McCulloch-Pitts Neuron

Neuroinformatics Tutorial 2

Duc Duy Pham[1]

[1]Intelligent Systems, Faculty of Engineering,
University of Duisburg-Essen, Germany

Content

- Revision
- Biological Motivation for ANNs
- McCulloch-Pitts Neuron
- Tasks

Content

- **Revision**
- Biological Motivation for ANNs
- McCulloch-Pitts Neuron
- Tasks

Revision

- Which of the following statements are true?
  1. AI $\subset$ ML
  2. DL $\subset$ AI
  3. ML $\subset$ DL
  4. ANN $\subset$ DL
  5. ANN $\subset$ ML

A: 2, 4, 5

B: 2, 5

C: 1, 2, 3, 4

D: all

Revision

- Which of the following statements are true?
  1. AI $\subset$ ML
  2. DL $\subset$ AI
  3. ML $\subset$ DL
  4. ANN $\subset$ DL
  5. ANN $\subset$ ML

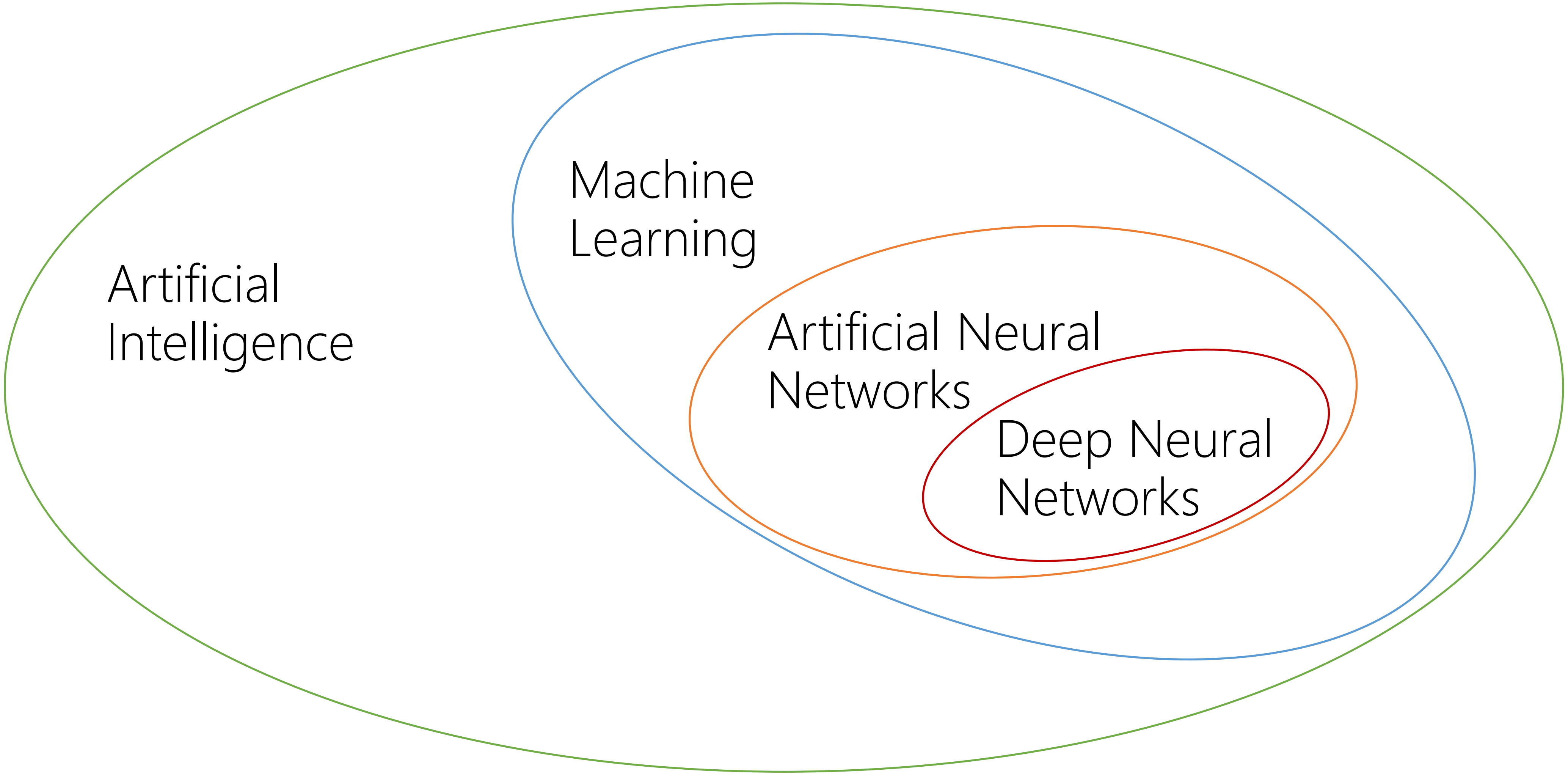
A: 2, 4, 5

B: 2, 5

C: 1, 2, 3, 4

D: all

Relation to AI



Revision

- What are tasks that can be addressed with ANNs?
    1. Classification
    2. Regression
    3. Image Synthesis

A: 1, 3

B: 1, 2

C: 2, 3

D: all

Revision

- What are tasks that can be addressed with ANNs?
    1. Classification
    2. Regression
    3. Image Synthesis

A: 1, 3

B: 1, 2

C: 2, 3

D: all

Revision

- Which statements regarding Artificial Neural Networks are true?
  1. ANNs are basically Turing Machines
  2. ANNs can work in parallel
  3. ANNs can have connections between all computing components

A: 1, 3

B: 1, 2

C: 2, 3

D: all

Revision

- Which statements regarding Artificial Neural Networks are true?
  1. ANNs are basically Turing Machines
  2. ANNs can work in parallel
  3. ANNs can have connections between all computing components

A: 1, 3

B: 1, 2

C: 2, 3

D: all

Revision

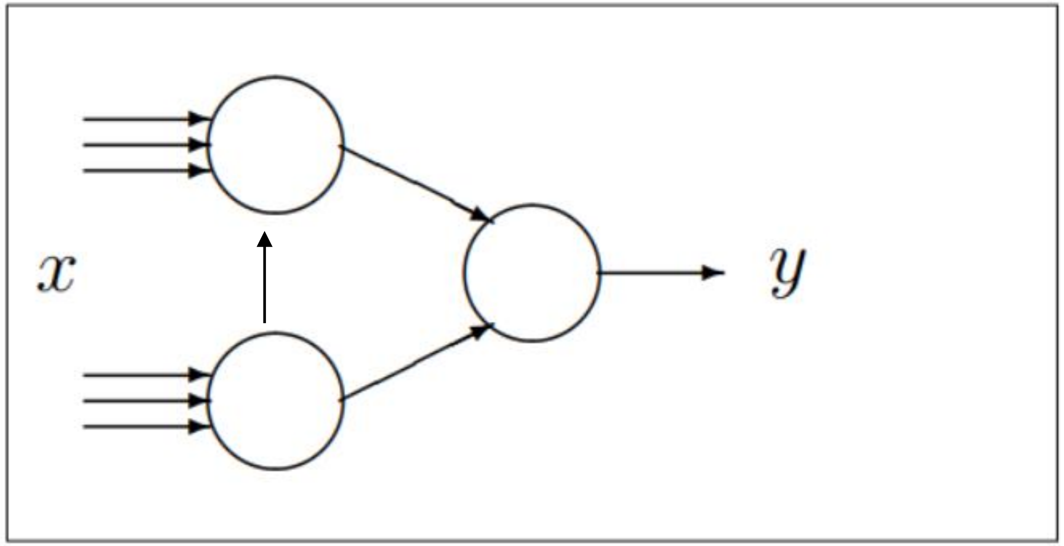
- Which statements regarding Artificial Neural Networks are true?
  1. ANNs are basically Turing Machines
  2. ANNs can work in parallel
  3. ANNs can have connections between all computing components

A: 1, 3

B: 1, 2

C: 2, 3

D: all



Revision

- How can an ANN be formalized according to the lecture?
  1. Cauchy-Schwartz Equation
  2. Quintuple
  3. It is not possible, since ANNs are black boxes
  4. As a dictionary

A: 1

B: 2

C: 3

D: 4

Revision

- How can an ANN be formalized according to the lecture?
  1. Cauchy-Schwartz Equation
  2. Quintuple
  3. It is not possible, since ANNs are black boxes
  4. As a dictionary

A: 1

B: 2

C: 3

D: 4

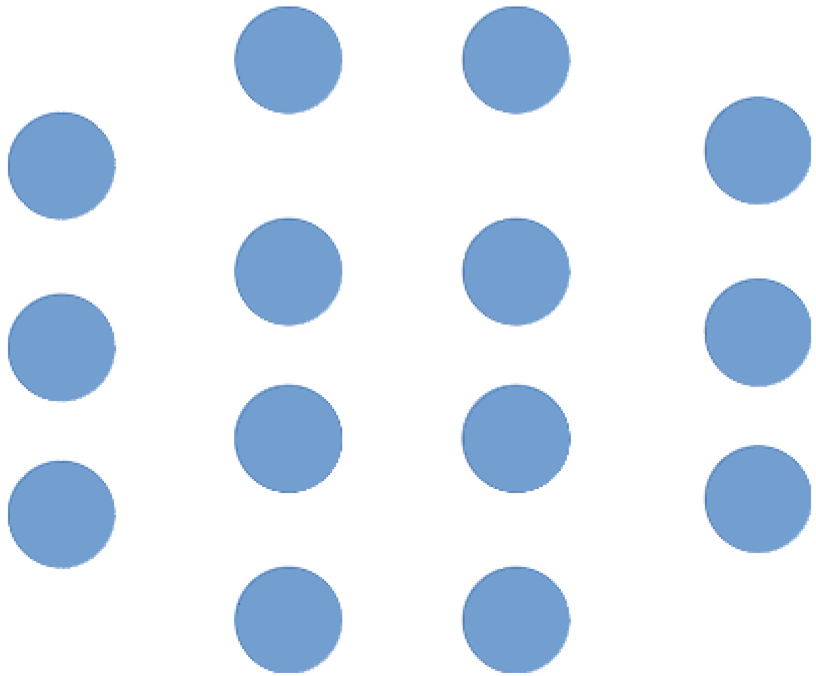
ANN Formalization

Quintupel $\mathcal{A} := (\mathcal{K}, \mathcal{V}, \mathcal{I}, \mathcal{O}, \mathcal{H})$

ANN Formalization

Quintupel $\mathcal{A} := (\mathcal{K}, \mathcal{V}, \mathcal{I}, \mathcal{O}, \mathcal{H})$

$\mathcal{K}$: Knotenmenge

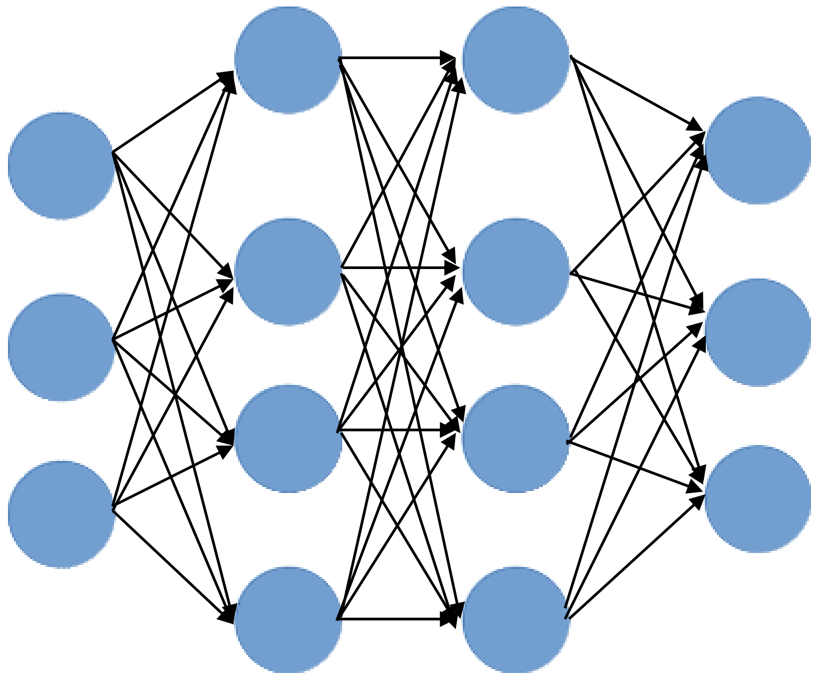


ANN Formalization

Quintupel $\mathcal{A} := (\mathcal{K}, \mathcal{V}, \mathcal{I}, \mathcal{O}, \mathcal{H})$

$\mathcal{K}$: Knotenmenge

$\mathcal{V}$: Kantenmenge $\mathcal{V} \subset \mathcal{K} \times \mathcal{K}$



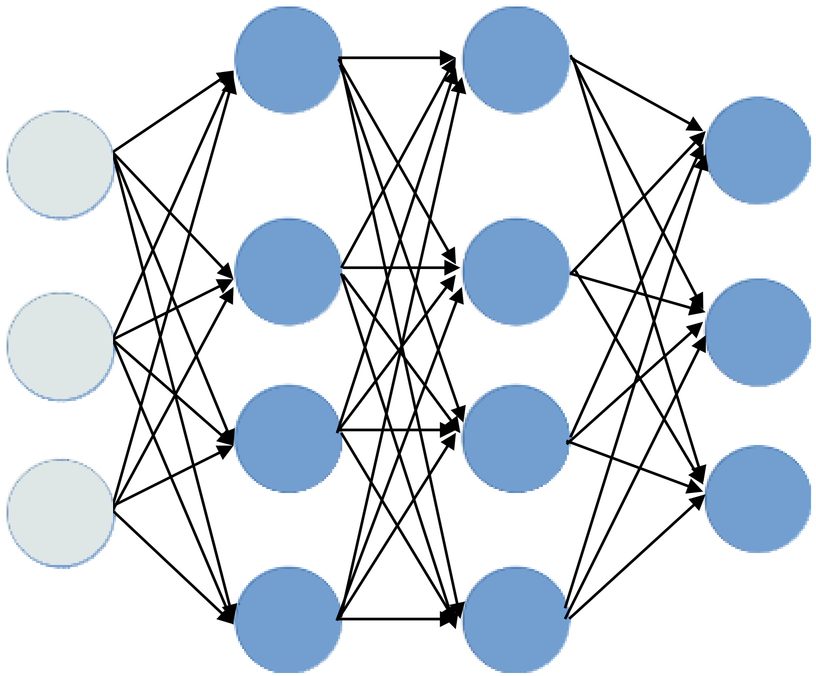
ANN Formalization

Quintupel $\mathcal{A} := (\mathcal{K}, \mathcal{V}, \mathcal{I}, \mathcal{O}, \mathcal{H})$

