Aprendizaje de árboles de decisión

José M. Sempere
Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia

Aprendizaje de árboles de decisión

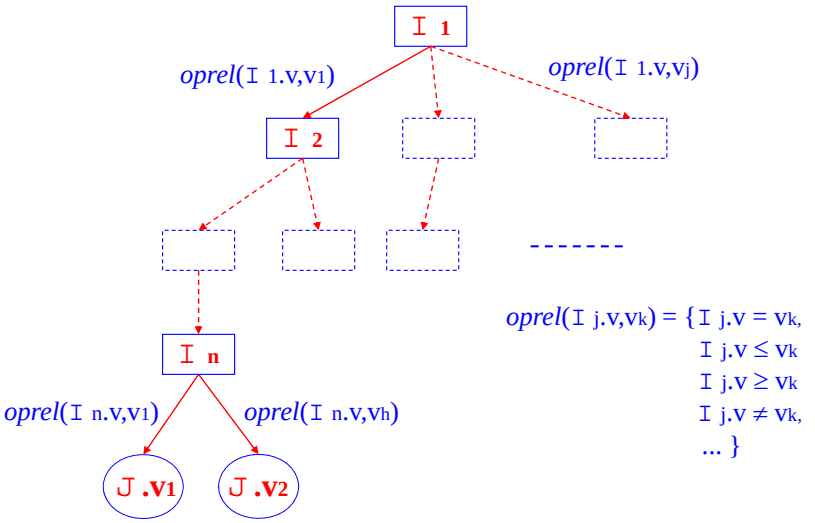
1. **Introducción. Definición y algoritmos de aprendizaje.**
2. **Conceptos básicos de teoría de la información**
3. **El algoritmo *ID3***
4. **Extensiones hacia el algoritmo *C4.5***
5. **Una versión incremental del *ID3*: El algoritmo ID5R**

Bibliografía

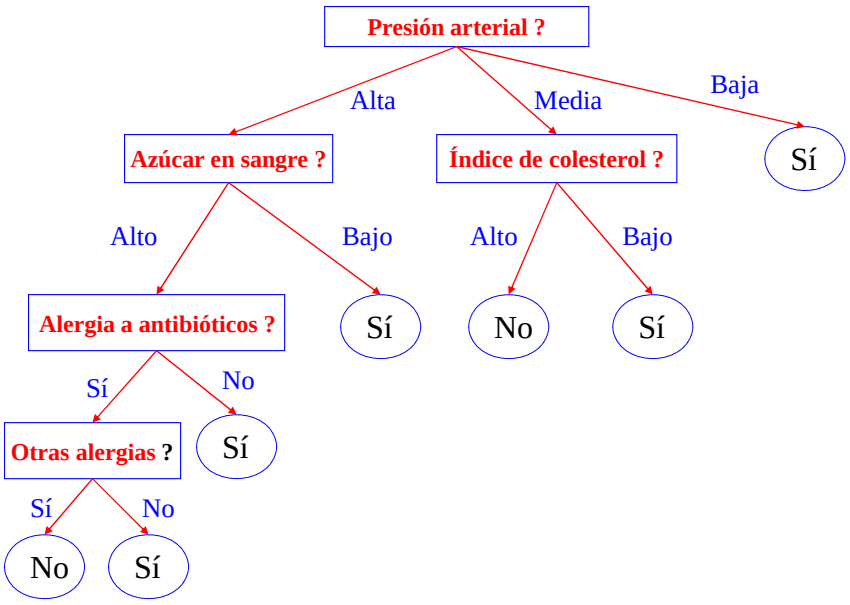
- **T. Mitchell. Machine Learning. Ed. McGraw-Hill. 1997.**
- **B. Sierra. Aprendizaje Automático. Ed. Pearson- Prentice Hall. 2006.**
- **J.R. Quinlan. C 4.5: programs for machine learning. Ed. Morgan Kaufmann. 1993.**

Un ***árbol de decisión*** es una representación de una función multievaluada

$n: I_1 \times I_2 \times \ldots \times I_n \rightarrow J$



Ejemplo: Árbol de decisión para "¿ *Administrar fármaco* F ?"



