Ταξινόμηση ΙΙ

Οι διαφάνειες στηρίζονται στο **P.-N. Tan, M.Steinbach, V. Kumar, «Introduction to Data Mining»**, Addison Wesley, 2006

Σύντομη Ανακεφαλαίωση

Ορισμός

Ταξινόμηση (classification)

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

Είσοδος: συλλογή από εγγραφές (αντικείμενα)
Κάθε εγγραφή περιέχει ένα σύνολο από γνωρίσματα (attributes)
Ένα από τα γνωρίσματα είναι η κλάση (class)

Έξοδος: ένα μοντέλο (model) για το γνώρισμα κλάση ως μια συνάρτηση των τιμών των άλλων γνωρισμάτων

Στόχος: νέες εγγραφές θα πρέπει να ανατίθενται σε μία από τις κλάσεις με τη μεγαλύτερη δυνατή ακρίβεια.

γνώρισμα κλάση

Tid	Επιστροφή	Οικογενειακή κατάσταση	Φορολογητέο εισόδημα	Απάτη
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

Συνήθως το σύνολο δεδομένων εισόδου χωρίζεται σε:
- ένα σύνολο εκπαίδευσης (training set) και
- ένα σύνολο ελέγχου (test test)

Το *σύνολο εκπαίδευσης* χρησιμοποιείται για να κατασκευαστεί το μοντέλο και το *σύνολο ελέγχου* για να το επικυρώσει.

Εισαγωγή

❑ Χρησιμοποιείται ως:
- Περιγραφικό μοντέλο (descriptive modeling): ως επεξηγηματικό εργαλείο – πχ ποια χαρακτηριστικά κάνουν ένα ζώο να χαρακτηριστεί ως θηλαστικό
- Μοντέλο πρόβλεψης (predictive modeling): για τη πρόβλεψη της κλάσης άγνωστων εγγραφών – πχ δοσμένων των χαρακτηριστικών κάποιου ζώου να προβλέψουμε αν είναι θηλαστικό, πτηνό, ερπετό ή αμφίβιο

❑ Κατάλληλη κυρίως για:
- δυαδικές κατηγορίες ή κατηγορίες για τις οποίες δεν υπάρχει διάταξη διακριτές (nominal) vs διατεταγμένες (ordinal)
- για μη ιεραρχικές κατηγορίες

❑ Θεωρούμε ότι τιμή (ετικέτα) της κλάσης (γνώρισμα y) είναι διακριτή τιμή

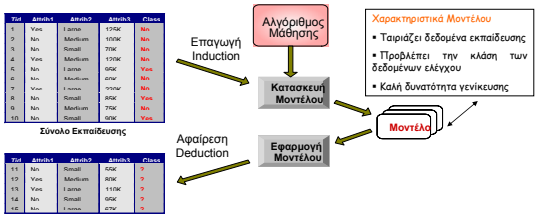
Αν όχι, regression (οπισθοδρόμηση) όπου το γνώρισμα y παίρνει *συνεχείς τιμές*

Εισαγωγή

Βήματα Ταξινόμησης

1. Κατασκευή Μοντέλου
Χρησιμοποιώντας το σύνολο εκπαίδευσης (στις εγγραφές του το γνώρισμα της κλάσης είναι προκαθορισμένο)
Το μοντέλο μπορεί να είναι ένα δέντρο ταξινόμησης, κανόνες, μαθηματικοί τύποι κλπ)

2. Εφαρμογή Μοντέλου για την ταξινόμηση μελλοντικών ή άγνωστων αντικειμένων
Εκτίμηση της ακρίβειας του μοντέλου με χρήση συνόλου ελέγχου
Accuracy rate: το ποσοστό των εγγραφών του συνόλου ελέγχου που ταξινομούνται σωστά από το μοντέλο
Πρέπει να είναι ανεξάρτητα από τα δεδομένα εκπαίδευσης (αλλιώς over-fitting)



Προεπεξεργασία

1. Καθαρισμός Δεδομένων (data cleaning)
   Προεπεξεργασία δεδομένων και χειρισμός τιμών που λείπουν (πχ τις αγνοούμε ή τις αντικαθιστούμε με ειδικές τιμές)

2. Ανάλυση Σχετικότητα (Relevance analysis) (επιλογή χαρακτηριστικών (γνωρισμάτων) -- feature selection)
   Απομάκρυνση των μη σχετικών ή περιττών γνωρισμάτων

3. Μετασχηματισμοί Δεδομένων (Data transformation)
   Κανονικοποίηση ή/και Γενίκευση
   Πιθανών αριθμητικά γνωρίσματα ⇒ κατηγορικά {low,medium,high}
   Κανονικοποίηση αριθμητικών δεδομένων στο [0,1]

Εκτίμηση Μεθόδων Ταξινόμηση

- Προβλεπόμενη πιστότητα - Predictive accuracy
- Ταχύτητα (speed)
  - Χρόνος κατασκευής του μοντέλου
  - Χρόνος χρήσης/εφαρμογής του μοντέλου
- Robustness
  - Χειρισμός θορύβου και τιμών που λείπουν
- Scalability
  - Αποδοτικότητα σε βάσεις δεδομένων αποθηκευμένες στο δίσκο
- Interpretability:
  - Πόσο κατανοητό είναι το μοντέλο και τι νέα πληροφορία προσφέρει
- Ποιότητα - Goodness of rules (quality)
  - Πχ μέγεθος του δέντρου

