DEEPHEALTH

EDDL

Deep Learning with EDDL

Winter School 24/01/2022

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Contents

Monday 24 and Tuesday 25

What is Deep Learning?	3
Neural Networks	18
EDDL basics	60
Layers and Tensors	68

DEEPHEALTH

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

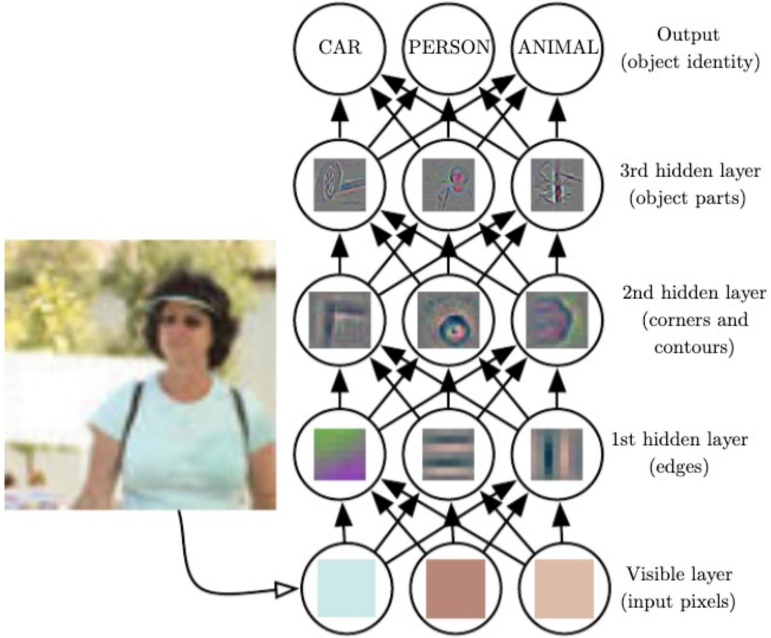
What is Deep Learning?

- Machine learning models based on neural networks:
  - Non-linear models
  - Learning by iteratively showing pairs (input,target) examples
  - Topology defined from know-how, hand-crafted
  - Learn the weights (parameters) by **gradient descent** of a particular **loss function**, iterative procedure
- Deep means that we stack lot of layers (20, 50, 100, 1000)
- Historically neural nets failed to learn with more than (let's say) 10 layers
- Why we need to go deep ? → Representation Learning

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

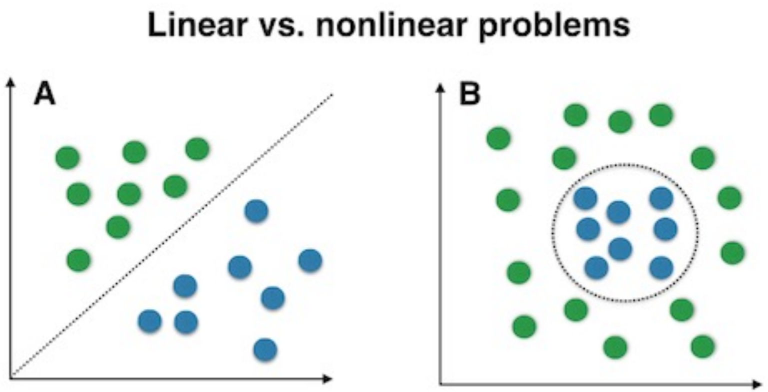
- Going deep



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

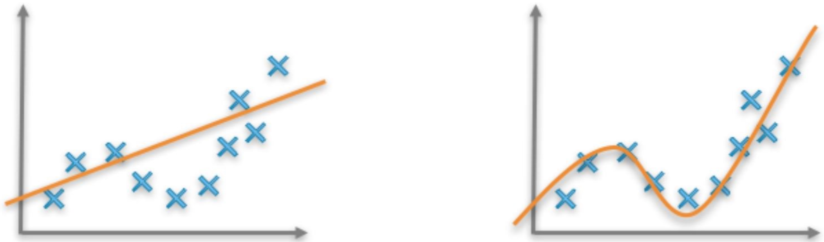
- Non-linear models



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

- Non-linear models



Linear function

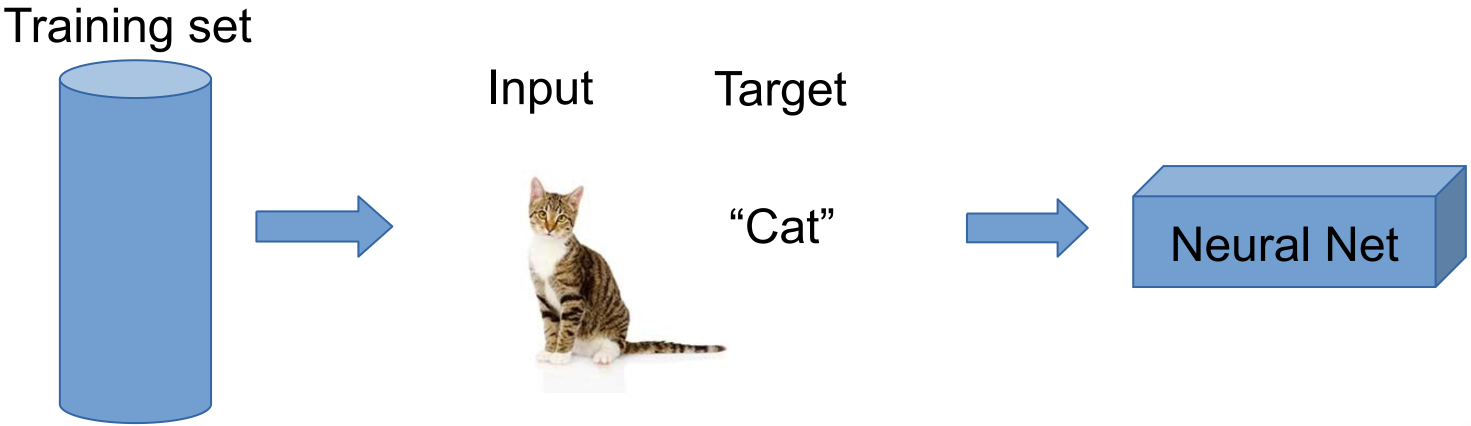
Non-linear function

Best fit linear and non-linear models

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

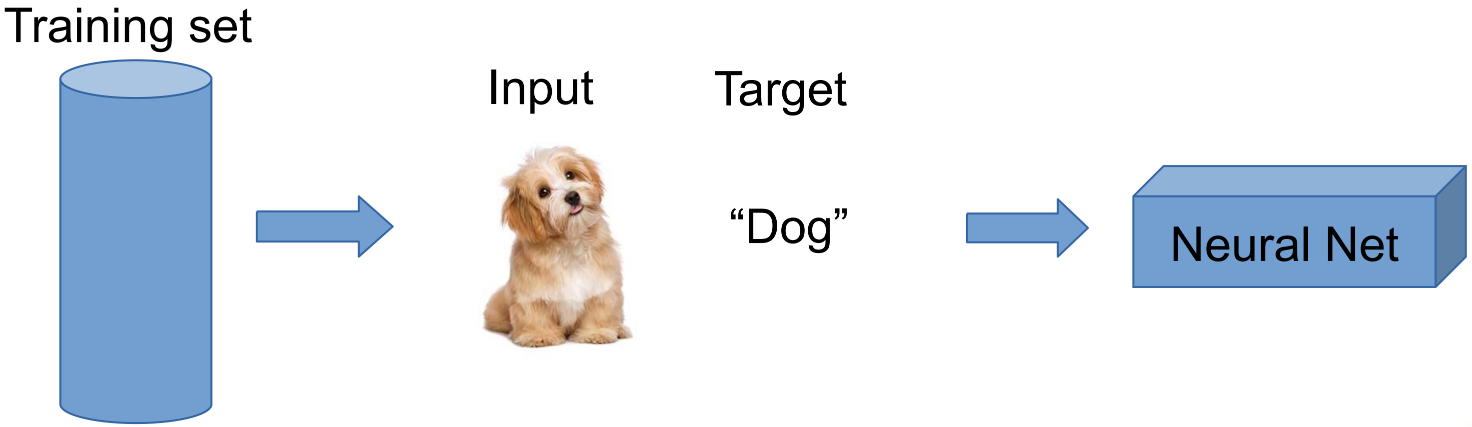
- Learning by iteratively showing pairs (input,target) examples



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

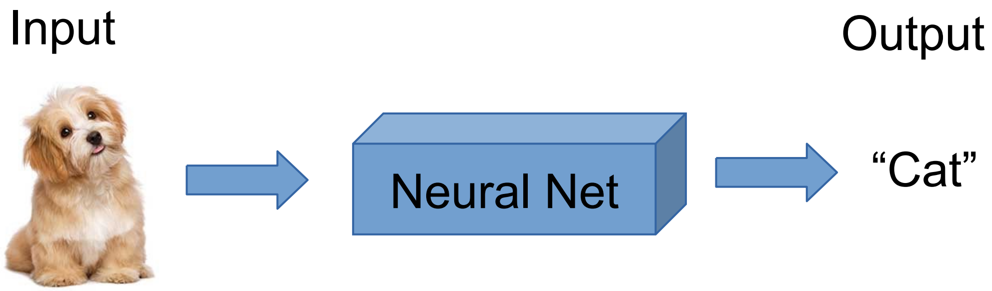
- Learning by iteratively showing pairs (input,target) examples



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

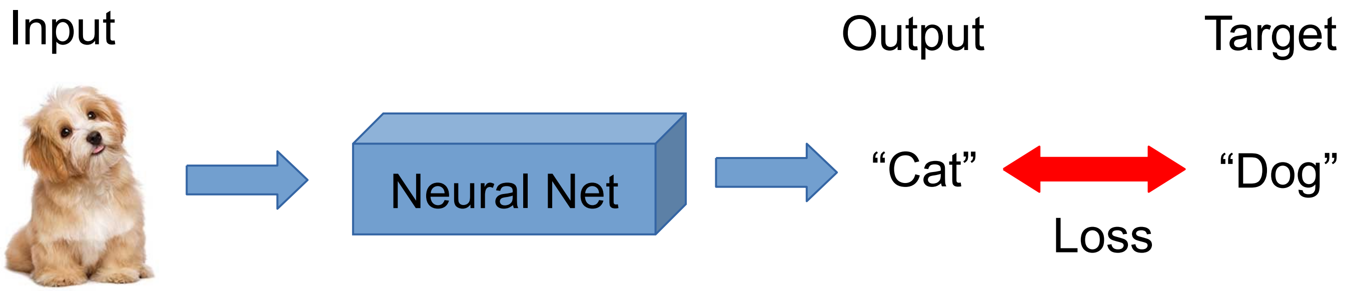
- Learning by iteratively showing pairs (input,target) examples



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

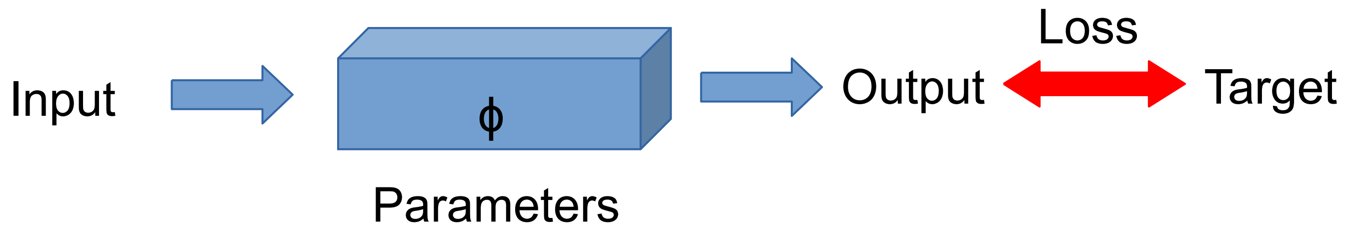
- Learning by iteratively showing pairs (input,target) examples



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

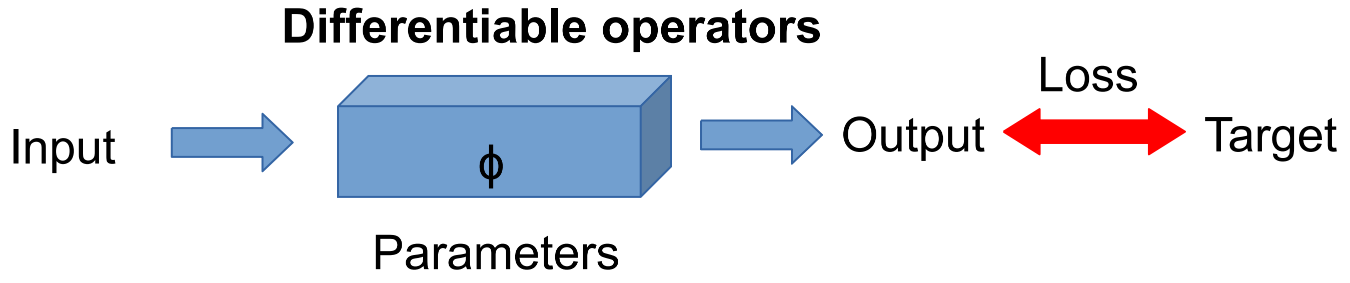
- Learn the weights (parameters) by **gradient descent** of a particular **loss function**, iterative procedure



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

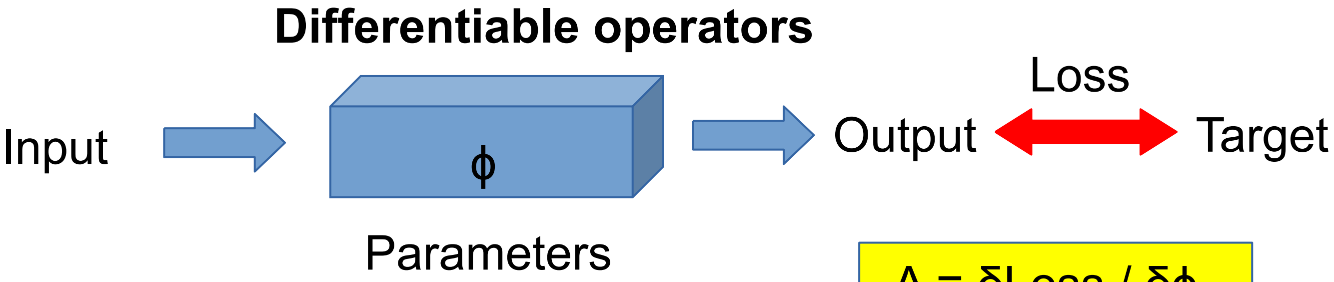
- Learn the weights (parameters) by **gradient descent** of a particular **loss function**, iterative procedure



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

- Learn the weights (parameters) by **gradient descent** of a particular **loss function**, iterative procedure



$\Delta = \delta Loss / \delta\phi$

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

- Learn the weights (parameters) by **gradient descent** of a particular **loss function**, iterative procedure

$$\boxed{\Delta = \delta Loss / \delta\phi}$$

$$\boxed{\phi = \phi - \mu\Delta}$$

- μ is what we call learning rate

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

- Topology defined from know-how

  - Raw Data : Dense Layers
  - Temporal: Recurrent Layers
  - Images: Covolutional Layers

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is Deep Learning?

- Topology defined from know-how

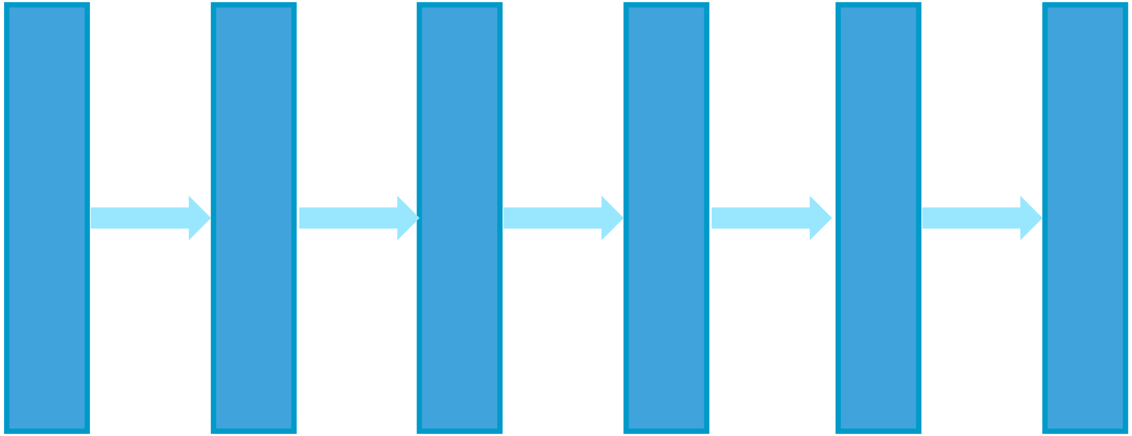
  - Raw Data : Dense Layers
  - Temporal: Recurrent Layers
  - Images: Covolutional Layers
    - Classification: resnet, densenet,
    - Segmentation: U-net
    - Detection: FasterRCNN, Yolo, SSD
    - Pixel annotation: MaskRCNN
    - Image Generation: DCGAN, CylceGan
    - Fine grain classification: Bilinear CNN

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

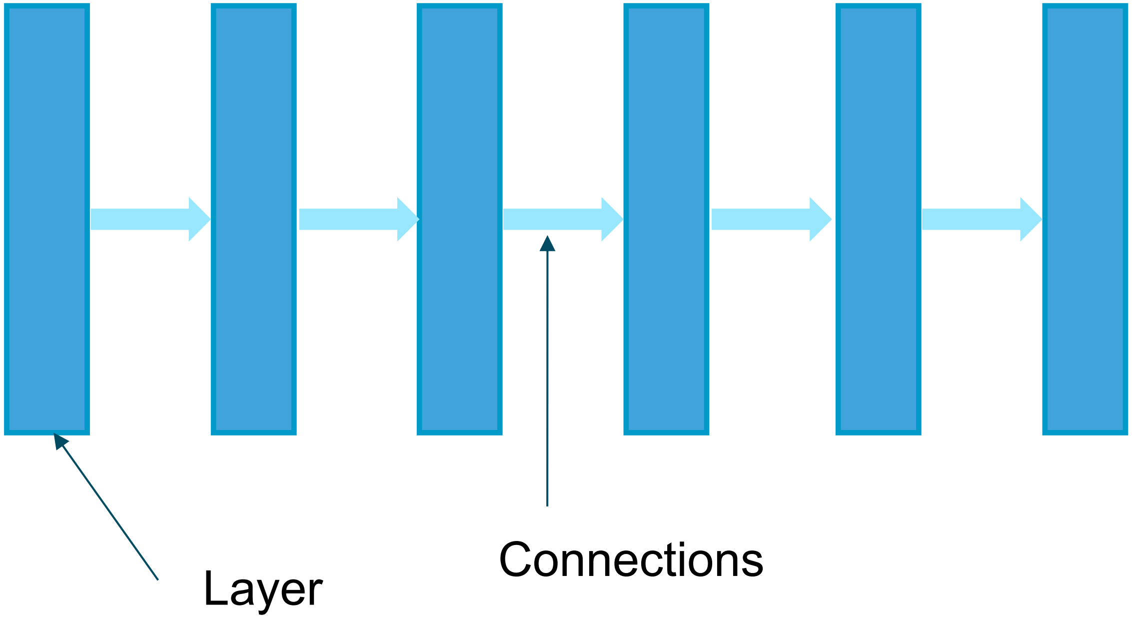
What is a Neural Network?



