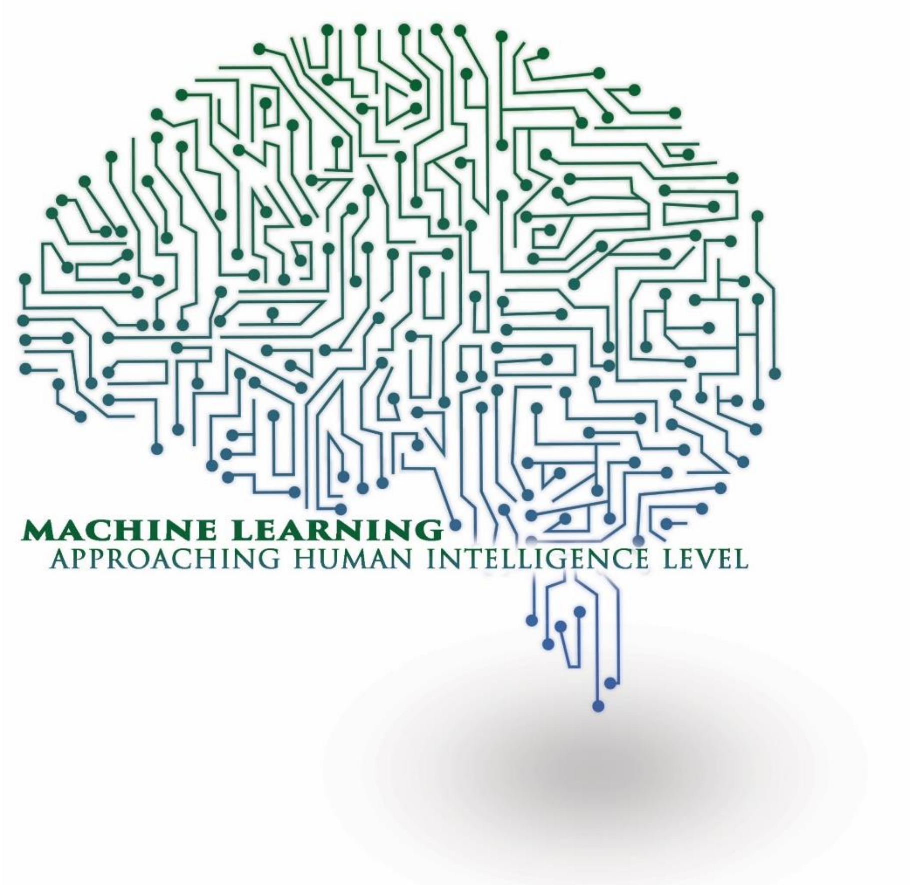
