

# Inteligencia Artificial

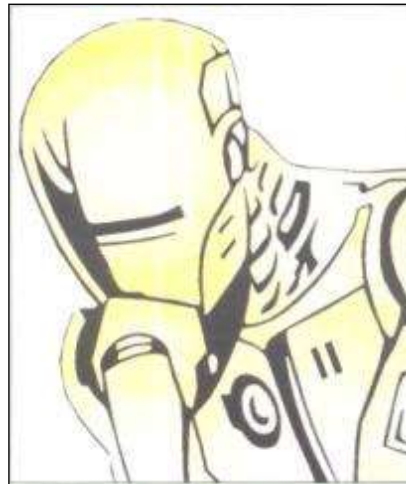
Oscar Bedoya

`oscar.bedoya@correounivalle.edu.co`

\* Machine learning

\* Redes neuronales

- Arquitectura
- Funciones de activación
- scikit learn



El viento del otoño

la margarita

tristemente resiste

# Machine learning

# Machine learning

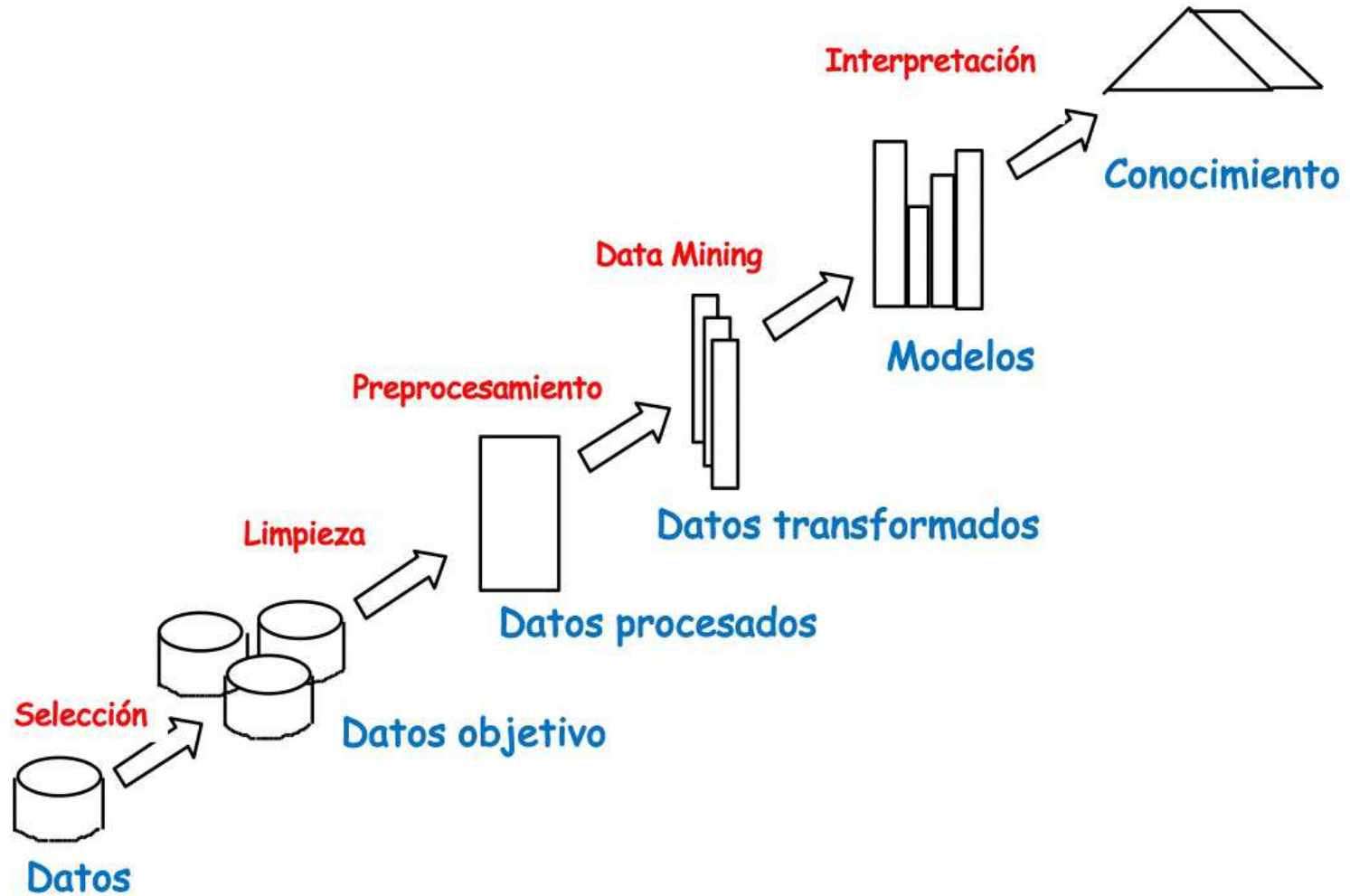
---

## Definición

- Es una disciplina de la inteligencia artificial que a través de algoritmos le da a los computadores la capacidad de identificar patrones en datos masivos y elaborar predicciones
- Es el proceso de descubrir patrones interesantes, potencialmente útiles y no obvios

# Machine learning

---



# Machine learning

---

## Técnicas principales

- Clustering
- Clasificación
- Reglas de asociación
- Patrones de secuencia

# Machine learning

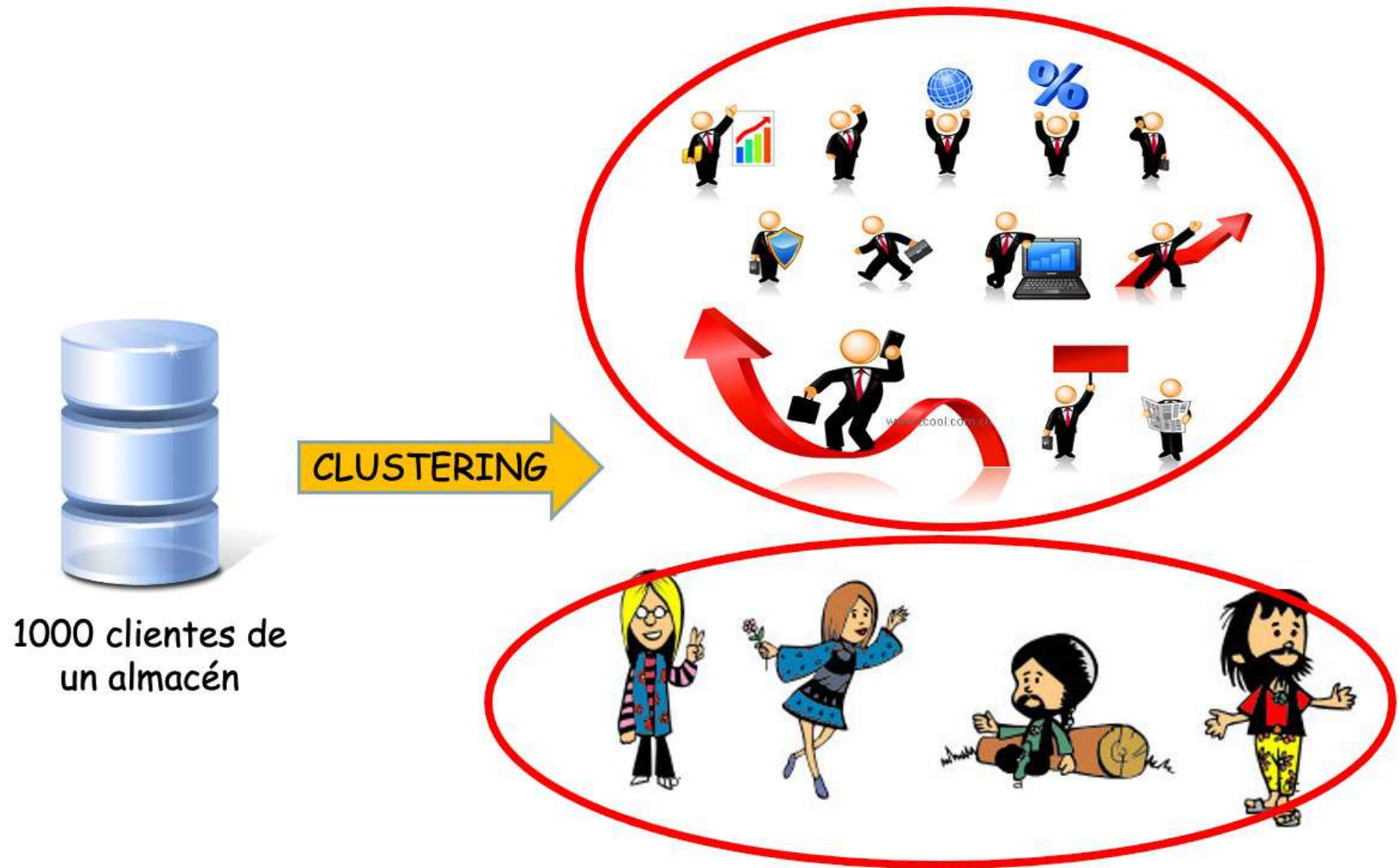
---

## Clustering

Divide un conjunto de datos en grupos de tal forma que los elementos en un grupo sean similares y los elementos en grupos diferentes tengan la menor similitud posible



# Machine learning



# Machine learning

---

## Así funcionan los algoritmos con los que Netflix quiere liderar el vídeo bajo demanda

Netflix no se preocupa por la geografía, el género y la edad del usuario. Lo que hace es agrupar sus títulos en **lo que llaman clusters**, que se basan en los gustos de los usuarios según publicaron en [Wired](#). Estos grupos pueden contener a millones de usuarios, ayudando a mostrar a cada uno los 40 ó 50 títulos que aparecen en la pantalla principal.

La plataforma asigna a cada suscriptor a de **entre tres a cinco** clusters, que se determinan por el grado en el que cada uno se ajusta a sus gustos. De esta forma la información que recibe puede parecerse mucho más a lo que quiere ver.

Netflix como plataforma sabe que **sólo tiene 90 segundos** para convencer al usuario, según Business Insider, antes de que abandone el servicio y pase a otra cosa. La personalización es clave para conseguir que vuelvan. Pero tampoco se puede personalizar en exceso, con lo que el servicio debe introducir variantes.

# Machine learning

---

## Técnicas principales

- Clustering
- Clasificación
- Reglas de asociación
- Patrones de secuencia

# Machine learning

---

## Clasificación

Técnica que permite obtener modelos predictivos dado un conjunto de entrenamiento que se caracteriza por tener unas variables independientes y una variable dependiente o etiqueta de clase

# Machine learning

---

age	income	student	credit_rating	buys computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

# Machine learning

---

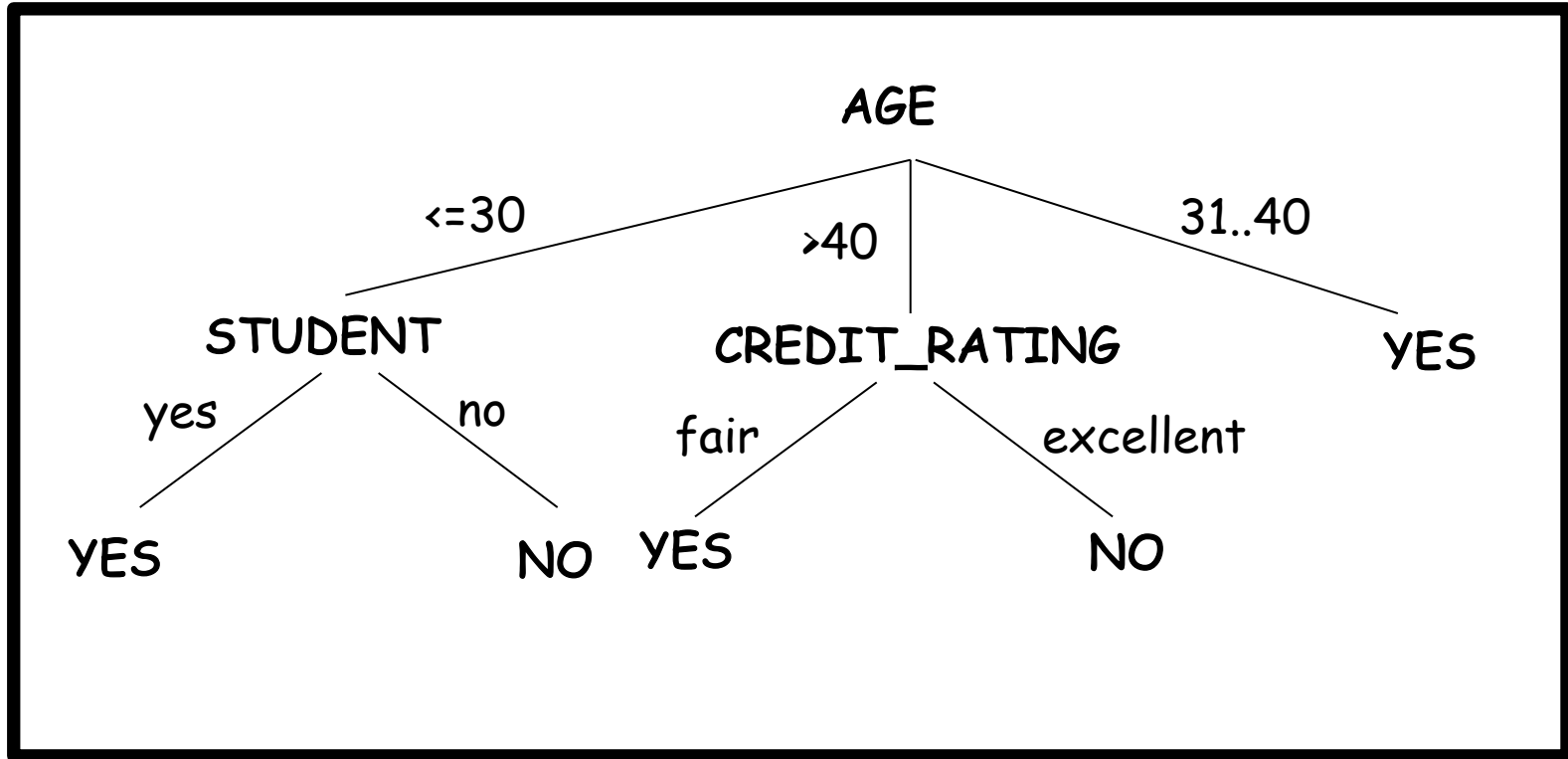
*Variables independientes*

*Variable  
dependiente*

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

# Machine learning

---



# Machine learning

---

age	income	student	credit_rating	buys computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

25

medium

yes

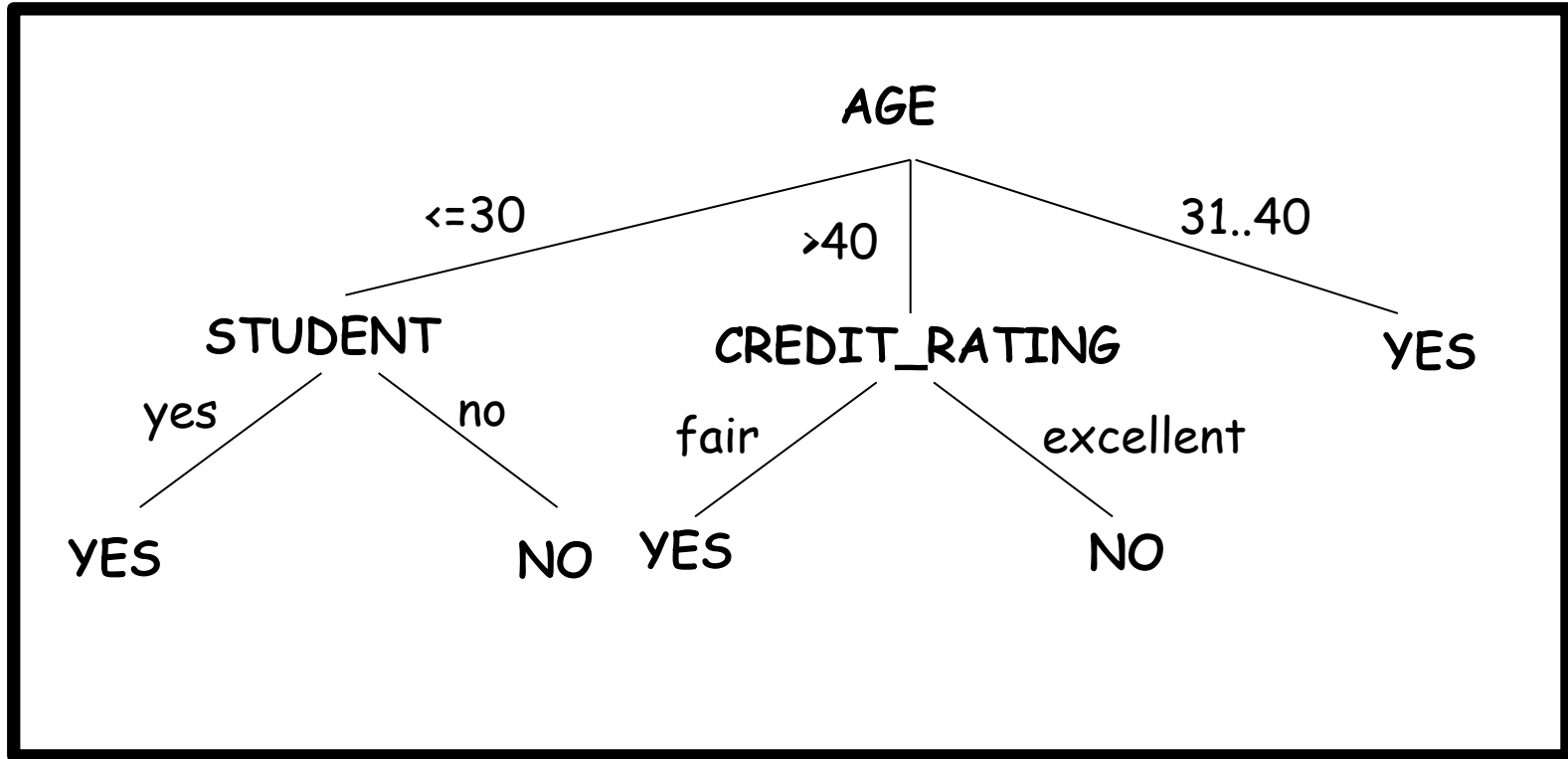
fair

?



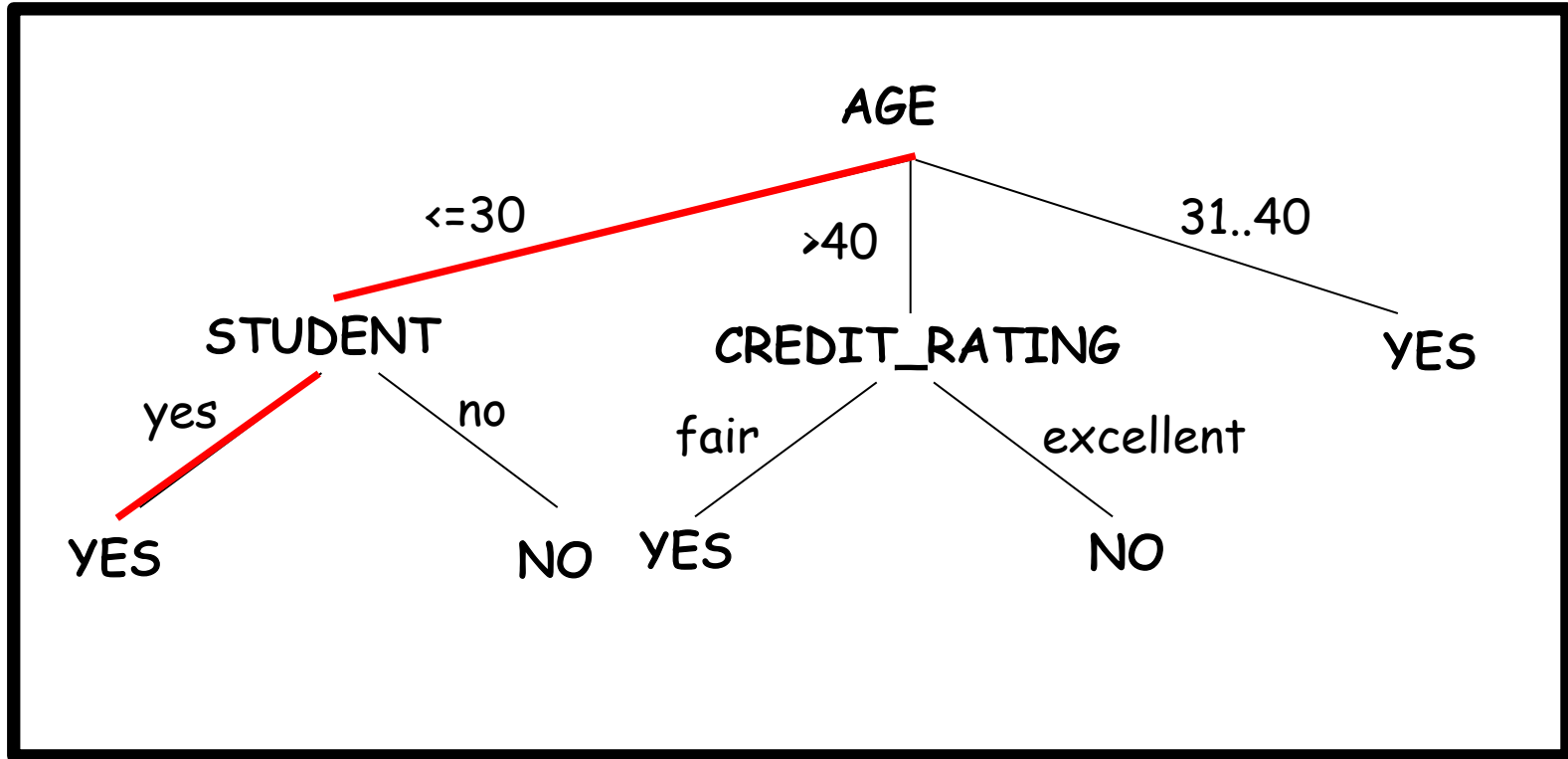
# Machine learning

---



AGE	INCOME	STUDENT	CREDIT_RATING	BUYS_COMPUTER
25	medium	yes	fair	?

# Machine learning



AGE	INCOME	STUDENT	CREDIT_RATING	BUYS_COMPUTER
25	medium	yes	fair	<b>YES</b>

# Machine learning

---

?	?	?	?	GanaIA

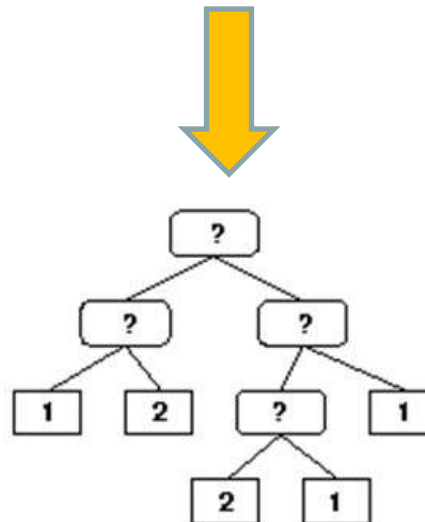
# Machine learning

---

Edad	Trabaja	EstaViendoFADA	Programa en Python	GanaIA
23	SI	SI	NO	NO
21	NO	NO	SI	SI
22	NO	SI	SI	NO
21	NO	SI	NO	NO
20	SI	NO	NO	SI

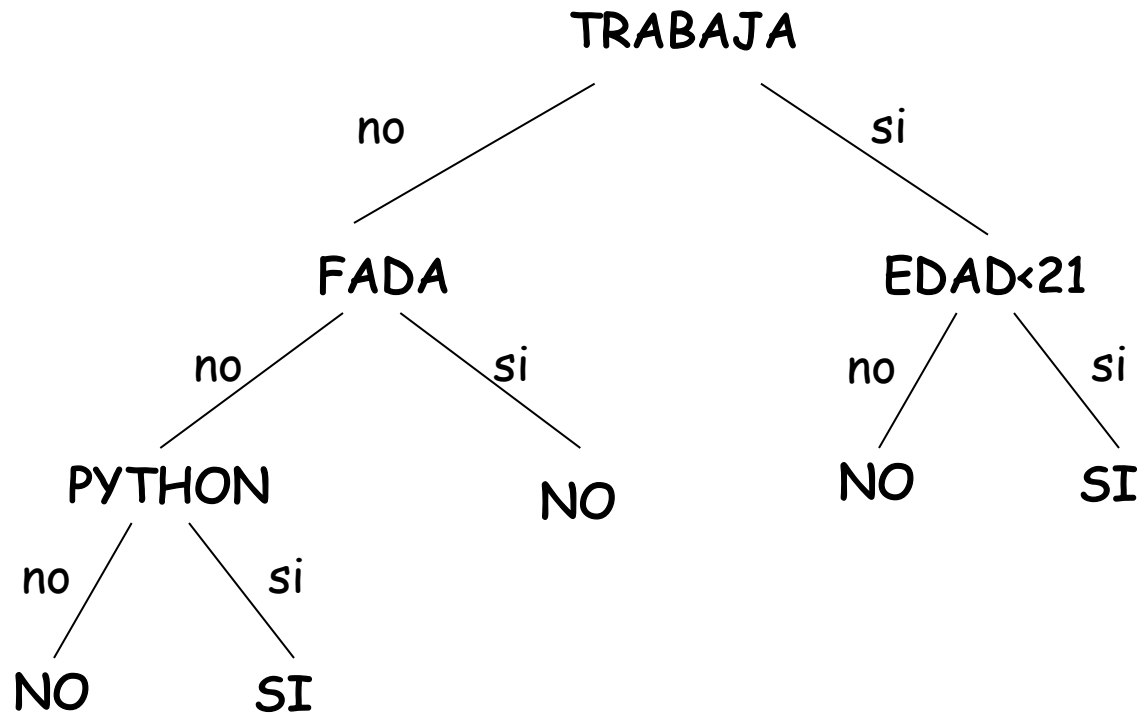
# Machine learning

Edad	Trabaja	EstaViendoFADA	Programa en Python	GanaIA
23	SI	SI	NO	NO
21	NO	NO	SI	SI
22	NO	SI	SI	NO
21	NO	SI	NO	NO
20	SI	NO	NO	SI



# Machine learning

---



# Machine learning

---

## Reglas de asociación

Dado un conjunto de registros, encontrar reglas que predecirán la ocurrencia de un ítem, basándose en las ocurrencias de los otros ítems en el registro

# Machine learning

---

## Market-Basket transaction

<i>TID</i>	<i>Items</i>
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke




# Machine learning

## Market-Basket transactions

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Example:



TID	Bread	Milk	Diaper	Beer	Eggs	Coke
1	1	1	0	0	0	0
2	1	0	1	1	1	0
3	0	1	1	1	0	1
4	1	1	1	1	0	0
5	1	1	1	0	0	1

# Machine learning

---

## Técnicas principales

- Clustering
- Clasificación
- Reglas de asociación
- Patrones de secuencia

# Machine learning

---

## Patrones de secuencia

Dado un conjunto de objetos, cada uno asociado con su propia línea de tiempo (eventos), encuentra reglas que predicen dependencias fuertes entre diferentes eventos

*(Intro to visual C) → (Perl for dummies) → (Ruby on rails for dummies)*  
*(C++Primer)                      (TCL TK)*

# Redes neuronales

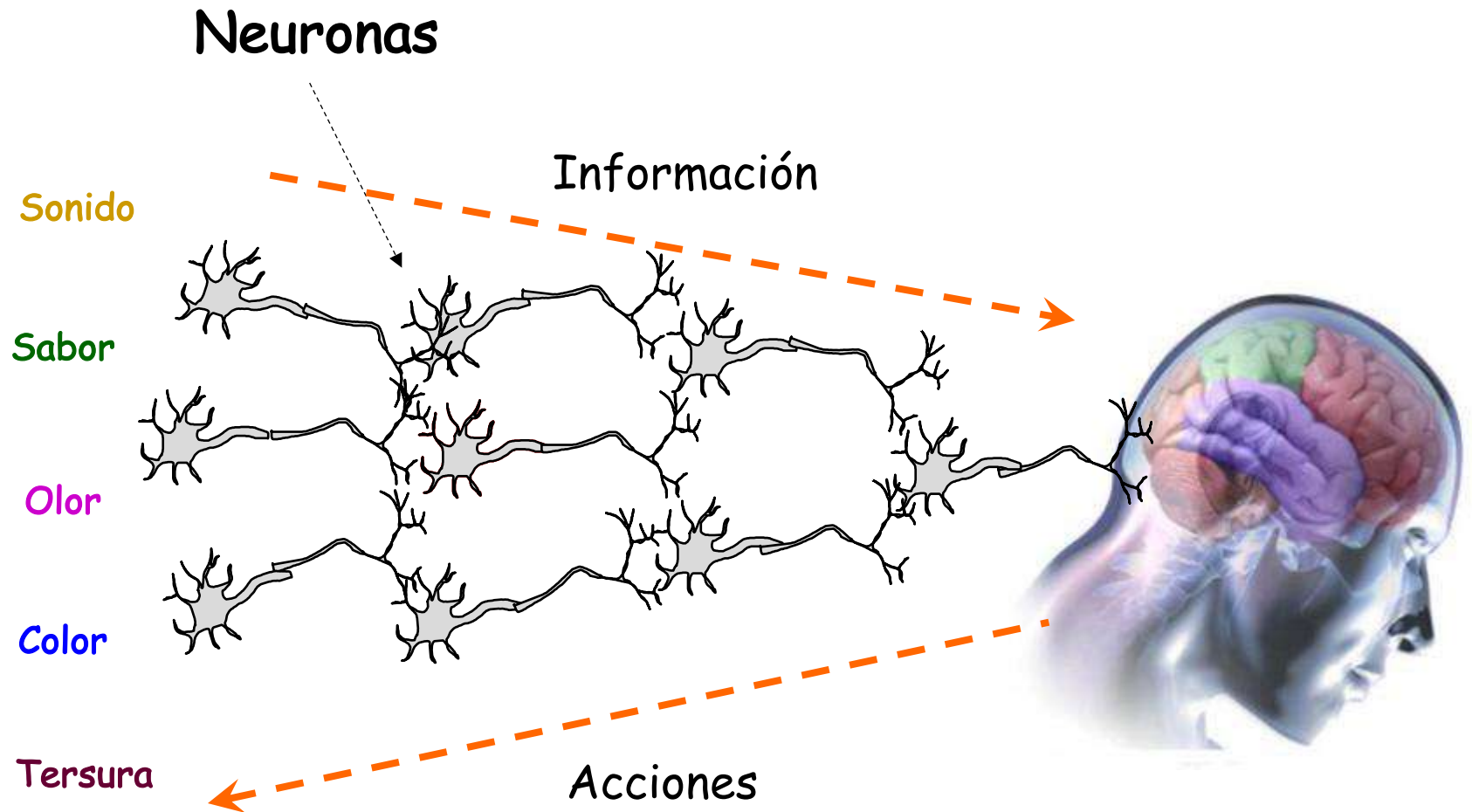
# Redes neuronales

---

## Definición

Las redes neuronales artificiales son una técnica inspirada en el sistema nervioso humano que permite obtener modelos predictivos

# Redes neuronales



# Redes neuronales

---



# Redes neuronales

---





# Redes neuronales

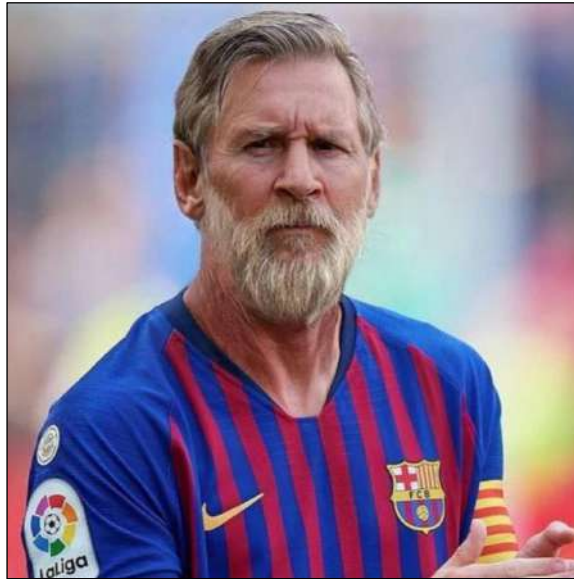
---

## Características

- **Aprendizaje adaptativo:** aprender mediante un proceso de entrenamiento
- **Capacidad de generalización:** predecir correctamente casos de prueba que son muy diferentes a los de entrenamiento

# Redes neuronales

---



# Redes neuronales

---

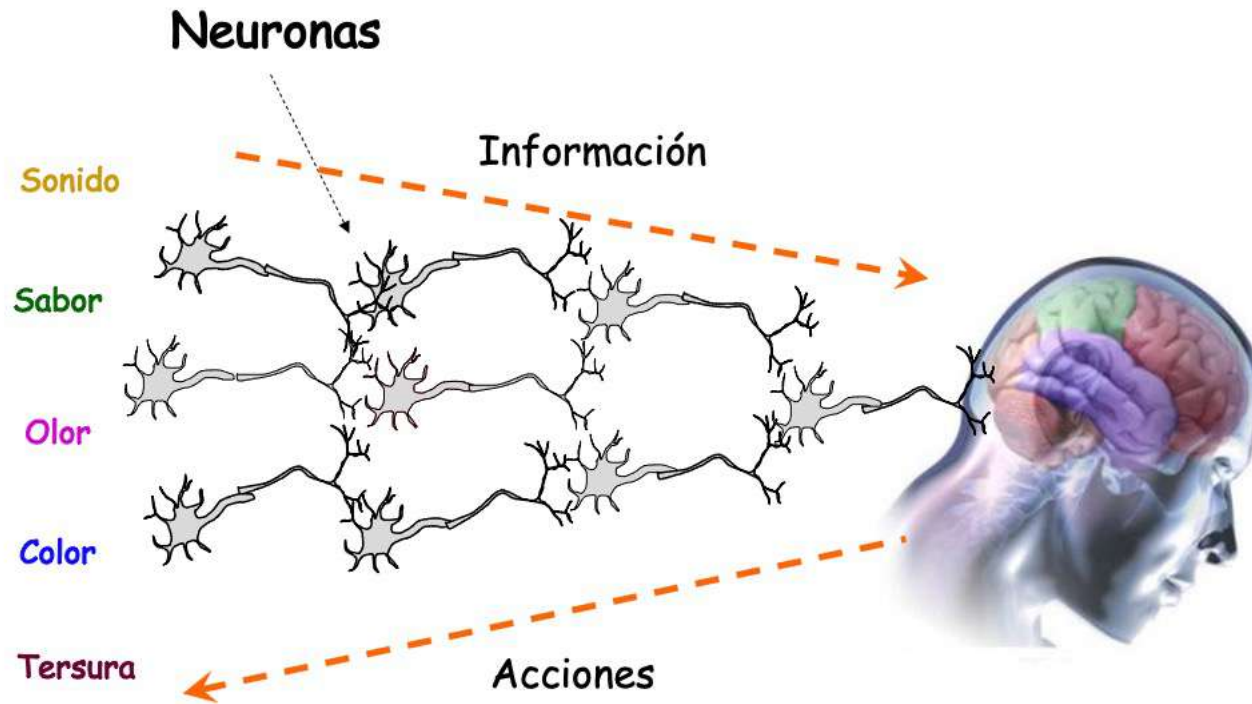
## McCulloch-Pitts

- Sus experimentos permitieron simular en un programa la forma como los seres humanos realizamos el proceso de aprendizaje



