HB → [Neural Network.]

Robust → ANN
CNN
RNN ← NLP

Computer V.

[1. Input
2. Hidden
3. Output] ← Design

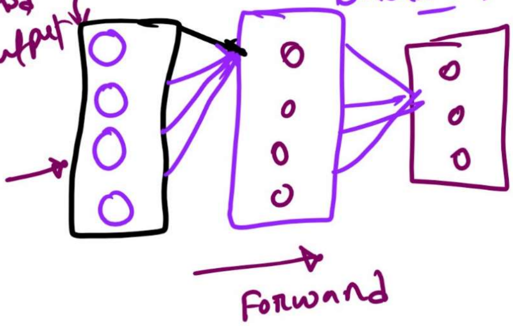
efficient

good output

Neuron/Node

1. Dir
2. Unn → Uni
→ Bi-Dir

Backward



Forward

Cats and dogs [2]

Number of Nodes $\frac{16 + 2}{2} = 9$ Nodes

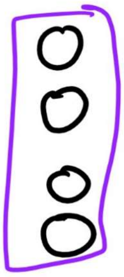
Node [Any Number =

Step → 01: input layer = n = 1, 2, 2^n

2
4
8
16
32

← Easy To Calculate

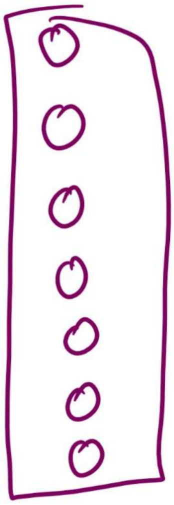
02: Hidden Layer



Number of Nodes = $\frac{\text{Input Nodes} + \text{output Nodes}}{2}$

03: output

1. Regression → 1 Node
2. Binary → 2 n
3. Multi → n Class



01. Rule of Thumb = $\frac{In + On}{2} = Hn$

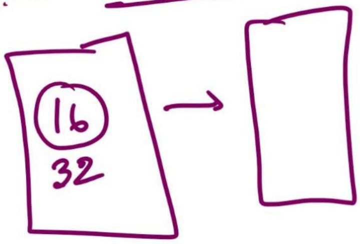
02. 2/3 Rules = $\frac{2}{3} \times In$

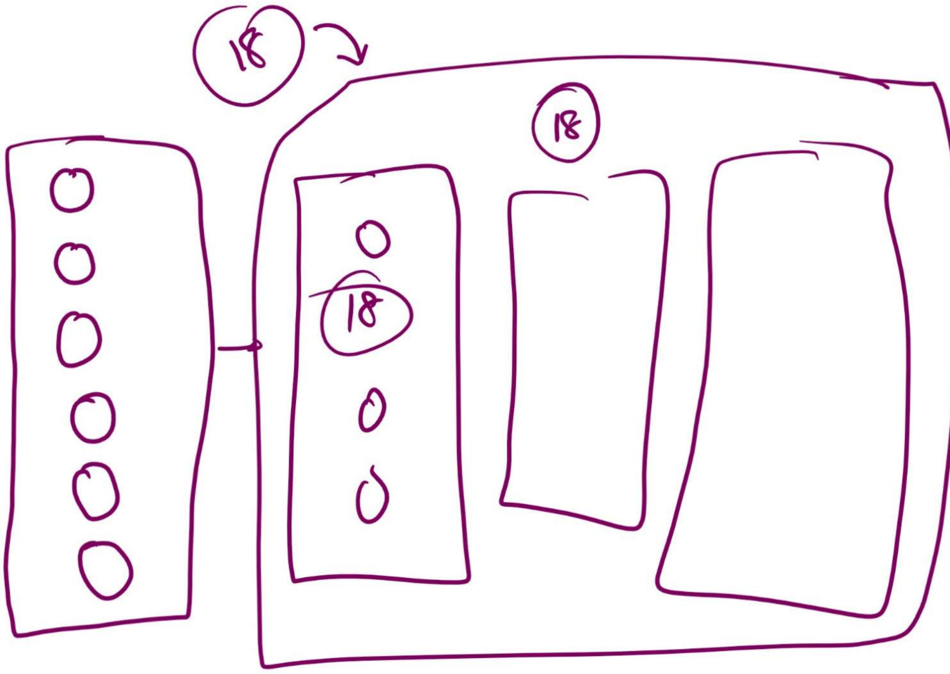
$= \frac{2}{\cancel{3}} \times \cancel{27}^{9}$

$= 18$

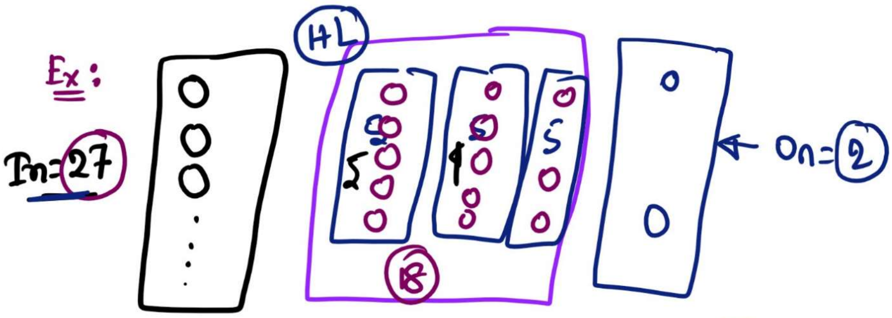
$= \frac{2}{3} \times 10$

$= 6.67 \approx 7$





Ex:



