

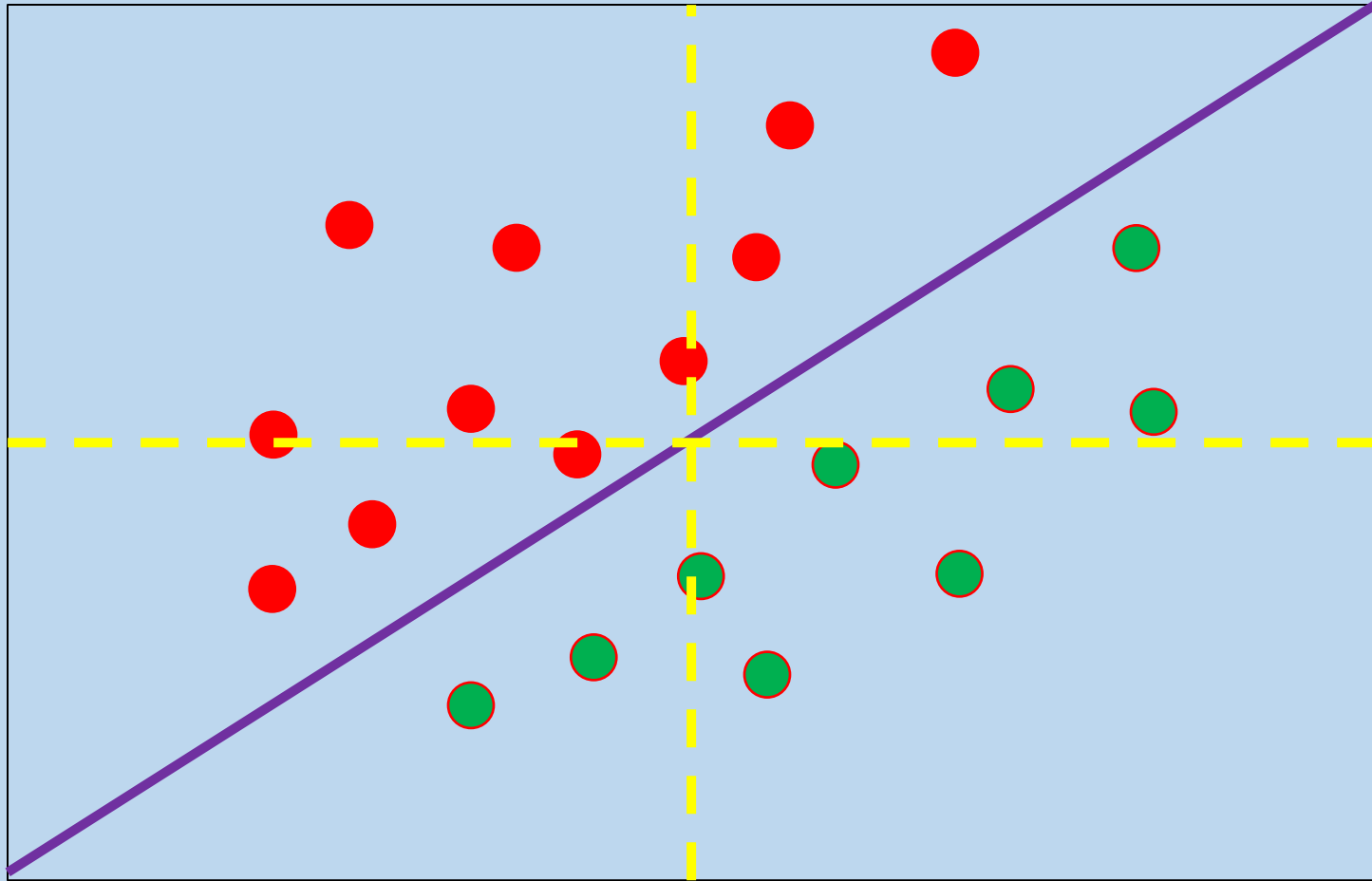
Deriving Rules From Data

Deriving Rules from Data Machine Learning Algorithms

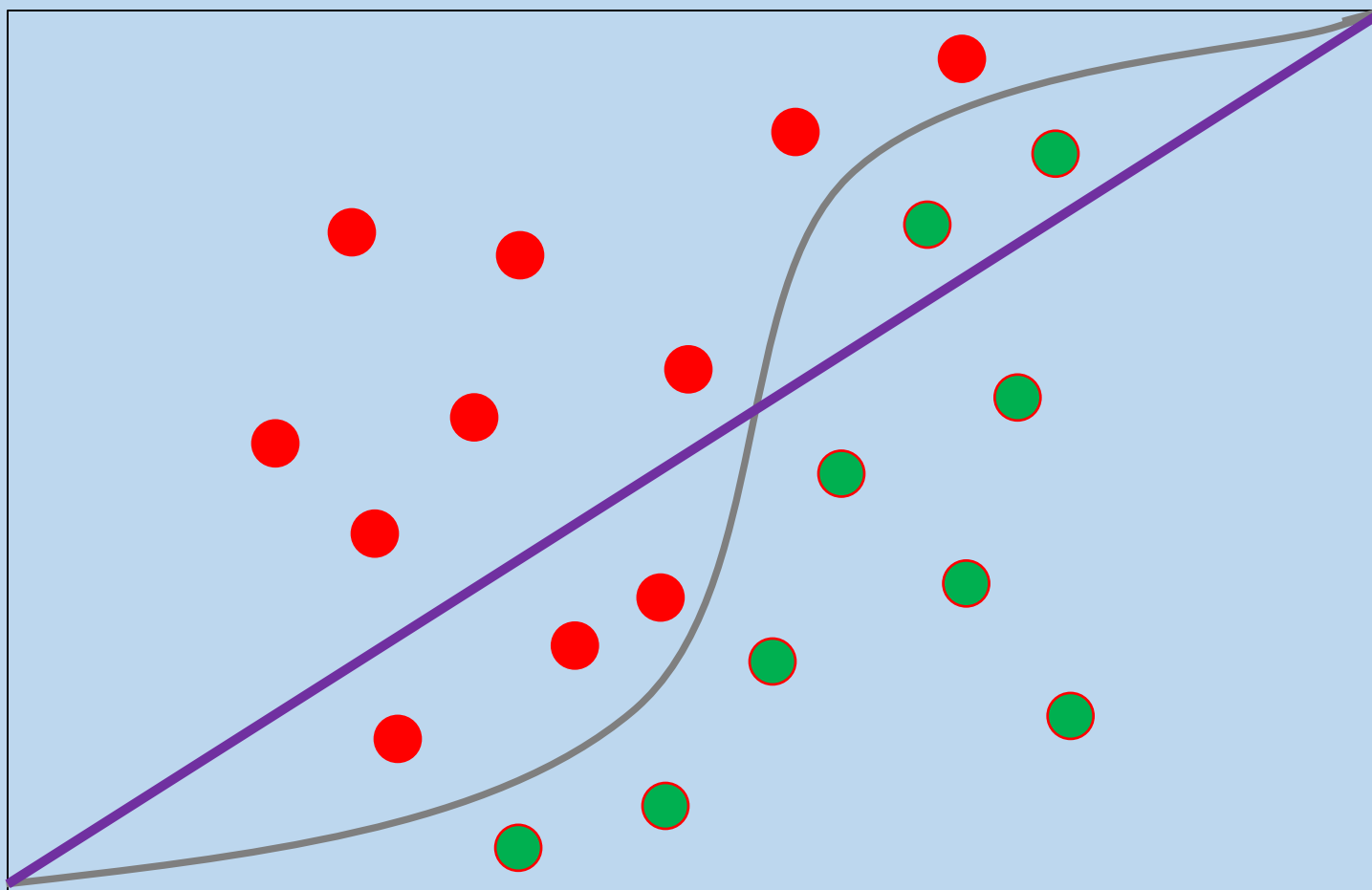
Neural Nets

Khasha Dehnad

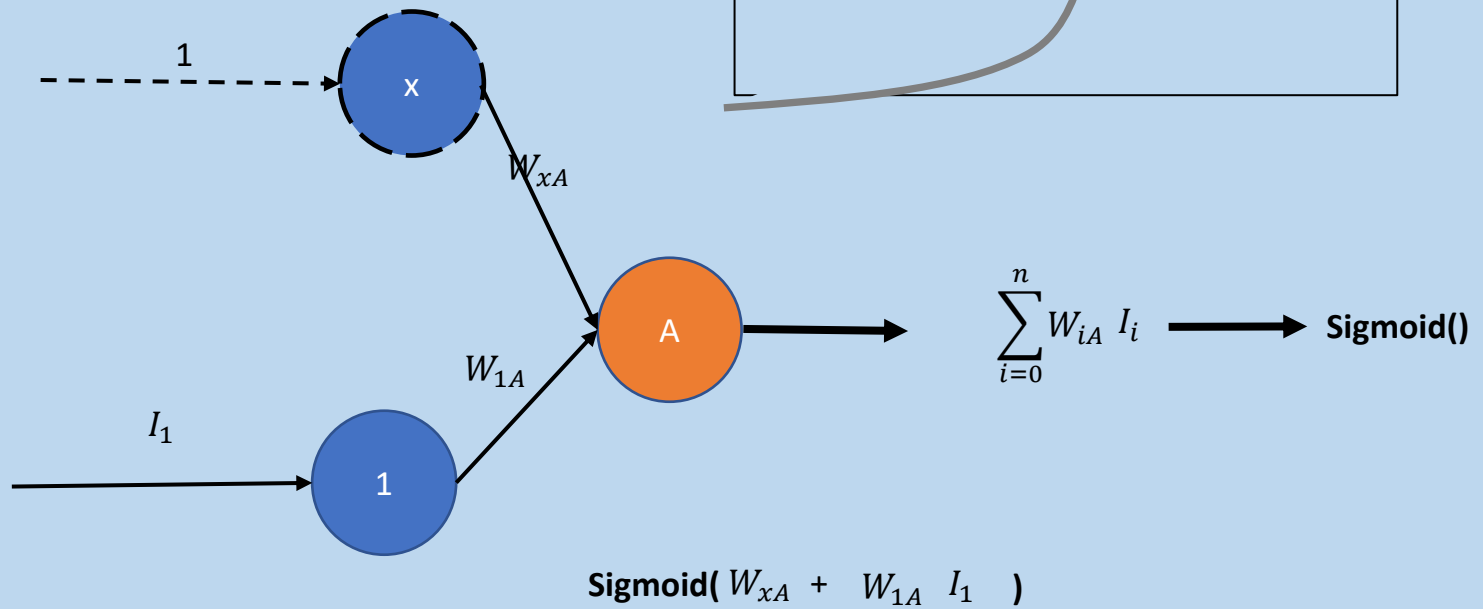
Non-vertical separator (SVM)



Sigmoid Separator

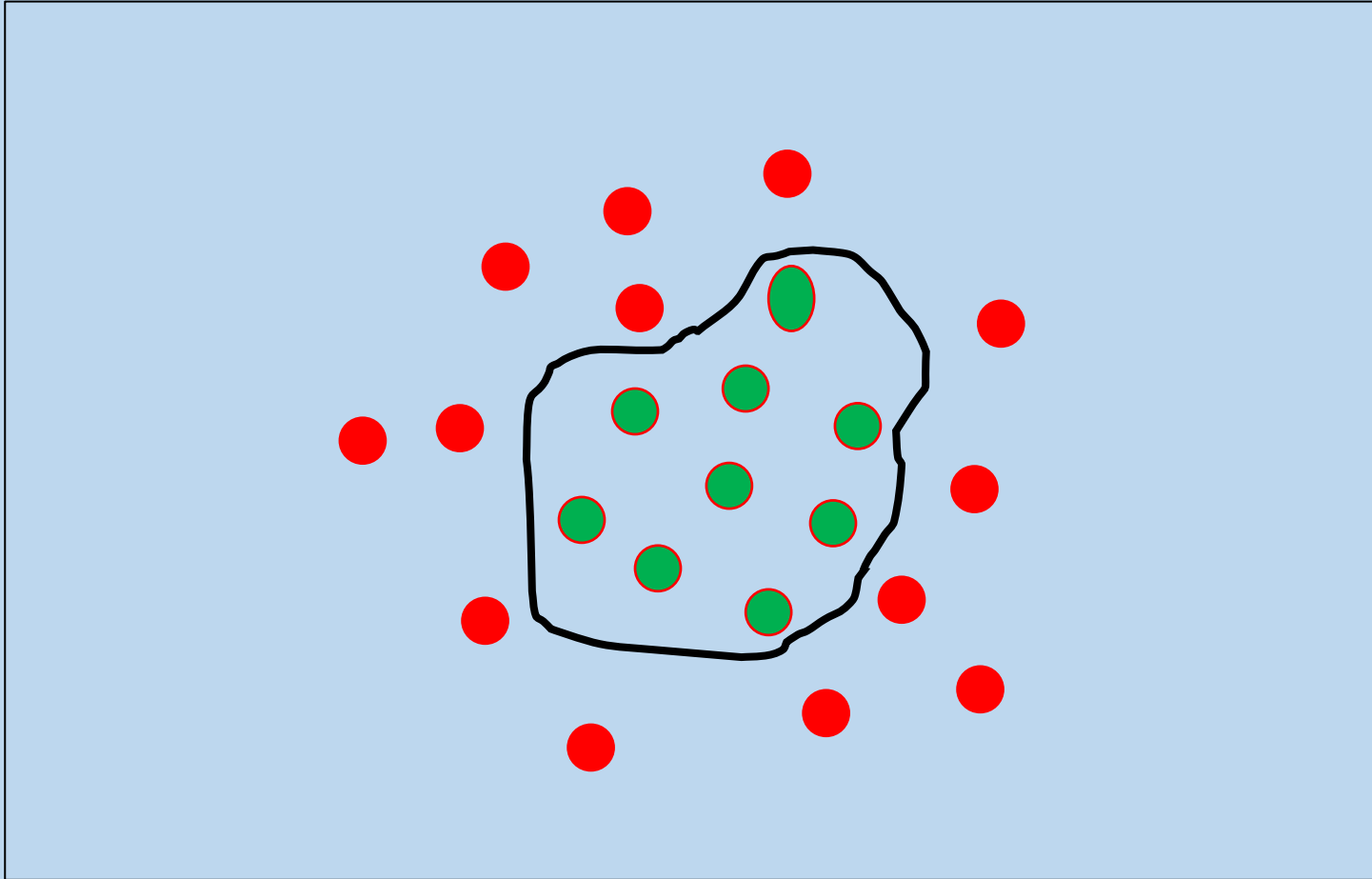


Perceptron-



$$f(\text{net} - z) = 1 / (1 + e^{-x})$$

Sigmoid Separator



Neural Networks

Simulating the Brain to Solve Problems Artificial Neural Networks (ANN)

Overview

- **Computer emulation of biological neural systems for building models:**
 - initially theorized in 1943 by McCulloch and Pitts of University of Chicago,
 - simulates the brain's cognitive learning process,
 - “learns” patterns directly from the data,
 - searches for complex relationships,
 - automatically builds models,
 - predicts - compares - adjusts,
 - corrects the model's mistakes over and over again,
 - **input: Data,**
 - **output: Prediction**
 - **tool: the Model “learned” from the Data.**

