

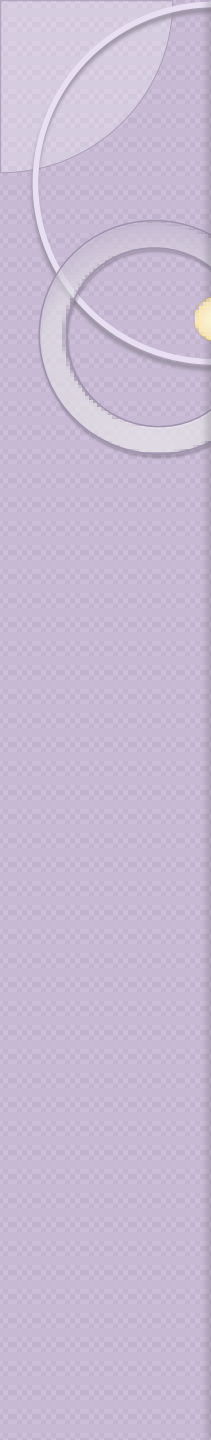


Εξόρυξη Δεδομένων

6: Κατηγοριοποίηση – Αξιολόγηση αποτελεσμάτων

Περιεχόμενα

- Προβλήματα κατηγοριοποίησης
- Αποτίμηση μοντέλου
 - Μέτρα αξιολόγησης
 - Τεχνικές επικύρωσης
- Σημαντικότητα αποτελεσμάτων
- Βελτιώσεις
- Άλλοι αλγόριθμοι



Γενικά προβλήματα

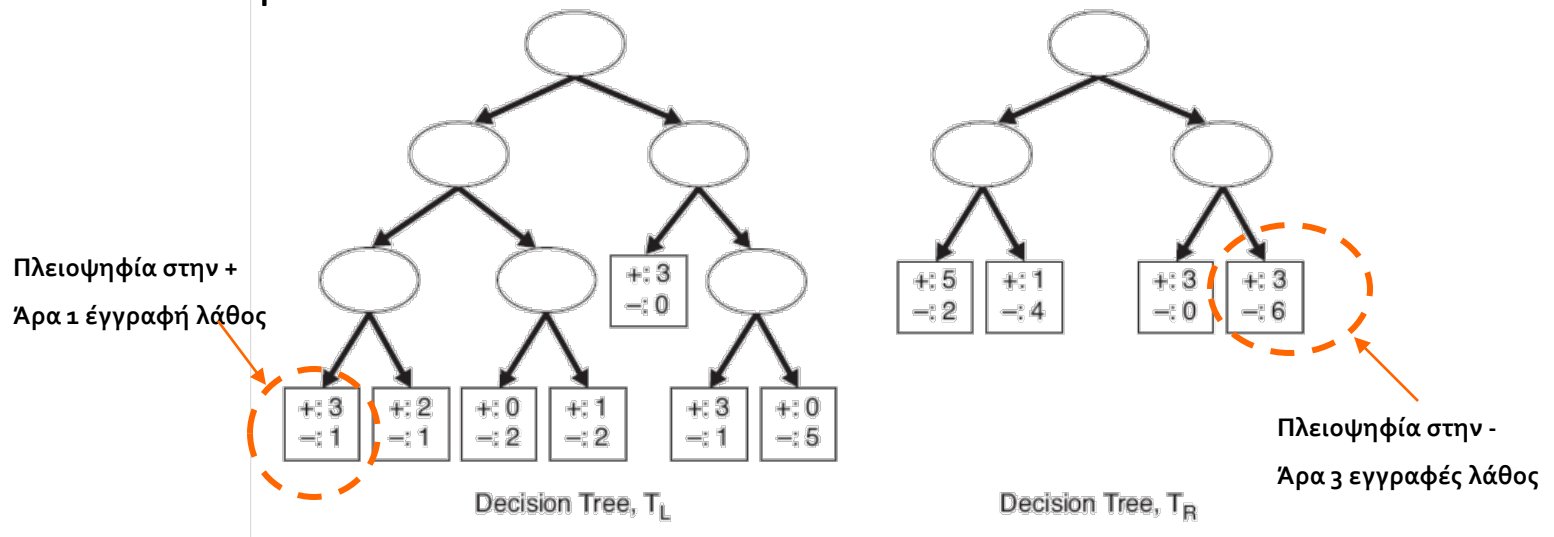
- Λάθος Ταξινόμηση (εκτίμηση λάθους)
- Underfitting and Overfitting
- Τιμές που λείπουν

Λάθη ταξινόμησης

- Errors: οι εγγραφές που ο ταξινομητής τοποθετεί σε λάθος κλάση
 - Εκπαίδευσης (training, resubstitution, apparent): λάθη ταξινόμησης στα δεδομένα του συνόλου εκπαίδευσης
 - Γενίκευσης (generalization): τα αναμενόμενα λάθη ταξινόμησης του μοντέλου σε δεδομένα που δεν έχει δει
- Error rate: ποσοστό εγγραφών που ταξινομούνται σε λάθος κλάση

Εκτίμηση του λάθους

- Λάθη και στα δεδομένα εκπαίδευσης, γιατί χρησιμοποιούμε την πλειοψηφία των εγγραφών σε ένα φύλλο για να αποδώσουμε κλάση



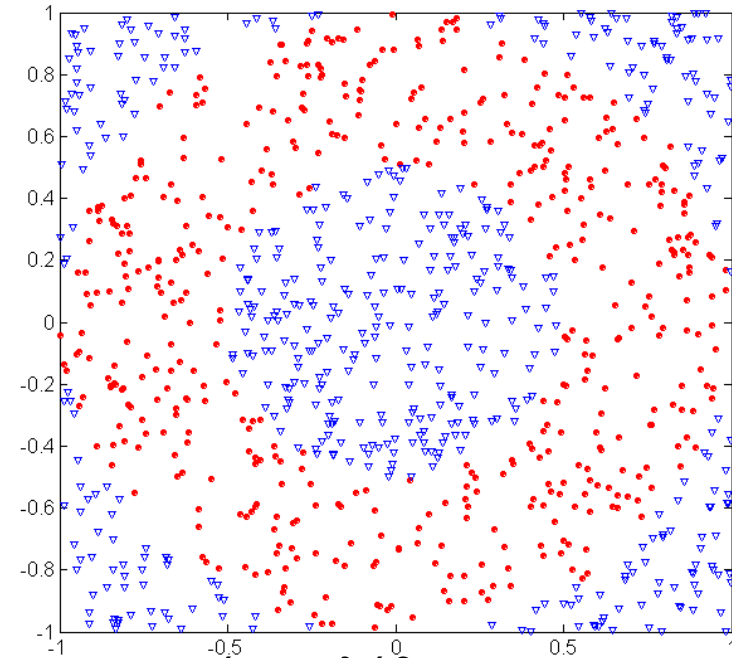
- Παράδειγμα δύο δέντρων για τα ίδια δεδομένα εκπαίδευσης
- Με βάση το λάθος εκπαίδευσης
Αριστερό $4/24 = 0.167$
Δεξί: $6/24 = 0.25$

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

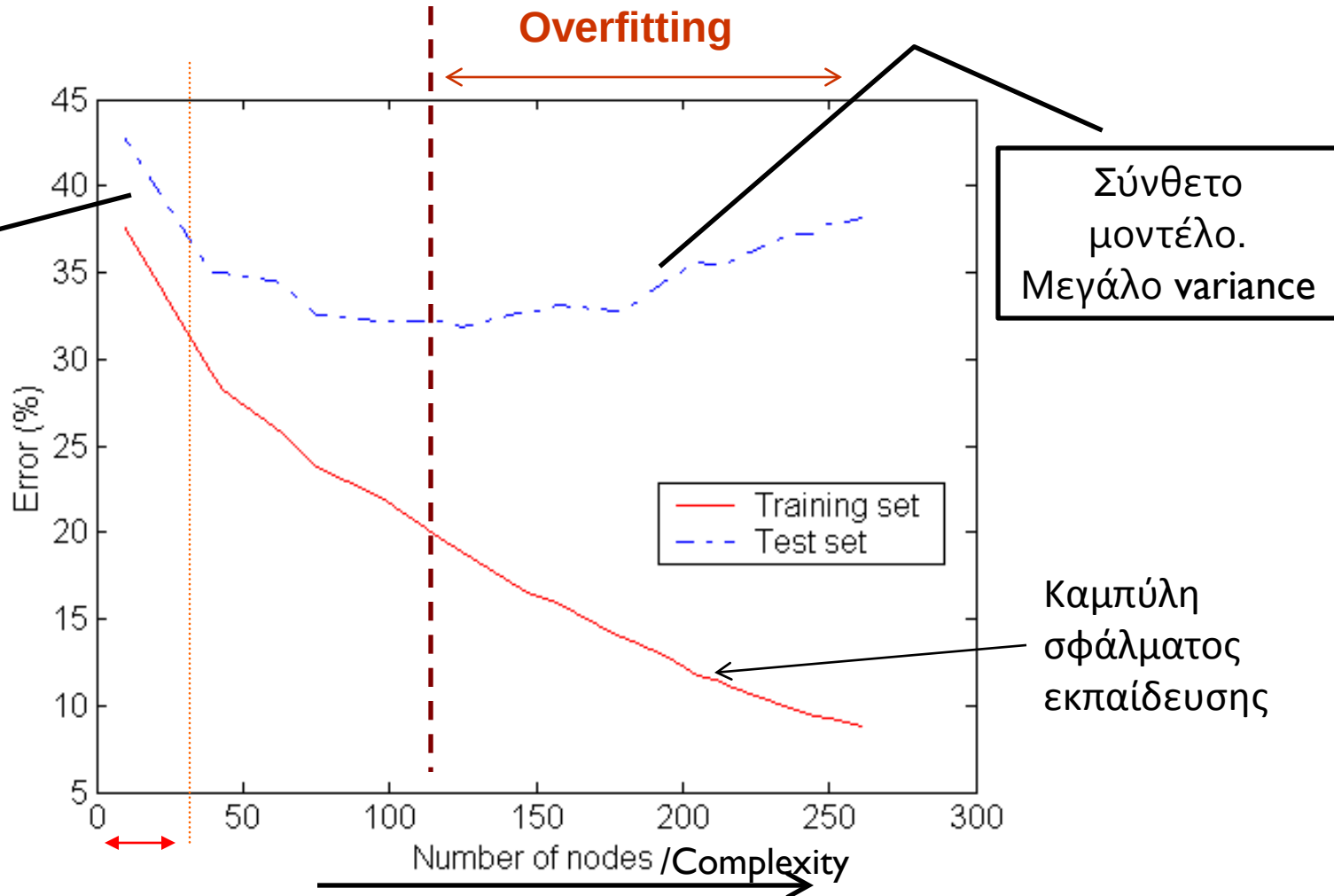
Παράδειγμα

- κλάση 1 (500 κόκκινα σημεία) και
- κλάση 2 (500 μπλε σημεία)
- Για τα σημεία της κλάσης 1 (κόκκινο):
 $0.5 \leq \sqrt{x_{12} + x_{22}} \leq 1$
- Για τα σημεία της κλάσης 2 (μπλε):
 $\sqrt{x_{12} + x_{22}} < 0.5$ or $\sqrt{x_{12} + x_{22}} > 1$



