One of the most popular models for ML & DL: ANN

Where is an ANN (Artificial Neural Network) used?

- ANNs are **inspired by the structure of the human brain**, the way our brains evolve every time we learn something new, and that is what the artificial neural network tries to mimic.
- ANNs are particularly useful for **solving nonlinear problems**.
- They can be used in a **wide range of applications**, from image processing and face recognition to speech recognition and generation, stock market predictions and many more.

How do ANNs work?

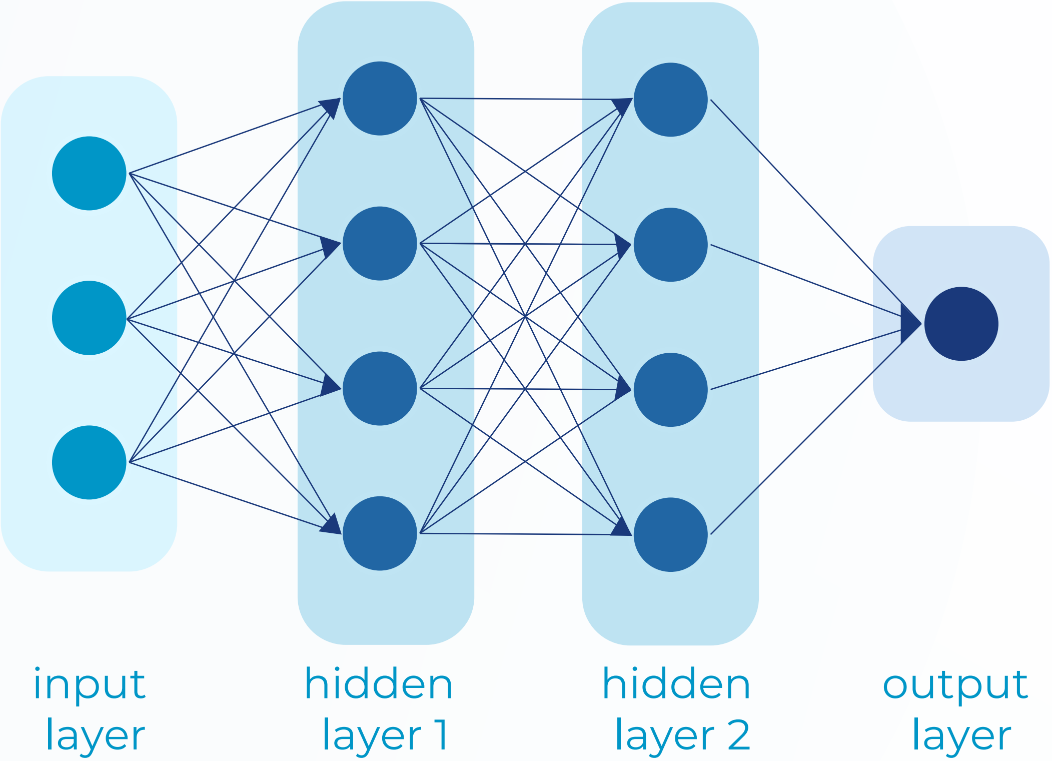
AI | BUSINESS SCHOOL

1. The input layer takes the input data and sends it to the hidden layers

2. The hidden layers transfer the information to the output layer

3. The output layer calculates the output

▶ The data is transferred through the network until it reaches the neurons of the output layer – **“forward propagation”**



What are perceptrons?

The ANN layers are made up of a number of interconnected nodes that are called **“perceptrons”** and these mimic human neurons.

