Εξόρυξη Δεδομένων

5: Κατηγοριοποίηση

Περιεχόμενα

- Γενικά
- Σύνοψη μεθόδων κατηγοριοποίησης
- Δέντρα απόφασης
- Αλγόριθμοι
- Πληροφοριακό κέρδος και Εντροπία
- Παραδείγματα

Γενικά

Το γενικό πρόβλημα της ανάθεσης ενός αντικειμένου σε μια ή περισσότερες *προκαθορισμένες* κατηγορίες (κλάσεις)

Παραδείγματα

- Εντοπισμός spam emails, με βάση πχ την επικεφαλίδα τους ή το περιεχόμενό τους
- Πρόβλεψη καρκινικών κυττάρων χαρακτηρίζοντας τα ως καλοήθη ή κακοήθη
- Κατηγοριοποίηση συναλλαγών με πιστωτικές κάρτες ως νόμιμες ή προϊόν απάτης
- Κατηγοριοποίηση δευτερευόντων δομών πρωτείνης ως alpha-helix, beta-sheet, ή random coil
- Χαρακτηρισμός ειδήσεων ως οικονομικές, αθλητικές, πολιτιστικές, πρόβλεψης καιρού, κλπ

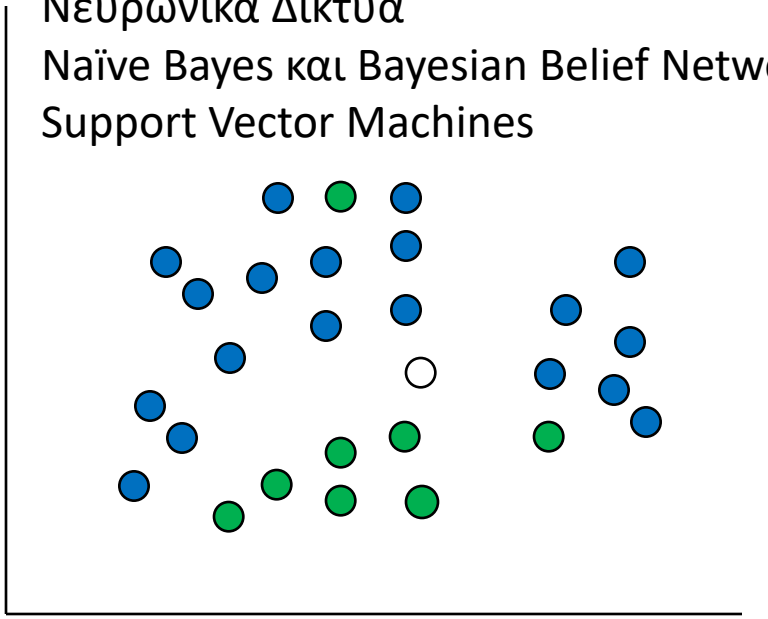
Γενικά

- Είναι εποπτευόμενη μέθοδος μάθησης
  - Τα δεδομένα εκπαίδευσης συνοδεύονται από πληροφορία που υποδεικνύει την κατηγορία (κλάση) στην οποία ανήκουν
  - Ένα νέο μοντέλο εκπαιδεύεται με τα δεδομένα αυτά
  - Νέα δεδομένα κατηγοριοποιούνται με βάση το μοντέλο που εκπαιδεύτηκε
- Δεν προβλέπει άγνωστες και ελλιπείς τιμές καθώς δε μοντελοποιεί το πρόβλημα με μια συνάρτηση (όπως κάνει η παλινδρόμηση)

Με μια ματιά

Τεχνικές

Decision Trees, Regression,
Κανόνες (Rule-based Methods)
Αλγόριθμοι Κοντινότερου Γείτονα
Νευρωνικά Δίκτυα
Naïve Bayes και Bayesian Belief Networks
Support Vector Machines



Εφαρμογές

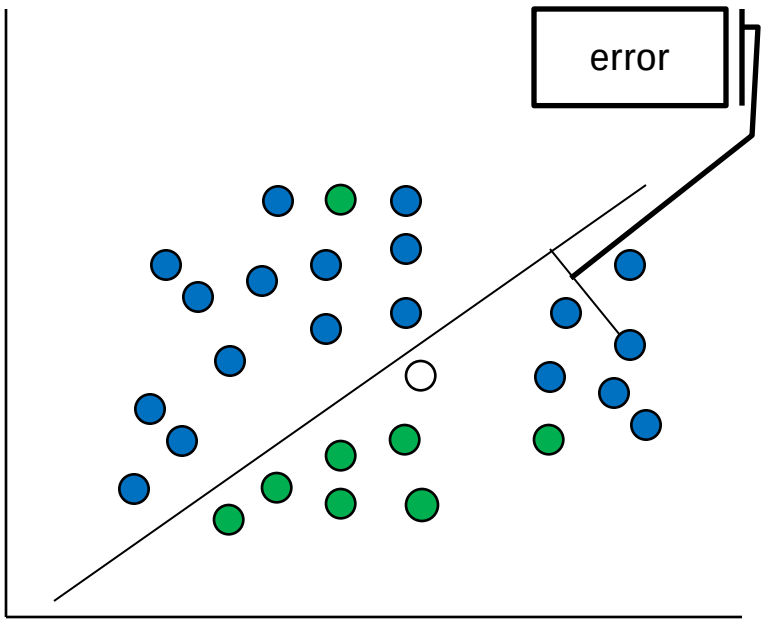
Credit approval
Target marketing
Medical diagnosis
Fraud detection

Είσοδος

Ένα καλά ορισμένο σύνολο από κατηγορίες
Ένα σύνολο εκπαίδευσης από προκαθορισμένα παραδείγματα γνωστής κατηγορίας

Με δεδομένο ότι κάποια σημεία ανήκουν στις κλάσεις ● ●
Ποια είναι η κλάση του νέου σημείου ○

Classification: Linear Regression

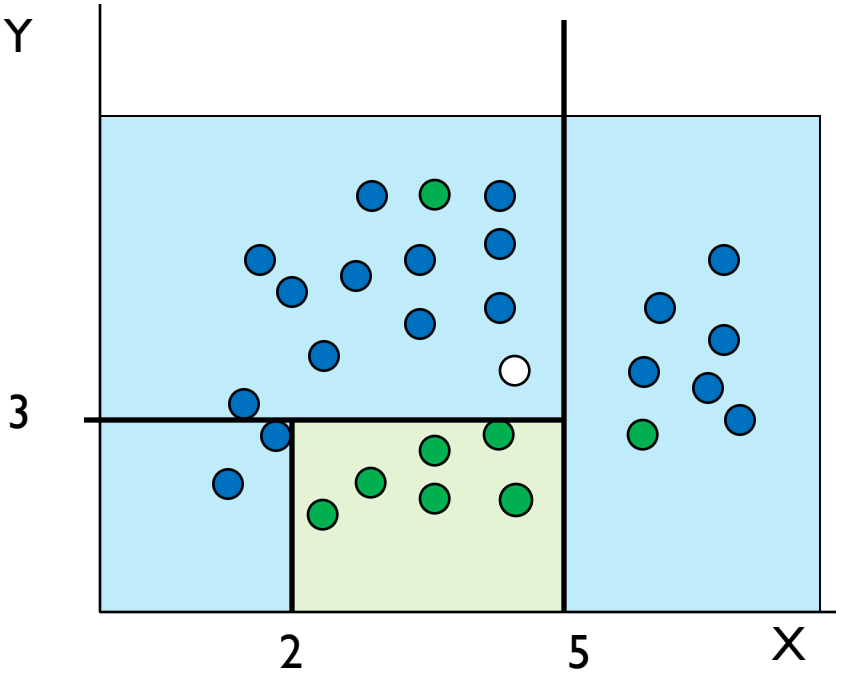


- Linear Regression

$$y=\beta x+\varepsilon$$

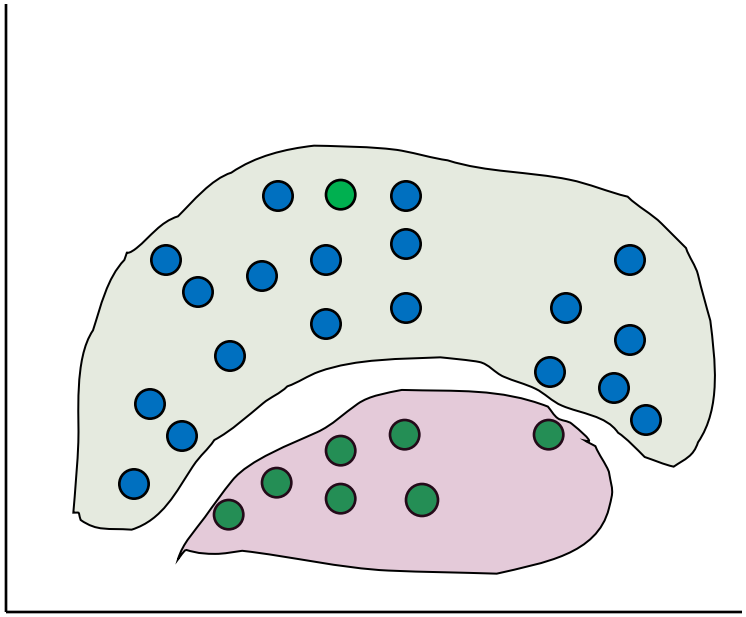
- Υπολογίζω τα β,ε από τα δεδομένα έτσι ώστε να ελαχιστοποιώ το squared error της καμπύλης της συνάρτησης ως προς τα δεδομένα
- Δεν είναι ευέλικτη μέθοδος καθώς δεν ταιριάζει σε πολλά φυσικά προβλήματα
- Είναι ευαίσθητη στην παρουσία outliers

