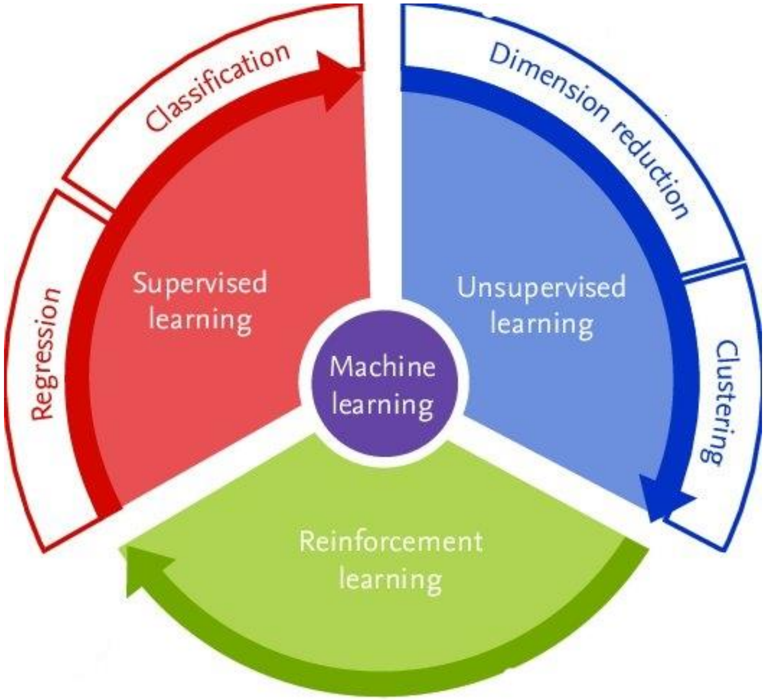
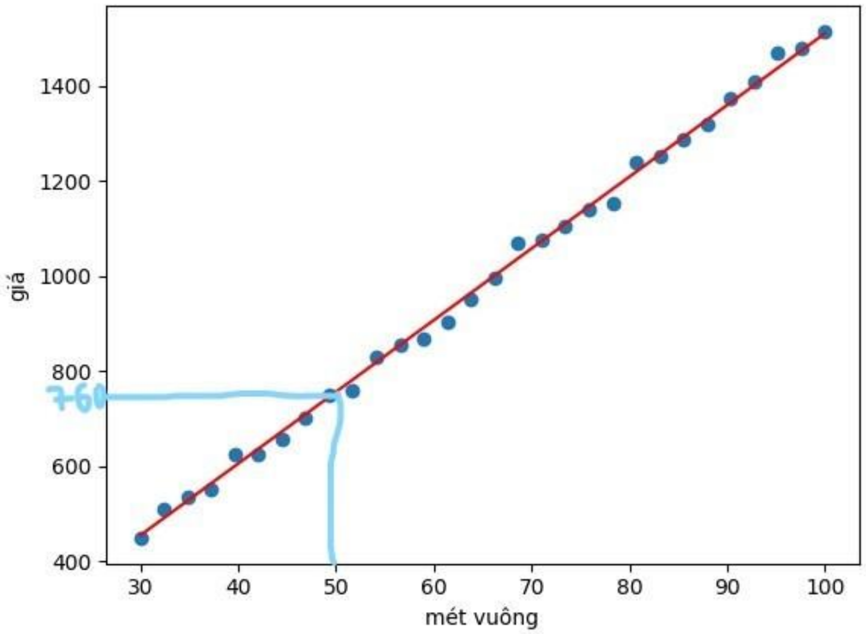
Neural Network

