MEDLAB
MEDICAL TECHNOLOGY & INTELLIGENT INFORMATION SYSTEMS

Δέντρα Απόφασης (Decision Trees)



- Διδάσκων: Δημήτριος Ι. Φωτιάδης
- Τμήμα Μηχανικών Επιστήμης Υλικών
- Website: https://medlab.cc.uoi.gr/

- Ιωάννινα 2026 - 2027

Περιεχόμενα



1. Η βασική ιδέα των δέντρων απόφασης
2. Η εντροπία και τα κριτήρια διάσπασης: ID3, C4.5, CART
3. Από τον αλγόριθμο στο δέντρο του scikit-learn
4. Παράδειγμα βήμα-βήμα με δεδομένα χάλυβα
5. Εφαρμογές στην Επιστήμη Υλικών
6. Υπερπροσαρμογή και πρόωρο κλάδεμα
7. Κλάδεμα μετά την ανάπτυξη του δέντρου: CCP, REP, PEP, MEP

Ανακεφαλαίωση

Ανακεφαλαίωση · Μάθημα 3

Γραμμικά μοντέλα: από τα δεδομένα στην πρόβλεψη σε τέσσερα βήματα

1 · Δεδομένα
πίνακας X (δείγματα × χαρακτηριστικά) και ετικέτες y

→

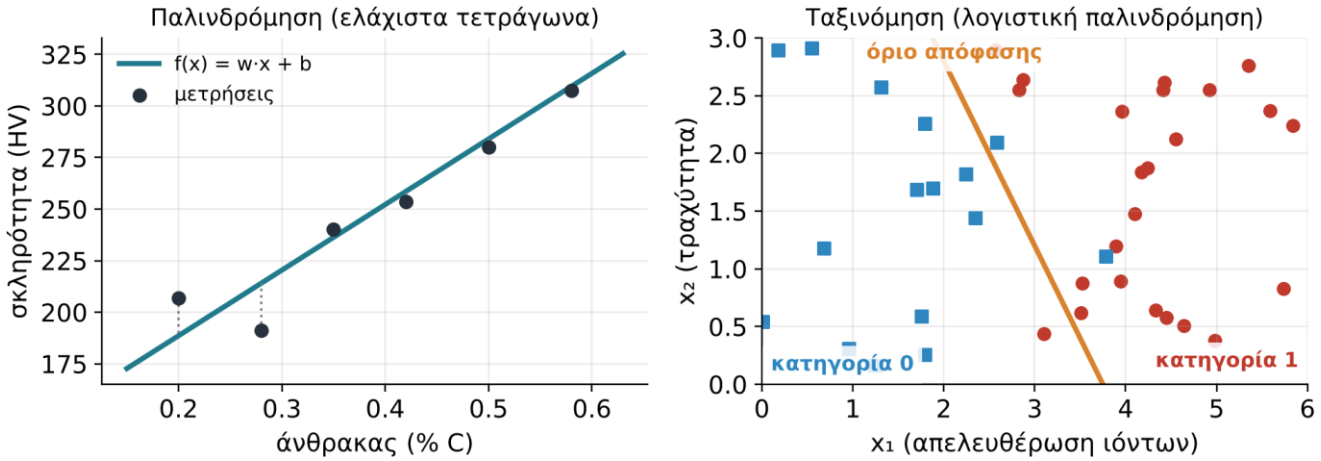
2 · Μοντέλο
f(x) = w·x + b · κάθε w δείχνει την επίδραση ενός χαρακτηριστικού

→

3 · Εκπαίδευση
ελαχιστοποίηση απώλειας · ποινή λ: Ridge (L2), Lasso (L1)

→

4 · Έξοδος
αριθμός (παλινδρόμηση) ή πιθανότητα → κατηγορία (ταξινόμηση)



Τι δείχνει το σχήμα: αριστερά, μια ευθεία που περνά «ανάμεσα» από μετρήσεις σκληρότητας· οι κουκκίδες γραμμές είναι τα υπόλοιπα που ελαχιστοποιούμε. Δεξιά, δύο κατηγορίες υλικών χωρίζονται από ευθεία (το όριο απόφασης). Συνθετικά δεδομένα.

ΚΟΙΝΟΣ ΠΥΡΗΝΑΣ

Όλα τα μοντέλα του Μαθήματος 3 έχουν τον ίδιο πυρήνα, τον γραμμικό συνδυασμό

$$w \cdot x + b.$$

Αλλάζει μόνο η απώλεια (ελάχιστα τετράγωνα ή cross-entropy), η ποινή (λ) ή η συνάρτηση στην έξοδο (σιγμοειδής, Softmax).

ΤΟ ΟΡΙΟ ΤΟΥΣ

Το όριο απόφασης είναι πάντα ευθεία ή επίπεδο. Για καμπύλες σχέσεις πρέπει να φτιάξουμε νέα χαρακτηριστικά (π.χ. x^2) ή να χρησιμοποιήσουμε πυρήνες.

Επισκόπηση του μαθήματος

1 · Βασική λογική

Τι θα μάθουμε σήμερα και με ποια σειρά

Ορισμός

Τα δέντρα απόφασης (decision trees) είναι οικογένεια αλγορίθμων που χρησιμοποιούν δενδρική δομή για να μιμηθούν τη διαδικασία λήψης αποφάσεων του ανθρώπου.

Βασικές αρχές
εκπαίδευση & πρόβλεψη

→

Κριτήρια
εντροπία, διαχωρισμός

→

Information Gain, Gain Ratio, Gini Index
κλασικοί αλγόριθμοι

→

Υπερπροσαρμογή
βασικό πρόβλημα

→

Κλάδεμα (Pruning)
πριν & μετά την ανάπτυξη

→

Πρακτική εφαρμογή
scikit-learn, Iris

Το κεφάλαιο προχωρά με λογική σειρά: πώς χτίζεται και πώς χρησιμοποιείται ένα δέντρο → με ποιο κριτήριο επιλέγεται κάθε διάσπαση → ποιοι αλγόριθμοι προκύπτουν από τα διαφορετικά κριτήρια → ποιο είναι το βασικό τους πρόβλημα (overfitting) → πώς το αντιμετωπίζουμε (pruning) → εφαρμογή σε Python.

Στο τέλος του μαθήματος θα μπορείτε να:

- εξηγείτε τη λογική των βασικών κριτηρίων διάσπασης, όπως η εντροπία, το information gain, το gain ratio και ο Gini index
- εξηγείτε γιατί οι Information Gain, Gain Ratio, και Gini Index μπορούν να επιλέγουν διαφορετικό χαρακτηριστικό
- αναγνωρίζετε την υπερπροσαρμογή σε ένα δέντρο και να εξηγείτε πώς το pruning βοηθά στον έλεγχο της πολυπλοκότητας

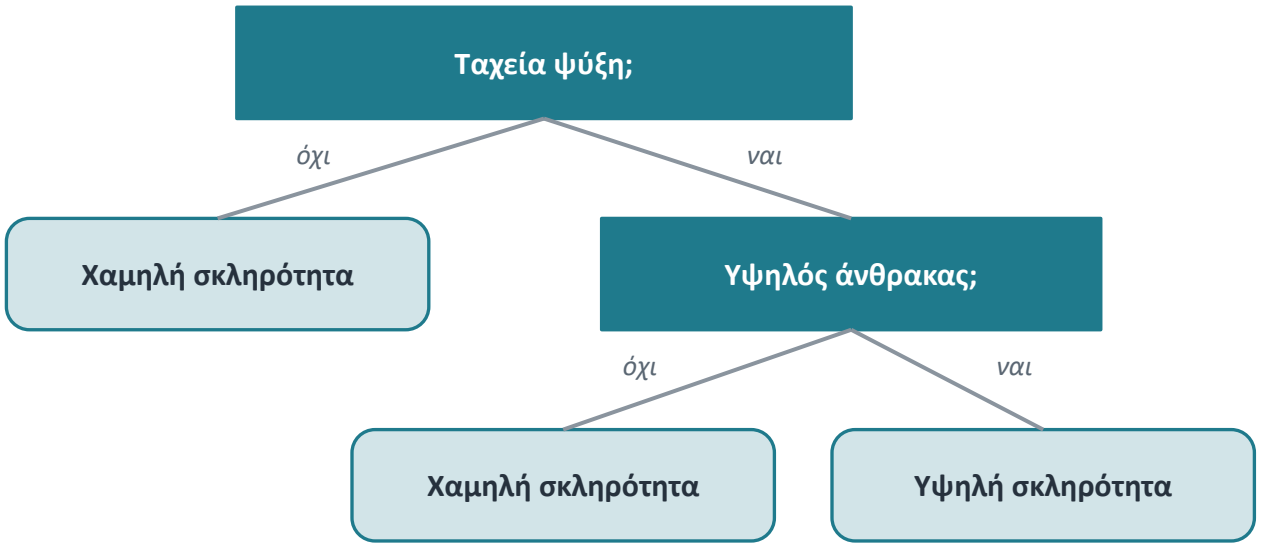
Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Η βασική ιδέα: μια σειρά από ερωτήσεις

1 · Βασική λογική

Το δέντρο δομεί αποφάσεις και τις πιθανές συνέπειές τους

Διαίσθηση: πώς σκέφτεται ένας μηχανικός



Κάθε εσωτερικός κόμβος (inner node) είναι μια ερώτηση για ένα χαρακτηριστικό (attribute), κάθε κλάδος (branch) αντιστοιχεί σε μία πιθανή απάντηση και κάθε φύλλο (leaf node) σε μία τελική απόφαση. Ένα δείγμα ξεκινά από τη ρίζα (root node) και ακολουθεί τις διαδοχικές απαντήσεις μέχρι να καταλήξει σε ένα φύλλο.

Δενδρικό μοντέλο

Δομεί αποφάσεις και τις πιθανές συνέπειές τους, ώστε να βοηθά στη λήψη απόφασης. Οι κανόνες δεν γράφονται από εμάς· μαθαίνονται από τα δεδομένα.

Σταδιακή διάσπαση

Η εκπαίδευση (classification ή regression) είναι σταδιακή διάσπαση (splitting) των δειγμάτων, ξεκινώντας από τη ρίζα.

Επιλογή attribute

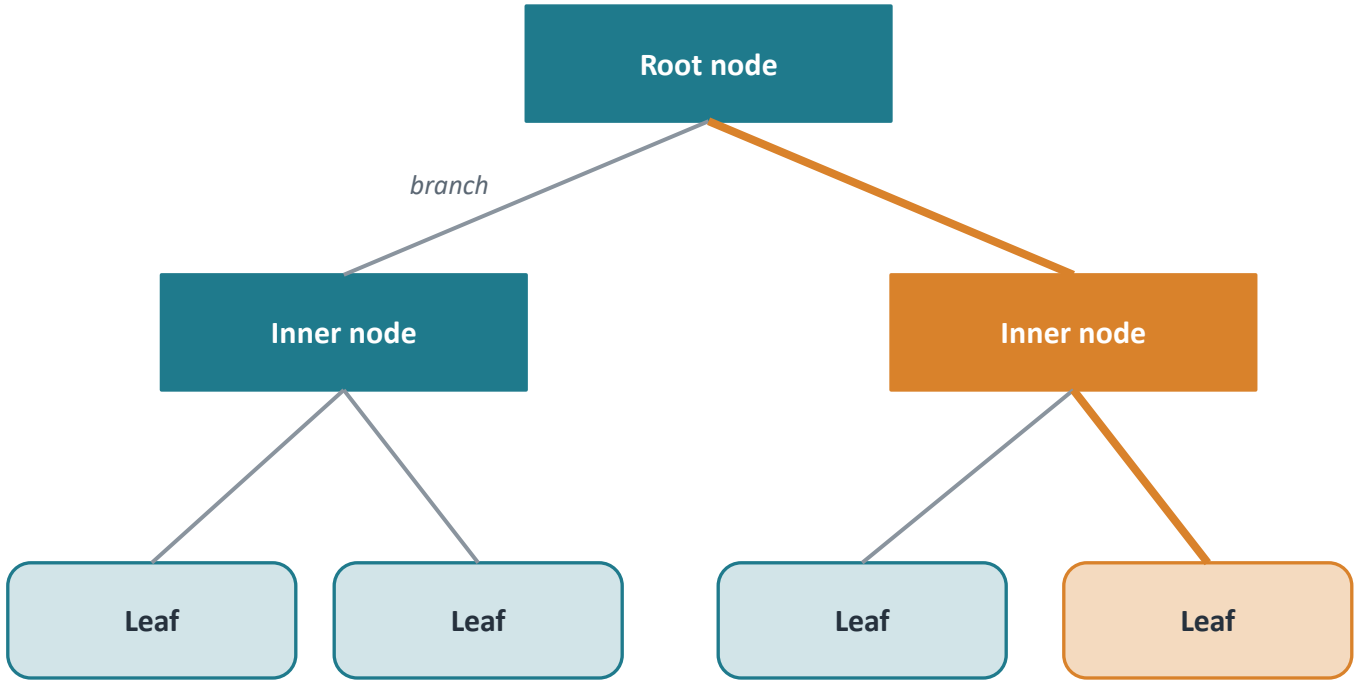
Κάθε διάσπαση γίνεται με ένα attribute, που επιλέγεται για υψηλότερη καθαρότητα (purity). Άλλος κανόνας επιλογής → άλλος αλγόριθμος.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Τα μέρη ενός δέντρου απόφασης

1 · Βασική λογική

Μία ρίζα, πολλά εσωτερικά nodes, αρκετά φύλλα



Root	Ξεκινά με όλα τα δείγματα εκπαίδευσης
Inner	Υποσύνολο που διασπάται ξανά με νέο attribute
Branch	Μία τιμή του attribute → ένα υποσύνολο
Leaf	Δεν διασπάται άλλο· φέρει την «απόφαση», π.χ. ετικέτα κλάσης

Ένα path από τη ρίζα σε φύλλο. Κάθε path είναι μια σειρά συνθηκών που οδηγεί σε μία απόφαση, γι' αυτό το δέντρο διαβάζεται ως σύνολο κανόνων «αν… τότε…». Κάθε node περιέχει υποσύνολο των δειγμάτων του γονέα του.

ΠΡΟΣΟΧΗ

Τα attributes «καταναλώνονται»: ένα attribute που χρησιμοποιήθηκε σε ένα node δεν ξαναεπιλέγεται στη συνέχεια του ίδιου path.