Neural Networks

Approximate Number of Neurons

Human Brain: 100 Billion (10^{11}) Neurons

Fruit fly: 100,000 Neurons

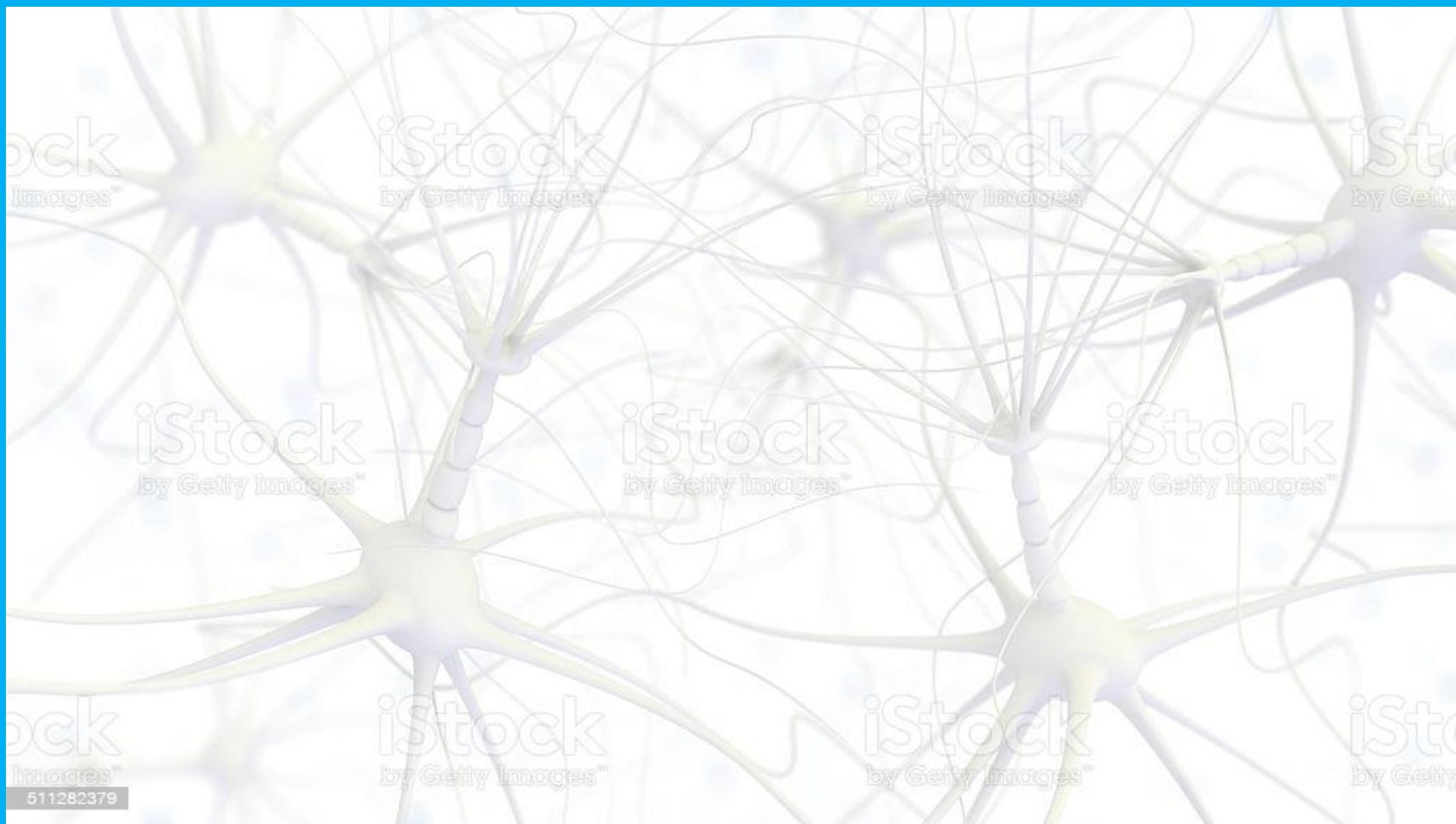
Nematode worm: 302 Neurons

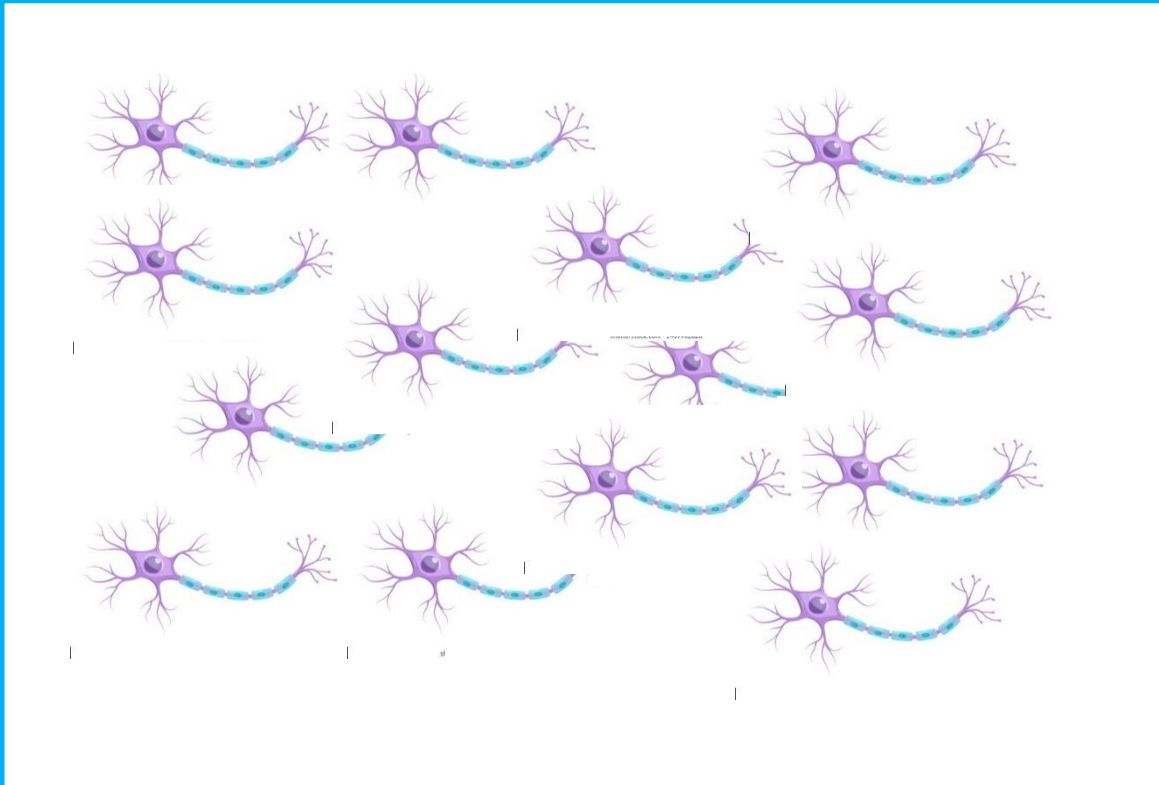
—

The idea of neurons as the structural constituent of the brain was first introduced by Ramon y Cajal (French) 1911

• Human Brain:

- a network of individual but interconnected nerve cells called *neurons* (10^{11} *neurons*)
- neurons are connected to each other via huge number of so-called *synapses* (10^{15} synapses or connections),
- a given neuron is connected to 10 thousand other neurons by these synapses,
- neurons can receive information from the outside world at various points in the network,
- these pieces of information are called *stimuli*,
- a neuron transfers information on to other neurons by firing chemicals called *neurotransmitters*,
- these transfers occur over synapses like bursts of electricity,
- the more important a particular stimulus is, the stronger the burst will be at the synapses,
- the information received by a nerve cell at one of the synapses either *excite* or *inhibit* the cell,
- if the receiving cell is excited, it will pass the information to other neurons,
- if the receiving cell is inhibited, it will damp the impact of the information,
- each nerve cell processes the raw input but passes it on only if it is important,
- the information travels through the network by generating new internal signals,
- the stimuli are processed by brain and nervous system and ultimately a response is produced.

