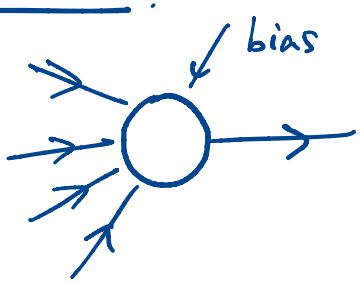
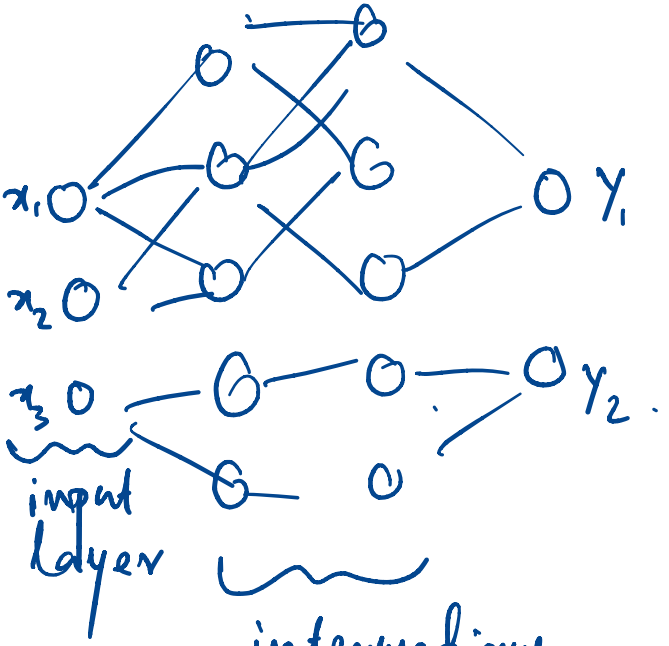
Lecture 3.



weighted linear comb^n



$$y = f_\theta(x)$$

weights & biases.

Data: $(x^1, y^1), \ldots, (x^N, y^N)$

Find θ, that minimizes $\frac{1}{N}\sum_{i=1}^{N} \text{loss}\left(f_\theta(x^i), y^i\right) := E(\theta)$

minimize $E(\theta)$
θ.

Setting $\nabla E(\theta) = 0$
... solve θ^*.
Think line fitting.

} Closed-form expression.

Computational procedure
... Gradient descent.

$-\nabla E(\theta)$ ... denotes the direction of steepest descent.

... $\theta^0 \longrightarrow$ walk along $-\nabla E(\theta^0)$

θ^1 ← walk along $-\nabla E(\theta^1)$

⋮

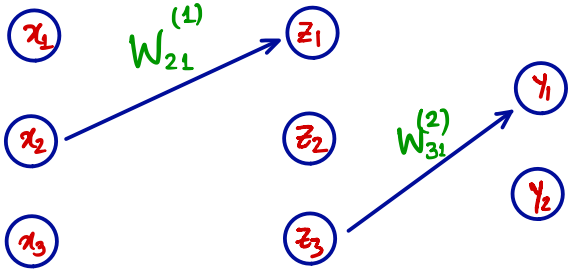
step-size / learning rate

- $\theta^{k+1} \leftarrow \theta^k - \eta \nabla E(\theta^k)$
  Compute.

① how far to walk?
② when to stop?

Non-linear regression via neural networks (NN)

Neural networks provide a parametric map between input and output vectors in terms of weights and biases between each of its layers. Consider a NN which has $x \in \mathbb{R}^3$ as inputs and $y \in \mathbb{R}^2$ as outputs, and has one hidden layer with 3 neurons. Let's draw the NN architecture.



$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \in \mathbb{R}^3$$

$$z = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \in \mathbb{R}^3$$

$$y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \in \mathbb{R}^2.$$

The functional relationships are given by

$$z = \varphi^{(1)}\left(W^{(1)} x + b^{(1)} \right),$$

$$y = \varphi^{(2)}\left(W^{(2)} z + b^{(2)} \right).$$

φ's are activation functions.

$$z = \varphi^{(1)}\left(W^{(1)}x + b^{(1)}\right),$$
$$y = \varphi^{(2)}\left(W^{(2)}z + b^{(2)}\right).$$

$W^{(1)}$ and $W^{(2)}$ are weights.

$b^{(1)}$ and $b^{(2)}$ are biases.

- Dimensions of W's and b's?

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

- How do we tune the weights and biases?

Suppose you are given a pair (x, t), where $x \in \mathbb{R}^3$ and $t \in \mathbb{R}^2$. We aim to find $\theta := \left(W^{(1)}, W^{(2)}, b^{(1)}, b^{(2)}\right)$ such that the error between t and y is minimized.

Consider an error (aka loss function) defined as

$$E(\theta) := \frac{1}{2}\left[\left(t_1^{(1)} - y_1\right)^2 + \left(t_2^{(1)} - y_2\right)^2\right].$$

$$E(\theta) = \frac{1}{N}\sum_{i=1}^{N} E^{(i)}(\theta)$$

To minimize E, start with some θ and

take a step along the negative gradient of E, i.e., update θ as

$$\theta \leftarrow \theta - \eta \cdot \nabla_\theta E(\theta)$$

for a suitable step-length η.