Ορισμός

Τεχνικές ταξινόμησης

- Δέντρα Απόφασης (decision trees)
- Κανόνες (Rule-based Methods)
- Κοντινότερος Γείτονας
- Memory based reasoning
- Νευρωνικά Δίκτυα
- Naïve Bayes and Bayesian Belief Networks
- Support Vector Machines

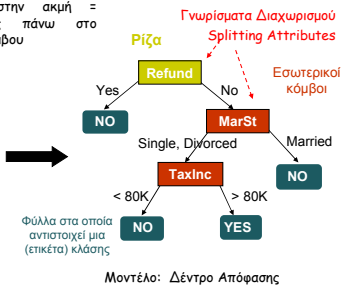
Δέντρα Απόφασης

Δέντρο Απόφασης: Παράδειγμα

- Εσωτερικοί κόμβοι αντιστοιχούν σε κάποιο γνώρισμα
- Φύλλα αντιστοιχούν σε κλάσεις
- Διαχωρισμός (split) ενός κόμβου σε παιδιά
- Η ετικέτα στην ακμή = συνθήκη/έλεγχος πάνω στο γνώρισμα του κόμβου

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

Δεδομένα Εκπαίδευσης



Μοντέλο: Δέντρο Απόφασης

Δέντρο Απόφασης

Αφού κατασκευαστεί το δέντρο, η χρήση του στην ταξινόμηση είναι απλή

- Διαπέραση του δέντρου από πάνω-προς-τα-κάτω

Θα δούμε στη συνέχεια αλγορίθμους για την κατασκευή του (βήμα επαγωγής)

Κατασκευή του δέντρου (με λίγα λόγια):

1. ξεκίνα με ένα 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)

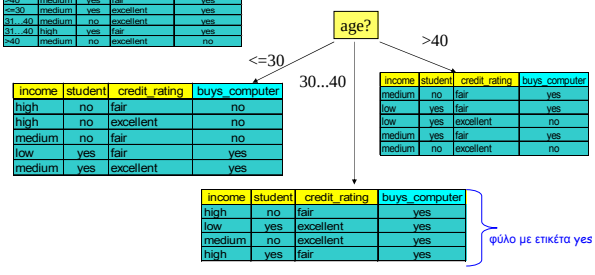
Το βασικό θέμα είναι

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

Δέντρο Απόφασης

Παράδειγμα

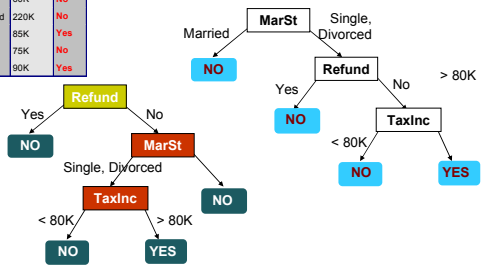
age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes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



Δέντρο Απόφασης: Παράδειγμα

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

Για το ίδιο σύνολο εκπαίδευσης υπάρχουν διαφορετικά δέντρα



Δέντρο Απόφασης: Κατασκευή

Γενικά, ο αριθμός των πιθανών Δέντρων Απόφασης είναι εκθετικός.

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

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

Δέντρο Απόφασης: Αλγόριθμος του Hunt

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

D_t

D_t: το σύνολο των εγγραφών εκπαίδευσης που έχουν φτάσει στον κόμβο t

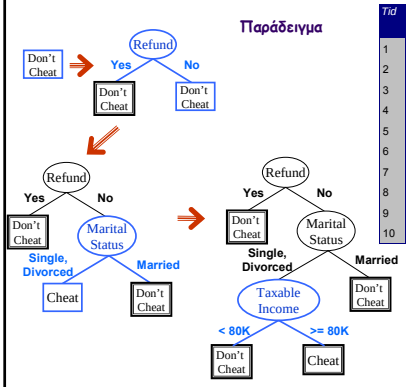
Γενική Διαδικασία (αναδρομικά σε κάθε κόμβο)

- 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 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

Δέντρο Απόφασης: Αλγόριθμος του Hunt

Παράδειγμα



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

Δέντρο Απόφασης: Κατασκευή Δέντρου

Πως θα γίνει η διάσπαση του κόμβου;

Greedy στρατηγική

Διάσπαση εγγραφών με βάση έναν έλεγχο γνωρίσματος που βελτιστοποιεί ένα συγκεκριμένο κριτήριο

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

Δέντρο Απόφασης: Κατασκευή Δέντρου

Καθορισμός των συνθηκών του ελέγχου για τα γνωρίσματα

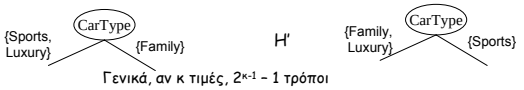
- Εξαρτάται από τον τύπο των γνωρισμάτων
  - Διακριτές - Nominal
  - Διατεταγμένες - Ordinal
  - Συνεχείς - Continuous
- Είδη διασπάσεων:
  - 2-αδική διάσπαση - 2-way split
  - Πολλαπλή διάσπαση - Multi-way split

Δέντρο Απόφασης: Κατασκευή Δέντρου

Διαχωρισμός βασισμένος σε διακριτές τιμές

- Πολλαπλός διαχωρισμός: Χρησιμοποίησε τόσες διασπάσεις όσες οι διαφορετικές τιμές

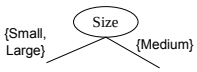
- Δυαδικός Διαχωρισμός: Χωρίζει τις τιμές σε δύο υποσύνολα. Πρέπει να βρει το *βέλτιστο* διαχωρισμό (partitioning)



