

# ARTIFICIAL NEURAL NETWORKS

HIDDEN LAYER, NEURON  $i$



$$a_i^{(1)} = \sum_{j=1}^D w_{ij}^{(1)} x_j + w_{i0}^{(1)}$$

$$z_i^{(1)} = h(a_i^{(1)})$$

,  $h$  NON-LINEAR  
ACTIVATION  
FUNCTION

$k$ -th hidden layer

$$a_i^{(k)} = \sum_{j=1}^{M_{k-1}} w_{ij}^{(k)} z_j^{(k-1)} + w_{i0}^{(k)}$$

$$z_i^{(k)} = h(a_i^{(k)})$$

$M_k$ : # NEURONS OF LAYER  $k$

$M_0 = D$  = # inputs

$w_{ij}^{(k)}$  = NEURON  $i$  at level  $k$ ,  
CONNECTION FROM NEURON  
 $j$  AT LEVEL  $k-1$

$w_{i0}^{(k)}$  = bias term of  
neuron  $i$  at level  $k$ .

OUTPUT  $y = y_1, \dots, y_K$   
 $L$  hidden layers

$$a_i^{(L+1)} = \sum_{j=1}^{M_L} w_{ij}^{(L+1)} z_j^{(L)} + w_{i0}^{(L+1)}$$

$y_i = \sigma(a_i^{(L+1)})$  OUTPUT  
ACTIVATION  
FUNCTION

FNN  $L$  hidden layers, each with  $M$  neurons,  $K=1, \dots, L$   
 $D$  inputs,  $K$  outputs

$$W \text{ or } \theta = \left( \omega^{(1)}, \omega^{(2)} \dots \omega^{(L)}, \omega^{(L+1)}, \omega_0^{(1)}, \dots, \omega_0^{(L+1)} \right)$$

$$\omega^{(k)} = \left( \omega_{ij}^{(k)} \right) \begin{matrix} i=1, \dots, M \\ j=1, \dots, D \end{matrix}$$

$$\left| \begin{array}{l} M \times D + (L-1)M^2 + MK + \\ + L \cdot M + K \end{array} \right| \text{ parameters in total.}$$

