



# Lic. Ciencias de la Computación



UMT

Aprendizaje por refuerzo

Tabla de enrutamiento

Redes Neuronales

Y mucho más...

Redes Semánticas

Dispositivos Móviles

Wolfram

MySQL

PROLOG LTD



# Directorio

## Director

M.C. Michel García García

## Comité Editorial

M.C.C. Michel García García

M.C.C. Cinhtia Maribel González Segura

M.C.C. Víctor Manuel Chi Pech

M.C.C. Lizzie Edmea Narváez Díaz

LM. Teresita del Jesús Montañez May

LM. Manuel Jesús David Escalante Torres

M.C.C. Erika Rosanna Llanes Castro

M.C.C. Carlos Andrés Miranda Palma

M.C.C. Sergio Alejandro González Segura

M.C.C. Maximiliano Canche Euan

M.C.C. Luis Colorado Martínez

Dr. Jose Luis López Martínez

**Diseño de Portada:** Br. Landy G. López Gómez

**Colaboradores Invitados:** Br. Fanny del Rosario Pérez Pech

Br. María Rebeca Aguayo Navarro

Editores

Michel García García

Cinhtia González Segura

Maximiliano Canche Euán

# Contenido

<b>Editorial</b> .....	1
Cinhtia González Segura	
<b>La tabla de enrutamiento en el proceso de toma de decisiones para reenvío de paquetes</b> .....	3
Lizzie E. Narváez Díaz y Victor M. Chi Pech	
<b>Historia de las Redes Neuronales artificiales</b> .....	8
Michel García García, Cinhtia González Segura y Sergio González Segura	
<b>Redes semánticas para representar el conocimiento</b> .....	14
Carlos Miranda Palma	
<b>Lenguaje de programación Wolfram</b> .....	19
José Luis López Martínez	
<b>Aprendizaje por refuerzo</b> .....	23
Michel García García, Cinhtia González Segura y Sergio González Segura	
<b>Aplicaciones de salud en dispositivos móviles</b> .....	28
Maximiliano Canché Euán y Erika Llanes Castro	
<b>Colabora con Nosotros</b> .....	32

# Editorial

## INICIANDO CON ACTITUD

Por: Cinhtia González Segura

Estamos llegando a la recta final del primer quindenio de este siglo, época en la que ve la luz nuestro décimo número de la revista. Con este ejemplar cumplimos también nuestro primer lustro trabajando en ella, lo cual nos ha permitido estar en contacto con todos ustedes, nuestros amables lectores, a quienes agradecemos estar pendientes de cada publicación.

En los últimos ejemplares realizados, hemos podido abordar diversas temáticas específicas relacionadas con las ciencias de la computación. En particular, ahora presentamos información del área de redes de computadoras, inteligencia artificial, tecnología móvil, programación, tecnología educativa, estadística y algunos otros temas. Nos enorgullece decir que en ocasiones se incluye información proveniente de avances de proyectos en los que han colaborado nuestros estudiantes universitarios, ya sea como parte del trabajo realizado en alguna asignatura optativa o bien en algún tema de investigación.

Cabe recordar que nuestro interés principal consiste en introducir al lector en el mundo moderno y sofisticado de la tecnología computacional, por lo que es importante profundizar posteriormente en aquél tópico de interés particular, poniendo en práctica nuestras habilidades para buscar, obtener, procesar y compartir la información, que se transformará en parte de nuestro conocimiento.



*"No hay mayor signo de demencia que hacer la misma cosa una y otra vez y esperar que los resultados sean distintos".*

**Albert Einstein**

Esperemos que los avances que se van generando paulatinamente sean para beneficio de nuestra sociedad; una sociedad en la que todavía quedan muchas problemáticas latentes por resolver. Por lo tanto, recordemos que siempre es posible aportar un granito de arena en aquél área que identifiquemos adecuado para nosotros, considerando nuestros intereses y capacidades.

Debido al vertiginoso avance de las ciencias de la computación, en los siguientes ejemplares encontrarás diversos y novedosos tópicos. Desde luego, si deseas colaborar con algún artículo o te interesa que se incluya alguna temática en particular, no dudes en contactarnos.

Como universitarios y profesionales, mantengamos firme nuestro compromiso por compartir la información novedosa que obtenemos, esto nos permitirá mantenernos a la vanguardia y lograr nuevas mejoras que redunden en beneficio de nuestra comunidad y de la sociedad en general.

**¡Excelente inicio de semestre!**



# La tabla de enrutamiento en el proceso de toma de decisiones para reenvío de paquetes.

Por: Lizzie Narváez Díaz y Víctor Chi Pech

Tal como lo hacen los dispositivos finales de red, un router incorpora información de las rutas que tiene directamente conectadas a una tabla conocida como tabla de enrutamiento. De igual manera, también incorporan en dicha tabla información acerca de las redes remotas conectadas y que ha aprendido a través de un protocolo de enrutamiento estático o dinámico. Por lo tanto, podemos considerar una tabla de enrutamiento como un archivo de datos almacenado en la RAM usado para almacenar información de rutas sobre redes directamente conectadas y redes remotas.

En terminología de redes de computadoras, una red directamente conectada es aquella que está unida directamente a una de las interfaces de un router, las cuales son los gateways para los host en las distintas redes locales. Cuando configuramos la interfaz de un router con una dirección IP y una máscara de subred, la interfaz se convierte en parte de esa red, y en consecuencia la tabla de enrutamiento incluye esa red como si estuviera conectada directamente.

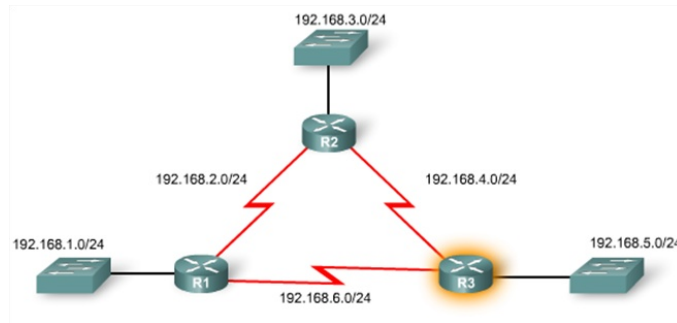
Por otra parte, las redes remotas son aquellas que no están directamente conectadas al router. Estas redes se añaden a la tabla de enrutamiento usando un protocolo de enrutamiento dinámico, o configurando rutas estáticas. Las rutas dinámicas son rutas a redes remotas que se aprenden automáticamente por el router usando un protocolo de enrutamiento dinámico, como por ejemplo RIP, EIGRP, OSPF, etc. Las rutas estáticas son rutas a redes que pueden ser configuradas manualmente por un administrador de red.