Classification: Decision Trees



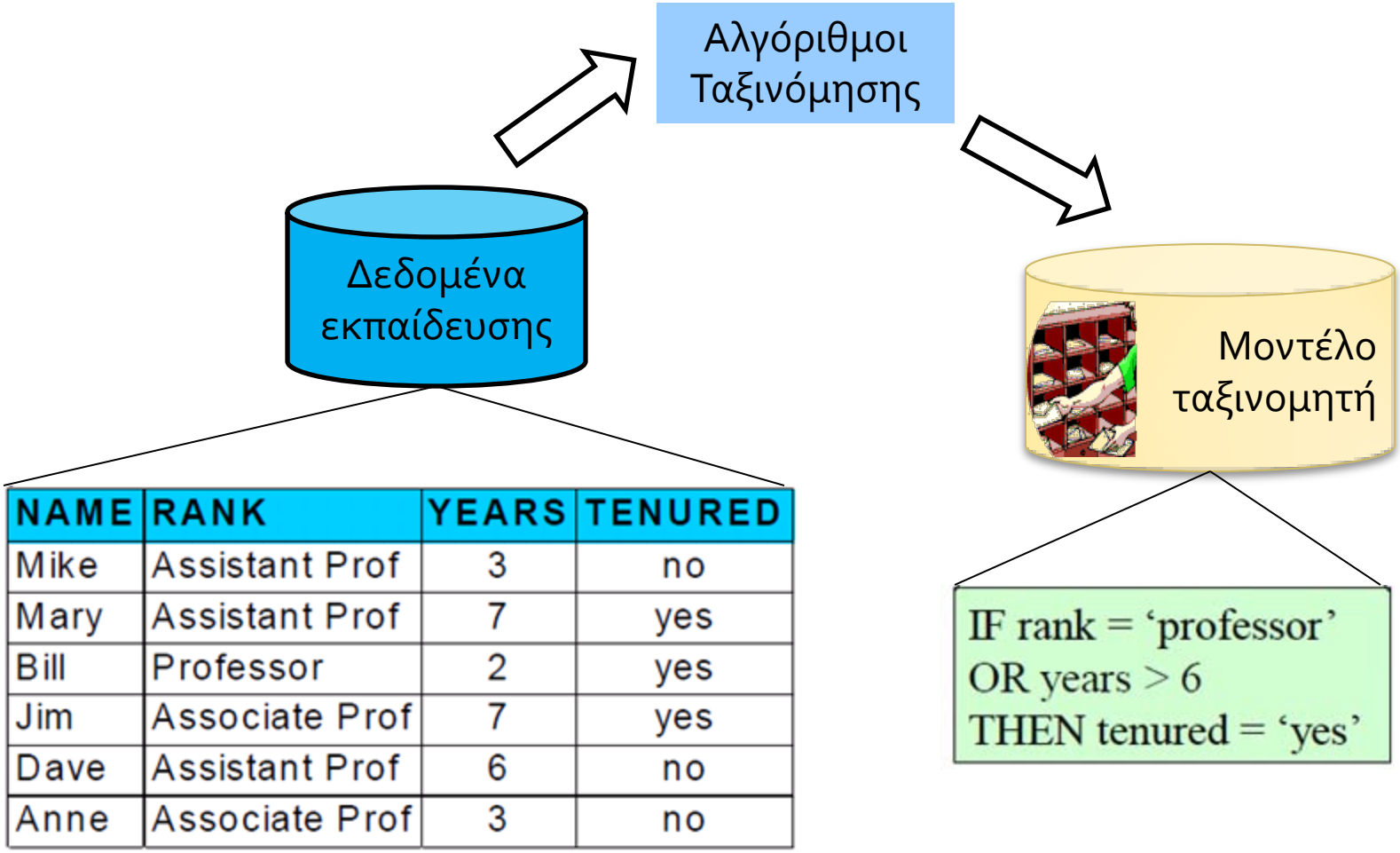
if X > 5 then blue
else if Y > 3 then blue
else if X > 2 then green
else blue

Classification: Neural Nets



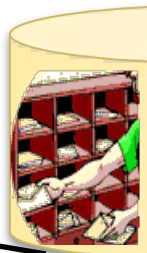
- Επιλέγουν πιο σύνθετες περιοχές
- Είναι πιο ακριβείς
- Μπορεί να υπερ-εκπαιδευτούν (overfit) και αν βρουν πρότυπα μέσα στο θόρυβο

Βήμα 1: Κατασκευή μοντέλου



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

Βήμα 2: Αξιολόγηση μοντέλου



Μοντέλο ταξινομητή

Δεδομένα ελέγχου

Θεωρώ άγνωστο

Merlisa	Associate Prof	7	?

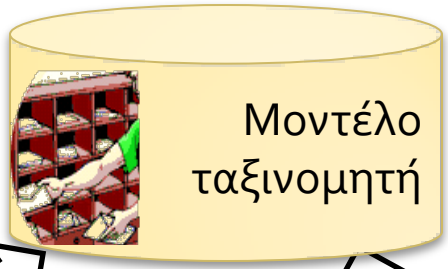
Πρόβλεψη

YES

Επαλήθευση

NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

Βήμα 3: Χρήση μοντέλου



Μοντέλο ταξινομητή

Άγνωστα δεδομένα

George	Assistant Prof	5	

Πρόβλεψη

YES

NAME	RANK	YEARS	TENURED
George	Assistant Prof	5	
Joseph	Professor	2	

Αξιολόγηση μεθόδων κατηγοριοποίησης

- Κριτήρια
  - Ακρίβεια πρόβλεψης (prediction accuracy)
  - Ταχύτητα (speed): Χρόνος κατασκευής του μοντέλου, Χρόνος εκτέλεσης του μοντέλου
  - Ευρωστία (robustness): Διαχείριση θορύβου, ελλιπών τιμών
  - Κλιμάκωση (scalability): Αποδοτικότητα διαχείρισης μεγάλων ΒΔ
  - Ικανότητα ερμηνείας αποτελεσμάτων (interpretability)
  - Ποιότητα κανόνων: Μέγεθος δέντρου αποφάσεων, πόσο συμπαγείς είναι οι κανόνες κατηγοριοποίησης

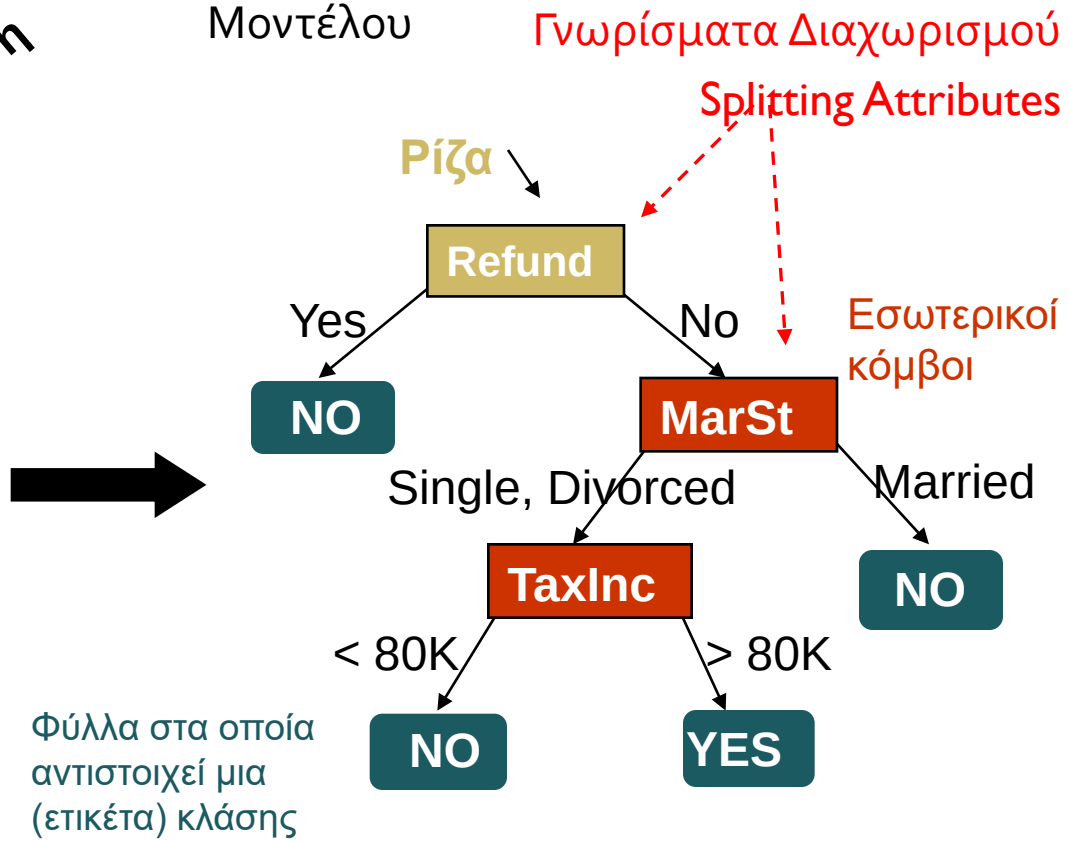
Δέντρα Απόφασης

Παράδειγμα

Δεδομένα Εκπαίδευσης

Tid	Refund (κατηγορικό)	Marital Status (κατηγορικό)	Taxable Income (συνεχές)	Cheat (κλάση)
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Παράδειγμα Μοντέλου



Μοντέλο: Δέντρο Απόφασης

Παράδειγμα

Μοντέλο = Δέντρο Απόφασης

- Εσωτερικοί κόμβοι αντιστοιχούν σε κάποιο γνώρισμα

- Διαχωρισμός (split) ενός κόμβου σε παιδιά
  - η ετικέτα στην ακμή = συνθήκη/έλεγχος

