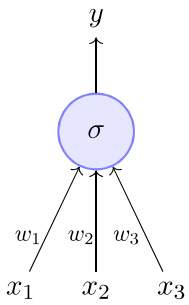
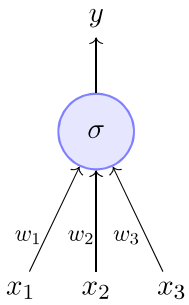
Module 2.1: Biological Neurons



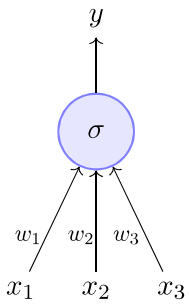
Artificial Neuron

- The most fundamental unit of a deep neural network is called an *artificial neuron*



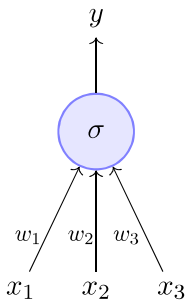
Artificial Neuron

- The most fundamental unit of a deep neural network is called an *artificial neuron*
- Why is it called a neuron ? Where does the inspiration come from ?



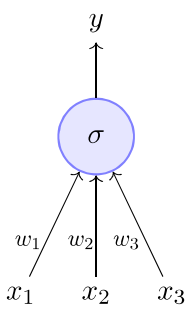
Artificial Neuron

- The most fundamental unit of a deep neural network is called an *artificial neuron*
- Why is it called a neuron ? Where does the inspiration come from ?
- The inspiration comes from biology (more specifically, from the *brain*)



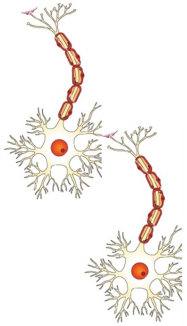
Artificial Neuron

- The most fundamental unit of a deep neural network is called an *artificial neuron*
- Why is it called a neuron ? Where does the inspiration come from ?
- The inspiration comes from biology (more specifically, from the *brain*)
- *biological neurons = neural cells = neural processing units*



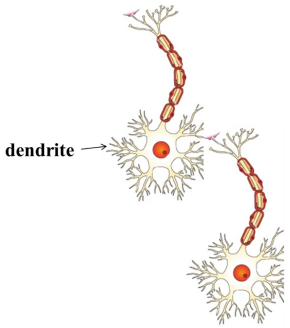
Artificial Neuron

- The most fundamental unit of a deep neural network is called an *artificial neuron*
- Why is it called a neuron ? Where does the inspiration come from ?
- The inspiration comes from biology (more specifically, from the *brain*)
- *biological neurons = neural cells = neural processing units*
- We will first see what a biological neuron looks like ...



Biological Neurons*

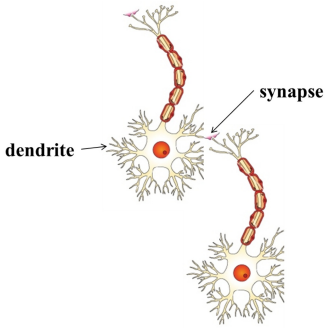
*Image adapted from
https://cdn.vectorstock.com/i/composite/12,25/neuron-cell-vector-81225.jpg



Biological Neurons*

- **dendrite:** receives signals from other neurons

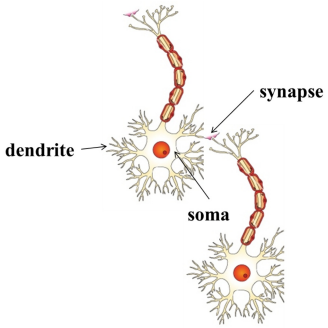
*Image adapted from
https://cdn.vectorstock.com/i/composite/12,25/neuron-cell-vector-81225.jpg



Biological Neurons*

- **dendrite:** receives signals from other neurons
- **synapse:** point of connection to other neurons

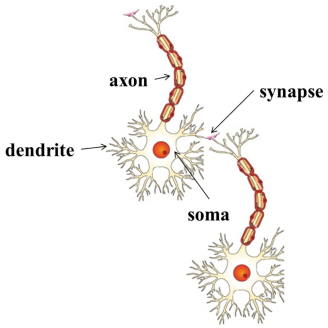
*Image adapted from
https://cdn.vectorstock.com/i/composite/12,25/neuron-cell-vector-81225.jpg



Biological Neurons*

- **dendrite:** receives signals from other neurons
- **synapse:** point of connection to other neurons
- **soma:** processes the information

*Image adapted from
https://cdn.vectorstock.com/i/composite/12,25/neuron-cell-vector-81225.jpg



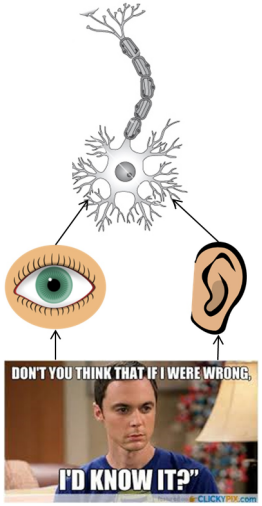
Biological Neurons*

- **dendrite:** receives signals from other neurons
- **synapse:** point of connection to other neurons
- **soma:** processes the information
- **axon:** transmits the output of this neuron

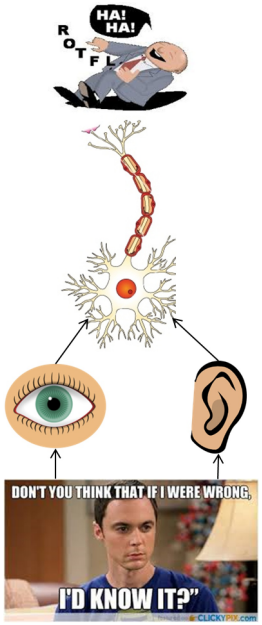
*Image adapted from
https://cdn.vectorstock.com/i/composite/12,25/neuron-cell-vector-81225.jpg

- Let us see a very cartoonish illustration of how a neuron works

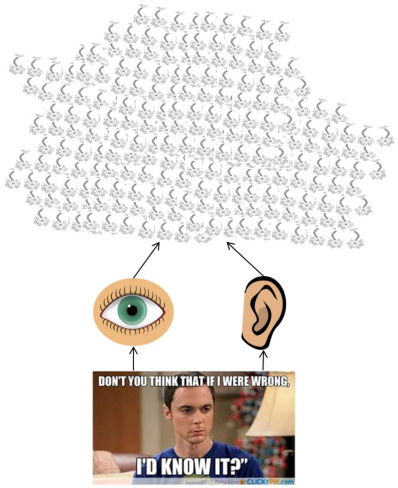
- Let us see a very cartoonish illustration of how a neuron works
- Our sense organs interact with the outside world



