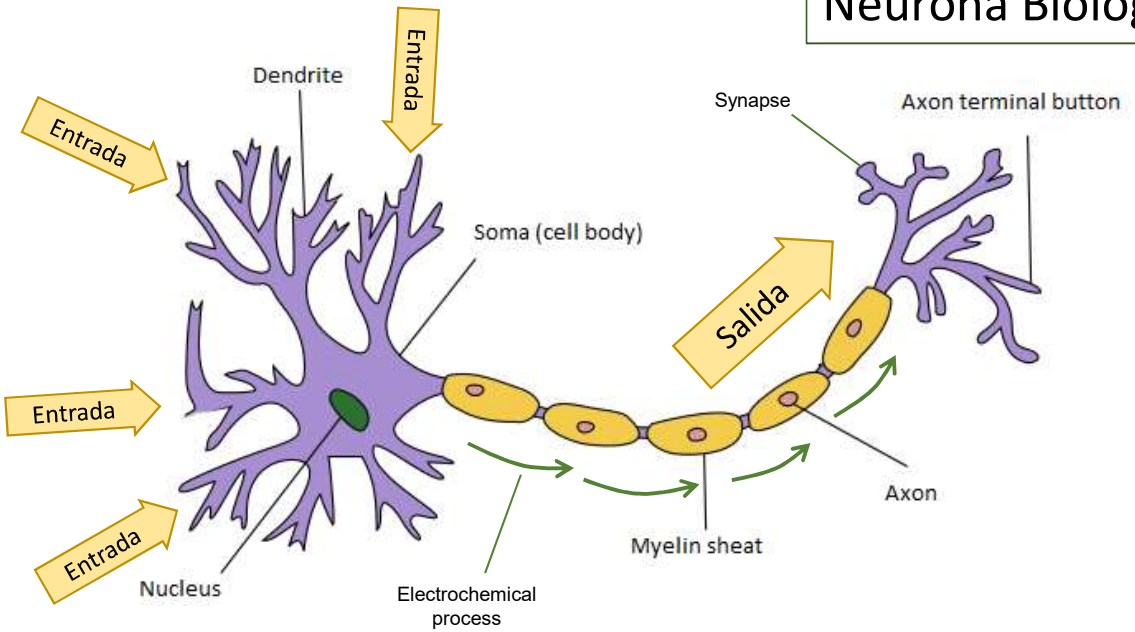
Redes Neuronales Artificiales:

Perceptrón Simple

Aprendizaje Automático

Dr. Luis Eduardo Falcón Morales
ITESM
Campus Guadalajara

Neurona Biológica



By Quasar Jarosz. From Wikipedia en inglés, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=7616130

Santiago Ramón y Cajal (español 1852 – 1934, premio Nobel de Medicina 1906) propone el modelo de la neurona biológica.

- El sistema nervioso está compuesto por una red de células individuales, llamadas neuronas, ampliamente interconectadas entre sí.
- Desde un punto de vista funcional, son procesadores de información **sencillos**.
- Existen muchas cualidades del individuo que no son innatas, sino que se adquieren por la influencia de la información obtenida del medio externo a través de sus sensores.
- Las neuronas **aprenden** a ponderar la información de acuerdo a su importancia o frecuencia.

¿cuántas neuronas requiere un ser vivo para tomar decisiones?

302 neuronas
Caenorhabditis elegans
Tipo de gusano de ~1 mm de longitud.

~ 1’000,000
abeja

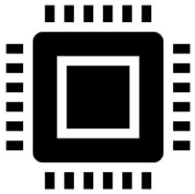
~ 16’000,000
rana

$\sim 90,000',000,000$
Homo sapiens

$\sim 250,000',000,000$
Elefante

~ 100,000
mosca de la fruta

TrueNorth - IBM
Neuromorphic chip



Neuromorphic chip: sistemas que tratan de emular la arquitectura neuro-biológica del cerebro humano.

~1’000,000 en 2014

Está claro que las neuronas artificiales están muy lejos de realizar las mismas funciones que las neuronas biológicas. El chip con 1 millón de neuronas artificiales está muy lejos de hacer lo que una abeja con esa misma cantidad de neuronas biológicas.

