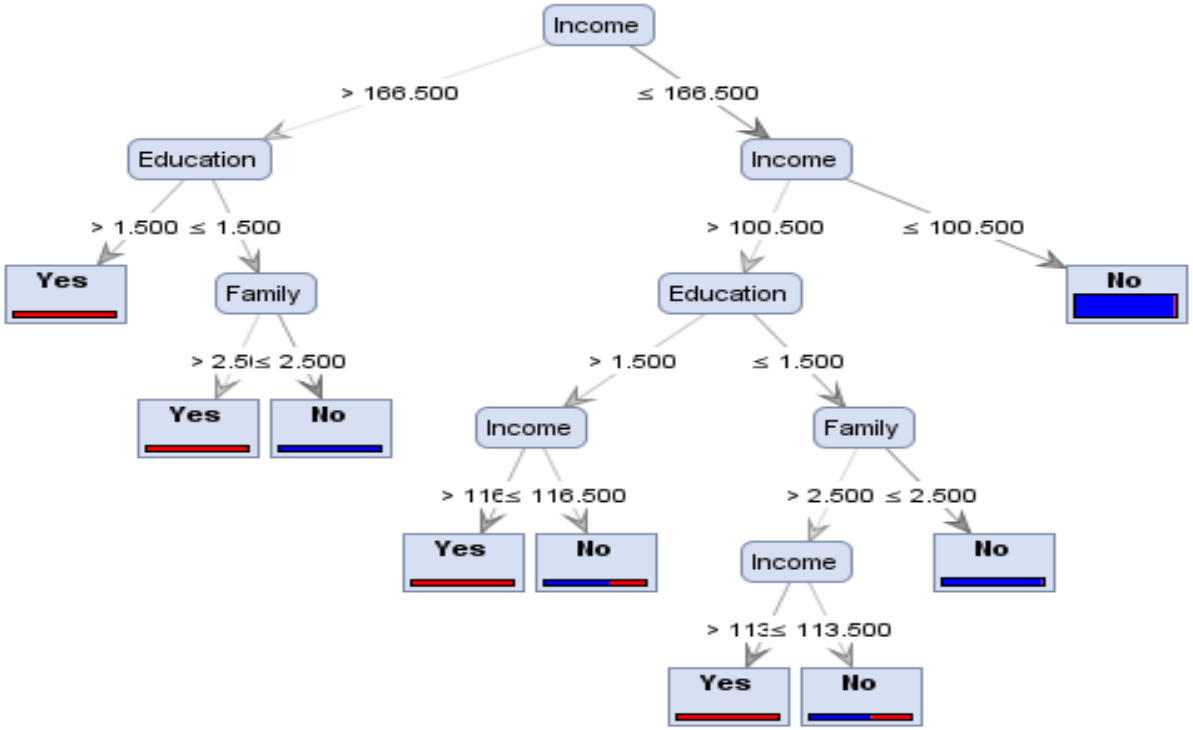
Εξόρυξη Δεδομένων

Κατηγοριοποίηση:

Βασικές Έννοιες, Δέντρα Απόφασης, Μοντέλα Αξιολόγησης

(Σημειώσεις μεταφρασμένες από το Κεφάλαιο 4 του βιβλίου των Tan, Steinbach, Kumar)



Κατηγοριοποίηση: Ορισμός

- Δίνεται μία συλλογή από εγγραφές (*training set*)
  - Κάθε εγγραφή (record) περιλαμβάνει ένα σύνολο από *χαρακτηριστικά*, από τα οποία το ένα θα είναι η *κατηγορία*.
- Πρέπει να βρεθεί ένα *μοντέλο* για το χαρακτηριστικό της κατηγορίας το οποίο να είναι συνάρτηση των τιμών των άλλων χαρακτηριστικών.
- Στόχος: οι εγγραφές που δεν έχουν εμφανιστεί ακόμα πρέπει να τοποθετηθούν σε μία κατηγορία με όσο πιο μεγάλη ακρίβεια γίνεται.
  - Ένα *test set* χρησιμοποιείται για να προσδιορίσει την ακρίβεια του μοντέλου. Συνήθως τα διαθέσιμα δεδομένα χωρίζονται σε σύνολα training και test, όπου το training set χρησιμοποιείται για κα κατασκευαστεί το μοντέλο ενώ το test set για να το επαληθεύσει.

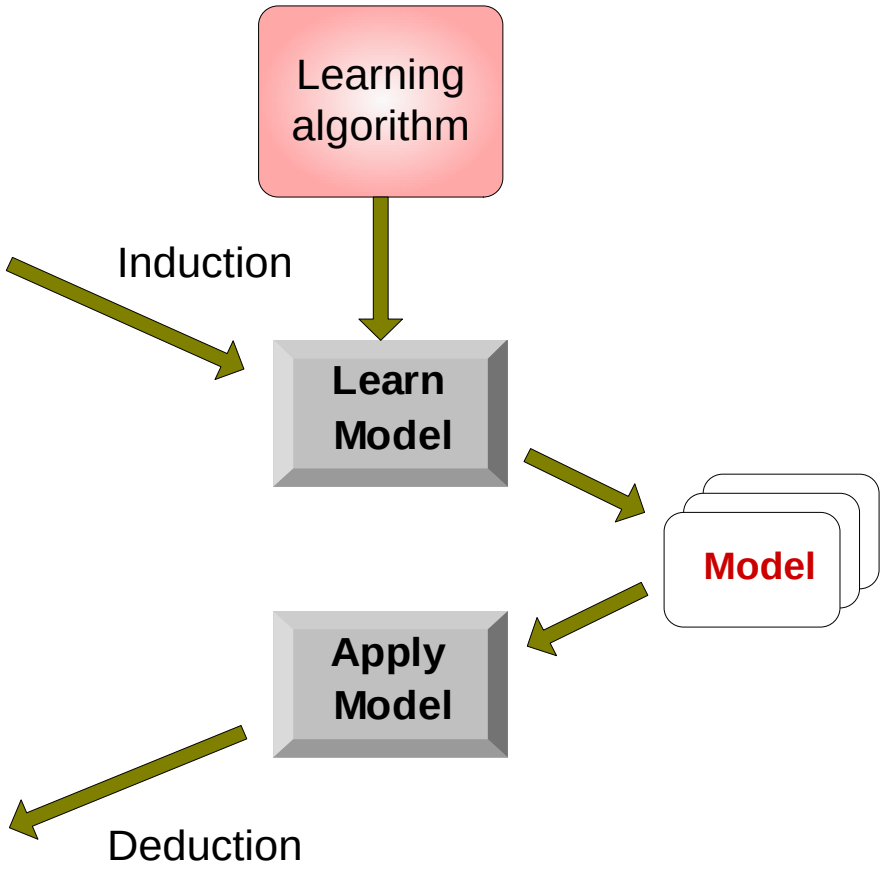
Απεικόνιση της διαδικασίας της Κατηγοριοποίησης

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

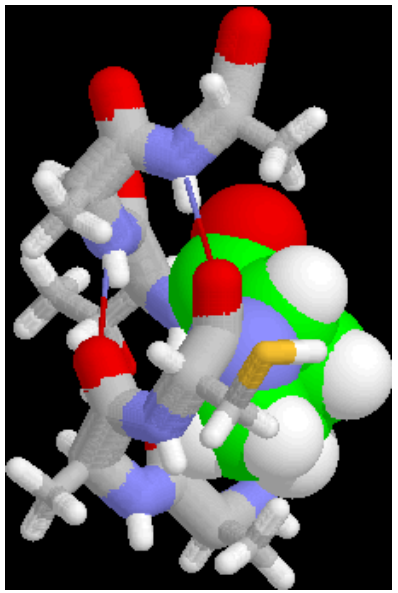
Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set



Παραδείγματα Κατηγοριοποίησης

- Πρόβλεψη κυττάρων όγκου αν είναι καλοήθη ή κακοήθη
- Κατηγοριοποίηση συναλλαγών από πιστωτικές κάρτες ως νόμιμες ή μη
- Κατηγοριοποίηση δευτερεύουσας δομής πρωτεϊνών ως άλφα-έλικας ή βήτα ή τυχαίο σπείραμα
- Κατηγοριοποίηση των ειδήσεων ως οικονομικές, καιρού, ψυχαγωγίας, αθλητικές, κλπ.



Μέθοδοι Κατηγοριοποίησης

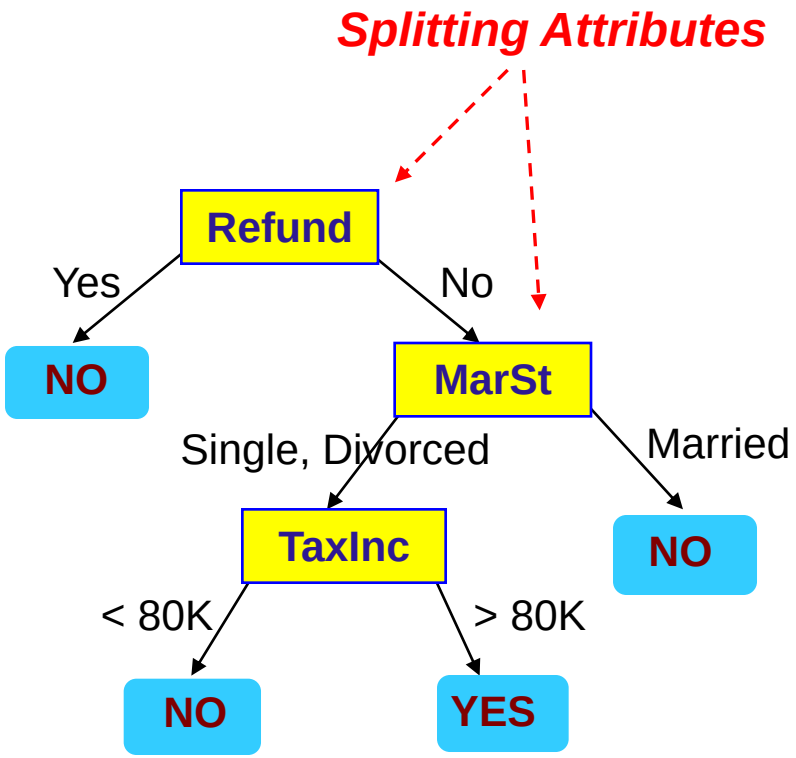
- Μέθοδοι που εφαρμόζουν **Δέντρα Απόφασης**
- Μέθοδοι που βασίζονται σε Κανόνες
- Μέθοδοι που βασίζονται σε απομνημόνευση
- Νευρωνικά Δίκτυα
- Naïve Bayes και Bayesian Belief Networks
- Support Vector Machines

Παράδειγμα ενός Δέντρου Απόφασης

categorical categorical continuous class

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

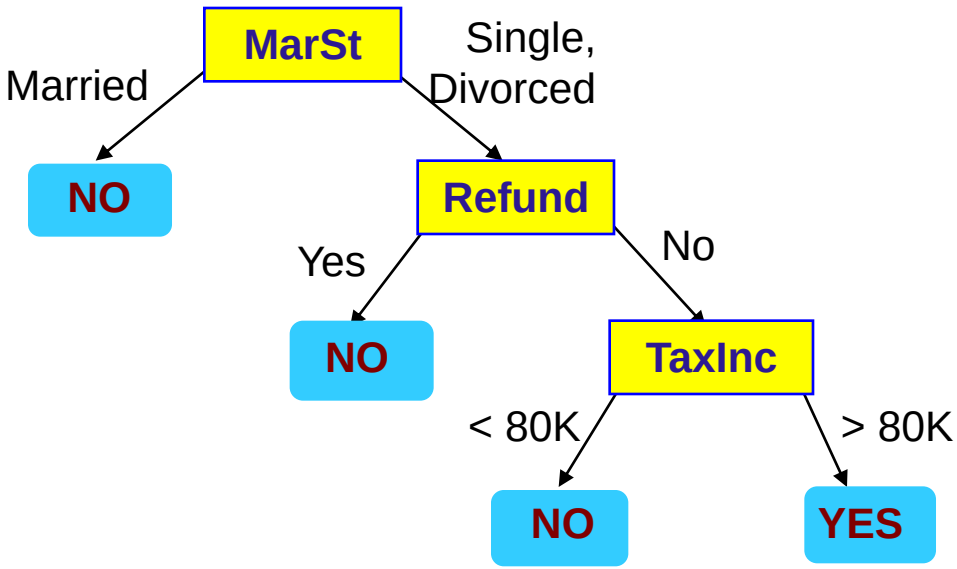
Training Data



Model: Decision Tree

Παράδειγμα Δέντρου Απόφασης (2ο)

Tid	Refund	Marital Status	Taxable Income	Cheat
		categorical	categorical	continuous	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



Μπορεί να υπάρχουν περισσότερα από ένα δέντρα που ταιριάζουν στα ίδια δεδομένα!

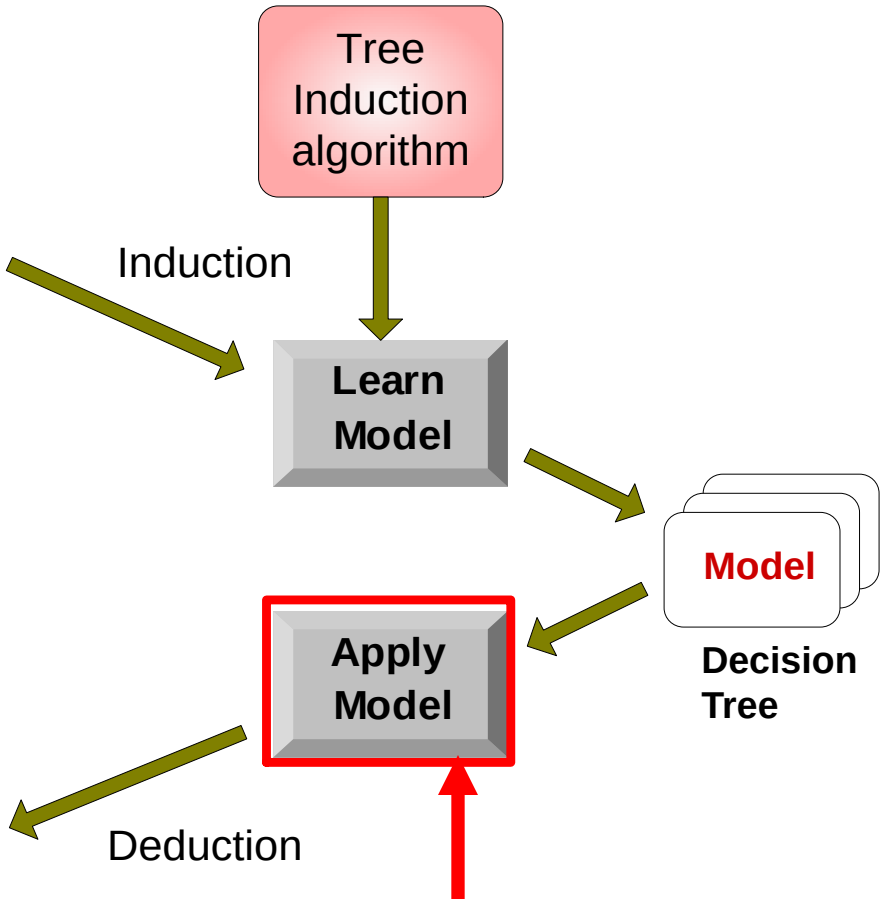
Διαδικασία Κατηγοριοποίησης με Δέντρο Απόφασης

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

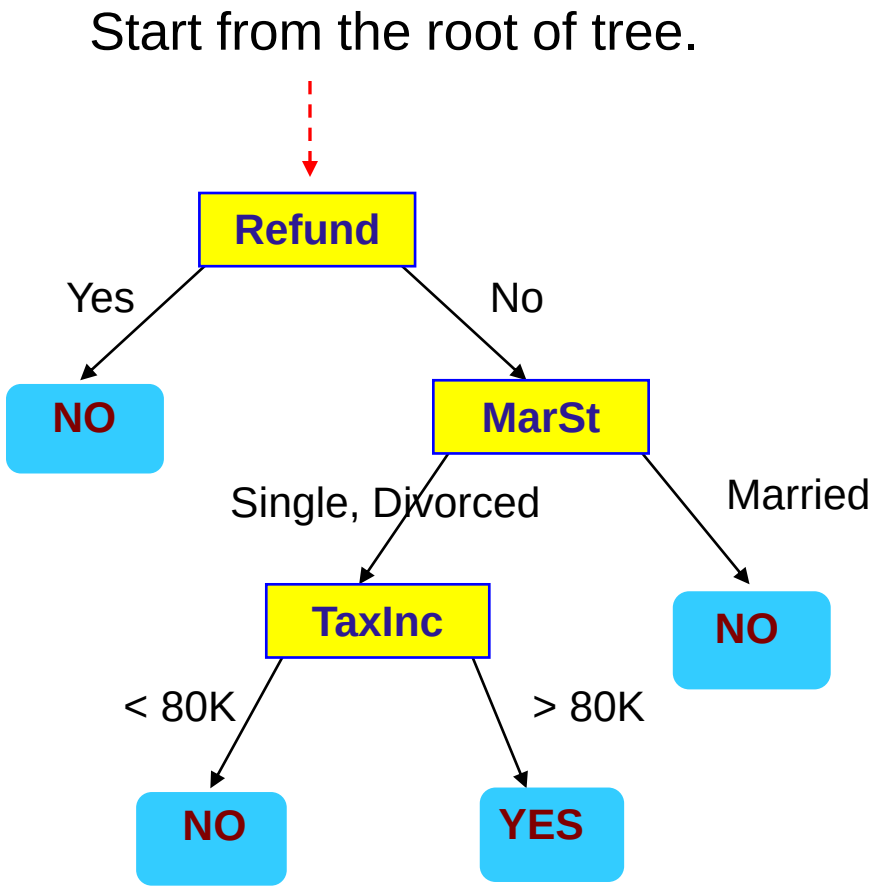
Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set



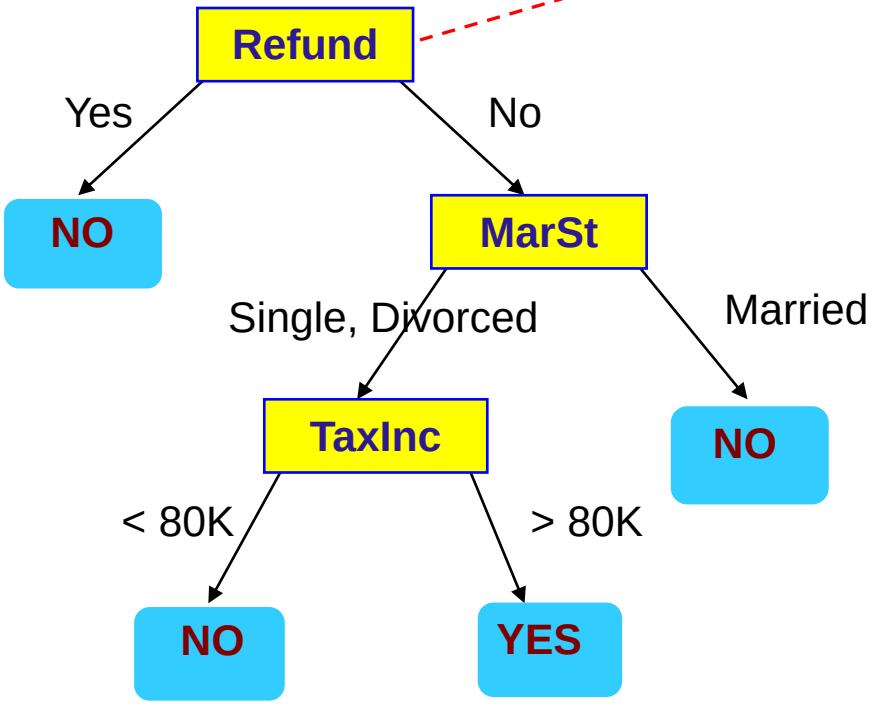
Εφαρμογή Μοντέλου στα Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

