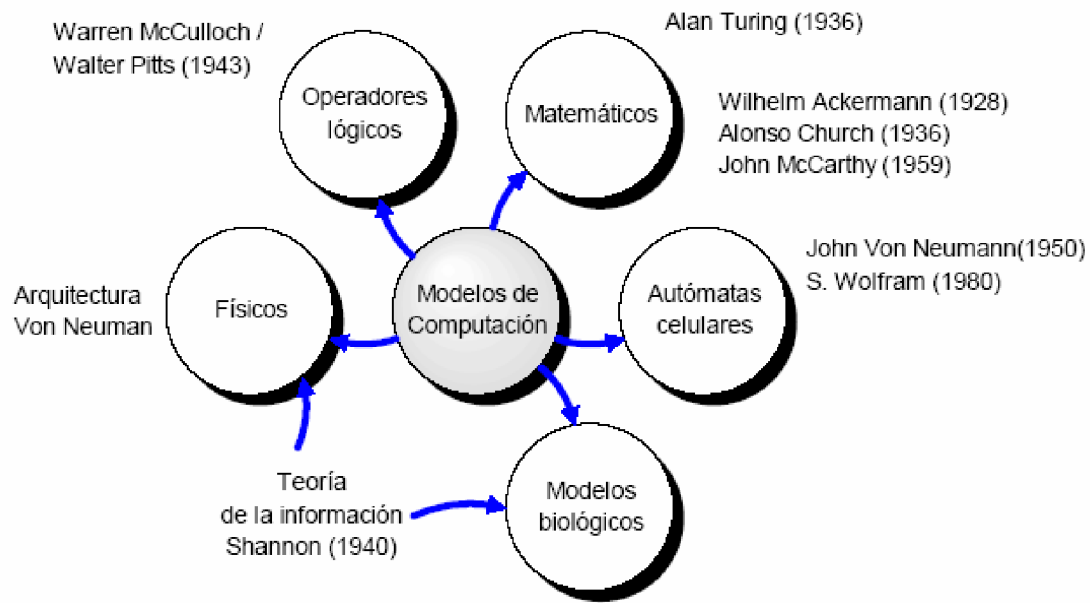


Cerebro Vs Máquinas

El procesamiento de información de carácter redundante, imprecisa y distorsionada posee un papel primordial para la resolución de problemas reales de clasificación o de predicción en muchas áreas científicas.



Modelos de *computación*

Una de las metodologías con un mayor auge en la última década son los modelos de redes neuronales (*Neural Networks*), que en esencia son estructuras formales de carácter matemático y estadístico con la propiedad del *aprendizaje*, es decir, la adquisición de conocimientos que en la mayoría de los casos es a partir de ejemplos.

Este aprendizaje se produce mediante un estilo de computación³ denominado en *paralelo* que intenta simular algunas de las capacidades que posee nuestro cerebro. Por esta razón se las definen como redes neuronales artificiales para distinguirlas de los modelos biológicos.

Los tres elementos clave de los sistemas biológicos que pretenden emular los artificiales son: el *procesamiento en paralelo*, la *memoria distribuida* y la *adaptabilidad*.

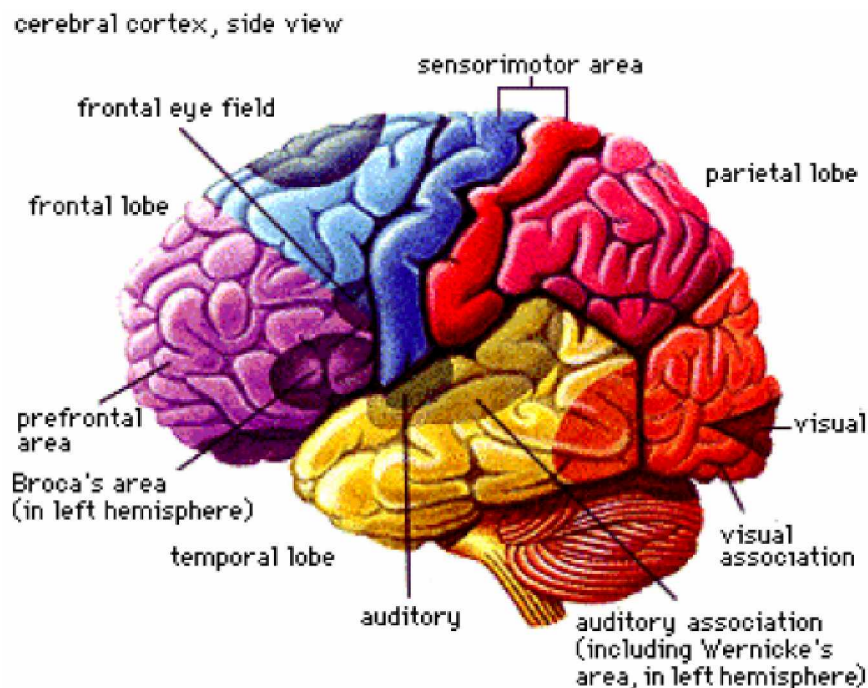
En referencia al primero de los elementos clave, es importante remarcar que aún siendo las neuronas biológicas más simples, lentas y menos fiables que las artificiales, el cerebro resuelve problemas complejos imposibles para sistemas simulados, a través de su trabajo en paralelo.

En segundo lugar, la memoria distribuida permite a los modelos biológicos ser tolerantes a los fallos, debido a que muchas neuronas pueden realizar tareas similares produciéndose intercambios de funciones. Y por último, la adaptabilidad garantiza el proceso de aprendizaje.

Pero en rigor, debemos notar que existen otras metodologías que permiten atajar los mismos problemas, como por ejemplo, los sistemas borrosos (*Fuzzy systems*), algoritmos genéticos (*Genetic algorithms*), sistemas expertos (*Expert systems*), etc. Todas estas metodologías conforman un tronco común definido como Inteligencia Artificial⁶ (*Artificial Intelligence*) de clara inspiración biológica, cuya finalidad es replicar parcialmente la naturaleza de la inteligencia humana apostando en muchos casos por estrategias mixtas que potencien aún más su desarrollo.

El concepto filosófico de inteligencia artificial se debe a un gran matemático, Alan Turing. Este lógico y matemático sometió a debate, desde la óptica filosófica, si las máquinas podían o no pensar.

Ideó un experimento de laboratorio para demostrar tal afirmación, que denominó "*prueba de Turing*". Tal experimento consistió en considerar que una máquina posee inteligencia si puede ser confundida por un ser humano. Ahora bien, el campo de la inteligencia artificial (AI) propiamente dicho recibió su nombre de John McCarthy. Fue el primer investigador que logró reunir a principios de los años 60 a todas aquellas personas interesadas en la inteligencia de los computadores. McCarthy realizó aportaciones de gran importancia, como por ejemplo, la invención del LISP (lenguaje de programación) y la creación del primer sistema de tiempo compartido.



Zonas del Córtex humano

El cerebro humano es una red biológica muy compleja con cientos de millones (10^{12}) de células¹² de varios tipos. La neurona biológica, como caso particular, posee la propiedad conductiva de señales eléctricas o químicas, que permiten formar una red, en la que la interconexión es muy alta. El número aproximado de neuronas del cerebro humano es de cien mil millones, las cuales están interconectadas entre sí, generando un volumen de conexiones en promedio del orden de diez mil para cada una de ellas, lo que lo convierte en un sistema con una complejidad muy superior a cualquier producto humano. El proceso de comunicación entre neuronas se produce gracias a los neurotransmisores liberados por la neurona emisora y captados por la neurona

receptora. Este último aspecto permite afirmar que gracias a este número tan elevado de conexiones (del orden de 10^{14}), puede considerarse como una gran red neuronal con fuerte especialización funcional entre las diferentes partes del *córtex*.

A continuación se detallan algunos de los principios computacionales más importantes muy relacionados con detalles de la organización del cerebro, que posteriormente se utilizaron en el diseño de los modelos artificiales.

1	<i>Paralelismo masivo</i> , las operaciones de cálculo se realizan al mismo tiempo en diferentes regiones del <i>córtex</i> y en cada región funcional del mismo, el cálculo lo realizan simultáneamente todas las neuronas.
2	Alto grado de <i>complejidad en las conexiones</i> definidas entre las neuronas.
3	<i>Habilidad para el aprendizaje</i> , es decir, existe una adaptación constante de las neuronas a partir de la experiencia acumulada.
4	<i>Estados binarios y variables continuas</i> . Las neuronas poseen dos estados, activado o desactivado y la información continua está codificada por la frecuencia de pulsos. (Pulsos por unidad de tiempo)
5	<i>Numerosos tipos de neuronas y de señales</i> .
6	Interacción entre neuronas de <i>carácter no lineal</i> .
7	El cerebro se organiza <i>descomponiendo</i> las conexiones en subredes.
8	Asociado al anterior aspecto cada subred tiene una <i>función</i> específica.

El diseño de las redes neuronales artificiales (véase ilustración, también llamadas sistemas de procesamiento en paralelo (*Parallel distributed processing systems* (PDPs)) o sistemas conexionistas (*Connectionist systems*), incorporan características biológicas generando un claro paralelismo entre estos y el modelo neuronal biológico. Así conceptos, como por ejemplo, *dendritas*, *axón*, *sinapsis* y cuerpo o *soma* que son de uso corriente en el entorno biológico, toman un papel relevante en los modelos matemáticos que replican la forma de transmisión de la señal eléctrica que por ellas circula. En 1901 *Santiago Ramón y Cajal* postuló que la dirección de la transmisión de la información es determinada por la naturaleza en forma de red de las células nerviosas, a partir de los conocimientos sobre la naturaleza eléctrica de los impulsos nerviosos y de la velocidad de su transmisión.

En la actualidad el crecimiento más importante se sitúa en el entorno de las Aplicaciones, en especial, los campos relacionados con la industria y la empresa. Dicho desarrollo está supeditado al avance científico paralelo en el conocimiento del comportamiento de la neurona biológica, permitiendo su posterior implementación en los modelos artificiales.

El incremento constante en el número de aplicaciones está provocando un impacto trascendente en la industria en general, específicamente sectores como por ejemplo, sector bancario, asegurador, etc. A través de estas aplicaciones las ventajas competitivas son claras en términos de productividad.

Smith y Gupta (2000) consideran que el desarrollo histórico de la investigación en este campo posee cinco fases vinculado a su impacto en la industria, la primera de ellas, desde 1834 a 1943, coincide con la creación de IBM, la segunda, de 1946 a 1957, donde merece resaltar por parte de General Electric Co, la creación del primer sistema computerizado en paralelo, en tercer lugar, de 1969 a 1982, aparece el primer microprocesador de Intel Corp, se crean las empresas SPSS, Inc., SAS Institute y Apple Computer Corporation, etc, en cuarto lugar, desde 1983 a 1990, resaltamos el informe DARPA, la creación de la empresa NeuralWare Inc y en la última fase, 1990 hasta hoy en día, la investigación continua con fuerza pero más orientada a la generación de aplicaciones que solucionen problemas reales.

La Estructura Neuronal

INTRODUCCIÓN

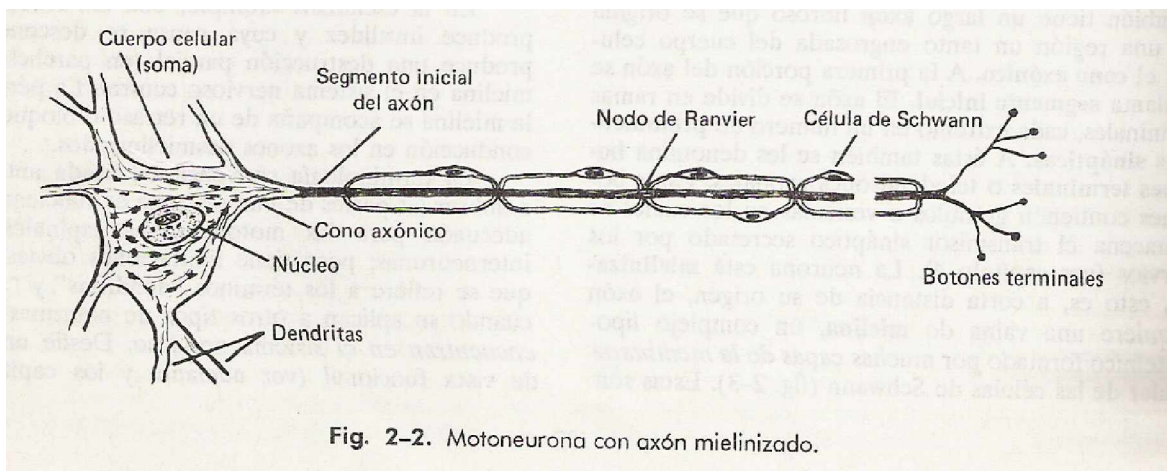
El sistema nervioso humano contiene alrededor de 10^{12} neuronas (un billón). También posee de 10 a 50 veces este número de células gliales. Las neuronas, bloques básicos del edificio del sistema nervioso, han evolucionado de las células neuroefectoras primitivas que responden a varios estímulos por contracción. En animales más complejos, la contracción se ha convertido en la función especializada de las células musculares. En tanto que la integración y transmisión de los impulsos nerviosos se han convertido en las funciones especializadas de las neuronas.

CÉLULAS NERVIOSAS

Morfología

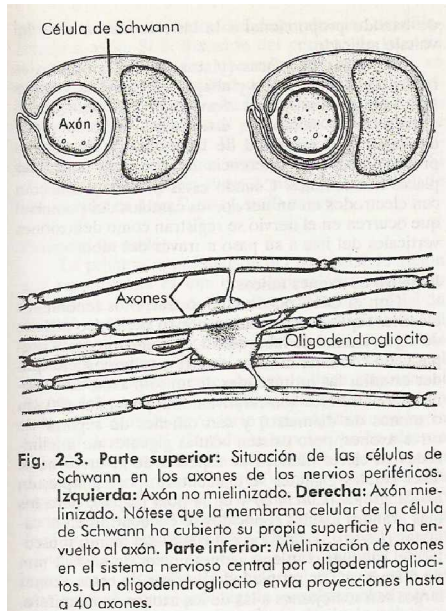
Las neuronas del sistema nervioso del cerebro tienen formas y tamaños diferentes.

Sin embargo, la mayoría tienen las mismas partes como la típica neurona motora espinal.



Esta célula posee de cinco a siete procesos llamados dendritas que se extienden hacia el exterior del cuerpo neuronal y se arborizan extensamente. También posee un largo axón fibroso que se origina en una región un tanto engrosada del cuerpo celular, el cono axónico. A la primera porción del axón se le llama segmento inicial. El axón se divide en ramas terminales, cada extremo en un número de prominencias sinápticas. A éstas también se les denomina botones terminales o telodendrones axonales. Estos botones contienen gránulos o vesículas en los cuales se almacena el transistor sináptico secretado por los nervios. La neurona está mielinizada, esto es, a corta distancia de su origen, el axón adquiere una vaina de mielina, un complejo lipoproteínico formado por muchas capas de la membrana celular de las células de Schwann.

Estas son células similares a la glía y se encuentran a lo largo de los nervios periféricos. La vaina de mielina envuelve el axón excepto en su terminación y en los nodos de Ranvier, constricciones periódicas de 1 μm con una separación entre ella de aproximadamente 1 mm. No todas las neuronas de mamíferos están mielinizadas; algunas son amielínicas; esto es, están simplemente rodeadas por células de Schwann sin la envoltura de las membranas de éstas alrededor del axón la cual produce mielina. La mayor parte de las neuronas de los invertebrados son no mielinizadas; pero las células que forman la mielina son oligodendroglíocitos en lugar de células de Schwann.



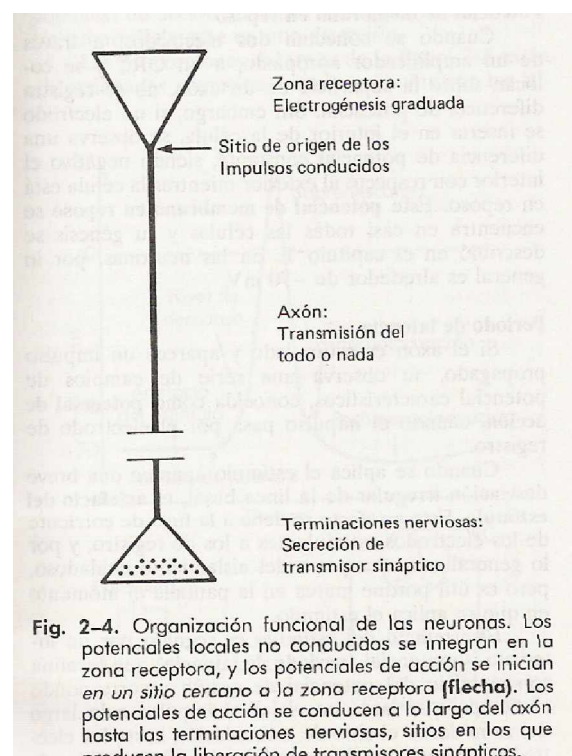
Es más, a diferencia de las células de Schwann, que forman la mielina existente entre dos nódulos de Ranvier sobre una sola neurona, los oligodendroglíocitos emiten múltiples proyecciones que forman la mielina sobre numerosas células circunvecinas.