Tuning

RTH

03 Rules

① $\frac{In + On}{2} = \frac{27+2}{2} \simeq \boxed{15} = Hn$

Fixed

② $\boxed{\frac{2}{3} In} = \frac{2}{3} \times 27$

$= 2 \times 9 = \boxed{18}$

③ $2 \times In = 2 \times 27 = 54$ ← HN

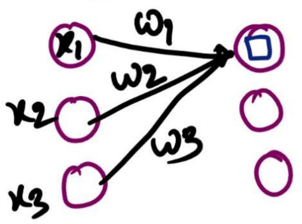
Feature/Attribute

Importance

1. Node

bias = b

02 Weights



x_1	w_1
x_2	w_2
x_3	w_3

Source

NN

$\boxed{y = mx + c}$ (w, bias)

$m_1x_1 + m_2x_2 + m_3x_3 + b$

Simple linear Regression (H)

$x_1 = 0.6$
$x_2 = 0.2$
$x_3 = 0.1$

$w_1 = 0.5$
$w_2 = 0.4$
$w_3 = 0.3$

$0.6 \times 0.5 = 0.3$

$0.2 \times 0.4 = 0.08$

$0.1 \times 0.3 = 0.03$

✓ linear Combination

weighted sum

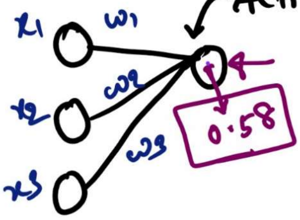
bias

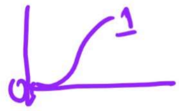
$= 0.3 + 0.08 + 0.03 + 0.1$

update $y = \boxed{0.51}$

$\boxed{uy} = \boxed{0.51}$

Activation Function [door]





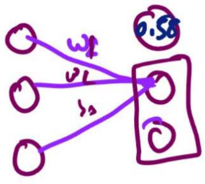
Sigmoid Activation Function

$$= \frac{1}{1+e^{-x}} \qquad x = uy = 0.51$$

$$= \frac{1}{1+e^{-0.51}} = 1$$

$$\approx \boxed{0.625}$$

y ~~pre~~ = 0.625 prediction



0.99

Actual = 100% → 1

(S)

$\boxed{\text{Loss}}$ $\boxed{\text{MSE}} = \frac{1}{2}(\text{pre} - \text{Actual})^2$

M E

↓

$\boxed{\text{FFNN}}$

$$= \frac{1}{2}(0.625 - 1)^2$$

$$= \frac{(0.375)^2}{2}$$

$$= \boxed{0.07} \leftarrow 7\%$$

Backpropagation

dL = loss
y = pred
uy = linear combination

$$L = \frac{1}{2}(P - A)^2$$

$$\left[\frac{dL}{db} = \frac{dL}{dy} \cdot \frac{dy}{duy} \cdot \frac{duy}{db}\right]$$

−0.375 (01)

$$\frac{dL}{dy} = \boxed{\frac{1}{2}(y - y_A)^2} \quad 2 \times \frac{1}{2}(y - y_A)^{2-1}$$

$$\cancel{2}(y - y_A)^1$$

$$y - y_A$$

$$\frac{dL}{dy} = y - y_A$$

$$= 0.625 - 1$$

$$= -0.375$$

(e^x)

$f(x) \cdot (1 - f(x))$

x

$\frac{dy}{duy} = \boxed{\frac{1}{1+e^{-x}}}$ ← $f(z) = 0.625$ (sigmoid)

$\boxed{up = x}$

linear combination

$= f(z) \cdot (1 - f(z))$

$= \boxed{0.625} \cdot (1 - 0.625)$

$= 0.625 \times 0.375$

$= 0.239$

[bias = 1 by default]

Assigning $\boxed{\frac{duy}{db}} = 1$ $\quad$ 0.1

$\boxed{\frac{dL}{dy} \cdot \frac{dy}{duy}} \cdot \frac{duy}{db}$

$= -0.375 \times 0.239 \times 1$

[0-1] ← Learning Rate

$\frac{dL}{db} = \boxed{-0.087}$

$\boxed{b^{new}} = b_{prev} - (\eta \cdot \frac{dL}{db})$

$= \boxed{0.1} - (0.1 \times (-0.087))$

$= 0.1 + 0.00875$

$= 0.10875$

optimal

7% Error ↓ Reduce ↓

$+ \frac{-}{(1+e^{-z})^2}$ $\quad f(z)$

$\frac{e^{-z}}{(1+e^{-z})^2}$

$= \frac{d}{dz}(1)$

$= 0 + (-e^{-z})$

$$\frac{e^{z}}{(1+e^{-z})^2} \rightarrow f(z) \cdot (1-f(z))$$