Prueba de Turing

- De manera simple, pordemos decir que la prueba de Turing nos habla de llegar a construir computadoras-sistemas que interactúen con un ser humano y que este no sea capaz de distinguir si la interacción la está llevando a cabo con una computadora o con otra persona.

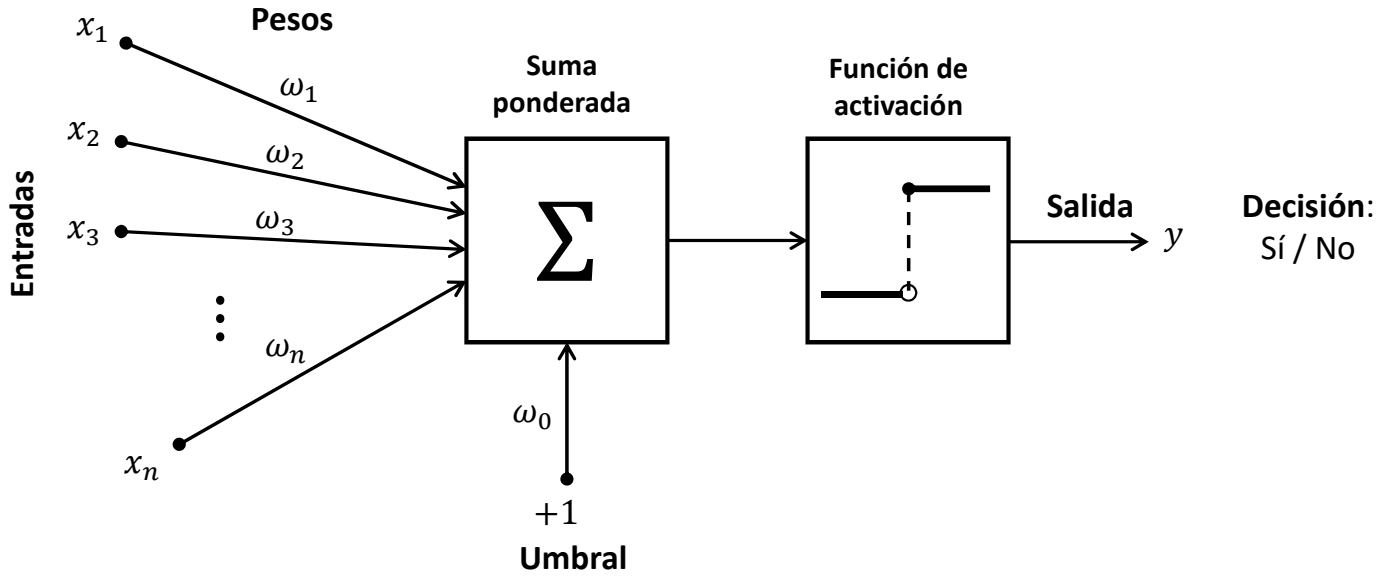
- Pero, ¿es necesario tratar de modelar o reconstruir el cerebro de un ser vivo de manera fidedigna, o es suficiente diseñar algoritmos y sistemas que generen comportamientos y soluciones que resuelvan problemas particulares de nuestra vida diaria?

Por el momento, en el área de Aprendizaje Automático (machine learning), lo que nos interesa es generar sistemas que nos ayuden a resuelver problemas de nuestra vida diaria. Dichos sistemas pueden partir o basarse en la forma en que lo hace o funciona el cerebro humano, pero el objetivo no es imitarlo o copiarlo.

Neurona de McCulloch & Pitts
Primer modelo matemático de una neurona artificial - 1943

Aunque el primer modelo de la neurona artificial simple fue propuesto en 1943, no fue sino hasta los años 60s y posteriormente en los 80s, que se extendió el estudio y alcance de las ahora llamadas redes neuronales artificiales.

