Tema 10: Árboles de Clasificación

Pedro Larrañaga, Iñaki Inza, Abdelmalik Moujahid

Departamento de Ciencias de la Computación e Inteligencia Artificial

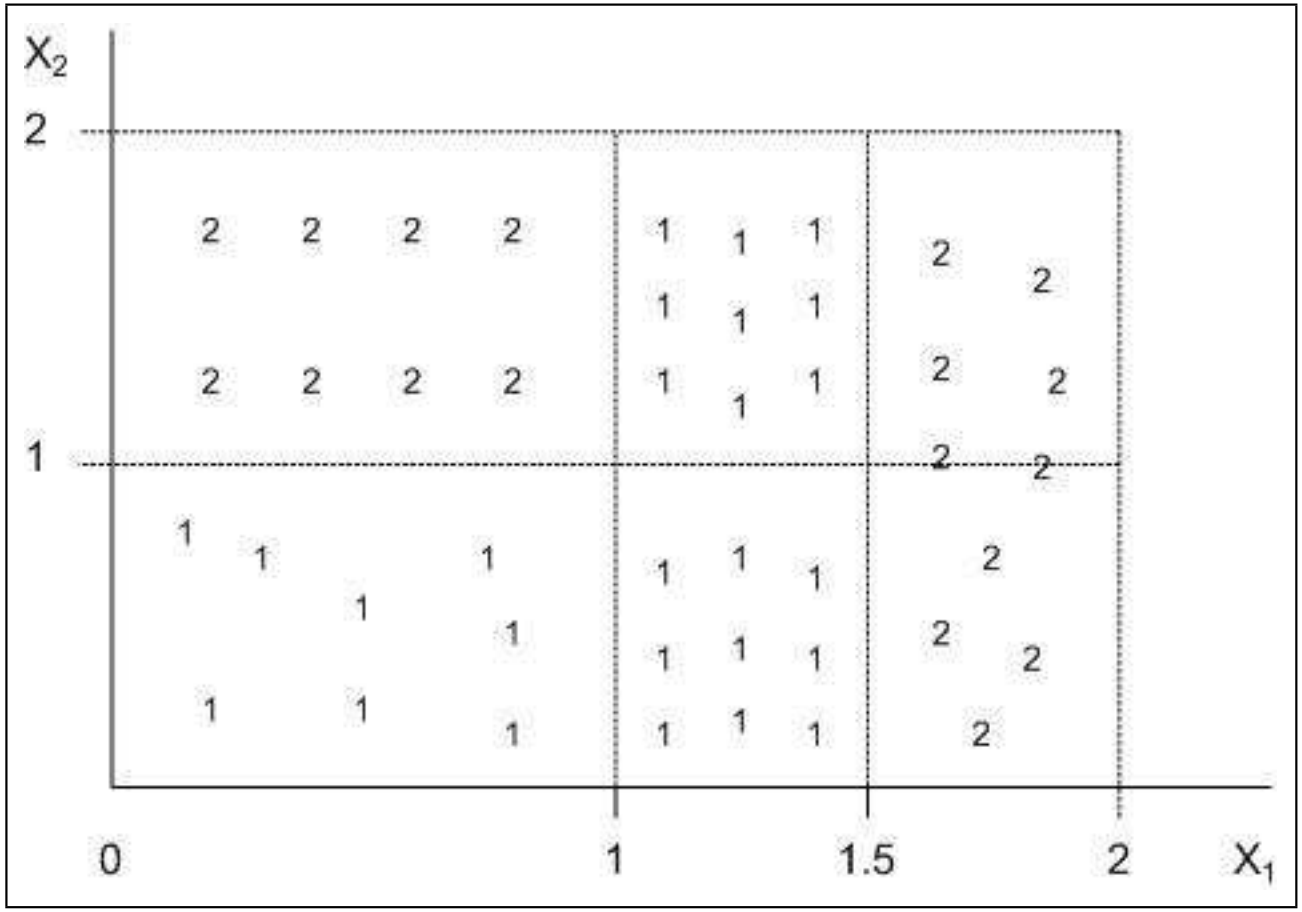
Universidad del País Vasco

http://www.sc.ehu.es/isg/

Introducción

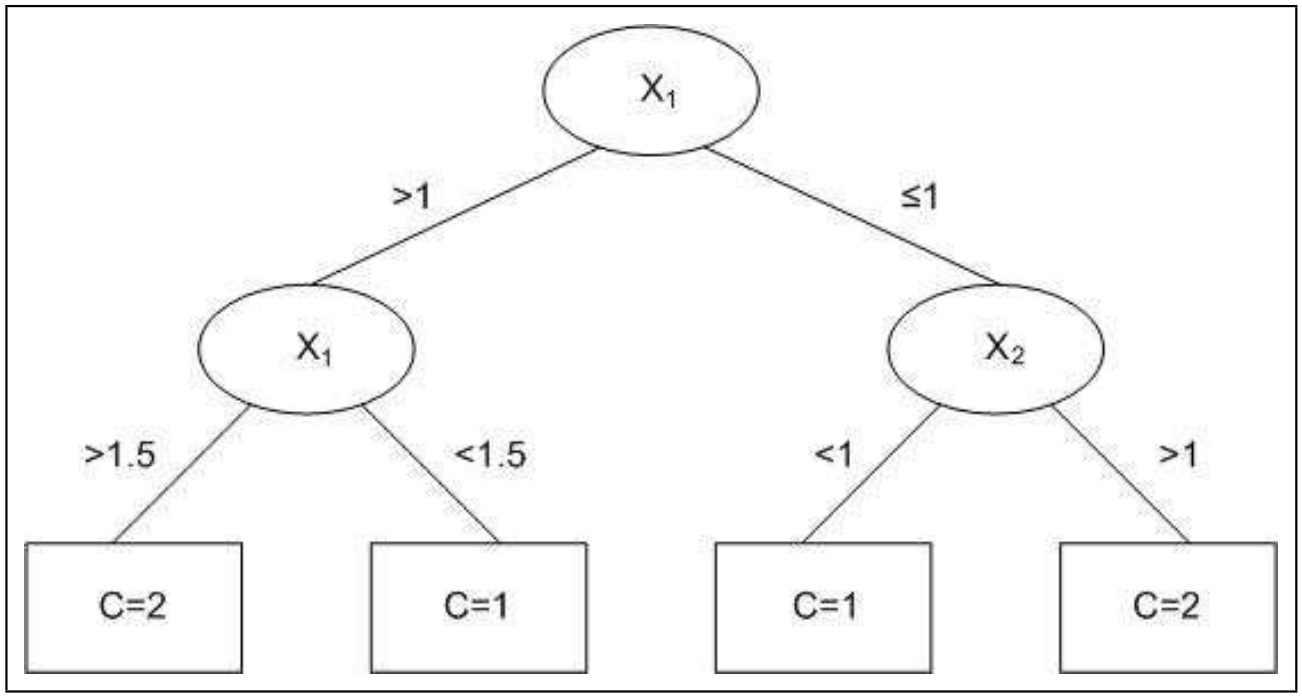
- Árbol de decisión
- Particionamiento recursivo
- Expresión en reglas `IF ... THEN ...`
- Cercanía con la manera en que los humanos estructuramos un dominio
- Alta transparencia

El algoritmo básico



Representación en el plano de distintos patrones caracterizados por 2 variables predictoras X_1 y X_2 y una variable clase C con dos posibles valores

El algoritmo básico



Árbol de clasificación correspondiente al ejemplo representado anterior

Conjunto de reglas equivalentes al árbol de clasificación:

R_1 : If $X_1 > 1{,}5$ then $C = 2$

R_2 : If $1 < X_1 < 1{,}5$ then $C = 1$

R_3 : If $X_1 < 1$ and $X_2 < 1$ then $C = 1$

R_4 : If $X_1 < 1$ and $X_2 > 1$ then $C = 2$

El algoritmo básico

Input: D conjunto de N patrones etiquetados, cada uno de los cuales está caracterizado por n variables predictoras $X_1, \ldots, X_n$ y la variable clase C

Output: Árbol de clasificación

Begin TDIDT (Top Down Induction of Decision Trees)

