# Machine Learning I

Francisco Javier Mercader Martínez

# Índice

# Tema 1: Introducción al Machine Learning

## ¿Qué es el Machine Learning?

- Definición de Machine Learning

"Descubrir regularidades en datos mediante el uso de algoritmos, y mediante el uso de esas regularidades realizar alguna acción" (C. M. Bishop)

- Tareas básicas

Fundamentalmente cuatro:

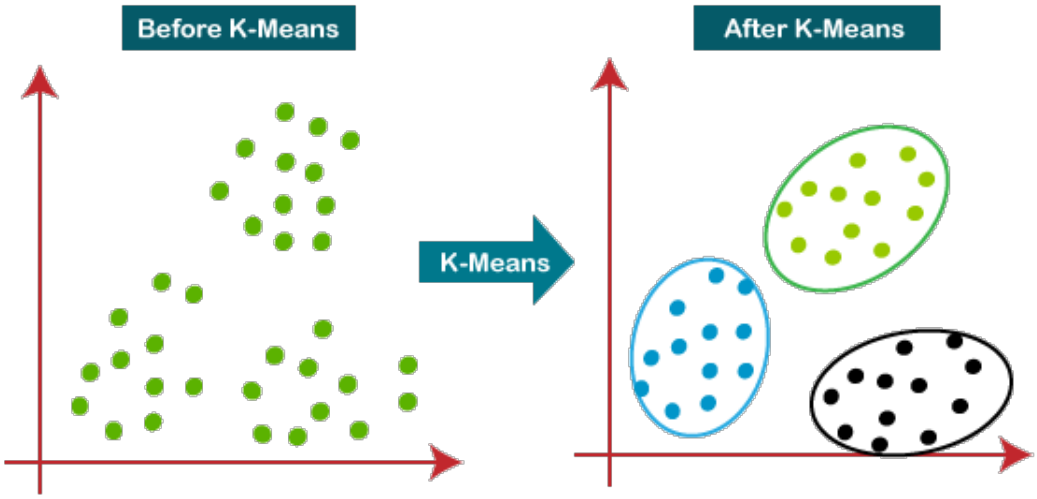
- Clasificación
  - → **Detección de spam:** Se trata de clasificar, mediante identificación de patrones, los correos electrónicos como spam o no spam.
  - → **Detección de fraudes:** Distinción entre transacciones legítimas y sospechosas basándose en patrones y características relevantes.
  - → **Análisis de sentimientos:** Los algoritmos de clasificación pueden utilizarse para determinar el sentimiento expresado en un texto, como positivo, negativo o neutro. Esto es útil para el análisis de opiniones en redes sociales, comentarios de clientes, revisiones de productos, etc.
  - → **Detección de objetos en imágenes:** Especialmente útil en la conducción de coches autónomos.
- Regresión
  - → **Estimación de la demanda de un producto:** Predicción de la demanda de un producto en función de variables como el precio, la publicidad, las tendencias del mercado, entre otras.
  - → **Predicción de la contaminación atmosférica:** Utilizando datos históricos de contaminantes, meteorología y otras variables relevantes, se puede aplicar la regresión para predecir los niveles de contaminación en una ubicación específica.
  - → **Análisis de la relación entre variables económicas:** La regresión puede utilizarse para explorar la relación entre variables económicas, como el crecimiento del PIB y el desempleo, con el fin de entender mejor su interdependencia y tomar decisiones políticas o empresariales informadas.
- Agrupamiento
- Asociación

- Tarea de agrupación en Machine Learning

El **agrupamiento** o **clustering** consiste en detectar agrupaciones en datos no etiquetados empleando alguna media de similitud entre ellas. El objetivo es descubrir patrones y estructuras dentro de los datos.

Algoritmos populares para clustering incluyen el K-Means, el DSCAN, el clustering jerárquico y Mapas Autoorganizados (SOM).

Ejemplo K-Means




- Tarea de asociación en Machine Learning

La tarea de **asociación** se centra en descubrir reglas de asociación entre eventos en un conjunto de datos, lo que significa identificar qué elementos tienden a aparecer juntos en dichos eventos. El objetivo es revelar después del afeitado, hay un 80% de posibilidades de que el cliente compre también crema de afeitado.