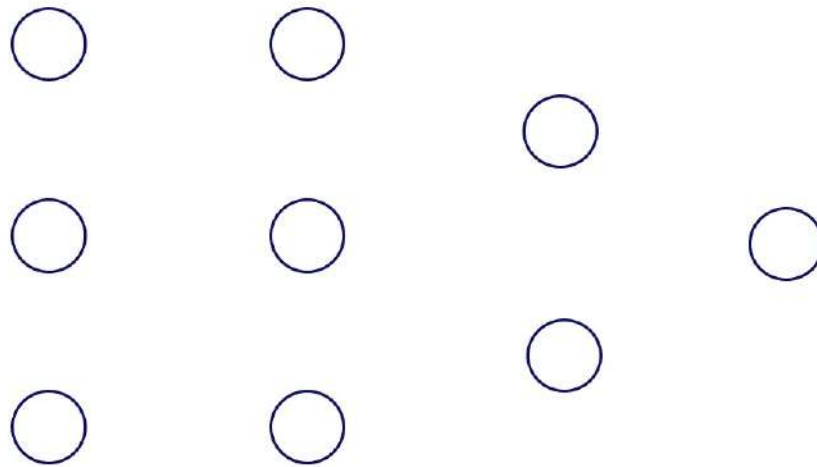
# Redes neuronales

## Arquitectura



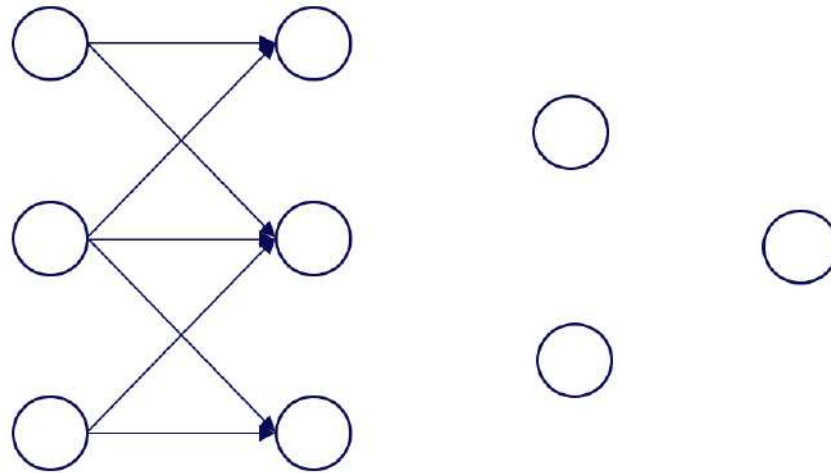
# Redes neuronales

---



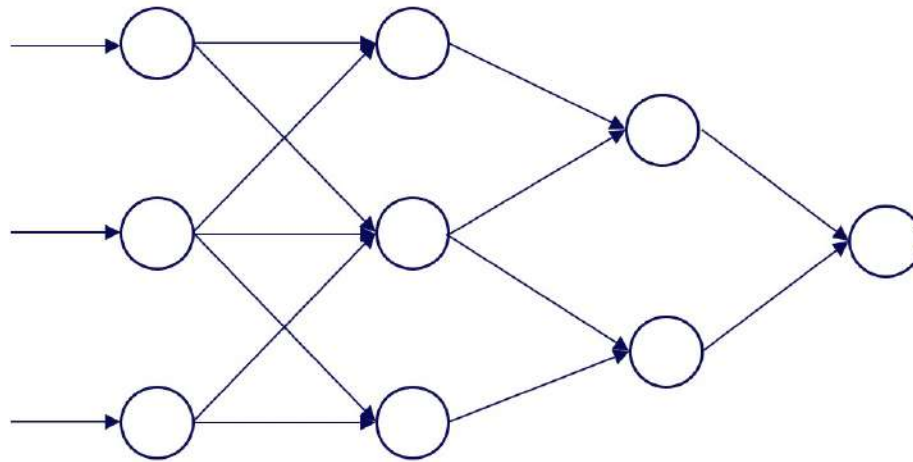
# Redes neuronales

---



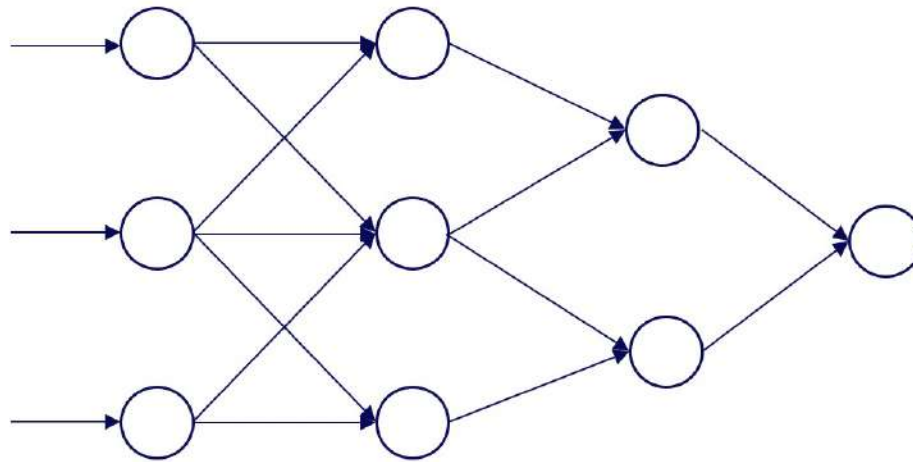
# Redes neuronales

---



# Redes neuronales

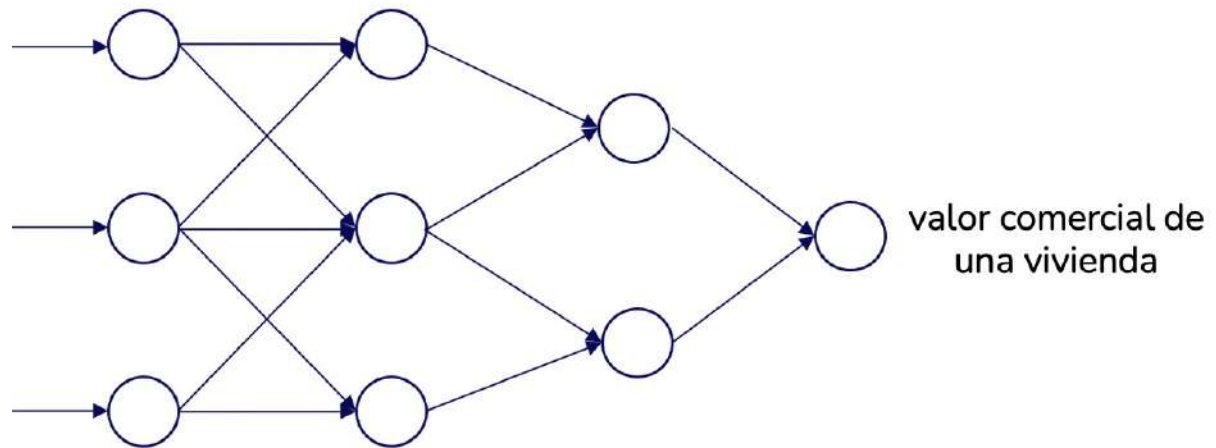
---





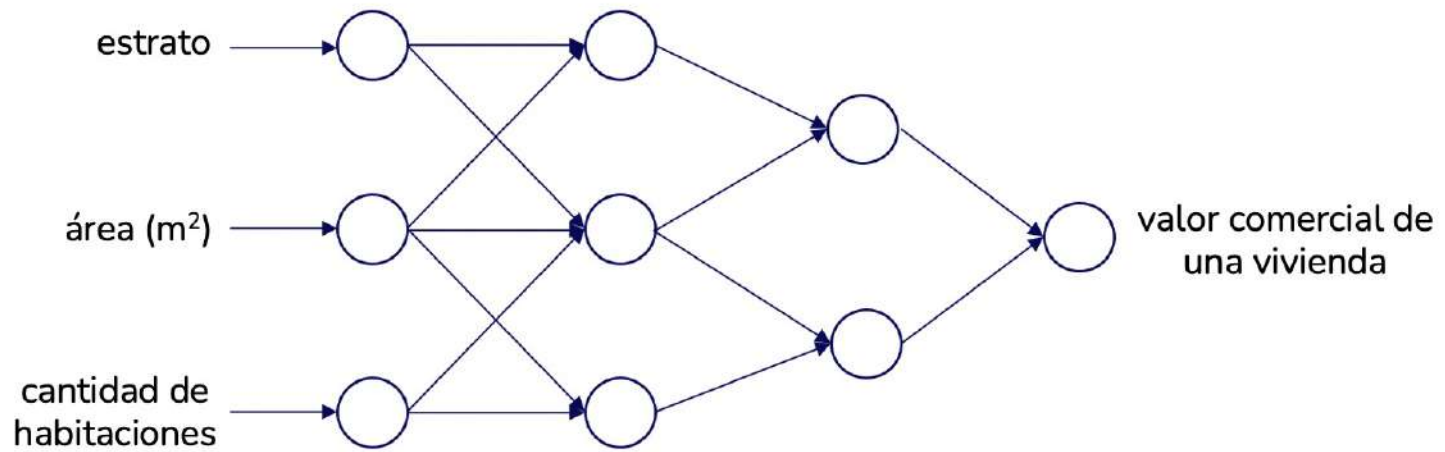
# Redes neuronales

---



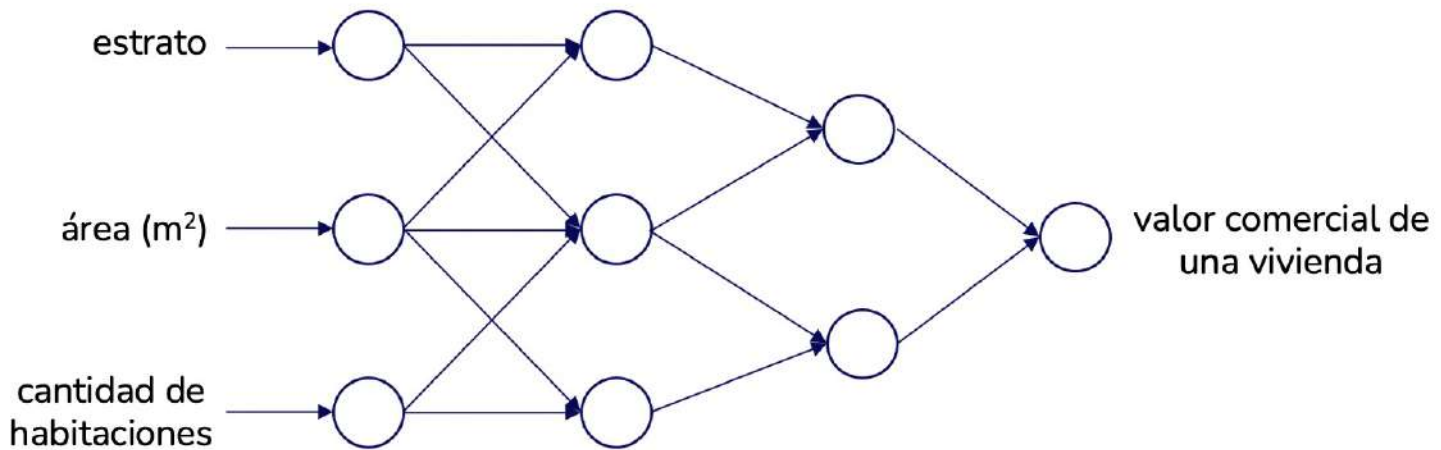
# Redes neuronales

---



# Redes neuronales

---

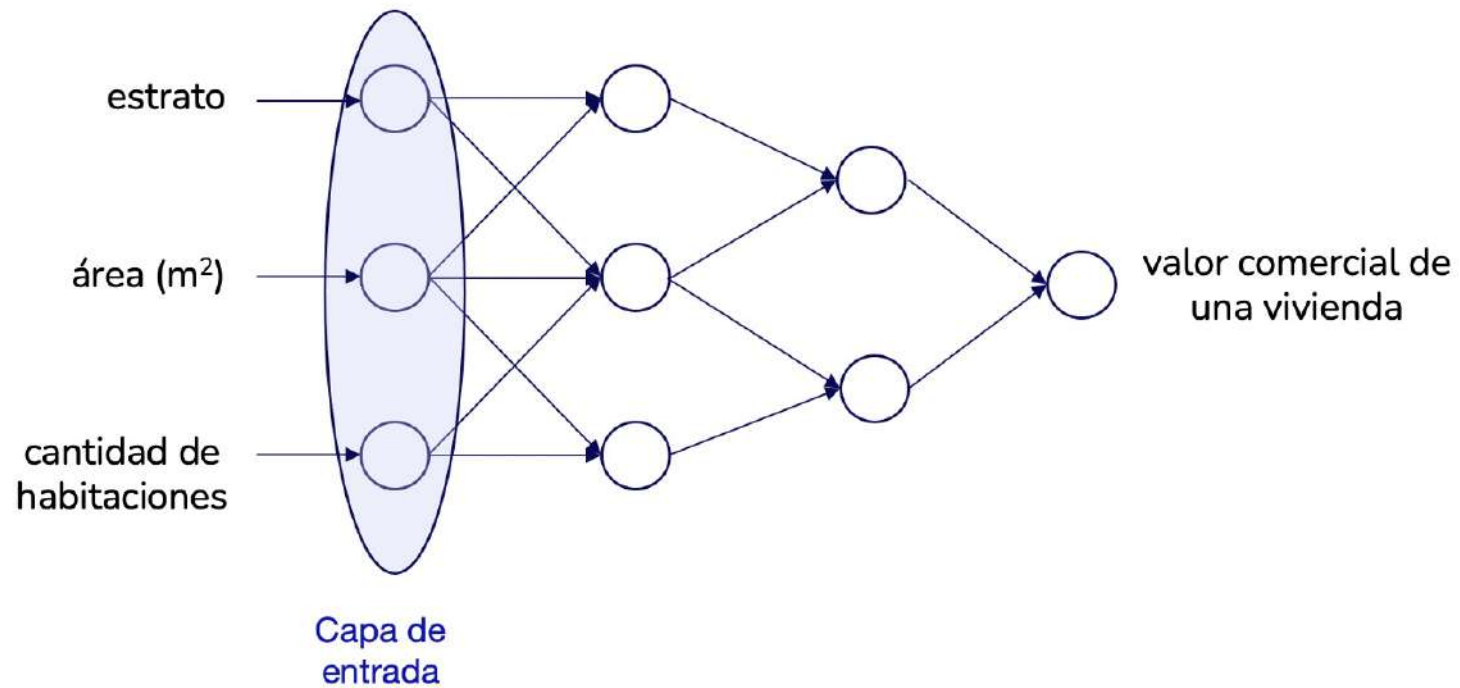


Variables  
independientes

Variable  
dependiente

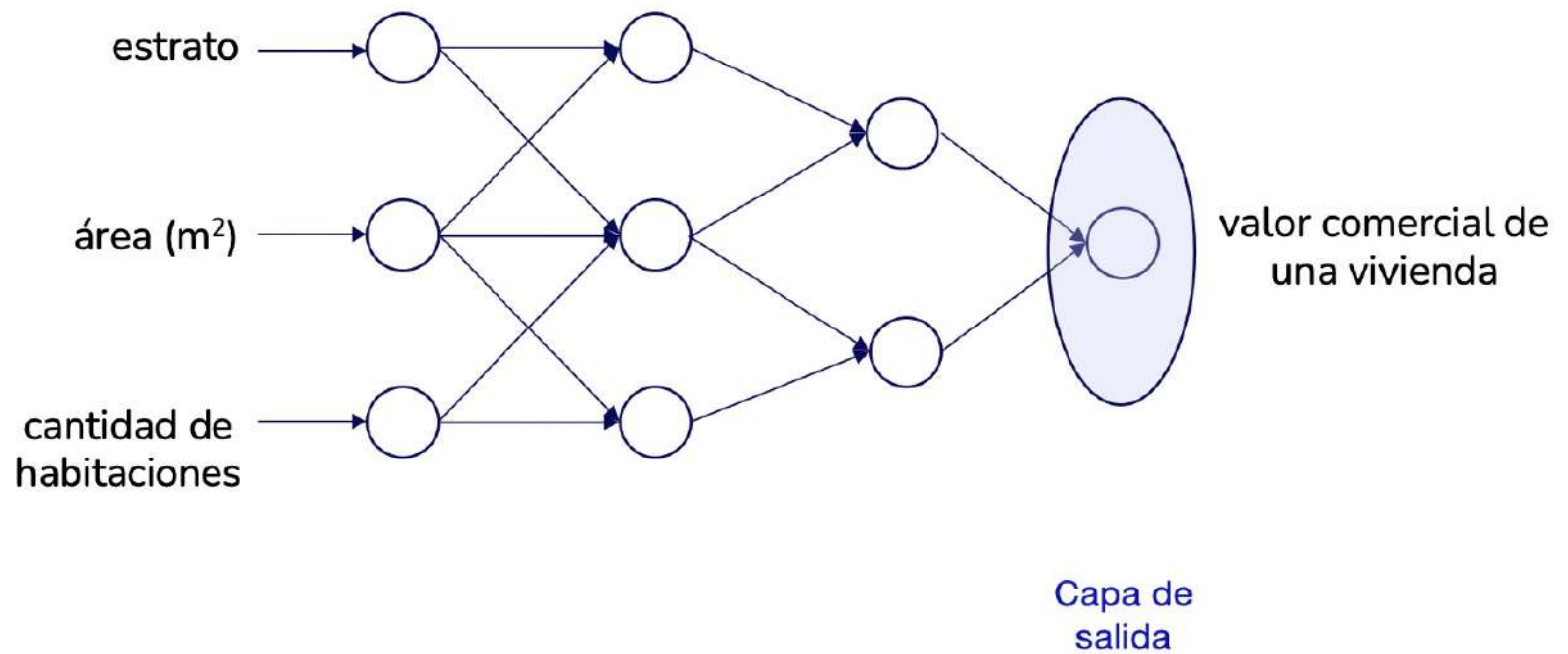
# Redes neuronales

---



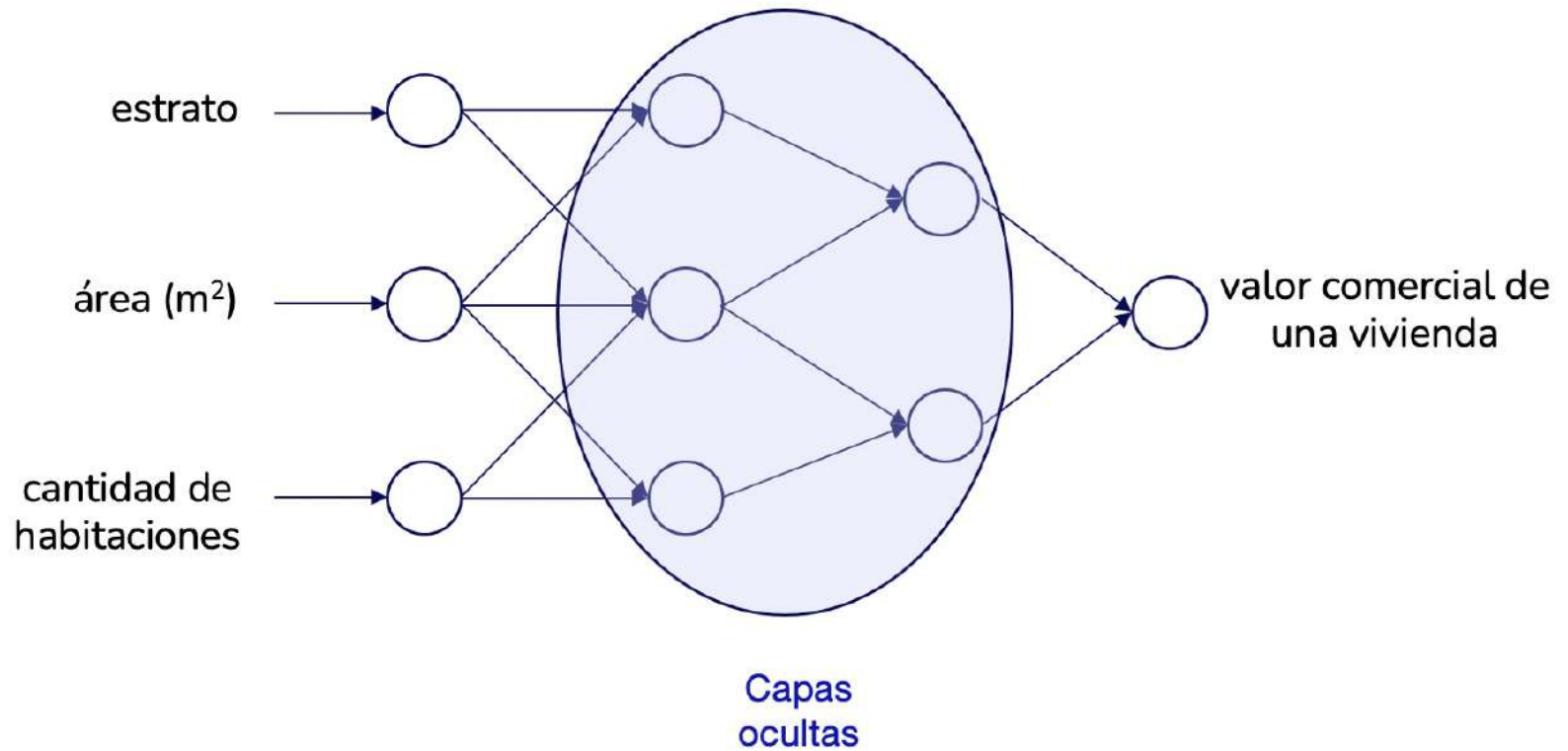
# Redes neuronales

---

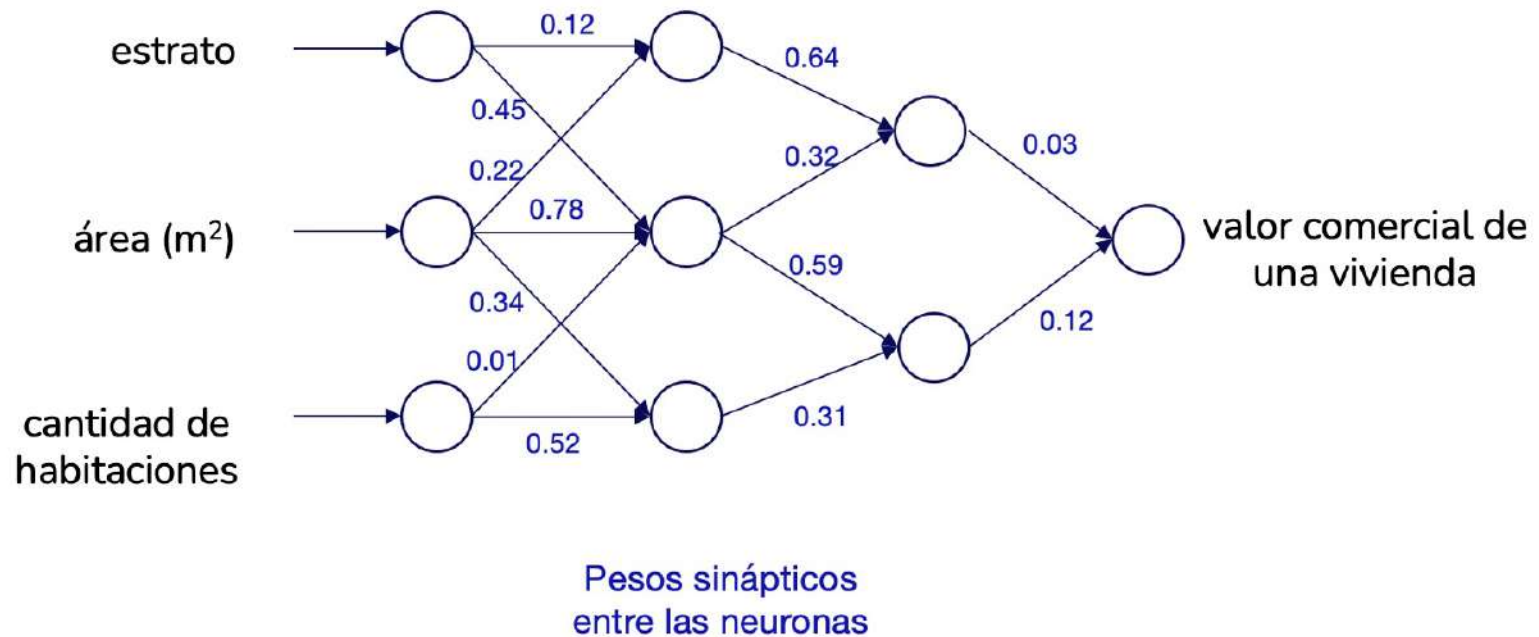


# Redes neuronales

---



# Redes neuronales



# Redes neuronales

---



**Estrato: 5**

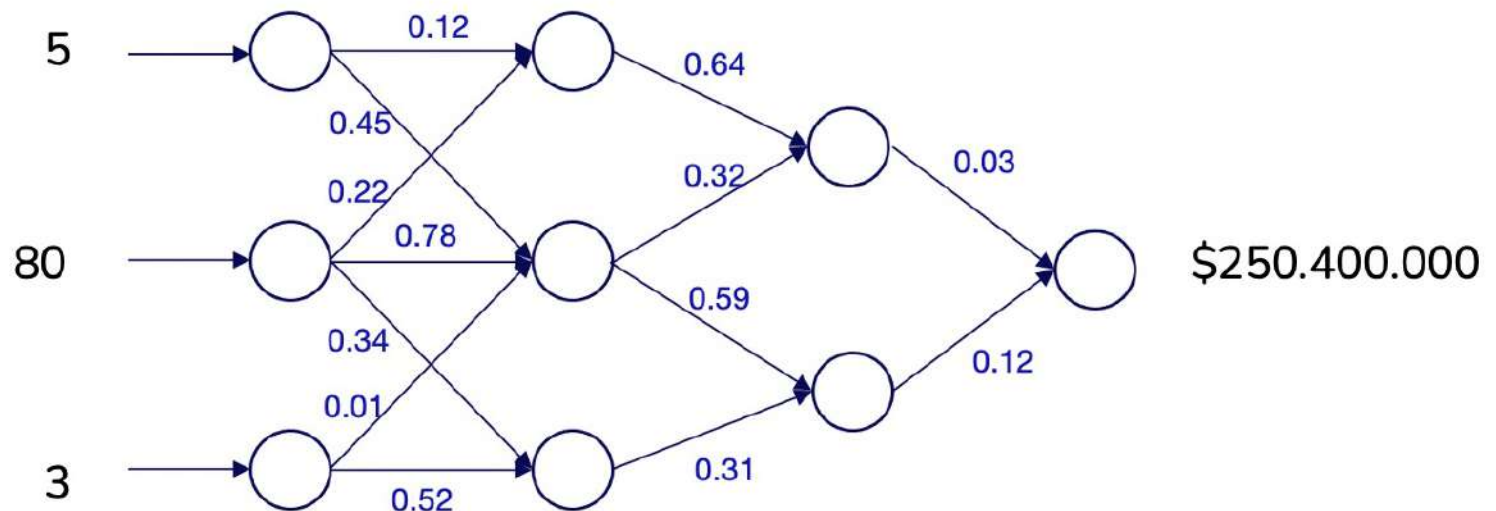
**Área: 80 m<sup>2</sup>**

**Habitaciones: 3**

**Precio: \$250.400.000**

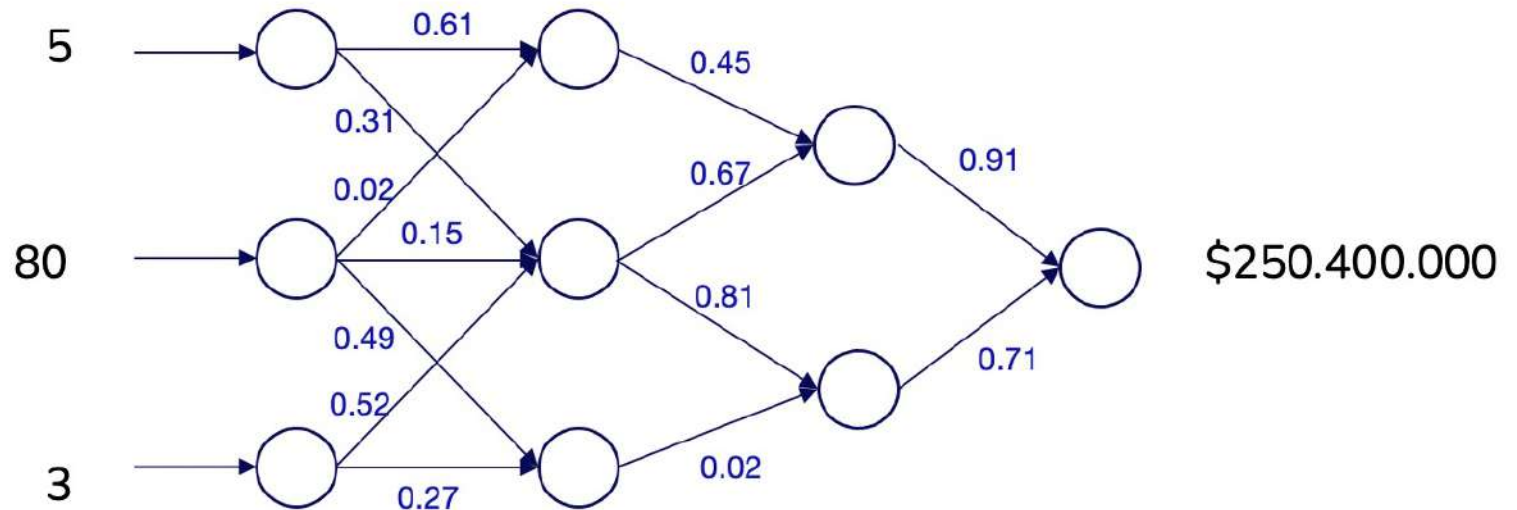


# Redes neuronales



**Estrato: 5**  
**Área: 80 m<sup>2</sup>**  
**Habitaciones: 3**  
**Precio: \$250.400.000**

# Redes neuronales



**Estrato: 5**  
**Área: 80 m<sup>2</sup>**  
**Habitaciones: 3**  
**Precio: \$250.400.000**

# Redes neuronales

---



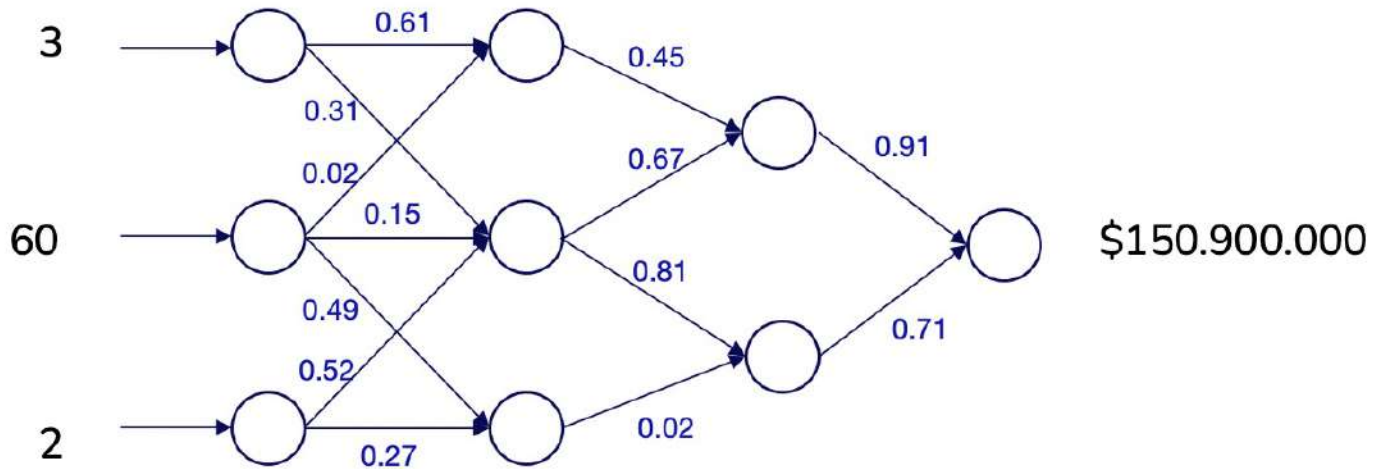
**Estrato: 3**

**Área: 60 m<sup>2</sup>**

**Habitaciones: 2**

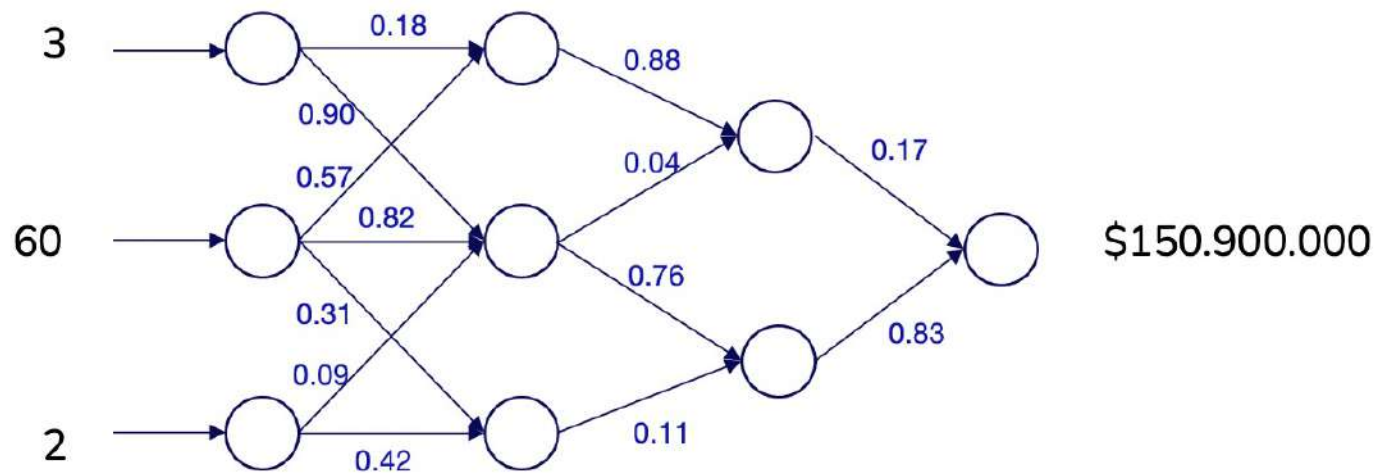
**Precio: \$159.900.000**

# Redes neuronales



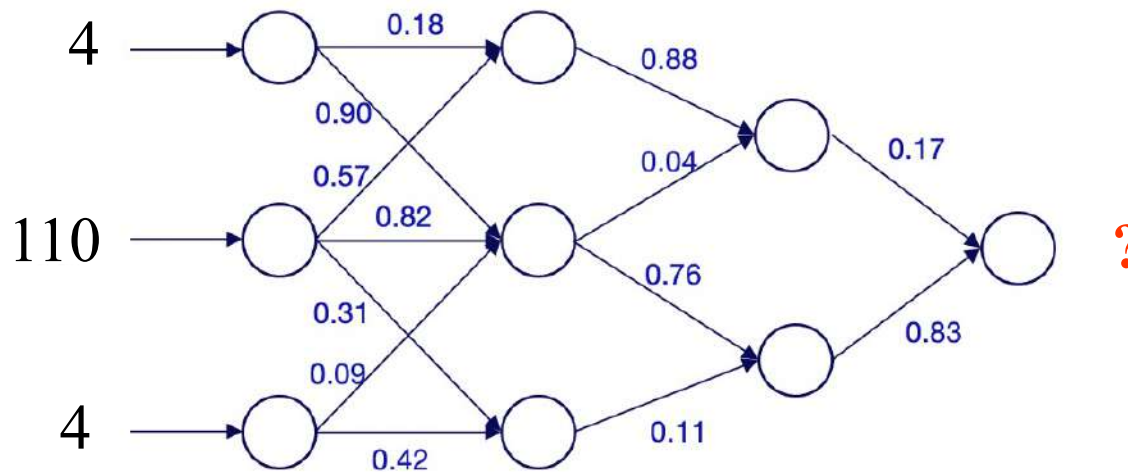
**Estrato: 3**  
**Área: 60 m<sup>2</sup>**  
**Habitaciones: 2**  
**Precio: \$159.900.000**

