



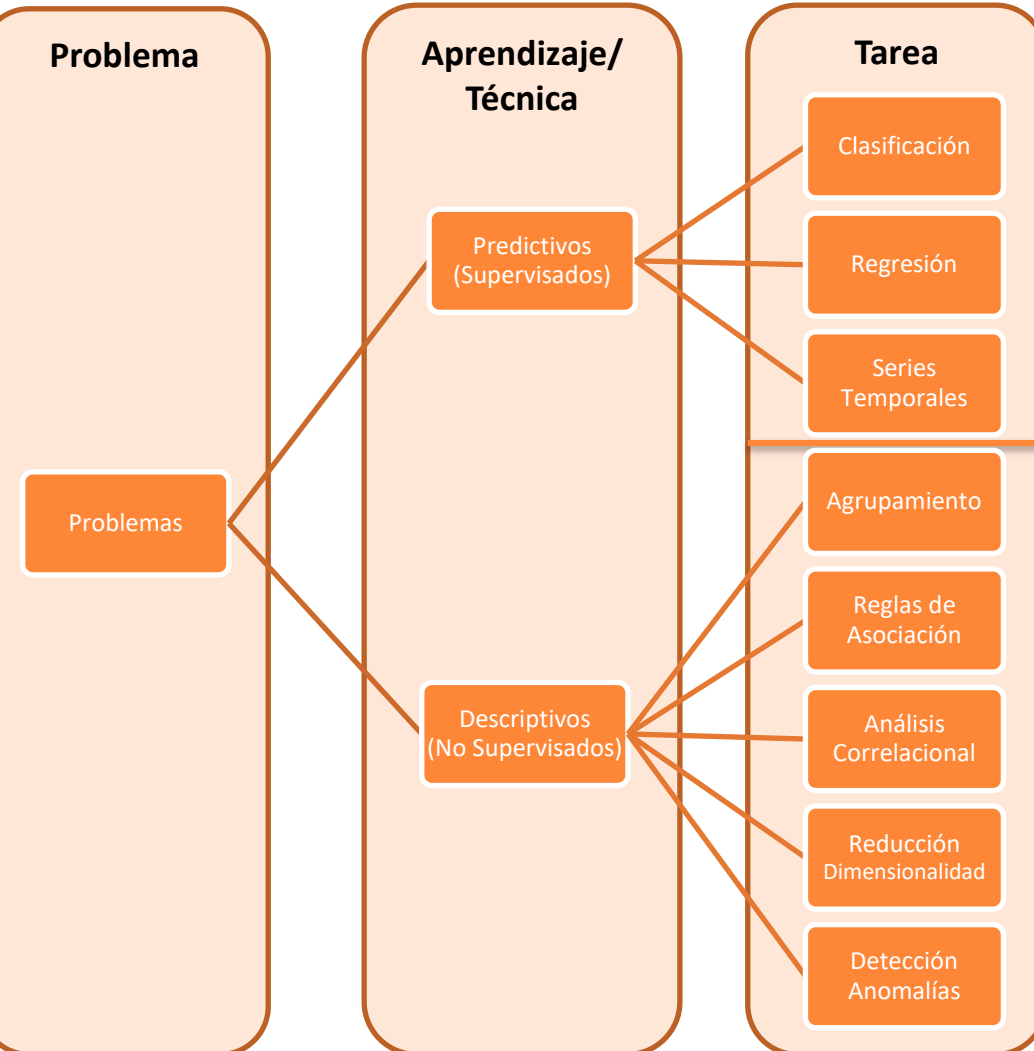
# **UNIVERSIDAD NACIONAL DE LA MATANZA**

## **INTELIGENCIA DE NEGOCIOS**

---

**Tecnologías Inteligentes  
para Explotación de Información**

**Docentes: ING. LORENA R. MATTEO**  
**Autores ppt orig.: Lic. HUGO M. CASTRO / MG. DIEGO BASSO**



Las técnicas de minería de datos son herramientas que facilitan el descubrimiento de conocimiento.

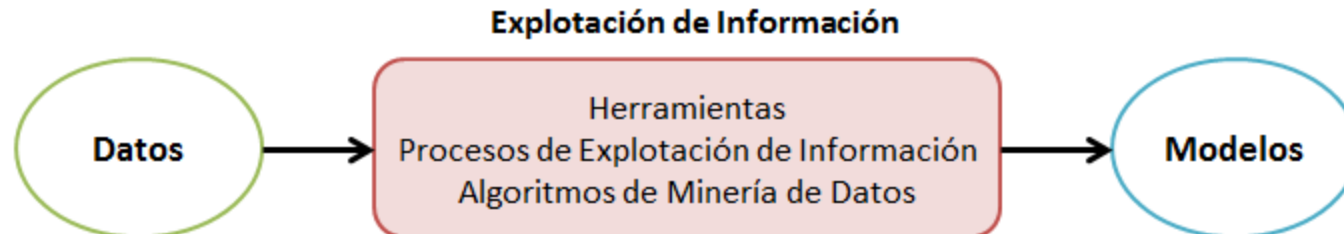


Esta foto de Autor desconocido está bajo licencia [CC BY-SA-NC](#)



# USO DE LAS TECNOLOGÍAS

- ¿Cómo se usan las **tecnologías** para resolver un **problema**?
  - Tecnologías  $\Rightarrow$  Explotación de Información
  - Problema  $\Rightarrow$  Inteligencia de Negocio
- Construcción de modelos para descubrir conocimiento y soporte a la toma de decisiones:
  - Predictivos o Descriptivos
  - Entrenamiento + Prueba
  - Evaluación del modelo construido

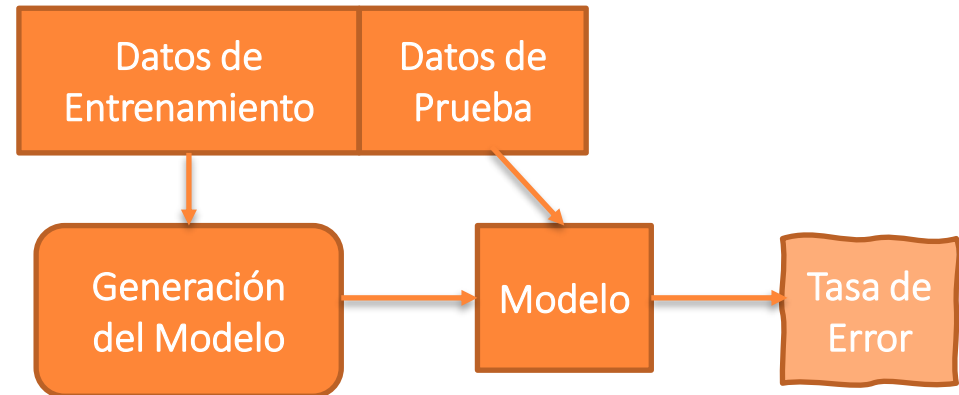




# CONSTRUCCIÓN DE MODELOS DE MINERÍA DE DATOS

## ○ Entrenamiento (Aprendizaje o Inducción)

- Supervisado
- No supervisado



## ○ Prueba

- De los casos históricos disponibles se destina una cantidad para entrenar el modelo y se reserva una porción para probar el modelo
- Se presentan los casos como si fueran nuevos y se coteja la respuesta del modelo con los valores reales

## ○ Evaluación

- Despliegue (Producción): se dispone del modelo apto para su explotación, casos nuevos.



# TECNOLOGÍAS DE EXPLOTACIÓN DE INFORMACIÓN

- Basadas en Análisis Estadístico
  - Análisis de varianza
  - Regresión
  - Prueba Chi-cuadrado
  - Análisis de agrupamiento
  - Análisis de determinantes
  - Series de tiempo
- Basadas en Sistemas Inteligentes
  - Algoritmos de inducción TDIDT
  - Redes Neuronales SOM
  - Redes Bayesianas
  - Redes Neuronales Back-Propagation

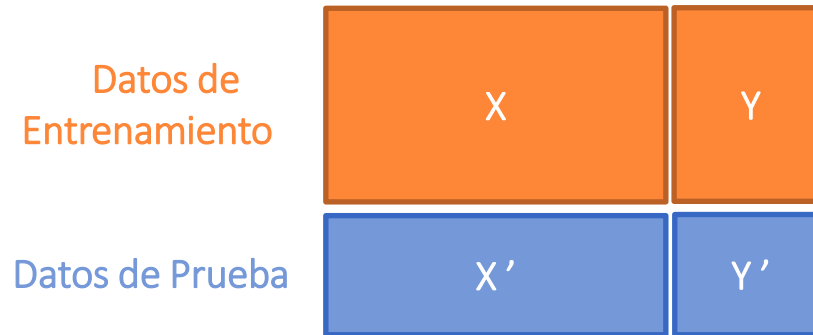


# MODELOS PREDICTIVOS

- Son modelos de aprendizaje supervisado que permiten predecir el resultado de variables de interés a partir de los valores de otras variables.
  - Variables de entrada  $\Rightarrow$  Atributos predictores
  - Variable a predecir  $\Rightarrow$  Atributo clase
- Se tiene un conjunto de *casos de entrenamiento* donde cada caso contiene un conjunto de **atributos** y uno de ellos es la **clase** a clasificar.
- Se separa un conjunto de *casos de prueba* para predecir nuevos casos y probar el modelo.
- Los nuevos casos deben ser asignados a su clase con la máxima exactitud y precisión posible.