Desde el punto de vista funcional, las neuronas tienen cuatro zonas importantes. Hay una zona receptora o dendrítica, en la que se integran los cambios múltiples locales de potencial generados por las conexiones sinápticas; un sitio en el que se generan los potenciales de acción propagados; un proceso axoniano que transmite los impulsos propagados hasta las terminaciones nerviosas; y las propias terminaciones nerviosas, en las que los potenciales de acción hacen que se liberen los transmisores sinápticos. El cuerpo celular a menudo

está situado en el extremo de la zona dendrítica del axón (por ejemplo, en las neuronas auditivas) o unido a él lateralmente (por ejemplo, en las neuronas cutáneas). Su situación no importa en lo que se refiere a la función receptora de la zona dendrítica y la función transmisora del axón. Debe notarse que la integración de la actividad no es la única función que las dendritas. Algunas neuronas del sistema nervioso central no tienen axones y los potenciales locales pasan de una dendrita a otra.

Excitación

La célula nerviosa tiene un umbral bajo de excitación; los estímulos pueden ser eléctricos, químicos o mecánicos. Se producen dos tipos de modificaciones fisicoquímicas, a saber: potenciales locales no propagados, que según su localización, reciben el nombre de potenciales sinápticos, generadores o electrotónicos; y cambios propagados, es decir, los potenciales de acción o impulsos nerviosos. Estas son las únicas respuestas de las neuronas y de otros tejidos excitables, y son el lenguaje general del sistema nervioso. Estas se deben a cambios en la conducción de iones a través de la membrana celular, que son producidos por alteraciones en los canales iónicos.



El impulso es normalmente transmitido (conducido) a lo largo del cilindro eje hasta su terminación. Los nervios no son "hilos telefónicos" que transmiten los impulsos pasivamente; la conducción de los impulsos nerviosos, aunque rápida, es muchísimo más lenta que la de la electricidad. Ciertamente, el tejido nervioso es un conductor pasivo relativamente malo y sería necesario un potencial de muchos voltios para producir una señal de una fracción de 1 voltio en el otro extremo del axón de 1 metro de longitud si faltasen los procesos activos en él. La conducción es un fenómeno activo, autopropagado, y el impulso se desplaza a lo largo de él con velocidad y amplitud constantes. El proceso a menudo se compara con lo que pasa cuando se aplica una cerilla al extremo de un reguero de pólvora. Al encenderse las partículas inmediatas a la cerilla, la llama se mueve sin parar por todo el camino hasta su fin.

FENÓMENOS ELÉCTRICOS EN LAS CÉLULAS NERVIOSAS.

Potencial de membrana en reposo

Si un electrodo se inserta en el interior de la célula, se observa una diferencia de potencial constante, siendo negativo el interior con respecto al exterior mientras la célula está en reposo. En las neuronas, por lo general es alrededor de -70mV .

Periodo de latencia

Si el axón es estimulado y aparece un impulso propagado, se observa una serie de cambios de potencial característicos, conocida como potencial de acción.

Cuando se aplica el estímulo aparece una breve desviación irregular de la línea basal y es seguido de un intervalo isopotencial (periodo de latencia) que termina con el inicio del potencial de acción y corresponde al tiempo que tarda en viajar el impulso a lo largo del axón desde el sitio de estimulación hasta los electrodos de registro. Su duración es proporcional a la distancia entre los electrodos estimulantes y los de registro, así como a la velocidad de conducción del cilindro eje o axón.

Potencial de acción

La primera manifestación del potencial de acción que se aproxima es una despolarización inicial de la membrana. Después de una despolarización inicial de 15 mV , aumenta la velocidad de despolarización. El punto en el cual ocurre este cambio se llama nivel de descarga (disparo). De ahí en adelante, el trazo en el osciloscopio rápidamente alcanza la línea isopotencial (potencial cero) y la sobrepasa aproximadamente en $+35\text{ mV}$. Entonces se revierte y cae rápidamente hacia el nivel de reposo. Cuando la despolarización alcanza aproximadamente 70%, la velocidad de repolarización disminuye y el trazo se acerca al nivel de reposo más lentamente. El ascenso brusco y el descenso rápido forman el potencial de espiga del axón, y la caída lenta final es la postdespolarización. Después de que alcanza el valor de reposo previo, el trazo asciende ligeramente en la dirección de la hiperpolarización para dar una pequeña pero prolongada posthiperpolarización. Cuando se registra con un electrodo en la célula el potencial de acción se llama monofásico porque ocurre primordialmente en una dirección.

Ley del "todo o nada"

Si un axón se dispone para registro, con los electrodos de registro a una distancia considerable de los electrodos estimulantes, es posible determinar la intensidad mínima de la corriente estimulante (intensidad umbral) capaz de producir un impulso. Este umbral varía con las condiciones experimentales y el tipo

de axón; pero una vez se alcanza se produce un potencial de acción completo. Los incrementos ulteriores en la intensidad del estímulo no producen incrementos u otros cambios en el potencial de acción mientras las condiciones experimentales permanezcan constantes. El potencial de acción deja de presentarse si el estímulo es de magnitud subumbral; pero aparece con una forma y amplitud constantes, sin que importe la intensidad del estímulo, siempre y cuando sea de intensidad umbral o mayor. El potencial de acción es, por tanto de carácter "todo o nada" y se dice que obedece a la ley del "todo o nada".

PROPIEDADES DE LOS NERVIOS MIXTOS

Los nervios periféricos de los mamíferos están hechos de muchos axones unidos dentro de una envoltura fibrosa llamada epineuro; debido a esto, los cambios de potencial registrados en tales nervios representan la suma algebraica de todos los potenciales de acción, del tipo de todo o nada, de muchos cilindroejes. Los umbrales de los axones individuales del nervio, así como su distancia de los electrodos estimulantes varían. Cuando los estímulos son subumbrales, ningún axón es estimulado y no se provoca una respuesta; pero si aquéllos son de mayor intensidad, los axones con umbrales bajos descargan y se observa un pequeño cambio de potencial. Cuando la intensidad de la corriente estimulante es aún mayor, los axones de umbral más alto también descargan y la respuesta eléctrica crece proporcionalmente hasta que el estímulo es suficientemente intenso para excitar a todos los cilindroejes del nervio. El estímulo que produce la excitación de todos los axones del nervio es el estímulo máximo y la aplicación de estímulos todavía mayores, supramáximos, ya no produce un incremento en el tamaño del potencial observado.

Potenciales de acción compuestos

Otra característica de los nervios mixtos, en oposición con los cilindroejes únicos, es la aparición de múltiples picos en el potencial de acción, por lo cual se llama potencial de acción compuesto. Su forma se debe al hecho de que un nervio mixto está compuesto de familias de fibras con diferentes velocidades de conducción. Por lo tanto, cuando todas las fibras son estimuladas, la actividad de las fibras de conducción rápida llega más pronto a los electrodos de registro que la actividad de las fibras más lentas, y mientras más lejos se registre el potencial de acción de los electrodos estimulantes mayor será la separación entre los picos de las fibras rápidas y lentas. El número y tamaño de los picos varía con los tipos de fibras del nervio particular en estudio. Si se emplean estímulos menos intensos que el máximo, la forma del potencial de acción compuesto también depende del número y tipo de fibras estimuladas.

TIPOS DE FIBRAS NERVIOSAS

Cuadro 2-2. Tipos de fibras nerviosas en los nervios de los mamíferos

Tipo de fibra	Función	Diámetro de la fibra (μm)	Velocidad de conducción (mseg)	Duración de la espiga (mseg)	Periodo refractario absoluto (mseg)
A α	Propiocepción; cinestesia	12-20	70-120	0.4-0.5	0.4-1
β	Tacto, presión	5-12	30-70		
γ	Motora para los husos musculares	3-6	15-30		
δ	Dolor, frío, tacto	2-5	12-30		
B	Autónomas preganglionares	< 3	3-15	1.2	1.2
C f.r.d.*	Dolor, temperatura, algo de mecanorecepción respuestas reflejas	0.4-1.2	0.5-2	2	2
f.s.†	Simpáticas posganglionares	0.3-1.3	0.7-2.3	2	2

* Fibras de las raíces dorsales.

† Fibras simpáticas.

LA SINAPSIS

Los impulsos son transmitidos de una célula nerviosa a otra en la sinapsis. Estas son las uniones donde el axón o alguna otra porción de la célula (la célula presináptica) termina en las dendrita, soma o axón de otra neurona fglandular (célula postsináptica). Vale la pena observar que las dendritas, así como los axones, pueden ser presinápricas o postsinápricas. La transmisión en la mayor parte de las uniones sinápticas es química; el impulso en el axón presináptico provoca la secreción de neurotransmisores, como acetilcolina o serotonina. Este mediador químico se une a los receptores sobre la superficie de la célula postsináptica, lo cual desencadena los fenómenos intracelulares que abren o cierran los conductos en la membrana de la célula postsináptica. Sin embargo, en alguna de las sinapsis, la transmisión es eléctrica, y en algunas sinapsis mixtas es eléctrica y química. De cualquier manera, la transmisión no es un simple salto de un potencial de acción de una célula presináptica a la postsináptica. Los efectos de descarga en las terminaciones sinápticas individuales pueden ser excitatorios o inhibitorios y, cuando la célula postsináptica es un neurona, la suma de todos los efectos excitatorios o inhibitorios determina si se genera un potencial de acción. Por tanto, la transmisión sináptica es un proceso complejo que permite la graduación y ajuste de la actividad neural necesaria para la función normal.

Desarrollo sináptico

Un planteamiento fascinante que ha llama do la atención consiste en la manera en que, durante el desarrollo, las neuronas encuentran los blancos "correctos" y establecen las conexiones sinápticas "correctas". En la actualidad, se considera que participan diversos factores. Está establecido que las señales químicas permiten que las neuronas reconozcan sus blancos e ignoren a las otras neuronas de la misma región, y que existen interacciones trópicas entre las células pre y postsinápticas una vez que se han establecido las conexiones. Sin embargo se ha demostrado que, en muchas situaciones, numerosas neuronas o músculos esqueléticos y que, a continuación, desaparecen conexiones "inaporopiadas" y dejan sólo las conexiones que se encuentran en el adulto. Son muchas la neuronas que mueren durante el desarrollo, y quizá las que persisten sean sólo las neuronas y uniones más activas. Además, existe competencia entre las neuronas por los sitios sinápticos y en las áreas del cerebro que se han desnervado crecen terminaciones nerviosas a partir de neuronas adyacentes.

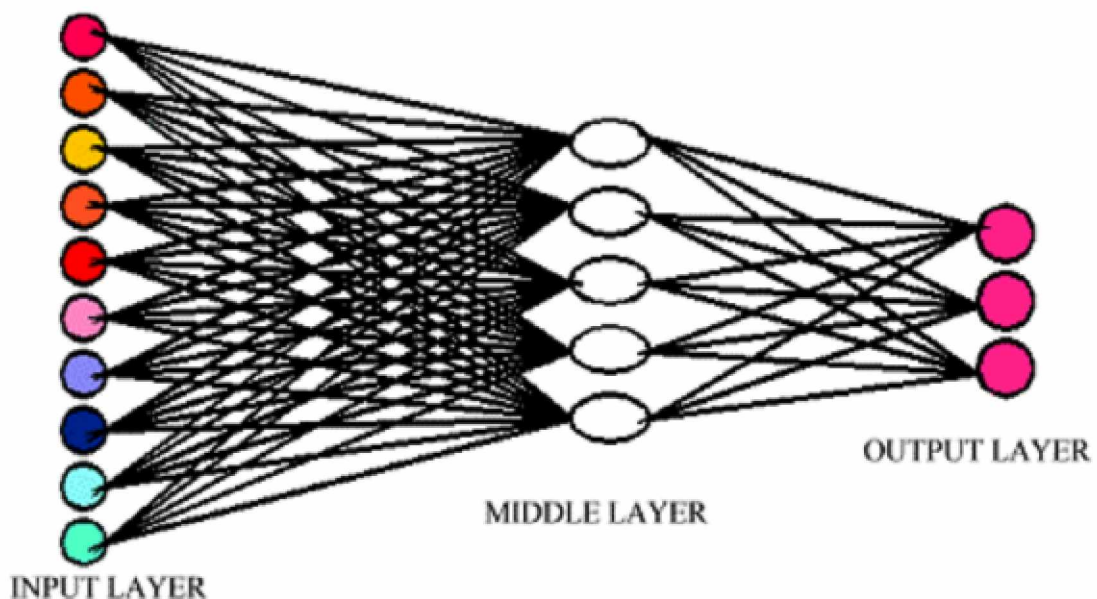
La Neurona Artificial

INTRODUCCIÓN

Las Redes Neuronales Artificiales (ANNs) son una simulación abstracta de un sistema nervioso real que está formado por un conjunto de unidades neuronales conectadas unas con otras por medio de conexiones del tipo axón. Estas conexiones tienen una gran semejanza con las dendritas y los axones en los sistemas nerviosos biológicos.

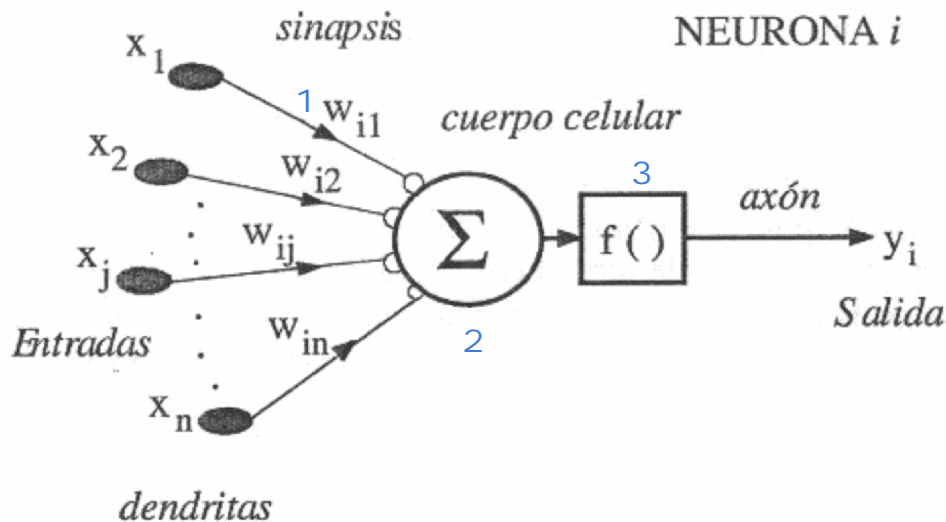
Entre los pioneros en el modelado de neuronas se encuentra Warren McCulloch y Walter Pitts. Estos dos investigadores propusieron un modelo matemático de neurona. En este modelo cada neurona estaba dotada de un conjunto de entradas y salidas. Cada entrada está afectada por un peso. La activación de la neurona se calcula mediante la suma de los productos de cada entrada y la salida es una función de esta activación. La principal clave de este sistema se encuentra en los pesos de las diferentes entradas. Como se ha visto, las entradas son modificadas por el peso y las salidas son función de estas modificaciones. Esto nos lleva a concluir que los pesos influyen de forma decisiva en la salida y por lo tanto pueden ser utilizados para controlar la salida que se desea.

En realidad cuando se tienen interconectadas muchas de estas neuronas artificiales lo que se hace inicialmente es entrenar el sistema. El entrenamiento consiste en aplicar unas entradas determinadas a la red y observar la salida que produce. Si la salida que produce no se adecua a la que se esperaba, se ajustan los pesos de cada neurona para interactivamente ir obteniendo las respuestas adecuadas del sistema. A la red se le somete a varios ejemplos representativos, de forma que mediante la modificación de los pesos de cada neurona, la red va "aprendiendo".



DEFINICIÓN

Una **neurona artificial** es un procesador elemental, en el sentido de que procesa un vector x (x_1, x_2, \dots, x_N) de entradas y produce una respuesta o salida única. Los elementos clave de una neurona artificial son los siguientes:



Las **señales de entrada** a una neurona artificial x_1, x_2, \dots, x_n son variables continuas en lugar de pulsos discretos, como se presentan en una neurona biológica. Mediante estas entradas recibimos los datos de otras neuronas o de fuentes externas.