Modelo propuesto por Warren McCulloch (EEUU 1898 – 1969) y Walter Pitts (EEUU, 1923 – 1969)



Lo podemos ver como una función de la forma:

$$f(\omega_0 + \omega_1 x_1 + \omega_2 x_2 + \cdots + \omega_m x_m) = y$$

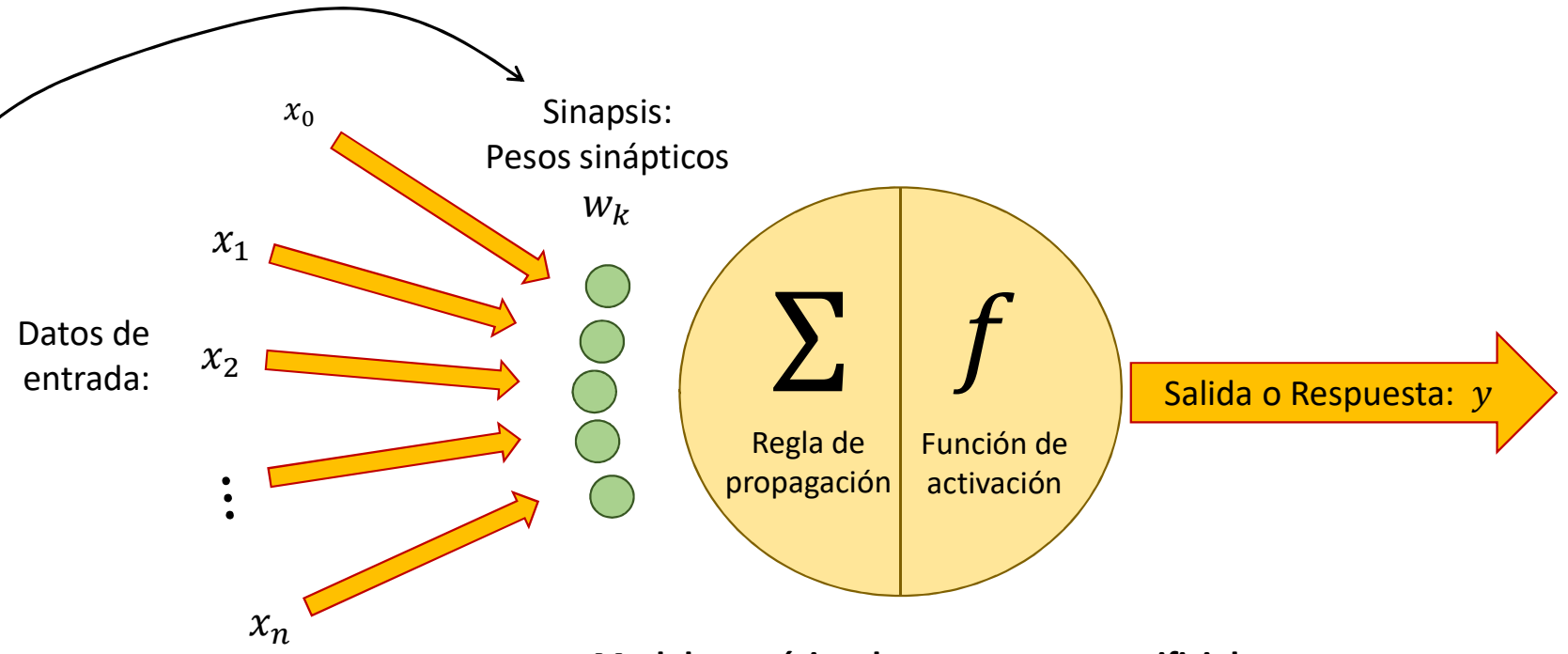
W. McCulloch, W. Pitts.
A logical calculus of the ideas immanent in nervous activity.
Bulletin of Mathematical Biophysics, 1943.

Neurona Artificial

Modelo genérico: Un procesador elemental o neurona artificial es un dispositivo simple de cálculo, que a partir de un vector de entrada procedente del exterior o de otras neuronas, proporciona una única respuesta o salida.

Sinapsis excitadora de la neurona si $w_k > 0$.

Sinapsis inhibidora de la neurona si $w_k < 0$.

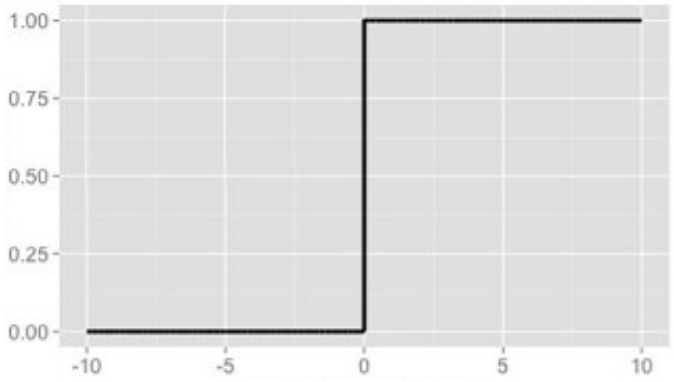


Modelo genérico de una neurona artificial

Función de Activación

La función de activación es el mecanismo mediante el cual la neurona artificial procesa la información que será propagada a lo largo de la red.