Γενικά, αν κ τιμές, $2^{κ-1} - 1$ τρόποι

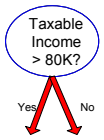
Όταν υπάρχει διάταξη, πρέπει οι διασπάσεις να μη την παραβιάζουν

Αυτός ο διαχωρισμός;

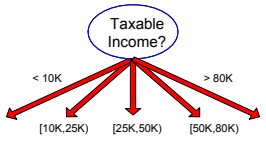


Δέντρο Απόφασης: Κατασκευή Δέντρου

Διαχωρισμός βασισμένος σε συνεχείς τιμές



Δυαδικός διαχωρισμός



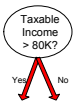
Πολλαπλός διαχωρισμός

Δέντρο Απόφασης: GINI

Συνεχή Γνωρίσματα

- Πχ, χρήση **δυαδικών αποφάσεων** πάνω σε μία τιμή
- Πολλές επιλογές για την τιμή διαχωρισμού
  - Αριθμός πιθανών διαχωρισμών = Αριθμός διαφορετικών τιμών – έστω N
- Κάθε τιμή διαχωρισμού v συσχετίζεται με έναν πίνακα μετρητών
  - Μετρητές των κλάσεων για κάθε μια από τις δύο διασπάσεις, $A < v$ and $A \geq v$
- Απλή μέθοδος για την επιλογή της καλύτερης τιμής v (βέλτιστη τιμή διαχωρισμού – best split point)
  - Διάταξε τις τιμές του A σε αύξουσα διάταξη
  - Συνήθως επιλέγεται το μεσαίο σημείο ανάμεσα σε γειτονικές τιμές ας υποψήφιο
  - $(a_i+a_{i+1})/2$ μέσο των τιμών a_i και a_{i+1}
  - Επέλεξε το «βέλτιστο» ανάμεσα στα υποψήφια

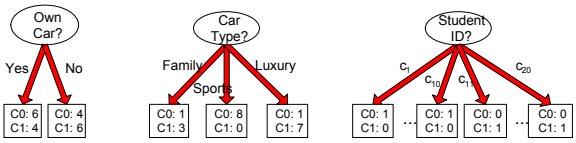
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Δέντρο Απόφασης: Κατασκευή Δέντρου

Ορισμός «Βέλτιστου» Διαχωρισμού

Πριν το διαχωρισμό: **10 εγγραφές της κλάσης 0, 10 εγγραφές της κλάσης 1**



Ποια από τις 3 διασπάσεις να προτιμήσουμε;

Ποια συνθήκη ελέγχου είναι καλύτερη -> ορισμός «Βέλτιστου» διαχωρισμού;

Δέντρο Απόφασης: Κατασκευή Δέντρου

Ορισμός «Βέλτιστου» Διαχωρισμού (συνέχεια)

Greedy προσέγγιση:
Σε κάθε βήμα, προτιμούνται οι κόμβοι με ομοιογενείς κατανομές κλάσεων (homogeneous class distribution)

Χρειαζόμαστε μία μέτρηση της μη καθαρότητας ενός κόμβου (node impurity)

«Καλός» κόμβος!!

C0: 5 C1: 5	Μη-ομοιογενής, Μεγάλος βαθμός μη καθαρότητας

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)

Δέντρο Απόφασης: Κατασκευή Δέντρου

Πως θα χρησιμοποιήσουμε τη μέτρηση καθαρότητας;

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 $N(u_i)$ ο αριθμός εγγραφών κάθε παιδιού ($\sum N(u_i) = N$)

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)

Διαλέγουμε την «καλύτερη» διάσπαση (μεγαλύτερο Δ)

Δέντρο Απόφασης: Αλγόριθμος του Hunt

Ψευδό-κώδικας

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 ∈ 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)

Δέντρο Απόφασης: Κατασκευή Δέντρου

Μέτρα μη Καθαρότητας

1. Ευρετήριο Gini - Gini Index
2. Εντροπία - Entropy
3. Λάθος ταξινομήσεις - Misclassification error

Δέντρο Απόφασης: GINI

Ευρετήριο Gini για τον κόμβο t :

$$GINI(t) = 1 - \sum_{j=1}^{c} [p(j \mid t)]^2$$

p(j | t) σχετική συχνότητα της κλάσης j στον κόμβο t

c αριθμός κλάσεων

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

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

Παραδείγματα:

N1	
C1	0
C2	6
Gini=0.000	

N2	
C1	1
C2	5
Gini=0.278	

N3	
C1	2
C2	4
Gini=0.444	

N4	
C1	3
C2	3
Gini=0.500	

Δέντρο Απόφασης: GINI

Χρήση του στην κατασκευή του δέντρου απόφασης

- Χρησιμοποιείται στα CART, SLIQ, SPRINT, IBM Intellignet Miner

Όταν ένας κόμβος p διασπάται σε k κόμβους (παιδιά), (που σημαίνει ότι το σύνολο των εγγραφών του κόμβου χωρίζεται σε k υποσύνολα), η ποιότητα του διαχωρισμού υπολογίζεται ως:

$$GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$$

όπου, n_i = αριθμός εγγραφών του παιδιού i,
n = αριθμός εγγραφών του κόμβου p.

