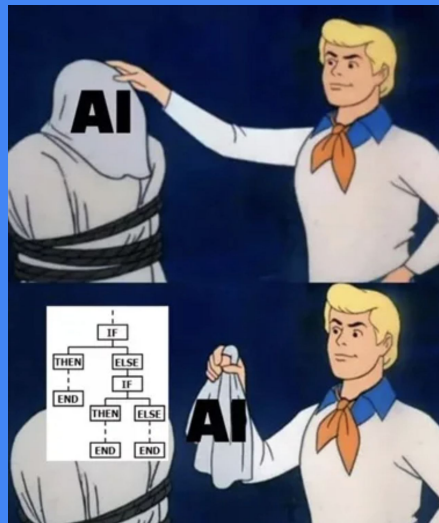


Aprendizaje Supervisado II

Facundo Carrillo
fcarrillo@udesa.edu.ar

Taller II

Hoy: Taller II



Temas:

- Clases de python
- Árboles de decisión
- Funciones de selección de atributos
- Sklearn

¿Dónde estamos parados?

- Ya sabemos que es un clasificador
- ¿Por qué otro?

Bibliografía:

- Mitchell Machine Learning: [website libro](#) , [pdf](#)
- Scikit-learn: <https://scikit-learn.org/>

Antes... Clases de Python

Python venía con diversos tipos de datos:

- Números enteros y decimales
- Caracteres y Strings
- Diccionarios
- Listas
- ...

¿Cómo construimos nuestros propios tipos de datos?

clases

¿Cómo son las clases?

- Tienen un nombre
- Tienen atributos
- Tienen métodos

Antes... Clases de Python

Por ejemplo, queremos modelar un alumno, con los atributos nombre, apellido y legajo

```
class Alumno():
    def __init__(self):
        # Puedo definir aca los atributos
        self.nombre = '<aun_no_definie_el_nombre>'
        self.apellido = '<aun_no_definie_el_apellido>'
        self.legajo = -1

# Creo un alumno
a1 = Alumno()

# Imprimo el nombre
print(a1.nombre) # imprime por pantalla: <aun_no_definie_el_nombre>

# Cambio el atributo nombre
a1.nombre = 'María'

print(a1.nombre) # imprime por pantalla: María
```

Antes... Clases de Python

Podemos definir un método para que imprima por pantalla nombre y apellido

```
class Alumno():
    def __init__(self):
        # Puedo definir aca los atributos
        self.nombre = '<aun_no_definie_el_nombre>'
        self.apellido = '<aun_no_definie_el_apellido>'
        self.legajo = -1

    def imprimir(self):
        print(self.nombre+' '+self.apellido)

# Creo un alumno
a1 = Alumno()
a1.nombre = 'Juan'
a1.apellido = 'Perez'

# Uso el método
a1.imprimir()

# imprime por pantalla: Juan Perez
```

Antes... Clases de Python

Supongamos que siempre que creamos un alumno queremos inicializar el nombre y apellido. Para eso modificamos el método especial `__init__` y le agregamos dos parametros

```
class Alumno():
    def __init__(self, el_nombre, el_apellido):
        self.nombre = el_nombre
        self.apellido = el_apellido
        self.legajo = -1

    def imprimir(self):
        print(self.nombre+' '+self.apellido)

# Creo un alumno pasándole ahora dos valores
a1 = Alumno('Pedro', 'Gonzalez')

# Uso el método
a1.imprimir()
```

Antes... Clases de Python

Le agregamos un campo para guardar numeros de telefono (0 o más)

```
class AlumnoConTel():  
    def __init__(self,el_nombre,el_apellido):  
        self.nombre = el_nombre  
        self.apellido = el_apellido  
        self.legajo = -1  
        self.telefonos = list()  
  
    def imprimir(self):  
        print(self.nombre+ ' '+self.apellido)  
  
    def agregar_telefono(self,numero):  
        self.telefonos.append(numero)  
  
    def tiene_telefono(self):  
        return ( len(self.telefonos)>0 )  
  
    def imprimir_telefonos(self):  
        print(self.telefonos)
```

```
# Creo el alumno  
a2 = AlumnoConTel('Lucía' , 'Fernandez')  
  
# Si tiene telefono los imprimo  
if not a2.tiene_telefono() :  
    print('El alumno no tiene teléfonos')  
  
# Agrego un numero de telefono  
a2.agregar_telefono('+5491145433232')  
  
a2.imprimir_telefonos() # muestra ['+5491145433232']
```

Árboles de Decisión

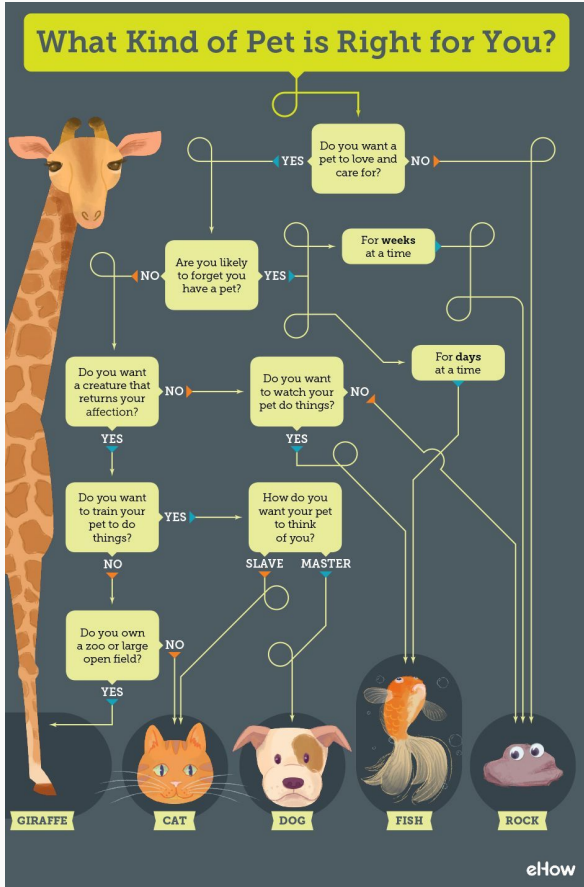
Clasificadores:

- Modelos que aproximan funciones que van de ATRIBUTOS -> CLASES
- Las funciones que queremos aproximar no son accesibles directamente pero sí con datos

Árboles de decisión:

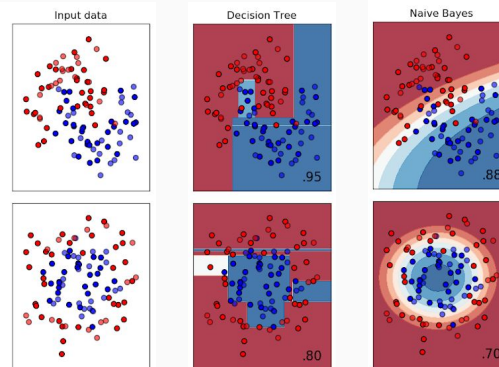
- Modelo para aproximar funciones (fitteando datos)
- Bueno para relaciones no lineales
- ¡Los modelos entrenados son de fácil interpretación!
- Muchos sabores:
 - máxima altura y poda
 - función de elección de atributo
 - Políticas de parada

Árboles de Decisión



Bibliografía:

- Mitchell, T.M. Machine Learning (McGraw-Hill, 1997)
- [Scikit-learn Decision Tree Classifier](#)
- Quinlan, J. Ross. "Induction of decision trees." *Machine learning* 1.1 (1986): 81-106.
- Quinlan, J. Ross. "Simplifying decision trees." *International Journal of Human-Computer Studies* 51.2 (1999): 497-510.
- Breiman, Leo, et al. "Classification and regression trees. Belmont, CA: Wadsworth." *International Group* 432 (1984): 151-166.



Árboles de Decisión

Ejemplo: El tenis según el clima de nuevo (ejemplo canónico del libro de Mitchell)

Medimos el estado del tiempo con los siguientes atributos:

- Cielo: *[Sol, Nublado, Lluvia]*
- Temperatura: *[Calor, Templado, Frío]*
- Humedad: *[Alta, Normal]*
- Viento: *[Fuerte, Débil]*

¿Juega al tenis?

< Cielo , Temperatura , Humedad , Viento > --- > [Si, No]

Tenemos datos de entrenamiento anotados, es decir, la agenda de alguien, si jugó o no y los datos del clima

Árboles de Decisión

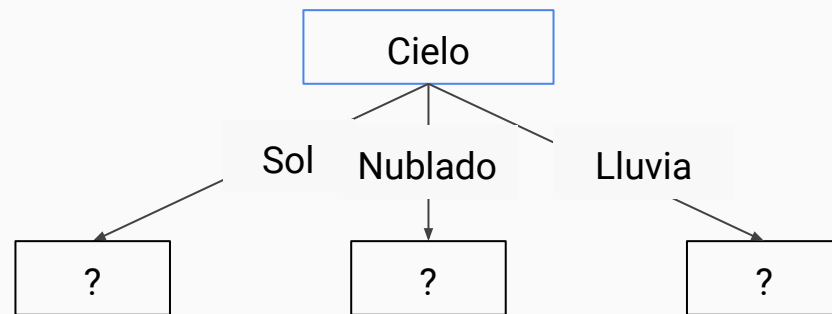
Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Nublado	Calor	Alta	Débil	Sí
Lluvia	Templado	Alta	Débil	Sí
Lluvia	Frío	Normal	Débil	Sí
Lluvia	Frío	Normal	Fuerte	No
Nublado	Frío	Normal	Fuerte	Sí
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Lluvia	Templado	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí
Nublado	Templado	Alta	Fuerte	Sí
Nublado	Calor	Normal	Débil	Sí
Lluvia	Templado	Alta	Fuerte	No

Armemos un árbol con reglas poco claras que voy a ir inventando.

Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Nublado	Calor	Alta	Débil	Sí
Lluvia	Templado	Alta	Débil	Sí
Lluvia	Frío	Normal	Débil	Sí
Lluvia	Frío	Normal	Fuerte	No
Nublado	Frío	Normal	Fuerte	Sí
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Lluvia	Templado	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí
Nublado	Templado	Alta	Fuerte	Sí
Nublado	Calor	Normal	Débil	Sí
Lluvia	Templado	Alta	Fuerte	No

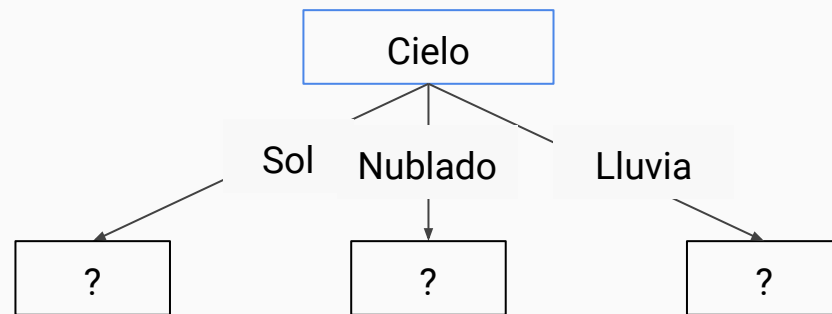
Entrenemos el árbol, empecemos con el primer atributo: **Cielo**



Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Nublado	Calor	Alta	Débil	Si
Lluvia	Templado	Alta	Débil	Si
Lluvia	Frío	Normal	Débil	Si
Lluvia	Frío	Normal	Fuerte	No
Nublado	Frío	Normal	Fuerte	Si
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Si
Lluvia	Templado	Normal	Débil	Si
Sol	Templado	Normal	Fuerte	Si
Nublado	Templado	Alta	Fuerte	Si
Nublado	Calor	Normal	Débil	Si
Lluvia	Templado	Alta	Fuerte	No

Entrenemos el árbol, empecemos con el primer atributo: **Cielo**

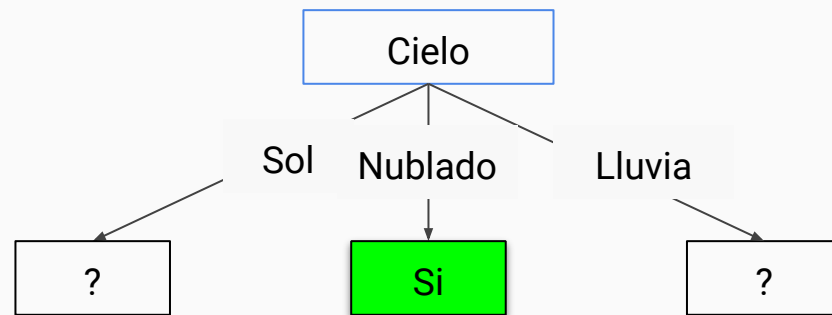


Todos los que [Cielo==Nublado] tienen clase **Si**

Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Nublado	Calor	Alta	Débil	Si
Lluvia	Templado	Alta	Débil	Si
Lluvia	Frío	Normal	Débil	Si
Lluvia	Frío	Normal	Fuerte	No
Nublado	Frío	Normal	Fuerte	Si
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Si
Lluvia	Templado	Normal	Débil	Si
Sol	Templado	Normal	Fuerte	Si
Nublado	Templado	Alta	Fuerte	Si
Nublado	Calor	Normal	Débil	Si
Lluvia	Templado	Alta	Fuerte	No

Entrenemos el árbol, empecemos con el primer atributo: **Cielo**

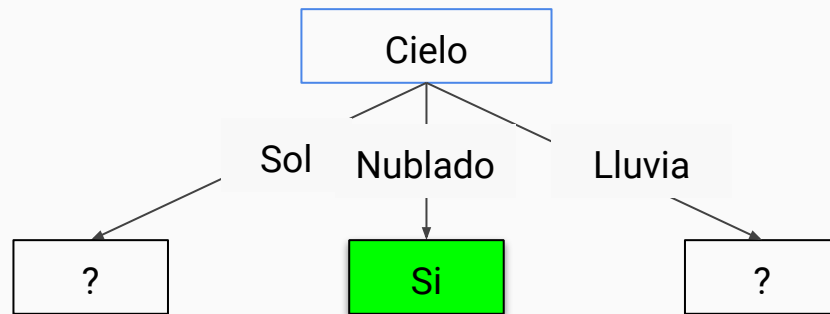


Todos los que [Cielo==Nublado] tienen clase **Si**

Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Lluvia	Templado	Alta	Débil	Sí
Lluvia	Frío	Normal	Débil	Sí
Lluvia	Frío	Normal	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Lluvia	Templado	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí
Lluvia	Templado	Alta	Fuerte	No

Entrenemos el árbol, empecemos con el primer atributo: **Cielo**

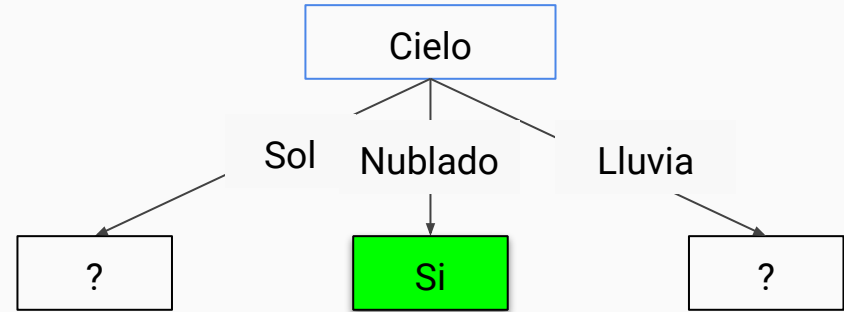


Todos los que [Cielo==Nublado] tienen clase **Si**

Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Lluvia	Templado	Alta	Débil	Sí
Lluvia	Frío	Normal	Débil	Sí
Lluvia	Frío	Normal	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Lluvia	Templado	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí
Lluvia	Templado	Alta	Fuerte	No

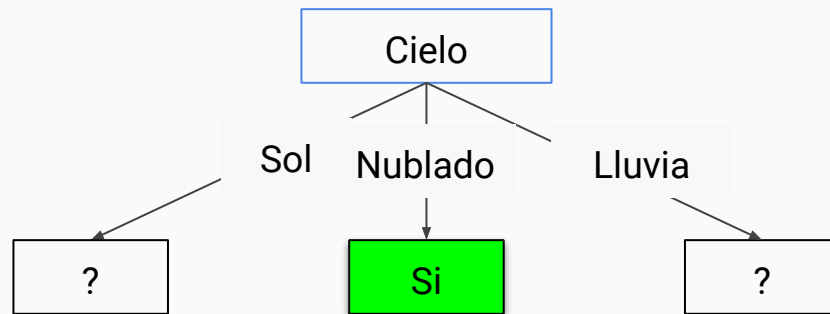
Ahora vamos por la rama **Lluvia**



Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Lluvia	Templado	Alta	Débil	Sí
Lluvia	Frío	Normal	Débil	Sí
Lluvia	Frío	Normal	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Lluvia	Templado	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí
Lluvia	Templado	Alta	Fuerte	No

Ahora vamos por la rama **Lluvia**



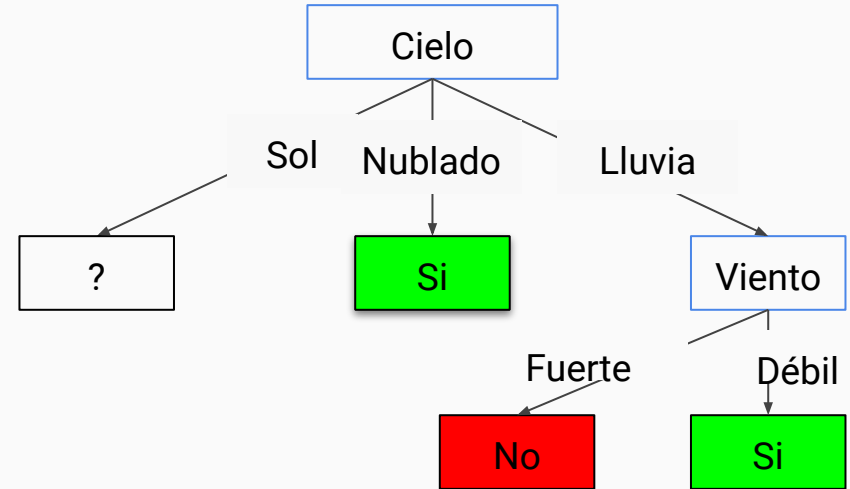
El atributo **Viento** separa bien el target!