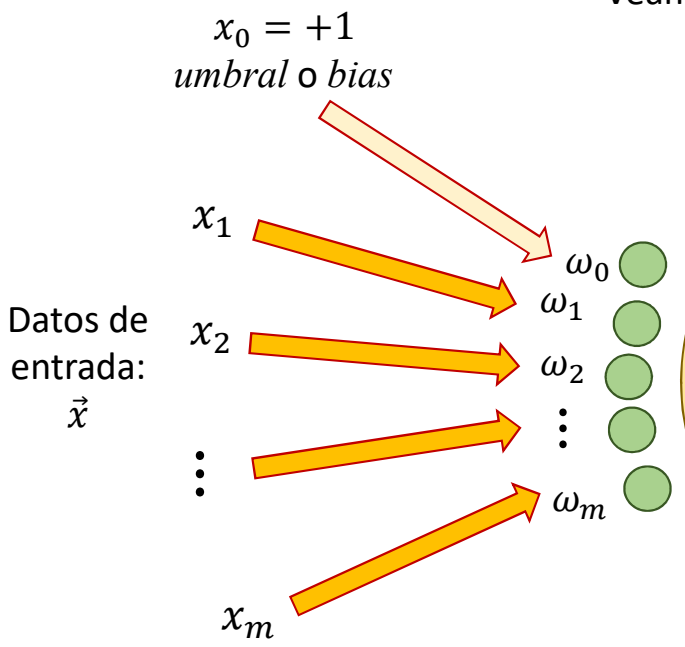
Función Escalón



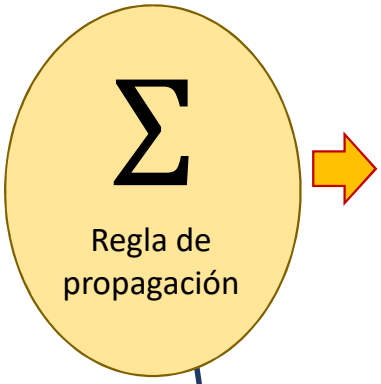
Modelo de la Neurona de McCulloch-Pitts
Rango: $\{0, +1\}$

Veamos cómo podemos interpretar la expresión matemática:

$$f(\omega_0 + \omega_1 x_1 + \omega_2 x_2 + \cdots + \omega_m x_m) = y$$



Datos de entrada: $\vec{x}$



En su forma más simple, la función de activación se toma como la función escalón:

Salida {Sí:1, No:0} y

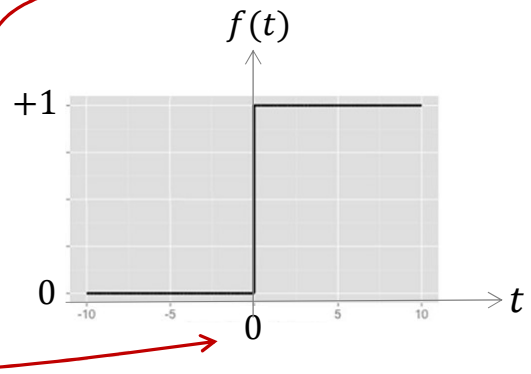
$$y = f(t) = \begin{cases} +1 & si \quad t > 0 \\ 0 & si \quad t \leq 0 \end{cases}$$

Función Escalón

$$t = \vec{\omega}_k^T \vec{x}_j = \sum_{j=0}^{m} \omega_j x_j = \sum_{j=1}^{m} \omega_j x_j + \omega_0$$

Regla de propagación representada como producto matricial o producto interior de dos vectores.

Podemos decir que con esta función escalón el umbral de activación se encuentra en 0.



