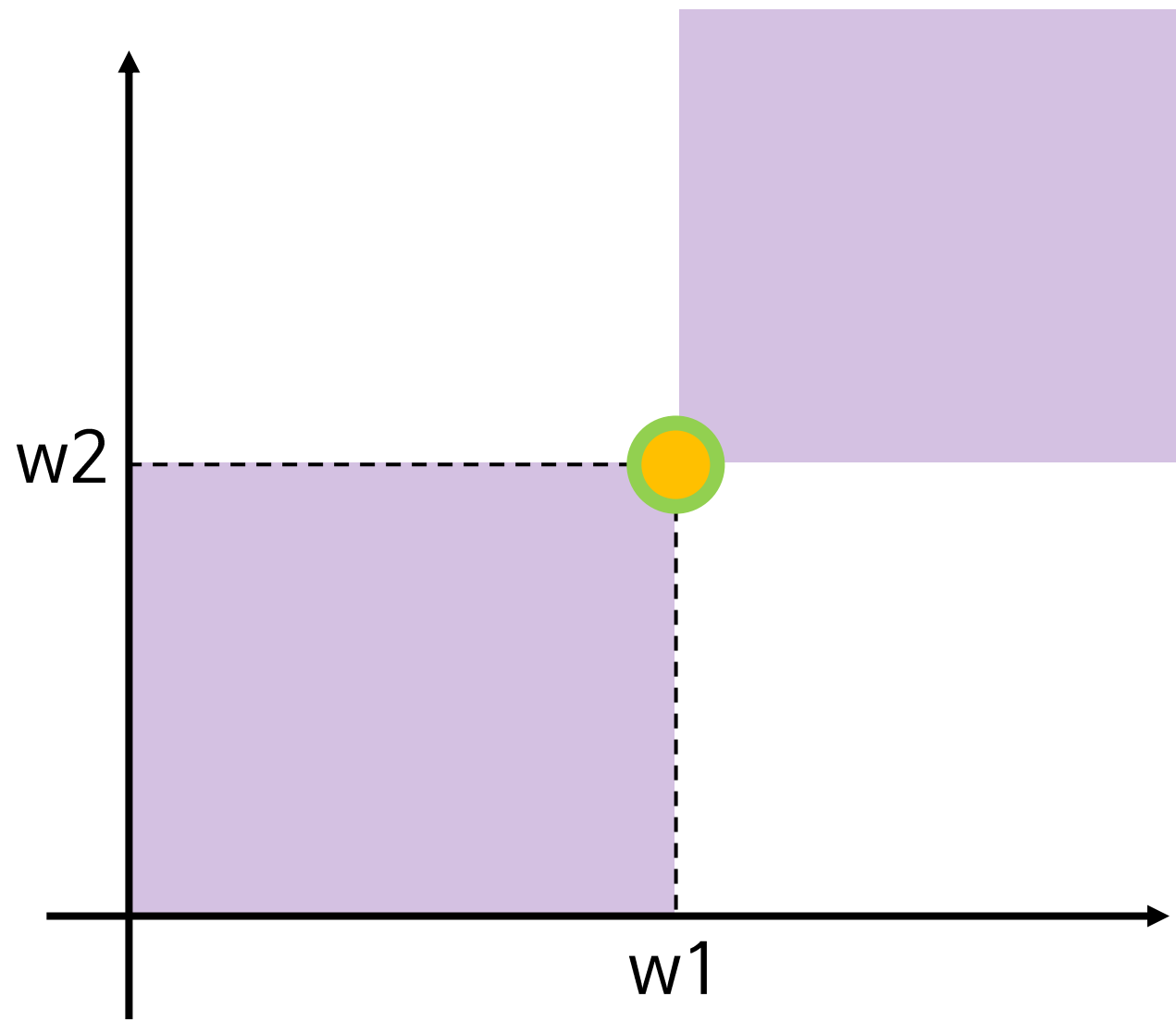
이와 같은 w_1 , w_2 평면에서



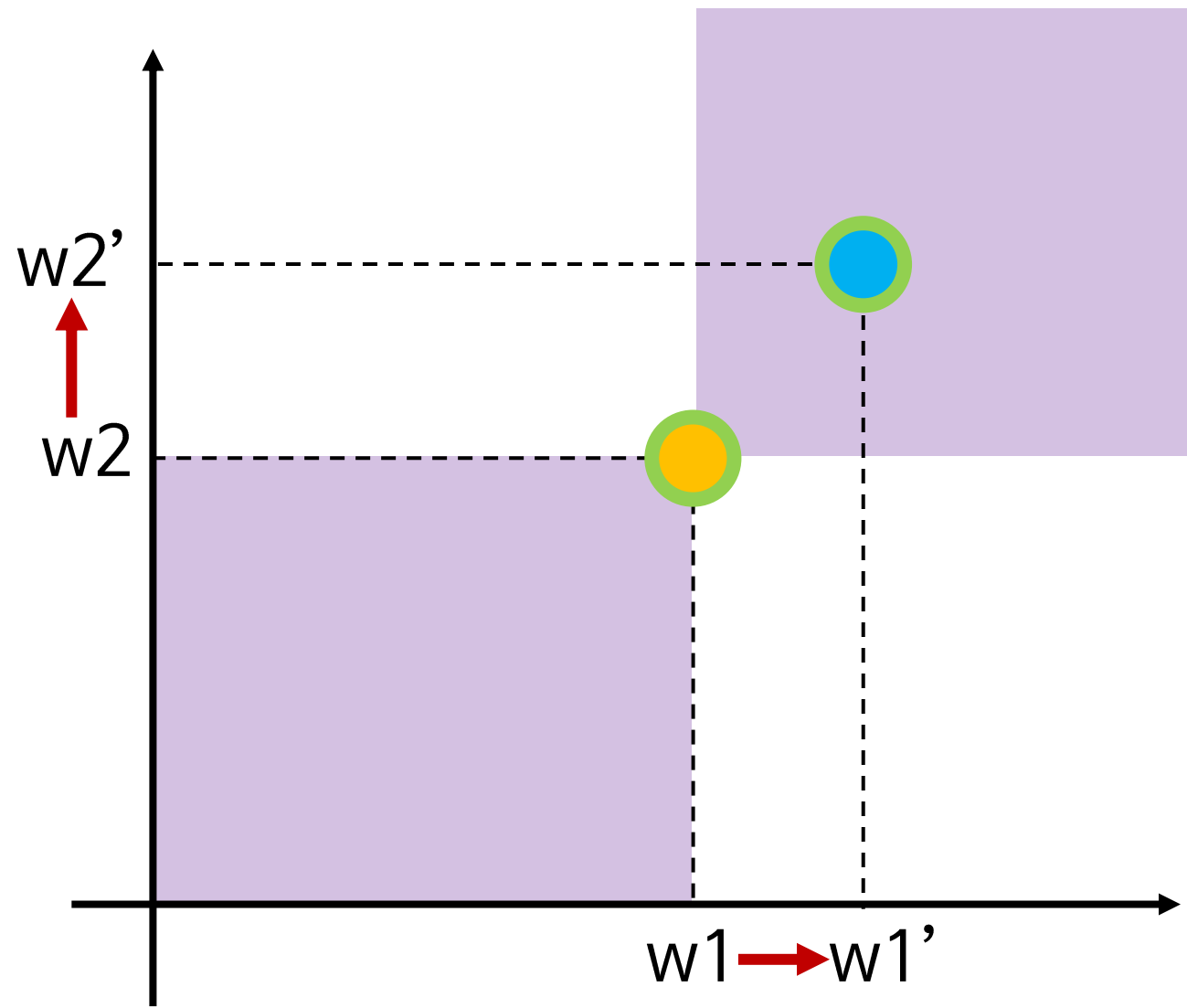
새로운 w_1 , w_2 는 다음과 같은 지역으로만 움직일 수 있다는 뜻입니다



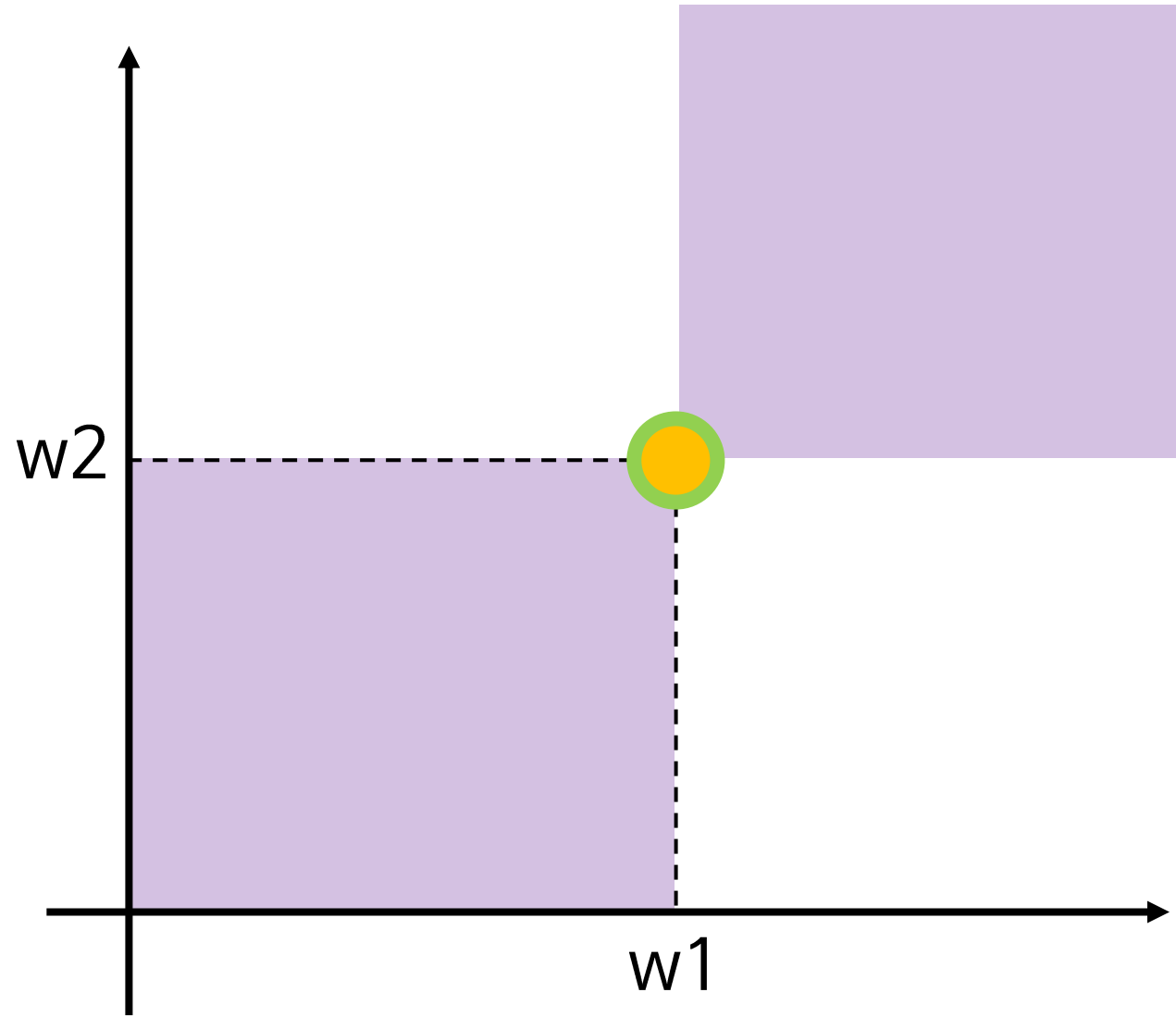
왜냐하면 이와같은 w_1 , w_2 평면에서 둘다 변화량이 양수일 경우



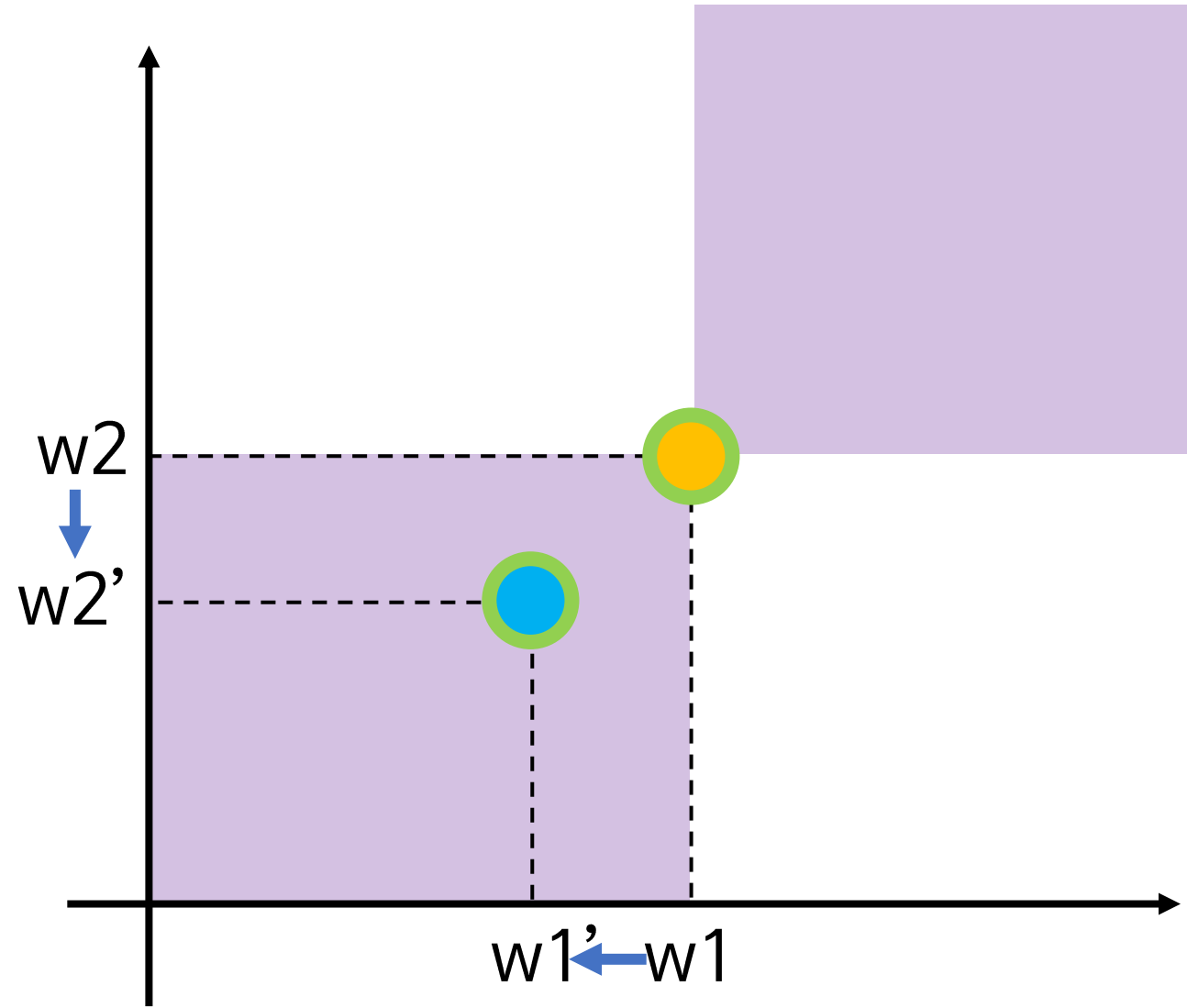
새로운 연결 가중치 $w1'$, $w2'$ 는 다음과 같은 위치가 되고



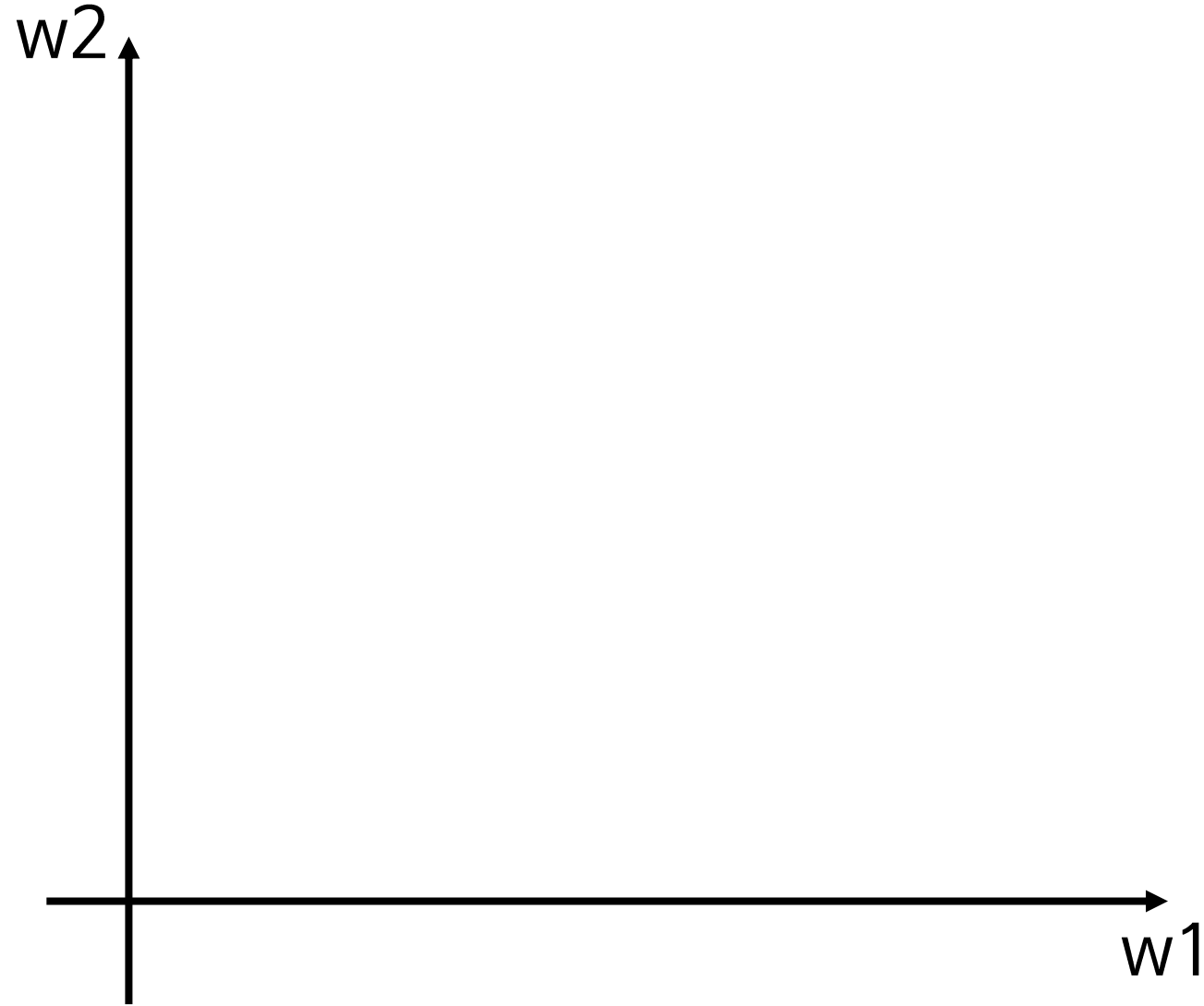
또 이와같은 w_1 , w_2 평면에서 둘다 변화량이 음수일 경우



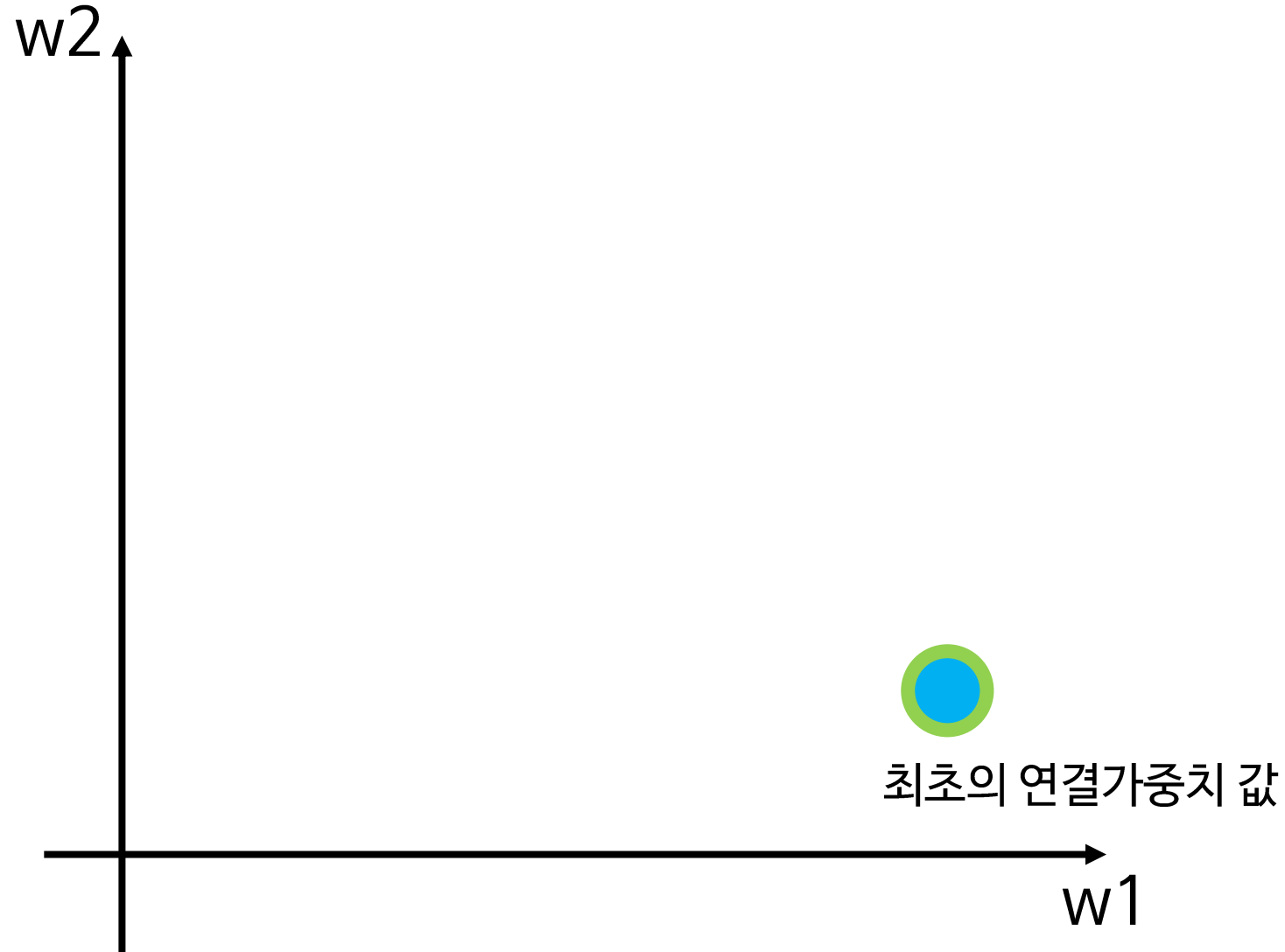
새로운 연결 가중치 $w1'$, $w2'$ 는 다음과 같은 위치가 되기 때문입니다