Occam's Razor: Δοθέντων δυο μοντέλων με παρόμοια λάθη γενίκευσης, πρέπει να προτιμάται το απλούστερο από το πιο περίπλοκο

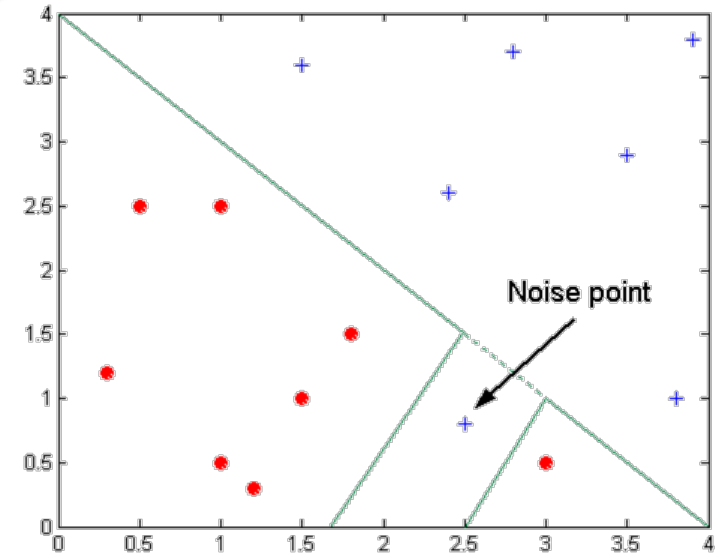
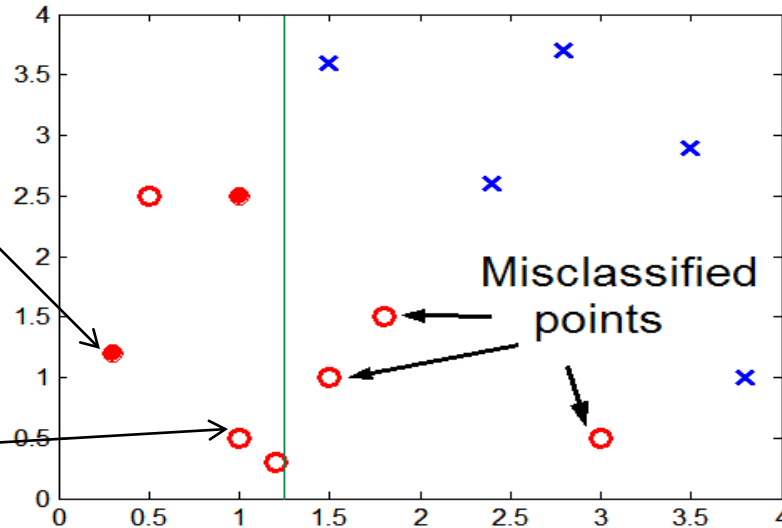
Overfitting vs Underfitting



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

Αιτίες overfitting

- Η ύπαρξη θορύβου
- Ανεπαρκή δείγματα εκπαίδευσης



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 (γνωρισμάτων) και μειώνεται ο αριθμός των δειγμάτων

Λύσεις για το overfitting

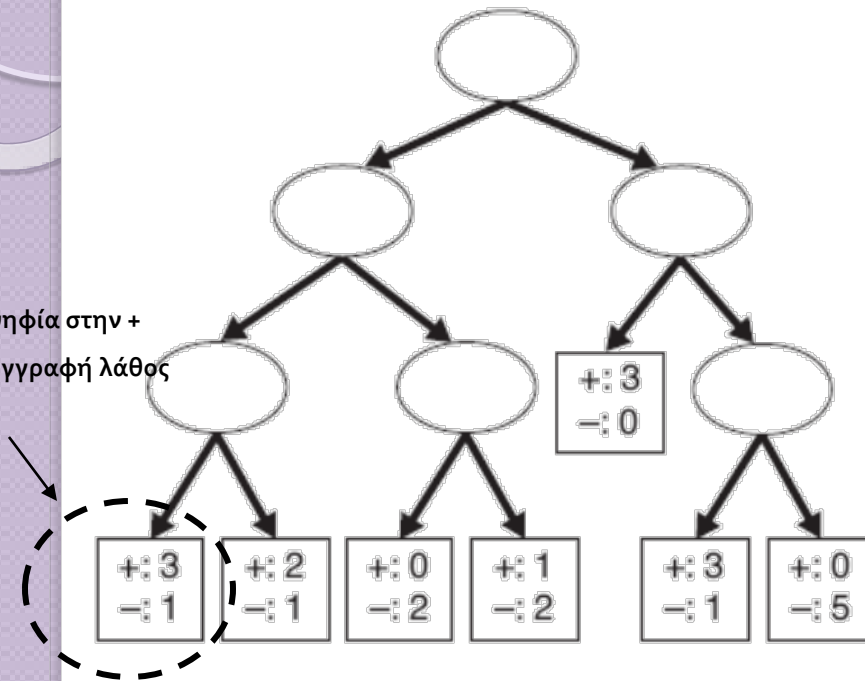
- Pre-pruning: Σταμάτα τον αλγόριθμο ανάπτυξης του δέντρου πριν σχηματιστεί ένα πλήρες δέντρο
 - Σταμάτα όταν όλες οι εγγραφές ανήκουν στην ίδια κλάση
 - Σταμάτα όταν όλες οι τιμές των γνωρισμάτων είναι οι ίδιες
 - Σταμάτα όταν ο αριθμός των εγγραφών είναι μικρότερος από κάποιο προκαθορισμένο κατώφλι
 - Σταμάτα όταν η επέκταση ενός κόμβου δεν βελτιώνει την καθαρότητα (information gain) ή το λάθος γενίκευσης περισσότερο από κάποιο κατώφλι

Λύσεις για το overfitting

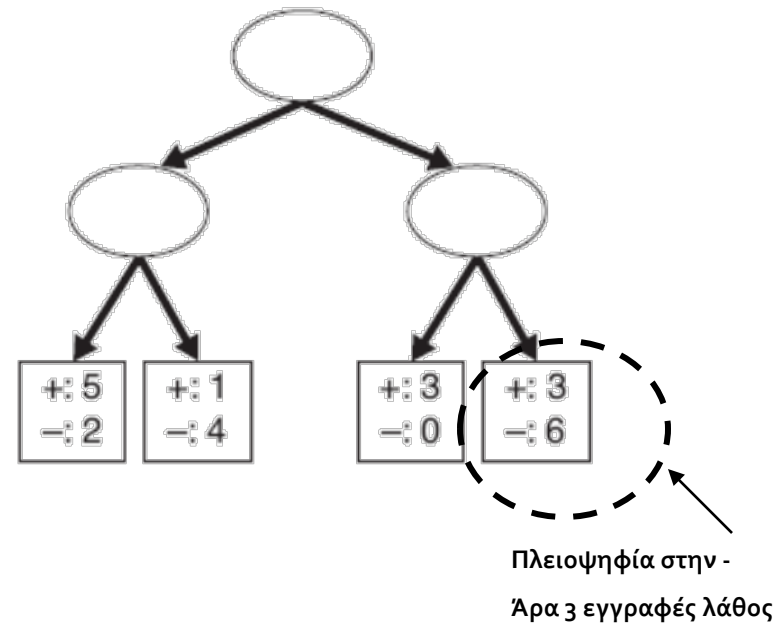
- Post-pruning: Η κατασκευή του δέντρου χωρίζεται σε δύο φάσεις:
 - Φάση (πλήρους) Ανάπτυξης
 - Φάση Ψαλιδίσματος (bottom-up): Αν το λάθος γενίκευσης μειώνεται με το ψαλίδισμα, αντικατέστησε το υποδέντρο με
 - ένα φύλλο - οι ετικέτες κλάσεις του φύλλου καθορίζεται από την πλειοψηφία των κλάσεων των εγγράφων του υποδέντρου (subtree replacement)
 - ένα από τα κλαδιά του (Branch), αυτό που χρησιμοποιείται συχνότερα (subtree raising)

Pruning και λάθος γενίκευσης

Πλειοψηφία στην +
Αρα 1 έγγραφη λάθος



Decision Tree, T_L



Πλειοψηφία στην -
Αρα 3 εγγραφές λάθος

Decision Tree, T_R

- Το δεξί δέντρο (T_R) είναι από ψαλίδισμα του αριστερού δέντρου (T_L) – *sub-tree raising*
- Με βάση το λάθος εκπαίδευσης
Αριστερό $4/24 = 0.167$

Δεξί: $6/24 = 0.25$

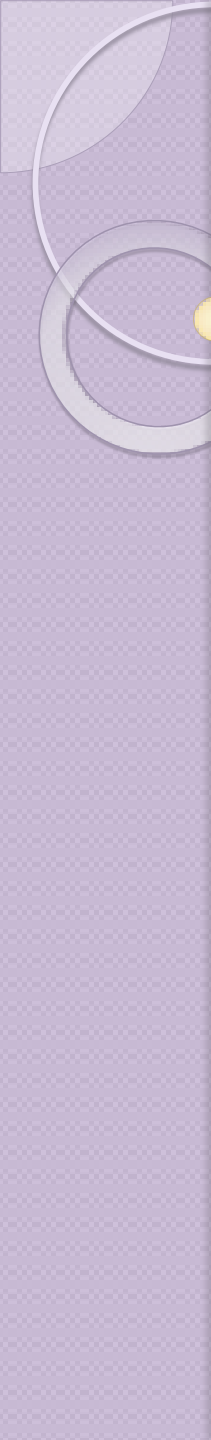
Εκτίμηση του Λάθους Γενίκευσης

- 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

Τιμές που λείπουν

Οι τιμές που λείπουν (missing values) δημιουργούν επιπλέον προβλήματα στην κατασκευή του δέντρου:

- Πώς υπολογίζονται τα μέτρα καθαρότητας;
 - Αγνοούν την εγγραφή με τις ελλείψεις
- Πώς κατανέμονται στα φύλλα οι εγγραφές με τιμές που λείπουν;
 - Με διαφορετικό βάρος σε κάθε φύλλο
- Πώς ταξινομείται μια εγγραφή εκπαίδευσης στην οποία λείπει μια τιμή;
 - Με πιθανότητα σε κάθε κλάση