Pesos sinápticos (1) de la neurona i , w_{ij} que representan la intensidad de interacción entre cada neurona presináptica " j " y la neurona postsináptica " i ". Al igual que en una neurona biológica se establecen sinápsis entre las dendritas de una neurona y el axón de otra, en una neurona artificial a las entradas que vienen de otras neuronas se les asigna un peso, un factor de importancia. Este peso, que es un número, se modifica durante el entrenamiento de la red neuronal, y es aquí por tanto donde se almacena la información que hará que la red sirva para un propósito u otro. Dada una entrada positiva (procedente de un sensor o simplemente la salida de otra neurona), si el peso es positivo tenderá a excitar a la neurona postsináptica, si el peso es negativo tenderá a inhibirla. Así se habla de sinapsis excitadoras (de peso positivo) e inhibidoras (de peso negativo).

Una **regla de propagación** (2). Con esas entradas y los pesos sinápticos, se suele hacer algún tipo de operación para obtener el valor del potencial postsináptico o entrada neta h_i (valor que es función de las entradas y los pesos y que es el que se utilizan en último término para realizar el procesamiento).

Una **función de salida** (3) $f_i(h_i(t))$, que proporciona la salida actual $y_i(t) = f_i(h_i(t))$ de la neurona " i " en función de su estado de activación. Dependiendo del tipo de salida, las neuronas suelen recibir nombres específicos. Así, las neuronas estándar cuya salida sólo puede tomar los valores "0" ó "1" se suelen denominar genéricamente neuronas de tipo *McCulloch Pitts*, mientras que aquellas que únicamente pueden tener por salidas (-1 ó $+1$) se suelen denominar neuronas tipo *Ising* (debido al paralelismo con los modelos físicos de partículas con espín que adoptan únicamente dos estados, hacia arriba y hacia abajo). Si adopta diversos valores discretos en la salida (por ejemplo, $-2, -1, 0, +1, +2$), se dice que se trata de una neurona de tipo *Potts*. En ocasiones, el rango de los valores

que una neurona de salida continua puede proporcionar se limita a un intervalo definido, por ejemplo, $(0, +1)$ ó $(-1, +1)$.

LA ENTRADA NETA

En general, la entrada neta se calcula como:

$$h_i(t) = g(w_{ij}, x_j(t))$$

Ejemplos de funciones para calcular la entrada neta:

- § La función más habitual es de tipo **lineal**, y se basa en la **suma ponderada** de las entradas con los pesos sinápticos

$$h_i(t) = \sum_j w_{ij} x_j$$

- § Una regla de tipo **no lineal**, de uso más limitado, es la siguiente:

$$h_i(t) = \sum_{j_1 j_2 \dots j_p} w_{i, j_1 j_2 \dots j_p} x_{j_1} x_{j_2} \dots x_{j_p}$$

Implica una interacción de tipo multiplicativo entre las entradas de la neurona (como se observa realmente en determinadas sinápsis biológicas). El uso de esta última regla de propagación determina que una neurona se denomine de orden superior o **neurona sigma-pi** (por emplear sumas y productos), e implica una mayor complejidad, tanto en el estudio de la dinámica de la red neuronal, como en su realización hardware.

- § Otra regla de propagación habitual, especialmente en los modelos de RNA basados en el cálculo de distancias entre vectores (como RBF, mapas de Kohonen o LVQ), es la **distancia euclídea**

$$h_i^2(t) = \sum_j (x_j - w_{ij})^2$$

Representa la distancia (al cuadrado) existente entre el vector de entradas y el de pesos. Cuando ambos vectores son muy similares, la distancia es muy pequeña; cuando son muy diferentes, la distancia crece. Por tanto, este tipo de regla opera de manera diferente a las anteriormente comentadas.

Separar el concepto de regla de propagación y función de activación permite considerar desde un punto de vista unificado muchos modelos que de otra manera habría que tratar como casos especiales de una neurona estándar, (tal como sucede, por ejemplo, en el RBF o los mapas autoorganizados).

La Función de Activación

INTRODUCCIÓN

El valor obtenido con la regla de propagación, se filtra a través de una función conocida como función de activación y es la que nos da la salida de la neurona. Según para lo que se desee entrenar la red neuronal, se suele escoger una función de activación u otra en ciertas neuronas de la red.

En muchas ocasiones la razón para la aplicación de una función de activación distinta de la identidad surge de la necesidad de que las neuronas produzcan una salida acotada. Esto desde un punto de vista de similitud con el sistema biológico, no es tan descabellado, ya que las respuestas de las neuronas biológicas están acotadas en amplitud. Además cada neurona tiene asociado un número denominado **bías o umbral**, que puede verse como un número que indica a partir de que valor del potencial postsináptico la neurona produce una salida significativa.

DEFINICIÓN

La **función de activación** ó de transferencia proporciona el estado de activación actual $y_i(t)$ a partir del potencial postsináptico o entrada neta $h_i(t)$ y del propio estado de activación anterior $y_i(t-1)$

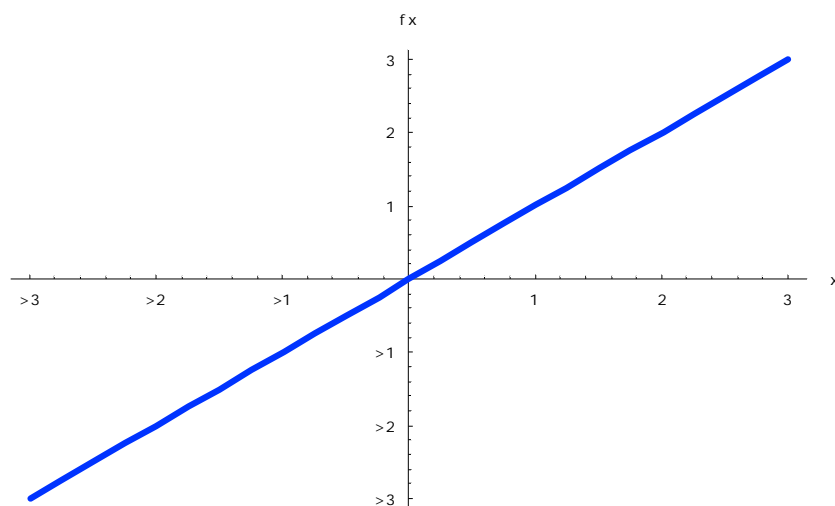
$$y_i(t) = f(h_i, y_i(t-1))$$

Sin embargo, en muchos modelos de sistemas neuronales artificiales (ANS) se considera que el estado actual de la neurona no depende de su estado anterior, sino únicamente del actual.

$$y_i(t) = f(h_i)$$

La función de activación $f(.)$ se suele considerar determinista, y en la mayor parte de los modelos es monótona creciente y continua, como se observa habitualmente en las neuronas biológicas.

La más simple de todas es la **función identidad** (que se puede generalizar al caso de una función lineal cualquiera), empleada, por ejemplo, en el Adaline.

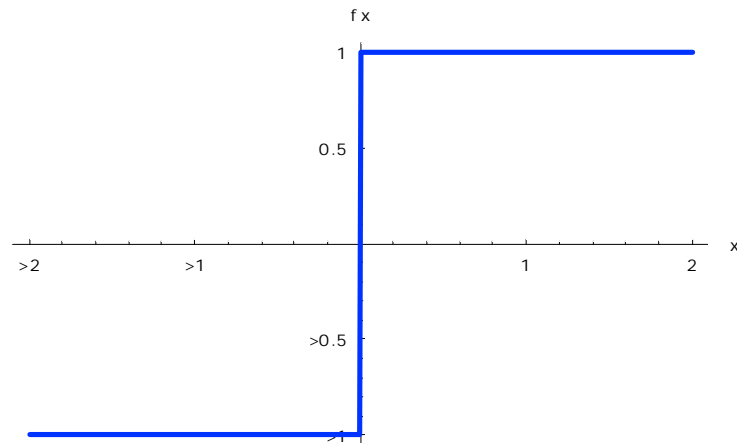


$$y = x$$

Rango :

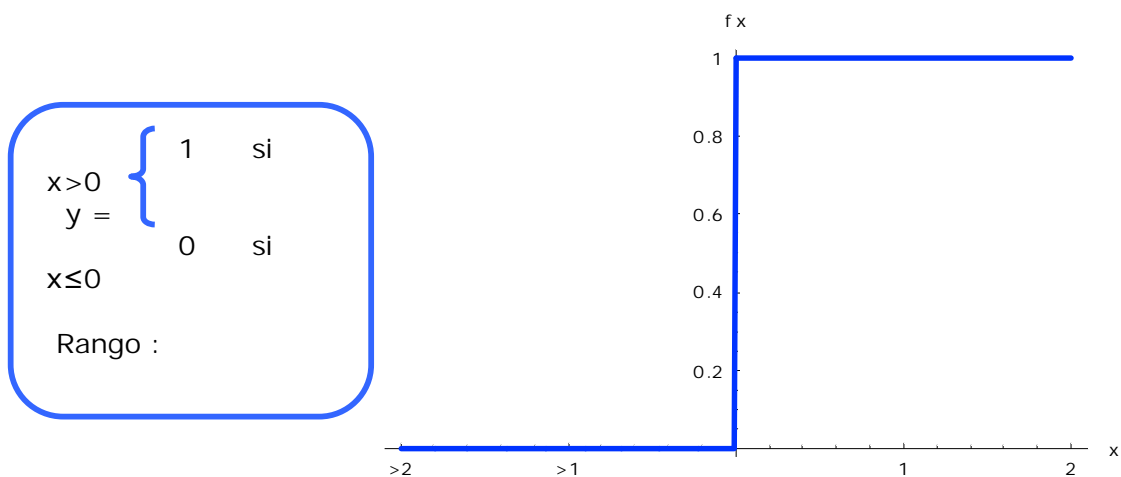
$$[-\infty, +\infty]$$

Otro caso simple es la **función escalón**, empleada en el Perceptrón Simple y en la red de Hopfield discreta, así como en la neurona clásica de Mc Culloch-Pitts. La función lineal a tramos se puede considerar como una lineal saturada en sus extremos, es de gran sencillez computacional y resulta más plausible desde un punto de vista biológico pues, las neuronas se activan más a mayor excitación, hasta saturarse a la máxima respuesta que pueden proporcionar.



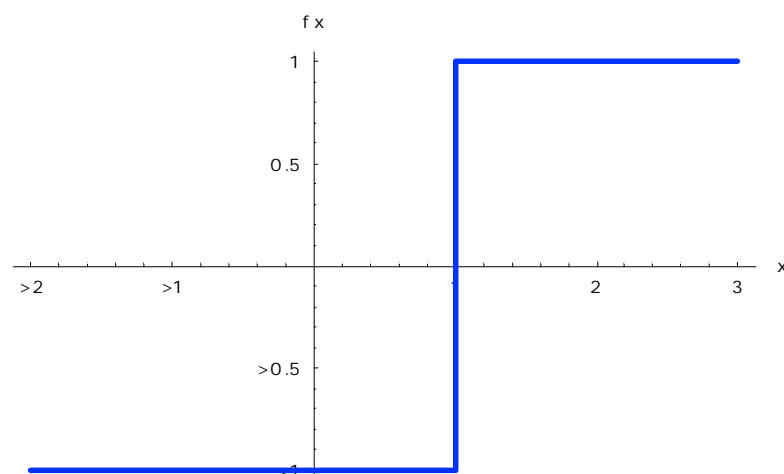
$$y = \begin{cases} 1 & \text{si } x > 0 \\ -1 & \text{si } x \leq 0 \end{cases}$$

Rango :



$$y = \begin{cases} 1 & \text{si } x > 0 \\ 0 & \text{si } x \leq 0 \end{cases}$$

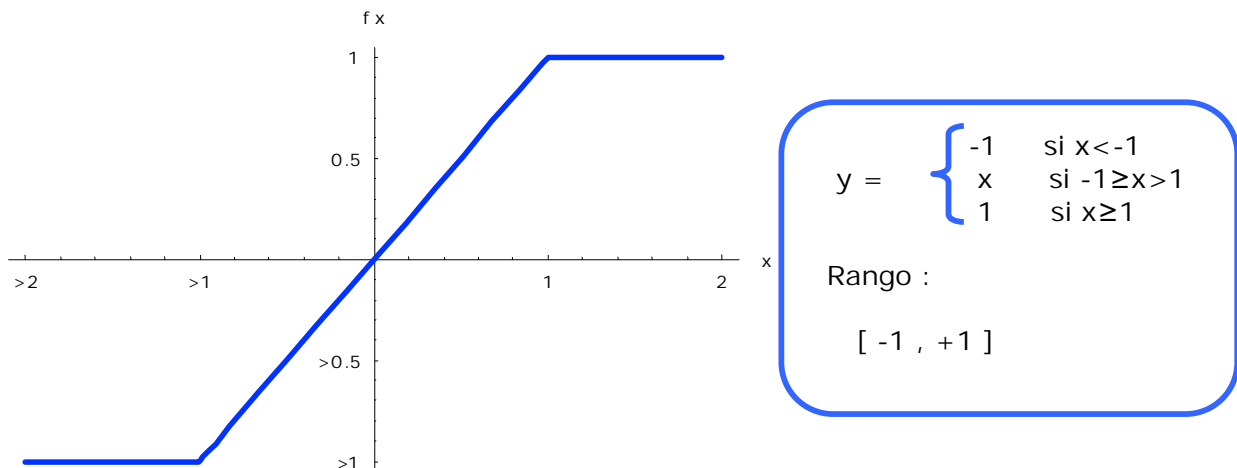
Rango :



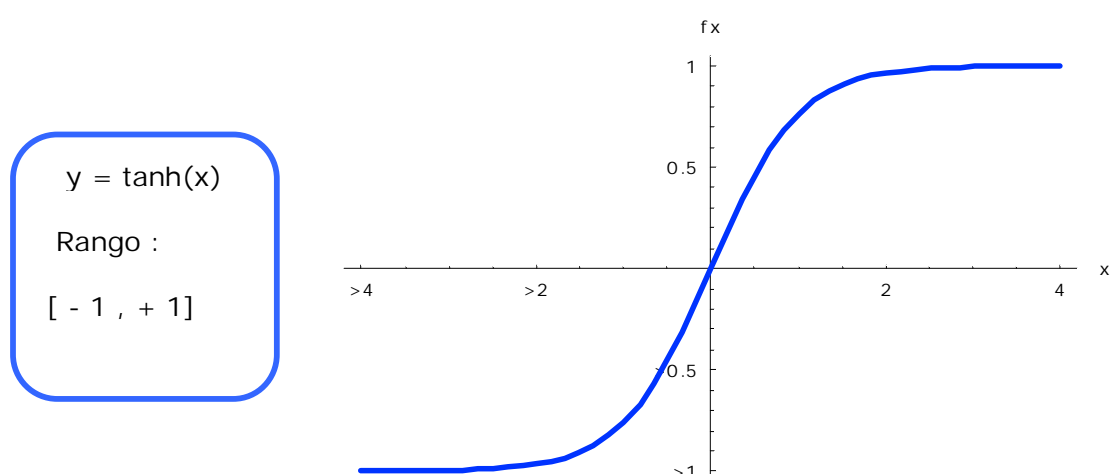
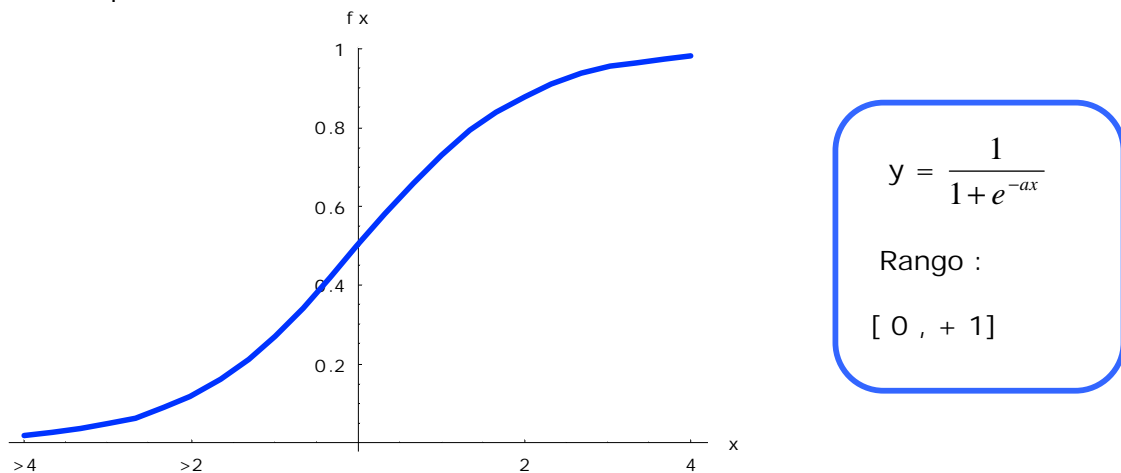
$$y = \begin{cases} 1 & \text{si } x > 1 \\ -1 & \text{si } x \leq 1 \end{cases}$$

Rango :

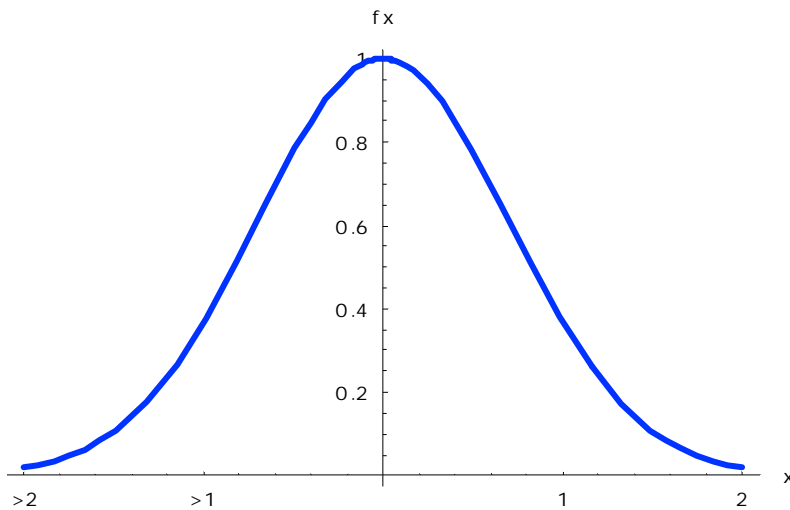
Con la función **lineal-mixta** el estado de activación de la unidad está obligado a permanecer dentro de un intervalo de valores reales prefijados



Los algoritmos de aprendizaje requieren que la función de activación cumpla la condición de ser derivable. Las más empleadas en este sentido son las funciones de tipo sigmoide, como la del BP. Con la función **sigmoide**, para la mayoría de los valores del estímulo de entrada, el valor dado por la función es cercano a uno de los valores asintóticos. Esto hace posible que en la mayoría de los casos, el valor de salida esté comprendido en la zona alta o baja del sigmoide. De hecho cuando la pendiente es elevada, esta función tiende a la función escalón. La importancia de ésta función es que su derivada es siempre positiva y cercana a cero para los valores grandes positivos o negativos; además toma su valor máximo cuando x es cero. Esto hace que se puedan utilizar las reglas de aprendizaje definidas para la función escalón, con la ventaja respecto a esta función, que la derivada está definida para todo el intervalo.



