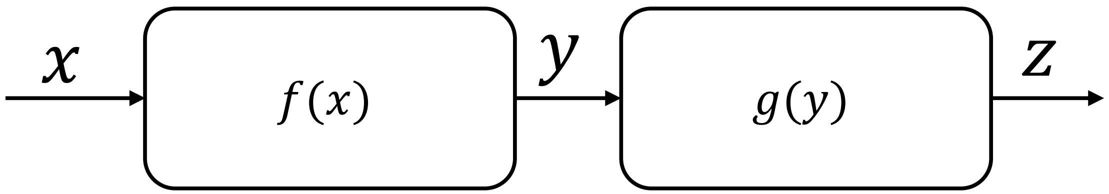
Part. 04

Back Propagation

| 역전파 알고리즘

FASTCAMPUS
ONLINE
강사. 신제용

| 직렬 연결된 두 함수의 미분



$$y = f(x) \qquad z = g(y)$$

$$\frac{dy}{dx} = \frac{df(x)}{dx} \qquad \frac{dz}{dy} = \frac{dg(y)}{dy}$$

$$\frac{dz}{dx} = ?$$

직렬 연결된(Cascaded) 두 함수의 미분. 가볍게 고등학교때 배운 것 부터 떠올려 봅시다.

| 미분과 연쇄법칙

$$x \longrightarrow \boxed{f(x)} \xrightarrow{\;y\;} \boxed{g(y)} \xrightarrow{\;z\;}$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$y = f(x) \qquad\qquad z = g(y)$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$\frac{dy}{dx} = \frac{df(x)}{dx} \qquad\qquad \frac{dz}{dy} = \frac{dg(y)}{dy}$$

연쇄 법칙 (Chain Rule)

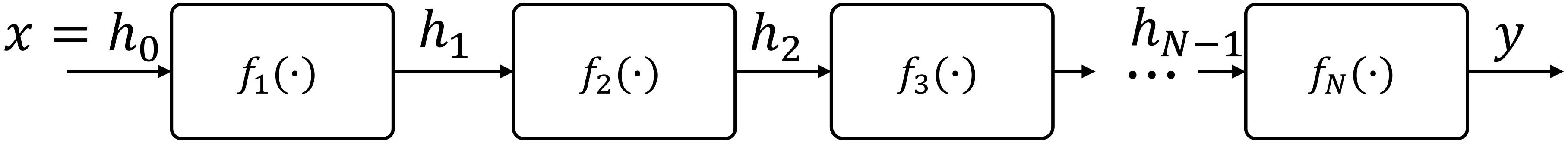
$$\boxed{\frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx} = \frac{dg(y)}{dy}\frac{df(x)}{dx}}$$

합성 함수의 미분 (겉미분과 속미분)

$$\boxed{\frac{dz}{dx} = \frac{dg\big(f(x)\big)}{dx} = g'\big(f(x)\big)f'(x)}$$

연쇄 법칙을 이용한 미분의 계산. 고등학교 과정에서 배우는 합성 함수의 미분과 동일

ㅣ연쇄 법칙 확장



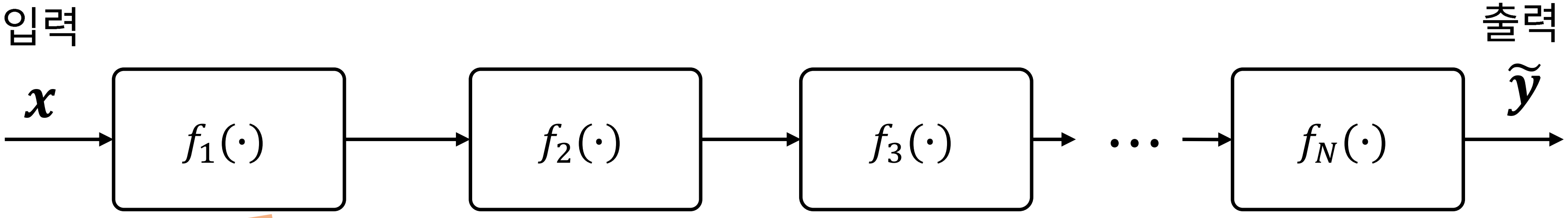
$$y = f_N(f_{N-1}(\cdots f_2(f_1(x))))$$

