

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Árboles de Decisión

Eduardo Morales, Hugo Jair Escalante

INAOE

# Contenido

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

1 Aprendizaje Basado en Similaridades (SBL)

2 Árboles de Decisión (TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos y manejo de ruido

Ruido y Sobreajuste

Árboles de Regresión y de Modelos

# SBL

Aprendizaje Basado en Similaridades (SBL)

Árboles de Decisión (TDIDT)

Algoritmo ID3

Cómo le hace

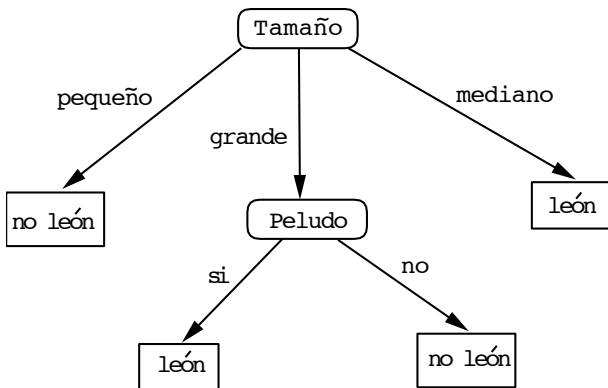
Atributos numéricos y manejo de ruido

Ruido y Sobreajuste

Árboles de Regresión y de Modelos

	Atributos			Clase
	Peludo?	Edad?	Tamaño?	
si	viejo	grande	león	
no	joven	grande	no león	
si	joven	mediano	león	
si	viejo	pequeño	no león	
si	joven	pequeño	no león	
si	joven	grande	león	
no	joven	pequeño	no león	
no	viejo	grande	no león	

# Árbol de Decisión



# Reglas de Clasificación

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

If Tamaño = mediano  
Then león

If Tamaño = pequeño  
Then no león

If Tamaño = grande  
and Peludo = si  
Then león

If Tamaño = grande  
and Peludo = no  
Then no león

# Inducción de Árboles de Decisión

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Desarrollados desde principios de los 60's: CLS (Hunt et al., 1966), ID3 (Quinlan, 1979), CART (Breiman et al., 1984), ACLS (Niblett et al., 1982), ASSISTANT (Cestnik et al., 1987), C4.5 (Quinlan, 1993), etc.
- Muchos convertidos en herramientas comerciales: RuleMaster (1984), Ex-Tran (1984), Expert-Ease (1983), y C5/See5 (2000).
- Existen en la mayoría de los ambientes de ML, KDD, etc.

# Inducción de Árboles de Decisión

- El aprendizaje de árboles de decisión es sencillo, fácil de implementar y poderoso.
- Un árbol recibe un objeto o situación descrita por un conjunto de atributos y regresa una decisión
- Por simplicidad consideraremos primero sólo simples funciones Booleanas (“verdadero/falso”)
- Cada nodo interno corresponde a una prueba en el valor de uno de los atributos y las ramas están etiquetadas con los posibles valores de la prueba.
- Cada hoja especifica el valor de la clase.

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Expresividad

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Están limitados a hablar de un solo objeto (son proposicionales)
- No pueden expresar pruebas sobre dos o más objetos diferentes, e.g.  $\exists r_2 \text{Cercano}(r_2, r) \wedge \text{Precio}(r_2, p_2) \wedge \text{Precio}(r, p) \wedge \text{MasBarato}(p_2, p)$
- Podríamos añadir un atributo Booleano que se llame: *RestMásBaratoCerca*, pero es intratable para todas las combinaciones de atributos



# Expresividad

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Por otro lado son completamente expresivos dentro de la clase de lenguajes proposicionales; cualquier función Booleana puede ser descrita por un árbol de decisión
- Trivialmente, podemos tomar cada fila como un camino en la construcción de un árbol
- Para muchas funciones, los árboles son relativamente pequeños, pero para otras funciones requieren un árbol exponencialmente grande (e.g., *paridad* o *mayoría*)

# Complejidad

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Para  $n$  atributos, hay  $2^n$  filas y podemos considerar la salida como una función definida por  $2^n$  bits
- Con esto hay  $2^{2^n}$  posibles funciones diferentes para  $n$  atributos (para 6 atributos, hay  $2 \times 10^{19}$ )
- Por lo que tenemos que usar algún algoritmo ingenioso para encontrar una hipótesis consistente en un espacio de búsqueda tan grande

# Inducción de árboles de decisión

- Un ejemplo es descrito por los valores de los atributos y el valor del predicado meta (clase)
- Si el predicado es verdadero, entonces el ejemplo es positivo, sino el ejemplo es negativo
- En caso de existir más clases, los ejemplos de una sola clase son positivos y el resto de los ejemplos son considerados negativos
- El conjunto de ejemplos (datos), se divide aleatoriamente en un subconjunto de entrenamiento (con el que se construye la hipótesis) y uno de prueba (con el que se prueba la hipótesis)

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Inducción de Árboles de Decisión

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

Más formalmente:

- 1 Junta una gran cantidad de ejemplos
- 2 Divídelos en dos conjuntos disjuntos: entrenamiento y prueba
- 3 Usa el algoritmo de aprendizaje para generar una hipótesis  $H$
- 4 Mide el porcentaje de clasificación correcta de  $H$  en el conjunto de prueba
- 5 Repite los pasos 1 - 4 para diferentes tamaños de conjuntos seleccionados aleatoriamente