Otra función clásica es la [gaussiana](#), que se utiliza junto con reglas de propagación que involucran el cálculo de cuadrados de distancias (por ejemplo, la euclídea) entre los vectores de entradas y pesos.

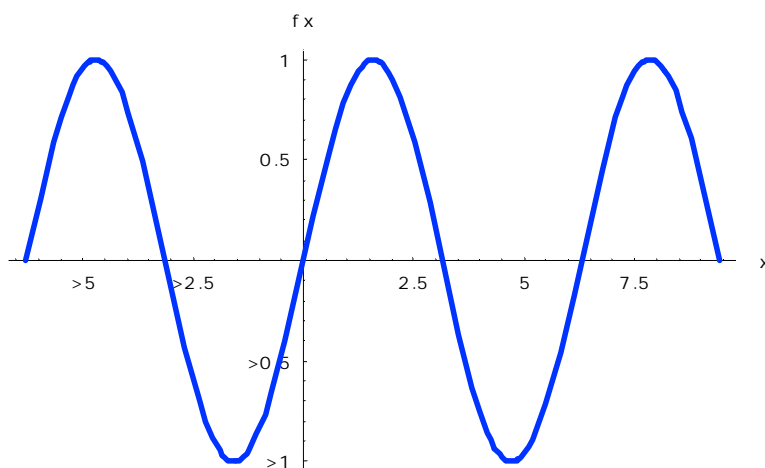


$$y = Ae^{-Bx^2}$$

Rango :

[0 , + 1]

Por último, en ocasiones se emplean funciones [sinusoidales](#), como en aquellos casos en los que se requiere expresar explícitamente una periodicidad temporal.



$$y = A\sin(\omega x + \varphi)$$

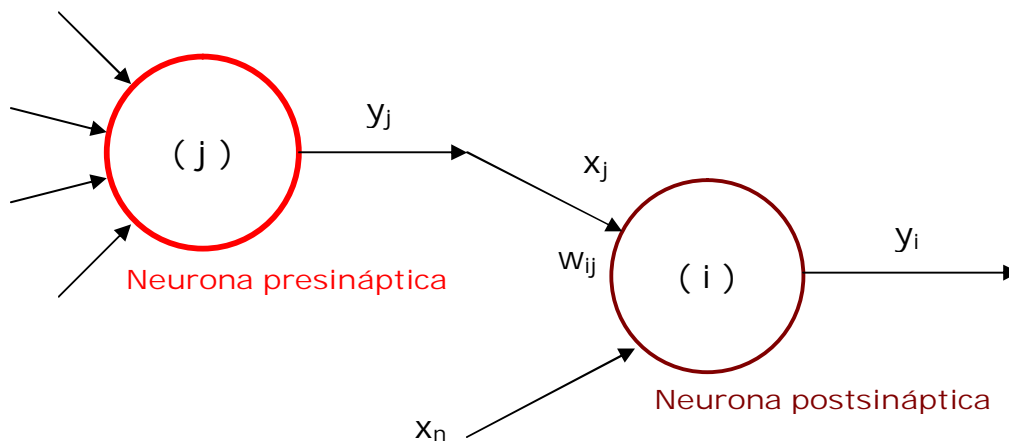
Rango :

[- 1 , + 1]

Nombre	Relación Entrada /Salida	Icono	Función
Limitador Fuerte	$a = 0 \quad n < 0$ $a = 1 \quad n \geq 0$		<i>hardlim</i>
Limitador Fuerte Simétrico	$a = -1 \quad n < 0$ $a = +1 \quad n \geq 0$		<i>hardlims</i>
Lineal Positiva	$a = 0 \quad n < 0$ $a = n \quad 0 \leq n$		<i>poslin</i>
Lineal	$a = n$		<i>purelin</i>
Lineal Saturado	$a = 0 \quad n < 0$ $a = n \quad 0 \leq n \leq 1$ $a = 1 \quad n > 1$		<i>satlin</i>
Lineal Saturado Simétrico	$a = -1 \quad n < -1$ $a = n \quad -1 \leq n \leq 1$ $a = +1 \quad n > 1$		<i>satlins</i>
Sigmoidal Logarítmico	$a = \frac{1}{1 + e^{-n}}$		<i>logsig</i>
Tangente Sigmoidal Hiperbólica	$a = \frac{e^n - e^{-n}}{e^n + e^{-n}}$		<i>tansig</i>
Competitiva	$a = 1$ Neurona con n max $a = 0$ El resto de neuronas		<i>compet</i>

Conexiones entre las Unidades

Desde un punto de vista matemático, se puede ver una red neuronal como un grafo dirigido y ponderado donde cada uno de los nodos son neuronas artificiales y los arcos que unen los nodos son las conexiones sinápticas. Al ser dirigido, los arcos son unidireccionales. ¿Que quiere decir esto? En el lenguaje de neuronas y conexiones significa que la información se propaga en un único sentido, desde una neurona presináptica (neurona origen) a una neurona postsináptica (neurona destino).



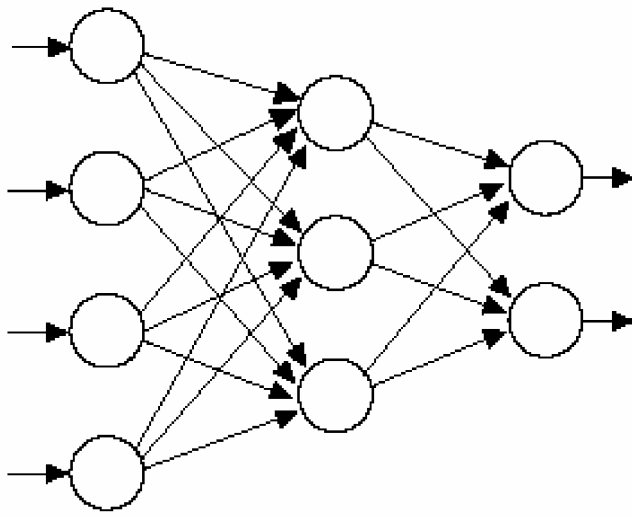
Por otra parte es ponderado, lo que significa que las conexiones tienen asociado un número real, un peso, que indica la importancia de esa conexión con respecto al resto de las conexiones y hace que la red adquiera conocimiento. Si dicho peso es positivo la conexión se dice que es excitadora, mientras que si es negativa se dice que es inhibidora. Si el peso es nulo se consideran conexiones inexistentes.

FORMAS DE CONEXIÓN ENTRE NEURONAS

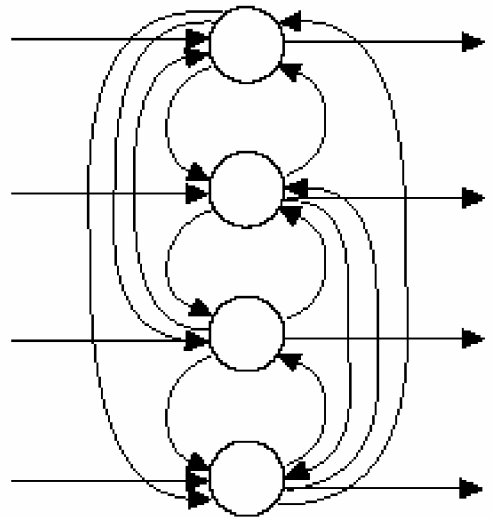
La conectividad entre los nodos de una red neuronal está relacionada con la forma en que las salidas de las neuronas están canalizadas para convertirse en entradas de otras neuronas. La señal de salida de un nodo puede ser una entrada de otro elemento de proceso, o incluso de sí mismo (conexión auto-recurrente).

Cuando ninguna salida de las neuronas es entrada de neuronas del mismo nivel o de niveles precedentes, la red se describe como **propagación hacia delante**. Cuando las salidas pueden estar conectadas como entradas de neuronas de niveles previos o del mismo nivel, incluyéndose ellas mismas, la red es de **propagación hacia atrás**. Las **redes** de propagación hacia atrás que tiene lazos cerrados son **sistemas recurrentes**.

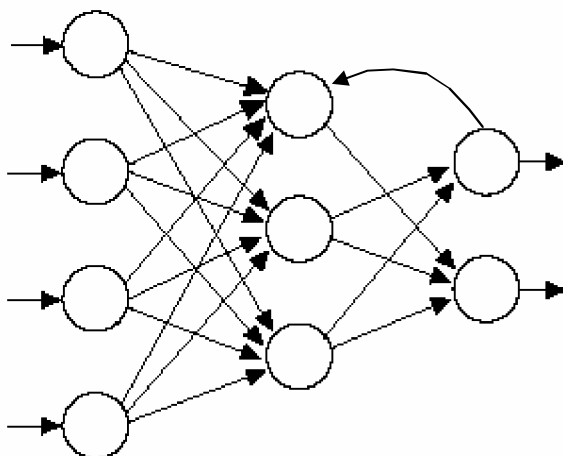
Conexiones hacia delante



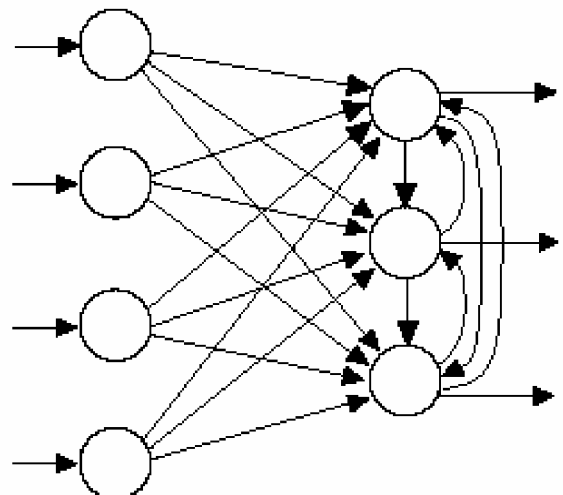
Conexiones laterales



Conexiones hacia atrás

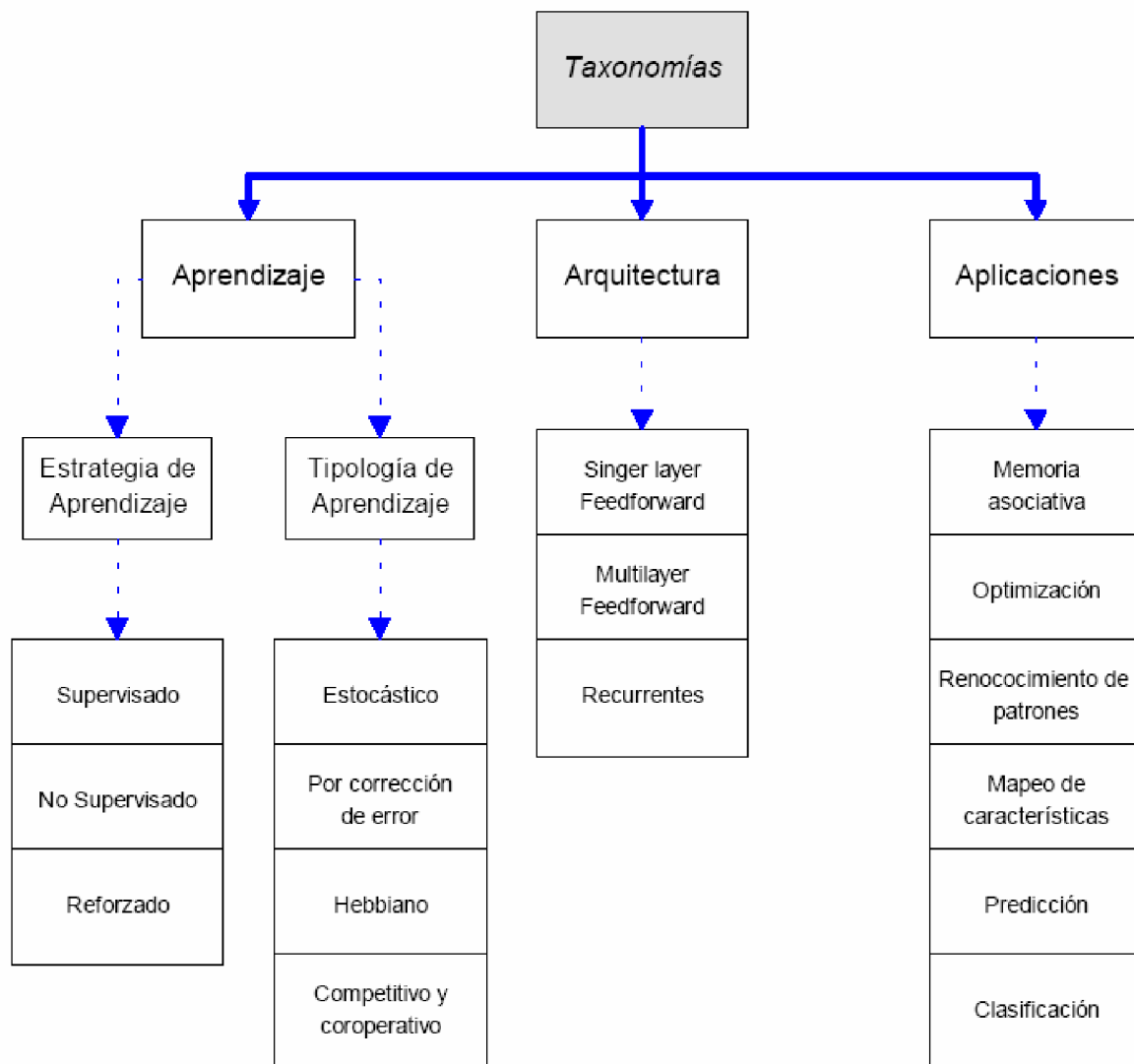


Conexiones Recurrentes



Modelos de Redes Neuronales. Taxonomía.

La gran variedad de modelos de redes neuronales existentes en la actualidad obliga en cierta medida a la realización de clasificaciones o taxonomías. De esta forma, los modelos neuronales se pueden clasificar desde una triple óptica: en función de la **arquitectura** ("network architecture"), en función de la forma del **aprendizaje** ("learning paradigm") y la tercera, que está situada en la área de las **aplicaciones**.



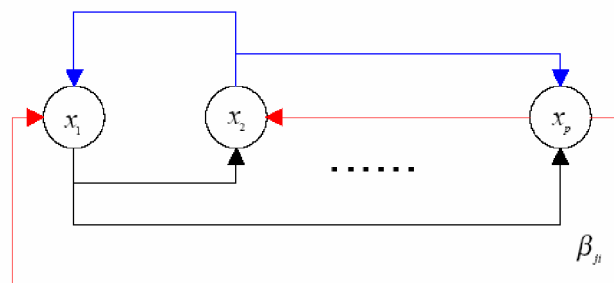
Arquitectura ó Topología

Los parámetros fundamentales de una red neuronal son: el número de capas, el número de neuronas por capa, el grado de conectividad y el tipo de conexiones ente neuronas.