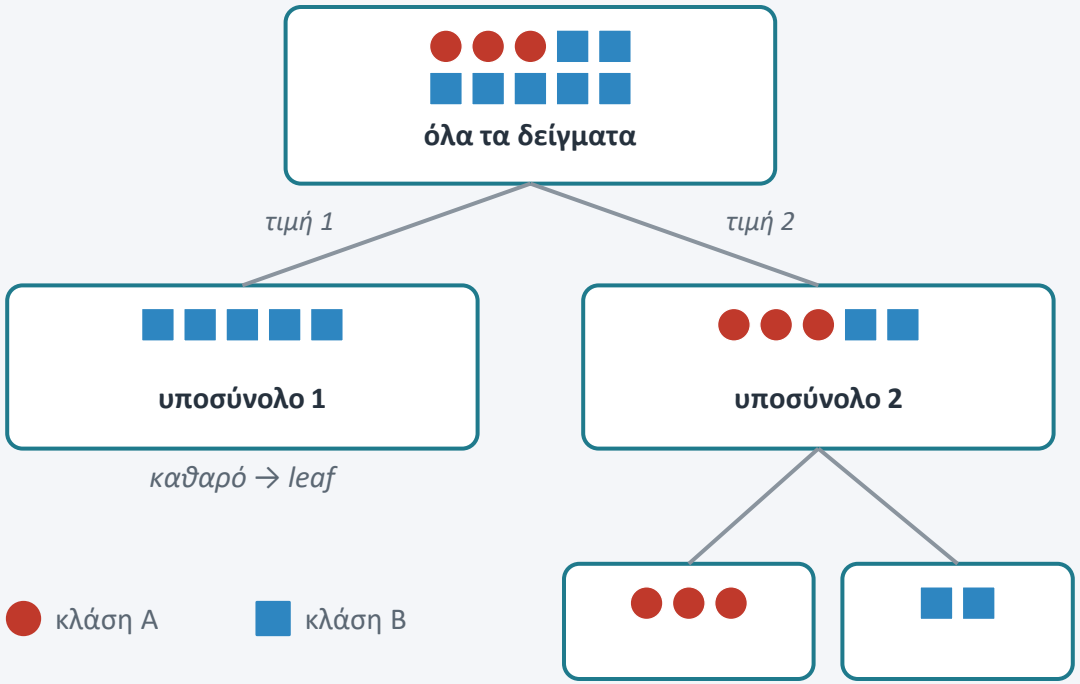
Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Από τα δείγματα εκπαίδευσης στην πρόβλεψη

1 · Βασική λογική

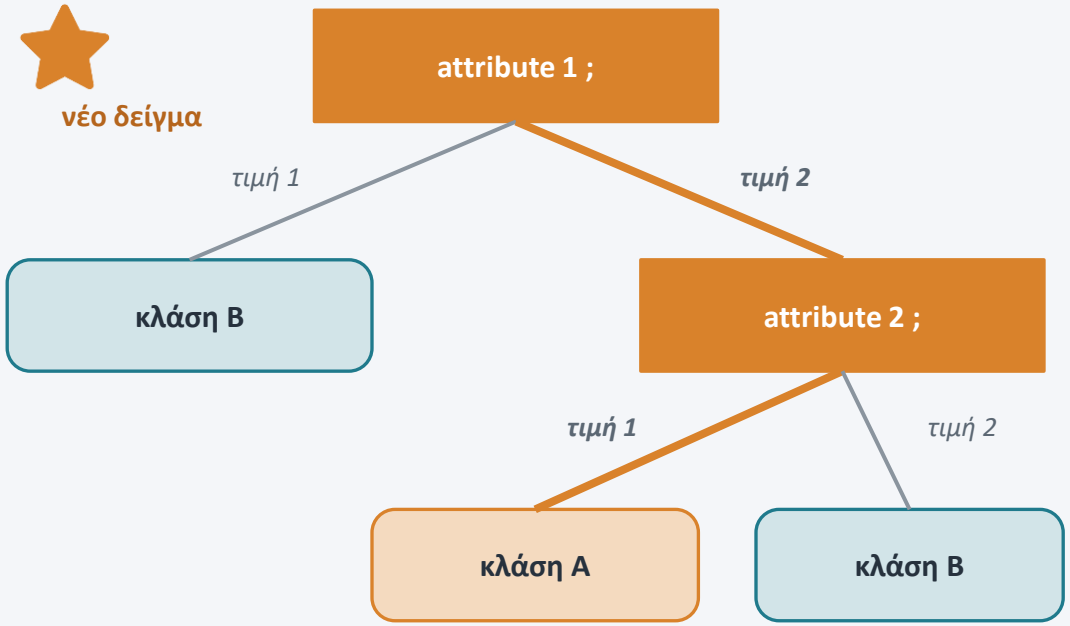
Στην εκπαίδευση τα δείγματα μοιράζονται· στην πρόβλεψη ένα νέο δείγμα ακολουθεί ένα path

1 · Εκπαίδευση (training): διάσπαση



Σε κάθε node επιλέγεται ένα attribute και τα δείγματα μοιράζονται στα παιδιά, μέχρι κάθε δείγμα να καταλήξει σε φύλλο.

2 · Πρόβλεψη (prediction): ένα path



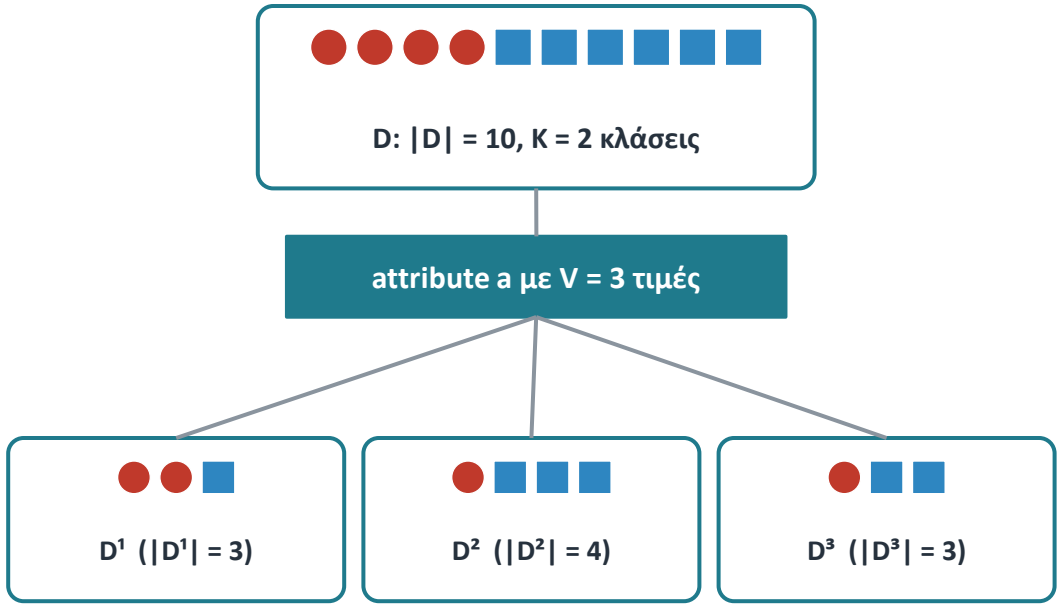
Το δέντρο δεν αλλάζει: το νέο δείγμα απαντά τις ερωτήσεις από τη ρίζα προς τα κάτω. Πρόβλεψη = η απόφαση του φύλλου όπου καταλήγει.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Splitting ≠ classification · πότε έχουμε φύλλο

1 · Βασική λογική

Το V αφορά τη διάσπαση (τιμές του attribute), το K τις κλάσεις



Splitting → V

Πόσα υποσύνολα δίνει το attribute (nodes στο επόμενο επίπεδο)

Classification → K

Πόσες κλάσεις υπάρχουν· κάθε υποσύνολο μπορεί να είναι ακόμη ανακατεμένο

Πότε ένα node γίνεται φύλλο

1. Όλα τα δείγματα ανήκουν σε μία κλάση → δεν ταξινομούνται περισσότερο
2. Όλα τα attributes έχουν ήδη χρησιμοποιηθεί στο path, ή όλα τα δείγματα έχουν τις ίδιες τιμές attributes → δεν διασπώνται
3. Δεν υπάρχουν δείγματα στο node → τίποτα για διάσπαση/ταξινόμηση

ΚΥΡΙΟ ΜΗΝΥΜΑ

Το αποτέλεσμα της ταξινόμησης το βλέπουμε μόνο στα φύλλα. Τα ενδιάμεσα υποσύνολα είναι συνήθως μείγμα κλάσεων.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

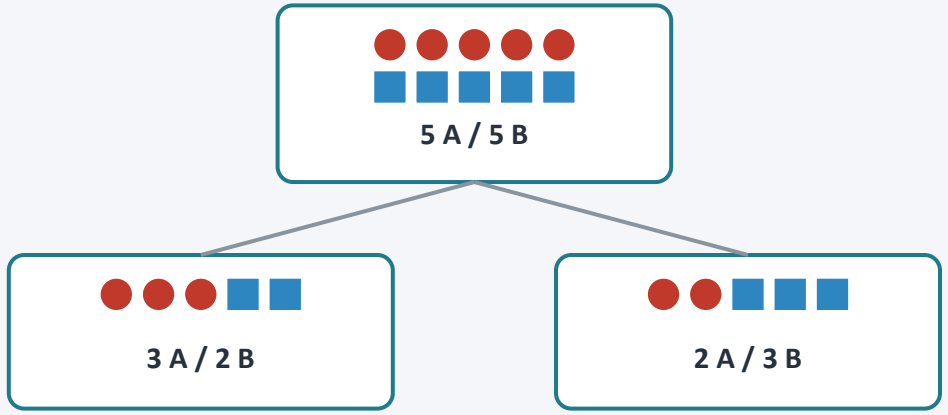
Η καλύτερη διάσπαση χρειάζεται ένα μέτρο

1 · Βασική λογική

Liu, §4.2 · σσ. 116–117

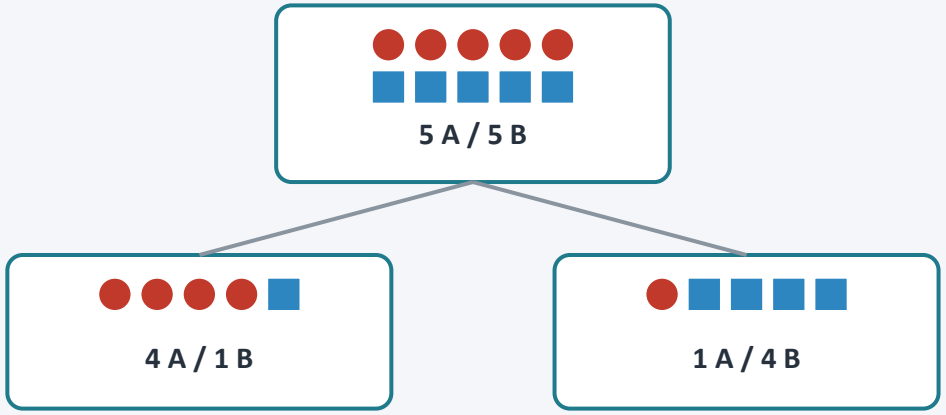
Η δεύτερη βασική ερώτηση του κεφαλαίου: χρειαζόμαστε ένα μέτρο «καλής» διάσπασης

Διάσπαση A (attribute a_1)



σχεδόν όσο ανακατεμένα όσο πριν

Διάσπαση B (attribute a_2)



πολύ πιο καθαρά υποσύνολα

Η B είναι καλύτερη: κάθε υποσύνολό της κυριαρχείται από μία κλάση (4 προς 1), ενώ στην A τα υποσύνολα μένουν σχεδόν μισό-μισό (3 προς 2). Για να το αποφασίζει ένας αλγόριθμος για πολλά attributes, χρειάζεται ένας αριθμός που μετρά την καθαρότητα (purity) κάθε διάσπασης.

ΚΥΡΙΟ ΜΗΝΥΜΑ

Ένα ευρέως αποδεκτό μέτρο προέρχεται από τη θεωρία πληροφορίας: την information entropy.

Information entropy: μέτρο αβεβαιότητας

2 · Entropy & κριτήρια splitting

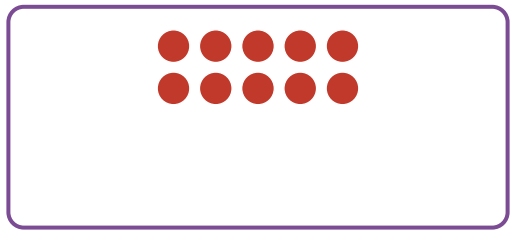
Πόσο «ανακατεμένες» είναι οι κλάσεις σε ένα σύνολο δειγμάτων

$$H(D) = -\sum_{k=1}^{K} P_k \log_2 P_k \qquad (4.1)$$

P_k	ποσοστό δειγμάτων της κλάσης k: $P_k = D_k / D$
K	πλήθος κλάσεων στο σύνολο D
$\log_2$	μονάδα: bit (δική μας σημείωση)
$0 \cdot \log 0$	ορίζεται ίσο με 0 (σύμβαση· δεν αναφέρεται στο κεφάλαιο)

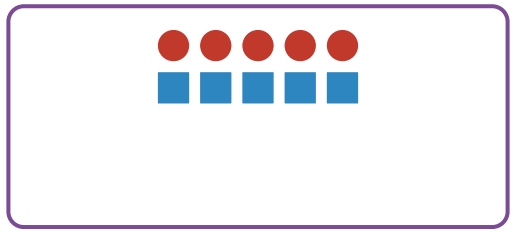
ΚΥΡΙΟ ΜΗΝΥΜΑ

H = 0 όταν όλα τα δείγματα ανήκουν σε μία κλάση· μέγιστη ($\log_2 K$) όταν οι K κλάσεις είναι ισόποσες. Μικρή H → πιο καθαρή πληροφορία. Στόχος κάθε διάσπασης: να μειωθεί η H.



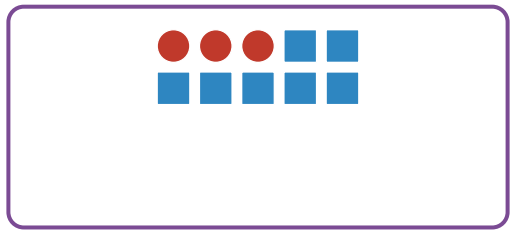
H = 0

καθαρό



H = 1

μέγιστη (K = 2)



H = 0.881

ενδιάμεσο

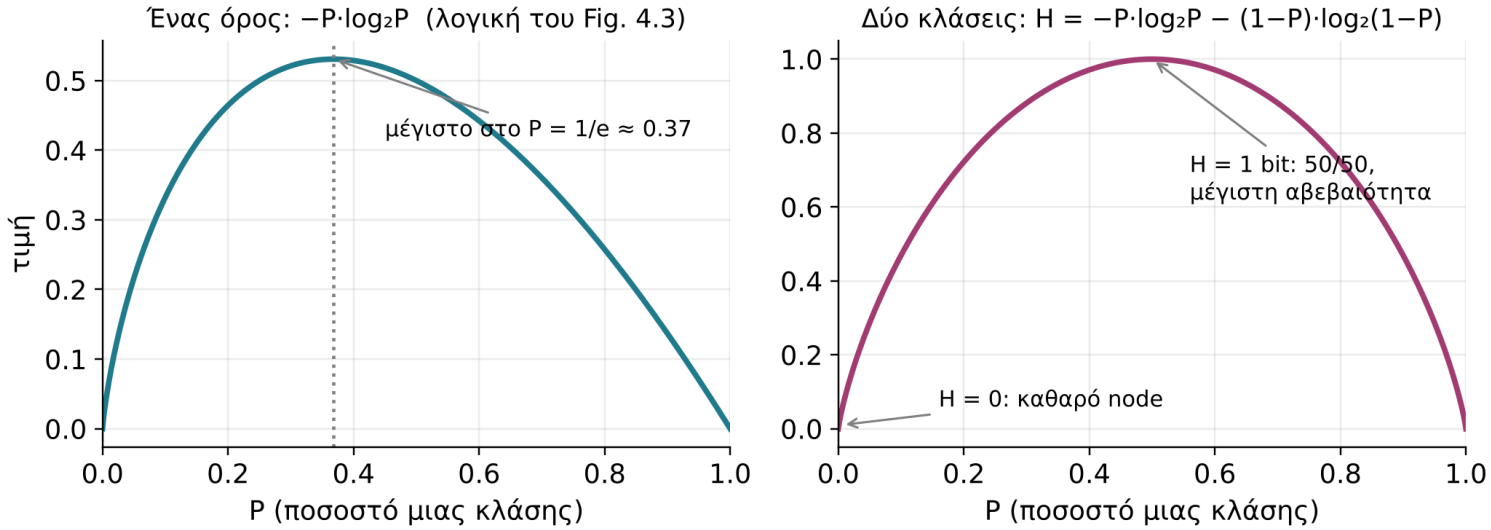
Κάθε κουτί είναι ένα σύνολο 10 δειγμάτων (κόκκινο: κλάση A, μπλε: κλάση B) και δίπλα η entropy του. Όσο πιο ανακατεμένο, τόσο μεγαλύτερη η H. Με 3 κλάσεις (Fig. 4.2: 3/10, 5/10, 2/10) H = 1.485 bit.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Πώς μεταβάλλεται η entropy

2 · Entropy & κριτήρια splitting

Χαμηλή τιμή όταν το P_k είναι πολύ μικρό ή πολύ μεγάλο



Στον οριζόντιο άξονα το ποσοστό P μιας κλάσης, στον κατακόρυφο η τιμή. Αριστερά ένας μόνο όρος $-P \cdot \log_2 P$ (η λογική του Fig. 4.3)· δεξιά η entropy δύο κλάσεων, που είναι 0 στα άκρα (καθαρό σύνολο) και μέγιστη, 1 bit, στο 50/50.