La asociación es una tarea no supervisada, los datos a menudo provienen de transacciones o eventos, y no se requieren etiquetas previas.

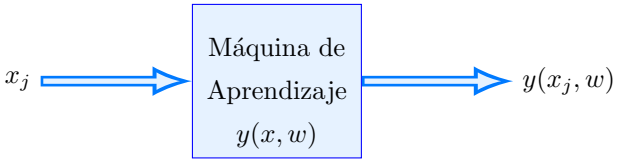
Algoritmos como Apriori se utilizan comúnmente para generar reglas de asociación en los datos, reglas como "Si A, entonces B". Estas reglas se utilizan en análisis de mercado y sistemas de recomendación.

## 1.1) Planteamiento del problema

En el contexto del Machine Learning, el **conjunto de hipótesis** se refiere a un conjunto de funciones o modelos matemáticos que se utilzian para aproximar una relación desconocida entre las **entradas** $(x)$ y las **salidas deseadas o targets** $(t)$ de un conjunto de datos.

Cada hipótesis representa una posible aproximación de la relación subyacente en los datos.

El objetivo del **aprendizaje supervisado** es encontrar la hipótesis que mejor se ajuste a los datos de entrenamiento manteniendo la capacidad de hacer predicciones precisas para datos nuevos **(capacidad de generalización)**.




- Necesario: Conjunto de entrenamiento

  Pares: $\{x_j, t_j\}$ con $j = 1, 2, \ldots, N$.
  $x_j = \{x_{j1}, x_{j2}, \ldots, x_{jD}\}$ entrada $j$-ésima; vector con $D$ **componentes o características**.
  $t_j = \{x_{j1}, x_{j2}, \ldots, x_{jT}\}$ target $j$-ésimo; vector con $T$ componentes.

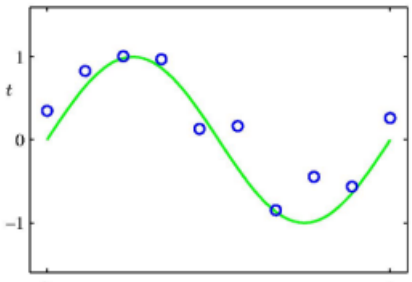
- Objetivo: Aprendizaje supervisado

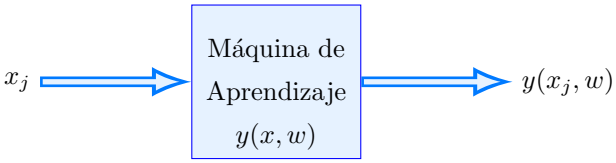
  Encontrar las **variables o pesos** del modelo $(\mathbf{w})$ que resuelvan el problema: $y(x_j, \mathbf{w}) \approx t_j$ para $j = 1, 2, \ldots, N$. A esta tarea se la denomina **entrenamiento**.

Ejemplo: Problema de **regresión**

$y = \sin(2\pi x) + n(x)$, donde $n(x)$ es un ruido gausiano pequeño.

Conjunto de entrenamiento: $\{x_j, t_j\}_{j=1}^{N=10}$






Aproximador polinómico: $y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + x_M x^M = \sum_{j=0}^{M} x_j x^j$

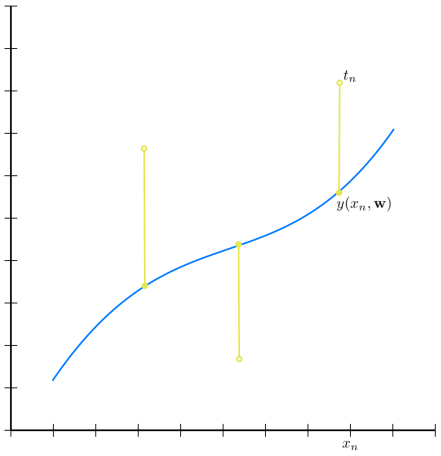
- $M$ es un parámetro que determina la complejidad del modelo (orden del polinomio).
- Los parámetros no entrenables que determinan el modelo o el entrenamiento se denominan en Machine Learning **hiperparámetros**.