[Viento==Débil] -> **Si**
[Viento==Fuerte] -> **No**

Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Lluvia	Templado	Alta	Débil	Sí
Lluvia	Frío	Normal	Débil	Sí
Lluvia	Frío	Normal	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Lluvia	Templado	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí
Lluvia	Templado	Alta	Fuerte	No

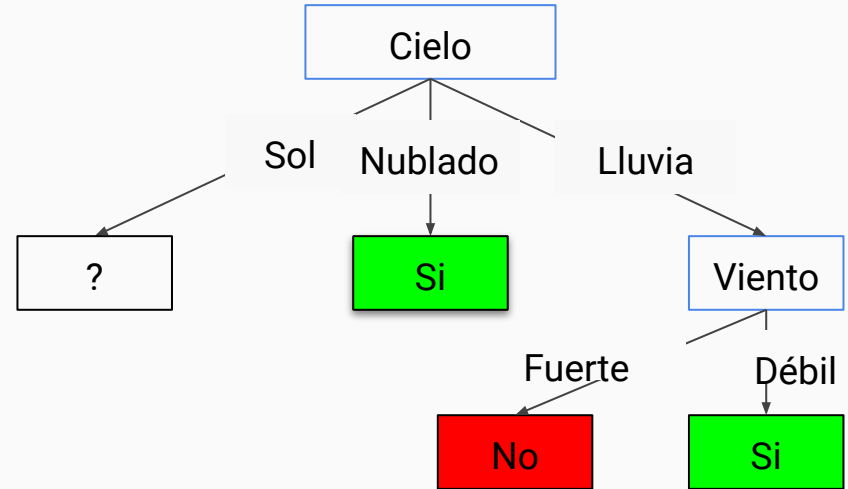
Ahora vamos por la rama **Lluvia** y elijamos **Viento**



Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí

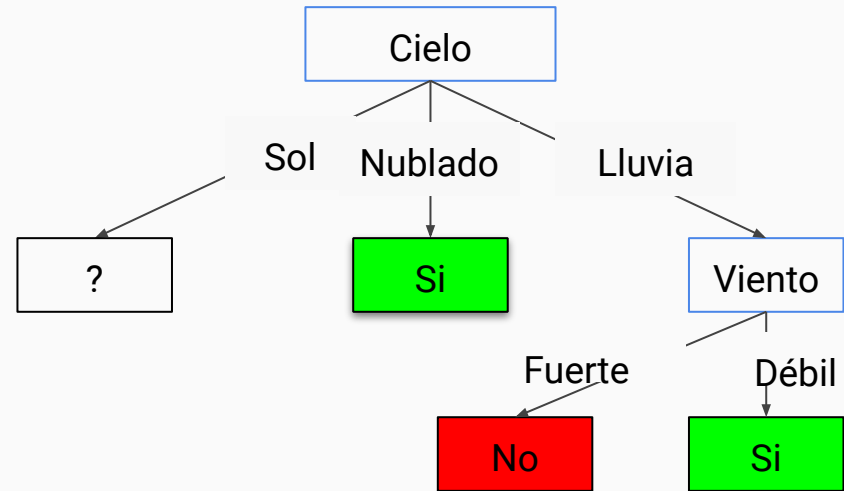
Ahora vamos por la rama **Lluvia** y elijamos **Viento**



Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí

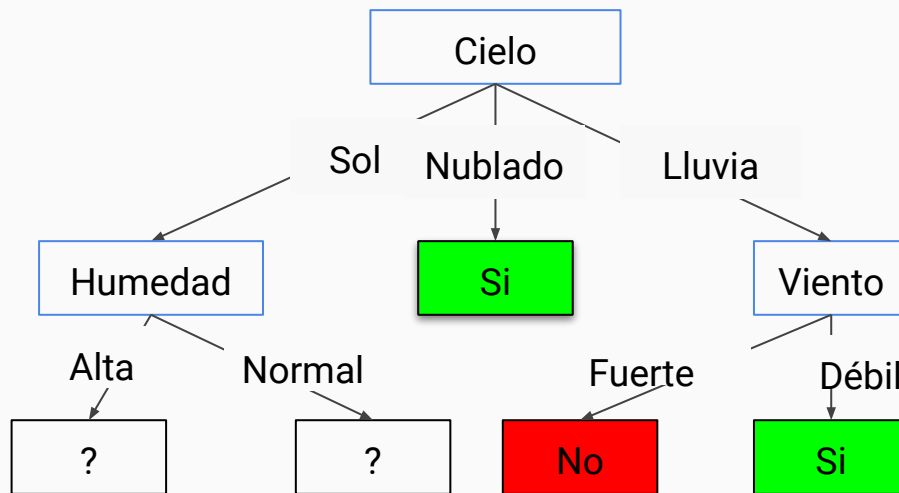
Ahora tenemos que explotar la rama de Cielo==Sol, ¿Qué atributo elegimos ahora?



Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí

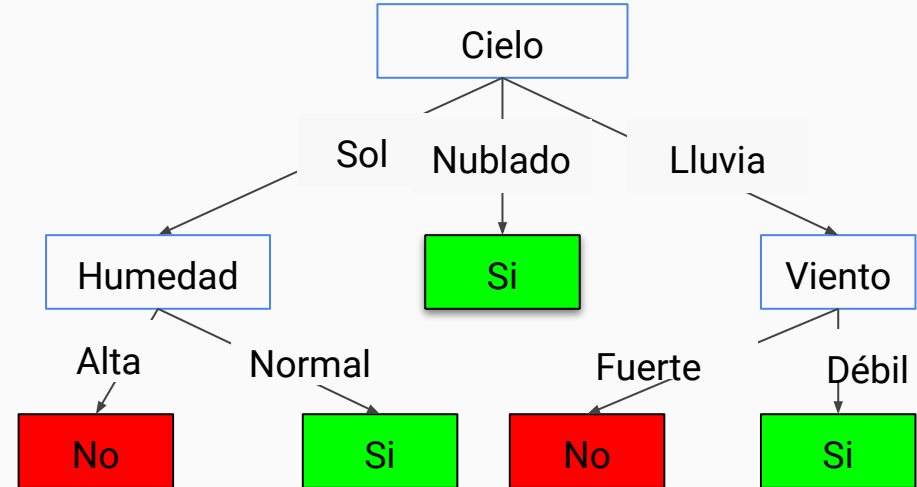
Ahora tenemos que explotar la rama de Cielo==Sol,
¿Qué atributo elegimos ahora? **Humedad**



Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí

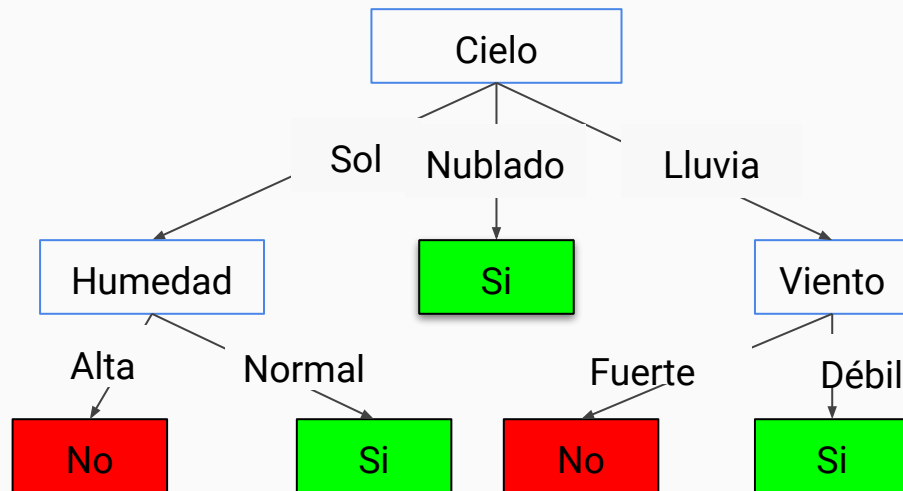
Listo, tenemos entrenado el modelo. ¿Cómo hacemos para evaluar una instancia nueva?



Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí

Listo, tenemos entrenado el modelo. ¿Cómo hacemos para evaluar una instancia nueva?