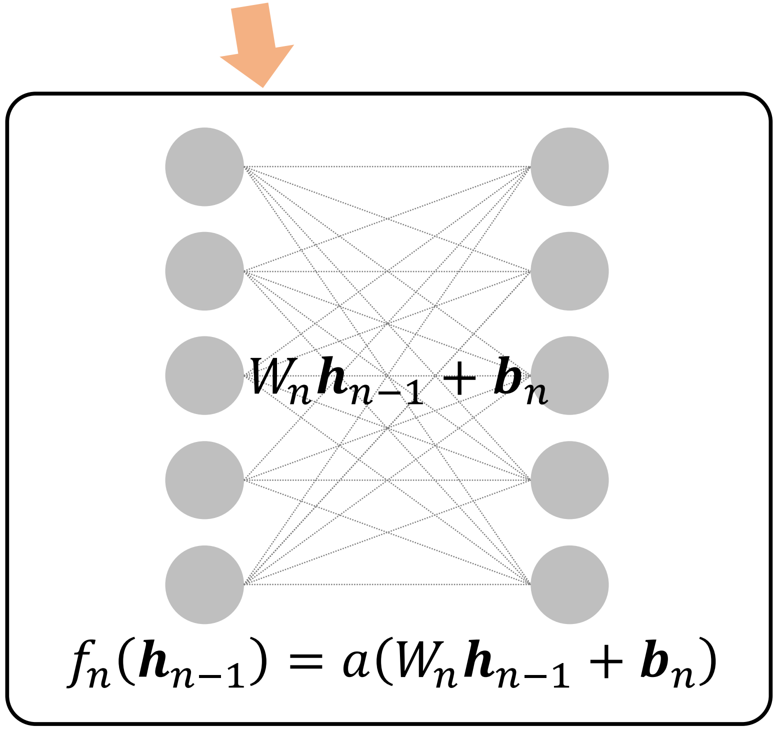
$$\frac{dy}{dx} = \frac{df_N(h_{N-1})}{dh_{N-1}} \frac{df_{N-1}(h_{N-2})}{dh_{N-2}} \cdots \frac{df_1(x)}{dx}$$

$$= \boxed{\prod_{n=1}^{N} \frac{df_n(h_{n-1})}{dh_{n-1}}}$$

연쇄 법칙을 이용하면 연속된 함수의 미분을 각각의 미분의 곱으로 표현할 수 있다.

ㅣ합성 함수로서의 심층신경망

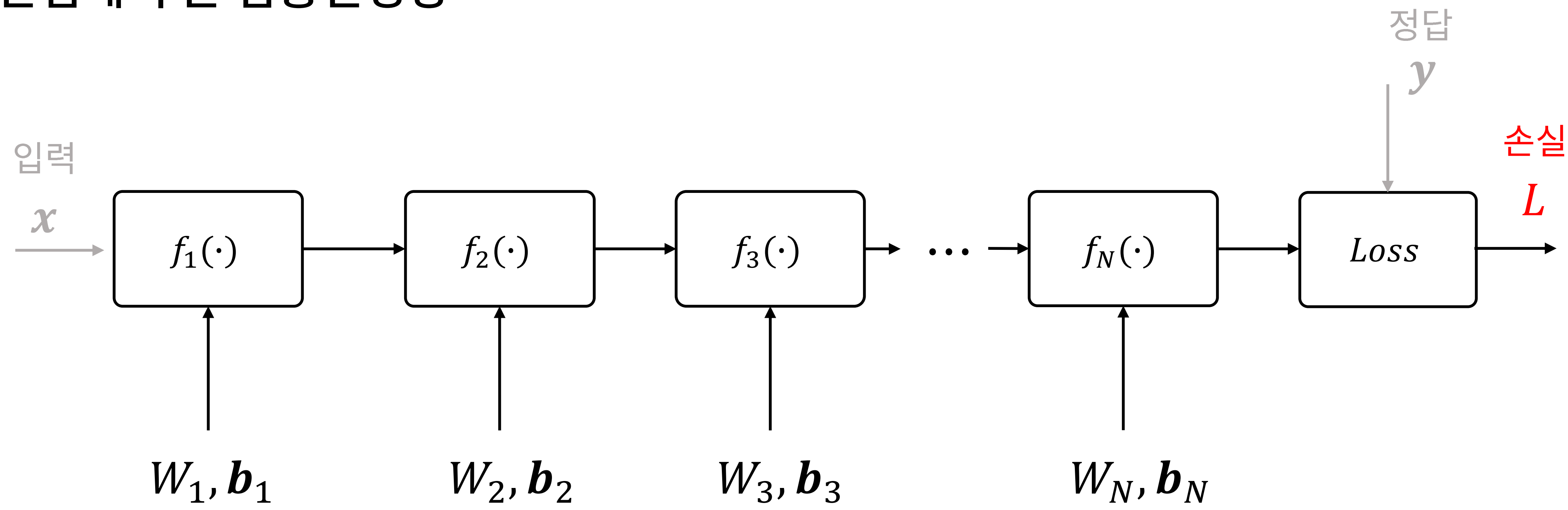




$$\tilde{\boldsymbol{y}} = f_N(f_{N-1}(\cdots f_2(f_1(x))))$$

심층 신경망의 각 Layer를 하나의 함수로 본다면, 신경망을 합성 함수로 표현할 수 있다.