# MODELOS PREDICTIVOS



- El entrenamiento busca descubrir las relaciones entre las variables de entrada (X y X') y la variable objetivo (clase) (Y e Y').
- En “producción” se usa ese conocimiento para predecir el valor de la variable objetivo (Y=?) de un nuevo caso no incluido en los datos de entrenamiento ni de prueba.



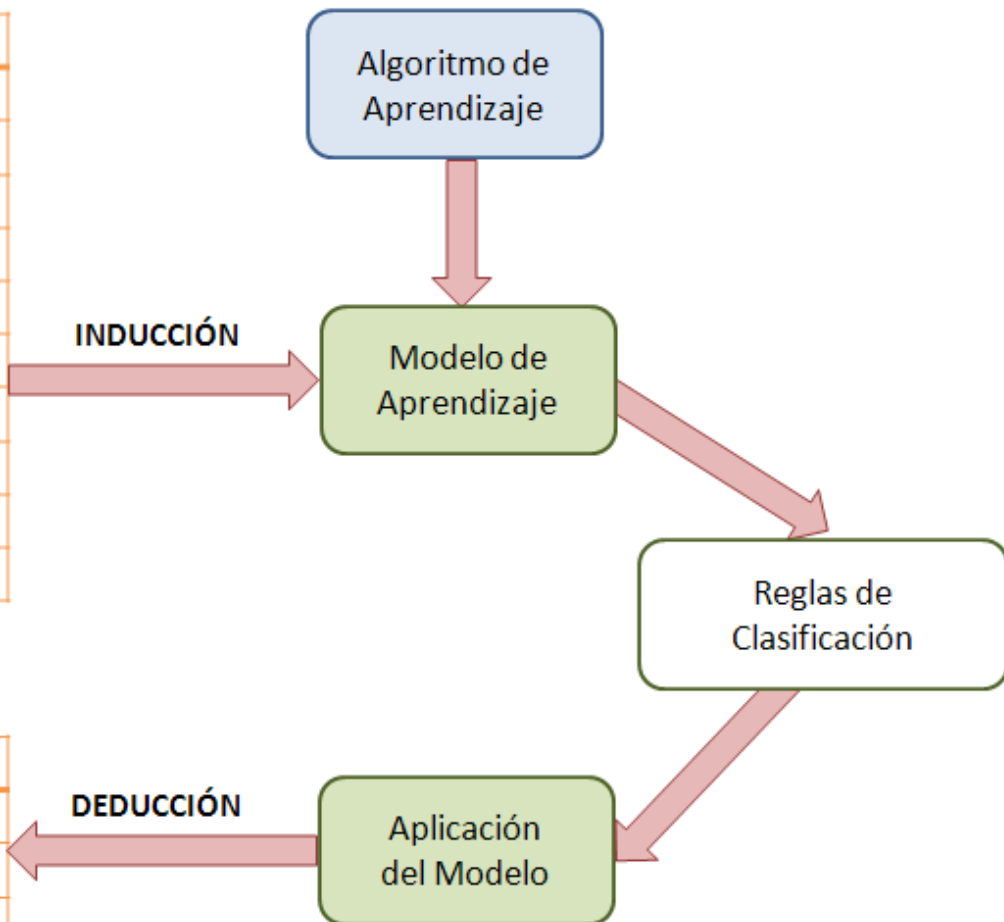
# TAREA DE CLASIFICACIÓN

ATRIB 1	ATRIB 2	ATRIB 3	CLASE
Si	Large	125K	No
No	Medium	100K	No
No	Small	70K	No
Si	Medium	120K	No
No	Large	95K	Si
No	Medium	60K	No
Si	Large	220K	No
No	Small	85K	Si
No	Medium	75K	No
No	Small	90K	Si

Casos de Entrenamiento

ATRIB 1	ATRIB 2	ATRIB 3	CLASE
No	Small	55K	?
Si	Medium	80K	?
Si	Large	110K	?
No	Small	95K	?
No	Large	67K	?

Casos de Prueba







# TECNOLOGÍAS PARA CLASIFICACIÓN

- Árboles de Decisión
  - Algoritmos de inducción TDIDT
  - Métodos basados en reglas
- Redes Bayesianas
  - Naïve-Bayes (Bayes Ingenuo)
- Vecinos más cercanos
  - K-vecinos (CBR)



# ALGORITMOS DE INDUCCIÓN TDIDT

- La familia TDIDT (*Top Down Induction Trees*) pertenece a los métodos inductivos del Aprendizaje Automático que aprenden a partir de ejemplos preclasificados.
  - Atributos **predictores**  $\Rightarrow$  Se particionan en diferentes ramas de acuerdo a los valores que el atributo puede tomar.
    - Pueden ser discretos o continuos.
  - Atributo **clase**  $\Rightarrow$  Decide la clase asignada (variable objetivo)
    - Debe ser discretizado.
- Generan árboles y reglas de decisión a partir de ejemplos preclasificados.



# ALGORITMOS DE INDUCCIÓN TDIDT

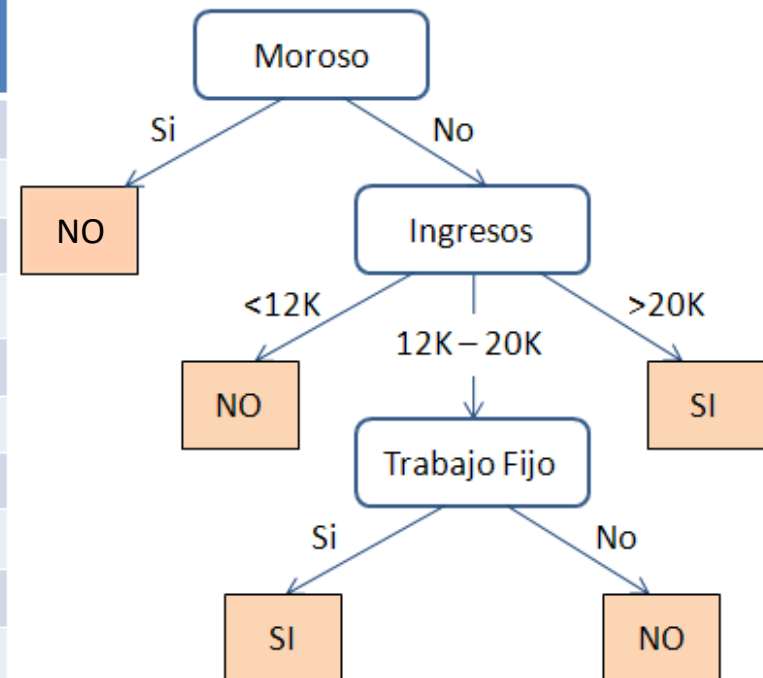
- Se trata de identificar y ubicar en la parte superior del árbol a los atributos que mejor separan los ejemplos o muestras.
- Para encontrar los mejores atributos utiliza la teoría de la información, determinando qué atributo aporta la mayor ganancia de información (o menor pérdida de información) al tomar un determinado valor.
- Algoritmos utilizados ID3, C4.5 y C5



# ALGORITMO TDIDT – EJEMPLO 1

- Presentación intuitiva del proceso de inducción.
- Evaluación de otorgamiento de préstamos a clientes
  - Atributo clase: **Otorgar Préstamo**

Ciente	Moroso	Antigüedad	Ingresos	Trabajo Fijo	Otorgar Préstamo
1	Si	> 5	12K – 20K	Si	No
2	No	< 1	12K – 20K	Si	Si
3	Si	1 - 5	> 20K	Si	No
4	No	> 5	> 20K	No	Si
5	No	< 1	> 20K	Si	Si
6	Si	1 - 5	12K – 20K	Si	No
7	No	1 - 5	> 20K	Si	Si
8	No	< 1	< 12K	Si	No
9	No	> 5	12K – 20K	No	No
10	Si	1 - 5	< 12K	No	No