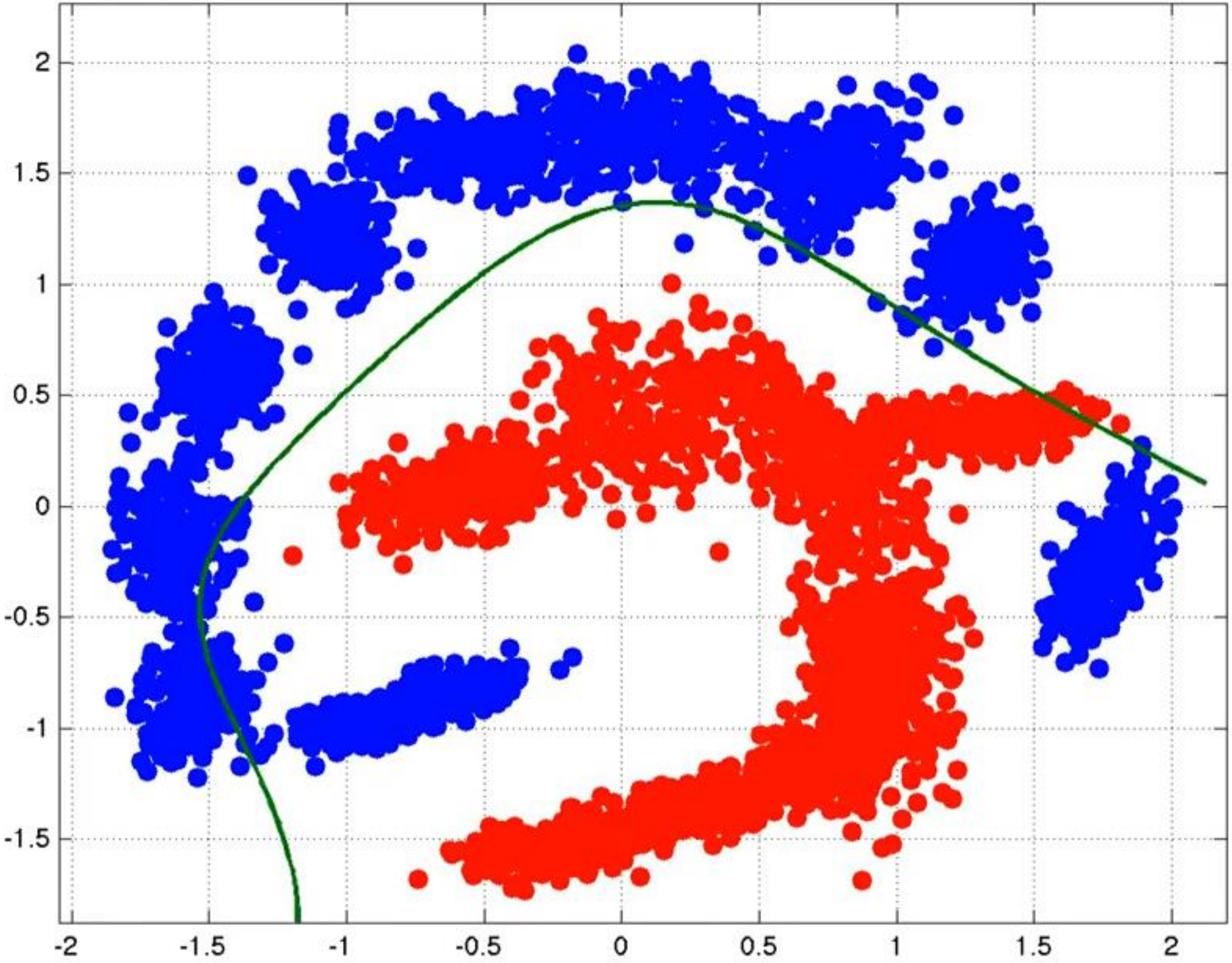