Por ejemplo:

< Cielo: Sol,
Temperatura: Calor,
Humedad: Normal,
Viento: Fuerte

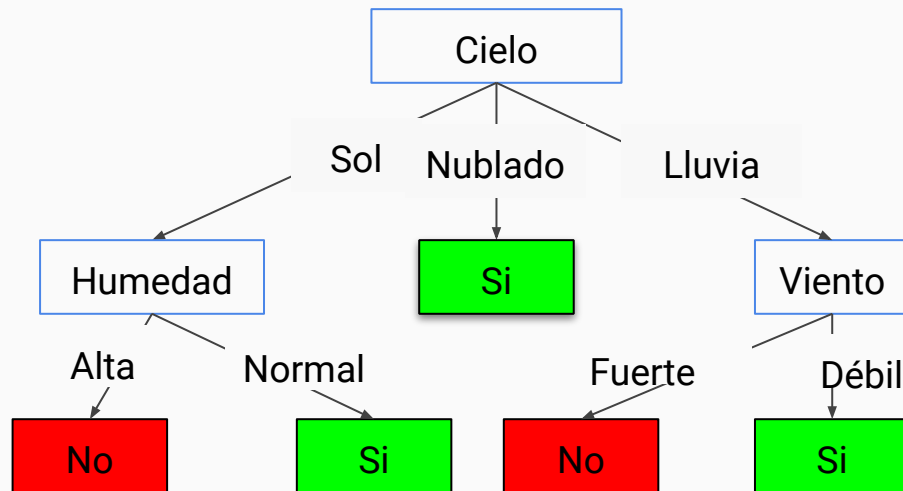


>

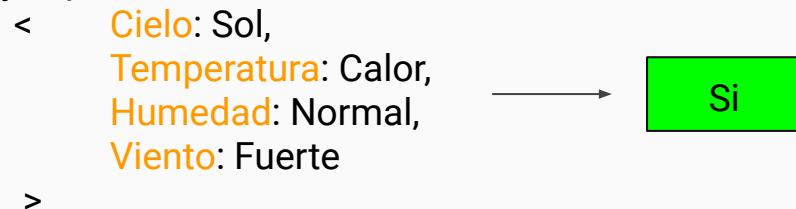
Árboles de Decisión

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí

Listo, tenemos entrenado el modelo. ¿Cómo hacemos para evaluar una instancia nueva?



Por ejemplo:



Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

- 1) Elijo el “mejor” atributo para nodo actual lo guardo en A
- 2) Asignar A como atributo de decisión del nodo_actual.
- 3) Para cada valor de A, crear un nuevo hijo del nodo_actual.
- 4) Clasificar (repartir) las instancias en los nuevos nodos (según el valor de A).
- 5) Si las instancias están clasificadas perfectamente: ¡FIN!
Si no: iterar sobre los nuevos nodos.

Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Agregamos reglas:

- El “mejor atributo” es el de más a la izquierda
- El árbol no puede tener más de profundidad 2

Cielo	Temperatura	Humedad	Viento	Tenis
Sol	Calor	Alta	Débil	No
Sol	Calor	Alta	Fuerte	No
Nublado	Calor	Alta	Débil	Sí
Lluvia	Templado	Alta	Débil	Sí
Lluvia	Frío	Normal	Débil	Sí
Lluvia	Frío	Normal	Fuerte	No
Nublado	Frío	Normal	Fuerte	Sí
Sol	Templado	Alta	Débil	No
Sol	Frío	Normal	Débil	Sí
Lluvia	Templado	Normal	Débil	Sí
Sol	Templado	Normal	Fuerte	Sí
Nublado	Templado	Alta	Fuerte	Sí
Nublado	Calor	Normal	Débil	Sí
Lluvia	Templado	Alta	Fuerte	No

Pseudo-algoritmo

- 1) Elijo el “mejor” atributo para nodo actual lo guardo en A
- 2) Asignar A como atributo de decisión del nodo_actual.
- 3) Para cada valor de A, crear un nuevo hijo del nodo_actual.
- 4) Clasificar (repartir) las instancias en los nuevos nodos (según el valor de A).
- 5) Si las instancias están clasificadas perfectamente: ¡FIN!
Si no: iterar sobre los nuevos nodos.

Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Pasos

1. Elijo **Cielo** como el “mejor” atributo para nodo actual
2. Para cada valor de **Cielo** creamos un nodo hijo: Estos serian Sol, Nublado y Lluvia
3. Repartimos las instancias en los nuevos nodos
4. Si las instancias están clasificadas ok: fin.
Sino itero

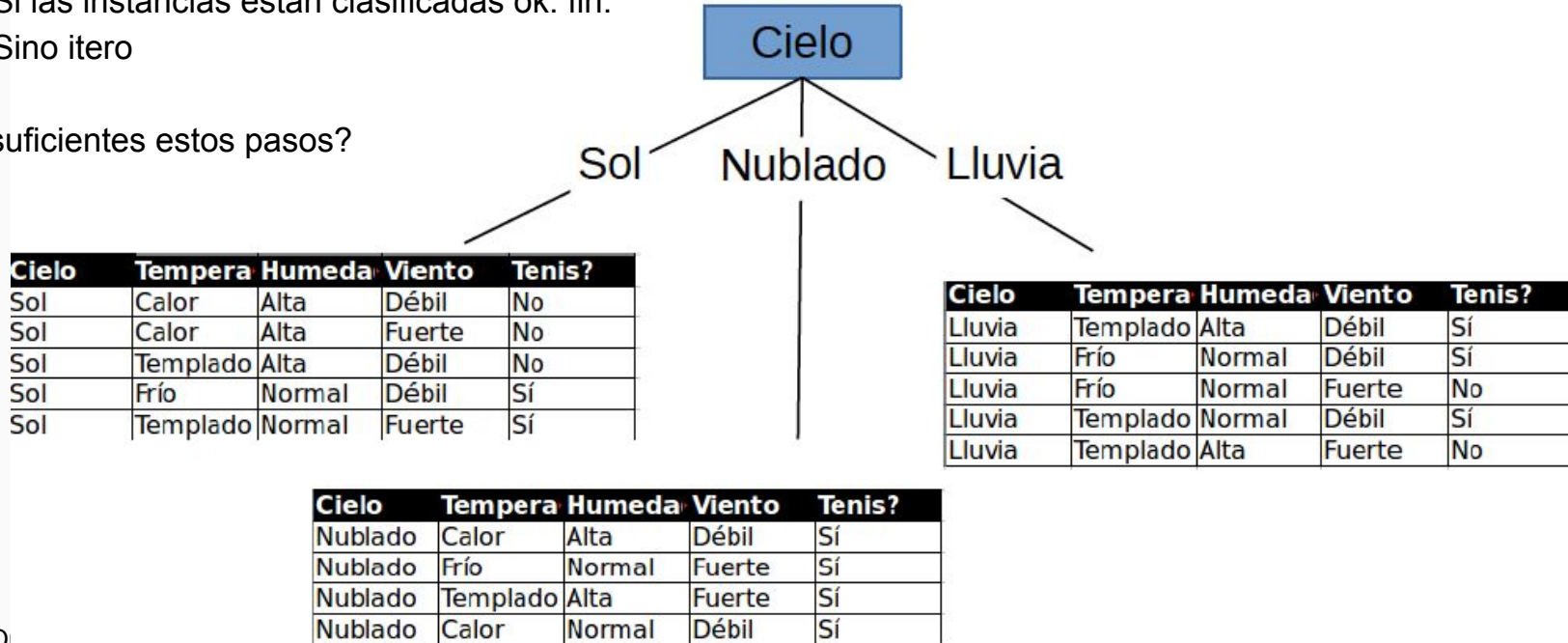
¿Son suficientes estos pasos?

Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Pasos

1. Elijo **Cielo** como el “mejor” atributo para nodo actual
2. Para cada valor de **Cielo** creamos un nodo hijo: Estos serian Sol, Nublado y Lluvia
3. Repartimos las instancias en los nuevos nodos
4. Si las instancias están clasificadas ok: fin.
Sino itero

¿Son suficientes estos pasos?

