Computación Blanda

Soft Computing

Autor: Luz Adriana Quitumbo Santa

*IS&C, Universidad Tecnológica de Pereira, Pereira, Colombia*

Correo-e: adriana.quitumbo@utp.edu.co

*Resumen*— Este documento presenta un resumen de las líneas clásicas de la Computación Blanda: redes neuronales, lógica difusa, sistemas expertos, algoritmos genéticos y machine learning. El objetivo del documento es brindar una panorámica general de las temáticas, mostrando su relación con las técnicas de inteligencia artificial. La diferencia entre el paradigma de Inteligencia Artificial y la computación blanda está centrada en el mecanismo de inferencia utilizado y su aplicación a la solución de problemas tomados de lo cotidiano, de las teorías de conocimiento y de su relación con ciencias afines.

***Palabras clave—* sistemas, redes, inteligencia artificial, software, computación, investigación, industria, genético, aprendizaje.**

*Abstract*— This document presents a summary of the classic lines of Soft Computing: neural networks, fuzzy logic, expert systems, genetic algorithms and machine learning. The objective of the document is to provide a general overview of the topics, showing their relationship with artificial intelligence techniques. The difference between the Artificial Intelligence paradigm and soft computing is centered on the inference mechanism used and its application to the solution of problems taken from everyday life, from knowledge theories and their relationship with related sciences.

*Key Word*— systems, networks, artificial intelligence, software, computing, research, industry, genetic, learning.

### INTRODUCCIÓN

La temática de la Computación Blanda se encuentra enmarcada en el paradigma de la Inteligencia Artificial. La diferencia con dicho paradigma radica en que la Computación Blanda está centrada en la aplicación pragmática de las teorías de la Inteligencia Artificial a la solución de problemas complejos en diversos campos del conocimiento.

Las líneas derivadas de la Computación Blanda, se configuran en las siguientes tendencias: a) Redes Neuronales Artificiales, b) Lógica Difusa, c) Sistemas Expertos, d) Algoritmos Genéticos, e) Deep Learning (Machine Learning).

En los siguientes apartados se presenta un resumen de dichas tendencias.

* 1. **REDES NEURONALES**

Aproximadamente en el año 1936 con Alan Turing ya se empezaba a formar la idea de tomar como referencia el cerebro humano, para la resolución de problemas complejos y desarrollarlos por medio del mundo de la computación. Con los aportes del neurofisiólogo Warren McCulloch y del matemático Walter Pitts. Quienes, en el año de 1943 lanzaron una teoría acerca de la forma de trabajar de las neuronas. Además, modelaron una red neuronal simple mediante circuitos eléctricos. En el año 1974 Paul Werbos desarrollo la idea del algoritmo de aprendizaje de programación hacia atrás (backpropagation). Actualmente se publican numerosos trabajos y aplicaciones en el campo de las redes neuronales. Como se ha observado a través de la historia siempre se ha querido construir maquinas que faciliten nuestra vida, sino también se ha querido que estas máquinas tengan la capacidad de realizar los procesos que se le asignan con “cierta inteligencia humana”. Pero aún no se ha logrado cumplir con dicho objetivo. [1]

Hay muchas formas para definir las redes neuronales algunas de ellas son: consiste en un modelo matemático compuesto por un gran número de elementos procesales organizados en niveles. Otra definición más corta seria, las redes neuronales son una nueva forma de computación que está inspirada en modelos biológicos. Un sistema neuronal se compone de los siguientes elementos: conjunto de neuronas, patrón de conexiones (arquitectura), dinámicas de recuerdo y de aprendizaje y entorno.