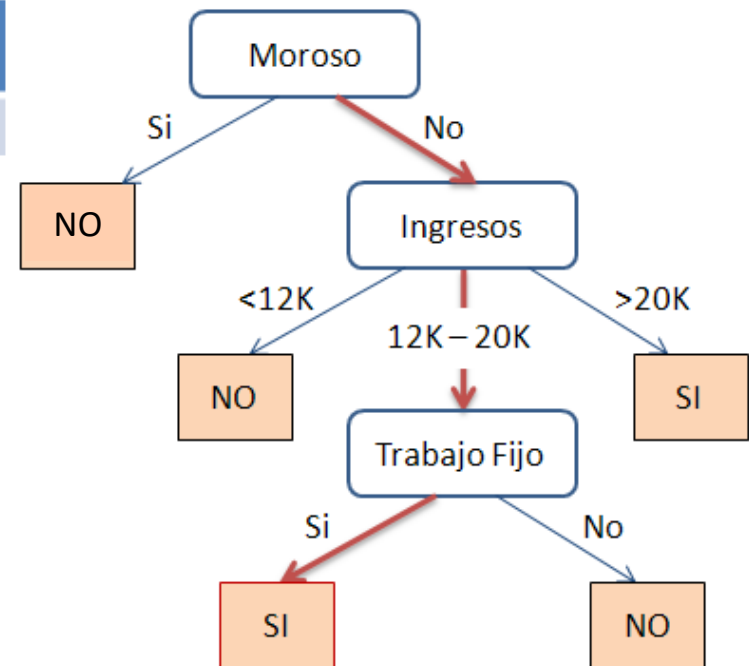
Casos de Entrenamiento



# ALGORITMO TDIDT - APLICACIÓN

## ○ Caso de Prueba

Cliente	Moroso	Antigüedad	Ingresos	Trabajo Fijo	Otorgar Préstamo
11	No	1 - 5	12K – 20K	Si	?

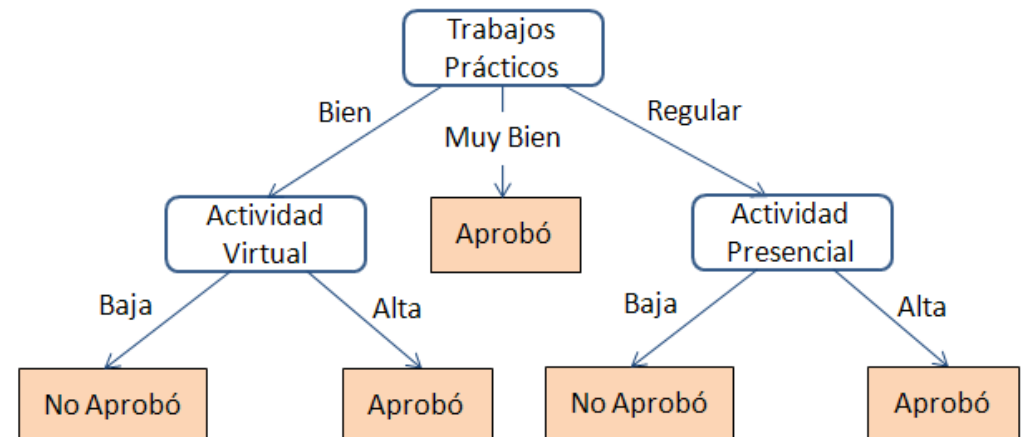




# ALGORITMO TDIDT – EJEMPLO 2

- Predicción de resultados de exámenes
  - Atributo clase: **Resultado Parcial**

Trabajos Prácticos	Actividad Virtual	Actividad Presencial	Resultado Parcial
Bien	Alta	Alta	Aprobó
Bien	Baja	Alta	No Aprobó
Muy Bien	Alta	Alta	Aprobó
Regular	Alta	Alta	Aprobó
Regular	Alta	Baja	No Aprobó
Regular	Baja	Baja	No Aprobó
Muy Bien	Baja	Baja	Aprobó
Bien	Baja	Baja	No Aprobó
Bien	Alta	Baja	Aprobó
Regular	Baja	Baja	No Aprobó
Bien	Alta	Baja	Aprobó
Muy Bien	Alta	Alta	Aprobó
Regular	Baja	Baja	Aprobó
Regular	Alta	Alta	Aprobó





# ALGORITMO TDIDT - APRENDIZAJE

## ○ Construcción de reglas del tipo IF-THEN

$R_1$ : IF Trabajos Prácticos = 'Muy Bien' THEN Resultado Parcial = 'Aprobó'

$R_2$ : IF (Trabajos Prácticos = 'Bien') AND (Actividad Virtual = 'Baja')  
THEN Resultado Parcial = 'No Aprobó'

$R_3$ : IF (Trabajos Prácticos = 'Bien') AND (Actividad Virtual = 'Alta')  
THEN Resultado Parcial = 'Aprobó'

$R_4$ : IF (Trabajos Prácticos = 'Regular') AND (Actividad Presencial = 'Baja')  
THEN Resultado Parcial = 'No Aprobó'

$R_5$ : IF (Trabajos Prácticos = 'Regular') AND (Actividad Presencial = 'Alta')  
THEN Resultado Parcial = 'Aprobó'



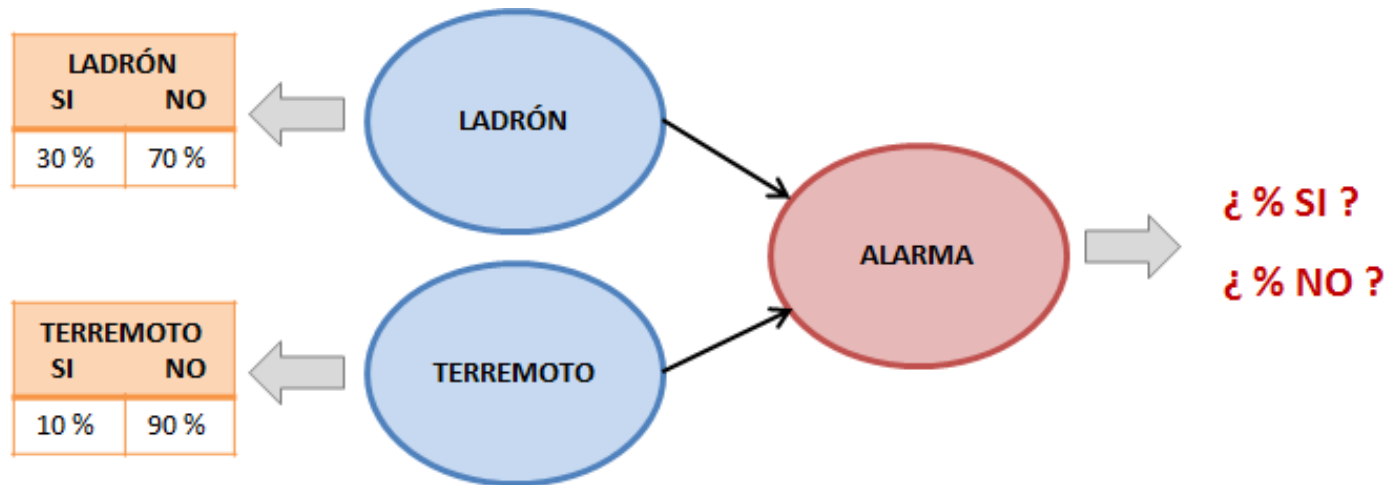
# REDES BAYESIANAS

- Una red bayesiana es un grafo acíclico dirigido compuesto de nodos y arcos.
- Los nodos representan las variables aleatorias (o atributos).
- Los arcos representan dependencias probabilísticas de cada variable.
  - El arco entre dos variables significa una influencia directa de una variable sobre otra.
  - Probabilidad condicional (Teorema de Bayes).
- Representan la relación causa-efecto entre atributos.
- Dan a una medida cuantitativa y probabilística de la importancia de los atributos en un problema de clasificación de clases.





# REDES BAYESIANAS – EJEMPLO

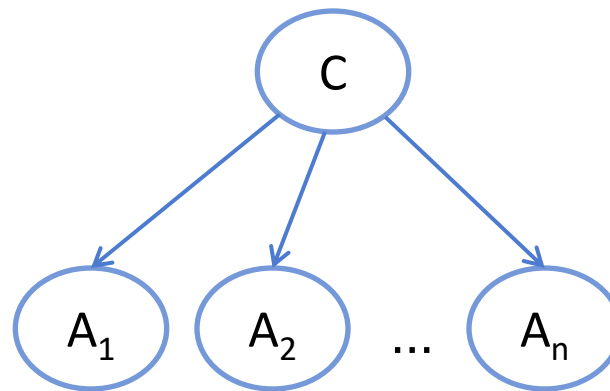


- Las variables *Ladrón* y *Terremoto* son causas para que se dispare una Alarma.
  - Existe una probabilidad a priori para Ladrón y Terremoto.
  - ¿Cuál es la probabilidad de que suene o no la alarma?
- *Ladrón* y *Terremoto* son condicionalmente independientes entre sí dada la variable *Alarma*.