El umbral o bias suele tomarse como $x_0 = -1$ para generalizar la expresión de la función de activación a un umbral diferente de cero y que la expresión final quede "más simple". Veamos cómo es esto:

Así, "más simple" significa que quede $t > \omega_0$, en lugar de $t > -\omega_0$.

Al considerar $x_0 = -1$, la regla de propagación ahora quedaría como sigue:

$$t = \sum_{j=1}^{m} \omega_j x_j - \omega_0$$

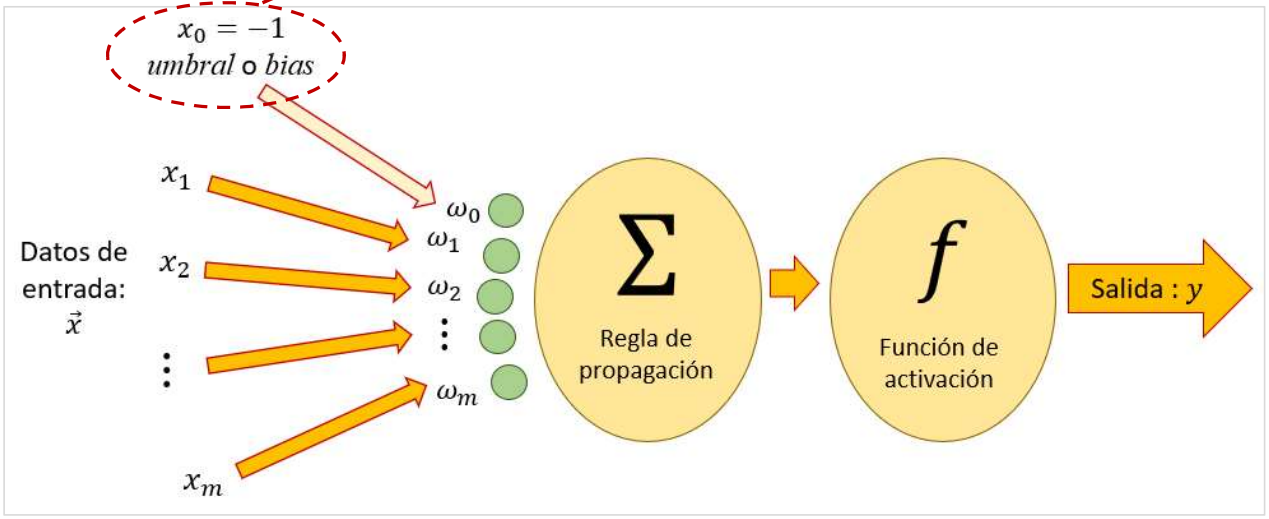
En particular la condición $t > 0$ con la regla de propagación, se puede expresar ahora como:

$$t = \sum_{j=1}^{m} \omega_j x_j - \omega_0 > 0$$

O bien:

$$t + \omega_0 = \sum_{j=1}^{m} \omega_j x_j > \omega_0$$

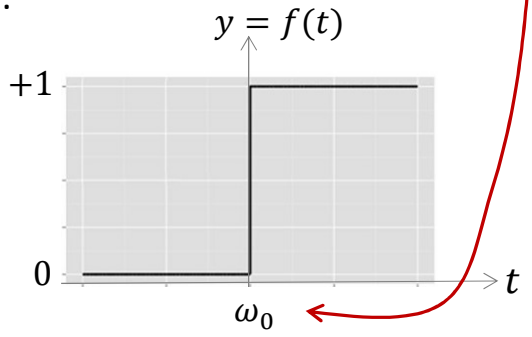
Es decir, la condición $t > 0$, ahora sería $t + \omega_0 > \omega_0$



Y con un cambio de variable la "nueva" función de activación puedea expresarse ahora como:

$$y = f(t) = \begin{cases} +1 & si \quad t > \omega_0 \\ 0 & si \quad t \leq \omega_0 \end{cases}$$

Y su gráfica:



Función de Activación

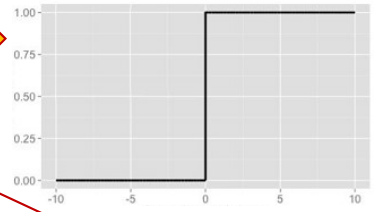
Existe una gran variedad de funciones de activación. Algunas de ellas son las siguientes:

