

Data Mining

Classification: Alternative Techniques

Lecture Notes for Chapter 5

(cont.)

Classification Problem

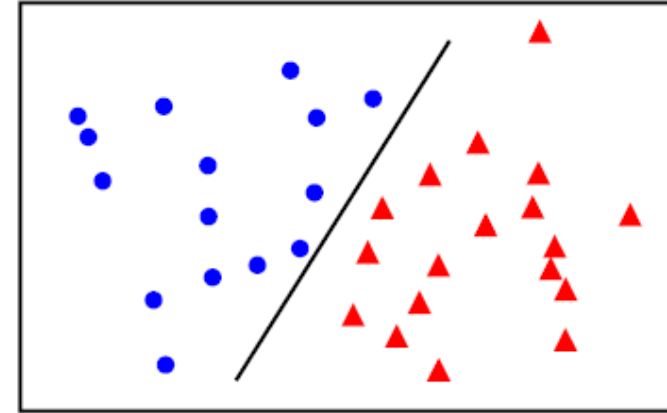
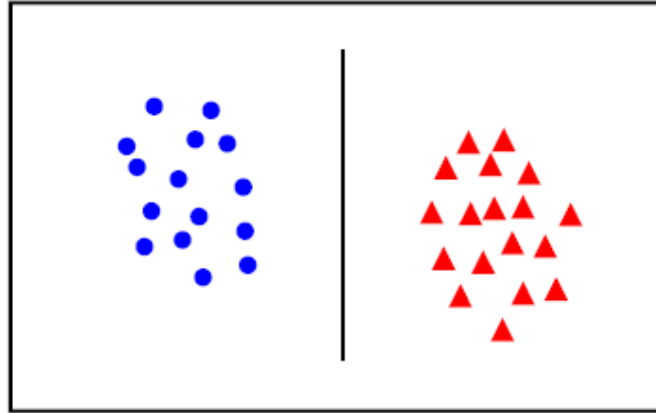
- Πρόβλημα μάθησης με επίβλεψη
(**Supervised learning**)
- Δεδομένα του συνόλου εκπαίδευσης $X = \{(x_n, y_n)\}_{n=1}^N$ αποτελούμενα από ζεύγη
 - σημείων σε ένα χώρο ($x_n \in R^d$) και
 - ετικέτας ή πρόβλεψης y_n .
- **Στόχος** να φτιάξουμε μία συνάρτηση $f(x)$:
$$\forall x_n \in X \quad f(x_n) = y_n$$

Linear classifiers (Γραμμικοί ταξινομητές)

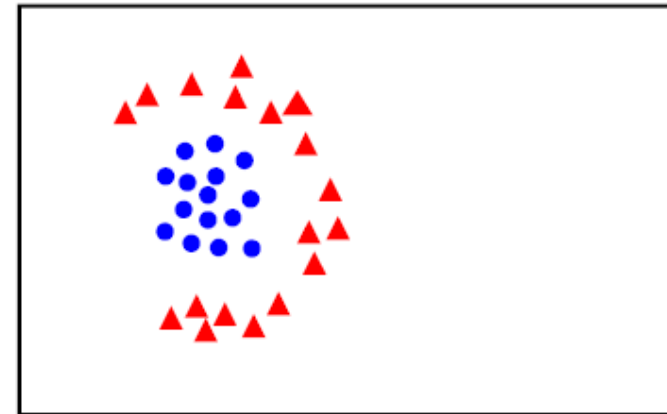
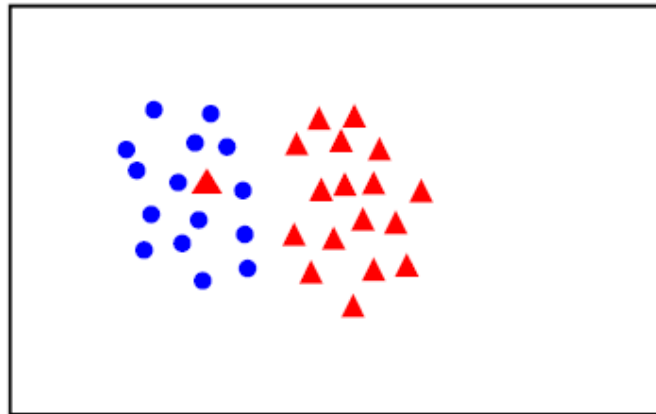
- Έστω πρόβλημα ταξινόμησης σε 2 κατηγορίες ($K=2$), ετικέτες Ω_1 και Ω_2 (**binary classification**)
- Κατασκευή συνάρτησης $f(x;w)$ για να αποφασίσουμε την κατηγορία ενός δείγματος x
- w είναι οι παράμετροι της μεθόδου-συνάρτησης
 - $if \quad f(x_n;w) \geq 0 \quad then \quad x_n \in \Omega_1$
- Κανόνας απόφασης: *else*
 - $if \quad f(x_n;w) < 0 \quad then \quad x_n \in \Omega_2$

Πρόβλημα δυαδικής ταξινόμησης (binary classification)

Γραμμική
περίπτωση



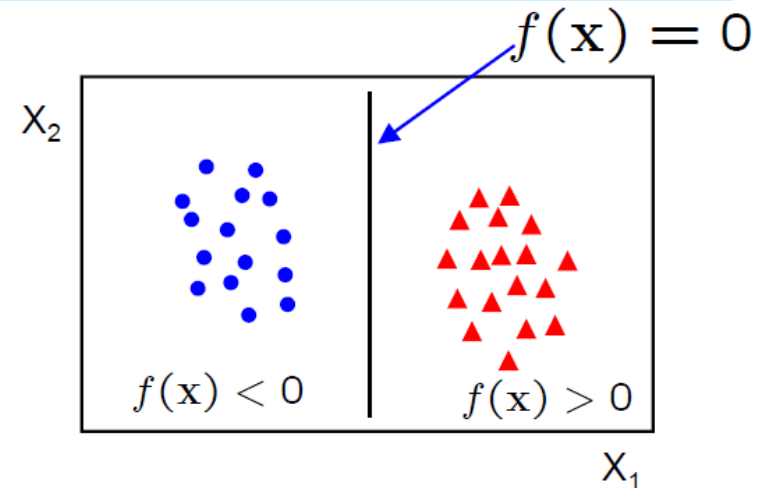
Μη γραμμική
περίπτωση



Το γραμμικό μοντέλο (Linear model)

- Υποθέτουμε ότι

$$f(x, w) = w_0 + \sum_{j=1}^d w_j x_j = w_0 + w^T x$$



όπου $w = (w_1, \dots, w_d)$ το διάνυσμα των βαρών
 w_0 πόλωση (bias) ο σταθερός όρος

- Ορίζει μία γραμμική διαχωριστική επιφάνεια μεταξύ των 2 κατηγοριών

- Στη περίπτωση όπου οι δυαδικές κατηγορίες έχουν τιμές $y_n \in \{+1, -1\}$, τότε:

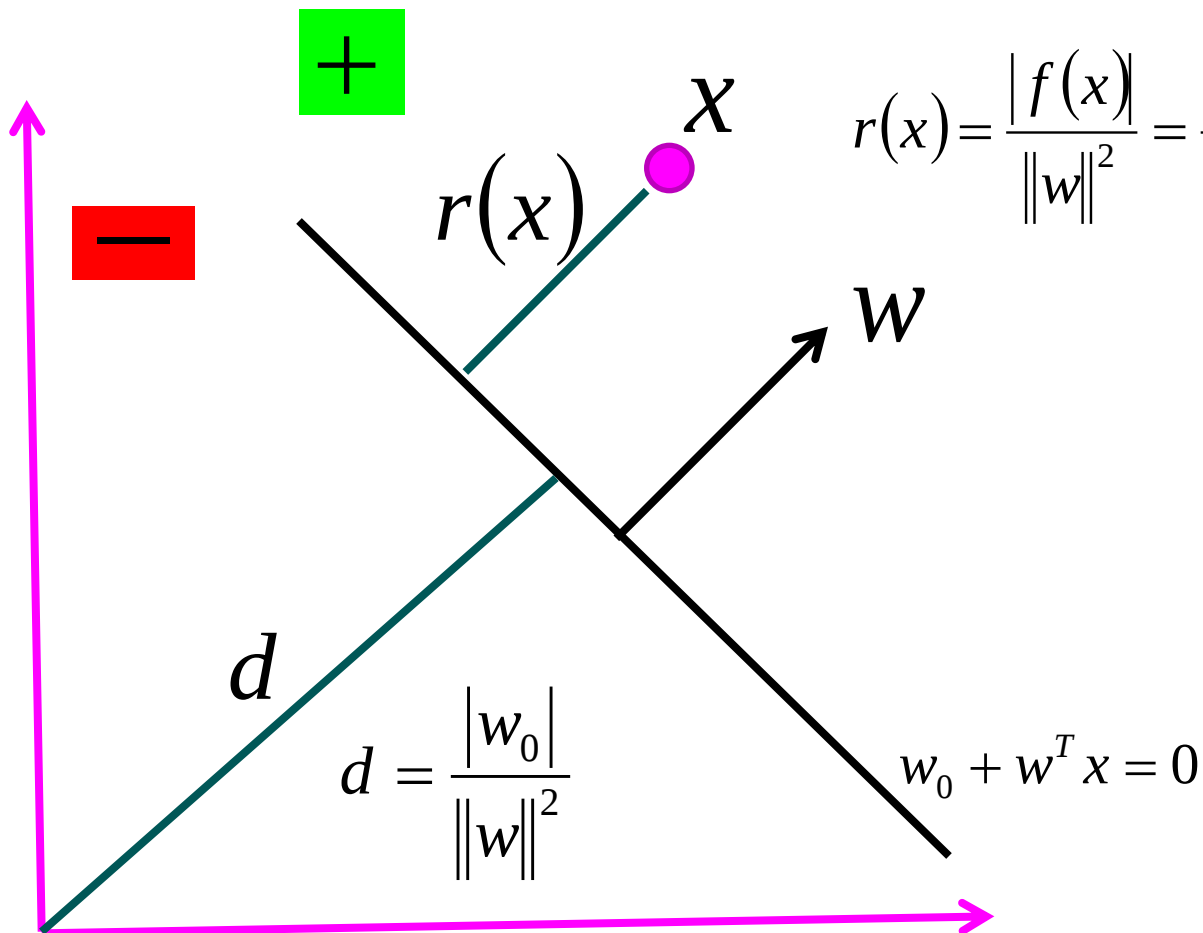
$$\text{if } y_n = +1 \quad \Rightarrow \quad w^T x_n + w_0 \geq 0$$

