utad

Inteligência Artificial

Redes Neuronais Artificiais (Parte I)

Paulo Moura Oliveira

Departamento de Engenharias

Gabinete F2.15, ECT-1

UTAD

email: oliveira@utad.pt

Introdução

Porquê Utilizar Redes Neuronais Artificiais?

✓ Atualmente as redes neuronais artificiais são um dos paradigmas da Inteligência Artificial e dos Sistemas Inteligentes.

Algumas (potenciais) vantagens das redes neuronais:

1. Não necessitam de ser programadas, pois **podem aprender a partir de exemplos**.
2. A sua utilização pode-se **generalizar para dados que não foram utilizados no seu treino**.
3. São tolerantes a falhas: podem reproduzir saídas corretas a partir de **dados com ruído ou incompletos**.
4. São **rápidas**: as suas unidades de processamento trabalham em paralelo.
5. São relativamente "simples" de construir e treinar.

Introdução

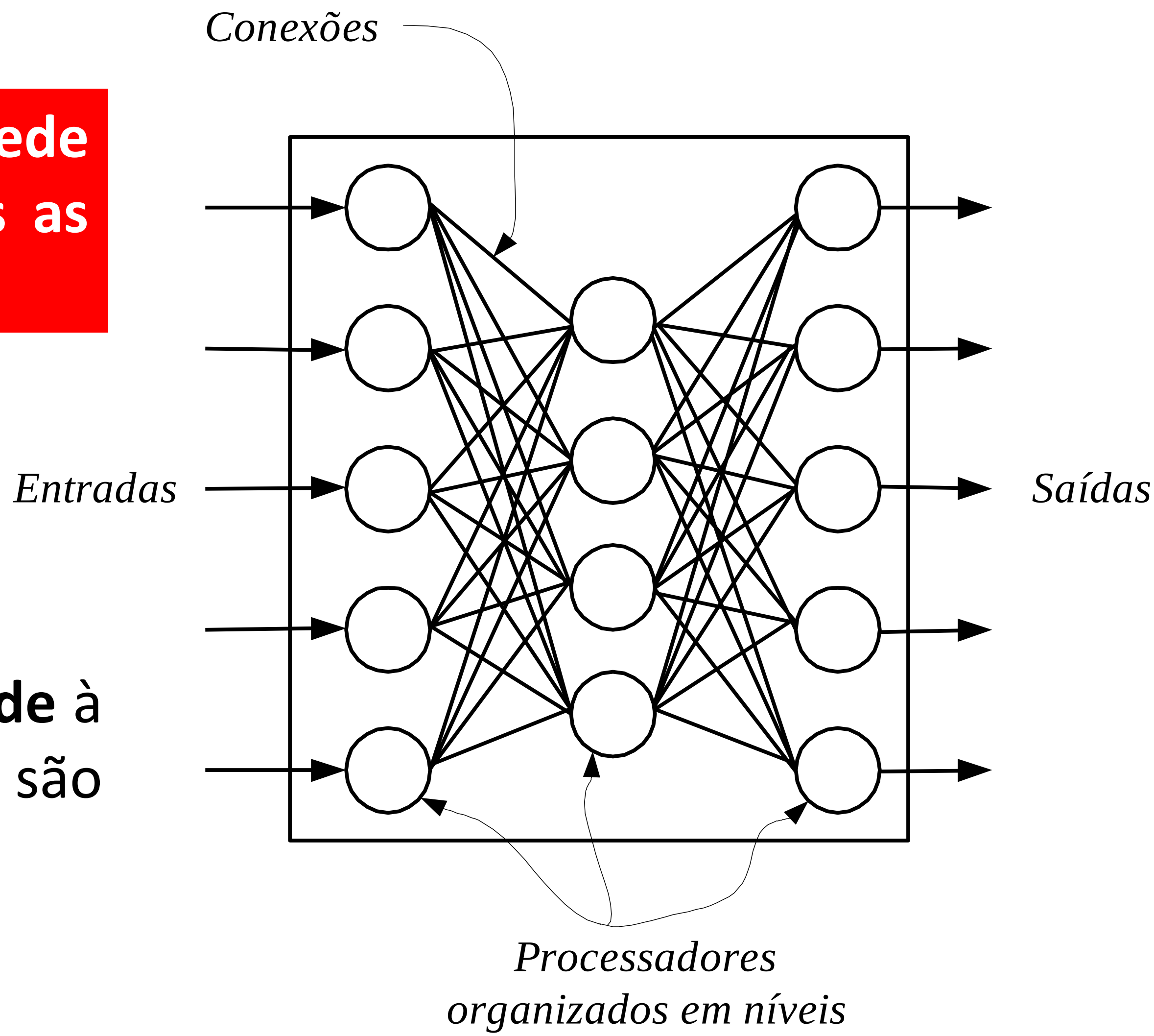
Quais as suas aplicações nas Engenharias?

- Sistemas que detetam explosivos nas portas dos aeroportos.
- Reconhecimento automático de caracteres em sistemas de leitura.
- Sistemas de visão em robótica.
- Sistemas de reconhecimento de voz (e.g.: computadores e sistemas telefónicos).
- Sistemas de previsão (e.g.: clima , investimentos financeiros, etc.)
- Reconhecimento de padrões.

utad

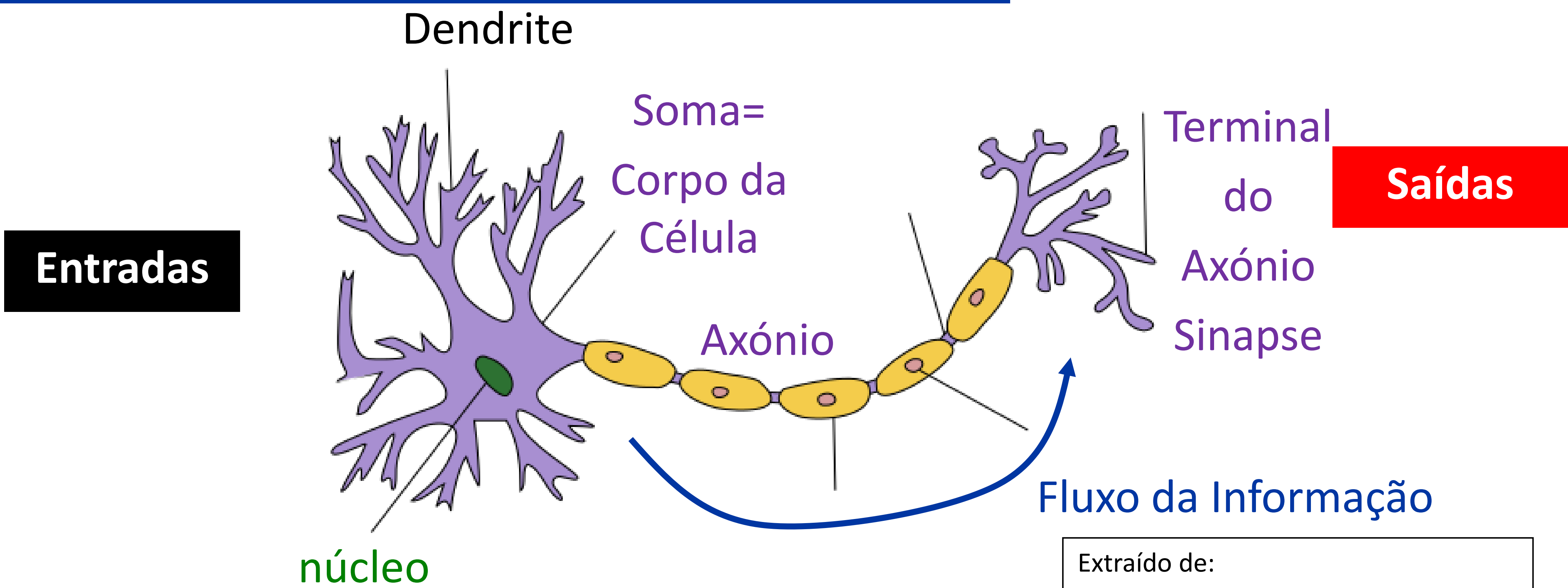
Arquitetura geral de uma Rede Neuronal

O poder computacional da rede advém das conexões entres as múltiplas unidades.



- Chama-se **Topologia da Rede** à forma como as unidades são ligadas.

