9장 — 완전연결 신경망과 역전파

선형모델의 한계에서 신경망까지 — 층을 쌓고, 순전파 계산 및 역전파 공식 유도

조현태

2026

아주대학교 수학과

경고

수업자료에는 저작권법 제25조 제2항에 의해 수업 목적으로 이용한 저작물이 포함되어 있으므로 본 수업자료를 외부에 공개, 게시하는 것을 금지하며 이를 위반하는 경우 저작권 침해로서 관련법에 따라 처벌될 수 있습니다.

또한, 수업 내용 중 일부를 무단 캡처하여 배포, 전송할 경우 저작권 침해 또는 초상권 침해에 해당할 수 있으니 주의 바랍니다.

차례

선형모델의 한계에서 신경망으로

손실함수와 경사하강법

계산그래프와 역전파: 수치 예제

역전파의 일반 공식

역전파 계산 예시와 NumPy 구현: 1-3-2-1 신경망

함수족의 표현력: 보편 근사 정리

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

학습목표

1. 선형모델이 표현할 수 없는 데이터의 비선형 관계를 신경망으로 접근
2. 수식을 통한 1-1-1, 1-1-1-1, 1-2-1 구조 이해
3. 일반적인 d_x-d_1-$\cdots$-d_{L-1}-d_y 구조의 순전파를 벡터-행렬 표기로 서술
4. 계산그래프 위에서 역전파의 공식 $\boldsymbol{\delta}^{(\ell)} = \mathbf{D}^{(\ell)}{\mathbf{W}^{(\ell+1)}}^{\top}\boldsymbol{\delta}^{(\ell+1)}$과 $\partial y/\partial \mathbf{W}^{(\ell)} = \boldsymbol{\delta}^{(\ell)}{\boldsymbol{h}^{(\ell-1)}}^{\top}$을 유도
5. 역전파 공식의 이점, 의미 파악
6. NumPy로 순전파·역전파를 직접 구현하고, PyTorch로 회귀용 FCNN을 학습
7. 보편 근사 정리가 의미하는 것과 그렇지 않은 것을 이해

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

선형모델의 한계에서 신경망으로

비선형성이 필요한 이유 — 선형변환을 두 번 합성하면?

- 편향(b)이 있는 $wx + b$: 수학적으로 affine 변환(편의상 선형변환)
- 선형모델 $y = wx + b$는 파라미터 하나당 (x, y) 평면 위의 직선 하나만 표현
- 비선형 함수(예: $y = e^{3x} - \log(|x| + 1)$)는 어떤 (w, b)로도 재현 불가
- 한계는 파라미터의 개수가 아니라 함수족 $\mathcal{F}_{\text{lin}} = \{f(x) = wx + b : (w, b) \in \mathbb{R}^2\}$의 크기
- 선형변환 추가? $\Rightarrow$ 파라미터를 넷으로 늘려도 결과는 다시 $\mathcal{F}_{\text{lin}}$의 원소

$$g(x) = w^{(2)}\left(w^{(1)}x + b^{(1)}\right) + b^{(2)} = \underbrace{\left(w^{(2)}w^{(1)}\right)}_{=:\,w}x + \underbrace{\left(w^{(2)}b^{(1)} + b^{(2)}\right)}_{=:\,b}$$

$$\Rightarrow\ g(x) \in \mathcal{F}_{\text{lin}}$$

- 결론: 선형변환의 선형변환은 다시 선형변환 — 만들 수 있는 함수의 집합은 그대로

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

아이디어 — 선형변환 사이에 비선형 함수를

- $\mathcal{F}_{\text{lin}}$에 없는 함수를 얻으려면 합성하는 함수 가운데 적어도 하나는 선형이 아니어야 함
- 선형변환 사이에 비선형 함수 σ를 추가

$$x \longmapsto w^{(2)}\,\sigma\big(w^{(1)}x + b^{(1)}\big) + b^{(2)}$$

  σ가 affine이 아니고 $w^{(1)}, w^{(2)} \neq 0$이면 이 함수는 $\mathcal{F}_{\text{lin}}$에 속하지 않음
- (1장) “기저함수의 모양까지 파라미터가 정하는” 구조
  - $\sigma(w^{(1)}x + b^{(1)})$이 선형회귀의 x, x^2 같은 기저함수 역할.
  - 다만 x, x^2은 모양이 고정이고, 이것은 $w^{(1)}, b^{(1)}$에 따라 모양이 달라짐
- 신경망의 핵심 설계 원리. 층을 쌓아 새로운 함수를 얻을 수 있는 것도 이 비선형 함수 덕분

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

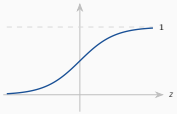
정의 9.1 — 활성화함수(activation function)

선형변환의 결과에 성분별로 적용하는 함수 σ를 **활성화함수(activation function)**라 한다.

- 대표적인 예: Sigmoid $\sigma(z) = 1/(1+e^{-z})$ (참고: 로지스틱 회귀 출력도 동일)
- 비선형성의 추가가 주목적 $\rightarrow$ 반드시 시그모이드일 필요는 없음 (다음 페이지)
- 어떤 활성화함수가 유리한지는 문제와 신경망의 깊이에 따라 다름 (10장 일부)
- 이 노트 예제 등에서 특별한 언급이 없으면 σ = 시그모이드

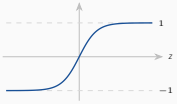
© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

네 가지 활성화함수의 정의와 모양



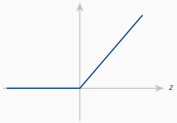
(a) Sigmoid

$$\sigma(z) = \frac{1}{1+e^{-z}}$$



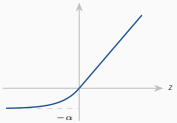
(b) hyperbolic tangent

$$\tanh(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}}$$



(c) **ReLU**

$$\mathrm{ReLU}(z) = \max(0, z)$$



(d) **ELU** (exponential linear unit)

($\alpha > 0$, 그림은 $\alpha = 1$)

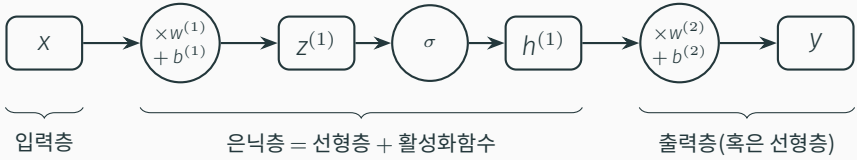
$$\mathrm{ELU}(z) = \begin{cases} z, & z > 0 \\ \alpha(e^{z}-1), & z \le 0 \end{cases}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

1-1-1 구조 — 계산그래프로 그리기

선형변환 $z^{(1)} = w^{(1)}x + b^{(1)}$ 의 결과에 σ 를 적용해 $h^{(1)} = \sigma(z^{(1)})$ 을 만들고, $h^{(1)}$ 에서 y 로 가는 선형변환을 하나 더 둠

$$z^{(1)} = w^{(1)}x + b^{(1)}, \qquad h^{(1)} = \sigma(z^{(1)}), \qquad y = w^{(2)}h^{(1)} + b^{(2)}$$



- 계산그래프 내 둥근 상자: 변수 노드($x, z^{(1)}, h^{(1)}, y$), 원: 연산 노드(선형변환, σ)
- 파라미터 $\boldsymbol{\theta} = (w^{(1)}, b^{(1)}, w^{(2)}, b^{(2)})$.

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

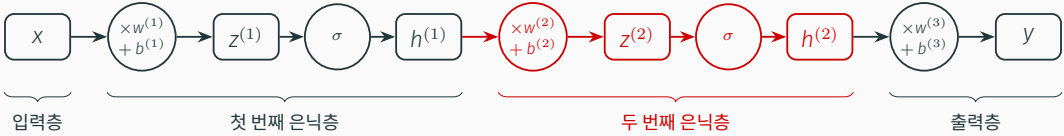
계산그래프 내 새로운 용어 – 노드(node), 층(layer)

- 신경망의 **노드**: 변수 노드 가운데 x, $h^{(1)}$, y. $z^{(1)}$은 $h^{(1)}$을 만드는 중간 계산이므로 노드로 세지 않음
- **입력노드** x, **출력노드** y (이 예시에서는 실수값)
- **은닉노드** $h^{(1)}$: 입력·출력을 제외한 변수; 특정 조건 없이 중간에 설계된 값 (숨어 있다)
  - 은닉노드 = 선형변환 + 활성화함수 (순서 중요)
- **층(layer)**: x에서 y로 가는 단계마다 놓인 노드의 위치.
  - 지금은 층별 노드가 하나 $\Rightarrow$ 입력층 x, 은닉층 $h^{(1)}$, 출력층 y
- **구조 표기**: 층마다 노드 수를 입력층부터 하이픈(-)으로 연결 $\Rightarrow$ 1-1-1.
- 위첨자 $(\ell) = \ell$번째 층의 변수. 입력층은 0번째 은닉층 $h^{(0)} = x$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

1-1-1-1 구조 (1-1-1 구조에 은닉층 추가)

첫 번째 은닉층 $h^{(1)}$에 이어서 두 번째 은닉층 $h^{(2)}$을 추가

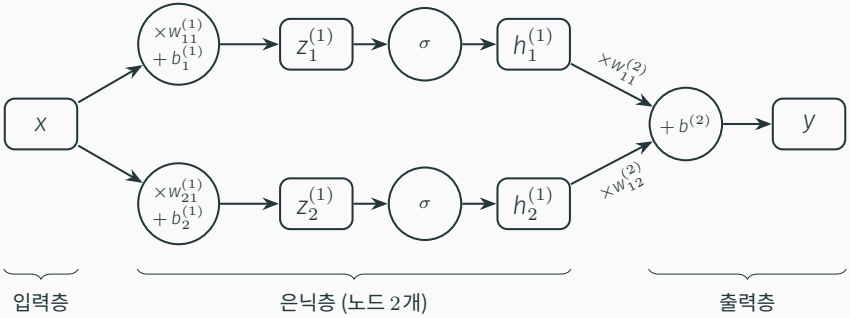


$$z^{(1)} = w^{(1)}x + b^{(1)}, \qquad h^{(1)} = \sigma(z^{(1)}),$$
$$z^{(2)} = w^{(2)}h^{(1)} + b^{(2)}, \qquad h^{(2)} = \sigma(z^{(2)}),$$
$$y = w^{(3)}h^{(2)} + b^{(3)}$$

파라미터 $\boldsymbol{\theta} = (w^{(1)}, b^{(1)}, w^{(2)}, b^{(2)}, w^{(3)}, b^{(3)})$, 1-1-1-1 구조

