

# ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΤΥΠΩΝ (PATTERN RECOGNITION)

**Σέργιος Θεοδωρίδης**  
**Κωνσταντίνος Κουτρούμπας**

## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

Πρόκειται για μια οικογένεια μη γραμμικών ταξινομητών. Είναι συστήματα απόφασης **πολλών σταδίων (multistage)**, όπου οι κλάσεις απορρίπτονται **διαδοχικά (sequentially)**, έως ότου φτάσουμε σε μια κλάση που θα είναι τελικά αποδεκτή. Για το λόγο αυτό:

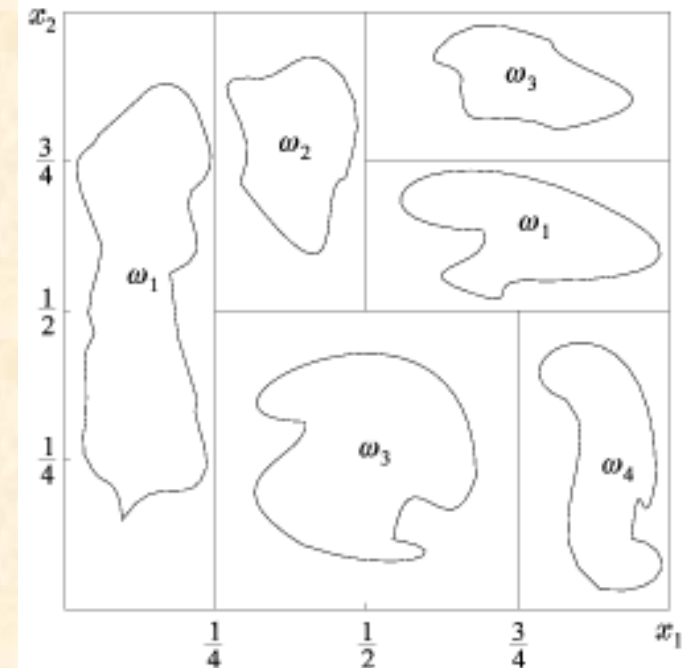
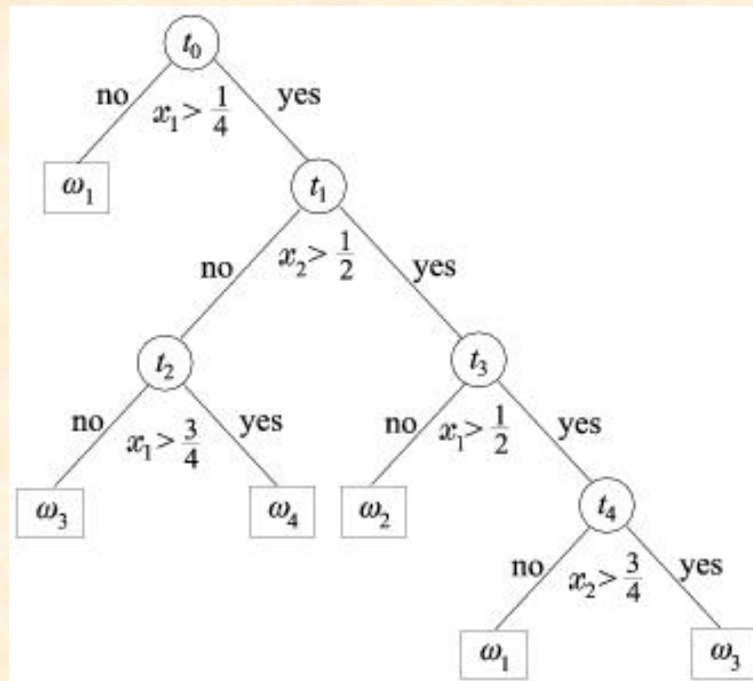
- Ο χώρος των χαρακτηριστικών **τεμαχίζεται σε μοναδικές περιοχές με έναν ακολουθιακό τρόπο.**
- Με την άφιξη ενός διανύσματος χαρακτηριστικών, λαμβάνονται διαδοχικές αποφάσεις καταχώρησης χαρακτηριστικών σε συγκεκριμένες περιοχές, ακολουθώντας ένα μονοπάτι **κόμβων (nodes)** ενός κατάλληλα κατασκευασμένου **δένδρου**.
- Η ακολουθία των αποφάσεων εφαρμόζεται (συνήθως) σε μεμονωμένα χαρακτηριστικά και οι ερωτήσεις που εξετάζονται σε κάθε κόμβο είναι του **τύπου:**

είναι το χαρακτηριστικό  $x_i \leq a$

όπου  $a$  είναι ένα προεπιλεγμένο κατώφλι (επιλέγεται κατά τη διάρκεια της εκπαίδευσης).

## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

- Τα παρακάτω σχήματα αντιστοιχούν σε τέτοιο παράδειγμα. Τα δένδρα αυτού του τύπου είναι γνωστά ως **Συνήθη Δυαδικά Δένδρα Ταξινόμησης (Ordinary Binary Classification Trees (OBCT))**. Τα υπερπέπιεδα απόφασης που διαιρούν το χώρο σε περιοχές, είναι παράλληλα στους άξονες του χώρου των δειγμάτων. Άλλοι τύποι διαίρεσης του χώρου είναι επίσης δυνατοί, παρότι είναι λιγότερο δημοφιλείς.



## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

- Σχεδιαστικά στοιχεία που ορίζουν ένα δένδρο απόφασης.
  - Κάθε κόμβος,  $t$ , αντιστοιχεί σε ένα υποσύνολο  $X_t \subseteq X$ , όπου  $X$  είναι το σύνολο εκπαίδευσης. Σε κάθε κόμβο, το αντίστοιχο σύνολο,  $X_t$  διαιρείται σε δύο (δυαδική διαίρεση) ξένα μεταξύ τους υποσύνολα-απογόνους  $X_{t,Y}$  and  $X_{t,N}$ , ώστε

$$X_{t,Y} \cap X_{t,N} = \emptyset$$

$$X_{t,Y} \cup X_{t,N} = X_t$$

$X_{t,Y}$  είναι το υποσύνολο του  $X_t$  για το οποίο η απάντηση στην ερώτηση του κόμβου  $t$  είναι **ΝΑΙ**.  $X_{t,N}$  είναι το υποσύνολο που αντιστοιχεί στο **ΟΧΙ**. Η διαίρεση αποφασίζεται με βάση την ερώτηση (**query**) που υιοθετείται.

Στόχος είναι το καθένα από τα σύνολα που θα προκύψουν,  $X_{t,Y}$  και  $X_{t,N}$ , να είναι «καθαρότερα» (να περιέχουν δηλ. το καθένα σημεία από λιγότερες κλάσεις) σε σχέση με το  $X_t$ .

## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

Προαπαιτούμενα για τη δημιουργία ενός δένδρου απόφασης

- **Κριτήριο διαμερισμού:** καθορίζει τη **βέλτιστη** δυνατή διαμέριση του  $X_t$  στα  $X_{t,Y}$  και  $X_{t,N}$ .
- **Κριτήριο τερματισμού-διαμέρισης (stop-splitting):** ελέγχει την ανάπτυξη του δένδρου έως τους **τερματικούς κόμβους (φύλλα – leafs)**.
- **Κανόνας καταχώρησης τερματικού κόμβου** σε κατηγορία.

## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

- **Σύνολο ερωτήσεων:** Στα δένδρα OBCT το σύνολο των ερωτήσεων είναι του τύπου

$$\text{είναι } x_i \leq a;$$

Η επιλογή του συγκεκριμένου χαρακτηριστικού  $x_i$  και της τιμής του κατωφλίου  $a$ , για κάθε κόμβο  $t$ , είναι το αποτέλεσμα αναζήτησης, κατά την εκπαίδευση, ανάμεσα στα χαρακτηριστικά και ένα σύνολο από δυνατές τιμές κατωφλίου. Ο τελικός συνδυασμός είναι αυτός που οδηγεί στη **βέλτιστη τιμή** ενός κατάλληλου κριτηρίου.

## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

- **Κριτήριο διαμερισμού:** Η κύρια ιδέα πίσω από τη διαίρεση σε κάθε κόμβο είναι τα υποσύνολα-απόγονοι  $X_{t,Y}$  και  $X_{t,N}$  που θα προκύψουν να παρουσιάζουν μεγαλύτερο βαθμό **ομογενοποίησης ως προς τις κλάσεις**, σε σχέση με αυτόν του  $X_t$ . Έτσι το κριτήριο που θα επιλεγεί θα πρέπει να είναι σε συμφωνία με αυτόν το στόχο. Ένα συχνά χρησιμοποιούμενο κριτήριο είναι ο **βαθμός μη-καθαρότητας ενός κόμβου (node impurity)**:

$$I(t) = -\sum_{i=1}^M P(\omega_i | t) \log_2 P(\omega_i | t)$$

και

$$P(\omega_i | t) \approx \frac{N_t^i}{N_t}$$

όπου  $N_t^i$  είναι ο αριθμός των στοιχείων του συνόλου  $X_t$ , τα οποία ανήκουν στην κλάση  $\omega_i$ . Η **μείωση της μη-καθαρότητας ενός κόμβου (decrease in node impurity)** ορίζεται ως:

$$\Delta I(t) = I(t) - \frac{N_{t,Y}}{N_t} I(t_Y) - \frac{N_{t,N}}{N_t} I(t_N)$$

## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

### Κριτήριο διαμερισμού (συν.):

- Ο στόχος είναι να επιλεγούν σε κάθε κόμβο εκείνοι οι παράμετροι (**χαρακτηριστικό** και **κατώφλι**), που οδηγούν σε μία διαίρεση, η οποία παρουσιάζει τη **μεγαλύτερη δυνατή μείωση της μη-καθαρότητας**.
- Γιατί η μεγαλύτερη δυνατή μείωση; Παρατηρείστε ότι η μέγιστη τιμή για την  $I(t)$  λαμβάνεται όταν όλες οι κλάσεις είναι **ισοπίθανες**, δηλ. όταν το  $X_t$  παρουσιάζει τον **ελάχιστο δυνατό βαθμό** ομογενοποίησης.



## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

**Παράδειγμα:** Έστω ένα παράδειγμα δύο κλάσεων με τα ακόλουθα πέντε σημεία:  
 $(1,10) - \omega_1, (2,7) - \omega_2, (3,6) - \omega_1, (4,8) - \omega_2, (5,9) - \omega_2$ .

Για το σύνολο αυτό είναι:  $P(\omega_1) = 2/5 = 0.4$  και  $P(\omega_2) = 3/5 = 0.6$ .

Άρα η εντροπία του είναι  $I = - ( P(\omega_1) \log_2 P(\omega_1) + P(\omega_2) \log_2 P(\omega_2) ) = 0.9710$ .

**Υπολογισμός της μείωσης της εντροπίας για την 1<sup>η</sup> συνιστώσα και τιμή 1 ( $x_1 \leq 1$ ):**

•  $PY(\omega_1) = 1/1 = 1$  και  $PY(\omega_2) = 0/1 = 0 \Rightarrow IY = - (PY(\omega_1) \log_2 PY(\omega_1) + PY(\omega_2) \log_2 PY(\omega_2)) = 0$ .

•  $PN(\omega_1) = 1/4 = 0.25$  και  $PN(\omega_2) = 3/4 = 0.75$

$\Rightarrow IN = - (PN(\omega_1) \log_2 PN(\omega_1) + PN(\omega_2) \log_2 PN(\omega_2)) = 0.8113$

•  $\Delta I = I - (1/5)IY - (4/5)IN = 0.9710 - 0.2*0 - 0.8*0.8113 = 0.3219$ .

Εργαζόμενοι ομοίως για τις υπόλοιπες περιπτώσεις έχουμε:

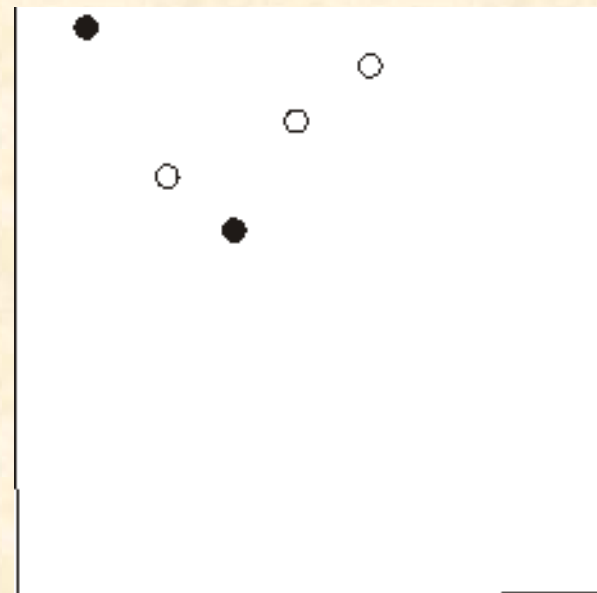
$x_1$	1	2	3	4	5
$\Delta I$	0.32	0.02	<b>0.42</b>	0.17	0

$x_2$	6	7	8	9	10
$\Delta I$	0	0.02	0.32	0.02	0.32

Η μέγιστη μείωση στην εντροπία επιτυγχάνεται για το 1<sup>ο</sup> χαρακτηριστικό και τιμή κατωφλίου 3.

Άρα ο κανόνας για τον πρώτο κόμβο είναι:  $x_1 \leq 3$



## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

- **Κανόνας τερματισμού διαμέρισης.** Υιοθέτησε ένα κατώφλι  $T$  και σταμάτα την περαιτέρω διαίρεση ενός κόμβου (δηλ. καταχώρησέ τον σαν **φύλλο-τερματικό κόμβο**), αν η μείωση της μη-καθαρότητας είναι μικρότερη από  $T$ . Δηλ. όταν ο κόμβος  $t$  είναι “**αρκετά καθαρός**”.
- **Κανόνας αντιστοίχισης σε κλάση:** Καταχώρησε ένα φύλλο στην κλάση  $\omega_j$ , για την οποία:  
$$j = \arg \max_i P(\omega_i | t)$$

➤ Summary of an OBCT algorithmic scheme:

- Begin with the root node, i.e.,  $X_t = X$
- For each new node  $t$ 
  - \* For every feature  $x_k, k = 1, 2, \dots, l$ 
    - For every value  $\alpha_{kn}, n = 1, 2, \dots, N_{tk}$ 
      - Generate  $X_{tY}$  and  $X_{tN}$  according to the answer in the question: is  $x_k(i) \leq \alpha_{kn}, i = 1, 2, \dots, N_t$
      - Compute the impurity decrease
      - End
      - Choose  $\alpha_{kn_0}$  leading to the maximum decrease w.r. to  $x_k$
    - \* End
    - \* Choose  $x_{k_0}$  and associated  $\alpha_{k_0n_0}$  leading to the overall maximum decrease of impurity
    - \* If stop-splitting rule is met declare node  $t$  as a leaf and designate it with a class label
    - \* If not, generate two descendant nodes  $t_Y$  and  $t_N$  with associated subsets  $X_{tY}$  and  $X_{tN}$ , depending on the answer to the question: is  $x_{k_0} \leq \alpha_{k_0n_0}$
  - End

## ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΔΕΝΔΡΑ ΑΠΟΦΑΣΗΣ

### Παρατηρήσεις:

- Ένας κρίσιμος παράγοντας κατά τη φάση σχεδιασμού είναι το μέγεθος του δένδρου. Συνήθως το δένδρο αναπτύσσεται έως ότου φτάσει σε μεγάλο μέγεθος και στη συνέχεια εφαρμόζονται διάφορες τεχνικές **κλαδέματος (pruning)**.
- Τα δένδρα απόφασης ανήκουν στην κατηγορία των **ασταθών (unstable)** ταξινομητών. Αυτό μπορεί να αντιμετωπιστεί με τεχνικές «μέσου όρου» (“averaging” techniques). Π.χ. χρησιμοποιώντας τεχνικές **bootstrapping** στο  $X$ , κατασκευάζονται διάφορα δένδρα,  $T_i$ ,  $i=1, 2, \dots, B$ . Η απόφαση ταξινόμησης λαμβάνεται με βάση έναν **κανόνα πλειοψηφίας (majority voting)**.

# Example 1: A Simple Tree

Consider the following  $n = 16$  points in two dimensions for training a binary CART tree ( $B = 2$ ) using the entropy impurity (Eq. 1).

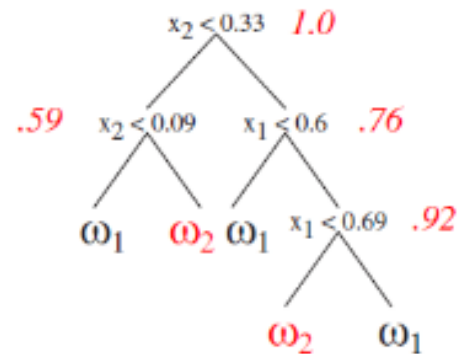
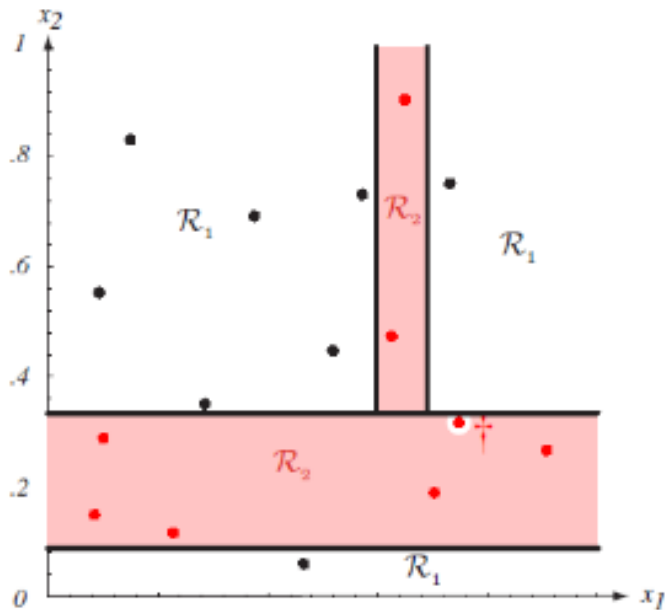
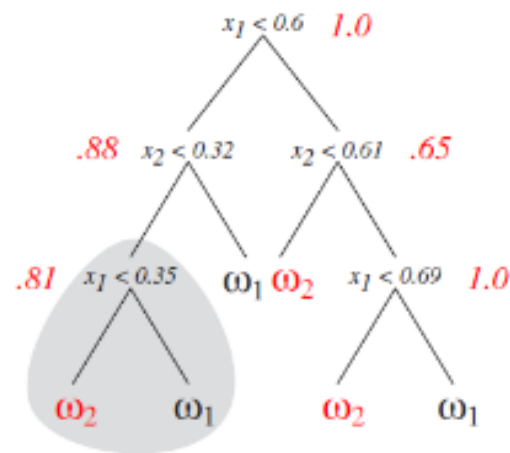
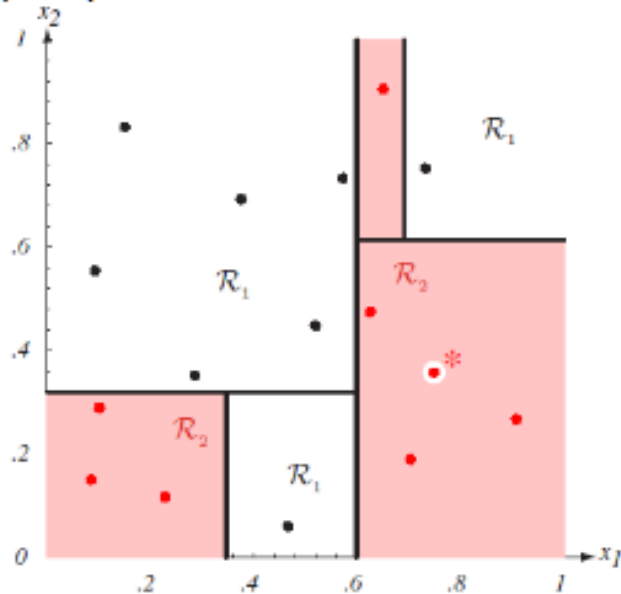
$$i(N) = - \sum_j P(\omega_j) \log_2 P(\omega_j), \quad (1)$$