# CLASIFICADOR BAYESIANO NAÏVE-BAYES

- Considera que cada atributo predictor  $A_i$  y el atributo clase  $C$  son variables aleatorias.
- Las relaciones de dependencia entre los atributos  $A_i$  son condicionalmente independientes entre sí dado el atributo clase  $C$ .



- Dado un registro con atributos  $A_1, A_2, \dots, A_n$  el objetivo es predecir la clase  $C$ .
- Se busca encontrar el valor de  $C$  que maximice la probabilidad  $p(C/A_1, A_2, \dots, A_n)$ .



# REDES BAYESIANAS

- Obtener una red bayesiana a partir de datos, es un proceso de aprendizaje.
  - Aprendizaje Estructural
  - Aprendizaje Paramétrico
- Proceso de inferencia
  - Predicciones a partir de observaciones



# REDES BAYESIANAS – EJEMPLO

- Se tienen los siguientes datos:

	Ambiente	Temperatura	Humedad	Viento	Juega Tenis
1	Soleado	Alta	Alta	Leve	No
2	Soleado	Alta	Alta	Fuerte	No
3	Nublado	Alta	Alta	Leve	Si
4	Lluvioso	Media	Alta	Leve	No
5	Lluvioso	Baja	Normal	Fuerte	No
6	Lluvioso	Baja	Normal	Fuerte	No
7	Nublado	Baja	Normal	Leve	Si
8	Soleado	Media	Alta	Leve	Si
9	Soleado	Baja	Normal	Leve	Si
10	Lluvioso	Media	Normal	Leve	No
11	Soleado	Media	Normal	Fuerte	Si
12	Nublado	Media	Alta	Fuerte	Si
13	Nublado	Alta	Normal	Leve	Si
14	Lluvioso	Media	Alta	Fuerte	No



## REDES BAYESIANAS – EJEMPLO

- Queremos saber si se jugará al tenis bajo las siguientes condiciones:

Ambiente	Temperatura	Humedad	Viento	Juega Tenis
Soleado	Baja	Alta	Fuerte	?

- El atributo clase a predecir es **Juega Tenis** cuyos valores serán **Si** o **No**.
- El nuevo caso será clasificado como clase  $C_j$  si

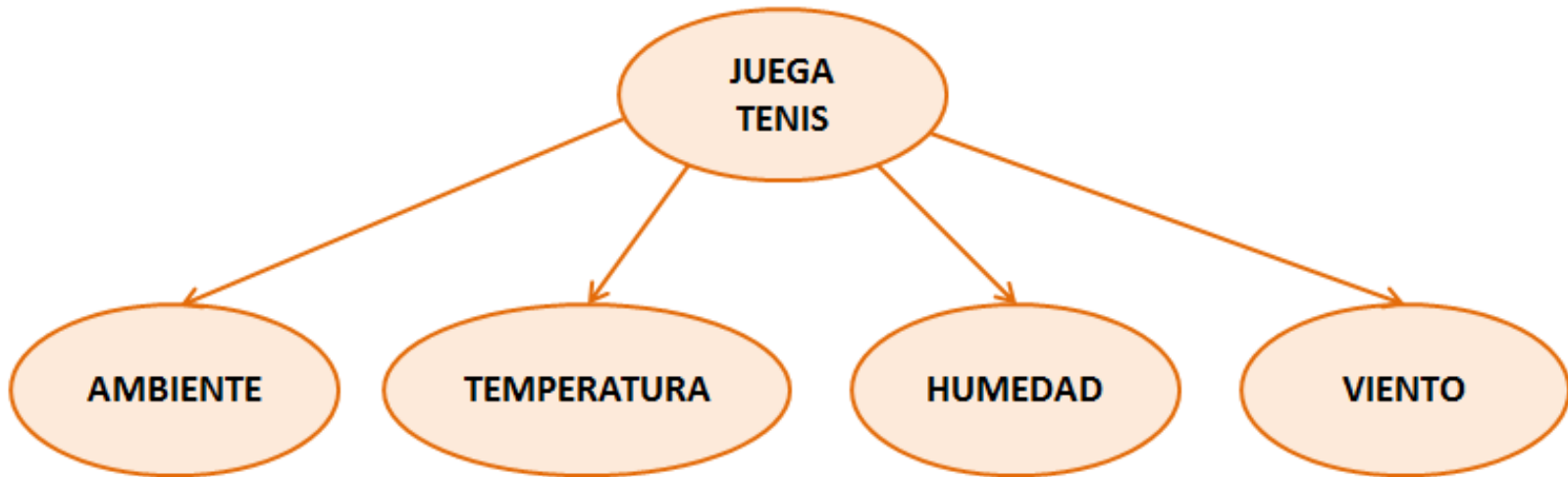
$$P(C_j) \prod_{i=1}^n P(A_i | C_j) \text{ es máximo.}$$



# REDES BAYESIANAS – EJEMPLO

## Aprendizaje Estructural

- Relaciones de dependencia e independencia





# REDES BAYESIANAS – EJEMPLO

## Aprendizaje Paramétrico

- Determinar probabilidades a priori de cada clase y las probabilidades condicionales.
- Analizando los 14 casos tenemos:

	Valores que toma	Cantidad de Casos	% casos totales
Ambiente	Soleado	5	35,7%
	Nublado	4	28,6%
	Lluvioso	5	35,7%
Temperatura	Alta	4	28,6%
	Media	6	42,8%
	Baja	4	28,6%
Humedad	Alta	7	50%
	Normal	7	50%
Viento	Leve	8	57,2%
	Fuerte	6	42,8%

Casos **Juega Tenis = Si** = 7

Casos **Juega Tenis = No** = 7

$P(\text{Juega Si}) = 0,5 = 50\%$

$P(\text{Juega No}) = 0,5 = 50\%$



## REDES BAYESIANAS – EJEMPLO

- Desglosando los casos según si juegan o no al tenis:

Cantidad Casos			
	Valores que toma	Clase = Juega Tenis	
		Si	No
Ambiente	Soleado	3	2
	Nublado	4	0
	Lluvioso	0	5
Temperatura	Alta	2	2
	Media	3	3
	Baja	2	2
Humedad	Alta	3	4
	Normal	4	3
Viento	Leve	5	3
	Fuerte	2	4

Casos totales = 14

Casos  $\text{Juega Tenis} = \text{Si} = 7$

Casos  $\text{Juega Tenis} = \text{No} = 7$





# REDES BAYESIANAS – EJEMPLO

- Obtenemos las probabilidades condicionales:

## Probabilidades

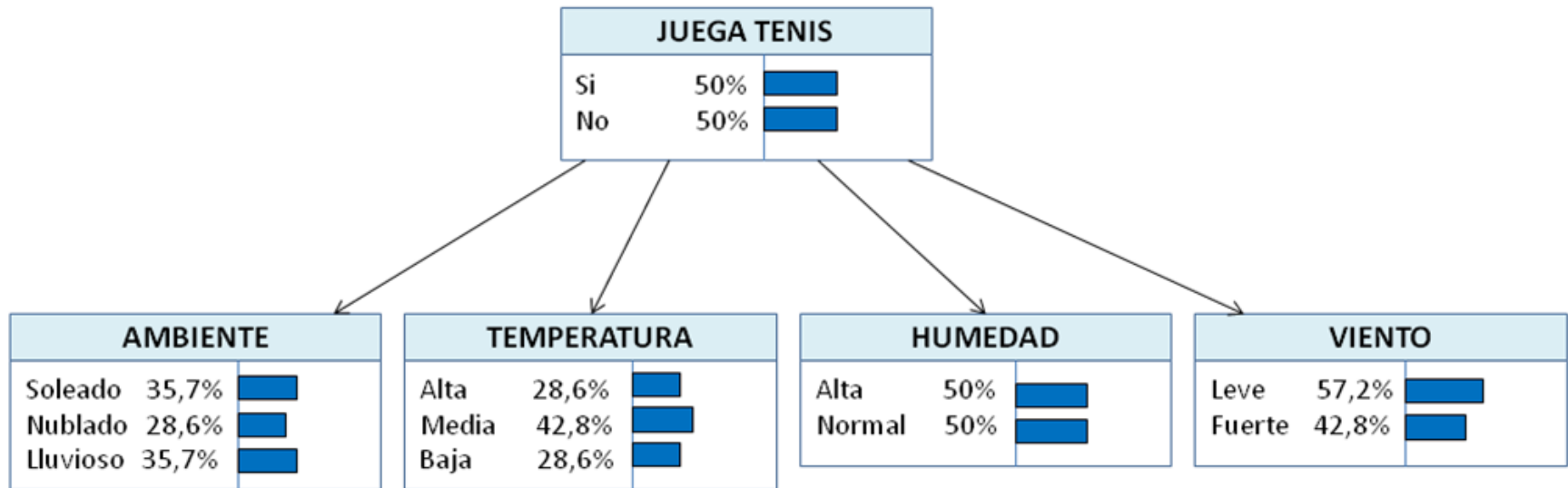
	Valores que toma	Clase = Juega Tenis	
		Si	No
Ambiente	Soleado	$3/7 = 42,8\%$	$2/7 = 28,6\%$
	Nublado	$4/7 = 57,2\%$	0
	Lluvioso	0	$5/7 = 71,4\%$
Temperatura	Alta	$2/7 = 28,6\%$	$2/7 = 28,6\%$
	Media	$3/7 = 42,8\%$	$3/7 = 42,8\%$
	Baja	$2/7 = 28,6\%$	$2/7 = 28,6\%$
Humedad	Alta	$3/7 = 42,8\%$	$4/7 = 57,2\%$
	Normal	$4/7 = 57,2\%$	$3/7 = 42,8\%$
Viento	Leve	$5/7 = 71,4\%$	$3/7 = 42,8\%$
	Fuerte	$2/7 = 28,6\%$	$4/7 = 57,2\%$

