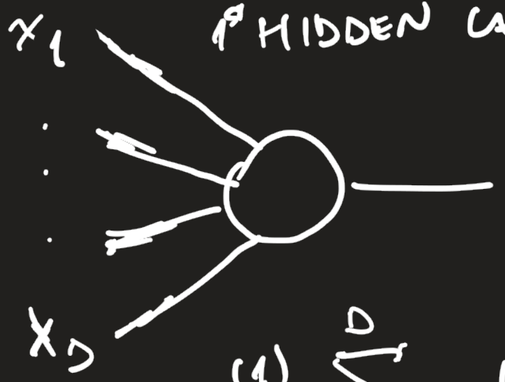
ARTIFICIAL NEURAL NETWORKS



1st 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

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.

K-Th hidden layer

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

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

OUTPUT $y = y_1 \dots y_K$

L hidden layers

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

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

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

$$W_{ORD} = \left(\omega^{(1)}_{|}, \omega^{(2)}, \dots, \omega^{(L)}, \omega^{(L+1)}, \omega_0^{(1)}, \dots, \omega_0^{(L+1)}\right)$$

$$\omega^{(1)} : \left(\omega^{(1)}_{ij}\right)_{\substack{i=1,\dots,M \\ j=1,\dots,D}}$$

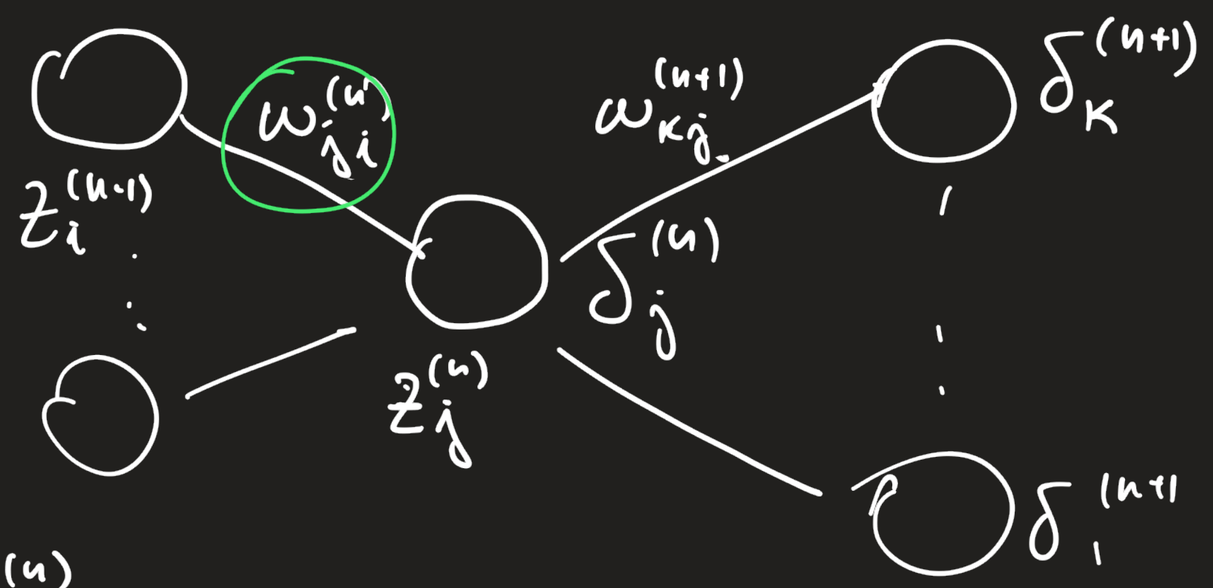
$M\times D + (L-1)M^2 + MK + L\cdot M + K$	parameters in total.

