



# Redes Neuronales

Detalles no vistos en la clase pasada

FRANCISCO CISTERNAS

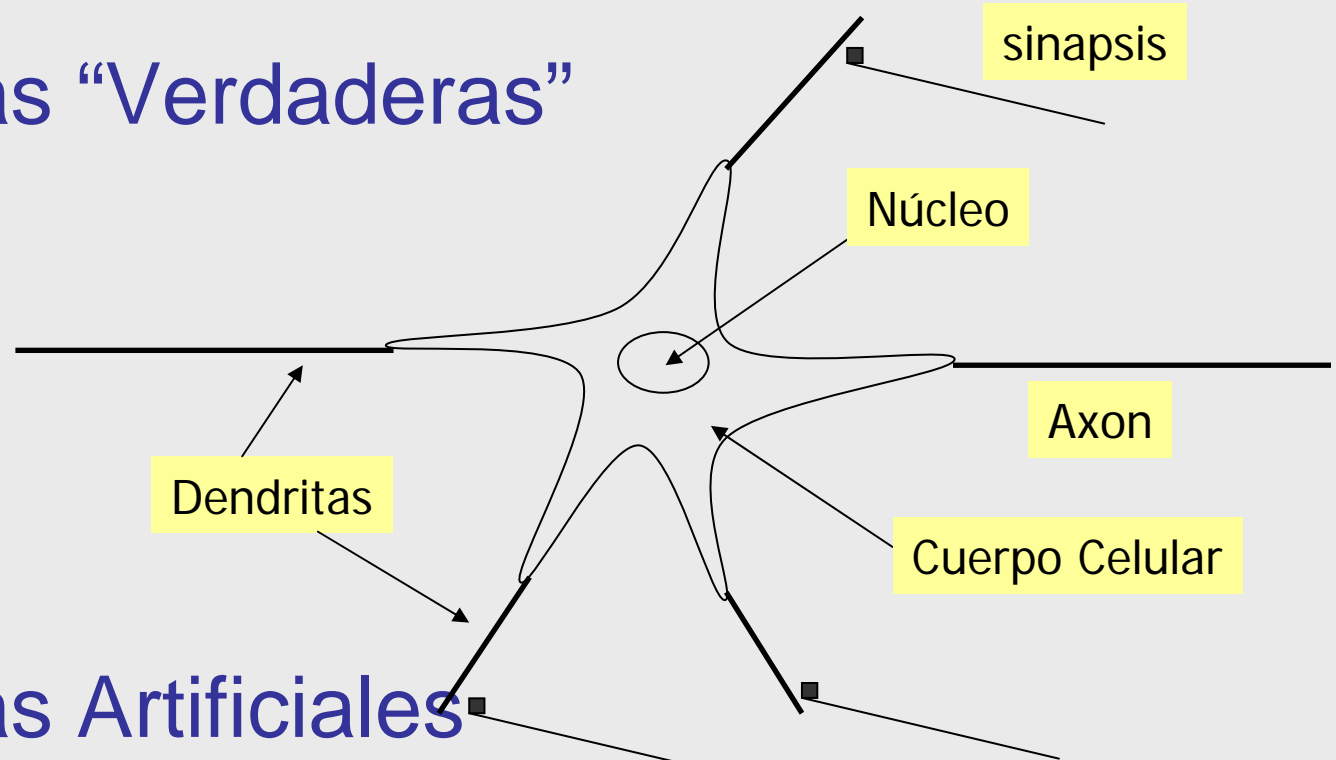
Departamento de Ingeniería  
Industrial  
Universidad de Chile

# Redes neuronales: Historia

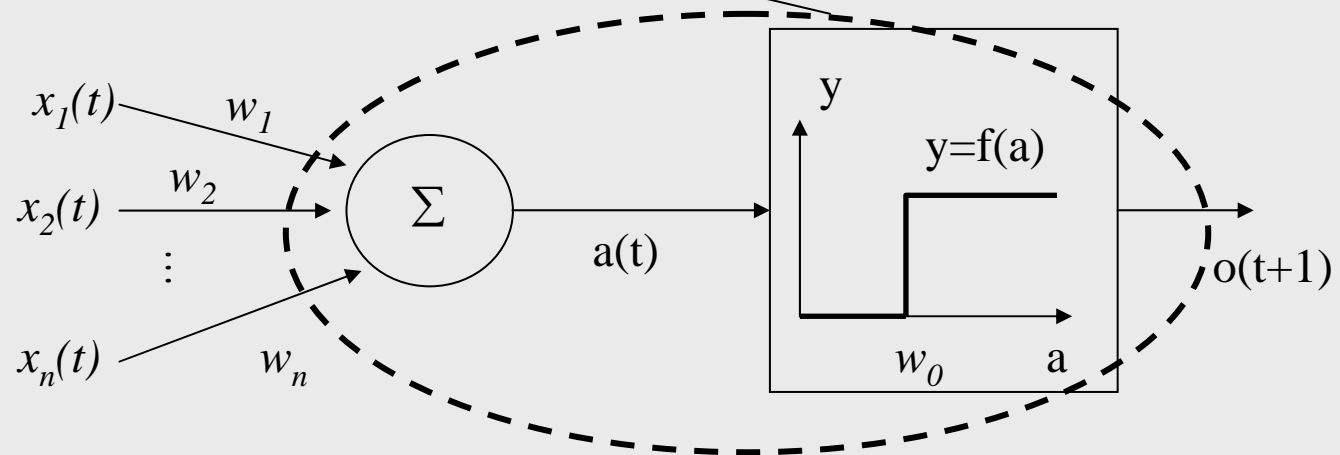
1. 1943 McCulloch/Pitts: Descripción formal de una neurona
2. 1949 Donald O. Hebb: “The Organization of Behavior”; primera regla de aprendizaje
3. 1958 Rosenblatt: Presentación de un Perceptron
4. 1960 Widrow/Hoff: Adaline (adaptive linear element), madaline (multiple adalines)
5. 1969 Minsky/Papert: “Perceptrons”; crítica de perceptrons
6. 1982 Hopfield: Redes auto-asociativas para reconocimiento de patrones
7. 1982 Kohonen: Self-organizing feature maps
8. 1986 Rumelhart, Hinton, Williams: Backpropagation algorithm
9. [1974 Paul Werbos: Backpropagation Algorithm!! ]

