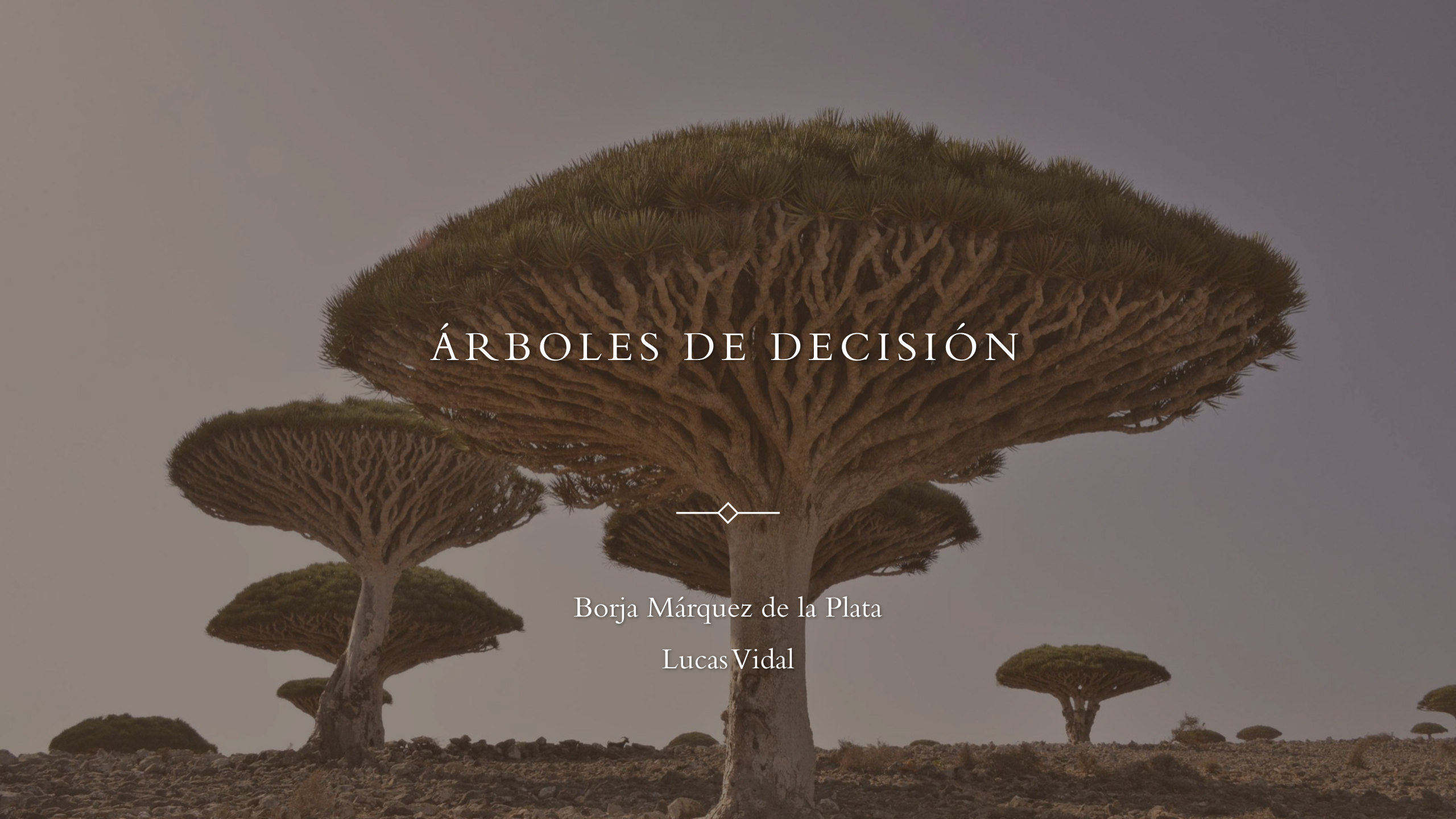


ÁRBOLES DE DECISIÓN

Borja Márquez de la Plata

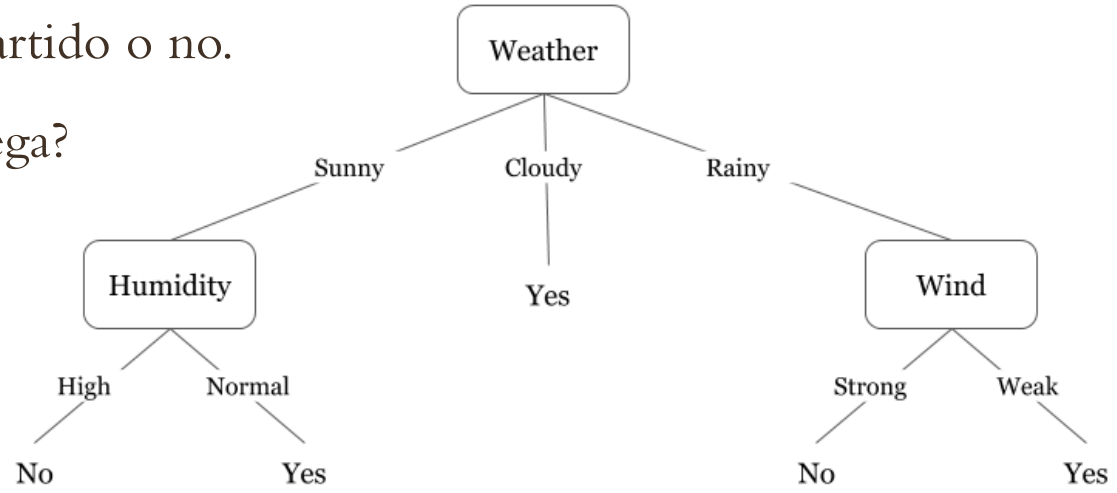
Lucas Vidal

Parte 1: Árboles de Decisión