A connectionist model

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

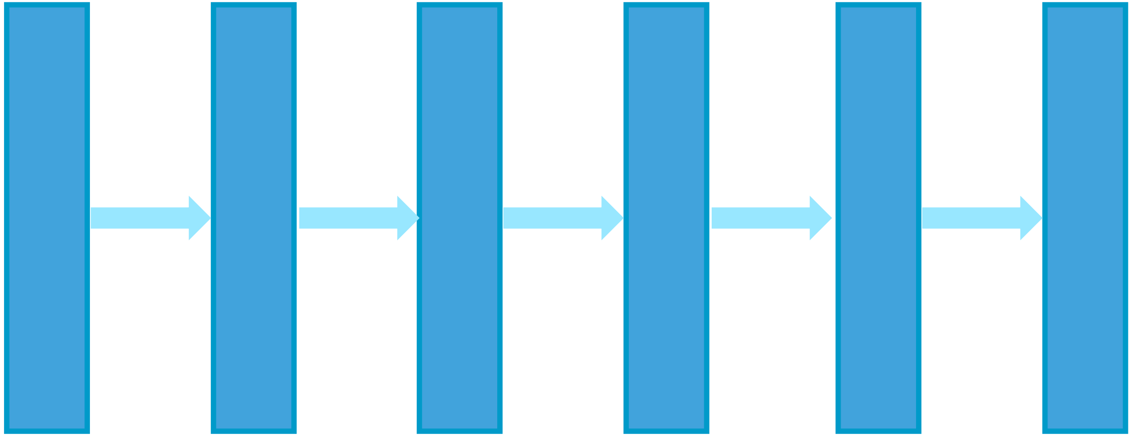
Layer: Operation over tensors (1D, 2D, 3D …)

Connection: the output tensor of a layer (result) is used as input tensor of the following layer

We would refer “*x*” as input and “*y*” as output

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



Input Layer

Dense Layer

ReLu Layer

…

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

Input Layer: Receive the input examples (images of cats)

Dense Layer: Is a **parametric** layer with the following operation:

$$y = Wx + b$$

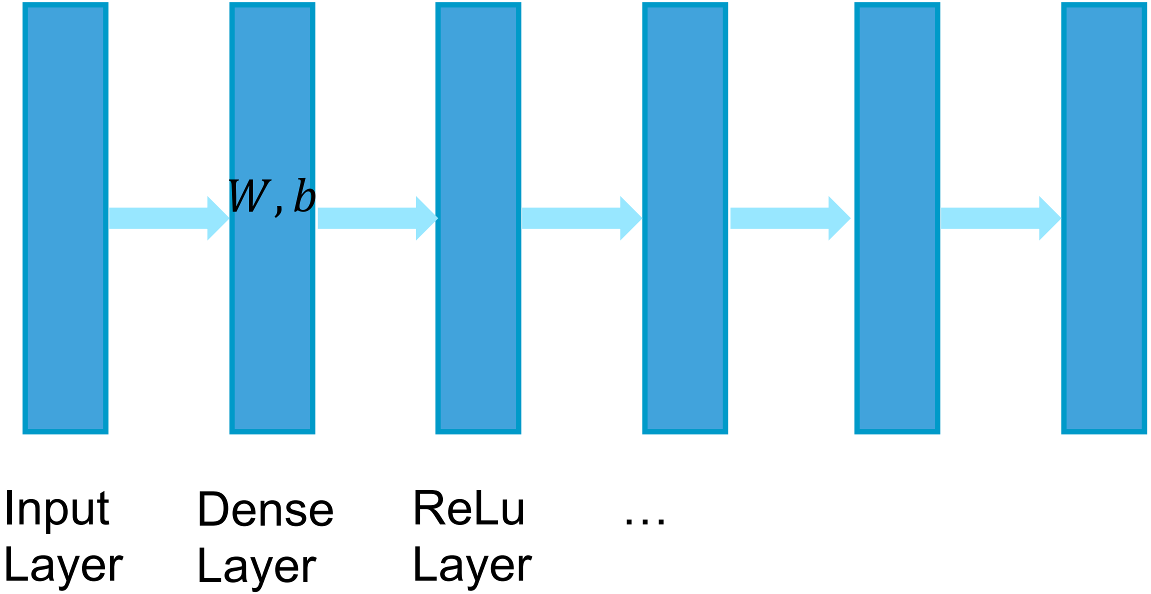
W and *b* are the parameters to learn

Activation Layer: Normally is a non-parametric layer. In the case of ReLu:

$$y = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

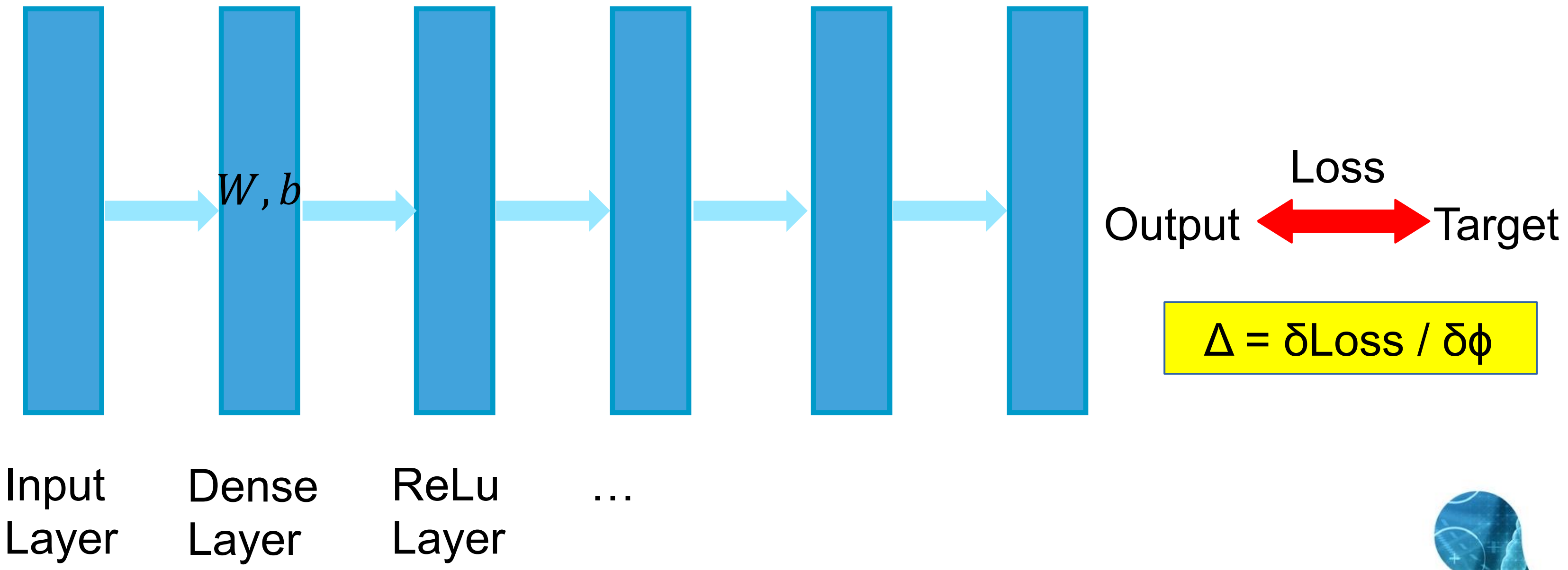
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

Classification problems. n clases output with n neurons

Output: Softmax

$$y_i = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \qquad y_i = p(c = i|\boldsymbol{x})$$

Loss: Categorical cross-entropy

$$\text{Loss} = -\sum_{i=1}^{n} y_i \cdot \log \hat{y}_i$$

target

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

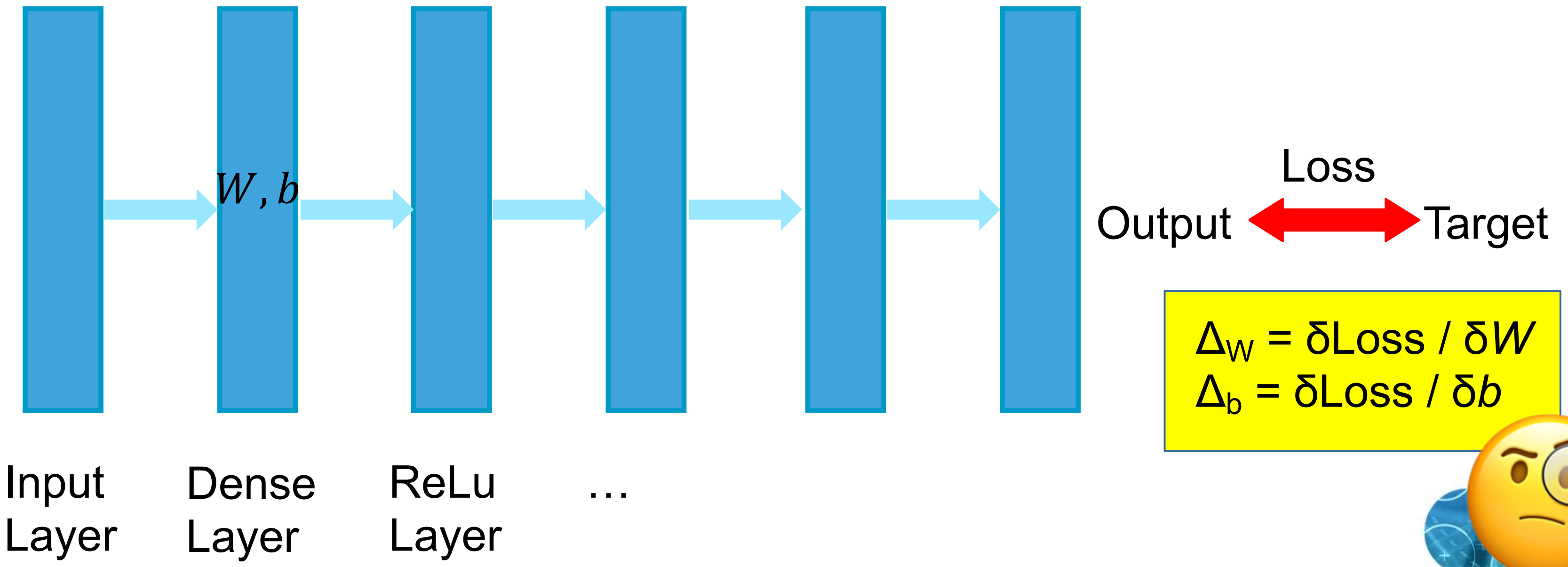
Regresion problems. d dimensions output with d neurons

Output: Linear $$y_i = x_i$$

Loss: Sum of Quadratic Errors $$Loss = \sum_{i=1}^{d}(y_i - \hat{y}_i)^2$$

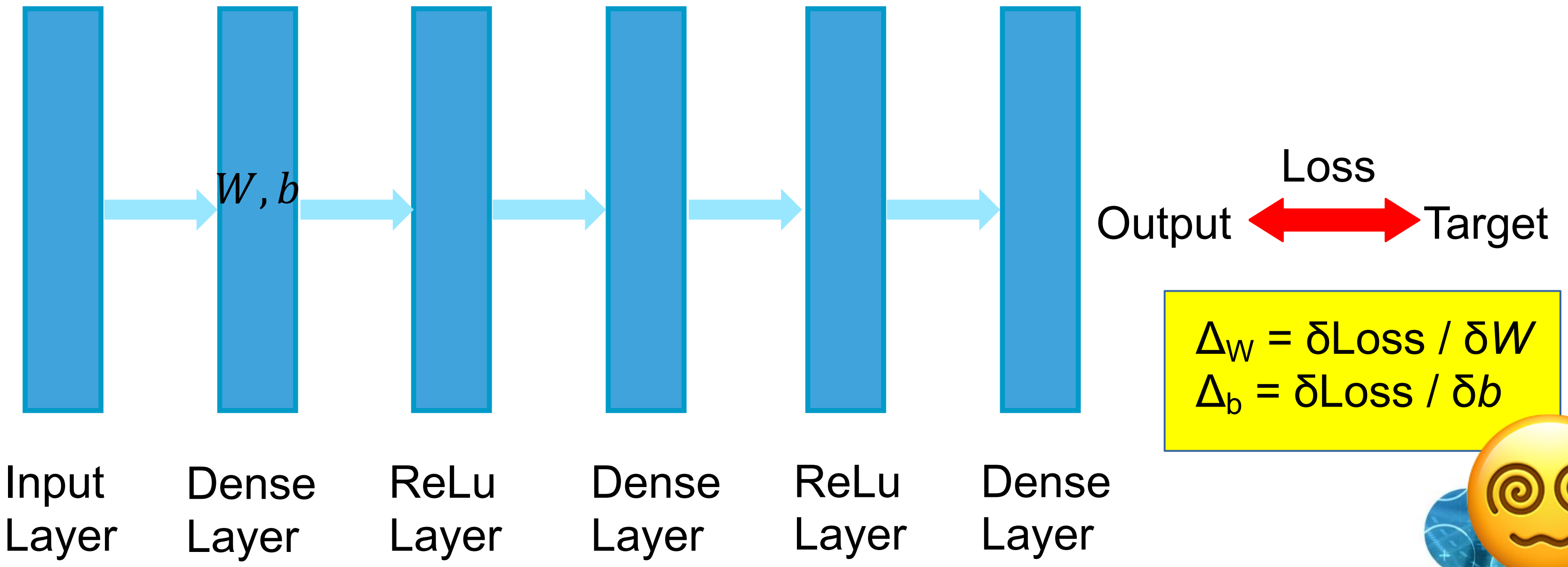
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



Δ_W = δLoss / δW
Δ_b = δLoss / δb

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

The Chain Rule

For $F(x) = f(\,g(x)\,)$

$$F'(x) = f'(\,g(x)\,)\,.\,g'(x)$$

Derivative of outer function

Derivative of inner function

www.mathsathome.com

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

In order to compute the gradient $\Delta = \delta Loss / \delta\phi$ w.r.t any parameter we can use **the chain rule** to back-propagate the loss

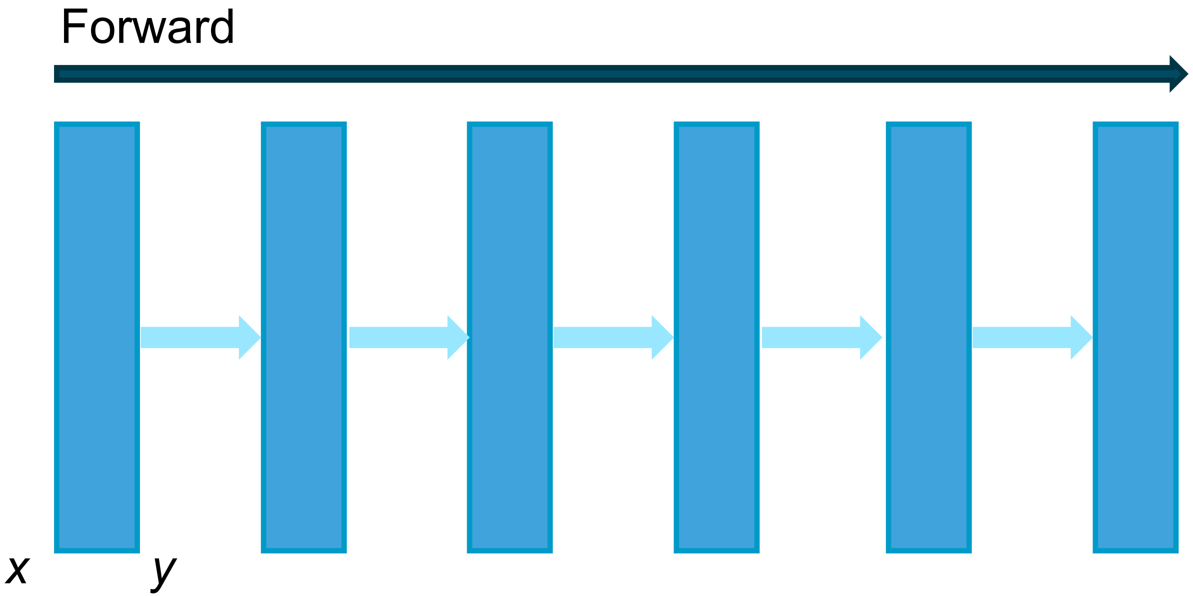
This leads us to the well-know Backpropagation algorithm