ΚΥΡΙΟ ΜΗΝΥΜΑ

Η μείωση της H κατά τη διάσπαση σημαίνει ότι όλο και περισσότερα δείγματα ενός υποσυνόλου ανήκουν σε μία ή λίγες κλάσεις.

ΠΡΟΣΟΧΗ ΣΤΗΝ ΑΝΑΓΝΩΣΗ

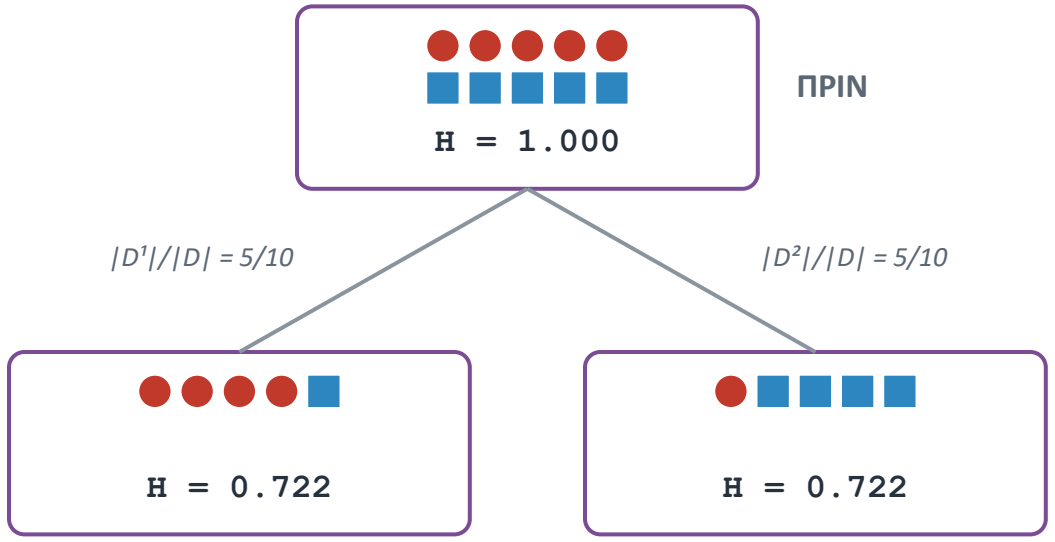
Το Fig. 4.3 δείχνει στην ουσία έναν όρο $-P \cdot \log_2 P$ (μέγιστο ≈ 0.53 στο P ≈ 0.37). Η H δύο κλάσεων (δεξιά) έχει μέγιστο 1 στο P = 0.5.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Εντροπία πριν και μετά τη διάσπαση

2 · Entropy & κριτήρια splitting

Μετά τη διάσπαση μετράμε τη σταθμισμένη μέση entropy των υποσυνόλων



Αρχικά, ο γονικός κόμβος περιέχει 10 δείγματα, μοιρασμένα ισόποσα στις δύο κλάσεις (5/5), οπότε η εντροπία του είναι (H=1). Μετά τη διάσπαση, προκύπτουν δύο υποσύνολα των 5 δειγμάτων: το ένα έχει αναλογία 4/1 και το άλλο 1/4. Και τα δύο έχουν εντροπία (H=0.722) και, επειδή περιέχουν από 5 στα 10 συνολικά δείγματα, συμμετέχουν με βάρος (5/10) στον υπολογισμό της εντροπίας μετά τη διάσπαση.

H(μετά) = 5/10 · 0.722 + 5/10 · 0.722 = 0.722

Μείωση = 1.000 − 0.722 = 0.278 bit

$$H(D|a) = \sum_{v=1}^{V} \frac{|D^v|}{|D|} H(D^v)$$

$\|D^v\|/\|D\|$	βάρος: τι ποσοστό των δειγμάτων πήγε στο υποσύνολο v
$H(D^v)$	entropy κάθε υποσυνόλου με την (4.1)
$H(D\|a)$	η entropy «αφού ξέρουμε» το a, μετά τη διάσπαση

ΣΥΧΝΟ ΛΑΘΟΣ

Δεν παίρνουμε απλό μέσο όρο. Ένα καθαρό υποσύνολο με 1 δείγμα δεν πρέπει να «μετράει» όσο ένα με 100.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Information gain

2 · Entropy & κριτήρια splitting

Πόσο μειώνεται η αβεβαιότητα όταν χωρίζουμε τα δεδομένα με ένα χαρακτηριστικό (attribute);

$$\text{Gain}(D, a) = H(D) - \sum_{v=1}^{V} \frac{|D^v|}{|D|} H(D^v) \qquad (4.2)$$

Gain(D, a) = H(D) − H(D|a) = εντροπία πριν − σταθμισμένη εντροπία μετά

ΠΡΙΝ

$H(D) = 1.000$

αρχική αβεβαιότητα στον γονικό κόμβο

−

ΜΕΤΑ

$H(D|a) = 0.722$

σταθμισμένη αβεβαιότητα των υποσυνόλων

=

ΚΕΡΔΟΣ ΠΛΗΡΟΦΟΡΙΑΣ

0.278 bit

αβεβαιότητα που αφαιρέθηκε από τη διάσπαση

Μεγάλο κέρδος πληροφορίας

Η διάσπαση δημιουργεί καθαρότερα υποσύνολα → είναι πιο χρήσιμη.

Κέρδος πληροφορίας ≈ 0

Η αβεβαιότητα σχεδόν δεν μειώνεται → η διάσπαση βοηθά ελάχιστα.

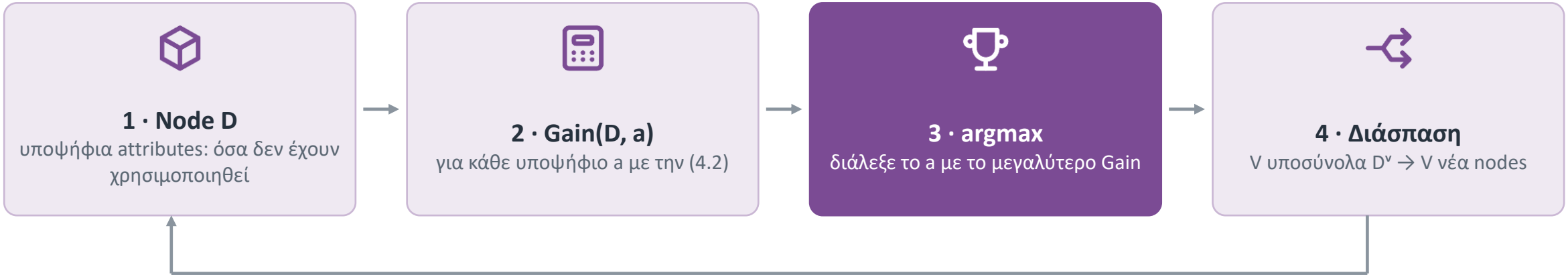
Κύριο μήνυμα: όσο μεγαλύτερο το Gain, τόσο περισσότερο μειώνεται η αβεβαιότητα. Στον ID3 επιλέγουμε το χαρακτηριστικό με το μεγαλύτερο Gain.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Επιλογή attribute με το μέγιστο information gain

2 · Entropy & κριτήρια splitting

Iterative Dichotomiser 3 (ID3): ένας από τους πρώτους αλγορίθμους δέντρων απόφασης



ΚΥΡΙΟ ΜΗΝΥΜΑ (ερμηνεία μας)

Το ID3 είναι «άπληστο» (greedy): σε κάθε node κοιτάζει μόνο την καλύτερη διάσπαση τώρα, χωρίς να ελέγχει τι θα συμβεί πιο κάτω.

ΑΔΥΝΑΜΙΑ ΤΟΥ ID3

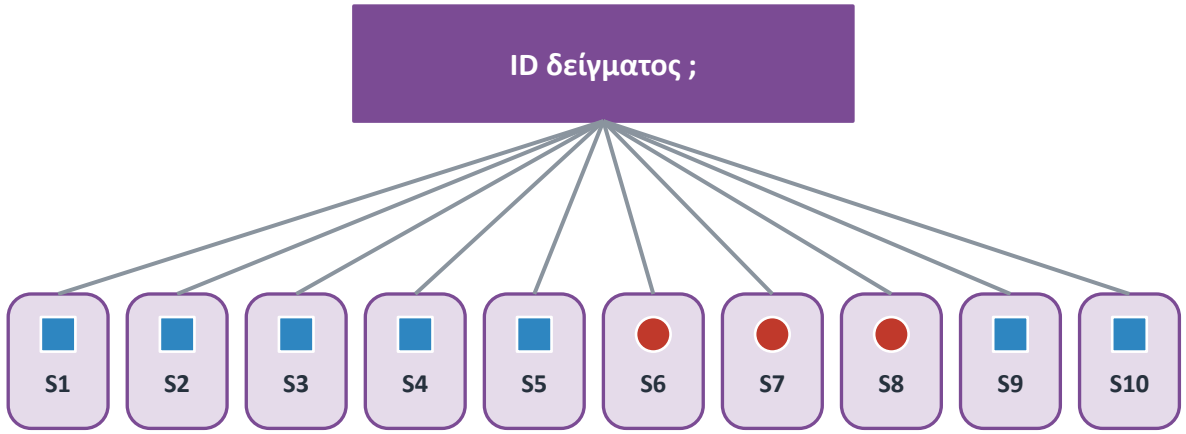
Το Gain ανταμείβει τα καθαρά υποσύνολα όσο μικρά κι αν είναι. Γι' αυτό το ID3 ευνοεί attributes που σπάνε τα δεδομένα σε πολλά μικρά υποσύνολα.

Η διαδικασία είναι αναδρομική: κάθε νέο node αντιμετωπίζεται σαν μικρότερο πρόβλημα με τα δικά του δεδομένα και τα attributes που απέμειναν.

Το πρόβλημα του ID3: attributes με πολλές τιμές

2 · Entropy & κριτήρια splitting

Ένας μοναδικός κωδικός (ID) δίνει το μέγιστο Gain, αλλά δεν γενικεύει



Τι δείχνει: κάθε δείγμα πηγαίνει σε δικό του φύλλο (10 φύλλα του ενός δείγματος). Όλα είναι καθαρά (H = 0), άρα Gain = H(D), το μέγιστο, χωρίς καμία γενίκευση.

Attribute	V	Gain
ID δείγματος	**10**	**0.881 = H(D)**
Ψύξη	2	0.396

«Δεν ταξινομούμε ανθρώπους με τον αριθμό διαβατηρίου τους»

Ένας μοναδικός κωδικός (ID) μπορεί να ξεχωρίσει τέλεια όλα τα δείγματα που έχουμε ήδη δει. Αυτό όμως δεν σημαίνει ότι βοηθά το μοντέλο να προβλέψει σωστά ένα νέο δείγμα.

- **Τι κάνει ο ID3:** προτιμά συχνά χαρακτηριστικά (attributes) που χωρίζουν τα δεδομένα σε πολλές μικρές ομάδες.
- **Γιατί αυτό είναι πρόβλημα:** αν χρησιμοποιήσουμε έναν μοναδικό κωδικό δείγματος, κάθε δείγμα καταλήγει μόνο του σε μία ομάδα. Έτσι το δέντρο φαίνεται «τέλειο» στα δεδομένα εκπαίδευσης.
- **Τι συμβαίνει σε νέα δεδομένα:** ο κωδικός ενός νέου δείγματος δεν έχει κάποια χρήσιμη σχέση με την κατηγορία που θέλουμε να προβλέψουμε. Άρα το δέντρο δεν γενικεύει καλά.

ΚΥΡΙΟ ΜΗΝΥΜΑ

Το C4.5 είναι μια βελτιωμένη εκδοχή του ID3. Χρησιμοποιεί τον λόγο κέρδους (gain ratio) ώστε να μην ευνοούνται τόσο εύκολα χαρακτηριστικά που απλώς χωρίζουν τα δεδομένα σε πάρα πολλές μικρές ομάδες.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

C4.5: Λόγος κέρδους (Gain Ratio)

2 · Entropy & κριτήρια splitting

α διορθώσουμε την τάση του ID3 να προτιμά χαρακτηριστικά (attributes) που χωρίζουν τα δεδομένα σε πάρα πολλές ομάδες.

$$\text{Gain_ratio}(D, a) = \frac{\text{Gain}(D, a)}{IV(a)} \quad (4.3)$$

$$IV(a) = -\sum_{v=1}^{V} \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|} \quad (4.4)$$

`IV(a) ↑` το χαρακτηριστικό δημιουργεί πολλά, ισομεγέθη υποσύνολα (μεγάλο V) → ο Gain διορθώνεται περισσότερο.

`IV(a) ↓` το χαρακτηριστικό δημιουργεί λίγα υποσύνολα→ η διόρθωση είναι μικρότερη.

ΠΡΟΣΟΧΗ
Το gain ratio μόνο του έχει την αντίθετη ροπή: προτιμά attributes με λίγες τιμές, που επίσης βλάπτει τη γενίκευση.

Ο κανόνας του C4.5 (δύο βήματα)

Όλα τα υποψήφια
ID, Άνθρακας, Ψύξη, Επαναφορά
→
Gain > μέσος όρος
μέσος = 0.391 → ID, Ψύξη
→
max Gain_ratio
→ Ψύξη

Attribute	Gain	IV	Gain_ratio
ID δείγματος	0.881	3.322	0.265
Ψύξη	**0.396**	**1.000**	**0.396**

Παρότι το ID έχει μεγαλύτερο Information Gain, δημιουργεί πάρα πολλά υποσύνολα και «τιμωρείται» μέσω του (IV).

ΚΥΡΙΟ ΜΗΝΥΜΑ
Το C4.5 δεν επιλέγει απλώς το μεγαλύτερο Information Gain. Χρησιμοποιεί το Gain Ratio ώστε να αποφεύγει χαρακτηριστικά που φαίνονται καλά μόνο επειδή χωρίζουν τα δεδομένα σε πολλές μικρές ομάδες.

