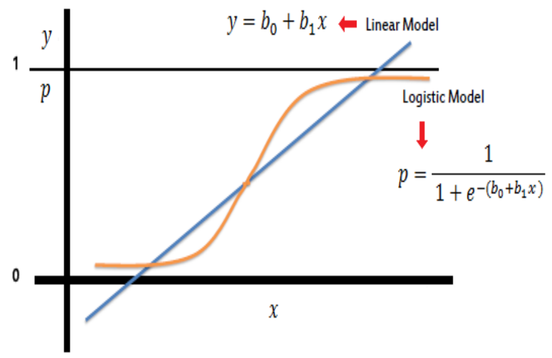
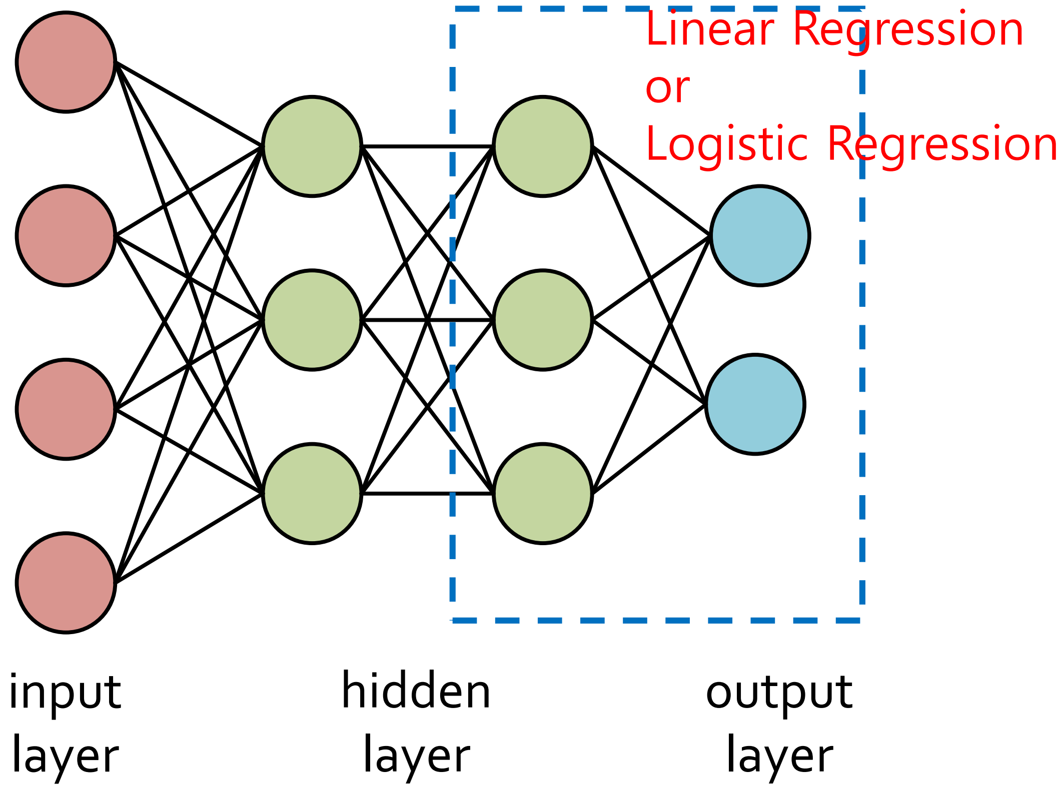
Linear Regression &

Logistic Regression

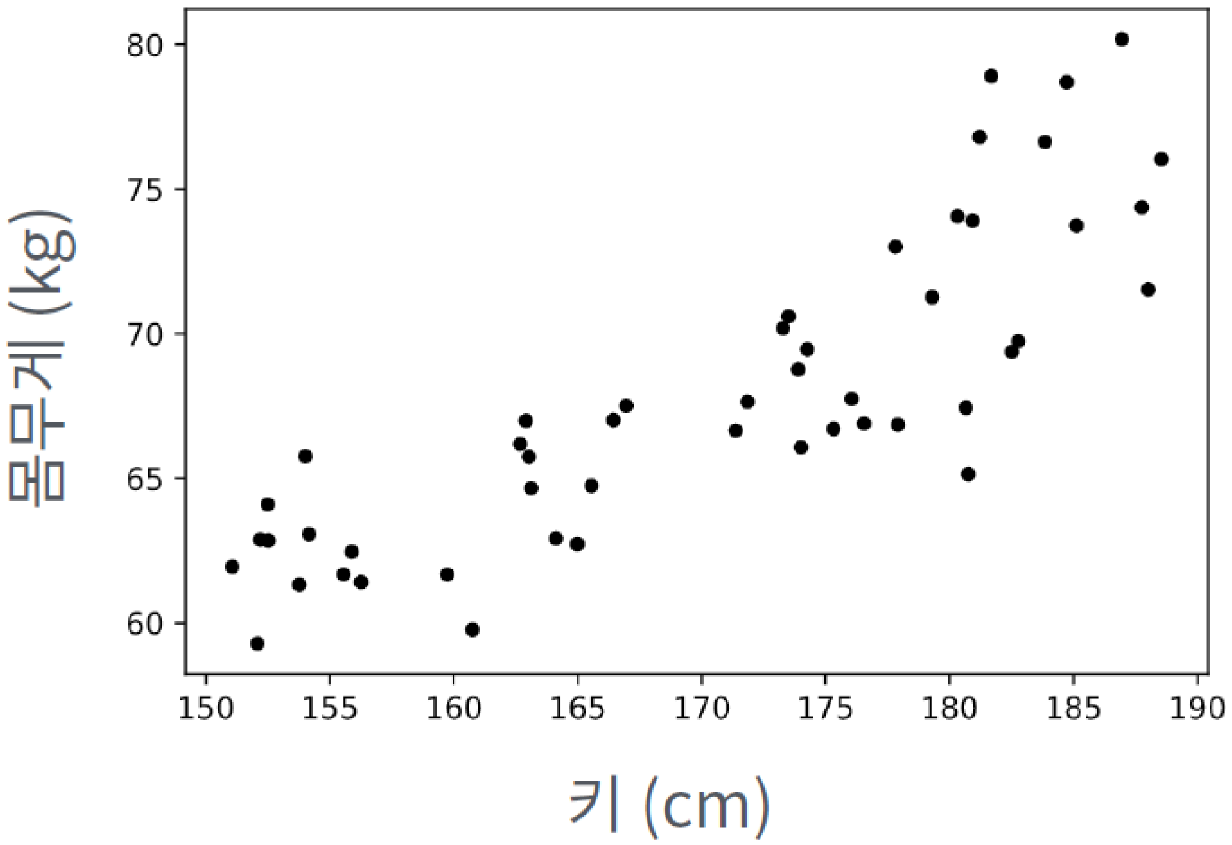


Deep Learning Uses
Linear Regression/Logistic Regression



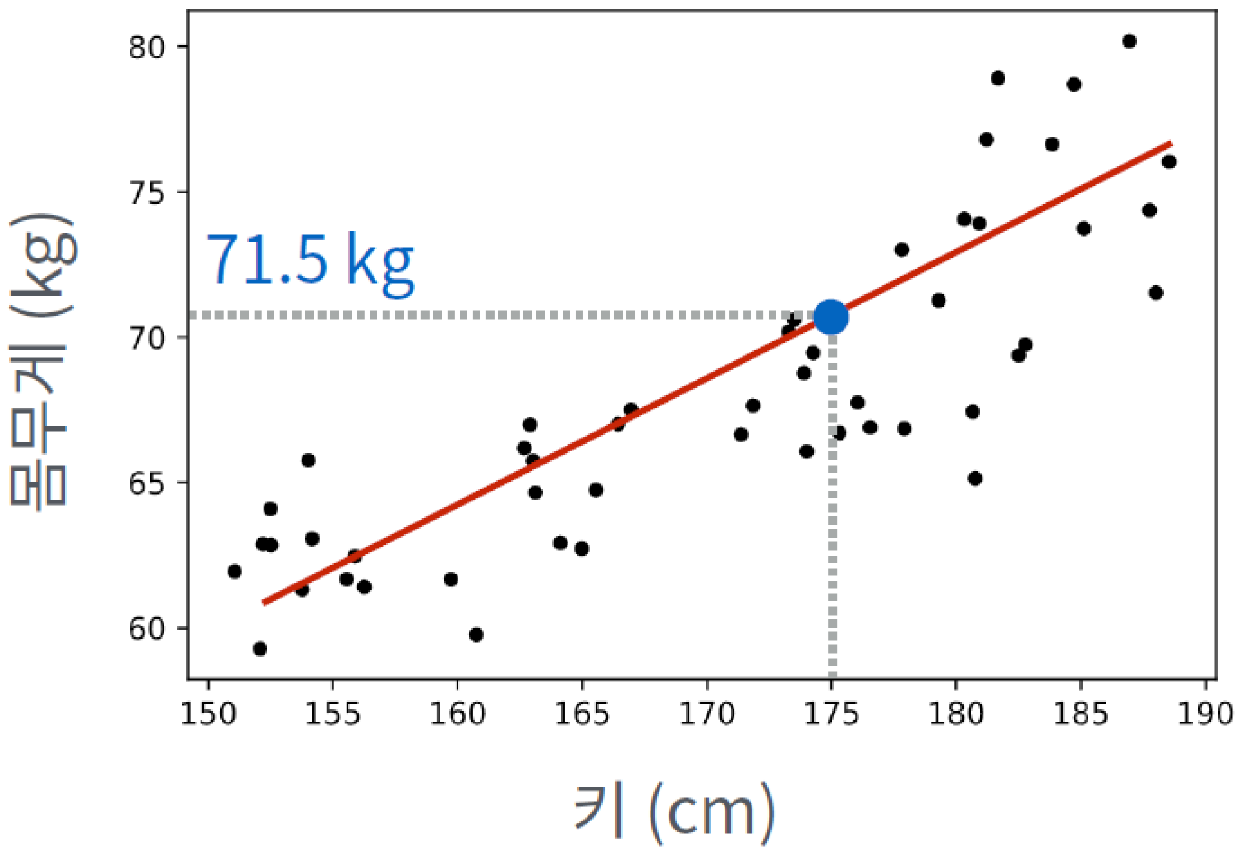
Linear Regression

- 어느 학교 학생들의 신체검사 자료
- 새로 전학온 학생 A의 키가 175cm일 때 예상 몸무게는?



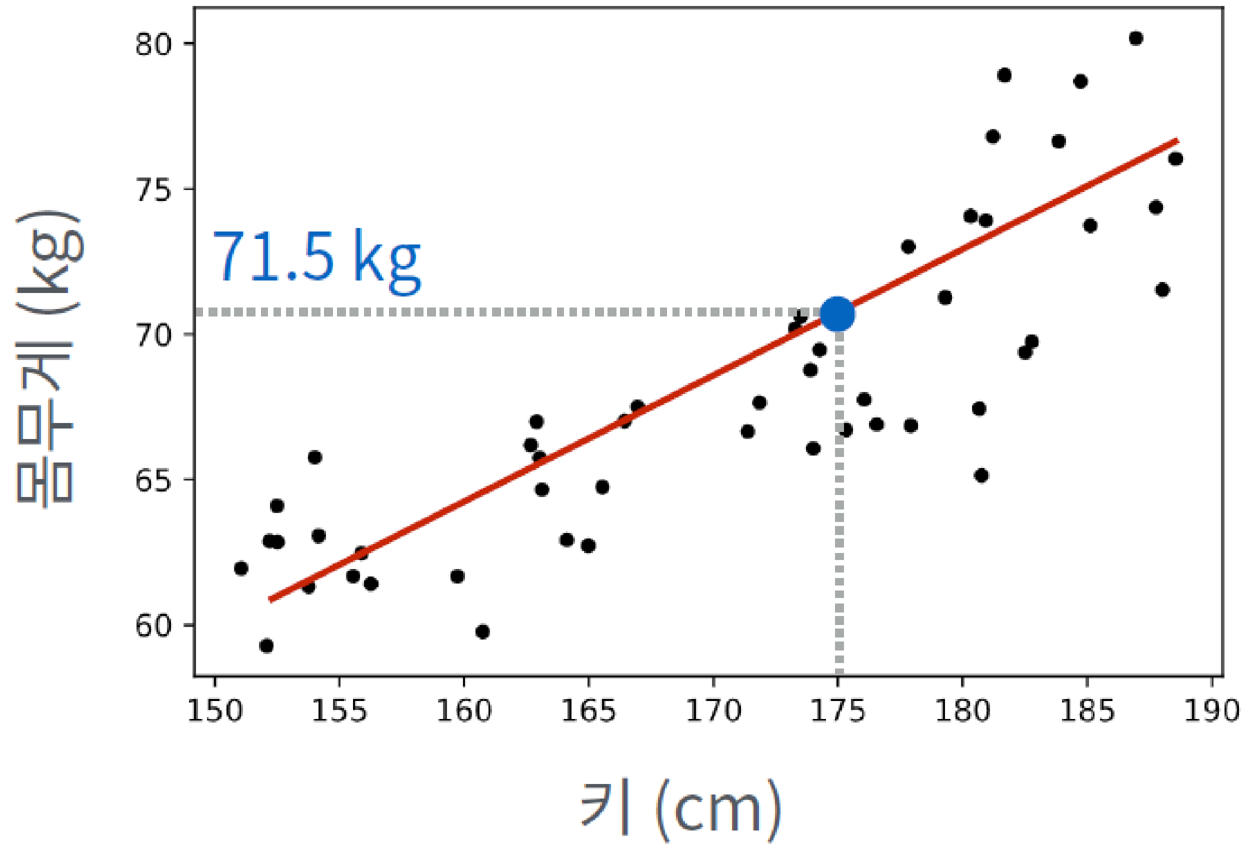
Linear Regression

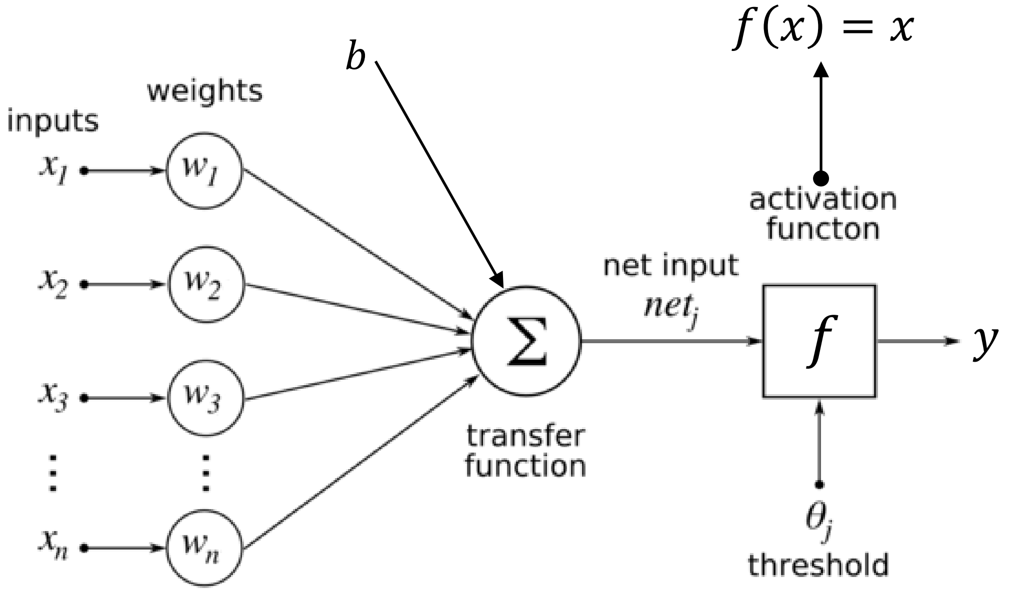
- 어느 학교 학생들의 신체검사 자료
- 새로 전학온 학생 A의 키가 175cm일 때 예상 몸무게는?