Casos totales = 14



# REDES BAYESIANAS – EJEMPLO

## Proceso de Inferencia

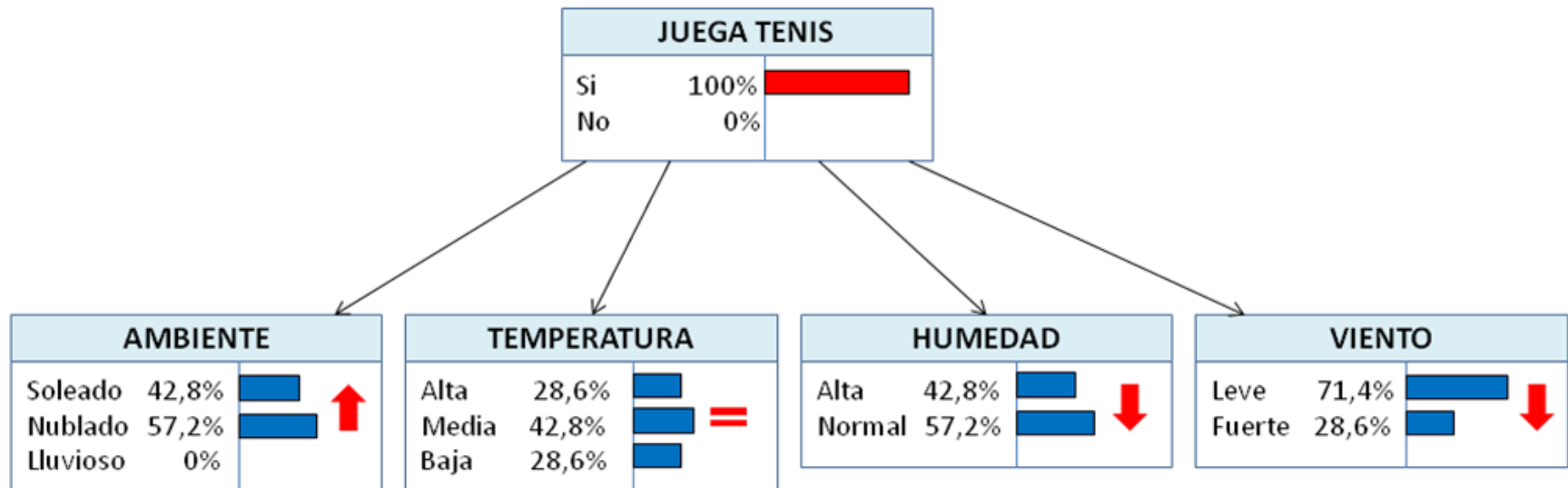




# REDES BAYESIANAS – EJEMPLO

## Proceso de Inferencia

- Juega Tenis = Si

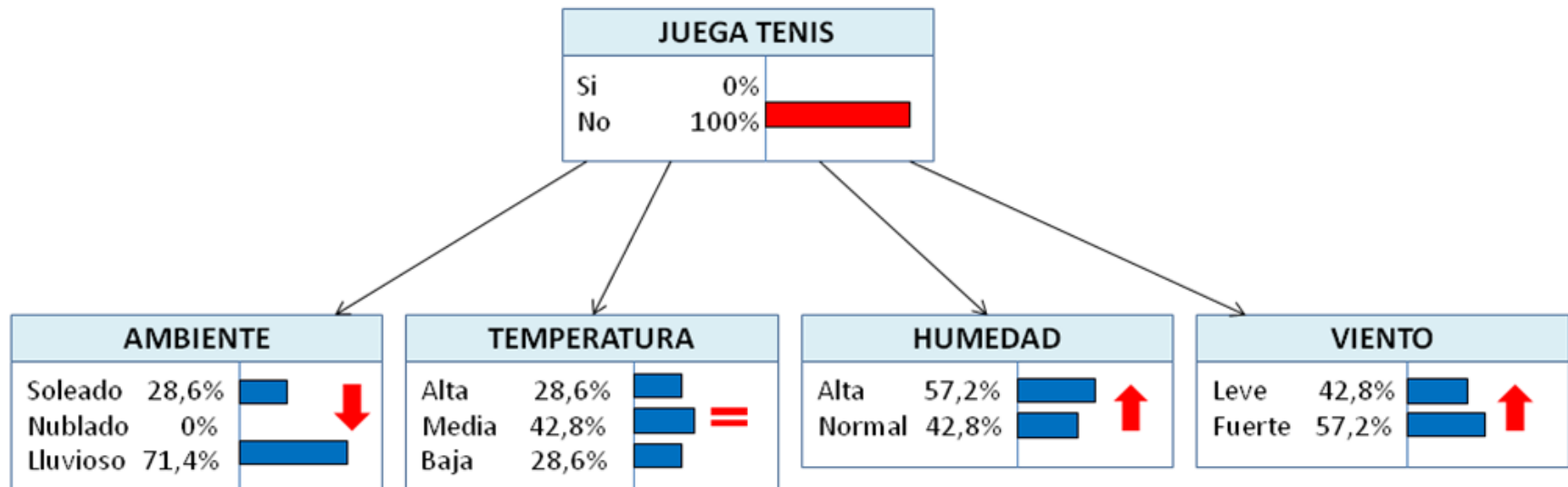




# REDES BAYESIANAS – EJEMPLO

## Proceso de Inferencia

- Juega Tenis = No





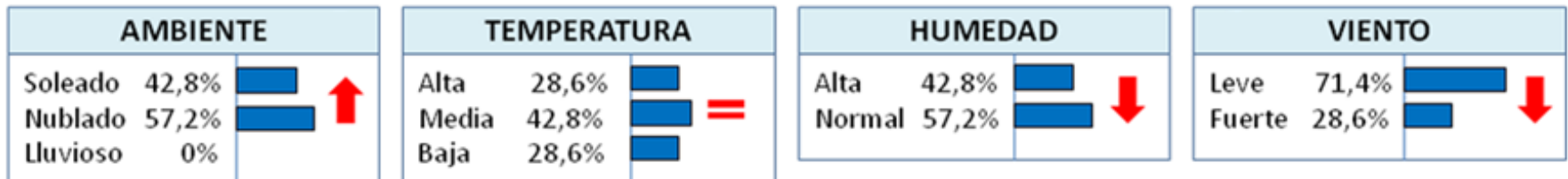
# REDES BAYESIANAS – EJEMPLO

## ○ Predicción a realizar:

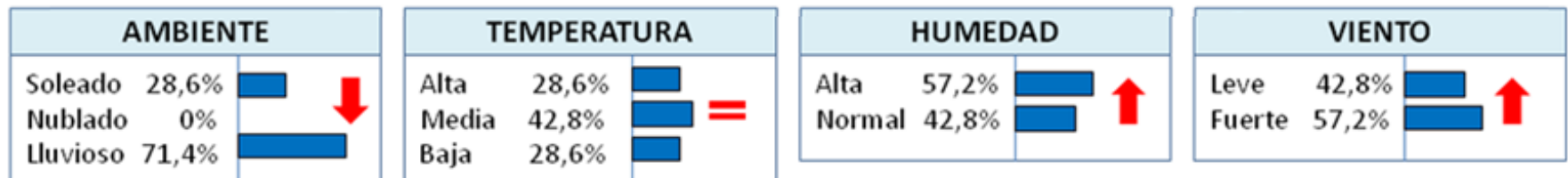
Ambiente	Temperatura	Humedad	Viento	Juega Tenis
Soleado	Baja	Alta	Fuerte	?

$$P(\text{Juega Si}) = 0,5$$
$$P(\text{Juega No}) = 0,5$$

- $P(\text{Juega Si}) = 0,428 \times 0,286 \times 0,428 \times 0,286 \times 0,5 = 0,0075$



- $P(\text{Juega No}) = 0,286 \times 0,286 \times 0,572 \times 0,572 \times 0,5 = 0,0133$





## REDES BAYESIANAS – EJEMPLO

### ○ Normalizando:

- $P(\text{Juega}_{\text{Si}}) = 0,0075 / (0,0075 + 0,0133) = 36\%$
- $P(\text{Juega}_{\text{No}}) = 0,0133 / (0,0075 + 0,0133) = \mathbf{64\%}$

Ambiente	Temperatura	Humedad	Viento	Juega Tenis
Soleado	Baja	Alta	Fuerte	?