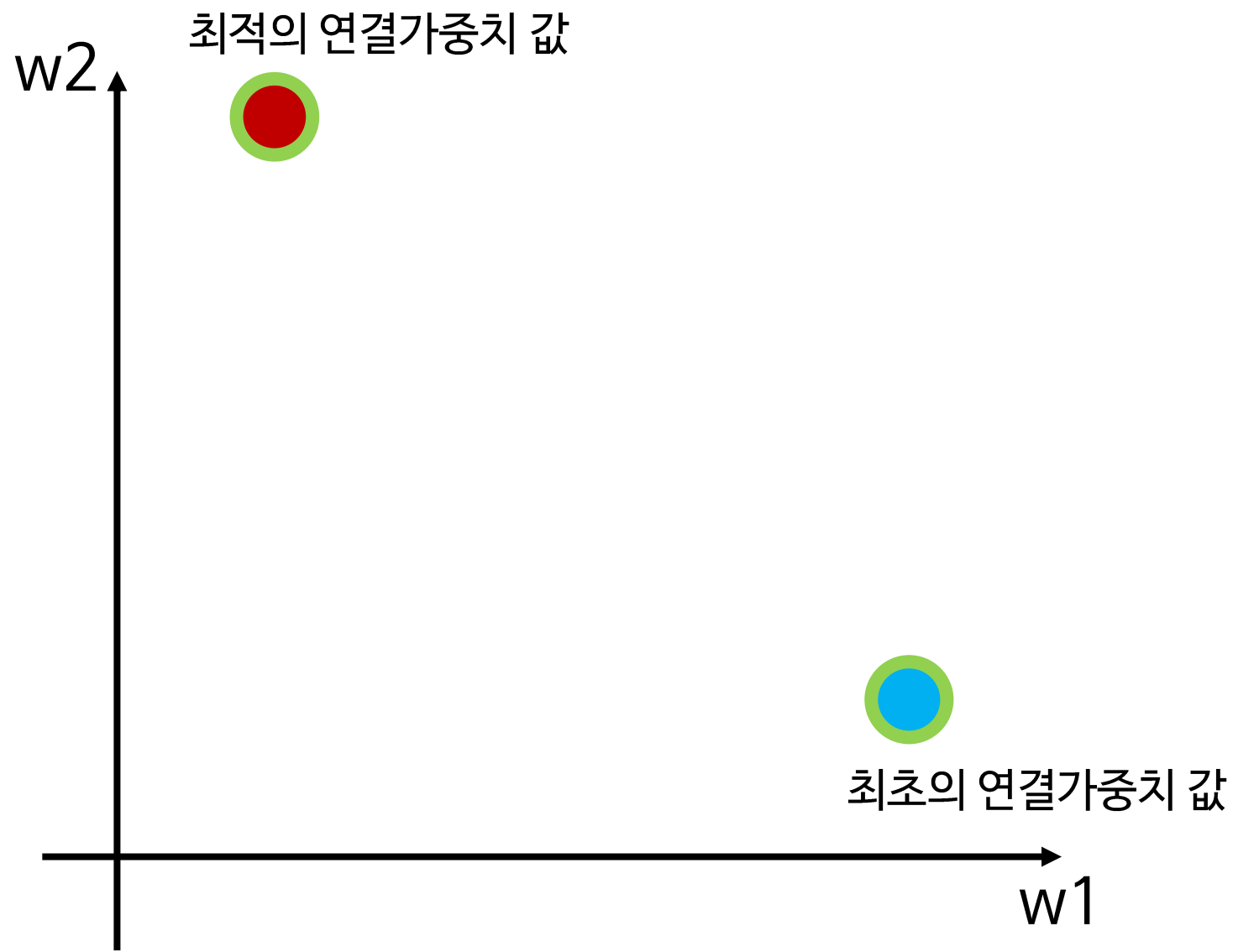
그래서 이런 w_1 , w_2 공간을 다시 가정해봅시다



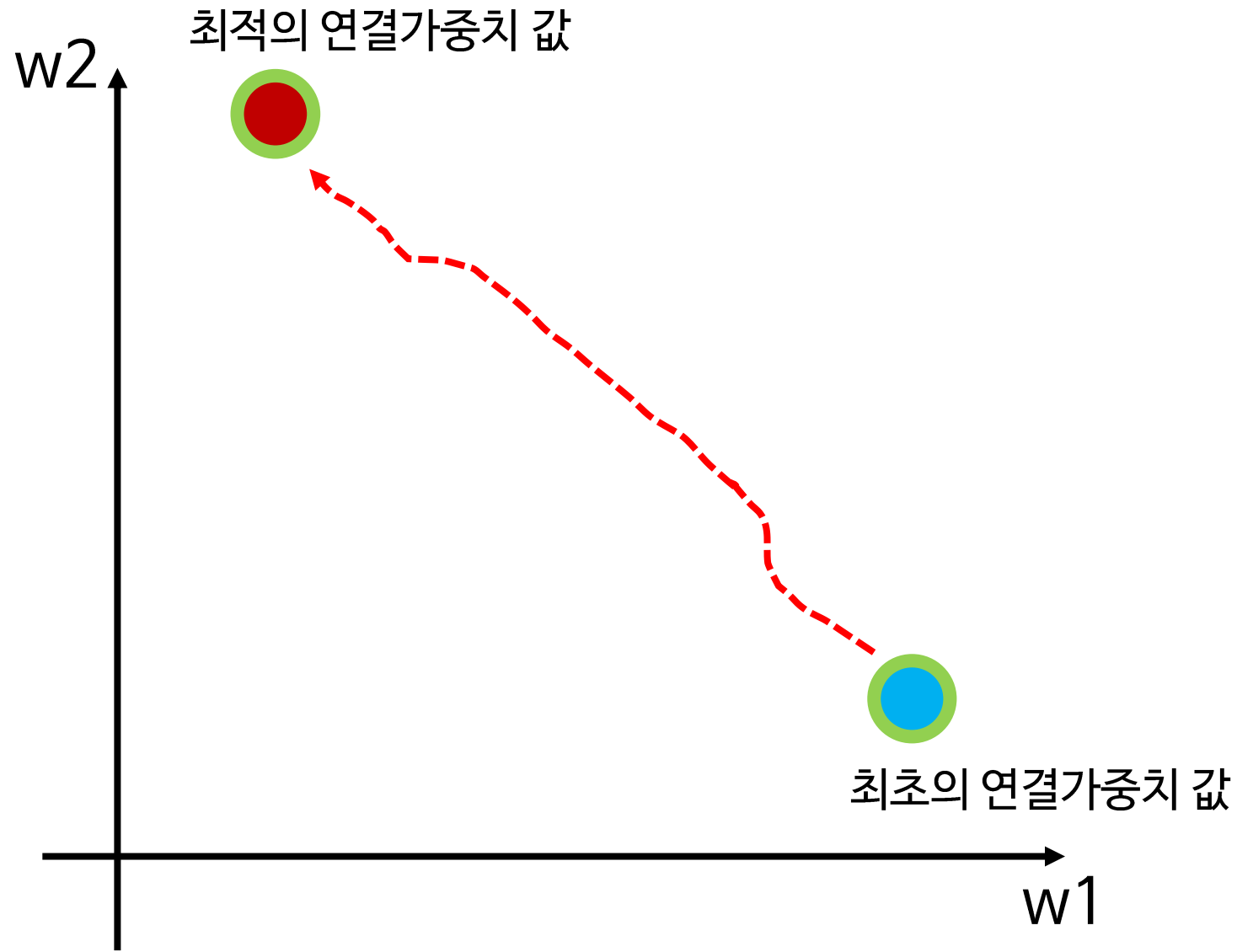
그리고 다음과 같이 최초의 연결가중치 값이 다음과 같고,



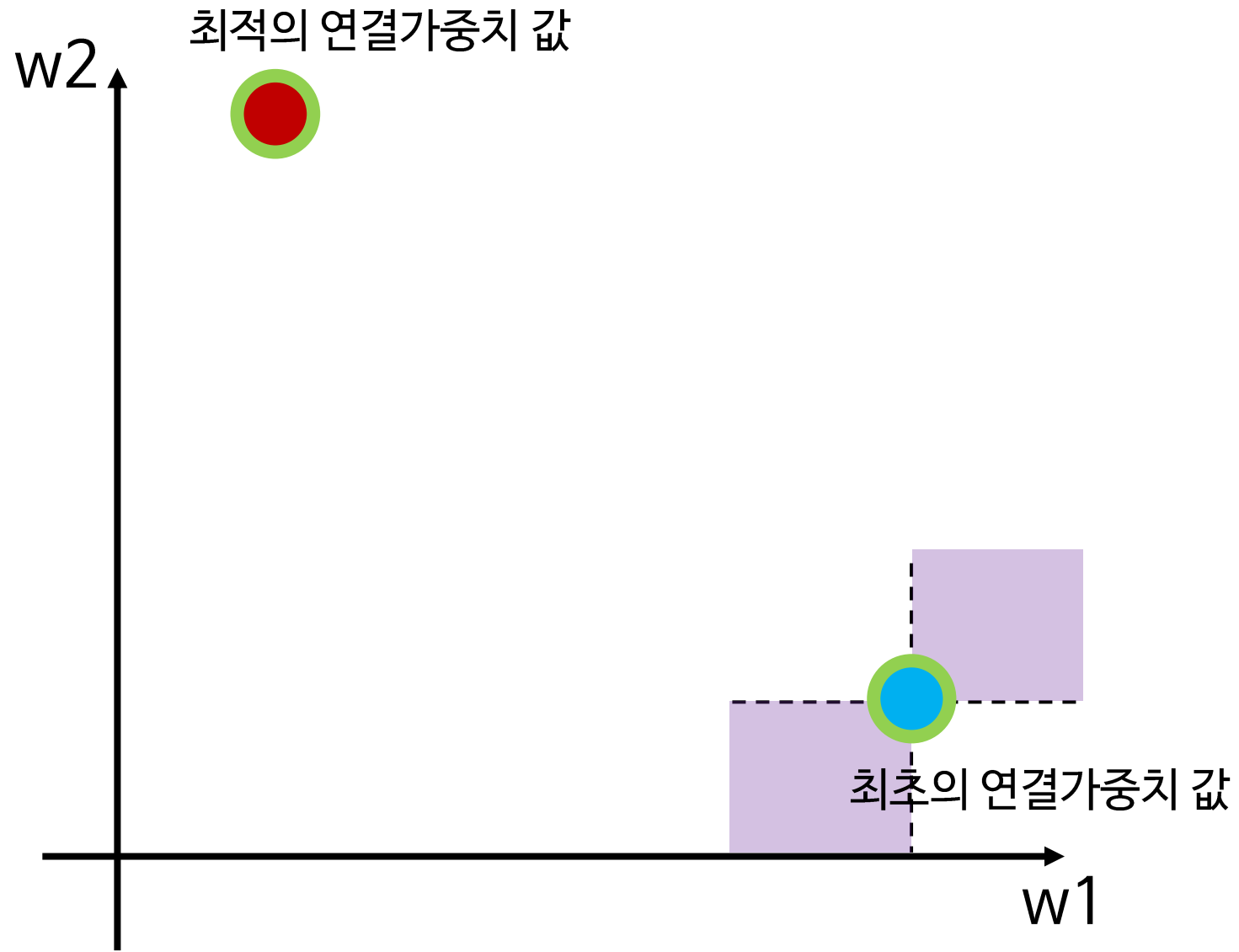
학습을 통해 도달해야할 최적의 연결가중치가 다음과 같다면,



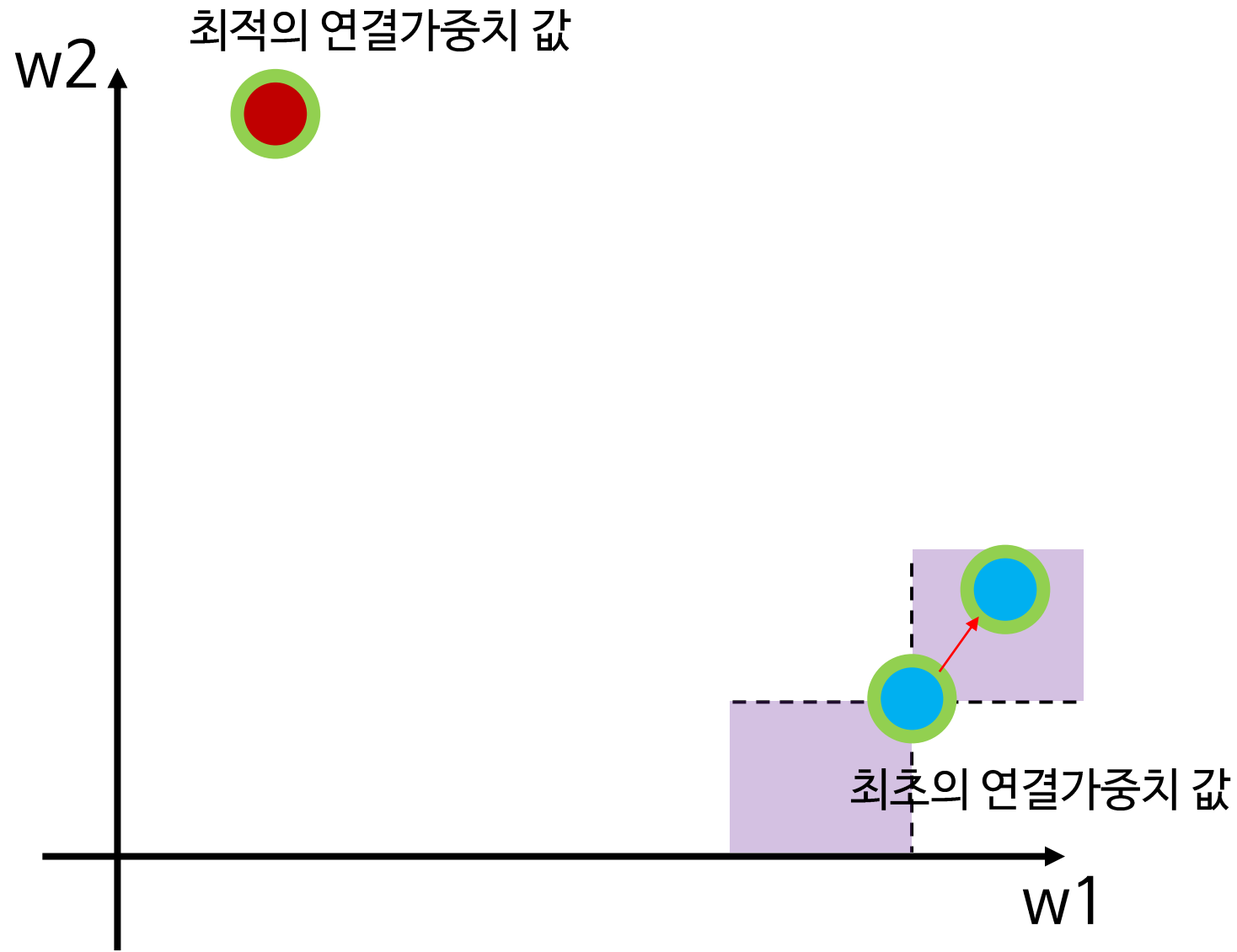
다음과 같이 좌상향으로 점진적으로 w_1 과 w_2 값이 변해가면 좋겠지만..



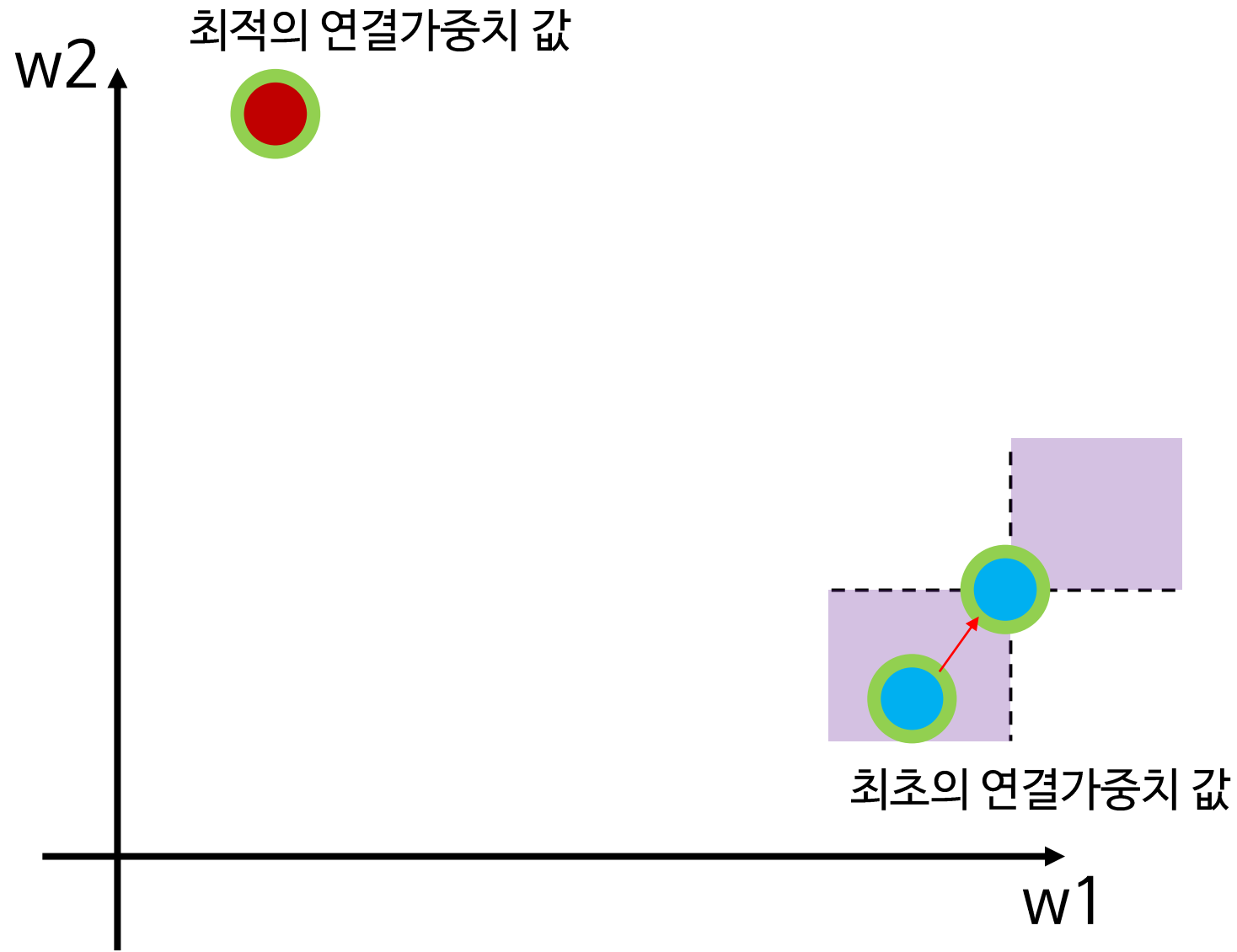
학습과정에서 $w1$ 과 $w2$ 의 변화에는 이와같은 제약이 있으므로..



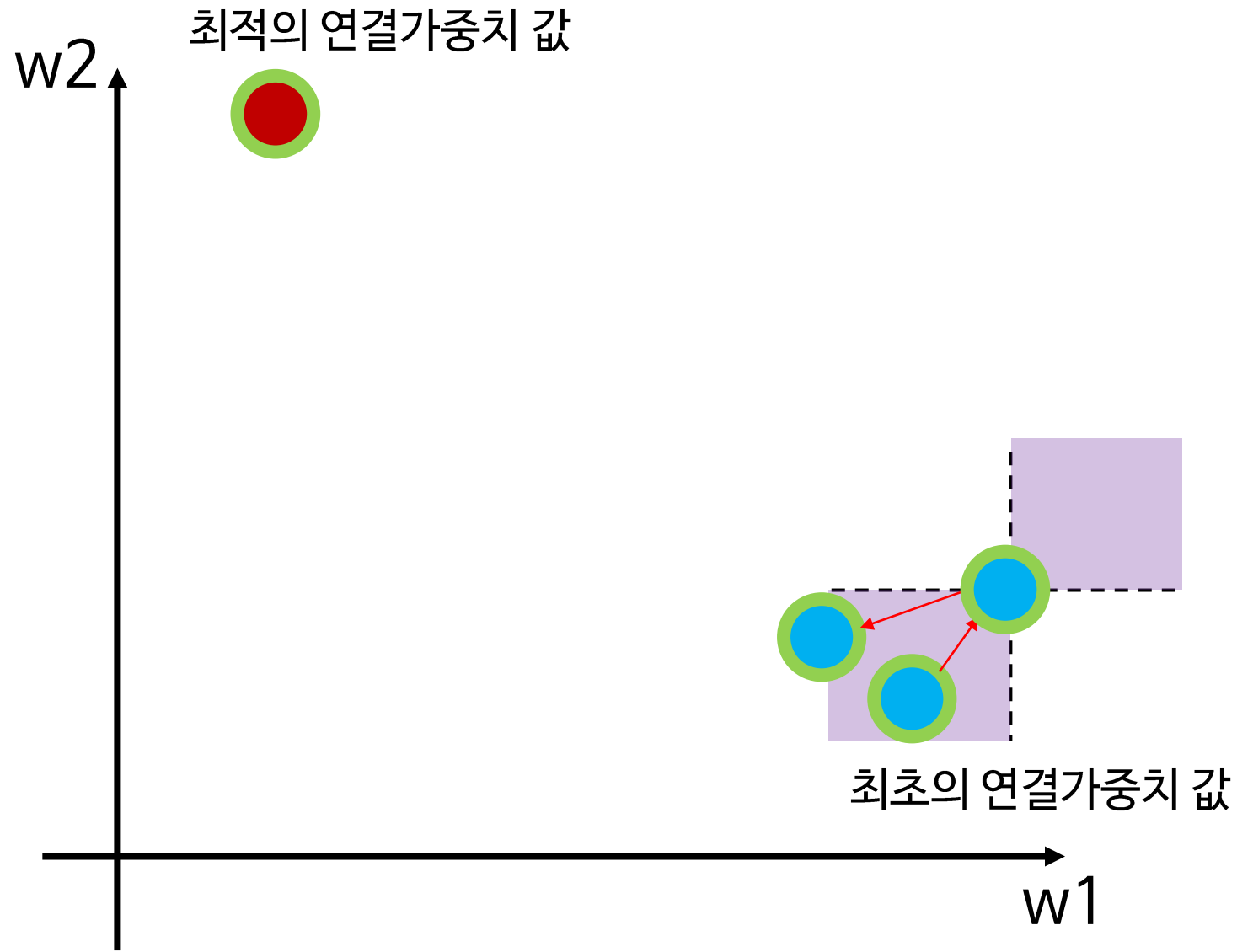
새로운 w_1, w_2 가 이렇게 업데이트가 되고



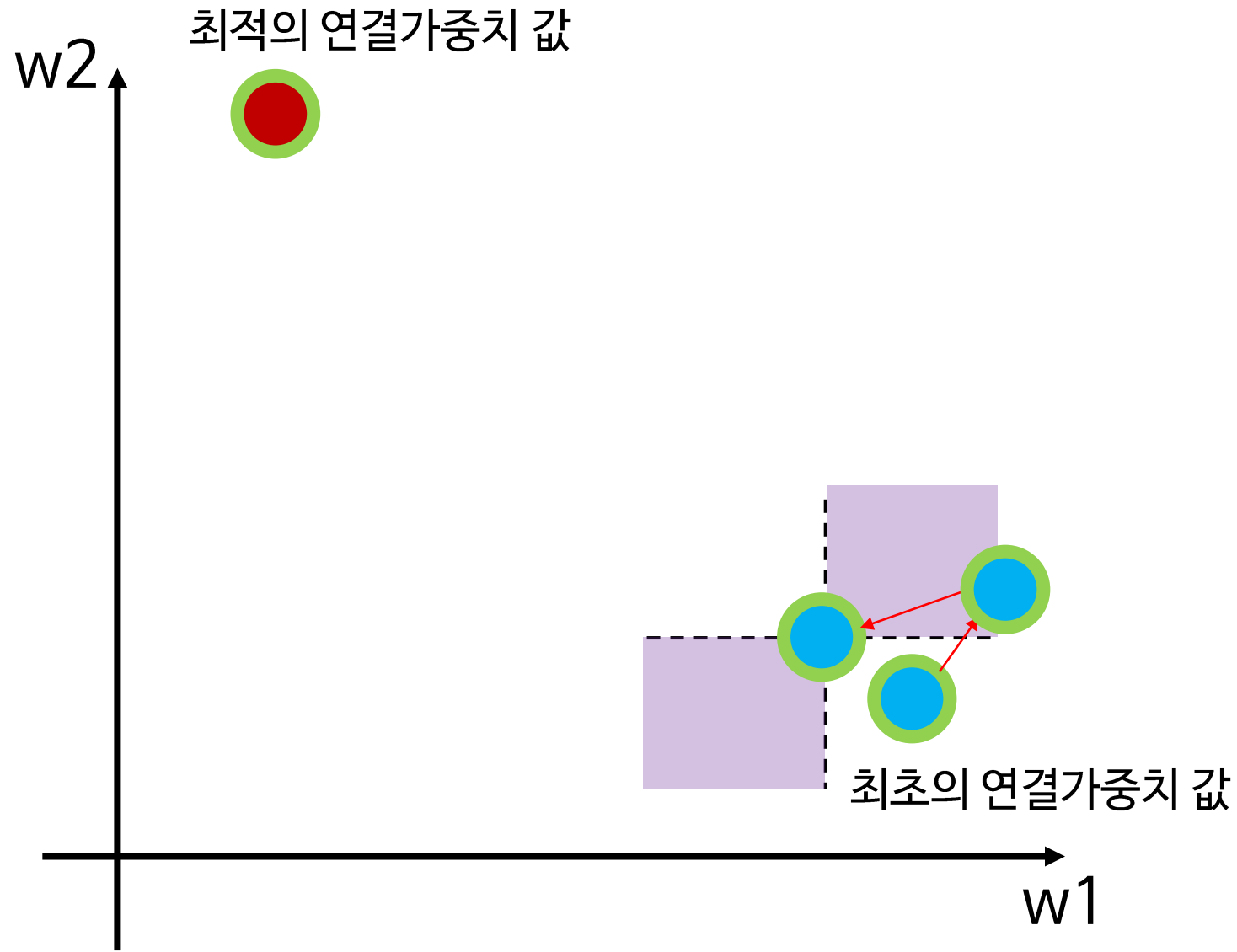
또 여기에서 다음번 w_1, w_2 가 가능한 지역은 보라색, 즉 1과 3사분면



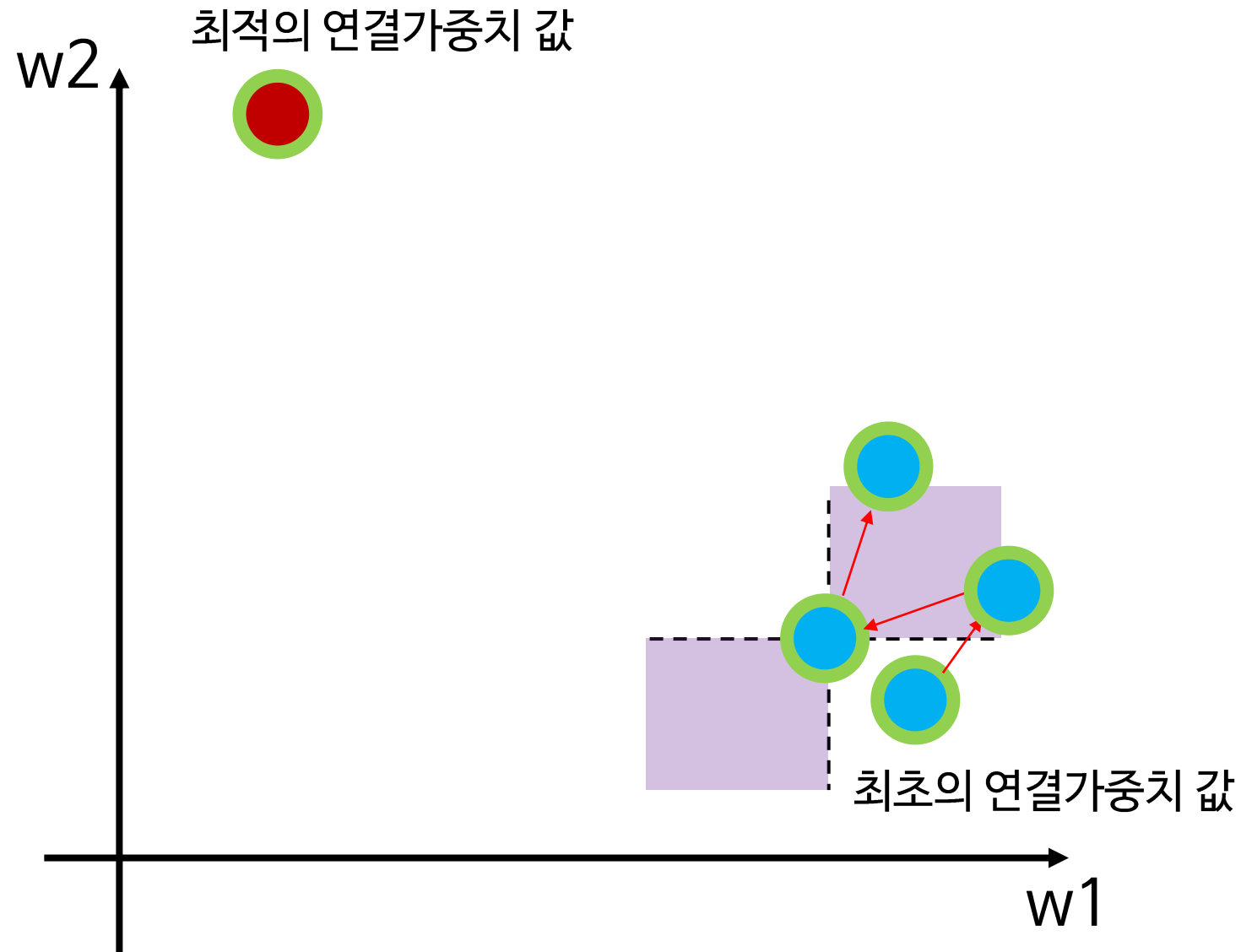
그러므로 다시 새로운 w_1, w_2 가 다음과 같이 업데이트 될 수 있습니다



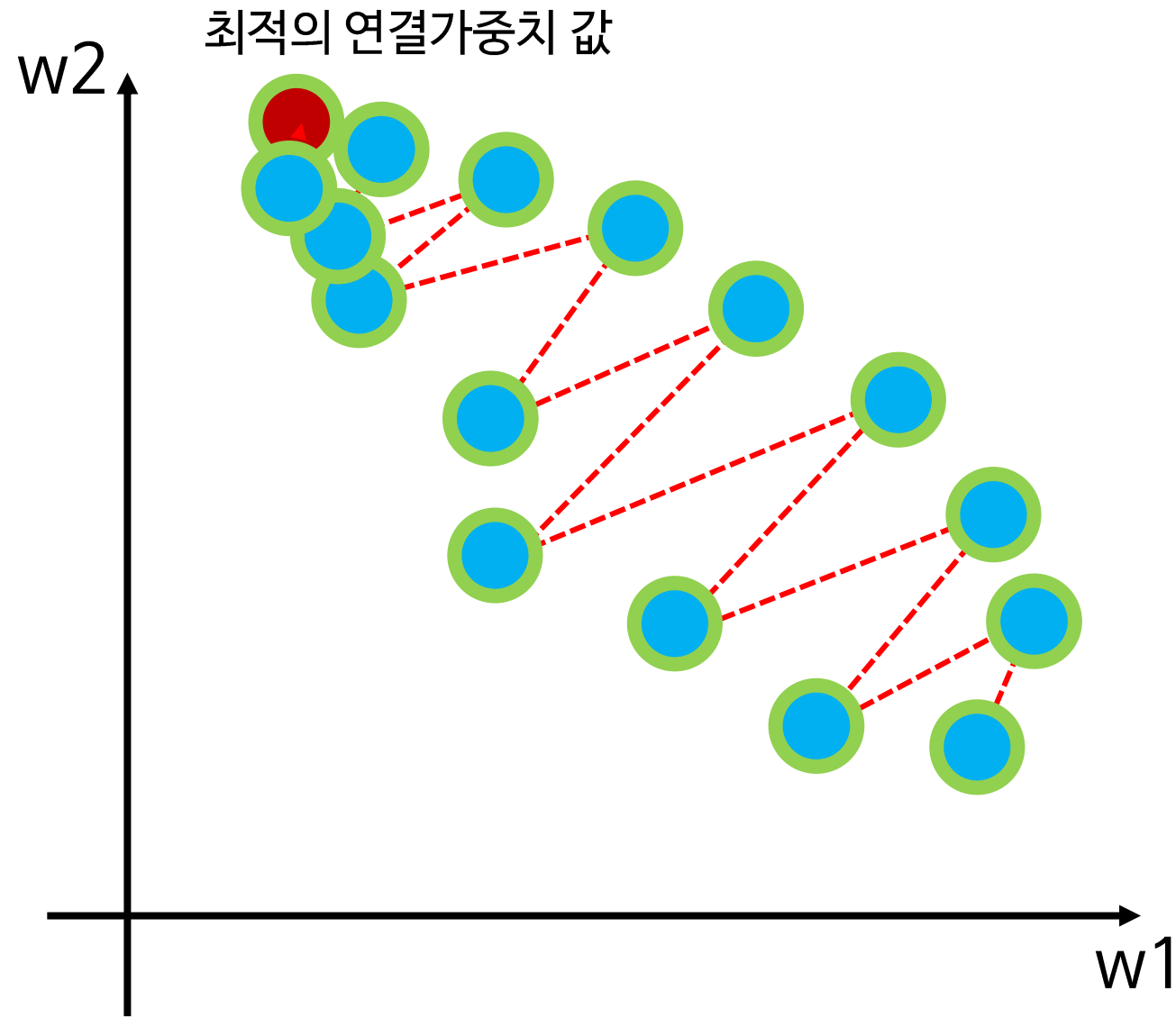
다시 또 이곳에서부터 1, 3사분면으로 새로운 w_1, w_2 가 나오기 때문에



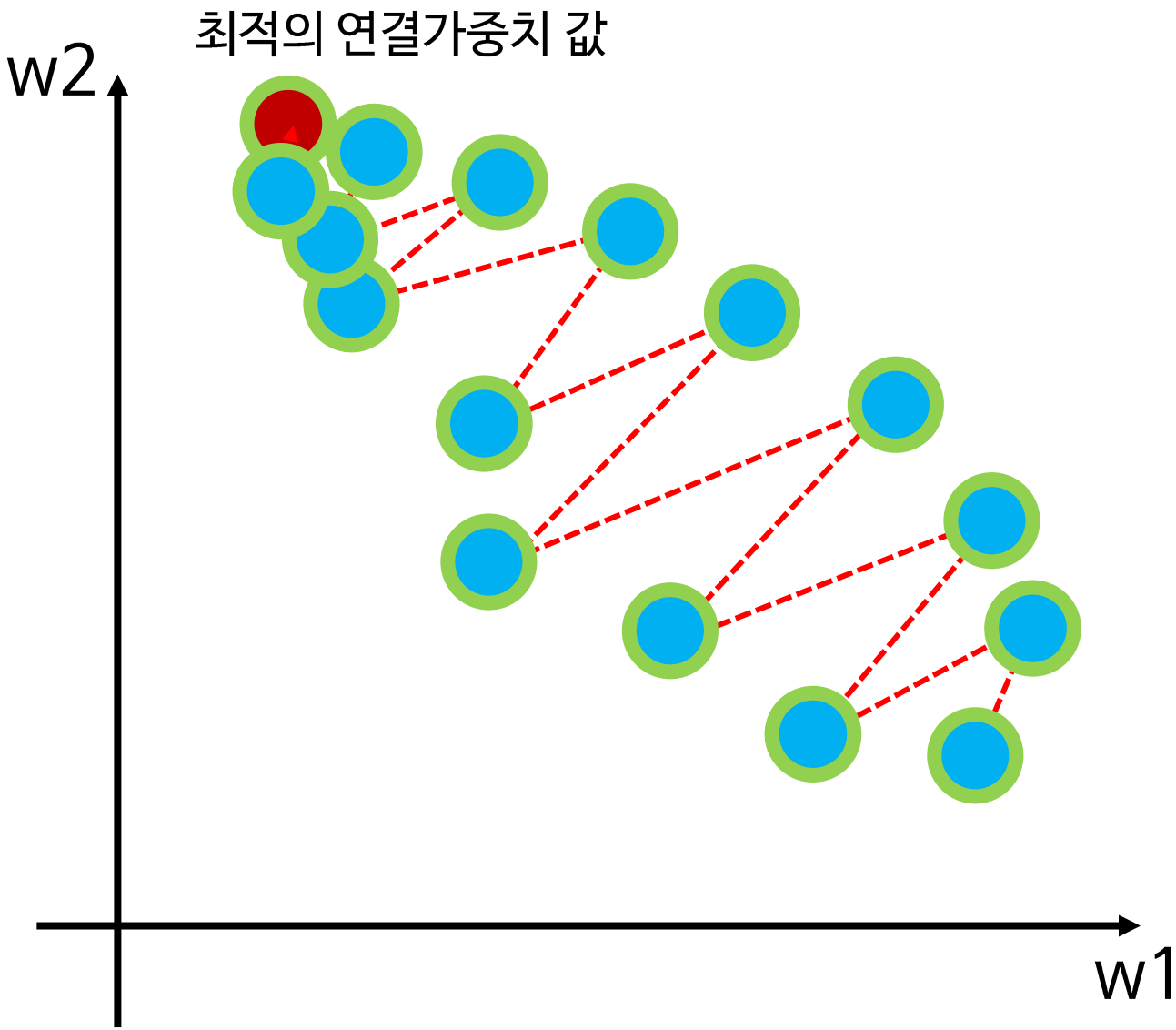
다시 또 이곳에서부터 1, 3사분면으로 새로운 w_1, w_2 가 나오기 때문에



