Neural networks

Ens'IA

Ensimag 2022-2023

November 28, 2022

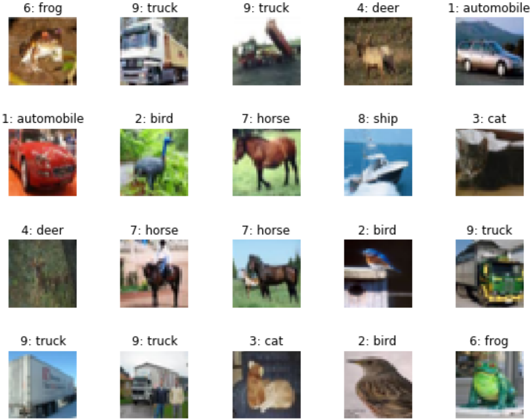
Ens'IA

The program:

- Introduction
- My first neuron
- **My first neural network**
- My first convolution
- AI challenge
- Reinforcement Learning?
- GANs?

And after? → you tell us!

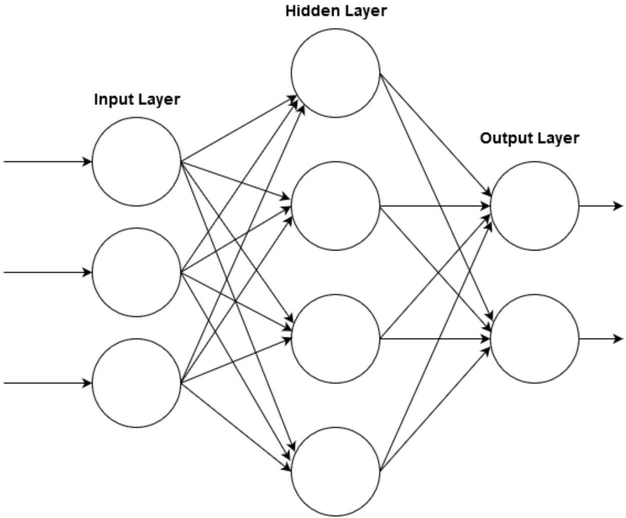
Reminder



Images classification
→ We need a function f
such that $f(x) = y$

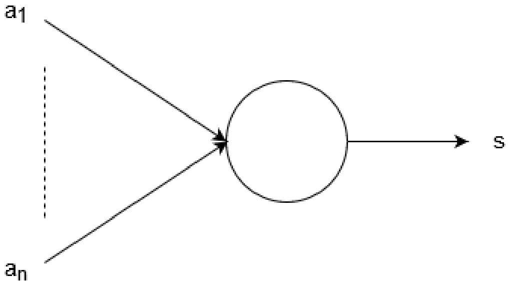
How to approximate this function? → Neural networks

Neural network



→ Succession of neuron layers

Sigmoid neuron



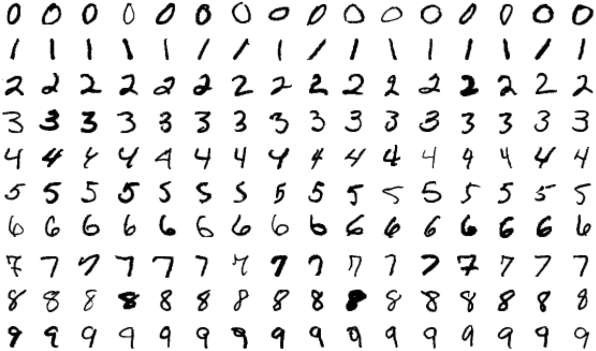
$a_1, ..., a_n \in [0, 1]$
$s = \sigma(\sum_{i=0}^{n} a_i * w_i + b)$ with $\sigma(x) = \frac{1}{1+e^{-x}}$

Training loop

How to train the neuron?