- Φύλλα αντιστοιχούν σε κλάσεις

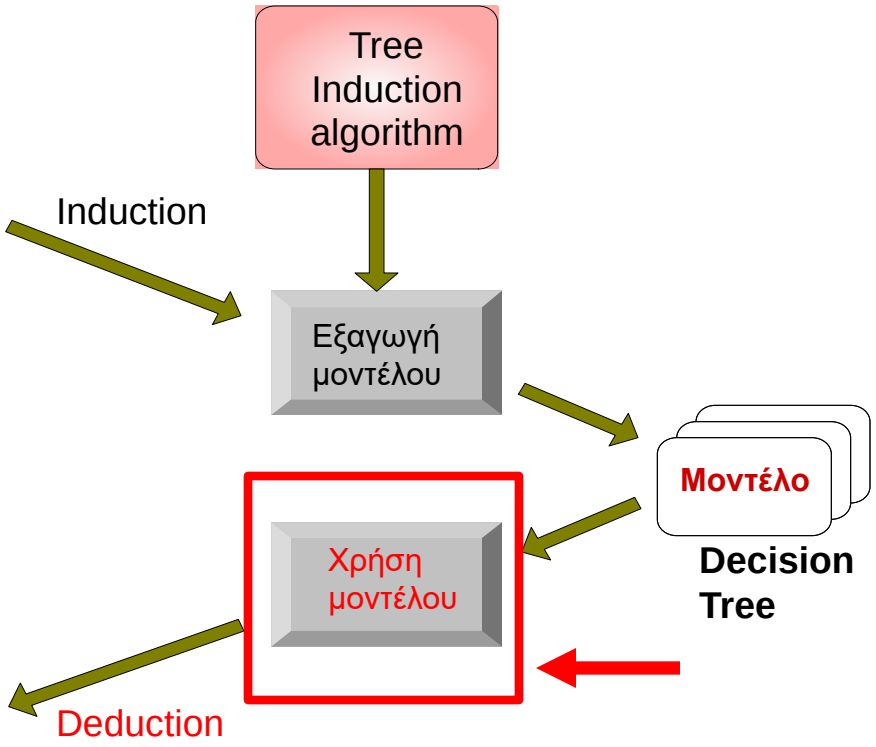
Δέντρο Απόφασης

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Set

Tid	Refund	Marital Status	Taxable Income	Cheat
11	No	Single	55K	?
12	Yes	Married	80K	?
13	Yes	Single	110K	?
14	No	Married	95K	?
15	No	Divorced	67K	?

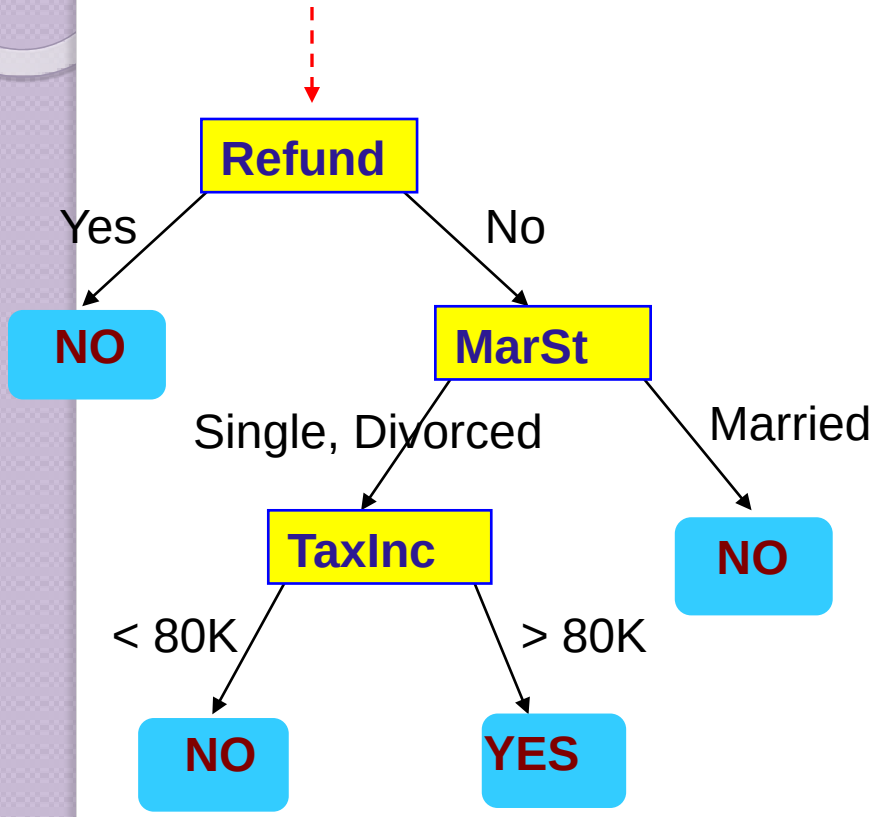
Test Set



Αφού κατασκευαστεί το δέντρο, η εφαρμογή (χρήση) του στην ταξινόμηση νέων εγγραφών είναι απλή -> διαπέραση από τη ρίζα στα φύλλα του

Εφαρμογή Μοντέλου

Ξεκίνα από τη ρίζα του δέντρου.



Δεδομένα Ελέγχου

Tid	Refund	Marital Status	Taxable Income	Cheat
11	No	Single	55K	?
12	Yes	Married	80K	?
13	Yes	Single	110K	?
14	No	Married	95K	?
15	No	Divorced	67K	?

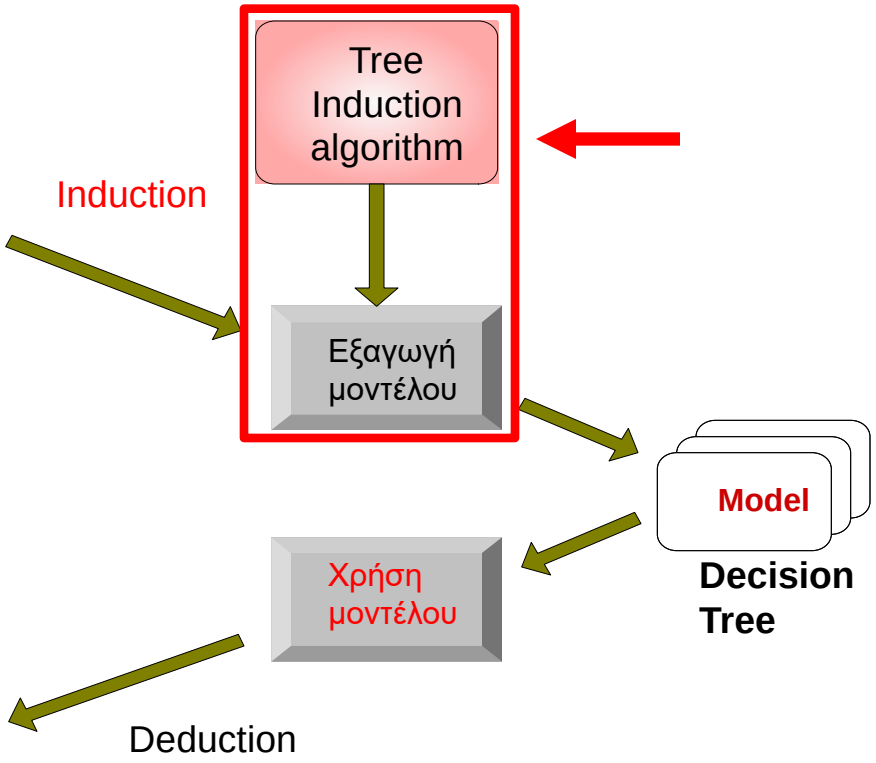
Δέντρο Απόφασης

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Set

Tid	Refund	Marital Status	Taxable Income	Cheat
11	No	Single	55K	?
12	Yes	Married	80K	?
13	Yes	Single	110K	?
14	No	Married	95K	?
15	No	Divorced	67K	?

Test Set



Θα δούμε πως θα το κατασκευάσουμε

Υπενθύμιση – Είσοδος μας είναι το σύνολο εκπαίδευσης

Κατασκευή δέντρου απόφασης

Κατασκευή του δέντρου (με λίγα λόγια):

1. ξεκίνα με *έναν* κόμβο που περιέχει *όλες* τις εγγραφές

2. *διάσπαση* του κόμβου (μοίρασμα των εγγραφών) με βάση μια συνθήκη-διαχωρισμού σε κάποιο από τα γνωρίσματα

3. Αναδρομική κλήση του 2 σε κάθε κόμβο (top-down, recursive, divide-and-conquer προσέγγιση)

4. Αφού κατασκευαστεί το δέντρο, κάποιες βελτιστοποιήσεις (tree pruning)

Το βασικό θέμα είναι

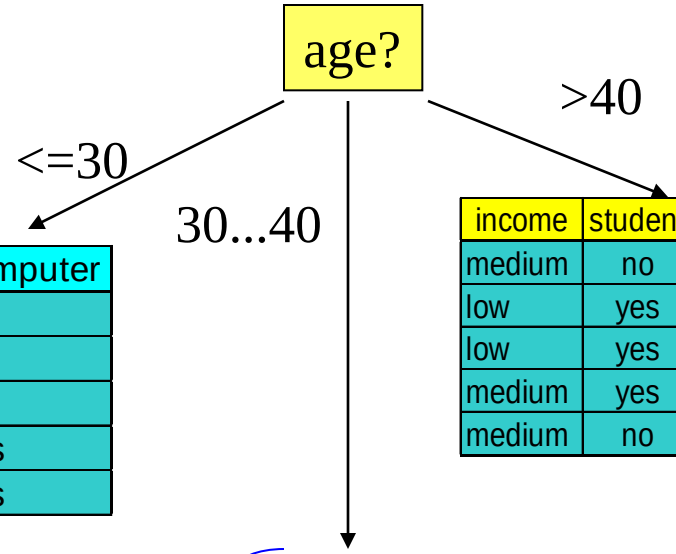
Ποιο γνώρισμα-συνθήκη διαχωρισμού να χρησιμοποιήσουμε για τη διάσπαση των εγγραφών κάθε κόμβου;

