Aprendizaje automático

Daniel Borrajo Millán

Universidad Carlos III de Madrid
Avda. de la Universidad, 30
28911 Madrid, SPAIN

Teléfono: +(34 91) 624 9459
Email: dborrajo@ia.uc3m.es
Web: http://scalab.uc3m.es/∼dborrajo

Aprendizaje Automático

- Introducción
- Aprendizaje de árboles de decisión
- Aprendizaje basado en casos
- Algoritmos genéticos

`http://www.aic.nrl.navy.mil/~aha/research/machine-learning.html`

Introducción

Aprendizaje y adaptación

Psicología: no se pueden separar los términos **inteligencia** y **aprendizaje** (adaptación)

Biología: la adaptación, en todas sus vertientes, no se puede separar de nosotros

Arte: permite **nuevas** formas de creación

Matemáticas: estadística e inducción son dos formas clásicas de análisis de datos y generalización

Ingeniería: conceptos como adaptación o realimentación son clave

Informática: en la mayor parte de los campos se ha llegado al límite de lo que se puede hacer y, como siempre, se debe recurrir a la Inteligencia Artificial para que nos proporcione nuevas soluciones

Introducción

- El **desarrollo de software** es un cuello de botella
- Introducir **conocimiento a través de ejemplos** es atractivo
- Especialmente cierto en los problemas:
  - ⋆ en los que no existen algoritmos
  - ⋆ mal definidos
  - ⋆ propuestos informalmente
- La sociedad de la información genera una **explosión de datos**
- **No hay suficiente gente** que pueda analizar tal cantidad de datos
- Se tiende hacia la **personalización**: adaptación al individuo
- Es muy importante la **comprensibilidad de la salida**

Árboles de decisión. ID3

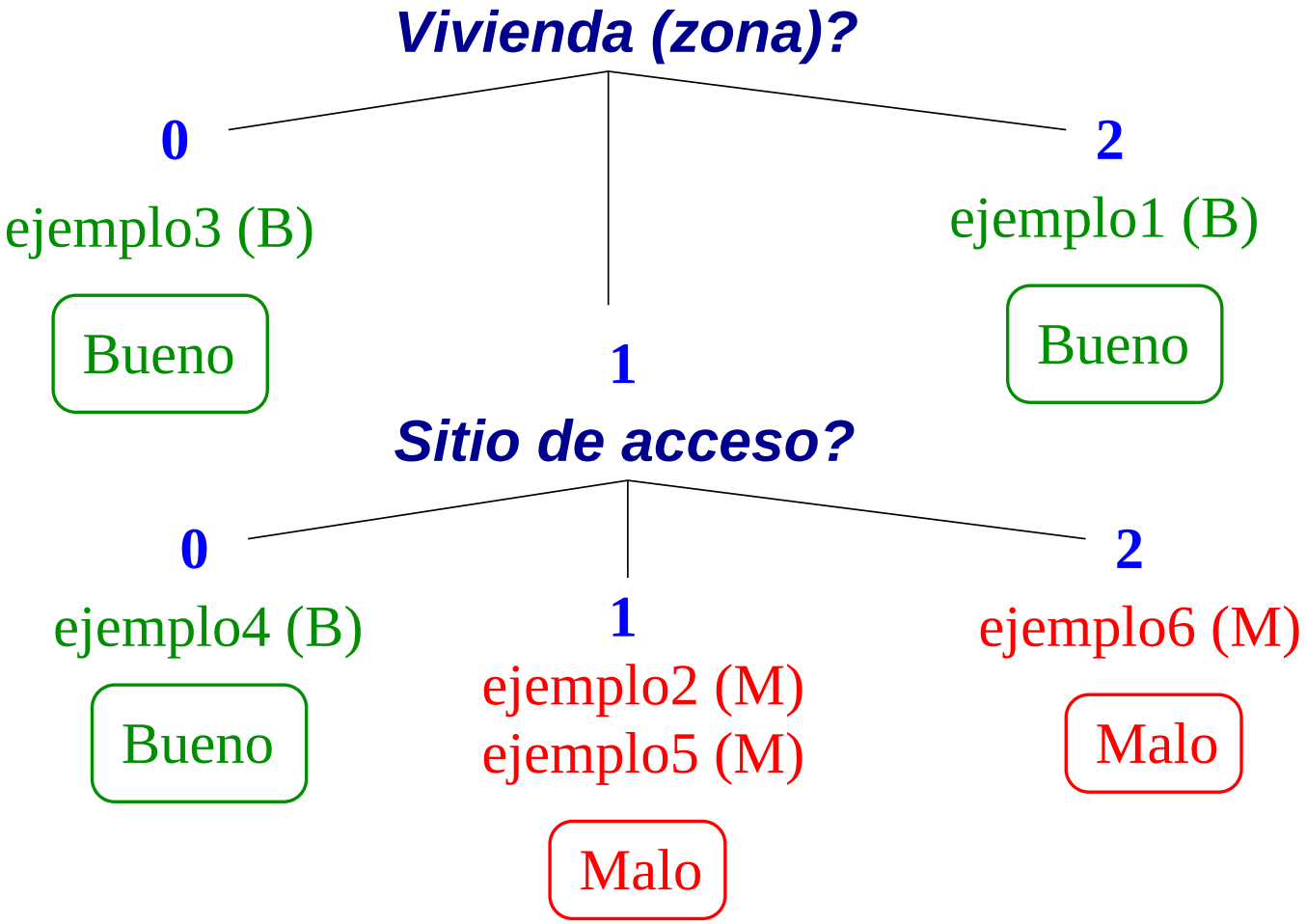
ID3 (Quinlan, 83)