Tuan Nguyen - AI4E

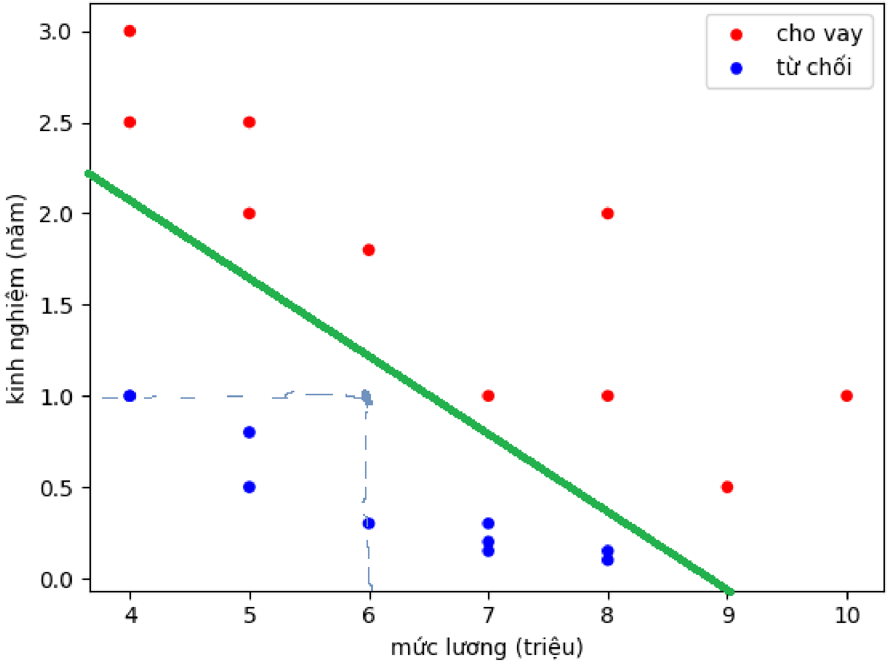
Machine Learning



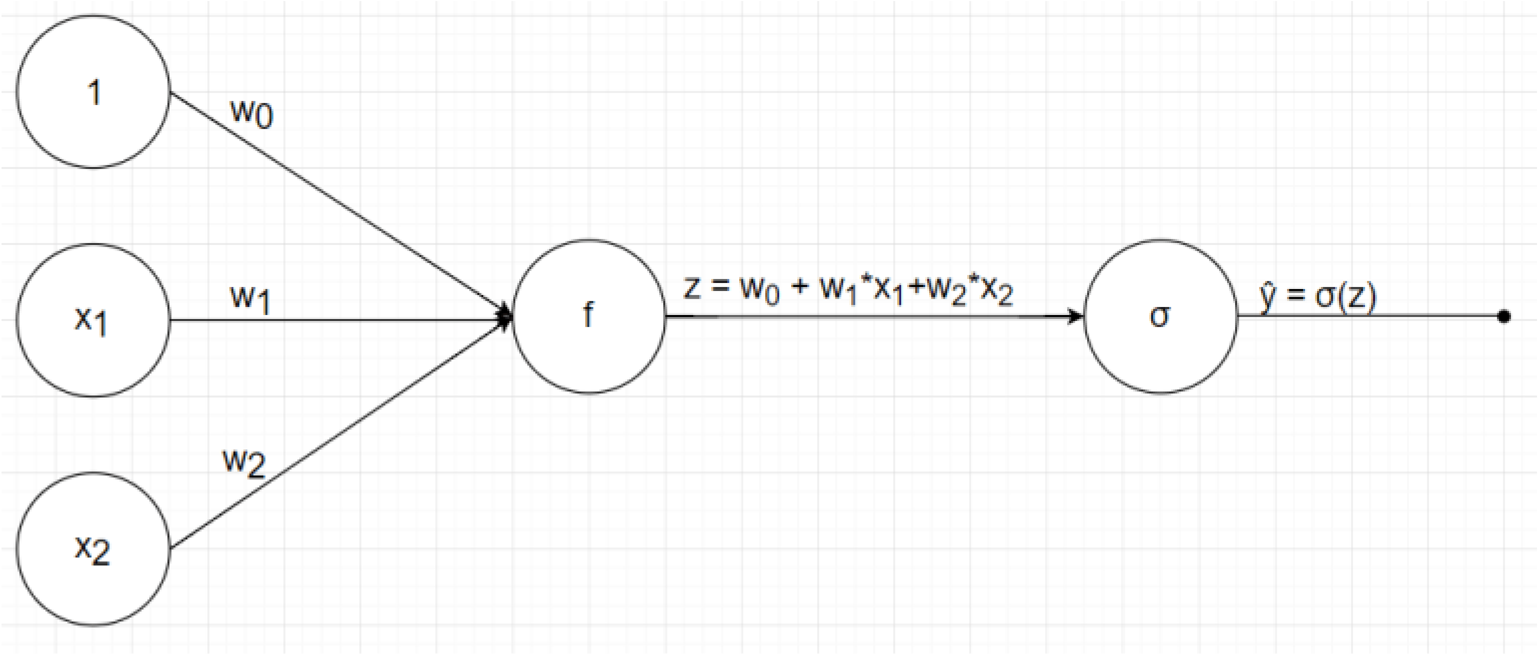
Linear regression



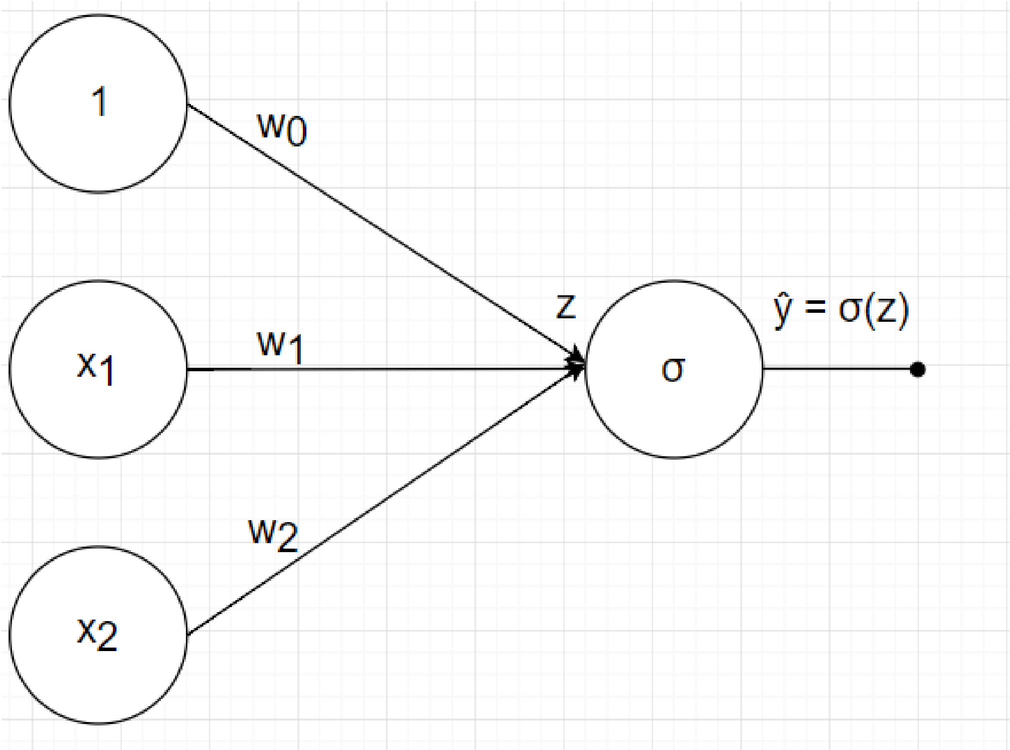
Logistic regression



Logistic regression



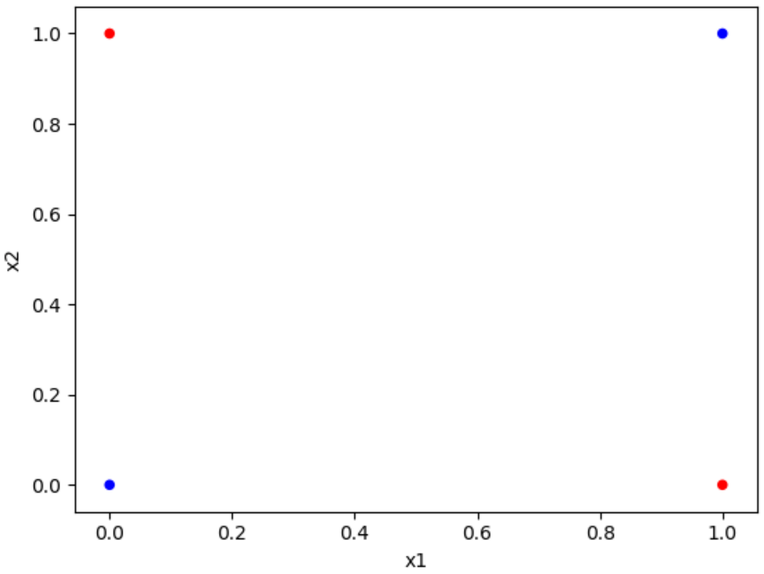
Logistic regression

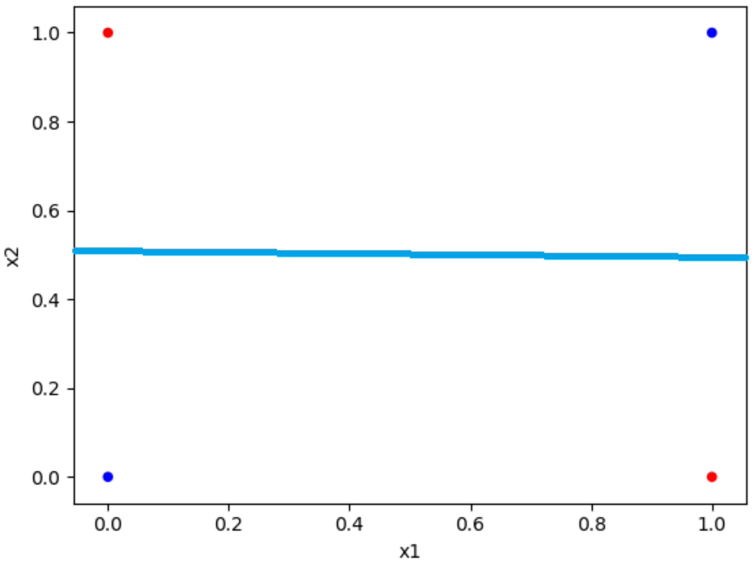


How?

1. Define the problem
2. Visualize the data
3. Choose the model
4. Define loss function
5. Minimize loss function
6. Use the model to predict new data.

XOR problem


1.0
0.8
0.6
0.4
0.2
0.0
x2
0.0
0.2
0.4
0.6
0.8
1.0
x1

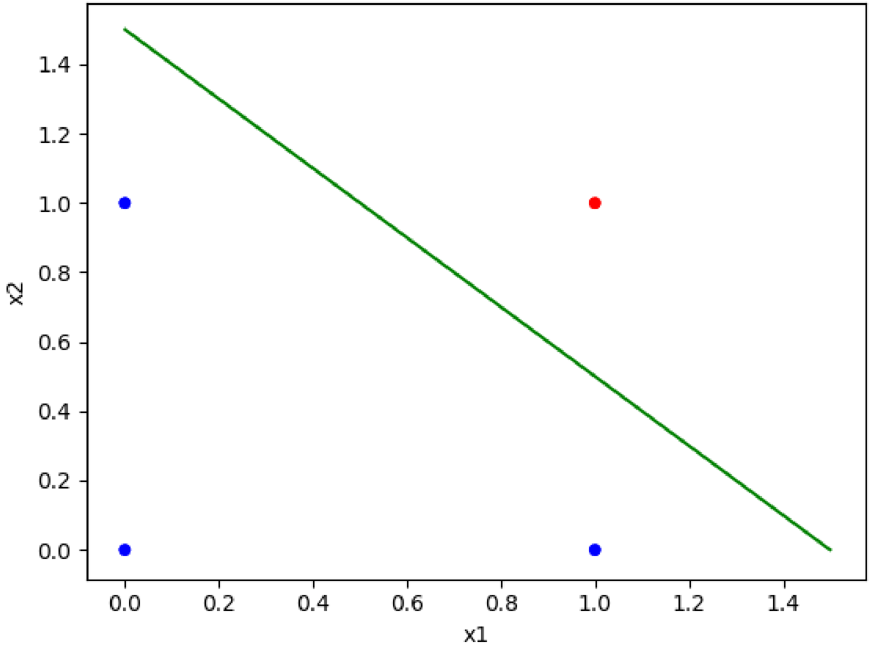

1.0
0.8
0.6
0.4
0.2
0.0
x2
0.0
0.2
0.4
0.6
0.8
1.0
x1

How to solve XOR problem

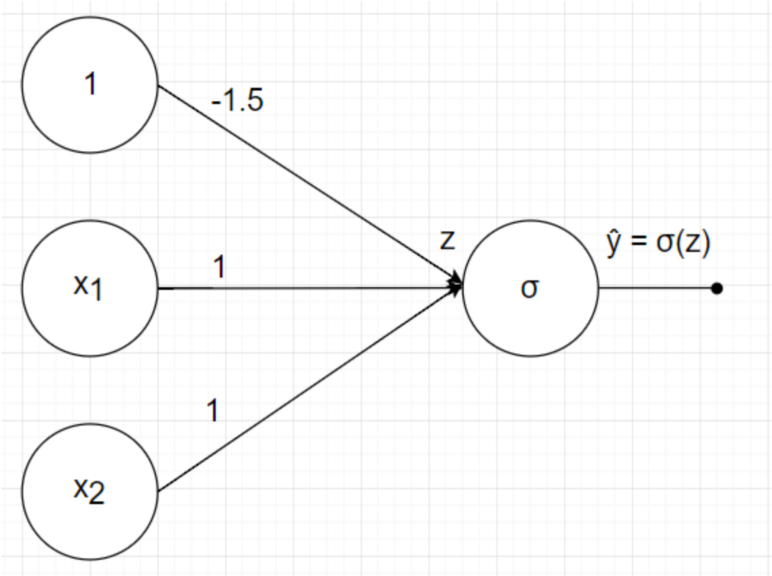
A XOR B = (NOT(A AND B) AND (A OR B))