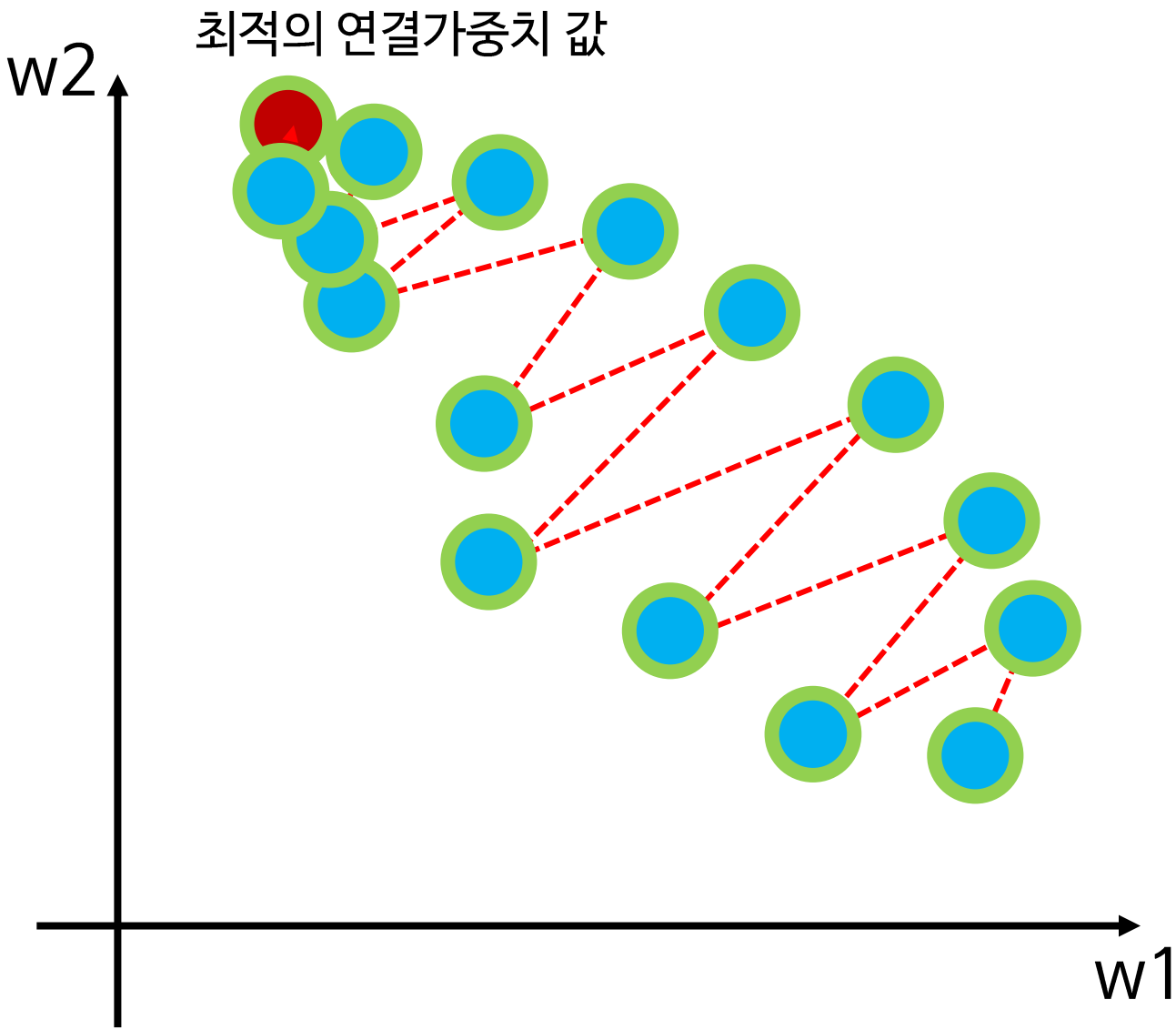
연결가중치의 학습과정이 아래와 같은 지그재그 패턴을 보이게 됩니다.



지그재그패턴은 결국 학습과정의 능률이 떨어진다는 말입니다



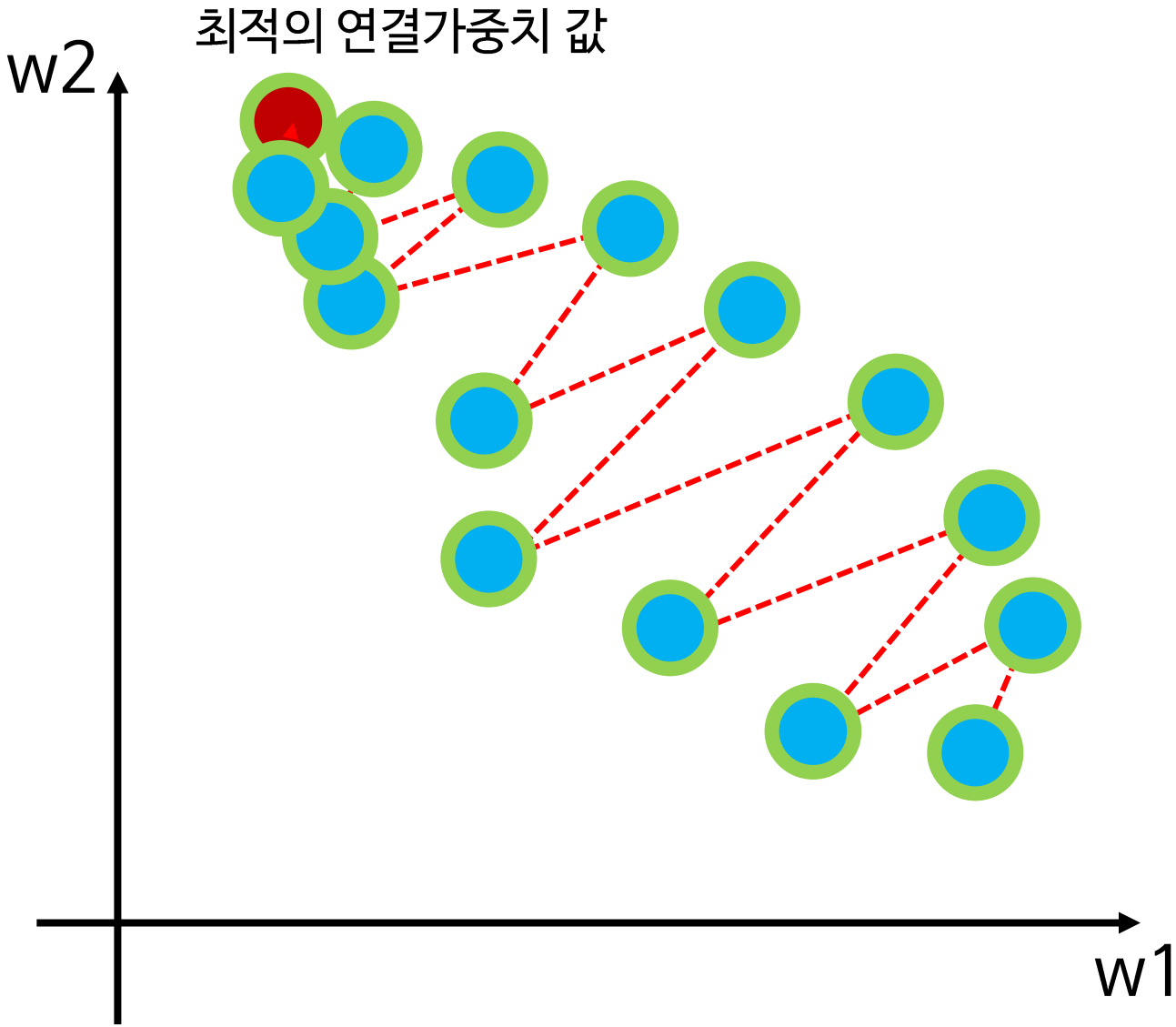
즉 시그모이드 학습은 때로는 이런 비효율적인 특성이 있습니다



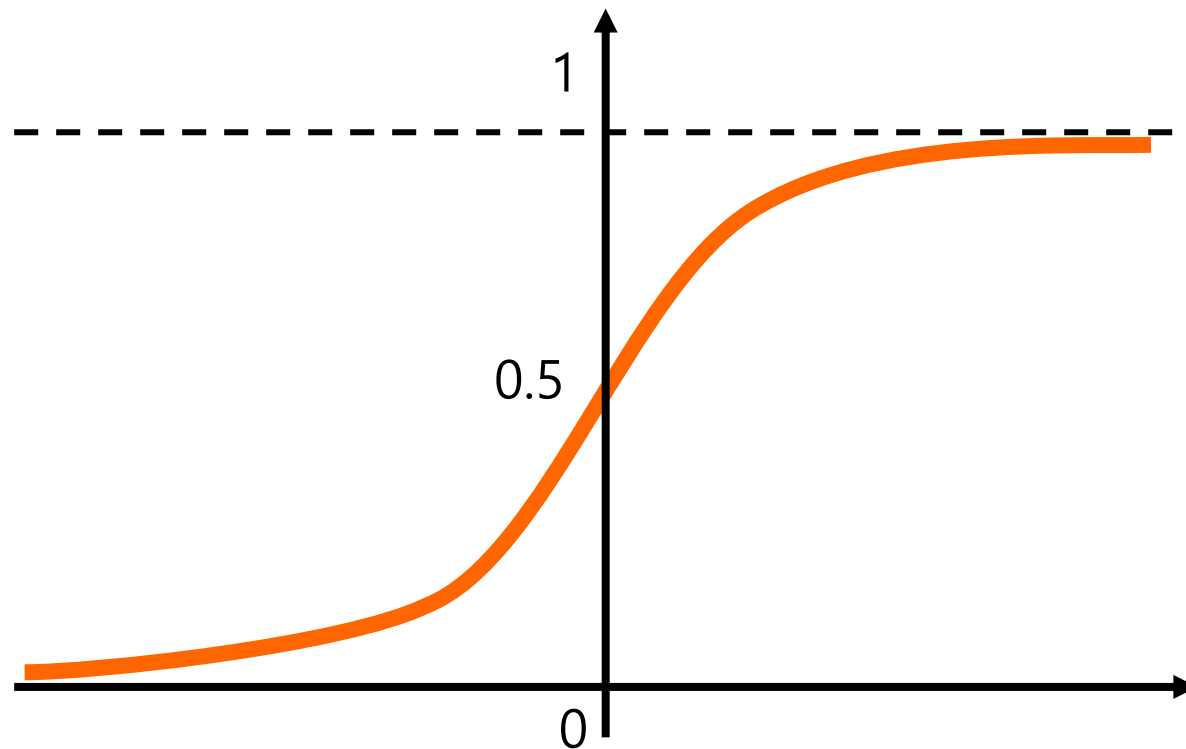
Chapter 2

시그모이드의 약점을 넘어서..

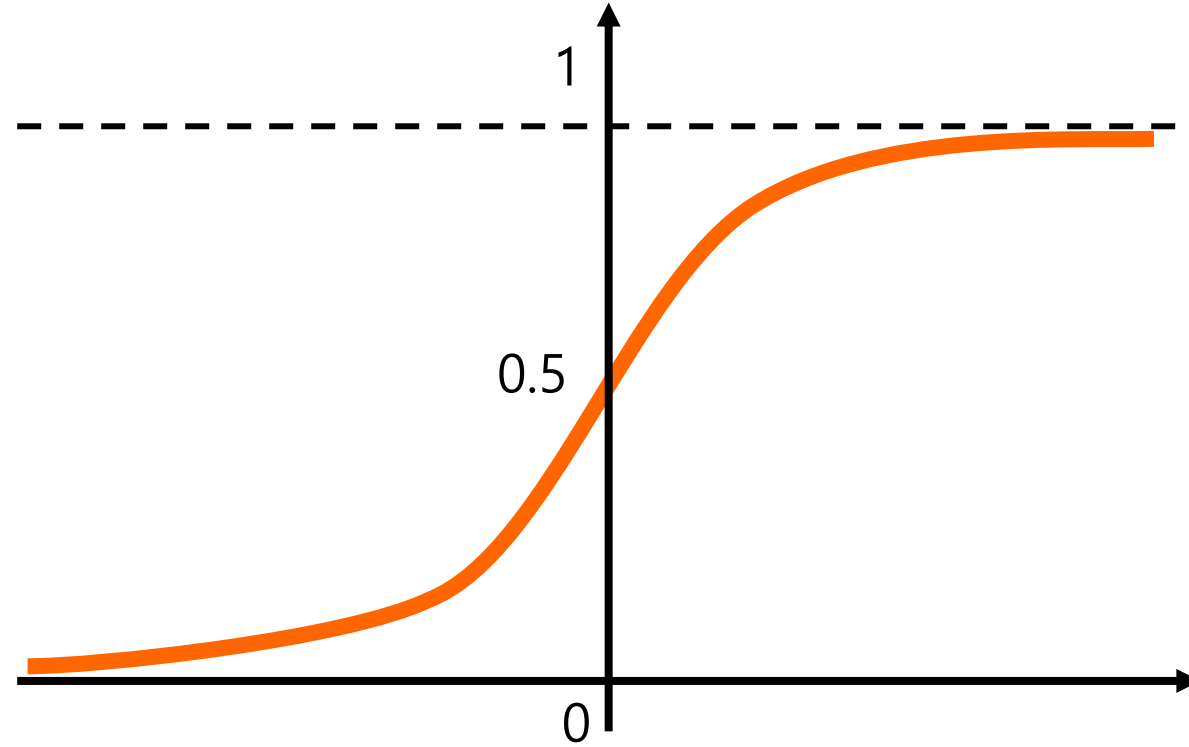
앞서 보았던 시그모이드 함수의 지그재그 현상은



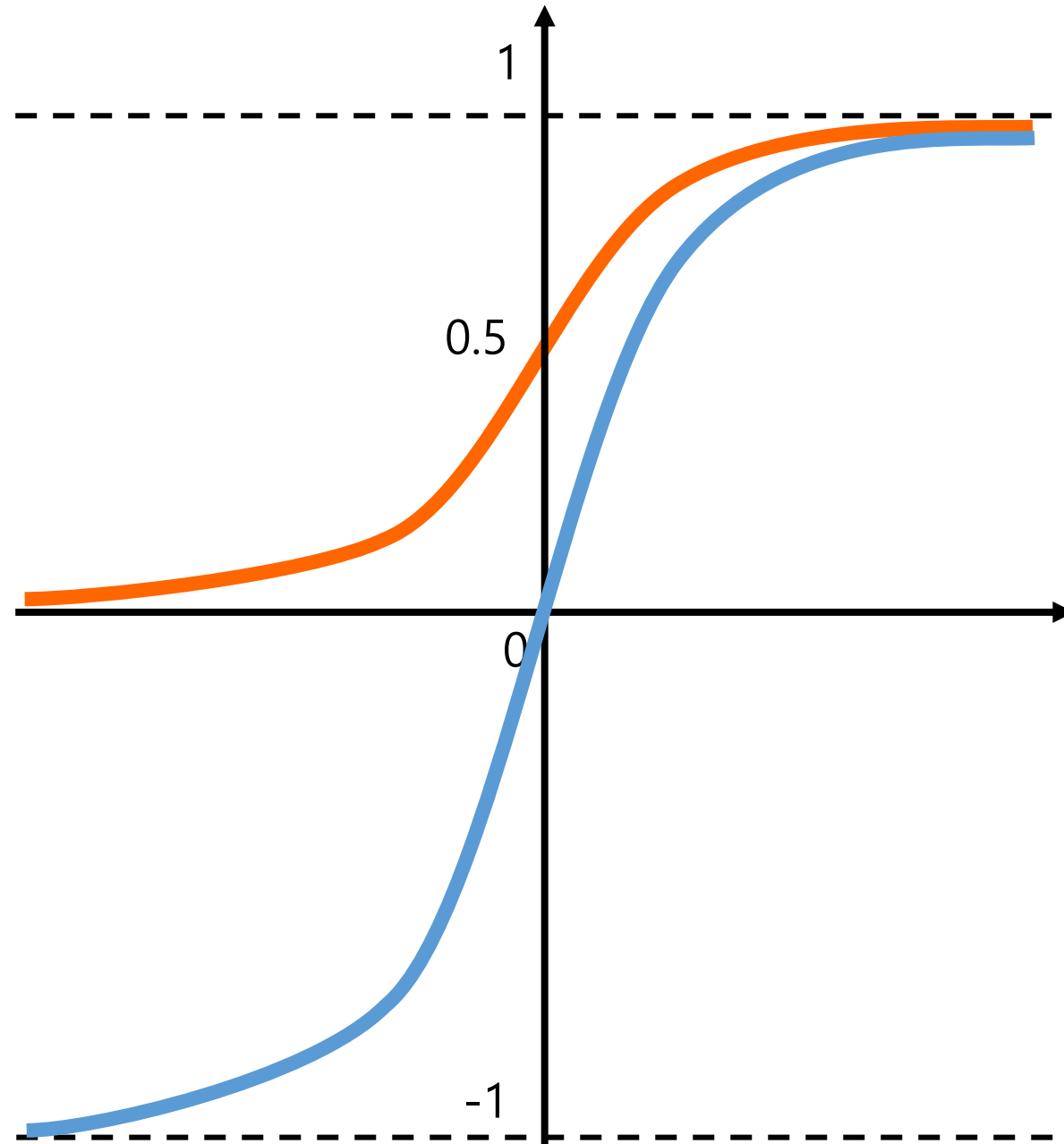
그 원인이 바로 활성화 함수의 출력값이 항상 양수였기 때문이었습니다



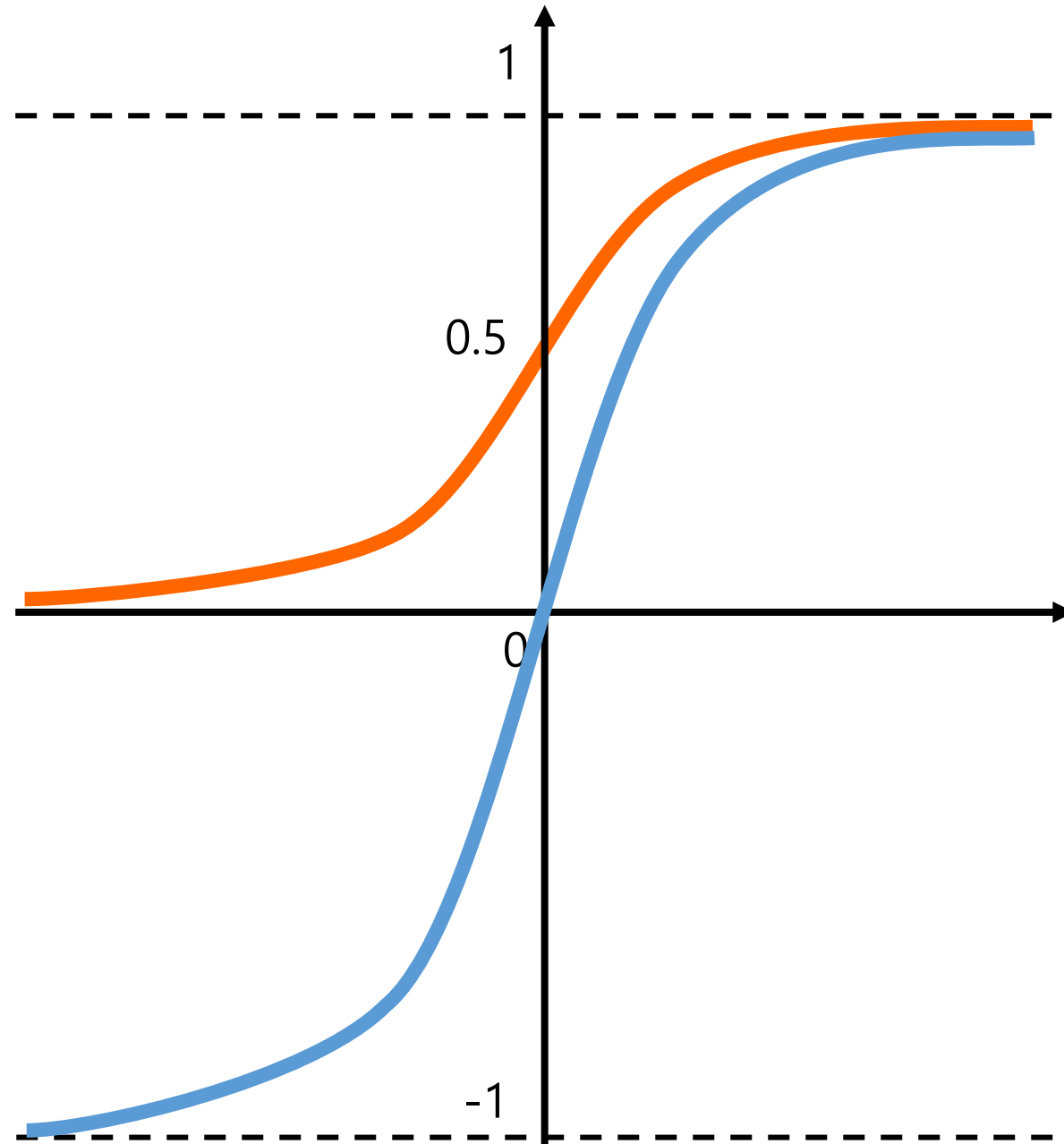
그래서 시그모이드 함수와 유사하게 생겼지만 출력값이 $-1 \sim 1$ 사이인



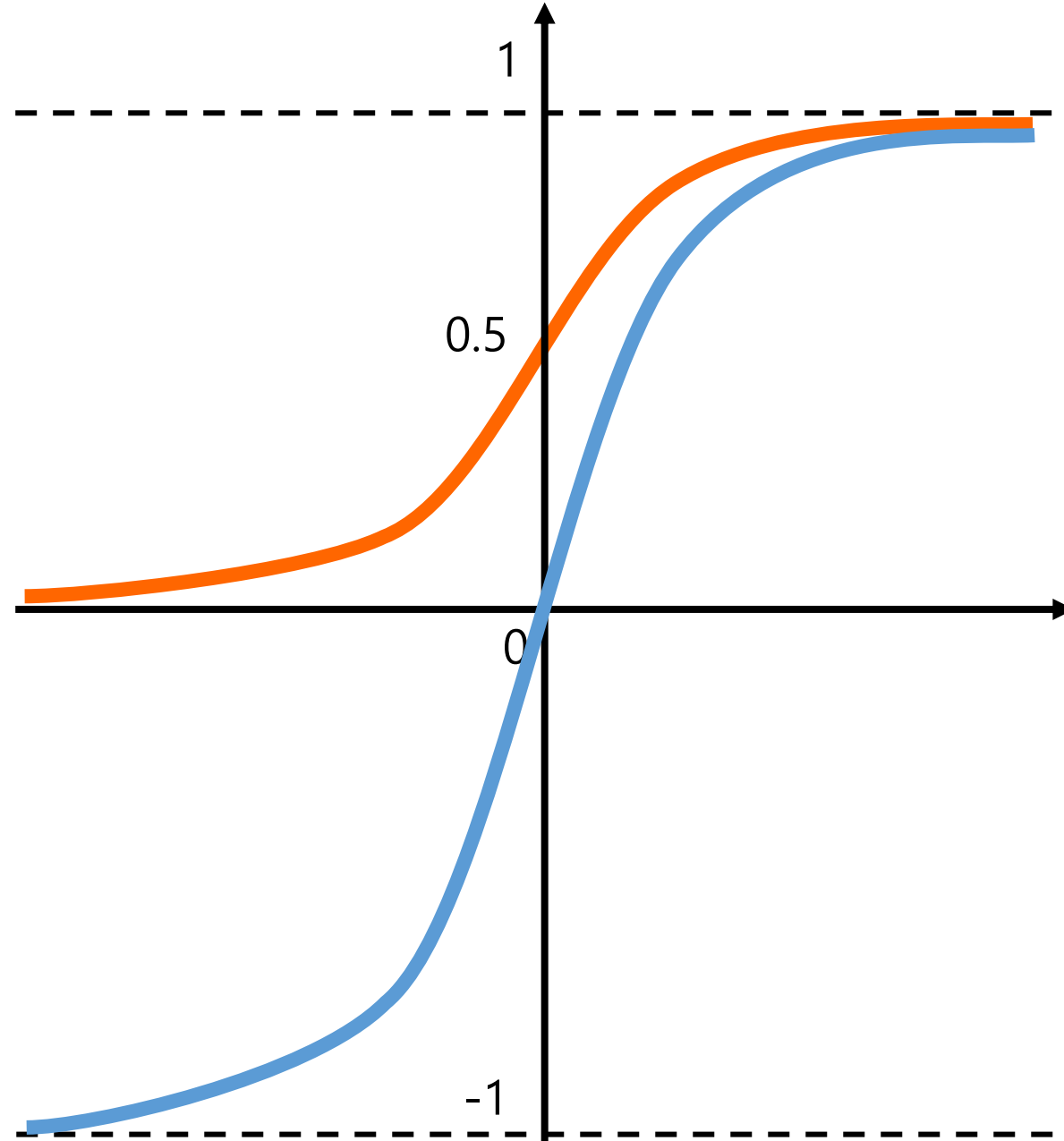
tanh 함수가 시그모이드 함수의 좋은 대안이 될 수 있습니다



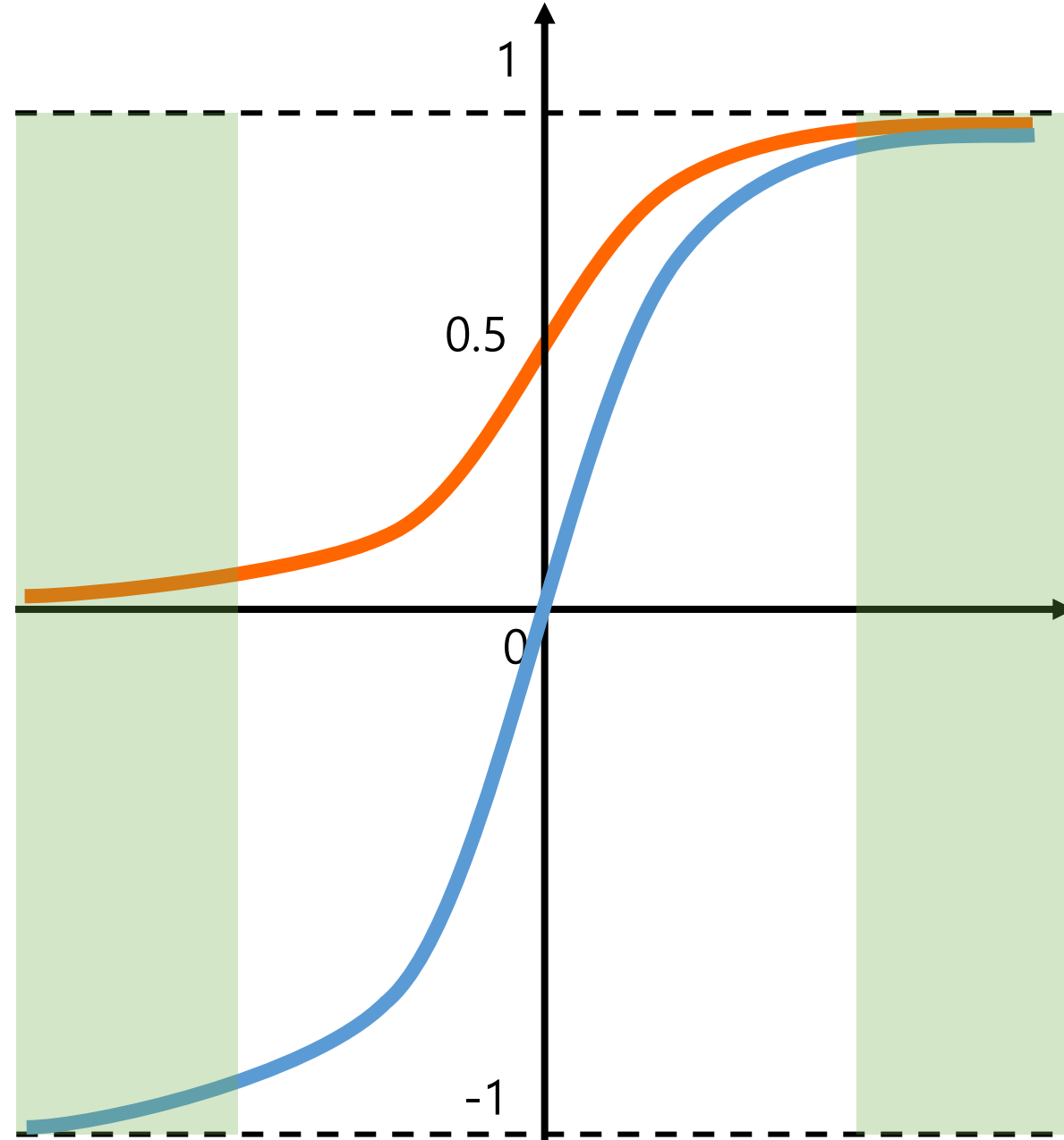
tanh함수는 시그모이드 함수의 장점을 그대로 가져오면서



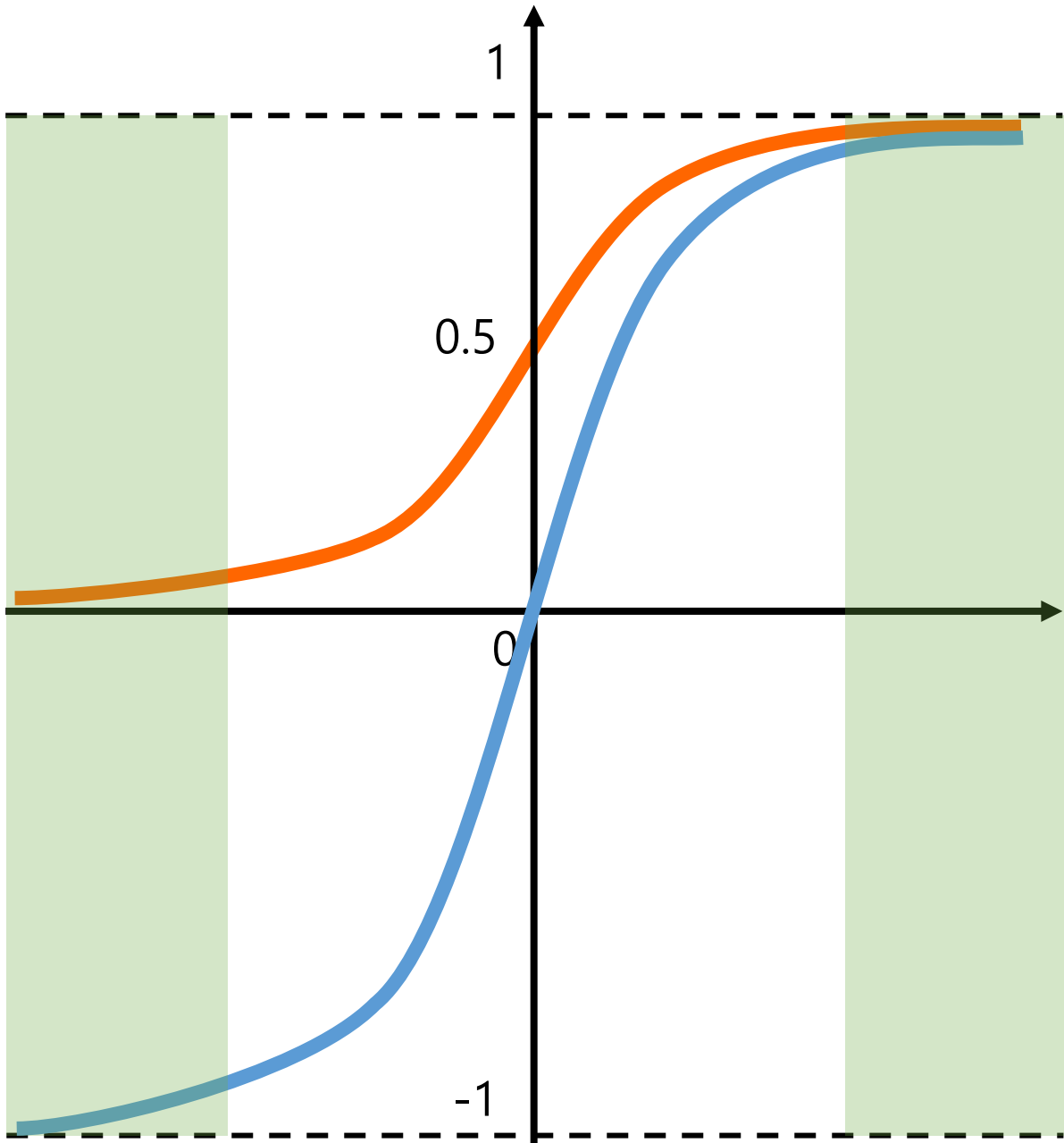
시그모이드 함수의 단점인 non-zero 문제를 해결할 수 있기 때문입니다



