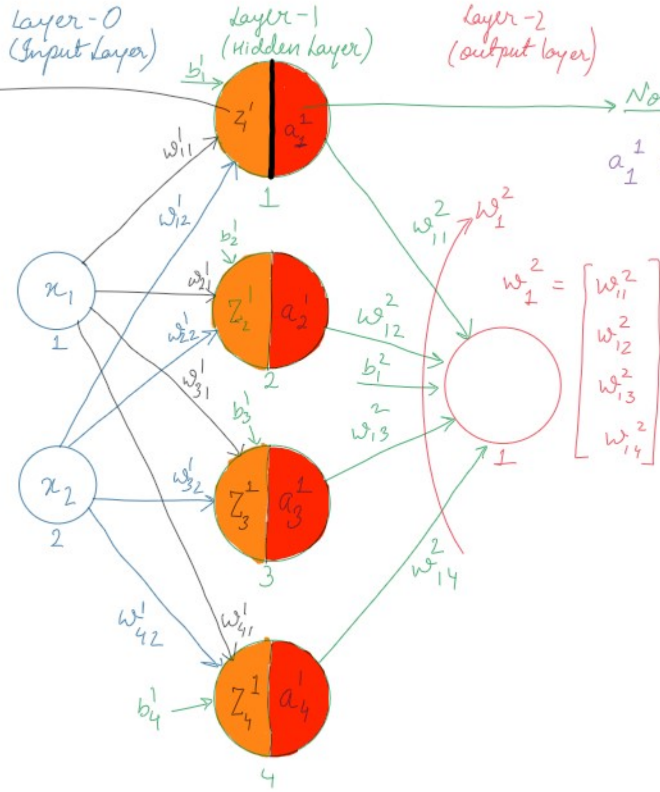
ANN Part-3 Output Functions

06 August 2023 08:59 PM

Number of Layers
$L = 2$



Linear Combination

$$z_1^1 = w_{11}^1 x_1 + w_{12}^1 x_2 + b_1^1$$
$$= (w_1^1)^T x + b_1^1$$

where $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$$a_1^1 = f(z_1^1)$$

Similarly,

$$a_2^1 = f(z_2^1)$$
$$a_3^1 = f(z_3^1)$$
$$a_4^1 = f(z_4^1)$$

$$w_1^2 = \begin{bmatrix} w_{11}^2 \\ w_{12}^2 \\ w_{13}^2 \\ w_{14}^2 \end{bmatrix}$$

* w_1^1 is represented in column vector by default but in order to perform the vector multiplication (Dot Product) we need to take the transpose.

Similarly,

$$z_2^1 = (w_2^1)^T x + b_2^1$$
$$z_3^1 = (w_3^1)^T x + b_3^1$$
$$z_4^1 = (w_4^1)^T x + b_4^1$$

Now,

x_1 & x_2 → Inputs for layer 1.

$a_1^1, a_2^1, a_3^1, a_4^1$ → Output for Layer 1

These outputs of layer 1 will be given as input to layer 2.

$\Rightarrow a^1 = \begin{bmatrix} a_1^1 \\ a_2^1 \\ a_3^1 \\ a_4^1 \end{bmatrix}$ ⇒ This will be the input to layer 2.

$$z^1 = \begin{bmatrix} z_1^1 \\ z_2^1 \\ z_3^1 \\ z_4^1 \end{bmatrix} = \begin{bmatrix} (w_1^1)^T \\ (w_2^1)^T \\ (w_3^1)^T \\ (w_4^1)^T \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1^1 \\ b_2^1 \\ b_3^1 \\ b_4^1 \end{bmatrix}$$

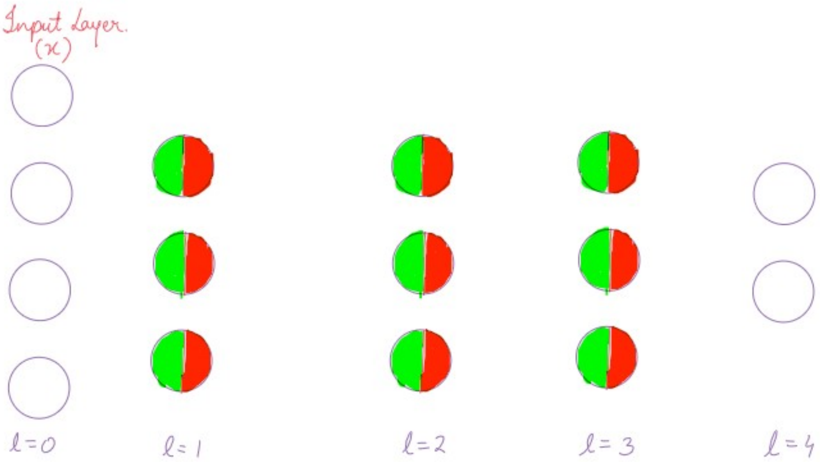
Similarly, for layer 3 we will have a^2, and so on so forth. ⇒ for layer l we will have $a^{(l-1)}$ as the input.

