

## 7. Conjunto de datos o datasets.

Todo proceso de aprendizaje automático parte de un **conjunto de datos (dataset)** que describe un fenómeno observable.

Un *dataset* es una **colección de observaciones** (también llamadas **instancias** o **ejemplos**), en las que cada observación se representa mediante un conjunto de **atributos o variables**.

- **Observaciones** → cada fila representa un caso concreto (ejemplo: un paciente, una transacción, una imagen).
- **Atributos o variables** → cada columna describe una característica de la observación (ejemplo: presión arterial, edad, colesterol).
- **Valores faltantes (missing values)** → algunos atributos pueden no tener valor registrado para ciertas observaciones, ya sea porque no se conoce o porque no se ha podido medir.

### Ejemplo

Un *dataset* sobre **enfermedades del corazón** podría contener miles de registros de pacientes. Cada observación incluiría valores de atributos como:

- Presión arterial
- Nivel de colesterol
- Peso y altura
- Sexo
- Pulsaciones en reposo
- Práctica de actividad física

Además, para que el modelo pueda **aprender a predecir** la presencia de enfermedades del corazón, el *dataset* debería incluir una variable adicional denominada **etiqueta o variable objetivo** (target), que indique si cada paciente ha padecido o no una enfermedad cardíaca.

- **Aprendizaje supervisado:** si se dispone de esa etiqueta, el modelo puede entrenarse para predecir el riesgo de enfermedad en nuevos pacientes.
- **Aprendizaje no supervisado:** si no se dispone de la etiqueta, solo sería posible **agrupar** a los pacientes en clústeres según sus similitudes, pero no predecir la probabilidad de enfermedad.

## Formato y organización

Un *dataset* suele organizarse de manera muy similar a una **tabla en una base de datos relacional** o a un fichero de texto en **formato CSV** (*Comma-Separated Values*):

- Cada **fila** → una observación.
- Cada **columna** → un atributo o variable.
- La **primera fila** en un CSV suele contener los nombres de los atributos.

Ejemplo simplificado en CSV:

```
ID,Edad,Sexo,Colesterol,Presión_Arterial,Actividad_Física,Enfermedad_Corazón
1,55,Hombre,240,140,Sí,Sí
2,47,Mujer,200,120,No,No
3,63,Hombre,180,150,Sí,Sí
```

En conclusión: el *dataset* es la **base fundamental del aprendizaje automático**, ya que sin datos de calidad, completos y representativos, ningún modelo de IA puede entrenarse de manera efectiva.

## 8. Modelo.

Un **modelo de inteligencia artificial (IA)** es una **representación matemática o computacional** que se entrena para llevar a cabo tareas específicas, como **reconocer patrones, hacer predicciones o tomar decisiones**, a partir de datos de entrada.

Los modelos de IA se construyen mediante **algoritmos y técnicas matemáticas**, con el objetivo de **aprender de los datos** y mejorar progresivamente su rendimiento en la tarea para la que fueron diseñados.

### Relación entre modelo y algoritmo

En el ámbito del *Machine Learning*, los términos **modelo** y **algoritmo** suelen usarse de manera cercana, aunque no son exactamente lo mismo:

- Un **modelo** es la **función matemática aprendida** que asocia a cada valor de entrada  $x$  una salida estimada  $\hat{y}=f(x)$ .

$$\hat{y}=f(x)$$

- El **algoritmo** es el **procedimiento de cálculo** que permite construir y ajustar el modelo, es decir, encontrar la mejor forma de la función  $f$  a partir de los datos.

Dicho de otro modo: el modelo es el **resultado entrenado** que realiza predicciones, mientras que el algoritmo es el **método utilizado para entrenarlo**.



Cuando trabajamos con técnicas de **Machine Learning** o **Deep Learning**, el punto de partida es un **conjunto de datos de entrenamiento**. Estos datos representan experiencias pasadas y sirven para que el algoritmo de aprendizaje —ya sea un modelo clásico de *machine learning* o una red neuronal— pueda **extraer patrones y construir una representación interna** del problema.

Este aprendizaje se lleva a cabo **ajustando una función matemática** que depende de una serie de **parámetros internos**. Dichos parámetros se van modificando de manera progresiva hasta que la función es capaz de realizar **predicciones precisas**.

A esa función matemática con parámetros ajustables la llamamos **función hipótesis**. El entrenamiento consiste, por tanto, en **modificar los valores de los parámetros** de la función hipótesis para que encaje lo mejor posible con los datos de entrenamiento.

