MACHINE LEARNING - INDUCCIÓN A ÁRBOLES DE DECISIÓN

Víctor Guevara Ponce
victor.guevarap@urp.edu.pe

Ciencia de datos - IA

2020

AGENDA

1. Clasificación
   - Introducción
2. Arbóles de clasificación
   - Arbóles de clasificación
   - Construcción de un árbol de decisión
   - Algoritmo de Hunt
   - Medidas de impureza
   - Evaluación de modelos

Machine learning - Definición

Machine learning

Machine learning es una herramienta de la inteligencia artificial (IA) que proporciona a los sistemas la capacidad de aprender y mejorar automáticamente a partir de la experiencia sin ser programado explícitamente. El ML se centra en el desarrollo de programas informáticos que pueden acceder a los datos y utilizarlos para aprender por sí mismos.

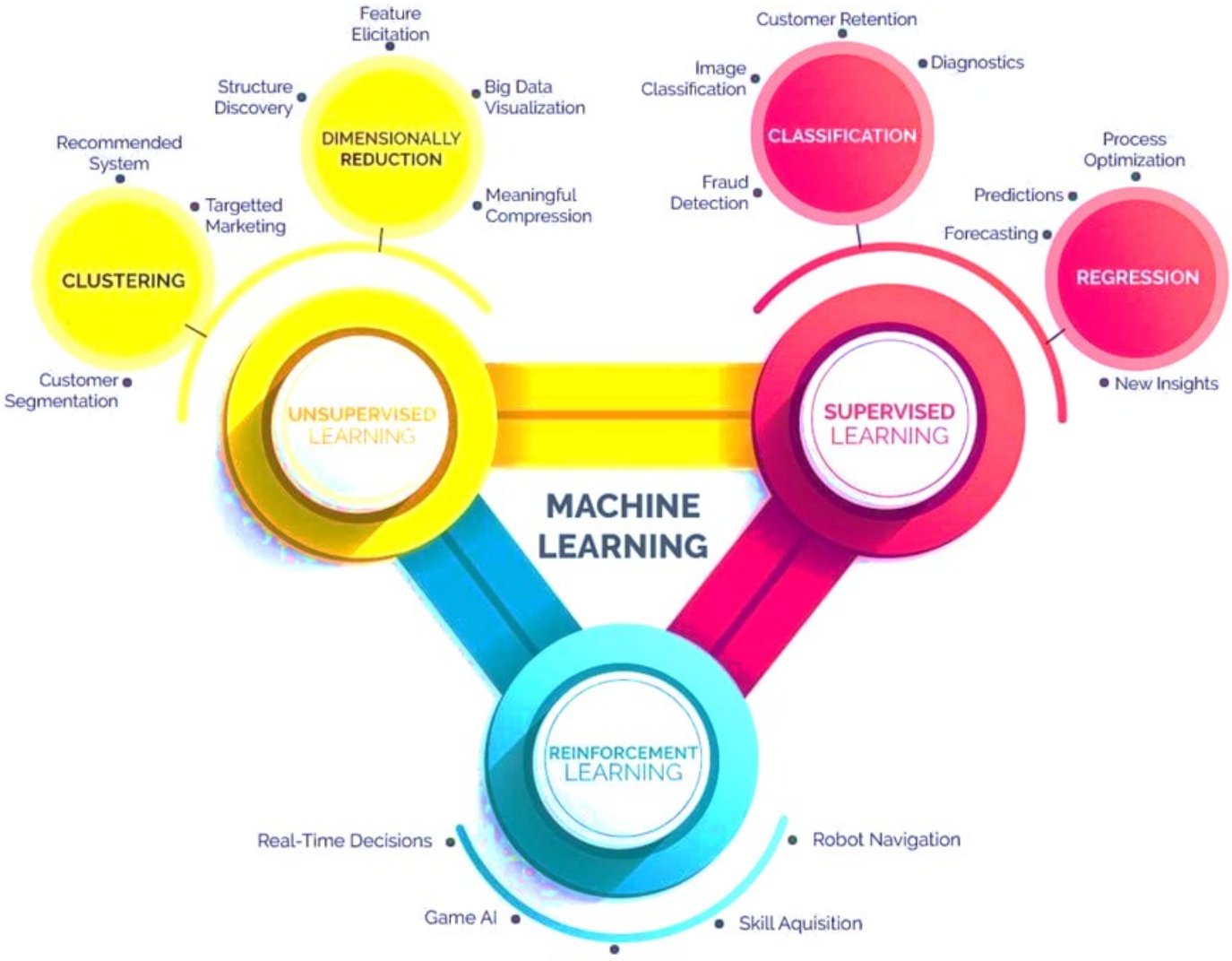
Tom Mitchell (1997)

Un programa de computadora aprende de la experiencia **E** con respecto a alguna clase de tareas **T** y la medida de rendimiento **P**, si su desempeño en las tareas en **T**, medido por **P**, mejora con la experiencia **E**.

Machine learning - Aplicaciones

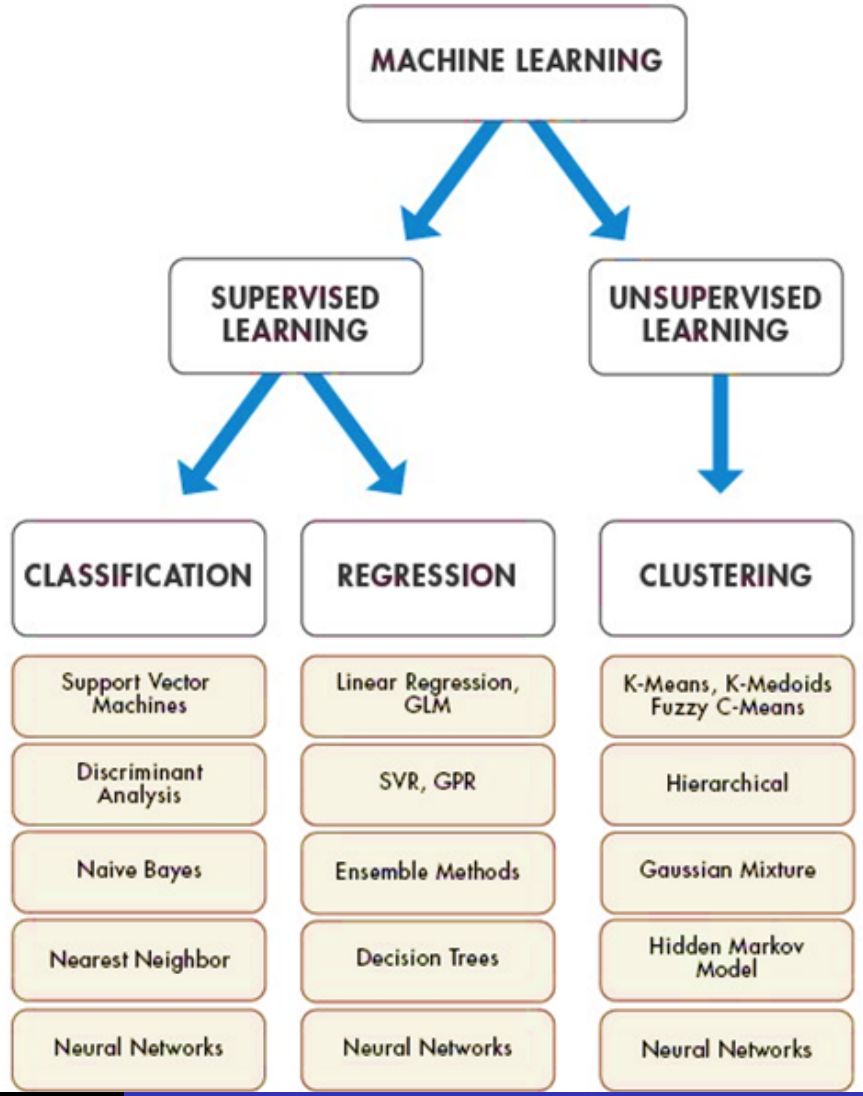
- Identificar personas basadas en imágenes o grabaciones de voz
- Identificación de clientes rentables
- Identificar si una transacción bancaria es fraudulenta o no.
- Identificar de forma proactiva las piezas de automóviles que pueden fallar
- Identificación de tumores y diversas enfermedades
- Predecir la cantidad de dinero que una persona gastará en un producto X
- Predecir los ingresos anuales de su empresa (regresión)
- Predecir qué equipo ganará la Champions League en fútbol (clasificación)
- Detección de fraude en transacciones de tarjetas de crédito

