Ángel L. Martín de Francisco

Catedrático de Nefrología Universidad de Cantabria

INTELIGENCIA ARTIFICIAL EN MEDICINA: PASO A PASO

Conceptos claros y prácticos para comenzar

NEFROLOGÍA AL DÍA



Sociedad Española de Nefrología

Víctor Lorenzo Sellares
Juan Manuel López Gómez
Miquel Blasco Pelicano
Editores Nefrología al Día

INTELIGENCIA ARTIFICIAL EN MEDICINA PASO A PASO

Conceptos claros y prácticos para comenzar

Ángel Luis Martín de Francisco

Inteligencia Artificial en medicina paso a paso. Conceptos claros y prácticos para comenzar
Monografías de Nefrología al Día.
Grupo Editorial de la Sociedad Española de Nefrología. Edición 2025.

ISBN 978-84-129769-5-3

Editores

Víctor Lorenzo Sellares
Juan Manuel López Gómez
Miquel Blasco Pelicano

ÍNDICE

Presentación **1**

1.

Funcionamiento de la IA **3**

1.1 Lenguaje de Programación 4

1.2 ¿Por qué se sigue usando programación con IA? 4

1.3 Similitudes entre la Inteligencia Humana y la Inteligencia artificial 5

1.4 Fuentes de información de la IA 6

1.5 Programación: el punto de partida de la IA 6

2.

Subtipos de la IA **11**

2.1 Modelos de Aprendizaje Automático (Machine Learning ML) 11

2.1.1 Tipos de aprendizaje automático (ML) 13

2.2 Modelos de Aprendizaje Profundo (Deep Learning): Redes Neuronales Artificiales 16

2.2.1 Cómo funcionan las redes neuronales 17

2.2.1.1 Estructura y componentes básicos de una red neuronal 17

2.2.1.2 Proceso de entrenamiento y ajuste de pesos 17

2.2.1.3 Funciones de activación y su importancia 17

2.2.2 Ejemplos de funcionamiento de redes neuronales 17

2.2.3 Aplicaciones de Aprendizaje profundo (DL) 22

2.2.3.1 Vision artificial 22

2.2.3.2 Procesamiento del lenguaje natural (PLN) 25

2.2.3.3 Inteligencia artificial multimodal 31

2.3 Ética y peligros de la utilización de PLN en Medicina 31

2.4 Plataformas y webs especializadas en IA médica 32

PRESENTACIÓN

Adentrarse en el mundo de la inteligencia artificial puede parecer, para muchos, un desafío lleno de palabras extrañas, conceptos complejos y tecnologías que parecen inalcanzables. Sin embargo, detrás de esa primera impresión se esconde un universo apasionante y lleno de oportunidades. El objetivo de este libro es acompañarte desde los conceptos más básicos, paso a paso y de forma sencilla, para que descubras cómo funciona realmente la IA y cómo puedes empezar a utilizarla con confianza y curiosidad.

Se invita al lector a consultar el siguiente video en *Nefrología al Día*, que incluye una sencilla explicación del texto que sigue.

Inteligencia artificial desde cero. Una guía esencial en 50 minutos
Angel Luis Martin de Francisco Hernández
[https://www.nefrologiaaldia.org/es-inteligencia-artificial-desde-cero]



Ángel Luis Martín de Francisco
Catedrático de Nefrología. Universidad de Cantabria
Servicio de Nefrología. HU Valdecilla Santander Cantabria

La inteligencia artificial (IA) es un conjunto de tecnologías que permiten a las máquinas simular la inteligencia humana para realizar tareas como el reconocimiento de voz, la toma de decisiones, el procesamiento del lenguaje natural y la visión por computadora. Funciona a través de algoritmos y modelos de aprendizaje que procesan grandes cantidades de datos para

identificar patrones y hacer predicciones o tomar decisiones. Los sistemas de IA «aprenden» a través de numerosas repeticiones y el correspondiente feedback (entrenamiento) cómo alcanzar mejor una solución predeterminada (por ejemplo, diagnóstico, pronóstico de una enfermedad), lo que permite que el resultado sea más preciso y los procedimientos más eficientes [1][2].

REFERENCIAS

1 De Francisco A. L. M. Artificial. *La Nueva Inteligencia y su Aplicación en Nefrología*. En Lorenzo, V., López Gómez, J. M. (Eds.). Nefrología al Día. ISBN 978-84-127624-1-9.

2 De Francisco A. L. M. *Medicina Renal Inteligente. El futuro de la nefrología clínica con inteligencia artificial*. En Lorenzo, V., López Gómez, J. M. (Eds.). Nefrología al Día. ISBN 978-84-129769-3-9.

1.

FUNCIONAMIENTO DE LA IA

La IA funciona gracias a (*Figura 1*):

1. Herramientas informáticas: son los ordenadores y los programas que usan.

2. Fórmulas matemáticas: que ayudan a esos programas a aprender de los datos, como si buscaran patrones o repeticiones.

3. Y con estas fórmulas se crean Algoritmos, que son las instrucciones o pasos que se suceden secuencialmente para llevar a cabo una acción concreta. Supongamos que queremos ordenar una lista de nombres.

El algoritmo podría ser:

- Comparar el primer nombre con el segundo.
- Si el primero va antes que el segundo, dejarlos en ese orden.
- Si el segundo va antes que el primero, intercambiarlos.

- Repetir estos pasos para todos los nombres de la lista.

Veamos otros ejemplos prácticos de algoritmos:

- Reconocimiento facial: Un algoritmo analiza las características de una cara (ojos, nariz,

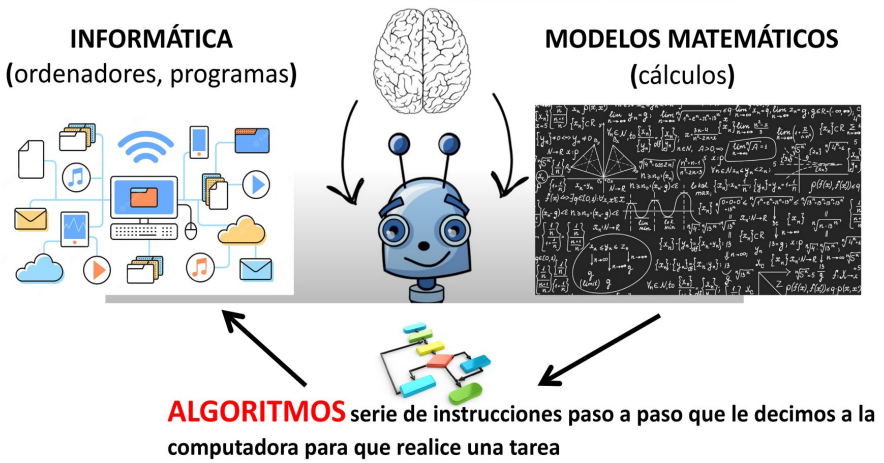


Figura 1. Funcionamiento de la Inteligencia Artificial

boca) y las compara con una base de datos para identificar a una persona.
- Recomendaciones en Netflix: Un algoritmo analiza tus películas favoritas y te sugiere otras que podrían gustarte, basándose en los gustos de otros usuarios con preferencias similares.
- Traducción automática: Un algoritmo analiza un texto en un idioma y lo traduce a otro, identificando patrones en las palabras y las frases.
- Coches autónomos: Un algoritmo procesa información de sensores (cámaras, radares) para tomar decisiones sobre la conducción, como acelerar, frenar o girar

1.1 LENGUAJE DE PROGRAMACIÓN

Es la herramienta para decirle a las computadoras qué hacer. Explicado de una manera sencilla es como un idioma especial que usamos para decirle a un ordenador lo que queremos que haga. En vez de hablarle como a una persona, usamos palabras y reglas que la máquina sí puede entender, para que siga nuestras instrucciones y realice tareas como mostrar una página web, sumar números o mover un personaje en un videojuego. Hay diferentes tipos según lo fácil que sean de usar y lo cerca que estén del lenguaje de las máquinas o de las personas. Los más usados son Python (el más utilizado) y Java Script.

Un ejemplo fácil:

- Usamos **input()** para pedirle al usuario que escriba algo.
- Usamos **print()** para mostrar mensajes o resultados en pantalla.

¿Cómo funciona un programa?

Un programa sigue una serie de pasos (llamados instrucciones o algoritmo) para hacer lo que queremos. Estos pasos pueden incluir operaciones matemáticas y decisiones, como:

- **if** (si pasa algo, hacer esto)
- **else** (si no, hacer otra cosa)
- **for** y **while** (repetir acciones varias veces)

Veamos un ejemplo sencillo en Python:

- nombre = input («¿Cómo te llamas? «)
- **print**(«Hola, « + nombre)

¿Qué hace este programa?

1. La primera línea pregunta tu nombre y guarda lo que escribes en una "caja" llamada **nombre**.
2. La segunda línea muestra un saludo usando el nombre que escribiste.

Cuando lo ejecutas, vemos algo así:

text
¿Cómo te llamas? Ana
Hola, Ana

El lenguaje de programación se sigue utilizando (y más que nunca) con la inteligencia artificial. De hecho, los lenguajes de programación son fundamentales para crear, entrenar y desplegar sistemas de IA. La IA no reemplaza la necesidad de programar; más bien, cambia la forma en que se programa y amplía lo que se puede hacer con la programación

1.2 ¿POR QUÉ SE SIGUE USANDO PROGRAMACIÓN CON IA?

Aunque la IA ha avanzado muchísimo —incluso puede escribir código—, los lenguajes de programación siguen siendo la base de todo sistema inteligente. Así como necesitamos palabras para construir frases, necesitamos

lenguajes de programación para construir modelos de IA y aplicaciones que los usen. Se explica porque:

1. La IA se programa: Para que una IA pueda reconocer una imagen o entender una pregunta, primero hay que escribir instrucciones en un lenguaje que la computadora entienda, como Python. Con Python, usamos programas especiales llamados TensorFlow o PyTorch. Estos programas ayudan a "enseñar" a la IA, mostrándole muchos ejemplos para que aprenda a reconocer cosas, como fotos de gatos o el significado de una pregunta.
2. La IA necesita integrarse en aplicaciones: Una IA por sí sola no sirve si no está conectada a un programa que la use. Para eso, se sigue programando. Por ejemplo, se puede usar JavaScript para hacer que una IA funcione dentro de una página web.
3. La IA no es mágica: Aunque hay herramientas que generan código automáticamente, es importante que una persona sepa leer, entender y corregir ese código. La programación sigue siendo clave para tener el control y garantizar que la IA funcione como queremos.
4. La IA ayuda, pero no sustituye al programador: Herramientas como GitHub Copilot o ChatGPT pueden ayudar a escribir código más rápido, pero no reemplazan la necesidad de saber programar. Al contrario, hacen que aprender a programar sea más accesible y productivo.

