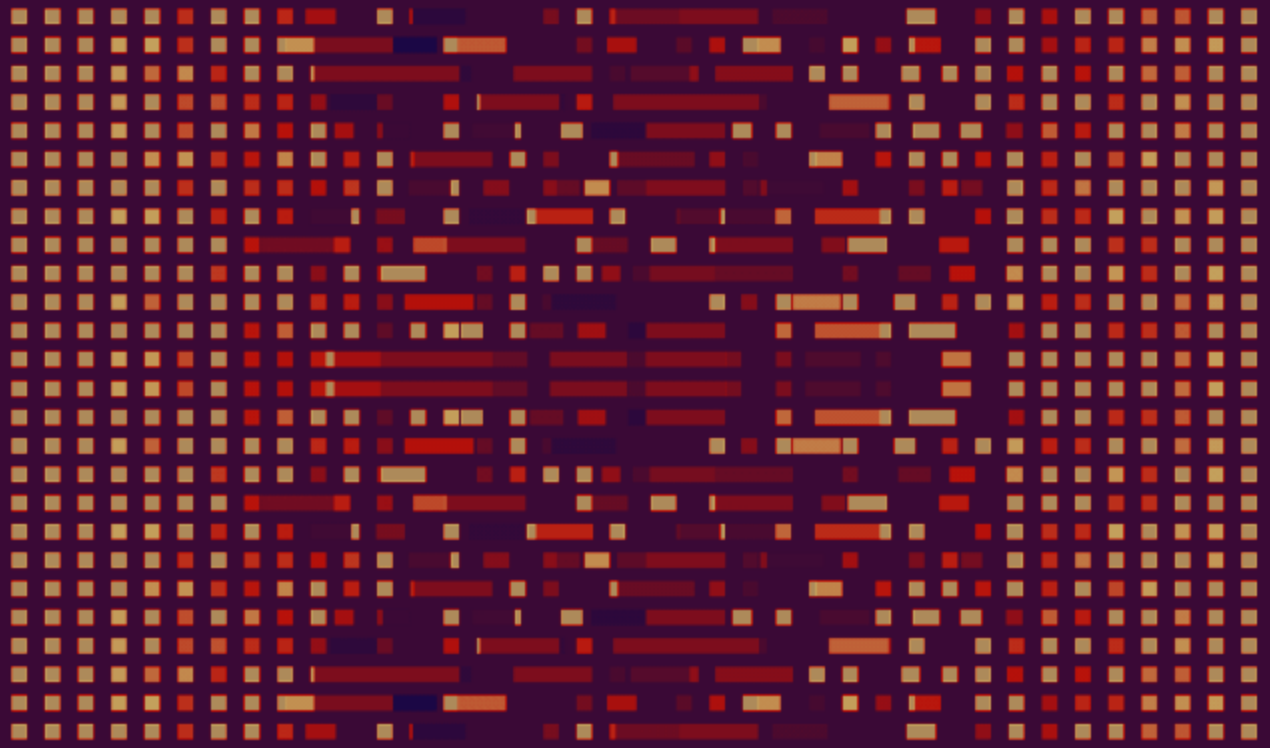
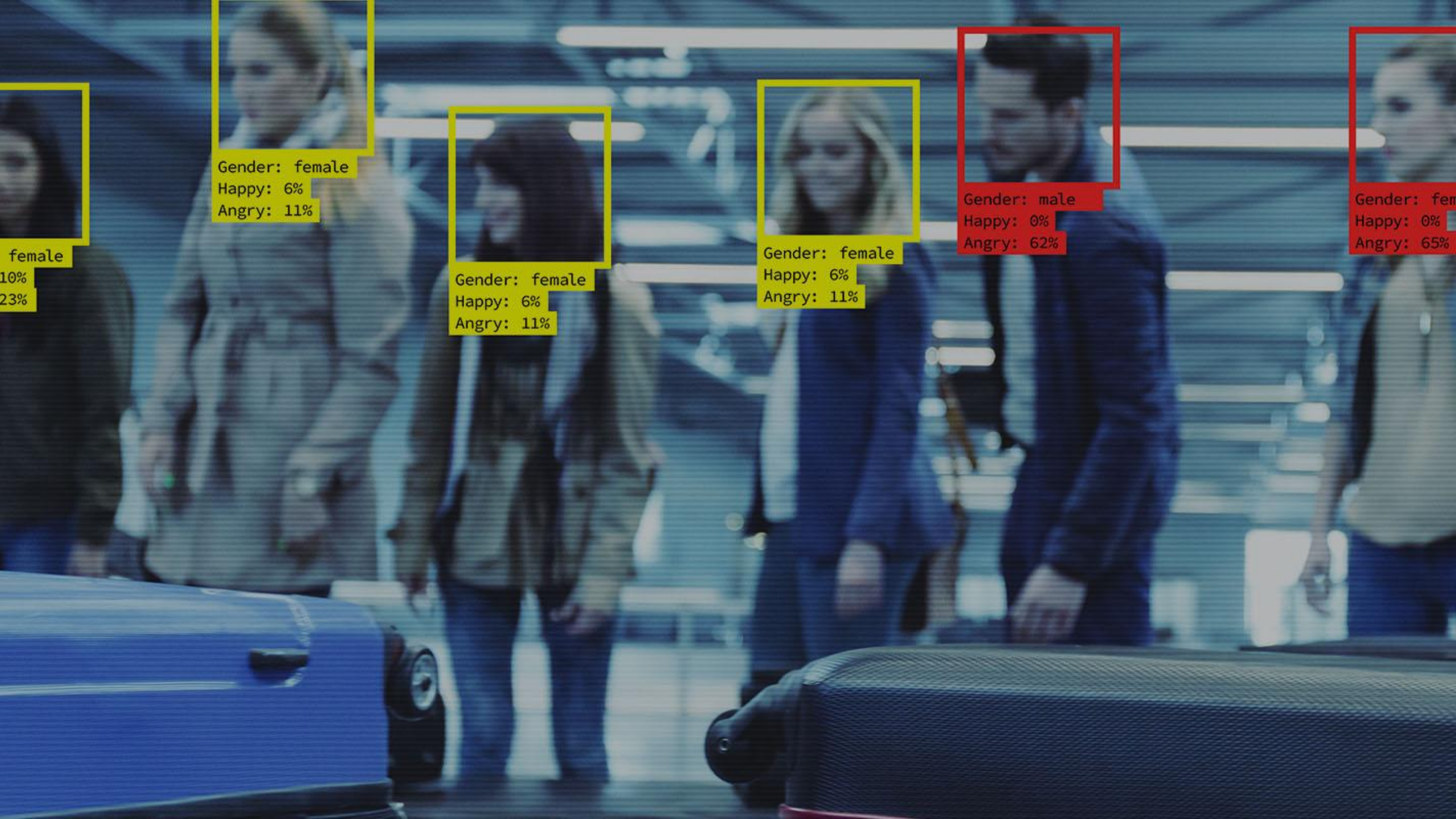
Intellectus