Αποτίμηση Μοντέλου

Confusion Matrix (Πίνακας Σύγχυσης)

f_{ij} : αριθμός των εγγραφών της κλάσης i που προβλέπονται ως κλάση j

πραγματική ACTUAL CLASS	πρόβλεψη PREDICTED CLASS	
	Class=Yes	Class=No
Class=Yes	f_{11} TP	f_{10} FN
	Class=No	f_{01} FP
Class=No	f_{00} TN	

TP (true positive) f_{11}

FN (false negative) f_{10}

FP (false positive) f_{01}

TN (true negative) f_{00}

Πιστότητα - Accuracy

Το πιο
συνηθισμένο
μέτρο

		PREDICTED CLASS	
ACTUAL CLASS		Class=Yes	Class=No
	Class=Yes	TP f_{11}	FN f_{10}
	Class=No	FP f_{01}	TN f_{00}

= 0

$$\text{Accuracy} = \frac{f_{11} + f_{00}}{f_{11} + f_{00} + f_{01} + f_{10}} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Λόγος Λάθους: Error rate} = \frac{f_{01} + f_{10}}{f_{11} + f_{00} + f_{01} + f_{10}}$$

$$\text{ErrorRate(C)} = 1 - \text{Accuracy(C)}$$

Recall (ανάκληση) – Precision (ακρίβεια)

ACTUAL CLASS	PREDICTED CLASS	
	Class=Yes	Class=No
Class=Yes	TP	FN
Class=No	FP	TN

Precision

$$p = \frac{TP}{TP + FP}$$

Πόσα από τα παραδείγματα που ο ταξινομητής έχει ταξινομήσει ως θετικά είναι πραγματικά θετικά

Όσο πιο μεγάλη η ακρίβεια, τόσο μικρότερος ο αριθμός των FP

Recall

$$r = \frac{TP}{TP + FN}$$

Πόσα από τα θετικά παραδείγματα κατάφερε ο ταξινομητής να βρει

Όσο πιο μεγάλη η ανάκληση, τόσο λιγότερα θετικά παραδείγματα έχουν ταξινομεί λάθος (=TPR)

F measure

$$F_1 = \frac{2rp}{r + p} = \frac{2TP}{2TP + FP + FN}$$

$$F_1 = \frac{2}{1/r + 1/p}$$

Αρμονικό μέσο (Harmonic mean)

- Τείνει να είναι πιο κοντά στο μικρότερο από τα δύο
- Υψηλή τιμή σημαίνει ότι και τα δύο είναι ικανοποιητικά μεγάλα

Καμπύλη ROC

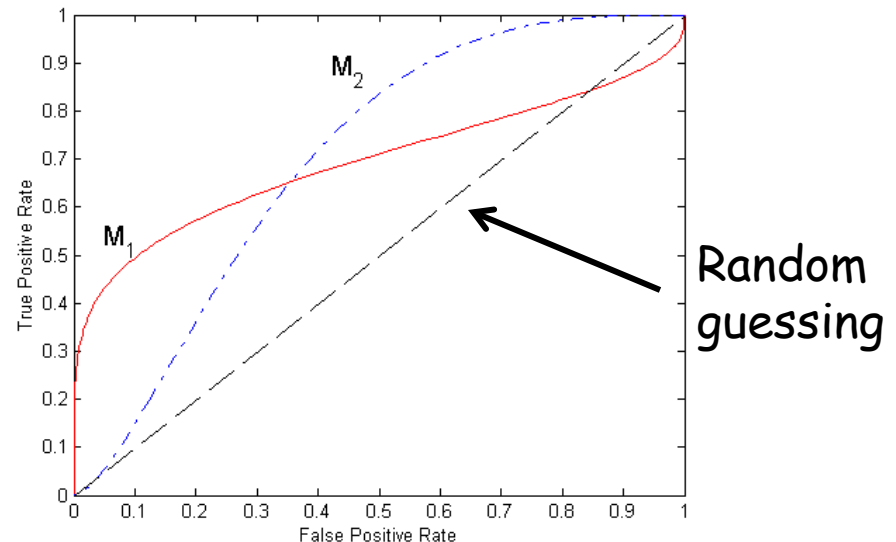
- ROC (Receiver Operating Characteristic Curve)
- Χαρακτηρίζει το trade-off μεταξύ positive hits και false alarms
- Η καμπύλη ROC δείχνει τα TPR [TruePositiveRate] (στον άξονα των y) προς τα FPR [FalsePositiveRate] (στον άξονα των x)
- Η απόδοση κάθε ταξινομητή αναπαρίσταται ως ένα σημείο στην καμπύλη ROC

Πόσα από τα αρνητικά
βρίσκει

$$FPR = \frac{FP}{TN + FP}$$

Πόσα από τα θετικά βρίσκει

$$TPR = \frac{TP}{TP + FN}$$



Έλεγχος Σημαντικότητας (Significance test)

- Έστω δύο μοντέλα:
 - Μοντέλο M1: ακρίβεια = 85%, έλεγχος σε 30 εγγραφές
 - Μοντέλο M2: ακρίβεια = 75%, έλεγχος σε 5000 εγγραφές
- Είναι το M1 καλύτερο από το M2;
 - Πόση εμπιστοσύνη (confidence) μπορούμε να έχουμε για την πιστότητα του M1 και πόση για την πιστότητα του M2;
 - Μπορεί η διαφορά στην απόδοση να αποδοθεί σε τυχαία διακύμανση του συνόλου ελέγχου;

Διάστημα Εμπιστοσύνης (Accuracy's Confidence Interval)

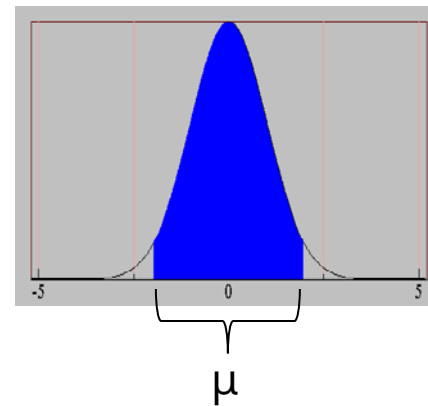
- Η πρόβλεψη μπορεί να θεωρηθεί ως ένα πείραμα Bernoulli
 - Ένα Bernoulli πείραμα έχει δύο πιθανά αποτελέσματα : σωστό ή λάθος
 - Μια συλλογή από πειράματα έχει δυωνυμική κατανομή (Binomial distribution):
 $x \sim \text{Bin}(N, p)$ x : αριθμός σωστών προβλέψεων, N : αριθμός δοκιμών, p (πιθανότητα αποτελέσματος)
 - Πχ: ρίξιμο τίμιου νομίσματος (κορώνα/γράμματα) 50 φορές, αριθμός κεφαλών;

Expected number of heads = $N \times p = 50 \times 0.5 = 25$
- Δοθέντος του x (# σωστών προβλέψεων) ή ισοδύναμα, $\text{acc}=x/N$, και του N (# εγγραφών ελέγχου), μπορούμε να προβλέψουμε το p (την πραγματική πιστότητα του μοντέλο);

- Για μεγάλα σύνολα ελέγχου ($N > 30$), Κανονική κατανομή της ακρίβειας με μέσο p και διακύμανση $\sigma = \sqrt{p(1-p)}$
- Το διάστημα εμπιστοσύνης είναι:

$$\mu = \frac{2 \cdot Z \cdot \sigma}{\sqrt{N}} = \frac{2 Z \sqrt{p(1-p)}}{\sqrt{N}}$$

- όπου p η εκτιμώμενη ακρίβεια στα N δείγματα και
- Z μια συνάρτηση για το Level of Confidence που δίνεται από πίνακες κανονικής κατανομής



Παράδειγμα

- Έστω ένα μοντέλο που έχει accuracy 80% όταν αποτιμάται σε 100 στιγμιότυπα ελέγχου: Ποιο είναι το **διάστημα εμπιστοσύνης** για την πραγματική του πιστότητα (p) με επίπεδο εμπιστοσύνης 95%

- $N=100$, $\text{acc} = 0.8$, $\text{LOC}=95\%$,

- $Z_{\text{LOC}}=1.96$

$$\mu = \frac{2 \cdot 1.96 \cdot \sqrt{0.8 \cdot (1-0.8)}}{\sqrt{100}} = 0.1568$$

- Συνεπώς το $\text{acc} \pm (\mu/2) = 80\% \pm 7.84\%$

LOC	Z
99.9%	3.3
99.0%	2.577
98.5%	2.43
97.5%	2.243
95.0%	1.96
90.0%	1.645
85.0%	1.439
75.0%	1.151

N	50	100	500	1000	5000
p(lower)	0,689126	0,7216	0,764938	0,775208	0,788913
p(upper)	0,910874	0,8784	0,835062	0,824792	0,811087

Πλησιάζει το 80% όσο το N μεγαλώνει



Τεχνικές επικύρωσης

Validation techniques

Validation

- Leave one out (holdout)
- Διαμέριση του αρχικού συνόλου σε δύο ξένα σύνολα: Σύνολο εκπαίδευσης ($2/3$) – Σύνολο Ελέγχου ($1/3$)
 - Κατασκευή μοντέλου με βάση το σύνολο εκπαίδευσης
 - Αποτίμηση μοντέλου με βάση το σύνολο ελέγχου
- (-) Λιγότερες εγγραφές για εκπαίδευση
- (-) Το μοντέλο εξαρτάται από τη σύνθεση των συνόλων εκπαίδευσης και ελέγχου – όσο μεγαλύτερο το σύνολο εκπαίδευσης, τόσο λιγότερο αξιόπιστη η πιστότητα του μοντέλου που υπολογίζεται με το σύνολο ελέγχου
- (-) Τα σύνολα ελέγχου και εκπαίδευσης δεν είναι ανεξάρτητα μεταξύ τους

Validation

- Random Subsampling
 - Τυχαία Λήψη Δειγμάτων: Επανάληψη της μεθόδου για τη βελτίωσή της. Έστω k επαναλήψεις, παίρνουμε το μέσο όρο της ακρίβειας
- Cross validation
 - Κάθε εγγραφή χρησιμοποιείται τον ίδιο αριθμό φορές στην εκπαίδευση και ακριβώς μια φορά για έλεγχο
 - Διαμοίραση των δεδομένων σε k ίσα διαστήματα
 - Κατασκευή του μοντέλου αφήνοντας κάθε φορά ένα διάστημα ως σύνολο ελέγχου και χρησιμοποιώντας όλα τα υπόλοιπα ως σύνολα εκπαίδευσης
 - Επανάληψη k φορές
- Bootstrapping: Δειγματοληψία με επανένταξη
 - Μια εγγραφή που επιλέχθηκε ως δεδομένο εκπαίδευσης, ξαναπαίρνει στο αρχικό σύνολο