Ψάχνουμε για:

- Πιο καθαρές
- Πιο μεγάλες (σε αριθμό) μικρές διασπάσεις

Δέντρο Απόφασης: GINI

Παράδειγμα

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

$$gini(D) = 1 - \left(\frac{9}{14}\right)^2 - \left(\frac{5}{14}\right)^2 = 0.459$$

Έστω ότι το διασπάμε με βάση το income
Πρέπει να θεωρήσουμε όλες τις δυνατές διασπάσεις
Έστω μόνο δυαδικές
D1: {low, medium} και D2 {high}
D3: {low} και D4 {medium, high} …

Αν πολλαπλές διασπάσεις, πρέπει να θεωρήσουμε και άλλες διασπάσεις

Με τον ίδιο τρόπο εξετάζουμε και πιθανές διασπάσεις με βάση τα άλλα τρία γνωρίσματα (δηλαδή, age, student, credit_rating)

Δέντρο Απόφασης: Εντροπία

Εντροπία για τον κόμβο t :

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

p(j |t) σχετική συχνότητα της κλάσης j στον κόμβο t
c αριθμός κλάσεων
log είναι λογάριθμος με βάση το 2

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:

N1	
C1	0
C2	6
Entropy=0.000	
Gini = 0.000	

N2	
C1	1
C2	5
Entropy=0.650	
Gini = 0.278	

N3	
C1	2
C2	4
Entropy = 0.92	
Gini = 0.444	

N4	
C1	3
C2	3
Entropy = 1.000	
Gini = 0.500	

Δέντρο Απόφασης: Εντροπία

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

Παράδειγμα

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

Κλάση

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

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

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

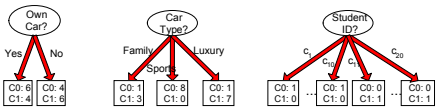
age	p_i	n_i	$I(p_i, n_i)$
<=30	2	3	0.971
31…40	4	0	0
>40	3	2	0.971

$$Gain(income) = 0.029$$
$$Gain(student) = 0.151$$
$$Gain(credit\_rating) = 0.048$$

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 -> αλλά όχι το μέγιστο κέρδος!

Δέντρο Απόφασης: Λόγος Κέρδους

- Μία λύση είναι να έχουμε μόνο δυαδικές διασπάσεις
- Εναλλακτικά, μπορούμε να λάβουμε υπό όψιν μας *τον αριθμό* των κόμβων – ένα είδος κανονικοποίησης

$$GainRATIO_{split} = \frac{GAIN_{Split}}{SplitINFO}$$

Όπου:

$$SplitINFO = -\sum_{i=1}^{k} \frac{n_i}{n} \log \frac{n_i}{n}$$

SplitINFO: εντροπία της διάσπασης
Μεγάλος αριθμός μικρών διασπάσεων (υψηλή εντροπία) τιμωρείται

Χρησιμοποιείται στο C4.5

Παράδειγμα

Έστω N εγγραφές αν τις χωρίσουμε

Σε 3 ίσους κόμβους SplitINFO = - log(1/3) = log3

Σε 2 ίσους κόμβους SplitINFO = - log(1/2) = log2 = 1

Άρα οι 2 ευνοούνται

Δέντρο Απόφασης: Σύγκριση

Και τα τρία μέτρα επιστρέφουν καλά αποτελέσματα

- Κέρδος Πληροφορίας:
  Δουλεύει καλύτερα σε γνωρίσματα με πολλαπλές τιμές
- Λόγος Κέρδους:
  Τείνει να ευνοεί διαχωρισμούς όπου μία διαμέριση είναι πολύ μικρότερη από τις υπόλοιπες
- Ευρετήριο Gini:
  Δουλεύει καλύτερα σε γνωρίσματα με πολλαπλές τιμές
  Δε δουλεύει τόσο καλά όταν ο αριθμός των κλάσεων είναι μεγάλος
  Τείνει να ευνοεί ελέγχους που οδηγούν σε ισομεγέθεις διαμερίσεις που και οι δύο είναι καθαρές

Δέντρο Απόφασης: Λάθος Ταξινόμησης

Λάθος ταξινόμησης (classification error) για τον κόμβο t :

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

Όσες ταξινομούνται σωστά

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)

Παραδείγματα:

N1

C1	0
C2	6
Error=0.000	

Gini = 0.000
Entropy = 0.000

N2

C1	1
C2	5
Error=0.167	

Gini = 0.278
Entropy = 0.650

N3

C1	2
C2	4
Error = 0.333	

Gini = 0.444
Entropy = 0.920

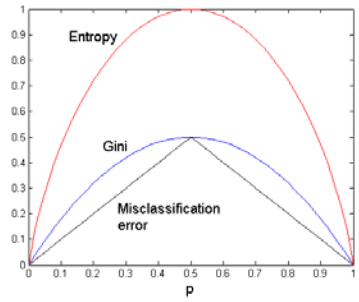
N4

C1	3
C2	3
Error = 0.500	

Gini = 0.500
Entropy = 1.000

Δέντρο Απόφασης: Σύγκριση

Για ένα πρόβλημα δύο κλάσεων



p ποσοστό εγγραφών που ανήκει σε μία από τις δύο κλάσεις

(p κλάση +, 1-p κλάση -)

Όλες την μεγαλύτερη τιμή για 0.5 (ομοιόμορφη κατανομή)

Όλες μικρότερη τιμή όταν όλες οι εγγραφές σε μία μόνο κλάση (0 και στο 1)

Δέντρο Απόφασης: Σύγκριση

- Όπως είδαμε και στα παραδείγματα οι τρεις μετρήσεις είναι συνεπής μεταξύ τους, πχ N1 μικρότερη τιμή από το N2 και με τις τρεις μετρήσεις
- Ωστόσο το γνώρισμα που θα επιλεγεί για τη συνθήκη ελέγχου εξαρτάται από το ποια μέτρηση χρησιμοποιείται