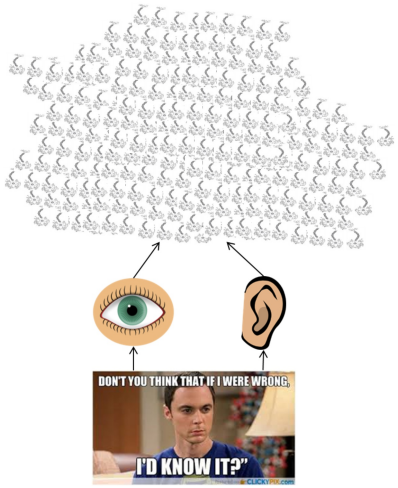
- Let us see a very cartoonish illustration of how a neuron works
- Our sense organs interact with the outside world
- They relay information to the neurons



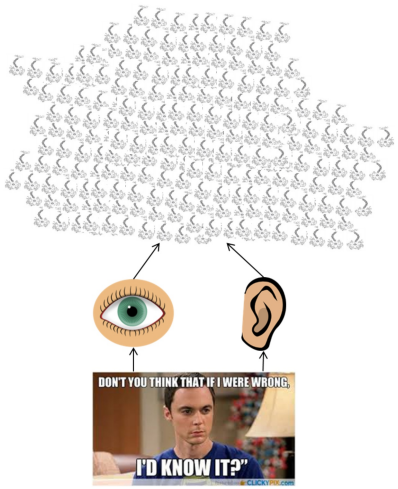
- Let us see a very cartoonish illustration of how a neuron works
- Our sense organs interact with the outside world
- They relay information to the neurons
- The neurons (may) get activated and produces a response (laughter in this case)



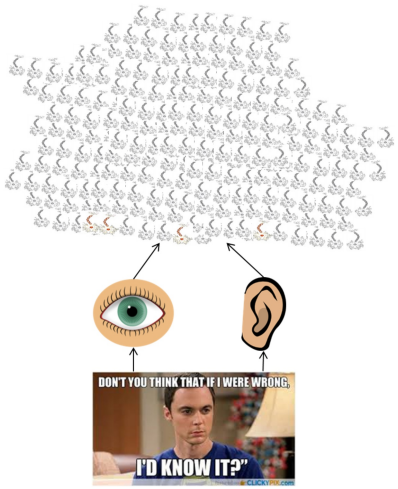
- Of course, in reality, it is not just a single neuron which does all this



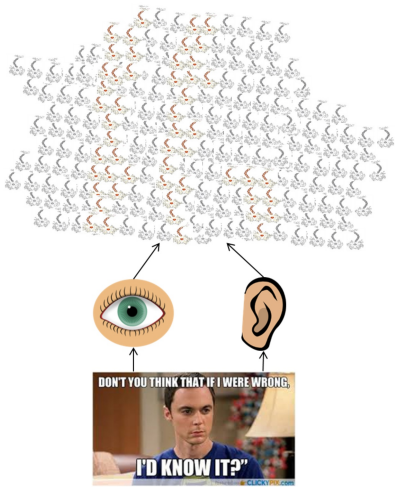
- Of course, in reality, it is not just a single neuron which does all this
- There is a massively parallel interconnected network of neurons



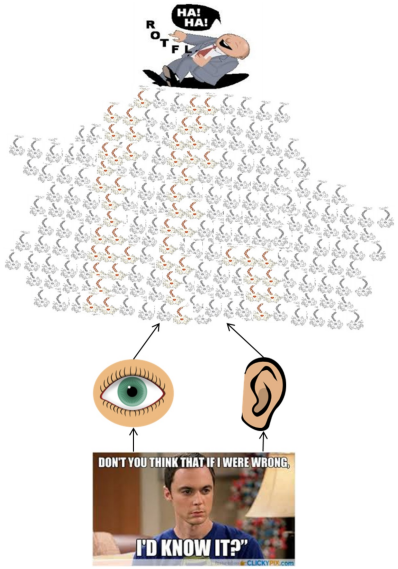
- Of course, in reality, it is not just a single neuron which does all this
- There is a massively parallel interconnected network of neurons
- The sense organs relay information to the lowest layer of neurons



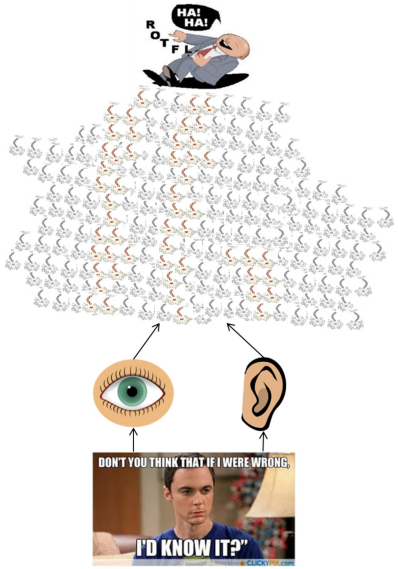
- Of course, in reality, it is not just a single neuron which does all this
- There is a massively parallel interconnected network of neurons
- The sense organs relay information to the lowest layer of neurons
- Some of these neurons may fire (in red) in response to this information and in turn relay information to other neurons they are connected to



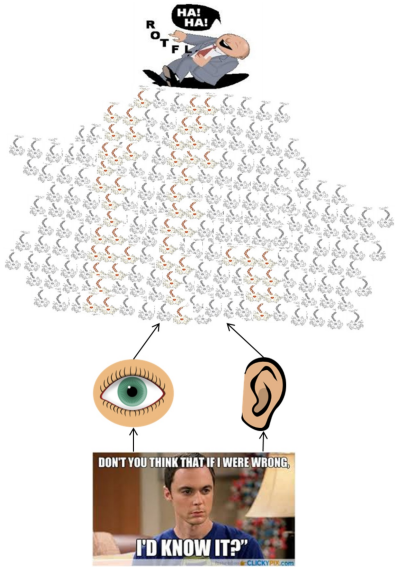
- Of course, in reality, it is not just a single neuron which does all this
- There is a massively parallel interconnected network of neurons
- The sense organs relay information to the lowest layer of neurons
- Some of these neurons may fire (in red) in response to this information and in turn relay information to other neurons they are connected to
- These neurons may also fire (again, in red) and the process continues