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

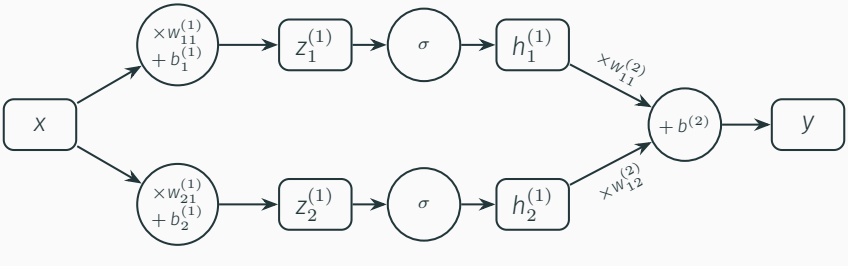
1-2-1 구조 — 같은 층에 노드를 늘리기



- 하나의 은닉층 안에 노드를 두 개 설정 $\rightarrow$ 폭(width)을 늘림
- 층 안에 노드가 여러 개면, 아래첨자로 노드 번호 추가
  - $z_j^{(1)}$: (1)번째 층의 j번째 노드
  - 가중치 $w_{j1}^{(1)}$ 의 아래첨자: (1)번째 층의 j번째 노드, 직전층 ((0) 번째 층)의 1번째 노드에 할당한 파라미터

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

1-2-1 구조 — 같은 층에 노드를 늘리기



$$x \longrightarrow \begin{aligned} z_1^{(1)} &= w_{11}^{(1)} x + b_1^{(1)}, \quad h_1^{(1)} = \sigma(z_1^{(1)}) \\ z_2^{(1)} &= w_{21}^{(1)} x + b_2^{(1)}, \quad h_2^{(1)} = \sigma(z_2^{(1)}) \end{aligned} \longrightarrow y = w_{11}^{(2)} h_1^{(1)} + w_{12}^{(2)} h_2^{(1)} + b^{(2)}$$

파라미터 $\boldsymbol{\theta} = (w_{11}^{(1)}, w_{21}^{(1)}, b_1^{(1)}, b_2^{(1)}, w_{11}^{(2)}, w_{12}^{(2)}, b^{(2)})$, 총 7개

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

폭을 늘리면 어떤 이점이? — 기저함수의 관점

두 은닉 노드는 같은 활성화함수 σ를 쓰지만, 각 노드의 가중치와 편향을 따로 학습하므로 $h_1^{(1)}$과 $h_2^{(1)}$는 서로 다른 함수가 될 수 있음

- (1,6장) 기저함수 표기 $\varphi_j(x) := \sigma(w_{j1}^{(1)}x + b_j^{(1)})$로 쓰면 1-2-1 구조의 경우

$$y = w_{11}^{(2)}\,\varphi_1(x) + w_{12}^{(2)}\,\varphi_2(x) + b^{(2)}$$

  — 기저함수의 선형결합. 기저함수의 모양 φ_j까지 데이터로 결정

- 두 노드의 $(w_{j1}^{(1)}, b_j^{(1)})$가 같으면 $\varphi_1 = \varphi_2$가 되어 기저함수 하나짜리 표현으로 축소

폭을 늘려 표현가능한 함수의 종류를 높인다

노드마다 다른 기저함수를 사용하여 신경망이 만들 수 있는 함수의 종류를 넓힌다는것

(챕터 마지막 universal approximation theorem참고!)

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

정의 9.2 — 완전연결 신경망(fully connected neural network, FCNN)

입력층 차원 $d_0 = d_x$, 출력층 차원 $d_L = d_y$, $L-1$개의 은닉층(층별 노드 수 $d_1, \dots, d_{L-1}$)으로 구성된 $L+1$개의 층을 가지는 **완전연결 신경망** $f : \mathbb{R}^{d_x} \to \mathbb{R}^{d_y}$은 다음 구조로 정의:

입력 $h^{(0)} := x \in \mathbb{R}^{d_0}$을 시작으로, 각 $\ell = 1, \dots, L$에 대하여 가중치 행렬 $\mathbf{W}^{(\ell)} \in \mathbb{R}^{d_\ell \times d_{\ell-1}}$과 편향 벡터 $b^{(\ell)} \in \mathbb{R}^{d_\ell}$이 주어졌을 때,

$$z^{(\ell)} = \mathbf{W}^{(\ell)} h^{(\ell-1)} + b^{(\ell)}, \qquad h^{(\ell)} = \sigma\big(z^{(\ell)}\big), \qquad \ell = 1, \dots, L-1$$

으로 ℓ번째 은닉층 $h^{(\ell)} \in \mathbb{R}^{d_\ell}$을 계산한다. 여기서 활성화함수 σ는 벡터에 성분별로 적용한다.

마지막 출력층은 활성화함수 없이 선형변환

$$y = \mathbf{W}^{(L)} h^{(L-1)} + b^{(L)} \in \mathbb{R}^{d_L}$$

만으로 최종 출력 y를 결정한다.

완전연결: ℓ번째 층의 i번째 노드가 $(\ell-1)$번째 층의 모든 노드와 연결

연결 여부는 $\mathbf{W}^{(\ell)}$의 성분이 0인지로 표현. 일반적으로 $w_{i1}^{(\ell)}, w_{i2}^{(\ell)}, \cdots, w_{id_{\ell-1}}^{(\ell)}$가 모두 0이 아닌 값으로 시작 $\to$ 모든 노드가 연결

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

정의 9.2 — 완전연결 신경망(fully connected neural network, FCNN)

요약: $L+1$개의 층을 가지는 d_0-d_1-$\cdots$-d_{L-1}-d_L FCNN $f:\mathbb{R}^{d_0}\to\mathbb{R}^{d_L}$는 다음의 구조로 구성

$$\begin{aligned} h^{(0)} &= x \in \mathbb{R}^{d_0} \\ h^{(\ell)} &= \sigma\big(\mathbf{W}^{(\ell)} h^{(\ell-1)} + b^{(\ell)}\big) \in \mathbb{R}^{d_\ell}, \qquad \ell = 1,\dots,L-1 \\ y &= \mathbf{W}^{(L)} h^{(L-1)} + b^{(L)} \in \mathbb{R}^{d_L} \end{aligned}$$

이때 데이터로부터 정해야 하는 가중치 행렬과 편향 벡터를 모두 모은

$$\theta := \big(\mathbf{W}^{(1)}, b^{(1)}, \dots, \mathbf{W}^{(L)}, b^{(L)}\big)$$

를 이 신경망의 파라미터라 하고, 그 성분의 개수는 $m := \sum_{\ell=1}^{L}\big(d_\ell d_{\ell-1} + d_\ell\big)$이다.

가중치 행렬과 편향 벡터의 성분에 아무런 제약을 두지 않으므로 파라미터 공간(정의 1.1)은 $\Theta = \mathbb{R}^m$이며, 파라미터를 명시할 필요가 있을 때에는 이 신경망을 $f(x;\theta)$로 쓴다.

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

가중치 첨자와 파라미터 개수

- $\mathbf{W}^{(\ell)}$의 (j,k) 성분 $w_{jk}^{(\ell)}$ = $(\ell-1)$번째 층의 k번째 입력노드에서 ℓ번째 층의 j번째 출력노드로 가는 가중치

$$z_j^{(\ell)} = \sum_k w_{jk}^{(\ell)}\, h_k^{(\ell-1)} + b_j^{(\ell)}$$

- 첫 인덱스는 도착 노드, 둘째는 출발 노드
- 층(L)이 늘어나면, 총 가중치 개수 m이 폭발적으로 증가

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

예제 9.1 — 2-3-2-1 신경망의 각 성분

입력 $x \in \mathbb{R}^2$, 2개 은닉층 (각 노드 $3, 2$개), 출력 스칼라 $y \in \mathbb{R}$

$$\mathbf{W}^{(1)} \in \mathbb{R}^{3\times 2},\ \mathbf{W}^{(2)} \in \mathbb{R}^{2\times 3},\ \mathbf{W}^{(3)} \in \mathbb{R}^{1\times 2}, \qquad b^{(1)} \in \mathbb{R}^3,\ b^{(2)} \in \mathbb{R}^2,\ b^{(3)} \in \mathbb{R}$$

순전파: $h^{(1)} = \sigma(\mathbf{W}^{(1)}x + b^{(1)}) \in \mathbb{R}^3$, $h^{(2)} = \sigma(\mathbf{W}^{(2)}h^{(1)} + b^{(2)}) \in \mathbb{R}^2$,
$y = \mathbf{W}^{(3)}h^{(2)} + b^{(3)} \in \mathbb{R}$

- 파라미터: $\mathbf{W}^{(1)}, b^{(1)}$에서 $6+3=9$개, $\mathbf{W}^{(2)}, b^{(2)}$에서 $6+2=8$개, $\mathbf{W}^{(3)}, b^{(3)}$에서 $2+1=3$개, 합하여 $m=20$개

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

손실함수와 경사하강법

손실함수와 경사하강법

- 선형회귀처럼, 출력 변수 y의 표본별 계산값을 예측값 $\hat{y} = f(x;\boldsymbol{\theta})$이라 함
- 스칼라 출력 회귀에서 신경망 $f(x;\boldsymbol{\theta})$가 데이터 $\{(x_i, y_i)\}_{i=1}^{n}$을 얼마나 잘 설명하는지 — 6장의 제곱손실을 그대로 사용

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}\big(f(x_i;\boldsymbol{\theta}) - y_i\big)^2$$

손실을 가능한 한 작게 만드는 파라미터를 찾는 문제. 최솟값이 달성되는 경우에 한하여 $\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})$

- 7장 경사하강법: 초기값 $\boldsymbol{\theta}_0$, 반복 번호 $t = 0, 1, \ldots$, 학습률 $\eta > 0$에 대해 $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta\,\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}_t)$를 반복
- 문제: 층이 깊어질수록 $\nabla_{\boldsymbol{\theta}}\mathcal{L}$를 직접 미분하기가 매우 복잡
- 해결: 역전파(4장) = 연쇄법칙을 계산그래프 위에서 체계적으로 적용하는 절차

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

계산그래프와 역전파: 수치 예제

수치 예제 — 계산그래프 위에서의 역전파

- 역전파를 이해하는 가장 좋은 방법 = 아주 작은 신경망$(1\text{-}2\text{-}1)$에서 모든 수를 직접 따라가기
- 모든 파라미터를 1로 초기화

$$\boldsymbol{\theta} = (w_{11}^{(1)}, w_{21}^{(1)}, b_1^{(1)}, b_2^{(1)}, w_{11}^{(2)}, w_{12}^{(2)}, b^{(2)}) = (1,1,1,1,1,1,1), \qquad \sigma(z) = z^2$$

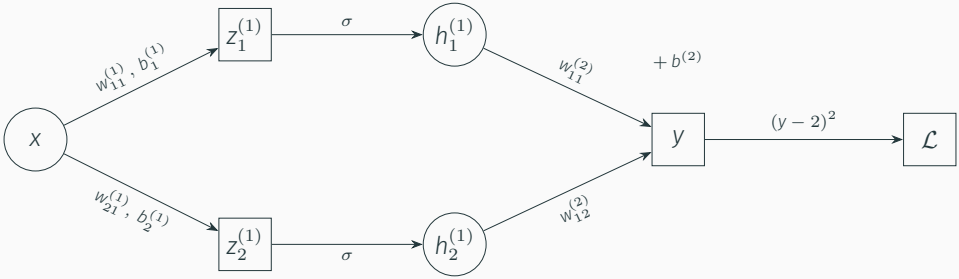
- 데이터는 $(x, y_{\text{data}}) = (1, 2)$ 한 쌍. 신경망의 출력 y와 구별하기 위해 정답에 아래첨자
- $\sigma(z) = z^2$은 실무용 활성화함수[1]는 아니지만 도함수가 $2z$로 단순해 직접 계산하기에 적합

[1] 마지막 절 함수족의 표현력에서 이유를 다룰 예정

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

1-2-1 FCNN의 계산그래프 설계

앞의 1-2-1 계산그래프 뒤에 제곱손실까지 이어 붙이면 준비 끝.