- CLS (Hunt, Marin, y Stone, 66) fue el precursor de ID3
  - ⋆ Utilizaba sólo atributos binarios
  - ⋆ Tenía heurísticas para decidir qué atributo escoger
- Precursor de un conjunto de técnicas que han tenido mucho éxito comercial
- Genera árboles de decisión a partir de ejemplos de partida
- Intenta encontrar el árbol más sencillo que separa mejor los ejemplos
- Utiliza la entropía para elegir

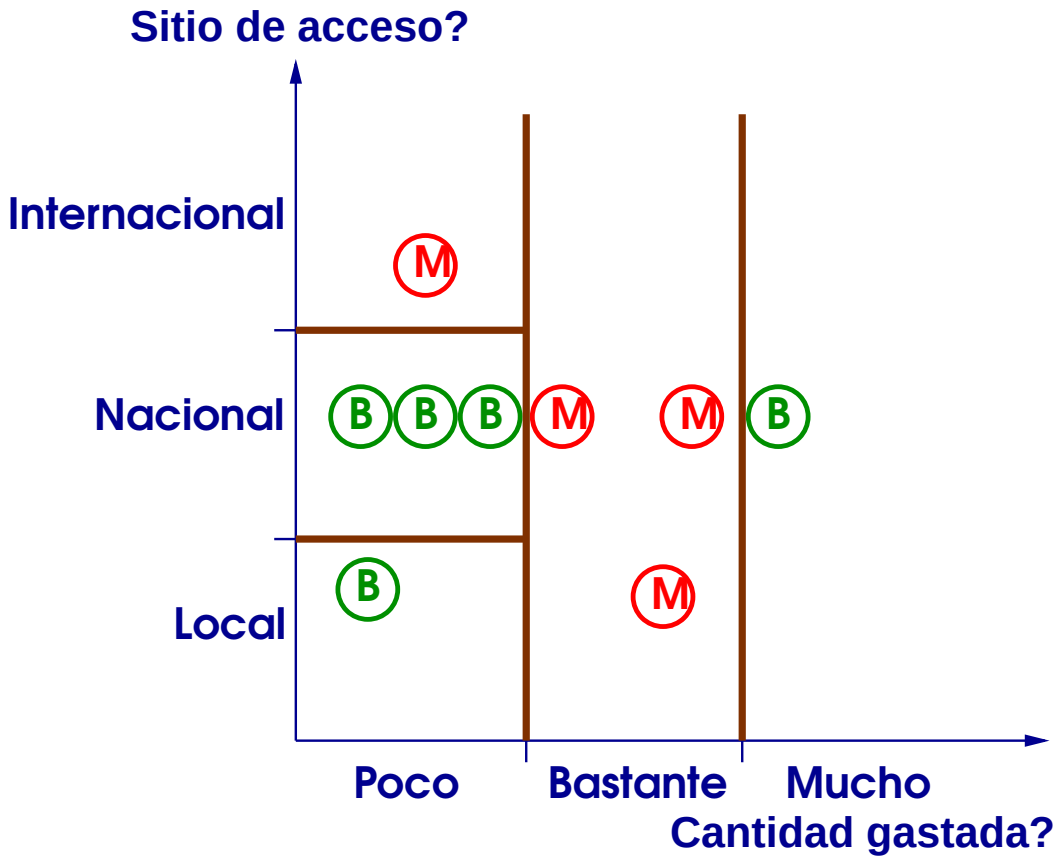
Ejemplo de entrada

Ejemplo	Sitio de acceso A_1	1ª cantidad gastada A_2	Vivienda (zona) A_3	Última compra A_4	Clase
1	1	0	2	Libro	Bueno
2	1	0	1	Disco	Malo
3	1	2	0	Libro	Bueno
4	0	2	1	Libro	Bueno
5	1	1	1	Libro	Malo
6	2	2	1	Libro	Malo

Ejemplo de árbol de decisión



Estrategia del ID3



Algoritmo

1. Seleccionar el atributo A_i que maximice la ganancia $G(A_i)$

2. Crear un nodo para ese atributo con tantos sucesores como valores tenga

3. Introducir los ejemplos en los sucesores según el valor que tenga el atributo A_i

4. Por cada sucesor,

   Si sólo hay ejemplos de una clase c_k
   Entonces etiquetarlo con c_k
   Si no, llamar al ID3 con una tabla formada por los ejemplos
   de ese nodo, eliminando la columna del atributo A_i

Heurística

- Seleccionar el atributo que mejor separe (ordene) los ejemplos de acuerdo a las clases
- La entropía es una medida de cómo está ordenado el universo