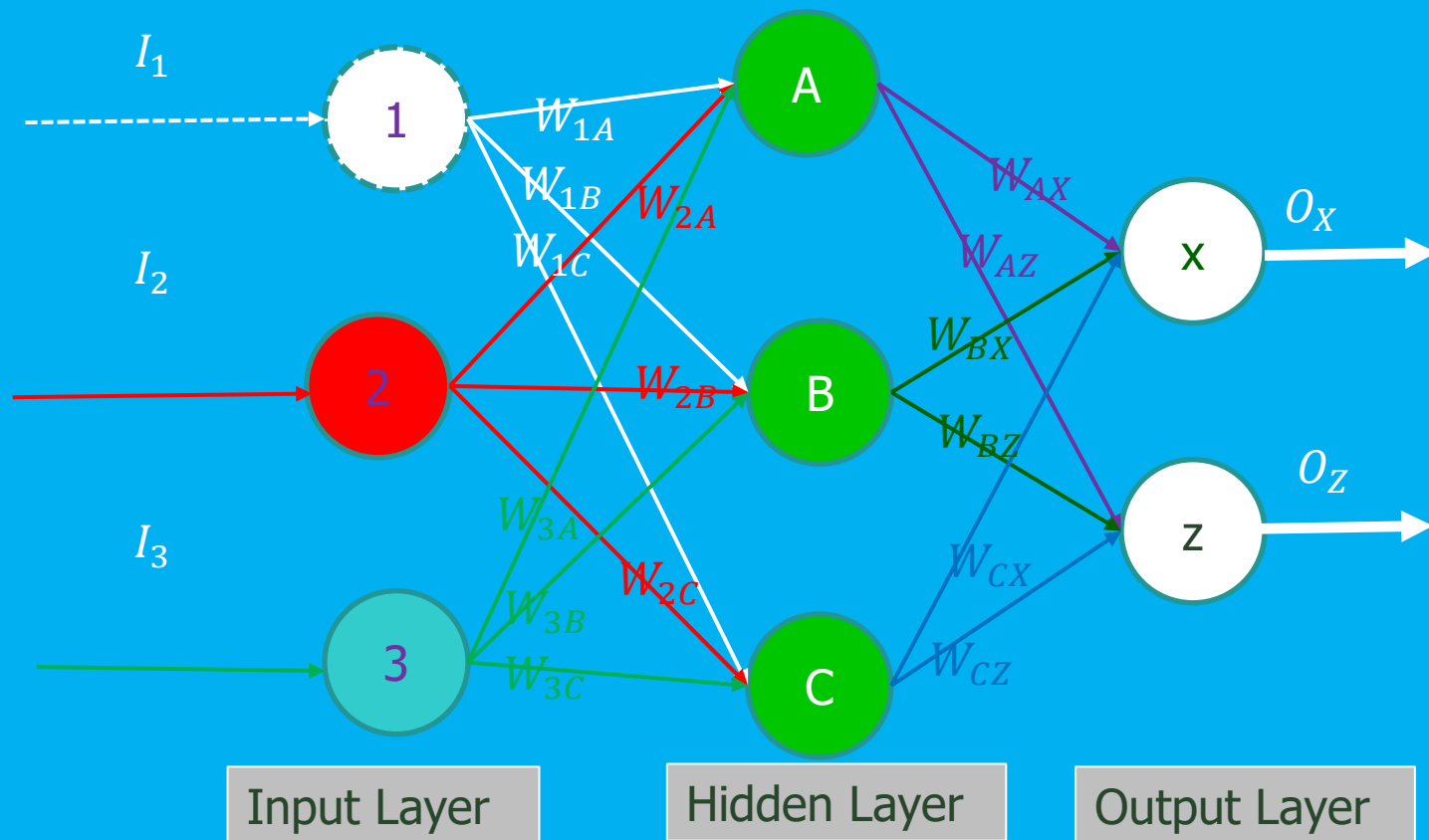




Neural Networks

Artificial Neural Networks

- a system of **neurodes** (*nodes*) and *weighted connections* (synapses) inside the memory of a computer,
- nodes are data storage locations (like variables in a program, cells in a spreadsheet),
- nodes are arranged in *layers* with weighted connections running between layers,
- *balls* represent nodes and *lines* represent connection weights,
- *input* layer nodes receive the data,
- *output* layer nodes relay the response of the neural network out of the net,



Neural Networks

ANN (Continued)

- *hidden* layer nodes (hidden from the outside world) conduct the internal processing,
- data are fed into the net through the input nodes,
- data are processed internally by hidden nodes, based on the inter-node connection weights,
- result are passed on to the outside world by output nodes,
- “learning” takes place through adjusting connection weights,
- a “learned” neural network has adjusted its weights properly

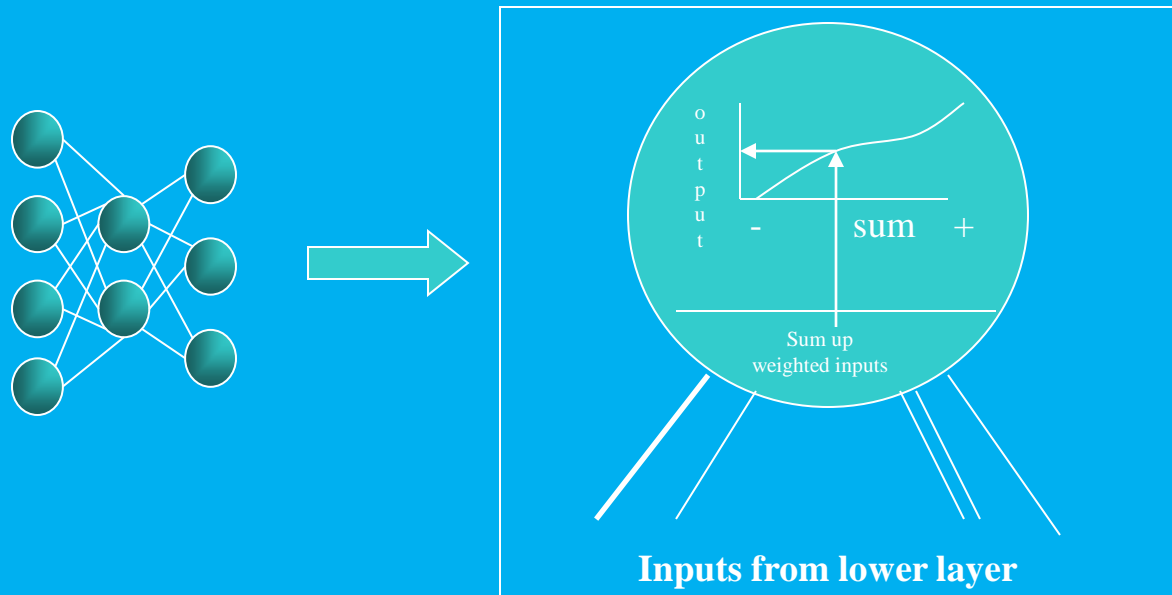
ANN operates in the same way as the biological model on which it is based.