Los árboles de decisión son adecuados cuando ...

- Las instancias del concepto son representadas por pares atributo-valor
- La función objetivo tiene valores de salida discretos
- Las descripciones del objeto son disyuntivas
- El conjunto de aprendizaje tiene errores
- El conjunto de aprendizaje es incompleto

Algunos algoritmos para el aprendizaje de árboles de decisión

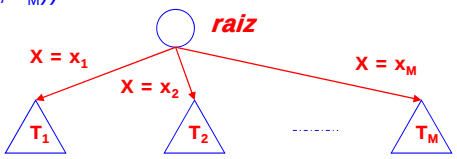
- **Hoveland y Hunt (1950):** ***Concept Learning Systems*** **(CLS)**
- **Breiman, Friendman, Olshen y Stone (1984): Método CART**
- **J.R. Quinlan (1973, 1986): Método ID3**
- **J.R. Quinlan (1994): Método C4.5**
- **G.V. Kass (1980): Método CHAID**
- **Otras mejoras del C4.5: J4.8, C5.0**
- **J. Schlimmer y D. Fisher (1986): ID4 e ID4R**
- **P. Utgoff (1990): ID5 e ID5R**

Un algoritmo genérico para el aprendizaje de árboles de decisión

Entrada: Un conjunto de ejemplos de aprendizaje (tuplas supervisadas) E
Salida: Un árbol de decisión T
Método:

Crear una *raiz* para el árbol.
si todos los ejemplos pertenecen a la clase C_j
entonces *return* (*raiz*,C_j)
sino
Seleccionar un atributo X con valores x_1, x_2, ..., x_M
Particionar E de acuerdo con los valores del atributo E_1, E_2, ..., E_M
Construir árboles de decisión para cada partición T_1, T_2, ..., T_M
return (raiz(T_1, T_2, ..., T_M))

finMétodo



Conceptos básicos de la teoría de la información (I)

Teoría de la probabilidad

Variables independientes

$$p(x,y)=p(x)p(y)$$

Probabilidad condicional y conjunta

$$p(x,y)=p(x|y)p(y)$$
$$p(x,y)=p(y|x)p(x)$$

***Teorema de Bayes* (*regla de Bayes*)**

$$p(x \mid y) = \frac{p(y \mid x)\,p(x)}{p(y)}$$

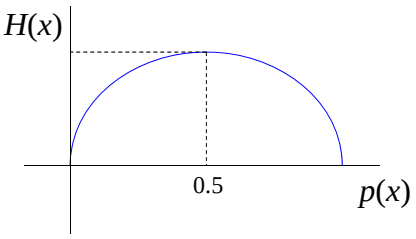
Conceptos básicos de la teoría de la información (II)

Entropía

$$H(X) = -\sum_{i=1}^{n} p(X = x_i) \log_2 p(X = x_i)$$

$$H(X \mid y) = -\sum_{x} p(x \mid y) \log_2 p(x \mid y)$$

$$H(X \mid Y) = -\sum_{y} \sum_{x} p(y)\, p(x \mid y) \log_2 p(x \mid y)$$



Teorema. (1) $H(X,Y) \leq H(X) + H(Y)$
(2) $H(X,Y) = H(X) + H(Y)$ sii X e Y son independientes

Teorema. $H(X,Y) = H(Y) + H(X \mid Y) = H(X) + H(Y \mid X)$

Corolario (1) $H(X \mid Y) \leq H(X)$
(2) $H(X \mid Y) = H(X)$ *sii* X *e* Y *son independientes.*

El algoritmo de aprendizaje de árboles de decisión ID3

J.R Quinlan (1986)

- **Búsqueda voraz top-down**
- **Basado en un criterio estadístico**
- **Selección de atributos mediante el** Principio de ganancia de información