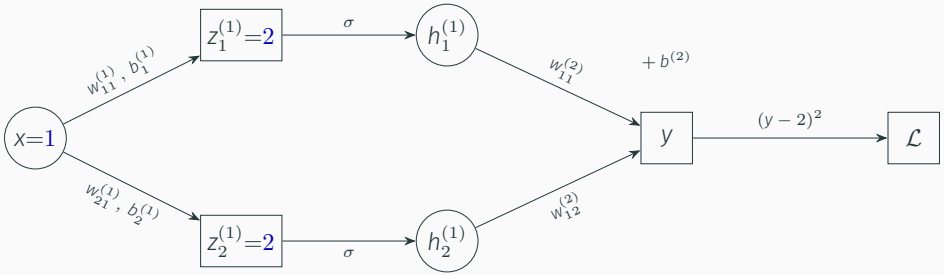
4장의 계산그래프 표현을 신경망에 그대로 적용

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

순전파 (1) — 선형결합

$(w_{11}^{(1)}, w_{21}^{(1)}, b_1^{(1)}, b_2^{(1)}, w_{11}^{(2)}, w_{12}^{(2)}, b^{(2)}) = (1, 1, 1, 1, 1, 1, 1)$

입력 $x = 1$에서 출발하여 그래프를 따라 오른쪽으로 값을 계산

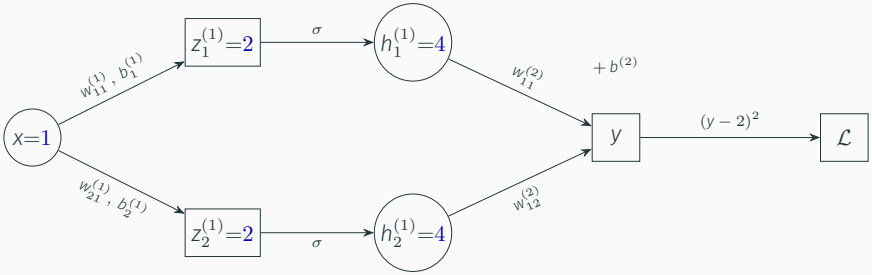


$$z_1^{(1)} = w_{11}^{(1)} x + b_1^{(1)} = 1 \cdot 1 + 1 = 2, \qquad z_2^{(1)} = w_{21}^{(1)} x + b_2^{(1)} = 1 \cdot 1 + 1 = 2$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

순전파 (2) — 활성화

$$(w_{11}^{(1)}, w_{21}^{(1)}, b_1^{(1)}, b_2^{(1)}, w_{11}^{(2)}, w_{12}^{(2)}, b^{(2)}) = (1,1,1,1,1,1,1)$$

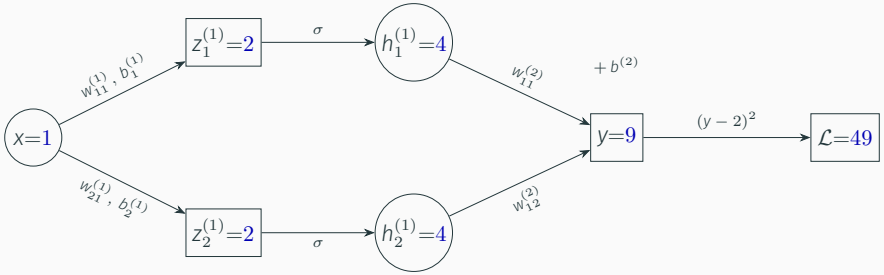


$$h_1^{(1)} = \sigma(z_1^{(1)}) = 2^2 = 4, \qquad h_2^{(1)} = \sigma(z_2^{(1)}) = 2^2 = 4$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

순전파 (3) — 출력과 손실

$$(w_{11}^{(1)}, w_{21}^{(1)}, b_1^{(1)}, b_2^{(1)}, w_{11}^{(2)}, w_{12}^{(2)}, b^{(2)}) = (1, 1, 1, 1, 1, 1, 1)$$



$$y = w_{11}^{(2)} h_1^{(1)} + w_{12}^{(2)} h_2^{(1)} + b^{(2)} = 1 \cdot 4 + 1 \cdot 4 + 1 = 9, \qquad \mathcal{L} = (y - y_{\text{data}})^2 = (9-2)^2 = 49$$

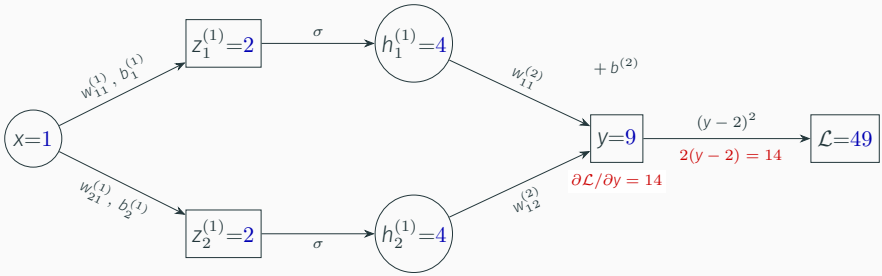
© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파 $\frac{\partial \mathcal{L}}{\partial w_{11}^{(1)}}$ (1) — 끝점 $\mathcal{L}$에서 출력으로: $\partial \mathcal{L}/\partial y$

$(w_{11}^{(1)}, w_{21}^{(1)}, b_1^{(1)}, b_2^{(1)}, w_{11}^{(2)}, w_{12}^{(2)}, b^{(2)}) = (1,1,1,1,1,1,1)$

연쇄법칙을 그래프의 역방향(오른쪽에서 왼쪽)으로 적용: $w_{11}^{(1)}$ 까지 가는 경로

$\mathcal{L} \to y \to h_1^{(1)} \to z_1^{(1)} \to w_{11}^{(1)}$



$$\frac{\partial \mathcal{L}}{\partial y} = 2(y-2) = 2(9-2) = 14$$

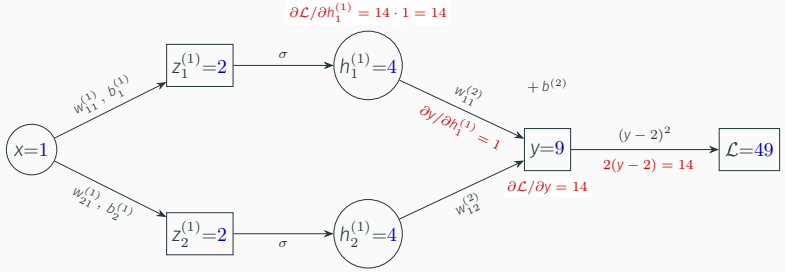
© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파 $\frac{\partial \mathcal{L}}{\partial w_{11}^{(1)}}$ (2) — 출력에서 은닉 노드로: $\partial y / \partial h_1^{(1)}$

$(w_{11}^{(1)}, w_{21}^{(1)}, b_1^{(1)}, b_2^{(1)}, w_{11}^{(2)}, w_{12}^{(2)}, b^{(2)}) = (1, 1, 1, 1, 1, 1, 1)$

연쇄법칙을 그래프의 역방향(오른쪽에서 왼쪽)으로 적용: $w_{11}^{(1)}$ 까지 가는 경로

$\mathcal{L} \to y \to h_1^{(1)} \to z_1^{(1)} \to w_{11}^{(1)}$

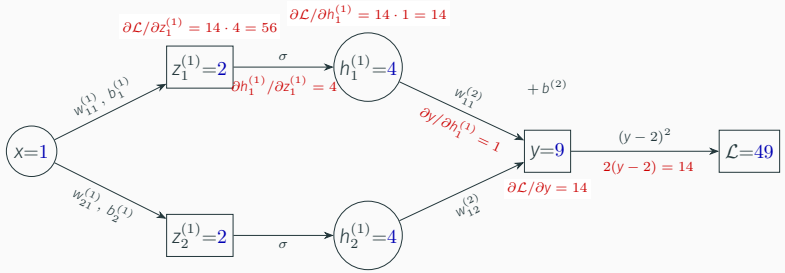


$$\frac{\partial y}{\partial h_1^{(1)}} = w_{11}^{(2)} = 1, \qquad \frac{\partial \mathcal{L}}{\partial h_1^{(1)}} = \frac{\partial \mathcal{L}}{\partial y} \cdot \frac{\partial y}{\partial h_1^{(1)}} = 14 \cdot 1 = 14$$

$h_2^{(1)}$를 지나는 경로는 $w_{11}^{(1)}$을 포함하지 않으므로 이 계산에서는 무시

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

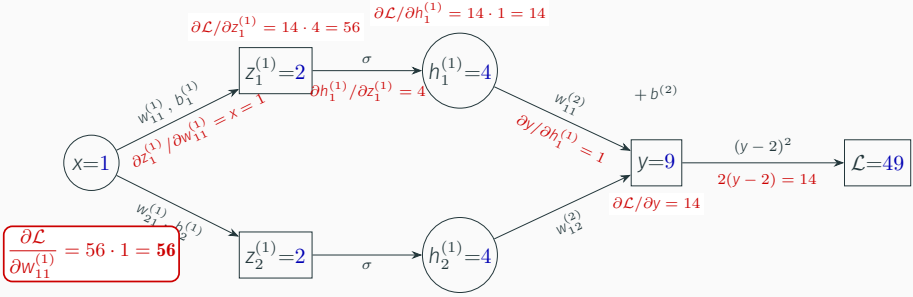
역전파 $\frac{\partial \mathcal{L}}{\partial w_{11}^{(1)}}$ (3) — 은닉 노드에서 선형결합으로: 활성화함수의 미분 σ'



$$\frac{\partial h_1^{(1)}}{\partial z_1^{(1)}} = \sigma'(z_1^{(1)}) = 2z_1^{(1)} = 4, \qquad \frac{\partial \mathcal{L}}{\partial z_1^{(1)}} = \frac{\partial \mathcal{L}}{\partial h_1^{(1)}} \cdot \frac{\partial h_1^{(1)}}{\partial z_1^{(1)}} = 14 \cdot 4 = 56$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파 $\frac{\partial \mathcal{L}}{\partial w_{11}^{(1)}}$ (4) — 파라미터에 도달: $\partial z_1^{(1)}/\partial w_{11}^{(1)}$



$$\frac{\partial \mathcal{L}}{\partial w_{11}^{(1)}} = \underbrace{14}_{\partial \mathcal{L}/\partial y} \times \underbrace{1}_{\partial y/\partial h_1^{(1)}} \times \underbrace{4}_{\partial h_1^{(1)}/\partial z_1^{(1)}} \times \underbrace{1}_{\partial z_1^{(1)}/\partial w_{11}^{(1)}} = \mathbf{56}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