Inspiração Natural: Estrutura de um neunónio

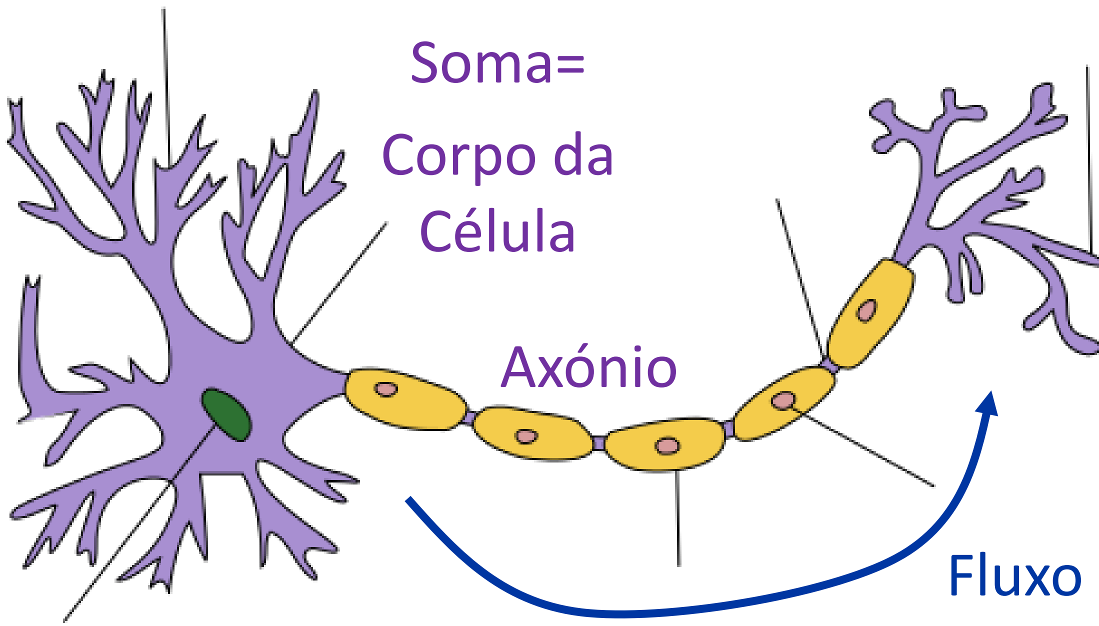


Extraído de:
https://en.wikipedia.org/wiki/Dendrite
Acedido em 11-4-2020

Os neurónios estão conectados por sinapses, que produzem uma resposta química a “uma” entrada.

Introdução

Inspiração Natural: Estrutura de um neunónio



Extraído de:
https://en.wikipedia.org/wiki/Dendrite
Acedido em 11-4-2020

O tamanho da resposta varia de acordo com a soma das reações da sinapse.

Se esse valor ultrapassar um determinado valor o neurónio dispara.

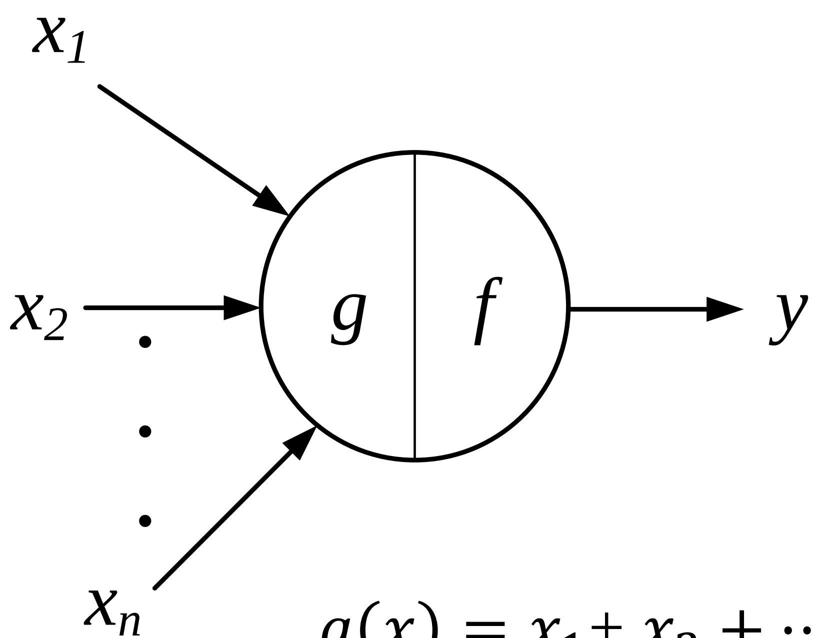
Unidade Neuronal Artificial

Os primeiros modelos para representar neurónios foram desenvolvidos por Mculloch e Pitts.

As entradas e saídas são variáveis binárias!

$$x = (x_1, x_2, \cdots, x_n)$$

$$x_n \in [0,1]$$

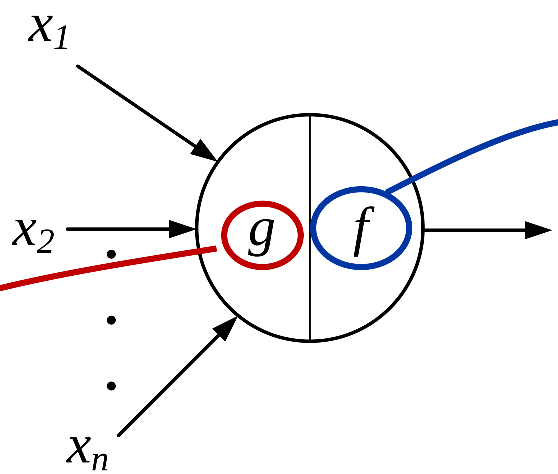


$$y \in [0,1]$$

$$g(x) = x_1 + x_2 + \cdots + x_n$$

Unidade Neuronal Artificial

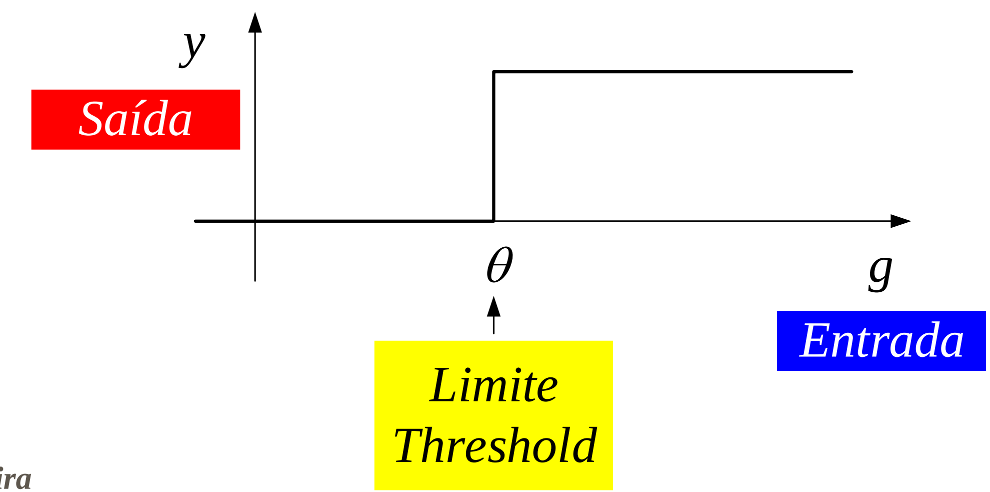
$$x = (x_1, x_2, \cdots, x_n)$$



$$y = f\big(g(x)\big) = \begin{cases} 1 \Leftarrow g(x) \geq \theta \\ 0 \Leftarrow g(x) < \theta \end{cases}$$

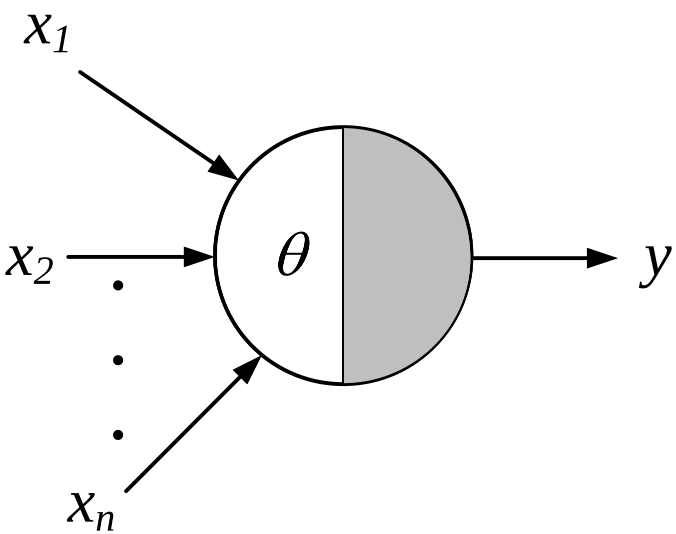
$$g(x) = x_1 + x_2 + \cdots + x_n$$

f **- Função de Ativação**



Neurónio de Mculloch e Pitts

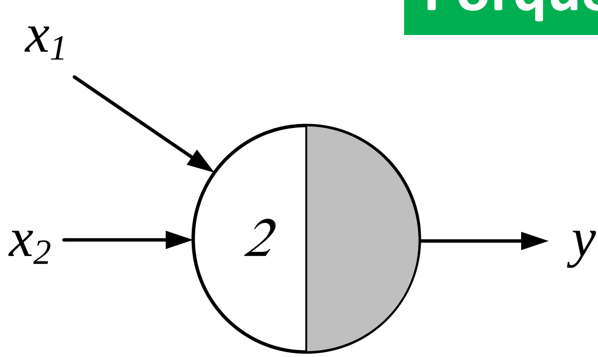
✓ Representação alternativa:



$$y = f\big(g(x)\big) = \begin{cases} 1 \Leftarrow g(x) \geq \theta \\ 0 \Leftarrow g(x) < \theta \end{cases}$$

$$g(x) = x_1 + x_2 + \cdots + x_n$$

Função AND

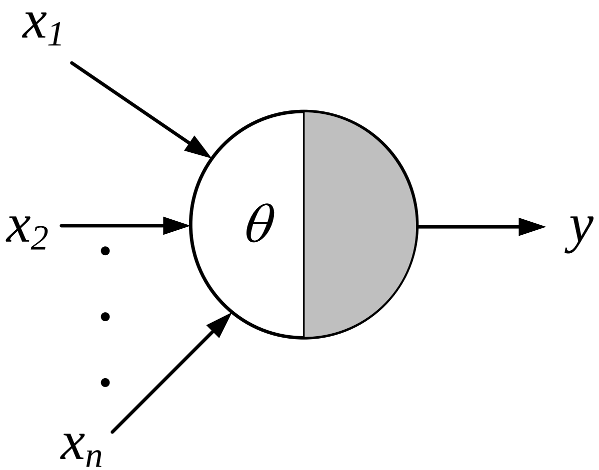


Porquê?