Gini value: πιθανότητα διαφορετικών ετικετών

2 · Entropy & κριτήρια splitting

Αν τραβήξουμε δύο δείγματα στην τύχη, πόσο πιθανό είναι να ανήκουν σε διαφορετικές κλάσεις;

$$Gini(D) = \sum_{k=1}^{K} \sum_{k' \neq k} P_k \, P_{k'} \; (4.5) \;\rightarrow\; Gini(D) = 1 - \sum_{k=1}^{K} P_k^2 \; (4.6)$$

Η (4.6): το άθροισμα των P_k' για $k' \neq k$ είναι $1 - P_k$, άρα Gini = Σ $P_k(1 - P_k)$ = 1 − Σ P_k^2



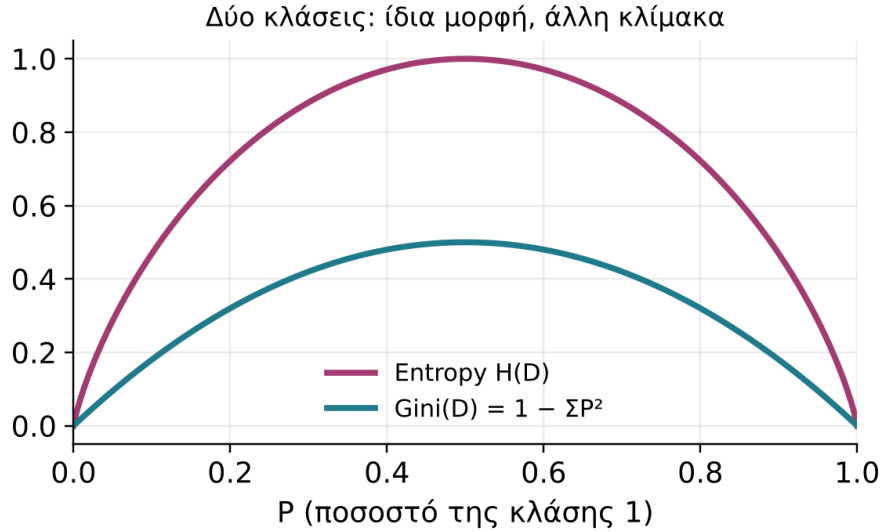
τράβα 2 →



διαφορετικές;

`3 A / 7 B:  Gini = 1 − (0.3² + 0.7²) = 0.42`

`Gini = 0`	καθαρό σύνολο: δύο δείγματα έχουν πάντα ίδια ετικέτα
`Gini = 0.5`	μέγιστο για K = 2 (μισό-μισό)



Η entropy και το Gini δύο κλάσεων ως προς το ποσοστό P της κλάσης 1: 0 στα άκρα, μέγιστο στο 50/50, άλλη κλίμακα.

ΚΥΡΙΟ ΜΗΝΥΜΑ

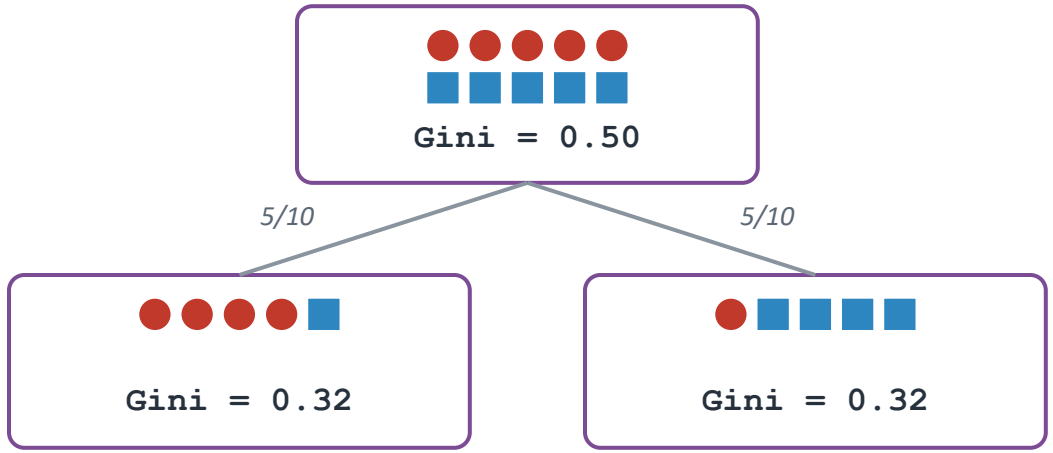
Μικρότερο Gini → πιο καθαρά δεδομένα, λιγότερη αβεβαιότητα. Ένα δεύτερο μέτρο, εκτός της entropy.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Gini index: το κριτήριο του CART

Σταθμισμένο άθροισμα των Gini values των υποσυνόλων, διαλέγουμε το μικρότερο

$$\text{Gini_index}(D, a) = \sum_{v=1}^{V} \frac{|D^v|}{|D|} \text{Gini}(D^v) \qquad (4.7)$$



Gini_index = 0.5·0.32 + 0.5·0.32 = 0.32

Η διάσπαση B της διαφ. 8 με Gini: γονέας 0.50, κάθε παιδί 0.32, βάρος 5/10 (η A δίνει 0.48).

Κανόνας CART: το μικρότερο Gini index

Αντίθετα από το Gain, εδώ μετράμε την αβεβαιότητα που απομένει, άρα θέλουμε ελάχιστο.

CART στο scikit-learn

Ο DecisionTreeClassifier χρησιμοποιεί εξ ορισμού μια βελτιστοποιημένη έκδοση του CART (classification and regression tree).

ID3, C4.5, CART: σύγκριση

2 · Entropy & κριτήρια splitting

Ίδια ιδέα (μείωση αβεβαιότητας), διαφορετικό μέτρο και κανόνας επιλογής

	ID3	C4.5	CART
Μέτρο καθαρότητας	entropy H (4.1)	entropy H (4.1)	Gini value (4.5)–(4.6)
Κριτήριο διάσπασης	Gain (4.2)	Gain_ratio = Gain / IV (4.3)–(4.4)	Gini_index (4.7)
Κανόνας επιλογής	**μέγιστο Gain**	**Gain > μέσος όρος, μετά μέγιστο Gain_ratio**	**ελάχιστο Gini_index**
Γνωστή ροπή	attributes με πολλές τιμές (π.χ. ID)	το Gain_ratio μόνο του: λίγες τιμές	—
Στο βιβλίο	ένας από τους πρώτους	διόρθωση του ID3	νεότερος, πιο διαδεδομένος· scikit-learn

ΚΥΡΙΟ ΜΗΝΥΜΑ

Στο worked example και τα τρία κριτήρια επιλέγουν το ίδιο attribute (Ψύξη). Αυτό δεν είναι εγγυημένο: το ID δείχνει πότε διαφωνούν.

ΣΥΧΝΟ ΛΑΘΟΣ

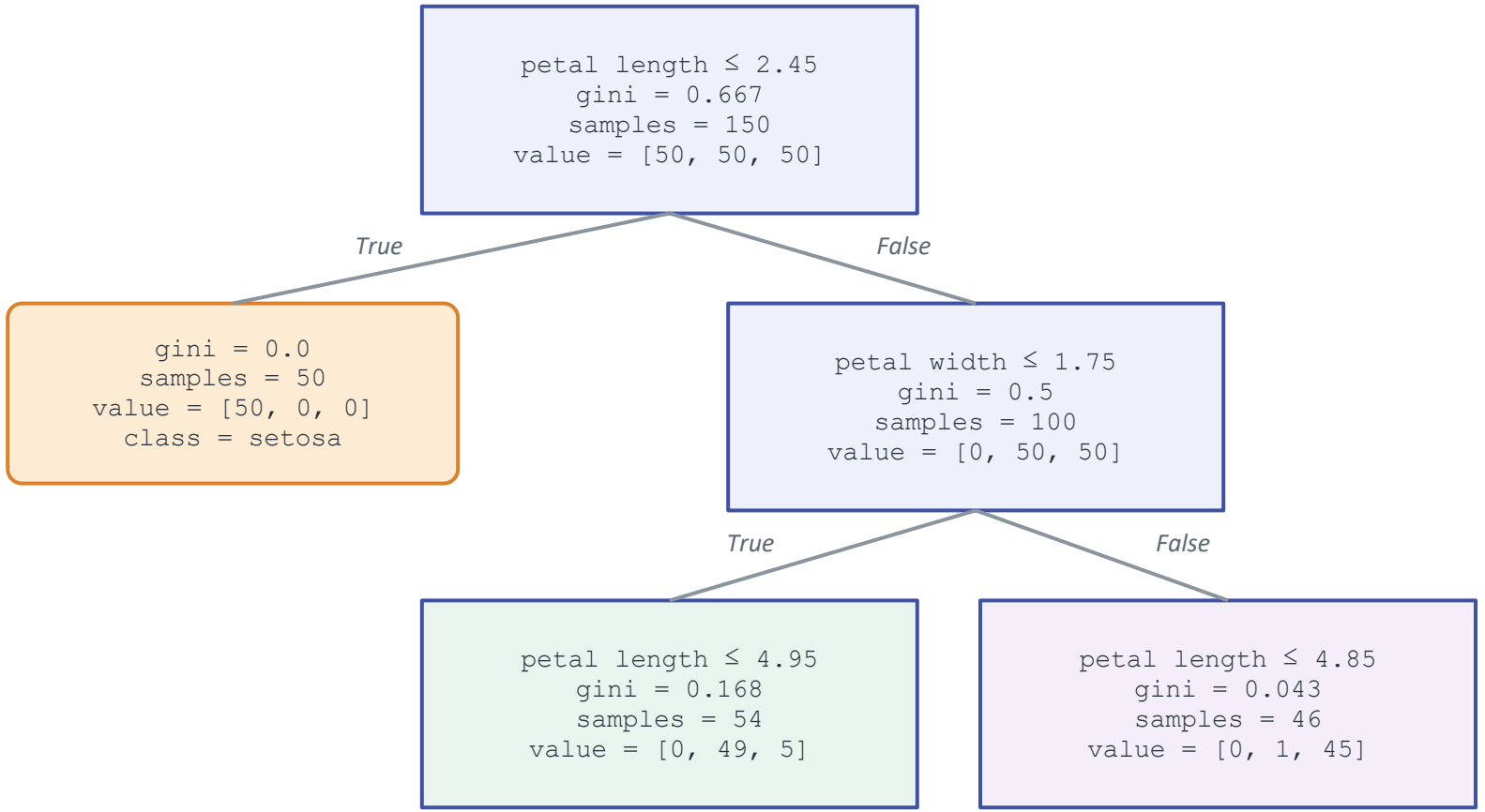
Gain και Gain_ratio: μέγιστο. Gini_index: ελάχιστο.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Πώς διαβάζουμε ένα δέντρο του scikit-learn

3 · Από τον αλγόριθμο στο tree

Το Iris dataset με CART: συνθήκη, gini, samples, value, class σε κάθε node



Οδηγός ανάγνωσης ενός node

`≤ 2.45`	συνθήκη: True → αριστερά
`gini`	Gini value του node (4.6)
`samples`	πόσα δείγματα έφτασαν εδώ
`value`	πλήθος ανά κλάση
`class`	η πλειοψηφούσα κλάση

ΕΛΕΓΧΟΣ ΚΑΙ ΠΑΡΑΤΗΡΗΣΕΙΣ

Ρίζα: $1 - 3 \cdot (1/3)^2 = 0.667$. Το setosa ξεχωρίζει με μία ερώτηση (gini 0). Με συνεχή attributes κάθε ερώτηση είναι κατώφλι με δύο κλάδους, και το ίδιο attribute μπορεί να ξαναχρησιμοποιηθεί.

Το δέντρο ταξινομεί δείγματα του Iris ακολουθώντας διαδοχικές συνθήκες. Κάθε κόμβος (node) δείχνει τη συνθήκη διάσπασης, την τιμή Gini, πόσα δείγματα έφτασαν εκεί και πόσα ανήκουν σε κάθε κλάση. Η διαδρομή από τη ρίζα προς ένα φύλλο οδηγεί στην τελική πρόβλεψη.

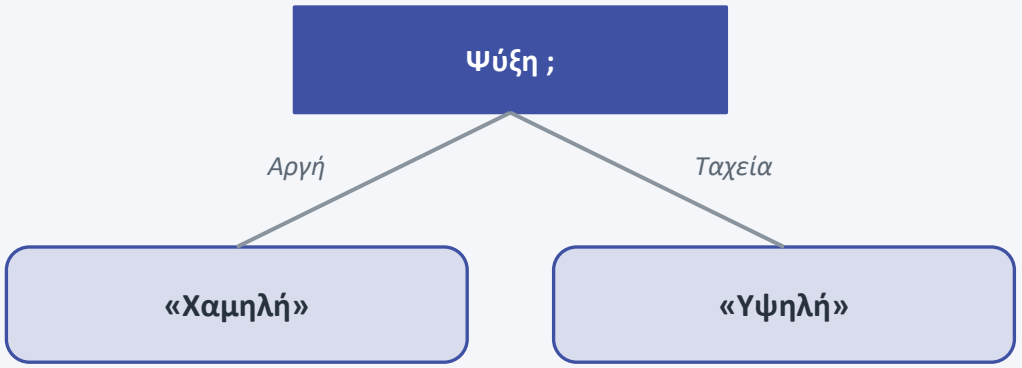
Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Classification και regression trees

3 · Από τον αλγόριθμο στο tree

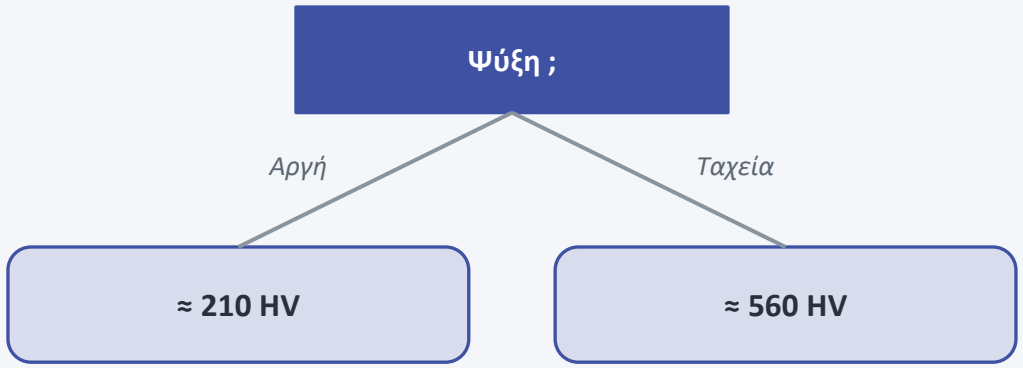
Τα δέντρα απόφασης μπορούν να χρησιμοποιηθούν για δύο διαφορετικά είδη προβλημάτων:

Classification: προβλέπει μια κατηγορία (class).



Κάθε φύλλο δίνει μια ετικέτα κλάσης, π.χ. «Υψηλή»

Regression: προβλέπει μια αριθμητική τιμή.



Το ίδιο δέντρο, αλλά κάθε φύλλο δίνει έναν αριθμό

Στο κεφάλαιο

Η εκπαίδευση classification ή regression μοντέλου με δέντρο είναι σταδιακή διάσπαση (σ. 115). CART = classification and regression tree.

Επέκταση

Σε regression το φύλλο δίνει συνήθως τον μέσο όρο των τιμών του, στο scikit-learn: DecisionTreeRegressor.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Worked example: 10 δείγματα χάλυβα

4 · Παράδειγμα βήμα-βήμα

Στόχος: να δούμε πώς χτίζεται ένα δέντρο απόφασης, από τα δεδομένα μέχρι την τελική πρόβλεψη.

ID	Άνθρακας	Ψύξη	Επαναφορά	Σκληρότητα
S1	Χαμηλός	Αργή	Όχι	Χαμηλή
S2	Χαμηλός	Αργή	Ναι	Χαμηλή
S3	Υψηλός	Αργή	Όχι	Χαμηλή
S4	Υψηλός	Αργή	Ναι	Χαμηλή
S5	Υψηλός	Αργή	Όχι	Χαμηλή
S6	Υψηλός	Ταχεία	Όχι	**Υψηλή**
S7	Υψηλός	Ταχεία	Όχι	**Υψηλή**
S8	Υψηλός	Ταχεία	Ναι	**Υψηλή**
S9	Χαμηλός	Ταχεία	Όχι	Χαμηλή
S10	Χαμηλός	Ταχεία	Ναι	Χαμηλή

Τι είναι οι στήλες;

Χαρακτηριστικά (attributes): Άνθρακας, Ψύξη, Επαναφορά. Είναι οι πληροφορίες που δίνουμε στο δέντρο για κάθε δείγμα.