$L \cdot M + K + D$  # neurons

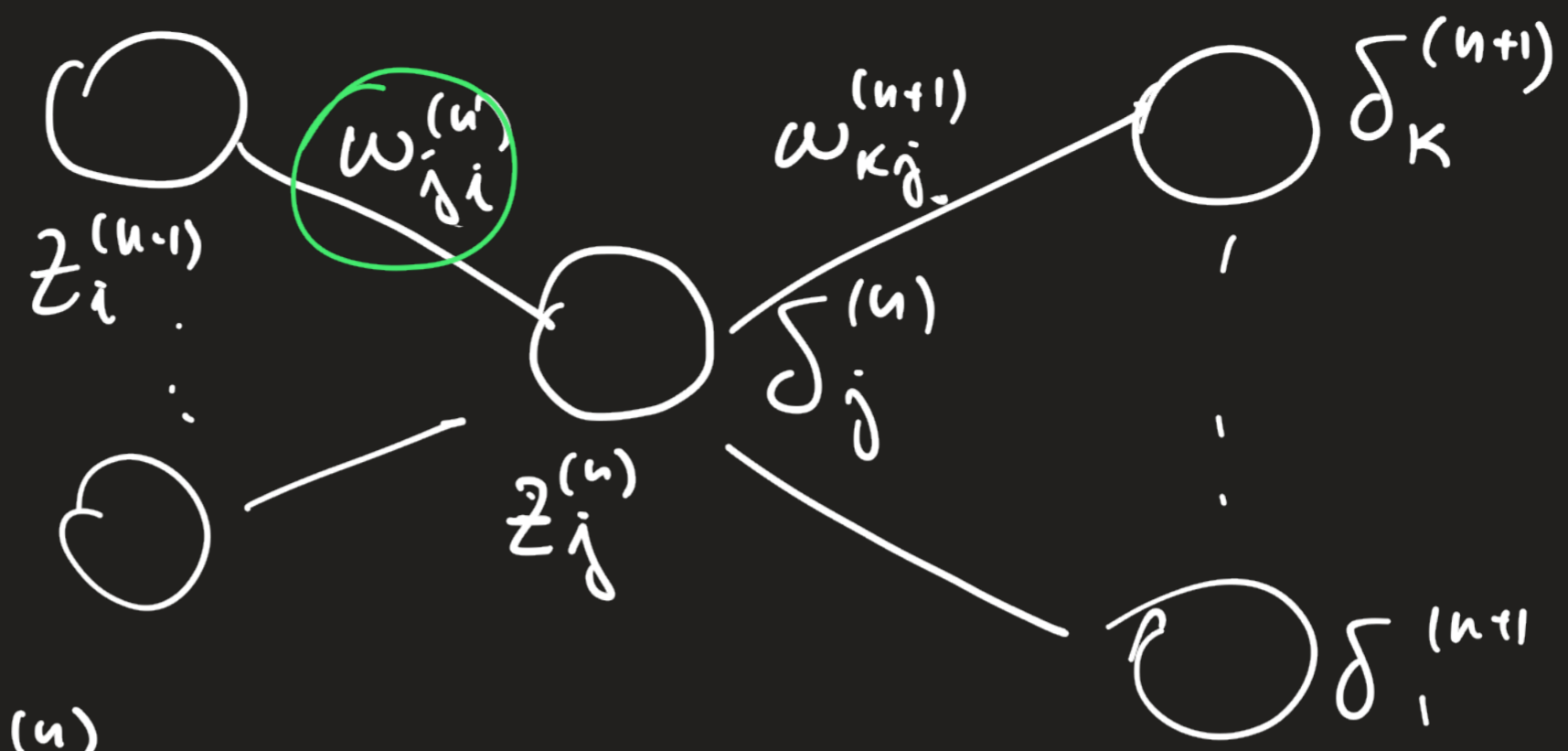
$L \cdot M$  # hidden neurons

$K$  # output neurons

$(x_n, t_n)$  OBSERVATIONS

# BACKPROP

$E_n$  = LOSS EVAL ON INPUT POINT  $x_n$



$$a_j^{(n)} = \sum_{i=1}^{M_{n-1}} w_{ji}^{(n)} z_i^{(n-1)} + w_{j0}^{(n)}$$

$$z_j^{(n)} = h(a_j^{(n)})$$

$$\theta = (w^{(1)}, \dots, w^{(L)}, w^{(L+1)})$$

$$E_n = E(\theta)$$

$$\frac{\partial E_n}{\partial w_{ji}^{(n)}} = \frac{\partial E_n}{\partial a_j^{(n)}} \cdot \frac{\partial a_j^{(n)}}{\partial w_{ji}^{(n)}} = z_i^{(n-1)}$$

The derivative  $\frac{\partial E_n}{\partial w_{ji}^{(n)}}$  is circled in green. The term  $\frac{\partial a_j^{(n)}}{\partial w_{ji}^{(n)}}$  is also circled in green and labeled as  $z_i^{(n-1)}$ . The term  $\frac{\partial E_n}{\partial a_j^{(n)}}$  is circled in blue and labeled as  $\delta_j^{(n)}$ .