5-fold Cross Validation

ONE ITERATION OF A 5-FOLD CROSS-VALIDATION:

1-ST FOLD:



2-ND FOLD:



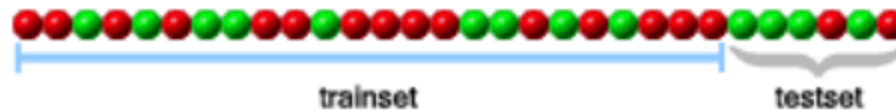
3-RD FOLD:



4-TH FOLD:

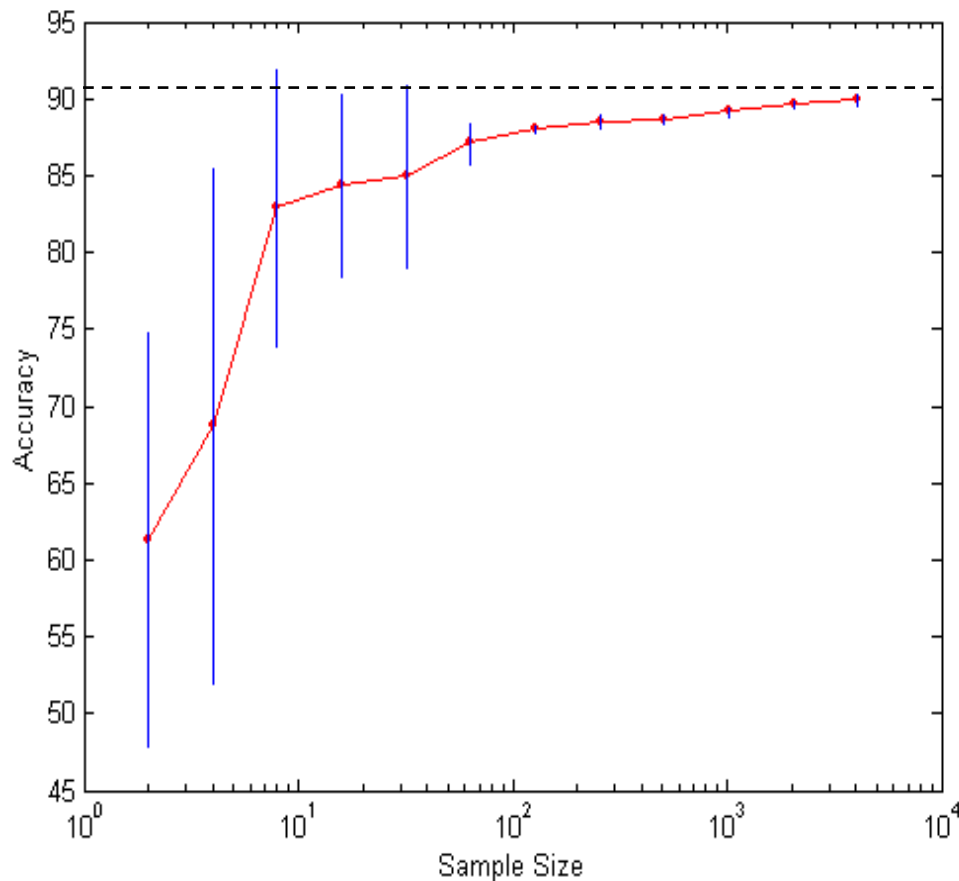


5-TH FOLD:



Καμπύλη μάθησης

Learning Curve



- Η καμπύλη μάθησης δείχνει πως μεταβάλλεται η πιστότητα (accuracy) με την αύξηση του μεγέθους του δείγματος
- Accuracy και Error rate μεταβάλλονται αντίστροφα
- Επίδραση δείγματος μικρού μεγέθους:
 - Bias in the estimate
 - Variance of estimate

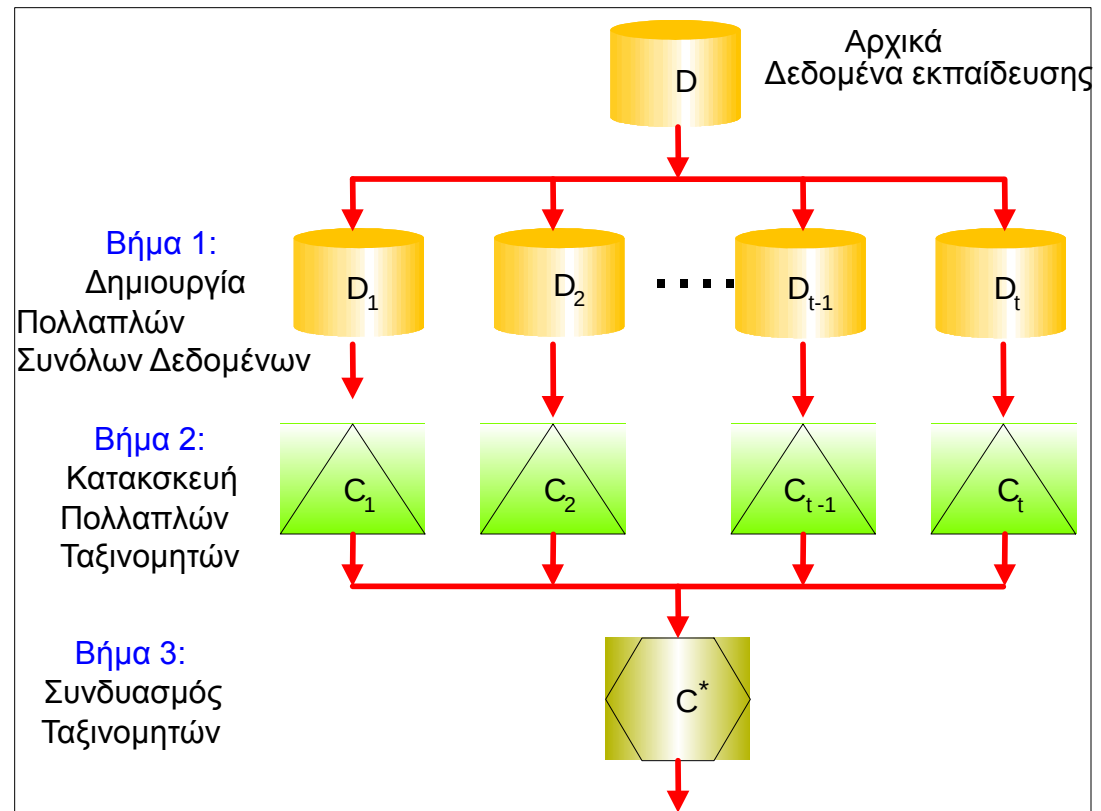
Βελτίωση Απόδοσης

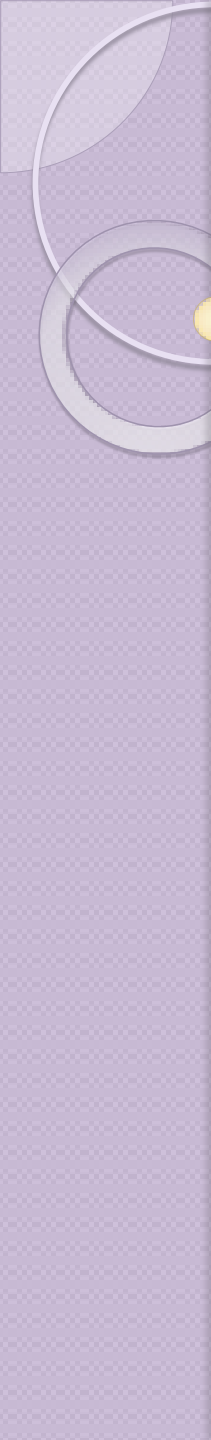
- Ensemble Methods – Σύνολο Μεθόδων

Κατασκευή ενός συνόλου από ταξινομητές από τα δεδομένα εκπαίδευσης $C_1, C_2, \dots, C_t \rightarrow C^*$

Υπολογισμός της κλάσης των δεδομένων συναθροίζοντας (aggregating) τις προβλέψεις των t ταξινομητών

Πως: πχ με πλειοψηφικό σύστημα (Voting majority)





Άλλες μέθοδοι κατηγοριοποίησης

Κατηγοριοποίηση με κανόνες

Κατηγοριοποίηση με Κανόνες

- Κατηγοριοποίηση των εγγραφών με βάση ένα σύνολο από κανόνες της μορφής “if...then...”
- **Κανόνας:** $(\text{Συνθήκη}) \rightarrow y$
 - όπου
 - *Συνθήκη (Condition)* είναι σύζευξη συνθηκών στα γνωρίσματα
 - y η ετικέτα της κλάσης
 - *LHS*: rule antecedent (πρότερο) ή condition (συνθήκη)
 - *RHS*: rule consequent (επακόλουθο ή απότοκο)
 - **Παραδείγματα κανόνων ταξινόμησης:**
 - $(\text{Blood Type}=\text{Warm}) \wedge (\text{Lay Eggs}=\text{Yes}) \rightarrow \text{Birds}$
 - $(\text{Taxable Income} < 50\text{K}) \wedge (\text{Refund}=\text{Yes}) \rightarrow \text{Evade}=\text{No}$

Παράδειγμα

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
human	warm	yes	no	no	mammals
python	cold	no	no	no	reptiles
salmon	cold	no	no	yes	fishes
whale	warm	yes	no	yes	mammals
frog	cold	no	no	sometimes	amphibians
komodo	cold	no	no	no	reptiles
bat	warm	yes	yes	no	mammals
pigeon	warm	no	yes	no	birds
cat	warm	yes	no	no	mammals
leopard shark	cold	yes	no	yes	fishes
turtle	cold	no	no	sometimes	reptiles
penguin	warm	no	no	sometimes	birds
porcupine	warm	yes	no	no	mammals
eel	cold	no	no	yes	fishes
salamander	cold	no	no	sometimes	amphibians
gila monster	cold	no	no	no	reptiles
platypus	warm	no	no	no	mammals
owl	warm	no	yes	no	birds
dolphin	warm	yes	no	yes	mammals
eagle	warm	no	yes	no	birds

R1: (Give Birth = no) $\wedge$ (Can Fly = yes) $\rightarrow$ Birds

R2: (Give Birth = no) $\wedge$ (Live in Water = yes) $\rightarrow$ Fishes

R3: (Give Birth = yes) $\wedge$ (Blood Type = warm) $\rightarrow$ Mammals

R4: (Give Birth = no) $\wedge$ (Can Fly = no) $\rightarrow$ Reptiles

R5: (Live in Water = sometimes) $\rightarrow$ Amphibians

Αξιολόγηση κανόνα

Ένας κανόνας **καλύπτει (covers)** ένα στιγμιότυπο (εγγραφή) αν τα γνωρίσματα του στιγμιότυπου ικανοποιούν τη συνθήκη του κανόνα