κλάση

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Διάσπαση

-Με ποια σειρά επιλέγουμε τα γνωρίσματα;
-Πώς θέτουμε τα όρια;
Στόχος: να διαχωρίσουμε το αρχικό σύνολο σε συμπαγείς και πολυπληθείς ομάδες (όλα τα στοιχεία να ανήκουν στην ίδια κλάση)



income	student	credit_rating	buys_computer
high	no	fair	no
high	no	excellent	no
medium	no	fair	no
low	yes	fair	yes
medium	yes	excellent	yes

income	student	credit_rating	buys_computer
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

income	student	credit_rating	buys_computer
high	no	fair	yes
low	yes	excellent	yes
medium	no	excellent	yes
high	yes	fair	yes

φύλο με ετικέτα yes

Για το ίδιο σύνολο εκπαίδευσης υπάρχουν διαφορετικά δέντρα

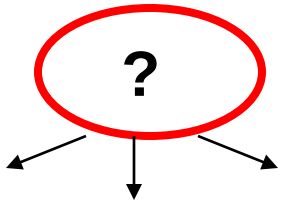
Αλγόριθμοι

Ο αριθμός των πιθανών Δέντρων Απόφασης έχει εκθετική πολυπλοκότητα.

Πολλοί αλγόριθμοι για την **επαγωγή (induction)** του δέντρου οι οποίοι ακολουθούν μια άπληστη στρατηγική: για να κτίσουν το δέντρο απόφασης παίρνοντας μια σειρά από *τοπικά βέλτιστες* αποφάσεις

- Hunt's Algorithm (από τους πρώτους)
- ID3, C4.5
- CART
- SLIQ, SPRINT

Αλγόριθμος του Hunt



Κτίζει το δέντρο αναδρομικά, αρχικά όλες οι εγγραφές σε έναν κόμβο (ρίζα)

$\mathbf{D_t}$: το σύνολο των εγγραφών εκπαίδευσης που έχουν φτάσει στον κόμβο t

Γενική Διαδικασία (αναδρομικά σε κάθε κόμβο)

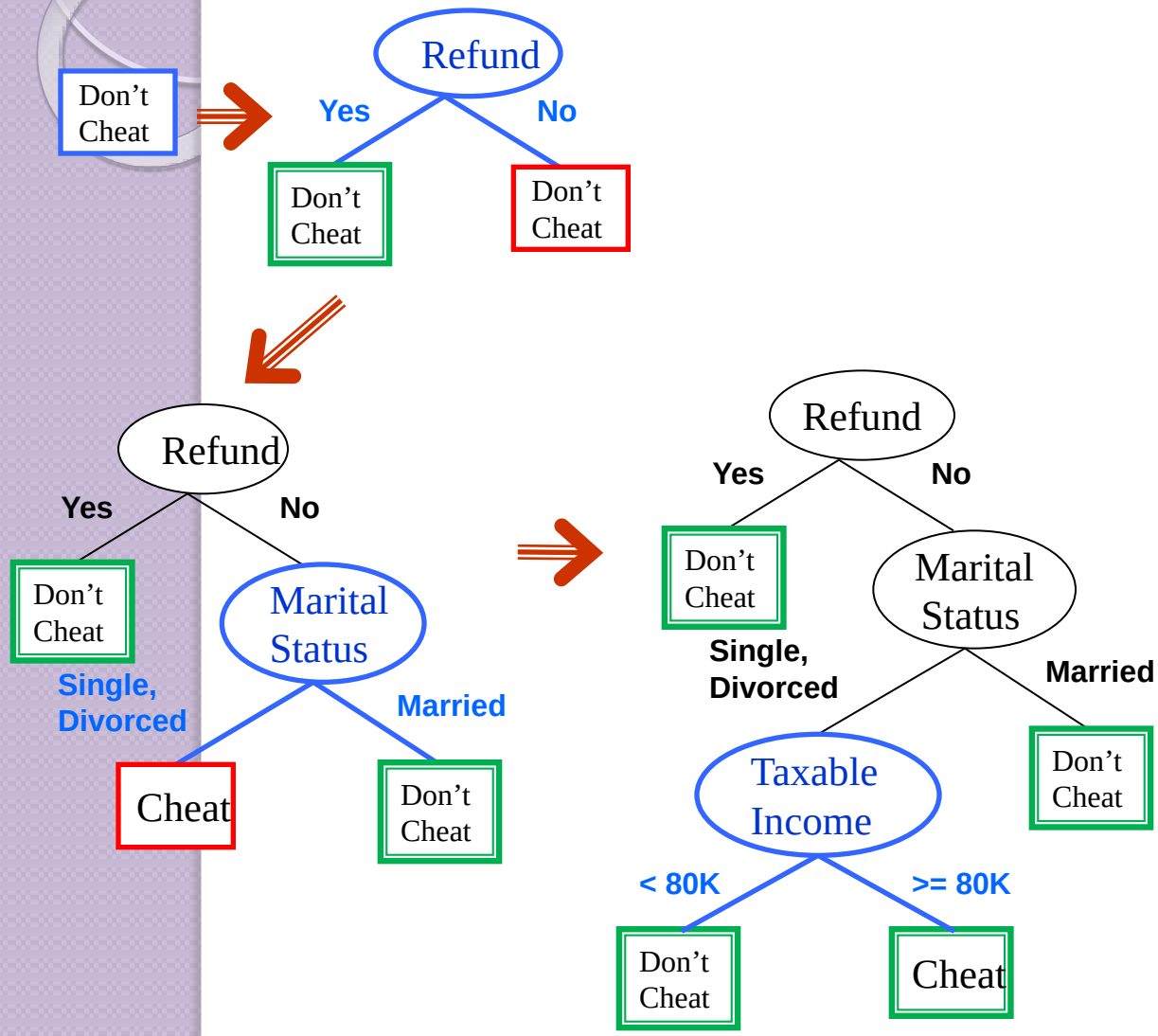
- Αν το D_t *περιέχει εγγραφές που ανήκουν στην ίδια κλάση* y_t,
  τότε ο κόμβος t είναι κόμβος φύλλο με ετικέτα y_t

- Αν D_t είναι το *κενό σύνολο* (αυτό σημαίνει ότι δεν υπάρχει εγγραφή στο σύνολο εκπαίδευσης με αυτό το συνδυασμό τιμών),
  τότε D_t γίνεται φύλλο με κλάση αυτή *της πλειοψηφίας* των εγγραφών εκπαίδευσης (συνολικά) ή ανάθεση κάποιας default κλάσης

- Αν το D_t *περιέχει εγγραφές που ανήκουν σε περισσότερες από μία κλάσεις*,
  τότε χρησιμοποίησε έναν έλεγχο-γνωρίσματος για το διαχωρισμό των δεδομένων σε μικρότερα υποσύνολα

Σημείωση: ο διαχωρισμός δεν είναι δυνατός αν όλες οι εγγραφές έχουν τις ίδιες τιμές σε όλα τα γνωρίσματα (δηλαδή, ο ίδιος συνδυασμός αντιστοιχεί σε περισσότερες από μία κλάσεις) τότε φύλλο με κλάση αυτής της πλειοψηφίας των εγγραφών εκπαίδευσης

Αλγόριθμος του Hunt



Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Ψευδοκώδικας

Algorithm GenDecTree(Sample S, Attlist A)

1. create a node N
2. If all samples are of the same class C then label N with C; terminate;
3. If A is empty then label N with the most common class C in S (majority voting); terminate;
4. Select a $\in$ A, with the highest gain; Label N with a;
5. For each value v of a:
   a. Grow a branch from N with condition a=v;
   b. Let S_v be the subset of samples in S with a=v;
   c. If S_v is empty then attach a leaf labeled with the most common class in S;
   d. Else attach the node generated by GenDecTree(S_v, A-a)

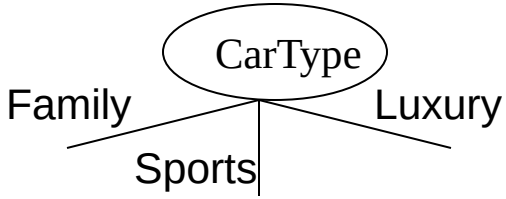
Κατασκευή δέντρου απόφασης

- Θέματα
  1. Καθορισμός του τρόπου διαχωρισμού των εγγραφών
     - Καθορισμός του ελέγχου γνωρίσματος
     - Ποιος είναι ο βέλτιστος διαχωρισμός
  2. Πότε θα σταματήσει ο διαχωρισμός (συνθήκη τερματισμού)
- Άπληστη στρατηγική διαχωρισμού

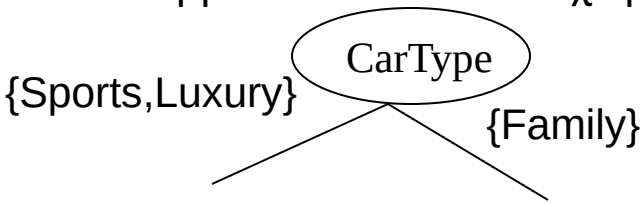
  Διαχωρισμός εγγραφών με βάση έναν έλεγχο γνωρίσματος που βελτιστοποιεί ένα συγκεκριμένο κριτήριο
- Είδη διαχωρισμού:
  - 2-αδικός διαχωρισμός - 2-way split
  - Πολλαπλός διαχωρισμός - Multi-way split