결론 — 갱신 규칙과 전략

- 1-2-1 신경망, $\boldsymbol{\theta} = (1,1,1,1,1,1,1)$, $\sigma(z) = z^2$, $(x, y_{\text{data}}) = (1,2)$에서 $\partial \mathcal{L}/\partial w_{11}^{(1)} = 56$
- 학습률 η의 경사하강법 갱신: $w_{11}^{(1),\text{new}} = w_{11}^{(1),\text{old}} - \eta \cdot 56$
- 다른 파라미터 $w_{21}^{(1)}, b_1^{(1)}, b_2^{(1)}, w_{11}^{(2)}, w_{12}^{(2)}, b^{(2)}$도 같은 방식으로 갱신

주목할 점은 전략 자체

$\mathcal{L}$의 식에 $w_{11}^{(1)}$을 대입해 한꺼번에 미분하면 복잡한 합성함수. 계산그래프에서는 각 노드가 직접 이웃에 대한 국소 미분만 알면 되고, 경로를 따라 곱하고 경로가 여럿이면 더하기만 하면 됨.

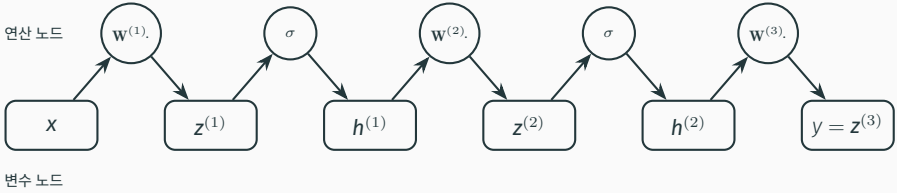
한 개의 파라미터가 아닌, 여러 개 파라미터의 미분값을 동시에 유도할 수 있는가?

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파의 일반 공식

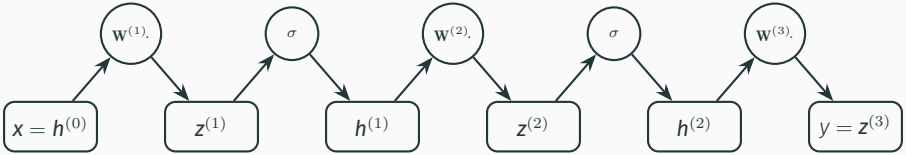
목표 — $\partial y/\partial \mathbf{W}^{(\ell)}$의 일반 공식?

목표: 2-3-2-1 FCNN의 벡터 단위 계산그래프에서 각 $\mathbf{W}^{(\ell)}$에 대한 출력 y의 그래디언트 $\partial y/\partial \mathbf{W}^{(\ell)}$ 구하기 (편의상 편향(bias)은 생략)



© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

성분마다 연쇄법칙을 새로 시작하면?

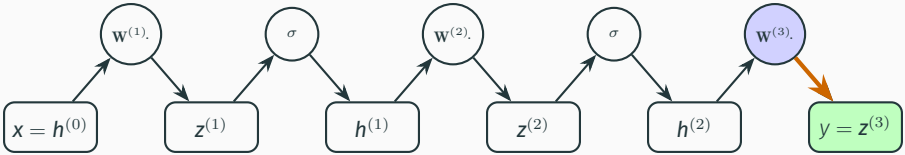


- 모두 스칼라인 1-1-1-1 신경망 $y = w^{(3)}\sigma(w^{(2)}\sigma(w^{(1)}x))$ 에서는 연쇄법칙 한 번으로 끝남
  - 예: $\frac{\partial y}{\partial w^{(1)}} = \frac{\partial y}{\partial h^{(2)}}\frac{\partial h^{(2)}}{\partial z^{(2)}}\frac{\partial z^{(2)}}{\partial h^{(1)}}\frac{\partial h^{(1)}}{\partial z^{(1)}}\frac{\partial z^{(1)}}{\partial w^{(1)}}$
- 모든 파라미터에 대해 일일이 할 수는 없음. 너무 많다.
- 한 층에 노드가 여러개라면, 미분 대상을 행렬 $\mathbf{W}^{(\ell)}$ 로 바라보자

$\partial y/\partial \mathbf{W}^{(\ell)}$: 성분이 $\partial y/\partial w_{jk}^{(\ell)}$ 인 $d_\ell \times d_{\ell-1}$ 행렬

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

출력층 직전 파라미터 $\mathbf{W}^{(3)}$에 대한 그래디언트 계산은 쉽다



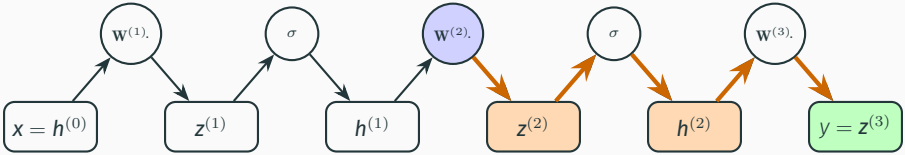
- $\partial y/\partial \mathbf{W}^{(\ell)}$: 성분이 $\partial y/\partial w_{jk}^{(\ell)}$인 $d_\ell \times d_{\ell-1}$ 행렬
- $\mathbf{W}^{(3)}$에서 y까지는 선형변환 하나뿐

$$y = \mathbf{W}^{(3)}\boldsymbol{h}^{(2)} = w_{11}^{(3)}h_1^{(2)} + w_{12}^{(3)}h_2^{(2)} \quad \Rightarrow \quad \frac{\partial y}{\partial \mathbf{W}^{(3)}} = \begin{pmatrix} \frac{\partial y}{\partial w_{11}^{(3)}} & \frac{\partial y}{\partial w_{12}^{(3)}} \end{pmatrix} = \begin{pmatrix} h_1^{(2)} & h_2^{(2)} \end{pmatrix} = \boldsymbol{h}^{(2)\top}$$

- $\boldsymbol{h}^{(2)\top}$는 순전파 단계에서 이미 계산한 값

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

$\mathbf{W}^{(2)}$ 부터는 많은 스텝을 거친다



- $\mathbf{W}^{(2)}$에서 y까지는 변수 $z^{(2)}$, $h^{(2)}$를 통과함: $z^{(2)} = \mathbf{W}^{(2)} h^{(1)}$
- 연쇄법칙을 중간 노드 $z^{(2)}$에서 적용

$$\frac{\partial y}{\partial \mathbf{W}^{(2)}} = \underbrace{\frac{\partial y}{\partial z^{(2)}}}_{?} \cdot \underbrace{\frac{\partial z^{(2)}}{\partial \mathbf{W}^{(2)}}}_{h^{(1)\top}} \quad \Rightarrow \quad \frac{\partial y}{\partial \mathbf{W}^{(2)}} = \frac{\partial y}{\partial z^{(2)}} h^{(1)\top}$$

- $h^{(1)\top}$는 순전파에서 이미 아는 값
- 남은 작업은 $\partial y / \partial z^{(2)}$을 알고 있는 값들로 계산하기.

$$\frac{\partial y}{\partial z^{(2)}} = \text{오차 신호 } \boldsymbol{\delta}^{(2)}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

정의 9.3 — 오차 신호(error signal), 델타(delta)

ℓ번째 층의 선형결합 벡터 $z^{(\ell)} \in \mathbb{R}^{d_\ell}$에 대하여, **오차 신호**(error signal) 또는 **델타**(delta) $\boldsymbol{\delta}^{(\ell)} \in \mathbb{R}^{d_\ell}$를

$$\boldsymbol{\delta}^{(\ell)} := \frac{\partial y}{\partial z^{(\ell)}} = \begin{pmatrix} \partial y/\partial z_1^{(\ell)} \\ \vdots \\ \partial y/\partial z_{d_\ell}^{(\ell)} \end{pmatrix}$$

로 정의한다.

- 출력층에서는 $y = z^{(L)}$이므로 자명하게 $\boldsymbol{\delta}^{(L)} = \partial y/\partial z^{(L)} = 1$
- 스칼라 y를 벡터 $z^{(\ell)}$로 미분한 그래디언트 $\to z^{(\ell)}$와 같은 $d_\ell \times 1$ 열벡터
- 내용 정리: 앞 페이지의 식에서 $\partial y/\partial z^{(\ell)}$를 $\boldsymbol{\delta}^{(\ell)}$로 바꾸면

$$\frac{\partial y}{\partial \mathbf{W}^{(\ell)}} = \frac{\partial y}{\partial z^{(\ell)}} \frac{\partial z^{(\ell)}}{\partial \mathbf{W}^{(\ell)}} = \boldsymbol{\delta}^{(\ell)} \, h^{(\ell-1)\top}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

정의 9.4 — 활성화 함수의 미분으로 구성된 대각행렬

ℓ번째 층의 선형결합 벡터 $\boldsymbol{z}^{(\ell)} \in \mathbb{R}^{d_\ell}$에 대하여

$$\mathbf{D}^{(\ell)} := \operatorname{diag}\big(\sigma'(\boldsymbol{z}^{(\ell)})\big) \in \mathbb{R}^{d_\ell \times d_\ell}$$

는 $\sigma'(\boldsymbol{z}^{(\ell)})$의 성분들을 대각선에 배치한 대각행렬이다.

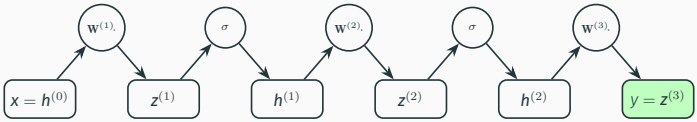
- 예: $\sigma(z) = z^2$, $\boldsymbol{z}^{(2)} = (1,\,-1,\,2)^\top \in \mathbb{R}^3$이면 $\sigma'(\boldsymbol{z}^{(2)}) = (2,\,-2,\,4)^\top$이고

$$\mathbf{D}^{(2)} = \operatorname{diag}(2, -2, 4) = \begin{pmatrix} 2 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 4 \end{pmatrix} \in \mathbb{R}^{3\times 3}$$

목표: $\boldsymbol{\delta}^{(\ell)} = \mathbf{D}^{(\ell)} {\mathbf{W}^{(\ell+1)}}^\top \boldsymbol{\delta}^{(\ell+1)}$ 유도

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

$\boldsymbol{\delta}^{(\ell)} = \mathbf{D}^{(\ell)}\mathbf{W}^{(\ell+1)\top}\boldsymbol{\delta}^{(\ell+1)}$를 알면, 모든 $\mathbf{W}$에 대한 미분을 한 번에 계산 가능!



