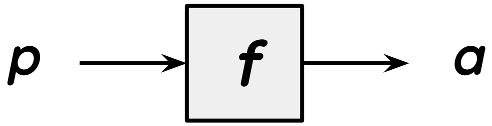
Introduction to Neural Networks

José Carlos Ortiz Bayliss
jcobayliss@itesm.mx

Tecnológico de Monterrey

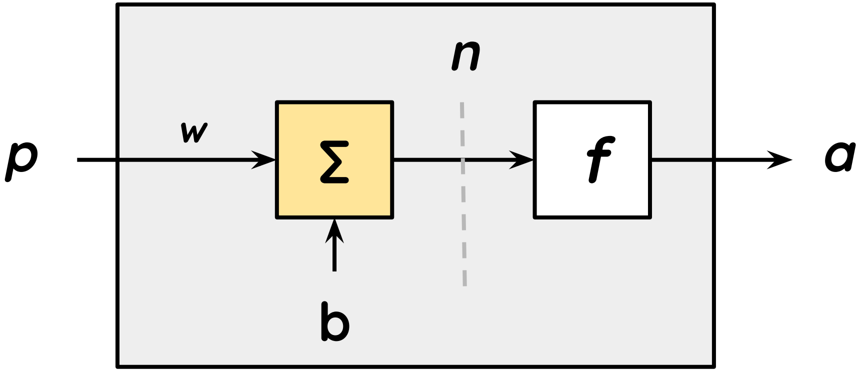
The single-input neuron

The single-input neuron is an extremely simple computational unit that, given an input, produces an output.



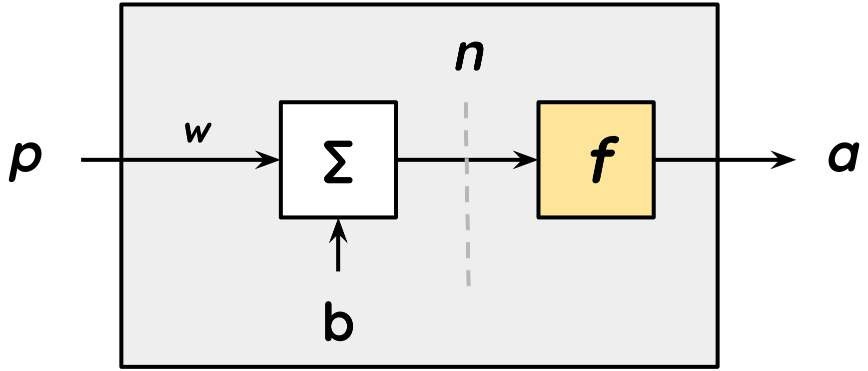
The single-input neuron

- The summer output *n*, often referred to as the **net input,** goes into the activation function *f*, which produces the scalar output *a*.



The single-input neuron

- The summer output *n*, often referred to as the net input, goes into a activation function *f*, which produces the scalar neuron output *a*.
- The neuron output is calculated as $a = f\,(wp+b)$. The actual output depends on the **activation function** that is chosen.



Tecnológico de Monterrey

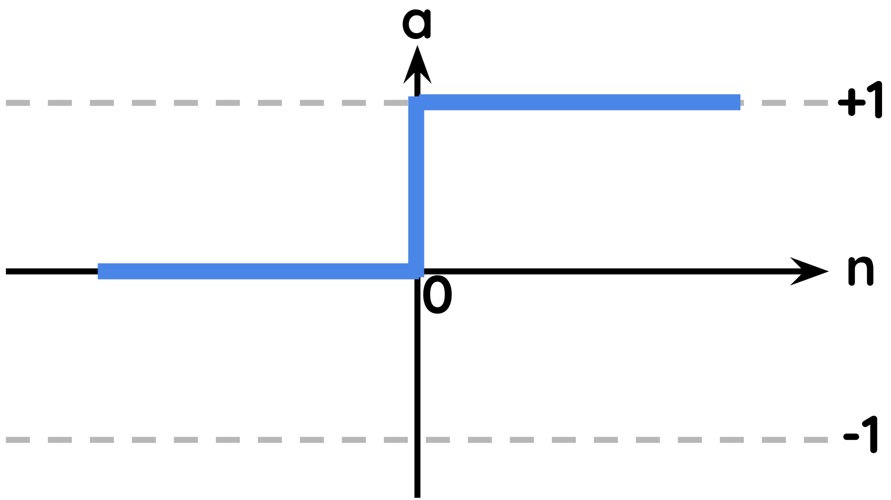
Activation functions

Changing the activation function affects the behaviour of the neuron (and its possible output values):

- **Binary (0 or 1)**: hard limit.
- **Bipolar (-1 or 1)**: symmetrical hard limit.
- **Continuous (in the range -1 to 1)**: symmetrical saturating linear
- **Continuous (in the range 0 to 1)**: saturating linear, sigmoid.
- **Continuous (any value)**: linear.

