Arboles de Decisión (I)

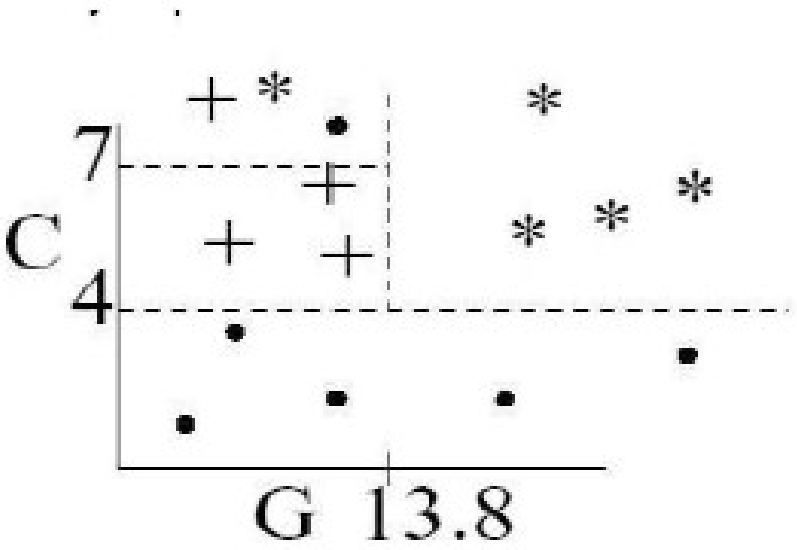
Carlos Hurtado L.

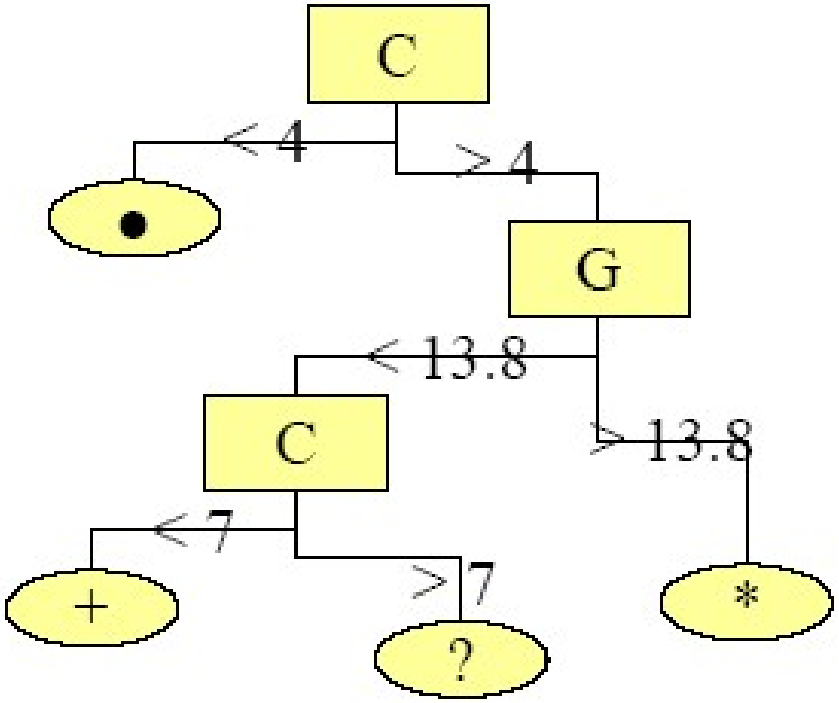
Depto de Ciencias de la Computación, Universidad de Chile

Clasificación: Tipos de Modelos

- Enfoque Discriminante
  - Arboles de Decisión
  - Reglas de Decisión
  - Discriminantes lineales
- Enfoque Generativo
  - Redes Bayesianas
  - Modelos paramétricos
- Enfoque de Regresión
  - Redes Neuronales
  - Regresión Logística