S: conjunto de ejemplos clasificados en C clases
A: Atributo de los ejemplos
S_v: Ejemplos que en el atributo A tienen el valor v

$$Ganancia(S, A) \equiv Entropía(S) - \sum_{v \in Valores(A)} \frac{|S_v|}{|S|} Entropía(S_v)$$

***Algoritmo ID3*(*Ejemplos, Atributo_salida, Atributos*)**

***Ejemplos*: Ejemplos de aprendizaje.**
***Atributo_salida*: Atributo a predecir por el árbol.**
***Atributos*: Lista de atributos a comprobar por el árbol.**

(1) Crear una *raiz* para el árbol.
(2) **Si** todos los ejemplos son positivos **Return**(*raiz*,+)
(3) **Si** todos los ejemplos son negativos **Return**(*raiz*,-)
(4) **Si** *Atributos*=∅ **Return**(*raiz*,*l*)
(*l* es el máximo valor común de *Atributo_salida* en *Ejemplos*)
Si ninguna de las anteriores condiciones se cumple
Begin
(1) Seleccionar el atributo *A* con mayor *Ganancia*(Ejemplos,A)
(2) El atributo de decisión para *raiz* es *A*
(3) Para cada posible valor v_i de *A*
(3.1) Añadir una rama a *raiz* con el test $A=v_i$
(3.2) $Ejemplos\_v_i$ es el subconjunto de *Ejemplos* con valor v_i para *A*
(3.3) **Si** $Ejemplos\_v_i=\emptyset$
entonces añadir un nodo (*n*,*l*) a partir de la rama creada.
(*l* es el máximo valor común de *Atributo_salida* en *Ejemplos*).
sino añadir a la rama creada el subárbol
ID3($Ejemplos\_v_i$, *Atributo_salida*, *Atributos*-{*A*})
End

Return(*raiz*)

Un ejemplo: ¿Administrar fármaco F?

Paciente	Presión arterial	Azúcar en sangre	Índice de colesterol	Alergia a antibióticos	Otras alergias	Administrar fármaco F
1	Alta	Alto	Alto	No	No	Sí
2	Alta	Alto	Alto	Sí	No	Sí
3	Baja	Alto	Bajo	No	No	Sí
4	Media	Alto	Alto	No	Sí	No
5	Media	Bajo	Alto	Sí	Sí	No
6	Baja	Bajo	Alto	Sí	Sí	Sí
7	Alta	Bajo	Alto	Sí	No	Sí
8	Alta	Bajo	Bajo	No	Sí	Sí
9	Alta	Alto	Bajo	Sí	Sí	No
10	Baja	Bajo	Alto	Sí	Sí	Sí
11	Media	Bajo	Bajo	Sí	Sí	Sí
12	Alta	Bajo	Alto	Sí	Sí	No
13	Baja	Alto	Alto	Sí	Sí	Sí
14	Baja	Alto	Bajo	No	No	Sí

Cálculo de entropías y ganancia de la información respecto del atributo *Presión arterial*

$$Entropía\ (S) = \sum_{i=1}^{c} - p_i \log_2 p_i = -\frac{10}{14}\log_2\frac{10}{14} - \frac{4}{14}\log_2\frac{4}{14} = 0.863121$$

$$Entropía(S_{PA=Alta}) = -\frac{4}{6}\log_2\frac{4}{6} - \frac{2}{6}\log_2\frac{2}{6} = 0.918296$$

$$Entropía\ (S_{PA=Baja}) = -\frac{5}{5}\log_2\frac{5}{5} = 0$$

$$Entropía(S_{PA=Media}) = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} = 0.918296$$

$$Ganancia(S, PA) = 0.863121 - \frac{6}{14}0.918296 - \frac{5}{14}0 - \frac{3}{14}0.918296 = 0.272788$$