## 1.2) Planteamiento de la solución

Se quiere encontrar las variables del modelos (coeficientes del polinomio) para que éste minimice una función de coste o error, por ejemplo, la función de error SSE ("Sum of Square Error") dada por

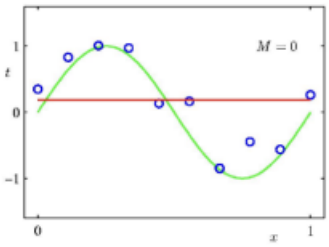
$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2$$

Existe una solución analítica única mediante álgebra lineal.

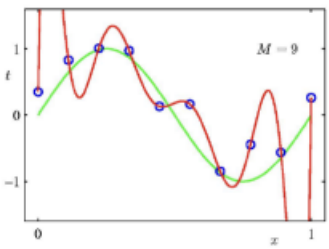



## 1.3) Peligros

### 1.3.1) Subajuste y sobreajuste

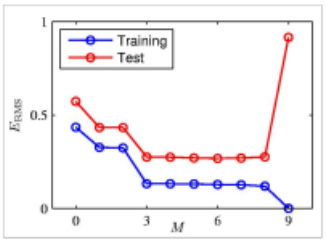



Ajuste pobre




Sobreajuste ("overfitting")

## 1.4) Complejidad del modelo vs número de datos




- Comoportamiento con $M$($N$ fijo)

Fijado $N$, la complejidad del modelo determina la generalización

- Comportamiento con $N$ ($M$ fijo)

Fijado $N(M = 9)$, $N$ condiciona la solución del problema: si es bajo, se puede sobreajustar, si es alto (con relación a la dimensión) se reduce el sobreajuste.

# Tema 2: Aprendizaje Supervisado

## 2.1) Árboles de Decisión

- Definición

Los árboles de decisión son máquinas de aprendizaje supervisado que sirven para clasificar o aproximar.

Supongamos el siguiente problema

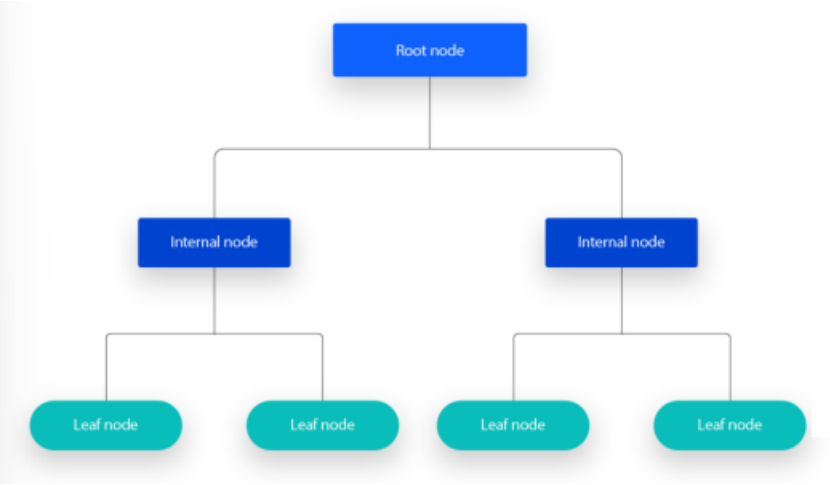
| Paciente | Presión Arterial | Urea en sangre | Gota | Hipotiroidismo | Administrar Tratamiento |
|---|---|---|---|---|---|
| 1 | Alta | Alta | Sí | No | No |
| 2 | Alta | Alta | Sí | Sí | No |
| 3 | Normal | Alta | Sí | No | Sí |
| 4 | Baja | Normal | Sí | No | Sí |
| 5 | Baja | Baja | No | No | Sí |
| 6 | Baja | Baja | No | Sí | No |
| 7 | Normal | Baja | No | Sí | Sí |
| 8 | Alta | Normal | Sí | No | No |
| 9 | Alta | Baja | No | No | Sí |
| 10 | Baja | Normal | No | No | Sí |
| 11 | Alta | Normal | No | Sí | Sí |
| 12 | Normal | Normal | Sí | Sí | Sí |
| 13 | Normal | Alta | No | No | Sí |
| 14 | Baja | Normal | Si | Sí | No |