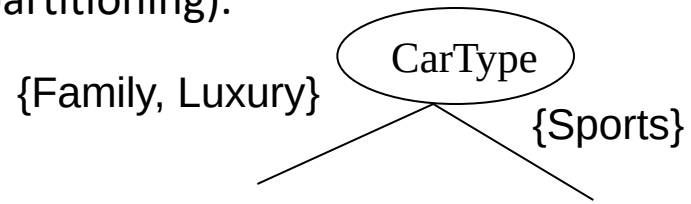
1. Διαχωρισμός σε πεδίο με διακριτές τιμές

- Πολλαπλός διαχωρισμός: Χρησιμοποίησε τόσες διασπάσεις όσες οι διαφορετικές τιμές

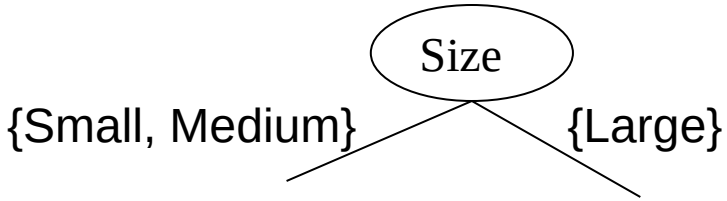


- Δυαδικός Διαχωρισμός: Χωρίζει τις τιμές σε δύο υποσύνολα. Πρέπει να βρει το *βέλτιστο* διαχωρισμό (partitioning).



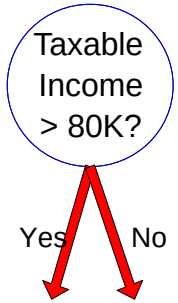


- Αν έχω κ τιμές, υπάρχουν $2^{κ-1}$-1 τρόποι διαχωρισμού
- Όταν έχω <u>διάταξη</u>, πρέπει ο διαχωρισμός να τη διατηρεί

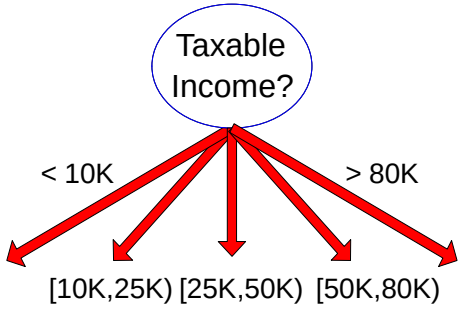


1. Διαχωρισμός σε πεδίο με συνεχείς τιμές

- Διακριτοποίηση ώστε να προκύψει ένα διατεταγμένο κατηγορικό γνώρισμα
  - Ταξινόμηση των τιμών και χωρισμός τους σε περιοχές καθορίζοντας n – 1 σημεία διαχωρισμού, απεικόνιση όλων των τιμών μιας περιοχής στην ίδια κατηγορική τιμή
    - Στατικό – μια φορά στην αρχή
    - Δυναμικό – εύρεση των περιοχών πχ έτσι ώστε οι περιοχές να έχουν το ίδιο διάστημα ή τις ίδιες συχνότητες εμφάνισης ή με χρήση συσταδοποίησης
- Δυαδική Απόφαση: (A < v) or (A >= v)
  - εξετάζει όλους τους δυνατούς διαχωρισμούς (τιμές του v) και επιλέγει τον καλύτερο – υπολογιστικά βαρύ



Δυαδικός διαχωρισμός

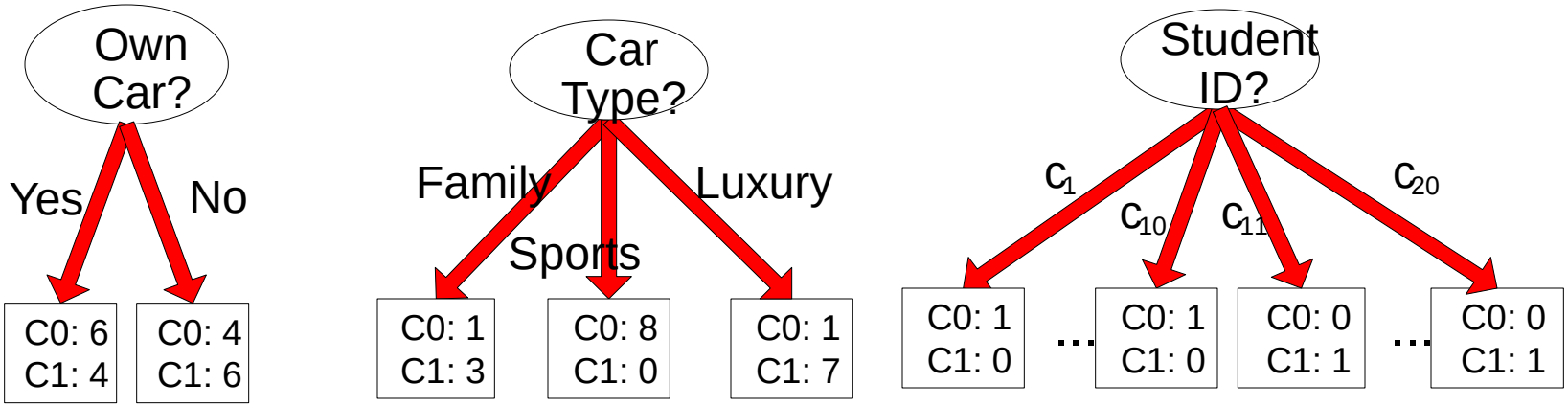


Πολλαπλός διαχωρισμός

- Καλύτερο=Ισοπληθείς ομάδες
  - Διάταξε τις τιμές σε αύξουσα διάταξη
  - Βρες τις ενδιάμεσες γειτονικές τιμές a_i και a_{i+1} (median όχι average)
  - Υπολόγισε το σημείο διαχωρισμού στη μέση των διαχωριστικών τιμών $(a_i+a_{i+1})/2$

2. Βέλτιστος Διαχωρισμός

- Πριν το διαχωρισμό: 10 εγγραφές από κάθε κλάση (0,1)



- Ποια από τις 3 διασπάσεις να προτιμήσουμε; (Δηλαδή, ποια συνθήκη ελέγχου είναι καλύτερη;)
- => ορισμός κριτηρίου βέλτιστου διαχωρισμού

2. Greedy προσέγγιση

- Επιλέγω τη διάσπαση που δίνει κόμβους με ομοιογενείς κατανομές κλάσεων (homogeneous class distribution)
- Χρειάζομαι ένα μέτρο της μη-καθαρότητας ενός κόμβου (node impurity). Δλδ πόσες διαφορετικές κλάσεις περιέχει.

C0: 5 C1: 5

Μη-ομοιογενής,

Μεγάλος βαθμός μη καθαρότητας

«Καλός» κόμβος!!

C0: 9 C1: 1

Ομοιογενής,

Μικρός βαθμός μη καθαρότητας

N1

C1	**0**
C2	**6**
Μη καθαρότητα ~ 0	

N2

C1	**1**
C2	**5**
ενδιάμεση	

N3

C1	**2**
C2	**4**
ενδιάμεση αλλά μεγαλύτερη	

N4

C1	**3**
C2	**3**
Μεγάλη μη καθαρότητα	

I(N1) < I(N2) < I(N3) < I(N4)

2. Καθαρότητα κόμβου

- Για κάθε κόμβο n, μετράμε την καθαρότητα του, I(n)
  - Έστω μια διάσπαση ενός κόμβου (parent) με N εγγραφές σε k παιδιά u_i
  - Έστω N(ui) ο αριθμός εγγραφών κάθε παιδιού ($\Sigma N(u_i) = N$)
- Για να χαρακτηρίσουμε μια διάσπαση, κοιτάμε το κέρδος, δηλαδή τη διαφορά μεταξύ της καθαρότητας του γονέα (πριν τη διάσπαση) και των παιδιών του (μετά τη διάσπαση)

$$\Delta = I(parent) - \sum_{i=1}^{k} \frac{N(u_i)}{N} I(u_i)$$

Βάρος (εξαρτάται από τον αριθμό εγγραφών $N(u_i)$ του κάθε παιδιού)

- "Καλύτερη" διάσπαση = μεγαλύτερο Δ

Μέτρα μη Καθαρότητας

- Πληροφορία κατηγοριοποίησης
- Εντροπία – Entropy
- Κέρδος πληροφορίας
- Λάθος ταξινομήσεις - Misclassification error

Πληροφορία κατηγοριοποίησης

- Είναι η πληροφορία που απαιτείται για την κατηγοριοποίηση ενός δείγματος
- Αν s τα δείγματα και m οι κατηγορίες

$$I(s_1, \dots, s_m) = -\sum_{i=1}^{m} p_i log_2(p_i)$$

- s_i είναι ο αριθμός των δειγμάτων στην κατηγορία C_i και p_i η πιθανότητα ένα δείγμα να ανήκει στην κατηγορία C_i ($p_i=s_i/s$)

Εντροπία κόμβου

- Εντροπία για τον κόμβο t :

$$Entropy(t) = -\sum_{j=1}^{c} p(j \mid t) \log_2 p(j \mid t)$$

p(j | t): σχετική συχνότητα της κλάσης j στον κόμβο t
c: αριθμός κλάσεων

- **Μέγιστη τιμή** log(c) όταν όλες οι εγγραφές είναι ομοιόμορφα κατανεμημένες στις κλάσεις (που σημαίνει τη λιγότερο ενδιαφέρουσα πληροφορία)