# Inducción de Árboles de Decisión

- Encontrar un árbol puede ser trivial, pero no necesariamente es bueno para predecir casos no vistos
- El extraer un patrón significa el poder describir una gran cantidad de ejemplos en forma concisa
- Sigue un principio general en los algoritmos de inducción llamado: *Ockham's razor* (muchas veces escrito como Occam): dar preferencia a hipótesis más simples que sean consistentes con todas las observaciones.
- Encontrar el árbol más pequeño es intratable, por lo que se usan heurísticas para encontrar árboles pequeños

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Inducción de Árboles de Decisión

- Idea: Probar primero el atributo más “importante”
- Este particiona los ejemplos y cada subconjunto es un nuevo problema con menos ejemplos y un atributo menos
- Este proceso recursivo tiene 4 posibles resultados:
  - 1 Si existen ejemplos positivos y negativos, escoge el mejor atributo
  - 2 Si todos los ejemplos son positivos (o negativos), termina y regresa True (o False)
  - 3 No quedan ejemplos, regresa un *default* con base en la clasificación mayoritaria de su nodo padre
  - 4 No hay más atributos, pero seguimos con ejemplos positivos y negativos. Posible solución: Toma la clase mayoritaria

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

## Algoritmo ID3

```

if ejemplos = vacío then regresa default
else if todos los ejemplos tienen la misma clasificación
    then regresa la clasificación
else if atributos = vacío
then regresa VALOR-MAYORITARIO(ejemplos)
else Mejor ← ESCOGE-ATRIBUTO(atributos, ejemplos)
    Arbol ← nuevo árbol de decisión con Mejor como raíz
    para cada valor  $v_i$  de Mejor do
         $ejemplos_i$  ← {ejemplos con Mejor =  $v_i$ }
        Subárbol ← ARBOL-DECISION(ejemplosi,
            atributos - mejor,
            VALOR-MAYORITARIO(ejemplos))
        añade una rama a Arbol con etiqueta  $v_i$  y
        subárbol Subárbol
    end
return Arbol
  
```

## Tabla de Ejemplo

Ambiente	Temp.	Humedad	Viento	Clase
soleado	alta	alta	no	N
soleado	alta	alta	si	N
nublado	alta	alta	no	P
lluvioso	media	alta	no	P
lluvioso	baja	normal	no	P
lluvioso	baja	normal	si	N
nublado	baja	normal	si	P
soleado	media	alta	no	N
soleado	baja	normal	no	P
lluvioso	media	normal	no	P
soleado	media	normal	si	P
nublado	media	alta	si	P
nublado	alta	normal	no	P
lluvioso	media	alta	si	N

Aprendizaje Basado en Similaridades (SBL)

Árboles de Decisión (TDIDT)

Algoritmo ID3

Cómo le hace

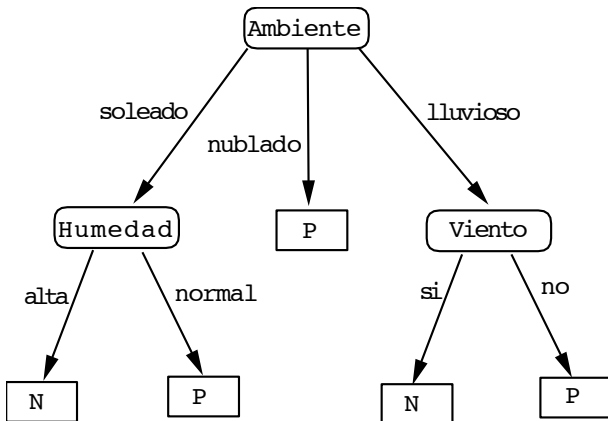
Atributos numéricos y manejo de ruido

Ruido y Sobreajuste

Árboles de Regresión y de Modelos



# Árbol de Salida



# Aplicaciones

Es la técnica que posiblemente más se ha usado en aplicaciones reales:

- GASOIL (1986): Diseño de sistemas de separación de hidrocarburos en plataformas petroleras de BP, 2,800 reglas, 1 año-hombre de tiempo de desarrollo, 0.1 de mantenimiento
- BMT (1990): Configuración de equipo de protección de incendios en edificios, > 30,000 reglas, 9 años hombre de desarrollo y 2 de mantenimiento
- Aprendiendo a volar (1992): Datos de pilotos haciendo plan de vuelo 30 veces. Cada acción es un ejemplo. Se usaron 90,000 ejemplos con 20 atributos y se observó *clean-up effect*

## ¿Cómo le hace?

- La medida en ESCOGE-ATRIBUTO debe ser máxima cuando el atributo discrimine perfectamente ejemplos positivos y negativos, y mínima cuando el atributo no sea relevante
- Una posibilidad es usar una medida basada en cantidad de información (basado en la teoría de Shannon y Weaver '49)
- La cantidad de información mide la (im)pureza en una colección arbitraria de ejemplos
- La cantidad de información (medida en bits) recibida respecto a la ocurrencia de un evento es inversamente proporcional a la probabilidad de ocurrencia de dicho evento