I 학습 관점에서 본 심층신경망

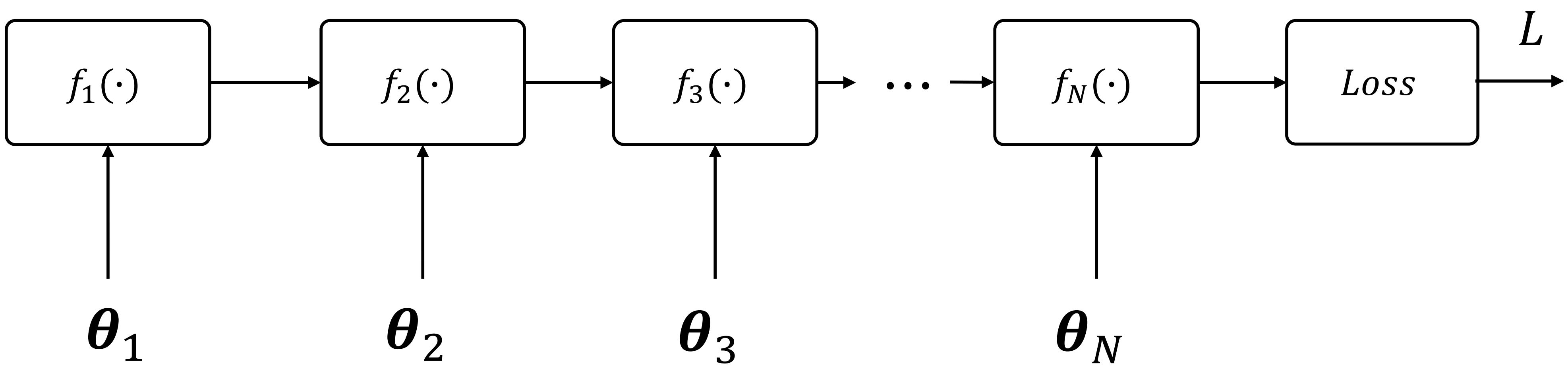


$$f_n(\boldsymbol{h}_{n-1}; W_n, \boldsymbol{b}_n) = a(W_n \boldsymbol{h}_{n-1} + \boldsymbol{b}_n)$$

이미 손실을 구했다면, 데이터셋의 입력과 출력은 학습 과정에서 중요하지 않다.

손실을 최소화하는 파라미터만 찾으면 되기 때문!

심층신경망의 연쇄 법칙



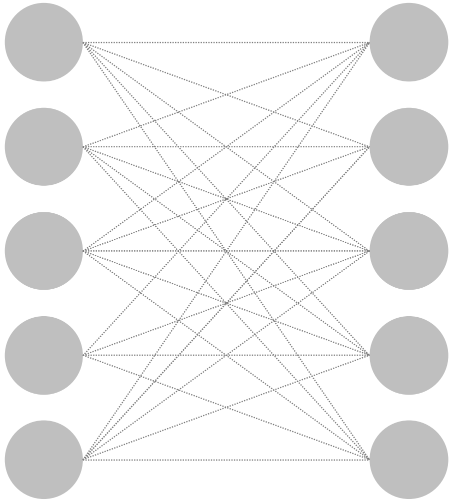
$$f_n(\boldsymbol{\theta}_n) = a(W_n \boldsymbol{h}_{n-1} + \boldsymbol{b}_n)$$

$$\boldsymbol{\theta}_n = [vec(W_n)^T | \boldsymbol{b}_n^T]^T$$

$$\boxed{\frac{\partial L}{\partial \boldsymbol{\theta}_n} = \frac{\partial L}{\partial \boldsymbol{h}_N} \prod_{i=n+1}^{N} \frac{\partial f_i(\boldsymbol{h}_{i-1})}{\partial \boldsymbol{h}_{i-1}} \frac{\partial f_n(\boldsymbol{\theta}_n)}{\partial \boldsymbol{\theta}_n}}$$

미분하고자 하는 경로 사이에 있는 모든 미분 값을 곱하면 원하는 미분을 구할 수 있다.