- La teoría de la información (basada en la entropía) calcula el número de bits (información, preguntas sobre atributos) que hace falta suministrar para conocer la clase a la que pertenece un ejemplo

Fórmulas

Se puede medir lo que discrimina (se gana por usar) un atributo A_i como:

$$G(A_i) = I - I(A_i)$$

donde

$$I(A_i) = \sum_{j=1}^{nv(A_i)} \frac{n_{ij}}{n} I_{ij}$$

$$I_{ij} = -\sum_{k=1}^{nc} \frac{n_{ijk}}{n_{ij}} \log_2 \frac{n_{ijk}}{n_{ij}}$$

Ejemplo de ID3

Ejemplo	Sitio de acceso A_1	1ª cantidad gastada A_2	Vivienda (zona) A_3	Última compra A_4	Clase
1	1	0	2	Libro	Bueno
2	1	0	1	Disco	Malo
3	1	2	0	Libro	Bueno
4	0	2	1	Libro	Bueno
5	1	1	1	Libro	Malo
6	2	2	1	Libro	Malo

Ejemplo de ID3

$$I(A_1) = \sum_{j=1}^{nv(A_1)} \frac{n_{ij}}{n} I_{ij} = \sum_{j=1}^{3} \frac{n_{ij}}{6} I_{ij} =$$
$$\frac{n_{10}}{6} I_{10} + \frac{n_{11}}{6} I_{11} + \frac{n_{12}}{6} I_{12} = \frac{1}{6} I_{10} + \frac{4}{6} I_{11} + \frac{1}{6} I_{12}$$
$$I_{10} = -\sum_{k=1}^{2} \frac{n_{10k}}{n_{10}} \log_2 \frac{n_{10k}}{n_{10}} = -\frac{1}{1} \log_2 \frac{1}{1} - \frac{0}{1} \log_2 \frac{0}{1} = 0$$
$$I_{11} = -\sum_{k=1}^{2} \frac{n_{11k}}{n_{11}} \log_2 \frac{n_{11k}}{n_{11}} = -\frac{2}{4} \log_2 \frac{2}{4} - \frac{2}{4} \log_2 \frac{2}{4} = 1$$
$$I_{12} = -\sum_{k=1}^{2} \frac{n_{12k}}{n_{12}} \log_2 \frac{n_{12k}}{n_{12}} = -\frac{0}{1} \log_2 \frac{0}{1} - \frac{1}{1} \log_2 \frac{1}{1} = 0$$