Κάλυψη Κανόνα - **Coverage**:
Το ποσοστό των εγγραφών που ικανοποιούν το LHS του κανόνα

Πιστότητα Κανόνα – **Accuracy**:
Το ποσοστό των κανόνων που καλύπτουν και το LHS και το RHS του κανόνα

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

(Status=Single) → No

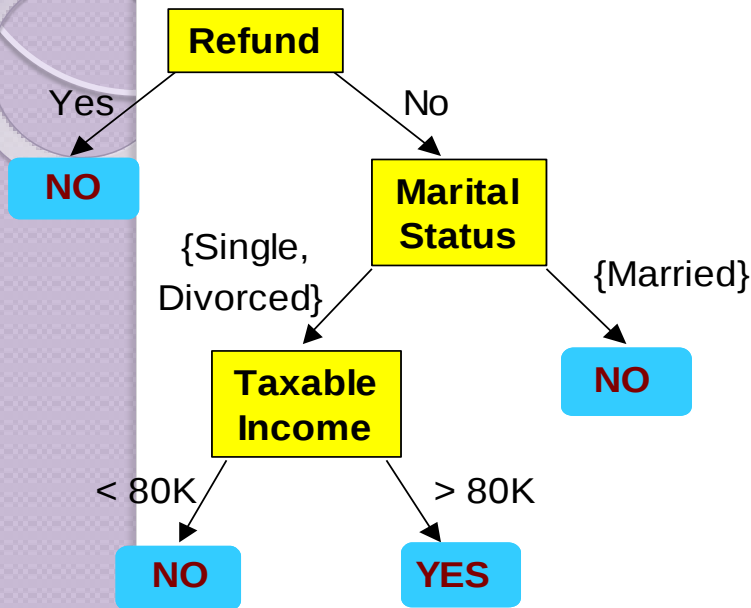
Coverage = 40%

Accuracy = 50%

Περιπτώσεις κάλυψης

- Αμοιβαία αποκλειόμενοι κανόνες (Mutually exclusive rules)
 - Ένας ταξινομητής περιέχει αμοιβαία αποκλειόμενους κανόνες, αν οι κανόνες είναι ανεξάρτητοι ο ένας από τον άλλο
 - Κάθε εγγραφή καλύπτεται από το πολύ έναν κανόνα
- Εξαντλητικοί κανόνες (Exhaustive rules)
 - Ένας ταξινομητής έχει εξαντλητική κάλυψη (coverage) αν καλύπτει όλους τους πιθανούς συνδυασμούς τιμών γνωρισμάτων
 - Κάθε εγγραφή καλύπτεται από τουλάχιστον έναν κανόνα

Κανόνες από δέντρα απόφασης



- Έμμεση μέθοδος
- Αποτυπώνει όλη την πληροφορία του δέντρου
- Ένας κανόνας για κάθε μονοπάτι από τη ρίζα σε φύλλο
 - LHS=κόμβοι στο μονοπάτι
 - RHS=κλάση στο φύλλο
- Αμοιβαία αποκλειόμενοι και εξαντλητικοί κανόνες

(Refund=Yes) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income<80K) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income>80K) ==> Yes

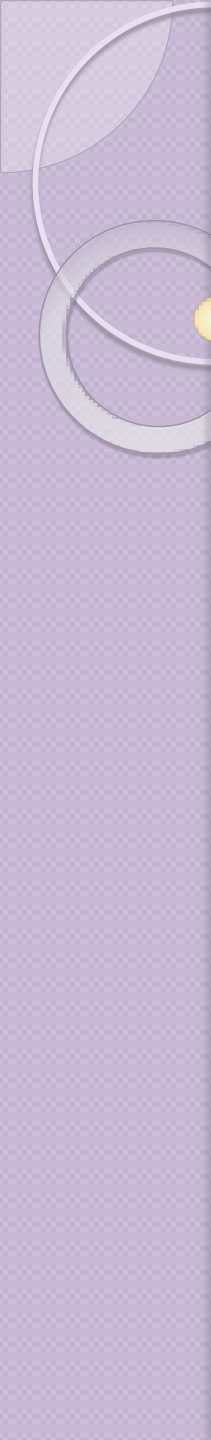
(Refund=No, Marital Status={Married}) ==> No

Απλοποιημένος Κανόνας: (Status=Married) → No

Απλοποίηση (κλάδεμα)

- Όχι αμοιβαία αποκλειόμενοι κανόνες: Μια εγγραφή μπορεί να ενεργοποιήσει παραπάνω από έναν κανόνα
 - Με Διάταξη του συνόλου κανόνων
 - Αν μια εγγραφή ενεργοποιεί πολλούς κανόνες, της ανατίθεται αυτός με τη μεγαλύτερη προτεραιότητα) (decision list)
 - Επιλέγεται ο κανόνας με τις πιο πολλές απαιτήσεις (πχ με το μεγαλύτερο αριθμό όρων) (size ordering)
 - Γίνεται διάταξη των κλάσεων και ανατίθεται στην εγγραφή η κλάση με τη μεγαλύτερη προτεραιότητα (misclassification cost)
 - Χωρίς διάταξη του συνόλου κανόνων – χρήση σχήματος ψηφοφορίας
- Οι κανόνες δεν είναι πια εξαντλητικοί: Μια εγγραφή μπορεί να μην ενεργοποιεί κάποιον κανόνα
 - Επίλυση με χρήση default κλάσης

ΛΥΣΗ



Άλλες μέθοδοι κατηγοριοποίησης

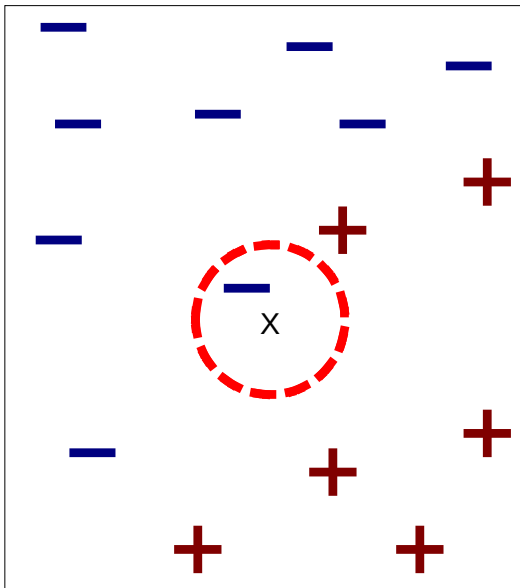
Σε επίπεδο στιγμιοτύπου

Κατηγοριοποίηση με τα στιγμιότυπα

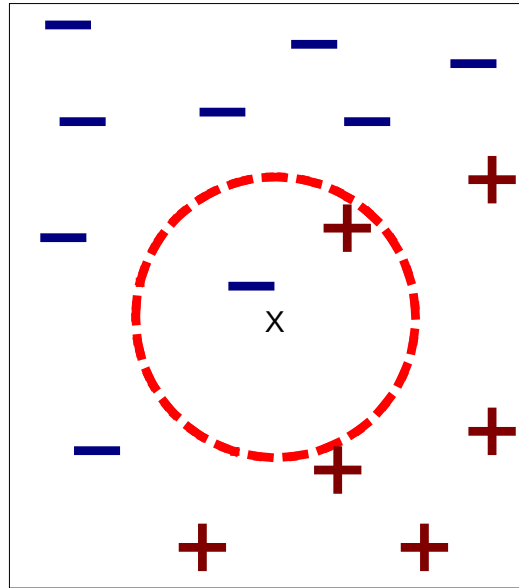
- Μην κατασκευάσεις μοντέλο αν δε χρειαστεί - Lazy Learners
- Αποθήκευσε τις εγγραφές του συνόλου εκπαίδευσης
- Χρησιμοποίησε τις αποθηκευμένες εγγραφές για την εκτίμηση της κλάσης των νέων περιπτώσεων
 - Rote-learner: Θυμάται όλο το σύνολο των δεδομένων εκπαίδευσης και κατηγοριοποιεί μια εγγραφή αν ταιριάζει πλήρως με κάποιο από τα δεδομένα εκπαίδευσης
 - Nearest neighbor – Κοντινότερος Γείτονας: Χρήση των k κοντινότερων “closest” σημείων (nearest neighbors) για την επιλογή κλάσης

Κατηγοριοποίηση Κοντινότερου Γείτονα

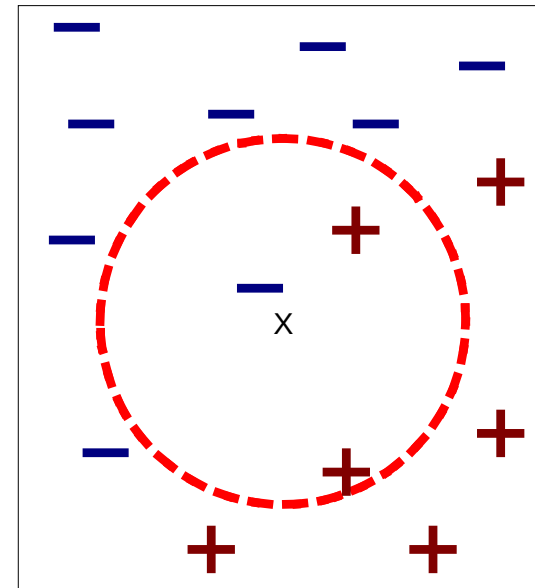
- k -κοντινότεροι γείτονες μιας εγγραφής x είναι τα σημεία που έχουν την k -οστή μικρότερη απόσταση από το x