*"La matemática es la ciencia del orden y la medida, de bellas cadenas de razonamientos, todos sencillos y fáciles".*

**René Descartes**

Por ejemplo, en la siguiente figura para el router denominado R3 sus redes directamente conectadas son 192.168.4.0/24 y 192.168.5.0/24 y sus redes remotas son: 192.168.1.0/24, 192.168.2.0/24 y 192.168.3.0/24.



La información almacenada en una tabla de enrutamiento es de vital importancia para la comunicación ya que para reenviar un paquete un router necesita saber si puede o no enviarlo y a dónde, y esto sólo lo puede decidir con base a la información de su tabla.

Una entrada de ruta, o de red de destino, en la tabla de enrutamiento representa un rango de direcciones de host, y a veces, también de redes. Para examinar el contenido de la tabla de enrutamiento de un router se requiere del comando show iproute en el modo de configuración de ejecución privilegiado. Por ejemplo, para la topología presentada anteriormente y asumiendo que ya se han configurados las interfaces de cada uno de los routers y el protocolo de enrutamiento dinámico RIP, el resultado de la tabla de enrutamiento para el router denominado R3 es similar a:

```
R3#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

R    192.168.1.0/24 [120/1] via 192.168.6.2, 00:00:05, Serial0/0/0
R    192.168.2.0/24 [120/1] via 192.168.6.2, 00:00:05, Serial0/0/0
                               [120/1] via 192.168.4.2, 00:00:05, Serial0/0/1
R    192.168.3.0/24 [120/1] via 192.168.4.2, 00:00:05, Serial0/0/1
C    192.168.4.0/24 is directly connected, Serial0/0/1
C    192.168.5.0/24 is directly connected, FastEthernet0/0
C    192.168.6.0/24 is directly connected, Serial0/0/0
```



¿Cómo se interpreta el resultado de la tabla de enrutamiento? Analicemos dos entradas de ruta: una que corresponda a una red remota aprendida por un protocolo de enrutamiento dinámico (en este caso RIP) y otra que corresponda a una red directamente conectada.

Respecto a la red remota, consideremos la entrada de ruta correspondiente a la red 192.168.1.0/24, que aparece en la tabla de enrutamiento como:

```
R 192.168.1.0/24 [120/1] via 192.168.6.2, 00:00:05, Serial0/0/0
```

El significado de la línea anterior es el siguiente:

- La "R" indica que la ruta fue aprendida por la configuración del protocolo de enrutamiento dinámico RIP.
- A continuación, se enumeran la dirección de la red remota con su máscara de subred (192.168.1.0/24).
- Entre paréntesis se muestra el valor de la distancia administrativa (120 para el RIP) y la métrica de dicho protocolo de enrutamiento, es decir, su distancia a la red (en este caso 1 salto).
- Posteriormente se enumera la dirección IP del siguiente salto del router que realiza la publicación (R1 en 192.168.6.2) y la cantidad de segundos que pasaron desde la última actualización (00:00:05, en este caso).
- Por último, se indica la interfaz de salida que utilizará este router para el tráfico destinado a la red remota (Serial 0/0/0).

Respecto a la red directamente conectada, consideremos la entrada de ruta correspondiente a la red 192.168.4.0/24, que aparece en la tabla de enrutamiento como:

```
C 192.168.4.0/24 is directly connected, Serial0/0/1
```



*"Los encantos de esta ciencia sublime, las matemáticas, sólo se le revelan a aquellos que tienen el valor de profundizar en ella".*

**Carl Friedrich Gauss**



El significado de la línea anterior es el siguiente:

- La “C” significa que el origen de la información de la ruta corresponde a una red conectada directamente.
- Posteriormente se muestra la dirección de red y la máscara de subred de la red conectada directamente, en este caso 192.168.4.0/24.
- La información al final de la entrada de la ruta representa la interfaz de salida. En este ejemplo, la interfaz serial 0/0/1 es la interfaz de salida usada para alcanzar esta red.

Es importante señalar que las rutas estáticas y dinámicas no pueden existir en la tabla de enrutamiento sin las redes conectadas directamente propias del router.

Después de haber analizado el contenido de las rutas directamente conectadas y remotas, podemos mencionar de manera general que las rutas de una tabla de enrutamiento tienen tres características fundamentales: la red de destino, el siguiente salto y la métrica. Esta información es usada por el router ya que primero éste compara la dirección de destino contenida en la cabecera de un paquete con la red de destino de una ruta de la tabla de enrutamiento. Posteriormente, el router reenvía el paquete al router de siguiente salto especificado por esa ruta. En caso de que existan dos o más caminos al mismo destino, el router usa la métrica para decidir la ruta que aparecerá en la tabla de enrutamiento.

El reenvío de los paquetes implica dos funciones: la determinación de la ruta y la conmutación.

La **determinación de la ruta** es el proceso según el cual el router determina qué ruta usar cuando reenvía un paquete. Para determinar el mejor camino, el router busca en su tabla de enrutamiento una dirección de red que coincida con la dirección IP de destino del paquete.



El resultado de esta búsqueda es una de cuatro determinaciones de ruta:

- Red conectada directamente: si la dirección IP de destino del paquete pertenece a un dispositivo en una red que está conectada directamente a una de las interfaces del router, ese paquete se reenvía directamente a ese dispositivo. Esto significa que la dirección IP de destino del paquete es una dirección host en la misma red que la interfaz de este router.
- Red remota: si la dirección IP de destino del paquete pertenece a una red remota, entonces el paquete se reenvía a otro router. Sólo se pueden alcanzar las redes remotas mediante el reenvío de paquetes hacia otra red.
- Ruta predeterminada: si la tabla de enrutamiento no dispone de una ruta más específica para el paquete entrante, éste se reenvía a la interfaz indicada por la ruta predeterminada, en caso de que exista. En esta interfaz, se encapsula el paquete y se envía al router de siguiente salto.
- Sin determinación de ruta: si la dirección IP de destino del paquete no pertenece ni a una red conectada ni a una remota, y si el router no tiene una ruta predeterminada, entonces el paquete se descarta. El router envía un mensaje ICMP de destino inalcanzable a la dirección IP de origen del paquete.