- **Ελάχιστη τιμή** (0.0) όταν όλες οι εγγραφές ανήκουν σε μία κλάση (που σημαίνει την πιο ενδιαφέρουσα πληροφορία)

Παραδείγματα

$$Entropy(t) = -\sum_{j=1}^{c} p(j \mid t) \log_2 p(j \mid t)$$

C1	**0**
C2	**6**

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Entropy = – 0 log 0 – 1 log 1 = – 0 – 0 = 0

C1	**1**
C2	**5**

P(C1) = 1/6 P(C2) = 5/6

Entropy = – (1/6) $\log_2$ (1/6) – (5/6) $\log_2$ (5/6) = 0.65

C1	**2**
C2	**4**

P(C1) = 2/6 P(C2) = 4/6

Entropy = – (2/6) $\log_2$ (2/6) – (4/6) $\log_2$ (4/6) = 0.92

Εντροπία γνωρίσματος διάσπασης

- Έστω ότι από τα s δείγματα που έχω συνολικά θεωρώ ένα υποσύνολο S_j με τα δείγματα που έχουν τιμή a_j για το γνώρισμα A
- Η εντροπία του A ορίζεται ως εξής:

$$E(A) = \sum_{j=1}^{v} \frac{\sum_{i=1}^{m} s_{ij}}{s} I(s_{1j}, \dots, s_{mj})$$

- όπου $$I(s_{1j}, \dots, s_{mj}) = -\sum_{i=1}^{m} \frac{s_{ij}}{|S_j|} log_2 \left(\frac{s_{ij}}{|S_j|}\right)$$

- Το κέρδος πληροφορίας (information gain) από την επιλογή του A είναι:

$$Gain(A) = E(S) - \sum_{k \text{ οι διαχωρισμοί του } S} E(S_k)$$

Παράδειγμα

- Έχω συνολικά 100 δείγματα πιστωτών (καλοί και κακοί)
- 40 από αυτά είναι άντρες και 60 από αυτά είναι γυναίκες
- Θέλω να υπολογίσω την εντροπία του γνωρίσματος ‘φύλο’ ως προς την τελική απόφαση που διαχωρίζει τα δείγματα σε καλούς και κακούς πιστωτές

	i=1:Άντρες	i=2:Γυναίκες
j=1:Καλοί	$s_{1,1}=5$	$s_{2,1}=50$
j=2:Κακοί	$s_{1,2}=35$	$S_{2,2}=10$

$$I(s_{11},s_{21}) = -\left(\frac{5}{40}log_2\left(\frac{5}{40}\right)+\frac{50}{60}log_2\left(\frac{50}{60}\right)\right) = 0.594$$

$$I(s_{12},s_{22}) = -\left(\frac{35}{40}log_2\left(\frac{35}{40}\right)+\frac{10}{60}log_2\left(\frac{10}{60}\right)\right) = 0.599$$

$$E(A) = \frac{50+5}{100}0.594+\frac{35+10}{100}0.599 = 0.597$$

Κέρδος πληροφορίας

Και σε αυτήν την περίπτωση, όταν ένας κόμβος p διασπάται σε k σύνολα (παιδιά), η ποιότητα του διαχωρισμού υπολογίζεται ως:

$$GAIN_{split} = Entropy(p) - \left(\sum_{i=1}^{k} \frac{n_i}{n} Entropy(i) \right)$$

όπου, n_i = αριθμός εγγραφών του παιδιού i,

n = αριθμός εγγραφών του κόμβου p.

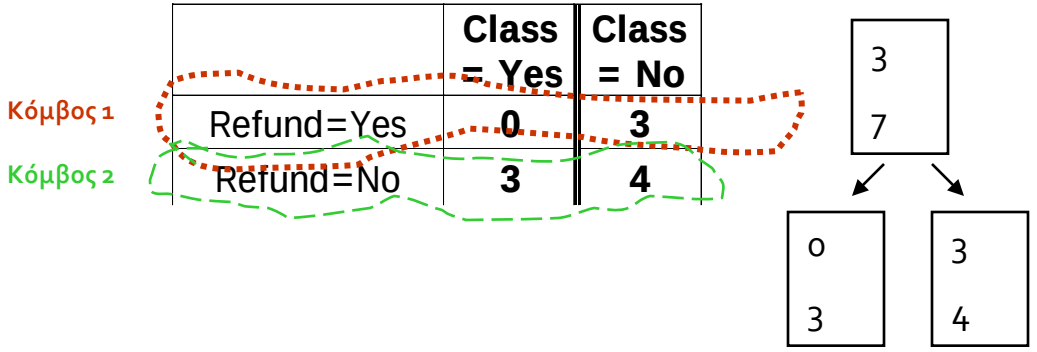
- Χρησιμοποιείται στα ID3 and C4.5
- Όταν χρησιμοποιούμε την εντροπία για τη μέτρηση της μη καθαρότητας τότε η διαφορά καλείται **κέρδος πληροφορίας (information gain)**

Δέντρο Απόφασης: Κέρδος Πληροφορίας

Tid	Refund	Marital Status	Taxable Income	Class
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Πριν τη διάσπαση:

Entropy(Parent) = -0.3 log(0.3)-(0.7)log(0.7) = 0.8813



Διάσπαση στο Refund:

Entropy(Refund=Yes) = 0

Entropy(Refund=No)
= -(3/7)log(3/7) – (4/7)log(4/7) = 0.9852

Entropy(Children)
= **0.3** (0) + **0.7** (0.9852) = 0.6897

Gain = **1** × (0.8813 – 0.6897) = 0.1916

Δέντρο Απόφασης: Κέρδος Πληροφορίας

Κλάση

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

$$Info(D) = I(9,5) = -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14}) = 0.940$$

$$Info_{age}(D) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0) + \frac{5}{14} I(3,2) = 0.694$$

$$Gain(age) = Info(D) - Info_{age}(D) = 0.246$$

age	p_i	n_i	$I(p_i, n_i)$
<=30	2	3	0.971
31…40	4	0	0
>40	3	2	0.971

$$Gain(income) = 0.029$$

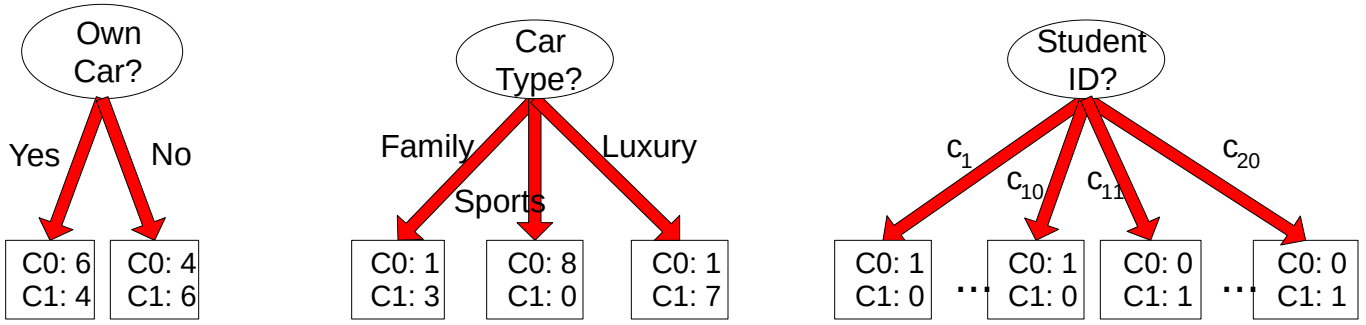
$$Gain(student) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

Δέντρο Απόφασης

$$\Delta = I(parent) - \sum_{i=1}^{k} \frac{N(u_i)}{N} I(u_i)$$

Τείνει να ευνοεί διαχωρισμούς που καταλήγουν σε μεγάλο αριθμό από διασπάσεις που η κάθε μία είναι μικρή αλλά καθαρή



Μπορεί να καταλήξουμε σε πολύ μικρούς κόμβους (με πολύ λίγες εγγραφές) για αξιόπιστες προβλέψεις

Στο παράδειγμα, το **student-id** είναι κλειδί, όχι χρήσιμο για προβλέψεις

Λάθος ταξινόμησης

$$Error(t) = 1 - \max_{class\ i} P(i \mid t)$$

Μετράει το λάθος ενός κόμβου

- **Μέγιστη τιμή** 1-1/c όταν όλες οι εγγραφές είναι ομοιόμορφα κατανεμημένες στις κλάσεις (που σημαίνει τη λιγότερο ενδιαφέρουσα πληροφορία)

- **Ελάχιστη τιμή** (0.0) όταν όλες οι εγγραφές ανήκουν σε μία κλάση (που σημαίνει την πιο ενδιαφέρουσα πληροφορία)

Λάθος ταξινόμησης

$$Error(t) = 1 - \max_{class\ i} P(i \mid t)$$

Μετράει το λάθος ενός κόμβου

Όσες ταξινομούνται σωστά

Παράδειγμα

C1	**0**
C2	**6**

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1

Error = 1 – max (0, 1) = 1 – 1 = 0

C1	**1**
C2	**5**

P(C1) = 1/6 P(C2) = 5/6

Error = 1 – max (1/6, 5/6) = 1 – 5/6 = 1/6

C1	**2**
C2	**4**

P(C1) = 2/6 P(C2) = 4/6

Error = 1 – max (2/6, 4/6) = 1 – 4/6 = 1/3

Κριτήρια Τερματισμού

- Σταματάμε την επέκταση ενός κόμβου όταν όλες οι εγγραφές του ανήκουν στην ίδια κλάση
- Σταματάμε την επέκταση ενός κόμβου όταν όλα τα γνωρίσματα έχουν τις ίδιες τιμές
- Γρήγορος τερματισμός