Machine Learning - Tipos

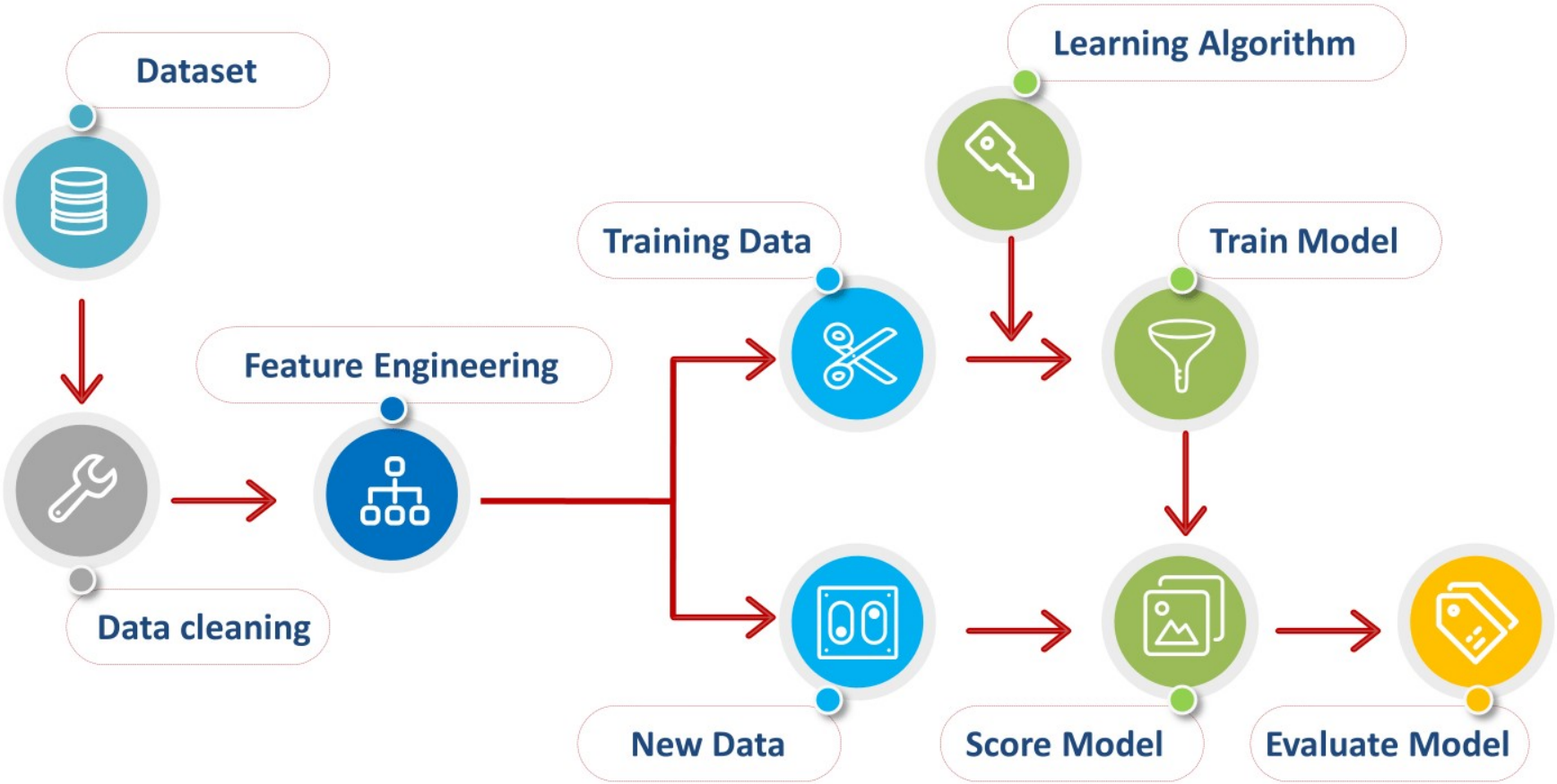


Machine Learning - Algoritmos

Una forma posible es refinar la tarea del machine learning es observar las clases de problemas que puede resolver. Éstos son algunos de los más comunes:



Machine Learning - workflow



Machine Learning - Clasificación

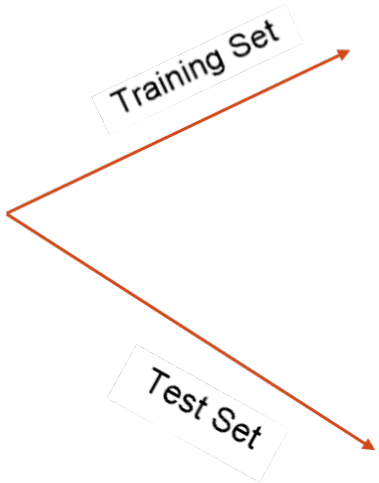
- Dado un conjunto de registros (conjunto de entrenamiento)
- Cada registro contiene un conjunto de atributos, donde uno de ellos es la clase.
- Encontrar un modelo para el atributo de clase en función de los valores de los demás atributos.
- Objetivo: Nuevos registros sean asignados a una clase con la mayor precisión posible.

Un conjunto de prueba es usada para determinar la precisión del modelo. Usualmente, el conjunto de datos original es dividido en un conjunto de prueba y de entrenamiento, donde el conjunto de entrenamiento es usado para construir el modelo y el de prueba para validarlo.

Modelo de clasificación

CONJUNTO DE DATOS

ID	Casa propia	Estado civil	Ingreso anual	Default
1	Si	Soltero	125	No
2	No	Casado	100	No
3	No	Soltero	70	No
4	Si	Casado	120	No
5	No	Divorciado	95	Si
6	No	Casado	60	No
7	Si	Divorciado	220	No
8	No	Soltero	85	Si
9	No	Casado	75	No
10	No	Soltero	90	Si
11	No	Soltero	55	No
12	Si	Casado	80	No
13	Si	Divorciado	110	Si
14	No	Soltero	95	No
15	No	Casado	67	Si



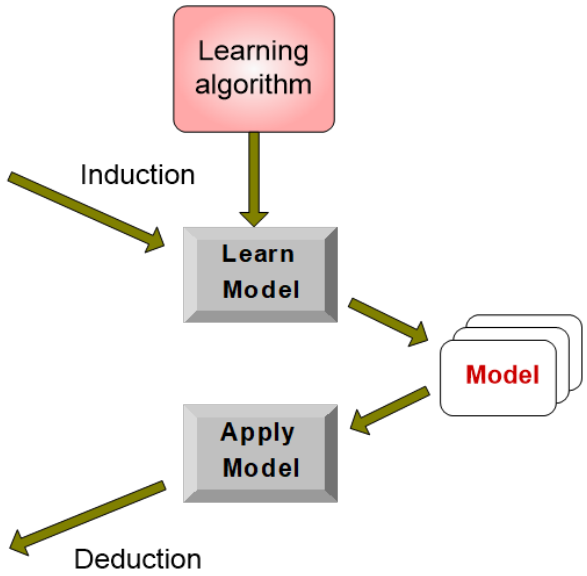
ID	Casa propia	Estado civil	Ingreso anual	Default
1	Si	Soltero	125	No
2	No	Casado	100	No
3	No	Soltero	70	No
4	Si	Casado	120	No
5	No	Divorciado	95	Si
6	No	Casado	60	No
7	Si	Divorciado	220	No
8	No	Soltero	85	Si
9	No	Casado	75	No
10	No	Soltero	90	Si