else

$$\text{if } y_n = -1 \quad \Rightarrow \quad w^T x_n + w_0 < 0$$

- Εναλλακτικά:

$$\boxed{\forall x_n \in X \quad y_n (w^T x_n + w_0) \geq 0}$$



Πρόβλημα: εύρεση καταλληλότητας του γραμμικού μοντέλου

- Σύνολο εκπαίδευσης $X = \{(x_n, y_n)\}_{n=1}^N$
όπου $x_n \in R^d$ και $y_n \in \{+1, -1\}$
- **Εκτίμηση** των παραμέτρων $\{w, w_0\}$ του γραμμικού μοντέλου, έτσι ώστε:

$$\forall n \quad y_n (w^T x_n + w_0) \geq 0$$

[1]. Ο αλγόριθμος Perceptron (Rosenblatt 1958)

To Perceptron criterion:

- Αν το πρότυπο x_n ταξινομείται **σωστά**
(δηλ. $y_n(w^T x_n + w_0) \geq 0$) τότε καμία ενέργεια.
- Αν το πρότυπο ταξινομείται **λάθος** ($y_n(w^T x_n + w_0) < 0$)
τότε η ποσότητα $-y_n(w^T x_n + w_0)$ εκφράζει την
συνεισφορά της λάθος εκτίμησης.

To Perceptron criterion:

Συνάρτηση καταλληλότητας (**ελαχιστοποίησης**):

$$E(w) = - \sum_{x_n \in \mathbf{M}} y_n (w^T x_n + w_0)$$

όπου $\mathbf{M} = \{x_n : y_n (w^T x_n + w_0) < 0\}$

το σύνολο των **λάθος ταξινομημένων** προτύπων

Ο αλγόριθμος Perceptron

- Αρχικοποίηση των παραμέτρων $\{w, w_0\}$
- Επαναληπτικά
 1. Τυχαία επιλογή ενός σημείου x_n του συνόλου εκπαίδευσης
 2. If $y_n (w^T x_n + w_0) \geq 0$ (**correct** classification) go to 1
 3. If $y_n (w^T x_n + w_0) < 0$ (**wrong** classification) then
update weights
$$w^{(new)} = w^{(old)} + \eta y_n x_n$$
$$w_0^{(new)} = w_0^{(old)} + \eta y_n$$
- Μέχρι σύγκλιση ή μέγιστο αριθμό επαναλήψεων

Ο αλγόριθμος Perceptron (συν.)

Παρατήρηση: Σε κάθε βήμα επανάληψης το σφάλμα ενός λάθος ταξινομημένου σημείου μειώνεται. Γιατί;;;

$$\begin{aligned} -y_n \left(\mathbf{w}^{(new)} \right)^T \tilde{\mathbf{x}}_n &= -y_n \left(\mathbf{w}^{(old)} + \eta y_n \tilde{\mathbf{x}}_n \right)^T = \\ &= -y_n \left(\mathbf{w}^{(old)} \right)^T \tilde{\mathbf{x}}_n - \eta y_n \left(y_n \tilde{\mathbf{x}}_n \right)^T \tilde{\mathbf{x}}_n = \\ &= -y_n \left(\mathbf{w}^{(old)} \right)^T \tilde{\mathbf{x}}_n - \underbrace{\eta y_n^2 \tilde{\mathbf{x}}_n^T \tilde{\mathbf{x}}_n}_{>0} < -y_n \left(\mathbf{w}^{(old)} \right)^T \tilde{\mathbf{x}}_n \end{aligned}$$

Παρόλ' αυτά **δεν** εξασφαλίζεται η μείωση του συνολικού σφάλματος σε κάθε επανάληψη!

[2]. Μέθοδος των ελαχίστων τετραγώνων (Least Squares)

- Συνάρτηση του γραμμικού μοντέλου

$$f(x_n) = w_0 + w^T x_n$$

- Η οποία προβλέπει την κατηγορία (y_n) των προτύπων x_n
- Ορίζουμε συνάρτηση τετραγωνικού σφάλματος

$$E(w) = \frac{1}{2} \sum_{n=1}^N (f(x_n) - y_n)^2 = \frac{1}{2} \sum_{n=1}^N (w^T x_n + w_0 - y_n)^2$$

- Εναλλακτική μορφή της συνάρτησης τετρ. σφάλματος:

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N (f(x_n) - y_n)^2 = \frac{1}{2} \|\Phi \mathbf{w} - \mathbf{Y}\|^2$$

- ΟΠΟΥ

$$\mathbf{Y} = (y_1, y_2, \dots, y_N)$$

$$\mathbf{w} = (w_0, w_1, \dots, w_d)$$

$$\Phi = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1d} \\ 1 & x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{N1} & x_{N2} & \cdots & x_{Nd} \end{bmatrix}$$

- Έτσι έχουμε:

$$\begin{aligned} E(\mathbf{w}) &= \frac{1}{2} (\mathbf{\Phi w} - \mathbf{Y})^T (\mathbf{\Phi w} - \mathbf{Y}) = \\ &= \frac{1}{2} (\mathbf{w}^T \mathbf{\Phi}^T \mathbf{\Phi w} - 2\mathbf{w}^T \mathbf{\Phi}^T \mathbf{Y} + \mathbf{Y}^T \mathbf{Y}) \end{aligned}$$

- Πρόβλημα εκτίμησης (ελαχιστοποίησης)

$$\text{find } \hat{\mathbf{w}} : \min \{E(\mathbf{w})\}$$

- Επίλυση

$$\frac{\partial E(\mathbf{w})}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{\Phi}^T \mathbf{\Phi} \hat{\mathbf{w}} - \mathbf{\Phi}^T \mathbf{Y} = 0 \Rightarrow \mathbf{\Phi}^T \mathbf{\Phi} \hat{\mathbf{w}} = \mathbf{\Phi}^T \mathbf{Y}$$

$$\hat{\mathbf{w}} = \left(\mathbf{\Phi}^T \mathbf{\Phi} \right)^{-1} \mathbf{\Phi}^T \mathbf{Y}$$

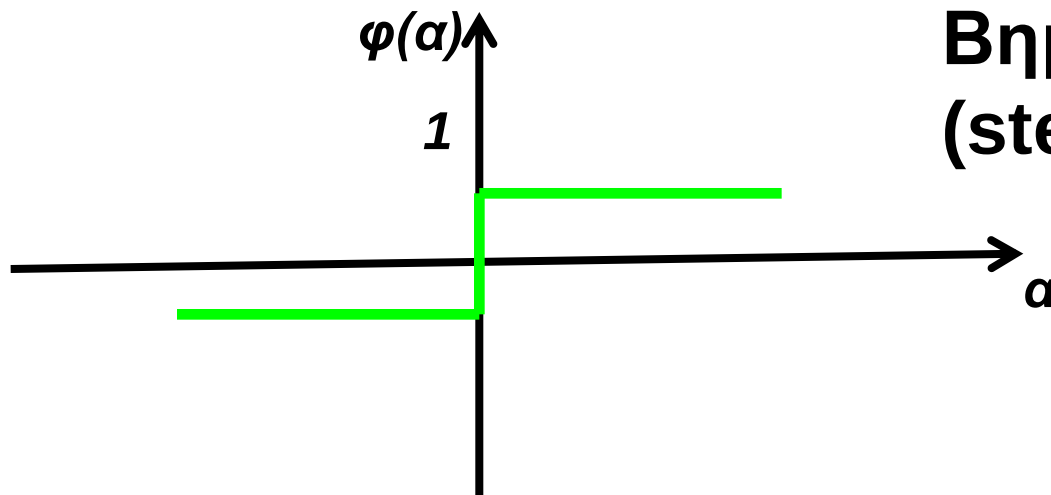
- Εκτιμητής των ελαχίστων τετραγώνων

- $\mathbf{\Phi}^{-*} = \left(\mathbf{\Phi}^T \mathbf{\Phi} \right)^{-1} \mathbf{\Phi}^T$ Ψευδοαντίστροφος του $\mathbf{\Phi}$

