k-Nearest Neighbors

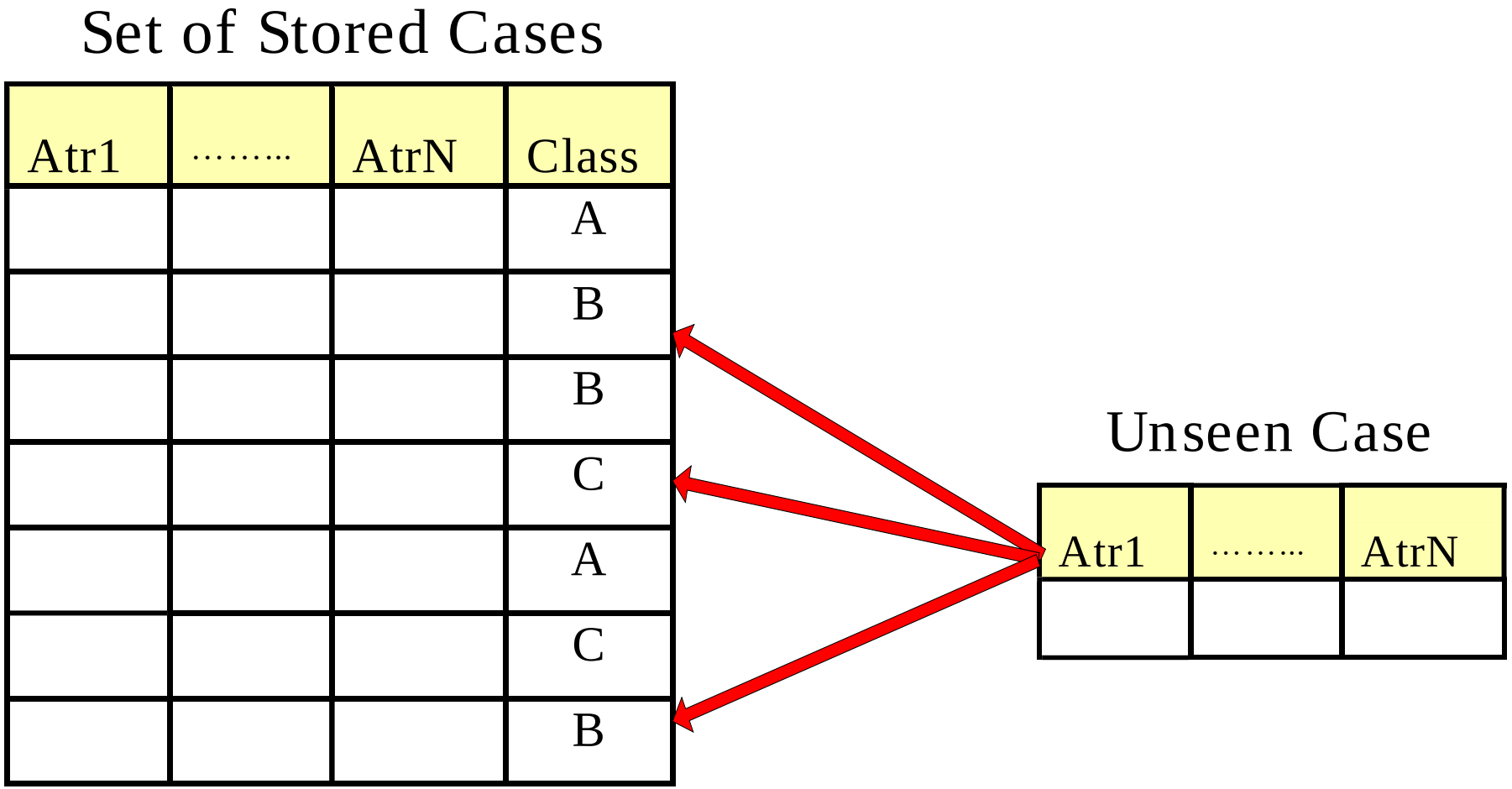
Instance based Class.

- Δημιουργήστε μια βάση δεδομένων με προηγούμενες παρατηρήσεις
- Για να κάνετε μια πρόβλεψη για ένα νέο στοιχείο x', βρείτε το πιο παρόμοιο στοιχείο βάσης δεδομένων x και χρησιμοποιήστε την έξοδο του f(x) για f(x')
- Παρέχει μια τοπική προσέγγιση για τη συνάρτηση ή την έννοια του στόχου

- Χρειάζεστε:
  - Μια μέτρηση απόστασης (για τον προσδιορισμό της ομοιότητας)
  - Αριθμός γειτόνων που πρέπει να συμβουλευτείτε
  - Μέθοδος για το συνδυασμό των εξόδων των γειτόνων

Instance based Class.

- Αποθηκεύστε τα αρχεία εκπαίδευσης
- Χρησιμοποιήστε αρχεία εκπαίδευσης για να προβλέψετε την ετικέτα τάξης των αγνώστων περιπτώσεων

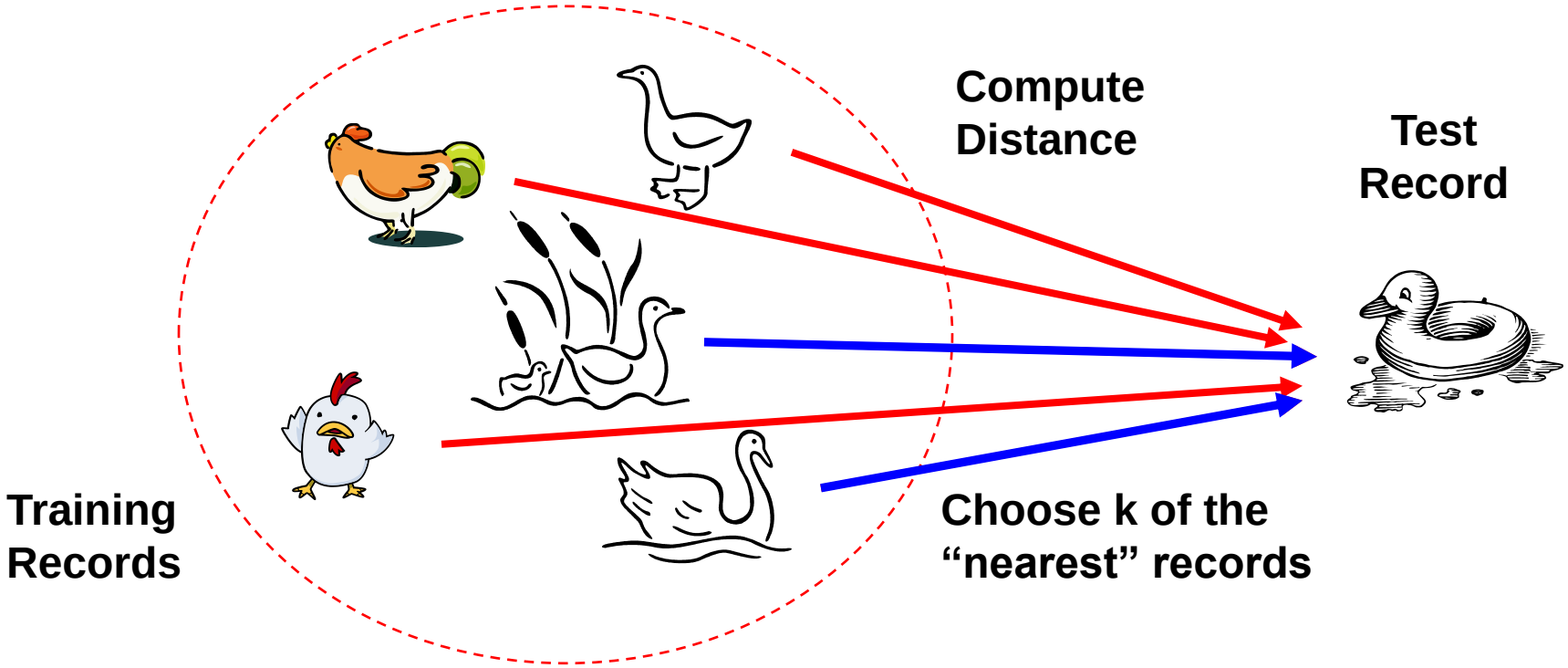


Instance based Class.

- Ιδέα:
  - Παρόμοια παραδείγματα έχουν παρόμοια ετικέτα.
  - Ταξινομήστε νέα παραδείγματα όπως παρόμοια παραδείγματα εκπαίδευσης.
- Αλγόριθμος:
  - Δίνεται κάποιο νέο παράδειγμα x για το οποίο πρέπει να προβλέψουμε την κλάση του y
  - Βρείτε τα περισσότερα παρόμοια παραδείγματα εκπαίδευσης
  - Ταξινομήστε το "μου αρέσει" αυτά τα πιο παρόμοια παραδείγματα
- Ερωτήσεις:
  - Πώς να προσδιορίσετε την ομοιότητα;
  - Πόσα παρόμοια παραδείγματα εκπαίδευσης πρέπει να ληφθούν υπόψη;
  - Πώς να επιλύσετε τις ασυνέπειες μεταξύ των παραδειγμάτων εκπαίδευσης;

Instance based Class.

- Αν περπατάει σαν πάπια, σαν πάπια, τότε μάλλον είναι πάπια

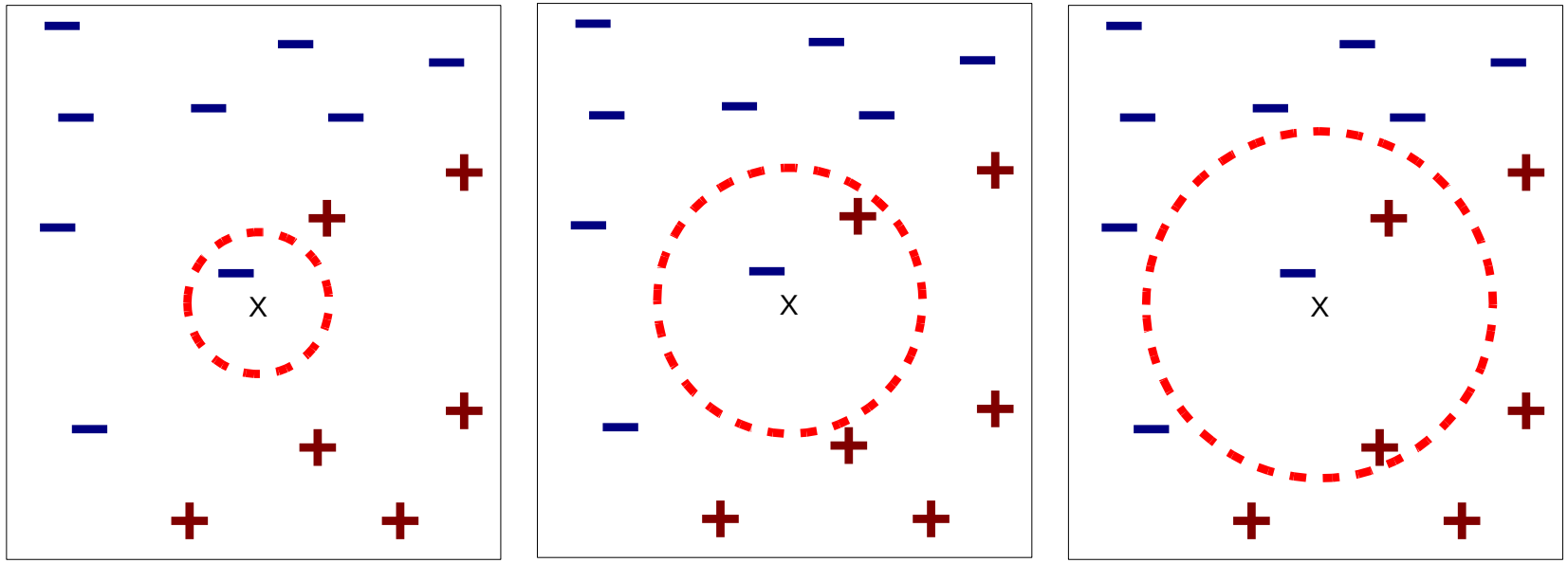


Classification

- Απαιτεί τρία πράγματα
  - Το σύνολο των αποθηκευμένων εγγραφών
  - Μετρική απόστασης για τον υπολογισμό της απόστασης μεταξύ των εγγραφών
  - Η τιμή του k, ο αριθμός των πλησιέστερων γειτόνων προς ανάκτηση
- Για να ταξινομήσετε μια άγνωστη εγγραφή:
  - Υπολογίστε την απόσταση από άλλες εγγραφές εκπαίδευσης
  - Προσδιορίστε k πλησιέστερους γείτονες
  - Χρησιμοποιήστε τις ετικέτες κλάσης των πλησιέστερων γειτόνων για να προσδιορίσετε την ετικέτα κλάσης της άγνωστης εγγραφής (π.χ. εστιάζοντας στην πλειοψηφία)

Ορισμός

- K-πλησιέστεροι γείτονες μιας εγγραφής x είναι σημεία δεδομένων που έχουν την k μικρότερη απόσταση από το x

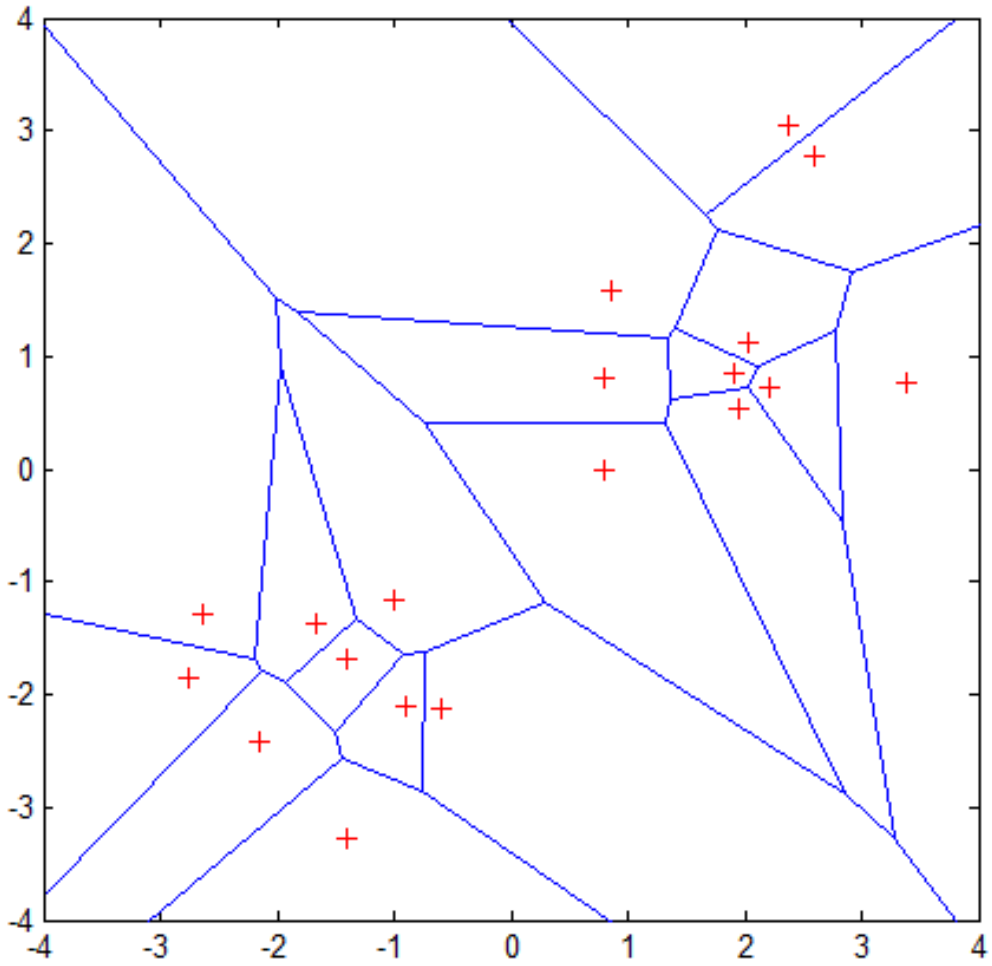


(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

Διάγραμμα Voronoi

Ιδιότητες

- Όλα τα πιθανά σημεία μέσα στο κελί Voronoi ενός δείγματος είναι τα πλησιέστερα γειτονικά σημεία για αυτό το δείγμα
- Για οποιοδήποτε δείγμα, το πλησιέστερο δείγμα προσδιορίζεται από το πλησιέστερο άκρο κελιού Voronoi

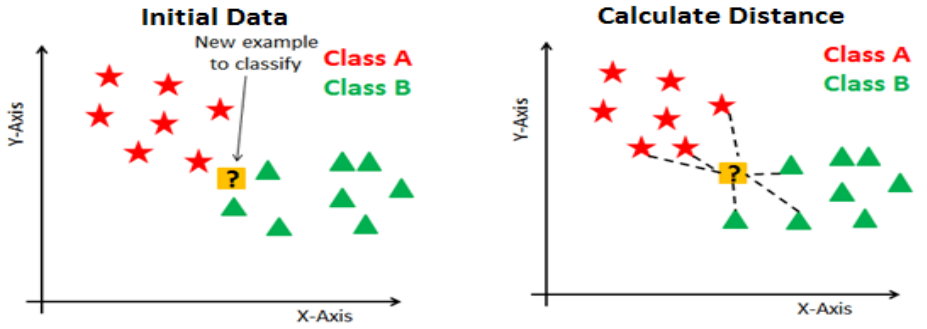


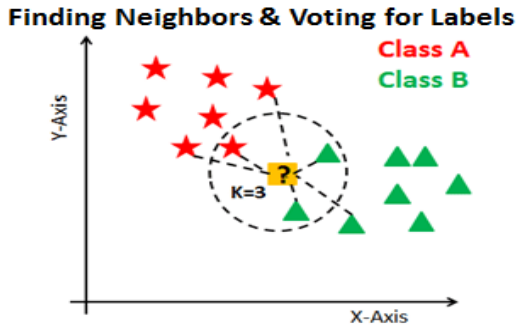
Classification

- Οι ταξινομητές k-NN είναι lazy μοντέλα
- Δεν δημιουργούν ρητά μοντέλα
- Σε αντίθεση με τους eager learners όπως η επαγωγή του δέντρου αποφάσεων και τα συστήματα που βασίζονται σε κανόνες
- Η ταξινόμηση άγνωστων εγγραφών είναι σχετικά ακριβή
- Για κάθε δοκιμή, πρέπει να σαρώσετε όλα τα δεδομένα εκπαίδευσης

Κατηγοριοποίηση

- Υπολογίστε την απόσταση μεταξύ κάθε ζεύγους από τα k ζεύγη σημείων
- Προσδιορίστε την τάξη από τη λίστα πλησιέστερων γειτόνων
  - πάρτε την **πλειοψηφία των ετικετών τάξης μεταξύ των k-πλησιέστερων γειτόνων**
  - Ζυγίστε την ψήφο ανάλογα με την απόσταση συντελεστής βάρους, **$w = 1/d^2$**





Απόσταση

Αντικατάσταση του

$$\hat{f}(q) = \underset{v \in V}{\operatorname{argmax}} \sum_{i=1}^{k} \delta(v, f(x_i))$$

με

$$\hat{f}(q) = \underset{v \in V}{\operatorname{argmax}} \sum_{i=1}^{k} \frac{1}{d(x_i, x_q)^2} \delta(v, f(x_i))$$

Μπορεί να επιλεγούν γενικές Kernel Functions όπως Parzen Windows αντί για την αντίστροφη απόσταση

Απόσταση

Minkowsky:

$$D(\boldsymbol{x},\boldsymbol{y}) = \left(\sum_{i=1}^{m} |x_i - y_i|^r\right)^{1/r}$$

Euclidean:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

Manhattan / city-block:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{m} |x_i - y_i|$$

Camberra:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{m} \frac{|x_i - y_i|}{|x_i + y_i|}$$

Chebychev:

$$D(\boldsymbol{x},\boldsymbol{y}) = \max_{i=1}^{m} |x_i - y_i|$$

Quadratic:

$$D(\boldsymbol{x},\boldsymbol{y}) = (\boldsymbol{x} - \boldsymbol{y})^T Q(\boldsymbol{x} - \boldsymbol{y}) = \sum_{j=1}^{m}\left(\sum_{i=1}^{m}(x_i - y_i)q_{ji}\right)(x_j - y_j)$$

Q is a problem-specific positive definite $m \times m$ weight matrix

Mahalanobis:

$$D(\boldsymbol{x},\boldsymbol{y}) = [\det V]^{1/m}(\boldsymbol{x} - \boldsymbol{y})^{\mathrm{T}} V^{-1}(\boldsymbol{x} - \boldsymbol{y})$$

V is the covariance matrix of $A_1..A_m$, and A_j is the vector of values for attribute j occuring in the training set instances $1..n$.

Correlation:

$$D(\boldsymbol{x},\boldsymbol{y}) = \frac{\sum_{i=1}^{m}(x_i - \overline{x_i})(y_i - \overline{y_i})}{\sqrt{\sum_{i=1}^{m}(x_i - \overline{x_i})^2 \sum_{i=1}^{m}(y_i - \overline{y_i})^2}}$$

$\overline{x_i} = \overline{y_i}$ and is the average value for attribute i occuring in the training set.

Chi-square:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{m} \frac{1}{sum_i}\left(\frac{x_i}{size_{\boldsymbol{x}}} - \frac{y_i}{size_{\boldsymbol{y}}}\right)^2$$

sum_i is the sum of all values for attribute i occuring in the training set, and $size_{\boldsymbol{x}}$ is the sum of all values in the vector $\boldsymbol{x}$.

Kendall's Rank Correlation:

$$D(\boldsymbol{x},\boldsymbol{y}) = 1 - \frac{2}{n(n-1)}\sum_{i=1}^{m}\sum_{j=1}^{i-1}\mathrm{sign}(x_i - x_j)\mathrm{sign}(y_i - y_j)$$

sign(x)=-1, 0 or 1 if $x < 0$, $x = 0$, or $x > 0$, respectively.