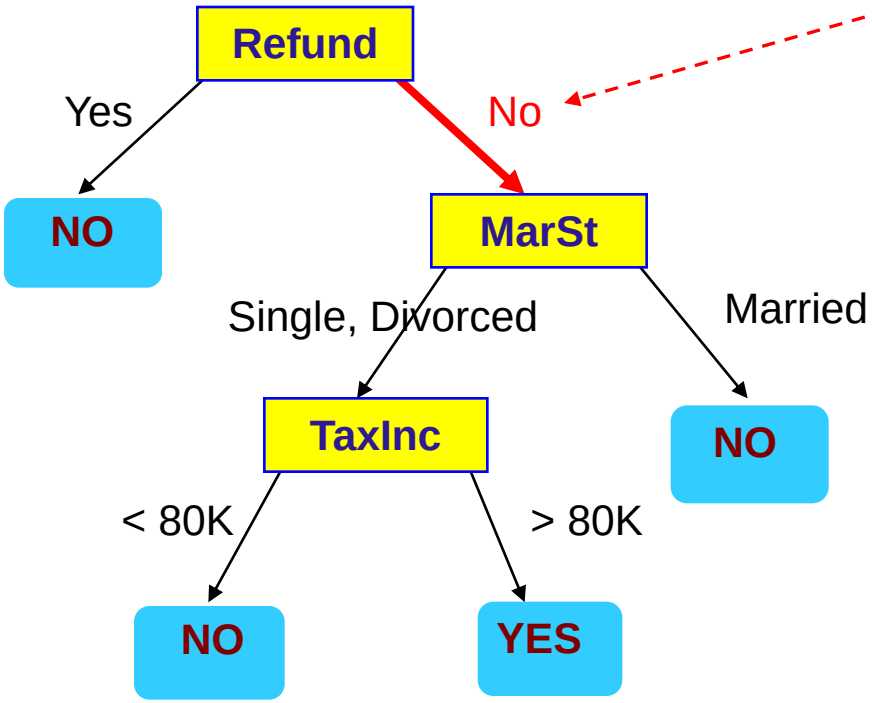
Εφαρμογή Μοντέλου στα Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

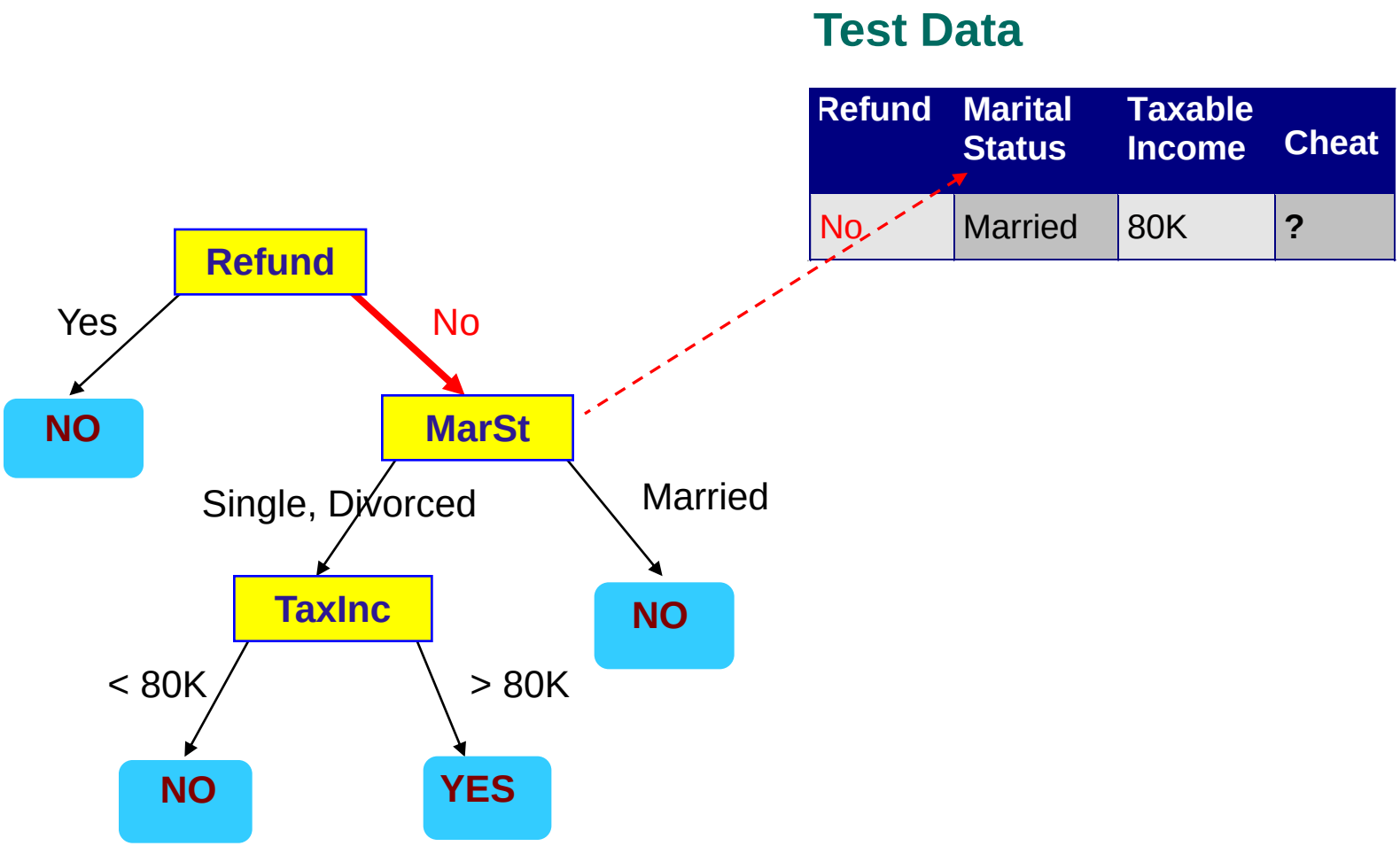
Εφαρμογή Μοντέλου στα Test Data



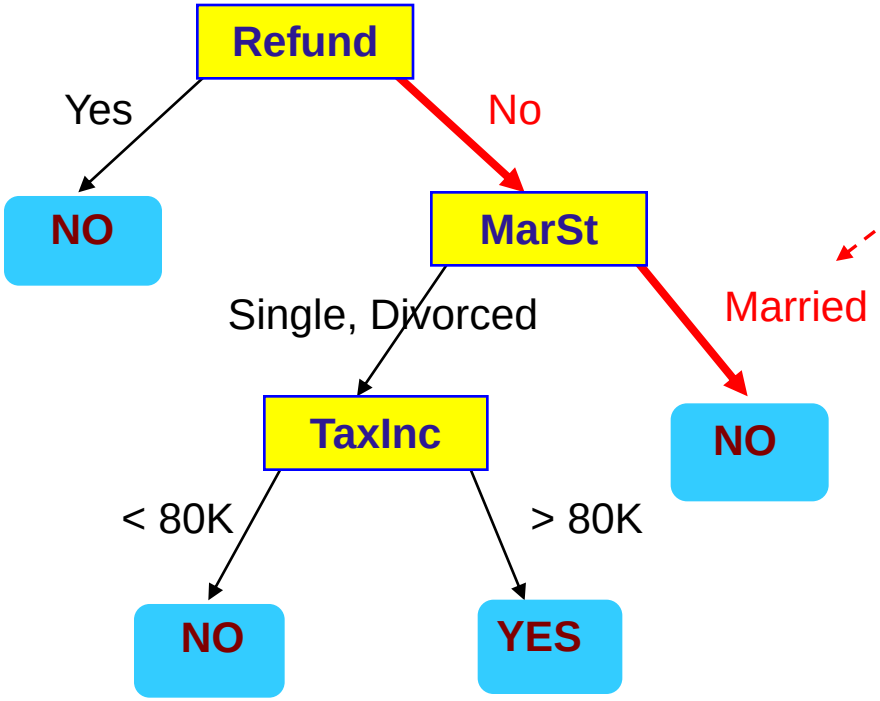
Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Εφαρμογή Μοντέλου στα Test Data



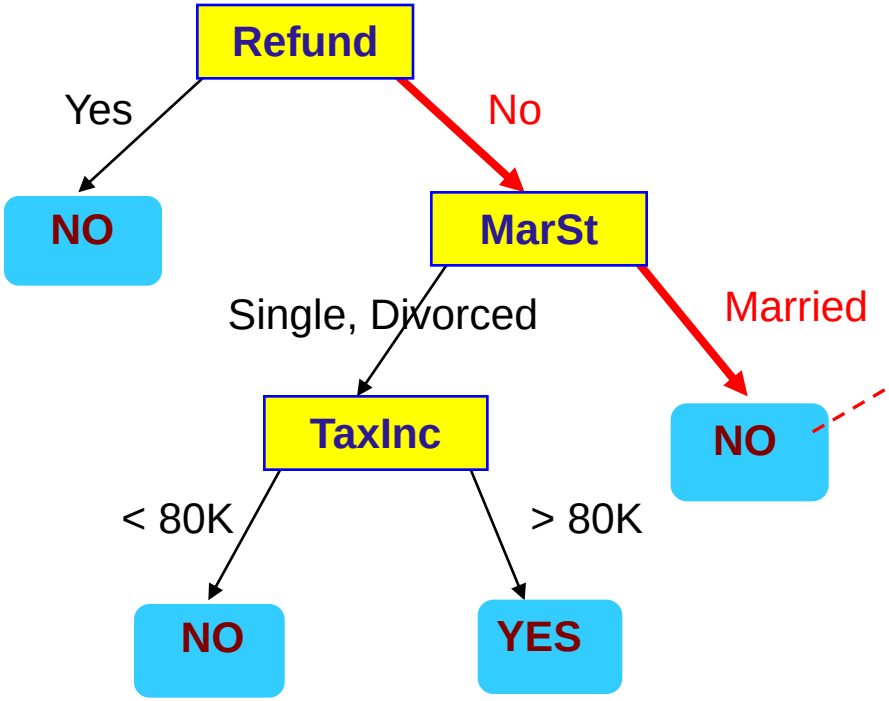
Εφαρμογή Μοντέλου στα Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Εφαρμογή Μοντέλου στα Test Data



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Assign Cheat to “No”

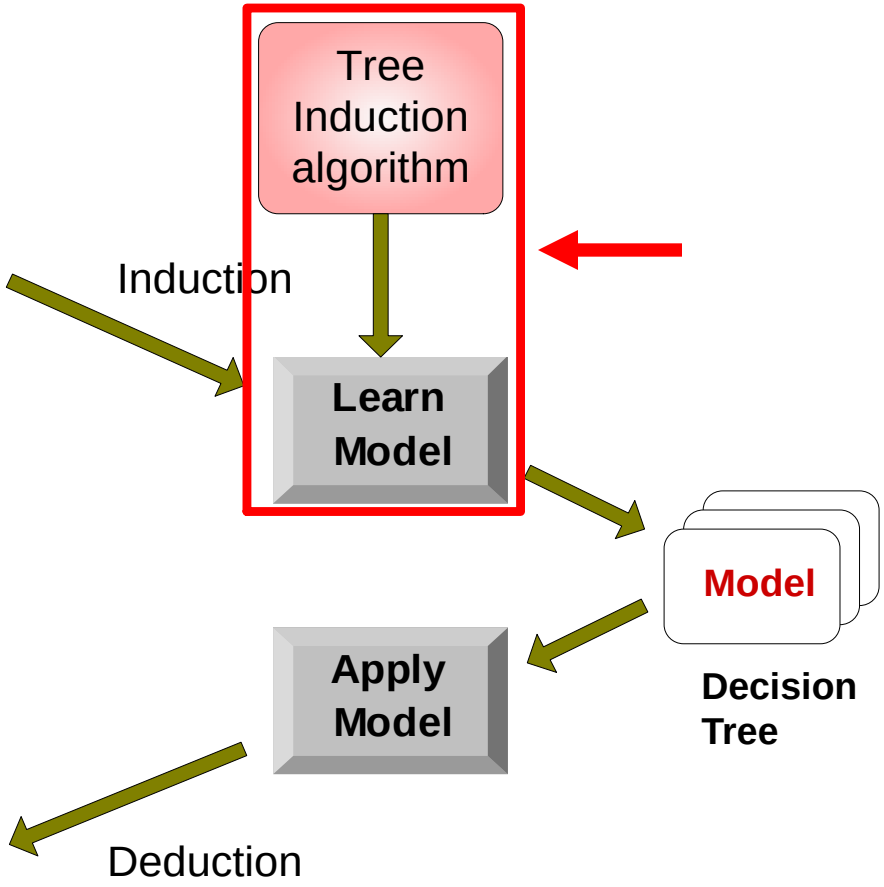
Διαδικασία Κατηγοριοποίησης με Δέντρο Απόφασης

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set



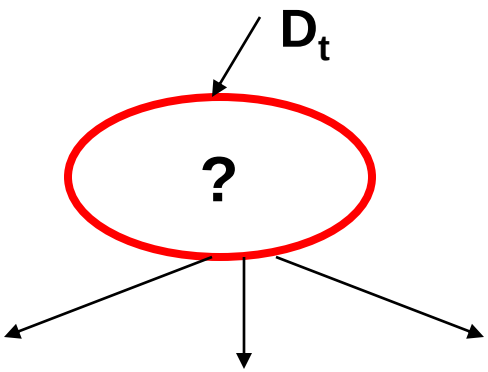
Παραγωγή του Δέντρου Απόφασης

- Υπάρχουν αρκετοί αλγόριθμοι:
  - Hunt's Algorithm
  - CART
  - ID3, C4.5
  - SLIQ,SPRINT

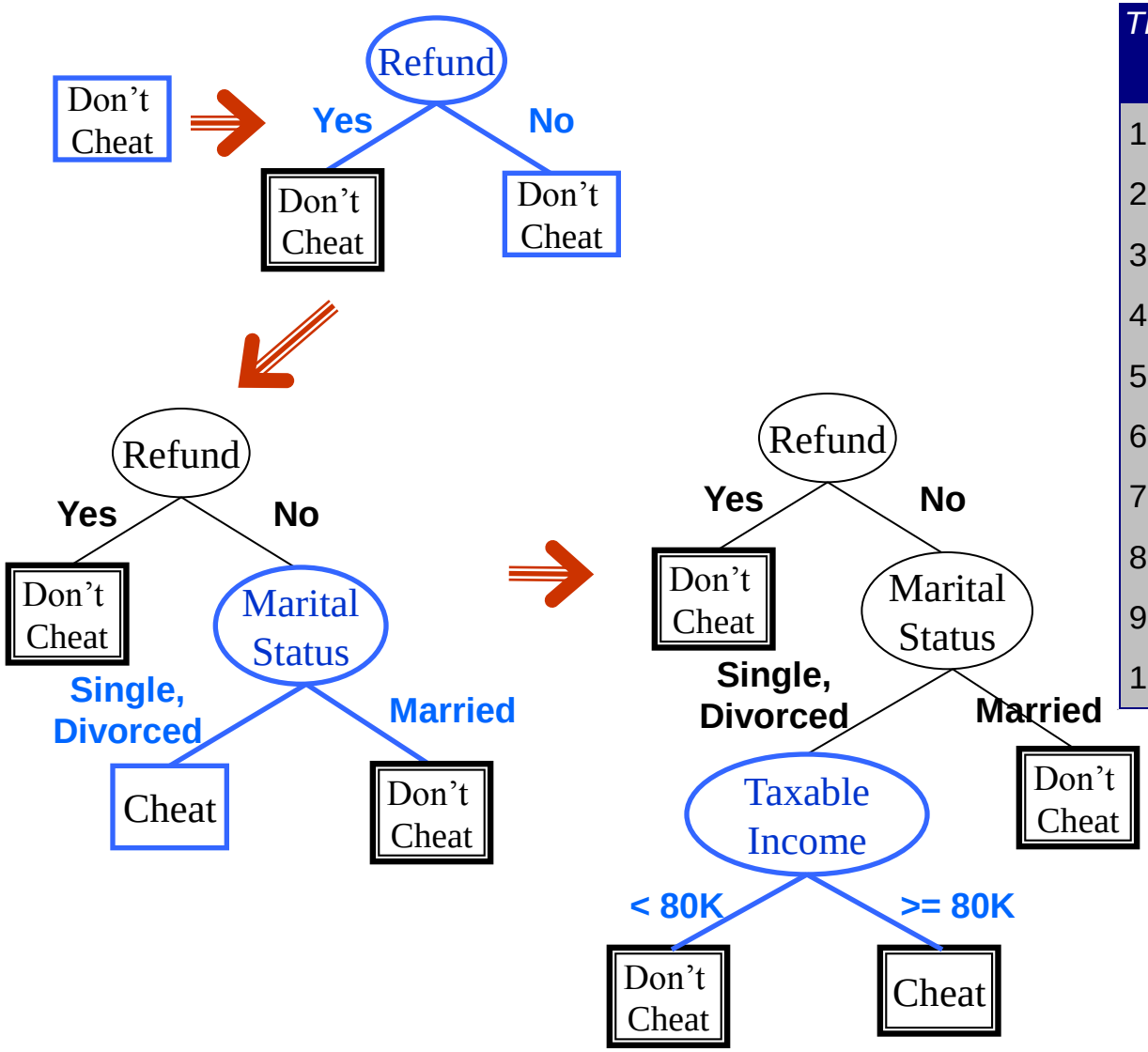
Γενική δομή του αλγορίθμου του Hunt

- Έστω D_t ένα σύνολο από εγγραφές training που επαληθεύονται μέχρι έναν κόμβο t
- Γενική διαδικασία:
  - Αν στο D_t περιλαμβάνονται εγγραφές που ανήκουν στην ίδια κατηγορία y_t, τότε ο t γίνεται κόμβος φύλλο με ετικέτα y_t
  - Αν το D_t είναι κενό σύνολο, τότε ο t γίνεται κόμβος φύλλο με ετικέτα την default τιμή y_d
  - Αν το D_t περιλαμβάνει εγγραφές που ανήκουν σε περισσότερες από μία κατηγορίες, γίνεται ένα τεστ στα χαρακτηριστικά ώστε να διαχωριστούν τα δεδομένα σε μικρότερα σύνολα. Εφαρμόζεται η διαδικασία αναδρομικά σε κάθε υποσύνολο.