Linear Regression

- 선형함수(예 : 1차함수)로 주어진 data를 근사한다
- $y = wx + b$





<Perceptron>

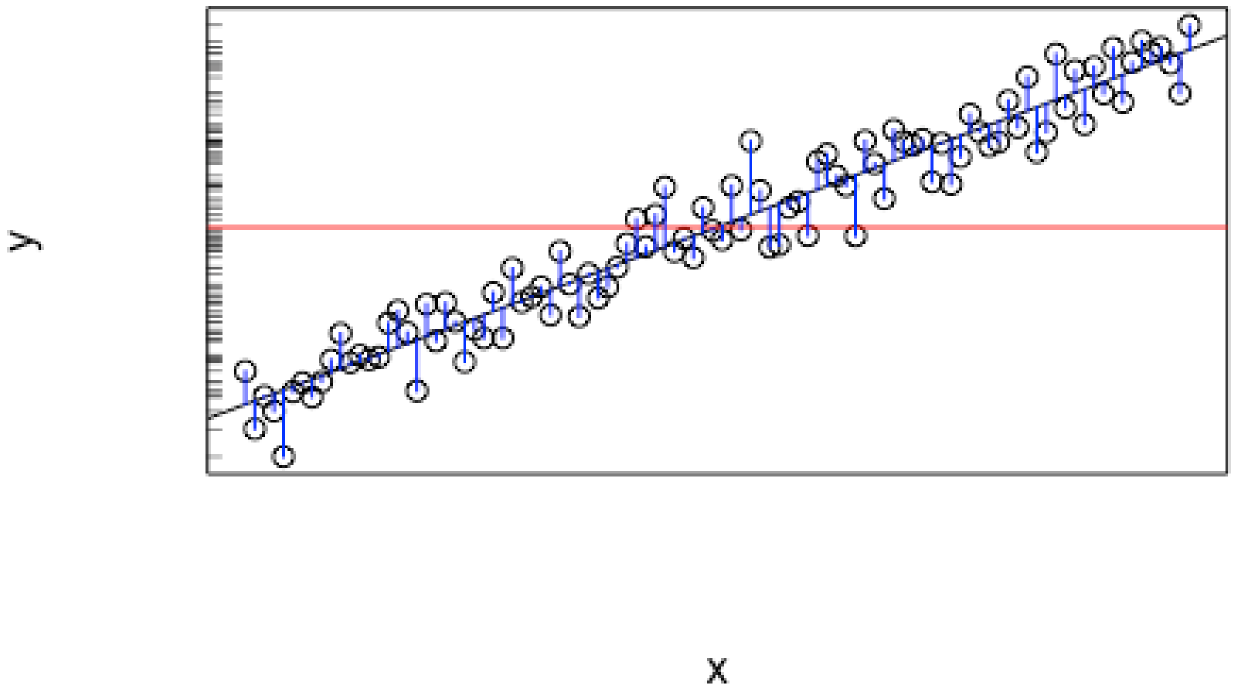
$$y = f(\mathbf{wx} + \mathrm{b})$$

$$\boldsymbol{w} = [w_1\ w_2\ w_3\ \dots\ w_n]$$

$$\boldsymbol{x} = [x_1\ x_2\ x_3\ \dots\ x_n]^T$$

Linear Regression

- 잘 예측했는지 측정할 척도(metric)가 필요함



$$y^* = wx + b \text{ (예측값)}$$

$$Loss = \sum_i (y_i - y_i^*)^2$$

$$= \sum_i (y_i - wx_i - b)^2$$

Linear Regression

- Loss 값을 minimize하는 w와 b를 구하면 될텐데.... 어떻게?
  - Random Search – 가능????
  - Cost function을 미분해서 최솟값(미분=0이되는 점)을 찾자!

b 구하기

$$L = \sum_i (y_i - wx_i - b)^2$$

$$\frac{\delta L}{\delta b} = \frac{\delta \sum_i (y_i - wx_i - b)^2}{\delta b}$$

$$= -2\sum_i (y_i - wx_i - b) = ny_{avg} - nwx_{avg} - nb = 0$$

$$\therefore \boldsymbol{b = y_{avg} - wx_{avg}}$$

w 구하기

$$L = \sum_i (y_i \ - wx_i \ - b)^2$$

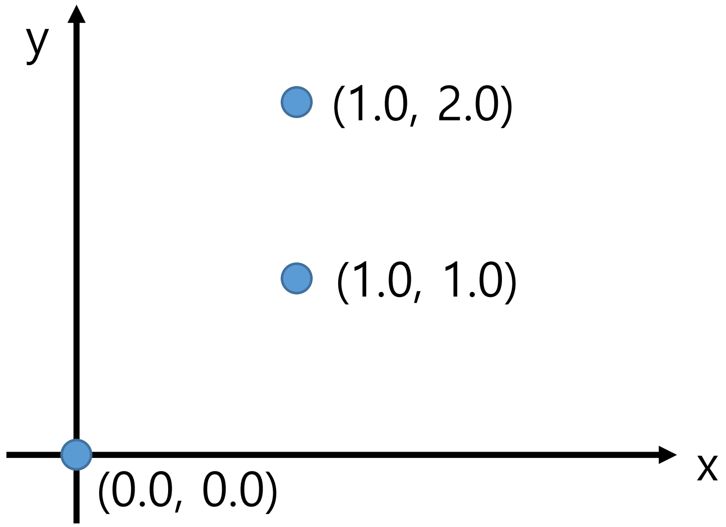
$$\frac{\delta L}{\delta w} = \frac{\delta \sum_i (y_i \ - wx_i \ - b)^2}{\delta w}$$

$$= -2 \sum_i x_i (y_i \ - wx_i \ - b) = -2 \sum_i x_i \left(y_i \ - wx_i \ - y_{avg} + wx_{avg}\right)$$

$$= 0$$

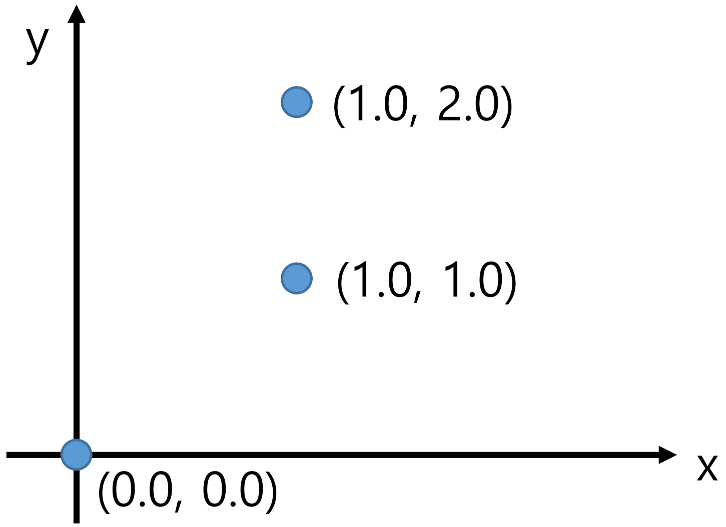
Example

- Find the linear function(f) that best describes the given data
  - $H(x, w_0, w_1) = w_1 x + w_0$



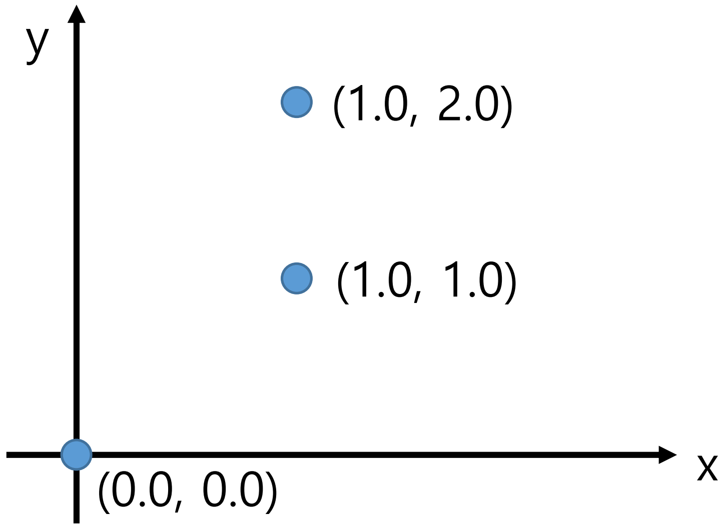
Example

- $H(0, w_0, w_1) \approx 0.0$
- $H(1, w_0, w_1) \approx 1.0$
- $H(1, w_0, w_1) \approx 2.0$



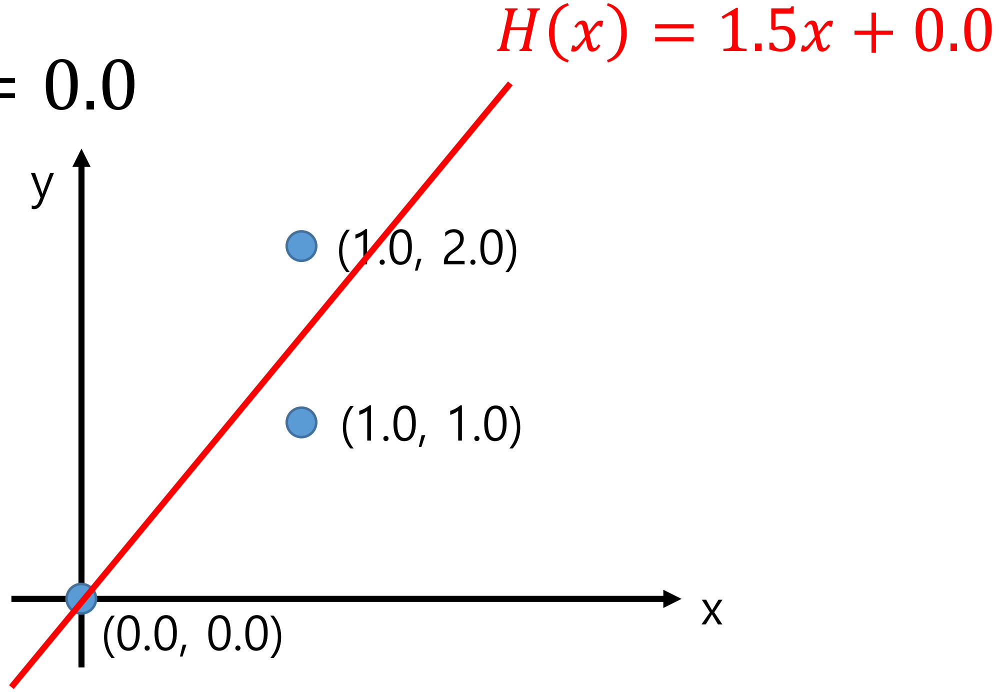
Example

- $L = \sum_i (y_i - w_1 x_i - w_0)^2$
  $= (0.0 - w_1 \cdot 0.0 - w_0)^2 + (1.0 - w_1 \cdot 1.0 - w_0)^2 + (2.0 - w_1 \cdot 1.0 - w_0)^2$
  $= 2{w_1}^2 + 3{w_0}^2 - 6w_1 - 6w_0 + 4w_1 w_0 + 5$



Example

- $\frac{\partial L}{\partial w_1} = 4w_1 + 4w_0 - 6 = 0$
- $\frac{\partial L}{\partial w_0} = 4w_1 + 6w_0 - 6 = 0$
- $\therefore\ w_1 = 1.5, w_0 = 0.0$



Multi Variable Linear Regression

- x가 scalar값(1개)가 아니라 vector가 된다면??

- Input
  - X_1 : Facebook 광고료
  - X_2 : TV 광고료
  - X_3 : 신문 광고료
- Output
  - 판매량

FB	TV	신문	판매량
X_1	X_2	X_3	Y
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	9.3
151.5	41.3	58.5	18.5
180.8	10.8	58.4	12.9
8.7	48.9	75	7.2
57.5	32.8	23.5	11.8
⋮	⋮	⋮	⋮

Multi Variable Linear Regression

- Two scenarios
  - If $X^T X$ is invertible

    Moore-Penrose Pseudoinverse

    $$\boldsymbol{w} = \quad y$$

  - If $X^T X$ is not invertible

    Pseudo-inverse defined, but no unique solution

<Note>

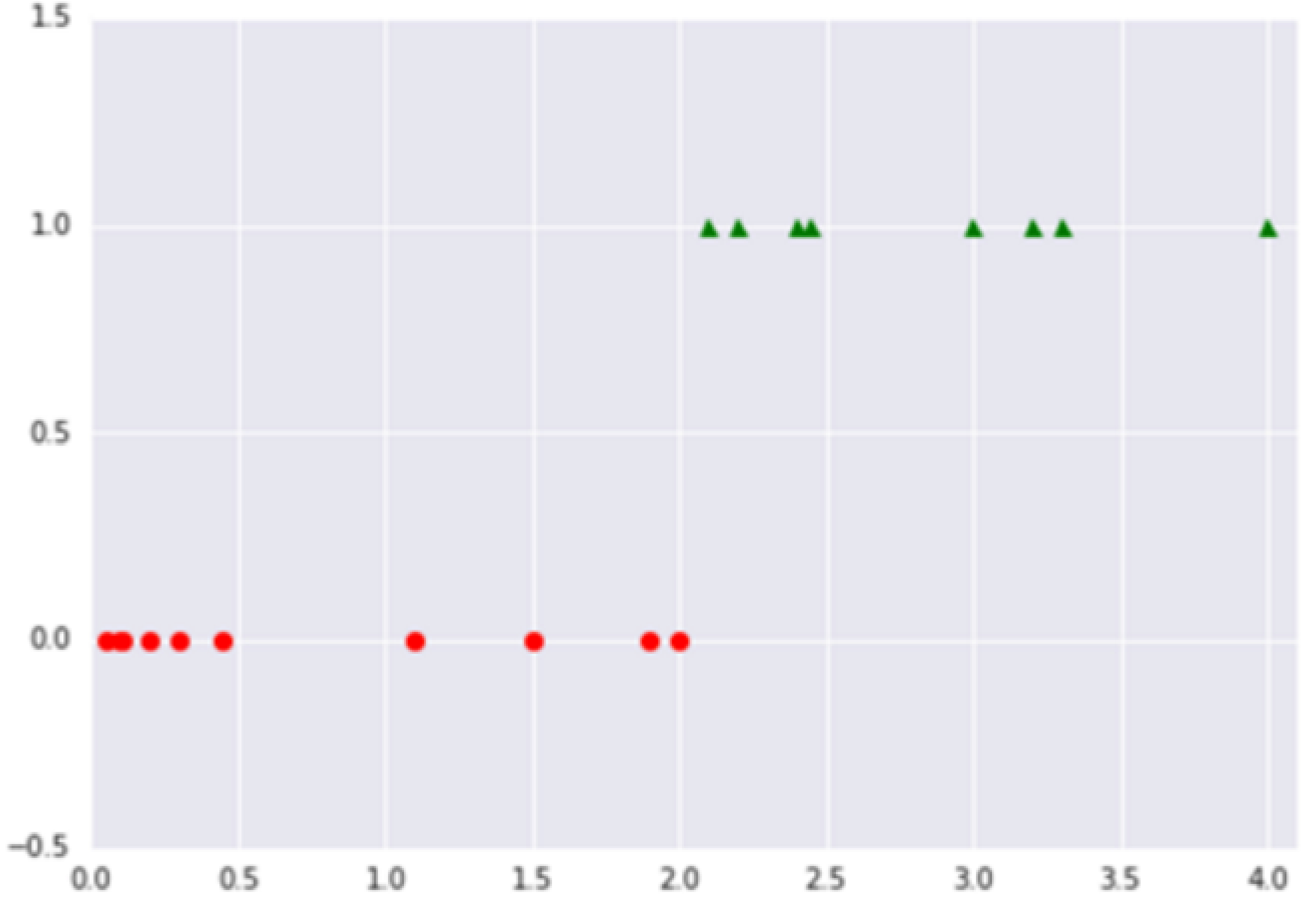
$X \in \mathbb{R}^{n \times p}$ 일 때(n: sample 수, p: input vector size),

- p가 커질수록 matrix inversion 연산량이 많아짐
- p>n 이면 $X^T X$ is not invertible

Classification도 할 수 있지 않을까요?