$\mathcal{K}$: Knotenmenge

$\mathcal{V}$: Kantenmenge $\mathcal{V} \subset \mathcal{K} \times \mathcal{K}$

$\mathcal{I}$: Eingabeknoten $\mathcal{I} \subset \mathcal{K}$



ANN Formalization

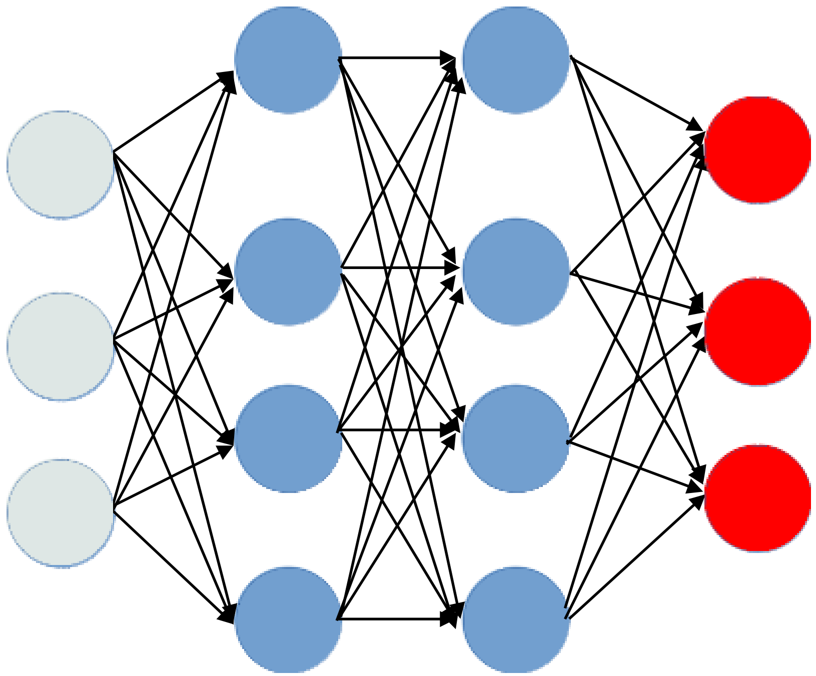
Quintupel $\mathcal{A} := (\mathcal{K}, \mathcal{V}, \mathcal{I}, \mathcal{O}, \mathcal{H})$

$\mathcal{K}$: Knotenmenge

$\mathcal{V}$: Kantenmenge $\mathcal{V} \subset \mathcal{K} \times \mathcal{K}$

$\mathcal{I}$: Eingabeknoten $\mathcal{I} \subset \mathcal{K}$

$\mathcal{O}$: Ausgabeknoten $\mathcal{O} \subset \mathcal{K}$



ANN Formalization

Quintupel $\mathcal{A} := (\mathcal{K}, \mathcal{V}, \mathcal{I}, \mathcal{O}, \mathcal{H})$

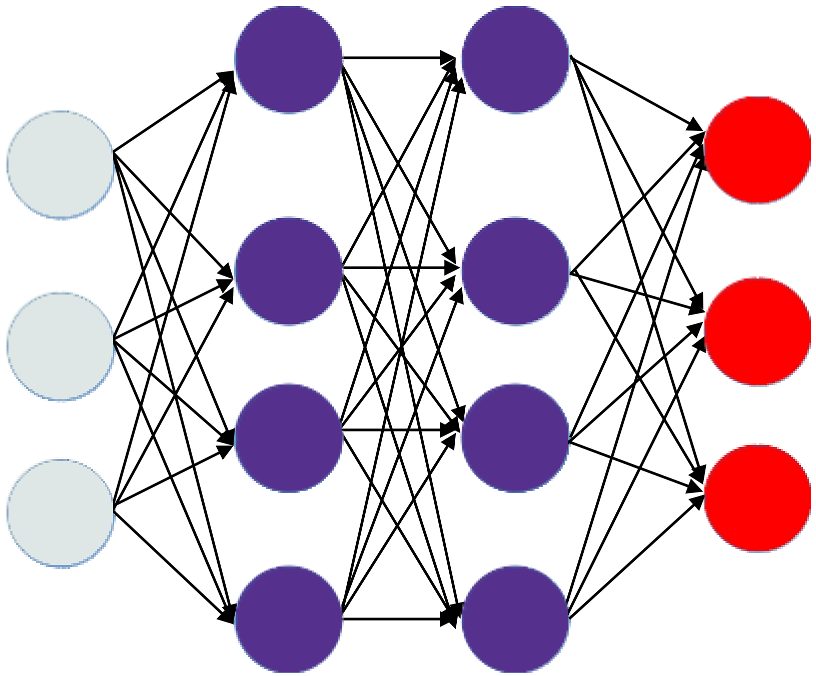
$\mathcal{K}$: Knotenmenge

$\mathcal{V}$: Kantenmenge $\mathcal{V} \subset \mathcal{K} \times \mathcal{K}$

$\mathcal{I}$: Eingabeknoten $\mathcal{I} \subset \mathcal{K}$

$\mathcal{O}$: Ausgabeknoten $\mathcal{O} \subset \mathcal{K}$

$\mathcal{H}$: Verdeckte Knoten: $\mathcal{H} := \mathcal{K} \backslash (\mathcal{I} \cup \mathcal{O})$



Revision

- What is the general purpose of the propagation function f_p?
    1. Weighted summation of incoming signals to one scalar
    2. Weighted fusion of incoming signals to one scalar
    3. Weighted product of incoming signals to a vector

A: 1, 2

B: all

C: 3

D: 2

Revision

- What is the general purpose of the propagation function f_p?
  1. Weighted summation of incoming signals to one scalar
  2. Weighted fusion of incoming signals to one scalar
  3. Weighted product of incoming signals to a vector

A: 1, 2

B: all

C: 3

D: 2

Revision

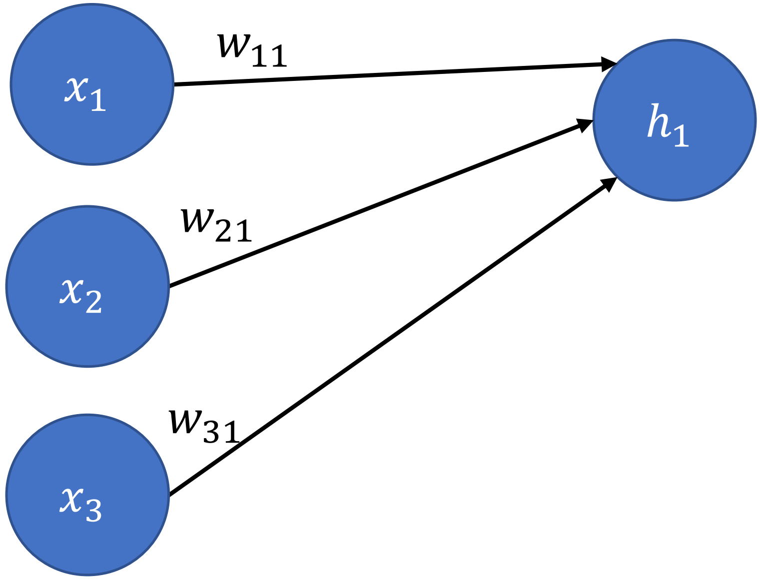
- What is the general purpose of the propagation function f_p?
    1. Weighted summation of incoming signals to one scalar
    2. Weighted fusion of incoming signals to one scalar
    3. Weighted product of incoming signals to a vector

A: 1, 2

B: all

C: 3

D: 2



Revision

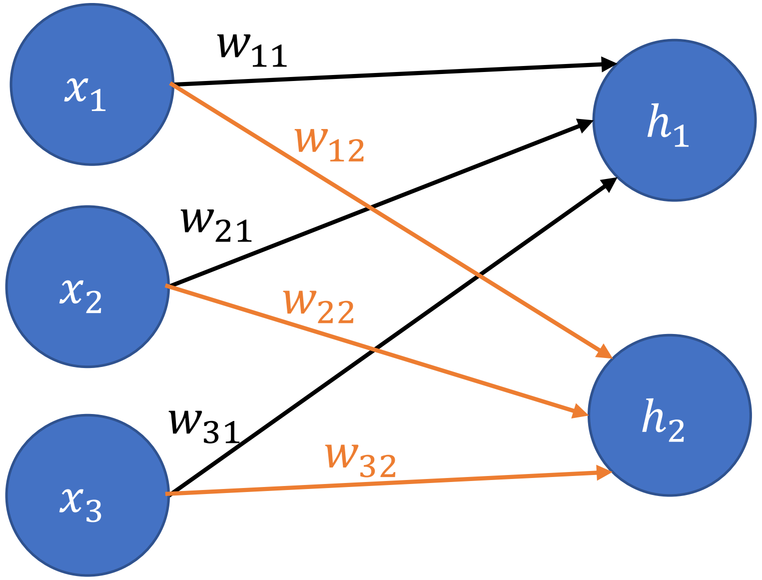
- What is the general purpose of the propagation function f_p?
  1. Weighted summation of incoming signals to one scalar
  2. Weighted fusion of incoming signals to one scalar
  3. Weighted product of incoming signals to a vector

A: 1, 2

B: all

C: 3

D: 2



Revision

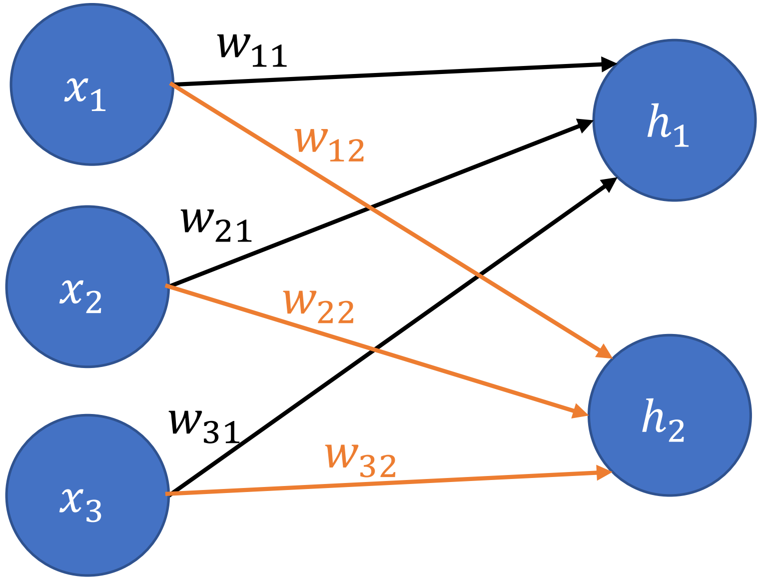
- What is the general purpose of the propagation function f_p?
  1. Weighted summation of incoming signals to one scalar
  2. Weighted fusion of incoming signals to one scalar
  3. Weighted product of incoming signals to a vector

A: 1, 2

B: all

C: 3

D: 2



$$W = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}$$

Revision

- What is the linear associator?
  1. Weighted summation of input signals to one scalar
  2. Weighted product of input signals to one scalar
  3. A possible propagation function f_p

A: 2, 3

B: all

C: 1, 3

D: 1

Revision

- What is the linear associator?
  1. Weighted summation of input signals to one scalar
  2. Weighted product of input signals to one scalar
  3. A possible propagation function f_p

A: 2, 3

B: all

C: 1, 3

D: 1

Revision

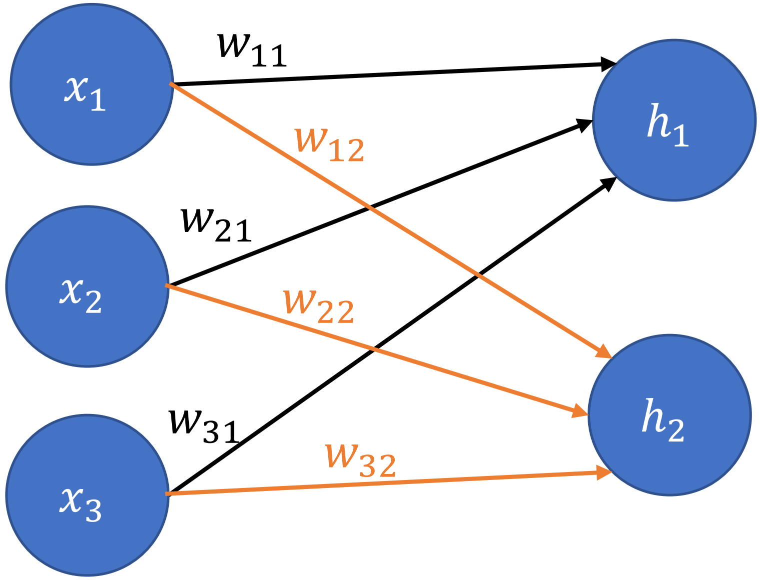
- What is the linear associator?
  1. Weighted summation of input signals to one scalar
  2. Weighted product of input signals to one scalar
  3. A possible propagation function f_p

A:	2, 3
B:	all
C:	1, 3
D:	1

Beispiele für Propagierungsfunktionen:

- $u_j := \sum_i w_{ij} x_i$ (linearer Assoziator)
- $u_j := \prod_i w_{ij} x_i$ (nicht-linearer Assoziator)
- $u_j := \max_i \{w_{ij} x_i\}$ (Maximum gewichtete Eingaben)
- $u_j := \sum_i s_i$, mit $s_i := \begin{cases} +1 & : \text{falls } w_{ij} x_i > 0 \\ -1 & : \text{sonst} \end{cases}$

