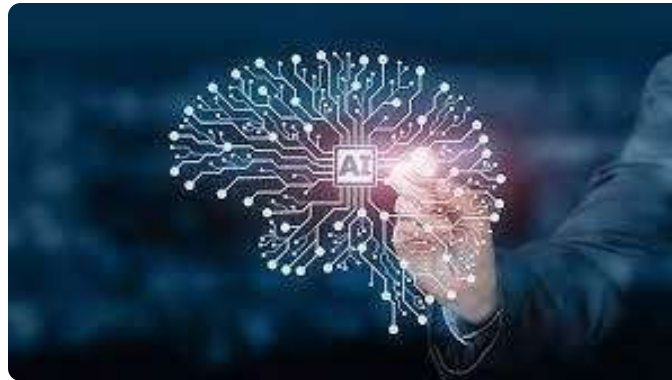


Introducción a la inteligencia artificial

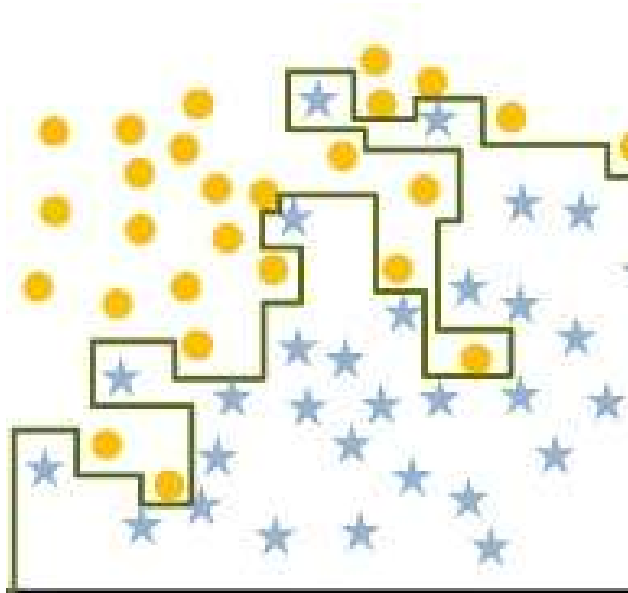
Qué es la IA? La IA es un campo de la informática que busca crear sistemas capaces de **simular capacidades cognitivas humanas**: percibir su entorno, interpretar datos, aprender, razonar, tomar decisiones, adaptarse a nuevas situaciones o resolver problemas complejos.

¿Qué no es IA? No basta con automatizar tareas o ejecutar reglas fijas. Para considerarse IA, un sistema debe ser capaz de **aprender, generalizar o adaptarse ante cambios**, no solo repetir pasos preprogramados.



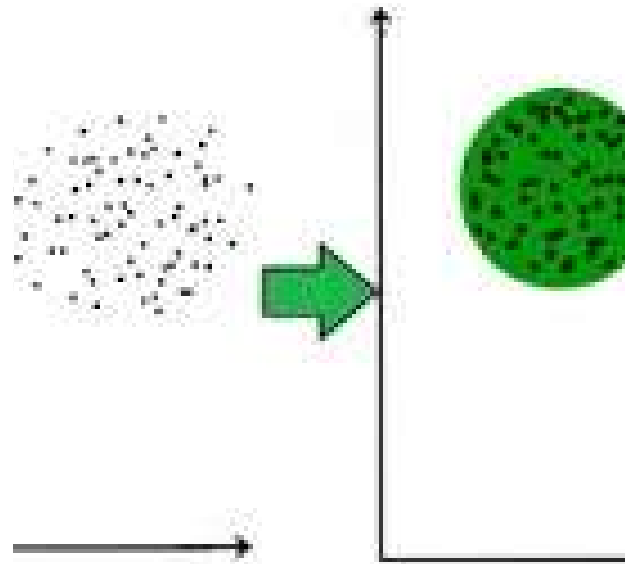
Machine Learning (ML): ML es un subconjunto de la IA. En lugar de programar cada regla, los algoritmos de ML **aprenden patrones a partir de datos** y usan ese aprendizaje para predecir, clasificar o generar comportamientos.