Απόσταση

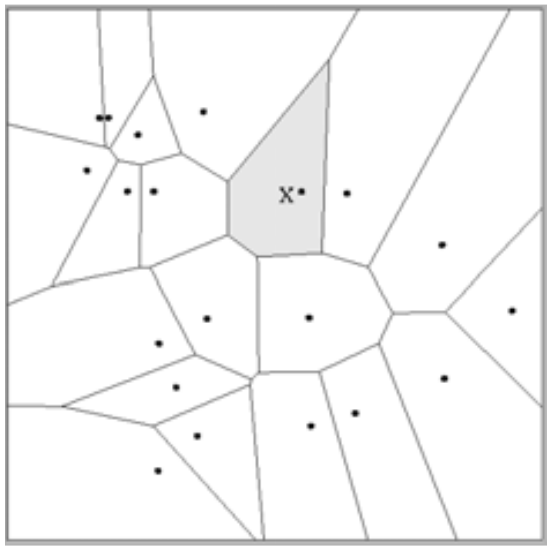
Μετατρέψτε τις τιμές χαρακτηριστικών σε z-scores

$$z_{ij} = \frac{x_{ij} - m_j}{s_j}$$

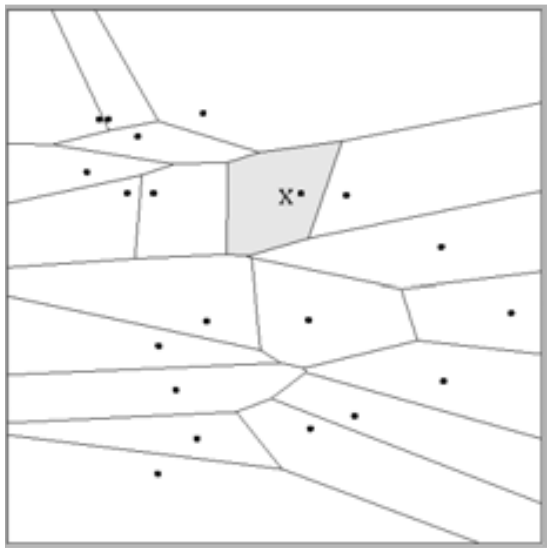
Το εύρος και η κλίμακα των τιμών z θα πρέπει να είναι παρόμοια (με την προϋπόθεση ότι οι κατανομές των μη επεξεργασμένων τιμών χαρακτηριστικών είναι ίδιες)

Απόσταση

Διαφορετικές μετρήσεις μπορούν να αλλάξουν την επιφάνεια απόφασης



$Dist(\mathbf{a},\mathbf{b}) = (a_1 - b_1)^2 + (a_2 - b_2)^2$



$Dist(\mathbf{a},\mathbf{b}) = (a_1 - b_1)^2 + (3a_2 - 3b_2)^2$

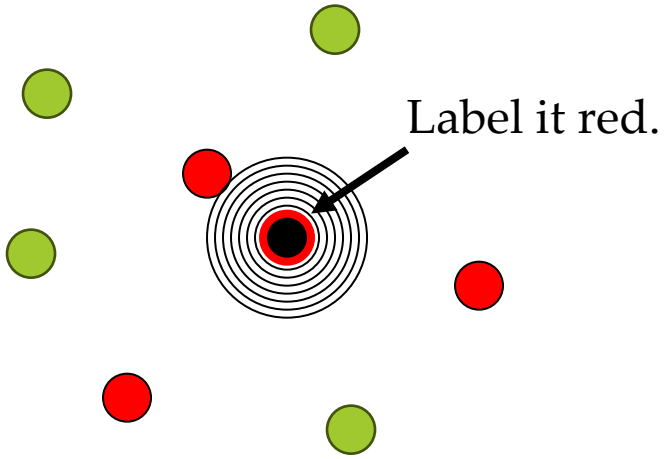
Τυπική Ευκλείδεια μέτρηση απόστασης:

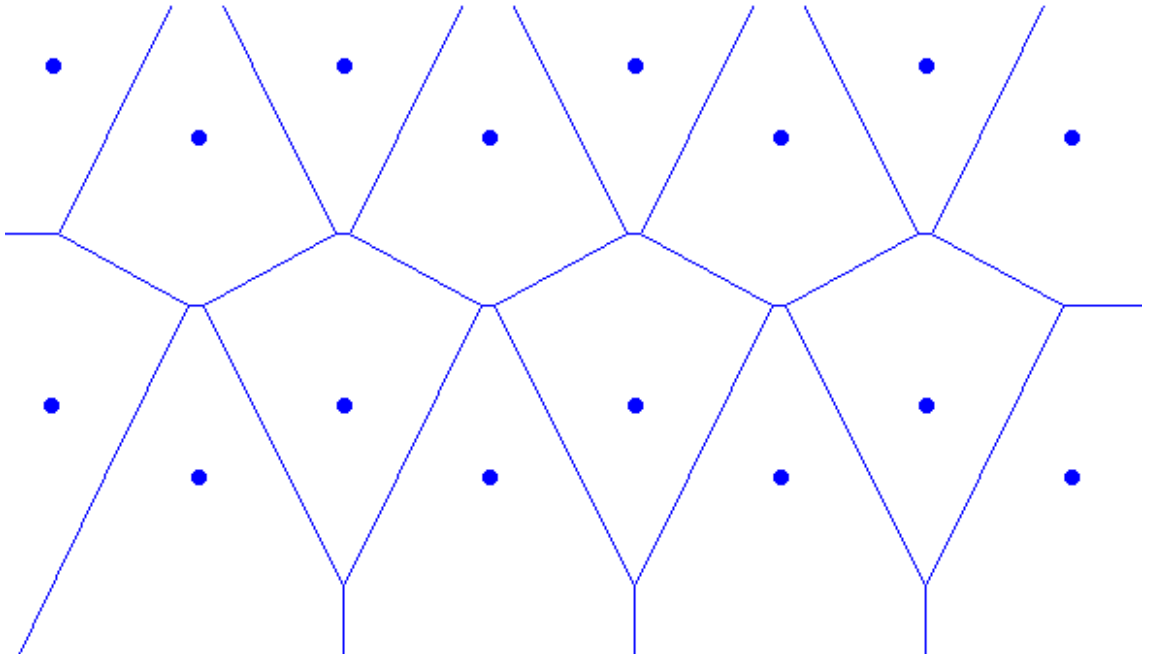
Two-dimensional: $Dist(a,b) = sqrt((a_1 - b_1)^2 + (a_2 - b_2)^2)$

Multivariate: $Dist(a,b) = sqrt(\sum (a_i - b_i)^2)$

1-NN

- Ένας από τους απλούστερους ταξινομητές μηχανικής μάθησης
- Απλή ιδέα: επισημάνετε ένα νέο σημείο όπως το πλησιέστερο γνωστό σημείο





Αλγόριθμος

- Given training data $(x_1, y_1) \ldots (x_n, y_n)$, determine y_{new} for x_{new}
- Find x' most similar to x_{new} using Euclidean distance
- Assign $y_{new} = y'$

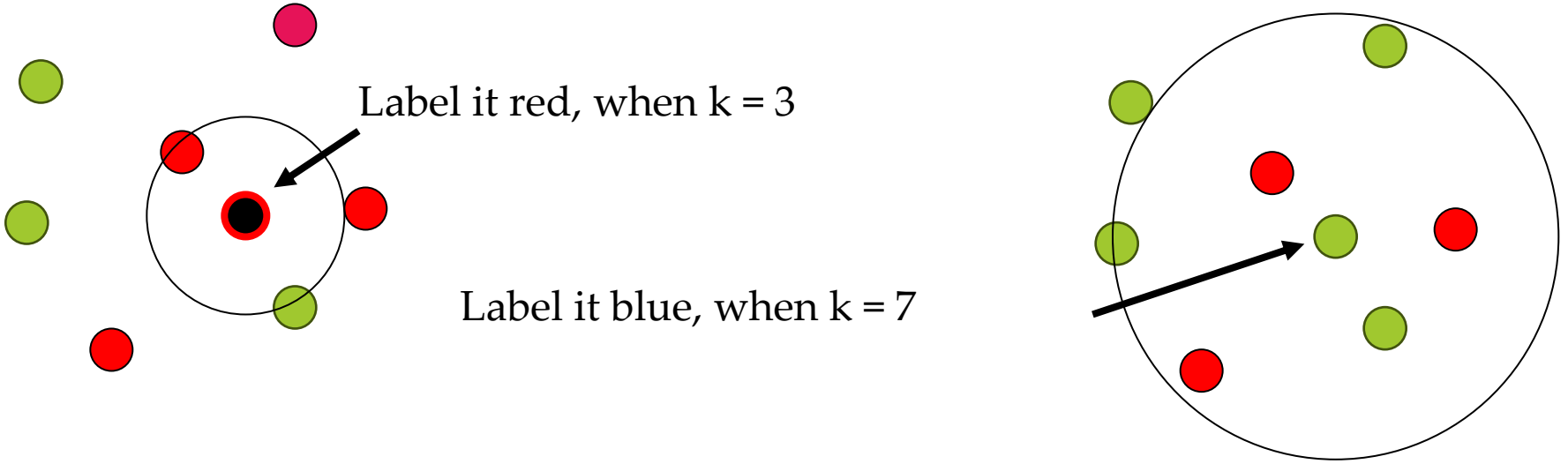
Works for classification or regression

Μειονεκτήματα

- Το 1-NN ταιριάζει ακριβώς στα δεδομένα, συμπεριλαμβανομένου τυχόν θορύβου
- Μπορεί να μην γενικεύεται καλά σε νέα δεδομένα

k-NN

- Γενικεύει το 1-NN για την εξομάλυνση του θορύβου στις ετικέτες
- Σε ένα νέο σημείο εκχωρείται πλέον η πιο συχνή ετικέτα των k πλησιέστερων γειτόνων του



k-NN

- Distance metric: Euclidean
- Number of neighbors to consult: k
- Combining neighbors' outputs:
  - Classification
    - Majority vote
    - Weighted majority vote: nearer have more influence
  - Regression
    - Average (real-valued)
    - Weighted average: nearer have more influence
- Result: Smoother, more generalizable result

$$y_{new} = \text{argmax}_c \sum_k (y_k = c)$$

$$y_{new} = \text{argmax}_c \sum_k w_k (y_k = c)$$

$$y_{new} = \sum_k \frac{1}{k} y_k$$

$$y_{new} = \sum_k w_k y_k$$

k-NN

- Το K είναι μια παράμετρος του αλγορίθμου k-NN
  - Αυτό δεν το κάνει «παραμετρικό».
- Ορίστε τις παραμέτρους χρησιμοποιώντας το σύνολο δεδομένων επικύρωσης
  - Όχι το σετ εκπαίδευσης (overfitting)

Παράδειγμα

Similarity metric: Number of matching attributes (k=2)

New examples:

Example 1 (great, no, no, normal, no)

→ most similar: number 2 (1 mismatch, 4 match) → yes

→Second most similar example: number 1 (2 mismatch, 3 match) → yes

Example 2 (mediocre, yes, no, normal, no)

→ Most similar: number 3 (1 mismatch, 4 match) → no

→Second most similar example: number 1 (2 mismatch, 3 match) → yes

	Food (3)	Chat (2)	Fast (2)	Price (3)	Bar (2)	BigTip
1	great	yes	yes	normal	no	yes
2	great	no	yes	normal	no	yes
3	mediocre	yes	no	high	no	no
4	great	yes	yes	normal	yes	yes

Επιλογή του k

- Εάν το k είναι πολύ μικρό, ευαίσθητο σε σημεία θορύβου
- Εάν το k είναι πολύ μεγάλο, η γειτονιά μπορεί να περιλαμβάνει σημεία από άλλες κλάσεις

- Επιλέξτε k όχι πολύ μεγάλο, αλλά όχι πολύ μικρό (εξαρτάται από τα δεδομένα)

Προβλήματα

- Πρόβλημα με την ευκλείδεια απόσταση:
  - Δεδομένα υψηλών διαστάσεων - κατάρα των διαστάσεων
  - Μπορεί να παράγει αντίθετα διαισθητικά αποτελέσματα
  - Συρρίκνωση της πυκνότητας – αποτέλεσμα αραιότητας

1 1 1 1 1 1 1 1 1 1 1 0		1 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 1 1 1 1	vs	0 0 0 0 0 0 0 0 0 0 0 1
d = 1.4142		d = 1.4142

Προβλήματα

- Η ακρίβεια της πρόβλεψης μπορεί γρήγορα να υποβαθμιστεί όταν αυξάνεται ο αριθμός των χαρακτηριστικών.
  - Άσχετα χαρακτηριστικά «βάζουν» εύκολα πληροφορίες από σχετικά χαρακτηριστικά
  - Όταν είναι πολλά τα άσχετα χαρακτηριστικά, το μέτρο ομοιότητας/απόστασης γίνεται λιγότερο αξιόπιστο
- Θεραπεία
  - Προσπαθήστε να αφαιρέσετε άσχετα χαρακτηριστικά στο βήμα προεπεξεργασίας
  - Το βάρος αποδίδει διαφορετικά
  - Αύξηση k (αλλά όχι πολύ)

Πολυπλοκότητα

- **Expensive**
  - Για να προσδιορίσετε τον πλησιέστερο γείτονα ενός σημείου ερωτήματος q, πρέπει να υπολογίσετε την απόσταση σε όλα τα N παραδείγματα εκπαίδευσης
  - Προ-ταξινόμηση των παραδειγμάτων εκπαίδευσης σε γρήγορες δομές δεδομένων (kd-trees) O(log n)
  - Υπολογίστε μόνο μια κατά προσέγγιση απόσταση (LSH)
  - Κατάργηση περιττών δεδομένων (συμπύκνωση)
- **Απαιτήσεις αποθήκευσης**
  - Πρέπει να αποθηκεύονται όλα τα δεδομένα εκπαίδευσης P
  - Κατάργηση περιττών δεδομένων (συμπύκνωση)
  - Η προδιαλογή συχνά αυξάνει τις απαιτήσεις αποθήκευσης
- **Δεδομένα υψηλών διαστάσεων**
  - «Η κατάρα των διαστάσεων»
  - Η απαιτούμενη ποσότητα δεδομένων εκπαίδευσης αυξάνεται εκθετικά με τη διάσταση
  - Το υπολογιστικό κόστος αυξάνεται επίσης δραματικά
  - Οι τεχνικές διαχωρισμού υποβαθμίζονται σε γραμμική αναζήτηση σε υψηλή διάσταση

Pros & Cons

- Πλεονεκτήματα
  - Το k-NN είναι απλό! (για κατανόηση, εφαρμογή)
  - Συχνά χρησιμοποιείται ως βάση για άλλους αλγόριθμους
  - Η «εκπαίδευση" είναι γρήγορη: απλώς προσθέστε νέο στοιχείο στη βάση δεδομένων
- Μειονεκτήματα
  - Οι περισσότερες εργασίες που γίνονται κατά τη στιγμή του ερωτήματος: μπορεί να είναι ακριβές
  - Πρέπει να αποθηκεύονται δεδομένα O(n) για μεταγενέστερα ερωτήματα
  - Η απόδοση είναι ευαίσθητη στην επιλογή της μέτρησης απόστασης
  - Και κανονικοποίηση των τιμών των χαρακτηριστικών

Παράδειγμα

https://people.revoledu.com/kardi/tutorial/KNN/KNN_Numerical-example.html

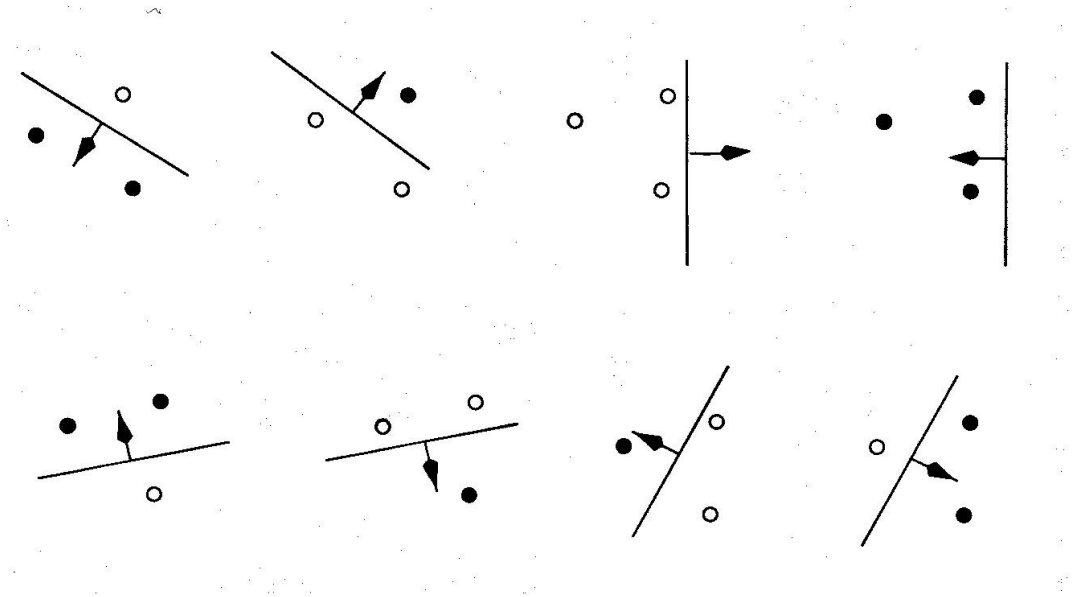
Support Vector Machines

Εισαγωγή

- Ας υποθέσουμε ότι επιλέγουμε n σημεία δεδομένων και αντιστοιχίζουμε ετικέτες + ή – σε αυτά τυχαία
- Εάν η κατηγορία μοντέλου μας (π.χ. ένα νευρωνικό δίκτυο με έναν ορισμένο αριθμό κρυφών νευρώνων/επιπέδων) είναι αρκετά ισχυρή ώστε να μάθει οποιαδήποτε συσχέτιση ετικετών με τα δεδομένα, είναι πολύ ισχυρή!
- Ίσως μπορούμε να χαρακτηρίσουμε τη δύναμη μιας κλάσης μοντέλου ρωτώντας πόσα σημεία δεδομένων μπορεί να «σπάσει», δηλαδή να μάθει τέλεια για όλες τις πιθανές εκχωρήσεις ετικετών
  - Αυτός ο αριθμός των σημείων δεδομένων ονομάζεται διάσταση Vapnik-Chervonenkis
  - Το μοντέλο δεν χρειάζεται να θρυμματίσει όλα τα σύνολα σημείων δεδομένων μεγέθους h. Ένα σύνολο αρκεί.
  - Για επίπεδα σε 3-D, h=4, παρόλο που 4 ομοεπίπεδα σημεία δεν μπορούν να θρυμματιστούν

Παράδειγμα

- Ας υποθέσουμε ότι η κλάση του μοντέλου μας είναι ένα υπερεπίπεδο (hyperplane)
- Στο 2-D, μπορούμε να βρούμε ένα επίπεδο (δηλαδή μια γραμμή) για να αντιμετωπίσουμε οποιαδήποτε επισήμανση τριών σημείων
- Ένα δισδιάστατο υπερεπίπεδο θρυμματίζει 3 σημεία



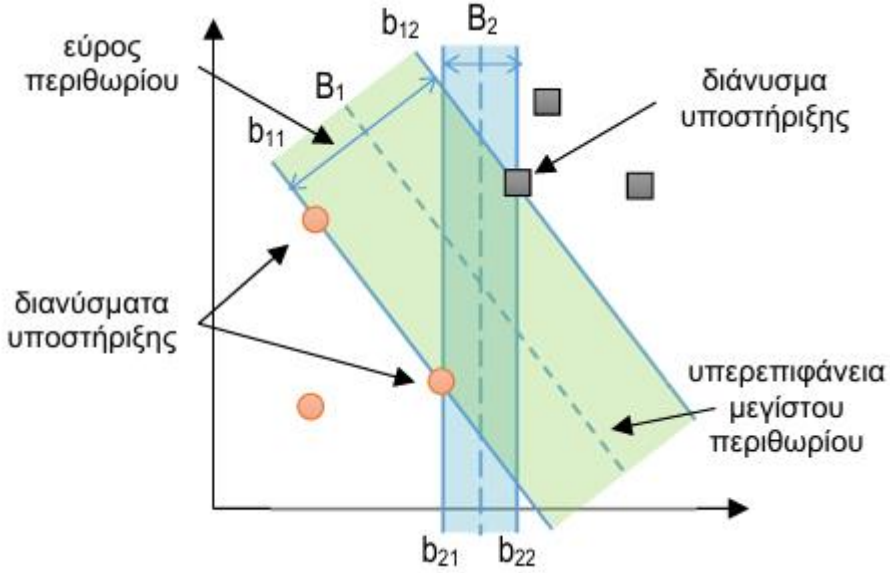
- Αλλά δεν μπορούμε να ασχοληθούμε με ορισμένες από τις πιθανές επισημάνσεις τεσσάρων σημείων

SVM

- Οι μηχανές διανυσμάτων υποστήριξης ή ΜΔΥ (Support Vector Machines, SVM) προτάθηκαν από τον Vladimir Vapnik ως γραμμικοί ταξινομητές, το 1963
- Ενισχύθηκαν από τον ίδιο και τους συνεργάτες του με το κόλπο του πυρήνα (kernel trick), που επέτρεψε τη χρήση τους και σε μη γραμμικώς διαχωρίσιμα προβλήματα
- Οι SVM βασίζονται στη θεωρία στατιστικής μάθησης (statistical learning theory)
- Οι SVM μπήκαν στο mainstream λόγω της εξαιρετικής απόδοσής της στο Handwritten Digit Recognition
    - Ποσοστό σφάλματος 1,1% που ήταν συγκρίσιμο με ένα πολύ προσεκτικά κατασκευασμένο (και πολύπλοκο) τεχνητό νευρωνικό δίκτυο

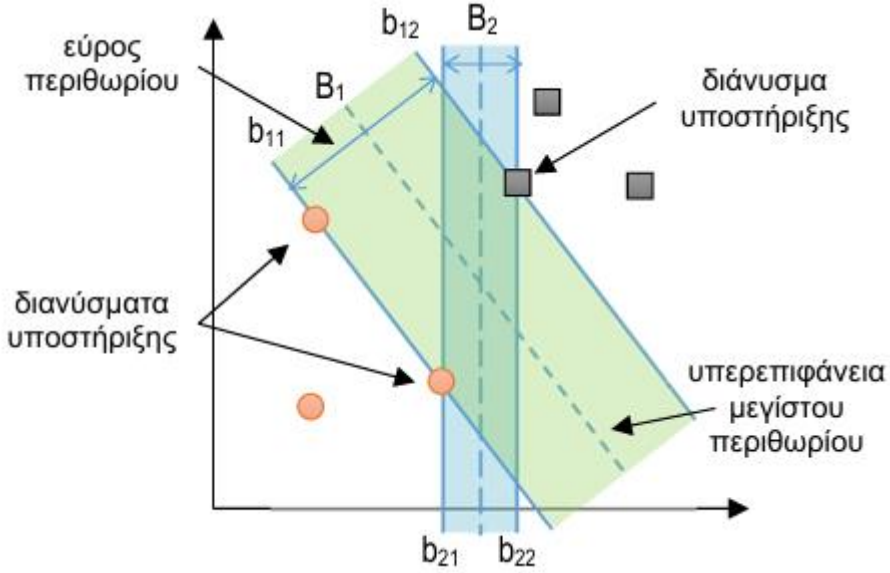
SVM

- Θεωρώντας ένα πρόβλημα δυαδικής ταξινόμησης με θετικά και αρνητικά παραδείγματα, μια επιφάνεια που λειτουργεί ως σύνορο μεταξύ των κλάσεων χαρακτηρίζεται από το γεγονός ότι χωρίζει τα θετικά από τα αρνητικά παραδείγματα
- Τέτοια σύνορα υπάρχουν εν γένει πολλά
- Η γραμμή που μεγιστοποιεί το ελάχιστο περιθώριο είναι ένα καλό στοίχημα
- Αυτό το διαχωριστικό μέγιστου περιθωρίου καθορίζεται από ένα υποσύνολο των σημείων δεδομένων.
- Τα σημεία δεδομένων σε αυτό το υποσύνολο ονομάζονται "διανύσματα υποστήριξης«
- Θα είναι χρήσιμο υπολογιστικά εάν μόνο ένα μικρό κλάσμα των σημείων δεδομένων είναι διανύσματα υποστήριξης, επειδή χρησιμοποιούμε τα διανύσματα υποστήριξης για να αποφασίσουμε σε ποια πλευρά του διαχωριστή βρίσκεται μια δοκιμαστική περίπτωση



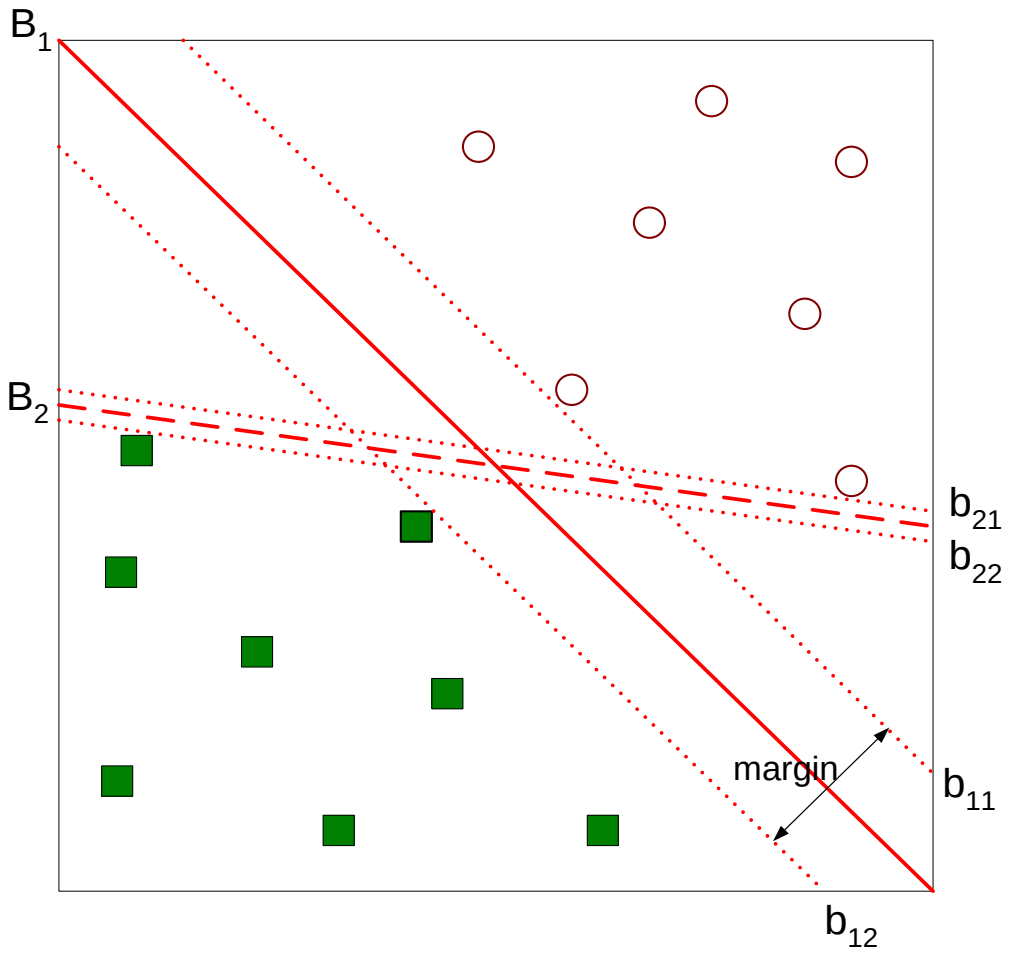
SVM

- Η μέθοδος SVM επιδιώκει να βρει το σύνορο που απέχει όσο το δυνατόν περισσότερο από τα παραδείγματα των κλάσεων που διαχωρίζει
- Η υπερεπιφάνεια αυτή ονομάζεται υπερεπιφάνεια μεγίστου περιθωρίου (maximum margin hypersurface) και σε γραμμικώς διαχωρίσιμα προβλήματα ορίζεται από τα διανύσματα υποστήριξης (support vectors).
- Επιπλέον, μέσω των συναρτήσεων πυρήνα (kernel functions), οι SVM μπορούν να μετασχηματίσουν τον αρχικό χώρο υποθέσεων έτσι ώστε μη-γραμμικώς διαχωρίσιμα προβλήματα να μετατραπούν σε γραμμικώς διαχωρίσιμα και τελικά να λυθούν με την ίδια μεθοδολογία.