1.3 SIMILITUDES ENTRE LA INTELIGENCIA HUMANA Y LA INTELIGENCIA ARTIFICIAL

(*Figura 2*). La inteligencia artificial tiene muchas similitudes con la inteligencia humana. Nosotros tenemos 5 biosensores (vista, oído, olfato, gusto y tacto) que están permanentemente captando información del medio que nos rodea, y transformándola en información bioeléctrica para mandarla a nuestro procesador, que es nuestro cerebro, donde se encuentran los mejores algoritmos que jamás se hayan diseñado sobre esta tierra y que tienen cientos de miles de años de evolución.

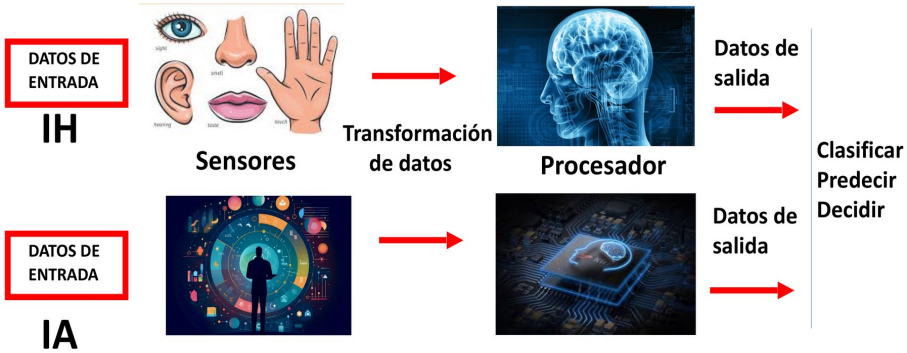


Figura 2. Similitudes Inteligencia humana e Inteligencia artificial

Con esa información buscan patrones, los clasifican y generan conocimiento para tomar decisiones.
La IA actúa de la misma manera. Tenemos sensores que captan información que a través del procesador la analiza y clasifica para predecir y decidir

1.4 FUENTES DE INFORMACIÓN DE LA IA

(Figura 3). La IA puede obtener información y datos de diversas fuentes, toda la imaginable de cuantas fuentes existen en el mundo. Algunas de las cuales incluyen:

- Bases de datos y conjuntos de datos
- Sensores y dispositivos de temperatura, humedad, luz
- Registros médicos electrónicos y otros datos de atención médica
- Publicaciones
- Redes sociales y otras plataformas en línea
- Cámaras y sistemas de vigilancia
- Transcripciones de conversaciones y grabaciones de audio
- Información de encuestas y estudios de mercado
- Datos meteorológicos y geoespaciales

1.5 PROGRAMACIÓN: EL PUNTO DE PARTIDA DE LA IA

Aunque ya hemos visto brevemente este tema de programación, vamos a volver ya que es el punto de inicio de la IA. Para estructurar un capítulo sobre los subtipos de IA, es fundamental comenzar desde la base: la programación. Ya hemos dicho que la programación es la disciplina fundacional que permite el desarrollo de cualquier sistema informático, pero no es IA en sí misma. Sin embargo, es el punto de partida imprescindible para comprender cómo se construyen y evolucionan los sistemas inteligentes. A partir de ahí, se puede avanzar hacia los distintos subtipos de IA: Machine Learning (ML), Deep Learning (DL) e Inteligencia Artificial Generativa (GAI o GenAI) *(Figura 4).*
Programar consiste en dar instrucciones explícitas a una máquina para que realice tareas

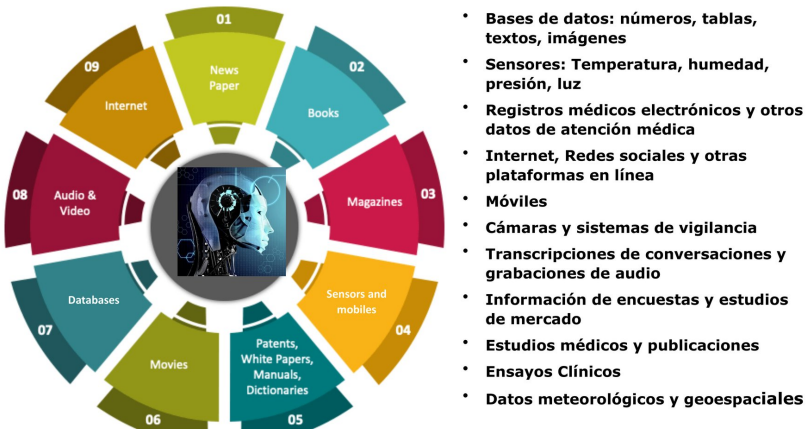


Figura 3. Fuentes de información de la IA

determinadas. Esto implica lógica, algoritmos y estructuras de datos, pero no inteligencia ni capacidad de aprendizaje autónomo. La programación clásica resuelve problemas mediante reglas fijas y procesos definidos por humanos, sin que el sistema aprenda o se adapte por sí mismo [1]

La programación es la base sobre la que se construyen los sistemas de IA. Sin conocimientos sólidos de programación, es imposible desarrollar algoritmos de aprendizaje automático o redes neuronales. Ya hemos visto que lenguajes como Python o JavaScript son ampliamente utilizados tanto en programación tradicional como en IA, ya que ofrecen potentes bibliotecas y herramientas especializadas para el desarrollo de modelos inteligentes [2]. En la programación tradicional tenemos los datos de entrada, un algoritmo conocido escrito por programadores y los datos de salida no conocidos *(Figura 5)*.

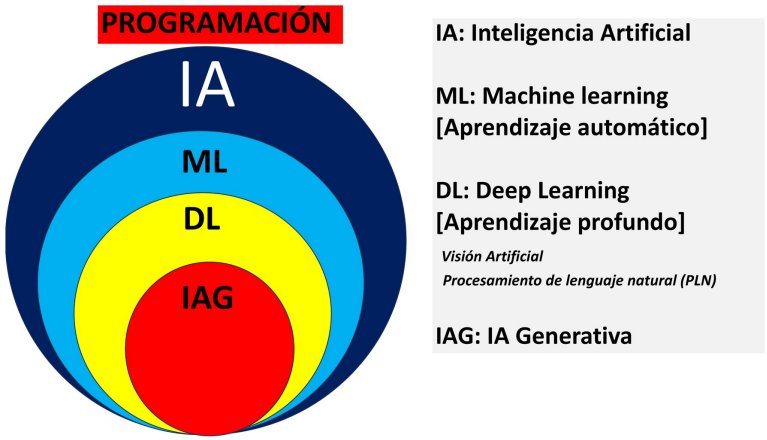


Figura 4. Programación y subtipos de inteligencia artificial



Figura 5. Ejemplo de programación

Veamos un **ejemplo de Programación de facturación de una tienda**. En la programación tradicional, como hemos dicho, tenemos los datos de entrada conocidos, un algoritmo conocido escrito por programadores y los datos de salida que no conocemos *(Figura 6)*.

- Entradas conocidas

  A_1: Precio unitario del producto; B_1: Cantidad de productos; C_1: % de descuento

- Algoritmo conocido:

  1. Multiplicar el precio unitario (A_1) por la cantidad (B_1) para obtener el subtotal.
  2. Calcular el monto de descuento aplicando el porcentaje (C_1) al subtotal.
  3. Restar el descuento al subtotal para obtener el total final.

- Salidas (no conocidas hasta ejecutar el algoritmo)

  A_2: Subtotal (precio unitario × cantidad); B_2: Monto de descuento; C_2: Total a pagar (subtotal - descuento)

Ejemplo con valores concretos:

- A_1 = €20 (precio unitario); B_1 = 3 (cantidad); C_1 = 10% (descuento)

Aplicando el algoritmo:

- A_2 = 20 × 3 = €60 (subtotal)
- B_2 = 10% de €60 = $6 (descuento)
- C_2 = $60 - $6 = €54 (total a pagar)

Así, el proceso sigue exactamente el flujo: partimos de tres entradas conocidas, aplicamos un algoritmo definido por el programador y obtenemos tres salidas que no se conocen hasta ejecutar el cálculo. Así que como hemos comentado previamente la programación tiene reglas fijas y procesos definidos por humanos, sin que el sistema aprenda o se adapte por sí mismo como es la Inteligencia Artificial.



Figura 6. Ejemplo de programación en la compra de tres ejemplares de un producto

REFERENCIAS

1 Joyanes Aguilar L. *Fundamentos de programación: algoritmos, estructura de datos y objetos*. 4.ª ed. México: McGraw-Hill; 2016. Accesible en: https://combomix.net/wp-content/uploads/2017/03/Fundamentos-de-programaci%C3%B3n-4ta-Edici%C3%B3n-Luis-Joyanes-Aguilar-2.pdf

2 KeepCoding. 5 mejores IA para programar en Python que tienes que probar [Internet]. 2025 disponible en: https://keepcoding.io/blog/mejores-ia-para-programar-en-python/.

2.

SUBTIPOS DE LA IA

La IA surge cuando los sistemas informáticos dejan de limitarse a ejecutar instrucciones fijas y empiezan a simular capacidades humanas como el razonamiento, el aprendizaje y la toma de decisiones autónomas. A continuación, se presentan los principales subtipos de IA, en orden de especialización y complejidad

1. Modelos de Aprendizaje Automático (ML: Machine Learning)
2. Modelos de Aprendizaje Profundo (DL: Deep Learning).

2.1 MODELOS DE APRENDIZAJE AUTOMÁTICO (MACHINE LEARNING ML)

En el Machine Learning tenemos los datos de entrada y de salida, pero lo que no tenemos es el algoritmo. Es decir, conocemos los datos de entrada y los de salida y con estos datos entrenamos al sistema para que encuentre el algoritmo por prueba y error. Se dan ejemplos y resultados, y el sistema aprende solo las reglas internas, incluso sin intervención humana. Luego, probamos el sistema para determinar si aprendió correctamente y le seguimos dando más datos para que siga perfeccionando su algoritmo *(Figura 7)*.

Ejemplo 1 de ML: Predecir el precio de una casa

- Entrada (Conocida): A1 Tamaño en m²; B1 número de habitaciones, C1: ubicación
- Salida (Conocida): Precios de venta de cada casa según las diferentes características: 250.000 € ;300.000 € ;350.000 € ;400.000 €…



Figura 7. Aprendizaje Automatico o Machine Learning

- El sistema aprende a estimar precios en función de esos factores.