Tipos de Aprendizaje Automático



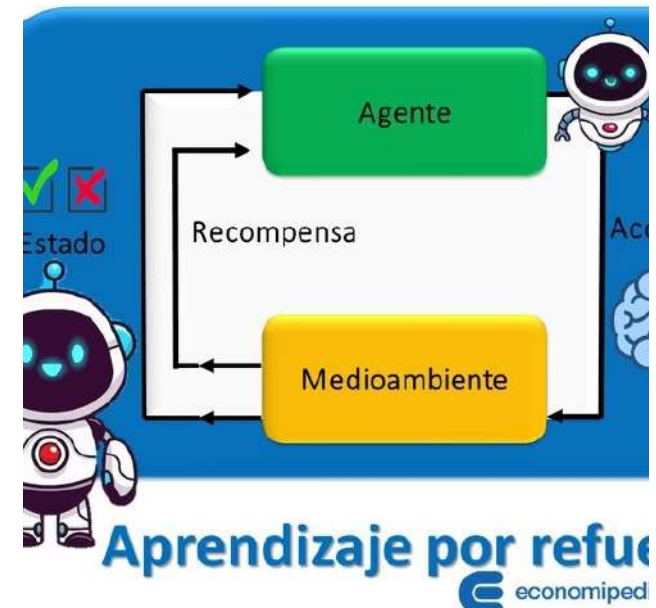
Aprendizaje Supervisado: Imagina que tienes muchas frutas ya identificadas: "esta es una manzana", "esta es una naranja", "esta es una pera". El algoritmo aprende con esos ejemplos — entradas + respuestas correctas — y luego eres capaz de clasificar frutas nuevas que nunca habías visto.

Uso típico: clasificación, regresión, predicción de valores cuando tienes datos etiquetados.



Aprendizaje No Supervisado: Tienes muchos juguetes sin etiquetar, sin que nadie te diga cómo agruparlos. Entonces decides organizarlos tú mismo según sus características: color, tamaño, forma, tipo. El algoritmo hace lo mismo: agrupa datos similares, descubre patrones o estructura, sin etiquetas previas.

Uso típico: detección de patrones, clustering, segmentación, reducción de dimensionalidad.



Aprendizaje por Refuerzo: Eres un jugador que aprende a jugar un videojuego: al principio no sabes qué hacer, pero aprendes — probando acciones — que algunas te dan puntos o recompensas, otras te hacen perder vidas. Con el tiempo, desarrollas una estrategia que maximiza tu puntaje. Así actúa un agente de IA: interactúa con un entorno, recibe recompensas o castigos, y aprende a tomar decisiones óptimas.

Uso típico: optimización secuencial, control, robótica, juegos, decisiones en entornos dinámicos.

Aprendizaje no Supervisado

Un enfoque de machine learning en el que el modelo trabaja con **datos sin etiquetas**, es decir, sin una “respuesta correcta” predefinida. Su objetivo es **descubrir patrones, relaciones o estructuras ocultas** en los datos, sin intervención externa.