Después de que el router ha determinado la interfaz de salida utilizando la función de determinación de ruta, el router debe encapsular el paquete en la trama de enlace de datos de la interfaz de salida. La **función de conmutación** es el proceso utilizado por un router para aceptar un paquete en una interfaz y reenviarlo hacia otra interfaz. Una responsabilidad de la función de conmutación es la de encapsular los paquetes en el tipo de trama de enlace de datos correcto para el enlace de datos de salida.

### Referencias:

Mark A. Dye; Rick McDonald; Antoon W. Ruff. Aspectos básicos de networking. Guía de estudio de CCNA Exploration. Cisco Press. 2008.

Rick Graziani; Allan Johnson. Conceptos y protocolos de enrutamiento. Guía de estudio de CCNA Exploration. Cisco Press. 2008.

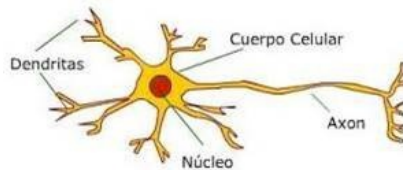
Ernesto Ariganello. Redes Cisco. Guía de estudio para la certificación. CCNA Routing y Switching. Alfaomega Ra-Ma. 2014.

# Historia de las Redes Neuronales artificiales

Por: Sergio Alejandro González Segura, Cinhtia Maribel González Segura, Michel García García.

¿En dónde radican las más altas funciones mentales, como son: aprendizaje, razonamiento, lenguaje, creatividad y conciencia? Actualmente se cree que éstas son funciones originadas en el cerebro humano, más específicamente en la corteza cerebral, aunque recientes estudios sugieren que una parte de la inteligencia podría deberse incluso a interacciones hormonales.

La unidad funcional básica del cerebro es la neurona natural [1], la cual es una célula con 3 partes funcionales diferenciadas: dendritas, núcleo y axón. Se cree que una neurona natural recibe información por las dendritas, la procesa en su núcleo, y emite información por el axón. Aunque actualmente hay una investigación que sugiere que las dendritas también son capaces de realizar cálculos [2].



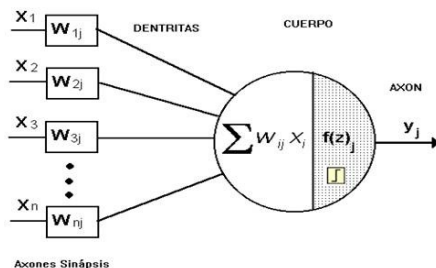
*Fig 1. Neurona natural*

Las redes neuronales artificiales (RNA) surgieron como un intento de formalizar matemáticamente la estructura y funcionamiento del cerebro. Después se utilizaron para intentar dotar de inteligencia “humana” a las máquinas. Uno podría pensar que es una meta muy ambiciosa, pero recordemos que la máquina de Turing, inspiración de las computadoras, surgió como un intento de modelar todo lo que una mente humana podía hacer, y así investigar los límites de la inteligencia humana.



Los primeros modelos matemáticos que imitaban el comportamiento de las neuronas se deben a Warren McCulloch y Walter Pitts (neurofisiólogo y matemático respectivamente), que presentaron en 1943 el primer modelo de una neurona artificial [3].

En 1957 Frank Rosenblatt creó el perceptrón [4], un modelo formado por una sola neurona artificial y que era capaz de aprender patrones, es decir, después de enseñársele varios ejemplos, era capaz de reconocer uno similar que no se le hubiera mostrado previamente, es decir, una vez entrenado es un clasificador.



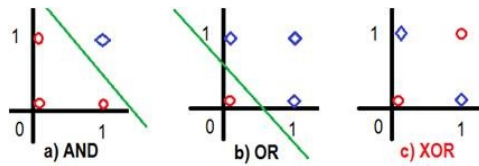
**Fig 2.** Perceptrón, tomado de:

<http://electronica.com.mx/neural/informacion/perceptron.html>

La función de activación (“f”) es no lineal (por ejemplo, si se supera un umbral entonces la entrada pertenece al patrón), sin embargo los datos se procesan de forma lineal (la suma de productos), lo cual significa que el perceptrón conceptualmente divide el espacio de entrada con un hiperplano (por ejemplo: si sólo hay dos entradas, el espacio es un plano y se divide con una línea), y acepta las entradas que quedan de un lado y rechaza a las que quedan del otro.

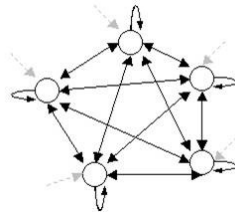
Esto significa que un perceptrón puede aprender las funciones AND, OR, NOT, pero no puede aprender la función XOR. Ésta fue la base de una fuerte crítica en 1969 en el libro “Perceptrons”, por Marvin Minsky y Seymour Papert, del Instituto Tecnológico de Massachussets (MIT), y probablemente por el renombre de esos autores, el interés por las RNA (redes neuronales artificiales) decreció enormemente por varios años. Una respuesta a la crítica pudo ser, que reconocían el 75% de los casos de prueba de la función XOR (en la Figura 3, inciso “c”, es posible dibujar una línea que clasifique 3 casos correctamente y uno incorrectamente), que en ciertos casos son bastante buenos.





**Fig 3.** El fracaso del perceptrón para aprender la función XOR

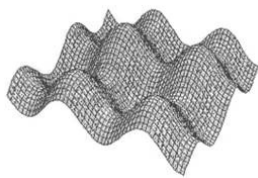
Despertaron de nuevo el interés varios trabajos [5] como la “Red de Hopfield” [6], en 1982, la cual es una variante del asociador lineal pero inspirada en la reducción de energía de sistemas físicos; los “Mapas de Kohonen” en 1984, los cuales usan una idea tomada de la biología: las unidades de procesos físicamente adyacentes aprenderán a representar patrones similares; y es muy notable la formulación en 1986 del algoritmo “backpropagation” (retro propagación) por David Rumelhart y otros [7], el cual permite un eficiente aprendizaje en las redes multicapa de perceptrones, permitiendo aprender funciones no lineales como la XOR.:



**Fig 4.** Red de Hopfield [6]

A una red de Hopfield se le presenta una entrada, y la red lo aprende. La información se propaga hacia adelante y hacia atrás hasta que se llega a un estado estable. Experimentos empíricos han demostrado que se puede aprender una entrada cada 7 neuronas [6]. Una vez aprendido, se le puede presentar una porción de la entrada, y la red es capaz de reconstruir la parte faltante. Por ejemplo: puede reconocer o reconstruir imágenes [8]. Este modelo proporciona una explicación a cómo ocurren las asociaciones de ideas (o recuerdos) en el cerebro.





