# Arbori de decizie, Random Forest

→ procesarea datelor (TRAIN / VALID / TEST)

→ regresia liniară → învățăm o valoare

↘ învățăm o clasă

Input $x_i$ Output-uri $y_i$

Obținem o funcție ai. $f(x_i) \approx y_i$

Reg. liniară: funcție afină $ax_i + b$

↙ coef. ↘ intercept

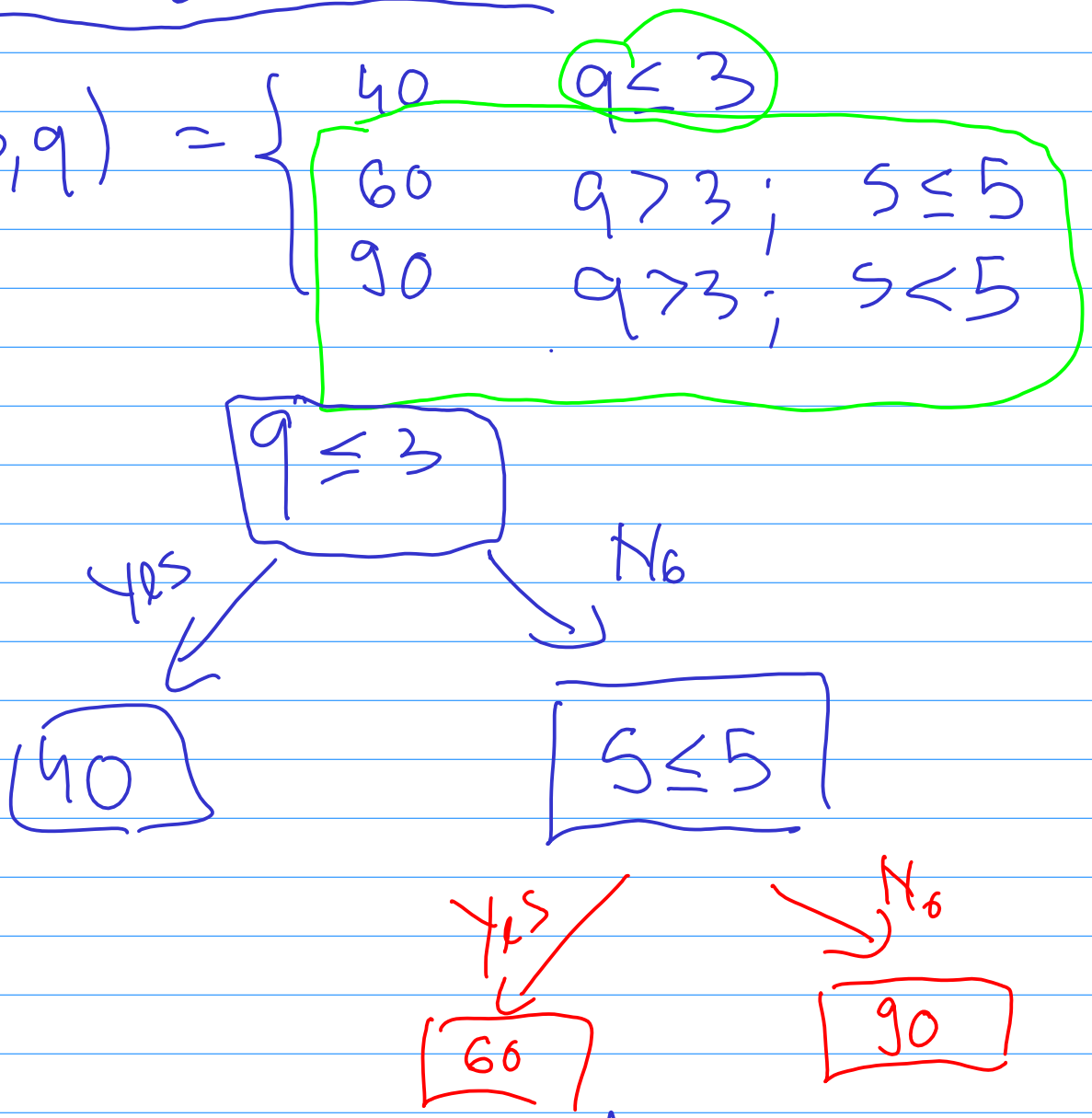
Regresia Logistică: Învățăm o ineg de forma $a_1x_1 + a_2x_2 - b <= 0$ | · Const

$a_1, a_2, b$ nu sunt unice

Notiunea 2: Prezicem dacă clientul plătește rata la bancă.

Arbori de decizie

$$f(s,q) = \begin{cases} 40 & q \le 3 \\ 60 & q > 3;\ s \le 5 \\ 90 & q > 3;\ s < 5 \end{cases}$$



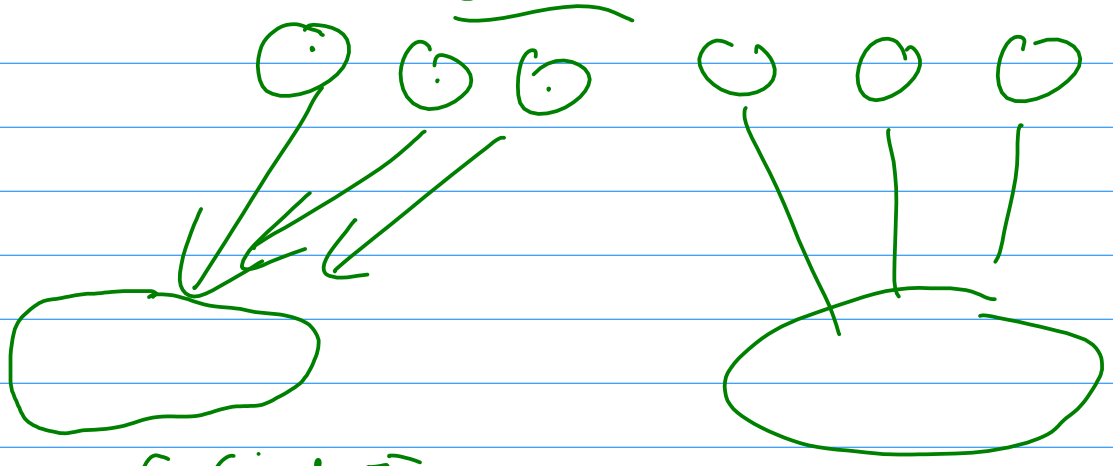

→ Care parametru e relevant

→ cum alegem nivelul la care se face bifurcația

→ care este profunzimea arborelui?