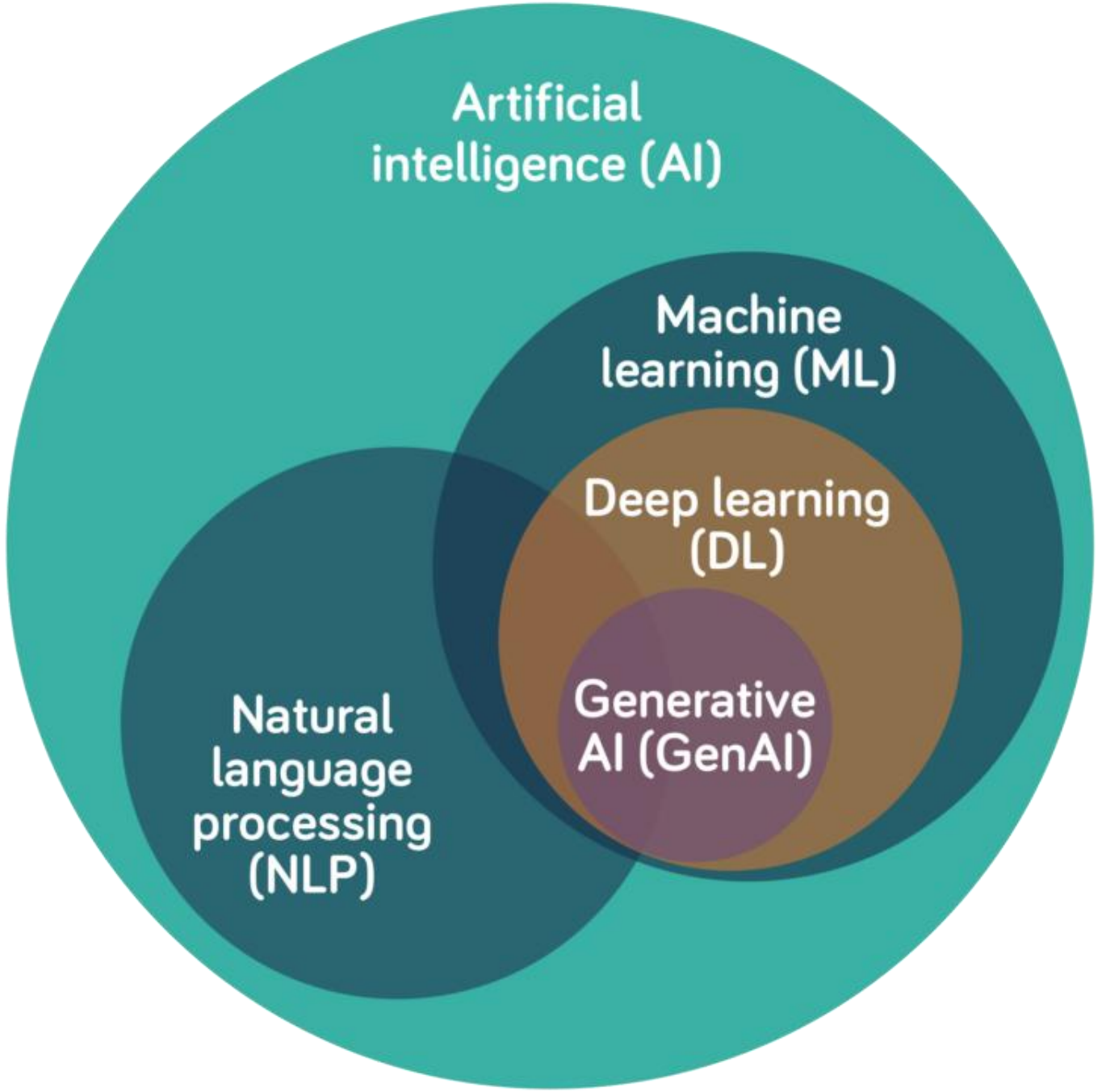
Mehdi Maleki

2024

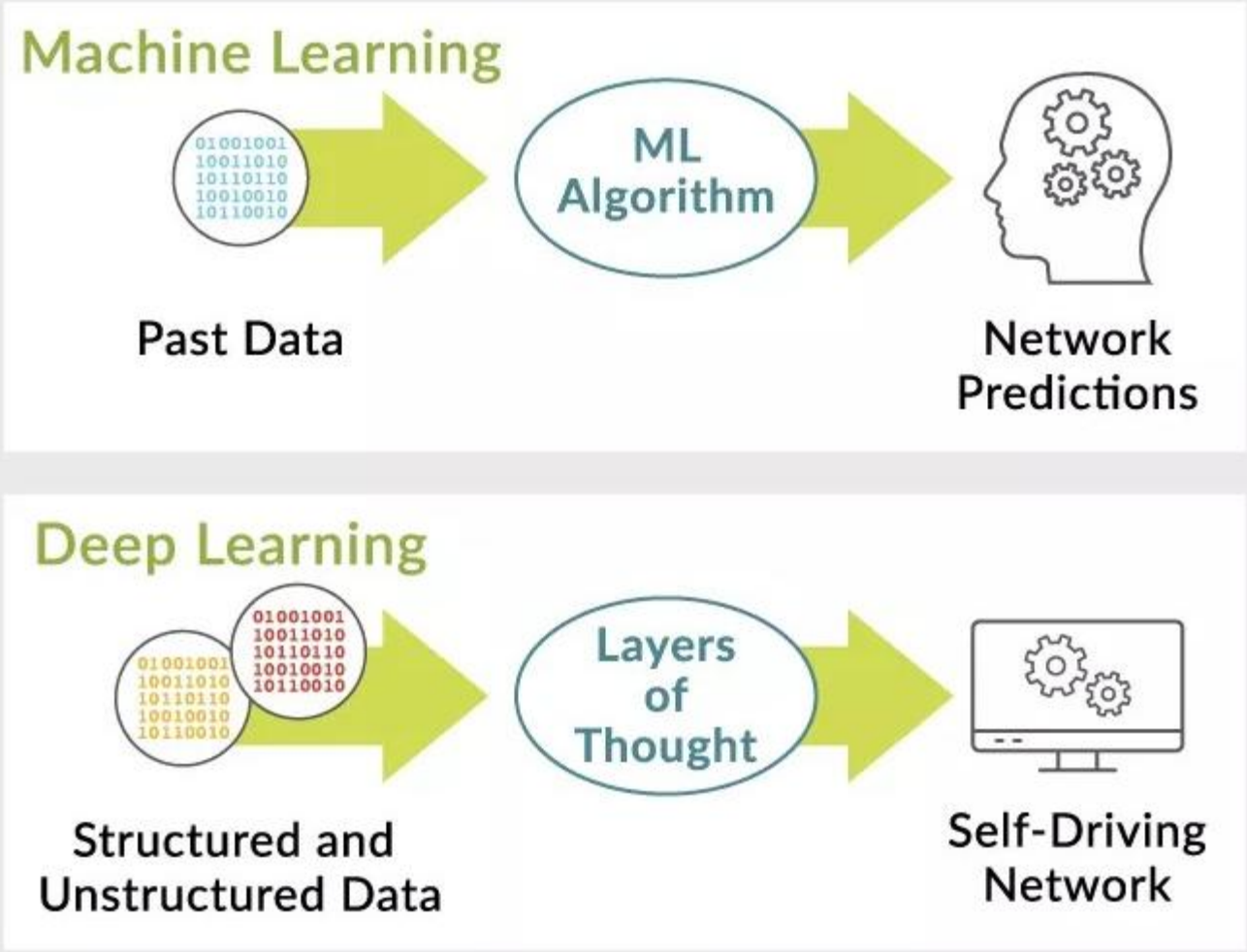


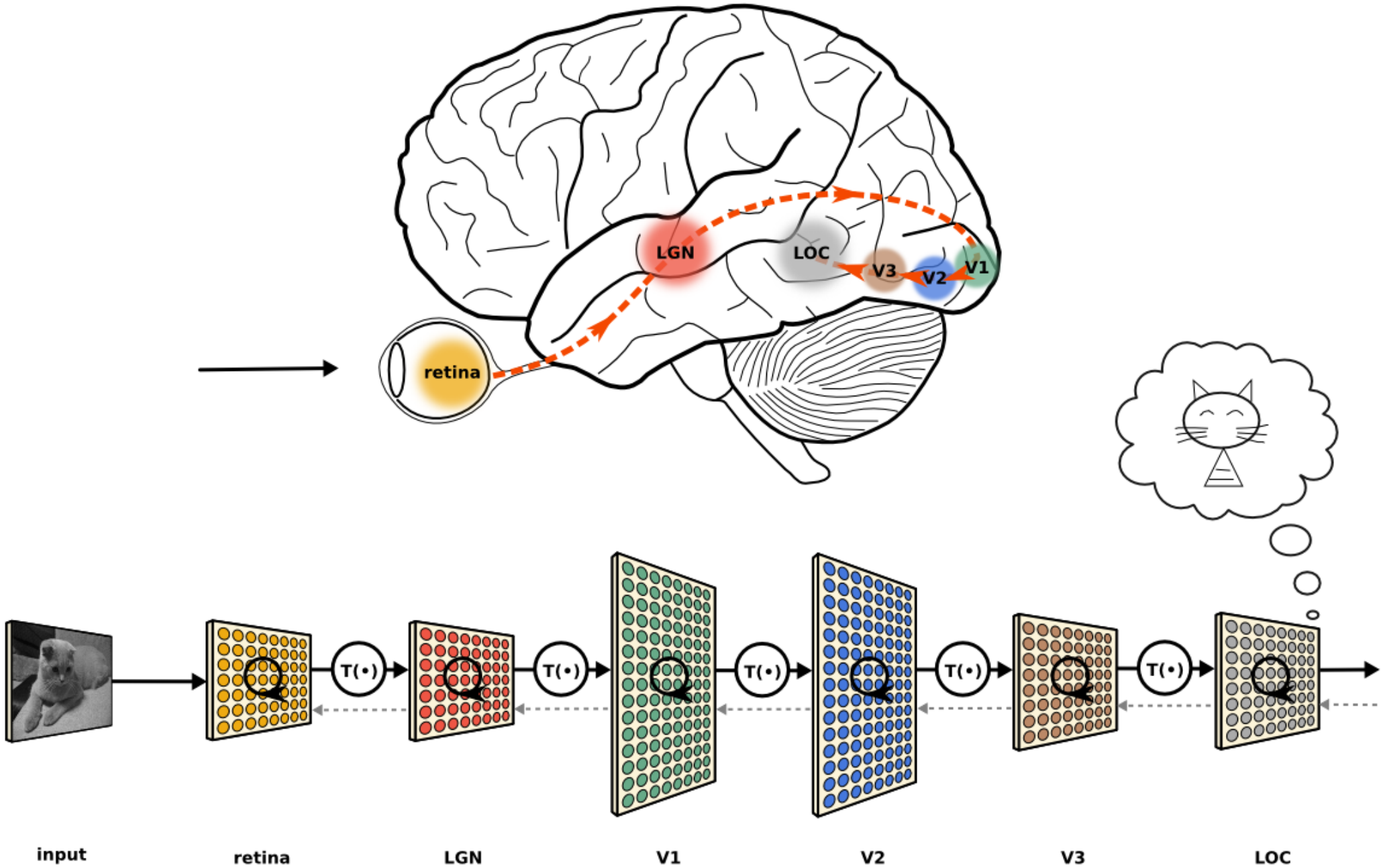

.5
1×
2
3
CINEMATIC
VIDEO
PHOTO
PORTRAIT
PANO

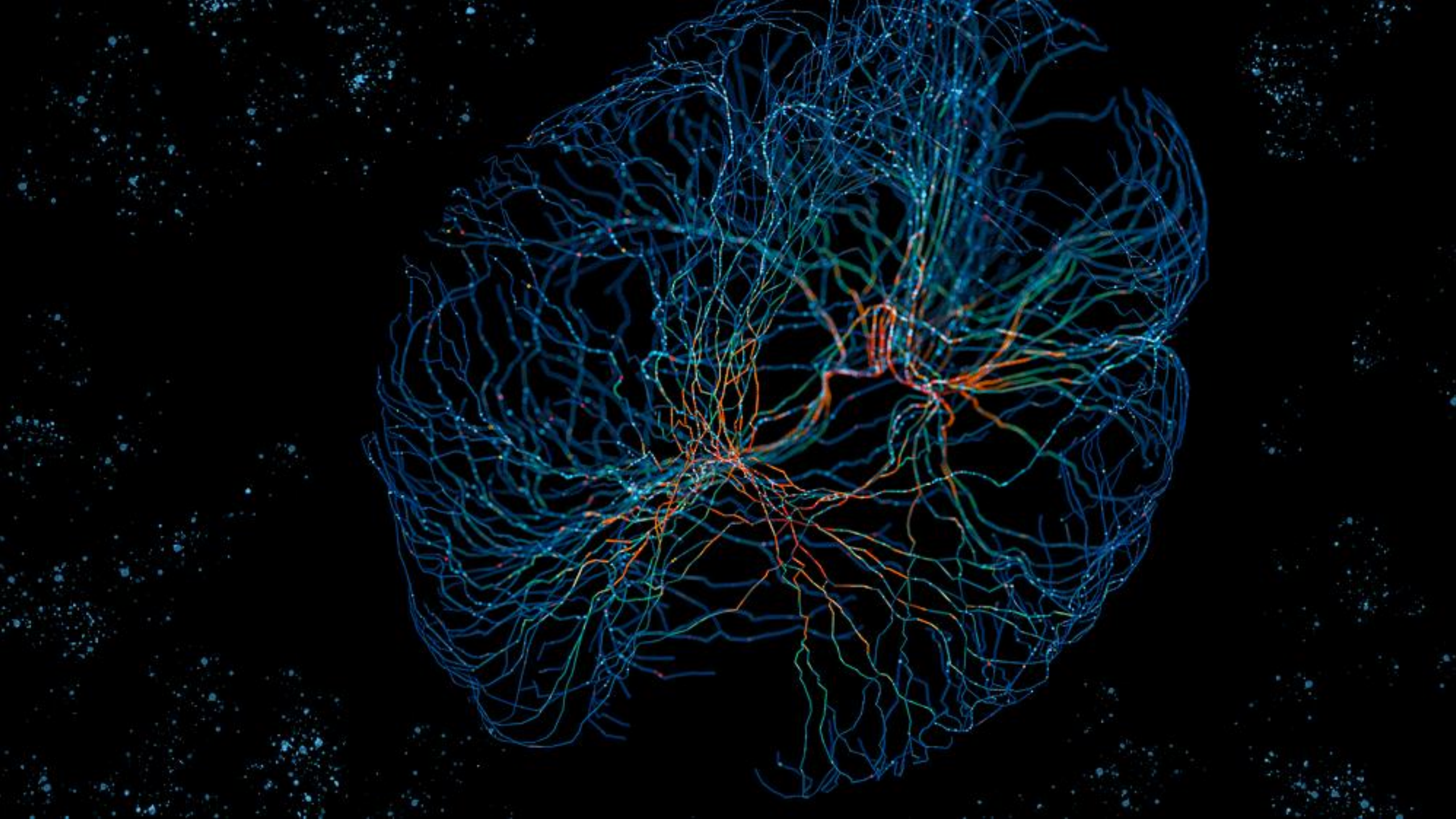

female
10%
23%
Gender: female
Happy: 6%
Angry: 11%
Gender: female
Happy: 6%
Angry: 11%
Gender: female
Happy: 6%
Angry: 11%
Gender: male
Happy: 0%
Angry: 62%
Gender: fem
Happy: 0%
Angry: 65%

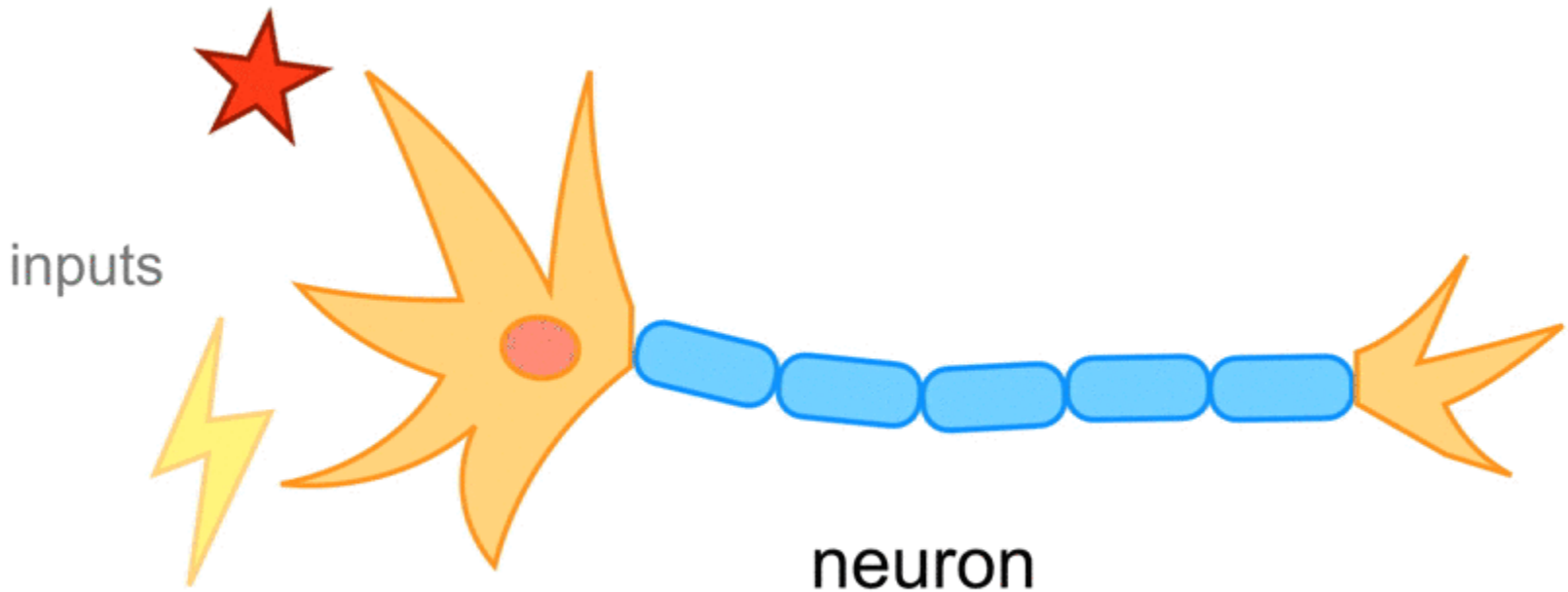

Artificial intelligence (AI)
Machine learning (ML)
Deep learning (DL)
Generative AI (GenAI)
Natural language processing (NLP)

