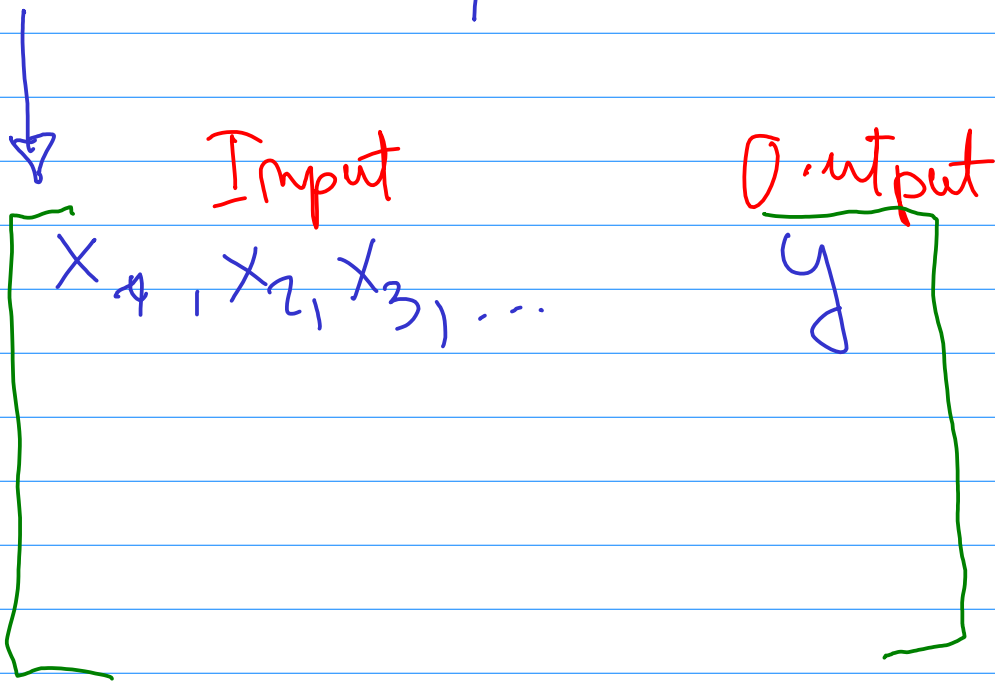
Arbori de decizie:

- Bootstrap
- Random Forest.

Date -> Predicții



Găsim $f_\theta(x_1, \ldots, x_k) \approx y$

(θ ← parametrii)

- Output valoare: o valoare din $\mathbb{R}$
- Output clasă: 0/1

Arbori de decizie:

- împărțim recursiv datele folosind un "threshold", prag pentru un input
- clasele rezultante vor fi asimilate unei singure valori/clase finale.

Ex 1:

x	1	2	3	4	5	6
y	2	2	3	8	9	9

6 singure clase

Q1 Ce valoare să dau unei clase de output-uri?

Media aritmetică: $\frac{\sum_{i=1}^{N} y_i}{N}$

Q2 Ce funcție măsoară eroarea de aproximare?