- El clasificador va a predecir que no se juega al tenis con una probabilidad del 64%.



# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

## Pautas para la evaluación

- Exactitud
  - No hay un algoritmo que sea siempre mejor que otro u otros
- Precisión, Recall (Exhaustividad/Sensibilidad/TPR y Especificidad/TNR) + F1-Score, Kappa (Cohen)
  - Útiles cuando el dataset no está balanceado, dan una mejor idea de la calidad del modelo.
- Interpretabilidad
  - Facilidad para interpretar los resultados
- Velocidad
  - Entrenamiento
  - Producción



# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

## Métricas de Evaluación

- Se focalizan en analizar la capacidad de predicción y clasificación de clases del modelo construido.
- **Matriz de Confusión:** Permite comparar el resultado obtenido a partir del modelo predictivo construido con los resultados de los datos de prueba del modelo.
- Métricas utilizadas (en entrenamiento y prueba)
  - Exactitud del modelo
  - Precisión del modelo
  - Recall (Exhaustividad/Sensibilidad/TPR y Especificidad/TNR)
  - F1-Score + Coeficiente Kappa
- Otras métricas utilizadas
  - Cobertura de una regla
  - Precisión de una regla

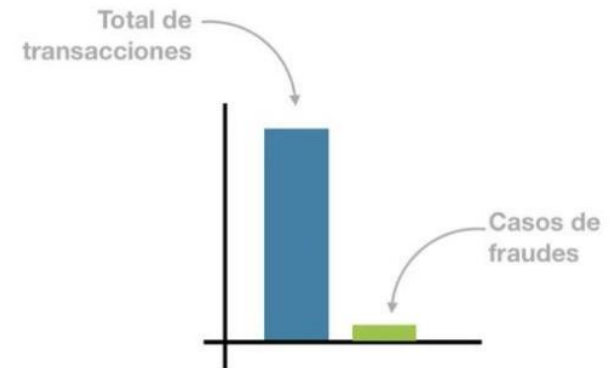




# MODELOS DE CLASIFICACIÓN

## Dataset desbalanceado

- Es aquel que tiene muchas instancias de una clase y muy pocas de la otra, dificultando así el entrenamiento.
- Algo de desbalance de clases es de esperar y no afecta a nuestro análisis; pero bajo ciertas problemáticas, suelen haber datasets muy desbalanceados:
  - Detección de fraudes.
  - Diagnóstico médico.
  - Falla en cadena de producción.
- Atención en:
  - Cómo se entrenan los modelos.
  - Qué métricas se usan para evaluarlos: las siguientes son útiles cuando el dataset no está balanceado, dan una mejor idea de la calidad del modelo,
    - **Precisión / Recall** (Exhaustividad/Sensibilidad/TPR y Especificidad/TNR) / **F1-Score** / **Kappa (Cohen)**
    - Por el contrario, la **Exactitud** no es buena cuando el dataset está desbalanceado.





# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

- Estructura de una **matriz de confusión** de 2 clases.

		Clase Clasificada		
		Clase A	Clase B	Total
Clase Real	Clase A	Nº casos clasificados como A y son de clase A (NCVA)	Nº casos clasificados como B pero son de clase A (NCFB)	Total de casos de la clase A
	Clase B	Nº casos clasificados como A pero son de clase B (NCFA)	Nº casos clasificados como B y son de clase B (NCVB)	Total de casos de la clase B
	Total	Total de casos clasificados como clase A	Total de casos clasificados como clase B	Número total de casos (NTC)

- Las métricas NCVA y NCVB representan los valores clasificados correctamente por el modelo (V: Verdadero)
- Las métricas NCFA y NCFB representan los errores (la confusión) entre las clases (F: Falso)



# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

- Forma más usual de encontrar un Matriz de Confusión, incluyendo métricas para evaluar data sets desbalanceados

		Clase Clasificada			
		Clase A (Positive)	Case B (Negative)	Total	
Clase Real	Clase A (Positive)	TP (VP ó NCVA)	FN (NCFB)	Total de Casos Reales de la Clase A	Exahustividad (Recall, Sensibilidad o TPR) (% casos positivos detectados) $= TP / (TP+FN)$
	Clase B (Negative)	FP (NCFA)	TN (VN ó NCVB)	Total de Casos Reales de la Clase B	Especificidad (TNR) (% casos negativos detectados) $= TN / (TN+FP)$
Total		Total de Casos Clasificados como Clase A	Total de Casos Clasificados como Clase B	Nro Total de casos (NTC)	Exactitud (% predicciones positivas correctas) no es util en DS desbalanceados $= (TN + TP) / (TN + TP + FN + FP)$
		Precisión (A=P) (% predicciones positivas correctas) $= TP / (TP + FP)$	Precisión (B=N) (% predicciones negativas correctas) $= TN / (TN + FN)$		F1-Score (A=P) $= 2*((Precisión* Recall) / (Precisión+Recall))$
					F1-Score (B=N) $= 2*((Precisión* Especificidad) / (Precisión + Especificidad))$
					Coficiente Kappa $= 2*(TP * TN - FN * FP) / (TP + FP) * (FP + TN) + (TP + FN) * (FN + TN)$

- Fuente: Mejora del artículo: Telefónica Think Big / Empresas - Cómo interpretar la matriz de confusión: ejemplo práctico Paloma Recuero de los Santos



# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

- **Exactitud del modelo:** Proporción de casos clasificados correctamente respecto del número total de casos utilizados. Evalúa la capacidad de generalización del modelo para predecir y clasificar nuevos casos.

$$\text{Exactitud (M)} = \frac{\sum_{i=1}^n \text{NCV}_i}{\text{N}^\circ \text{ casos usados}}$$

- ALTA exactitud  $\Rightarrow$  Clasificaciones correctas  $\geq 70\%$  casos.
- BAJA exactitud  $\Rightarrow$  Clasificaciones correctas  $< 70\%$  casos. Modelo poco confiable.

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$



# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

- **Precisión del modelo:** proporción de casos reales de una clase respecto del total de casos clasificados por el modelo en esa misma clase. Evalúa la efectividad del modelo para clasificar casos a una clase particular.

$$\text{Precisión } (C_i) = \frac{NCV_i}{\text{Total Casos Clasificados Positivos}} = \frac{NCV_i}{NCV_i + NCF_i}$$

- ALTA precisión  $\Rightarrow$  Modelo efectivo para predecir y clasificar nuevos casos.
- Precisión MEDIA  $\Rightarrow$  Modelo inestable. Posible confusión en clasificación y predicción.
- BAJA precisión  $\Rightarrow$  El modelo confunde las clases. Modelo poco efectivo.

		predicción	
		0	1
realidad	0	TN	FP
	1	FN	TP

$$\text{precision}_{(0)} = \frac{TN}{TN + FN}$$

$$\text{precision}_{(1)} = \frac{TP}{TP + FP}$$



# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

- **Exhaustividad (Recall / Sensibilidad / TPR)**: tasa de verdaderos positivos (True Positive Rate) ó TP; es decir la proporción entre los casos positivos bien clasificados por el modelo, respecto a todos los elementos que en realidad son positivos.

$$\text{Exhaustividad} = \frac{NCV_A}{\text{Total Casos Reales Positivos}} = \frac{NCV_A}{NCV_A + NCF_B}$$

- Expresa qué tan bien el modelo es capaz de detectar a la clase positiva.

$$\text{recall} = \frac{TP}{TP + FN}$$

		predicción	
		0	1
realidad	0	TN	FP
	1	FN	TP

Exhaustividad (recall)



# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

- **Especificidad (TNR)**: tasa de verdaderos negativos, (True Negative Rate) ó TN, es decir la proporción entre los casos negativos bien clasificados por el modelo, respecto a todos los elementos que en realidad son negativos.

$$\text{Especificidad} = \frac{NCV_B}{\text{Total Casos Reales Negativos}} = \frac{NCV_B}{NCF_A + NCV_B}$$

- Si lo que nos interesa es identificar los verdaderos negativos, (evitar falsos positivos) debemos elegir especificidad alta.