Neural Networks

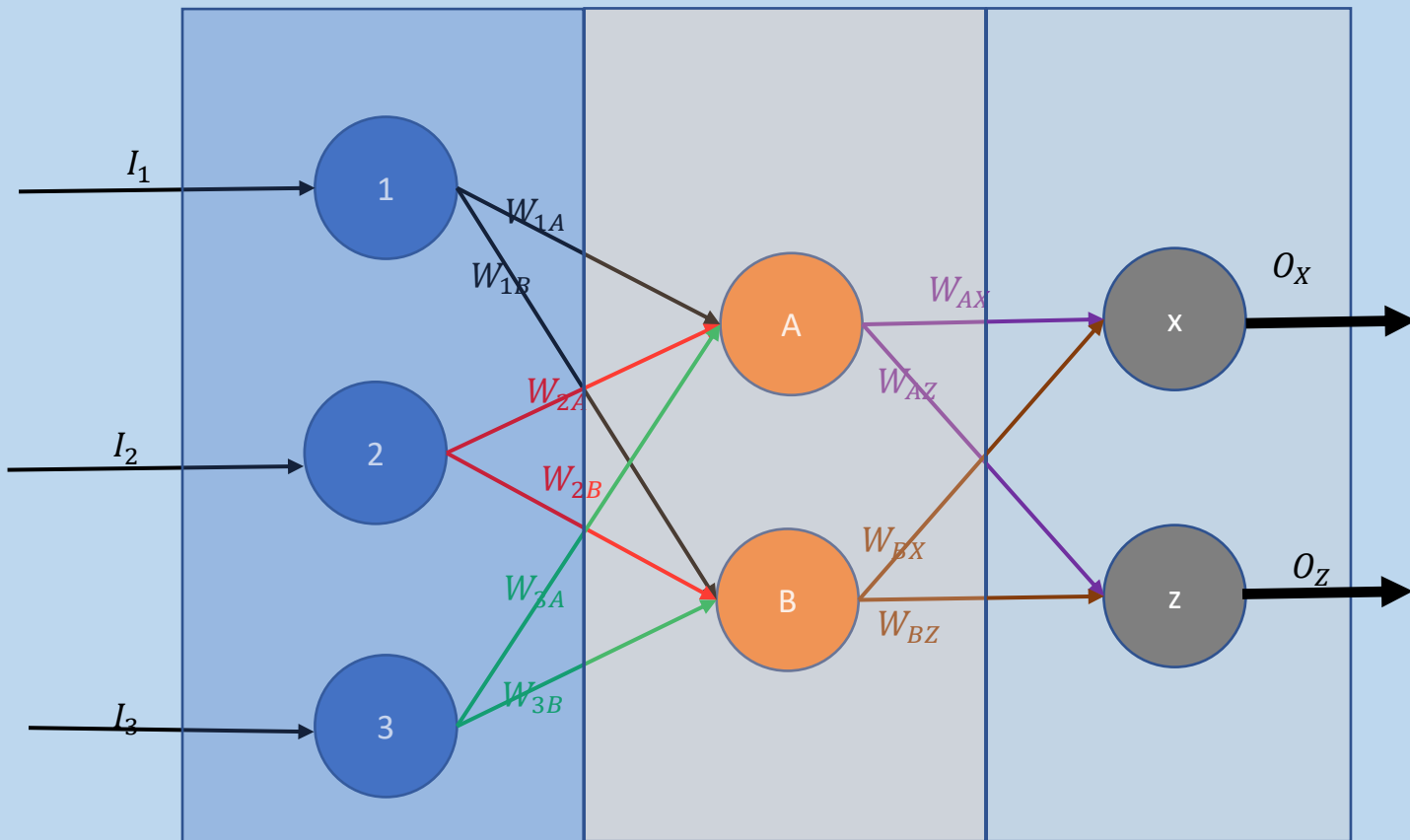
Application of a Learned Neural Network

- **Integration Function:**

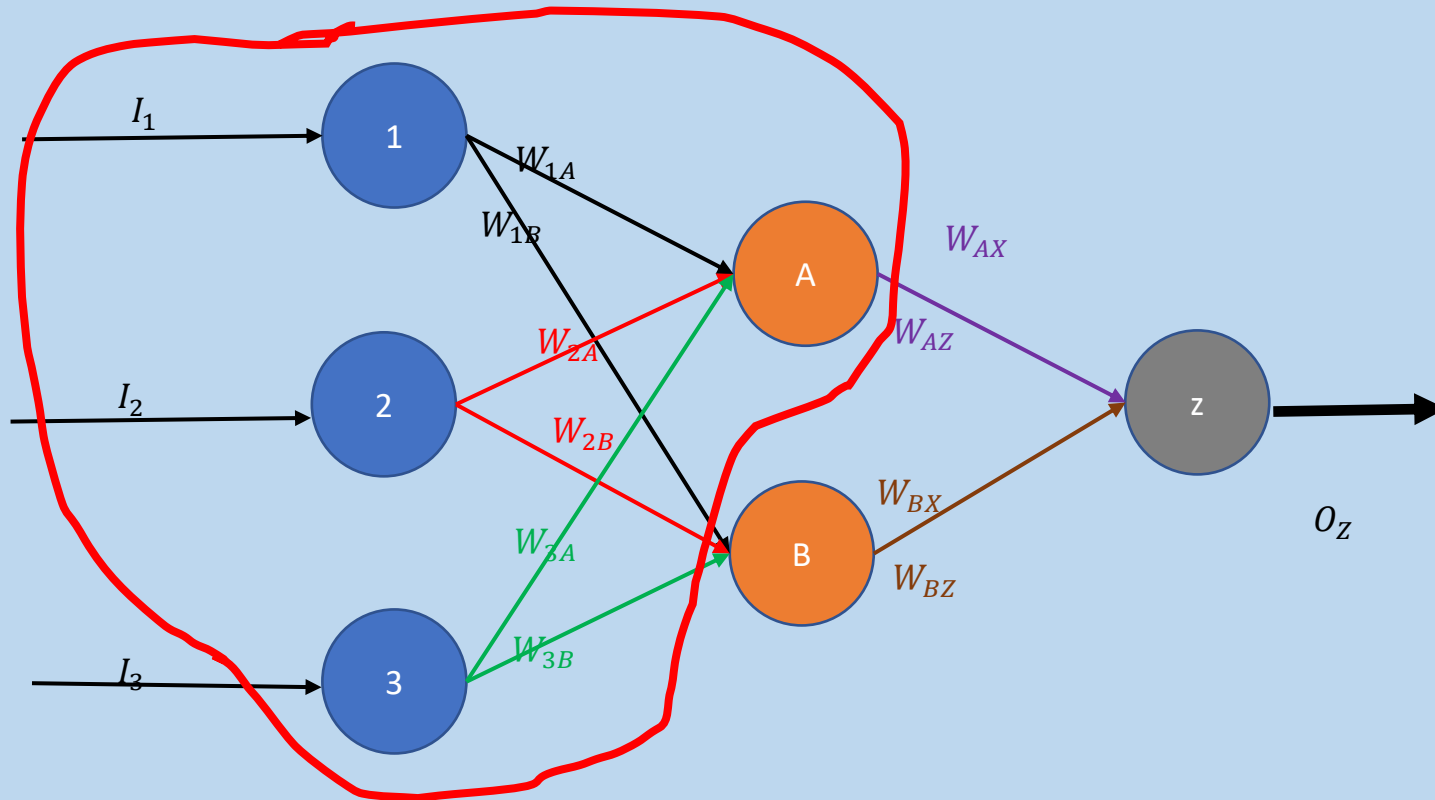
- each neuron receives a set of raw data (input),
- the neuron multiplies each input by the connecting weight leading into it,
- connection weight determines the importance of a given input in contribution to the output of the neuron,
- more important inputs will have bigger weights and less important ones will have smaller weights,
- the *integration function* of the neurode calculates a weighted sum of all inputs.