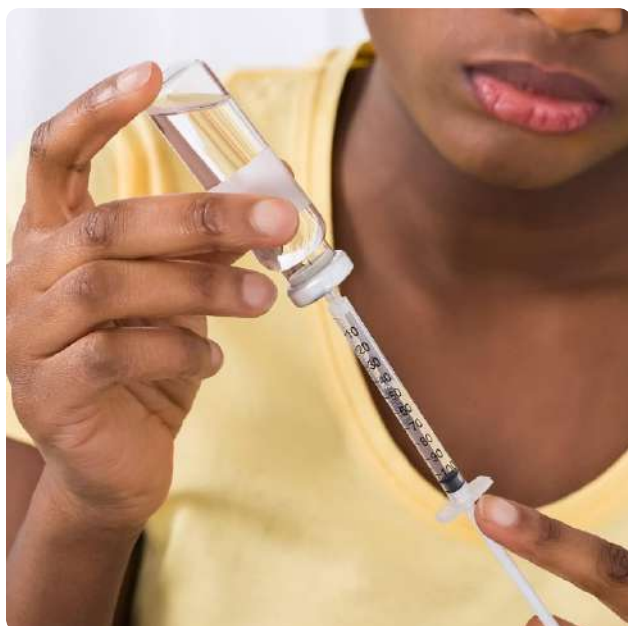
Principales técnicas / tareas

- **Clustering / Agrupamiento** — agrupa los datos en clusters basados en similitudes, útil para segmentación, análisis exploratorio, detección de grupos naturales.
- **Reducción de dimensionalidad** — transforma datos complejos y de muchas variables en representaciones más simples y manejables, preservando la estructura esencial. Útil para visualización, compresión de datos y preparación para otros métodos.
- **Detección de anomalías / descubrimiento de patrones** — identificar registros atípicos, relaciones inesperadas, estructuras ocultas o outliers.

¿Para qué sirve?

- Explorar datos sin información previa
- Segmentar clientes, agrupar comportamientos, descubrir perfiles
- Preparar datos para un análisis más profundo
- Detectar anomalías o patrones invisibles a simple vista
- Reducir dimensionalidad para facilitar visualización o procesamiento

Ejemplos de aprendizaje no supervisado



Salud: Identificación de patrones sin diagnóstico previo

El aprendizaje no supervisado permite agrupar pacientes según características clínicas similares sin necesidad de etiquetas o diagnósticos previos.

Esto permite descubrir:

- grupos de pacientes que responden de forma similar a un tratamiento,
- patrones comunes en síntomas,
- perfiles basados en resultados de laboratorio o historial médico.

Es especialmente útil para **descubrir fenotipos ocultos** o **subpoblaciones clínicas**



Industria y mantenimiento: Detección de anomalías

En sistemas industriales, flotas o maquinaria, el aprendizaje no supervisado analiza señales y sensores para identificar **comportamientos atípicos** respecto al funcionamiento normal.

Esto permite:

- detectar fallas incipientes,
- anticipar mantenimiento,
- evitar paradas inesperadas,
- identificar desgaste o condiciones anormales.

Es clave para **mantenimiento predictivo** cuando no existe una etiqueta previa que diga "esto es una falla".



Retail y marketing: Segmentación de clientes

El aprendizaje no supervisado permite agrupar clientes según sus hábitos y preferencias, sin tener categorías definidas previamente.

Puede identificar:

- clientes con patrones de compra similares,
 - perfiles que responden igual a promociones,
 - comportamientos recurrentes según horarios o productos.
- Esto habilita **segmentación dinámica**, mejor personalización y estrategias de marketing más precisas.

Tipos de algoritmos de aprendizaje no supervisado

K-means (K-medias): Algoritmo de **agrupamiento basado en proximidad**.

Divide los datos en k grupos asignando cada punto al "centroide" más cercano, y ajustando esos centroides iterativamente.

Es rápido, eficiente y muy usado cuando se conoce o se puede estimar el número de clusters.

Cuándo usarlo:

- Datos bien distribuidos
- Clusters más o menos esféricos
- Gran volumen de datos

Agrupamiento Jerárquico: Construye una **jerarquía de clusters** al fusionar gradualmente los puntos o grupos más cercanos entre sí.

El resultado se suele visualizar con un **dendrograma**, que muestra cómo se forman los grupos a diferentes niveles.

Cuándo usarlo:

- Cuando no sabes el número de clusters
- Para explorar estructura interna
- Para análisis exploratorio detallado

DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*): Algoritmo de **agrupamiento basado en densidad**.

Detecta regiones densas de puntos y separa áreas dispersas o aisladas, lo que lo hace ideal para identificar **outliers** de forma natural.

Ventajas clave:

- No requiere especificar k
- Identifica formas arbitrarias de clusters
- Maneja bien ruido y anomalías

PCA (Análisis de Componentes Principales): Técnica de **reducción de dimensionalidad** que transforma las variables originales en nuevos ejes llamados **componentes principales**, que capturan la mayor variabilidad posible de los datos.