Funciones Signo y Umbral

Neurona de McCulloch-Pitts

Neurona Hopfield o *de signo*

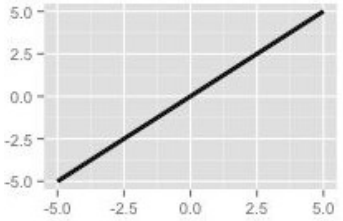
Escalón



Rango: $\{0, +1\}$

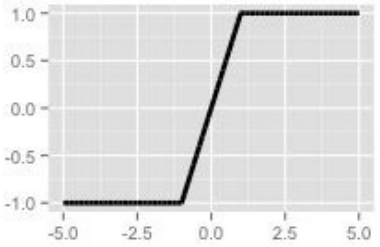
Rango: $\{-1, +1\}$

Lineal



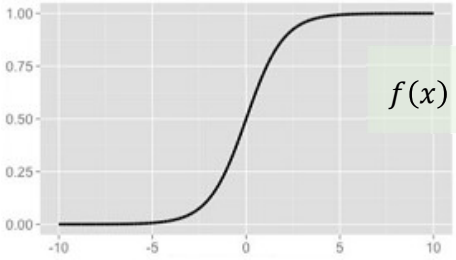
Rango: $(-\infty, +\infty)$

Lineal Seccionada



Rango: $[-1, +1]$

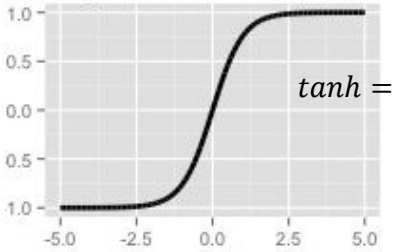
Sigmoide (exponencial)



$$f(x) = \frac{1}{1 + e^{-x}}$$

Rango: $(0, +1)$

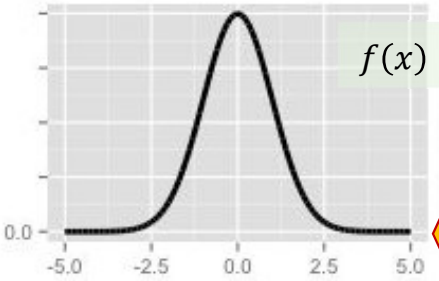
Sigmoide (tangente hiperbólica)