# Neuronas

## 1. Neuronas “Verdaderas”

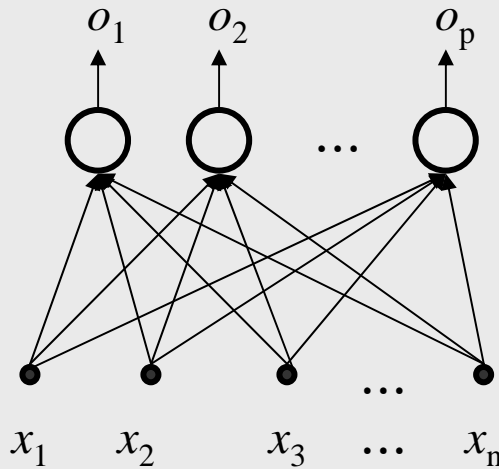


## 2. Neuronas Artificiales



# Perceptron (1962)

## 1. Generalización y formalización de las redes neuronales.

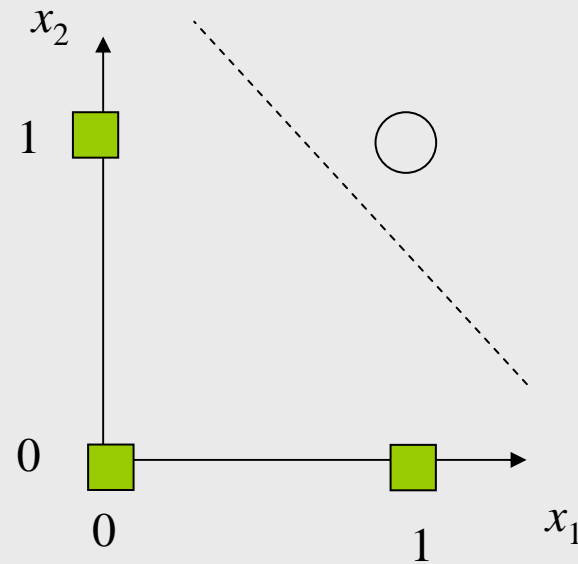


$$o_i = f(a_i) = f\left(\sum_{k=0}^n w_{ik} x_k\right) \quad i = 1, \dots, p$$

# Aprendizaje con Perceptron

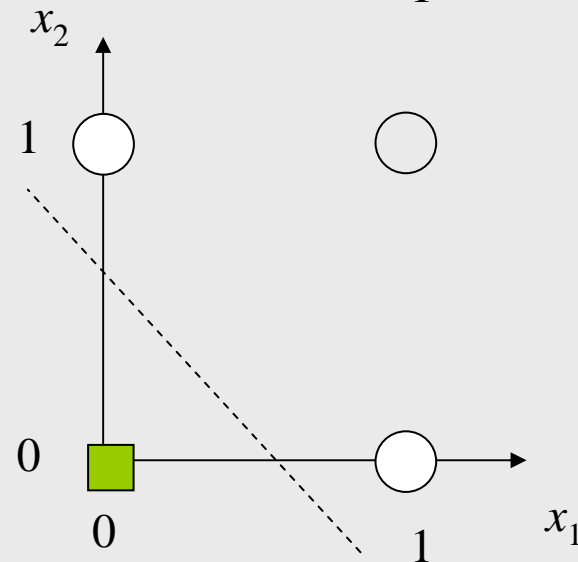
Función “y”

$x_1$	$x_2$	$y$
0	0	0
0	1	0
1	0	0
1	1	1



Función “o”

