Aprendizaje Automatizado

Árboles de Clasificación

Árboles de Clasificación

- Estudiaremos un algoritmo para la creación del árbol.
- Selección de atributos comenzando en el nodo raíz.
- Proceso recursivo.

Árboles de Clasificación

- Entrada: Objetos caracterizables mediante propiedades.
- Salida:
  - En árboles de decisión: una decisión (sí o no).
  - En árboles de clasificación: una clase.
- Conjunto de reglas.

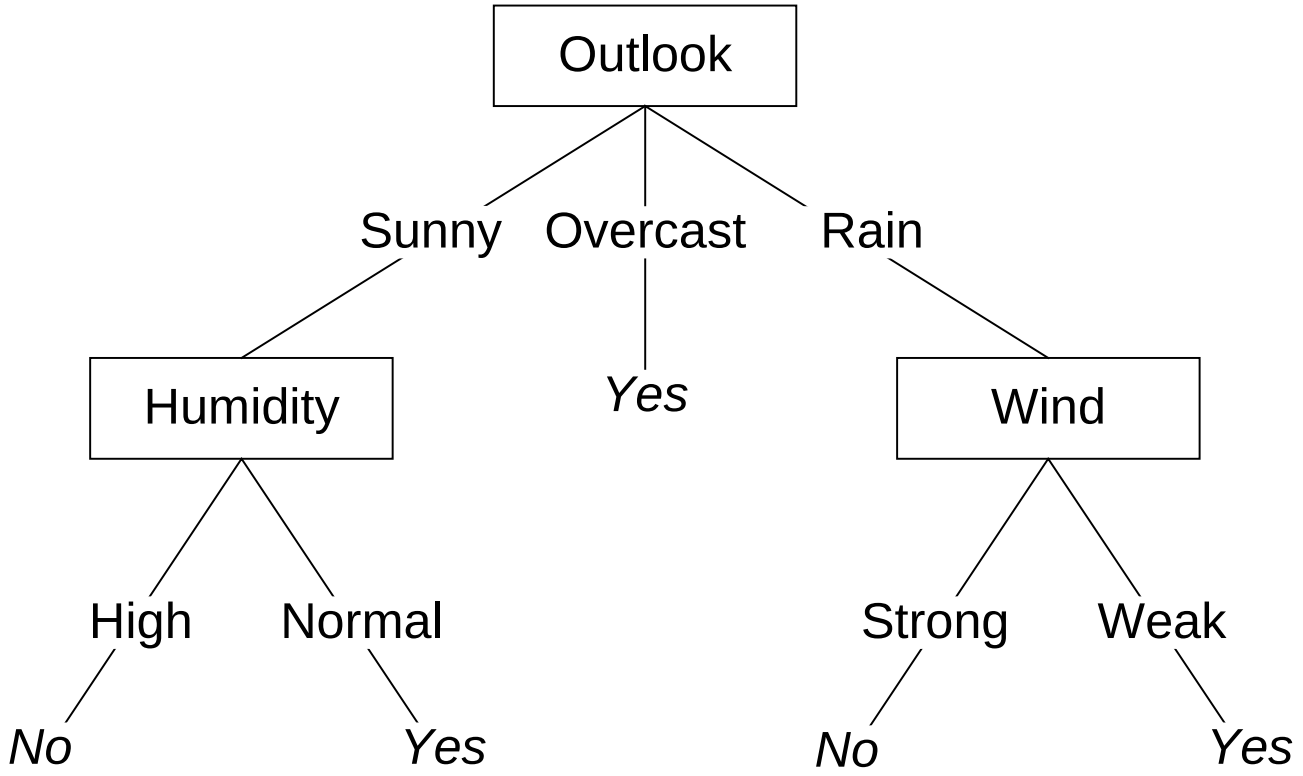
Árboles de Clasificación

- Se clasifican las instancias desde la raíz hacia las hojas, las cuales proveen la clasificación.
- Cada nodo especifica el test de algún atributo.
- Ejemplo: Si

(Outlook = Sunny, Humedity = High, Temperature = Hot, Wind = Strong)

Juego al tenis?