Arboles de Decisión





Arboles de Decisión

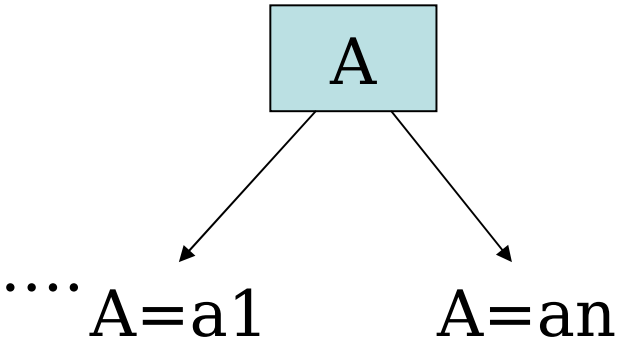
- Fáciles de construir
- Fáciles de interpretar
- Buena precisión en muchas aplicaciones

Construcción de árboles de decisión: algoritmos

- Algritmos de Memoria Principal
  - Manejan miles de datos
  - Algoritmo de Hunt (CLS, 1960´s)
  - ID3 (Quinlin 70’s and 80’s), C4.5 (Quinlin 90’s)
- Algritms Escalables
  - Manejas millones de datos
  - SLIQ, SPRINT

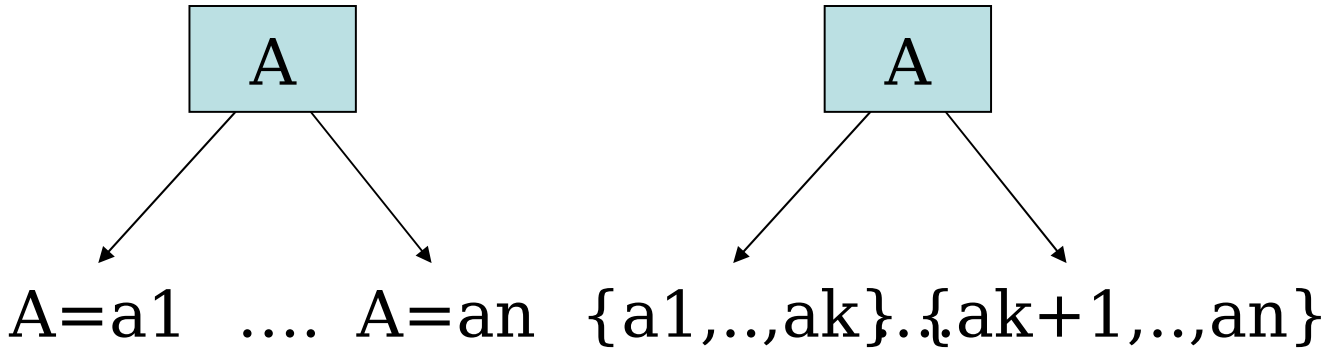
Split

- Definición: un *split* es una variable (atributo) más una lista de condiciones sobre la variable.