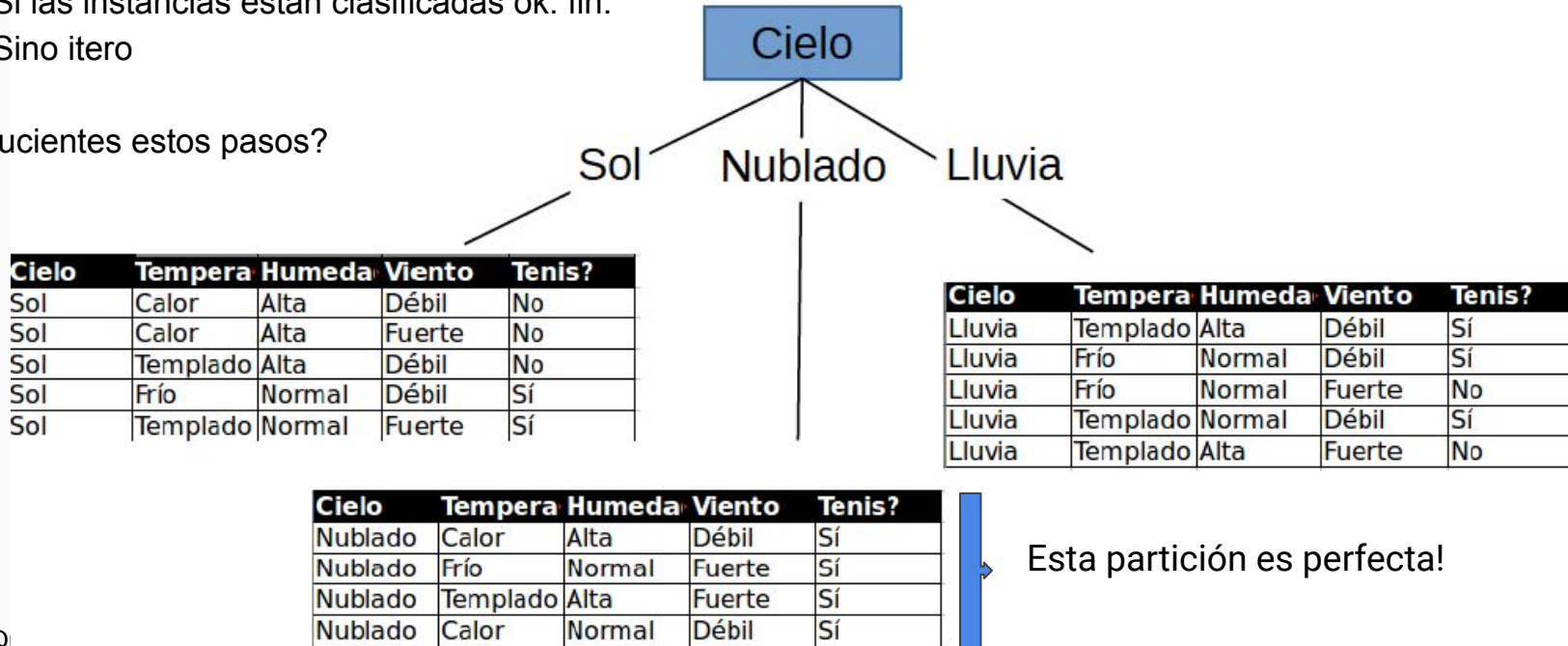


Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Pasos

1. Elijo **Cielo** como el “mejor” atributo para nodo actual
2. Para cada valor de **Cielo** creamos un nodo hijo: Estos serian Sol, Nublado y Lluvia
3. Repartimos las instancias en los nuevos nodos
4. Si las instancias están clasificadas ok: fin.
Sino itero

¿Son fucientes estos pasos?

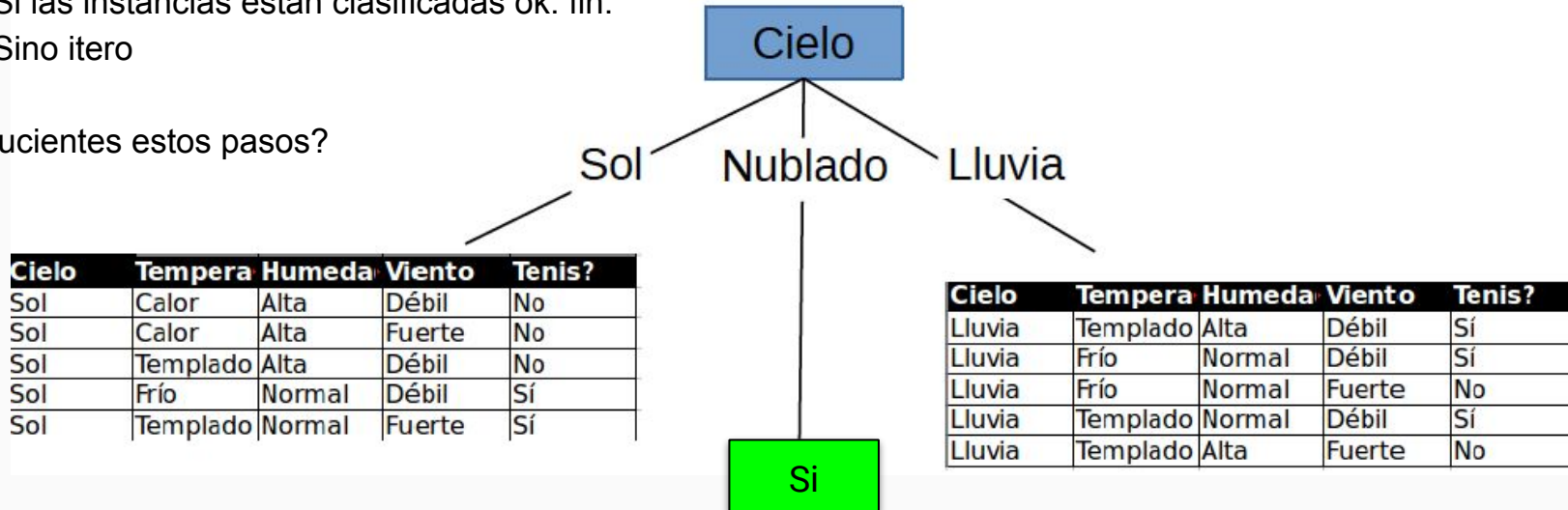


Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Pasos

1. Elijo **Cielo** como el “mejor” atributo para nodo actual
2. Para cada valor de **Cielo** creamos un nodo hijo: Estos serian Sol, Nublado y Lluvia
3. Repartimos las instancias en los nuevos nodos
4. Si las instancias están clasificadas ok: fin.
Sino itero

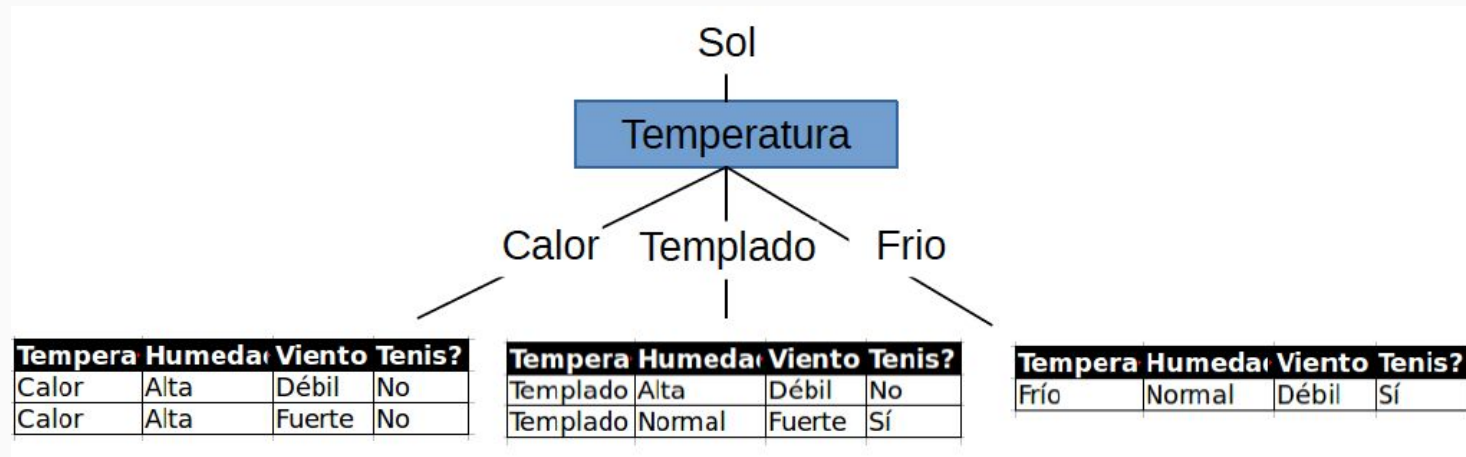
¿Son suficientes estos pasos?



Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Pasos

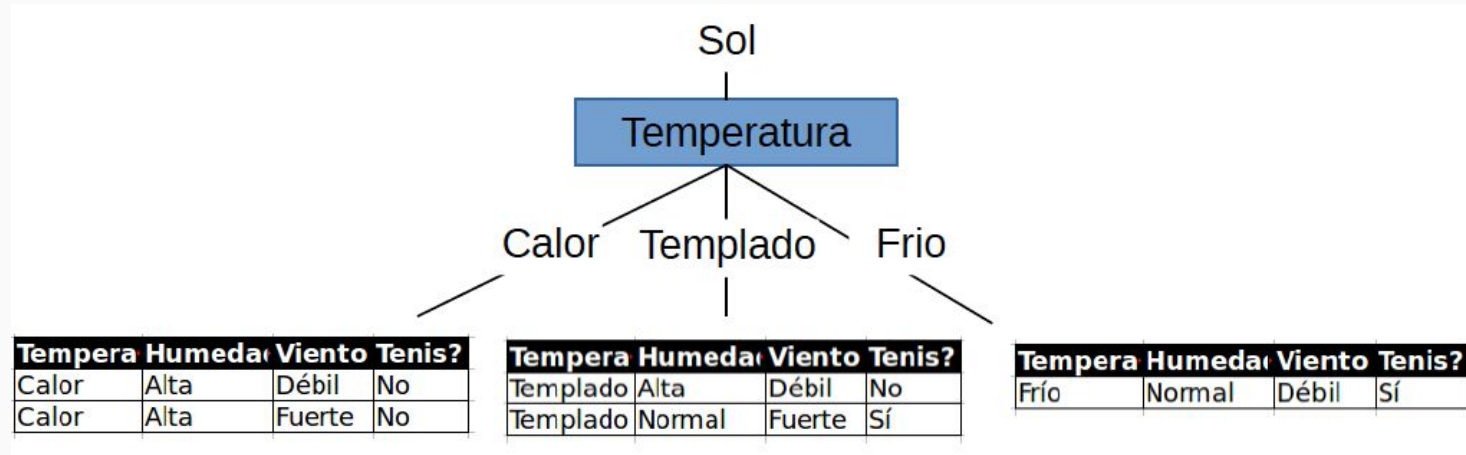
5. Sigamos ahora por la rama Sol



Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Pasos

5. Sigamos ahora por la rama Sol
6. Las ramas que salen de [Temperatura==Calor] y [Temperatura==Frío] son perfectas, pero [Temperatura==Templado] no!