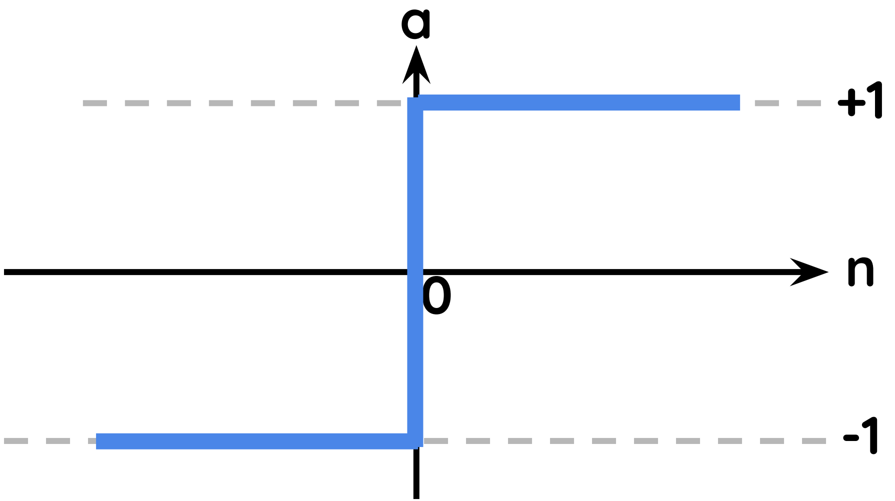
Hard limit

$$a = \begin{cases} 0 & \text{if } n < 0 \\ 1 & \text{if } n \geq 0 \end{cases}$$



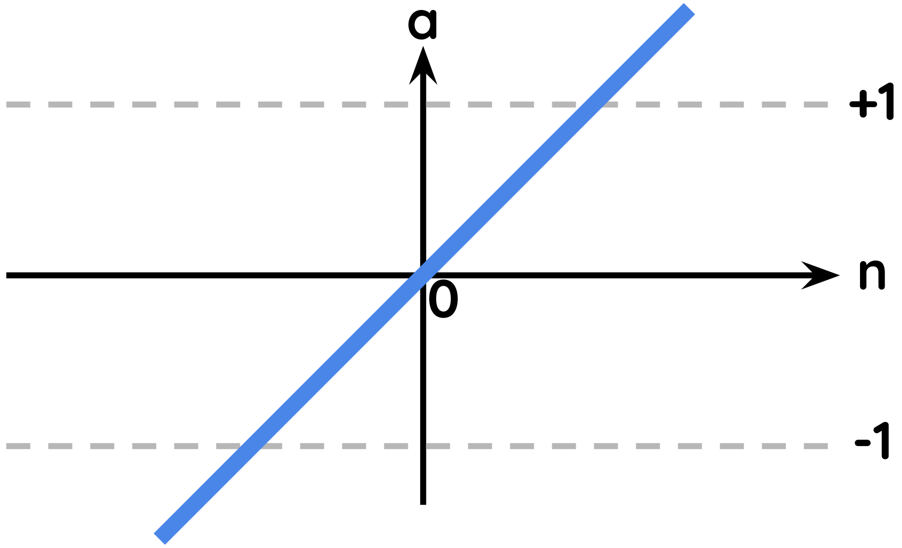
Symmetrical hard limit

$$a = \begin{cases} -1 & \text{if } n < 0 \\ 1 & \text{if } n \geq 0 \end{cases}$$