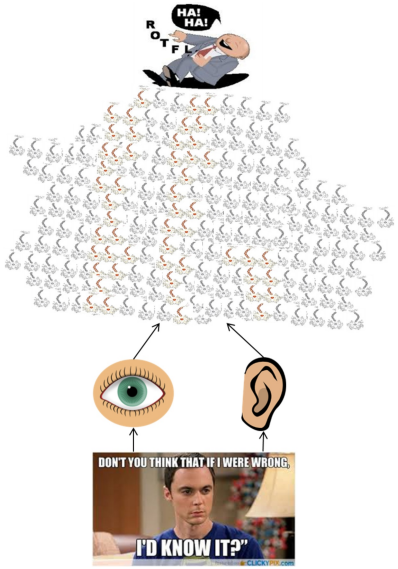
- Of course, in reality, it is not just a single neuron which does all this
- There is a massively parallel interconnected network of neurons
- The sense organs relay information to the lowest layer of neurons
- Some of these neurons may fire (in red) in response to this information and in turn relay information to other neurons they are connected to
- These neurons may also fire (again, in red) and the process continues eventually resulting in a response (laughter in this case)



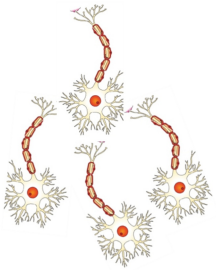
- Of course, in reality, it is not just a single neuron which does all this
- There is a massively parallel interconnected network of neurons
- The sense organs relay information to the lowest layer of neurons
- Some of these neurons may fire (in red) in response to this information and in turn relay information to other neurons they are connected to
- These neurons may also fire (again, in red) and the process continues eventually resulting in a response (laughter in this case)
- An average human brain has around 10^{11} (100 billion) neurons!



- This massively parallel network also ensures that there is division of work



- This massively parallel network also ensures that there is division of work
- Each neuron may perform a certain role or respond to a certain stimulus



A simplified illustration

- This massively parallel network also ensures that there is division of work
- Each neuron may perform a certain role or respond to a certain stimulus

A simplified illustration

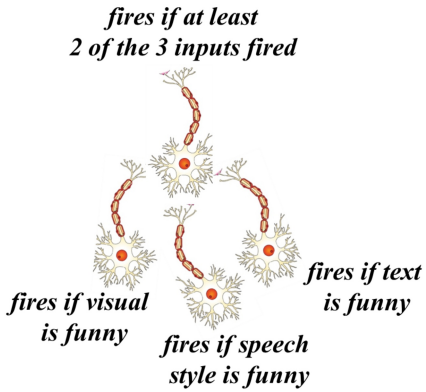
- This massively parallel network also ensures that there is division of work
- Each neuron may perform a certain role or respond to a certain stimulus

A simplified illustration

- This massively parallel network also ensures that there is division of work
- Each neuron may perform a certain role or respond to a certain stimulus

A simplified illustration

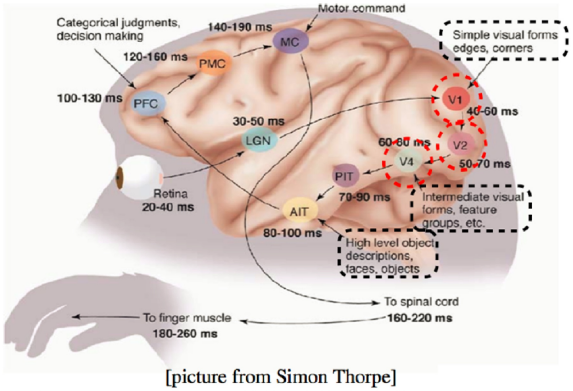
- This massively parallel network also ensures that there is division of work
- Each neuron may perform a certain role or respond to a certain stimulus



A simplified illustration

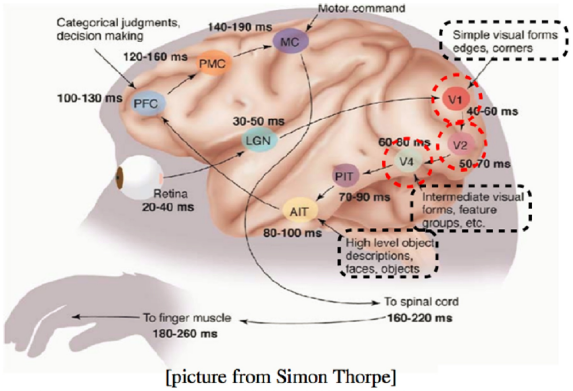
- This massively parallel network also ensures that there is division of work
- Each neuron may perform a certain role or respond to a certain stimulus

- The neurons in the brain are arranged in a hierarchy



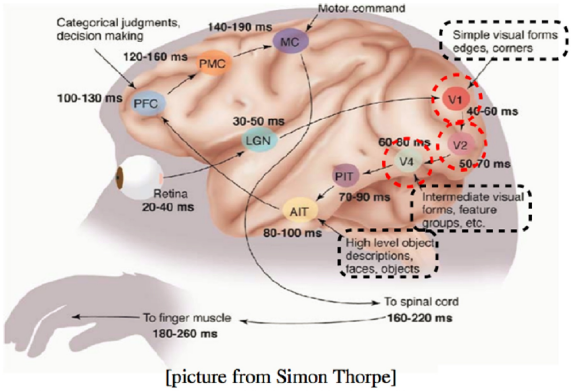
[picture from Simon Thorpe]

- The neurons in the brain are arranged in a hierarchy
- We illustrate this with the help of visual cortex (part of the brain) which deals with processing visual information



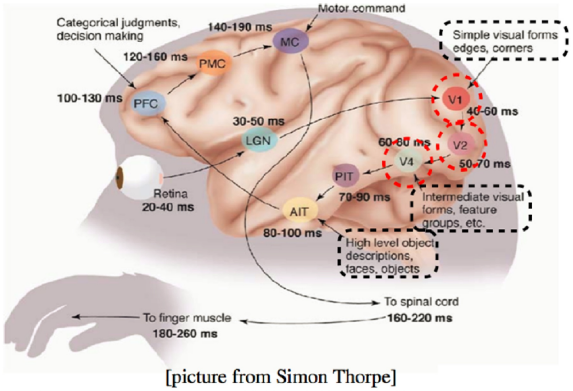
[picture from Simon Thorpe]

- The neurons in the brain are arranged in a hierarchy
- We illustrate this with the help of visual cortex (part of the brain) which deals with processing visual information
- Starting from the retina, the information is relayed to several layers (follow the arrows)



[picture from Simon Thorpe]

- The neurons in the brain are arranged in a hierarchy
- We illustrate this with the help of visual cortex (part of the brain) which deals with processing visual information
- Starting from the retina, the information is relayed to several layers (follow the arrows)
- We observe that the layers $V1$, $V2$ to AIT form a hierarchy (from identifying simple visual forms to high level objects)