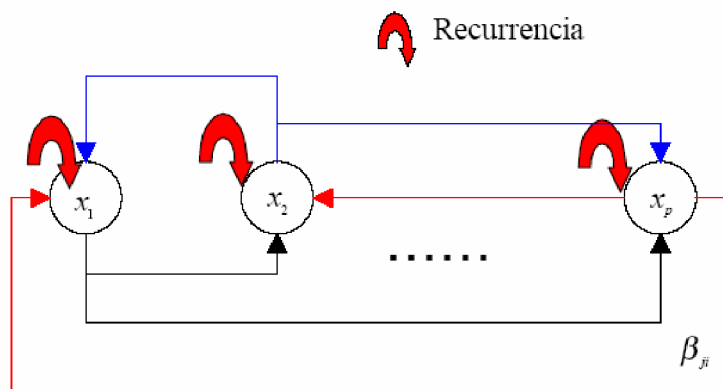
Clasificación según el número de capas.

Redes monocapa

En las redes monocapa, como la red HOPFIELD y la red BRAIN-STATE-IN-A-BOX, se establecen conexiones laterales entre las neuronas que pertenecen a la única capa de la red. Además pueden existir conexiones auto recurrente.



Modelo neuronal Hopfield



Modelo neuronal Brain-In-State-In-A-Box

Las redes monocapa se utilizan típicamente en tareas relacionadas con lo que se conoce como auto asociación, por ejemplo para regenerar informaciones de entrada que se presentan distorsionadas o incompletas.

Redes Multicapa

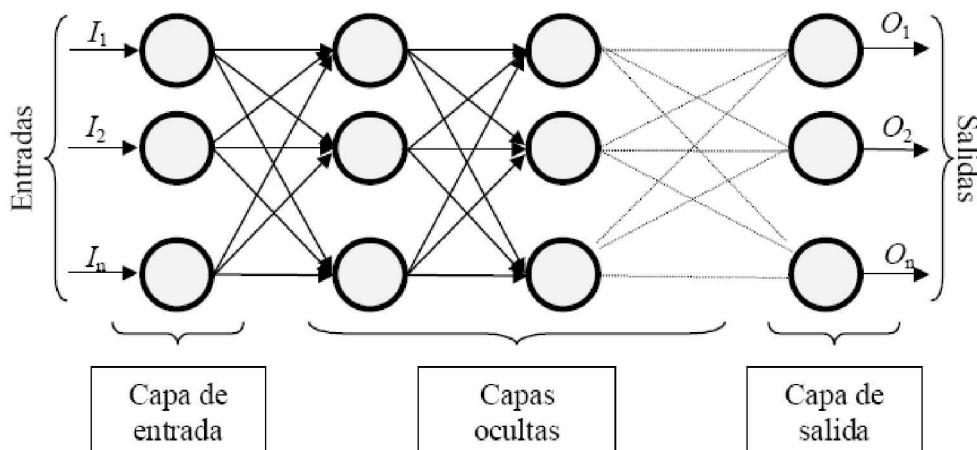
Son aquellas que disponen las neuronas agrupadas en varias capas. Una forma para distinguir la capa a la que pertenece una **neurona**, consistiría en fijarse en el origen de las señales que recibe a la entrada y el destino de la señal de salida.

Normalmente, todas las neuronas de una capa reciben señales de entrada de otra capa anterior, más cercana a la entrada de la red, y envían su señal de salida a una capa posterior, más cercana a la salida de la red. A estas conexiones se les denominan conexiones hacia delante o feedforward.

Sin embargo en un gran número de estas redes también existe la posibilidad de conectar las salidas de las neuronas de capas posteriores a las entradas de capas anteriores, a estas conexiones se les denomina conexiones hacia atrás o feedback.

Lo usual es que las neuronas se agrupen en capas de manera que una RNA esta formada por varias capas de neuronas. Aunque todas las capas son conjuntos de neuronas, según la función que desempeñan, suelen recibir un nombre específico. Las más comunes son las siguientes:

- Capa de entrada: las neuronas de la capa de entrada, reciben los datos que se proporcionan a la RNA para que los procese.
- Capas ocultas: estas capas introducen grados de libertad adicionales en la RNA. El número de ellas puede depender del tipo de red que estemos considerando. Este tipo de capas realiza gran parte del procesamiento.
- Capa de salida: Esta capa proporciona la respuesta de la red neuronal. Normalmente también realiza parte del procesamiento.

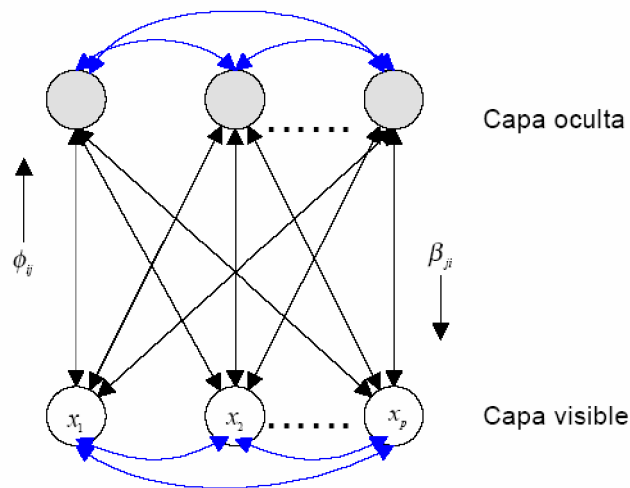


Clasificación según el tipo de conexiones:

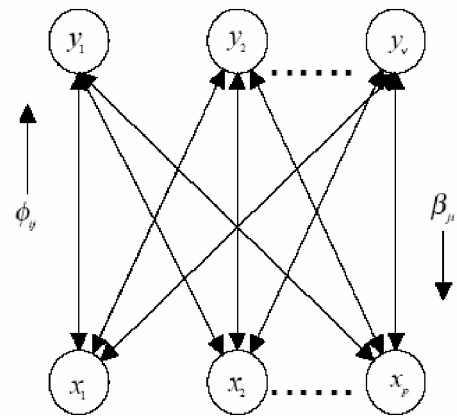
La salida de una neurona puede ser la entrada de un nodo de una capa siguiente, como también una entrada de sí misma o de neuronas de su mismo nivel o capa. Las redes **feed-forward** que consisten en un tipo de arquitectura donde no existen conexiones hacia atrás. Las redes **feedback**, que permiten conexiones laterales y hacia atrás. Y

las redes **recurrentes o realimentadas**, cuyas conexiones están permitidas tanto hacia atrás, como adelante, y la realimentación.

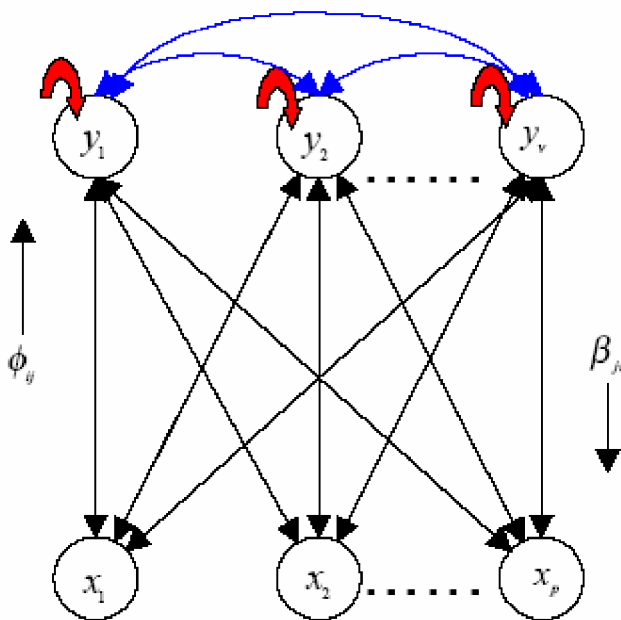
Ejemplos de redes recurrentes:



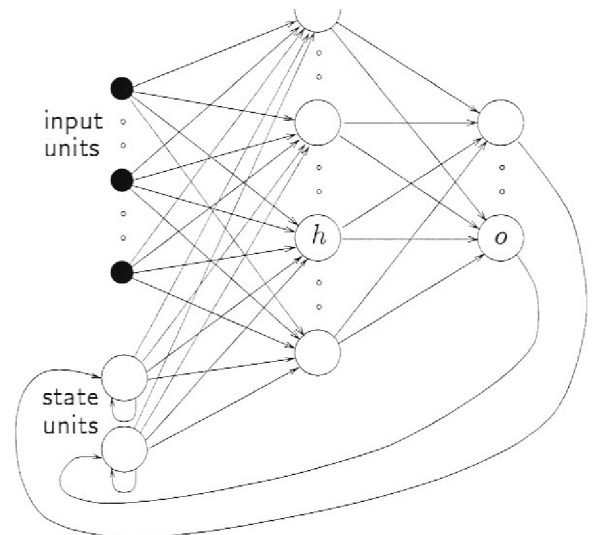
Modelo neuronal Boltzmann machine



Modelo neuronal Bam



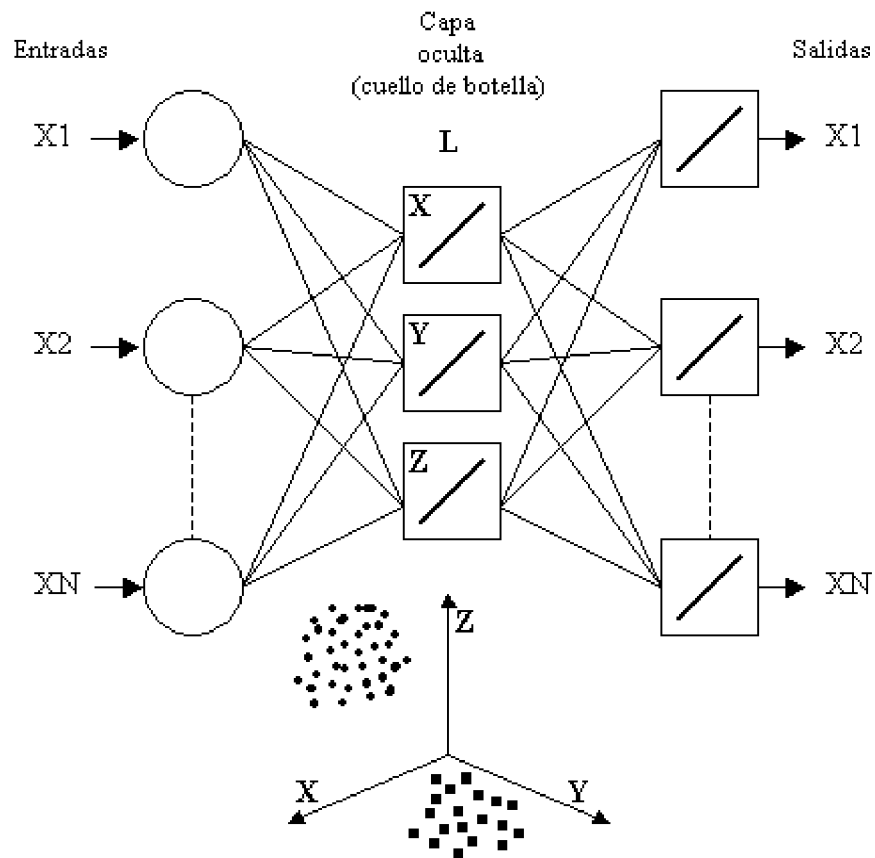
Modelo neuronal Arti



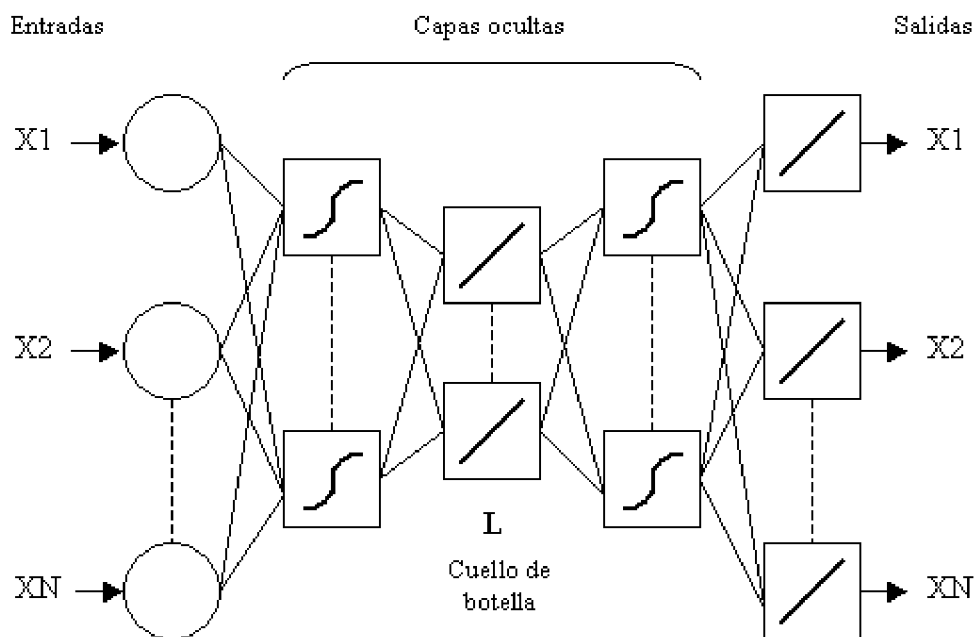
Modelo neuronal Jordan

En una red neuronal puede haber una función de activación común a todas las neuronas o por el contrario, diferentes funciones de activación en dicha red neuronal.

Función de activación lineal común a todas las capas



Función de activación lineal en la capa de salida y en la capa oculta 2.
Función de activación sigmoideal en las capas ocultas 1 y 3.



Fase de Procesamiento

1 ° - FASE DE APRENDIZAJE

Para que la red conozca situaciones, debe aprender, y para ello, se le debe introducir a la red una serie de entradas que se quiere que ella aprenda, de tal forma que entregue una salida deseada para cada una de las entradas proporcionadas. Es en este aprendizaje donde está la clave del funcionamiento de la red, ya que con un nivel adecuado de entrenamiento, la red podrá **generalizar**, es decir, responder correctamente, en tiempo real, a situaciones (entradas) que ella nunca ha visto. A su vez, si la red es sobreentrenada, se producirá un efecto de memorización en la red, es decir, sólo responderá correctamente ante entradas iguales o muy similares a las que fue entrenada.

En consecuencia, **aprendizaje** es el proceso en el cual la red modifica los pesos de sus conexiones en función de una entrada alimentada y una salida deseada, ampliando el concepto también para un conjunto de vectores de entrada, con sus respectivos vectores de salida deseados. En otras palabras, la red va modificando los valores de los pesos entre cada conexión, de tal forma que el vector obtenido en la capa de salida corresponda al vector de salida deseada, y esto para cada uno de los vectores de entrada con los cuales se forma el 'grupo de entrenamiento'.

La forma de efectuar el entrenamiento difiere enormemente, existiendo diferentes algoritmos de aprendizaje para cada esquema.

Hay cuatro clases de aprendizaje distintos:

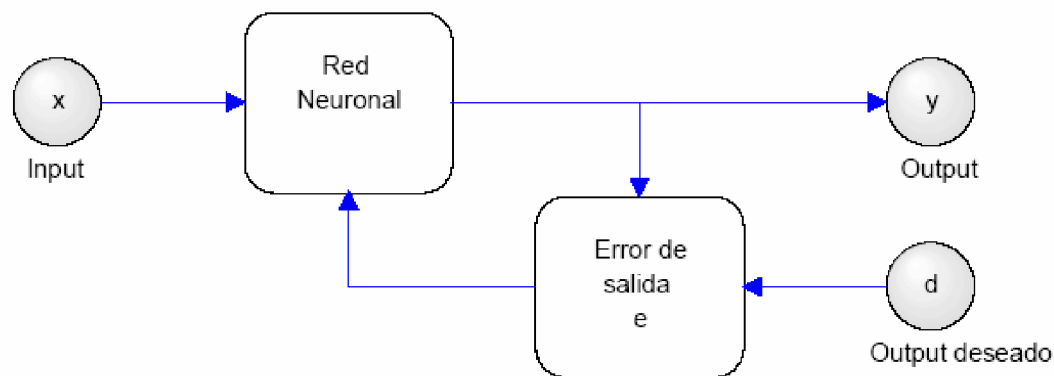
- § **Aprendizaje supervisado:** En este tipo de aprendizaje se le proporciona a la red neuronal una serie de ejemplos consistentes en unos patrones de entrada, junto con la salida que debería dar la red. El proceso de entrenamiento consiste en el ajuste de los pesos para que la salida de la red sea lo más parecida posible a la salida deseada. Es por ello que en cada iteración se use alguna función que nos de cuenta del error o el grado de acierto que esta cometiendo la red.
- § **Aprendizaje no supervisado o autoorganizado:** En este tipo de aprendizaje se presenta a la red una serie de ejemplos pero no se presenta la respuesta deseada. Lo que hace la red neuronal es reconocer regularidades en el conjunto de entradas, es decir, estimar una función densidad de probabilidad $p(x)$ que describe la distribución de patrones x en el espacio de entrada R^n .
- § **Aprendizaje Híbrido:** Es una mezcla de los anteriores. Unas capas de la red tienen un aprendizaje supervisado y otras capas de la red tienen un aprendizaje de tipo no supervisado. Este tipo de entrenamiento es el que tienen redes como las RBF.
- § **Aprendizaje reforzado (reinforcement learning):** Es un aprendizaje con características del supervisado y con características del autoorganizado. No se proporciona una salida deseada, pero sí que se le indica a la red en cierta medida el error que comete, aunque es un error global.