# Entropía

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

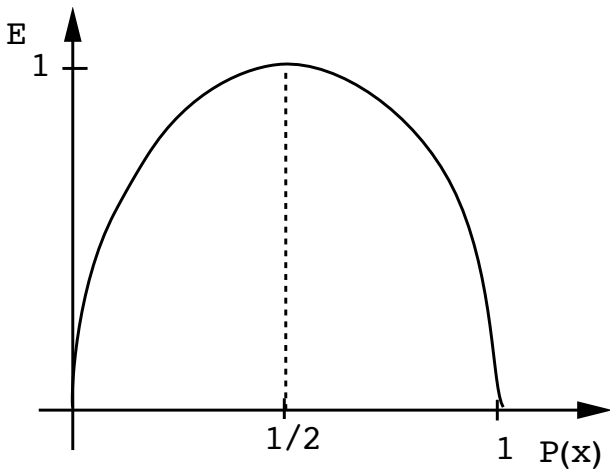
Árboles de  
Regresión y de  
Modelos

Si se tienen  $v_i$  posibles respuestas con probabilidades  $P(v_i)$ , el contenido de información es:

$$I(P(v_1), \dots, P(v_n)) = - \sum_{i=1}^n P(v_i) \log_2 P(v_i)$$

Nos representa el contenido promedio de información para los diferentes eventos.

# Función de Entropía



Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Entropía

- En el caso de los árboles queremos estimar las probabilidades de las respuestas, lo que se hace con la proporción de ejemplos positivos y negativos
- Si se tienen  $p$  ejemplos positivos y  $n$  ejemplos negativos, entonces:

$$I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) = -\frac{p}{p+n} \log_2 \frac{p}{p+n} - \frac{n}{p+n} \log_2 \frac{n}{p+n}$$

- Un solo atributo normalmente no nos proporciona toda esta información, pero podemos estimar cuanta, viendo cuanta información necesitamos después de utilizar ese atributo

# Entropía

- Cada atributo  $A$ , divide a los ejemplos en subconjuntos  $E_1, E_2, \dots, E_v$  de acuerdo a los  $v$  valores del atributo
- Cada subconjunto  $E_i$  tiene  $p_i$  ejemplos positivos y  $n_i$  ejemplos negativos, por lo que para cada rama necesitamos:  $I(\frac{p_i}{p_i+n_i}, \frac{n_i}{p_i+n_i})$  cantidad de información para responder a una pregunta
- Un ejemplo aleatorio tiene el valor  $i$ -ésimo del atributo  $A$  con probabilidad:  $\frac{p_i+n_i}{p+n}$ . Por lo que en promedio, después de probar el atributo  $A$ , necesitamos:

$$E(A) = \sum_{i=1}^v \frac{p_i + n_i}{p + n} I\left(\frac{p_i}{p_i + n_i}, \frac{n_i}{p_i + n_i}\right)$$

# Ganancia de Información

- La cantidad de información que ganamos al seleccionar un atributo está dada por:

$$Ganancia(A) = I\left(\frac{p}{p+n}, \frac{n}{p+n}\right) - E(A)$$

- La ganancia de  $A$  me dice el número de bits que ahorramos para responder a la pregunta de la clase de un ejemplo, dado que conocemos el valor del atributo  $A$ .
- Mide que tan bien un atributo separa a los ejemplos de entrenamiento de acuerdo a la clase
- La función de evaluación escoge el atributo de mayor ganancia



## Ejemplo

- Por ejemplo, si calculamos las ganancias para los atributos con los datos de la tabla de Golf (suponemos que:  $0 \times \log_2(0) = 0$ ):

$$I(9, 5) = -\frac{9}{14} \log_2\left(\frac{9}{14}\right) - \frac{5}{14} \log_2\left(\frac{5}{14}\right) = 0.941$$

- Para *Ambiente*:

soleado:  $p_1 = 2, n_1 = 3, I(p_1, n_1) = 0.971$

nublado:  $p_2 = 4, n_2 = 0, I(p_2, n_2) = 0$

lluvioso:  $p_3 = 3, n_3 = 2, I(p_3, n_3) = 0.971$

Entropía(Ambiente) =

$$\frac{5}{14} I(p_1, n_1) + \frac{4}{14} I(p_2, n_2) + \frac{5}{14} I(p_3, n_3) = 0.694$$

# Ejemplo

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Para *Humedad*:  
alta:  $p_1 = 3, n_1 = 4, I(p_1, n_1) = 0.985$   
normal:  $p_2 = 6, n_2 = 1, I(p_2, n_2) = 0.592$   
Entropía(Humedad) = 0.798
- Para *Viento*:  
no:  $p_1 = 6, n_1 = 2, I(p_1, n_1) = 0.811$   
sí:  $p_2 = 3, n_2 = 3, I(p_2, n_2) = 1.0$   
Entropía(Viento) = 0.892
- Para *Temperatura*, Entropía(Temperatura) = 0.9111

# Ejemplo

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Las ganancias son entonces:  
Ganancia(Ambiente) = 0.246 (MAX)  
Ganancia(Humedad) = 0.151  
Ganancia(Viento) = 0.048  
Ganancia(Temperatura) = 0.029
- Por lo que ID3 escoge el atributo *Ambiente* como nodo raíz y procede a realizar el mismo proceso con los ejemplos de cada rama

