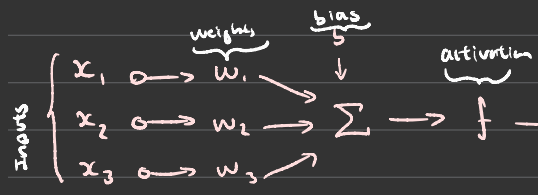
ANN - Artificial Neural Networks

for regression and classification



overfitting is good for training data bad on new data, learns specific noise and trends that may not be general to all data.

$$f(w_1x_1 + w_2x_2 + b) = y$$ ← feed forward equation

↑ activation function

* carried out using backpropagation

Backpropagation : propagate error backwards
adjusts the weights and bias by measuring the error btw the true output and output. The gradient tells us how the error would change, and we use gradient descent, moving in the direction that reduces error most.
* allows nonlinear mapping btw inputs and outputs

Learning Rate : step size of optimisation algorithm.
too high → oscillate around minimum, overshoot
too low → get stuck in local optimum, never converge.

LAB 1

Sigmoid f : $\frac{1}{1+e^{-(w^Tx+b)}} = \sigma$

squared error : $E = (\hat{y} - o)^2$ $\hat{y}$: true output, o : output $\quad x = w^Tx + b$

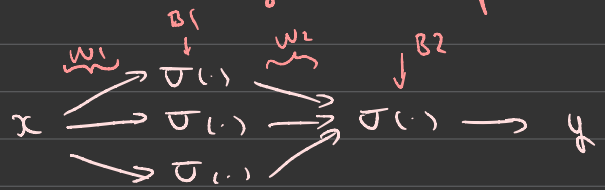
gradient descent using chain rule

$$\frac{dE}{do} = -2(\hat{y} - o) \qquad \frac{do}{dx} = \sigma(1-\sigma) \qquad \frac{dx}{dw} = x \qquad \frac{dx}{db} = 1$$

$$\frac{dE}{dw} = \frac{dE}{do} \times \frac{do}{dx} \times \frac{dx}{dw} \qquad \frac{dE}{db} = \frac{dE}{do} \times \frac{do}{dx} \times \frac{dx}{db}$$

↓ multiply by learning rate

LAB 2 - adding hidden layer



Different dimensions? depends on the # of nodes.
first layer (hidden) is (3,2) because it connects 3 sigmoid nodes for 2 inputs (1 or 0)
second layer is (1,3) as it is reduced to 1 sigmoid node for the 3 inputs (outputs from prior layer)

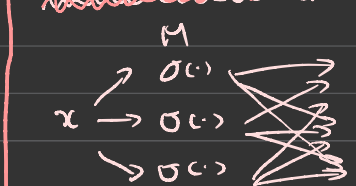
Backpropagation Derivation

W2: $w = w + \alpha \frac{dE}{dw}$, B2, Both same as lab 1

W1: $\frac{dL}{dw_1} = \frac{dL}{dH} \times \frac{dH}{dz} \times \frac{dz}{dw_1}$ $\quad \frac{dL}{dH} = \text{output.dot(W2)}$

$\frac{dL}{db_1} = \frac{dL}{dH} \cdot \frac{dH}{dz} \cdot \frac{dz}{db_1}$

Softmax function: Classification Problem



M = # of neurons in hidden layer

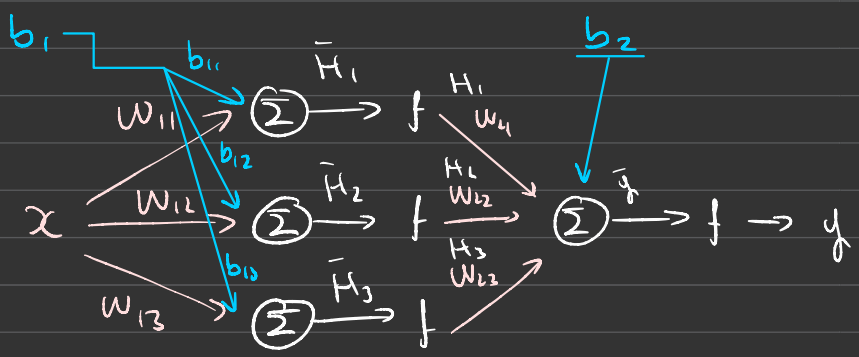
N = # of classes

$O_i = f_i(z) = \frac{e^{z_i}}{\sum_{i=1}^{N} e^{z_i}}$ rescales output to range [0, 1] and sets sum = 1, so output is a probability.