Δέντρο Απόφασης: Αλγόριθμος του Hunt

Ψευδό-κώδικας (πάλι)

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 ∈ A, with the highest information gain (gini, error); 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)

Δέντρο Απόφασης

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

Δέντρο Απόφασης

Πλεονεκτήματα

- Καλή συμπεριφορά στο θόρυβο
- Η ύπαρξη πλεοναζόντων γνωρισμάτων (γνωρίσματα των οποίων η τιμή εξαρτάται από κάποιο άλλο) δεν είναι καταστροφική για την κατασκευή. Χρησιμοποιείται ένα από τα δύο.
  Αν πάρα πολλά, μπορεί να οδηγήσουν σε δέντρα πιο μεγάλα από ότι χρειάζεται

Δέντρο Απόφασης

Εκφραστικότητα

- Δυνατότητα αναπαράστασης για συναρτήσεις διακριτών τιμών, αλλά δε δουλεύουν σε κάποια είδη δυαδικών προβλημάτων –
  πχ, parity 0(1) αν υπάρχει μονός (ζυγός) αριθμός από δυαδικά γνωρίσματα 2^d κόμβοι για d γνωρίσματα
- Όχι καλή συμπεριφορά για συνεχείς μεταβλητές
  Ιδιαίτερα όταν η συνθήκη ελέγχου αφορά ένα γνώρισμα τη φορά

Δέντρο Απόφασης

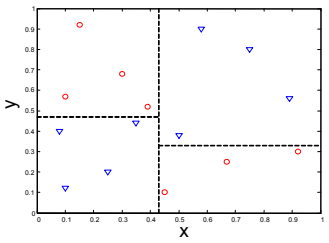
Decision Boundary

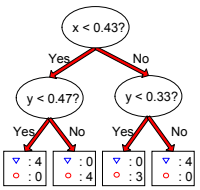
Μέχρι στιγμής είδαμε ελέγχους που αφορούν μόνο ένα γνώρισμα τη φορά, μπορούμε να δούμε τη διαδικασία ως τη διαδικασία *διαμερισμού του χώρου* των γνωρισμάτων σε ξένες περιοχές μέχρι κάθε περιοχή να περιέχει εγγραφές που να ανήκουν στην ίδια κλάση

Η οριακή γραμμή (Border line) μεταξύ δυο γειτονικών περιοχών που ανήκουν σε διαφορετικές κλάσεις ονομάζεται και **decision boundary (όριο απόφασης)**

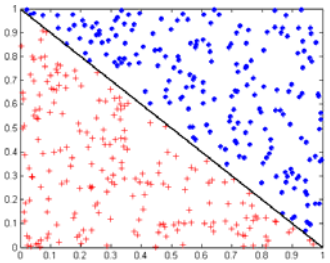
Δέντρο Απόφασης

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

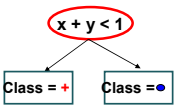




Δέντρο Απόφασης



Oblique (πλάγιο) Δέντρο Απόφασης



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

Δέντρο Απόφασης - Περίληψη

- 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

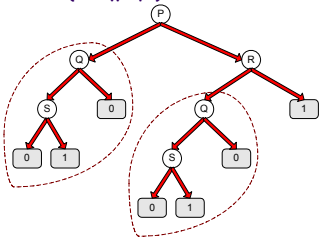
Δέντρο Απόφασης

Στρατηγική αναζήτησης

- Ο αλγόριθμος που είδαμε χρησιμοποιεί μια greedy, top-down, αναδρομική διάσπαση για να φτάσει σε μια αποδεκτή λύση
- Άλλες στρατηγικές?
  - Bottom-up (από τα φύλλα, αρχικά κάθε εγγραφή και φύλλο)
  - Bi-directional

Δέντρο Απόφασης

Tree Replication (Αντίγραφα)



Το ίδιο υπο-δέντρο να εμφανίζεται πολλές φορές σε ένα δέντρο απόφασης

Αυτό κάνει το δέντρο πιο περίπλοκο και πιθανών δυσκολότερο στην κατανόηση

Σε περιπτώσεις διάσπασης ενός γνωρίσματος σε κάθε εσωτερικό κόμβο – ο ίδιος έλεγχος σε διαφορετικά σημεία

Δέντρο Απόφασης: 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 (δεν είναι δυνατός επιπλέον διαχωρισμός)
  - Γρήγορος τερματισμός

Δέντρο Απόφασης

Data Fragmentation – 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

Overfitting

Overfitting

Λάθη

- *Εκπαίδευσης – training error* (training, resubstitution, apparent): Λάθη ταξινόμησης στα δεδομένα του συνόλου εκπαίδευσης (ποσοστό δεδομένων εκπαίδευσης που ταξινομούνται σε λάθος κλάση)
- *Γενίκευσης – generalization error* (generalization): τα αναμενόμενα λάθη ταξινόμησης του μοντέλου σε δεδομένα που δεν έχει δει

Overfitting

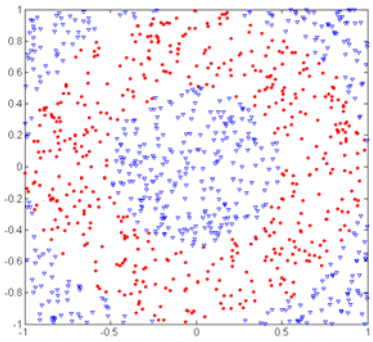
Μπορεί ένα μοντέλο που ταιριάζει πολύ καλά με τα δεδομένα εκπαίδευσης να έχει μεγαλύτερο λάθος γενίκευσης από ένα μοντέλο που ταιριάζει λιγότερο καλά στα δεδομένα εκπαίδευσης