# Redes neuronales



**Estrato: 3**  
**Área: 60 m<sup>2</sup>**  
**Habitaciones: 2**  
**Precio: \$159.900.000**

# Redes neuronales

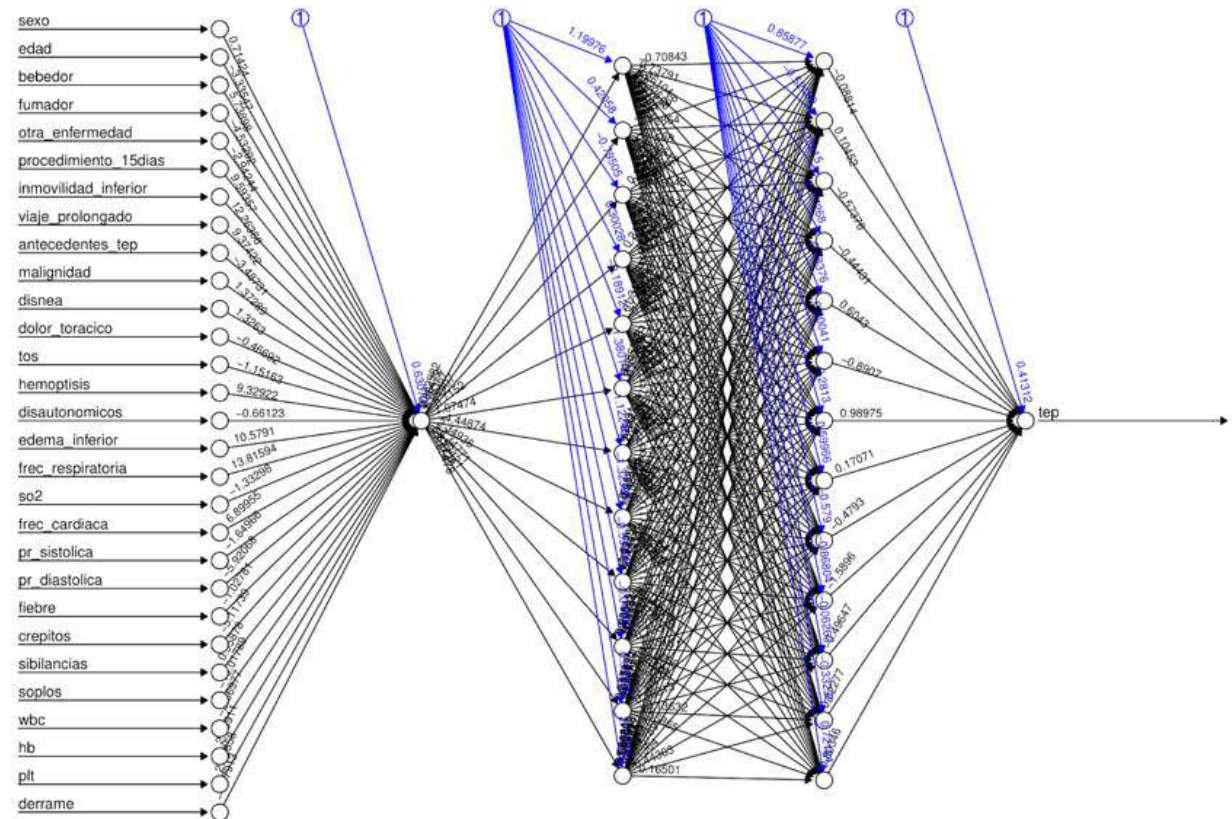


**Estrato:** 4  
**Área:** 110 m<sup>2</sup>  
**Habitaciones:** 4  
**Precio:** ?

# Redes neuronales

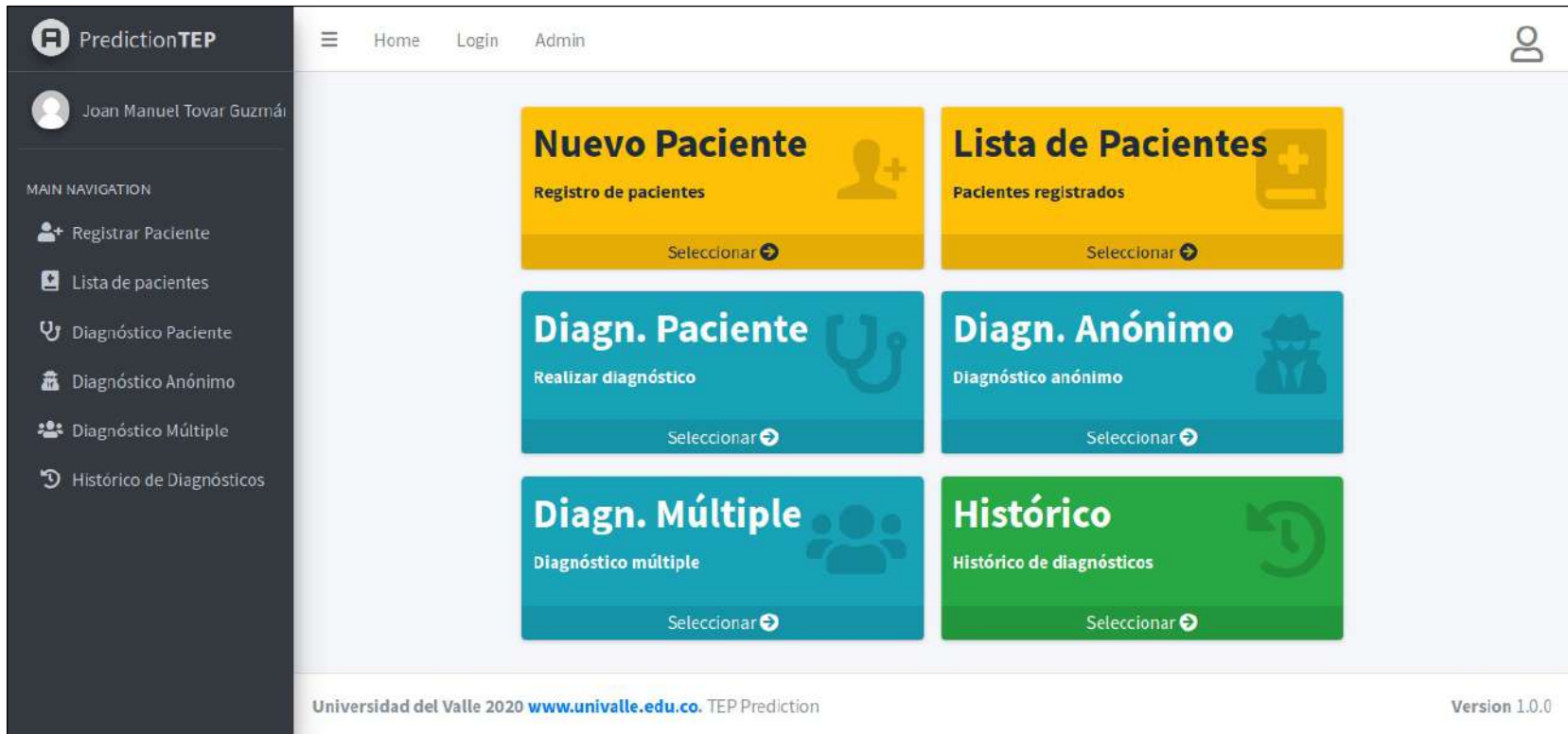
## Aplicación de técnicas de inteligencia artificial para la detección de Tromboembolismo pulmonar

Atributos
Sexo
Edad
Bebedor(a)
Fumador(a)
Otra Enfermedad
Procedimiento quirúrgico en los últimos 15 días
Inmovilidad miembros inferiores
Viaje prolongado
TEP - TVP previo
Malignidad
Disnea
Dolor Torácico
Tos
Hemoptisis
Síntomas disautonómicos
Edema de M. Inferiores
Frecuencia respiratoria
Saturación de la sangre(SO2)
Frecuencia cardíaca
Presión sistólica
Presión diastólica
WBC (Cuento de glóbulos blancos en la sangre)
HB (Nivel de hemoglobina)
PLT Cuento de plaquetas
Soplos
Fiebre
Crepitaciones
Sibilancias
Derrame
<b>TEP (Trombo embolismo pulmonar)</b>





# Redes neuronales





# Redes neuronales

PredictionTEP

Home Login Admin

Juan Manuel Tovar Guzmán

NAVIGATION

- Registrar Paciente
- Lista de pacientes
- Diagnóstico Paciente
- Diagnóstico Archivos
- Diagnóstico Multiple
- Historia de Diagnósticos

Registro de datos Médicos

Paciente\*

Perez Cifuentes Juan Camilo - 1919191919

☒ Bebedor

☒ Fumador

☐ Otra enfermedad

☐ Procedimiento Quirúrgico o Traumatismo grave en los últimos 15 días

☐ Inestabilidad en miembro inferior

☒ Viaje prolongado

☐ Antecedentes de tromboembolismo pulmonar (TEP/TVE)

☐ Malignidad

☒ Drona

☐ Dolor torácico

☒ Tos

☐ Hemoptisis

☐ Disautonomía

☐ Síndrome inferior

☐ Fiebre

☐ Cefaleas

☐ Síbilancias

Frecuencia respiratoria\*

36 - 40

Saturación de oxígeno en sangre (SO2)\*

88 - 94

Frecuencia cardíaca\*

60 - 100

Presión sistólica\*

91 - 130

Presión diastólica\*

41 - 50

Temperatura\*

AGÜTICO

Cuento de glóbulos blancos (WBC)\*

10000 - 12000

Hemoglobina (Hb)\*

10 - 11,49

Cuento de plaquetas (PLT)\*

100000 - 149999

Demencia\*

NO

Diagnosticar

Universidad del Valle 2020 [www.univalle.edu.co](http://www.univalle.edu.co) TEP Prediction Version 1.0.0

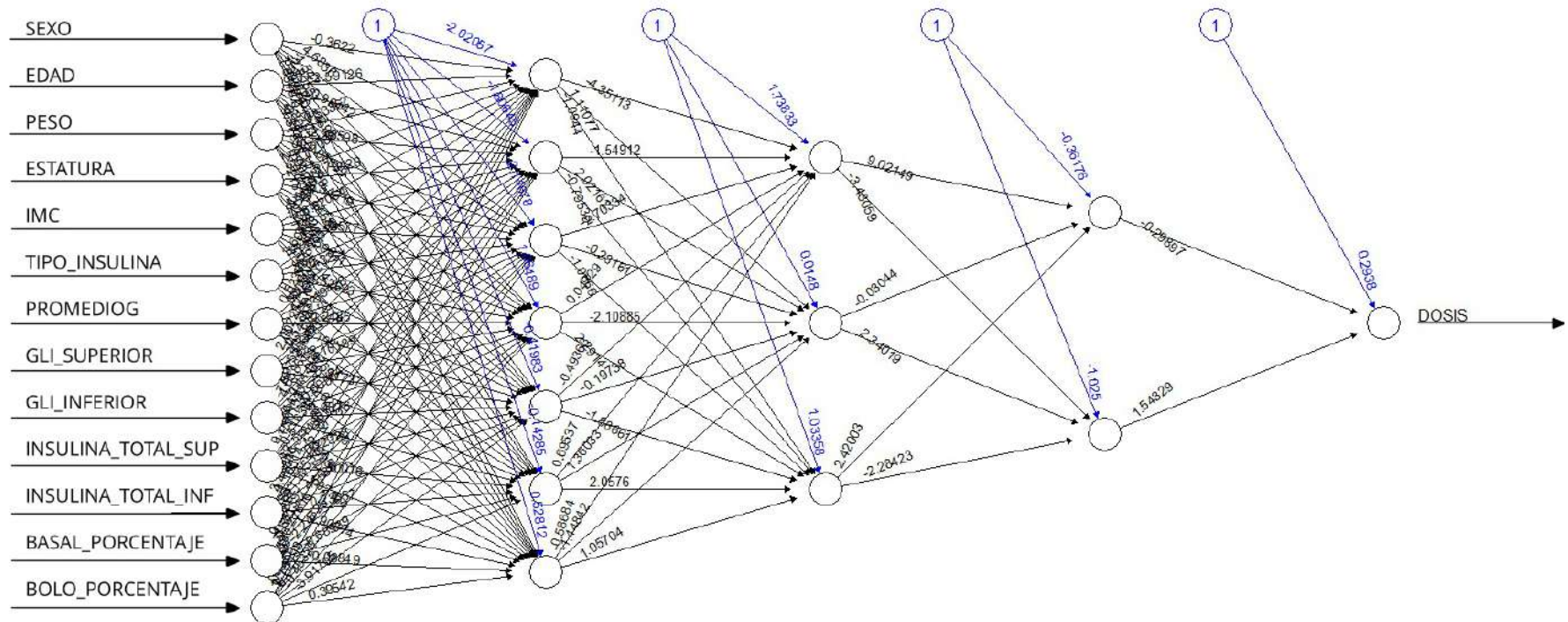
**El paciente ha sido diagnosticado con TEP**

Perez Cifuentes Juan Camilo - 1919191919

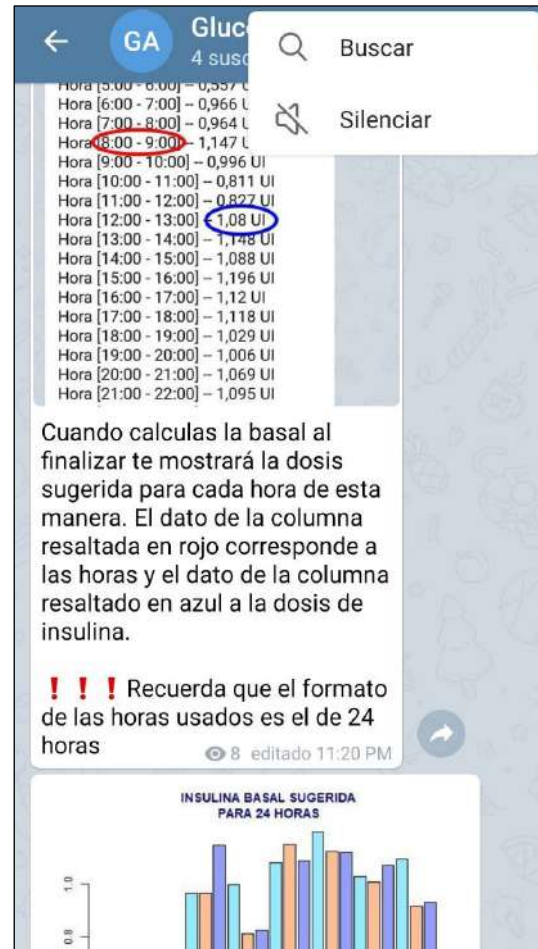


# Redes neuronales

## Aplicación de técnicas de inteligencia artificial para el ajuste de insulina en pacientes con diabetes tipo I

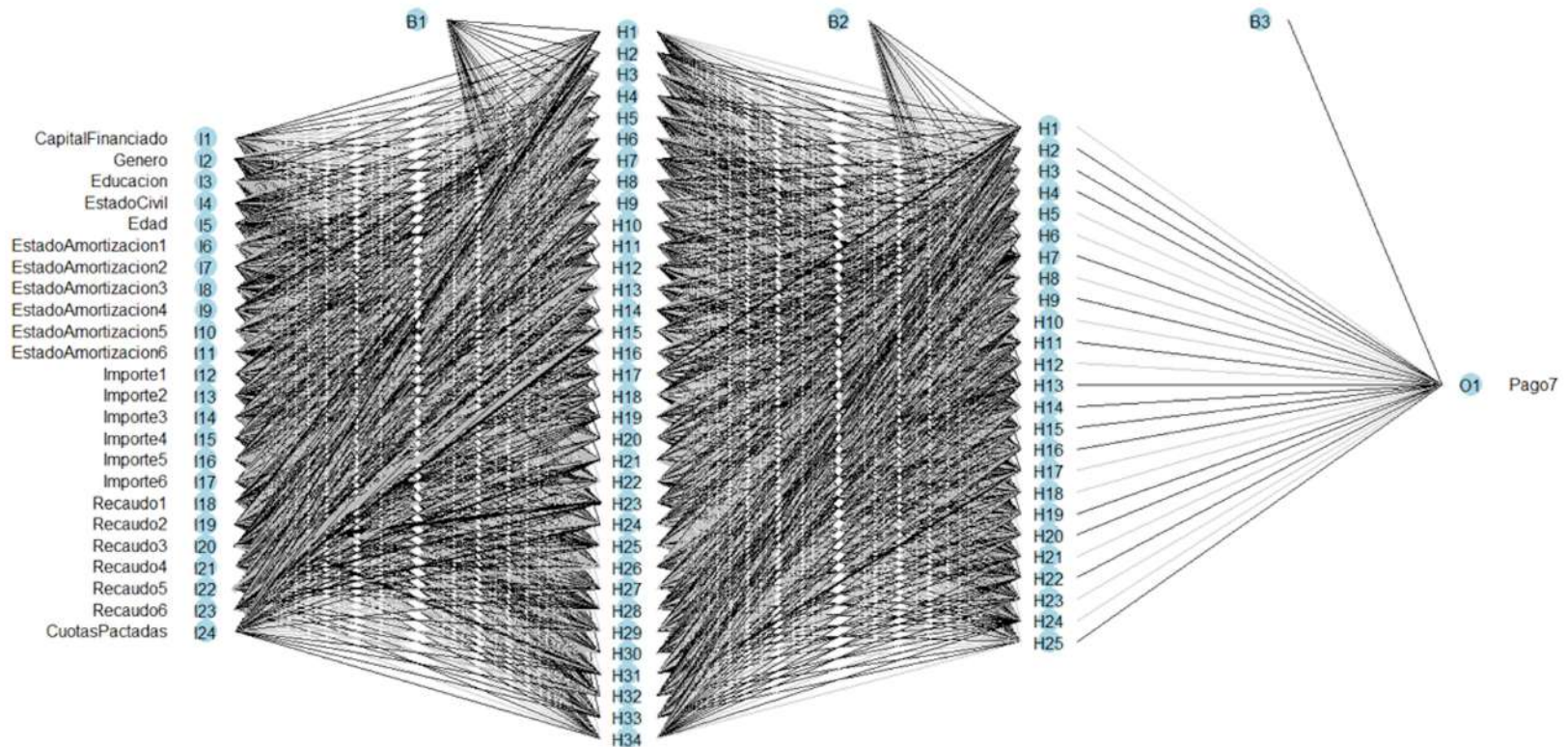


# Redes neuronales



# Redes neuronales

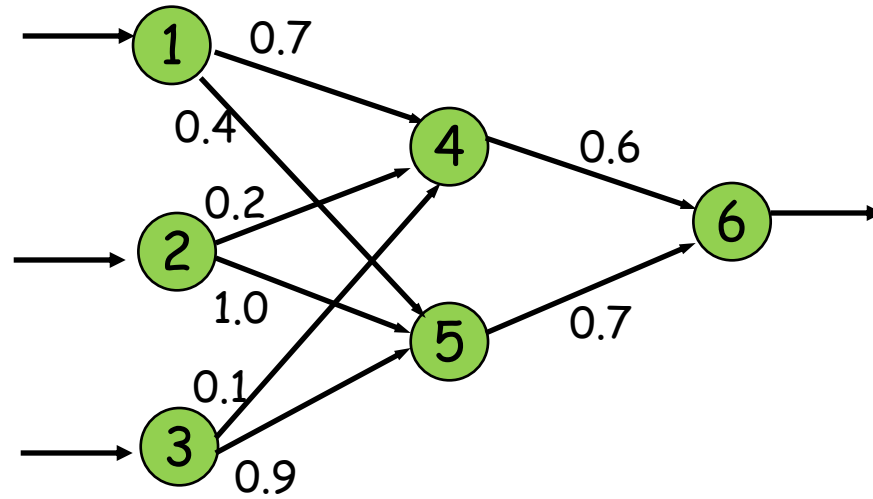
## Predicción de riesgo crediticio en Colombia usando técnicas de inteligencia artificial



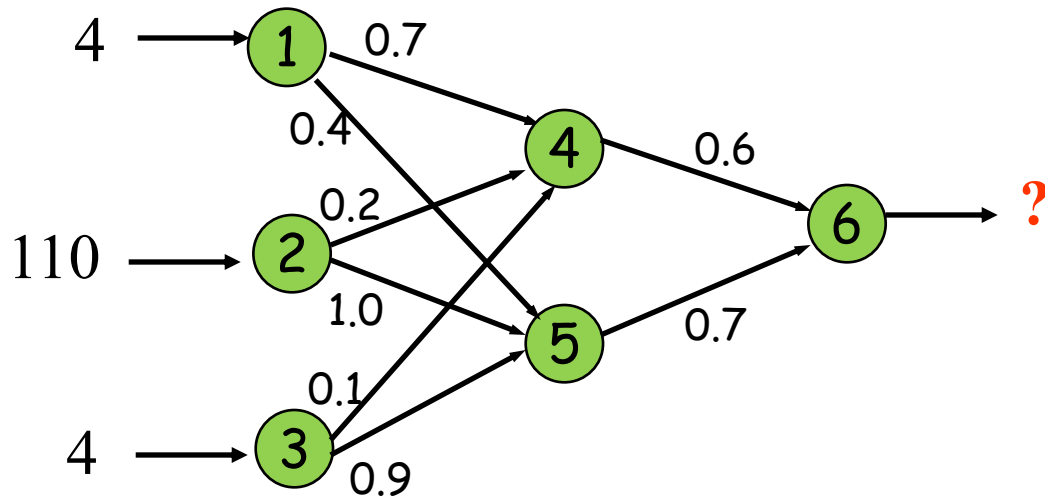


# Redes neuronales

---



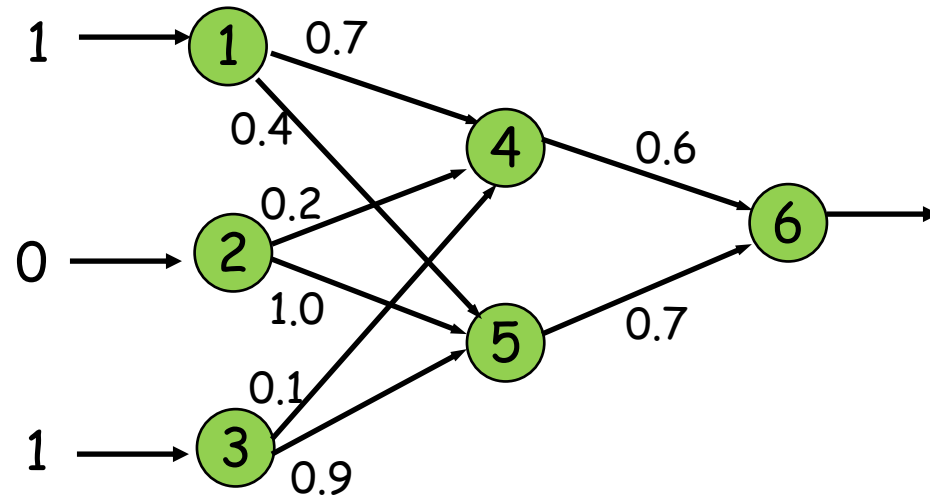
# Redes neuronales



**Estrato:** 4  
**Área:** 110 m<sup>2</sup>  
**Habitaciones:** 4  
**Precio:** ?

# Redes neuronales

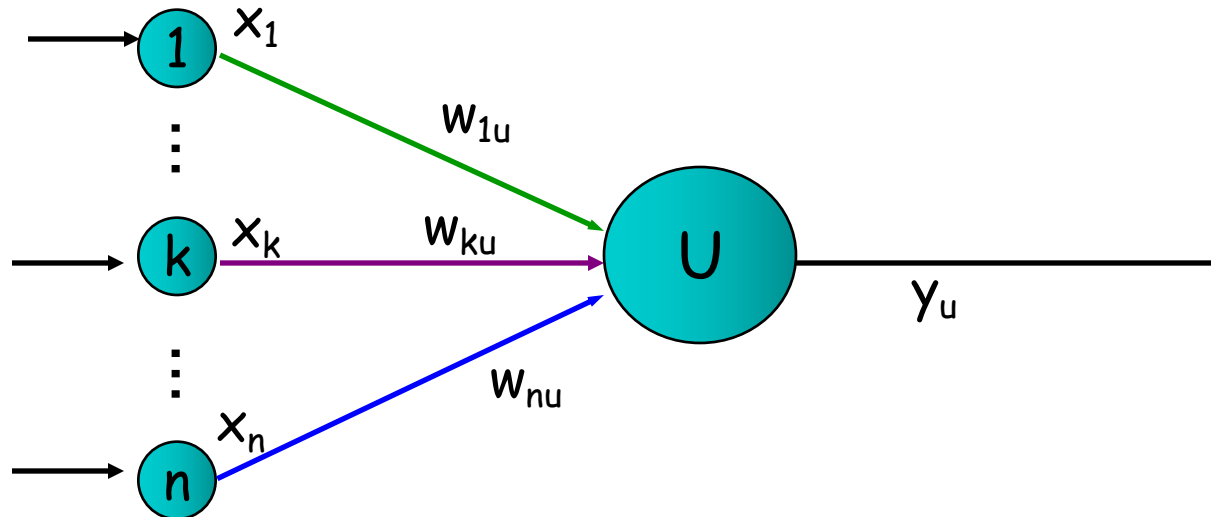
---





# Redes neuronales

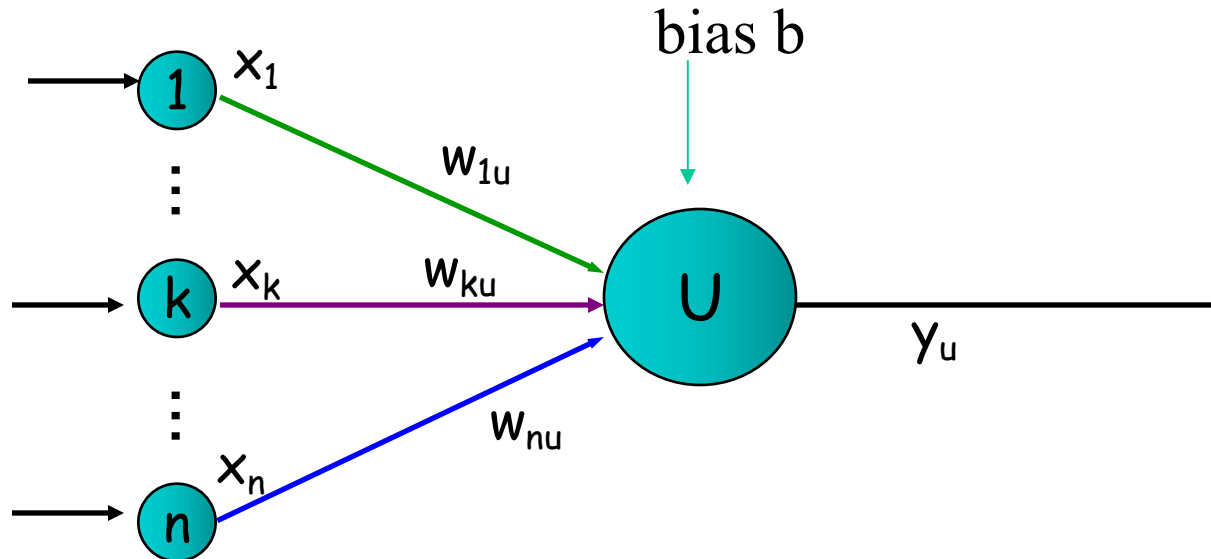
## Estructura de cada neurona



Entrada Neta  $U = \sum_{i=1}^n x_i * w_{iu}$

# Redes neuronales

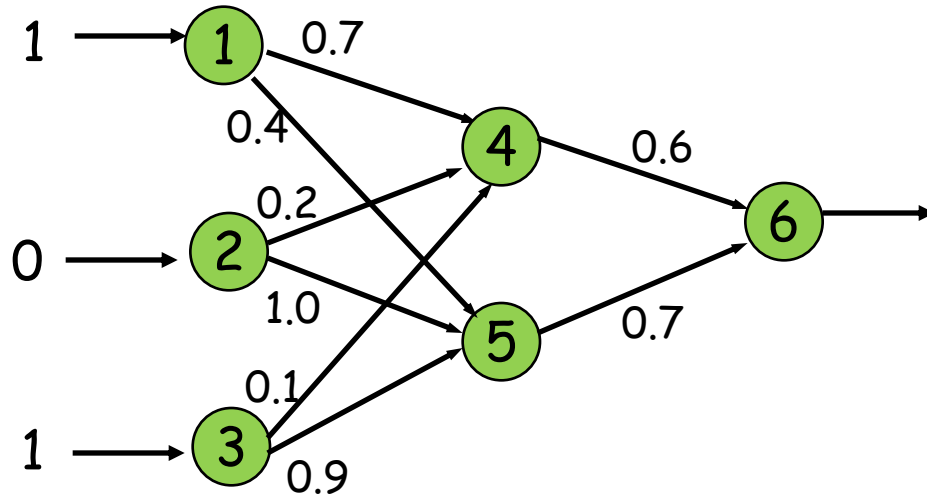
## Estructura de cada neurona



$$\text{Entrada Neta } U = \sum_{i=1}^n x_i * w_{iu} + b$$

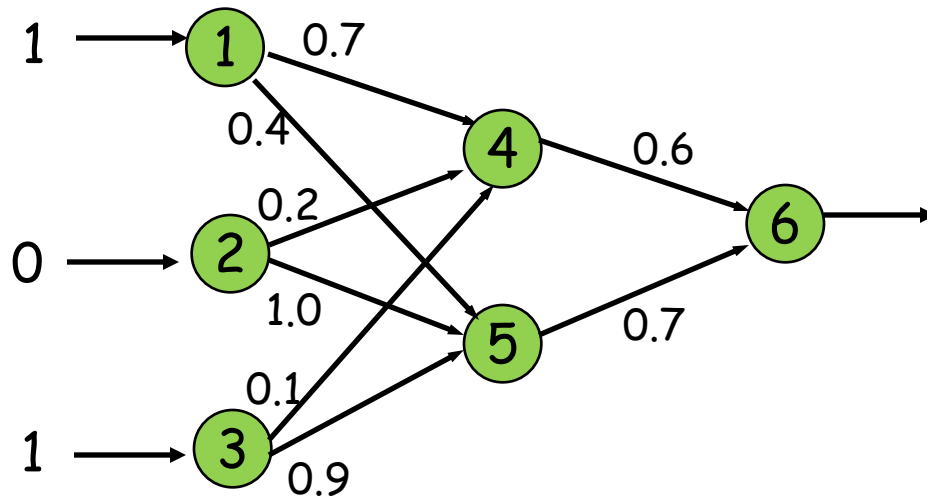
# Redes neuronales

---



# Redes neuronales

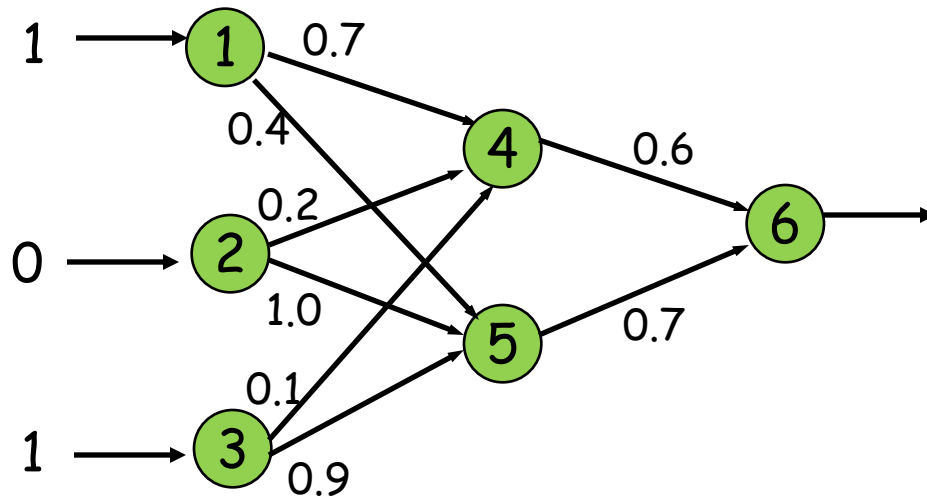
---



Entrada Neta para 4 = ?

# Redes neuronales

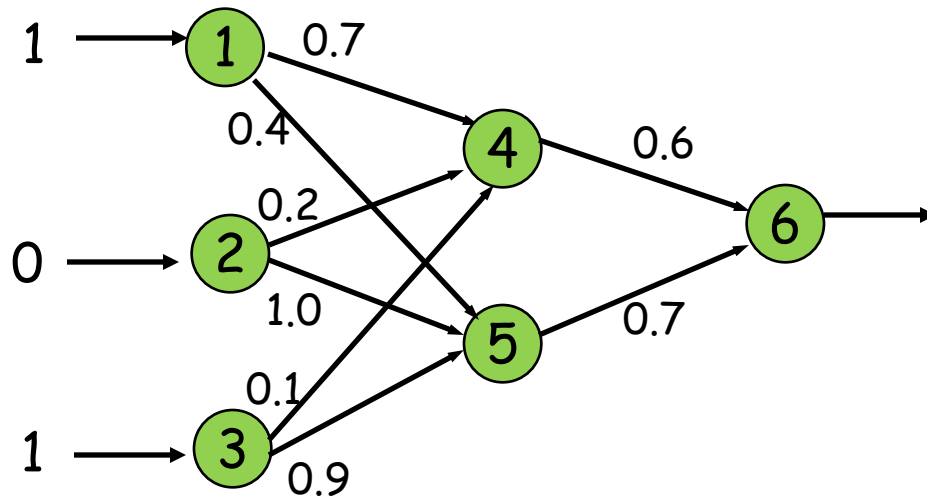
---



Entrada Neta para 4 =  $(1 \cdot 0.7) + (0 \cdot 0.2) + (1 \cdot 0.1) = 0.8$

# Redes neuronales

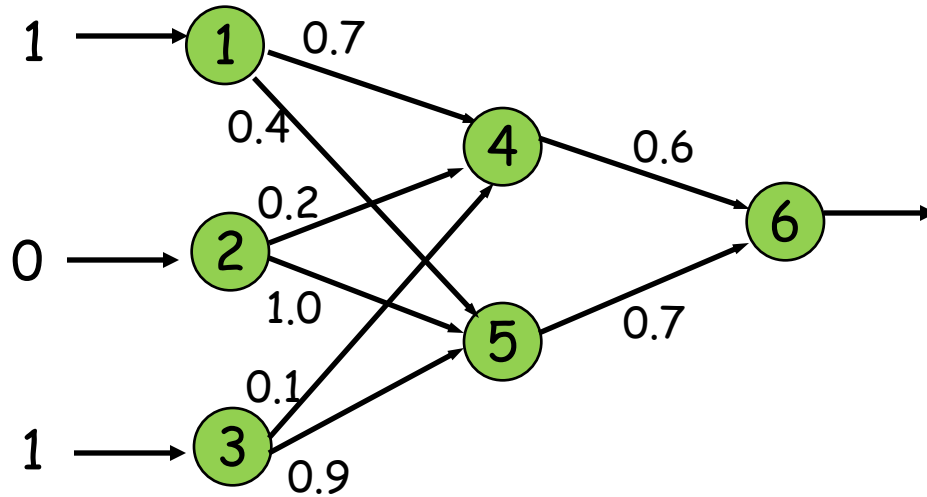
---



Entrada Neta para 5 = ?