A	B	**A XOR B**	A AND B	NOT(A AND B)	A OR B	**(NOT(A AND B) AND (A OR B))**
0	0	**0**	0	1	0	**0**
0	1	**1**	0	1	1	**1**
1	0	**1**	0	1	1	**1**
1	1	**0**	1	0	1	**0**

AND



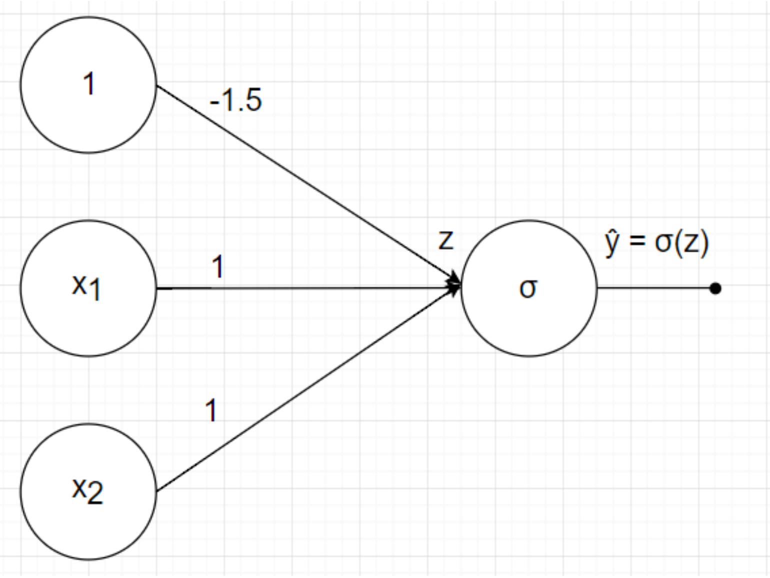
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
x2
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
x1



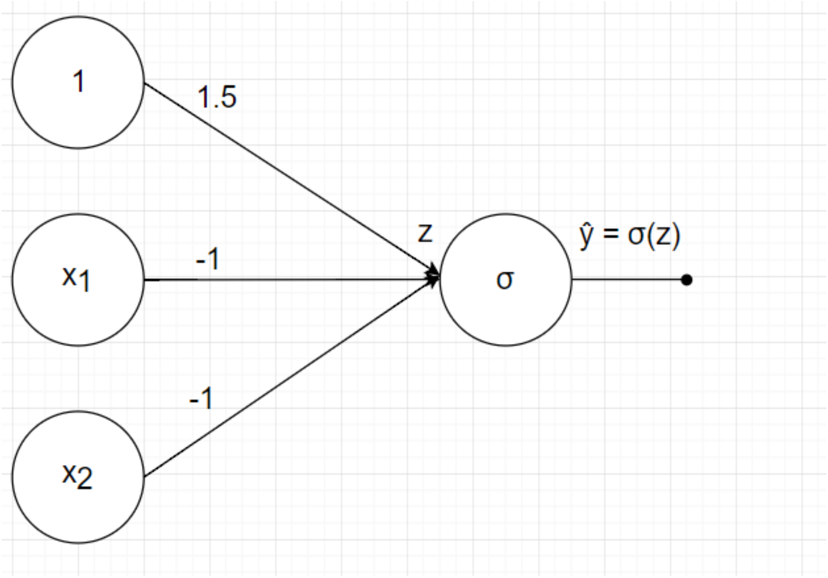
1
-1.5
x1
1
x2
1
z
σ
ŷ = σ(z)