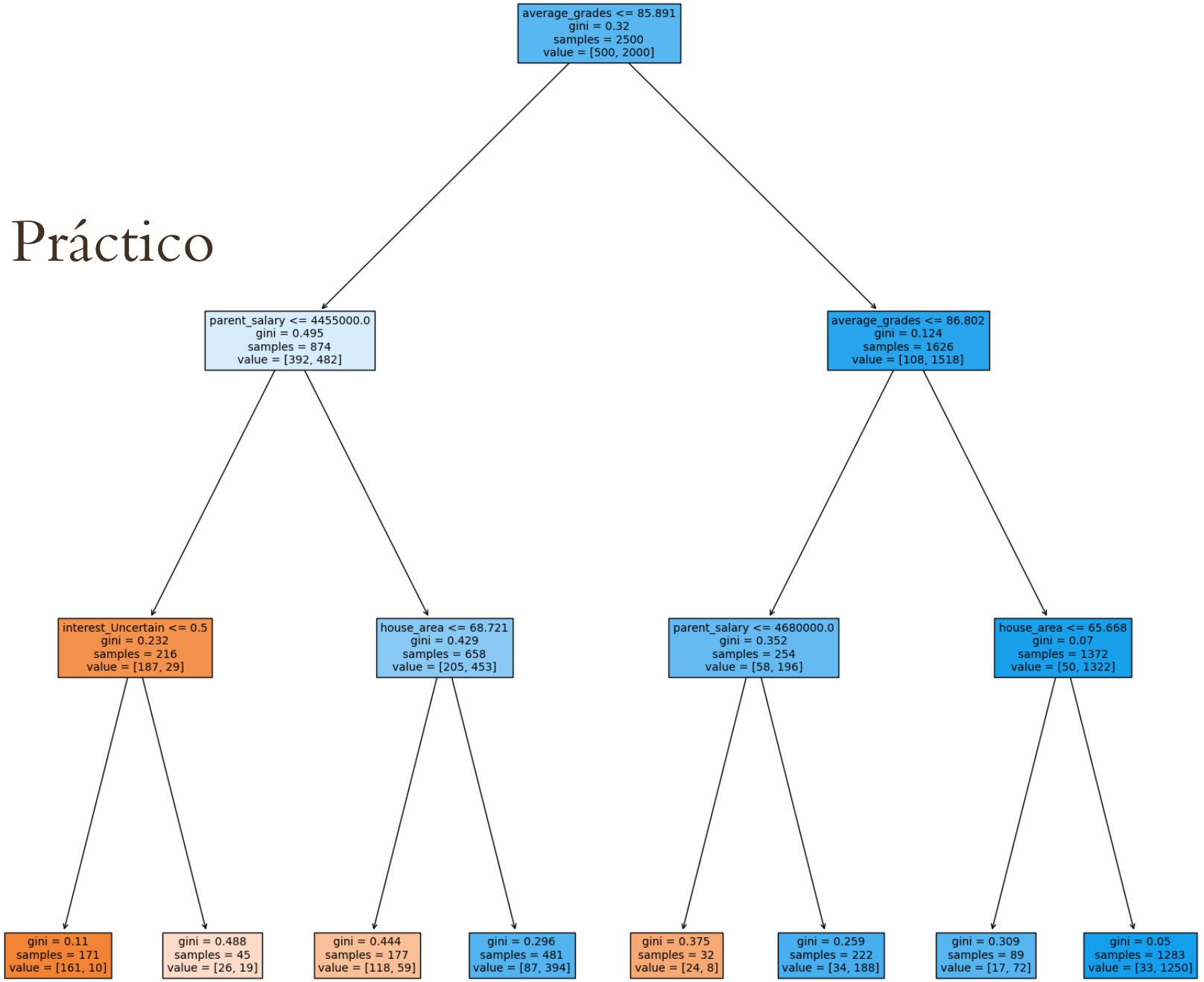
Objetivo: hacer preguntas que dividan a los datos.

Ejemplo: este árbol indica si se va a jugar un partido o no.