Στόχος / κλάση (class): Σκληρότητα = Υψηλή ή Χαμηλή. Αυτή είναι η κατηγορία που θέλουμε να προβλέψουμε.

Τι θα κάνουμε στα επόμενα βήματα;

1. Μετράμε την αβεβαιότητα στη ρίζα (root).
2. Δοκιμάζουμε τα τρία χαρακτηριστικά ως πιθανές διασπάσεις (splits).
3. Επιλέγουμε την καλύτερη διάσπαση και επαναλαμβάνουμε.
4. Χρησιμοποιούμε το τελικό δέντρο για ένα νέο δείγμα.

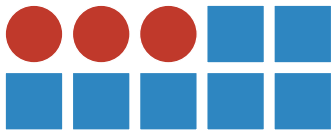
Συνθετικά διδακτικά δεδομένα: σκοπός είναι να κατανοήσουμε τον αλγόριθμο, όχι να εξαγάγουμε φυσικό νόμο για τους χάλυβες.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Βήμα 1: αβεβαιότητα στη ρίζα

4 · Worked example

Πριν χωρίσουμε τα δεδομένα, μετράμε πόσο «ανακατεμένες» είναι οι δύο κλάσεις.



Ρίζα: |D| = 10

Υψηλή (3) Χαμηλή (7)

$$P(\text{Υψηλή}) = 3/10 = 0.3$$
$$P(\text{Χαμηλή}) = 7/10 = 0.7$$

ΚΥΡΙΟ ΜΗΝΥΜΑ

Γιατί μετράμε αβεβαιότητα;
Για να έχουμε μια τιμή αναφοράς «πριν» και να κρίνουμε αν μια διάσπαση κάνει τα δεδομένα πιο καθαρά.

Αυτές είναι οι τιμές «πριν». Κάθε υποψήφια διάσπαση θα συγκριθεί με αυτές.

Entropy (4.1) - για ID3 και C4.5

$$H(D) = -0.3 \cdot \log_2 0.3 - 0.7 \cdot \log_2 0.7$$
$$= 0.521 + 0.360$$

$$\mathbf{H(D) = 0.881\ bit}$$

Όσο μικρότερη η H, τόσο πιο καθαρό το σύνολο.

Gini value (4.6) - για CART

$$Gini(D) = 1 - (0.3^2 + 0.7^2)$$
$$= 1 - (0.09 + 0.49)$$

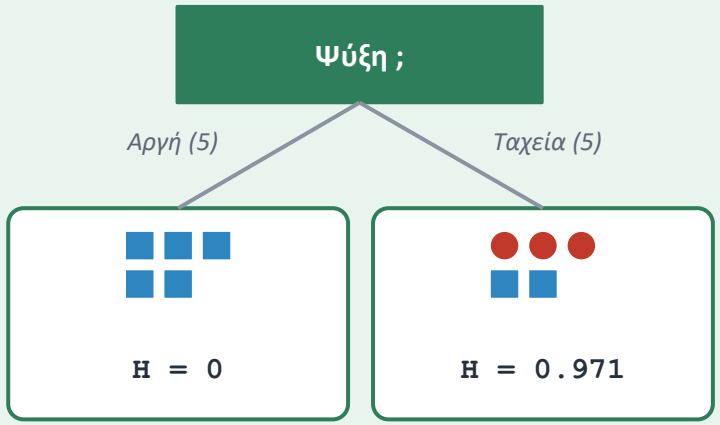
$$\mathbf{Gini(D) = 0.42}$$

Και εδώ: μικρότερη τιμή = καθαρότερο σύνολο.

Βήμα 2: τρεις υποψήφιες διασπάσεις (ID3)

4 · Worked example

Για κάθε υποψήφιο χαρακτηριστικό: διάσπαση → αβεβαιότητα υποσυνόλων → σταθμισμένος μέσος → κέρδος πληροφορίας (Gain).



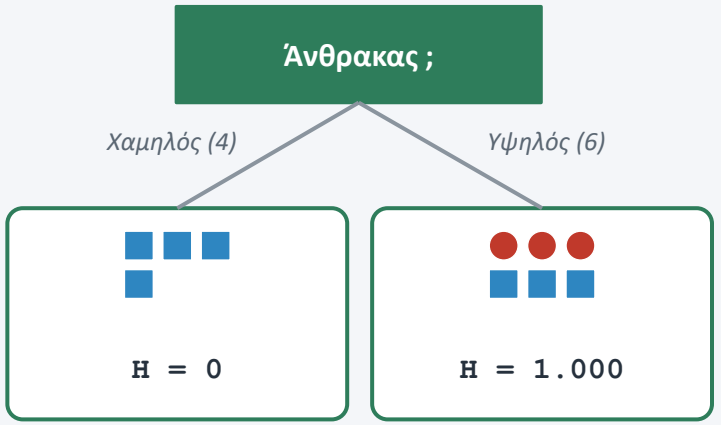
H(D|a) =

5/10·0 + 5/10·0.971 = 0.486

Gain = 0.881 − 0.486

Gain = 0.396

μέγιστο → επιλογή ID3

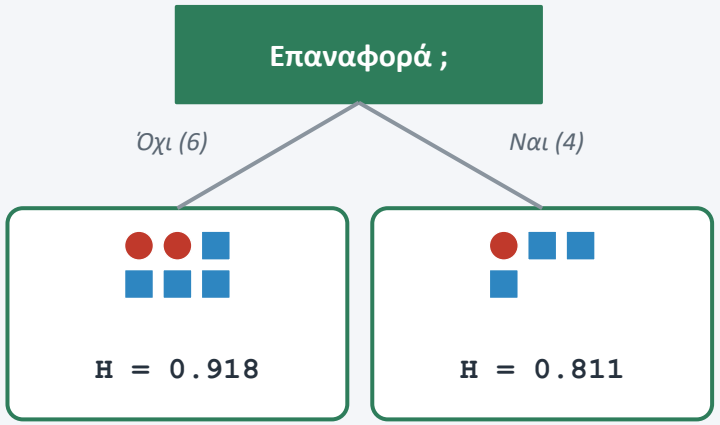


H(D|a) =

4/10·0 + 6/10·1.000 = 0.600

Gain = 0.881 − 0.600

Gain = 0.281



H(D|a) =

6/10·0.918 + 4/10·0.811 = 0.876

Gain = 0.881 − 0.876

Gain = 0.006

Βήμα 3: τα τρία κριτήρια στη ρίζα

4 · Worked example

Οι τρεις αλγόριθμοι χρησιμοποιούν διαφορετικό κριτήριο, αλλά σε αυτό το dataset καταλήγουν στην ίδια πρώτη διάσπαση: Ψύξη.

Attribute	H(D|a)	Gain	IV(a)	Gain_ratio	Gini_index
Ψύξη	**0.486**	**0.396**	**1.000**	**0.396**	**0.240**
Άνθρακας	0.600	0.281	0.971	0.290	0.300
Επαναφορά	0.876	0.006	0.971	0.006	0.417
(ID δείγματος)	*0*	*0.881*	*3.322*	*0.265*	*0*

ID3: max Gain
Ψύξη (0.396)

→

C4.5: Gain > 0.391, μετά max ratio
Ψύξη (0.396 > 0.265 του ID)

→

CART: min Gini_index
Ψύξη (0.240)

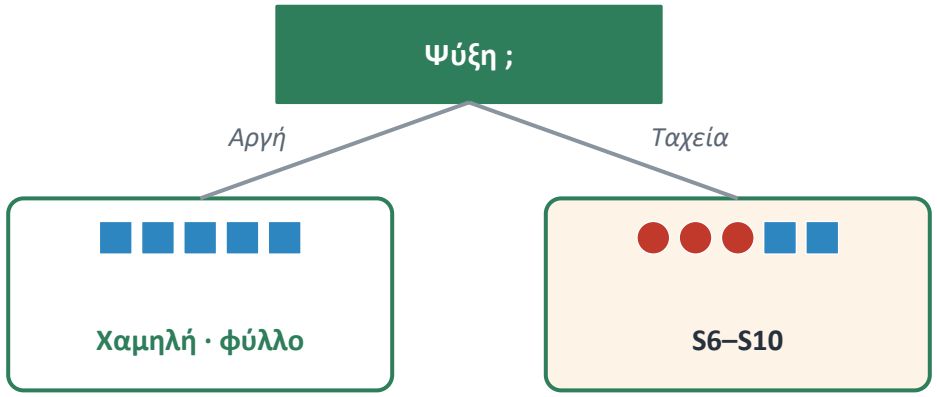
$$\text{Gini_index}(\Psi\acute{\upsilon}\xi\eta) = 5/10 \cdot 0 + 5/10 \cdot 0.48 = 0.24 \quad \cdot \quad \text{IV}(\Psi\acute{\upsilon}\xi\eta) = -2 \cdot (0.5 \cdot \log_2 0.5) = 1$$

Ο κωδικός ID δεν είναι χρήσιμο χαρακτηριστικό πρόβλεψης. Τον κρατάμε μόνο ως ελεγκτικό παράδειγμα: δείχνει γιατί το ID3 μπορεί να προτιμήσει μια «τέλεια» αλλά άχρηστη διάσπαση.

Βήμα 4: το δεύτερο επίπεδο

4 · Worked example

Το «Αργή» είναι φύλλο (κανόνας 1)· το «Ταχεία» διασπάται ξανά



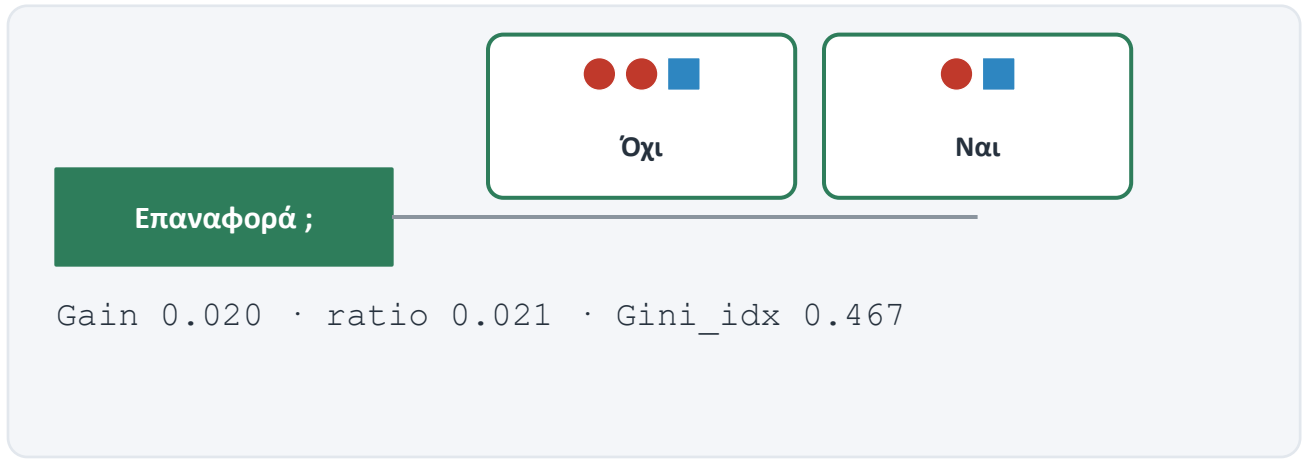
Κανόνας 1: όλα ίδιας κλάσης → φύλλο

Αριστερά το δέντρο μετά το 1ο βήμα: το «Αργή» είναι καθαρό φύλλο, ενώ το «Ταχεία» (S6–S10, H = 0.971, Gini = 0.48) θα διασπαστεί με Άνθρακα ή Επαναφορά (η Ψύξη «καταναλώθηκε»). Δεξιά οι δύο υποψήφιες διασπάσεις του.

ΕΛΕΓΧΟΣ

Μέγιστο δυνατό Gain εδώ = H(node) = 0.971: ο Άνθρακας αφαιρεί όλη την αβεβαιότητα.



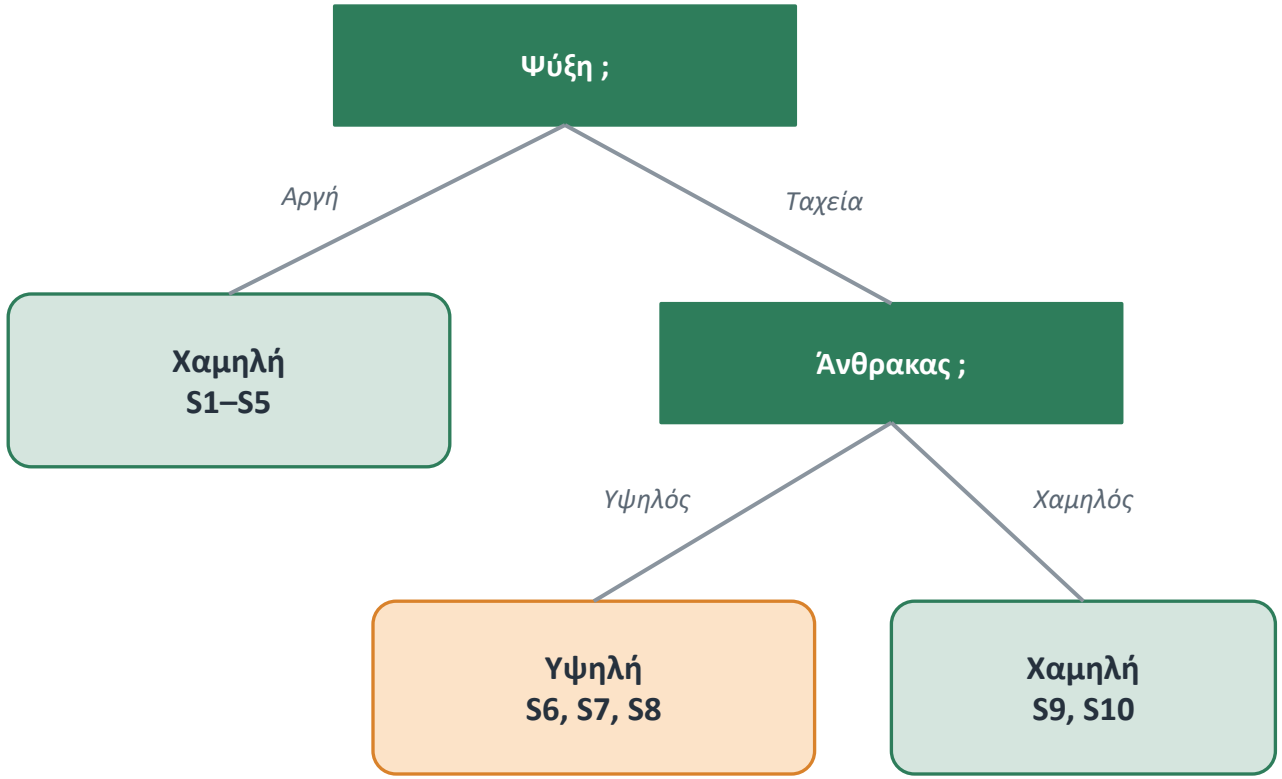


Βήμα 5: το τελικό δέντρο και οι κανόνες του

4 · Worked example

Διδακτική εφαρμογή των εννοιών του Chapter 4

Βάθος 2, τρία φύλλα, όλα καθαρά · η Επαναφορά δεν χρησιμοποιήθηκε



Τι δείχνει το σχήμα: το τελικό δέντρο του παραδείγματος. Σε κάθε φύλλο η κλάση και τα δείγματα που κατέληξαν εκεί· το πορτοκαλί είναι το μόνο φύλλο «Υψηλή». Ακρίβεια στα δεδομένα εκπαίδευσης: 10/10.