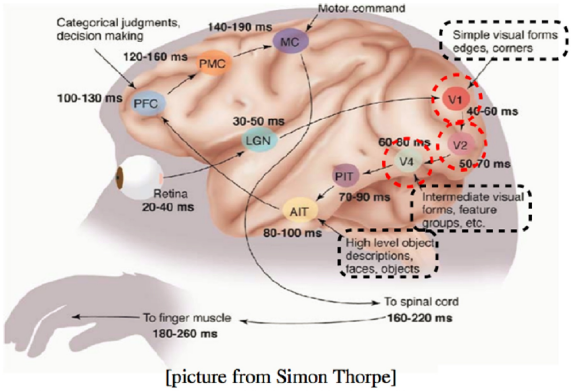


[picture from Simon Thorpe]

Layer 1: detect edges & corners

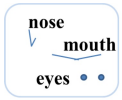
Sample illustration of hierarchical processing*

*Idea borrowed from Hugo Larochelle's lecture slides



[picture from Simon Thorpe]

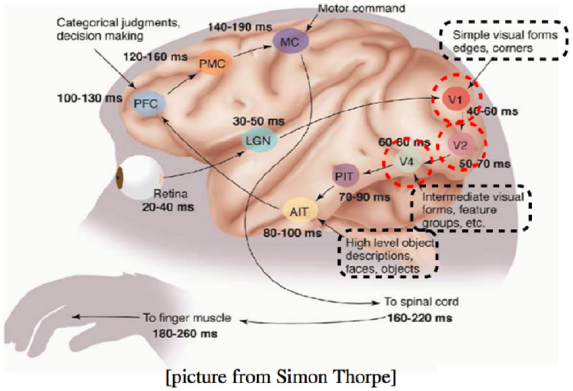
Layer 1: detect edges & corners



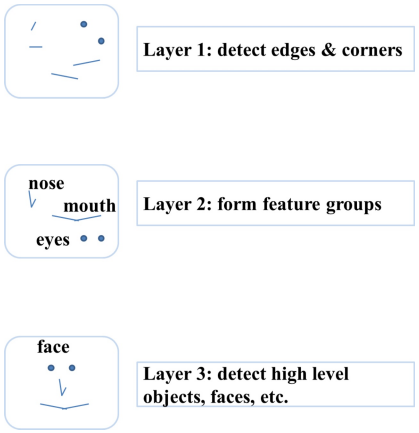
Layer 2: form feature groups

Sample illustration of hierarchical processing*

*Idea borrowed from Hugo Larochelle's lecture slides



[picture from Simon Thorpe]



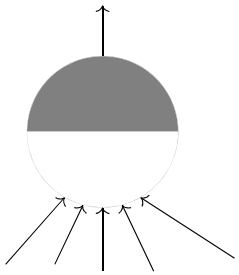
Sample illustration of hierarchical processing*

*Idea borrowed from Hugo Larochelle's lecture slides

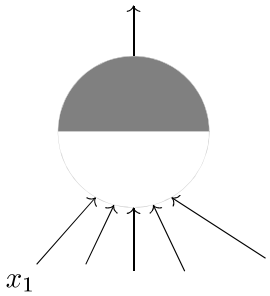
Disclaimer

- I understand very little about how the brain works!
- What you saw so far is an overly simplified explanation of how the brain works!
- But this explanation suffices for the purpose of this course!

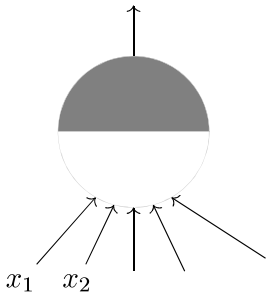
Module 2.2: McCulloch Pitts Neuron



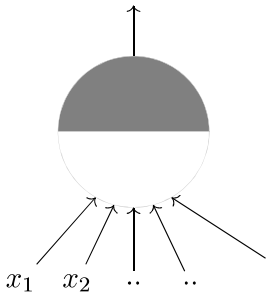
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)



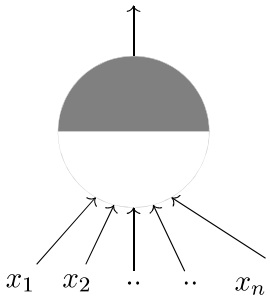
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)



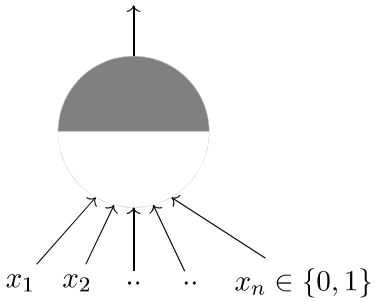
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)



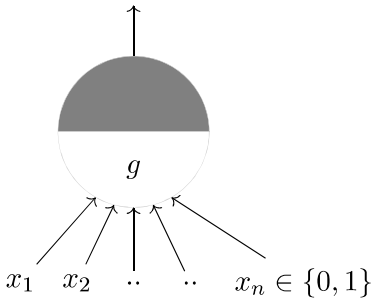
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)



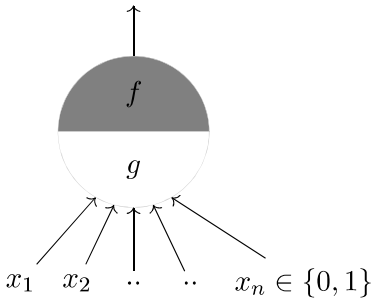
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)



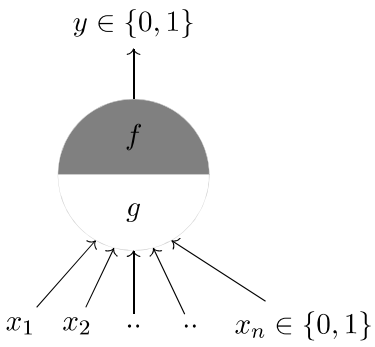
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)



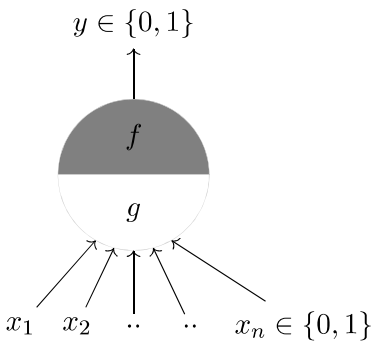
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs



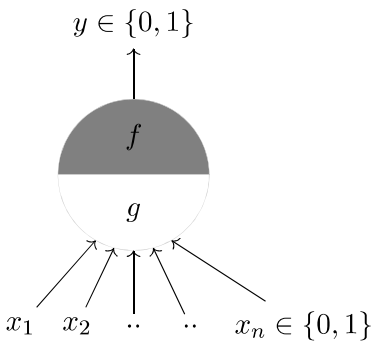
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs and the function f takes a decision based on this aggregation