SVM

- Εύρεση υπερεπιπέδου που μεγιστοποιεί το περιθώριο => Το B1 είναι καλύτερο από το B2

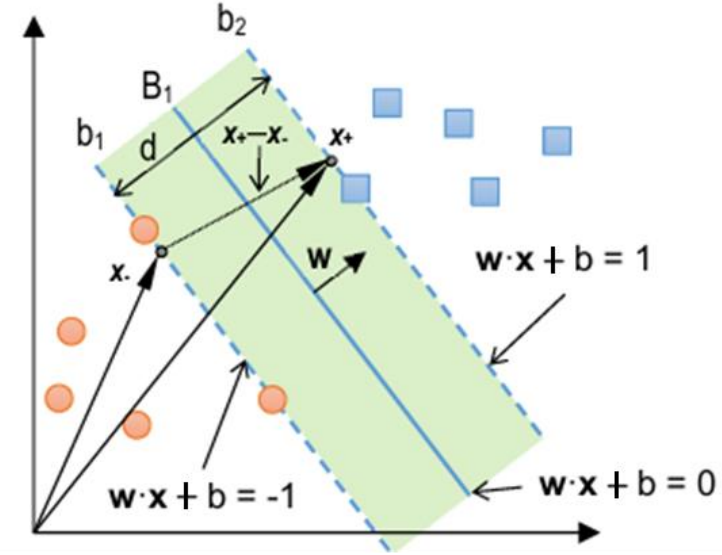


SVM

- Θεωρούμε ένα σύνολο από πολυδιάστατα σημεία x (x_1, x_2, …, x_n) για τα οποία γνωρίζουμε ότι ανήκουν στην κλάση (+) (τετράγωνα) ή στην κλάση (–) (κύκλοι), τις οποίες έστω ότι τις κωδικοποιούμε αριθμητικά με 1 και –1 αντίστοιχα
- Σύνορο: w1x1 + w2x2+b=0 ή **w**·x+b=0
- Εφαρμόζοντας τη σχέση αυτή για δύο σημεία α και β που βρίσκονται πάνω στο σύνορο και αφαιρώντας

$$\mathbf{w}\cdot(x_\alpha - x_\beta)=0$$

- Το διάνυσμα των βαρών είναι κάθετο στο ζητούμενο σύνορο
- Σε γραμμικώς διαχωρίσιμα προβλήματα, μπορούμε να ορίσουμε δύο επιπλέον σύνορα εκατέρωθεν του w·x+b=0 που επίσης χωρίζουν τις δύο κλάσεις και απέχουν όσο γίνεται περισσότερο μεταξύ τους



SVM

- Για ένα σημείο z στο χώρο του προβλήματος, η κλάση y του σημείου θα είναι

$$y = \begin{cases} 1 & \alpha\nu\ \boldsymbol{w}\cdot\boldsymbol{z}+b>0 \\ -1 & \alpha\nu\ \boldsymbol{w}\cdot\boldsymbol{z}+b<0 \end{cases}$$

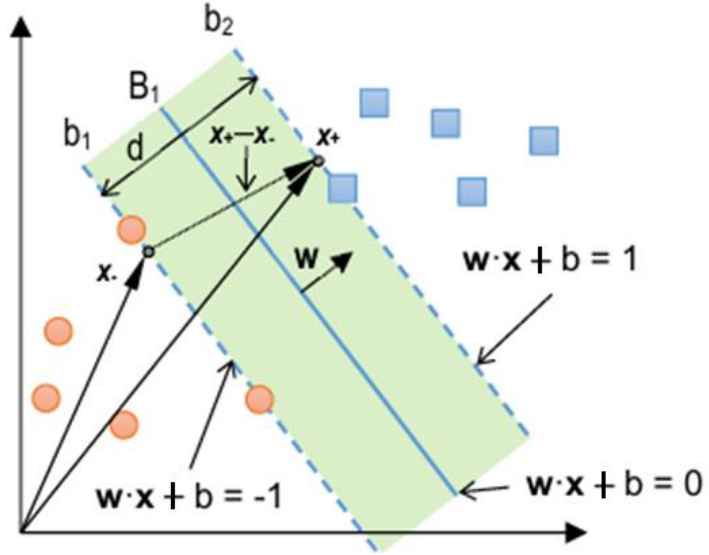
- Έστω x_+ και x_- είναι σημεία στα πάνω και κάτω όρια του περιθωρίου
- Καθώς το διάνυσμα w είναι κάθετο στο ζητούμενο σύνορο, διαιρώντας το με το μήκος του ||w|| προκύπτει ένα μοναδιαίο διάνυσμα που πολλαπλασιαζόμενο (εσωτερικό γινόμενο) με το διάνυσμα της διαφοράς (x_+ - x_-), δίνει το ζητούμενο d

$$(\boldsymbol{x}_+ - \boldsymbol{x}_-)\cdot\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|} = d$$

- Σε αυτή τη σχέση, για το $x_+\cdot w$, η σχέση υπολογισμού του y δίνει 1-b ενώ για το $x_-\cdot w$ δίνει -1-b. Αντικαθιστώντας

$$d = \frac{2}{\|\boldsymbol{w}\|}$$

$$\min_{\boldsymbol{w}} \frac{\|\boldsymbol{w}\|^2}{2}$$



$$\|\mathbf{x}\| := \sqrt{x_1^2 + \cdots + x_n^2}.$$

SVM

- Το όριο απόφασης πρέπει να ταξινομεί σωστά όλα τα σημεία ➔ Ένα περιορισμένο πρόβλημα βελτιστοποίησης

$$m = \frac{2}{||\mathbf{w}||}$$

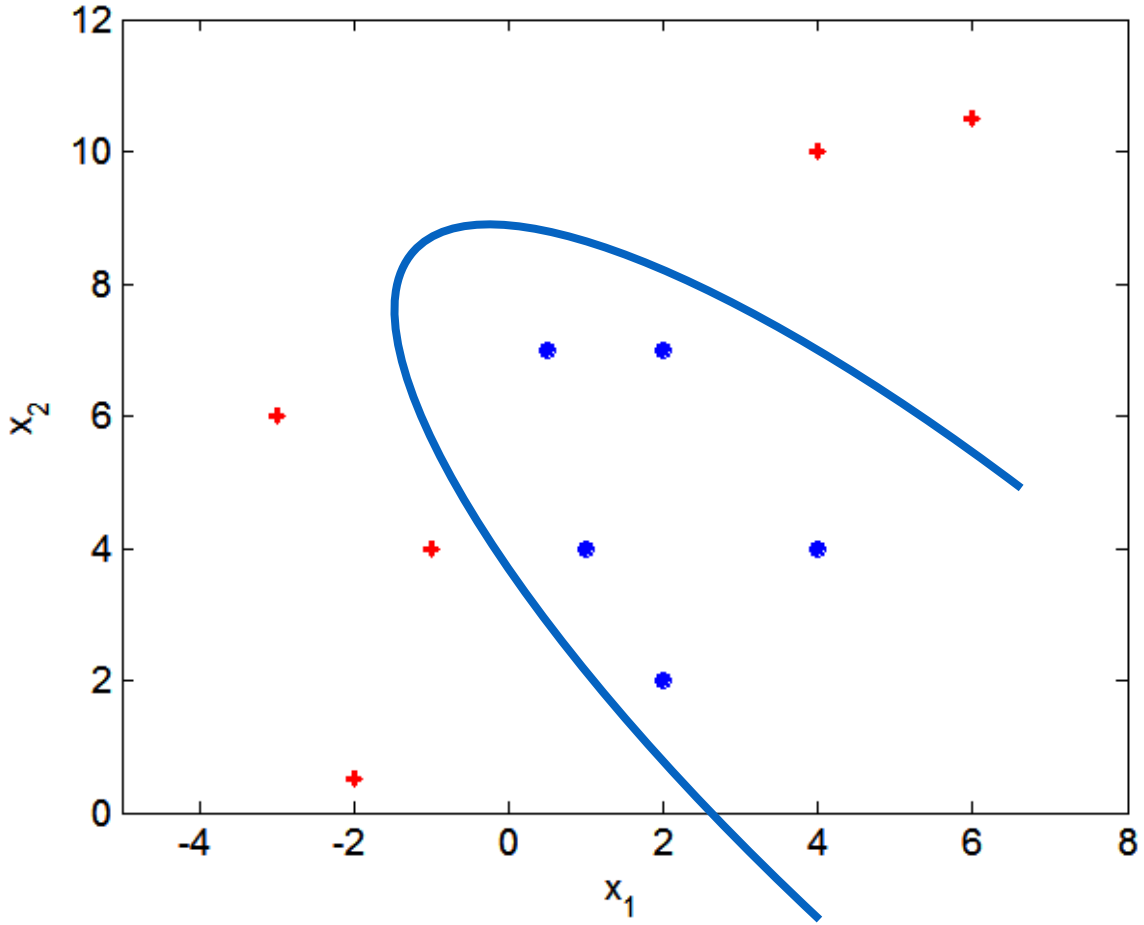
$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, \qquad \forall i$$

$$\text{Minimize } \frac{1}{2}||\mathbf{w}||^2 \qquad ||\mathbf{w}||^2 = \mathbf{w}^T\mathbf{w}$$

$$\text{subject to } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \qquad \forall i$$

SVM

- Τι γίνεται αν το όριο απόφασης δεν είναι γραμμικό;



SVM

- Εισαγωγή μεταβλητών slack
- Ανάγκη ελαχιστοποίησης:

$$L(w) = \frac{\| \vec{w} \|^2}{2} + C\left(\sum_{i=1}^{N} \xi_i^k \right)$$

δεδομένου ότι

$$\vec{w} \cdot \vec{x_i} + b \geq 1 - \xi_i \quad \text{if } y_i = 1$$
$$\vec{w} \cdot \vec{x_i} + b \leq -1 + \xi_i \quad \text{if } y_i = -1$$

SVM

https://www.youtube.com/watch?v=OdlNM96sHio

Clustering

Εισαγωγή

- Cluster - Συστάδα: Μια συλλογή αντικειμένων δεδομένων
  - παρόμοια (ή σχετιζόμενα) μεταξύ τους εντός της ίδιας ομάδας
  - ανόμοια (ή άσχετα) με τα αντικείμενα άλλων ομάδων
- Ανάλυση συστάδων (ή ομαδοποίηση, τμηματοποίηση δεδομένων,…)
  - Εύρεση ομοιοτήτων μεταξύ δεδομένων σύμφωνα με τα χαρακτηριστικά που βρίσκονται στα δεδομένα και ομαδοποίηση παρόμοιων αντικειμένων δεδομένων σε συστάδες
  - Εκμάθηση χωρίς επίβλεψη: χωρίς προκαθορισμένες τάξεις (δηλαδή, μάθηση με παρατηρήσεις έναντι μάθησης με παραδείγματα: υπό επίβλεψη)
- Τυπικές εφαρμογές
  - Ως αυτόνομο εργαλείο για να αποκτήσετε πληροφορίες σχετικά με την κατανομή των δεδομένων
  - Ως βήμα προεπεξεργασίας για άλλους αλγόριθμους

Εισαγωγή

- Biology: taxonomy of living things: kingdom, phylum, class, order, family, genus and species
- Information retrieval: document clustering
- Land use: Identification of areas of similar land use in an earth observation database
- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- City-planning: Identifying groups of houses according to their house type, value, and geographical location
- Earth-quake studies: Observed earth quake epicenters should be clustered along continent faults
- Climate: understanding earth climate, find patterns of atmospheric and ocean
- Economic Science: market research

Εισαγωγή

- Summarization:
- Preprocessing for regression, PCA, classification, and association analysis
- Compression:
- Image processing: vector quantization
- Finding K-nearest Neighbors
- Localizing search to one or a small number of clusters
- Outlier detection
- Outliers are often viewed as those “far away” from any cluster

Ποιότητα

- Μια καλή μέθοδος ομαδοποίησης θα παράγει συστάδες υψηλής ποιότητας
  - υψηλή ενδοταξική ομοιότητα (intra-cluster): συνεκτική εντός συστάδων
  - χαμηλή διαταξική ομοιότητα (inter-cluster): διακριτική μεταξύ συστάδων
- Η ποιότητα μιας μεθόδου ομαδοποίησης εξαρτάται από
  - το μέτρο ομοιότητας που χρησιμοποιείται από τη μέθοδο
  - την εφαρμογή του και
  - η ικανότητά του να ανακαλύπτει μερικά ή όλα τα κρυμμένα μοτίβα

Ποιότητα

- Μετρική ανομοιότητας/ομοιότητας
  - Η ομοιότητα εκφράζεται ως συνάρτηση απόστασης, τυπικά μετρική: d(i, j)
  - Οι ορισμοί των συναρτήσεων απόστασης είναι συνήθως μάλλον διαφορετικοί για μεταβλητές με κλίμακα διαστήματος, boolean, κατηγορικές, τακτικές αναλογίες και διανυσματικές μεταβλητές
  - Τα βάρη πρέπει να συσχετίζονται με διαφορετικές μεταβλητές με βάση τις εφαρμογές και τη σημασιολογία δεδομένων
- Ποιότητα ομαδοποίησης:
  - Συνήθως υπάρχει μια ξεχωριστή συνάρτηση «ποιότητας» που μετρά το πόσο καλή είναι μια συστάδα
  - Είναι δύσκολο να ορίσουμε το «αρκετά παρόμοιο» ή «αρκετά καλό»
  - Η απάντηση είναι συνήθως άκρως υποκειμενική

Σημεία

- Κριτήρια κατάτμησης
  - Μονό επίπεδο έναντι ιεραρχικής κατάτμησης (συχνά, η ιεραρχική κατάτμηση πολλαπλών επιπέδων είναι επιθυμητή)
- Διαχωρισμός συστάδων
  - Αποκλειστική (π.χ. ένας πελάτης ανήκει σε μία μόνο περιοχή) έναντι μη αποκλειστικών (π.χ. ένα έγγραφο μπορεί να ανήκει σε περισσότερες από μία κατηγορίες)
- Μέτρο ομοιότητας
  - Βάσει απόστασης (π.χ. Ευκλείδεια, οδικό δίκτυο, διάνυσμα) έναντι συνδεσιμότητας (π.χ. πυκνότητα ή γειτνίαση)
- Ομαδοποίηση χώρου
  - Πλήρης χώρος (συχνά όταν είναι χαμηλής διάστασης) έναντι υποχώρων (συχνά σε ομαδοποίηση υψηλών διαστάσεων)

Σημεία

- Επεκτασιμότητα
  - Ομαδοποίηση όλων των δεδομένων αντί μόνο σε δείγματα
- Ικανότητα αντιμετώπισης διαφορετικών τύπων ιδιοτήτων
  - Αριθμητικά, δυαδικά, κατηγορικά, τακτικά, συνδεδεμένα και μείγμα αυτών
- Ομαδοποίηση με βάση περιορισμούς
  - Ο χρήστης μπορεί να δώσει στοιχεία για περιορισμούς
  - Χρησιμοποιήστε τις γνώσεις τομέα για να προσδιορίσετε τις παραμέτρους εισόδου
- Ερμηνευσιμότητα και χρηστικότητα
- Άλλοι
  - Ανακάλυψη συστάδων με αυθαίρετο σχήμα
  - Δυνατότητα αντιμετώπισης θορυβωδών δεδομένων
  - Αυξητική ομαδοποίηση και έλλειψη ευαισθησίας στη σειρά εισόδου
  - Υψηλός αριθμός διαστάσεων

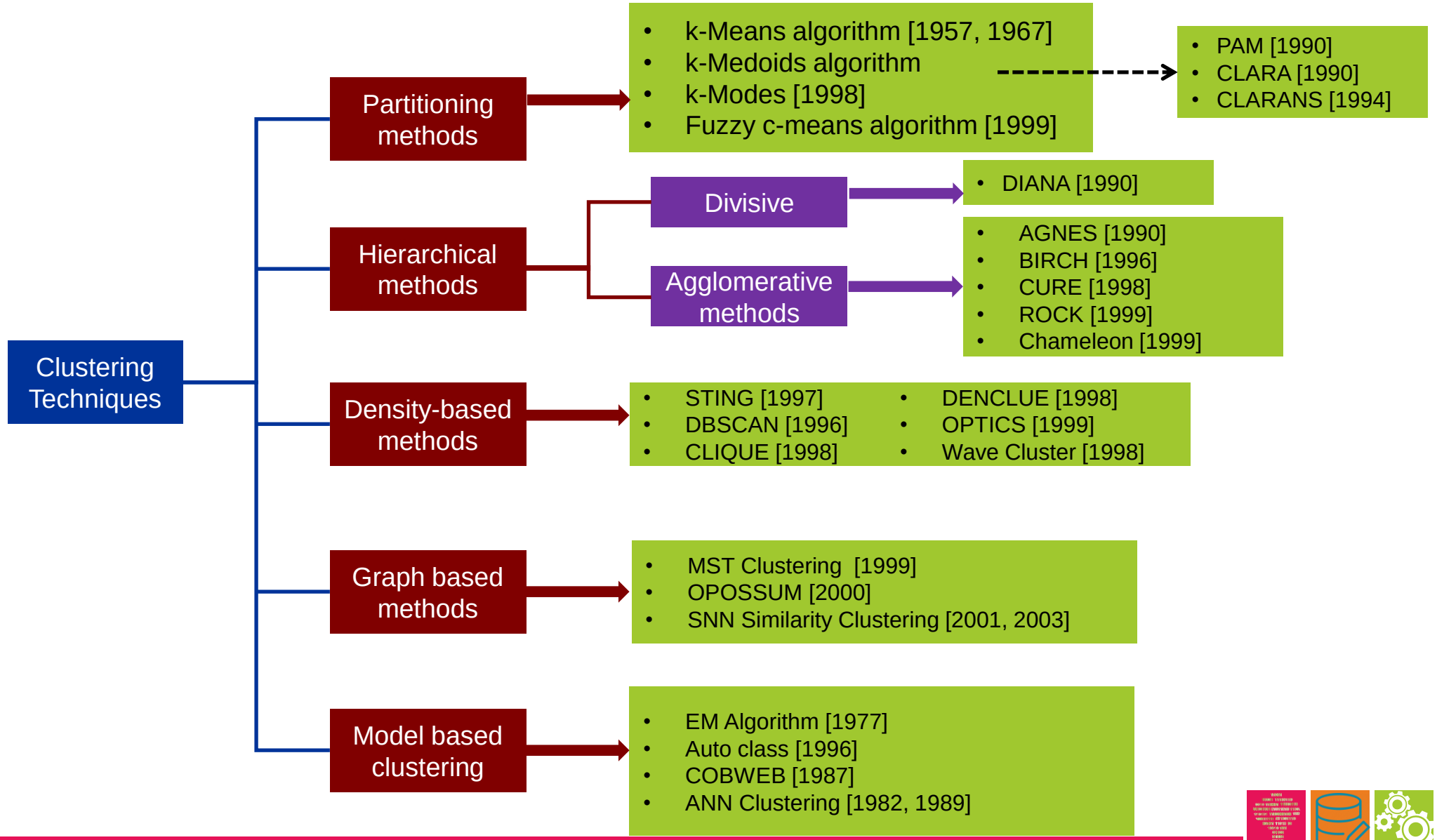
Κατηγορίες

- Προσέγγιση κατάτμησης:
  - Κατασκευάστε διάφορα τμήματα και στη συνέχεια αξιολογήστε τα με κάποιο κριτήριο, π.χ. ελαχιστοποιώντας το άθροισμα των τετραγωνικών σφαλμάτων
  - Τυπικές μέθοδοι: k-means, k-medoids, CLARANS
- Ιεραρχική προσέγγιση:
  - Δημιουργήστε μια ιεραρχική αποσύνθεση του συνόλου δεδομένων (ή αντικειμένων) χρησιμοποιώντας κάποιο κριτήριο
  - Τυπικές μέθοδοι: Diana, Agnes, BIRCH, CHAMELEON
- Προσέγγιση με βάση την πυκνότητα:
  - Με βάση τις λειτουργίες συνδεσιμότητας και πυκνότητας
  - Τυπικές μέθοδοι: DBSCAN, OPTICS, DenClue
- Προσέγγιση βάσει πλέγματος:
  - βασίζεται σε μια δομή κοκκοποίησης πολλαπλών επιπέδων
  - Τυπικές μέθοδοι: STING, WaveCluster, CLIQUE

Κατηγορίες

- Βάσει μοντέλου:
  - Υποτίθεται ένα μοντέλο για κάθε ένα από τα συμπλέγματα και προσπαθεί να βρει την καλύτερη προσαρμογή αυτού του μοντέλου μεταξύ τους
  - Τυπικές μέθοδοι: EM, SOM, COBWEB
- Συχνά βασισμένα σε μοτίβα:
  - Με βάση την ανάλυση συχνών προτύπων
  - Τυπικές μέθοδοι: p-Cluster
- Καθοδηγούμενη από το χρήστη ή βάσει περιορισμών:
  - Ομαδοποίηση λαμβάνοντας υπόψη περιορισμούς που καθορίζονται από το χρήστη ή για συγκεκριμένες εφαρμογές
  - Τυπικές μέθοδοι: COD (εμπόδια), περιορισμένη ομαδοποίηση
- Ομαδοποίηση βάσει συνδέσμων:
  - Τα αντικείμενα συχνά συνδέονται μεταξύ τους με διάφορους τρόπους
  - Μπορούν να χρησιμοποιηθούν μαζικοί σύνδεσμοι για τη ομαδοποίηση : SimRank, LinkClus

Κατηγορίες


Clustering Techniques
Partitioning methods
k-Means algorithm [1957, 1967]
k-Medoids algorithm
k-Modes [1998]
Fuzzy c-means algorithm [1999]
PAM [1990]
CLARA [1990]
CLARANS [1994]
Hierarchical methods
Divisive
DIANA [1990]
Agglomerative methods
AGNES [1990]
BIRCH [1996]
CURE [1998]
ROCK [1999]
Chameleon [1999]
Density-based methods
STING [1997]
DBSCAN [1996]
CLIQUE [1998]
DENCLUE [1998]
OPTICS [1999]
Wave Cluster [1998]
Graph based methods
MST Clustering [1999]
OPOSSUM [2000]
SNN Similarity Clustering [2001, 2003]
Model based clustering
EM Algorithm [1977]
Auto class [1996]
COBWEB [1987]
ANN Clustering [1982, 1989]

Πρόβλημα

- Μέθοδος κατάτμησης: Διαμερισμός μιας βάσης δεδομένων D με n αντικείμενα σε ένα σύνολο k συστάδων, έτσι ώστε το άθροισμα των τετραγωνικών αποστάσεων να ελαχιστοποιείται (όπου ci είναι το κέντρο - centroid ή το μέσο - medioid της ομάδας Ci)

$$E = \Sigma_{i=1}^{k} \Sigma_{p \in C_i} (p - c_i)^2$$

- Με δεδομένο το k, βρείτε ένα διαμέρισμα k συστάδων που βελτιστοποιεί το επιλεγμένο κριτήριο κατάτμησης
  - Καθολική βέλτιστη: απαριθμήστε εξαντλητικά όλα τα τμήματα
  - Ευρετικές μέθοδοι: αλγόριθμοι k-means και k-medoids
  - k-means (MacQueen'67, Lloyd'57/'82): Κάθε συστάδα αντιπροσωπεύεται από το κέντρο της
  - k-medoids ή PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Κάθε συστάδα αντιπροσωπεύεται από ένα από τα αντικείμενα της ομάδας

k-Means

- Δεδομένου ενός συνόλου n διακριτών αντικειμένων, ο αλγόριθμος ομαδοποίησης k-Means χωρίζει τα αντικείμενα σε k αριθμό συστάδων έτσι ώστε η ομοιότητα εντός συστάδων είναι υψηλή αλλά η ομοιότητα μεταξύ συστάδων είναι χαμηλή
- Ο χρήστης πρέπει να καθορίσει το k, τον αριθμό των συστάδων και να εξετάσει ότι τα αντικείμενα ορίζονται με αριθμητικά χαρακτηριστικά και επομένως να χρησιμοποιήσει οποιαδήποτε από τις μετρικές απόστασης για να οριοθετήσει τα συμπλέγματα

k-Means

Algorithm k-Means clustering

Input: D is a dataset containing *n* objects, *k* is the number of cluster
Output: A set of *k* clusters
Steps:

1. Randomly choose *k* objects from D as the initial cluster centroids.

2. **For** each of the objects in D **do**
   - Compute distance between the current objects and *k* cluster centroids
   - Assign the current object to that cluster to which it is closest.

3. Compute the "cluster centers" of each cluster. These become the new cluster centroids.

4. Repeat step 2-3 until the convergence criterion is satisfied

5. Stop

k-Means

1) Objects are defined in terms of set of attributes. $A = \{A_1, A_2, \ldots\ldots, A_m\}$ where each A_i is continuous data type.