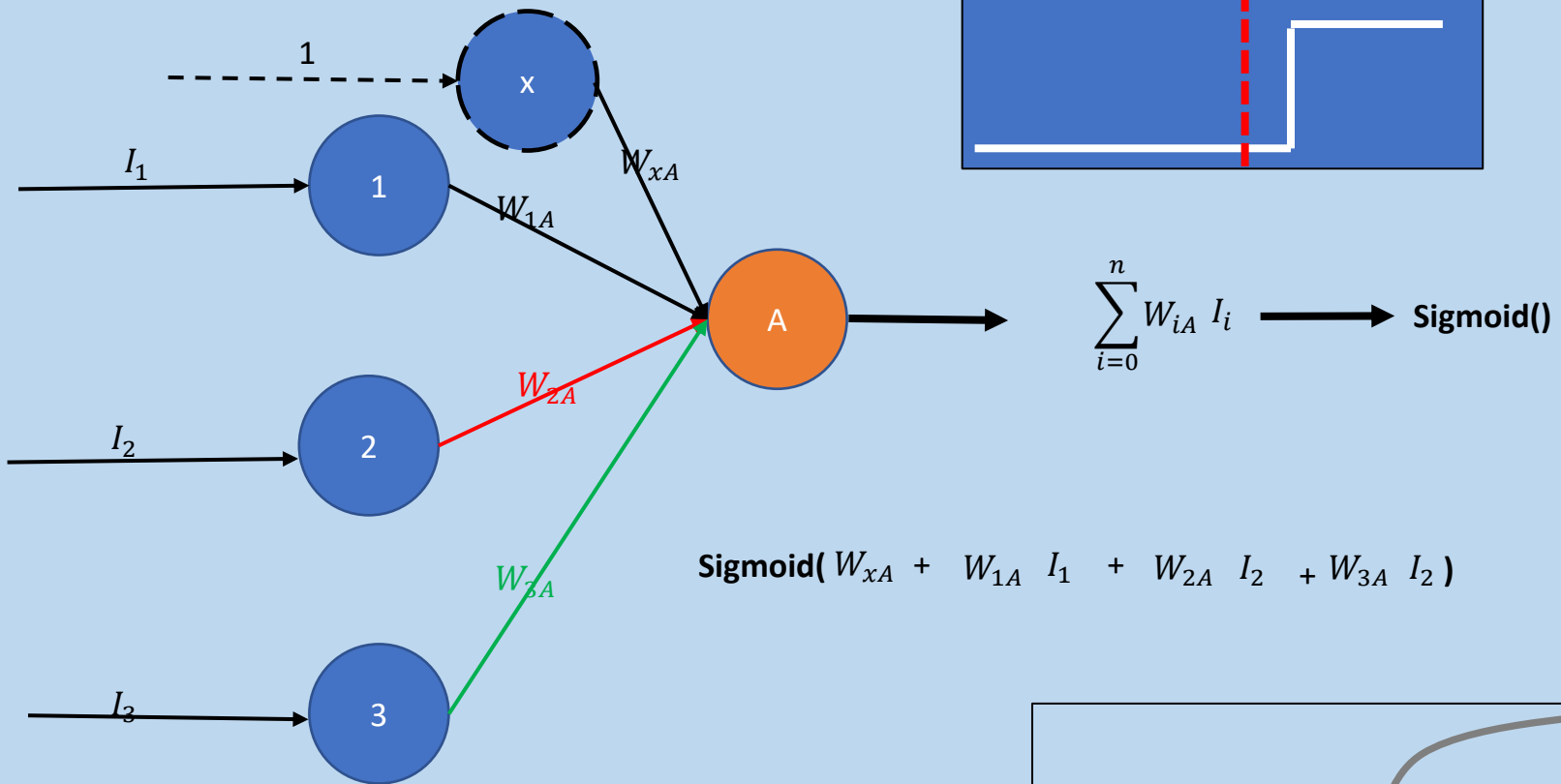
Neural Network with three Layers



Neural Net with One Node in the Output Layer

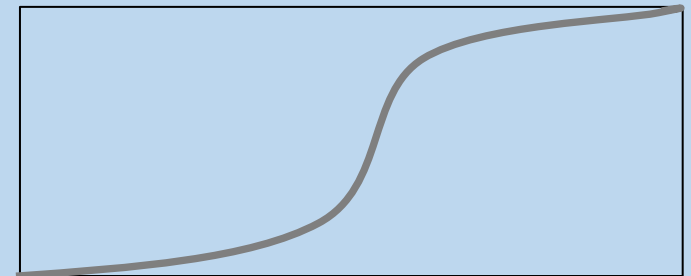


Perceptron-



$$\text{Sigmoid}(W_{xA} + W_{1A} I_1 + W_{2A} I_2 + W_{3A} I_2)$$

$$f(\text{net} - z) = 1 / (1 + e^{-x})$$



Neural Networks

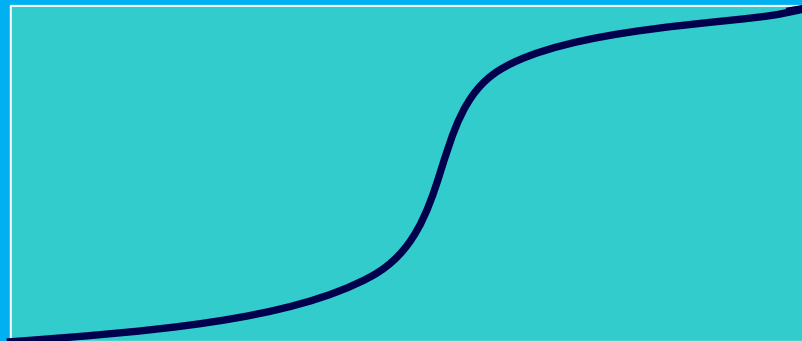
Application of a Learned Neural Network (Continued)

- **Transfer Function:**

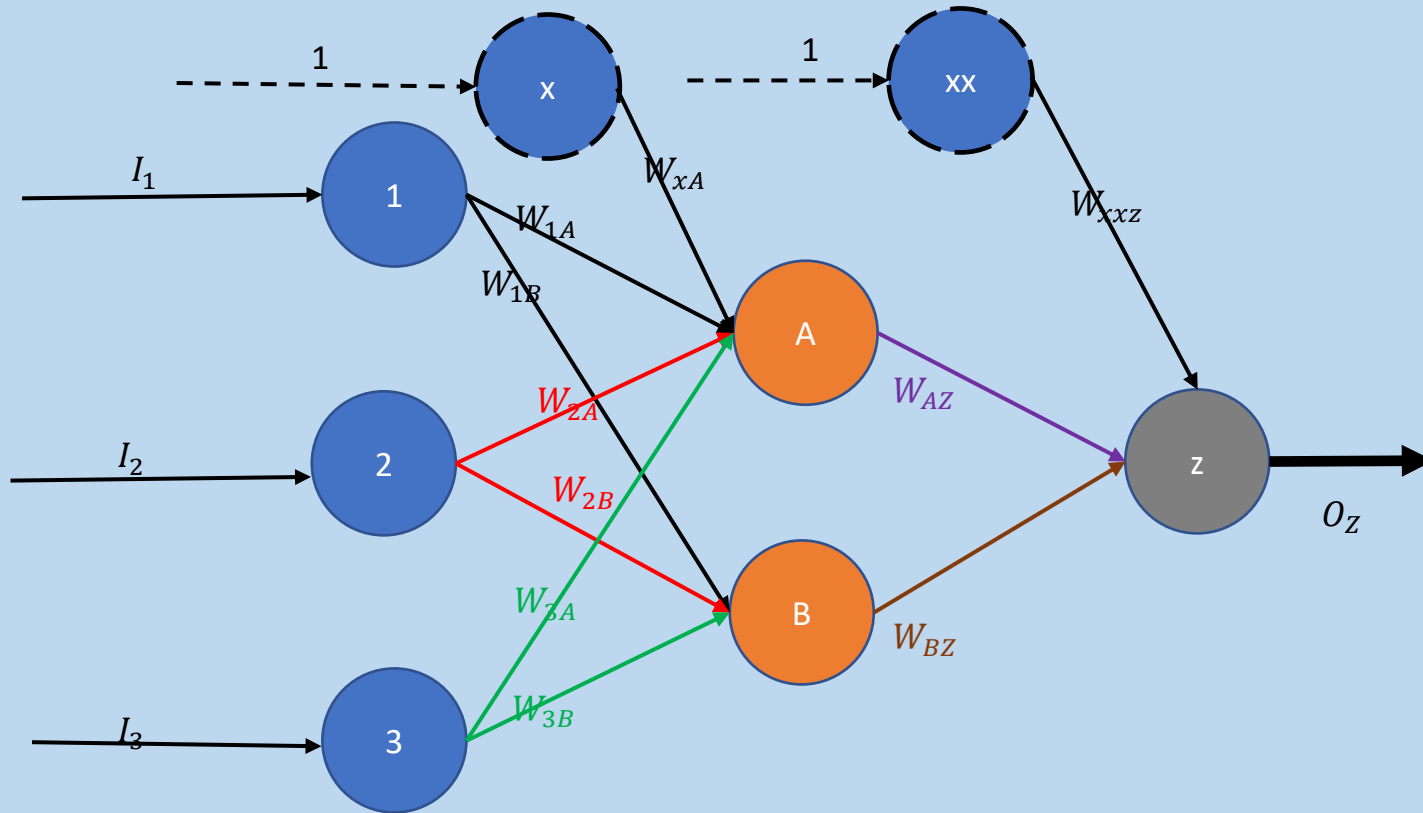
- the weighted sum is converted into an output value using a mathematical function called *transfer function*,
- transfer function normalizes the output into the range of [0,1],
- it serves as a kind of “dimmer” switch for turning the neuron “on” and “off”,
- the transfer function’s value will be *high* (excited) when the sum of the inputs is large & positive; and *low* (inhibited) when the sum is large negative,
- the transfer function determines the degree at which a given sum will cause a neurode to fire.