Introduction to Artificial neural networks.



2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2

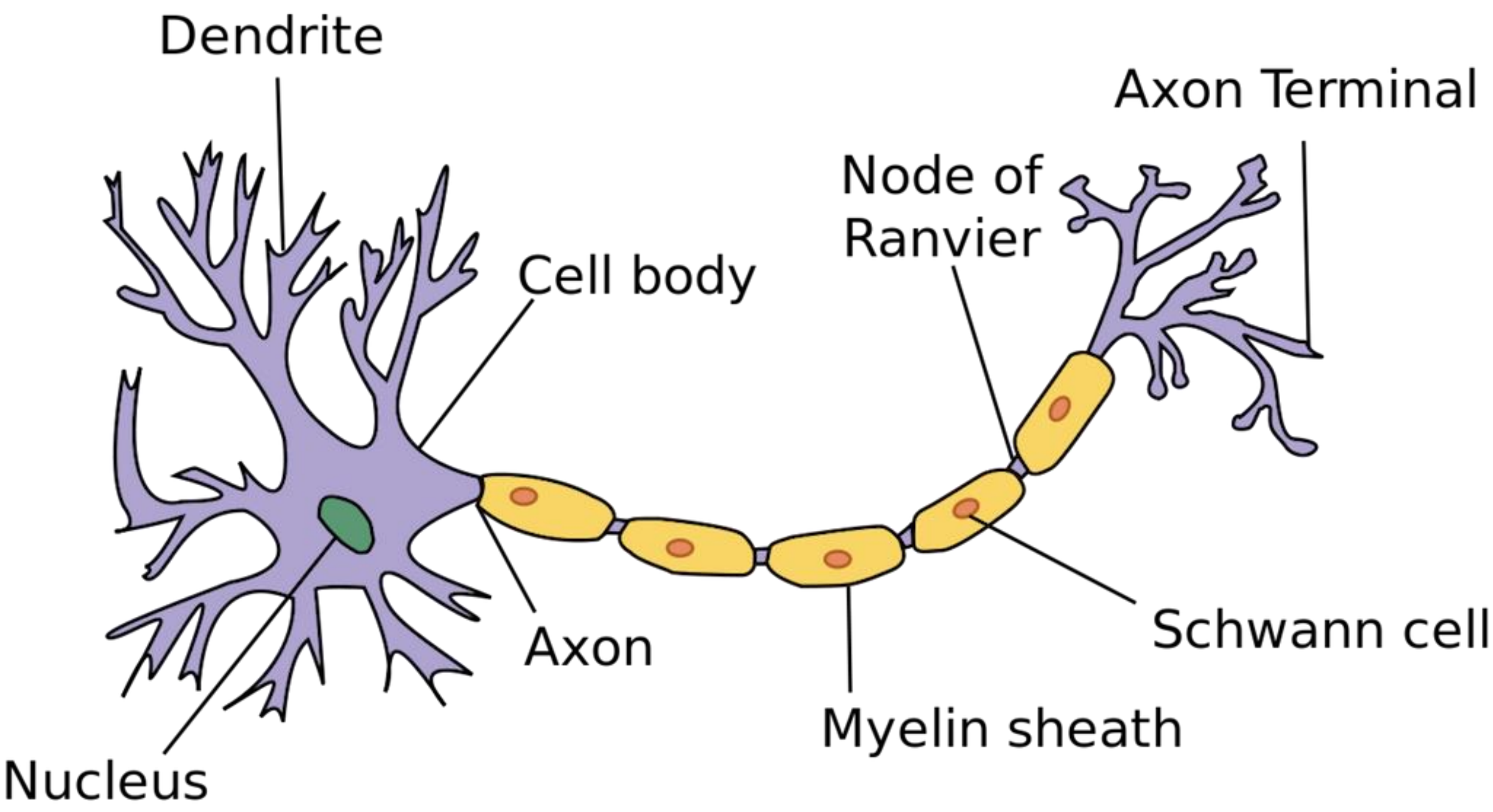
Introduction to biological neural

Dendrites: receives messages from other neurons

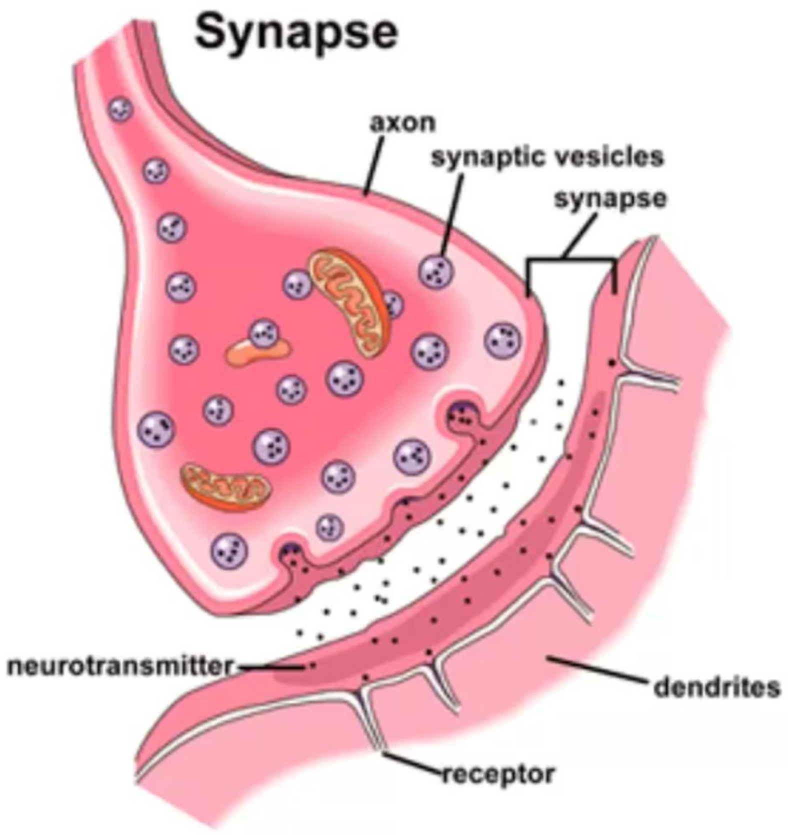
Nucleus:

Axon: send messages to other neurons

Synapse: when the axon connect other neuron’s dendritic tree



How Synapse works & learning



When an impulse travel down the axon, it eventually reaches the axon terminal.