$LM+K+D$ # neurons LM # hidden neurons
K # output neurons

(x_n, t_n) OBSERVATIONS

BACKPROP

E_n : LOSS EVAL ON INPUT POINT x_n



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

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

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

$$E_n = E(\vartheta)$$

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

$$\delta_j^{(n)} := \frac{\partial E_n}{\partial a_j^{(n)}}$$

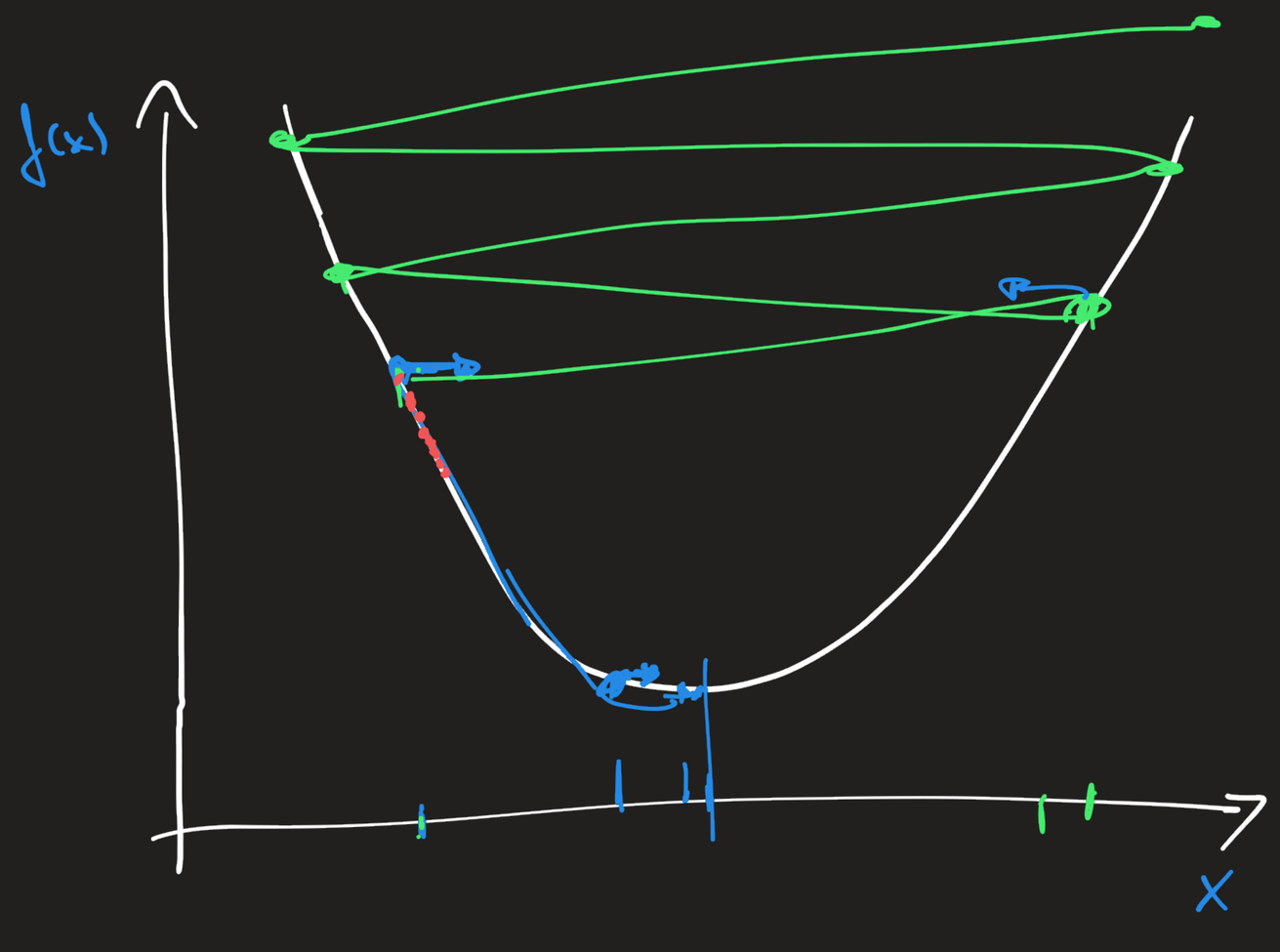
$$\delta_K^{(L+1)} := \frac{\partial E_n}{\partial a_K^{(L+1)}}$$