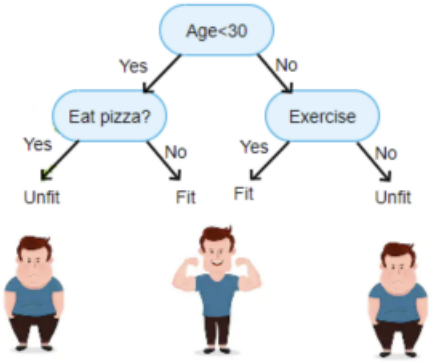
- Planteamiento del problema: ¿Cuál es la **mejor secuencia de preguntas** para saber la clase a la que pertenece un objeto descrito por sus atributos?
- Evidentemente, la "mejor respuesta" es aquella que con el **menor número de preguntas**, devuevle una respuesta suficientemente buena.
- ¿Qué es mejor preguntar primero si tiene gota o cómo tiene la presión arterial?

### 2.1.1) Arquitectura

Un árbol de decisión es una estructura jerárquica que consta de un noda raíz, ramas, nodos internos y nodos hoja.

- Comienzo con un **nodo raíz** sin ramas entrantes. Las ramas salientes del nodo raíz alimentan los nodos internos.
- Los **nodos internos** evalúan características disponibles para formar subconjuntos homogéneos, indicados por nodos hoja o nodos terminales.
- Los **nodos hoja** representan todos los resultados posibles dentro del conjunto de datos.




### 2.1.2) Ventajas y desventajas

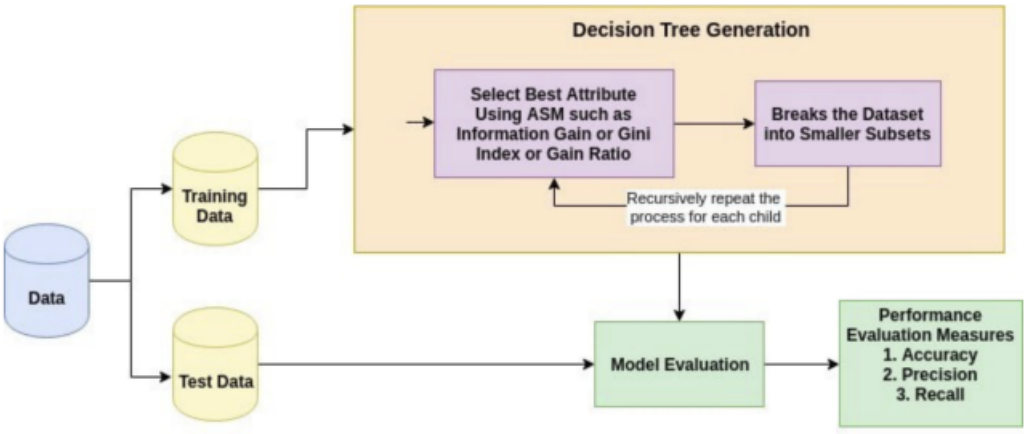
- Pros
  - Fáciles de entender e interpretar.
  - Sirven también para establecer reglas
  - No lineales
  - Menos pre-procesado de los datos: son robustos ante presencia de datos erróneos (outlier), valores faltantes o tipo de datos.
  - Es un método no paramétrico (por ejemplo, no hay suposición acerca del espacio de distribución y la estructura del clasificador).
- Contras
  - **Sobreajuste:** Los árboles más pequeños son más fáciles de interpretar, pero los más grandes pueden resultar en sobreajuste.
  - Perdida de información al categorizar variables continuas.
  - **Precisión:** Otros métodos (por ejemplo, SVM) a menudo tienen tasas de error 30% más bajas que los árboles básico (ID.3 y CART).
  - **Inestabilidad:** un pequeó cambio en los datos puede modificar ampliamente la estructura del árbol (distintos conjuntos, distintos árboles). Varianza elevada.