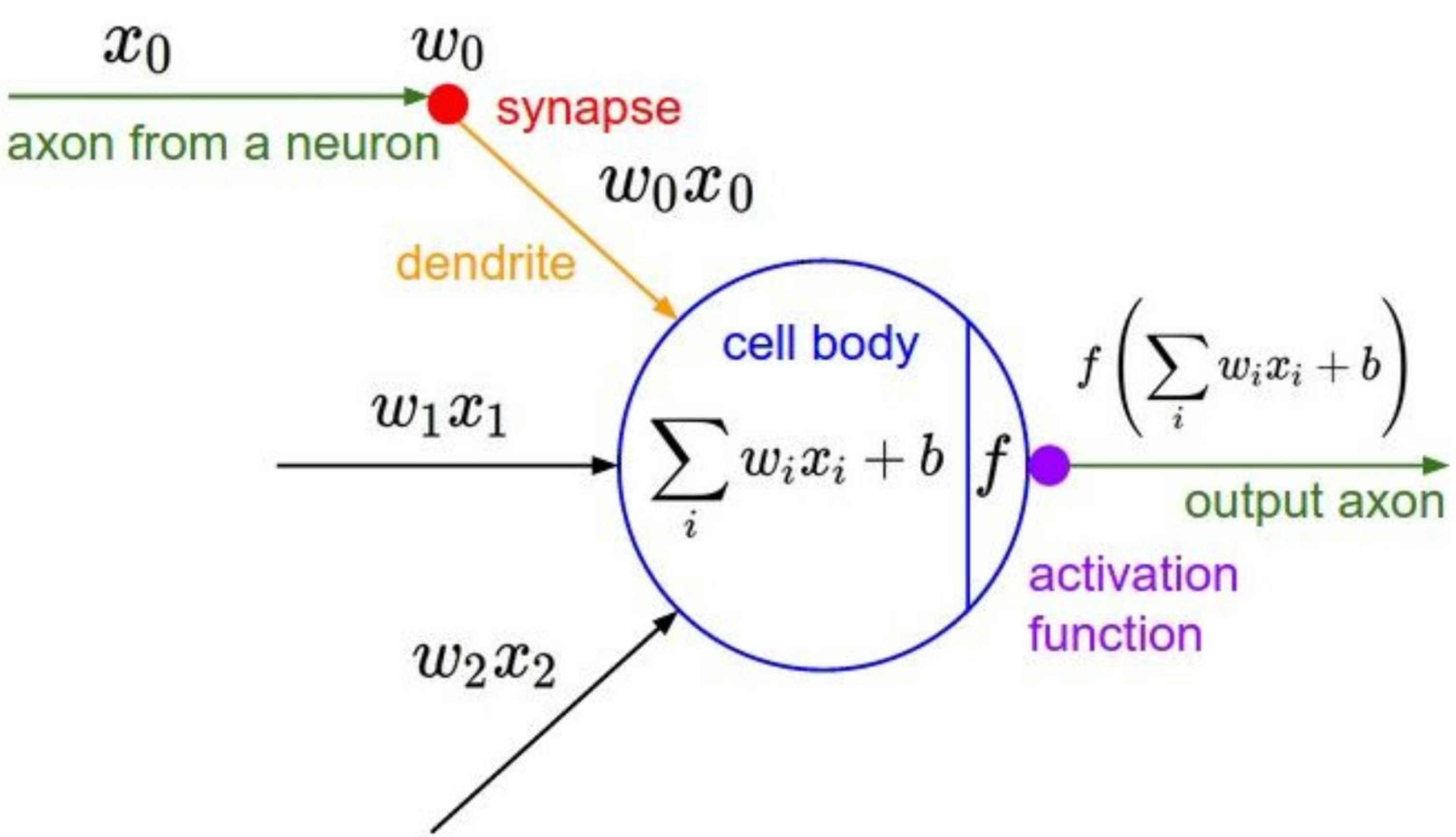
When it reach the terminal, the change in electrical charge triggers biochemical reactions that lead to the release of neurotransmitter.

They can adapt by varying the number of vesicles of transmitter.