Calculation of propagated value



$$h_1 = \sum_{i=1}^{3} w_{i1} x_i$$

$$h_2 = \sum_{i=1}^{3} w_{i2} x_i$$

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \qquad W = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}$$

$$\begin{bmatrix} h_1 \\ h_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{21} & w_{31} \\ w_{12} & w_{22} & w_{32} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = W^T \cdot x$$

Revision

- Which of the following functions can be used as an activation function f_a?
  1. Identity function
  2. Ramp function
  3. Step function
  4. Signum function
  5. Sigmoid function

A: all

B: 1, 2

C: 2, 3

D: 1, 3, 4, 5

Revision

- Which of the following functions can be used as an activation function f_a?
  1. Identity function
  2. Ramp function
  3. Step function
  4. Signum function
  5. Sigmoid function

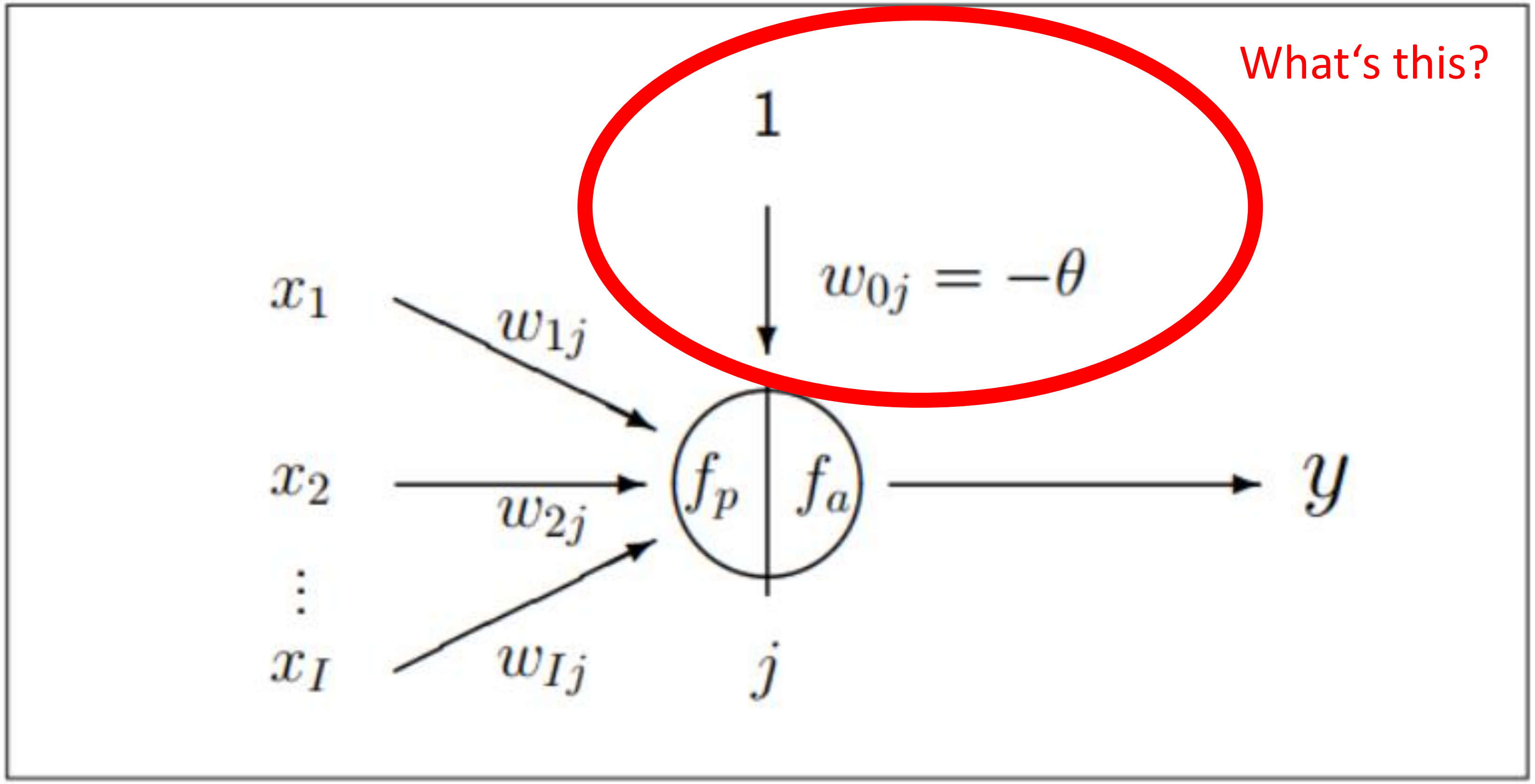
A: all

B: 1, 2

C: 2, 3

D: 1, 3, 4, 5

Scheme of Artificial Neuron



$f_p | f_a$ wird oft weggelassen, wenn aus dem Zusammenhang klar.

Simplification of Step function

- Propagation function (linear associator):

$$\mathrm{f}_p(x_1, x_2, \ldots, x_n) := \sum_{i=1}^{n} w_i x_i$$

- Activation function (step function):

$$\mathrm{f}_a(\hat{x}, \Theta) := \begin{cases} 1, & \text{if } \hat{x} > \Theta \\ 0, & \text{else} \end{cases}$$

- Output of a neuron:

$$\mathrm{f}_a(f_p(x_1, x_2, \ldots, x_n), \Theta) := \begin{cases} 1, & \text{if } f_p(x_1, x_2, \ldots, x_n) > \Theta \\ 0, & \text{else} \end{cases}$$

Simplification of Step function

$$\mathrm{f}_p(x_1, x_2, \ldots, x_n) > \Theta$$

$$\Updownarrow$$

$$\sum_{i=1}^{n} w_i x_i > \Theta$$

$$\Updownarrow$$

$$\sum_{i=1}^{n} w_i x_i - \Theta > 0$$

$$\Updownarrow \longleftarrow \mathrm{x}_0 := 1, w_0 := -\Theta$$

$$\sum_{i=0}^{n} w_i x_i > 0$$

Simplification of Step function

- (New) Propagation function (linear associator):

$$\mathrm{f}_p(x_1, x_2, \dots, x_n) := \sum_{i=1}^{n} w_i x_i \quad \Rightarrow \quad \mathrm{f}_p(x_0, x_1, \dots, x_n) := \sum_{i=0}^{n} w_i x_i$$

Simplification of Step function

- (New) Propagation function (linear associator):

$$\mathrm{f}_p(x_1, x_2, ..., x_n) := \sum_{i=1}^{n} w_i x_i \Rightarrow \mathrm{f}_p(x_0, x_1, ..., x_n) := \sum_{i=0}^{n} w_i x_i$$

- (Simplified) Activation function (step function):

$$\mathrm{f}_a(\hat{x}, \Theta) := \begin{cases} 1, & \text{if } \hat{x} > \Theta \\ 0, & \text{else} \end{cases} \Rightarrow \mathrm{f}_a(\hat{x}) := \begin{cases} 1, & \text{if } \hat{x} > 0 \\ 0, & \text{else} \end{cases}$$

Simplification of Step function

- (New) Propagation function (linear associator):

$$\mathrm{f}_p(x_1, x_2, \ldots, x_n) := \sum_{i=1}^{n} w_i x_i \quad \Rightarrow \quad \mathrm{f}_p(x_0, x_1, \ldots, x_n) := \sum_{i=0}^{n} w_i x_i$$

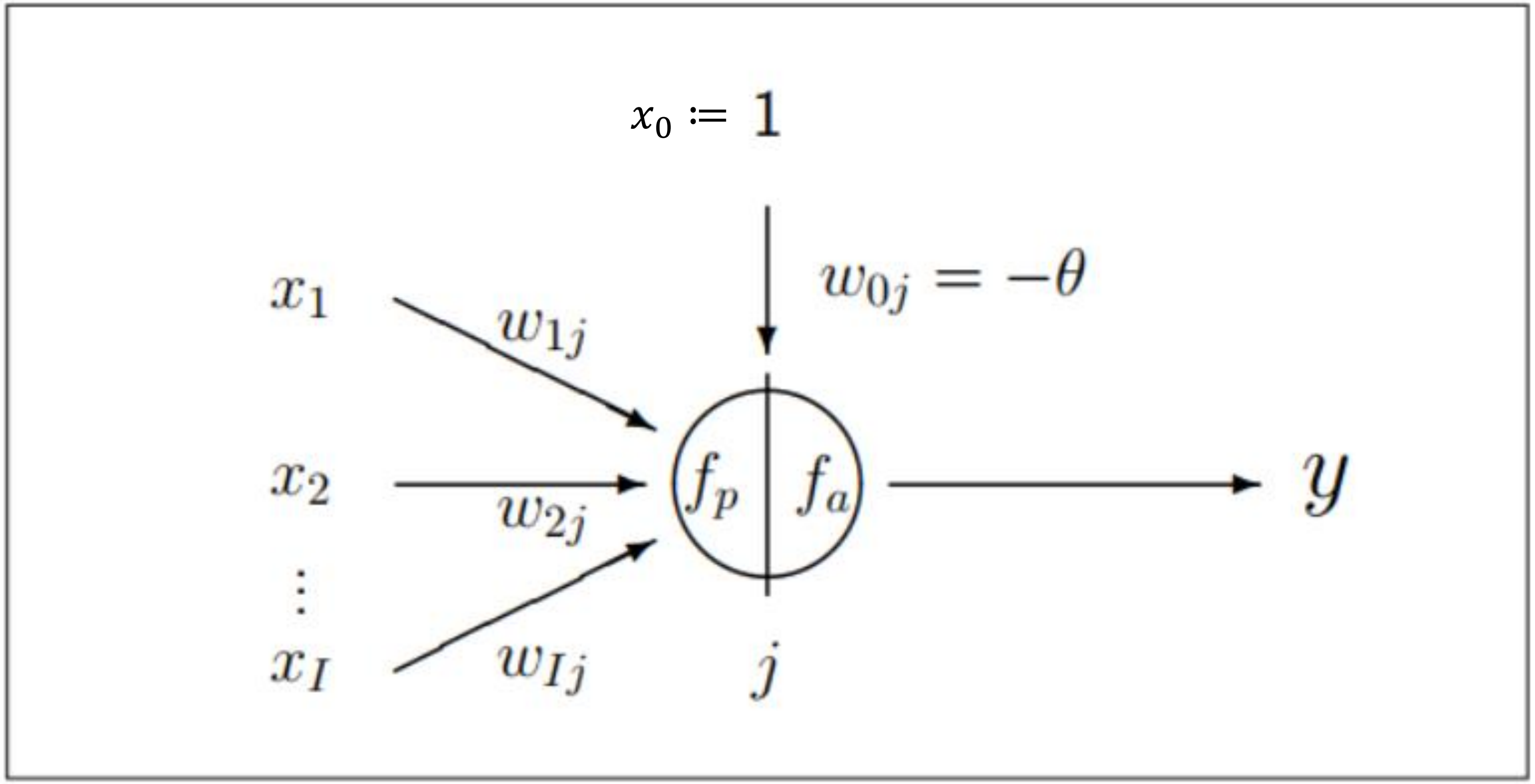
- (Simplified) Activation function (step function):

$$\mathrm{f}_a(\hat{x}, \Theta) := \begin{cases} 1, & \text{if } \hat{x} > \Theta \\ 0, & \text{else} \end{cases} \quad \Rightarrow \quad \mathrm{f}_a(\hat{x}) := \begin{cases} 1, & \text{if } \hat{x} > 0 \\ 0, & \text{else} \end{cases}$$

- Output of a neuron:

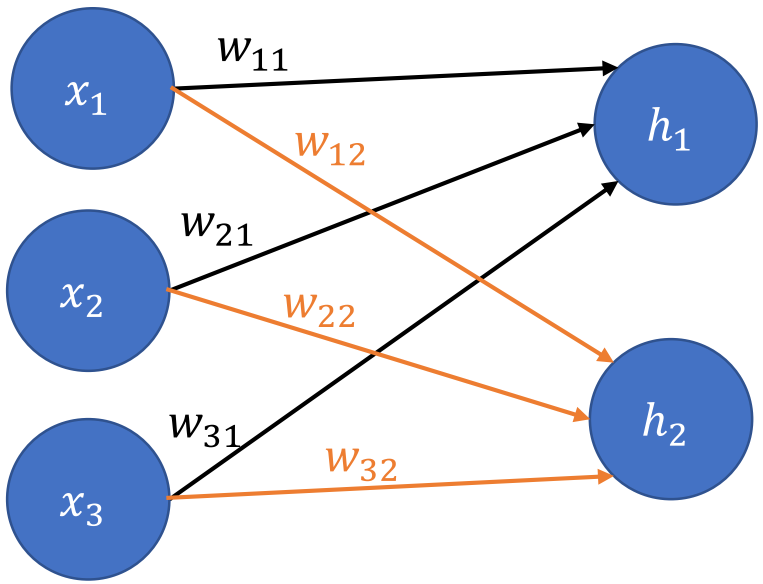
$$\mathrm{f}_a(f_p(x_1, x_2, \ldots, x_n), \Theta) := \begin{cases} 1, & \text{if } f_p(x_1, x_2, \ldots, x_n) > \Theta \\ 0, & \text{else} \end{cases} \quad \Rightarrow \quad \mathrm{f}_a(f_p(x_0, x_1, \ldots, x_n)) := \begin{cases} 1, & \text{if } f_p(x_0, x_1, x_2, \ldots, x_n) > 0 \\ 0, & \text{else} \end{cases}$$

Scheme of Artificial Neuron

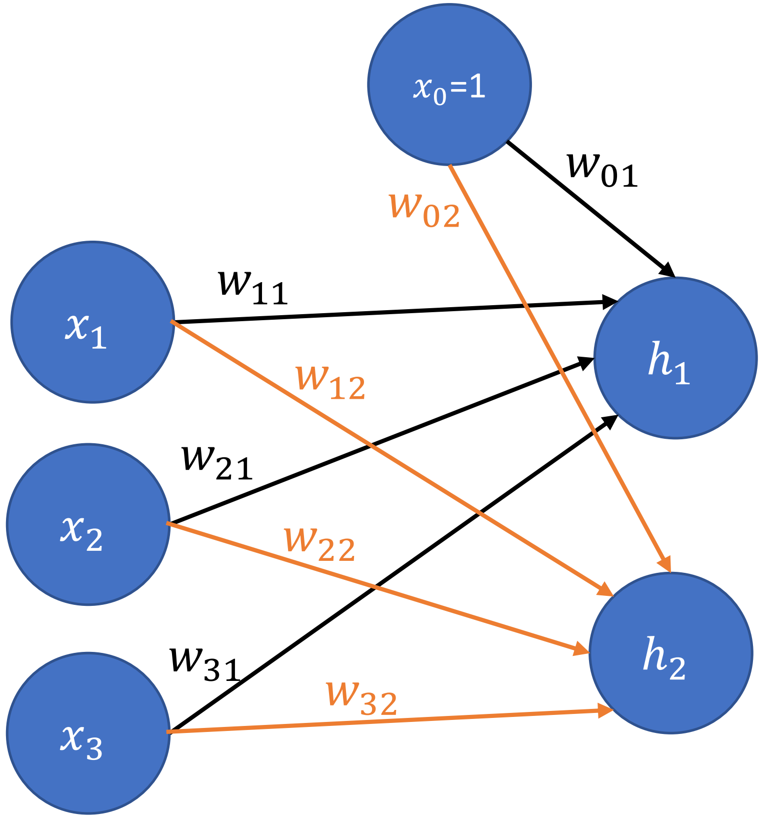


$f_p | f_a$ wird oft weggelassen, wenn aus dem Zusammenhang klar.