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



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

Η Παραγωγή του Δέντρου

- Άπληστη στρατηγική:
  - Γίνεται διαχωρισμός των εγγραφών με βάση ένα τεστ στα χαρακτηριστικά, τα οποία βελτιστοποιεί ένα συγκεκριμένο κριτήριο.

- Θέματα που τίθενται:
  - Πώς θα χωρίσουν οι εγγραφές
    - Πώς θα οριστεί η συνθήκη στο τεστ χαρακτηριστικών;
    - Πώς θα οριστεί ο βέλτιστος διαχωρισμός;
  - Πότε θα σταματάει ο διαχωρισμός

Η Παραγωγή του Δέντρου

- Άπληστη στρατηγική:
  - Γίνεται διαχωρισμός των εγγραφών με βάση ένα τεστ στα χαρακτηριστικά, τα οποία βελτιστοποιεί ένα συγκεκριμένο κριτήριο.

- Θέματα που τίθενται:
  - Πώς θα χωρίσουν οι εγγραφές
    - Πώς θα οριστεί η συνθήκη στο τεστ χαρακτηριστικών;
    - Πώς θα οριστεί ο βέλτιστος διαχωρισμός;
  - Πότε θα σταματάει ο διαχωρισμός

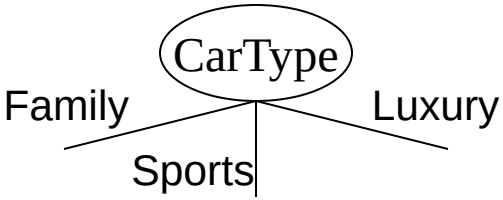
Πώς θα οριστεί η συνθήκη στο τεστ χαρακτηριστικών

- Εξαρτάται από τους τύπους των χαρακτηριστικών:
  - Λεκτικοί
  - Διατεταγμένοι, Κατηγορηματικοί
  - Συνεχείς

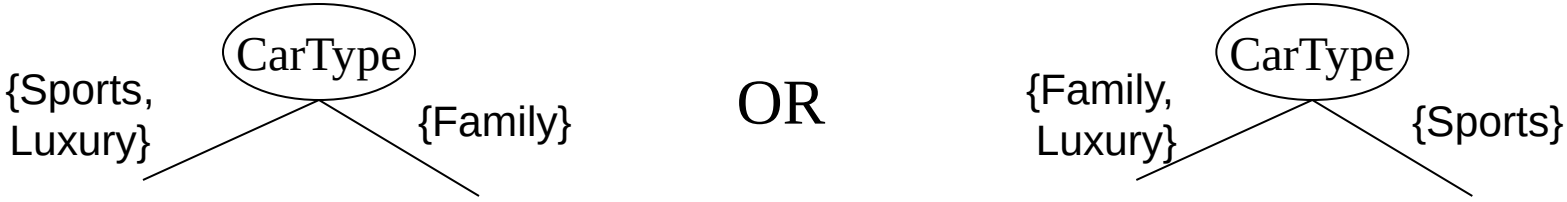
- Εξαρτάται από το πλήθος των μερών του διαχωρισμού:
  - 2-way split (διαχωρισμός σε 2 μέρη)
  - Multi-way split (διαχωρισμός σε περισσότερα)

Διαχωρισμός σε Λεκτικά χαρακτηριστικά

- Multi-way split: χρησιμοποιούνται τόσες διαμερίσεις όσες είναι και οι διακριτές τιμές:

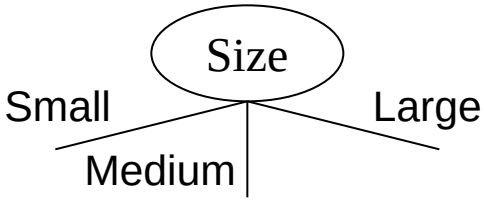


- Binary split: χωρίζονται οι τιμές σε δύο υποσύνολα. Πρέπει να βρεθεί ο βέλτιστος διαχωρισμός:

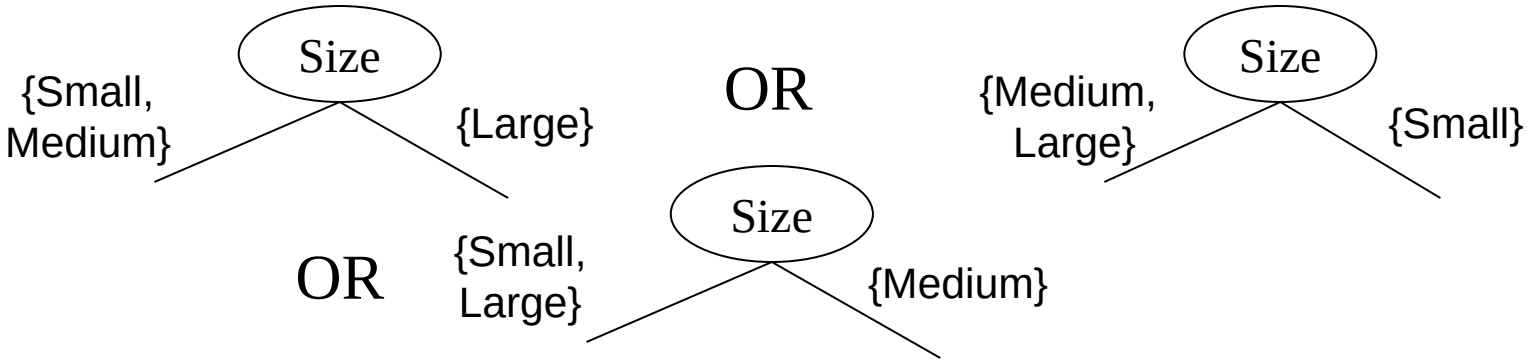


Διαχωρισμός σε Λεκτικά χαρακτηριστικά

- Multi-way split: χρησιμοποιούνται τόσες διαμερίσεις όσες είναι και οι διακριτές τιμές:



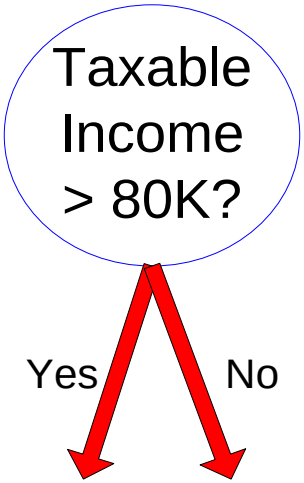
- Binary split: χωρίζονται οι τιμές σε δύο υποσύνολα. Πρέπει να βρεθεί ο βέλτιστος διαχωρισμός:



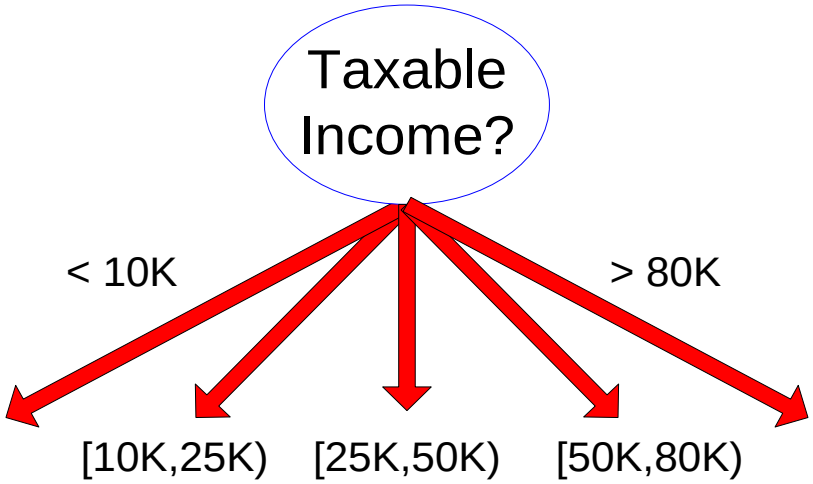
Διαχωρισμός σε Συνεχή χαρακτηριστικά

- Διαφορετικοί τρόποι χειρισμού:
  - Διακριτοποίηση ώστε να σχηματιστεί ένα διατεταγμένο κατηγορηματικό χαρακτηριστικό
    - Στατική – διακριτοποίηση μόνο στην αρχή
    - Δυναμική – το εύρος των διαστημάτων μπορεί να βρεθεί από μεθόδους equal interval bucketing, equal frequency bucketing (percentiles), ή clustering.
  - Δυαδική Απόφαση: $(A < v)$ ή $(A \geq v)$
    - εξετάζονται όλοι οι πιθανοί διαχωρισμοί και εντοπίζεται ο καλύτερος
    - αλλά, η μέθοδος αυτή μπορεί να γίνει πολύ απαιτητική σε υπολογισμούς

Διαχωρισμός σε Συνεχή χαρακτηριστικά



(i) Binary split



(ii) Multi-way split

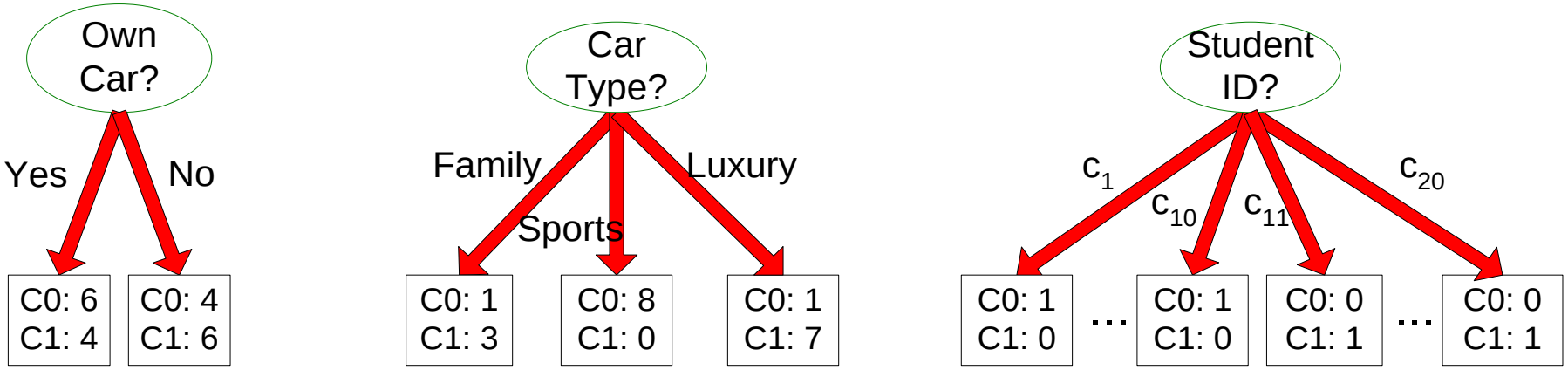
Η Παραγωγή του Δέντρου

- Άπληστη στρατηγική:
  - Γίνεται διαχωρισμός των εγγραφών με βάση ένα τεστ στα χαρακτηριστικά, τα οποία βελτιστοποιεί ένα συγκεκριμένο κριτήριο.

- Θέματα που τίθενται:
  - Πώς θα χωρίσουν οι εγγραφές
    - Πώς θα οριστεί η συνθήκη στο τεστ χαρακτηριστικών;
    - Πώς θα οριστεί ο βέλτιστος διαχωρισμός;
  - Πότε θα σταματάει ο διαχωρισμός

Πώς θα οριστεί ο βέλτιστος διαχωρισμός

Πριν τον διαχωρισμό υπάρχουν: 10 εγγραφές στην κατηγορία 0 (C0), και 10 εγγραφές στην κατηγορία 1 (C1)



Πώς θα κρίνουμε ποια συνθήκη θα είναι η καλύτερη;

Πώς θα οριστεί ο βέλτιστος διαχωρισμός

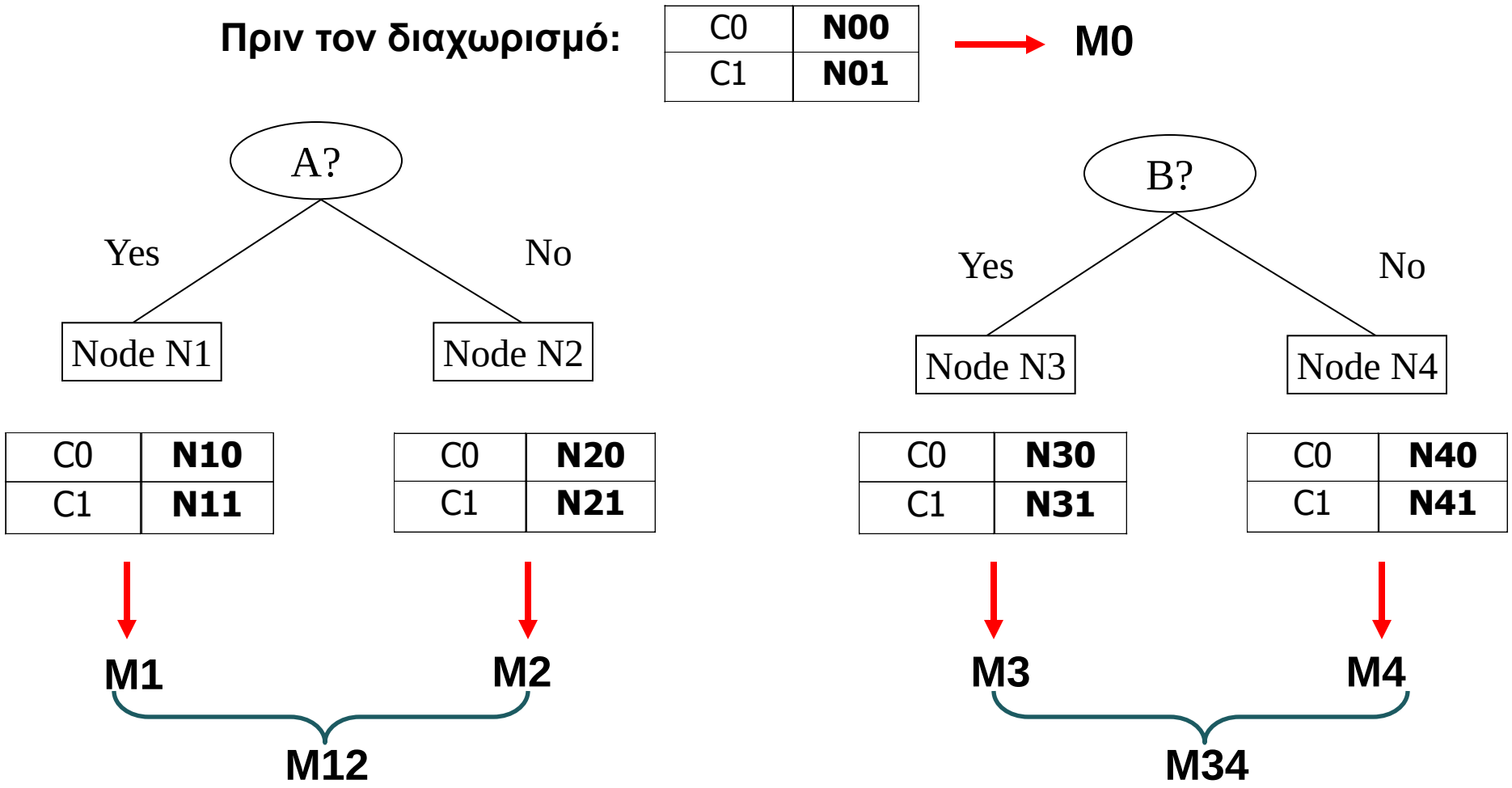
- Άπληστη στρατηγική:
  - Προτιμούνται οι κόμβοι που εμφανίζουν ομοιογενή κατανομή
- Χρειάζεται ένα μέτρο για το πόσο οι εγγραφές έχουν αναμειχθεί (διασπαρθεί) στους κόμβους:

C0: 5 C1: 5	C0: 9 C1: 1
Μη-ομοιογενή, **Υψηλός βαθμός ανάμειξης**	**Ομοιογενή,** **Χαμηλός βαθμός ανάμειξης**

Μέτρα ανάμειξης των κόμβων

- Δείκτης GINI (Gini Index)
- Εντροπία (Entropy)
- Σφάλμα Κατηγοριοποίησης (Misclassification error)

Πώς θα βρεθεί ο βέλτιστος διαχωρισμός



Συγκρίνουμε το M0 – M12 με το M0 – M34

Μέτρο ανάμειξης: GINI

- Ο δείκτης **Gini** για έναν δεδομένο κόμβο *t* ορίζεται ως:

$$GINI(t) = 1 - \sum_j [p(j \mid t)]^2$$

(όπου $p(j \mid t)$ είναι η σχετική συχνότητα της κατηγορίας *j* στον κόμβο *t*).

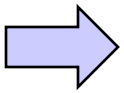
  - Μέγιστη τιμή: (1 - 1/n_c) όταν οι εγγραφές ισοκατανέμονται μεταξύ όλων των (n_c) κατηγοριών, παράγοντας την ελάχιστη ενδιαφέρουσα πληροφορία
  - Ελάχιστη τιμή: (0.0) όταν όλες οι εγγραφές βρίσκονται σε μία κατηγορία, παράγοντας την μέγιστη ενδιαφέρουσα πληροφορία

C1	**0**
C2	**6**
Gini=0.000	

C1	**1**
C2	**5**
Gini=0.278	

C1	**2**
C2	**4**
Gini=0.444	

C1	**3**
C2	**3**
Gini=0.500	



Παραδείγματα υπολογισμού του GINI

$$GINI(t) = 1 - \sum_j [p(j \mid t)]^2$$

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

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

Gini = 1 – P(C1)² – P(C2)² = 1 – 0 – 1 = 0

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

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

Gini = 1 – (1/6)² – (5/6)² = 0.278

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

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

Gini = 1 – (2/6)² – (4/6)² = 0.444

Διαχωρισμός με βάση τον δείκτη GINI

- Χρησιμοποιείται στις μεθόδους CART, SLIQ, SPRINT.
- Όταν ένας κόμβος *p* διαχωρίζεται σε *k* μέρη (παιδιά), τότε η ποιότητα του διαχωρισμού υπολογίζεται από τον τύπο:

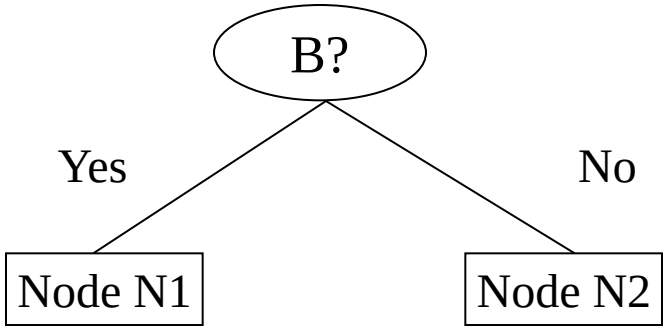
$$GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$$

όπου, n_i = πλήθος εγγραφών στο παιδί *i*,

n = πλήθος εγγραφών στον κόμβο *p*.