Recalling two familiar terms from regression: weight & bias

Weight

The impact of the input on the output

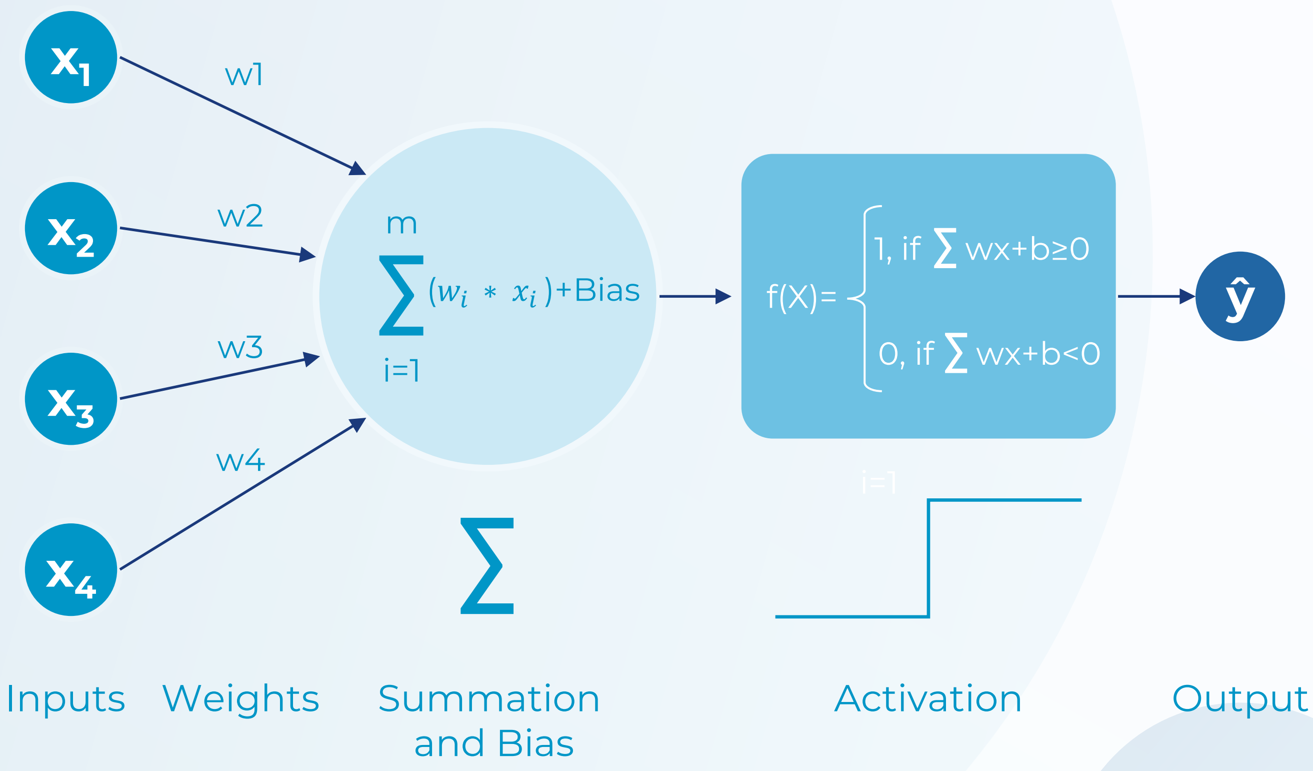
Bias

The equivalent to the intercept in linear regression

Activation functions transform the input

How do activation functions work?

They work like a threshold and determine the output between the desired range (e.g., 0,1 or -1,1) so that they can **activate** or **deactivate** the perceptron and determine whether it will **pass its value to the next layer.**



x_1 x_2 x_3 x_4

w1 w2 w3 w4

$$\sum_{i=1}^{m}(w_i * x_i)+\text{Bias}$$

$$f(X)=\begin{cases}1, \text{ if } \sum wx+b\geq 0\\ 0, \text{ if } \sum wx+b<0\end{cases}$$

