Perceptron Algorithms

St. Joseph's University, Bengaluru

ML 2 BDA3321

Neurons

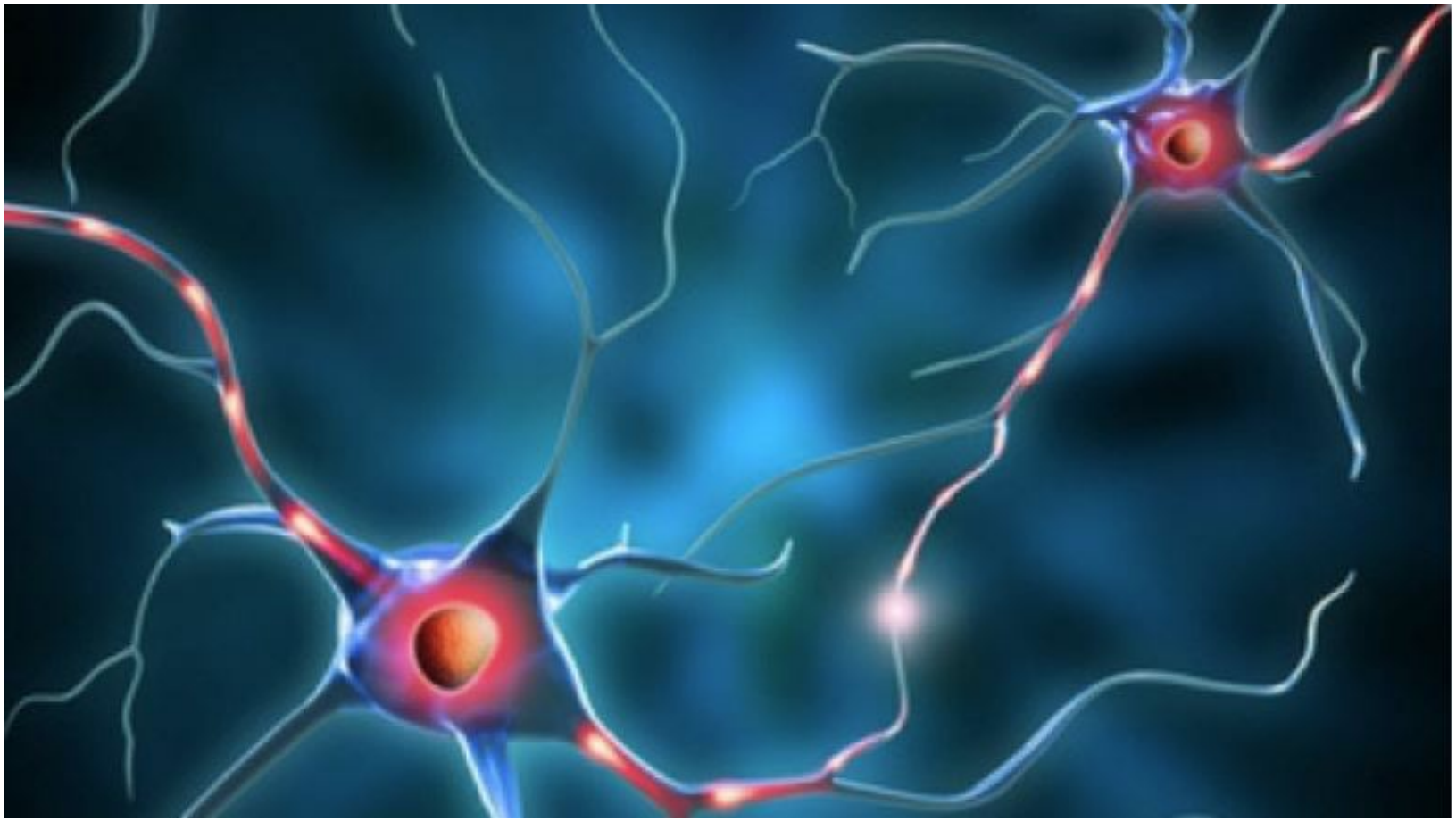

Figure: https://readbiology.com/types-of-neurons/

Neorons

McCulloch and Pitts Neurons

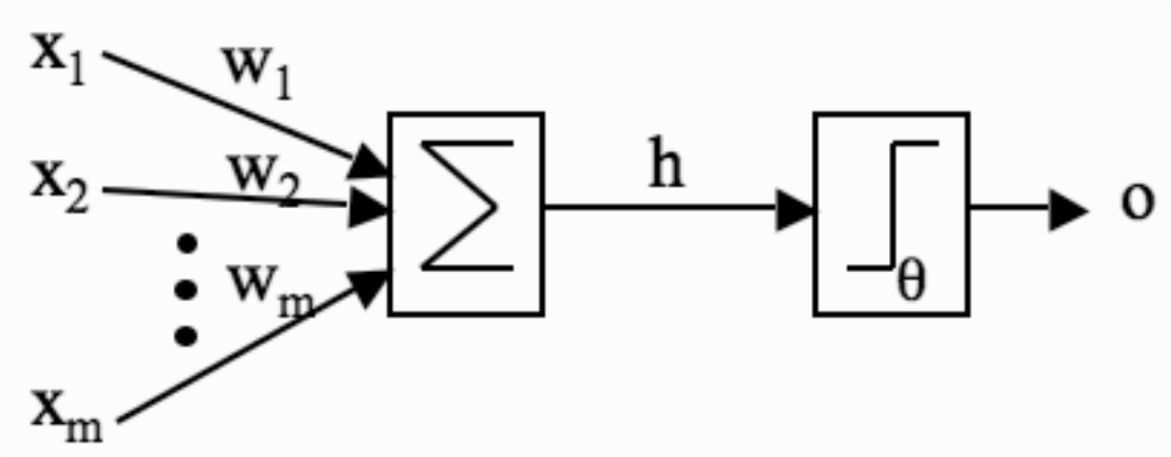

Figure: Neuron, [Mar14]

1. Weighted Inputs
2. Adder
3. Activation Function

McCulloch and Pitt Neurons

Components

Weighted Inputs

For each iput node, x_i a weight w_i is assinged.

Adder

$h = \sum_{i=1}^{m} x_i w_i$

Activation Function and Output

A threshold function decided if the neoron fires.

$$o = g(h) = \begin{cases} 1 \text{ if } h > \theta \\ 0 \text{ if } h \leq \theta \end{cases}$$

