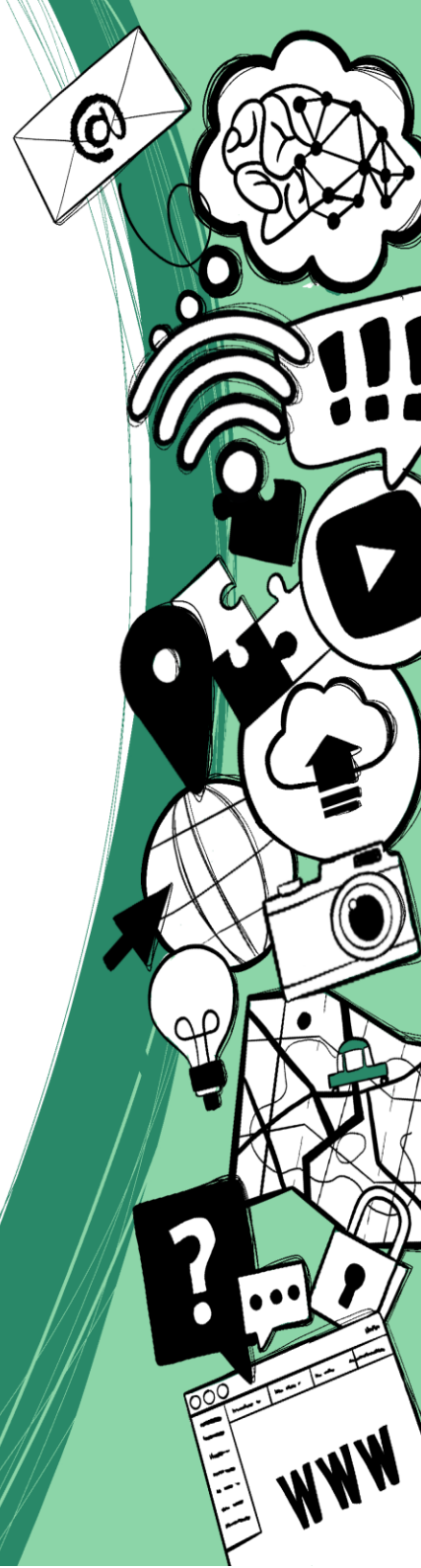
ENARIS

Neural Networks I
Basics



mobilis
interaktív kiállítási központ

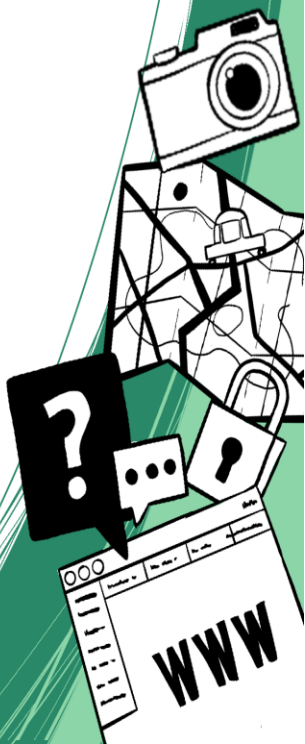
TU Graz

OESTERREICHISCHE COMPUTER GESELLSCHAFT
AUSTRIAN COMPUTER SOCIETY

Interreg
Austria-Hungary 2014-2020
European Union – European Regional Development Fund

The basis of thought processes

Presumably, all (thought) processes in our brain are based on the activation of neurons that are interconnected to form networks.
But these neurons aren't just wired together in a disorderly fashion, they're structured in different ways depending on the brain region.



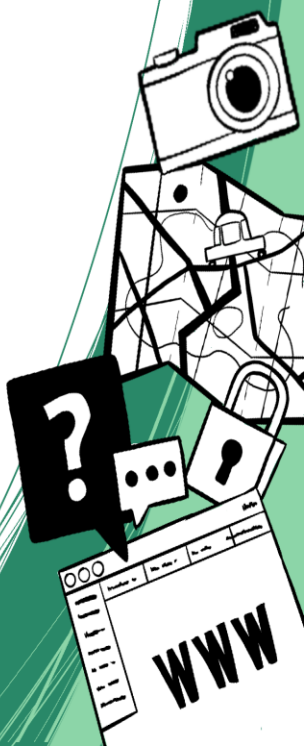
Artificial Neural Networks

Artificial neural networks try to replicate

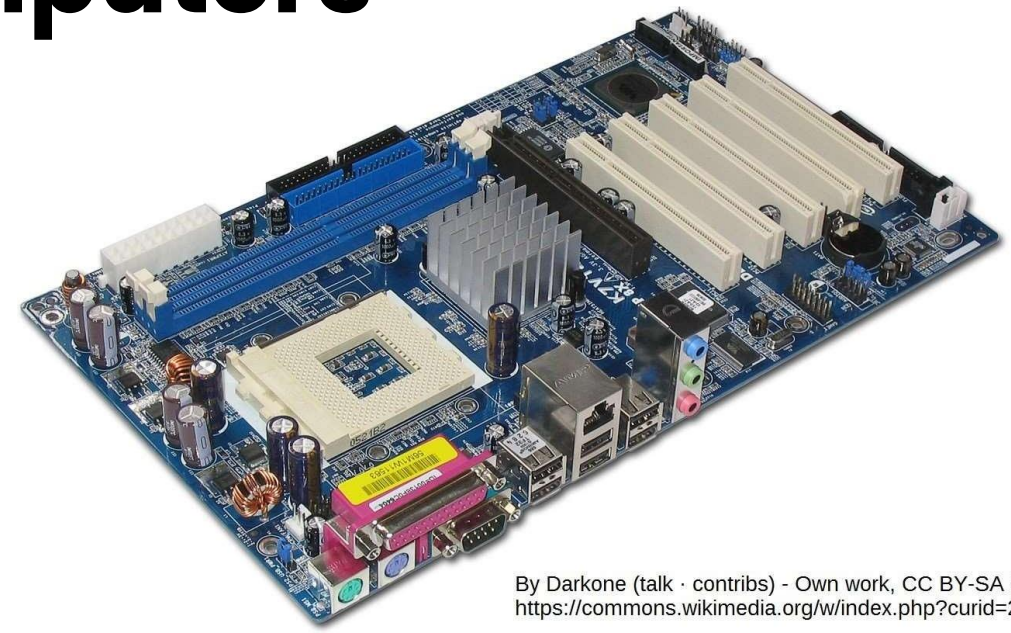
- the properties of neurons
- the structure of their arrangement (topology)
- learning processes

- What is a neural network?
- What is a neuron?
- What is the difference to a conventional computer (von Neumann architecture)?

We do not want to go into details about the learning process (backpropagation).

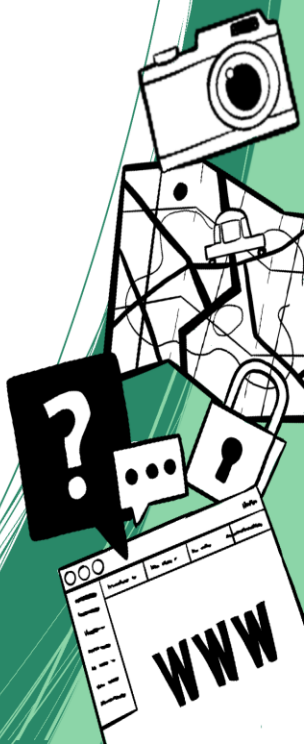


Neural Networks vs Conventional Computers



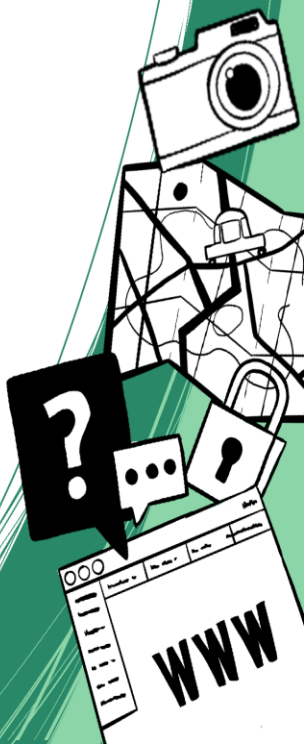
By Darkone (talk · contribs) - Own work, CC BY-SA 2.5, https://commons.wikimedia.org/w/index.php?curid=235545

In contrast to a conventional computer (picture above), there is no central processing unit in the brain but a large number of small processing units that work together to achieve thought processes. The individual small computing units only perform relatively small tasks, but many of them work in parallel.



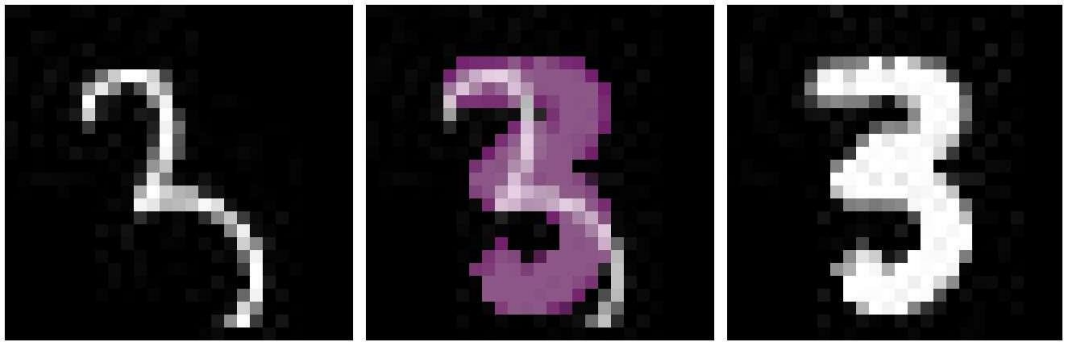
Feed-Forward-Netzwerk

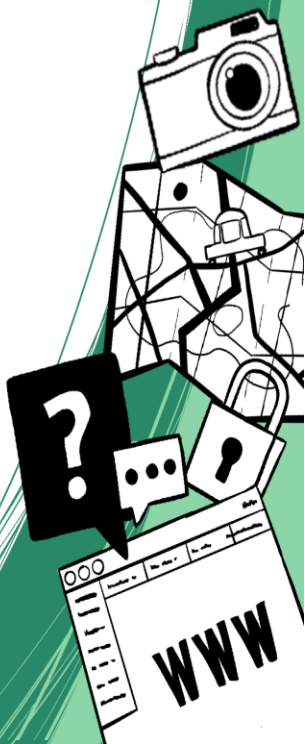
In our first example we will see how an artificial neural network solves a graphical task. There are separate network types for this, but we will use a simple socalled feed-forward network. This also works, but a more suitable network type would increase the recognition accuracy.



Digit and character recognition with a neural network

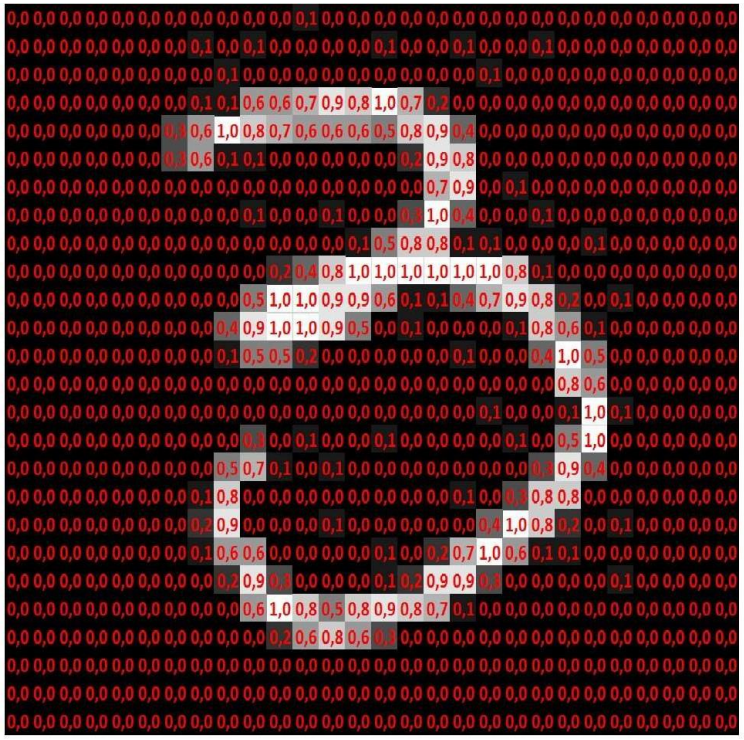
Digit and character recognition is a demanding task.
The task is difficult to solve with conventional programming methods.
All spellings for "3" should be recognized, at first glance different spellings may not have much in common.