Κανόνες (ένας ανά path)

1 ΑΝ Ψύξη = Αργή ΤΟΤΕ Χαμηλή

2 ΑΝ Ψύξη = Ταχεία ΚΑΙ Άνθρακας = Υψηλός ΤΟΤΕ Υψηλή

3 ΑΝ Ψύξη = Ταχεία ΚΑΙ Άνθρακας = Χαμηλός ΤΟΤΕ Χαμηλή

ΕΦΑΡΜΟΓΗ ΣΤΑ ΥΛΙΚΑ
Ο κανόνας 2 είναι η ποιοτική εικόνα της βαφής: χρειάζονται και ταχεία ψύξη και αρκετός άνθρακας.

ΠΡΟΣΟΧΗ
Η Επαναφορά λείπει. Αυτό σημαίνει ότι δεν χρειάστηκε σε αυτά τα 10 δείγματα, όχι ότι δεν επηρεάζει τη σκληρότητα.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Βήμα 6: πρόβλεψη για ένα νέο υλικό

4 · Worked example

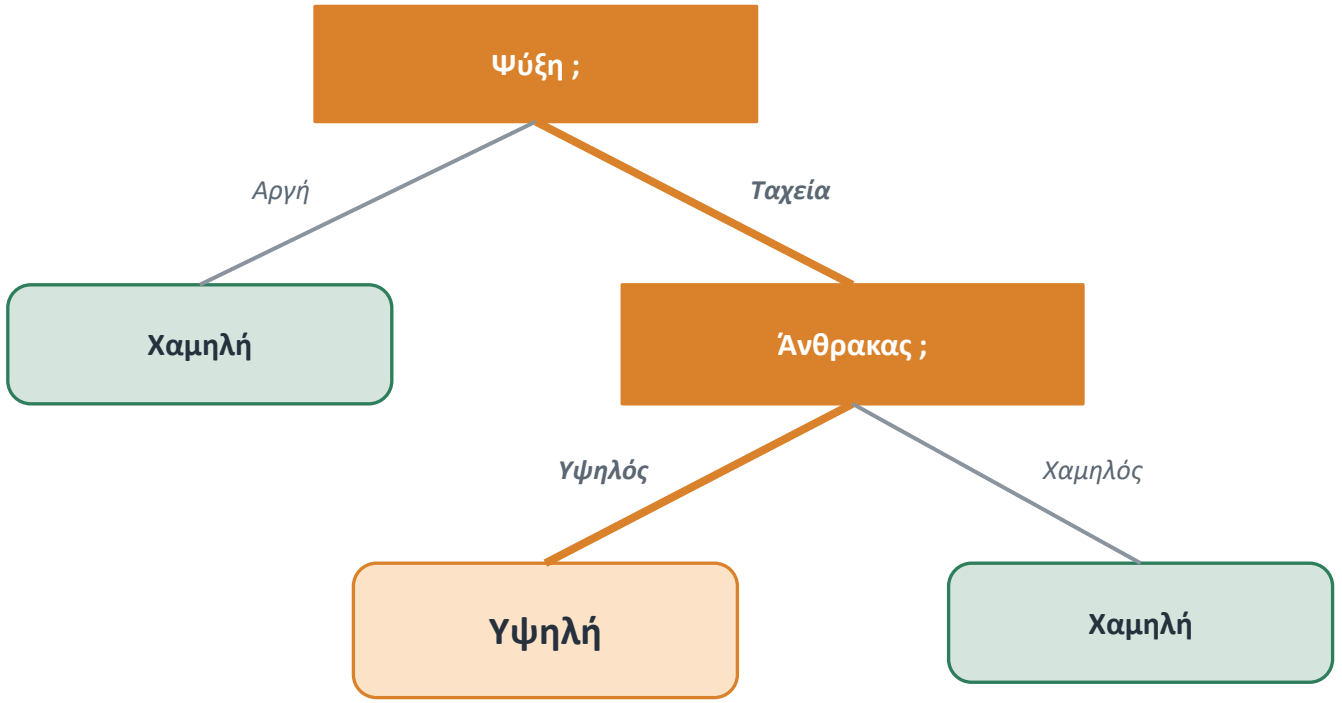
Ένα νέο δείγμα ακολουθεί το path του · δεν ξαναϋπολογίζουμε τίποτα

Νέο δείγμα (δεν είναι στα δεδομένα)

C = 0.60 wt%
Βαφή σε νερό
Επαναφορά στους 200 °C

Μετάφραση σε τιμές attributes

Άνθρακας	0.60 ≥ 0.40 → Υψηλός
Ψύξη	βαφή → Ταχεία
Επαναφορά	Ναι (δεν ρωτιέται)



Τι δείχνει: το πορτοκαλί path είναι οι απαντήσεις του νέου δείγματος (Ψύξη = Ταχεία, Άνθρακας = Υψηλός). Πρόβλεψη: Υψηλή σκληρότητα, με 2 ερωτήσεις.

ΠΡΟΣΟΧΗ
Το δέντρο ξέρει μόνο ό,τι είδε: 10 δείγματα, δύο τιμές ανά attribute. Δεν «ξέρει» π.χ. ότι επαναφορά σε 600 °C μαλακώνει τον χάλυβα.

Οι έννοιες του κεφαλαίου στη γλώσσα των υλικών

5 · Επιστήμη Υλικών

Από sample / attribute / class σε δοκίμιο / σύσταση-κατεργασία / ιδιότητα

Όρος του κεφαλαίου	Στα υλικά (παραδείγματα)
Sample (δείγμα)	ένα δοκίμιο, μια σύνθεση, μια παρτίδα παραγωγής
Attribute (χαρακτηριστικό)	σύσταση (wt%), θερμοκρασία ανόπτησης, ρυθμός ψύξης, μέγεθος κόκκου
Class (κλάση), K	κατηγορία ιδιότητας ή φάσης: Υψηλή/Χαμηλή, FCC/BCC, αποδεκτό/απορριπτέο
Splitting, V	μια ερώτηση για ένα attribute: «ψύξη ταχεία;», «C ≥ 0.40;»
Leaf (φύλλο)	η πρόβλεψη για όσα δείγματα ικανοποιούν όλο το path
Path	ένας κανόνας σύστασης–κατεργασίας–ιδιότητας που μπορούμε να ελέγξουμε πειραματικά

Ερμηνευσιμότητα

Ο μηχανικός διαβάζει και ελέγχει τους κανόνες

Μικτά δεδομένα

Κατηγορικά (διεργασία) και αριθμητικά (σύσταση) μαζί

Επόμενο πείραμα

Ένα path γίνεται υπόθεση για έλεγχο στο εργαστήριο

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Τρία προβλήματα ταξινόμησης στα υλικά

5 · Επιστήμη Υλικών

Για το καθένα: τι είναι η κλάση (K) και ποια attributes είναι υποψήφια

Κατηγορία σκληρότητας χάλυβα

K = 2: Υψηλή / Χαμηλή

Υποψήφια attributes

C wt%, ρυθμός ψύξης, θερμοκρασία επαναφοράς, διαδρομή κατεργασίας

Ψύξη ταχεία;

ενδεικτικό πρώτο split

Φάση κράματος υψηλής εντροπίας

K = 3: FCC / BCC / FCC+BCC

Υποψήφια attributes

descriptors από τη σύσταση: VEC, διαφορά ατομικών ακτίνων δ, ΔHmix

VEC ≥ κατώφλι;

ενδεικτικό πρώτο split

Το δέντρο βρίσκει το κατώφλι από τα δεδομένα

Ποιοτικός έλεγχος παραγωγής

K = 2: αποδεκτό / απορριπτέο

Υποψήφια attributes

θερμοκρασία φούρνου, χρόνος, πάχος, μετρήσεις υπερήχων

Χρόνος < όριο;

ενδεικτικό πρώτο split

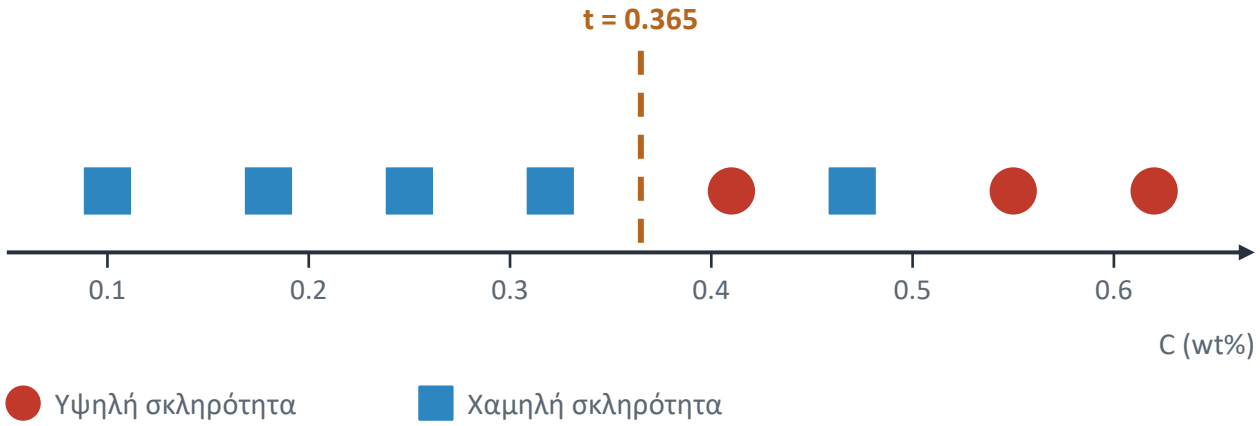
Οι κανόνες γίνονται οδηγίες γραμμής παραγωγής

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Συνεχή attributes: από τιμές σε κατώφλια

5 · Επιστήμη Υλικών

Ερώτηση «C ≤ t;» → δύο υποσύνολα · δοκιμάζουμε υποψήφια t και κρατάμε το καλύτερο



8 δείγματα στον άξονα του C (wt%). Τα υποψήφια κατώφλια t είναι τα 7 μέσα σημεία διαδοχικών τιμών· το καλύτερο, t = 0.365 (διακεκομμένη), αφήνει αριστερά μόνο Χαμηλή.

t	αριστερά (≤ t)	δεξιά (> t)	Gini_index
0.285	0 Y / 3 X	3 Y / 2 X	0.300
0.365	**0 Y / 4 X**	**3 Y / 1 X**	**0.188**
0.440	1 Y / 4 X	2 Y / 1 X	0.367
0.510	1 Y / 5 X	2 Y / 0 X	0.208

Gini(γονέας, 3 Y / 5 X) = 0.469 · εμφανίζονται 4 από τα 7 κατώφλια

Από το κεφάλαιο

Ο τύπος (4.7) και ο κανόνας «ελάχιστο Gini_index» εφαρμόζονται αυτούσια. Το σχήμα δείχνει τέτοια κατώφλια (petal length ≤ 2.45).

Επέκταση

Το πώς επιλέγονται τα υποψήφια t δεν περιγράφεται στο κεφάλαιο. Το scikit-learn ελέγχει κατώφλια ανάμεσα σε διαδοχικές ταξινομημένες τιμές.

Ερμηνεία και παγίδες σε δεδομένα υλικών

5 · Επιστήμη Υλικών

Τι μας λέει ένα δέντρο, τι δεν μας λέει, και ποιες στήλες δεν πρέπει να του δώσουμε

Τι μας λέει

- ποιο attribute μείωσε περισσότερο την αβεβαιότητα σε αυτά τα δεδομένα
- κατώφλια που μπορούμε να ελέγξουμε πειραματικά
- κανόνες που συγκρίνουμε με τη γνώση μας

Τι δεν μας λέει

- αιτιότητα: ένα attribute στη ρίζα δεν είναι απαραίτητα η φυσική αιτία
- ότι ένα attribute που λείπει δεν επηρεάζει (βλ. Επαναφορά)
- τι συμβαίνει έξω από το εύρος των δεδομένων

Παγίδες attributes

ID	Κωδικός δοκιμίου, αριθμός παρτίδας, ημερομηνία: το πρόβλημα του «διαβατηρίου» (§4.3.2)
V ↑	Θερμοκρασία γραμμένη ως κατηγορία με 20 τιμές: ροπή του ID3 προς πολλά υποσύνολα
Λάθη	Λανθασμένες ή ελλείπουσες μετρήσεις → το δέντρο τις μαθαίνει (§4.4)
Λίγα	Λίγα δείγματα → τυχαία μοτίβα (§4.4)

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

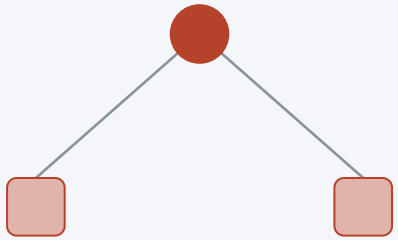
Πολύ μικρό, κατάλληλο, πολύ μεγάλο δέντρο