Split para variables categóricas

- Split Simple vs. Split Complejo



Split para variables numéricas

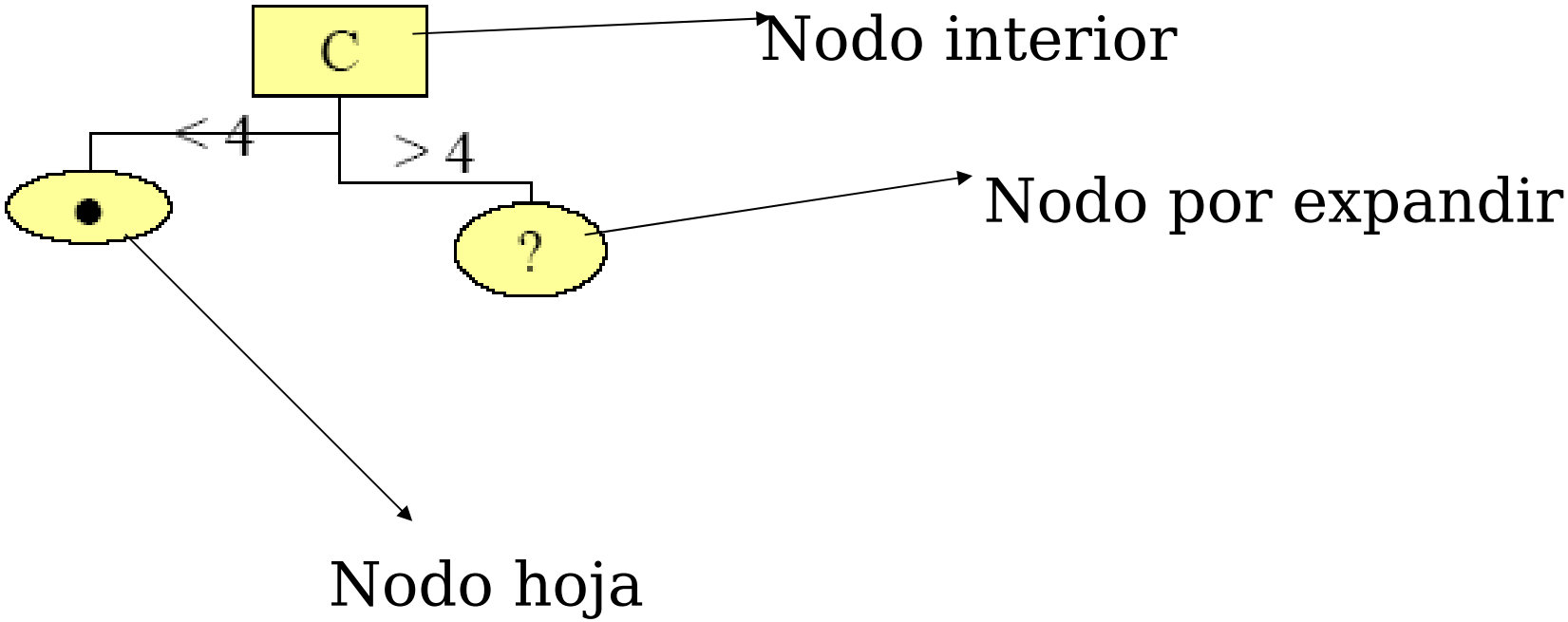
- Decisión binaria: A<= v, A > v
- Rangos: [0,15k), [15k,60k), [60k,100k), etc.
- Combinación lineal de variables