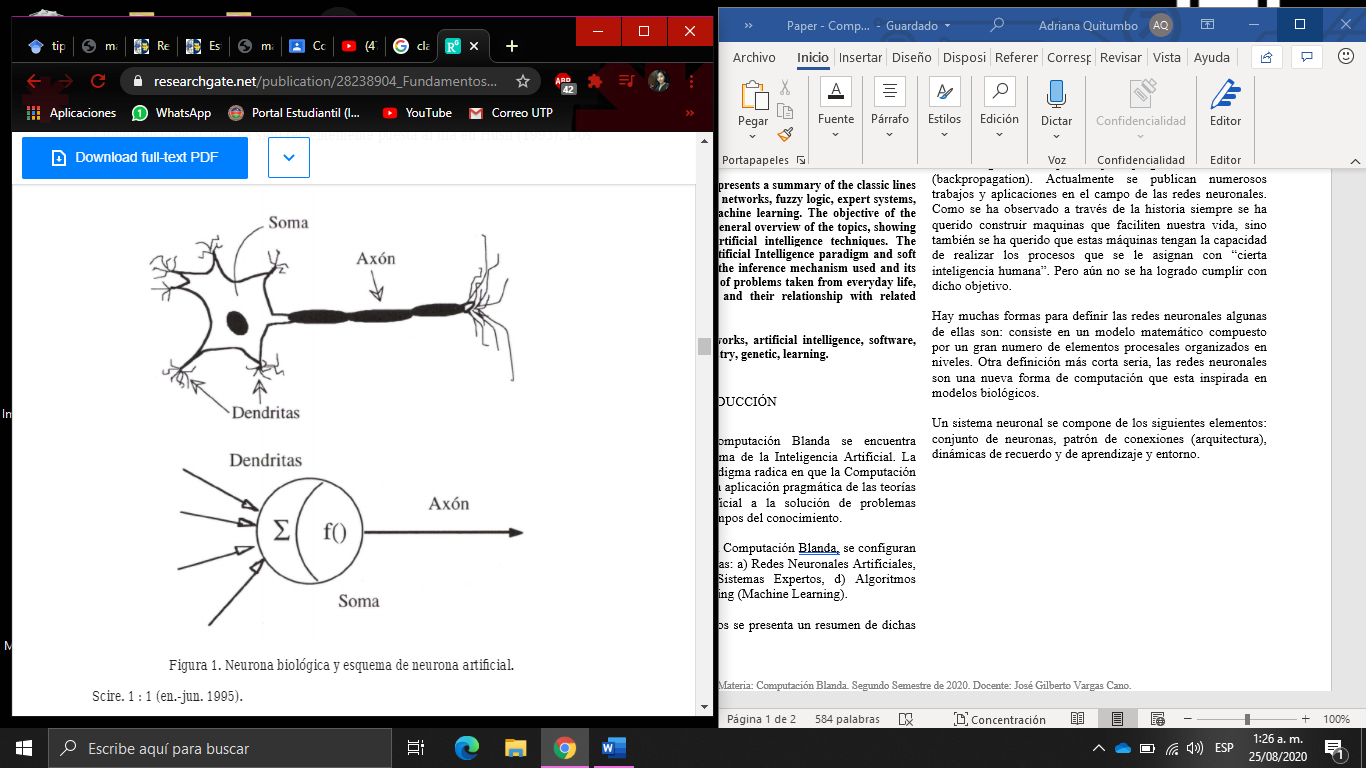
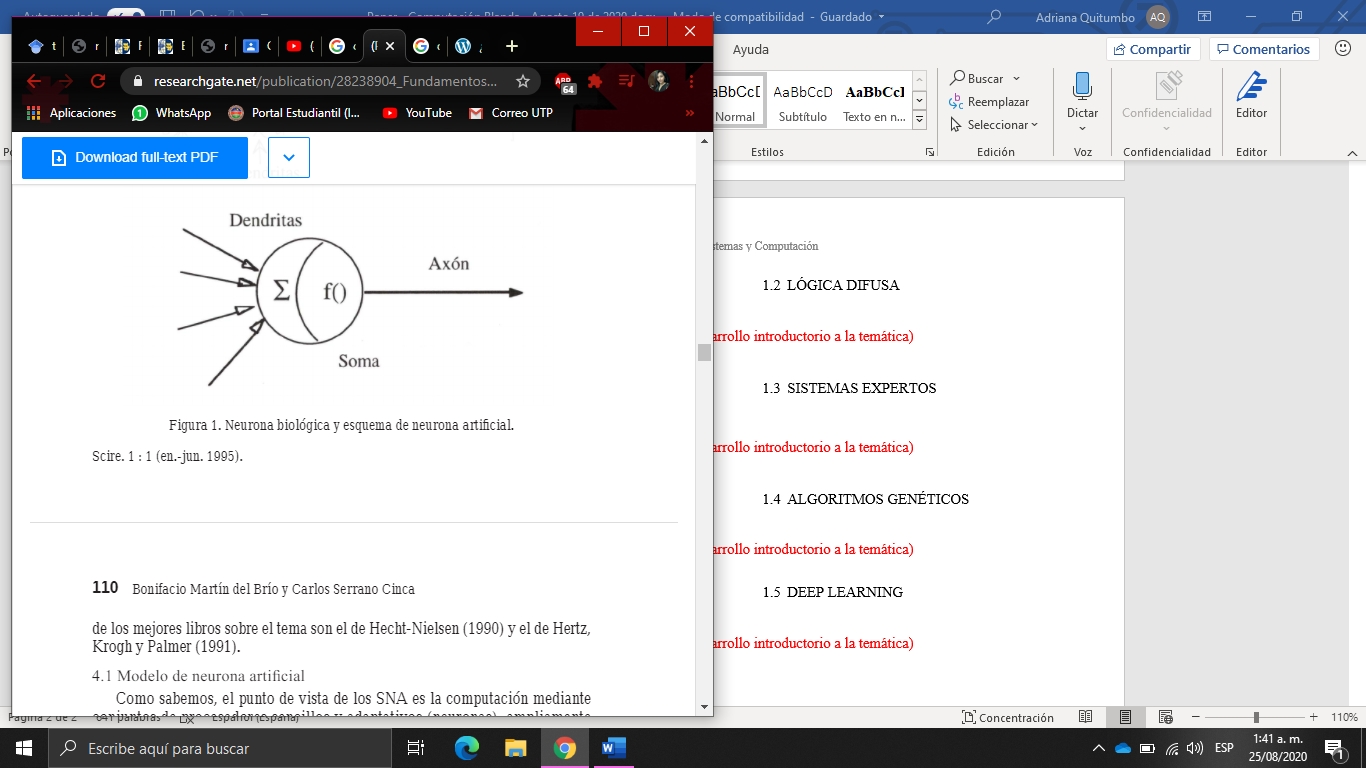


Figura 1. Neurona Biológica

Una neurona artificial se organiza en capas; varias capas constituyen una red neuronal y una o más redes neuronales, junto con los interfaces de entra-da/salida y otros módulos adicionales constituyen el sistema global. Forman parte de los denominados “Sistemas Inteligentes “, dentro de la rama de la Inteligencia Artificial.