Cálculo de entropías y ganancia de la información respecto del atributo *Azúcar en sangre*

$$Entropía(S) = \sum_{i=1}^{c} - p_i \log_2 p_i = -\frac{10}{14}\log_2\frac{10}{14} - \frac{4}{14}\log_2\frac{4}{14} = 0.863121$$

$$Entropía(S_{AS=Alto}) = -\frac{5}{7}\log_2\frac{5}{7} - \frac{2}{7}\log_2\frac{2}{7} = 0.863121$$

$$Entropía(S_{AS=Bajo}) = -\frac{5}{7}\log_2\frac{5}{7} - \frac{2}{7}\log_2\frac{2}{7} = 0.863121$$

$$Ganancia(S, AS) = 0.863121 - \frac{7}{14}0.863121 - \frac{7}{14}0.863121 = 0$$

Cálculo de entropías y ganancia de la información respecto del atributo *Índice de colesterol*

$$Entropía(S) = \sum_{i=1}^{c} - p_i \log_2 p_i = -\frac{10}{14}\log_2\frac{10}{14} - \frac{4}{14}\log_2\frac{4}{14} = 0.863121$$

$$Entropía(S_{IC=Alto}) = -\frac{6}{9}\log_2\frac{6}{9} - \frac{3}{9}\log_2\frac{3}{9} = 0.918296$$

$$Entropía(S_{IC=Bajo}) = -\frac{4}{5}\log_2\frac{4}{5} - \frac{1}{5}\log_2\frac{1}{5} = 0.721928$$

$$Ganancia(S, IC) = 0.863121 - \frac{9}{14}0.918296 - \frac{5}{14}0.721928 = 0.0149564$$

Cálculo de entropías y ganancia de la información respecto del atributo *Alergia a antibióticos*

$$Entropía(S) = \sum_{i=1}^{c} - p_i \log_2 p_i = -\frac{10}{14}\log_2\frac{10}{14} - \frac{4}{14}\log_2\frac{4}{14} = 0.863121$$

$$Entropía(S_{AA=SI}) = -\frac{6}{9}\log_2\frac{6}{9} - \frac{3}{9}\log_2\frac{3}{9} = 0.918296$$

$$Entropía(S_{AA=NO}) = -\frac{4}{5}\log_2\frac{4}{5} - \frac{1}{5}\log_2\frac{1}{5} = 0.721928$$

$$Ganancia(S, AA) = 0.863121 - \frac{9}{14}0.918296 - \frac{5}{14}0.721928 = 0.0149564$$

Cálculo de entropías y ganancia de la información respecto del atributo *Otras alergias*

$$Entropía(S) = \sum_{i=1}^{c} - p_i \log_2 p_i = -\frac{10}{14}\log_2\frac{10}{14} - \frac{4}{14}\log_2\frac{4}{14} = 0.863121$$

$$Entropía(S_{OA=SI}) = -\frac{5}{9}\log_2\frac{5}{9} - \frac{4}{9}\log_2\frac{4}{9} = 0.991076$$