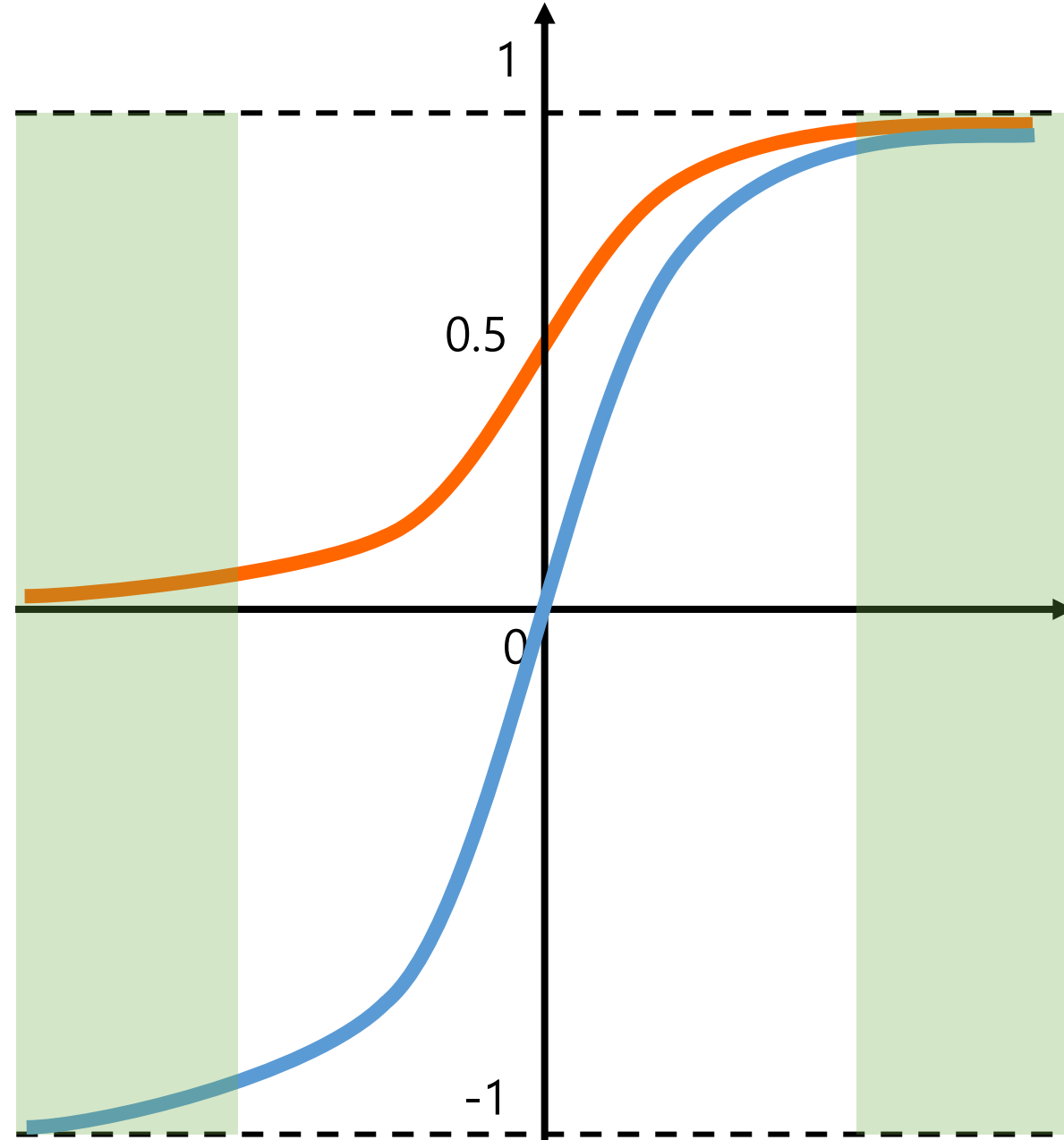
하지만 이 tanh함수도 saturation and killing gradient이슈를



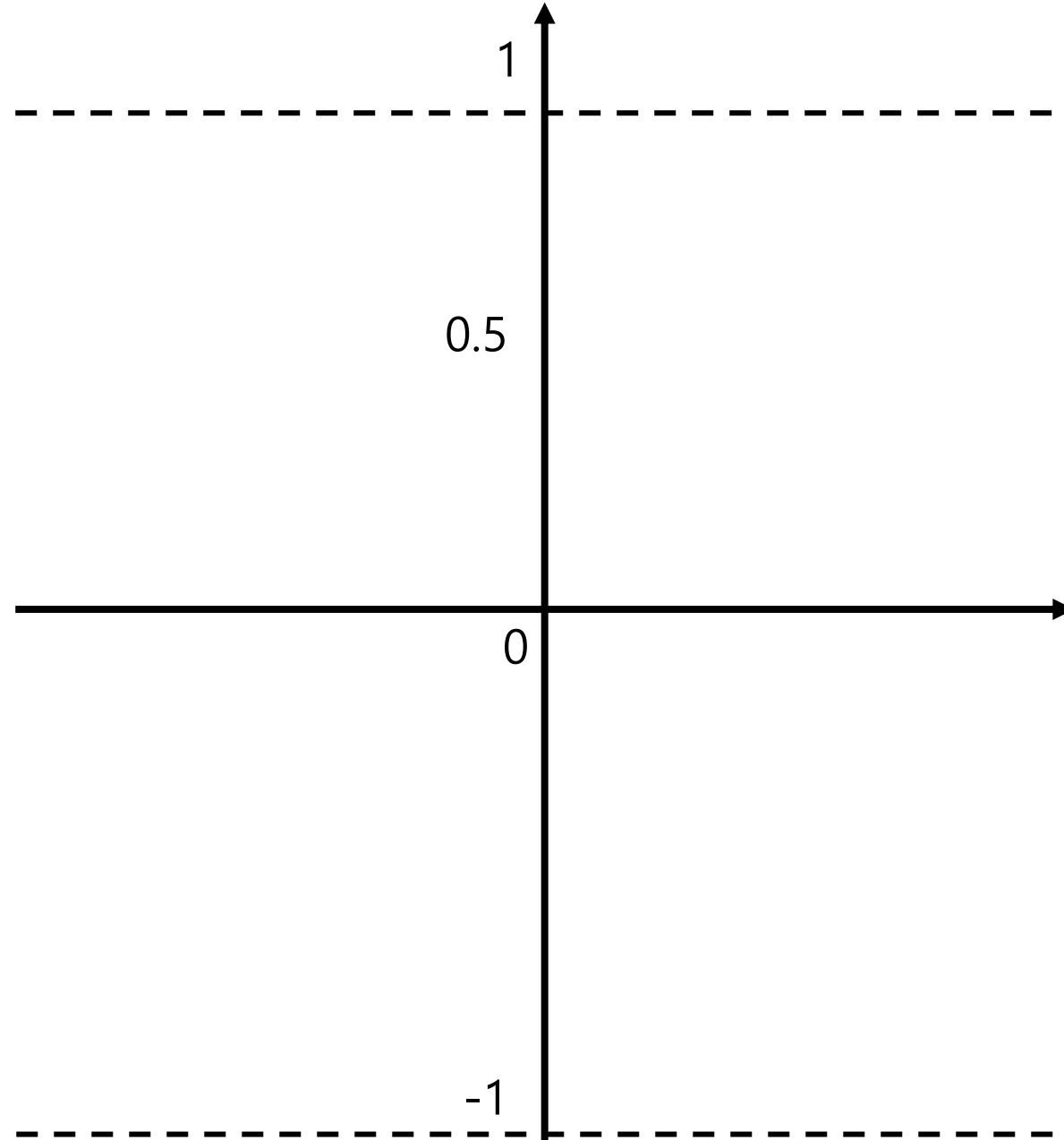
해결할 수는 없었습니다..



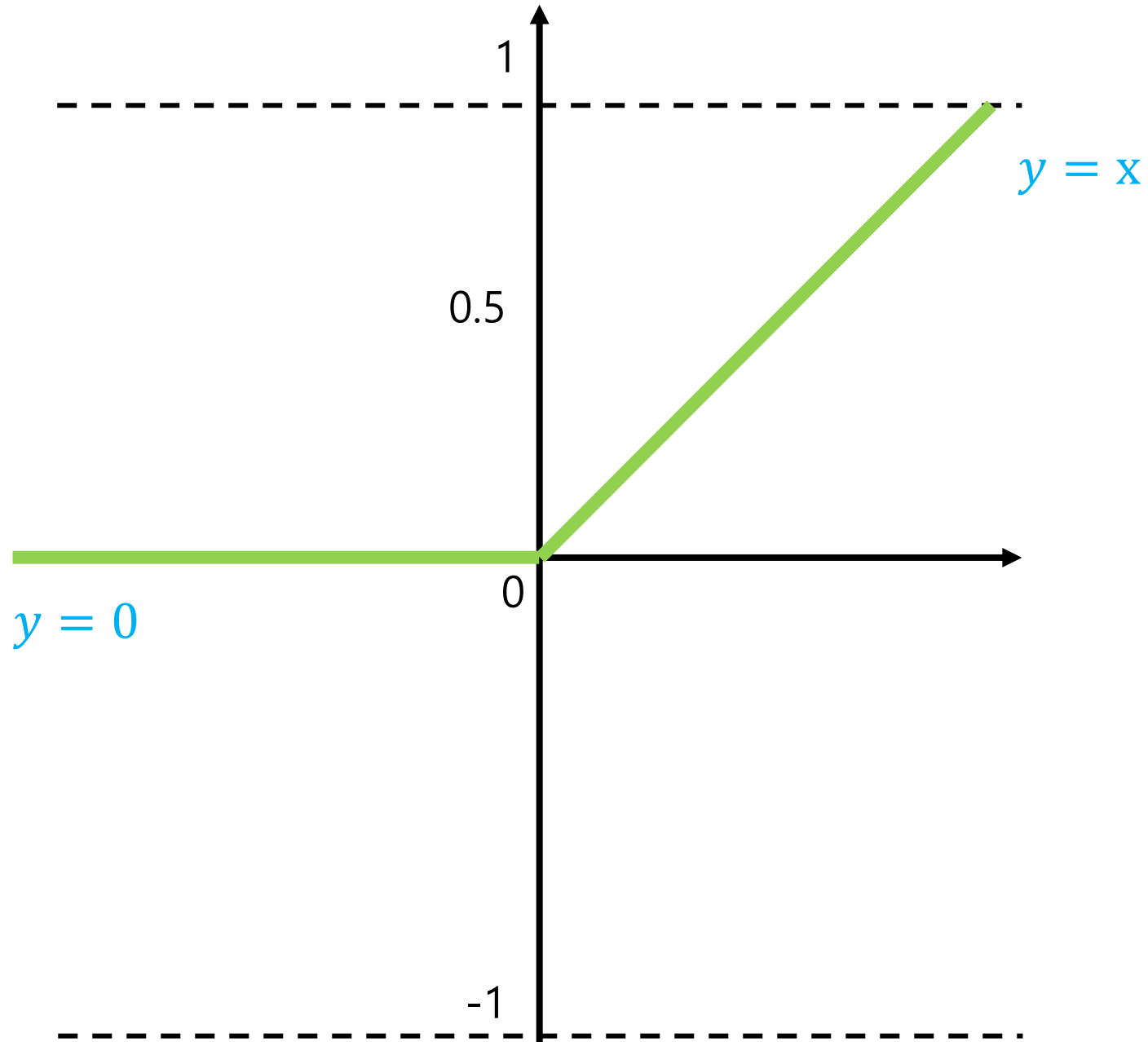
그래서 이 gradient killing and saturation 문제를 해결하기 위해서



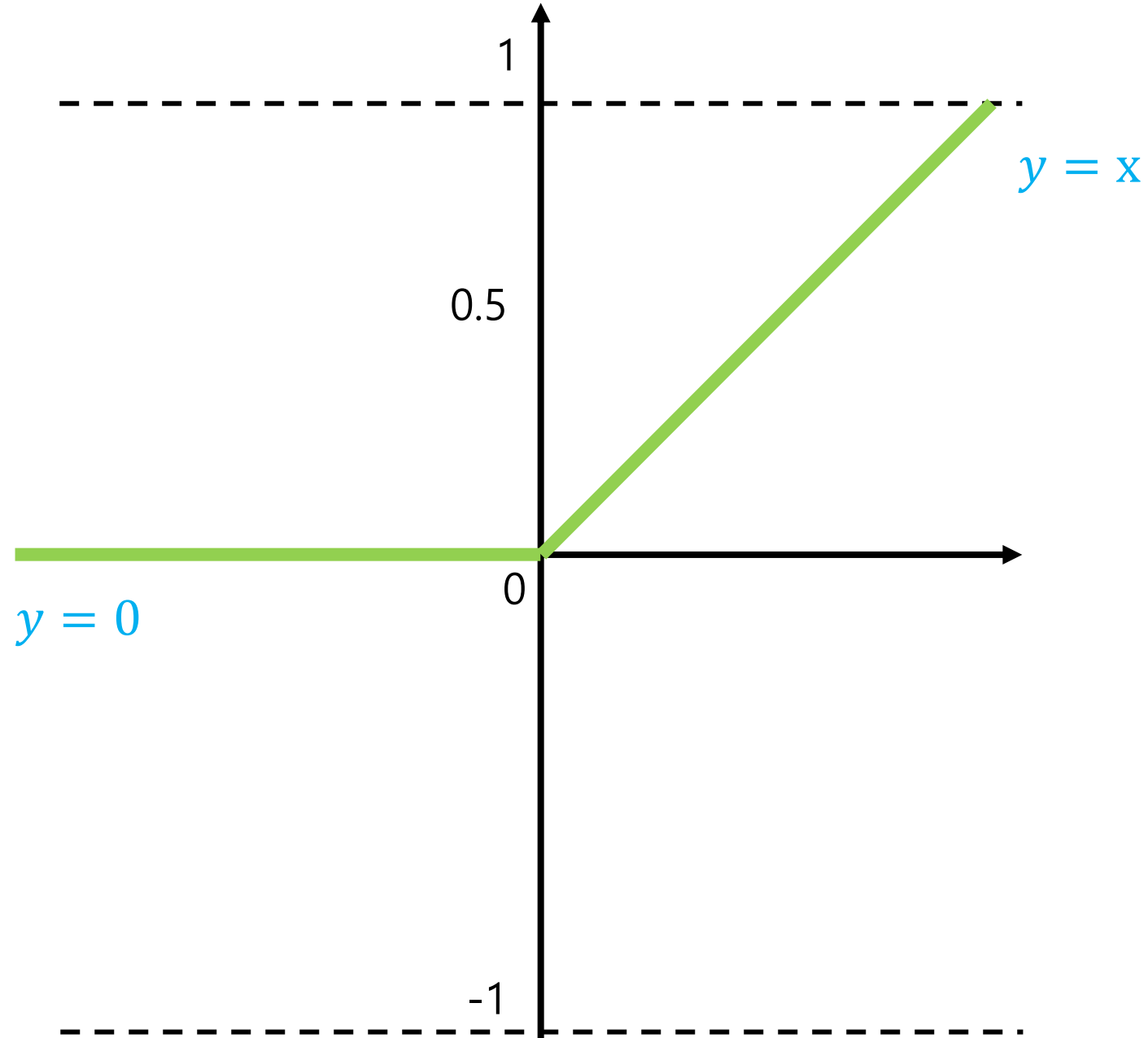
새로운 형태의 활성화 함수가 등장하였는데요,



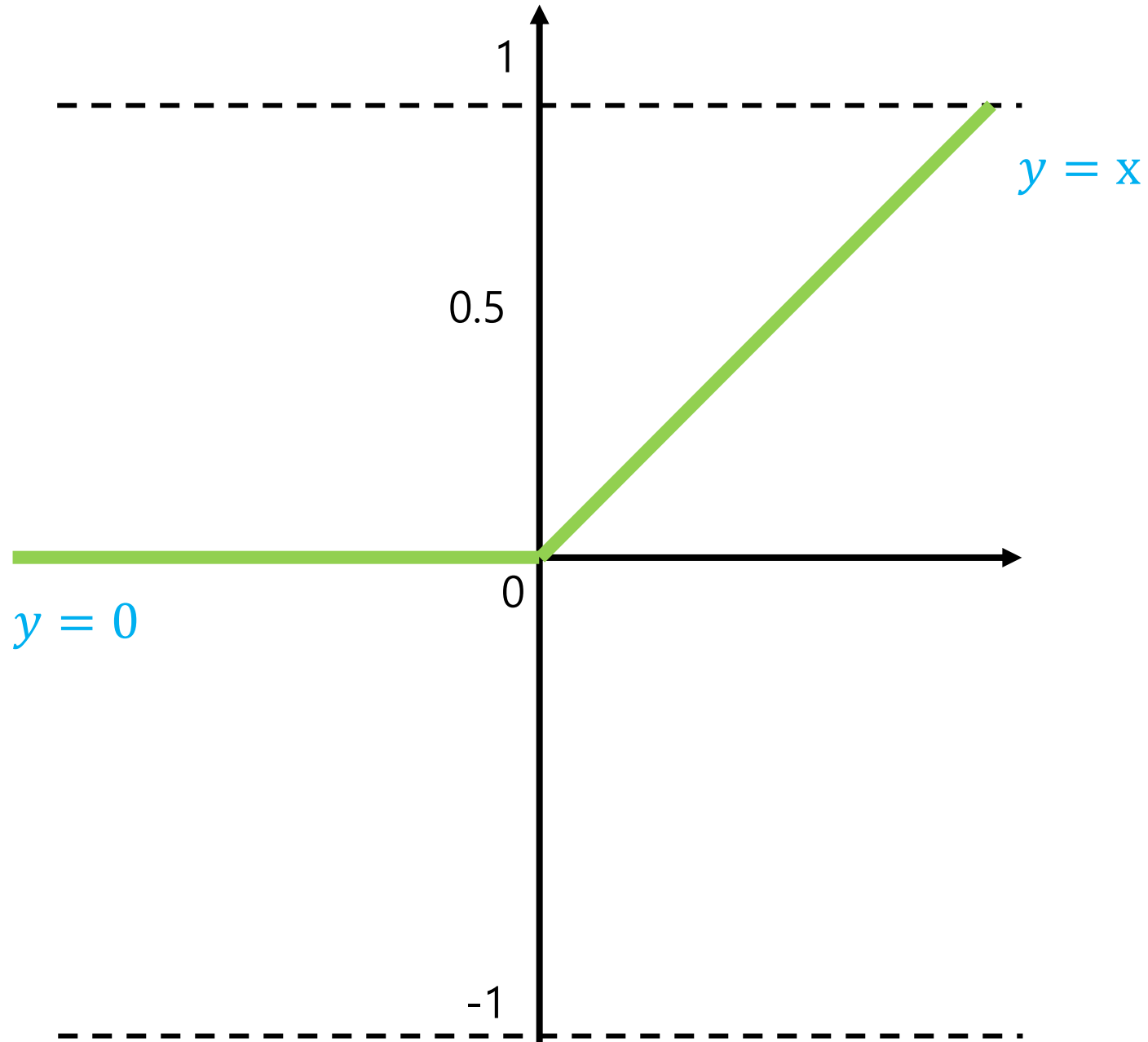
바로 그 유명한 ReLU (렐루) 함수 입니다



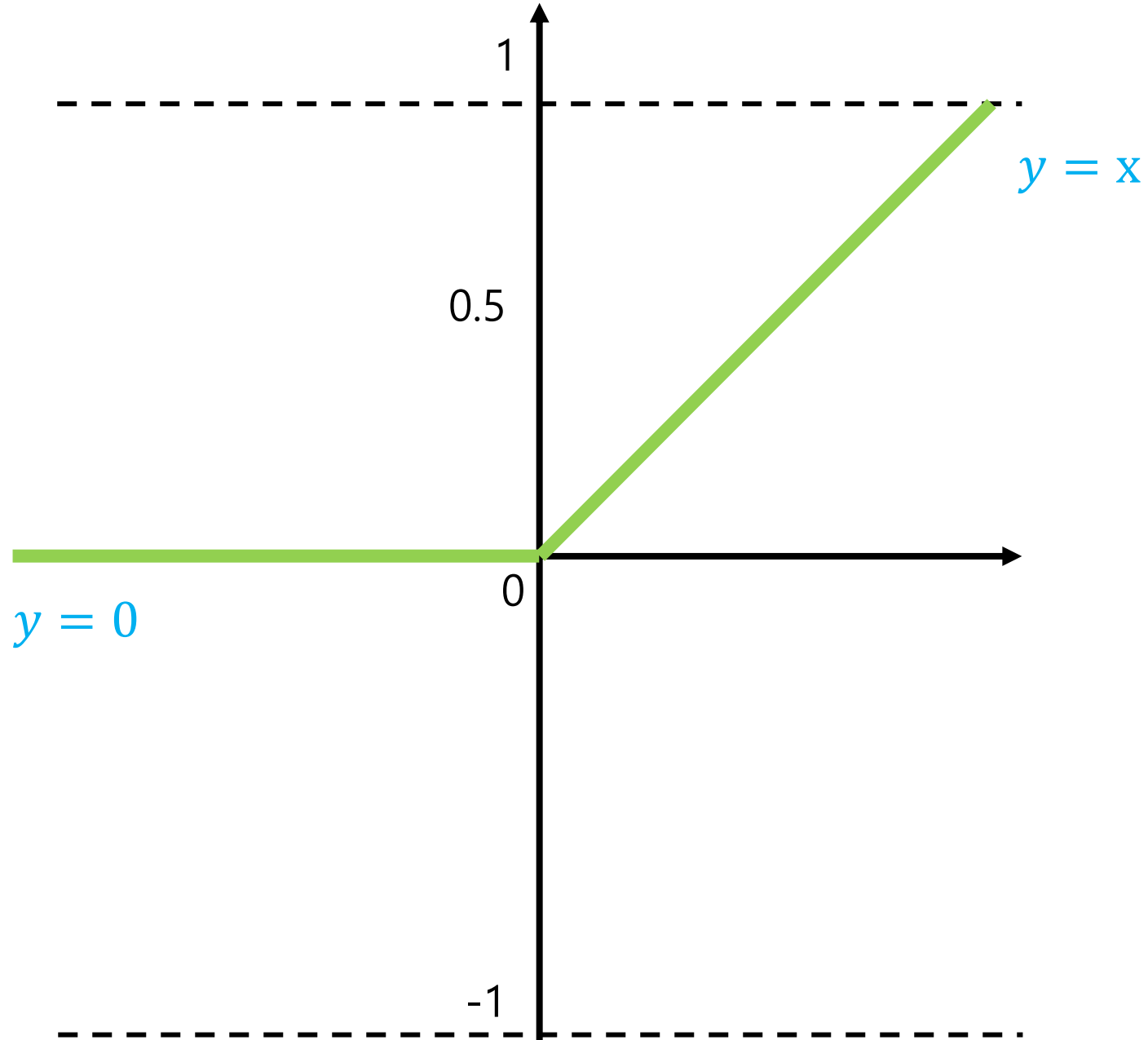
ReLU (Rectified linear unit) 함수는 비교적 최근에 등장한 함수로써