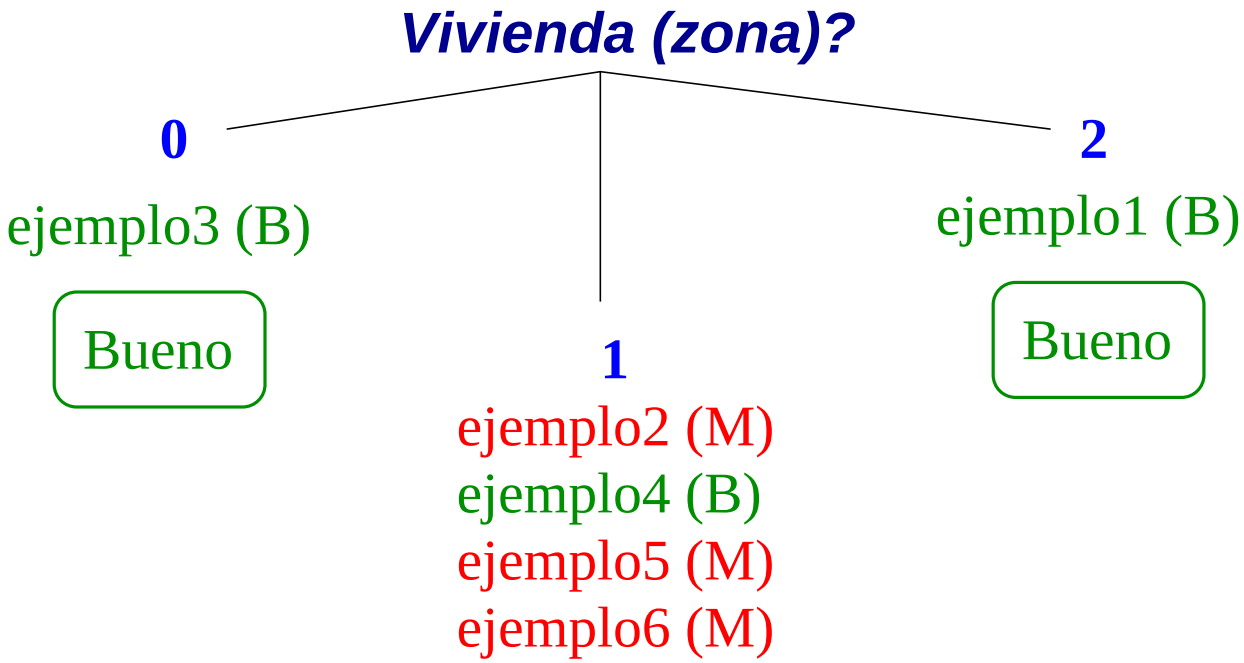
$$I(A_1) = \frac{1}{6} I_{10} + \frac{4}{6} I_{11} + \frac{1}{6} I_{12} = \frac{1}{6} 0 + \frac{4}{6} 1 + \frac{1}{6} 0 = 0,66$$

$$I(A_2) = \frac{2}{6} I_{20} + \frac{1}{6} I_{21} + \frac{3}{6} I_{22} = \frac{2}{6} 1 + \frac{1}{6} 0 + \frac{3}{6} (-\frac{2}{3} \log_2 \frac{2}{3} - \frac{1}{3} \log_2 \frac{1}{3}) = 0,79$$

$$I(A_3) = \frac{1}{6} I_{30} + \frac{4}{6} I_{31} + \frac{1}{6} I_{32} = \frac{1}{6} 0 + \frac{4}{6} (-\frac{1}{4} \log_2 \frac{1}{4} - \frac{3}{4} \log_2 \frac{3}{4}) + \frac{1}{6} 0 = 0,54$$

$$I(A_4) = \frac{1}{6} I_{4Disco} + \frac{5}{6} I_{4Libro} = \frac{1}{6} 0 + \frac{5}{6} (-\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5}) = 0,81$$