# Redes neuronales

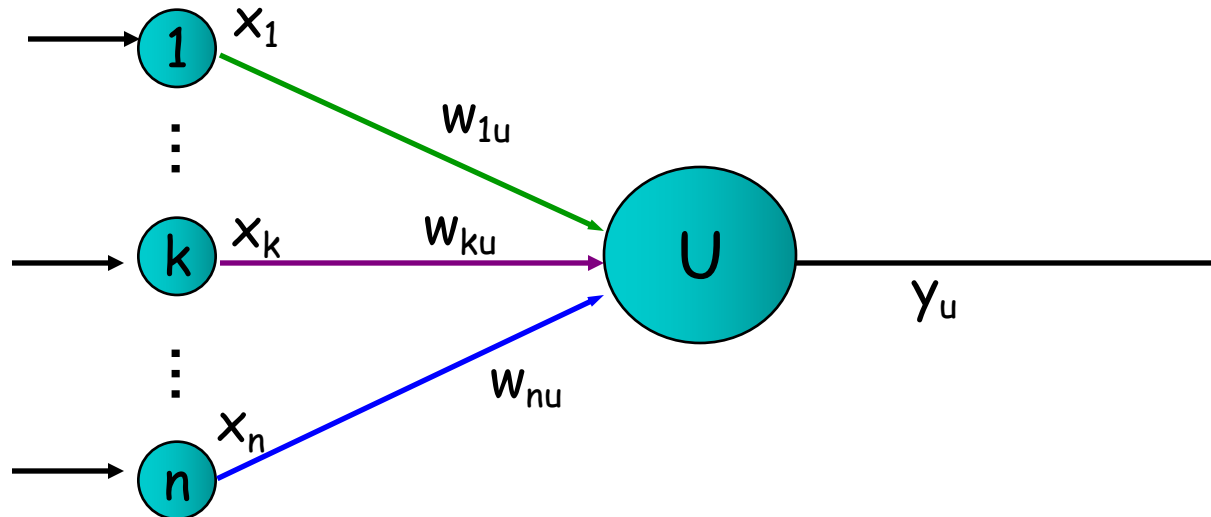
---



Entrada Neta para **5** =  $(1 \cdot 0.4) + (0 \cdot 1.0) + (1 \cdot 0.9) = 1.3$

# Redes neuronales

## Estructura de cada neurona



$$\text{Entrada Neta } U = \sum_{i=1}^n x_i * w_{iu}$$

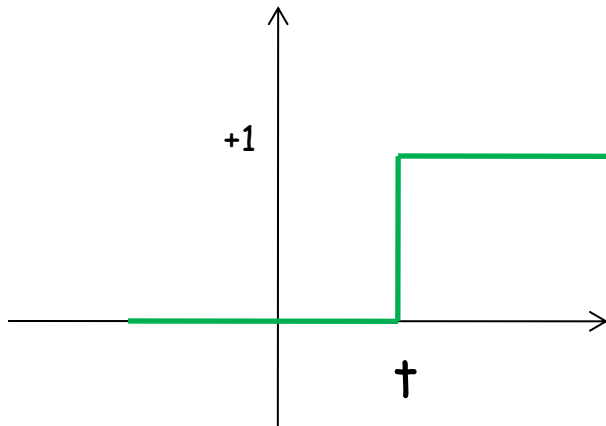
$$y_u = f(U)$$

- $f$  se conoce como la función de activación
- $y_u$  es la salida de la neurona

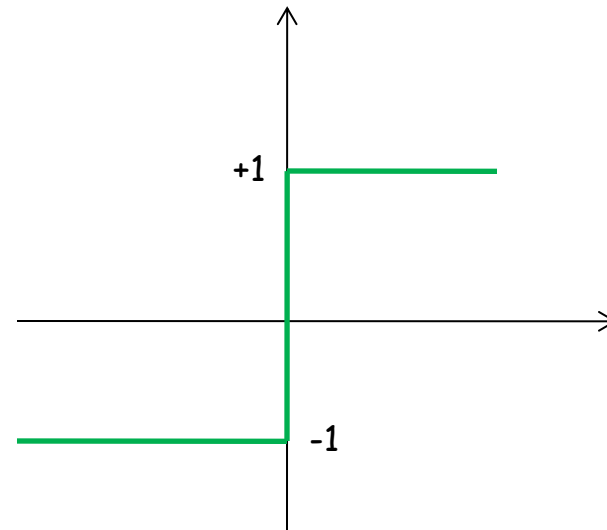


# Redes neuronales

## Funciones de activación



$$\text{escalon}(u) = \begin{cases} 1, & \text{si } u \geq t \\ 0, & \text{si } u < t \end{cases}$$

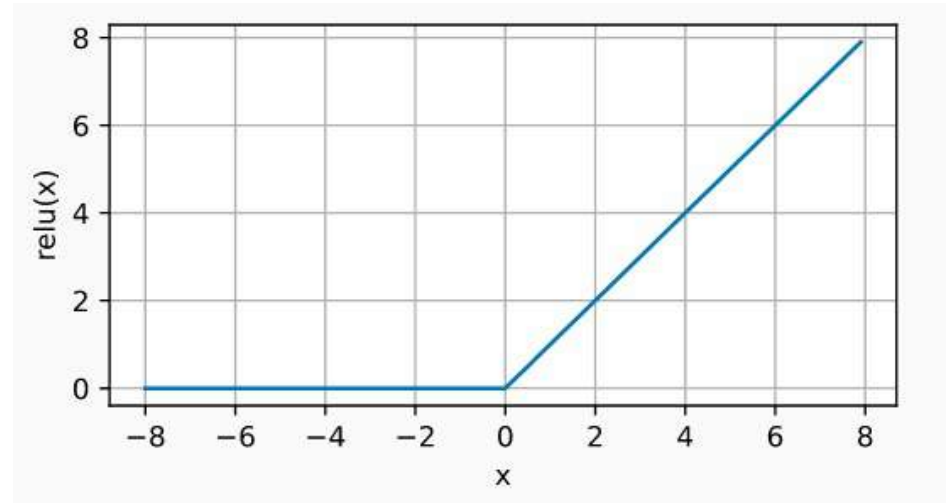


$$\text{signo}(u) = \begin{cases} 1, & \text{si } u \geq 0 \\ -1, & \text{si } u < 0 \end{cases}$$

# Redes neuronales

## Funciones de activación

La función de activación **ReLU** (*Rectified Linear Unit*) aplica una transformación no lineal muy simple, activa la neurona solo si la entrada está por encima de cero



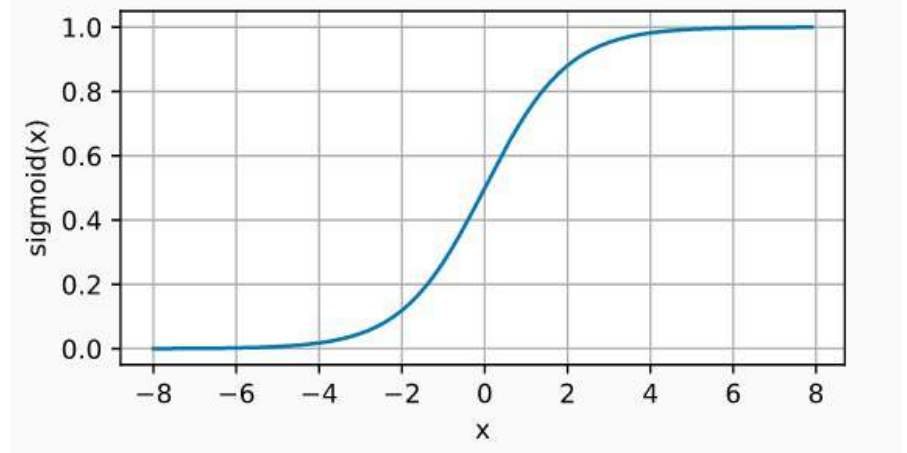
$$\text{ReLU}(x) = \max(x, 0)$$

# Redes neuronales

---









## Funciones de activación



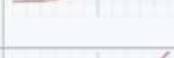


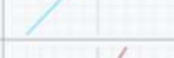
La función sigmoide transforma valores en el rango de  $(-\infty, \infty)$  a valores en el rango  $(0, 1)$ .



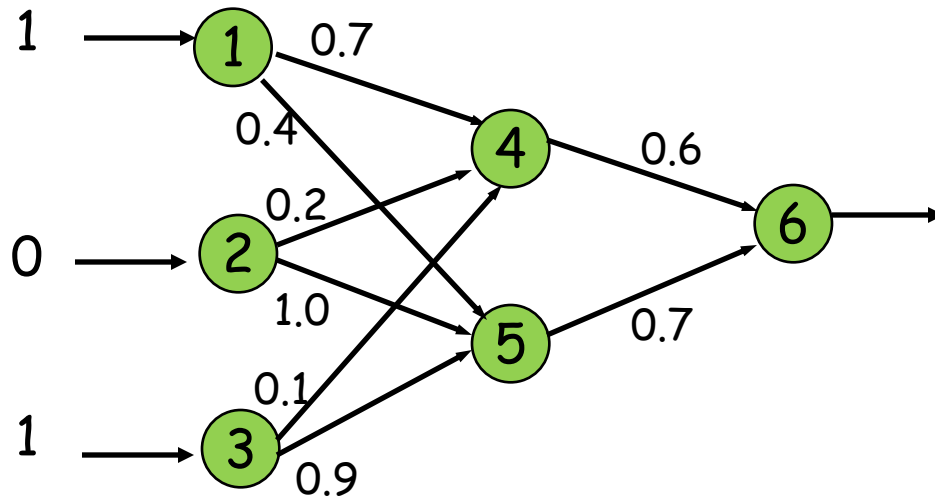
# Redes neuronales

## Funciones de activación

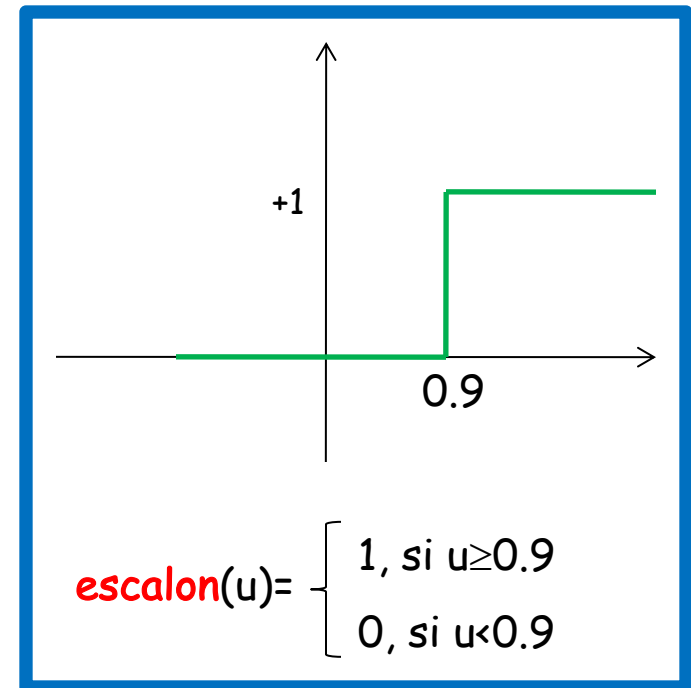
Name	Plot	Equation
Identity		$f(x) = x$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Logistic (a.k.a Sigmoid or Soft step)		$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$ <sup>[1]</sup>
TanH		$f(x) = \tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$
ArcTan		$f(x) = \tan^{-1}(x)$
ElliotSig <sup>[9][10][11]</sup> Softsign <sup>[12][13]</sup>		$f(x) = \frac{x}{1 +  x }$
Inverse square root unit (ISRU) <sup>[14]</sup>		$f(x) = \frac{x}{\sqrt{1 + \alpha x^2}}$
Inverse square root linear unit (ISRLU) <sup>[14]</sup>		$f(x) = \begin{cases} \frac{x}{\sqrt{1 + \alpha x^2}} & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$

Name	Plot	Equation
Inverse square root linear unit (ISRLU) <sup>[14]</sup>		$f(x) = \begin{cases} \frac{x}{\sqrt{1 + \alpha x^2}} & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$
Square Nonlinearity (SQNL) <sup>[11]</sup>		$f(x) = \begin{cases} 1 & : x > 2.0 \\ x - \frac{x^2}{4} & : 0 \leq x \leq 2.0 \\ x + \frac{x^2}{4} & : -2.0 \leq x < 0 \\ -1 & : x < -2.0 \end{cases}$
Rectified linear unit (ReLU) <sup>[15]</sup>		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$
Bipolar rectified linear unit (BReLU) <sup>[16]</sup>		$f(x_i) = \begin{cases} ReLU(x_i) & \text{if } i \bmod 2 = 0 \\ -ReLU(-x_i) & \text{if } i \bmod 2 \neq 0 \end{cases}$
Leaky rectified linear unit (Leaky ReLU) <sup>[17]</sup>		$f(x) = \begin{cases} 0.01x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$
Parametric rectified linear unit (PReLU) <sup>[18]</sup>		$f(\alpha, x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$

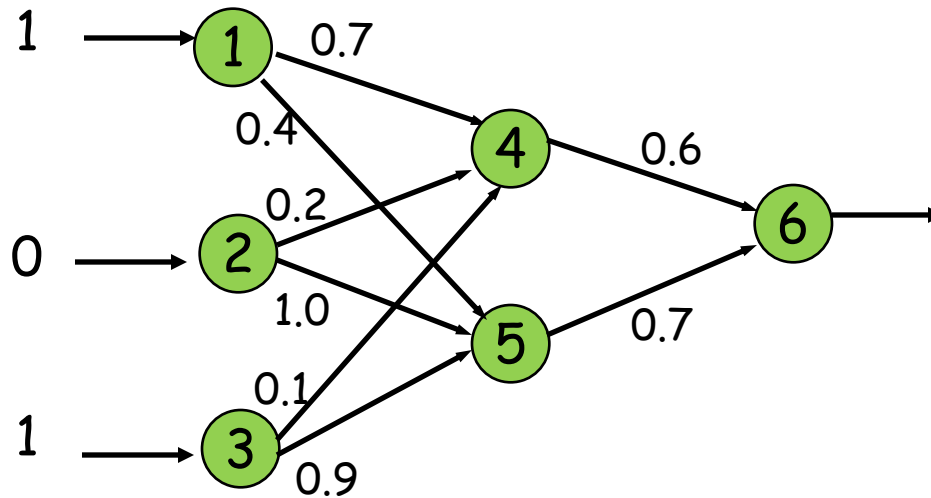
# Redes neuronales



- Calcule la salida de ④ y ⑤

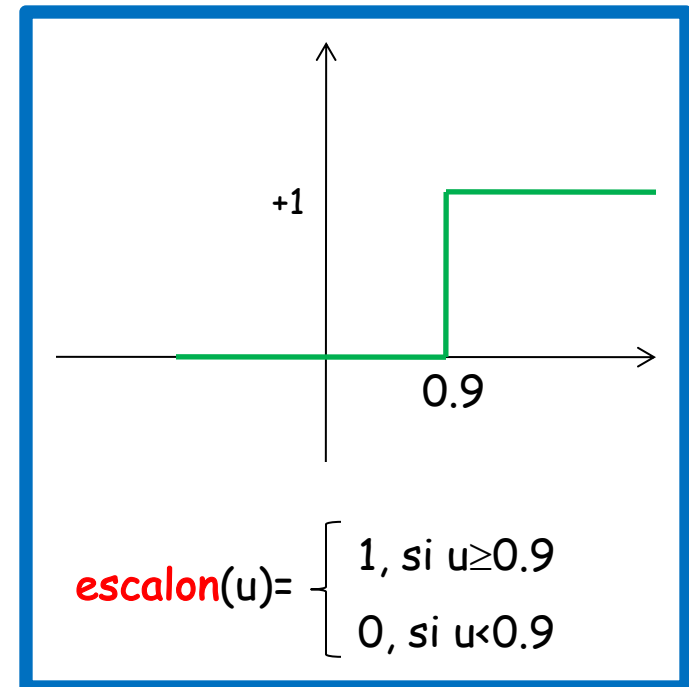


# Redes neuronales

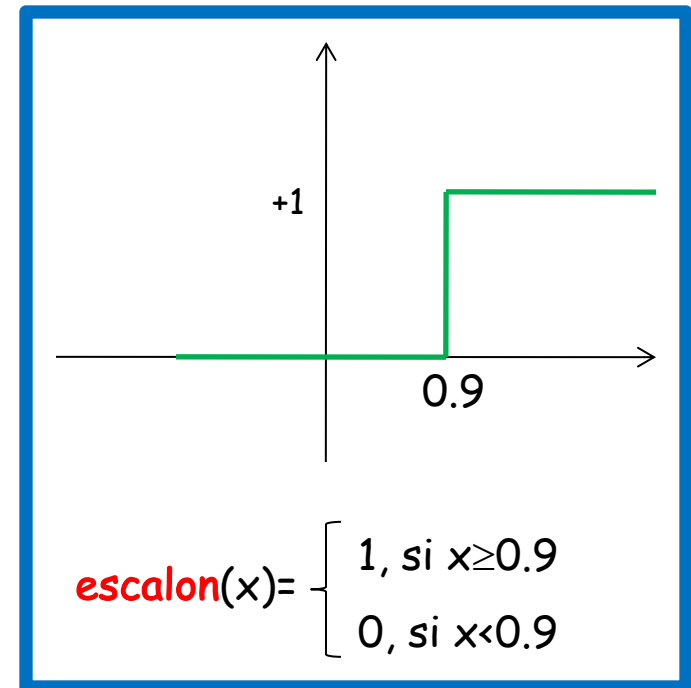
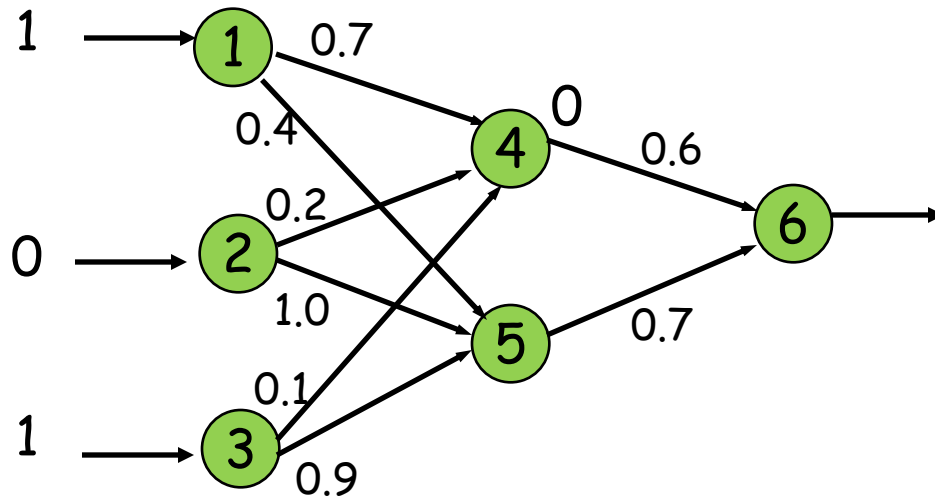


- Calcule la salida de 4 y 5

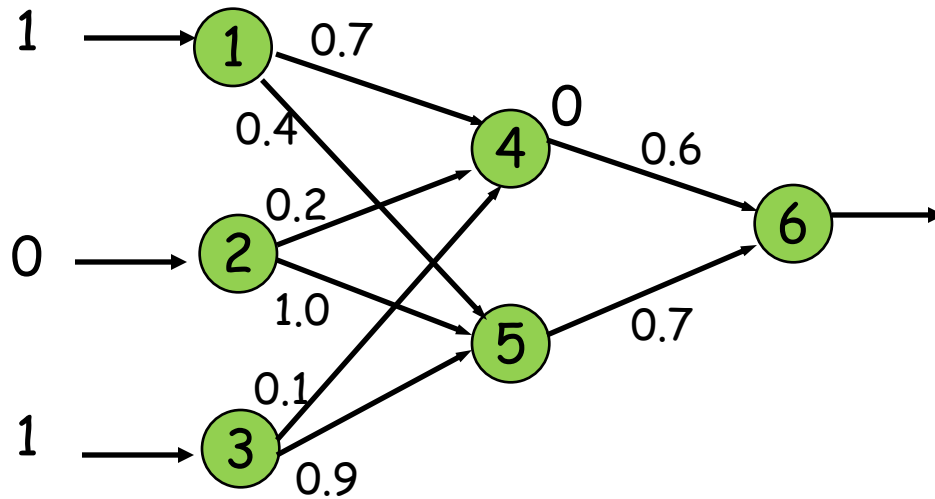
Entrada Neta para 4 = 0.8



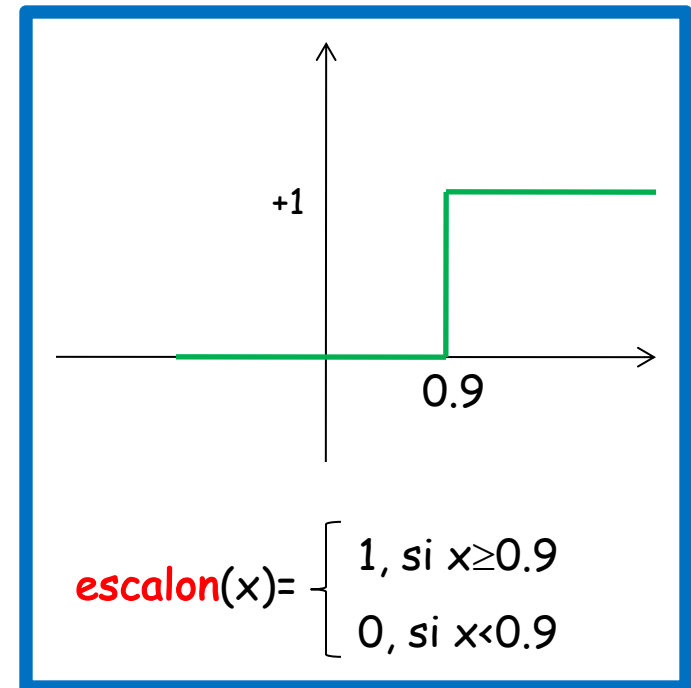
# Redes neuronales



# Redes neuronales

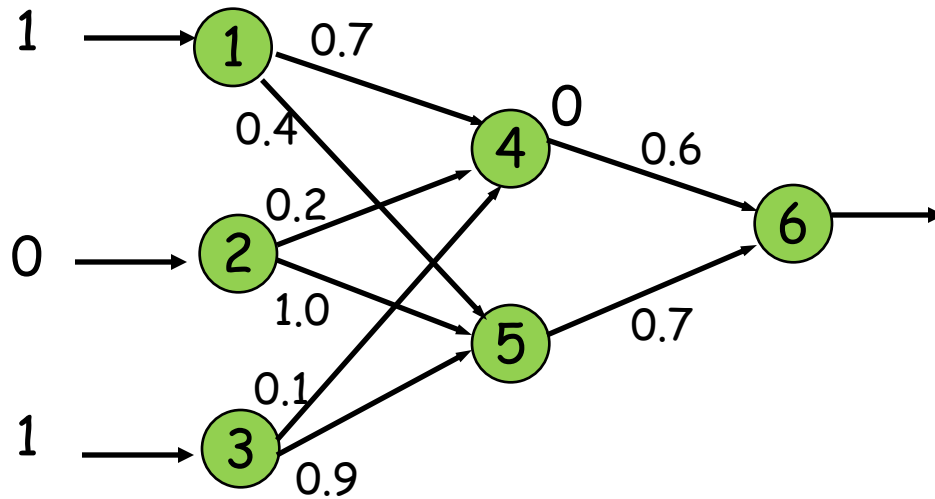


Entrada Neta para ⑤ = ?

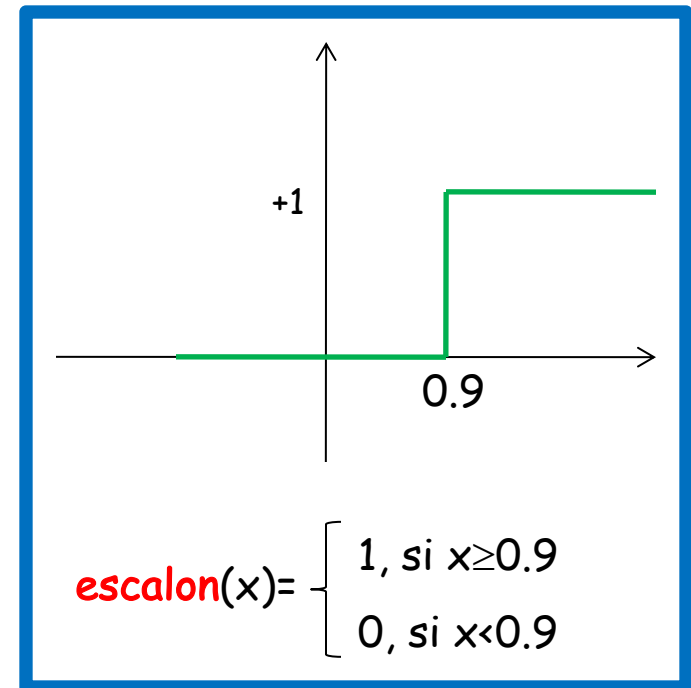




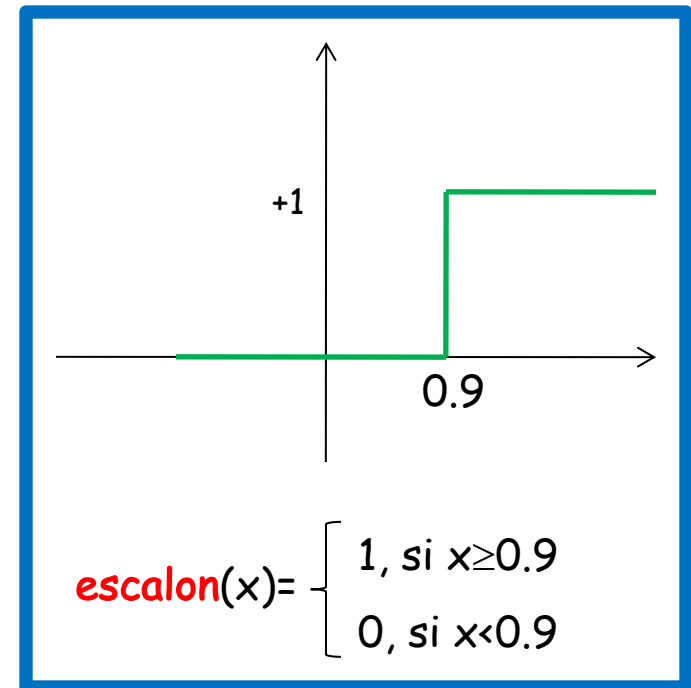
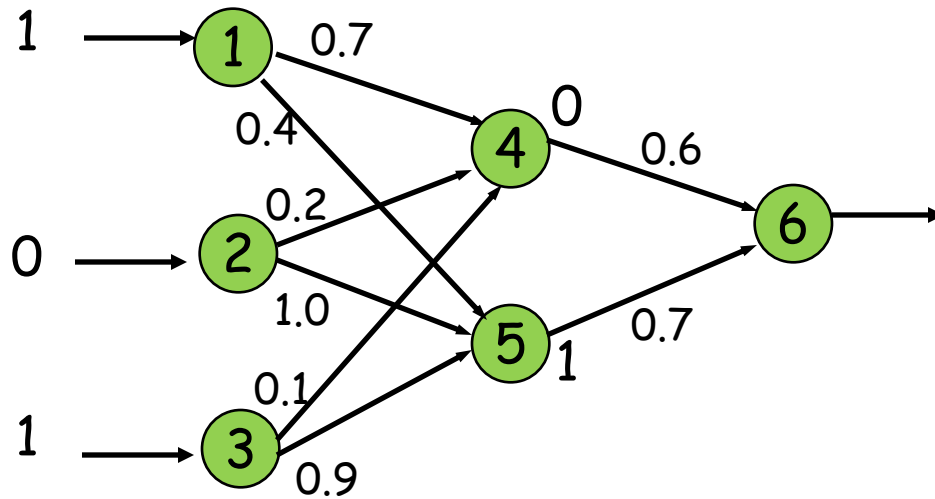
# Redes neuronales



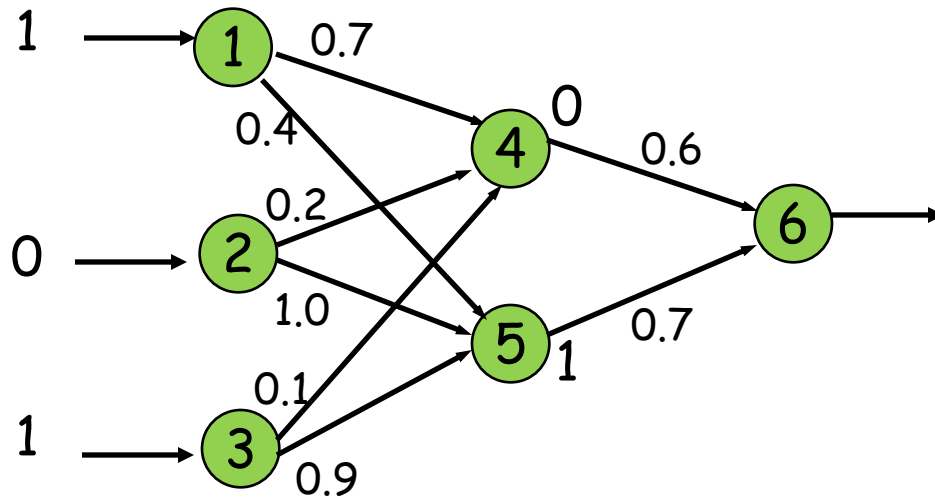
Entrada Neta para ⑤ = 1.3



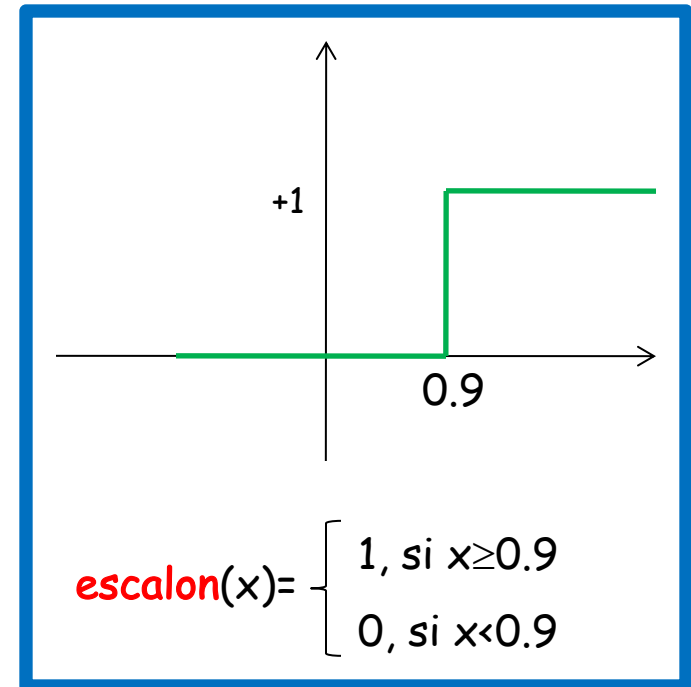
# Redes neuronales



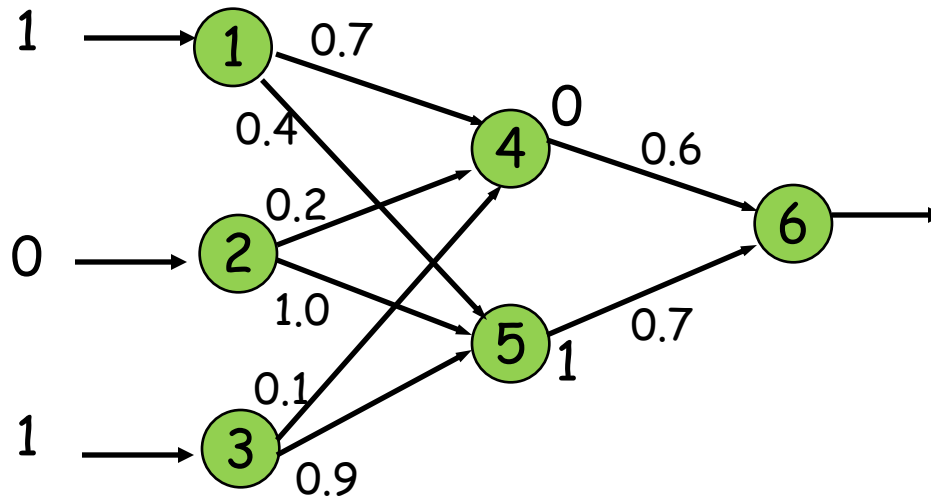
# Redes neuronales



- Calcule la salida de la red

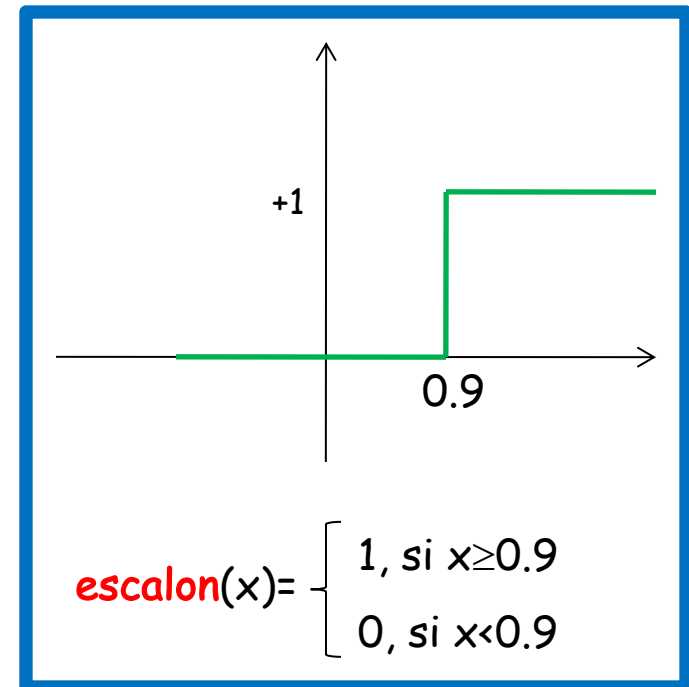


# Redes neuronales

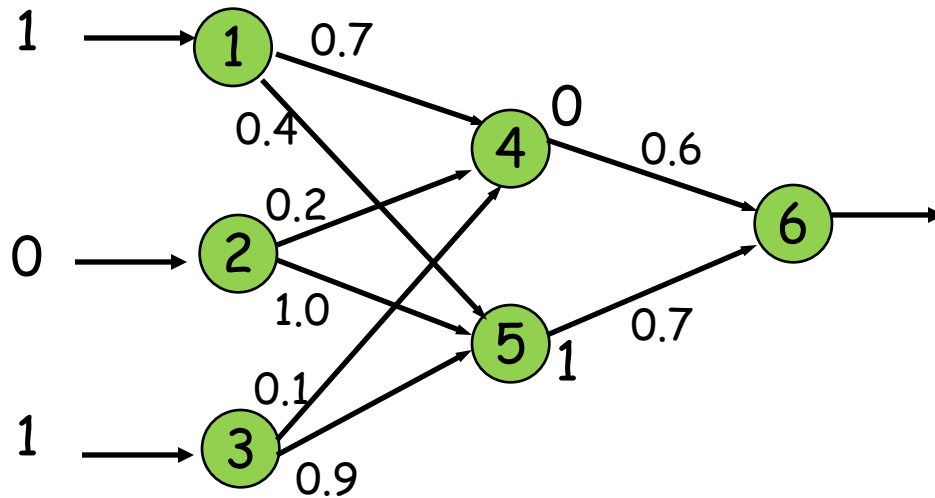


- Calcule la salida de la red

Entrada Neta para 6 = ?