$$Entropía(S_{OA=NO}) = -\frac{5}{5}\log_2\frac{5}{5} = 0$$

$$Ganancia(S, OA) = 0.863121 - \frac{9}{14}0.991076 - \frac{5}{14}0 = 0.226001$$

Selección del mejor atributo que explica las decisiones

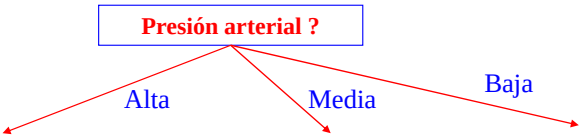
Ganancia(S,PA) = 0.272788

Ganancia(S,AS) = 0

Ganancia(S,IC) = 0.0149564

Ganancia(S,AA) = 0.0149564

Ganancia(S,OA) = 0.226001



Paciente	Presión arterial	Azúcar en sangre	Índice de colesterol	Alergia a antibióticos	Otras alergias	Administrar fármaco F
1	Alta	Alto	Alto	No	No	Sí
2	Alta	Alto	Alto	Sí	No	Sí
3	Baja	Alto	Bajo	No	No	Sí
4	Media	Alto	Alto	No	Sí	No
5	Media	Bajo	Alto	Sí	Sí	No
6	Baja	Bajo	Alto	Sí	Sí	Sí
7	Alta	Bajo	Alto	Sí	No	Sí
8	Alta	Bajo	Bajo	No	Sí	Sí
9	Alta	Alto	Bajo	Sí	Sí	No
10	Baja	Bajo	Alto	Sí	Sí	Sí
11	Media	Bajo	Bajo	Sí	Sí	Sí
12	Alta	Bajo	Alto	Sí	Sí	No
13	Baja	Alto	Alto	Sí	Sí	Sí
14	Baja	Alto	Bajo	No	No	Sí

Tabla de datos para calcular el subárbol de PA=Alta

Paciente	Azúcar en sangre	Índice de colesterol	Alergia a antibióticos	Otras alergias	Administrar fármaco F
1	Alto	Alto	No	No	Sí
2	Alto	Alto	Sí	No	Sí
7	Bajo	Alto	Sí	No	Sí
8	Bajo	Bajo	No	Sí	Sí
9	Alto	Bajo	Sí	Sí	No
12	Bajo	Alto	Sí	Sí	No

Características del *ID3*

Espacio de hipótesis completo

Hipótesis única en cada momento de tiempo

No se realiza "*backtracking*"

Búsqueda no incremental

Principio de "*la navaja de Occam*" (MDL)

Saturación sobre los datos ("*overfitting*")

Hacia el algoritmo de aprendizaje de árboles de decisión C4.5

El problema de la saturación (*"overfitting"*)

Dado un espacio de hipótesis H, una hipótesis h ∈ H diremos que satura un conjunto de aprendizaje si existe otra hipótesis h' tal que h tiene menor error que h' sobre el conjunto de aprendizaje, pero h' tiene menor error que h sobre la distribución total de instancias de aprendizaje.

Posible solución (C4.5)

Utilizad un conjunto de aprendizaje A y un conjunto de validación V.

1. Inferir el árbol con el conjunto A
2. Establecer todas las posibles podas del árbol (convirtiendo los caminos desde la raíz en reglas de decisión y eliminando precondiciones)
3. Para cada poda medir el error respecto del conjunto V
4. Ordenad los mejores resultados y aplicadlos en la fase de test.

Si (PA=Media) ∧ (IC=alto) entonces NO administrar F ⇒

Si (PA=Media) entonces NO administrar F

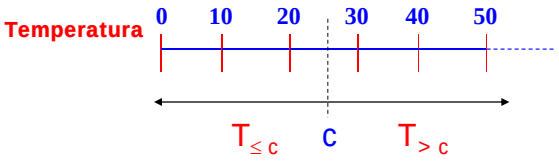
Si (IC=alto) entonces NO administrar F

Hacia el algoritmo de aprendizaje de árboles de decisión C4.5

Evaluación de atributos continuos

Cómo incorporar atributos continuos en las fases de aprendizaje y de test:

Dada una variable x de carácter continuo, estableced los intervalos adecuados en sus valores para proporcionar variables discretas



Problema: ¿ Cómo seleccionar el (los) valor(es) de c ?
Posible solución: Seleccionad aquellos valores que mayor ganancia de información proporcionen

Hacia el algoritmo de aprendizaje de árboles de decisión C4.5

Incorporación de otras medidas para la selección de atributos

Problema: La medida Ganancia favorece aquellas variables con mayor número de posibles valores

Posible solución
Dados S (un conjunto de ejemplos de aprendizaje) y A (un atributo de los ejemplos que puede tomar c posibles valores) definimos ...

$$SplitInformation(S,A) \equiv -\sum_{i=1}^{c} \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}$$

SplitInformation(S,A) denota la entropía de S con respecto a los valores de A

$$RatiodeGanancia(S,A) \equiv \frac{Ganancia(S,A)}{SplitInformation(S,A)}$$

El **RatiodeGanacia(S,A)** favorece aquellos atributos que, en igualdad de Ganacia, separen los datos en menos clases.