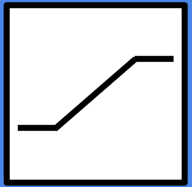


Linear activation function

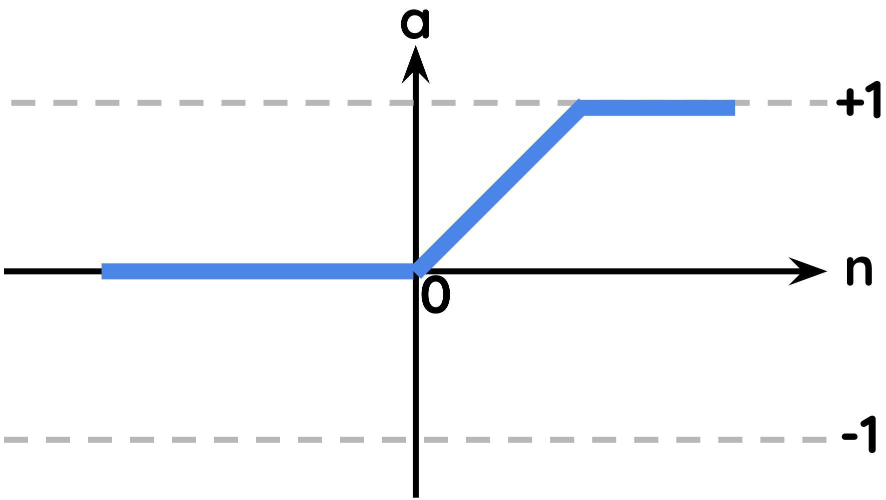
$$a = n$$



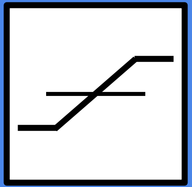
Saturating linear



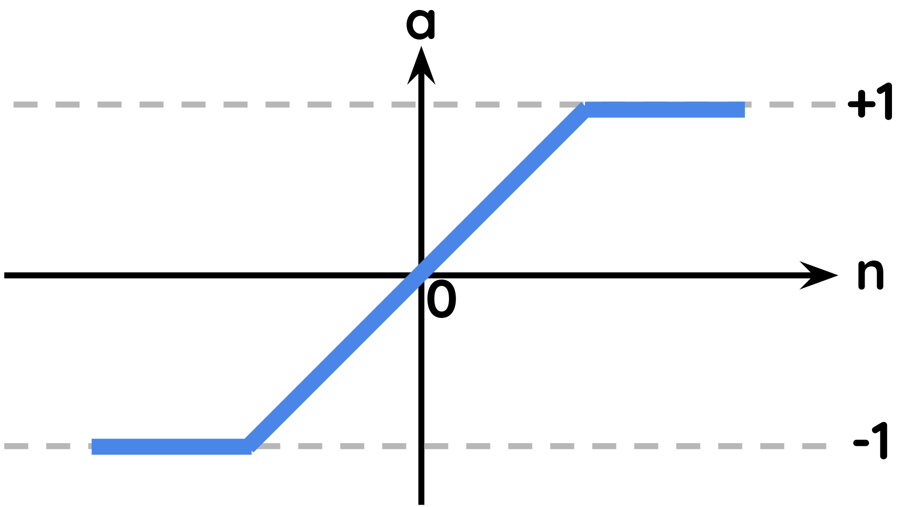
$$a = \begin{cases} 0 & \text{if } n < 0 \\ n & \text{if } 0 \le n \le 1 \\ 1 & \text{if } n > 1 \end{cases}$$



Symmetric saturating linear

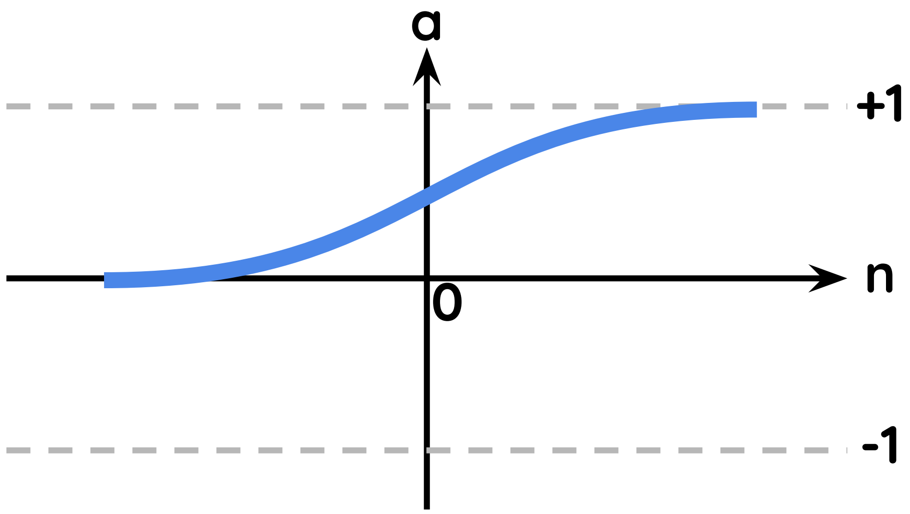


$$a = \begin{cases} -1 & \text{if } n < -1 \\ n & \text{if } -1 \leq n \leq 1 \\ 1 & \text{if } n > 1 \end{cases}$$