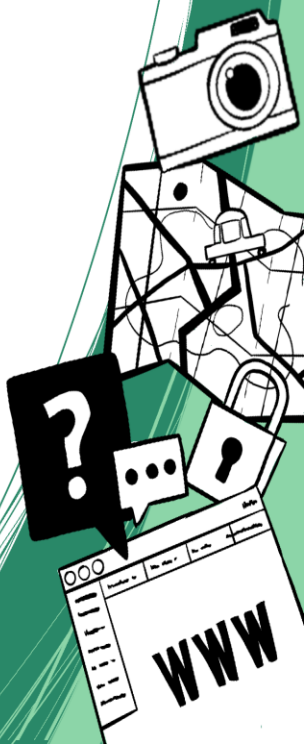


Activation

We proceed in such a way that we interpret the individual pixel brightnesses as activation variables and ask which activation pattern should be recognized as which number.

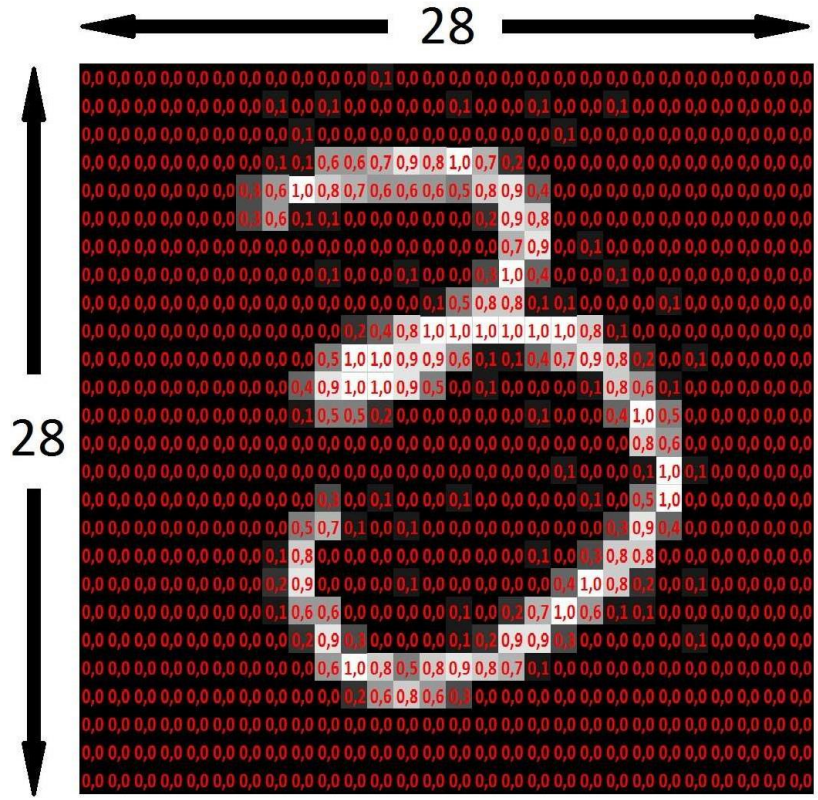


0
1
2
3 ?
4
5
6
7
8
9

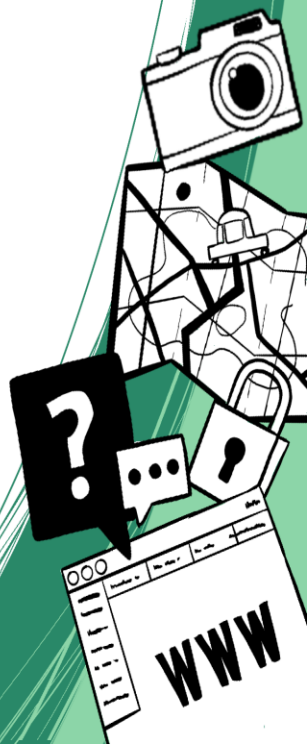


784 Pixels

The size of our images is 28 x 28 = 784 pixels.

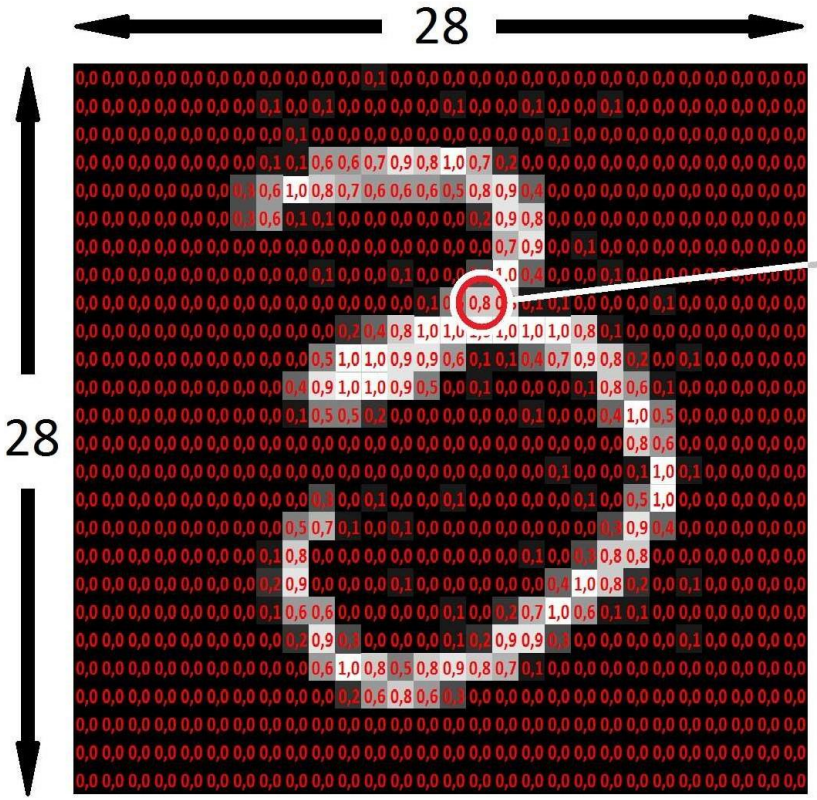


28 * 28 = 784

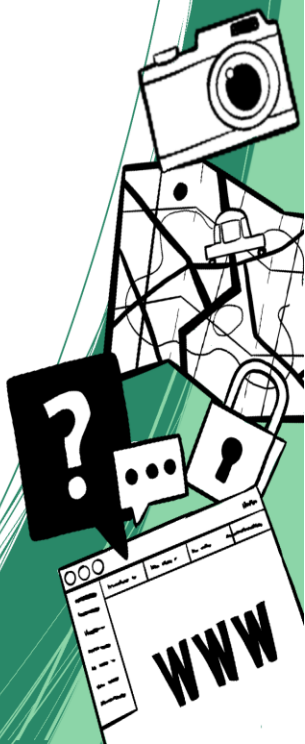


Activation

As activation we use a value between 0 and 1.

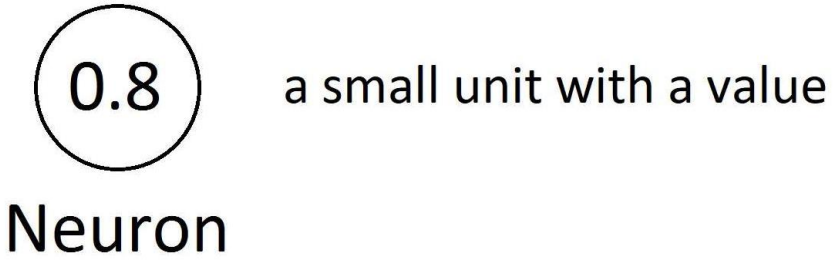


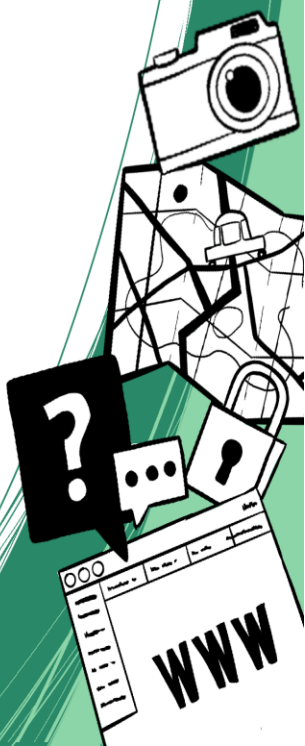
28 * 28 = 784



Neuron

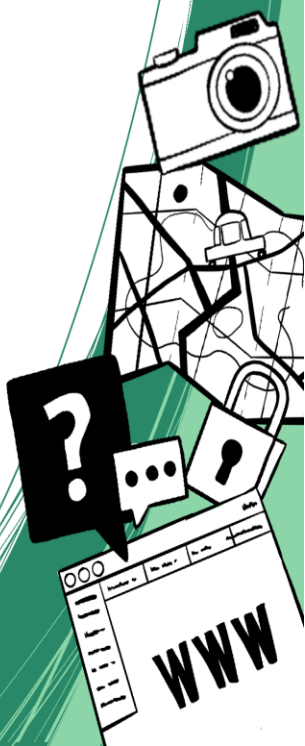
A neuron is a small unit with a value.