[3]. Συνάρτηση ενεργοποίησης

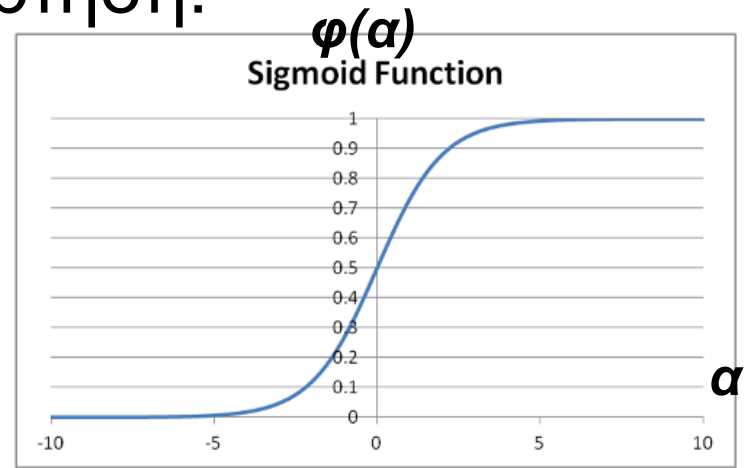
- Γραμμικό μοντέλο: $f(x_n) = w_0 + w^T x_n$
- Απόφαση με βάση το πρόσημο (συνάρτηση ενεργοποίησης):

$$\varphi(x_i) = \text{sign}(w_0 + w^T x_n) = \begin{cases} 1 & w_0 + w^T x_n \geq 0 \\ -1 & w_0 + w^T x_n < 0 \end{cases}$$

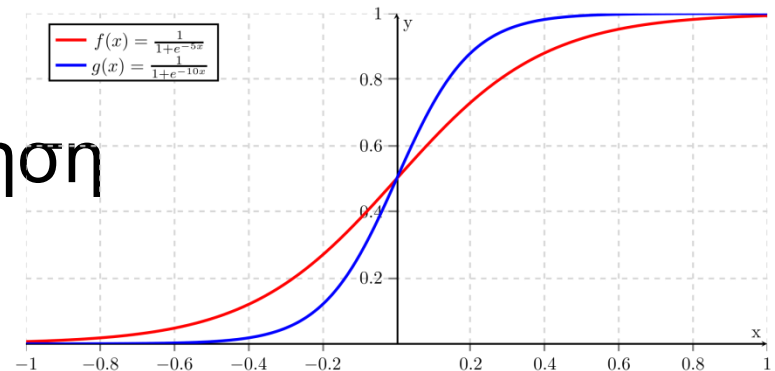


- Άλλες (μη γραμμικές) συναρτήσεις ενεργοποίησης
- Λογιστική ή σιγμοειδής συνάρτηση:

$$\varphi(a) = \frac{1}{1 + e^{-\lambda a}}$$



- λ : κλίση της καμπύλης
- Για $\lambda \rightarrow \infty$ βηματική συνάρτηση

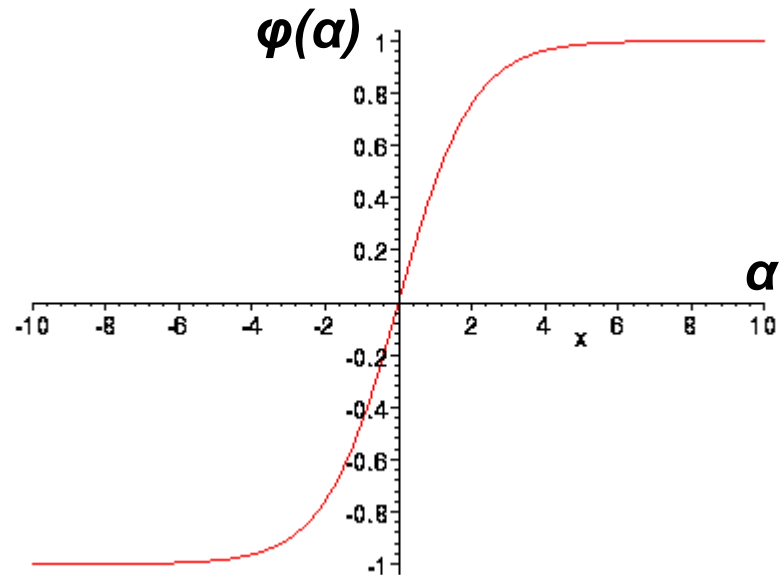


- Υπερβολική εφαπτομένη:

$$\varphi(a) = \tanh(a) = \frac{1 - e^{-\lambda a}}{1 + e^{-\lambda a}}$$

- Πρώτη παράγωγος της σιγμοειδούς

$$\frac{d\varphi(a)}{da} = \lambda \varphi(a)(1 - \varphi(a))$$



[4]. Εκτίμηση γραμμικού μοντέλου με συναρτήσεις ενεργοποίησης

- Η συνάρτηση απόφασης είναι η σιγμοειδής:

$$f(x_n) = \varphi(w_0 + w^T x_n) \quad \text{όπου} \quad \varphi(a) = \frac{1}{1 + e^{-\lambda a}}$$

- Συνάρτηση τετραγωνικού σφάλματος

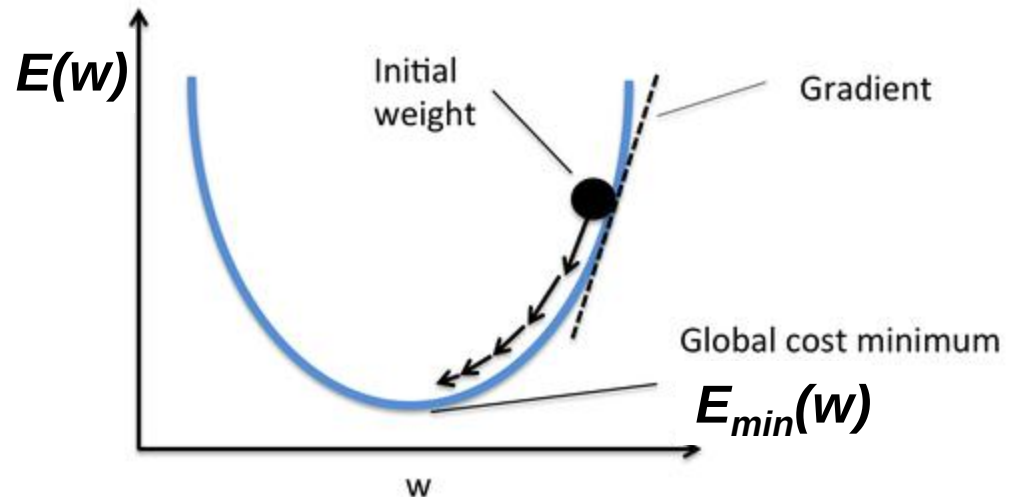
$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N (f(x_n) - y_n)^2$$

- Πρόβλημα ελαχιστοποίησης

$$\text{Find } \hat{\mathbf{w}} : \min\{E(\mathbf{w})\}$$

- Ο αλγόριθμος βελτιστοποίησης «**βαθμωτής πτώσης**» (**gradient descent**)

Επαναληπτική διαδικασία ενημέρωσης των παραμέτρων κατά την οποία η ενημέρωση πραγματοποιείται στην κατεύθυνση της αρνητικής κλίσης



$$w^{(new)} = w^{(old)} - \eta \frac{\partial E(w)}{\partial w}$$

- Εφαρμογή της μεθόδου βελτιστοποίησης στο πρόβλημα ελαχιστοποίησης της συνάρτησης σφάλματος

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \left(\varphi(w^T x_n + w_0) - y_n \right)^2$$

$$\begin{aligned} \frac{\partial E(w)}{\partial w_j} &= \sum_{n=1}^N \left(\varphi(w^T x_n + w_0) - y_n \right) \frac{\partial \varphi(w^T x_n + w_0)}{\partial w_j} = \\ &= \sum_{n=1}^N \delta_n \frac{\partial \varphi(a_n)}{\partial a_n} \frac{\partial a_n}{\partial w_j} \end{aligned}$$


- Για $j=0$ (σταθερός όρος)

$$\frac{\partial E(w)}{\partial w_0} = \lambda \sum_{n=1}^N \delta_n \varphi(a_n)(1 - \varphi(a_n))$$