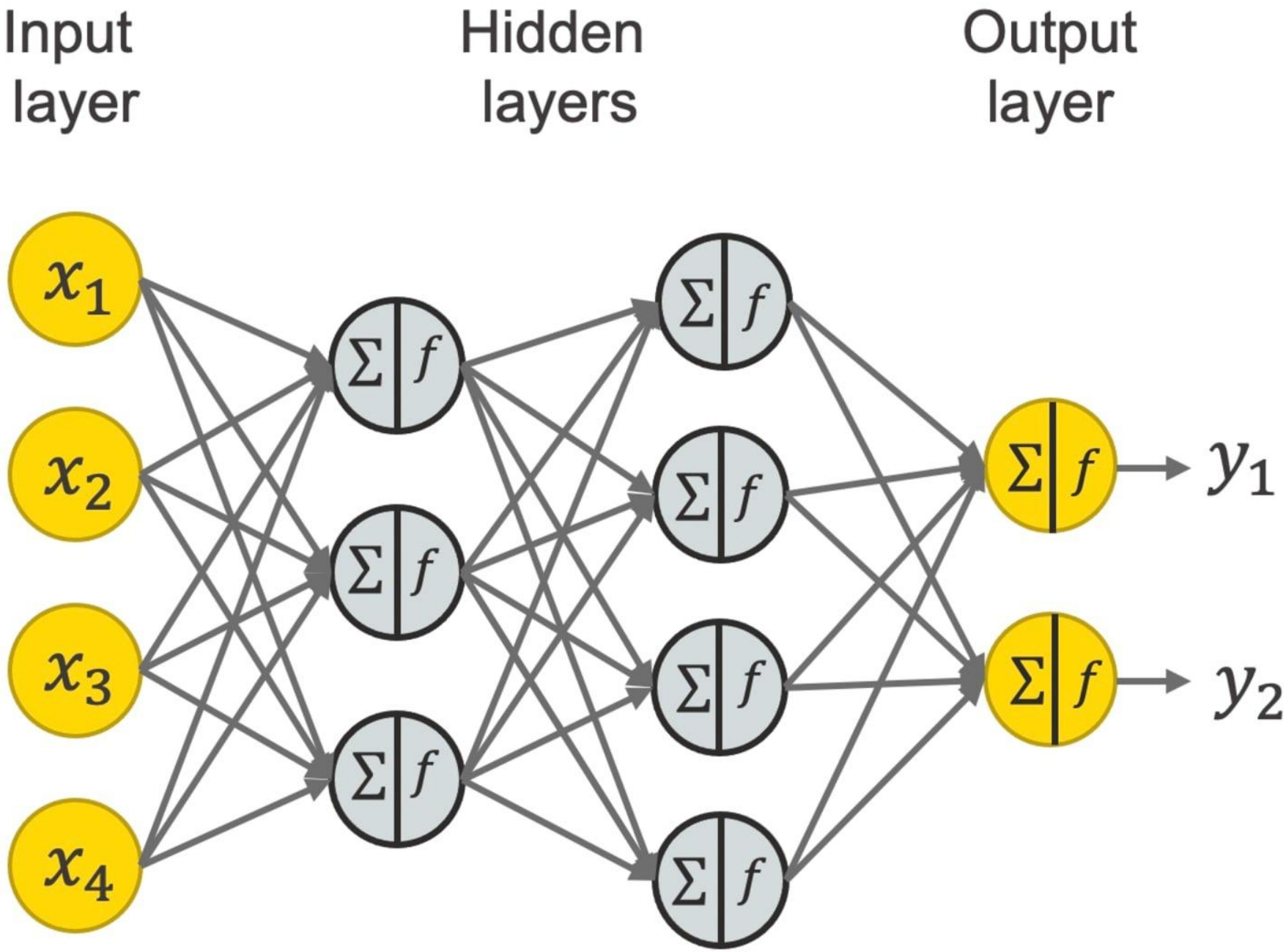
Artificial Intelligence

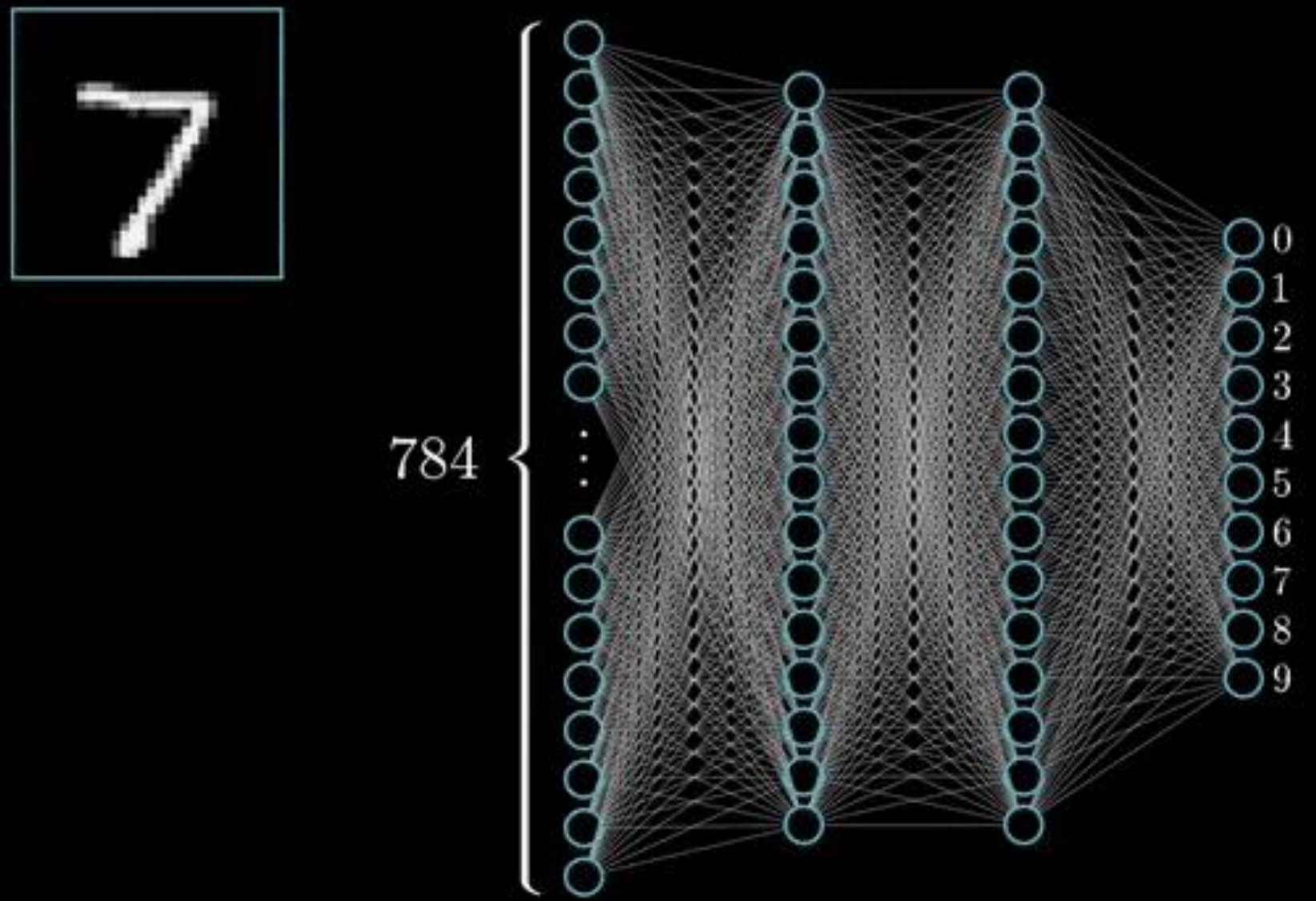


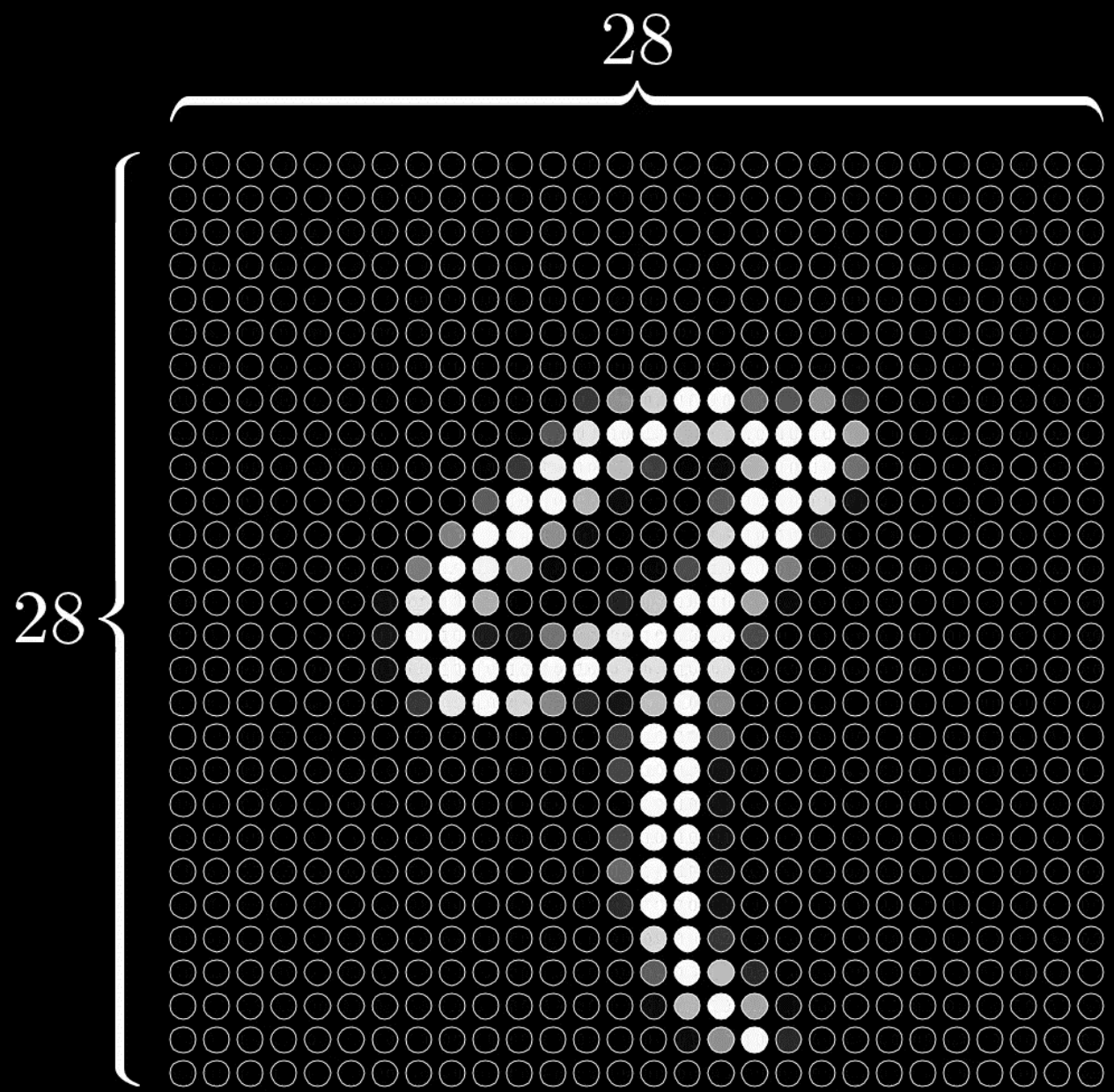

LGN
LOC
V3
V2
V1
retina
T(•)
T(•)
T(•)
T(•)
T(•)
input
retina
LGN
V1
V2
V3
LOC



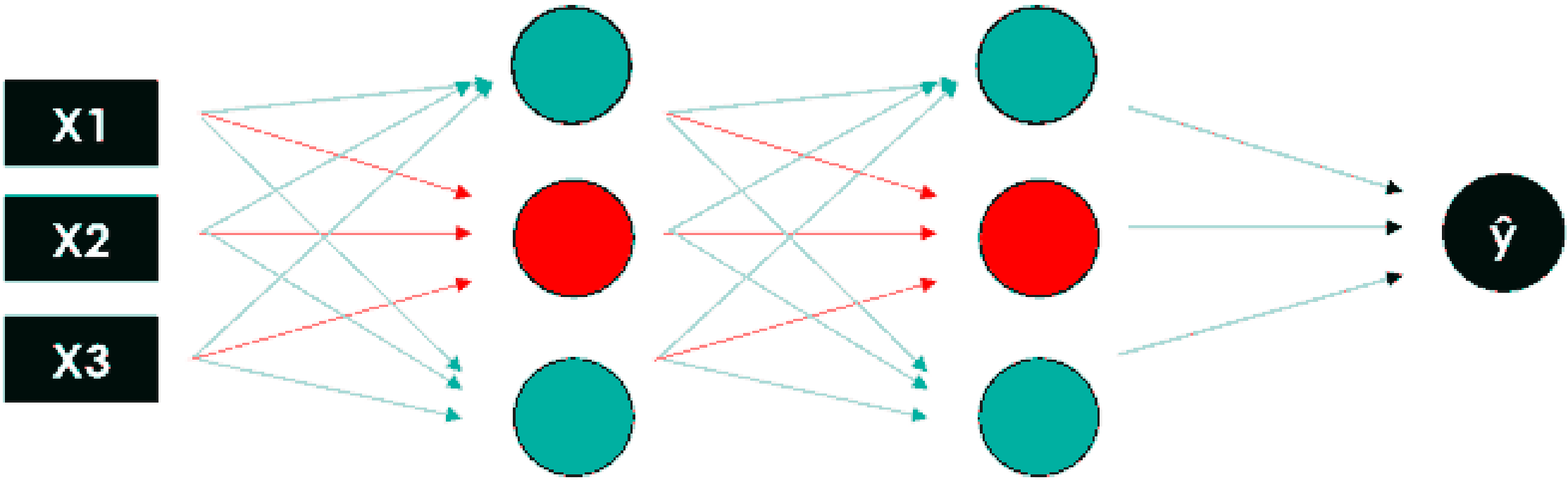

inputs
neuron

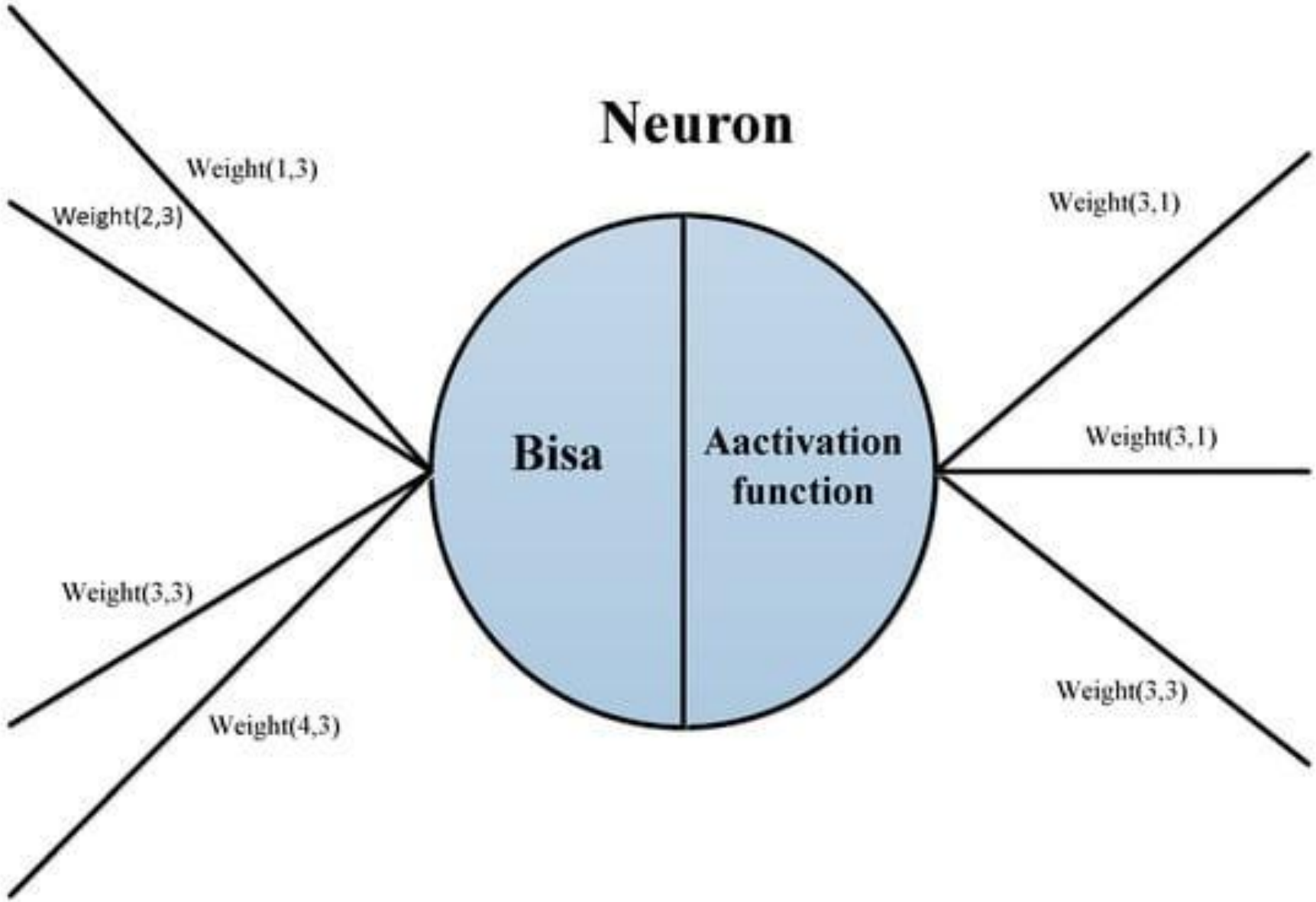

Input layer
Hidden layers
Output layer
x_1
x_2
x_3
x_4
Σ f
y_1
y_2

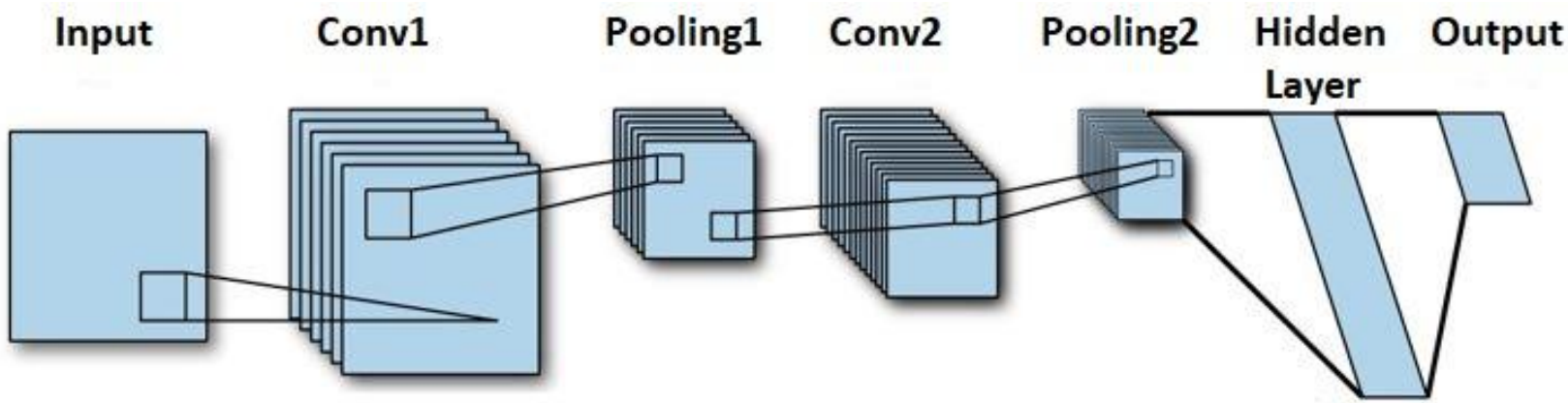

784
0
1
2
3
4
5
6
7
8
9



$$28 \times 28 = 784$$

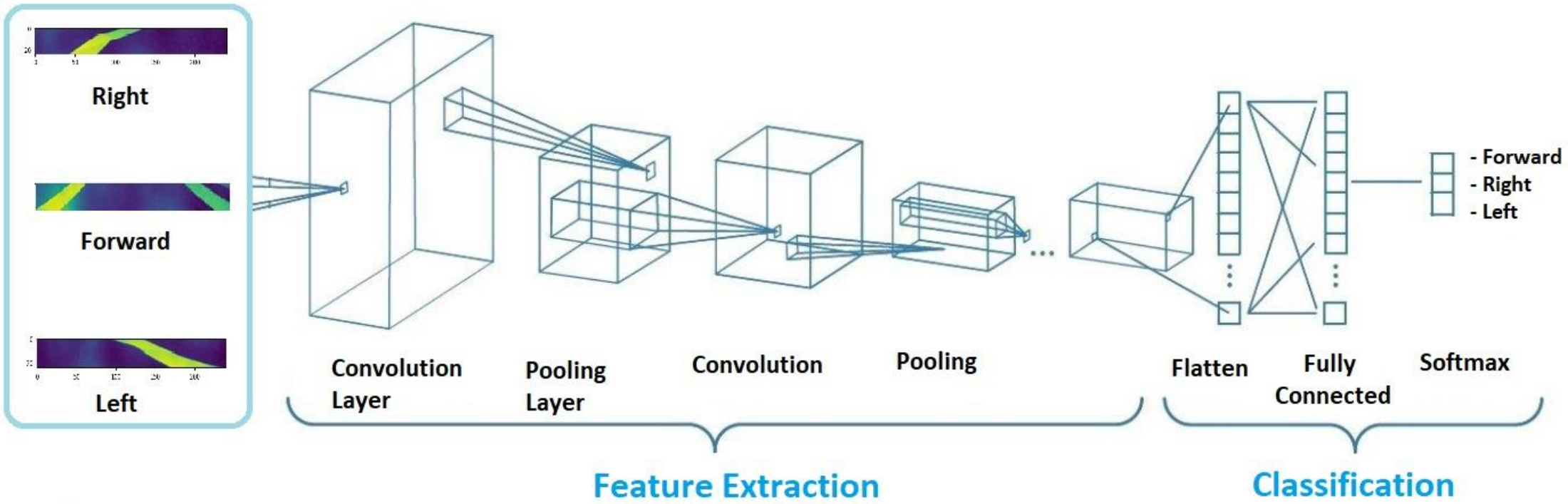

X1
X2
X3
ŷ

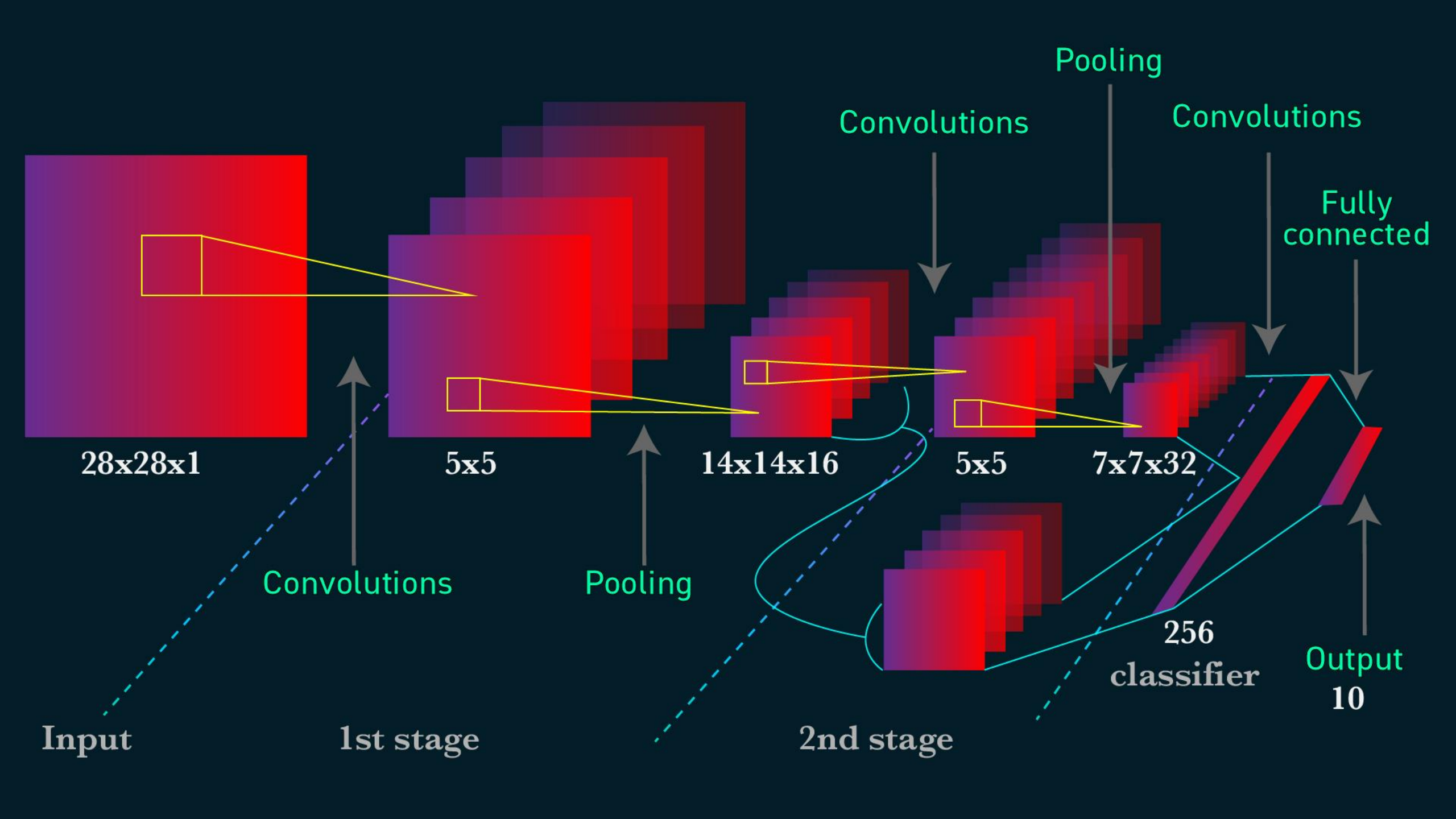

Neuron
Weight(1,3)
Weight(2,3)
Weight(3,3)
Weight(4,3)
Bisa
Aactivation function
Weight(3,1)
Weight(3,1)
Weight(3,3)