Υπολογισμός του δείκτη GINI (δυαδικά χαρακτηριστικά)

- Γίνεται διαχωρισμός σε δύο μέρη
- Ζύγιση των μερών: Αναζητούνται τα μεγαλύτερα και τα πιο αμιγή (ως προς τα σύνολα των εγγραφών) μέρη.



	Parent
C1	**6**
C2	**6**
Gini = 0.500	

Gini(N1)
= 1 – (5/6)² – (2/6)²
= 0.194

Gini(N2)
= 1 – (1/6)² – (4/6)²
= 0.528

	N1	N2
C1	**5**	**1**
C2	**2**	**4**
Gini=0.333		

Gini(Children)
= 7/12 * 0.194 +
5/12 * 0.528
= 0.333

Υπολογισμός του δείκτη GINI (κατηγορικά χαρακτηριστικά)

- Για κάθε διακριτή τιμή, συγκεντρώνονται οι μετρήσεις κάθε κατηγορίας του συνόλου δεδομένων
- Χρησιμοποιείται ο count matrix για τις αποφάσεις

Multi-way split

	CarType		
	Family	Sports	Luxury
C1	1	2	1
C2	4	1	1
Gini	0.393		

Two-way split
(εντοπίζεται ο βέλτιστος διαχωρισμός)

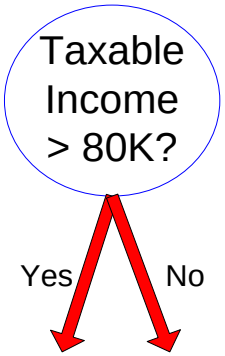
	CarType	
	{Sports, Luxury}	{Family}
C1	3	1
C2	2	4
Gini	0.400	

	CarType	
	{Sports}	{Family, Luxury}
C1	2	2
C2	1	5
Gini	0.419	

Υπολογισμός του δείκτη GINI (συνεχή χαρακτηριστικά)

- Χρησιμοποιούνται δυαδικές αποφάσεις με βάση μία τιμή v του χαρακτηριστικού A
- Υπάρχουν αρκετές επιλογές για την τιμή που θα γίνει διαχωρισμός
  - Πλήθος πιθανών διαχωρισμών = Πλήθος διακριτών τιμών
- Κάθε τιμή διαχωρισμού έχει και έναν count matrix που σχετίζεται με αυτή
  - Οι μετρήσεις στις κατηγορίες γίνονται για $A < v$ και για $A \geq v$
- Μία απλή μέθοδος για να βρεθεί η καλύτερη τιμή v είναι:
  - Για κάθε τιμή v, σαρώνεται η βάση δεδομένων για να συλλεχθούν τα στοιχεία του count matrix και να υπολογιστεί ο δείκτης GINI του
  - Μη αποτελεσματική υπολογιστικά μέθοδος (Επανάληψη υπολογισμών).

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



Υπολογισμός του δείκτη GINI (συνεχή χαρακτηριστικά)

- Πιο αποτελεσματική υπολογιστικά μέθοδος:
  - Ταξινομούνται οι εγγραφές ως προς τις τιμές του χαρακτηριστικού
  - Σαρώνονται γραμμικά οι τιμές αυτές, ενημερώνοντας κάθε φορά τον count matrix και υπολογίζοντας τον αντίστοιχο δείκτη GINI
  - Επιλέγεται η θέση διαχωρισμού που ελαχιστοποιεί τον δείκτη GINI

Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
Taxable Income										
Sorted Values →	60	70	75	85	90	95	100	120	125	220
Split Positions →	55	65	72	80	87	92	97	110	122	172	230

	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>	<=	>
Yes	0	3	0	3	0	3	0	3	1	2	2	1	3	0	3	0	3	0	3	0	3	0
No	0	7	1	6	2	5	3	4	3	4	3	4	3	4	4	3	5	2	6	1	7	0
Gini	0.420		0.400		0.375		0.343		0.417		0.400		***0.300***		0.343		0.375		0.400		0.420	

Εναλλακτικά Κριτήρια Διαχωρισμού

- Η εντροπία ενός κόμβου *t* ορίζεται ως εξής:

$$Entropy(t) = -\sum_j p(j \mid t) \log p(j \mid t)$$

(όπου $p(j \mid t)$ είναι η σχετική συχνότητα της κατηγορίας *j* στον κόμβο *t*).

  - Μετράει την ομοιογένεια ενός κόμβου
    - Μέγιστη τιμή: (log n_c) όταν οι εγγραφές είναι ισοκατανεμημένες σε όλες τις n_c κατηγορίες, παράγοντας την ελάχιστη πληροφορία
    - Ελάχιστη τιμή: (0.0) όταν όλες οι εγγραφές ανήκουν σε μία κατηγορία, παράγοντας την μέγιστη πληροφορία
  - Οι υπολογισμοί για την εντροπία είναι παρόμοιοι με τους υπολογισμούς για τον δείκτη GINI

Παραδείγματα υπολογισμού της Εντροπίας

$$Entropy(t) = -\sum_j p(j \mid t) \log_2 p(j \mid t)$$

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) $\log_2$ (1/6) = 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

Διαχωρισμός με βάση το Κέρδος Πληροφορίας

- Κέρδος Πληροφορίας (Information Gain):

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

Ο αρχικός κόμβος *p* (πατέρας) διαχωρίζεται σε *k* μέρη/σύνολα
n_i είναι το πλήθος των εγγραφών στο σύνολο *i*

- Μετράει την μείωση της εντροπίας που επιτυγχάνεται από τον διαχωρισμό. Επιλέγεται ο κατάλληλος διαχωρισμός που επιτυγχάνει την μεγαλύτερη μείωση (μεγιστοποιεί το κέρδος GAIN)
- Χρησιμοποιείται στους αλγορίθμους ID3 και C4.5
- Μειονέκτημα: Έχει την τάση να προτιμά διαχωρισμούς που έχουν ως αποτέλεσμα μεγάλο πλήθος υποσυνόλων που είναι μεν καθαρά αλλά είναι μικρά.

Διαχωρισμός με βάση το Κέρδος Πληροφορίας

- Λόγος Κέρδους (Gain Ratio):

$$GainRATIO_{split} = \frac{GAIN_{Split}}{SplitINFO} \qquad SplitINFO = -\sum_{i=1}^{k} \frac{n_i}{n} \log \frac{n_i}{n}$$

Ο αρχικός κόμβος *p* (πατέρας) διαχωρίζεται σε *k* μέρη/σύνολα

n_i είναι το πλήθος των εγγραφών στο σύνολο *i*

  - Ρυθμίζει το Κέρδος Πληροφορίας από την εντροπία του διαχωρισμού (SplitINFO). Η υψηλότερη εντροπία διαχωρισμού (μεγάλο πλήθος μικρών συνόλων) βαθμολογείται αρνητικά!
  - Χρησιμοποιείται στον αλγόριθμο C4.5
  - Σχεδιάστηκε για να ξεπεράσει το μειονέκτημα του Κέρδους Πληροφορίας

Διαχωρισμός με βάση το Σφάλμα Κατηγοριοποίησης

- Το Σφάλμα Κατηγοριοποίησης (Classification error) σε έναν κόμβο t ορίζεται ως εξής:

$$Error(t) = 1 - \max_i P(i \mid t)$$

(όπου $p(i \mid t)$ είναι η σχετική συχνότητα της κατηγορίας *i* στον κόμβο *t*)

- Μετρά το σφάλμα της λάθους κατηγοριοποίησης που έγινε σε έναν κόμβο.
  - Μέγιστη τιμή: $(1 - 1/n_c)$ όταν οι εγγραφές είναι ισοκατανεμημένες μεταξύ όλων των n_c κατηγοριών, παράγοντας την ελάχιστη ενδιαφέρουσα πληροφορία
  - Ελάχιστη τιμή: (0.0) όταν όλες οι εγγραφές ανήκουν σε μία κατηγορία, παράγοντας την μέγιστη ενδιαφέρουσα πληροφορία

Παραδείγματα υπολογισμού του Σφάλματος Κατηγ.

$$Error(t) = 1 - \max_i P(i \mid t)$$

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

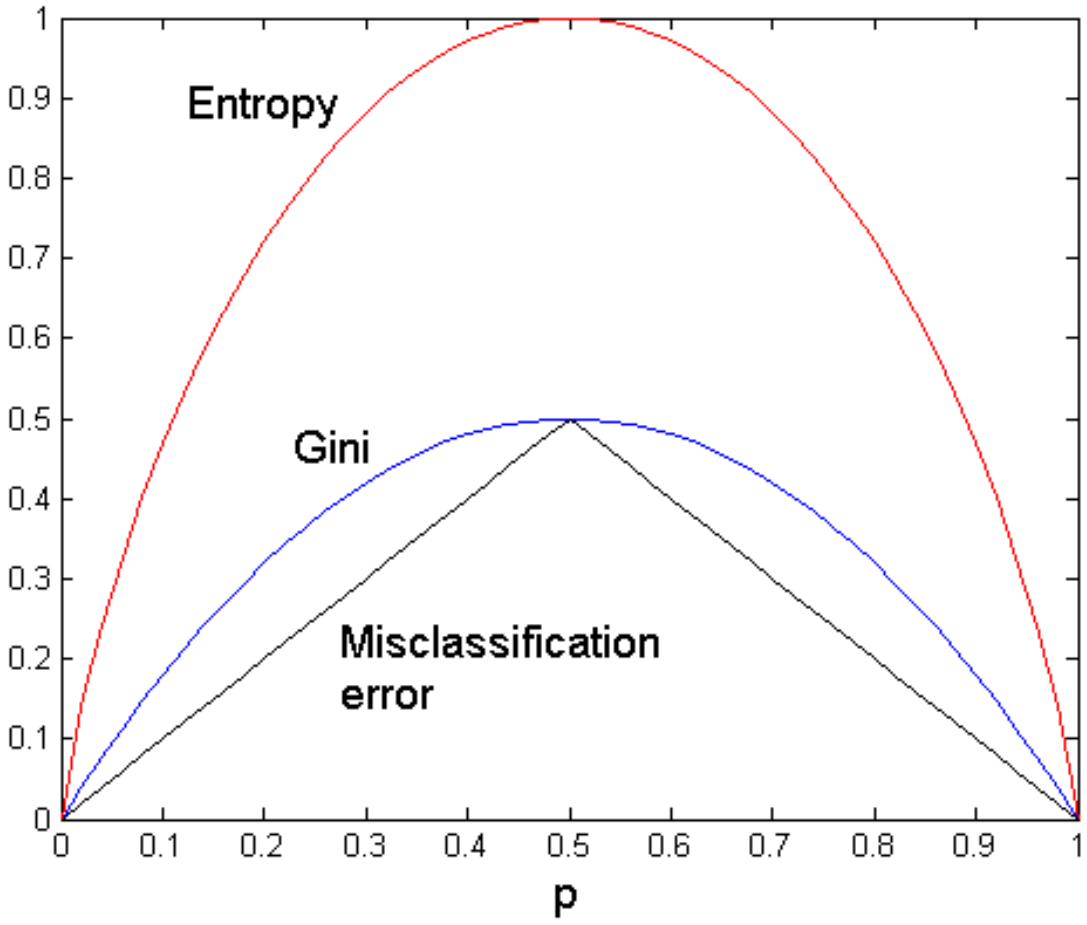
C1	**2**
C2	**4**

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

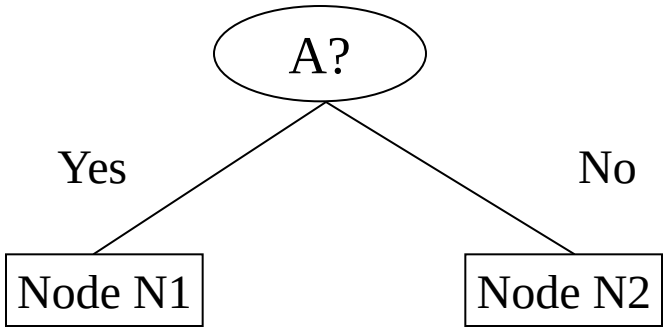
Error = 1 – max (2/6, 4/6) = 1 – 4/6 = 1/3

Σύγκριση μεταξύ των Κριτηρίων Διαχωρισμού

Για ένα πρόβλημα 2 κατηγοριών:



Σφάλμα Κατηγοριοποίησης και GINI



	Parent
C1	7
C2	3
Gini = 0.42	

	N1	N2
C1	3	4
C2	0	3
Gini=0.342		

Gini(N1)
= 1 – (3/3)² – (0/3)² = 0

Gini(N2)
= 1 – (4/7)² – (3/7)² = 0.489

Gini(Children)
= 3/10 * 0 + 7/10 * 0.489 = 0.342

Error(Parent) = 1 – max (7/10, 3/10) = 1 – 7/10 = 0.3
Error(N1) = 1 – max (3/3, 0/3) = 1 – 1 = 0
Error(N2) = 1 – max (4/7, 3/7) = 1 – 4/7 = 3/7
Error(Children) = 3/10 * 0 + 7/10 * 3/7 = 0.3

Η τιμή του Misclassification Error δεν αλλάζει με τον διαχωρισμό αλλά του Gini βελτιώνεται !!

Η Παραγωγή του Δέντρου

- Άπληστη στρατηγική:
  - Γίνεται διαχωρισμός των εγγραφών με βάση ένα τεστ στα χαρακτηριστικά, τα οποία βελτιστοποιεί ένα συγκεκριμένο κριτήριο.

- Θέματα που τίθενται:
  - Πώς θα χωρίσουν οι εγγραφές
    - Πώς θα οριστεί η συνθήκη στο τεστ χαρακτηριστικών;
    - Πώς θα οριστεί ο βέλτιστος διαχωρισμός;
  - Πότε θα σταματάει ο διαχωρισμός

Κριτήρια Τερματισμού της Παραγωγής Δέντρου

- Η επέκταση ενός κόμβου σταματά όταν σε αυτόν όλες οι εγγραφές ανήκουν σε μία κατηγορία
- Η επέκταση ενός κόμβου σταματά όταν όλες οι εγγραφές έχουν παρόμοιες τιμές στα χαρακτηριστικά τους
- Πρόωρος τερματισμός (θα συζητηθεί αργότερα)

Κατηγοριοποίηση με Δέντρα Απόφασης

- Πλεονεκτήματα:
  - Η κατασκευή τους γίνεται χωρίς μεγάλο κόστος
  - Είναι εξαιρετικά γρήγορα στο να κατηγοριοποιούν νέες/άγνωστες εγγραφές
  - Είναι εύκολο να ερμηνευτούν όταν έχουν μικρό μέγεθος ανάπτυξης
  - Η ακρίβειά τους είναι συγκρίσιμη με άλλες τεχνικές κατηγοριοποίησης για πολλά απλά σύνολα δεδομένων

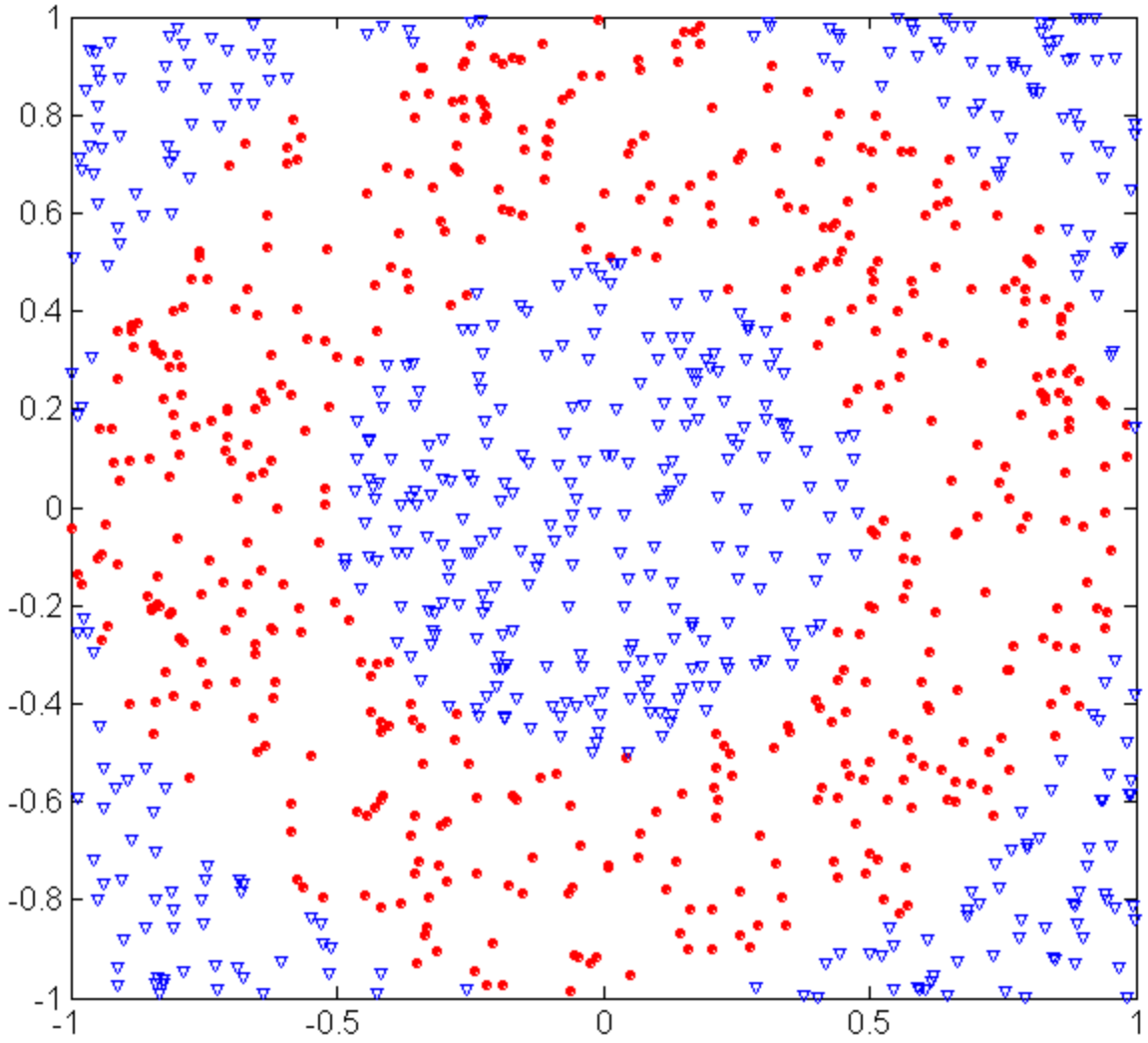
Παράδειγμα: Ο αλγόριθμος C4.5

- Κάνει απλή κατασκευή του δέντρου κατά βάθος (depth-first).
- Χρησιμοποιεί το Κέρδος Πληροφορίας (Information Gain).
- Ταξινομεί τα Συνεχή Χαρακτηριστικά σε κάθε κόμβο.
- Απαιτεί το σύνολο των δεδομένων να χωρούν στην κύρια μνήμη.
- Δεν είναι κατάλληλος για μεγάλα σύνολα δεδομένων
  - καθώς απαιτείται ταξινόμηση out-of-core.
- Μπορείτε να κάνετε download την υλοποίησή του: c4.5r8.tar.gz