$$\sigma' = \sigma (1 - \sigma)$$

$$f(\text{net} \cdot \mathbf{z}) = 1 / (1 + e^{-x})$$



Neural Net with Dummy nodes



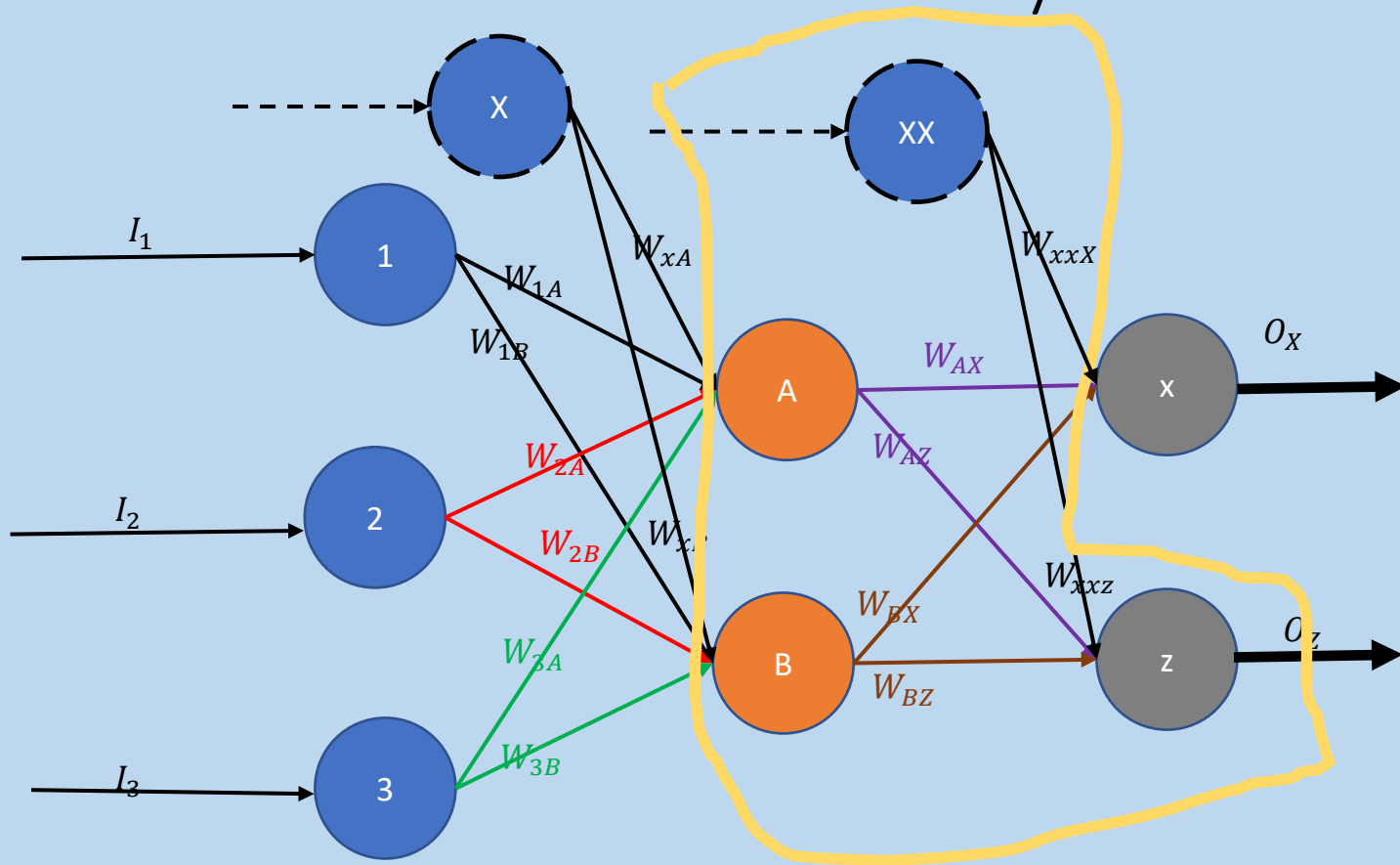
Matrix Representation of the Input-Hidden Layer

$$\text{Sigmoid}\left(\begin{bmatrix} 1 & I_1 & I_2 & I_3 \end{bmatrix} \begin{bmatrix} W_{xA} & W_{xB} \\ W_{1A} & W_{1B} \\ W_{2A} & W_{2B} \\ W_{3A} & W_{3B} \end{bmatrix} \right)$$

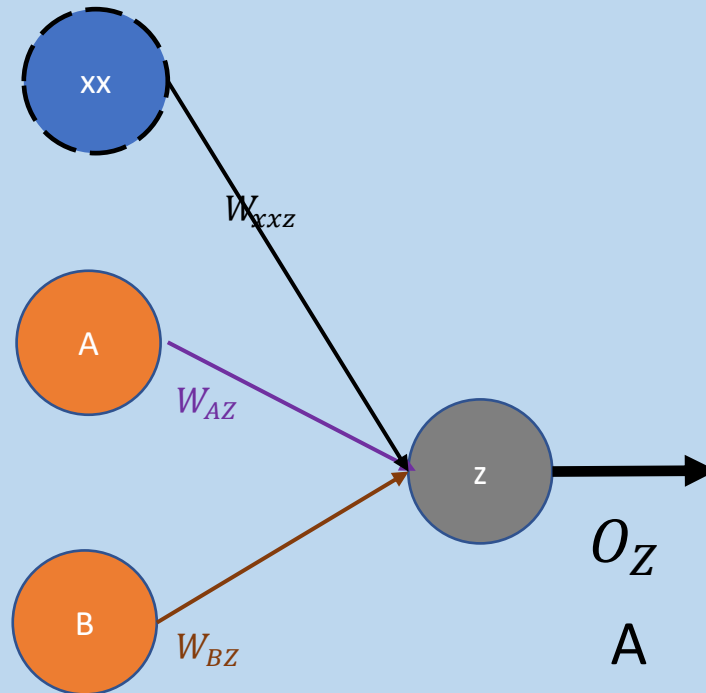
Matrix Representation of the Hidden-Output Layer

$$\text{Sigmoid}\left(\begin{bmatrix} 1 & o_A & o_B \end{bmatrix} \begin{bmatrix} W_{xxz} \\ W_{Az} \\ W_{Bz} \end{bmatrix} \right)$$

Neural Network with three Layers



Neural Net with Dummy nodes



$$e^2 = (A - O_Z)^2$$

Neural Net- Weight Adjustments

$$\frac{\partial e^2}{\partial W_{AZ}} = \frac{\partial e^2}{\partial o_Z} * \frac{\partial o_Z}{\partial \Sigma} * \frac{\partial \Sigma}{\partial W_{AZ}}$$

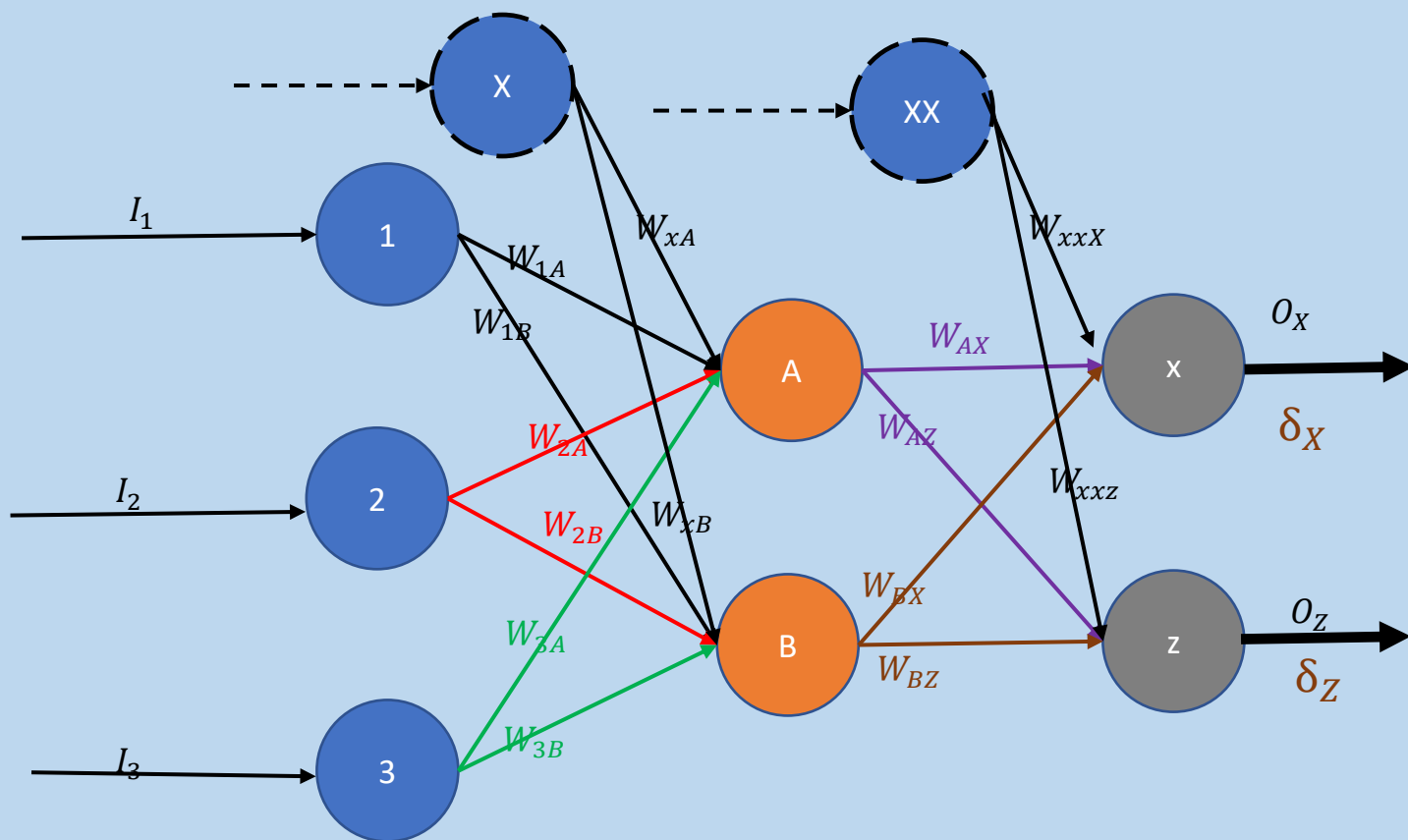
$$\frac{\partial (A - o_Z)^2}{\partial o_Z} = -2(A - o_Z) = -2e$$

$$\frac{\partial o_Z}{\partial \Sigma} = \frac{\partial \text{Sigmoid}(W_{xz} + W_{AZ} o_A + W_{BZ} o_B)}{\partial (W_{xz} + W_{AZ} o_A + W_{BZ} o_B)} = o_Z * (1 - o_Z)$$

$$\frac{\partial \Sigma}{\partial W_{AZ}} = \frac{\partial (W_{xz} + W_{AZ} o_A + W_{BZ} o_B)}{\partial W_{AZ}} = o_A$$

$$\Delta = \frac{\partial e^2}{\partial W_{AZ}} = -e * o_Z * (1 - o_Z) * o_A * -\eta$$