**Fig 5.** Función de energía de una red Hopfield [8]

La función de energía de una red Hopfield puede representarse con una manta tridimensional [8], donde los mínimos locales son los recuerdos, y las regiones intermedias las entradas incompletas.

El algoritmo backpropagation permite el entrenamiento de redes multicapa de perceptrones, en las cuales las neuronas están organizadas en capas. La capa inferior se conecta a los datos de entrada, las capas intermedias reciben información de una capa anterior y proporcionan información a la capa siguiente, y la última capa proporciona la salida. Las neuronas de una capa no se conectan con otras de la misma capa, ni reciben información de las capas superiores, excepto durante el aprendizaje. Ejemplos de éstas y sus capacidades pueden verse en la Figura 6.

Estructura	Tipo de región de decisión	Problema del OR-Exclusivo	Clases in no separables	Formas más generales
<p>Una capa</p>	Zonas separadas por hiperplanos			
<p>Dos capas</p>	Zonas convexas			
<p>Tres capas</p>	Zonas de complejidad arbitraria			

**Fig 6.** Capacidades del perceptrón multicapa [9].



Las redes neuronales artificiales no tienen que existir físicamente, con mucho menor esfuerzo pueden ser simuladas en las computadoras con prácticamente el mismo resultado que si se construyeran físicamente. No se cree que exista una diferencia esencial entre una red neuronal física y una simulada, lo que ha llevado a pensar que las mentes radicadas en los cerebros de las personas puedan ser descargadas a una computadora y "simularse", produciendo un comportamiento prácticamente igual a cuando existían en las personas... incluso se calculó cuál debería ser la capacidad de una computadora para albergar y simular a un cerebro humano ( $10^2$  TB y  $10^{15}$  Flops) [10][11]. Sin embargo, si las dendritas contribuyen al cálculo [2], la capacidad de cómputo necesaria para simular un cerebro humano aumenta en varios órdenes de magnitud.

Actualmente las redes neuronales se utilizan en aprendizaje máquina con buenos resultados, sin embargo, suelen perder la batalla cuando existe un algoritmo especializado que resuelva el mismo problema. Su mayor ventaja es que pueden resolver un problema bastante bien cuando no se tiene un algoritmo especializado.

### Referencias:

[1] "Neurona y conducción nerviosa".

<http://www.educarchile.cl/ech/pro/app/detalle?ID=137486> . Consultado en diciembre de 2015.

[2] "Smart neurons: single dendrites can perform computations".

<http://www.ucl.ac.uk/news/news-articles/1310/28102013-Smart-neurons-Single-neuronal-dendrites-can-perform-computations-Hausser#sthash.Fh5sCOw1.dpuf> . En español: [http://noticiasdelaciencia.com/not/8791/una\\_dendrita\\_de\\_una\\_neurona\\_es\\_capaz\\_de\\_hacer\\_algunos\\_calculos\\_por\\_su\\_cuenta/](http://noticiasdelaciencia.com/not/8791/una_dendrita_de_una_neurona_es_capaz_de_hacer_algunos_calculos_por_su_cuenta/) . Consultado en diciembre de 2015.



*"Nuestra recompensa se encuentra en el esfuerzo y no en el resultado. Un esfuerzo total es una victoria".*

**Mahatma Gandhi**

[3] Warren S. McCulloch and Walter H. Pitts. "A logical calculus of the ideas immanent in nervous activity". Originally published in: Bulletin of Mathematical Biophysics, Vol.5, 1943, p. 115-133. <http://www.cse.chalmers.se/~coquand/AUTOMATA/mcp.pdf> . Consultado endiciembre de 2015.

[4] Rosenblatt, F. "The perceptron: a probabilistic model for information storage and organization in the brain". Psychological review, 65(6), 386. 1958.

[5] Raquel Flórez López, José Miguel Fernández Fernández. "Las Redes Neuronales Artificiales". Editor: Netbiblo, 2008. ISBN: 8497452461, 9788497452465.

[6] "Redes de Hopfield". <http://avellano.fis.usal.es/~lalonso/RNA/introHOP.htm> . Consultado endiciembre de 2015.

[7] David E. Rumelhart, Geoffrey E. Hilton, Ronald J. Williams. "Learning representations by back-propagating errors". Nature Vol. 323, 9 de octubre de 1986. Pj. 533-536.

[8] "Procesamiento de Imágenes con Redes Neuronales". [http://www.sav.us.es/formaciononline/asignaturas/asigpid/apartados/textos/recursos/redes neuronales/PIRN.htm](http://www.sav.us.es/formaciononline/asignaturas/asigpid/apartados/textos/recursos/redes_neuronales/PIRN.htm) . Consultado en diciembre de 2015.

[9] "Redes Multicapa". <ftp://decsai.ugr.es/pub/usuarios/castro/Actividades/Redes-Neuronales/Apuntos/Apuntos%20Javier%20Rodriguez%20Blazquez/Redes%20Multicapa%20-%20Algoritmo%20Backpropagation.pdf> . Consultado en diciembre de 2015.

[10] "Minduploading". [https://en.wikipedia.org/wiki/Mind\\_uploading](https://en.wikipedia.org/wiki/Mind_uploading) . Consultado endiciembre de 2015.

[11] Anders Sandberg;Nick Boström. "Whole Brain Emulation: A Roadmap". Technical Report. Future of Humanity Institute, Oxford University. 2008. "The basic idea is to take a particular brain, scan its structure in detail, and construct a software model of it that is so faithful to the original that, when run on appropriate hardware, it will behave in essentially the same way as the original brain". <http://www.fhi.ox.ac.uk/brain-emulation-roadmap-report.pdf> . Consultado en diciembre de 2015.