NOT AND

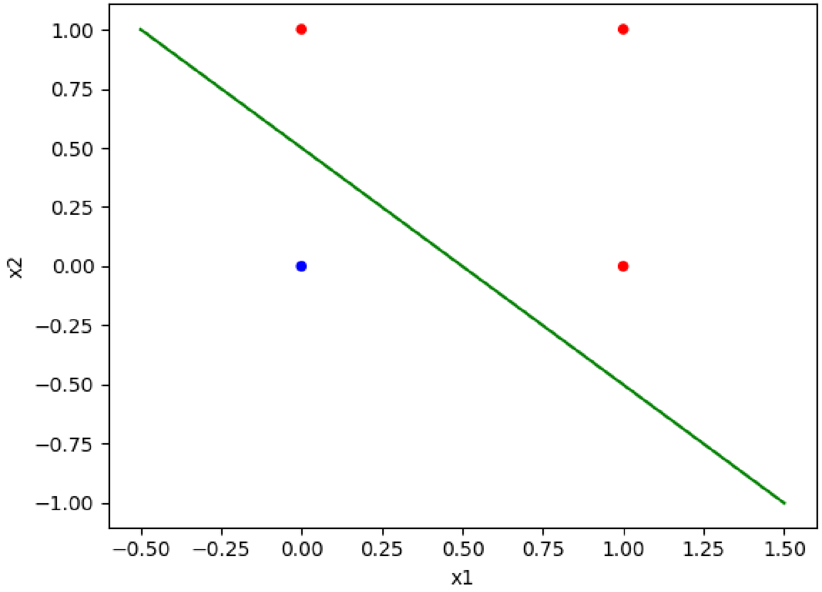
AND

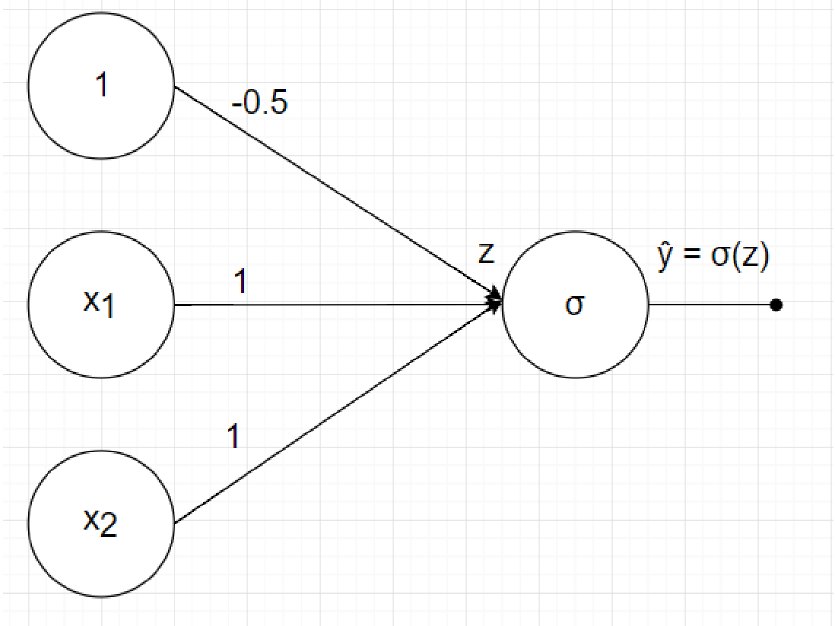


NOT AND



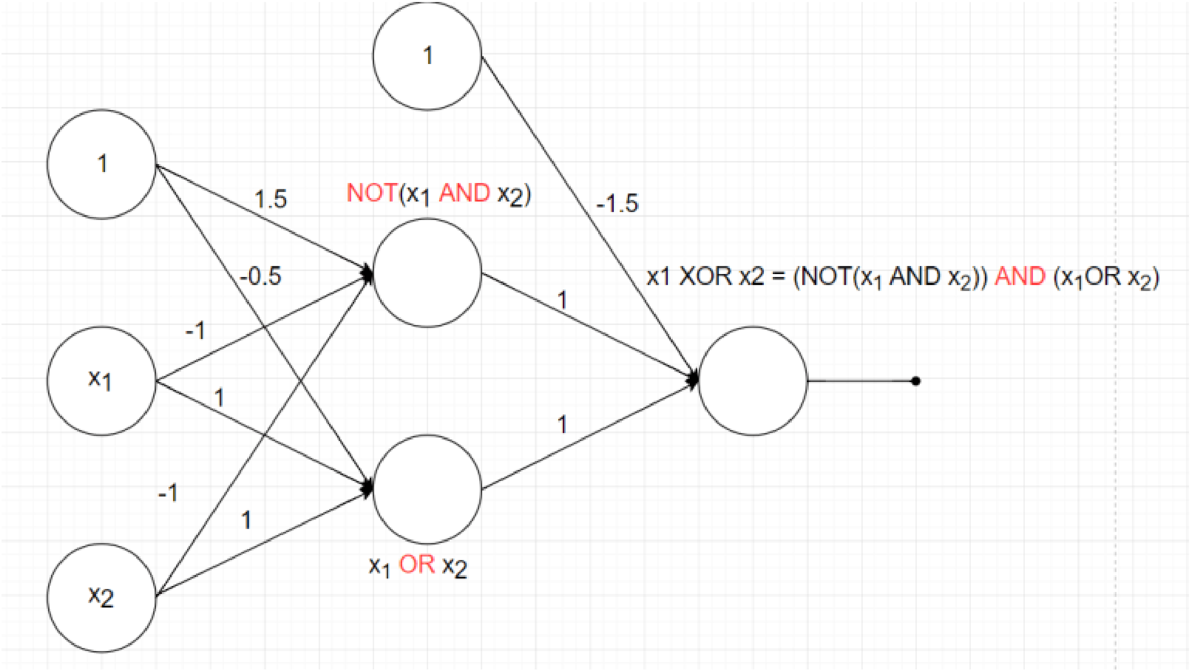
OR


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
x2
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
1.25
1.50
x1

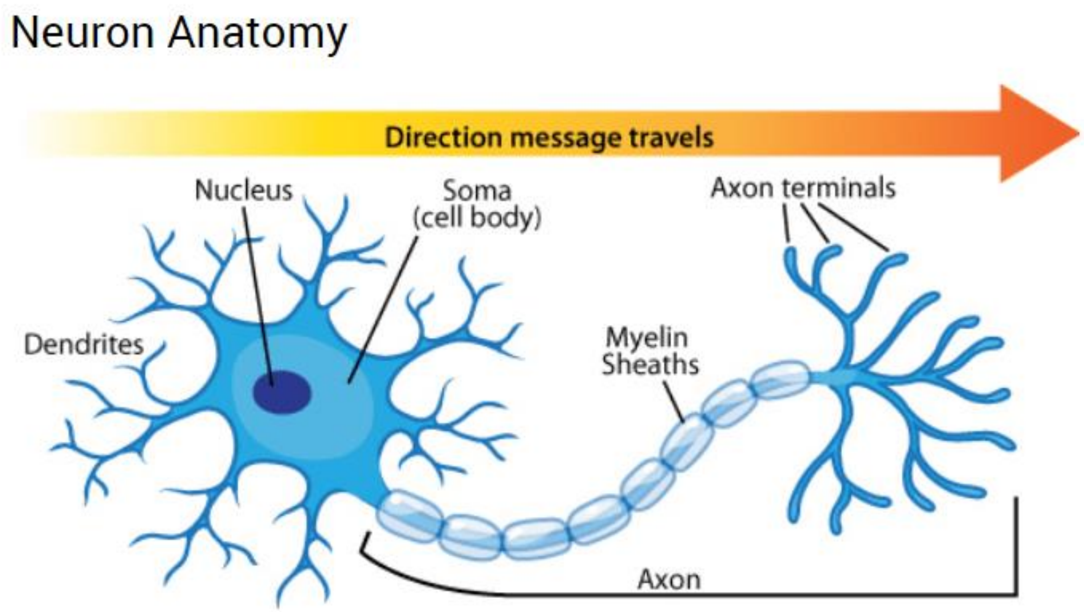

1
-0.5
x1
1
x2
1
z
σ
ŷ = σ(z)

XOR

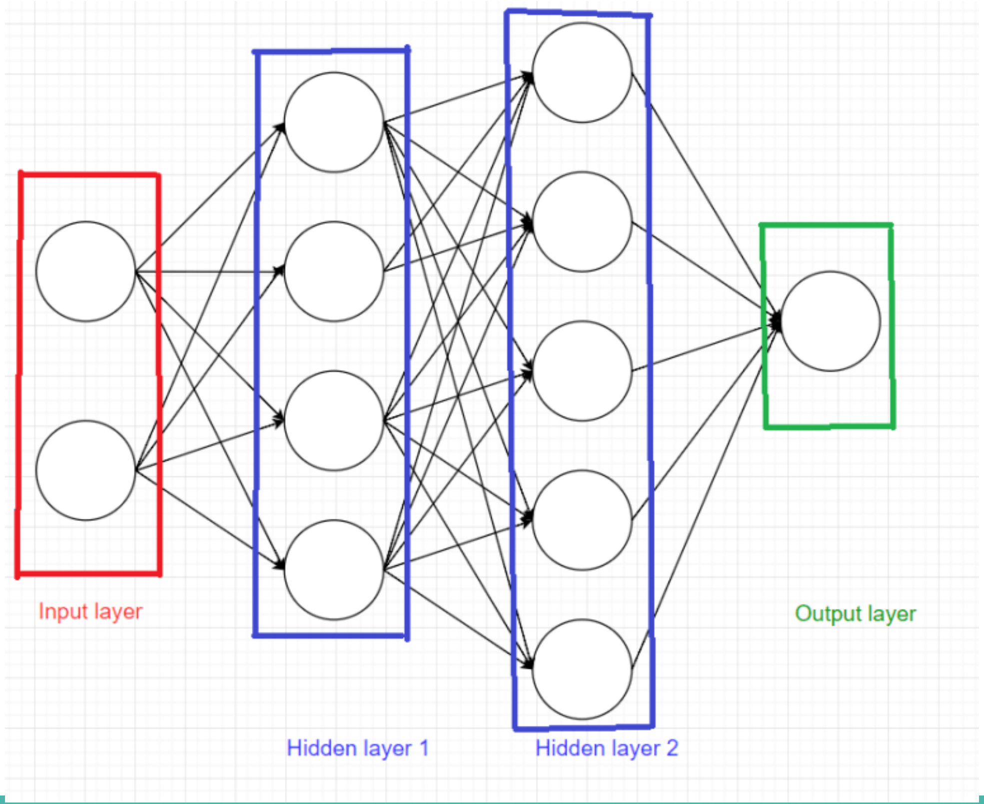
A XOR B = (NOT(A AND B) AND (A OR B))