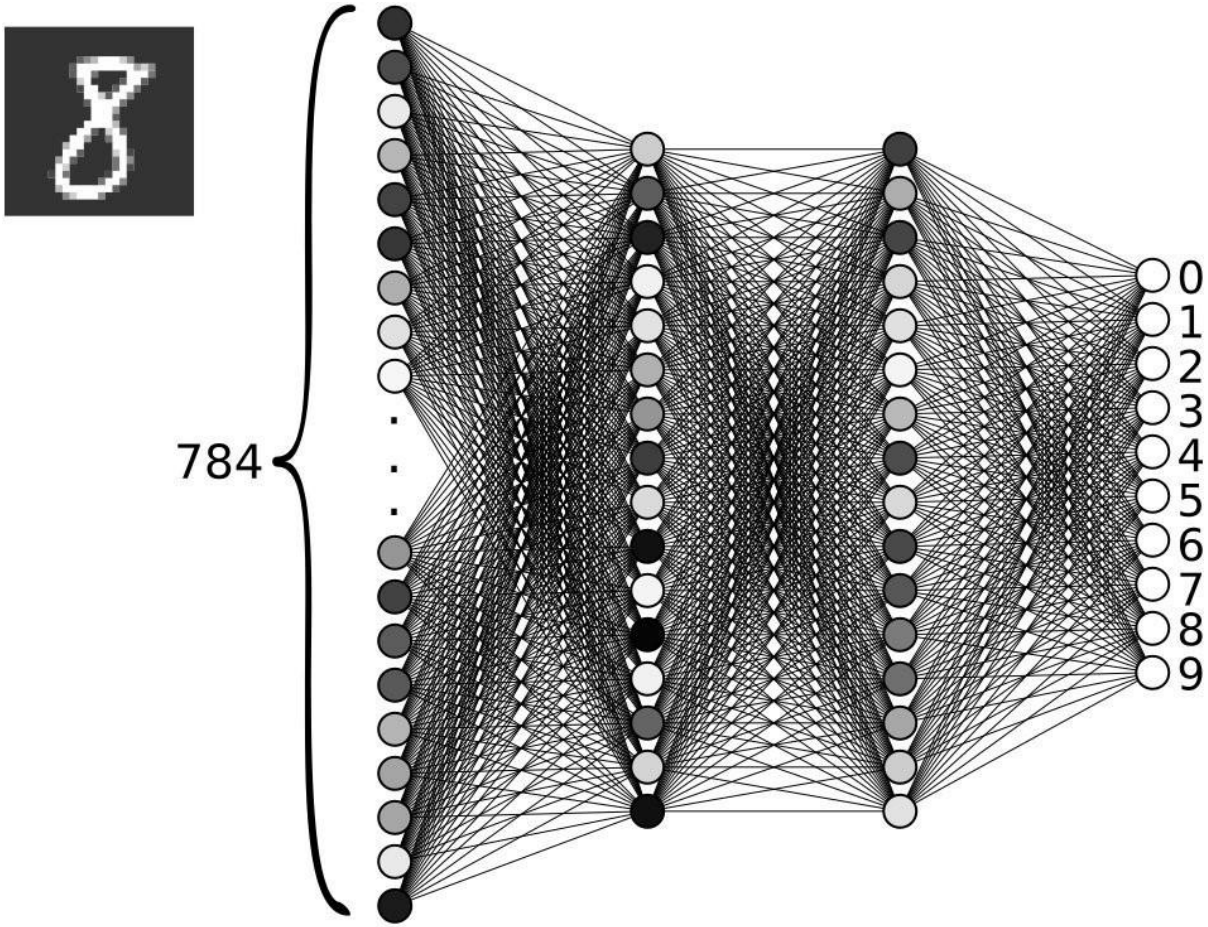


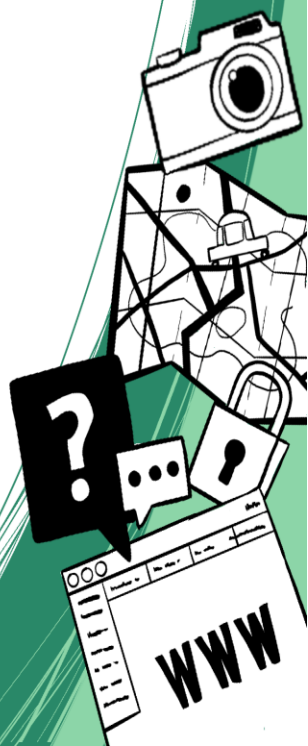
Network Structure

In the input layer, the inputs are made available as activations.
Within the network, the values represent activation states of the individual neurons.
The number of input neurons results from the number of pixels.
Spatial information is lost.
The number of neurons in the second and third layer can be chosen freely and was chosen at 16 quite arbitrarily.
The number of output neurons corresponds to the number of possible outcomes.



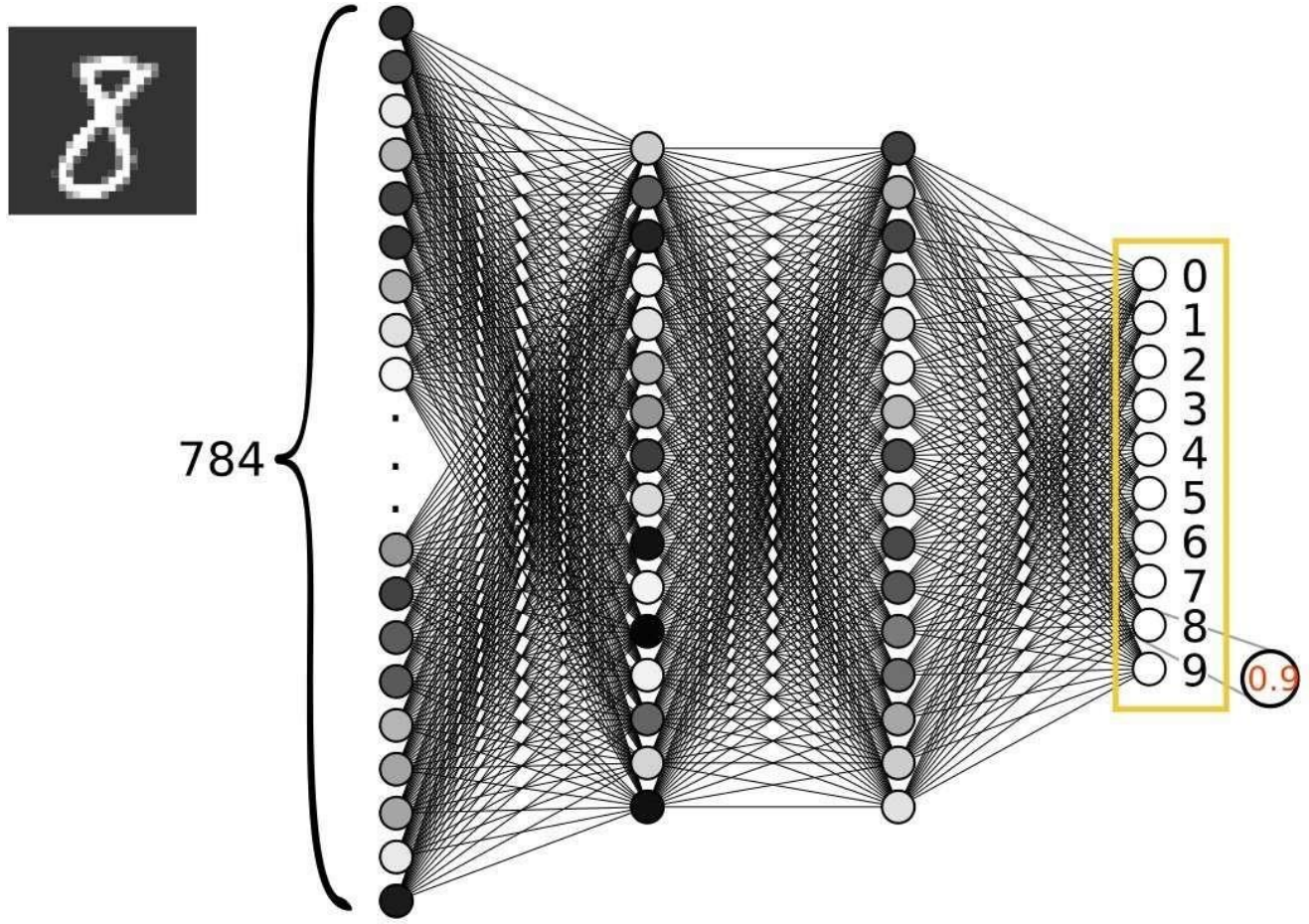
Layers

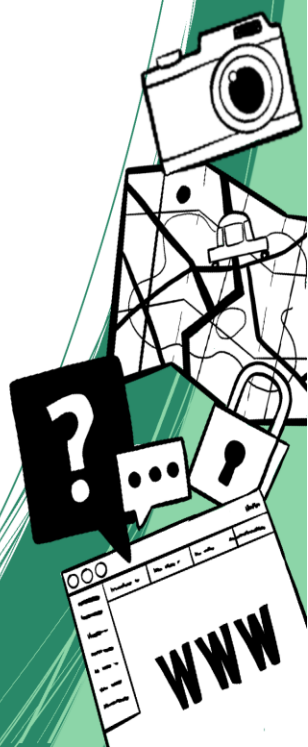




Output Layer

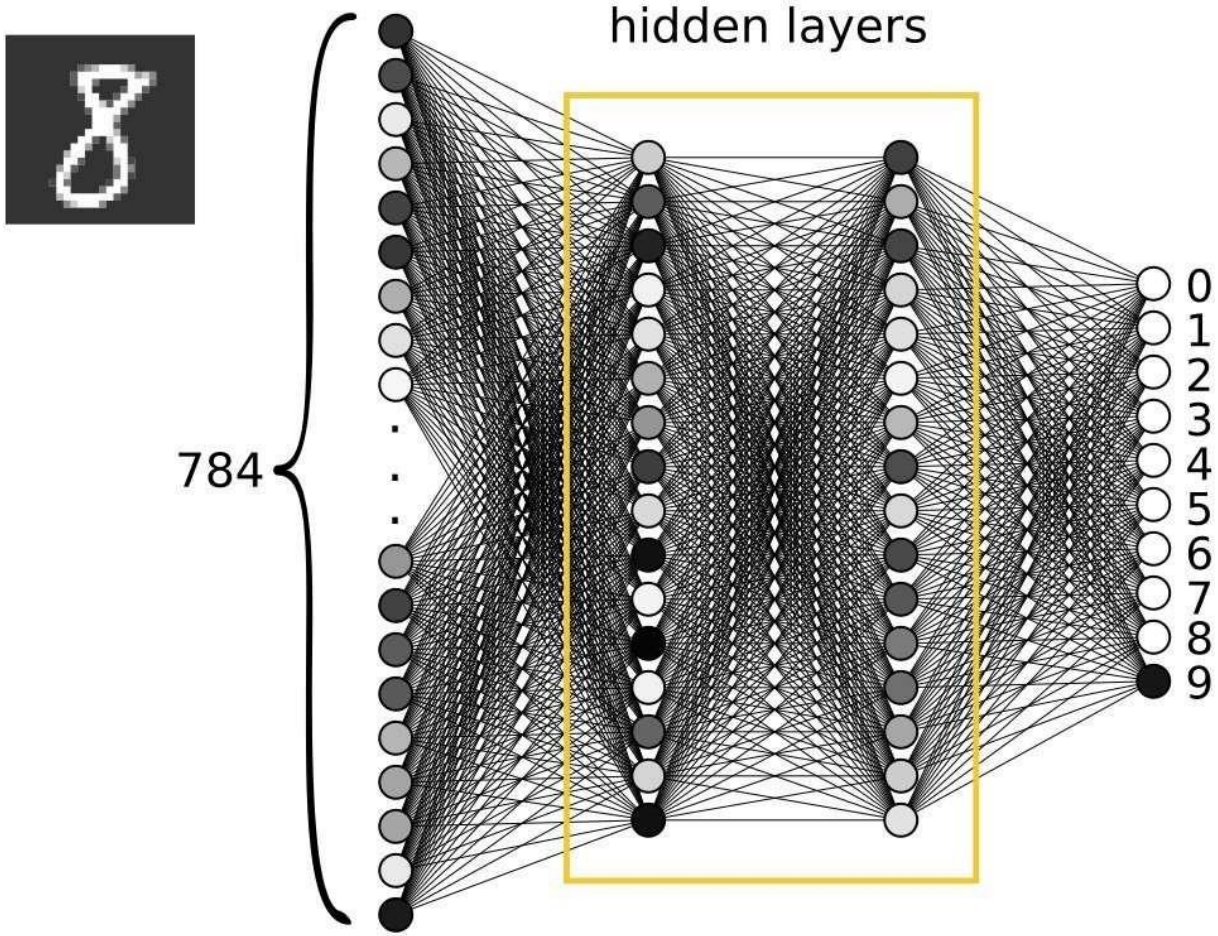
Result values are provided as activations in the output layer.