Play Tennis


Outlook
Sunny
Overcast
Rain
Humidity
Yes
Wind
High
Normal
Strong
Weak
No
Yes
No
Yes

Play Tennis

- Disyunción de conjunciones:

(Outlook = Sunny **And** Humidity = Normal)
Or (Outlook = Overcast)
Or (Outlook = Rain **And** Wind = Weak)

Play Tennis

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Problemas Apropiados

- Las instancias pueden ser representadas por pares (atributo, valor) .
- La función objetivo tiene valores discretos (o pueden ser discretizados).
- Pueden ser requeridas descripciones en forma de disjunción.
- Posiblemente existen errores en los datos de entrenamiento (robustos al ruido).
- Posiblemente falta información en algunos de los datos de entrenamiento.

Algoritmo básico para obtener un árbol de decisión (I)

- Búsqueda exhaustiva, en profundidad (de arriba hacia abajo), a través del espacio de posibles árboles de decisión (ID3 y C4.5).
- Raíz: el atributo que mejor clasifica los datos

  Cuál atributo es el mejor clasificador?

  $\Rightarrow$ respuesta basada en la **ganancia de información.**

Algoritmo básico para obtener un árbol de decisión (II)

- Hay ganancia de información cuando la división envía instancias con clases distintas a los distintos nodos.
- El atributo que permite obtener mayor ganancia de información es el seleccionado para dividir el nodo.

Algoritmo básico para obtener un árbol de decisión (III)

- El algoritmo ID3 se aplica a atributos discretos.
  - En cada nodo queda seleccionado un atributo y un valor (ej. temperatura = alta).
- El algoritmo C4.5 además se puede aplicar a atributos continuos.
  - En cada nodo queda seleccionado un atributo y un umbral para realizar la división (ej. temperatura > 26).

Algoritmo básico para obtener un árbol de decisión (IV)

- ID3 nunca produce árboles demasiado grandes.
- C4.5 sí, pues puede repetir atributos (temp < 26, temp > 24, temp < 25, etc).
- Un árbol demasiado grande puede producir sobreajuste (*overfitting*).
- Es necesario podar los árboles (*pruning*).

Algoritmos: ID3 (Interactive Dichotomizer Version 3)

- Entropía

  $Entropía(S) \equiv - p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$

  $p_{\oplus}$ = proporción de ejemplos positivos.

  $p_{\ominus}$ = proporción de ejemplos negativos.

  S: conjunto de datos actual.

  Por ejemplo, en el conjunto de datos Play Tennis

  $p_{\oplus} = 9/14$, $p_{\ominus} = 5/14$ y $E(S) = 0.940$

  En general: $Entropía(S) = - \sum_{i=1,c} p_i \log_2 p_i$

Algoritmos: ID3 (Interactive Dichotomizer Version 3)

- Por ejemplo:

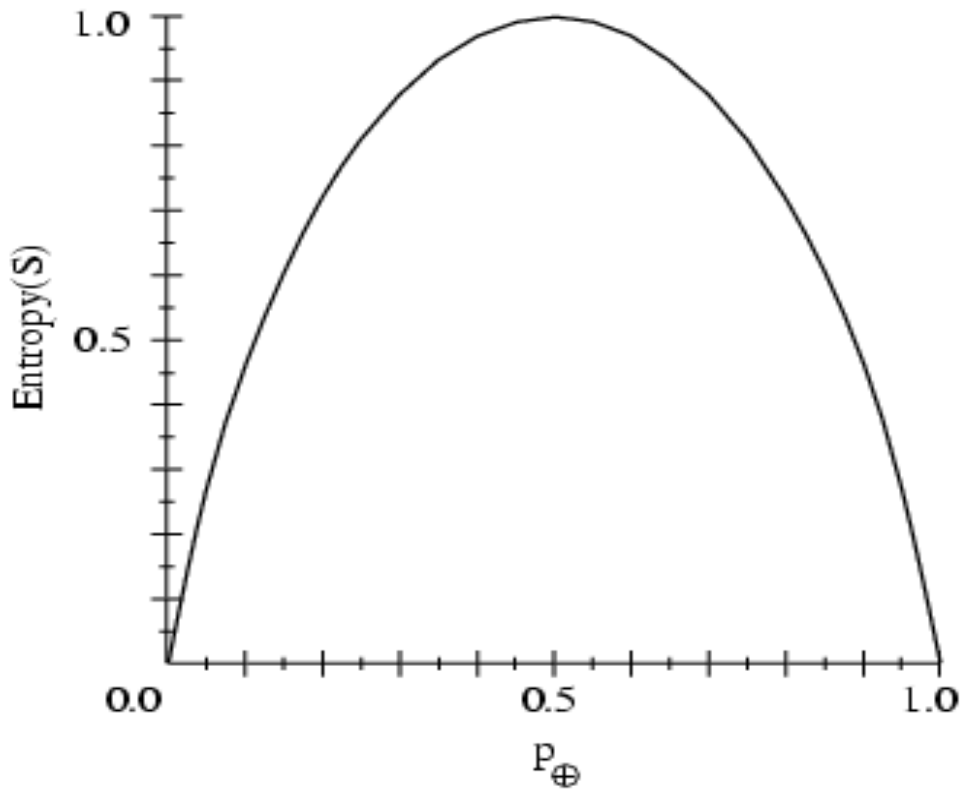
Si S_1 es el subconjunto de S en el cual