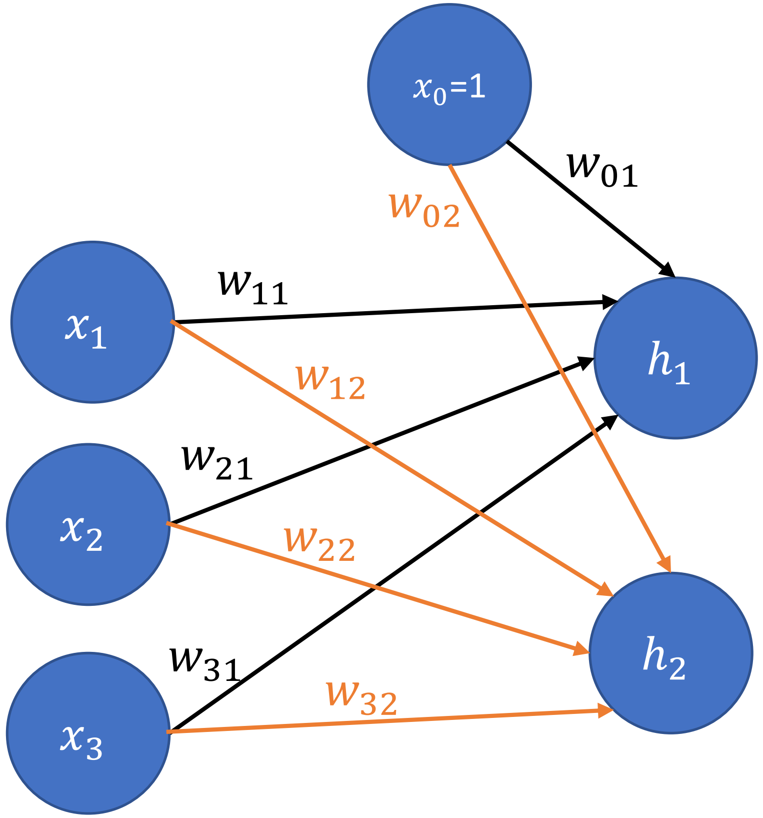
Calculation of propagated value



Calculation of propagated value



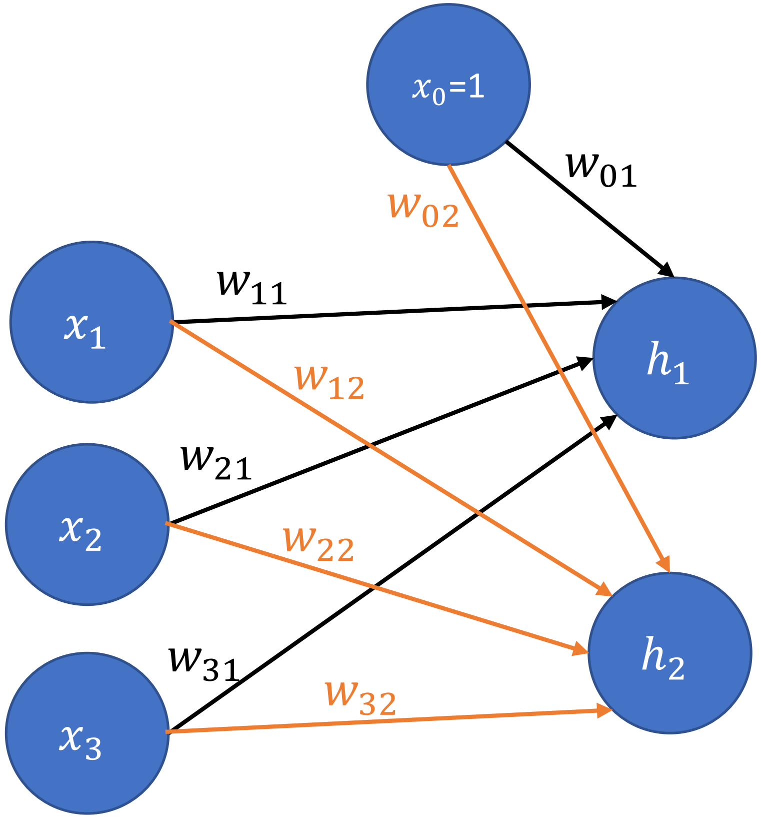
Calculation of propagated value



$$h_1 = \sum_{i=0}^{3} w_{i1} x_i$$

$$h_2 = \sum_{i=0}^{3} w_{i2} x_i$$

Calculation of propagated value



$$h_1 = \sum_{i=0}^{3} w_{i1} x_i$$

$$h_2 = \sum_{i=0}^{3} w_{i2} x_i$$

$$x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \qquad W = \begin{bmatrix} w_{01} & w_{02} \\ w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \end{bmatrix}$$

$$\begin{bmatrix} h_1 \\ h_2 \end{bmatrix} = W^T \cdot x$$

Revision

- Which of the following terms describe a parts of a biological neuron?
  1. Perikaryon
  2. Dentrites
  3. Axon
  4. Axon hillhock
  5. Glia
  6. Soma

A: all

B: 1-4

C: 1-3, 5, 6

D: 1-4, 6

Revision

- Which of the following terms describe a parts of a biological neuron?
  1. Perikaryon
  2. Dentrites
  3. Axon
  4. Axon hillhock
  5. Glia
  6. Soma