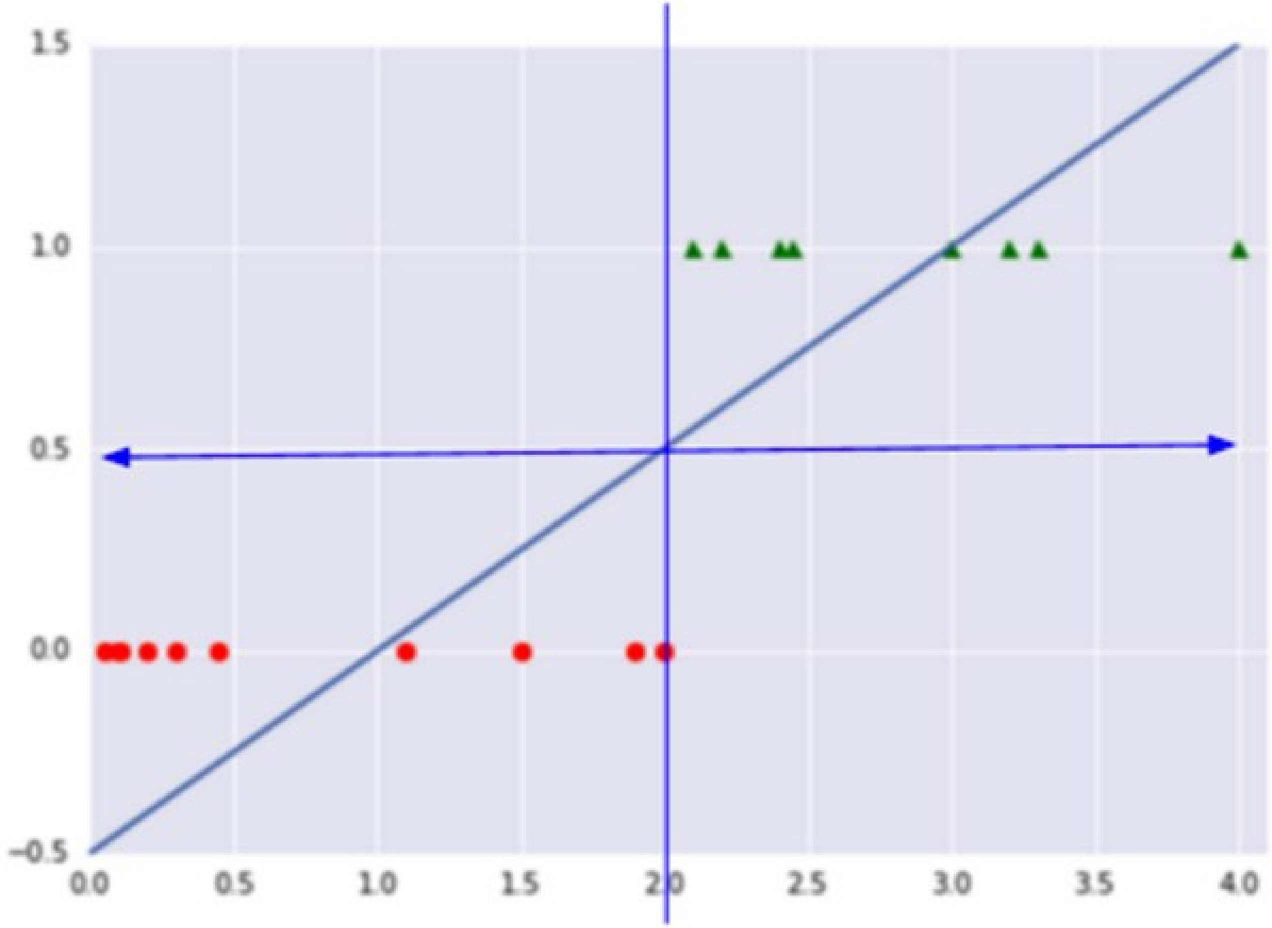
Binary Classification

- 종양의 크기에 따른 양성/음성 판별 문제
  - 1 : 양성(암), 0: 음성(정상)



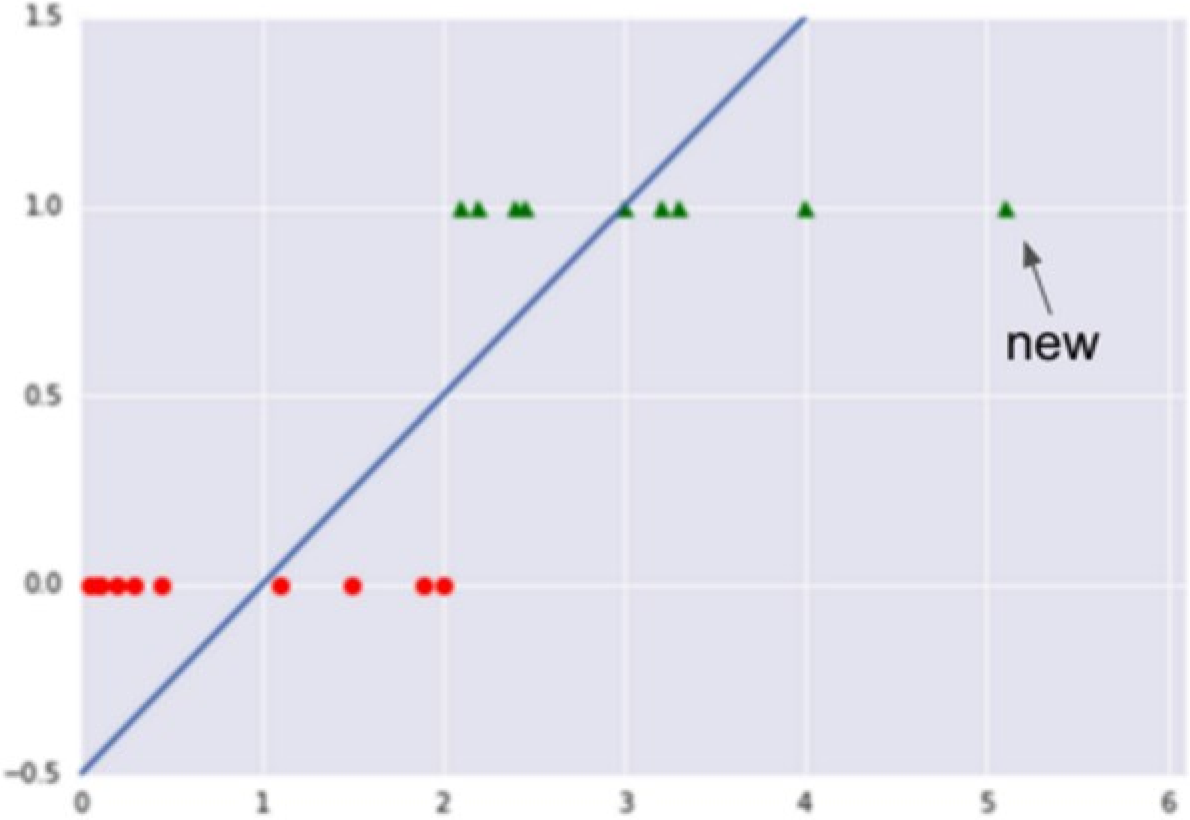
Binary Classification

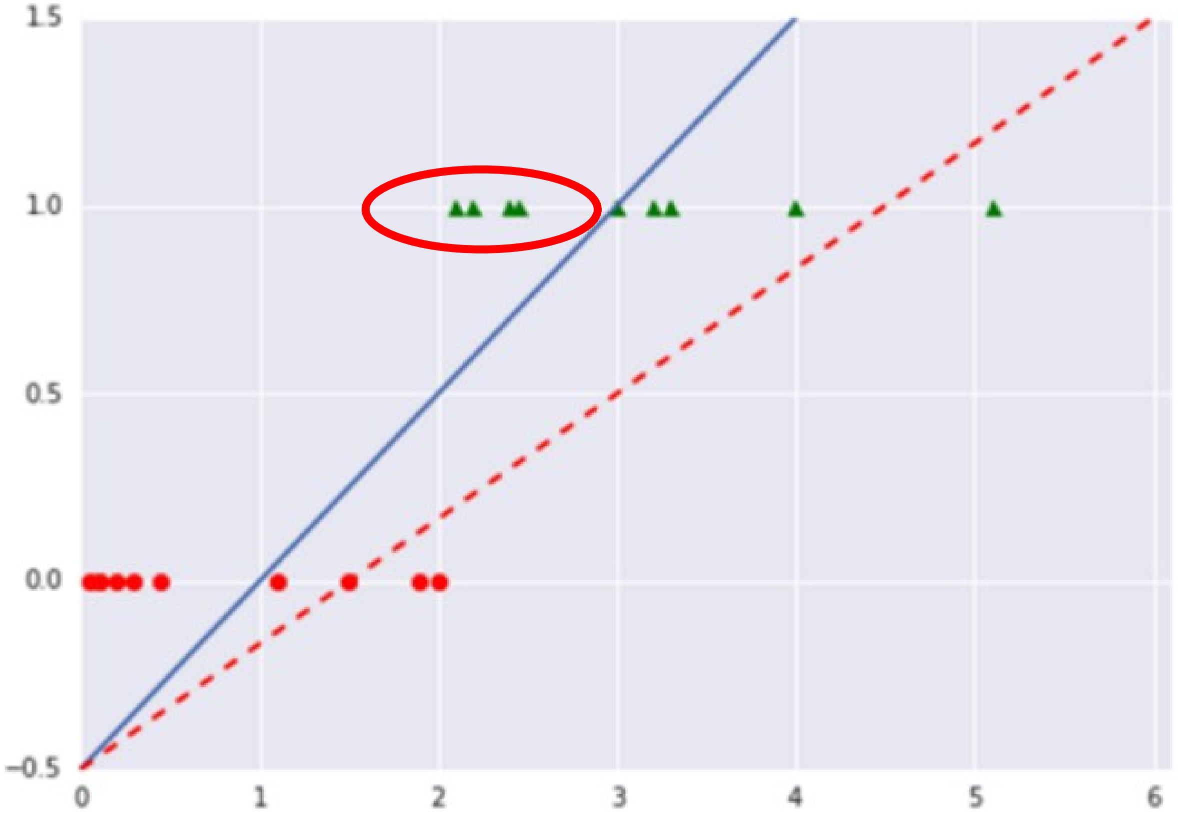
- Linear Regression으로 해봅시다
  - Regression 예측값이 0.5 이상이면 양성, 0.5 이하면 음성으로 판별



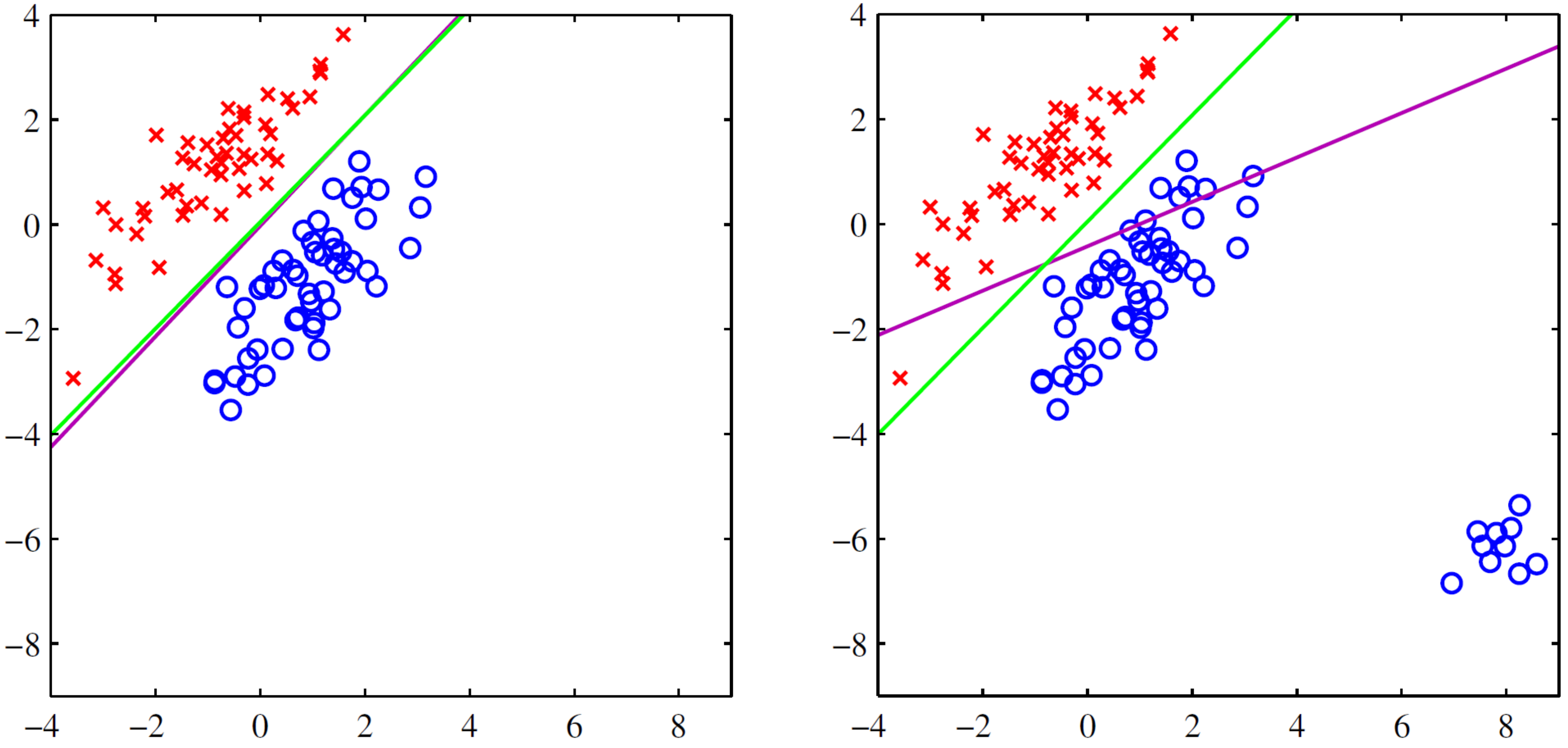
Binary Classification

- 종양의 크기가 매우 큰 data(outlier)가 추가된 경우





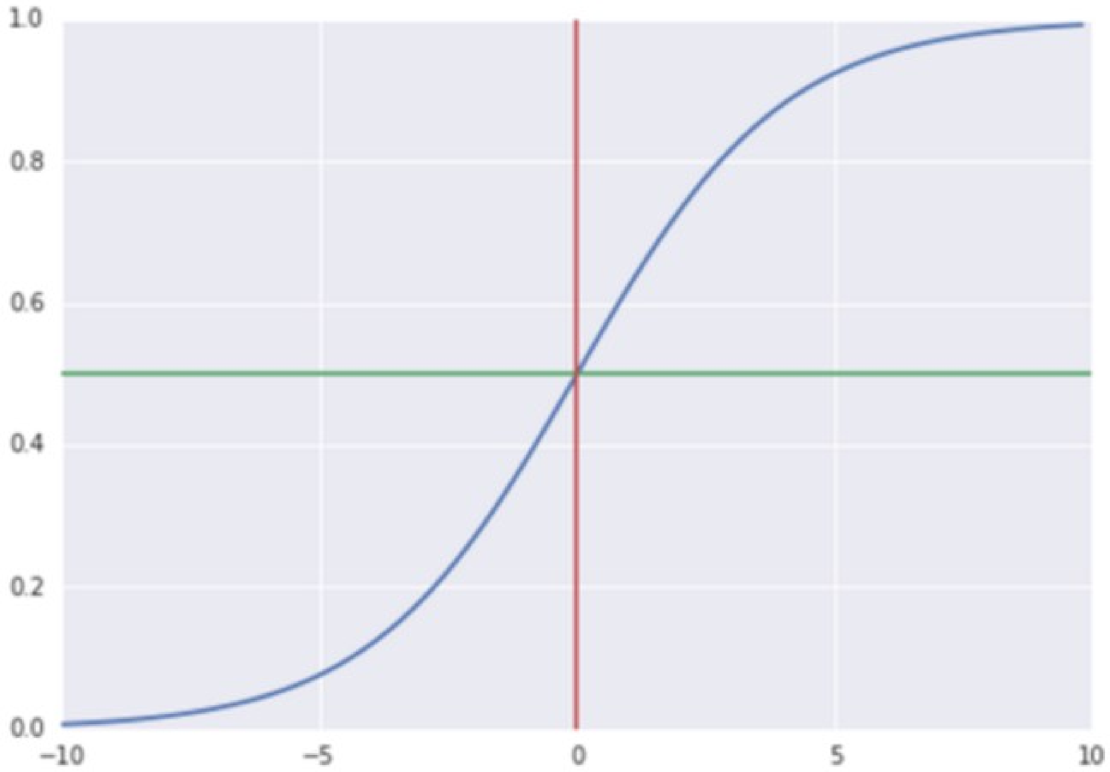
Problems of Linear Regression for Classification