Πρακτικά Θέματα της Κατηγοριοποίησης

- Underfitting και Overfitting στα δεδομένα
- Τιμές που λείπουν (Missing Values)
- Το Κόστος της Κατηγοριοποίησης

Underfitting και Overfitting (παράδειγμα)



Έστω 500 κυκλικά και 500 τριγωνικά σημεία ως δεδομένα. Για τις συντεταγμένες τους (x_1,x_2) ισχύει:

Για τα κυκλικά σημεία:

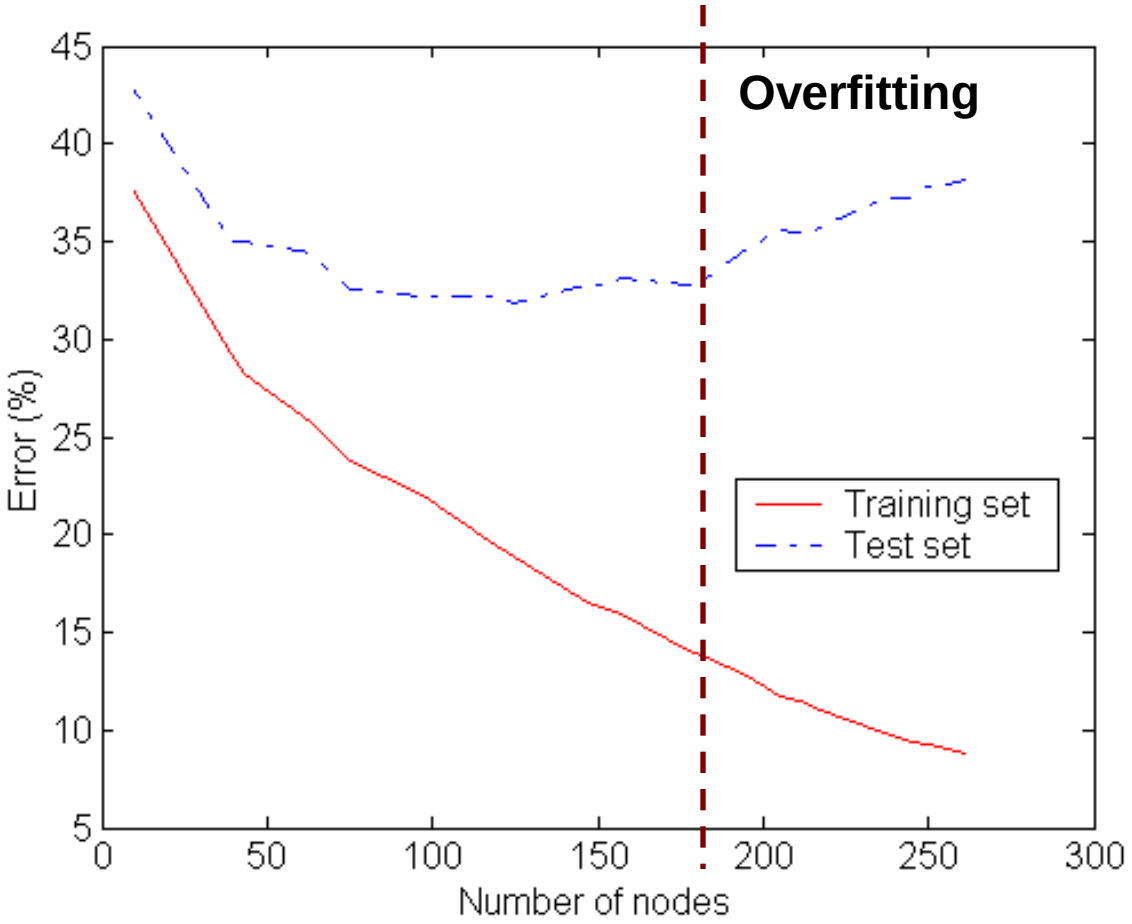
$0.5 \leq \text{sqrt}(x_1^2+x_2^2) \leq 1$

Για τα τριγωνικά:

$\text{sqrt}(x_1^2+x_2^2) > 0.5$ ή

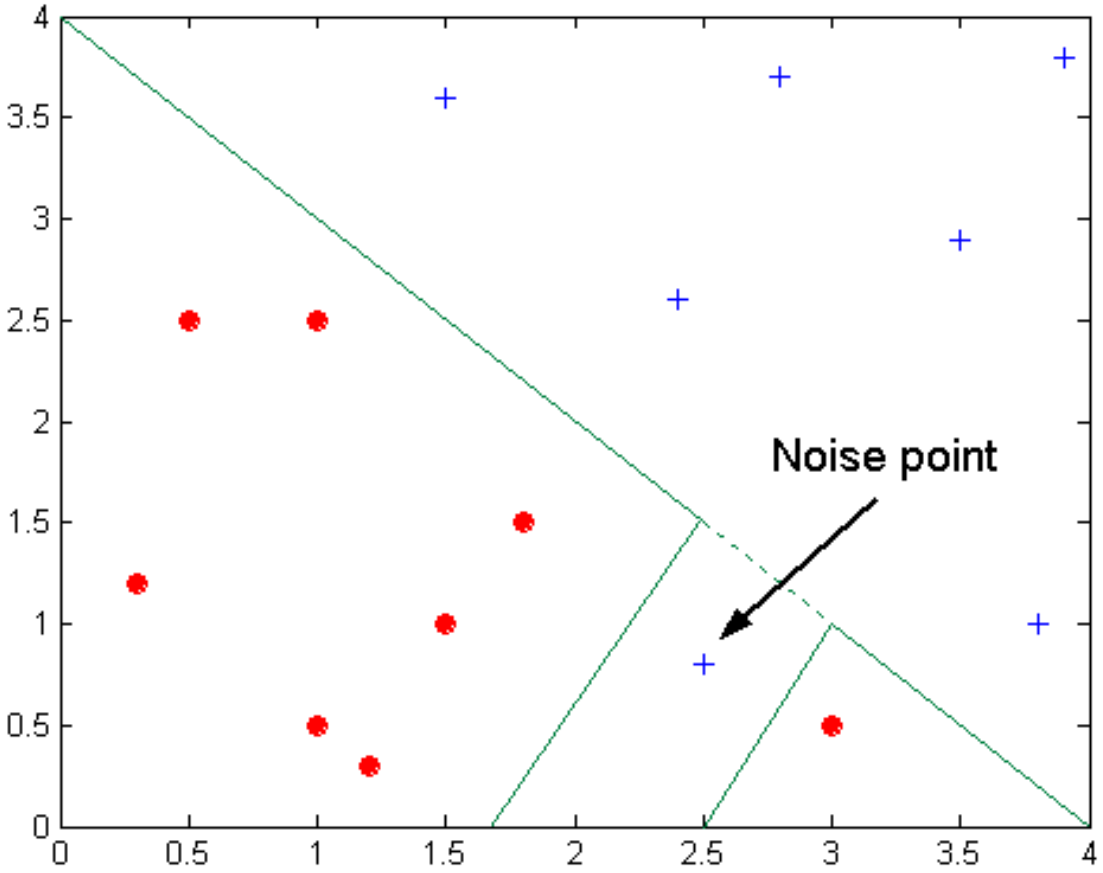
$\text{sqrt}(x_1^2+x_2^2) < 1$

Underfitting και Overfitting



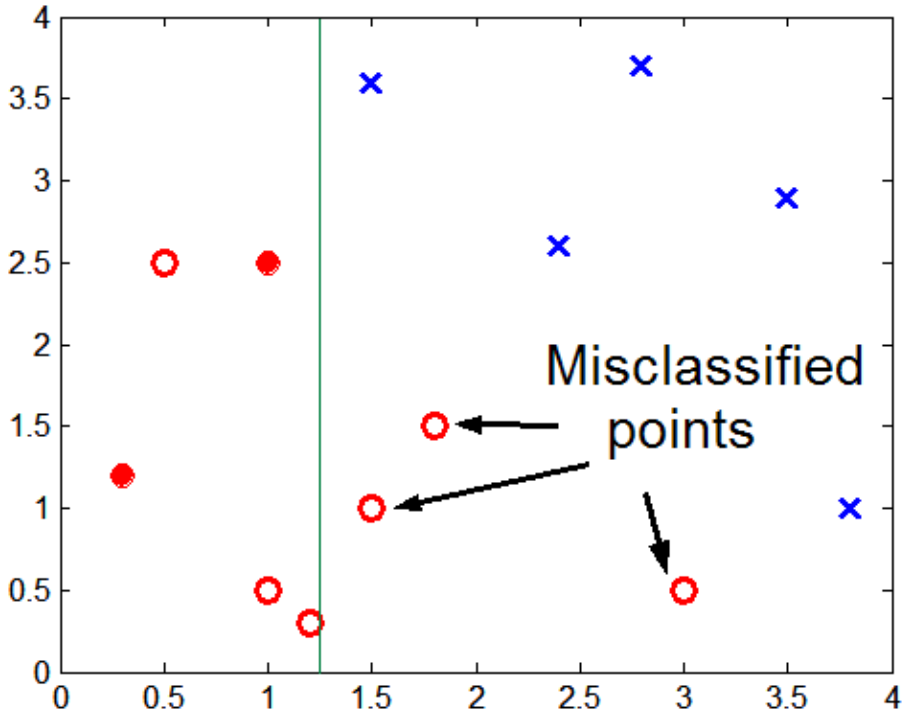
Underfitting: όταν το μοντέλο είναι πολύ απλό και το σφάλμα του training και το σφάλμα του test είναι μεγάλα.

Overfitting εξαιτίας «θορύβου»



Τα όρια των αποφάσεων παραμορφώνονται εξαιτίας της ύπαρξης ενός σημείου «θορύβου» στα δεδομένα

Overfitting λόγω ανεπαρκών δεδομένων



Η έλλειψη δεδομένων σημείων στο κάτω μισό του διαγράμματος καθιστά δύσκολο το να προβλεφθούν σωστά οι κατηγορίες αυτής της περιοχής

- Ο ανεπαρκής αριθμός από training εγγραφές στην περιοχή προκαλεί στο δέντρο απόφασης να προβλέψει τις test εγγραφές χρησιμοποιώντας άλλες training εγγραφές οι οποίες δεν σχετίζονται με την κατηγοριοποίηση

Παρατηρήσεις για το Overfitting

- Το overfitting έχει ως αποτέλεσμα να δημιουργούνται δέντρα απόφασης τα οποία είναι πιο περίπλοκα από όσο είναι απαραίτητο
- Το σφάλμα του training δεν αποτελεί πλέον μία καλή εκτίμηση για το πόσο αποτελεσματικό θα είναι το δέντρο σε ένα σύνολο νέων άγνωστων εγγραφών
- Απαιτούνται νέοι τρόποι για να εκτιμήσουμε τα σφάλματα

Εκτιμώντας τα σφάλματα της γενίκευσης

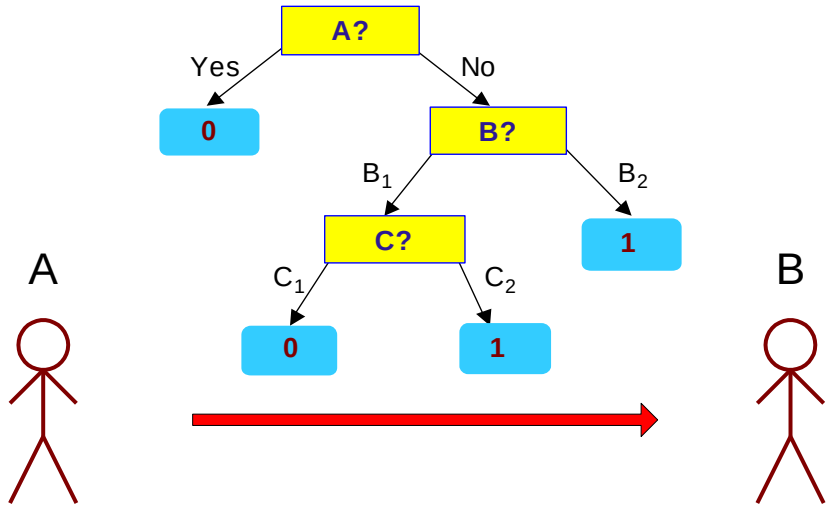
- Re-substitution errors: σφάλμα στο training (Σ e(t))
- Generalization errors: σφάλμα στο testing (Σ e'(t))
- Μέθοδοι εκτίμησης σφαλμάτων γενίκευσης:
  - Αισιόδοξη προσέγγιση: e'(t) = e(t)
  - Απαισιόδοξη προσέγγιση:
    - Για κάθε κόμβο-φύλλο: e'(t) = (e(t)+0.5)
    - Συνολικό σφάλμα: e'(T) = e(T) + N $\times$ 0.5 (όπου N το πλήθος των κόμβων-φύλλων στο δέντρο)
    - π.χ. για ένα δέντρο με 30 κόμβους-φύλλα και 10 σφάλματα στο training (από 1000 περιπτώσεις), έχουμε:
      Training error = 10/1000 = 1%
      Generalization error = (10 + 30$\times$0.5)/1000 = 2.5%
  - Reduced error pruning (REP):
    - χρησιμοποιεί δεδομένα επαλήθευσης για να εκτιμήσει το σφάλμα γενίκευσης.

Occam's Razor (κανόνας του Occam)

- Δεδομένου δύο μοντέλων με παρόμοιο σφάλμα γενίκευσης, θα πρέπει να προτιμάται το απλούστερο μοντέλο αντί του πιο σύνθετου
- Σε πολύπλοκα μοντέλα, υπάρχει μεγαλύτερη πιθανότητα μία εγγραφή να τοποθετηθεί τυχαία λόγω σφαλμάτων στα δεδομένα
- Συνεπώς θα πρέπει όταν αξιολογείται ένα μοντέλο να συμπεριλάβουμε και το πόσο πολύπλοκο είναι

Ελάχιστο Μήκος Περιγραφής
Minimum Description Length (MDL)

X	y
X_1	1
X_2	0
X_3	0
X_4	1
…	…
X_n	1



X	y
X_1	?
X_2	?
X_3	?
X_4	?
…	…
X_n	?

- Cost(Model,Data) = Cost(Data|Model) + Cost(Model)
  - Το κόστος (Cost) είναι το πλήθος των bits που απαιτούνται για την κωδικοποίηση.
  - Αναζητείται το μοντέλο με το ελάχιστο κόστος.
- Το Cost(Data|Model) κωδικοποιεί τα σφάλματα κατηγοριοποίησης.
- Το Cost(Model) χρησιμοποιεί κωδικοποίηση κόμβων (πλήθος παιδιών) συν κωδικοποίηση για τις συνθήκες διαχωρισμού.

Πώς να διαχειριστείτε το Overfitting

- Πρόωρο Κλάδεμα Pre-Pruning (Κανόνες Πρόωρης Διακοπής)
  - Ο αλγόριθμος σταματά πριν να αναπτυχθεί πλήρως ένα δέντρο απόφασης
  - Τυπικές συνθήκες τερματισμού σε έναν κόμβο:
    - όταν όλες οι περιπτώσεις-εγγραφές ανήκουν στην ίδια κατηγορία
    - όταν όλες οι τιμές των χαρακτηριστικών στις εγγραφές είναι ίδιες
  - Πιο περιοριστικές συνθήκες τερματισμού:
    - Όταν το πλήθος των περιπτώσεων-εγγραφών είναι μικρότερο από ένα όριο που καθορίζεται από τον χρήστη
    - Όταν η κατανομή των περιπτώσεων-εγγραφών στις κατηγορίες είναι ανεξάρτητη από τα διαθέσιμα χαρακτηριστικά (ο έλεγχος γίνεται π.χ. χρησιμοποιώντας το χ^2 test)
    - Όταν η επέκταση του τρέχοντος κόμβου δεν βελτιώνει μέτρα ανάμειξης εγγραφών (π.χ. Gini ή information gain).

Πώς να διαχειριστείτε το Overfitting

- Κλάδεμα μετά την ανάπτυξη (Post-pruning)
  - Αρχικά αναπτύσσεται πλήρως το δέντρο απόφασης (στο σύνολό του)
  - Γίνεται αποκοπή των κόμβων του δέντρου από κάτω προς τα πάνω
  - Όταν το σφάλμα γενίκευσης βελτιώνεται μετά από μία αποκοπή υπόδεντρου, αντικαθίσταται το υπόδεντρο με έναν κόμβο-φύλλο
  - Η κατηγορία του κόμβου-φύλλο καθορίζεται από την κατηγορία της πλειοψηφίας των περιπτώσεων-εγγραφών στο υπόδεντρο
  - Μπορεί να χρησιμοποιηθεί MDL για post-pruning

Παράδειγμα του Post-Pruning

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

Training Error (Before splitting) = 10/30

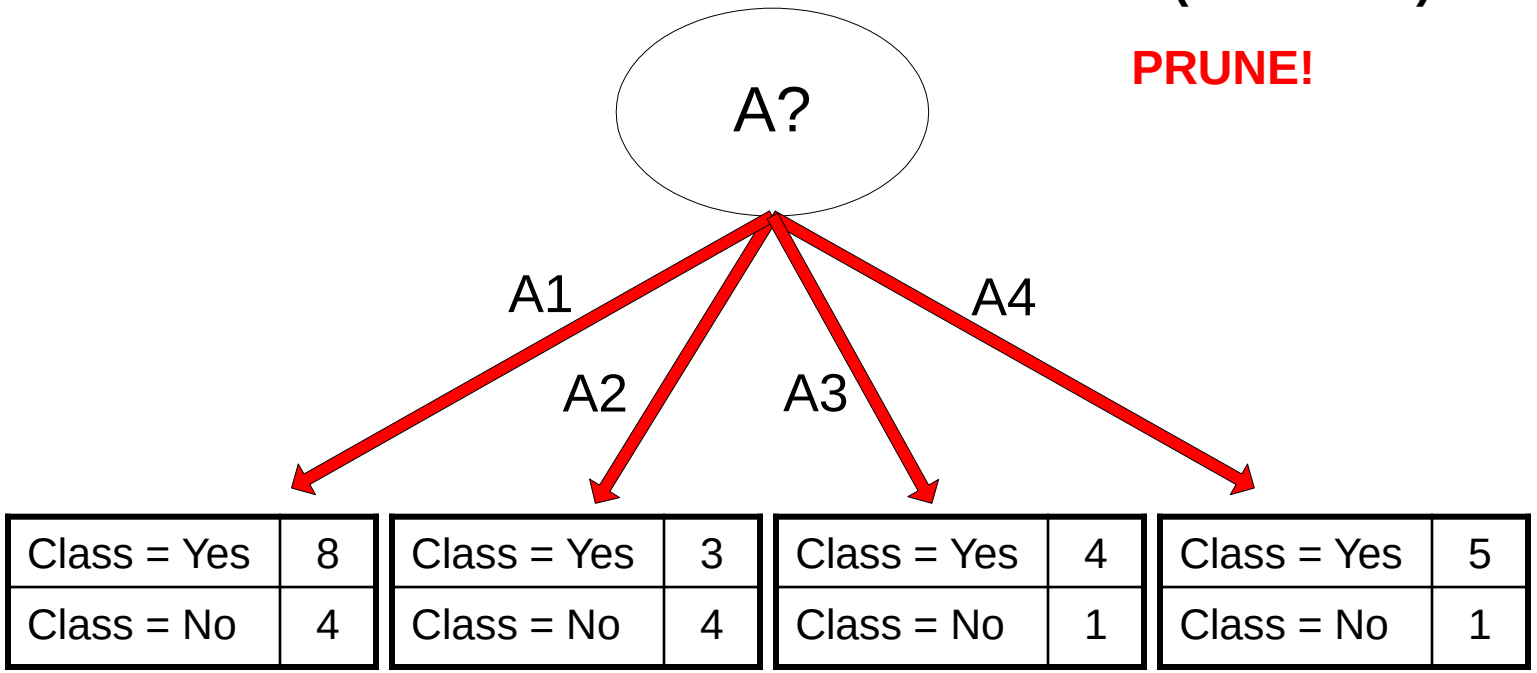
Pessimistic error = (10 + 0.5)/30 = 10.5/30

Training Error (After splitting) = 9/30

Pessimistic error (After splitting)

= (9 + 4 × 0.5)/30 = 11/30

PRUNE!