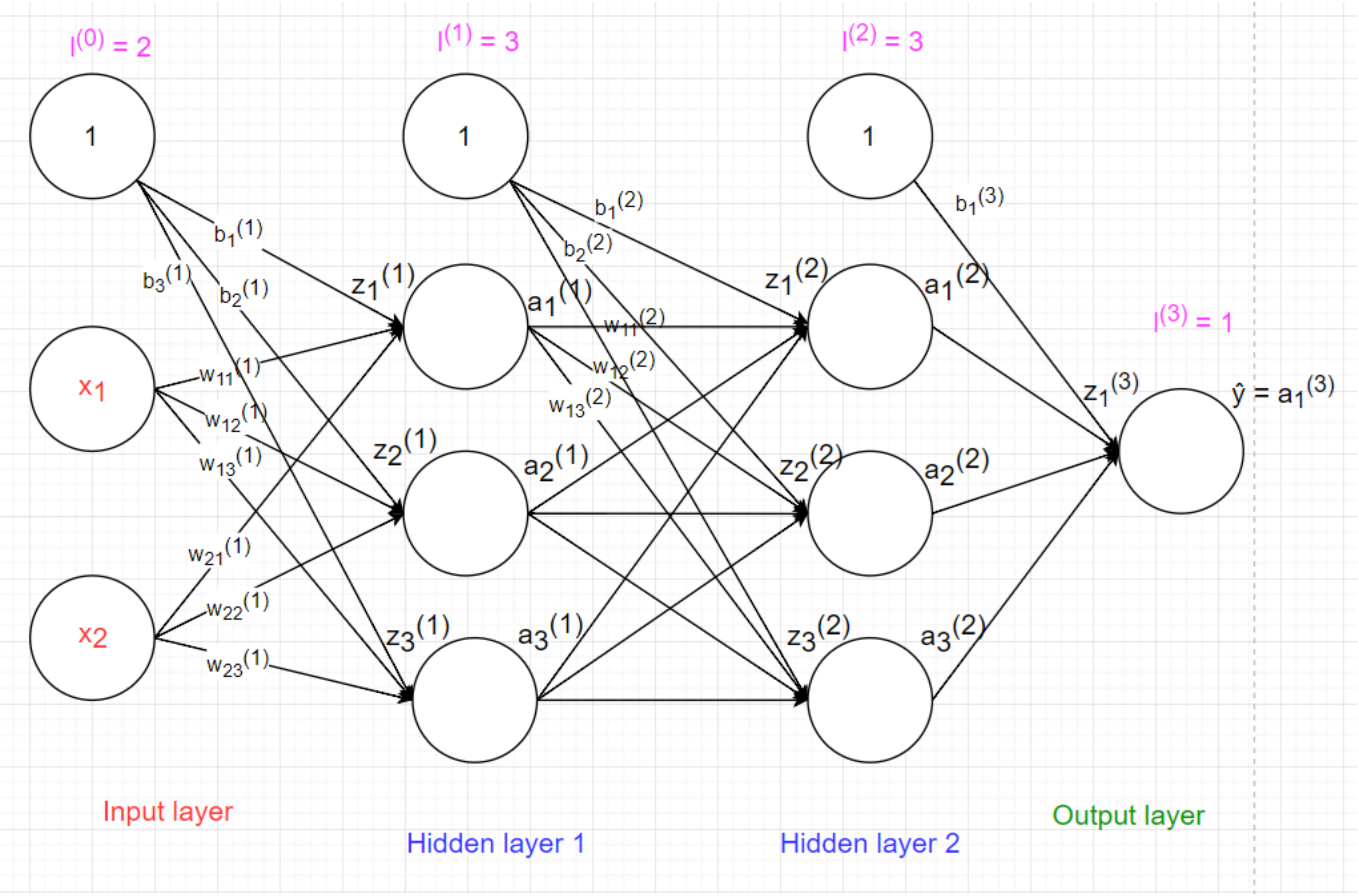
Neural network inspiration



Neural network



Notation



l(0) = 2
l(1) = 3
l(2) = 3
l(3) = 1
1
1
1
x1
x2
b1(1)
b2(1)
b3(1)
w11(1)
w12(1)
w13(1)
w21(1)
w22(1)
w23(1)
z1(1)
z2(1)
z3(1)
a1(1)
a2(1)
a3(1)
b1(2)
b2(2)
w11(2)
w12(2)
w13(2)
z1(2)
z2(2)
z3(2)
a1(2)
a2(2)
a3(2)
b1(3)
z1(3)
ŷ = a1(3)
Input layer
Hidden layer 1
Hidden layer 2
Output layer

Feedforward

Với node thứ i trong layer l có bias $b_i^{(l)}$ thực hiện 2 bước:

- Tính tổng linear: $z_i^{(l)} = \sum_{j=1}^{l^{(l-1)}} a_j^{(l-1)} * w_{ji}^{(l)} + b_i^{(l)}$
- Áp dụng activation function: $a_i^{(l)} = \sigma(z_i^{(l)})$

Feedforward

$$z^{(1)} = \begin{bmatrix} z_1^{(1)} \\ z_2^{(1)} \\ z_3^{(1)} \end{bmatrix} = \begin{bmatrix} a_1^{(0)} * w_{11}^{(1)} + a_2^{(0)} * w_{21}^{(1)} + a_3^{(0)} * w_{31}^{(1)} + b_1^{(1)} \\ a_1^{(0)} * w_{12}^{(1)} + a_2^{(0)} * w_{22}^{(1)} + a_3^{(0)} * w_{32}^{(1)} + b_2^{(1)} \\ a_1^{(0)} * w_{13}^{(1)} + a_2^{(0)} * w_{23}^{(1)} + a_3^{(0)} * w_{33}^{(1)} + b_3^{(1)} \end{bmatrix}$$

$$= (W^{(1)})^T * a^{(0)} + b^{(1)}$$

$$a^{(1)} = \sigma(z^{(1)})$$

$$z^{(2)} = (W^{(2)})^T * a^{(1)} + b^{(2)}, a^{(2)} = \sigma(z^{(2)})$$

Feedforward

$$z^{(2)} = (W^{(2)})^T * a^{(1)} + b^{(2)}$$
$$a^{(2)} = \sigma(z^{(2)})$$
$$z^{(3)} = (W^{(3)})^T * a^{(2)} + b^{(3)}$$
$$\hat{y} = a^{(3)} = \sigma(z^{(3)})$$

$$a^{(0)} \Rightarrow z^{(1)} \Rightarrow a^{(1)} \Rightarrow z^{(2)} \Rightarrow a^{(2)} \Rightarrow z^{(3)} \Rightarrow a^{(3)} = \hat{y}$$

Dataset

Lương	Thời gian làm việc
10	1
5	2
6	1.8
7	1

thì $n = 4, d = 2, x_1^{[1]} = 10, x_2^{[1]} = 1, x_1^{[3]} = 6, x_2^{[2]} = 2.$

Matrix presentation

$$X = \begin{bmatrix} x_1^{[1]} & x_2^{[1]} & \dots & x_d^{[1]} \\ x_1^{[2]} & x_2^{[2]} & \dots & x_d^{[2]} \\ \dots & \dots & \dots & \dots \\ x_1^{[n]} & x_2^{[n]} & \dots & x_d^{[n]} \end{bmatrix} = \begin{bmatrix} -(x^{[1]})^T- \\ -(x^{[2]})^T- \\ \dots \\ -(x^{[n]})^T- \end{bmatrix}$$

Matrix presentation

$$Z^{(i)} = \begin{bmatrix} z_1^{(i)[1]} & z_2^{(i)[1]} & \dots & z_{l^{(i)}}^{(i)[1]} \\ z_1^{(i)[2]} & z_2^{(i)[2]} & \dots & z_{l^{(i)}}^{(i)[2]} \\ \dots & & & \\ z_1^{(i)[n]} & z_2^{(i)[n]} & \dots & z_{l^{(i)}}^{(i)[n]} \end{bmatrix} = \begin{bmatrix} -(z^{(i)[1]})^T- \\ -(z^{(i)[2]})^T- \\ \dots \\ -(z^{(i)[n]})^T- \end{bmatrix}$$

Matrix presentation

$$
Z^{(1)} = \begin{bmatrix} (z^{(1)[1]})^T \\ (z^{(1)[2]})^T \\ \dots \\ (z^{(1)[n]})^T \end{bmatrix} = \begin{bmatrix} (x^{[1]})^T * w^{(1)} + (b^{(1)})^T \\ (x^{[2]})^T * w^{(1)} + (b^{(1)})^T \\ \dots \\ (x^{[n]})^T * w^{(1)} + (b^{(1)})^T \end{bmatrix}
$$

$$
= X * W^{(1)} + \begin{bmatrix} (b^{(1)})^T \\ (b^{(1)})^T \\ \dots \\ (b^{(1)})^T \end{bmatrix}
$$