Para qué sirve:

- Visualizar datos complejos en 2D o 3D
- Reducir ruido
- Acelerar modelos posteriores
- Comprender la estructura subyacente

Aprendizaje Supervisado

Qué es el aprendizaje supervisado?

Es un tipo de aprendizaje automático en el que el modelo aprende a partir de un **conjunto de datos etiquetados**. Cada ejemplo del entrenamiento incluye:

- **Entrada (X):** los datos que el modelo recibe
- **Etiqueta o respuesta correcta (y):** el resultado esperado

El objetivo del modelo es **aprender la relación** entre $X \rightarrow y$ para poder predecir correctamente la etiqueta de datos nuevos que nunca ha visto.

Tipos de aprendizaje supervisado

- **Clasificación:** El modelo aprende a asignar cada dato a una **categoría**.

Ejemplos típicos:

- clasificar imágenes como *gato* o *perro*,
- identificar si un correo es *spam* o *no spam*,
- detectar si un cliente cancelará o no su servicio.

→ La salida es **discreta** o **categorica**.

- **Regresión:** El modelo aprende a predecir un **valor numérico continuo**.

Ejemplos comunes:

- predecir el precio de una casa,
- estimar la demanda de un producto,
- calcular la temperatura esperada mañana.

→ La salida es **numérica y continua**.

Ejemplos de aprendizaje supervisado



Diagnóstico médico asistido por IA:

El aprendizaje supervisado puede analizar **imágenes médicas etiquetadas por expertos** (radiografías, TAC, resonancias) para clasificar si una imagen es **normal** o muestra señales de una **patología específica**.

Esto permite apoyar el diagnóstico, acelerar tiempos de análisis y reducir errores.



Mantenimiento predictivo: Los datos históricos de **sensores, vibración, temperatura, presión, ciclos de uso** combinados con los registros de **fallos reales ocurridos** permiten entrenar modelos capaces de **predecir fallas futuras**.

Esto habilita un mantenimiento **proactivo**, reduce tiempos de inactividad y evita reparaciones costosas.



Detección de fraude financiero: A partir de transacciones históricas **etiquetadas como legítimas o fraudulentas**, un modelo aprende patrones anómalos de comportamiento. Luego puede clasificar nuevas transacciones en tiempo real como **normales o sospechosas**, mejorando la seguridad y reduciendo pérdidas.

Tipos de algoritmos de aprendizaje supervisado

Máquinas de Vectores de Soporte (SVM): Algoritmos potentes para **clasificación** (y también regresión). Buscan encontrar el **hiperplano óptimo** que separa las clases con el mayor margen posible. Funcionan muy bien cuando las clases son separables y en espacios de alta dimensión.

Random Forest : Método basado en **muchos árboles de decisión**. Cada árbol se entrena con una muestra aleatoria de datos y características (bagging), y la predicción final se obtiene mediante **votación** (clasificación) o **promedio** (regresión). Es robusto, reduce el sobreajuste y funciona bien sin mucho ajuste de parámetros.

Redes Neuronales: Modelos inspirados en la estructura del cerebro humano, formados por “neuronas” interconectadas que transforman y aprenden representaciones complejas de los datos. Se emplean para tareas donde los patrones no son lineales y requieren capacidad de abstracción.

Gradient Boosting : Técnica que combina múltiples árboles, pero **de forma secuencial y correctiva**. Cada árbol nuevo aprende a corregir los errores de los anteriores minimizando el **gradiente de la pérdida**. Es uno de los métodos de mayor rendimiento en datos estructurados.

Aprendizaje por reforzamiento

El aprendizaje por refuerzo funciona **igual que cuando una persona aprende jugando un videojuego**:

- El agente (la IA) **prueba cosas**.
- Si hace algo que está bien → **gana puntos o "recompensa"**.
- Si hace algo mal → **pierde puntos o recibe "castigo"**.
- Con el tiempo, va aprendiendo **qué acciones le convienen** para obtener más recompensas.