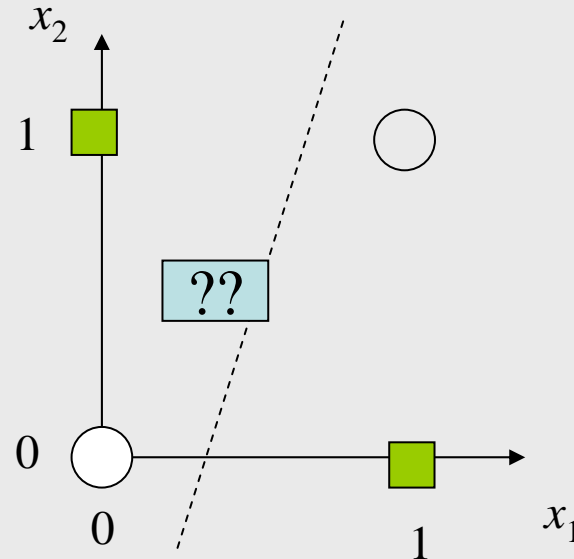
$x_1$	$x_2$	$y$
0	0	0
0	1	1
1	0	1
1	1	1



# Perceptron la falla

## 1. La función XOR (exclusive or):

$x_1$	$x_2$	$y$
0	0	0
0	1	1
1	0	1
1	1	0



Minsky, Papert (1969)

# Multilayer Perceptron (MLP)

1. El 90% de las aplicaciones de redes neuronales están referidas a MLP

$$o_i = f\left(\sum_{j=0}^n W_j f\left(\sum_{k=0}^n w_{ik} x_k\right)\right)$$

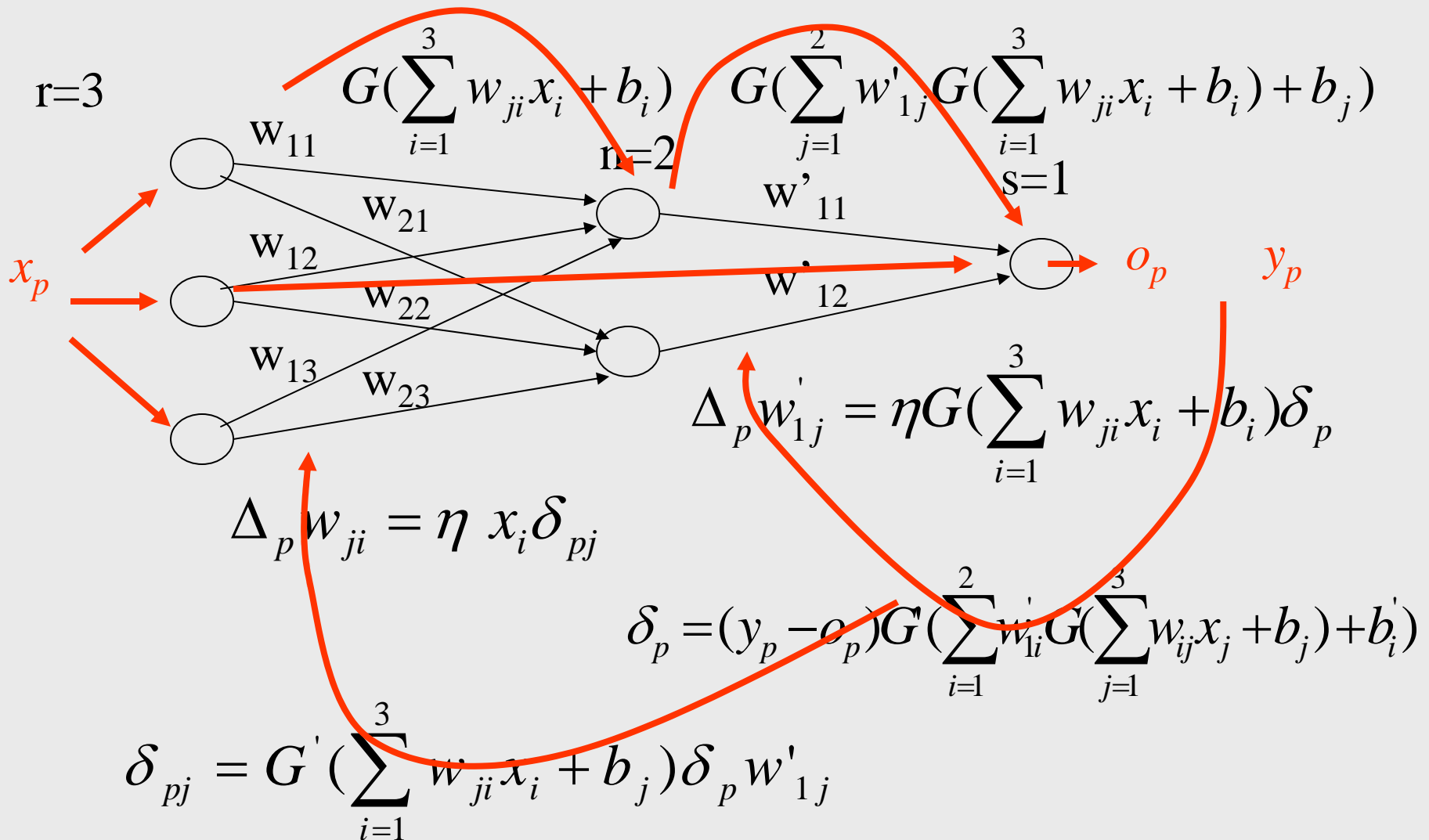
*Es una función no lineal, de una combinación lineal de funciones no lineales de funciones de combinaciones lineales de los datos de entrada; => **Clasificación y Regresión no lineal!!***