Essentially the training phase of a NN is composed by three steps:

- Forward : the output of all the layers are computed
- Backward: the gradient w.r.t all the parameters of all the layers is computed (backpropagation)
- Update: the parameters are updated in the opposite direction of the gradient (optimizer)

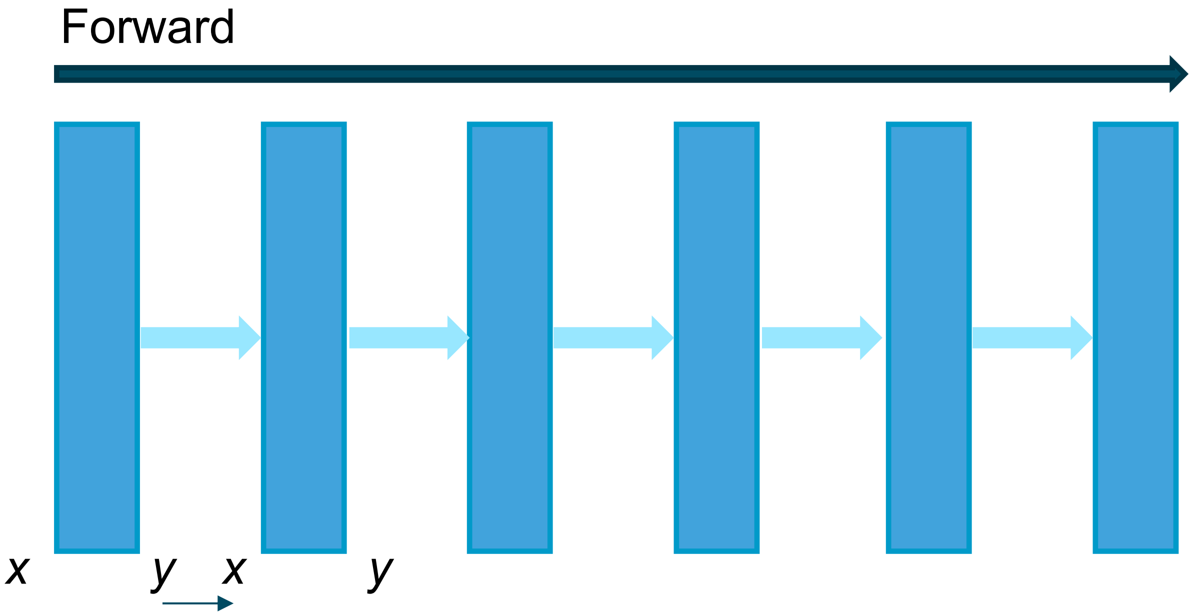
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



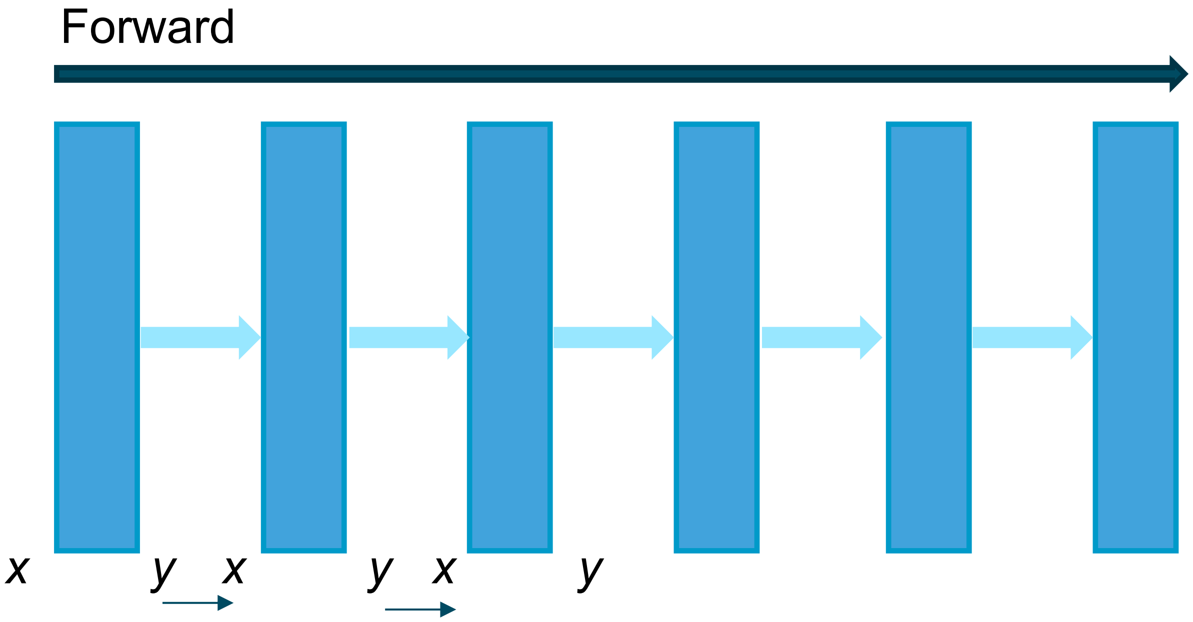
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



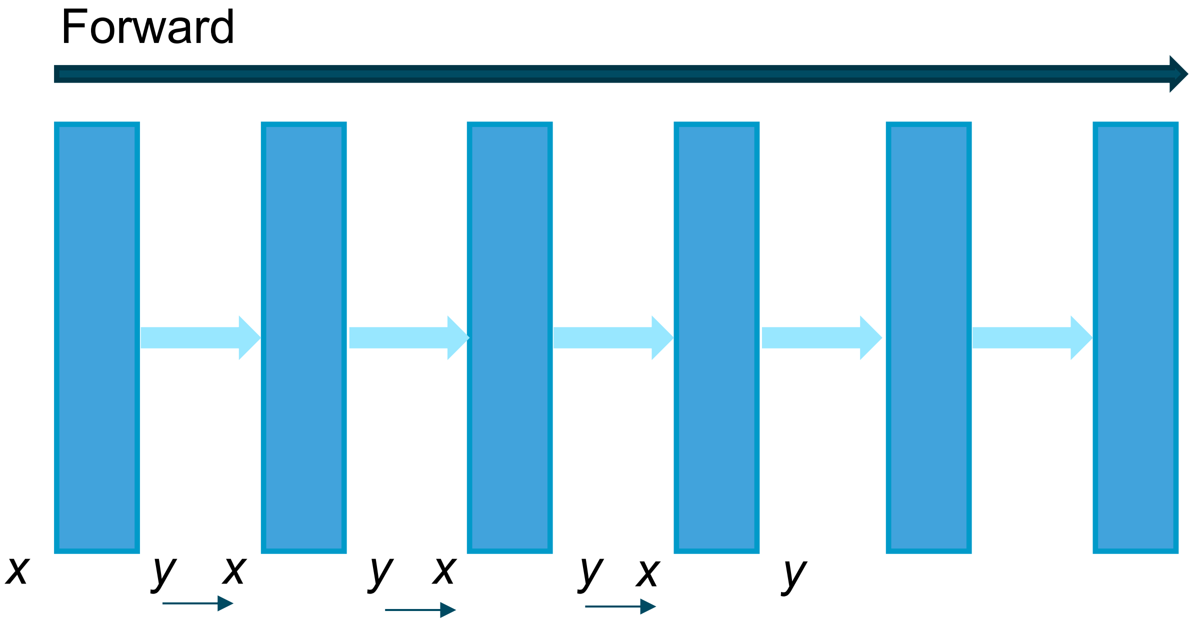
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



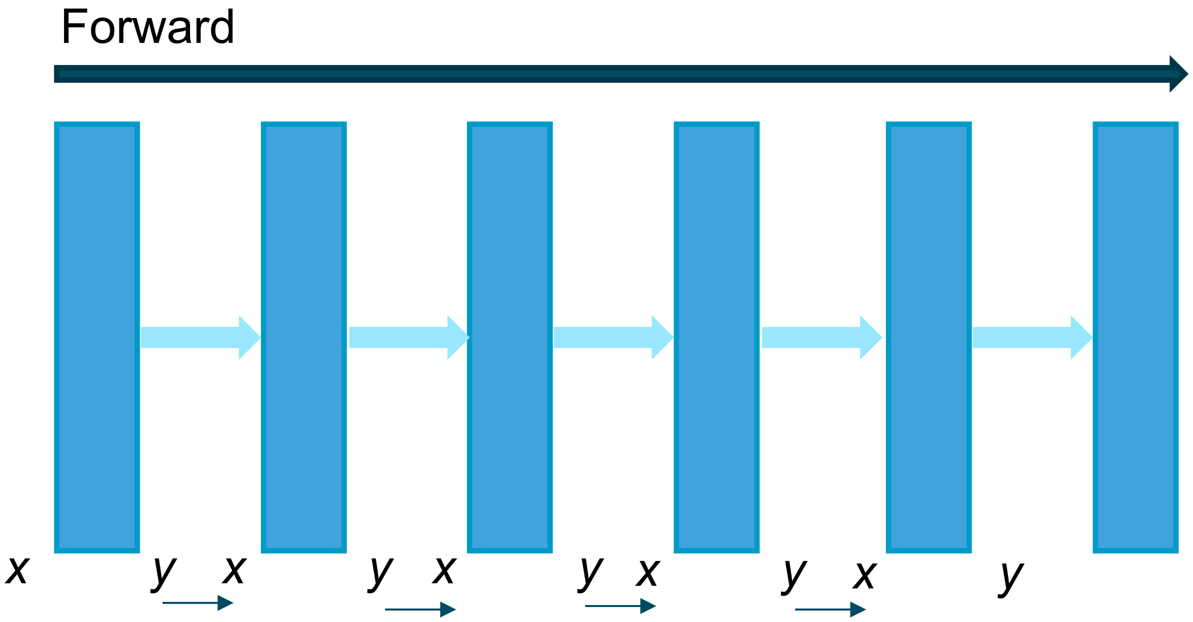
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



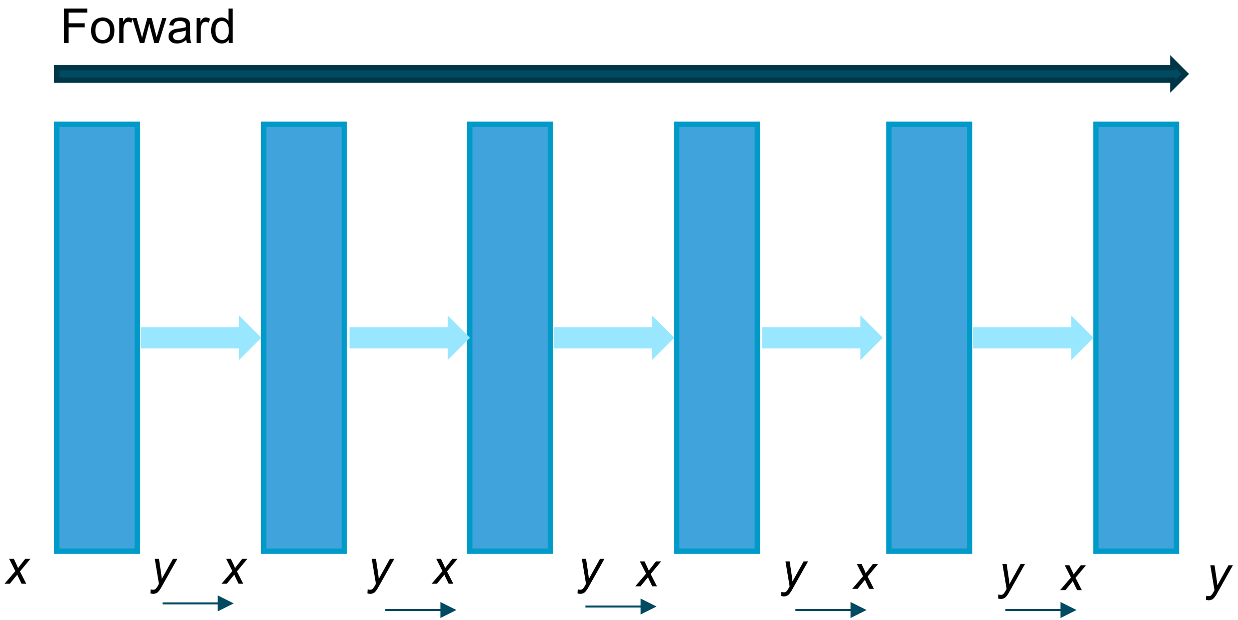
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



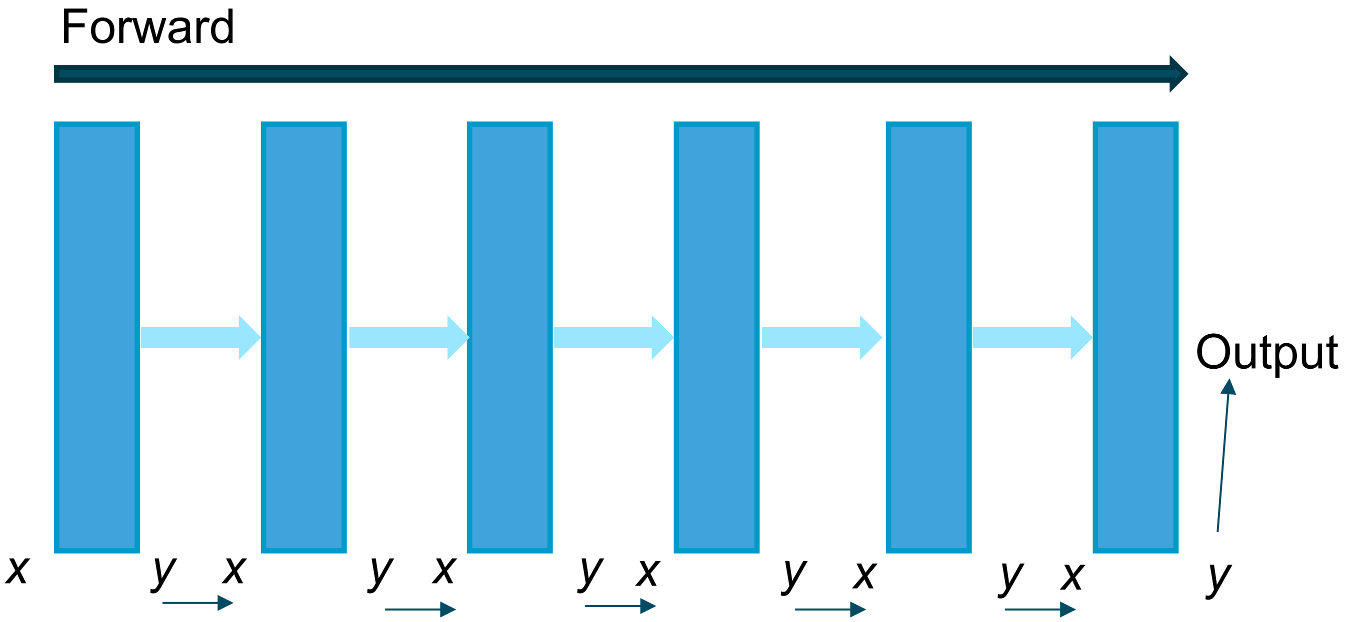
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



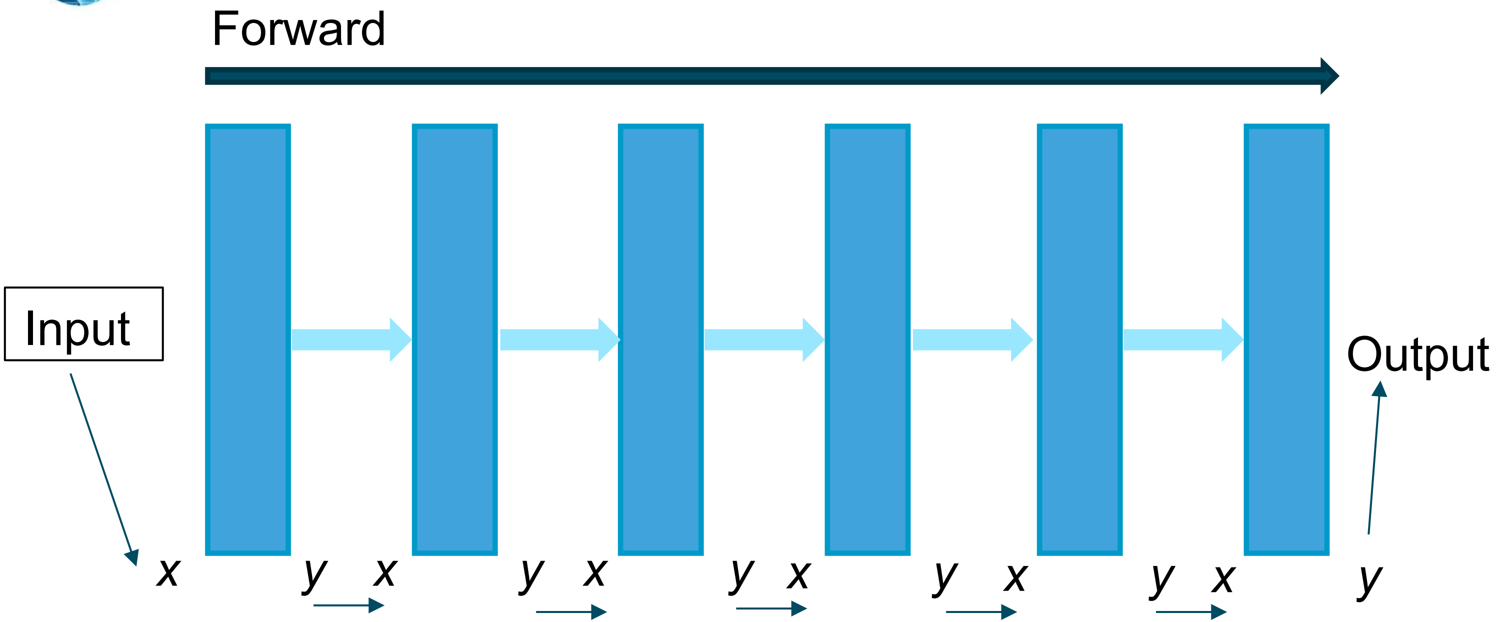
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