# Ejemplo

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Para *Ambiente* tenemos tres subconjuntos: soleado (2+, 3-), nublado (4+, 0-), lluvioso (3+, 2-). Para nublado, no tenemos que hacer nada, más que asignarle la clase  $P$
- Por ejemplo, para *soleado* haríamos el mismo proceso:  
Ganancia(Humedad) =  $0.97 - [(3/5)0 + (2/5)0] = 0.97$   
(MAX)  
Ganancia(Temperatura) =  $0.97 - [(2/5)0 + (2/5)1 + (1/5)0] = 0.570$   
Ganancia(Viento) =  $0.97 - [(2/5)1 + (3/5)0.918] = 0.019$

# Uso del Árbol de Decisión

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Con el árbol construido, podemos preguntar si está bien jugar el sábado en la mañana con ambiente soleado, temperatura alta, humedad alta y con viento, a lo cual el árbol me responde que no
- ID3 sigue una estrategia *hill-climbing*, sin *backtracking*
- Tiende a preferir construir árboles pequeños con atributos con ganancia de información alta cerca de la raíz

# Criterio de Selección

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- El criterio de selección tiende a favorecer atributos que tienen más valores
- Por ejemplo, un atributo con valores aleatorios. Con esto el algoritmo básico construye un árbol de un sólo nivel o *decision stump*
- Posible solución: Árbol binario. Desventaja: Árboles difíciles de entender + computacionalmente caro ( $2^n$  subconjuntos para  $n$  valores)

## Criterio de Selección

- Otra solución: Compensar dividiendo entre la información de la división (*split information*) que se define como:

$$SI(A) = - \sum_{i=1}^n P(A_i) \log_2 P(A_i)$$

- E.g., si un atributo binario divide los datos en dos subconjuntos de igual tamaño, el contenido de información de su división es 1. Mientras que para uno que los divide en 14 subconjuntos de tamaño 1, sería:  $14(-1/14 \log_2(1/14)) = -\log_2(1/14)$ .
- No siempre funciona y muchas veces se usa el atributo de la razón de ganancia de información máxima si su ganancia es al menos tan grande como el promedio de ganancia del resto de los atributos

# CART

- Se han propuesto un gran número de diferentes heurísticas
- Una medida muy utilizada es el índice Gini (CART):

$$Gini(t) = 1 - \sum_{j=1}^m (p(j | t))^2$$

donde  $p(j | t)$  es la frecuencia relativa de la clase  $j$  en  $t$ .

- Lo que se quiere es minimizar el índice al seleccionar un atributo. Para esto se calcula el índice en cada rama del atributo tomando en cuenta su proporción de ejemplos
- Si se divide en  $k$  ramas:

$$Gini_A = \sum_{i=1}^k \frac{n_i}{n} Gini(k)$$

donde  $n_i$  son los ejemplos de la rama y  $n$  los del nodo.



# CART

Aprendizaje Basado en Similaridades (SBL)

Árboles de Decisión (TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos y manejo de ruido

Ruido y Sobreajuste

Árboles de Regresión y de Modelos

- Por ejemplo, para la tabla de jugar golf, para el atributo *ambiente*:

$$\text{Gini}(\text{ambiente}=\text{soleado}) =$$

$$1 - \left(\frac{2}{5}\right)^2 - \left(\frac{3}{5}\right)^2 = 1 - 0.16 - 0.36 = 0.48$$

$$\text{Gini}(\text{ambiente}=\text{numblado}) = 1 - \left(\frac{4}{4}\right)^2 - \left(\frac{0}{4}\right)^2 = 0$$

$$\text{Gini}(\text{ambiente}=\text{lluvia}) = 1 - \left(\frac{3}{5}\right)^2 - \left(\frac{2}{5}\right)^2 = 0.48$$

- Por lo que para el atributo:  $\text{Gini}(\text{ambiente}) =$

$$\frac{5}{14}0.48 + \frac{4}{14}0 + \frac{5}{14}0.48 = 0.342$$

- Para temperatura:

$$\text{Gini}(\text{temperatura} = \text{alta}) = 1 - \left(\frac{2}{4}\right)^2 - \left(\frac{2}{4}\right)^2 = 0.5$$

$$\text{Gini}(\text{temperatura} = \text{baja}) = 1 - \left(\frac{3}{4}\right)^2 - \left(\frac{1}{4}\right)^2 = 0.375$$

$$\text{Gini}(\text{temperatura} = \text{media}) = 1 - \left(\frac{4}{6}\right)^2 - \left(\frac{2}{6}\right)^2 = 0.445$$

$$\text{Gini}(\text{temperatura}) = \frac{4}{14}0.5 + \frac{4}{14}0.375 + \frac{6}{14}0.445 = 0.439$$