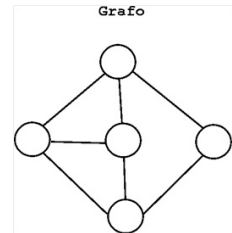


# Redes semánticas para representar el conocimiento

Por: Carlos Miranda Palma.

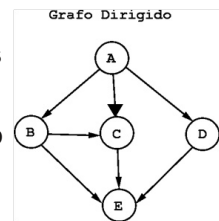
Las limitaciones para representar el conocimiento, en particular, la necesidad de representar aspectos como estructura y relaciones, llevaron a otros esquemas que en general se engloban como representaciones estructuradas. Dentro de este tipo de representaciones las dos más significativas, son: Redes Semánticas y los prototipos o Marcos (*frames*).

Estas representaciones se basan en el uso de grafos, es decir, en representaciones en base a nodos y sus relaciones, por lo que comentaremos algo de la teoría de grafos, su notación y conceptos básicos.



Un grafo consiste de un conjunto finito de vértices o nodos ( $V$ ), y un conjunto de relaciones entre dichos nodos ( $E$ ).  $E$  representa qué nodos son adyacentes, es decir si  $(v,w)$  pertenece a  $E$ , entonces  $v$  y  $w$  son adyacentes y esto se representa con un arco entre dichos nodos.

Si las relaciones tienen una dirección, esto es, el orden en los pares  $(v,w)$  es importante entonces  $G$  es un **grafo dirigido**. Si restringimos a que no existan ciclos, se denomina **grafo acíclico dirigido o grafo simple (DAG)**.



Dada una relación dirigida  $(v,w)$ :  $v \rightarrow w$ ,  $v$  es el padre de  $w$  y  $w$  es el hijo de  $v$ . Si existe una trayectoria de  $v$  a  $w$ , entonces  $w$  es un descendiente de  $v$ , y  $v$  es un ascendiente de  $w$ . Si  $v$  no tiene padres es un nodo raíz, si  $w$  no tiene hijos es un nodo hoja.



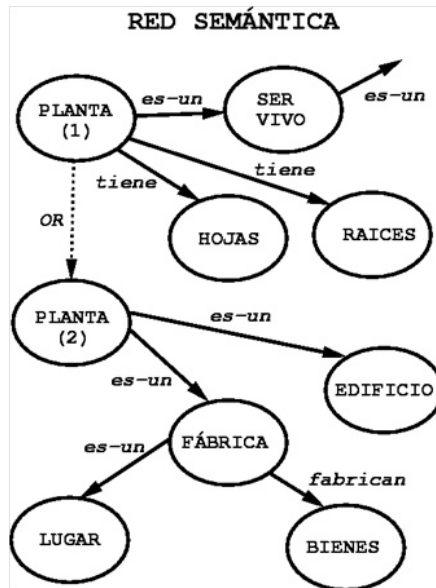
Una **red** es un grafo con pesos, es decir, es una red con etiquetas numéricas o pesos asociada a los arcos y/o nodos.

Una **Red Semántica** es un modelo de memoria humana para capturar la semántica de las palabras y lograr un uso del significado parecido a los humanos.

Es un tipo de red en la cual los nodos representan objetos, conceptos o situaciones y los arcos representan relaciones entre ellos. Se llama red semántica porque se usaron originalmente para representar el sentido en expresiones de lenguaje natural.

En las redes semánticas los nodos representan los conceptos de palabras y los arcos ligan conceptos para establecer la definición. Cada palabra o nodo conceptual se consideraba la cabeza de un “plano” que tiene su definición (v.g., si *banco* tiene 3 significados, entonces existen 3 planos para él).

Las ligas en el plano representan su definición. Apuntadores fuera del plano hacen referencia a otros objetos (y planos) en donde se definen.





Veamos un ejemplo:

El problema que se quiere resolver consiste en darle un comando al robot móvil Nomad Scout que este sea capaz de monitorear e identificar los objetos que se encuentran en el ambiente y con esa información ejecutar una instrucción o elegir una opción diferente para evitar la colisión con algún obstáculo (planeación). Las *entradas* serían los comandos que el usuario le indica al robot. El *resultado esperado* sería que el robot ejecute el comando o realice otro comando evitando colisiones.

A continuación presentamos la representación de este problema utilizando una red semántica:



*"Si la gente no piensa que las matemáticas son simples, es solo porque no se dan cuenta de lo complicada que es la vida".*

**John Von Neumann**

Para este problema la representación por redes semánticas nos ayuda a entender con mejor claridad los conceptos que se emplean en el problema y la forma en la que se puede manejar estos conceptos.

### Referencias:

- Shafer y Pearl (Eds.), *Reading in Uncertain Reasoning*, Morgan-Kaufmann, 1990.
- Whittaker, *Graphical Models*, Wiley, 1990.
- F. Jensen, *Bayesian Networks and Decision Graphs*, Springer, 2001.
- C. Borgelt, R. Kruse, *Graphical Models*, Wiley, 2002.



*"Sería posible describir todo científicamente, pero no tendría ningún sentido; carecería de significado el que usted describiera a la sinfonía de Beethoven como una variación de la presión de la onda auditiva".*

**Albert Einstein**

# Lenguaje de programación Wolfram

Por: José Luis López Martínez

El lenguaje de programación Wolfram es un lenguaje basado en el conocimiento, lo anterior significa que es un lenguaje con mucho “conocimiento”, por lo que al utilizarlo el programador puede hacer las cosas que desea realizar utilizando menos líneas de código. En palabras de su creador Stephen Wolfram, “el lenguaje es como una herramienta optimizada para convertir las ideas en realidad. Inicias con una idea para realizar algún proyecto y posteriormente lo expresas utilizando el lenguaje Wolfram. Entonces, el lenguaje Wolfram lo realiza de forma automática como sea posible”[1]. Permite crear aplicaciones interactivas, para web o para dispositivos como los Raspberry Pi ó Arduino.

Algunas personas confunden la aplicación Mathematica con el lenguaje Wolfram; sin embargo, el lenguaje es más extenso y versátil, y funciona en otras aplicaciones como son SystemModeler, Data Science Platform , etc. Por ejemplo, el sistema WolframAlpha (<http://www.wolframalpha.com>) fue escrito en el lenguaje Wolfram.