2. ¿Cómo resuelvo esto?, →

Backpropagation, Un ejemplo:

$$f(x) = G\left(\sum_{j=1}^2 w'_{1j} G\left(\sum_{i=1}^3 w_{ji} x_i + b_j\right) + b'_1\right)$$

# Backpropagation un ejemplo





# Árboles de decisión

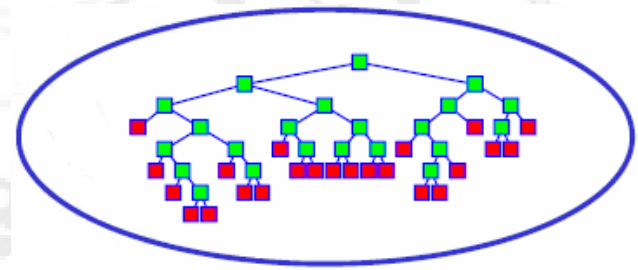
Definiciones y aplicaciones básicas

**FRANCISCO CISTERNAS**

**JAIME MIRANDA**

Departamento de Ingeniería  
Industrial

Universidad de Chile

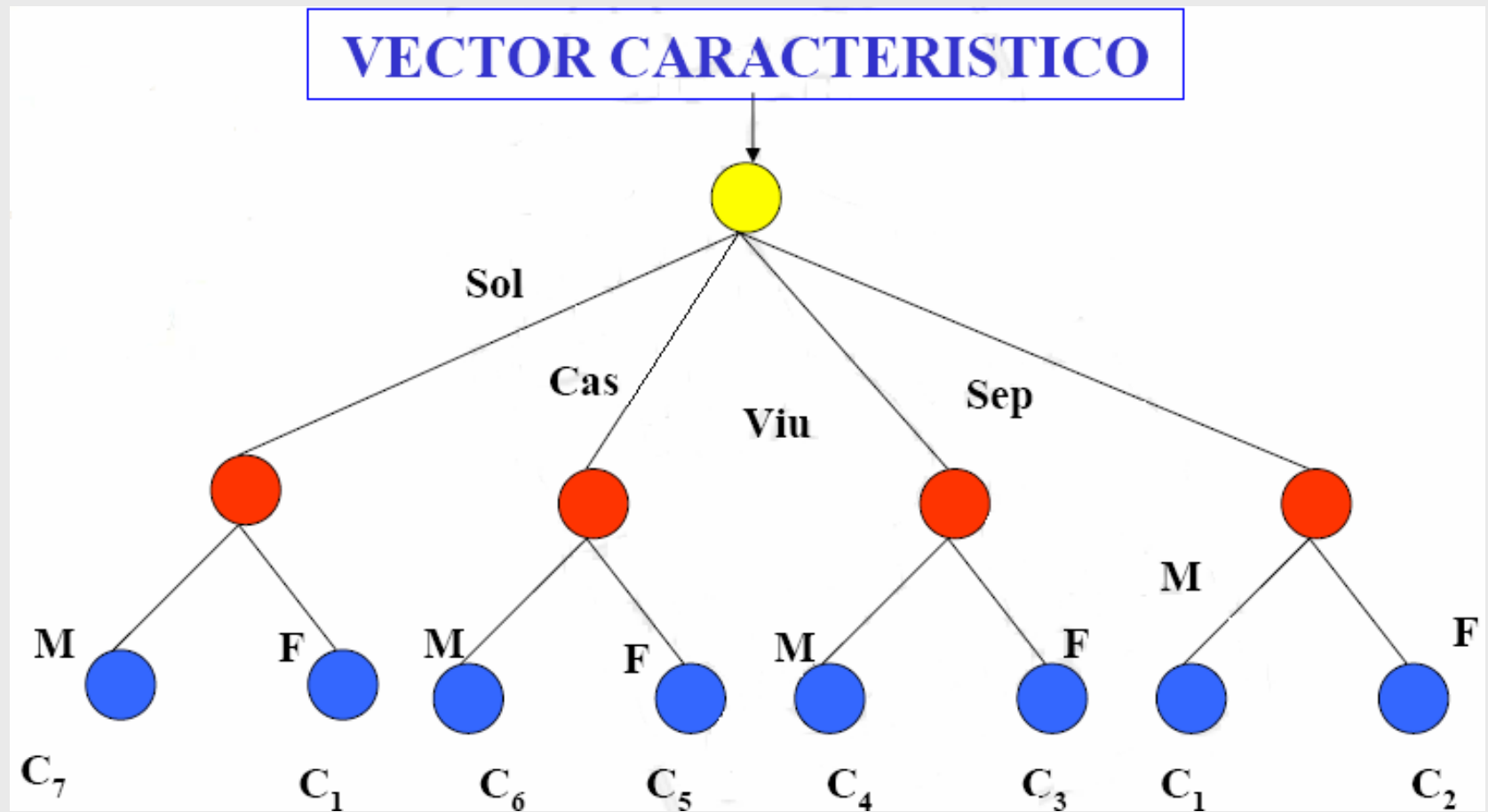


# Un pequeño ejemplo

## Determinación de la renta usando variables sociodemográficas

	Ck	X1	X2	X3	X4
k	Nivel de Renta	Estado Civil	Nivel Educacional	Rango Edad	Sexo
1	0-200	Soltero	Ed. Media	20-29	Hombres
2	201-400			30-39	
3	401-600	Casado	EUN	40-49	
4	601-800			50-59	
5	801-1000	Separado	Tecnico	60-69	Mujeres
6	1001-2000			70-79	
7	2001-15000	Viudo	Universitario	80-89	