$$
= X * W^{(1)} + b^{(1)}
$$

Matrix presentation

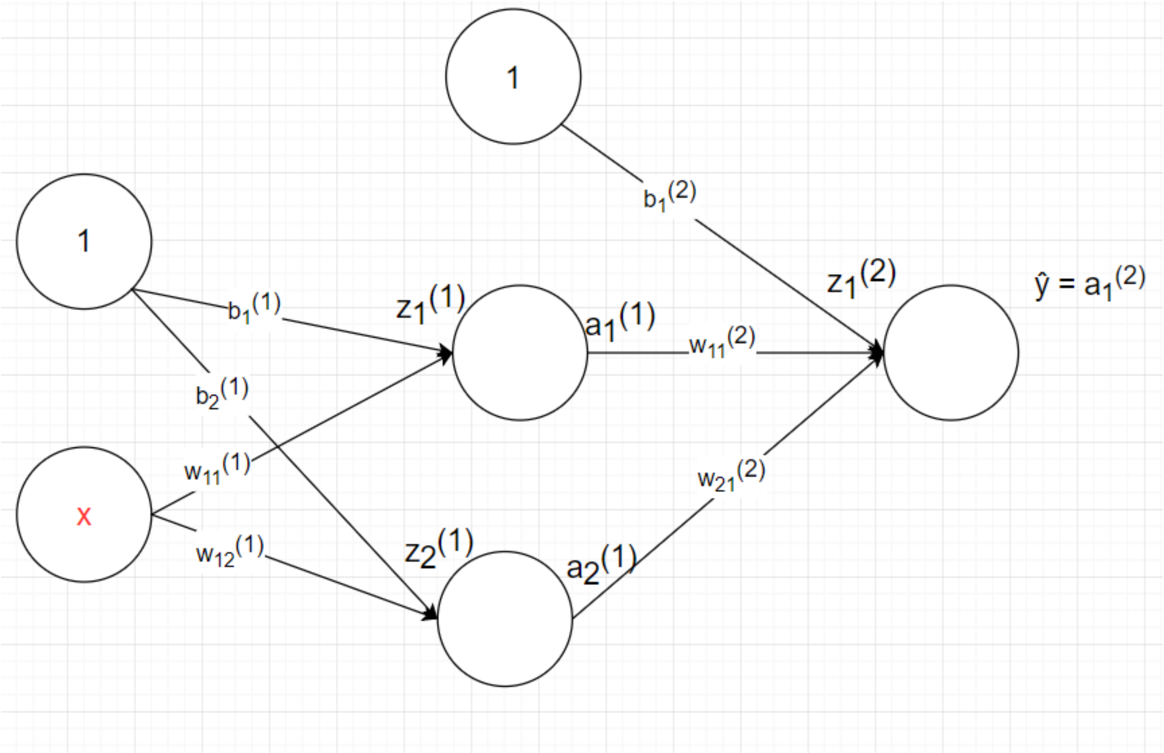
$$A^{(1)} = \sigma(Z^{(1)})$$

$$Z^{(2)} = A^{(1)} * W^{(2)}, A^{(2)} = \sigma(Z^{(2)})$$

$$Z^{(3)} = A^{(2)} * W^{(3)}$$

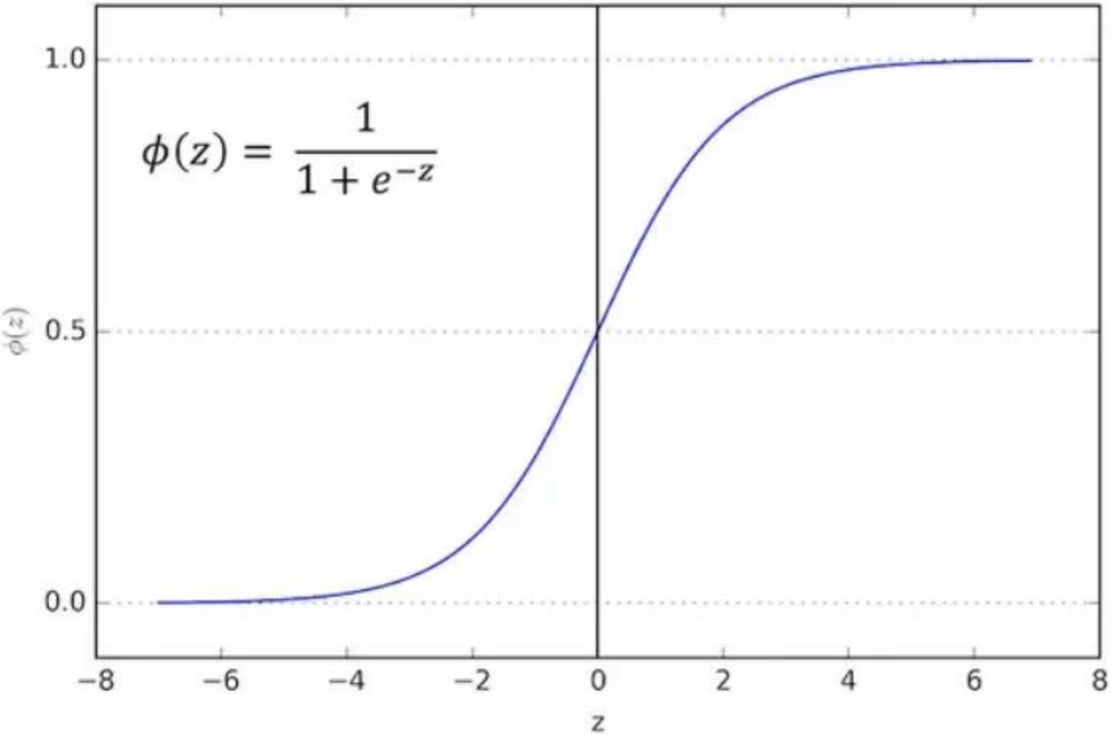
$$\hat{Y} = A^{(3)} = \sigma(Z^{(3)})$$

Why non-linear activation?

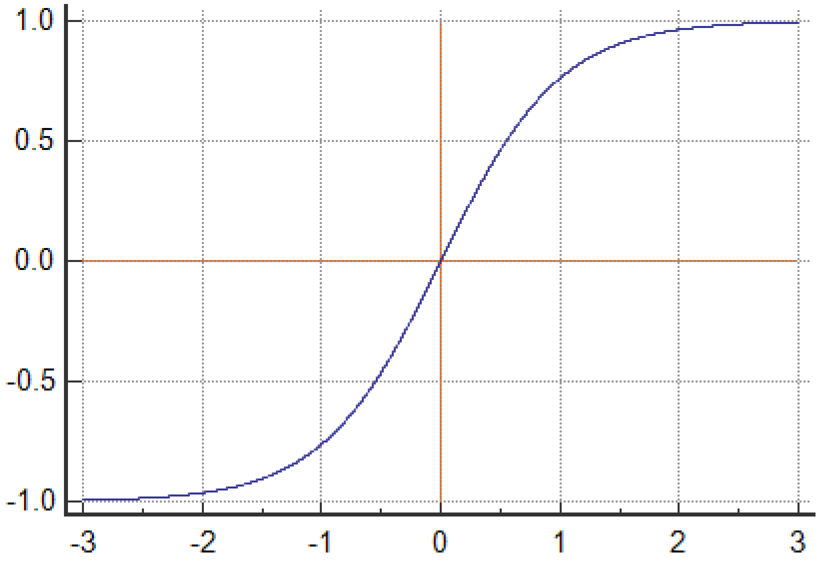


Suppose we use linear activation function, what is the output?