Ejemplo 2 de ML: Predecir supervivencia a un año de un paciente con enfermedad renal crónica

- Datos de Entrada (conocidos): Edad del paciente, estadio de la ERC (Ej. Estadio 3b, 4, 5)), tasade filtrado glomerular (TFG), presión arterial sistólica y diastólica, niveles de creatinina y potasio, presencia de diabetes o hipertensión, albúmina en orina (proteinuria), adherencia al tratamiento
- Datos de Salida (conocidos): Paciente sobrevivió o no al año (Sí / No). (Estos datos se recogen de registros clínicos reales)
- Entrenamiento del Modelo: con los datos de miles de pacientes, el algoritmo aprende a detectar qué combinaciones de variables aumentan o reducen la probabilidad de supervivencia
- Predicción para Nuevos Pacientes: se introducen los datos de un nuevo paciente y el sistema te devuelve: la probabilidad de supervivencia a 1 año, digamos 85%. Incluso puede recomendarte variables críticas a corregir (ejemplo: controlar mejor la hipertensión o la proteinuria).
- Este tipo de modelo se usa en la práctica con algoritmos como regresión logística, árboles de decisión o XGBoost, y ayuda a los médicos a identificar pacientes de mayor riesgo y ajustar su manejo clínico.

Obviamente, el Machine Learning necesita muchos datos para entrenar. Pongamos otro ejemplo: miles de fotos de manzana (eso lo llamamos DATASET) para que aprenda (color, forma, etc) y luego probamos con una naranja a ver si aprendió o no.

Los tipos de algoritmos en ML son:

- Regresión: Predice valores numéricos. Ejemplo: predecir el precio de una casa según sus características.
- Clasificación: Asigna una categoría o etiqueta. Ejemplo: diferenciar manzana de pera
- Clustering o agrupamiento: Agrupa elementos similares sin saber de antemano a qué grupo pertenecen. Ejemplo: segmentar clientes por comportamiento de compra
- Asociación: Encuentra relaciones entre elementos. Ejemplo: "quien compra pan suele comprar leche" (como en los supermercados).

AUROC: Área bajo la curva

Cada algoritmo de IA tiene una sensibilidad y especificidad determinadas. Gráficamente, esto se representa en la (*Figura 8)* en un diagrama donde el eje x corresponde a (1 - especificidad) es decir la tasa de falsos positivos y el eje y a la sensibilidad que representa la tasa de verdaderos positivos El «área bajo la curva de la característica operativa del receptor» (AUROC) se utiliza para evaluar el rendimiento del modelo.

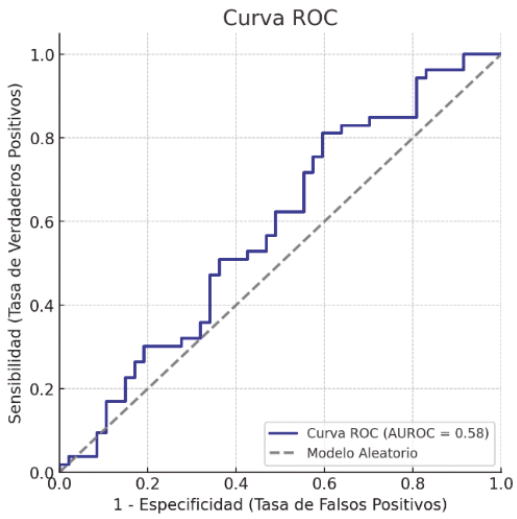


Figura 8. Área bajo la curva de la característica operativa del receptor (AUROC)

Cuanto mayor sea la AUROC, mejor distinguirá el modelo: valores por debajo de 0.7 suelen ser poco informativos y valores superiores a 0.9 se consideran excelentes.

2.1.1 Tipos de aprendizaje automático (ML)

Existen tres modelos: supervisado, no supervisado y reforzado *(Figura 9)*.

- Aprendizaje Supervisado: En este modelo, el algoritmo aprende a partir de datos etiquetados, es decir, ejemplos en los que la respuesta correcta ya es conocida. Enseñamos al sistema los resultados que queremos obtener, es decir la respuesta correcta (*Figura 10)*. Es lo que venimos comentando con el ejemplo anterior de predicción del precio de una casa. Si te digo **al 1 le corresponde el 2** y al 2 el 4 y al 4 el 8, con seguridad tú habrás asociado a que el algoritmo es multiplicar por 2. Otro ejemplo sería el de la **diferenciación de perros y gatos**: el aprendizaje automático supervisado distingue perros de gatos analizando miles de imágenes etiquetadas y aprendiendo patrones visuales como la forma de las orejas, el tamaño del hocico o la textura del pelaje para predecir



Figura 9. tipos de Machine Learning (aprendizaje automático)



Figura 10. Aprendizaje Supervisado

a qué clase pertenece una imagen nueva. Otro ejemplo sería la relación en un correo electrónico entre **spam y correo deseado**. Solo hay que dar muchos ejemplos de spam y correo deseado y su clasificación, y aprenderá a distinguirlos. Detecta palabras o frases clave: como «gratis», «haz clic aquí», «felicitaciones», uso excesivo de mayúsculas, símbolos ($$$) o enlaces.
La IA ajusta su modelo en función de los datos de entrenamiento para hacer predicciones precisas en nuevos casos.

- Aprendizaje no supervisado (agrupación): Solo disponemos de datos de entrada No enseñamos al sistema los resultados que queremos obtener. Encuentran los patrones subyacentes (clusters) en los datos a analizar *(Figura 11)*. Su objetivo es encontrar la estructura oculta del conjunto de datos, agruparlos según sus semejanzas y devolver una representación útil del conjunto buscando patrones de similitud de los datos de entrada. Esta solución de inteligencia artificial es perfecta para incluir en la estrategia de marketing. Por ejemplo, las personas que compran el artículo X (supongamos que un teléfono móvil) también tienden a comprar el artículo Y (un cargador o unos auriculares inalámbricos). Es lo que hace **Netflix** para recomendarte películas en función de tus gustos. O sea, agrupa los datos a analizar por características comunes.
En triaje de urgencias, un algoritmo de **clustering** analiza signos vitales y datos iniciales sin etiquetas previas, agrupando a los pacientes en perfiles de riesgo (bajo, medio, alto) para priorizar la atención
- Aprendizaje por refuerzo: Este modelo se basa en un agente que aprende a **tomar decisiones mediante prueba y error, recibiendo recompensas o penalizaciones según sus acciones** (*Figura 12*). Veamos un ejemplo simple de aprendizaje reforzado que ilustra los conceptos básicos de esta

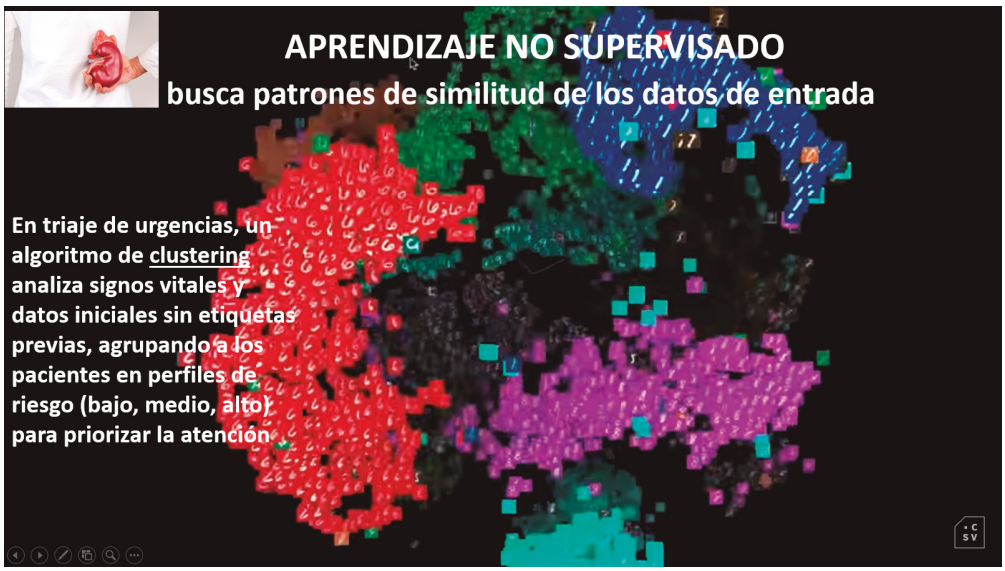


Figura 11. Aprendizaje no supervisado



Figura 12. Aprendizaje reforzado

técnica de aprendizaje automático. Imaginemos un ***robot aspirador*** que debe limpiar una habitación. El robot no tiene instrucciones explícitas sobre cómo limpiar la habitación, sino que debe aprender por sí mismo cómo hacerlo. El robot comienza por explorar la habitación y recopilando información sobre el entorno. Luego, el robot comienza a tomar acciones, como mover sus ruedas y su cepillo. El robot recibe una recompensa cuando limpia una zona de la habitación y una penalización cuando se atasca o choca con un obstáculo: un valor numérico positivo o negativo. A medida que el robot toma más acciones, aprende a limpiar la habitación de manera más eficiente. El robot aprende a evitar obstáculos, a limpiar todas las zonas de la habitación y a terminar su trabajo en el menor tiempo posible. Imagina que el robot está en una habitación y encuentra una zona sucia. Si se mueve hacia esa zona y la limpia, recibirá una recompensa alta. Esta recompensa positiva reforzará la acción de moverse hacia zonas sucias. Por el contrario, si choca con una pared, recibirá una recompensa negativa, lo que le enseñará a evitar los obstáculos.

Otro ejemplo muy práctico de aprendizaje reforzado es un ***coche autónomo*** que debe conducir por una carretera. El coche no tiene instrucciones explícitas sobre cómo conducir, sino que debe aprender por sí mismo cómo hacerlo. El coche utiliza sensores para recopilar información sobre el entorno, como la posición de otros vehículos, las señales de tráfico y las condiciones de la carretera. Luego, el coche utiliza esta información para tomar decisiones sobre cómo conducir, como acelerar, frenar o girar. El coche recibe una recompensa cuando conduce de manera segura y eficiente y una penalización cuando se produce un accidente o cuando se incumplen las normas de tráfico.