El lenguaje de programación Wolfram hace énfasis en la programación funcional y simbólica, lo cual lo convierte en un lenguaje multiparadigma.

Un ejemplo de código en Wolfram, sería:

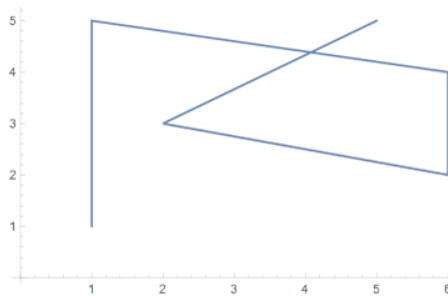
```
In=ListLinePlot[{{1,1},{1,5},{6,4},{6,2},{2,3},{5,5}}]
```



*"Las matemáticas son uno de los descubrimientos de la humanidad. Por lo tanto no pueden ser más complicadas de lo que los hombres son capaces de comprender".*

**Richard Phillips Feynman**

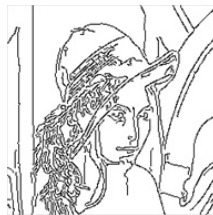
El cual utiliza una función predefinida "ListLinePlot" que dibujará líneas a partir de una lista de coordenadas que recibe como argumento, la salida sería:



El lenguaje puede trabajar con funciones anidadas; por ejemplo, se utilizan las siguientes funciones dando como argumento una imagen, primero detectará los bordes y posteriormente, realizará el negativo de los bordes detectados

In= ColorNegate[ EdgeDetect [  ] ]

por lo que el resultado será



De igual forma, es posible que el lenguaje realice operaciones con computación simbólica como:

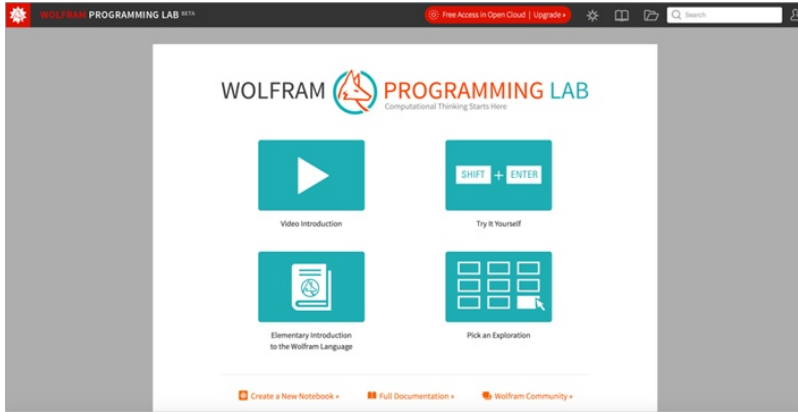
Solve[{2 x + 3 y, 3 x - y} == {1, 0}], el cual resuelve un sistema lineal de ecuaciones de dos incógnitas proporcionando el siguiente resultado {{x -> 1/11, y -> 3/11}}.



*"No hay certidumbre allí donde no es posible aplicar ninguna de las ciencias matemáticas ni ninguna de las basadas en las matemáticas".*

**Karl Weierstrass**

Como se puede observar, el lenguaje Wolfram es muy versátil y es posible aprovechar las miles de funciones programadas con las que cuenta. Una desventaja sería que no es gratuito; sin embargo, recientemente el sitio web de wolfram ([www.wolfram.com](http://www.wolfram.com)) ha lanzado una iniciativa denominada “Wolfram programming lab” (<http://www.wolfram.com/programming-lab>) mediante la cual es posible iniciar en el aprendizaje del lenguaje sin ningún costo. En la figura 1, se presenta la pantalla web inicial del sitio para aprender el lenguaje mencionado.



**Fig 1.** Sitio web para aprender a programar en el lenguaje Wolfram.  
Imagen tomada de [2].

Es importante comentar que existe una galería extensa (<http://www.wolfram.com/language/gallery/>) donde se encuentran infinidad de proyectos de todas las áreas; por ejemplo, en la figura 2, se muestra un simulador de un brazo robótico junto con las ecuaciones matemáticas de rotación y traslación necesarias para llevar a cabo los movimientos.



*"Las matemáticas son una gimnasia del espíritu y una preparación para la filosofía".*

**Isócrates**



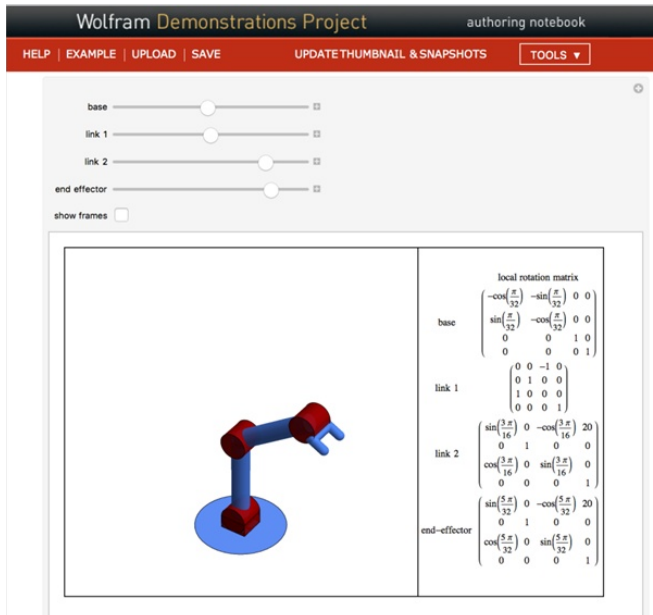


Fig 2. Ejemplo de un proyecto de un simulador de brazo robótico realizado en el lenguaje Wolfram. Imagen tomada de [2].

Para finalizar, te invitamos a revisar las páginas web sugeridas y conocer acerca del lenguaje Wolfram, tal vez sea el lenguaje que necesites para tu próximo proyecto escolar.

