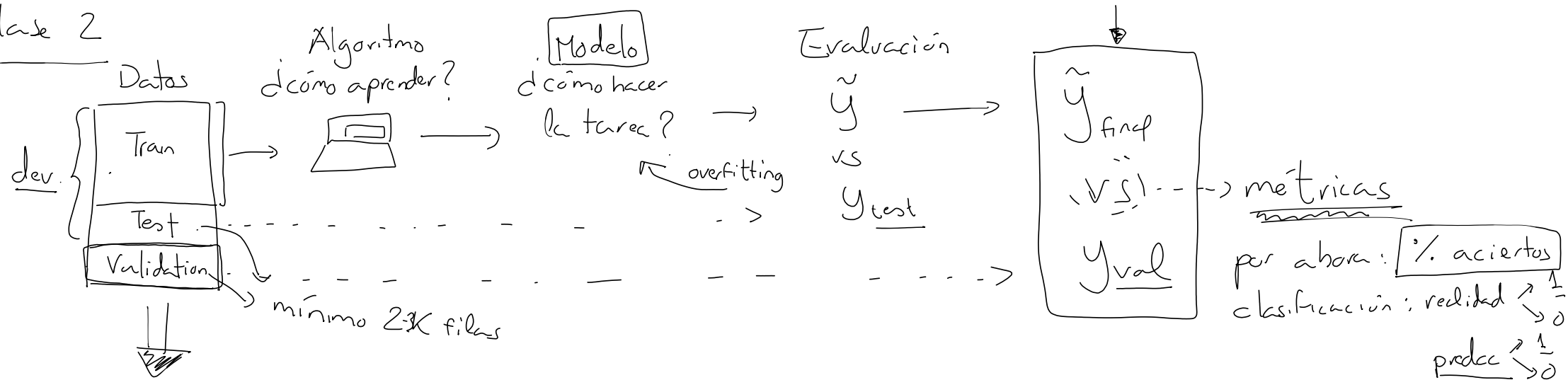


## Repaso clase 2



¿Por qué mi modelo puede ir mal?

1. Memorización (overfitting): aprende normas específicas de "Train"  
Comparar el resultado del modelo  $\text{train vs test}$  (sample aleatoria de "dev")  
↳ misma distribución
2. Distribución que ve mi máquina es distinta a la realidad  
train vs validation

Data preparation : cambiar todo a número (la "traducción" es clave para dar buena información a la máquina)

## 1. Nulos

→ según el algoritmo: decision tree (simbolistas) → valor extremo o fuera del rango

Boolean

Bool	avg(target) #	Pregunta:
True	0,5	50K
→ False	0,3	20K
→ Nulos	0,2	200

I. Nulo = True  
II. Nulo = False  
III. Nulo = 'Nueva cat'

## 2. Categóricas

3. Fechas : extraer a número todas sus componentes (5/1/2018)

- día : 5.
- mes : 1.
- año : 2018.
- semana : 1
- trimestre : 1
- año + mes + día : 20180105
- boolean : días festivos

## Aprendizaje supervisado

Dataset: Atributos:  $X_1, \dots, X_n$   $\rightarrow$  la máquina busca la mejor combinación de los atributos  
target:  $Y$   $\tilde{y} = f(X_1, \dots, X_n)$   
 $\uparrow$  combinar

¿Cómo combinar los atributos?  $\rightarrow$  lo decide la máquina con el framework algoritmos

Atributo explicativo: Según el valor del atributo  $X_n$ , reducamos la incertidumbre con respecto al target.

## Decision tree

Familia: simbolistas

Tarea: clasificación y regresión

Idea matemática: selección iterativa de atributos, de forma que consiga un subconjunto de datos más puro con respecto al target.

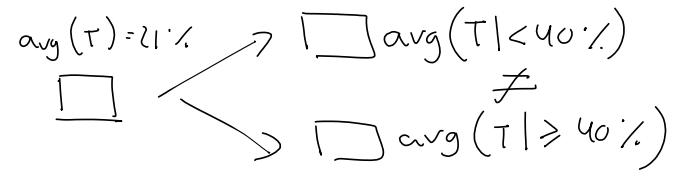
1. Iteración: escoger atributos uno a uno, secuencialmente