A medida que el coche toma más decisiones, aprende a conducir de manera más segura y eficiente. El coche aprende a evitar otros vehículos, a seguir las señales de tráfico y a adaptarse a las condiciones de la carretera. Así que, en el aprendizaje por refuerzo, el agente no solo toma decisiones, sino que

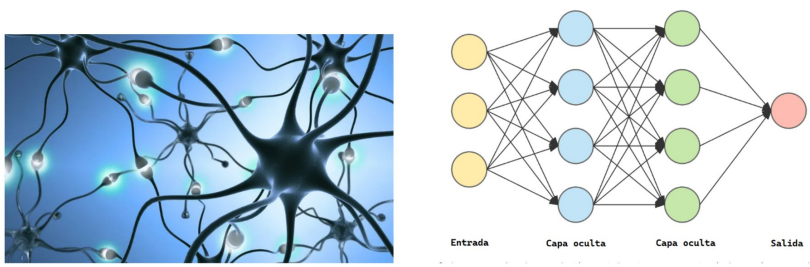
aprende de sus errores mediante un proceso de prueba y error. En el caso del ***triaje en urgencias***, cuando el sistema prioriza mal a un paciente (por ejemplo, atendiendo primero al paciente con fractura en lugar del que tiene dolor torácico agudo), recibe una recompensa negativa o penalización basada en los resultados clínicos adversos, como empeoramiento o mayor tiempo de espera para casos urgentes. Entonces, el agente ajusta su política para evitar repetir ese error en el futuro, «volviendo» a probar nuevas acciones y estrategias hasta optimizar la toma de decisiones. Este ciclo de ensayo, error y ajuste es fundamental en aprendizaje por refuerzo y permite que el sistema mejore continuamente, aprendiendo a priorizar correctamente.

2.2 MODELOS DE APRENDIZAJE PROFUNDO (DEEP LEARNING): REDES NEURONALES ARTIFICIALES

El Aprendizaje Profundo (DL) es una rama de la inteligencia artificial que utiliza redes neuronales para imitar la forma en que el cerebro humano aprende y procesa información [1]. El DL es un tipo de Machine Learning supervisado, donde el modelo aprende patrones a partir de grandes volúmenes de datos etiquetados, es decir, **se le proporciona tanto la entrada como la salida esperada**. Por ejemplo, para entrenar una red neuronal a reconocer cáncer de pulmón en imágenes, se le muestran miles de fotos de entrada, junto con la etiqueta correspondiente conocida (cáncer sí, o cáncer no) como salida. Con el tiempo, la red aprende a identificar cáncer de pulmón por sí sola en nuevas imágenes.

Las neuronas artificiales son iguales que las biológicas (*Figura 13*). Se llaman perceptrón y son unidades matemáticas: reciben, procesan y envían la información a otras neuronas formando una gran red. Cada neurona recibe entradas, realiza cálculos y produce una salida. Estas salidas se transmiten a través de las conexiones a otras neuronas y así sucesivamente, hasta obtener una salida final. Cada neurona procesa una tarea especifica y envía sus resultados a las otras neuronas de su red para trabajar juntas. Cuantas más capas más problemas complejos podrá resolver

Una neurona artificial (o perceptrón) es una unidad matemática. Podemos imaginar que una red neuronal es una hoja de cálculo gigante con miles de fórmulas interconectadas: cada celda es como una neurona que calcula algo,



Conexiones neuronales **Red Neuronal Artificial**

Figura 13. Conexiones y redes neuronales

pasa el resultado a otra, y así se construye una cadena de decisiones. Las neuronas artificiales no existen físicamente como elementos separados, sino que son estructuras lógicas dentro de un programa de ordenador. Están almacenadas como números en la memoria del ordenador.

2.2.1 Cómo funcionan las redes neuronales

Como ya hemos comentado, las redes neuronales son un tipo de algoritmo de aprendizaje automático que se inspira en el cerebro humano. Aunque pueden parecer complejas, las redes neuronales se componen de estructuras y componentes básicos que trabajan juntos para procesar información y producir una salida.

2.2.1.1 *Estructura y componentes básicos de una red neuronal*

La estructura básica de una red neuronal consta de capas de neuronas interconectadas. La capa de entrada recibe los datos y los envía a través de la red, mientras que la capa de salida produce la salida final. Entre la capa de entrada y la capa de salida, hay una o más capas ocultas que procesan la información. Cada neurona en la red neuronal está conectada a otras neuronas a través de conexiones que, al igual que en el mundo biológico, son conocidas como sinapsis. Cada sinapsis tiene un peso asociado que determina la fuerza de la conexión, (lo veremos en la práctica más adelante) y durante el proceso de entrenamiento, estos pesos se ajustan mediante el aprendizaje para mejorar la precisión y el resultado de la salida de la red.

2.2.1.2 *Proceso de entrenamiento y ajuste de pesos*

El proceso de entrenamiento de una red neuronal implica proporcionarle datos de entrada y esperar una salida ya conocida. Si la salida no coincide con la salida deseada, retrocede y ajusta los pesos de las conexiones en la red para mejorar la precisión. Este proceso se repite muchas veces hasta que la red aprende a producir la salida deseada. Hay varios algoritmos de entrenamiento utilizados en las redes neuronales, incluyendo el algoritmo de retropropagación. En este algoritmo, se calcula el error de la salida de la red en comparación con la salida deseada y se ajustan los pesos de las conexiones para minimizar ese error.

2.2.1.3 *Funciones de activación y su importancia*

Las funciones de activación son una parte fundamental de cómo funcionan las redes neuronales. Cada neurona en la red neuronal utiliza una función de activación para determinar su salida en función de la entrada que recibe. Las funciones de activación pueden ser lineales o no lineales. Las funciones lineales simplemente multiplican la entrada por un peso y agregan un término de sesgo. Las funciones no lineales, por otro lado, son más complejas y permiten a la red modelar relaciones no lineales en los datos. La elección de la función de activación puede tener un impacto significativo en la capacidad de la red para aprender y generalizar a nuevos datos. Algunas funciones de activación comunes incluyen la función sigmoide, la función ReLU y la función tangente hiperbólica.

2.2.2 Ejemplos de funcionamiento de redes neuronales *(Figura 14)*

Las neuronas funcionan de la siguiente manera: En primer lugar, tienen datos de entrada y cada uno tiene un peso w que significa la importancia de este dato. Por ejemplo, la edad tendrá un peso de gran valor porque es muy importante en el cálculo de la supervivencia mientras que por ejemplo los hábitos de lectura no lo serán tanto. Luego estos datos y sus pesos se multiplican y se suman en el cuerpo de la neurona o perceptrón. Después, el resultado de ese cálculo se pasa por

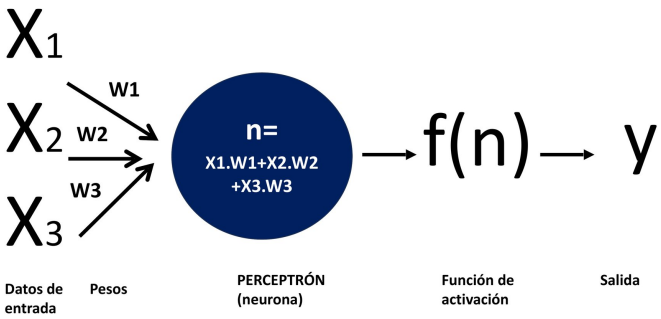


Figura 14. Funcionamiento de red neuronal

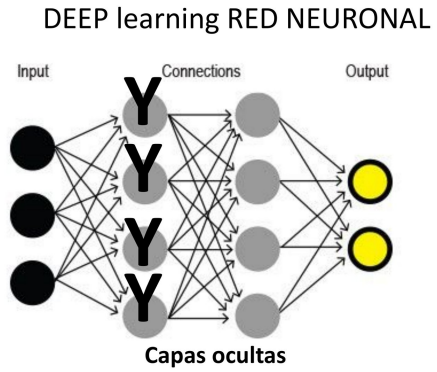


Figura 15. El dato de salida es el dato e entrada de una o más neuronas en la red

una función matemática de activación que da el dato de salida. Y como vemos en la siguiente *(Figura 15),* ese dato Y de salida es el dato de entrada de una o más neuronas en la red.

Las redes neuronales son aprendizaje supervisado. Es decir, reciben una serie de datos de entrenamiento y sus respectivos valores de salida correctos, lo que significa que ya conocemos cuales son las respuestas. Con estos datos, el sistema intenta calcular los pesos y aplica la función de activación al resultado. La salida se compara con el resultado real, el que conocemos de antemano y se encuentra el error promedio, es decir la diferencia con el resultado que ya conocíamos. Con ese error promedio se regresa hacia atrás a través de todas las capas de la red (back) para ajustar los pesos. Con los pesos ajustados, se vuelven a hacer todos los cálculos (forward) y se vuelve a comparar la salida con el resultado de entrenamiento, se calcula el error y de nuevo se vuelve hacia atrás. Este proceso de ajustar, volver a encontrar lo errores, volver a ajustar y en cada vuelta volver a hacerlo mejor se repite varias veces hasta que el error de 0 o lo más cercano a 0. Es por eso por lo que decimos que los modelos se perfeccionan a sí mismo sin intervención humana. Cada uno de estos pesos son ajustados en cada vuelta o época y cada uno de ellos se conoce como parámetro. Como ejemplo, Chat GPT tiene 175 millones de parámetros. Veamos ejemplos

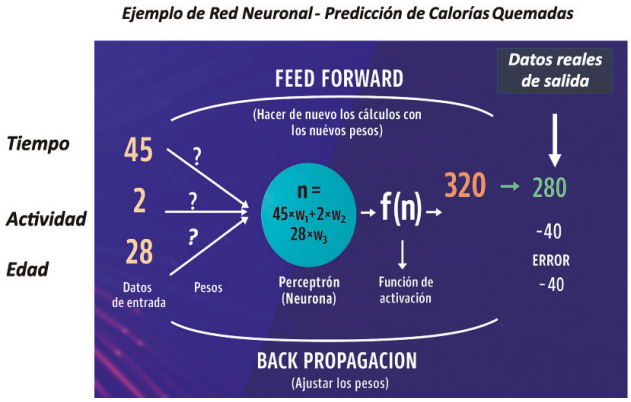


Figura 16. Predicción de calorías quemadas

Ejemplo 1: Predicción de calorías quemadas *(Figura 16)*

Datos de entrada:

- Minutos de ejercicio: 45
- Tipo de actividad (codificado): 2 (por ejemplo, 1=caminata, 2=correr, 3=bicicleta)
- Edad del usuario: 28

Fórmula del perceptrón (neurona):

$n = (45 \times w_1) + (2 \times w_2) + (28 \times w_3)$

Los pesos w_1, w_2, w_3 son los que la red debe aprender automáticamente para predecir bien.