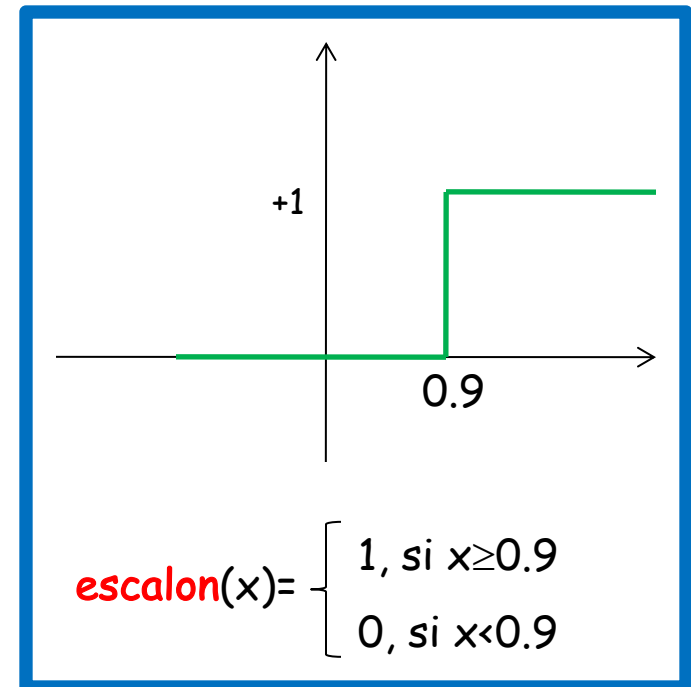


# Redes neuronales

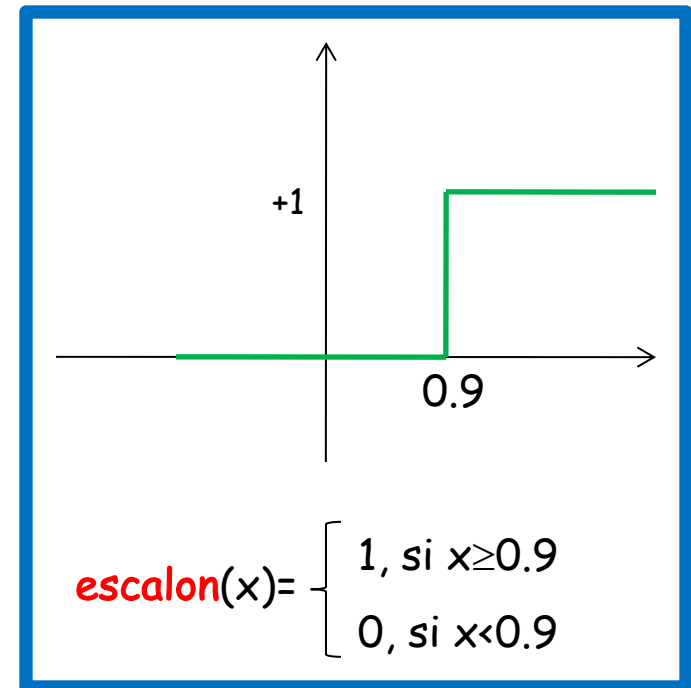
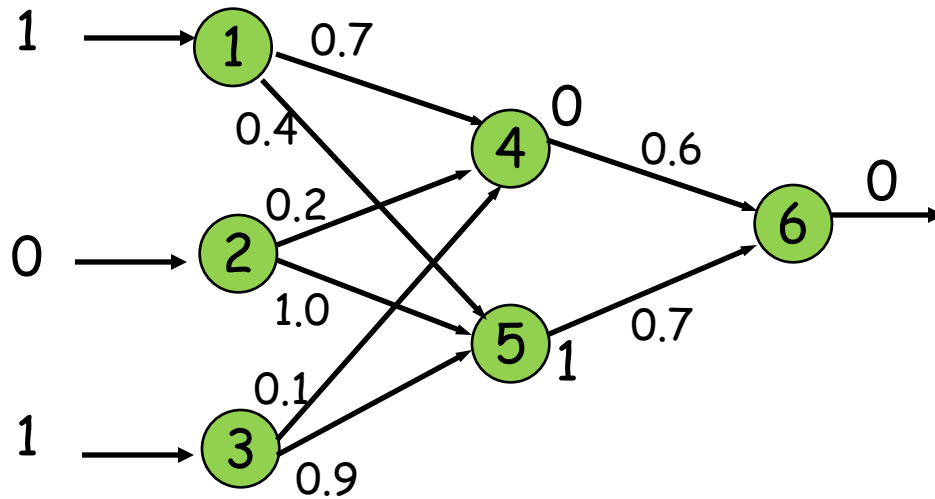


- Calcule la salida de la red

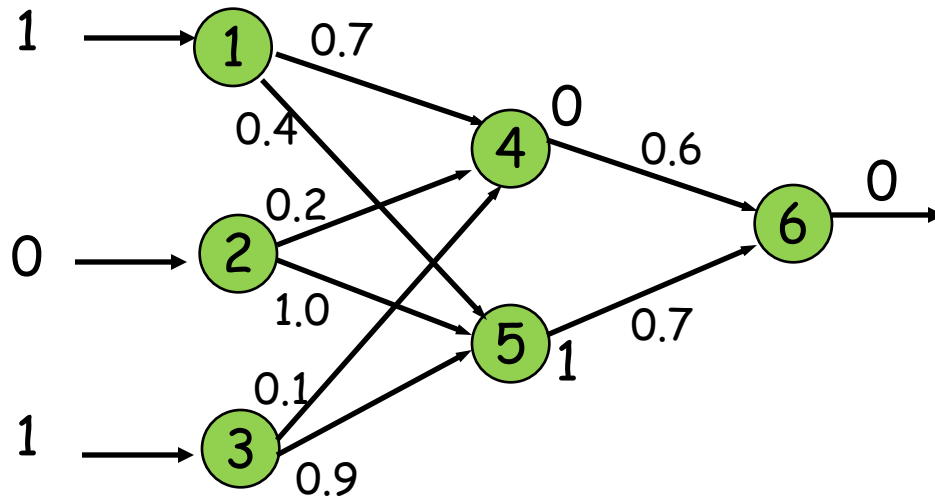
Entrada Neta para 6 = 0.7



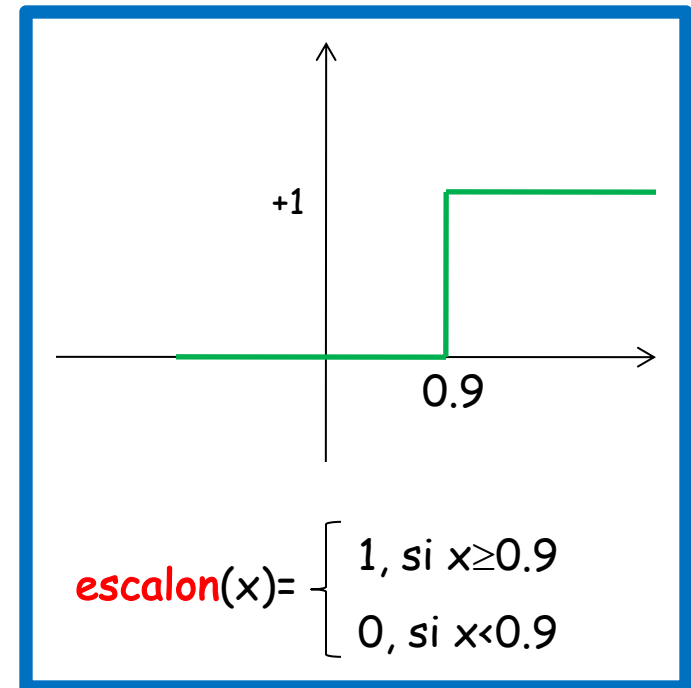
# Redes neuronales



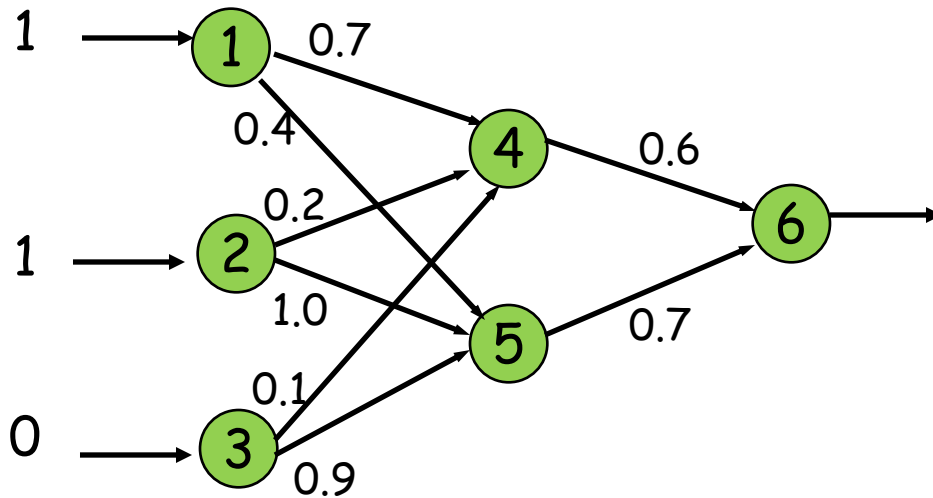
# Redes neuronales



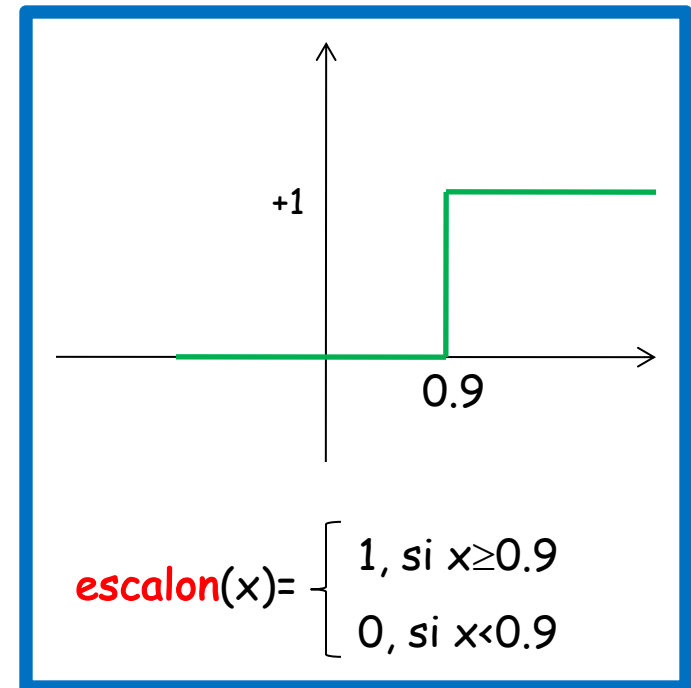
- Para la entrada (1,0,1), la salida de la red es 0



# Redes neuronales

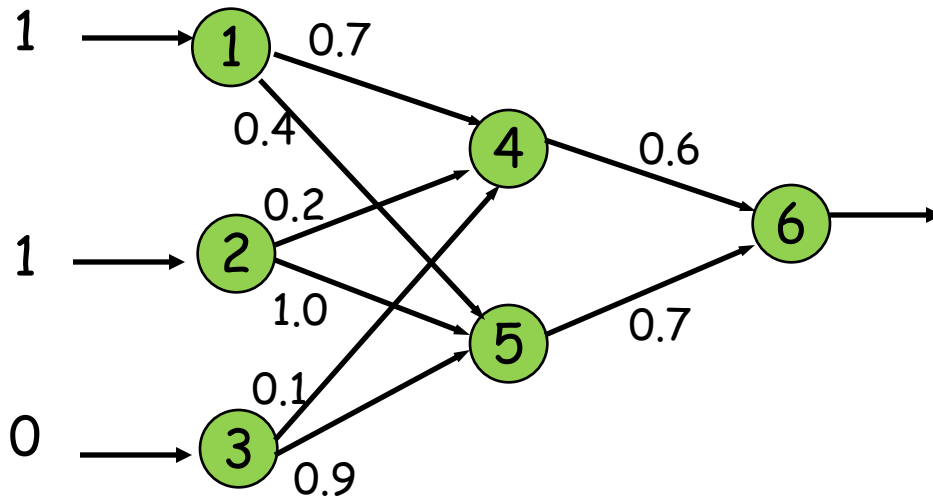


- Calcule la salida de la red para la entrada (1,1,0)

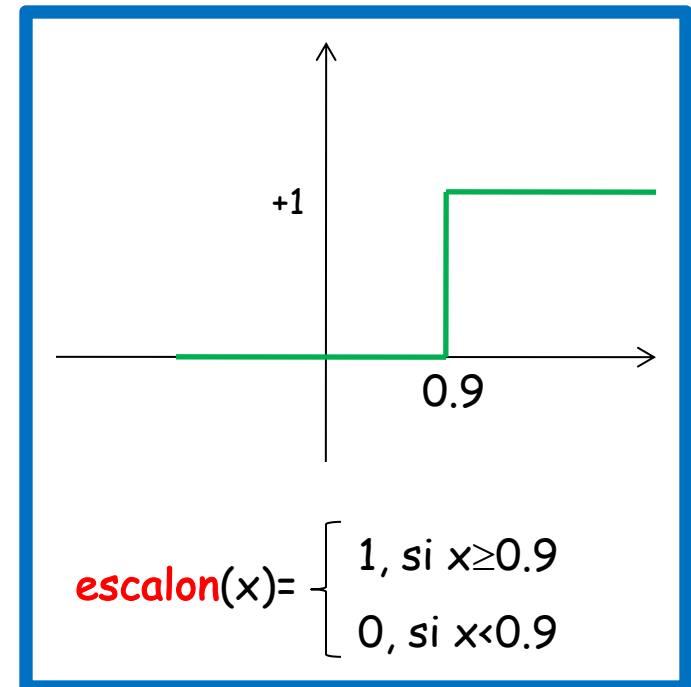




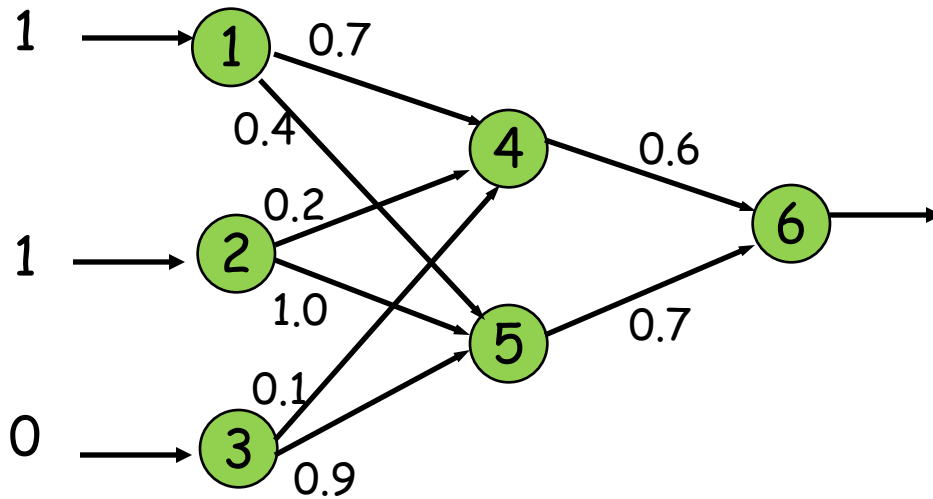
# Redes neuronales



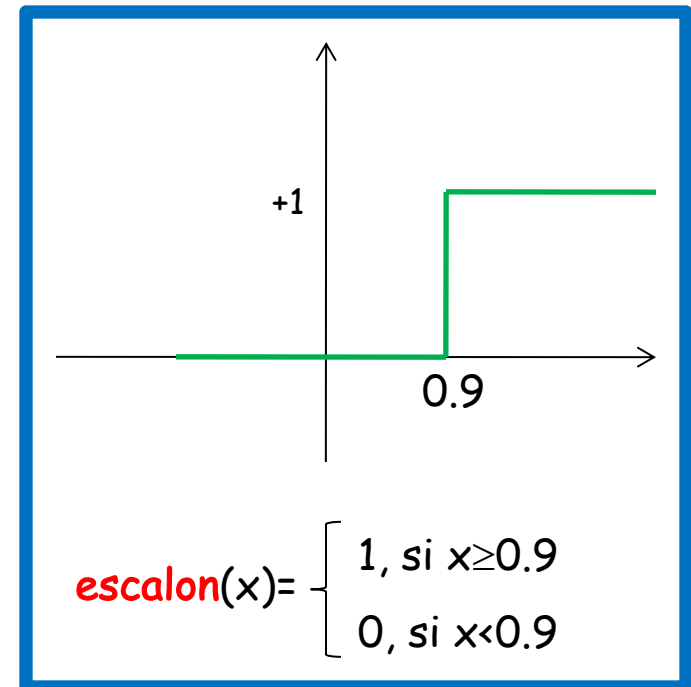
Neurona	Entrada neta	Salida
4		
5		
6		



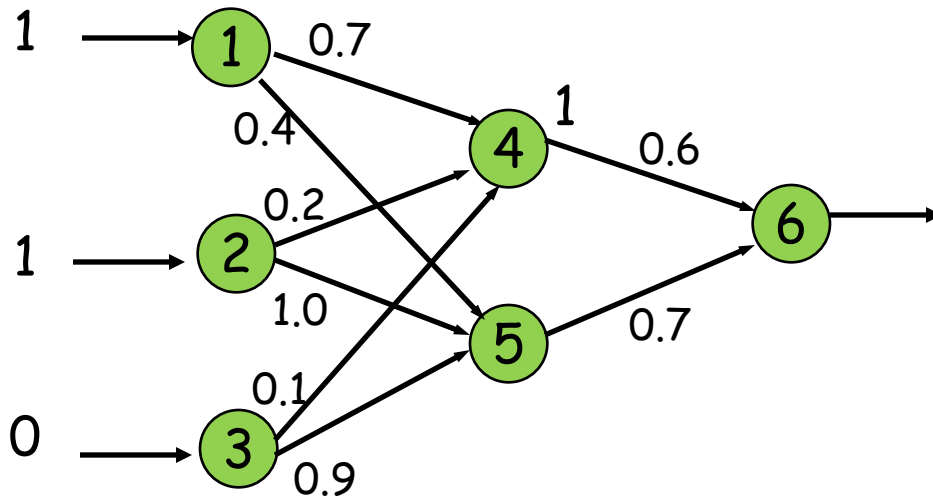
# Redes neuronales



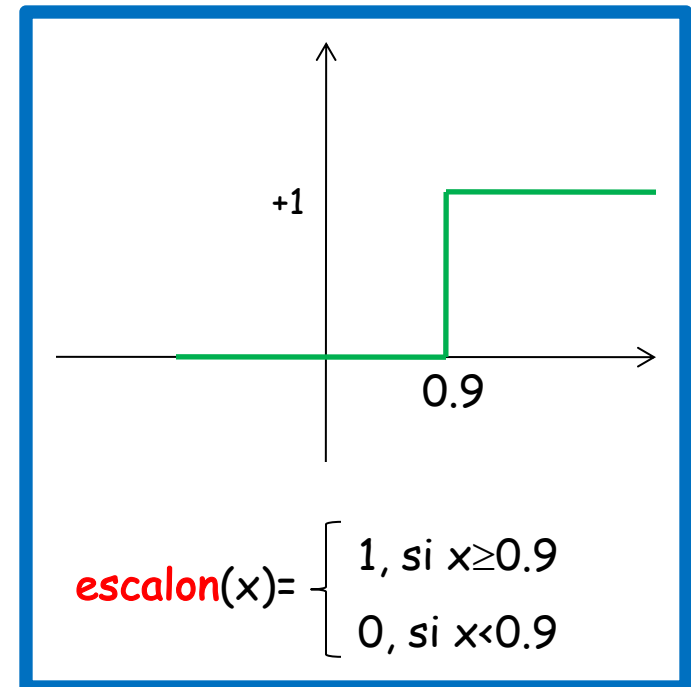
Neurona	Entrada neta	Salida
4	0.9	
5		
6		



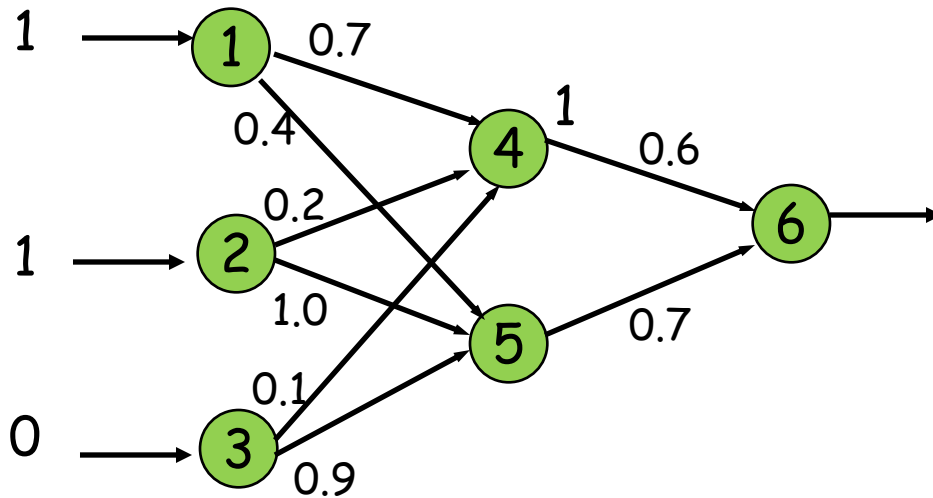
# Redes neuronales



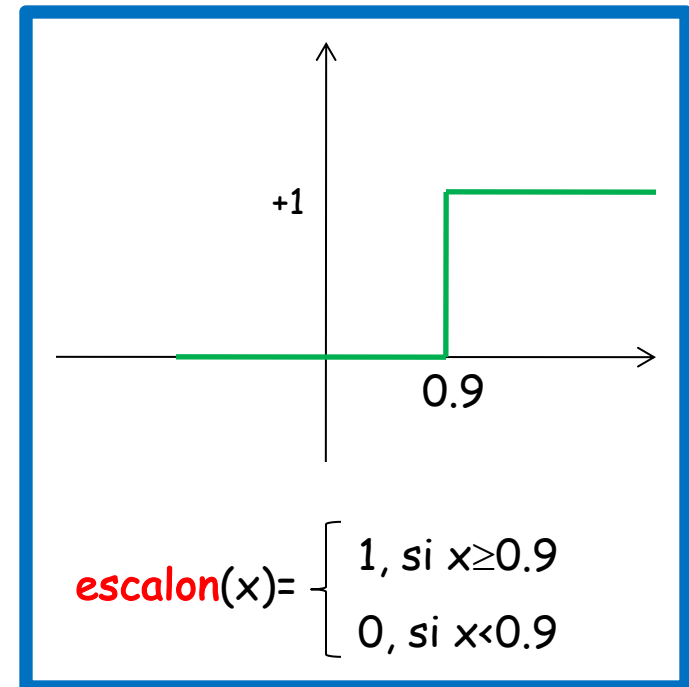
Neurona	Entrada neta	Salida
4	0.9	1
5		
6		



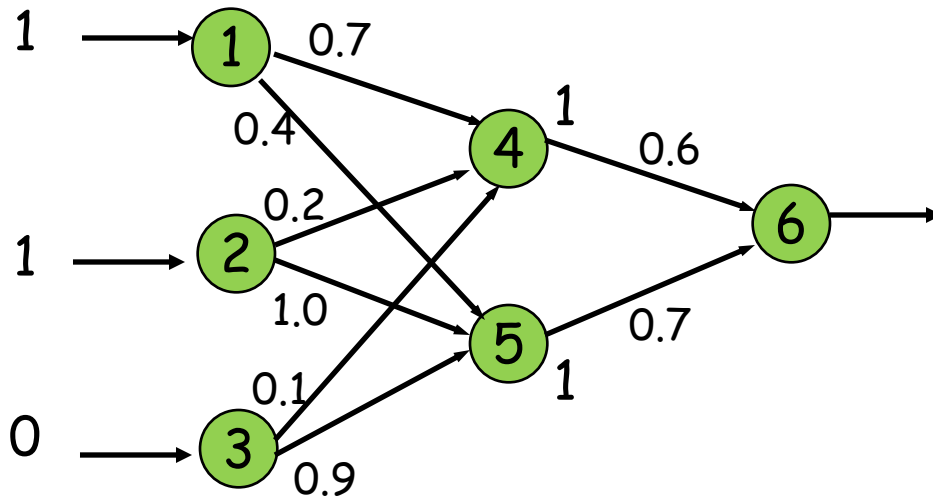
# Redes neuronales



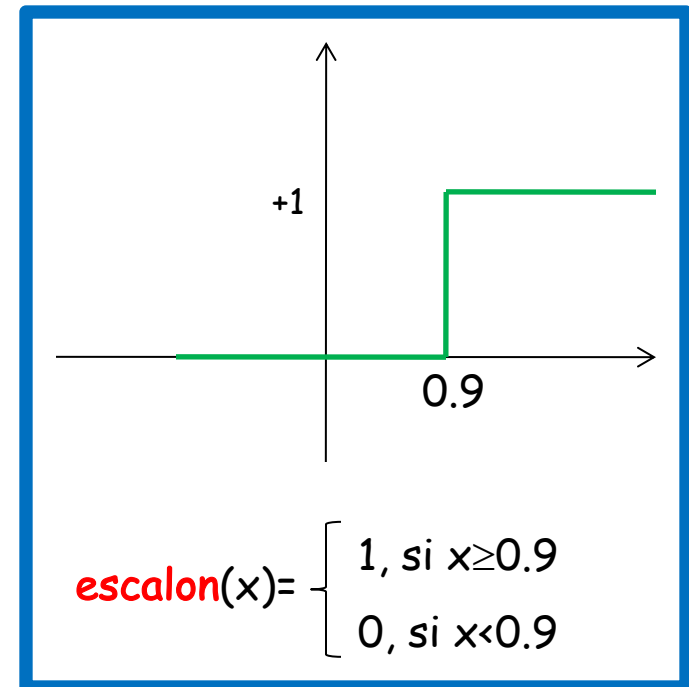
Neurona	Entrada neta	Salida
4	0.9	1
5	1.4	
6		



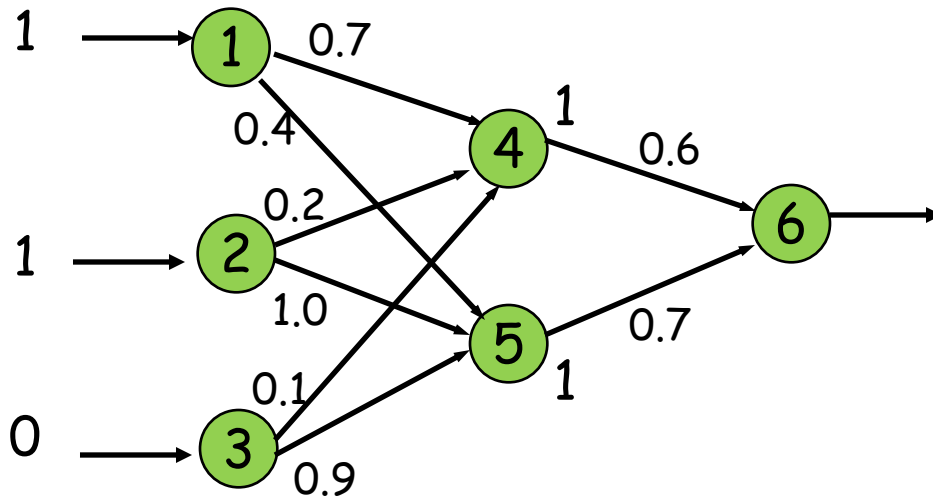
# Redes neuronales



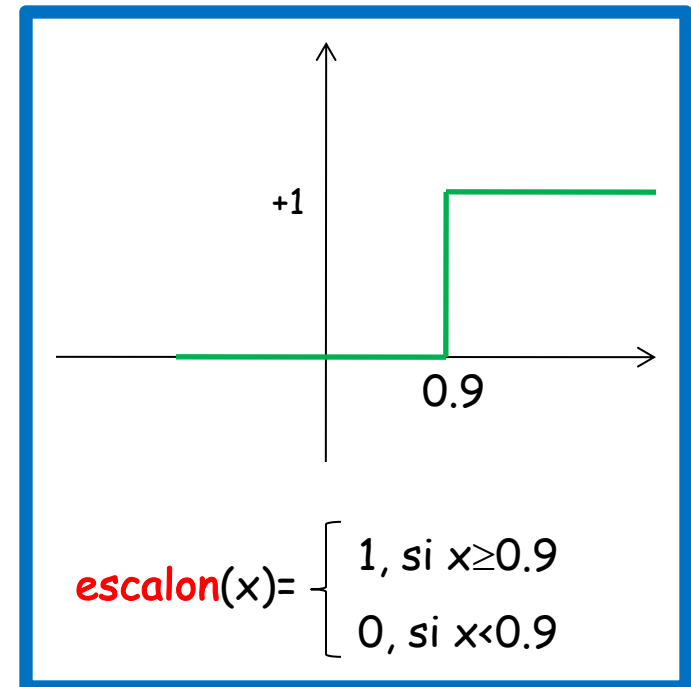
Neurona	Entrada neta	Salida
4	0.9	1
5	1.4	1
6		



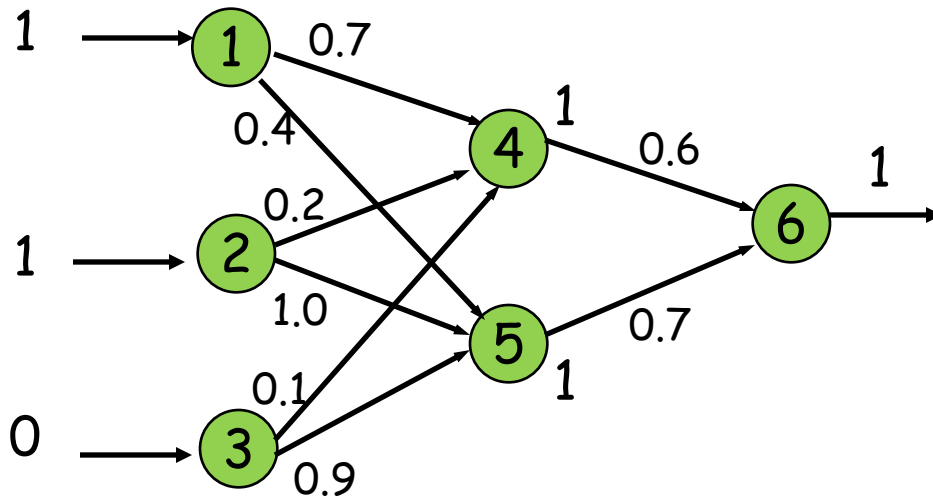
# Redes neuronales



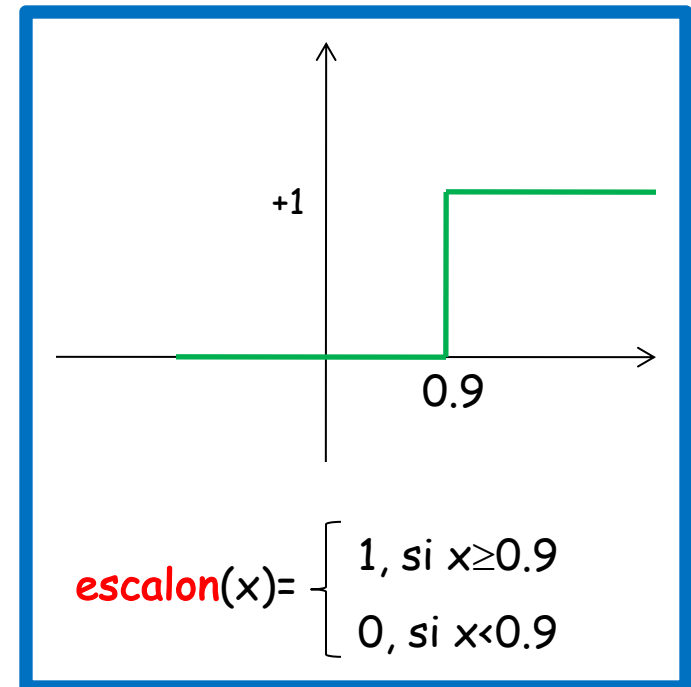
Neurona	Entrada neta	Salida
4	0.9	1
5	1.4	1
6	1.3	



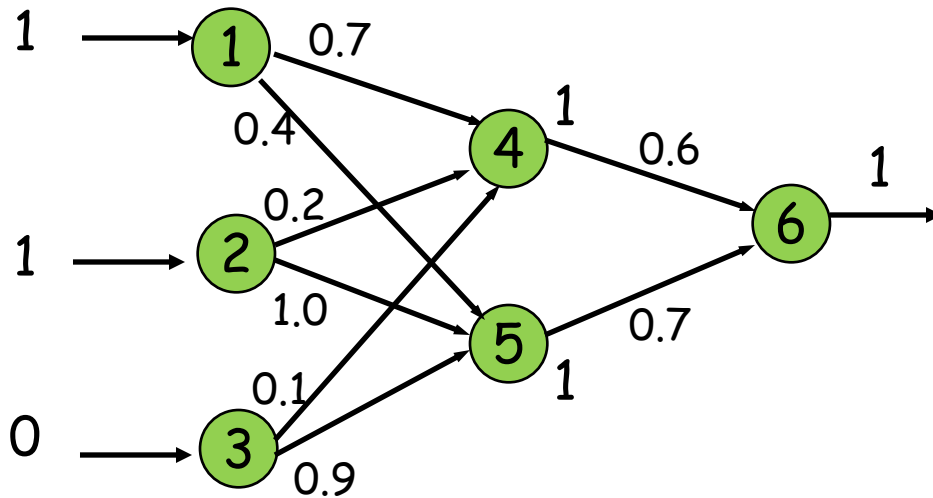
# Redes neuronales



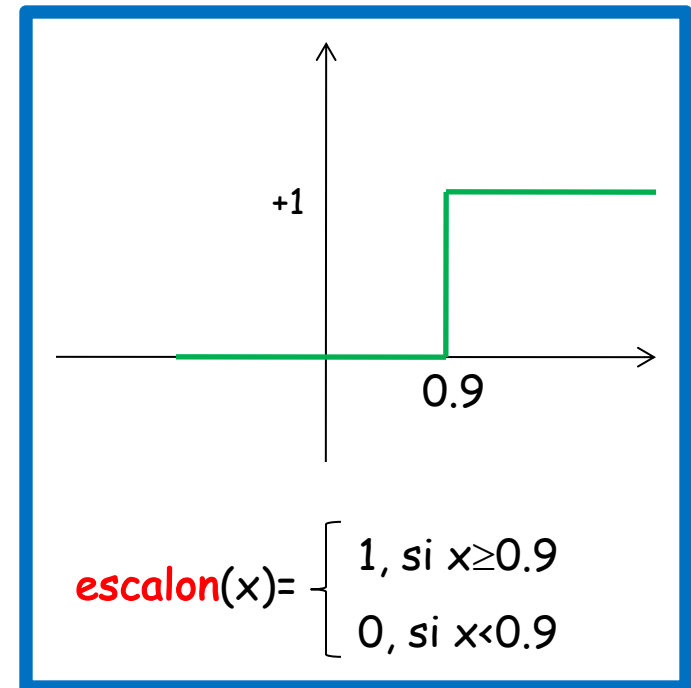
Neurona	Entrada neta	Salida
4	0.9	1
5	1.4	1
6	1.3	1



# Redes neuronales



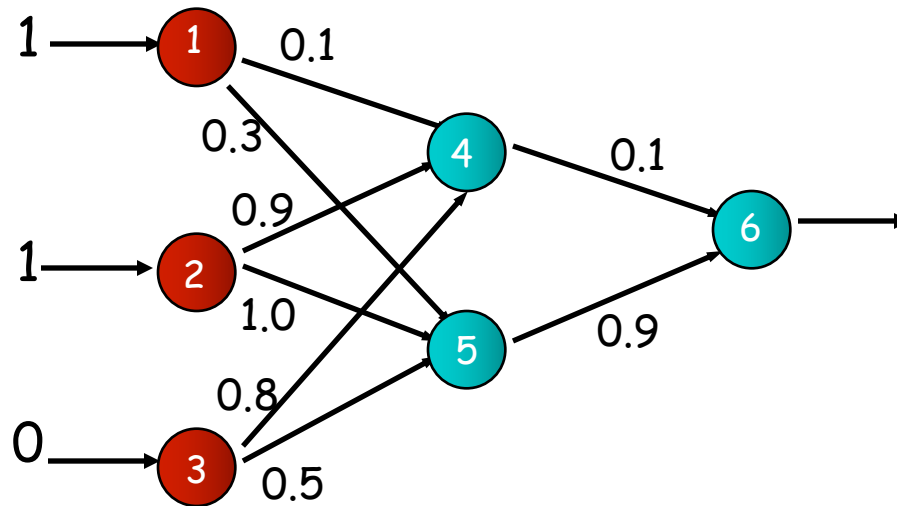
- Para la entrada (1,1,0), la salida de la red es 1





# Redes neuronales

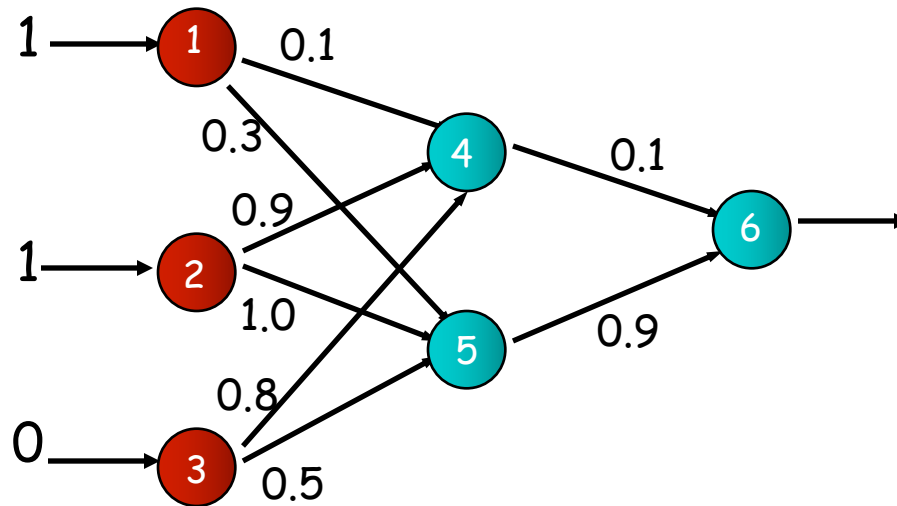
- Calcule la salida para las siguientes entradas usando la función escalón con  $t=0.6$



Neurona	Entrada neta	Salida
4		
5		
6		

# Redes neuronales

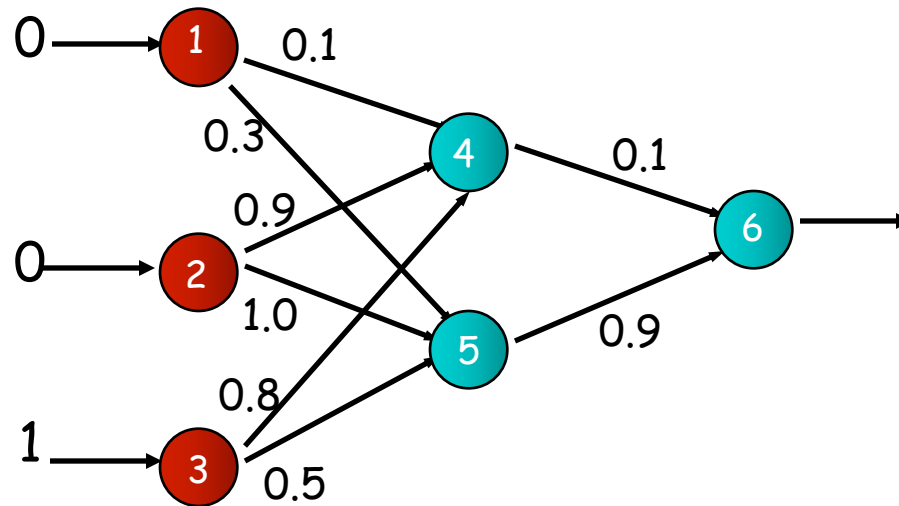
- Calcule la salida para las siguientes entradas usando la función escalón con  $t=0.6$



Neurona	Entrada neta	Salida
4	1.0	1
5	1.3	1
6	1.0	1

# Redes neuronales

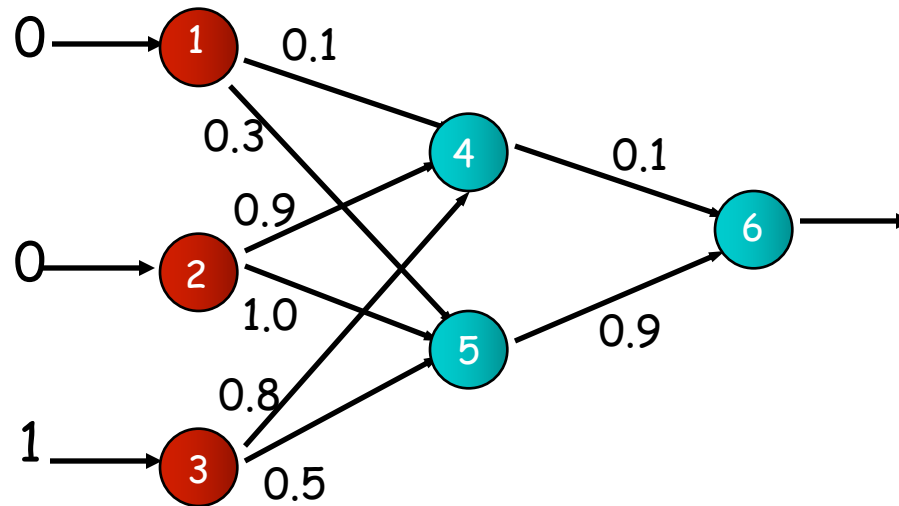
- Calcule la salida para las siguientes entradas usando la función escalón con  $t=0.6$



Neurona	Entrada neta	Salida
4		
5		
6		

# Redes neuronales

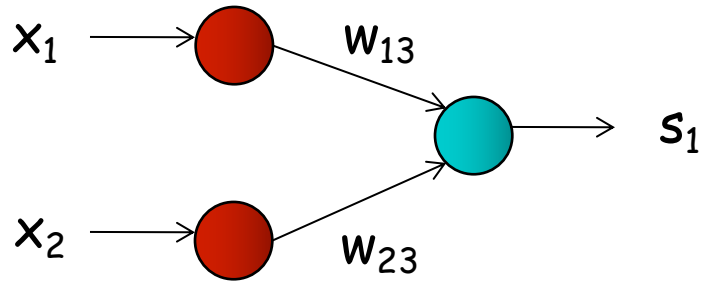
- Calcule la salida para las siguientes entradas usando la función escalón con  $t=0.6$



Neurona	Entrada neta	Salida
4	0.8	1
5	0.5	0
6	0.1	0

# Redes neuronales

## Redes y compuertas

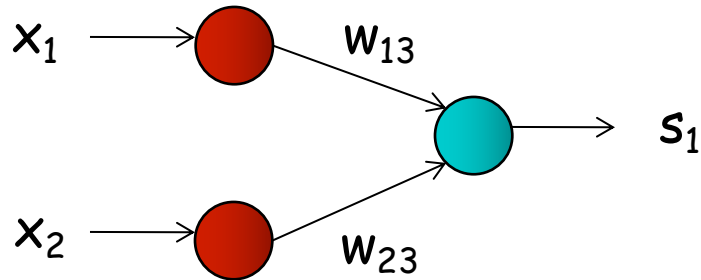


## Compuerta AND

$x_1$	$x_2$	$s_1$
1	1	1
1	0	0
0	1	0
0	0	0

# Redes neuronales

## Redes y compuertas



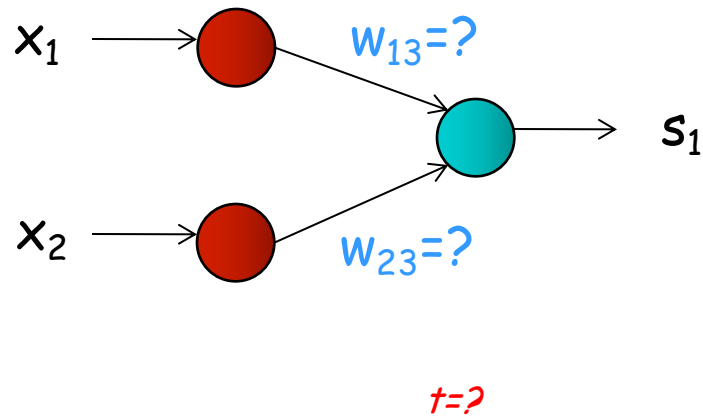
## Compuerta AND

$x_1$	$x_2$	$s_1$
1	1	1
1	0	0
0	1	0
0	0	0

conjunto de entrenamiento

# Redes neuronales

## Redes y compuertas

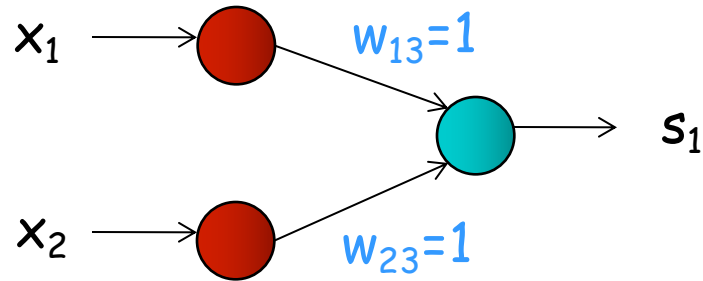


## Compuerta AND

$x_1$	$x_2$	$s_1$
1	1	1
1	0	0
0	1	0
0	0	0

# Redes neuronales

## Redes y compuertas



$t=?$

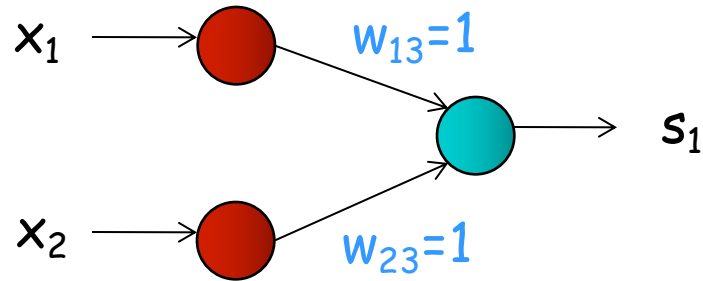
## Compuerta AND

$x_1$	$x_2$	$s_1$
1	1	1
1	0	0
0	1	0
0	0	0



# Redes neuronales

## Redes y compuertas



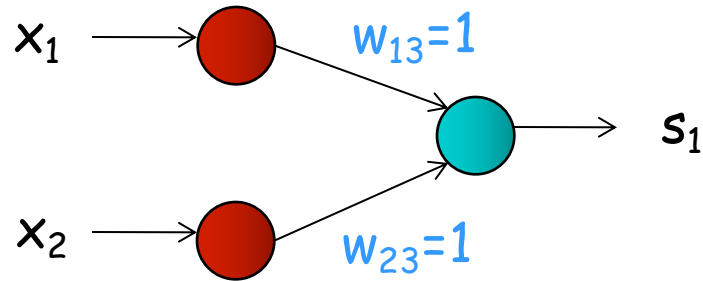
## Compuerta AND

$x_1$	$x_2$	$s_1$
1	1	1
1	0	0
0	1	0
0	0	0

$x_1$	$x_2$	Entrada neta
1	1	?
1	0	?
0	1	?
0	0	?