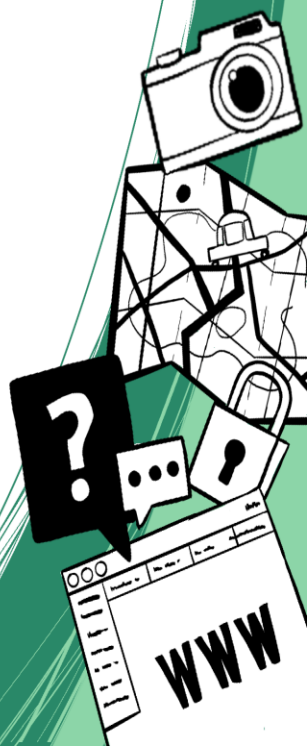




Hidden Layers

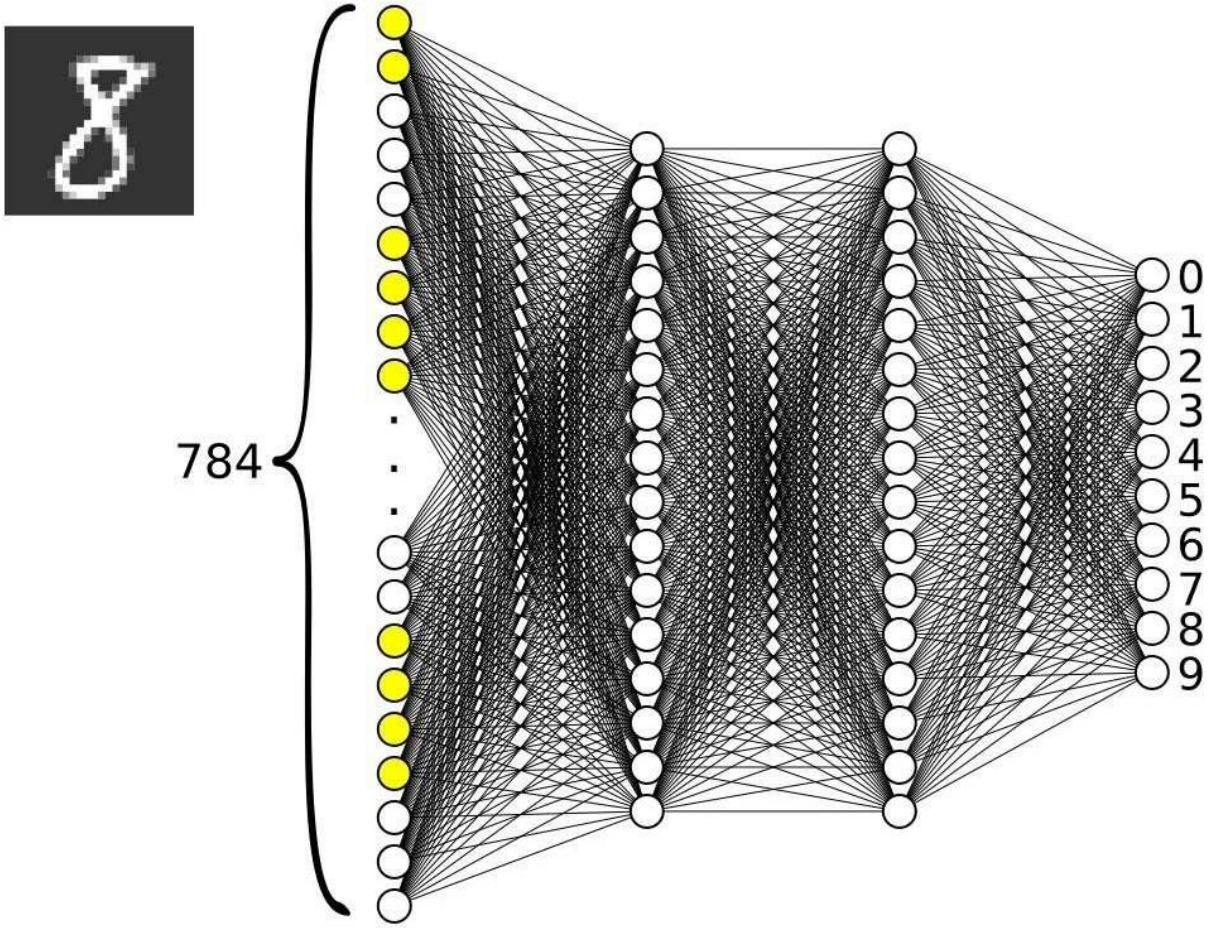
The layers between the input and output layers are called hidden layers.

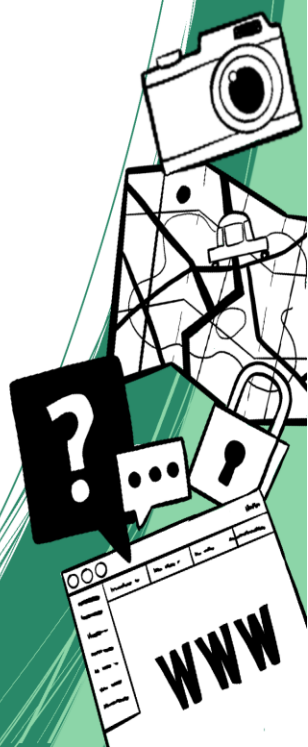




Flow of Data through the Network

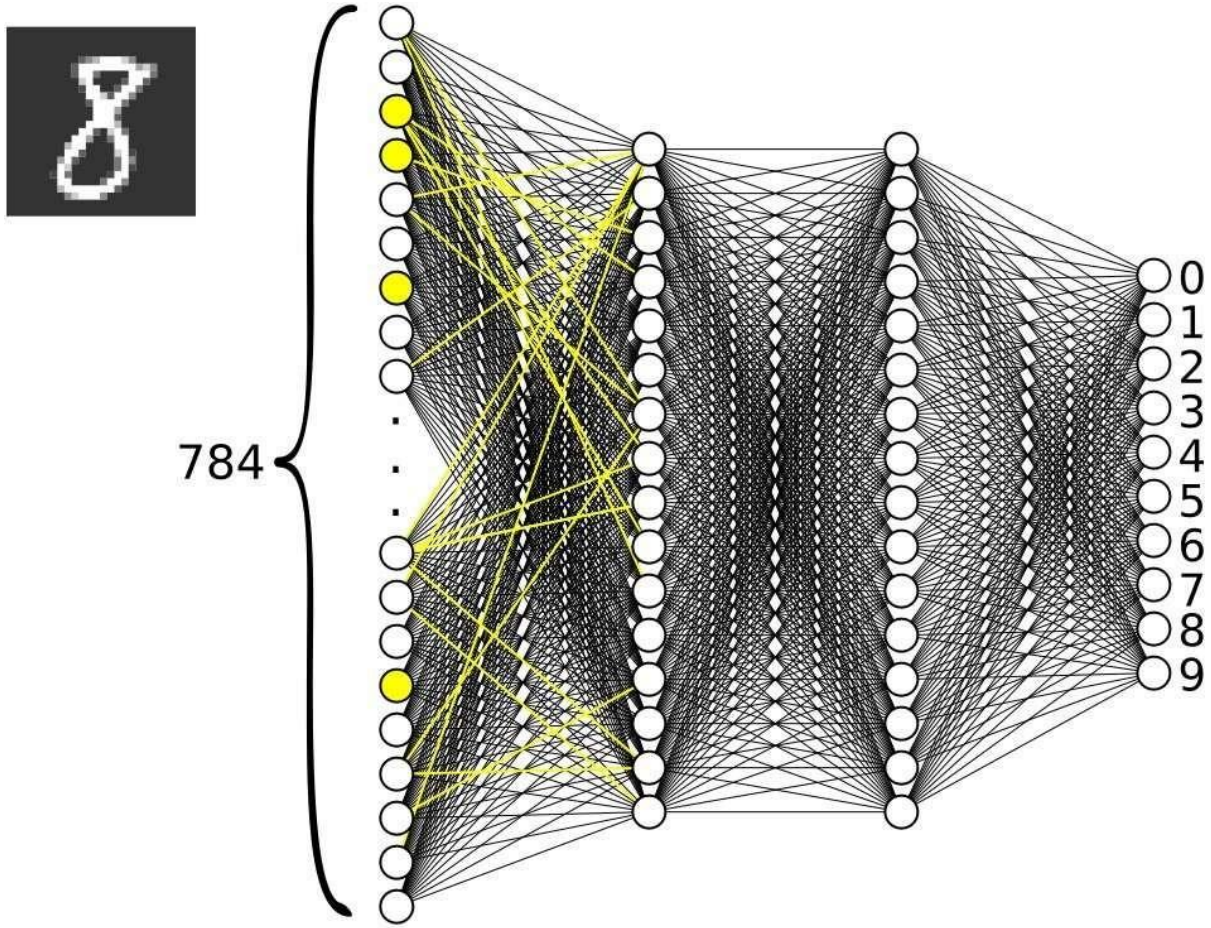
The activations of one layer become the inputs of the next layer.

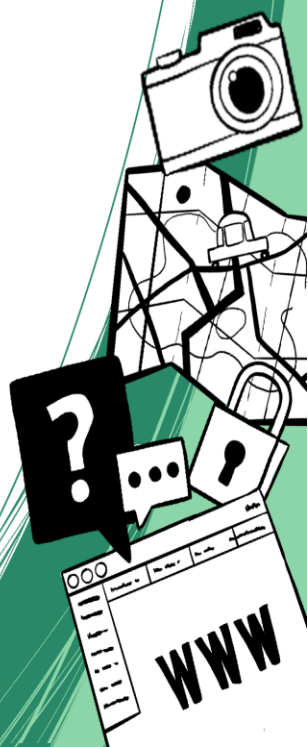




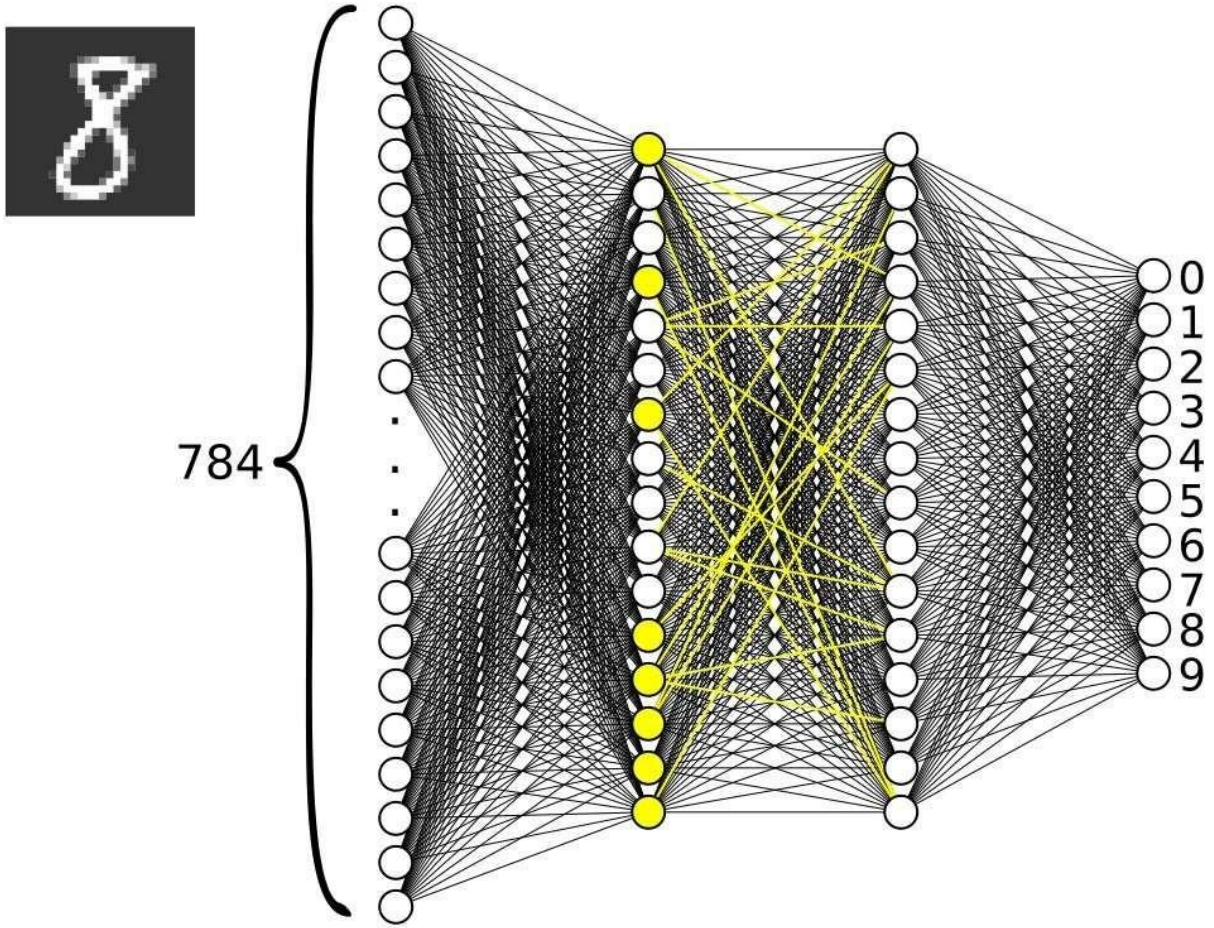
Flow of Data through the Network

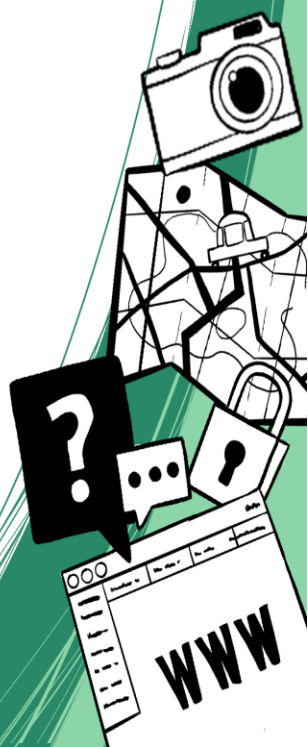
This is how the data "flows" through the network and this is how the calculation process occurs.