Input
Conv1
Pooling1
Conv2
Pooling2
Hidden Layer
Output

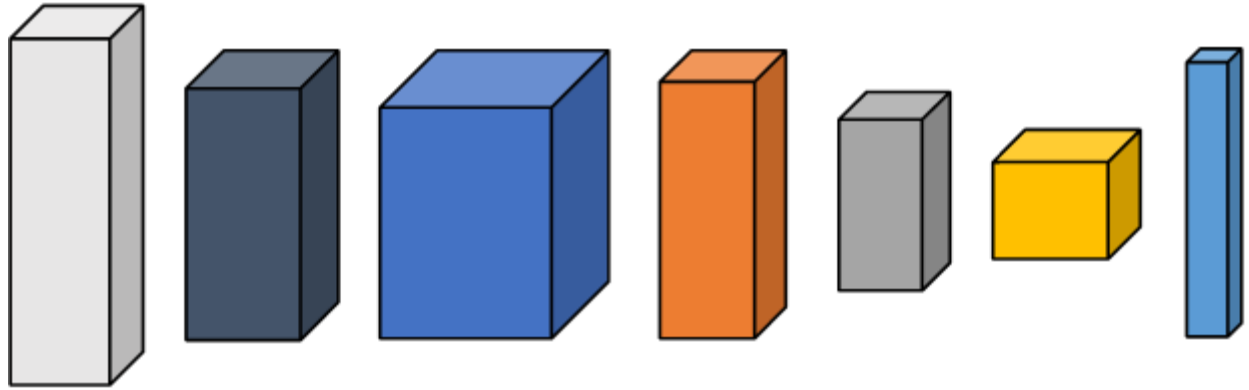
S: Spatial => Size
B: Batch => Sample Count
C: Channel => Filter Count

Sigmoid: 0 -1 Range
Element-wise Multiplication: Element x Element
SplitLayer: Split Inputs to Different Paths

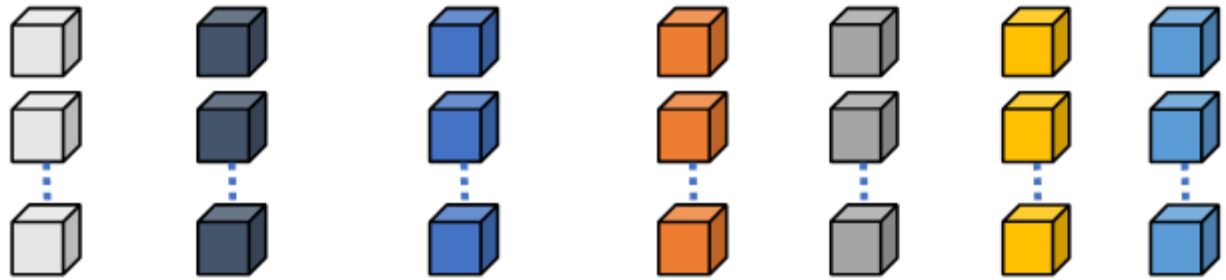

Right
Forward
Left
Convolution Layer
Pooling Layer
Convolution
Pooling
Flatten
Fully Connected
Softmax
- Forward
- Right
- Left
Feature Extraction
Classification


Pooling
Convolutions
Convolutions
Fully
connected
28x28x1
5x5
14x14x16
5x5
7x7x32
Convolutions
Pooling
256
classifier
Output
10
Input
1st stage
2nd stage

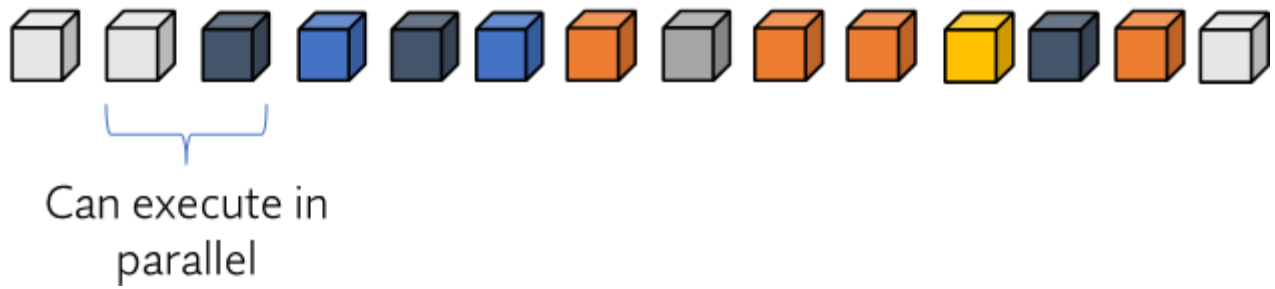
Neural Network layers

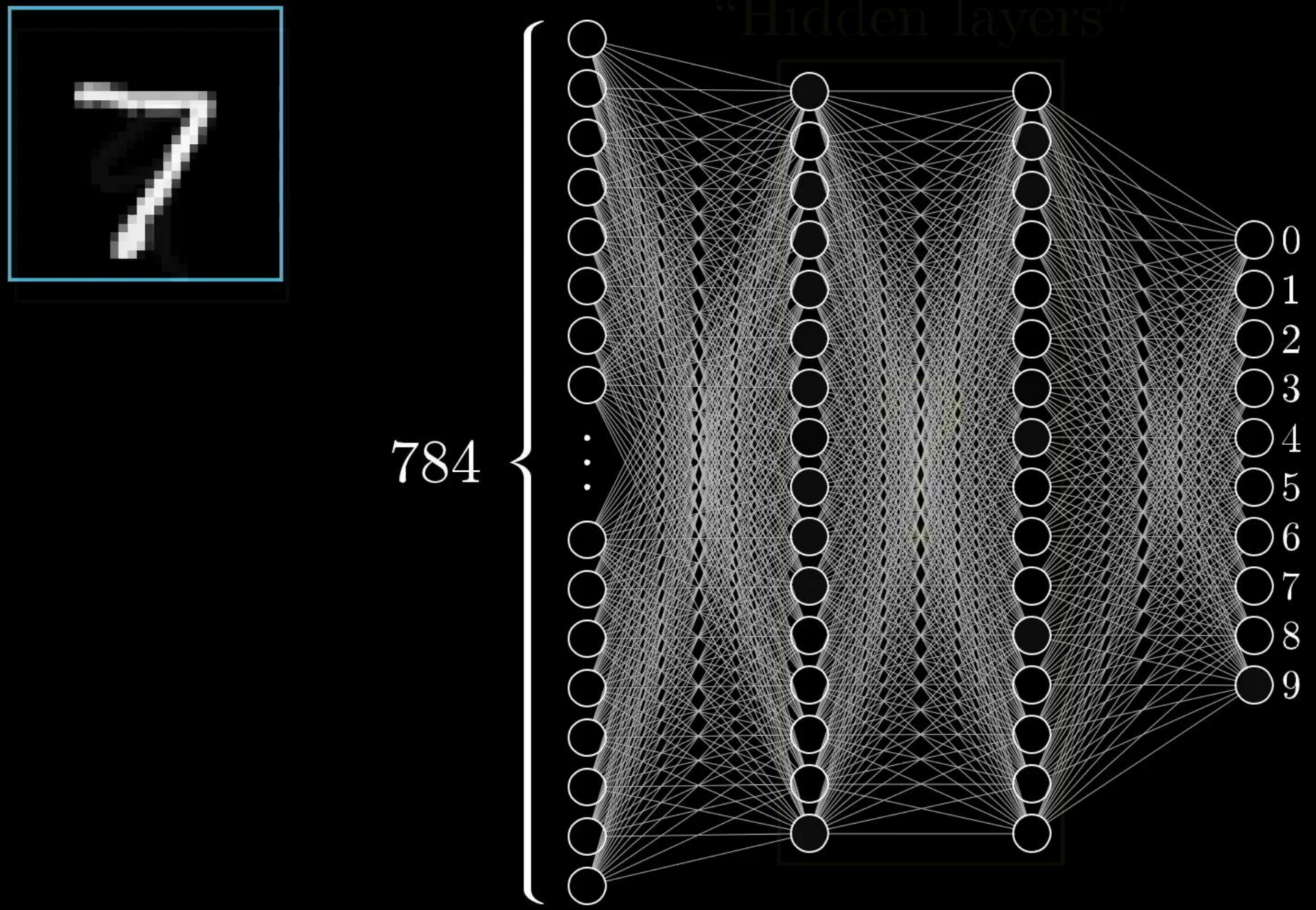


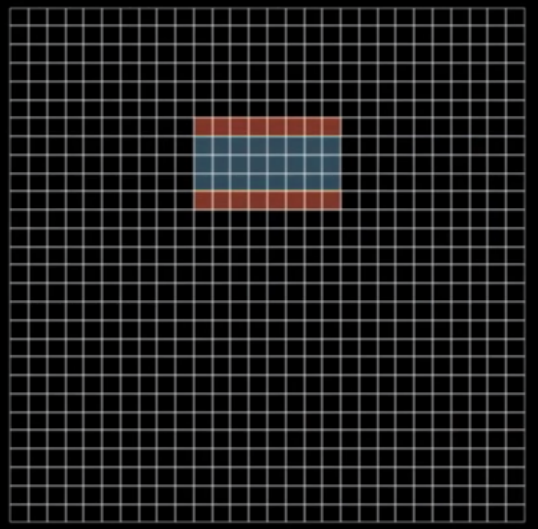
Layer broken down into packets

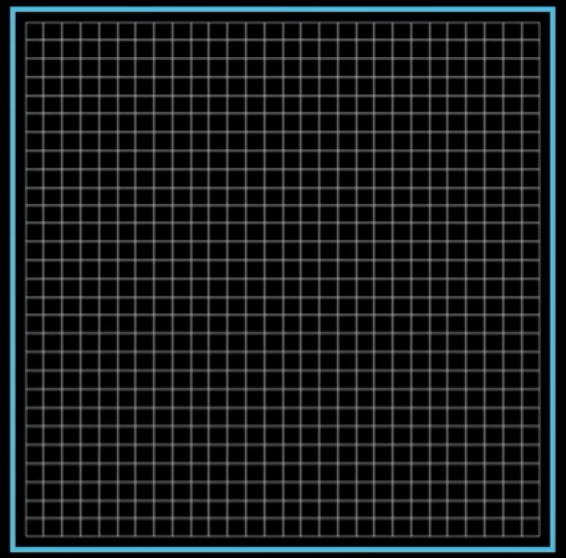


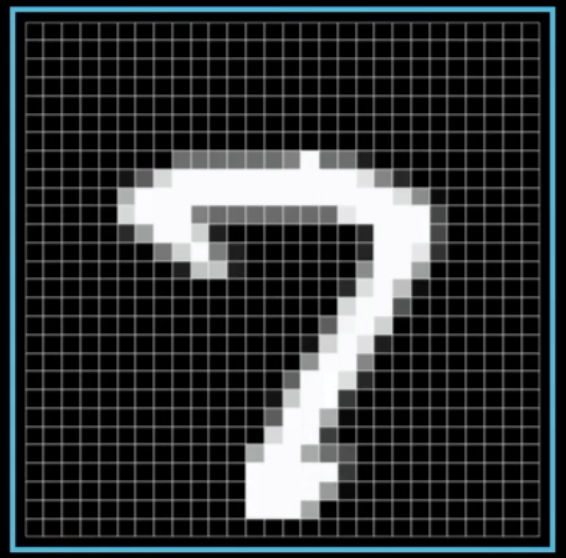
Packet stream natively executable on NPU

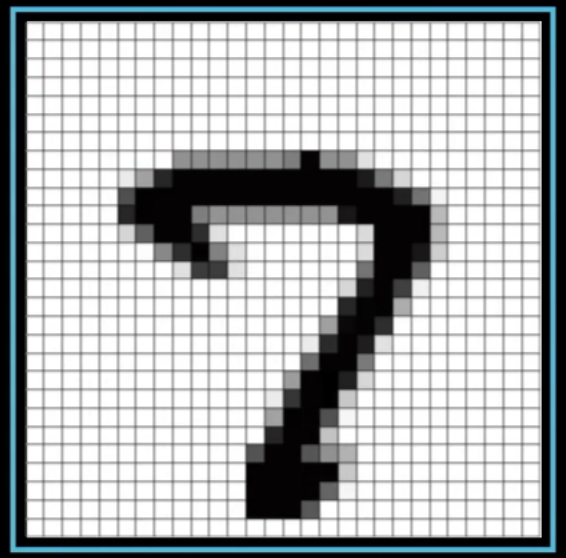


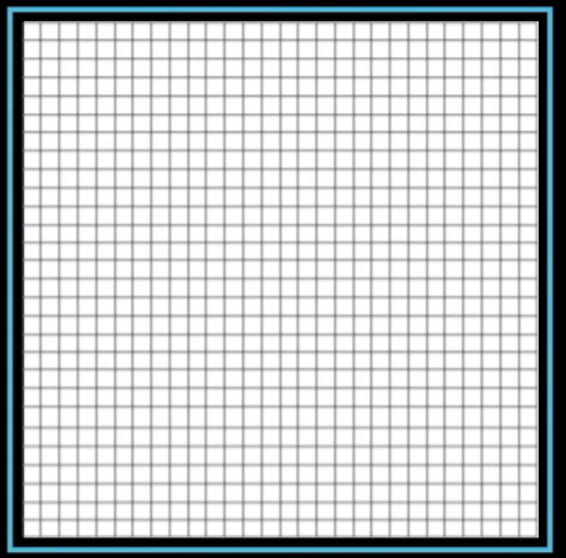

"Hidden" layers
784
0
1
2
3
4
5
6
7
8
9



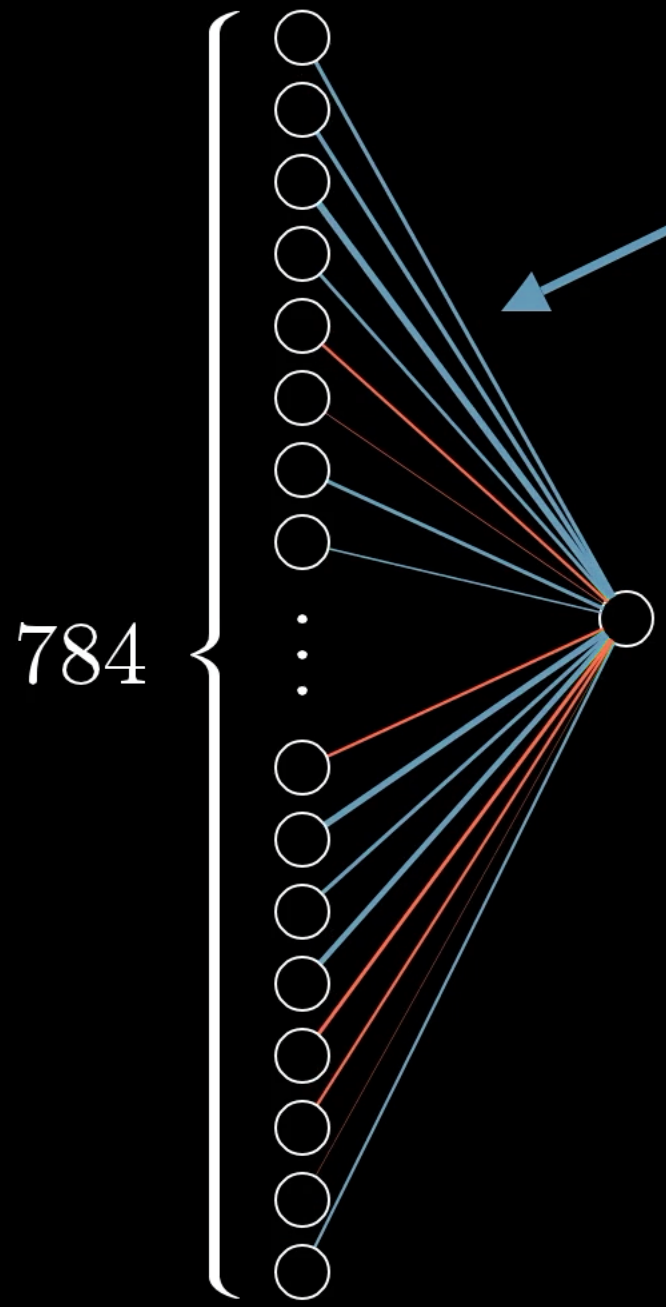






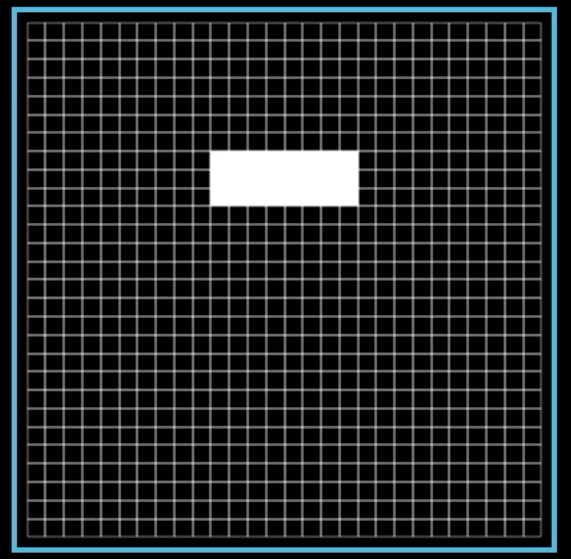


A B C D



Weights

w_1: 2.07
w_2: 2.31
w_3: 3.64
w_4: 1.87
w_5: −1.51
w_6: −0.43
w_7: 2.01
w_8: 1.07
⋮



$$\sigma\left(w_1 a_1 + w_2 a_2 + w_3 a_3 + \cdots + w_n a_n \boxed{-10}\right)$$