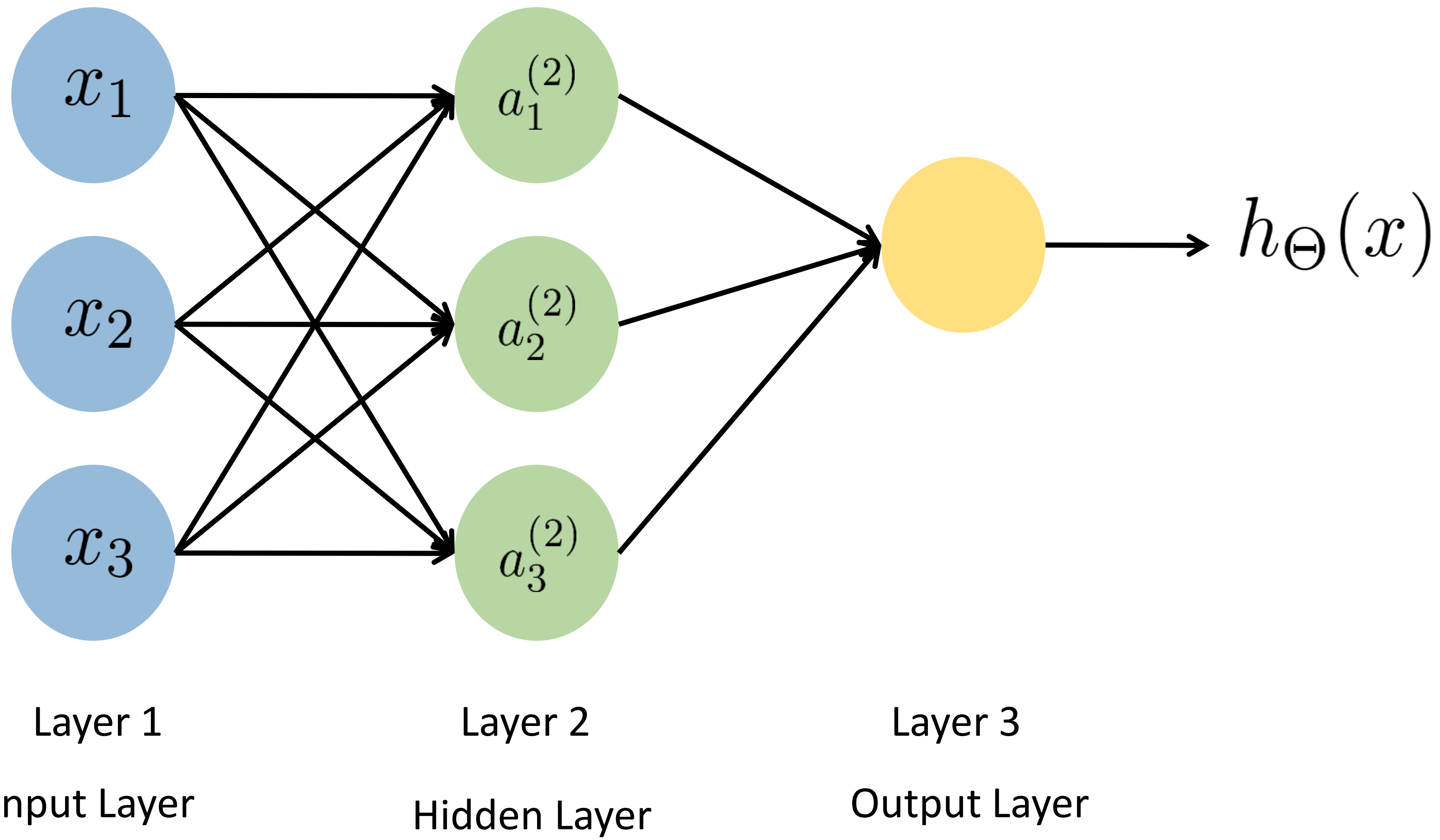
They can adapt by varying the number of receptor molecules.

How Synapse works

Artificial neural



Neural network



Neuron model: Activation Functions

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

tanh: $tanh(x)$

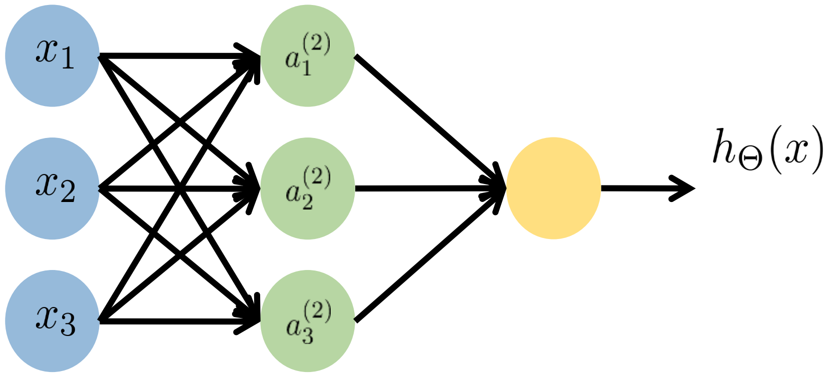
ReLu: $\max(x, 0)$

LeakyReLu:

Maxout:

ELU:

Neural network



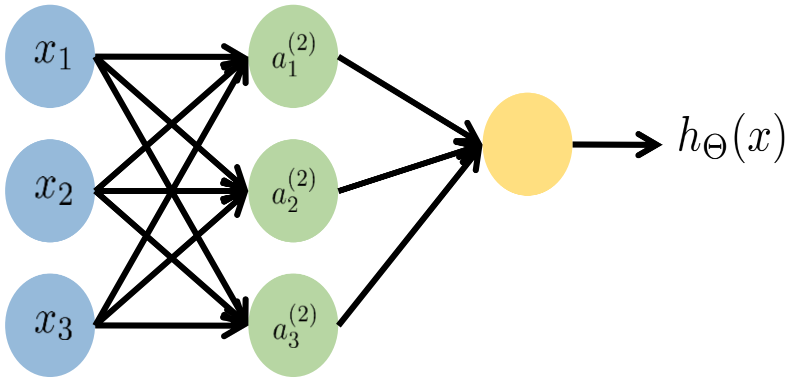
$a_i^{(j)} =$ "activation" of unit i in layer j

$\Theta^{(j)} =$ matrix of weights controlling function mapping from layer j to layer $j + 1$.

$$
\begin{aligned}
a_1^{(2)} &= g(\Theta_{10}^{(1)} x_0 + \Theta_{11}^{(1)} x_1 + \Theta_{12}^{(1)} x_2 + \Theta_{13}^{(1)} x_3) \\
a_2^{(2)} &= g(\Theta_{20}^{(1)} x_0 + \Theta_{21}^{(1)} x_1 + \Theta_{22}^{(1)} x_2 + \Theta_{23}^{(1)} x_3) \\
a_3^{(2)} &= g(\Theta_{30}^{(1)} x_0 + \Theta_{31}^{(1)} x_1 + \Theta_{32}^{(1)} x_2 + \Theta_{33}^{(1)} x_3) \\
h_\Theta(x) = a_1^{(3)} &= g(\Theta_{10}^{(2)} a_0^{(2)} + \Theta_{11}^{(2)} a_1^{(2)} + \Theta_{12}^{(2)} a_2^{(2)} + \Theta_{13}^{(2)} a_3^{(2)})
\end{aligned}
$$

If network has u_j units in layer j , u_{j+1} units in layer $j + 1$, then $\theta^{(j)}$ will be of dimension $u_{j+1} \times (u_j + 1)$.

Neural network forward propagation



$$a_1^{(2)} = g(\Theta_{10}^{(1)} x_0 + \Theta_{11}^{(1)} x_1 + \Theta_{12}^{(1)} x_2 + \Theta_{13}^{(1)} x_3)$$
$$a_2^{(2)} = g(\Theta_{20}^{(1)} x_0 + \Theta_{21}^{(1)} x_1 + \Theta_{22}^{(1)} x_2 + \Theta_{23}^{(1)} x_3)$$
$$a_3^{(2)} = g(\Theta_{30}^{(1)} x_0 + \Theta_{31}^{(1)} x_1 + \Theta_{32}^{(1)} x_2 + \Theta_{33}^{(1)} x_3)$$
$$h_\Theta(x) = a_1^{(3)} = g(\Theta_{10}^{(2)} a_0^{(2)} + \Theta_{11}^{(2)} a_1^{(2)} + \Theta_{12}^{(2)} a_2^{(2)} + \Theta_{13}^{(2)} a_3^{(2)})$$

$$x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \qquad z^{(2)} = \begin{bmatrix} z_1^{(2)} \\ z_2^{(2)} \\ z_3^{(2)} \end{bmatrix}$$