- Για $j=\{1, \dots, d\}$

$$\frac{\partial E(w)}{\partial w_j} = \lambda \sum_{n=1}^N \delta_n \varphi(a_n)(1 - \varphi(a_n))x_j$$

Αλγόριθμος εκπαίδευσης (εκτίμησης) βαρών

- Initial ($t=0$): $\{w_0^{(0)}, w_1^{(0)}, \dots, w_d^{(0)}\}$

$$E^{(0)}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \left(\varphi \left(w_0^{(0)} + \sum_{j=1}^d w_j^{(0)} x_{nj} \right) - y_n \right)^2$$

- repeat

$$w_0^{(t+1)} = w_0^{(t)} - \eta \frac{\partial E(\mathbf{w})}{\partial w_0} = w_0^{(t)} - \eta \sum_{n=1}^N \delta_n \varphi(a_n) (1 - \varphi(a_n))$$

$$w_j^{(t+1)} = w_j^{(t)} - \eta \frac{\partial E(\mathbf{w})}{\partial w_j} = w_j^{(t)} - \eta \sum_{n=1}^N \delta_n \varphi(a_n) (1 - \varphi(a_n)) x_{nj}$$

$$E^{(t+1)}(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \left(\varphi \left(w_0^{(t+1)} + \sum_{j=1}^d w_j^{(t+1)} x_{nj} \right) - y_n \right)^2$$

$$t = t + 1$$

- until:

$$\Delta E = |E^{(t)}(\mathbf{w}) - E^{(t+1)}(\mathbf{w})| < \varepsilon$$

Παρατηρήσεις

- Κανόνας **delta**, όπου

$$\delta_n = \varphi(a_n) - y_n = \varphi(w_0 + w^T x_n) - y_n$$

- Η ποσότητα δ_n είναι το **αλγεβρικό σφάλμα** με βάση το οποίο γίνεται η ενημέρωση των βαρών.
- Ένα πρότυπο με μεγάλο σφάλμα δ_n έχει μεγάλη συνεισφορά στη **διόρθωση** των βαρών.

[5]. Επέκταση για πολλές κατηγορίες $K > 2$

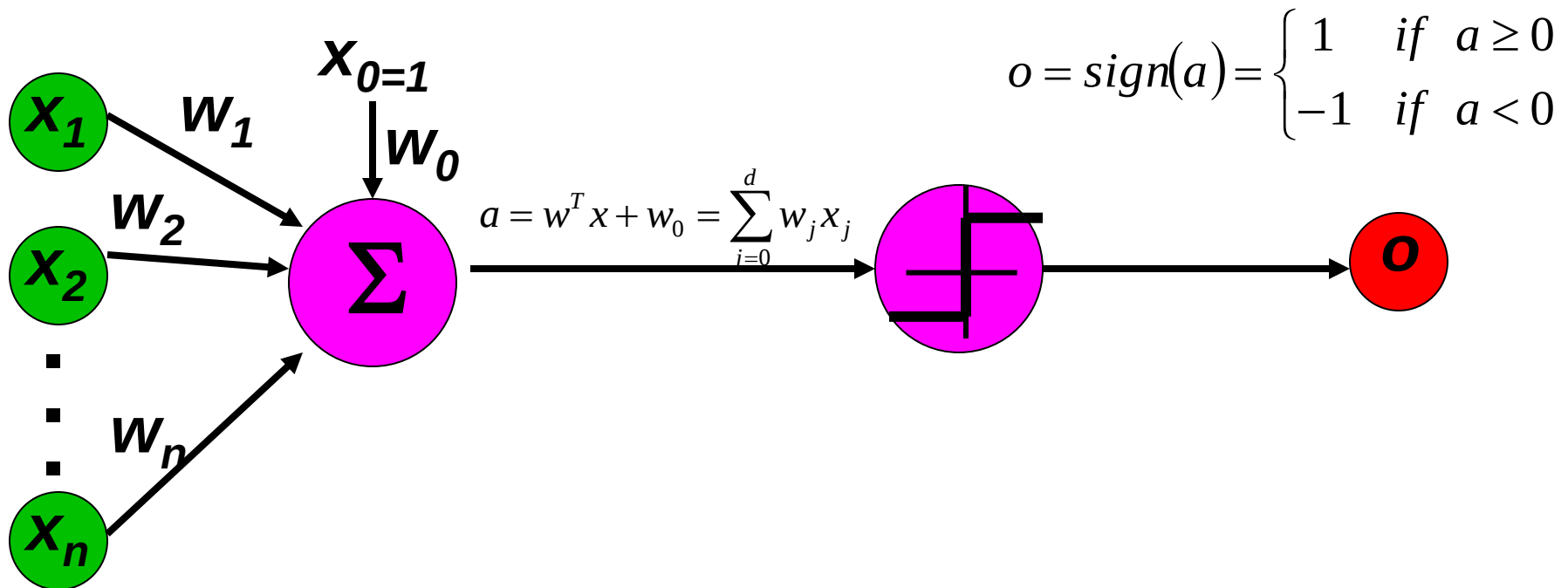
Δύο (2) παραλλαγές

- Κατασκευή ενός perceptron για κάθε μία κατηγορία j
 - Όλα τα πρότυπα της υπό-εξέταση κατηγορίας έχουν ετικέτα +1, ενώ όλων των υπολοίπων $K-1$ κατηγοριών θεωρούνται «αρνητικά» παραδείγματα κατηγορίας -1.
 - Εκπαίδευση των K perceptrons και κατασκευή των συναρτήσεων $f_j(x) = \varphi\left(w_{j0} + \sum_{i=1}^d w_{ji}x_i\right)$
 - Σε ένα άγνωστο πρότυπο x δίνεται η κατηγορία του perceptron με την μεγαλύτερη τιμή εξόδου.

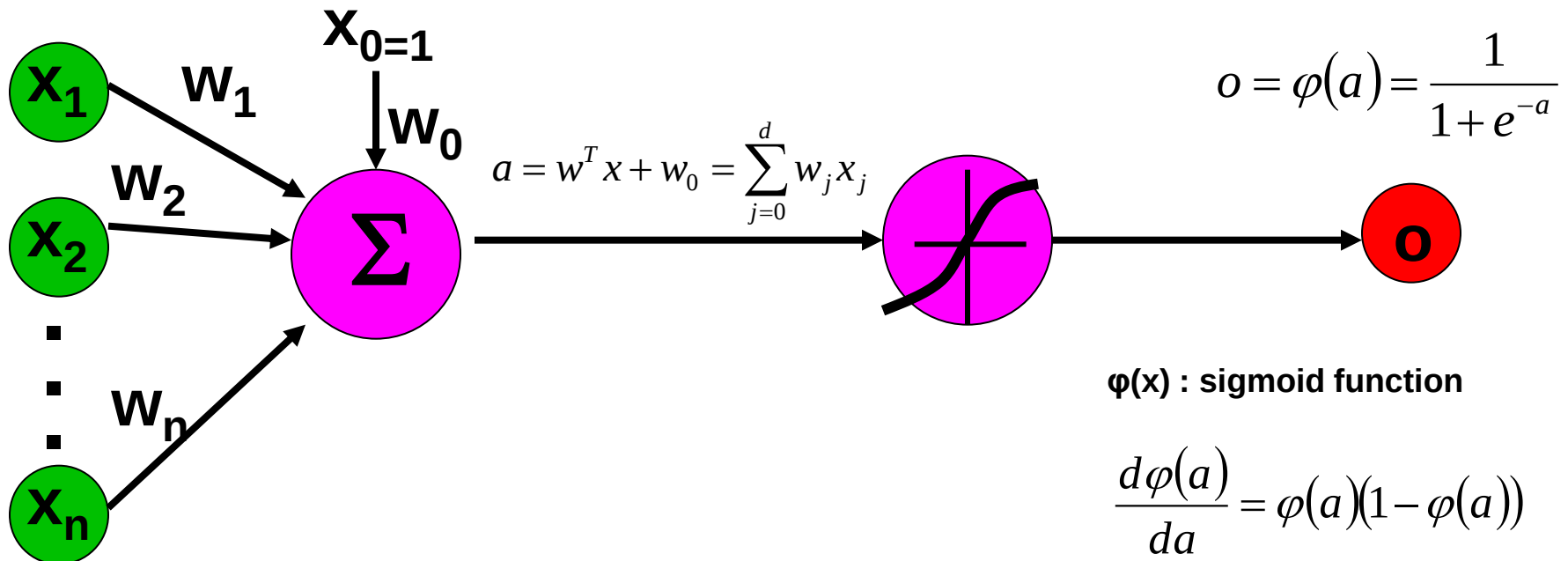
$$\max_{j=1,\dots,K} f_j(x)$$

- Ένα perceptron για κάθε ζεύγος κατηγοριών (j, k)
 - Η εκπαίδευση περιορίζεται μεταξύ των προτύπων των δύο κατηγοριών.
 - Εκπαίδευση των $K^*(K-1)$ perceptrons και κατασκευή των συναρτήσεων
$$f_{jk}(x) = \varphi \left(w_0^{(j,k)} + \sum_{i=1}^d w_i^{(j,k)} x_i \right)$$
 - Σε ένα άγνωστο πρότυπο x δίνεται η κατηγορία με τις περισσότερες νίκες (votes) μεταξύ των perceptrons.
 - Σε περίπτωση ισοπαλίας απόφαση με βάση τις τιμές των συναρτήσεων των γραμμικών μοντέλων.