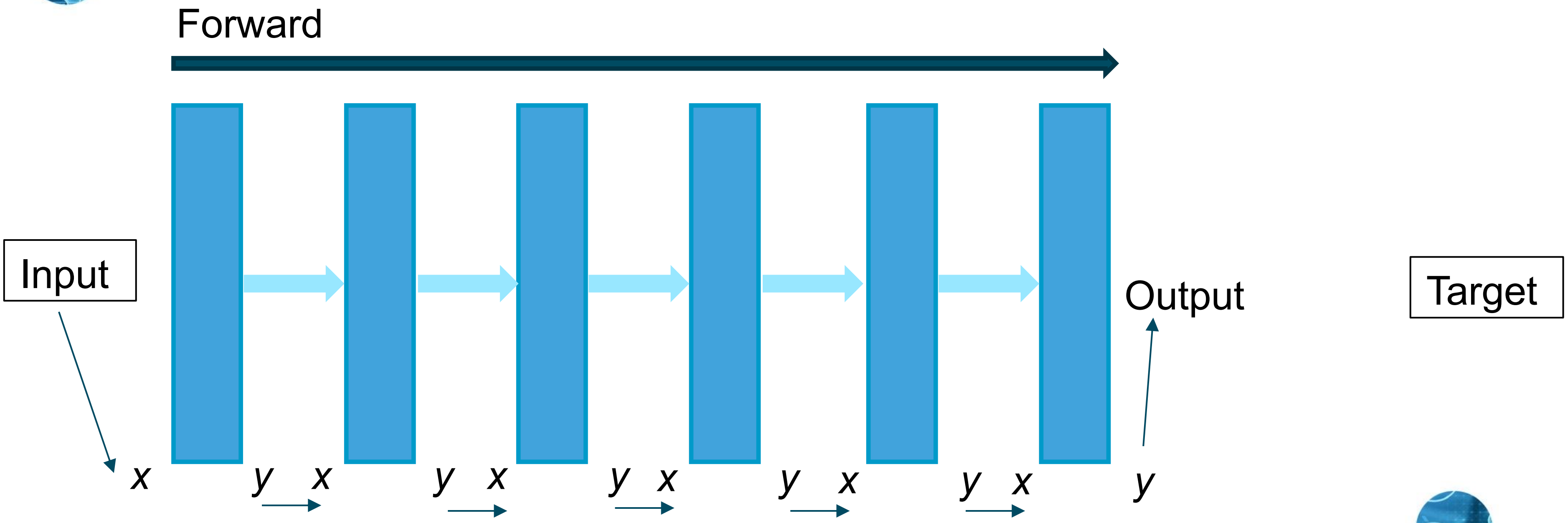
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



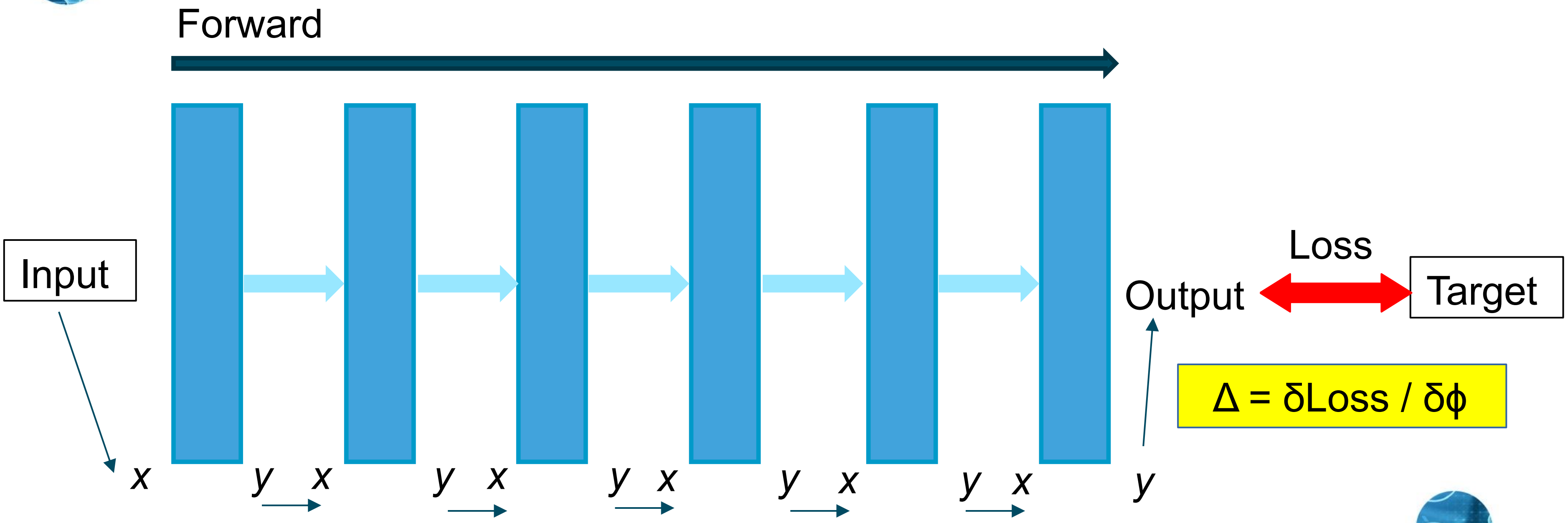
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



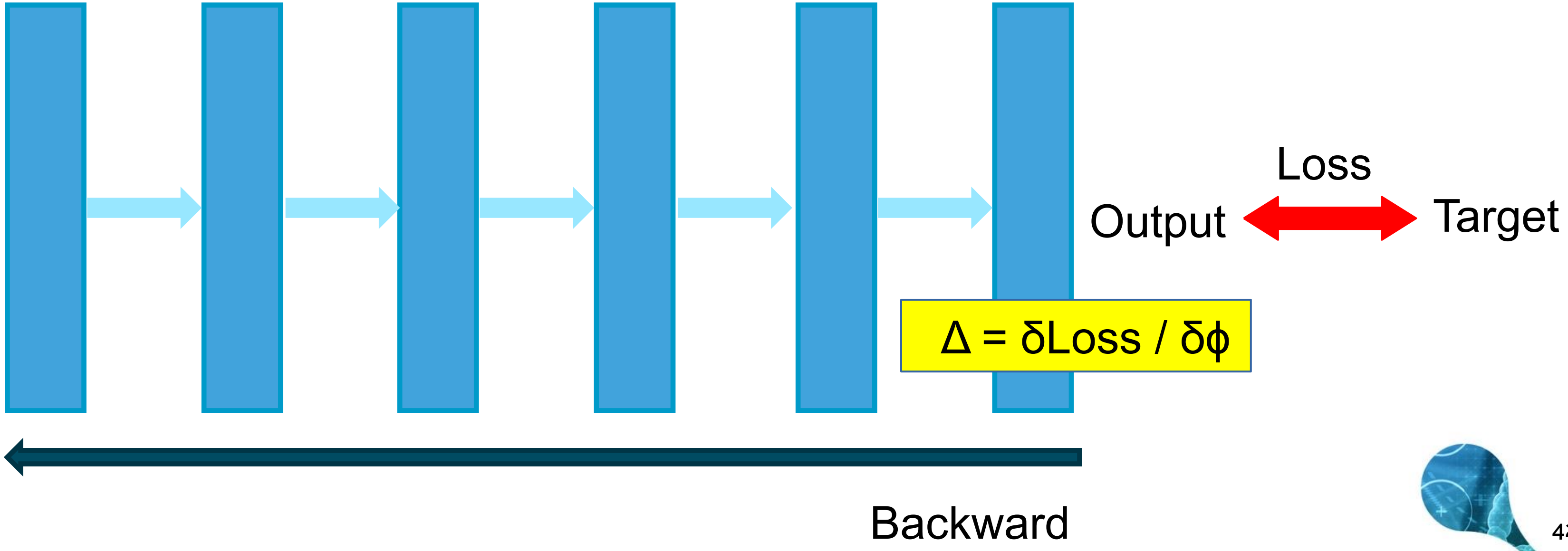
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



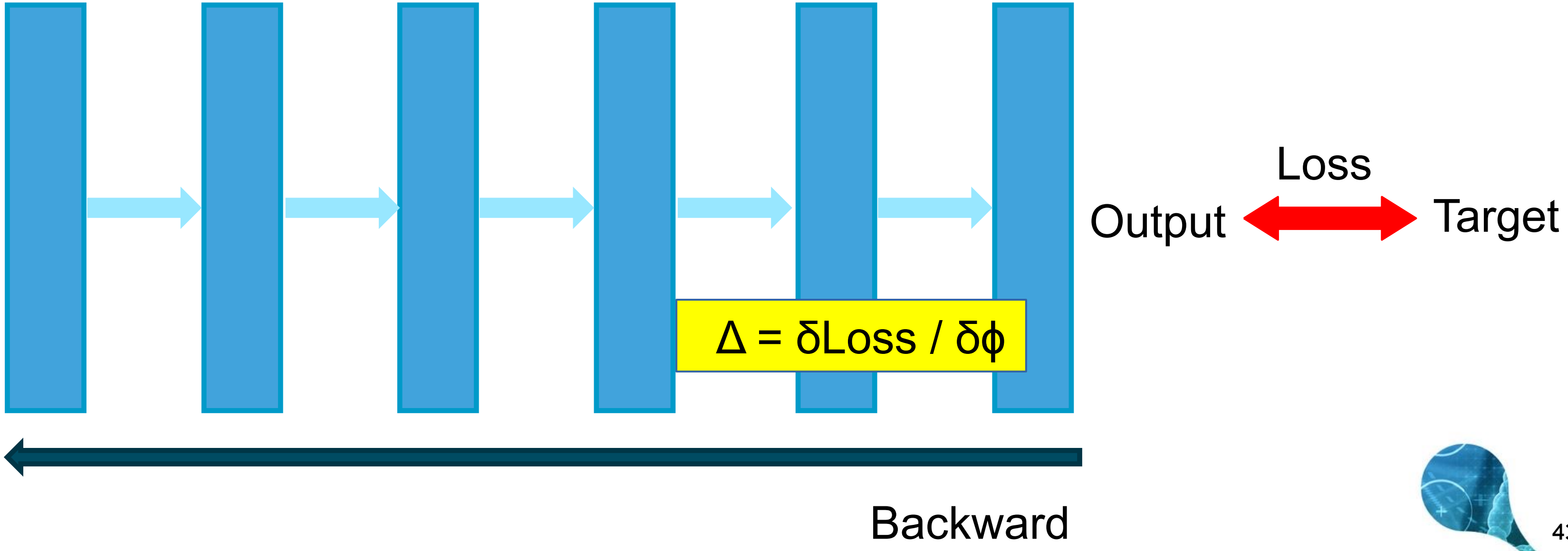
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



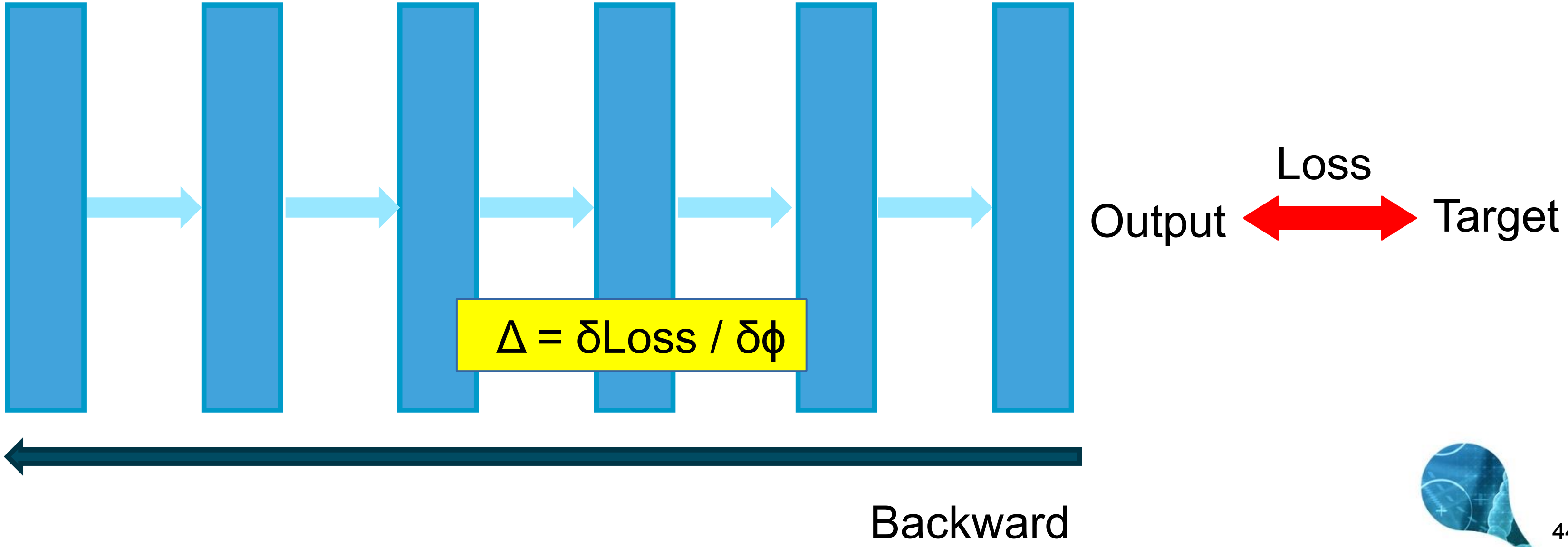
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



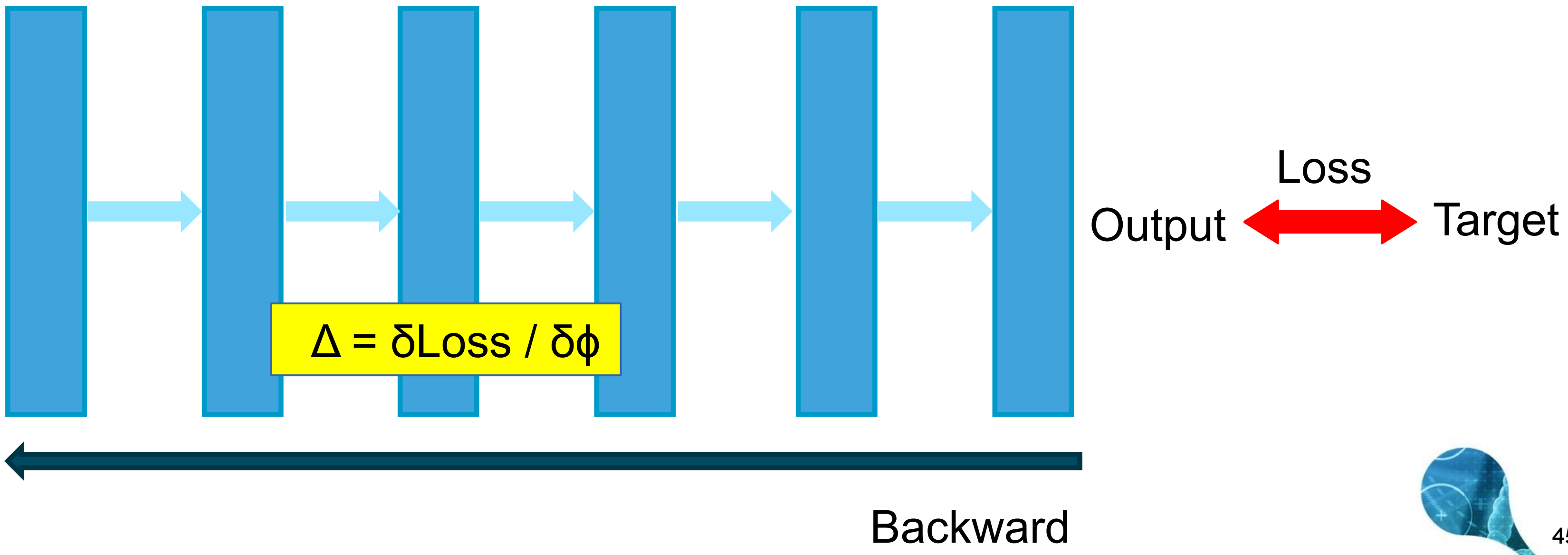
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