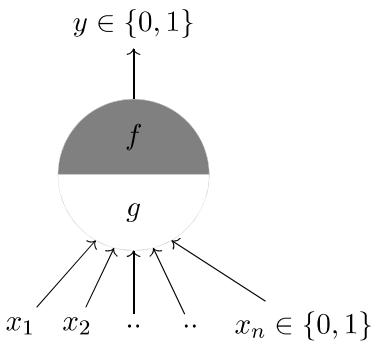
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs and the function f takes a decision based on this aggregation



- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs and the function f takes a decision based on this aggregation
- The inputs can be excitatory or inhibitory

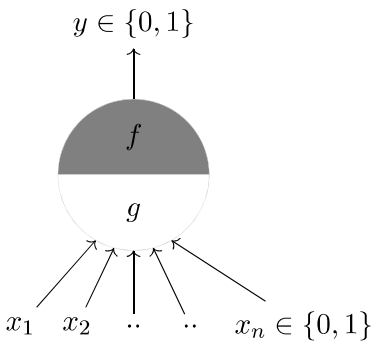


- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs and the function f takes a decision based on this aggregation
- The inputs can be excitatory or inhibitory
- $y = 0$ if any x_i is inhibitory, else



- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs and the function f takes a decision based on this aggregation
- The inputs can be excitatory or inhibitory
- $y = 0$ if any x_i is inhibitory, else

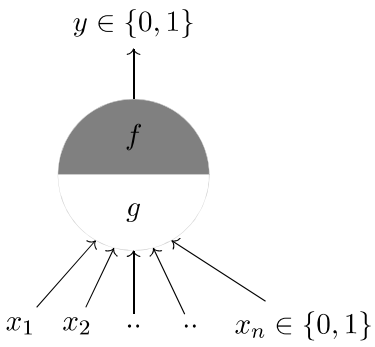
$$g(x_1, x_2, ..., x_n) = g(\mathbf{x}) = \sum_{i=1}^{n} x_i$$



- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs and the function f takes a decision based on this aggregation
- The inputs can be excitatory or inhibitory
- $y = 0$ if any x_i is inhibitory, else

$$g(x_1, x_2, ..., x_n) = g(\mathbf{x}) = \sum_{i=1}^{n} x_i$$

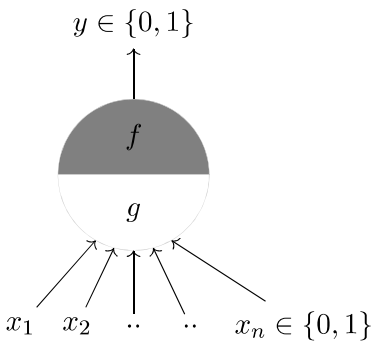
$$y = f(g(\mathbf{x})) = 1 \quad if \quad g(\mathbf{x}) \geq \theta$$



- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs and the function f takes a decision based on this aggregation
- The inputs can be excitatory or inhibitory
- $y = 0$ if any x_i is inhibitory, else

$$g(x_1, x_2, ..., x_n) = g(\mathbf{x}) = \sum_{i=1}^{n} x_i$$

$$\begin{aligned} y = f(g(\mathbf{x})) &= 1 \quad if \quad g(\mathbf{x}) \geq \theta \\ &= 0 \quad if \quad g(\mathbf{x}) < \theta \end{aligned}$$

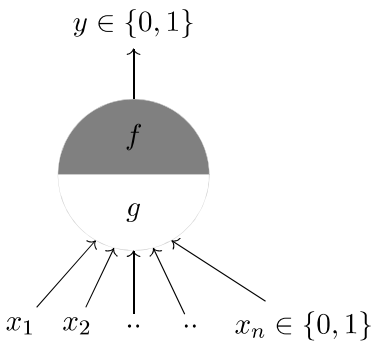


- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs and the function f takes a decision based on this aggregation
- The inputs can be excitatory or inhibitory
- $y = 0$ if any x_i is inhibitory, else

$$g(x_1, x_2, ..., x_n) = g(\mathbf{x}) = \sum_{i=1}^{n} x_i$$

$$\begin{aligned} y = f(g(\mathbf{x})) &= 1 \quad if \quad g(\mathbf{x}) \geq \theta \\ &= 0 \quad if \quad g(\mathbf{x}) < \theta \end{aligned}$$

- θ is called the thresholding parameter



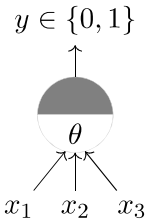
- McCulloch (neuroscientist) and Pitts (logician) proposed a highly simplified computational model of the neuron (1943)
- g aggregates the inputs and the function f takes a decision based on this aggregation
- The inputs can be excitatory or inhibitory
- $y = 0$ if any x_i is inhibitory, else

$$g(x_1, x_2, ..., x_n) = g(\mathbf{x}) = \sum_{i=1}^{n} x_i$$

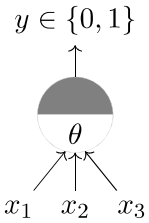
$$\begin{aligned} y = f(g(\mathbf{x})) &= 1 \quad if \quad g(\mathbf{x}) \geq \theta \\ &= 0 \quad if \quad g(\mathbf{x}) < \theta \end{aligned}$$

- θ is called the thresholding parameter
- This is called Thresholding Logic

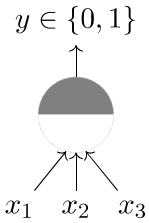
Let us implement some boolean functions using this McCulloch Pitts (MP) neuron ...