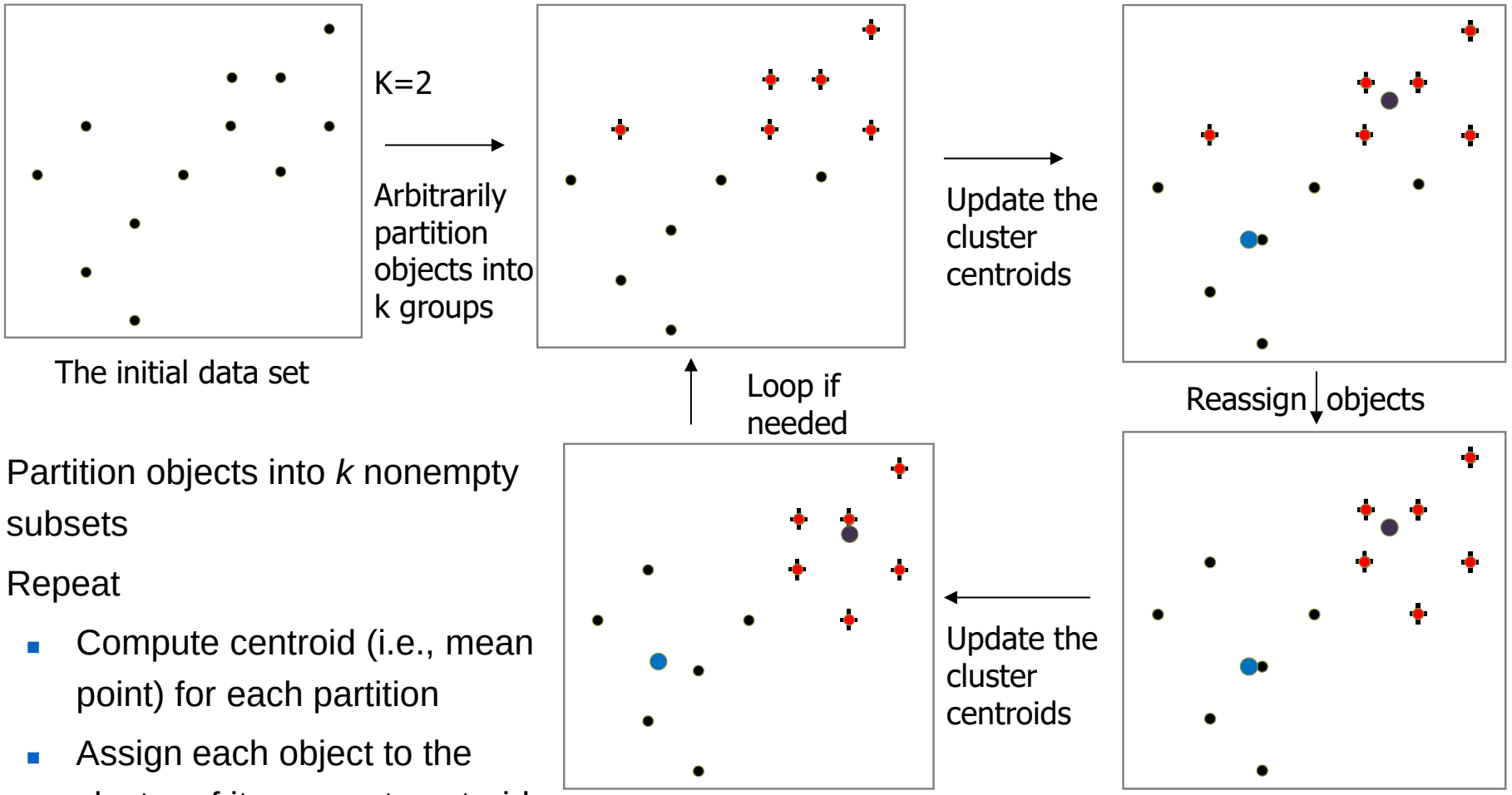
2) Distance computation: Any distance such as L_1, L_2, L_3 or cosine similarity.

3) Minimum distance is the measure of closeness between an object and centroid.

4) Mean Calculation: It is the mean value of each attribute values of all objects.

5) Convergence criteria: Any one of the following are termination condition of the algorithm.
   - Number of maximum iteration permissible.
   - No change of centroid values in any cluster.
   - Zero (or no significant) movement(s) of object from one cluster to another.
   - Cluster quality reaches to a certain level of acceptance.

k-Means



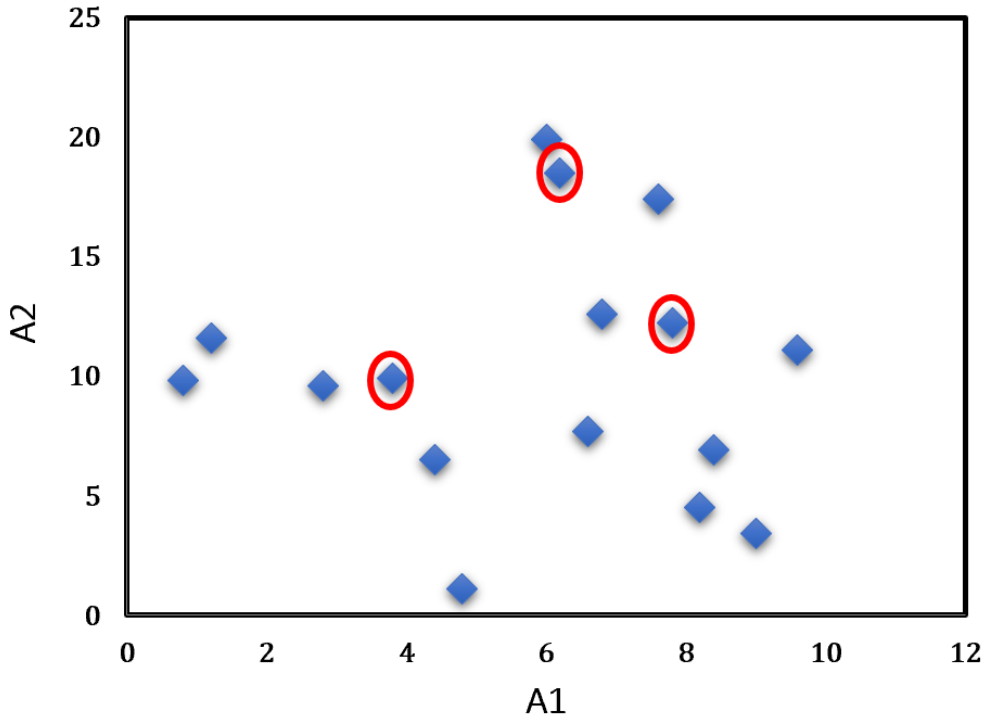
- Partition objects into *k* nonempty subsets
- Repeat
  - Compute centroid (i.e., mean point) for each partition
  - Assign each object to the cluster of its nearest centroid
- Until no change

k-Means

Table : objects with two attributes A_1 and A_2.

A_1	A_2
6.8	12.6
0.8	9.8
1.2	11.6
2.8	9.6
3.8	9.9
4.4	6.5
4.8	1.1
6.0	19.9
6.2	18.5
7.6	17.4
7.8	12.2
6.6	7.7
8.2	4.5
8.4	6.9
9.0	3.4
9.6	11.1

Fig: Plotting data of Table



k-Means

- Ας υποθέσουμε, k=3. Τρία αντικείμενα επιλέγονται τυχαία που φαίνονται κυκλωμένα (βλ. προηγούμενο Σχήμα). Αυτά τα τρία κεντροειδή φαίνονται παρακάτω.

Τα αρχικά Centroids επιλέχθηκαν τυχαία

Centroid	Objects	
	A1	A2
c_1	3.8	9.9
c_2	7.8	12.2
c_3	6.2	18.5

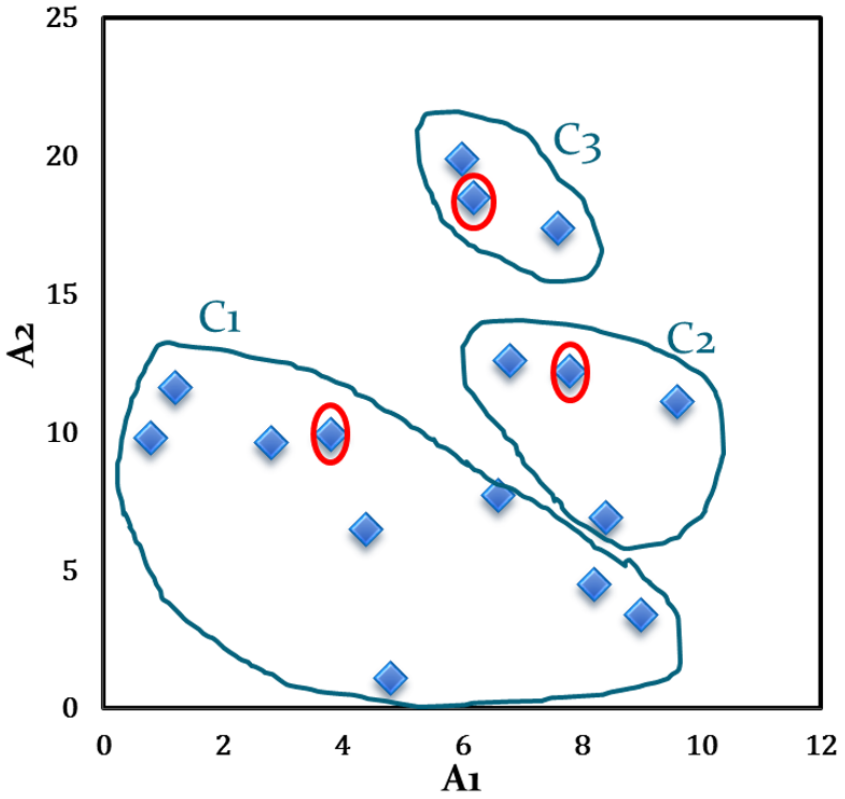
- Ας θεωρήσουμε το μέτρο της Ευκλείδειας απόστασης (L2 Norm) ως τη μέτρηση της απόστασης στην εικόνα μας.
- Έστω d1, d2 και d3 δηλώνουν την απόσταση από ένα αντικείμενο στα c1, c2 και c3 αντίστοιχα. Οι υπολογισμοί της απόστασης φαίνονται στον Πίνακα.
- Η αντιστοίχιση κάθε αντικειμένου στο αντίστοιχο κέντρο εμφανίζεται στη δεξιά στήλη και η ομαδοποίηση που προκύπτει εμφανίζεται στο επόμενο Σχήμα.

k-Means

Table: Distance calculation

A_1	A_2	d_1	d_2	d_3	cluster
6.8	12.6	4.0	1.1	5.9	2
0.8	9.8	3.0	7.4	10.2	1
1.2	11.6	3.1	6.6	8.5	1
2.8	9.6	1.0	5.6	9.5	1
3.8	9.9	0.0	4.6	8.9	1
4.4	6.5	3.5	6.6	12.1	1
4.8	1.1	8.9	11.5	17.5	1
6.0	19.9	10.2	7.9	1.4	3
6.2	18.5	8.9	6.5	0.0	3
7.6	17.4	8.4	5.2	1.8	3
7.8	12.2	4.6	0.0	6.5	2
6.6	7.7	3.6	4.7	10.8	1
8.2	4.5	7.0	7.7	14.1	1
8.4	6.9	5.5	5.3	11.8	2
9.0	3.4	8.3	8.9	15.4	1
9.6	11.1	5.9	2.1	8.1	2

Fig: Initial cluster with respect to Table



k-Means

- Ο υπολογισμός των νέων κεντροειδών των τριών συστάδων με χρήση του μέσου όρου των τιμών των χαρακτηριστικών των A1 και A2 φαίνεται στον παρακάτω Πίνακα. Το σύμπλεγμα με νέα κεντροειδή φαίνεται στο Σχ.

Calculation of new centroids

New Centroid	Objects	
	A1	A2
c_1	4.6	7.1
c_2	8.2	10.7
c_3	6.6	18.6

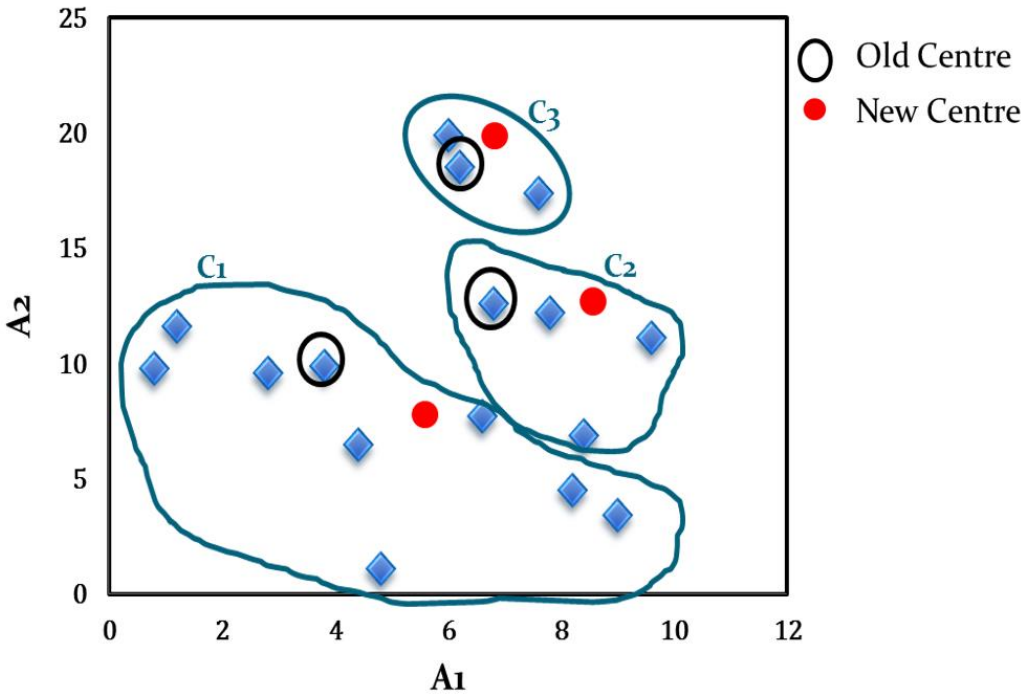


Fig: Initial cluster with new centroids

k-Means

- Στη συνέχεια, εκχωρούμε εκ νέου τα 16 αντικείμενα σε τρία συμπλέγματα, προσδιορίζοντας ποιο κέντρο βρίσκεται πιο κοντά σε κάθε ένα. Αυτό δίνει το αναθεωρημένο σύνολο συμπλεγμάτων που φαίνεται στο Σχ.
- Σημειώστε ότι το σημείο p μετακινείται από το σύμπλεγμα C2 στο σύμπλεγμα C1.

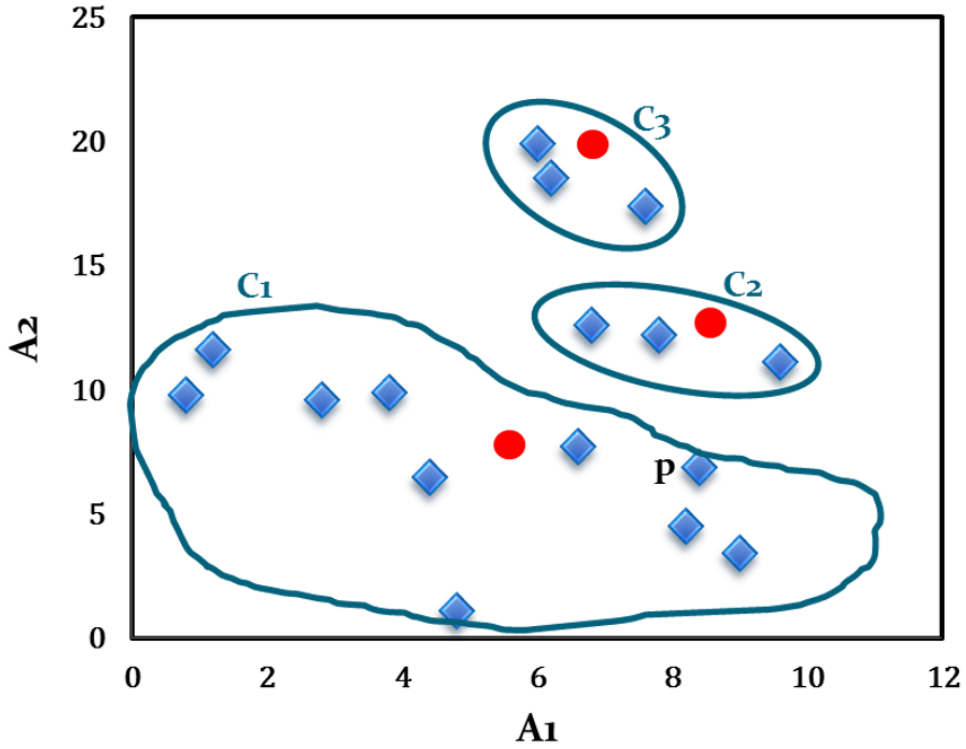


Fig : Cluster after first iteration

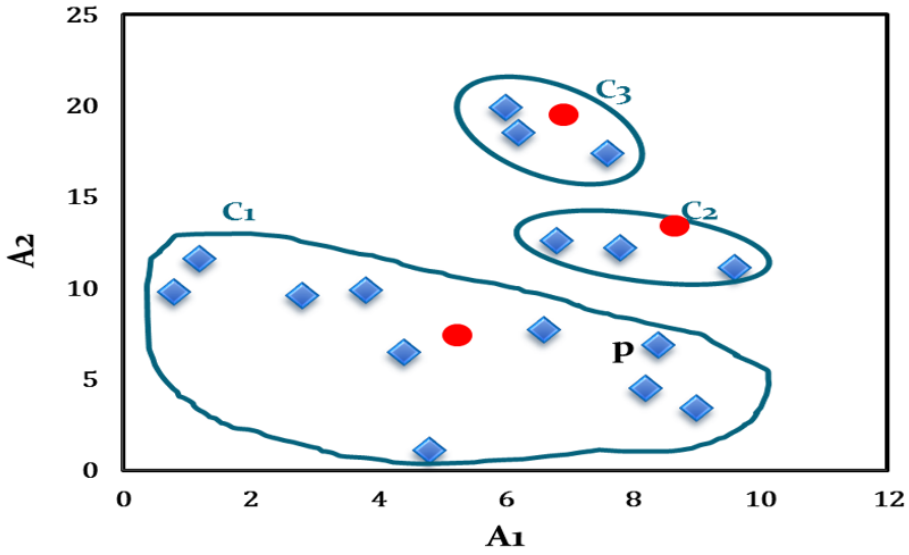
k-Means

- Τα πρόσφατα ληφθέντα κεντροειδή μετά από δεύτερη επανάληψη δίνονται στον παρακάτω πίνακα. Σημειώστε ότι το κεντροειδές c3 παραμένει αμετάβλητο, όπου τα c2 και c1 άλλαξαν λίγο.
- Όσον αφορά τα νεοαποκτηθέντα κέντρα συμπλέγματος, 16 βαθμοί επανατοποθετούνται ξανά. Αυτά είναι τα ίδια συμπλέγματα με πριν. Ως εκ τούτου, τα κεντροειδή τους παραμένουν επίσης αμετάβλητα.
- Θεωρώντας αυτό ως κριτήρια τερματισμού, ο αλγόριθμος k-means σταματά εδώ. Ως εκ τούτου, το τελικό σύμπλεγμα στο τρέχον Σχήμα είναι το ίδιο όπως στο προηγούμενο Σχήμα.

Cluster centres after second iteration

Centroid	Revised Centroids	
	A1	A2
c_1	5.0	7.1
c_2	8.1	12.0
c_3	6.6	18.6

Fig : Cluster after Second iteration



k-Means

https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

k-Means

- Τιμή του k:
  - Ο αλγόριθμος k-means παράγει μόνο ένα σύνολο συστάδων, για τις οποίες ο χρήστης πρέπει να καθορίσει τον επιθυμητό αριθμό, k των συστάδων.
  - Στην πραγματικότητα, το k θα πρέπει να είναι η καλύτερη εικασία για τον αριθμό των συστάδων που υπάρχουν στα δεδομένα. Επομένως, η επιλογή της καλύτερης τιμής του k για ένα δεδομένο σύνολο δεδομένων είναι ένα ζήτημα.
  - Μπορεί να μην έχουμε ιδέα για τον πιθανό αριθμό συμπλεγμάτων για δεδομένα υψηλών διαστάσεων και για δεδομένα που δεν έχουν γραφική διασπορά.
  - Επιπλέον, ο πιθανός αριθμός συμπλεγμάτων είναι κρυμμένος ή διφορούμενος σε εφαρμογές ομαδοποίησης εικόνας, ήχου, βίντεο και πολυμέσων κ.λπ.
  - Δεν υπάρχει τρόπος αρχής για να γνωρίζουμε ποια πρέπει να είναι η τιμή του k. Μπορούμε να δοκιμάσουμε με διαδοχική τιμή του k ξεκινώντας από 2.Η διαδικασία διακόπτεται όταν δύο διαδοχικές τιμές k παράγουν σχεδόν πανομοιότυπα αποτελέσματα (σε σχέση με κάποια εκτίμηση ποιότητας συμπλέγματος).
  - Κανονικά $k \ll n$ και υπάρχει ευρετική τεχνική που πρέπει να ακολουθήσει $k \approx \sqrt{n}$.

k-Means

- Συνήθως, υπάρχει κάποια αντικειμενική συνάρτηση που πρέπει να επιτευχθεί ως στόχος της ομαδοποίησης.
- Μια τέτοια αντικειμενική συνάρτηση είναι το άθροισμα-τετράγωνο-σφάλμα που συμβολίζεται με SSE και ορίζεται ως

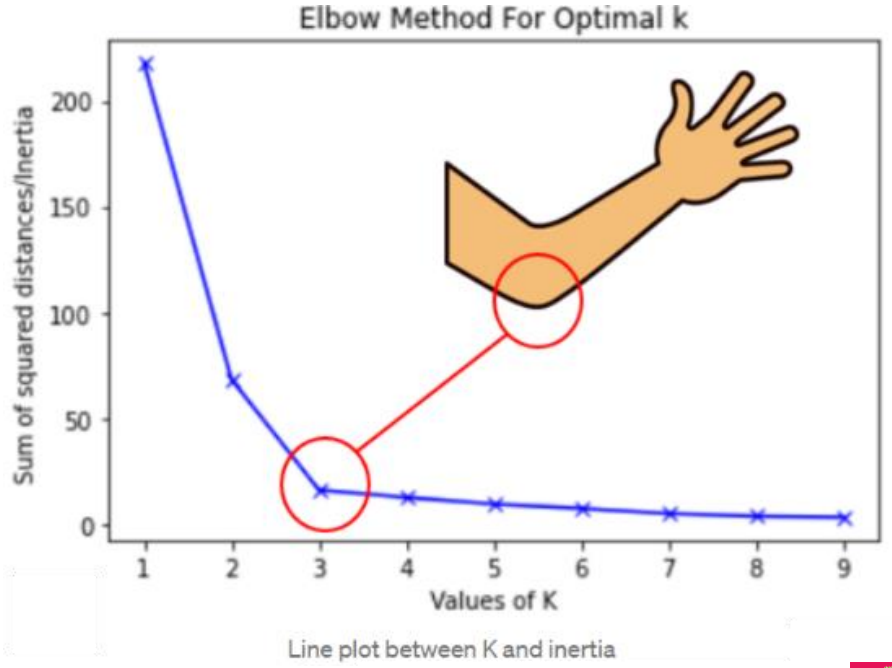
$$SSE = \sum_{i=1}^{k} \sum_{x \in \boldsymbol{C}_i} (x - c_i)^2$$

- Εδώ, το $x-c\_i$ υποδηλώνει το σφάλμα, αν το x βρίσκεται στο σύμπλεγμα $\boldsymbol{C}\_i$ με το κέντρο του συμπλέγματος c_i.
- Συνήθως, αυτό το σφάλμα μετριέται ως νόρμες απόστασης όπως ομοιότητα L1, L2, L3 ή συνημιτόνου κ.λπ.

k-Means

- Αναφορικά με ένα αυθαίρετο πείραμα, ας υποθέσουμε ότι λαμβάνονται τα ακόλουθα αποτελέσματα.
- Σε σχέση με αυτήν την παρατήρηση, μπορούμε να επιλέξουμε την τιμή του $k \approx 3$, καθώς με αυτή τη μικρότερη τιμή του k δίνει αρκετά καλό αποτέλεσμα.
- Σημείωση: Αν $k=n$, τότε SSE=0; Ωστόσο, το αποτέλεσμα είναι άχρηστο! Αυτό είναι ένα άλλο παράδειγμα υπερπροσαρμογής.