Capa de entrada: es la capa que recibe directamente la información proveniente de las fuentes externas de la red.

Capa Oculta: son internas a la red y no tienen contacto directo con el entorno exterior. El número de niveles ocultos puede estar entre cero y un número elevado.

Capa de salida: transfieren información de la red hacia el exterior.[1]

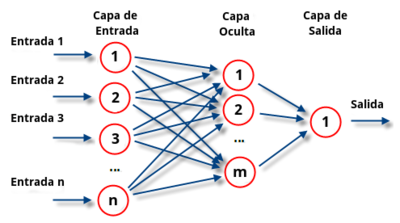


Figura 2. Neurona biológica.

**Tipos de redes neuronales artificiales**

Las redes neuronales artificiales se pueden clasificar ya sea por su arquitectura o con respecto al aprendizaje. A continuación, se explicará cada uno de ellos.

**Clasificación con respecto a su arquitectura:** Las neuronas se agrupan formando capas, que pueden tener diferentes características. Además, las capas se organizan para formar la estructura de la red. Para clasificar la arquitectura de las redes neuronales se tendrá en cuenta el número de capas las cuales pueden ser Redes Monocapa o Redes Multicapas.

**Redes neuronales de una sola capa:** Son aquellas que solo tienen una capa. Un ejemplo, es la red de tipo Hopfield, esta red tiene el mismo número de entras y salidas, se utiliza como memoria asociativa con unidades binarias; esta aprende un patrón, lo recuerda y además tiene la capacidad de restaurarlo.

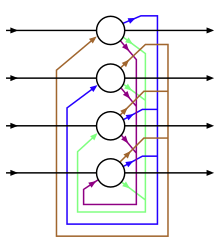


Figura 3. Red de Hopfield o de una sola capa.

**Redes neuronales multicapa:** como su nombre lo indica son varias capas de red neuronal, esta red está totalmente conectada, esto dado a que todas las salidas de la primera neurona son conectadas a todas las entradas de la siguiente capa y así sucesivamente. El procesamiento para la CPU de este tipo de red es muy grande y en ocasiones se hace inmanejable. En la figura 2 se puede observar un ejemplo de una red multicapa.

**Clasificación con respecto a su aprendizaje:** las redes neuronales manejan dos tipos de información las cuales son volátil y no volátil. Para que este tipo de redes aprendan se utilizan algoritmos que están basados en fórmulas matemáticas. Se pueden distinguir tres tipos de aprendizaje los cuales son:

**Aprendizaje no Supervisado**: este tipo de aprendizaje consiste en no presentar patrones objetivos, sino solo patrones de entrada, y dejar a la red clasificar dichos patrones en función de las características comunes de los patrones.

**Aprendizaje Supervisado:** se basa en que la red dispone de los patrones de entrada y los patrones de salida que deseamos para esa entrada y en función de ellos se modifican los pesos de las sinapsis para ajustar la entrada a esa salida.

**Aprendizaje reforzado:** este tipo de aprendizaje usa una formula híbrida, el supervisor no enseña patrones objetivos si no que solo le dice si acierta o falla en su respuesta ante un patrón de entrada.[2]

* 1. **LÓGICA DIFUSA**

El termino de lógica difusa fue concebido por el profesor de la Universidad de California, Lotfi A. Zadeh; en el año1965. Surgió como una herramienta importante para controlar sistemas y procesos industriales complejos. De una manera simple y elegante para obtener una conclusión a partir de información vaga, imprecisa o que este incompleta. Lo que hace la lógica difusa es tomar una decisión con la información que se ha recolectado sin importar que este incompleta y las demás características que se mencionaron anteriormente.

La lógica difusa se puede aplicar en procesos demasiados complejos y aún más cuando no existe un modelo de solución simple o un modelo matemático preciso. También es muy útil también para aplicarlo en el conocimiento de un experto que utiliza conceptos ambiguos o imprecisos. No es recomendable utilizar la lógica difusa cuando algún modelo matemático ya soluciona eficientemente el problema, cuando los problemas son lineales o cuando no tienen solución [3].

**Conjuntos difusos:** Un conjunto difuso se encuentra asociado por un valor lingüístico que está definido por una palabra, etiqueta lingüística o adjetivo. En los conjuntos difusos la función de pertenencia puede tomar valores del intervalo entre 0 y 1, y la transición del valor entre cero y uno es gradual y no cambia de manera instantánea como pasa con los conjuntos clásicos [3].