$$3A + 4B > 5$$

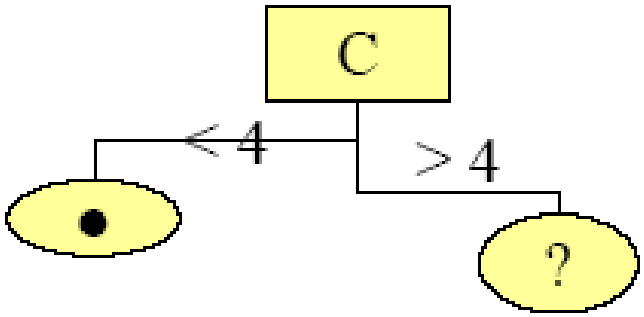
Algoritmo de Hunt

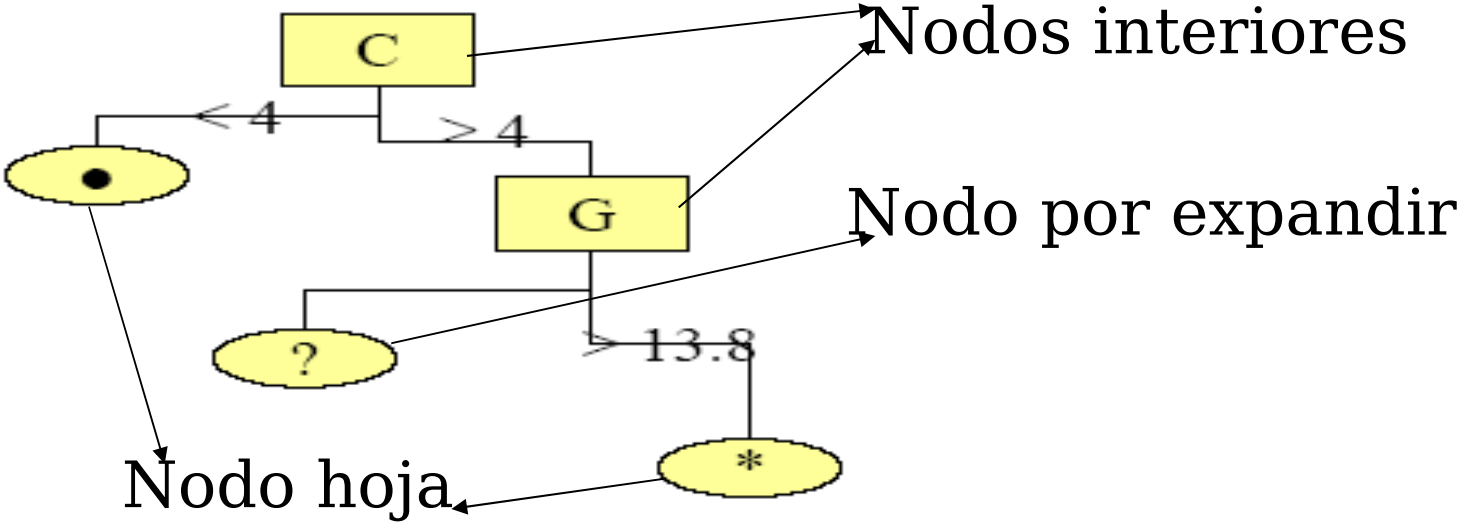
- Idea básica: cada nodo en el árbol de decisión tiene asociado un subconjunto de los datos de entrenamiento
- Inicialmente, el nodo raíz tiene asociado todo el conjunto de entrenamiento
- Construimos un árbol parcial que tiene tres tipos de nodos:
  - Expandidos (interiores)
  - Hojas: serán hojas en el árbol final y tienen asociada una clase
  - Nodos por expandir: son hojas en el árbol parcial, pero deben ser expandidos
- Operación de expansión de un nodo t:
  - Encontrar el mejor split para t
  - Particionar los datos de t en nodos hijos de acuerdo al split
  - Etiquetar t y sus nodos hijos con el mejor split

Algoritmo de Hunt (I)