Proceso:

1. Se hace el cálculo con pesos iniciales al azar (Feed Forward).
2. Se obtiene un resultado: por ejemplo, 320 calorías.
3. Se compara con el valor real que conocemos por estudios previos en esas condiciones exactas: por ejemplo, 280 calorías.
4. Se mide el error.
5. Se vuelve atrás y se reajustan los pesos para mejorar (esto se llama Backpropagation).
6. Se repite el proceso hasta que el error sea muy pequeño.

Resultado:

La red aprende a predecir, con buena precisión, cuántas calorías quema una persona según su tiempo de ejercicio, tipo de actividad y edad.

Ejemplo 2: préstamo *(Figura 17)*

Imaginemos que somos un banco y queremos decidir si otorgar o no un préstamo a un cliente. Tradicionalmente, esta decisión se basaba en un conjunto de reglas establecidas (edad, ingresos, historial crediticio, etc.). Sin embargo, con el aprendizaje profundo, podemos analizar una cantidad mucho mayor de datos y de una manera más compleja.

¿Cómo funciona?

A. Recopilación de datos: El banco recopila una gran cantidad de información sobre sus clientes, como:
   1. Datos demográficos (edad, sexo, ubicación)
   2. Historial financiero (ingresos, deudas, pagos)
   3. Comportamiento en línea (interacciones con la app, búsquedas)

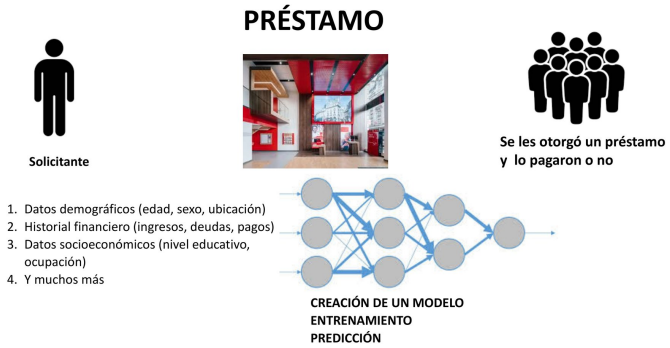


Figura 17. Ejemplo de concesión de préstamo

4. Datos socioeconómicos (nivel educativo, ocupación)
5. Y muchos más...

B. Creación de un modelo: Se crea un modelo de aprendizaje profundo (una red neuronal) que aprende a identificar patrones en estos datos. Estos patrones pueden ser muy complejos y difíciles de detectar por un humano.

C. Entrenamiento del modelo: El modelo se entrena con una gran cantidad de datos históricos de clientes a los que se les otorgó un préstamo y sabemos si lo pagaron o no. De esta manera, el modelo aprende a asociar ciertas características de los clientes con una mayor o menor probabilidad de impago.

D. Predicción: Una vez entrenado, el modelo puede analizar los datos de un nuevo cliente y predecir la probabilidad de que pague el préstamo.

Supongamos que el modelo identifica que los clientes con un historial de pagos puntuales en servicios públicos, un alto nivel de educación y una búsqueda frecuente de productos financieros en línea tienen una probabilidad muy baja de impago. Si un nuevo cliente cumple con estas características, el modelo recomendará otorgarle el préstamo

Ejemplo 3: ¿será contratado un candidato? *(Figura 18)*

En este ejemplo, los datos de entrada son los años de experiencia, el nivel de inglés y el número de certificaciones de un candidato. Pero todas estas entradas no tienen la misma importancia. Cada dato se multiplica por un peso que representa su importancia (por ejemplo, las certificaciones pesan más que el inglés). Luego, todos los resultados se suman en una "neurona" y se aplica una función de activación que decide si el valor total supera un umbral. Si lo supera, la red responde **«Sí» (será contratado)**; si no, responde **«No»**. Así, la red aprende a tomar decisiones evaluando múltiples factores a la vez.

Entradas (características del candidato):
- Años de experiencia: 5
- Nivel de inglés (0 a 10): 7
- Número de certificaciones: 2

Pesos asignados (importancia de cada característica):
- Experiencia × 2
- Inglés × 1
- Certificaciones × 3

Cálculo del perceptrón (n):

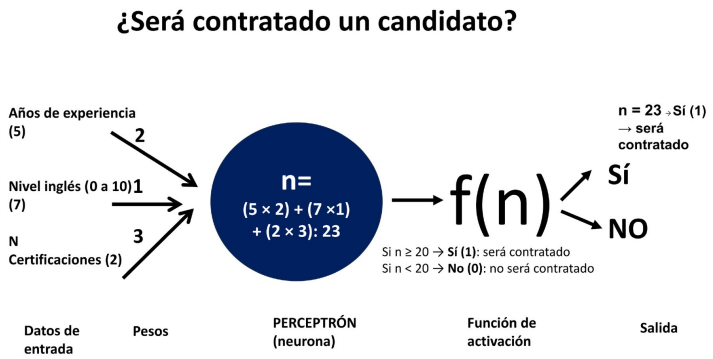


Figura 18. Ejemplo de contratación de un candidato

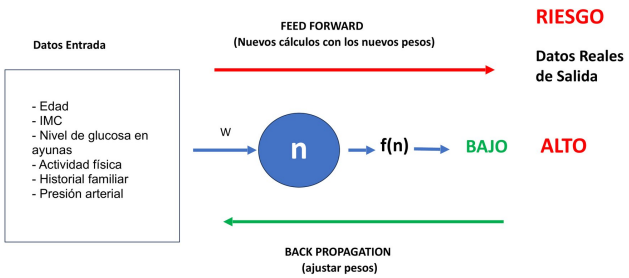


Figura 19. Red neuronal para predicción de riesgo de diabetes

n = (Experiencia × 2) + (inglés × 1) + (Certificaciones × 3) n = (5 × 2) + (7 × 1) + (2 × 3)
n = 10 + 7 + 6 = 23

Función de activación:
- Si n ≥ 20 → **Sí (1)**: será contratado
- Si n < 20 → **No (0)**: no será contratado

Resultado final:
- n = 23 → **Sí (1) → El candidato es contratado**

Ejemplo 4: predicción del riesgo de diabetes *(Figura 19)*
Una red neuronal puede predecir el riesgo de diabetes tipo 2 analizando variables clínicas como:

- Edad
- IMC
- Nivel de glucosa en ayunas
- Actividad física
- Historial familiar
- Presión arterial

Cada valor se pondera con un peso aprendido y se combina en una neurona. La función de activación transforma el total en una predicción:

0 = bajo riesgo, 1 = alto riesgo.

El sistema ajusta sus pesos comparando la predicción con la realidad que conocemos basados

en miles de casos registrados, mejorando con cada nuevo ejemplo analizado.

Ejemplo 5 Triaje en Urgencias *(Figura 20)*

- Datos de entrada: Se incluyen variables clínicas relevantes como signos vitales, síntomas, dolor torácico, disnea, antecedentes y edad. Estos datos son los que habitualmente recopila el personal de triaje y constituyen la base para la toma de decisiones automatizada.
- Datos de salida conocidos: Riesgo alto o bajo según la realidad previa de miles de casos
- Red Neuronal (n): Representa el nodo o neurona artificial que recibe los datos ponderados (w) y aplica una función de activación (f(n)) para generar una salida. El proceso de "feed forward" implica calcular la salida a partir de los datos y los pesos actuales de la red.
- Backpropagation: Es el mecanismo de aprendizaje supervisado mediante el cual la red ajusta sus pesos en función del error cometido, (por ejemplo, calculó inicialmente riesgo bajo cuando en base a los datos conocidos con experiencia práctica anterior introducida en el sistema a ese paciente le corresponde riesgo alto) optimizando así la precisión en la clasificación de los niveles de urgencia.
- Salida Nivel de Urgencia: La red neuronal clasifica a los pacientes en niveles de urgencia: bajo, medio o alto. Esta categorización es fundamental para priorizar la atención en situaciones de saturación y mejorar la gestión de los recursos disponibles.

2.2.3 Aplicaciones de Aprendizaje profundo (DL)

2.2.3.1 Vision artificial

La visión artificial entrena a las computadoras para «ver» y entender imágenes o videos mediante algoritmos, cámaras y datos, replicando aspectos de la visión humana. A diferencia de los humanos, que usamos retinas y el cerebro, las máquinas usan sensores (como cámaras), bases de datos y redes neuronales para procesar información visual.

Se apoya principalmente en algoritmos de aprendizaje profundo (Deep Learning), sobre todo en redes neuronales convolucionales (CNNs), que están especialmente diseñadas para identificar patrones espaciales (como bordes, formas, colores y texturas) en imágenes [2][3].

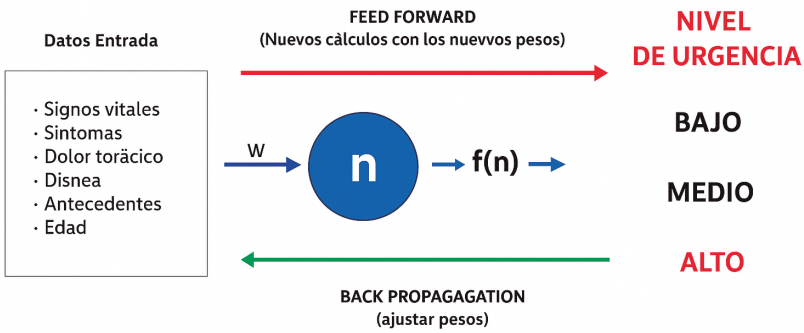


Figura 20. Red neuronal para triaje en urgencias

2.2.3.1.1 Pixeles

Los píxeles son la base de las imágenes digitales. Cada imagen se descompone en una cuadrícula de píxeles, que son pequeños puntos con valores numéricos (*Figura 21)* que representan:

- Intensidad de color: En imágenes en escala de grises, cada píxel tiene un valor entre 0 (negro) y 255 (blanco). En imágenes a color, los píxeles se dividen en canales RGB (rojo, verde, azul), donde cada canal tiene un valor de 0 a 255, combinándose para formar colores.
- Información espacial: Los píxeles están organizados en una matriz (por ejemplo, 1920x1080 píxeles), lo que permite a los algoritmos analizar patrones espaciales.

Como vemos la imagen se reduce a una matriz numérica donde cada píxel describe un fragmento de la imagen. Los algoritmos de visión artificial analizan estos números para identificar formas, bordes o patrones.