# Redes neuronales

## Redes y compuertas



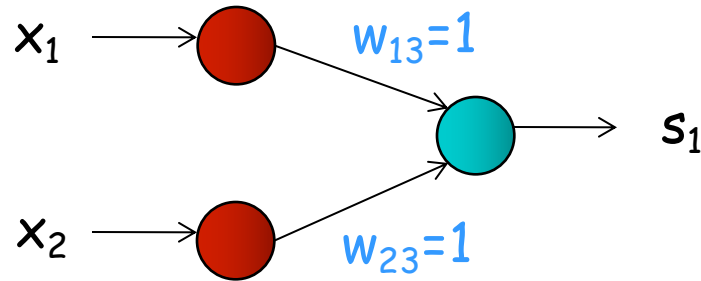
## Compuerta AND

$x_1$	$x_2$	$s_1$
1	1	1
1	0	0
0	1	0
0	0	0

$x_1$	$x_2$	Entrada neta
1	1	2
1	0	1
0	1	1
0	0	0

# Redes neuronales

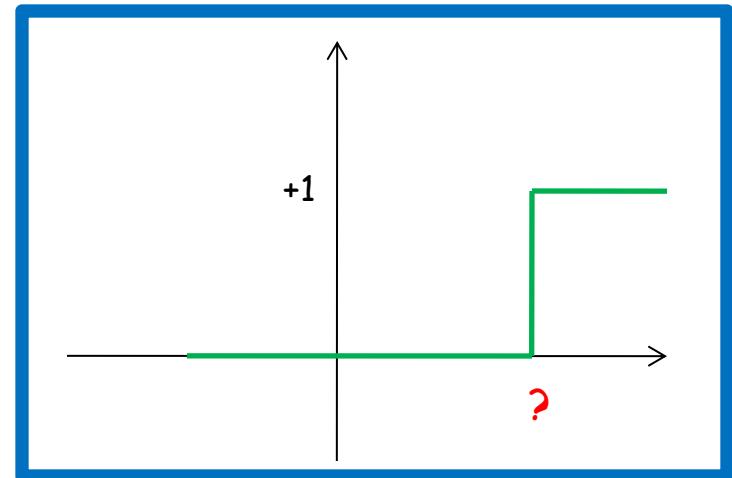
## Redes y compuertas



## Compuerta AND

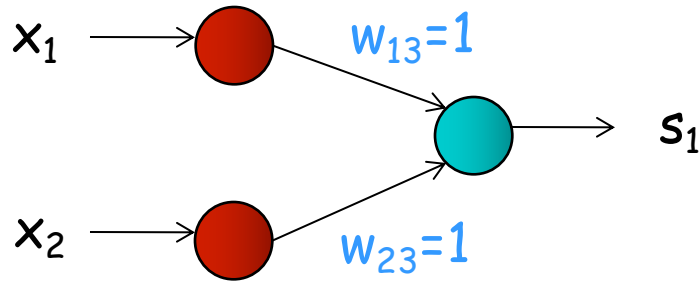
$x_1$	$x_2$	$s_1$
1	1	1
1	0	0
0	1	0
0	0	0

$x_1$	$x_2$	Entrada neta
1	1	2
1	0	1
0	1	1
0	0	0



# Redes neuronales

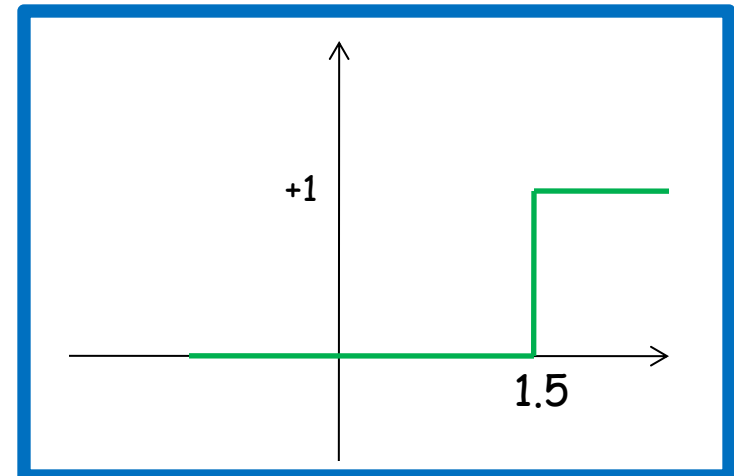
## Redes y compuertas



## Compuerta AND

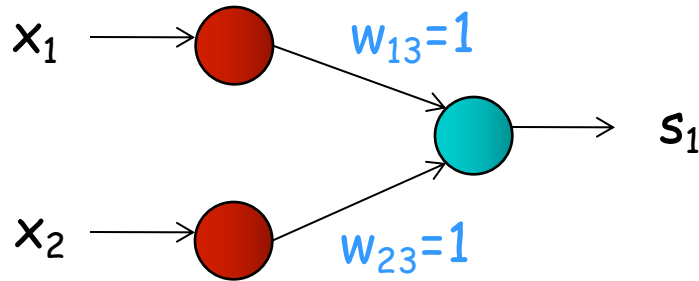
$x_1$	$x_2$	$s_1$
1	1	1
1	0	0
0	1	0
0	0	0

$x_1$	$x_2$	Entrada neta
1	1	2
1	0	1
0	1	1
0	0	0



# Redes neuronales

## Redes y compuertas



*$t=1.5$  para la  
función escalón*

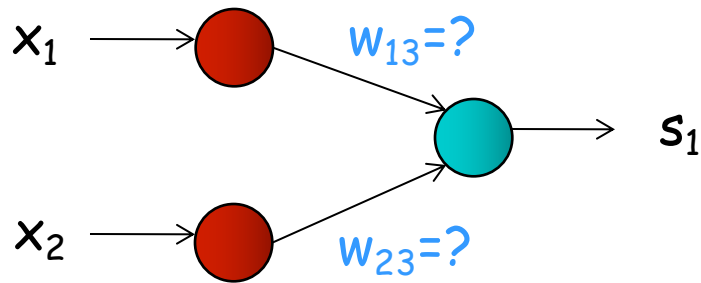
## Compuerta AND

$x_1$	$x_2$	$s_1$
1	1	1
1	0	0
0	1	0
0	0	0

Red neuronal que representa  
la compuerta AND

# Redes neuronales

¿Cómo representar la compuerta or?

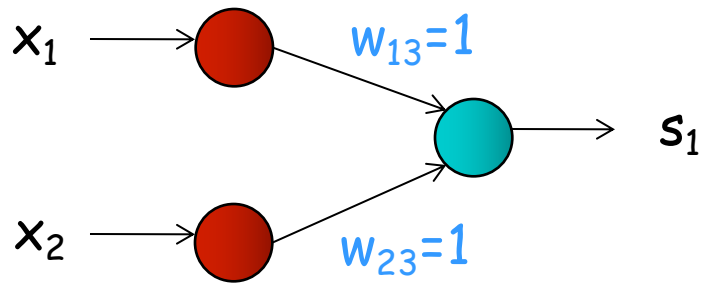


$t=?$

$x_1$	$x_2$	$s_1$
1	1	1
1	0	1
0	1	1
0	0	0

# Redes neuronales

¿Cómo representar la compuerta or?

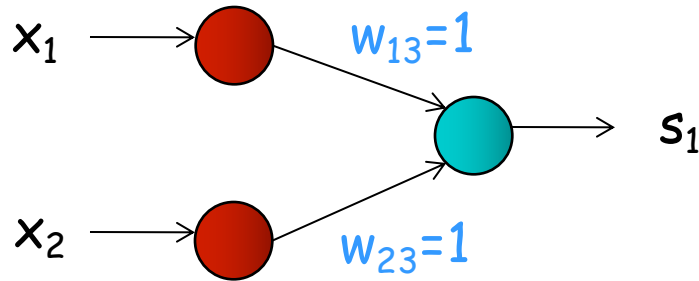


$t=?$

$x_1$	$x_2$	$s_1$
1	1	1
1	0	1
0	1	1
0	0	0

# Redes neuronales

¿Cómo representar la compuerta or?



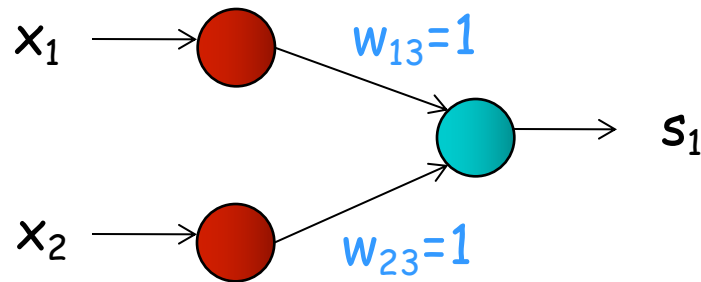
$x_1$	$x_2$	$s_1$
1	1	1
1	0	1
0	1	1
0	0	0

$x_1$	$x_2$	Entrada neta
1	1	2
1	0	1
0	1	1
0	0	0



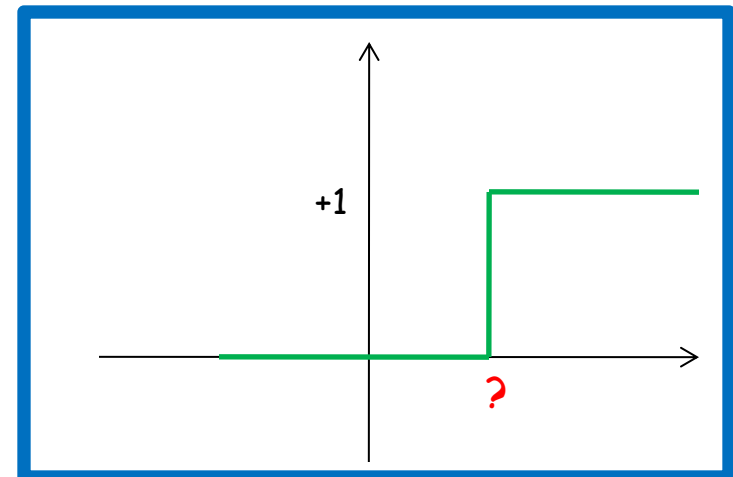
# Redes neuronales

¿Cómo representar la compuerta or?



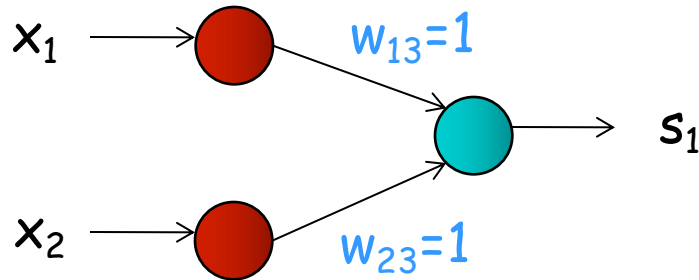
$x_1$	$x_2$	$s_1$
1	1	1
1	0	1
0	1	1
0	0	0

$x_1$	$x_2$	Entrada neta
1	1	2
1	0	1
0	1	1
0	0	0



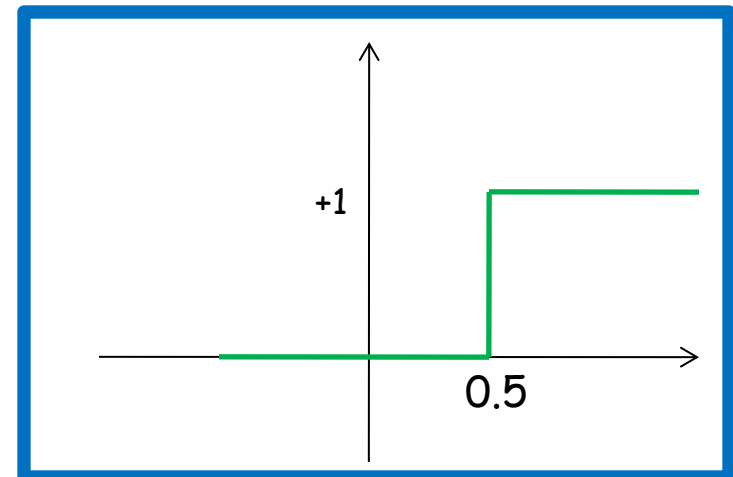
# Redes neuronales

¿Cómo representar la compuerta or?



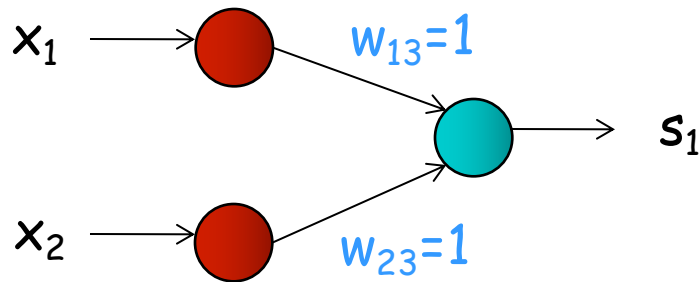
$x_1$	$x_2$	$s_1$
1	1	1
1	0	1
0	1	1
0	0	0

$x_1$	$x_2$	Entrada neta
1	1	2
1	0	1
0	1	1
0	0	0



# Redes neuronales

¿Cómo representar la compuerta or?



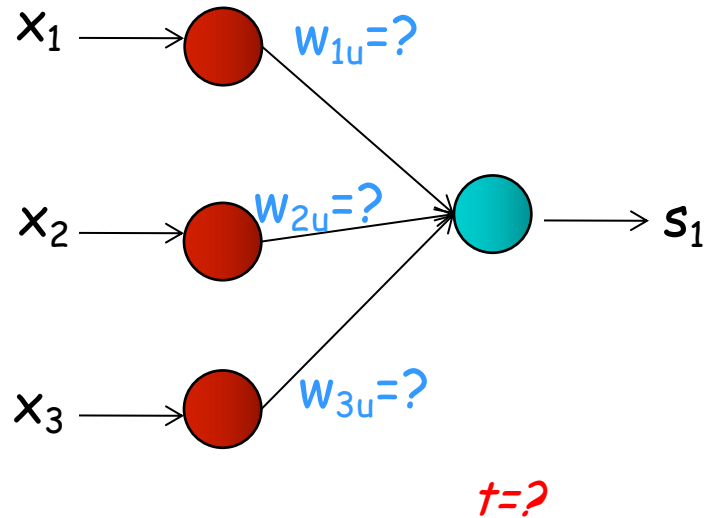
$x_1$	$x_2$	$s_1$
1	1	1
1	0	1
0	1	1
0	0	0

$t=0.5$

Red neuronal que representa  
la compuerta OR

# Redes neuronales

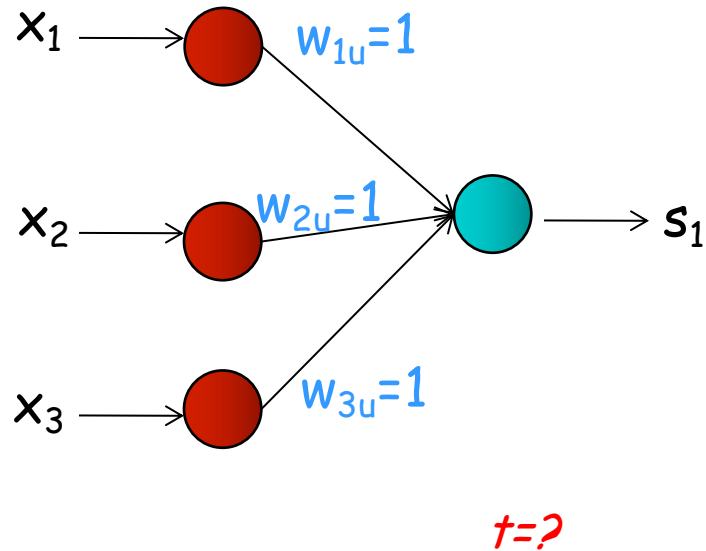
Función de mayoría: para  $n=3$



$x_1$	$x_2$	$x_3$	$s_1$
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	0
0	0	0	0

# Redes neuronales

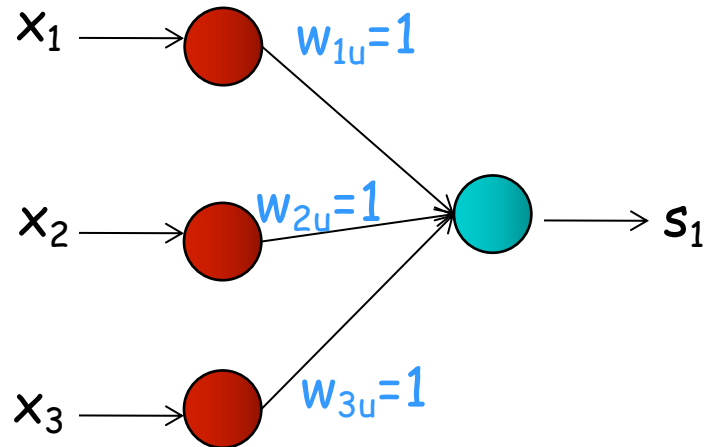
Función de mayoría: para  $n=3$



$x_1$	$x_2$	$x_3$	$s_1$
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	0
0	0	0	0

# Redes neuronales

Función de mayoría: para  $n=3$



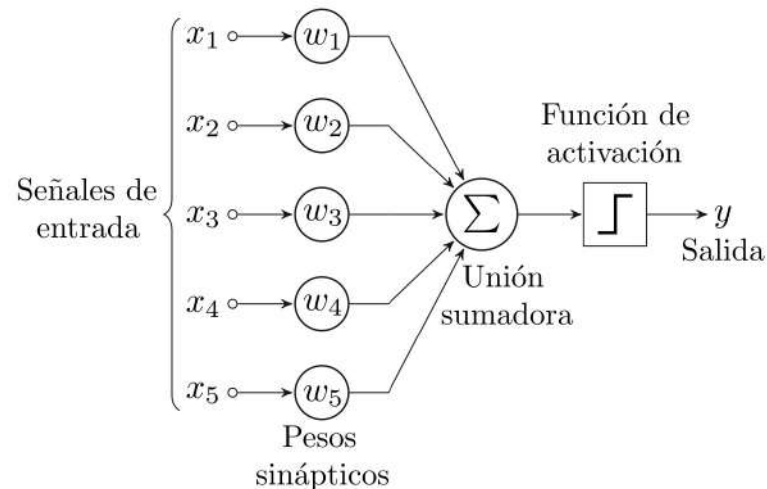
$t=1.5$

$x_1$	$x_2$	$x_3$	$s_1$
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	0
0	0	0	0

# Redes neuronales

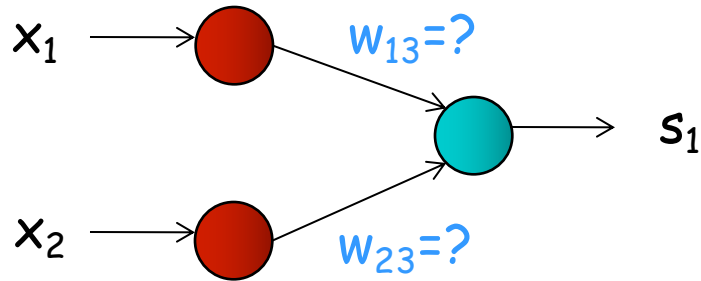
## Red perceptrón

Son redes neuronales que solo tienen capa de entrada y capa de salida, es decir, no tienen capas intermedias u ocultas



# Redes neuronales

¿Cómo representar la compuerta xor?



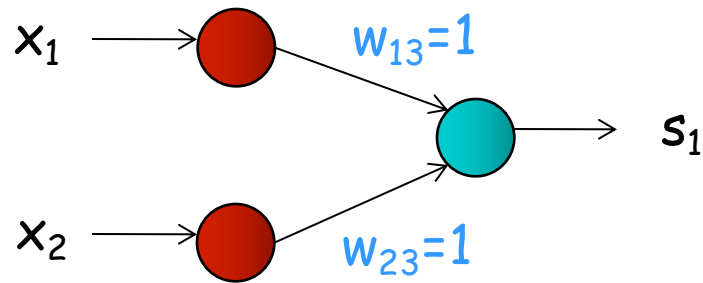
$t=?$

$x_1$	$x_2$	$s_1$
1	1	0
1	0	1
0	1	1
0	0	0



# Redes neuronales

¿Cómo representar la compuerta xor?

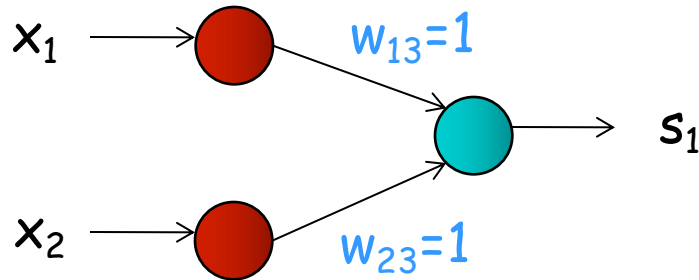


$t=?$

$x_1$	$x_2$	$s_1$
1	1	0
1	0	1
0	1	1
0	0	0

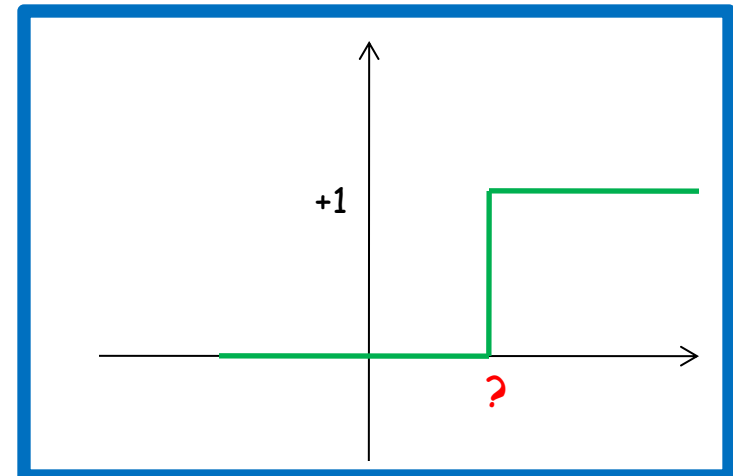
# Redes neuronales

¿Cómo representar la compuerta xor?



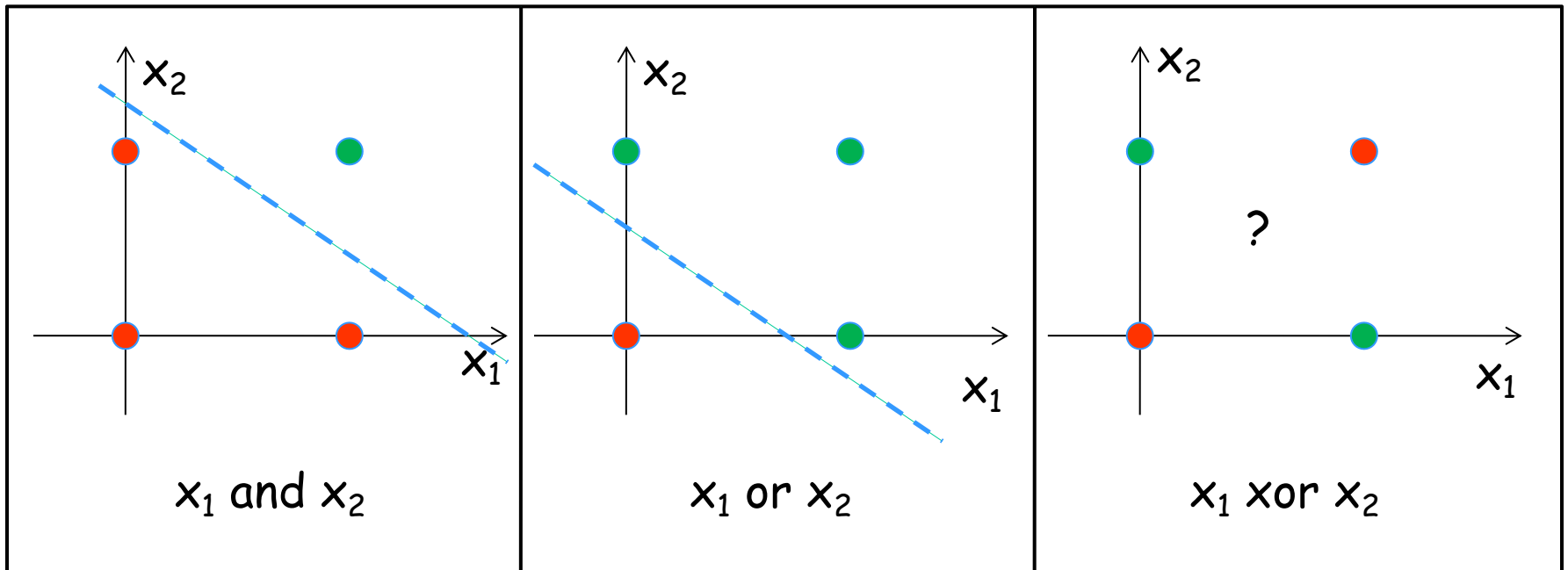
$x_1$	$x_2$	$s_1$
1	1	0
1	0	1
0	1	1
0	0	0

$x_1$	$x_2$	Entrada neta
1	1	2
1	0	1
0	1	1
0	0	0



# Redes neuronales

La red perceptrón es capaz de representar una función solo si existe una línea que separe los patrones de dos clases dadas

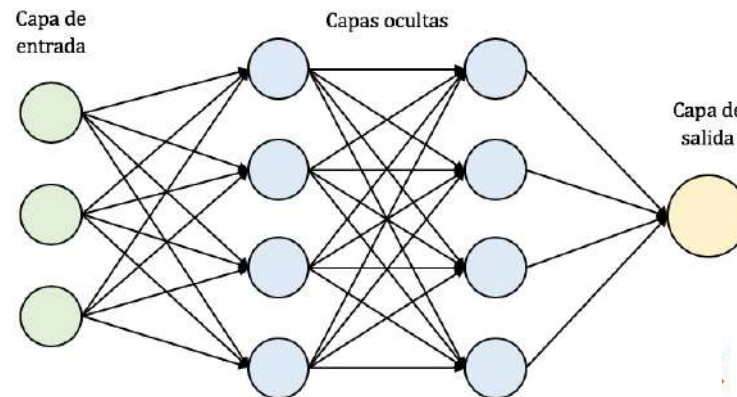


# Redes neuronales

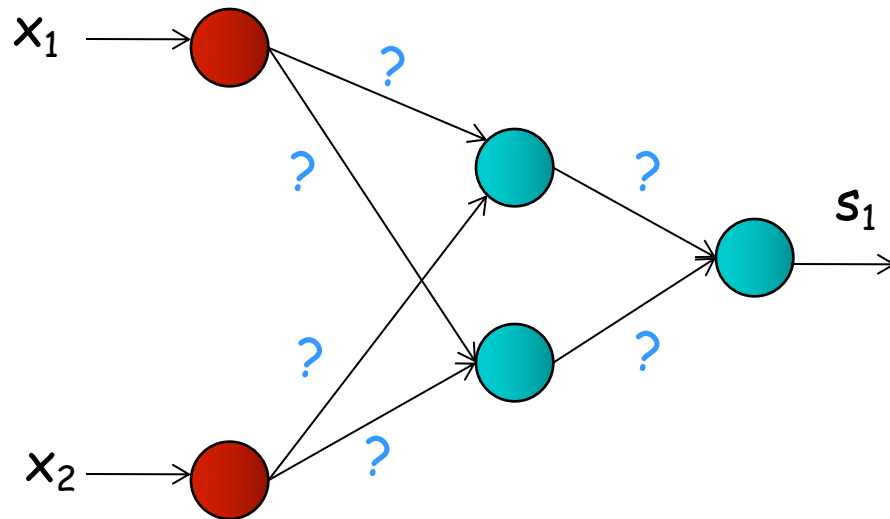
---

## Red perceptrón multicapa

Son redes neuronales que tienen una o varias capas ocultas y que permiten modelar más problemas que las redes perceptrón simples



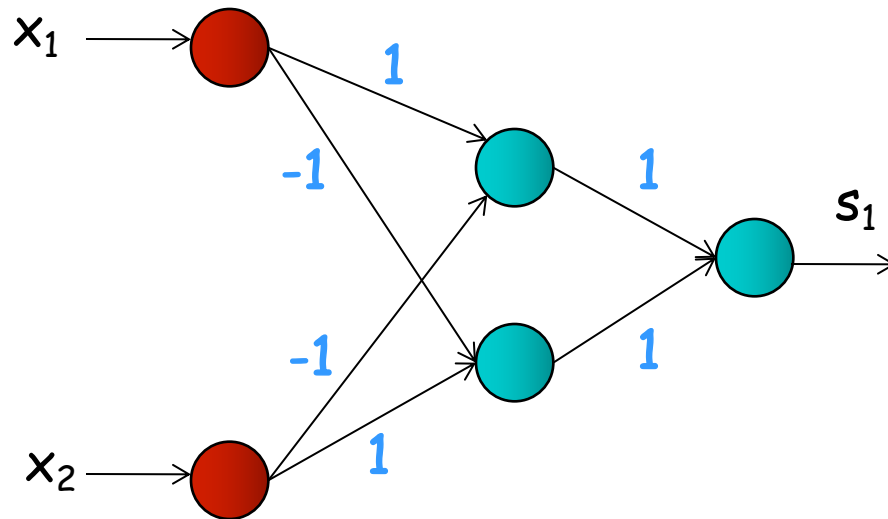
# Redes neuronales



$x_1$	$x_2$	$s_1$
1	1	0
1	0	1
0	1	1
0	0	0

$t=?$

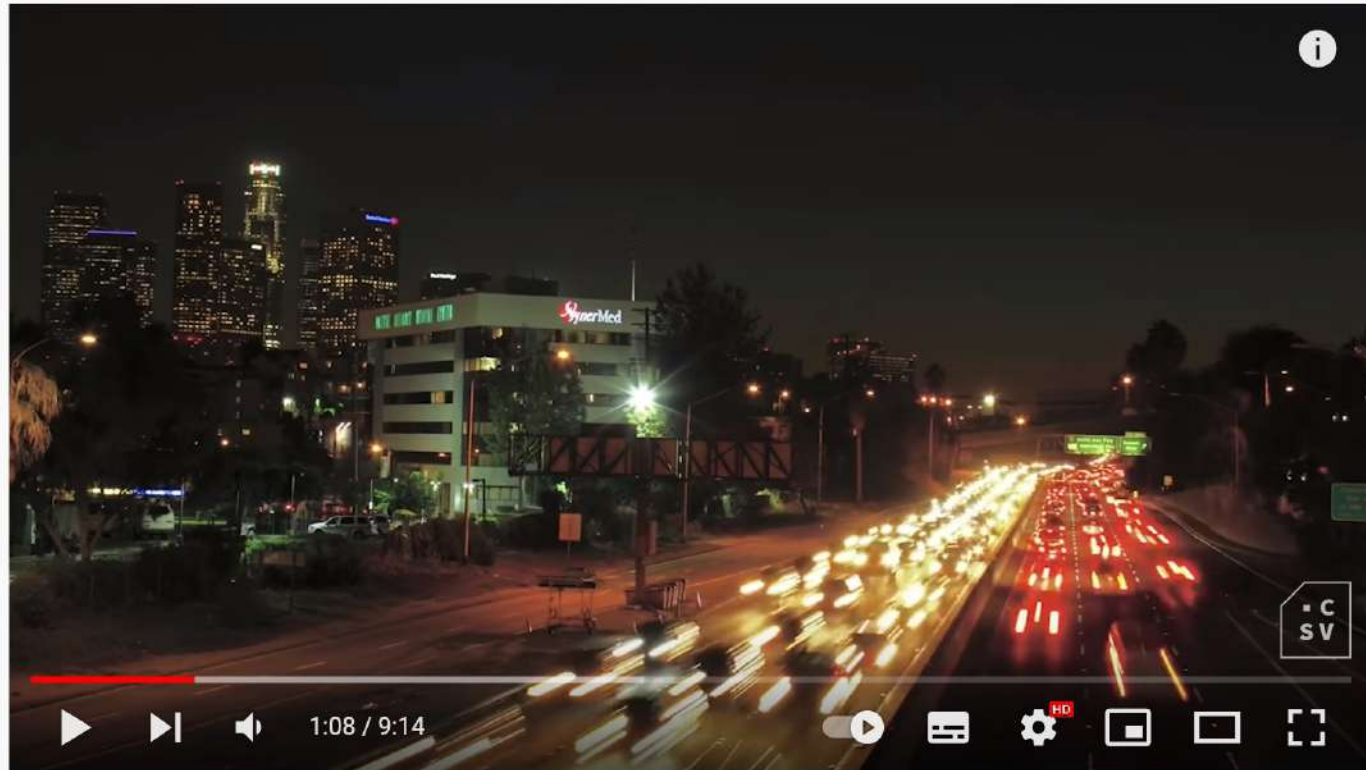
# Redes neuronales



*$t=1.0$*

$x_1$	$x_2$	$s_1$
1	1	0
1	0	1
0	1	1
0	0	0

# Redes neuronales



#Scenio

¿Qué es una Red Neuronal? Parte 1 : La Neurona | DotCSV

<https://www.youtube.com/watch?v=MRIV2IwFTPg>

# Redes neuronales

---

## Codificación de variables categóricas

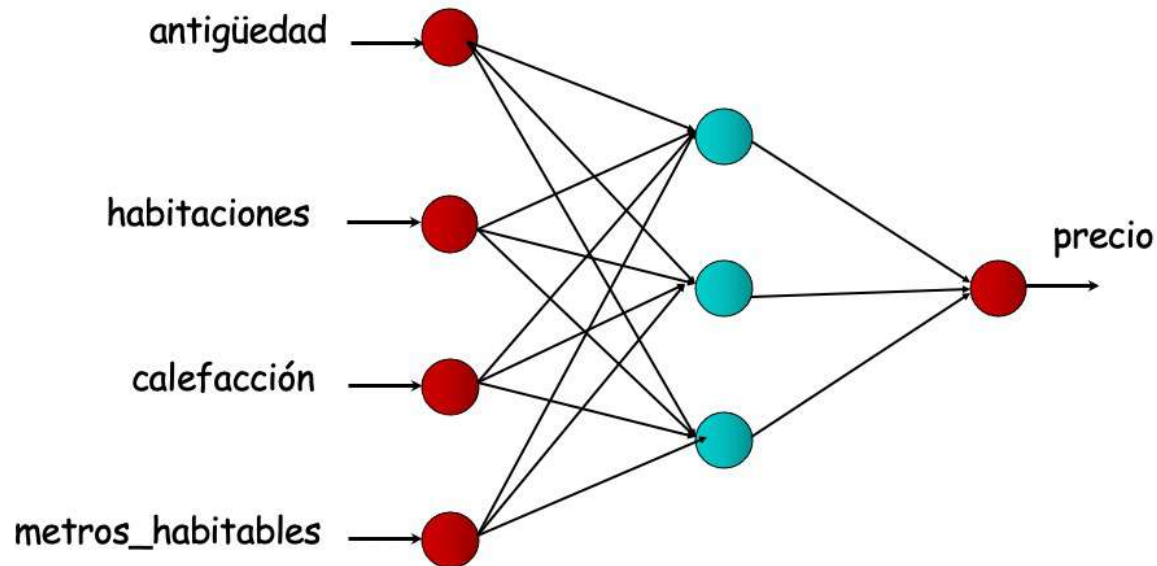
Las redes neuronales solo manejan entradas y salidas de tipo numérico. Si los datos contienen valores categóricos, éstos se deben transformar usando alguna estrategia