ID	Casa propia	Estado civil	Ingreso anual	Default
11	No	Soltero	55	No
12	Si	Casado	80	No
13	Si	Divorciado	110	Si
14	No	Soltero	95	No
15	No	Casado	67	Si

Modelo de clasificación

Training Set

ID	Casa propia	Estado civil	Ingreso anual	Default
1	Si	Soltero	125	No
2	No	Casado	100	No
3	No	Soltero	70	No
4	Si	Casado	120	No
5	No	Divorciado	95	Si
6	No	Casado	60	No
7	Si	Divorciado	220	No
8	No	Soltero	85	Si
9	No	Casado	75	No
10	No	Soltero	90	Si



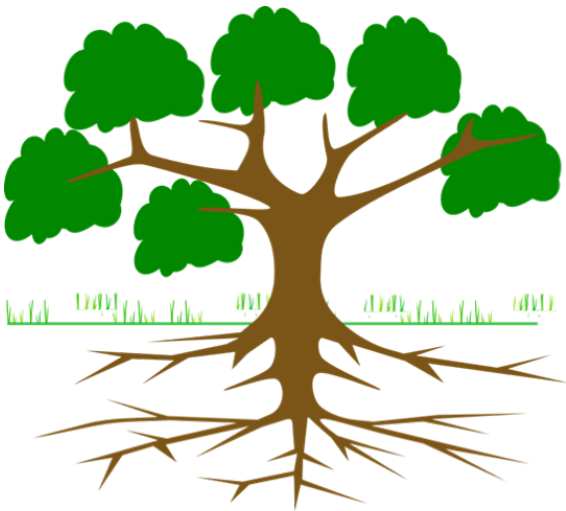
Test Set

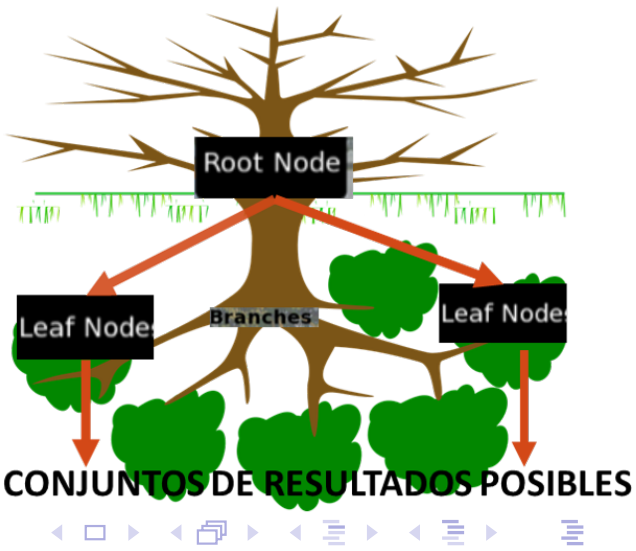
ID	Casa propia	Estado civil	Ingreso anual	Default
11	No	Soltero	55	No
12	Si	Casado	80	No
13	Si	Divorciado	110	Si
14	No	Soltero	95	No
15	No	Casado	67	Si

Arboles de decisión

¿Qué son árboles de decisión?

Es una herramienta de soporte de decisiones que utiliza un modelo de decisiones tipo árbol y sus posibles consecuencias. Es una forma de mostrar un algoritmo que solo contiene sentencias de control condicional.





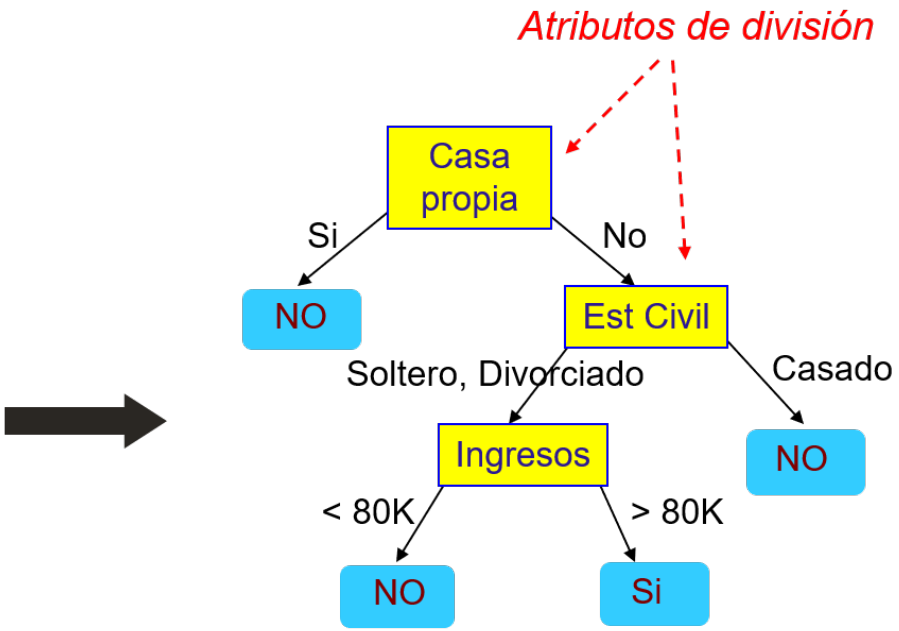
Arboles de decisión

- Nodo Interno (Root Node): denota una prueba sobre un atributo.
- Rama (Branch): corresponde a un valor de atributo y representa el resultado de una prueba
- Nodo Terminal (Leaf Node): representa una etiqueta de clase o de distribución de clase Cada camino es una conjunción de valores de atributos

Arboles de decisión

	Categorica	Categorica	Continua	clase
ID	Casa propia	Estado civil	Ingreso anual	Default
1	Si	Soltero	125	No
2	No	Casado	100	No
3	No	Soltero	70	No
4	Si	Casado	120	No
5	No	Divorciado	95	Si
6	No	Casado	60	No
7	Si	Divorciado	220	No
8	No	Soltero	85	Si
9	No	Casado	75	No
10	No	Soltero	90	Si

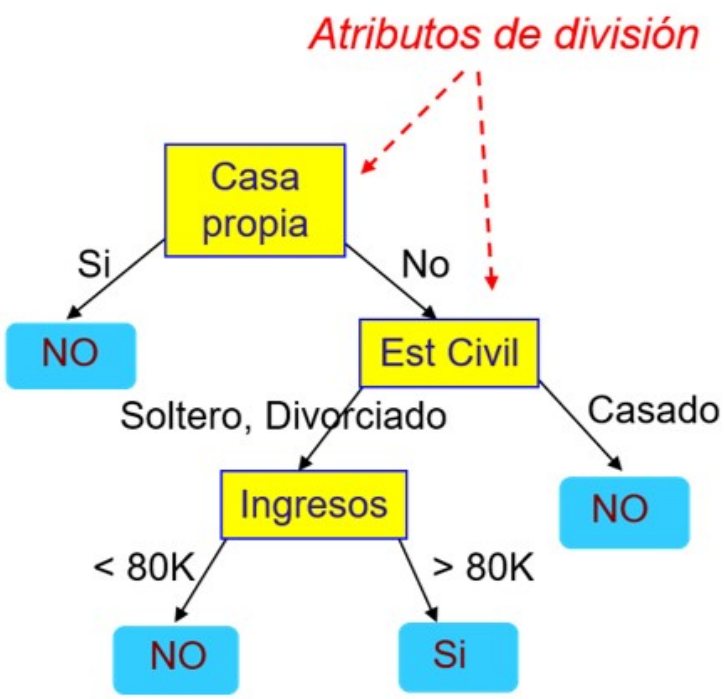
Training Data



Modelo: Árbol de desición

Arboles de decisión

Aplicación en la data de prueba (test)



Data de prueba (test)

ID	Casa propia	Estado civil	Ingreso anual	Default
20	No	Casado	80K	?

Modelo: Árbol de desición

aplicación de árboles de decisión nuevos datos

¿Cómo se utilizan los árboles de decisión para la clasificación?