Hacia el algoritmo de aprendizaje de árboles de decisión C4.5

Manejo de ejemplos incompletos (atributos no evaluados)

Problema: Dado un conjunto de ejemplos ¿ qué hacer cuando algunos atributos no tienen valor ?

Posibles soluciones

(1) Estimar el valor desconocido como el valor mayoritario que aparece en el el resto de ejemplos

(2) Asignar a cada posible valor una probabilidad (frecuencia) de acuerdo con el resto de ejemplos. A continuación repartir el ejemplo en cada uno de sus valores de acuerdo con la probabilidad y hacer el cálculo de la Ganancia

En el caso de la clasificación, los casos con valores desconocidos se clasifican de acuerdo con la mayor probabilidad que proporcione el árbol.

Hacia el algoritmo de aprendizaje de árboles de decisión C4.5

Introduciendo costes en los atributos

Problema: ¿ Todos los atributos son igual de valiosos al hacer una clasificación ?

Posible solución

Incorporar el coste de evaluar cada atributo a la hora de estimar el mejor de todos ellos. Algunas medidas pueden ser las siguientes

$$\frac{Ganancia^2(S,A)}{Coste(A)} \qquad \frac{2^{Ganancia(S,A)}-1}{(Coste(A)+1)^w}$$

w **es una constante entre 0 y 1 que evalúa la importancia del Coste frente a la Ganancia**

Una versión incremental del algoritmo ID3: *ID5R*

(para versiones dicotómicas en la decisión)

Nueva información en los nodos de decisión

Sea A el conjunto de atributos presentes en los ejemplos.
Sea a_i el i-ésimo atributo de un ejemplo
Sea V_i el conjunto de valores posibles para a_i
Sea v_{ij} el valor j-esimo del atributo a_i

Definimos la función E

$$E(a_i)=\sum_{j=1}^{|V_i|}\frac{p_{ij}+n_{ij}}{p+n}I(p_{ij},n_{ij})$$

p = número de ejemplos positivos
n = número de ejemplos negativos
p_{ij} = número de ejemplos positivos con valor v_{ij}
n_{ij} = número de ejemplos negativos con valor v_{ij}

$$I(x,y)=\begin{cases}0 & \text{si } x=0\\ 0 & \text{si } y=0\\ -\frac{x}{x+y}\log\frac{x}{x+y}-\frac{y}{x+y}\log\frac{y}{x+y} & \text{en otro caso}\end{cases}$$

Algoritmo ID5R(*T, Ejemplo*)

***T* : árbol en curso**
***Ejemplo* : Nuevo ejemplo de aprendizaje.**
***Atributo_salida*: Atributo a predecir por el árbol.**
***Atributos*: Lista de atributos a comprobar por el árbol.**

si *T* es nulo
entonces definid un árbol trivial con *Ejemplo*
sino si T sólo contiene una raíz de la misma clase que *Ejemplo*
entonces Incorporad *Ejemplo* a la raíz
sino Begin

(1) **si** *T* sólo contiene una raíz de distinta clase que *Ejemplo*
entonces Expandir el árbol un nivel eligiendo el Atributo de *Atributos* de forma arbitraria
(2) Actualizad los contadores de instancias (+,-) en cada valor de cada atributo
(3) **si** la raíz actual contiene un *Atributo* con un valor E que no es minimal
entonces (3.a) Reestructurad el árbol situando en la raíz un nodo con E minimal
(3.b) Recursivamente situad el mejor nodo en cada subárbol excepto en el nodo referido en (4)
(4) Recursivamente actualizad el subárbol dependiente de cada nodo de acuerdo con *Ejemplo*