Humedity = High

Entonces:

- $p_{\oplus} = 3/7$
- $p_{\ominus} = 4/7$
- Entropía(S_1) = $-3/7 \log_2 3/7 - 4/7 \log_2 4/7 = 0.985$

Entropía y proporción de positivos



Ganancia de información

- Mide la reducción esperada de entropía sabiendo el valor del atributo A

$$Gain(S,A) \equiv$$

$$Entropía(S) - \sum_{v \in Valores(A)} (|S_v|/|S|)Entropía(S_v)$$

Valores(A): Conjunto de posibles valores del atributo A

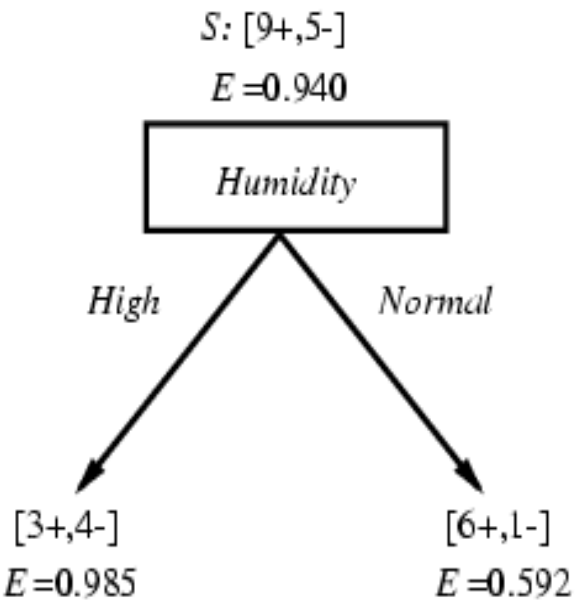
S_v: Subconjunto de S en el cual el atributo A tiene el valor *v*

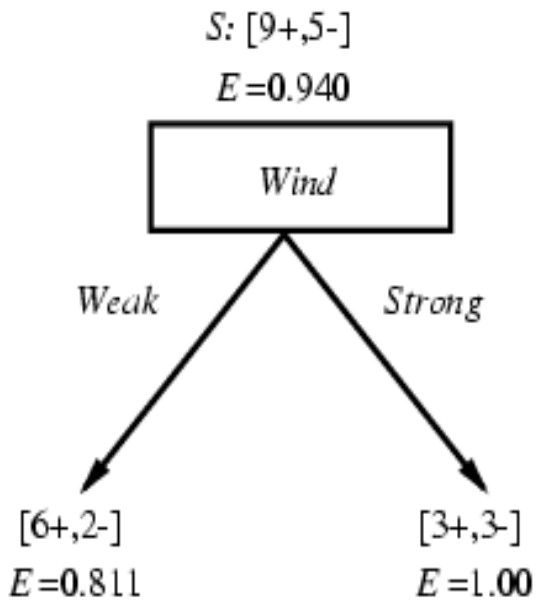
Ej: Gain(S, Humedad) = 0.940 - (7/14) 0.985 - (7/14) 0.592

proporción de humedad alta

proporción de humedad normal

Play Tennis





Gain (S, Humidity)

= .940 - (7/14).985 - (7/14).592
= .151

Gain (S, Wind)