6 · Overfitting & pre-pruning

Underfitting και overfitting στα δέντρα απόφασης

Πολύ μικρό

underfitting

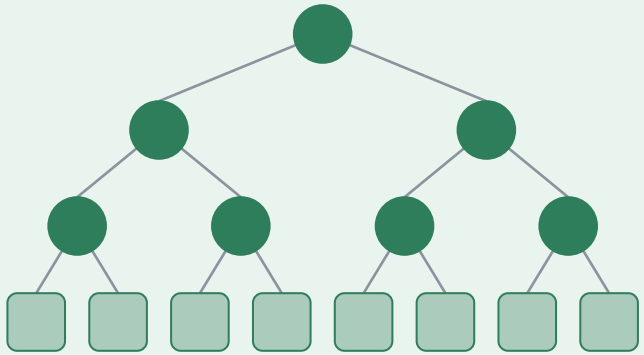


Δεν «πιάνει» σημαντική δομή του χώρου των δειγμάτων

training: χαμηλή · testing: χαμηλή

Κατάλληλο

καλή γενίκευση

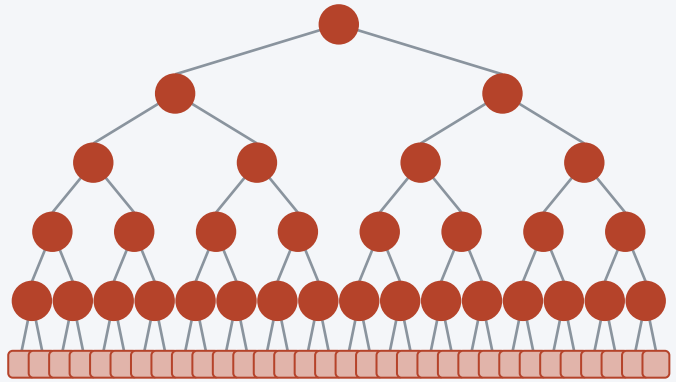


Μαθαίνει τα γενικά χαρακτηριστικά των δεδομένων

training: υψηλή · testing: υψηλή

Πολύ μεγάλο

overfitting



Φύλλα με ένα ή λίγα δείγματα: μαθαίνει και τον θόρυβο

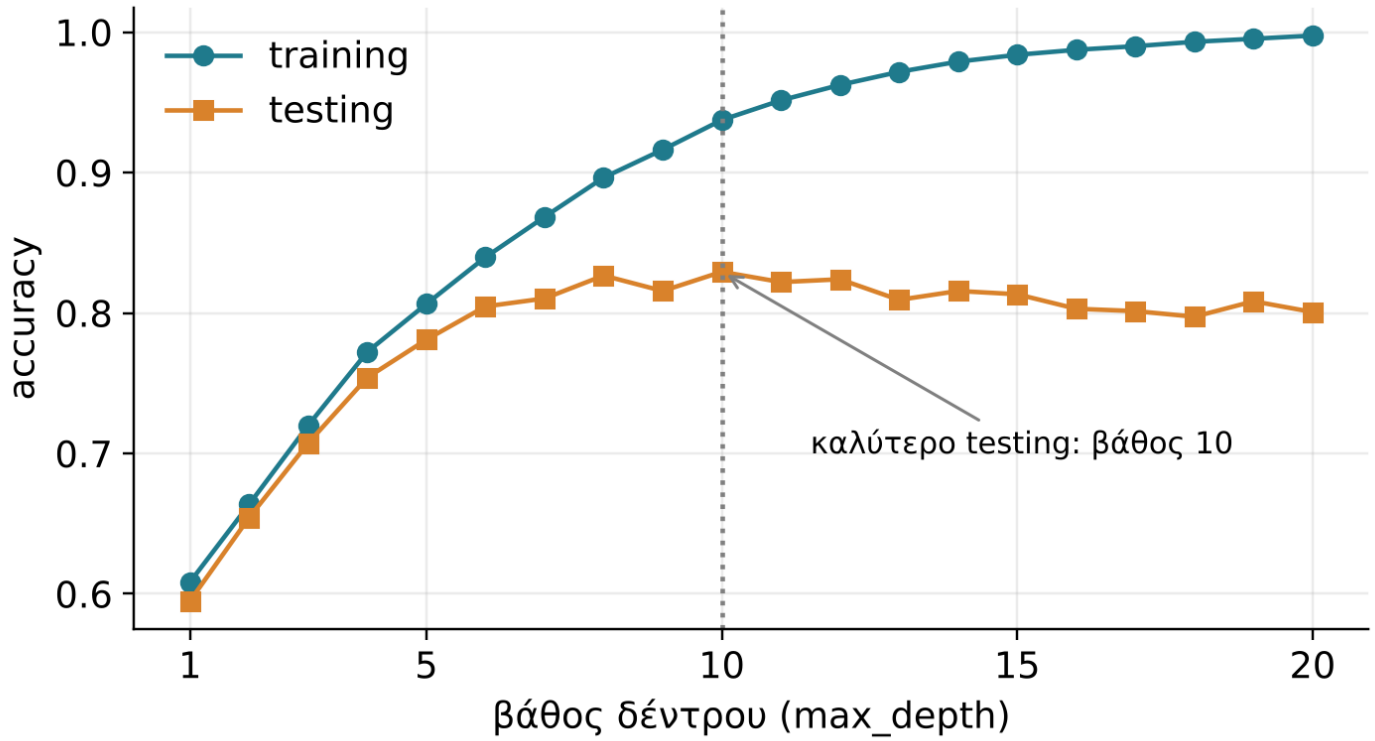
training: ≈ 100% · testing: πέφτει

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Βάθος δέντρου και γενίκευση

6 · Overfitting & pre-pruning

Η ακρίβεια στο training ανεβαίνει συνεχώς· στο testing κάποια στιγμή σταματά και πέφτει



Βάθος	Training	Testing	Φύλλα
1	0.608	0.595	2
4	0.772	0.754	16
10	**0.937**	**0.829**	**308**
απεριόριστο	1.000	0.791	704

ΚΥΡΙΟ ΜΗΝΥΜΑ

Το κενό training–testing μεγαλώνει με το βάθος: το δέντρο μαθαίνει χαρακτηριστικά ειδικά για τα δεδομένα εκπαίδευσης.

Η ακρίβεια σε training (μπλε) και testing (πορτοκαλί) καθώς μεγαλώνει το μέγιστο βάθος. Το training ανεβαίνει συνεχώς· το testing κορυφώνεται στο βάθος 10 και μετά πέφτει.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Τρεις αιτίες overfitting μέσα στα δεδομένα

6 · Overfitting & pre-pruning

Το δέντρο μαθαίνει ό,τι υπάρχει στα δεδομένα, και τα λάθη τους

1 · Λανθασμένες μετρήσεις

Η τιμή ενός attribute ή της κλάσης μετρήθηκε λάθος ή λείπει: λάθος αντίληψη, μέτρηση, καταγραφή ή μεταγραφή.

Στην Επιστήμη Υλικών

Σκληρόμετρο εκτός βαθμονόμησης· δοκίμιο με λάθος ετικέτα παρτίδας.

2 · Ελλιπή attributes

Τα attributes δεν είναι τέλειο κριτήριο ταξινόμησης· σημαντικοί εξωτερικοί παράγοντες μπορεί να λείπουν.

Στην Επιστήμη Υλικών

Δεν καταγράφηκε το μέγεθος κόκκου ή το πάχος του δοκιμίου.

3 · Σύμπτωση

Σε μικρό dataset τα δεδομένα μπορεί να σχηματίσουν τυχαία μοτίβα που δεν είναι γενικοί κανόνες.

Στην Επιστήμη Υλικών

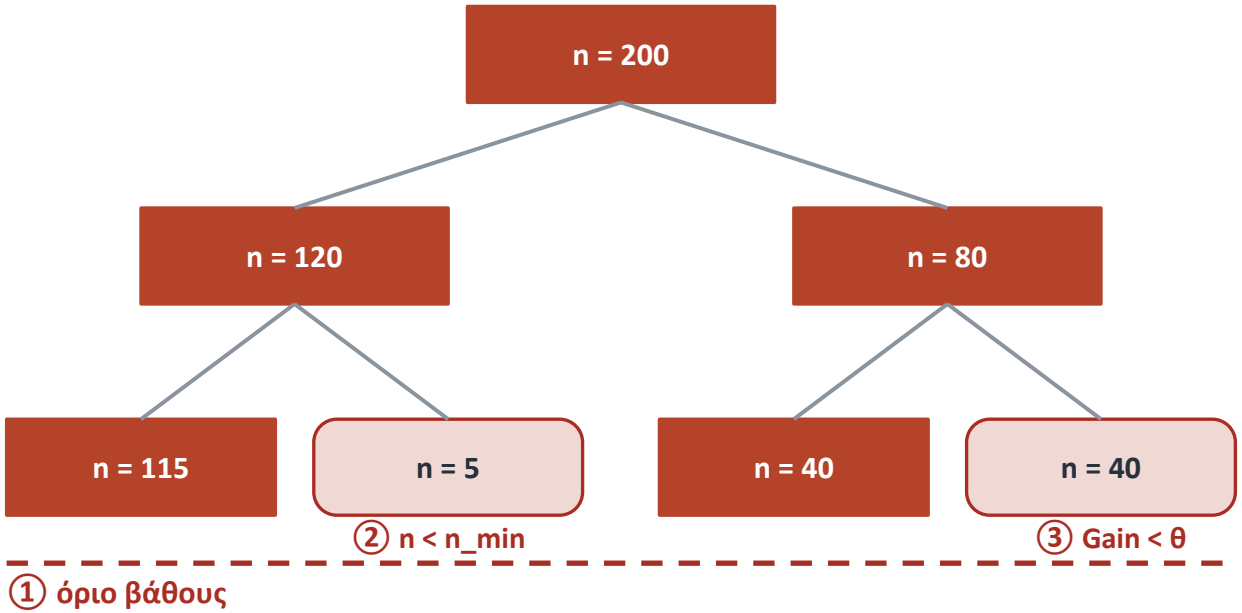
20 δοκίμια: «όσα μετρήθηκαν Τρίτη είναι σκληρά».

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Pre-pruning: σταματάμε την ανάπτυξη νωρίτερα

6 · Overfitting & pre-pruning

Παρόμοιο με το early stopping άλλων μεθόδων · τρεις τρόποι να ορίσουμε το «πότε»



① Ο πιο απλός: σταματάμε όταν το δέντρο φτάσει ένα μέγεθος, π.χ. ορισμένο αριθμό επιπέδων nodes

② Σταματάμε όταν τα δείγματα σε ένα node είναι λιγότερα από μια τιμή

③ Πιο ευρέως αποδεκτός: σταματάμε όταν το information gain της διάσπασης είναι κάτω από ένα κατώφλι

Ένα δέντρο που μεγαλώνει (n = δείγματα σε κάθε node) και τα τρία σημεία όπου το σταματά το pre-pruning. Τα nodes ② και ③ γίνονται φύλλα με την πλειοψηφούσα κλάση (σχήμα δικό μας).

ΚΥΡΙΟ ΜΗΝΥΜΑ

Pre-pruning: πριν τελειώσει η ανάπτυξη. Post-pruning: αφού έχουμε το δέντρο.

ΔΥΟ ΕΙΔΗ ΚΑΝΟΝΩΝ

Fixed κανόνες (χωρίς αξιολόγηση των δεδομένων) ή adaptive κανόνες (με ανάδραση από την ανάλυση/εκπαίδευση), Liu σ. 124.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Pre-pruning: πλεονεκτήματα, μειονεκτήματα, scikit-learn

6 · Overfitting & pre-pruning

Απλό και γρήγορο, αλλά «κοντόφθαλμο»

Πλεονεκτήματα

- σχετικά απλό στην υλοποίηση
- καλή αποδοτικότητα
- εφαρμόζεται σε προβλήματα μεγάλης κλίμακας

Μειονεκτήματα

- δύσκολο να βρεθεί κατάλληλος κανόνας
- η κατάσταση διαφέρει από node σε node
- «κοντόφθαλμο»: δεν ξέρει τα επόμενα βήματα

ΒΡΑΧΥΠΡΟΘΕΣΜΟ ΟΦΕΛΟΣ, ΜΑΚΡΟΠΡΟΘΕΣΜΟ ΚΟΣΤΟΣ

Μια διάσπαση με μικρό Gain τώρα μπορεί να ανοίγει δρόμο για πολύ χρήσιμες διασπάσεις στο επόμενο επίπεδο. Το pre-pruning τη χάνει.

Σύνδεση με scikit-learn

Κανόνας του βιβλίου	Υπερπαράμετρος DecisionTreeClassifier	Σημείωση
① μέγεθος / επίπεδα	max_depth, max_leaf_nodes	όριο βάθους ή πλήθους φύλλων
② λίγα δείγματα στο node	min_samples_split, min_samples_leaf	ελάχιστα δείγματα για διάσπαση / ανά φύλλο
③ μικρό information gain	min_impurity_decrease	ελάχιστη (σταθμισμένη) μείωση impurity

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Pre-pruning και post-pruning: πότε κόβουμε

7 · Post-pruning & practice

Πότε κόβουμε: κατά την ανάπτυξη ή αφού έχουμε ολόκληρο το δέντρο

Pre-pruning

Ρίζα
όλα τα δείγματα

→

Ανάπτυξη
σε κάθε node: έλεγχος κανόνα

→

STOP
όριο / λίγα δείγματα / μικρό Gain

→

Δέντρο
μικρότερο εξ αρχής

Post-pruning

Ρίζα
όλα τα δείγματα

→

Πλήρες δέντρο
μέχρι τους κανόνες φύλλου

→

Κλάδεμα
υποδέντρα → φύλλα

→

Δέντρο
απλούστερο

ΚΥΡΙΟ ΜΗΝΥΜΑ

Post-pruning: απλοποιεί το δέντρο κόβοντας τμήματα που μαθαίνουν ειδικά χαρακτηριστικά των δεδομένων εκπαίδευσης, τα οποία δεν ανήκουν στα γενικά δεδομένα.

Τέσσερις μέθοδοι στο κεφάλαιο

CCP | REP | PEP | MEP

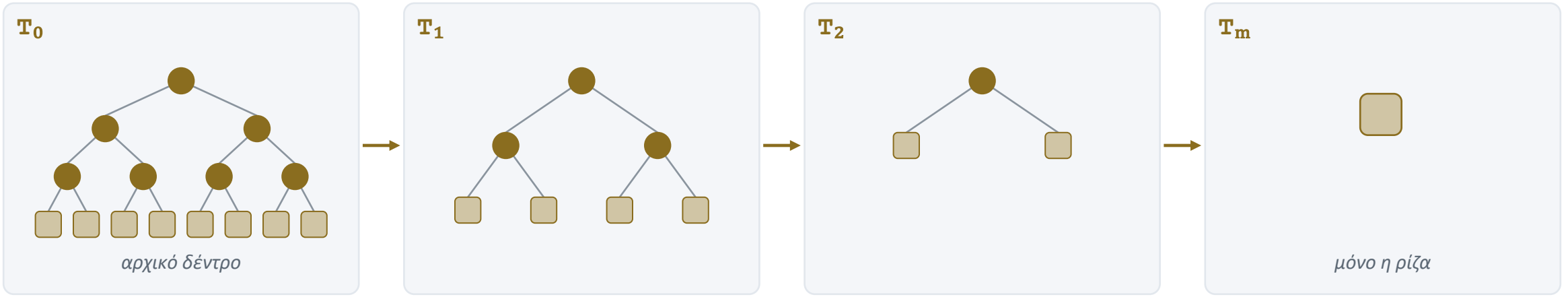
Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Cost-Complexity Pruning (CCP)

7 · Post-pruning & practice

Liu, §4.4.2 · σ. 126

Μια σειρά όλο και μικρότερων δέντρων T_0 … T_m · κρατάμε το καλύτερο



Κάθε δέντρο είναι το προηγούμενο με ένα υποδέντρο κομμένο σε φύλλο, μέχρι να μείνει μόνο η ρίζα. Ποιο υποδέντρο κόβεται; Αυτό που ελαχιστοποιεί.

$$\frac{err(\text{prune}(T,t),S) - err(T,S)}{|leaves(T)| - |leaves(\text{prune}(T,t))|}$$

αύξηση σφάλματος ανά φύλλο που αφαιρείται · err(T, S): σφάλμα του T στο σύνολο S

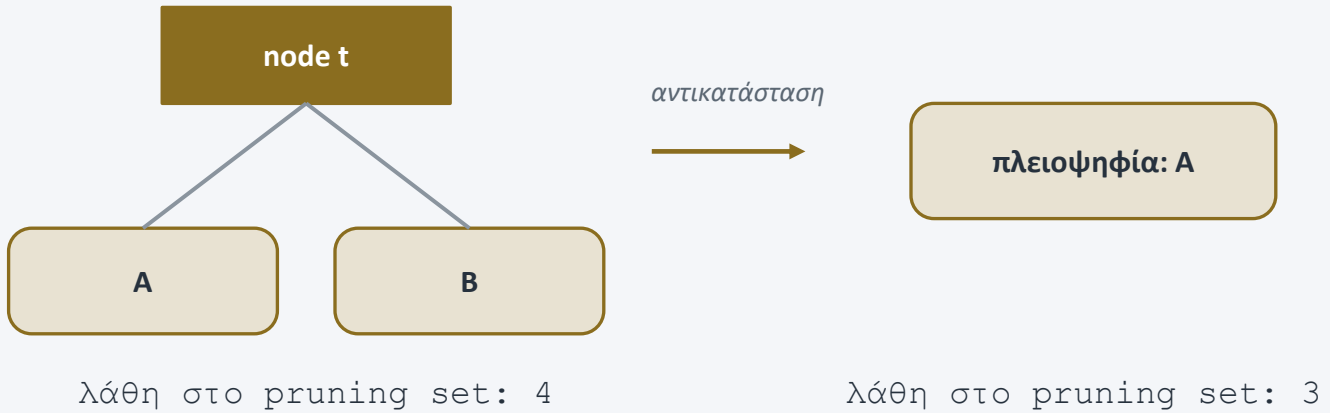
Επιλογή του τελικού δέντρου

Από την οικογένεια T_0 … T_m κρατάμε το δέντρο με την καλύτερη απόδοση, εκτιμώντας με cross-validation ή με ανεξάρτητο pruning set.

