Where are the fancy neural networks?

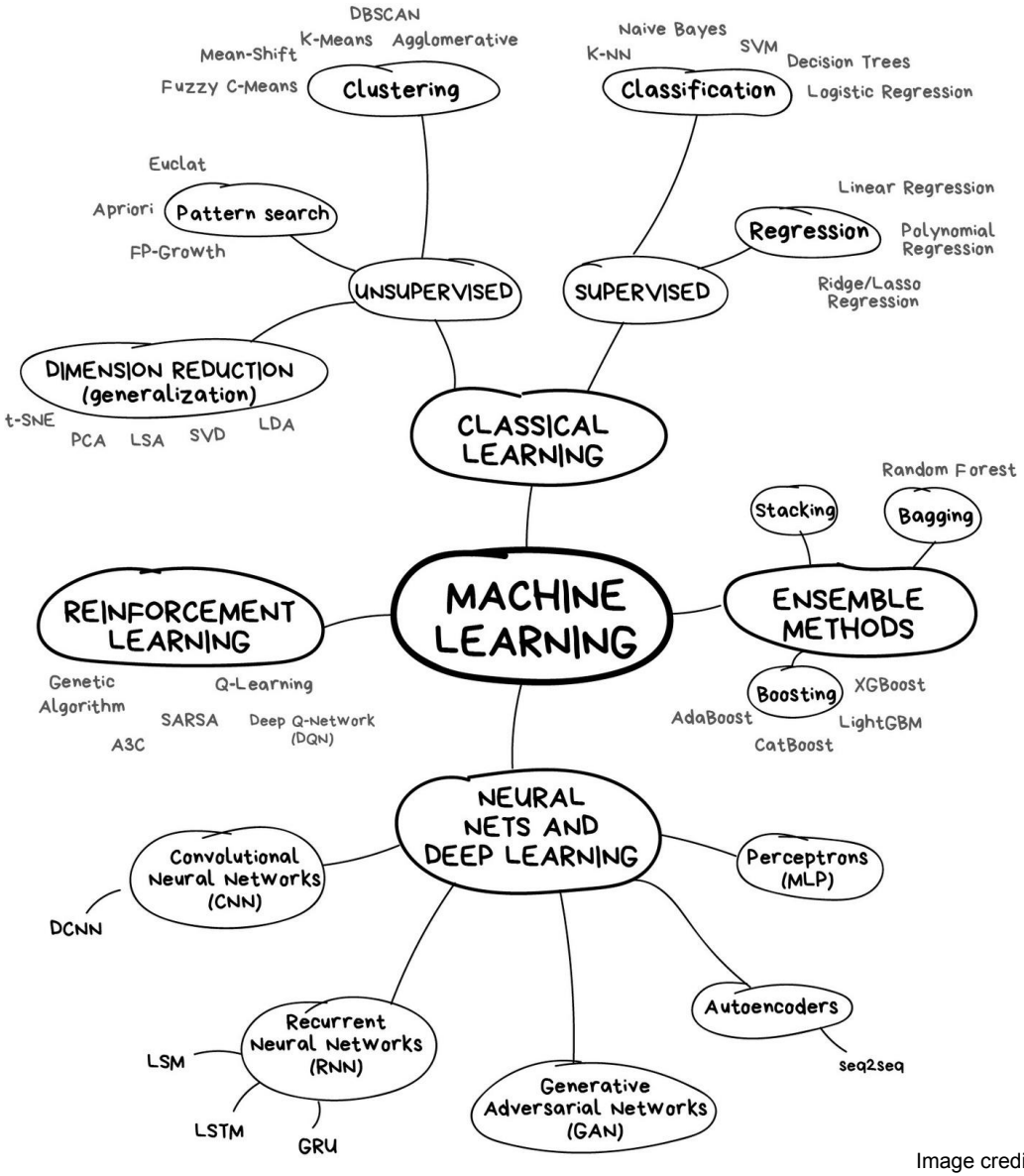


Image credit: https://vas3k.com/blog/machine_learning/

Neural Networks

Any neural network is a collection of **neurons** and **connections** between them.

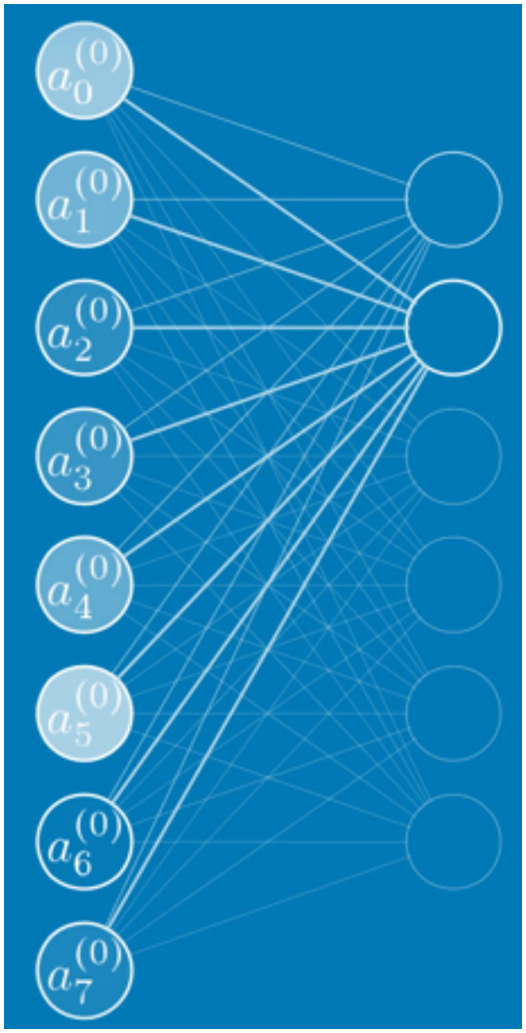
Neuron is a function with a set of inputs and one output. Its task is to take all numbers from its input, apply a function on them and send the result to the output.