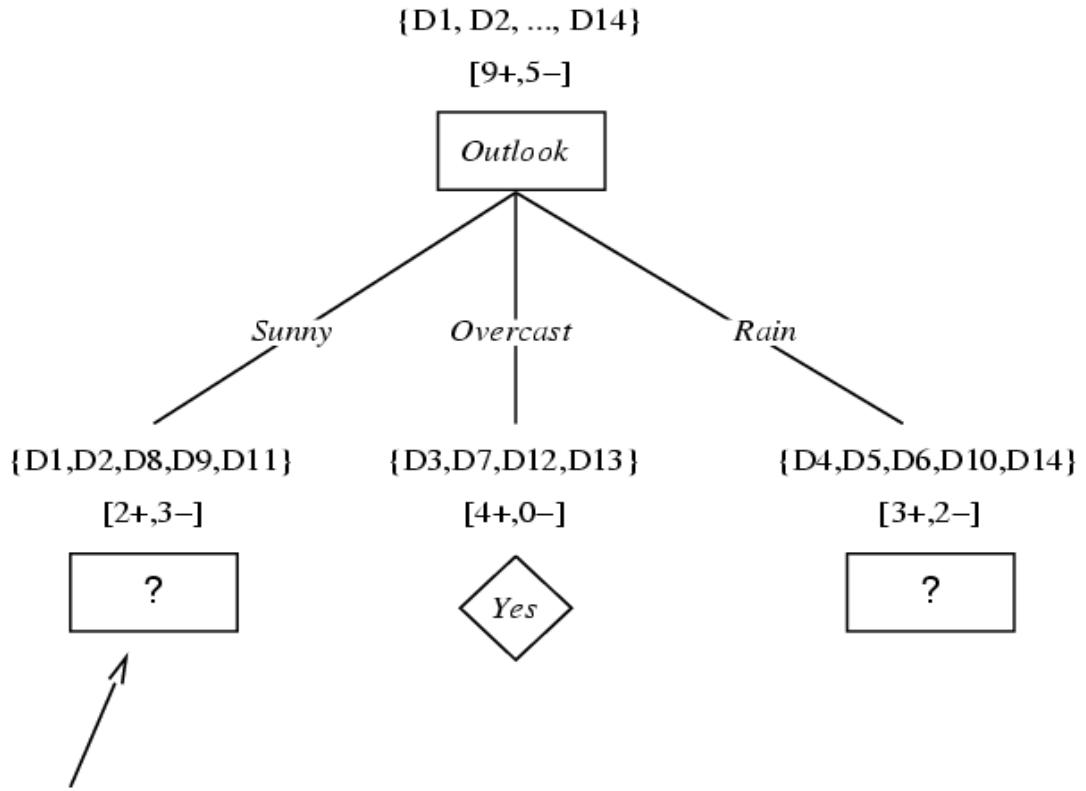
= .940 - (8/14).811 - (6/14)1.0
= .048

Play Tennis

- Gain(*S*,Outlook) = 0.246
- Gain(S,Humidity) = 0.151
- Gain(S,Wind) = 0.048
- Gain(S,Temperature) = 0.029

$\Rightarrow$ Outlook es el atributo del nodo raíz.

Play Tennis



Which attribute should be tested here?

S_{sunny} = {D1,D2,D8,D9,D11}

$Gain\ (S_{sunny}, Humidity) = .970 - (3/5)\,0.0 - (2/5)\,0.0 = .970$

$Gain\ (S_{sunny}, Temperature) = .970 - (2/5)\,0.0 - (2/5)\,1.0 - (1/5)\,0.0 = .570$