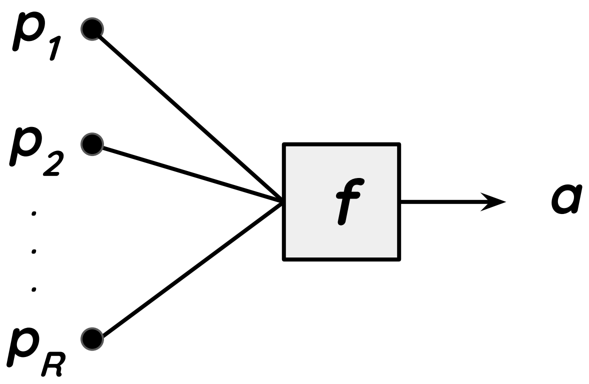


Log-sigmoid

$$a = \frac{1}{1+e^{-n}}$$

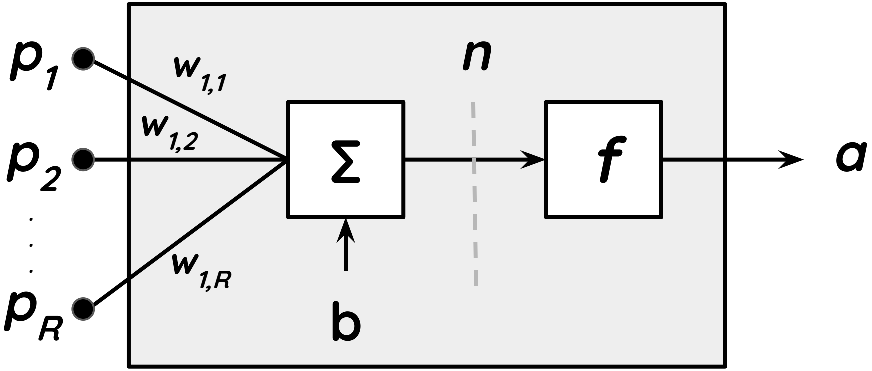


The multiple-input neuron



The multiple-input neuron

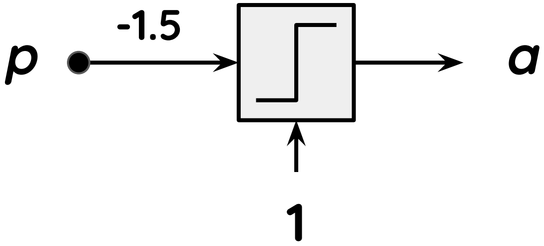
- The summer output n, often referred to as the net input, goes into a activation function f, which produces the scalar neuron output a.
- The neuron output is calculated as $a = f\,(\mathbf{wp}+b)$. The actual output depends on the **activation function** that is chosen.



The perceptron

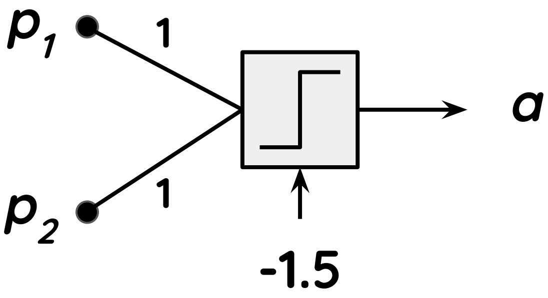
- The perceptron is a neural network with only one layer of neurons (usually using a hard limit activation function).
- Single-neuron perceptrons can only classify input vectors into two categories.
- Unfortunately, the perceptron network is inherently limited (it can only classify classes which are linearly separable!).
- Although limited, the perceptron is still considered an important network because it provides a good basis for understanding more complex networks.