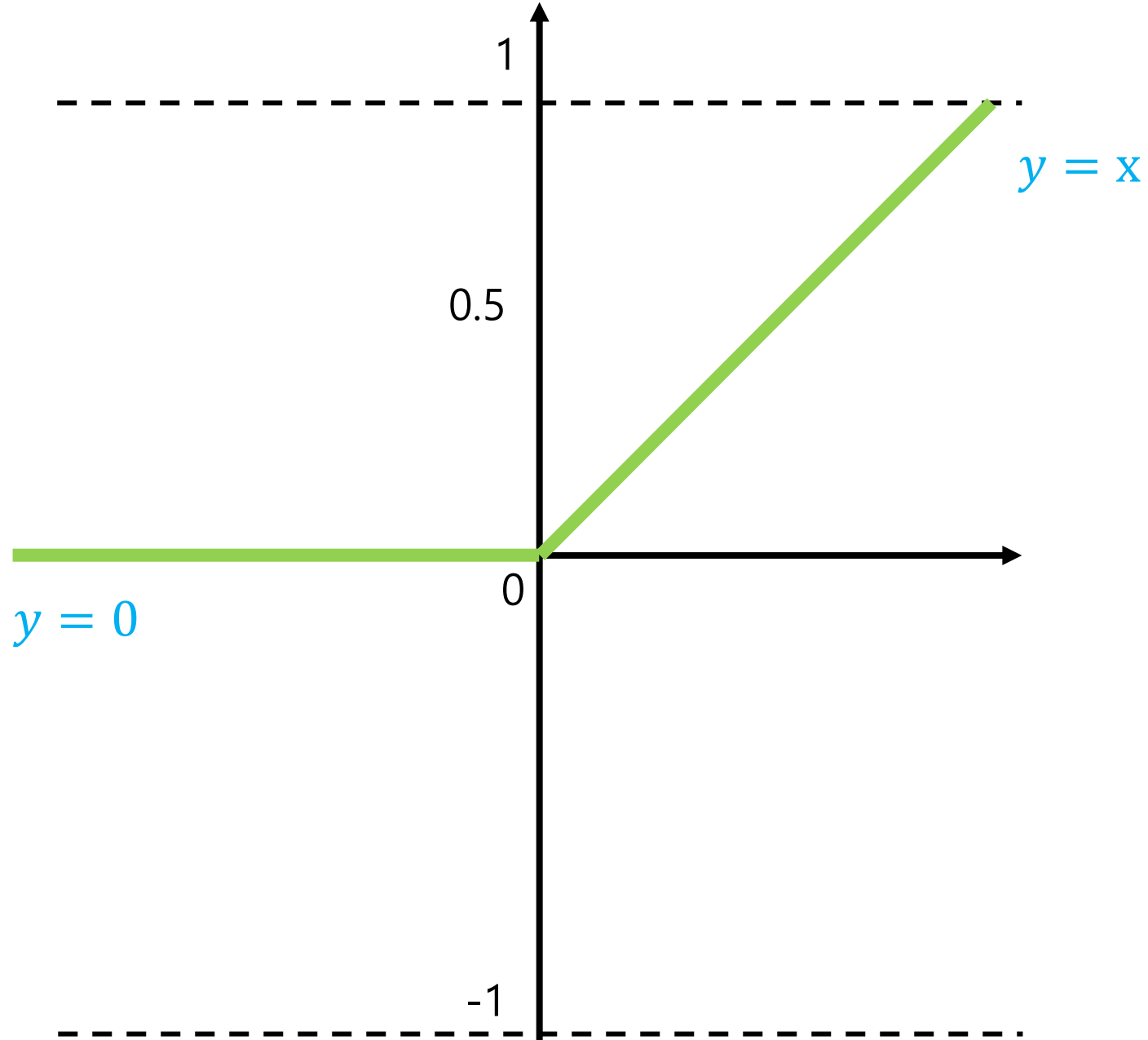
시그모이드나 tanh함수의 고질적인 문제였던



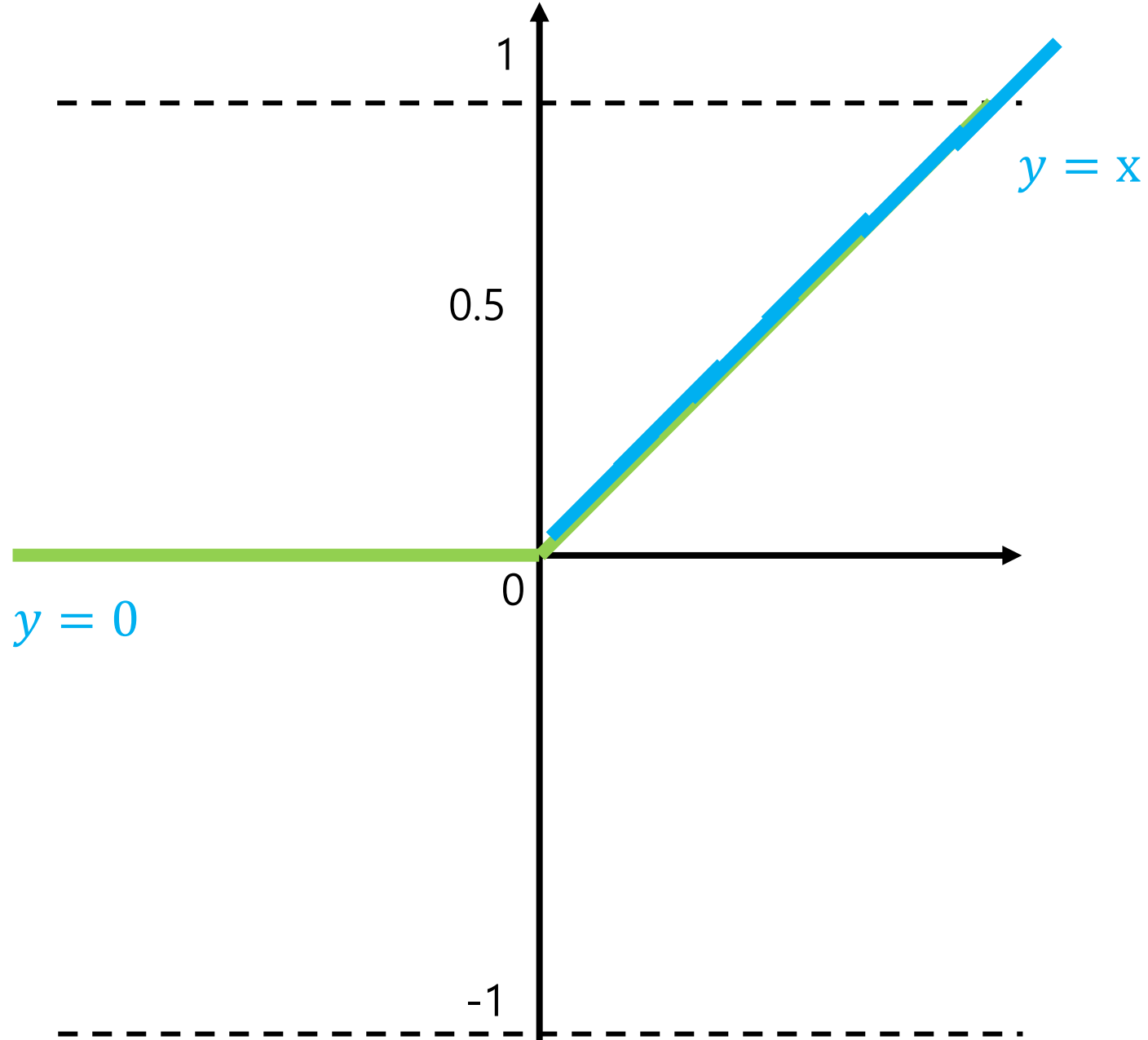
Vanishing gradient problem (기울기소실문제)를 해결함으로서



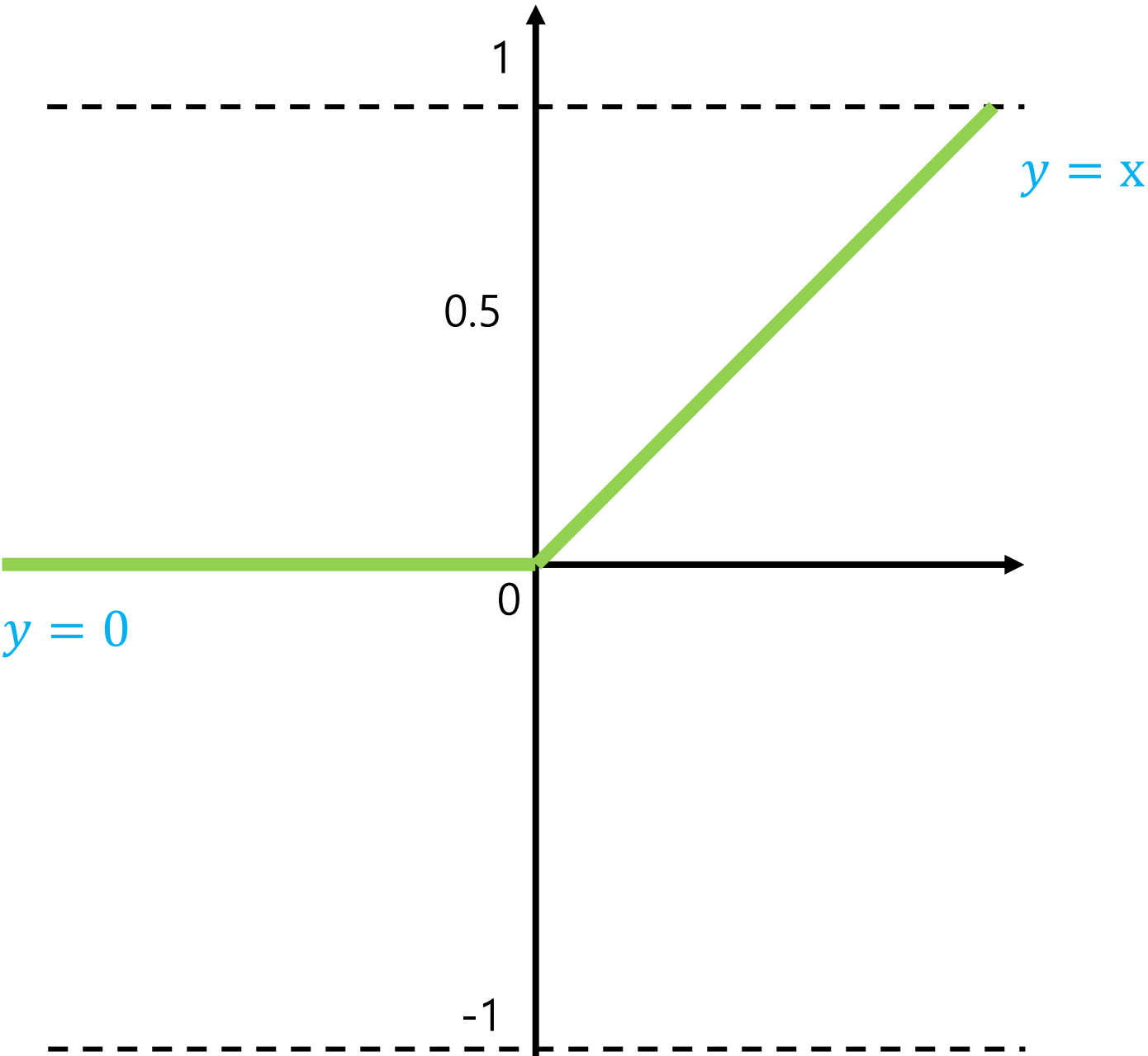
오늘날 deep learning의 혁신을 가져온 주요한 변화중 하나로 여겨지고 있습니다



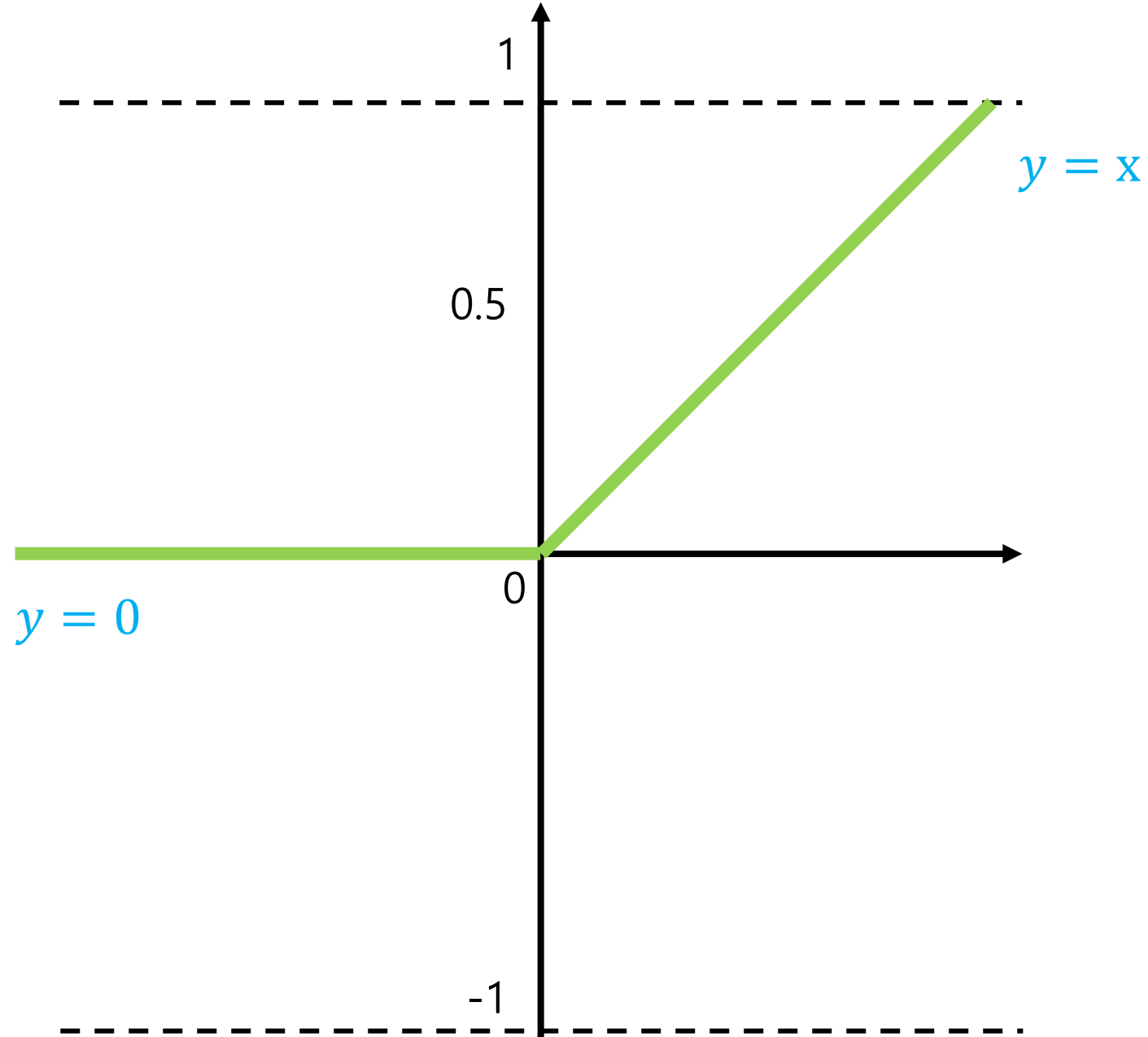
보시다시피 함수의 기울기가 늘 1로 고정되어 있어서 (양수에서)