A: all

B: 1-4

C: 1-3, 5, 6

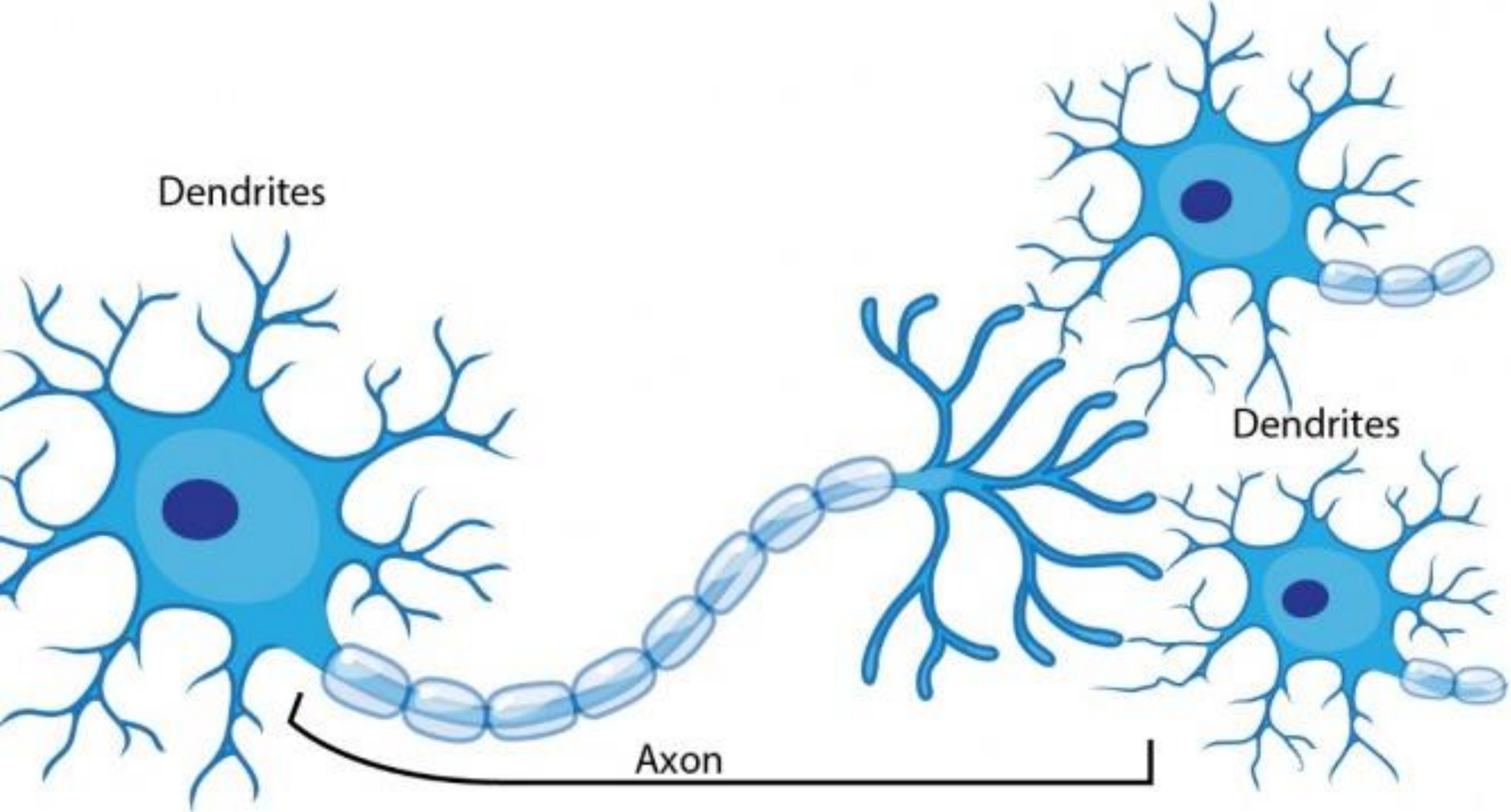
D: 1-4, 6

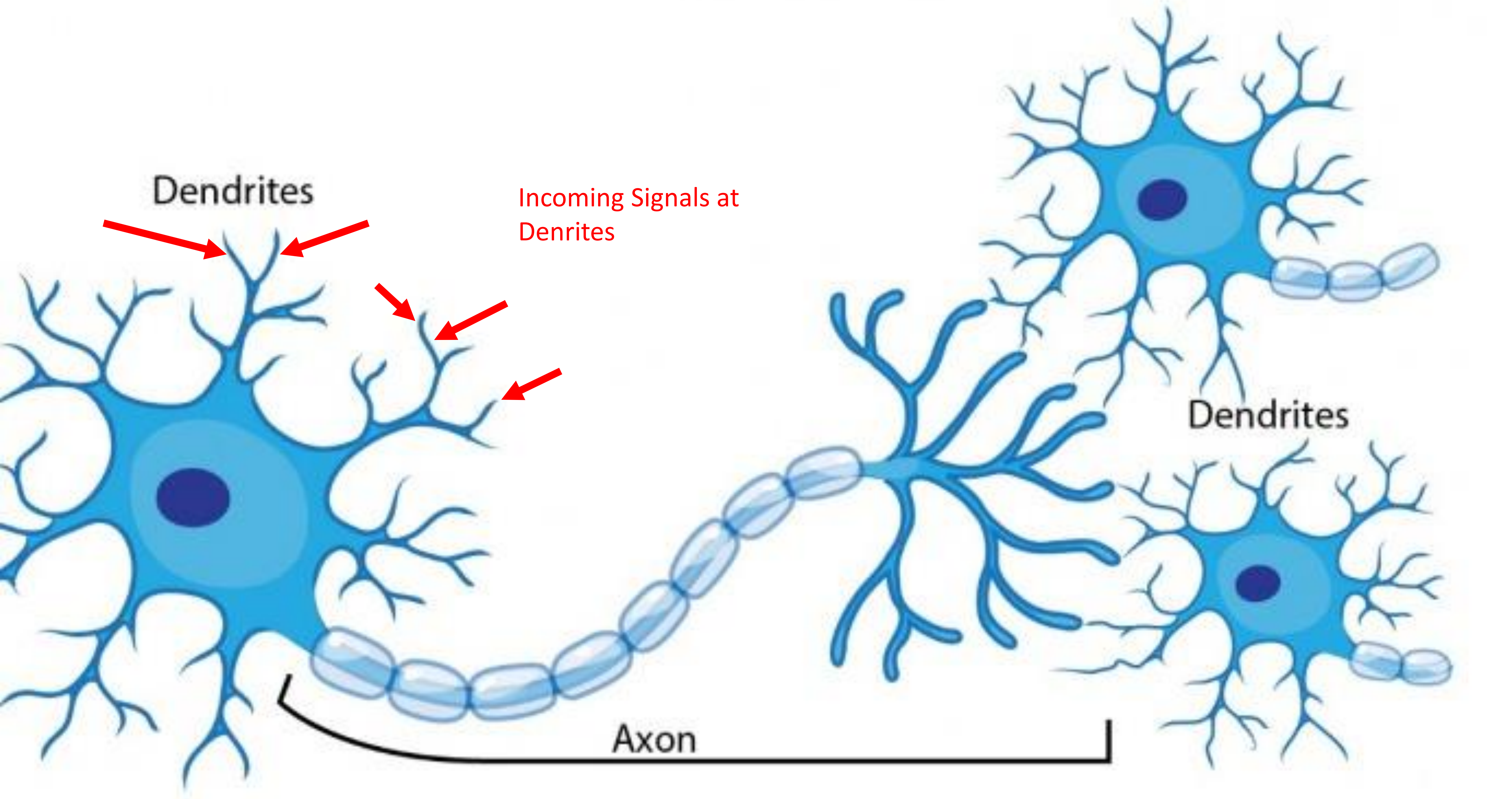
Content

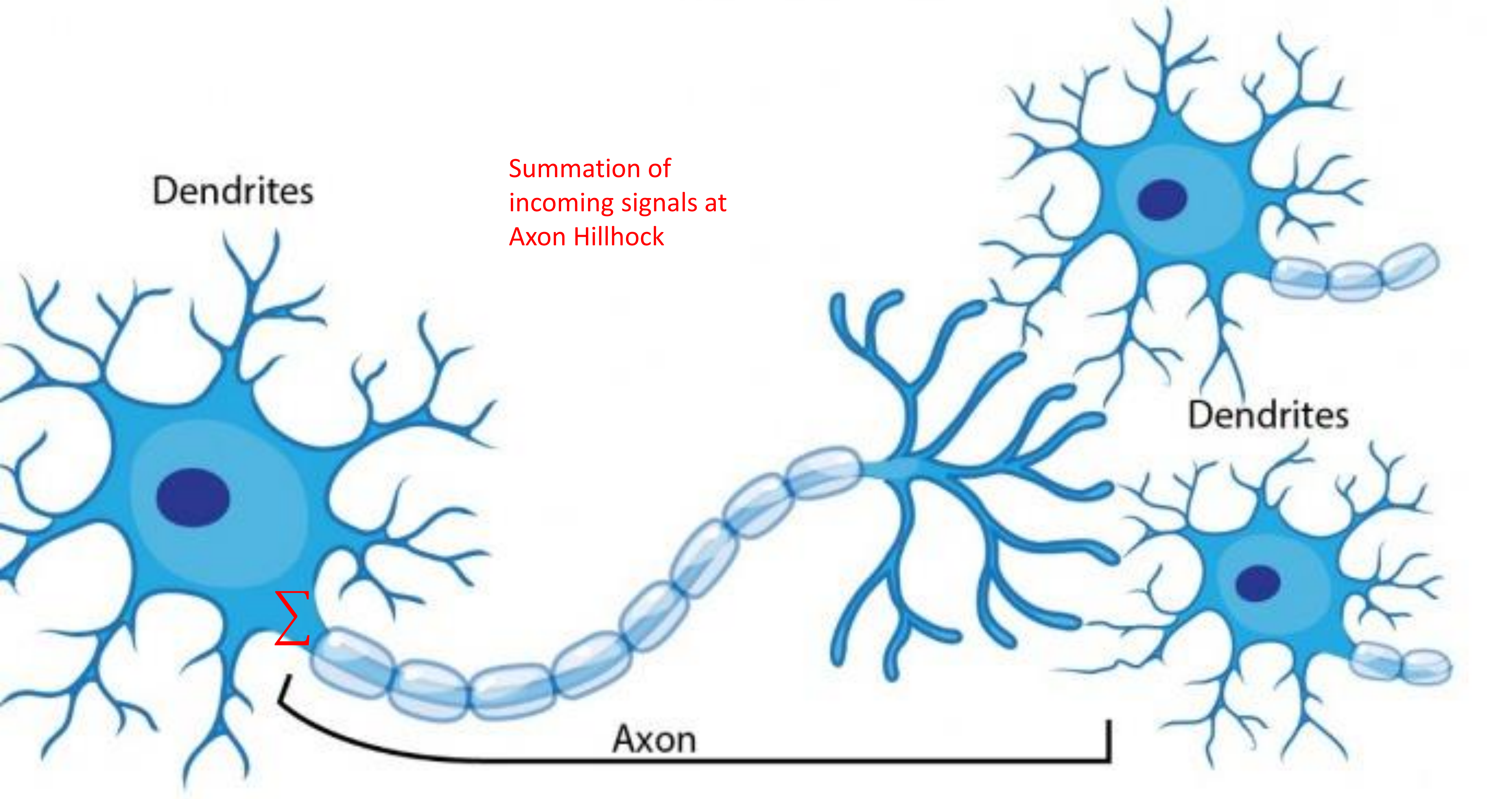
- Revision
- Biological Motivation for ANNs
- McCulloch-Pitts Neuron
- Tasks

Biological Motivation for ANNs

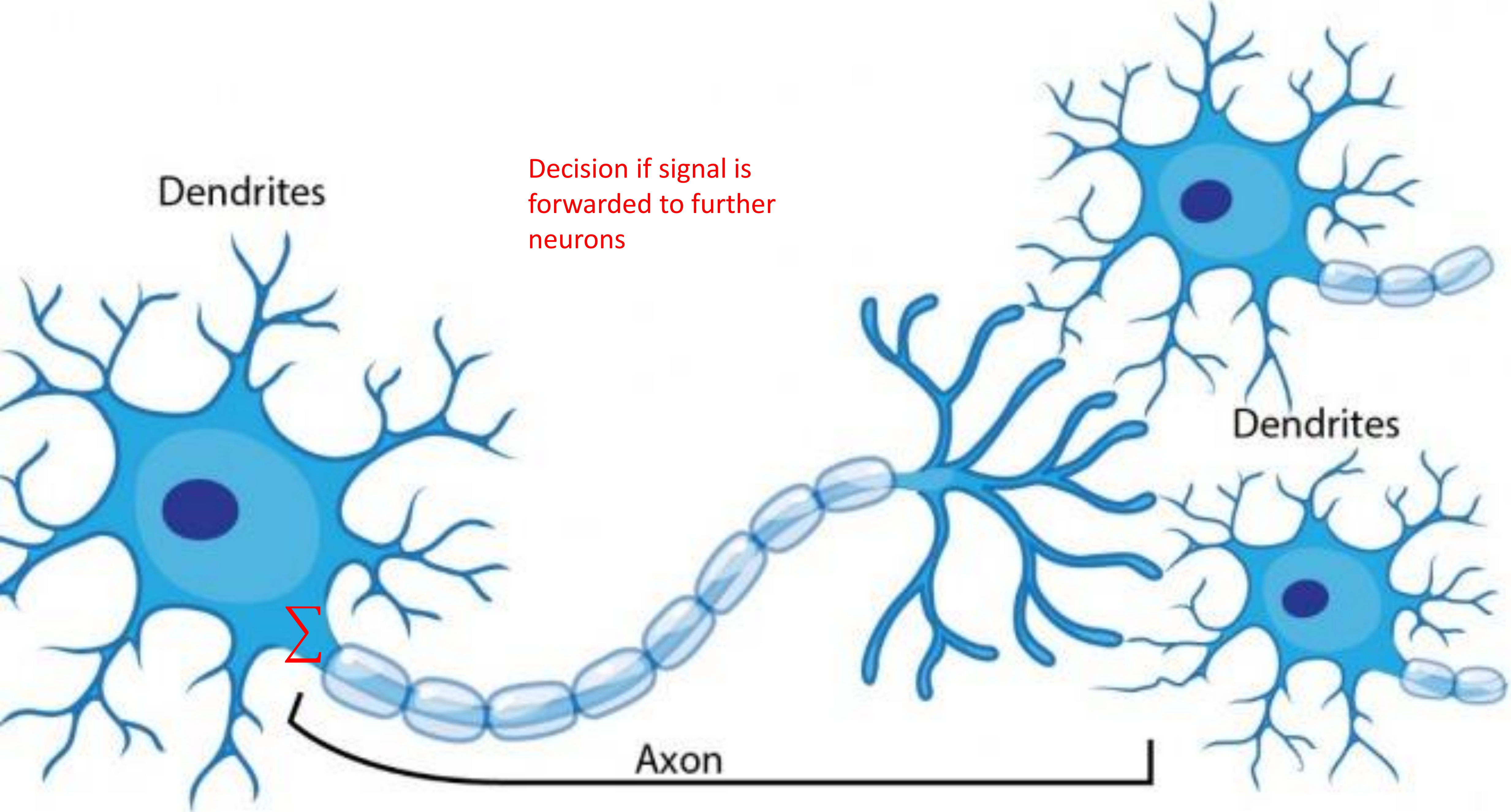
- Multiple signals are received at dendrites
- Depending on synapse there are
  - Excitatory and
  - Inhibiting signals
- Incoming signals are processed in axon hillhock
  - Summation of postsynaptic potentials
- Resulting signal is transferred along axon to terminals