1. Dado un conjunto de datos de entrada (X) Para la cual la etiqueta de clase asociada es desconocida (y), los valores de atributo de los datos de entrada se comparan con el árbol de decisión.

aplicación de árboles de decisión nuevos datos

¿Cómo se utilizan los árboles de decisión para la clasificación?

1. Dado un conjunto de datos de entrada (X) Para la cual la etiqueta de clase asociada es desconocida (y), los valores de atributo de los datos de entrada se comparan con el árbol de decisión.
2. Se rastrea una ruta desde la raíz a un nodo hoja, que contiene la predicción de clase para tales datos.

aplicación de árboles de decisión nuevos datos

¿Cómo se utilizan los árboles de decisión para la clasificación?

1. Dado un conjunto de datos de entrada (X) Para la cual la etiqueta de clase asociada es desconocida (y), los valores de atributo de los datos de entrada se comparan con el árbol de decisión.
2. Se rastrea una ruta desde la raíz a un nodo hoja, que contiene la predicción de clase para tales datos.
3. Los árboles de decisión se pueden convertir fácilmente en reglas de clasificación.

Arboles de decisión

Los árboles de Decisión son atractivos debido a que:

- Se usan mucho por su interpretabilidad

Arboles de decisión

Los árboles de Decisión son atractivos debido a que:

- Se usan mucho por su interpretabilidad
- Para evaluar que variables son importantes y como ellas interactúan una con otra.

Arboles de decisión

Los árboles de Decisión son atractivos debido a que:

- Se usan mucho por su interpretabilidad
- Para evaluar que variables son importantes y como ellas interactúan una con otra.
- Los resultados se pueden expresar fácilmente mediante reglas.

Arboles de decisión

Los árboles de Decisión son atractivos debido a que:

- Se usan mucho por su interpretabilidad
- Para evaluar que variables son importantes y como ellas interactúan una con otra.
- Los resultados se pueden expresar fácilmente mediante reglas.
- Son robustos a los outliers.

Arboles de decisión

Los árboles de Decisión son atractivos debido a que:

- Se usan mucho por su interpretabilidad
- Para evaluar que variables son importantes y como ellas interactúan una con otra.
- Los resultados se pueden expresar fácilmente mediante reglas.
- Son robustos a los outliers.
- Los algoritmos recursivos tienen una forma especial de tratar los valores faltantes: como un nivel aparte de la variable objetivo.

Arboles de decisión

Los árboles de Decisión son atractivos debido a que:

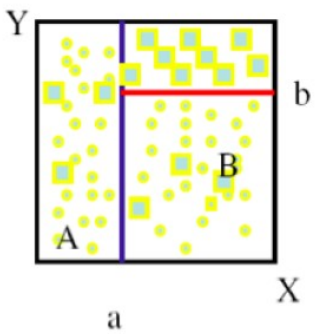
- Se usan mucho por su interpretabilidad
- Para evaluar que variables son importantes y como ellas interactúan una con otra.
- Los resultados se pueden expresar fácilmente mediante reglas.
- Son robustos a los outliers.
- Los algoritmos recursivos tienen una forma especial de tratar los valores faltantes: como un nivel aparte de la variable objetivo.
- Los valores faltantes pueden ser agrupados con otros valores y ser tratados en un nodo.

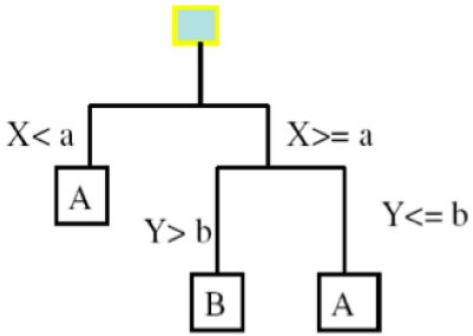
Arboles de decisión

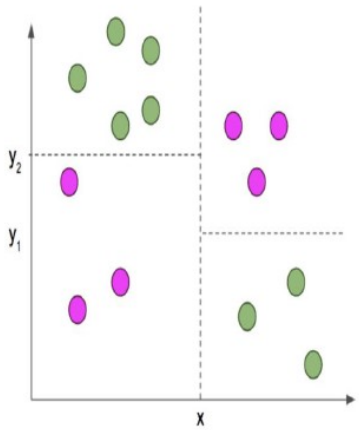
Existen muchos algoritmos, la gran mayoría varia de acuerdo a su criterio de división:

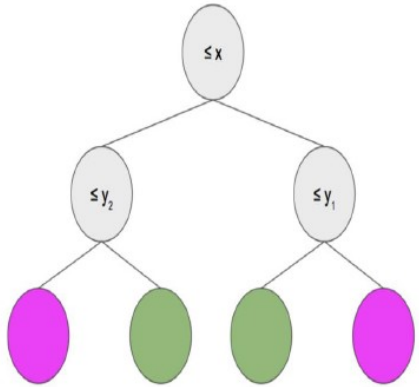
- Hunt's Algorithm (Uno de los primeros)
- CART
- CHAID
- ID3 (Machine learning)
- C4.5 (Sucesor de ID3)
- SLIQ,SPRINT

Construcción de un árbol de decisión









Construcción de un árbol de decisión

La generación de un árbol de decisión consiste de dos fases

- Construcción del Árbol
  - Al inicio, todas las observaciones de entrenamiento están en la raiz.
  - Se realizan particiones recursivas basadas en los atributos seleccionados.
- Poda del Árbol
  - Identifican y remueven ramas que causen ruido o tengan outliers.

Construcción de un árbol de decisión

- Un árbol de decisión particiona el espacio de variables predictoras en un conjunto de hiperrectangulos y en cada uno de ellos ajusta un modelo sencillo, generalmente una constante. Es decir, $y = c$, donde y es la variable de respuesta.

Construcción de un árbol de decisión

- Un árbol de decisión particiona el espacio de variables predictoras en un conjunto de hiperrectangulos y en cada uno de ellos ajusta un modelo sencillo, generalmente una constante. Es decir, $y = c$, donde y es la variable de respuesta.
- La construcción de un árbol de decisión se basa en cuatro elementos

Construcción de un árbol de decisión

- Un árbol de decisión particiona el espacio de variables predictoras en un conjunto de hiperrectangulos y en cada uno de ellos ajusta un modelo sencillo, generalmente una constante. Es decir, y = c, donde y es la variable de respuesta.
- La construcción de un árbol de decisión se basa en cuatro elementos
  - Un conjunto de preguntas binarias Q de la forma x A

Construcción de un árbol de decisión

- Un árbol de decisión particiona el espacio de variables predictoras en un conjunto de hiperrectangulos y en cada uno de ellos ajusta un modelo sencillo, generalmente una constante. Es decir, y = c, donde y es la variable de respuesta.
- La construcción de un árbol de decisión se basa en cuatro elementos
  - Un conjunto de preguntas binarias Q de la forma x A
  - El método usado para particionar los nodos.

Construcción de un árbol de decisión

- Un árbol de decisión particiona el espacio de variables predictoras en un conjunto de hiperrectangulos y en cada uno de ellos ajusta un modelo sencillo, generalmente una constante. Es decir, y = c, donde y es la variable de respuesta.
- La construcción de un árbol de decisión se basa en cuatro elementos
  - Un conjunto de preguntas binarias Q de la forma x A
  - El método usado para particionar los nodos.
  - La estrategia requerida para el crecimiento del árbol.

Construcción de un árbol de decisión

- Un árbol de decisión particiona el espacio de variables predictoras en un conjunto de hiperrectangulos y en cada uno de ellos ajusta un modelo sencillo, generalmente una constante. Es decir, y = c, donde y es la variable de respuesta.
- La construcción de un árbol de decisión se basa en cuatro elementos
  - Un conjunto de preguntas binarias Q de la forma x A
  - El método usado para particionar los nodos.
  - La estrategia requerida para el crecimiento del árbol.
  - La asignación de cada nodo terminal a una clase de la variable respuesta.