Binary Classification

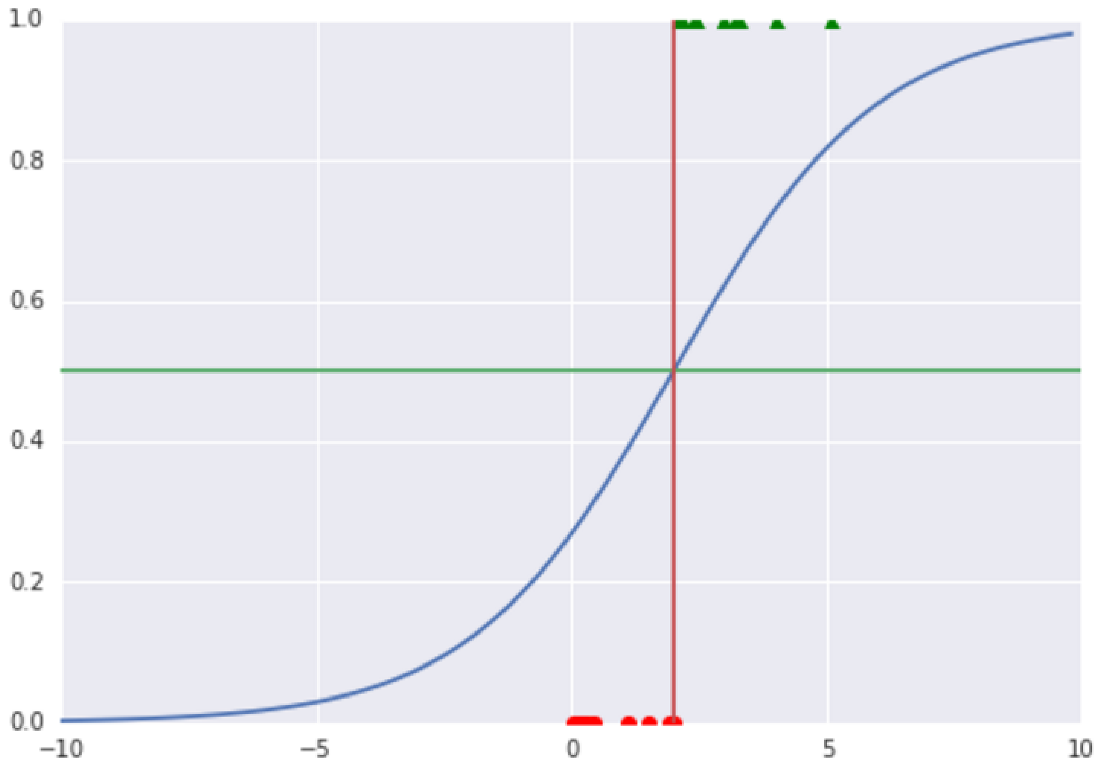
- 아주 크거나 아주 작은 data에 영향을 많이 받지 않았으면 좋겠다
- Binary classification에 맞게 0에서 1사이 값으로 나오면 좋겠다

→ Sigmoid 를 써보자



Logistic Regression

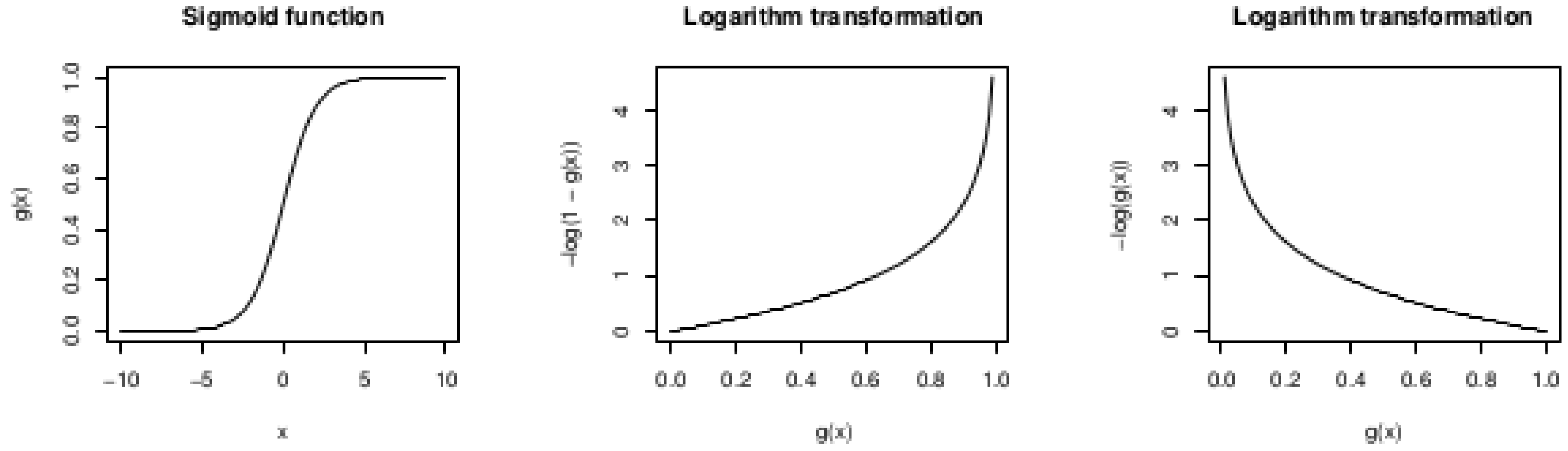
- Linear Regression 식에 Sigmoid 함수를 통과시킨 것
  - $H(x) = \sigma(\boldsymbol{w}^T\boldsymbol{x} + b) = \dfrac{1}{1+e^{-(\boldsymbol{w}^T\boldsymbol{x}+b)}} = P(y = 1|\boldsymbol{x})$



Logistic Regression

- 새로운 Cost(Loss) function을 정의 – Cross-Entropy

$$cost(W) = -\frac{1}{m}\sum ylog(H(x)) + (1-y)log(1-H(x))$$



(a) *Sigmoid function.* **(b)** *Cost for $y = 0$.* **(c)** *Cost for $y = 1$.*

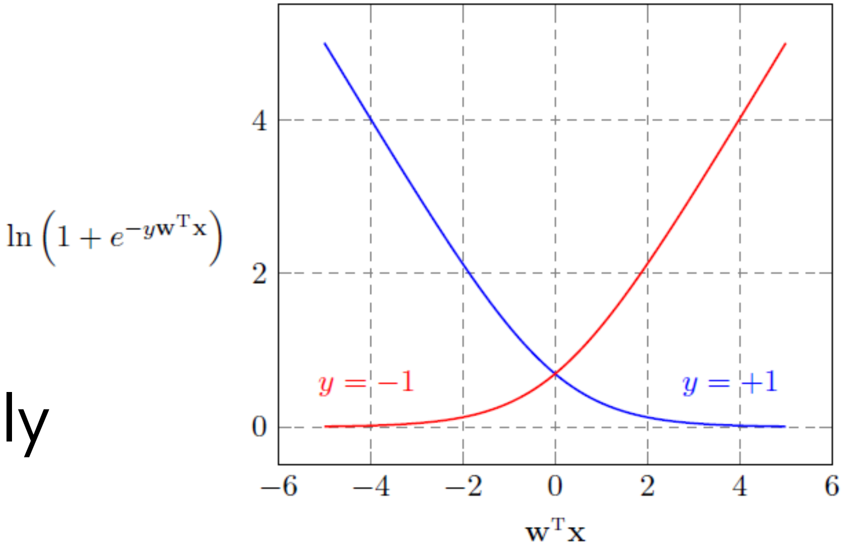
Figure B.1: *Logarithmic transformation of the sigmoid function.*

Minimizing NLL

$$\mathbf{e}(h(\mathbf{x}_n), y_n) = \ln\left(1 + e^{-y_n \mathbf{w}^\top \mathbf{x}_n}\right)$$

- We can define loss(error) function as below

$$\mathrm{E}_{\mathrm{in}}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N} \log\left(1 + e^{-y_n \mathbf{w}^\top \mathbf{x}_n}\right)$$



- Unfortunately, not easy to manipulate analytically

$$\nabla \mathrm{E}_{\mathrm{in}}(\mathbf{w}) = -\frac{1}{N}\sum_{n=1}^{N} \frac{y_n \mathbf{x}_n}{1 + e^{y_n \mathbf{w}^\mathrm{T} \mathbf{x}_n}}$$

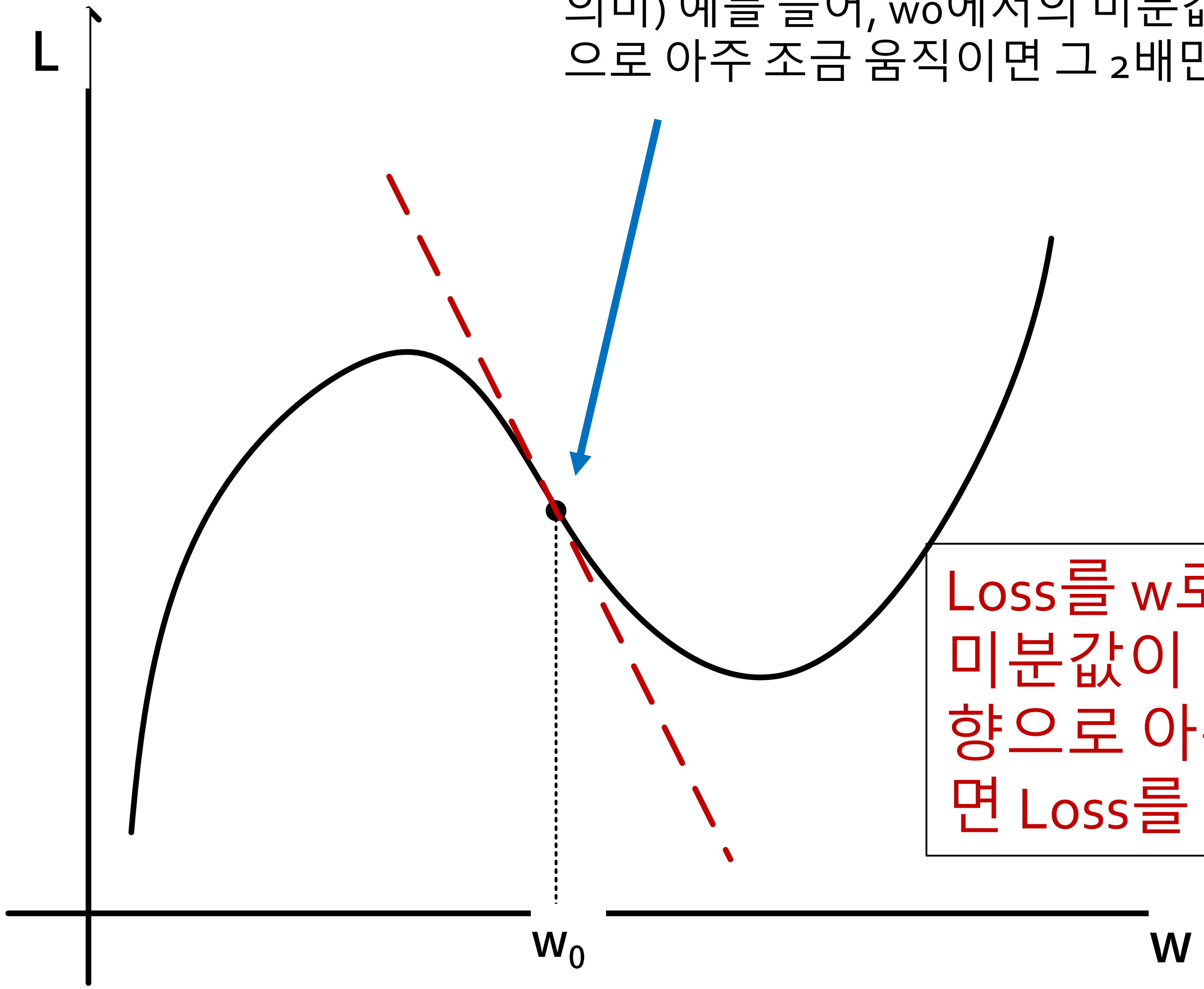
$$= \frac{1}{N}\sum_{n=1}^{N} -y_n \mathbf{x}_n \theta(-y_n \mathbf{w}^\mathrm{T} \mathbf{x}_n)$$

- We need iterative optimization
- Use 미분!

미분??

$w=w_0$에서의 미분값
= 이 점에서의 접선의 기울기

의미) 예를 들어, wo에서의 미분값이 -2라면, w를 wo에서 왼쪽(-방향)으로 아주 조금 움직이면 그 2배만큼 L 값이 증가한다는 뜻!



Loss를 w로 미분하고,
미분값이 가리키는 방향의 반대방향으로 아주 조금씩 w를 바꿔나가면 Loss를 감소시킬 수 있다!!

Gradient Descent

Loss Function의 미분(Gradient)를 이용하여 weight를 update하는 방법

$$w_{new} = w_{old} - \eta \nabla_w L$$

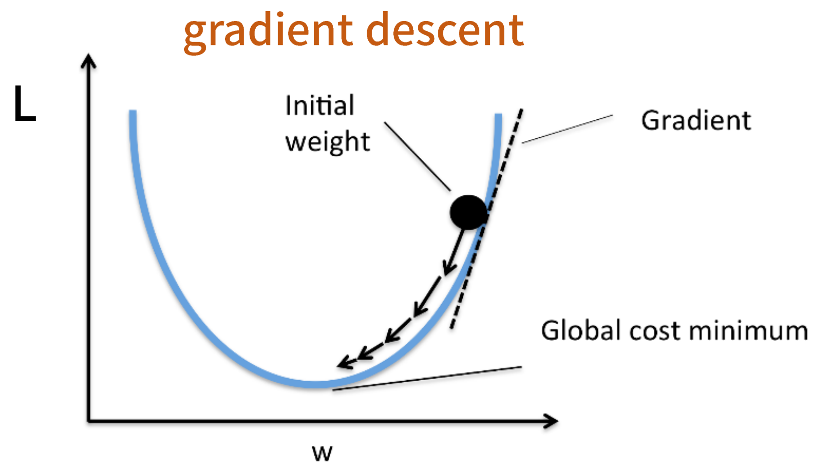
Weight update $= w_{new} - w_{old}$

$$= - \eta \nabla_w L$$

Loss를 감소
시키는 방향
(Descent)

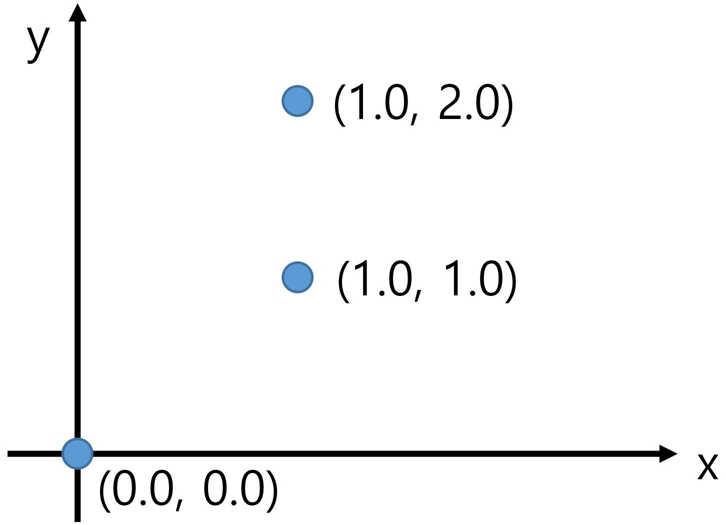
아주 조금씩 이동
(Learning Rate)

미분값
(Gradient)



Linear Regression Again

- Find the linear function(f) that best describes the given data
  - $H(x, w_0, w_1) = w_1 x + w_0$



$$L = \sum_i (y_i - w_1 x_i - w_0)^2$$

$$= (0.0 - w_1 \cdot 0.0 - w_0)^2 + (1.0 - w_1 \cdot 1.0 - w_0)^2 + (2.0 - w_1 \cdot 1.0 - w_0)^2$$

$$= 2{w_1}^2 + 3{w_0}^2 - 6w_1 - 6w_0 + 4w_1 w_0 + 5$$

Solving Linear Regression Using GD

- Choose a small value for η such as $\eta = 0.1$
- Randomly select $w_0^0 = 1, w_1^0 = 1$, initially
- Repeat

$$w_0^{t+1} = w_0^t - \eta(4w_1^t + 6w_0^t - 6)$$
$$w_1^{t+1} = w_1^t - \eta(4w_1^t + 4w_0^t - 6)$$

$$\boxed{\begin{aligned} \frac{\partial L}{\partial w_1} &= 4w_1 + 4w_0 - 6 = 0 \\ \frac{\partial L}{\partial w_0} &= 4w_1 + 6w_0 - 6 = 0 \end{aligned}}$$

Solving Linear Regression Using GD

$$w_0^0 = 1$$
$$w_1^0 = 1$$

$$w_0^1 = 1 - 0.1(4 \times 1 + 6 \times 1 - 6) = 0.6$$
$$w_1^1 = 1 - 0.1(4 \times 1 + 4 \times 1 - 6) = 0.8$$

$$w_0^2 = 0.6 - 0.1(4 \times 0.8 + 6 \times 0.6 - 6) = 0.54$$
$$w_1^2 = 0.8 - 0.1(4 \times 0.8 + 4 \times 0.6 - 6) = 0.84$$

$$w_0^3 = 0.54 - 0.1(4 \times 0.84 + 6 \times 0.54 - 6) = 0.480$$
$$w_1^3 = 0.84 - 0.1(4 \times 0.84 + 4 \times 0.54 - 6) = 0.888$$