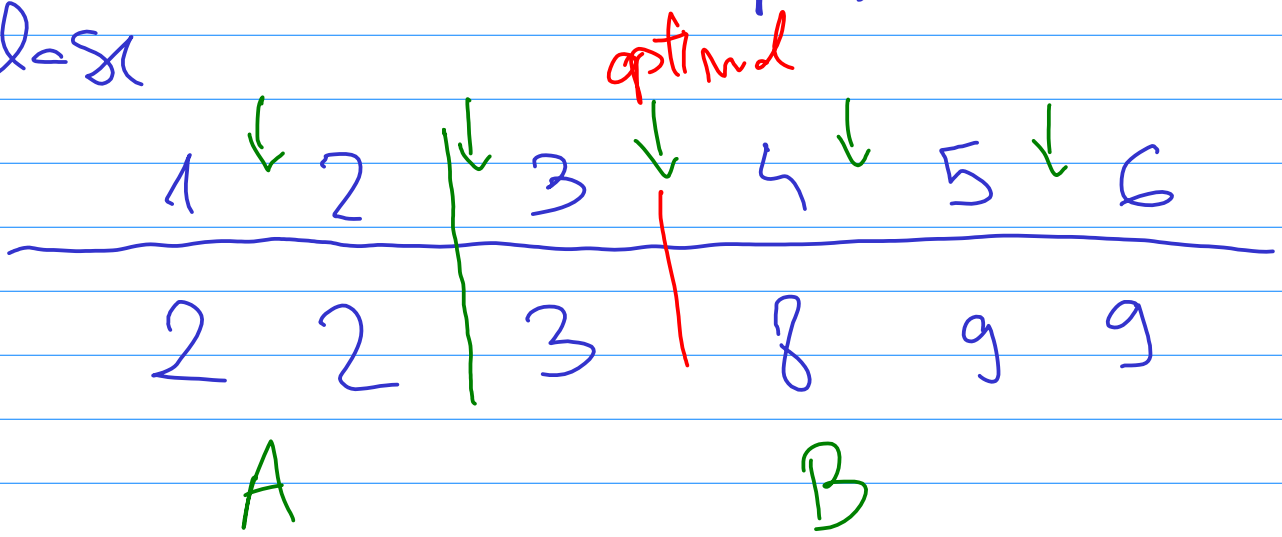
$RSS = \sum (y_i - c)^2$

c - val prezisă, media

Proprietate: $c = \text{media}(y_i)$ atunci RSS minimă

E2

Cea mai bună împărțire în 2 clase



→ Câte opțiuni de divizare în 2 clase avem? 5 cazuri aici pt 6 x_i diferiți

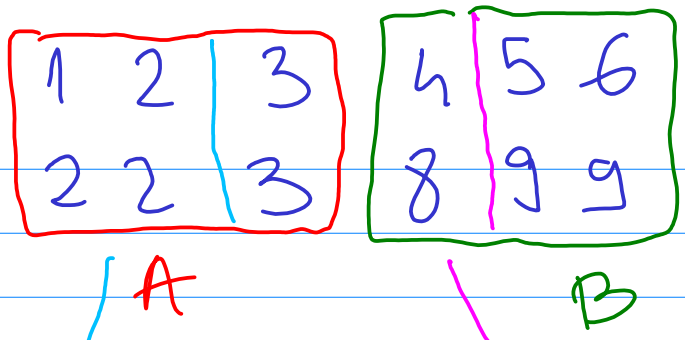
→ Care împărțire e cea mai bună?

$RSS(A) + RSS(B) \rightarrow$ minimă

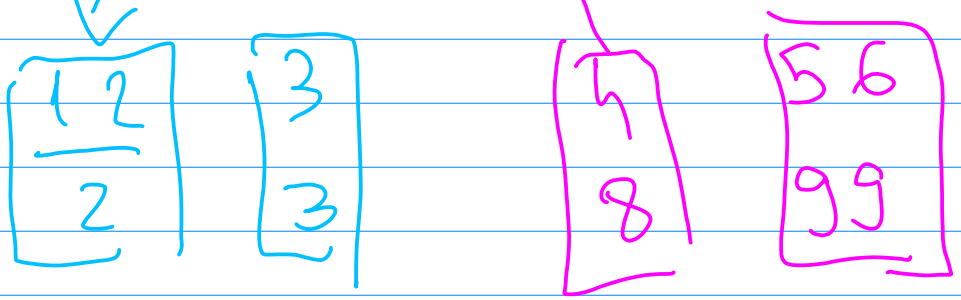
$RSS(\{y_i\}) = 0 = (y_i - \text{media}(y_i))^2$