- How do we calculate $\nabla_\theta E(\theta)$?

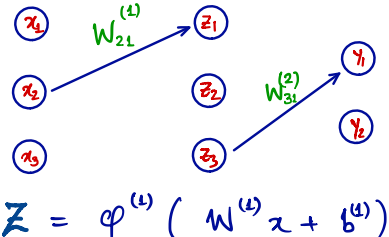
Unpacking the gradient, we seek to calculate

$\frac{\partial E}{\partial W_{11}^{(1)}}, \frac{\partial E}{\partial W_{12}^{(1)}},$

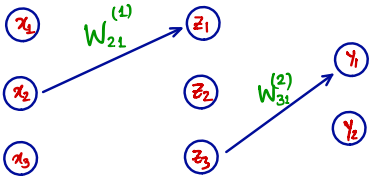
$\dots \frac{\partial E}{\partial W_{11}^{(2)}}, \frac{\partial E}{\partial W_{12}^{(2)}},$

$\dots \frac{\partial E}{\partial b_1^{(1)}}, \dots$

$\frac{\partial E}{\partial b_1^{(2)}}, \dots \frac{\partial E}{\partial b_2^{(2)}}.$



We calculate these derivatives beginning from the last layer first, and proceed backwards.



$$z = \varphi^{(1)}\left(W^{(1)}x + b^{(1)}\right),$$

$$y = \varphi^{(2)}\left(W^{(2)}z + b^{(2)}\right).$$

// Let's calculate $\frac{\partial E}{\partial W_{31}^{(2)}}$.

New notation:

$$z^{in} = W^{(1)}x + b^{(1)}$$

$$y^{in} = W^{(2)}z + b^{(2)}.$$

$\therefore$ $z = \varphi^{(1)}(z^{in})$ and $y = \varphi^{(2)}(y^{in})$

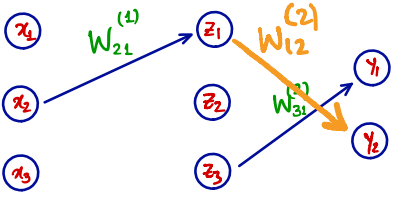
Then, we have

$$\frac{\partial E}{\partial W_{31}^{(2)}} = \underbrace{\frac{\partial E}{\partial y_1} \cdot \frac{\partial y_1}{\partial y_1^{in}}}_{\text{Call it } \delta_1^{(2)}} \cdot \underbrace{\frac{\partial y_1^{in}}{\partial W_{31}^{(2)}}}_{= z_3} = \delta_1^{(2)} \cdot z_3$$

$$y_1^{in} = z_1 W_{11}^{(2)} + z_2 \cdot W_{21}^{(2)} + z_3 \cdot W_{31}^{(2)}$$

Also, we have

$$\frac{\partial E}{\partial b_1^{(2)}} = \underbrace{\frac{\partial E}{\partial y_1} \cdot \frac{\partial y_1}{\partial y_1^{in}}}_{= \delta_1^{(2)}} \cdot \underbrace{\frac{\partial y_1^{in}}{\partial b_1^{(2)}}}_{1} = \delta_1^{(2)} \cdot$$



$$z = \varphi^{(1)}\left(W^{(1)}x + b^{(1)}\right),$$

$$y = \varphi^{(2)}\left(W^{(2)}z + b^{(2)}\right).$$

// Similarly, let's calculate $\frac{\partial E}{\partial w_{12}^{(2)}}$.

$$\frac{\partial E}{\partial w_{12}^{(2)}} = \frac{\partial E}{\partial y_2} \cdot \frac{\partial y_2}{\partial y_2^{in}} \cdot \frac{\partial y_2^{in}}{\partial w_{12}^{(2)}}$$

$$= \delta_2^{(2)} \cdot z_1$$

and $\frac{\partial E}{\partial b_2^{(2)}} = \delta_2^{(2)}$.

We see a pattern here...

$$\frac{\partial E}{\partial w_{jk}^{(2)}} = \delta_k^{(2)} \cdot z_j, \qquad \frac{\partial E}{\partial b_k^{(2)}} = \delta_k^{(2)}$$

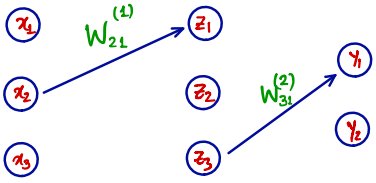
// Calculating $\delta_k^{(2)}$...

$$E(\theta) := \frac{1}{2}\left[(t_1 - y_1)^2 + (t_2 - y_2)^2\right].$$

$$\delta_k^{(2)} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial y_k^{in}} = (y_k - t_k) \cdot \left[\varphi^{(2)}\right]'(y_k^{in}).$$

$$y_k = \varphi^{(2)}(y_k^{in}).$$

// March on to calculate $\frac{\partial E}{\partial w_{jk}^{(1)}}$, $\frac{\partial E}{\partial b_k^{(1)}}$.