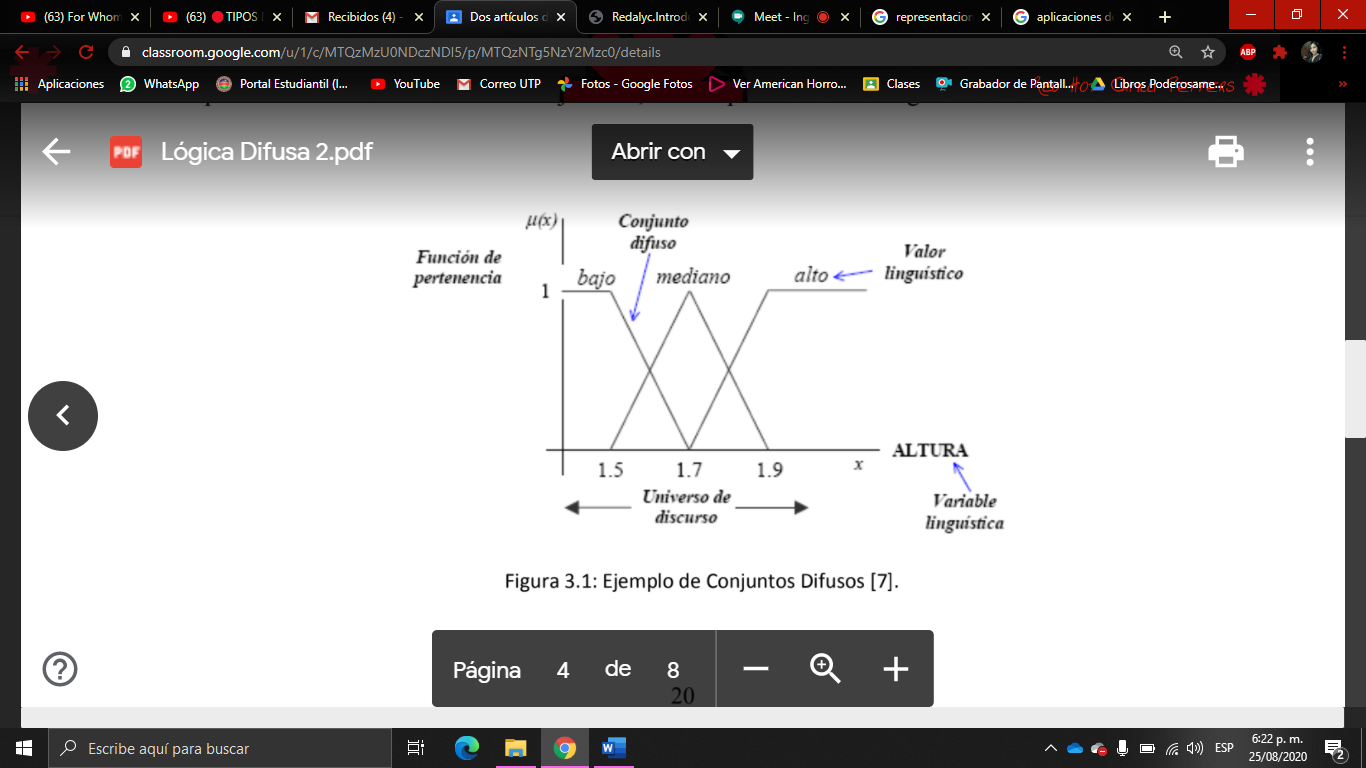
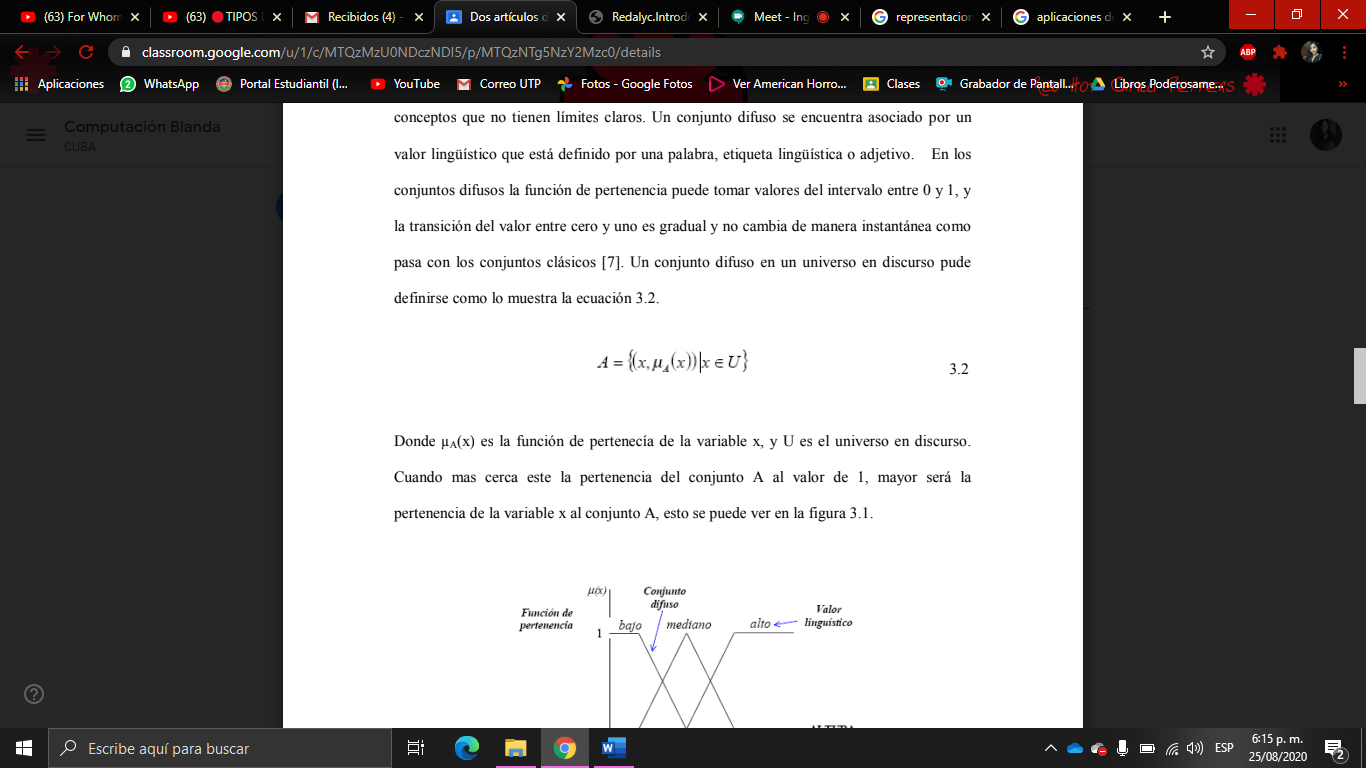