Class = Yes	8
Class = No	4

Class = Yes	3
Class = No	4

Class = Yes	4
Class = No	1

Class = Yes	5
Class = No	1

Παραδείγματα Post-pruning

- Optimistic error?

  Δεν γίνεται κλάδεμα σε καμία από τις δύο περιπτώσεις

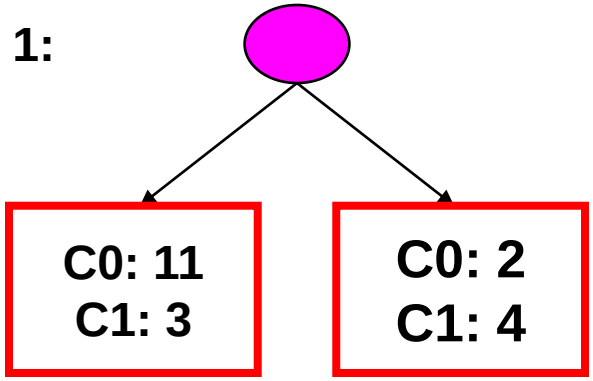
- Pessimistic error?

  Δεν γίνεται κλάδεμα στην 1η περίπτωση, αλλά γίνεται στην 2η

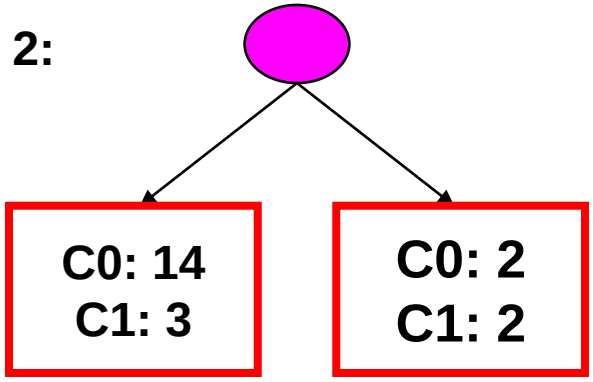
- Reduced error pruning?

  Εξαρτάται από το σύνολο εγγραφών επαλήθευσης (validation set)

Case 1:



Case 2:



Διαχείριση τιμών χαρακτηριστικών που λείπουν

- Τιμές που λείπουν επηρεάζουν την κατασκευή του δέντρου απόφασης με τρεις διαφορετικούς τρόπους:
  - Επηρεάζουν το πώς υπολογίζονται τα μέτρα ανάμειξης των εγγραφών
  - Επηρεάζουν το πώς κατανέμονται οι περιπτώσεις εγγραφών που έχουν ελλιπείς τιμές στους κόμβους-παιδιά
  - Επηρεάζουν το πώς μία εγγραφή test με ελλιπείς τιμές θα κατηγοριοποιηθεί

Υπολογίζοντας τα μέτρα ανάμειξης

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

Πριν από τον διαχωρισμό:

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**

Διαχωρισμός στο χαρ/κό Refund:

Entropy(Refund=Yes) = 0

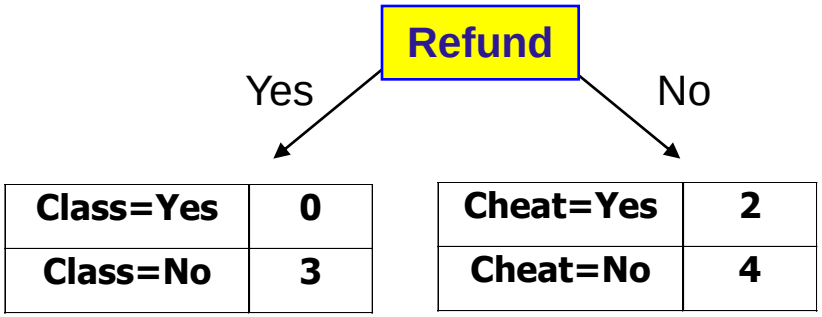
Entropy(Refund=No)
= -(2/6)log(2/6) – (4/6)log(4/6) = 0.9183

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

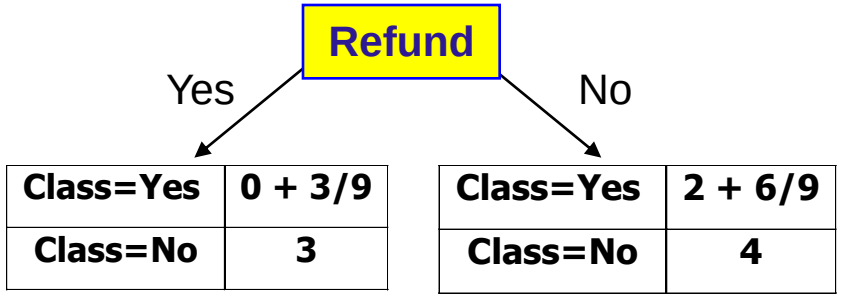
Gain = 0.9 × (0.8813 – 0.551) = 0.3303

Κατανέμοντας τις περιπτώσεις εγγραφών

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



Η πιθανότητα να είναι Refund=Yes είναι 3/9

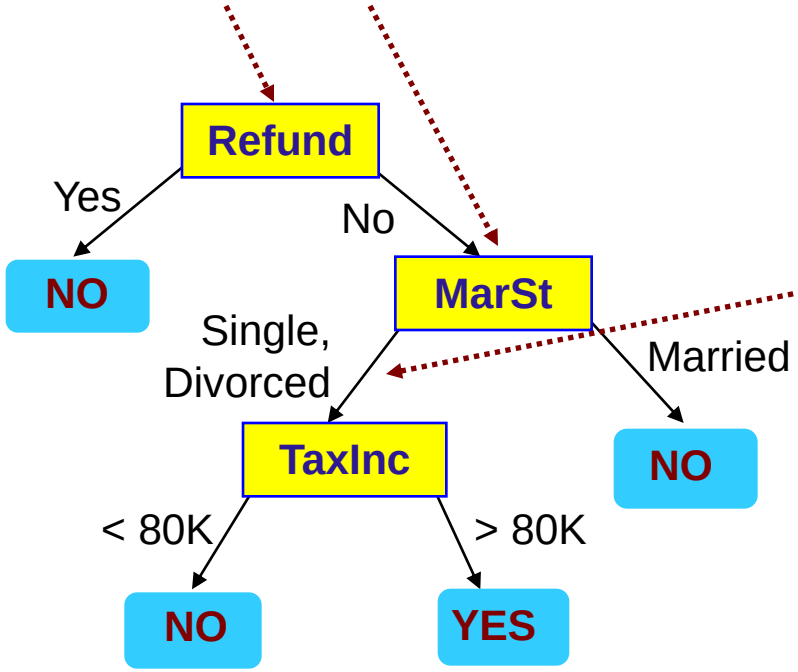
Η πιθανότητα να είναι Refund=No είναι 6/9

Η εγγραφή τοποθετείται στο αριστερό παιδί με βάρος = 3/9 και στο δεξί με βάρος = 6/9

Κατηγοριοποιώντας νέες περιπτώσεις

Έστω η 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

Η πιθανότητα να είναι Marital Status = Married είναι 3.67/6.67

Η πιθανότητα να είναι Marital Status ={Single, Divorced} είναι 3/6.67

Άλλα Θέματα και Ιδιότητες των Δέντρων

- Κατακερματισμός Δεδομένων (Data Fragmentation)
- Στρατηγική Αναζήτησης (Search Strategy)
- Εκφραστικότητα (Expressiveness)
- Αναπαραγωγή (Tree Replication)

Κατακερματισμός Δεδομένων

- Ο αριθμός των περιπτώσεων-εγγραφών γίνεται μικρότερος καθώς διασχίζουμε προς τα κάτω το δέντρο
- Ο αριθμός των περιπτώσεων-εγγραφών στους κόμβους-φύλλα μπορεί να είναι πολύ μικρός για να πάρουμε μία στατιστικά σημαντική απόφαση

Στρατηγική Αναζήτησης

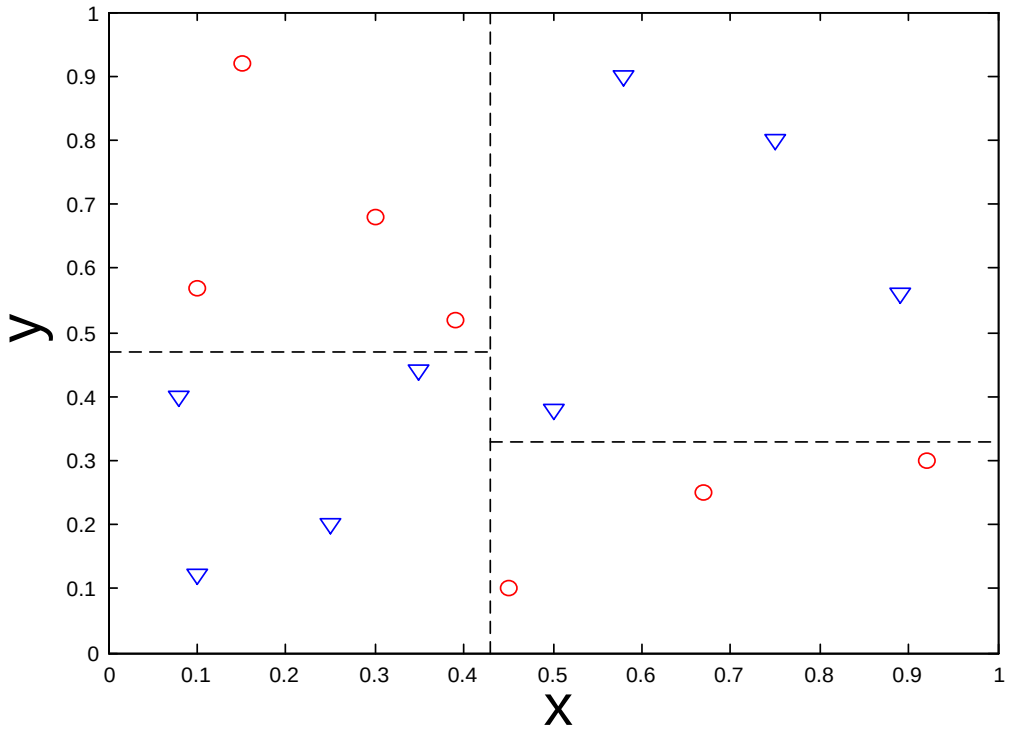
- Το να βρεθεί το βέλτιστο δέντρο απόφασης είναι πρόβλημα NP-hard
- Ο αλγόριθμος που παρουσιάστηκε χρησιμοποιεί μία άπληστη στρατηγική, ξεκινώντας από πάνω προς τα κάτω, με αναδρομικό διαχωρισμό, ώστε να παράγει μία λογική λύση σε λογικό χρόνο

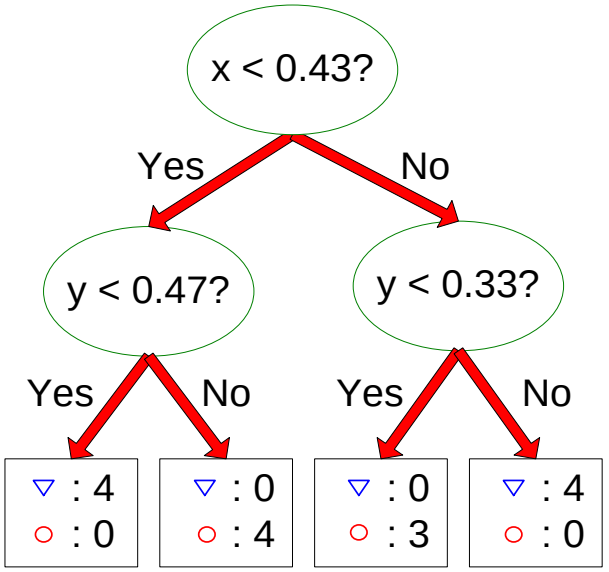
- Ποιες άλλες στρατηγικές υπάρχουν;
  - Bottom-up (από κάτω προς τα πάνω)
  - Bi-directional (αμφίδρομης κατεύθυνσης)

Εκφραστικότητα

- Το δέντρο απόφασης παρέχει εκφραστική αναπαράσταση για την εκμάθηση συναρτήσεων διακριτών τιμών
  - Αλλά δεν κάνει καλή γενίκευση σε ορισμένους τύπους Boolean συναρτήσεων
    - Για παράδειγμα: parity function:
      - Class = 1 αν υπάρχει άρτιος αριθμός Boolean χαρακτηριστικών με τιμή αληθείας = True
      - Class = 0 αν υπάρχει περιττός αριθμός Boolean χαρακτηριστικών με τιμή αληθείας = True
    - Για να γίνει ακριβής μοντελοποίηση στο παράδειγμα αυτό, πρέπει να αναπτυχθεί ένα πλήρες δέντρο
- Δεν είναι αρκετά εκφραστικό για την μοντελοποίηση συνεχών μεταβλητών
  - Ιδιαίτερα όταν μία συνθήκη ελέγχου περιλαμβάνει μόνο ένα απλό χαρακτηριστικό τη φορά

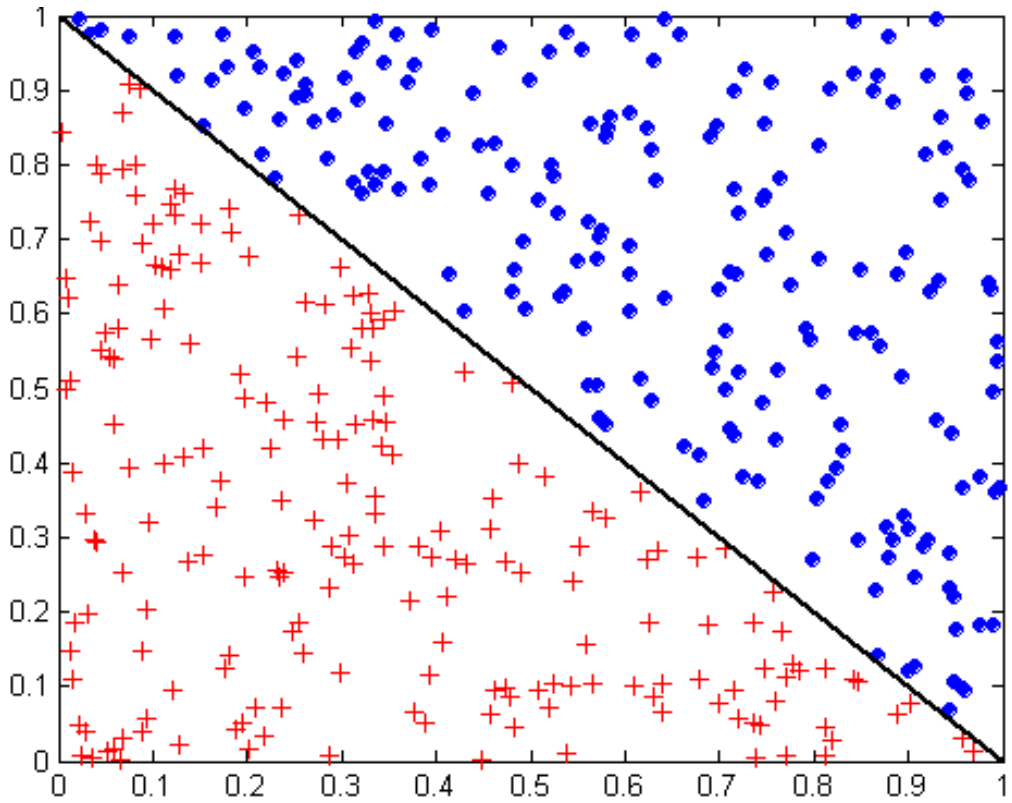
Όρια Αποφάσεων

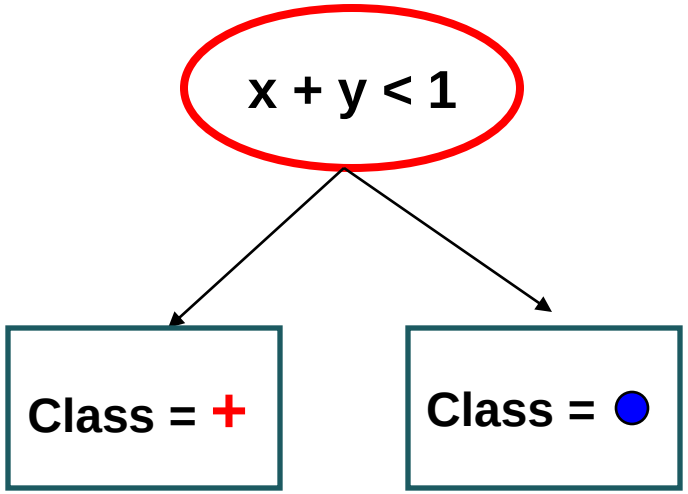




- **Οι συνοριακές γραμμές μεταξύ δύο γειτονικών περιοχών από διαφορετικές κατηγορίες είναι γνωστές ως όρια αποφάσεων**
- **Τα όρια αποφάσεων είναι παράλληλα στους άξονες όταν η συνθήκη ελέγχου αφορά μόνο ένα απλό χαρακτηριστικό τη φορά**