x1	x2	g	y
0	0		
1	0		
0	1		
1	1		

Unidade Neuronal Artificial

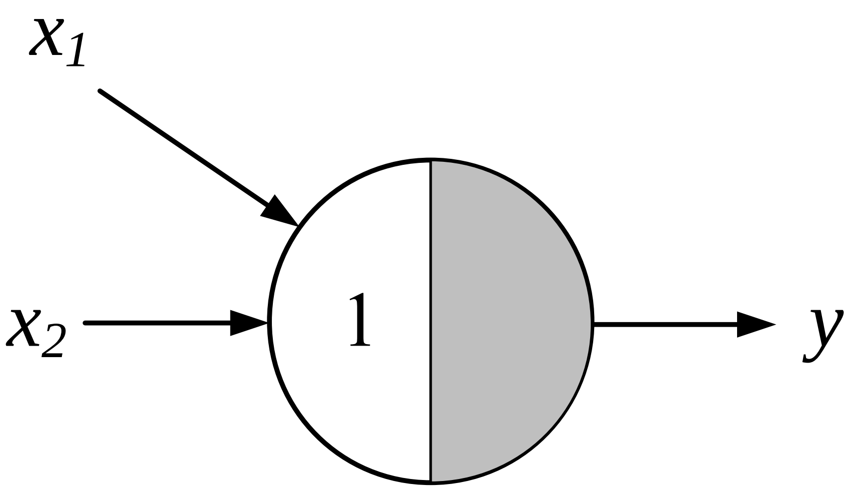
Neurónio de Mculloch e Pitts



$$y = f\big(g(x)\big) = \begin{cases} 1 \Leftarrow g(x) \geq \theta \\ 0 \Leftarrow g(x) < \theta \end{cases}$$

$$g(x) = x_1 + x_2 + \cdots + x_n$$

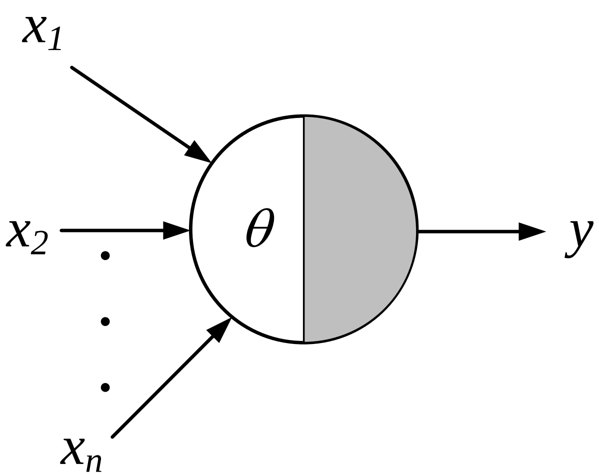
Função OR



Porquê?

x1	**x2**	**g**	**y**
0	*0*		
1	*0*		
0	*1*		
1	*1*		

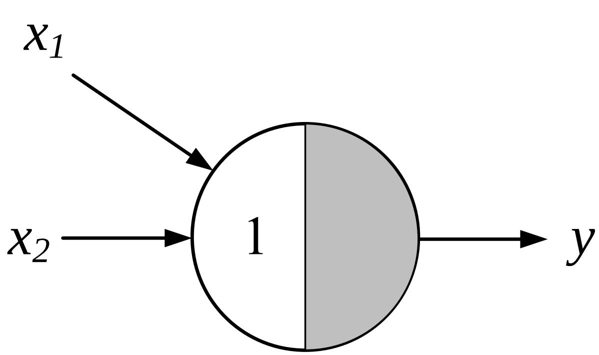
Neurónio de Mculloch e Pitts



$$y = f\big(g(x)\big) = \begin{cases} 1 \Leftarrow g(x) \geq \theta \\ 0 \Leftarrow g(x) < \theta \end{cases}$$

$$g(x) = x_1 + x_2 + \cdots + x_n$$

Função OR

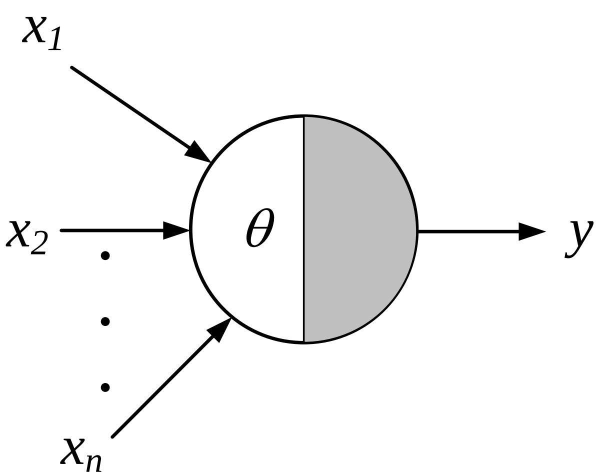


Porquê?

x1	*x2*	*g*	*y*
0	*0*		
1	*0*		
0	*1*		
1	*1*		

Unidade Neuronal Artificial

Neurónio de Mculloch e Pitts

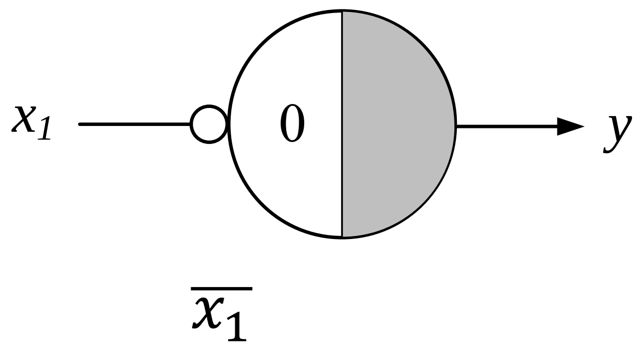


$$y = f\big(g(x)\big) = \begin{cases} 1 \Leftarrow g(x) \geq \theta \\ 0 \Leftarrow g(x) < \theta \end{cases}$$

$$g(x) = x_1 + x_2 + \cdots + x_n$$

Função NOT

✓ Com Entrada Inibidora:

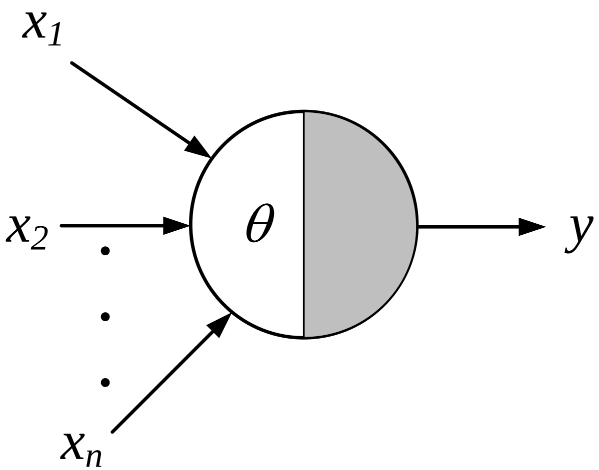


Porquê?

x1	*g*	*y*
0	*0*	*1*
1	-	*0*

Unidade Neuronal Artificial

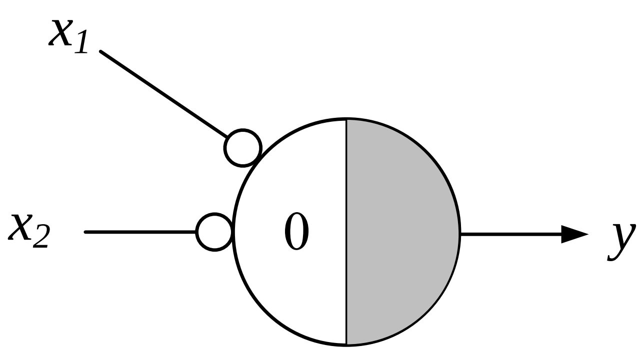
Neurónio de Mculloch e Pitts



$$y = f\big(g(x)\big) = \begin{cases} 1 \Leftarrow g(x) \geq \theta \\ 0 \Leftarrow g(x) < \theta \end{cases}$$

$$g(x) = x_1 + x_2 + \cdots + x_n$$

✓ Que função lógica implementa o seguinte neurónio?



x1	***x2***	***g***	***y***
0	*0*		
1	*0*		
0	*1*		
1	*1*		

Neurónio de Mculloch and Pitts

✓ Limitações:

- E se as entradas forem não Booleanas?
- É necessário de codificar o limite (*threshold*) para cada caso?
- E se as entradas tiverem relevâncias distintas?
- Pode classificar funções não linearmente separáveis (e.g. XOR)?