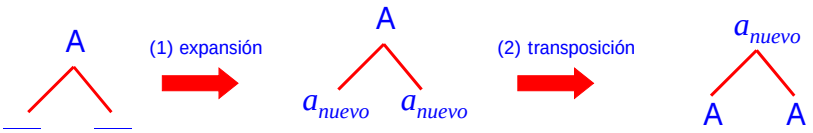
End

Reestructurando árboles *(pull-up)*

La reestructuración de un árbol consiste en situar en los nodos superiores aquellos atributos que mejor expliquen la clasificación de instancias de acuerdo con el indicador *E*. A este proceso se le denomina "*pull-up*" y obedece al siguiente esquema algorítmico

si el atributo a subir a_{nuevo} está en la raíz
entonces fin_del_método
sino (1) Recursivamente subid el atributo a_{nuevo} a la raíz del subárbol inmediato. Convertid cualquier cualquier árbol no expandido en uno expandido eligiendo a_{nuevo} como el atributo a testear
(2) Transponer el árbol de forma que a_{nuevo} se sitúe en la raíz y la antigua raíz como raíz de cada subárbol dependiente de a_{nuevo}

Ejemplo



Un ejemplo (I)

Atributo de salida	Altura	Color de pelo	Color de ojos
-	Bajo	Rubio	Marrones
-	Alto	Moreno	Marrones
+	Alto	Rubio	Azules
-	Alto	Moreno	Azules
-	Bajo	Moreno	Azules
+	Alto	Rojo	Azules
-	Alto	Rubio	Marrones
+	Bajo	Rubio	Azules

Un ejemplo (II)

Inicialmente el árbol es nulo

Ejemplo = {-, Bajo, Rubio, Marrones}

Salida

(Altura=Bajo, Pelo=Rubio, Ojos=Marrones)

Ejemplo = {-, Alto, Moreno, Marrones}

Salida

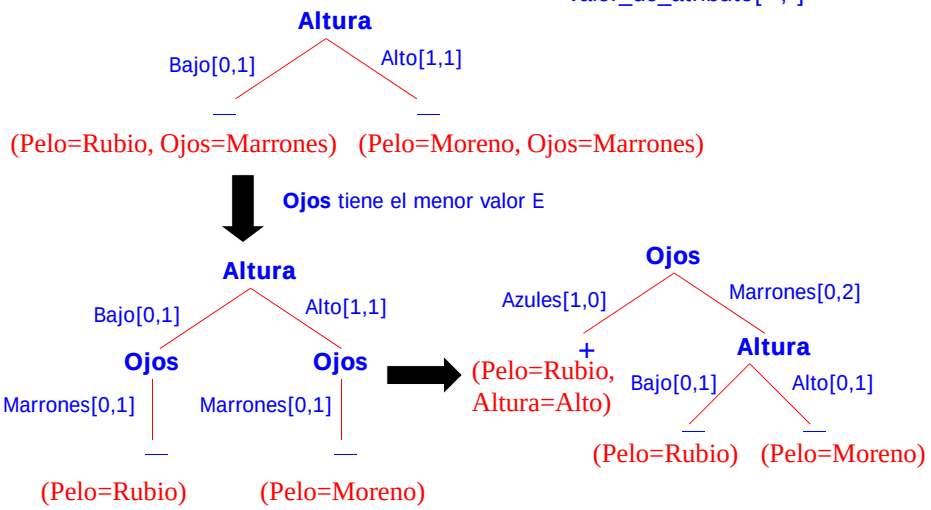
(Altura=Bajo, Pelo=Rubio, Ojos=Marrones)
(Altura=Alto, Pelo=Moreno, Ojos=Marrones)

Un ejemplo (III)

Ejemplo = {+, Alto, Rubio, Azules}

Salida

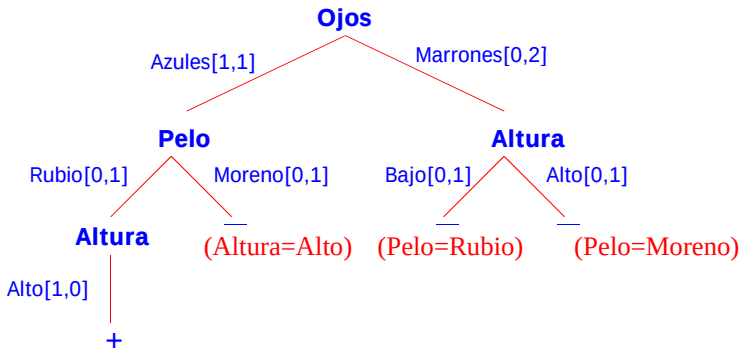
valor_de_atributo[+,-]