- Definición alternativa: **recursividad**

Un árbol de decisión es una estructura recursiva formada por nodos, en el que existe:

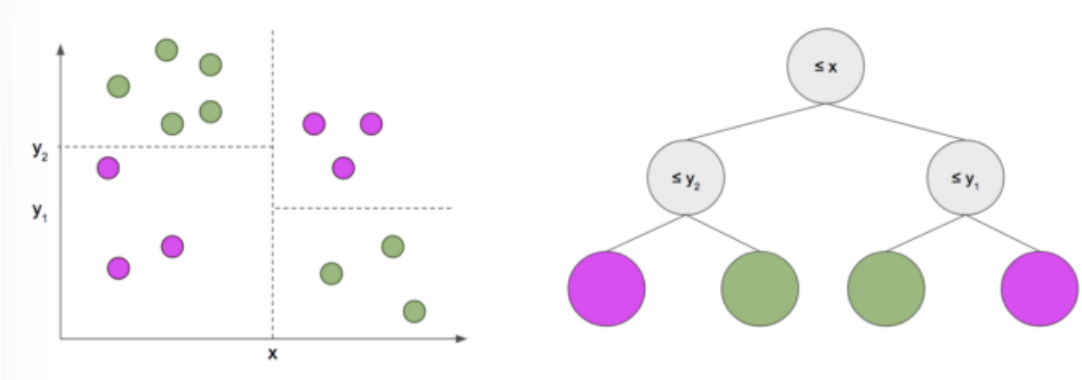
- Un nodo raíz
- El nodo raíz tiene uno o más subnodos.
- Cada uno de los subnodos puede ser, a su vez, raíz de un árbol

Esta característica recursiva hace que muchos de los algoritmos para crearlos se comporten también de manera recursiva.

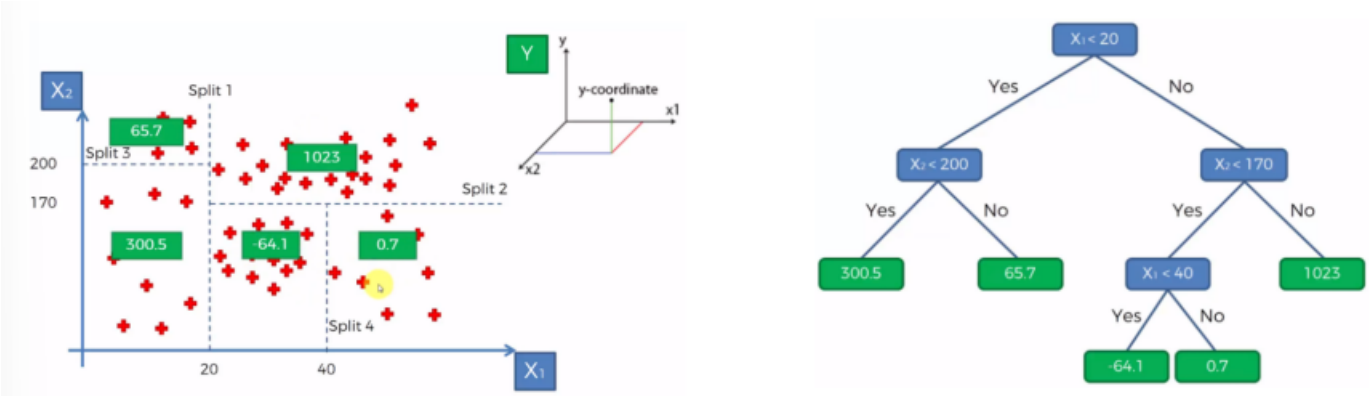



### 2.1.3) Clasificación vs Regresión

- Clasificación
  - La variable dependiente es categórica.
  - Los valores de los nodos hoja son la **moda** de las observaciones de la región




- Regresión
  - La variable dependiente es continua.
  - Los valores de los nodos hoja son la **media** de las observaciones de la región.