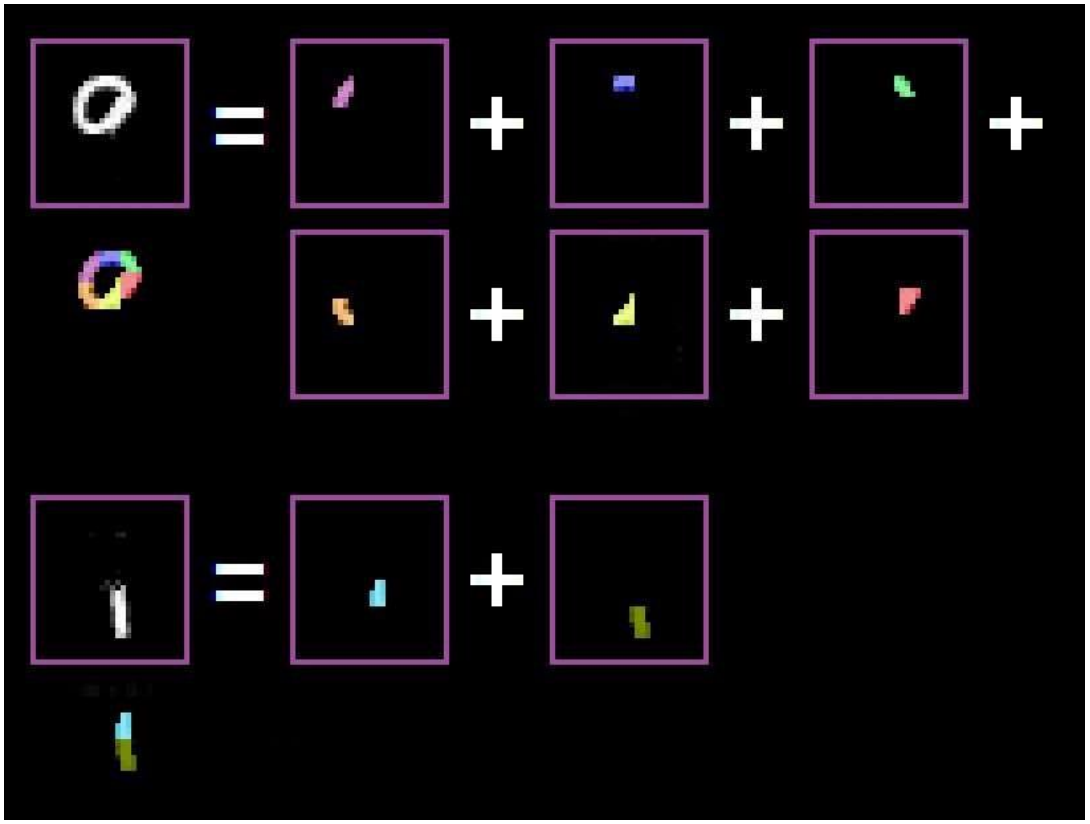
Flow of Data through the Network

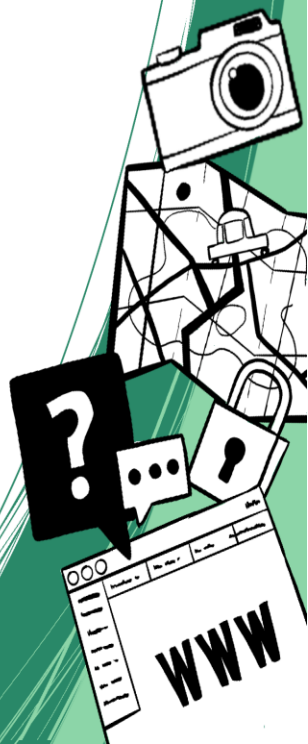




Composition

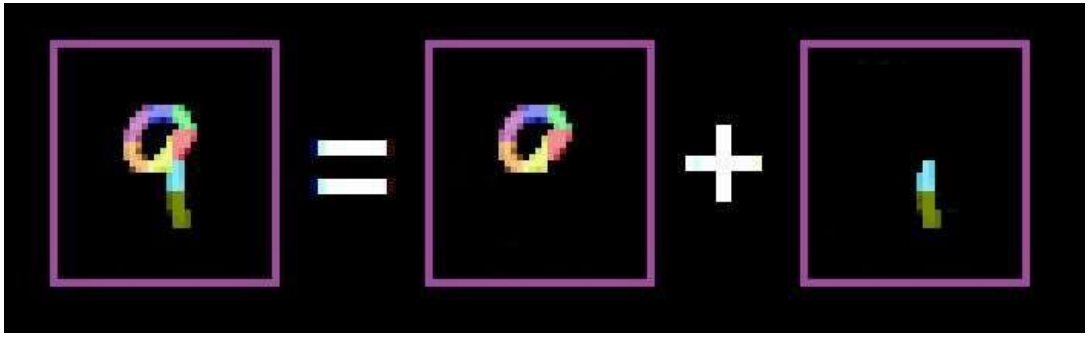
Now let's take a closer look at some characters. You can imagine them composed of smaller basic units.



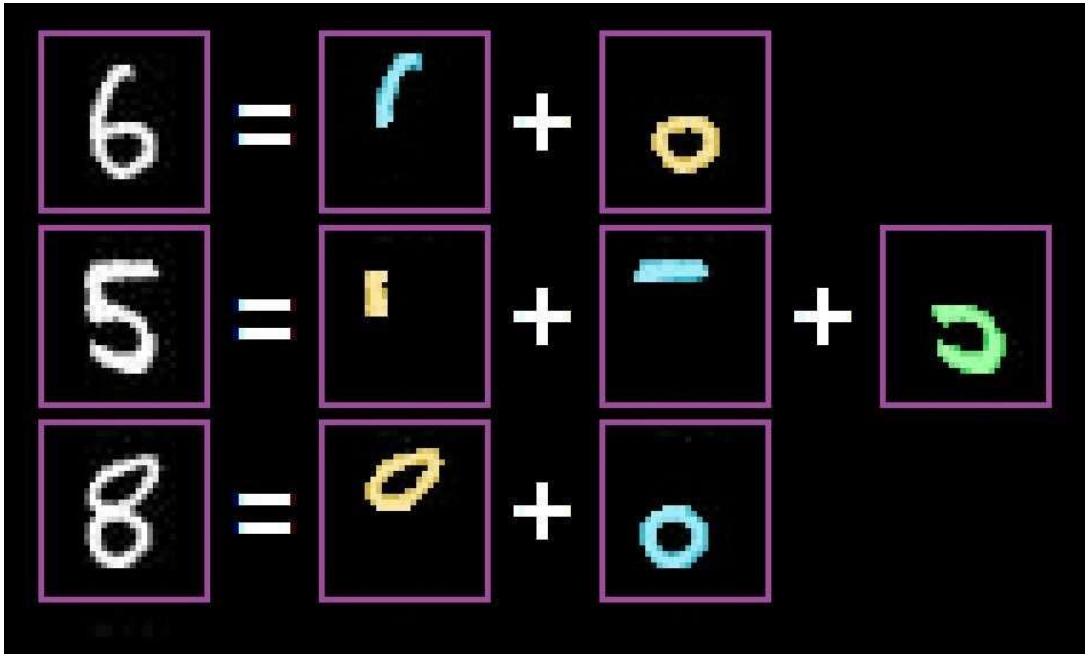


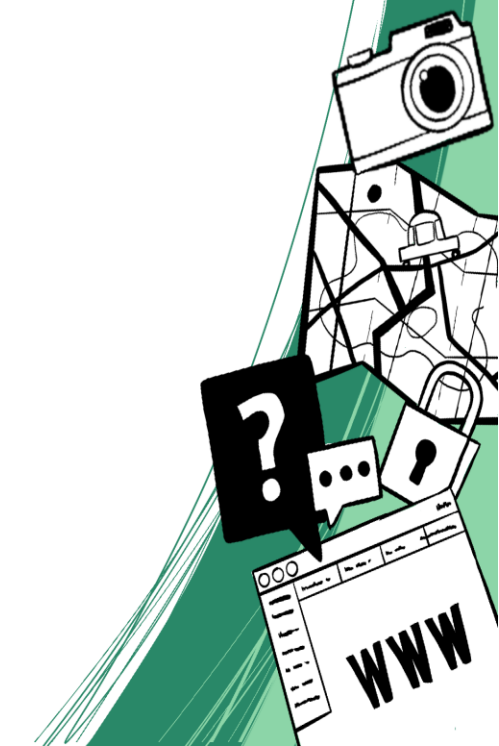
Composition

From these assembled units you can in turn assemble other larger units.



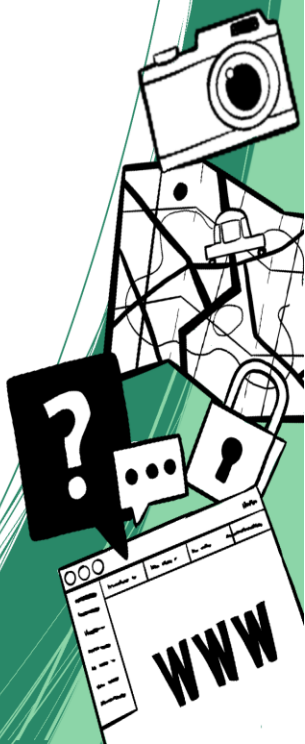
or:



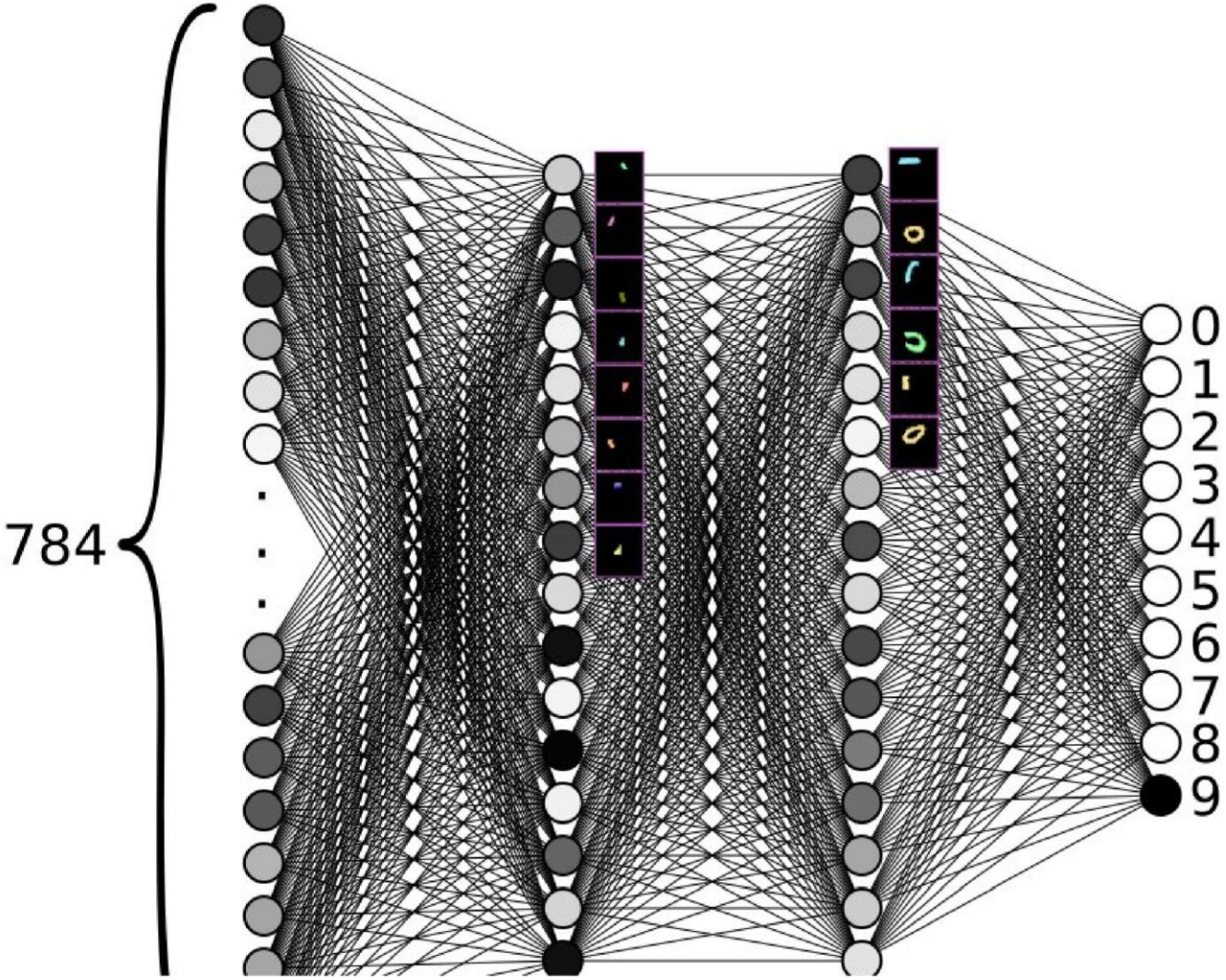


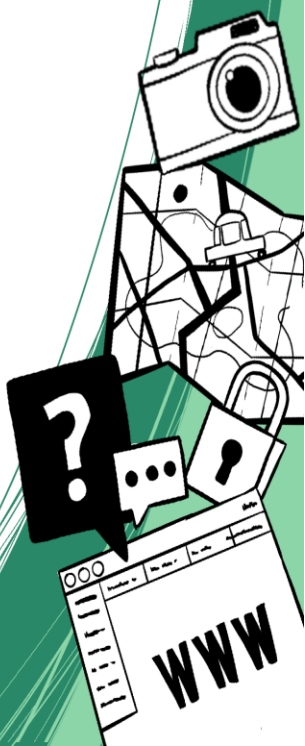
Layer Structure and Processing Steps

One interpretation of the layered structure of a neural network is that different neurons specialize in different tasks and that the different levels correspond to different processing steps.
Neurons in layers further to the left would then search for simpler patterns, neurons in layers further to the right would search for more complicated patterns.



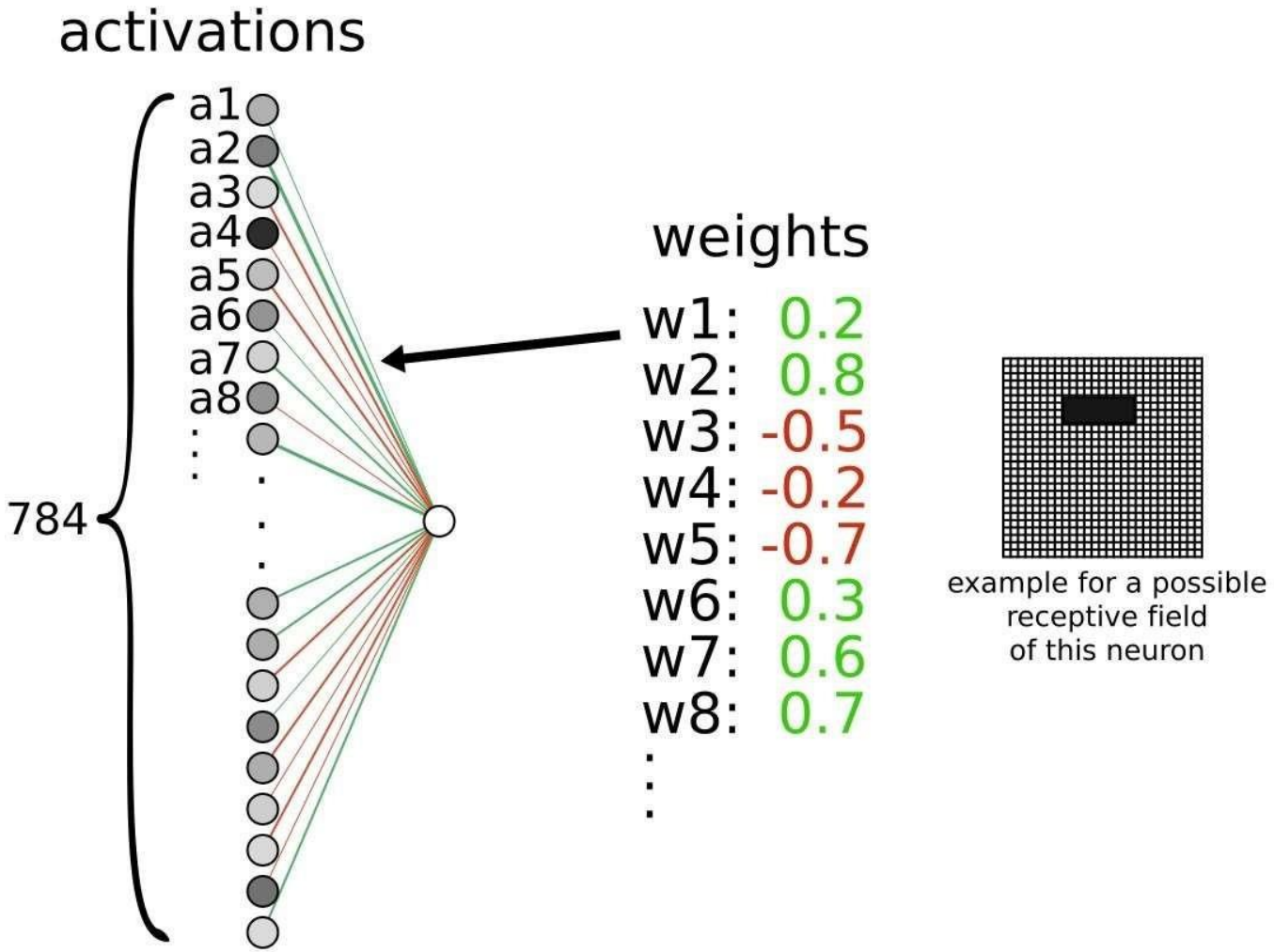
ENARIS

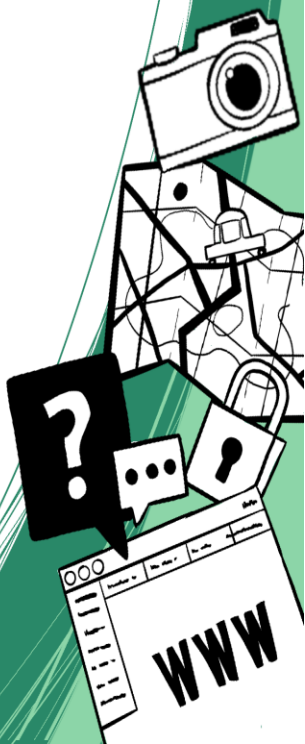




Implementation details

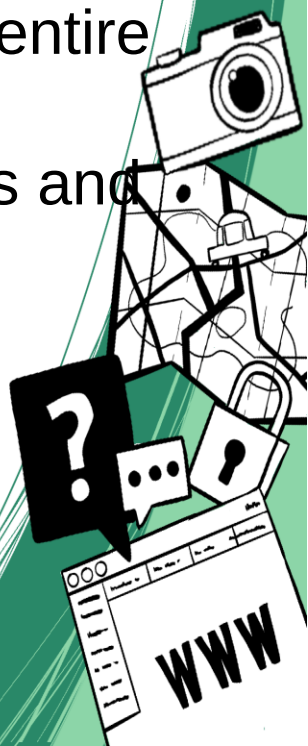
Each connection has a specific weight.





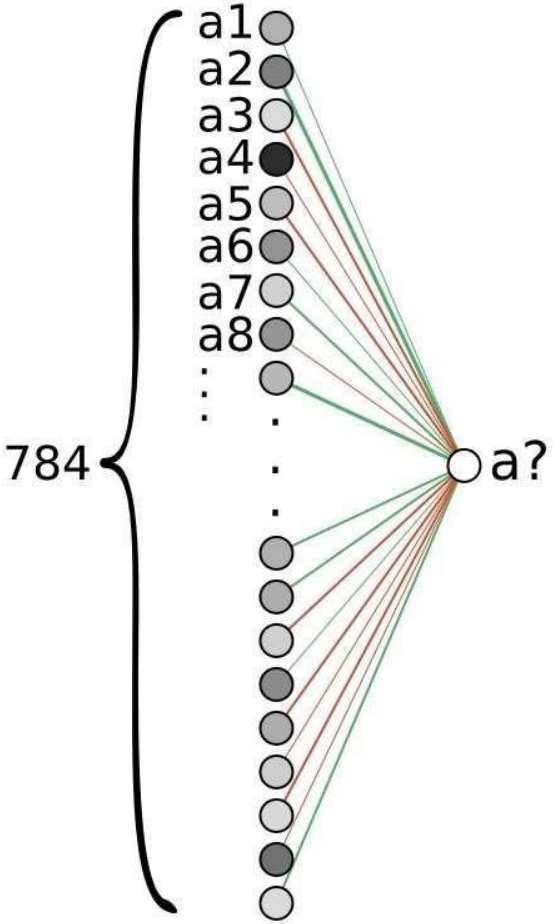
Receptive Field

The neuron sums all incoming activations. The so-called "receptive field" is indicated on the right. This is the area of input neurons that relays information to a single downstream neuron. All weights that are not equal to 0 (or their amount is below some small chosen value) must therefore be taken into account. In our example, the receptive field of the neuron under consideration is still very large and may encompass the entire input area. In the course of the training (explained later), however, the neurons will each specialize in certain areas and the receptive field will shrink.



Receptive Field

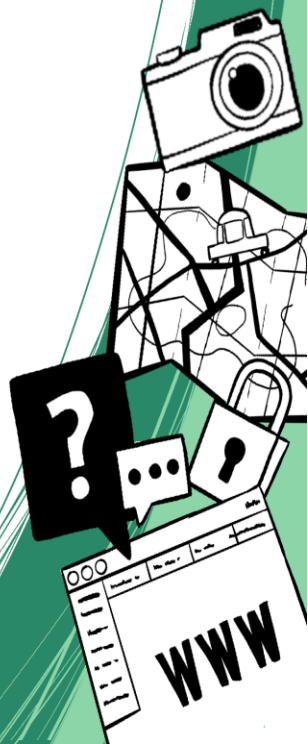
new activation:
w1*a1+w2*a2+w3*a3+ ... +wn*an



weights

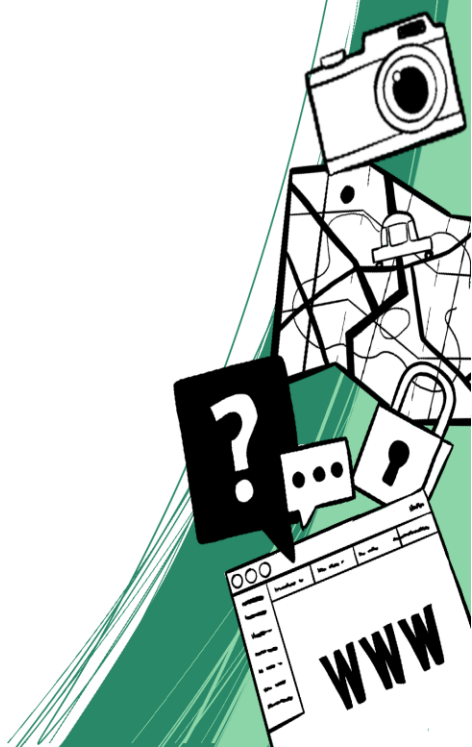
w1: 0.2
w2: 0.8
w3: -0.5
w4: -0.2
w5: -0.7
w6: 0.3
w7: 0.6
w8: 0.7

example for a possible receptive field of this neuron (blue)



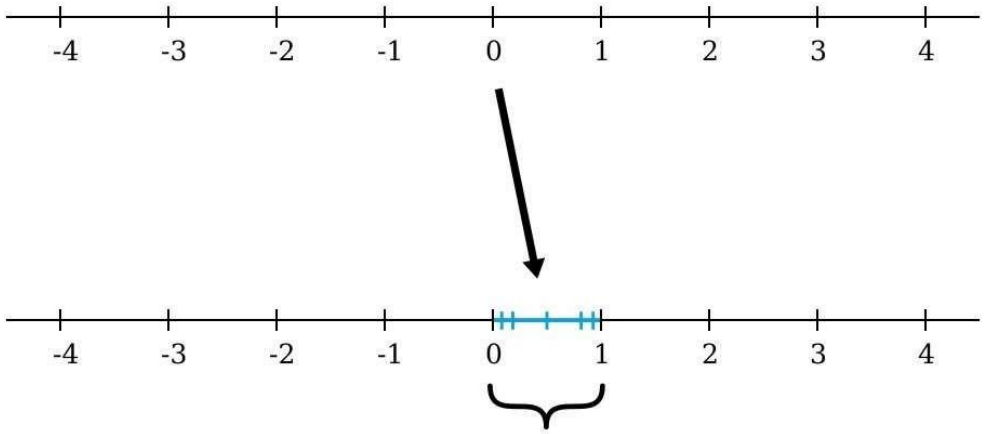
Restrict()-Function

A multi-stage process with multiplications and sums takes place in the network. The values could easily (in a positive or negative direction) get out of hand (increase strongly). We will therefore use a function "restrict()" which ensures that the values are always restricted to the range between 0 and 1. Values that are already between 0 and 1 are hardly changed or not changed at all, values that are too large or too small are compressed so that they fall within the desired limits.