Tipos principales

- **Aprendizaje por Valor:** El agente aprende **qué tan bueno es estar en cada situación**. Piensa: *"Si estoy aquí, ¿qué tan bueno es? ¿Qué tanto puedo ganar desde este estado?"* Luego elige la acción que lo lleve al mejor estado posible.
- **Aprendizaje por Política:** En lugar de evaluar estados, el agente aprende **directamente qué acción tomar en cada situación**. Una "política" es solo una **regla simple**: *"Si estoy en esta situación, hago esta acción."*

Aprendizaje por Refuerzo Basado en Valor

1. **¿Quién es el agente?** Steve, tu personaje en Minecraft. Él es quien toma decisiones.
2. **¿Qué es "el valor"?** Steve intenta aprender **qué tan buena o mala** es cada situación o lugar del juego. Ejemplo:
 - Un lugar con diamantes = **alto valor**
 - Un lugar lleno de zombies = **bajo valor**
3. **¿Qué hace con ese valor?** Steve usa esas estimaciones para **decidir qué hacer**:
 - Ir hacia donde hay **alto valor**
 - Alejarse de donde hay **bajo valor**
4. **¿Cuál es la meta?** Que Steve aprenda, con la experiencia, a:
 - **Calcular mejor el valor de cada situación**
 - **Elegir acciones** que le den más recompensas (recursos, puntos, supervivencia)
5. Aprendizaje por refuerzo basado en valor es cuando Steve aprende qué lugares o situaciones "valen la pena", y toma decisiones siguiendo esos valores.

Aprendizaje por Refuerzo por politica

1. **¿Quién es el agente?** Steve, tu personaje en Minecraft. Él es quien toma decisiones.
2. **¿Qué es una política?** Es como un **manual de instrucciones** que le dice a Steve qué acción tomar según lo que ve.
Ejemplo:
 - *"Si veo un zombie, corro."*
 - *"Si veo carbón, lo recojo."*
3. **¿Cómo aprende Steve?** Con la experiencia, Steve **mejora ese manual**:
 - *"Si veo un zombie, mejor busco refugio o me armo antes."*
 - *"Si es de noche, evito zonas peligrosas."*
 - Cada vez que prueba algo y le va bien, **actualiza su política** para hacerlo de nuevo.
Si le va mal, deja de hacerlo.
4. **¿Cuál es la meta?** Encontrar la **mejor política posible**: la lista de reglas que le permite **sobrevivir más, recolectar más recursos y obtener más recompensas**.
5. *Aprendizaje por Refuerzo*: Aprendizaje basado en política es cuando Steve aprende qué acción tomar en cada situación usando un conjunto de reglas que va mejorando con la experiencia.

Ejemplos de aprendizaje por reforzamiento



Juegos: El aprendizaje por refuerzo sirve para entrenar agentes que juegan **ajedrez, Go, póker o videojuegos**.

¿Cómo funciona?

El agente prueba jugadas, gana puntos cuando juega bien y pierde puntos cuando juega mal.

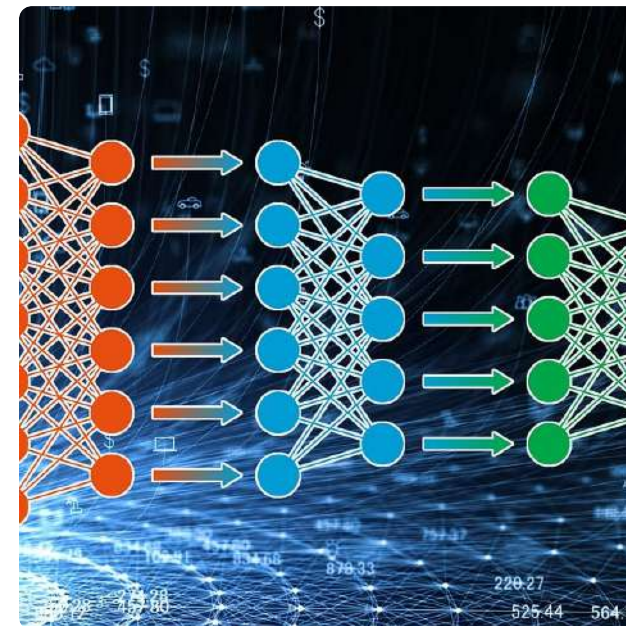
Con el tiempo aprende **qué jugadas lo acercan a ganar**.