(a) 1-nearest neighbor



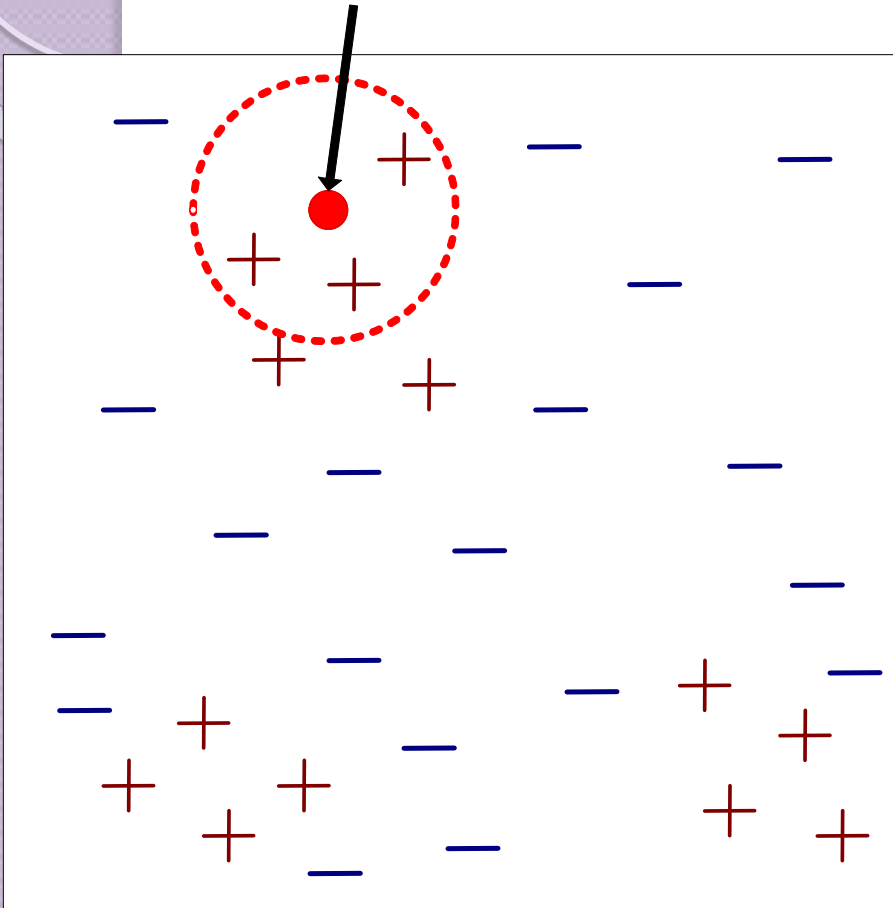
(b) 2-nearest neighbor



(c) 3-nearest neighbor

Κατηγοριοποίηση Κοντινότερου Γείτονα

Άγνωστη Εγγραφή

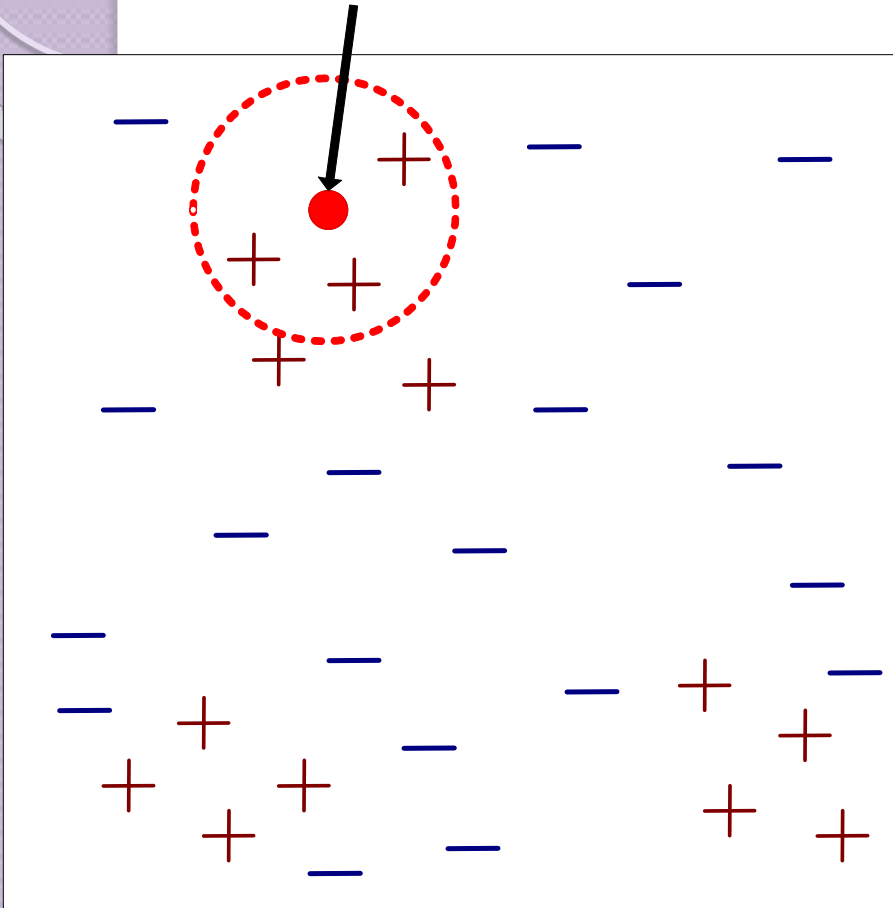


Χρειάζεται

1. Το σύνολο των αποθηκευμένων εγγραφών
2. **Distance Metric** Μετρική απόστασης για να υπολογίσουμε την απόσταση μεταξύ εγγραφών
3. Την τιμή του **k**, δηλαδή τον αριθμό των κοντινότερων γειτόνων που πρέπει να ανακληθούν

Κατηγοριοποίηση Κοντινότερου Γείτονα

Άγνωστη Εγγραφή



Για να ταξινομηθεί μια άγνωστη εγγραφή:

- Υπολογισμός της απόστασης από τις εγγραφές του συνόλου
- Εύρεση των k κοντινότερων γειτόνων
- Χρήση των κλάσεων των κοντινότερων γειτόνων για τον καθορισμό της κλάσης της άγνωστης εγγραφής - π.χ., με βάση την πλειοψηφία (majority vote)

Κατηγοριοποίηση Κοντινότερου Γείτονα

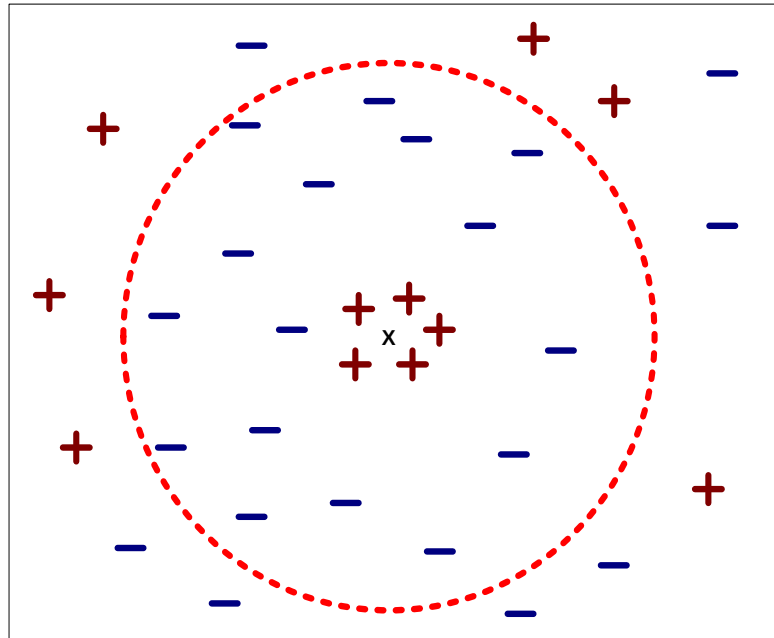
- Απόσταση μεταξύ εγγραφών:
 - Πχ ευκλείδεια απόσταση

$$d(p, q) = \sqrt{\sum_i (p_i - q_i)^2}$$

- Καθορισμός κατηγορίας
 - Απλά τη πλειοψηφική κατηγορία
 - Βάρος σε κάθε ψήφο με βάση την απόσταση
 - weight factor, $w = 1/d^2$

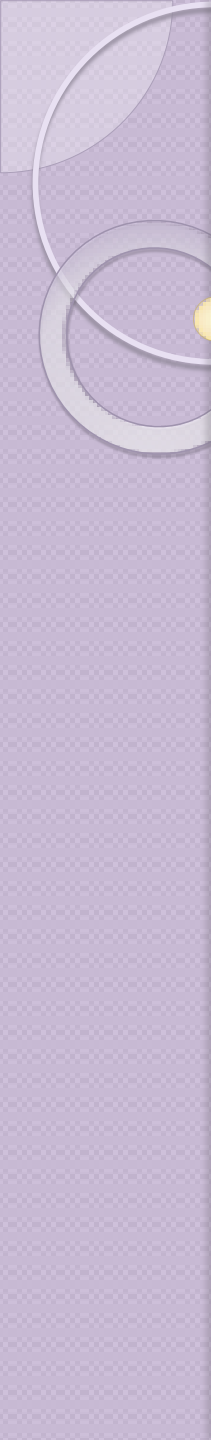
Κατηγοριοποίηση Κοντινότερου Γείτονα

- Επιλογή της τιμής του k :
 - k πολύ μικρό, ευαισθησία στα σημεία θορύβου
 - k πολύ μεγάλο, η γειτονιά μπορεί να περιέχει σημεία από άλλες κλάσεις



Κατηγοριοποίηση Κοντινότερου Γείτονα

- Θέματα Κλιμάκωσης
 - Τα γνωρίσματα ίσως πρέπει να κλιμακωθούν ώστε οι αποστάσεις να μην κυριαρχηθούν από κάποιο γνώρισμα
- Παράδειγμα:
 - Το ύψος κυμαίνεται από 1.5m ως 1.8m
 - Το βάρος κυμαίνεται από 90lb ως 300lb
 - Το εισόδημα από 10K € ως 100K €
- Δεν κατασκευάζεται μοντέλο, μεγάλο κόστος για την ταξινόμηση
- Πολλές διαστάσεις (κατάρα των διαστάσεων)
- Θόρυβο (ελάττωση μέσω k-γειτόνων)



Πιθανοτικές Μέθοδοι

Bayesian Classification

- Ένας κατηγοριοποιητής που βασίζεται στην στατιστική: κάνει πιθανοτική πρόβλεψη, π.χ., προβλέπει την πιθανότητα ένα δείγμα να ανήκει σε κάποια κλάση
- Στηρίζεται στο θεώρημα του Bayes.
- Απόδοση: Ο απλός Bayesian classifier (*naïve*), έχει απόδοση συγκρίσιμη με τα δέντρα απόφασης και ορισμένους κατηγοριοποιητές με νευρωνικά δίκτυα

Θεώρημα Bayes

- Έστω $\mathbf{X}$ ένα δείγμα
- Έστω H η υπόθεση ότι το $\mathbf{X}$ ανήκει στην κλάση C
- Η μέθοδος κατηγοριοποίησης καθορίζει την πιθανότητα $P(H|\mathbf{X})$ η υπόθεση να ισχύει δοθείσας της παρατήρησης του συγκεκριμένου δείγματος $\mathbf{X}$
 - **Παράδειγμα:** ποια η πιθανότητα ο πελάτης $\mathbf{X}$ να αγοράσει έναν υπολογιστή με δεδομένη την ηλικία του και το εισόδημά του