Overfitting

Εκτίμηση Λάθους Γενίκευσης

- Χρήση Δεδομένων Εκπαίδευσης
  - 1. αισιόδοξη εκτίμηση
  - 2. απαισιόδοξη εκτίμηση
- 3. Χρήση Δεδομένων Ελέγχου

Overfitting



Δύο κλάσεις: κλάση 1 (500 κυκλικά σημεία) και κλάση 2 (500 τριγωνικά σημεία)

Για τα σημεία της κλάσης 1 (κυκλικά σημεία):

$0.5 \leq \text{sqrt}(x_1^2+x_2^2) \leq 1$

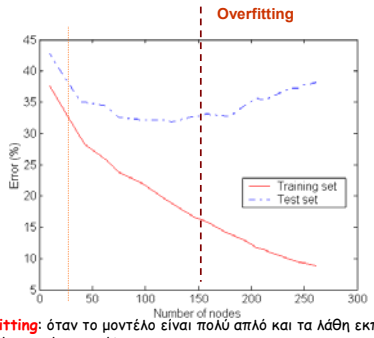
Για τα σημεία της κλάσης 2 (τριγωνικά σημεία):

$\text{sqrt}(x_1^2+x_2^2) > 0.5$ or

$\text{sqrt}(x_1^2+x_2^2) < 1$

Overfitting

"Everything should be made as simple as possible, but not simpler", Einstein



Το δέντρο απόφασης για το προηγούμενα δεδομένα

30% εκπαίδευση

70% έλεγχο

Gini

Στη συνέχεια, pruning

Underfitting: όταν το μοντέλο είναι πολύ απλό και τα λάθη εκπαίδευσης και τα λάθη ελέγχου είναι μεγάλα

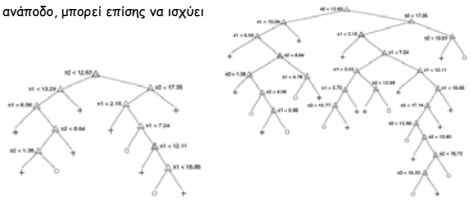
Overfitting

Μπορούμε να διασπάμε το δέντρο μέχρι να φτάσουμε στο σημείο κάθε φύλλο να ταιριάζει απολύτως στα δεδομένα

Μικρό (μηδενικό) λάθος εκπαίδευσης

Μεγάλο λάθος ελέγχου

Και το ανάποδο, μπορεί επίσης να ισχύει

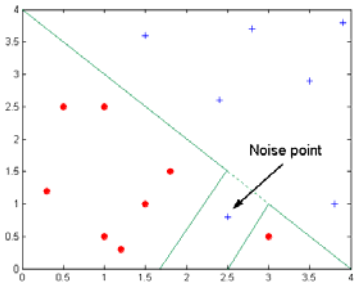


(a) Decision tree with 11 leaf nodes. (b) Decision tree with 24 leaf nodes.

Δέντρα απόφασης με διαφορετική πολυπλοκότητα

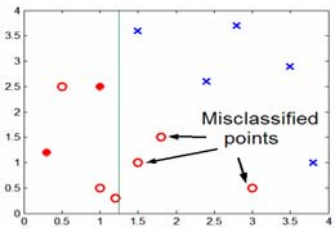
Overfitting

Overfitting εξαιτίας Θορύβου



Overfitting

Overfitting εξαιτίας μη Επαρκών Δειγμάτων



Κόκκινοι κύκλοι ανήκουν στην ίδια κλάση

Οι γεμάτοι είναι στο σύνολο εκπαίδευσης, οι άδειοι στο σύνολο ελέγχου

Η έλλειψη κόκκινων σημείων στο κάτω μισό του διαγράμματος κάνει δύσκολη την πρόβλεψη των κλάσεων σε αυτήν την περιοχή

Μη επαρκής αριθμός εγγραφών εκπαίδευσης έχει ως αποτέλεσμα το δέντρο απόφασης να κάνει πρόβλεψη για τα σημεία αυτής της περιοχής χρησιμοποιώντας εγγραφές εκπαίδευσης μη σχετικές με το έργο της ταξινόμησης

Overfitting

Πρόβλημα λόγω πολλαπλών επιλογών

- Επειδή σε κάθε βήμα εξετάζουμε πάρα πολλές διαφορετικές διασπάσεις,
  - κάποια διάσπαση βελτιώνει το δέντρο *κατά τύχη*

Το πρόβλημα χειροτερεύει όταν αυξάνει ο αριθμός των επιλογών και μειώνεται ο αριθμός των δειγμάτων

Overfitting

- Το overfitting έχει ως αποτέλεσμα δέντρα απόφασης που είναι πιο περίπλοκα από ό,τι χρειάζεται
- Τα λάθη εκπαίδευσης δεν αποτελούν πια μια καλή εκτίμηση για τη συμπεριφορά του δέντρου σε εγγραφές που δεν έχει δει ξανά
- Νέοι μέθοδοι για την εκτίμηση του λάθους

Αντιμετώπιση Overfitting

Δύο βασικές προσεγγίσεις:

Pre-pruning

Σταμάτημα της ανάπτυξης του δέντρου μετά από κάποιο σημείο

Post-pruning

Η κατασκευή του δέντρου χωρίζεται σε δύο φάσεις:

1. Φάση Ανάπτυξης
2. Φάση Ψαλιδίσματος

Αντιμετώπιση Overfitting

Pre-Pruning (Early Stopping Rule)

Σταμάτα τον αλγόριθμο πριν σχηματιστεί ένα πλήρες δέντρο

Συνήθεις συνθήκες τερματισμού για έναν κόμβο:
- Σταμάτα όταν όλες οι εγγραφές ανήκουν στην ίδια κλάση
- Σταμάτα όταν όλες οι τιμές των γνωρισμάτων είναι οι ίδιες

Πιο περιοριστικές συνθήκες:
- Σταμάτα όταν ο αριθμός των εγγραφών είναι μικρότερος από κάποιο προκαθορισμένο κατώφλι
- Σταμάτα όταν η επέκταση ενός κόμβου δεν βελτιώνει την καθαρότητα (π.χ., Gini ή information gain) ή το λάθος γενίκευσης περισσότερο από κάποιο κατώφλι.
  (-) δύσκολος ο καθορισμός του κατωφλιού,
  (-) αν και το κέρδος μικρό, κατοπινοί διαχωρισμοί μπορεί να καταλήξουν σε καλύτερα δέντρα

Overfitting

Post-pruning

- Ανάπτυξε το δέντρο πλήρως
- Trim – ψαλίδισε τους κόμβους bottom-up
- Αν το λάθος γενίκευσης μειώνεται με το ψαλίδισμα, αντικατέστησε το υποδέντρο με
  - ένα φύλλο - οι ετικέτες κλάσεις του φύλλου καθορίζεται από την πλειοψηφία των κλάσεων των εγγραφών του υποδέντρου (subtree replacement)
  - ένα από τα κλαδιά του (Branch), αυτό που χρησιμοποιείται συχνότερα (subtree raising)
- Χρησιμοποιείται πιο συχνά
- Χρήση άλλων δεδομένων για τον υπολογισμό του καλύτερου δέντρου (δηλαδή του λάθους γενίκευσης)

Εκτίμηση του Λάθους Γενίκευσης

- Re-substitution errors: Λάθος στην εκπαίδευση (Σ e(t))
- Generalization errors: Λάθος στον έλεγχο (Σ e'(t))

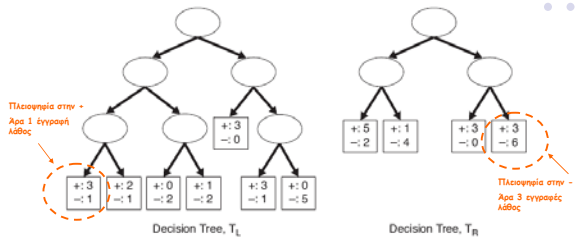
Ως λάθος μετράμε το ποσοστό των εγγραφών που ο ταξινομητής τοποθετεί σε λάθος κλάση

Μέθοδοι εκτίμησης του λάθους γενίκευσης:

1. Optimistic approach – Αισιόδοξη προσέγγιση:

$$e'(t) = e(t)$$

Εκτίμηση του Λάθους Γενίκευσης



Παράδειγμα δύο δέντρων για τα ίδια δεδομένα – Το δέντρο στο δεξί (T_R) μετά από ψαλίδισμα του δέντρου στα αριστερά (T_L)

Με βάση το λάθος εκπαίδευσης

Αριστερό 4/24 = 0.167 Δεξί: 6/24 = 0.25

Πολυπλοκότητα Μοντέλου

Occam's Razor

- Δοθέντων δυο μοντέλων με παρόμοια λάθη γενίκευσης, πρέπει να προτιμάται το απλούστερο από το πιο περίπλοκο
- Ένα πολύπλοκο μοντέλο είναι πιο πιθανό να έχει ταιριαστεί (Fitted) τυχαία λόγω λαθών στα δεδομένα
- Για αυτό η πολυπλοκότητα του μοντέλου θα πρέπει να αποτελεί έναν από τους παράγοντες της αξιολόγησής του

Εκτίμηση του Λάθους Γενίκευσης

2. Pessimistic approach - Απαισιόδοξη προσέγγιση:

k: αριθμός φύλλων,

για κάθε φύλλο t_i προσθέτουμε ένα κόστος $V(t_i)$

$$e'(T) = \frac{\sum_{i=1}^{k}[e(t_i)+V(t_i)]}{\sum_{i=1}^{k} n(t_i)}$$

Αν για κάθε φύλλο t: $e'(t) = e(t) + 0.5$

Συνολικό λάθος: $e'(T) = e(T) + k \times 0.5$ (k: αριθμός φύλλων)

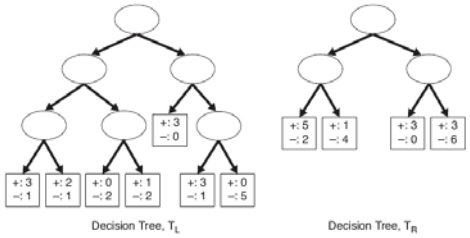
Για ένα δέντρο με 30 φύλλα και 10 λάθη στο σύνολο εκπαίδευσης (από σύνολο 1000 εγγραφών):

Training error = 10/1000 = 1%

Generalization error = (10 + 30×0.5)/1000 = 2.5%

Το 0.5 σημαίνει ότι διαχωρισμός ενός κόμβου δικαιολογείται αν βελτιώνει τουλάχιστον μία εγγραφή