Cada píxel de una imagen a color se representa por tres números (RGB: rojo, verde y azul), cada uno con un valor entre 0 y 255. Estos números determinan la intensidad de cada color primario en ese píxel, y al combinarlos obtenemos la amplia gama de colores que vemos en las imágenes digitales

2.2.3.1.2 Fases del funcionamiento de la visión artificial (*Tabla 1*)

Su funcionamiento se organiza en una serie de fases secuenciales que permiten transformar una imagen en información útil para la toma de decisiones automatizadas. Estas fases comienzan con la **adquisición de la imagen** mediante cámaras o sensores, seguida del **preprocesamiento**, donde se mejora la calidad visual para facilitar el análisis. Posteriormente, se realiza la **segmentación** de la imagen en regiones relevantes, y la **extracción de características** para identificar patrones visuales significativos. A continuación, se procede a la **clasificación o reconocimiento** de objetos mediante algoritmos de aprendizaje

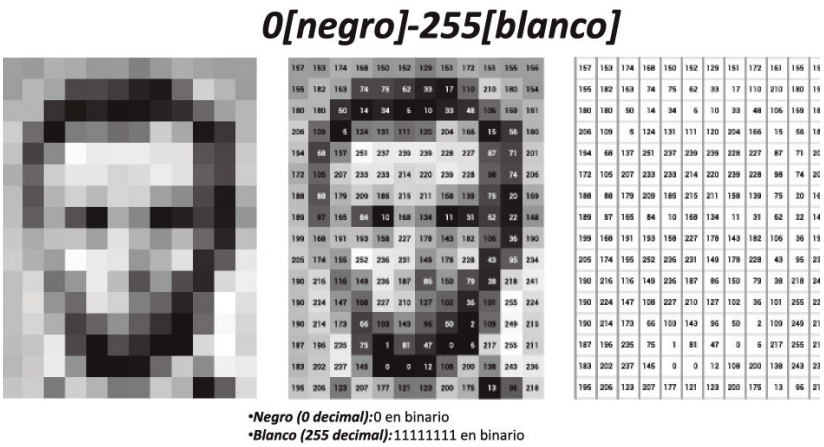


Figura 21. Matriz numérica donde cada píxel describe un fragmento de la imagen.

Tabla 1. Fases del funcionamiento de la visión artificial

FASE	DESCRIPCIÓN
1. Adquisición de imagen	Captura de imágenes mediante sensores ópticos o cámaras digitales que transforman la realidad en datos visuales.
2. Preprocesamiento	Optimización de la imagen: reducción de ruido, mejora de contraste, filtrado y normalización.
3. Segmentación	Separación de la imagen en regiones o estructuras relevantes (objetos, bordes, áreas de interés).
4. Extracción de características	Obtención de atributos como formas, colores, texturas o patrones geométricos.
5. Clasificación / Reconocimiento	Asignación de etiquetas o categorías a los elementos detectados mediante Machine Learning o redes neuronales profundas.
6. Interpretación / Decisión	Análisis contextual para diagnóstico, predicción o control de procesos (ej.: detección de tumores, control de calidad).
7. Retroalimentación / Acción	En sistemas autónomos, ejecución de la respuesta adecuada (ej.: mover un robot, emitir una alerta, ajustar parámetros).

automático. La fase de **interpretación o decisión** analiza los datos visuales en su contexto para generar una respuesta adecuada, que finalmente puede traducirse en una **acción automatizada** si el sistema lo requiere. Esta secuencia es clave para aplicaciones en medicina, robótica, industria y vehículos autónomos, entre otros campos.

2.2.3.1.3. Aplicación de la visión artificial en Medicina

La IA está transformando el análisis de imágenes médicas. Gracias a técnicas como el aprendizaje profundo, hoy es posible automatizar la detección de alteraciones en estudios como ecografías, tomografías y biopsias digitalizadas. Estas herramientas no solo aumentan la precisión diagnóstica, sino que también reducen el tiempo de interpretación y facilitan la identificación de patrones sutiles que pueden pasar desapercibidos al ojo humano. La IA representa, así, un aliado prometedor para los médicos, con el potencial de mejorar el seguimiento clínico, optimizar recursos y ofrecer una medicina más personalizada [1]. Un ejemplo puede ser:

Detección de melanoma cutáneo con visión artificial

1. Se toman miles de fotos de lunares benignos y malignos ya clasificados por dermatólogos.
2. Se entrena una red neuronal convolucional con estas imágenes.
3. El sistema aprende a distinguir características sutiles: forma irregular, bordes borrosos, variación de color.
4. Cuando se le da una imagen nueva, el sistema compara y predice si se parece más a un melanoma o no.
5. Resultado *"Probabilidad de melanoma: 87% – Requiere derivación a dermatología urgente"*

Este tipo de sistemas, entrenados adecuadamente, han demostrado en algunos estudios igualar o superar la precisión de dermatólogos humanos en la detección temprana de cáncer de piel [4].

Detección de neumonía con IA en un departamento de urgencias [5]

En la (Tabla 2) podemos ver el flujo diagnóstico de detección de neumonía.

Una explicación más desarrollada `puede encontrarla el lector en el capitulo La Inteligencia Artificial en la Imagen Medica (radiología e Histopatologia) en el libro Artificial La nueva Inteligencia y su aplicación en Nefrologia[1].

2.2.3.2 Procesamiento del lenguaje natural (PLN)

El Procesamiento del Lenguaje Natural (PLN), o Natural Language Processing (NLP), es una rama de la inteligencia artificial que permite a las máquinas comprender, interpretar, generar y responder al lenguaje humano, ya sea en formato oral o escrito [6] A diferencia de entradas más estructuradas como imágenes o datos numéricos, el lenguaje humano es inherentemente complejo debido a su ambigüedad, contexto, emociones, errores gramaticales y matices culturales. El PLN busca que los sistemas informáticos procesen este tipo de información de manera similar a como lo hace un ser humano, pero claro existen limitaciones

Tabla 2. Detección de neumonía con IA en un departamento de urgencias

ETAPA	DESCRIPCIÓN
1. Adquisición de la imagen	Al paciente con síntomas respiratorios se le realiza una radiografía de tórax, que es el método de imagen de primera línea en urgencias para evaluar anormalidades pulmonares.
2. Análisis automatizado	La imagen se introduce en un sistema de IA entrenado previamente con miles de radiografías de tórax de pacientes con y sin neumonía. Estos sistemas suelen estar basados en redes neuronales convolucionales (CNN), capaces de identificar patrones y características sutiles en las imágenes que pueden pasar desapercibidos al ojo humano.
3. Procesamiento y predicción	El algoritmo analiza la radiografía, detectando signos típicos de neumonía (infiltrados, consolidaciones, distribución de las lesiones, etc.). El sistema compara la imagen nueva con los patrones aprendidos y calcula la probabilidad de que exista neumonía, e incluso puede sugerir la severidad o el tipo (viral o bacteriana).
4. Resultado inmediato	En cuestión de segundos, el sistema emite un informe automático, por ejemplo: "Probabilidad de neumonía: 92%. Hallazgos compatibles con neumonía en lóbulo inferior derecho. Requiere valoración clínica y tratamiento antibiótico urgente."
5. Apoyo a la decisión médica	El resultado se presenta al médico de urgencias, quien lo utiliza como apoyo para tomar decisiones rápidas sobre el tratamiento, la necesidad de aislamiento, ingreso hospitalario o realización de pruebas adicionales.

por ejemplo en la interpretación de contextos altamente subjetivos o especiales de culturas diferentes

2.2.3.2.1 Funcionamiento del PLN

El PLN se basa en múltiples niveles de análisis lingüístico y etapas técnicas que permiten descomponer y procesar el lenguaje humano de manera sistemática. Estas etapas y niveles se detallan en las (*Tabla 3*) y (*Tabla 4*).

Tareas del PLN

El PLN abarca una amplia variedad de tareas que permiten a los sistemas interpretar, generar y analizar lenguaje humano. Estas tareas tienen aplicaciones prácticas en diversos sectores, desde el comercio hasta la medicina (*Tabla 5*).

Tabla 3. Niveles de análisis lingüístico en PLN

NIVEL DE ANÁLISIS	DESCRIPCIÓN
Análisis morfológico	Descompone palabras en morfemas para identificar raíces y afijos.
Análisis sintáctico	Reconoce la estructura gramatical de las oraciones.
Análisis semántico	Interpreta el significado contextual, resolviendo ambigüedades léxicas.
Análisis pragmático	Considera el contexto para interpretar ironía, metáforas o intenciones.

Tabla 4. Procesamiento técnico en PLN

ETAPA	DESCRIPCIÓN BREVE
Tokenización	Divide el texto en unidades básicas: palabras, frases o subunidades léxicas.
Lematización / Stemming	Reduce las palabras a su forma base o raíz (por ejemplo, “corriendo” → “correr”).
Etiquetado gramatical	Asigna a cada palabra su categoría gramatical (sustantivo, verbo, adjetivo...).
Análisis sintáctico	Identifica la estructura gramatical de las oraciones (sujeto, predicado, etc.).
Análisis semántico	Interpreta el significado contextual de las palabras y oraciones.

Tabla 5. Tipos de tareas del PLN

TAREA	EJEMPLO DE APLICACIÓN
Clasificación de texto	Filtrado automático de correos electrónicos como spam o no spam.
Análisis de sentimientos	Monitorizar opiniones de usuarios sobre productos en redes sociales.
Reconocimiento de entidades	Detectar nombres de empresas o ubicaciones en noticias financieras.
Resumen automático	Resumir artículos de prensa para plataformas de noticias.
Generación de texto	Redactar descripciones automáticas de productos en tiendas en línea.
Traducción automática	Traducir manuales técnicos entre diferentes idiomas.
Chatbots	Asistentes virtuales para atención al cliente en sitios web de e-commerce.

En resumen, el funcionamiento del PLN se basa en los siguientes pasos:

Preentrenamiento: Antes de ser utilizado en conversaciones en tiempo real, el modelo es entrenado en grandes cantidades de texto de diferentes fuentes, como libros, artículos y sitios web. Durante este proceso, el modelo aprende a entender el lenguaje y a predecir la siguiente palabra en una oración dada su comprensión del contexto.

Procesamiento del texto de entrada: Cuando proporcionas una consulta o una pregunta, el modelo analiza y procesa cada palabra y su relación con las palabras circundantes. Utiliza técnicas avanzadas de procesamiento del lenguaje natural para comprender la semántica y el contexto de la consulta.

Generación de respuesta: Después de comprender la entrada, el modelo utiliza su conocimiento previo para predecir y generar una respuesta coherente y relevante. La respuesta se forma en función de patrones y estructuras lingüísticas aprendidas durante el entrenamiento, y el modelo intenta generar una respuesta que sea apropiada y contextualmente relevante.

Optimización del texto de salida: Antes de proporcionar la respuesta final, el modelo verifica y ajusta la coherencia y la gramática de la respuesta generada. Esto implica asegurarse de que la respuesta sea lógica, coherente y esté bien redactada.