# Un pequeño ejemplo..



# Nociones sobre árboles

1. Son un conjunto conexo, acíclico y dirigido.
2. Permite tener :
  - ☐ Valores mal clasificados.
  - ☐ Valores perdidos.
  - ☐ Una ilustración sobre la manera en que se pueden desglosar los problemas y la secuencia del proceso de decisión (subproblemas).
3. Los árboles de decisión separan datos en conjuntos de reglas que probablemente respondan a un efecto o variable objetivo

# Son útiles

## 1. Cuando

- ☐ tengamos instancias pares atributo/Valor.
- ☐ Nuestra función de salida tenga valores de salida discretos (0, 1).

## 2. Aplicables problemas de:

- ☐ Clasificación.
- ☐ Regresión.

## 1. CLASIFICACIÓN

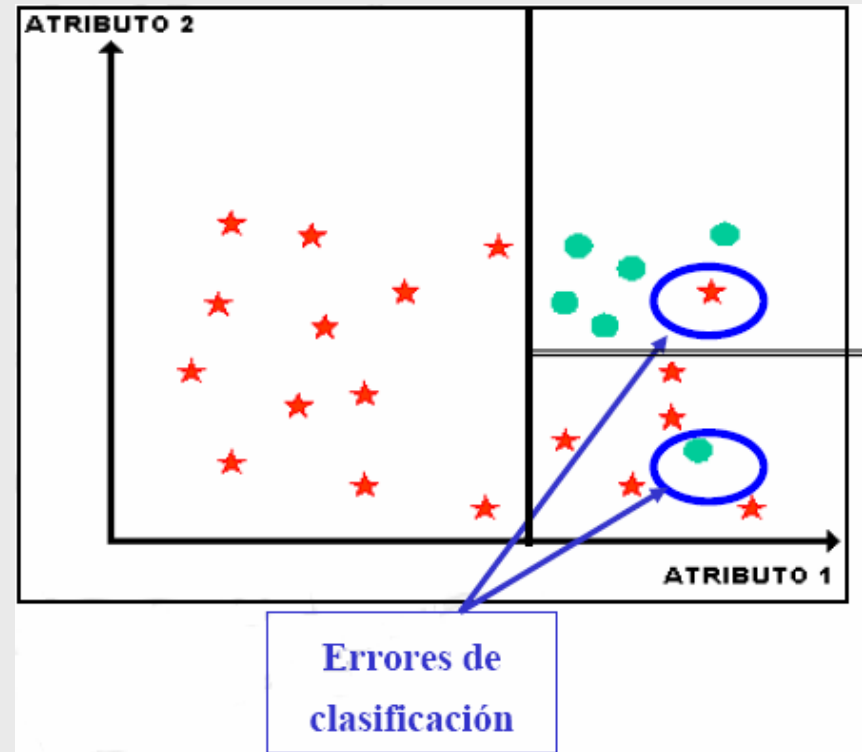
- ☐ Se trata de encontrar el grado de pertenencia de un objeto a una clase específica.

## 2. REGRESIÓN

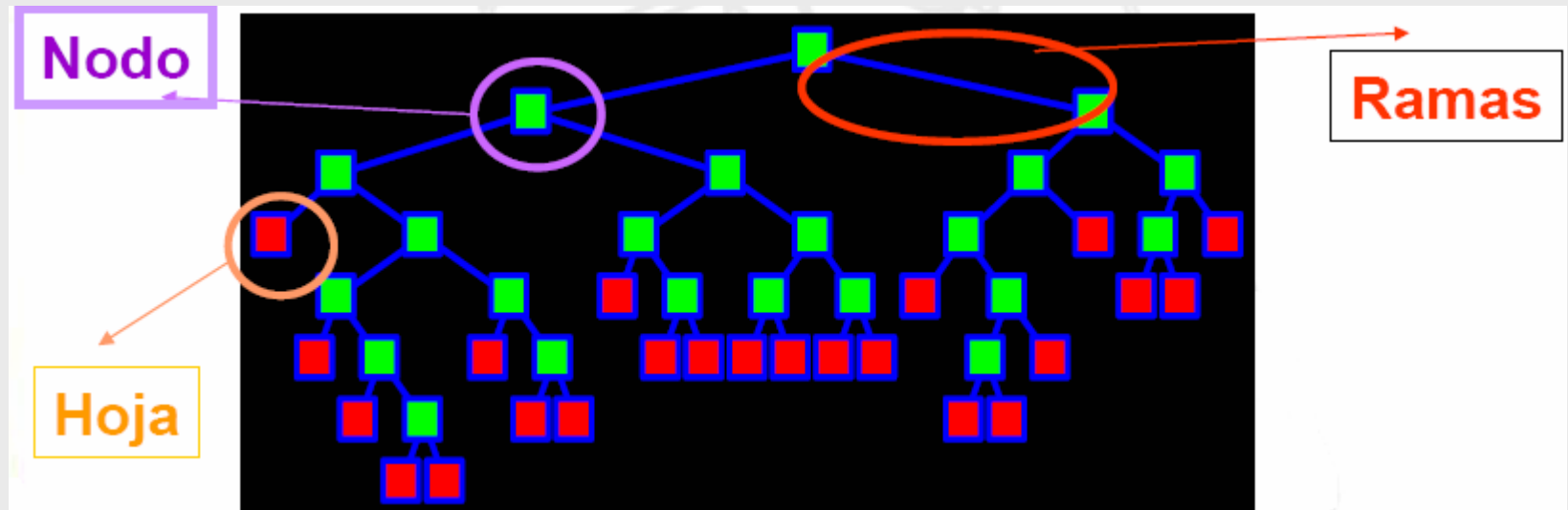
- ☐ Se trata de predecir un valor futuro de una variable en base a su comportamiento pasado.