The single-neuron perceptron as a logic gate



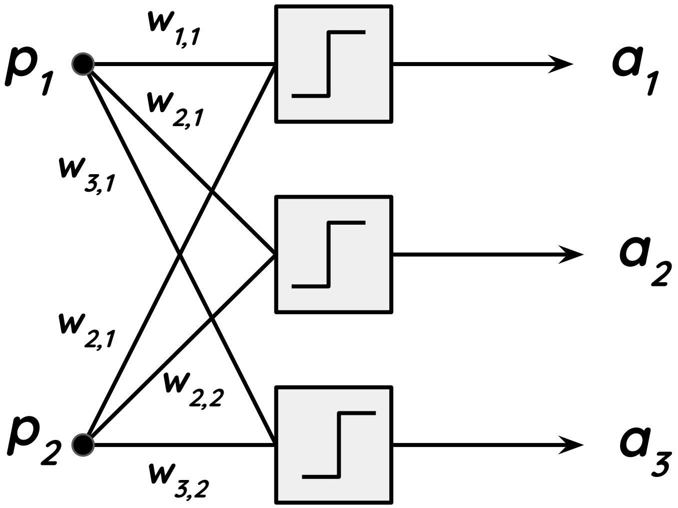
p_1	a
0	
1	

The single-neuron perceptron as a logic gate



p_1	p_2	a
0	0	
0	1	
1	0	
1	1	

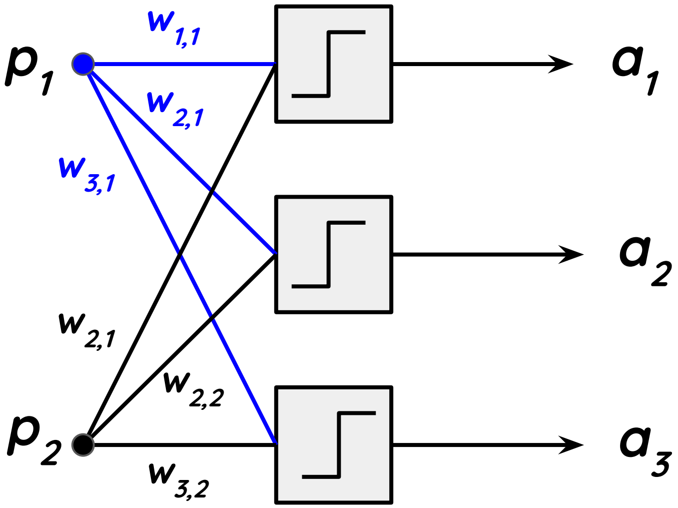
A multi-neuron perceptron



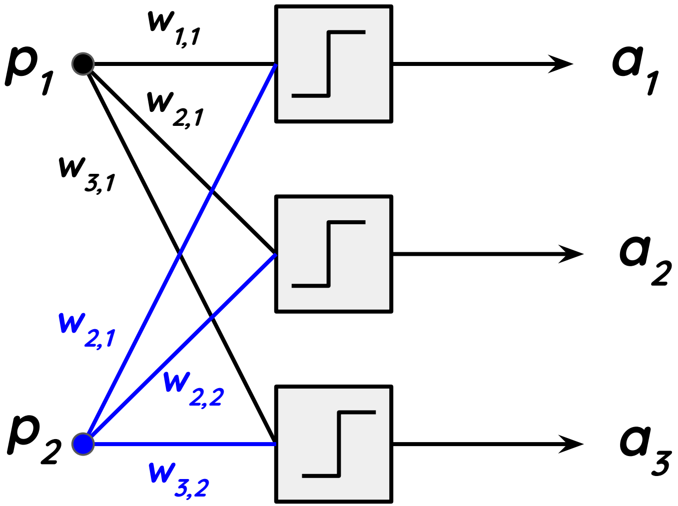
Tecnológico de Monterrey

The multi-neuron perceptron

$W_{TO,\ FROM}$

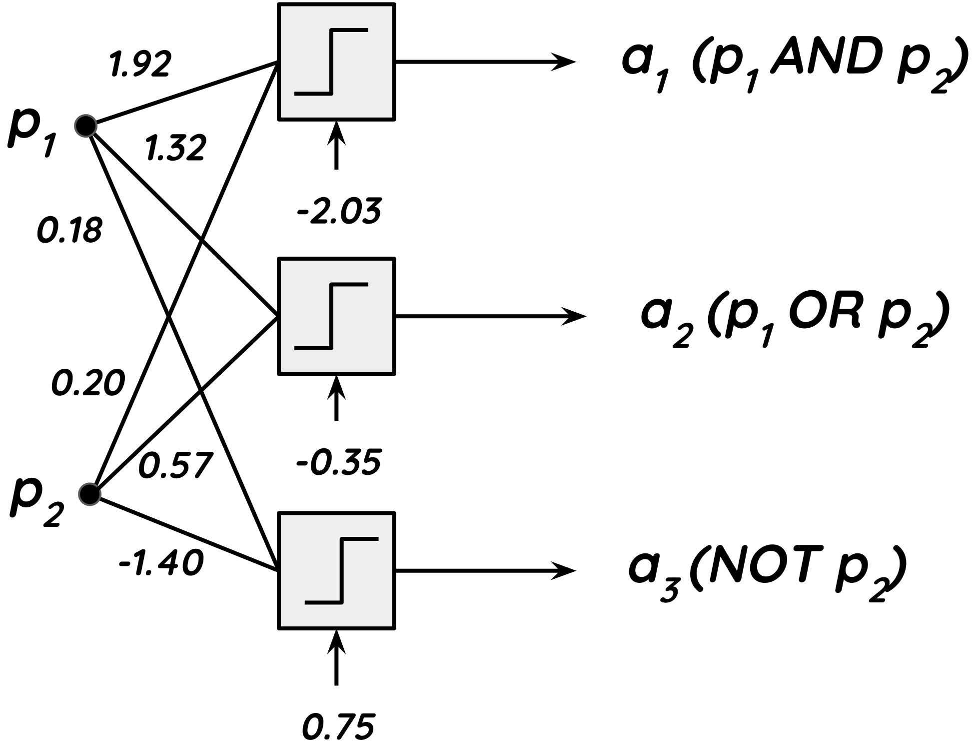


The multi-neuron perceptron



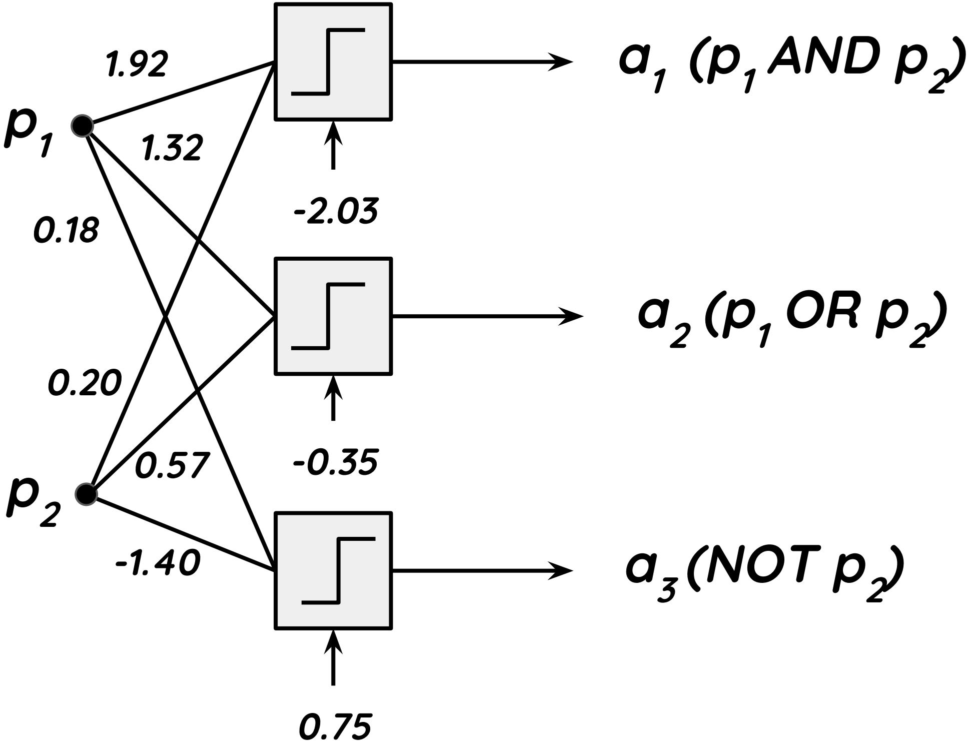
Tecnológico de Monterrey

Multiple logic gates in one perceptron



Tecnológico de Monterrey

Multiple logic gates in one perceptron

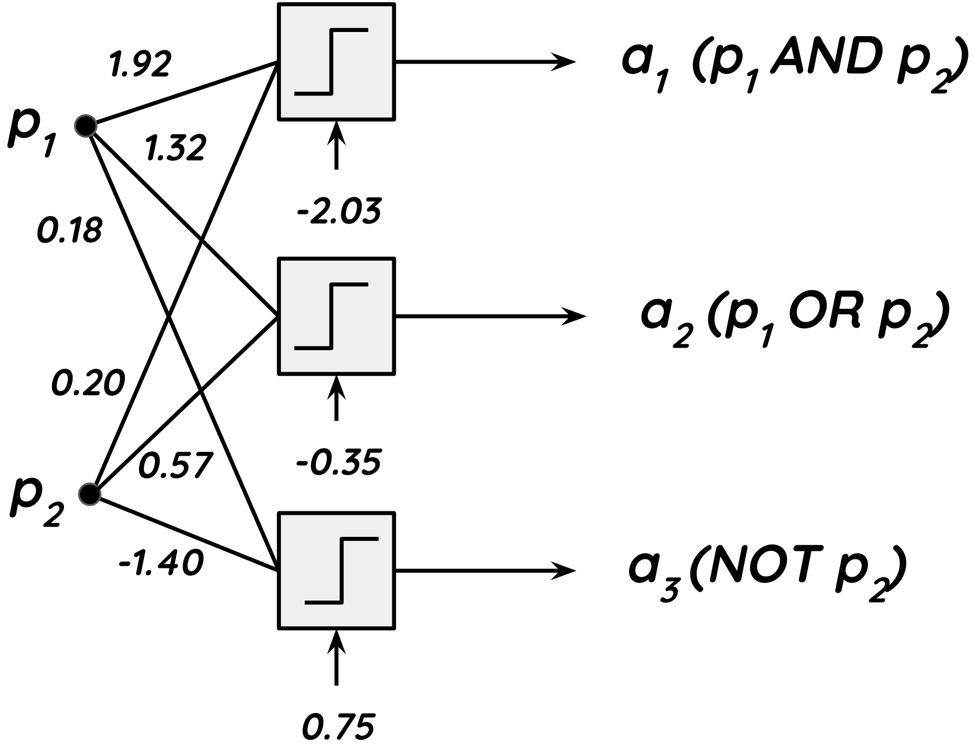