Cuando proporcionamos los datos al algoritmo, este inicia un **proceso de optimización**, en el que compara las predicciones con los valores reales y corrige los parámetros para reducir los errores. De este modo, el algoritmo “aprende” de la experiencia pasada.

Al finalizar el entrenamiento, obtenemos una **función hipótesis ajustada**, es decir, un **modelo entrenado**. Este modelo es capaz de **generalizar a datos nuevos**: puede recibir ejemplos que no estaban en el conjunto de entrenamiento y realizar predicciones sobre ellos.



## Implementación práctica

El algoritmo se implementa mediante un **programa escrito en un lenguaje de programación** (como Python, R o Java), utilizando librerías y entornos de *machine learning* (ej. TensorFlow, PyTorch, Scikit-learn).

En resumen:

- **El modelo es la función entrenada** que transforma entradas en salidas.
- **El algoritmo es el proceso** que permite entrenar esa función a partir de los datos.

### Pseudocódigo de entrenamiento:

```
datos = cargar_dataset()
modelo = inicializar_modelo()
mientras no coincide:
    predicciones = modelo.predecir(datos.entradas)
    error = calcular_error(predicciones, datos.salidas)
    modelo.ajustar_parametros(error)
guardar(modelo)
```

## Pasos clave en el entrenamiento de un modelo de IA

1. **Datos de entrenamiento**
  - El modelo recibe datos **etiquetados** (en aprendizaje supervisado) o **no etiquetados** (en aprendizaje no supervisado).
  - A partir de ellos, identifica **patrones, relaciones y estructuras**.
2. **Algoritmo de aprendizaje**
  - Es el método matemático que ajusta los parámetros internos del modelo.
  - Ejemplo: un algoritmo de clasificación aprende a distinguir entre “gato” y “perro” corrigiendo sus parámetros para reducir errores.
3. **Función de pérdida (o función objetivo)**
  - Calcula la diferencia entre las predicciones del modelo y los resultados reales.
  - El objetivo es **minimizar esta pérdida**, acercando las predicciones del modelo a la realidad.
4. **Optimización**
  - El algoritmo ajusta los parámetros internos (por ejemplo, los pesos en una red neuronal) usando técnicas como **gradiente descendente**.
  - Con cada iteración, el modelo mejora su rendimiento.

**Ejemplo:** Un modelo entrenado para **clasificar imágenes** puede recibir como entrada una foto y determinar si contiene un “gato” o un “perro”. Durante el entrenamiento, el modelo ajusta sus parámetros comparando sus predicciones con las etiquetas reales. Una vez entrenado, será capaz de analizar imágenes nuevas y realizar predicciones de forma autónoma.

En conclusión: **un modelo de IA se construye y entrena mediante un proceso de aprendizaje**, en el que los algoritmos ajustan los parámetros de una función hipótesis hasta que el modelo puede generalizar a nuevas situaciones y realizar predicciones útiles.

#### A. Similitudes con un algoritmo matemático:

Un **modelo de inteligencia artificial (IA)** comparte varias características con un **algoritmo matemático tradicional**, ya que ambos se fundamentan en principios matemáticos y lógicos para resolver problemas. Entre las similitudes más destacadas se encuentran:

1. **Uso de ecuaciones y funciones matemáticas.** Tanto en los algoritmos matemáticos como en los modelos de IA se emplean **funciones y ecuaciones** para representar relaciones entre variables.
  - En los algoritmos matemáticos, estas funciones permiten calcular soluciones exactas a un problema.
  - En IA, las funciones se utilizan para calcular la **función de pérdida**, actualizar los **parámetros internos** y realizar **predicciones** sobre los datos.
2. **Optimización.** Ambos enfoques buscan **encontrar la mejor solución posible** dentro de un espacio de alternativas.
  - En IA, el entrenamiento del modelo consiste en **ajustar parámetros** para minimizar una función de pérdida.
  - En matemáticas, los algoritmos de optimización buscan **minimizar o maximizar** una función objetivo, o resolver un sistema de ecuaciones de manera óptima.
3. **Entradas y salidas.** Tanto un modelo de IA como un algoritmo matemático funcionan como un **proceso de transformación**:
  - Reciben **datos de entrada** (valores iniciales, variables, ejemplos).
  - Procesan esa información según un conjunto de reglas o funciones.
  - Devuelven un **resultado de salida**, ya sea una solución exacta o una predicción estimada.