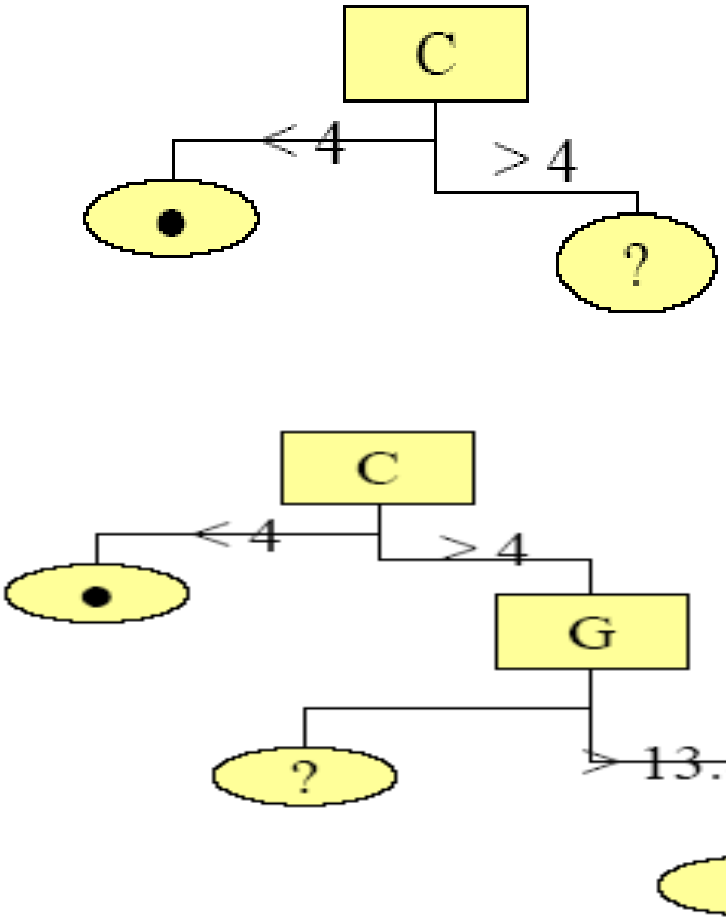


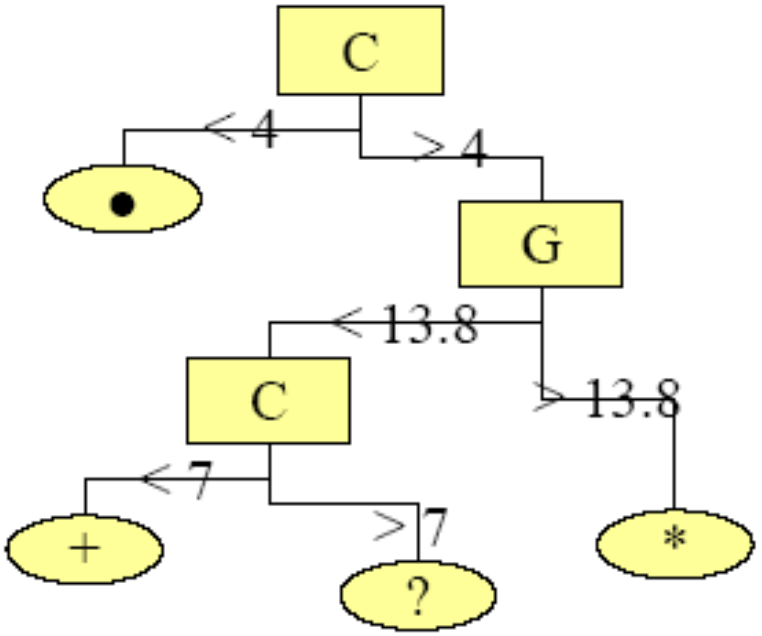
Algoritmo de Hunt (II)





Algoritmo de Hunt (III)





Algoritmo de Hunt

- Main(Conjunto de Datos T)
  - Expandir(T)
- Expandir(Conjunto de Datos S)
  - If (todos los datos están en la misma clase) then return
  - Encontrar el mejor split r
  - Usar r para particonar S en S1 y S2
  - Expandir(S1)
  - Expandir(S2)

Algoritmo de Hunt: observaciones

- Las operaciones de expansión se realizan “primero en profundidad”
- Lo complejo es encontrar el mejor split en cada operación de expansión
- Número de splits a buscar depende del tipo de split que consideramos.

¿Cuál es el mejor split?

- Buscamos splits que generen nodos hijos con la menor impureza posible (mayor pureza posible)
- Existen distintos métodos para evaluar splits. **Criterios de Selección**:
  - Indice Gini
  - Entropía (Ganancia de información)
  - Test Chi-cuadrado
  - Proporción de Ganancia de Información

Selección de splits usando índice Gini

- Recordemos que cada nodo del árbol define un subconjunto de los datos de entrenamientos
- Dado un nodo t del árbol, $Gini(t)$ mide el grado de impureza de t con respecto a las clases
  - Mayor $Gini(t)$ implica mayor impureza
  - $Gini(t)$ = 1 – Prob. De sacar dos registros de la misma clase