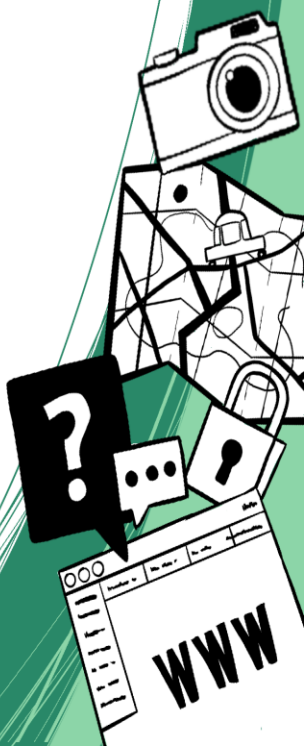


Restrict()-Function

w1*a1+w2*a2+w3*a3+ ... +wn*an

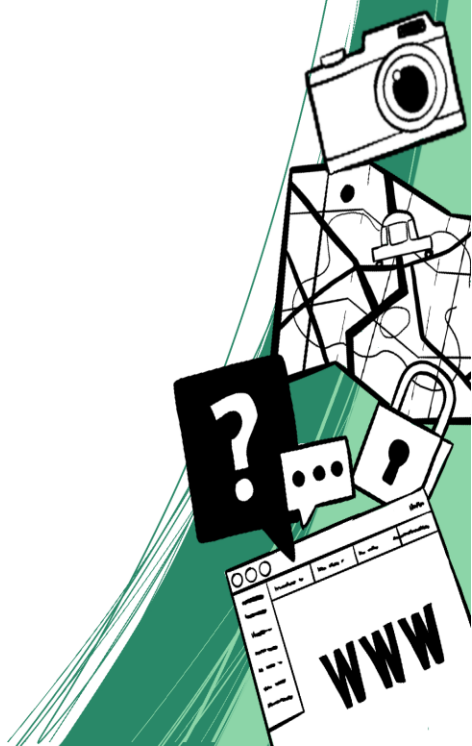


activations should be
restricted to this range



Bias

We have one addition.
Neurons can have a certain bias.
This can manifest itself in a certain sluggishness of excitement (then more input is needed to activate the neuron) or in pre-excitation (then less input is needed to activate the neuron).



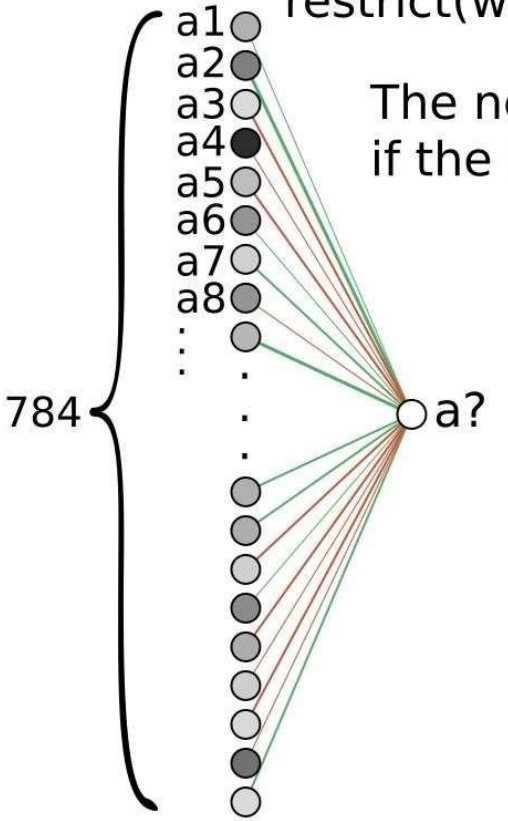
Bias

restrict function

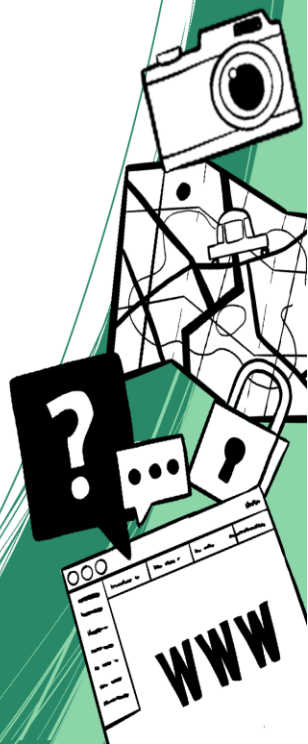
restrict(w1*a1+w2*a2+w3*a3+ ... +wn*an-10

bias

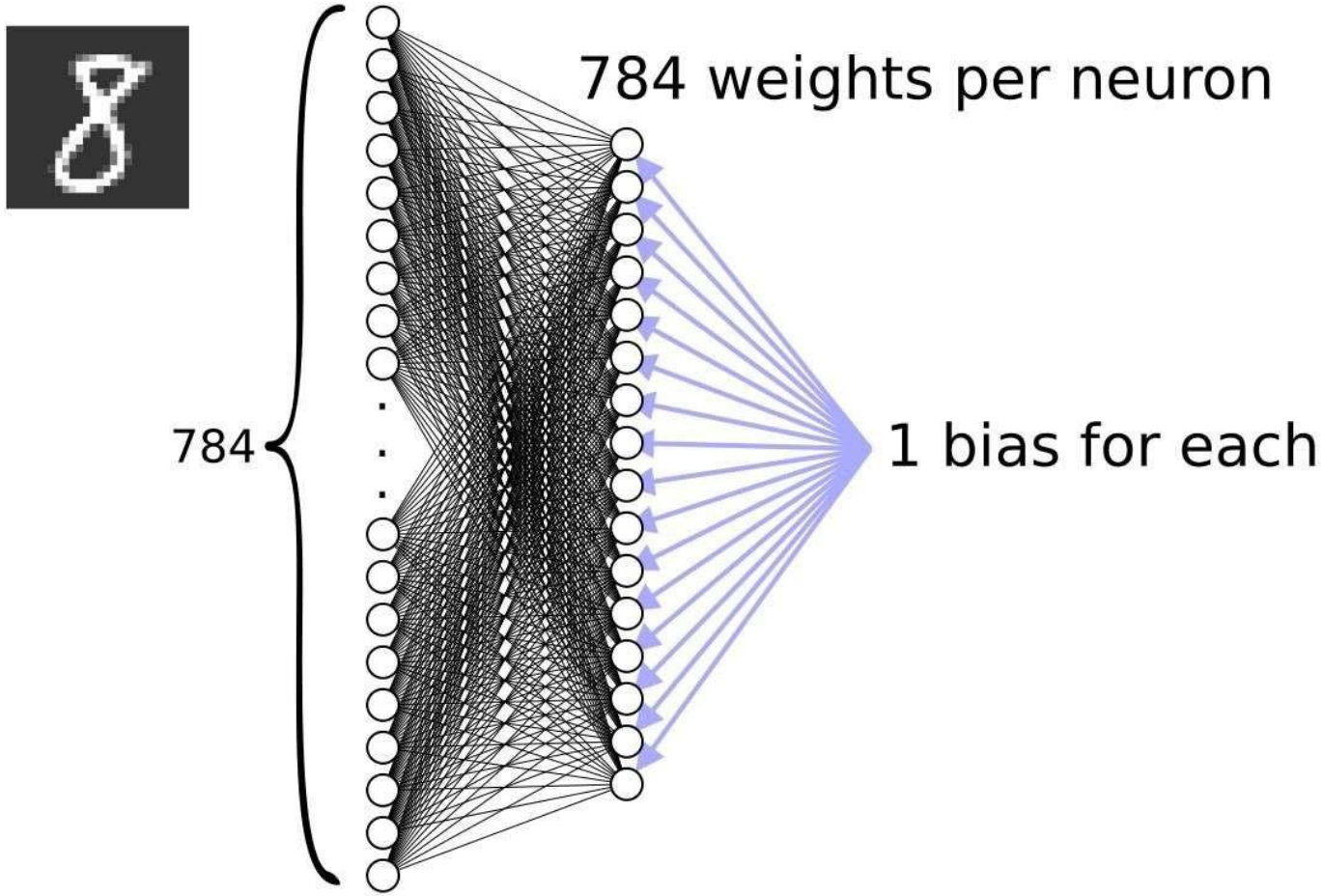
The neuron will only be activated
if the weighted sum is > 10

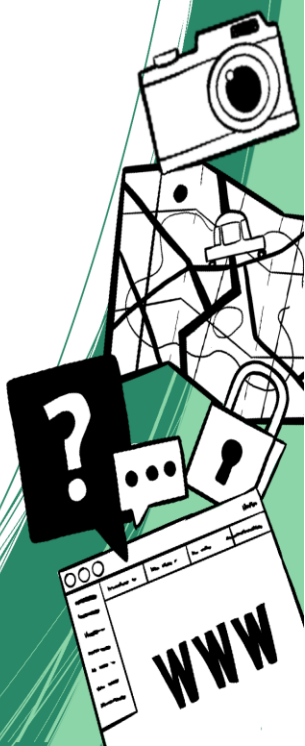


weights of the
receptive field
visualized
(green positive, red negative)



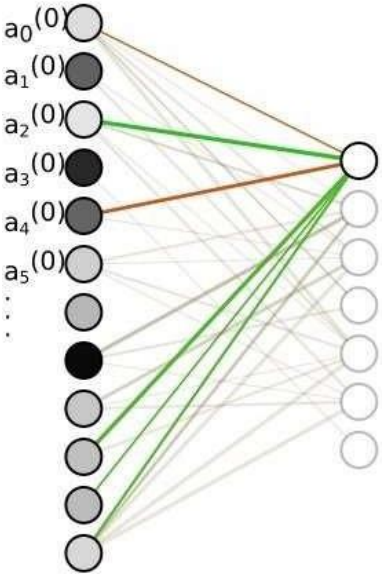
Bias





Complete Activation Function

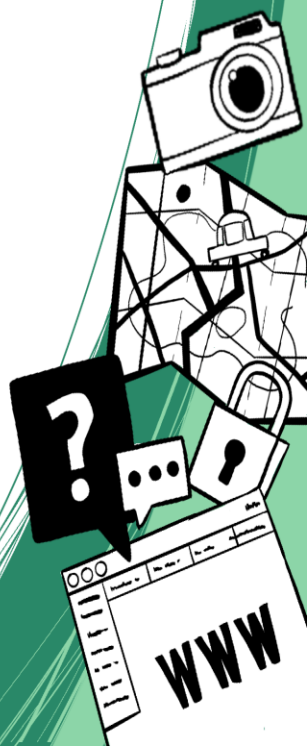
Altogether one can write:



restrict function

bias

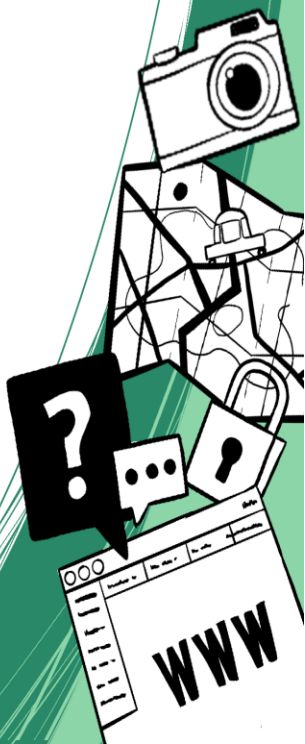
$$a_0^{(1)} = \text{restrict}(w_{0,0} * a_0^{(0)} + w_{1,0} * a_1^{(0)} + \ldots + w_{2,0} * a_2^{(0)} + b_0)$$



