

Aprendizaje Automático
Segundo Cuatrimestre de 2016

Árboles de Decisión

Clase práctica



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Pipeline trabajo

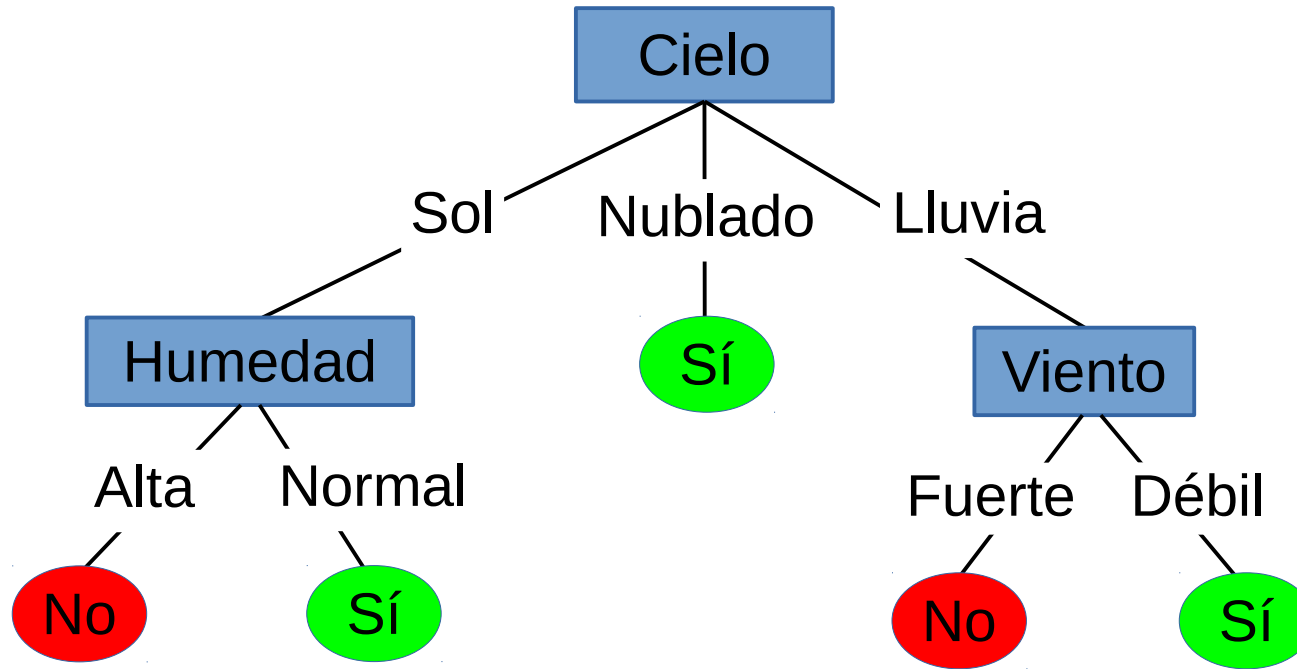
1) Armamos/Tenemos un dataset con **n** instancias y **m** atributos. A su vez, tenemos un vector **target**, es decir un vector que nos dice **para cada instancia cual es la clase a la que pertenece**.

2) Entrenamos el modelo con el dataset de entrenamiento. Este paso genera un árbol entrenado

3) Con el modelo entrenado (el árbol) podemos predecir nuevas instancias

Vamos de atras para adelante...

3) ¿Cómo predecimos nuevas instancias?



(Modelo entrenado)

Por ejemplo:

- { Humedad:Alta , Cielo:Lluvia, Viento:Debil }
- { Humedad: Normal, Cielo:Nublado, Viento:Fuerte}

2) ¿Cómo entrenamos el modelo?