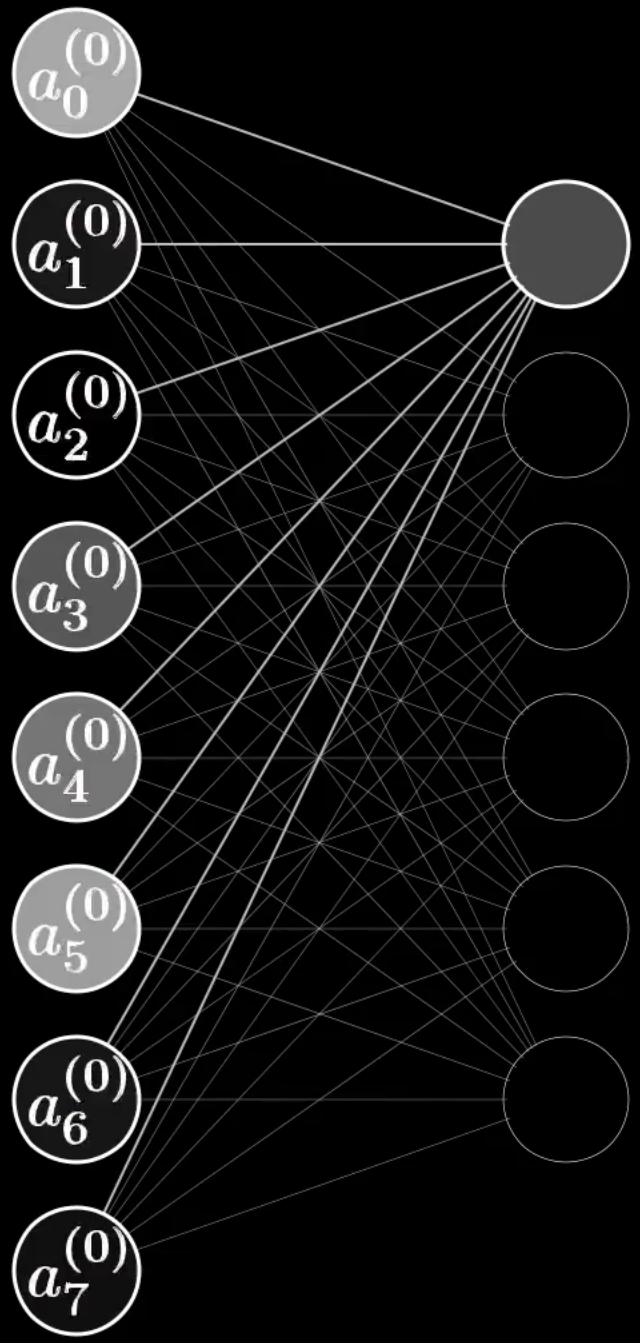
"bias"

Only activate meaningfully
when weighted sum > 10

Superscript corresponds to the layer

$$a_0^{(1)} = \sigma\left(w_{0,0}a_0^{(0)} + w_{0,1}a_1^{(0)} + \cdots + w_{0,n}a_n^{(0)} + b_0\right)$$

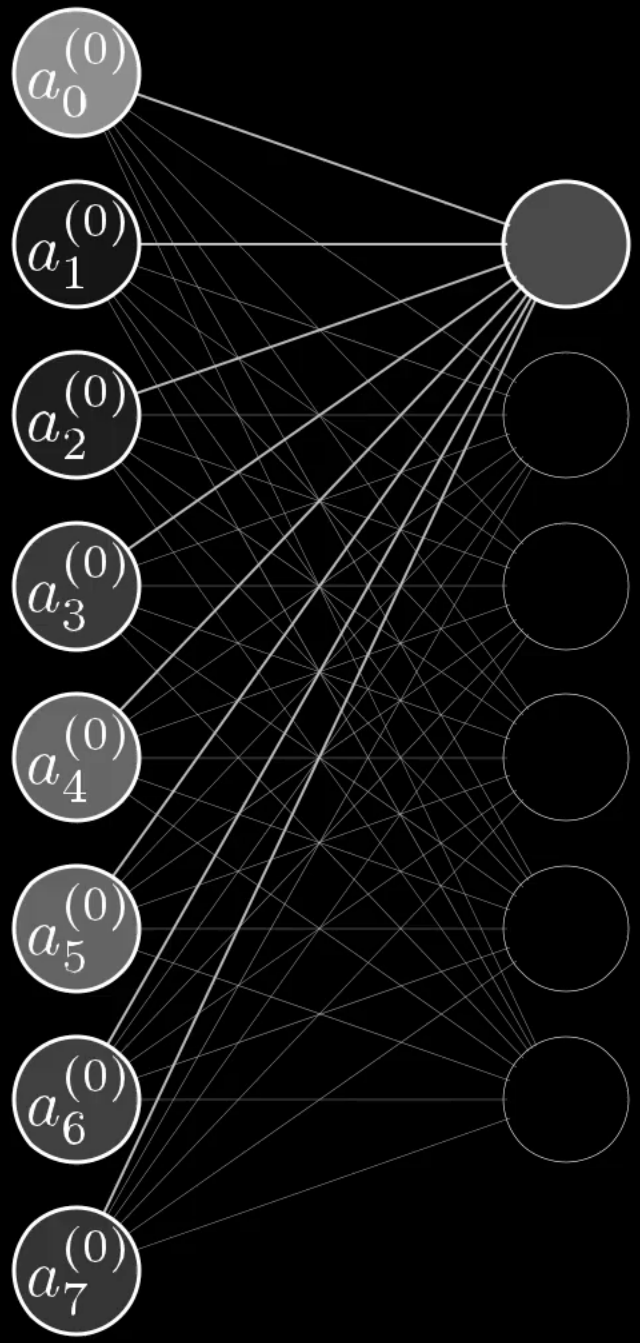
Subscript corresponds to a neuron in the layer



Sigmoid

$$a_0^{(1)} = \sigma\left(w_{0,0}\ a_0^{(0)} + w_{0,1}\ a_1^{(0)} + \cdots + w_{0,n}\ a_n^{(0)} + b_0 \right)$$

Bias

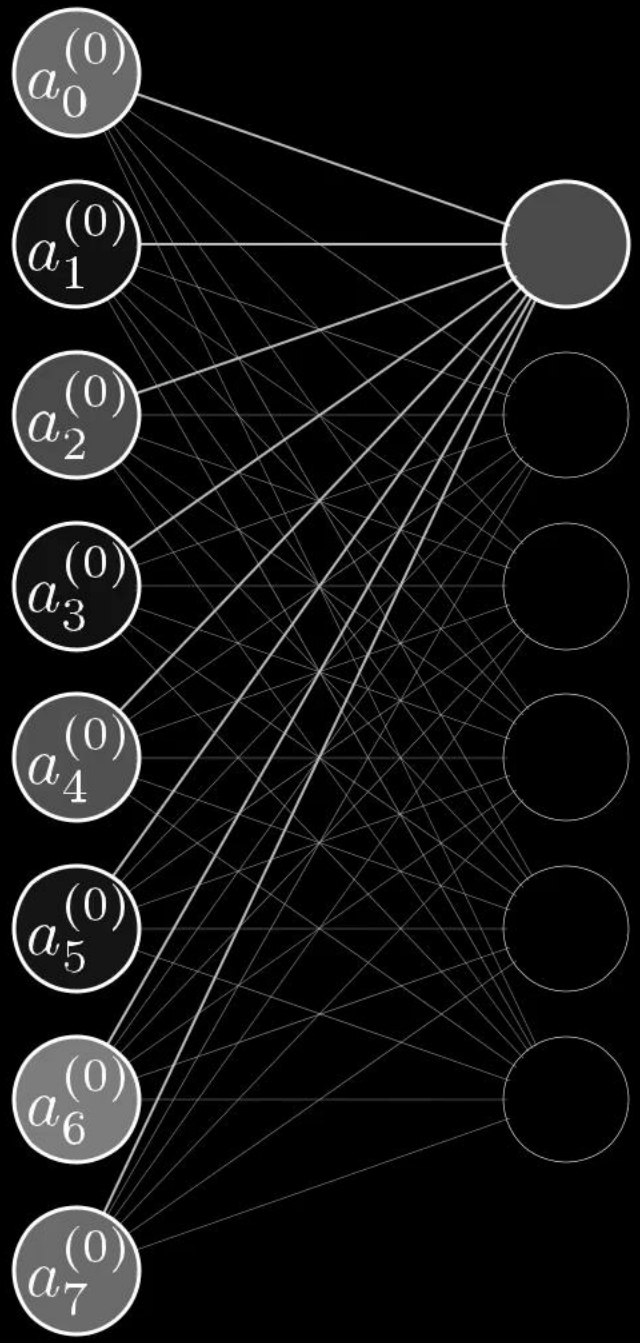


Sigmoid

$$a_0^{(1)} = \sigma\left(w_{0,0}\ a_0^{(0)} + w_{0,1}\ a_1^{(0)} + \cdots + w_{0,n}\ a_n^{(0)} + b_0 \right)$$

Bias

$$\begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \vdots \\ a_n^{(0)} \end{bmatrix}$$

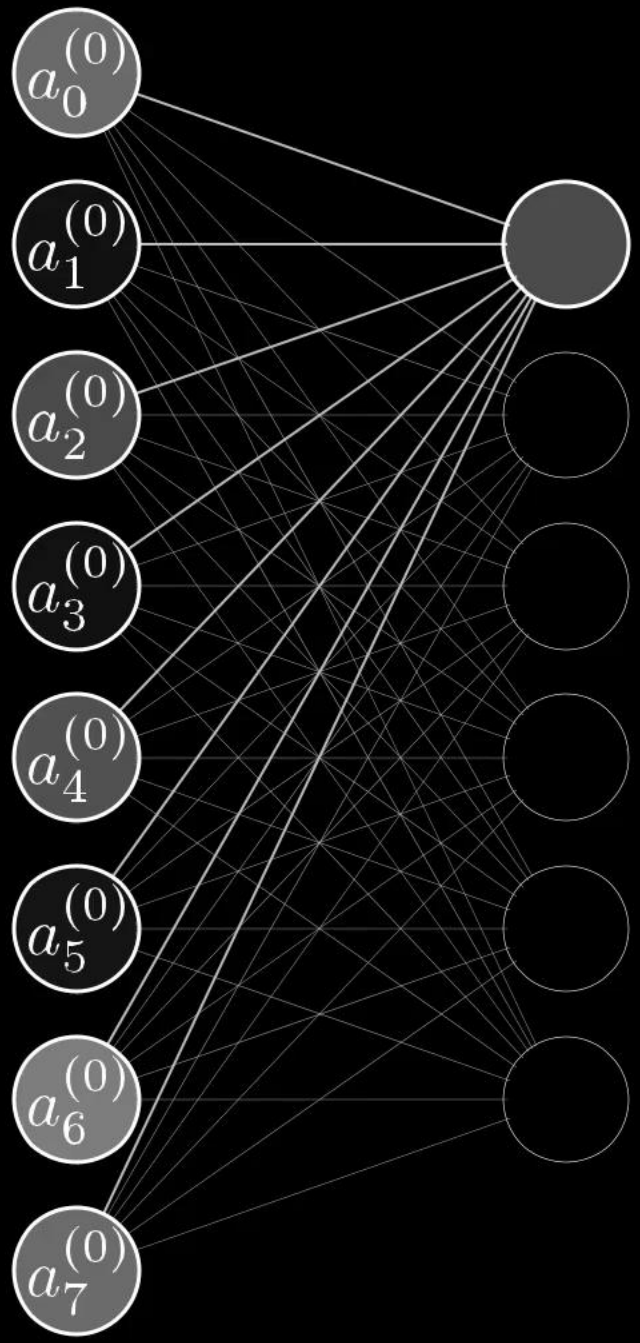


Sigmoid

$$a_0^{(1)} = \sigma\left(w_{0,0}\, a_0^{(0)} + w_{0,1}\, a_1^{(0)} + \cdots + w_{0,n}\, a_n^{(0)} + b_0 \right)$$

Bias

$$\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \vdots \\ a_n^{(0)} \end{bmatrix}$$

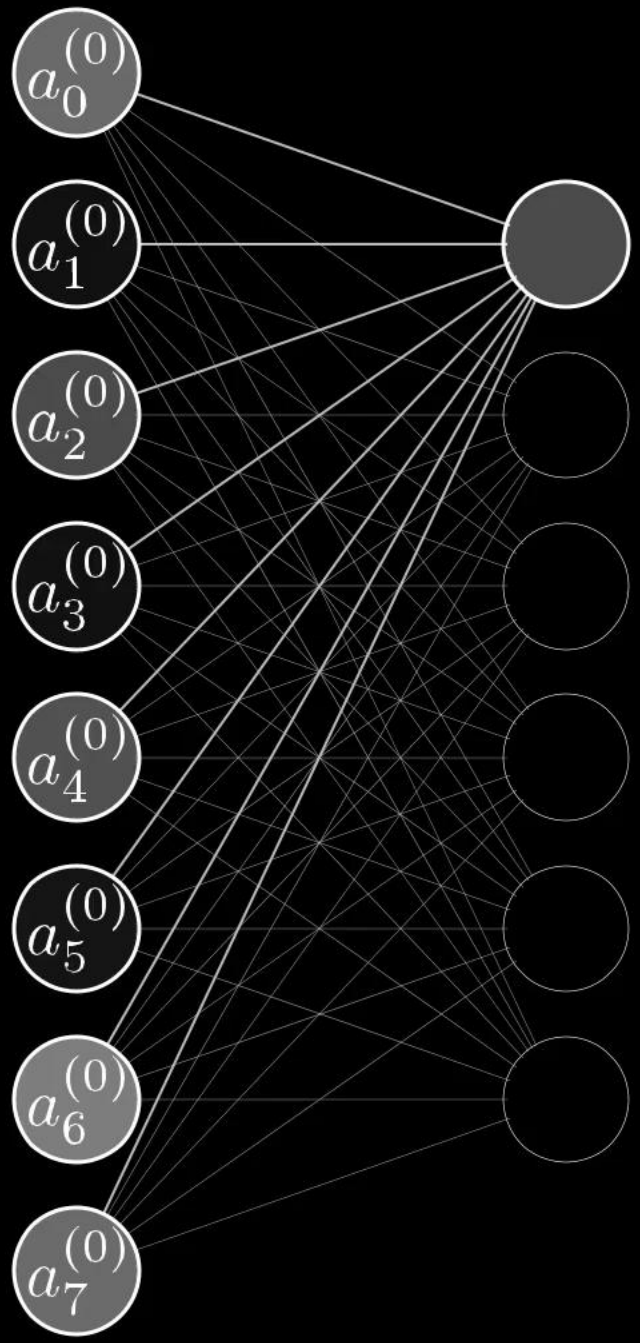


Sigmoid

$$a_0^{(1)} = \sigma\left(w_{0,0}\, a_0^{(0)} + w_{0,1}\, a_1^{(0)} + \cdots + w_{0,n}\, a_n^{(0)} + b_0 \right)$$

Bias

$$\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \vdots \\ a_n^{(0)} \end{bmatrix}$$

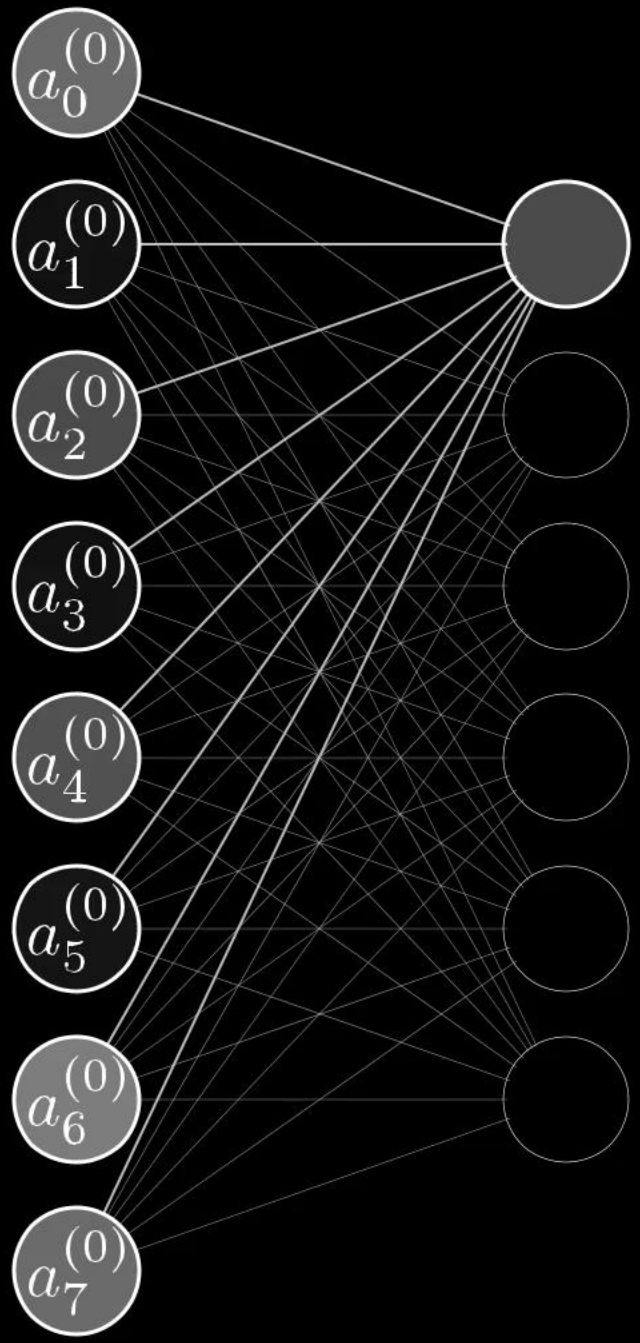


Sigmoid

$$a_0^{(1)} = \sigma\left(w_{0,0}\, a_0^{(0)} + w_{0,1}\, a_1^{(0)} + \cdots + w_{0,n}\, a_n^{(0)} + b_0 \right)$$