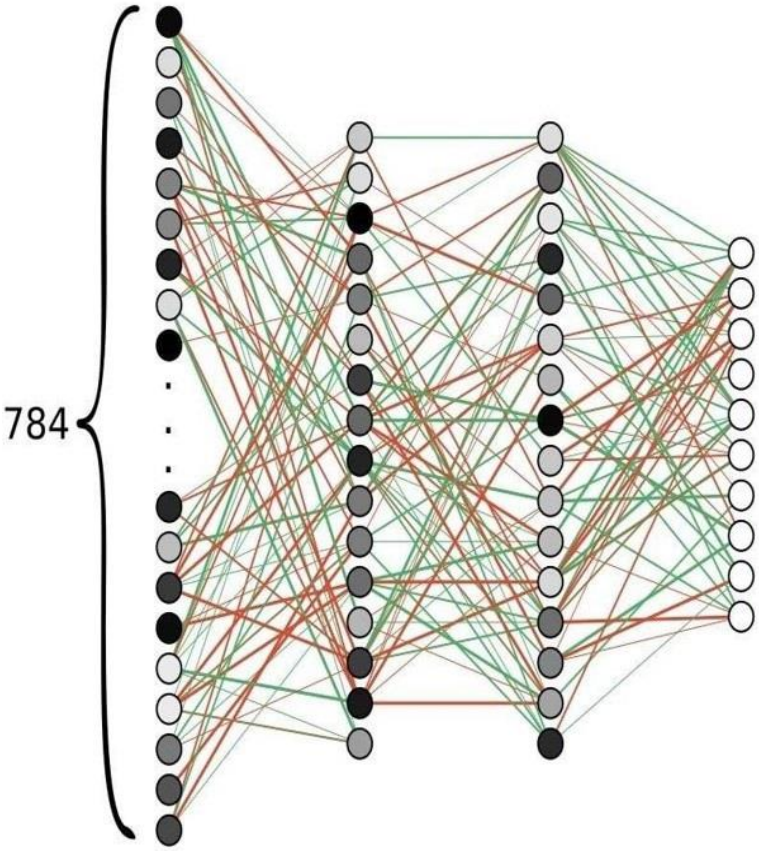
Exercises

Find the answers:

- Where is the knowledge of a neural network stored?
- What happens during the learning process?
- How many degrees of freedom (these are all parameters that change during the process) does the neural network have?

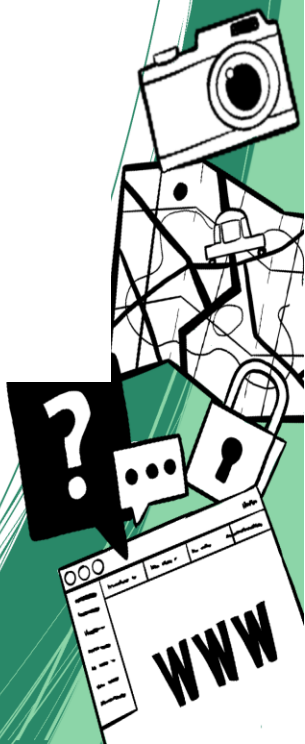


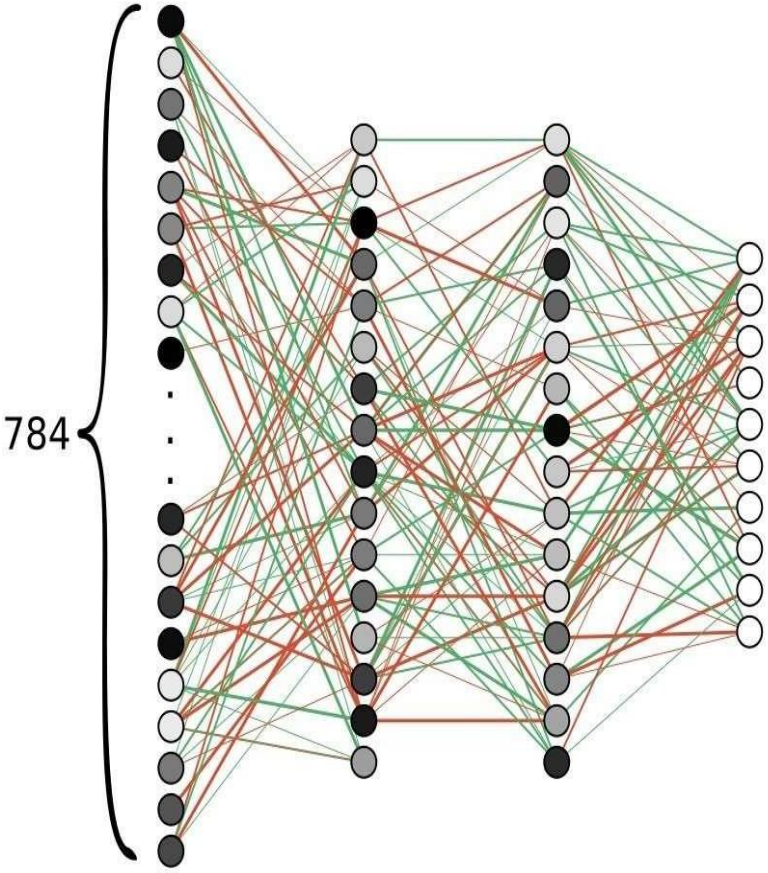
ENARIS



Exercises

Where is the knowledge of a neural network stored?
What happens during the learning process?
How many degrees of freedom (these are all parameters that change during the process) does the neural network have?





Exercises

Where is the knowledge of a neural network stored?

In the weights and biases.

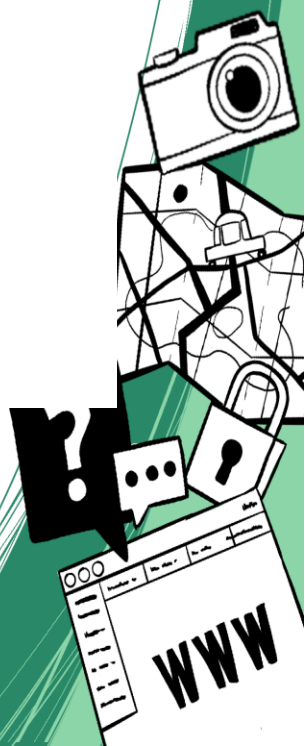
What happens during the learning process?

The weights and biases are adjusted (changed).

How many degrees of freedom (these are all parameters that change during the process) does the neural network have?

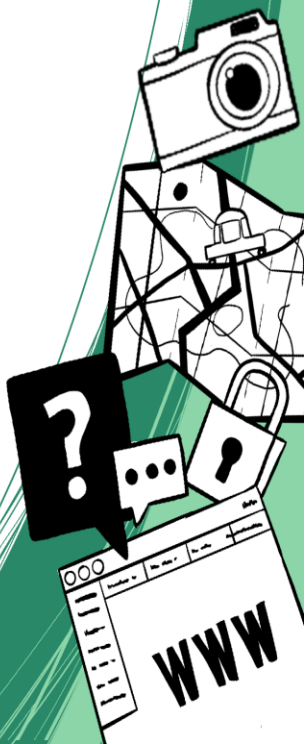
784*16 + 16*16 + 16*10
weights
16 + 16 + 10
biases

13 002

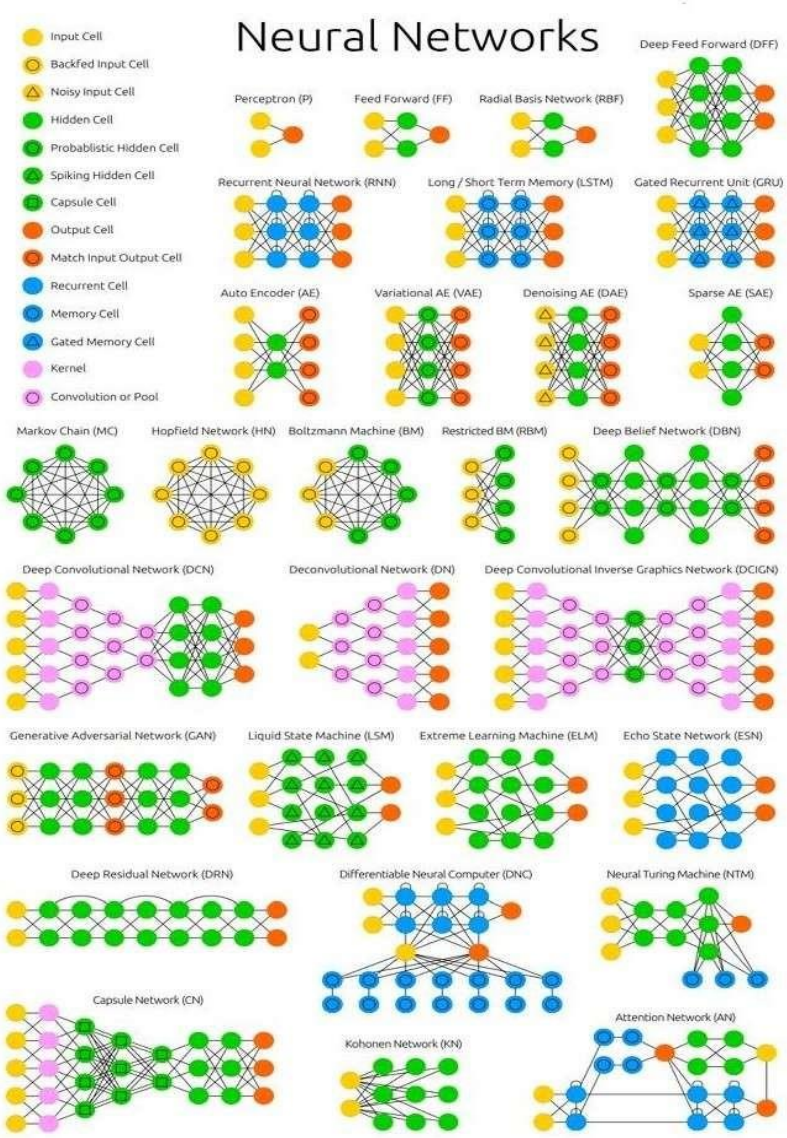


Network arrangements

Here we've looked at a fairly simple and widely used type of network, a feed-forward network. There is a large and rapidly growing number of different network configurations (topologies) that are used for specific applications.

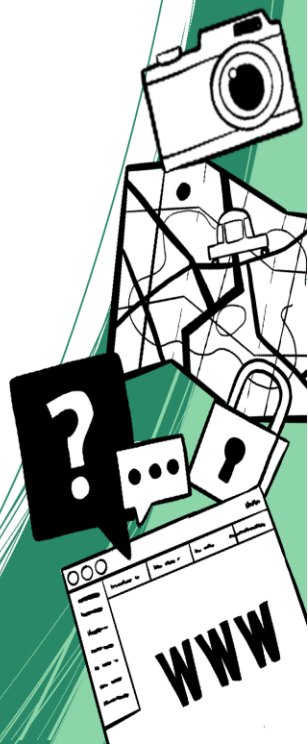


„Neural Network Zoo“



Stefan Leijnen and Fjodor van Veen

https://www.researchgate.net/publication/341373030_The_Neural_Network_Zoo/fulltext/5ebd3f60a6fdcc90d6752941/The Neural-Network-Zoo.pdf



Links

But what *is* a neural network? | Part 1, Deep Learning
https://www.youtube.com/watch?v=aircAruvnKk