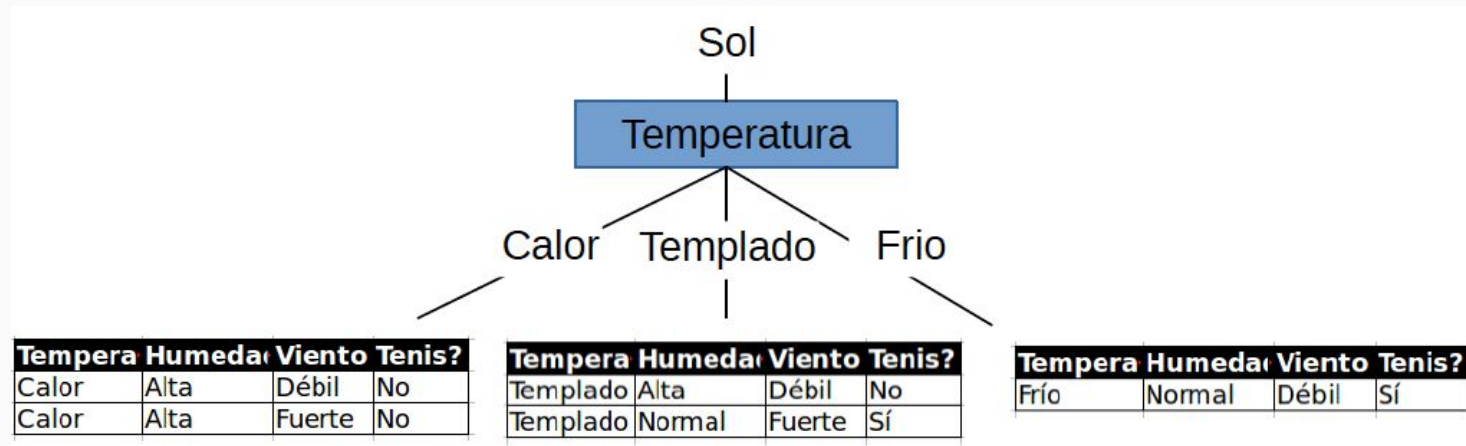


Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Pasos

5. Sigamos ahora por la rama Sol

6. Las ramas que salen de [Temperatura==Calor] y [Temperatura==Frío] son perfectas, pero [Temperatura==Templado] no!



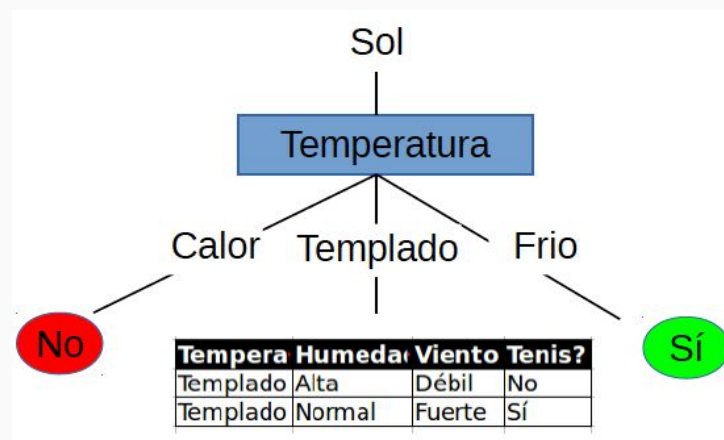
Debemos seguir partiendo para templado, pero... podemos? El sesgo inductivo decía que teníamos que tener altura máxima de 2!

Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Pasos

5. Sigamos ahora por la rama Sol

6. Las ramas que salen de [Temperatura==Calor] y [Temperatura==Frío] son perfectas, pero [Temperatura==Templado] no!

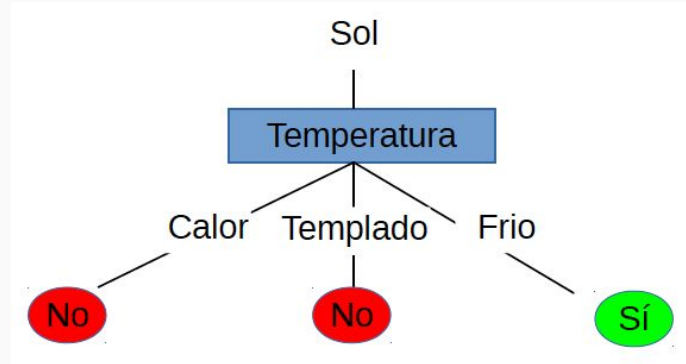


Debemos seguir partiendo para templado, pero... podemos? El sesgo inductivo decía que teníamos que tener altura máxima de 2!

¿Que clase elegimos para [Temperatura==Templado]? Falta información en el sesgo inductivo!

Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

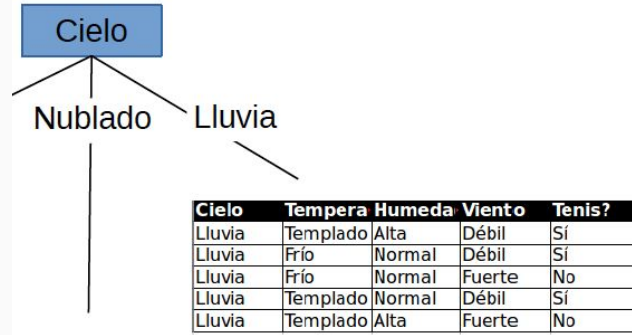
Nueva regla: Sino puede partir más (bien porque llegué a la altura máxima o porque me quede sin atributos) devolver la clase más frecuente, si empatan devolver la primera.



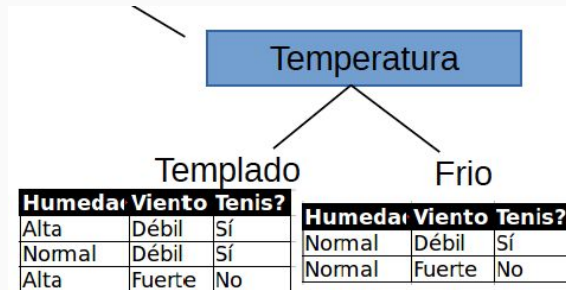
Falta hacer lo mismo para Cielo==Lluvia

Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Falta hacer lo mismo para Cielo==Lluvia