# Objetivo general

*“Obtener modelos que discrimine las instancias de entrada en diferentes clases de equivalencia por medio de los valores de diferentes atributos.”*



# Arquitectura





## 1. Un nodo

- ☐ es un punto de unión, donde se representa un lugar en el que se debe tomar una decisión.

## 2. Una rama

- ☐ representa un arco de conexión entre nodos.

## 3. Una hoja

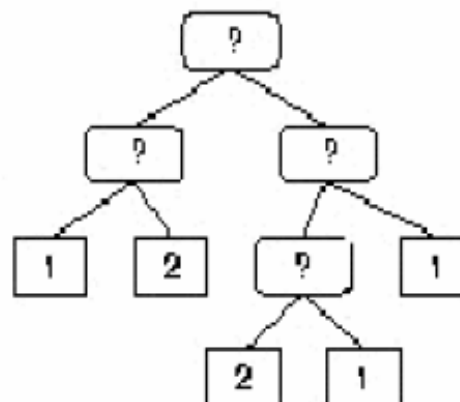
- ☐ es un nodo terminal (sin hijos)

# Etapas

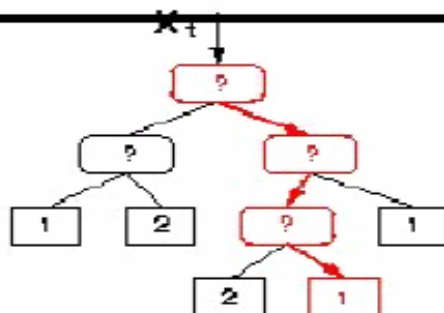
## Aprendizaje

$S$

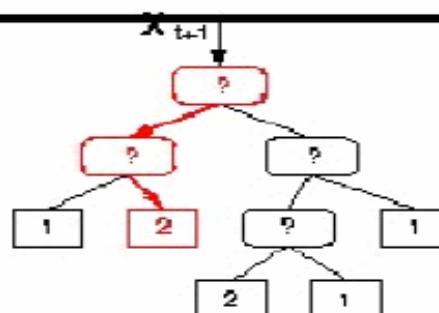
Aprendizaje



## Clasificación



$$d(X_t) = 1$$



$$d(X_{t+1}) = 2$$

# Respecto a las divisiones

1. Cada partición depende de un único atributo.
2. •Existen varios criterios para poder hacer la división en el nodo.
3. •La gran mayoría se basa en la diversidad del nodo.

## 1. Gini Index

$$Gini(V) = \sum_{i=j} p(v) * \sum_{i \neq j} p(i|v) * p(j|v)$$

Probabilidad de estar en el nodo v

Probabilidad de pertenecer a la clase i/j dado que estoy en el nodo v

- 1. Se elige el atributo que posee el mayor índice de GINI.**
- 2. A medida que se baja en el árbol el atributo posee menor índice de GINI.**
- 3. Este índice ve que tan heterogéneo es el nodo respecto a los elementos que lo conforman.**

- 1. Un árbol de decisión da una buena descripción visual en problemas relativamente simples, pero su complejidad aumenta exponencialmente a medida que se agregan etapas adicionales.**
- 2. Atributos con numerosos valores**
  - ☐ debido a que inducen particiones más finas, que no sean significativos

## 1. Ruido

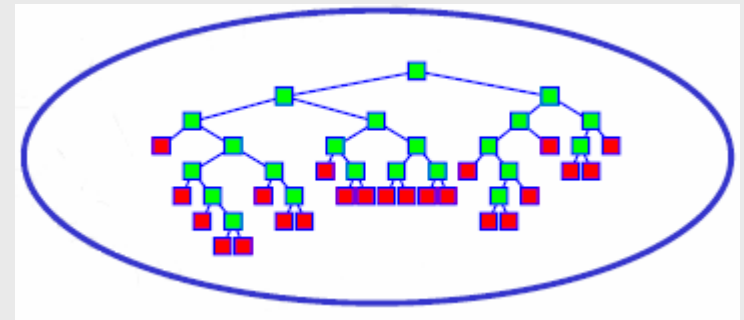
- ☐ Ejemplos con la misma descripción pero distinta clase
- ☐ Consecuencia: error no nulo en ejemplos de entrenamiento.