Construcción de un árbol de decisión

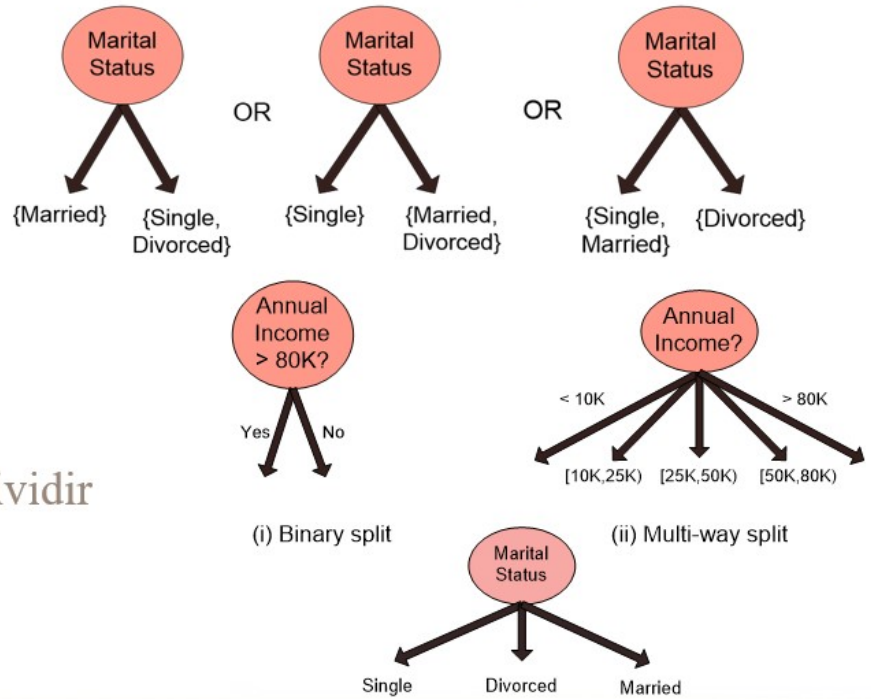
- Un árbol de decisión particiona el espacio de variables predictoras en un conjunto de hiperrectangulos y en cada uno de ellos ajusta un modelo sencillo, generalmente una constante. Es decir, y = c, donde y es la variable de respuesta.
- La construcción de un árbol de decisión se basa en cuatro elementos
  - Un conjunto de preguntas binarias Q de la forma x A
  - El método usado para particionar los nodos.
  - La estrategia requerida para el crecimiento del árbol.
  - La asignación de cada nodo terminal a una clase de la variable respuesta.
- Las diferencias entre los algoritmos para construir arboles se hallan en la regla para particionar los nodos, la estrategia para podar los arboles, y el tratamiento de valores perdidos ("missing values")

Especificación de la condición a evaluar

Métodos para expresar condiciones de prueba

- Depende de los tipos de atributos.
  - Binario
  - Nominal
  - Ordinal
  - Continuo

- Depende de la cantidad de maneras de dividir
  - División de 2 vías
  - División de múltiples vías



Algoritmo de Hunt's

- En el algoritmo de Hunt, un árbol de decisión crece de manera recursiva dividiendo (particionando) los registros de entrenamiento en subconjuntos sucesivamente más puros.

Algoritmo de Hunt's

- En el algoritmo de Hunt, un árbol de decisión crece de manera recursiva dividiendo (particionando) los registros de entrenamiento en subconjuntos sucesivamente más puros.
- Dt es el conjunto de registros de entrenamiento que están asociados con los nodos t y y = y1, y2,. . . , yc sean las etiquetas de clase. La siguiente es una definición recursiva del algoritmo de Hunt.

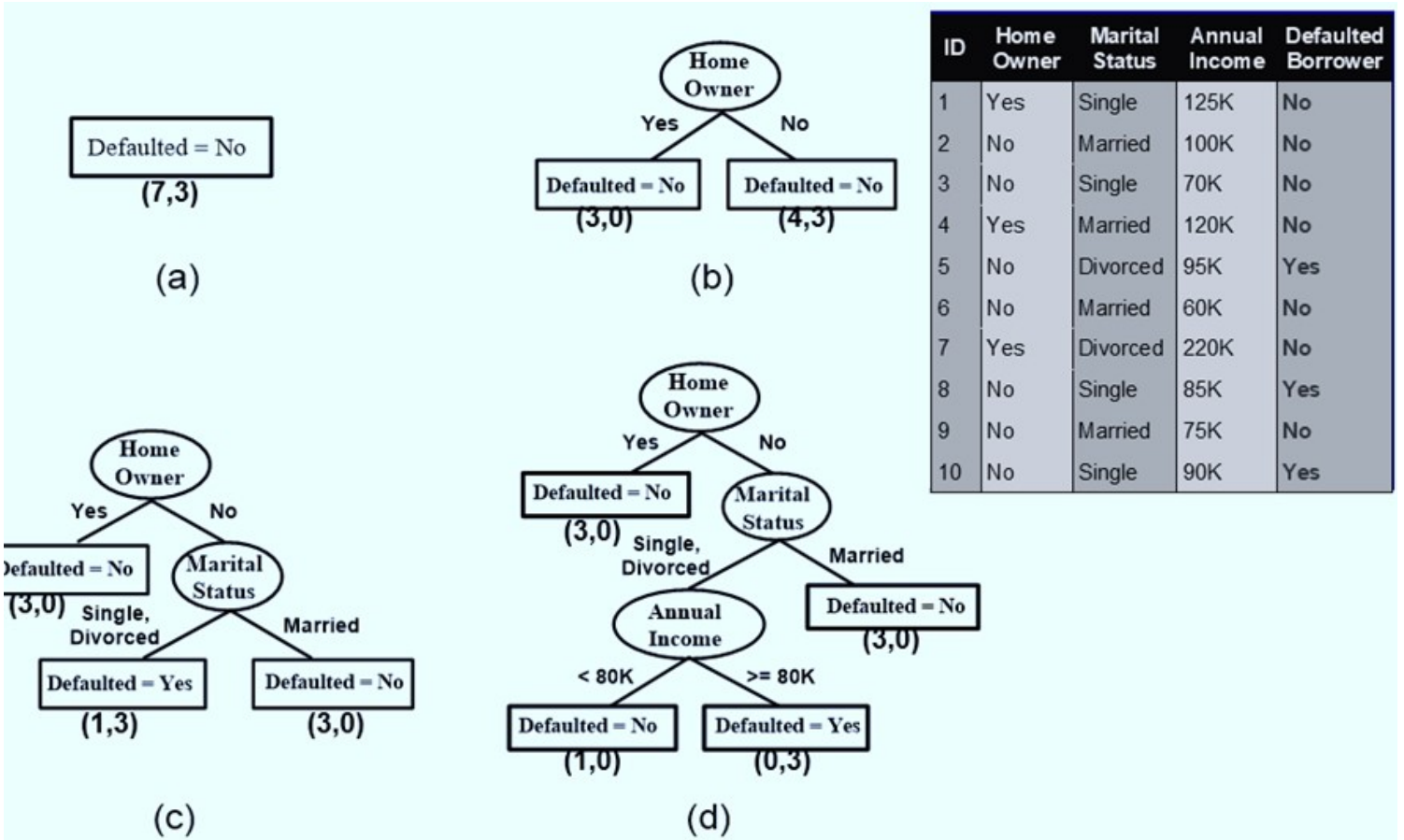
Algoritmo de Hunt's

- En el algoritmo de Hunt, un árbol de decisión crece de manera recursiva dividiendo (particionando) los registros de entrenamiento en subconjuntos sucesivamente más puros.
- Dt es el conjunto de registros de entrenamiento que están asociados con los nodos t y y = y1, y2,. . . , yc sean las etiquetas de clase. La siguiente es una definición recursiva del algoritmo de Hunt.
  1. Paso 1: Si todos los registros en Dt pertenecen a la misma clase yt, entonces t es un nodo terminal etiquetado como yt.