$$\text{Especificidad} = \frac{TN}{(TN + FP)}$$

		predicción	
		0	1
realidad	0	TN	FP
	1	FN	TP



# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

- **F1-Score:** se utiliza para combinar las medidas de precisión y Recall en un sólo valor. Esto es práctico porque hace más fácil el poder comparar el rendimiento combinado de la precisión y la exhaustividad entre varias soluciones.
- F1 se calcula haciendo la media armónica entre la precisión y la exhaustividad:

$$F1 - Score_{A(P)} = 2 \left( \frac{(Precision_{A(P)} * Recall_{Sensibilidad})}{(Precision_{A(P)} + Recall_{Sensibilidad})} \right)$$

$$F1 - Score_{B(N)} = 2 \left( \frac{(Precision_{B(N)} * Recall_{Especificidad})}{(Precision_{B(N)} + Recall_{Especificidad})} \right)$$





# MODELOS DE CLASIFICACIÓN - EVALUACIÓN

- **Coeficiente Kappa - Concordancia entre Predicciones y Realidad:** Este es una métrica que mide el grado de acuerdo entre las predicciones de un modelo y la realidad, más allá del simple azar.
- **Interpretación de Valores Kappa:** Los valores oscilan entre -1 y 1, donde 0 indica un acuerdo únicamente por azar, y 1 representa un acuerdo perfecto entre predicciones y realidad.
- Es especialmente útil cuando las clases en una tarea de clasificación tienen diferentes prevalencias, ya que ofrece una mejor evaluación que la simple precisión.

$$k = \frac{2 * (TP * TN - FN * FP)}{(TP + FP) * (FP + TN) + (TP + FN) * (FN + TN)}$$

Valoración del coeficiente kappa  
(Landis y Koch, 1977)

Coeficiente kappa	Fuerza de la concordancia
0,00	Pobre ( <i>Poor</i> )
0,01 - 0,20	Leve ( <i>Slight</i> )
0,21 - 0,40	Aceptable ( <i>Fair</i> )
0,41 - 0,60	Moderada ( <i>Moderate</i> )
0,61 - 0,80	Considerable ( <i>Substantial</i> )
0,81 - 1,00	Casi perfecta ( <i>Almost perfect</i> )



# EVALUACIÓN - RESUMEN

## Exactitud: (modelo)

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

## Tasa Error: (modelo)

$$Error = 1 - Exactitud$$

## Precisión: (para cada clase)

$$precision_{(0)} = \frac{TN}{TN + FN}$$

$$precision_{(1)} = \frac{TP}{TP + FP}$$

		predicción	
		0	1
realidad	0	TN	FP
	1	FN	TP

## Exhaustividad: (para cada clase)

(Recall/Sensibilidad/ TPR)

$$recall = \frac{TP}{TP + FN}$$

		predicción	
		0	1
realidad	0	TN	FP
	1	FN	TP

## Especificidad: (para cada clase)

(TNR)

$$Especificidad = \frac{TN}{(TN + FP)}$$

		predicción	
		0	1
realidad	0	TN	FP
	1	FN	TP

## F1-Score: (para cada clase)

$$F1 - Score_{A(P)} = 2 \left( \frac{(Precision_{A(P)} * Recall_{Sensibilidad})}{(Precision_{A(P)} + Recall_{Sensibilidad})} \right) \quad F1 - Score_{B(N)} = 2 \left( \frac{(Precision_{B(N)} * Recall_{Especificidad})}{(Precision_{B(N)} + Recall_{Especificidad})} \right)$$

## Kappa: (modelo)

$$k = \frac{2 * (TP * TN - FN * FP)}{(TP + FP) * (FP + TN) + (TP + FN) * (FN + TN)}$$



# MODELOS DE CLASIFICACIÓN - EJEMPLO

- Consideremos una matriz de confusión con 900 casos de clientes que pueden o no cerrar sus cuentas bancarias:
  - Clase SI – Se va del banco
  - Clase No – No se va del banco

		Clase Clasificada	
		Si	No
Clase Real	Si	455	29
	No	32	384
		487	413

**839** predicciones correctas

**61** predicciones incorrectas

- **Exactitud (M)** =  $(455+384)/(455+384+29+32) = 93,2\%$
- **Precisión ( $C_{Si}$ )** =  $455/(455+32) = 93,4\%$
- **Precisión ( $C_{No}$ )** =  $384/(29+384) = 92,9\%$







# MODELOS DE CLASIFICACIÓN - EJEMPLO

		Clase Clasificada		Total			
		Clase A (Positive)	Case B (Negative)				
Clase Real	Clase A (Positive)	455	29	Total de Casos Reales de la Clase A	484	Exhaustividad (Recall o TPR)	$= TP / (TP + FN)$ 94,01%
	Clase B (Negative)	32	384	Total de Casos Reales de la Clase B	416	Especificidad (TNR)	$= TN / (TN + FP)$ 92,31%
	Total	Total de Casos Clasificados como Clase A	Total de Casos Clasificados como Clase B	Nro Total de casos (NTC)		Exactitud	$= (TN + TP) / (TN + TP + FN + FP)$ 93,22%
		487	413			F1-Score (A=P)	$= 2 * ((Precisión * Recall) / (Precisión + Recall))$ 92,64%
		Precisión (A=P)	Precisión (B=N)			F1-Score (B=N)	$= 2 * ((Precisión * Especificidad) / (Precisión + Especificidad))$ 92,64%
		$= TP / (TP + FP)$	$= TN / (TN + FN)$			Coeficiente Kappa	$= 2 * (TP * TN - FN * FP) / (TP + FP) * (FP + TN) + (TP + FN) * (FN + TN)$ 86,36%
		93,43%	92,98%				

- No todos los errores tienen el mismo costo para el banco.
  - El error en los falsos **No** (29 casos) es mucho más costoso para el banco ya que no se va a tomar ninguna acción y el cliente se va a ir.
  - El banco puede asignar un valor de costo a cada una de las celdas que representan un error en la clasificación para poder comparar soluciones de modelos.



# MODELOS DE CLASIFICACIÓN

		Actual					
Predicha	Positivo	VP	FP	Positivo	Negativo	Positivo	Negativo
		 ¡Eres un gato!	 ¡Eres un gato!	VP	FP	VP	FP
	Negativo	FN	VN	Negativo		Negativo	
		 ¡No eres un gato!	 ¡No eres un gato!	FN	VN	FN	VN

## Recall

De todas las clases positivas cuantas se predijo correctamente

$$\frac{VP}{VP + FN}$$

$$\frac{VN}{VN + FP}$$

## Precisión

De todas las positivas que se han predicho correctamente cuántas son realmente positivas

$$\frac{VP}{VP + FP}$$

$$\frac{VN}{VN + FN}$$

## Accuracy

De todas las clases cuantas se predijeron correctamente

$$\frac{VP + VN}{Total}$$

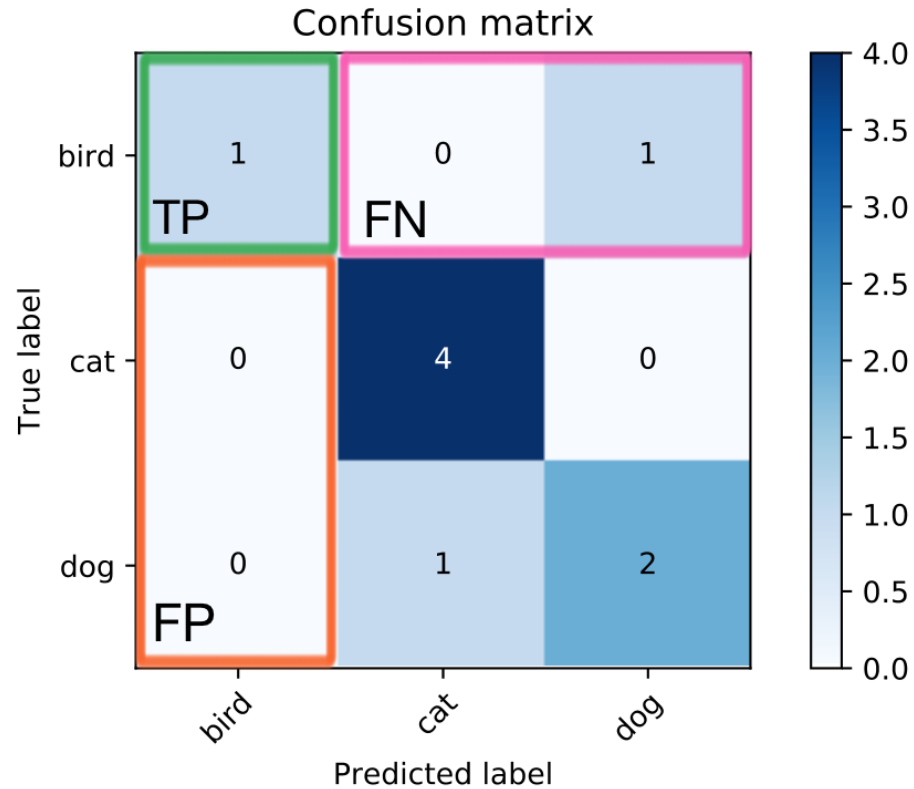
## Medida F

Permite comparar dos modelos de baja precisión y alta exhaustividad (recall) utiliza la media armónica para castigar los valores extremos

$$\frac{2 * recall * precision}{Recall + precision}$$



# MODELOS DE CLASIFICACIÓN









Fuentes: <https://www.iartificial.net/precision-recall-f1-accuracy-en-clasificacion>  
<https://www.youtube.com/watch?v=D5iuHDepUFw>