Putem prezice valori

clase



O singură valoare prezisă.

→ Valoarea prezisă va fi media valorilor din training!

→ minimizează $c \longmapsto \sum (y_i - c)^2$

$$c = \frac{1}{N} \sum_{i=1}^{N} y_i$$

$$RSS(A) = \sum_{i \in A} (y_i - \bar{y}_A)^2$$

| $x_i$ | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| $y_i$ | 2 | 3 | 2 | 8 | 9 | 10 |

Fără split: $\bar{y} = \frac{34}{6}$

$$RSS = \sum (y_i - \bar{y})^2 = \frac{208}{3} \approx 70$$

Split: Treshold $t$:

$$A_L = \{ x_i \leq t \} \qquad A_R = \{ x_i \geq t \}$$

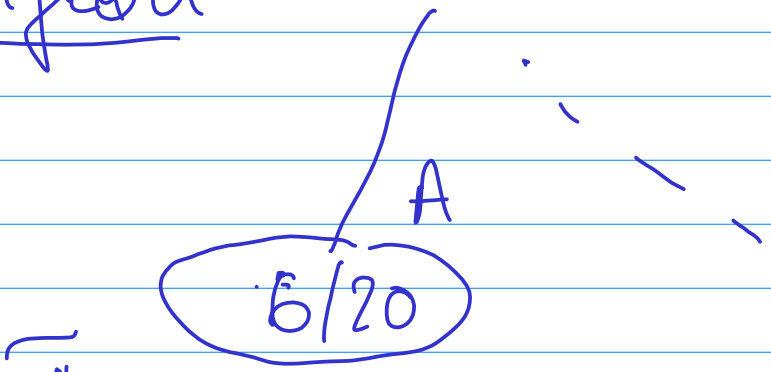
| $x_i$ | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| $y_i$ | 2 | 3 | 2 | 8 | 9 | 10 |

$A_L$, $\bar{y} = 2{,}5$ $\qquad$ $A_R$, $\bar{y} = \frac{29}{4}$

$RSS(A_L) + RSS(A_R) = 39{,}5$

importă care produce cea mai mică eroare totală RSS

# Clasificare




$\pi_A = 0,3$

Decidem dacă clasa este 1 sau 0

→ alegem un prag t ptr a

predicția $\hat{y} = \begin{cases} 1 & \pi_A \geq t \\ 0 & \pi_A < t \end{cases}$

Trebuie să alegem o funcție I

| A | $A_L$ | $A_R$ |
|---|---|---|

Care este câștigul din împărțirea făcută

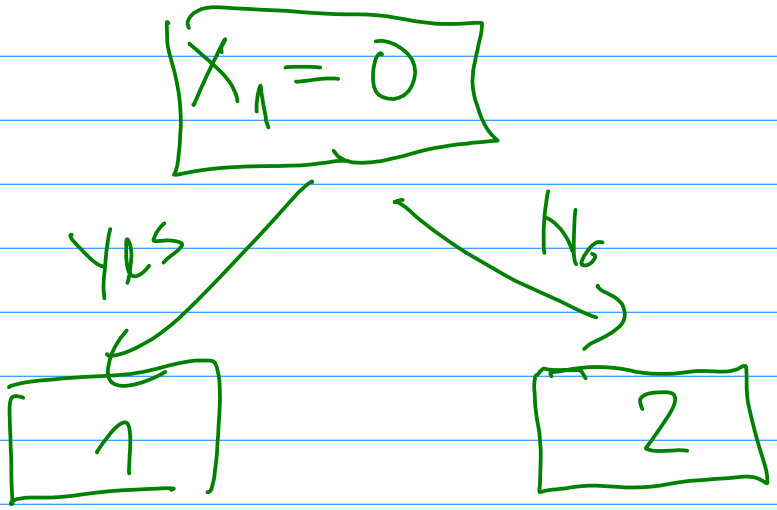
$$\text{gain} = I(A) - \frac{n_L}{n_A} I(A_L) - \frac{n_R}{n_A} I(A_R)$$

Un arbore fără restricții poate învăța datele pe de rost: overfitting

→ nu se va generaliza bine!

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 0 | 1 | 1 |
| 1 | 2 | 2 |
| 0 | 0 | 1 |

Overfitting: modelul e exact pe date




Parametrii cu limitează complexitatea

- max_depth: adâncimea maximă
- min_samples_leaf: fiecare clasă finală ar trebui să conțină un nr. minim de puncte

din baza de date.

- Un singur arbore riscă să producă rezultate cu mai să generalizeze

$$D = \{x_1, x_2, x_3, x_4, x_5, x_6\}$$

Bagging:

1. $D_1 = \{x_1, x_1, x_3, x_3, x_5, x_6\}$
   → antrenament → $A_1$
2. $D_2 = \{x_2, x_2, x_3, x_4, x_5, x_6\}$
   → antrenament → $A_2$

→ scade varianța rezultatului dacă arborii nu sunt corelați!