Fully-Connected Layer의 미분 (1)



$$\boldsymbol{h}_n = a(W_n \boldsymbol{h}_{n-1} + \boldsymbol{b}_n)$$

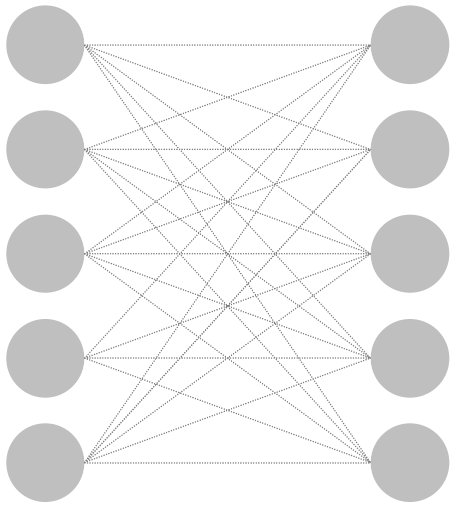
$$\boxed{\widehat{\boldsymbol{h}}_n = W_n \boldsymbol{h}_{n-1} + \boldsymbol{b}_n}$$

$$\hat{h}_n^j = \sum_i w_n^{i,j} h_{n-1}^i + b_{n-1}^j$$

$k = i$인 항만 살아남는다.

$$\begin{cases} \dfrac{\partial \hat{h}_n^j}{\partial h_{n-1}^i} = \dfrac{\partial\left[\sum_k w_n^{k,j} h_{n-1}^k + \cancel{b_n^j}\right]}{\partial h_{n-1}^i} = \boxed{w_{n-1}^{i,j}} \\ \dfrac{\partial \hat{h}_n^j}{\partial w_{n-1}^{i,j}} = \dfrac{\partial\left[\sum_k w_n^{k,j} h_{n-1}^k + \cancel{b_n^j}\right]}{\partial w_{n-1}^{i,j}} = \boxed{h_{n-1}^i} \\ \dfrac{\partial \hat{h}_n^j}{\partial b_{n-1}^j} = \dfrac{\partial\left[\cancel{\sum_k w_n^{k,j} h_{n-1}^k} + b_n^j\right]}{\partial b_{n-1}^j} = \boxed{1} \end{cases}$$

Fully-Connected Layer의 미분 (2)



$$\boldsymbol{h}_n = a(W_n \boldsymbol{h}_{n-1} + \boldsymbol{b}_n)$$

$$\widehat{\boldsymbol{h}}_n = W_n \boldsymbol{h}_{n-1} + \boldsymbol{b}_n$$

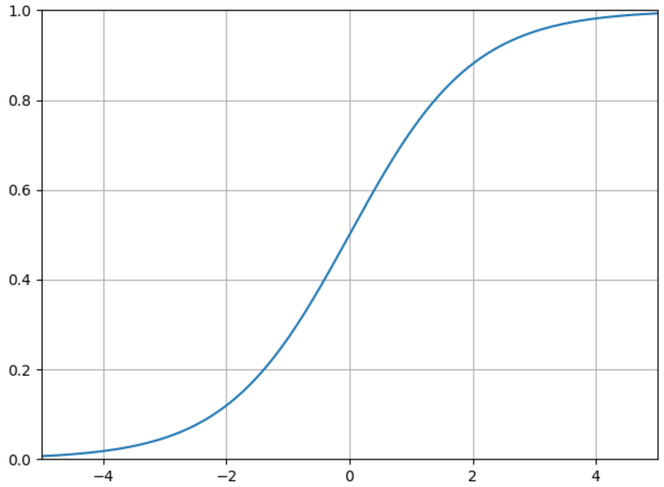
Activation function의 미분

$$\begin{cases} \dfrac{\partial h_n^j}{\partial h_{n-1}^i} = \dfrac{\partial a^j}{\partial \hat{h}_n^j}\dfrac{\partial \hat{h}_n^j}{\partial h_{n-1}^i} = \boxed{\dfrac{\partial a^j}{\partial \hat{h}_n^j}}\boxed{w_n^{i,j}} \\ \dfrac{\partial h_n^j}{\partial w_n^{i,j}} = \dfrac{\partial a^j}{\partial \hat{h}_n^j}\dfrac{\partial \hat{h}_n^j}{\partial w_n^{i,j}} = \boxed{\dfrac{\partial a^j}{\partial \hat{h}_n^j}}\boxed{h_{n-1}^i} \\ \dfrac{\partial h_N^j}{\partial b_N^i} = \dfrac{\partial a^j}{\partial \hat{h}_N^j}\dfrac{\partial \hat{h}_N^j}{\partial b_N^i} = \boxed{\dfrac{\partial a^j}{\partial \hat{h}_N^j}} \end{cases}$$

Activation function의 미분을 알고 있다면, 쉽게 Fully-connected layer를 미분할 수 있다.