Robótica: Los robots también aprenden por prueba y error:

- caminar sin caerse,
- agarrar objetos,
- moverse por lugares desconocidos.

Reciben **recompensas** cuando hacen algo bien (avanzar, no caerse, coger el objeto) y **castigos** cuando lo hacen mal.

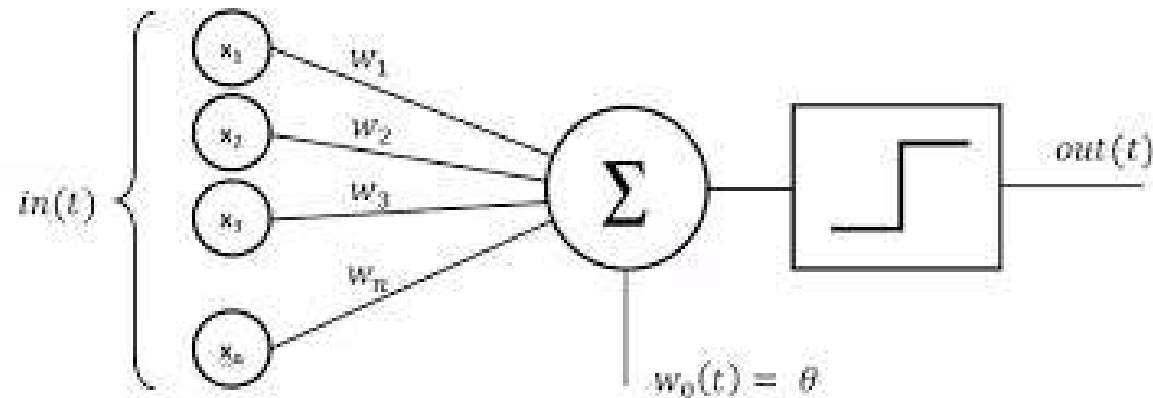


Grandes Modelos de Lenguaje (LLM) como ChatGPT

Los modelos como ChatGPT también usan aprendizaje por refuerzo para mejorar.

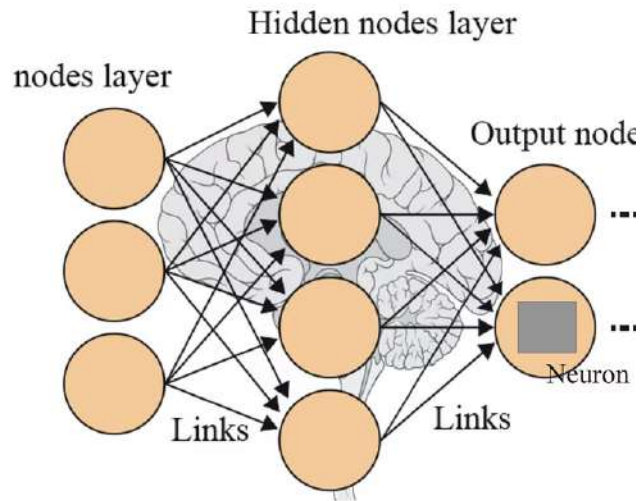
Después de entrenarse con muchos textos, se les enseña a **responder mejor** mediante un proceso llamado *Reinforcement Learning from Human Feedback* (RLHF).