[6]. Γραφική αναπαράσταση μιας μονάδας Perceptron

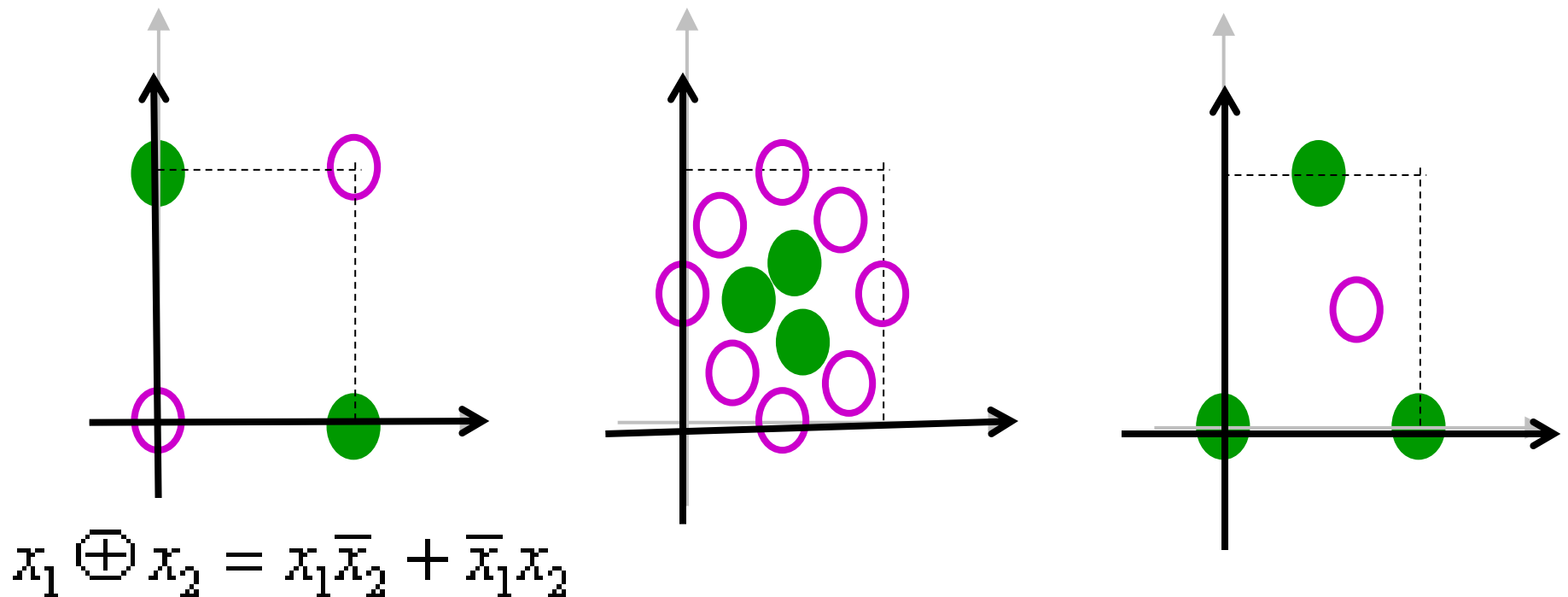


Σιγμοειδής μονάδας Perceptron

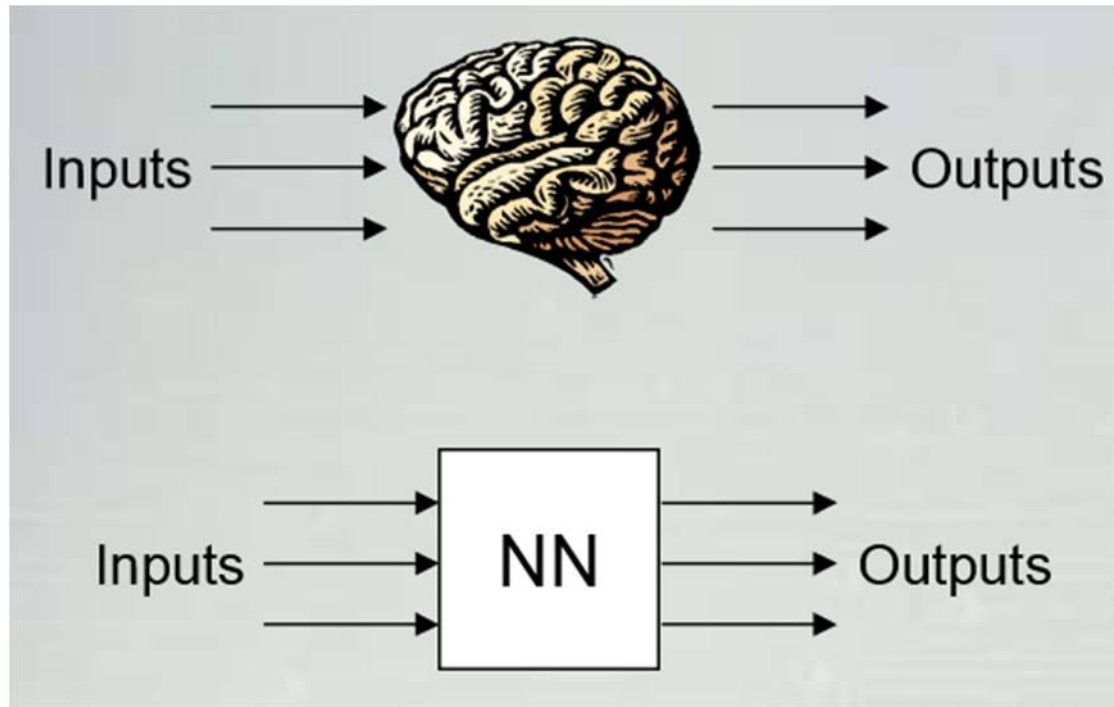


[7]. Περιορισμοί της μονάδας Perceptron

- Αδυναμία σε περιπτώσεις με μη-γραμμικά διαχωρίσιμα δεδομένα (π.χ. XOR)

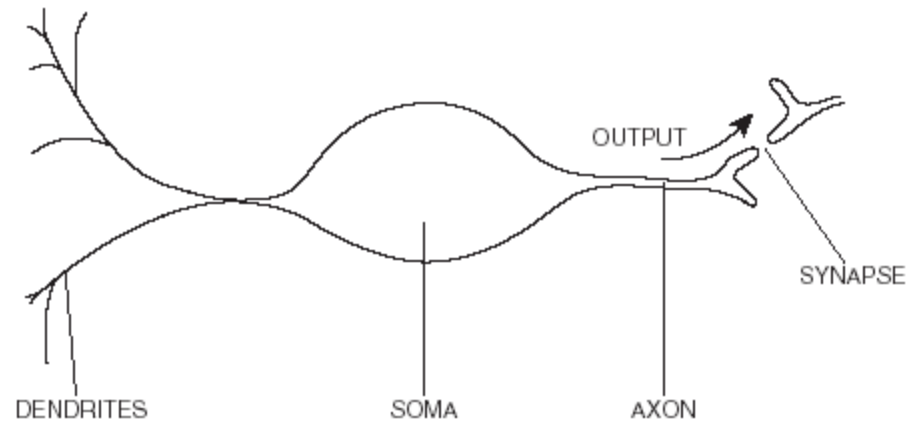


Νευρωνικά Δίκτυα (Neural Networks)



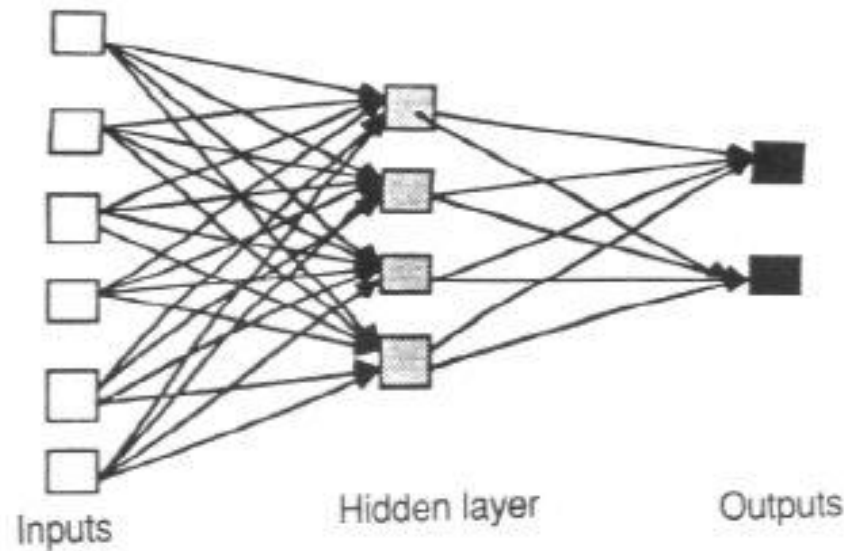
- Υπολογιστικές μηχανές που εκτελούν πολύπλοκες πράξεις και επεξεργάζονται δεδομένα.