Θεώρημα Bayes

- Ποια η πιθανότητα ο πελάτης X να αγοράσει έναν υπολογιστή με δεδομένη την ηλικία του και το εισόδημά του
- $P(H)$ (*prior probability*), αρχική πιθανότητα
 - Π.χ. ο X να αγοράσει υπολογιστή ανεξάρτητα από ηλικία, εισόδημα, κλπ.
- $P(X)$: πιθανότητα να εμφανιστεί το συγκεκριμένο δείγμα
- $P(X|H)$ (*posterior probability*), η πιθανότητα να δούμε το δείγμα X αν ισχύει η υπόθεση
 - Π.χ. αν ο X αγοράσει υπολογιστεί ποια η πιθανότητα να έχει μέσο εισόδημα και ηλικία 31..40

$$P(H | X) = \frac{P(X|H)P(H)}{P(X)}$$

Naïve Bayesian Classifier

- Έστω D ένα σύνολο από δείγματα εκπαίδευσης $\mathbf{X} = (x_1, x_2, \dots, x_n)$ γνωστών κλάσεων, όπου x_i τα γνωρίσματα κάθε δείγματος
- Έστω m κλάσεις $C_1, C_2, \dots, C_m$.
- Στόχος είναι να βρούμε τη μέγιστη posterior πιθανότητα, π.χ. το μέγιστο $P(C_i | \mathbf{X})$ για κάθε κλάση:

$$P(C_i | \mathbf{X}) = \frac{P(\mathbf{X} | C_i)P(C_i)}{P(\mathbf{X})}$$

- Αφού το $P(\mathbf{X})$ είναι σταθερό για όλες τις κλάσεις αρκεί να μεγιστοποιείται το:

$$P(C_i | \mathbf{X}) = P(\mathbf{X} | C_i)P(C_i)$$

Παραδοχή

- Τα γνωρίσματα είναι ανεξάρτητα μεταξύ τους
- Η πιθανότητα του x_i δεν επηρεάζει αυτή του x_j δοθείσης μιας κλάσης

$$P(X|C_i) = \prod_{k=1}^n P(x_k | C_i) = P(x_1 | C_i) \times P(x_2 | C_i) \times \dots \times P(x_n | C_i)$$

- Η παραδοχή μειώνει το κόστος υπολογισμού σημαντικά: Αρκεί να μετρήσουμε την κατανομή των κλάσεων

ΣΥΝΕΠΩΣ

- Αν το A_k είναι κατηγορικό, $P(x_k | C_i)$ είναι ο αριθμός των δειγμάτων κλάσης C_i που έχουν τιμή x_k στο A_k δια το $|C_{i,D}|$ (αριθμός δειγμάτων κλάσης C_i συνολικά)
- Αν το A_k έχει συνεχείς τιμές, το $P(x_k | C_i)$ συνήθως υπολογίζεται βάσει μιας Gaussian κατανομής με μέση τιμή μ και τυπ. απόκλιση σ

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- Οπότε $P(x_k | C_i)$ είναι

$$P(X | C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i})$$

Παράδειγμα

Κλάση: C1:buys_computer = 'yes' C2:buys_computer = 'no'

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

Δείγμα

X = (age <= 30 , income = medium, student = yes, credit_rating = fair)

Παράδειγμα

- $P(C_i)$: $P(\text{buys_computer} = \text{"yes"}) = 9/14 = 0.643$
 $P(\text{buys_computer} = \text{"no"}) = 5/14 = 0.357$
- $P(X | C_i)$ για κάθε κλάση
 - $P(\text{age} = \text{"<=30"} | \text{buys_computer} = \text{"yes"}) = 2/9 = 0.222$
 - $P(\text{age} = \text{"<= 30"} | \text{buys_computer} = \text{"no"}) = 3/5 = 0.6$
 - $P(\text{income} = \text{"medium"} | \text{buys_computer} = \text{"yes"}) = 4/9 = 0.444$
 - $P(\text{income} = \text{"medium"} | \text{buys_computer} = \text{"no"}) = 2/5 = 0.4$
 - $P(\text{student} = \text{"yes"} | \text{buys_computer} = \text{"yes"}) = 6/9 = 0.667$
 - $P(\text{student} = \text{"yes"} | \text{buys_computer} = \text{"no"}) = 1/5 = 0.2$
 - $P(\text{credit_rating} = \text{"fair"} | \text{buys_computer} = \text{"yes"}) = 6/9 = 0.667$
 - $P(\text{credit_rating} = \text{"fair"} | \text{buys_computer} = \text{"no"}) = 2/5 = 0.4$

Δείγμα

$X = (\text{age} \leq 30, \text{income} = \text{medium}, \text{student} = \text{yes}, \text{credit_rating} = \text{fair})$

Παράδειγμα

$P(X|C_i)$:

$$P(X|\text{buys_computer} = \text{"yes"}) = 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044$$

$$P(X|\text{buys_computer} = \text{"no"}) = 0.6 \times 0.4 \times 0.2 \times 0.4 = 0.019$$

$P(X|C_i) * P(C_i)$:

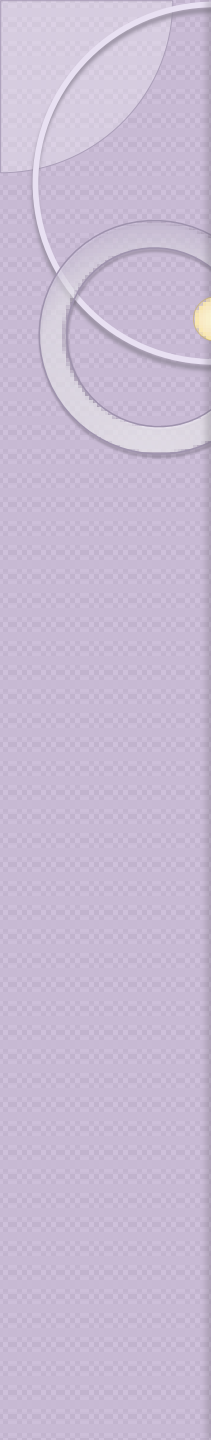
$$P(X|\text{buys_computer} = \text{"yes"}) * P(\text{buys_computer} = \text{"yes"}) = 0.028$$

$$P(X|\text{buys_computer} = \text{"no"}) * P(\text{buys_computer} = \text{"no"}) = 0.007$$

Άρα το X ανήκει στην κλάση (" $\text{buys_computer} = \text{yes}$ ")

Δείγμα

$X = (\text{age} \leq 30, \text{income} = \text{medium}, \text{student} = \text{yes}, \text{credit_rating} = \text{fair})$