Deep learning

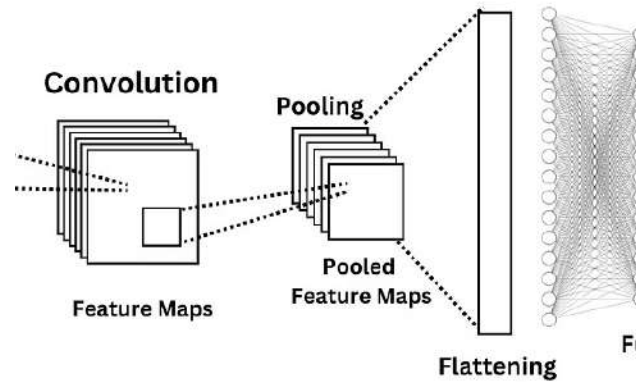


1. ¿Qué es una red neuronal? Una red neuronal es un modelo de aprendizaje automático **inspirado en cómo funciona el cerebro humano**. Está formada por muchas "neuronas" artificiales conectadas entre sí, que trabajan juntas para aprender patrones complejos a partir de los datos.
2. ¿Qué es una neurona artificial?
 - Recibe varias **entradas**
 - Cada entrada tiene un **peso** (qué tan importante es)
 - La neurona **suma** esas entradas
 - Luego pasa el resultado por una **función de activación**, que decide qué salida producir
3. ¿Qué es "Deep" en Deep Learning? Significa que la red tiene **muchas capas de neuronas**, lo que le permite aprender patrones cada vez más complejos, como reconocer voces, imágenes o lenguaje.

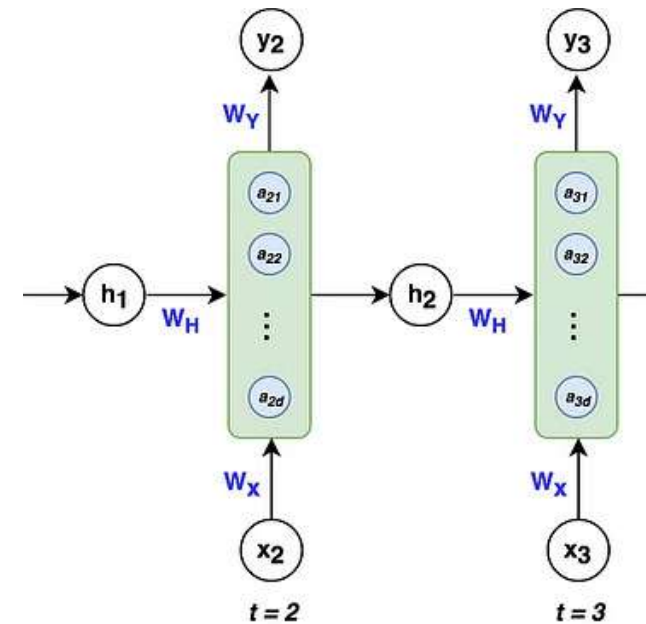
Tipos de Arquitecturas de redes neuronales



En un MLP, tienes múltiples capas de neuronas. Las neuronas en la primera capa reciben los datos de entrada (como los píxeles de una imagen), cada una de estas neuronas procesa una parte de la información y pasa los resultados a las neuronas de la siguiente capa. Este proceso se repite a través de todas las capas de la red.



Imagina que tienes una gran pila de fotos y quieres encontrar todas las fotos que contienen un gato. Un humano tendría que revisar cada foto una por una, pero una CNN puede hacer este trabajo mucho más rápido. Esto se debe a que una CNN tiene una estructura especializada que la hace realmente buena para tareas relacionadas con imágenes.



Estas son útiles cuando la información tiene un orden o secuencia. Por ejemplo, para entender este mensaje, necesitas leer las palabras en orden. No puedes simplemente mezclar todas las palabras y esperar entenderlo. Una RNN es buena para estas tareas porque puede recordar la información anterior y utilizarla para ayudar a procesar la siguiente pieza de información.