Modificamos sesgo intuitivo de la teoría con las siguientes dos reglas:

- el mejor atributo ahora es el primero de izquierda a derecha (y no mas el que tenga maxima **Ganancia de Información**)
- el árbol tiene como **máxima altura 2**

Día	Cielo	Temperatura	Humedad	Viento	Tenis?
1	Sol	Calor	Alta	Débil	No
2	Sol	Calor	Alta	Fuerte	No
3	Nublado	Calor	Alta	Débil	Sí
4	Lluvia	Templado	Alta	Débil	Sí
5	Lluvia	Frío	Normal	Débil	Sí
6	Lluvia	Frío	Normal	Fuerte	No
7	Nublado	Frío	Normal	Fuerte	Sí
8	Sol	Templado	Alta	Débil	No
9	Sol	Frío	Normal	Débil	Sí
10	Lluvia	Templado	Normal	Débil	Sí
11	Sol	Templado	Normal	Fuerte	Sí
12	Nublado	Templado	Alta	Fuerte	Sí
13	Nublado	Calor	Normal	Débil	Sí
14	Lluvia	Templado	Alta	Fuerte	No

2) ¿Cómo entrenamos el modelo?

Primera iteracion...

1) Elegimos el mejor atributo:

Cielo pues es el primero de izquierda a derecha

2) Para cada valor de **Cielo** creamos un nodos hijos:

Sol, Nublado, Lluvia

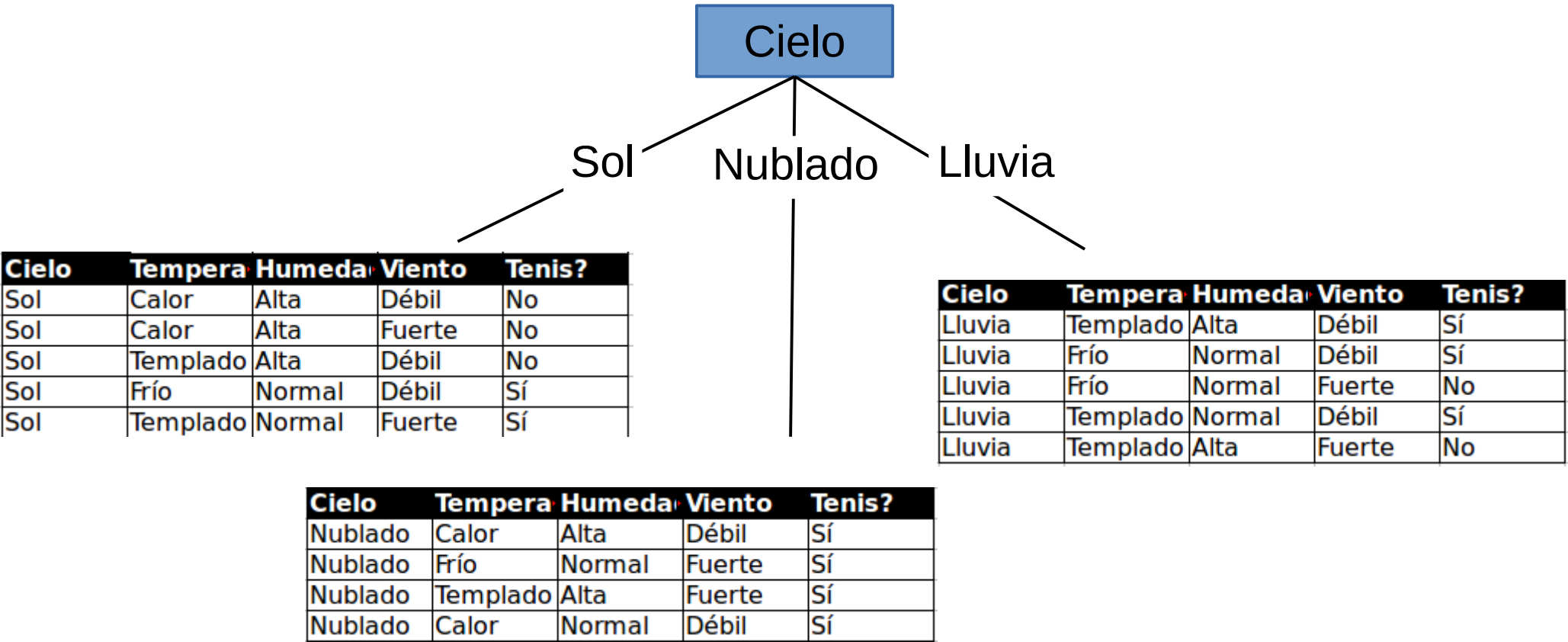
3) Repartimos las instancias en los nuevos nodos (según el valor de **Cielo**).