Εκτίμηση του Λάθους Γενίκευσης



Decision Tree, T_L Decision Tree, T_R

Παράδειγμα δύο δέντρων για τα ίδια δεδομένα

Με βάση το λάθος εκπαίδευσης

Αριστερό (4 + 7*0.5)/24 = 0.3125

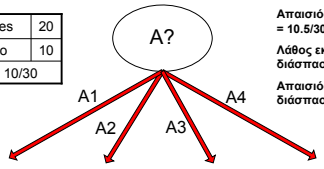
Δεξί: (6 + 4*0.5)/24 = 0.3333

Αν αντί για 0.5, κάτι μεγαλύτερο;

Overfitting

Παράδειγμα Post-Pruning

Class = Yes	20
Class = No	10
Error = 10/30	



Λάθος εκπαίδευσης (Πριν τη διάσπαση) = 10/30

Απαισιόδοξο λάθος = (10 + 0.5)/30 = 10.5/30

Λάθος εκπαίδευσης (Μετά τη διάσπαση) = 9/30

Απαισιόδοξο λάθος (Μετά τη διάσπαση)

= (9 + 4 x 0.5)/30 = 11/30

PRUNE (ΨΑΛΙΔΙΣΕ)!

Class = Yes	8
Class = No	4

Class = Yes	4
Class = No	1

Class = Yes	5
Class = No	1

Class = Yes	3
Class = No	4

Εκτίμηση του Λάθους Γενίκευσης

3. Reduced error pruning (REP):

- χρήση ενός συνόλου επαλήθευσης για την εκτίμηση του λάθους γενίκευσης

Χώρισε τα δεδομένα εκπαίδευσης:

2/3 εκπαίδευση

1/3 (σύνολο επαλήθευσης – validation set) για υπολογισμό λάθους

Χρήση για εύρεση του κατάλληλου μοντέλου

Overfitting

Παράδειγμα post-pruning

- Αισιόδοξη προσέγγιση?

  Όχι διάσπαση

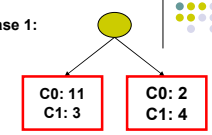
- Απαισιόδοξη προσέγγιση?

  όχι case 1, ναι case 2

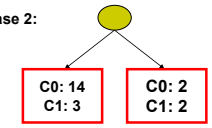
- REP?

Εξαρτάται από το σύνολο επαλήθευσης

Case 1:



Case 2:



Τιμές που λείπουν

Τιμές που λείπουν

- Οι τιμές που λείπουν επηρεάζουν την κατασκευή του δέντρου με τρεις τρόπους:
  - Πως υπολογίζονται τα μέτρα καθαρότητας
  - Πως κατανέμονται στα φύλλα οι εγγραφές με τιμές που λείπουν
  - Πως ταξινομείται μια εγγραφή στην οποία λείπει μια τιμή

Τιμές που λείπουν

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	?	Single	90K	Yes

Missing value

Υπολογισμό μέτρων καθαρότητας

Πριν τη διάσπαση:

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

	Class = Yes	Class = No
Refund=Yes	0	3
Refund=No	2	4
Refund=?	1	0

Διάσπαση στο Refund:

Entropy(Refund=Yes) = 0

Entropy(Refund=No)
= -(2/6)log(2/6) – (4/6)log(4/6) = 0.9183

Entropy(Children)
= **0.3** (0) + **0.6** (0.9183) = 0.551

Gain = **0.9** × (0.8813 – 0.551) = 0.3303

Τιμές που λείπουν

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

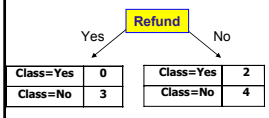
Tid	Refund	Marital Status	Taxable Income	Class
10	?	Single	90K	Yes

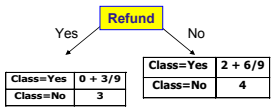
Σε ποιο φύλλο;

Πιθανότητα Refund=Yes is 3/9 (3 από τις 9 εγγραφές έχουν refund=Yes)

Πιθανότητα Refund=No is 6/9

Ανάθεση εγγραφής στο αριστερό παιδί με βάρος 3/9 και στο δεξί παιδί με βάρος 6/9



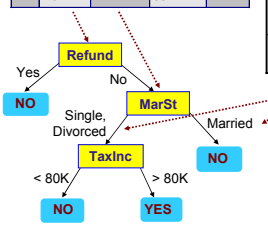


Τιμές που λείπουν

Νέα εγγραφή

Tid	Refund	Marital Status	Taxable Income	Class
11	No	?	85K	?

	Married	Single	Divorced	Total
Class=No	3	1	0	4
Class=Yes	6/9	1	1	2.67
Total	3.67	2	1	6.67



Πιθανότητα οικογενειακή κατάσταση (MarSt) = Married is 3.67/6.67

Πιθανότητα οικογενειακή κατάσταση (MarSt) ={Single,Divorced} is 3/6.67

Αποτίμηση Μοντέλου

Επιλογή Μοντέλου (model selection): το μοντέλο που έχει την απαιτούμενη πολυπλοκότητα χρησιμοποιώντας την εκτίμηση του λάθους γενίκευσης

Αφού κατασκευαστεί μπορεί να χρησιμοποιηθεί στα δεδομένα ελέγχου για να προβλέψει σε ποιες κλάσεις ανήκουν

Για να γίνει αυτό πρέπει να ξέρουμε τις κλάσεις των δεδομένων ελέγχου