Figura 4 Ejemplo de conjuntos difusos.

La lógica difusa, se basa en reglas de la forma SI (antecedente) ENTONCES (consecuente). Los métodos de inferencia para las reglas deben ser sencillos y eficientes. Para escoger una salida concreta a partir de las hipótesis, el método más usado es el ‘centroide’(defusificación) en el que la salida final será el centro de gravedad del área resultante. Si las reglas no son formuladas por expertos, son aprendidas por

el sistema, éste hace uso de redes neuronales. Los tipos más populares de sistemas de inferencia difusa son: sistemas difusos de tipo Mamdani y sistemas difusos de tipo Sugeno.[4]

**Sistemas de tipo Mamdani.** Este modelo fue propuesto por Ebrahim Mamdani en 1974; las reglas están dadas por expresiones lingüística modelo tipo Mamdami.

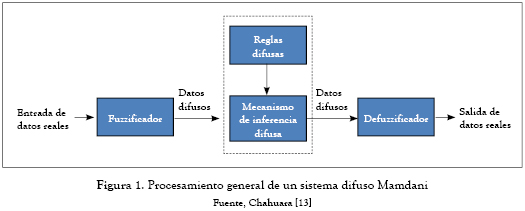


Figura 5 procesamiento de un sistema de Mamdani.

**Aplicaciones de la lógica difusa:** como se ha mencionado la lógica difusa se puede utilizar en procesos muy complejos, cuando no hay un modelo matemático simple a continuación se mencionará en que ámbitos se puede aplicar.

* Reconocimiento de patrones y Visión por ordenador: Seguimiento de objetos con cámara, reconocimiento de escritura manuscrita, reconocimiento de objetos, compensación de vibraciones en la cámara, sistemas de enfoque automático.
* Sistemas de información o conocimiento: Bases de datos, sistemas expertos.
* Control de sistemas: Control de tráfico, control de vehículos, control de compuertas en plantas hidroeléctricas, centrales térmicas, control en máquinas lavadoras, control de metros (mejora de su conducción, precisión en las paradas y ahorro de energía), ascensores, Etc.
  1. **SISTEMAS EXPERTOS**

El término de sistemas expertos fue utilizado por primera vez por un doctor Edward A. Feigenbaum. El estipuló que el poder de resolución de un problema en un programa de computadora viene del conocimiento de un dominio específico, no solo de las técnicas de programación y el formalismo que contiene. En el desarrollo de los sistemas expertos también se ha producido a medida que se ha ido desarrollando la Inteligencia artificial y los diferentes métodos que se han empleado para su resolución. El primer sistema experto aparece en el año 1965 y fue terminado en el año 1975, el cual se le dio el nombre de DENDRAL; estar era tomar unas hipótesis relevantes como soluciones posibles, y someterlas a prueba comparándolas con los datos. En 1965 se empezaron a utilizar técnicas de búsqueda heurística para resolución de problemas. En el año 1972, en la Universidad de Standford desarrollo MYCIN, sistema experto dentro del campo de la medicina para diagnóstico de enfermedades infecciosas en la sangre. Otro hecho importante para los sistemas expertos fue la creación del lenguaje PROLOG basado en las teorías de Minsky. En 1979 aparece XCON, primer programa que sale del laboratorio Su usuario fue la Digital Equipament Corporation (DEC). En 1987 aparecieron los microordenadores Apple y compatibles IBM con una potencia parecida a los LISP. El software se transfirió a máquinas convencionales utilizando el lenguaje "C" lo que acabó con el LISP. A partir de los 1990 y con el desarrollo de la Informática, se produce un amplio desarrollo en el campo de la Inteligencia artificial y los Sistemas Expertos [5].