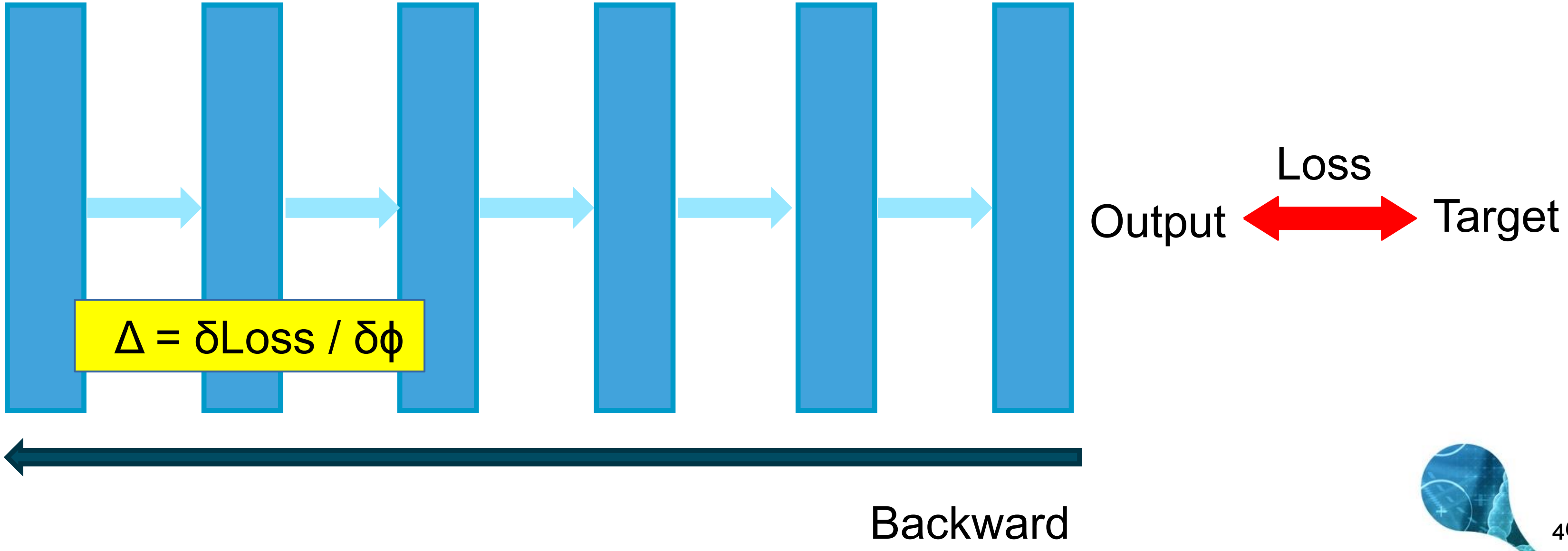
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



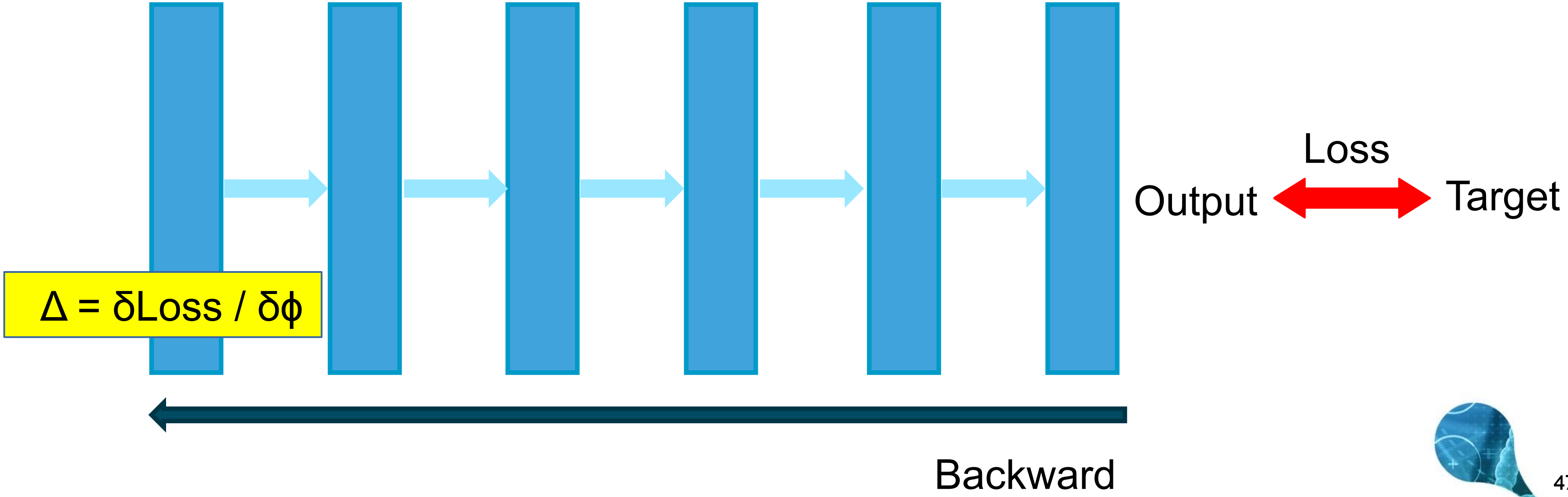
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



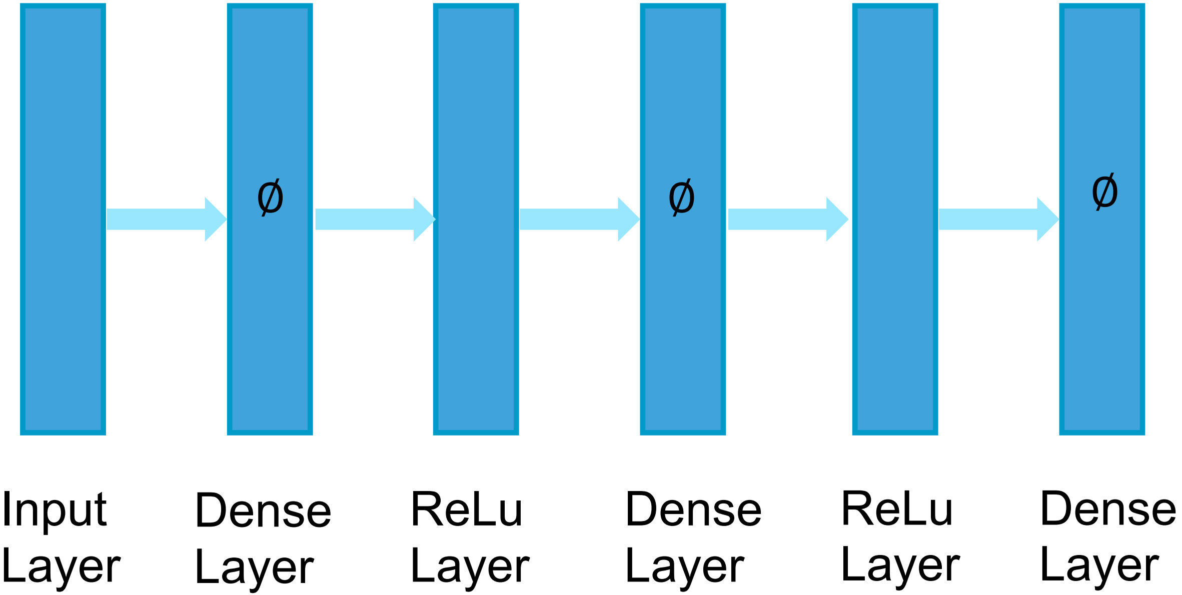
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

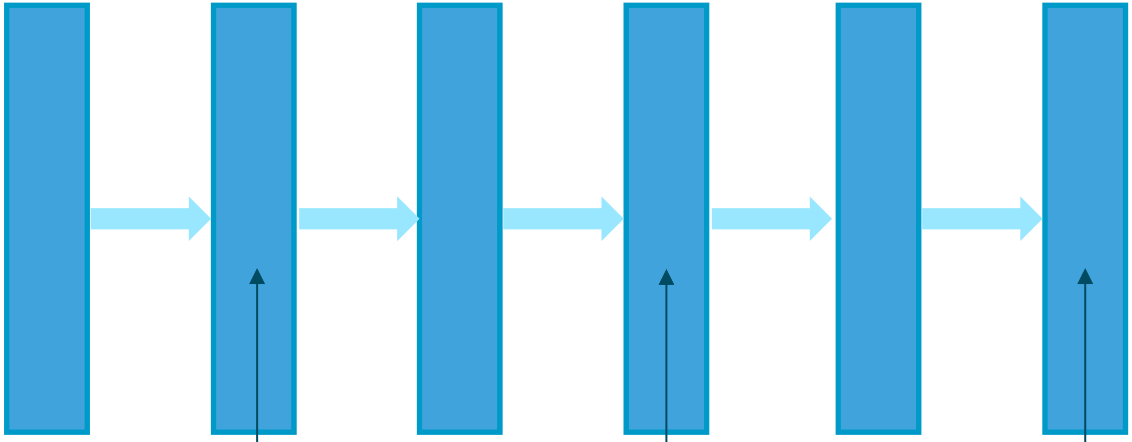
What is a Neural Network?



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

Update



$$W_1 = W_1 - \mu\Delta_{W1}$$
$$b_1 = b_1 - \mu\Delta_{b1}$$

$$W_2 = W_2 - \mu\Delta_{W2}$$
$$b_2 = b_2 - \mu\Delta_{b2}$$

$$W_3 = W_3 - \mu\Delta_{W3}$$
$$b_3 = b_3 - \mu\Delta_{b3}$$

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

Training:

- Forward
- Backward
- Update

Inference:

- Forward

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

Training:

- Forward
- Backward
- Update

EDDL: `model.fit( input, target)`

Inference:

- Forward

EDDL: `output = model.predict( input )`

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

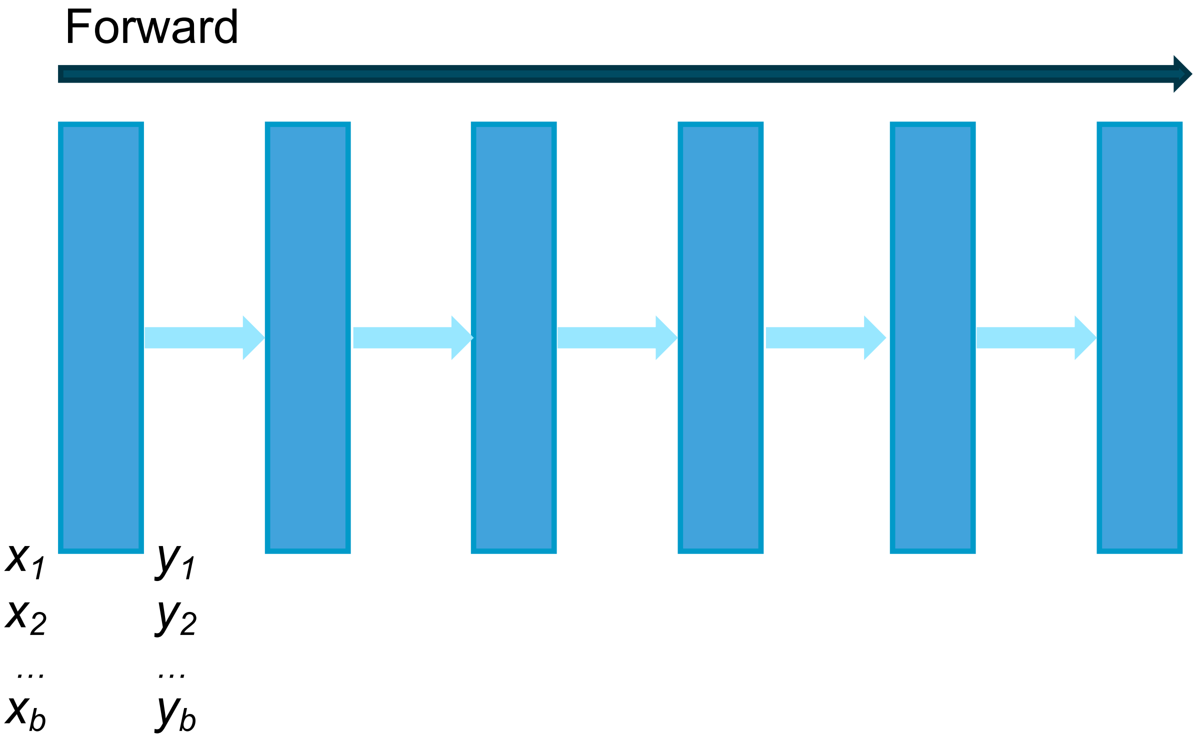
Training:

- Forward
- Backward
- Update

The samples (input,target) are presented in batch, then the gradient is computed for all the samples in a **batch**

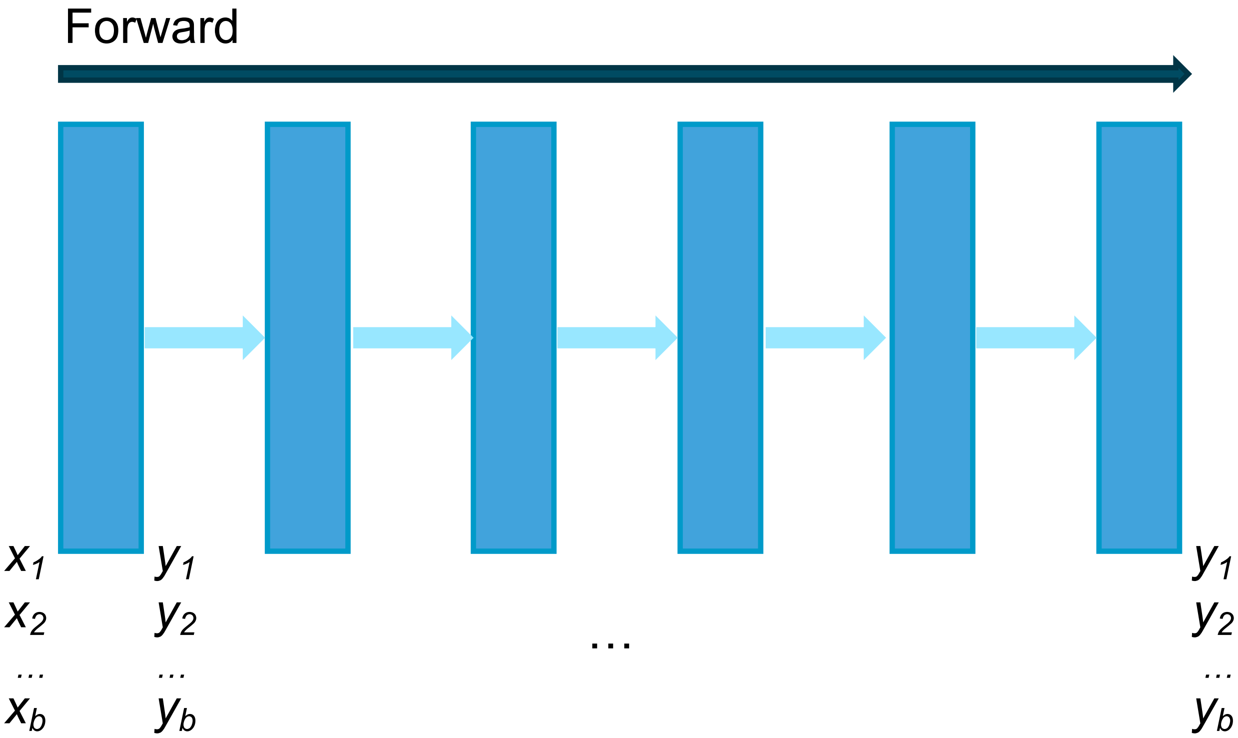
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



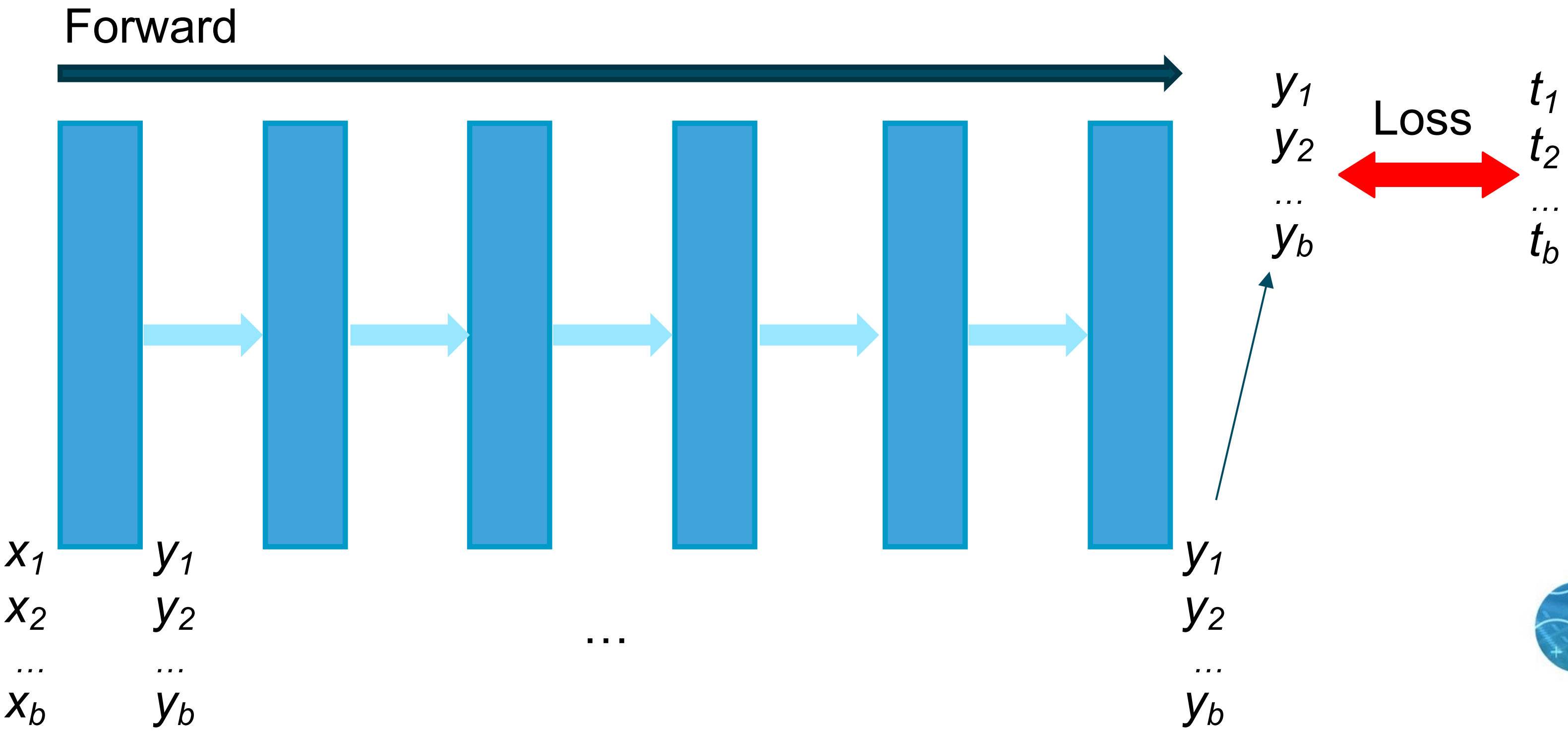
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