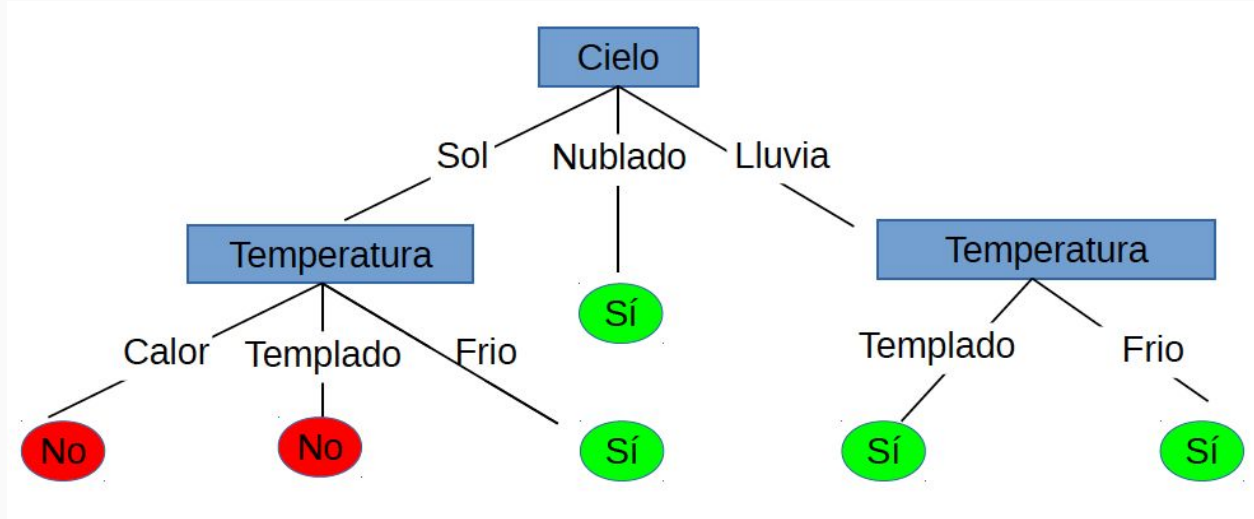


Obtengo esta partición, y llegue a la máxima altura, aplico la nueva regla



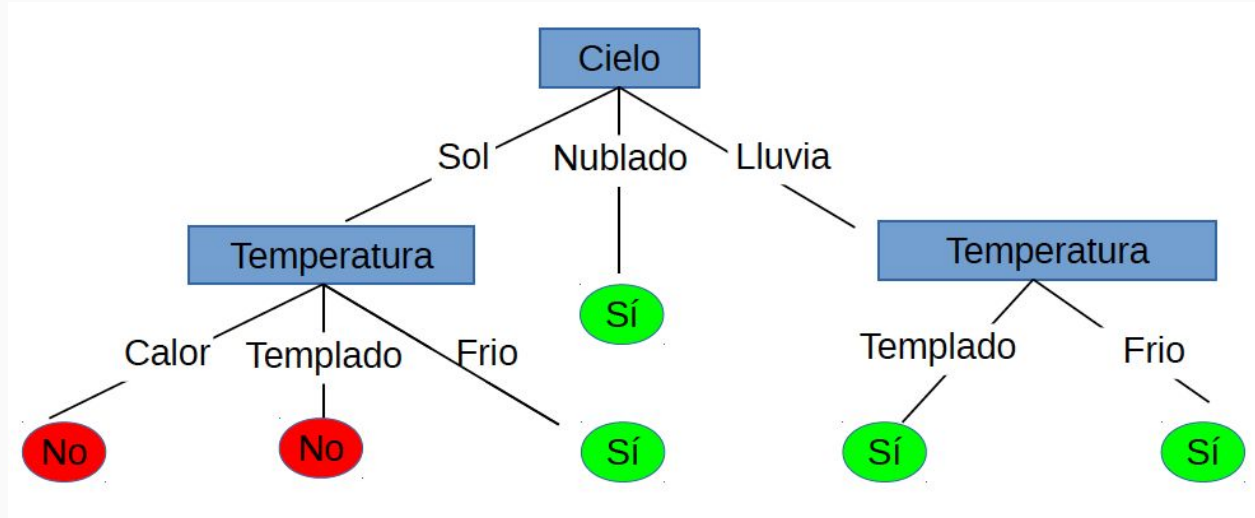
Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Modelo entrenado con el pseudo-algoritmo



Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Modelo entrenado con el pseudo-algoritmo

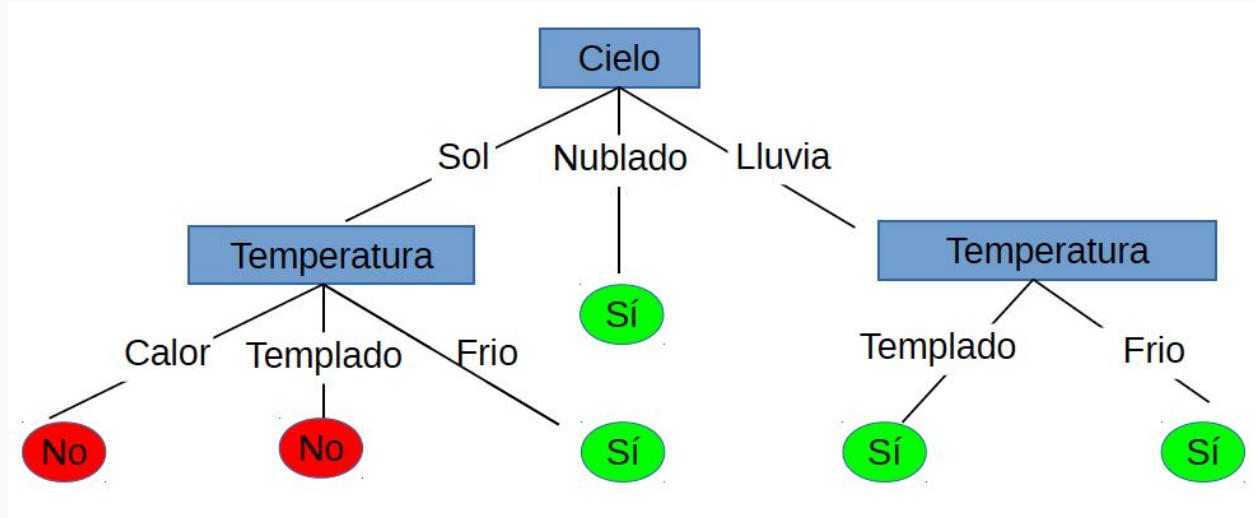


Lo pruebo:

- {Cielo: Nublado, ... }
- {Cielo: Sol , Temperatura: Templeado, ...}
- {Cielo: Lluvia, Temperatura: Calor }

Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Modelo entrenado con el pseudo-algoritmo

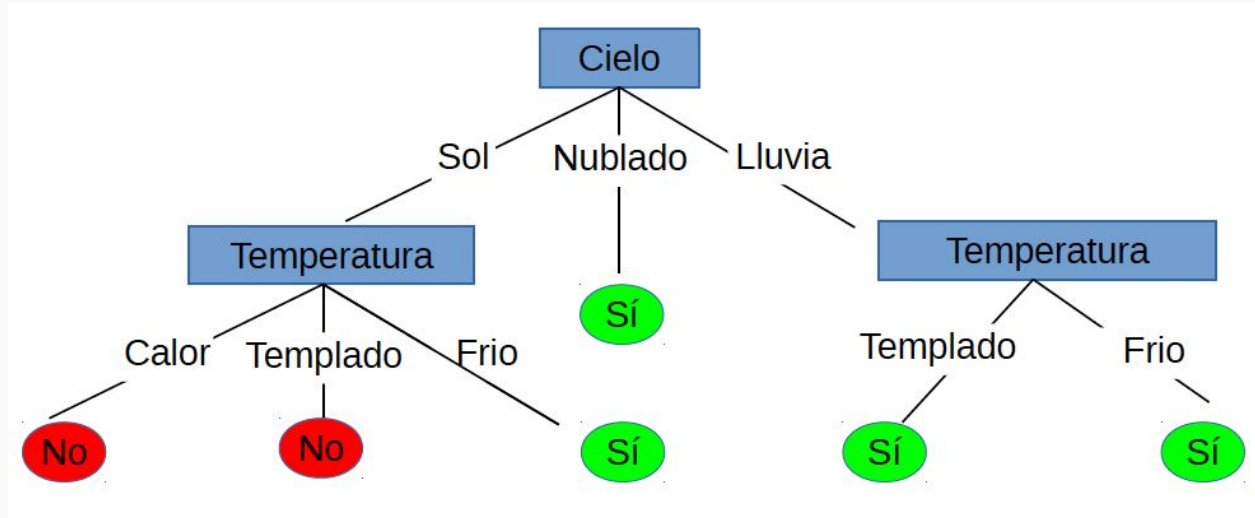


Lo pruebo:

- {Cielo: Nublado, ... } **Si**
- {Cielo: Sol , Temperatura: Templeado, ...}
- {Cielo: Lluvia, Temperatura: Calor }

Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Modelo entrenado con el pseudo-algoritmo

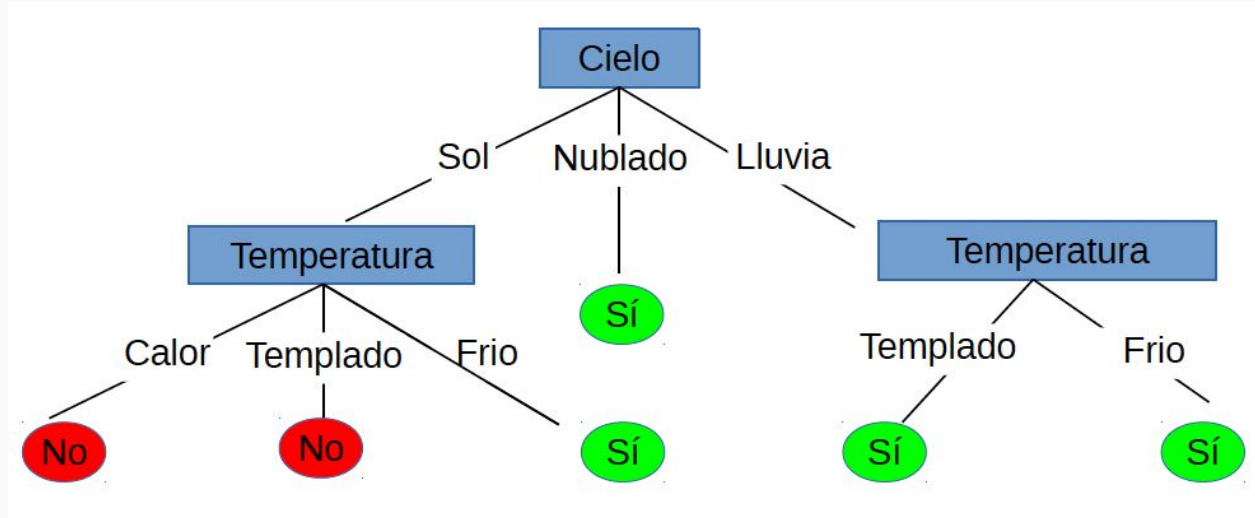


Lo pruebo:

- {Cielo: Nublado, ... } **Si**
- {Cielo: Sol , Temperatura: Templeado, ...} **No**
- {Cielo: Lluvia, Temperatura: Calor }

Árboles de Decisión: Pseudo-Algoritmo (ID3) para entrenar

Modelo entrenado con el pseudo-algoritmo



Lo pruebo:

- {Cielo: Nublado, ... } **Si**
- {Cielo: Sol , Temperatura: Templeado, ...} **No**
- {Cielo: Lluvia, Temperatura: Calor } **¿¿¿¿¿?????**

¿Qué pasó? **Nos sigue faltando info en el sesgo inductivo.** Idea: default por nodo?

Árboles de Decisión: Mejor atributo

¿Qué pasa con la búsqueda del mejor atributo?