¿Si está soleado y con humedad muy alta, se juega?

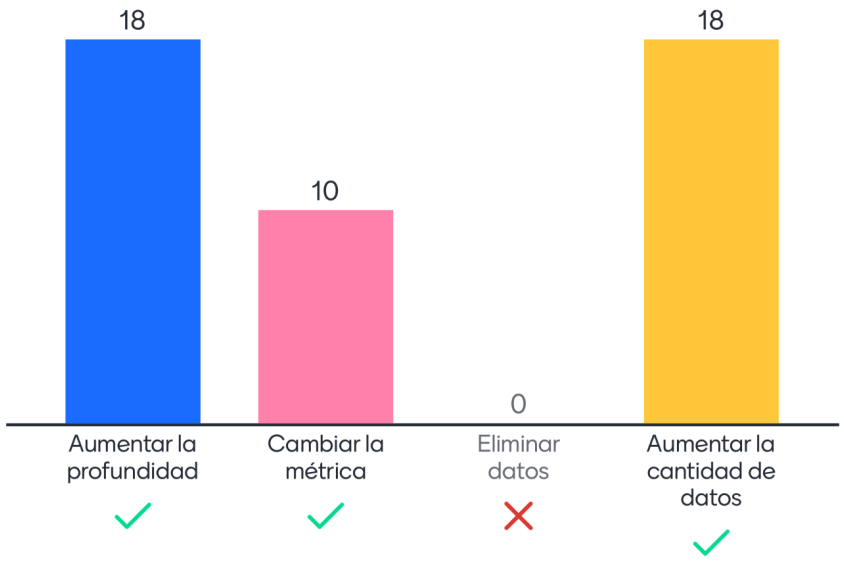


Ejemplo Práctico



Vamos al Menti…

¿Cómo se puede mejorar la precisión del árbol?