$$W = \begin{bmatrix} 1.92 & 0.20 \\ 1.32 & 0.57 \\ 0.18 & -1.40 \end{bmatrix}$$

$$b = \begin{bmatrix} -2.03 \\ -0.35 \\ 0.75 \end{bmatrix}$$

Tecnológico de Monterrey

Multiple logic gates in one perceptron



p_1	p_2	a_1	a_2	a_3
0	0	0	0	1
0	1	0	1	0
1	0	0	1	1
1	1	1	1	0

Tecnológico de Monterrey

How can a perceptron be trained?

While we do not reach a termination criterion:

- Calculate the output of the network:

  $a = f(\mathbf{Wp} + b)$

- Is there an error (expected output vs actual output)?
  - *Yes*: Update the weights and biases:

    $\mathbf{W} = \mathbf{W} + \mathbf{e}\mathbf{p}^T$ and $\mathbf{b} = \mathbf{b} + \mathbf{e}$

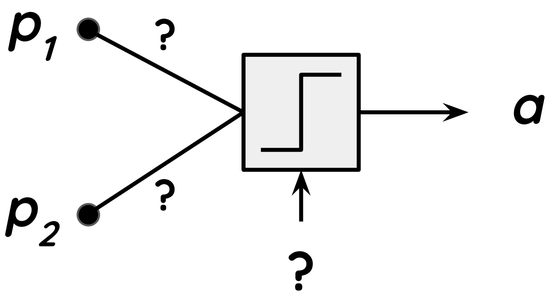
  - *No*: Do nothing.

Perceptron's learning rule in action

- In this example, we will calculate the weights and bias of a single-neuron perceptron that will be used as an AND gate.
- The perceptron receives two inputs and produces one output (the AND of the two inputs).

Perceptron's learning rule in action

- In this example, we will calculate the weights and bias of a single-neuron perceptron that will be used as an AND gate.
- The perceptron receives two inputs and produces one output (the AND of the two inputs).



p_1	p_2	a
0	0	0
0	1	0
1	0	0
1	1	1

Tecnológico de Monterrey