{Special Case $a^0 = x$}

$$\Rightarrow \boxed{z^l = w^l a^{(l-1)} + b^l}$$ Linear Part

$$\Rightarrow \boxed{a^l = f(z^l)}$$ Non-Linear Part

Visual Understanding of the above equations



$Z^1 = w^1 a^0 + b^1$ $\quad$ $Z^2 = w^2 a^1 + b^2$ $\quad$ $Z^3 = w^3 a^2 + b^3$

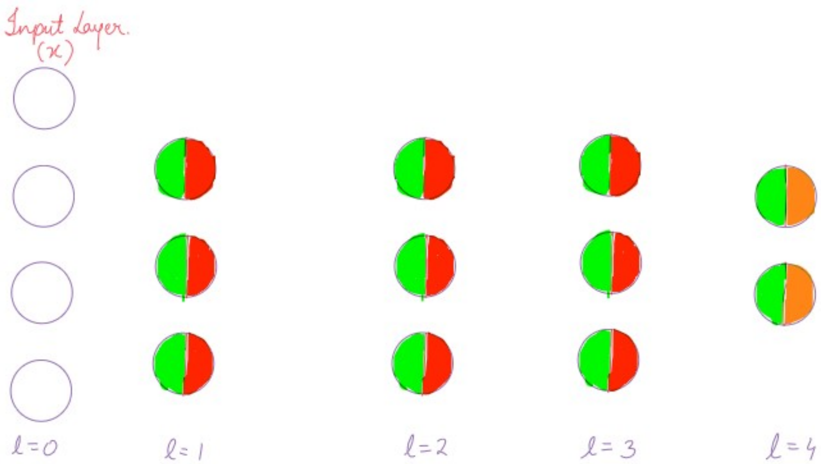
$a^1 = f(Z^1)$ $\quad$ $a^2 = f(Z^2)$ $\quad$ $a^3 = f(Z^3)$

In a neural network → Number of Layers
→ Number of Neurons in each layer
} These both are hyper-parameters and need to be tuned.

The above notation and figure representation work only for feed forward neural network.

Output function

Output function is the activation function in the last layer of a neural network.



$l=0$ is input layer

$\boxed{Z^l = w^l a^{l-1} + b^l}$ ⇒ where $l \in \{1, 2, \ldots, L\}$

$\boxed{a^l = f(Z^l)}$ ⇒ where $l \in \{1, 2, \ldots, L-1\}$

$\boxed{a^0 = x}$

and for the last layer 'L' ⇒ $\boxed{a^L = O(Z^L)}$

$Z^4 = w^4 a^3 + b^4$

$\hat{y} = a^4 = O(Z^4)$

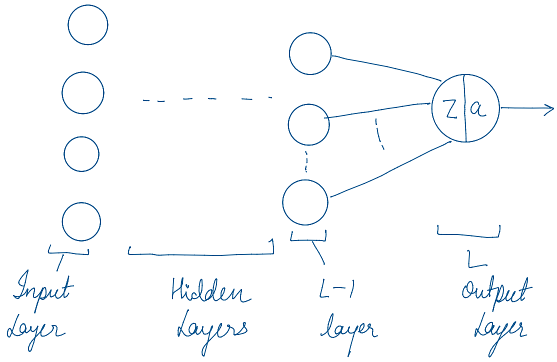
$\hat{y}$ → Prediction

O → Output function

→ Output functions :

Classification :

1.) Binary Classification : There will be 2 classes.



$a = \sigma(z)$

⇓

$a \rightarrow$ logit funct^n / Sigmoid function

$a \in [0,1]$ $\quad \sigma(z) = \frac{1}{1+e^{-z}}$

$$\boxed{\hat{y} = \begin{cases} 1 \ ; & a^L \geq 0.5 \\ 0 \ ; & a^L < 0.5 \end{cases}}$$

2.) Multi-Class Classification : eg 3 classes → {1, 2, 3}

1 → a_1^L

2 → a_2^L

3 → a_3^L

$a \rightarrow$ Output function → Softmax Activation function

$$\boxed{\text{Softmax}(a_j^L) = \frac{e^{a_j^L}}{e^{a_1^L} + e^{a_2^L} + e^{a_3^L}}}$$

$$\boxed{0 \leq \text{Softmax}(a_j^L) \leq 1} \text{ — (1)}$$

{ e^x is always +ve }

eg

$\text{Softmax}(a_1^L) = \frac{e^{a_1^L}}{e^{a_1^L} + e^{a_2^L} + e^{a_3^L}}$ | $\text{Softmax}(a_2^L) = \frac{e^{a_2^L}}{e^{a_1^L} + e^{a_2^L} + e^{a_3^L}}$ | $\text{Softmax}(a_3^L) = \frac{e^{a_3^L}}{e^{a_1^L} + e^{a_2^L} + e^{a_3^L}}$

$$\boxed{\sum_{j=1}^{3} \text{softmax}(a_j^L) = 1} \text{ — (2)}$$

(1) <=> (2)

Softmax gives a probabilistic value for each class

1 → a_1^L → $S.M(a_1^L)$ → 0.3

2 → a_2^L → $S.M.(a_2^L)$ → 0.5

3 → a_3^L → $S.M.(a_3^L)$ → 0.2

1

Softmax gives the probability for each class.

i.e. the probability of input belonging to each of the classes.

In the above eg. input will belong to class 2 as it has the highest probability. ⇒ $\boxed{\hat{y} = 2}$