4) Si las instancias están clasificadas **perfectamente**: FIN

Si no: iterar sobre los nuevos nodos.

Alcanza este procedimiento?

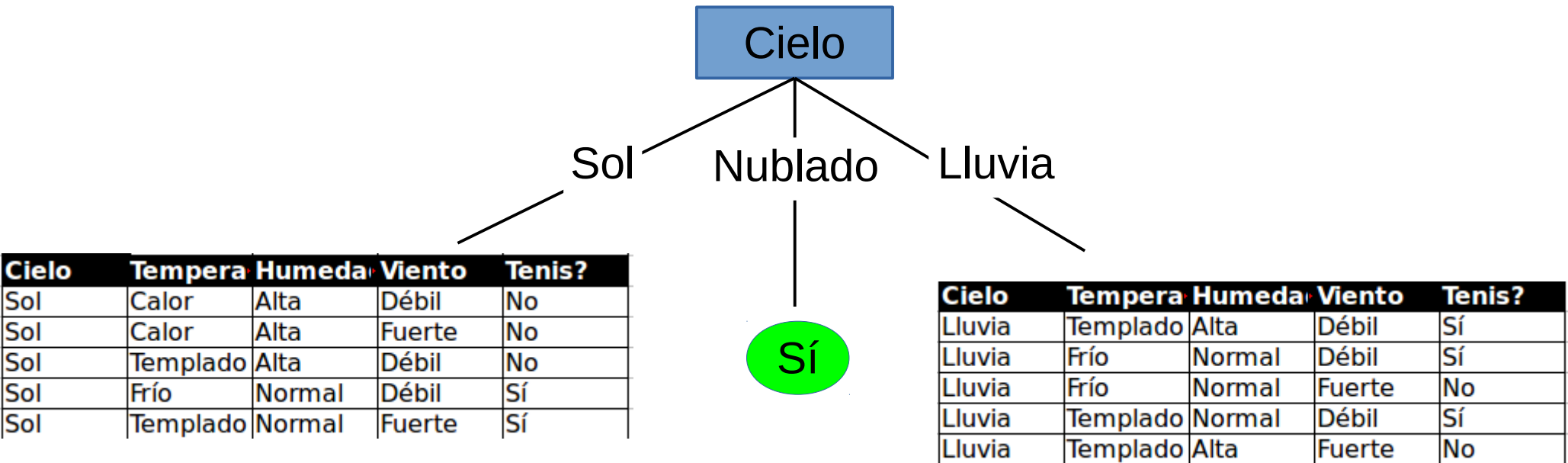
2) ¿Cómo entrenamos el modelo?



4) Si las instancias están clasificadas **perfectamente**: FIN
Si no: iterar sobre los nuevos nodos.

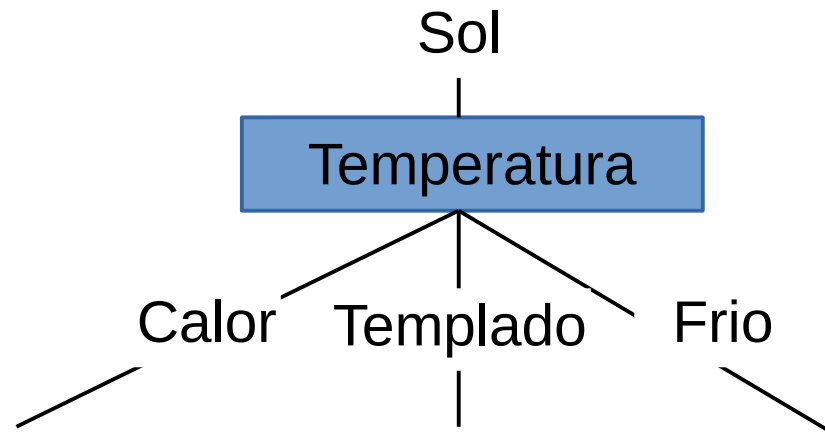
Hay alguna instancia perfectamente clasificada? **Nublado**

2) ¿Cómo entrenamos el modelo?