$\omega_1$ (black)		$\omega_2$ (red)	
$x_1$	$x_2$	$x_1$	$x_2$
.15	.83	.10	.29
.09	.55	.08	.15
.29	.35	.23	.16
.38	.70	.70	.19
.52	.48	.62	.47
.57	.73	.91	.27
.73	.75	.65	.90
.47	.06	.75	.36* (.32 <sup>†</sup> )

Example taken from <http://www.cse.msu.edu/~cse802/DecisionTrees.pdf>

# Example 1. Simple Tree

Entropy impurity at nonterminal nodes is shown in red and impurity at each leaf node is 0



**Instability or sensitivity of tree to training points; alteration of a single point leads to a very different tree; due to discrete & greedy nature of CART**

## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

Η βασική φιλοσοφία πίσω από το συνδυασμό διαφορετικών ταξινομητών βασίζεται στο γεγονός ότι ακόμα και ο «καλύτερος» ταξινομητής αποτυγχάνει σε μερικά διανύσματα όπου άλλοι ταξινομητές μπορεί να δώσουν σωστή ταξινόμηση. Ο συνδυασμός ταξινομητών σκοπεύει στην εκμετάλλευση αυτής της **συμπληρωματικής πληροφορίας** (**complementary information**) που δίνεται από διάφορους ταξινομητές.

Έτσι κάποιος σχεδιάζει διαφορετικούς βέλτιστους ταξινομητές και στη συνέχεια συνδυάζει τα αποτελέσματα με βάση ένα συγκεκριμένο κανόνα.

➤ Έστω ότι καθένας από τους,  $L$  ταξινομητές που σχεδιάστηκαν δίνει στην έξοδό του τις εκ των υστέρων πιθανότητες

$$P(\omega_i | \underline{x}), i = 1, 2, \dots, M$$

## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

- **Κανόνας γινομένου:** Καταχώρησε το  $x$  στην κλάση  $\omega_i$  :

$$i = \arg \max_k \prod_{j=1}^L P_j(\omega_k | \underline{x})$$

όπου  $P_j(\omega_k | \underline{x})$  είναι η αντίστοιχη εκ των υστέρων πιθανότητα του  $j^{\text{th}}$  ταξινομητή.

- **Κανόνας άθροισης:** Καταχώρησε το  $x$  στην κλάση  $\omega_i$  :

$$i = \arg \max_k \sum_{j=1}^L P_j(\omega_k | \underline{x})$$



## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

**Παράδειγμα:** Έστω ένα πρόβλημα **3** κλάσεων,  $\omega_1, \omega_2, \omega_3$ , με  $L=4$  ταξινομητές. Σχηματικά η ταξινόμηση ενός διανύσματος  $\mathbf{x}$  σε μια από τις τρεις κλάσεις γίνεται ως εξής:

	$\omega_1$	$\omega_2$	$\omega_3$	
Ταξ. 1	$P_1(\omega_1 \mathbf{x})$	$P_1(\omega_2 \mathbf{x})$	$P_1(\omega_3 \mathbf{x})$	
Ταξ. 2	$P_2(\omega_1 \mathbf{x})$	$P_2(\omega_2 \mathbf{x})$	$P_2(\omega_3 \mathbf{x})$	
Ταξ. 3	$P_3(\omega_1 \mathbf{x})$	$P_3(\omega_2 \mathbf{x})$	$P_3(\omega_3 \mathbf{x})$	
Ταξ. 4	$P_4(\omega_1 \mathbf{x})$	$P_4(\omega_2 \mathbf{x})$	$P_4(\omega_3 \mathbf{x})$	
Γιν.	$\prod_j P_j(\omega_1 \mathbf{x})$	$\prod_j P_j(\omega_2 \mathbf{x})$	$\prod_j P_j(\omega_3 \mathbf{x})$	$\longrightarrow i: \operatorname{argmax}_{k=1,2,3} \prod_j P_j(\omega_k \mathbf{x})$
Άθρ.	$\sum_j P_j(\omega_1 \mathbf{x})$	$\sum_j P_j(\omega_2 \mathbf{x})$	$\sum_j P_j(\omega_3 \mathbf{x})$	$\longrightarrow i: \operatorname{argmax}_{k=1,2,3} \sum_j P_j(\omega_k \mathbf{x})$

## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

- **Κανόνας πλειοψηφίας:** Καταχώρησε το  $x$  στην κλάση για την οποία υπάρχει ομοφωνία ή όταν τουλάχιστον  $I_c$  από τους ταξινομητές συμφωνούν στην κλάση

$$I_c = \begin{cases} \frac{L}{2} + 1, & L \text{ even} \\ \frac{L+1}{2}, & L \text{ odd} \end{cases}$$

Διαφορετικά η απόφαση είναι **απόρριψη (rejection)**, δηλ. δεν λαμβάνεται **καμία απόφαση**.