Entrega de la respuesta: Finalmente, el modelo proporciona la respuesta generada en función de la entrada original, intentando proporcionar una solución completa y comprensible.

2.2.3.2.2 Herramientas comerciales de PLN (julio 2025)

El ecosistema del PLN en julio de 2025 está dominado por herramientas comerciales avanzadas que integran capacidades multimodales, búsqueda inteligente y automatización de tareas (*Tabla 6*).

Tabla 6. Herramientas comerciales de PLN (julio 2025)

HERRAMIENTA	DESCRIPCIÓN BREVE
ChatGPT 5 / GPT-4o (OpenAI)	Asistente multimodal con razonamiento "Strawberry", navegación web autónoma ("Deep Research"), grabación y resumen de reuniones, integración de APIs y agentes ("Operator").
OpenAI Operator	Agente autónomo que completa formularios, pedidos online y gestiona citas.
OpenAI Deep Research (DeepSearch)	Agente de investigación que busca, lee y sintetiza múltiples fuentes online, entregando informes con citas.
Microsoft 365 Copilot / Copilot Vision	Integrado en Windows 11/365/Teams/Office; permite transcribir, generar contenido, interpretar pantallas ("Vision"), automatizar acciones web ("Copilot Actions") y personalizar agentes ("Copilot Tuning").
GitHub Copilot	Autocompletado y generación de código en IDEs, compatible con múltiples LLMs como GPT, Claude, Gemini y LLaMA.
Google Gemini	Asistente de búsqueda y codificación, incluido en "AI Mode" de Search; también ofrece capacidades de investigación profunda.
Perplexity AI / Perplexity Deep Research	Motor conversacional que procesa y sintetiza docenas de fuentes web en informes autónomos con citas.
Grok 3 (xAI)	Chatbot en X/Twitter con razonamiento avanzado ("Think", "Big Brain"), tono informal, y motor "DeepSearch" para investigación.
Anthropic Claude 4	Modelos "Opus 4" y "Sonnet 4" diseñados para diálogos extensos, con enfoque en razonamiento ético y uso empresarial.
Meta LLaMA 3/4	Modelos open-source utilizados en investigación, desarrollo y aplicaciones personalizadas.
IBM Deep Search	Plataforma empresarial para búsqueda semántica en grandes colecciones (PDFs, multimedia), que convierte documentos en JSON y crea entornos de búsqueda semántica.

2.2.3.2.3 Ejemplo de Aplicación de PLN: Detección de Testimonios Verdaderos o Falsos en declaraciones policiales

En el ámbito judicial, el procesamiento del lenguaje natural (PLN) se está utilizando como herramienta de apoyo para analizar declaraciones de testigos y acusados, con el fin de identificar posibles indicios de veracidad o falsedad. Durante un juicio por fraude financiero, los investigadores recopilan las transcripciones de los interrogatorios de varios implicados. Un sistema basado en PLN analiza estas declaraciones en busca de patrones lingüísticos que históricamente se han asociado con el engaño o la verdad.

1. Análisis lingüístico: El sistema analiza elementos como la longitud de las respuestas, el uso de evasivas, contradicciones gramaticales, cambios repentinos de tiempo verbal, exceso de detalles irrelevantes o el uso repetido de negaciones («yo no hice nada», «nunca supe nada»).
2. Modelos entrenados en patrones de engaño: Se utilizan modelos entrenados previamente con corpus de declaraciones clasificadas como verdaderas o falsas (por ejemplo, estudios psicológicos, bases de datos forenses). Estos modelos aprenden a reconocer señales sutiles en el lenguaje que podrían ser indicadores de mentira.
3. Análisis de sentimientos y emociones: El sistema también evalúa la carga emocional del discurso. Por ejemplo, un testigo que muestra emociones inconsistentes con la situación descrita (frialdad al relatar un crimen violento) puede despertar sospechas.
4. Detección de entidades y hechos clave: El modelo extrae nombres, fechas y lugares, y compara estos datos con otras evidencias del caso (como registros telefónicos o cámaras de seguridad), para detectar posibles incongruencias.

Un estudio reciente presenta un dataset específico para analizar declaraciones dadas durante entrevistas policiales. Utiliza un modelo basado en DistilBERT, entrenado para distinguir declaraciones verdaderas de falsas, logrando una precisión del 86 % [7]. El PLN no sustituye al juez ni al perito, pero entrega un informe con indicadores de posible engaño, lo que permite enfocar mejor las preguntas y cruzar los datos más críticos. Esto mejora la eficiencia y calidad del proceso judicial.

2.2.3.2.4 Ejemplo de aplicación del PLN en medicina de urgencias

En el contexto de la medicina de urgencias, el PLN tiene un potencial transformador al procesar grandes volúmenes de datos no estructurados, como notas clínicas, informes de triaje o registros de pacientes, para mejorar la toma de decisiones y optimizar los flujos de trabajo. Un ejemplo concreto es el uso del PLN para la clasificación automática de la gravedad en el triaje de urgencias.

Imaginemos un hospital que recibe a un paciente con dolor torácico. El médico registra una nota clínica inicial: "Paciente de 55 años, dolor torácico opresivo de inicio súbito, acompañado de disnea y sudoración". Un sistema basado en PLN, como los integrados en herramientas como Chat GPT, Microsoft 365 Copilot, Gemini, Grok 3…etc puede procesar esta nota en tiempo real mediante las siguientes etapas:

- Tokenización y lematización: El sistema descompone la nota en palabras clave como "dolor torácico", "opresivo", "súbito", "disnea" y "diaforesis".
- Etiquetado gramatical y análisis sintáctico: Identifica que "dolor torácico" es el sujeto principal y que términos como "opresivo" y "súbito" son descriptores clave.
- Análisis semántico y reconocimiento de entidades: Reconoce que "dolor torácico opresivo" y "disnea" son síntomas asociados con condiciones graves como el infarto de miocardio.
- Clasificación de texto: Utilizando modelos entrenados, el sistema clasifica la nota como de alta prioridad, sugiriendo un triaje de nivel 1 o 2 según escalas como el Sistema de Triaje de Emergencia (ETS).
- Generación de texto: El sistema puede generar una alerta automática para el equipo médico, como: "Paciente con posible evento cardiovascular agudo, requiere evaluación inmediata".

Este proceso, que combina varias tareas del PLN (clasificación de texto, reconocimiento de entidades y generación de texto), permite priorizar casos críticos en segundos, reduciendo el tiempo de respuesta en un entorno de alta presión como las urgencias. Estudios han demostrado que el uso de PLN en triaje puede mejorar la precisión en la clasificación de pacientes hasta en un 20% en comparación con métodos manuales [8]

2.2.3.2.5 Aplicaciones potenciales en el día a día del uso de PLN en Medicina

El PLN tiene varias aplicaciones potenciales en el día a día de un médico, que pueden ayudar en la toma de decisiones clínicas, la educación de los pacientes y la gestión de la carga de trabajo. Algunos ejemplos de casos de uso de utilización son:

1. **Asistencia en la toma de decisiones clínicas**:
   - Diagnóstico diferencial: Un médico puede utilizar el chat GPT para generar una lista inicial de posibles diagnósticos en función de los síntomas y la historia clínica proporcionada por el paciente.
   - Revisión de literatura: El médico puede solicitar al chat GPT que resuma los últimos estudios médicos relevantes sobre un tema específico para estar al día con las últimas investigaciones.
     Ejemplo: «¿Cuáles son las últimas investigaciones sobre el tratamiento de la diabetes tipo 2?»
2. **Educación de pacientes:**
   - Explicación de condiciones médicas: El médico puede generar descripciones fáciles de entender sobre condiciones médicas, procedimientos o tratamientos para compartir con los pacientes.
     Ejemplo: «Explícame en términos sencillos qué es la hipertensión y cómo puedo controlarla.»
   - Consejos de salud: Proporcionar recomendaciones generales sobre la dieta, el ejercicio y el estilo de vida saludable.
     Ejemplo: «¿Cuáles son algunos consejos para mantener un corazón saludable?»
3. **Documentación médica y notas clínicas:**
   - Transcripción de notas clínicas: El chat GPT puede ayudar a los médicos a transcribir notas clínicas dictadas durante las consultas.
     Ejemplo: «Por favor, transcribe las notas de la consulta de hoy con el paciente X.»
4. **Resolución de dudas rápidas:**
   - Respuestas a preguntas frecuentes: Los médicos pueden utilizar el PLN para obtener respuestas rápidas a preguntas comunes sobre medicamentos, dosis recomendadas, interacciones medicamentosas, etc.
     Ejemplo: «¿Cuál es la dosis recomendada de ibuprofeno para un adulto?»
5. **Planificación de la atención al paciente:**
   - Generación de planes de tratamiento preliminares: Ayuda en la creación de planes de tratamiento iniciales antes de la consulta con el paciente.
     Ejemplo: «Crea un plan de tratamiento inicial para un paciente con diabetes tipo 1 recién diagnosticado.»
6. **Investigación clínica:**
   - Análisis de datos médicos: El PLN puede ayudar a analizar datos clínicos, identificar tendencias y generar informes preliminares [9]
     Ejemplo: «Analiza los datos de los pacientes en el último ensayo clínico y resalta las tendencias más relevantes».
7. **Gestión del tiempo:**
   - Gestión de citas y recordatorios: El chat GPT puede ayudar a programar citas, enviar recordatorios a pacientes y gestionar la agenda del médico.

Ejemplo: «Programa una cita con el paciente Y para la próxima semana y envía un recordatorio.

2.2.3.3 *Inteligencia artificial multimodal*

Es un tipo de inteligencia artificial generativa capaz de procesar y combinar diferentes tipos de datos, como texto, imágenes, audio y video, para generar respuestas más completas y sofisticadas. A diferencia de los modelos tradicionales, que se enfocan en un solo tipo de entrada (por ejemplo, solo texto o solo imágenes), los modelos multimodales integran múltiples fuentes de información para mejorar su comprensión y generación de contenido (*Figura 22*).



Figura 22. IA generativa multimodal

Las ventajas de la IA multimodal son que puede interpretar imágenes y texto al mismo tiempo, y permite la comunicación con texto, voz e imágenes. La IA multimodal representa un gran avance en la inteligencia artificial, ya que permite una comprensión más parecida a la humana de la información. En la Tabla 6 pueden verse algunos ejemplos de IA multimodal.