Un sistema experto es una rama de la Inteligencia artificial; son maquinas que piensan y razonan como un experto lo haría en una cierta especialidad o campo. Simulando el proceso de aprendizaje, de memorización, de razonamiento, de comunicación y de acción en consecuencia. Por ejemplo, un sistema experto en diagnostico medico requeriría como datos los síntomas del paciente, los resultados de análisis clínicos y otros hechos relevantes y utilizando estos, buscaría en una base de datos la información necesaria para poder identificar la correspondiente enfermedad [6]. Para que saber que un sistema es experto debe contar con las siguientes características: la primera es tener Habilidad para adquirir conocimiento. La segunda es Fiabilidad, para poder confiar en sus resultados o apreciaciones. También debe caracterizarse por ser Solidez en el dominio de su conocimiento. Y por último tener la capacidad para resolver problemas [5].

**Componentes de un sistema experto:**

* Separan conocimientos (reglas y hechos) y el procesamiento; se le añade una interfase de usuario y un componente explicativo; los siguientes componentes pueden estar estructurados de formas muy variadas.
* Base de conocimientos: Contiene el conocimiento de los hechos y las experiencias de los expertos en un dominio determinado.
* Mecanismo de inferencia: Puede simular la estrategia de solución de un experto**.**
* Componente explicativo: Explica al usuario la estrategia de solución encontrada y el porqué de las decisiones tomadas
* Interfase de usuario: Sirve para que este pueda realizar una consulta en un lenguaje lo más natural posible
* Componente de adquisición: Ofrece ayuda a la estructuración e implementación del conocimiento en la base de conocimientos [5].

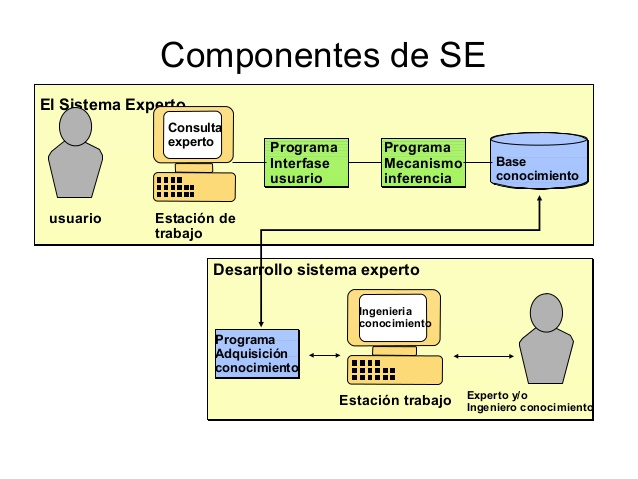


Figura 6 Componentes de un sistema experto.

**Tipos de sistemas expertos**: Existen tres tipos de sistemas expertos los cuales son:

* **Basados en reglas:** Aplicando reglas heurísticas apoyadas generalmente en lógica difusa para su evaluación y aplicación.
* **Basados en casos CBR (Case Based Reasoning):** Aplicando el razonamiento basado en casos, donde la solución a un problema similar planteado con anterioridad se adapta al nuevo problema.
* **Basados en redes bayesianas.:** Aplicando redes bayesianas, basadas en estadística y el teorema de Bayes [5].

**Ventajas de los sistemas expertos:** Se mencionará algunas ventajas que se obtiene al utilizar este tipo de sistemas.

* **Producción y productividad mayores**: Pueden trabajar más rápido que lo humanos.
* **Disponibilidad:** Están disponibles ininterrumpidamente de día y noche, ofreciendo siempre su máximo desempeño. Pueden duplicarse ilimitadamente.
* **Mayor calidad:** Dan la probabilidad de aumentar la calidad proporcionando asesoría consistente y reduciendo las tasas de error.
* **Operación en entornos peligrosos**: Muchas tareas requieren que los seres humanos operen en entornos hostiles y peligrosos.
* **Confiabilidad:** Los sistemas expertos son confiables. No padece de olvido, fatiga, dolor o comete errores de cálculo [5].

**Desventajas de los sistemas expertos:**

* Tienen una noción muy limitada acerca del contexto de problema, no pueden percibir todas las cosas que un experto humano puede apreciar de una situación.
* Pueden existir decisiones que sólo son de competencia para un ser humano y no una máquina.
* No saben cómo subsanar sus limitaciones, no son capaces de trabajar en equipo o investigar algo nuevo.
* Son muy costosos de desarrollar y mantener.
* La monitorización es un caso particular de la interpretación, y consiste en la comparación continua de los valores de las señales o datos de entrada y unos valores que actúan como criterios de normalidad o estándares [5].
  1. **ALGORITMOS GENÉTICOS**

El que inició los algoritmos genéticos fue John Henry Holland en el año 1975. Un algoritmo genético es un método de búsqueda que imita la teoría de la evolución biológica de Darwin para la resolución de problemas. Para ello, se parte de una población inicial de la cual se seleccionan los individuos más capacitados para luego reproducirlos y mutarlos para finalmente obtener la siguiente generación de individuos que estarán más adaptados que la anterior generación [7].