Iteramos nuevamente, tomamos el siguiente atributo: **Temperatura** (para cada sub árbol)

2) ¿Cómo entrenamos el modelo?



Tempera	Humeda	Viento	Tenis?
Calor	Alta	Débil	No
Calor	Alta	Fuerte	No

Tempera	Humeda	Viento	Tenis?
Templado	Alta	Débil	No
Templado	Normal	Fuerte	Sí

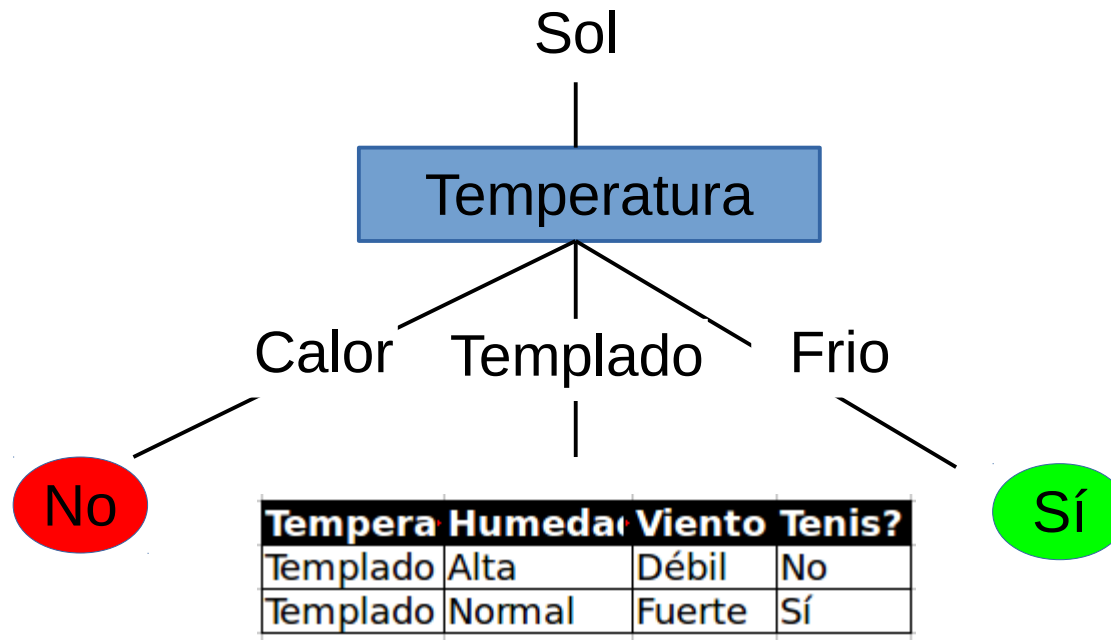
Tempera	Humeda	Viento	Tenis?
Frío	Normal	Débil	Sí

Las ramas que salen de Temp==Calor e Temp==Frio estan clasificadas **perfectas!**
Pero Temperatura==Templado **no :(**

Pero el sesgo inductivo dice que la **altura debe ser como maximo 2**, por lo que no podemos seguir dividiendo. ¿Qué clase elegimos?