Βιολογικός νευρώνας



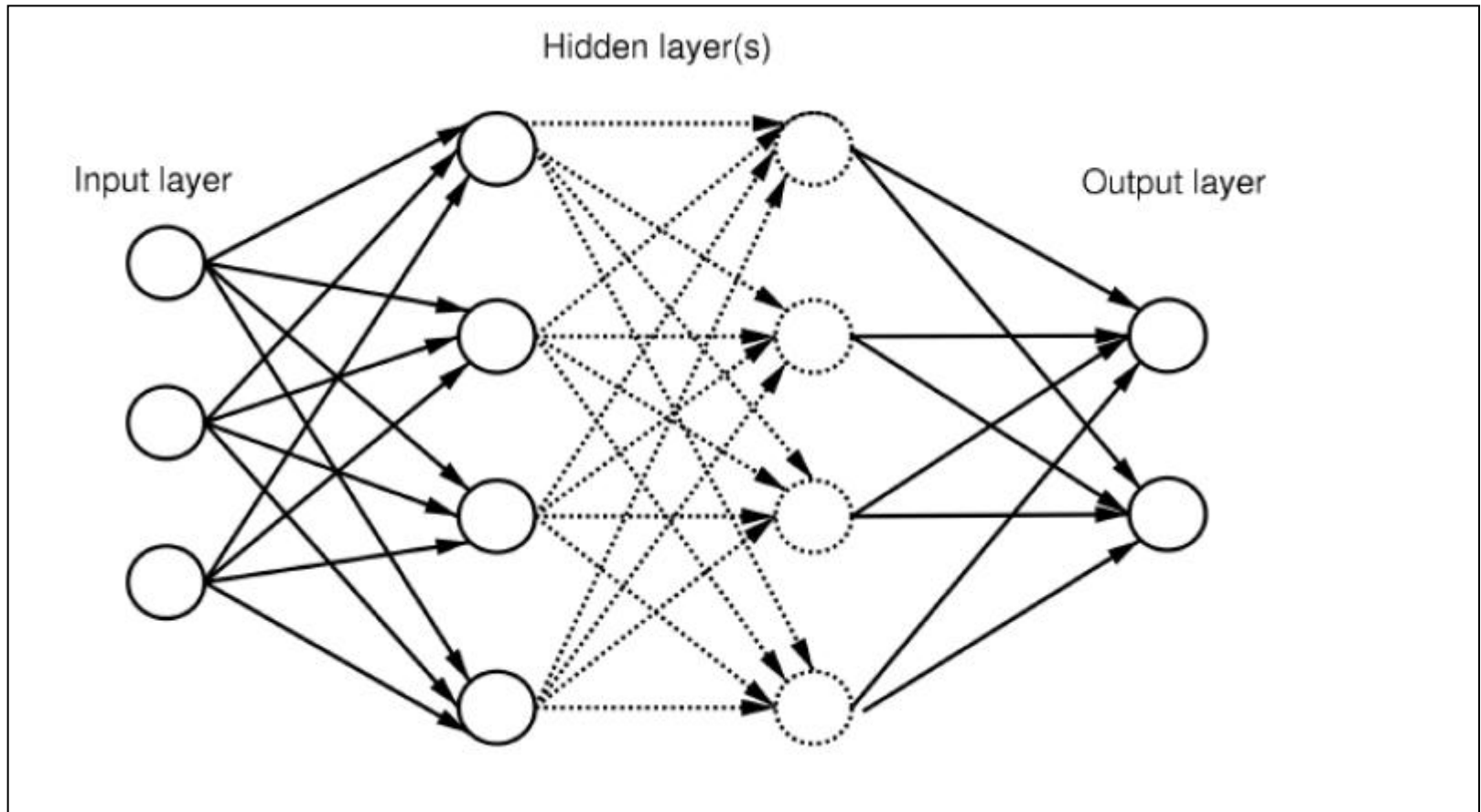
- Επεξεργαστική μονάδα του ανθρώπινου εγκεφάλου
- Οι **δενδρίτες** δέχονται την ερέθισμα (είσοδο) και πυροδοτούν το **σώμα** νευρώνα (neuron fires).
- Το σήμα που παράγεται μεταφέρεται μέσω του **άξονα** στις **συνάψεις** και διοχετεύεται σε άλλους νευρώνες.

[1]. Πολυεπίπεδο Perceptron (Multilayer Perceptron – MLP)

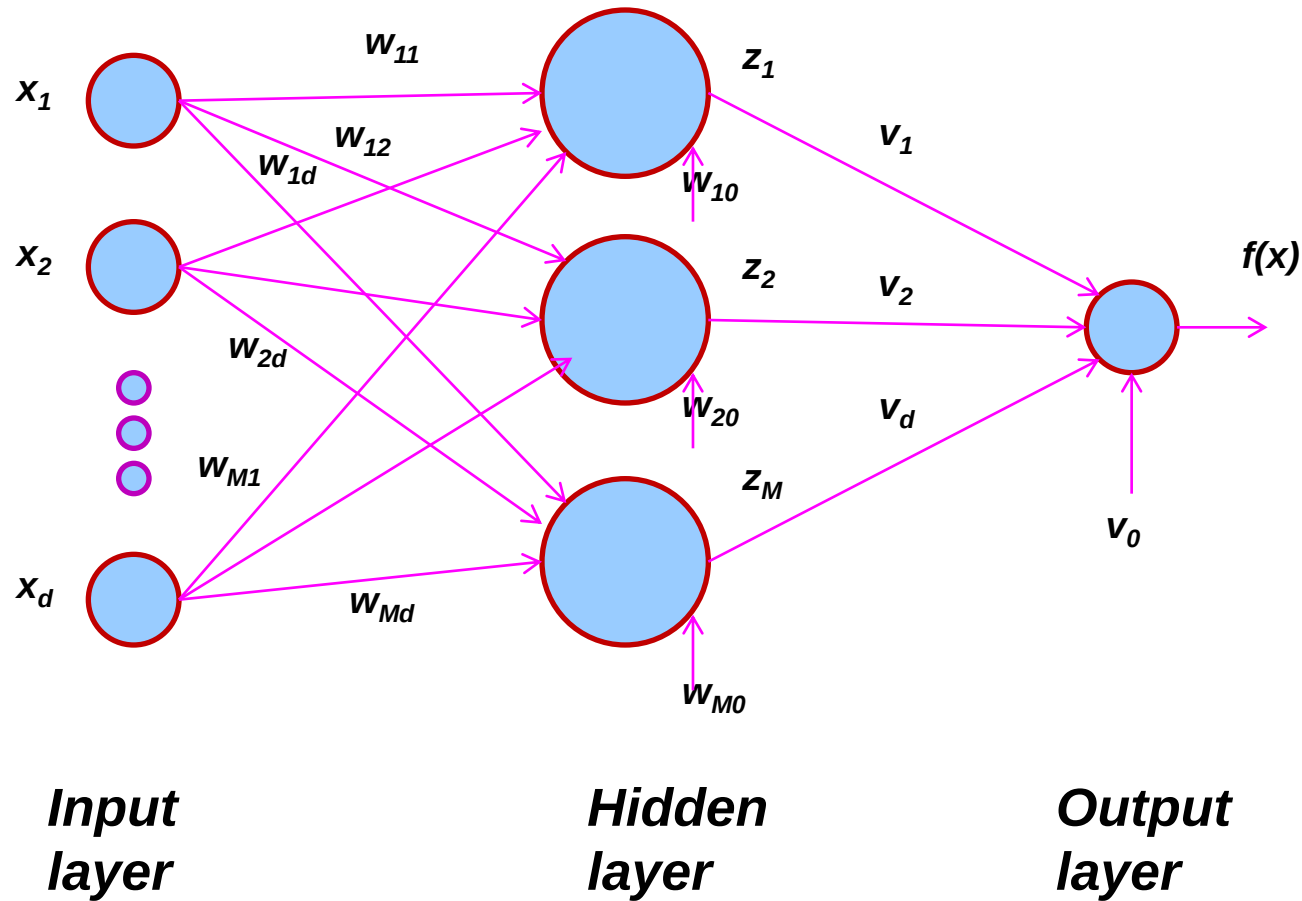


- Ένα ΤΝΔ είναι ένα δίκτυο από μονάδες perceptrons οι οποίες είναι διασυνδεδεμένες σε επίπεδα (**layers**)
- 3 βασικά επίπεδα (**Input, Hidden & Output Layers**)
- Είναι πιθανό να παρεμβάλλονται περισσότερα από ένα κρυμμένα (**hidden**) επίπεδα μεταξύ εισόδου και εξόδου.