$Gain\ (S_{sunny}, Wind) = .970 - (2/5)\,1.0 - (3/5)\,.918 = .019$

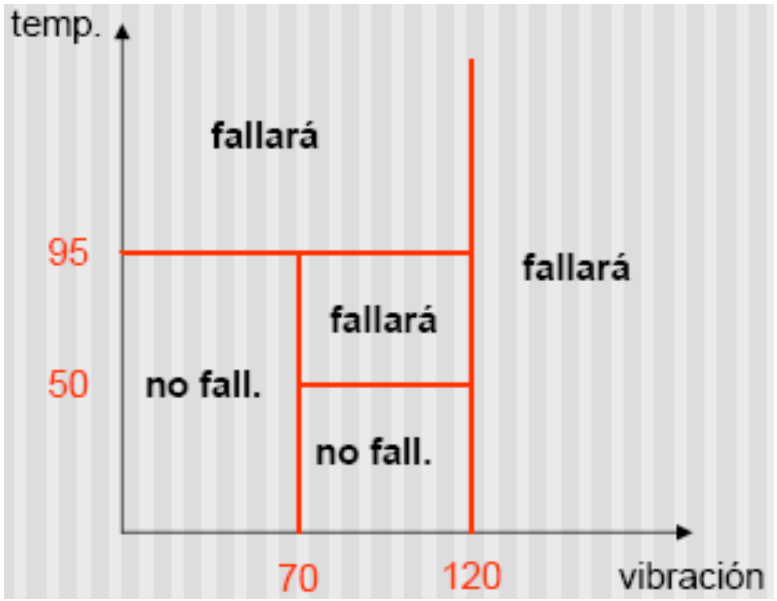
Sobreentrenamiento

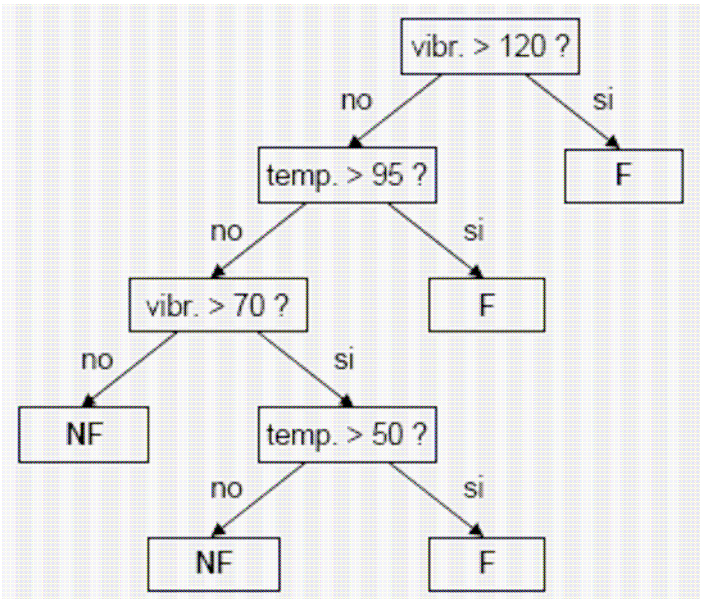
- Se debe evitar el sobreentrenamiento
  - Parar de crecer el árbol temprano.
  - Postprocesamiento del árbol (poda)

  Cómo?
  - Usar un conjunto de ejemplos de validación
  - Usar estadísticas

Árboles de Decisión - Resumen (I)

- Capacidad de representación:
  - No muy elevada, las superficies de decisión son siempre perpendiculares a los ejes:





Árboles de Decisión - Resumen (II)

- Legibilidad: muy alta. Uno de los mejores modelos en este sentido.
- Tiempo de cómputo on-line: muy rápido. Clasificar un nuevo ejemplo es recorrer el árbol hasta alcanzar un nodo hoja.
- Tiempo de cómputo off-line: rápido. Los algoritmos son simples.

Árboles de Decisión - Resumen (III)

- Parámetros a ajustar: nivel de confianza para la poda (el valor por defecto 25% da buenos resultados).
- Robustez ante instancias de entrenamiento ruidosas: robusto.
- Sobreentrenamiento o sobreajuste: No se produce siempre que se realice una poda.

Matlab - Statistics Toolbox (I)

- La clase `@classregtree` está diseñada para manipular árboles de regresión y árboles de decisión (CART).
- Ejemplo:

```
>> load fisheriris;
>> t = classregtree(datos, especies,
  'names', {'SL' 'SW' 'PL' 'PW'})
```