Sigmoid

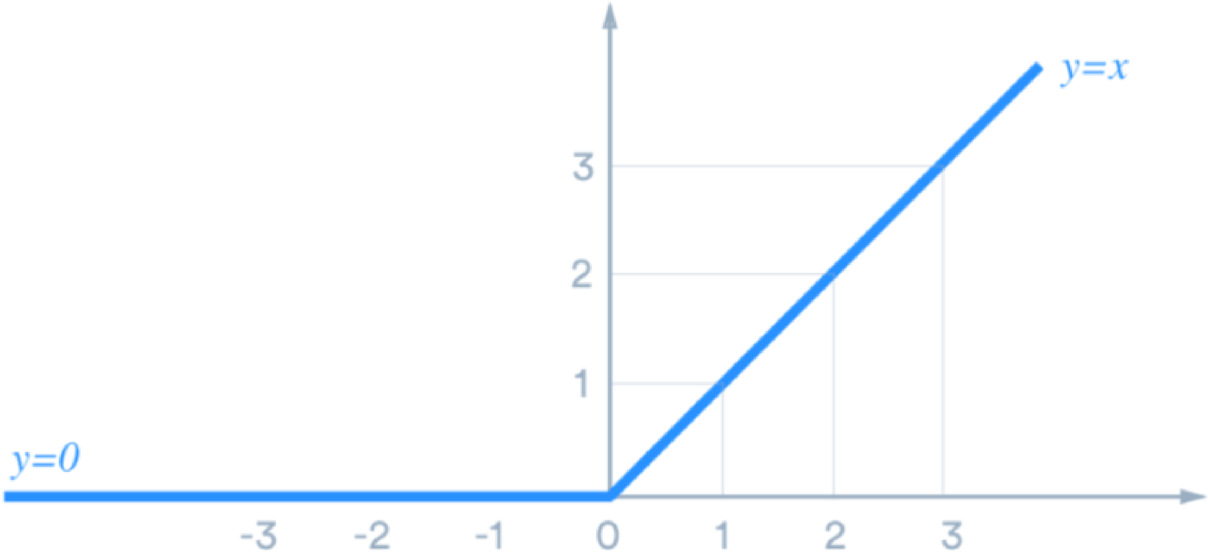


Tanh



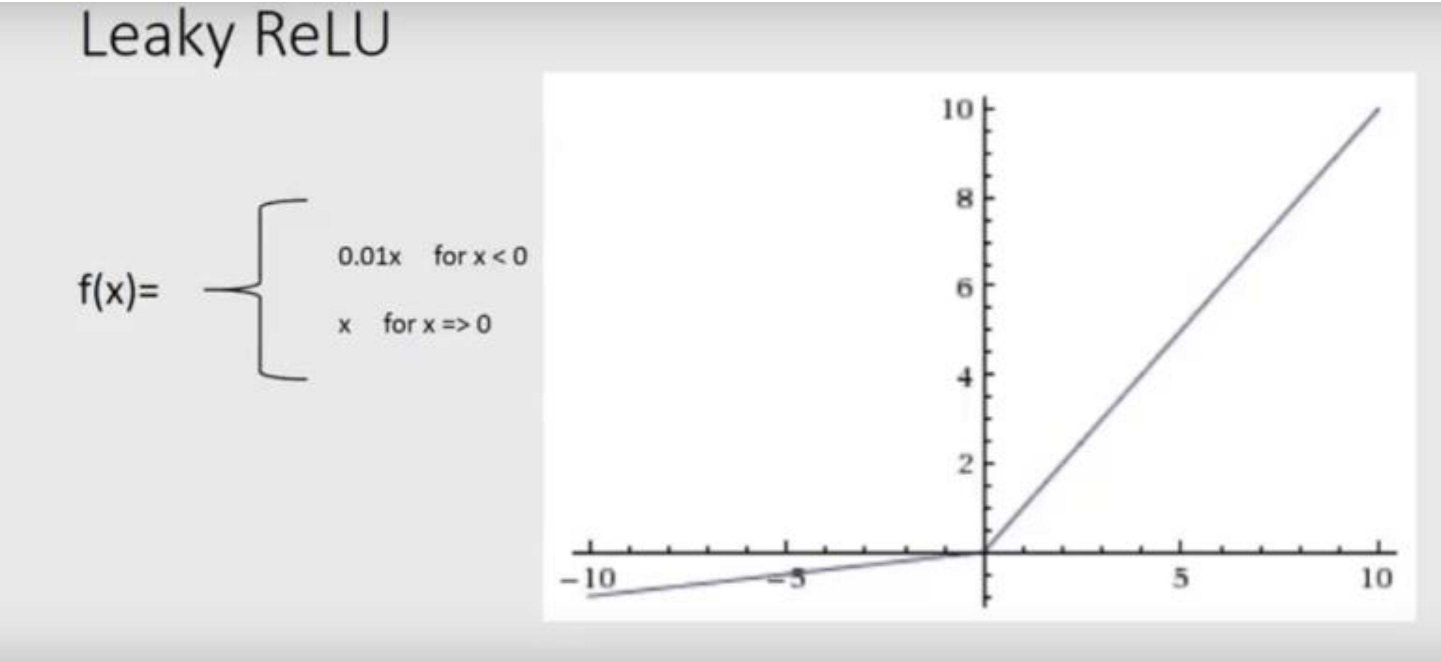
$$g(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

Relu

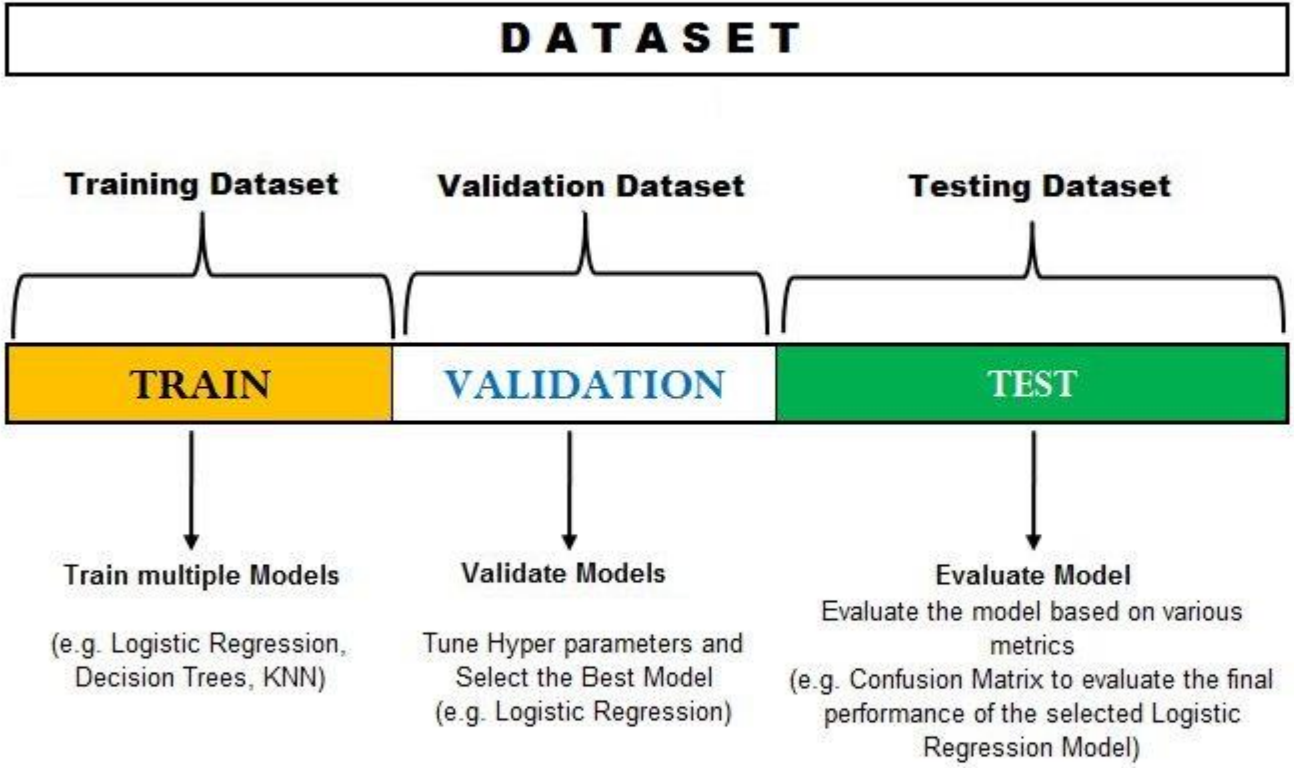


y = max(x, 0)