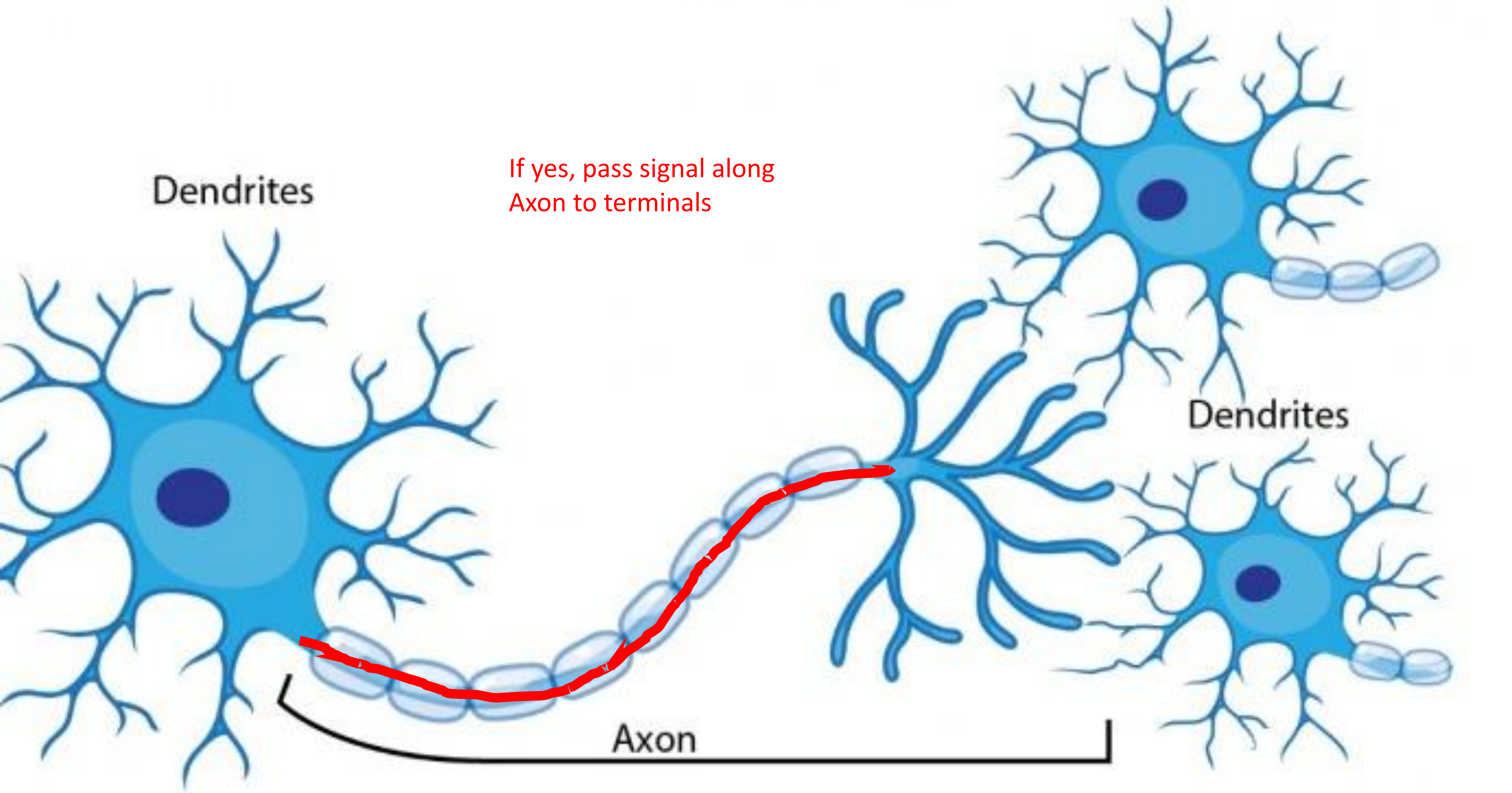

Dendrites
Dendrites
Axon

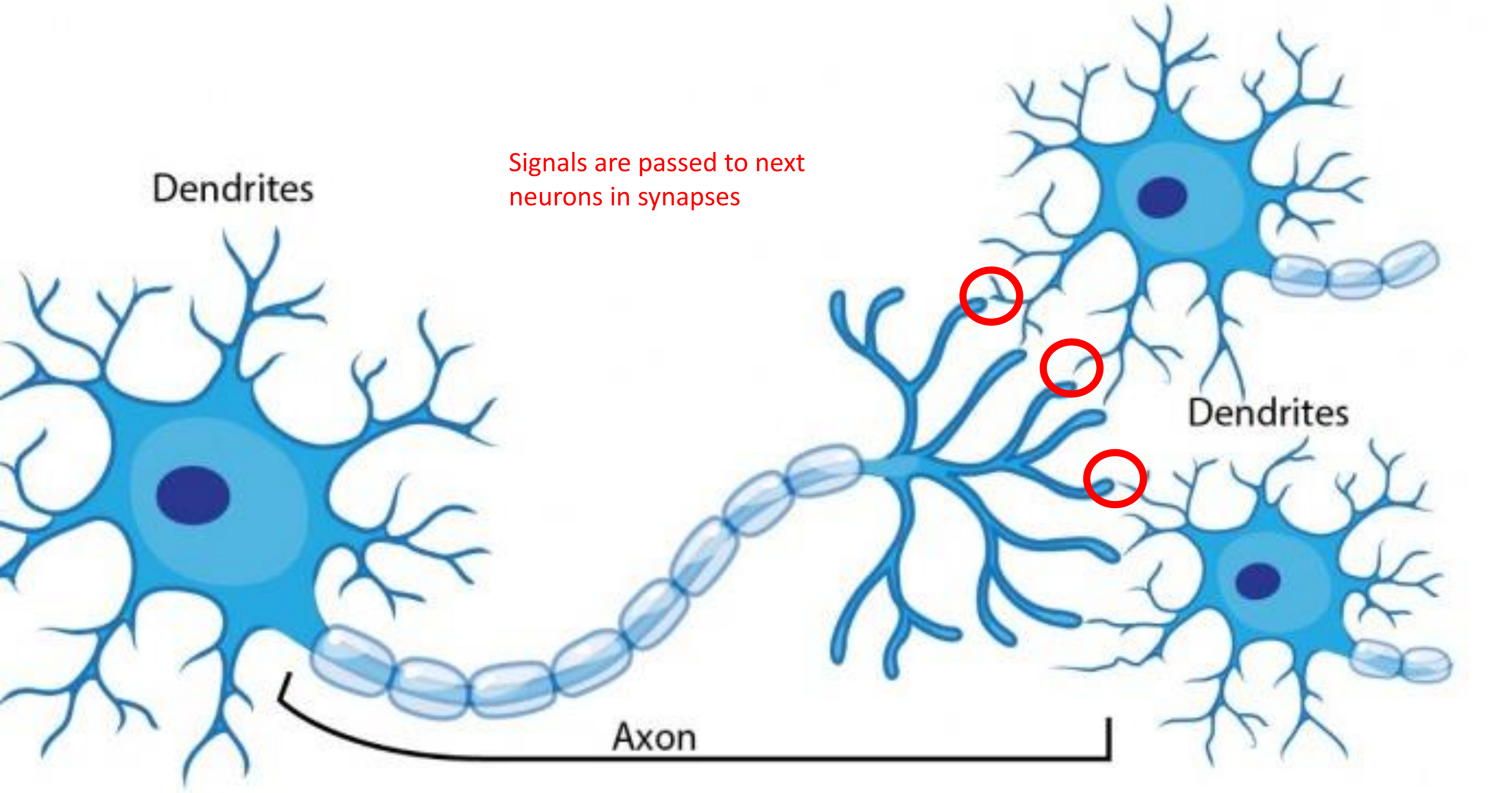

Dendrites
Incoming Signals at
Denrites
Dendrites
Axon


Summation of
incoming signals at
Axon Hillhock
Dendrites
Σ
Axon
Dendrites

Decision if signal is forwarded to further neurons




Dendrites
If yes, pass signal along
Axon to terminals
Dendrites
Axon


Dendrites
Signals are passed to next
neurons in synapses
Dendrites
Axon

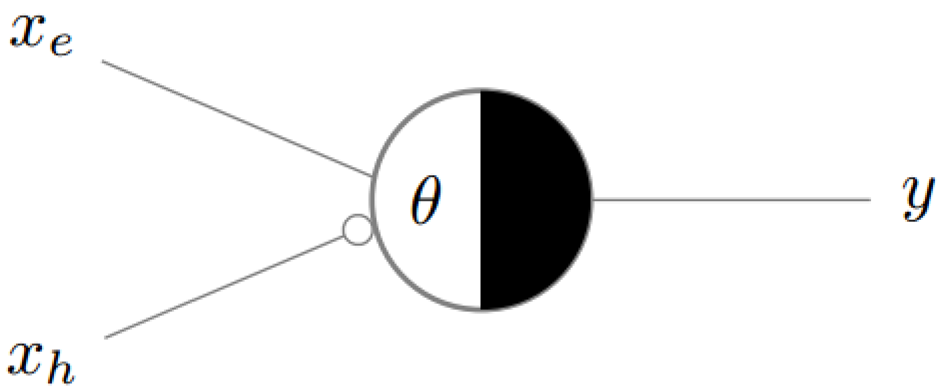
Content

- Revision
- Biological Motivation for ANNs
- McCulloch-Pitts Neuron
- Tasks

The McCulloch Pitts Neuron

- Very simple model of neuron by McCulloch and Pitts
- Input:
  - N-dimensional **binary** vector $(x_1, x_2, \ldots x_n) \in \{0,1\}^n$
  - Each component represents incoming signal
  - Each signal can be either:
    - Excitatory or
    - Inhibiting
- Output (binary):
  1. Sum over all excitatory signals
  2. If sum is greater than (or equal to) a threshold $\Theta \in \mathbb{R}$
     AND if **all** inhibiting signals are zero,
     then return 1, elso return 0