- Δίκτυο εμπρόσθιας τροφοδότησης δίκτυο (feed forward NN) με 2 κρυμμένα επίπεδα.



MLP με M κρυμμένους νευρώνες



MLP με M κρυμμένους νευρώνες (συν.)

- w_{jk} : το βάρος μεταξύ του k -οστού χαρακτηριστικού input x_k και της j -οστής κρυμμένης μονάδας.
($j=1,\dots,M$ και $k=1,\dots,d$)
- v_j : το βάρος του j -οστής κρυμμένης μονάδας.
- z_j : η τιμή της συνάρτησης ενεργοποίησης της j -οστής κρυμμένης μονάδας.

$$z_j = \varphi(a_j) = \varphi(w_{j0} + w_j^T x_n)$$

- Υπάρχουν M τέτοιες συναρτήσεις ενεργοποίησης των M κρυμμένων μονάδων

MLP με M κρυμμένους νευρώνες (συν.)

- Η τελική έξοδος $f(x)$ του δικτύου προκύπτει από τον συνδυασμό των M συναρτήσεων $\phi(a_j)$ ή τιμών z_j

$$f(x) = \phi(v_0 + v^T z) = \phi\left(v_0 + \sum_{j=1}^M v_j z_j\right)$$

$$f(x) = \phi\left(\overset{\mathbf{b}(x)}{v_0 + v^T z}\right) = \phi\left(v_0 + \sum_{j=1}^M v_j \phi(a_j)\right)$$

- Αναλυτικότερα:

$$f(x) = \phi(b(x)) = \phi\left(v_0 + \sum_{j=1}^M v_j \phi\left(w_{j0} + \sum_{k=1}^d w_{jk} x_k\right)\right)$$

MLP με M κρυμμένους νευρώνες (συν.)

$$f(x) = \phi(b(x)) = \phi\left(v_0 + \sum_{j=1}^M v_j \varphi\left(w_{j0} + \sum_{k=1}^d w_{jk} x_k\right)\right)$$

- Παράμετροι του νευρωνικού δικτύου
 - M διανύσματα βαρών $\mathbf{w}_j = \{w_{jk}, k=0, 1, \dots, d\} j=1, \dots, M$
 - $M+1$ βάρη $v_j, j=0, 1, \dots, M$
- Έτσι συνολικά $M \times (d+1) + (M+1) = M \times (d+2) + 1$ άγνωστοι παράμετροι:

$$\theta = \left\{ \left[w_{jk} \right]_{k=0,1,\dots,d}^{j=1,\dots,M}, \left[v_j \right]_{j=0,1,\dots,M} \right\}$$

[2]. Εκπαίδευση MLP με M κρυμμένους νευρώνες

- Συνάρτηση σφάλματος:

$$\begin{aligned} E(\theta) &= \frac{1}{2} \sum_{n=1}^N (f(x_n, \theta) - y_n)^2 = \\ &= \frac{1}{2} \sum_{n=1}^N \left(\phi \left(v_0 + \sum_{j=1}^M v_j \varphi \left(w_{j0} + \sum_{k=1}^d w_{jk} x_k \right) \right) - y_n \right)^2 \end{aligned}$$

$$\text{όπου } \phi(a_n) = \frac{1}{1 + e^{-a_n}} \quad \varphi(b_n) = \frac{1}{1 + e^{-b_n}}$$

- Υπολογισμός των **μερικών παραγώγων** των βαρών:

- $\mathbf{v}_j$:
$$\frac{\partial E(\mathbf{w})}{\partial v_j} = \sum_{n=1}^N \delta_n \phi(b_n)(1 - \phi(b_n)) \begin{cases} 1 & j = 0 \\ \phi(a_{nj}) & j > 0 \end{cases}$$

- $\mathbf{w}_{jk}$:
$$\begin{aligned} \frac{\partial E(\theta)}{\partial w_{jk}} &= \sum_{n=1}^N \delta_n \frac{\partial \delta_n}{\partial b_n} \frac{\partial b_n}{\partial z_{nj}} \frac{\partial z_{nj}}{\partial a_{nj}} \frac{\partial a_{nj}}{\partial w_{jk}} = \\ &= \sum_{n=1}^N \delta_n [\phi(b_n)(1 - \phi(b_n))] [v_j \mathbb{I}[\phi(a_{nj})(1 - \phi(a_{nj}))]] \begin{cases} 1 & k = 0 \\ x_{nk} & k > 0 \end{cases} \end{aligned}$$

- Επαναληπτικοί τύποι ενημέρωσης των βαρών:

$$v_0^{(new)} = v_0^{(old)} - \eta \sum_{n=1}^N \delta_n \phi(b_n)(1 - \phi(b_n))$$

$$v_j^{(new)} = v_j^{(old)} - \eta \sum_{n=1}^N \delta_n \phi(b_n)(1 - \phi(b_n)) z_{nj}$$

$$w_{j0}^{(new)} = w_{j0}^{(old)} - \eta \sum_{n=1}^N \delta_n [\phi(b_n)(1 - \phi(b_n))] [v_j \mathbb{I}[\phi(a_{nj})(1 - \phi(a_{nj}))]]$$

$$w_{jk}^{(new)} = w_{jk}^{(old)} - \eta \sum_{n=1}^N \delta_n [\phi(b_n)(1 - \phi(b_n))] [v_j \mathbb{I}[\phi(a_{nj})(1 - \phi(a_{nj}))]] x_{nk}$$

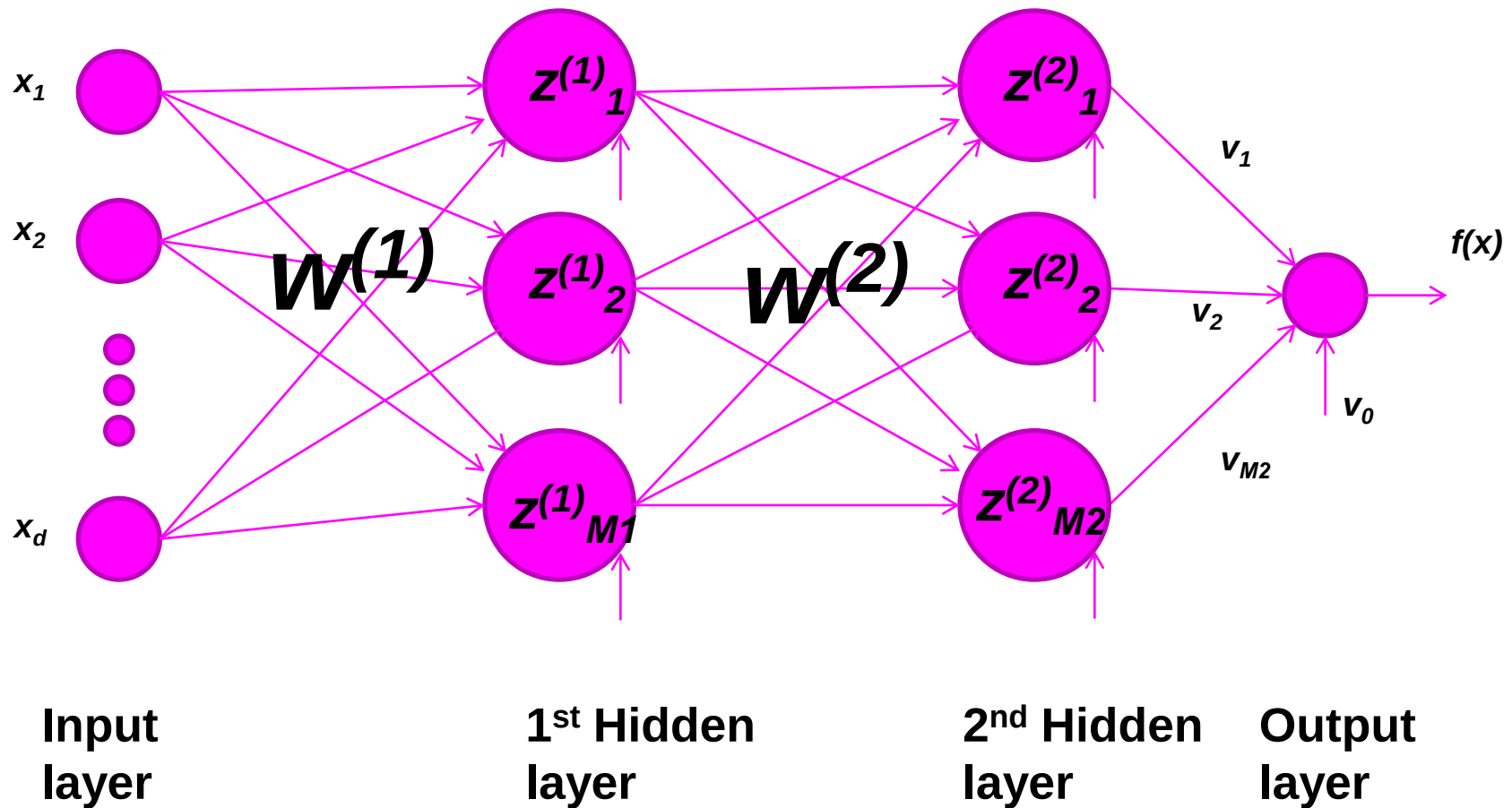
$$\delta_n = f(x_n) - y_n$$

Παρατηρήσεις

1. Ο αλγόριθμος εκπαίδευσης ονομάζεται **backpropagation (οπισθοδιάδοσης)** καθώς ενημερώνει τα βάρη διαχέοντας το σφάλμα προς τα πίσω .
2. Μέσω του κρυμμένου επιπέδου πραγματοποιείται ένας **μη-γραμμικός μετασχηματισμός** των δεδομένων με βάση τις συναρτήσεις ενεργοποίησης