k	SSE
1	62.8
2	12.3
3	9.4
4	9.3
5	9.2
6	9.1
7	9.05
8	9.0



k-Means

- Επιλογή αρχικών κεντροειδών:
  - Μια άλλη απαίτηση στον αλγόριθμο k-Means για την επιλογή του αρχικού κέντρου συμπλέγματος για κάθε k θα ήταν οι συστάδες.
  - Παρατηρείται ότι ο αλγόριθμος k-Means τερματίζει όποια και αν είναι η αρχική επιλογή των κεντροειδών συστάδων.
  - Παρατηρείται επίσης ότι η αρχική επιλογή επηρεάζει την τελική ποιότητα του cluster. Με άλλα λόγια, το αποτέλεσμα μπορεί να παγιδευτεί στο τοπικό βέλτιστο, εάν τα αρχικά κεντροειδή επιλεγούν σωστά.
  - Μια τεχνική που συνήθως ακολουθείται για την αποφυγή του παραπάνω προβλήματος είναι η επιλογή αρχικών κεντροειδών σε πολλαπλές εκτελέσεις, το καθένα με διαφορετικό σύνολο τυχαία επιλεγμένων αρχικών κεντροειδών, και στη συνέχεια η επιλογή του καλύτερου συμπλέγματος (σε σχέση με κάποιο κριτήριο μέτρησης ποιότητας, π.χ. SSE).
  - Ωστόσο, αυτή η στρατηγική πάσχει από το πρόβλημα συνδυαστικής έκρηξης λόγω του αριθμού όλων των πιθανών λύσεων.

k-Means

- Επιλογή αρχικών κεντροειδών:
  - Ένας λεπτομερής υπολογισμός αποκαλύπτει ότι υπάρχουν $c(n,k)$ πιθανοί συνδυασμοί για την εξέταση της αναζήτησης του καθολικού βέλτιστου.

$$c(n,k) = \frac{1}{k!}\sum_{i=1}^{k}(-1)^{k-i}\binom{k}{i}(i)^n$$

  - Για παράδειγμα, υπάρχουν o(10^10) διαφορετικοί τρόποι για να ομαδοποιήσετε 20 αντικείμενα σε 4 συστάδες!
  - Έτσι, η στρατηγική που έχει τον δικό της περιορισμό είναι πρακτική μόνο αν
    - Το δείγμα είναι αρνητικά μικρό (~100-1000), και
    - Το k είναι σχετικά μικρό σε σύγκριση με το n (δηλαδή $k \ll n$).

k-Means

- Μέτρηση απόστασης:
  - Για να αντιστοιχίσουμε ένα σημείο στο πλησιέστερο κέντρο, χρειαζόμαστε ένα μέτρο εγγύτητας που θα πρέπει να ποσοτικοποιεί την έννοια του "πλησιέστερου" για τα αντικείμενα υπό ομαδοποίηση.
  - Συνήθως η Ευκλείδεια απόσταση (L2 norm) είναι το καλύτερο μέτρο όταν τα σημεία αντικειμένων ορίζονται σε ν-διάστατο Ευκλείδειο χώρο.
  - Άλλο μέτρο, δηλαδή η ομοιότητα συνημιτόνου, είναι πιο κατάλληλο όταν τα αντικείμενα είναι τύπου εγγράφου.
  - Επιπλέον, ενδέχεται να υπάρχουν άλλου τύπου μέτρα εγγύτητας που είναι κατάλληλα στο πλαίσιο των αιτήσεων.
  - Για παράδειγμα, απόσταση Μανχάταν (L1 norm), μέτρο Jaccard, κ.λπ.

k-Means

- Distance Measurement:
  - Thus, in the context of different measures, the sum-of-squared error (i.e., objective function/convergence criteria) of a clustering can be stated as under.
  - Data in Euclidean space (L2 norm):

$$SSE = \sum_{i=1}^{k} \sum_{x \in C_i} (c_i - x)^2$$

  - Data in Euclidean space (L1 norm):
  - The Manhattan distance (L1 norm) is used as a proximity measure, where the objective is to minimize the sum-of-absolute error denoted as SAE and defined as

$$SAE = \sum_{i=1}^{k} \sum_{x \in C_i} |c_i - x|$$

k-Means

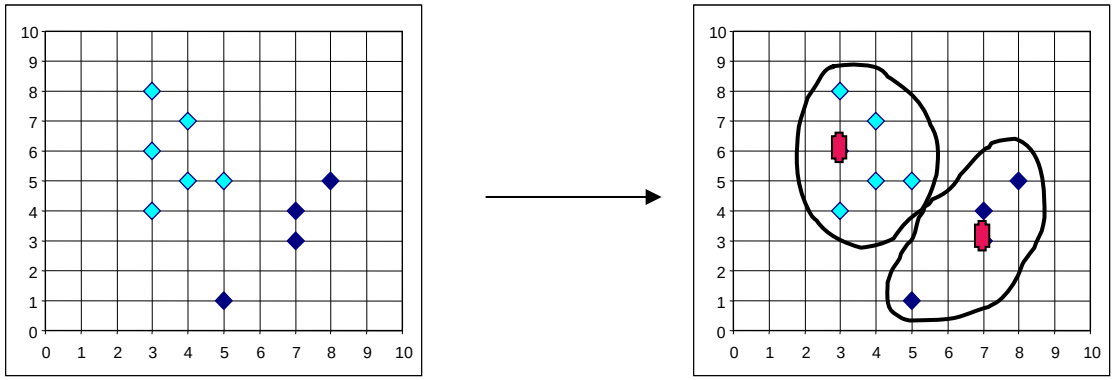
- Ισχύς:
  - Αποτελεσματικό: O(tkn), όπου n είναι # αντικείμενα, k είναι # οι συστάδες και t είναι # επαναλήψεις. Κανονικά, k, t << n.
    - Σύγκριση: PAM: $O(k(n-k)^2)$, CLARA: $O(ks^2 + k(n-k))$
  - Σχόλιο: Συχνά τερματίζεται σε τοπικό βέλτιστο.
  - Αδυναμία
    - Ισχύει μόνο για αντικείμενα σε συνεχή χώρο n-διαστάσεων
    - Χρησιμοποιώντας τη μέθοδο k-modes για κατηγορικά δεδομένα
    - Συγκριτικά, το k-medoids μπορεί να εφαρμοστούν σε ένα ευρύ φάσμα δεδομένων
  - Πρέπει να καθορίσετε το k, τον αριθμό των συστάδων, εκ των προτέρων (υπάρχουν τρόποι να προσδιορίσετε αυτόματα το καλύτερο k (βλέπε Hastie et al., 2009)
  - Ευαίσθητο σε θορυβώδη δεδομένα και ακραίες τιμές
  - Δεν είναι κατάλληλο για την ανακάλυψη συστάδων με μη κυρτά σχήματα

k-Means

- Οι περισσότερες από τις παραλλαγές του k-means που διαφέρουν σε
  - Επιλογή του αρχικού k σημαίνει
  - Υπολογισμοί ανομοιότητας
  - Στρατηγικές για τον υπολογισμό των μέσων συστάδων
- Χειρισμός κατηγορικών δεδομένων: k-modes
  - Αντικατάσταση μέσων συστάδων με λειτουργίες
  - Χρήση νέων μέτρων ανομοιότητας για την αντιμετώπιση κατηγορικών αντικειμένων
  - Χρήση μιας μεθόδου που βασίζεται στη συχνότητα για την ενημέρωση των τρόπων λειτουργίας των συστάδων
  - Ένα μείγμα κατηγορικών και αριθμητικών δεδομένων: μέθοδος k-prototype

k-Means

- Ο αλγόριθμος k-means είναι ευαίσθητος σε ακραίες τιμές!
- Δεδομένου ότι ένα αντικείμενο με εξαιρετικά μεγάλη τιμή μπορεί να παραμορφώσει ουσιαστικά την κατανομή των δεδομένων
- K-Medoids: Αντί να λαμβάνεται ως σημείο αναφοράς η μέση τιμή του αντικειμένου σε μια συστάδα, μπορούν να χρησιμοποιηθούν medoids, τα οποία είναι το πιο κεντρικό αντικείμενο σε μια συστάδα

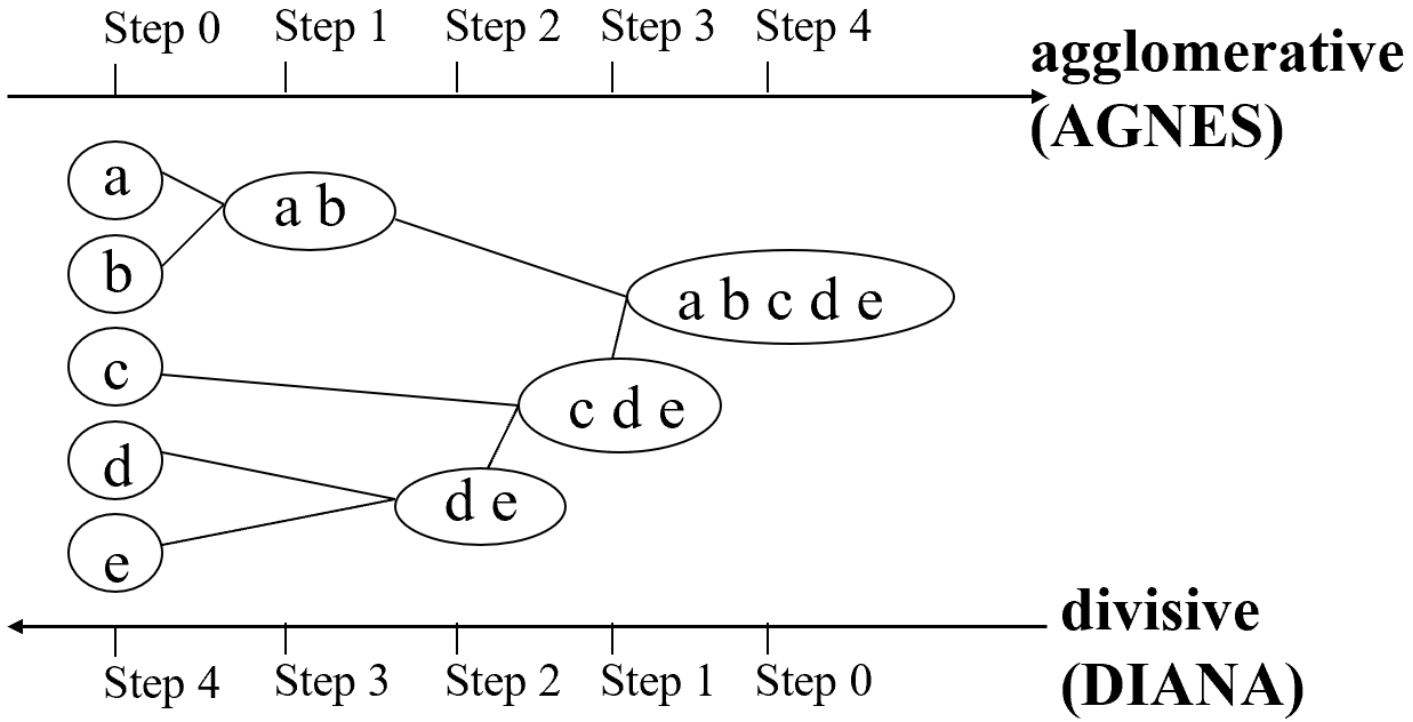


k-Means

- K-Medoids Clustering: Βρείτε αντιπροσωπευτικά αντικείμενα (medoids) σε συστάδες
  - PAM (Partitioning Around Medoids, Kaufmann & Rousseeuw 1987)
    - Ξεκινά από ένα αρχικό σύνολο medoids και αντικαθιστά επαναληπτικά ένα από τα medoids με ένα από τα non-medoids εάν βελτιώνει τη συνολική απόσταση της προκύπτουσας ομαδοποίησης
    - Το PAM λειτουργεί αποτελεσματικά για μικρά σύνολα δεδομένων, αλλά δεν κλιμακώνεται καλά για μεγάλα σύνολα δεδομένων (λόγω της υπολογιστικής πολυπλοκότητας)
  - Βελτίωση αποτελεσματικότητας στο PAM
    - CLARA (Kaufmann & Rousseeuw, 1990): PAM σε δείγματα
    - CLARANS (Ng & Han, 1994): Τυχαιοποιημένη επαναδειγματοληψία

Ιεραρχικό Σχήμα

- Χρησιμοποιήστε τον πίνακα απόστασης ως κριτήρια ομαδοποίησης
- Αυτή η μέθοδος δεν απαιτεί τον αριθμό των συστάδων k ως είσοδο, αλλά χρειάζεται μια συνθήκη τερματισμού



BIRCH

- Κατασκευάστε σταδιακά ένα δέντρο CF (Clustering Feature), μια ιεραρχική δομή δεδομένων για πολυφασική ομαδοποίηση
  - Φάση 1: σάρωση DB για τη δημιουργία ενός αρχικού δέντρου CF στη μνήμη (μια συμπίεση πολλαπλών επιπέδων των δεδομένων που προσπαθεί να διατηρήσει την εγγενή δομή ομαδοποίησης των δεδομένων)
  - Φάση 2: χρησιμοποιήστε έναν αυθαίρετο αλγόριθμο ομαδοποίησης για να ομαδοποιήσετε τους κόμβους των φύλλων του δέντρου CF
- Κλιμακώνει γραμμικά: βρίσκει μια καλή ομαδοποίηση με μία μόνο σάρωση και βελτιώνει την ποιότητα με μερικές επιπλέον σαρώσεις
- Αδυναμία: χειρίζεται μόνο αριθμητικά δεδομένα και είναι ευαίσθητο στη σειρά της εγγραφής δεδομένων

BIRCH

Clustering Feature (CF): ***CF = (N, LS, SS)***

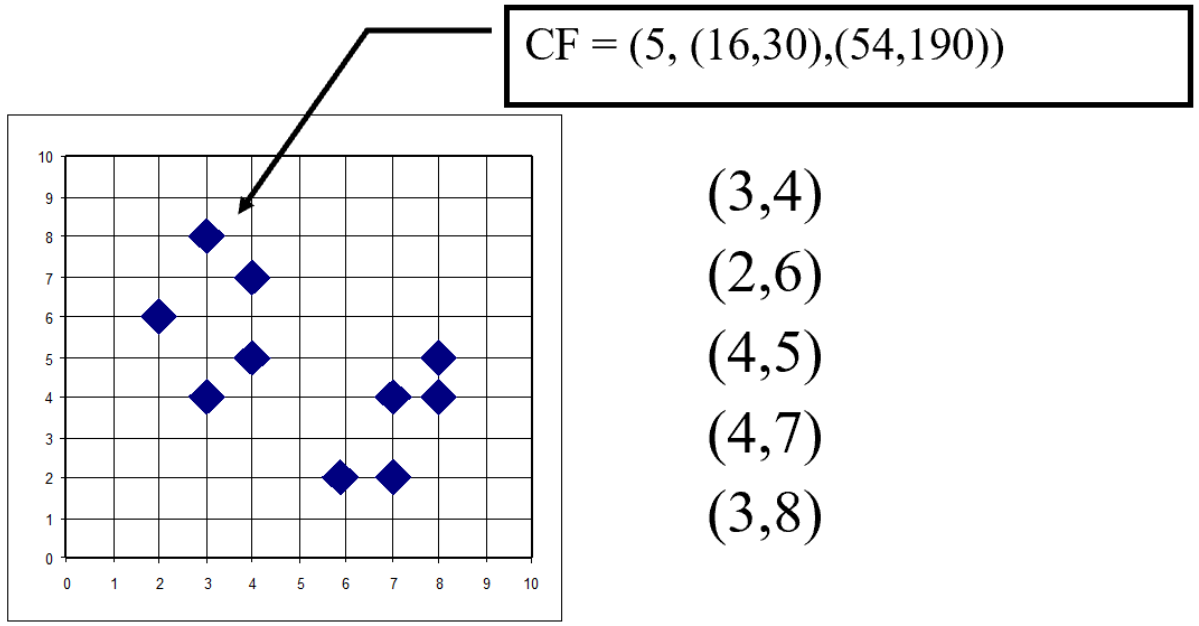
N: Number of data points

LS: linear sum of N points:

$$\sum_{i=1}^{N} X_i$$

SS: square sum of N points

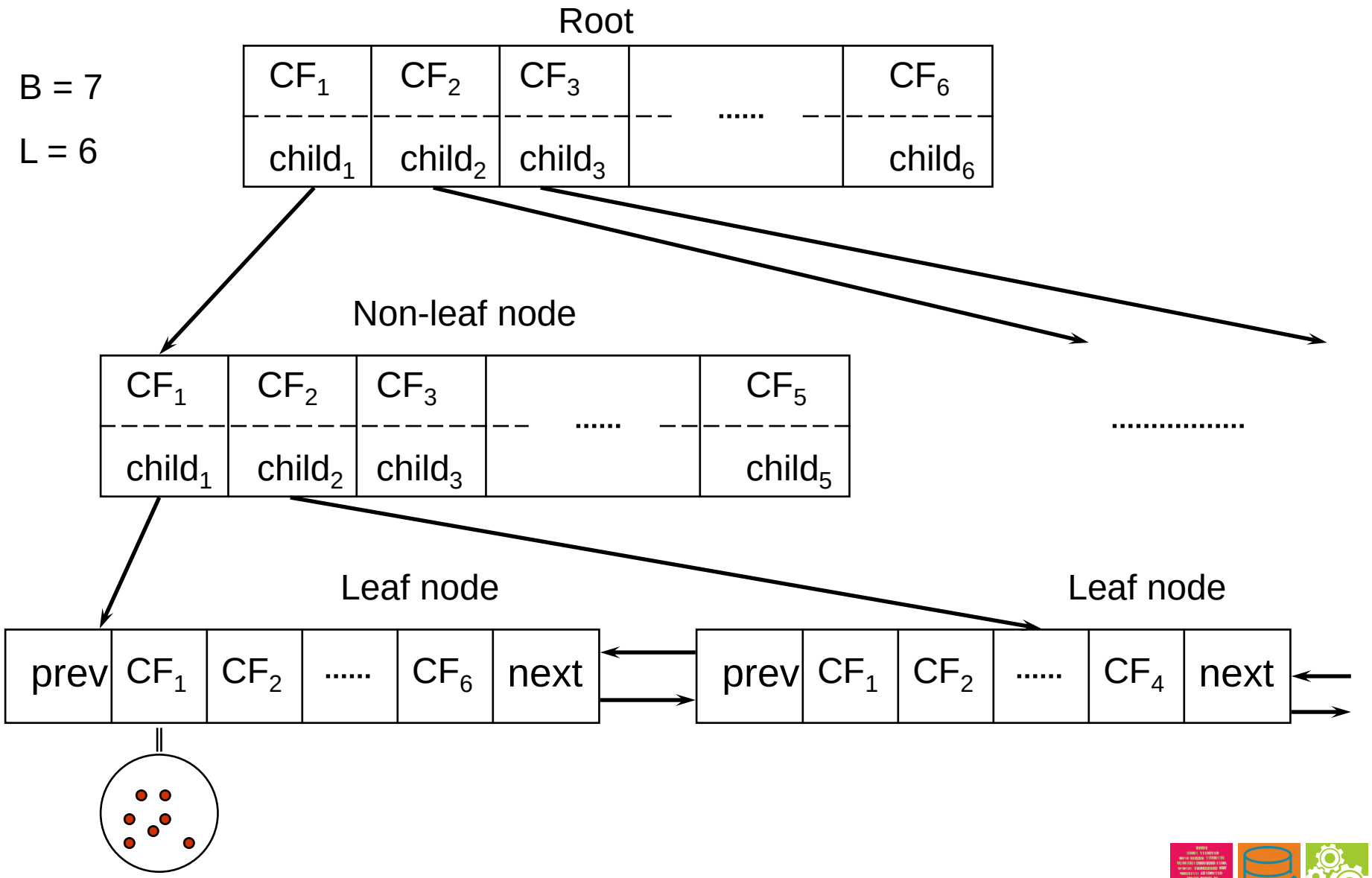
$$\sum_{i=1}^{N} X_i^2$$



BIRCH

- Χαρακτηριστικό ομαδοποίησης:
  - Σύνοψη των στατιστικών για ένα δεδομένο υποσυστάδα: η 0-η, 1η και 2η στιγμή της υποσυστάδας από στατιστική άποψη
  - Καταγράφει κρίσιμες μετρήσεις για υπολογιστικό σύμπλεγμα και χρησιμοποιεί αποτελεσματικά την αποθήκευση
- Ένα δέντρο CF είναι ένα δέντρο ισορροπημένου ύψους που αποθηκεύει τα χαρακτηριστικά ομαδοποίησης για μια ιεραρχική ομαδοποίηση
  - Ένας κόμβος χωρίς φύλλα σε ένα δέντρο έχει απογόνους ή «παιδιά»
  - Οι μη φύλλα κόμβοι αποθηκεύουν αθροίσματα των KI των παιδιών τους
- Ένα δέντρο CF έχει δύο παραμέτρους
- Συντελεστής διακλάδωσης (Branching factor) B: μέγιστος αριθμός παιδιών
- Κατώφλι L: μέγιστη διάμετρος των υποσυστάδων που αποθηκεύονται στους κόμβους των φύλλων

BIRCH


Root
B = 7
L = 6
CF1
CF2
CF3
.....
CF6
child1
child2
child3
child6
Non-leaf node
CF1
CF2
CF3
.....
CF5
child1
child2
child3
child5
.....
Leaf node
Leaf node
prev
CF1
CF2
.....
CF6
next
prev
CF1
CF2
.....
CF4
next

BIRCH

- Διάμετρος συστάδας

$$\sqrt{\frac{1}{n(n-1)}\sum(x_i - x_j)^2}$$

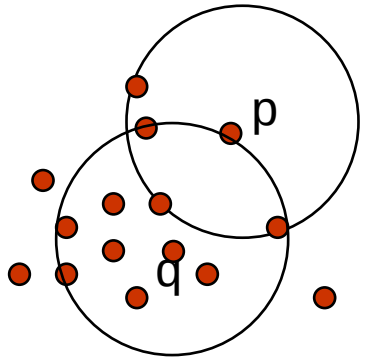
- Για κάθε σημείο στην είσοδο
  - Βρείτε την πλησιέστερη καταχώρηση φύλλου
  - Προσθήκη σημείου στην καταχώρηση φύλλου και ενημέρωση CF
  - Εάν η διάμετρος καταχώρισης > max_diameter, τότε χωρίστε το φύλλο, και πιθανώς γονείς
- Ο αλγόριθμος είναι O(n)
- Ανησυχίες
  - Ευαίσθητο στη σειρά εισαγωγής των σημείων δεδομένων
  - Δεδομένου ότι καθορίζουμε το μέγεθος των κόμβων των φύλλων, έτσι οι συστάδες μπορεί να μην είναι τόσο φυσικές
  - Οι συστάδες τείνουν να είναι σφαιρικές λόγω των μέτρων ακτίνας και διαμέτρου

Πυκνότητα

- Ομαδοποίηση με βάση την πυκνότητα (τοπικό κριτήριο συμπλέγματος), όπως σημεία που συνδέονται με την πυκνότητα
- Κύρια χαρακτηριστικά:
  - Ανακάλυψη συστάδες αυθαίρετου σχήματος
  - Χειρισμός του θορύβου
  - Μία σάρωση
  - Απαιτούνται παράμετροι πυκνότητας ως συνθήκη τερματισμού
- Αρκετές ενδιαφέρουσες μελέτες:
  - DBSCAN: Ester, et al. (KDD’96)
  - OPTICS: Ankerst, et al (SIGMOD'99)
  - DENCLUE: Hinneburg & D. Keim (KDD'98)
  - CLIQUE: Agrawal, et al. (SIGMOD'98) (περισσότερο βασισμένο σε πλέγμα)

Πυκνότητα

- Δύο παράμετροι:
  - Eps: Μέγιστη ακτίνα της γειτονιάς
  - MinPts: Ελάχιστος αριθμός σημείων σε μια γειτονιά Eps αυτού του σημείου
- $N_{Eps}(p)$: Το {q ανήκει στο D | dist(p,q) ≤ Eps}
- Άμεσα προσβάσιμη πυκνότητα (Directly density-reachable): Ένα σημείο p είναι άμεσα προσβάσιμο σε πυκνότητα από ένα σημείο q w.r.t. Eps, MinPts αν
  - Το p ανήκει στο $N_{Eps}(q)$
  - συνθήκη βασικού σημείου: $|N_{Eps}(q)| \geq MinPts$