- *Label encoding*
- *One hot encoding*



# Redes neuronales

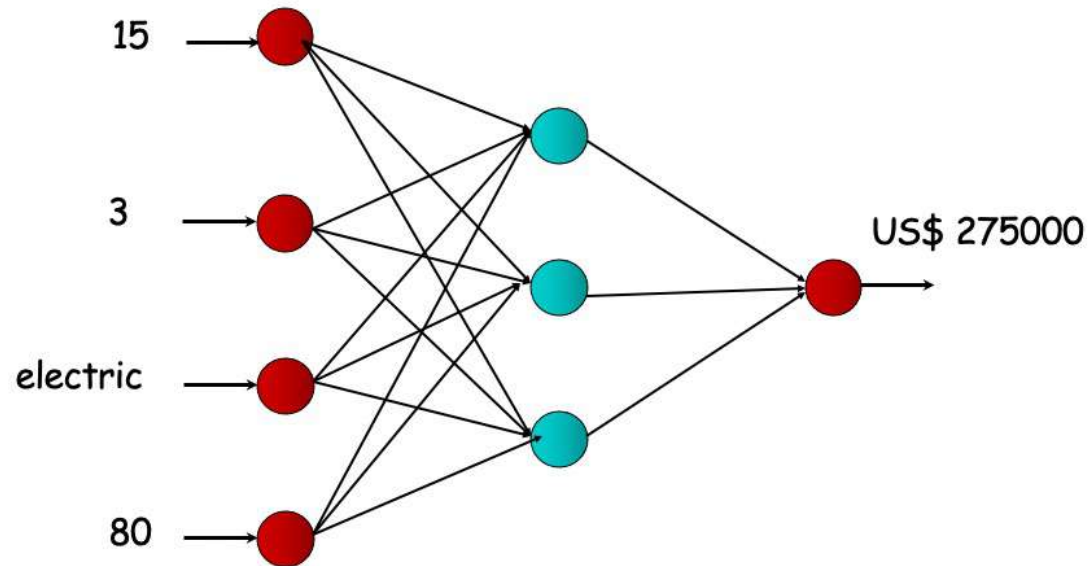
## Codificación de variables categóricas



antigüedad, habitaciones, y metros\_habitables, son atributos numéricos  
calefacción es una variable categórica que puede tomar los valores: *steam*,  
*electric*, o *hot air*

# Redes neuronales

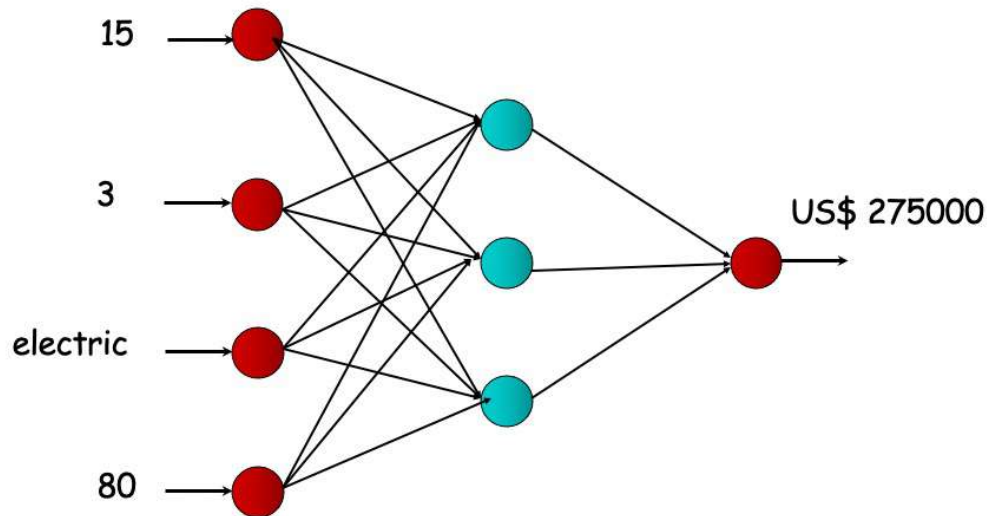
## Codificación de variables categóricas



antigüedad, habitaciones, y metros\_habitables, son atributos numéricos  
calefacción es una **variable categórica** que puede tomar los valores: *steam*,  
*electric*, o *hot air*

# Redes neuronales

## Label encoding

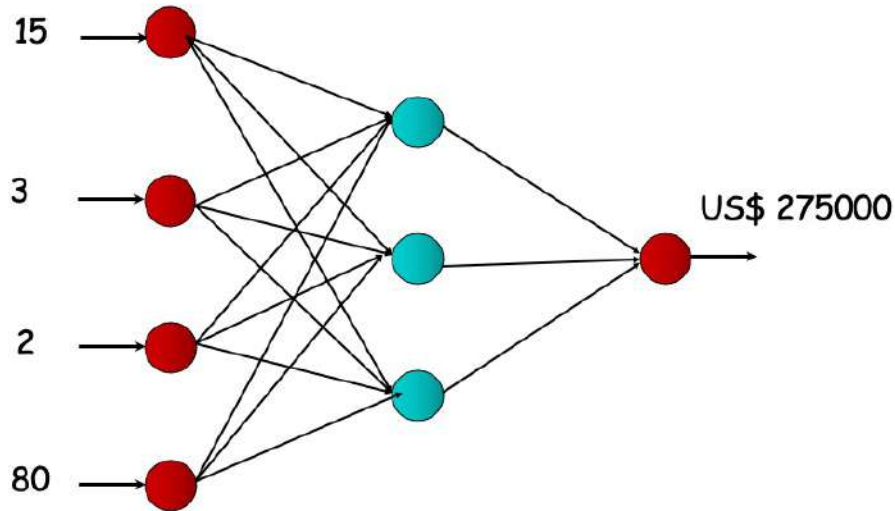


Usando label encoding un atributo categórico se reemplaza por un número según los valores que puede tomar dicha variable

calefacción	Label encoding
<i>steam</i>	1
<i>electric</i>	2
<i>hot air</i>	3

# Redes neuronales

## Label encoding

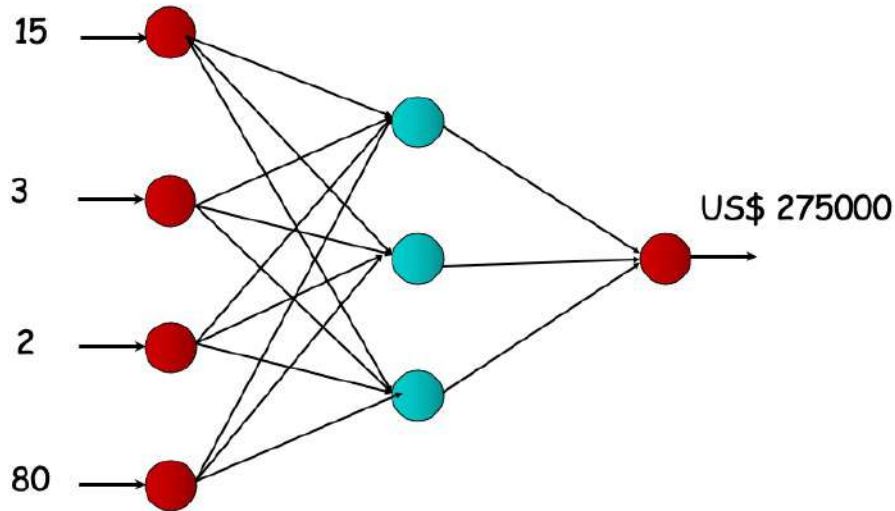


Usando label encoding un atributo categórico se reemplaza por un número según los valores que puede tomar dicha variable

calefacción	Label encoding
<i>steam</i>	1
<i>electric</i>	2
<i>hot air</i>	3

# Redes neuronales

## Label encoding



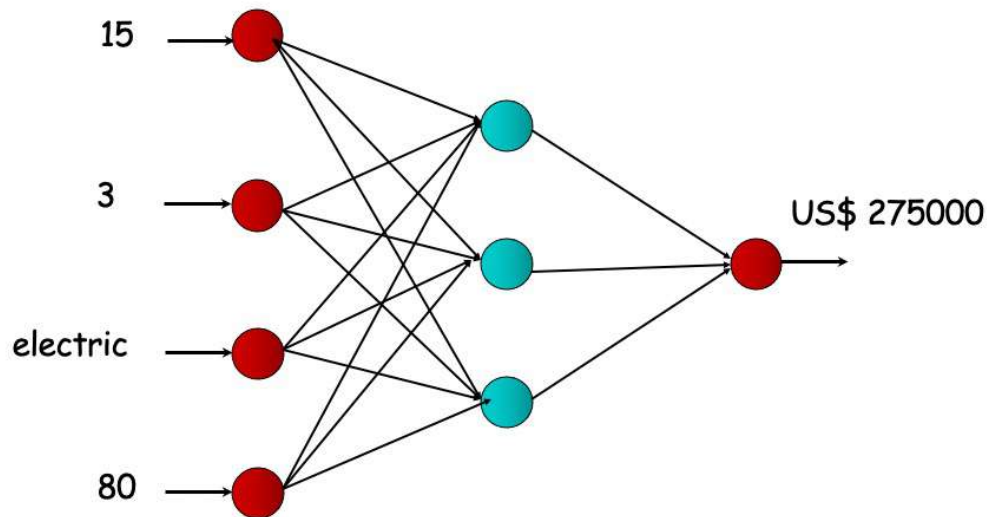
Usando label encoding un atributo categórico se reemplaza por un número según los valores que puede tomar dicha variable

calefacción	Label encoding
<i>steam</i>	1
<i>electric</i>	2
<i>hot air</i>	3

Los valores numéricos pueden influir en la construcción del modelo sin tener una interpretación matemática

# Redes neuronales

## One hot encoding

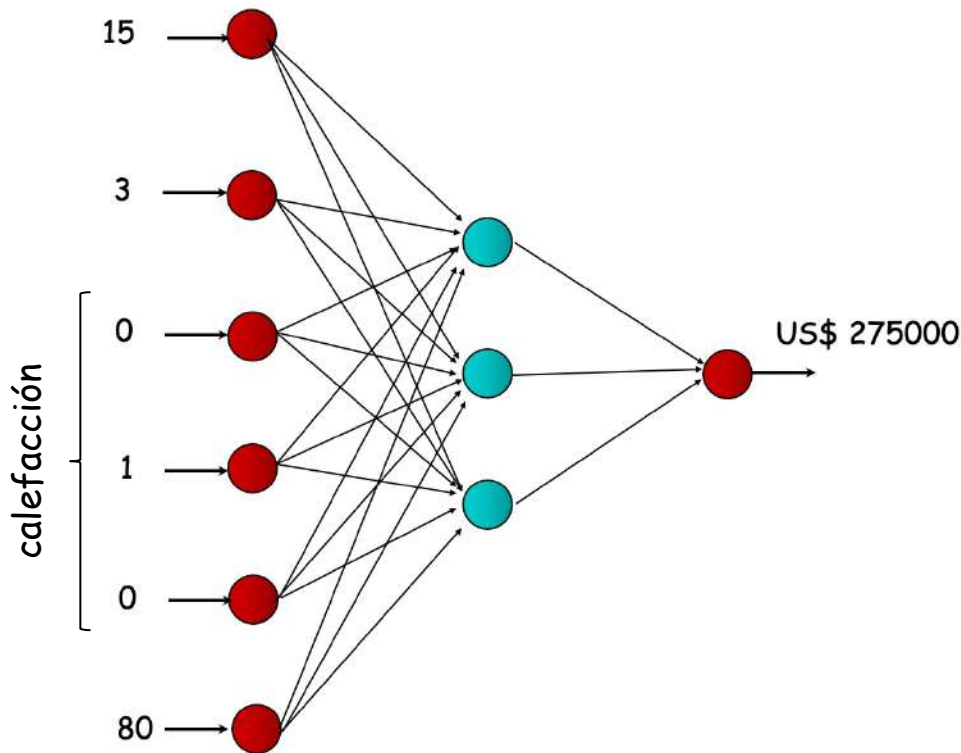


Con one hot encoding se crea una columna para cada valor distinto que exista en un atributo categórico

calefacción	One hot encoding		
<i>steam</i>	1	0	0
<i>electric</i>	0	1	0
<i>hot air</i>	0	0	1

# Redes neuronales

## One hot encoding

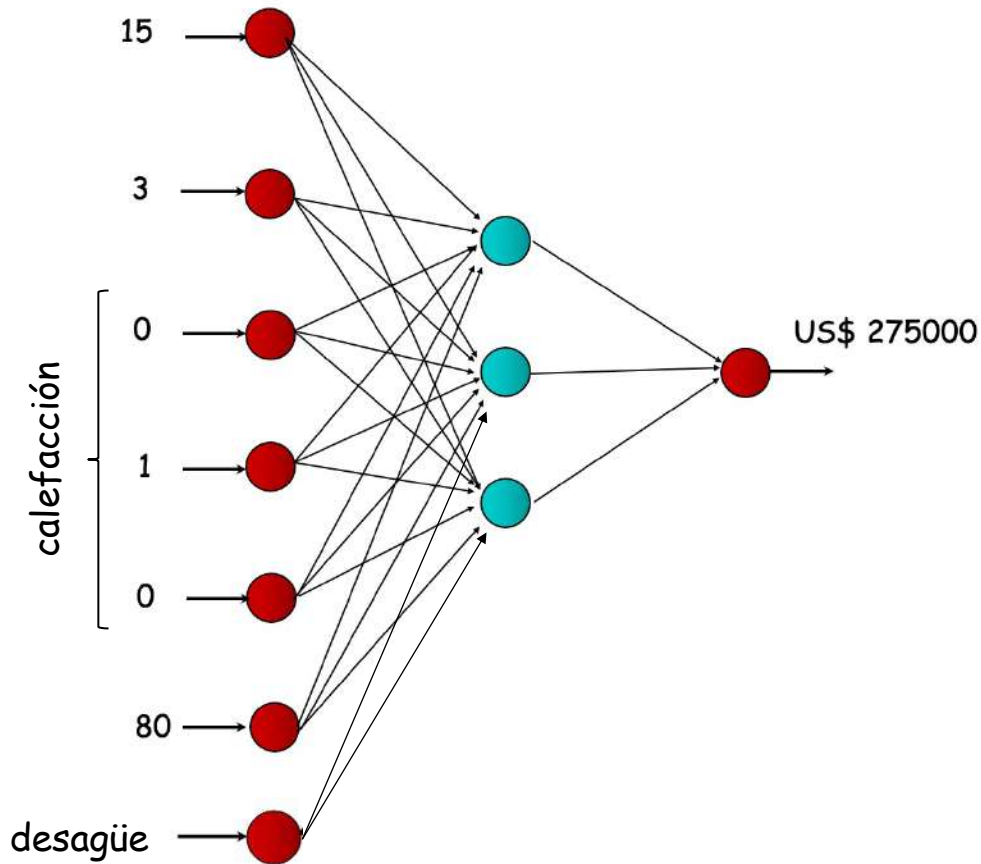


Con one hot encoding se crea una columna para cada valor distinto que exista en un atributo categórico

calefacción	One hot encoding		
<i>steam</i>	1	0	0
<i>electric</i>	0	1	0
<i>hot air</i>	0	0	1

# Redes neuronales

## One hot encoding



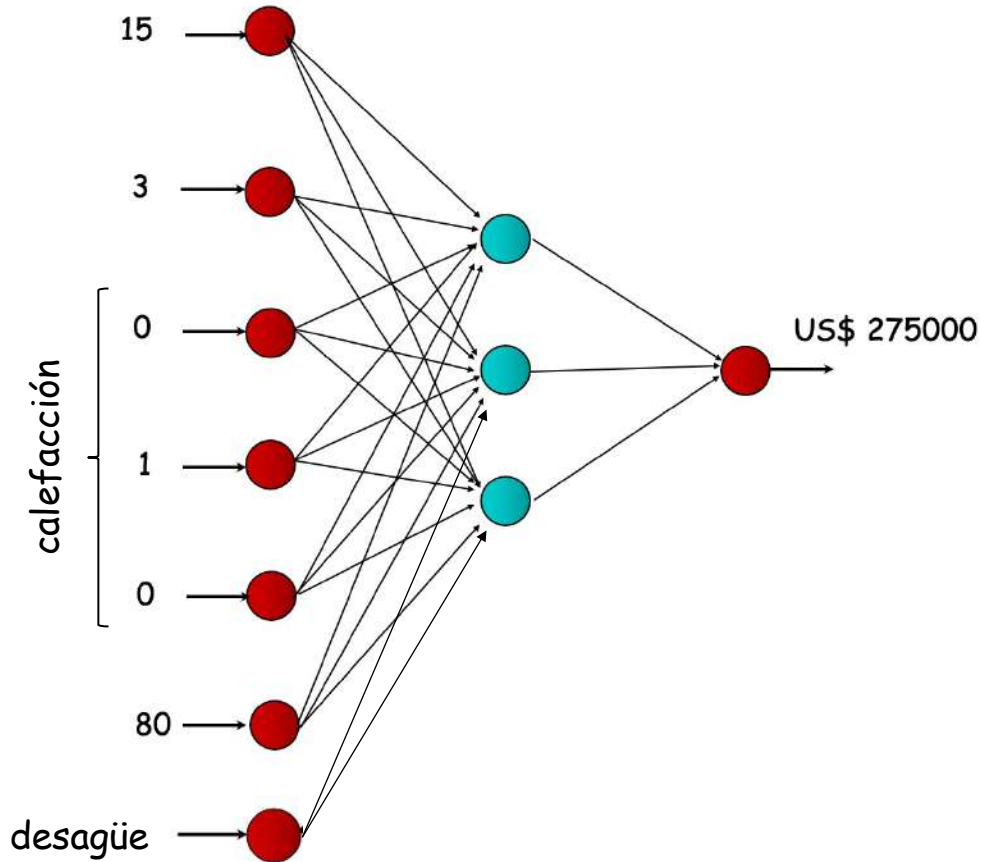
calefacción	One hot encoding		
<i>steam</i>	1	0	0
<i>electric</i>	0	1	0
<i>hot air</i>	0	0	1

Desagüe puede ser *septic* o *public*



# Redes neuronales

## One hot encoding

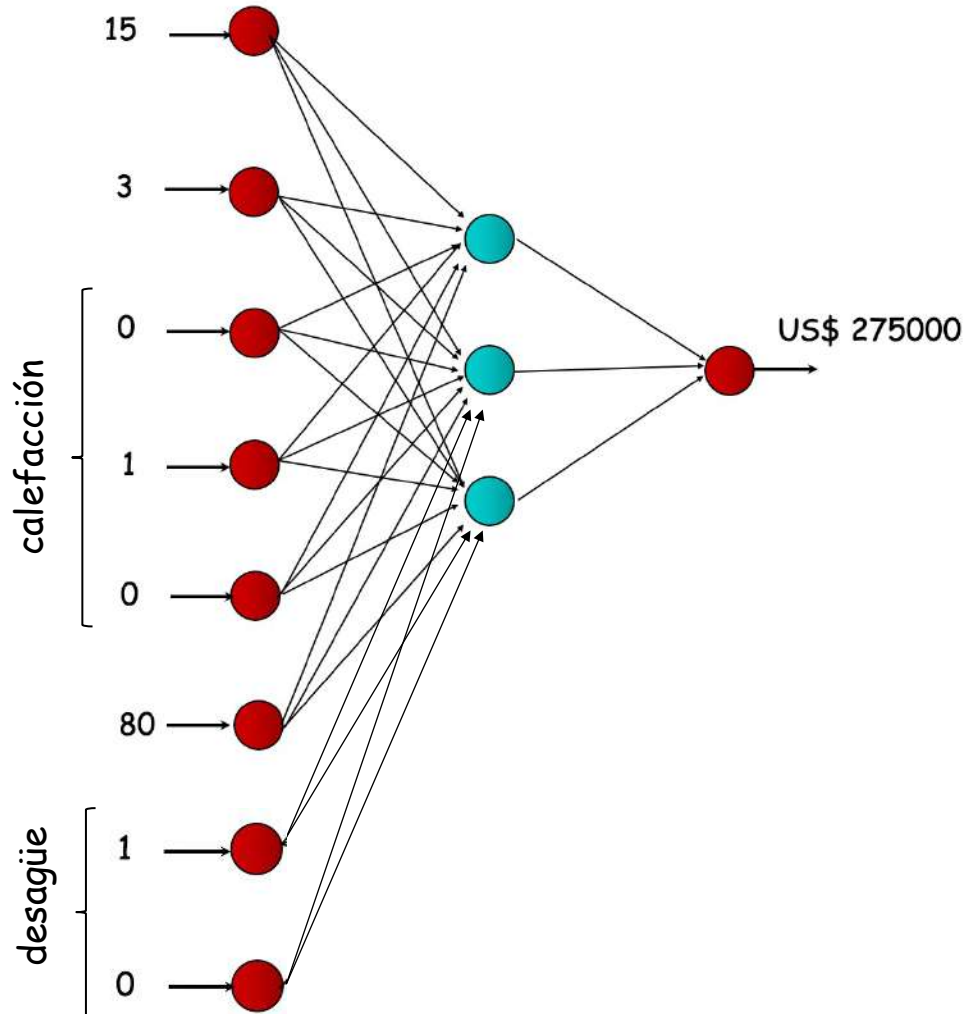


calefacción	One hot encoding		
<i>steam</i>	1	0	0
<i>electric</i>	0	1	0
<i>hot air</i>	0	0	1

desagüe	One hot encoding	
<i>septic</i>	1	0
<i>public</i>	0	1

# Redes neuronales

## One hot encoding

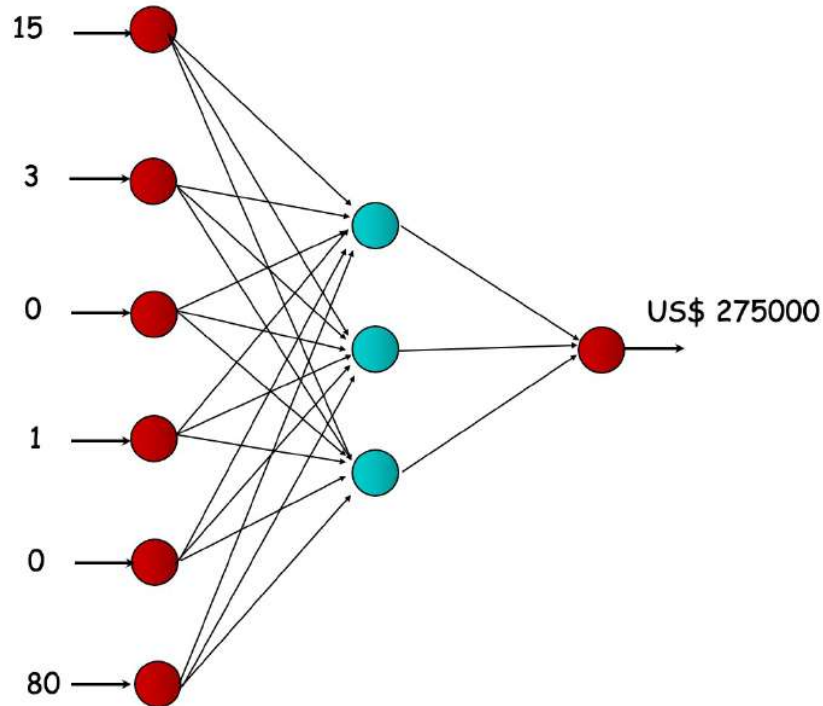


calefacción	One hot encoding		
<i>steam</i>	1	0	0
<i>electric</i>	0	1	0
<i>hot air</i>	0	0	1

desagüe	One hot encoding	
<i>septic</i>	1	0
<i>public</i>	0	1

# Redes neuronales

## Normalización de los datos



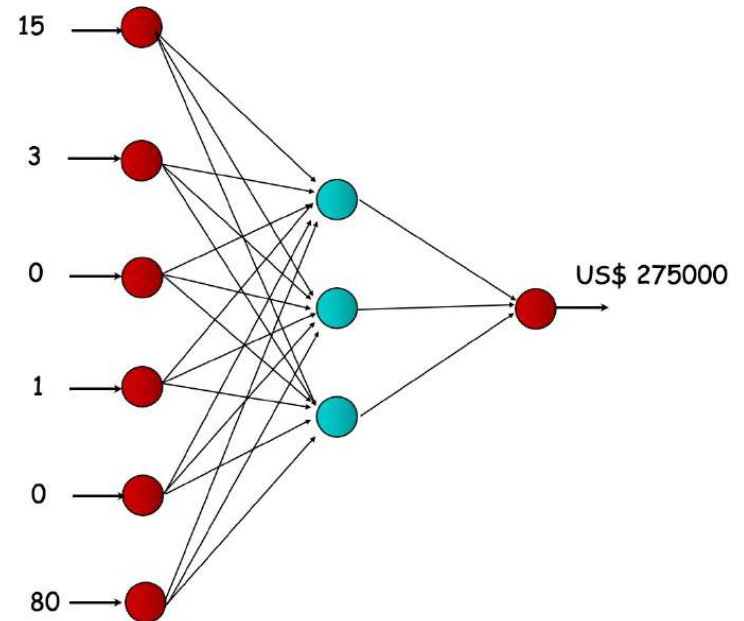
Los valores que toma el atributo **metros\_habitable** son mucho más grandes que el atributo **habitaciones**. Esto puede ocasionar que **metros\_habitable** domine los cálculos matemáticos en la construcción del modelo

# Redes neuronales

## Normalización de los datos

Se deben escalar los datos de tal forma que todos estén en un mismo rango

- *StandardScaler()*
- *MinMaxScaler()*  $[0,1]$

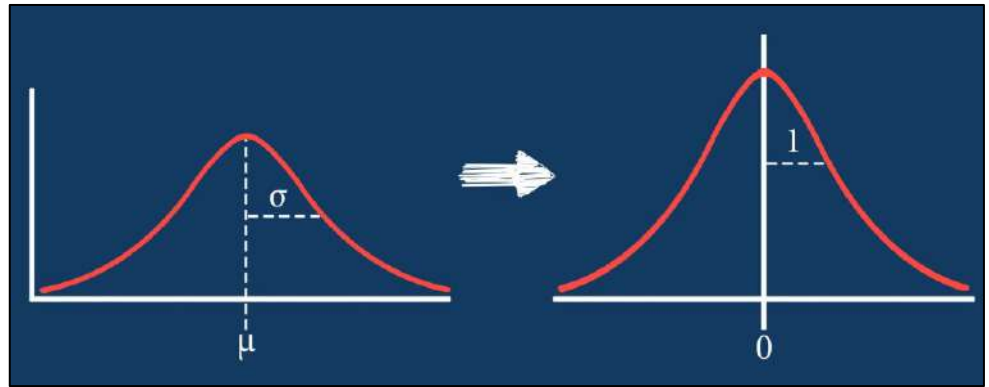


# Redes neuronales

---

## StandardScaler()

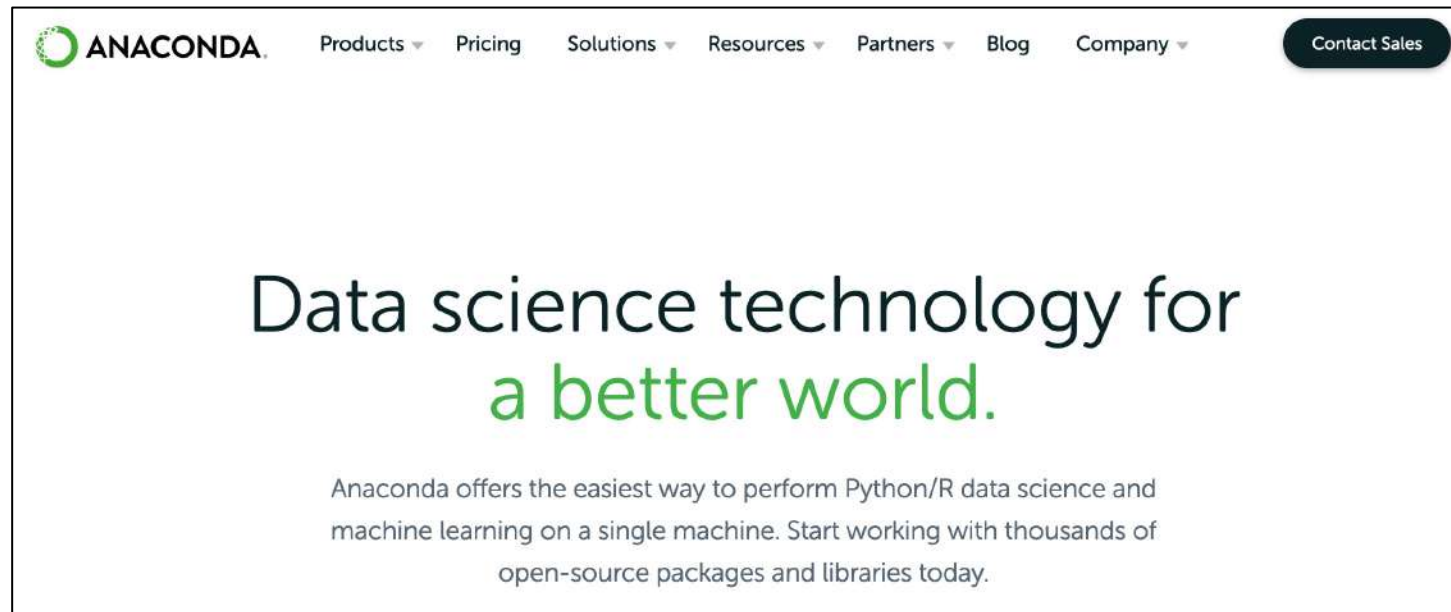
$$z = \frac{x_i - \mu}{\sigma}$$



# Redes neuronales

---

Anaconda es una distribución libre y abierta de los lenguajes Python y R, utilizada en ciencia de datos y machine learning



<https://www.anaconda.com/>

# Redes neuronales

---

**jupyter** es un ambiente web interactivo para trabajar con *notebooks*, código fuente, y datos



# Redes neuronales

scikit-learn es una biblioteca de *machine learning* en Python

## scikit-learn

Machine Learning in Python

[Getting Started](#) [Release Highlights for 1.1](#) [GitHub](#)

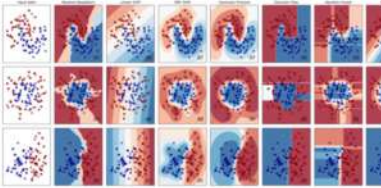
- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

### Classification

Identifying which category an object belongs to.

**Applications:** Spam detection, image recognition.

**Algorithms:** SVM, nearest neighbors, random forest, and more...

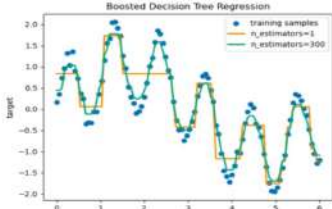


### Regression

Predicting a continuous-valued attribute associated with an object.

**Applications:** Drug response, Stock prices.

**Algorithms:** SVR, nearest neighbors, random forest, and more...




### Clustering

Automatic grouping of similar objects into sets.

**Applications:** Customer segmentation, Grouping experiment outcomes

**Algorithms:** k-Means, spectral clustering, mean-shift, and more...





# Redes neuronales

---

## Regresión y clasificación

Sexo	Edad	Tipo insulina	Peso	Glucosa media	Dosis insulina (UI)
F	34	Intermedia	55	136	1.192
M	41	Rápida	73	170	0.918
M	57	Rápida	65	89	0.929
F	22	Intermedia	58	170	0.827
M	29	Prolongada	69	215	1.148

# Redes neuronales

---

## Regresión y clasificación

Sexo	Edad	Rating crediticio	Hijos	Compra TV
F	45	Bueno	2	NO
M	23	Excelente	0	SÍ
F	31	Bueno	1	NO
F	37	Excelente	1	NO
M	40	Excelente	2	SÍ

# Redes neuronales

---

## Regresión y clasificación

Habitaciones	Área	Aire acondicionado	Antigüedad	Precio vivienda
3	80	NO	14	135000
2	65	NO	25	105000
4	120	NO	6	279000
2	85	SÍ	0	167000
2	74	SÍ	18	122000

# Redes neuronales

---

## Regresión y clasificación

Sexo	Edad	Malnutrición	Contactos	Trimetoprin	Tuberculosis
M	45	NO	0	SÍ	NO
M	25	NO	2	NO	SÍ
F	28	NO	1	SÍ	NO
F	37	SÍ	0	SÍ	NO
M	33	SÍ	3	NO	SÍ

# Redes neuronales

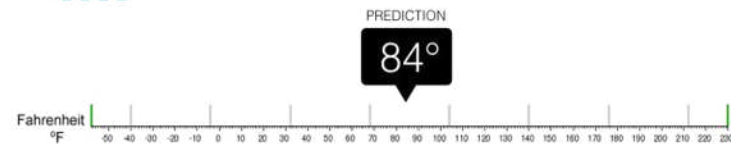


Regresión



**Regresión:**

¿Cuál va a ser la temperatura mañana?

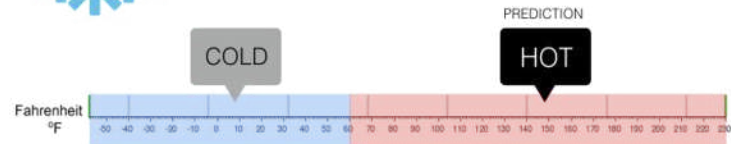


Clasificación



**Clasificación**

¿Mañana estará frío o caliente?