**Referencias:**

- [1] Wolfram, S. 2015. An Elementary Introduction to the Wolfram Language. Wolfram Media
- [2] [www.wolfram.com](http://www.wolfram.com). Consultado el 3 de noviembre de 2015



# Aprendizaje por refuerzo

Por: Michel García García, Cinhtia González Segura, Sergio González Segura

Al pensar en aprendizaje, una de las primeras cosas que nos vienen a la mente es la relación que los seres vivos tenemos con nuestro ambiente con el cual interactuamos. Gracias a ésta interacción obtenemos habilidades por medio de la evaluación de nuestras acciones en base a prueba y error. Como ejemplo podemos mencionar el proceso de aprendizaje que los seres humanos tenemos al comenzar a caminar, que por medio de múltiples ensayos de prueba y error (levantadas y caídas) durante el proceso de aprendizaje, nos es posible obtener la mejor forma de levantarnos sin caernos. En el ejemplo anterior, el aprendizaje se adquiere sin necesidad de tener un instructor (alguien que nos indique qué acciones son buenas y cuales son malas), pero si tenemos una interacción directa con el ambiente, al utilizar esta conexión obtenemos información sobre la acción-reacción, las consecuencias de las acciones realizadas, y sobre que se debe hacer para lograr las metas que se desean. A través del tiempo estas interacciones con el ambiente son nuestra mayor fuente de información y conocimiento sobre la relación aprendiz-ambiente. El proceso de aprendizaje por interacción es, sin duda una de las ideas fundamentales de la mayoría de las teorías de aprendizaje e inteligencia. Por tal motivo, se definen las técnicas de aprendizaje por refuerzo como aquellas que a través de prueba y error permiten definir un comportamiento óptimo [Mariano, 2001].

Sutton y Barto(1998) han formulado el aprendizaje por refuerzo como una estrategia de aprendizaje en la cual no se necesita un conjunto de ejemplos evaluados previamente, es decir, el aprendizaje es realizado sin conocimiento previo de las acciones que deben ser ejecutadas para llegar al comportamiento óptimo esperado. Por ejemplo, en un programa para jugar partidas de ajedrez que utiliza aprendizaje por refuerzo, la ubicación de las piezas del juego después de cada movimiento conduce a una nueva situación que corresponde a un estado. El estado anterior y el actual son evaluados y a partir del resultado obtenido se otorga un refuerzo positivo o negativo dependiendo si hubo o no mejora en la situación. De esta manera el programa por si solo aprende a realizar los mejores movimientos.

En este tipo de aprendizaje no se conoce la salida exacta deseada para cada entrada, aunque si se conoce como debería ser el comportamiento general ante diferentes entradas. Es un aprendizaje en el que se tiene cierta información tipo Entrada-Salida a través de un proceso de retroalimentación con mensajes de éxito o fracaso, correspondientes a una señal de refuerzo que mide el buen funcionamiento del sistema.

Cuando cierta acción realizada por el sistema de aprendizaje es seguida por un estado satisfactorio, entonces la tendencia del sistema a producir esa acción particular es reforzada positivamente. De lo contrario, la tendencia del sistema a producir dicha acción es disminuida o reforzada negativamente.

La característica más importante que distingue los métodos de aprendizaje por refuerzo de otros tipos de aprendizaje es que éstos usan una información de entrenamiento para evaluar las acciones tomadas, en lugar de recibir instrucciones en forma de "la acción correcta es". Esto crea la necesidad de exploración, de una búsqueda por ensayo y error del comportamiento correcto. La retroalimentación recibida, dice que tan buena es una acción, pero no si es la mejor o la peor posible (a diferencia de la retroalimentación por instrucción o supervisada)[Guerra, 2004].

### El problema de aprendizaje por refuerzo

El aprendizaje por refuerzo puede verse como el problema de aprender a cumplir ciertas metas, a partir de la interacción hasta lograr una meta. Tanto al aprendiz como al ejecutor de las decisiones se les conoce como agentes. Todo lo que este fuera del agente en la interacción, se conoce como ambiente. Las acciones del agente modifican el ambiente, por lo que éste presenta nuevas situaciones al agente. El ambiente también da lugar a la recompensa que el agente tratará de maximizar con el tiempo. El agente y el ambiente interactúan en cada paso de una secuencia discreta de tiempo  $t=0, 1, 2, \dots$ . En cada paso de tiempo  $t$ , el agente recibe una representación del estado del ambiente (situación)  $S_t \in S_t$  y con base en esto selecciona una acción,  $a_t \in A(s_t)$ . Un intervalo de tiempo después, y como consecuencia de la acción seleccionada, el agente recibe una recompensa numérica  $r_{t+1} \in \mathbb{R}$ , y se traslada al próximo estado  $S_{t+1}$ . Lo anterior se muestra en la Figura 1.



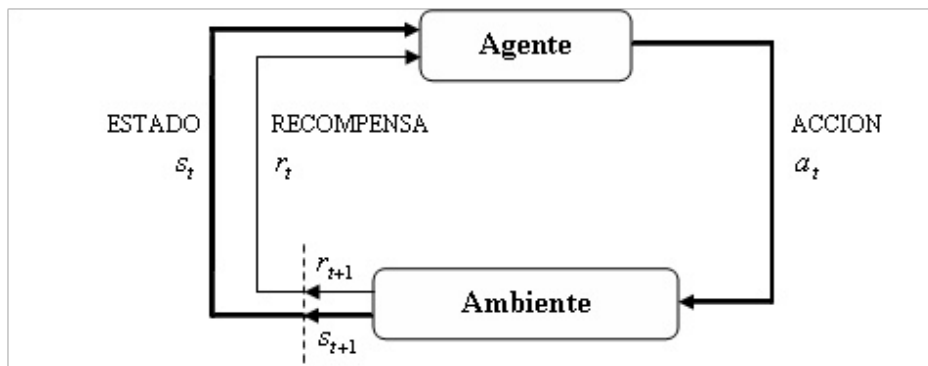


Fig 1. Interacción agente-ambiente en aprendizaje por refuerzo

En cada paso de tiempo, el agente hace un mapeo de los estados a las probabilidades de seleccionar cada posible acción. Éste mapeo se define como la política del agente  $\pi_t$ , donde  $\pi_t(s, a)$  es la probabilidad de que  $a_t = a$  dado que  $s_t = s$ . Ésta política determina el comportamiento del agente induciéndolo a seleccionar acciones que le permitan incrementar la suma de los valores de refuerzo a largo plazo. El agente aprende la política óptima aplicando prueba y error.

Formalmente, el modelo de aprendizaje por refuerzo consiste de:

- Un conjunto discreto de estados, **S**.
- Un conjunto discreto de acciones, **A**.
- Un conjunto de valores de recompensas, **R**.