$\hat{y}$

Σ

Inputs

Weights

Summation and Bias

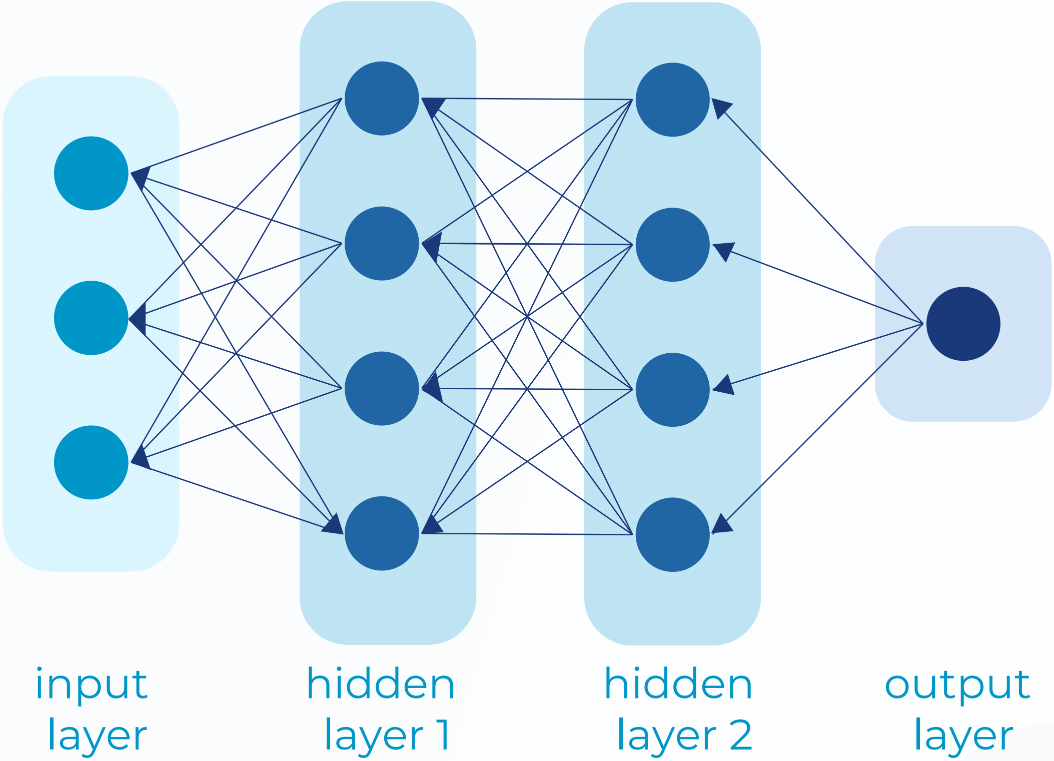
Activation

Output

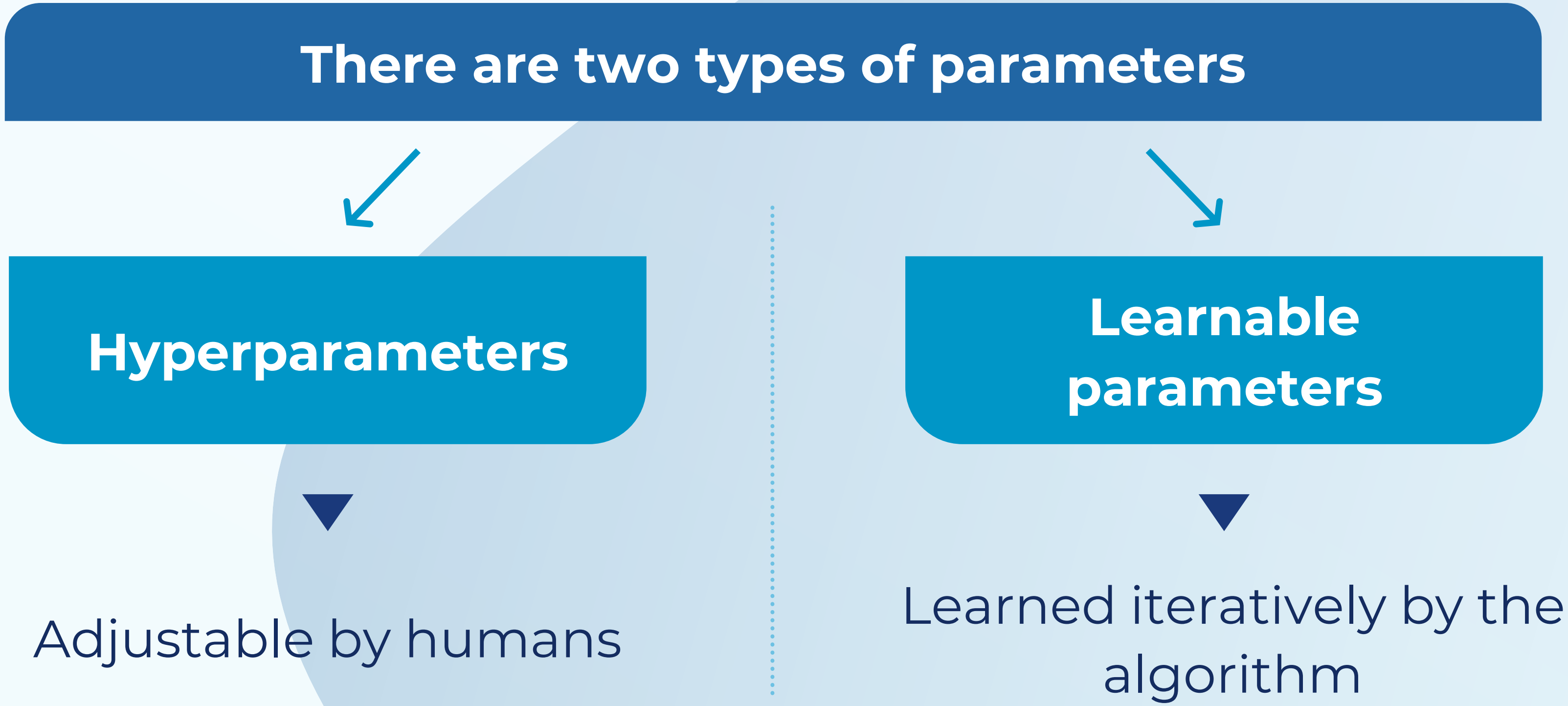
Evaluating & improving the model: backpropagation

What is backpropagation?