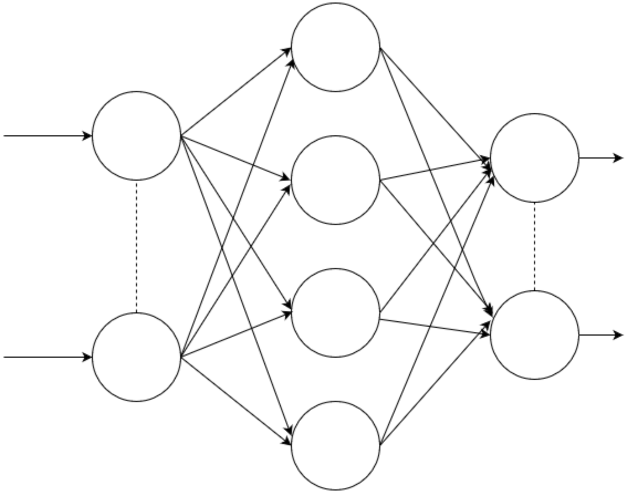
```
For each epoch
    For each x
        Forward pass
        Loss calculation
        Backward pass
    (Accuracy calculation)
```

Image classification

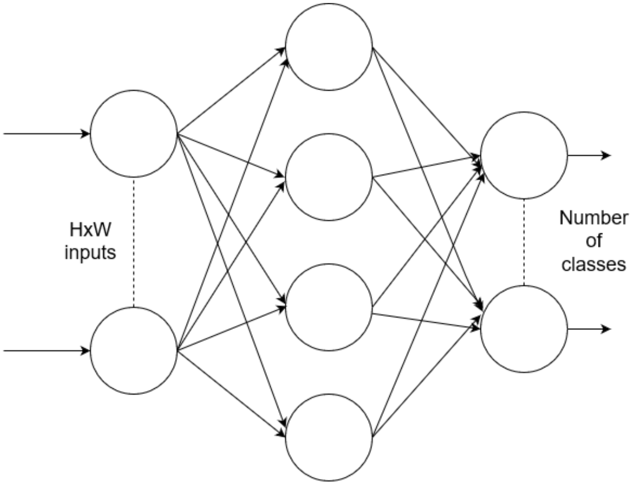


Images of numbers between 0 and 9: 10 classes
Black & white images
28x28 size

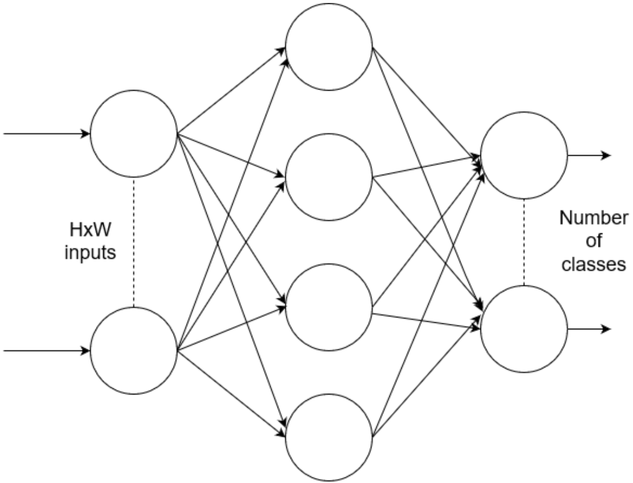
Neural network



Neural network

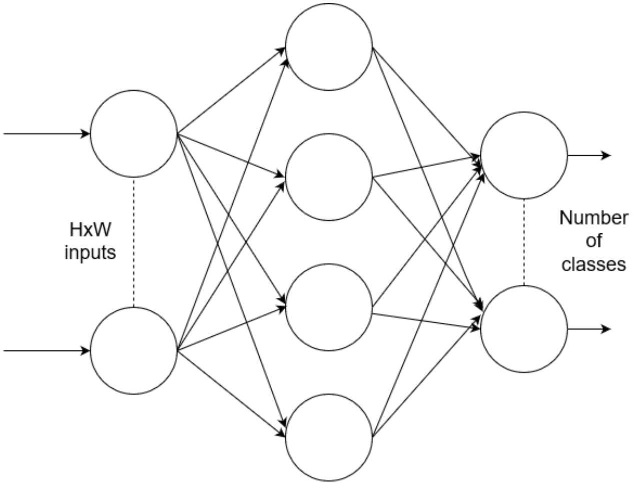


Neural network



Input: 1D vector

Neural network



Input: 1D vector

Output: 1 hot encoding

Training loop

How to train the neuron?

```
For each epoch
    For each x
        Forward pass
        Loss calculation
        Backward pass
    (Accuracy calculation)
```

Neural network parametrization

What about everything between the input and the output?

Neural network parametrization

What about everything between the input and the output?
→ We need to find the best hidden layer/neuron count

Neural network parametrization

What about everything between the input and the output?
→ We need to find the best hidden layer/neuron count

Which learning rate, batch size, epoch count and loss?

Neural network parametrization

What about everything between the input and the output?