# CART

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Para humedad:

$$\text{Gini}(\text{humedad}=\text{alta}) = 1 - \left(\frac{3}{7}\right)^2 - \left(\frac{4}{7}\right)^2 = 0.489$$

$$\text{Gini}(\text{humedad}=\text{baja}) = 1 - \left(\frac{6}{7}\right)^2 - \left(\frac{1}{7}\right)^2 = 0.244$$

$$\text{Gini}(\text{humedad}) = \frac{7}{14}0.489 + \frac{7}{14}0.244 = 0.367$$

- Para viento:

$$\text{Gini}(\text{viento}=\text{si}) = 1 - \left(\frac{6}{8}\right)^2 - \left(\frac{2}{8}\right)^2 = 0.375$$

$$\text{Gini}(\text{viento}=\text{no}) = 1 - \left(\frac{3}{6}\right)^2 - \left(\frac{3}{6}\right)^2 = 0.5$$

$$\text{Gini}(\text{viento}) = \frac{8}{14}0.375 + \frac{6}{14}0.5 = 0.428$$

# CART

- Los valores quedan entonces:  
Gini(ambiente) = 0.342 (MIN)  
Gini(temperatura) = 0.439  
Gini(humedad) = 0.367  
Gini (viento) = 0.428
- El algoritmo selecciona *ambiente* como el nodo del árbol
- Se continua el mismo proceso en cada rama (como en ID3)
- En este caso resulta el mismo árbol, pero en general no lo es

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Atributos Numéricos y Ruido

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

Hasta ahora hemos visto cómo funciona el algoritmo con atributos con valores discretos finitos y con datos sin ruido. Veremos ahora algunas extensiones para estos casos.

## Atributos Numéricos

- Normalmente se ordena el atributo, se identifican ejemplos adyacentes que tengan valor de clase diferente y se consideran como candidatos los puntos medios de división del valor del atributo
- A cada uno de estos se le calcula su ganancia de información
- Ejemplo de la tabla de Golf con temperatura:

64	65	68	69	70	71	72	75	80	81	83	85
P	N	P	P	P	N	N	P	N	P	P	N
						P	P				

- Existen 8 posibles lugares de corte que separan el valor de la clase y a cada uno se calcula su ganancia tomando el punto medio.
- Por ejemplo, para 71.5, Temperatura  $< 71.5$  tiene 4 P y 2 N, y Temperatura  $> 71.5$  tiene 5 P y 3 N.

# Atributos Numéricos

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Cada posible partición entra en competencia con el resto de los atributos
- Los atributos numéricos pueden aparecer varias veces en una rama de un árbol
- Para evitar ordenar ejemplos cada vez que se selecciona un atributo, se guarda (al principio) con cada subconjunto el orden de acuerdo a un atributo numérico

# Valores Faltantes

- Una forma de tratar valores faltantes es como si fueran otro posible valor del atributo (sólo funciona si el valor faltante tiene un significado especial)
- Ignorar los datos es demasiado drástico ya que algunas veces el valor del atributo puede no ser relevante para tomar una decisión
- Se han hecho diferentes propuestas, como llenar estos huecos con el valor más probable o con el valor más probable dada la clase

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Valores Faltantes

Lo que hace C4.5 es distribuir los objetos con valores desconocidos entre los demas. En donde se calcula la ganacia en información como si  $p_i$  fuera:

$$p_i + p_d \cdot \text{razon}_i$$

$$\text{razon}_i = \frac{p_i + n_i}{\sum_i (p_i + n_i)}$$

y  $p_d$  es la probabilidad de los datos desconocidos. Haciendo lo equivalente para  $n_i$ .



## ¿Cómo clasificar?

- La otra “cara de la moneda” es cómo clasificar con un objeto con atributos desconocidos.
- Idea: seguir todas las posibles ramas pero tomando en cuenta que algunas son más probables que otras (tienen más datos que la sorportan)

$T \cdot \text{razon}_i$

- Al final se puede calcular el nivel de “confianza” para una clasificación
- Si se sabe cierta información acerca del posible valor del atributo, se puede usar algún método probabilístico

# Costo de Clasificación

- Si existe un costo en la clasificación, y éste se conoce, entonces se puede incorporar a la probabilidad de la predicción de la clase (se multiplica)
- Normalmente se define una matriz de costo con la diagonal con 0's y los elementos fuera de la diagonal representan el costo de equivocarse en la clasificación
- Se multiplican las probabilidades de las clases predichas por el clasificador, por la columna correspondiente a la clase predicha en la matriz de costo y se selecciona la clase de menor costo esperado
- Variando las matrices de costo se puede variar la clasificación

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Costo de Clasificación

- También se ha estudiado como crear árboles cuando el costo de (error en la) clasificación es diferente (e.g., cancer).
- Una forma simple y general es generar datos de entrenamiento con diferente proporción en las clases
- Si nos interesa que clasifique bien una clase, entonces aumentamos la proporción de ejemplos de esa clase (duplicando instancias de la clase a predecir y/o reduciendo instancias de otras clases)
- Otros permiten incorporarle pesos a los ejemplos

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Ruido y *Overfitting*

- ID3 es útil en dominios no homogéneos (diferentes relaciones entre atributos en diferentes regiones del espacio de problemas) y alta dimensionalidad (muchos atributos)
- A pesar de que falte información relevante, se pueda construir un árbol con atributos irrelevantes
- Con muchas posibles hipótesis se pueden encontrar “regularidades con poco sentido”
- A este problema se le llama *overfitting* y afecta a todos los tipos de aprendizaje (i.e., no sólo a los árboles de decisión).