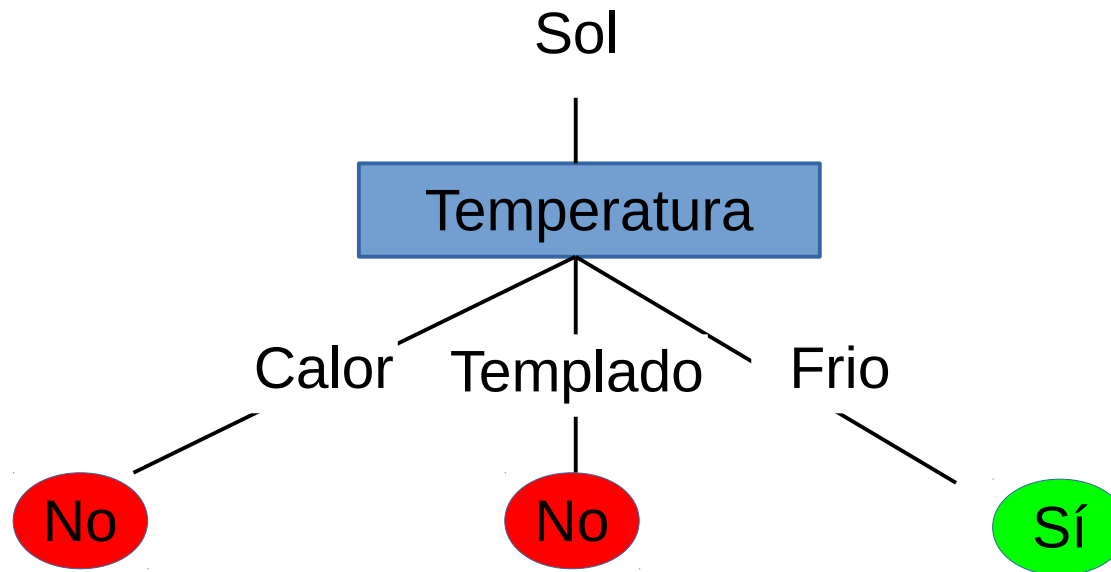
Falta informacion en el sesgo inductivo!

2) ¿Cómo entrenamos el modelo?



Agreguemos una **nueva regla**: devolver clase más frecuente, si empata devolver la primera.

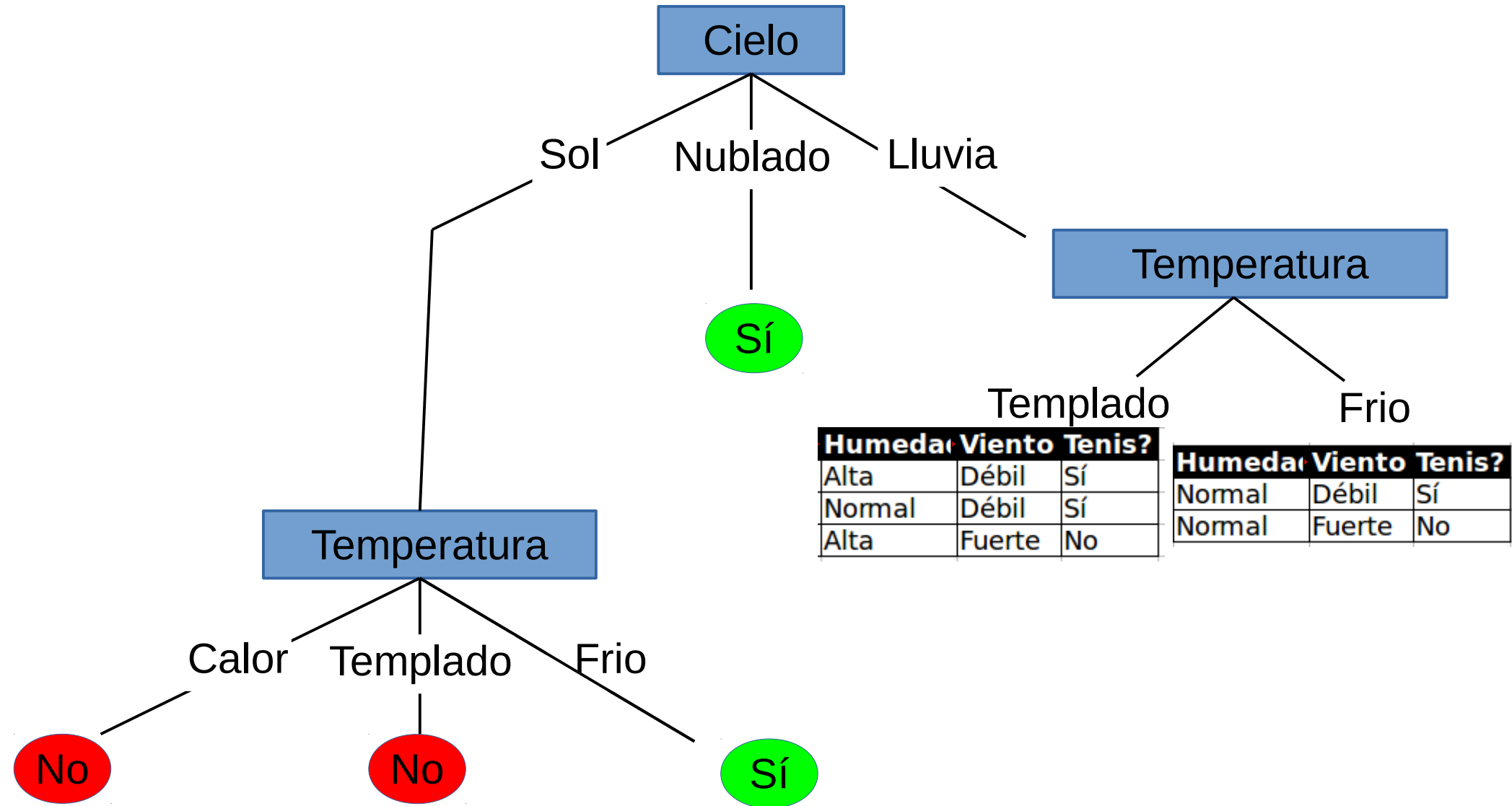
2) ¿Cómo entrenamos el modelo?



