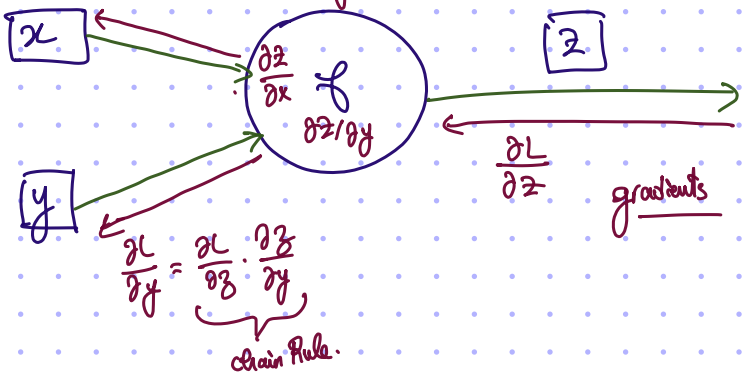
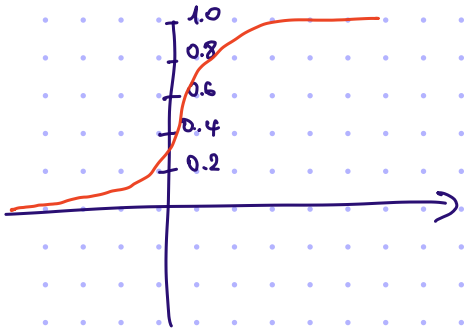
Backpropagation and Neural Networks: (XOR Problem Example):

Rawen Jendoubi
13/10/2025
(22:21)

$\partial L/\partial x = \frac{\partial L}{\partial z} \cdot \frac{\partial z}{\partial x}$ = "local gradient"



Sigmoid fuction:

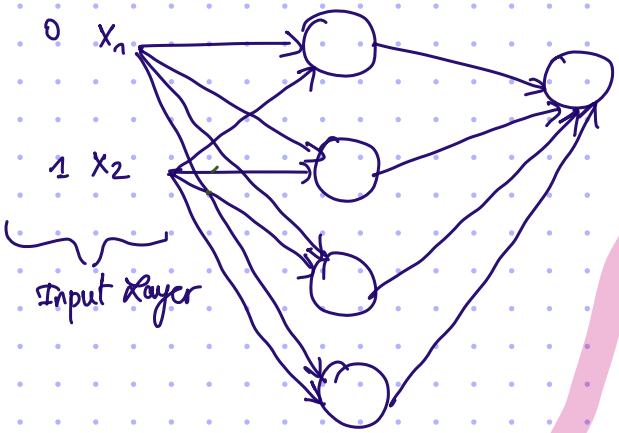


Sigmoid activation function

$$f(x) = \frac{1}{1+e^{-x}}$$

Neural Network (XOR Problem)

XOR Problem:



Possible Combination

(0,0)
(0,1)
(1,0)
(1,1)

Hidden Layer in possible Combinations

output Layer = either 0 or 1

C Code:

```
# define INPUT_NEURONS 2
# define output_NEURON 1
# define HIDDEN_LAYER
# define TRAINING_SAMPLES 4
# define LEARNING_RATE 0.5
# define EPOCHS 10000
```

Input

double Input [TRAINING_SAMPLES] [INPUT_NEURONS] = { {0,0}, {0,1}, {1,0}, {1,1} }

(X_1, X_2)

Weight

The weight array depends on what the Network layer Connects to

① if we have, one single_layer Percepton (no hidden layer)

→ double weight [INPUT_NEURONS] [output_NEURONS];

→ each Neuron has one weight connection to each output Neuron