# Ruido y Overfitting

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

*Definición: dado un espacio de hipótesis  $H$ , una hipótesis  $h \in H$  se dice que sobreajusta los datos de entrenamiento si existe otra hipótesis  $h' \in H$ , tal que  $h$  tiene errores más pequeños que  $h'$  en los ejemplos de entrenamiento, pero  $h'$  tiene errores más pequeños que  $h$  en toda la distribución de ejemplos.*

# Ruido y *Overfitting*

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

Uno de los problemas a los que se enfrentan los sistemas de aprendizaje, y que provocan el sobreajuste, es cuando los ejemplos de entrenamiento contienen ruido:

- Valores de atributos erróneos, subjetivos
- Clasificación equivocada
- Valores desconocidos

# Ruido y *Overfitting*

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Con ruido, se pueden tener dos ejemplos con los mismos valores de atributos, pero clase diferente
- En presencia de ruido, el algoritmo básico (ID3) construye árboles de decisión que son más grandes de lo necesario, y no clasifican adecuadamente
- En el caso de árboles de decisión se tiene que decidir:
  - Cómo trabajar con atributos inadecuados
  - Cuándo al añadir atributos extra no mejora la predicción del árbol de decisión

# Ruido y *Overfitting*

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

En general, existen dos métodos para manejar ruido (basados en la condición de terminación):

- *Pruning* (o *pre-pruning*): Cambiar el criterio de paro del árbol de decisión para “podar” ramas.
- *Post-pruning*: “Podar” ramas una vez construido el árbol.



# Pruning

- En este caso, se tiene que decidir cuándo parar o dejar de construir el árbol a pesar de no tener hojas con ejemplos de una sola clase
- Se han propuesto técnicas usando: (i) Un umbral de ganancia de información, (ii) usando validación cruzada (si no mejora la clasificación con datos desconocidos parar), (iii) usando medidas basadas en el principio de longitud de descripción mínima (MDL), ...
- El problema principal en todos estos métodos es que se basan en información local

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Post-Pruning

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

Esta es la técnica más utilizada, y algunos de los métodos mencionados arriba se han utilizado para podar el árbol una vez ya construido.

Pasos:

- 1 Crece el árbol como antes (sólo verifica cuando se tienen ejemplos iguales y clases diferentes).
- 2 “Poda” el árbol.

El problema aquí es cuál árbol podado considerar y cómo estimar el error de clasificación de los árboles.

# Post-Pruning

- El *estimado de re-substitución* está dado por la proporción de ejemplos en el nodo mal clasificados si se toma la clase mayoritaria en el nodo
- Para hacer una estimación directa del error por re-substitución (poda), se hace con los mismos datos, se supone que siguen una distribución Bernoulli (por ser unos cuantos ( $< 100$ ), la cual se aproxima a una Normal con muchos datos), y considerando una estimación pesimista dentro de un nivel de confianza
- Para estimar errores se especifican niveles de confianza y las estadísticas (media y varianza) se obtienen directamente de los datos y los valores de tablas (e.g.,  $z$ )

## Ejemplo

- Supongamos que medimos el error de un clasificador con datos de prueba, y nos da 25%. Lo que queremos saber es qué tan confiable es esa estimación
- Si obtenemos eso con 100 datos o con 10,000 le creemos más a la estimación con los 10,000 datos
- En estadística, un conjunto de eventos independientes se conoce como un proceso Bernoulli (e.g., lanzar una moneda)
- Podemos pensar en el porcentaje de éxito, por lo que queremos saber es qué tanto se acerca ese 75% de éxito al verdadero porcentaje de éxito, expresado con intervalos de confianza

## Ejemplo

- Para 750 éxitos de 1,000 pruebas se tiene un 80% de confianza de que el verdadero porcentaje de éxito este entre 73.3% y 76.8%.
- Para 75 éxitos de 100 pruebas el mismo 80% de confianza tiene un intervalo de 70% y 81%.
- La media y la varianza de un proceso Bernoulli con porcentaje de éxito  $p$  son  $p$  y  $p(1 - p)$  respectivamente.
- Para  $N$  pruebas del proceso Bernoulli, el porcentaje de éxito es  $f = E/N$  ( $E =$  número de éxitos), y la varianza se reduce por un factor  $N$  a  $p(1 - p)/N$
- El valor de éxito se puede tomar como una variable aleatoria, con su media y su varianza.

# Ejemplo

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- La probabilidad de que una variable aleatoria con media cero esté en un intervalo de confianza de tamaño  $2z$  es  $Pr[-z \leq X \leq z] = c$ .
- Para una distribución normal los valores de  $c$  y de  $z$  están dados en tablas (ver tabla) que expresan la probabilidad de que  $X$  sea mayor a  $z$  ( $Pr[X \geq z]$ ).

# Niveles de confianza de una distribución normal

Aprendizaje Basado en Similaridades (SBL)

Árboles de Decisión (TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos y manejo de ruido

Ruido y Sobreajuste

Árboles de Regresión y de Modelos

$Pr[X \geq z]$	$z$
0.1%	3.09
0.5%	2.58
1%	2.33
5%	1.65
10%	1.28
20%	0.84
40%	0.25

## Ejemplo

- Las tablas nos dan sólo una mitad, pero al ser simétrica la distribución normal podemos considerar la mitad del intervalo que queremos y ese buscarlo en la tabla.
- Las tablas suponen que se tiene una media 0 y varianza de 1 ( $z$  nos mide las desviaciones estandar fuera de la media).
- Osea que para un 5% nos dice que existe un 5% que la variable  $X$  se encuentre a más de 1.65 desviaciones estandar arriba de la media, o un 10% que esté 1.65 desviaciones estandar (arriba o abajo) de la media.