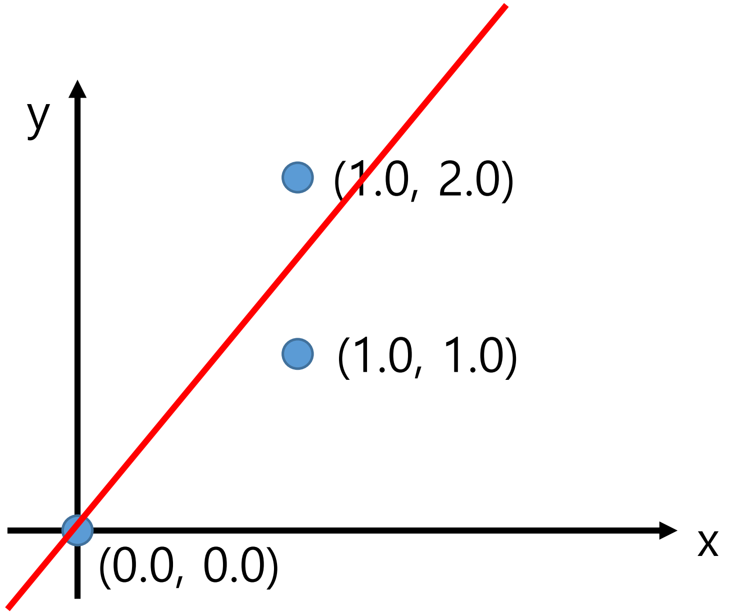
Solving Linear Regression Using GD

$$w_0^4 = 0.480 - 0.1(4 \times 0.888 + 6 \times 0.480 - 6) = 0.4368$$
$$w_1^4 = 0.888 - 0.1(4 \times 0.888 + 4 \times 0.480 - 6) = 0.9408$$

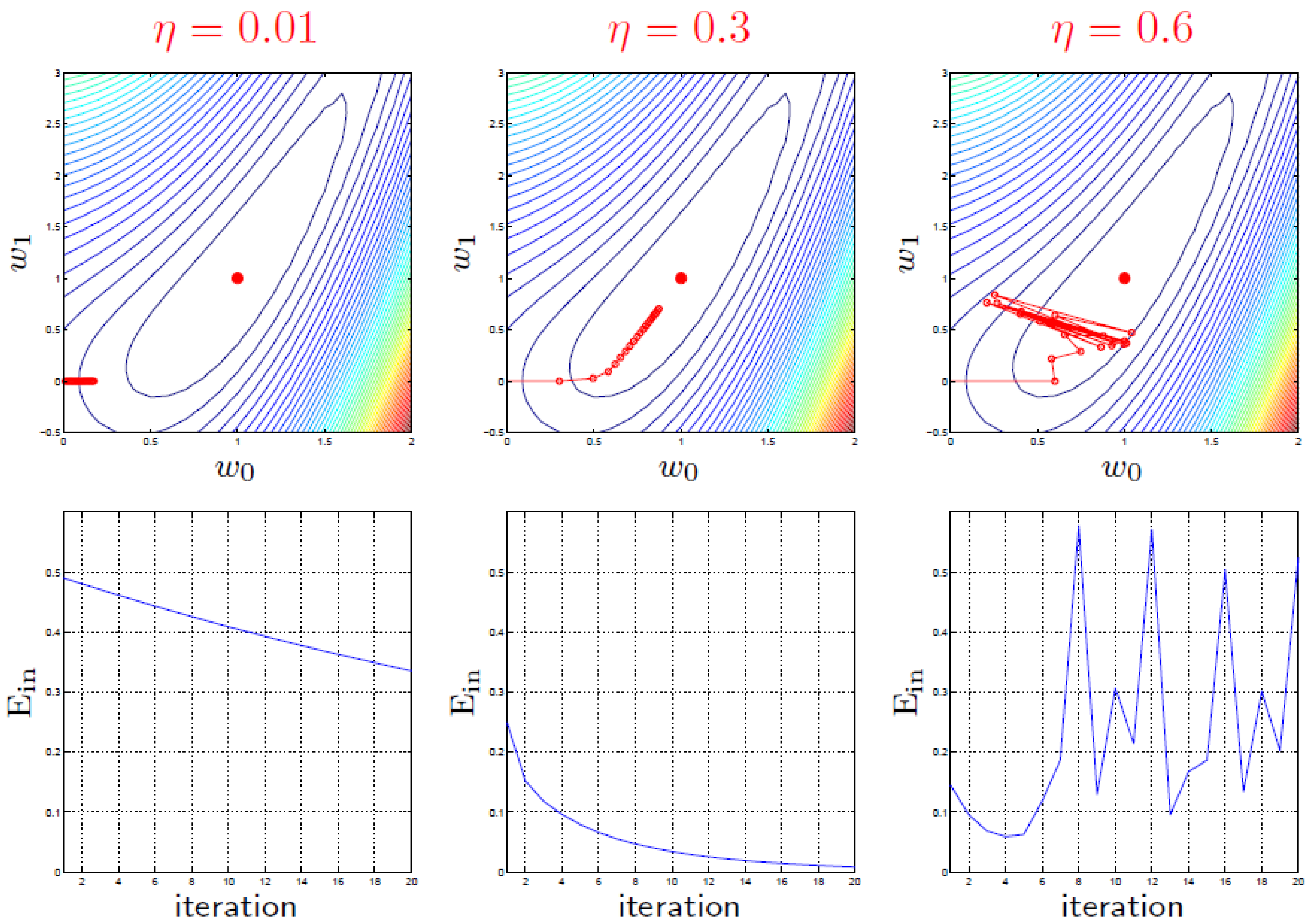
…

$$H(x) = 1.49989171x + 0.00007713$$

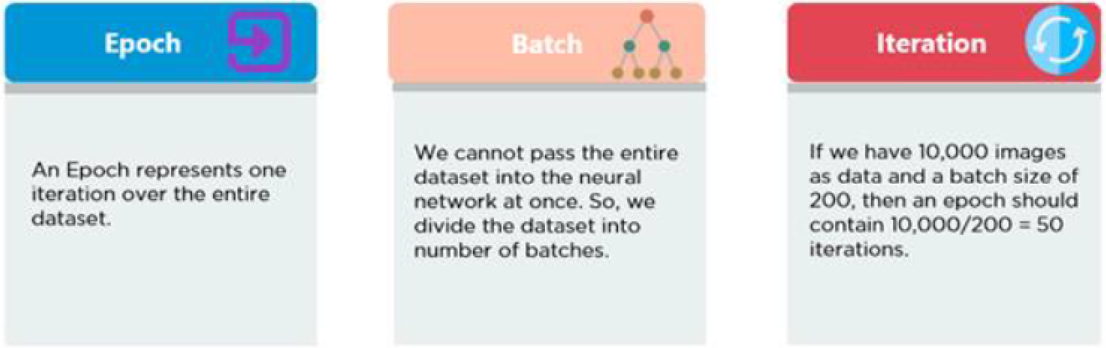
$$w_0^{100} = 0.00007713$$
$$w_1^{100} = 1.49989171$$



Learning Rate



Learning Rate & Mini-Batch



Mini-Batch size: Number of training instances the network evaluates per weight update step.

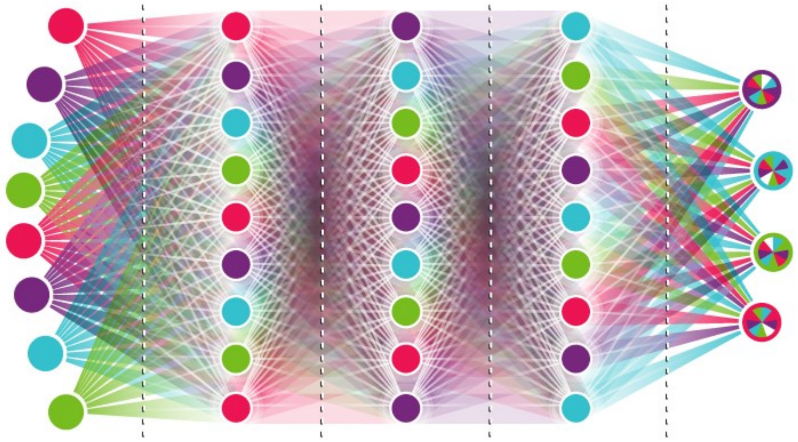
- Larger batch size = more computational speed
- Smaller batch size = (empirically) better generalization

"Training with large minibatches is bad for your health. More importantly, it's bad for your test error. Friends don't let friends use minibatches larger than 32."
- Yann LeCun

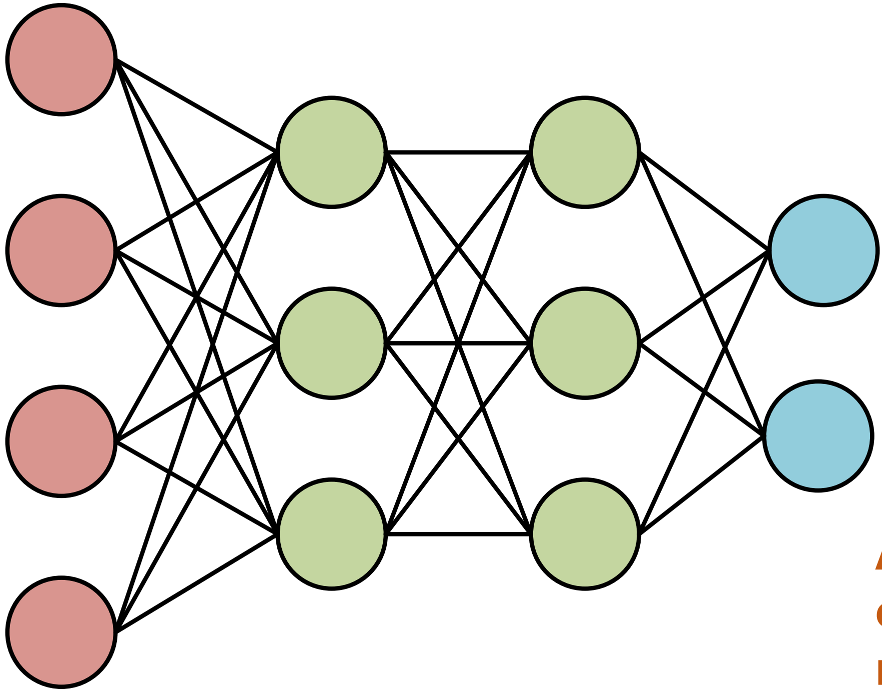
Revisiting Small Batch Training for Deep Neural Networks (2018)

"when increasing the batch size, a linear increase of the learning rate η with the batch size m is required to keep the mean SGD weight update per training example constant"

Multi-Layer Perceptron



Multi-Layer Perceptron



A many-layer network of perceptrons can engage in sophisticated decision making.

Network을 deep하게 쌓고, class도 여러 개일 때는
어떻게 학습할 수 있을까?

먼저 Multi-Layer부터 생각해봅시다

미분을 계산해봅시다!

$$z11 = x1 \cdot w11 + x2 \cdot w12 + x3 \cdot w13 + x4 \cdot w14$$

$$a11 = \sigma(z11) = \frac{1}{1 + e^{-z11}}$$

$$z2 = a11 \cdot w21 + a12 \cdot w22 + a13 \cdot w23$$

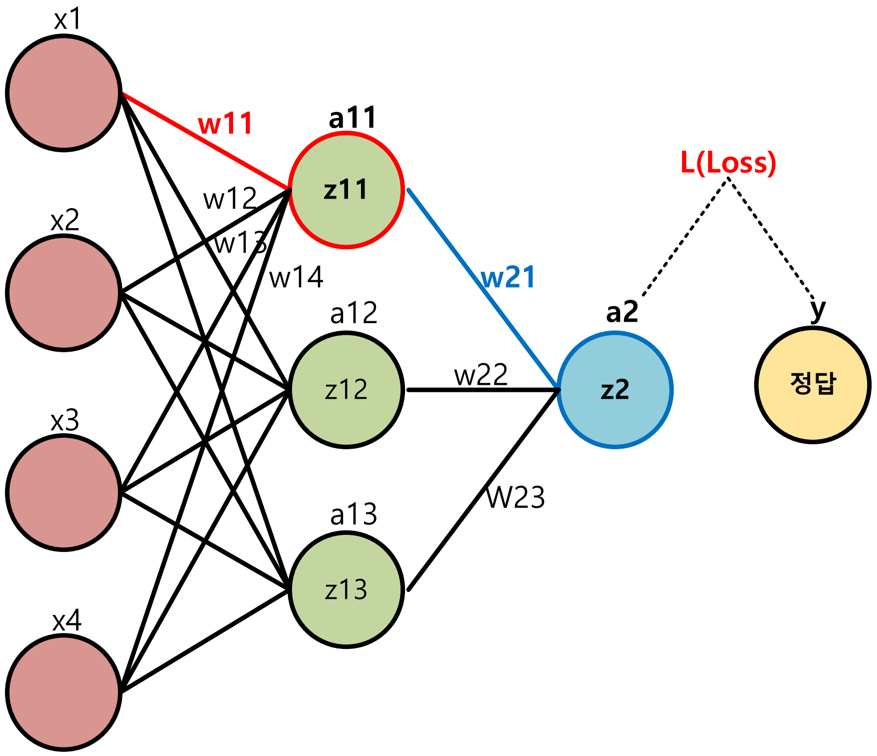
$$a2 = z2$$

$$L = (y - a2)^2$$

일 때,

w11을 update 하기 위해 필요한 미분값

$$\frac{\partial L}{\partial w11} = ??????$$



Back Propagation

$$z11 = x1 \cdot w11 + x2 \cdot w12 + x3 \cdot w13 + x4 \cdot w14$$
$$a11 = \sigma(z11) = \frac{1}{1 + e^{-z11}}$$
$$z2 = a11 \cdot w21 + a12 \cdot w22 + a13 \cdot w23$$
$$a2 = z2$$
$$L = (y - a2)^2$$

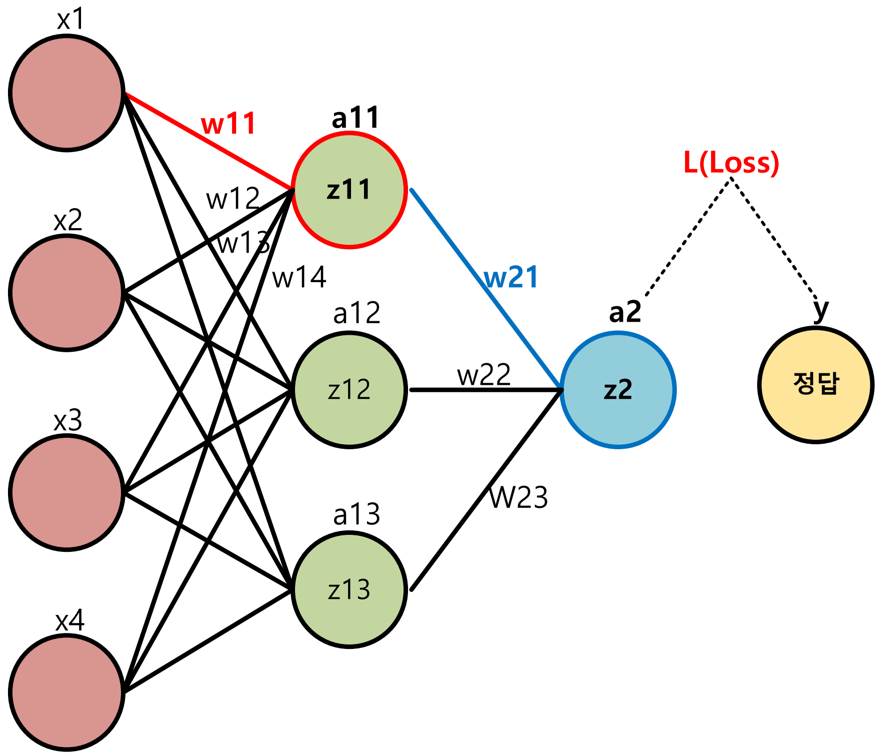
Loss부터 거꾸로 한 단계씩 미분을 해봅시다

$$\partial L / \partial a2 = -2(y - a2)$$
$$\partial a2 / \partial z2 = 1$$
$$\partial z2 / \partial a11 = w21$$
$$\partial a11 / \partial z11 = \sigma(z11) \cdot \big(1 - \sigma(z11)\big)$$
$$\partial z11 / \partial w11 = x1$$