## 2.2) Construcción de árboles de decisión

### 2.2.1) Particiones

Cada nodo define una **partición** del conjunto de entrenamiento en función de los datos que representa.
Las particiones producen subconjuntos que son **exhaustivos** y **excluyentes**.
Cuestiones clave:

- **Tipos de particiones:** cuantos más, más posibilidad de encontrar patrones y, por tanto, los árboles más precisos y expresivos.
- **Número de particiones:** A más particiones mayor complejidad. Equilibrio entre complejidad y precisión.
- Selección del **mejor atributo** en cada paso.
- Selección del **mejor valor** de umbral de los valores.

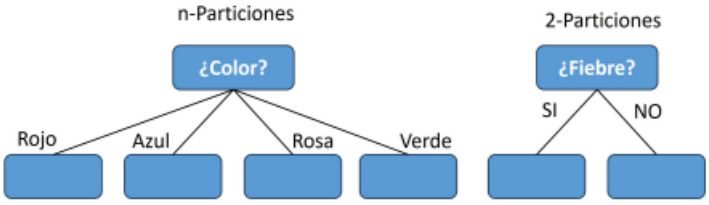
### 2.2.2) Particiones posibles

Los algoritmos más populares sólo proponen un tipo de partición para valores nominales y otro para valores numéricos:

- **Particiones nominales:** En el caso que tengamos un atributo $x_i$ que tenga como posibles valores $\{v_1, v_2, \ldots, v_n\}$ sólo es posible la partición

$$(x_1 = v_1, x_2 = v_2, \cdots, x_n = v_n)$$

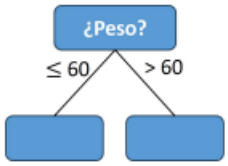
que da lugar a árboles con nodos con más de dos nodos hijos.



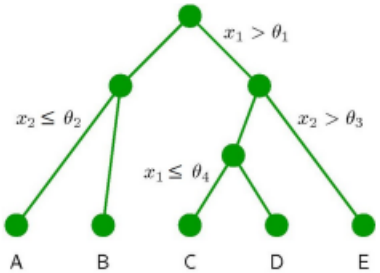

En el caso de árboles binarios se tienen que evaluar $n$ particiones (una por cada posible valor), definidas por $(x_i = v_i, x_i \neq v_i)$.

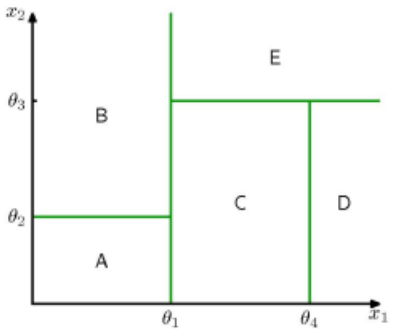
- **Particiones numéricas:** Si el atributo $x_i$ es numérico y continuo, se intenta definir particiones que separe las instancias en intervalos de la forma

$$(x_i \leq a, x_i > a)$$




eligiendo diferentes valores de $a$ tenemos diferentes particiones. La expresividad resultante se conoce como *expresividad cuadrangular* y que no relacionan atributos (sólo un atributo cada vez).



## 2.3) ID3: Algoritmo básico de aprendizaje

> El algoritmo básico de aprendizaje es el **ID3 (Iterative Dichotomiser 3)**, J. Ross Quinlan, investigador australiano que propuso el método en 1983

El método ID3 trata de encontrar una partición que asegure la **máxima capacidad predictiva y la máxima homogeneidad** de las clases
Medida de homogeneidad: la **entropía**
Repetición de **"cortes en dos"** hasta que se cumpla una determinada condición

### 2.3.1) Entropía

Para determinar el mejor atributo, el ID3 utiliza la **entropía**.

Sea $S$ un conjunto de entrenamiento. Sea $p_\oplus$ la proporción de instancias positivas en $S$ y $p_\ominus$ la proporción de instancias negativas en $S$. La **entropía de** $S$ es:

$$H(S) = p_\oplus \log_2 \frac{1}{p_\oplus} + p_\ominus \log_2 \frac{1}{p_\ominus} = -p_\oplus \log_2 p_\oplus - p_\ominus \log_2 p_\ominus$$

(Relación de la entropía con los conceptos de desorden, equiprbabilidad y homogeneidad).