$$Pr[-1.65 \leq X \leq 1.65] = 90\%$$



## Ejemplo

- Para cambiar a una media 0 y varianza 1 tenemos que restar la media  $p$  y dividirlo por la desviación estandar  $\sqrt{p(1-p)/N}$ . Esto nos da:

$$Pr[-z \leq \frac{f - p}{\sqrt{p(1-p)/N}} \leq z] = c$$

- Para esto dado un nivel de confianza  $c$  le restamos 1 y dividimos el resultado entre 2 y consultamos la tabla.
- Tenemos que encontrar una expresión para  $p$ . Después de cierta matemática nos queda:

$$p = \left( f + \frac{z^2}{2N} \pm z \sqrt{\frac{f}{N} - \frac{f^2}{N} + \frac{z^2}{4N^2}} \right) / \left( 1 + \frac{z^2}{N} \right)$$

# Ejemplo

- Esto nos da dos valores uno pesimista y otro optimista.
- La distribución normal es válida sólo para valores grandes de  $N$  (e.g.,  $N > 100$ ).
- Regresando a la poda de árboles, una forma es guardar datos y usarlos para estimar estos errores. Otra posibilidad es usar los mismos datos de entrenamiento para esta estimación, que es lo que hace C4.5.

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

## Ejemplo

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

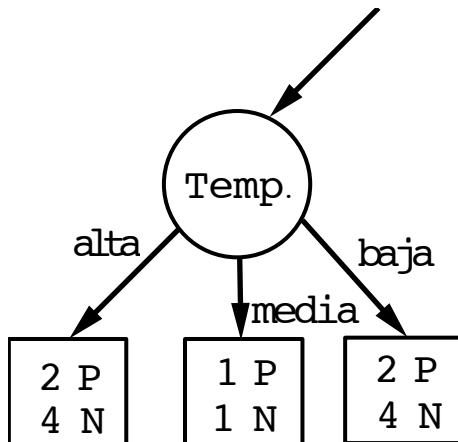
Árboles de  
Regresión y de  
Modelos

- El usar un estimado de éxito  $p$  o de error  $q$  es lo de menos,  $p + q = 1$ . Como los valores obtenidos son de los datos de entrenamiento se usa el valor pesimista que nos da:

$$p = \left( f + \frac{z^2}{2N} + z \sqrt{\frac{f}{N} - \frac{f^2}{N} + \frac{z^2}{4N^2}} \right) / \left( 1 + \frac{z^2}{N} \right)$$

- Para ver como funciona esto, supongamos que tenemos el subárbol de la figura

# Ejemplo de subárbol de decisión



Aprendizaje Basado en Similaridades (SBL)

Árboles de Decisión (TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos y manejo de ruido

Ruido y Sobreajuste

Árboles de Regresión y de Modelos

## Ejemplo

- Usamos  $c = 25\%$  ( $z = 0.69$ ). Para la rama izquierda tenemos 2 éxitos de 6 casos, lo cual nos da una  $f = 0.33$ . Poniendo esto en la fórmula nos da  $p = 0.47$  (aquí se ve la parte pesimista ya que en lugar un 33% nos da un 47%). Para la rama de en medio, tenemos 1 éxito de 2, lo cual nos da  $p = 0.72$ . La rama de la derecha es igual a la izquierda.
- La combinación de estos valores tomando en cuenta el porcentaje de ejemplos de cada uno, nos da 0.51
- Ahora para la clase mayoritaria del nodo tenemos  $f = 5/14$  lo cual nos da  $p = 0.46$ , que como es menor, entonces podemos esa rama

# Múltiples Clases

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

Al cambiar el criterio de paro, podemos tener hojas con ejemplos de diferentes clases. Posibles soluciones:

- Que las clases tomen valores fraccionarios ( $p/(p + n)$ )
- Tomar la clase mayoritaria (mejor si se quiere minimizar el error esperado)

En general con poco nivel de ruido, se comportan bien. No conviene quitarle ruido a los datos si se van a probar en ambientes ruidosos.

# Análisis de Complejidad

- La complejidad de contruir un árbol es:

$$O(mn \log n) + O(n(\log n)^2)$$

- Donde  $n$  es el número de datos y  $m$  el número de atributos
- El primer término se refiere a construir un árbol de decisión sin considerar podado. El segundo término se refiere cuando realizamos podado. Esto es independiente de que los atributos sean continuos o no.

# Algunas Extensiones

- En lugar de hacer una búsqueda tipo *hill climbing* usar *beam search*, o hacer búsqueda tipo *look-ahead*
- Generar varios árboles de decisión y luego combinar sus resultados.
  - En forma paralela con varios subconjuntos de los datos (*bagging*)
  - En forma secuencial considerando errores de clasificación y modificando los datos (*boosting*)
  - Usando varias muestras de los datos de entrenamiento e introduciendo aleatoriedad en la selección de atributos a considerar en cada nodo (*random forest*).

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos



# Árboles de Regresión y de Modelos

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

Cuando la clase a predecir es numérica existen dos variantes:

- *Regression trees*: Guardan el valor promedio de los valores en las hojas
- *Model trees*: Utilizan una regresión lineal para predecir los valores de las clases

Los dos tipos de árboles se construyen muy parecido a los árboles de decisión para construir un árbol inicial.