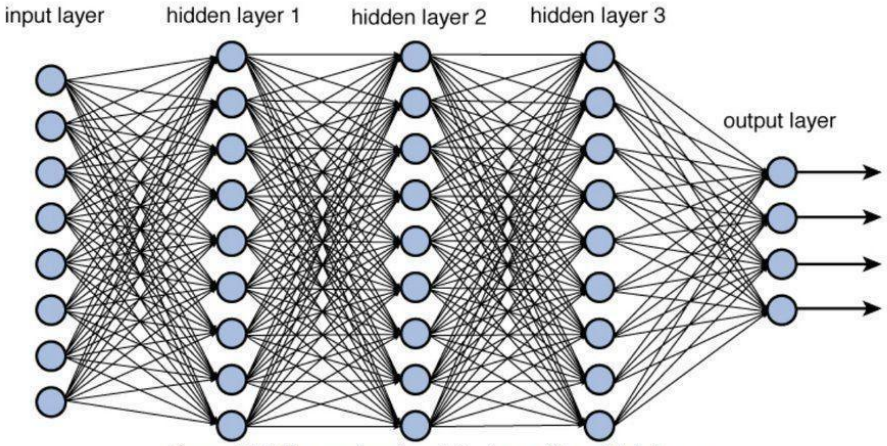
- Example: sum up all numbers from the inputs and if that sum is bigger than N give 1 as a result. Otherwise return zero.

Connections are like channels between neurons. They connect outputs of one neuron with the inputs of another so they can send digits to each other. Each connection has only one parameter the *weight*.

- These weights tell the neuron to respond more to one input and less to another. Weights are adjusted when training — that's how the network learns.

How do NNs work?





layer

$a_0^{(1)} = f(w_{0,0}\, a_0^{(0)} + w_{0,1}\, a_1^{(0)} + \ldots + w_{0,n}\, a_n^{(0)} + b_0)$

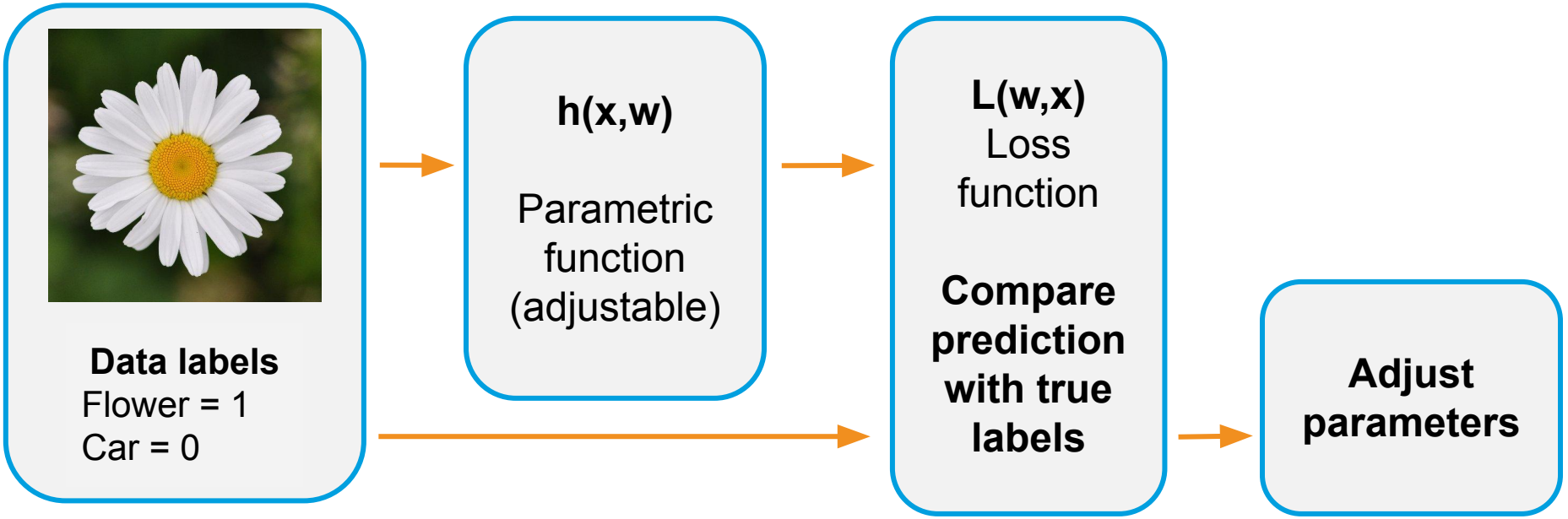
activation function

weights

bias

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

Supervised learning



Design function with adjustable parameters.

Design a loss (sometimes called cost) function.

Optimise the parameters:

- Compare with labels to compute loss
- Adjust parameters to minimise loss
- Repeat until parameters stabilise