Neural Networks

Perceptron

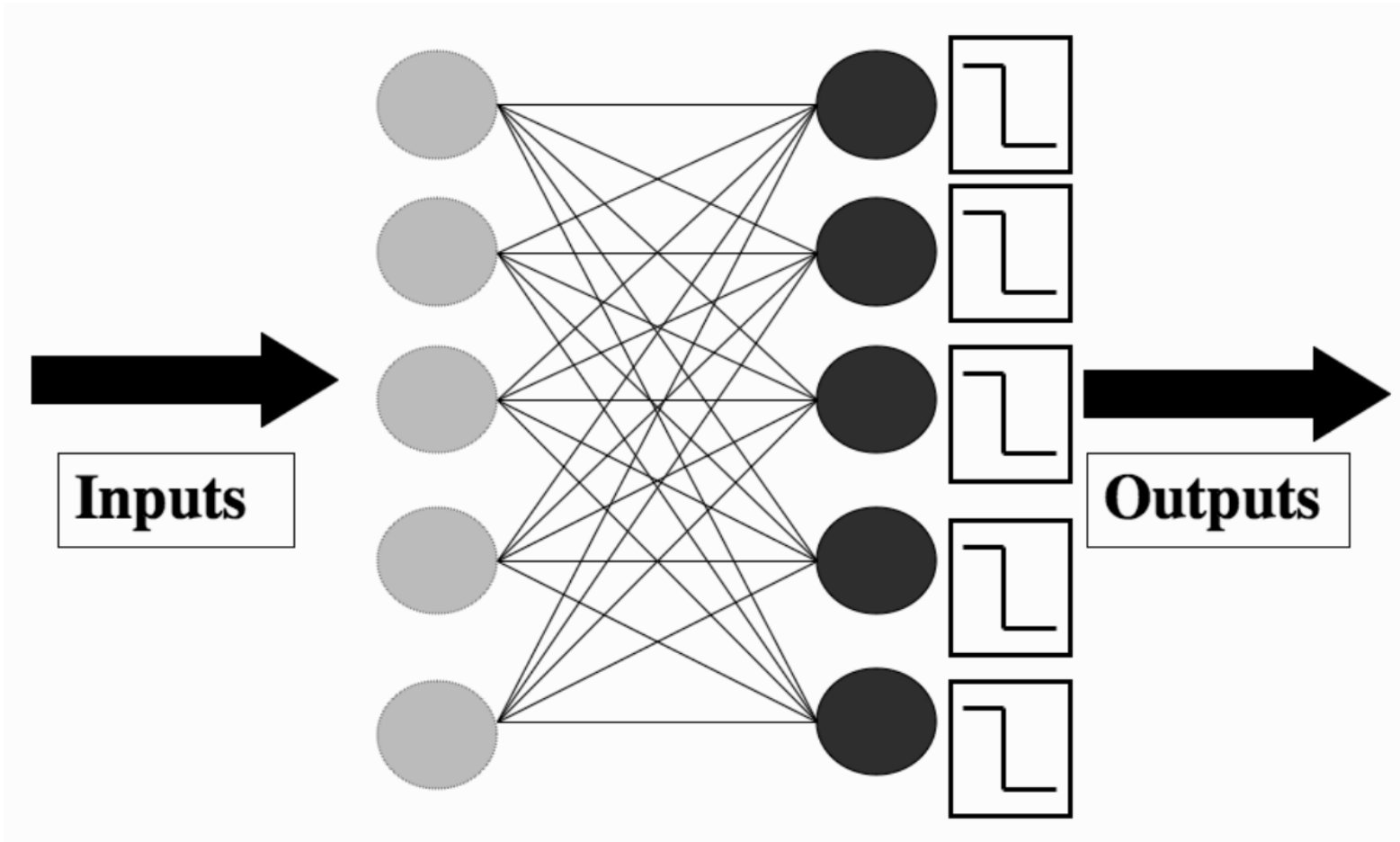

Figure: Perceptron, [Mar14]

Perceptron

- We label weights for neurons as w_{ij}; where i is the index running over number of inputs, and j index runs over number of neurons.
- To check if a neuron fires or not, we use adder and activation function as defined on slide 4.
- The output is a vector of $0s$ and $1s$.
- Compare this pattern to the target.

Perceptron

Training the perceptron

Learning Rate

$$w_{ij} \leftarrow w_{ij} - \eta(y_j - t_j) \cdot x_i$$

Bias Input

Add a fixed non zero input weight.

Perceptron

The Perceptron Algorithm

Initialize:

- Set all of the weights to small random numbers.

Training:

- For T itereations or until all outputs are correct

  For each input vector

  Compute the activation of each neuron j using

$$y_j = g(\sum_{i=0}^{m} w_{ij}x_i) = \begin{cases} 1 \text{ if } \sum_{i=1}^{m} w_{ij}x_i > 0 \\ 0 \text{ if } \sum_{i=1}^{m} w_{ij}x_i \leq 0 \end{cases}$$

  Update each weight individually using

$$w_{ij} \leftarrow w_{ij} - \eta(y_j - t_j) \cdot x_i.$$

Recall:

Compute the activation of each neuron j using:

$$y_j = g(\sum_{i=0}^{m} w_{ij}x_i) = \begin{cases} 1 \text{ if } w_{ij}x_i > 0 \\ 0 \text{ if } w_{ij}x_i \leq 0 \end{cases}$$

References I

[Mar14] Stephen Marsland. *Machine Learning, an algorithmic perspective*. CRC Press, 2014.