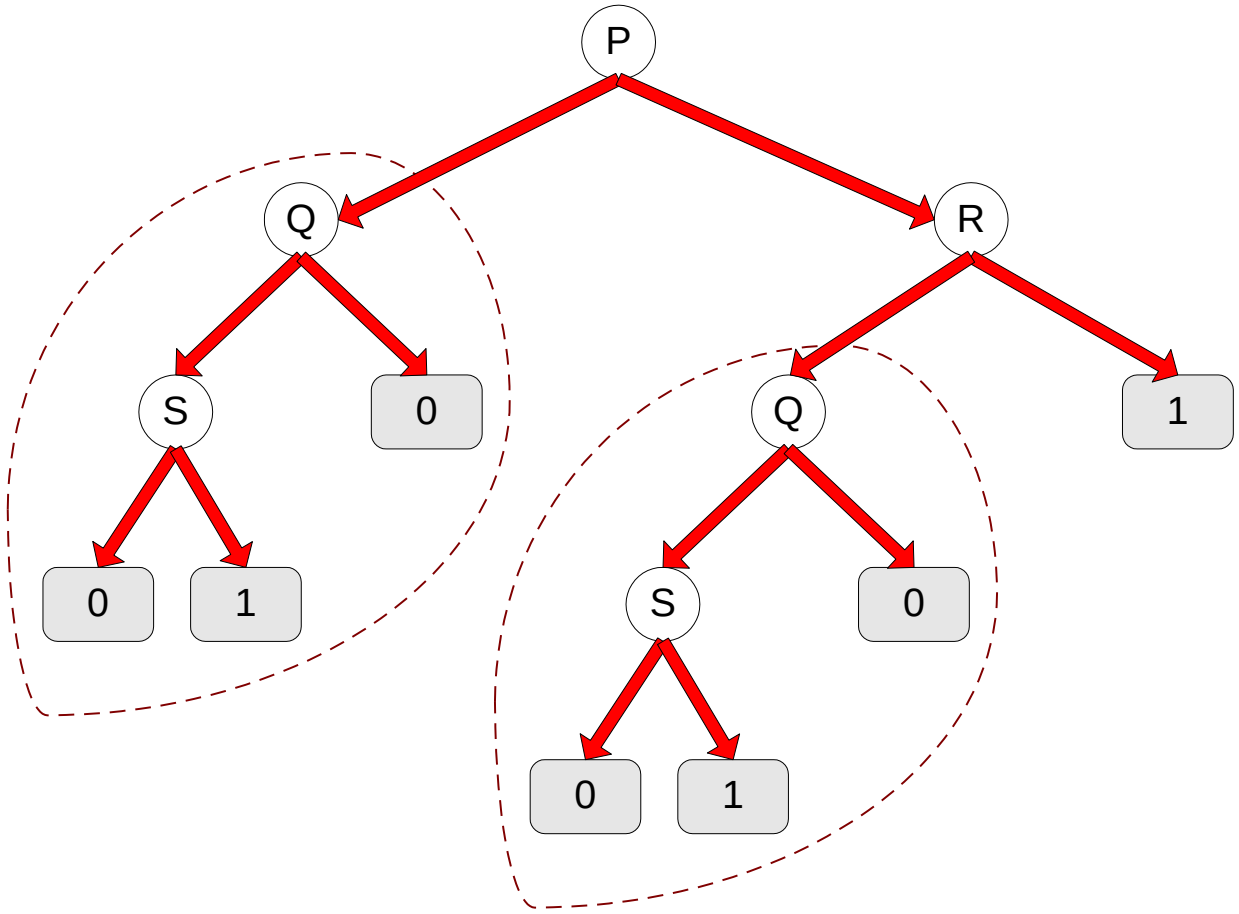
Πλάγια όρια σε Δέντρα Απόφασης





- Η συνθήκη ελέγχου μπορεί να αφορά πολλά χαρακτηριστικά
- Η αναπαράσταση γίνεται πιο εκφραστική
- Το να βρεθεί η βέλτιστη συνθήκη ελέγχου είναι υπολογιστικά ακριβό

Αναπαραγωγή



- **Το ίδιο υπόδεντρο εμφανίζεται σε πολλούς κλάδους**

Αξιολόγηση ενός Μοντέλου

- Μετρικές για την αξιολόγηση της απόδοσης
  - Πώς να αξιολογήσετε την αποδοτικότητα ενός μοντέλου;
- Μέθοδοι για την αξιολόγηση της απόδοσης
  - Πώς να προκύψουν αξιόπιστες εκτιμήσεις;
- Μέθοδοι για σύγκριση μοντέλων
  - Πως θα συγκρίνετε την σχετική αποδοτικότητα μεταξύ ανταγωνιστικών μοντέλων;

Αξιολόγηση ενός Μοντέλου

- Μετρικές για την αξιολόγηση της απόδοσης
  - Πώς να αξιολογήσετε την αποδοτικότητα ενός μοντέλου;
- Μέθοδοι για την αξιολόγηση της απόδοσης
  - Πώς να προκύψουν αξιόπιστες εκτιμήσεις;
- Μέθοδοι για σύγκριση μοντέλων
  - Πως θα συγκρίνετε την σχετική αποδοτικότητα μεταξύ ανταγωνιστικών μοντέλων;

Μετρικές για την αξιολόγηση της απόδοσης

- Επικεντρωνόμαστε στην προγνωστική ικανότητα ενός μοντέλου
  - αντί στο πόσο γρήγορα κάνει την κατηγοριοποίηση, ή κτίζεται το μοντέλο, η άλλα μέτρα όπως scalability κλπ.
- Χρησιμοποιείται ο Confusion Matrix:

Count	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

Μετρικές για την αξιολόγηση της απόδοσης

Count	PREDICTED CLASS		
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

- Είναι η ποιο συχνά χρησιμοποιούμενη τεχνική:

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

Περιορισμοί στην Ακρίβεια

- Έστω ένα πρόβλημα με 2 κατηγορίες:
  - Πλήθος εγγραφών στην Κατηγορία-0 = 9990
  - Πλήθος εγγραφών στην Κατηγορία-1 = 10
- Αν όλες οι προβλέψεις του μοντέλου είναι στην Κατηγορία-0, τότε η ακρίβειά του είναι: 9990/10000 = 99.9 %!
  - Η ακρίβεια είναι παραπλανητική επειδή το μοντέλο δεν εντοπίζει κανένα παράδειγμα για την Κατηγορία-1

Χρήση του Cost Matrix

Cost	PREDICTED CLASS		
ACTUAL CLASS	C(i|j)	**Class=Yes**	**Class=No**
	Class=Yes	C(Yes|Yes)	C(No|Yes)
	Class=No	C(Yes|No)	C(No|No)

C(i|j): Κόστος της εσφαλμένης κατηγοριοποίησης μιας εγγραφής κατηγορίας j στην κατηγορία i

Υπολογίζοντας το Κόστος της Κατηγοριοποίησης

Cost Matrix	PREDICTED CLASS		
	C(i|j)	+	-
ACTUAL CLASS	+	-1	100
	-	1	0

Model M_1	PREDICTED CLASS		
		+	-
ACTUAL CLASS	+	150	40
	-	60	250

Accuracy = 80%

Cost = 3910

Model M_2	PREDICTED CLASS		
		+	-
ACTUAL CLASS	+	250	45
	-	5	200

Accuracy = 90%

Cost = 4255

Σύγκριση Κόστους και Ακρίβειας

Count	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

Cost	PREDICTED CLASS		
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	p	q
	Class=No	q	p

Η ακρίβεια είναι ανάλογη του κόστους όταν:

1. C(Yes|No)=C(No|Yes) = q
2. C(Yes|Yes)=C(No|No) = p

$N = a + b + c + d$

$Accuracy = (a + d)/N$

$Cost = p\,(a + d) + q\,(b + c)$

$= p\,(a + d) + q\,(N - a - d)$

$= q\,N - (q - p)(a + d)$

$= N\,[q - (q-p) \times Accuracy]$

Μετρικές «ευαίσθητες» στο Κόστος

$$\text{Precision (p)} = \frac{a}{a+c}$$

$$\text{Recall (r)} = \frac{a}{a+b}$$

$$\text{F - measure (F)} = \frac{2rp}{r+p} = \frac{2a}{2a+b+c}$$

- Precision «μεροληπτεί» υπέρ των C(Yes|Yes) & C(Yes|No)
- Recall μεροληπτεί υπέρ των C(Yes|Yes) & C(No|Yes)
- F-measure μεροληπτεί σε όλα εκτός από το C(No|No)

$$\text{Weighted Accuracy} = \frac{w_1 a + w_4 d}{w_1 a + w_2 b + w_3 c + w_4 d}$$

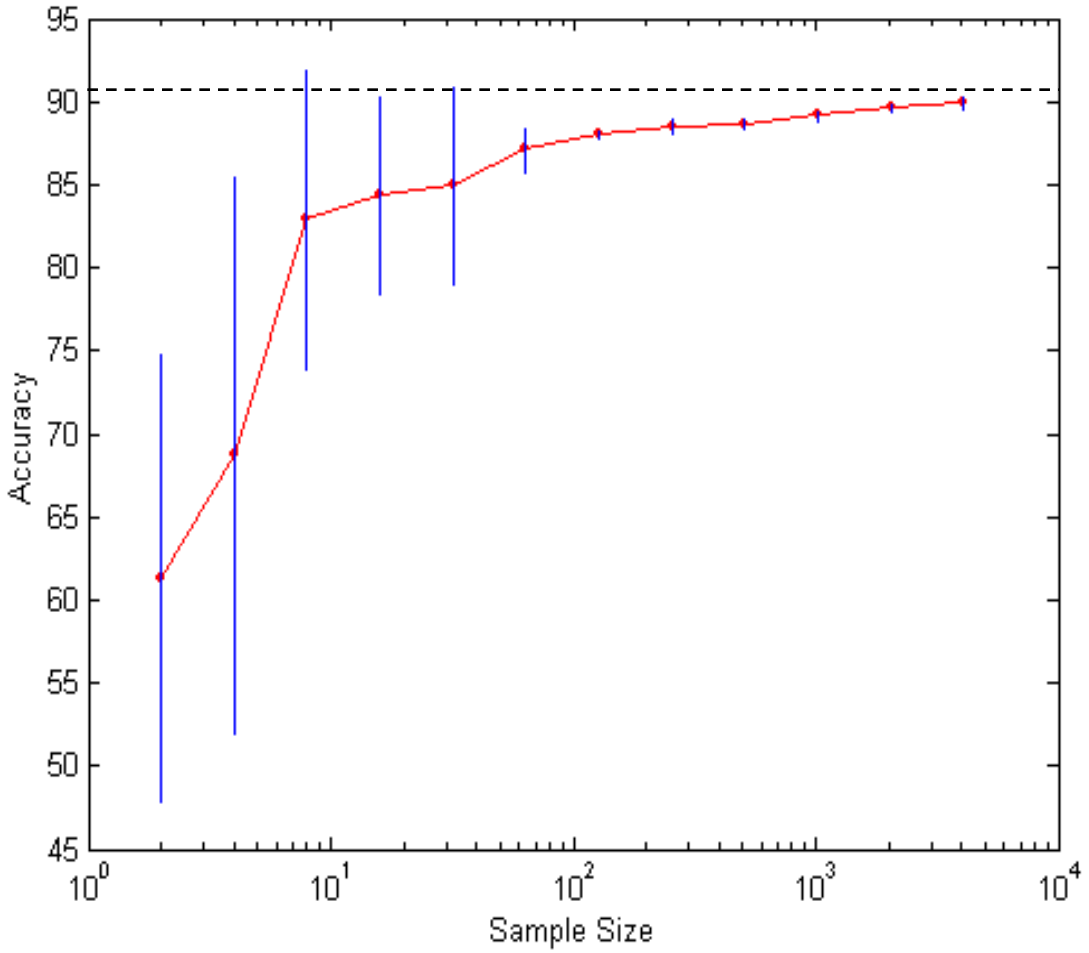
Αξιολόγηση ενός Μοντέλου

- Μετρικές για την αξιολόγηση της απόδοσης
  - Πώς να αξιολογήσετε την αποδοτικότητα ενός μοντέλου;
- Μέθοδοι για την αξιολόγηση της απόδοσης
  - Πώς να προκύψουν αξιόπιστες εκτιμήσεις;
- Μέθοδοι για σύγκριση μοντέλων
  - Πως θα συγκρίνετε την σχετική αποδοτικότητα μεταξύ ανταγωνιστικών μοντέλων;

Μέθοδοι για την αξιολόγηση της απόδοσης

- Πώς να αποκτήσουμε μία αξιόπιστη εκτίμηση της απόδοσης;
- Η απόδοση ενός μοντέλου μπορεί να εξαρτάται από άλλους παράγοντες εκτός από τον αλγόριθμο εκμάθησης:
  - Από την κατανομή των κατηγοριών
  - Από το κόστος της εσφαλμένης κατηγοριοποίησης
  - Από το μέγεθος των συνόλων training και test

Καμπύλη Εκμάθησης



- Η καμπύλη εκμάθησης δείχνει πώς αλλάζει η ακρίβεια καθώς μεταβάλλεται το μέγεθος του δείγματος.
- Απαιτείται μία μέθοδος δειγματοληψίας για να δημιουργηθεί η καμπύλη εκμάθησης:
  - Αριθμητική δειγματοληψία (Langley, et al)
  - Γεωμετρική δειγματοληψία (Provost et al)

Επίδραση του μικρού δείγματος:

- Αλλοίωση της εκτίμησης
- Διακύμανση της εκτίμησης

Μέθοδοι Εκτίμησης

- Holdout
  - Δεσμεύονται τα 2/3 των εγγραφών για training και το 1/3 για testing
- Random subsampling (τυχαία υποδειγματοληψία)
  - Επαναλαμβάνεται η μέθοδος holdout
- Cross validation
  - Τα δεδομένα χωρίζονται σε k ξένα μεταξύ τους υποσύνολα
  - k-fold: γίνεται train στα k-1 υποσύνολα, και test στο υποσύνολο που έχει απομείνει
  - Leave-one-out: k=n
- Stratified sampling (δειγματοληψία σε στρώματα)
  - oversampling vs undersampling
- Bootstrap
  - Δειγματοληψία με αντικατάσταση

Αξιολόγηση ενός Μοντέλου

- Μετρικές για την αξιολόγηση της απόδοσης
  - Πώς να αξιολογήσετε την αποδοτικότητα ενός μοντέλου;
- Μέθοδοι για την αξιολόγηση της απόδοσης
  - Πώς να προκύψουν αξιόπιστες εκτιμήσεις;
- Μέθοδοι για σύγκριση μοντέλων
  - Πως θα συγκρίνετε την σχετική αποδοτικότητα μεταξύ ανταγωνιστικών μοντέλων;

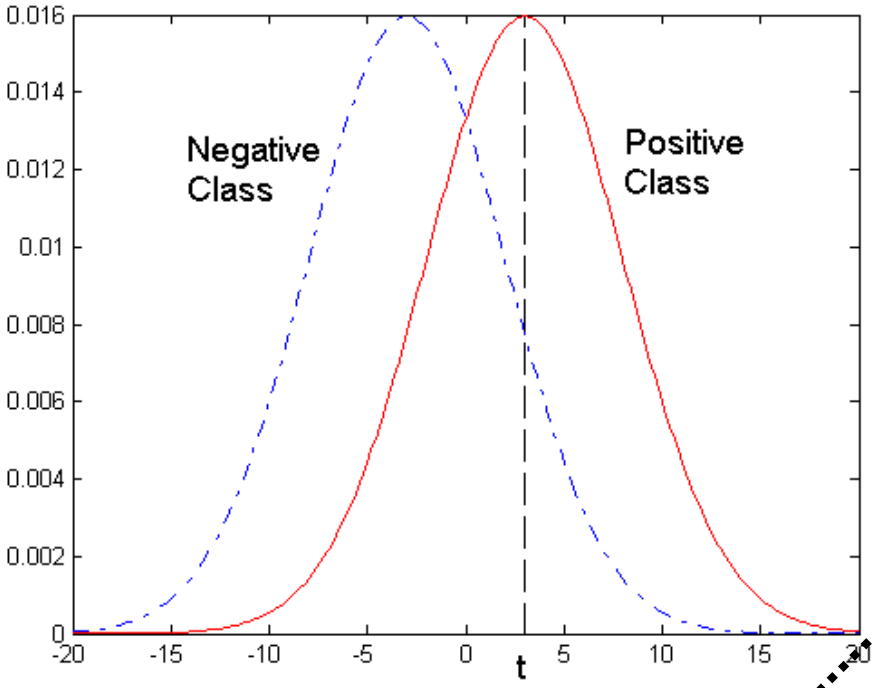
ROC (Receiver Operating Characteristic)

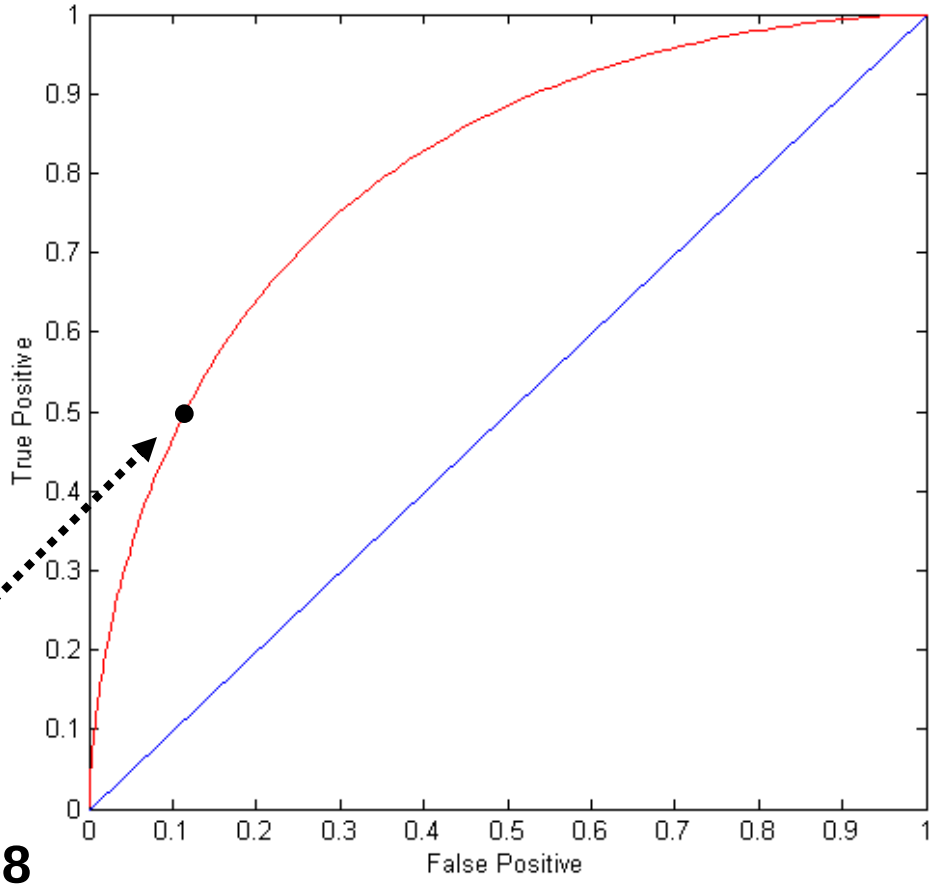
- Αναπτύχθηκε τη δεκαετία του 1950s στη θεωρία ανίχνευσης σημάτων με σκοπό την ανάλυση σημάτων που περιέχουν θόρυβο
  - Χαρακτηρίζει το trade-off μεταξύ των positive hits και των false alarms
- Η καμπύλη ROC απεικονίζει τα TP (στον άξονα y) σε σχέση με τα FP (στον άξονα x)
- Η απόδοση του κάθε μοντέλου κατηγοριοποίησης παριστάνεται ως ένα σημείο πάνω στην καμπύλη ROC
  - Αλλάζοντας την οριακή τιμή (threshold) του αλγορίθμου, την κατανομή δειγματοληψίας ή τον cost matrix, αλλάζει και η θέση του σημείου αυτού