**Metodología basada en teoría de la evolución de Darwin:** Los algoritmos genéticos tienen sus antecedentes en la biología y comienzan con Charles Darwin, que con su libro El origen de las especies por medio de la selección natural o la preservación de las razas favorecidas en su lucha por la vida, nos habla sobre los principios de la selección natural. Los principios básicos de los algoritmos genéticos se derivan de las Leyes de la Vida Natural descritos por Darwin [8].

* Población inicial.
* Evaluación de la capacidad de adaptación.
* Selección y cruce.
* Mutación.
* Reorganización de la población con las nuevas generaciones.

**Metodología anterior añadiendo las teorías de Lamarck:**

* Población inicial.
* Adaptación Lamarckiana.
* Evaluación de la adaptabilidad.
* Selección y cruce.
* Mutación.
* Reorganización de la población con las nuevas generaciones.

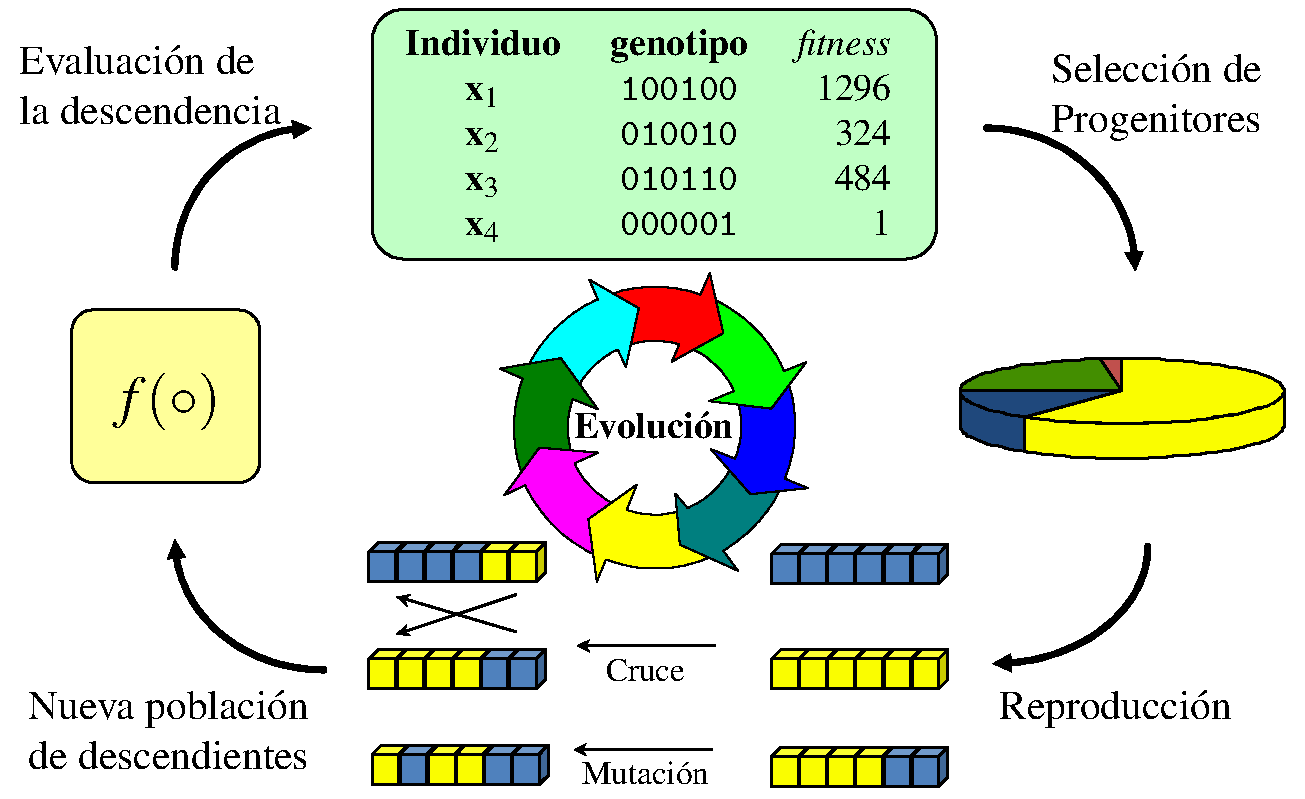


Figura 7 Esquema de un algoritmo genético.

**Algoritmo:**

Desarrollado por John H. Holland, el algoritmo genético opera entonces a nivel de genotipo de las soluciones mediante la siguiente secuencia:

1. Comenzar con una población inicial, la cual puede ser generada de manera aleatoria.

2. Calcular el fitness (aptitud) de cada individuo.

3. Aplicar el operador de selección con base en el fitness de la población.

4. Aplicar los operadores genéticos de reproducción, cruce y mutación a la población actual para generar a la población de la siguiente generación.

5. Ir al paso 2 hasta que la condición de parada se satisfaga.

6. Cuando se cumple la condición de parada, se devuelve al mejor individuo encontrado (bien el mejor de todas las generaciones, bien el mejor de la última generación).