이 미분들을 전부 각각 곱하면(chain rule),

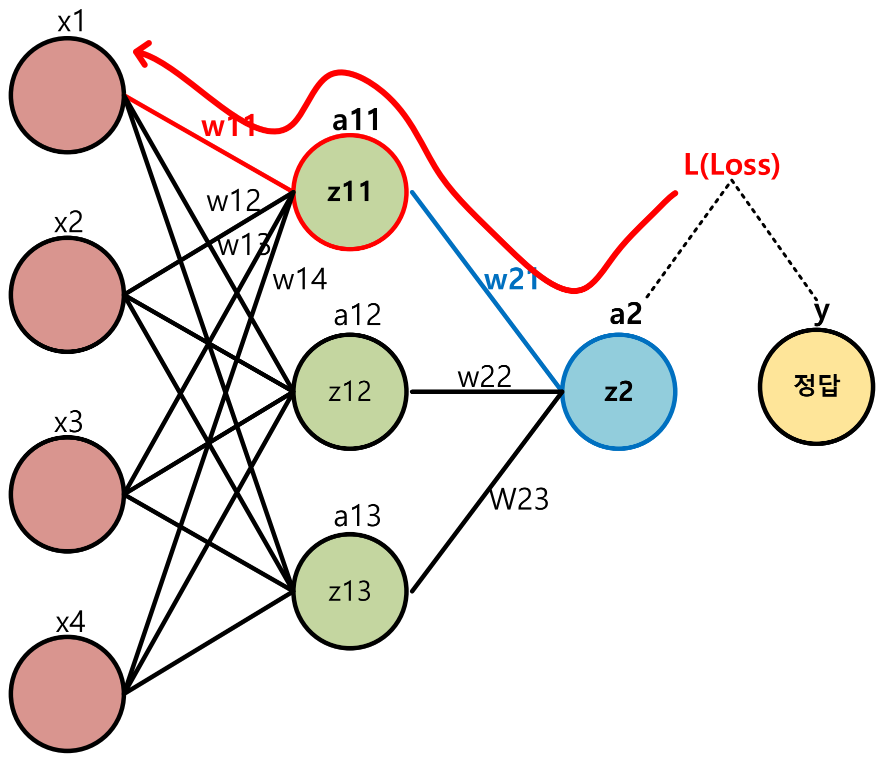
$$\frac{\partial L}{\partial a2} \cdot \frac{\partial a2}{\partial z2} \cdot \frac{\partial z2}{\partial a11} \cdot \frac{\partial a11}{\partial z11} \cdot \frac{\partial z11}{\partial w11} = \boldsymbol{\frac{\partial L}{\partial w11}}$$

우리가 구하려고 했던 미분값



Back Propagation

Loss로부터 거꾸로 한 단계씩 미분 값을 구하고 이 값들을 chain rule 에 의하여 곱해가면서 weight에 대한 gradient를 구하는 방법

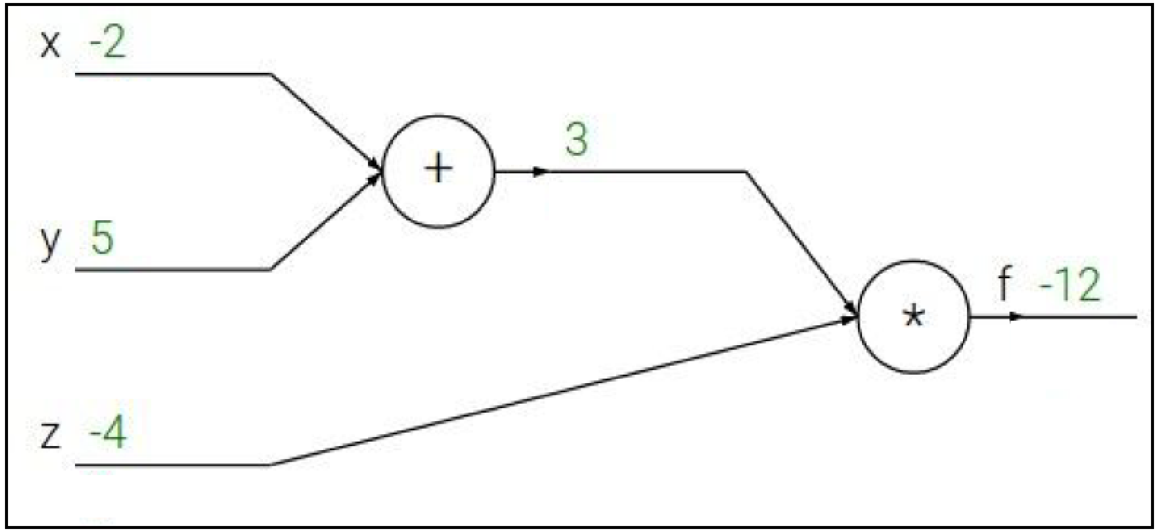


Back Propagation

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



Back Propagation

Backpropagation: a simple example

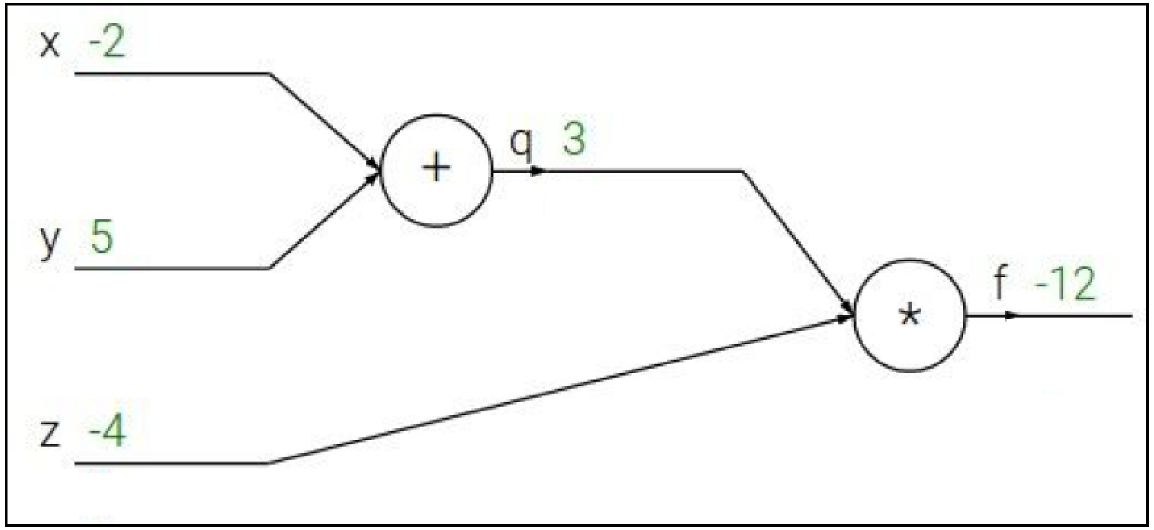
$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$



Back Propagation

Backpropagation: a simple example

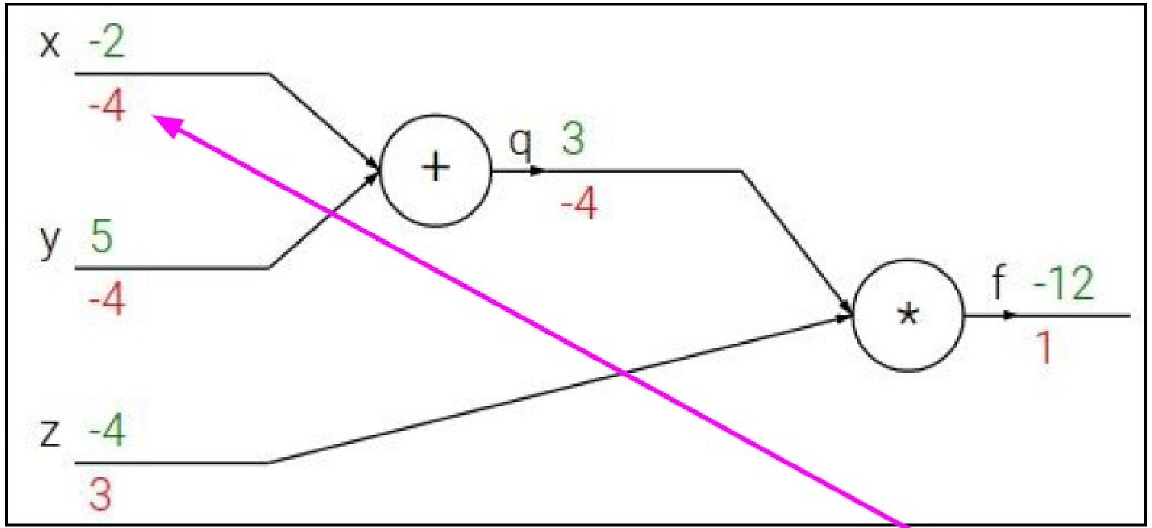
$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

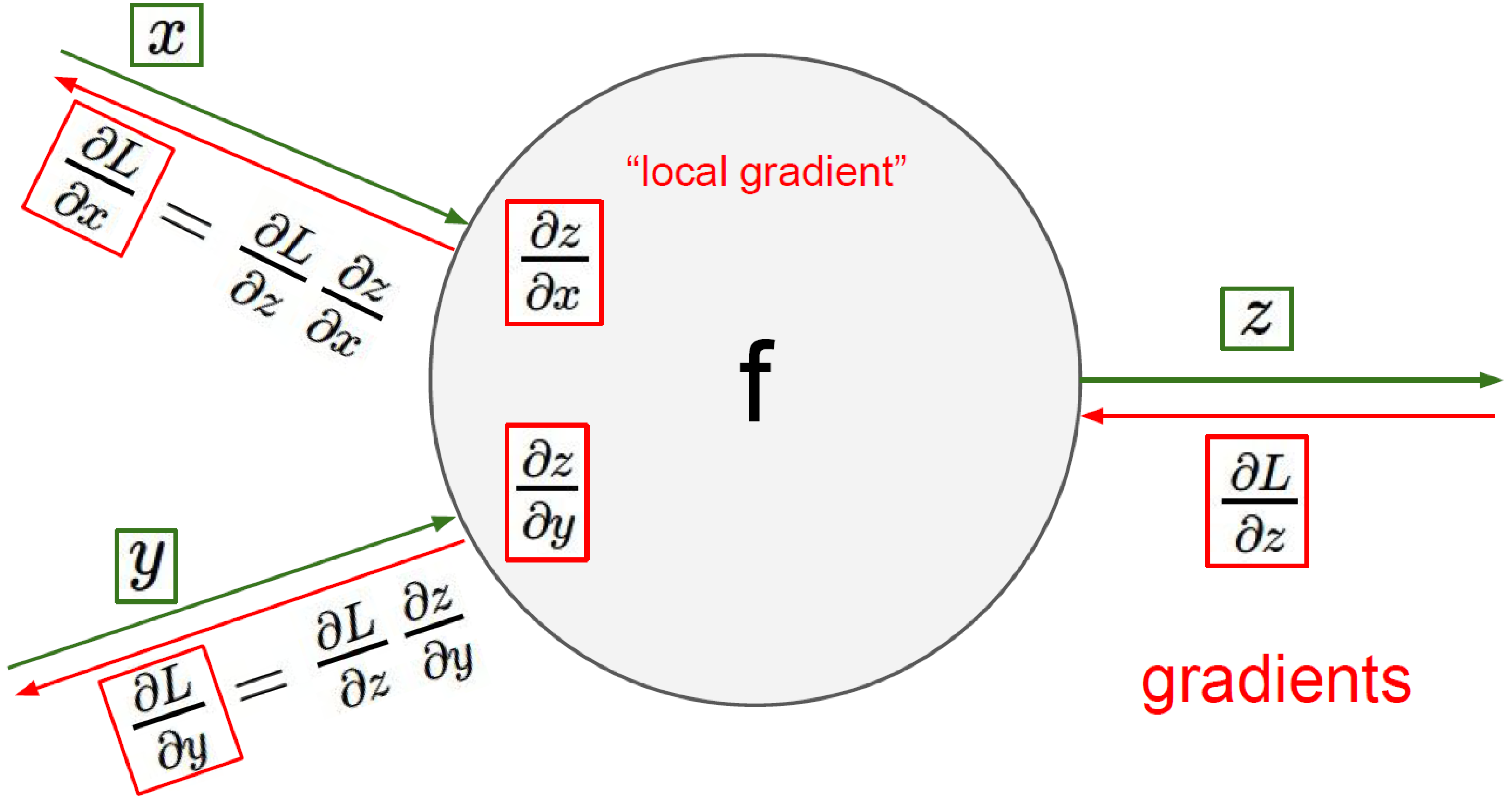


$$\frac{\partial f}{\partial x}$$

Chain rule:

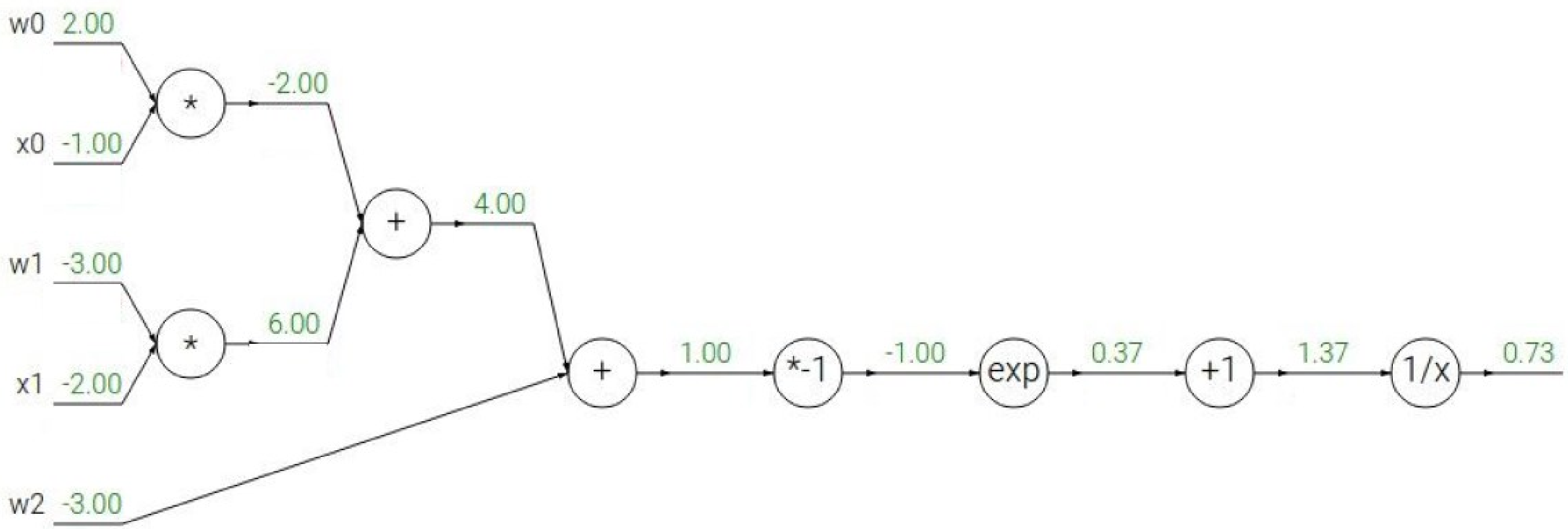
$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial q} \frac{\partial q}{\partial x}$$

Chain Rule(Local Gradient)



Back Propagation(Example)

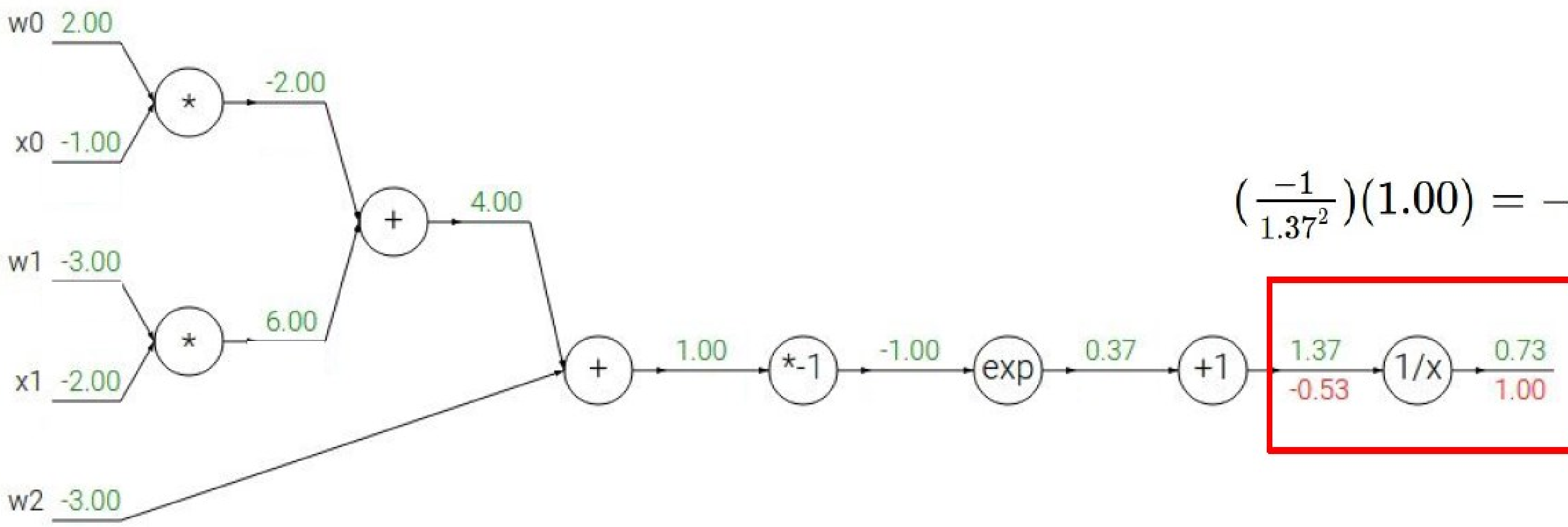
Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