$$tanh = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$$

Rango: $(-1, +1)$

Gaussiana



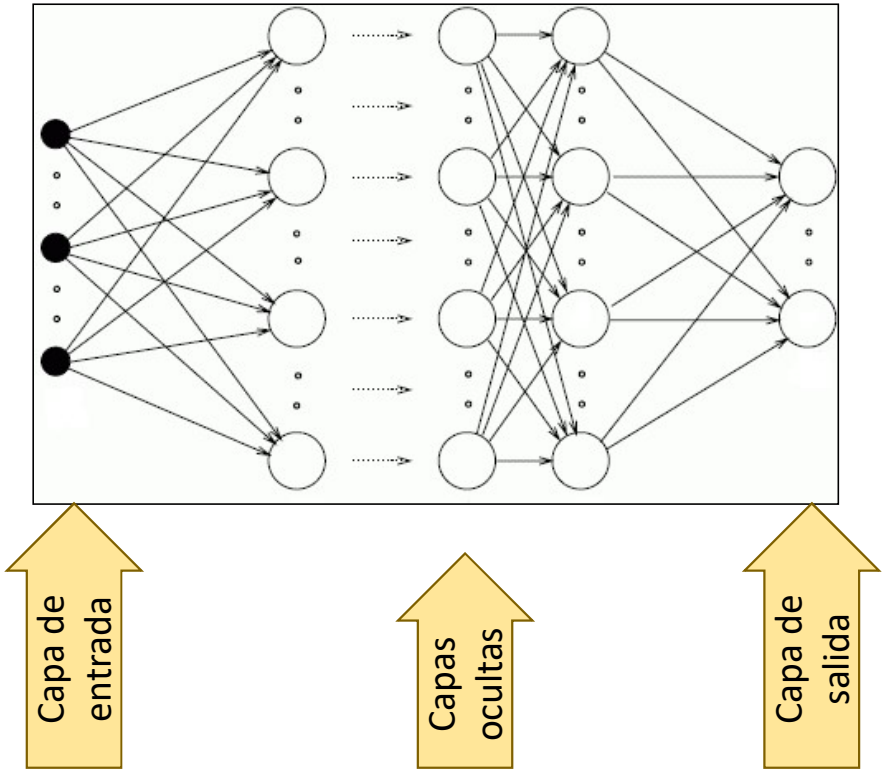
$$f(x) = e^{-x^2}$$

Rango: $(0, +1]$

Función de Base Radial

Red Neuronal Artificial: Topología

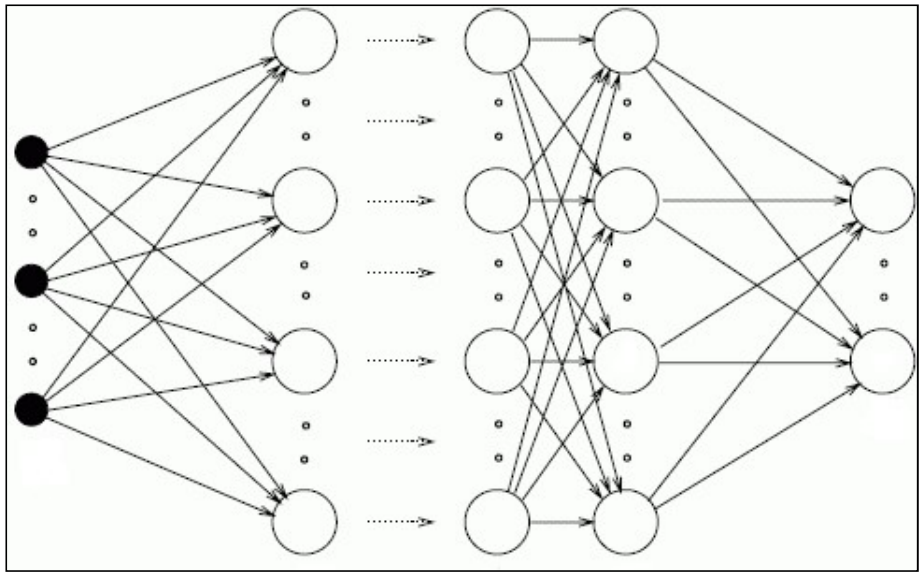
(*ANS*: *Artificial Neural Network*)



Aunque actualmente hay una gran variedad de arquitecturas. Podemos mencionar las siguientes tres características básicas que todas comparten:

- El número de capas que hay en la red:
  capa de entrada, capa de salida, capas ocultas.
- La forma en que la información dentro de la red se propaga: hacia adelante (*feedforward*) y hacia atrás (*feedback* y *backpropagation*).
- EL número de neuronas (nodos) en cada capa.

El número de nodos (neuronas) en la capa de entrada está determinado por el mismo número de factores considerados en los datos de entrada, más la neurona del *bias* (coordenada homogénea o umbral).



El número de nodos (neuronas) en la capa de salida, está determinado por la variable de predicción del problema de regresión o clasificación.

No hay una regla general para determinar el número de neuronas que debiera haber en las capas ocultas, aunque existen varias reglas empíricas.

Teorema de la Aproximación Universal (1989)

Sea φ la función de activación sigmoide, y $f \in C(I_n)$, entonces:

$$f \sim \sum_{k=0}^{N} \propto_k \varphi(w^T x)$$

Es decir, las funciones sigmoide son densas en $C(I_n)$.

Donde $C(I_n)$ es el conjunto de todas las funciones continuas con dominio en el cubo *n*-dimensional.

Una de las versiones demostrada por George Cybenko en relación al alcance de las redes neuronales artificiales.

Este es uno de los llamados *teoremas de exitencia* en Matemáticas, donde nos dice que existe dicha solución, pero no cómo llegar a ella.

"Traducción" del teorema:

Una red neuronal con propagación hacia adelante, con una sola capa oculta y una *suficiente* cantidad finita de neuronas en dicha capa oculta, es suficiente para resolver *casi* cualquier problema de aprendizaje automático de la *vida real*.

En particular, el Perceptrón Multicapa (**MLP**: Multi-Layer Perceptron) con la función sigmoide es suficiente para resolver *casi* cualquier problema de aprendizaje automático.