Καμπύλη ROC

- Έστω ένα μονοδιάστατο σύνολο δεδομένων που περιλαμβάνει 2 κατηγορίες (positive και negative)

- Κάθε σημείο που βρίσκεται στη θέση x > t κατηγοριοποιείται ως positive





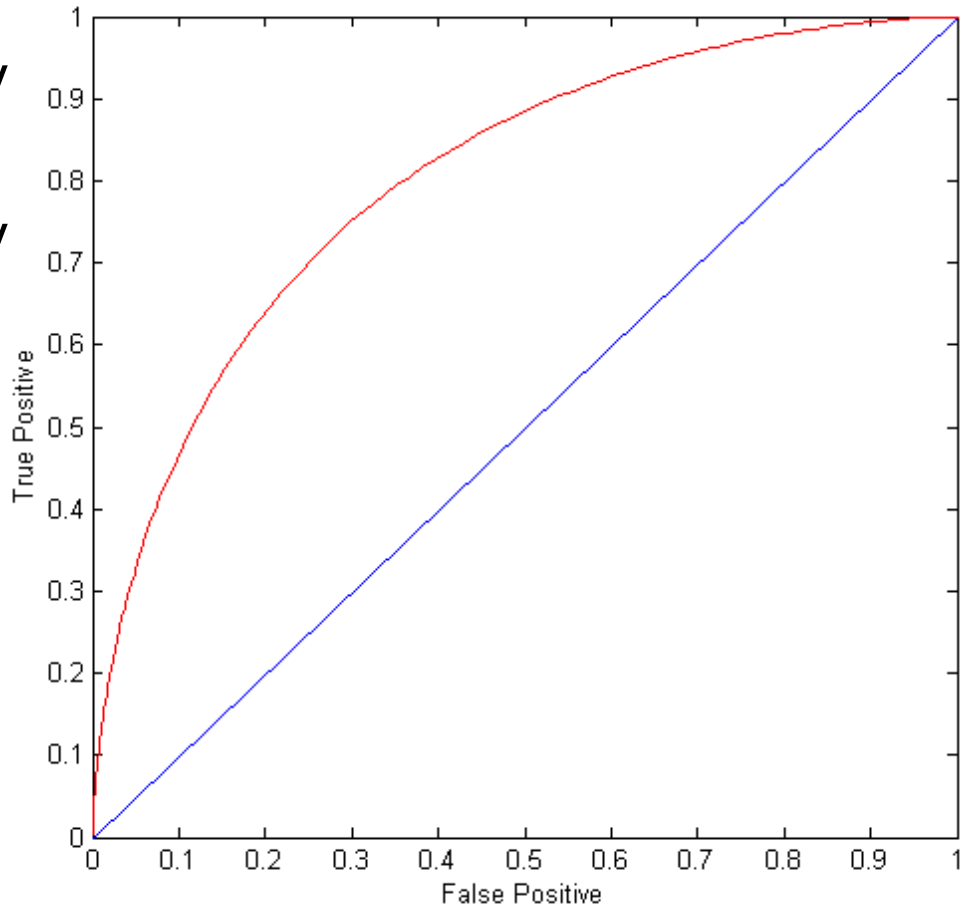
At threshold t:

TP=0.5, FN=0.5, FP=0.12, FN=0.88

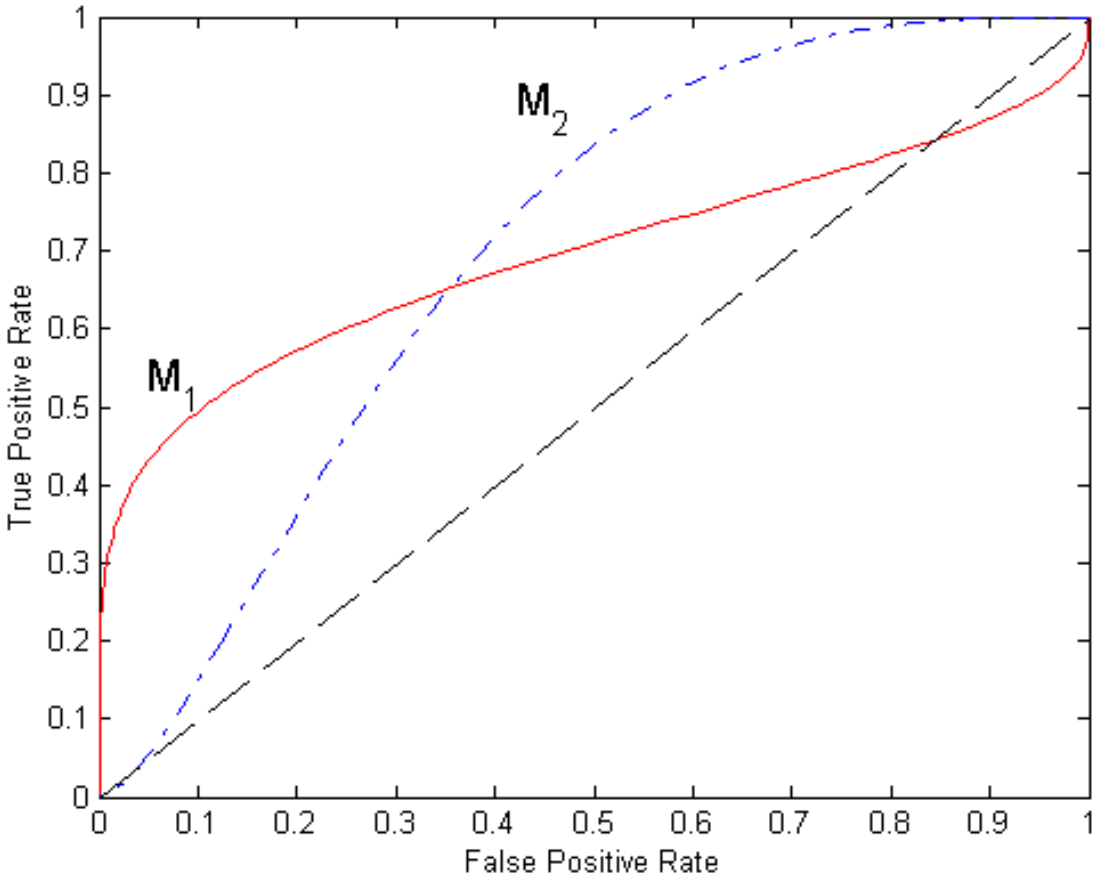
Καμπύλη ROC

(TP,FP):

- (0,0): δηλώνει ότι όλα ανήκουν στην κατηγορία negative
- (1,1): δηλώνει ότι όλα ανήκουν στην κατηγορία positive
- (1,0): ιδανικό σημείο
- Διαγώνιος γραμμή:
  - Δείχνει τυχαία πρόβλεψη
  - Κάτω από την διαγώνιο γραμμή:
    - η πρόβλεψη είναι αντίθετη από την πραγματική κατηγορία



Χρήση της ROC για σύγκριση μοντέλων



- Κανένα από τα δύο μοντέλα δεν ξεπερνά σταθερά το άλλο
  - Το M_1 είναι καλύτερο για μικρά FPR
  - Το M_2 είναι καλύτερο για μεγάλα FPR
- Εμβαδόν περιοχής κάτω από την καμπύλη ROC
  - Ιδανική τιμή:
    - Area = 1
  - Τυχαία πρόβλεψη:
    - Area = 0.5

Πώς κατασκευάζεται μία καμπύλη ROC

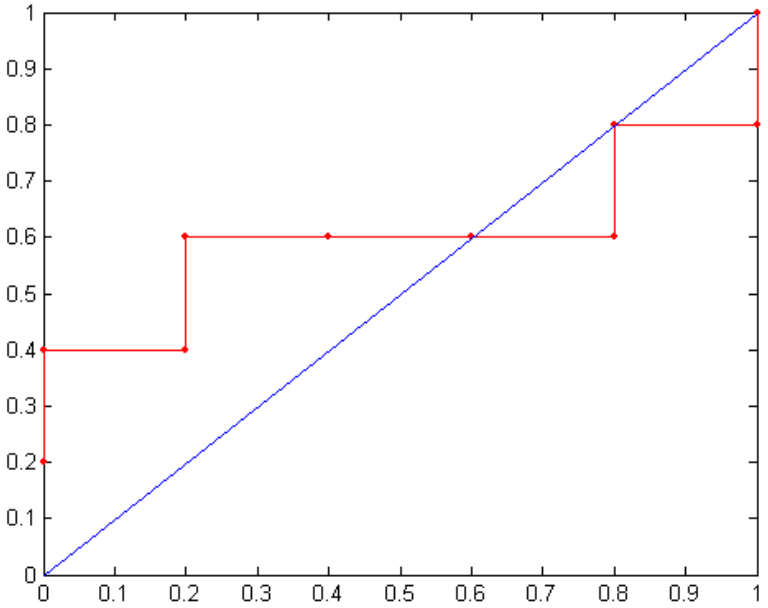
Instance	P(+|A)	True Class
1	0.95	+
2	0.93	+
3	0.87	-
4	0.85	-
5	0.85	-
6	0.85	+
7	0.76	-
8	0.53	+
9	0.43	-
10	0.25	+

- Χρησιμοποιείται ένας κατηγοριοποιητής που παράγει πιθανότητα posterior για κάθε στιγμιότυπο test P(+|A)
- Ταξινομούνται τα στιγμιότυπα σύμφωνα με τις τιμές τους P(+|A) σε φθίνουσα σειρά
- Το όριο (threshold) εφαρμόζεται σε κάθε μοναδική τιμή P(+|A)
- Μετριέται το πλήθος των TP, FP, TN, FN σε κάθε όριο
- TP rate, TPR = TP/(TP+FN)
- FP rate, FPR = FP/(FP + TN)

Πώς κατασκευάζεται μία καμπύλη ROC

	Class	+	-	+	-	-	-	+	-	+	+	
Threshold >=		0.25	0.43	0.53	0.76	0.85	0.85	0.85	0.87	0.93	0.95	1.00
	TP	5	4	4	3	3	3	3	2	2	1	0
	FP	5	5	4	4	3	2	1	1	0	0	0
	TN	0	0	1	1	2	3	4	4	5	5	5
	FN	0	1	1	2	2	2	2	3	3	4	5
→	TPR	1	0.8	0.8	0.6	0.6	0.6	0.6	0.4	0.4	0.2	0
→	FPR	1	1	0.8	0.8	0.6	0.4	0.2	0.2	0	0	0

ROC Curve:



Τεστ Σημαντικότητας

- Έστω ότι έχουμε δύο μοντέλα:
  - Model M1: με accuracy = 85%, που δοκιμάστηκε σε 30 στιγμιότυπα (test εγγραφές)
  - Model M2: με accuracy = 75%, που δοκιμάστηκε σε 5000 στιγμιότυπα
- Μπορούμε να πούμε ότι το μοντέλο M1 είναι καλύτερο από το μοντέλο M2;
  - Πόση εμπιστοσύνη μπορούμε να έχουμε για την ακρίβεια των μοντέλων M1 και M2;
  - Μπορεί η διαφορά στην μέτρηση της απόδοσης να εξηγηθεί ως αποτέλεσμα τυχαίων διακυμάνσεων στο σύνολο test;

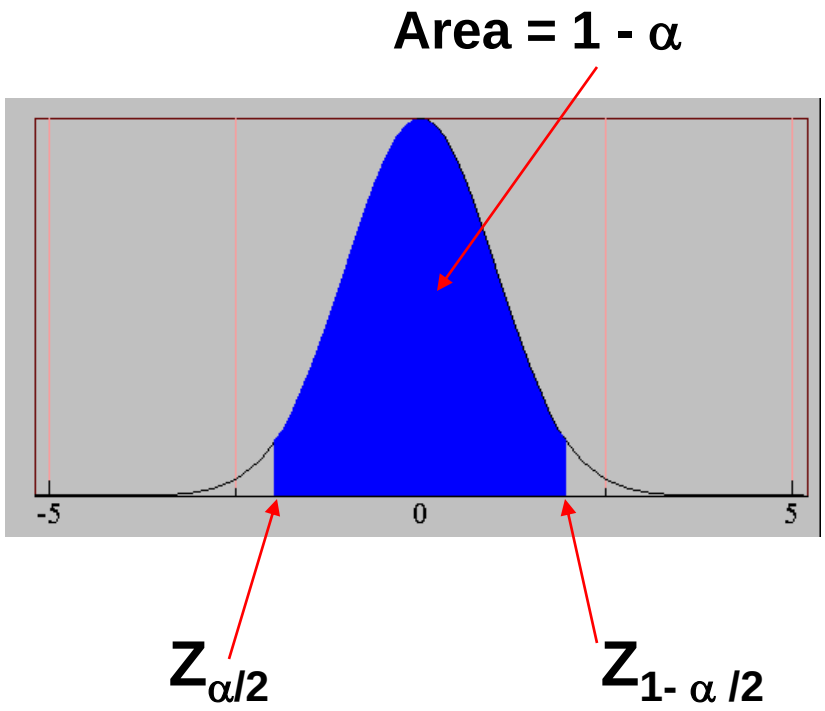
Διάστημα Εμπιστοσύνης της Ακρίβειας

- Η πρόβλεψη μπορεί να θεωρηθεί ως μία δοκιμή Bernoulli
  - Μία δοκιμή Bernoulli έχει μόνο 2 πιθανά αποτελέσματα
  - Πιθανά αποτελέσματα πρόβλεψης: σωστή ή λάθος
  - Μία συλλογή από δοκιμές Bernoulli ακολουθεί μία Διωνυμική Κατανομή:
    - $x \sim Bin(N, p)$ x: ο αριθμός των σωστών προβλέψεων
    - π.χ.: Αν ρίξουμε ένα δίκαιο νόμισμα 50 φορές, πόσες φορές να φέρει κορώνα;
      Αναμενόμενο πλήθος κορόνων = $N \times p = 50 \times 0.5 = 25$
- Δεδομένου του x (πλήθος σωστών προβλέψεων) ή ισοδύναμα του acc=x/N, και του N (πλήθος των στιγμιότυπων test), μπορούμε να προβλέπουμε το p (την πραγματική ακρίβεια του μοντέλου);

Διάστημα Εμπιστοσύνης της Ακρίβειας

- Για μεγάλα σύνολα test (N>30),
  - Η acc ακολουθεί κανονική κατανομή με μέση τιμή p και διακύμανση p(1-p)/N

$$P(Z_{\alpha/2} < \frac{acc - p}{\sqrt{p(1-p)/N}} < Z_{1-\alpha/2}) = 1-\alpha$$



- Διάστημα εμπιστοσύνης για το p:

$$p = \frac{2 \times N \times acc + Z^2_{\alpha/2} \pm \sqrt{Z^2_{\alpha/2} + 4 \times N \times acc - 4 \times N \times acc^2}}{2(N + Z^2_{\alpha/2})}$$

Διάστημα Εμπιστοσύνης της Ακρίβειας

- Θεωρείστε ένα μοντέλο που έχει ακρίβεια 80% όταν αξιολογείται από 100 test στιγμιότυπα:
  - N=100, acc = 0.8
  - Έστω 1-α = 0.95 (95% confidence)
  - Από πίνακες πιθανότητας: $Z_{\alpha/2}$=1.96

1-α	Z
0.99	2.58
0.98	2.33
0.95	1.96
0.90	1.65

N	50	100	500	1000	5000
p(lower)	0.670	0.711	0.763	0.774	0.789
p(upper)	0.888	0.866	0.833	0.824	0.811

Συγκρίνοντας την απόδοση 2 μοντέλων

- Δεδομένου δύο μοντέλων, έστω M1 και M2, ποιο είναι καλύτερο;
  - Το M1 δοκιμάστηκε στο D1 (μεγέθους n1) και προέκυψε ποσοστό σφάλματος e_1
  - Το M2 δοκιμάστηκε στο D2 (μεγέθους n2) και προέκυψε ποσοστό σφάλματος e_2
  - Θεωρούμε ότι τα D1 και D2 είναι ανεξάρτητα μεταξύ τους
  - Αν τα n1 και n2 είναι επαρκώς μεγάλα, τότε έχουμε:

$$e_1 \sim N(\mu_1, \sigma_1)$$
$$e_2 \sim N(\mu_2, \sigma_2)$$

  - Προσέγγιση: $\hat{\sigma}_i = \frac{e_i(1-e_i)}{n_i}$

Συγκρίνοντας την απόδοση 2 μοντέλων

- Για να εξεταστεί αν η διαφορά της απόδοσης είναι στατιστικά σημαντική, έχουμε: d = e1 – e2
  - d ~ $N(d_t,\sigma_t)$ όπου d_t είναι η πραγματική διαφορά
  - Από τη στιγμή που τα D1 και D2 είναι ανεξάρτητα, οι διακυμάνσεις τους αθροίζονται:

$$\sigma_t^2 = \sigma_1^2 + \sigma_2^2 \cong \hat{\sigma}_1^2 + \hat{\sigma}_2^2$$

$$= \frac{e1(1-e1)}{n1} + \frac{e2(1-e2)}{n2}$$

  - Στο (1-α) επίπεδο εμπιστοσύνης, θα έχουμε:

$$d_t = d \pm Z_{\alpha/2}\hat{\sigma}_t$$

Ένα επεξηγηματικό παράδειγμα

- Έστω: M1: n1 = 30, e1 = 0.15
  M2: n2 = 5000, e2 = 0.25
- Έχουμε: d = |e2 – e1| = 0.1 (2-sided test)

$$\hat{\sigma}_d = \frac{0.15(1-0.15)}{30} + \frac{0.25(1-0.25)}{5000} = 0.0043$$

- Στο επίπεδο εμπιστοσύνης 95%, είναι: $Z_{\alpha/2}$=1.96

$$d_t = 0.100 \pm 1.96 \times \sqrt{0.0043} = 0.100 \pm 0.128$$

=> Το διάστημα περιέχει το 0 => η διαφορά μπορεί να μην είναι στατιστικά σημαντική

Συγκρίνοντας την απόδοση 2 αλγορίθμων

- Κάθε αλγόριθμος μπορεί να παράγει k μοντέλα:
  - Ο L1 μπορεί να παράγει τα M11 , M12, …, M1k
  - Ο L2 μπορεί να παράγει τα M21 , M22, …, M2k
- Αν τα μοντέλα δημιουργούνται με τα ίδια σύνολα test D1,D2, …, Dk (π.χ., με cross-validation), τότε:
  - Για κάθε σύνολο υπολογίζουμε το: $d_j = e_{1j} – e_{2j}$
  - Το d_j έχει μέση τιμή d_t και διακύμανση σ_t
  - Εκτίμηση:

$$\hat{\sigma}_t^2 = \frac{\sum_{j=1}^{k}(d_j - \bar{d})^2}{k(k-1)}$$

$$d_t = d \pm t_{1-\alpha,k-1}\hat{\sigma}_t$$