Al igual que en muchas otras heurísticas, el comportamiento del algoritmo genético es altamente dependiente de los parámetros iniciales (tamaño de la población, porcentaje de cruce, porcentaje de mutación, número de generaciones, etc.), por lo que será necesario ajustar esos parámetros para tratar de mejorar la solución para los objetivos del problema [9].

**Métodos de algoritmos genéticos:** antes de que un algoritmo genético pueda ponerse a trabajar en un problema, se necesita un método para codificar las soluciones potenciales del problema de forma que el propio algoritmo pueda procesarlas aplicando las operaciones que le permiten evolucionar. Un enfoque común es codificar las soluciones como cadenas binarias: secuencias de 1's y 0's, donde el dígito de cada posición representa el valor de algún aspecto de la solución [10].

**Métodos de selección:**

* **Selección elitista:** se garantiza la selección de los miembros más aptos de cada generación. (La mayoría de los algoritmos no utilizan elitismo puro, sino una forma modificada por la que el individuo mejor, o algunos de los mejores, son copiados hacia la siguiente generación en caso de que no surja nada mejor).
* **Selección proporcional a la aptitud:** los individuos más aptos tienen más probabilidad de ser seleccionados, pero no la certeza.
* **Selección por torneo:** se eligen subgrupos de individuos de la población, y los miembros de cada subgrupo compiten entre ellos. Sólo se elige a un individuo de cada subgrupo para la reproducción.
* **Selección generacional**: la descendencia de los individuos seleccionados en cada generación se convierte en toda la siguiente generación. No se conservan individuos entre las generaciones.
  1. **DEEP LEARNING**

Los inicios del deep learning comenzó con los estudios sobre el aprendizaje biológico de McCulloch&Pitts (1943) y Hebb (1949). A su vez, dieron pie a la implementación de estos a través de primeros modelos, como por ejemplo el Perceptrón Rosenblatt (1958), que permitía el entrenamiento de una única neurona. Deep learning es un tipo de machine learning que entrena a una computadora para que realice tareas como las hacemos los seres humanos, como el reconocimiento del habla, la identificación de imágenes o hacer predicciones.

En lugar de organizar datos para que se ejecuten a través de ecuaciones predefinidas, el deep learning configura parámetros básicos acerca de los datos y entrena a la computadora para que aprenda por cuenta propia reconociendo patrones mediante el uso de muchas capas de procesamiento.

**Aplicaciones del deep learning**

Las empresas que aplican modelos de aprendizaje profundo pueden aprender elementos significativos sobre sus clientes para satisfacer de mejor forma sus necesidades o inclusive mejorar sus procesos internos. Veamos a continuación cuales son las aplicaciones más comunes donde tratamos con algoritmos de Deep Learning. Se utiliza para clasificar imágenes, reconocer el habla, detectar objetos y describir contenido. Sistemas como Siri y Cortana son potenciados, en parte, por el aprendizaje a fondo [11].

**Reconocimiento de imágenes:** El reconocimiento de imágenes es una de las aplicaciones más interesantes del Deep Learning ya que su grado de complejidad es superior. Las potencialidades de aplicación de este tipo de algoritmos permiten entender y comprender escenas e imágenes en su totalidad determinando un gran número de factores. Podemos analizar un escenario completo partiendo de miles de imágenes por solo citar un caso de aplicación.

**Reconocimiento del habla:** Gracias a los niveles de complejidad tan diversos que permite el aprendizaje profundo las tecnologías de reconocimiento de habla y patrones de voz han aumentado su presencia en el mundo académico y empresarial. Cada vez es más común la utilización de estos servicios en móviles y ordenadores de forma simple y esto es gracias a la capacidad de análisis de complejidad que brindan las redes neuronales [11].

### REFERENCIAS

Referencias en la Web:

[1]

<file:///C:/Users/adria/Downloads/matich-redesneuronales%20(1).pdf>

[2] <https://inteligenciartificiallmca.wordpress.com/2017/06/10/1-2-clasificacion-de-las-redes-neuronales/>

[3]

<https://www.redalyc.org/pdf/3821/382138367007.pdf>

[4]

<http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1692-33242012000200011>

[5]

<https://www.ecured.cu/Sistemas_expertos#:~:text=probabil%C3%ADsticas%20con%20posterioridad.-,Evoluci%C3%B3n%20hist%C3%B3rica,fue%20el%20primer%20sistema%20experto.&text=En%201972%2C%20en%20la%20Universidad,enfermedades%20infecciosas%20en%20la%20sangre.>

[6]

<http://computo.fismat.umich.mx/~htejeda/gutierjm/BookCGH.pdf>

[7]

<http://www.it.uc3m.es/jvillena/irc/practicas/06-07/05.pdf>

[8]

<https://planetachatbot.com/entendiendo-los-algoritmos-gen%C3%A9ticos-un-caso-de-uso-en-el-entorno-organizacional-a745c157fa8c>

[9]

<https://www.adictosaltrabajo.com/2010/10/07/jgap/>

[10]

<http://www.cs.us.es/~fsancho/?e=65>

[11]

<https://www.grapheverywhere.com/deep-learning/>