Ejemplo de ID3



Ejemplo de ID3

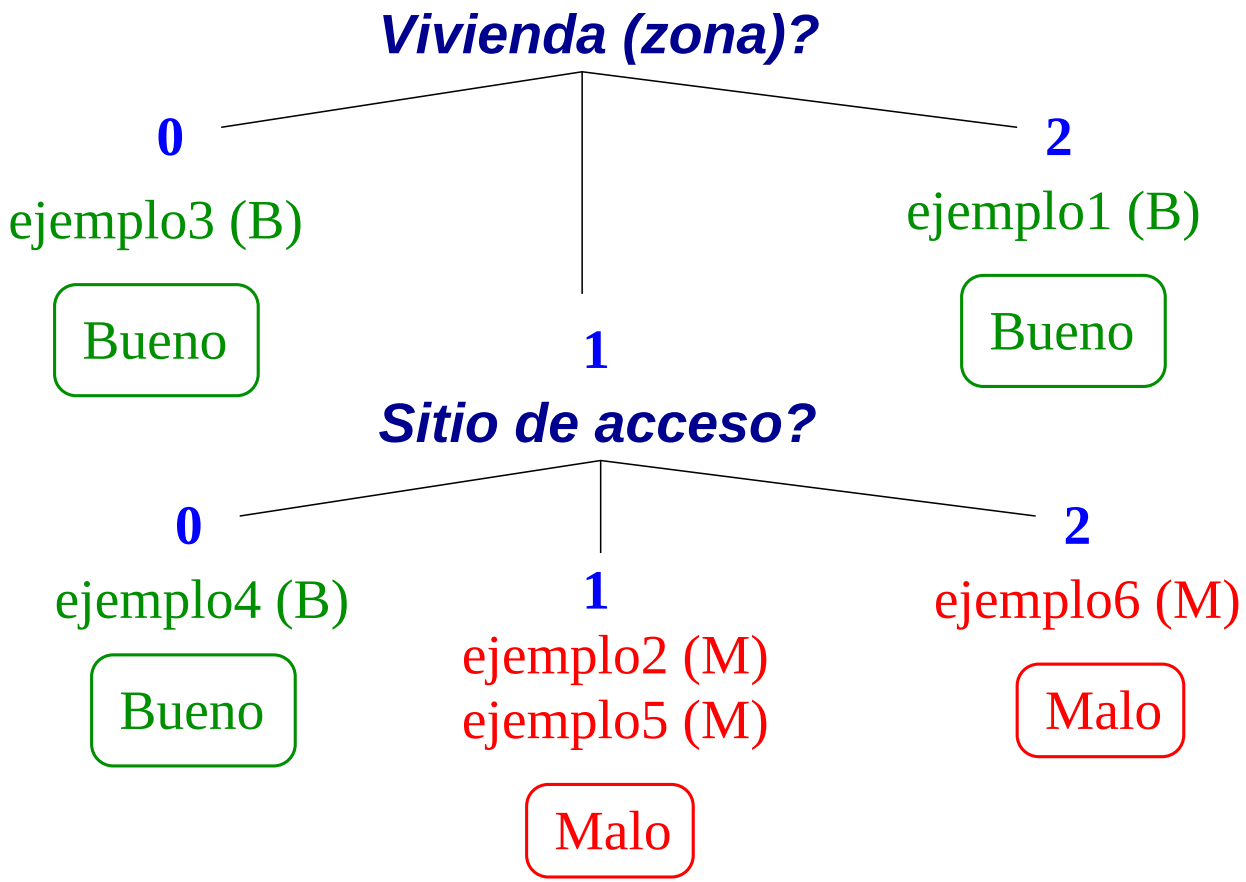
Ejemplo	Sitio de acceso A_1	1ª cantidad gastada A_2	Última compra A_4	Clase
2	1	0	Disco	Malo
4	0	2	Libro	Bueno
5	1	1	Libro	Malo
6	2	2	Libro	Malo

$$I(A_1) = \frac{1}{4}I_{10} + \frac{2}{4}I_{11} + \frac{1}{4}I_{12} = \frac{1}{4}0 + \frac{2}{4}0 + \frac{1}{4}0 = 0$$

$$I(A_2) = \frac{1}{4}I_{20} + \frac{1}{4}I_{21} + \frac{2}{4}I_{22} = \frac{1}{4}0 + \frac{1}{4}0 + \frac{2}{4}1 = 0,5$$

$$I(A_4) = \frac{1}{4}I_{4Disco} + \frac{3}{4}I_{4Libro} = \frac{1}{4}0 + \frac{3}{4}(-\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}) = 0,23$$

Ejemplo de ID3



Traducción a reglas

- Cualquier árbol de decisión se puede convertir a reglas
- Regla: estructura del tipo Si-Entonces
- Ejemplo

Si Vivienda (zona)=1
Sitio de acceso=0
Entonces Bueno

- Algoritmo: por cada rama del árbol, las preguntas y sus valores estarán en la parte izquierda de las reglas y la etiqueta del nodo hoja correspondiente será la parte derecha (clasificación)