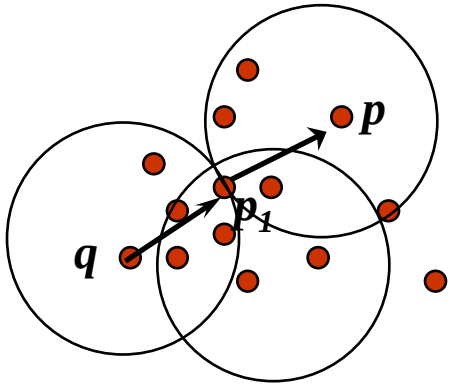


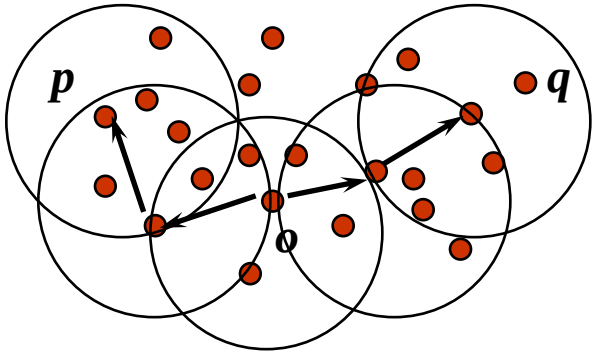
MinPts = 5

Eps = 1 cm

Πυκνότητα

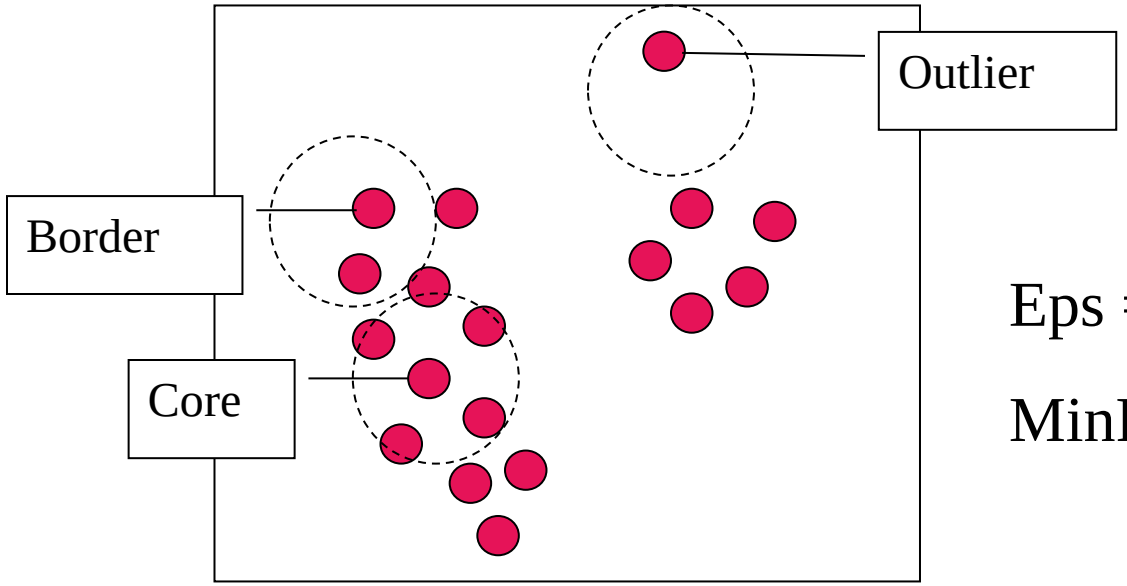
- **Density-reachable:**
  - A point *p* is density-reachable from a point *q* w.r.t. *Eps*, *MinPts* if there is a chain of points $p_1, \ldots, p_n, p_1 = q, p_n = p$ such that p_{i+1} is directly density-reachable from p_i
- **Density-connected**
  - A point *p* is density-connected to a point *q* w.r.t. *Eps*, *MinPts* if there is a point *o* such that both, *p* and *q* are density-reachable from *o* w.r.t. *Eps* and *MinPts*





DBSCAN

- Βασίζεται σε μια έννοια της συστάδας που βασίζεται στην πυκνότητα:
- Μια συστάδα ορίζεται ως ένα μέγιστο σύνολο σημείων που συνδέονται με την πυκνότητα
- Ανακαλύπτει συστάδες αυθαίρετου σχήματος σε χωρικές βάσεις δεδομένων με θόρυβο



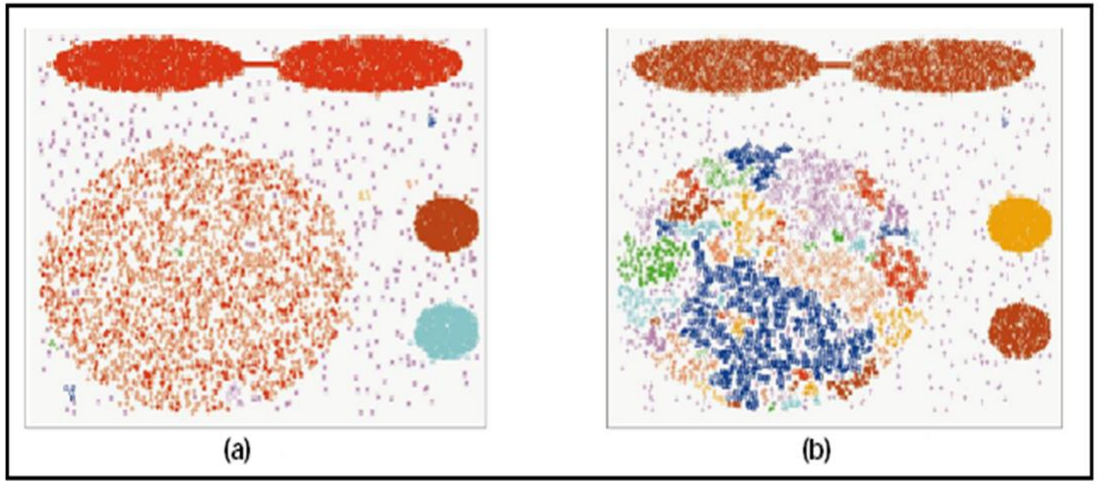
Eps = 1cm

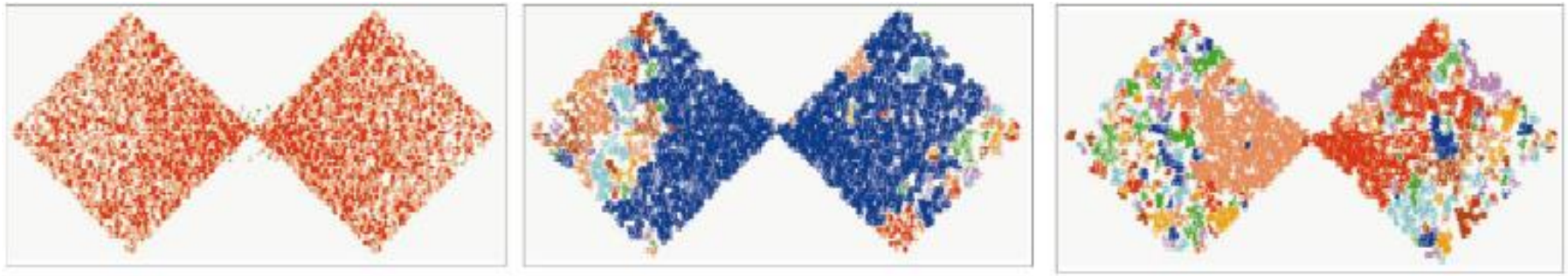
MinPts = 5

DBSCAN

- Arbitrary select a point *p*
- Retrieve all points density-reachable from *p* w.r.t. *Eps* and *MinPts*
- If *p* is a core point, a cluster is formed
- If *p* is a border point, no points are density-reachable from *p* and DBSCAN visits the next point of the database
- Continue the process until all of the points have been processed

DBSCAN





Αποτίμηση

- Εκτιμήστε εάν υπάρχει μη τυχαία δομή στα δεδομένα μετρώντας την πιθανότητα τα δεδομένα να δημιουργούνται από μια ομοιόμορφη κατανομή δεδομένων
- Έλεγχος χωρικής τυχαιότητας με στατιστική δοκιμή: Στατιστική Hopkins
  - Δεδομένου ενός συνόλου δεδομένων D που θεωρείται ως δείγμα μιας τυχαίας μεταβλητής ο, προσδιορίστε πόσο μακριά είναι η ο από την ομοιόμορφη κατανομή στο χώρο δεδομένων
  - Δείγμα n σημείων, p1, …, pn, ομοιόμορφα από το D.
  - Για κάθε pi, βρείτε τον πλησιέστερο γείτονά του στο D: xi = min{dist (pi, v)} όπου v στο D
  - Δείγμα n σημείων, q1, …, qn, ομοιόμορφα από το D. Για κάθε qi, βρείτε τον πλησιέστερο γείτονά του σε D – {qi}: yi = min{dist (qi, v)} όπου v στο D και v ≠ qi
  - Υπολογίστε τη στατιστική Hopkins:

$$H = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i + \sum_{i=1}^{n} y_i}$$

  - Εάν το D είναι ομοιόμορφα κατανεμημένο, το ∑ xi και το ∑ yi θα είναι κοντά το ένα στο άλλο και το H είναι κοντά στο 0,5. Εάν το D είναι πολύ κυρτό, το H είναι κοντά στο 0

Αποτίμηση

- Εμπειρική μέθοδος
  - # συστάδων ≈√n/2 για ένα σύνολο δεδομένων n σημείων
- Μέθοδος αγκώνα
  - Χρησιμοποιήστε το σημείο καμπής στην καμπύλη του αθροίσματος της διακύμανσης εντός συστάδων με το # των συστάδων
- Διασταυρούμενη μέθοδος επικύρωσης (Cross validation)
  - Διαιρέστε ένα δεδομένο σύνολο δεδομένων σε m μέρη
  - Χρησιμοποιήστε m – 1 μέρη για να αποκτήσετε ένα μοντέλο ομαδοποίησης
  - Χρησιμοποιήστε το υπόλοιπο μέρος για να ελέγξετε την ποιότητα της ομαδοποίησης
    - Π.χ., για κάθε σημείο του συνόλου δοκιμής, βρείτε το πλησιέστερο κέντρο και χρησιμοποιήστε το άθροισμα της τετραγωνικής απόστασης μεταξύ όλων των σημείων του δοκιμαστικού συνόλου και των πλησιέστερων κεντροειδών για να μετρήσετε πόσο καλά ταιριάζει το μοντέλο στο σύνολο δοκιμής
  - Για οποιοδήποτε k > 0, επαναλάβετε το m φορές, συγκρίνετε το συνολικό μέτρο ποιότητας w.r.t. διαφορετικά k και βρείτε # συστάδες που ταιριάζουν καλύτερα στα δεδομένα

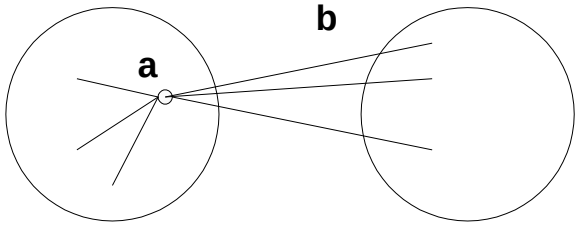
Αποτίμηση

- Δύο μέθοδοι: εξωγενής (extrinsic) έναντι ενδογενούς (intrinsic)
- Εξωγενής: υπό επίβλεψη, δηλ., η βασική αλήθεια είναι διαθέσιμη
  - Συγκρίνετε μια ομαδοποίηση με τη βασική αλήθεια χρησιμοποιώντας συγκεκριμένο μέτρο ποιότητας ομαδοποίησης
  - Παράδειγμα. Bcubed, μετρήσεις ακριβείας (precision) και ανάκλησης (recall)
- Εγγενής: χωρίς επίβλεψη, δηλαδή, η βασική αλήθεια δεν είναι διαθέσιμη
  - Αξιολογήστε την ωφέλεια μιας ομαδοποίησης λαμβάνοντας υπόψη πόσο καλά είναι διαχωρισμένα τα συμπλέγματα και πόσο συμπαγή είναι τα συμπλέγματα
  - παράδειγμα. Συντελεστής σιλουέτας (Silhouette coefficient)

Silhouette Coefficient

- Silhouette Coefficient combines ideas of both cohesion and separation, but for individual points, as well as clusters and clusterings
- For an individual point i
  - Calculate $\boldsymbol{a_i}$ = average distance of i to the points in its own cluster
  - Calculate $\boldsymbol{b_i}$ = min (over clusters) of the average distance of i to points in other clusters
  - The silhouette coefficient for a point i is then given by

$$\boldsymbol{s_i = 1 - a_i / b_i}$$



  - Typically between 0 and 1, the closer to 1 the better.
  - Can be less than 0 but this is a problematic case

- Can calculate the Average Silhouette coefficient for a cluster, or for a clustering

Silhouette Coefficient

https://www.mathworks.com/help/stats/silhouette.html

Silhouette Coefficient

Cluster 1 ={{1,0},{1,1}}
Cluster 2 ={{1,2},{2,3},{2,2},{1,2}},
Cluster 3 ={{3,1},{3,3},{2,1}}

Take a point {1,0} in cluster 1
Calculate its **average distance** to all other points in **it's cluster**, i.e. cluster 1

So a1 =√((1-1)^2 + (0-1)^2) =√(0+1)=√1=1

Now for the object {1,0} in cluster 1 calculate its average distance from all the objects in cluster 2 and cluster 3.
Of these take the **minimum** average distance.

So for cluster 2
{1,0} ----> {1,2} = distance = √((1-1)^2 + (0-2)^2) =√(0+4)=√4=2
{1,0} ----> {2,3} = distance = √((1-2)^2 + (0-3)^2) =√(1+9)=√10=3.16
{1,0} ----> {2,2} = distance = √((1-2)^2 + (0-2)^2) =√(1+4)=√5=2.24
{1,0} ----> {1,2} = distance = √((1-1)^2 + (0-2)^2) =√(0+4)=√4=2

Silhouette Coefficient

Cluster 1 ={{1,0},{1,1}}
Cluster 2 ={{1,2},{2,3},{2,2},{1,2}},
Cluster 3 ={{3,1},{3,3},{2,1}}

Therefore, the average distance of point {1,0} in cluster 1 to all the points in cluster 2 = (2+3.16+2.24+2)/4 = 2.325

Similarly, for cluster 3
{1,0} ----> {3,1} = distance = √((1-3)^2 + (0-1)^2) =√(4+1)=√5=2.24
{1,0} ----> {3,3} = distance = √((1-3)^2 + (0-3)^2) =√(4+9)=√13=3.61
{1,0} ----> {2,1} = distance = √((1-2)^2 + (0-1)^2) =√(1+1)=√2=2.24

Therefore, the **average distance** of point {1,0} in cluster 1 to all the points in cluster 3 = (2.24+3.61+2.24)/3 = 2.7

Now, the **minimum** average distance of the point {1,0} in cluster 1 to the other clusters 2 and 3 is, b1 =2.325 (2.325 < 2.7)

Silhouette Coefficient

Cluster 1 ={{1,0},{1,1}}
Cluster 2 ={{1,2},{2,3},{2,2},{1,2}},
Cluster 3 ={{3,1},{3,3},{2,1}}

So the silhouette coefficient of cluster 1
s1= 1-(a1/b1) = 1- (1/2.325)=1-0.4301=0.5699

In a similar fashion you need to calculate the silhouette coefficient for cluster 2 and cluster 3 separately by taking any single object point in each of the clusters and repeating the steps above.

Of these the cluster with the greatest silhouette coefficient is the best as per evaluation.

Silhouette Coefficient

Cluster 1 ={{1,0},{1,1}}
Cluster 2 ={{1,2},{2,3},{2,2},{1,2}},
Cluster 3 ={{3,1},{3,3},{2,1}}

So the silhouette coefficient of cluster 1
s1= 1-(a1/b1) = 1- (1/2.325)=1-0.4301=0.5699

In a similar fashion you need to calculate the silhouette coefficient for cluster 2 and cluster 3 separately by taking any single object point in each of the clusters and repeating the steps above.

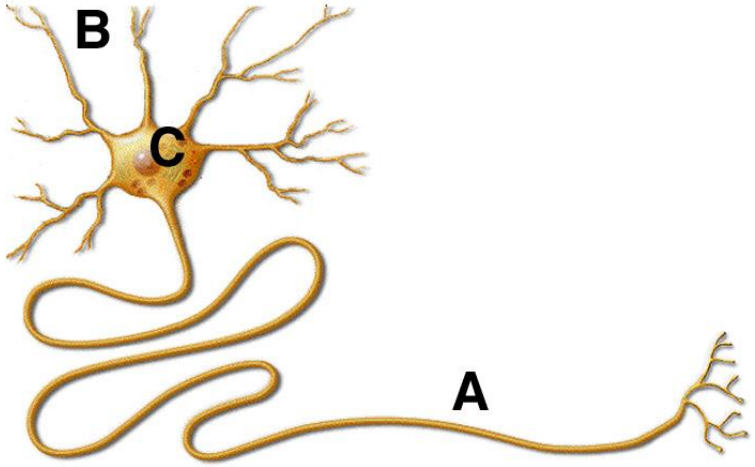
Of these the cluster with the greatest silhouette coefficient is the best as per evaluation.

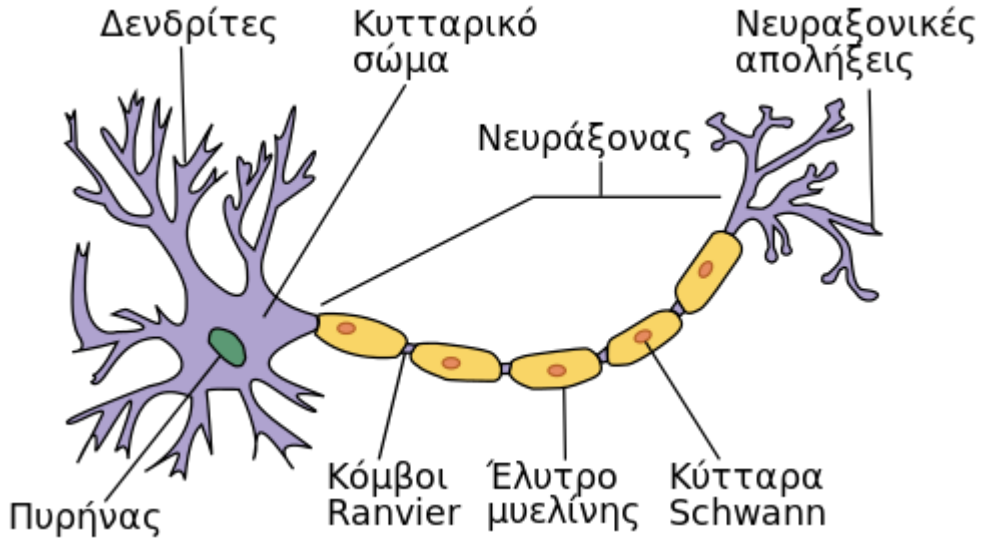
Artificial Neural Networks

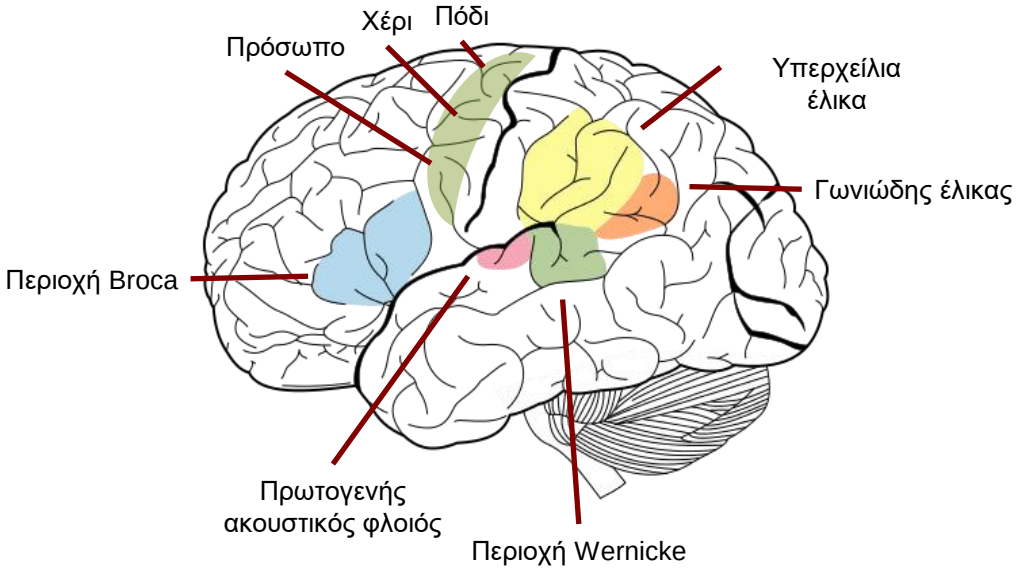
Εισαγωγή

- Έχουμε δισεκατομμύρια και δισεκατομμύρια νευρώνες που κατά κάποιο τρόπο συνεργάζονται για να δημιουργήσουν το μυαλό.
- Αυτοί οι νευρώνες συνδέονται με 10^{14} - 10^{15} συνάψεις, οι οποίες πιστεύουμε ότι κωδικοποιούν τη «γνώση» στο δίκτυο - πάρα πολλοί για να τους προγραμματίσουμε ρητά στα μοντέλα μας
- Μάλλον χρειαζόμαστε κάποιον τρόπο να τα ορίσουμε έμμεσα μέσω μιας διαδικασίας που θα πετύχει κάποιο στόχο αλλάζοντας τις συναπτικές δυνάμεις (που ονομάζουμε βάρη).
- Αυτό ονομάζεται ***μάθηση*** σε αυτά τα συστήματα.

Εισαγωγή



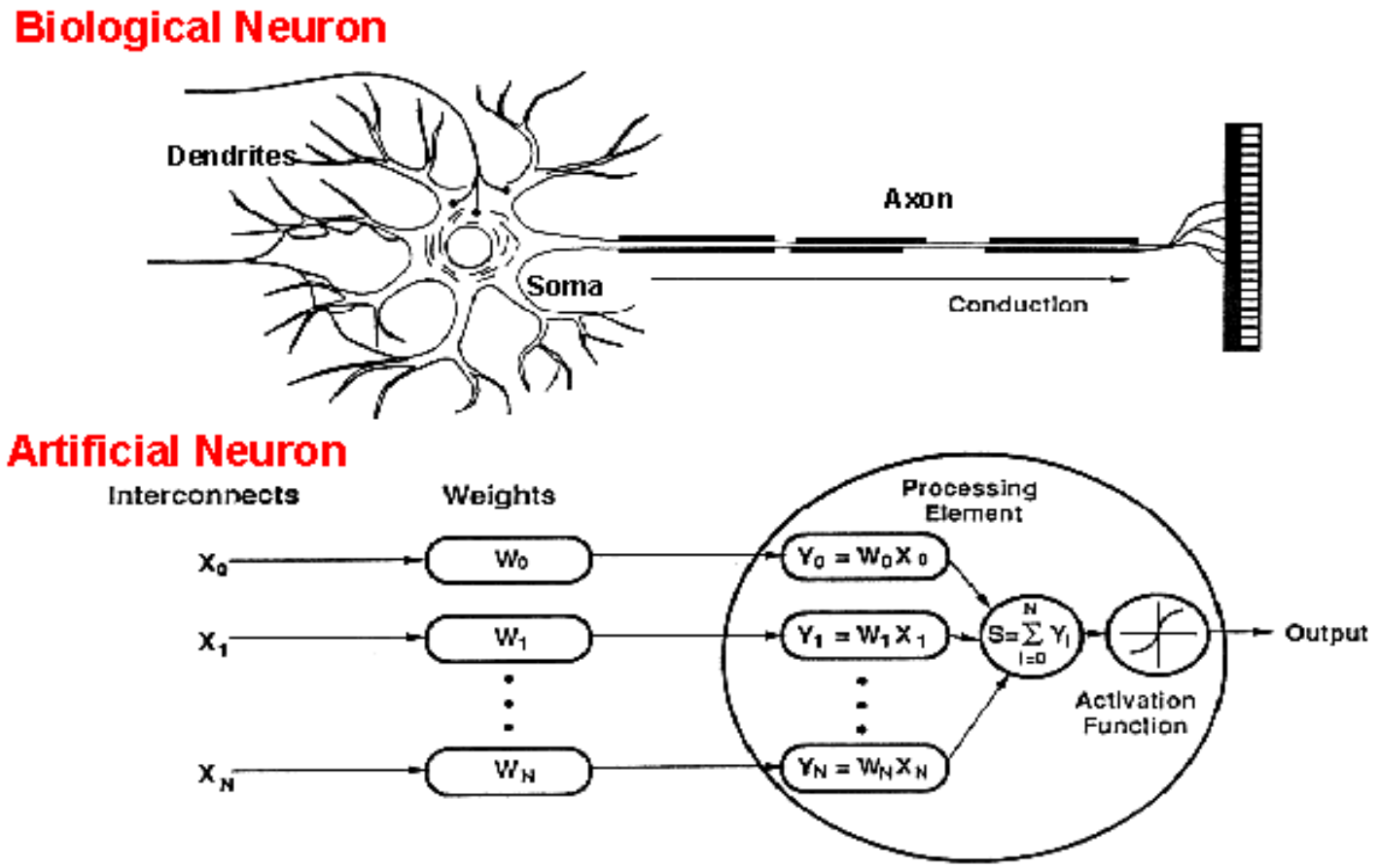




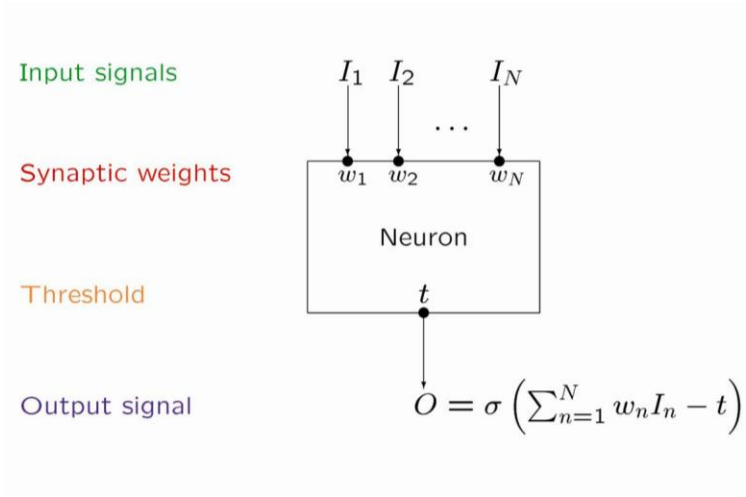
Ένας νευρώνας λαμβάνει είσοδο από άλλους νευρώνες. Οι εισροές συνδυάζονται.
Μόλις η είσοδος υπερβεί ένα κρίσιμο επίπεδο, ο νευρώνας εκφορτώνει μια ακίδα - έναν ηλεκτρικό παλμό που ταξιδεύει από το σώμα, κάτω από τον άξονα, στον επόμενο νευρώνα
Οι απολήξεις του άξονα σχεδόν αγγίζουν τους δενδρίτες ή το κυτταρικό σώμα του επόμενου νευρώνα.