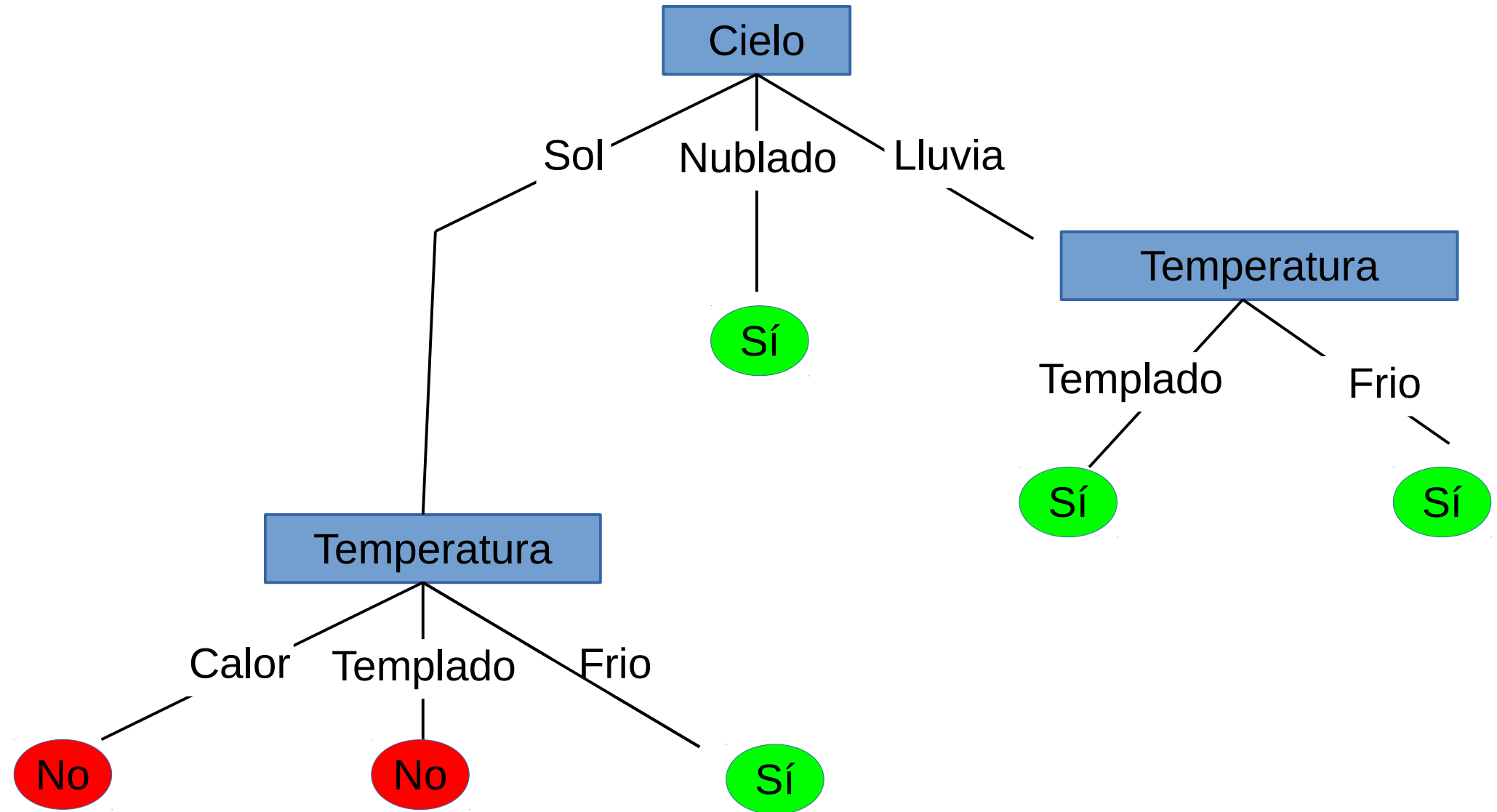
Agreguemos una **nueva regla**: devolver clase más frecuente, si empata devolver la primera.