Una forma intuitiva de entender la relación entre agente y ambiente es como el siguiente diálogo de ejemplo.





## Referencias:

- [Mariano, 2001] **Carlos E. Mariano**, *Aprendizaje por refuerzo en optimización multiobjetivo*, Tesis doctorado, ITESM campus Cuernavaca, 2001.
- [Sutton y Barto, 1998] **R. Sutton y G. Barto**, *Reinforcement Learning: An Introduction*, MIT Press Cambridge, MA, 1998.
- [Guerra, 2004] **Guerra, H. A.** *Aprendizaje Automático: Aprendizaje por reforzamiento*, Universidad Veracruzana, Facultad de Física e Inteligencia Artificial, 2004
- [Pack y Littman, 1996] **Leslie Pack Kaelbling, Michael L. Littman, Andrew W. Moore**, *Reinforcement Learning: A Survey*, *Journal of Artificial Intelligence Research*, 4, 237-285, 1996



# Aplicaciones de salud en dispositivos móviles

Por: Maximiliano Canché Euán y Erika Llanes Castro.

Utilizamos el término *mHealth* para referirnos a la salud móvil, que se define como informática para dispositivos móviles, sensores médicos y tecnologías de la comunicación que se utilizan para promover la salud, e incluyen el control de enfermedades crónicas y el bienestar. *mHealth* incluye aplicaciones médicas que podrían funcionar en un teléfono móvil, sensores que hacen un seguimiento de las constantes vitales y las actividades relacionadas con la salud y sistemas informáticos basados en la nube para la recopilación de datos sanitarios.

Se ha producido una explosión en la *mHealth* a lo largo de los últimos cinco años, con la aparición de más de 13.000 apps de temática sanitaria tan sólo para usuarios del Iphone de Apple y más de 6.000 apps médicas disponibles para usuarios de Android. Por ejemplo las apps que se centran en la diabetes están proliferando en el mercado, pero ¿cuántas apps y de qué tipo hay disponibles? Existen un sinnúmero de apps; entre ellas podemos distinguir algunas clasificaciones listadas a continuación:

## 1. Apps de Control médico.

Por ejemplo *Welldoc Diabetes Manager* (“Bluestar”) es la única app que ha obtenido la autorización de la FDA (*Food and Drug Administration*) en Estados Unidos de América para su uso en el control médico de la diabetes tipo 2 en adultos. El sistema Welldoc permite a los pacientes hacer un seguimiento y registrar sus niveles de glucemia e identifica tendencias de los patrones de glucemia, ofreciendo una respuesta en tiempo real de base clínica y sirviendo de ‘profesor’ a las personas que viven con diabetes. Además, esta app puede compartir los datos sobre diabetes directamente con el equipo sanitario. *Bluestar* se puede obtener tan sólo mediante prescripción médica..



## *2. Apps de seguimiento que muestran información sobre la salud*

El mayor porcentaje de apps se centra en el seguimiento de la salud. Estas apps permiten al usuario llevar un seguimiento de la glucemia, las dosis de insulina, carbohidratos, peso y actividad física y revisar sus datos en modos distintos, como cifras sin procesar, gráficos o resumen de valores, como medias. La mayoría de las apps de seguimientos exigen que el usuario introduzca manualmente sus datos sanitarios en la app. Tan sólo unas pocas app pueden subir directamente los datos sobre niveles de glucosa a un teléfono móvil, como el sistema Glooko, el glucómetro IBTStar o el glucómetro Telcare. MySugr es una app de uso diario y monitorización que aprovecha un estilo lúdico para mantener a los usuarios enganchados y motivados. Todas estas apps han obtenido la aprobación de la FDA.

## *3. Apps para enseñar y/o formar*

Aproximadamente el 22% de las apps se centran en enseñar y/o formar. Por ejemplo, algunas apps enseñan los principios del recuento de carbohidratos mediante gráficos y juegos interactivos. Otras apps son calculadoras de dosis de insulina que sugieren la dosis de esta hormona en base a un valor diana de glucemia, el factor de corrección, la proporción de carbohidratos, el nivel presente de glucemia y el cálculo de carbohidratos antes de una comida concreta. Las apps de seguimiento también ofrecían formación a los usuarios acerca de la administración de medicamentos, como el glucagón o ayuda en el uso de dispositivos.

## *4. Apps dirigidas a médicos*

Aunque la mayoría de las apps han sido desarrolladas para personas con alguna enfermedad, aproximadamente un 8% están pensadas para el proveedor sanitario como herramienta para ofrecer información médica. Otras apps han sido diseñadas para las revistas de salud, ofrecen acceso electrónico a artículos.







Actualmente, la mayoría con un costo para el usuario de unos 13 pesos, en promedio. Se espera que el número de aplicaciones y de usuarios crezca exponencialmente en los próximos dos años. De acuerdo con BernatGuitart, informático y CEO de AppFutura.com, los pacientes estarán dispuestos a pagar por apps “útiles y efectivas” en beneficio de su salud. De hecho, afirmaron que “actualmente hay más de 17,000 apps de salud disponibles, la mayoría con un costo para el usuario de unos 13 o 26 pesos en promedio”. Finalmente calcula que en 2 años habrá más de 500 millones de personas utilizando apps médicas desde el smartphone o tableta.

### Referencias:

- Estrin D, Sim I. Open mHealth architecture: an engine for health care innovation. *Scienc* 2010; 330: 759.
- Milosevi M, Shrove MT, Jovanov E. Applications of smartphones for ubiquitous health monitoring and wellbeing management. *JITA* 2011; 1: 7-15.
- Federal Trade Commission. FTC Staff Report Recommends Ways to Improve Mobile Privacy Disclosures. USA, 2013.
- <http://www.pcworld.com.mx/Articulos/28040.htm>
- <http://www.forbes.com.mx/apps-medicas-otro-bisturi-de-los-cirujanos/>



# Colabora con Nosotros

¿Deseas publicar avisos clasificados o enviar tus comentarios y sugerencias? puedes hacerlo a:

**[michel.garcia@uady.mx](mailto:michel.garcia@uady.mx)**

¿Deseas colaborar y/o participar en alguna sección? no dudes en contactarnos y acércate a tus profesores.

**Equipo responsable del octavo número:**