$$\delta_k^{(L+1)} = \frac{\partial E_n}{\partial a_k^{(L+1)}}$$

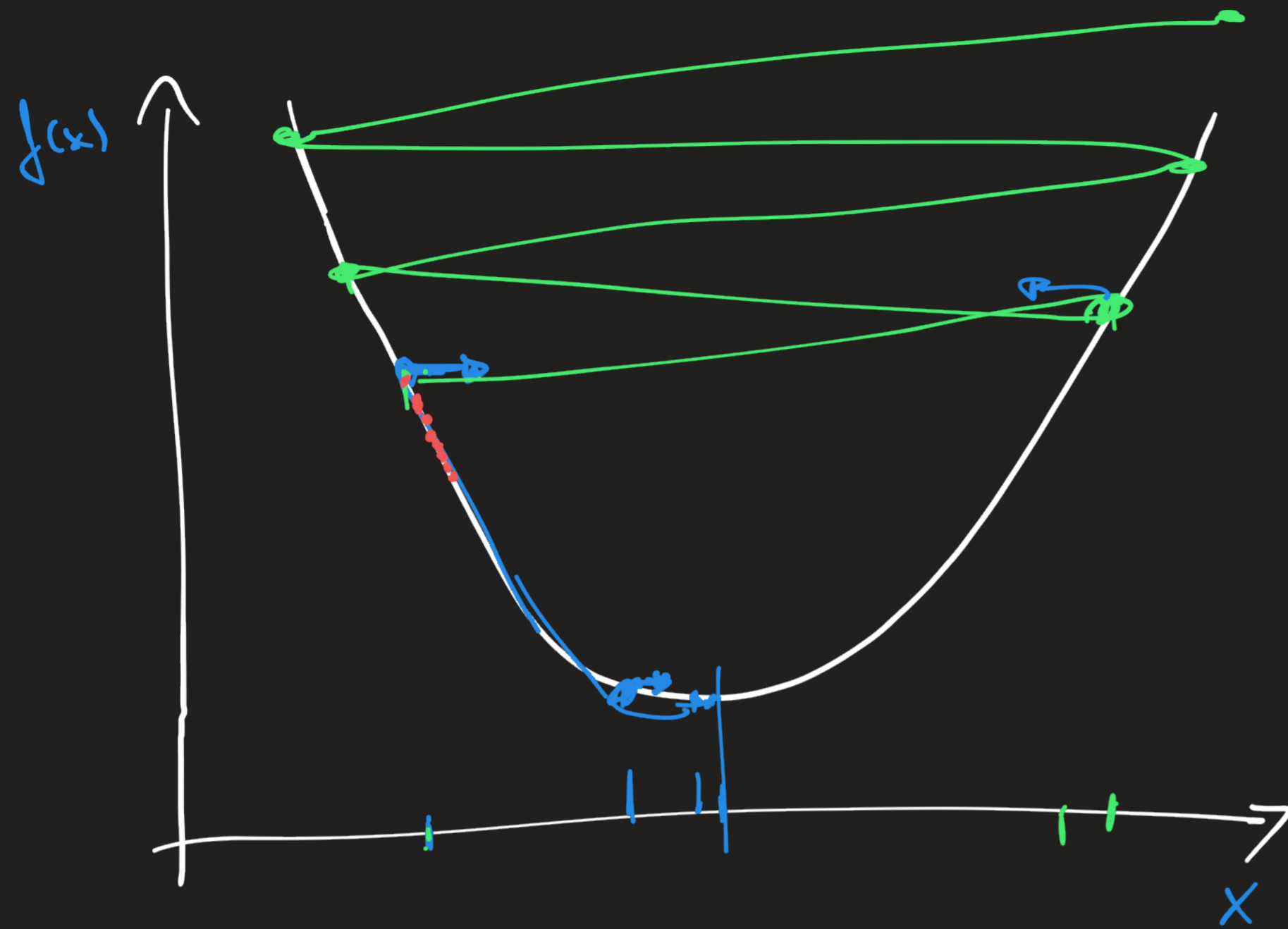
OUTPUT REGRESSION, we have  $\delta_k^{(L+1)} = y_k - t_k$

$$\delta_j^{(n)} = \frac{\partial E_n}{\partial a_j^{(n)}} = \sum_k \frac{\partial E_n}{\partial a_k^{(n+1)}} \frac{\partial a_k^{(n+1)}}{\partial a_j^{(n)}} = h'(a_j^{(n)}) \sum_k w_{kj}^{(n+1)} \delta_k^{(n+1)}$$

The entire equation is boxed in blue. The term  $\frac{\partial E_n}{\partial a_k^{(n+1)}}$  is circled in blue and labeled as  $\delta_k^{(n+1)}$ . The term  $\frac{\partial a_k^{(n+1)}}{\partial a_j^{(n)}}$  is circled in blue and labeled as  $w_{kj}^{(n+1)}$ . The term  $h'(a_j^{(n)})$  is circled in blue and labeled as  $\delta_j^{(n)}$ .