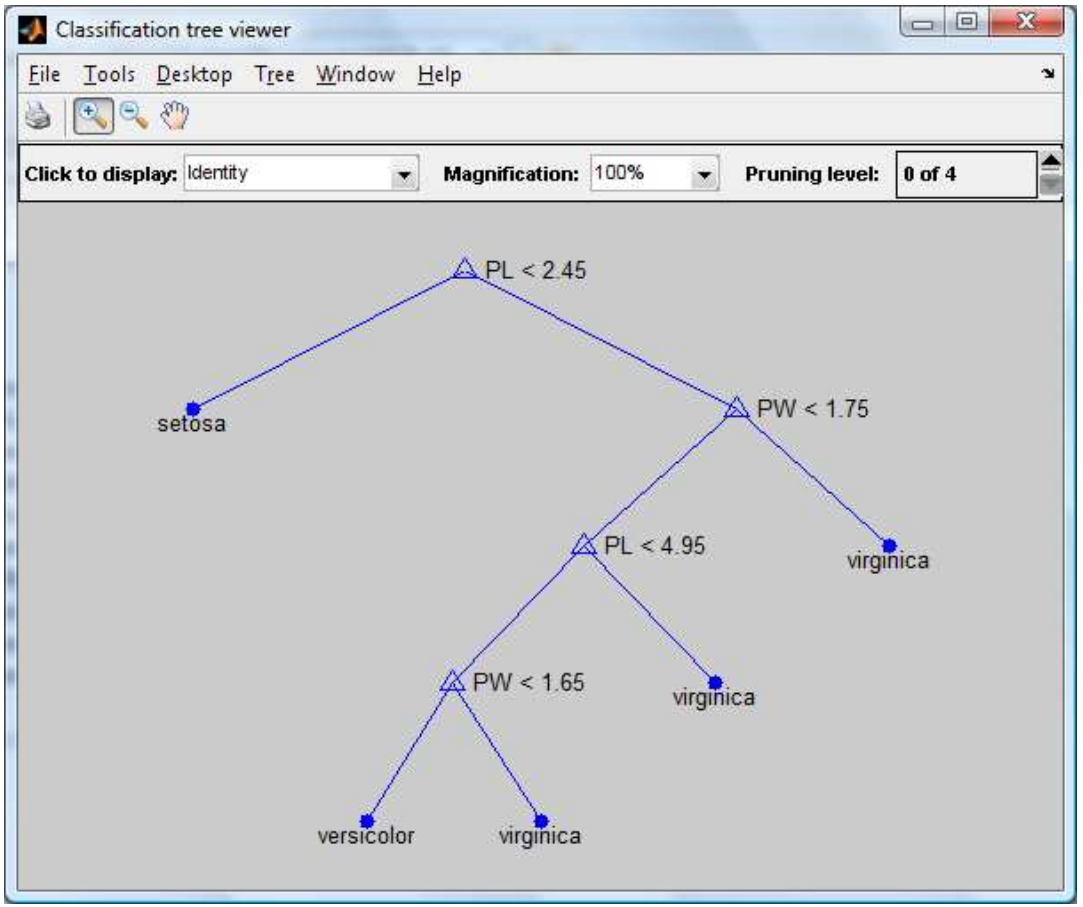
`t = classregtree(X,y)` crea un árbol de decisión `t` para una respuesta predicha `y` en función de los predictores en las columnas de `X`.

Matlab - Statistics Toolbox (II)

```
t =
Decision tree for classification
1  if PL<2.45 then node 2 else node 3
2  class = setosa
3  if PW<1.75 then node 4 else node 5
4  if PL<4.95 then node 6 else node 7
5  class = virginica
6  if PW<1.65 then node 8 else node 9
7  class = virginica
8  class = versicolor
9  class = virginica
```

Matlab - Statistics Toolbox (III)

```
>> view(t)
```



Matlab - Statistics Toolbox (IV)

- `prediccion = t([NaN NaN 4.8 1.6])`

  `prediccion = 'versicolor'`
- `var6 = cutvar(t,6) % ¿Qué variable determina la ramificación?`

  `var6 = 'PW'`
- `type6 = cuttype(t,6) % ¿Qué tipo de ramificación es?`

  `type6 = 'continuous'`

Matlab - Statistics Toolbox (V)

- `t = classregtree(X,y,param1,val1,param2,val2)`
- 'method' — Puede ser 'classification' (por defecto si y es texto o una variable categorica) o 'regression' (por defecto si y es numérica).
- 'names' — Un arreglo tipo `cell` de nombres para los atributos, en el orden en el cual aparecen en `X` .

Matlab - Statistics Toolbox (VI)

- Clasificar datos:

```
resultado = eval(t,meas);
```

- Computar la proporción de clasificados correctamente:

```
pct = mean(strcmp(sfit,species))
pct = 0.9800
```

- Podar el árbol:

```
t2 = prune(t, 'level', 1)
```

Bibliografía

- Machine Learning - Tom Mitchell – McGrawHill
- Statictics Toolbox User's Guide (http://www.mathworks.com/access/helpdesk/help/pdf_doc/stats/stats.pdf).
- Curso de doctorado "Aprendizaje Automatizado y Data Mining" Grupo de Ingeniería de Sistemas y Automática (Universidad Miguel Hernández) http://isa.umh.es/asignaturas/aprendizaje/index.html