Ejemplo de traducción a reglas

Si Vivienda (zona)=0
Entonces Bueno

Si Vivienda (zona)=1 y Sitio de acceso=0
Entonces Bueno

Si Vivienda (zona)=1 y Sitio de acceso=1
Entonces Malo

Si Vivienda (zona)=1 y Sitio de acceso=2
Entonces Malo

Si Vivienda (zona)=2
Entonces Bueno

Dificultades del ID3

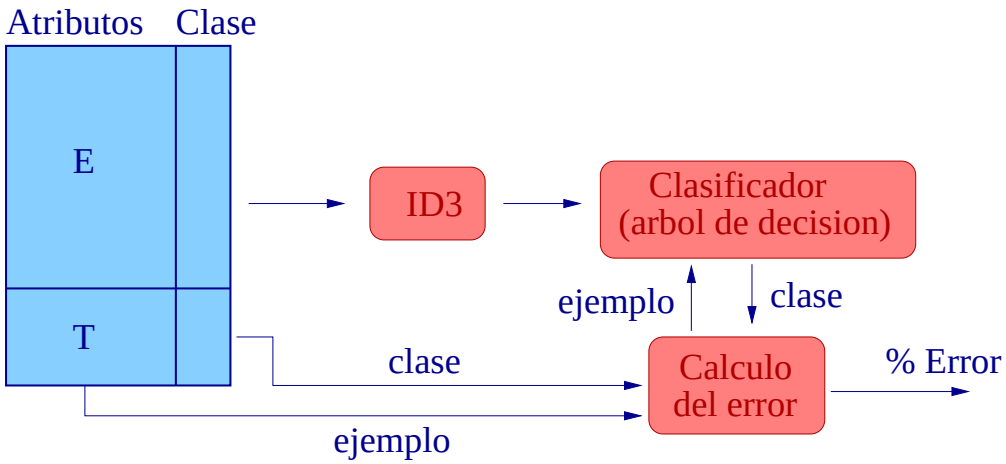
- ¿Cuándo se debe parar de subdividir el árbol? *sobreadecuación (overfitting), poda*
- ¿Qué se hace con valores contínuos de atributos? *peso*
- ¿Qué se hace con valores discretos con muchos valores? *día del cumpleaños*
- ¿Qué pasa si el coste de conocer el valor de un atributo no es constante? *presión sanguínea vs. biopsia*
- ¿Qué se hace cuando los ejemplos vienen incrementalmente? *ID4 e ID5*

Dificultades (II)

- ¿Qué se hace cuando los ejemplos están representados en lógica de primer orden? ILP
- ¿Qué ocurre cuando hay atributos con valores desconocidos?
  - ⋆ asignar el valor más probable
  - ⋆ asignar distribución de probabilidad
- ¿Qué ocurre cuando las clases son contínuas? $\mathrm{M5}$
- ¿Qué se hace cuando dos partes del árbol son iguales? $replicación$

Evaluación

- el conjunto de ejemplos se divide en dos partes: entrenamiento (E) y *test* (T)
- se aplica la técnica (p.e. el ID3) al conjunto de entrenamiento, generando un clasificador
- se calcula el número de errores (o aciertos) que el clasificador comete en el conjunto de *test*