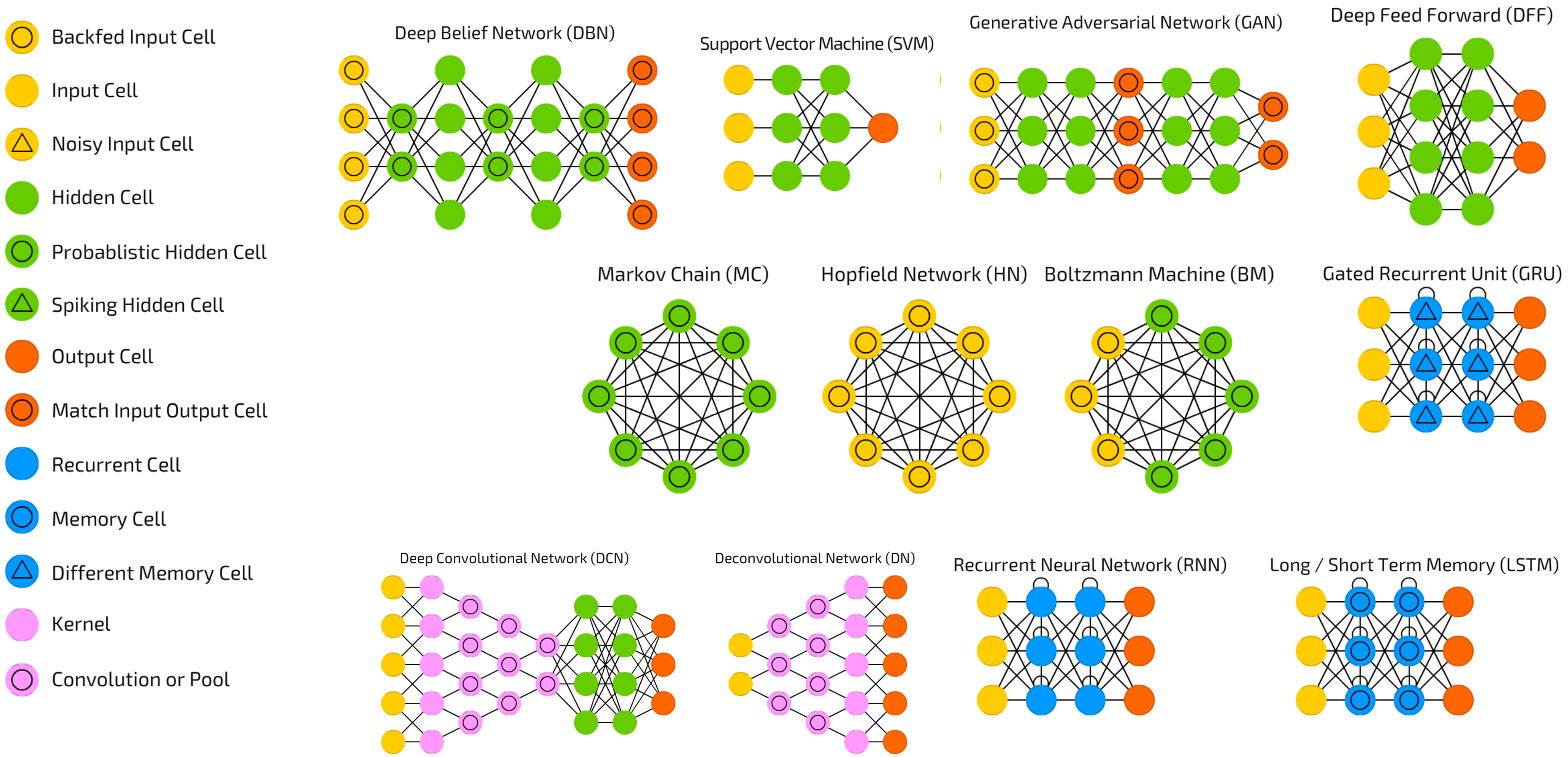
$$z^{(2)} = \Theta^{(1)} x$$
$$a^{(2)} = g(z^{(2)})$$

Add $a_0^{(1)} = 1$.