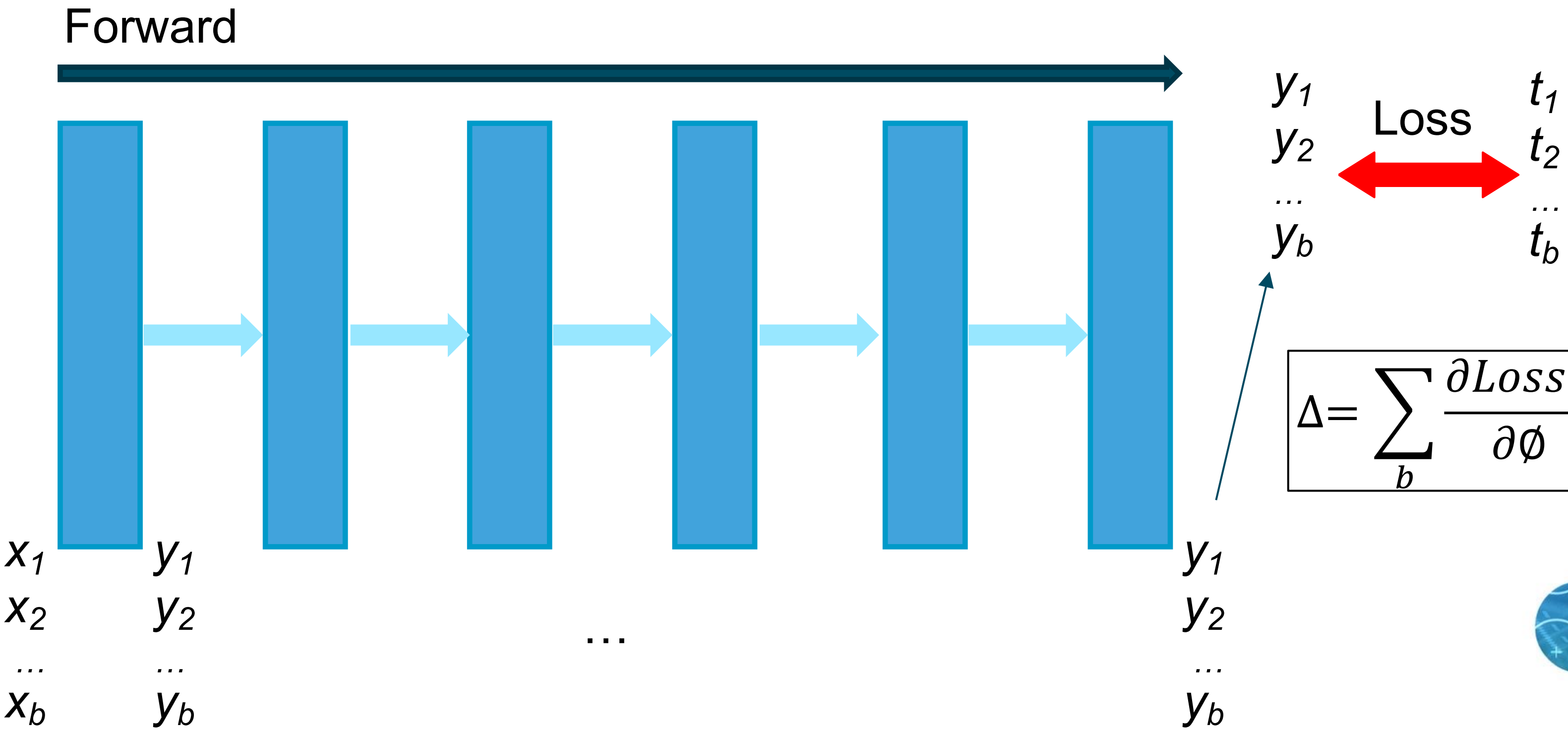
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



$$\boxed{\Delta = \sum_b \frac{\partial Lossb}{\partial \emptyset}}$$

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

Training:

- Forward
- Backward
- Update

EDDL: model.fit(input, target, batch_size)

Inference:

- Forward

EDDL: output = model.predict(input)

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

Training

Definition of **epoch**: train all the samples, then is to train the number of batches that fit in your training set.

Training set of 50.000 samples
Batch_size= 100

Epoch means to train 500 batches (randomly selected) covering all the samples of the training set

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?

Training:

- Forward
- Backward
- Update

EDDL: model.fit(input, target, batch_size, epochs)

Inference:

- Forward

EDDL: output = model.predict(input)

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL Basic

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL Basic

EDDL is an open source library for Distributed Deep Learning and Tensor Operations in C++ for **CPU**, **GPU** and **FPGA**. EDDL is developed inside the DeepHealth project.

EDDL provides a high level API to develop Deep learning projects in C++

PyEDDL is a Python wrapper for the EDDL

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL Basic

EDDL is like Keras in the way the models are defined and the training is carried out. The main difference between EDDL and other toolkits is the introduction of the Computing Service (CS) object.

The DL model created with EDDL has a CS attached where this model will be deployed. This CS could be CPU, GPU or FPGA and even we could define a distributed CS.

This CS definition only affects to a single line in your programs.

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL Basic

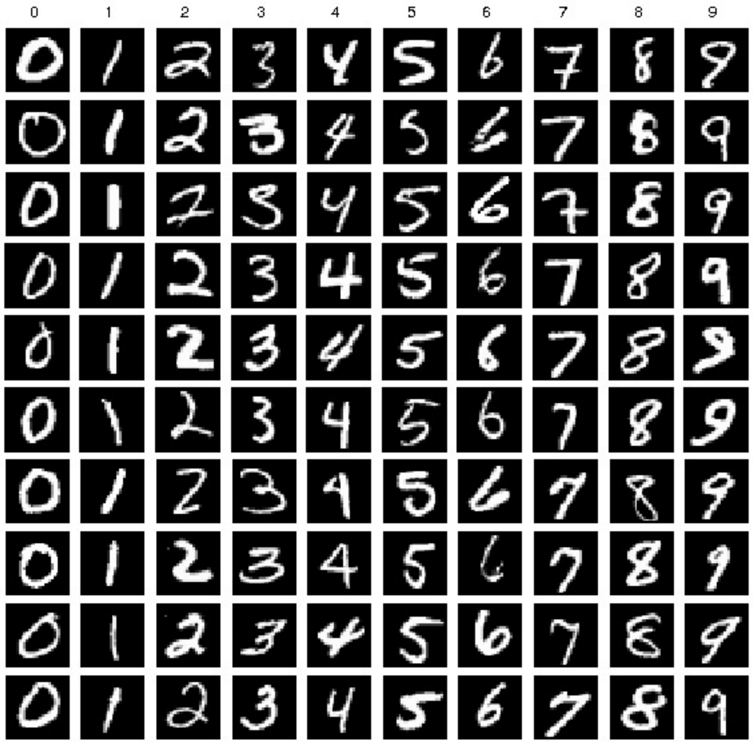
```
download_mnist();

// Define network
layer in = Input({784});
layer l = in;

l = LeakyReLu(Dense(l, 1024));
l = LeakyReLu(Dense(l, 1024));
l = LeakyReLu(Dense(l, 1024));

layer out = Softmax(Dense(l, 10), -1);
model net = Model({in}, {out});

// Computing service
compserv cs = CS_CPU();
// cs = CS_GPU({1},"low_mem"); // one GPU
// cs = CS_GPU({1,1},100); // two GPU
// cs = CS_FPGA({1});
```

```
// Build model
build(net,
      adam(0.001), // Optimizer
      {"softmax_cross_entropy"}, // Losses
      {"categorical_accuracy"}, // Metrics
      cs ); // Computing service

// Train model
fit(net, {x_train}, {y_train}, 32, 10);

// Evaluate
evaluate(net, {x_test}, {y_test});
```


The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL Basic

MNIST Digits Classification



28x28 images (2D)

stored as 784 pixels (1D)

10 classes

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL Basic

Data

```
# Download dataset
eddl.download_mnist()
# Load the dataset
x_train = Tensor.load('mnist_trX.bin')
y_train = Tensor.load('mnist_trY.bin')
x_test = Tensor.load('mnist_tsX.bin')
y_test = Tensor.load('mnist_tsY.bin')

# Preprocess the images. From [0-255] to [0-1]
x_train.div_(255.0)
x_test.div_(255.0)

# Show data shape
print('Dataset shape:')
print(f'Train split images: {x_train.shape}')
print(f'Train split labels: {y_train.shape}')
print(f'Test split images: {x_test.shape}')
print(f'Test split labels: {y_test.shape}')
```


The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL Basic

Model

```
# Define the model topology
num_classes = 10
in_ = Input([784])
layer = Reshape(in_, [1, 28, 28])  # EDDL needs channel first images
layer = ReLu(BatchNormalization(Conv(layer, 32, [3, 3]), affine=True))
layer = MaxPool(layer, [2, 2])
layer = ReLu(BatchNormalization(Conv(layer, 64, [3, 3]), affine=True))
layer = MaxPool(layer, [2, 2])
layer = ReLu(BatchNormalization(Conv(layer, 128, [3, 3]), affine=True))
layer = MaxPool(layer, [2, 2])
layer = ReLu(BatchNormalization(Conv(layer, 256, [3, 3]), affine=True))
layer = GlobalAveragePool(layer)
layer = Flatten(layer)
out_ = Softmax(Dense(layer, num_classes))

# Create the model
model = eddl.Model([in_], [out_])
```


The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL Basic Learning

```
# Build the model to prepare it for training or inference
eddl.build(model,
           opt,                                # Optimizer
           ['categorical_cross_entropy'],      # Losses
           ['accuracy'],                       # Metrics
           cs)                                 # Computing Service


# Show the model layers
eddl.summary(model)


eddl.fit(model, [x_train], [y_train], batch_size, epochs)  # Train

eddl.evaluate(model, [x_test], [y_test], args.batch_size)  # Validation
```


The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers and Tensors

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

Dense

- Input a 1D Tensor (d)
- Output a 1D Tensor (d')

- Operation: $\boldsymbol{y} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}$

- Parameters: W,b
  - W is a (d',d) 2D Tensor
  - B is a (d') 1D Tensor

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

Activation

- Input any D Tensor
- Output the same dim as input Tensor

- Operation: depends on activation function

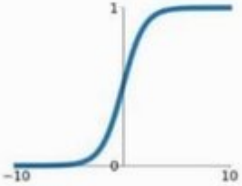
- Parameters: NO

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

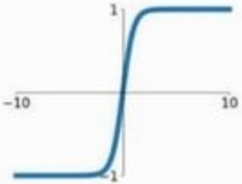
Sigmoid

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



tanh

$\tanh(x)$

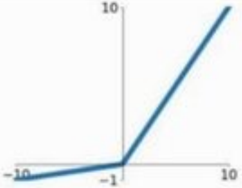


ReLU

$\max(0, x)$

Leaky ReLU

$\max(0.1x, x)$

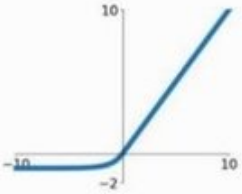


Maxout

$\max(w_1^T x + b_1, w_2^T x + b_2)$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

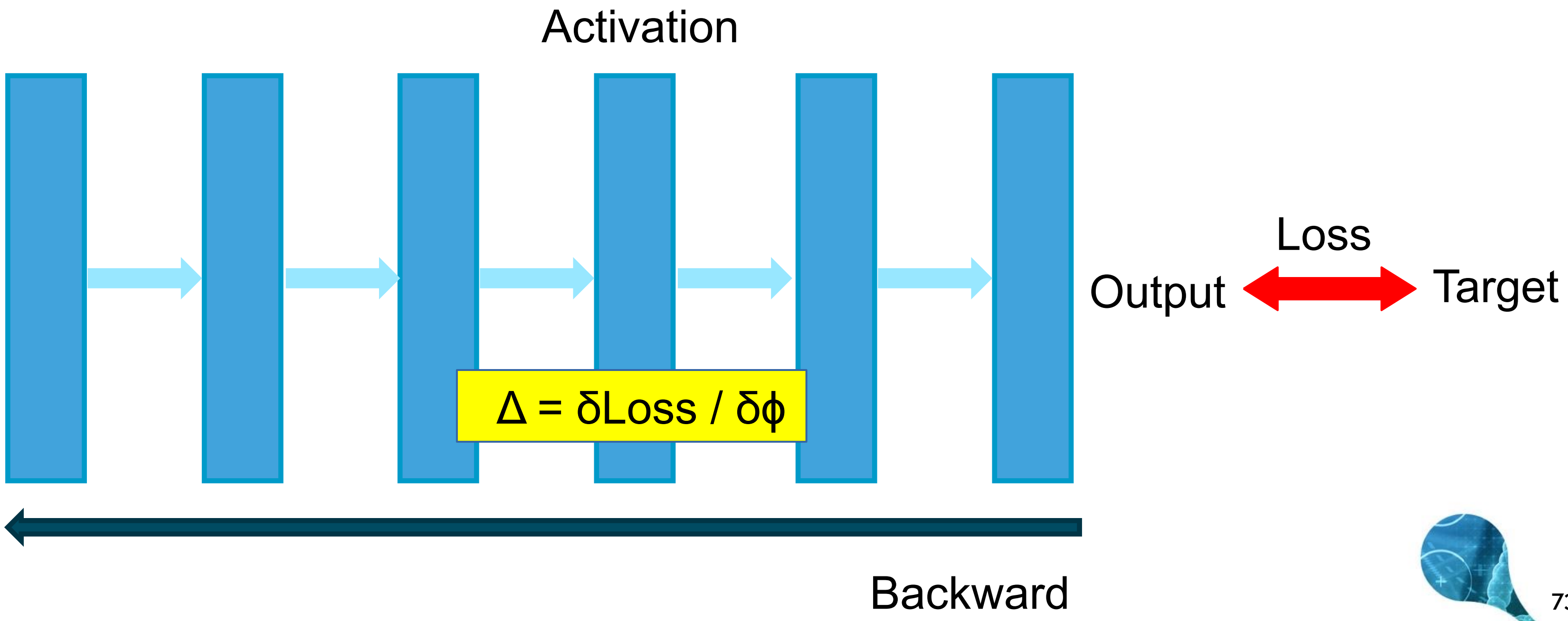
Layers

Activation

- The backpropagated gradient is multiply by the derivative of the activation function

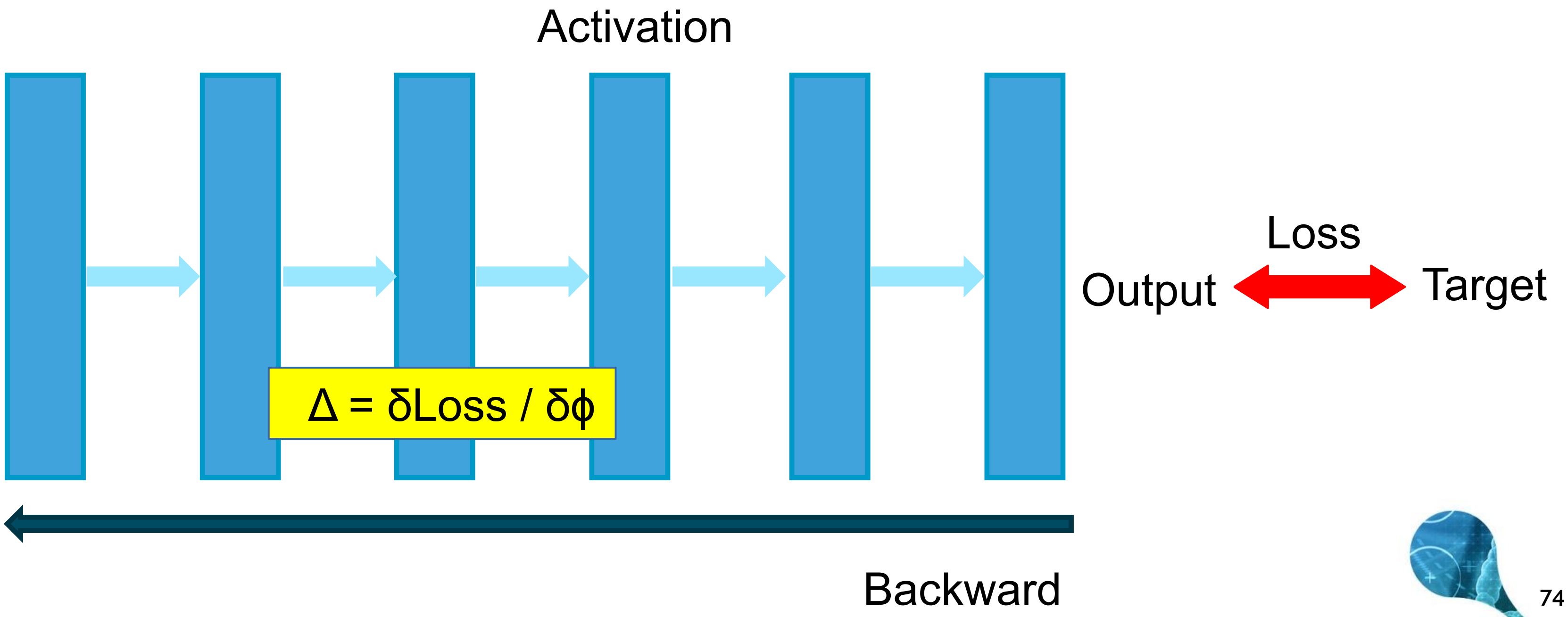
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

What is a Neural Network?