¿Cómo podemos mejorarlo?

Algunos de los parámetros que tenemos para manipular son:

- max_depth: altura máxima que el árbol puede tener.
- min_samples_leaf: cantidad mínima de datos que deben haber en una hoja.
- min_samples_split: cantidad mínima de datos que deben haber en un nodo para que se realice el split.
- criterion: “Gini”, “entropy”, “log_loss”

Vamos al Menti…

Accuracy vs Precision

$$accuracy = \frac{Aciertos\ correctos}{Total\ de\ datos}$$

$$precision = \frac{Verdaderos\ Positivos}{Verdaderos\ Positivos + Falsos\ Positivos}$$

Otras métricas

$$recall = \frac{TP}{TP + FN} \; ; TP = true\ positives \; ; FN = false\ negatives$$

$$f1\ score = 2 \times \frac{precision \times recall}{precision + recall}$$

Visualizemos las métricas

perro

perro

perro

conejo

conejo

conejo

Predicciones

no perro

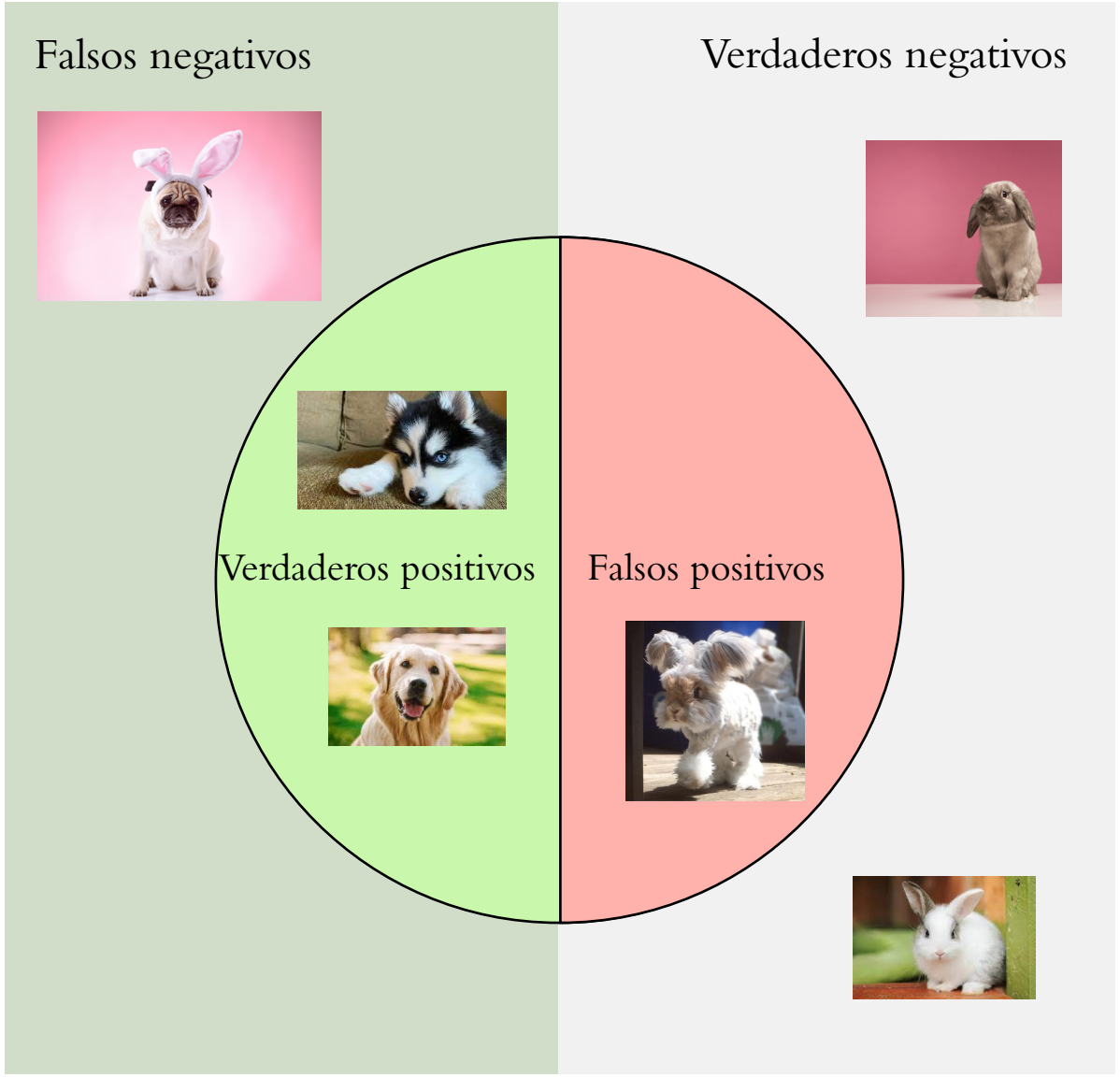
perro

perro

no perro

perro

no perro



$$accuracy = \frac{4}{6}$$

$$precision = \frac{2}{3}$$

$$recall = \frac{2}{3}$$

$$f1\ score = \frac{2}{3}$$

Entropía

Uso: nos ayuda a determinar qué feature utilizar para el split en un nodo dado.

Ejemplo:

10 registros con clase A

20 registros con clase B

30 registros con clase C

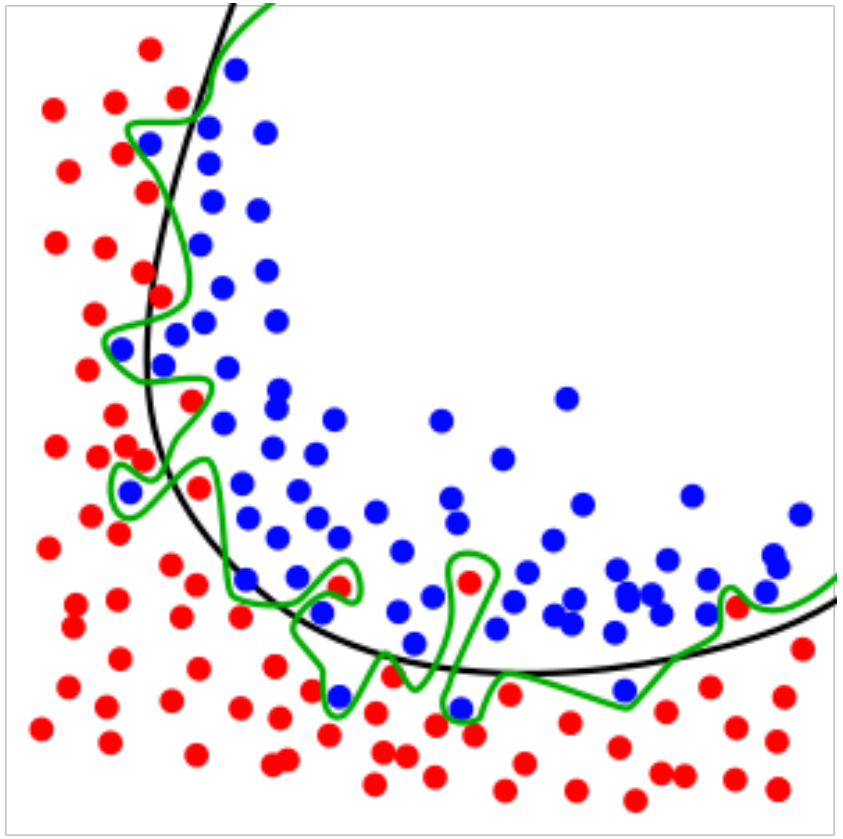
40 registros con clase D

Entropía= $-[(0.1 \log_2 0.1) + (0.2 \log_2 0.2) + (0.3 \log_2 0.3) + (0.4 \log_2 0.4)]$