After going all the way through forward propagation and comparing the results with the ground truth, we can use a method called **“Backpropagation”** to **improve** the model. This means **going backwards** to **do the crosscheck** and to adjust the weights and biases and to **minimize error**.



Improving the neural network by setting optimal parameters



There are two types of parameters

Hyperparameters

Adjustable by humans

Learnable parameters

Learned iteratively by the algorithm

Training a neural network

AI BUSINESS SCHOOL

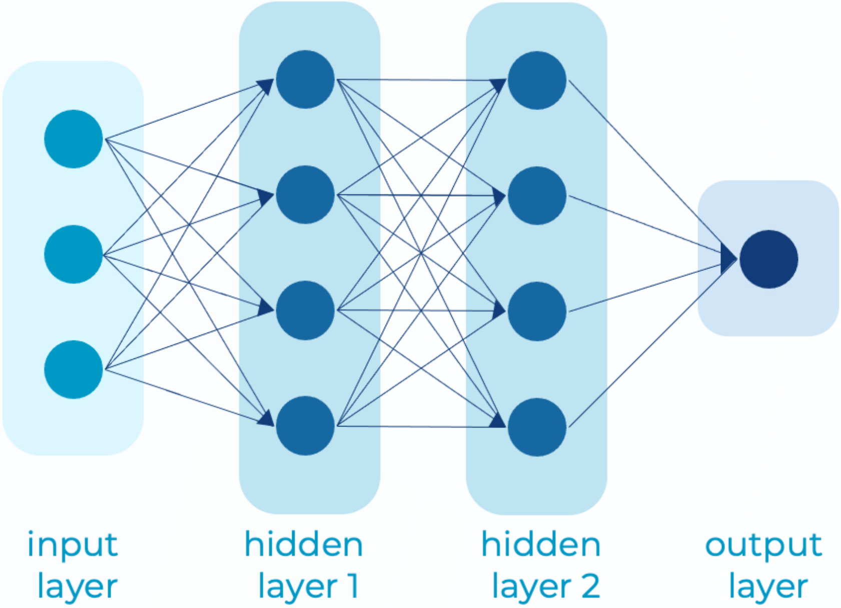
The process starts with forward propagation...

1. The data is received by the input layer and the inputs are multiplied by their corresponding weights & biases

2. Then, the activation function is applied

3. The information is passed to the next layer and the same process is repeated

4. Finally, propagated data from the input layer to the output layer is used to make the predictions



Next step: calculating the loss

How does it work?

- The process itself is simply comparing the predicted output with the expected output.
- The aim is to minimize the loss which means more accurate predictions and hence a better performing model.

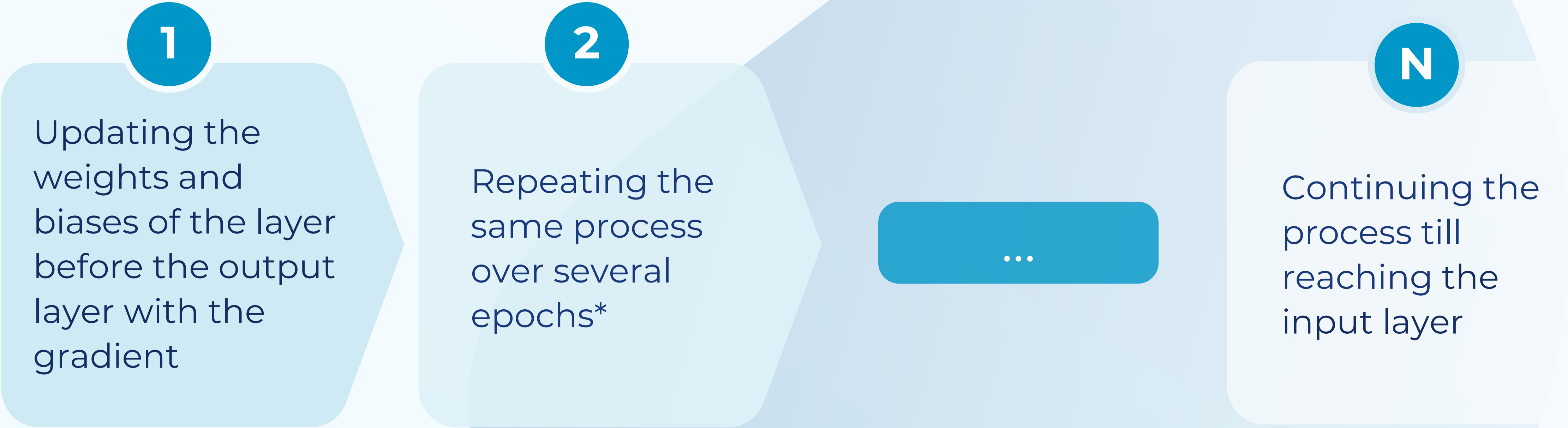
What happens when we change the weights and biases?