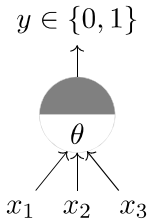
A McCulloch Pitts unit



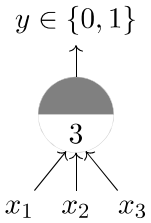
A McCulloch Pitts unit



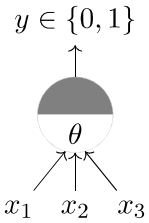
AND function



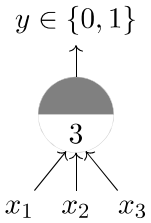
A McCulloch Pitts unit



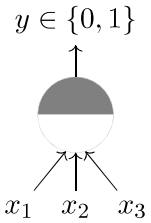
AND function



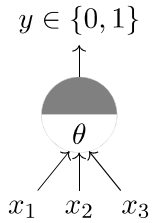
A McCulloch Pitts unit



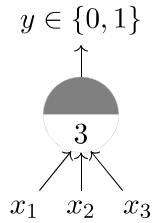
AND function



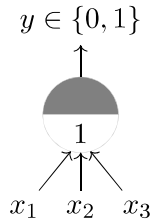
OR function



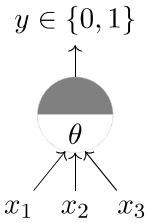
A McCulloch Pitts unit



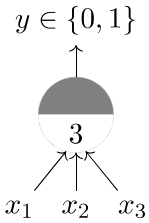
AND function



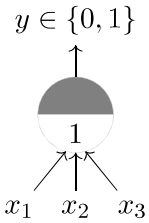
OR function



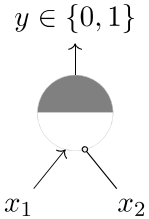
A McCulloch Pitts unit



AND function

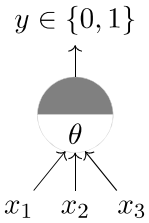


OR function

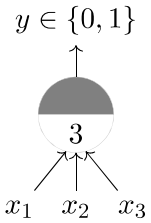


x_1 AND $!x_2$*

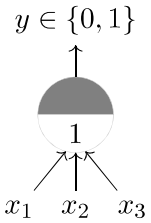
*circle at the end indicates inhibitory input: if any inhibitory input is 1 the output will be 0



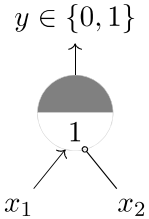
A McCulloch Pitts unit



AND function

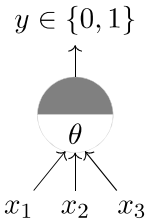


OR function

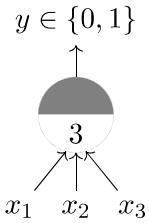


x_1 AND $!x_2$*

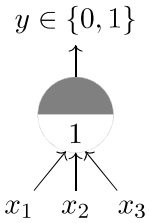
*circle at the end indicates inhibitory input: if any inhibitory input is 1 the output will be 0



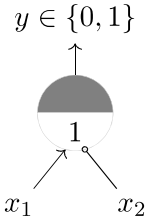
A McCulloch Pitts unit



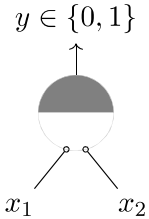
AND function



OR function

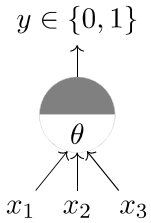


x_1 AND $!x_2$*

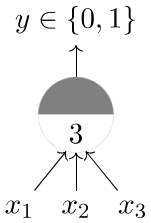


NOR function

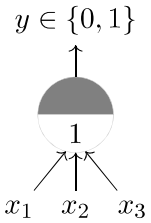
*circle at the end indicates inhibitory input: if any inhibitory input is 1 the output will be 0



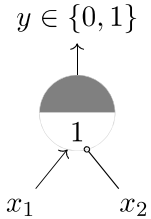
A McCulloch Pitts unit



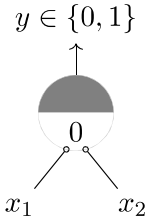
AND function



OR function

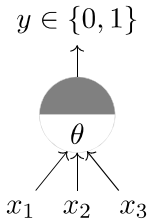


x_1 AND $!x_2$*

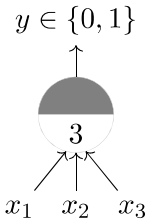


NOR function

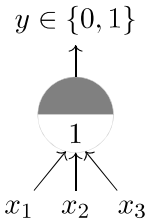
*circle at the end indicates inhibitory input: if any inhibitory input is 1 the output will be 0



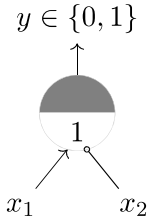
A McCulloch Pitts unit



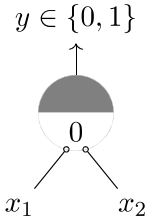
AND function



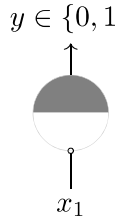
OR function



x_1 AND $!x_2$*

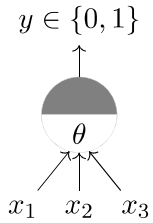


NOR function

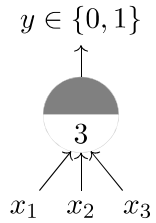


NOT function

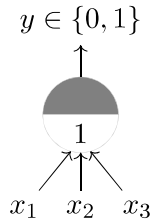
*circle at the end indicates inhibitory input: if any inhibitory input is 1 the output will be 0



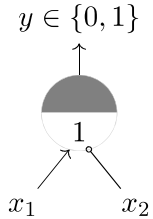
A McCulloch Pitts unit



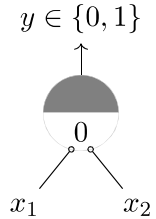
AND function



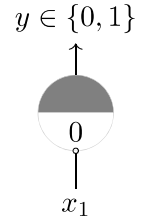
OR function



x_1 AND $!x_2$*



NOR function



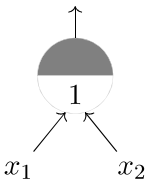
NOT function

*circle at the end indicates inhibitory input: if any inhibitory input is 1 the output will be 0

- Can any boolean function be represented using a McCulloch Pitts unit ?

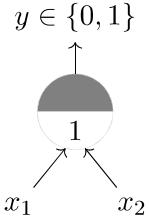
- Can any boolean function be represented using a McCulloch Pitts unit ?
- Before answering this question let us first see the geometric interpretation of a MP unit ...

$y \in \{0, 1\}$



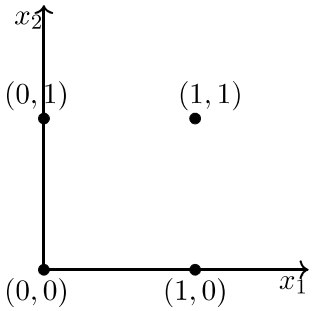
OR function

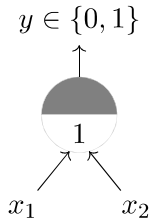
$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 1$



OR function

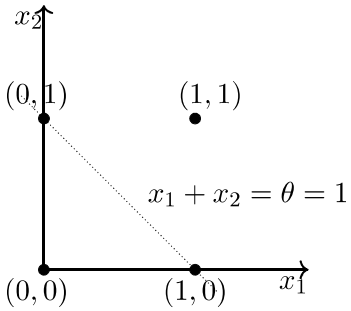
$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 1$