Asociada a las anteriores tipologías de aprendizaje están las diferentes estrategias para poder conseguirlas.

Aprendizaje Supervisado:

Consiste en construir un modelo neuronal que permita estimar relaciones entre los inputs y los outputs sin la necesidad de proponer una cierta forma funcional a priori, siendo desde la óptica computacional más complejo pero con resultados más exactos. Existen dos algoritmos de aprendizaje supervisado: el aprendizaje por correlación de error y el aprendizaje estocástico.

APRENDIZAJE SUPERVISADO



Aprendizaje por correlación de error

Este consiste en ajustar los pesos de las conexiones de la red en función de la diferencia entre los valores deseados y los obtenidos en la salida de la red, es decir, en función del error cometido. Ejemplos de este tipo de aprendizaje son los tres algoritmos siguientes: la regla de aprendizaje del Perceptron, utilizada en el aprendizaje de la red Perceptron diseñada por Rosenblatt en 1958; la regla delta o regla del error cuadrático medio mínimo ("Least-Mean-squared Error" (LMS)) propuesta por Widrow en 1960, utilizada en los modelos neuronales Adaline y Madaline (estos modelos mejoran el modelo de Perceptron ya que incorporan la definición de error global cometido y mecanismos para reducirlo con mayor rapidez); y la regla delta generalizada o retropropagación del error.

Estocástico

Consiste en realizar cambios aleatorios en los valores de los pesos de las conexiones de la red y evaluar el efecto a partir del objetivo deseado mediante distribuciones de probabilidad. A partir de conceptos de termodinámica, se asocia a la red un nivel de estado energético, de forma que, si su nivel es de mínimos entonces corresponde a una red bien entrenada. En algunos casos es necesario introducir "ruido" en el proceso de aprendizaje que facilita escapar de los mínimos locales o relativos. Nos referimos al procedimiento temple simulado (simulated annealing) que junto a la asignación probabilística forman parte del aprendizaje estocástico.

Aprendizaje No Supervisado:

Las redes con dicho aprendizaje no requieren de influencia externa para ajustar los pesos de las conexiones entre sus neuronas. La red no recibe ninguna información por parte del entorno que le indique si la salida generada en respuesta de una entrada es o no correcta. Suele decirse que estas redes son capaces de autoorganizarse.

Estas redes deben encontrar las **características**, regularidades, correlaciones o categorías que se pueden establecer entre los datos que se presentan en su entrada.

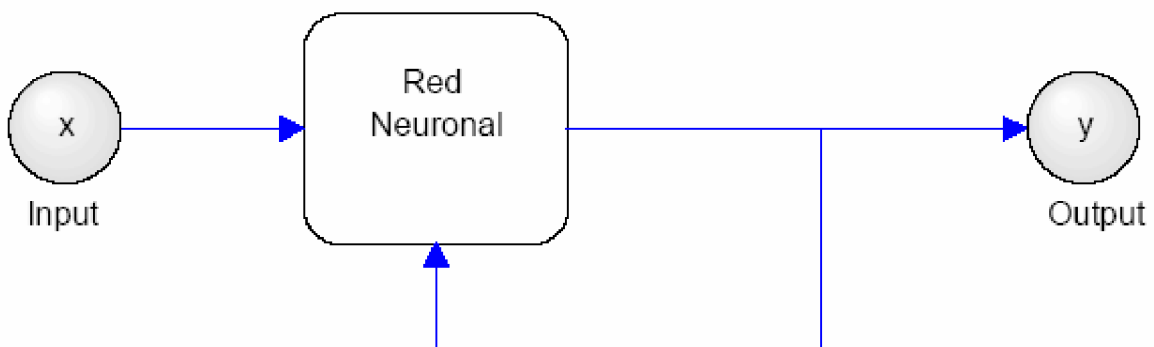
En algunos casos, la salida representa el grado de familiaridad o similitud entre la información que se le está presentando en la entrada y las que se le han mostrado en el pasado. En otro caso podría realizar una clusterización, indicando la red la salida a qué categoría pertenece la información presentada a la entrada, siendo la propia red quien debe encontrar las categorías apropiadas a partir de correlaciones entre las informaciones presentadas. Una variación de esta categorización es el prototipado. En este caso, la red obtiene prototipos representantes de las clases a las que pertenecen las informaciones de entrada.

También el aprendizaje sin **supervisión** permite realizar una codificación de los datos de entrada, generando a la salida una versión codificada de la entrada, con menos bits, pero manteniendo la información relevante de los datos.

Algunas redes con aprendizaje no supervisado, realizan un mapeo de características, obteniéndose en las neuronas de salida una disposición geométrica que representa un mapa topográfico de las características de los datos de entrada, de tal forma que si se presentan a la red informaciones similares, siempre sean afectadas neuronas de salidas próximas entre sí, en la misma zona del mapa

Suelen considerarse dos **algoritmos** de aprendizaje no supervisado: aprendizaje hebbiano y aprendizaje competitivo y cooperativo.

APRENDIZAJE NO SUPERVISADO



Aprendizaje hebbiano

Este tipo de aprendizaje se basa en el postulado formulado por Donald O. Hebb en 1949: "Cuando un axón de una celda A está suficientemente cerca como para conseguir excitar a una celda B y repetida o persistentemente toma parte en su activación, algún proceso de crecimiento o cambio metabólico tiene lugares en una o ambas celdas, de tal forma que la **eficiencia** de A, cuando la celda a activar es B, aumenta. Por celdas, Hebb entiende un conjunto de neuronas fuertemente conexas a través de una estructura compleja. La eficiencia podría identificarse por la intensidad o magnitud de la conexión, es decir, con el peso.

Se puede decir que el aprendizaje consiste en el ajuste de los pesos de las conexiones de acuerdo con la correlación (multiplicación en el caso de valores binarios +1 y -1) de los valores de activación (salidas) de las neuronas conectadas.

$$\Delta w_{ij} = y_i * y_j$$

Esta expresión responde a la idea de Hebb, puesto que si las dos unidades son activas (positivas), se refuerza la conexión; por el contrario, cuando una es activa y la otra pasiva, se debilita la conexión.

Existen muchas variaciones de dicho aprendizaje, por ejemplo, Sejnowski en 1977 utilizó la correlación de covarianza de los valores de activación de las neuronas. Sutton y Barto en 1981 utilizaron la correlación el valor medio de una neurona con la varianza de la otra. Klopff en 1986 propuso una correlación entre las variaciones de los valores de activación en dos instantes de **tiempo** sucesivos, aprendizaje que denominó drive-reinforcement y que utilizó en redes del mismo nombre con topología feedforward de dos capas.

Otra versión e este aprendizaje es el **hebbiano diferencial**, que utiliza la correlación de las **derivadas** en el tiempo de las **funciones** de activación de las neuronas.

Aprendizaje competitivo y cooperativo

En dicho aprendizaje suele decirse que las neuronas compiten (y cooperan) unas con otras con el fin de llevar a cabo una tarea dada.

La competición ente neuronas se realiza en todas las capas de la red, existiendo en estas neuronas conexiones recurrentes de autoexcitación y conexiones de inhibición por parte de neuronas vecinas. Si el aprendizaje es cooperativo, estas conexiones con las vecinas serán de excitación.

El objetivo de este aprendizaje es clusterizar los datos que se introducen en la red. De esta forma, las informaciones similares son clasificadas formando parte de la misma categoría, y por tanto deben activar la misma neurona de salida. Las categorías deben ser creadas por la misma red, puesto que se trata de aprendizaje no supervisado, a través de las correlaciones ente los datos.

En este tipo de redes, cada neurona tiene asignado un peso total, suma de todos los pesos de las conexiones que tiene a su entrada. El aprendizaje afecta sólo a las neuronas ganadoras (activas), redistribuyendo este peso total entre todas las conexiones que llegan a al neurona vencedora y repartiendo esta cantidad por igual entre todas las conexiones procedentes de unidades activas. Por tanto, la variación del peso de una conexión ente una unidad i y otra j será nula si la neurona j no

recibe excitación por parte de la neurona i y otra j será nula si la neurona j no recibe excitación por parte de la neurona i , y se modificará si es excitada por dicha neurona i .

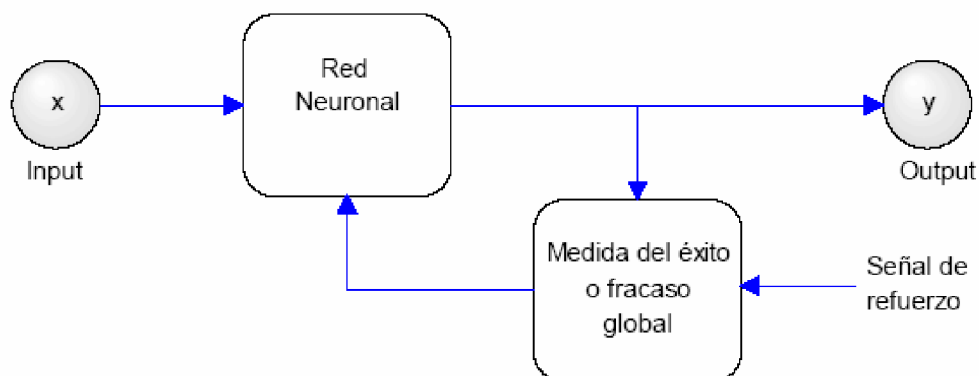
Existe otro caso particular de aprendizaje competitivo, denominado [teoría de la resonancia adaptativa](#), desarrollado por Carpenter y Grossberg en 1986 y utilizado en la red feedforward /feedback de dos capas conocida como ART. Esta red realiza un prototipado de las informaciones que recibe a la entrada, generando como salida un ejemplar o prototipo que representa a todas las informaciones que podrían considerarse pertenecientes a la misma categoría.

Aprendizaje por refuerzo

Se basa en la idea de no disponer de un ejemplo completo del comportamiento deseado; es decir, de no indicar durante el entrenamiento exactamente la salida que se desea que proporcione la red ante una determinada entrada.

En el aprendizaje por refuerzo la función del supervisor se reduce a indicar mediante una señal de refuerzo si la salida obtenida en la red se ajusta a la deseada (**éxito** = +1 o **fracaso** = -1), y en función de ello se ajustan los pesos basándose en un mecanismo de probabilidades.

APRENDIZAJE REFORZADO



Un ejemplo de algoritmo es el Linear Reward-Penalty o L_{R-P} (algoritmo lineal con recompensa y penalización) presentado por Narendra y Thathacher en 1974. Este algoritmo ha sido ampliado por Barto y Anandan, quienes en 1985 desarrollaron el denominado Associative Reward-Penalty o A_{R-P} (algoritmo asociativo con recompensa y penalización), que se aplica en redes con conexiones hacia adelante de dos capas cuyas neuronas de salida presentan una función de activación estocástica.

Otro algoritmo conocido es el Adaptive Heuristic Critic, introducido por Barto, Sutton y Anderson en 1983, que se utiliza en redes con conexiones hacia adelante de tres capas especialmente diseñadas para que una parte de la red sea capaz de generar un valor interno de refuerzo que es aplicado a las neuronas de salida de la red.

Tipologías de aprendizaje	Estrategias de aprendizaje	Ejemplos
Supervisado	Correlación de error	Perceptrón Regla Delta (Madaline) Backpropagation Counterpropagation
	Estocástico	Máquina de Boltzmann Máquina de Cauchy
No Supervisado	Hebbiano	Hopfield Linear Associative Memory Fuzzy Associative Memory Grossberg Bidirectional Associative Memory
	Competitivo y Cooperativo	Learning vector quantizer Cognitrón/Neocognitrón Teoría de Resonancia Adaptiva
Híbrido		Radial Basis Function Networks Cerebellar Model Articulation Controller Counter-Propagation Networks Logicon Projection Network
Por Refuerzo		Linear Reward Penalty Associative Reward Penalty Adaptive Heuristic Critic

2 ° - FASE DE PRESENTACIÓN

Construcción de Modelos

Paso 1: Recogida de datos

Asegurarse existen datos suficientes disponibles para un estudio significativo tanto en términos de calidad como de cantidad.

Paso 2: Seleccionar las variables a predecir

Clara condición de la aplicación del modelo propuesto, reconociendo que puede ser más apropiado para modelizar las variaciones de flujo si los datos contienen una gran varianza (y los datos de calibración podrían ser poco representativos de los extremos de larga duración). Verificar que los datos son adecuados para el modelo en cuestión.

Paso 3: Preprocesamiento de datos (etapa 1)

3.1. Limpieza de datos: quitar las tendencias subyacentes ascendentes o descendentes significativas. Si es necesario, eliminar las componentes estacionales y /o filtrar los datos para reducir el ruido y enfatizar la señal dominante.

Paso 4: Selección de la red neuronal

4.1. Tipo de red: seleccionar el tipo de red más apropiado para la aplicación en cuestión. Mientras no hay un tipo definitivo para elegir, en términos de problemas de predicción, tanto la red neuronal MLP como la RBF son apropiadas como punto de partida. Comenzar con la red MLP (entrenada usando el algoritmo de retropropagación estándar) ya que proporciona un estándar de referencia con el cual evaluar cualquier otro modelo.

Elegir una(s) función(s) de activación adecuada(s) para las neuronas. Para las MLP generalmente utilizar o la función sigmoide o la tangente hiperbólica como punto de partida. Para las RBF, las funciones que más se utilizan son las de base Gaussiana.

4.2. Algoritmo de entrenamiento: seleccionar un algoritmo de entrenamiento adecuado para modificar los pesos y los sesgos, y determinar la estructura de la red. Elegir valores apropiados para los parámetros de aprendizaje (índice de aprendizaje y de momentum) dentro del rango 0.01 – 0.9.

Paso 5: Preprocesamiento de datos (etapa 2)

5.1. Estandarización de los datos: de acuerdo al algoritmo elegido, estandarizar los datos a los intervalos [0,1], [-1,1], [0.1,0.9], etc. o normalizarlos.

5.2. Conjuntos de datos: crear conjuntos de datos para la validación cruzada mediante la división de los datos en grupos apropiados de calibración, ensayo y validación. Con un conjunto grande de datos esto es relativamente sencillo ya que los datos pueden ser divididos en tres grupos representativos. Sin embargo, para conjuntos de datos pequeños, debería de utilizarse la validación cruzada.

Paso 6: Entrenamiento de la red

6.1. Estructura: especificar el número de capas ocultas y el número de nodos en estas capas. Esto puede ser innecesario si se utiliza el algoritmo de eliminación o de correlación en cascada. Comenzar con una capa oculta.