Indice Gini

- Recordar que el nodo t tiene asociado un subconjunto de los datos

- $Gini(t)$: probabilidad de NO sacar dos registros de la misma clase del nodo t

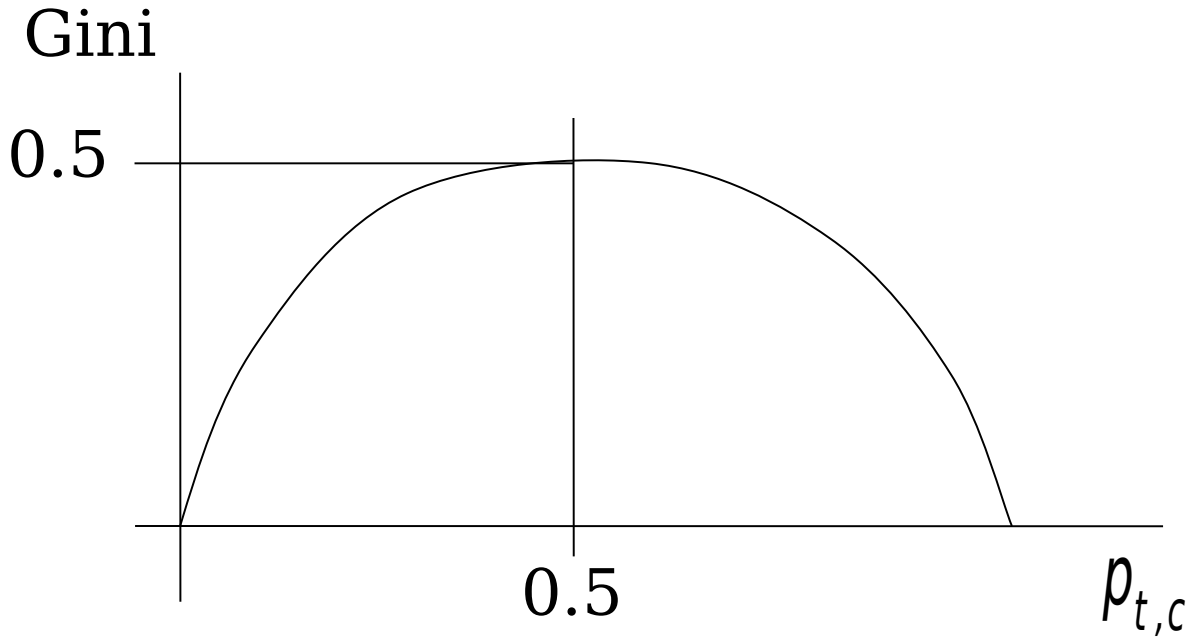
$$Gini(t) = 1 - \sum_{c \in C} p_{t,c}^2$$

donde C es el conjunto de clases y $p_{t,c}$ es la prob. de ocurrencia de la clase c en el nodo t

Indice Gini: ejemplo

C_1	C_2	Gini
0	6	0
1	5	0.278
2	4	0.444
3	3	0.5

Indice Gini



Selección de Splits: GiniSplit

- Criterio para elegir un split: selecciónar el split con menor gini ponderado (GiniSplit)
- Dado un split S={s1,...,sn} de t