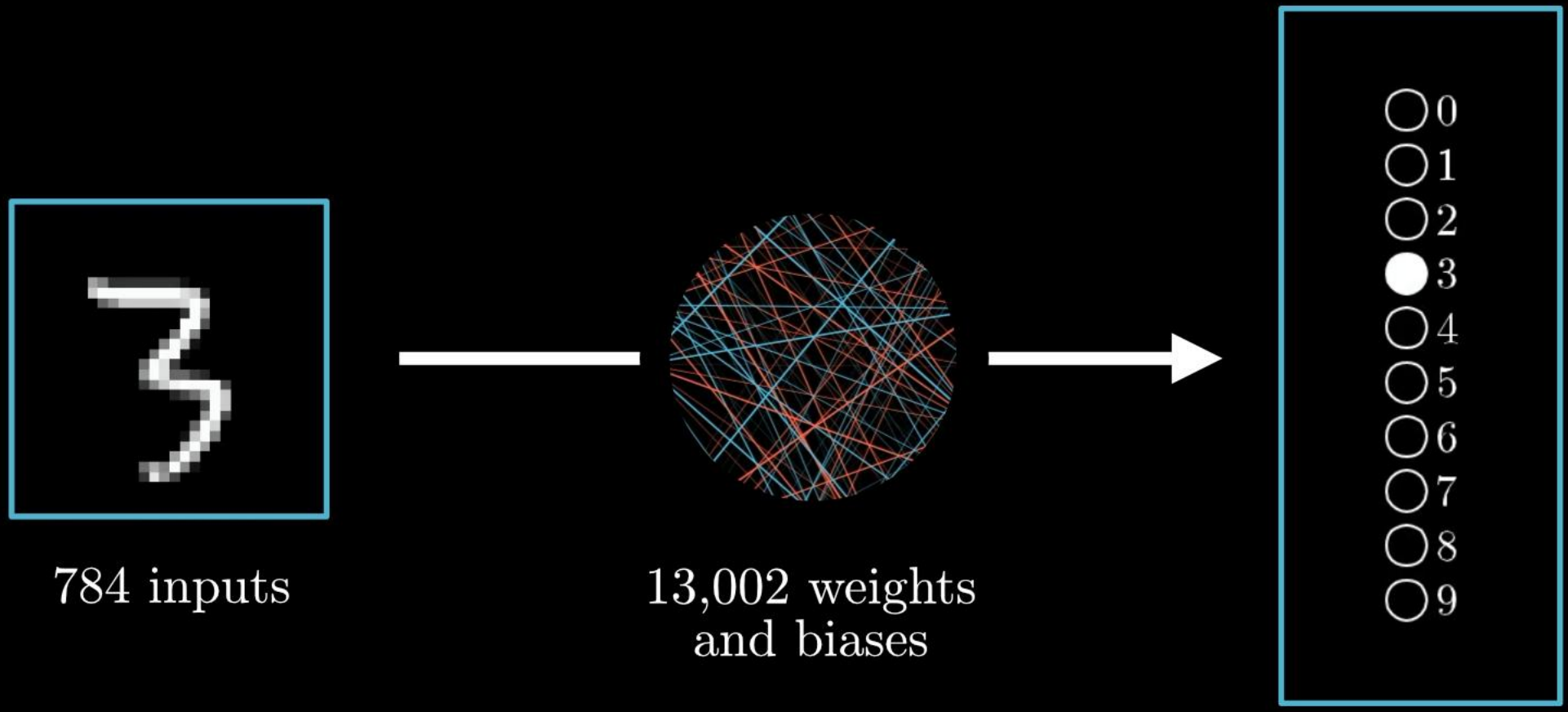
Bias

$$\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \vdots \\ a_n^{(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{bmatrix}$$



$$\sigma\left(\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \vdots \\ a_n^{(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{bmatrix}\right)$$

Neural network function



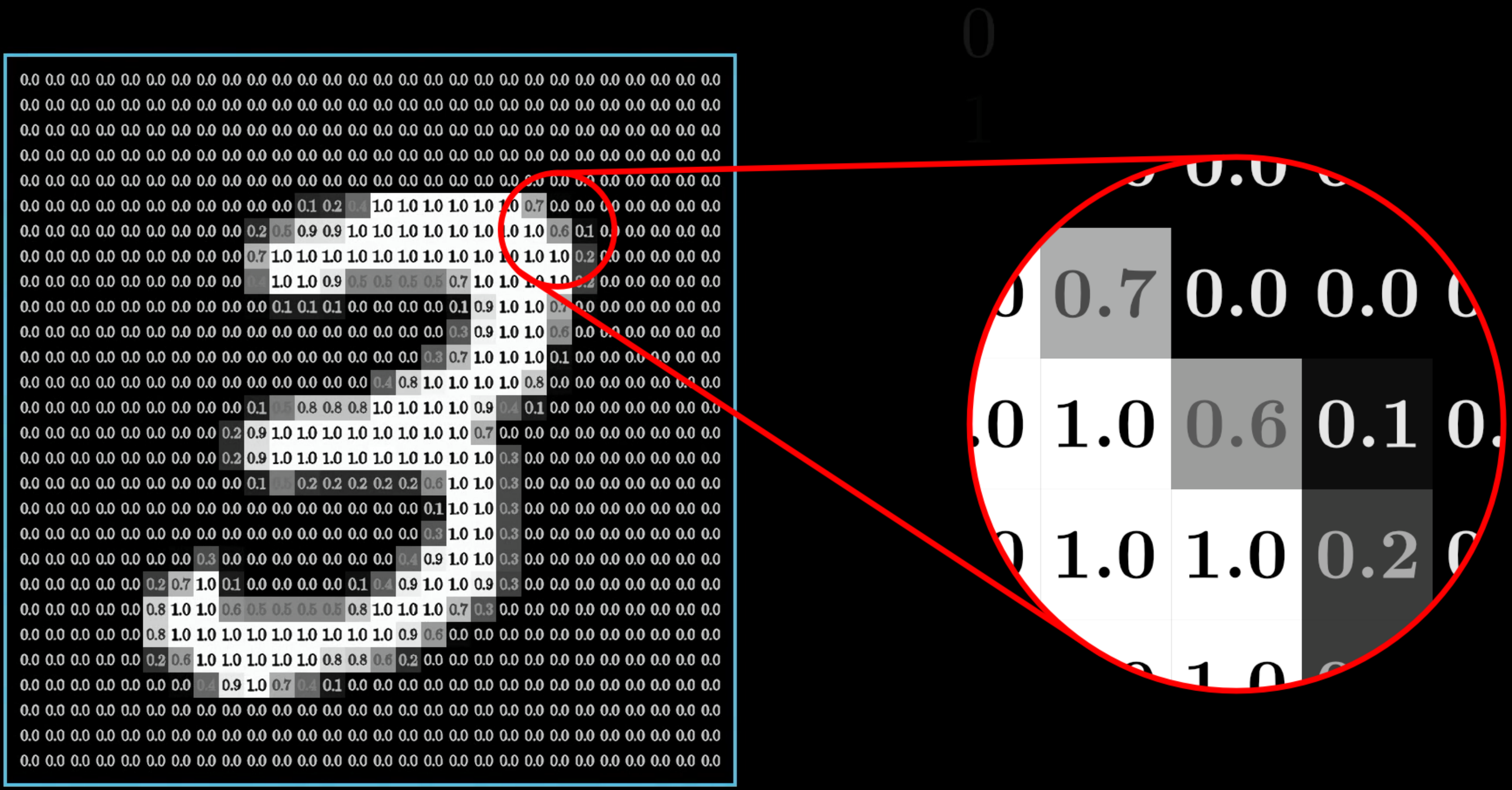
784 inputs

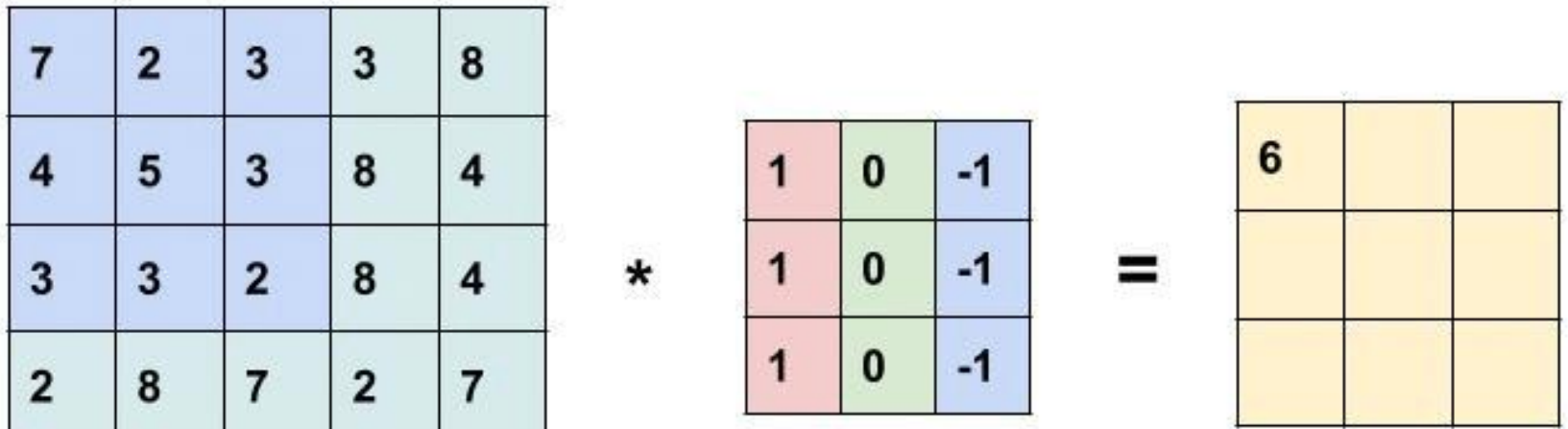
13,002 weights
and biases

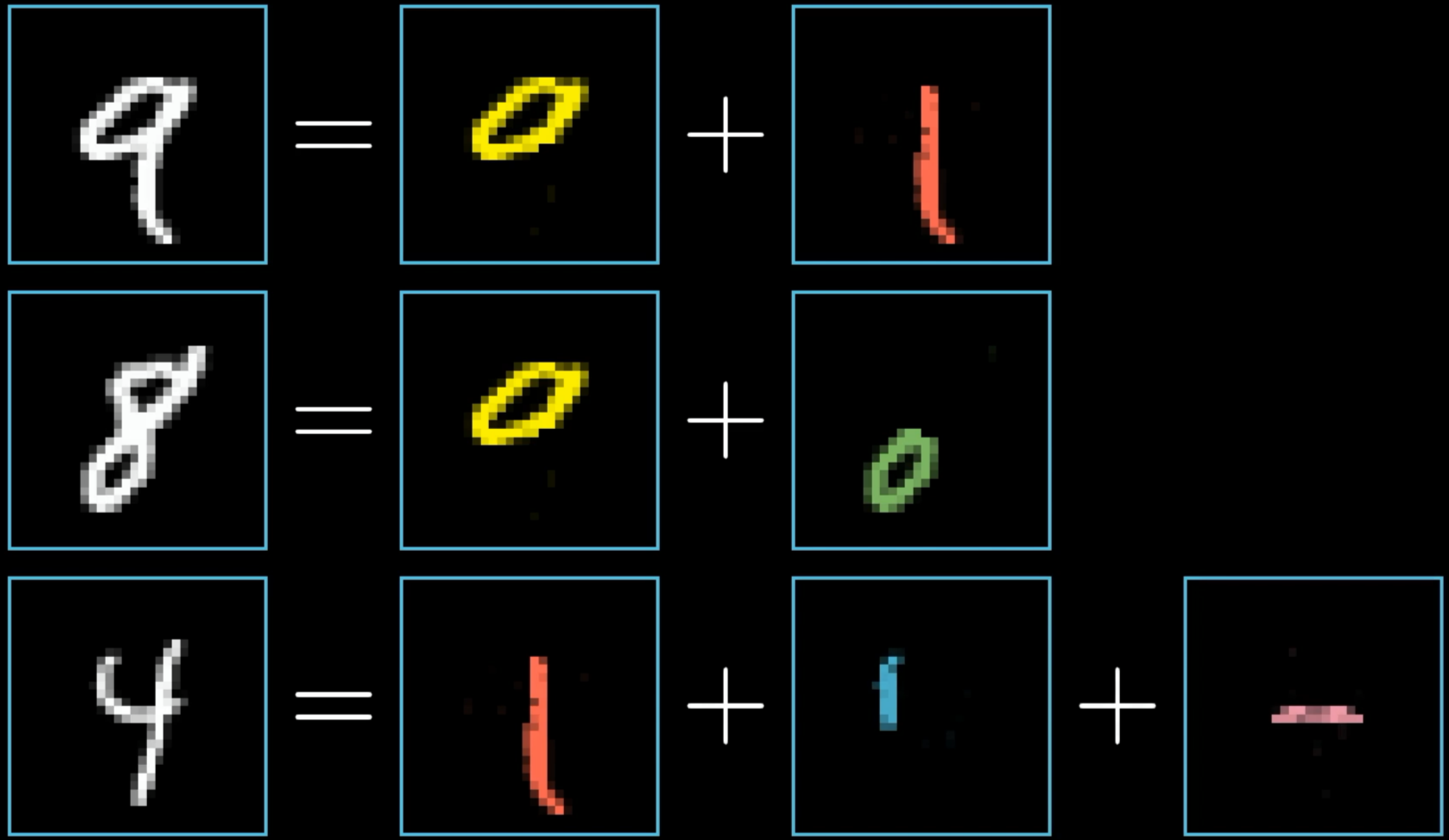
10 outputs

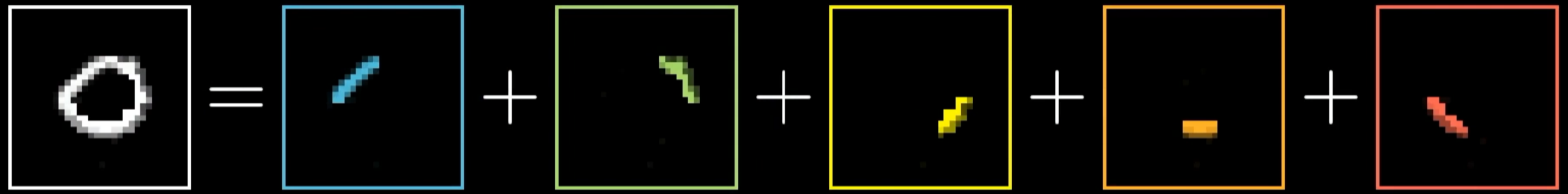
$$f * g$$

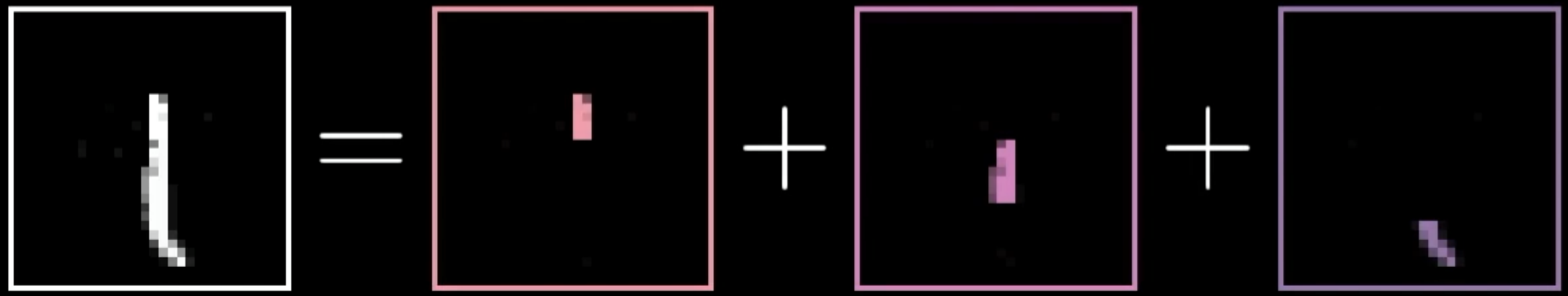
when you convolve two functions, you're basically combining them in such a way that tracks their interaction throughout time.