Back Propagation(Example)

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$

$$\left(\frac{-1}{1.37^2}\right)(1.00) = -0.53$$



$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$$

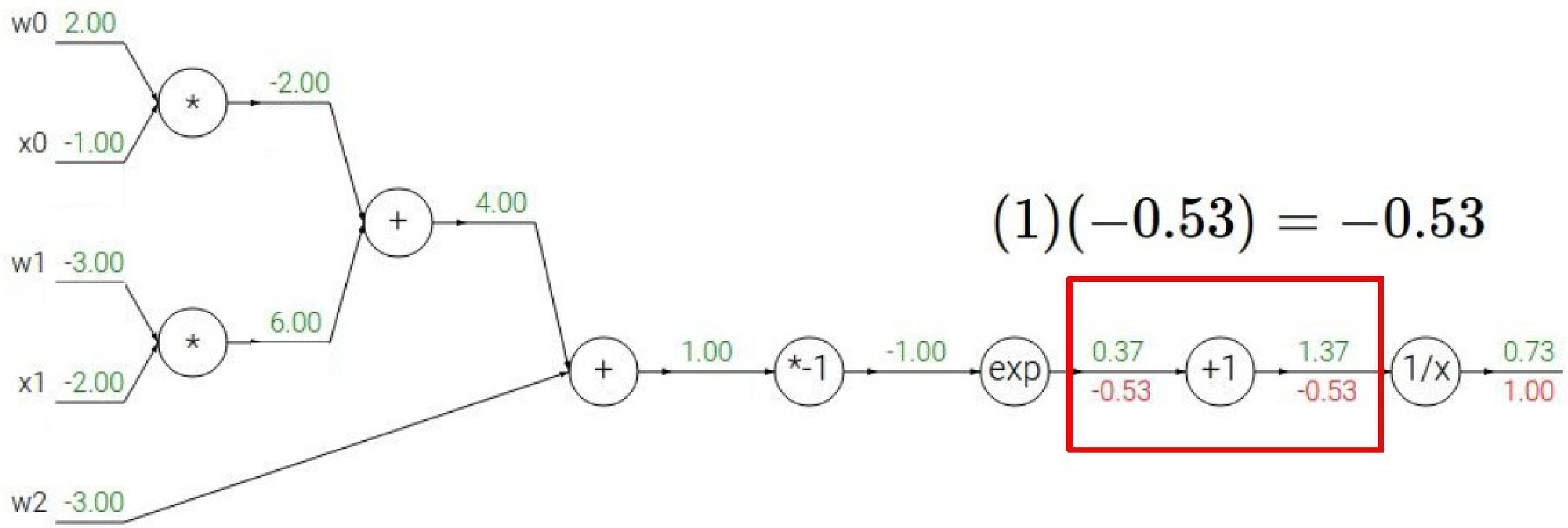
$$\boxed{f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2}$$

$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Back Propagation(Example)

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$$

$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

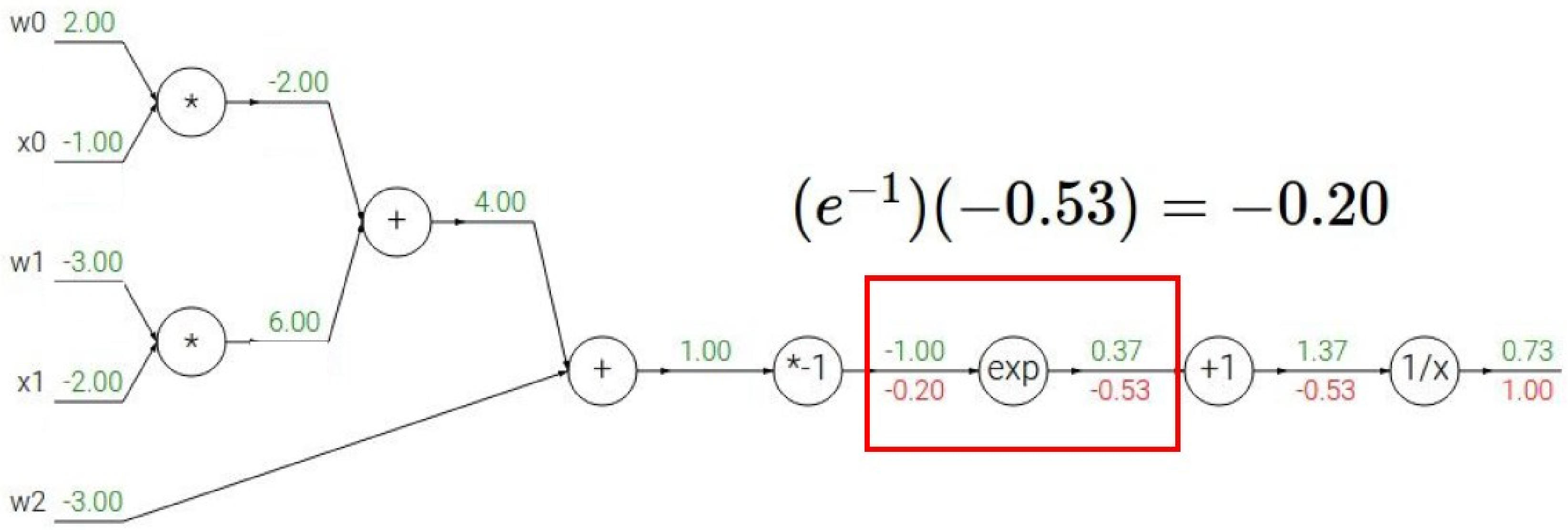
$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$\boxed{f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1}$$

Back Propagation(Example)

Another example: $f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$

$$(e^{-1})(-0.53) = -0.20$$



$$\boxed{f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x}$$

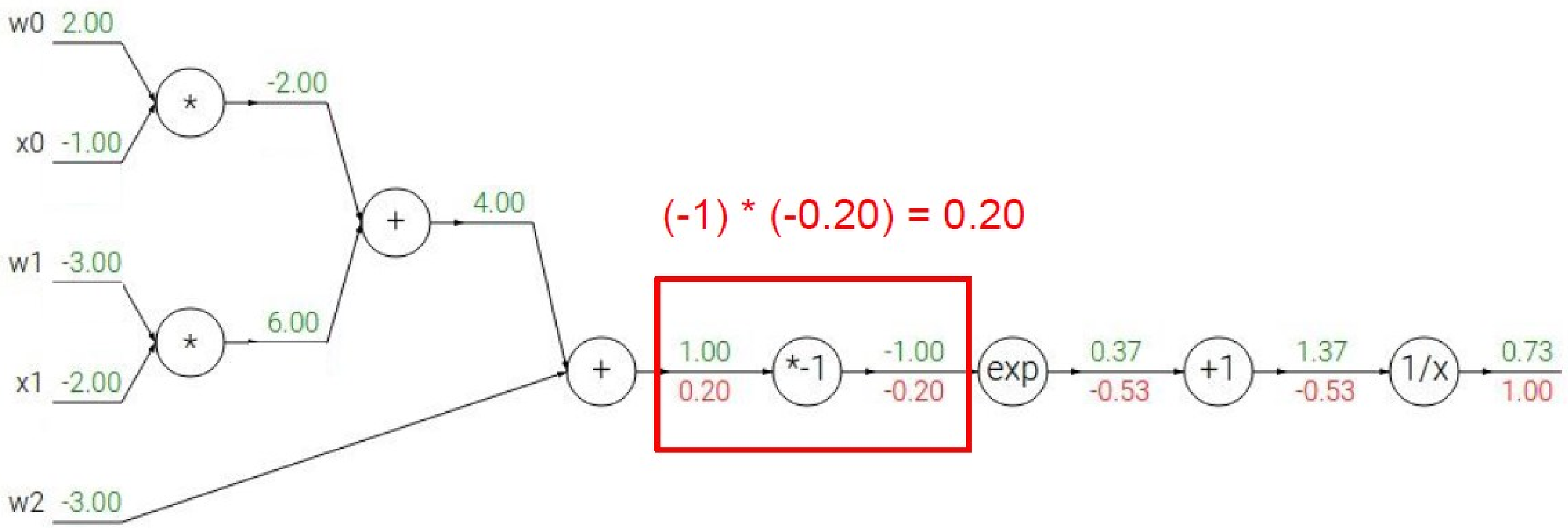
$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Back Propagation(Example)

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$$

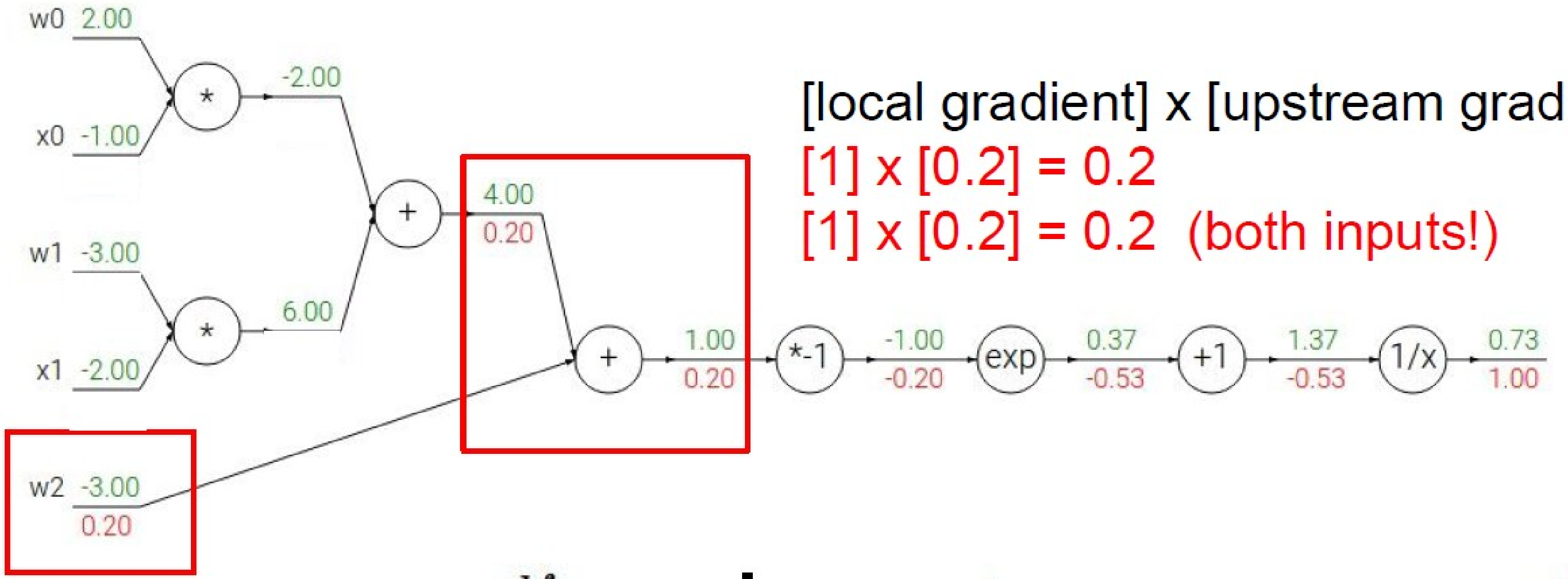
$$\boxed{f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a}$$

$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Back Propagation(Example)

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



[local gradient] x [upstream gradient]

[1] x [0.2] = 0.2

[1] x [0.2] = 0.2 (both inputs!)

$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$$

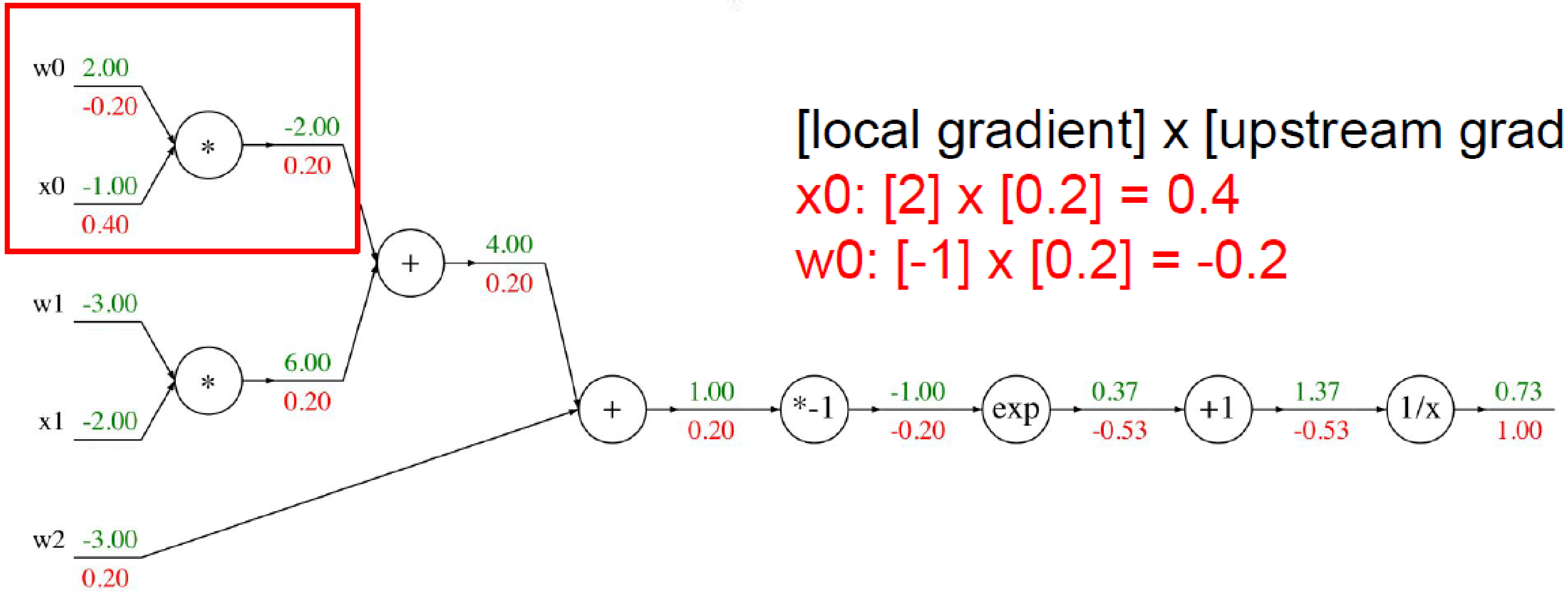
$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Back Propagation(Example)

Another example: $f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$



[local gradient] x [upstream gradient]

x0: [2] x [0.2] = 0.4

w0: [-1] x [0.2] = -0.2

$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$

$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$

$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$

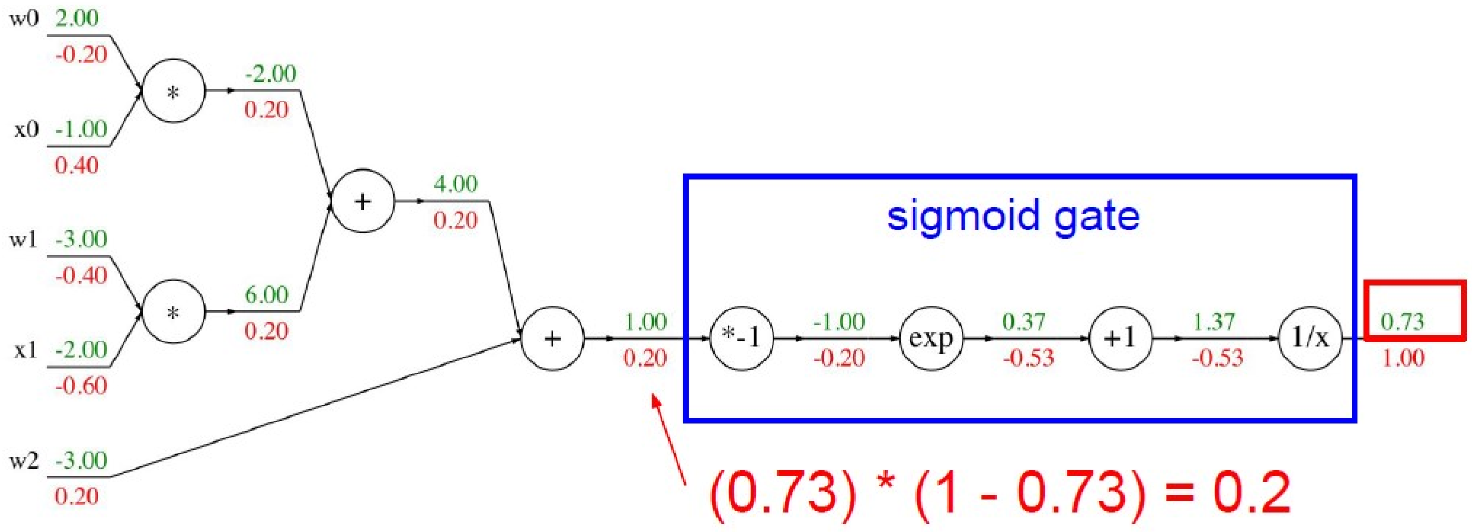
$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$