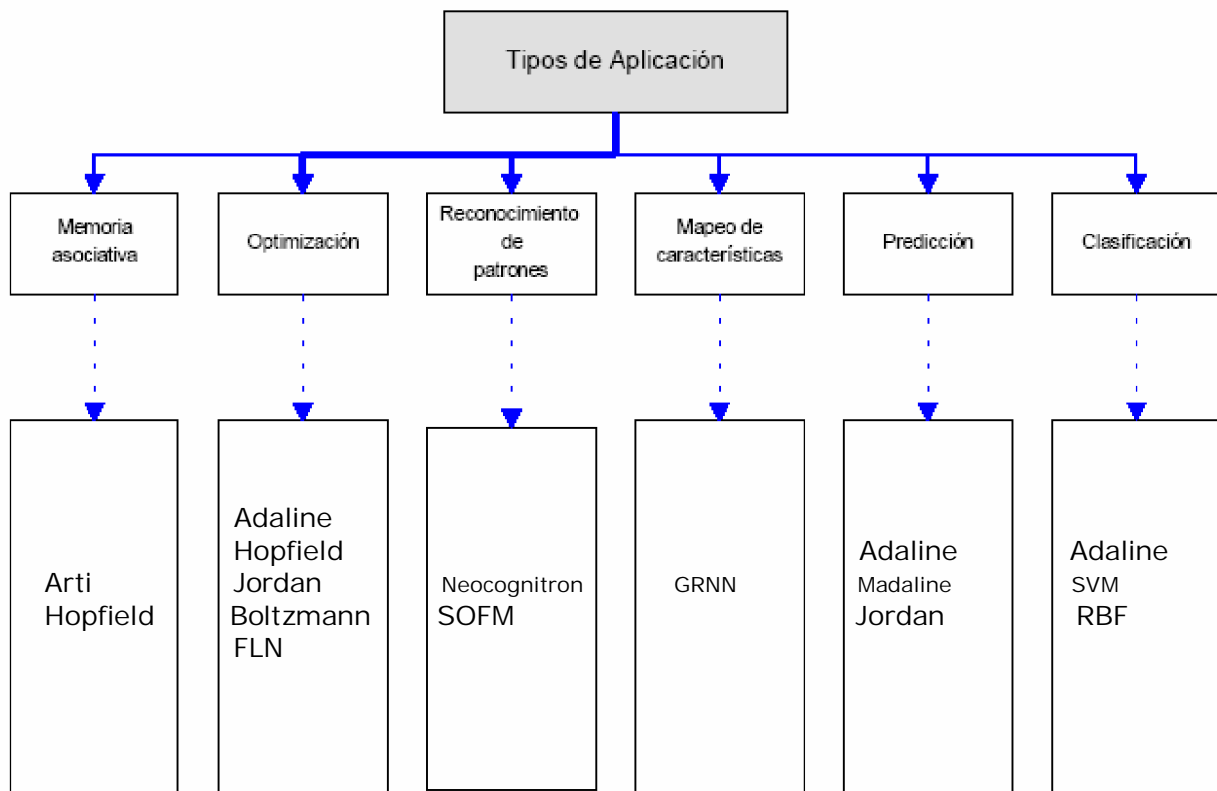
6.2. Entrenamiento: entrenar un número de redes usando los datos de calibración y de ensayo. Terminar el proceso de entrenamiento cuando los resultados de los datos de ensayo indican sobreadecuación (overfitting) de los de calibración.

Paso 7: Evaluación

Seleccionar las medidas de error que son apropiadas para el propósito y salida del modelo. Comparar los resultados con aquellos derivados de las configuraciones del modelo alternativo.

Ejemplos y Aplicaciones

Respecto a las diferentes aplicaciones tenemos, en primer lugar, **memoria asociativa**, que consiste en reconstruir una determinada información de entrada que se presenta incompleta o distorsionada, asociando la información de entrada con el ejemplar más parecido de los almacenados conocidos por la red. En segundo lugar, la **optimización**, es decir, la resolución de problemas de optimización combinatoria que si se utilizan sistemas convencionales requieren mucho tiempo de computación. En tercer lugar, el **reconocimiento de patrones**, consistente, desde una óptica general, en la detección de formas simples (podríamos hablar de la percepción artificial). En cuarto lugar, el **mapeo de características**, que parte de las ideas de Kohonen (1982) simulando la capacidad del cerebro humano de crear mapas topológicos de las informaciones recibidas del exterior. En quinto lugar, está la **predicción** y en último lugar, la **clasificación**.



Podemos observar finalmente que una misma red puede utilizarse en aplicaciones diferentes.

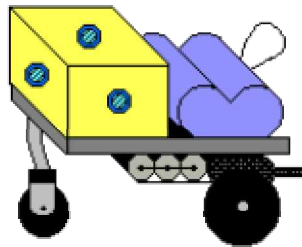
Campos concretos donde se aplican las redes neuronales:

- § Automóviles.
Sistemas de Piloto automático. Detección de fallas por reconocimiento externo de vibraciones.
- § Bancos.
Lectura de cheques y otros documentos. Evaluación de aplicaciones de créditos.
- § Electrónica.
Predicción de secuencia de códigos. Distribución de elementos en Circuitos Integrados. Control de Procesos. Análisis de fallas. Visión Artificial. Reconocimiento de Voz. Modelamiento no-lineal.
- § Finanzas.
Tasación real de los bienes. Asesoría de préstamos. Previsión en la evolución de precios. Seguimiento de hipotecas. Análisis de uso de Línea de Crédito. Evaluación del riesgo en créditos. Identificación de falsificaciones. Interpretación y reconocimiento de firmas.
- § Manufactura.
Control de la producción y del proceso. Análisis y diseño de productos. Diagnóstico de fallas en el proceso y maquinarias. Identificación de partículas en tiempo real. Inspección de calidad mediante sistemas visuales. Predicción de calidad del papel. Análisis de mantenimiento de máquinas. Modelamiento dinámico de procesos químicos.
- § Medicina.
Análisis de células portadoras de cáncer mamario. Análisis de Electroencefalograma y de Electrocardiograma. Reconocimiento de infartos mediante ECG. Diseño de prótesis. Optimización en tiempos de trasplante. Reducción de gastos hospitalarios.
- § Robótica.
Control dinámico de trayectoria. Robots elevadores. Controladores. Sistemas ópticos.
- § Seguridad.
Códigos de seguridad adaptivos. Criptografía. Reconocimiento de huellas digitales.
- § Telecomunicaciones.
Compresión de datos e imágenes. Automatización de servicios de información. Traslación en tiempo real de lenguaje hablado.
- § Transporte.
Diagnóstico de frenos en camiones. Sistemas de ruteo y seguimiento de flotas.
- § Voz.
Reconocimiento de voz. Compresión de voz. Clasificación de vocales. Transformación de texto escrito a voz.

DETECCIÓN DE OBSTÁCULOS POR MEDIO DE UN ROBOT

Descripción del problema:

Un robot es un dispositivo automático que realiza acciones específicas, que dependen de las necesidades del proceso en que se encuentre involucrado, en este caso se tiene un robot que cuenta con cuatro sensores de proximidad en distintas ubicaciones que permanentemente detectan si hay objetos que se encuentren a una distancia superior o inferior a la preestablecida, con base en esto se decide si dar marcha adelante o atrás a cada uno de los dos motores que posee; en las lecturas de los sensores podrían darse 16 posibles combinaciones ($16=2^4$) y para cada combinación cada uno de los dos motores podría dar marcha adelante o marcha atrás.



Robot

Cuando los sensores detecten un objeto que se encuentra a una distancia inferior a la predeterminada se dirá que el objeto se encuentra cerca y esto se representa por medio de un 1 y cuando se detecte un objeto que se encuentra a una distancia mayor que la predeterminada se dirá que el objeto esta lejos lo cual se indica con un -1; dependiendo de estas lecturas los motores podrán dar marcha adelante, lo que se representara por un 1 o dar marcha atrás con un -1.

S1	S2	S3	S4	M1	M2
1	1	1	1	-1	-1
-1	1	1	1	-1	1
1	1	-1	-1	1	-1
-1	-1	-1	-1	1	1
1	-1	1	1	1	-1
1	1	-1	1	-1	1
1	1	1	-1	1	-1

Comportamiento del robot

Justificación del tipo de red

Este tipo de problema generalmente es resuelto suministrándole al robot una base de datos que contiene todas las posibles situaciones que se podrían presentarse y sus respectivas soluciones, en este caso se necesitaría almacenar las respuestas para ambos motores ante las 16 posibles combinaciones en las lecturas de los sensores, cuando el número de variables de entrada y el número de salidas es mucho mayor, la cantidad de datos necesarios para especificar cada posible situación crece indefinidamente, debido a esto se requerirían dispositivos con gran capacidad de almacenamiento; en contraste una red neuronal puede entrenarse con un número representativo de patrones y aprender el comportamiento del sistema utilizando dispositivos de menos capacidad de almacenamiento y costo.

Una red tipo Perceptrón puede ser entrenada con patrones de cualquier dimensión en la entrada y en la salida con datos binarios, por la simplicidad del problema este tipo de red es la mas adecuada.

Para garantizar que el problema puede ser resuelto por una red neuronal tipo Perceptrón se debe comprobar que los patrones de entrenamiento son linealmente separables, para esto se deben plantear las desigualdades generadas por cada patrón de entrenamiento, en este caso cada patrón de cuatro dimensiones generara dos desigualdades (una por cada salida), estas desigualdades no deben contradecirse, si esto ocurriera el problema no podría ser resuelto por una red tipo Perceptrón de una sola capa y deberá buscarse otro tipo de solución.

Debido a la naturaleza bipolar de la salida, la función de transferencia a utilizar es *hardlims* la cual se rige por las siguientes condiciones:

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

La salida de la red está dada por la ecuación :

$$n = (Wp + b)$$

Aplicando esta ecuación a cada patrón de entrenamiento se tienen las desigualdades, las cuales se satisfacen plenamente, lo que implica que el problema es linealmente separable y puede ser resuelto por una red tipo Perceptron

$$\begin{aligned} p_1 \begin{cases} W_{11} + W_{12} + W_{13} + W_{14} + b_1 < 0 \\ W_{21} + W_{22} + W_{23} + W_{24} + b_2 < 0 \end{cases} & p_2 \begin{cases} -W_{11} + W_{12} + W_{13} + W_{14} + b_1 < 0 \\ -W_{21} + W_{22} + W_{23} + W_{24} + b_2 \geq 0 \end{cases} \\ p_3 \begin{cases} W_{11} + W_{12} - W_{13} - W_{14} + b_1 \geq 0 \\ W_{21} + W_{22} - W_{23} - W_{24} + b_2 < 0 \end{cases} & p_4 \begin{cases} -W_{11} - W_{12} - W_{13} - W_{14} + b_1 \geq 0 \\ -W_{21} - W_{22} - W_{23} - W_{24} + b_2 \geq 0 \end{cases} \end{aligned}$$

$$p_5 \begin{cases} W_{11} - W_{12} + W_{13} + W_{14} + b_1 \geq 0 \\ W_{21} - W_{22} + W_{23} + W_{24} + b_2 < 0 \end{cases} \quad p_6 \begin{cases} W_{11} + W_{12} - W_{13} + W_{14} + b_1 < 0 \\ W_{21} + W_{22} - W_{23} + W_{24} + b_2 \geq 0 \end{cases}$$

$$p_7 \begin{cases} W_{11} + W_{12} + W_{13} - W_{14} + b_1 \geq 0 \\ W_{21} + W_{22} + W_{23} - W_{24} + b_2 < 0 \end{cases}$$

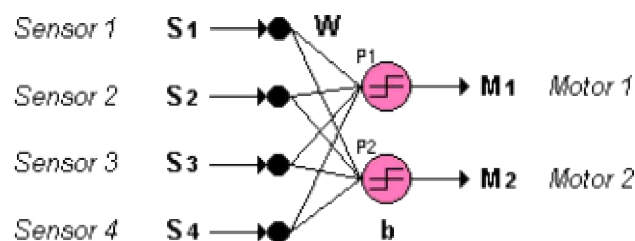
Desigualdades que garantizan que el problema sea linealmente separable

Entrenamiento de la red:

A la red se le presentaran 7 patrones de la tabla, para los cuales dependiendo de las lecturas de los sensores se le dirá al robot que hacer específicamente y luego se probará la red con los casos restantes para comprobar la capacidad de generalización de la red neuronal ante un patrón nunca antes visto.

Los estados de lecturas de los sensores y de operación de los motores fueron designados con 1 y -1, puesto que para la convergencia del proceso de entrenamiento resulta más ventajosos propagar valores de 1 y -1 que de 1 y 0.

Debido a la naturaleza de la salida y de la entrada de la red, la función de transferencia apropiada es *hardlims*, la cual trabaja con valores bipolares.



Red tipo Perceptrón

Se creará una red de 4 entradas con una neurona tipo Perceptrón para cada salida, teniendo así una salida bidimensional, los pesos iniciales aleatorios de la red se muestran en la siguiente tabla

Wi=net.IW{1,1}						bi=net.b{1}	
	S1	S2	S3	S4			
M1	0.8636	-0.1627	0.0503	0.3443		M1	1
M2	-0.0680	0.6924	-0.5947	0.6762		M2	1

Pesos Iniciales

El siguiente código crea una red tipo Perceptrón con función de transferencia *hardlims*, dos neuronas en la salida, utiliza como patrones de entrenamiento las lecturas de los cuatro sensores almacenados en *p* y como patrones objetivo o salidas deseadas las acciones de ambos motores almacenados en el vector *t*.

Los pesos finales de la red entrada que satisface todos los patrones de entrada y salida son:

Wf=net.IW{1,1}						bf=net.b{1}	
	S1	S2	S3	S4			
M1	0.8636	-4.1627	0.0503	-3.6557		M1	1
M2	-4.0680	0.6924	-4.5947	4.6762		M2	1

Pesos finales red entrenada

La red fue simulada para la totalidad de combinaciones posibles de entrada para comprobar que no exista error en el aprendizaje de los patrones de entrenamiento y para observar su capacidad de generalización en los casos restantes

La respuesta de la red a todos los patrones de entrenamiento fue exitosa, como puede observarse en la siguiente tabla:

	S1	S2	S3	S4	M1	M2
P1	1	1	1	1	-1	-1
P2	-1	1	1	1	-1	1
P3	1	1	-1	-1	1	-1
P4	-1	-1	-1	-1	1	1
P5	1	-1	1	1	1	-1
P6	1	1	-1	1	-1	1
P7	1	1	1	-1	1	-1

Simulación de la red para los Patrones de Entrenamiento

La red fue simulada para las posibles combinaciones restantes obteniéndose los siguientes resultados:

	S1	S2	S3	S4	M1	M2
C1	-1	-1	1	1	1	1
C2	-1	1	-1	1	-1	1
C3	1	-1	-1	1	1	1
C4	-1	-1	-1	1	1	1
C5	-1	1	1	-1	-1	-1
C6	1	-1	1	-1	1	-1
C7	-1	-1	1	-1	1	-1
C8	-1	1	-1	-1	-1	1
C9	1	-1	-1	-1	1	-1

Simulación de la red para las nuevas combinaciones

Las combinaciones que no hacían parte del set de entrenamiento, al ser presentadas a la red fueron aproximadas al patrón del set de entrenamiento aprendido con menor distancia euclidiana.

Para las combinaciones C1, C2 y C4 la red decidió dar marcha adelante a ambos motores; para la combinación C5 la red decidió dar marcha atrás a ambos motores, para las combinaciones C6, C7 y C9 la red decidió dar marcha adelante a M1 y marcha atrás a M2 y para las combinaciones C2 y C8 la red decidió dar marcha atrás a M1 y marcha adelante a M2

Una red tipo Perceptrón de una sola capa es una buena solución a un problema que involucre patrones linealmente separables, en el caso de contar con patrones que no son linealmente separables se tiene la alternativa de utilizar una red Perceptrón multicapa o cambiar definitivamente de red, nótese que una red Perceptrón multicapa puede solucionar el problema de separabilidad lineal a medida que aumenta el numero de capas de la red.

La capacidad de generalización de las redes neuronales juega un papel importante cuando las posibles combinaciones de patrones de entrada son tantas que resultaría imposible especificarle a un dispositivo que hacer en cada caso, puesto que la red se entrena con un número de patrones representativo y no con la totalidad de ellos, ahorrando tiempo de computo en la solución del problema.

En las aplicaciones desarrolladas con redes neuronales juega un papel importante la tolerancia a fallas que las caracteriza, pues en caso de fallar uno o varios sensores (como en este caso) la red siempre producirá una salida que en la mayoría de los casos es la mas acertada, debido a que la red después de un proceso de aprendizaje exitoso esta en capacidad de generalizar el comportamiento del sistema.

Historia de las Redes Neuronales

Conseguir diseñar y construir máquinas capaces de realizar procesos con cierta inteligencia ha sido uno de los principales **objetivos** de los científicos a lo largo de la **historia**. De los intentos realizados en este sentido se han llegado a definir las líneas fundamentales para la obtención de máquinas inteligentes: En un principio los esfuerzos estuvieron dirigidos a la obtención de **autómatas**, en el sentido de máquinas que realizaran, con más o menos **éxito**, alguna **función** típica de los seres humanos. Hoy en día se continúa estudiando en ésta misma línea, con resultados sorprendentes, existen maneras de realizar procesos similares a los inteligentes y que podemos encuadrar dentro de la llamada **Inteligencia Artificial (IA)**.

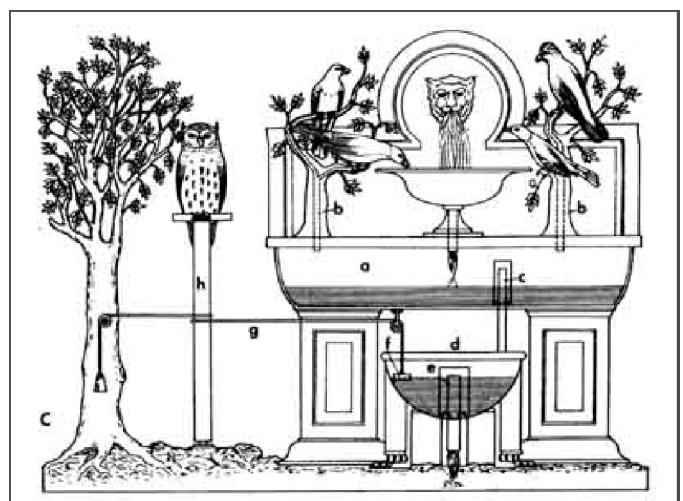
A pesar de disponer de **herramientas** y **lenguajes de programación** diseñados expresamente para el **desarrollo** de máquinas inteligentes, existe un enorme problema que limita los resultados que se pueden obtener: estas máquinas se implementan sobre **computadoras** basadas en la filosofía de Von Neumann, y que se apoyan en una **descripción** secuencial del **proceso** de tratamiento de la información. Si bien el desarrollo de estas computadoras es espectacular, no deja de seguir la línea antes expuesta: una máquina que es capaz de realizar tareas mecánicas de forma increíblemente rápida, como por ejemplo **cálculo**, ordenación o **control**, pero incapaz de obtener resultados aceptables cuando se trata de tareas como reconocimiento de formas, voz, etc.