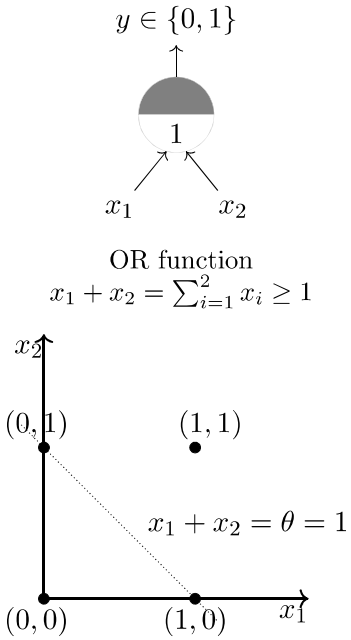




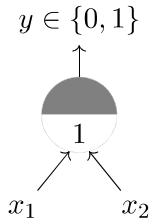
OR function

$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 1$



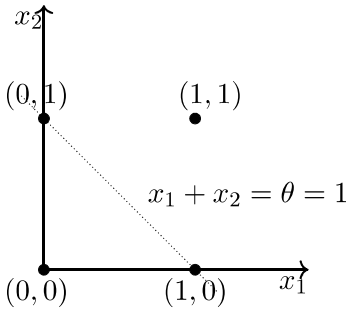


- A single MP neuron splits the input points (4 points for 2 binary inputs) into two halves

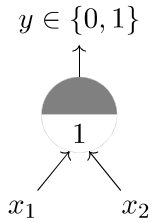


OR function

$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 1$

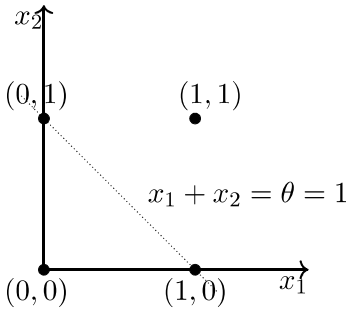


- A single MP neuron splits the input points (4 points for 2 binary inputs) into two halves
- Points lying on or above the line $\sum_{i=1}^{n} x_i - \theta = 0$ and points lying below this line

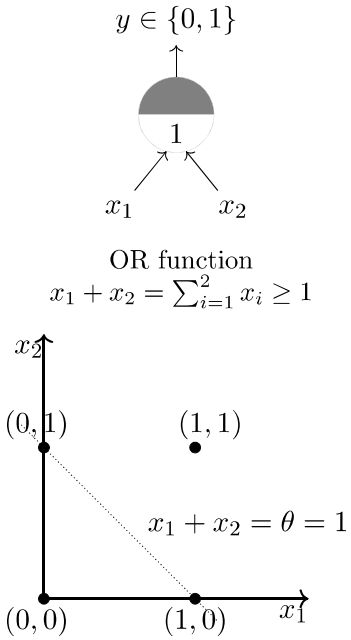


OR function

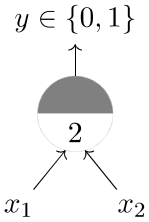
$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 1$



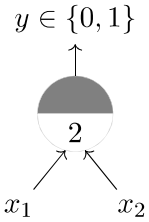
- A single MP neuron splits the input points (4 points for 2 binary inputs) into two halves
- Points lying on or above the line $\sum_{i=1}^{n} x_i - \theta = 0$ and points lying below this line
- In other words, all inputs which produce an output 0 will be on one side ($\sum_{i=1}^{n} x_i < \theta$) of the line and all inputs which produce an output 1 will lie on the other side ($\sum_{i=1}^{n} x_i \geq \theta$) of this line



- A single MP neuron splits the input points (4 points for 2 binary inputs) into two halves
- Points lying on or above the line $\sum_{i=1}^{n} x_i - \theta = 0$ and points lying below this line
- In other words, all inputs which produce an output 0 will be on one side ($\sum_{i=1}^{n} x_i < \theta$) of the line and all inputs which produce an output 1 will lie on the other side ($\sum_{i=1}^{n} x_i \geq \theta$) of this line
- Let us convince ourselves about this with a few more examples (if it is not already clear from the math)

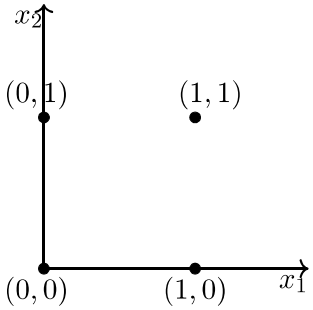


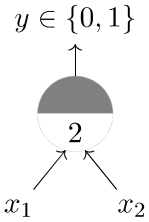
AND function
$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 2$



AND function

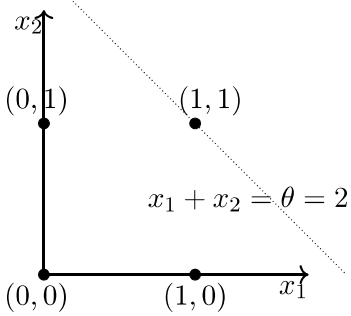
$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 2$

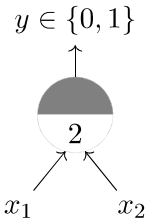




AND function

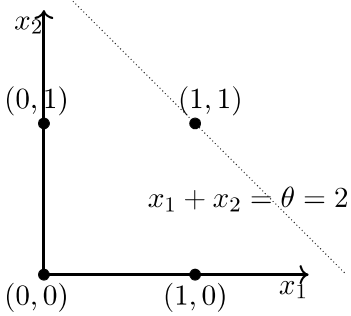
$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 2$

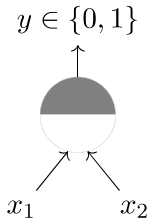




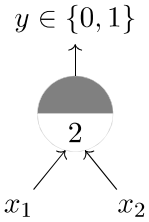
AND function

$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 2$



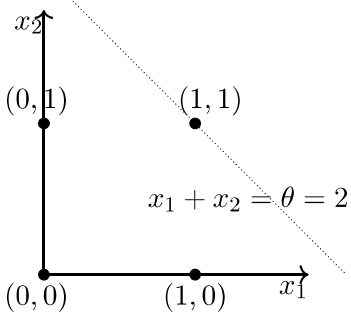


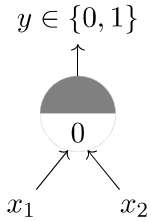
Tautology (always ON)



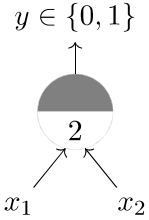
AND function

$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 2$



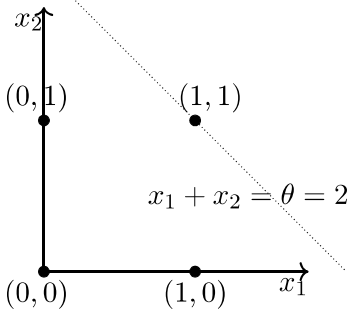


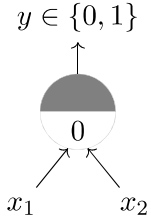
Tautology (always ON)