$\text{avg}(T) = 21\% \rightarrow \text{variable: cloud cover \%} \Rightarrow$

I. elige 1 atributo



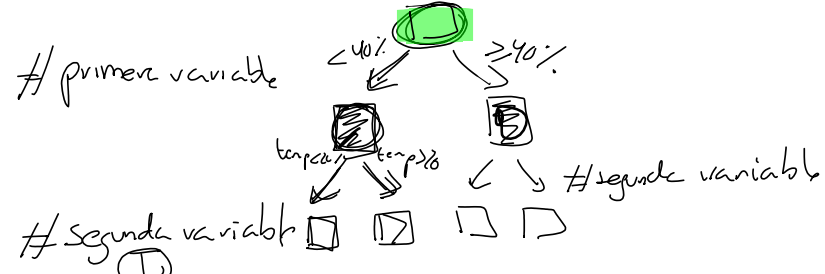
II. Decide un corte aleatorio



III. Evalúa pureza

III.1 Distintos cortes

III.2 Distintos atributos

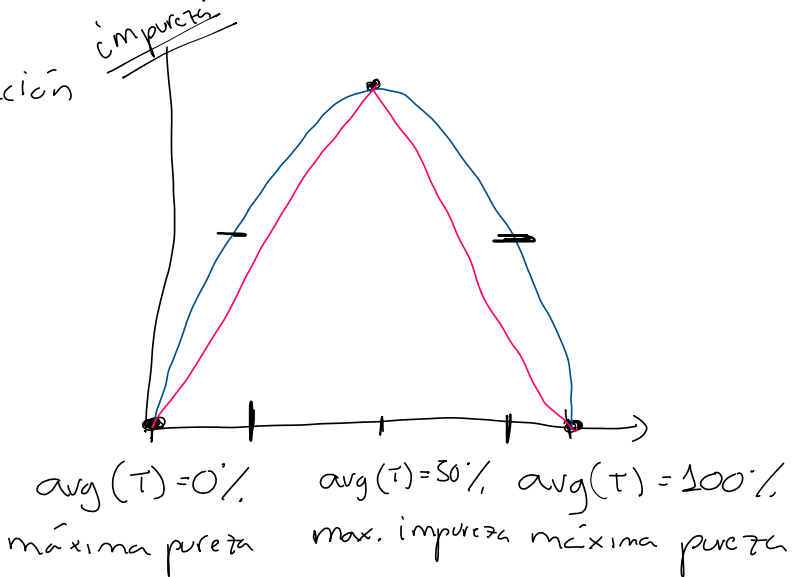


iteración

¿Hasta cuando hace preguntas?  $\rightarrow$  lo decidimos nosotros (fine-tuning)

2. Más puro: grado de parecido del subconjunto con respecto al target

En clasificación



Fórmulas de impureza

- entropía
- gini

Titanic

target muere = 1  
no muere = 0

Parametros

# preguntas: 3

# min ejemplos

Pregunta 1

sex <= 0.5  
gini = 0.4944  
samples = 834  
value = [461, 373]  
class = Died

métrica impureza gini [0, 0.5]  
número de filas  
462 mueren, 373 no mueren (55% mueren)

True sex = 0 Mujeres = 0

False sex = 1 Hombres = 1

pclass <= 2.5  
gini = 0.3025  
samples = 323  
value = [60, 263]  
class = Survived

class = 2, 2  
T

F class = 3

fare <= 26.125  
gini = 0.1274  
samples = 234  
value = [16, 218]  
class = Survived

fare <= 26.65  
gini = 0.4999  
samples = 89  
value = [44, 45]  
class = Survived

age <= 10.0  
gini = 0.3379  
samples = 511  
value = [401, 110]  
class = Died

→ 2a pregunta distinta para cada subconjunto  
→ puede ocurrir que uno de los subconjuntos resultantes tenga más impureza  
→ variables (excepto boolean) las puede reutilizar

sibsp <= 3.0  
gini = 0.375  
samples = 28  
value = [7, 21]  
class = Survived

pclass <= 1.5  
gini = 0.3006  
samples = 483  
value = [394, 89]  
class = Died

gini = 0.2449  
samples = 77  
value = [11, 66]  
class = Survived

gini = 0.0617  
samples = 157  
value = [5, 152]  
class = Survived

gini = 0.4913  
samples = 76  
value = [33, 43]  
class = Survived

gini = 0.2604  
samples = 13  
value = [11, 2]  
class = Died

gini = 0.1653  
samples = 22  
value = [2, 20]  
class = Survived

gini = 0.2778  
samples = 6  
value = [5, 1]  
class = Died

gini = 0.4474  
samples = 148  
value = [98, 50]  
class = Died

gini = 0.2057  
samples = 335  
value = [296, 39]  
class = Died

nodes terminales

1. Predicción: pred = media target en nodo terminal  $\frac{11}{77} \approx 15\%$

2. Certeza de la predicción: # samples