Algoritmo de Hunt's

- En el algoritmo de Hunt, un árbol de decisión crece de manera recursiva dividiendo (particionando) los registros de entrenamiento en subconjuntos sucesivamente más puros.
- Dt es el conjunto de registros de entrenamiento que están asociados con los nodos t y y = y1, y2,. . . , yc sean las etiquetas de clase. La siguiente es una definición recursiva del algoritmo de Hunt.
  1. Paso 1: Si todos los registros en Dt pertenecen a la misma clase yt, entonces t es un nodo terminal etiquetado como yt.
  2. Paso 2: si Dt contiene registros que pertenecen a más de una clase, se selecciona una condición de prueba de atributo para dividir los registros en subconjuntos más pequeños. Se crea un nodo hijo para cada resultado de la condición de prueba y los registros en Dt se distribuyen a los hijos según los resultados. El algoritmo se aplica recursivamente a cada nodo secundario.

Algoritmo de Hunt's



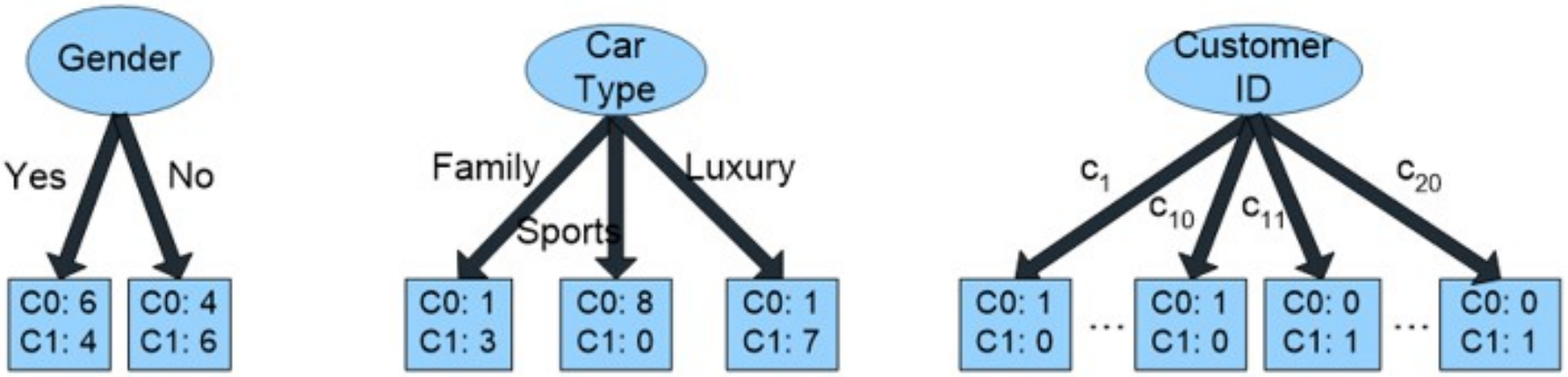
ID	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

¿Cómo determinar la mejor división?

Customer Id	Gender	Car Type	Shirt Size	Class
1	M	Family	Small	C0
2	M	Sports	Medium	C0
3	M	Sports	Medium	C0
4	M	Sports	Large	C0
5	M	Sports	Extra Large	C0
6	M	Sports	Extra Large	C0
7	F	Sports	Small	C0
8	F	Sports	Small	C0
9	F	Sports	Medium	C0
10	F	Luxury	Large	C0
11	M	Family	Large	C1
12	M	Family	Extra Large	C1
13	M	Family	Medium	C1
14	M	Luxury	Extra Large	C1
15	F	Luxury	Small	C1
16	F	Luxury	Small	C1
17	F	Luxury	Medium	C1
18	F	Luxury	Medium	C1
19	F	Luxury	Medium	C1
20	F	Luxury	Large	C1

¿Cómo determinar la mejor división?

¿Qué condición de prueba es la mejor?



- Se prefieren los nodos con una distribución de clase más pura.
- Necesita una medida de impureza de nodo:

C0: 5 C1: 5	C0: 9 C1: 1
Alto grado de impureza	**Bajo grado de impureza.**

Medidas de impureza de un nodo

Índice Gini

$$GINI(t) = 1 - \sum_j [p(j|t)]^2 \quad (1)$$

Entropía

$$Entropia(t) = -\sum_j p(j|t) \log p(j|t) \quad (2)$$

Error de clasificación

$$Error(t) = 1 - \max_k P(i|t) \quad (3)$$

¿Cómo determinar la mejor división?

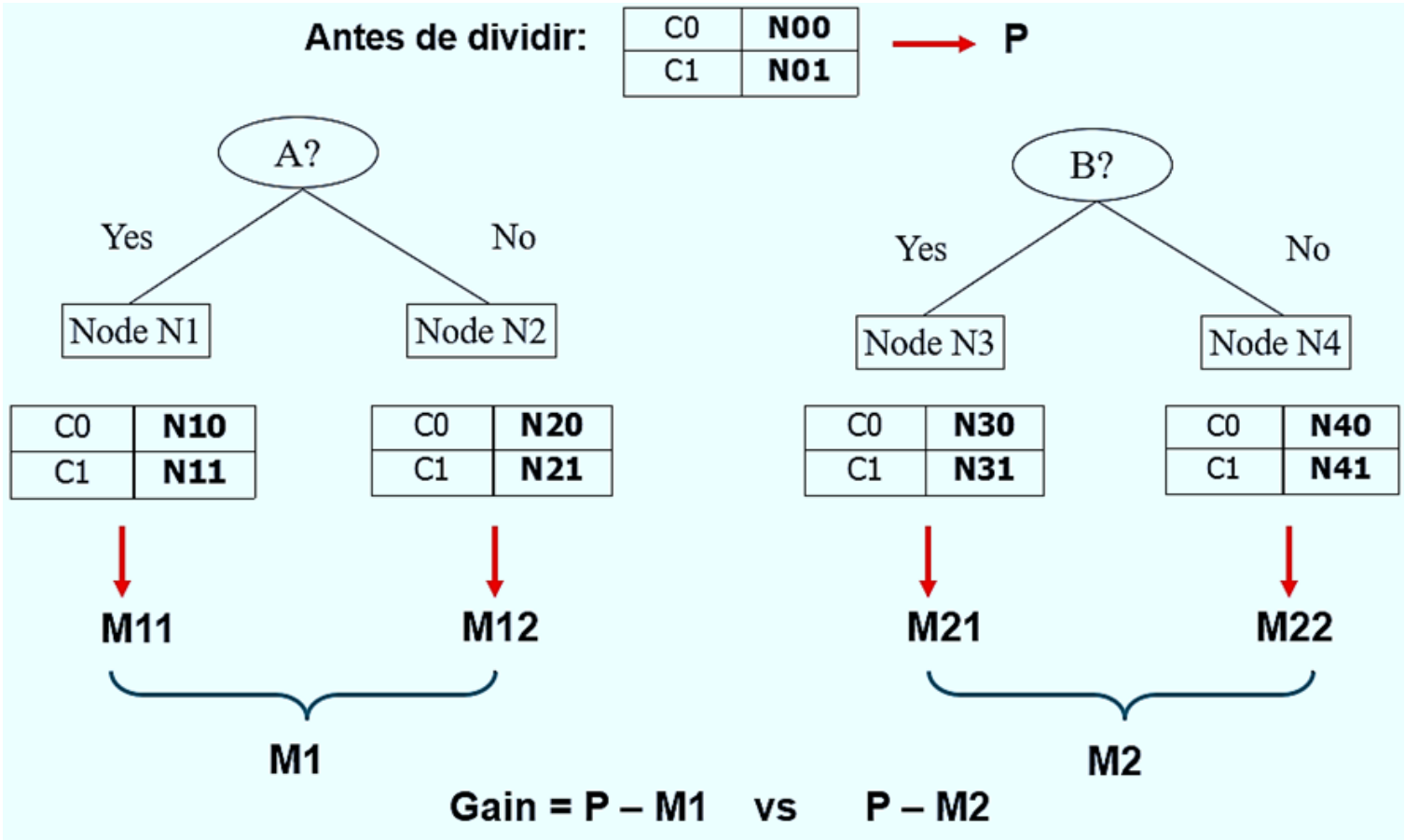
Para encontrar la mejor división

1. Calcular la medida de impureza (P) antes de dividir
2. Calcule la medida de impureza (M) después de dividir
   - Calcular la medida de impureza de cada nodo hijo.
   - M es la impureza ponderada de los hijos.
3. Elija la condición de prueba de atributo que produce la ganancia más alta

$$Gain = \mathrm{P} - \mathrm{M}$$

o equivalentemente, la medida de impureza más baja después de dividir (M)