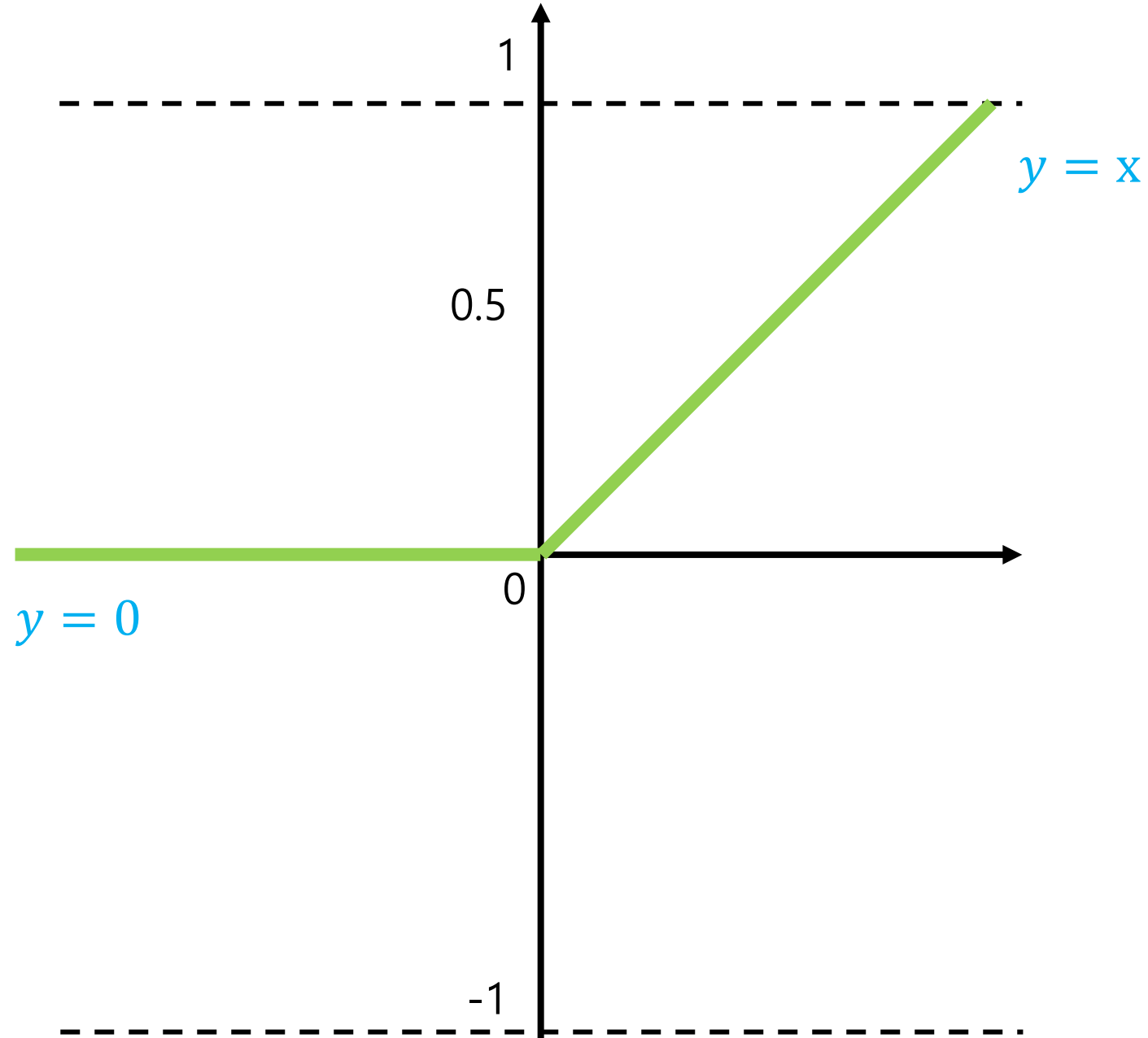
기울기 소실 문제가 발생할 여지가 없습니다



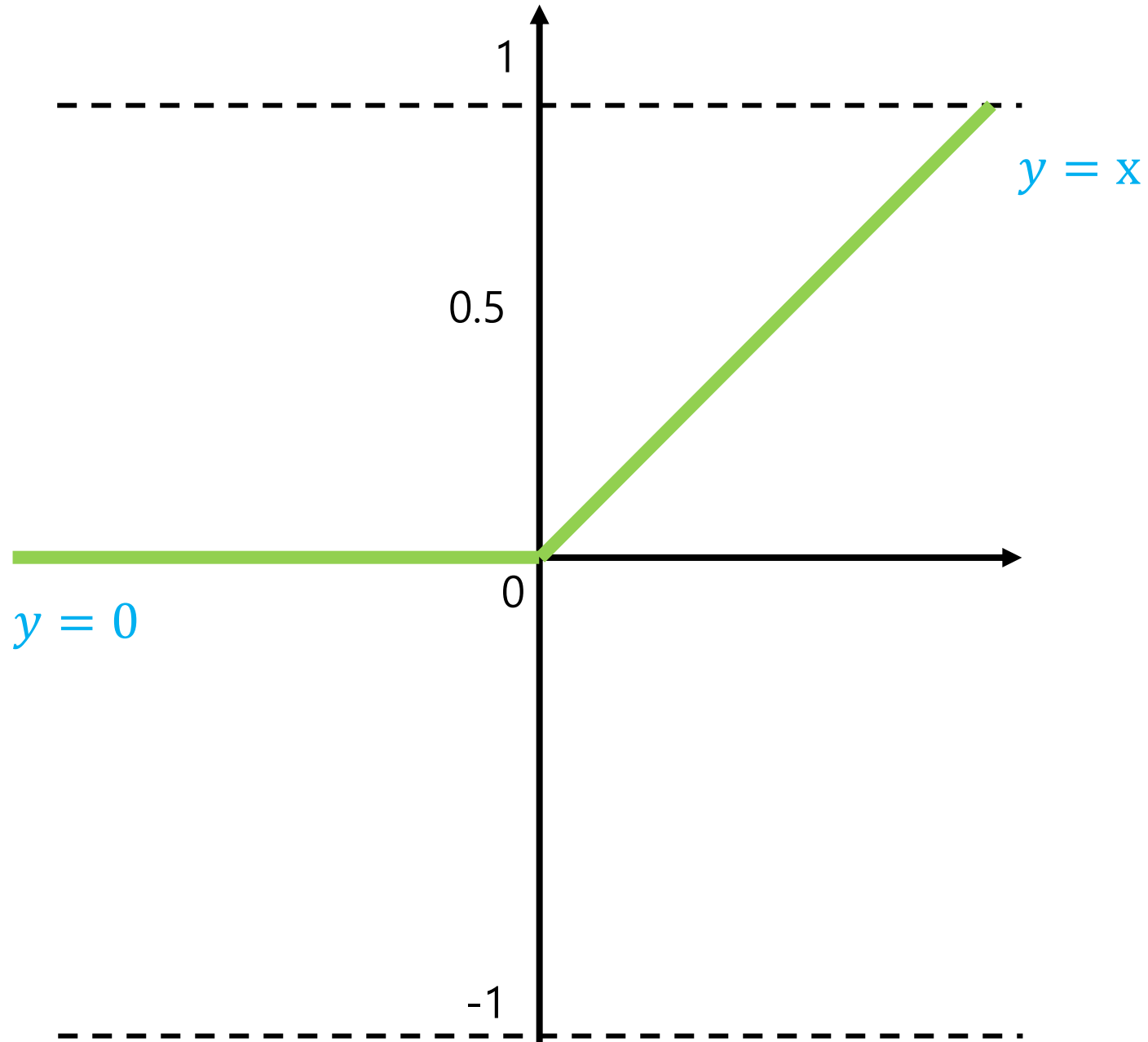
그래서 신경망의 깊은 층까지 오류역전파를 통한 학습이 가능합니다



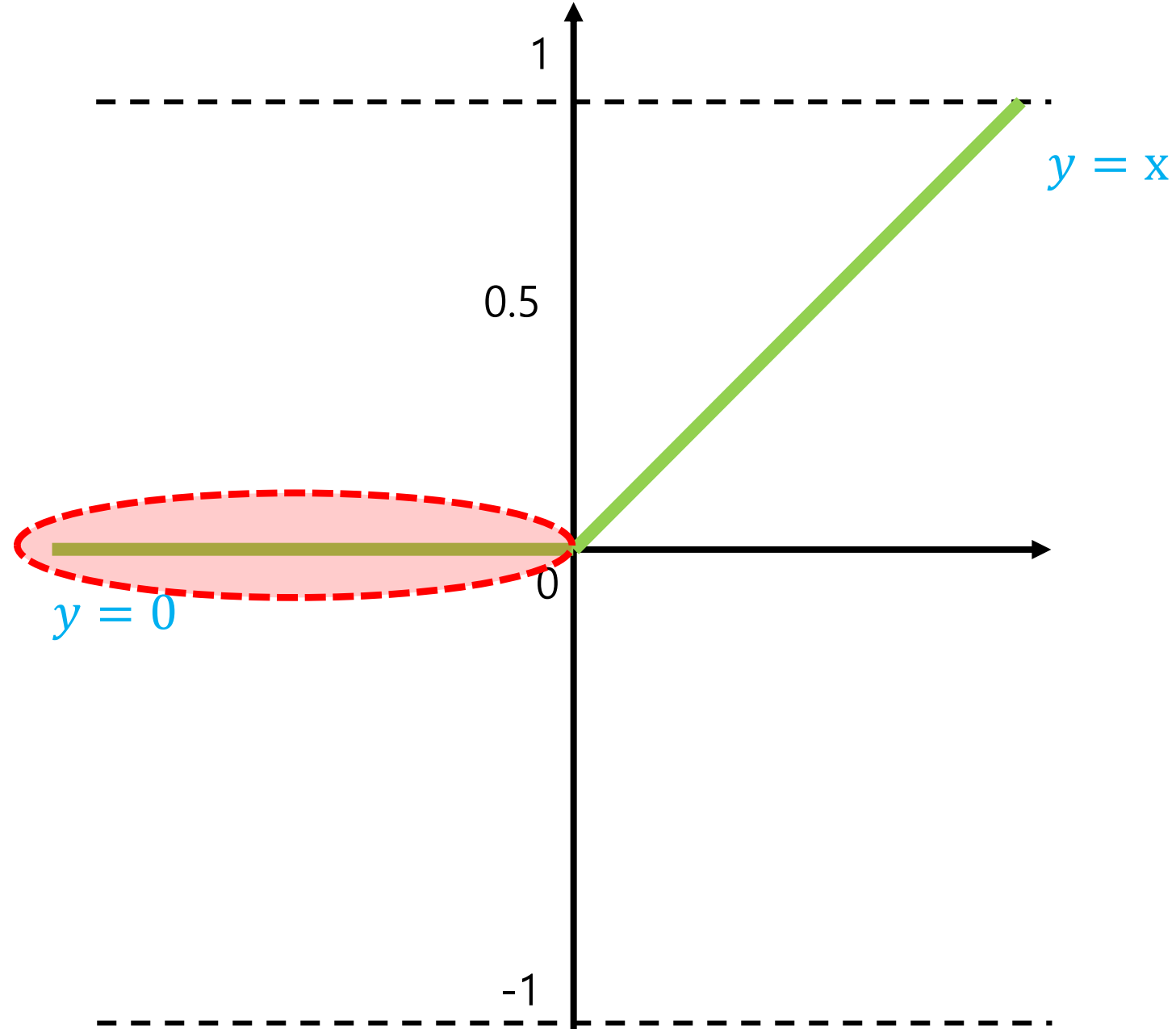
그래서 신경망의 깊은 층까지 오류역전파를 통한 학습이 가능합니다



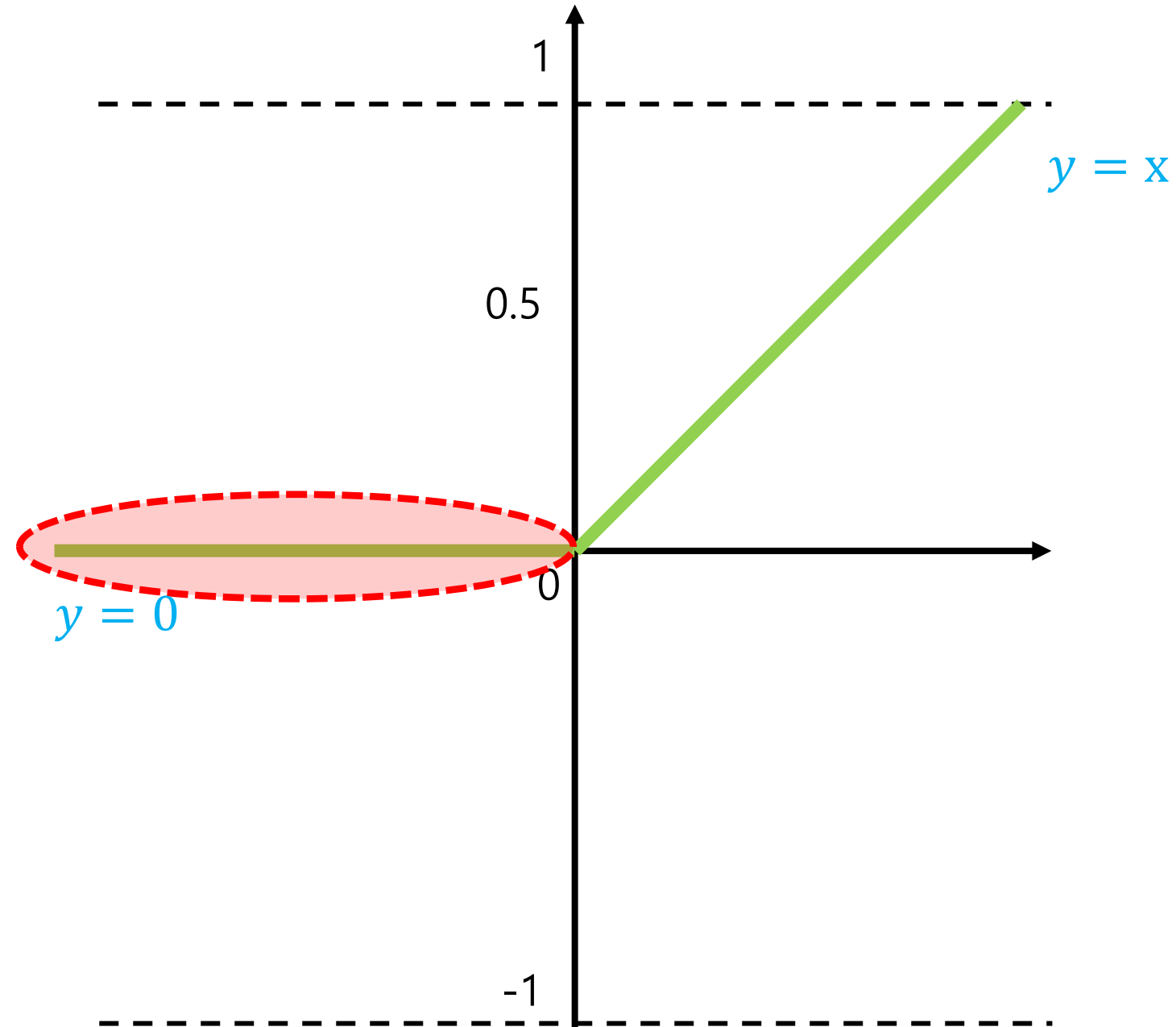
하지만 ReLU함수도 단점이 있는데요,



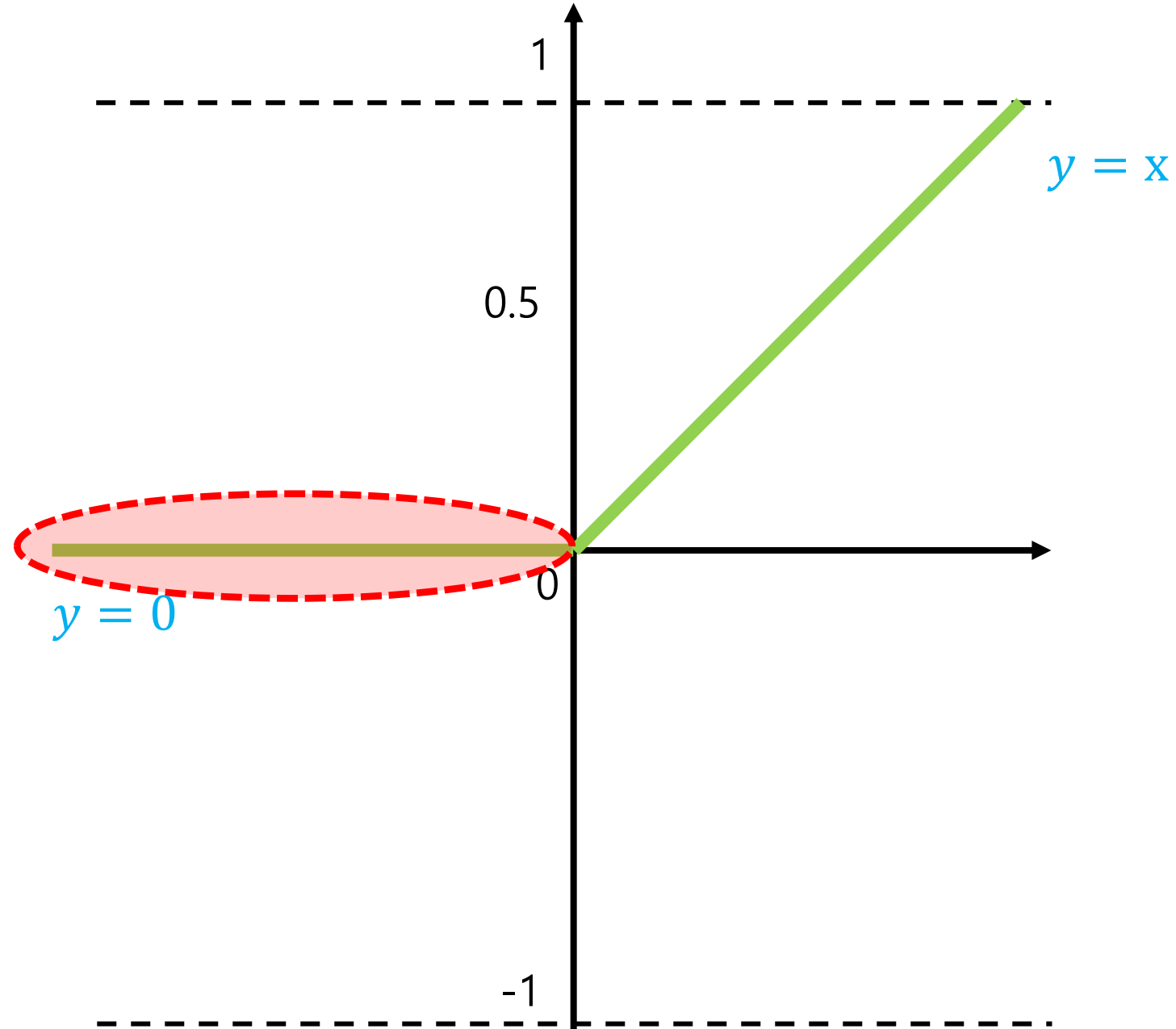
여러분도 잘 아시겠지만, 입력값이 음수일 경우 기울기가 0가 되어,



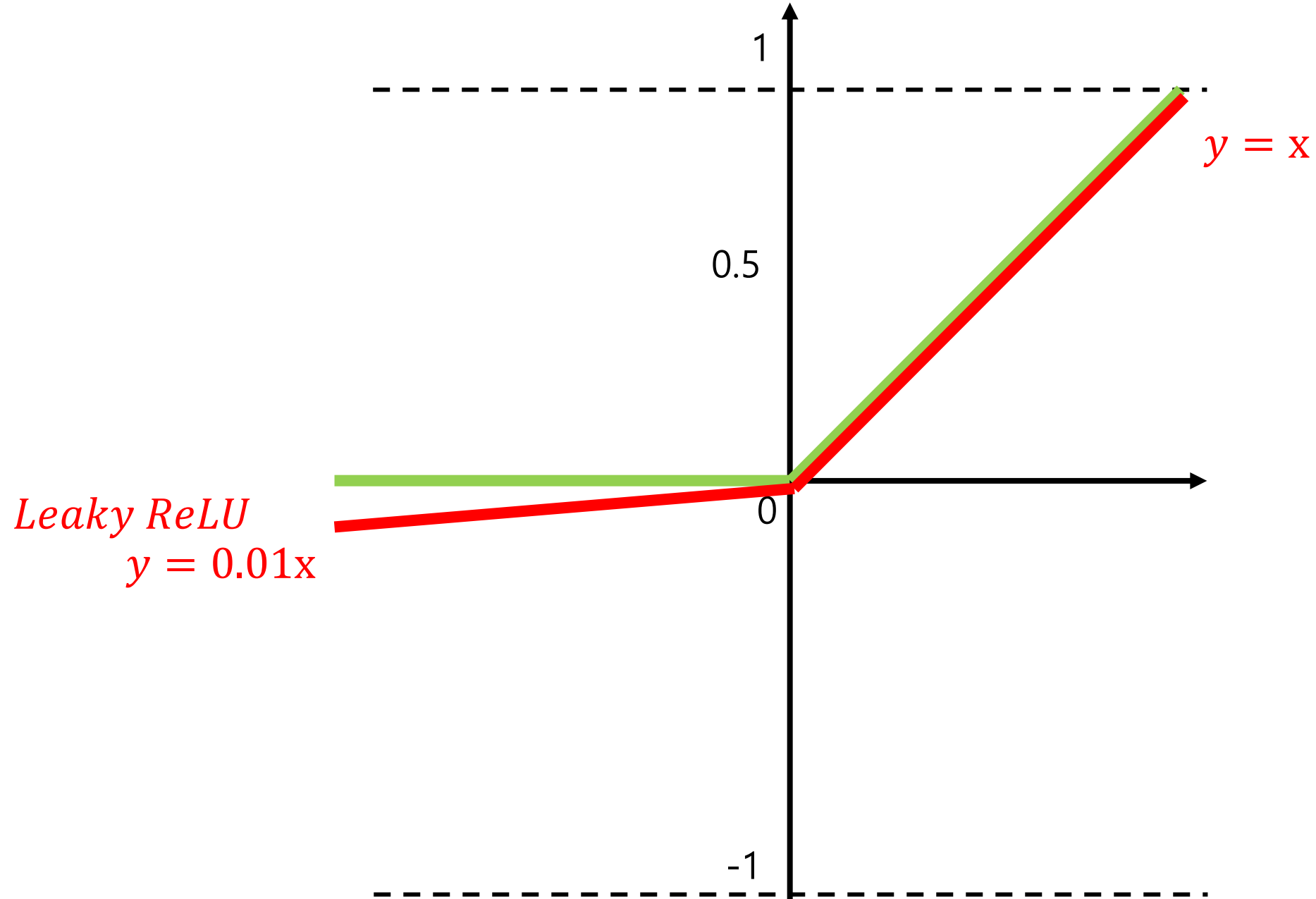
신경망 학습이 진행되지 않는



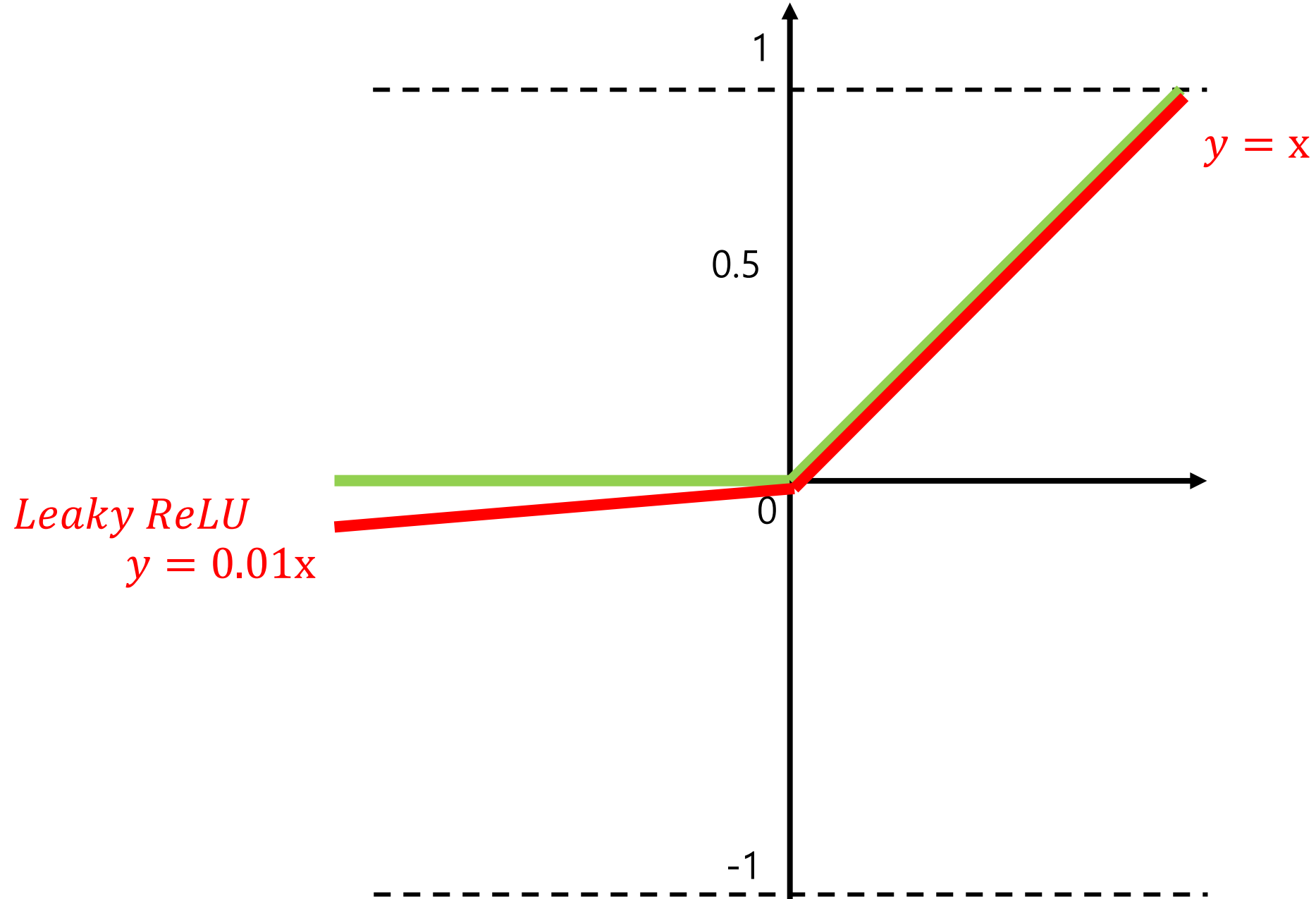
Dying ReLU현상 (죽어가는 렐루?) 현상이 발생하기도 합니다



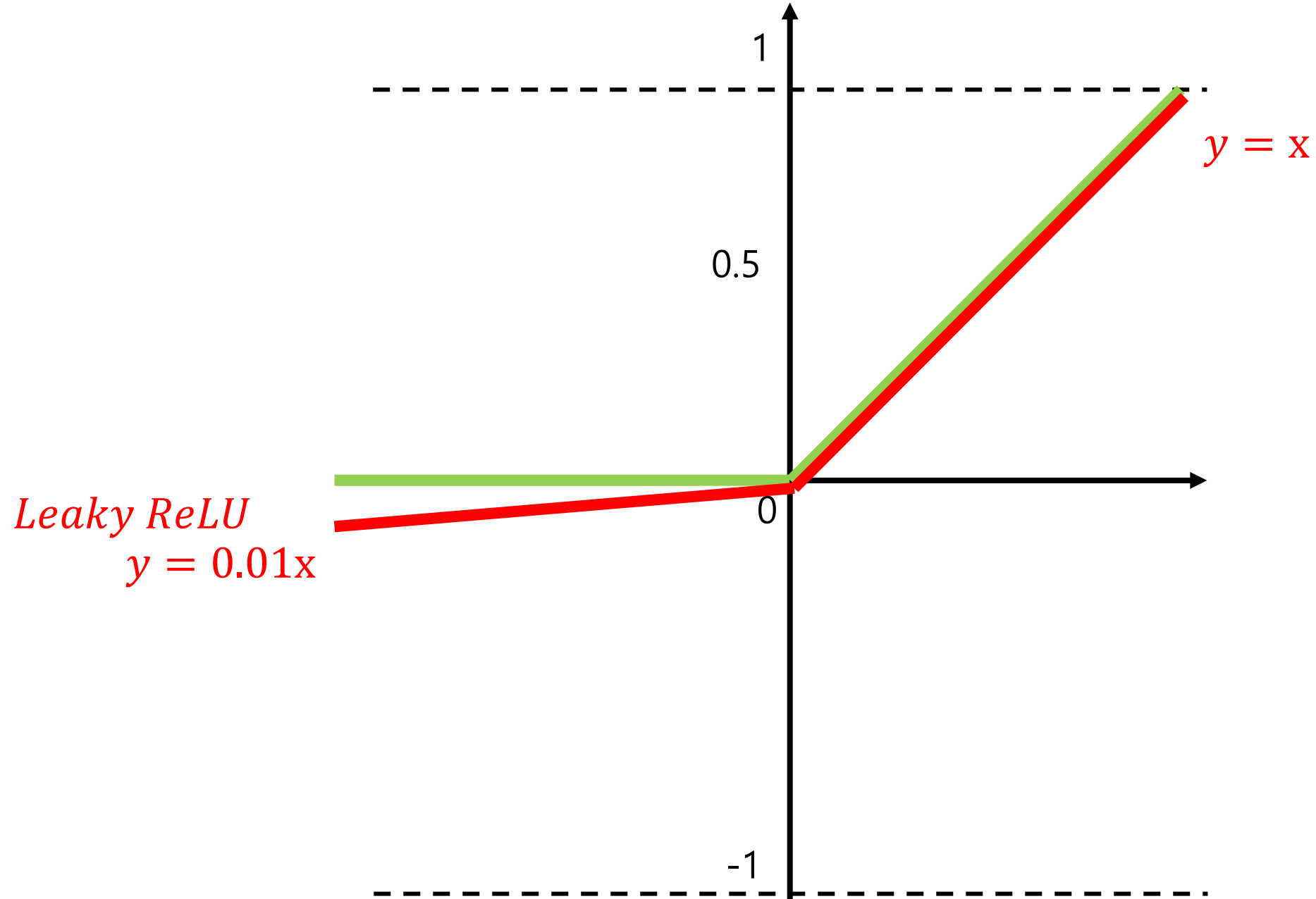
이런 현상을 없애기 위해 다음과 같은 변형 ReLU가 등장했습니다



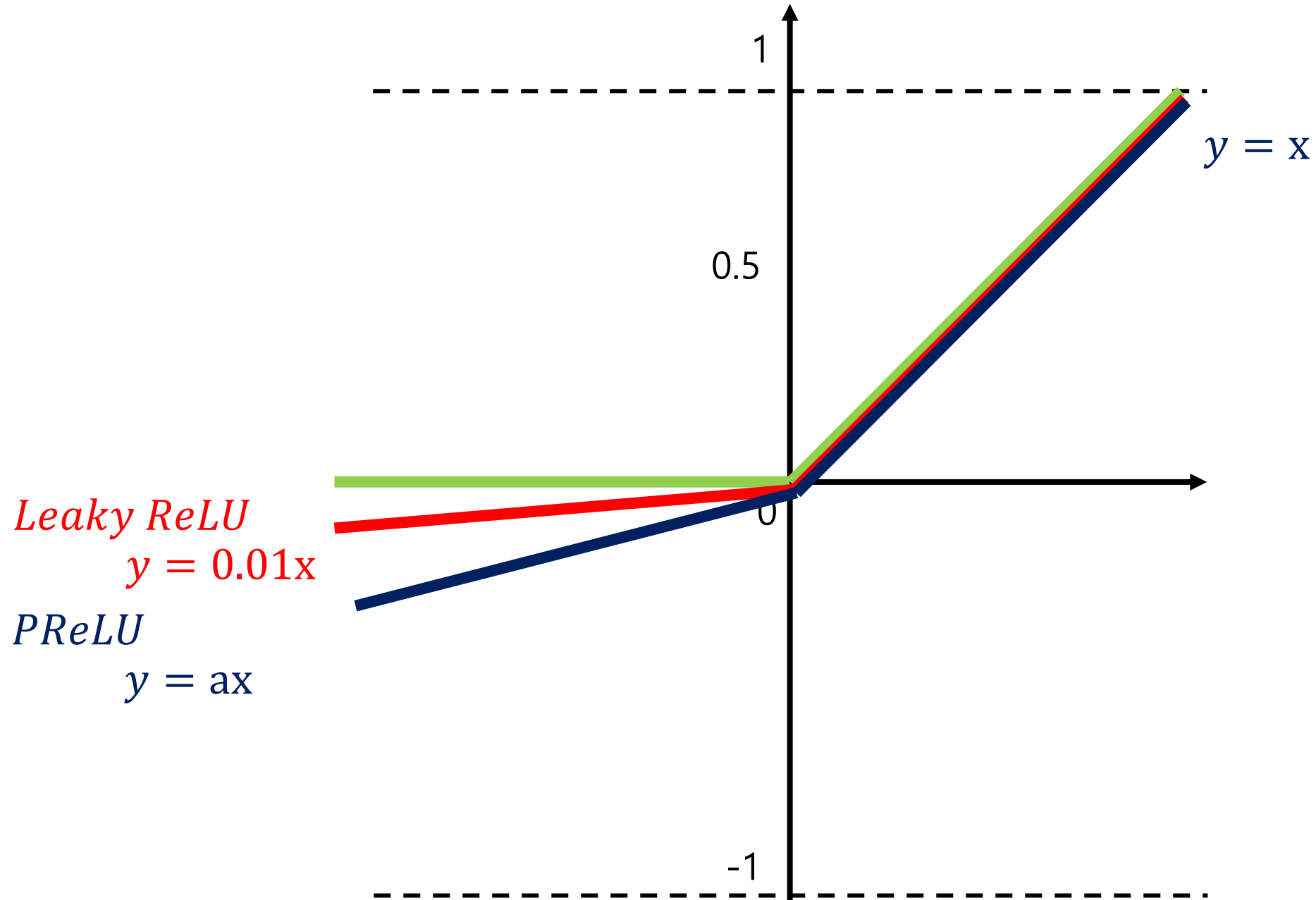
Leaky ReLU는 음수 입력값에 작은 slope를 주어서



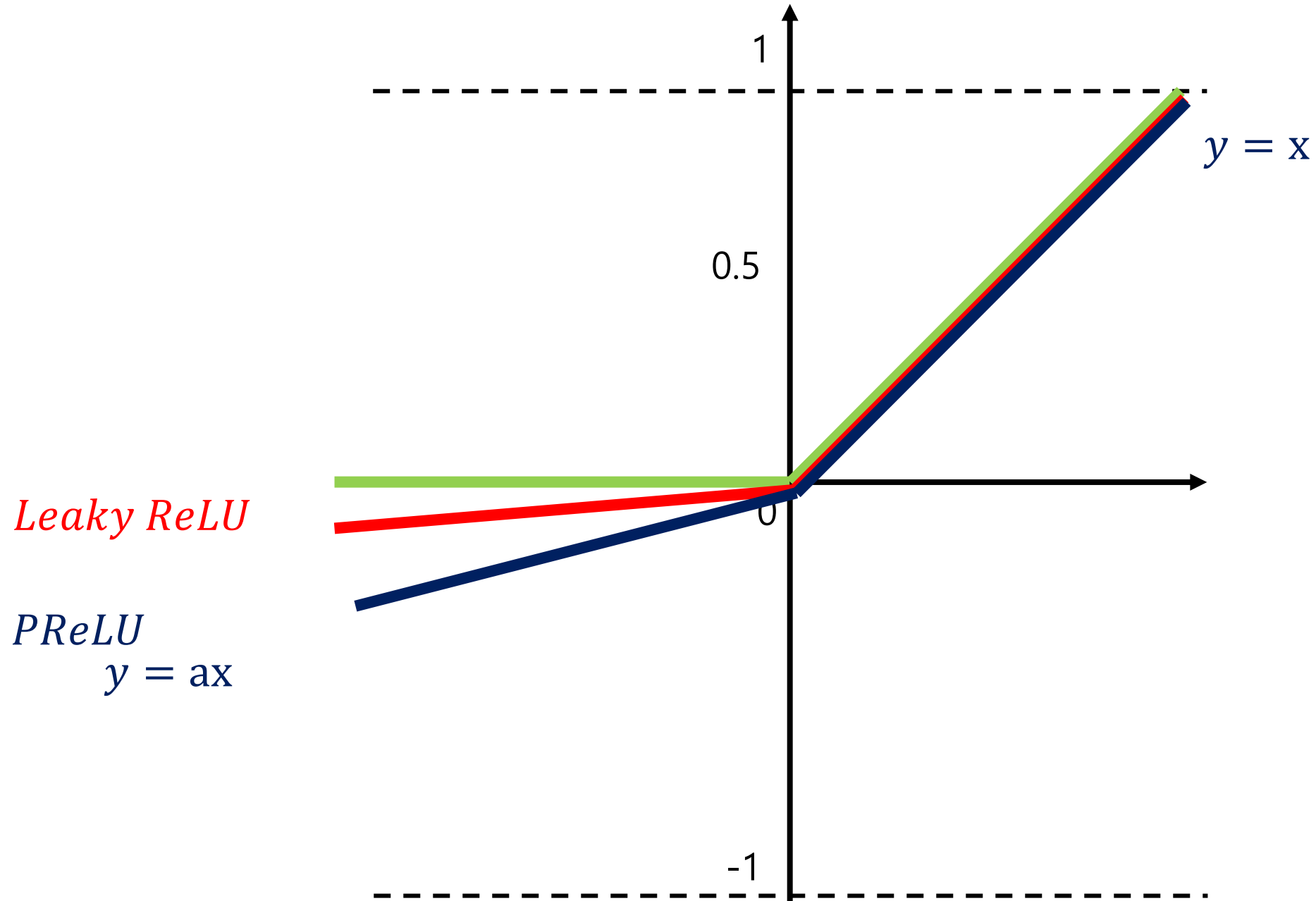
신경망 노드들의 입력값이 음수일 경우에도 학습이 가능하게 했습니다



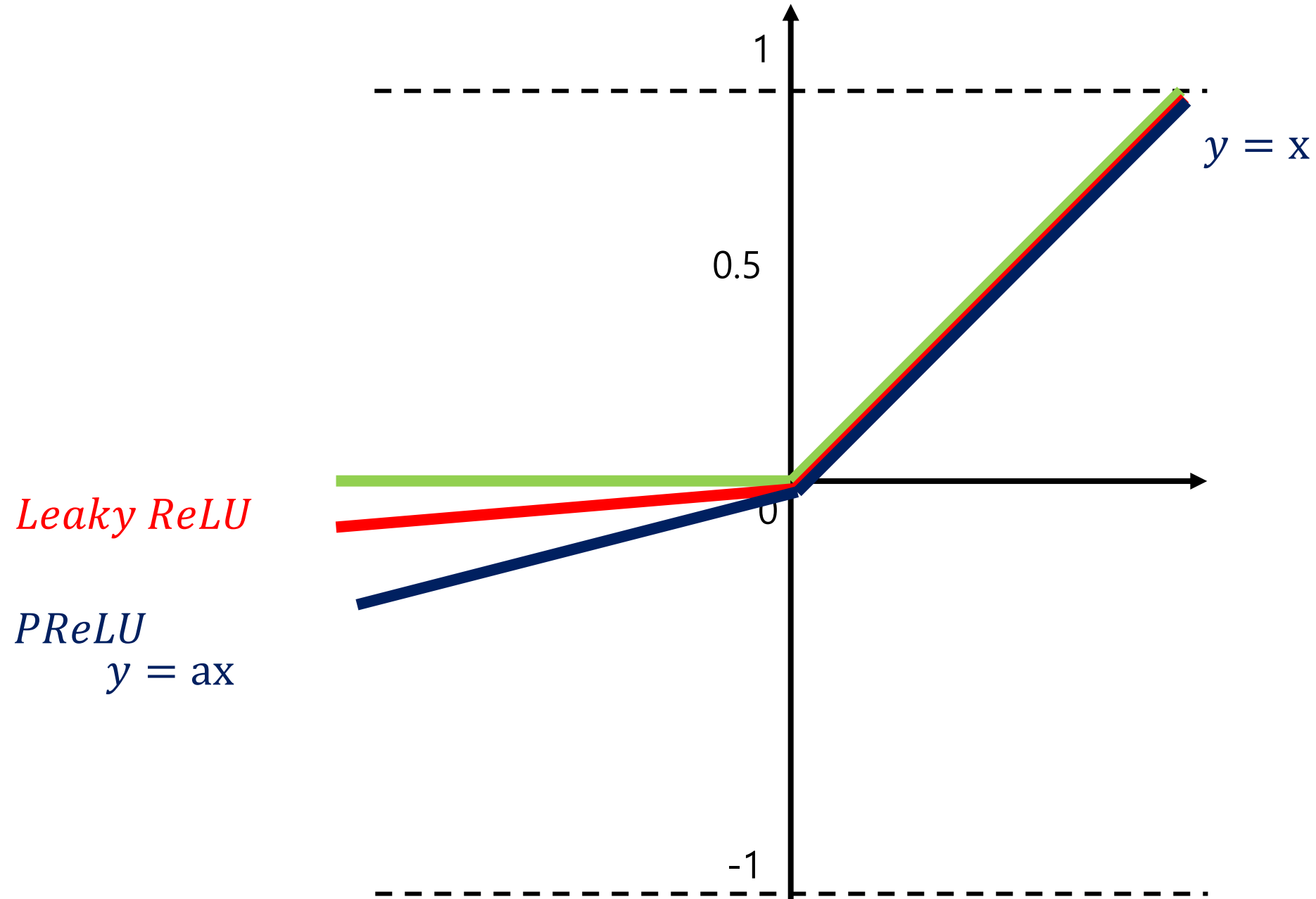
또다른 변형으로는 PReLU 활성화함수가 있습니다



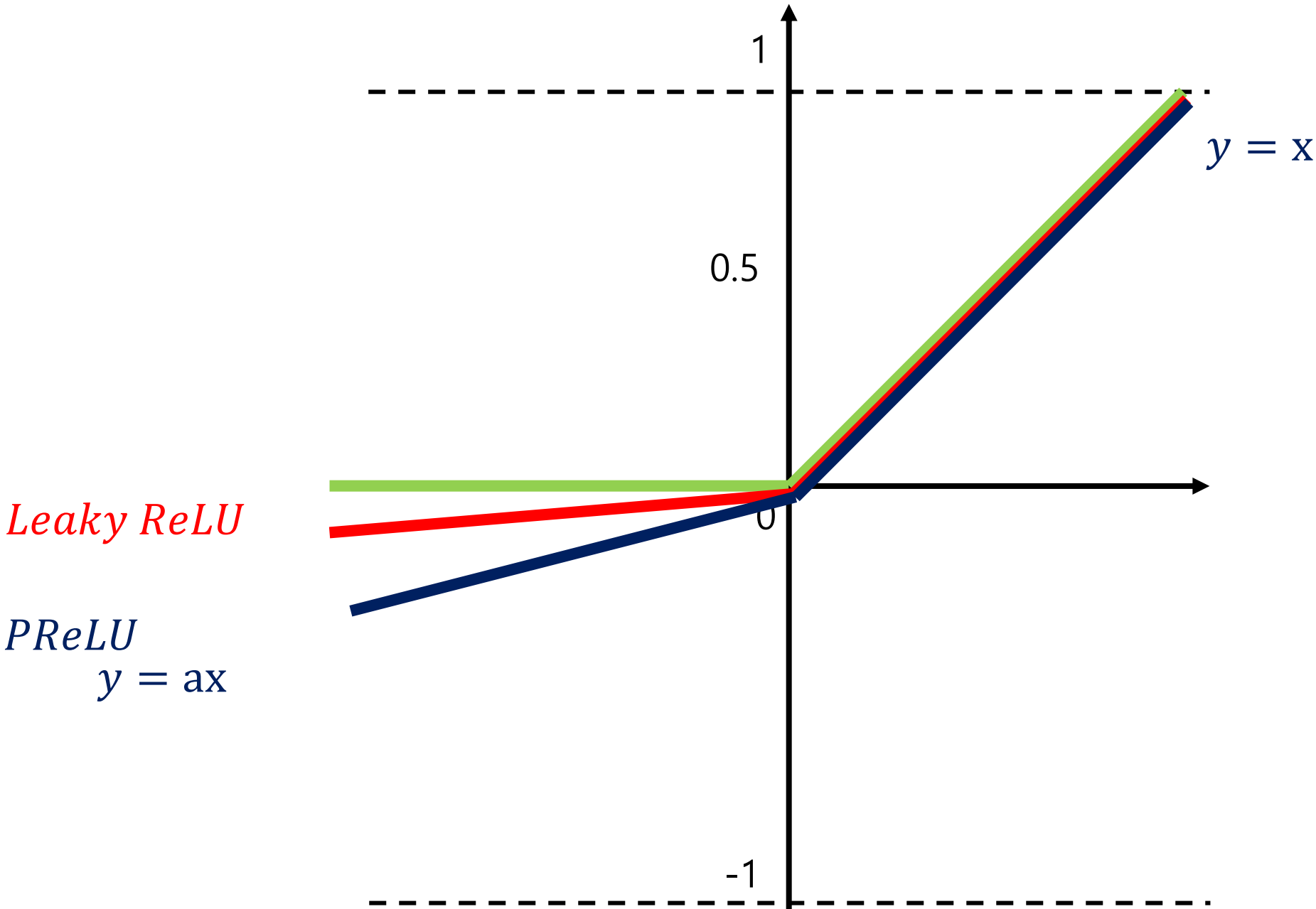
PReLU 는 음수기울기를 0.01 대신 a라는 변수를 줘서



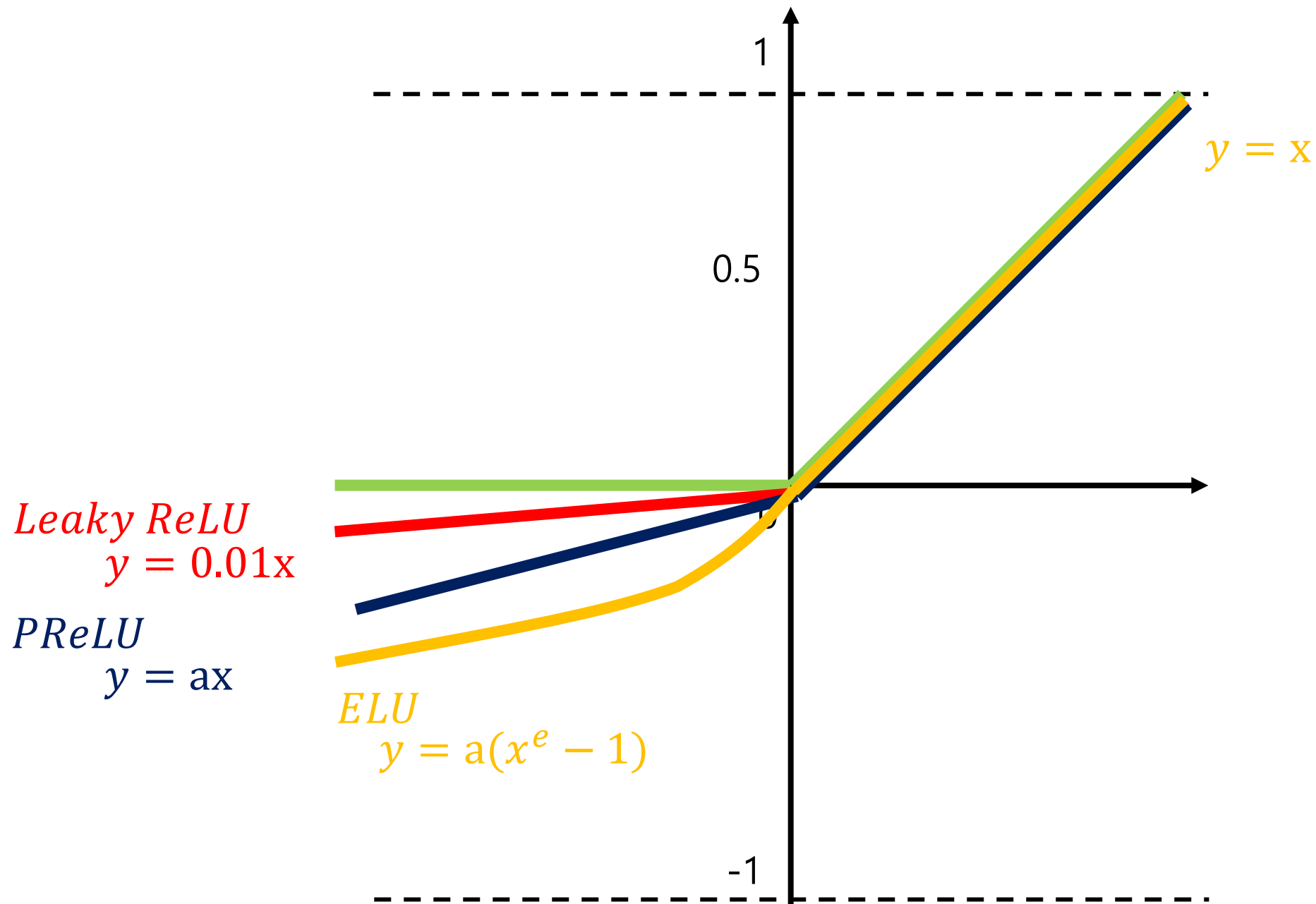