$$GiniSplit(t,s)=\sum_{s\in S}\frac{|s|}{|t|}Gini(s)$$

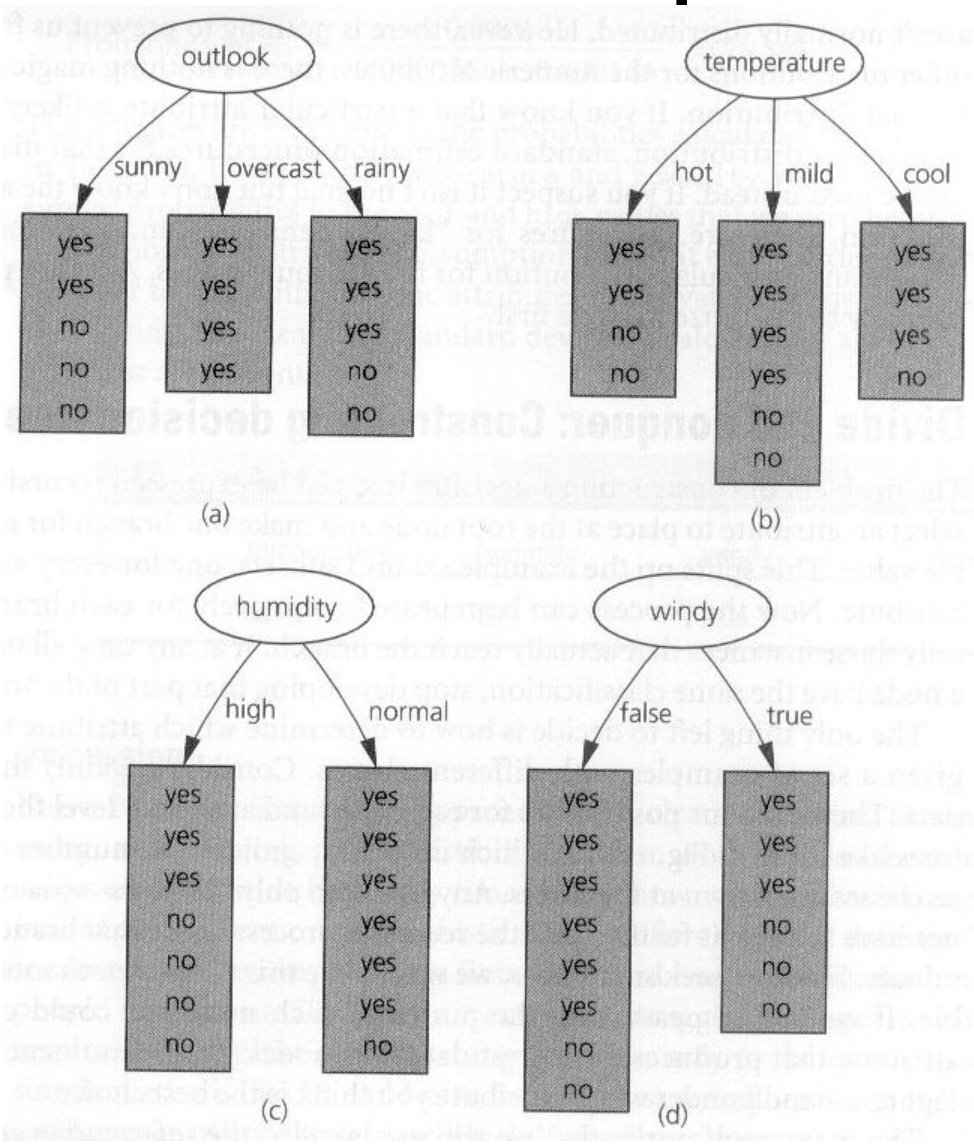
Ejemplo: weather.nominal

Outlook	Temp.	Humidity	Windy	Play
Sunny	Hot	High	FALSE	No
Sunny	Hot	High	TRUE	No
Overcast	Hot	High	FALSE	Yes
Rainy	Mild	High	FALSE	Yes
Rainy	Cool	Normal	FALSE	Yes
Rainy	Cool	Normal	TRUE	No
Overcast	Cool	Normal	TRUE	Yes
Sunny	Mild	High	FALSE	No
Sunny	Cool	Normal	FALSE	Yes
Rainy	Mild	Normal	FALSE	Yes
Sunny	Mild	Normal	TRUE	Yes
Overcast	Mild	High	TRUE	Yes
Overcast	Hot	Normal	FALSE	Yes
Rainy	Mild	High	TRUE	No

Weather.nominal: splits

- En este caso todos los atributos son nominales
- En este ejemplo usaremos *splits* simples