# Redes neuronales

---

## Saratoga houses

Datos de 1728 viviendas de Saratoga County (New York) de 2006. Cada vivienda se describe por medio de 16 variables



#	Atributo	Descripción
1	precio	precio de la vivienda en dólares
2	tamaño_lote	tamaño del lote en acres
3	antigüedad	años de construida la vivienda
4	precio_terreno	precio del terreno
5	area_construida	área construida en pies cuadrados
6	universitarios	porcentaje de habitantes del barrio que tienen título universitario
7	dormitorios	cantidad de dormitorios
8	chimenea	cantidad de chimeneas
9	baños	cantidad de baños (medio baño si no tiene ducha)
10	habitaciones	cantidad de habitaciones
11	calefacción	tipo de calefacción (electric, steam, hot air)
12	consumo_calefacción	tipo de combustible usado para la calefacción (electric, gas, oil)
13	desagüe	tipo de desagüe (septic, public, none)
14	vista_lago	indica si la vivienda tiene, o no, vista al lago
15	nueva_construcción	indica si la vivienda es, o no, nueva
16	aire_acondicionado	indica si la vivienda tiene, o no, aire acondicionado

#	Atributo	Descripción
1	precio	precio de la vivienda en dólares
2	tamaño_lote	tamaño del lote en acres
3	antigüedad	años de construida la vivienda
4	precio_terreno	precio del terreno
5	area_construida	área construida en pies cuadrados
6	universitarios	porcentaje de habitantes del barrio que tienen título universitario
7	dormitorios	cantidad de dormitorios
8	chimenea	cantidad de chimeneas
9	baños	cantidad de baños (medio baño si no tiene ducha)
10	habitaciones	cantidad de habitaciones
11	calefacción	tipo de calefacción (electric, steam, hot air)
12	consumo_calefacción	tipo de combustible usado para la calefacción (electric, gas, oil)
13	desagüe	tipo de desagüe (septic, public, none)
14	vista_lago	indica si la vivienda tiene, o no, vista al lago
15	nueva_construcción	indica si la vivienda es, o no, nueva
16	aire_acondicionado	indica si la vivienda tiene, o no, aire acondicionado

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
132500	0.09	42	50000	906	35	2	1	1	5	electric	electric	septic	No	No	No



#	Atributo	Descripción
1	precio	precio de la vivienda en dólares
2	tamaño_lote	tamaño del lote en acres
3	antigüedad	años de construida la vivienda
4	precio_terreno	precio del terreno
5	area_construida	área construida en pies cuadrados
6	universitarios	porcentaje de habitantes del barrio que tienen título universitario
7	dormitorios	cantidad de dormitorios
8	chimenea	cantidad de chimeneas
9	baños	cantidad de baños (medio baño si no tiene ducha)
10	habitaciones	cantidad de habitaciones
11	calefacción	tipo de calefacción (electric, steam, hot air)
12	consumo_calefacción	tipo de combustible usado para la calefacción (electric, gas, oil)
13	desagüe	tipo de desagüe (septic, public, none)
14	vista_lago	indica si la vivienda tiene, o no, vista al lago
15	nueva_construcción	indica si la vivienda es, o no, nueva
16	aire_acondicionado	indica si la vivienda tiene, o no, aire acondicionado

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
132500	0.09	42	50000	906	35	2	1	1	5	electric	electric	septic	No	No	No

# Redes neuronales

---

**datos (1728 viviendas)**

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
132500	0.09	42	50000	906	35	2	1	1	5	electric	electric	septic	No	No	No
155000	0.41	13	18700	998	51	3	1	1	4	hot air	gas	septic	No	No	No
86060	0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
120000	0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
122900	1.94	4	21200	1416	44	3	0	2	6	hot air	gas	none	No	No	No
158000	0.36	10	12100	1348	52	2	1	2	3	hot air	gas	public	No	No	Yes
257386	0.5	0	32000	2969	52	4	1	4	10	hot air	gas	public	No	Yes	No
62000	0.4	98	10500	1317	44	3	0	1	5	steam	gas	septic	No	No	No

## train\_data (80%)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
122900	1.94	4	21200	1416	44	3	0	2	6	hot air	gas	none	No	No	No
158000	0.36	10	12100	1348	52	2	1	2	3	hot air	gas	public	No	No	Yes
257386	0.5	0	32000	2969	52	4	1	4	10	hot air	gas	public	No	Yes	No
62000	0.4	98	10500	1317	44	3	0	1	5	steam	gas	septic	No	No	No
132500	0.09	42	50000	906	35	2	1	1	5	electric	electric	septic	No	No	No
155000	0.41	13	18700	998	51	3	1	1	4	hot air	gas	septic	No	No	No

## test\_data (20%)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
86060	0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
120000	0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
142700	0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
225000	0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No

## train\_data (1382 viviendas)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
122900	1.94	4	21200	1416	44	3	0	2	6	hot air	gas	none	No	No	No
158000	0.36	10	12100	1348	52	2	1	2	3	hot air	gas	public	No	No	Yes
257386	0.5	0	32000	2969	52	4	1	4	10	hot air	gas	public	No	Yes	No
62000	0.4	98	10500	1317	44	3	0	1	5	steam	gas	septic	No	No	No
132500	0.09	42	50000	906	35	2	1	1	5	electric	electric	septic	No	No	No
155000	0.41	13	18700	998	51	3	1	1	4	hot air	gas	septic	No	No	No

## test\_data (346 viviendas)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
86060	0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
120000	0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
142700	0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
225000	0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No


## train\_data (1382 viviendas)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
122900	1.94	4	21200	1416	44	3	0	2	6	hot air	gas	none	No	No	No
158000	0.36	10	12100	1348	52	2	1	2	3	hot air	gas	public	No	No	Yes
257386	0.5	0	32000	2969	52	4	1	4	10	hot air	gas	public	No	Yes	No
62000	0.4	98	10500	1317	44	3	0	1	5	steam	gas	septic	No	No	No
132500	0.09	42	50000	906	35	2	1	1	5	electric	electric	septic	No	No	No
155000	0.41	13	18700	998	51	3	1	1	4	hot air	gas	septic	No	No	No

# train\_data (1382 viviendas)

y\_train

X\_train




1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
122900	1.94	4	21200	1416	44	3	0	2	6	hot air	gas	none	No	No	No
158000	0.36	10	12100	1348	52	2	1	2	3	hot air	gas	public	No	No	Yes
257386	0.5	0	32000	2969	52	4	1	4	10	hot air	gas	public	No	Yes	No
62000	0.4	98	10500	1317	44	3	0	1	5	steam	gas	septic	No	No	No
132500	0.09	42	50000	906	35	2	1	1	5	electric	electric	septic	No	No	No
155000	0.41	13	18700	998	51	3	1	1	4	hot air	gas	septic	No	No	No

## train\_data (1382 viviendas)

y\_train

X\_train



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
122900	1.94	4	21200	1416	44	3	0	2	6	hot air	gas	none	No	No	No
158000	0.36	10	12100	1348	52	2	1	2	3	hot air	gas	public	No	No	Yes
257386	0.5	0	32000	2969	52	4	1	4	10	hot air	gas	public	No	Yes	No
62000	0.4	98	10500	1317	44	3	0	1	5	steam	gas	septic	No	No	No
132500	0.09	42	50000	906	35	2	1	1	5	electric	electric	septic	No	No	No
155000	0.41	13	18700	998	51	3	1	1	4	hot air	gas	septic	No	No	No

**y\_train:** es el valor que se quiere predecir en el conjunto de entrenamiento (variable dependiente)

**X\_train:** son todos los valores que influyen en el precio (variables independientes) en el conjunto de entrenamiento

# Redes neuronales

---

## Hiperparámetros de las redes neuronales

- **hidden\_layer\_sizes**: número y tamaño de las capas ocultas. Por ejemplo, (100) para una única capa oculta con 100 neuronas, y (4,7) para dos capas ocultas, la primera con 4 neuronas y la segunda con 7
- **activation**: función de activación de las capas ocultas. Puede ser: {'identity', 'logistic', 'tanh', 'relu'}. La misma función de activación se aplica a todas las capas ocultas, no se permiten distintas. El valor por defecto es 'relu'
- **solver**: el algoritmo de optimización utilizado para aprender los pesos y bias de la red. Puede ser: {'lbfgs', 'sgd', 'adam'}. Por defecto se utiliza 'adam', que es el que mejores resultados suele dar para conjuntos de datos con miles de observaciones. Para conjuntos de datos pequeños, 'lbfgs' converge más rápido y puede conseguir mejores resultados
- **alpha**: regularización L2 (weight decay). El valor por defecto es 0.0001

[https://scikit-learn.org/stable/modules/generated/sklearn.neural\\_network.MLPClassifier.html](https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html)



# Redes neuronales

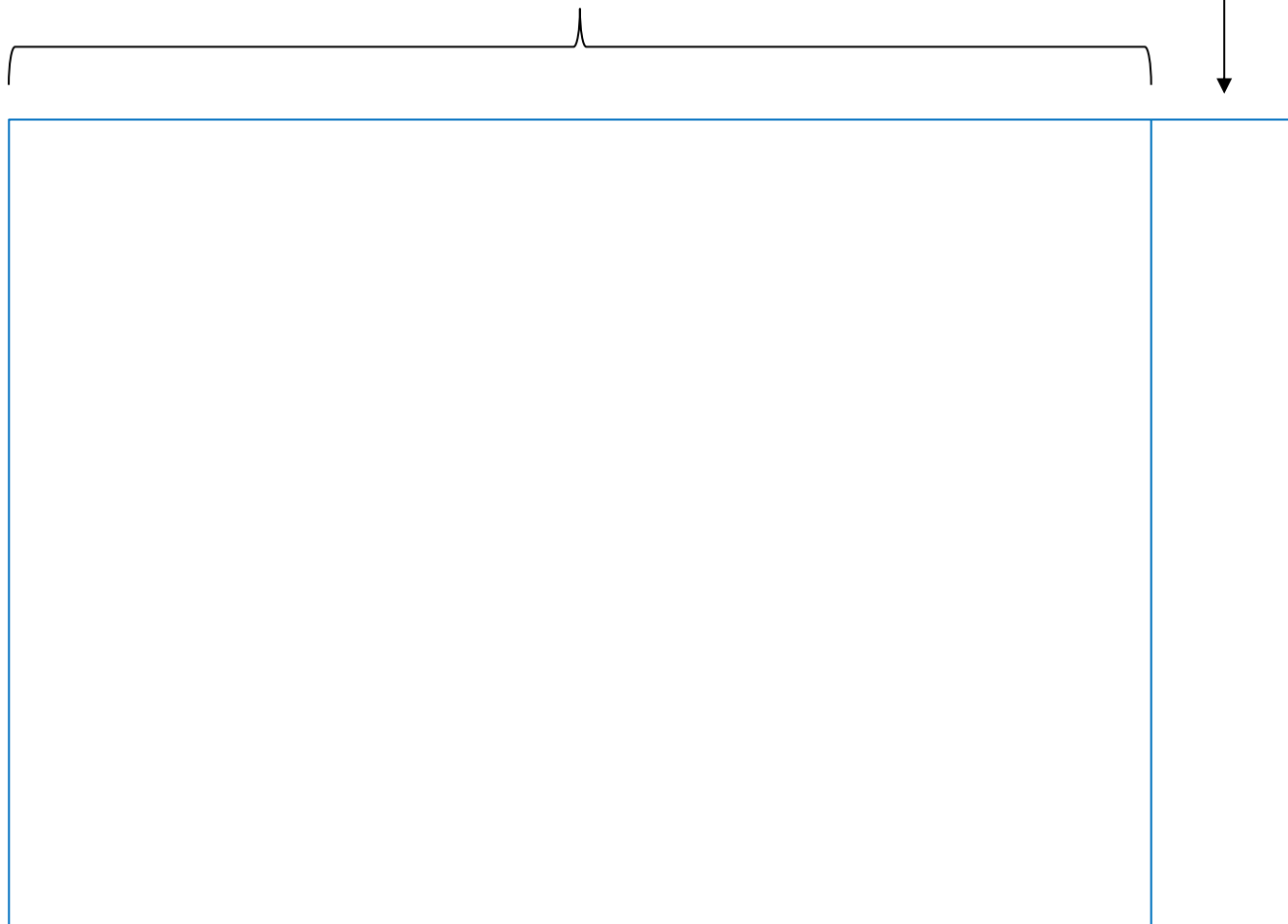
---

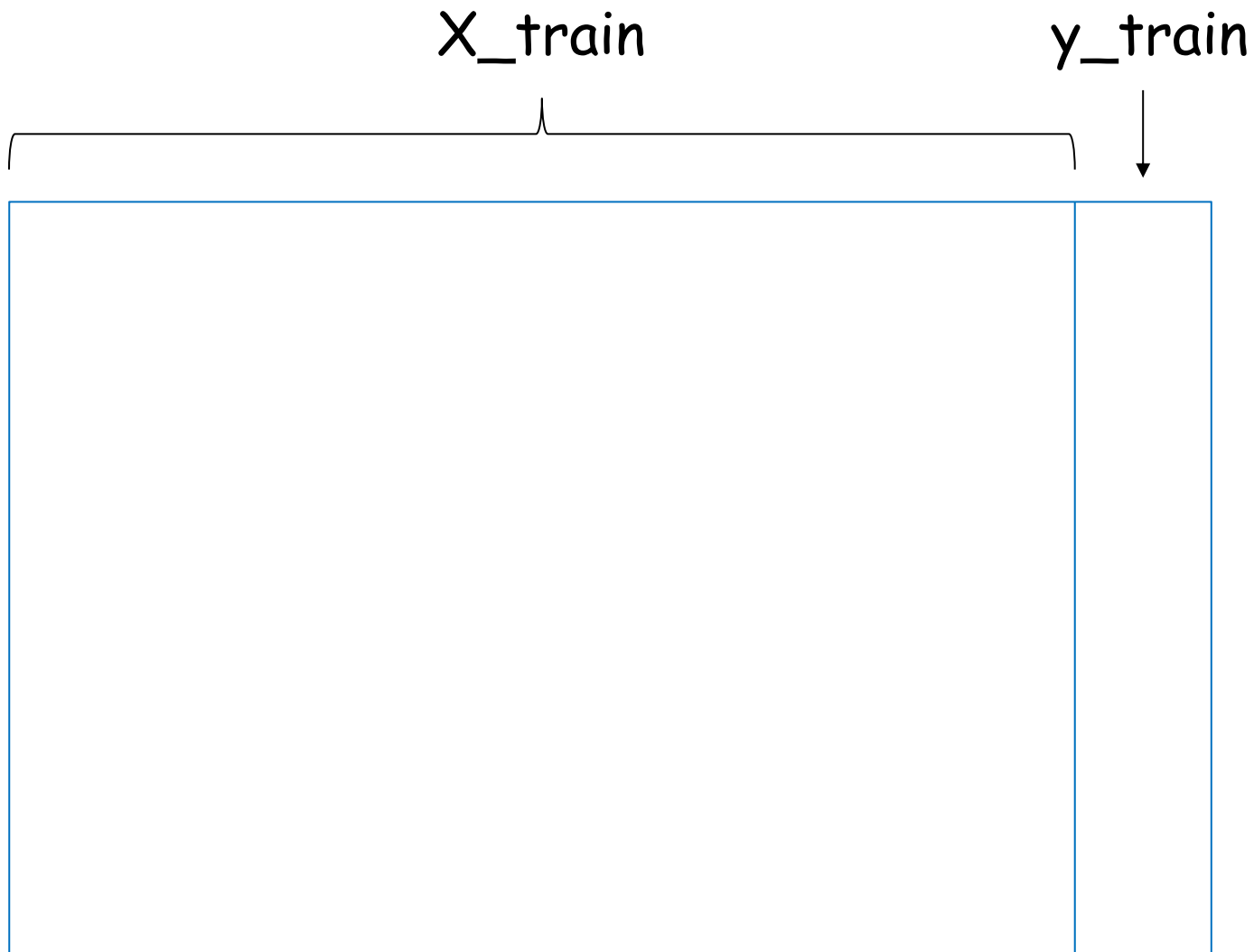
## Validación cruzada (*cross-validation*)

Es una técnica utilizada para evaluar los resultados de un modelo predictivo y garantizar que sean independientes de la partición entre datos de entrenamiento y de prueba

$X_{\text{train}}$

$y_{\text{train}}$





*Validación cruzada ( $k=5$  folds)*


Se divide el conjunto de entrenamiento en 5 partes


Se toman  $\frac{4}{5}$  para entrenar y  $\frac{1}{5}$  para probar


E1

Se calcula el error E1


E1

Se repite el procedimiento cambiando los datos que hicieron parte del conjunto de entrenamiento y prueba


Se toman  $\frac{4}{5}$  para entrenar y  $\frac{1}{5}$  para probar




E2

Se calcula el error E2


E2

Se repite el procedimiento cambiando los datos que hicieron parte del conjunto de entrenamiento y prueba


Se toman  $\frac{4}{5}$  para entrenar y  $\frac{1}{5}$  para probar


E3

Se calcula el error E3

	E3

Se repite el procedimiento cambiando los datos que hicieron parte del conjunto de entrenamiento y prueba


Se toman  $\frac{4}{5}$  para entrenar y  $\frac{1}{5}$  para probar


E4

Se calcula el error E4


E4

Se repite el procedimiento cambiando los datos que hicieron parte del conjunto de entrenamiento y prueba



Se toman  $\frac{4}{5}$  para entrenar y  $\frac{1}{5}$  para probar


E5

Se calcula el error E5

		E5
		E4
		E3
		E2
		E1

Se promedian los errores,  $(E1+E2+E3+E4+E5)/5$

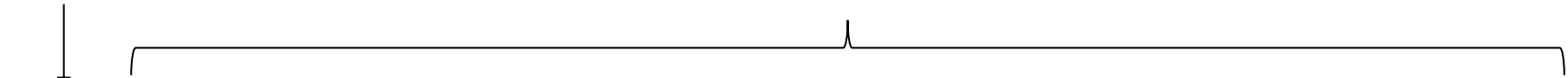
## test\_data (346 viviendas)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
86060	0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
120000	0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
142700	0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
225000	0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No

## test\_data (346 viviendas)

y\_test

X\_test

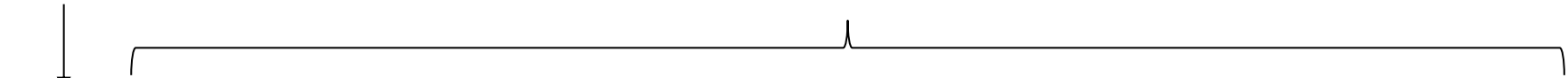


1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
86060	0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
120000	0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
142700	0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
225000	0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No

## test\_data (346 viviendas)

y\_test

X\_test



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
86060	0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
120000	0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
142700	0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
225000	0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No

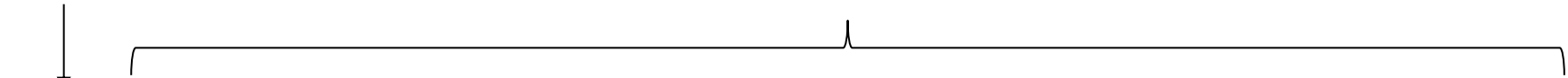
**y\_test:** es el valor que se quiere predecir en el conjunto de prueba

**X\_test:** son todos los valores que influyen en el precio en las viviendas del conjunto de prueba

## test\_data (346 viviendas)

y\_test

X\_test



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
86060	0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
120000	0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
142700	0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
225000	0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No

## test\_data (346 viviendas)

X\_test

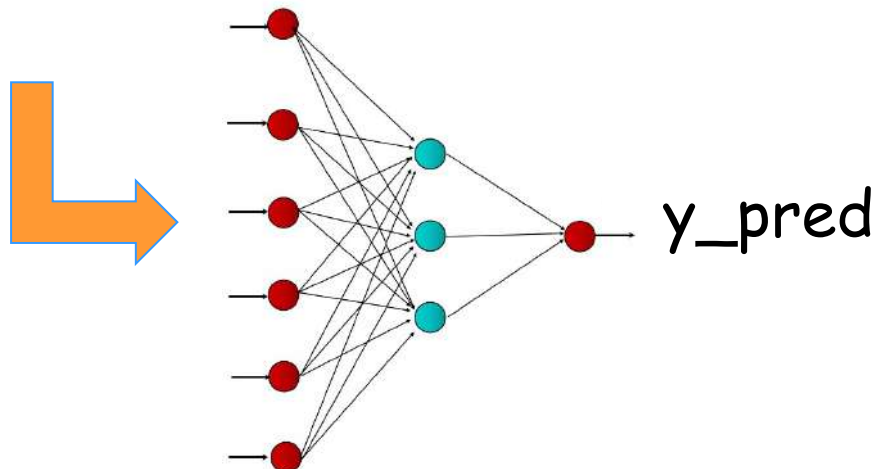
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No



## test\_data (346 viviendas)

X\_test

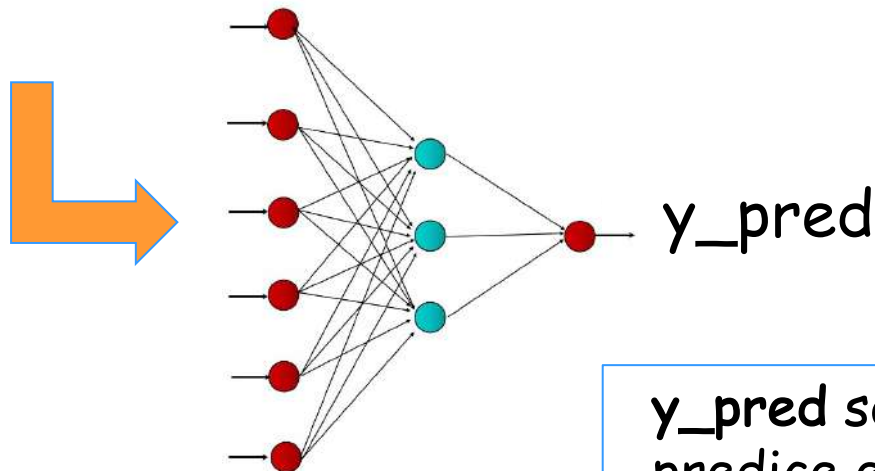
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No



## test\_data (346 viviendas)

X\_test

2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No



$y_{pred}$  son los valores que la red predice en el conjunto de prueba

y\_train

X\_train

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
122900	1.94	4	21200	1416	44	3	0	2	6	hot air	gas	none	No	No	No
158000	0.36	10	12100	1348	52	2	1	2	3	hot air	gas	public	No	No	Yes
257386	0.5	0	32000	2969	52	4	1	4	10	hot air	gas	public	No	Yes	No
62000	0.4	98	10500	1317	44	3	0	1	5	steam	gas	septic	No	No	No
132500	0.09	42	50000	906	35	2	1	1	5	electric	electric	septic	No	No	No
155000	0.41	13	18700	998	51	3	1	1	4	hot air	gas	septic	No	No	No

y\_test

X\_test

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
86060	0.11	0	15000	840	22	2	0	1	3	hot air	gas	public	No	Yes	Yes
120000	0.68	31	16500	1152	37	4	1	1	4	steam	electric	septic	No	No	No
142700	0.12	25	51000	1006	22	2	1	1	4	steam	egas	septic	No	No	No
225000	0.62	0	33000	2800	52	3	1	4	6	hot air	gas	septic	No	Yes	No

# Redes neuronales

---

## Error medio absoluto (MAE)

Es una medida de la diferencia entre dos variables continuas, en este caso, entre los valores que predice el modelo y los valores reales

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|$$

# Redes neuronales

---

## Error medio absoluto (MAE)

y_pred	y_test
125000	110000
134000	150000
86000	112000
94000	86000
250000	180000
118000	130000

# Redes neuronales

---

## Error medio absoluto (MAE)

y_pred
125000
134000
86000
94000
250000
118000

y_test
110000
150000
112000
86000
180000
130000

$ y_{\text{test}} - y_{\text{pred}} $
15000
16000
26000
8000
70000
12000

# Redes neuronales

---

## Error medio absoluto (MAE)

y_pred
125000
134000
86000
94000
250000
118000

y_test
110000
150000
112000
86000
180000
130000

y_test-y_pred
15000
16000
26000
8000
70000
12000

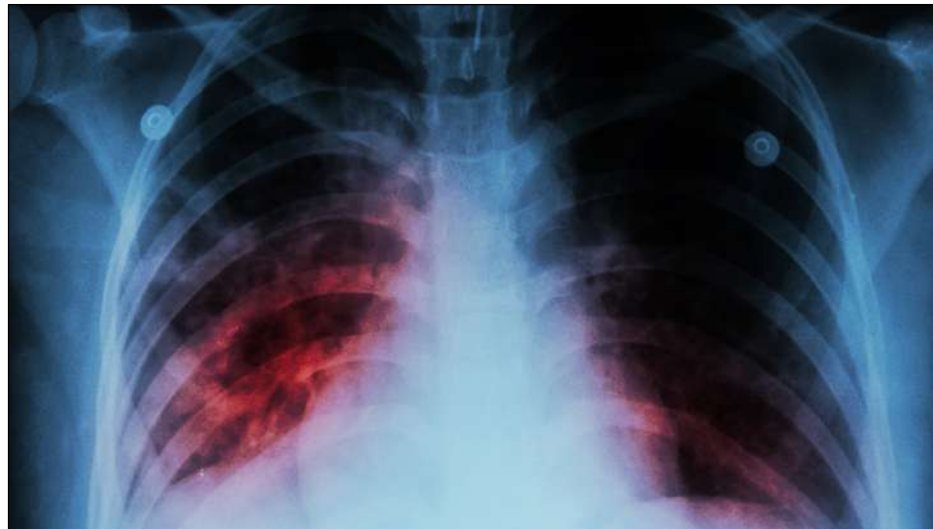
MAE=24500

# Redes neuronales

---

## Tuberculosis

Datos de 1455 pacientes con sospecha de Tuberculosis. Cada paciente se describe por medio de 15 variables





#	Atributo	Descripción
1	Sexo	0=Femenino, 1=Masculino
2	Edad	Edad en años del paciente
3	Etnia	0=Indígena, 1=Afrodescendiente, 2=Otro
4	Grupo poblacional	0= Trabajador de la salud, 1 = Habitante de calle, 2 = Fármaco dependiente, 3 = Población carcelaria, 4 = Ningún grupo poblacional
5	Comorbilidad	0 = Enfermedad renal, 1 = Otras Inmunosupresiones, 2 = Hipertensión, 3 = Diabetes, 4 = Otra enfermedad pulmonar, 5 = Ninguna
6	Malnutrición	0 = No, 1 = Si
7	Cantidad contactos	Cantidad de personas con las que tuvo contacto el paciente
8	Cantidad contactos sintomáticos respiratorios	Cantidad de personas con síntomas respiratorios con las que tuvo contacto el paciente
9	Cantidad contactos sintomáticos respiratorios con baciloscopia positiva	Cantidad de personas con síntomas respiratorios con baciloscopia positiva con las que tuvo contacto el paciente
10	Cantidad contactos con menores de 5 años	Cantidad de personas menores de 5 años con las que tuvo contacto el paciente
11	VIH	0=No realizado, 1=Positivo, 2=Negativo
12	Recibe terapia antirretroviral	0=No, 1=Si, 2=No aplica
13	Recibe Trimetoprim	0=No, 1=Si, 2=No aplica
14	Resultado de Baciloscopia	0=No realizado, 1=Positivo 1 a 9 BAAR, 2=+++, 3= ++, 4=+, 5=-
15	Tuberculosis	0=Negativo, 1=Positivo

#	Atributo	Descripción
1	Sexo	0=Femenino, 1=Masculino
2	Edad	Edad en años del paciente
3	Etnia	0=Indígena, 1=Afrodescendiente, 2=Otro
4	Grupo poblacional	0= Trabajador de la salud, 1 = Habitante de calle, 2 = Fármaco dependiente, 3 = Población carcelaria, 4 = Ningún grupo poblacional
5	Comorbilidad	0 = Enfermedad renal, 1 = Otras Inmunosupresiones, 2 = Hipertensión, 3 = Diabetes, 4 = Otra enfermedad pulmonar, 5 = Ninguna
6	Malnutrición	0=No, 1=Si
7	Cantidad contactos	Cantidad de personas con las que tuvo contacto el paciente
8	Cantidad contactos sintomáticos respiratorios	Cantidad de personas con síntomas respiratorios con las que tuvo contacto el paciente
9	Cantidad contactos sintomáticos respiratorios con baciloscopia positiva	Cantidad de personas con síntomas respiratorios con baciloscopia positiva con las que tuvo contacto el paciente
10	Cantidad contactos con menores de 5 años	Cantidad de personas menores de 5 años con las que tuvo contacto el paciente
11	VIH	0=No realizado, 1=Positivo, 2=Negativo
12	Recibe terapia antirretroviral	0=No, 1=Si, 2=No aplica
13	Recibe Trimetoprim	0=No, 1=Si, 2=No aplica
14	Resultado de Baciloscopia	0=No realizado, 1=Positivo 1 a 9 BAAR, 2=+++, 3= ++, 4=+, 5=-
15	Tuberculosis*	0=Negativo, 1=Positivo

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	39	2	4	5	0	1	0	0	0	2	2	2	5	0

# Redes neuronales

---

datos (1455 pacientes)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	22	2	4	5	0	2	0	0	0	2	2	2	5	0
0	25	2	4	5	0	10	10	6	7	2	2	2	4	1
1	39	2	2	5	0	0	0	0	0	2	2	2	4	1
0	66	1	4	5	0	11	0	11	2	2	2	2	2	0
0	39	2	4	5	0	1	0	0	0	2	2	2	5	0
0	76	2	4	5	0	0	0	0	0	2	2	2	2	0
1	37	2	4	3	0	0	0	0	0	2	2	2	4	1
0	39	2	4	5	0	1	0	0	0	2	2	2	5	0

## train\_data (80%)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	22	2	4	5	0	2	0	0	0	2	2	2	5	0
0	25	2	4	5	0	10	10	6	7	2	2	2	4	1
1	39	2	2	5	0	0	0	0	0	2	2	2	4	1
0	66	1	4	5	0	11	0	11	2	2	2	2	2	0

## test\_data (20%)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	39	2	4	5	0	1	0	0	0	2	2	2	5	0
0	76	2	4	5	0	0	0	0	0	2	2	2	2	0
1	37	2	4	3	0	0	0	0	0	2	2	2	4	1

## train\_data (1164 pacientes)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	22	2	4	5	0	2	0	0	0	2	2	2	5	0
0	25	2	4	5	0	10	10	6	7	2	2	2	4	1
1	39	2	2	5	0	0	0	0	0	2	2	2	4	1
0	66	1	4	5	0	11	0	11	2	2	2	2	2	0

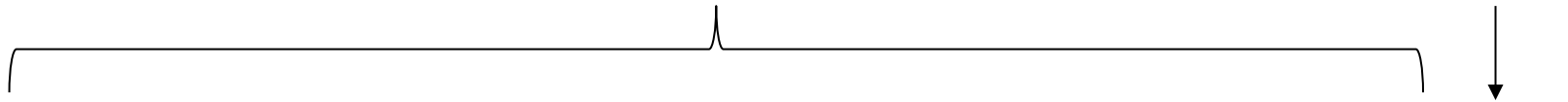
## test\_data (291 pacientes)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	39	2	4	5	0	1	0	0	0	2	2	2	5	0
0	76	2	4	5	0	0	0	0	0	2	2	2	2	0
1	37	2	4	3	0	0	0	0	0	2	2	2	4	1

train\_data (1164 pacientes)

X\_train

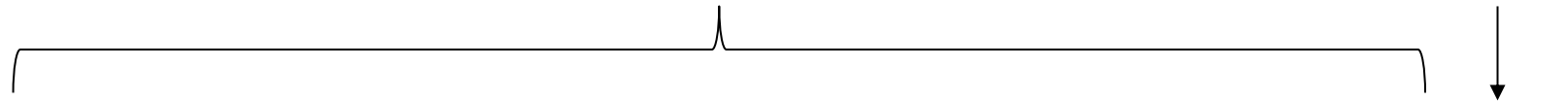
y\_train



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	22	2	4	5	0	2	0	0	0	2	2	2	5	0
0	25	2	4	5	0	10	10	6	7	2	2	2	4	1
1	39	2	2	5	0	0	0	0	0	2	2	2	4	1
0	66	1	4	5	0	11	0	11	2	2	2	2	2	0

X\_train

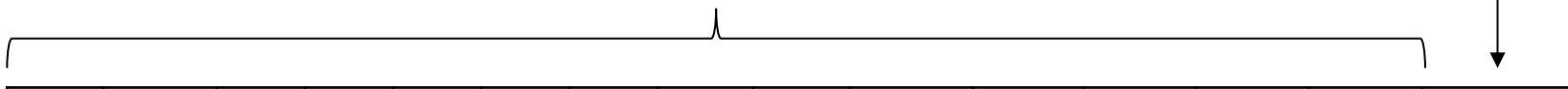
y\_train



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	22	2	4	5	0	2	0	0	0	2	2	2	5	0
0	25	2	4	5	0	10	10	6	7	2	2	2	4	1
1	39	2	2	5	0	0	0	0	0	2	2	2	4	1
0	66	1	4	5	0	11	0	11	2	2	2	2	2	0

X\_test

y\_test



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	39	2	4	5	0	1	0	0	0	2	2	2	5	0
0	76	2	4	5	0	0	0	0	0	2	2	2	2	0
1	37	2	4	3	0	0	0	0	0	2	2	2	4	1

# Redes neuronales

---

## Matriz de confusión

En problemas de clasificación se utiliza una matriz que permite visualizar la capacidad predictiva del modelo

Valor real	Valor predicho	
	Negativo (0)	Positivo (1)
Positivo (1)	80	9
Negativo (0)	27	175



# Redes neuronales

---

## Matriz de confusión

En problemas de clasificación se utiliza una matriz que permite visualizar la capacidad predictiva del modelo

Valor real	Negativo (0)	Positivo (1)
	Negativo (0)	Positivo (1)
Positivo (1)	27	175
Negativo (0)	TN (True Negative) 9	

# Redes neuronales

---

## Matriz de confusión

En problemas de clasificación se utiliza una matriz que permite visualizar la capacidad predictiva del modelo

Valor real	Negativo (0)	Positivo (1)
	Negativo (0)	Positivo (1)
Positivo (1)	27	TP (True Positive)
Negativo (0)	TN (True Negative)	9

# Redes neuronales

---

## Matriz de confusión

En problemas de clasificación se utiliza una matriz que permite visualizar la capacidad predictiva del modelo

Valor real	Negativo (0)	Positivo (1)
	Negativo (0)	Positivo (1)
Positivo (1)	TN (True Negative)	FP (False Positive)
Negativo (0)	27	TP (True Positive)

# Redes neuronales

## Matriz de confusión

En problemas de clasificación se utiliza una matriz que permite visualizar la capacidad predictiva del modelo

Valor real	Negativo (0)	Positivo (1)
	Negativo (0)	Positivo (1)
Positivo (1)	27	TP (True Positive)
Negativo (0)	TN (True Negative)	FP (False Positive)

FP: pacientes que el modelo predice como positivos (con Tuberculosis) pero que están sanos

# Redes neuronales

---

## Matriz de confusión

En problemas de clasificación se utiliza una matriz que permite visualizar la capacidad predictiva del modelo

Valor real	Valor predicho	
	Negativo (0)	Positivo (1)
Positivo (1)	TP (True Positive)	FP (False Positive)
Negativo (0)	TN (True Negative)	FN (False Negative)

# Redes neuronales

## Matriz de confusión

En problemas de clasificación se utiliza una matriz que permite visualizar la capacidad predictiva del modelo

Valor real	Negativo (0)	Positivo (1)
	Negativo (0)	Positivo (1)
Positivo (1)	TN (True Negative)	FP (False Positive)
Negativo (0)	FN (False Negative)	TP (True Positive)

FN: pacientes que el modelo predice como sanos pero realmente tienen tuberculosis

# Redes neuronales

---

## Matriz de confusión

En problemas de clasificación se utiliza una matriz que permite visualizar la capacidad predictiva del modelo

Valor real	Negativo (0)	Positivo (1)
	Negativo (0)	Positivo (1)
Positivo (1)	89	0
Negativo (0)	0	202

# Redes neuronales

---

## Métricas

La exactitud o *accuracy* indica la cantidad correcta de predicciones (positivas o negativas) sobre el total de datos

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$



# Redes neuronales

---

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Positivo (1) / Negativo (0)		
	Negativo (0)	Positivo (1)
Negativo (0)	80	9
Positivo (1)	27	175

Accuracy=?

Positivo (1) / Negativo (0)		
	Negativo (0)	Positivo (1)
Negativo (0)	53	36
Positivo (1)	42	160

Accuracy=?

# Redes neuronales

---

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Positivo (1) Negativo (0)	80	9
	27	175
	Negativo (0)	Positivo (1)

Accuracy=87.62%

Positivo (1) Negativo (0)	53	36
	42	160
	Negativo (0)	Positivo (1)

Accuracy=73.19%