4. **Estructura lógica.** Ambos siguen una **secuencia de pasos bien definidos** para resolver un problema.

- En los algoritmos matemáticos, estos pasos suelen ser fijos y deterministas.
- En los modelos de IA, la lógica incluye procesos como el **preprocesamiento de datos, ajuste iterativo de parámetros y generación de predicciones**, pero sigue siendo una estructura ordenada y coherente.

**B. Diferencias clave:**

**Capacidad de aprendizaje**

- La diferencia más importante es que un **modelo de IA puede aprender de los datos**. A medida que recibe más ejemplos, ajusta sus parámetros internos y mejora su rendimiento en la tarea.
- En cambio, un **algoritmo matemático tradicional** sigue siempre un conjunto de reglas fijas y predefinidas: no se adapta ni evoluciona con la experiencia, ni mejora sus resultados con el tiempo.

**Complejidad**

- Los **modelos de IA modernos**, especialmente las **redes neuronales profundas (deep learning)**, pueden contener miles o incluso millones de parámetros ajustables. Esta enorme cantidad de variables les permite representar patrones muy complejos, pero también los hace difíciles de interpretar.
- Los **algoritmos matemáticos tradicionales**, en general, son **más simples, transparentes y deterministas**, lo que significa que siempre producen el mismo resultado si se les da la misma entrada.

**Capacidad para resolver tareas no estructuradas**

- Los **algoritmos matemáticos** suelen aplicarse a **problemas bien definidos**, con soluciones exactas o procedimientos establecidos (por ejemplo, resolver una ecuación o calcular un mínimo).
- Los **modelos de IA** pueden abordar **tareas no estructuradas** y ambiguas, como el **reconocimiento de voz, la clasificación de imágenes o la traducción automática**, en las que no existe una solución matemática única ni exacta, sino predicciones aproximadas basadas en patrones aprendidos.

## 9. Limitaciones de los modelos.

Hasta ahora hemos asumido que es posible encontrar un modelo que se ajuste de manera adecuada a los datos disponibles. Sin embargo, en la práctica, **ningún modelo puede proporcionar una explicación completa y perfecta** de un fenómeno observable.

En general, siempre existen dos componentes en los datos:

- Una parte **predecible**, que puede ser explicada mediante un modelo.
- Una parte **aleatoria o impredecible**, que por su naturaleza no puede ser capturada por ningún modelo, por sofisticado que sea.

### Ejemplo extremo: el lanzamiento de una moneda

En este caso, no existen variables, atributos ni características que determinen el resultado. El evento es esencialmente **aleatorio**, y ningún modelo podrá predecir con certeza si saldrá cara o cruz.

### Ejemplo práctico: predicción de enfermedades

A diferencia del lanzamiento de una moneda, en el caso de las enfermedades es posible identificar **factores influyentes** (hábitos de vida, características biológicas, antecedentes médicos, etc.). Estos atributos permiten a un modelo estimar probabilidades, pero siempre quedará una parte de la variabilidad de los datos que el modelo **no podrá explicar del todo**.

## Causas por las que un modelo no logra explicar completamente los datos

### 1. Deficiencias en el modelo

- El modelo puede no considerar variables relevantes que influyen en el fenómeno.
- En estos casos, podría ampliarse el *dataset* con nuevos atributos o incluso probar un tipo de modelo diferente más adecuado.

### 2. Conjunto de datos insuficiente

- Para entrenar correctamente un modelo es necesario contar con un **volumen suficiente de datos**, que refleje la diversidad de situaciones reales.
- Un dataset pequeño o poco representativo limitará la capacidad de generalización del modelo.

### 3. Errores en los datos

- Los datos pueden contener **mediciones inexactas, registros incompletos o valores incorrectos**.

- Estos errores introducen ruido adicional y reducen el rendimiento del modelo.

#### 4. Componentes puramente aleatorias

- Hay factores imposibles de explicar mediante modelos, porque son esencialmente **aleatorios**.
- Esta parte incontrolable se denomina **ruido** (*noise*, en inglés) y afecta negativamente a las métricas de evaluación del modelo.

En resumen:

- Con un conjunto de datos **suficiente, representativo y de calidad**, junto con la elección de un **modelo adecuado**, es posible obtener resultados muy buenos.
- Sin embargo, siempre existirá una **componente aleatoria irreductible** que ningún modelo podrá explicar. Reconocer esta limitación es fundamental para interpretar correctamente los resultados de cualquier sistema de IA o de *machine learning*.