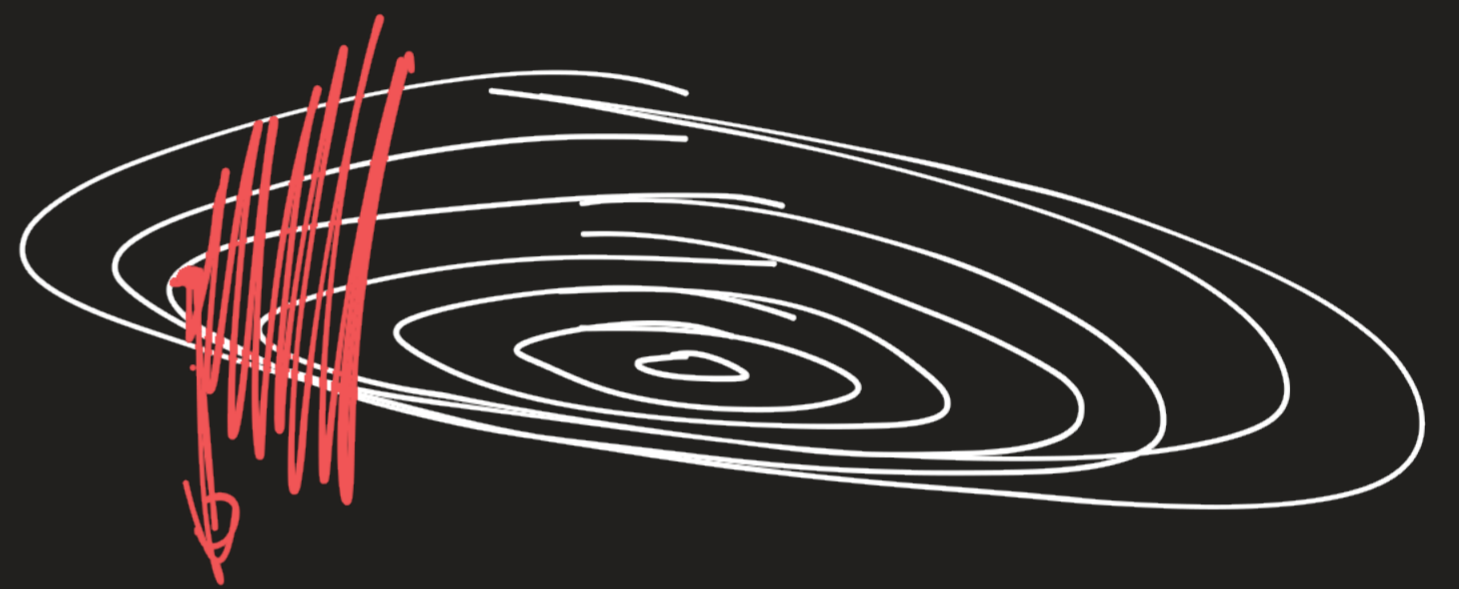
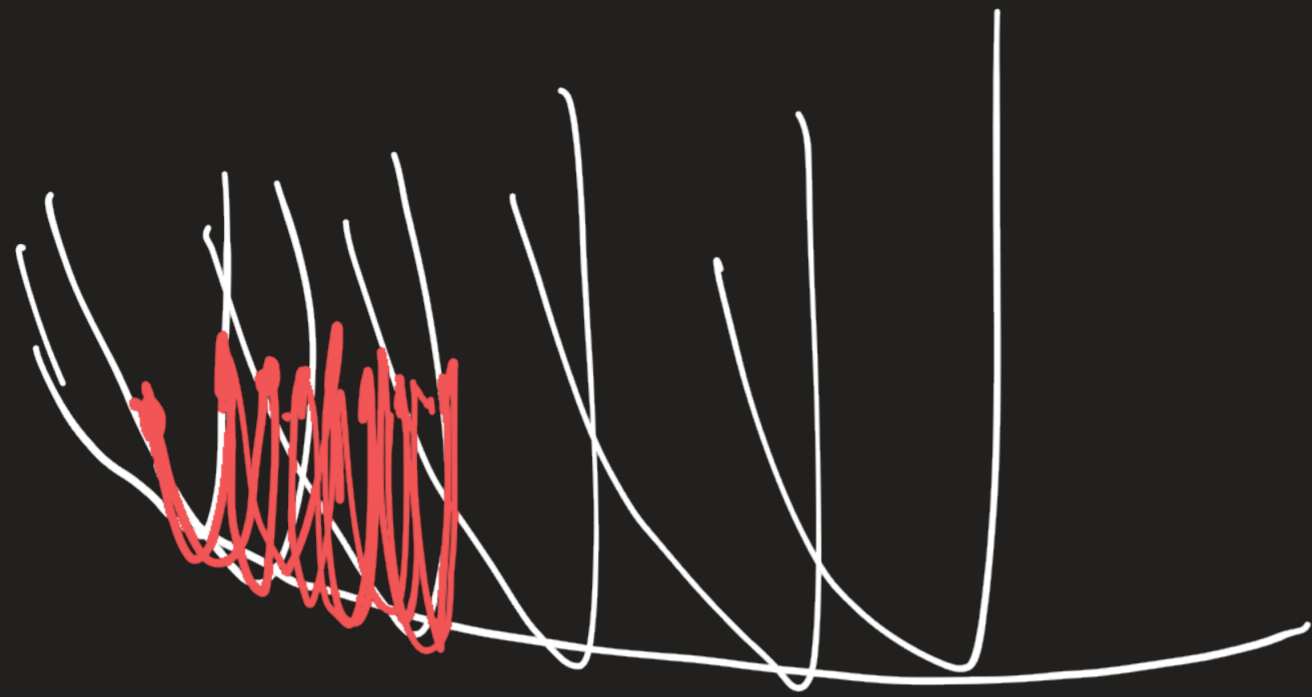
$$a_k^{(n+1)} = \sum_j w_{kj}^{(n+1)} \cdot h(a_j^{(n)})$$