핵심이 되는 두 관계식 ($\boldsymbol{\delta}^{(L)} = 1$, 여기서 $L = 3$):

$$\text{(1)}\ \frac{\partial y}{\partial \mathbf{W}^{(\ell)}} = \boldsymbol{\delta}^{(\ell)}\, h^{(\ell-1)\top} \qquad \text{(2)}\ \boldsymbol{\delta}^{(\ell)} = \mathbf{D}^{(\ell)}\mathbf{W}^{(\ell+1)\top}\boldsymbol{\delta}^{(\ell+1)}$$

$$\frac{\partial y}{\partial \mathbf{W}^{(3)}} \underset{(1)}{=} \boldsymbol{\delta}^{(3)}\, h^{(2)\top} = h^{(2)\top}$$

$$\frac{\partial y}{\partial \mathbf{W}^{(2)}} \underset{(1)}{=} \boldsymbol{\delta}^{(2)}\, h^{(1)\top} \underset{(2)}{=} \mathbf{D}^{(2)}\mathbf{W}^{(3)\top}\boldsymbol{\delta}^{(3)}\, h^{(1)\top} = \mathbf{D}^{(2)}\mathbf{W}^{(3)\top} h^{(1)\top}$$

$$\frac{\partial y}{\partial \mathbf{W}^{(1)}} \underset{(1)}{=} \boldsymbol{\delta}^{(1)}\, h^{(0)\top} \underset{(2)}{=} \mathbf{D}^{(1)}\mathbf{W}^{(2)\top}\boldsymbol{\delta}^{(2)}\, h^{(0)\top} \underset{(2)}{=} \mathbf{D}^{(1)}\mathbf{W}^{(2)\top}\mathbf{D}^{(2)}\mathbf{W}^{(3)\top} h^{(0)\top}$$

한 층 앞으로 갈 때마다 $\boldsymbol{\delta}$에 $\mathbf{D}\mathbf{W}^{\top}$가 하나씩 곱해지는 패턴. 이 식이 시사하는 바?

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

y에서 $\mathcal{L}$로 — 손실의 모양은 $\partial\mathcal{L}/\partial y$ 하나만 추가

- 앞 공식은 출력 y 기준의 미분. 학습에 필요한 것은 손실 $\mathcal{L}$ 기준의 그래디언트
- y가 스칼라이므로 연쇄법칙은 상수배 하나

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}^{(\ell)}} = \underbrace{\frac{\partial \mathcal{L}}{\partial y}}_{=:g\ \text{쉬움}} \cdot \underbrace{\frac{\partial y}{\partial \mathbf{W}^{(\ell)}}}_{\text{앞 공식}}, \qquad \ell = 1, 2, 3$$

- 손실함수의 $\mathbf{W}$ 미분은 스칼라 g가 곱해짐 $\rightarrow$ 보통 계산이 쉬움
  - 예: 제곱손실 $\mathcal{L} = (y - y_{\text{data}})^2 \rightarrow g = 2(y - y_{\text{data}})$
- 경사하강법 갱신 규칙(학습률 η): 세 가중치 행렬 모두 같은 g를 공유

$$\mathbf{W}^{(3)} \leftarrow \mathbf{W}^{(3)} - \eta\, g\, \boldsymbol{h}^{(2)\top}$$

$$\mathbf{W}^{(2)} \leftarrow \mathbf{W}^{(2)} - \eta\, g\, \mathbf{D}^{(2)} \mathbf{W}^{(3)\top} \boldsymbol{h}^{(1)\top}$$

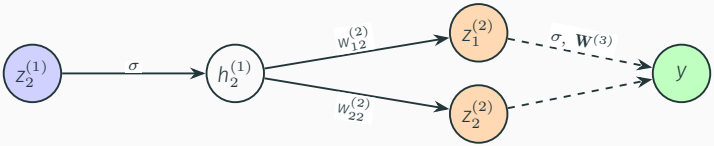
$$\mathbf{W}^{(1)} \leftarrow \mathbf{W}^{(1)} - \eta\, g\, \mathbf{D}^{(1)} \mathbf{W}^{(2)\top} \mathbf{D}^{(2)} \mathbf{W}^{(3)\top} \boldsymbol{h}^{(0)\top}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

전달 공식 유도 (1) — 하나의 노드에서 그래디언트

$z^{(1)}$ $\quad h^{(1)} = \sigma(z^{(1)})$ $\quad z^{(2)} = \mathbf{W}^{(2)} h^{(1)}$ $\quad y$

$\boldsymbol{\delta}^{(1)}$: 이제 구할 그래디언트 $\quad$ $\boldsymbol{\delta}^{(2)}$: 이미 구한 그래디언트



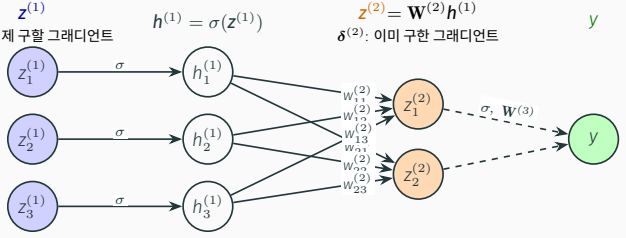
$z_2^{(1)}$에서 y로 가는 경로는 둘 ($z_1^{(2)}$ 혹은 $z_2^{(2)}$ 경유). 다변수 연쇄법칙(3장): 경로별 그래디언트 곱의 합

$$\delta_2^{(1)} = \frac{\partial y}{\partial z_2^{(1)}} = \underbrace{\frac{\partial y}{\partial z_1^{(2)}} \frac{\partial z_1^{(2)}}{\partial z_2^{(1)}}}_{\text{경로 } z_1^{(2)}} + \underbrace{\frac{\partial y}{\partial z_2^{(2)}} \frac{\partial z_2^{(2)}}{\partial z_2^{(1)}}}_{\text{경로 } z_2^{(2)}} = \delta_1^{(2)} \frac{\partial z_1^{(2)}}{\partial z_2^{(1)}} + \delta_2^{(2)} \frac{\partial z_2^{(2)}}{\partial z_2^{(1)}}$$

- 같은 계산을 $z_1^{(1)}, z_3^{(1)}$에 대해서도 하면 식이 셋 $\Rightarrow$ 다음 페이지

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

전달 공식 유도 (1) — $\boldsymbol{\delta}^{(1)} = \left[\dfrac{\partial \mathbf{z}^{(2)}}{\partial \mathbf{z}^{(1)}}\right]^{\top} \boldsymbol{\delta}^{(2)}$



$\boldsymbol{\delta}^{(1)} = \dfrac{\partial y}{\partial \mathbf{z}^{(1)}}$ 과 $\boldsymbol{\delta}^{(2)} = \dfrac{\partial y}{\partial \mathbf{z}^{(2)}}$ 의 관계: 앞 페이지의 계산을 $z_1^{(1)}, z_2^{(1)}, z_3^{(1)}$ 모두에 적용

$$\boldsymbol{\delta}^{(1)} \begin{cases} \delta_1^{(1)} = \dfrac{\partial y}{\partial z_1^{(1)}} = \delta_1^{(2)} \dfrac{\partial z_1^{(2)}}{\partial z_1^{(1)}} + \delta_2^{(2)} \dfrac{\partial z_2^{(2)}}{\partial z_1^{(1)}} \\ \delta_2^{(1)} = \dfrac{\partial y}{\partial z_2^{(1)}} = \delta_1^{(2)} \dfrac{\partial z_1^{(2)}}{\partial z_2^{(1)}} + \delta_2^{(2)} \dfrac{\partial z_2^{(2)}}{\partial z_2^{(1)}} \\ \delta_3^{(1)} = \dfrac{\partial y}{\partial z_3^{(1)}} = \delta_1^{(2)} \dfrac{\partial z_1^{(2)}}{\partial z_3^{(1)}} + \delta_2^{(2)} \dfrac{\partial z_2^{(2)}}{\partial z_3^{(1)}} \end{cases} = \begin{pmatrix} \dfrac{\partial z_1^{(2)}}{\partial z_1^{(1)}} & \dfrac{\partial z_2^{(2)}}{\partial z_1^{(1)}} \\ \dfrac{\partial z_1^{(2)}}{\partial z_2^{(1)}} & \dfrac{\partial z_2^{(2)}}{\partial z_2^{(1)}} \\ \dfrac{\partial z_1^{(2)}}{\partial z_3^{(1)}} & \dfrac{\partial z_2^{(2)}}{\partial z_3^{(1)}} \end{pmatrix} \begin{pmatrix} \delta_1^{(2)} \\ \delta_2^{(2)} \end{pmatrix} = \left[\frac{\partial \mathbf{z}^{(2)}}{\partial \mathbf{z}^{(1)}}\right]^{\top} \boldsymbol{\delta}^{(2)}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

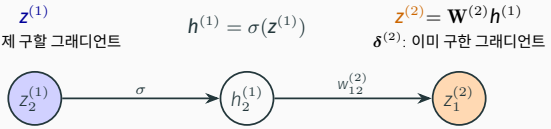
전달 공식 유도 (1) — $\frac{\partial \mathbf{z}^{(2)}}{\partial \mathbf{z}^{(1)}} = ?$

$$\left[\frac{\partial \mathbf{z}^{(2)}}{\partial \mathbf{z}^{(1)}}\right]^\top = \begin{pmatrix} \frac{\partial z_1^{(2)}}{\partial z_1^{(1)}} & \frac{\partial z_2^{(2)}}{\partial z_1^{(1)}} \\ \frac{\partial z_1^{(2)}}{\partial z_2^{(1)}} & \frac{\partial z_2^{(2)}}{\partial z_2^{(1)}} \\ \frac{\partial z_1^{(2)}}{\partial z_3^{(1)}} & \frac{\partial z_2^{(2)}}{\partial z_3^{(1)}} \end{pmatrix}$$

- $\boldsymbol{\delta}^{(1)} = \left[\frac{\partial \mathbf{z}^{(2)}}{\partial \mathbf{z}^{(1)}}\right]^\top \boldsymbol{\delta}^{(2)}$ 자체는 연쇄법칙 (cf: $\nabla(y \circ \boldsymbol{g}) = \mathbf{J}_{\boldsymbol{g}}^\top \nabla y$)
- 남은 과정: $\left[\frac{\partial \mathbf{z}^{(2)}}{\partial \mathbf{z}^{(1)}}\right]^\top$ 의 성분 $\partial z_j^{(2)} / \partial z_k^{(1)}$ 를 우리가 아는 변수 $\mathbf{W}^{(2)}, \boldsymbol{z}^{(1)}$ 로 표현하기