Neural Network with three Layers



$$e_A = \frac{W_{AZ}}{W_{xxZ} + W_{AZ} + W_{BZ}} \delta_Z + \frac{W_{AX}}{W_{xxX} + W_{AX} + W_{BX}} \delta_X$$

$$e_A = \frac{W_{AZ}}{W_{xxZ} + W_{AZ} + W_{BZ}} \delta_Z + \frac{W_{AX}}{W_{xxX} + W_{AX} + W_{BX}} \delta_X$$

$$e_B = \frac{W_{BZ}}{W_{xxZ} + W_{AZ} + W_{BZ}} \delta_Z + \frac{W_{BX}}{W_{xxX} + W_{AX} + W_{BX}} \delta_X$$

$$\begin{pmatrix} e_A \\ e_B \end{pmatrix} = \begin{pmatrix} W_{AZ} & W_{AX} \\ W_{BZ} & W_{BX} \end{pmatrix} * \begin{pmatrix} \delta_Z \\ \delta_X \end{pmatrix} = \begin{pmatrix} W_{AZ} \delta_Z + W_{AX} \delta_X \\ W_{BZ} \delta_Z + W_{BX} \delta_X \end{pmatrix}$$

Matrix Representation of the Hidden-Output Layer

$$W_{ij}New = W_{ij}Current + \Delta w_{ij}$$

$$\Delta W_{ij} = \eta \delta_j X_{ij}$$

$$\delta_j =$$

$$output_j (1 - output_j) (actual_j - output_j)$$

$$output_j (1 - output_j) \sum W_{jk} \delta_j$$

Output Nodes

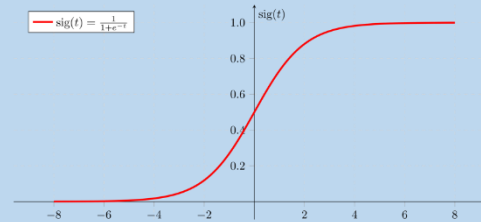
Hidden Nodes

$$\frac{\partial e^2}{\partial W_{AZ}} = - \overbrace{e * O_Z * (1 - O_Z)}^{\delta_j} * \overbrace{O_A}^{X_{ij}} * -\eta$$

$$W_{JK} \delta_j \Rightarrow \begin{pmatrix} W_{AZ} \delta_Z + W_{AX} \delta_X \\ W_{BZ} \delta_Z + W_{BX} \delta_X \end{pmatrix}$$

Appendix

Sigmoid function : First derivative



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

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

$$= \frac{d}{dx} (1 + e^{-x})^{-1} = -(1 + e^{-x})^{-2} \cdot \frac{d}{dx} (1 + e^{-x})$$

$$= -(1 + e^{-x})^{-2} \cdot \left(\frac{d}{dx} [1] + \frac{d}{dx} [e^{-x}] \right) = -(1 + e^{-x})^{-2} \cdot (0 + \frac{d}{dx} [e^{-x}])$$

$$= -(1 + e^{-x})^{-2} \cdot (e^{-x} \cdot \frac{d}{dx} [-x]) = -(1 + e^{-x})^{-2} \cdot (e^{-x} \cdot -\frac{d}{dx} [x])$$

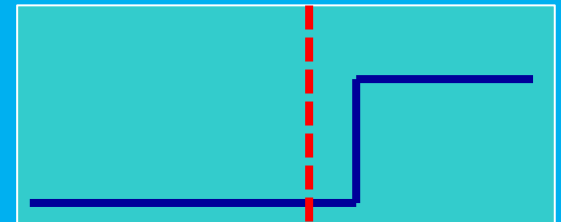
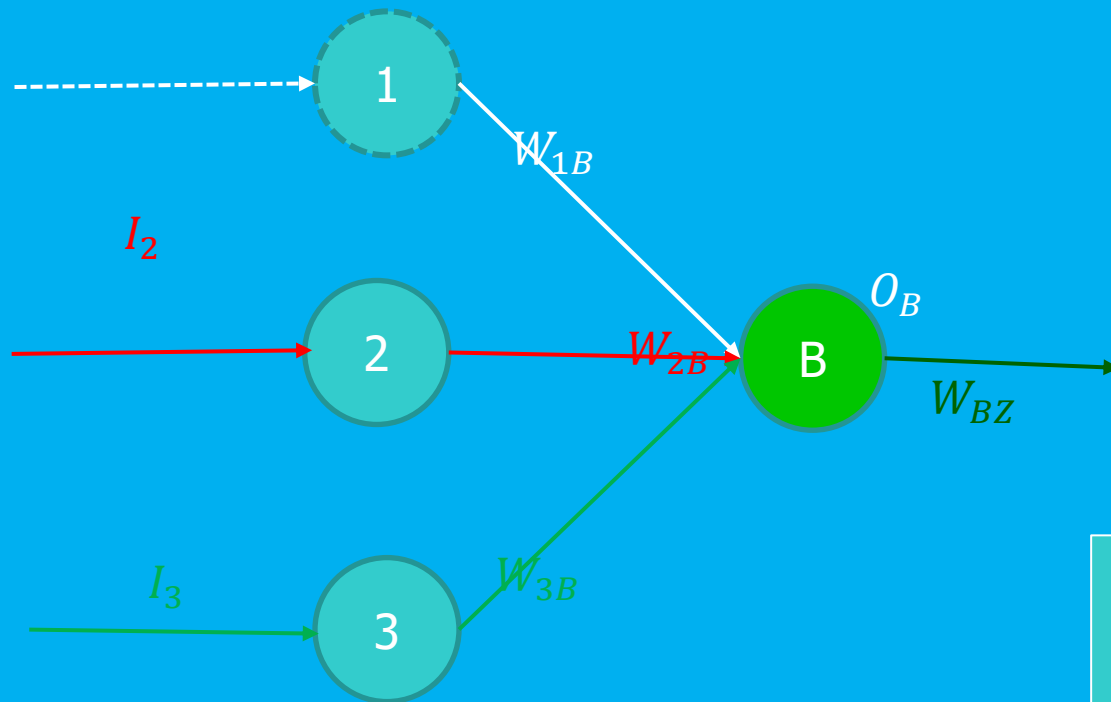
$$= -(1 + e^{-x})^{-2} \cdot (e^{-x} \cdot -1) = (1 + e^{-x})^{-2} \cdot e^{-x}$$

First derivative of sigmoid function

$$\begin{aligned} &= -(1 + e^{-x})^{-2} \cdot (e^{-x} \cdot -1) = (1 + e^{-x})^{-2} \cdot e^{-x} \\ &= \frac{1 \cdot e^{-x}}{(1 + e^{-x}) \cdot (1 + e^{-x})} = \frac{1}{(1 + e^{-x})} \cdot \frac{e^{-x}}{(1 + e^{-x})} \\ &= \frac{1}{(1 + e^{-x})} \cdot \frac{e^{-x} + 1 - 1}{(1 + e^{-x})} = \frac{1}{(1 + e^{-x})} \cdot \left(\frac{1 + e^{-x}}{1 + e^{-x}} - \frac{1}{1 + e^{-x}} \right) \\ &= \frac{1}{(1 + e^{-x})} \cdot \left(1 - \frac{1}{1 + e^{-x}} \right) = \sigma(x) \cdot (1 - \sigma(x)) \end{aligned}$$

$$I_1=1$$

$$W_{1B} * I_1 + W_{2B} * I_2 + W_{3B} * I_3 = O_B$$



Input Layer

Hidden Node