0.7 0.0 0.0
1.0 0.6 0.1
1.0 1.0 0.2


7 2 3 3 8
4 5 3 8 4
3 3 2 8 4
2 8 7 2 7
5 4 4 5 4
*
1 0 -1
1 0 -1
1 0 -1
=
6
7x1+4x1+3x1+
2x0+5x0+3x0+
3x-1+3x-1+2x-1
= 6

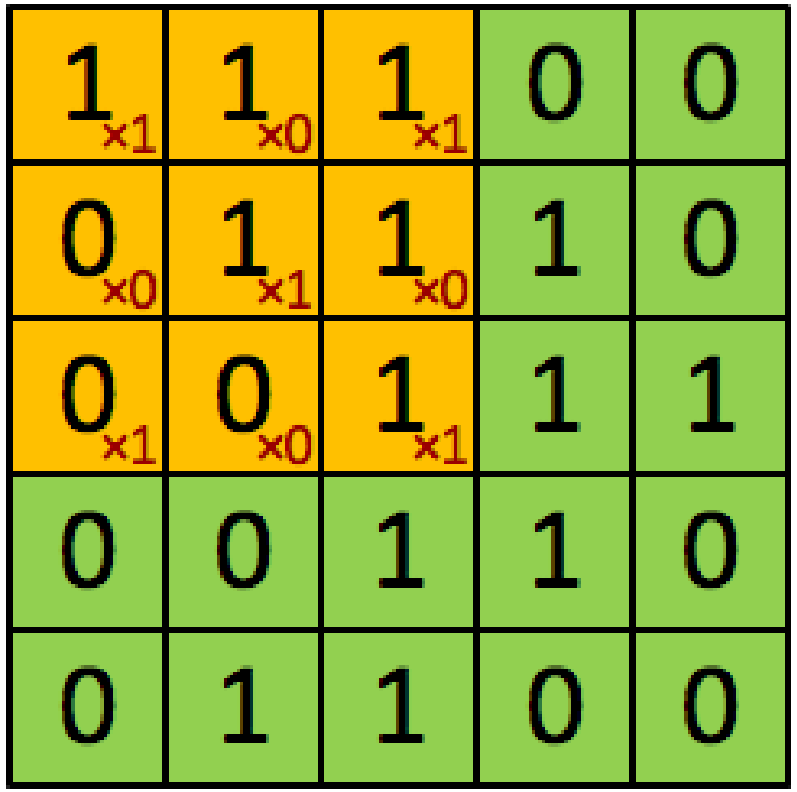




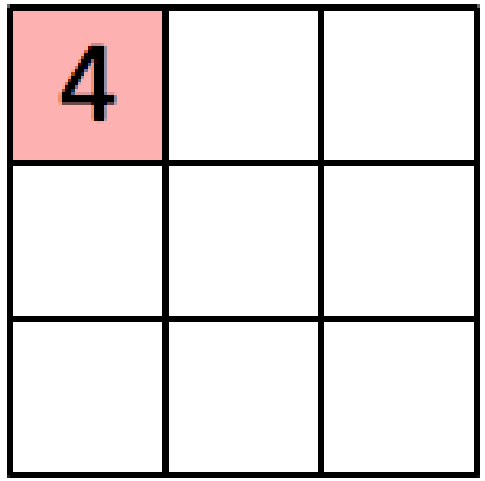


 → r e c o g n i t i o n → re·cog·ni·tion → recognition

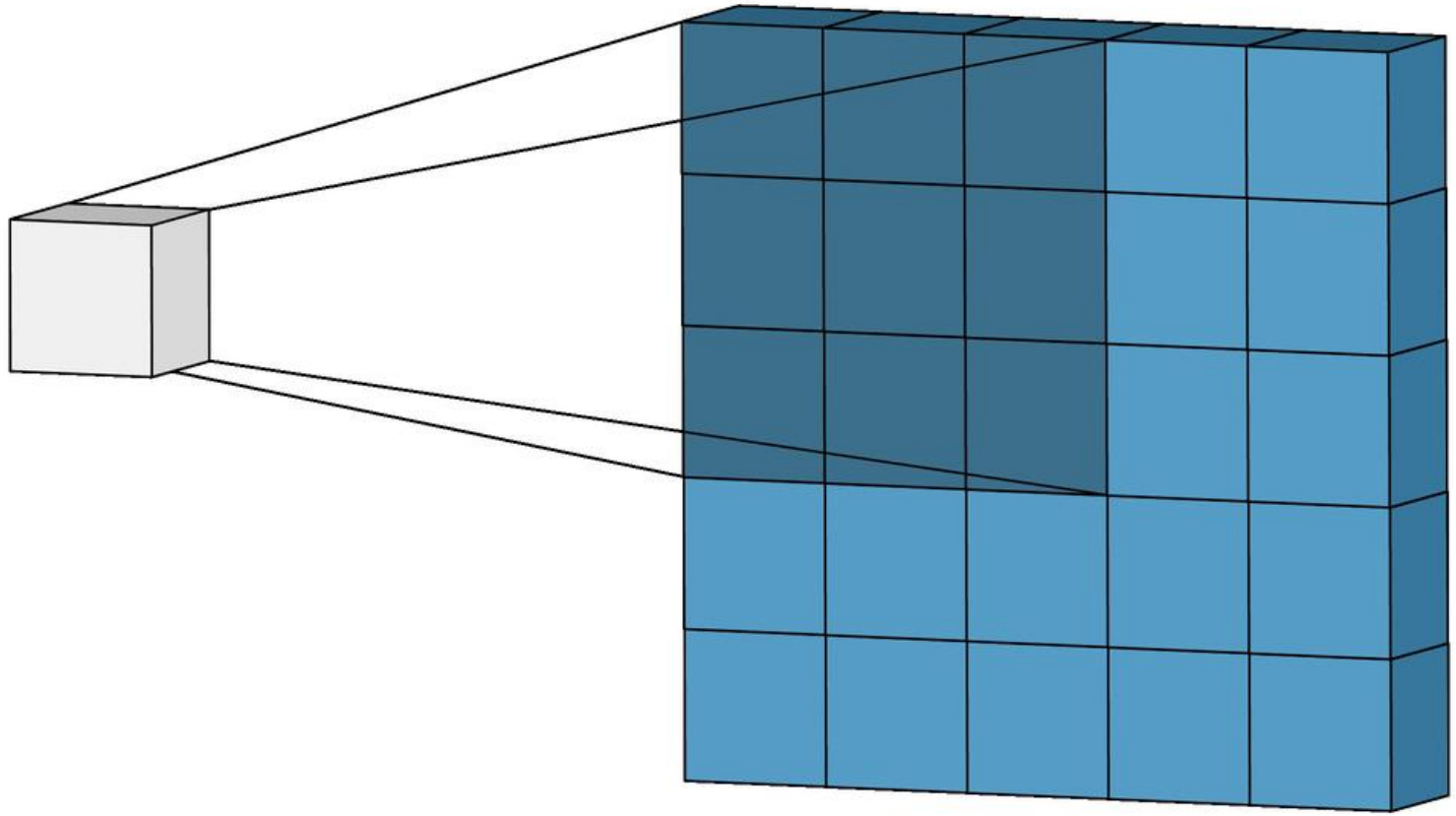
Raw audio

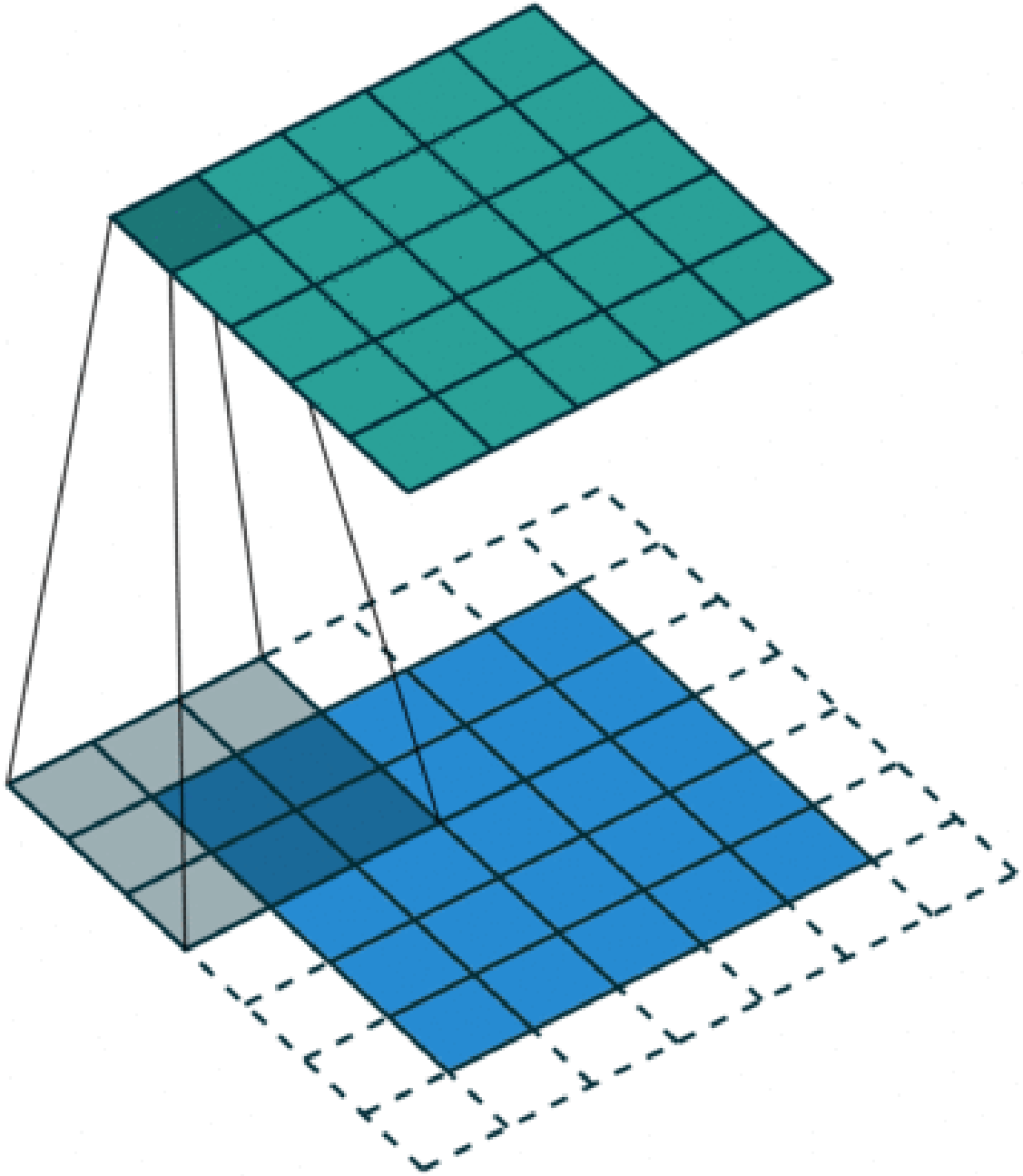


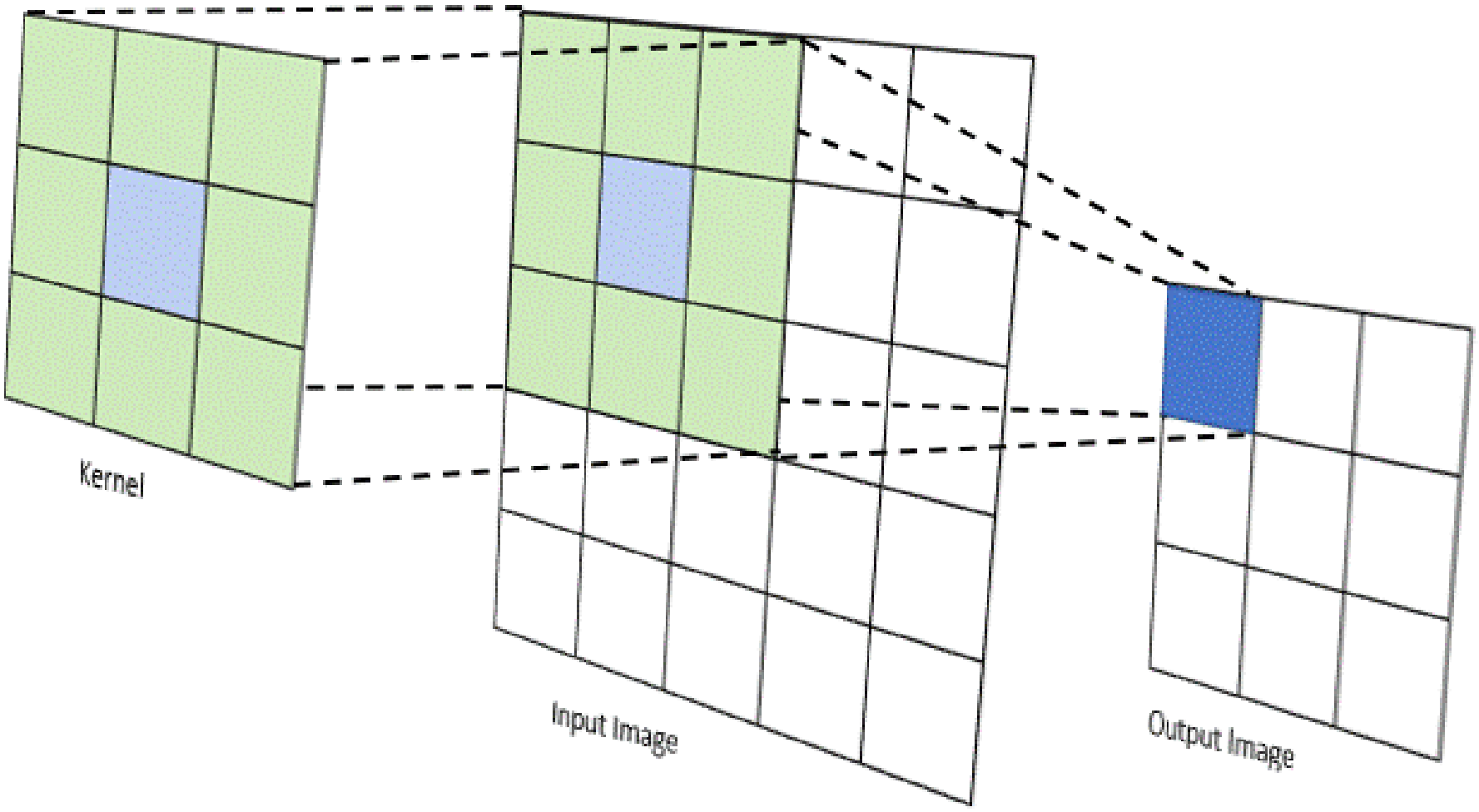
Image

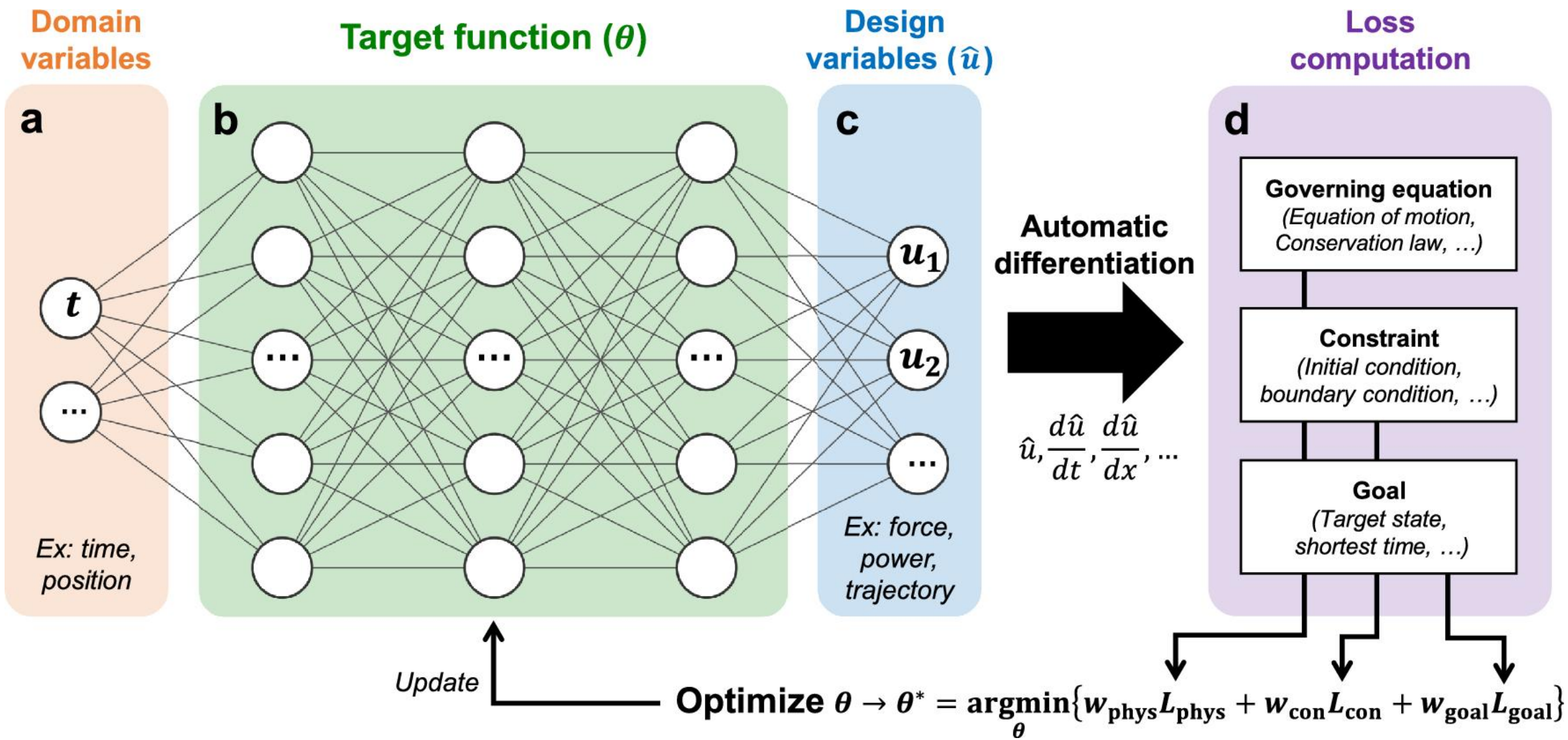


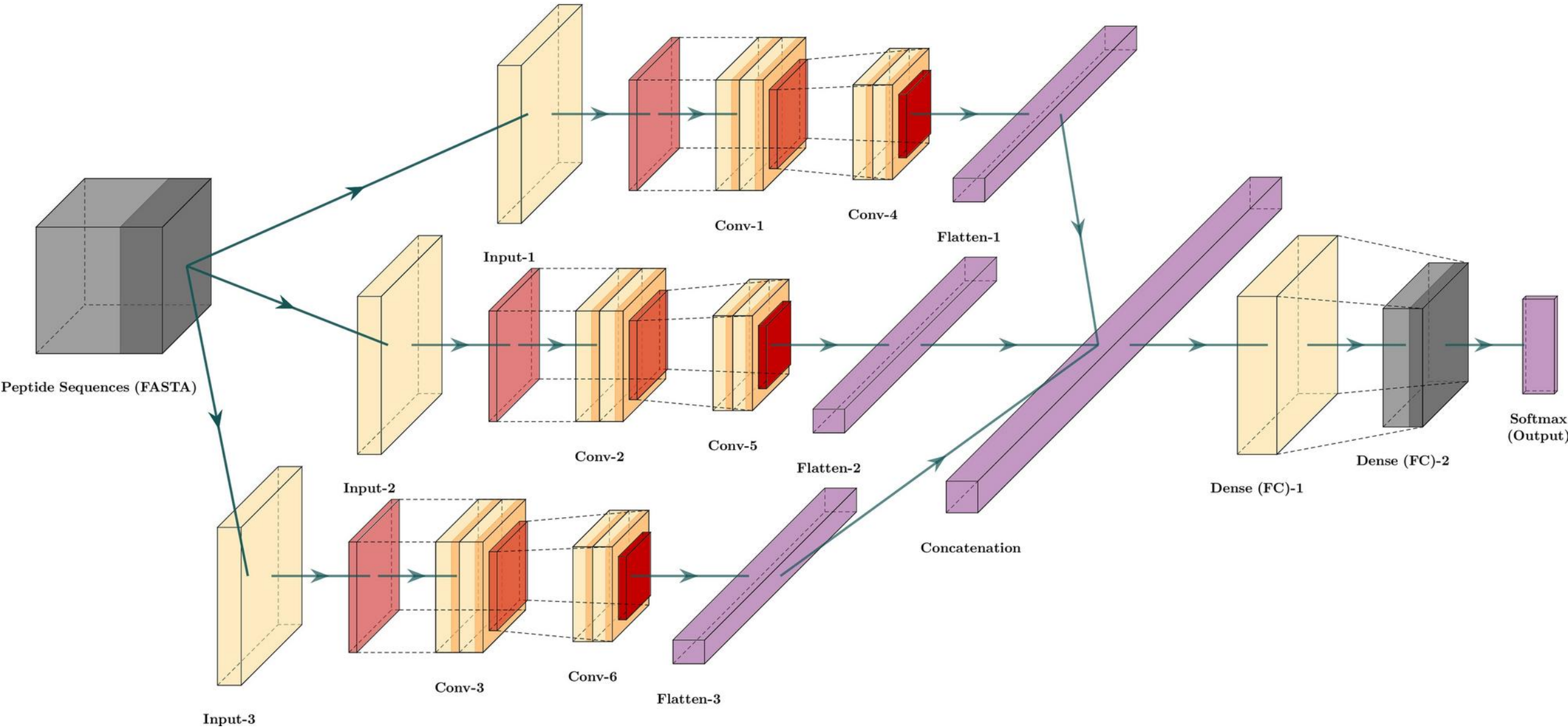
Convolved Feature

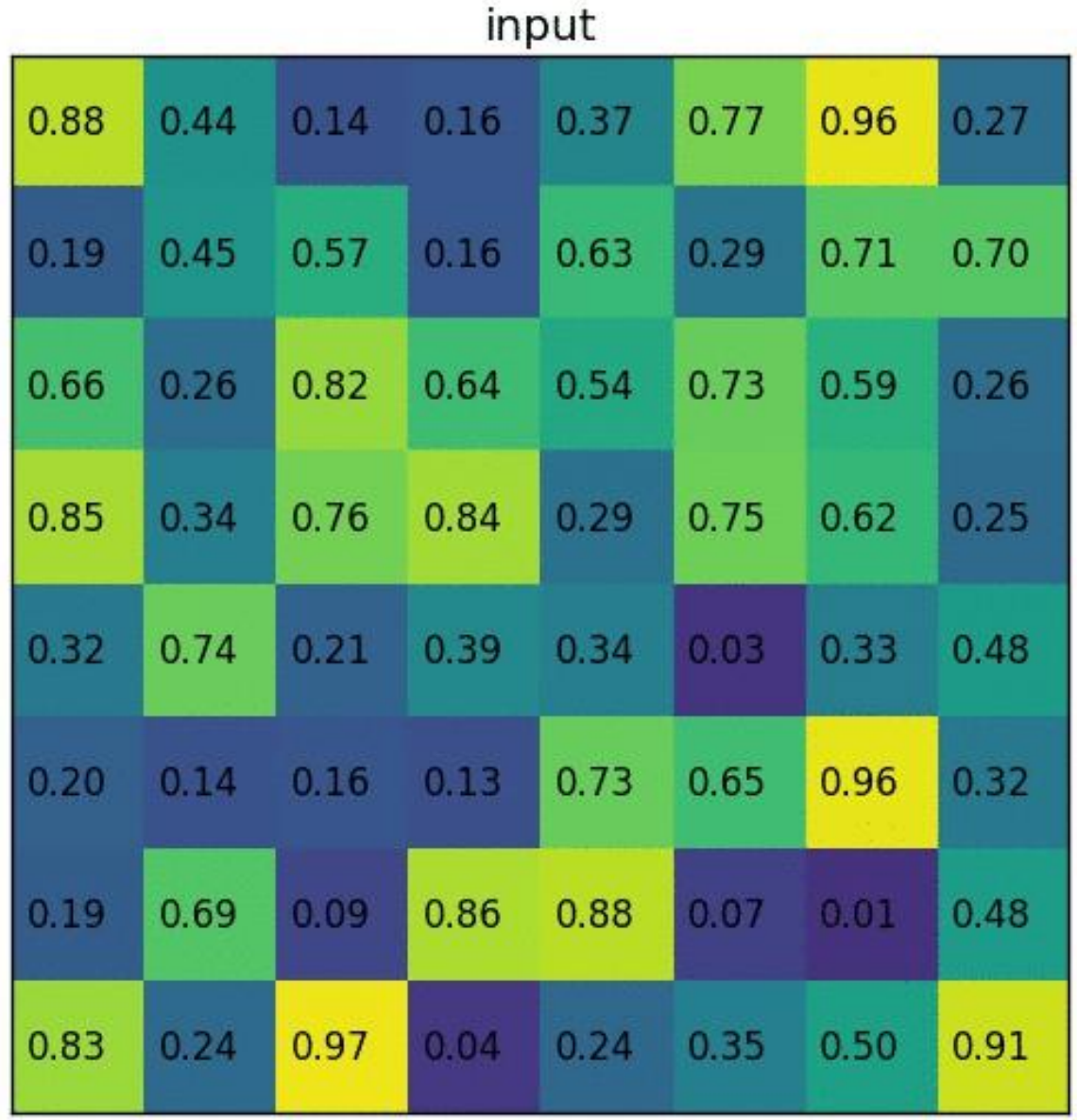


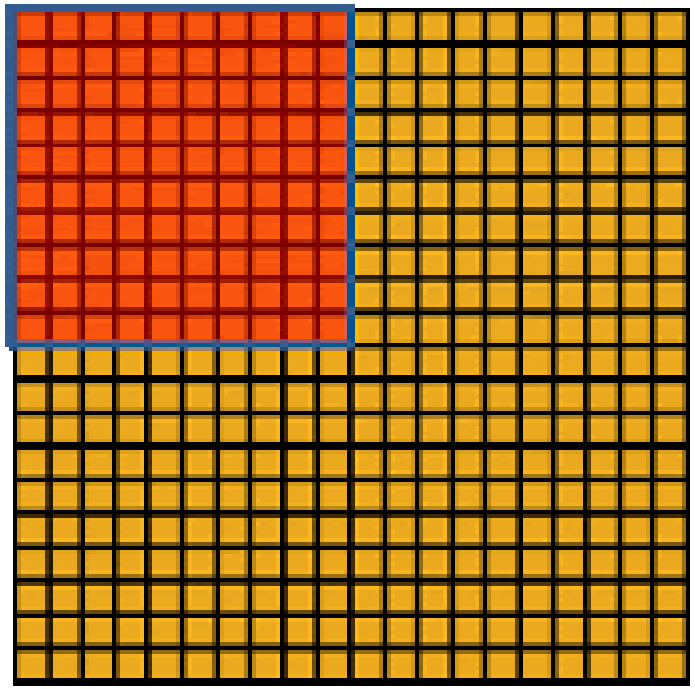



Kernel
Input Image
Output Image

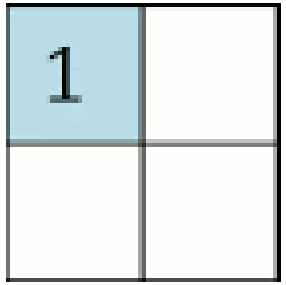

Domain variables
Target function (θ)
Design variables ($\hat{u}$)
Loss computation
a
b
c
d
t
...
Ex: time, position
u_1
u_2
...
Ex: force, power, trajectory
Automatic differentiation
$\hat{u}, \frac{d\hat{u}}{dt}, \frac{d\hat{u}}{dx}, \ldots$
Governing equation
(Equation of motion, Conservation law, …)
Constraint
(Initial condition, boundary condition, …)
Goal
(Target state, shortest time, …)
Update