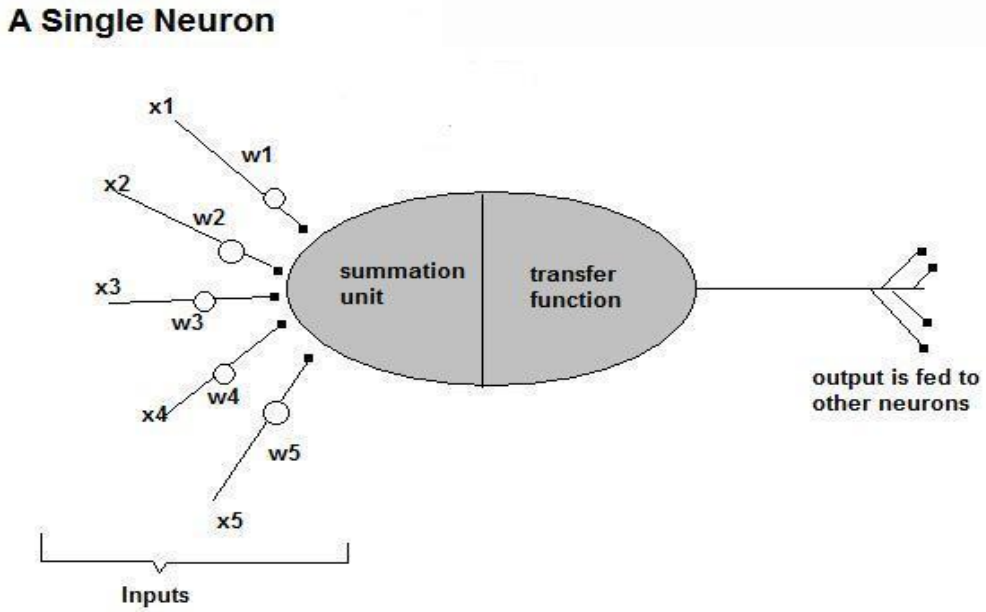
Λειτουργία

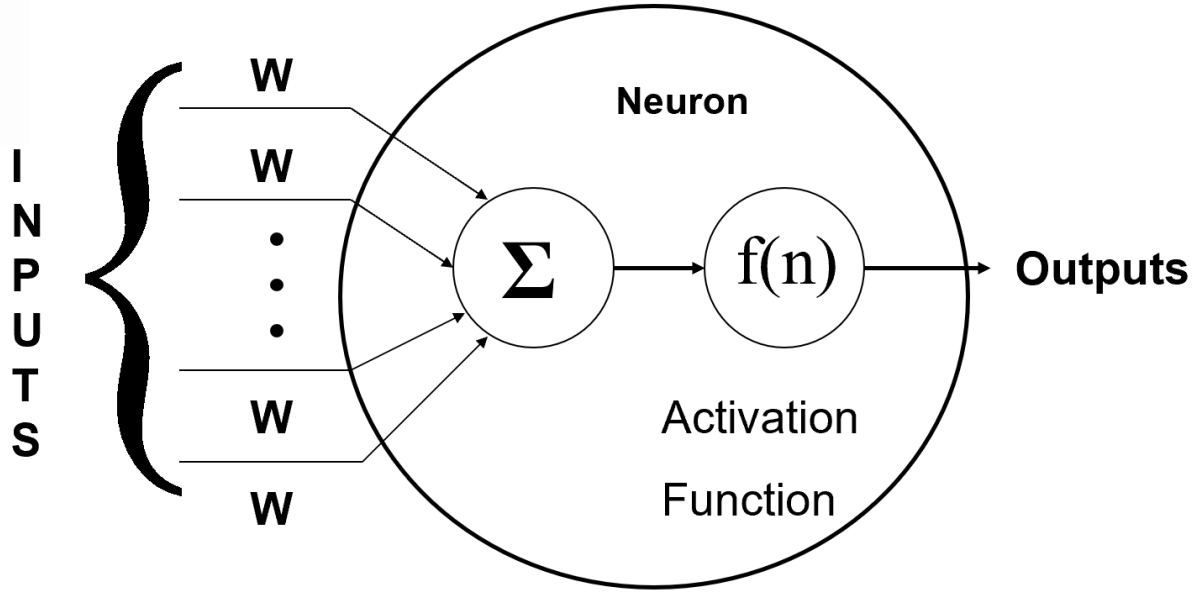
- Ένας τεχνητός νευρώνας είναι μια απομίμηση ενός ανθρώπινου νευρώνα



Λειτουργία



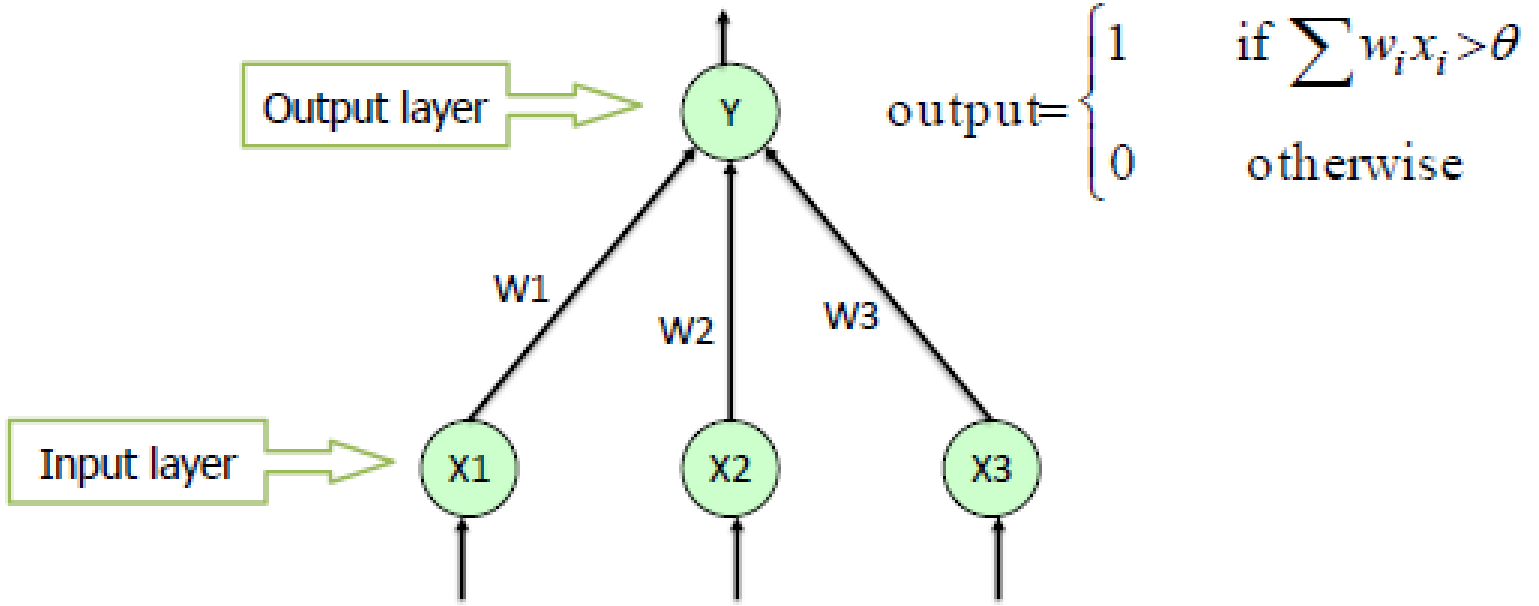




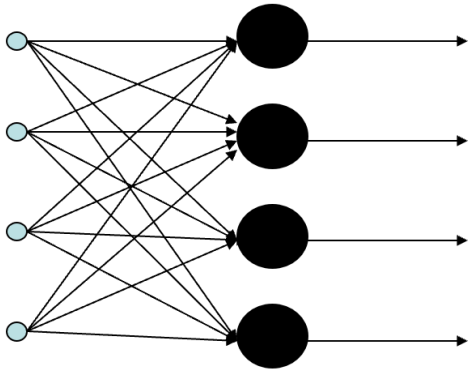
W=Weight

Λειτουργία

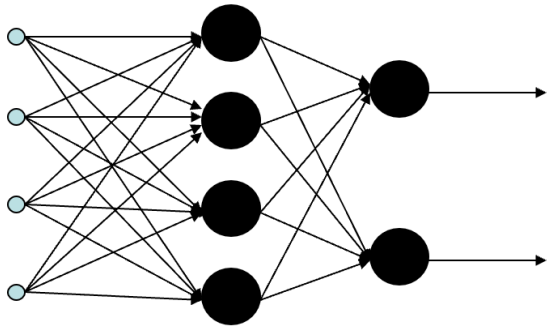
Single Layer Perceptron



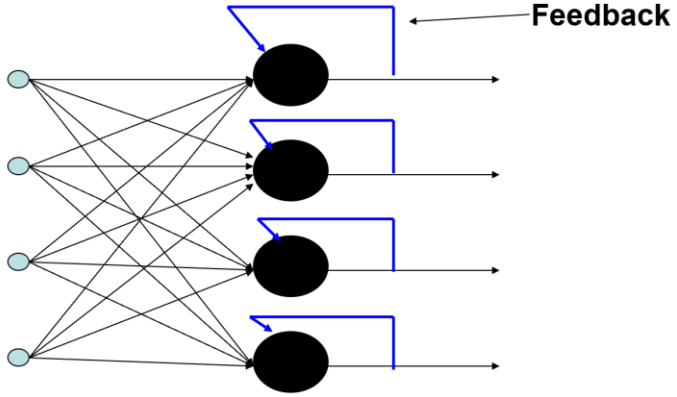
Κατηγορίες



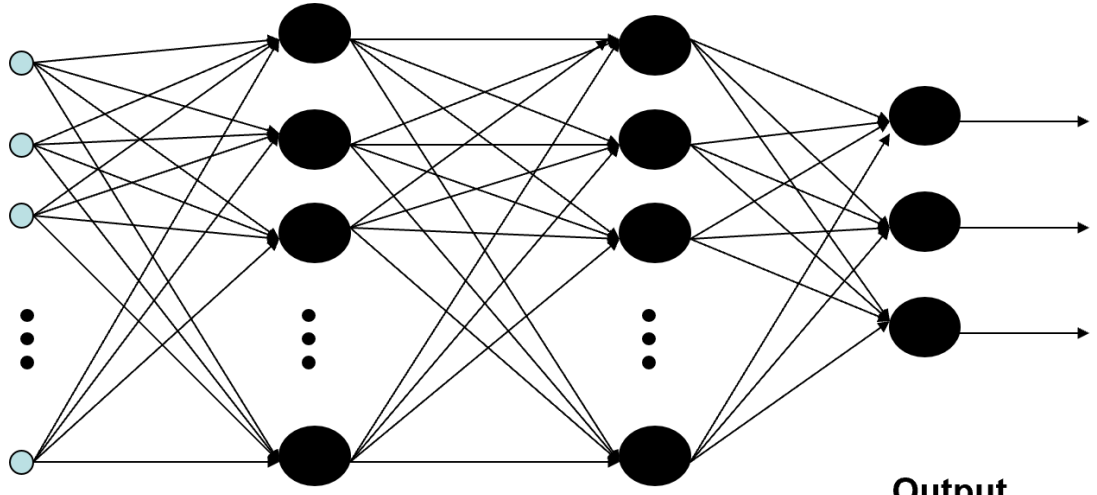
Multiple Inputs and Single Layer



Multiple Inputs and layers



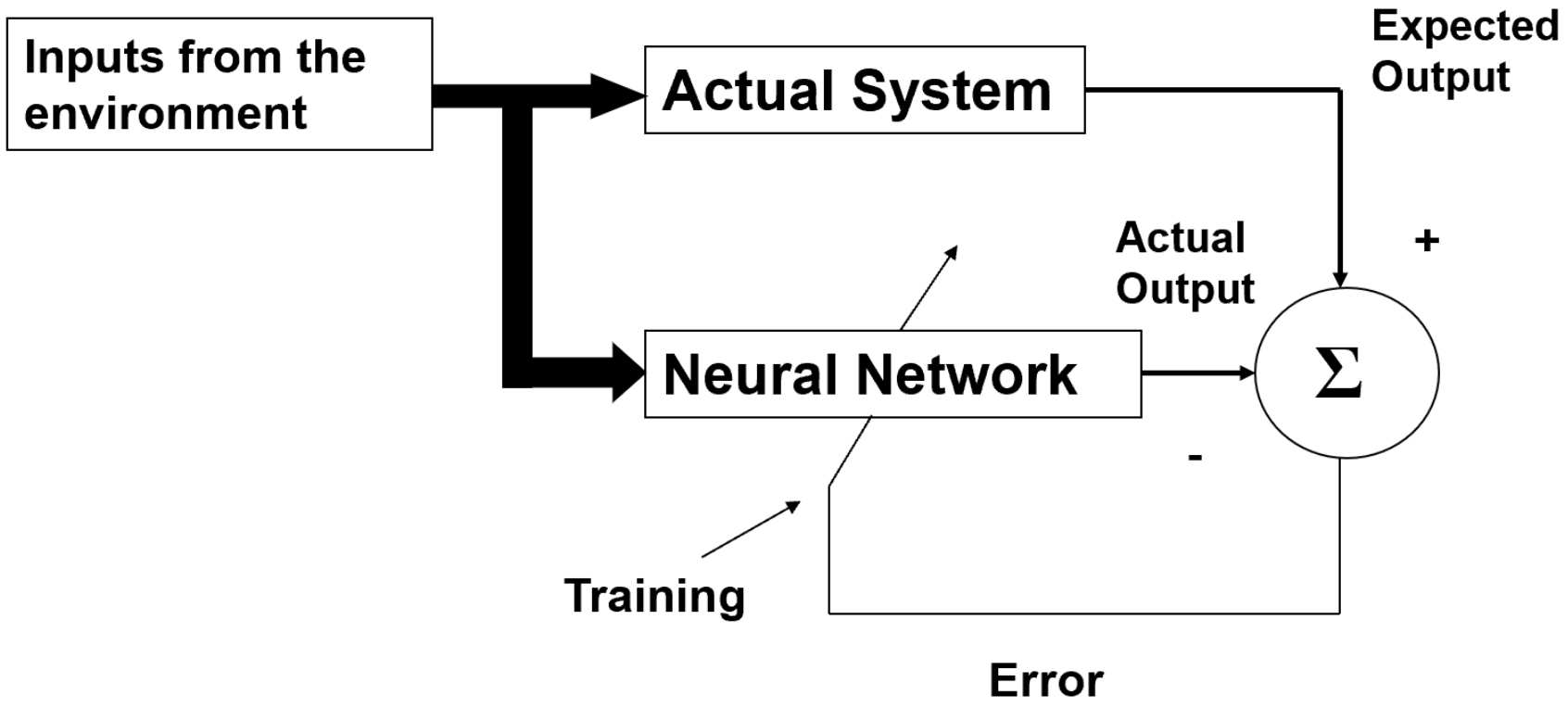
Recurrent Networks



Μάθηση

- Η διαδικασία που συνίσταται στην εκτίμηση των παραμέτρων των νευρώνων έτσι ώστε ολόκληρο το δίκτυο να μπορεί να εκτελέσει μια συγκεκριμένη εργασία
- 2 είδη μάθησης
  - Η εποπτευόμενη μάθηση
  - Η μάθηση χωρίς επίβλεψη
- Η διαδικασία μάθησης (υπό επίβλεψη)
  - Παρουσιάστε στο δίκτυο έναν αριθμό εισόδων και τις αντίστοιχες εξόδους τους
  - Δείτε πόσο πολύ ταιριάζουν οι πραγματικές έξοδοι με τις επιθυμητές
  - Τροποποιήστε τις παραμέτρους για να προσεγγίσετε καλύτερα τις επιθυμητές εξόδους

Μάθηση


Inputs from the environment
Actual System
Expected Output
Actual Output
+
Neural Network
Σ
-
Training
Error

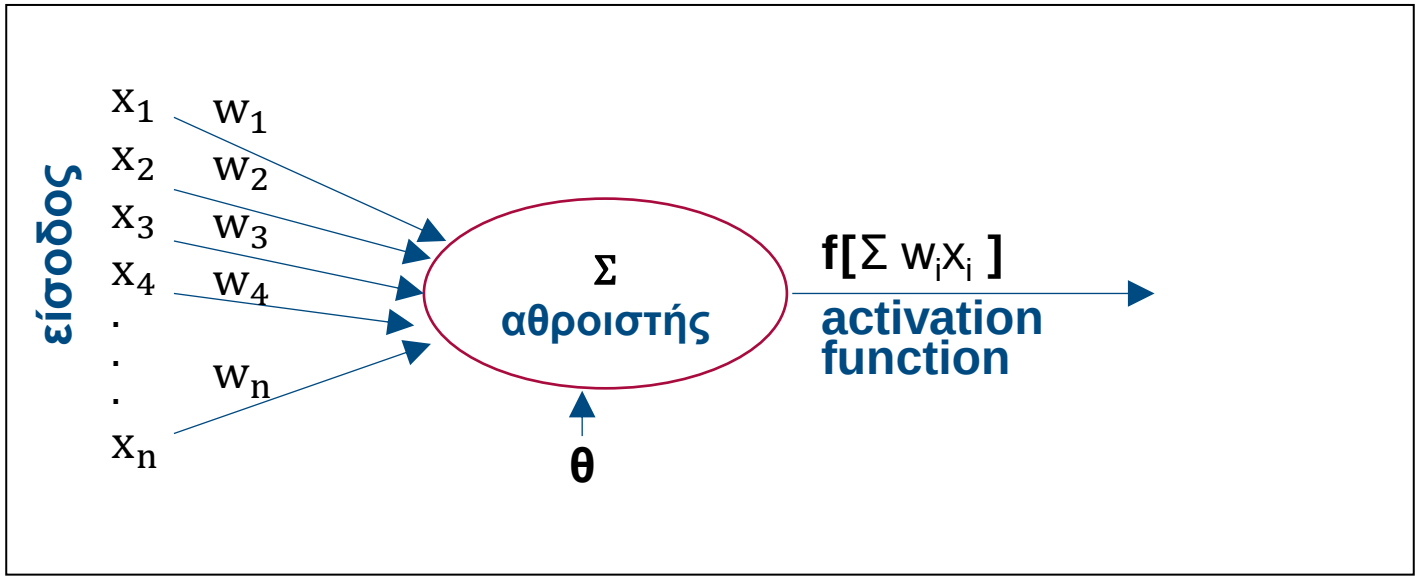
Συνδυασμός Εισόδων

- Συνάρτηση αθροίσματος Σ (summation function):

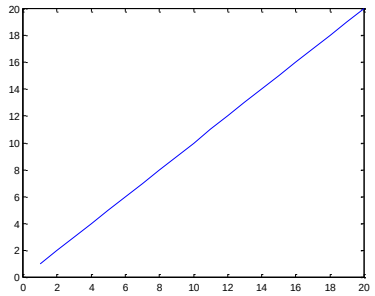
$$\sum^{n} x_i w_i$$

- Έξοδος μέσω συνάρτησης μετάβασης g (activation function) ή συνάρτηση ενεργοποίησης ή συνάρτηση κατωφλίου όταν:

$$\sum_{i=1}^{n} x_i w_I - \theta > 0$$

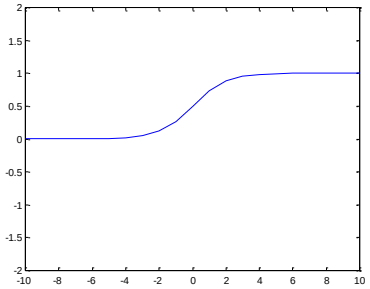


Activation



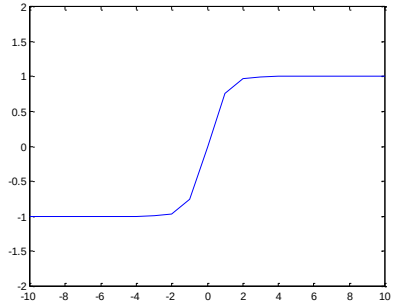
Linear

$$y = x$$



Logistic

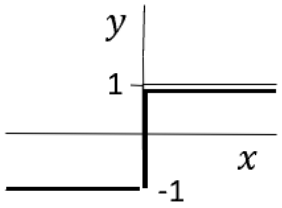
$$y = \frac{1}{1+\exp(-x)}$$



Hyperbolic tangent

$$y = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)}$$

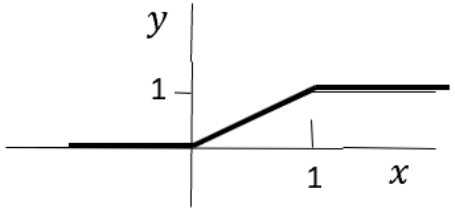
Hard Limiter



x<0, y=-1

x>=0, y=1

Ramping Function



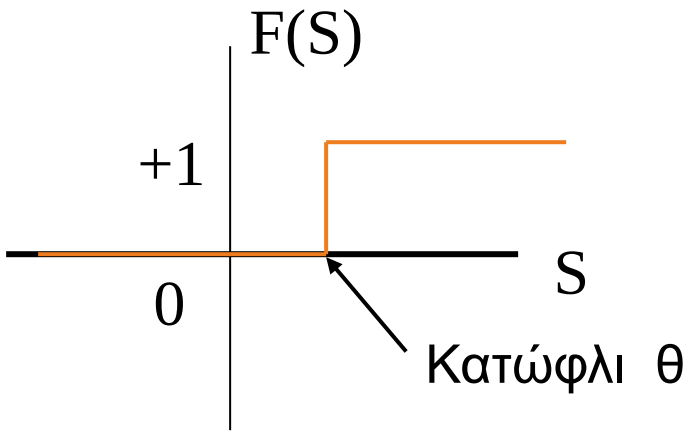
x<0, y=0

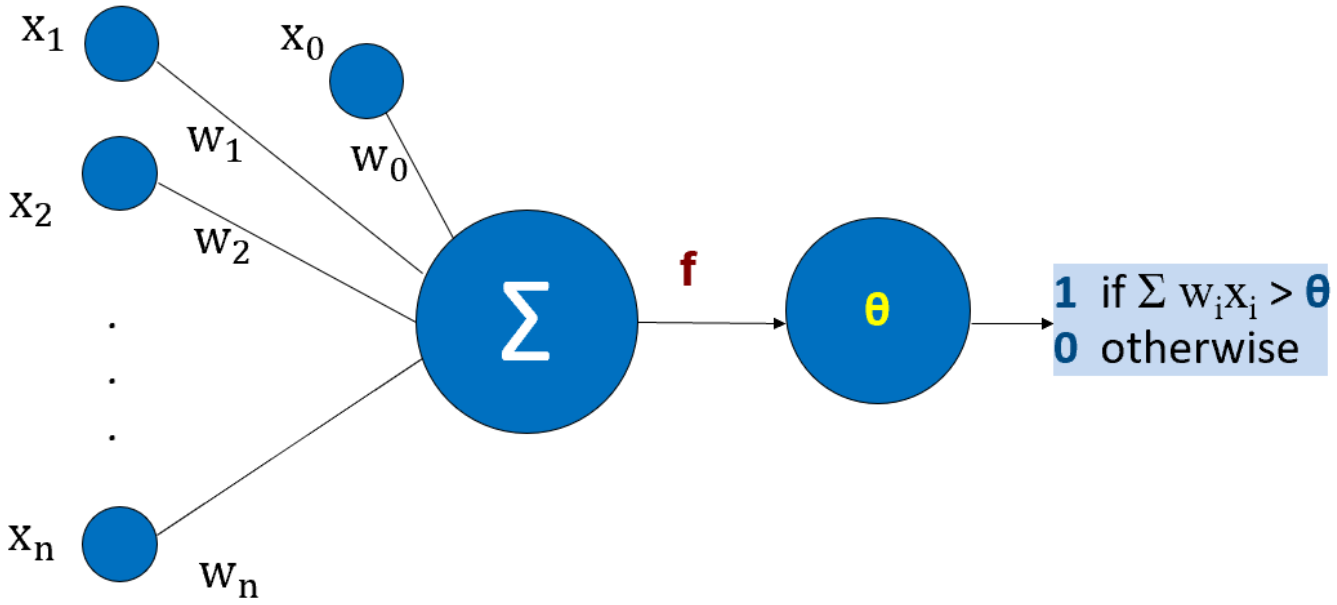
0<=x<=1, x=y

x>1, y=1

Activation

Βηματική (έξοδος 1 ή 0)





Feed Forward

Οι νευρώνες των διαφόρων στρωμάτων μπορεί να είναι:

- **Πλήρως συνδεδεμένοι (fully connected),**
- **Μερικώς συνδεδεμένοι (partially connected).**

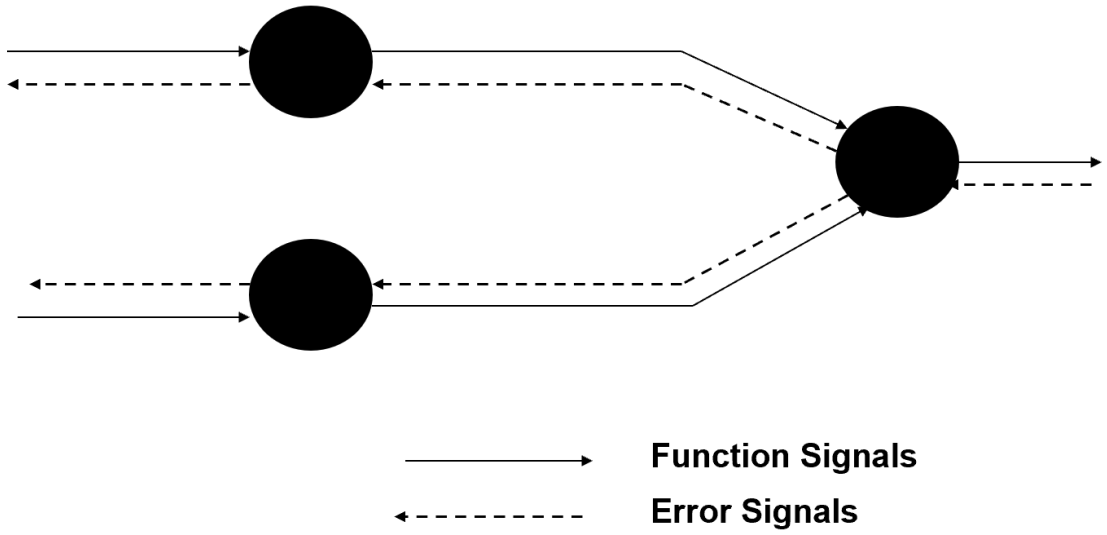
Τα ΤΝΔ χαρακτηρίζονται ως:

- **Δίκτυα με πρόσθια τροφοδότηση (feedforward),**
- **Δίκτυα με ανατροφοδότηση (feedback ή recurrent).**

Στην πλειοψηφία των εφαρμογών χρησιμοποιούνται δίκτυα απλής τροφοδότησης.