Falta hacer lo mismo para Cielo==Luvia

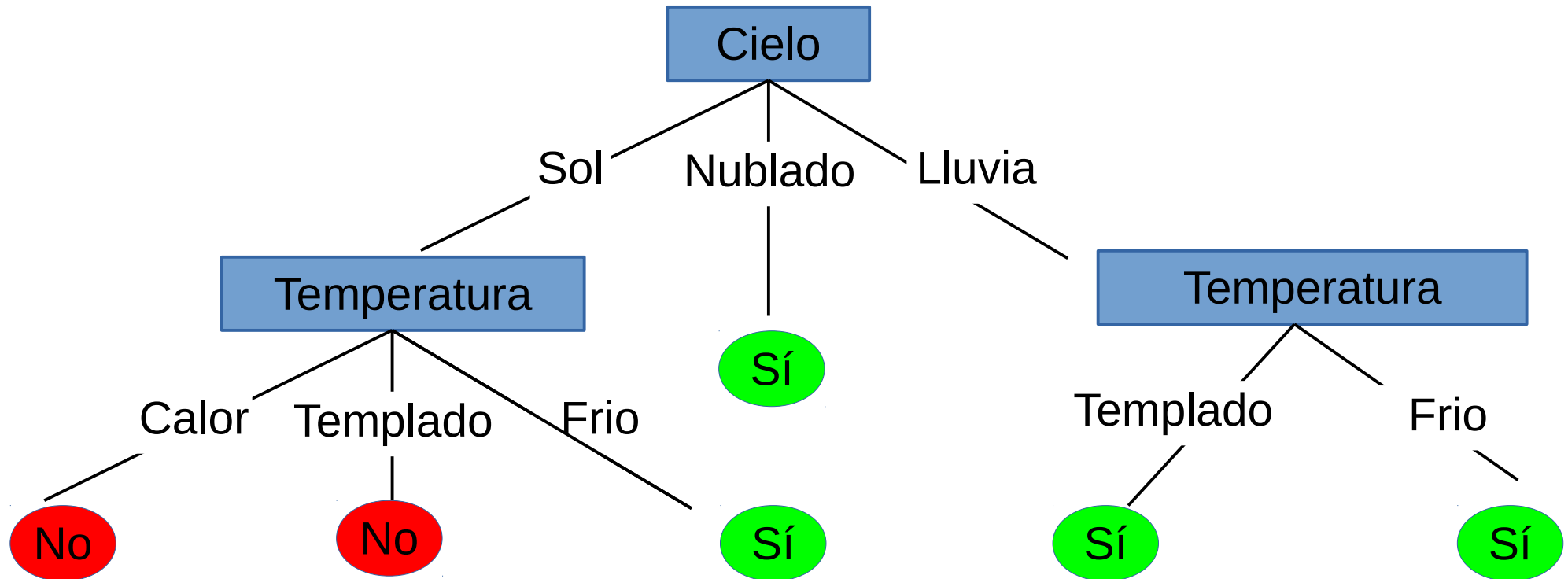
2) ¿Cómo entrenamos el modelo?