Back Propagation(Example)

$$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$

$$\boxed{\sigma(x) = \frac{1}{1+e^{-x}}}$$ sigmoid function

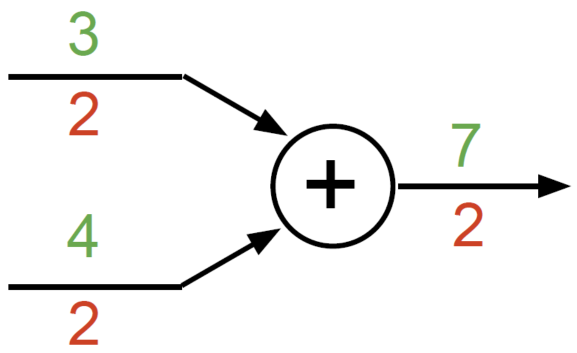
$$\frac{d\sigma(x)}{dx} = \frac{e^{-x}}{(1+e^{-x})^2} = \left(\frac{1+e^{-x}-1}{1+e^{-x}}\right)\left(\frac{1}{1+e^{-x}}\right) = (1-\sigma(x))\,\sigma(x)$$



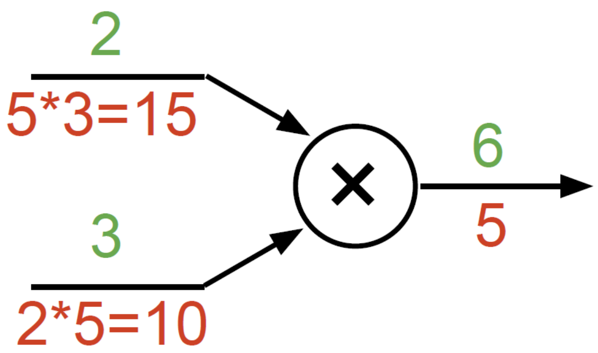
Back Propagation(Example)

Patterns in gradient flow

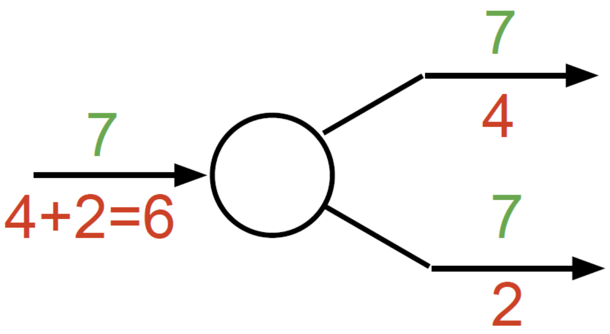
add gate: gradient distributor



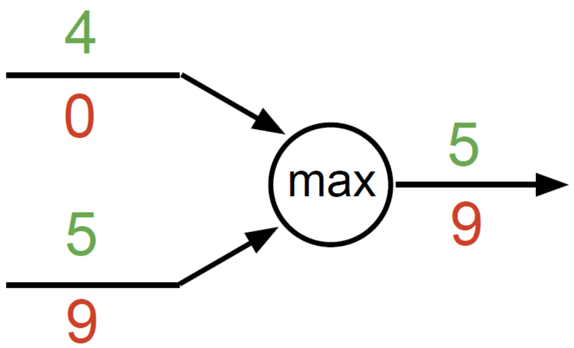
mul gate: “swap multiplier”



copy gate: gradient adder



max gate: gradient router

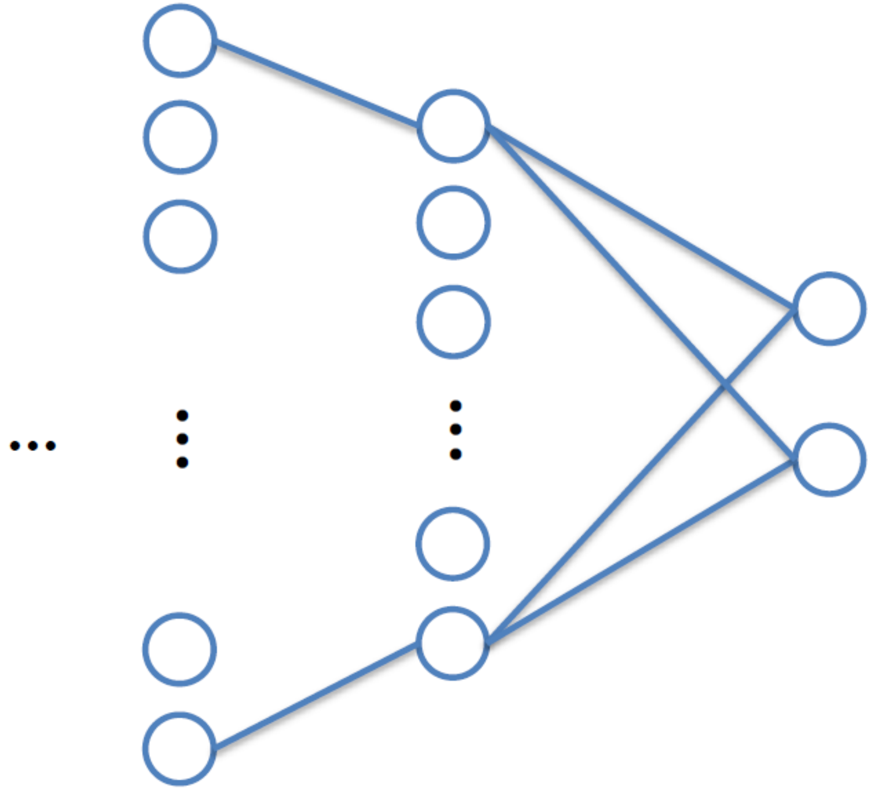


이제 Multi-Layer인 Network도 학습하는 방법은 알았는데, 그럼 Multi-Class는?

Softmax

- Output을 확률처럼 나타내보자

$$P(y=j \mid \mathbf{x}) = \frac{e^{\mathbf{x}^\top \mathbf{w}_j}}{\sum_{k=1}^{K} e^{\mathbf{x}^\top \mathbf{w}_k}}$$



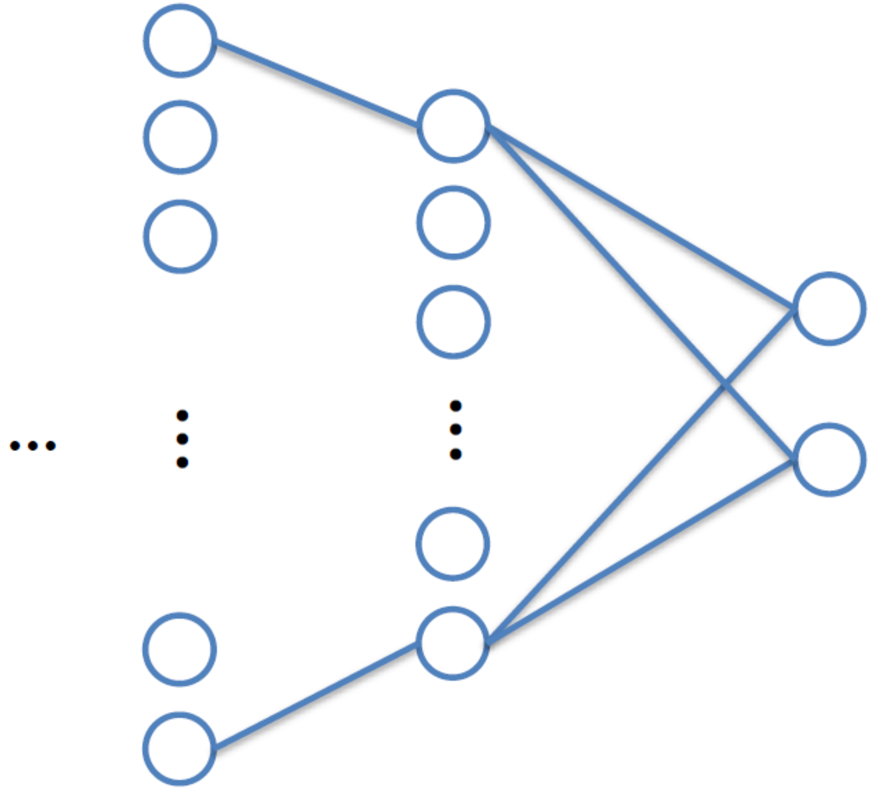
	$\mathbf{x}^\top \mathbf{w}_k$	$e^{\mathbf{x}^\top \mathbf{w}_k}$	$\frac{e^{\mathbf{x}^\top \mathbf{w}_j}}{\sum_{k=1}^{K} e^{\mathbf{x}^\top \mathbf{w}_k}}$	one-hot vector
강아지	0.37	1.4477	0.2296	0
고양이	1.58	4.8549	0.7704	1

수치 → exp → 확률 ↔ 확률

softmax!

Loss Function of Softmax

- Loss function은 어떻게 정의할까?
  - L1 loss or L2 loss(MSE)?



	$\mathbf{x}^\top \mathbf{w}_k$	$e^{\mathbf{x}^\top \mathbf{w}_k}$	$\frac{e^{\mathbf{x}^\top \mathbf{w}_j}}{\sum_{k=1}^{K} e^{\mathbf{x}^\top \mathbf{w}_k}}$	one-hot vector
강아지	0.37	1.4477	0.2296	0
고양이	1.58	4.8549	0.7704	1

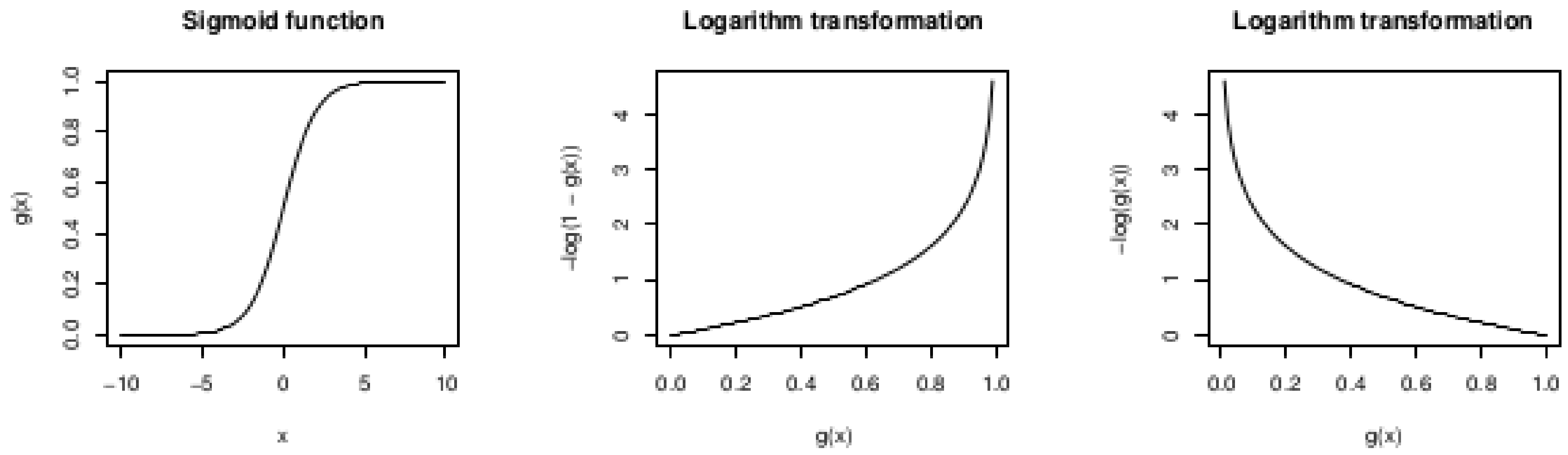
수치 → exp → 확률 ↔ 확률

softmax!

Loss Function of Logistic Regression

- Cross Entropy Loss!

$$\text{cost}(W) = -\frac{1}{m}\sum y\log(H(x)) + (1-y)\log(1-H(x))$$

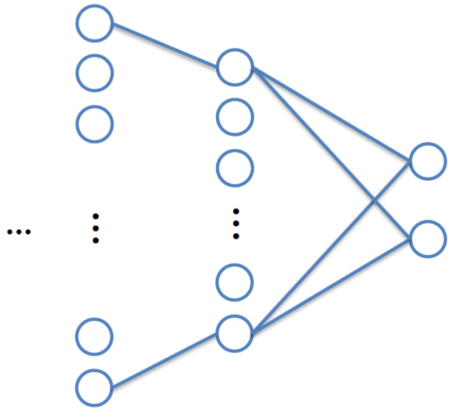


(a) *Sigmoid function.* **(b)** *Cost for $y = 0$.* **(c)** *Cost for $y = 1$.*

Figure B.1: *Logarithmic transformation of the sigmoid function.*

Loss Function – Classification

- 마지막 layer의 activation function
  - Softmax $P(y=j \mid \mathbf{x}) = \dfrac{e^{\mathbf{x}^\top \mathbf{w}_j}}{\sum_{k=1}^{K} e^{\mathbf{x}^\top \mathbf{w}_k}}$



	$\mathbf{x}^\top \mathbf{w}_k$	$e^{\mathbf{x}^\top \mathbf{w}_k}$	$\frac{e^{\mathbf{x}^\top \mathbf{w}_j}}{\sum_{k=1}^{K} e^{\mathbf{x}^\top \mathbf{w}_k}}$	Label
강아지	0.37	1.4477	0.2296	0
고양이	1.58	4.8549	0.7704	1

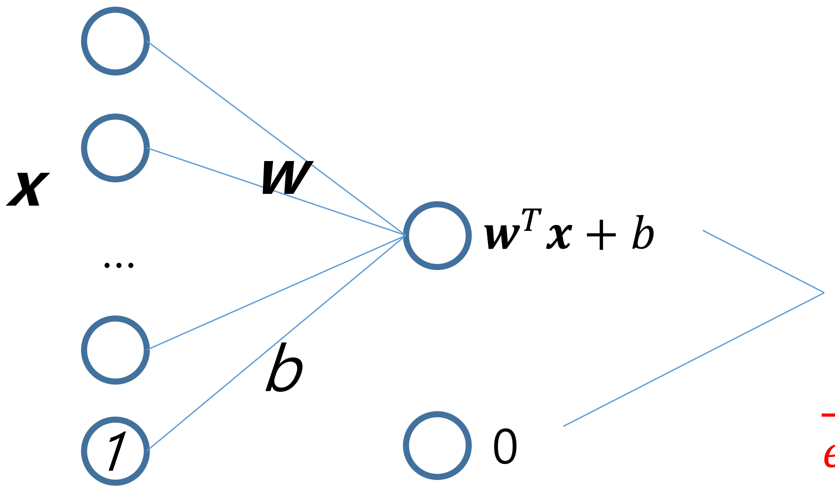
수치 → exp → 확률 ↔ 확률

softmax!

- Loss Function – Cross Entropy
  - $L = \sum -y \log H(x)$,
  - y는 $label$(정답), $H(x)$는 $network\ output$(예측값, softmax 결과)

Binary Classification

- 마지막 layer의 activation function
  - Sigmoid – $H(x) = \sigma(\boldsymbol{w}^T\boldsymbol{x} + b) = \dfrac{1}{1+e^{-(\boldsymbol{w}^T\boldsymbol{x}+b)}} = P(y|\boldsymbol{x})$



Softmax → sigmoid

$$\frac{e^{\boldsymbol{w}^T\boldsymbol{x}+b}}{e^{\boldsymbol{w}^T\boldsymbol{x}+b} + e^0} = \frac{1}{1 + e^{-(\boldsymbol{w}^T\boldsymbol{x}+b)}}$$