학습하는 모델에 따라 자유롭게 a값을 설정해서 학습효율을 높일 수 있는



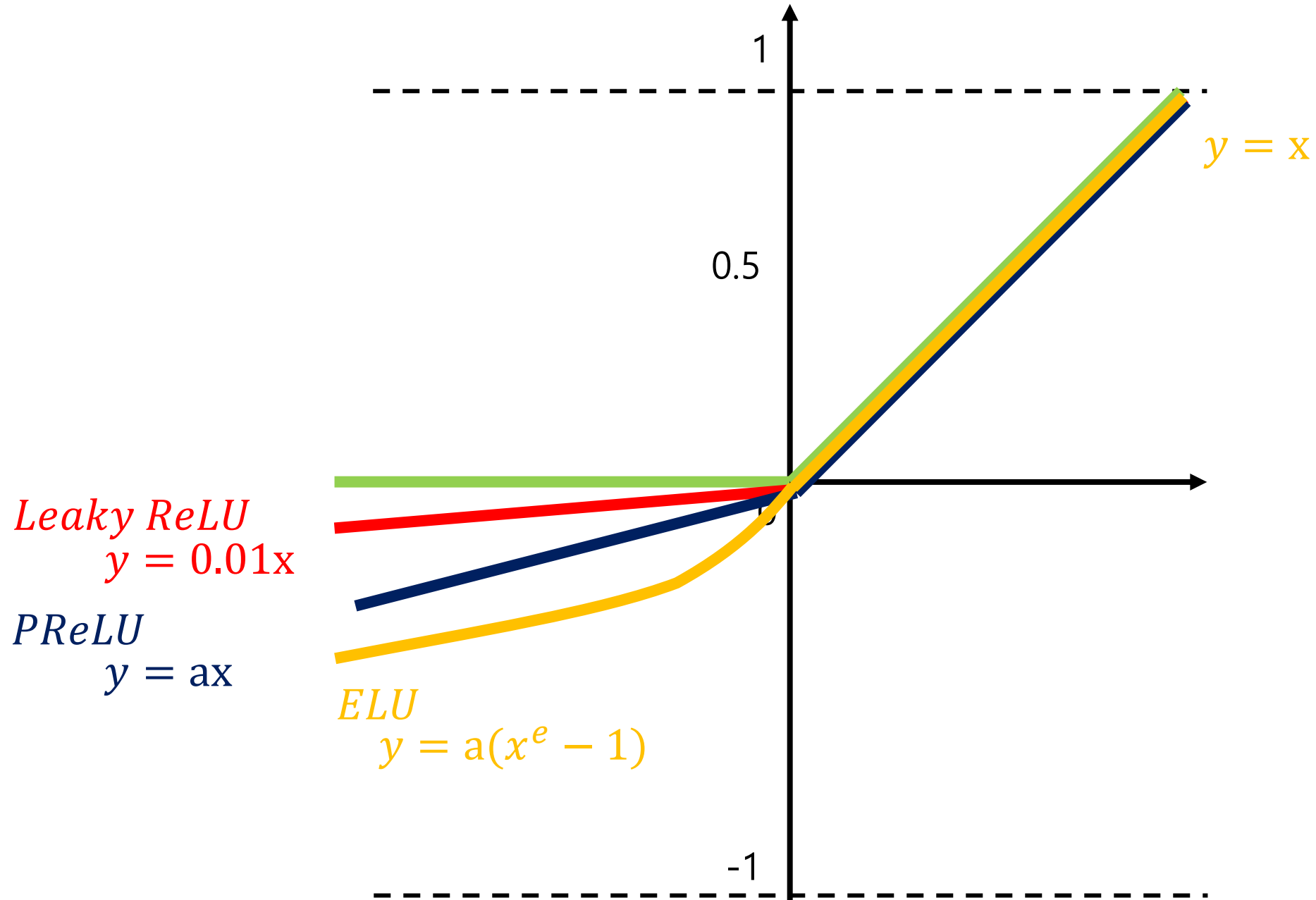
장점이 있습니다



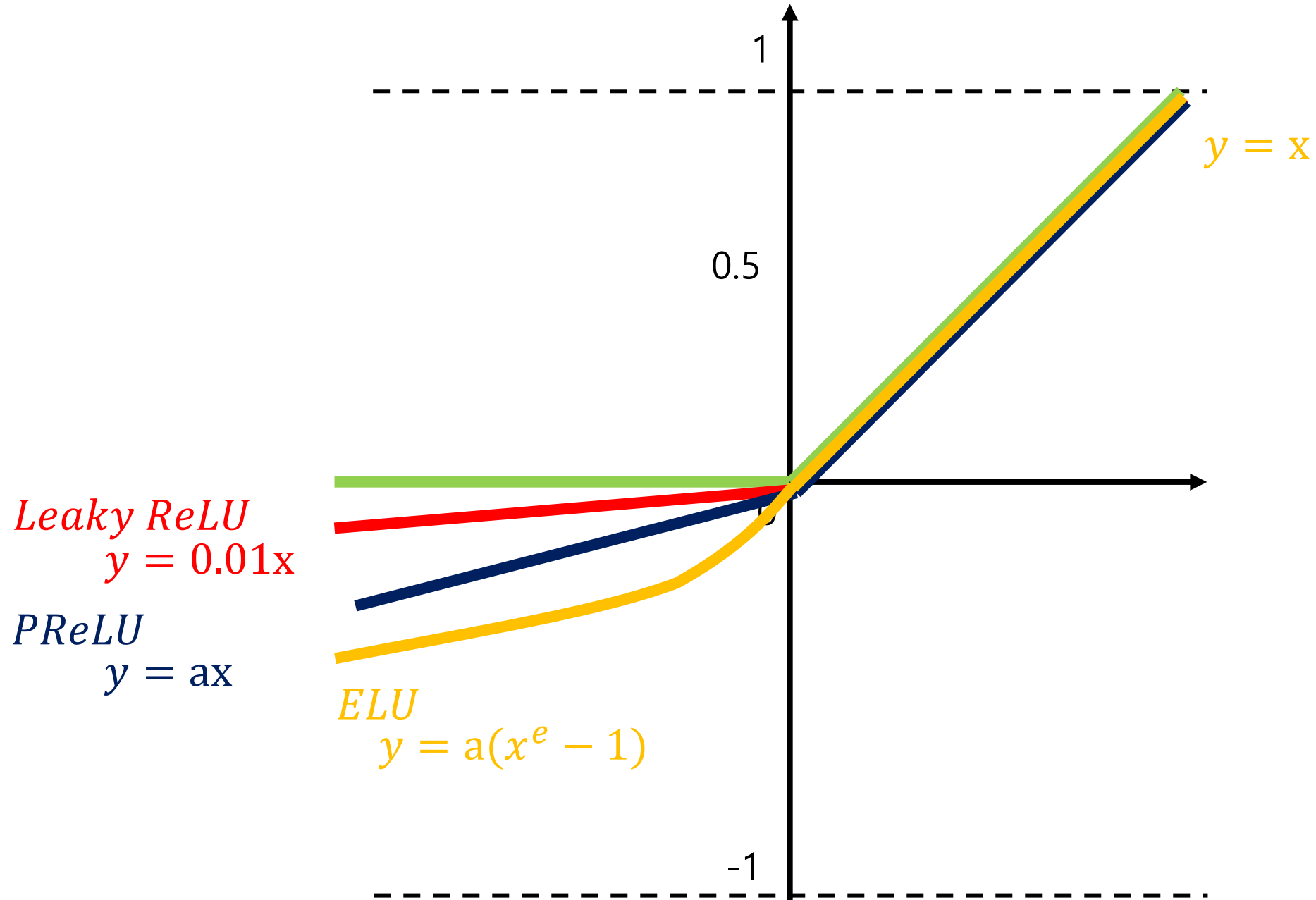
또 다른 변형으로는 ELU도 있습니다



ELU는 음수 입력값 부분을 곡선이 들어간 exponential함수를 사용하여

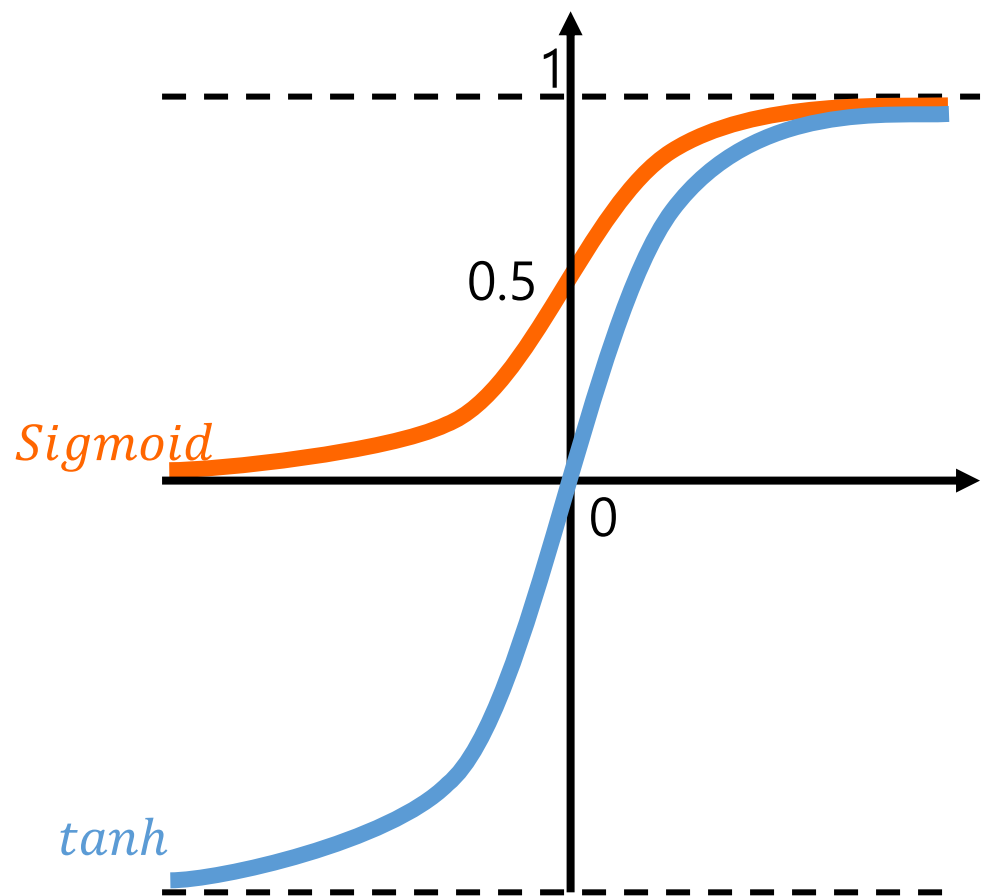


보다 학습이 빠르고 정확한 결과를 보여준다고 합니다

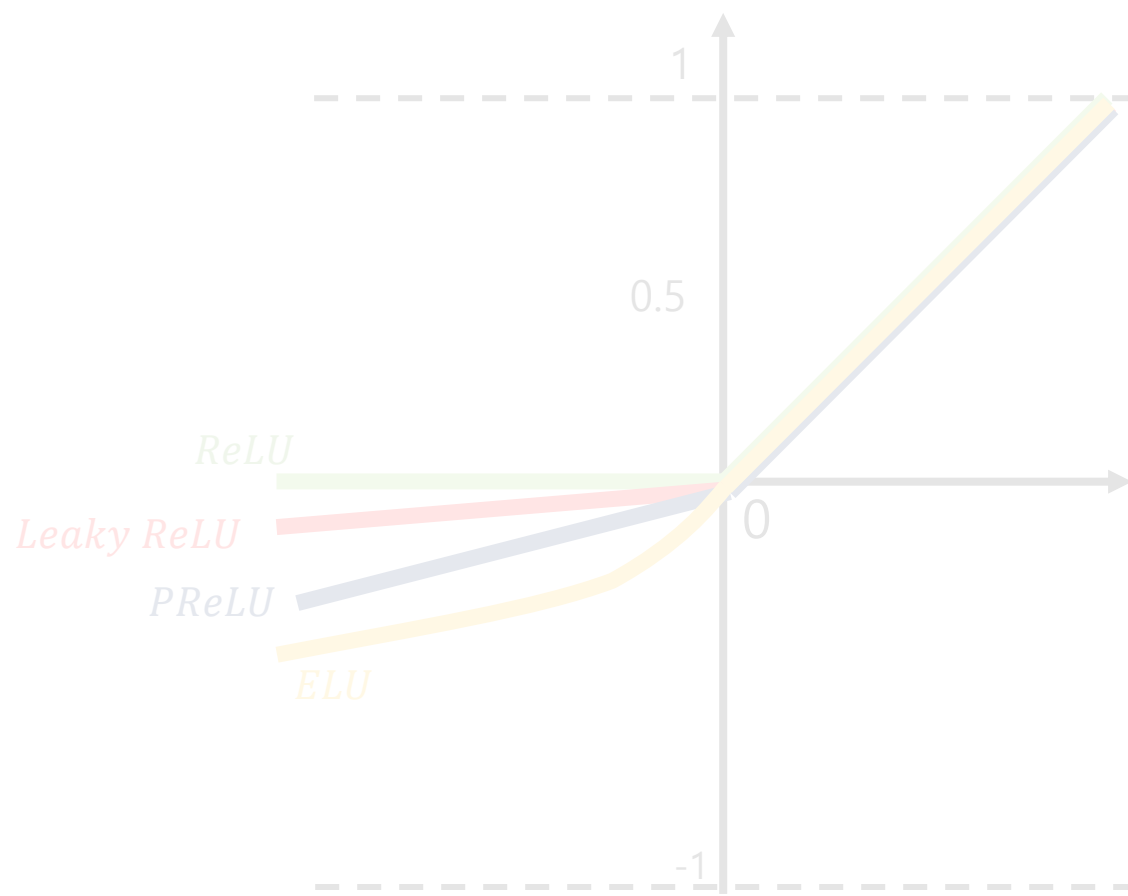


결론적으로는, 활성화 함수는 크게 sigmoid계열과

Sigmoid 계열

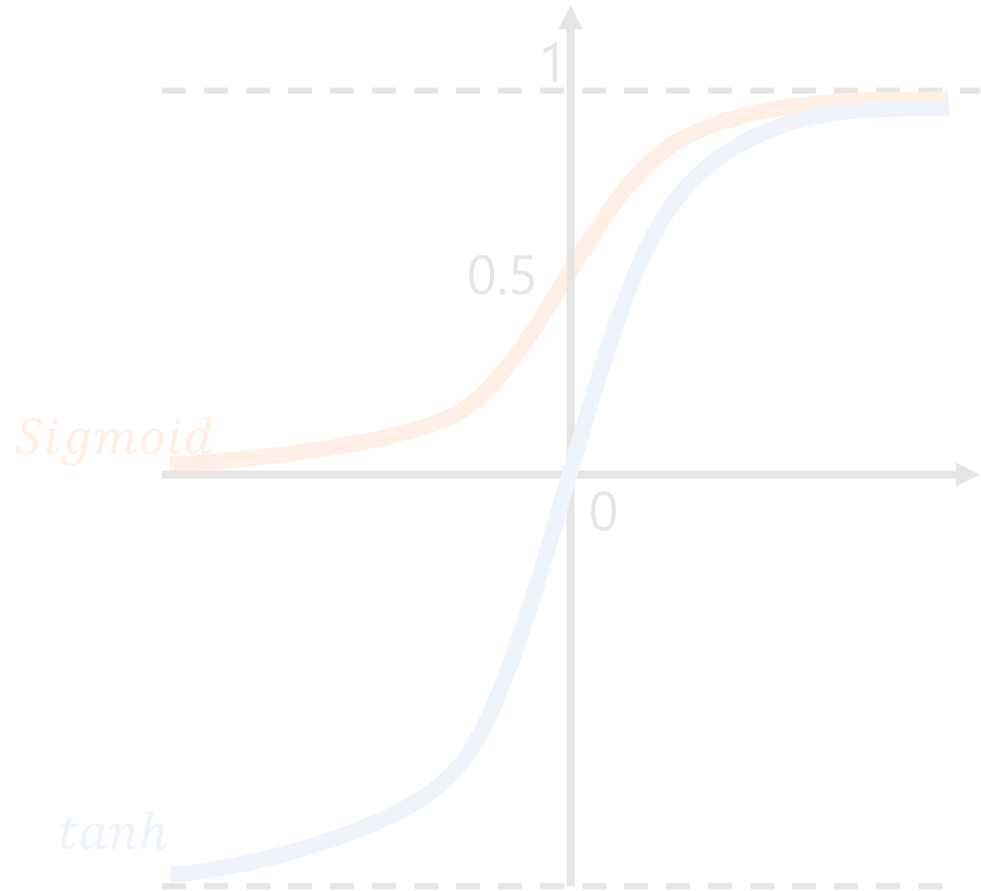


ReLU 계열

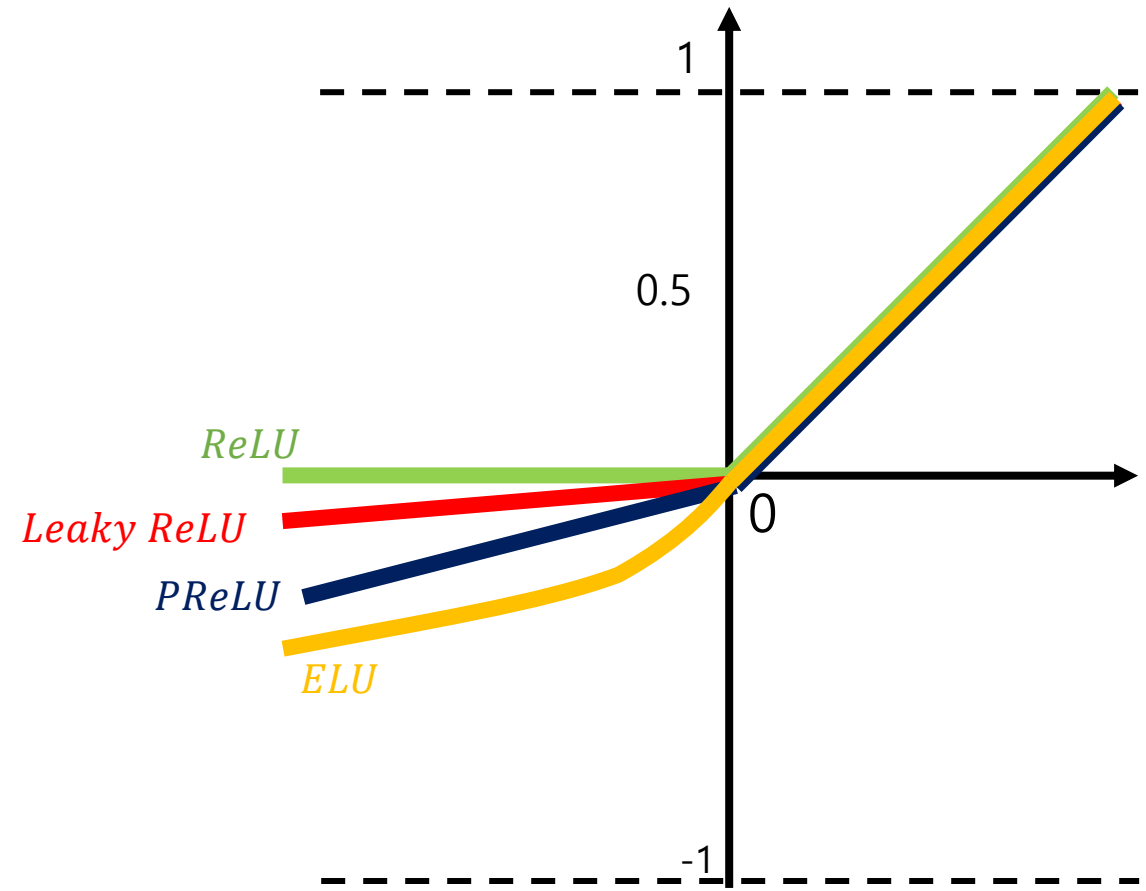


결론적으로는, 활성화 함수는 크게 ReLU 계열로 나눌수 있습니다

Sigmoid 계열

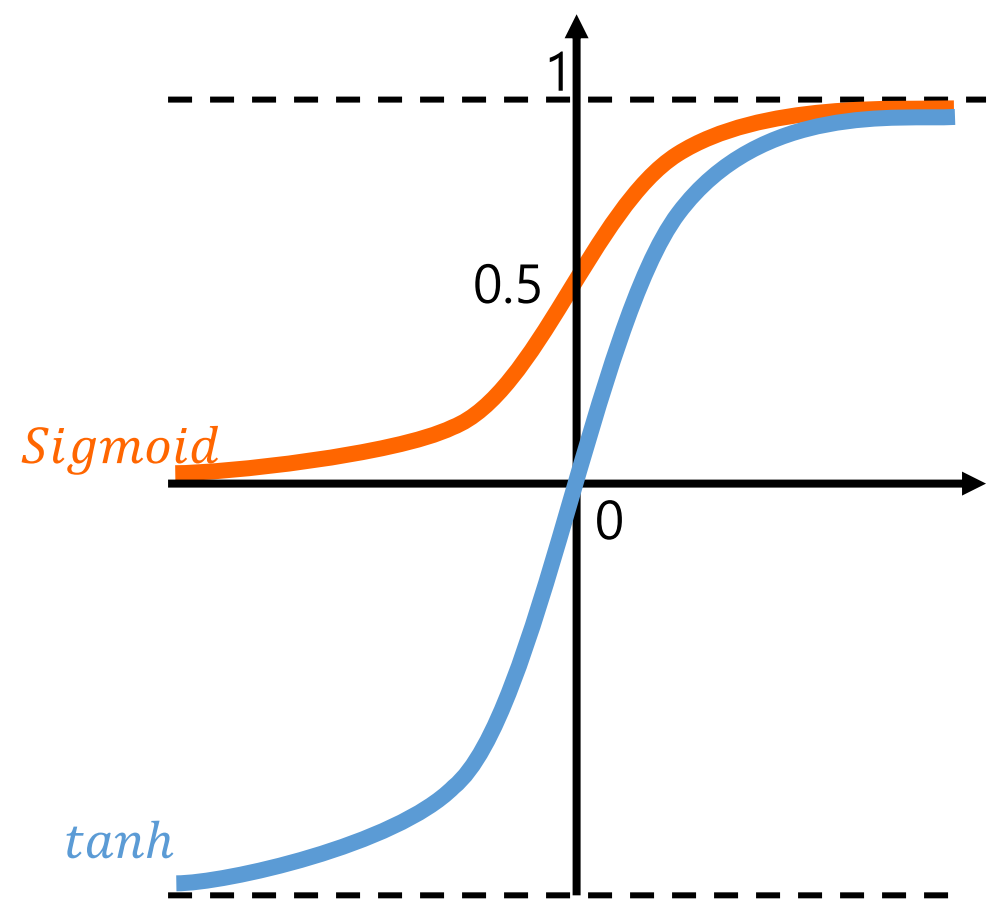


ReLU 계열

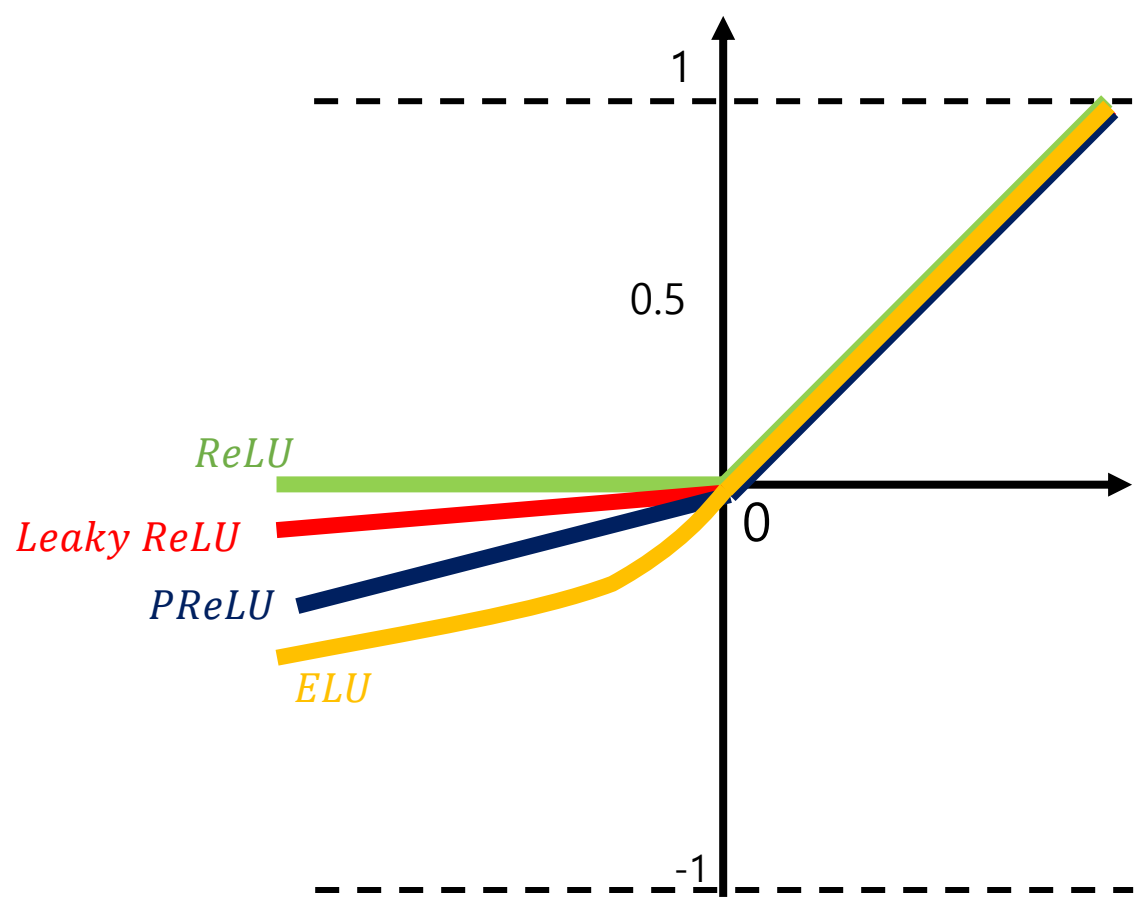


활성화 함수의 종류가 많다는 것은 그만큼 신경망 모델의 종류와

Sigmoid 계열

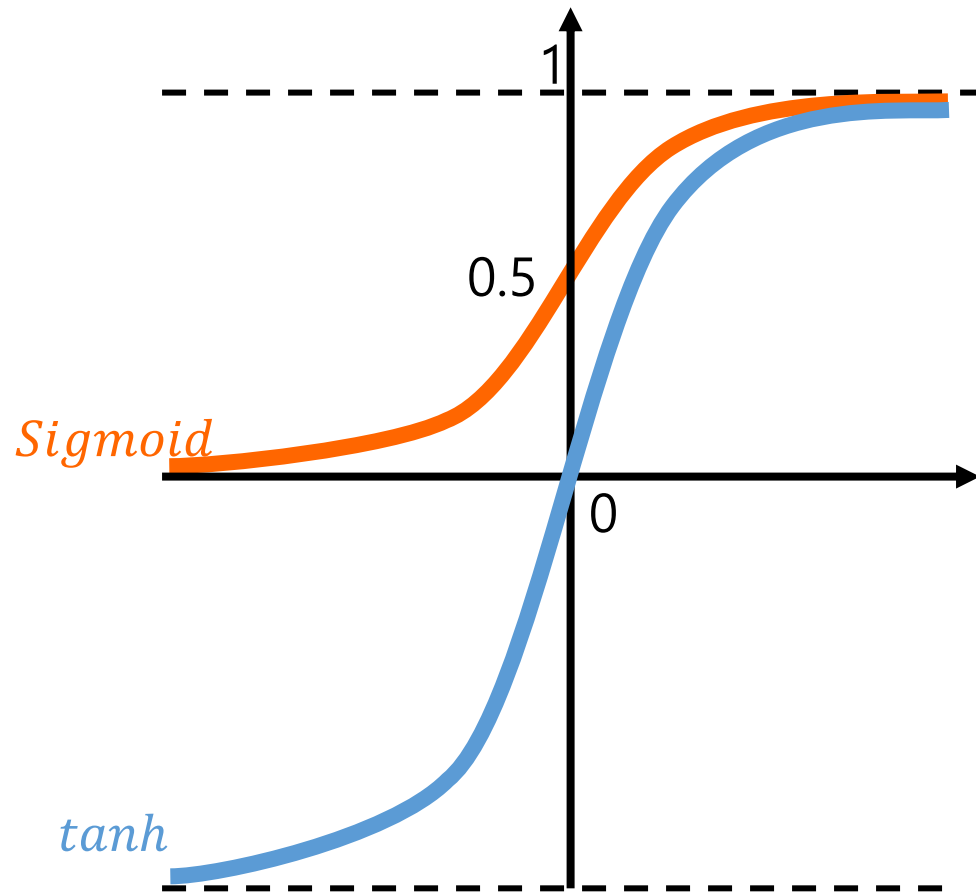


ReLU 계열

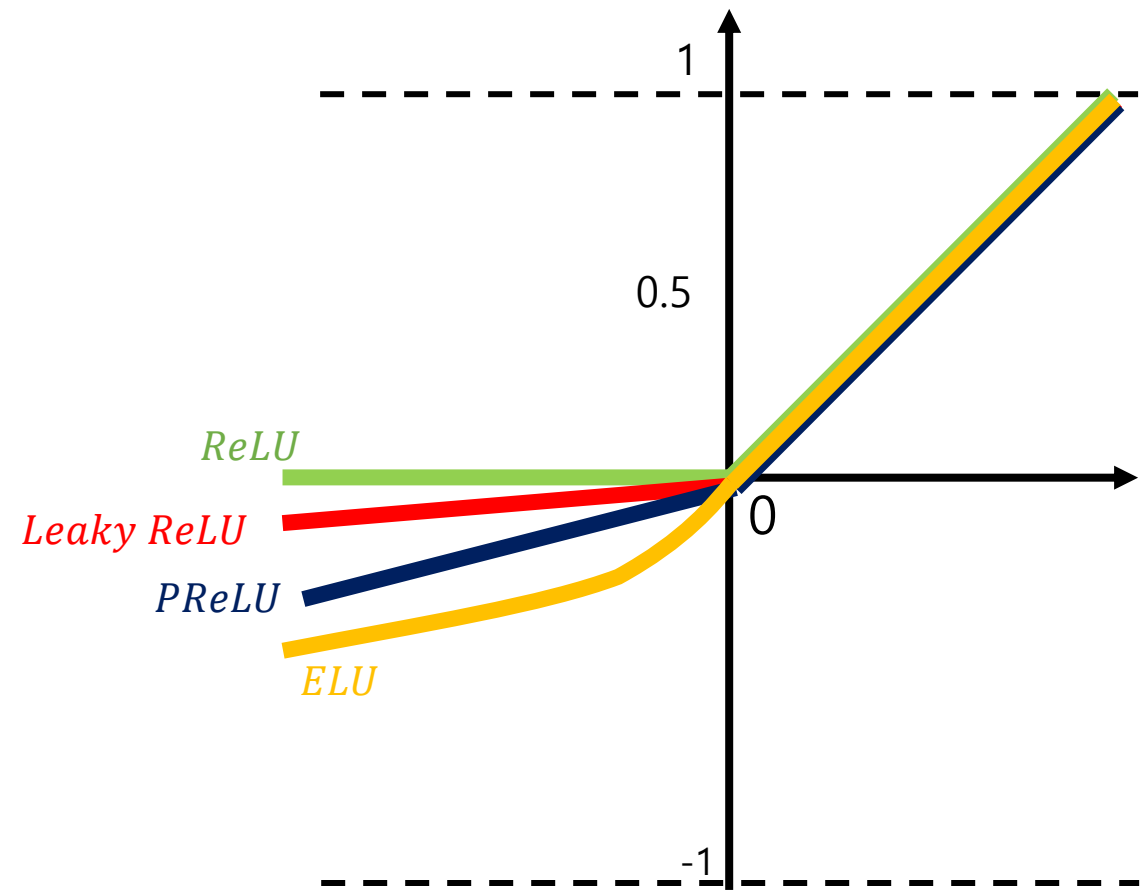


학습할 데이터의 종류가 다양하다는 것을 뜻하기도 하구요

Sigmoid 계열

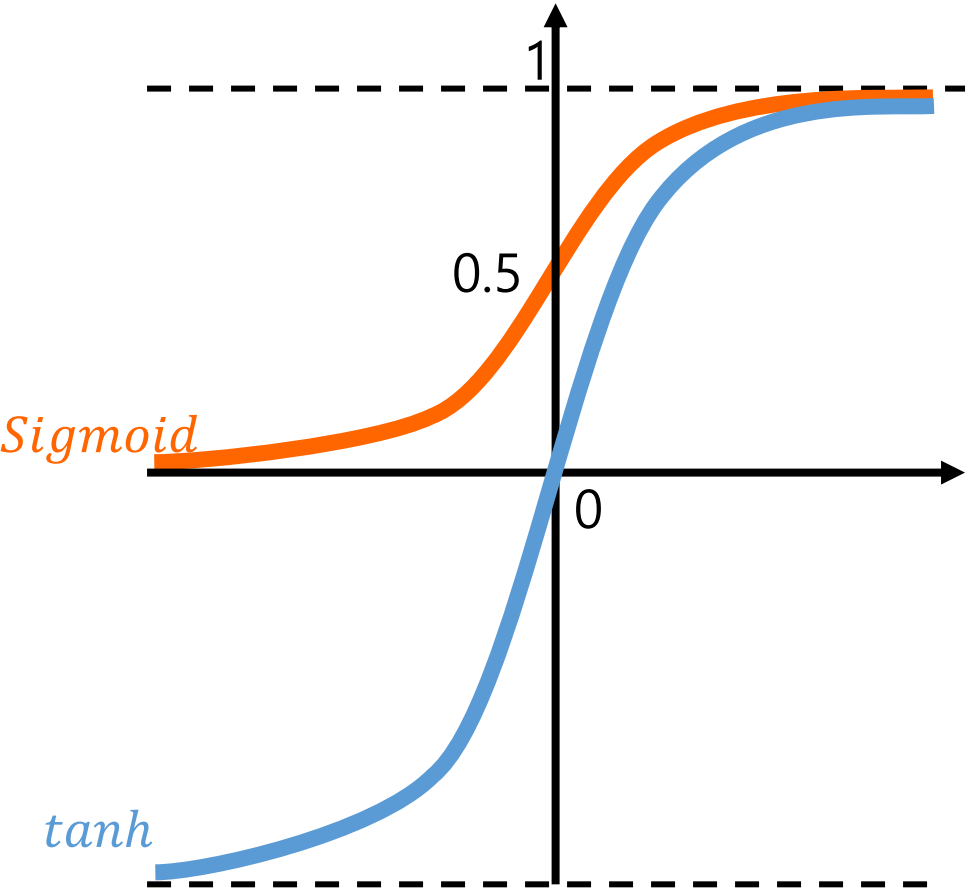


ReLU 계열

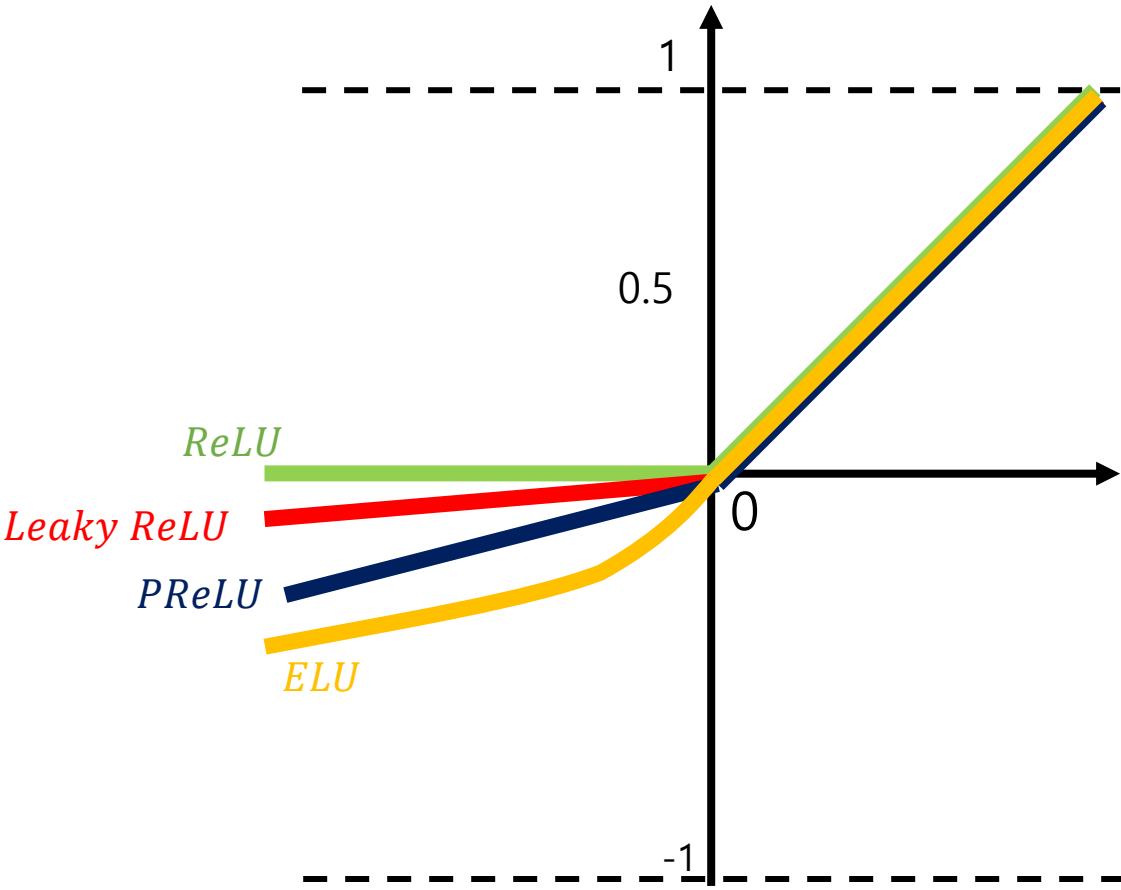


또한 완벽한 활성화 함수는 없다는 뜻도 됩니다

Sigmoid 계열

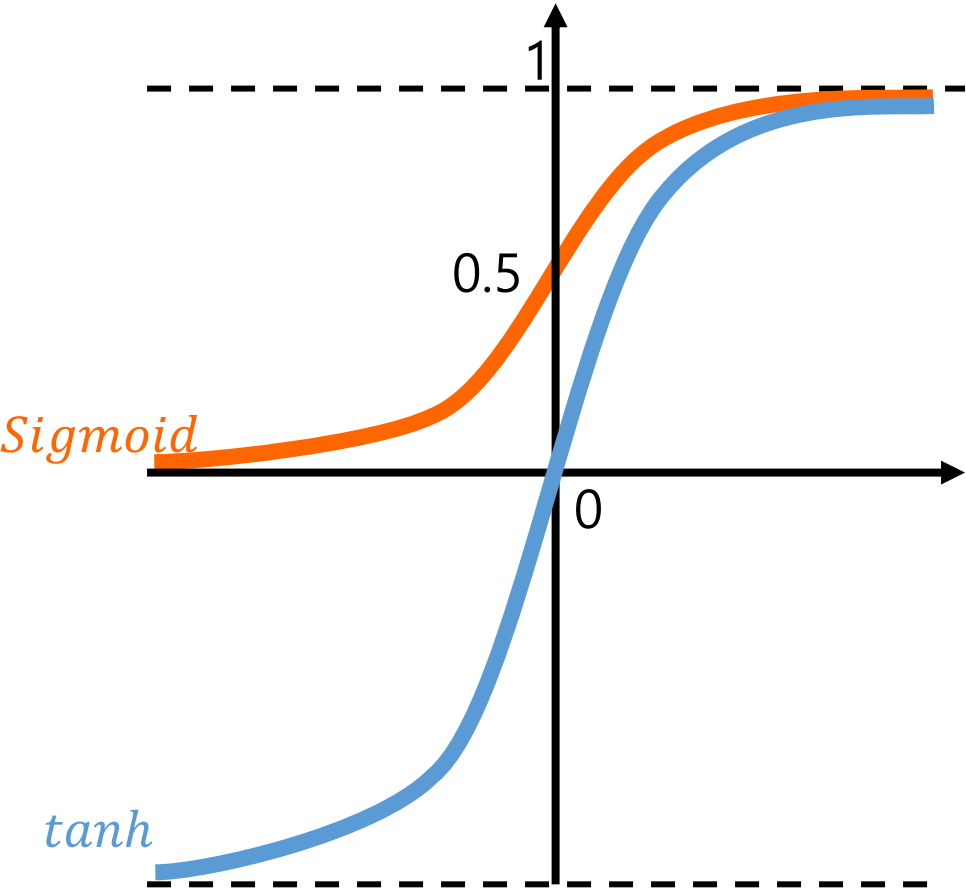


ReLU 계열