Un ejemplo (IV)

Ejemplo = {-, Alto, Moreno, Azules}

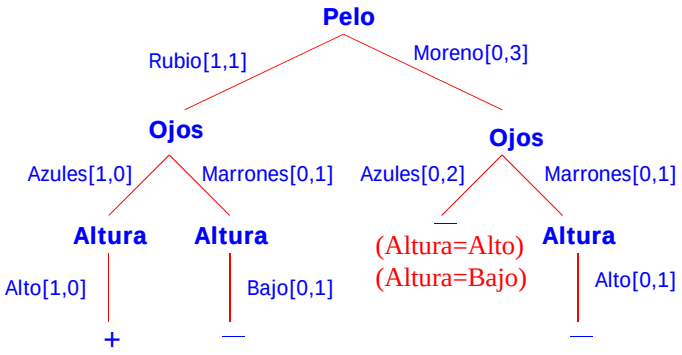
Salida



Un ejemplo (V)

Ejemplo = {-, Bajo, Moreno, Azules}

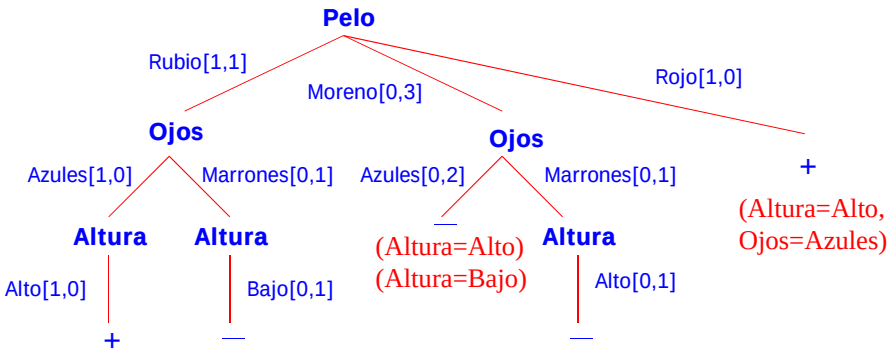
Salida



Un ejemplo (VI)

Ejemplo = {+, Alto, Rojo, Azules}

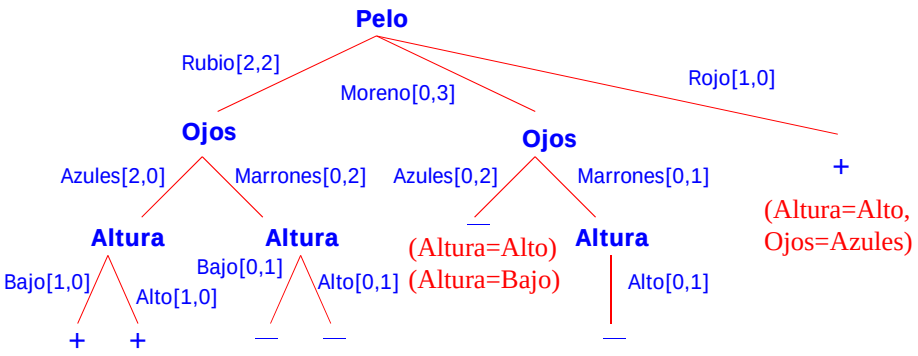
Salida



Un ejemplo (VII)

Ejemplo = {-, Alto, Rubio, Marrones}
Ejemplo = {+, Bajo, Rubio, Azules}

Salida: La estructura de los atributos del árbol no cambia.
Los contadores se actualizan.



El árbol del ejemplo anterior en formato *ID3*