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

전달 공식 유도 (2) — $\frac{\partial z^{(2)}}{\partial z^{(1)}}$ 를 알고 있는 값으로 표현



$z_k^{(1)}$가 $z_j^{(2)}$에 닿는 경로는 σ 다음 $w_{jk}^{(2)}$ 하나뿐 (그림: $k=2, j=1$):

$$z_j^{(2)} = \sum_{k=1}^{3} w_{jk}^{(2)}\, \sigma(z_k^{(1)}) \quad \Rightarrow \quad \frac{\partial z_j^{(2)}}{\partial z_k^{(1)}} = w_{jk}^{(2)}\, \sigma'(z_k^{(1)})$$

모든 (j,k)를 행렬로 모으면

$$\frac{\partial \mathbf{z}^{(2)}}{\partial \mathbf{z}^{(1)}} = \begin{pmatrix} w_{11}^{(2)}\sigma'(z_1^{(1)}) & w_{12}^{(2)}\sigma'(z_2^{(1)}) & w_{13}^{(2)}\sigma'(z_3^{(1)}) \\ w_{21}^{(2)}\sigma'(z_1^{(1)}) & w_{22}^{(2)}\sigma'(z_2^{(1)}) & w_{23}^{(2)}\sigma'(z_3^{(1)}) \end{pmatrix} = \mathbf{W}^{(2)}\,\mathbf{D}^{(1)}$$

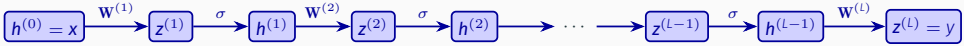
- $\frac{\partial \mathbf{z}^{(2)}}{\partial \mathbf{z}^{(1)}} = \mathbf{W}^{(2)}\,\mathbf{D}^{(1)}$ 이므로, $\boldsymbol{\delta}^{(1)} = \mathbf{D}^{(1)}\,{\mathbf{W}^{(2)}}^{\top}\,\boldsymbol{\delta}^{(2)}$
- 일반적인 $\ell, \ell+1$ 사이에서: $\boldsymbol{\delta}^{(\ell)} = \mathbf{D}^{(\ell)}\,{\mathbf{W}^{(\ell+1)}}^{\top}\,\boldsymbol{\delta}^{(\ell+1)}$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

정리 9.1 — 역전파 계산 과정 전체

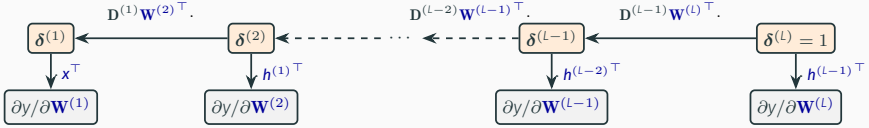
d_0-d_1-$\cdots$-d_{L-1}-d_L 구조의 FCNN이 다음을 만족한다고 하자 (편의상 $d_L = 1$, 편향 생략).

1) 순전파로 $\boldsymbol{h}^{(\ell)}, \boldsymbol{z}^{(\ell)}$를 계산 ($\mathbf{W}^{(\ell)}$와 함께 역전파에서 값을 바로 사용 가능)



2) $\ell = 1, \ldots, L$에 대해 $\boldsymbol{\delta}^{(\ell)} := \dfrac{\partial y}{\partial \boldsymbol{z}^{(\ell)}} \in \mathbb{R}^{d_\ell}$, $\mathbf{D}^{(\ell)} := \operatorname{diag}\big(\sigma'(\boldsymbol{z}^{(\ell)})\big) \in \mathbb{R}^{d_\ell \times d_\ell}$ 라 하면

$$
\begin{aligned}
\boldsymbol{\delta}^{(L)} &= 1, \\
\frac{\partial y}{\partial \mathbf{W}^{(\ell)}} &= \boldsymbol{\delta}^{(\ell)}\, \boldsymbol{h}^{(\ell-1)\top}, && \ell = L, L-1, \ldots, 1, \\
\boldsymbol{\delta}^{(\ell)} &= \mathbf{D}^{(\ell)}\, \mathbf{W}^{(\ell+1)\top}\, \boldsymbol{\delta}^{(\ell+1)}, && \ell = L-1, \ldots, 1.
\end{aligned}
$$



© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

정리 9.1의 확장 — 네 가지 추가 질문

Q1) **편향 $b^{(\ell)}$의 공식도 있는가?** 있음. $z^{(\ell)} = \mathbf{W}^{(\ell)} h^{(\ell-1)} + b^{(\ell)}$에서 $\partial z_j^{(\ell)} / \partial b_j^{(\ell)} = 1$이므로

$$\frac{\partial y}{\partial b^{(\ell)}} = \boldsymbol{\delta}^{(\ell)}$$

Q2) **출력 노드가 여러 개($d_L > 1$)이면?** 미분 대상은 스칼라 손실 $\mathcal{L}$. 출발점만

$$\boldsymbol{\delta}^{(L)} = \frac{\partial \mathcal{L}}{\partial \mathbf{z}^{(L)}} \in \mathbb{R}^{d_L}$$

로 바뀌고, 두 공식은 y를 $\mathcal{L}$로 바꾼 채 그대로 (예: 제곱손실 $\frac{1}{2}\|\mathbf{z}^{(L)} - \mathbf{y}^*\|^2$이면 $\boldsymbol{\delta}^{(L)} = \mathbf{z}^{(L)} - \mathbf{y}^*$)

Q3) **은닉 노드마다 활성화함수가 다르면 ($h_k^{(\ell)} = \sigma_k(z_k^{(\ell)})$)?** $\sigma'(z_k^{(\ell)})$가 $\sigma_k'(z_k^{(\ell)})$로 바뀔 뿐

$$\mathbf{D}^{(\ell)} = \operatorname{diag}\big(\sigma_1'(z_1^{(\ell)}), \ldots, \sigma_{d_\ell}'(z_{d_\ell}^{(\ell)})\big)$$

앞의 정리에 공식 형태는 그대로

정리 9.1의 확장 — 네 가지 추가 질문 (Q4)

Q4) **표본이 n개 (x_i, y_i^*)이면?** 정리 9.1은 입력 하나에 대한 $\partial y/\partial \mathbf{W}^{(\ell)}$. 학습에 쓰는 손실은 표본별 손실의 평균

$$\mathcal{L}(\mathbf{W}) = \frac{1}{n}\sum_{i=1}^{n} l_i, \qquad l_i = \big(y(x_i) - y_i^*\big)^2$$

이므로 그래디언트도 평균: 표본마다 순전파·역전파를 하고 "y에서 $\mathcal{L}$로" 인자 $2(y(x_i)-y_i^*)$를 곱해 더함

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}^{(\ell)}} = \frac{1}{n}\sum_{i=1}^{n} 2\big(y(x_i)-y_i^*\big)\frac{\partial y(x_i)}{\partial \mathbf{W}^{(\ell)}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{\delta}_i^{(\ell)}\, \boldsymbol{h}_i^{(\ell-1)\top}$$

($\boldsymbol{\delta}_i^{(\ell)}$, $\boldsymbol{h}_i^{(\ell-1)}$: 표본 i의 오차 신호와 은닉값, Q2처럼 $\delta_i^{(L)} = \partial l_i/\partial z^{(L)} = 2(y(x_i)-y_i^*)$에서 출발)

- $\mathbf{Z}^{(\ell)} = [\boldsymbol{z}_1^{(\ell)} \cdots \boldsymbol{z}_n^{(\ell)}]$, $\mathbf{H}^{(\ell-1)} = [\boldsymbol{h}_1^{(\ell-1)} \cdots \boldsymbol{h}_n^{(\ell-1)}] \in \mathbb{R}^{d_{\ell-1}\times n}$, $\boldsymbol{\Delta}^{(\ell)} = [\boldsymbol{\delta}_1^{(\ell)} \cdots \boldsymbol{\delta}_n^{(\ell)}] \in \mathbb{R}^{d_\ell \times n}$에 대해

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}^{(\ell)}} = \frac{1}{n}\,\boldsymbol{\Delta}^{(\ell)}\,\mathbf{H}^{(\ell-1)\top}, \qquad \boldsymbol{\Delta}^{(\ell)} = \sigma'(\mathbf{Z}^{(\ell)}) \odot \big(\mathbf{W}^{(\ell+1)\top}\boldsymbol{\Delta}^{(\ell+1)}\big)$$

표본 n개의 순전파·역전파 $\rightarrow$ 열 방향 배열 재구성 뒤 행렬 곱 하나 (병렬계산 관점)

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

왜 $\partial y/\partial w_{jk}$를 하나씩 계산하지 않는가: 병렬계산의 관점

성분마다 따로 계산해도 틀리지는 않음. 역전파를 행렬 단위로 쓰면 다음 네 가지 관점에서 효율적!

(1) **연산 횟수.** 성분 $w_{jk}^{(\ell)}$ 마다 출력에서부터 연쇄법칙을 새로 시작하면 동일한 연산을 반복. 오차 신호 $\boldsymbol{\delta}^{(\ell)}$를 한 번 구해 두면 그 층의 모든 성분이 공유하고, 아래층($\ell-1$층)의 오차 신호 계산에도 그대로 사용

(2) **순전파 값 재사용.** $\boldsymbol{\delta}^{(\ell)} = \mathbf{D}^{(\ell)}\mathbf{W}^{(\ell+1)\top}\boldsymbol{\delta}^{(\ell+1)}$, $\partial y/\partial \mathbf{W}^{(\ell)} = \boldsymbol{\delta}^{(\ell)}\boldsymbol{h}^{(\ell-1)\top}$ 에 들어가는 $\mathbf{W}, \boldsymbol{h}, \boldsymbol{z}$는 순전파에서 이미 계산(저장)한 값 (새로 하는 미분은 $\mathbf{D}^{(\ell)} = \operatorname{diag}(\sigma'(\boldsymbol{z}^{(\ell)}))$ 하나뿐)

(3) **메모리.** $\partial y/\partial \mathbf{W}^{(\ell)}$는 $d_\ell \times d_{\ell-1}$ 행렬이지만 외적 $\boldsymbol{\delta}^{(\ell)}\boldsymbol{h}^{(\ell-1)\top}$ 이므로, 갱신 전까지는 벡터 둘($d_\ell + d_{\ell-1}$개)만 저장하고 $\mathbf{W}^{(\ell)} \leftarrow \mathbf{W}^{(\ell)} - \eta\,\boldsymbol{\delta}^{(\ell)}\boldsymbol{h}^{(\ell-1)\top}$ 로 바로 갱신

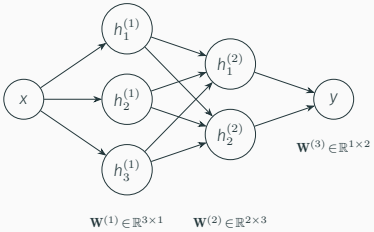
(4) **구현.** 층별 계산이 모두 벡터·행렬 연산이므로 GPU 같은 병렬 하드웨어에 바로 적용 가능. PyTorch·TensorFlow의 자동미분도 같은 구조