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

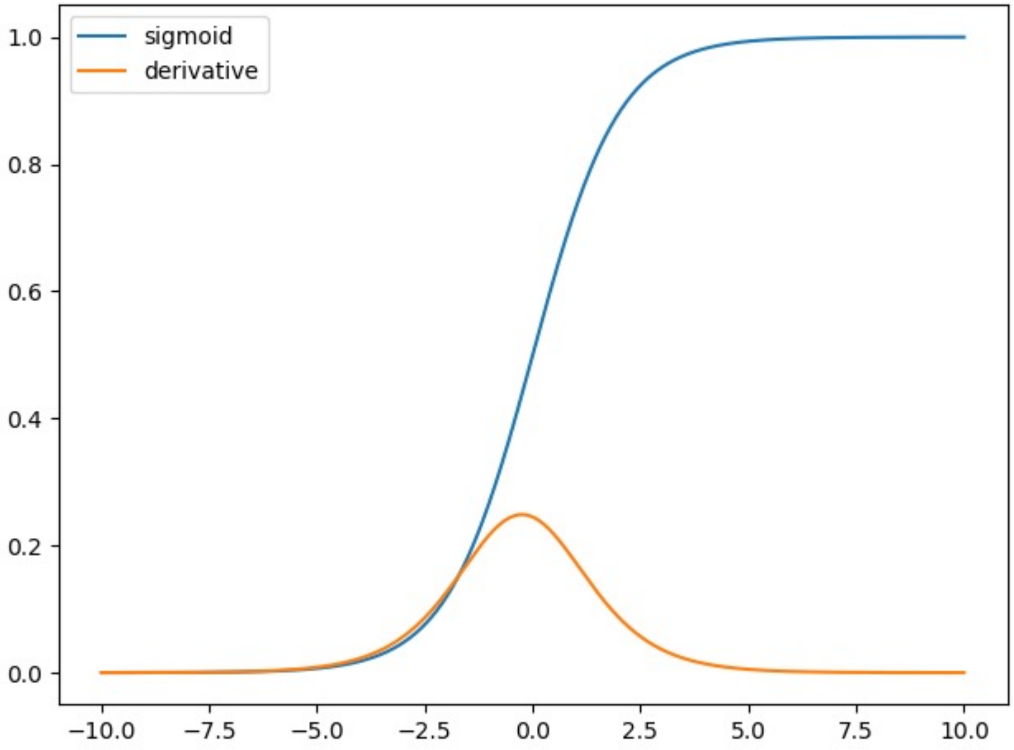
Activation

- The backpropagated gradient is multiply by the derivative of the activation function
- Vanishing gradient problem with some activation functions
- Most used: ReLu, ELU and similar

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

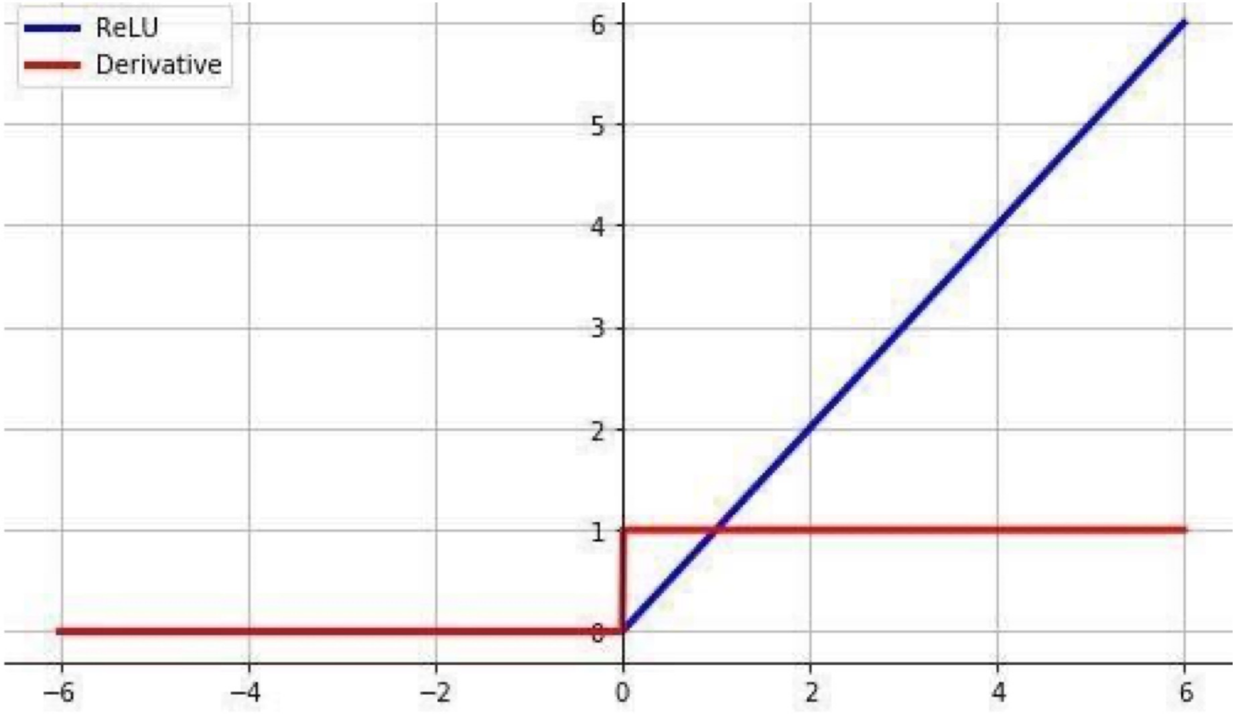
Activation



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

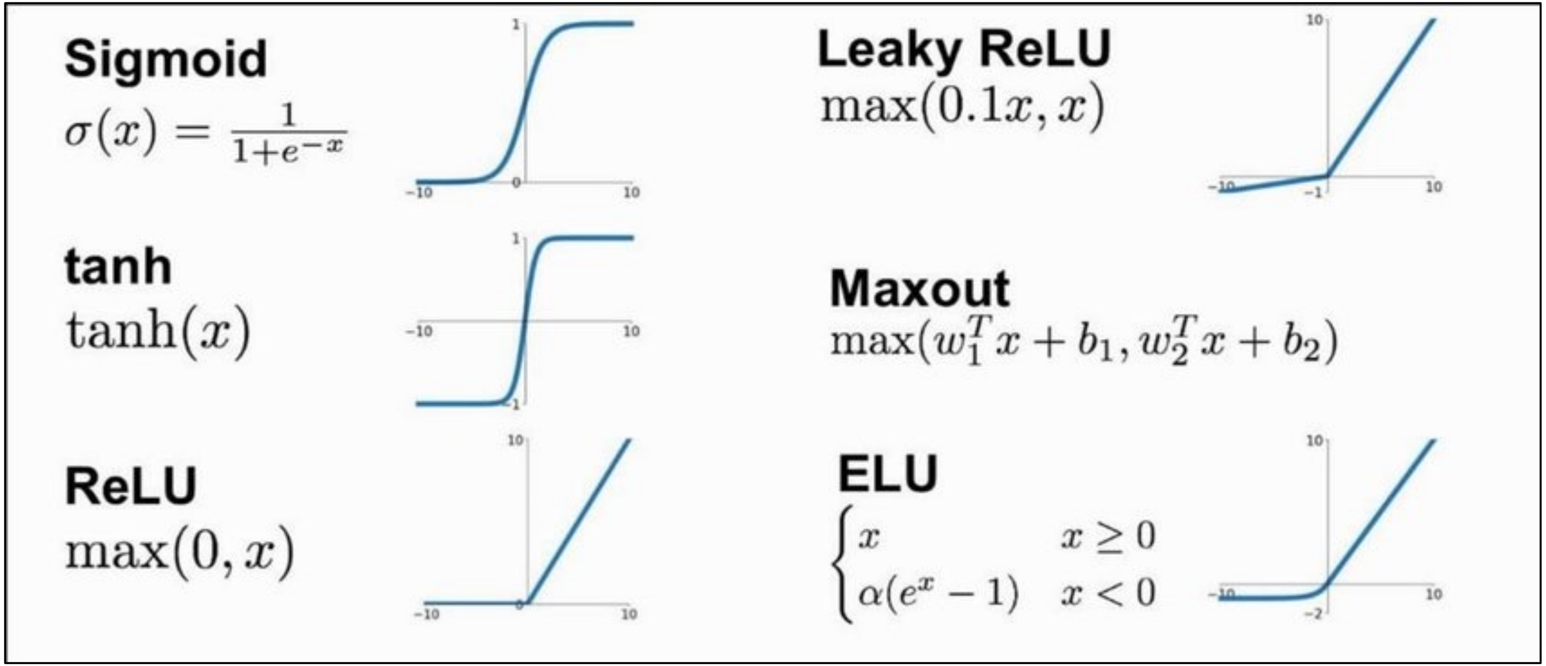
Layers

Activation



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

Activation - Softmax

Output $$y_i = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}$$ $$y_i = p(c = i|\boldsymbol{x})$$

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

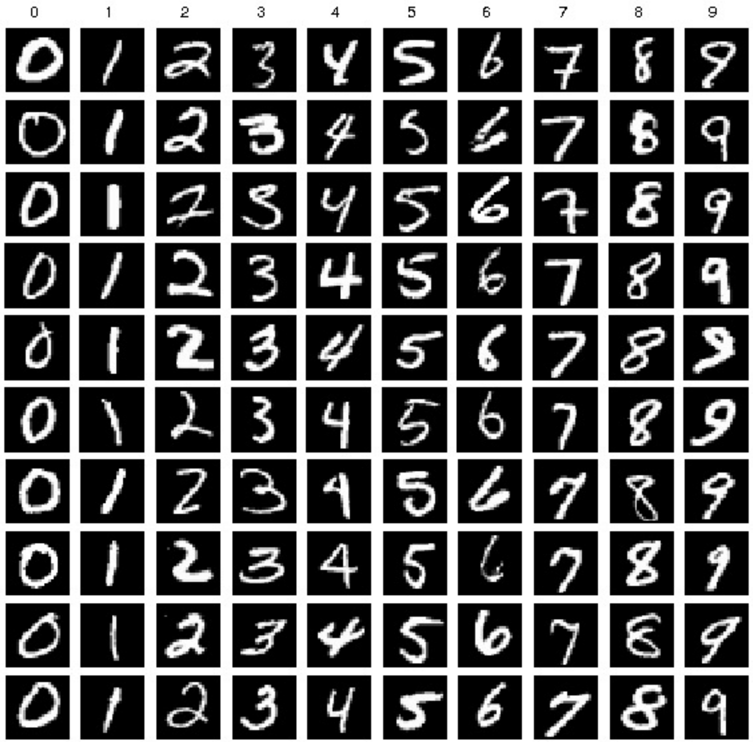
Layers

More Layers?

[Deep Learning features]

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL with Dense topology (MLP)



28x28 images (2D)

stored as 784 pixels (1D)

10 classes

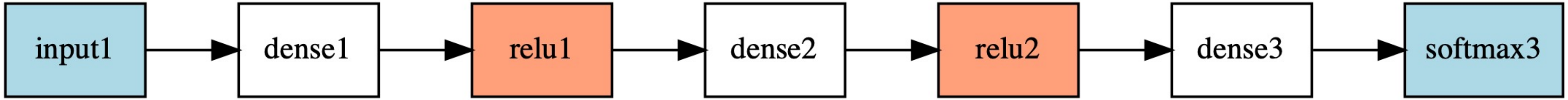
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL with Dense topology (MLP)

```
num_classes = 10

in  = eddl.Input([784])

layer = eddl.ReLu(eddl.Dense(in, 1024))
layer = eddl.ReLu(eddl.Dense(layer, 512))
out = eddl.Softmax(eddl.Dense(layer, num_classes))

net = eddl.Model([in], [out])
```

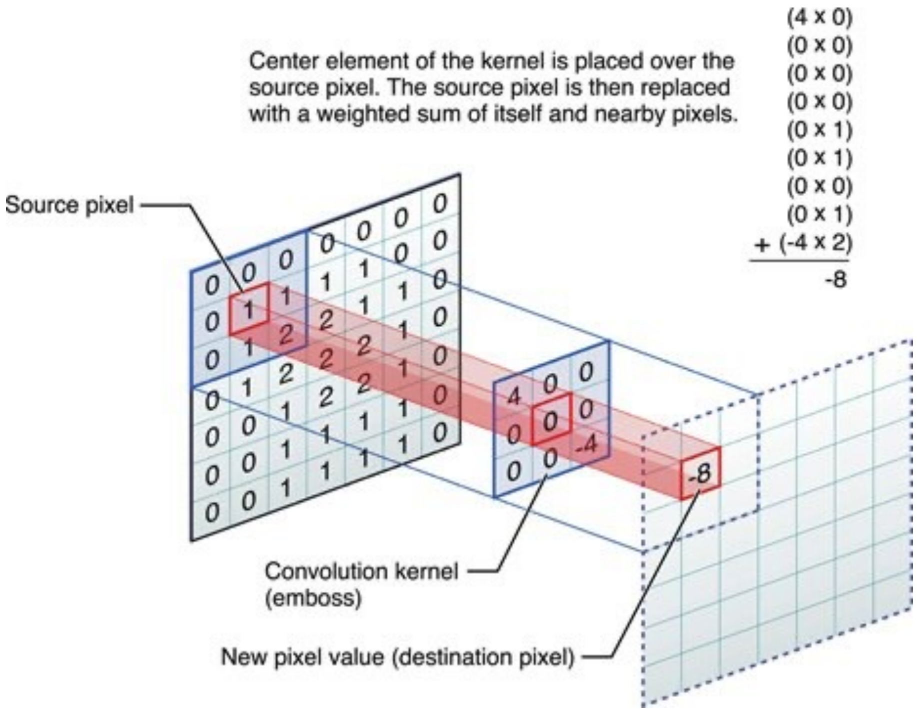

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL with Dense topology (MLP)



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers - Covolutions



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

0	0	0	0	0	0	...
0	156	155	156	158	158	...
0	153	154	157	159	159	...
0	149	151	155	158	159	...
0	146	146	149	153	158	...
0	145	143	143	148	158	...
...	...	...	...	...	...	...

Input Channel #1 (Red)

0	0	0	0	0	0	...
0	167	166	167	169	169	...
0	164	165	168	170	170	...
0	160	162	166	169	170	...
0	156	156	159	163	168	...
0	155	153	153	158	168	...
...	...	...	...	...	...	...

Input Channel #2 (Green)

0	0	0	0	0	0	...
0	163	162	163	165	165	...
0	160	161	164	166	166	...
0	156	158	162	165	166	...
0	155	155	158	162	167	...
0	154	152	152	157	167	...
...	...	...	...	...	...	...

Input Channel #3 (Blue)

-1	-1	1
0	1	-1
0	1	1

Kernel Channel #1

1	0	0
1	-1	-1
1	0	-1

Kernel Channel #2

0	1	1
0	1	0
1	-1	1

Kernel Channel #3

$$308 \quad + \quad -498 \quad + \quad 164 \quad + 1 = -25$$

Bias = 1

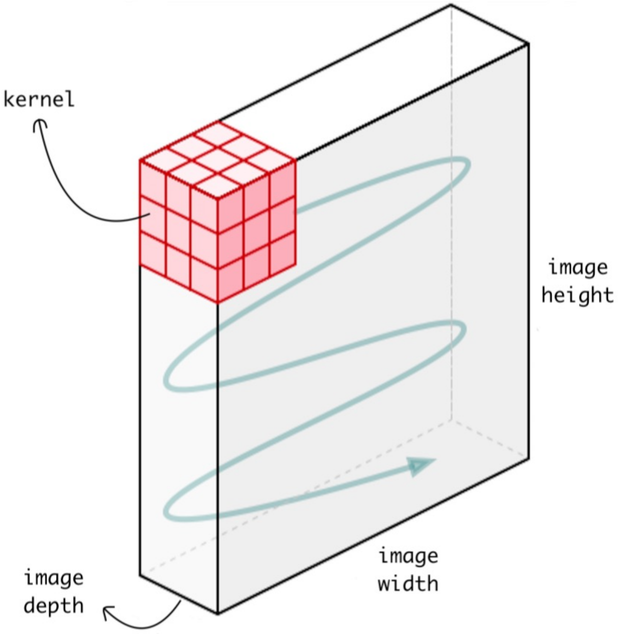
Output

-25				...
				...
				...
				...
...	...	...	...	...