for softmax, use cross-entropy loss for error function

* measures performance of classification, a perfect model would have a loss of 0.
* for backpropagation: $\frac{dH}{dS_i} = O_i - Z_i$

$$H = -\sum_{j=1}^{N} Z_j \log(O_j)$$



① work backwards and find diff_B2 & diff_B1

$$y = f(\bar{y}) \qquad \bar{y} = \Sigma w_2 H + b_2$$

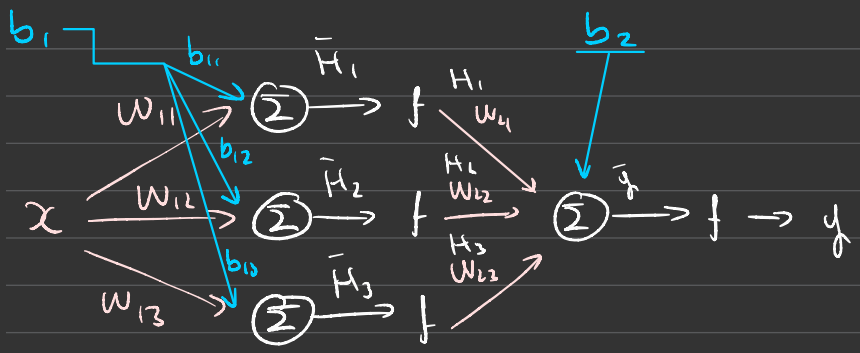
$$\therefore \bar{y} = w_{21}H_1 + w_{22}H_2 + w_{23}H_3 + b_2$$

$$E = -\frac{1}{2}(\hat{y} - y)^2 \qquad \frac{dE}{dy} = \hat{y} - y \qquad * \hat{y} = \text{real} \quad y = \text{calculated}$$

② $$\frac{dE}{dw_2} = \boxed{\frac{dE}{dy}\frac{dy}{d\bar{y}}\frac{d\bar{y}}{dw_2}} = (\hat{y} - y)(f'(\bar{y}))(H)$$

if f is sigmoid $f'(\bar{y}) = \bar{y}(1 - \bar{y})$

③ $$\frac{dE}{db_2} = \boxed{\frac{dE}{dy}\frac{dy}{d\bar{y}}\frac{d\bar{y}}{dw_2}} = (\hat{y} - y)(f'(\bar{y}))(1)$$



① work backwards and find diff_B2 & diff_B1

$y = f(\bar{y}) \quad \bar{y} = \Sigma w_2 H + b_2$

$\therefore \bar{y} = w_{21} H_1 + w_{22} H_2 + w_{23} H_3 + b_2$

$E = -\frac{1}{2}(\hat{y} - y)^2 \quad \frac{dE}{dy} = \hat{y} - y$ * $\hat{y}$ = real y = calculated

② $\frac{dE}{dw_2} = \boxed{\frac{dE}{dy} \cdot \frac{dy}{d\bar{y}} \cdot \frac{d\bar{y}}{dw_2}} = (\hat{y} - y)(f'(\bar{y}))(H)$

if f is sigmoid

$f'(\bar{y}) = \bar{y}(1 - \bar{y})$

③ $\frac{dE}{db_2} = \boxed{\frac{dE}{dy} \cdot \frac{dy}{d\bar{y}} \cdot \frac{d\bar{y}}{dw_2}} = (\hat{y} - y)(f'(\bar{y}))(1)$

④ find $\frac{dE}{dw_1}$ and $\frac{dE}{db_1}$

$= f(\bar{H}) \quad \bar{H} = \Sigma w_1 x + b_1 \quad E = -\frac{1}{2}(\hat{H} - H)^2$

$\frac{dE}{dH} = \hat{H} - H$

$\frac{dE}{dw_1} = \frac{dE}{dH} \cdot \frac{dH}{d\bar{H}} \cdot \frac{d\bar{H}}{dw_1} = \frac{dE}{dH} \cdot f'(\bar{H}) \cdot x = (\hat{y} - y)(y(1-y))w_2(H(1-H))x$

$\frac{dE}{dH} = \frac{dE}{dy} \cdot \frac{dy}{d\bar{y}} \cdot \frac{d\bar{y}}{dH} = (\hat{y} - y) \cdot y(1-y) \cdot w_2$

$\frac{dE}{db_1}$ is the same w/o w_2