2) ¿Cómo entrenamos el modelo?



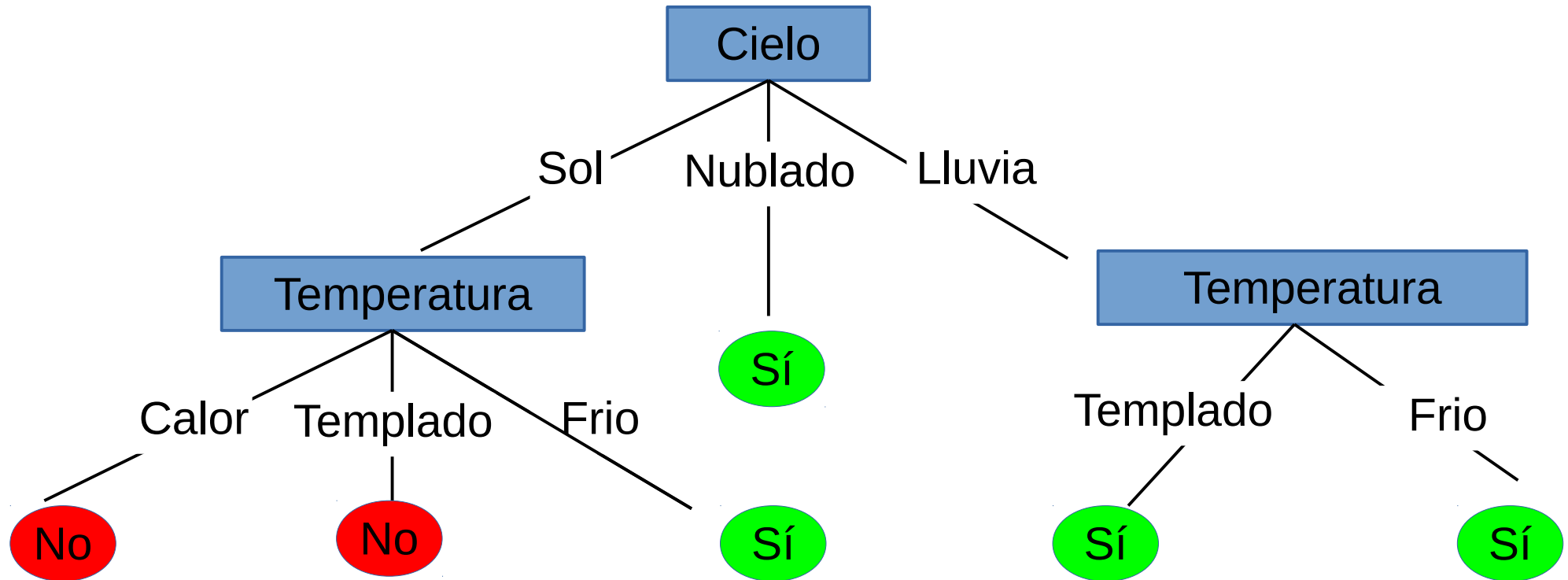
2) ¿Cómo entrenamos el modelo?



Qué clase predice para las entradas:

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

2) ¿Cómo entrenamos el modelo?



Hermoso... que predice para las entradas:

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

Nuevamente le falta algo al sesgo inductivo

¿Cómo seguimos?

- Ejercicios de la practica
 - Para pensar y discutir
 - Para hacer en papel
 - Para programar

Funciones de entropia y ganancia de informacion a completar en `tree_aa_alumnos.py` (bajar de pagina de la materia)

Nos vemos en el labo...