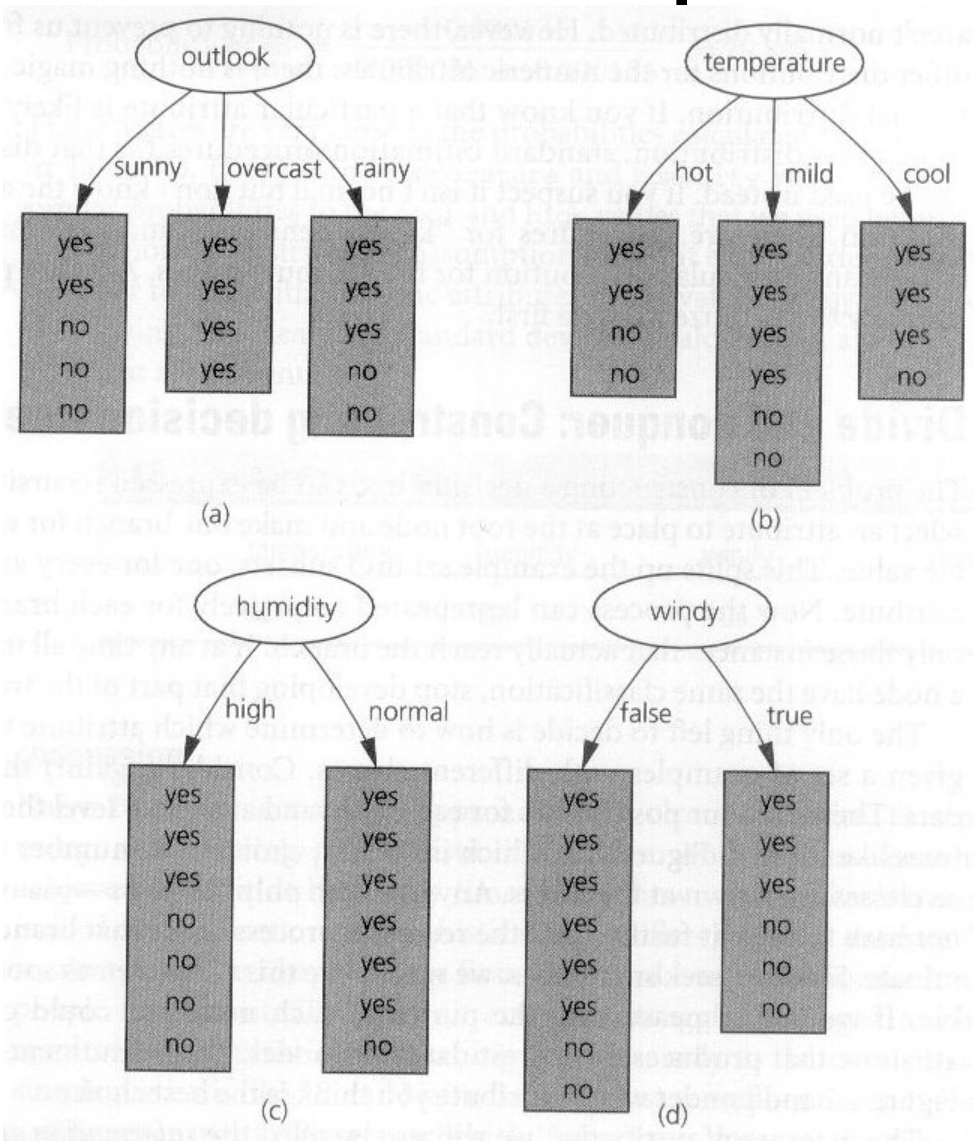
Posibles splits



weather.nominal: atributos vs. clases

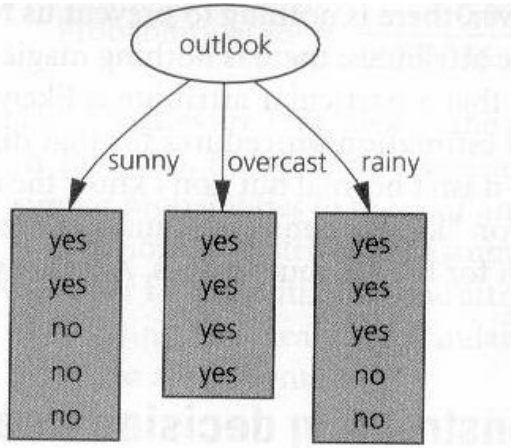
Outlook			*Temperature*			*Humidity*			*Windy*			*Play*	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	Hot	2	2	High	3	4	FALSE	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	TRUE	3	3		
Rainy	3	2	Cool	3	1								

Posibles splits



Ejemplo

- Split simple sobre variable Outlook
  - Gini(sunny) = 1- 0.16 - 0.36 = 0.48
  - Gini(overcast)= 0
  - Gini(rainy)= 0.48
  - GiniSplit = (5/14) 0.48 + 0 0.48 + (5/14) 0.48 = 0.35



Ejercicio: selección de splits usando Gini para weather.nominal

1. Dar el número de splits que necesitamos evaluar en la primera iteración del algoritmo de Hunt, para (a) splits complejos y (b) splits simples.
2. Seleccionar el mejor split usando el criterio del índice Gini
   - Calcular el GiniSplit de cada split
   - Determinar el mejor split