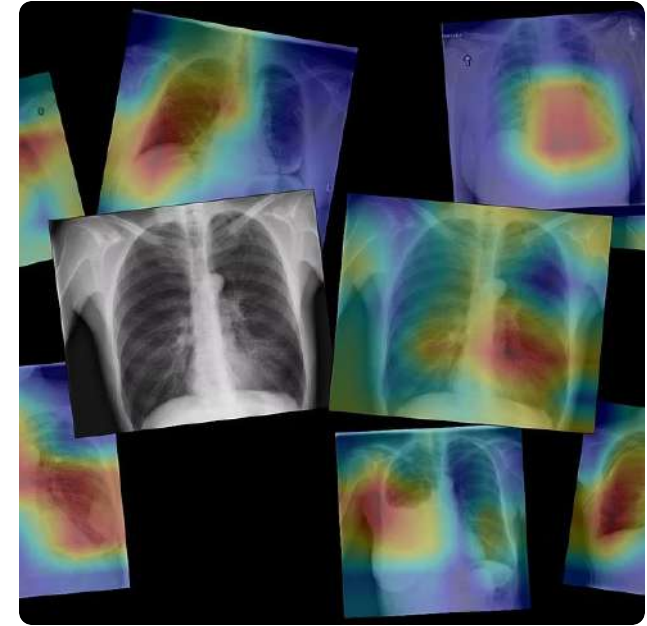
Aplicacion de tecnicas de deep learning



Asistentes Virtuales: Empresas como Amazon, Google y Apple utilizan IA para potenciar sus asistentes virtuales (Alexa, Google Assistant y Siri, respectivamente). Estos asistentes utilizan procesamiento del lenguaje natural y aprendizaje automático para entender y responder a las solicitudes del usuario en lenguaje natural.



Vehículos Autónomos: Compañías como Tesla y Waymo utilizan IA y aprendizaje automático para desarrollar coches autónomos. Estos vehículos usan una combinación de sensores y algoritmos de aprendizaje automático para interpretar su entorno y tomar decisiones de conducción.



Diagnóstico Médico Automatizado: Se utilizan redes neuronales CNN para analizar imágenes médicas, como radiografías, tomografías computarizadas y resonancias magnéticas. Estos modelos de aprendizaje profundo pueden aprender a identificar anomalías, como tumores o fracturas, a menudo con una precisión que rivaliza o incluso supera a la de los médicos humanos.

¿Qué es MLOps y por qué es clave?

MLOps: Operacionalización del Machine Learning

MLOps es la práctica que combina **Machine Learning**, **DevOps** y **gestión de datos** para llevar modelos de IA desde el desarrollo hasta la operación real dentro de una organización.

¿Por qué existe MLOps?

Porque entrenar un modelo hoy en día no es lo más difícil del proceso.

lo difícil es que funcione bien en producción, de forma estable, escalable y confiable.

¿Qué resuelve MLOps?

- **Automatización del ciclo de vida del modelo** (entrenamiento, validación, despliegue).
- **Versionamiento** de datos, código y modelos.
- **Monitoreo en producción:** deriva de datos, deriva del modelo, degradación.
- **Escalabilidad** para atender miles/millones de predicciones.
- **Reentrenamiento continuo** cuando cambian los datos.
- **Trazabilidad y gobernanza** para auditar decisiones automáticas.

El Ingeniero de Machine Learning y su Impacto en la Organización

¿Qué hace un Ingeniero de Machine Learning?

Es el profesional encargado de **construir, escalar y mantener** los sistemas de ML en producción.

No solo entrena modelos → habilita que la organización saque valor real de ellos.

Responsabilidades principales

- Diseñar pipelines de datos y entrenamiento robustos.
- Implementar modelos en producción (APIs, microservicios, batch, streaming).
- Optimizar rendimiento y latencia de inferencias.
- Monitorear el comportamiento del modelo en el mundo real.
- Automatizar reentrenamientos y actualizaciones.
- Garantizar seguridad, compliance y gobernanza del ML.

¿Por qué es tan importante dentro de la empresa?

- **Convierte prototipos en productos reales**, listos para ser usados por clientes o áreas internas.
- Asegura que los modelos sean **escalables, seguros y mantenibles**.
- Reduce riesgos: modelos que fallan, decisiones erróneas, costos operativos.
- Permite que la IA sea un **activo estratégico**, no un experimento.
- Conecta ciencia de datos con negocio, TI, operaciones y producto.

Un proyecto de ciencia de datos

Etapas de un Problema de Machine Learning