# Árboles de Regresión y de Modelos

Aprendizaje Basado en Similaridades (SBL)

Árboles de Decisión (TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos y manejo de ruido

Ruido y Sobreajuste

Árboles de Regresión y de Modelos

- En lugar de usar ganancia de información, seleccionar el atributo que minimiza la variación de la clase en cada rama
- Idea: Tratar la desviación estandar de la clase como medida de error y calcular la reducción esperada de error (*standard deviation reduction*) como resultado de probar cada atributo

$$SDR = destd(T) - \sum_i \frac{T_i}{T} \times destd(T_i)$$

- Donde  $T_1, T_2, \dots$  son los conjuntos que resultan de dividir al nodo de acuerdo al atributo seleccionado y *destd* es desviación estandar  
 $(destd = \sqrt{\sum_i^n (x(i) - \mu)^2 / (n - 1)}, \mu = \sum_i^n x(i) / n).$

# Árboles de Regresión y de Modelos

- El proceso termina cuando se tienen pocas instancias o la desviación estandar es una pequeña fracción (5%) de la original
- El modelo final es de la forma:  $w_0 + w_1 a_1 + \dots + w_k a_k$ , donde:  $a_i$ s son atributos y  $w_j$ s son pesos.
- Cuando se usa un árbol de modelos para predecir el valor, normalmente también se construyen modelos lineales en cada nodo interno del árbol para *suavizar* discontinuidades en las hojas, usando:

$$p' = \frac{np + kq}{n + k}$$

donde  $p'$  es el nuevo valor suavizado que se pasa al nodo de arriba,  $p$  es la predicción del nodo de abajo,  $q$  es el valor que se obtiene con el modelo asociado al nodo,  $n$  es el número de instancias asociadas al nodo de abajo y  $k$  es una constante.

# Árboles con combinaciones de atributos

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

En general, se han creado esquemas para mejorar el desempeño de los árboles en donde en cada nodo se combinan valores de varios atributos

- *Multivariate Decision Trees*: Combinan atributos en los nodos
- *Model Trees*: Combinan atributos en las hojas
- *Functional Trees*: Permiten combinaciones de atributos tanto en los nodos intermedios como en las hojas

# Árboles con combinaciones de atributos

Aprendizaje Basado en Similitudes (SBL)

Árboles de Decisión (TDIDT)

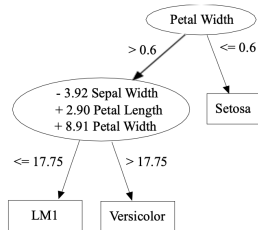
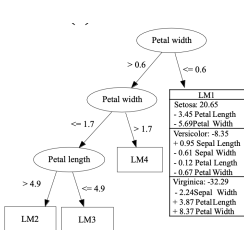
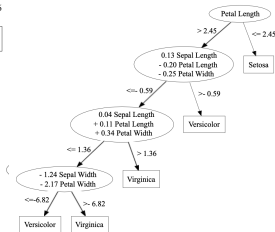
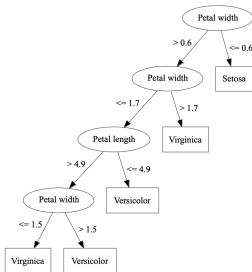
Algoritmo ID3

Cómo le hace

Atributos numéricos y manejo de ruido

Ruido y Sobreajuste

Árboles de Regresión y de Modelos



# Árboles con combinaciones de atributos

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- La combinación de atributos más común es una combinación lineal
- Las pruebas en los nodos son del tipo:  
$$\sum_{f \in F} w_f f(x) \leq v \text{ ó } \sum_{f \in F} w_f f(x) > v, \text{ donde } w_f \in \mathbb{R}$$
- En este caso, además de encontrar  $v$  deben de encontrar los valores de los pesos  $w_f$  para cada atributo

# Multivariate decision trees

- Para encontrar los pesos  $w$  se puede seguir un proceso analítico (considerando la distribución de las clases) o un proceso iterativo
- Para encontrar el punto de corte  $v$  se puede hacer igual con ambos procesos
- El nodo puede ser univariable (cómo cualquier árbol de decisión) o multivariable (con combinación lineal o no lineal)
- La selección de atributos en la combinación se puede hacer con un proceso de selección de atributos (e.g., *sequential feature selection*)

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

# Model Trees y Functional Trees

Aprendizaje  
Basado en  
Similitudes  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- El más popular es *Logistic Model Tree*, el cual usa regresión logística en cada hoja del árbol
- Para encontrar el modelo, generalmente se usa *LogitBoost*
- Los árboles funcionales permiten tener funciones tanto en los nodos intermedios como en las hojas
- Hacen combinaciones de *Model Trees* y *Multivariate Decision Trees*, pero hay pocos trabajos



# Árboles Óptimos

Aprendizaje  
Basado en  
Similaridades  
(SBL)

Árboles de  
Decisión  
(TDIDT)

Algoritmo ID3

Cómo le hace

Atributos numéricos  
y manejo de ruido

Ruido y Sobreajuste

Árboles de  
Regresión y de  
Modelos

- Avances recientes en técnicas de optimización entera + mejoras en hardware = mejoras en factores de miles de millones de veces más rápidas para encontrar soluciones
- Algoritmo basado en optimización entera mixta
- Resultados con árboles de profundidad 4 (máx.)