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파 계산 예시와 NumPy 구현: 1-3-2-1 신경망

1-3-2-1 신경망 — 설정

1-3-2-1 신경망에서 정리 9.1을 적용하여 $\partial y/\partial \mathbf{W}^{(\ell)}$ 을 직접 계산 (편향 없음)



- 계산 편의상 $\sigma(z) = z^2$ ($\sigma'(z) = 2z$)
- 입력 $x = 1$ ($\boldsymbol{h}^{(0)} = x$), 정답 $y_{\text{data}} = 1$
- 손실 $l = (y - y_{\text{data}})^2$
- $\partial l/\partial \mathbf{W} = 2(y - y_{\text{data}})\,\partial y/\partial \mathbf{W}$

$$\mathbf{W}^{(1)} = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}, \qquad \mathbf{W}^{(2)} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -2 \end{pmatrix}, \qquad \mathbf{W}^{(3)} = \begin{pmatrix} 1 & -1 \end{pmatrix}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파 계산 예시 (1) — 순전파

$$\mathbf{W}^{(1)} = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}, \quad \mathbf{W}^{(2)} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -2 \end{pmatrix}, \quad \mathbf{W}^{(3)} = \begin{pmatrix} 1 & -1 \end{pmatrix}, \quad h^{(0)} = x = 1, \quad \sigma(z) = z^2$$

$h^{(0)} = x = 1$이므로

$$z^{(1)} = \mathbf{W}^{(1)} x = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}, \qquad h^{(1)} = \sigma(z^{(1)}) = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$z^{(2)} = \mathbf{W}^{(2)} h^{(1)} = \begin{pmatrix} 2 \\ -1 \end{pmatrix}, \qquad h^{(2)} = \sigma(z^{(2)}) = \begin{pmatrix} 4 \\ 1 \end{pmatrix}$$

$$y = z^{(3)} = \mathbf{W}^{(3)} h^{(2)} = 4 - 1 = 3$$

모든 변수들을 어딘가에 적어두기...

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파 계산 예시 (2) — 역전파: 출력층과 두 번째 은닉층

$$\mathbf{W}^{(1)} = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}, \quad \mathbf{W}^{(2)} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -2 \end{pmatrix}, \quad \mathbf{W}^{(3)} = \begin{pmatrix} 1 & -1 \end{pmatrix}, \quad \boldsymbol{h}^{(0)} = x = 1, \quad \sigma(z) = z^2, \sigma'(z) = 2z$$

$$\boldsymbol{z}^{(1)} = (1, 1, -1)^\top, \quad \boldsymbol{h}^{(1)} = (1, 1, 1)^\top, \quad \boldsymbol{z}^{(2)} = (2, -1)^\top, \quad \boldsymbol{h}^{(2)} = (4, 1)^\top, \quad y = \boldsymbol{z}^{(3)} = 3$$

정리 9.1의 두 공식을 번갈아 적용: 그 층의 가중치 그래디언트, 그다음 아래층 오차 신호

- 출력층 ($\delta^{(3)} = 1$)

$$\frac{\partial y}{\partial \mathbf{W}^{(3)}} = \delta^{(3)} \boldsymbol{h}^{(2)\top} = \begin{pmatrix} 4 & 1 \end{pmatrix}$$

- 두 번째 은닉층: $\mathbf{D}^{(2)} = \operatorname{diag}(2\boldsymbol{z}^{(2)}) = \operatorname{diag}(4, -2)$

$$\boldsymbol{\delta}^{(2)} = \mathbf{D}^{(2)} \mathbf{W}^{(3)\top} \delta^{(3)} = \begin{pmatrix} 4 & 0 \\ 0 & -2 \end{pmatrix} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \cdot 1 = \begin{pmatrix} 4 \\ 2 \end{pmatrix}$$

$$\frac{\partial y}{\partial \mathbf{W}^{(2)}} = \boldsymbol{\delta}^{(2)} \boldsymbol{h}^{(1)\top} = \begin{pmatrix} 4 \\ 2 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 4 & 4 & 4 \\ 2 & 2 & 2 \end{pmatrix}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파 계산 예시 (3) — 역전파: 첫 번째 은닉층

$$\mathbf{W}^{(1)} = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}, \quad \mathbf{W}^{(2)} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -2 \end{pmatrix}, \quad \mathbf{W}^{(3)} = \begin{pmatrix} 1 & -1 \end{pmatrix}, \quad h^{(0)} = x = 1, \quad \sigma(z) = z^2,\ \sigma'(z) = 2z$$

$$z^{(1)} = (1, 1, -1)^\top, \quad h^{(1)} = (1, 1, 1)^\top, \quad z^{(2)} = (2, -1)^\top, \quad h^{(2)} = (4, 1)^\top, \quad y = 3$$

$$\delta^{(3)} = 1, \quad \boldsymbol{\delta}^{(2)} = (4, 2)^\top$$

$$\mathbf{D}^{(1)} = \operatorname{diag}(2z^{(1)}) = \operatorname{diag}(2, 2, -2), \quad {\mathbf{W}^{(2)}}^\top \boldsymbol{\delta}^{(2)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & -2 \end{pmatrix} \begin{pmatrix} 4 \\ 2 \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \\ 0 \end{pmatrix}$$

$$\boldsymbol{\delta}^{(1)} = \mathbf{D}^{(1)} {\mathbf{W}^{(2)}}^\top \boldsymbol{\delta}^{(2)} = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -2 \end{pmatrix} \begin{pmatrix} 4 \\ 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 8 \\ 4 \\ 0 \end{pmatrix}, \qquad \frac{\partial y}{\partial \mathbf{W}^{(1)}} = \boldsymbol{\delta}^{(1)} x^\top = \begin{pmatrix} 8 \\ 4 \\ 0 \end{pmatrix}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파 계산 예시 (4) — 경사하강법 갱신

$$\mathbf{W}^{(1)} = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}, \quad \mathbf{W}^{(2)} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -2 \end{pmatrix}, \quad \mathbf{W}^{(3)} = \begin{pmatrix} 1 & -1 \end{pmatrix}, \quad y = 3, \quad y_{\text{data}} = 1, \quad \eta = 0.01$$

$$\partial y/\partial \mathbf{W}^{(1)} = (8, 4, 0)^{\top}, \quad \partial y/\partial \mathbf{W}^{(2)} = \begin{pmatrix} 4 & 4 & 4 \\ 2 & 2 & 2 \end{pmatrix}, \quad \partial y/\partial \mathbf{W}^{(3)} = (4 \;\; 1)$$

손실 $l = (y - y_{\text{data}})^2$에서 $y_{\text{data}} = 1$ 이므로, $\partial l/\partial \mathbf{W}^{(\ell)} = 2(y - y_{\text{data}})\, \partial y/\partial \mathbf{W}^{(\ell)} = 4\partial y/\partial \mathbf{W}^{(\ell)}$

경사하강법(7장) 갱신식 $\mathbf{W}^{(\ell)} \leftarrow \mathbf{W}^{(\ell)} - \eta\, \partial l/\partial \mathbf{W}^{(\ell)}$에 대입

$$\mathbf{W}^{(3)} \leftarrow \begin{pmatrix} 1 & -1 \end{pmatrix} - 0.04 \begin{pmatrix} 4 & 1 \end{pmatrix} = \begin{pmatrix} 0.84 & -1.04 \end{pmatrix}$$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

역전파 계산 예시 (4) — 경사하강법 갱신

$$\mathbf{W}^{(1)} = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}, \quad \mathbf{W}^{(2)} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -2 \end{pmatrix}, \quad \mathbf{W}^{(3)} = \begin{pmatrix} 1 & -1 \end{pmatrix}, \quad y = 3, \quad y_{\text{data}} = 1, \quad \eta = 0.01$$

$$\partial y/\partial \mathbf{W}^{(1)} = (8, 4, 0)^{\top}, \quad \partial y/\partial \mathbf{W}^{(2)} = \begin{pmatrix} 4 & 4 & 4 \\ 2 & 2 & 2 \end{pmatrix}, \quad \partial y/\partial \mathbf{W}^{(3)} = (4 \;\; 1)$$

$$\mathbf{W}^{(3)} \leftarrow \begin{pmatrix} 1 & -1 \end{pmatrix} - 0.04 \begin{pmatrix} 4 & 1 \end{pmatrix} = \begin{pmatrix} 0.84 & -1.04 \end{pmatrix}$$

$$\mathbf{W}^{(2)} \leftarrow \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & -2 \end{pmatrix} - 0.04 \begin{pmatrix} 4 & 4 & 4 \\ 2 & 2 & 2 \end{pmatrix} = \begin{pmatrix} 0.84 & -0.16 & 0.84 \\ -0.08 & 0.92 & -2.08 \end{pmatrix}$$

$$\mathbf{W}^{(1)} \leftarrow \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} - 0.04 \begin{pmatrix} 8 \\ 4 \\ 0 \end{pmatrix} = \begin{pmatrix} 0.68 \\ 0.84 \\ -1 \end{pmatrix}$$

- $w_{31}^{(1)}$ 은 그래디언트가 0. why? (D.I.Y.)
- 위 식에 사용한 $\eta = 0.01$로 갱신 후 순전파: $y \approx -1.20$, 손실 $4 \to 4.82$로 증가. 엥?
- 만약 $\eta = 0.001$이면 $y \approx 2.40$, 손실 $4 \to 1.96$으로 감소 $\Rightarrow$ 학습률 선택의 중요성

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

NumPy 구현 — 셀 1: 파라미터

모든 벡터를 열벡터, 즉 모양 `(d,1)`의 2차원 배열로 두면 `@`와 `.T`가 행렬 곱·전치에 그대로 대응

```
import numpy as np
x  = np.array([[1.0]])                    # 입력 (1,1)
s  = lambda z: z**2                       # sigma(z) = z^2
ds = lambda z: 2*z                        # sigma'(z) = 2z
W1 = np.array([[1.0], [1.0], [-1.0]])     # (3,1)
W2 = np.array([[1.0, 0.0,  1.0],
               [0.0, 1.0, -2.0]])         # (2,3)
W3 = np.array([[1.0, -1.0]])              # (1,2)
```

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

NumPy 구현 — 셀 2: 순전파

역전파에서 다시 쓰므로 $z^{(\ell)}, h^{(\ell)}$ 을 모두 저장. `.ravel()`은 배열을 1차원으로 펼침

```
z1 = W1 @ x;   h1 = s(z1)
z2 = W2 @ h1;  h2 = s(z2)
y  = W3 @ h2                                  # 출력층: 활성화 없음
print("h1 =", h1.ravel(), " h2 =", h2.ravel(), " y =", y.ravel())
```

```
h1 = [1. 1. 1.]  h2 = [4. 1.]  y = [3.]
```