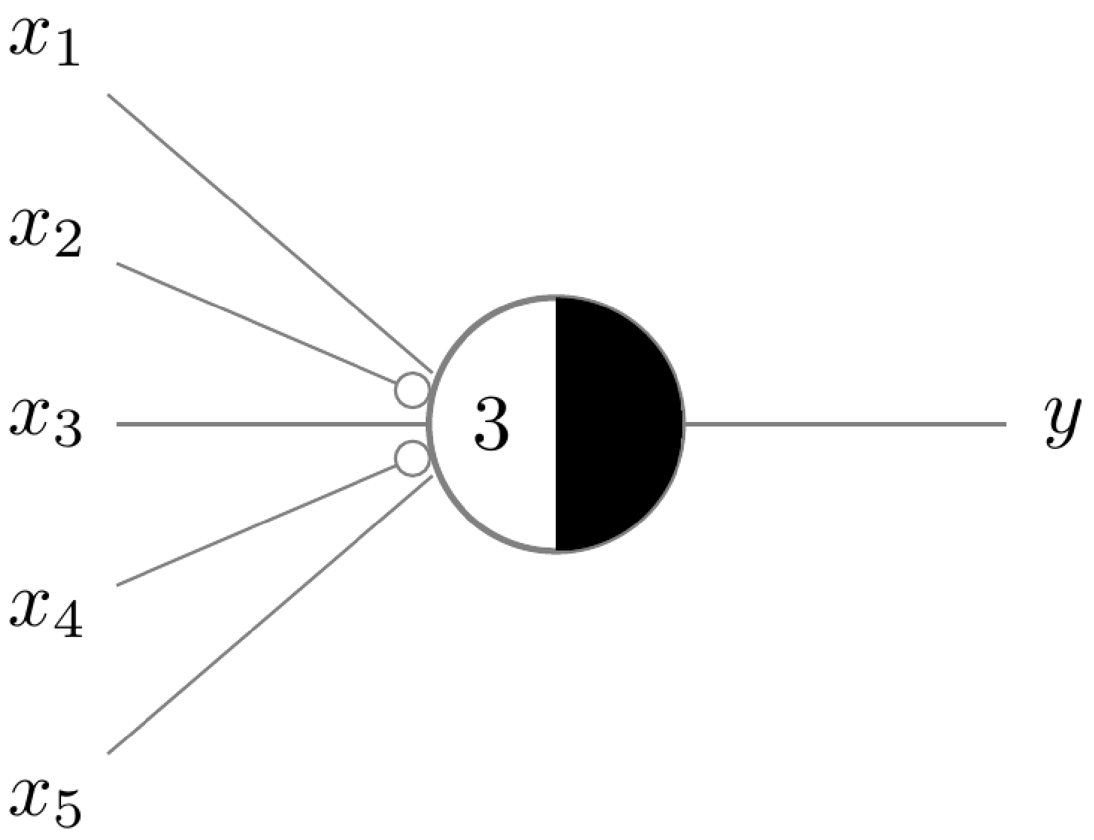
The McCulloch Pitts Neuron



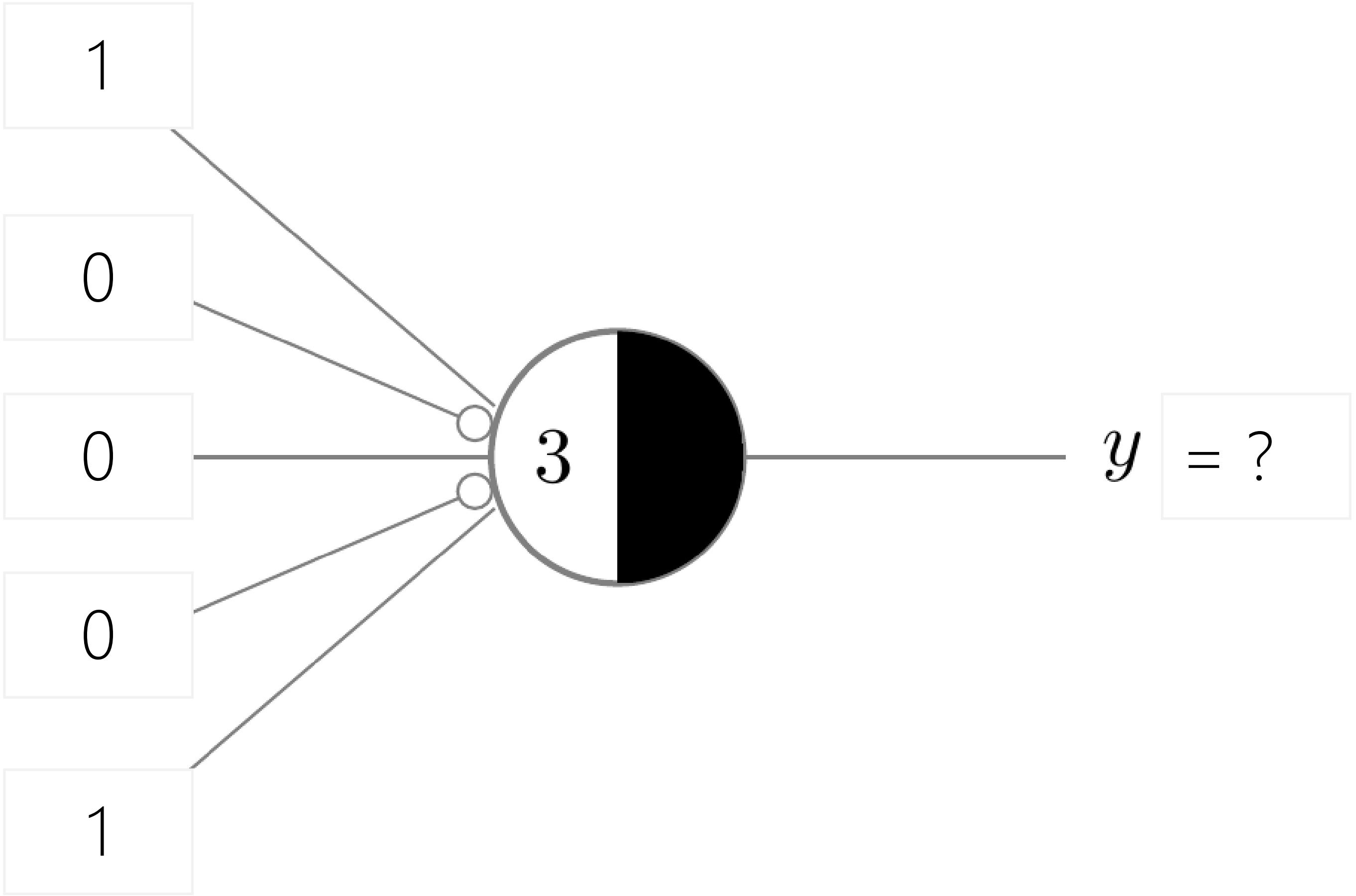
Content

- Revision
- Biological Motivation for ANNs
- McCulloch-Pitts Neuron
- **Tasks**

The McCulloch Pitts Neuron



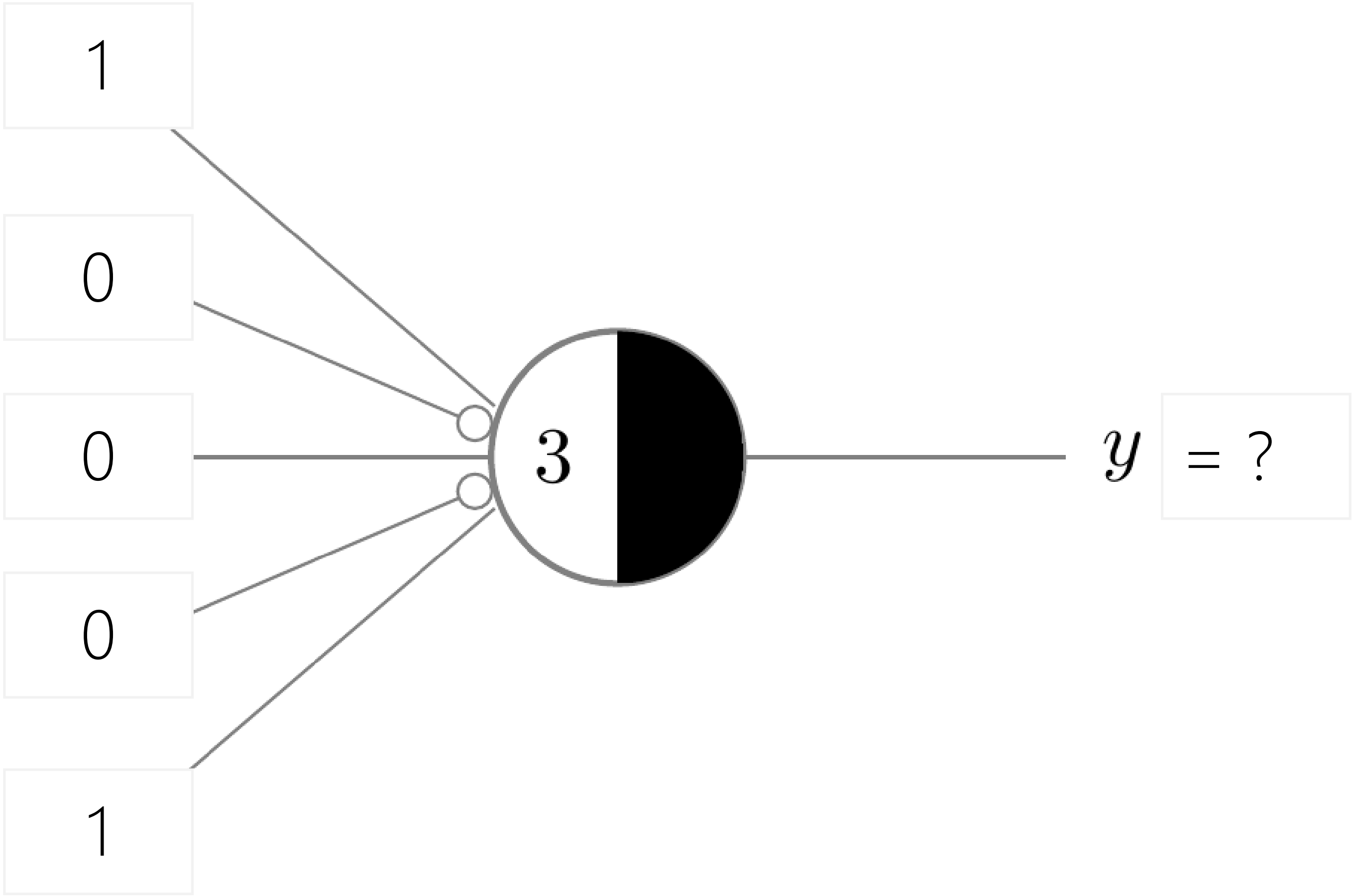
The McCulloch Pitts Neuron



A: y = 0

B: y = 1

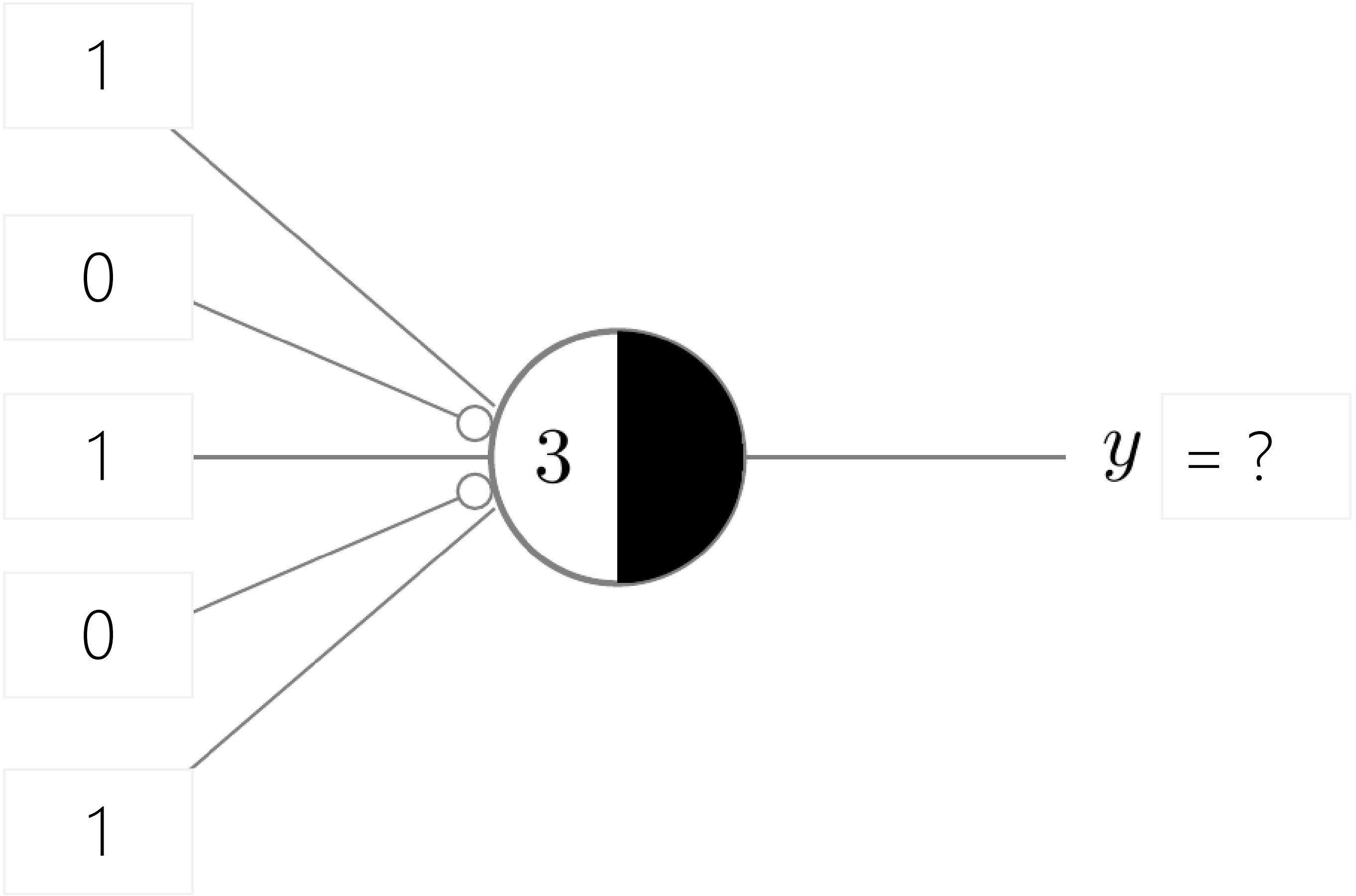
The McCulloch Pitts Neuron



A: y = 0

B: y = 1

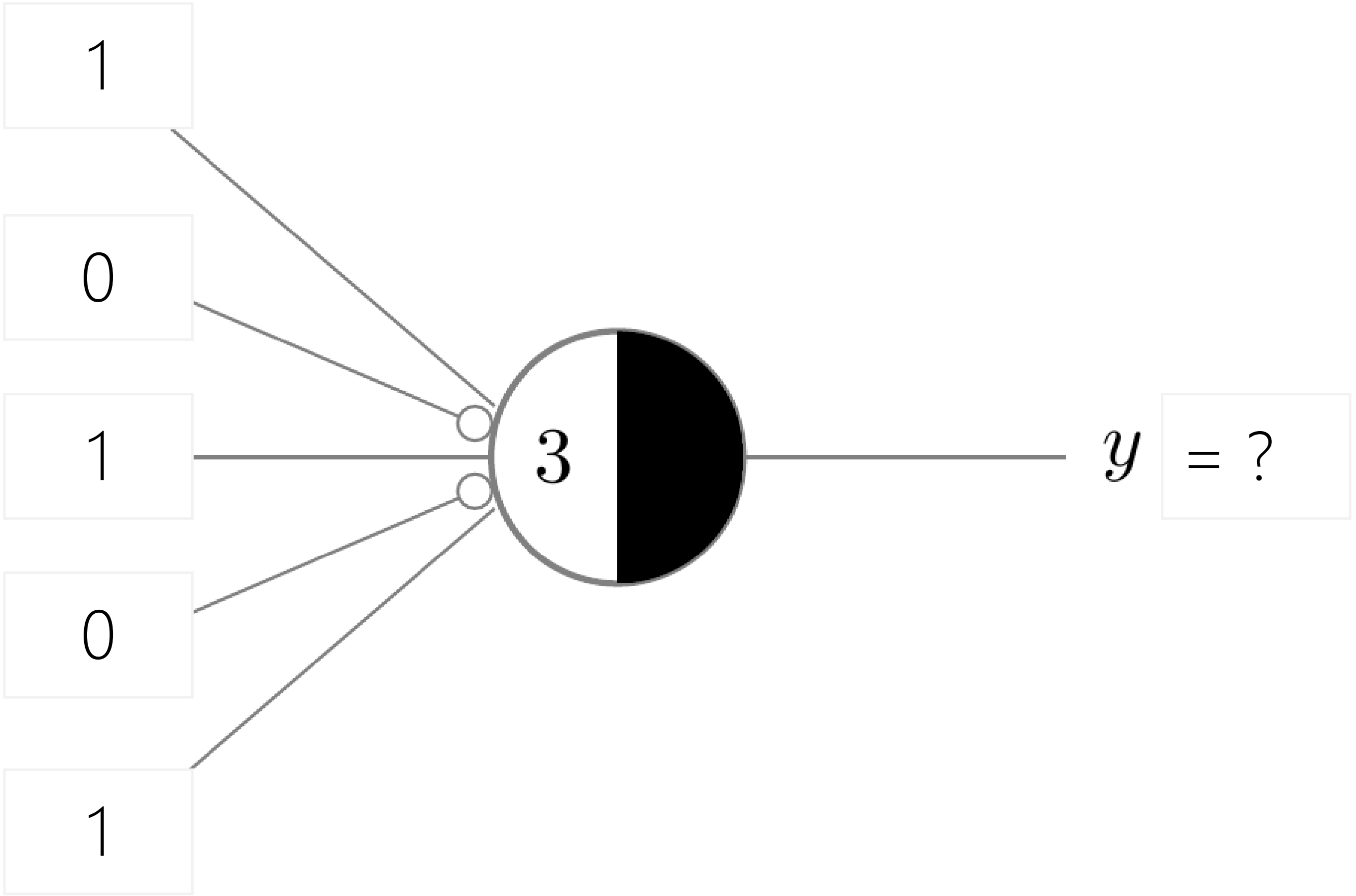
The McCulloch Pitts Neuron



A: y = 0

B: y = 1

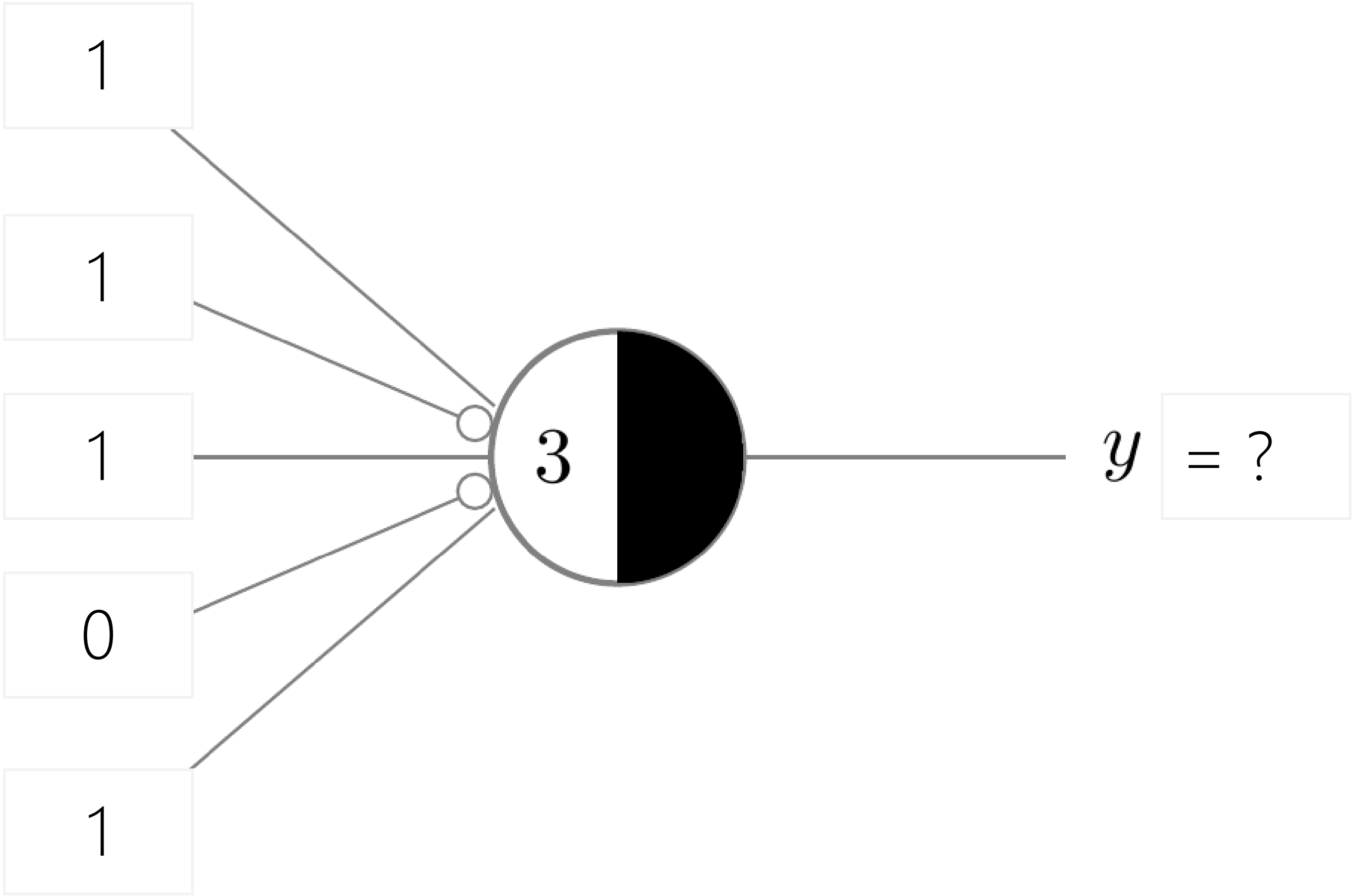
The McCulloch Pitts Neuron



A: y = 0

B: y = 1

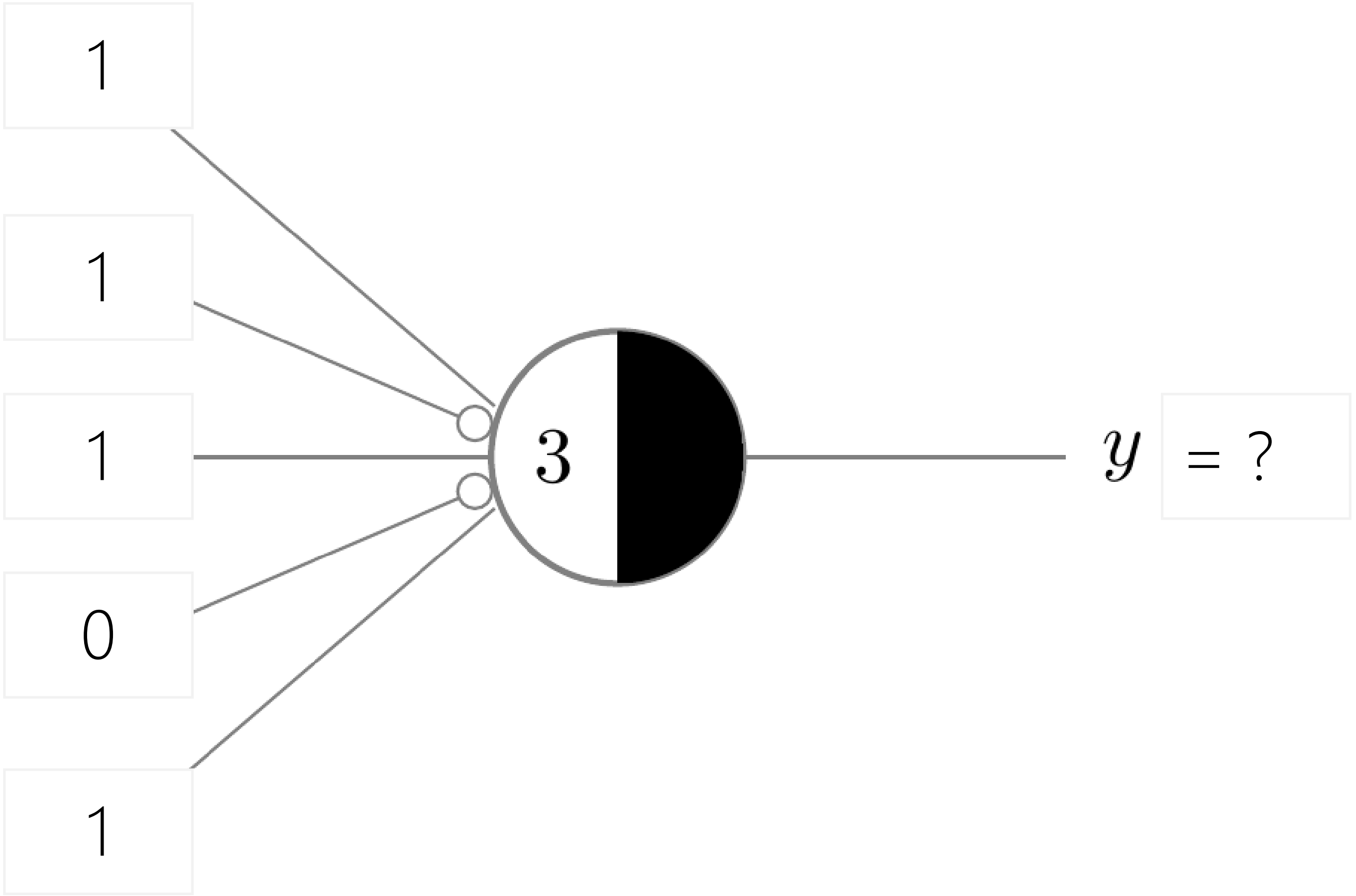
The McCulloch Pitts Neuron



A: y = 0

B: y = 1

The McCulloch Pitts Neuron



A: y = 0

B: y = 1

Excursion: Boolean Functions

- Given a binary input vector, assign a binary output, i.e.:

$$f: \{0,1\}^N \rightarrow \{0,1\}$$

- Example: AND function:

$$f_{AND} : \{0,1\}^2 \rightarrow \{0,1\}$$

with:

$$f_{AND}(0,0) \coloneqq 0$$
$$f_{AND}(0,1) \coloneqq 0$$
$$f_{AND}(1,0) \coloneqq 0$$
$$f_{AND}(1,1) \coloneqq 1$$

Excursion: Boolean Functions

- Representation of binary function as truth table:

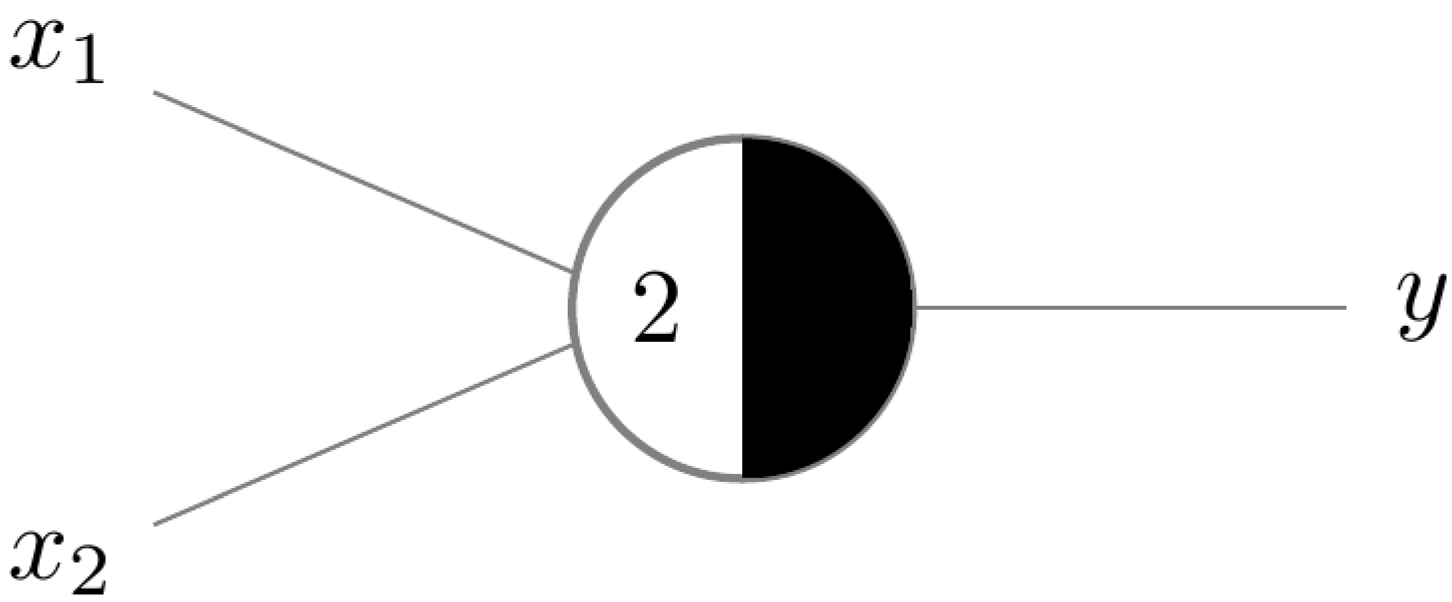
x_1	x_2	f_{AND}
0	0	0
0	1	0
1	0	0
1	1	1

The McCulloch Pitts Neuron

- Task: Construct a McCulloch Pitts Neuron, that models AND

The McCulloch Pitts Neuron

- Task: Construct a McCulloch Pitts Neuron, that models AND



Excursion: Boolean Functions

- Given a binary input vector, assign a binary output, i.e.:

$$f\colon \{0,1\}^N \rightarrow \{0,1\}$$

- Example: OR function:

$$f_{OR}\colon \{0,1\}^2 \rightarrow \{0,1\}$$

with:

$$\begin{aligned} f_{OR}(0,0) &\coloneqq 0 \\ f_{OR}(0,1) &\coloneqq 1 \\ f_{OR}(1,0) &\coloneqq 1 \\ f_{OR}(1,1) &\coloneqq 1 \end{aligned}$$