Έτσι σωστή απόφαση λαμβάνεται όταν η πλειοψηφία των ταξινομητών συμφωνεί με τη σωστή κατηγορία και λάθος όταν η πλειοψηφία συμφωνεί με μία λάθος κατηγορία.

## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

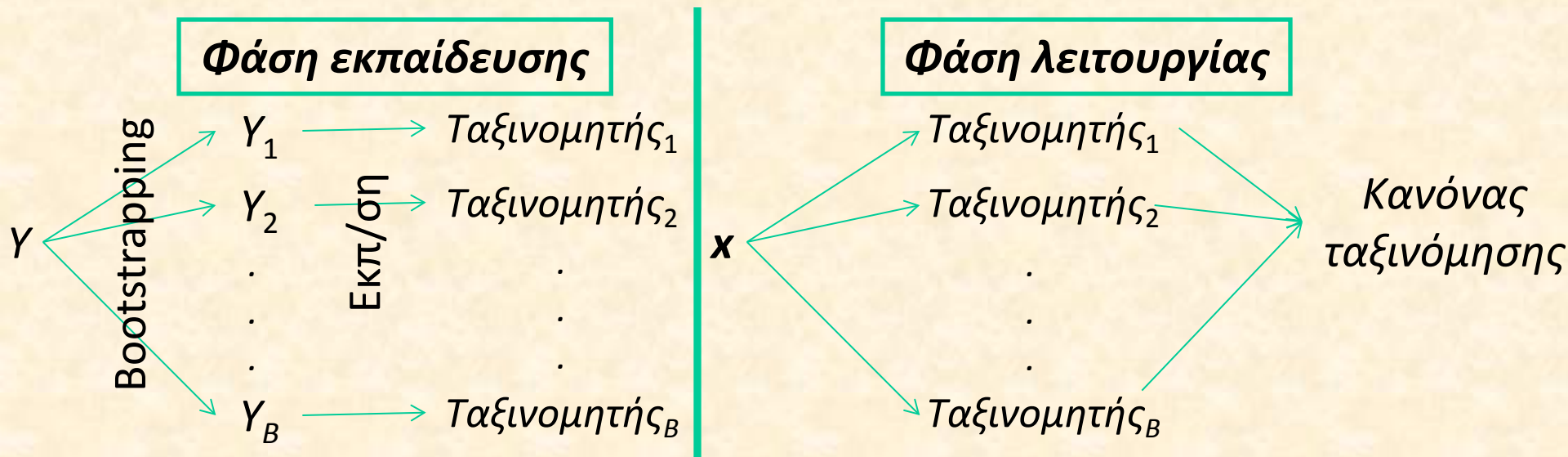
### Εξαρτημένοι ή ανεξάρτητοι ταξινομητές;

- Παρότι δεν υπάρχουν γενικά θεωρητικά αποτελέσματα, το πείραμα έδειξε ότι όσο πιο ανεξάρτητοι είναι οι ταξινομητές ως προς τις αποφάσεις τους, τόσο πιο πολύ αναμένεται η λήψη βελτιωμένων αποτελεσμάτων μετά το συνδυασμό των επιμέρους αποφάσεων. Ωστόσο, **δεν υπάρχει εγγύηση** ότι ο συνδυασμός ταξινομητών οδηγεί σε **καλύτερη** απόδοση συγκρινόμενος με την απόδοση του **“καλύτερου”** ανάμεσα στους ταξινομητές.

## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

Προς την ανεξαρτησία: Πιθανά σενάρια

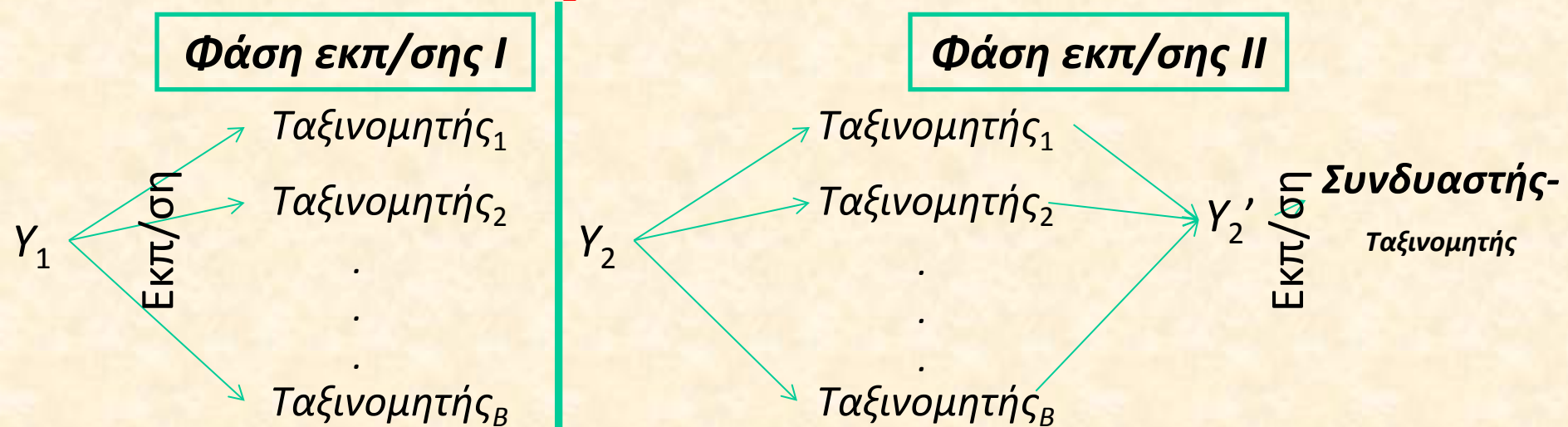
- **Bootstrapping:** Πρόκειται για μια δημοφιλή τεχνική **συνδυασμού ασταθών ταξινομητών**, όπως τα δένδρα απόφασης. Πιο συγκεκριμένα,
  - ✓ Παράγονται αρκετά **σύνολα δεδομένων** μέσω **ομοιόμορφης δειγματοληψίας (uniform sampling)** με **επανατοποθέτηση** από το διαθέσιμο σύνολο δεδομένων.
  - ✓ Με βάση κάθε σύνολο δεδομένων παράγεται και ένας ταξινομητής.
  - ✓ Η **απόφαση καταχώρησης δεδομένου διανύσματος σε μια κλάση** προκύπτει από το **συνδυασμό** (π.χ. με χρήση του κανόνα πλειοψηφίας) **των αποφάσεων** των μεμονομένων ταξινομητών.



## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

### Προς την ανεξαρτησία: Πιθανά σενάρια

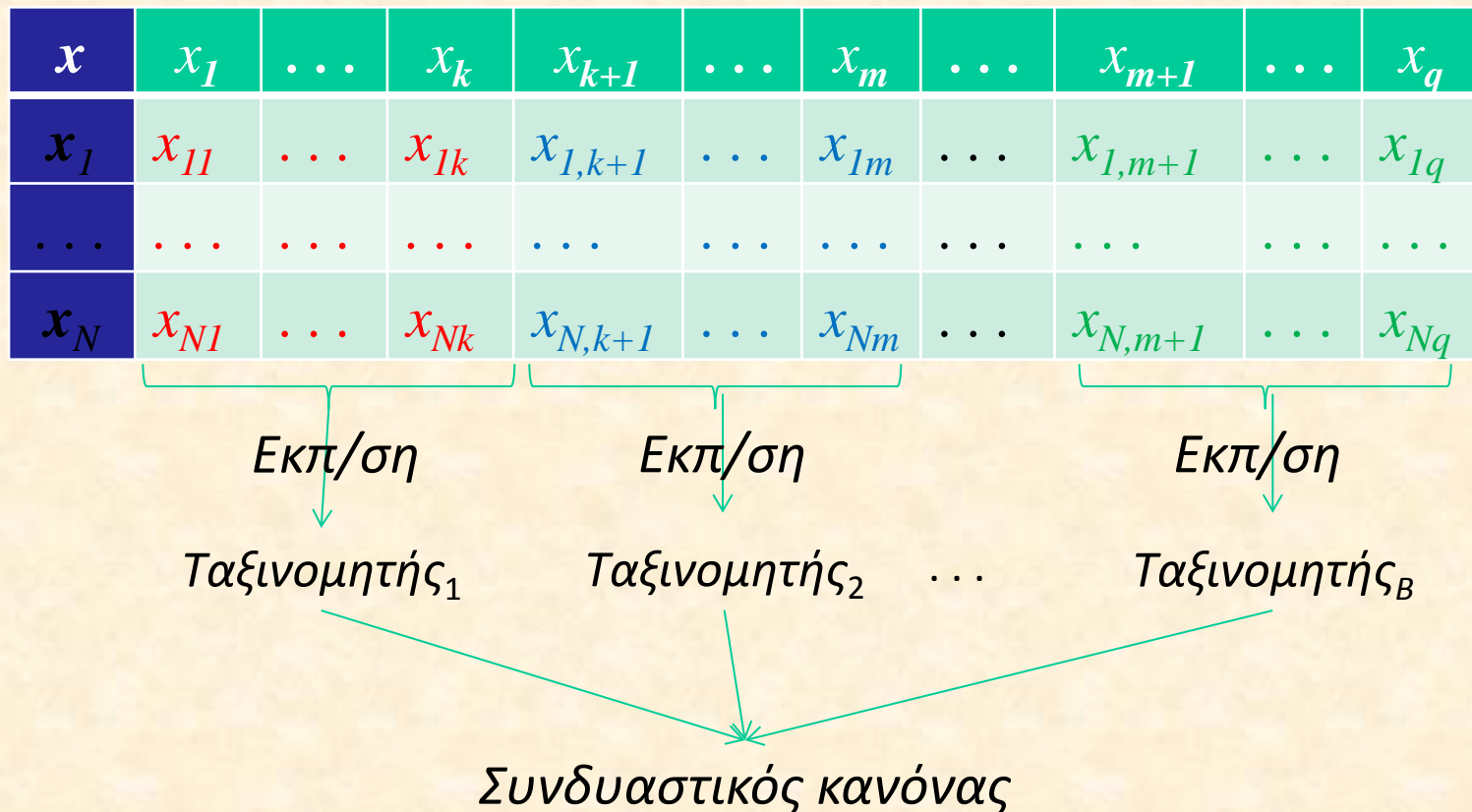
- **Stacking:** Εδώ χρησιμοποιείται ένας συνδυαστής-ταξινομητής που χρειάζεται και αυτός εκπαίδευση. Πιο συγκεκριμένα έχουμε την ακόλουθη διαδικασία:
  - ✓ **Διαίρεσε** το διαθέσιμο σύνολο δεδομένων  $Y$  σε δύο μέρη,  $Y_1$  και  $Y_2$ .
  - ✓ **Εκπαίδευσε**  $B$  μεμονωμένους **ταξινομητές** χρησιμοποιώντας το  $Y_1$ .
  - ✓ **Εφάρμοσε** κάθε διάνυσμα του  $Y_2$  στους  $B$  ταξινομητές και **σχημάτισε** το  $B$ -διάστατο διάνυσμα με τις εξόδους των ταξινομητών. (Έτσι, κάθε διάνυσμα του  $Y_2$  αναπαρίσταται τώρα από το παραπάνω  $B$ -διάστατο διάνυσμα). Έστω  $Y_2'$  το νέο σύνολο.
  - ✓ Χρησιμοποίησε το  $Y_2'$  για την εκπαίδευση του **συνδυαστή-ταξινομητή**.



## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

Προς την ανεξαρτησία: Πιθανά σενάρια

- Χρήση διαφορετικών υποχώρων για την εκπαίδευση μεμονωμένων ταξινομητών: Εδώ, κάθε μεμονωμένος ταξινομητής εκπαιδεύεται σε διαφορετικό υπόχωρο του χώρου χαρακτηριστικών. Δηλαδή, κάθε ταξινομητής εκπαιδεύεται χρησιμοποιώντας διαφορετικά χαρακτηριστικά των εμπλεκόμενων οντοτήτων.



## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

### Παρατηρήσεις:

- **Random forests:** Εδώ ένα σύνολο  $B$  ταξινομητών δένδρων απόφασης δημιουργούνται με τέτοιο τρόπο ώστε αυτοί να είναι όσο το δυνατόν πιο ανεξάρτητοι. Δοθέντος ενός συνόλου  $N$   $l$ -διάστατων διανυσμάτων δεδομένων,  $X$ , έχουμε:

#### Φάση δημιουργίας ταξινομητών (δένδρων απόφασης):

- Δημιούργησε  $B$  σύνολα δεδομένων,  $X_1, \dots, X_B$ , μεγέθους  $N$  το καθένα μέσω τυχαίας δειγματοληψίας με επανατοποθέτηση από το  $X$ .
- Με βάση τα  $X_1, \dots, X_B$ , δημιούργησε αντίστοιχα δένδρα απόφασης,  $T_1, \dots, T_B$ . Για τον καθορισμό του κανόνα ενός κόμβου σε καθένα από αυτά, εξέτασε μόνο  $\lambda (\ll l)$  χαρακτηριστικά, τυχαία επιλεγμένα, από το σύνολο των χαρακτηριστικών.
- Δεν υπάρχει διαδικασία «κλαδέματος».

#### Φάση λειτουργίας:

Για δεδομένο  $x$ :

- Ταξινόμησέ το με βάση τα  $T_1, \dots, T_B$ .
- Συνδύασε τα παραπάνω αποτελέσματα, ώστε να προκύψει η τελική ταξινόμηση για το  $x$ .

## ΣΥΝΔΥΑΖΟΝΤΑΣ ΤΑΞΙΝΟΜΗΤΕΣ

### Παρατηρήσεις:

- Οι κανόνες **πλειοψηφίας** και **αθροίσματος** συγκαταλέγονται ανάμεσα στα **πιο δημοφιλή** συνδυαστικά σχήματα.
- Η **εκπαίδευση** των μεμονωμένων ταξινομητών σε **διαφορετικούς υποχώρους** του χώρου των χαρακτηριστικών φαίνεται να οδηγεί σε **σημαντικά καλύτερα αποτελέσματα** σε σχέση με την περίπτωση όπου οι ταξινομητές εκπαιδεύονται στον ίδιο υπόχωρο.
- Εκτός από τους τρεις παραπάνω κανόνες, μπορούν να υιοθετηθούν και άλλοι όπως η τιμή της **διαμέσου (Median value)** των εξόδων των μεμονωμένων ταξινομητών.
- **Boosting προσέγγιση**: Ένας «**ισχυρός**» ταξινομητής δημιουργείται με τη διαδοχική προσθήκη «**ασθενών**» (weak) ταξινομητών, οι οποίοι εκπαιδεύονται δίνοντας έμφαση στα σημεία που απέτυχε να ταξινομήσει σωστά ο συνδυασμός των προηγούμενων ασθενών ταξινομητών.