Διάσπαση Δεδομένων

- Ο αριθμός των εγγραφών μειώνεται όσο κατεβαίνουμε στο δέντρο
- Ο αριθμός των εγγραφών στα φύλλα μπορεί να είναι *πολύ μικρός* για να πάρουμε οποιαδήποτε στατιστικά σημαντική απόφαση
- Μπορούμε να αποτρέψουμε την περαιτέρω διάσπαση όταν ο αριθμός των εγγραφών πέσει κάτω από ένα όριο

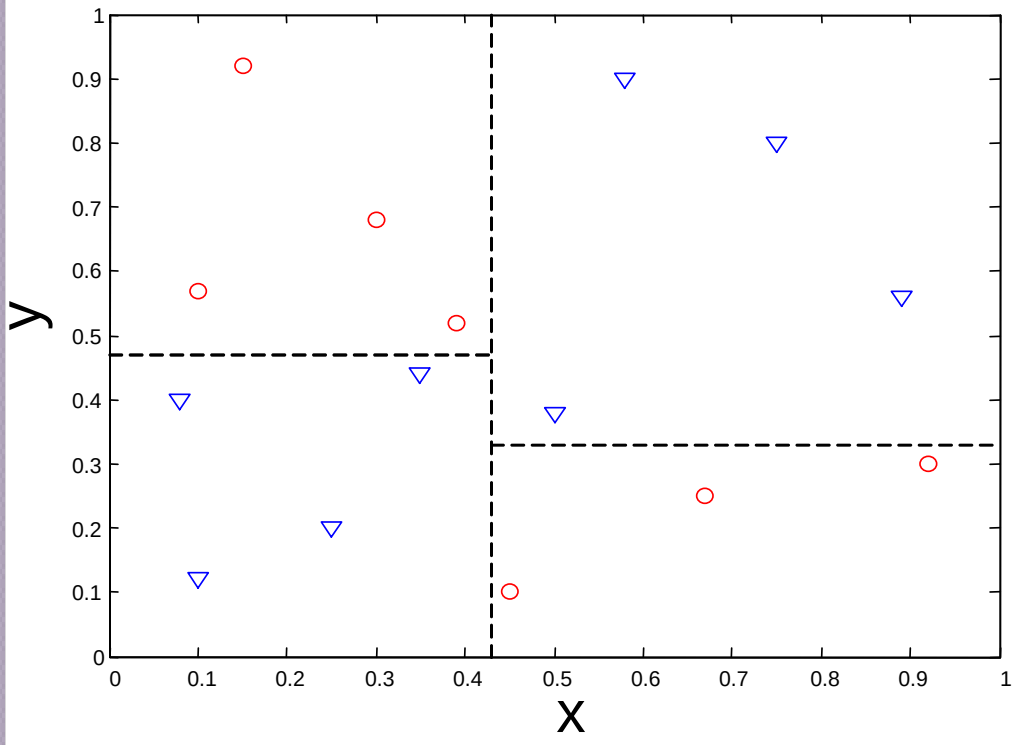
Πλεονεκτήματα Δέντρων Απόφασης

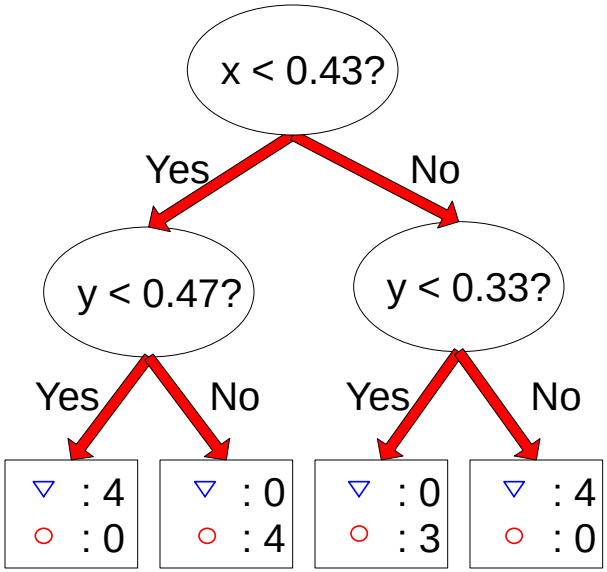
- Μη παραμετρική προσέγγιση: Δε στηρίζεται σε υπόθεση εκ των προτέρων γνώσης σχετικά με τον τύπο της κατανομής πιθανότητας που ικανοποιεί η κλάση ή τα άλλα γνωρίσματα

- Η κατασκευή του βέλτιστου δέντρου απόφασης είναι ένα NP-complete πρόβλημα. Ευριστικοί: Αποδοτική κατασκευή ακόμα και στην περίπτωση πολύ μεγάλου συνόλου δεδομένων
- Αφού το δέντρο κατασκευαστεί, η ταξινόμηση νέων εγγραφών πολύ γρήγορη O(h) όπου h το μέγιστο ύψος του δέντρου
- Εύκολα στην κατανόηση (ιδιαίτερα τα μικρά δέντρα)
- Η ακρίβεια τους συγκρίσιμη με άλλες τεχνικές για μικρά σύνολα δεδομένων
- Καλή συμπεριφορά στο θόρυβο

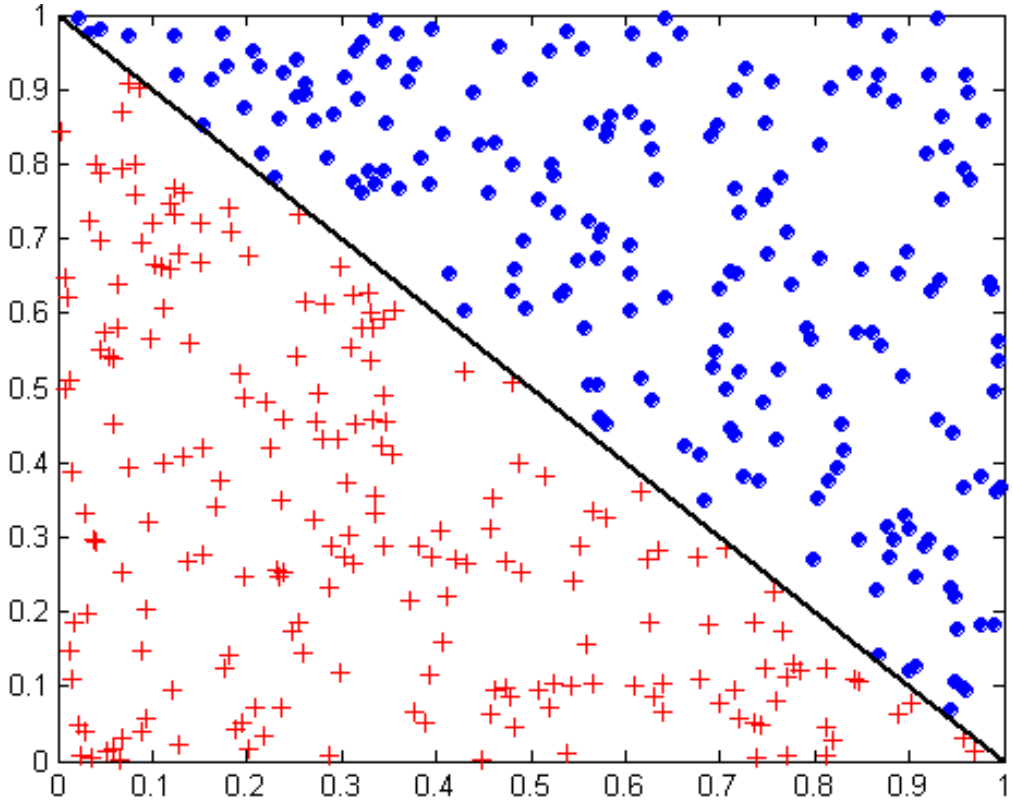
Διαχωρισμός

Όταν η συνθήκη ελέγχου περιλαμβάνει μόνο ένα γνώρισμα τη φορά τότε το **Decision boundary** είναι παράλληλη στους άξονες (τα **decision boundaries** είναι *ορθογώνια παραλληλόγραμμα*)

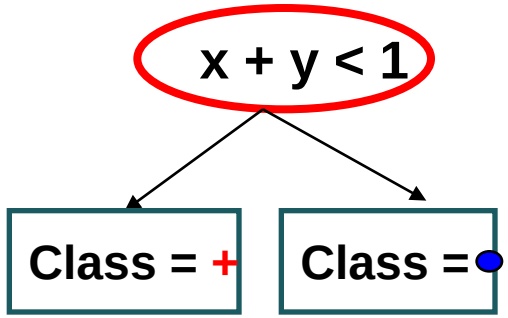




Δέντρο Απόφασης



Oblique (πλάγιο) Δέντρο Απόφασης



- Οι συνθήκες ελέγχου μπορούν να περιλαμβάνουν περισσότερα από ένα γνωρίσματα
- Μεγαλύτερη εκφραστικότητα
- Η εύρεση βέλτιστων συνθηκών ελέγχου είναι υπολογιστικά ακριβή

Αλγόριθμοι

- ID3: διασπά τους κόμβους επιλέγοντας το γνώρισμα ελέγχου με βάση το πληροφοριακό κέρδος
- C4.5: επέκταση του ID3. Λειτουργεί και σε συνεχή γνωρίσματα (κάνοντας αυτόματα διακριτοποίηση). Κάνει κλάδεμα του δέντρου

Συνοψίζοντας

- Προτερήματα - Pros
  - + Λογικός χρόνος εκπαίδευσης
  - + Γρήγορη εφαρμογή
  - + Ευκολία στην κατανόηση
  - + Εύκολη υλοποίηση
  - + Μπορεί να χειριστεί μεγάλο αριθμό γνωρισμάτων
- Μειονεκτήματα - Cons
  - – Δεν μπορεί να χειριστεί περίπλοκες σχέσεις μεταξύ των γνωρισμάτων
  - – Απλά όρια απόφασης (decision boundaries)
  - – Προβλήματα όταν λείπουν πολλά δεδομένα