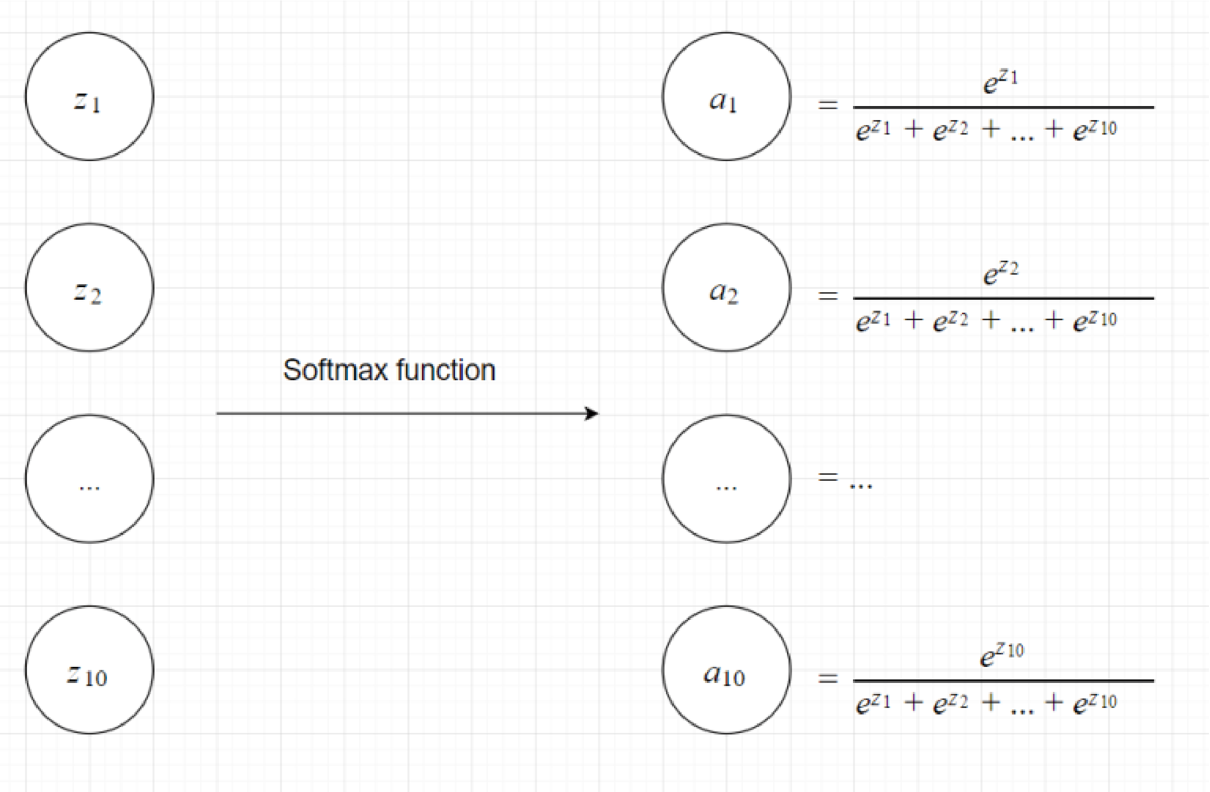
Leaky ReLU



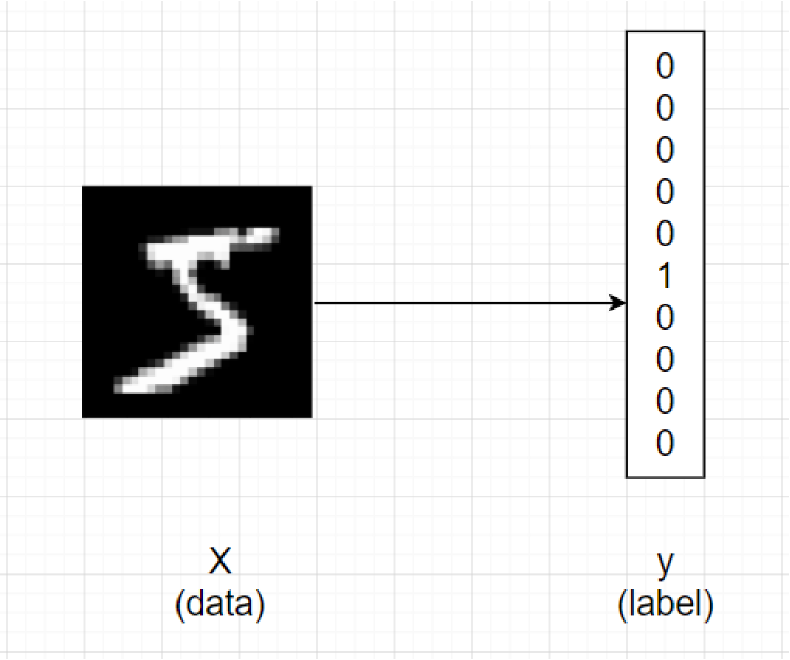
Training set/Validation set/Test set

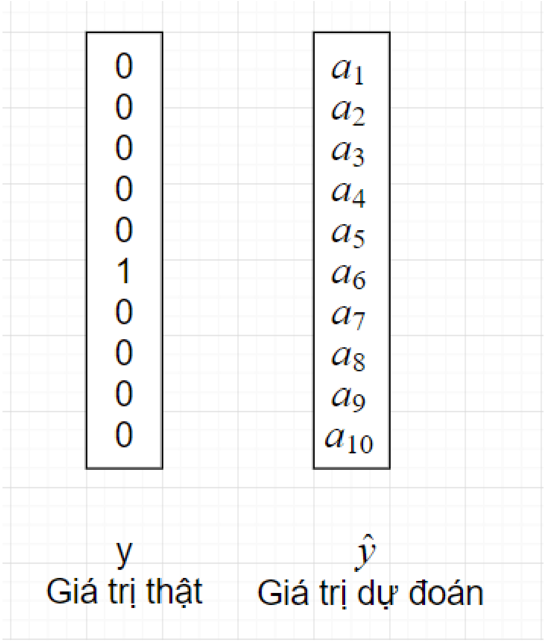


Softmax function

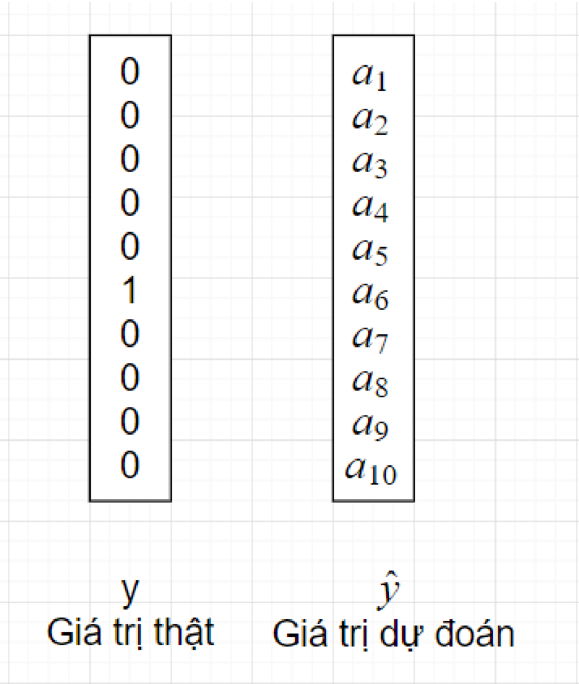


One-hot encoding





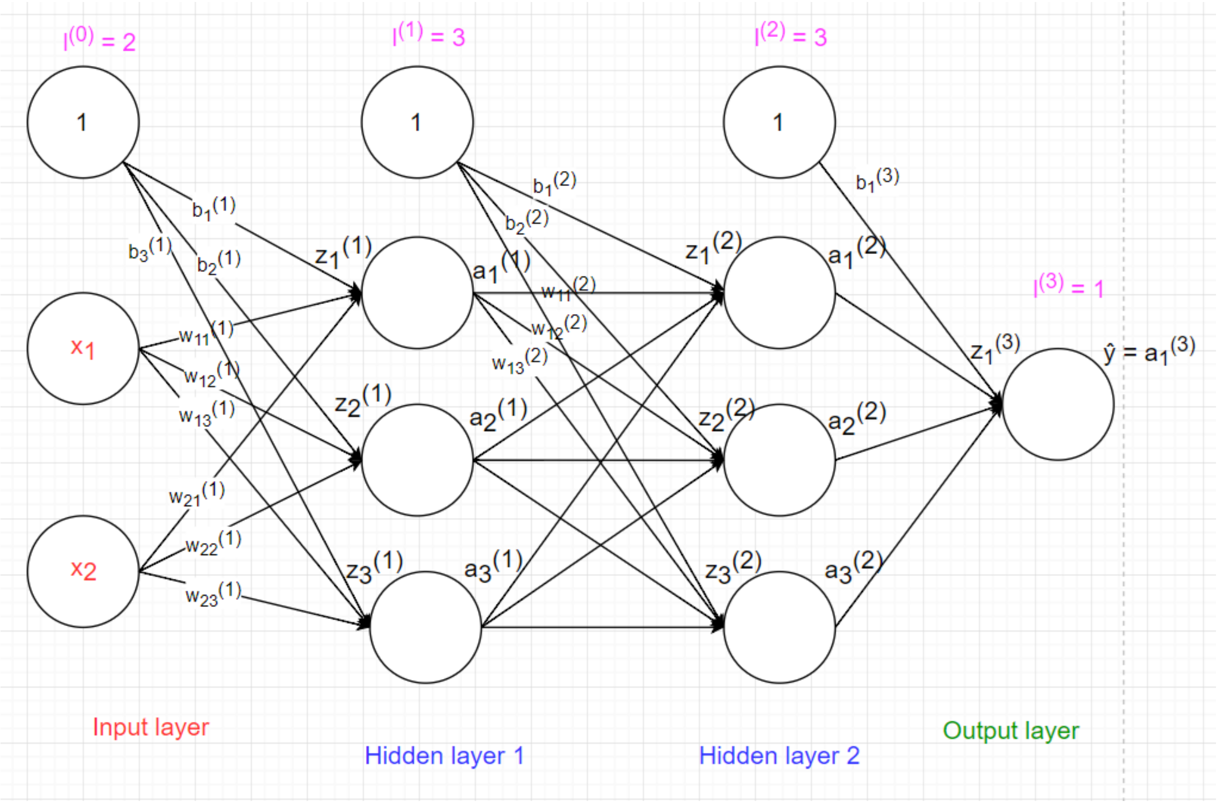
Cross-entropy loss



$$L = -\sum_{i=1}^{10} y_i * log(\hat{y_i})$$

Similar to binary_crossentropy

Backpropagation



Q&A

The end