$$z = \varphi^{(1)}\left(W^{(1)}x + b^{(1)}\right),$$

$$y = \varphi^{(2)}\left(W^{(2)}z + b^{(2)}\right).$$

Let's tackle an example first. What is $\frac{\partial E}{\partial w_{21}^{(1)}}$?

$$\frac{\partial E}{\partial w_{21}^{(1)}} = \frac{\partial E}{\partial z_1} \cdot \frac{\partial z_1}{\partial z_1^{in}} \cdot \frac{\partial z_1^{in}}{\partial w_{21}^{(1)}}$$

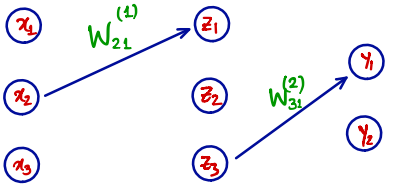
Looks familiar

$\rightarrow x_2$.

Call this quantity $\delta_1^{(1)}$.

$$\therefore \frac{\partial E}{\partial w_{21}^{(1)}} = \delta_1^{(1)} \cdot x_2$$

... more generally $\frac{\partial E}{\partial w_{jk}^{(1)}} = \delta_k^{(1)} \cdot x_j$



$$z = \varphi^{(1)}\left(W^{(1)}x + b^{(1)}\right),$$

$$y = \varphi^{(2)}\left(W^{(2)}z + b^{(2)}\right).$$

" $\dfrac{\partial E}{\partial b_2^{(1)}} = \dfrac{\partial E}{\partial y_2} \cdot \dfrac{\partial y_2}{\partial y_2^{in}} \cdot \dfrac{\partial y_2^{in}}{\partial b_2^{(1)}}$.

$$= \delta_2^{(1)}$$

... more generally

$$\frac{\partial E}{\partial b_k^{(1)}} = \delta_k^{(1)}.$$

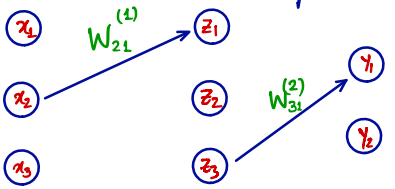
" Calculating $\delta_k^{(1)}$.

Notice that $\delta_k^{(1)} = \dfrac{\partial E}{\partial z_k} \cdot \dfrac{\partial z_k}{\partial z_k^{in}}$

$$= \frac{\partial E}{\partial z_k} \cdot \left[\varphi^{(1)}\right]'\left(z_k^{in}\right).$$

This quantity is not as easy to compute as $\dfrac{\partial E}{\partial y_k}$ in the last layer. Why? E depends explicitly on y's and hence, if is easy to compute $\dfrac{\partial E}{\partial y_k}$, but E only depends implicitly on z's.

Let's do a specific example: Calculate $\frac{\partial E}{\partial y_1}$.



$$z = \varphi^{(1)}\left(W^{(1)}x + b^{(1)}\right),$$

$$y = \varphi^{(2)}\left(W^{(2)}z + b^{(2)}\right).$$

Key idea:
E depends on (z_1, z_2) that in turn depends on y_1.

$$\frac{\partial E}{\partial z_1} = \frac{\partial E(y_1, y_2)}{\partial z_1}$$

$$= \frac{\partial E}{\partial y_1}\frac{\partial y_1}{\partial z_1} + \frac{\partial E}{\partial y_2}\cdot\frac{\partial y_2}{\partial z_1}.$$

$$= \underbrace{\frac{\partial E}{\partial y_1}\frac{\partial y_1}{\partial y_1^{in}}}_{=\delta_1^{(2)}}\cdot\underbrace{\frac{\partial y_1^{in}}{\partial z_1}}_{W_{11}^{(2)}} + \underbrace{\frac{\partial E}{\partial y_2}\cdot\frac{\partial y_2}{\partial y_2^{in}}}_{\delta_2^{(2)}}\cdot\underbrace{\frac{\partial y_2^{in}}{\partial z_1}}_{W_{12}^{(2)}}.$$

$y_1^{in} = z_1 \cdot W_{11}^{(2)} + z_2 W_{21}^{(2)} + z_3 \cdot W_{31}^{(2)}$

$$\Rightarrow \frac{\partial E}{\partial z_k} = \delta_1^{(2)} W_{k1}^{(2)} + \delta_2^{(2)} W_{k2}^{(2)}.$$

Notice that you have already calculated $\delta^{(2)}$'s.

$$\therefore \delta_k^{(1)} = \frac{\partial E}{\partial z_k} \cdot \left[\varphi^{(1)}\right]'\left(z_k^{in}\right)$$

$$= \left(\delta_1^{(2)} w_{k1}^{(2)} + \delta_2^{(2)} w_{k2}\right) \left[\varphi^{(1)}\right]'\left(z_k^{in}\right).$$

// Multi-layer neural network :

... left as an exercise.