Necessitamos do Perceptrão proposto por Rosenblatt em 1958

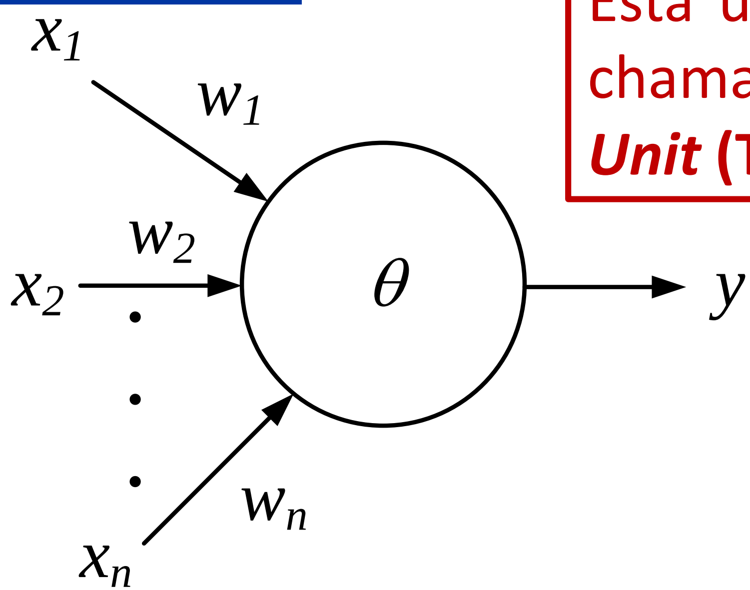
Perceptrão (*Perceptron*)

Perceptrão de Rosenblatt (TLU)

Esta unidade é frequentemente chamada de ***Threshold Logic Unit*** **(TLU).**

- **Entradas:**

$$x = (x_1, x_2, \cdots, x_n)$$

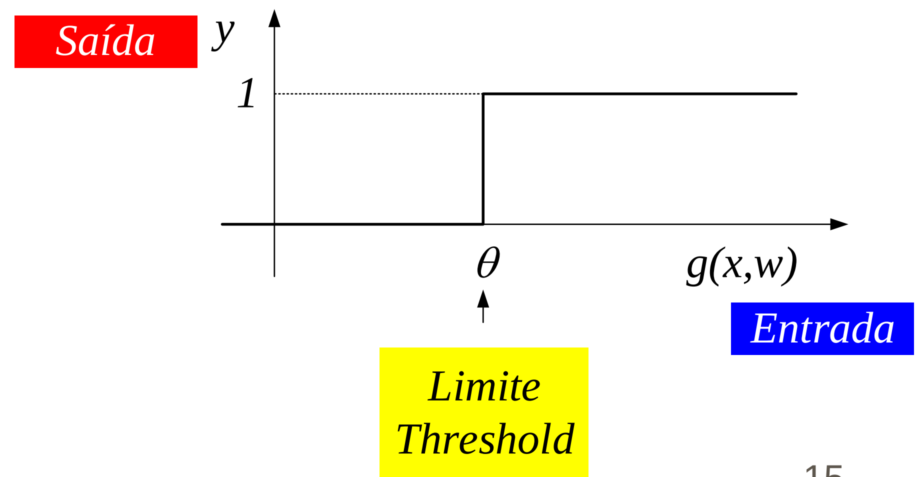


- **Fatores de peso:**

$$w = (w_1, w_2, \cdots, w_n)$$

$$g(x,w) = x \,.\, w = x_1\, w_1 + x_2 w_2 + \cdots + x_n w_n$$

$$y = f\big(g(x,w)\big) = \begin{cases} 1 \Leftarrow g(x,w) \geq \theta \\ 0 \Leftarrow g(x,w) < \theta \end{cases}$$



Perceptrão (*Perceptron*)

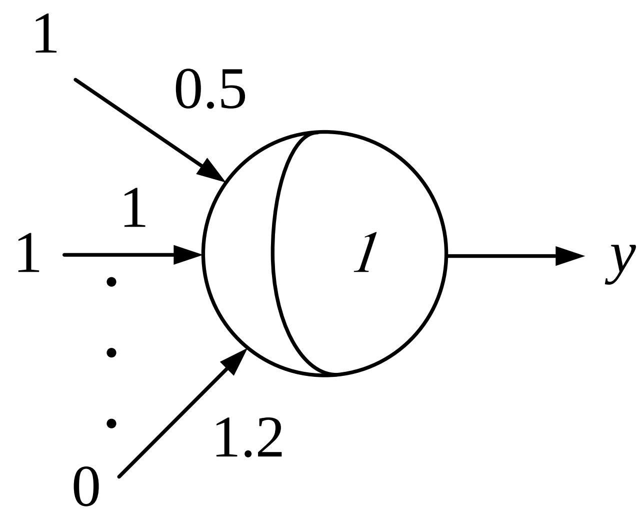
Exemplo:

- **Entradas:**

$x = (1,1,0\ )$

- **Fatores de peso:**

$w = (0.5,1,1.2)$



$$g(x,w) = x\,.w = x_1\ w_1 + x_2 w_2 + x_3 w_3$$

$$g = 1\ (0.5) + 1(1) + 0(1.2) = 1.5$$

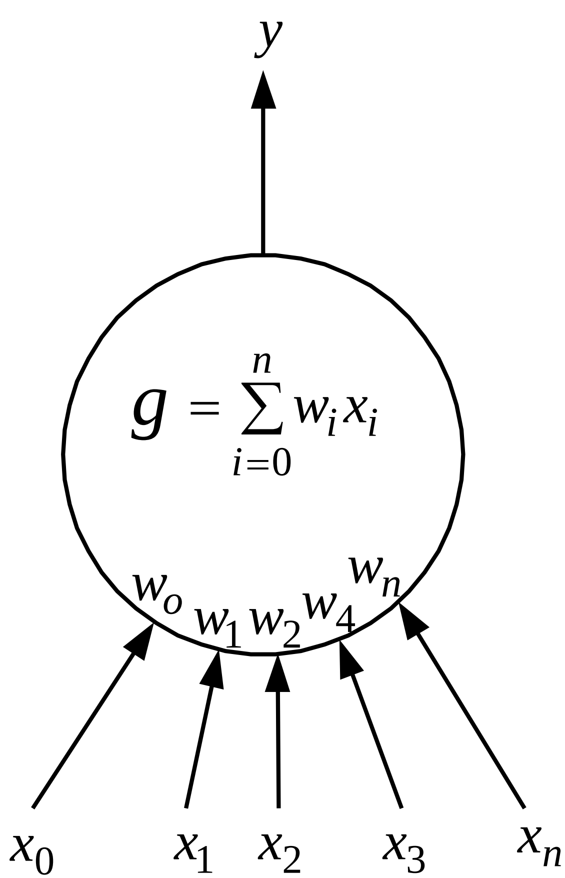
$$y = f\big(g(x,w)\big) = \begin{cases} 1 \Leftarrow g(x,w) \geq 1 \\ 0 \Leftarrow g(x,w) < 1 \end{cases}$$

- **Como neste caso g=1.5 ≥ 1 temos que y=1.**

Unidade Neuronal Artificial

A característica fundamental de qualquer rede neuronal é que é composta por muitas unidades interconectadas, em que cada uma executa uma soma ponderada das suas entradas.

- A unidade representada tem:
  - ***n+1* entradas, x_i, e uma única saída *y*.**
  - Cada entrada (ou arco) tem associada **um peso**, ***w_i***, que é um número real.



- Em cada unidade é feita uma soma ponderada:

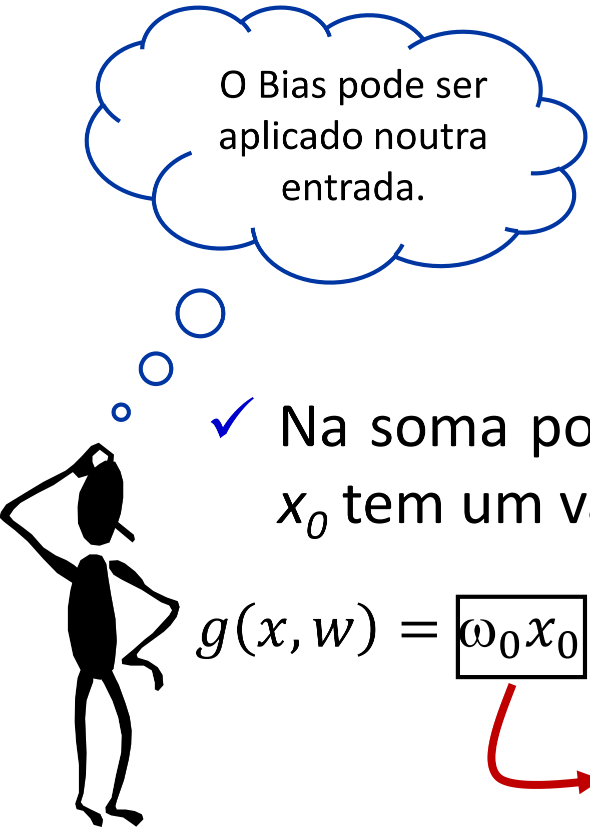
$$g(x,w) = \omega_0 x_0 + \omega_1 x_1 + \ldots + \omega_n x_n = \sum_{i=0}^{n} \omega_i x_i$$