Optimize $\theta \rightarrow \theta^* = \underset{\theta}{\mathrm{argmin}}\{w_{\mathrm{phys}}L_{\mathrm{phys}} + w_{\mathrm{con}}L_{\mathrm{con}} + w_{\mathrm{goal}}L_{\mathrm{goal}}\}$


Peptide Sequences (FASTA)
Input-1
Conv-1
Conv-4
Flatten-1
Input-2
Conv-2
Conv-5
Flatten-2
Input-3
Conv-3
Conv-6
Flatten-3
Concatenation
Dense (FC)-1
Dense (FC)-2
Softmax
(Output)


input
0.88 0.44 0.14 0.16 0.37 0.77 0.96 0.27
0.19 0.45 0.57 0.16 0.63 0.29 0.71 0.70
0.66 0.26 0.82 0.64 0.54 0.73 0.59 0.26
0.85 0.34 0.76 0.84 0.29 0.75 0.62 0.25
0.32 0.74 0.21 0.39 0.34 0.03 0.33 0.48
0.20 0.14 0.16 0.13 0.73 0.65 0.96 0.32
0.19 0.69 0.09 0.86 0.88 0.07 0.01 0.48
0.83 0.24 0.97 0.04 0.24 0.35 0.50 0.91



Convolved feature



Pooled feature

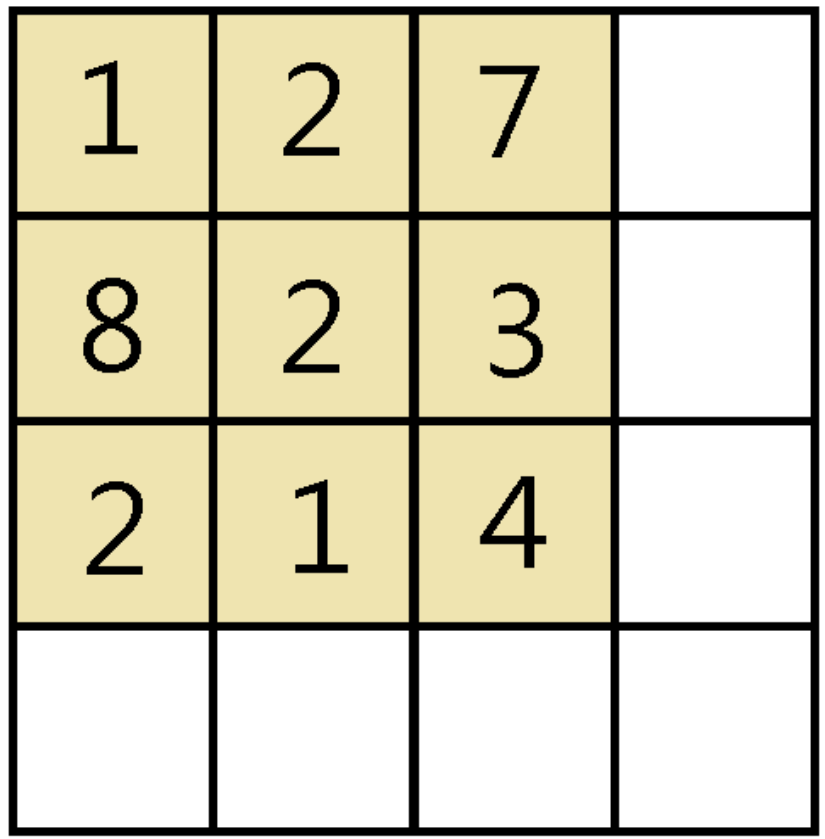
Feature Map

6	6	6	6
4	5	5	4
2	4	4	2
2	4	4	2

Max Pooling

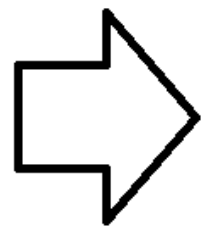
Average Pooling

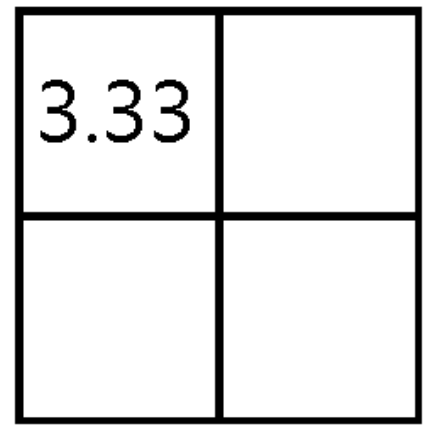
Sum Pooling



Forward

Propagation



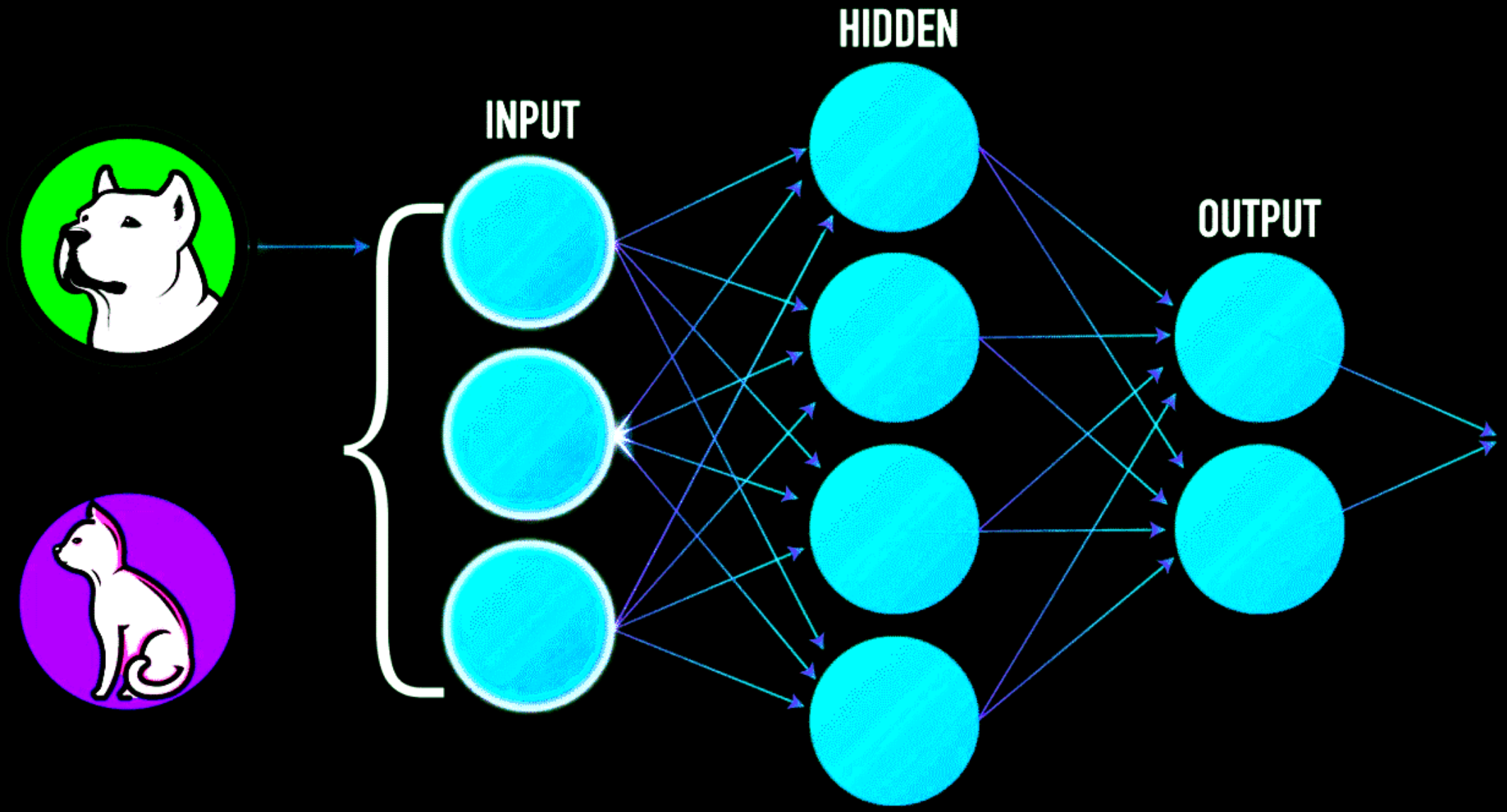


2	4	8	3	6
+1 9	3	4	2	5
5	4	+0 0 6	3	1
2	3	1	3	4
2	7	4	5	7

Backward Propagation

+1

1 9	3 8	1 8
1 9	4 6	2 6
6 7	2 7	1 7


HIDDEN
INPUT
OUTPUT