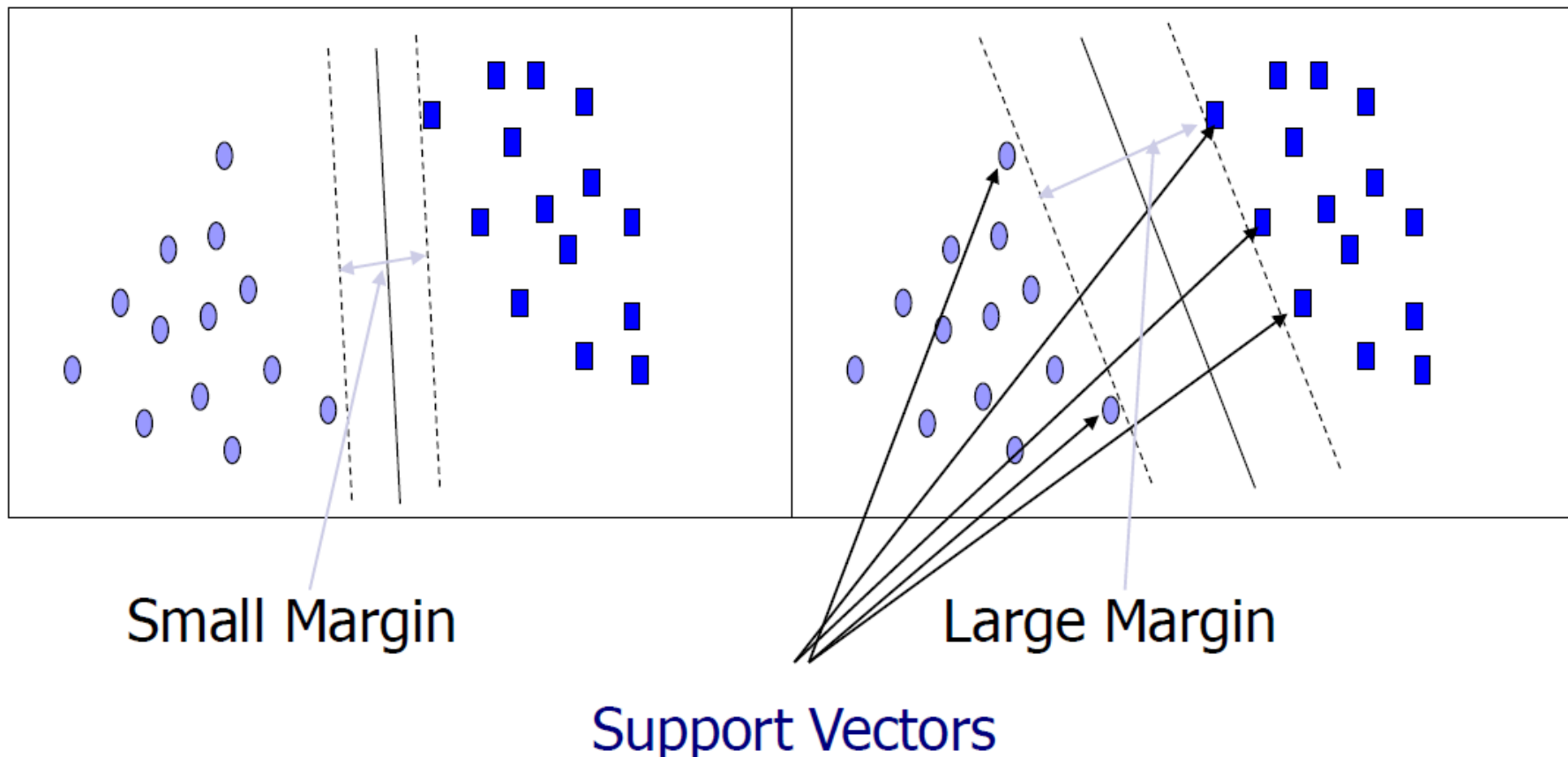
Support Vector Machines

SVM—Support Vector Machines

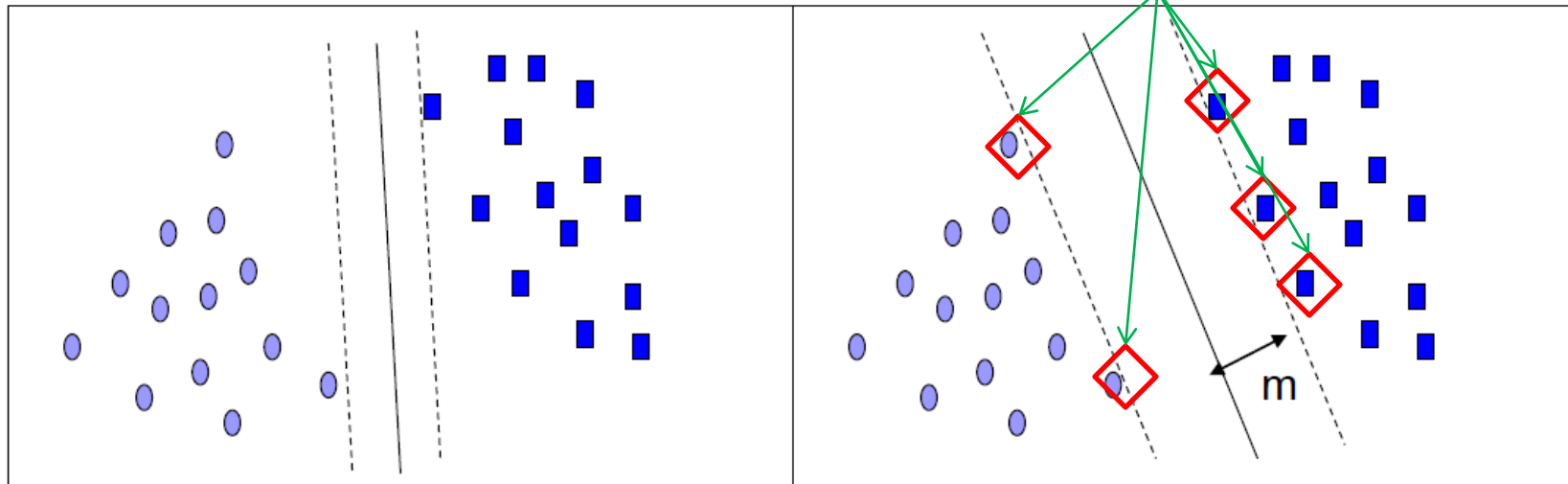
- Μια νέα μέθοδος κατηγοριοποίησης για γραμμικά και μη γραμμικά δεδομένα
- Χρησιμοποιεί μη γραμμική απεικόνιση για να μετασχηματίσει τα αρχικά δεδομένα εκπαίδευσης σε υψηλότερες διαστάσεις
- Με τη νέα διάσταση, αναζητεί για το βέλτιστο γραμμικό υπερεπίπεδο που διαχωρίζει τα δεδομένα
- Με την κατάλληλη μη γραμμική απεικόνιση σε μια ικανοποιητική υψηλότερη διάσταση, δεδομένα δύο κατηγοριών μπορούν πάντοτε να διαχωρίζονται από ένα υπερεπίπεδο
- Η μέθοδος βρίσκει το υπερεπίπεδο χρησιμοποιώντας support vectors και όρια που ορίζονται από αυτούς

Βασική ιδέα

- Βρες το περιθώριο ενός γραμμικού κατηγοριοποιητή μέχρι το οποίο μπορεί να μετακινηθεί χωρίς να χάσει κάποιο σημείο

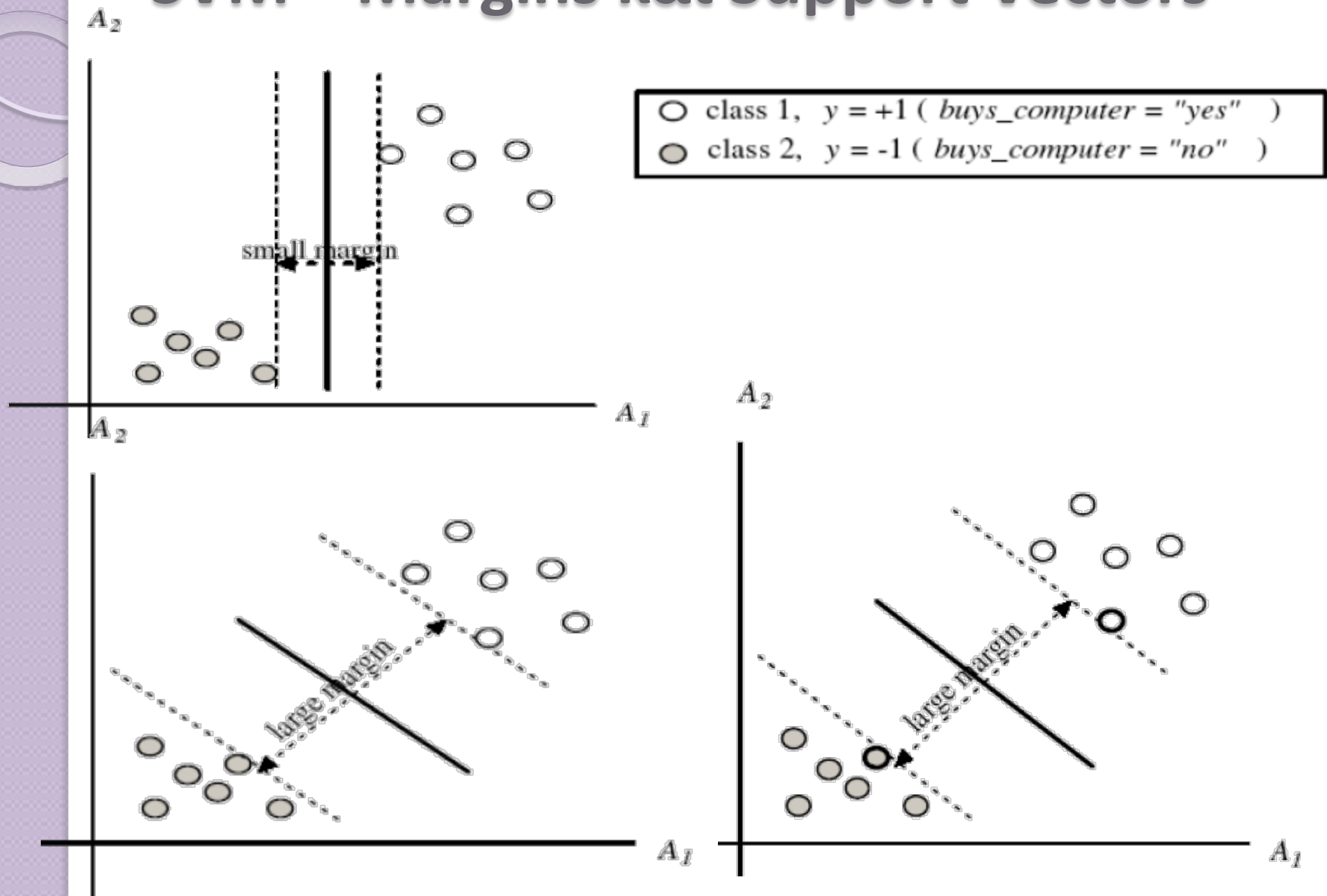


SVM – Δεδομένα γραμμικά διαχωρίσιμα



- Υπάρχει ένα μη πεπερασμένο σύνολο επιπέδων (hyperplanes) που χωρίζουν τις δύο κατηγορίες, αλλά θέλουμε να βρούμε την καλύτερη από αυτές
- Αυτή που ελαχιστοποιεί το λάθος κατηγοριοποίησης σε δεδομένα που δεν είναι γνωστά
- Το SVM αναζητεί το υπερεπίπεδο με το μεγαλύτερο περιθώριο (maximum marginal hyperplane)

SVM—Margins και Support Vectors



SVM – Γραμμικά διαχωρίσιμα

- Ένα υπερεπίπεδο διαχωρισμού ορίζεται:

$$W^*X+b=0$$

όπου $W=\{w_1, w_2, \dots, w_n\}$ ένα διάνυσμα βαρών και b μια σταθερά

- Για 2 διαστάσεις $w_0 + w_1x_1 + w_2x_2 = 0$
- Το υπερεπίπεδο ορίζει τις πλευρές των ορίων:
 - $H_1: w_0 + w_1x_1 + w_2x_2 \geq 1$ για $y_i = +1$ και
 - $H_2: w_0 + w_1x_1 + w_2x_2 \leq -1$ για $y_i = -1$
- Κάθε στοιχείο του συνόλου εκπαίδευσης που πέφτει πάνω στο υπερεπίπεδο H_1 ή H_2 (στις πλευρές που ορίζει το margin) είναι support vector
- Αυτό είναι πρόβλημα βελτιστοποίησης με περιορισμούς (Quadratic objective function with linear constraints)

Αποτελεσματικά σε πολλές διαστάσεις

- Η πολυπλοκότητα του κατηγοριοποιητή χαρακτηρίζεται από τον αριθμό των support vectors και όχι από τις διαστάσεις των δεδομένων
- Τα support vectors είναι βασικά ή κρίσιμα παραδείγματα εκπαίδευσης – βρίσκονται κοντά στο όριο απόφασης
- Αν όλα τα άλλα παραδείγματα αφαιρεθούν και ξαναγίνει εκπαίδευση, θα βρεθεί το ίδιο υπερεπίπεδο διαχωρισμού
- Ο αριθμός των support vectors που βρίσκεται μπορεί να χρησιμοποιηθεί για να υπολογίσουμε ένα (άνω) όριο του αναμενόμενου λάθους του SVM κατηγοριοποιητή, το οποίο είναι ανεξάρτητο από τις διαστάσεις των δεδομένων
- Ένα SVM με μικρό αριθμό support vectors μπορεί να έχει καλή γενίκευση και για υψηλής διάστασης δεδομένα

Μειονεκτήματα των SVMs

- Ευαίσθητα στο θόρυβο
- Υποθέτει δυαδική κατηγοριοποίηση (2 κλάσεις)
- Για να έχω κατηγοριοποιητή πολλών (μ) κλάσεων
 - Εκπαιδεύω μ -κατηγοριοποιητές SVM
 - Για να αποφασίσω για μια νέα είσοδο, προβλέπω την απάντηση για κάθε SVM και βρίσκω ποιος τοποθετεί την πρόβλεψη στην περιοχή των θετικών

Κατηγοριοποίηση κειμένων

- Κάθε κείμενο είναι μια συλλογή λέξεων (bag of words)

$$\phi_i(x) = \frac{\text{tf}_i \log(\text{idf}_i)}{\kappa},$$

- Για κάθε κείμενο x υπολογίζω το $\Phi(x)$
- Η απόσταση δυο κειμένων x και z είναι $\Phi(x) \cdot \Phi(z)$
- Εδώ χρησιμοποιείται μια συνάρτηση που καλείται **Kernel**
- Γιατί SVM
 - Χώρος υψηλών διαστάσεων
 - Λίγα ασύνδετα γνωρίσματα (dense concept)
 - Αραιά διανύσματα κειμένων (sparse instances)
 - Τα προβλήματα κατηγοριοποίησης κειμένου είναι γραμμικά διαχωρίσιμα