Previamente hicimos: “el de más a la izquierda”

Hay variantes un poco más interesantes

Por ejemplo: **Maximizar la ganancia de información**

Ganancia de información:

Es la reducción de **entropía** de la muestra S (respecto de la variable objetivo Y), después de clasificar las instancias según A .

Árboles de Decisión: Entropía

En este contexto:

Entropía de una muestra S con respecto a una variable objetivo Y :

$$H_Y(S) = \sum_{i=1}^c -p_i \log_2 p_i$$

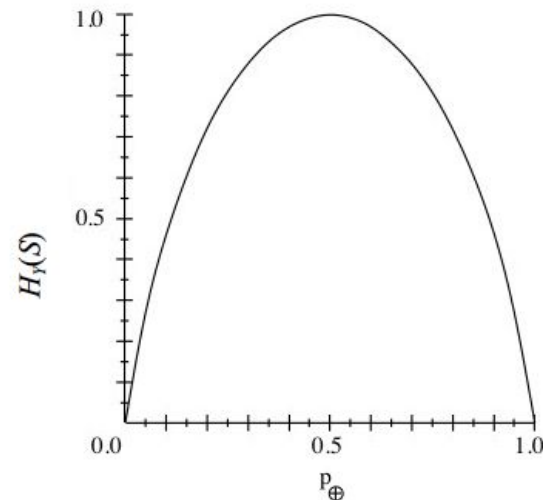
c : cantidad de valores posibles de Y

p_i : probabilidad en S de que Y tome el valor i -ésimo

La entropía mide el **grado de impureza** de S respecto a Y

Ejemplo: $c=2$

$$H_Y(S) = -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$$

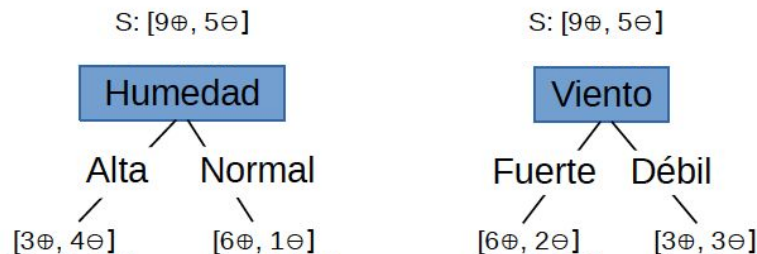


Árboles de Decisión: Ganancia de información

$$Ganancia(S, A) = H(S) - \sum_{v \in \text{Valores}(A)} \frac{|S_v|}{|S|} H(S_v)$$

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

$$S_v = \{ s \in S \mid A(s) = v \}$$



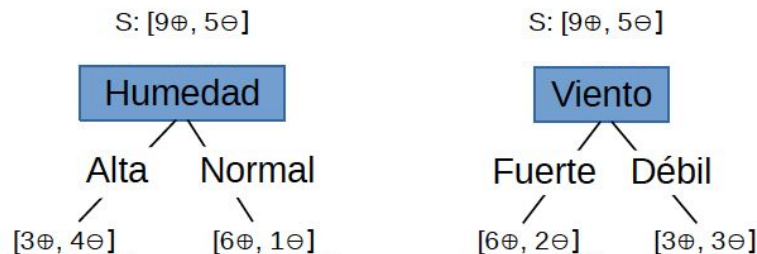
$$\begin{aligned} Ganancia(S, \text{Humedad}) &= \\ Ganancia(S, \text{Viento}) &= \end{aligned}$$

Árboles de Decisión: Ganancia de información

$$Ganancia(S, A) = H(S) - \sum_{v \in \text{Valores}(A)} \frac{|S_v|}{|S|} H(S_v)$$

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

$$S_v = \{ s \in S \mid A(s) = v \}$$



$$\begin{aligned} Ganancia(S, \text{Humedad}) &= \\ Ganancia(S, \text{Viento}) &= \end{aligned}$$

$$S: [9\oplus, 5\ominus]$$

$$E = -9/14 \cdot \log_2(9/14) - 5/14 \cdot \log_2(5/14) = 0.940$$

$$[3\oplus, 4\ominus]$$

$$E = -3/7 \cdot \log_2(3/7) - 4/7 \cdot \log_2(4/7) = 0.985$$

$$[6\oplus, 1\ominus]$$

$$E = -6/7 \cdot \log_2(6/7) - 1/7 \cdot \log_2(1/7) = 0.592$$

$$[6\oplus, 2\ominus]$$

$$E = -6/8 \cdot \log_2(6/8) - 2/8 \cdot \log_2(2/8) = 0.811$$

$$[3\oplus, 3\ominus]$$

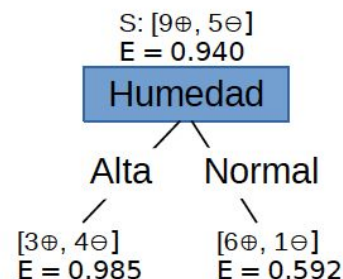
$$E = -3/6 \cdot \log_2(3/6) - 3/6 \cdot \log_2(3/6) = 1.00$$

Árboles de Decisión: Ganancia de información

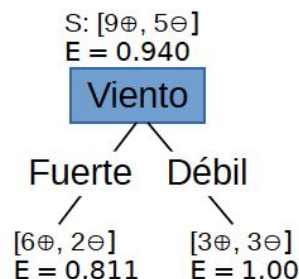
$$Ganancia(S, A) = H(S) - \sum_{v \in \text{Valores}(A)} \frac{|S_v|}{|S|} H(S_v)$$

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

$$S_v = \{ s \in S \mid A(s) = v \}$$



$$\begin{aligned} Ganancia(S, \text{Humedad}) &= \\ Ganancia(S, \text{Viento}) &= \end{aligned}$$



$$S: [9\oplus, 5\ominus]$$

$$E = -9/14 \cdot \log_2(9/14) - 5/14 \cdot \log_2(5/14) = 0.940$$

$$[3\oplus, 4\ominus]$$

$$E = -3/7 \cdot \log_2(3/7) - 4/7 \cdot \log_2(4/7) = 0.985$$

$$[6\oplus, 1\ominus]$$

$$E = -6/7 \cdot \log_2(6/7) - 1/7 \cdot \log_2(1/7) = 0.592$$

$$[6\oplus, 2\ominus]$$

$$E = -6/8 \cdot \log_2(6/8) - 2/8 \cdot \log_2(2/8) = 0.811$$

$$[3\oplus, 3\ominus]$$

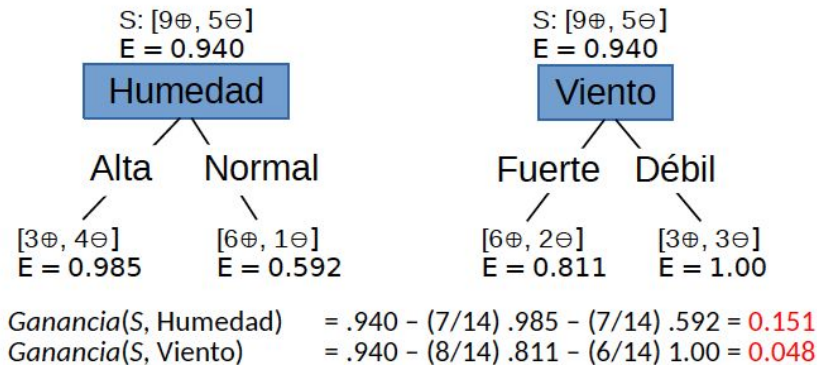
$$E = -3/6 \cdot \log_2(3/6) - 3/6 \cdot \log_2(3/6) = 1.00$$

Árboles de Decisión: Ganancia de información

$$Ganancia(S, A) = H(S) - \sum_{v \in \text{Valores}(A)} \frac{|S_v|}{|S|} H(S_v)$$

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

$$S_v = \{ s \in S \mid A(s) = v \}$$



Elegimos **Humedad**

$$S: [9\oplus, 5\ominus]$$

$$E = -9/14 \cdot \log_2(9/14) - 5/14 \cdot \log_2(5/14) = 0.940$$

$$[3\oplus, 4\ominus]$$

$$E = -3/7 \cdot \log_2(3/7) - 4/7 \cdot \log_2(4/7) = 0.985$$

$$[6\oplus, 1\ominus]$$

$$E = -6/7 \cdot \log_2(6/7) - 1/7 \cdot \log_2(1/7) = 0.592$$

$$[6\oplus, 2\ominus]$$

$$E = -6/8 \cdot \log_2(6/8) - 2/8 \cdot \log_2(2/8) = 0.811$$

$$[3\oplus, 3\ominus]$$

$$E = -3/6 \cdot \log_2(3/6) - 3/6 \cdot \log_2(3/6) = 1.00$$

Árboles de Decisión: ¿Taller?

¿Esto no era un taller? Si

Manos a la obra:

- Ejercicios para hacer en papel
- Preguntas para pensar
- Implementación incompleta de árboles de decisión que tienen que completar (o hacer desde cero?)
- Comparación con Scikit-learn

Link: colab <https://colab.research.google.com/drive/1zCZheJIsDh9W-zBPVCB874hSKFrqnL4F>