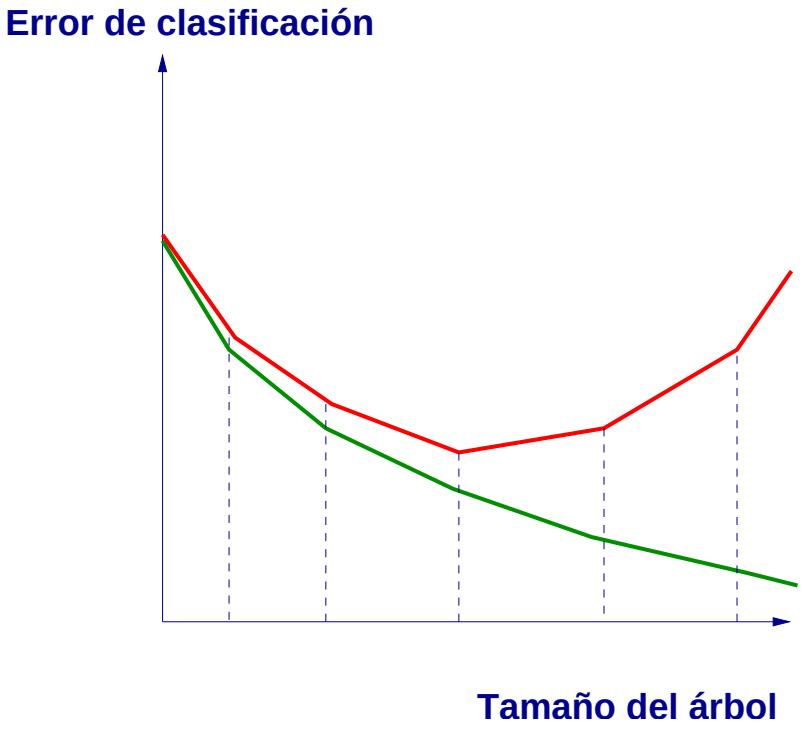
Validación cruzada

- Problema de la evaluación: sesgo de los conjuntos E y T seleccionados
- Solución: validación cruzada k-veces (k-*fold cross validation*)
  - ⋆ Se divide el conjunto de ejemplos en k partes iguales, E_i
  - ⋆ Se realiza lo siguiente k veces:
    - ∗ se entrena con $E - E_i$ (i=1..k)
    - ∗ se calcula el error con el E_i, e_i
  - ⋆ Se estima la tasa de error haciendo la media de los errores

$$r = \sum_{i=1}^{k} \frac{e_i}{k}$$

Sobreadecuación (*Overfitting*)

- Las hipótesis se refinan tanto que describen muy bien las instancias de aprendizaje, pero el error de clasificación crece en ejemplos externos



- Provocado por: ruido en los ejemplos, pocos ejemplos, o error en la representación
- Solución: poda

Pre-poda

- *Solución 1*: utilizar el χ^2 (Quinlan, 86)
  - ⋆ no se divide un nodo si se tiene poca confianza en él (no es significativa la diferencia de clases)
  - ⋆ ejemplo: cuando se tienen 40 ejemplos positivos y uno negativo
  - ⋆ es muy conservador y puede hacer que se pare el árbol antes de lo que conviene
- *Solución 2*: generar las curvas y parar cuando la curva del conjunto de test empieza a subir

Post-poda

- *Solución 3*: generar el árbol y analizar recursivamente y desde las hojas qué preguntas se pueden eliminar sin que afecten al error de clasificación con el conjunto de *test*
- *Solución 4*: generar las reglas equivalentes y eliminar condiciones/reglas si el error de clasificación es menor sin ellas (C4.5)

Poda de reglas. C4.5 (Quinlan)

Convertir el árbol de decisión a un conjunto de reglas $\mathcal{R}$
error $=$ error de clasificación con $\mathcal{R}$
Para cada regla $R_i \in \mathcal{R}$
    Para cada precondición p_j de $p(R_i)$
        nuevo-error $=$ error al eliminar p_j de $p(R_i)$
        Si nuevo-error $\leq$ error
        Entonces error $=$ nuevo-error
                    eliminar p_j de $p(R_i)$
    Si $p(R_i)$ está vacío
    Entonces eliminar R_i

Atributos con valores contínuos

- Se ordenan los valores del atributo, y se especifica la clase a la que pertenecen

peso	30	36	44	60	72	78
dodó	no	no	sí	sí	sí	no

- Hay dos puntos de corte, (36-44) y (72-78), en los que se pueden calcular los valores medio: 40 y 75
- Para crear el nodo de decisión
  - ⋆ Se pueden crear atributos dinámicamente

$$peso < 40$$

  - ⋆ Se puede hacer la distinción en el mismo nodo

$$peso < 40, 40 < peso < 75, 75 < peso$$

Atributos con muchos valores

- Por cada valor v del atributo A, se puede crear un atributo binario $(A = v)$
- Para mejorar en eficiencia, sólo se crean cuando son necesarios
- Dado que ID3 prefiere atributos con mayor número de valores, se les puede desfavorecer utilizando la medida Razón de ganancia ($GainRatio$, GR):

$$GR(A_i) = \frac{G(A_i)}{-\sum_{j=1}^{nv(A_i)} \frac{n_{ij}}{n} \log_2 \frac{n_{ij}}{n}}$$