Backpropagation

- **Διαδικασία Backpropagation:** οι διαφορές μεταξύ του υπολογιζόμενου και του επιθυμητού αποτελέσματος λαμβάνονται υπ' όψιν και προπαγανδίζονται προς τα πίσω στις κρυμμένες μονάδες έτσι ώστε να καθορίσουν τις **απαραίτητες αλλαγές** (**κανόνας εκμάθησης**) στα βάρη σύνδεσης μεταξύ των μονάδων.
- **Κανόνες Εκμάθησης**
  - Κανόνας Δέλτα (Delta rule learning)
  - Αλγόριθμος ανάστροφης μετάδοσης λάθους (back propagation)
  - Ανταγωνιστική μάθηση (competitive learning)
  - Τυχαία μάθηση (random learning)
- Το δίκτυο εν συνεχεία εφαρμόζει εκ νέου τις συναρτήσεις μετάβασης για να υπολογίσει το νέο σφάλμα.
- Η διαδικασία της εκμάθησης περιλαμβάνει ένα πλήθος τέτοιων κύκλων και τερματίζει με τη μείωση του λάθους κάτω από ένα επιθυμητό όριο.



Backpropagation

- Steps
  - Initialize weights to small random numbers, associated with biases
  - Propagate the inputs forward (by applying activation function)
  - Backpropagate the error (by updating weights and biases)
  - Terminating condition (when error is very small, etc.)

Backpropagation

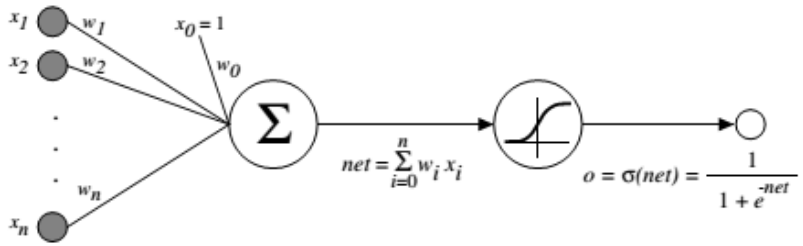
$$net_j = w_{j0} + \sum_i^n w_{ji} o_i$$

$$o_j = f_j(net_j)$$

$$\Delta w_{ji} = -\alpha \frac{\partial E}{\partial w_{ji}} = -\alpha \frac{\partial E}{\partial net_j} \frac{\partial net_j}{\partial w_{ji}} = \alpha \delta_j o_i$$

Credit assignment

$$\delta_j = -\frac{\partial E}{\partial net_j}$$



$\sigma(x)$ is the sigmoid function

$$\delta_j = -\frac{\partial E}{\partial o_j} \frac{\partial o_j}{\partial net_j} = -\frac{\partial E}{\partial o_j} f'(net_j)$$

$$E = \frac{1}{2}(t_j - o_j)^2 \Rightarrow \frac{\partial E}{\partial o_j} = -(t_j - o_j)$$

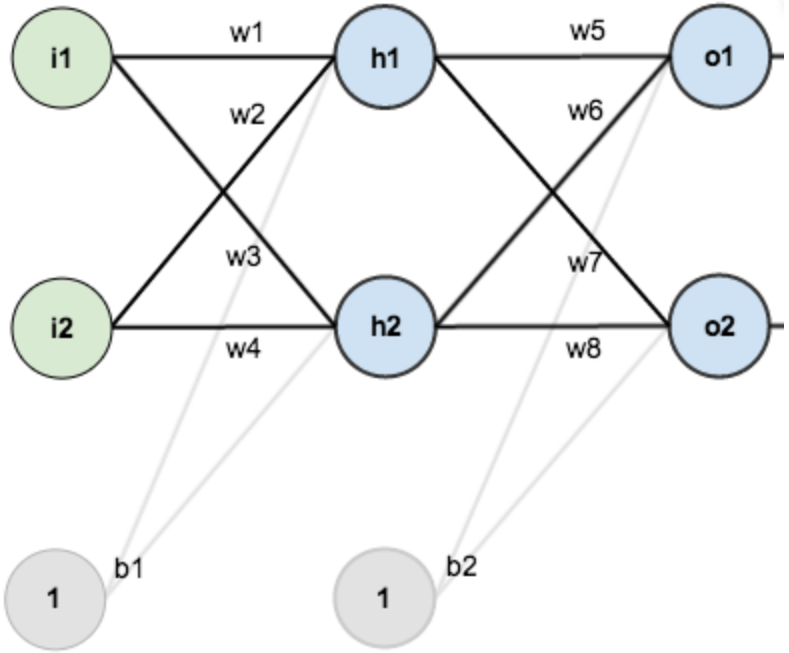
$$\delta_j = (t_j - o_j) f'(net_j)$$

$$w_{ji} \leftarrow w_{ji} + \Delta w_{ji}$$

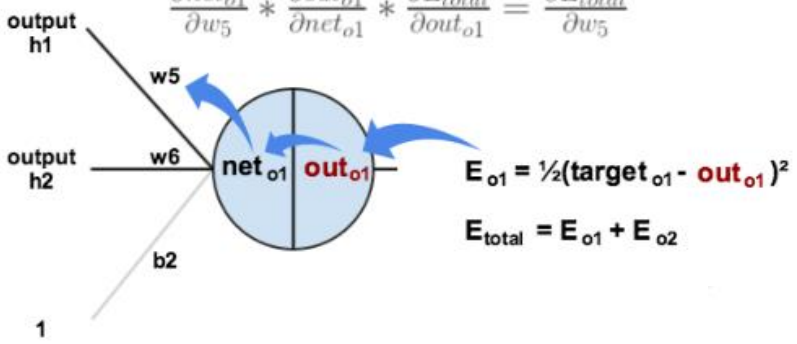
If the jth node is an output unit

Παράδειγμα

https://mattmazur.com/2015/03/17/a-step-by-step-backpropagation-example/

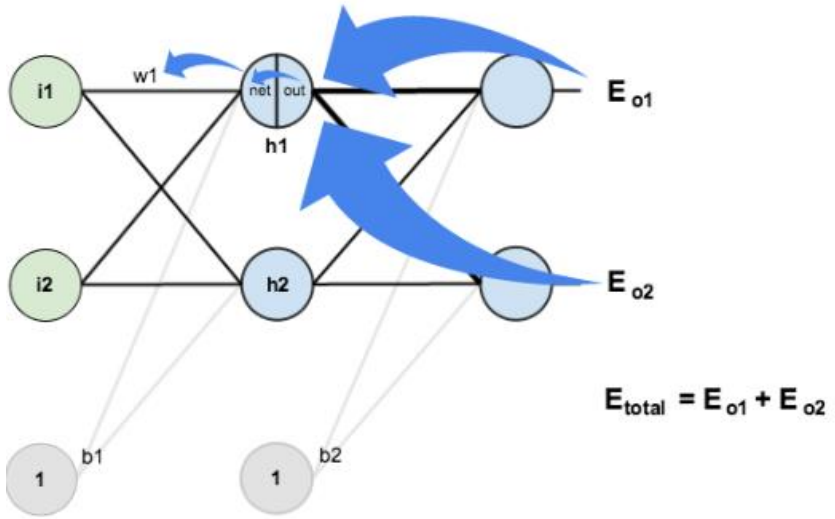


$$\frac{\partial net_{o1}}{\partial w_5} * \frac{\partial out_{o1}}{\partial net_{o1}} * \frac{\partial E_{total}}{\partial out_{o1}} = \frac{\partial E_{total}}{\partial w_5}$$



$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial out_{h1}} * \frac{\partial out_{h1}}{\partial net_{h1}} * \frac{\partial net_{h1}}{\partial w_1}$$

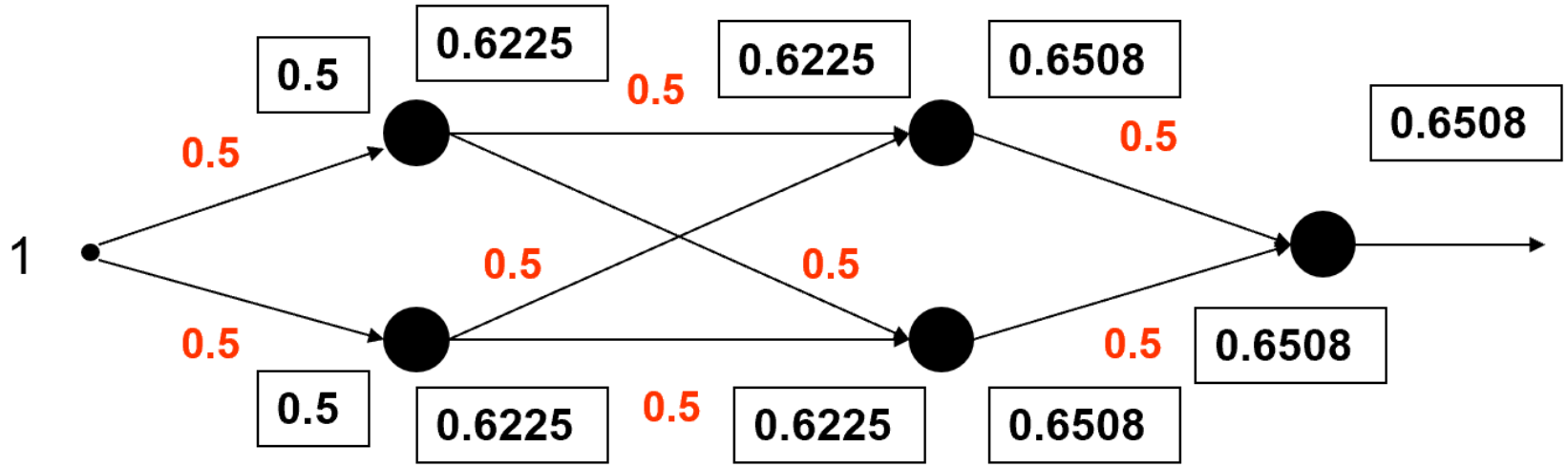
$$\frac{\partial E_{total}}{\partial out_{h1}} = \frac{\partial E_{o1}}{\partial out_{h1}} + \frac{\partial E_{o2}}{\partial out_{h1}}$$



Παράδειγμα

G1= (0.6225)(1-0.6225)(0.0397)(0.5)(2)=0.0093

G2= (0.6508)(1-0.6508)(0.3492)(0.5)=0.0397



Gradient of the neuron= **G** =slope of the transfer function×[Σ{(weight of the neuron to the next neuron) × (output of the neuron)}]

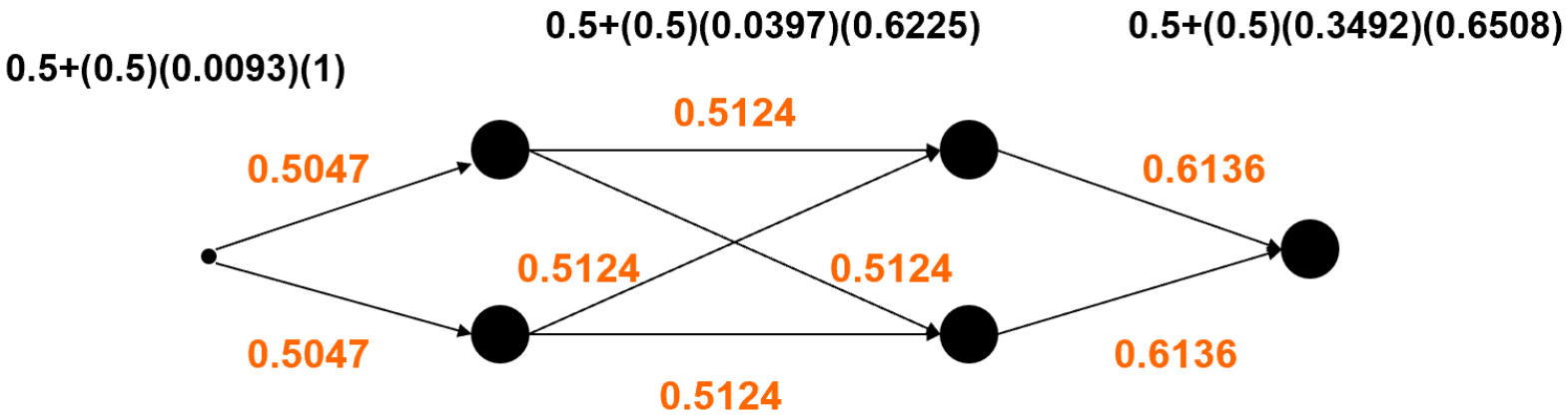
Gradient of the output neuron = slope of the transfer function × error

G3=(1)(0.3492)=0.3492

Error=1-0.6508=0.3492

Παράδειγμα

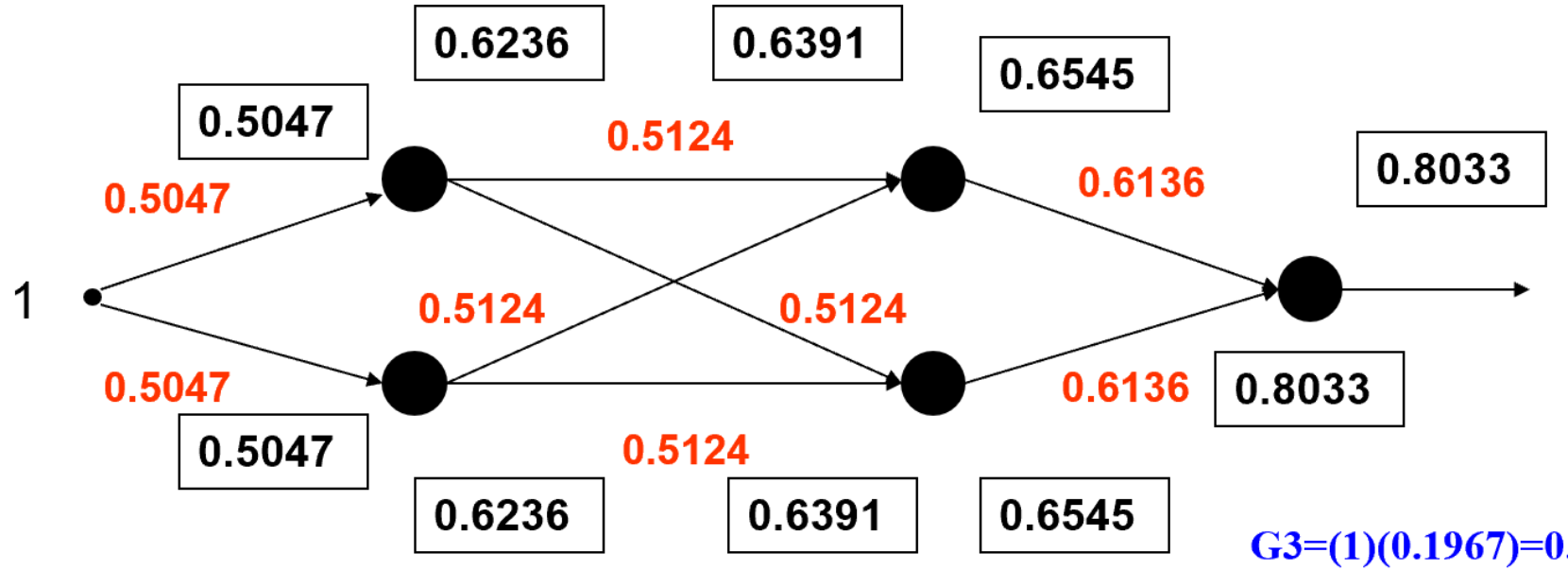
New Weight=Old Weight + {(learning rate)(gradient)(prior output)}



Παράδειγμα

G1= (0.6236)(1-0.6236)(0.5124)(0.0273)(2)=0.0066

G2= (0.6545)(1-0.6545)(0.1967)(0.6136)=0.0273

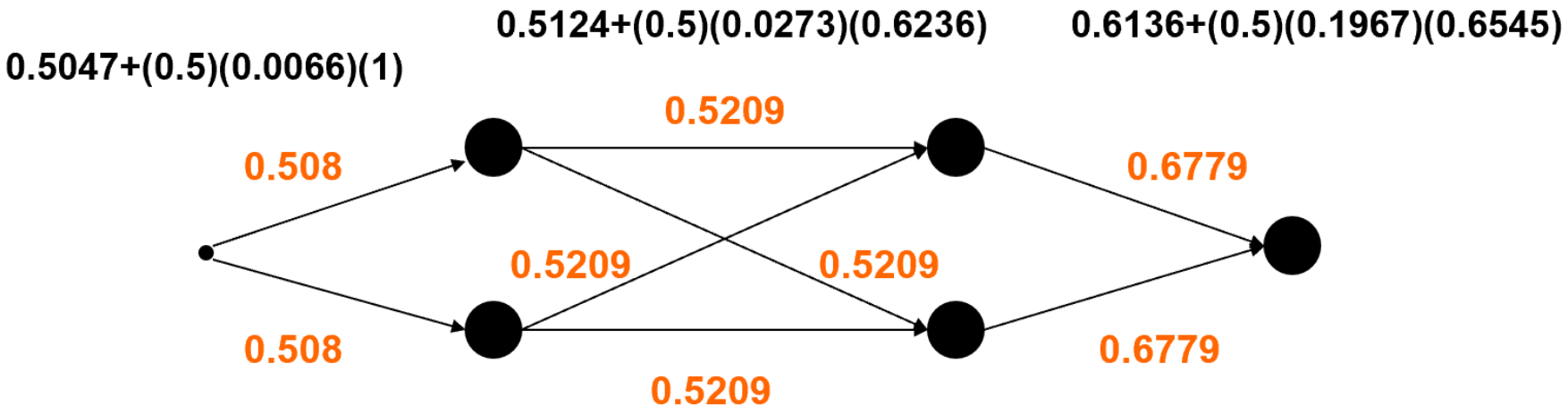


G3=(1)(0.1967)=0.1967

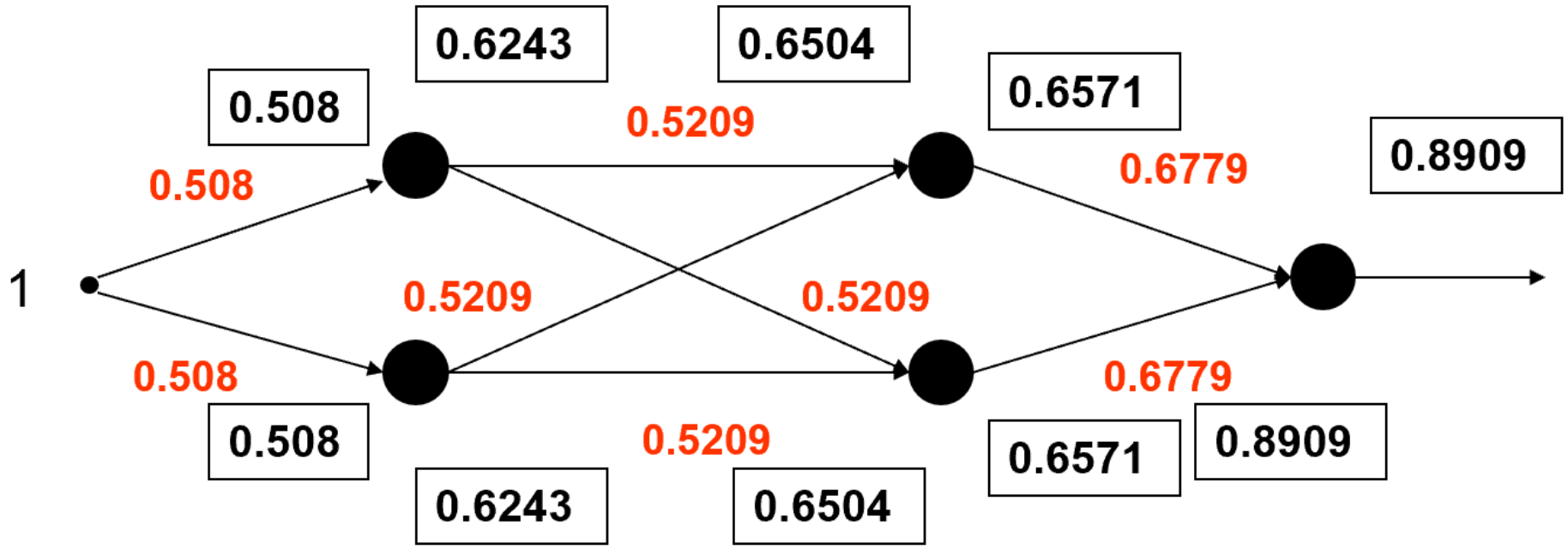
Error=1-0.8033=0.1967

Παράδειγμα

New Weight=Old Weight + {(learning rate)(gradient)(prior output)}



Παράδειγμα


1
0.508
0.508
0.6243
0.6504
0.5209
0.5209
0.5209
0.508
0.508
0.5209
0.6571
0.6779
0.6779
0.8909
0.6571
0.8909
0.6243
0.6504

Παράδειγμα

	Weights			Output	Expected (	Error
	w1	w2	w3			
Initial conditions	0.5	0.5	0.5	0.6508	1	0.3492
Pass 1 Update	0.5047	0.5124	0.6136	0.8033	1	0.1967
Pass 2 Update	0.508	0.5209	0.6779	0.8909	1	0.1091

W1: Weights from the input to the input layer
W2: Weights from the input layer to the hidden layer
W3: Weights from the hidden layer to the output layer

Παράδειγμα

https://medium.com/@SSiddhant/the-mathematics-of-neural-networks-a-complete-example-65f2b12cdea2

Σημειώσεις

- Cons
  - Μεγάλος χρόνος εκπαίδευσης
  - Απαιτείται ένας αριθμός παραμέτρων που συνήθως προσδιορίζονται καλύτερα εμπειρικά, π.χ. η τοπολογία ή η «δομή» του δικτύου.
  - Κακή ερμηνεία: Δύσκολη η ερμηνεία του συμβολικού νοήματος πίσω από τα μαθημένα βάρη και των «κρυμμένων μονάδων» στο δίκτυο
- Pros
  - Υψηλή ανοχή σε θορυβώδη δεδομένα
  - Δυνατότητα ταξινόμησης μη εκπαιδευμένων μοτίβων
  - Κατάλληλο για εισόδους και εξόδους συνεχούς αξίας
  - Επιτυχής σε μια σειρά από δεδομένα πραγματικού κόσμου, π.χ. χειρόγραφα γράμματα
  - Οι αλγόριθμοι είναι εγγενώς παράλληλοι
  - Πρόσφατα αναπτύχθηκαν τεχνικές για την εξαγωγή κανόνων από εκπαιδευμένα νευρωνικά δίκτυα