$$z^{(3)} = \Theta^{(2)} a^{(2)}$$

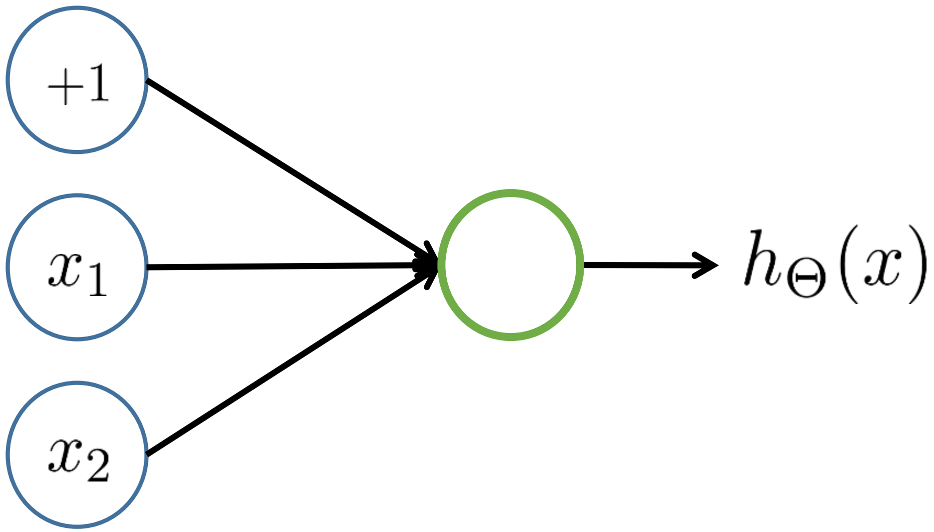
$$h_\Theta(x) = a^{(3)} = g(z^{(3)})$$

Neural network zoo



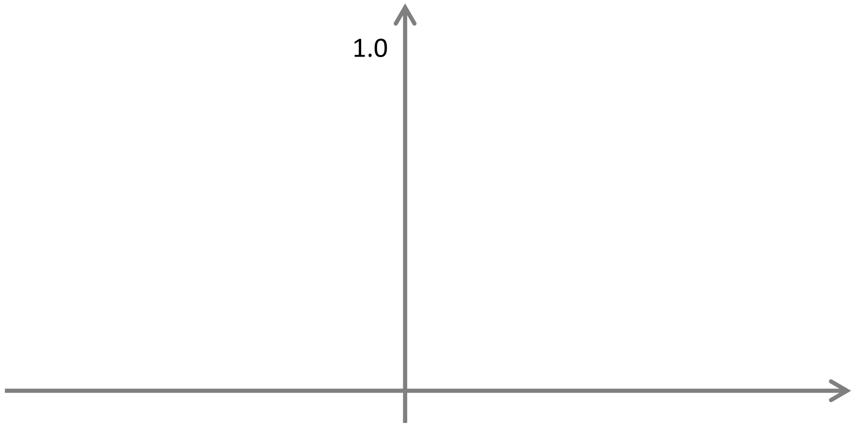
Simple example: AND

$$x_1, x_2 \in \{0, 1\}$$
$$y = x_1 \text{ AND } x_2$$



Example: OR function

$$x_1, x_2 \in \{0, 1\}$$
$$y = x_1 \text{ OR } x_2$$



x_1	x_2	$h_\Theta(x)$
0	0	
0	1	
1	0	
1	1	

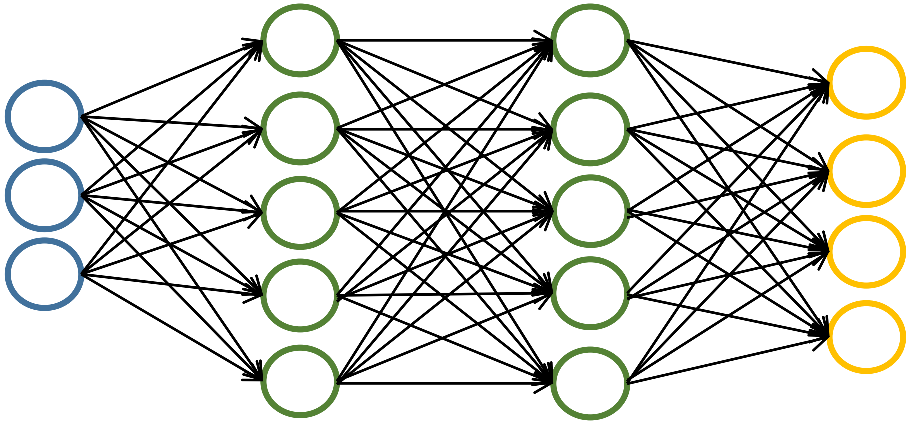
Multi-class classification

Pedestrian

Car

Motorcycle

Truck



$$h_\Theta(x) \in \mathbb{R}^4$$

Want $h_\Theta(x) \approx \begin{bmatrix}1\\0\\0\\0\end{bmatrix}$, $h_\Theta(x) \approx \begin{bmatrix}0\\1\\0\\0\end{bmatrix}$, $h_\Theta(x) \approx \begin{bmatrix}0\\0\\1\\0\end{bmatrix}$, etc.

when pedestrian when car when motorcycle

Neural network learning: Cost function

Logistic regression:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} y_i(\log(h_\theta(x_i)) + (1-y_i)log(1-h_\theta(x_i))$$

Neural network:

$$h_\Theta(x) \in \mathbb{R}^K \qquad (h_\Theta(x))_i = i^{th} \text{ output}$$

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{K} y_k^{(i)} \log\left(h_\theta\left(x^{(i)}\right)\right)_k + \left(1-y_k^{(i)}\right) log(1-\left(h_\theta\left(x^{(i)}\right)\right)_k$$

jupyter

ANACONDA®

scikit learn