$$h'(a) = \frac{\partial}{\partial a} h(a)$$



SGD,  
learning rate is  
crucial!

CURVATURE OF  $E$  LANDSCAPE  
and SGD



# BAYESIAN NEURAL NETWORKS

$$f(x; \omega) = f_1(f_2(\dots f_k(x, \omega_k), \omega_1, \omega_2) \omega_1) \quad \text{n.n.}$$

LIKELIHOOD  $P(\underline{y} | \omega) = P(y | f(x, \omega))$

$$f(\omega) = \log P(\underline{y} | \omega)$$

$$\frac{P(\underline{y} | \omega) P(\omega)}{P(\underline{y})} = P(\omega | \underline{y})$$

In n.n., computing  $P(\omega | \underline{y})$  is INTRACTABLE.

$\rightarrow$  APPROXIMATION:  $\rightarrow$  MCMC (HMC)  $\Delta \approx$   
 $\sim$  V.I.

$q_\theta(\omega) \approx P(\omega | \underline{y})$ , minimizes KL divergence  $KL(q_\theta(\omega), P(\omega | \underline{y}))$

$$q_\theta(\omega) = \mathcal{N}(\omega | \mu, \Sigma) = \prod \mathcal{N}(\omega_i | \mu_i, \sigma_i^2) \quad q_\theta[\cdot](\omega) = \prod \mathcal{N}(\omega_i | \mu_i, \sigma_i^2)$$