$= 0$

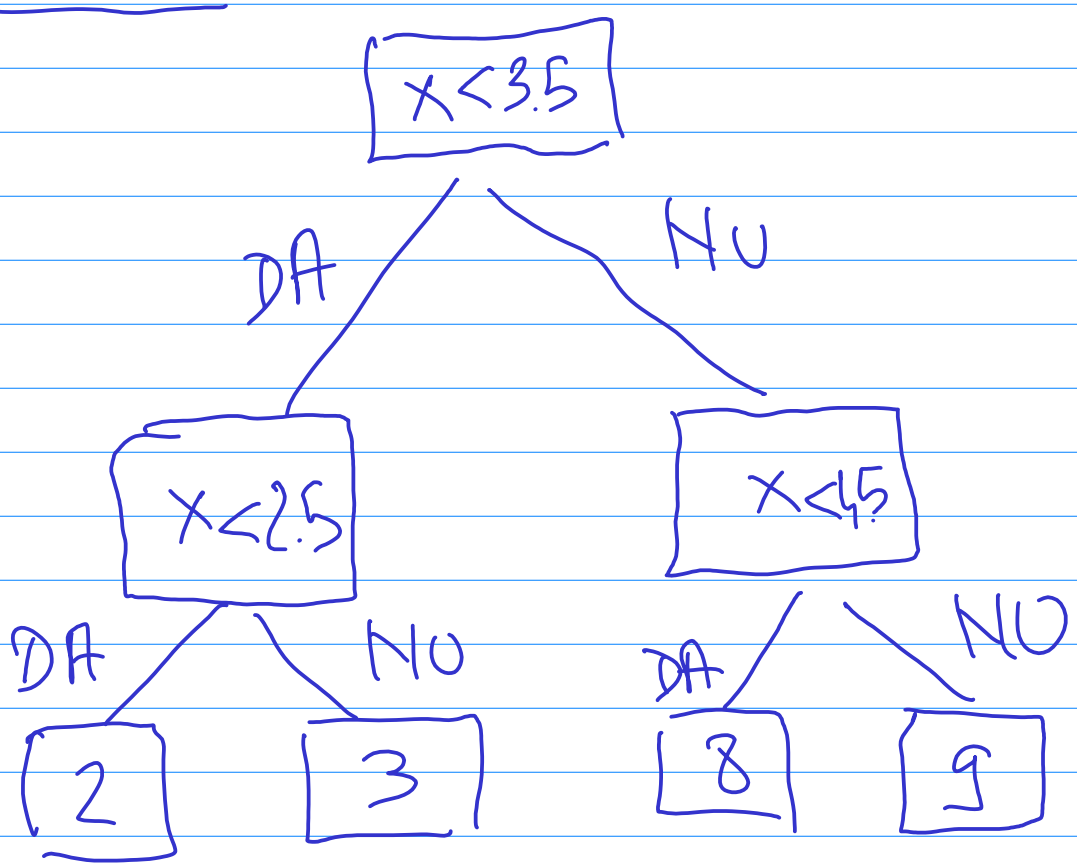
Split 1



Split 2



Arbeit:



Remarci: ① Putem aplica algoritmul recursiv până "învățăm pe de rost datele"
⇒ OVERFITTING → nu se generalizează.

② Pentru a limita overfitting-ul
- max_depth: adâncimea maximală a arborelui
- nr. minim de elemente în fiecare LEAF

Practică: nu folosim un singur arbore!

① Bootstrap and aggregation.
- construim date artificiale repetând aleatoriu elemente (random)
- pt fiecare set fităm un arbore
- combinăm predicțiile:
  - media (valori)
  - majoritatea (clase)

② Random Forest:

random

- alegem o submulțime de caracteristici
- antrenăm un arbore pe acele caracteristici
- repetăm un nr. prescris de arbori

Caracteristici					Output
x_1	x_2	x_3	x_4	x_5	y

$\text{Tree}_1: [x_1, x_2, x_3] \to y$

$\text{Tree}_2: [x_1, x_4, x_5] \to y$

⋮

$\text{Tree}_k: [x_2, x_3, x_4] \to y$

media majoritara

alegi random

Reduce similaritatea între arbori

Hyperparametri (Bagging / Rand forest)

- B = numărul arborilor.
- informații despre arborii individuali
  - max depth
  - min leaf.
- m (number of features.)

Exemplu în practică

- date sub formă tabulară (Excel, CSV)
- Obiectiv: prezicem valoarea unei proprietăți target care interesează pe client.

Exemplu:

a) Analiza de sânge → Boala X

b) Date bancare → Clientul plătește rata

Corelația x, y

$$r_{xy} = \frac{\langle x - \text{media}(x),\ y - \text{media}(y) \rangle}{|x - \text{media}(x)|\ |y - \text{media}(y)|}$$

Cauchy - Schwarz

$$\boxed{-1 \leq r_{xy} \leq 1}$$

Train test split, stratify

	0	1	
Train	90%	10%	acuși proporție
Valid	90%	10%	din ficare clasă
Test	90%	10%	în Train/Valid/Test

Threshold t

probabilitate

Predict.(X) = 0,55

$t = 0.5$ $\quad P(X) = 0.55 \rightarrow$ clasa 1

$t = 0.6$ $\quad P(X) = 0.55 \rightarrow$ clasa 0

→ + ridicat : acceptăm mai greu un element în clasa 1.
→ mai puțini detecții false

→ + ridicat → riscăm să detectăm
→ mai puțini probleme.

Confusion matrix

y clasa, $\hat{y}$ predicția

	$\hat{y}=1$	$\hat{y}=0$	
$y=1$	TP	FN	→ nu plătesc
$y=0$	FP	TN	→ plătesc

↓ modelul prezice că nu plătesc

↘ modelul prezice că plătesc

prec: $\frac{TP}{TP+FP}$ = indicator de pozitive false.

! recall: $\frac{TP}{TP+FN}$ = procentul de rău platnici care au fost detectați

Cost nu-detecții: 1000
Cost prezicere falsă de neplată: 100