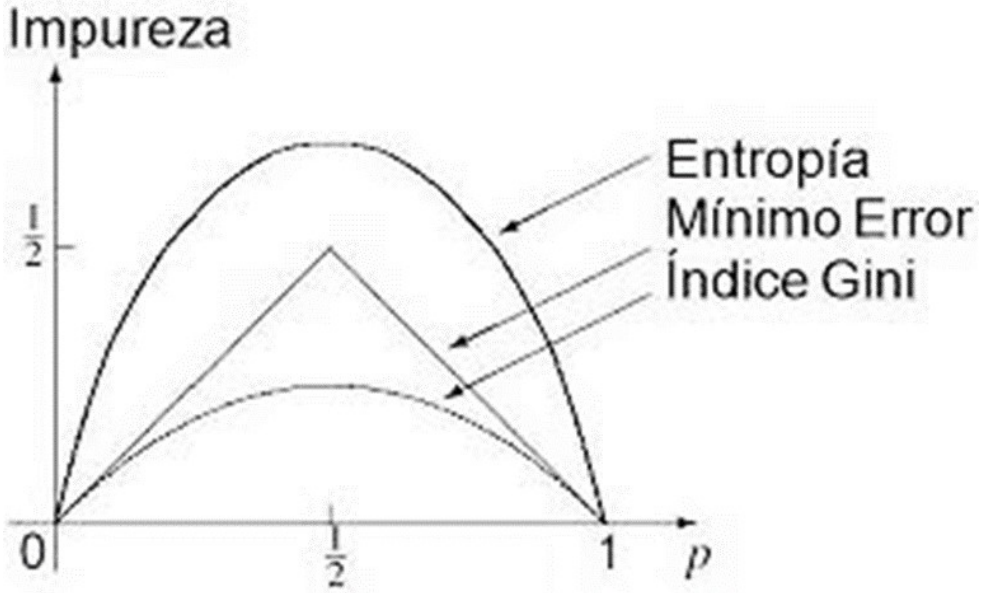
¿Cómo determinar la mejor división?



Medidas de Impureza

Máximo (1 - 1 /n_c) cuando los registros se distribuyen por igual entre todas las clases, lo que implica la información menos interesante

Mínimo (0.0) cuando todos los registros pertenecen a una clase, lo que implica la información más interesante.



Impureza del ärbol

$$I(T) = \sum_{t \in T} i(t)p(t)$$

donde T es el conjunto de nodos terminales del árbol y $p(t)$ es la probabilidad que un caso esté en el nodo t..

Cálculo del índice de Gini de un solo nodo

$$GINI(t) = 1 - \sum_j [p(j \mid t)]^2$$

C1	0
C2	6

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Gini = 1 – P(C1)² – P(C2)² = 1 – 0 – 1 = 0

C1	1
C2	5

P(C1) = 1/6 P(C2) = 5/6

Gini = 1 – (1/6)² – (5/6)² = 0.278

C1	2
C2	4

P(C1) = 2/6 P(C2) = 4/6

Gini = 1 – (2/6)² – (4/6)² = 0.444

Índice de Gini para una colección de nodos

Cuando un nodo p se divide en k particiones (hijos)

$$GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i) \tag{4}$$

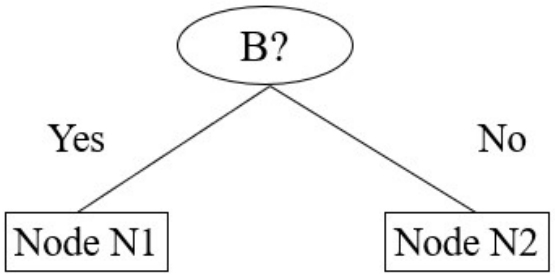
donde, ni = número de registros en el hijo i,
n = número de registros en el nodo padre p.

- Elija el atributo que minimiza el promedio ponderado del índice de Gini de los hijos

El índice de Gini se utiliza en algoritmos de árbol de decisión como CART, SLIQ, SPRINT

Cálculo del índice de Gini de un solo nodo

- Se divide en dos particiones
- Efecto del peso de las particiones:
  - Se buscan particiones más grandes y más puras.



	Parent
C1	**7**
C2	**5**
Gini = 0.486	

Gini(N1)
= 1 – (5/6)² – (1/6)²
= 0.278

Gini(N2)
= 1 – (2/6)² – (4/6)²
= 0.444

	N1	N2
C1	**5**	**2**
C2	**1**	**4**
Gini=0.361		

Weighted Gini of N1 N2
= 6/12 * 0.278 +
6/12 * 0.444
= 0.361

Gain = 0.486 – 0.361 = 0.125

Ejemplo: Cálculo de Impureza

- Sin hacer ninguna partición tenemos que 24 alumnos aprobaron (clase P) y 8 alumnos desaprobaron (clase F)

P	24
F	8

$$i_G(t) = 1 - \left(\frac{24}{32}\right)^2 - \left(\frac{8}{32}\right)^2 = 0.375$$

$$i_E(t) = \frac{24}{32}\log\left(\frac{24}{32}\right) + \frac{8}{32}\log\left(\frac{8}{32}\right) = 0.5623$$

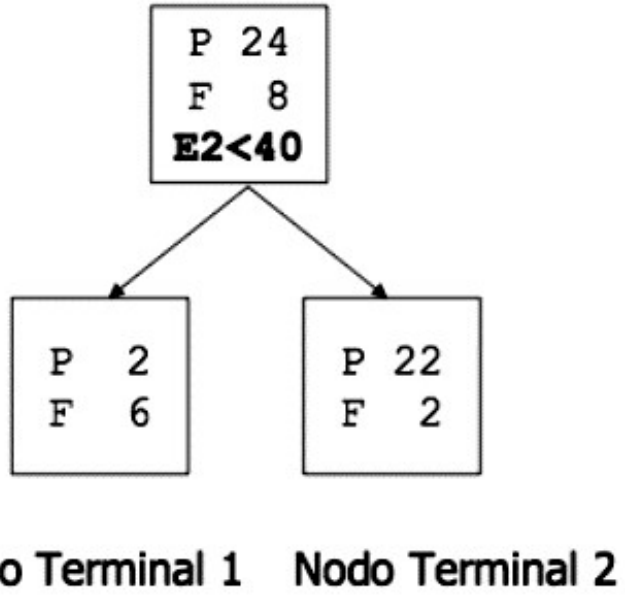
Ejemplo: Cálculo de Impureza

- **Partición 1:** E2<40

$$i_G(1) = 0.3750 \quad i_G(2) = 0.1528$$
$$i_E(1) = 0.5623 \quad i_E(2) = 0.2868$$
$$p(1) = 0.25 \quad p(2) = 0.75$$

- **Impureza del árbol**

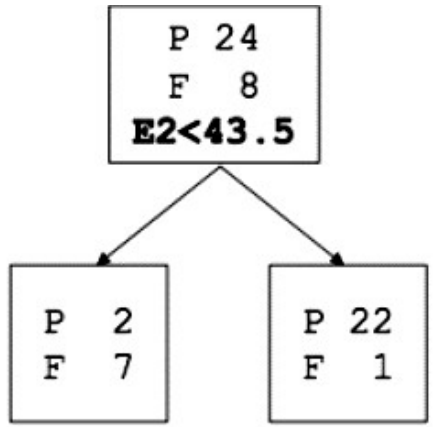
$$I_G(T) = 0.2083$$
$$I_E(T) = 0.3557$$



Ejemplo: Cálculo de Impureza

- **Partición 2:** `E2<43.5`

$$i_G(1) = 0.3457 \quad i_G(2) = 0.0832$$
$$i_E(1) = 0.5297 \quad i_E(2) = 0.1788$$
$$p(1) = 0.28125 \quad p(2) = 0.71875$$