This process creates an impact on the loss which is called **“gradient”.**

Why is the gradient important?

Gradients are really useful to understand which parameters should be increased and which ones should be decreased and how much.

The final step: updating the weights



1 Updating the weights and biases of the layer before the output layer with the gradient

2 Repeating the same process over several epochs*

...

N Continuing the process till reaching the input layer

* One epoch = Iterating over the entire dataset once

Achieving a smooth learning process...

What are the challenges?

ANNs may take several epochs to minimize the loss and to train

Gradients can disrupt the process

How can we solve them?

We can scale down the gradients by multiplying them by a learning rate*

***Learning rate = A constant which scales the change in the weights with respect to the gradients**

…thanks to an optimal learning rate

Smaller learning rate

Slow & smooth learning

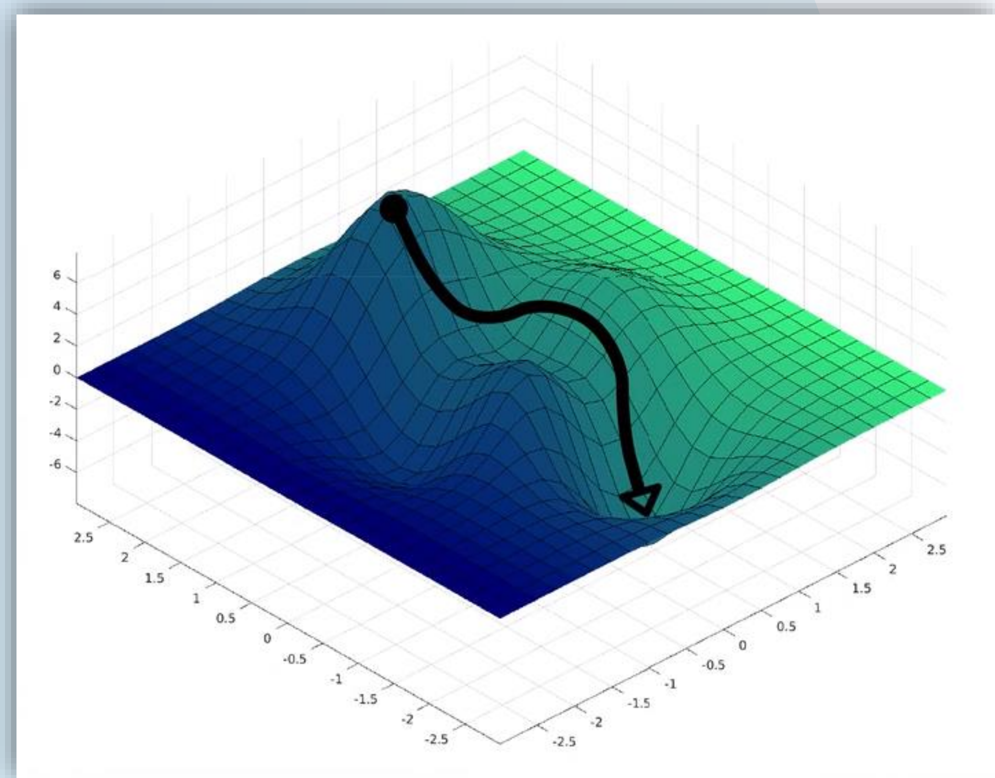
Larger learning rate

Fast & fluctuating learning

Finding the optimal learning rate and optimizing other parameters

What is an optimizer?

An **optimizer** is an algorithm that guides the model on how the weights and biases should be updated. It helps to modify the weights in the best way to **reduce the loss** efficiently.



One of the simplest optimizers: Stochastic gradient descent (SGD)

AI | BUSINESS SCHOOL

What does the SGD do?

- Updates the model parameters to minimize the loss
- Works with probability
- Selects only a batch* for each iteration
- Changes the learning rate during the training

***A batch = a few samples**

Benefits of SGD

👍 **Stochastic gradient descent (SGD) helps to …**

- … reduce the computation time of the optimizer especially for larger datasets

- … reduce the overall loss and improve accuracy