I Sigmoid 함수의 미분

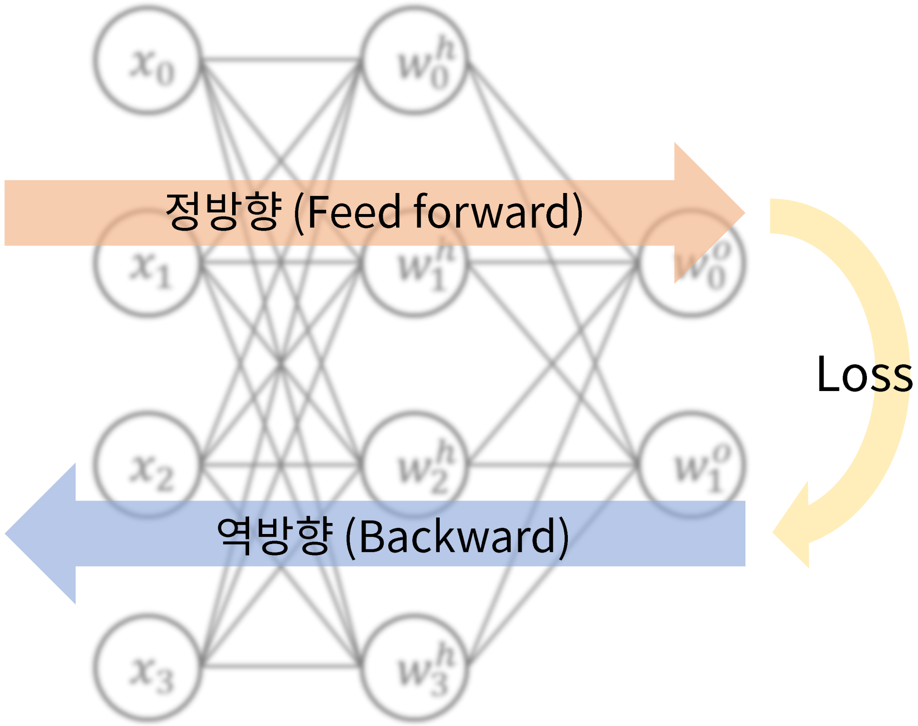
$$\frac{d}{dx}sigmoid(x) = \frac{d}{dx}(1+e^{-x})^{-1}$$
$$= (-1)\frac{1}{(1+e^{-x})^2}\frac{d}{dx}(1+e^{-x})$$
$$= (-1)\frac{1}{(1+e^{-x})^2}e^{-x}\frac{d}{dx}(-x)$$
$$= (-1)\frac{1}{(1+e^{-x})^2}e^{-x}(-1)$$
$$= \frac{e^{-x}}{(1+e^{-x})^2}$$
$$= \frac{1+e^{-x}-1}{(1+e^{-x})^2}$$
$$= \frac{1}{1+e^{-x}} - \frac{1}{(1+e^{-x})^2}$$
$$= sigmoid(x) - sigmoid(x)^2$$
$$= sigmoid(x)\big(1 - sigmoid(x)\big)$$



$$\mathrm{sigmoid}(x) = \frac{1}{1+e^{-x}}$$

초창기 신경망에 가장 많이 쓰인 Sigmoid 활성 함수의 미분. 정리된 결과를 이용하면 매우 간단하게 미분할 수 있다.

| 역전파 알고리즘



오류 역전파 알고리즘
(Backpropagation Algorithm; BP)

- 학습 데이터로 정방향 연산을 하여 Loss를 구한다.
- 정방향 연산 시, 계층별로 BP에 필요한 중간 결과를 저장한다.
- Loss를 각 파라미터로 미분한다. 연쇄 법칙(역방향 연산)을 이용한다.
  - 마지막 계층부터 하나씩 이전 계층으로 연쇄적으로 계산한다.
  - 역방향 연산 시, 정방향 연산에서 저장한 중간 결과를 사용한다.

미분의 연쇄 법칙과 각 함수의 수식적 미분을 이용하면, 단 한번의 손실함수 평가로 미분을 구할 수 있다.
단, 중간 결과를 저장해야 하므로 메모리를 추가로 사용한다.