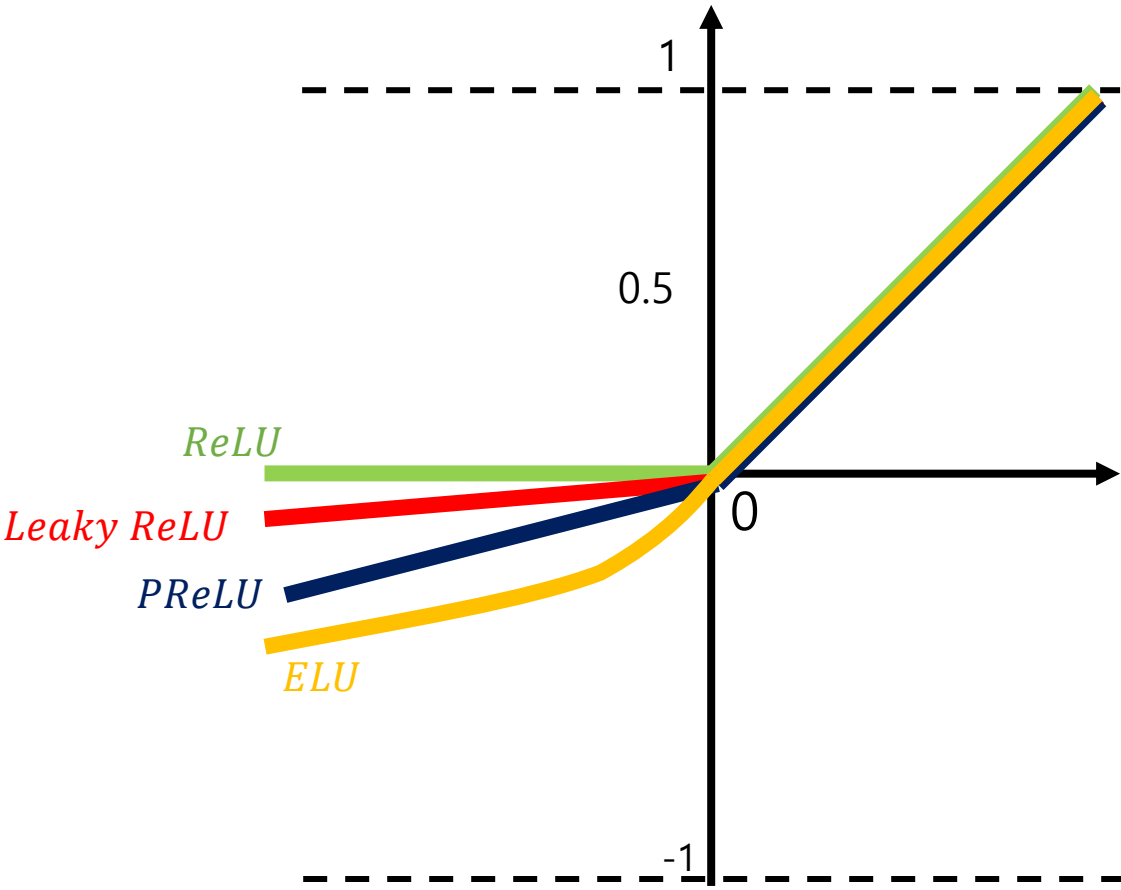


그러므로 각각의 특성을 잘 알아서 적절한 상황에 활용하면 좋겠습니다

Sigmoid 계열

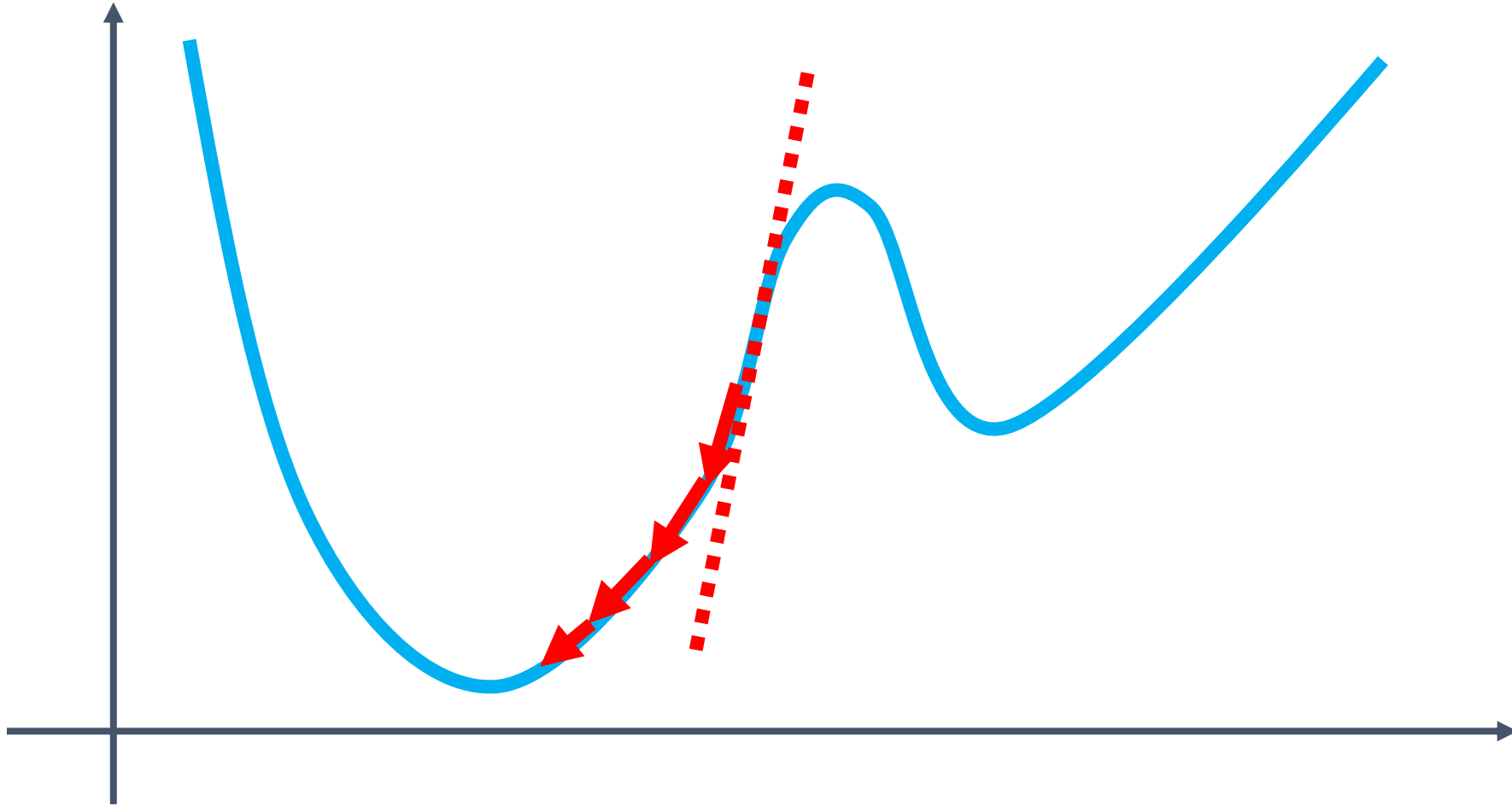


ReLU 계열

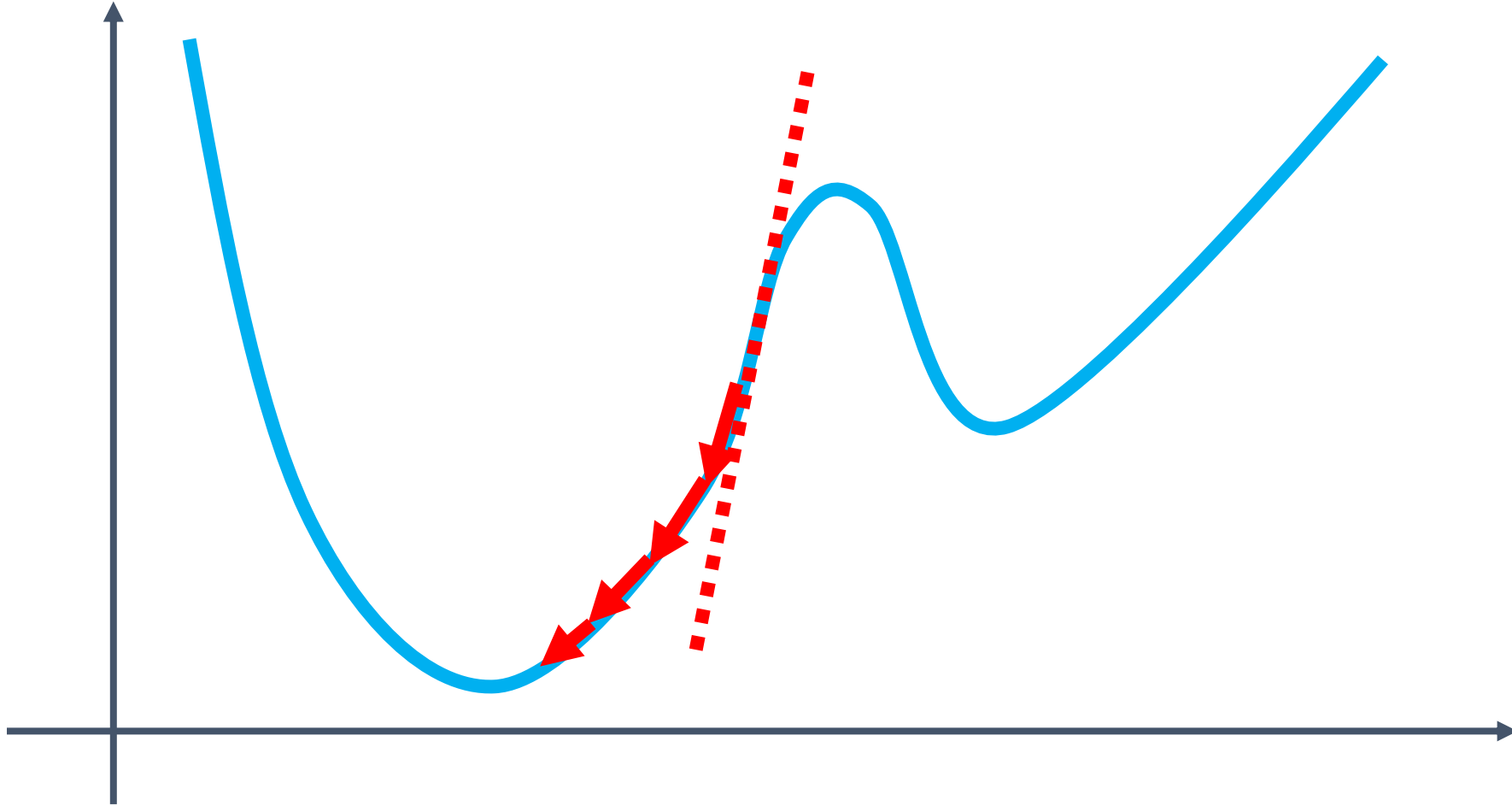


다음 시간에는 이번 영상에서도 잠깐 소개되었던

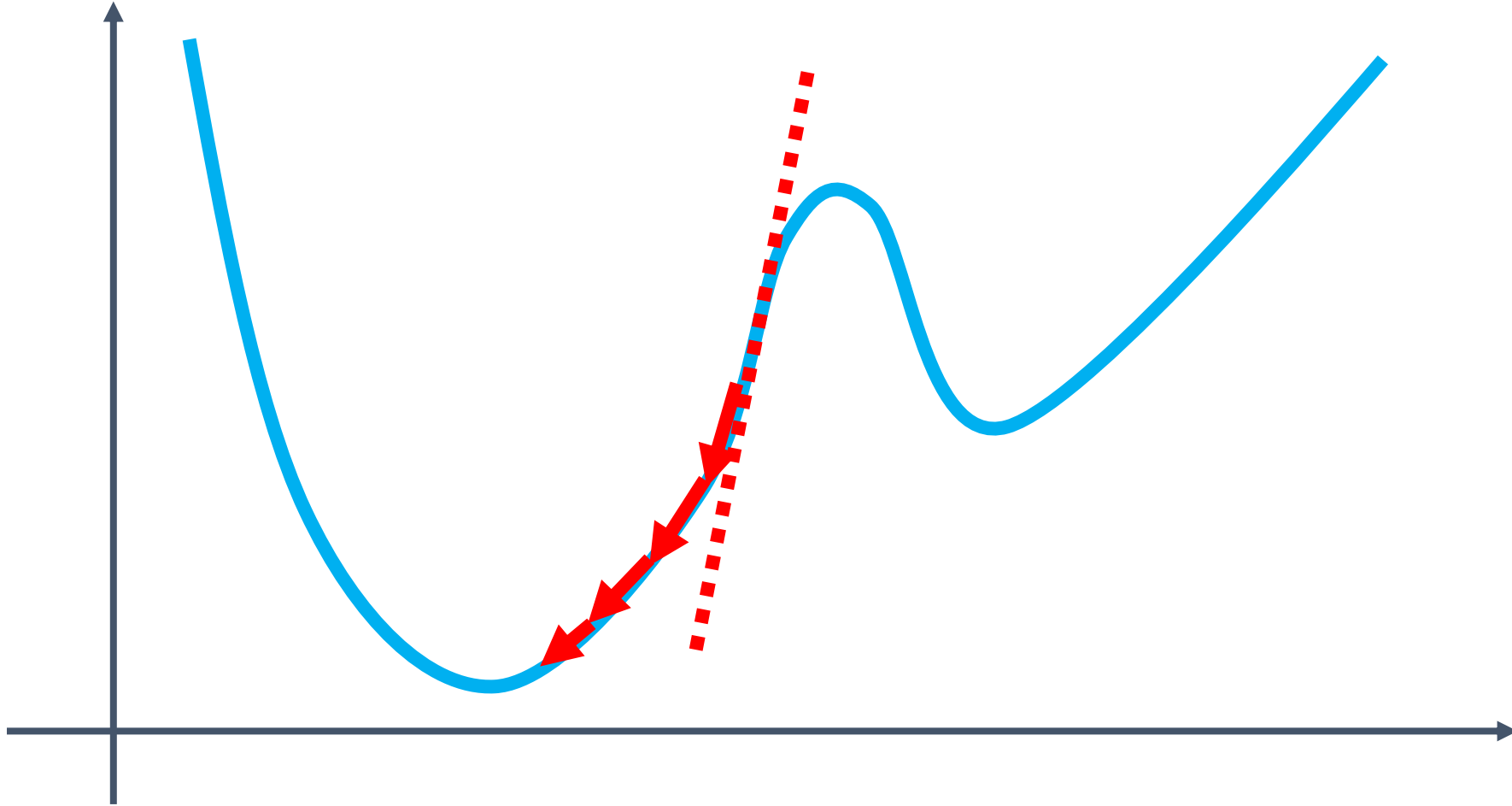
Gradient descent 경사하강법에 대해서 알아보도록 하겠습니다



Gradient descent 경사하강법은 신경망 학습에 있어서 너무너무



중요한 개념이니 꼭 시청해주시길 부탁드립니다!



감사합니다!

이 영상은 여러분의 관심과 사랑으로 제작됩니다
사용하실때는 출처 '신박AI'를 밝혀주세요





Copyright © 2024 by 신박AI

All rights reserved

본 문서(PDF)에 포함된 모든 내용과 자료는 저작권법에 의해 보호받고 있으며, 신박AI에 의해 제작되었습니다.

본 자료는 오직 개인적 학습 목적과 교육 기관 내에서의 교육용으로만 무료로 제공됩니다.

이를 위해, 사용자는 자료 내용의 출처를 명확히 밝히고,

원본 내용을 변경하지 않는 조건 하에 본 자료를 사용할 수 있습니다.

상업적 사용, 수정, 재배포, 또는 이 자료를 기반으로 한 2차적 저작물 생성은 엄격히 금지됩니다.

또한, 본 자료를 다른 유튜브 채널이나 어떠한 온라인 플랫폼에서도 무단으로 사용하는 것은 허용되지 않습니다.

본 자료의 어떠한 부분도 상업적 목적으로 사용하거나 다른 매체에 재배포하기 위해서는 신박AI의 명시적인 서면 동의가 필요합니다.

위의 조건들을 위반할 경우, 저작권법에 따른 법적 조치가 취해질 수 있음을 알려드립니다.

본 고지 사항에 동의하지 않는 경우, 본 문서의 사용을 즉시 중단해 주시기 바랍니다.