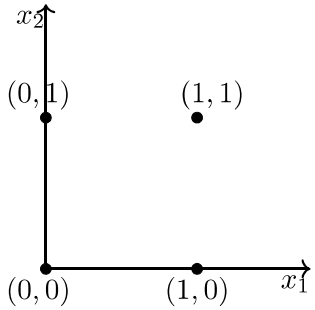
AND function

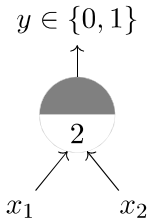
$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 2$





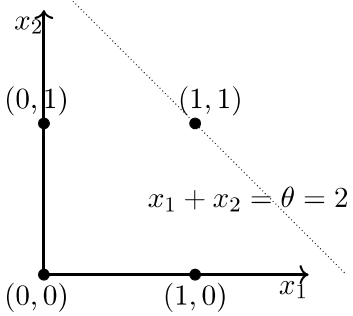
Tautology (always ON)

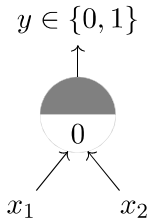




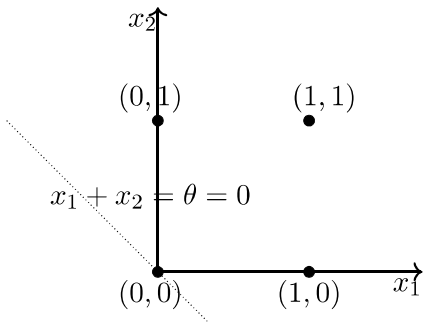
AND function

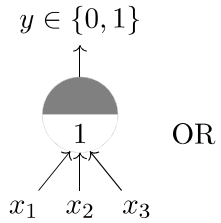
$x_1 + x_2 = \sum_{i=1}^{2} x_i \geq 2$





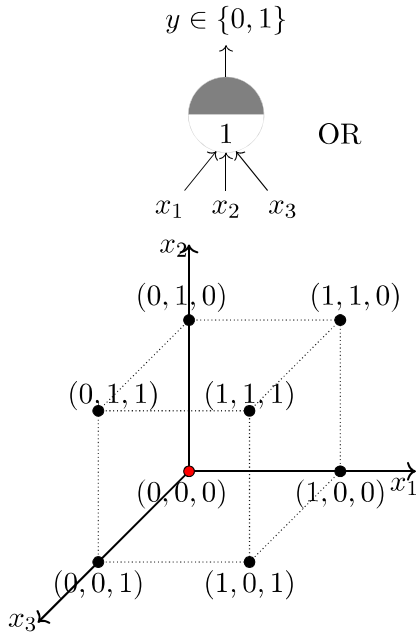
Tautology (always ON)



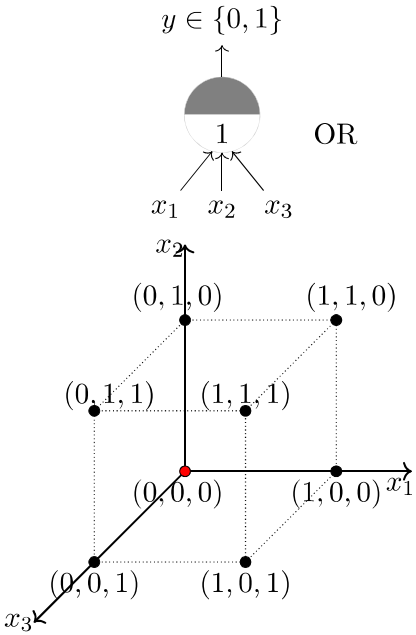


OR

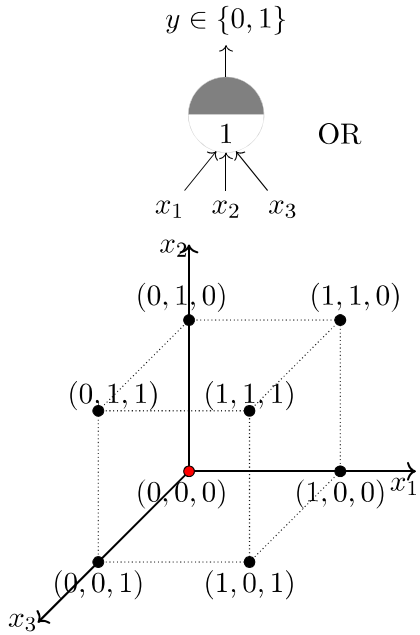
- What if we have more than 2 inputs?



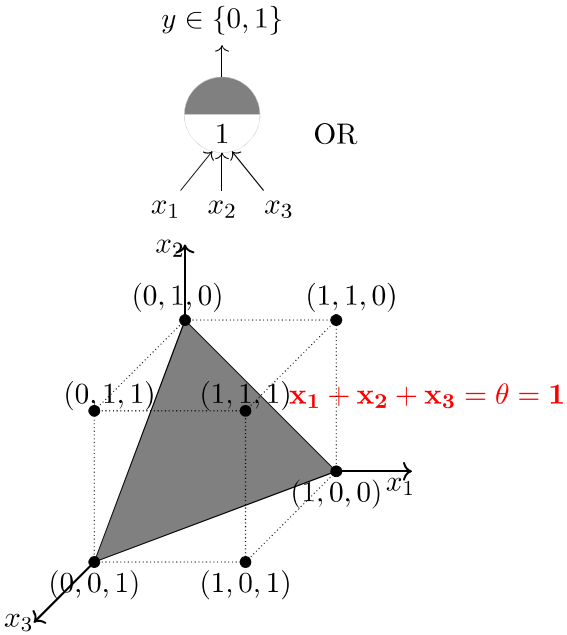
- What if we have more than 2 inputs?



- What if we have more than 2 inputs?
- Well, instead of a line we will have a plane



- What if we have more than 2 inputs?
- Well, instead of a line we will have a plane
- For the OR function, we want a plane such that the point (0,0,0) lies on one side and the remaining 7 points lie on the other side of the plane



- What if we have more than 2 inputs?
- Well, instead of a line we will have a plane
- For the OR function, we want a plane such that the point (0,0,0) lies on one side and the remaining 7 points lie on the other side of the plane

The story so far ...

- A single McCulloch Pitts Neuron can be used to represent boolean functions which are linearly separable

The story so far ...

- A single McCulloch Pitts Neuron can be used to represent boolean functions which are linearly separable
- Linear separability (for boolean functions) : There exists a line (plane) such that all inputs which produce a 1 lie on one side of the line (plane) and all inputs which produce a 0 lie on other side of the line (plane)