## 2. Posibilidades discretas

- ☐ Solo es posible tener un numero finito de “ramas” y no un continuo.

## 1. Sobreajuste

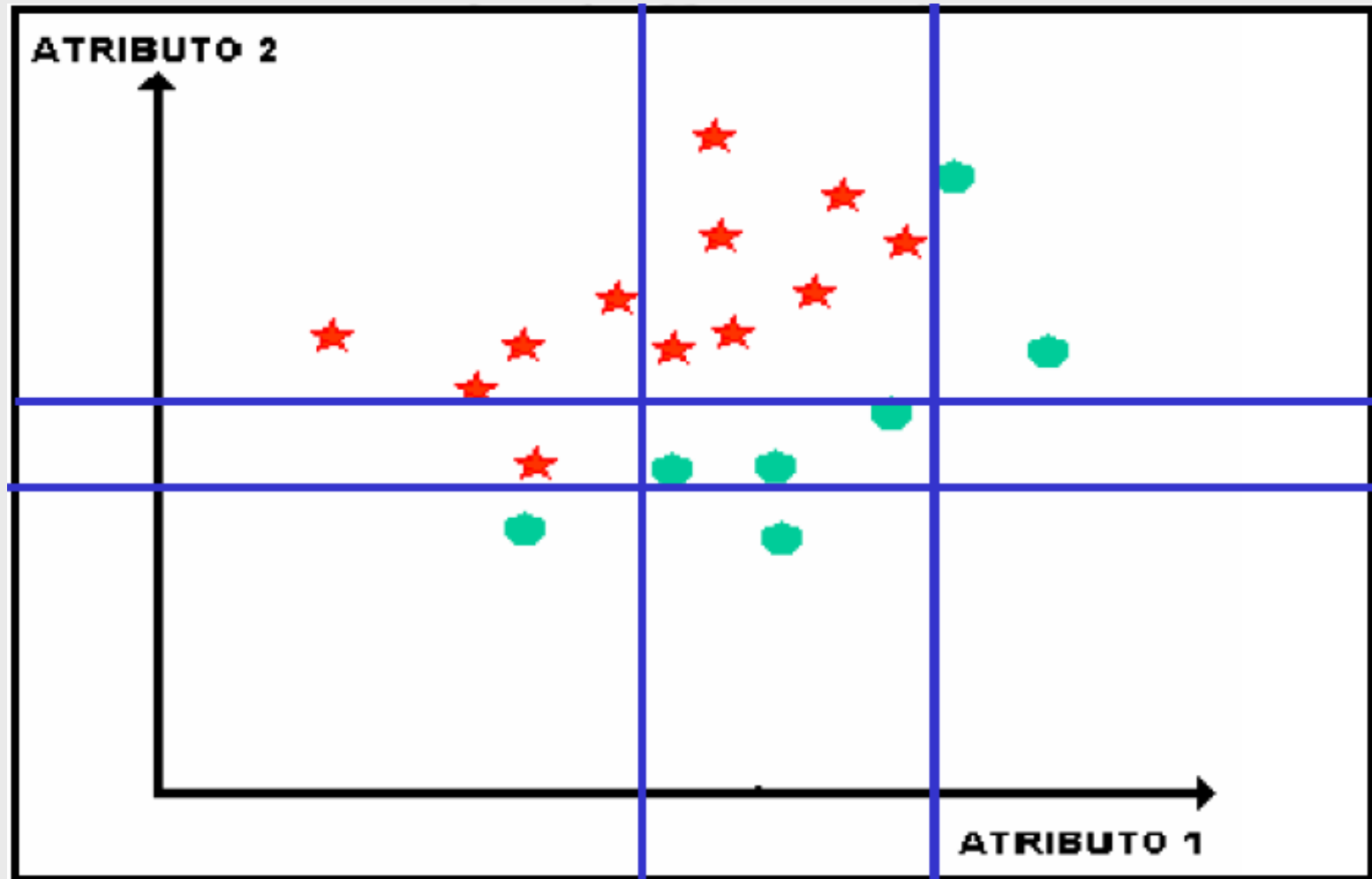
- ❑ •uso de atributos no relevantes para ajustar árbol a datos
- ❑ •consecuencia: disminuye capacidad generalización del modelo.