Entropía = 1.85

Parte 2: Overfitting

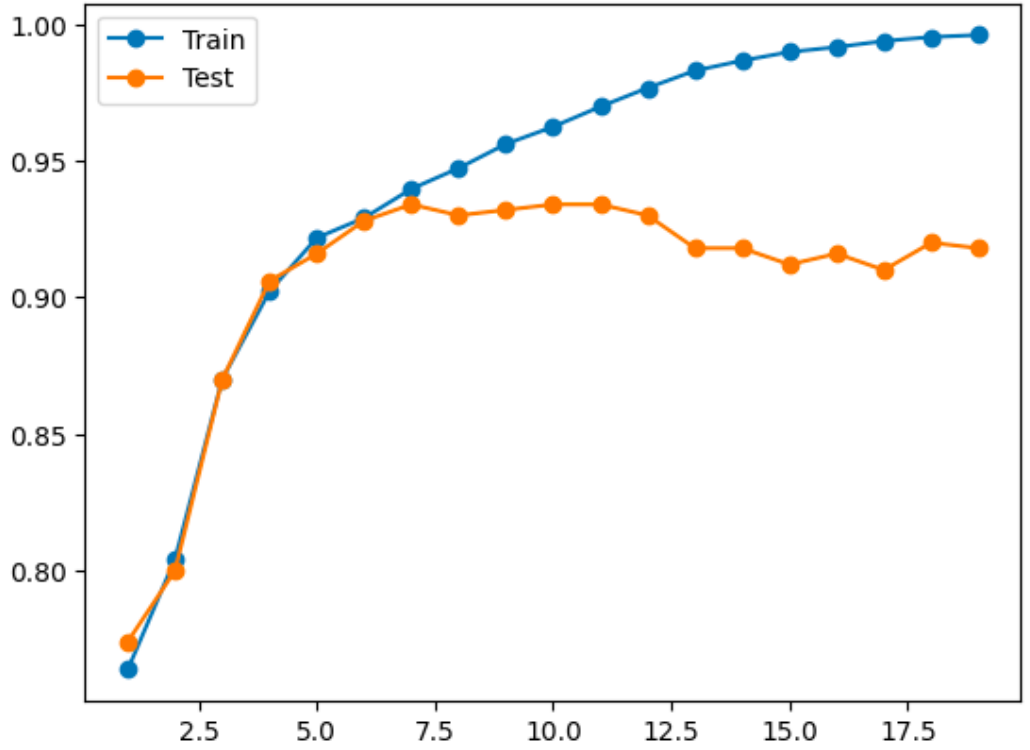
Matemáticamente, corresponde a un análisis demasiado similar a un conjunto de datos, lo cual puede causar una incapacidad de agregar nuevos datos o predecir observaciones futuras.



Vamos al Menti…

Overfitting en Árboles

En nuestro modelo, alterar la profundidad máxima del árbol puede llevar a overfitting.



¿Cómo podemos prevenirlo?

En el caso de árboles, tenemos varias opciones:

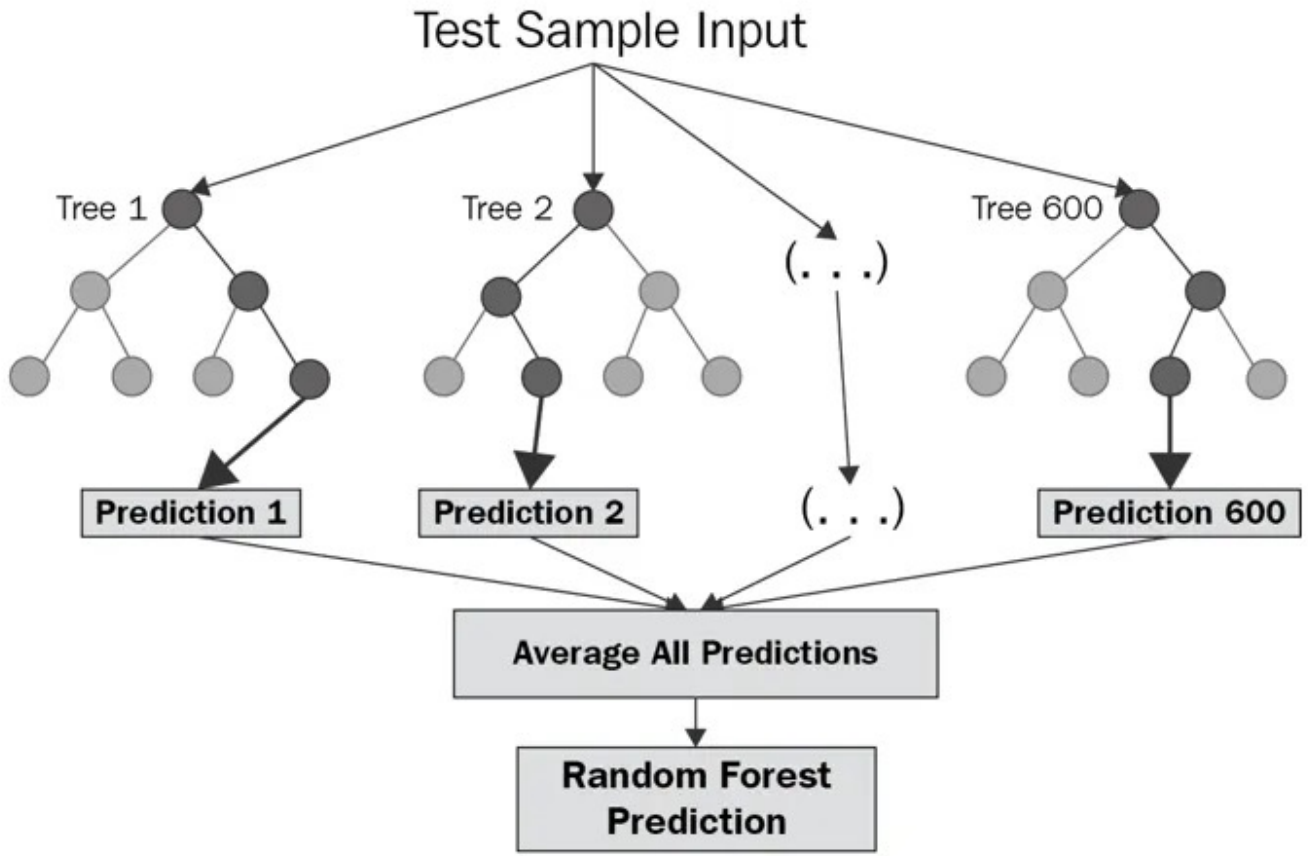
1. Disminuir la profundidad del árbol.
2. Podar nodos de acuerdo a alguna regla.
3. Limitar la cantidad de features que utilizaremos.
4. Usar un set de validación para detectar, durante el entrenamiento, cuándo conviene dejar de entrenar.

Parte 3: Random Forest

Son un tipo de ensamble, compuestos de varios árboles de decisión.

Lo más importante es que cada árbol utilizará un subconjunto aleatorio de features. Así, los árboles no se correlacionan entre sí.

Tomamos el output de todos los árboles para tomar una decisión final.



Ventajas y Desventajas

Ventajas:

- La aleatoriedad al elegir qué features utilizar produce árboles con correlación baja. Esto reduce el overfitting.
- Es fácil de determinar cuáles features son las más importantes.

Desventajas:

- Toman más tiempo para entrenar y consumen más recursos.
- Más complejos de interpretar que un árbol de decisión por sí solo.

Vamos al Menti…

¿Cómo funciona un random forest?