Aplicaciones de la IA multimodal

Con la IA multimodal se abre un amplio abanico de posibilidades en sectores como el de la salud, la automoción, la educación o en cuestiones como la accesibilidad. A continuación, desgranamos algunas de sus aplicaciones en estos ámbitos:

- Asistencia sanitaria: Permitirá mejorar el diagnóstico y ajustar el tratamiento de cada paciente, a través de la combinación de diferentes datos como imágenes médicas, el historial del paciente o los proporcionados por sensores médicos portátiles (ritmo cardiaco, presión arterial, glucosa, etc.).
- Automoción: Potenciará la seguridad vial mediante la fusión de datos procedentes de cámaras o radares.
- Educación: Analizará textos, vídeos de impartición de clases y exámenes para adaptar los contenidos y hacerlos más atractivos para el alumnado, ofreciendo una experiencia pedagógica personalizada en las aulas.
- Accesibilidad: Permitirá tener una comunicación más natural con la tecnología a las personas con discapacidad, por ejemplo, mediante la traducción del lenguaje oral al escrito o la manipulación de imágenes y vídeos a través de instrucciones habladas, lo que supondrá un gran avance para este colectivo.

2.3 ÉTICA Y PELIGROS DE LA UTILIZACIÓN DE PLN EN MEDICINA

En general, el uso de PLN en Medicina ofrece muchas oportunidades emocionantes, pero también plantea importantes desafíos éticos y de seguridad [10]. Es importante abordar estos desafíos de manera proactiva para garantizar que el uso de ChatGPT en la atención médica sea seguro, justo y beneficioso para los pacientes.

Preguntando a ChatGPT3 [11] cuales son en su consideración esos peligros nos resume así:

1. Privacidad y seguridad de los datos: El uso de ChatGPT en Medicina puede involucrar el manejo de información médica sensible, como historiales médicos, registros de diagnóstico y otras formas de datos de salud privados. Es crucial garantizar que esta información esté protegida adecuadamente y que se cumplan los estándares de privacidad y seguridad de los datos.
2. Sesgo y discriminación: Como modelo entrenado en grandes cantidades de datos, ChatGPT puede reflejar y amplificar los prejuicios y estereotipos presentes en los datos de entrenamiento. Esto puede llevar a respuestas inexactas, incompletas o incluso perjudiciales para ciertos grupos de pacientes. Es importante garantizar que ChatGPT sea entrenado y utilizado de manera justa y equitativa, sin perpetuar o amplificar la discriminación existente.
3. Responsabilidad y responsabilidad legal: Cuando se usa ChatGPT en el diagnóstico o tratamiento de pacientes, es importante que los médicos y otros profesionales de la salud sean responsables de las decisiones y acciones que toman en función de la información proporcionada por el modelo. Esto puede plantear cuestiones legales y de responsabilidad en caso de que las decisiones médicas resulten en daño o perjuicio para los pacientes.
4. Confianza y comprensión: Los pacientes pueden confiar demasiado en las respuestas proporcionadas por ChatGPT, sin comprender completamente la base de las recomendaciones y el proceso de toma de decisiones detrás de ellas. Es importante garantizar que los pacientes comprendan adecuadamente la información proporcionada por ChatGPT y que se fomente una relación sólida y clara entre el paciente y el médico.
5. Falta de regulación y estándares claros también es una preocupación importante en relación con el uso de ChatGPT en medicina. En muchos casos, no hay pautas o regulaciones específicas que se apliquen al uso de modelos de lenguaje en la atención médica, lo que puede dar lugar a prácticas inconsistentes y potencialmente peligrosas.

La falta de regulación también puede dificultar la evaluación y comparación de diferentes modelos de lenguaje, lo que hace difícil determinar cuáles son los modelos más precisos, confiables y seguros para su uso en la atención médica. Además, los estándares y regulaciones claros también son importantes para garantizar que los modelos de lenguaje se desarrollen y utilicen de manera ética y responsable.
Por lo tanto, es importante que los reguladores y las autoridades de salud trabajen para establecer pautas claras y estándares de calidad para el uso de modelos de lenguaje en la atención médica. Esto puede incluir el establecimiento de estándares para la recopilación y uso de datos, la evaluación de la precisión y seguridad de los modelos de lenguaje, y la implementación de prácticas y políticas éticas para garantizar que los modelos se utilicen de manera justa y equitativa.

2.4 PLATAFORMAS Y WEBS ESPECIALIZADAS EN IA MÉDICA

En los últimos años, el escenario de la Inteligencia Artificial Médica ha crecido exponencialmente, abarcando desde el análisis de imágenes y la documentación clínica automatizada, hasta la investigación farmacéutica y la personalización de tratamientos. A continuación, se presenta una tabla con una selección de plataformas y webs especializadas en IA médica. Estas herramientas ilustran la diversidad de enfoques actuales —clínicos, operativos, de investigación y tecnológicos— que impulsan la transformación digital en la práctica sanitaria.

Tabla 7.

NOMBRE / WEB / PLATA-FORMA	TIPO / ENFOQUE PRINCIPAL	DESCRIPCIÓN BREVE	NOTAS / PARTICU-LARIDADES
Viz.ai	Coordinación clínica / imagen médica	Analiza imágenes médicas con algoritmos de IA para alertas y optimizar flujos clínicos (https://www.viz.ai)	Cuenta con más de 50 algoritmos aprobados en EE.UU.
Heidi Health	Scribe médico automatizado	Convierte las consultas médico–paciente en notas clínicas estructuradas mediante IA (https://www.heidihealth.com)	Se integra con sistemas de registros médicos (EHR).
Aidoc	Análisis de imagen radiológica	Plataforma de IA para priorizar diagnósticos urgentes en radiología (https://www.aidoc.com)	Muy usada para detectar ACV, hemorragias y embolias.
MONAI	Framework / código abierto	Librerías de IA para imágenes médicas: segmentación, clasificación, registro (https://monai.io)	Usado en investigación e implementación clínica.
OpenEvidence	Soporte de decisión clínica / búsqueda médica	Motor de búsqueda médica con IA que cita publicaciones científicas (https://openevidence.com)	Incluye DeepConsult para casos clínicos complejos.
Notable Health	Automatización de operaciones sanitarias	Personaliza flujos de trabajo clínicos y administrativos con IA (https://www.notablehealth.com)	Optimiza tareas administrativas repetitivas.
IBM Watson Health	IA en salud amplia	Procesa datos clínicos con IA y NLP para apoyo diagnóstico (https://www.ibm.com/watson-health)	Proyectos en oncología, imagen médica, etc.
PathAI	Patología digital / diagnóstico	Aprendizaje profundo para interpretar muestras patológicas (https://www.pathai.com)	Reduce errores humanos en anatomía patológica.
Keragon	Automatización / operaciones en salud	IA para automatizar procesos clínicos y administrativos (https://www.keragon.com)	Integra sistemas clínicos y administrativos (EHR, CRM).
Aiddison	Descubrimiento de fármacos	IA para identificar moléculas candidatas a fármacos (https://www.aiddison.com)	Aplicada a investigación farmacéutica.
BioMorph	Análisis predictivo molecular	Predice efectos de compuestos mediante modelos de IA (https://www.biomorph.ai)	Útil en etapas tempranas de I+D farmacéutico.

(continúa)

(continuación)

NOMBRE / WEB / PLATA-FORMA	TIPO / ENFOQUE PRINCIPAL	DESCRIPCIÓN BREVE	NOTAS / PARTICU-LARIDADES
Microsoft Dragon Copilot	Asistente clínico / documentación	Genera notas clínicas a partir de conversaciones médico–paciente (https://www.microsoft.com)	Evolución del sistema DAX de Nuance.
Tempus	Datos clínicos / medicina personalizada	Integra datos genómicos y clínicos para terapias personalizadas (https://www.tempus.com)	Usado en oncología y ensayos clínicos.
Butterfly iQ	Dispositivo / hardware + IA	Ultrasonido portátil con guía de IA para análisis (https://www.butterflynetwork.com)	Hace accesible la imagen médica en entornos con pocos recursos.
SenseCare	Investigación / imágenes 3D	Plataforma con visualización 3D, segmentación y registro de imágenes (https://arxiv.org/abs/2004.07031)	Facilita investigación y planificación quirúrgica.
MAIA	Plataforma colaborativa de IA médica	Gestión de datos, modelos y despliegue de IA médica (https://arxiv.org/abs/2507.19489)	Promueve interoperabilidad médico–tecnológica.

Referencias

1 De Francisco ALM. Artificial. La Nueva Inteligencia y su Aplicación en Nefrología. En Lorenzo V, López Gómez JM (Eds). Nefrología al Día. ISBN 978-84-127624-1-9.

2 Szeliski R. *Computer Vision: Algorithms and Applications* [Internet]. 2nd ed. Cham: Springer; 2022. Disponible en: http://szeliski.org/Book

3 Goodfellow I, Bengio Y, Courville A. *Deep Learning* [Internet]. Cambridge, MA: MIT Press; 2016 [citado 2025 Jun 27]. Disponible en: https://www.deeplearningbook.org

4 Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, Thrun S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*. 2017 Feb 2;542(7639):115-8. DOI:10.1038/nature21056.

5 Rajpurkar P, Irvin J, Zhu K, Yang B, Mehta H, Duan T, et al. CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep learning. *arXiv* [preprint]. 2017. Disponible en : https://arxiv.org/abs/1711.05225

6 Chowdhury GG. Natural language processing. *Annu Rev Inf Sci Technol*. 2003;37(1):51-89. DOI:10.1002/aris.1440370103.

7 Spyridis Y, Younes JP, Deeb H, Argyriou V. Empowering prior to court legal analysis: A transparent and accessible dataset for defensive statement classification and interpretation. *arXiv* [Preprint]. 2024 May 17 [cited 2025 Aug 10]; Available from: arXiv:2405.10702

8 Levin S, Toerper M, Hamrock E, Hinson JS, Barnes S, Gardner H, et al. Machine-learning-based electronic triaje more accurately differentiates patients with respect to clinical outcomes compared with the emergency severity index. *Ann Emerg Med*. 2018;71(5):565-74. DOI:10.1016/j.annemergmed.2017.08.005.

9 Van Dis EAM, Bollen J, van Rooij R, Zuidema W, Bockting CL. ChatGPT: five priorities for research. *Nature*. 2023 Feb 9;614(7947):224-6. DOI:10.1038/d41586-023-00288-7.

10 Fernandes AC, Souto MEVC. Benefits, limits, and risks of GPT-4 as an AI chatbot for medicine. *N Engl J Med*. 2023 Jun 22;388(25):2399-400. DOI:10.1056/NEJMp2303819.

11 OpenAI. ChatGPT: desafíos éticos y de seguridad del ChatGPT-3 [Internet]. San Francisco (CA): OpenAI; [cited 2025 Aug 8]. Available from: https://chat.openai.com/