→ We need to find the best hidden layer/neuron count

Which learning rate, batch size, epoch count and loss?
→We need to find them :3

To go further...

Back to the *sigmoid* activation function:

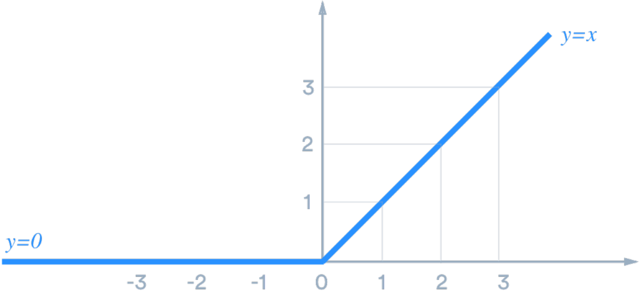
- Expensive to calculate
- *Vanishing Gradient*
- And more...

To go further...

Back to the *sigmoid* activation function:

- Expensive to calculate
- *Vanishing Gradient*
- And more...

In practice:
→ ReLU (*Rectified Linear Unit*)



To go further...

Back to the *backpropagation* and the gradient algorithm:

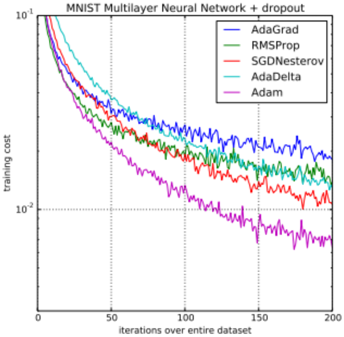
- Addition of a moment
- Adaptative η learning rate

To go further...

Back to the *backpropagation* and the gradient algorithm:

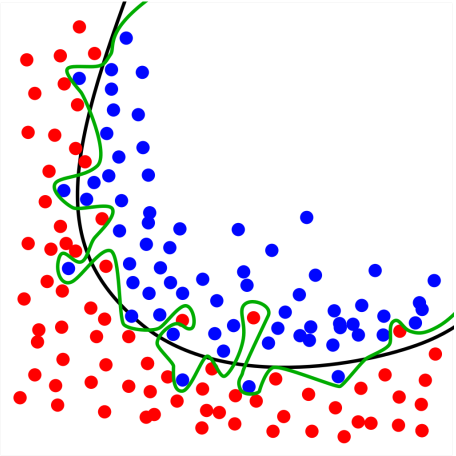
- Addition of a moment
- Adaptative η learning rate

In practice:
→ Adam



To go further...

A major problem: ***overfitting***



→ *Dropout*

Conclusion

It works great!
We have to search for the best model
Other neural network types exist (CNN, RNN, LSTM...)

Discord

Join us on Discord !

Useful to ask questions, contact us or to pass on information! → https://discord.gg/UgTRbRFqNv

To go further...

- Kaggle
- CS231N
- http://neuralnetworksanddeeplearning.com/
- http://www.deeplearningbook.org/