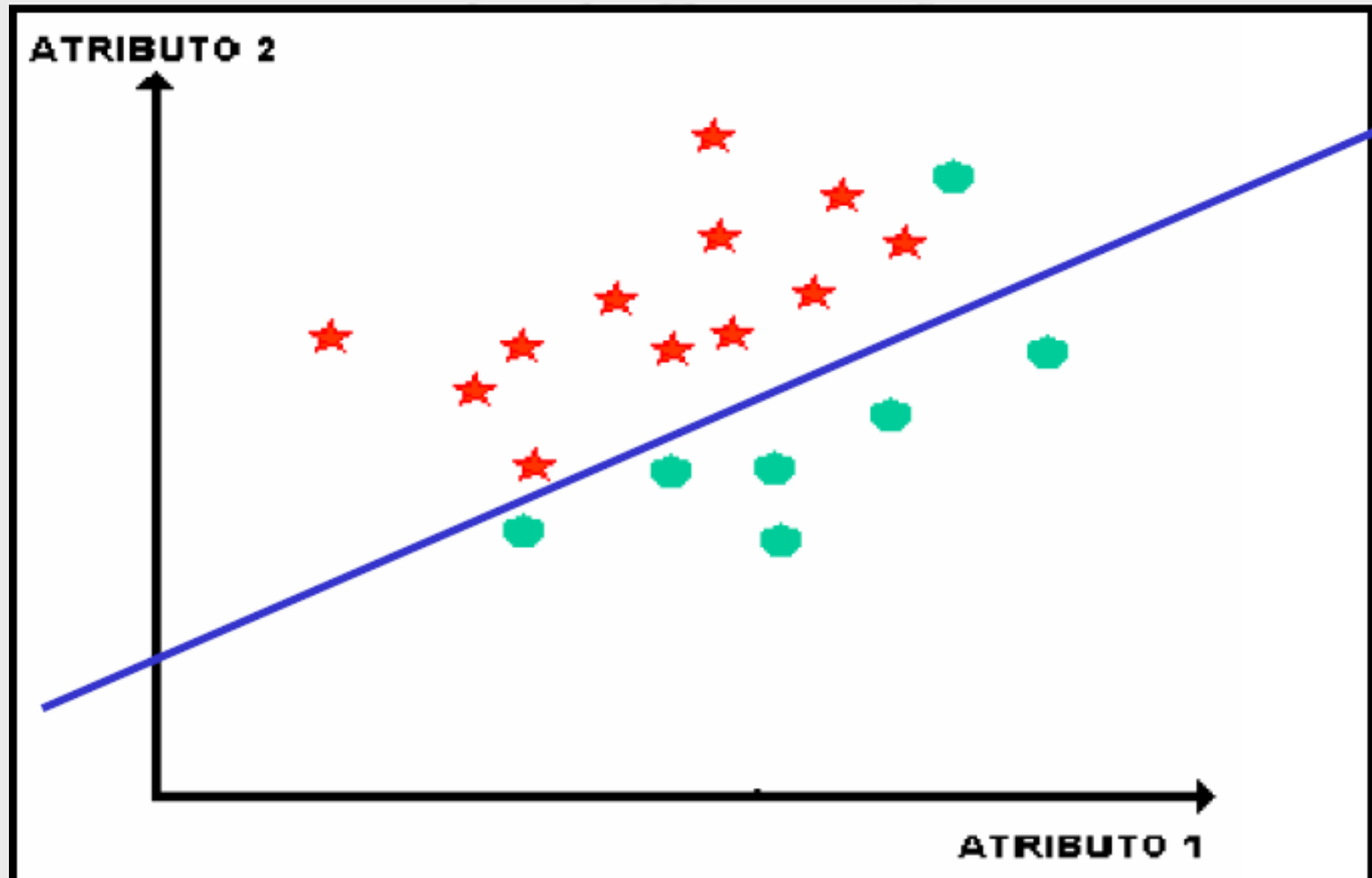


1. Cuando un árbol se expande demasiado el nivel de complejidad, tanto para el cálculo, como la aplicación, puede ser muy engorroso.
2. Dado que los árboles sólo utilizan restricciones paralelas a los ejes, es difícil segmentar clases que, teniendo separaciones lineales, estas separaciones no son paralelas a los ejes.

# Limitaciones y problemas



# Solución



## 1. Objetivo:

- ☐ **Construcción de árboles de decisión dividiendo el conjunto de instancias según el criterio de diferentes atributos.**

## 2. Ventajas:

- ☐ **El enfoque sobre el que mayor número de investigaciones se han desarrollado.**
- ☐ **Versiones de C4.5 para soportar: valores continuos, valores perdidos, datos con ruido, ...**

## 1. Método pesimista (Quinlan87):

- ❑ •Se basa en reemplazar subarbol por hoja si una estimación pesimista del error en la hoja es mejor que la estimación pesimista del error del subárbol.

# Árboles de decisión

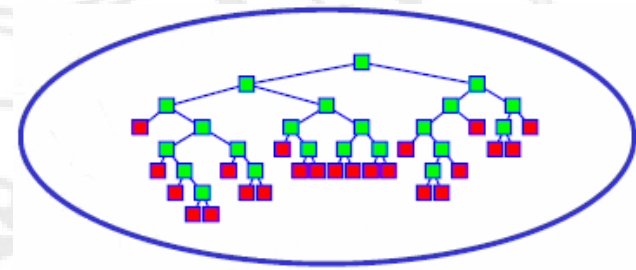
## Definiciones y aplicaciones básicas

**FRANCISCO CISTERNAS**

**JAIME MIRANDA**

Departamento de Ingeniería  
Industrial

Universidad de Chile



**Control de lectura**