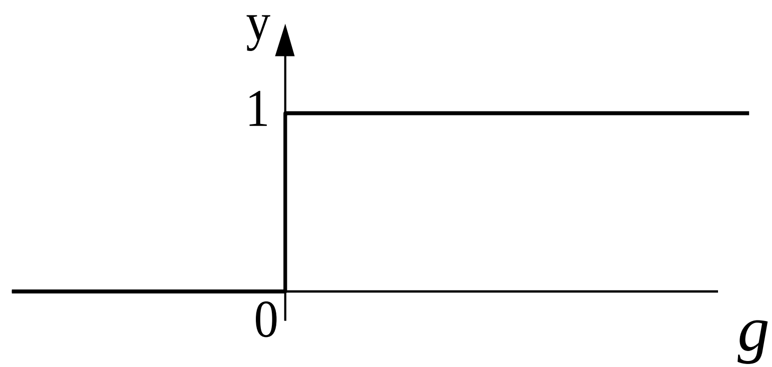
  - Esta soma ponderada vai ser o **argumento de uma função, f,** que vai determinar o valor da saída da unidade.

Unidade Neuronal Artificial

Função Ativação

- ✓ Uma função de saída típica (Heaviside) produz um 1 na saída quando a entrada é maior ou igual a zero e 0 no caso contrário:





- ✓ Na soma ponderada os termos x_1 a x_n são variáveis, mas o termo x_0 tem um valor fixo igual a *1:*

$$g(x,w) = \boxed{\omega_0 x_0} + \omega_1 x_1 + \ldots + \omega_n x_n = \sum_{i=0}^{n} \omega_i x_i$$

Termo de Offset ou Bias

- ✓ Este termo de bias está relacionado com a deslocação do *threshold* (θ) para a esquerda ou direita.

utad

Classificação de Padrões

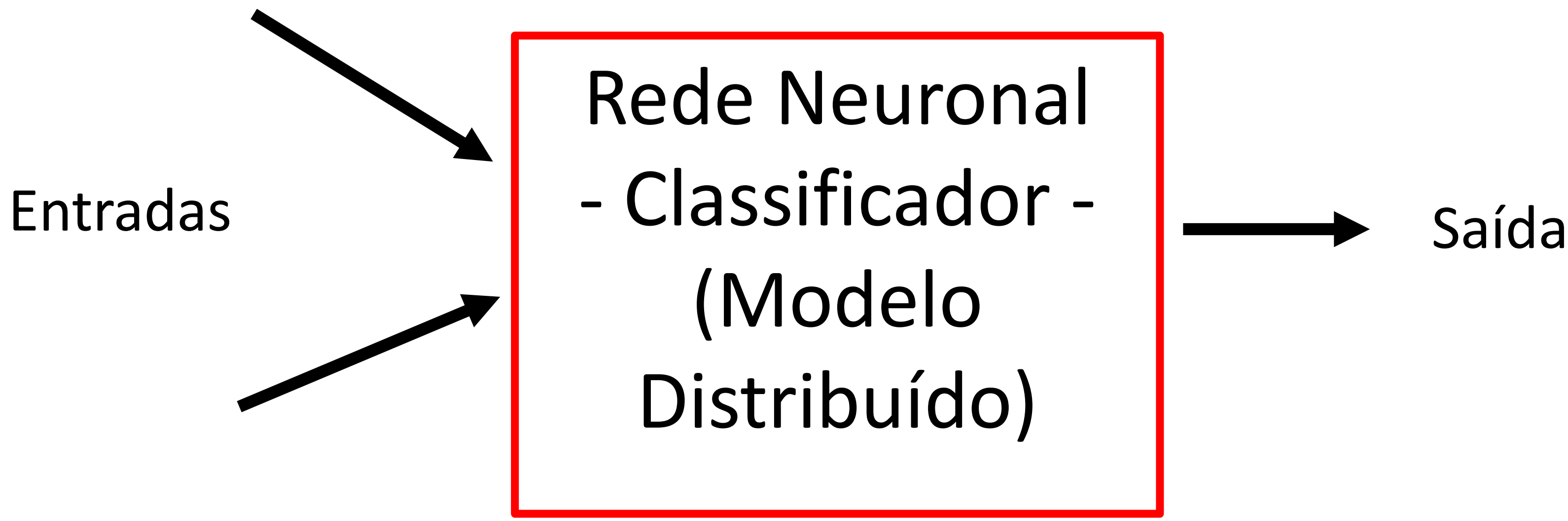
- Um **objeto pertence a uma certa classe** se tem **propriedades** que são **similares** a **outros objetos dessa classe**.

- De uma forma geral efetua-se uma comparação do objeto a classificar com um **<u>modelo ideal</u>** (para a banana e para a pera) do **objeto tipo que define a categoria**.

Classificação de Padrões

- Nos casos em que não existe um modelo para comparar os objetos a classificar **as redes neuronais são uma alternativa**.

- As características gerais de uma classe tem de ser **<u>inferidas</u>** de exemplos "vistos".



Aprendizagem Supervisionada

- ✓ Nos método de aprendizagem que vamos abordar com redes neuronais, os **dados utilizados no seu treino** incluem **a(s) entrada(s) e a saída(s) desejada(s) (** chamadas de **alvo ou *target*)** que são conhecidas à priori.

- ✓ Esta abordagem chama-se **Aprendizagem Supervisionada.**

- ✓ Como vamos ver melhor os dados têm de ser organizados em :
  - **conjunto de treino**
  - **conjunto de validação**
  - **conjunto de teste.**

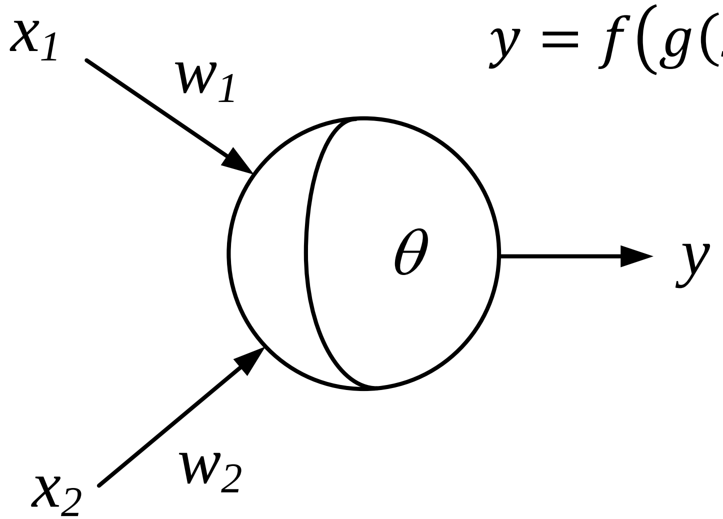
A rede neuronal deve possuir **Capacidade de Generalização**, ou seja, ser capaz de fornecer a resposta correta para dados não utilizados no seu treino.

Classificação de Padrões com uma Unidade Neuronal

✓ Considere-se a seguinte unidade neuronal artificial:

$$x = (x_1, x_2, \cdots, x_n)$$

$$w = (w_1, w_2, \cdots, w_n)$$



$$y = f\big(g(x,w)\big) = \begin{cases} 1 \Leftarrow g(x,w) \geq \theta \\ 0 \Leftarrow g(x,w) < \theta \end{cases}$$

$$x.w = \theta$$

$$x_1\, w_1 + x_2 w_2 = \theta$$

$$x_2 = \boxed{-\frac{w_1}{w_2}} x_1 \boxed{+\frac{\theta}{w_2}}$$

✓ Equação geral de uma reta: $y = \mathrm{mx} + \mathrm{c}$

$$\mathrm{m} = -\frac{w_1}{w_2} \qquad \mathrm{c} = \frac{\theta}{w_2}$$

utad

Classificação de Padrões com uma Unidade Neuronal

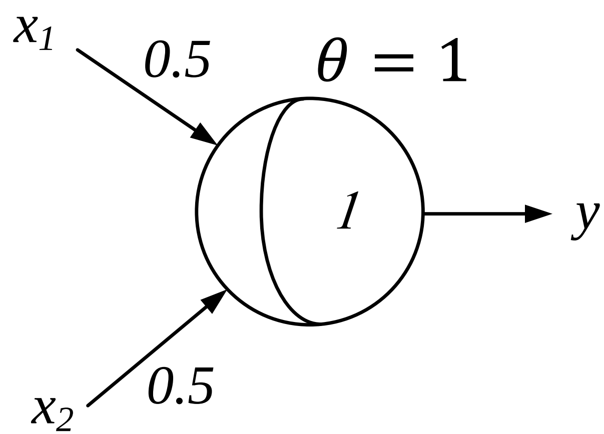
✓ Uma unidade neuronal pode classificar os padrões de uma função lógica **AND**?