La otra línea de la **investigación** ha tratado de aplicar **principios** físicos que rigen en la **naturaleza** para obtener máquinas que realicen trabajos pesados en nuestro lugar. De igual manera se puede pensar respecto a la forma y capacidad de razonamiento humano; se puede intentar obtener máquinas con esta capacidad basadas en el mismo principio de funcionamiento.

No se trata de construir máquinas que compitan con los seres humanos, sino que realicen ciertas tareas de rango intelectual con que ayudarle, principio básico de la Inteligencia Artificial.

Las primeras explicaciones teóricas sobre el cerebro y el **pensamiento** ya fueron dadas ya por **Platón** (427-347 a.C.) y **Aristóteles** (348-422 a.C.). Las mismas ideas también las mantuvo **Descartes** (1569-1650) y los **filósofos** empiristas del siglo XVIII.

La clase de las llamadas *máquinas cibernéticas*, a la cual la computación neuronal pertenece, tiene más historia de la que se cree: **Herón** (100 a.C) construyó un autómata hidráulico. Los primeros ejemplos de autómatas se registran en la antigua Etiopía. En el año 1500 a. C., Amenhotep, hermano de Hapu, construye una estatua de Memon, el rey de Etiopía, que emite sonidos cuando la iluminan los rayos del sol al amanecer. **King-su Tse**, en China, en el 500 a. C. inventa una urraca voladora de madera y bambú y un



caballo de madera que saltaba. Entre el 400 y 397 a. C., Archytar de Tarento construye un pichón de madera suspendido de un pivote, el cual rotaba con un surtidor de agua o vapor, simulando el vuelo. Archytar es el inventor del tornillo y la polea. En el año 206 a. C., fué encontrado el tesoro de Chin Shih Hueng Ti consistente en una orquesta mecánica de muñecos, encontrada por el primer emperador Han.

1936 - Alan Turing. Fue el primero en estudiar el cerebro como una forma de ver el mundo de la computación. Sin embargo, los primeros teóricos que concibieron los fundamentos de la computación neuronal fueron Warren McCulloch, un neurofisiólogo, y Walter Pitts, un matemático, quienes, en 1943, lanzaron una teoría acerca de la forma de trabajar de las neuronas (Un Cálculo Lógico de la Inminente Idea de la Actividad Nerviosa - Boletín de Matemática Biofísica 5: 115-133). Ellos modelaron una red neuronal simple mediante circuitos eléctricos.

1943 - Teoría de las Redes Neuronales Artificiales. Walter Pitts junto a Bertran Russell y Warren McCulloch intentaron explicar el funcionamiento del cerebro humano, por medio de una red de células conectadas entre sí, para experimentar ejecutando operaciones lógicas. Partiendo del menor suceso psíquico (estimado por ellos): el impulso todo/nada, generado por una célula nerviosa.

El bucle "sentidos - cerebro - músculos", mediante la retroalimentación producirían una reacción positiva si los músculos reducen la diferencia entre una condición percibida por los sentidos y un estado físico impuesto por el cerebro.

También definieron la memoria como un conjunto de ondas que reverberan en un circuito cerrado de neuronas.

1949 - Donald Hebb. Escribió un importante libro: La organización del comportamiento, en el que se establece una conexión entre psicología y fisiología. Fue el primero en explicar los procesos del aprendizaje (que es el elemento básico de la inteligencia humana) desde un punto de vista psicológico, desarrollando una regla de como el aprendizaje ocurría. Aun hoy, este es el fundamento de la mayoría de las funciones de aprendizaje que pueden hallarse en una red neuronal. Su idea fue que el aprendizaje ocurría cuando ciertos cambios en una neurona eran activados. También intentó encontrar semejanzas entre el aprendizaje y la actividad nerviosa. Los trabajos de Hebb formaron las bases de la Teoría de las Redes Neuronales.

1950 - Karl Lashley. En sus series de ensayos, encontró que la información no era almacenada en forma centralizada en el cerebro sino que era distribuida encima de él.

1951 - Primera Red Neuronal. El extraordinario estudiante de Harvard, Marvin Minsky conoció al científico Burrhus Frederic Skinner, con el que trabajó algún tiempo ayudándole en el diseño y creación de máquinas para sus experimentos. Minsky se inspiró en Skinner para gestar su primera idea "oficial" sobre inteligencia artificial, su Red Neuronal. Por aquel entonces entabló amistad con otro brillante estudiante, Dean Edmonds, el cual estaba interesado en el estudio de una nueva ciencia llamada Electrónica.

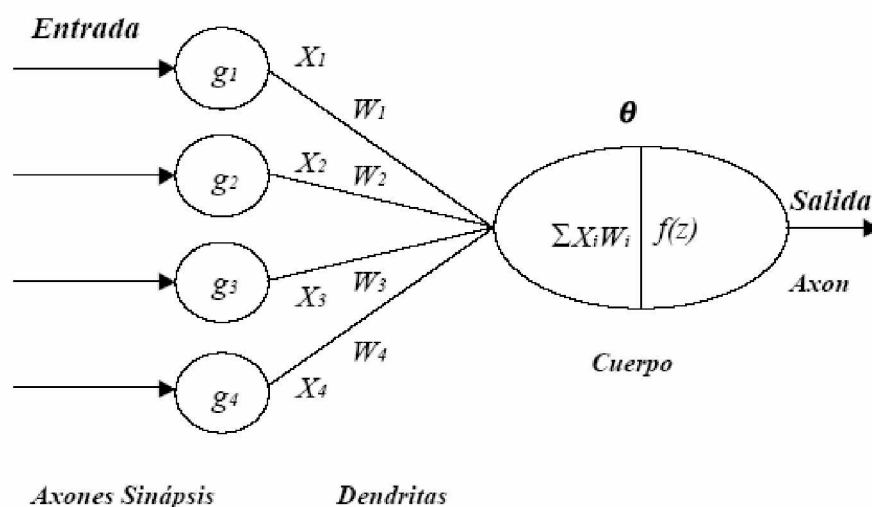
Durante el verano de 1951, Minsky y Edmonds montaron la primera máquina de redes neuronales, compuesta básicamente de 300 tubos de vacío y un piloto automático de un bombardero B-24. Llamaron a su creación "Sharc", se trataba nada menos que de una red de 40 neuronas artificiales que imitaban el cerebro de una rata. Cada neurona hacía el papel de una posición del laberinto y cuando se

activaba daba a entender que la "rata" sabía en que punto del laberinto estaba. Las neuronas que estaban conectadas alrededor de la activada, hacían la función de alternativas que seguir por el cerebro, la activación de la siguiente neurona, es decir, la elección entre "derecha" o "izquierda" en este caso estaría dada por la fuerza de sus conexiones con la neurona activada. Por ejemplo, la "rata" completaba bien el recorrido eligiendo a partir de la quinta neurona la opción "izquierda" (que correspondería a la sexta), es entonces cuando las conexiones entre la quinta y sexta se hacen más fuertes (dicha conexión era realizada por el piloto automático), haciendo desde este momento más propensa esta decisión en un futuro. Pero las técnicas Skinnerianas (que eran las que se habían puesto en funcionamiento en esta red neuronal) no podrían llevar muy lejos a este nuevo engendro, la razón pasa porque esto, en sí, no es inteligencia, pues la red neuronal nunca llegaría a trazar un plan.

Después de su Red Neuronal, Minsky escribió su tesis doctoral acerca de esta, en ella describía "cerebros mucho mayores", exponiendo que si se realizaba este proyecto a gran escala, con miles o millones de neuronas más y con diferentes sensores y tipos de retroalimentación... la máquina podría ser capaz de razonar, mas el sabía que la realización de esta Red Neuronal era imposible y decidió buscar otra forma de crear inteligencia...

1956 - Congreso de Dartmouth. Este Congreso frecuentemente se menciona para indicar el nacimiento de la inteligencia artificial.

1957 - Frank Rosenblatt. Comenzó el desarrollo del Perceptrón. Esta es la red neuronal más antigua; utilizándose hoy en día para aplicación como reconocedor de patrones. Este modelo era capaz de generalizar, es decir, después de haber aprendido una serie de patrones podía reconocer otros similares, aunque no se le hubiesen presentado anteriormente. Sin embargo, tenía una serie de limitaciones, por ejemplo, su incapacidad para resolver el problema de la función OR-exclusiva y, en general, era incapaz de clasificar clases no separables linealmente. En 1959, escribió el libro Principios de Neurodinámica, en el que confirmó que, bajo ciertas condiciones, el aprendizaje del Perceptrón convergía hacia un **estado** finito (Teorema de Convergencia del Perceptrón).



Modelo de un Perceptrón

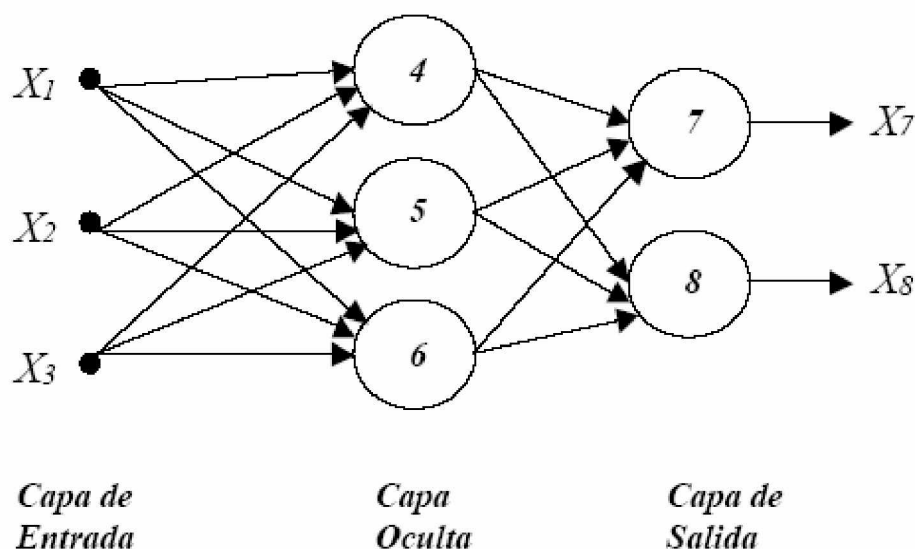
1960 - Bernard Widrow/Marcial Hoff. Desarrollaron el modelo Adaline (ADAPtative LINear Elements). Esta fue la primera red neuronal aplicada a un problema real (filtros adaptativos para eliminar ecos en las líneas telefónicas) que se ha utilizado comercialmente durante varias décadas.

1961 - Karl Steinbeck: Die Lernmatrix. Red neuronal para simples realizaciones técnicas (memoria asociativa).

1967 - Stephen Grossberg. A partir de sus conocimientos fisiológicos, ha escrito numerosos libros y desarrollado modelo de redes neuronales. Realizó una red: Avalancha, que consistía en elementos discretos con actividad que varía en el tiempo que satisface ecuaciones diferenciales continuas, para resolver actividades como reconocimiento continuo de habla y aprendizaje de los brazos de un robot.

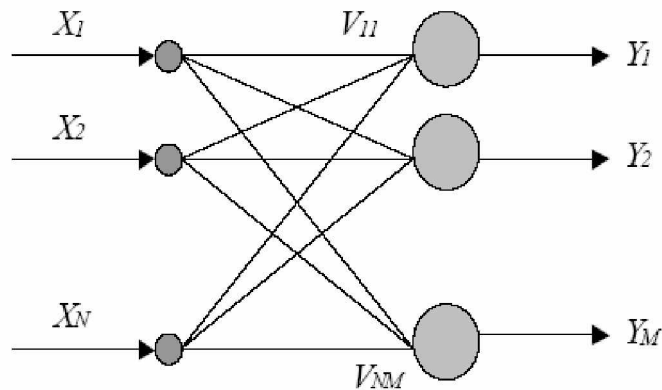
1969 - Marvin Minsky/Seymour Papert. En este año surgieron críticas que frenaron, hasta 1982, el crecimiento que estaban experimentando las investigaciones sobre redes neuronales. Minsky y Papera, del Instituto Tecnológico de Massachussets (MIT), publicaron un libro Perceptrons. Probaron (matemáticamente) que el Perceptrón no era capaz de resolver problemas relativamente fáciles, tales como el aprendizaje de una función no-lineal. Esto demostró que el Perceptrón era muy débil, dado que las funciones no-lineales son extensamente empleadas en computación y en los problemas del mundo real. A pesar del libro, algunos investigadores continuaron su trabajo. Tal fue el caso de James Anderson, que desarrolló un modelo lineal, llamado Asociador Lineal, que consistía en unos elementos integradores lineales (neuronas) que sumaban sus entradas. Este modelo se basa en el principio de que las conexiones entre neuronas son reforzadas cada vez que son activadas. Anderson diseñó una potente extensión del Asociador Lineal, llamada Brain State in a Box (BSB).

1974 - Paul Werbos. Desarrolló la idea básica del algoritmo de aprendizaje de propagación hacia atrás (backpropagation); cuyo significado quedó definitivamente aclarado en 1985.



Modelo de una red tipo *Backpropagation*

1977 - [Stephen Grossberg](#). Teoría de Resonancia Adaptada (TRA). La Teoría de Resonancia Adaptada es una arquitectura de red que se diferencia de todas las demás previamente inventadas. La misma simula otras habilidades del cerebro: memoria a largo y corto plazo.



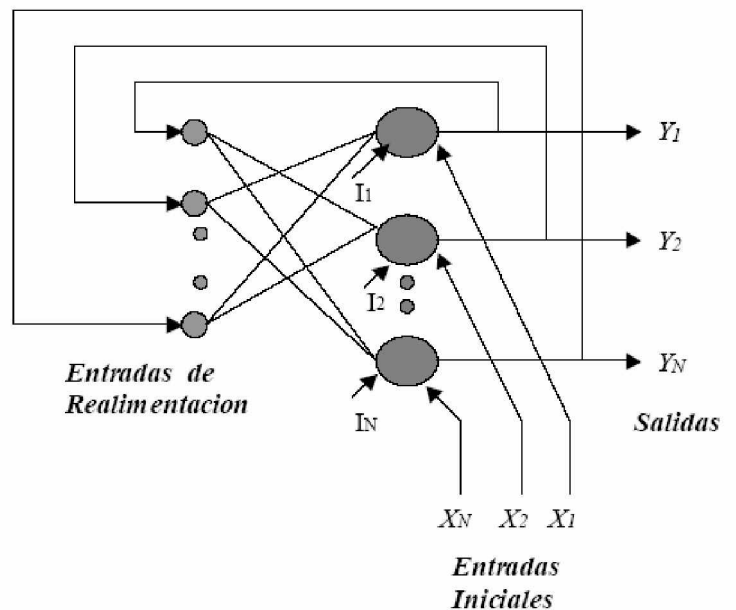
Modelo de red neuronal *Kohonen*

1977 - [Teuvo Kohonen](#). Ingeniero electrónico de la [Universidad de Helsinki](#), desarrolló un modelo similar al de Anderson, pero independientemente.

1980 - [Kunihiko Fukushima](#). Desarrolló un modelo neuronal para el reconocimiento de patrones visuales.

1985 - [John Hopfield](#). Provocó el [renacimiento](#) de las redes neuronales con su libro: "Computación neuronal de decisiones en problemas de optimización."

1986 - [David Rumelhart/G. Hinton](#). Redescubrieron el algoritmo de aprendizaje de propagación hacia atrás (backpropagation). A partir de 1986, el panorama fue alentador con respecto a las investigaciones y el desarrollo de las redes neuronales. En la actualidad, son numerosos los trabajos que se realizan y publican cada año, las aplicaciones nuevas que surgen (sobre todo en el área de control) y las [empresas](#) que lanzan al [mercado](#) productos nuevos, tanto [hardware](#) como [software](#) (sobre todo para [simulación](#)).



Modelo de *Hopfield*

A partir de 1986, el panorama fue alentador con respecto a las investigaciones y el desarrollo de las redes neuronales. En la actualidad, son numerosos los trabajos que se realizan y publican cada año, las aplicaciones nuevas que surgen (sobre todo en el área de control) y las empresas que lanzan al mercado productos nuevos, tanto hardware como software (sobre todo para simulación).