Reduced Error Pruning (REP)

7 · Post-pruning & practice

Αντικατέστησε ένα node με φύλλο αν το σφάλμα στο pruning set δεν χειροτερεύει



Τι δείχνει: αριστερά το υποδέντρο t, δεξιά το ίδιο σημείο αφού γίνει φύλλο. Το φύλλο κάνει 3 λάθη στο pruning set και το υποδέντρο 4: 3 ≤ 4 → κλάδεμα (ενδεικτικοί αριθμοί).

Ιδέα	κάθε μη-φύλλο node → φύλλο με την πιο συχνή κλάση
Κανόνας	κλάδεμα αν σφάλμα νέου δέντρου ≤ σφάλμα αρχικού
Σειρά	bottom-up (από τα φύλλα προς τη ρίζα)
Δεδομένα	χρειάζεται pruning (validation) set

Απλό και γρήγορο

Γραμμική πολυπλοκότητα: κάθε node επισκέπτεται μία φορά.

Over-pruning σε μικρά datasets

Αγνοεί ό,τι έδειξε το training set· χειρότερα όσο μικρότερο είναι το pruning set.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Pessimistic Error Pruning (PEP)

7 · Post-pruning & practice

Ίδιο training set για ανάπτυξη και κλάδεμα, με «απαισιόδοξη» διόρθωση του σφάλματος

Training set
το ίδιο για ανάπτυξη και κλάδεμα

→

Φαινομενικό σφάλμα
apparent error rate: αισιόδοξο

→

Continuity correction
για τη διωνυμική κατανομή

→

«Πιο ρεαλιστικό» σφάλμα
βάση για το κλάδεμα

Υπέρ

- μία από τις πιο ακριβείς μεθόδους
- top-down: κάθε υποδέντρο εξετάζεται μία φορά
- χειρότερη περίπτωση: γραμμική στα μη-φύλλα nodes

Κατά

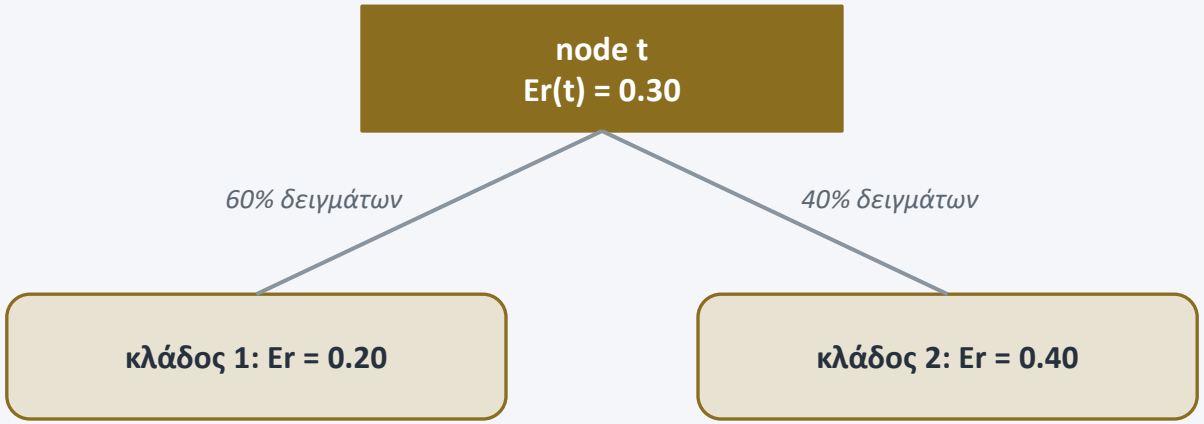
- τάση για over-pruning
- η διάσχιση μοιάζει με του pre-pruning
- δεν χρειάζεται όμως επιπλέον δεδομένα

ΓΙΑΤΙ ΤΟ ΣΦΑΛΜΑ ΕΚΠΑΙΔΕΥΣΗΣ ΕΙΝΑΙ ΑΙΣΙΟΔΟΞΟ
Ένα φύλλο με ένα δείγμα έχει πάντα σφάλμα 0 στα δεδομένα εκπαίδευσης. Αν κρίναμε με αυτό το σφάλμα, το πλήρες δέντρο θα φαινόταν πάντα καλύτερο.

Minimum Error Pruning (MEP)

7 · Post-pruning & practice

Κρατάμε το υποδέντρο μόνο αν το σταθμισμένο σφάλμα των κλάδων του είναι μικρότερο



$$Er(T_t) = 0.6 \cdot 0.20 + 0.4 \cdot 0.40 = 0.28$$

0.28 < 0.30 → κρατάμε το υποδέντρο

Το node t και οι δύο κλάδοι του, με το σφάλμα και το ποσοστό δειγμάτων καθενός. Το σταθμισμένο σφάλμα των κλάδων συγκρίνεται με το σφάλμα του t (ενδεικτικοί αριθμοί).

Στόχος	ένα δέντρο με το ελάχιστο «αναμενόμενο σφάλμα σε ανεξάρτητο dataset»
$Er(T_t) < Er(t)$	κρατάμε το υποδέντρο
αλλιώς	το κόβουμε (το t γίνεται φύλλο)
Σειρά	bottom-up

ΣΥΜΒΟΥΛΗ

Δεν χρειάζεται ανεξάρτητο pruning set· η πληροφορία έρχεται κυρίως από το training set. Γραμμική πολυπλοκότητα στα μη-φύλλα nodes. Er: εκτίμηση πιθανότητας με παράμετρο m (Table 4.1).

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Σύγκριση μεθόδων post-pruning

Χαρακτηριστικό	CCP	REP	PEP	MEP
Ανεξάρτητο pruning set	Δεν χρειάζεται	**Χρειάζεται**	Δεν χρειάζεται	Δεν χρειάζεται
Κατεύθυνση	bottom-up	bottom-up	top-down	bottom-up
Εκτίμηση σφάλματος	cross-validation ή standard error	pruning set	continuity correction	εκτίμηση πιθανότητας με παράμετρο m
Πολυπλοκότητα	$O(n^2)$	$O(n)$	$O(n)$	$O(n)$

Γενικοί κανόνες (Liu, σσ. 127–128)

MEP < PEP

Το MEP είναι γενικά λιγότερο ακριβές από το PEP

Μικρά datasets

PEP και MEP δεν θέλουν επιπλέον δεδομένα

REP

απλό και ακριβές, αλλά θέλει επιπλέον δεδομένα

CCP ≈ REP

συγκρίσιμη ακρίβεια

Δυνατά σημεία και περιορισμοί

8 · Σύνθεση

Liu, §4.2–4.4 · σσ. 116–127

Τι λέει το κεφάλαιο και τι προσθέτουμε (σημειωμένο ως επέκταση)

Δυνατά σημεία

Liu	Μιμείται την ανθρώπινη λήψη αποφάσεων (§4.1)
Liu	Κάθε path = σειρά συνθηκών → απόφαση: διαβάζεται ως κανόνας (§4.2)
Liu	Σαφή κριτήρια από τη θεωρία πληροφορίας (§4.2–4.3)
Επέκταση	Πολύ γρήγορη πρόβλεψη· δεν χρειάζεται κλιμάκωση attributes
Επέκταση	Δέχεται κατηγορικά και αριθμητικά attributes

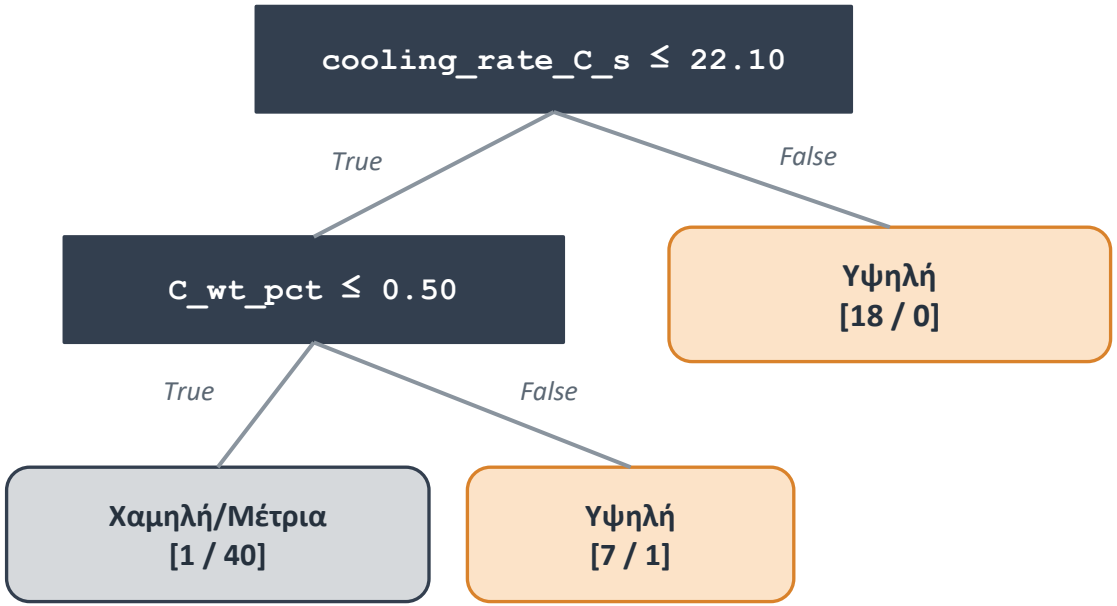
Περιορισμοί

Liu	Ευάλωτο σε overfitting, ιδιαίτερα λόγω της φύσης του (§4.4)
Liu	Ροπές κριτηρίων: ID3 → πολλές τιμές, gain ratio → λίγες (§4.3.2)
Liu	Δύσκολη επιλογή κανόνα pre-pruning· κοντόφθαλμο (§4.4.1)
Επέκταση	Αστάθεια: μικρή αλλαγή στα δεδομένα → διαφορετικό δέντρο
Επέκταση	Κλιμακωτές προβλέψεις· δεν κάνει extrapolation

Case study: κατηγορία ορίου διαρροής χάλυβα

8 · Σύνθεση

Συνθετικό διδακτικό dataset (96 δείγματα) · όριο διαρροής ≥ 550 MPa → «Υψηλή»



Το δέντρο βάθους 2 (max_depth = 2). Σε κάθε φύλλο [Υψηλή / Χαμηλή-Μέτρια] στα 67 δείγματα εκπαίδευσης· πορτοκαλί = πρόβλεψη «Υψηλή». Attributes: C_wt_pct, heat_treatment_C, cooling_rate_C_s, route_QT · split 70/30 stratified.

Μοντέλο	Train	Test	Φύλλα
max_depth = 1	0.881	0.793	2
max_depth = 2	**0.970**	**0.828**	**3**
max_depth = 3	0.985	0.793	5
πλήρες δέντρο	1.000	0.793	6

ccp_alpha	0	0.015	0.026	0.145	0.275
Φύλλα	6	4	**3**	2	1
CV acc.	0.924	0.924	**0.924**	0.865	0.718

ΠΡΟΣΟΧΗ

Test set = 29 δείγματα: 0.828 έναντι 0.793 είναι διαφορά ενός δείγματος. Τα route_QT και heat_treatment_C δεν χρησιμοποιήθηκαν.

ΚΥΡΙΟ ΜΗΝΥΜΑ

Pre-pruning (max_depth = 2) και post-pruning (CCP, α = 0.026) δίνουν εδώ το ίδιο δέντρο 3 φύλλων.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.

Έλεγχος κατανόησης: έξι τυπικές περιπτώσεις

8 · Σύνθεση

Κάθε περίπτωση με τη λύση της, με τα εργαλεία του κεφαλαίου

1 Node με 8 δείγματα, όλα της ίδιας κλάσης

H = 0 και Gini = 0 (P = 1 για μία κλάση). Το node γίνεται φύλλο με τον κανόνα 1 (σ. 116).

2 Κωδικός δοκιμίου ως attribute

Ένα δείγμα ανά υποσύνολο → Gain = H(D), το μέγιστο, άρα το ID3 τον διαλέγει. Το C4.5 τον «τιμωρεί» με την IV = $\log_2 N$ (εδώ 0.265 έναντι 0.396 της Ψύξης).

3 Gini_index: A = 0.24, B = 0.30

Το CART διαλέγει την A: μικρότερο Gini_index σημαίνει λιγότερη αβεβαιότητα μετά τη διάσπαση.

4 Training 100%, testing 70%

Overfitting. Λύσεις: pre-pruning (max_depth, min_samples_leaf) ή CCP, με επιλογή μέσω cross-validation· έλεγχος δεδομένων για λάθη και στήλες τύπου ID.

5 Μόνο 40 δοκίμια διαθέσιμα

Προτιμάμε PEP (ή MEP): δεν χρειάζεται ξεχωριστό pruning set. Το REP θέλει επιπλέον δεδομένα και κάνει over-pruning με μικρό pruning set (Table 4.1).

6 Η Επαναφορά λείπει από το δέντρο

Δεν σημαίνει ότι δεν επηρεάζει τη σκληρότητα· σε αυτά τα 10 δείγματα δεν πρόσθετε πληροφορία, αφού ήταν γνωστά η ψύξη και ο άνθρακας.

Σύνοψη: από τα δεδομένα στο κλαδεμένο δέντρο

8 · Σύνθεση

Liu, Chapter 4 · σσ. 115–128

Η αλυσίδα του Κεφαλαίου 4 σε έξι κρίκους



Μία ιδέα

Σε κάθε node διαλέγουμε το attribute που μειώνει περισσότερο την αβεβαιότητα, σταθμισμένα.

Τρία κριτήρια

max Gain (ID3) · Gain > μέσος, μετά max ratio (C4.5) · min Gini_index (CART).

Ένας κίνδυνος

Ένα δέντρο που μεγαλώνει ελεύθερα μαθαίνει τον θόρυβο· το pruning το ελέγχει.

Για εξάσκηση: επαληθεύστε το worked example σε Python · Iris (§4.5) · δοκιμάστε max_depth και ccp_alpha

Ανακεφαλαίωση και επόμενο μάθημα

Ανακεφαλαίωση · Μάθημα 4

Τι κρατάμε από σήμερα και τι ακολουθεί

1

Μία ιδέα: το δέντρο είναι μια σειρά ερωτήσεων

Κάθε διάσπαση επιλέγει ένα attribute για υψηλότερη καθαρότητα. Κάθε path από τη ρίζα σε φύλλο είναι ένας κανόνας «αν… τότε…».

2

Τρία κριτήρια, τρεις αλγόριθμοι

ID3: information gain · C4.5: gain ratio · CART: Gini index. Διαφορετικό κριτήριο μπορεί να διαλέξει διαφορετικό attribute.

3

Ένας κίνδυνος: το overfitting

Ένα δέντρο που μεγαλώνει ελεύθερα απομνημονεύει τα δεδομένα. Το περιορίζουμε με pre-pruning (π.χ. max_depth) και post-pruning (CCP, REP, PEP, MEP).

ΕΠΟΜΕΝΟ ΜΑΘΗΜΑ

Μάθημα 5 · Python για Machine Learning I

Workshop με live coding και εργαστήριο

- NumPy: πίνακες χωρίς βρόχους
- Pandas: CSV, επιλογή, καθαρισμός
- Matplotlib: γραφήματα και EDA
- Πρώτο μοντέλο: fit, predict, MAE

Θα εκπαιδεύσουμε ένα δέντρο απόφασης στο scikit-learn και θα δούμε το overfitting στον κώδικα.

Βασική Πηγή: Liu, Z. (2025). Artificial Intelligence for Engineers.