Παραδείγματα

Πρόγνωση καιρού

Ουρανός	Θερμοκρασία	Υγρασία	Άνεμος	Κατάλληλος για παιχνίδι;
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	mild	normal	false	yes
rainy	mild	normal	true	no
overcast	mild	normal	true	yes
sunny	mild	high	false	no
sunny	mild	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

Με βάση παλιότερα στοιχεία μπορώ να βρω τους κανόνες για το πότε ο καιρός είναι κατάλληλος για παιχνίδι και πότε όχι;

Ανάλυση συνθηκών

Ουρανός	Θερμοκρασία	Υγρασία	Άνεμος	Κατάλληλος για παιχνίδι;
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
…	…	…	…	…

If outlook = sunny and humidity = high then play = no

If outlook = rainy and windy = true then play = no

If outlook = overcast then play = yes

If humidity = normal then play = yes

If none of the above then play = yes

Μικτά δεδομένα

Outlook	Temperature	Humidity	Windy	Play
sunny	85	85	false	no
sunny	80	90	true	no
overcast	83	86	false	yes
rainy	70	96	false	yes
rainy	68	80	false	yes
rainy	65	70	true	no
overcast	64	65	true	yes
sunny	72	95	false	no
sunny	69	70	false	yes
rainy	75	80	false	yes
sunny	75	70	true	yes
overcast	72	90	true	yes
overcast	81	75	false	yes
rainy	71	91	true	no

Τι αλλάζει στους κανόνες μου αν εκτός από κατηγορικά γνωρίσματα έχω και γνωρίσματα διαστήματος;

Κανόνες με μεικτά δεδομένα

Ουρανός	Θερμοκρασία	Υγρασία	Άνεμος	Κατάλληλος για παιχνίδι;
sunny	85	85	false	no
sunny	80	90	true	no
overcast	83	86	false	yes
rainy	75	80	false	yes
…	…	…	…	…

If outlook = sunny and humidity > 83 then play = no

If outlook = rainy and windy = true then play = no

If outlook = overcast then play = yes

If humidity < 85 then play = yes

If none of the above then play = yes

Επιλογή φακών επαφής

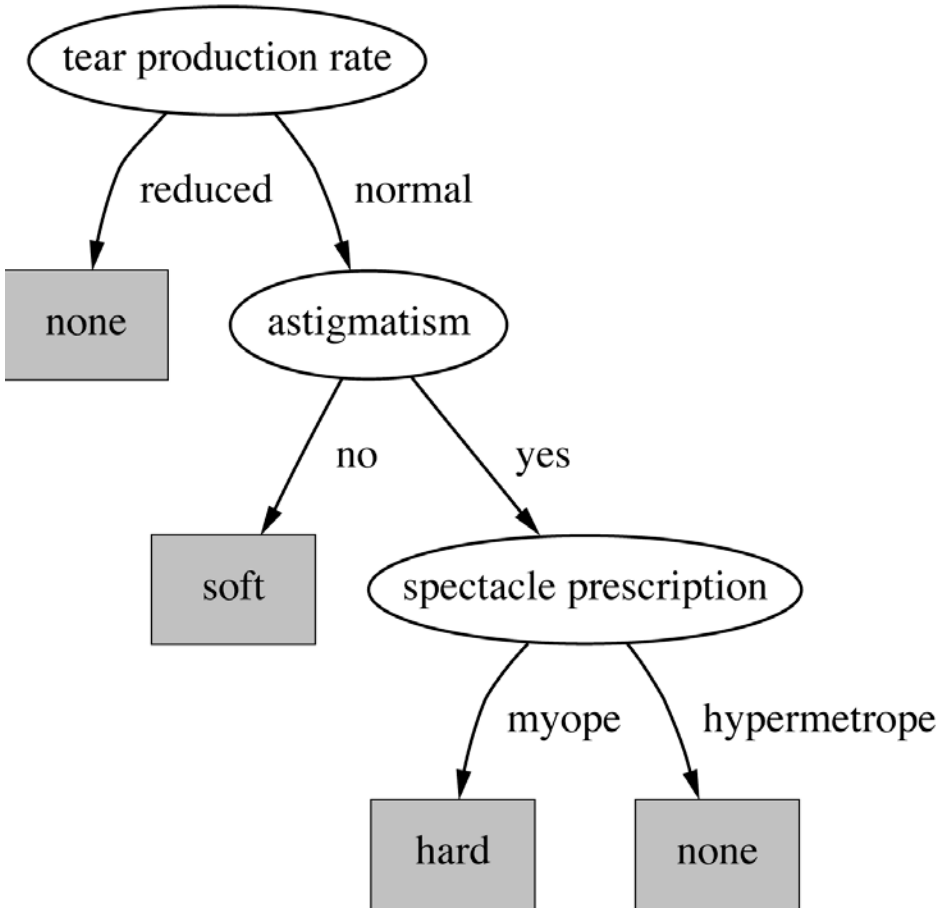
Ηλικία	Πάθηση	Αστιγματισμός	Δάκρυα	Είδος φακών
Young	Myope	No	Reduced	None
Young	Myope	No	Normal	Soft
Young	Myope	Yes	Reduced	None
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	No	Reduced	None
Young	Hypermetrope	No	Normal	Soft
Young	Hypermetrope	Yes	Reduced	None
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	No	Reduced	None
Pre-presbyopic	Myope	No	Normal	Soft
Pre-presbyopic	Myope	Yes	Reduced	None
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	No	Reduced	None
Pre-presbyopic	Hypermetrope	No	Normal	Soft
Pre-presbyopic	Hypermetrope	Yes	Reduced	None
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	No	Reduced	None
Presbyopic	Myope	No	Normal	None
Presbyopic	Myope	Yes	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	No	Reduced	None
Presbyopic	Hypermetrope	No	Normal	Soft
Presbyopic	Hypermetrope	Yes	Reduced	None
Presbyopic	Hypermetrope	Yes	Normal	None

Ένα πλήρες και ορθό σύνολο κανόνων

Πλήρες: Κάθε άλλος κανόνας παράγεται από αυτούς
Ορθό: Δεν υπάρχουν συγκρούσεις σε αυτά που παράγουν οι κανόνες

```
If tear production rate = reduced then recommendation = none

If age = young and astigmatic = no
    and tear production rate = normal then recommendation = soft

If age = pre-presbyopic and astigmatic = no
    and tear production rate = normal then recommendation = soft

If age = presbyopic and spectacle prescription = myope
    and astigmatic = no  then recommendation = none

If spectacle prescription = hypermetrope and astigmatic = no
    and tear production rate = normal then recommendation = soft

If spectacle prescription = myope and astigmatic = yes
    and tear production rate = normal then recommendation = hard

If age young and astigmatic = yes
    and tear production rate = normal then recommendation = hard

If age = pre-presbyopic
    and spectacle prescription = hypermetrope
    and astigmatic = yes then recommendation = none

If age = presbyopic and spectacle prescription = hypermetrope
    and astigmatic = yes then recommendation = none
```

Δέντρο απόφασης



Ταξινόμηση λουλουδιών

	Sepal length	Sepal width	Petal length	Petal width	Type
1	5.1	3.5	1.4	0.2	Iris setosa
2	4.9	3.0	1.4	0.2	Iris setosa
…					
51	7.0	3.2	4.7	1.4	Iris versicolor
52	6.4	3.2	4.5	1.5	Iris versicolor
…					
101	6.3	3.3	6.0	2.5	Iris virginica
102	5.8	2.7	5.1	1.9	Iris virginica
…					

```
If petal length < 2.45 then Iris setosa
If sepal width < 2.10 then Iris versicolor
...
```

Πρόβλεψη επιδόσεων CPU

- Παράδειγμα: 209 διαφορετικές συνθέσεις υπολογιστών

	Cycle time (ns)	Main memory (Kb)		Cache (Kb)	Channels		Performance
	MYCT	MMIN	MMAX	CACH	CHMIN	CHMAX	PRP
1	125	256	6000	256	16	128	198
2	29	8000	32000	32	8	32	269
…							
208	480	512	8000	32	0	0	67
209	480	1000	4000	0	0	0	45

- Συνάρτηση γραμμικής παλινδρόμησης

```
PRP =  -55.9 + 0.0489 MYCT + 0.0153 MMIN + 0.0056 MMAX
       + 0.6410 CACH - 0.2700 CHMIN + 1.480 CHMAX
```

Ταξινόμηση καρπών σόγιας

Ένα παράδειγμα με πολλά γνωρίσματα

	Attribute	Number of values	Sample value
Environment	Time of occurrence	7	July
	Precipitation	3	Above normal
…			
Seed	Condition	2	Normal
	Mold growth	2	Absent
…			
Fruit	Condition of fruit pods	4	Normal
	Fruit spots	5	?
Leaves	Condition	2	Abnormal
	Leaf spot size	3	?
…			
Stem	Condition	2	Abnormal
	Stem lodging	2	Yes
…			
Roots	Condition	3	Normal
Diagnosis		19	Diaporthe stem canker

Ο ρόλος της γνώσης πεδίου

```
If leaf condition is normal
   and stem condition is abnormal
   and stem cankers is below soil line
   and canker lesion color is brown

then
   diagnosis is rhizoctonia root rot
```

```
If leaf malformation is absent
   and stem condition is abnormal
   and stem cankers is below soil line
   and canker lesion color is brown

then
   diagnosis is rhizoctonia root rot
```

Γνώση πεδίου: “leaf condition is normal” ➔ “leaf malformation is absent”!