- Problema: cuando n_{ij} tiende a n, el denominador se hace 0

Atributos con costes variables

- Se puede usar

$$\frac{Ganancia\ de\ información}{unidad\ de\ coste}$$

o

$$\frac{Ganancia\ de\ información}{unidad\ de\ coste^2}$$

- Normalmente, no se llega al árbol de decisión óptimo
- Dominios en los que es útil
  - ⋆ Medicina (Nuñez, 88)
  - ⋆ Robótica (Tan y Schlimmer, 90)

Aprendizaje vago. IBL

Aprendizaje Basado en Ejemplares (IBL)

- Modelización de algunos procesos de aprendizaje humano
- Espacio de hipótesis: conjunto de ejemplares
- Ejemplar es: instancia o abstracción de instancia
- Características de los ejemplares:
  - ⋆ Los conceptos se representan por conjuntos de ejemplares sin información sobre las condiciones necesarias y/o suficientes
  - ⋆ La representación es explícitamente disyuntiva
  - ⋆ Las propiedades de un concepto son función de las propiedades de los ejemplares
- Similitud con el Razonamiento Basado en Casos (CBR)

Método kNN

- Aprendizaje: guarda todas las instancias
- Clasificación: si se tiene x_i, se elige aquella clase más común entre los k ejemplos (vecinos) más cercanos
- Fórmula de similitud:

$$\arg\min_{x_j} d(x_i, x_j)^2 = \sum_{l=1}^{a} (x_{il} - x_{jl})^2$$

- Si los atributos son no numéricos, se hace:

$$(x_{il} - x_{jl}) = \begin{cases} 0 & \text{si } x_{il} = x_{jl} \\ 1 & \text{si } x_{il} \neq x_{jl} \end{cases}$$

Otros aspectos

- Si la clase es contínua: elige la media de las clases de los k vecinos x_i más cercanos

$$c = \frac{\sum_{i=1}^{k} c(x_i)}{k}$$

- Si los atributos tienen diferentes rangos: se normalizan

$$\frac{x_{ij} - m_j}{M_j - m_j}$$

- Se puede dar mayor preferencia a los vecinos más cercanos dentro de los k: se multiplica el voto de cada vecino por:

$$\frac{1}{d(x_i, x_j)^2} \text{ o por } \frac{1}{1 + d(x_i, x_j)^2}$$

Otros aspectos

- Para las clases contínuas, se multiplica cada voto por esa cantidad y se divide por la suma de esas distancias a los k vecinos más cercanos
- Se pueden favorecer unos atributos frente a otros

$$\arg\min_{x_j} d(x_i, x_j)^2 = \sum_{l=1}^{a} w_l \times (x_{il} - x_{jl})^2$$

Tipos de Modelos

- Modelo de Proximidad (Smith y Medin, 81)
  - ⋆ Almacena todas las instancias
  - ⋆ No realiza abstracción
  - ⋆ La clasificación calcula la similitud con todas las instancias.
  - ⋆ Una instancia pertenece a una clase si la instancia más similar almacenada pertenece a esa clase
- Modelo de Mejores-Ejemplos (Smith y Medin, 81)
  - ⋆ Asume que existe un prototipo para cada clase
  - ⋆ Un prototipo es un conjunto de ejemplares de la clase
  - ⋆ Los prototipos son los ejemplares que contienen más valores de los atributos en común que el resto
  - ⋆ Sólo almacena la instancia prototipo de cada clase
  - ⋆ Clasifica una instancia en la clase con el prototipo más similar

Tipos de Modelos

- Modelo de Selección-Ejemplos (Kibler y Aha, 87)
  - ⋆ Salvan sólo una parte de las intancias
  - ⋆ No asumen que existen los prototipos de cada clase
  - ⋆ El orden de presentación de las instancias es importante
  - ⋆ **Método de crecimiento:** almacena sólo las instancias que no se clasifican correctamente
  - ⋆ **Método de decrecimiento:** almacena todas las instancias inicialmente y borra, por turno, aquéllas que se clasifiquen correctamente
  - ⋆ Ventajas:
    - ∗ Menor espacio
    - ∗ Mejores clasificadores, al rechazar las instancias atípicas (posible ruido)
  - ⋆ Inconveniente: Mayor complejidad computacional

Aprendizaje vago

- Métodos simples
- Coste de *aprendizaje* muy bajo
- Coste de clasificación alto: hay que comprobar en qué se parece la instancia a clasificar con cada una de las instancias almacenadas
- Utilizan todos los atributos y ejemplos
- No generalizan (espacio de Voronoi)