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Classe 2: C1

Classe 1: C2

Espaço dos padrões



$$m = -\frac{w_1}{w_2} = -\frac{0.5}{0.5} = -1$$

$$c = \frac{\theta}{w_2} = \frac{1}{0.5} = 2$$

✓ Resultado da classificação:

x_1	x_2	$x.w$	y
0	0	0	0
0	1	0.5	0
1	0	0.5	0
1	1	1	1

Classificação de Padrões com uma Unidade Neuronal

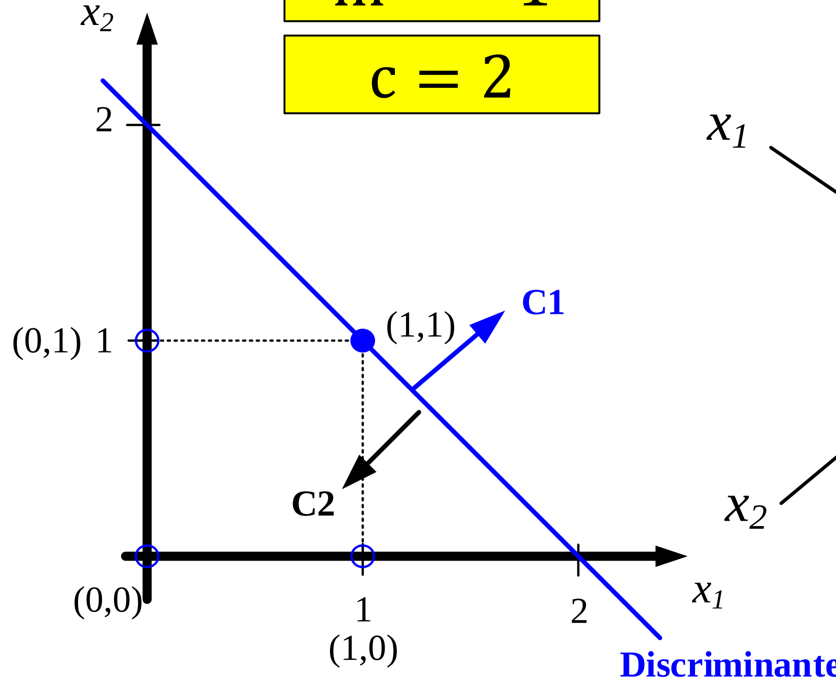
x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Classe 2: C1

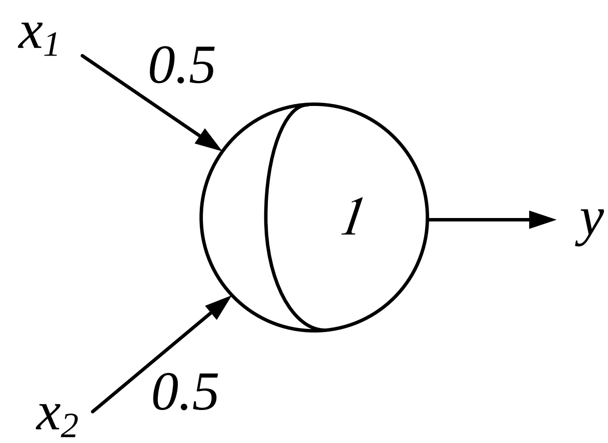
Classe 1: C2

$m = -1$

$c = 2$



✓ Classificação binária.

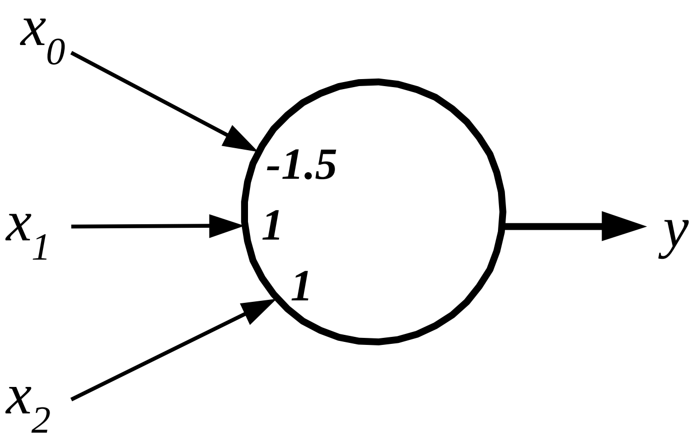


✓ Espaço dos padrões bidimensional⇒ discriminante é uma reta

✓ Espaço dos padrões n-bidimensional⇒ discriminante é uma superfície com dimensões (n-1): **Hiperplano**.

Classificação de Padrões com uma Unidade Neuronal

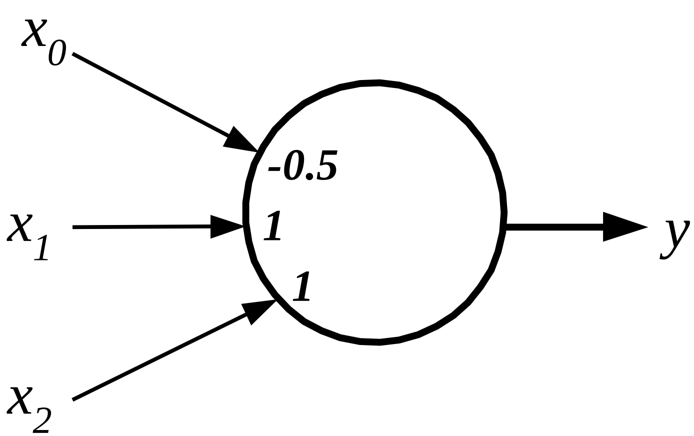
- A figura seguinte mostra: uma unidade com duas entradas x_1 e x_2 capaz de executar a função ***and*** e a tabela com os cálculos



x_0	x_1	x_2	g= $w_0x_0 + w_1x_1 + w_3x_3$	y
1	0	0	**-1.5x1 + 0x1 + 1x0=-1.5**	0
1	0	1	**-1.5x1 + 0x1 + 1x1=-0.5**	0
1	1	0	**-1.5x1 + 1x1 + 0x1=-0.5**	0
1	1	1	**-1.5x1 + 1x1 + 1x1=+0.5**	1

Classificação de Padrões com uma Unidade Neuronal

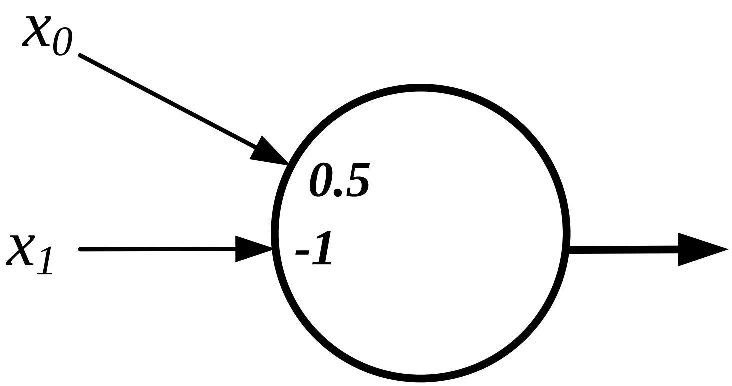
- ✓ A figura seguinte mostra: uma unidade com duas entradas x_1 e x_2 capaz de executar a função ***or*** e a tabela (confirme os cálculos).



x_0	x_1	x_2	g	y
1	0	0	**-0.5**	0
1	0	1	**+0.5**	1
1	1	0	**+0.5**	1
1	1	1	**+1.5**	1

Classificação de Padrões com uma Unidade Neuronal

✓ A figura seguinte mostra uma unidade com uma entrada x_1 capaz de executar a função de negação e a tabela.



x_0	x_1	g	y
1	0	**+0.5**	1
1	1	**-0.5**	0

A partir destas três funções (*and*, *or* e *not*) é possível implementar qualquer função lógica.

Isto significa que computadores convencionais *podiam* ser construídos usando estas unidades neuronais artificiais em vez dos usuais circuitos lógicos com transístores.