OUTPUT : REGRESSION, we have $\delta_K^{(L+1)} = y_K - t_K$

$$\delta_j^{(n)} = \frac{\partial E_n}{\partial a_j^{(n)}} = \sum_K \frac{\partial E_n}{\partial a_K^{(n+1)}} \frac{\partial a_K^{(n+1)}}{\partial a_j^{(n)}} = h'(a_j^{(n)}) \sum_K w_{Kj}^{(n+1)} \cdot \delta_K^{(n+1)}$$

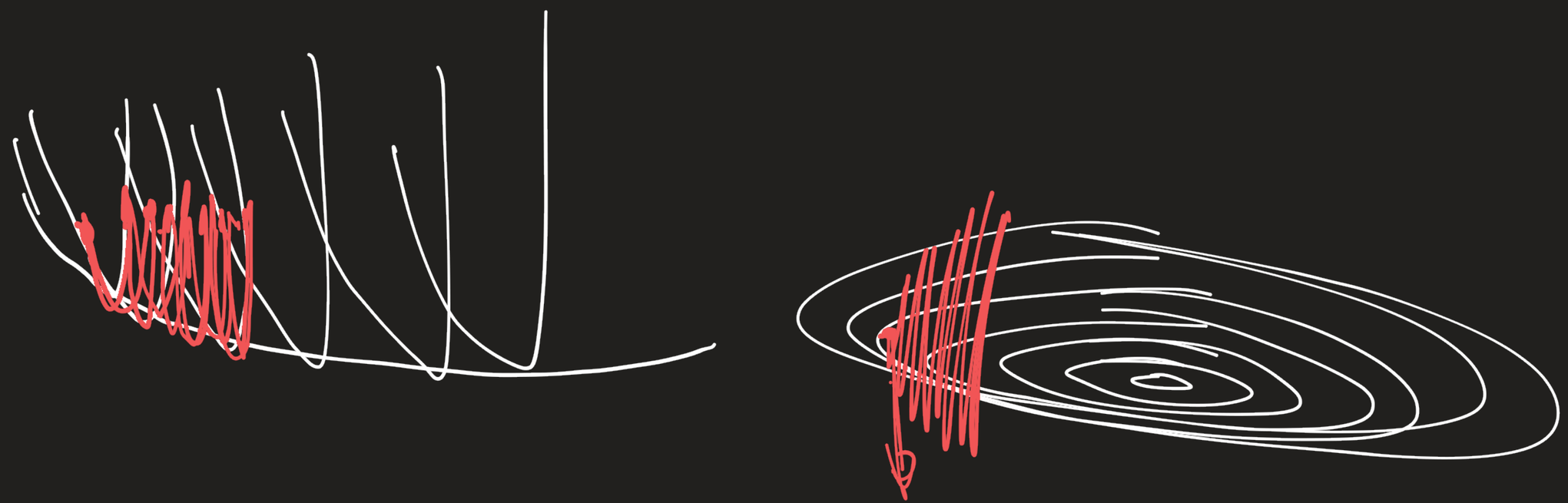
$$a_K^{(n+1)} = \sum_j w_{Kj}^{(n+1)} \cdot h(a_j^{(n)})$$

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



SGD, learning rate is crucial!

CURVATURE OF E LANDSCAPE
and SCD



BAYESIAN NEURAL NETWORKS

$$f(x;\omega) = f_1(f_2 \dots f_K(x, \omega_K), \dots, \omega_2), \omega_1) \quad \text{N.N.}$$

LIKELIHOOD $p(\underline{y} \mid \omega) = p(\underline{y} \mid f(x, \omega))$

$$f(\omega) = \log \underbrace{p(\underline{y} \mid \omega)}$$

$$\frac{p(\underline{y} \mid \omega)\, p(\omega)}{p(\underline{y})} = p(\omega \mid \underline{y})$$

In N.N., computing $p(\omega \mid \underline{y})$ is INTRACTABLE.

$\Rightarrow$ APPROXIMATION: $\rightarrow$ MCMC (HMC) ...

$\rightarrow$ V.I.

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

$q_\theta(\omega) = \mathcal{N}(\omega \mid \mu, \Sigma) = \prod \mathcal{N}(\omega_i \mid \mu_i, \sigma_i^2)$ $\quad q_\theta[1](\omega) = \mu_i + \sigma_i^2 \varepsilon_i$, $\mathcal{N}(0,1)$