Nodo Terminal 1 Nodo Terminal 2

- **Impureza del árbol**

$$I_G(T) = 0.1570$$
$$I_E(T) = 0.2775$$

- **Se obtiene menor impureza con esta partición**

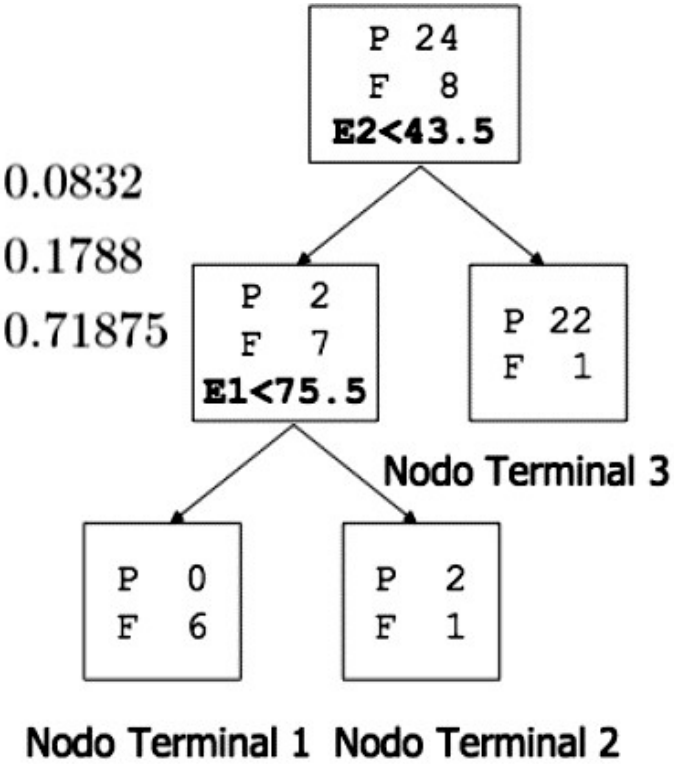
Ejemplo: Cálculo de Impureza

- **Impureza por nodo**

$$i_G(1) = 0 \qquad i_G(2) = 0.4444 \qquad i_G(3) = 0.0832$$
$$i_E(1) = 0 \qquad i_E(2) = 0.6365 \qquad i_E(3) = 0.1788$$
$$p(1) = 0.1875 \quad p(2) = 0.09375 \quad p(3) = 0.71875$$

- **Impureza del árbol**

$$I_G(T) = 0.1014$$
$$I_E(T) = 0.1882$$



Evaluación de las Clases Predichas

- Un método común para describir la performance de la clasificación es la matriz de confusión.
- Esta es un simple tabulación cruzada para las clases observadas y predichas.

Matriz de confusión

Se representa en matriz donde se compara los valores reales versus los predichos

	Predicted: Yes	Predicted: No
Actual: Yes	TP	FN
Actual: No	FP	TN

Matriz de confusión

- Verdaderos positivos (TPs): Los verdaderos positivos son casos en los que predecimos la enfermedad como sí cuando el paciente realmente tiene la enfermedad.
- Negativos verdaderos (TNs): Casos en los que predecimos la enfermedad como no cuando el paciente en realidad no tiene la enfermedad.
- Falsos positivos (FPs):Cuando predecimos la enfermedad como sí, cuando el paciente realmente no tiene la enfermedad. Los FP también se consideran errores de tipo I.
- Falsos negativos (FNs): Cuando predecimos la enfermedad como no, cuando el paciente realmente tiene la enfermedad. Las FN también se consideran errores de tipo II.

la matriz de confusión

Precisión (P)

Cuando se predice que sí, ¿con qué frecuencia es correcto?

$$P = (TP/TP + FP)$$

Recall (R) / sensibilidad / tasa positiva verdadera

Entre los sí reales, ¿qué fracción se predijo como sí?

$$R = (TP/TP + FN)$$

F1 score (F1)

Este es la media armónica de la precisión y el recall. Multiplicando la constante de 2 escala la puntuación a 1 cuando tanto la precisión como el recall son 1

$$F_1 = \frac{2}{\frac{1}{P}+\frac{1}{R}} \quad > \quad F_1 = \frac{2*P*R}{P+R}$$

la matriz de confusión

Especificidad

Entre los números reales, ¿qué fracción se predijo como no?
También equivalente a 1- tasa de falsos positivos

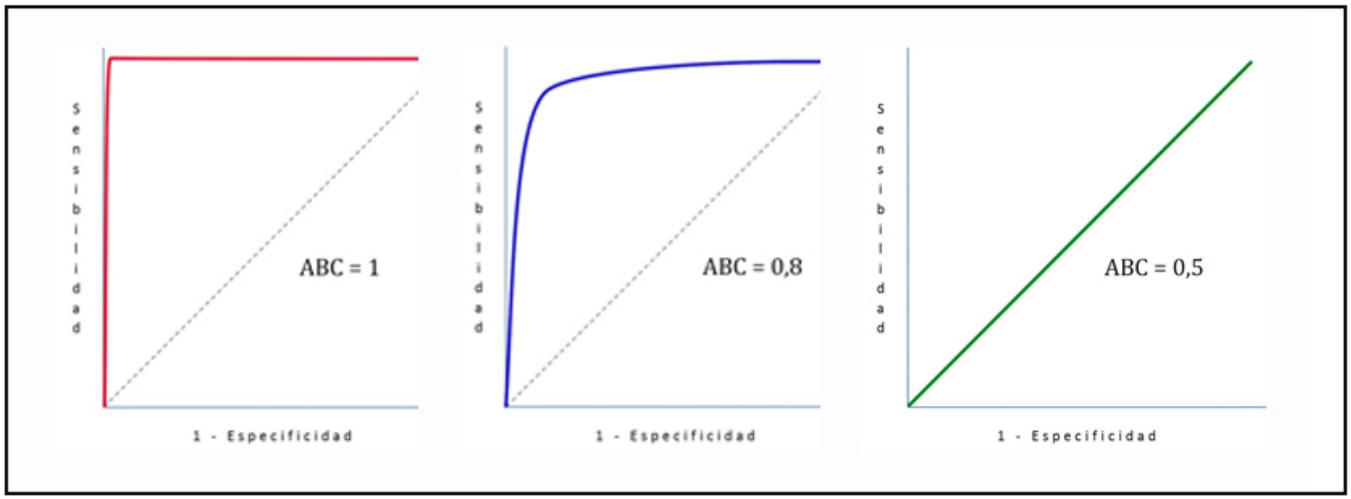
$$(TN/TN + FP)$$

Área bajo curva (ROC)

La curva característica de funcionamiento del receptor se utiliza para trazar entre tasa verdadera positiva (TPR) y tasa de falsos positivos (FPR).

Curva ROC

También conocida como sensibilidad y 1- especificidad. grafico



El área bajo la curva se utiliza para establecer el umbral de probabilidad de corte para clasificar la probabilidad pronosticada en varias clases

La Librería rpart

- En el R se puede utilizar la librería rpart con la función rpart cuya sintaxis es:
- rpart(formula, data,method)
  - Argumentos:
    formula: formula indicando las variable respuesta y las predictoras.
    data: conjunto de datos a ser utilizado.
    se usa "class" para árboles de decisión.

La Librería rpart

- minsplit: fija el número mínimo de observaciones en un nodo para que este sea dividido. Esta opción por defecto es 20.
- minbucket: indica el número mínimo de observaciones en cualquier nodo terminal. Por defecto esta opción es el valor redondeado de minsplit/3.
- cp: parámetro de complejidad. Indica que si el criterio de impureza no es reducido en mas de cp*100 % entonces se para. Por defecto cp=0.01. Es decir, la reducción en la impureza del nodo terminal debe ser de al menos 1 % de la impureza inicial.
- maxdepth: condiciona la profundidad máxima del arbol. Por defecto está establecida como 30.

... Ahora practicamos con dataset