if todos los patrones de D pertenecen a la misma clase c

then

resultado de la inducción es un nodo simple (nodo hoja) etiquetado como c

else

begin

1. Seleccionar la variable más informativa X_r con valores $x_r^1, \ldots, x_r^{nr}$

2. Particionar D de acorde con los n_r valores de X_r en $D_1, \ldots, D_{nr}$

3. Construir n_r subárboles $T_1, \ldots, T_{nr}$ para $D_1, \ldots, D_{nr}$

4. Unir X_r y los n_r subárboles $T_1, \ldots, T_{nr}$ con los valores $x_r^1, \ldots, x_r^{nr}$

end

endif

End TDIDT

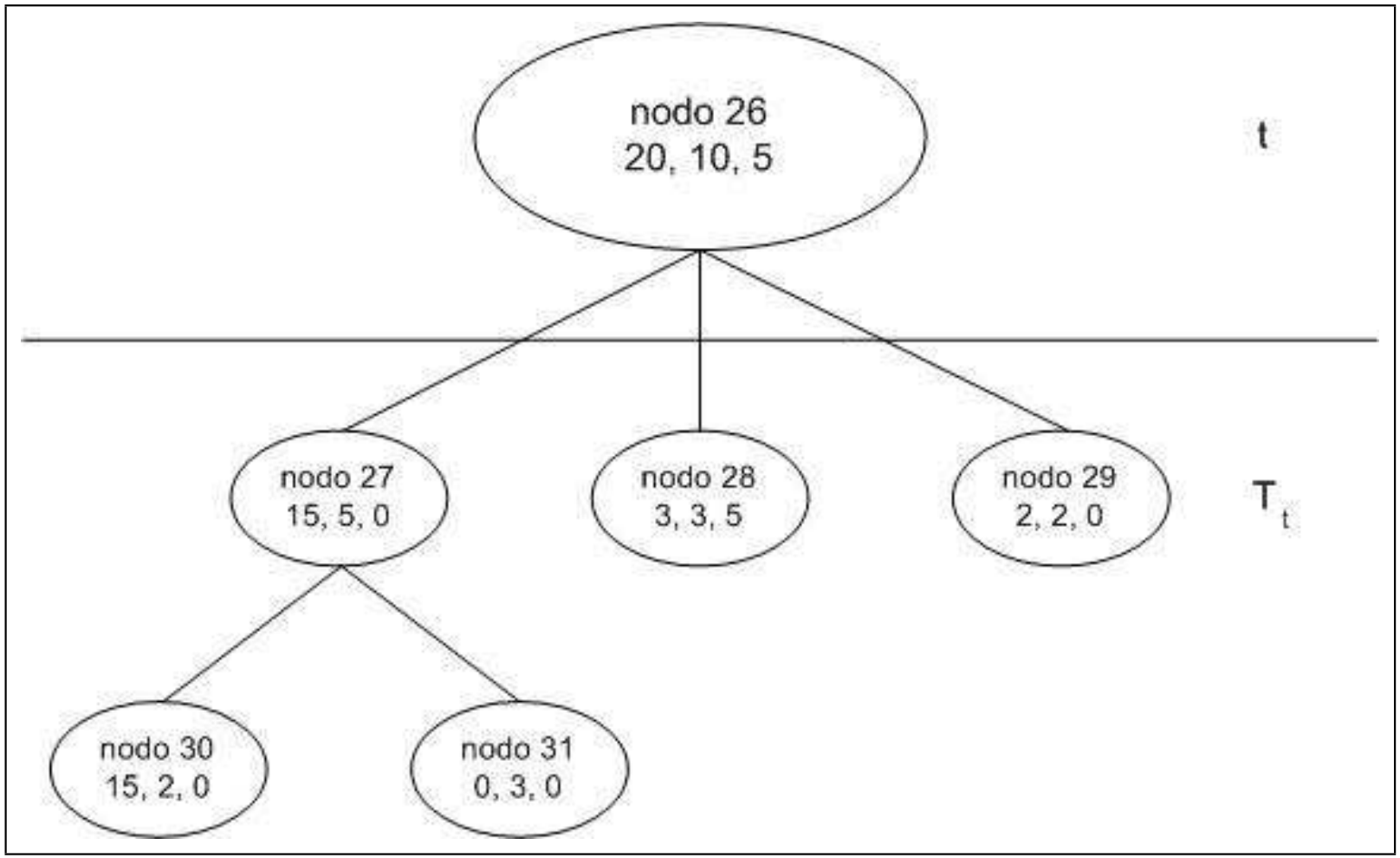
El algoritmo ID3

- ID3 (Quinlan, 1986) selecciona la variable más informativa en base a la cantidad de información mutua: $I(X_i, C) = H(C) - H(C|X_i)$ (*ganancia en información*)
- Matemáticamente se demuestra que este criterio favorece la elección de variables con mayor número de valores
- Selección de variables previa (*preprunning*) basada en un test de independencia entre cada variable predictora X_i y la variable clase C

El algoritmo C4.5

- C4.5 (Quinlan, 1993) selecciona la variable más informativa en base al *ratio de ganancia*: $I(X_i, C)/H(X_i)$
- Matemáticamente se demuestra que este criterio evita que se favorezca la elección de variables con mayor número de valores
- Incorporación de una poda del árbol inducido (*postpruning*), basada en un test de hipótesis que trata de responder a la pregunta de si merece la pena expandir o no una determinada rama

El algoritmo C4.5



Ejemplo para el proceso de pos-poda del algoritmo $C4.5$

El algoritmo C4.5

Proceso de poda del árbol

- $N(t) = 35$, ejemplos en el nodo $t = 26$
- $e(t) = 10 + 5 = 15$, ejemplos mal clasificados en el nodo t
- $n'(t) = e(t) + \frac{1}{2} = 15,5$, corrección por continuidad de $e(t)$
- T_t, subárbol a expandir a partir del nodo t
- $h(T_t) = 4$, número de hojas del subárbol T_t
- $n'(T_t) = \sum_{i=1}^{h(T_t)} e(i) + \frac{h(T_t)}{2} = 2 + 0 + 6 + 2 + \frac{4}{2} = 12$, número de errores existentes en las hojas terminales del subárbol T_t
- $S(n'(T_t)) = \sqrt{\frac{n'(T_t)[N(t)-n'(T_t)]}{N(t)}} = \sqrt{\frac{12(35-12)}{35}} \simeq 2,8$, desviación de $n'(T_t)$

El nodo t se expande $\Leftrightarrow n'(T_t) + S(n'(T_t)) < n'(t) \Leftrightarrow 12 + 2,8 < 15,5$

El nodo 26 se expande considerándose los nodos 28, 29, 30 y 31