계산 예시 (1)의 $h^{(1)}, h^{(2)}, y$와 같은 값

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

NumPy 구현 — 셀 3: 역전파 (코드)

각 줄이 계산 예시의 한 줄에 대응. `np.diag(ds(z2).ravel())`이 $\mathbf{D}^{(2)}$, `delta2 @ h1.T`가 외적 $\boldsymbol{\delta}^{(2)}\boldsymbol{h}^{(1)\top}$

```
delta3 = np.array([[1.0]])                # dy/dz3 = 1, (1,1)
dW3 = delta3 @ h2.T                       # (1,2)
D2 = np.diag(ds(z2).ravel())              # (2,2)
delta2 = D2 @ W3.T @ delta3               # (2,1)
dW2 = delta2 @ h1.T                       # (2,3)
D1 = np.diag(ds(z1).ravel())              # (3,3)
delta1 = D1 @ W2.T @ delta2               # (3,1)
dW1 = delta1 @ x.T                        # (3,1)
print("delta2 =", delta2.ravel(), " delta1 =", delta1.ravel())
print("dy/dW3 =", dW3, "\ndy/dW2 =\n", dW2, "\ndy/dW1 =\n", dW1)
```

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

NumPy 구현 — 셀 3: 역전파 (출력)

```
delta2 = [4. 2.]  delta1 = [8. 4. 0.]
dy/dW3 = [[4. 1.]]
dy/dW2 =
 [[4. 4. 4.]
 [2. 2. 2.]]
dy/dW1 =
 [[8.]
 [4.]
 [0.]]
```

계산 예시 (2), (3)과 같은 값. 세 그래디언트의 모양 $=$ $\mathbf{W}^{(3)}, \mathbf{W}^{(2)}, \mathbf{W}^{(1)}$ 의 모양

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

함수족의 표현력: 보편 근사 정리

신경망 함수족은 얼마나 넓은가

이 장이 한 두 가지 일: 완전연결 신경망이라는 함수족의 정의(정의 9.2), 그 안에서 손실을 줄이는 파라미터를 찾는 절차(역전파 + 경사하강법)

- 선형함수족 $\mathcal{F}_{\text{lin}}$에는 자유낙하 궤적처럼 어떤 파라미터로도 따라갈 수 없는 관계가 있었음. 활성화함수로 넓힌 함수족에도 여전히 그런 관계가 남아 있는가?
- 함수족의 이름: 은닉층 하나 (폭 d_1)인 d_x-d_1-1 FCNN 전체, 즉 $\boldsymbol{\theta} = (\mathbf{W}^{(1)}, \boldsymbol{b}^{(1)}, \mathbf{W}^{(2)}, b^{(2)})$를 모든 값으로 바꾸어 얻는 함수의 집합을 $\mathcal{F}_{d_x\text{-}d_1\text{-}1}$

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

정리 9.2 — 보편 근사 정리(universal approximation theorem)

다음 세 조건을 가정하자.

1) 활성화함수 $\sigma : \mathbb{R} \to \mathbb{R}$가 연속이고 다항식이 아니다(시그모이드, tanh, ReLU, ELU가 모두 해당한다).
2) 입력의 차원은 $d_x \geq 1$이고 정의역 $K \subset \mathbb{R}^{d_x}$가 공집합이 아닌 닫힌 유계 집합이다 ($\Leftrightarrow$ Compact).
3) 대상(목표) 함수 $f^* : K \to \mathbb{R}$가 연속함수이다.

그러면 임의의 $\varepsilon > 0$에 대하여

$$\max_{\boldsymbol{x} \in K} \left| f(\boldsymbol{x}; \boldsymbol{\theta}) - f^*(\boldsymbol{x}) \right| < \varepsilon$$

을 만족하는 폭 $d_1 \geq 1$과 파라미터 $\boldsymbol{\theta}$가 존재한다. 즉 $\mathcal{F}_{d_x\text{-}d_1\text{-}1}$ 안에는 f^*와의 최대 오차가 ε보다 작은 함수가 있다 [Leshno et al., 1993, 정리 1][2].

[2] Moshe Leshno, Vladimir Ya Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural networks*, 6(6), 1993

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

근사 정리 가정/조건의 역사

- 출력이 벡터인 경우는 성분마다 적용하면 되므로 $d_y = 1$이면 충분
- 은닉층이 더 많아지면? 근사 정리 관점에서 큰 문제는 아님: d_x-d_1-d_2-1에서 첫 은닉층이 K 위의 항등사상 $x \mapsto x$를 구현하면 앞 정리와 거의 같은 이야기
  - 폭을 $d_x + 3$ ($= d_x + d_y + 2$, ReLU는 $d_x + 2$)으로 고정하고 은닉층을 많이 쌓아도 같은 결론 Kidger and Lyons [2020][3]
- (역사적으로) 활성화함수 σ가 어떤 조건을 만족하는가에 따라 근사 정리 결과가 존재
  - Cybenko [1989][4]: 연속, $\lim_{z\to\infty} \sigma(z) = 1$, $\lim_{z\to-\infty} \sigma(z) = 0$ (시그모이드 형태)
  - Hornik [1991][5]: 유계이고 상수가 아닌 연속함수
  - Leshno et al. [1993][6]: "다항식이 아닌 함수" (앞 슬라이드)
- 예제에서 쓴 $\sigma(z) = z^2$은 다항식이므로 이 정리의 활성화함수 조건에 해당하지 않음. **실제로 쓰면 안됨.** Why?

[3] Patrick Kidger and Terry Lyons. Universal approximation with deep narrow networks. In *Proceedings of the 33rd Conference on Learning Theory* (*COLT*), volume 125 of *Proceedings of Machine Learning Research*, pages 2306–2327, 2020

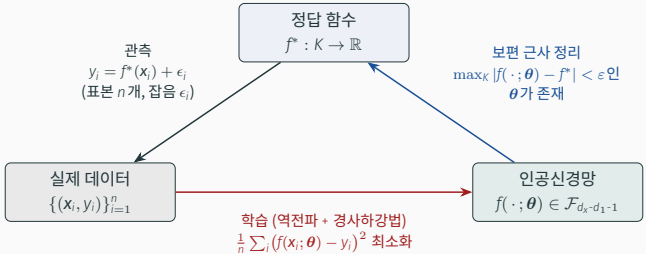
[4] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2, 1989

[5] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2), 1991

[6] Moshe Leshno, Vladimir Ya Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural networks*, 6(6), 1993

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

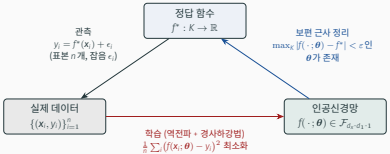
그림 9.8 — 근사 정리가 시사하는 바, 이론과 현실의 갭(gap)



- **이론-근사 정리**: 폭 d_1을 충분히 잡으면 근사하는 θ가 존재. 다만 d_1, θ의 값은 제시 x—근사 가능성
- **현실-학습**: 구조(폭 포함)를 먼저 정하고 그 안에서 총손실을 줄임—최적화
  - 폭이 모자라면 목표 오차에 못 미치고, 충분해도 손실이 볼록이 아니어서(nonconvex) 국소최소에 머물 수 있음. 너무 많으면 과적합
- **현실-관측:** 학습이 보는 것은 f^*가 아니라 표본 n개 위치에서 잡음 ϵ_i가 섞인 값 y_i

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

그림 9.8 — 삼각부등식 관계?



$$\underbrace{\left|\text{인공신경망} - \text{정답 함수}\right|}_{\text{근사 가능성: 수학(해석학)}} \le \underbrace{\left|\text{정답 함수} - \text{실제 데이터}\right|}_{\text{잡음: 수리통계}} + \underbrace{\left|\text{실제 데이터} - \text{인공신경망}\right|}_{\text{학습: 최적화}}$$

표본 위치 x_i에서: $\left|f(x_i;\theta) - f^*(x_i)\right| \le |\epsilon_i| + \left|y_i - f(x_i;\theta)\right|$ (잡음 + 학습 잔차)

- 잡음 제어와 학습 잔차 사이의 trade-off가 존재 (다음 chapter)
- 관측하지 않은 x에서는 우변을 잴 수 없어 부등식이 아무 정보도 주지 않음 (다음 chapter)
- 분석 목적에 따라 세 항의 위치는 바꿀 수 있음 (예: 학습 잔차 $\le$ 잡음 + 근사 오차)
- 참고: Bottou and Bousquet [2007][7], Berner et al. [2022][8]

[7] Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems 20* (*NeurIPS 2007*), pages 161–168, 2007

[8] Julius Berner, Philipp Grohs, Gitta Kutyniok, and Philipp Petersen. The modern mathematics of deep learning. In Philipp Grohs and Gitta Kutyniok, editors, *Mathematical Aspects of Deep Learning*, pages 1–111. Cambridge University Press, 2022. arXiv:2105.04026 (2021)

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

챕터 요약

- 선형모델은 비선형 관계를 파악하기엔 현실적 한계
  - 선형변환 사이에 활성화함수 σ를 추가하여 비선형성 자체를 학습 = 인공신경망
- 완전연결 신경망 d_x-d_1-$\cdots$-d_{L-1}-d_y: 선형 결합 + 활성화함수 합성
  - 선형 결합: $z^{(\ell)} = \mathbf{W}^{(\ell)} h^{(\ell-1)} + b^{(\ell)}, \qquad 1 \le \ell \le L-1$
  - 합성: $h^{(\ell)} = \sigma(z^{(\ell)}), \qquad 1 \le \ell \le L-1$
  - 선형 출력층: $y = \mathbf{W}^{(L)} h^{(L-1)} + b^{(L)}$
- 역전파의 일반 공식: $\boldsymbol{\delta}^{(\ell)} = \mathbf{D}^{(\ell)} \mathbf{W}^{(\ell+1)\top} \boldsymbol{\delta}^{(\ell+1)}$(역방향 전달), $\partial y / \partial \mathbf{W}^{(\ell)} = \boldsymbol{\delta}^{(\ell)} h^{(\ell-1)\top}$(가중치 그래디언트), $\mathbf{D}^{(\ell)} = \mathrm{diag}(\sigma'(z^{(\ell)}))$
  - 행렬 단위 그래디언트 계산 $\rightarrow$ 병렬계산이 가능!
- 보편 근사 정리: 닫힌 유계 집합 위의 연속함수 f^* 와의 원하는 오차 수준 내 근사할 수 있는 인공신경망 구조를 항상 설계 가능. 단, 가능성만 제시하고 폭과 파라미터의 값은 모름.
  - 구조를 정해 놓고 손실을 줄이는 학습 방식과는 gap이 존재함.

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지

감사합니다

다음 장: 학습의 실제: 안정성, 초기화, 일반화 — 역전파 공식에 숨은 위험과 표본 밖의 오차

© 2026 조현태 · 아주대학교 수학과 | 수강생 전용, 재배포·상업적 이용 금지