Classificação de Padrões com uma Unidade Neuronal

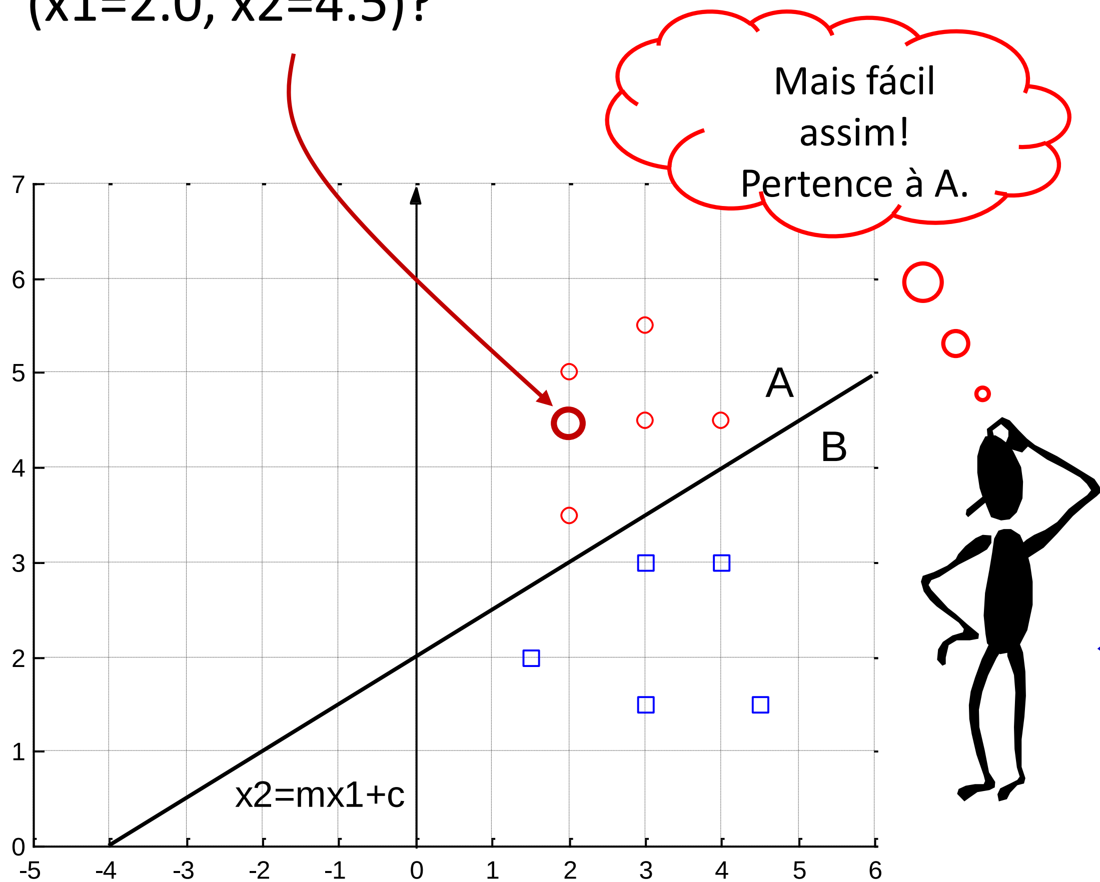
- ✓ Considere-se o seguinte exemplo [1]:
- ✓ A que classe é que pertence a amostra (x1=2.0, x2=4.5)?

Amostra	x_1	x_2	*Classificação*
1	2.0	3.5	A
2	3.0	1.5	B
3	4.5	1.5	B
4	1.5	2.0	B
5	3.0	4.5	A
6	2.0	5.0	A
7	4.0	3.0	B
8	3.0	3.0	B
9	3.0	5.5	A
10	4.0	4.5	A

[1] Johnson J. and Picton P, Concepts on Artificial Intelligence, Designing Intelligent Machines, Part II, Butterworth-Heinemann

Classificação de Padrões com uma Unidade Neuronal

- ✓ A que classe é que pertence a amostra (x1=2.0, x2=4.5)?



Amostra	x_1	x_2	*Classificação*
1	2.0	3.5	A
2	3.0	1.5	B
3	4.5	1.5	B
4	1.5	2.0	B
5	3.0	4.5	A
6	2.0	5.0	A
7	4.0	3.0	B
8	3.0	3.0	B
9	3.0	5.5	A
10	4.0	4.5	A

- ✓ Para determinar se um ponto pertence à classe A ou B só temos de determinar de que lado da reta está.

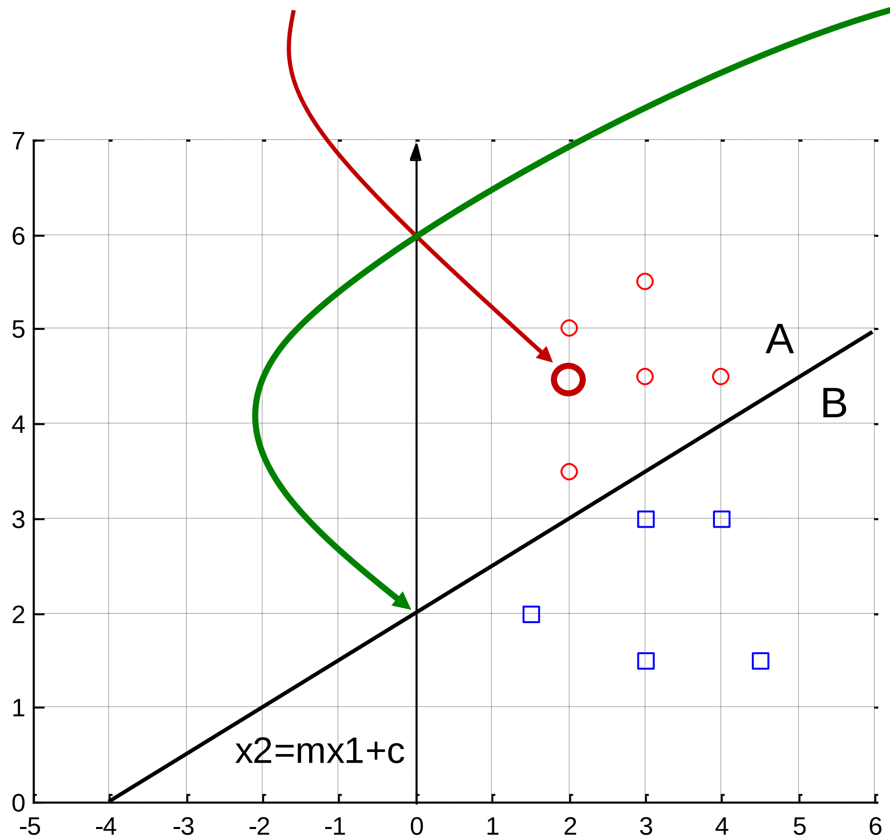
Estas classes são linearmente separáveis!

Classificação de Padrões com uma Unidade Neuronal

Como se pode testar se um ponto está à esquerda ou à direita da linha?

$$\boxed{y = mx + c}$$

$$x_2 = mx_1 + c$$



$$x_1 = 0 \Rightarrow c = x_2 = 2$$

$$x_2 = 0 \Rightarrow 0 = -4m + 2$$

$$\therefore m = 0.5$$

- ✓ Assim a equação da reta é: $\boxed{x_2 = 0.5x_1 + 2}$

ou: $x_2 - 0.5x_1 - 2 = 0$

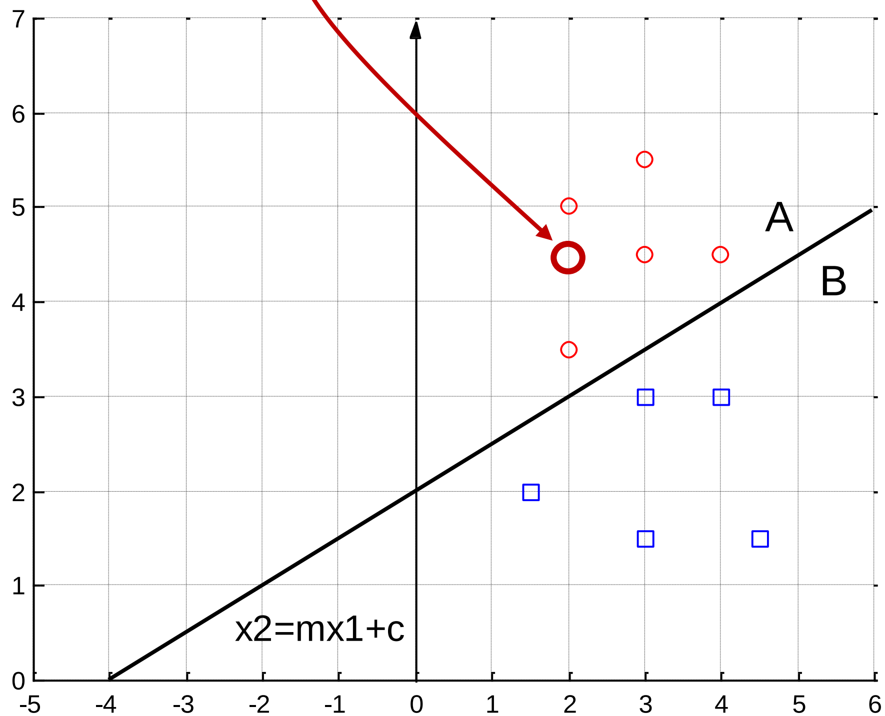
- ✓ Substituindo x_1 e x_2 nesta equação se o resultado for:
  - = 0, significa que o ponto está na reta;
  - > 0, significa que o ponto pertence a A;
  - < 0, significa que o ponto pertence a B.

Classificação de Padrões com uma Unidade Neuronal

- ✓ Vamos confirmar que (x_1=2.0, x_2=4.5) pertence a A:

$$x_2 - 0.5x_1 - 2 = 0$$

$$x_2 - 0.5x_1 - 2.0 = 4.5 - 0.5(2.0) - 2.0 = 1.5$$

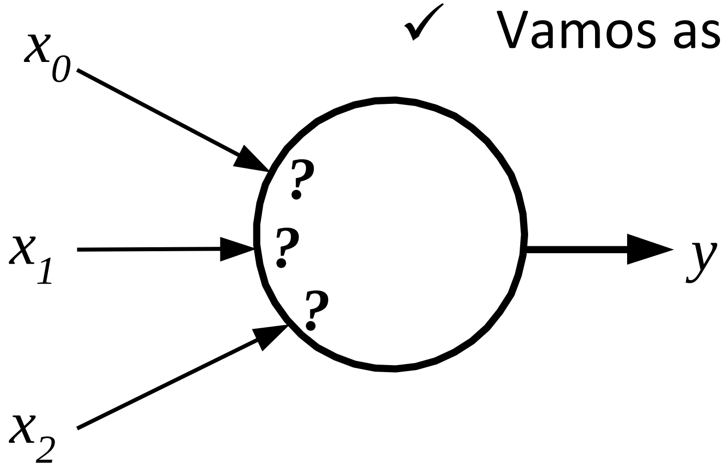


✓ como o resultado é > 0, significa que o ponto pertence à classe A;

Pode-se utilizar uma unidade neuronal para o mesmo fim?