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

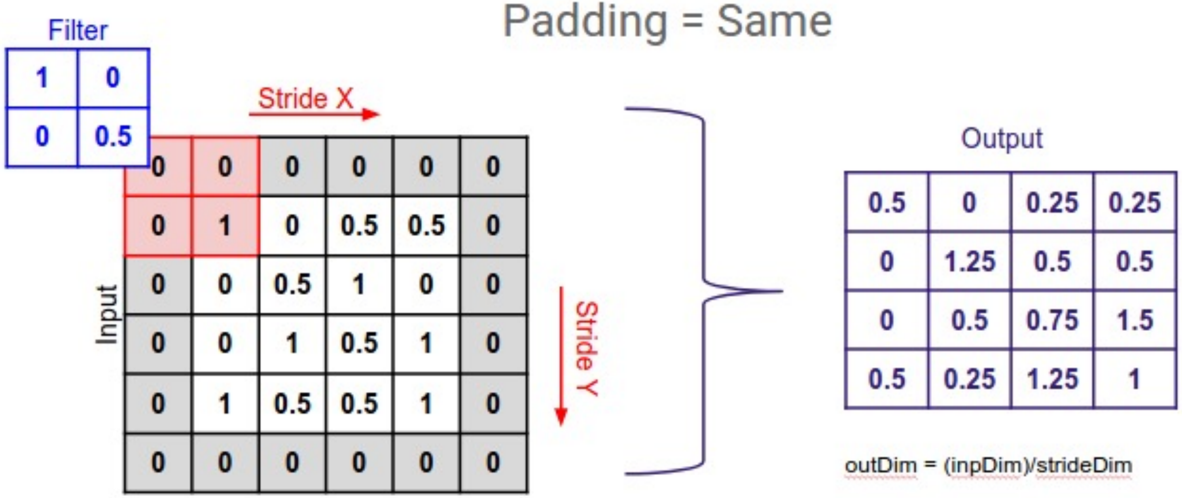
Conv2D



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

Padding = Same



outDim = (inpDim)/strideDim

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

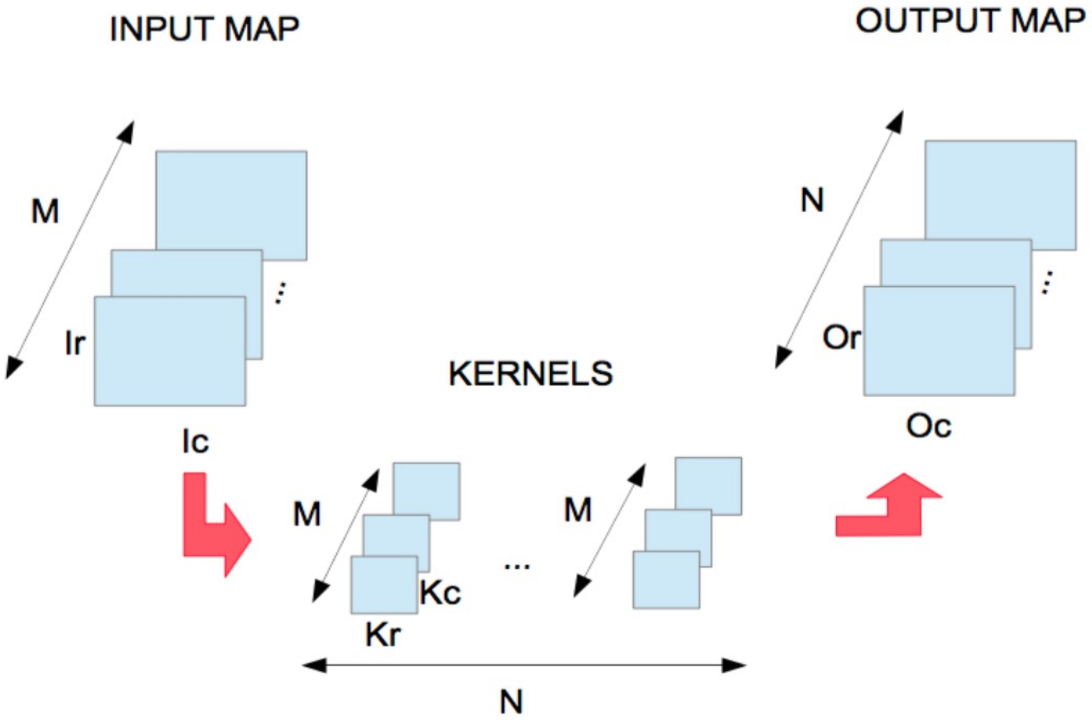
Conv2D

- Input a 3D Tensor (channels_in, height_in, width_in)
- Output a 3D Tensor (channels_out, height_in, width_out)

- Operation: $\boldsymbol{y} = \boldsymbol{K} * \boldsymbol{x} + \boldsymbol{b}$

- Parameters: K,b
  - K is a (num_filters, channels_in, kernel_height, kernel_width) 4D Tensor
  - B is a (num_filters) 1D Tensor

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

Conv2D

- Input a 3D Tensor (channels_in, height_in, width_in)
- Output a 3D Tensor (channels_out, height_in, width_out)

- channels_out = num_filters
- with padding="same"
    - height_out= height_in
    - width_out= width_in

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

Maxpool2D

- **Input a 3D Tensor (channels_in, height_in, width_in)**
- **Output a 3D Tensor (channels_in, height_in/s, width_in/s)**

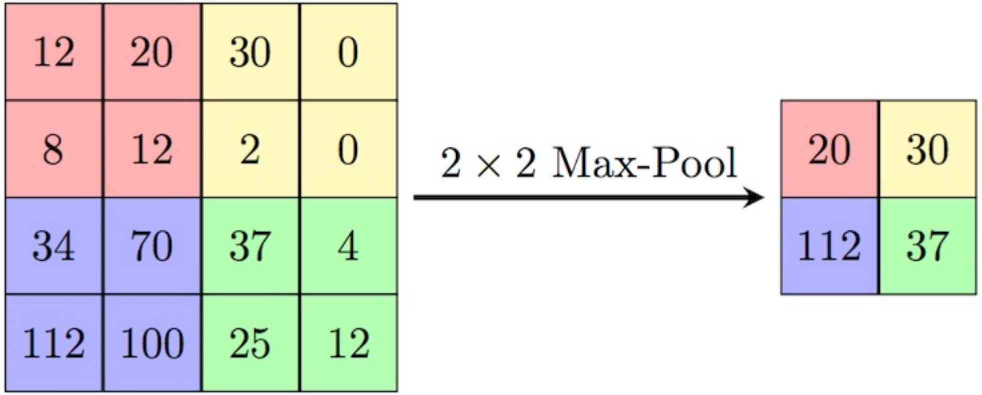
  Normally s=2

- **Parameters: None**

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

Maxpool2D

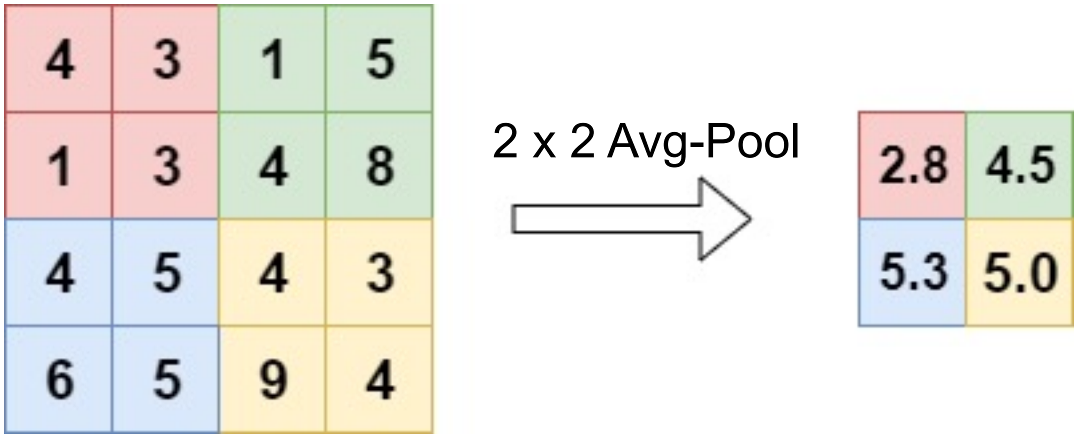


Usually stride=kernel_size

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

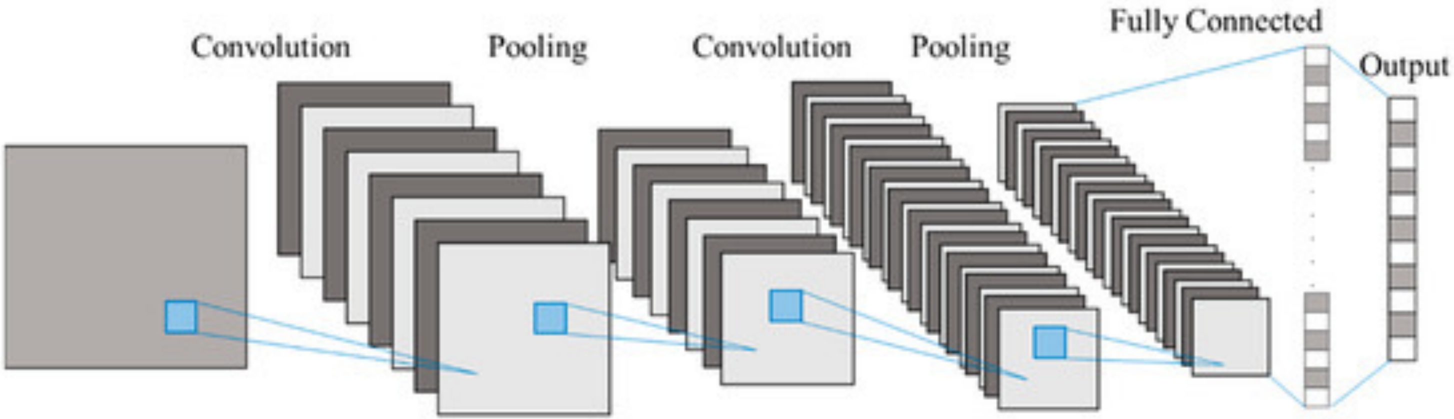
Layers

AveragePool2D



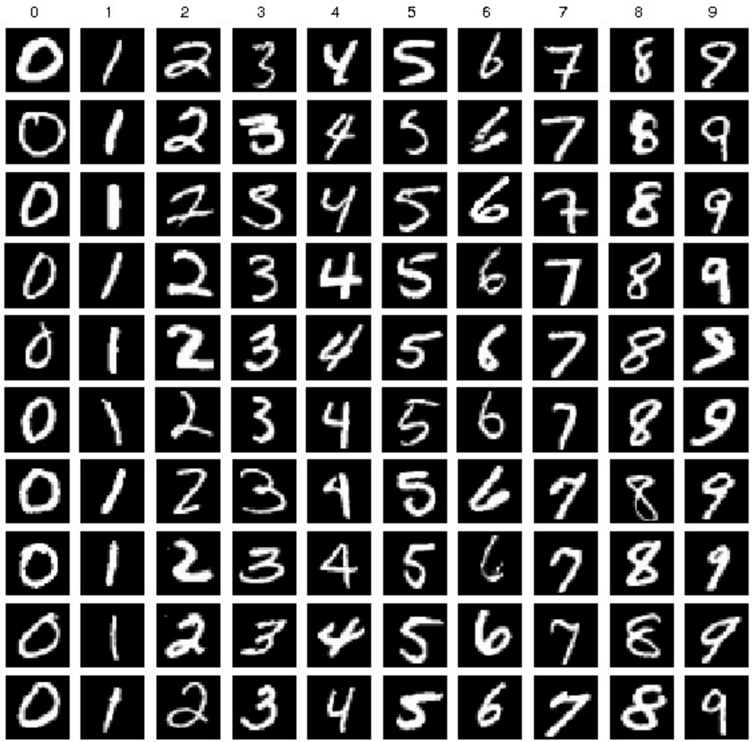
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL with Convolutions (CNN)



28x28 images (2D)

stored as 784 pixels (1D)

10 classes

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

```
num_classes = 10

in = eddl.Input([784])                         # 784 components vector
layer = eddl.Reshape(in, [1, 28, 28])          # 28 x 28 gray-level image

layer = eddl.MaxPool(eddl.ReLu(eddl.Conv(layer, 32, [3, 3])))     #(32,14,14)
layer = eddl.MaxPool(eddl.ReLu(eddl.Conv(layer, 64, [3, 3])))     #(64,7,7)
layer = eddl.MaxPool(eddl.ReLu(eddl.Conv(layer, 128, [3, 3])))    #(128,3,3)
layer = eddl.MaxPool(eddl.ReLu(eddl.Conv(layer, 256, [3, 3])))    #(256,1,1)

layer = eddl.Reshape(layer, [-1])              # 256 components vector
out = eddl.Softmax(eddl.Dense(layer, num_classes))
```


The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

eddl.summary(net)

```
-------------------------------------------------------------------------------
model
-------------------------------------------------------------------------------
input1      |  (784)              =>   (784)              0
reshape1    |  (784)              =>   (1, 28, 28)        0
conv2d1     |  (1, 28, 28)        =>   (32, 28, 28)       320
relu1       |  (32, 28, 28)       =>   (32, 28, 28)       0
maxpool2d2|  (32, 28, 28)       =>   (32, 14, 14)       0
conv2d2     |  (32, 14, 14)       =>   (64, 14, 14)       18496
relu2       |  (64, 14, 14)       =>   (64, 14, 14)       0
maxpool2d4|  (64, 14, 14)       =>   (64, 7, 7)         0
conv2d3     |  (64, 7, 7)         =>   (128, 7, 7)        73856
relu3       |  (128, 7, 7)        =>   (128, 7, 7)        0
maxpool2d6|  (128, 7, 7)        =>   (128, 3, 3)        0
conv2d4     |  (128, 3, 3)        =>   (256, 3, 3)        295168
relu4       |  (256, 3, 3)        =>   (256, 3, 3)        0
maxpool2d8|  (256, 3, 3)        =>   (256, 1, 1)        0
reshape2    |  (256, 1, 1)        =>   (256)              0
dense1      |  (256)              =>   (10)               2570
softmax5    |  (10)               =>   (10)               0
-------------------------------------------------------------------------------
```


The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

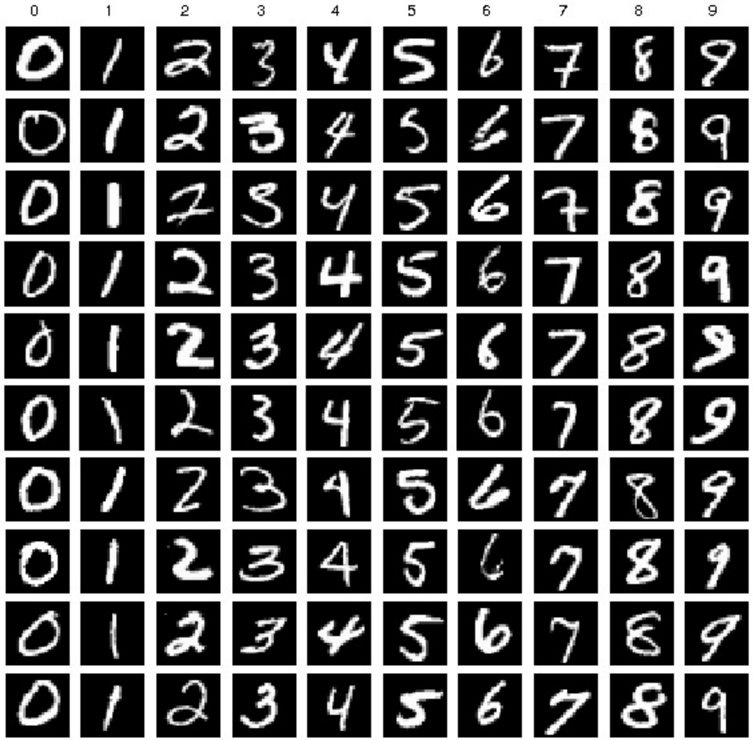
Recurrent Layers:

- Vanilla RNN
- LSTM
- GRU

All accept a 1D tensor (vector) as Input and returns 1D vector as Output

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL with Recurrent topology (RNN)



28x28 images (2D)

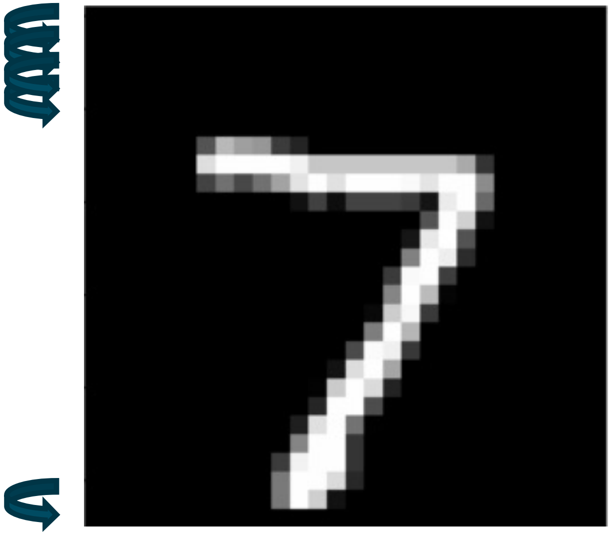
stored as 784 pixels (1D)

10 classes

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

EDDL with Dense topology (MLP)

s
e
q
u
e
n
c
e



We are going to consider the digits as a sequence of 28 rows of 28 pixels

Then the recurrent layers expect to receive a 1D tensor of 28 components

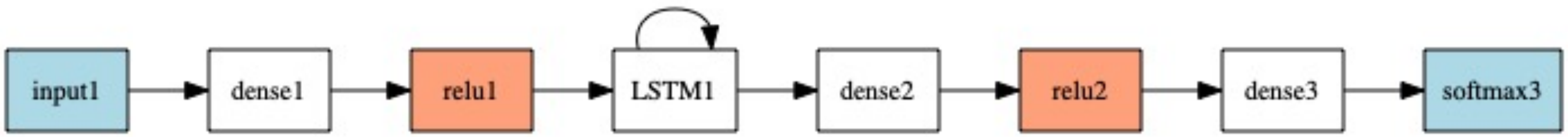
The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers

```
num_classes = 10

in = eddl.Input([28])
layer = eddl.ReLu(eddl.Dense(in, 32))
layer = eddl.LSTM(layer, 128)
out = eddl.Softmax(eddl.Dense(layer, num_classes))

…

x_train.reshape_([x_train.shape[0], 28, 28]).      # (N, sequence_length, dim)
```


The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Layers



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.

Thank you!

Roberto Paredes

rparedes@prhlt.upv.es

The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825111.