Excursion: Boolean Functions

- Representation of binary function as truth table:

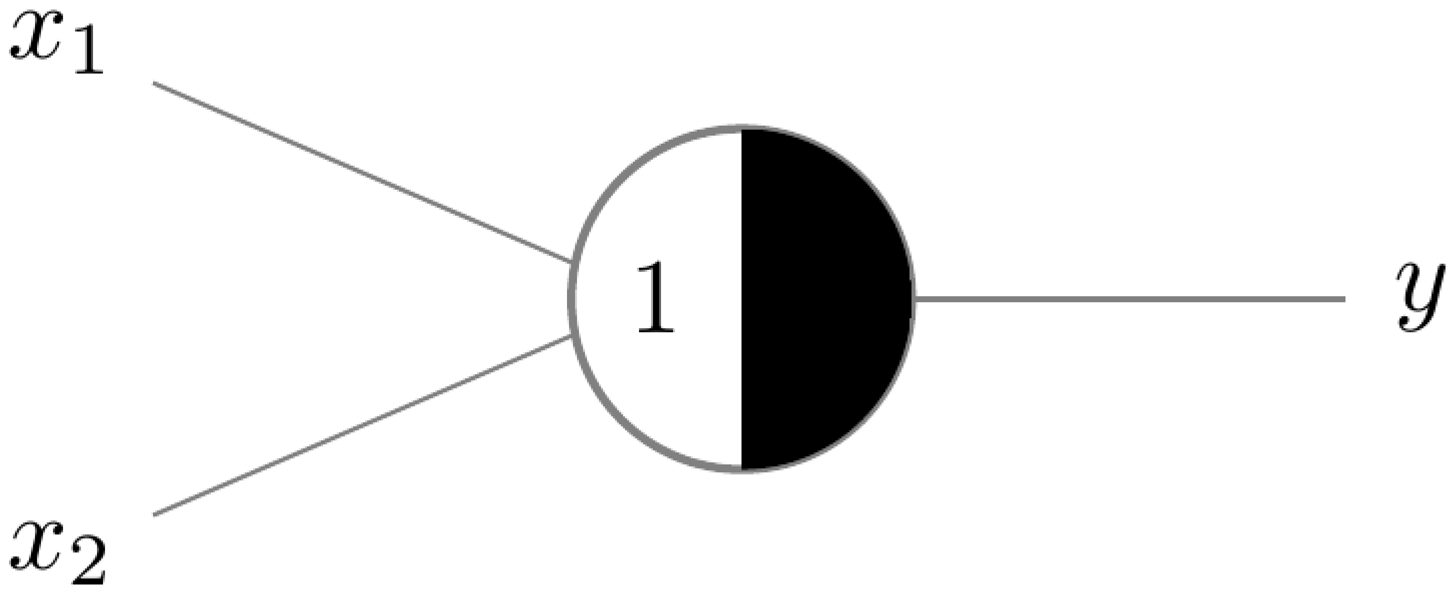
x_1	x_2	f_{OR}
0	0	0
0	1	1
1	0	1
1	1	1

The McCulloch Pitts Neuron

- Task: Construct a McCulloch Pitts Neuron, that models OR

The McCulloch Pitts Neuron

- Task: Construct a McCulloch Pitts Neuron, that models OR



Excursion: Boolean Functions

- Notation:

$$f(x_1, x_2, x_3) := x_1 \neg x_2 x_3$$

means:

$$„\ x_1 \text{and } (\text{not } x_2) \text{ and } x_3\ “$$

Excursion: Boolean Functions

x_1	x_2	x_3	$x_1 \neg x_2 x_3$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

The McCulloch Pitts Neuron

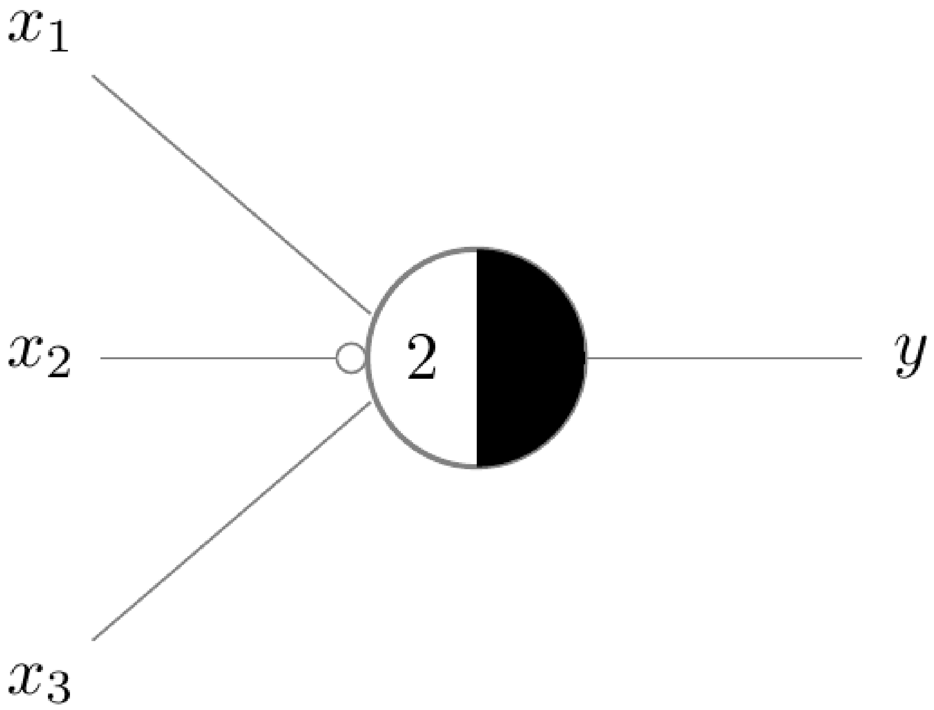
- Task: Construct a McCulloch Pitts Neuron, that models

$$f(x_1, x_2, x_3) := x_1 \neg x_2 x_3$$

The McCulloch Pitts Neuron

- Task: Construct a McCulloch Pitts Neuron, that models

$$f(x_1, x_2, x_3) \coloneqq x_1 \neg x_2 x_3$$



Excursion: Boolean Functions

- Notation:

$$f(x_1, x_2, x_3) \coloneqq \; x_1 \neg x_2 x_3 \vee x_2$$

means:

$$„[\, x_1 \text{and } (\text{not } x_2) \text{ and } x_3] \; or \; x_2 \quad "$$

Excursion: Boolean Functions

x_1	x_2	x_3	$x_1 \neg x_2 x_3 \vee x_2$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Excursion: Boolean Functions

x_1	x_2	x_3	$x_1 \neg x_2 x_3 \lor x_2$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

The McCulloch Pitts Neuron

- Task: Construct a McCulloch Pitts Net, that models

$$f(x_1, x_2, x_3) \coloneqq x_1 \neg x_2 x_3 \vee x_2$$

The McCulloch Pitts Neuron

- Task: Construct a McCulloch Pitts Net, that models

$$f(x_1, x_2, x_3) \coloneqq x_1 \neg x_2 x_3 \vee x_2$$