Classificação de Padrões com uma Unidade Neuronal

Unidade com duas entradas:



- ✓ Vamos assumir a seguinte classificação. Se a saída, y, for:

$$y = 1 \Leftarrow g \geq 0 \quad Objecto\ Classe\ A$$
$$y = 0 \Leftarrow g < 0 \quad Objecto\ Classe\ B$$

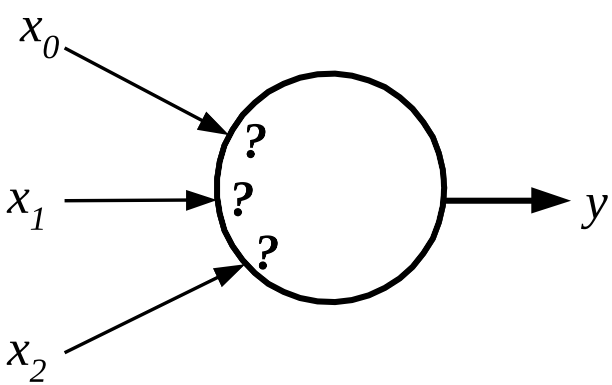
- ✓ A título de exercício vamos determinar **quais os pesos necessários** para obter a classificação destas duas classes com esta unidade neuronal.

Classificação de Padrões com uma Unidade Neuronal

Quais são os pesos?

✓ A partir da soma:

$$g = \sum_{i=0}^{2} \omega_i x_i = \omega_0 x_0 + \omega_1 x_1 + \omega_2 x_2$$



$$\boxed{\begin{array}{ll} y = 1 \Leftarrow \omega_0 x_0 + \omega x_1 + \omega_2 x_2 \geq 0 & \textit{Objecto ClasseA} \\ y = 0 \Leftarrow \omega_0 x_0 + \omega_1 x_1 + \omega_2 x_2 < 0 & \textit{Objecto ClasseB} \end{array}}$$

- A linha divisora entre correspondente à divisão entre as duas classes é representada por:

$$w_0 x_0 + w_1 x_1 + w_2 x_2 = 0$$

$$w_2 x_2 = -w_1 x_1 - w_0 x_0 \Leftrightarrow x_2 = \boxed{-\frac{w_1}{w_2}} x_1 \boxed{-\frac{w_0}{w_2}} x_0$$

$$x_2 = m x_1 + c$$

$$\boxed{m = -\frac{w_1}{w_2}} \quad \boxed{c = -\frac{w_0}{w_2}}$$

$$m = 0.5 \qquad c = 2$$

Classificação de Padrões com uma Unidade Neuronal

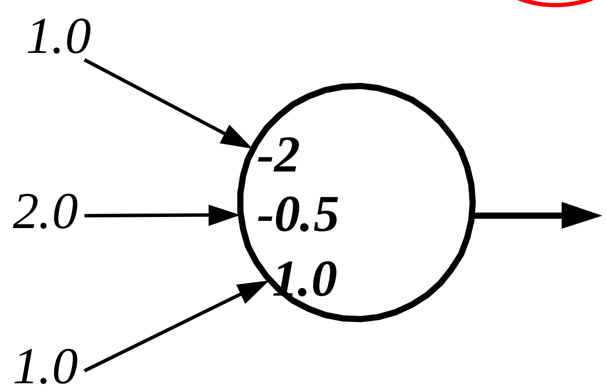
Quais são os pesos?

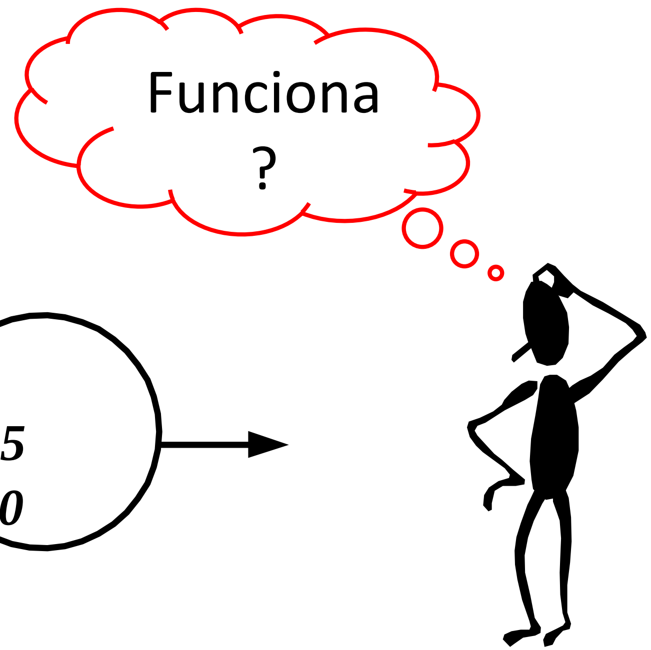
- Já tínhamos concluído que neste caso:

$$m = 0.5 \qquad c = 2$$

$$\boxed{m = -\frac{w_1}{w_2}} \qquad \boxed{c = -\frac{w_0}{w_2}}$$

$$w_2 = 1.0 \Rightarrow \begin{cases} w_0 = -2.0 \\ w_1 = -0.5 \end{cases}$$





✓ Vamos testar com (x_1=2.0, x_2=4.5) :

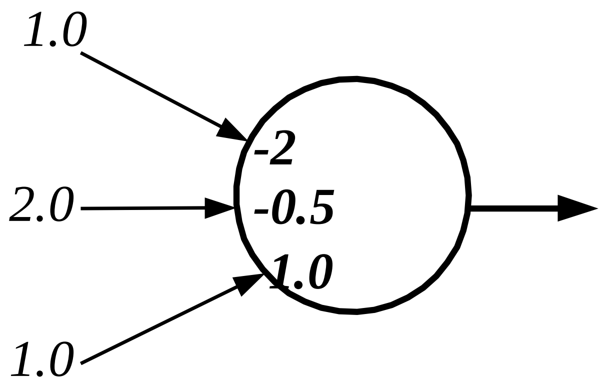
$$g = w_0x_0 + w_1x_1 + w_2x_2 = -2(1) + (-0.5)2 + 4.5(1) = 1.5$$

$$g = 1.5 > 0 \Rightarrow y = 1 \Rightarrow Categoria\ A$$

Classificação de Padrões com uma Unidade Neuronal

Generalização

O ponto x_1=2.0, x_2=4.5 não estava no conjunto de amostras iniciais



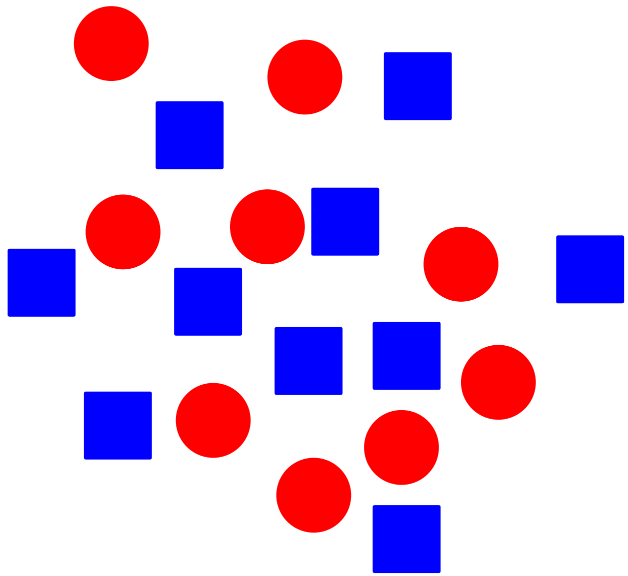
Amostra	x_1	x_2	*Classificação*
1	2.0	3.5	A
2	3.0	1.5	B
3	4.5	1.5	B
4	1.5	2.0	B
5	3.0	4.5	A
6	2.0	5.0	A
7	4.0	3.0	B
8	3.0	3.0	B
9	3.0	5.5	A
10	4.0	4.5	A

Uma unidade com este pesos consegue classificar amostras que não estavam no conjunto inicial. A esta capacidade de classificar exemplos não "vistos" antes chama-se generalização.

Classificação de Padrões com uma Unidade Neuronal

Conclui-se que uma unidade é capaz de efetuar a classificação se os dados forem linearmente separáveis:

E para dados não separáveis linearmente?



- Como é que as unidades neuronais podem ser utilizadas para classificar dados não linearmente separáveis?

- **Antes de abordar esta questão vamos ver como se processa a aprendizagem num perceptrão.**