# MODELOS DE CLASIFICACIÓN – EJEMPLO MULTICLASE

- Se cuenta con una matriz de confusión para un problema de clases múltiples donde se debe predecir si una persona prefiere Facebook, Instagram o Snapchat.
- La matriz de confusión sería de 3 x 3.
- El verdadero positivo, verdadero negativo, falso positivo y falso negativo de cada clase se calcularía sumando los valores de celda de la siguiente manera:

		ACTUAL VALUES		
				
PREDICTED VALUES		<div>+ve</div> <div>1</div>	<div>-ve</div> <div>2</div>	<div>-ve</div> <div>3</div>
		<div>-ve</div> <div>4</div>	<div>+ve</div> <div>5</div>	<div>-ve</div> <div>6</div>
		<div>-ve</div> <div>7</div>	<div>-ve</div> <div>8</div>	<div>+ve</div> <div>9</div>

Facebook

$$TP = Cell_1$$

$$FP = Cell_2 + Cell_3$$

$$TN = Cell_5 + Cell_6 + Cell_8 + Cell_9$$

$$FN = Cell_4 + Cell_7$$

Instagram

$$TP = Cell_5$$

$$FP = Cell_4 + Cell_6$$

$$TN = Cell_1 + Cell_3 + Cell_7 + Cell_9$$

$$FN = Cell_2 + Cell_8$$

Snapchat

$$TP = Cell_9$$

$$FP = Cell_7 + Cell_8$$

$$TN = Cell_1 + Cell_2 + Cell_4 + Cell_5$$

$$FN = Cell_3 + Cell_6$$



# MODELOS DE CLASIFICACIÓN - CONCLUSIONES

## Resumen 1:

- No hay que guiarse solamente por la matriz de confusión.
- No solamente cuenta la exactitud y precisión.
  - Proporción de aciertos en la matriz de confusión
  - No hay un algoritmo que siempre sea “mejor” que otros
- No necesariamente un nivel bajo de aciertos en la predicción invalida el uso del modelo.
- Hay que tener en cuenta otros factores dependiendo del caso de estudio.
  - Interpretabilidad
  - Velocidad
    - Entrenamiento
    - Producción





# MODELOS DE CLASIFICACIÓN - CONCLUSIONES

## Resumen 2:

- Vimos las métricas más extendidas para evaluar el rendimiento de un modelo supervisado en tareas de clasificación.
- La Matriz de Confusión indica qué tipos de errores se cometen
- La métrica Exactitud es engañosa cuando las clases están desbalanceadas, nos hace creer que el modelo es mejor de lo que en realidad es.
- Las medidas de Precisión, Recall y F1 son más representativas y funcionan tanto si las clases están balanceadas como si no:
  - **Precisión** nos da la calidad de la predicción: ¿qué porcentaje de los que hemos dicho que son la clase positiva, en realidad lo son?
  - **Recall** nos da la cantidad: ¿qué porcentaje de la clase positiva hemos sido capaces de identificar? La sensibilidad (+) y la especificidad (-) indican la capacidad del estimador para discriminar los casos positivos, de los negativos.
  - **F1** combina Precisión y Recall en una sola medida.

Fuente: <https://www.iartificial.net/precision-recall-f1-accuracy-en-clasificacion/>

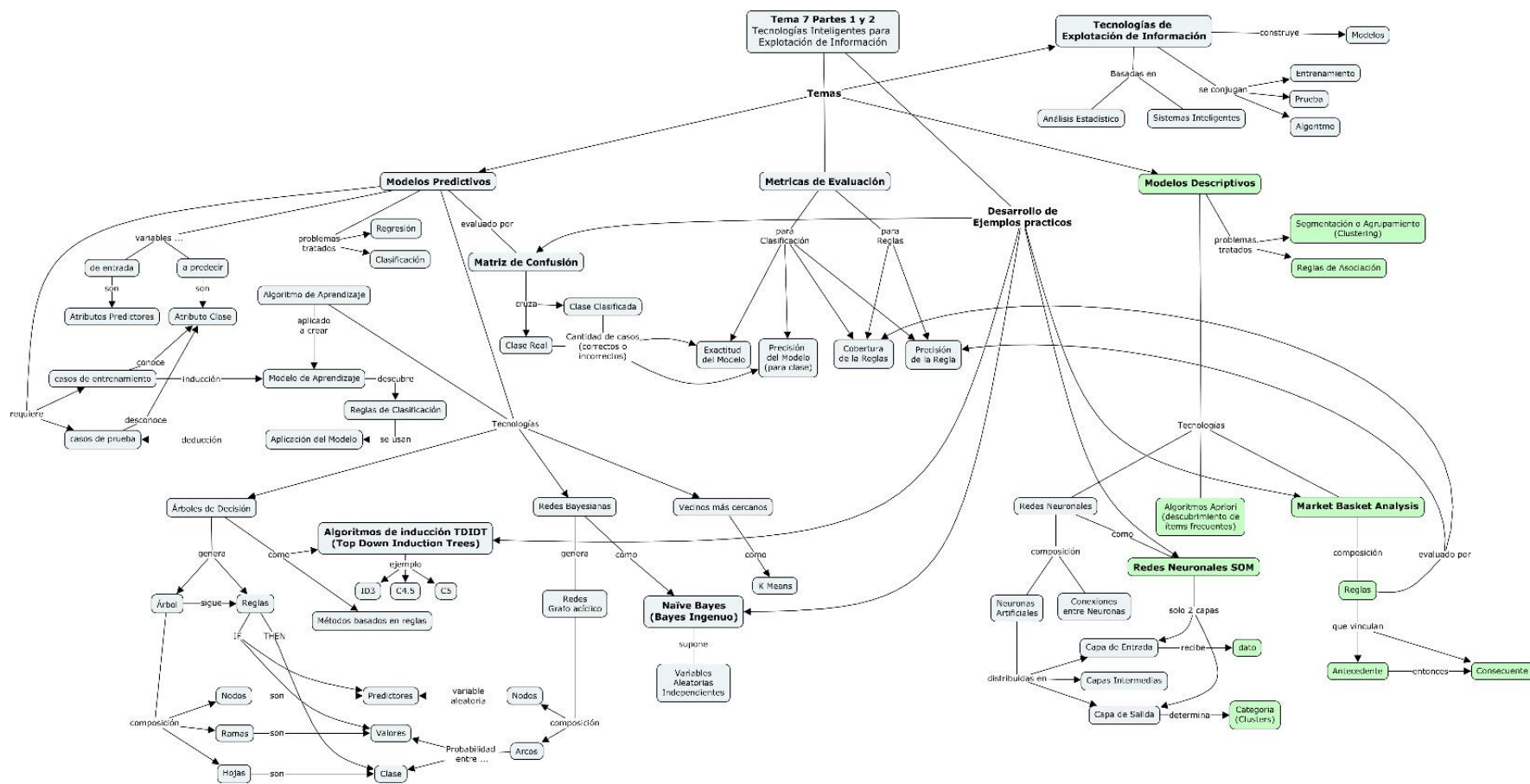
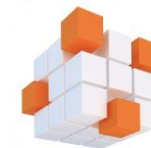


# MATERIAL COMPLEMENTARIO

## Fuentes:

- **Canal Electronics and Technology**
  - **Lista Machine Learning:** [Machine Learning - YouTube](#)
    - Modelo Naive Bayes: [Modelo Naive Bayes - Machine Learning | aprendizaje automático \(youtube.com\)](#)
    - Evaluación de Modelos: [Evaluación del Modelo - Machine Learning | aprendizaje automático \(youtube.com\)](#)

# RESUMEN CLASE



# TRABAJO PRÁCTICO DE MINERÍA DE DATOS

## CASO: CRÉDITOS BANCARIOS

ENTREGA **18/06/2025**



- Utilice la herramienta KNIMNE para desarrollar los procesos de explotación de información identificados en el Caso de Estudio; incluyendo tareas de Preprocesamiento, Modelos Predictivos, Descriptivos y Evaluación.
- Entregue un informe que contenga resultados, conclusiones obtenidas, gráficos, una tabla comparativa de métodos aplicados, y las recomendaciones que daría, de acuerdo con lo requerido en el enunciado del trabajo práctico.