$$\begin{array}{ccc} \mathbf{x} \in \mathbb{R}^d & \xrightarrow{\varphi} & \mathbf{z} \in \mathbb{R}^M \xrightarrow{\varphi} \mathbf{y} \\ z_j = \varphi(a_j) = \varphi(w_{j0} + w_j^T \mathbf{x}_n) & & f(\mathbf{x}) = \phi(v_0 + v^T \mathbf{z}) \end{array}$$

[3]. MLP με 2 κρυμμένα επίπεδα (M1, M2)



- 1^ο κρυμμένο επίπεδο

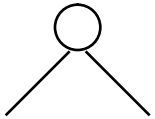
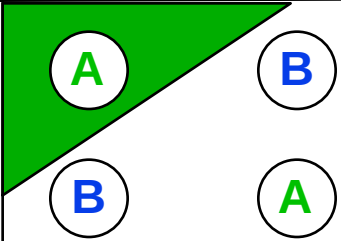
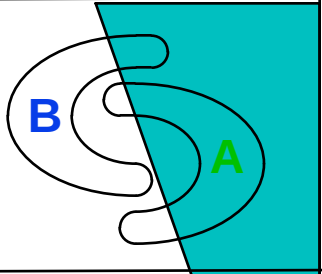
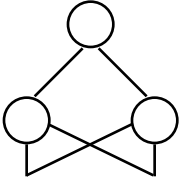
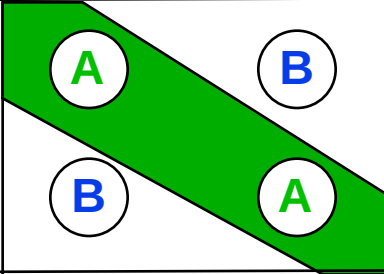
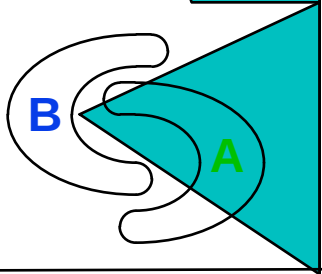
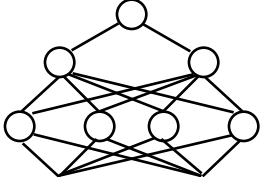
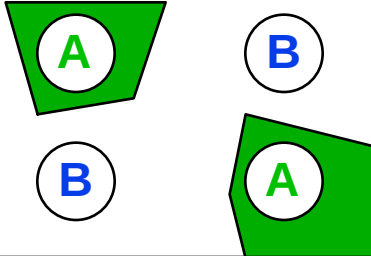
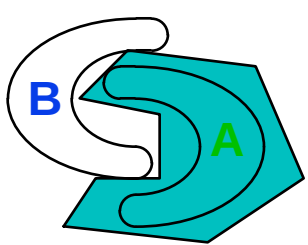
- M1 κόμβους, δηλ. M1 συναρτήσεις ενεργοποίησης $\varphi^{(1)}$
- $w_{j1,k}^{(1)}$ βάρη μεταξύ εισόδου και 1^{ου} κρυμμένου επιπέδου

- 2^ο κρυμμένο επίπεδο

- M2 κόμβους, δηλ. M2 συναρτήσεις ενεργοποίησης $\varphi^{(2)}$
- $w_{j2,j1}^{(2)}$ βάρη μεταξύ 1^{ου} και 2^{ου} κρυμμένου επιπέδου

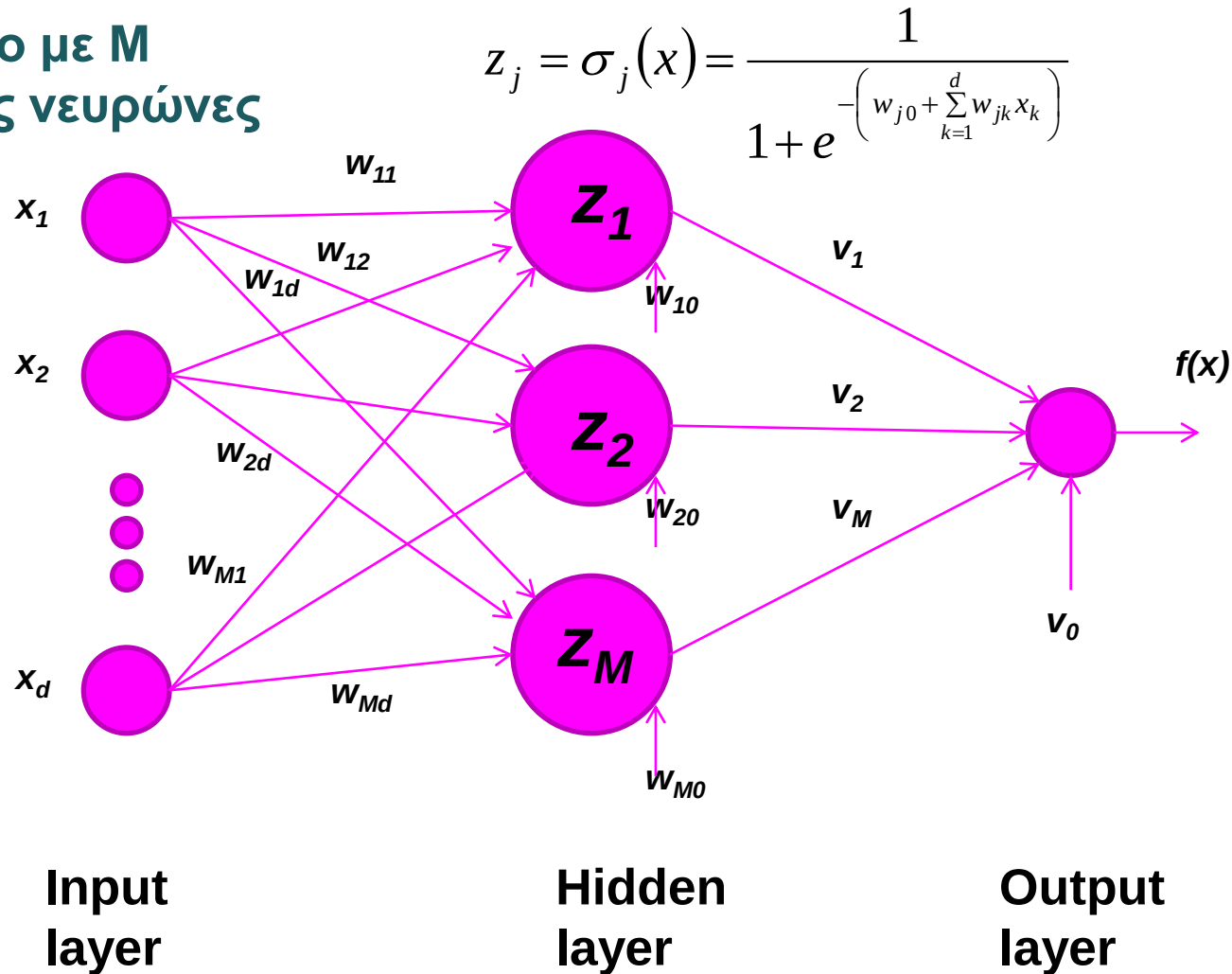
$$f(x) = \phi \left(v_0 + \sum_{j2=1}^{M2} v_{j2} \varphi^{(2)} \left(w_{j2,0}^{(2)} + \sum_{j1=1}^{M1} w_{j2,j1}^{(2)} \varphi^{(1)} \left(w_{j1,0}^{(1)} + \sum_{k=1}^d w_{j1,k}^{(1)} x_k \right) \right) \right)$$

Διάφορα μη-γραμμικά διαχωρίσιμα προβλήματα

Δομή δικτύου	Περιοχές απόφασης	Exclusive-OR Problem	Classes with Meshed regions
Single-Layer 	Half Plane Bounded By Hyperplane		
Two-Layer 	Convex Open Or Closed Regions		
Three-Layer 	ΟΤΙΔΗΠΟΤΕ (Complexity Limited by No. of Nodes)		

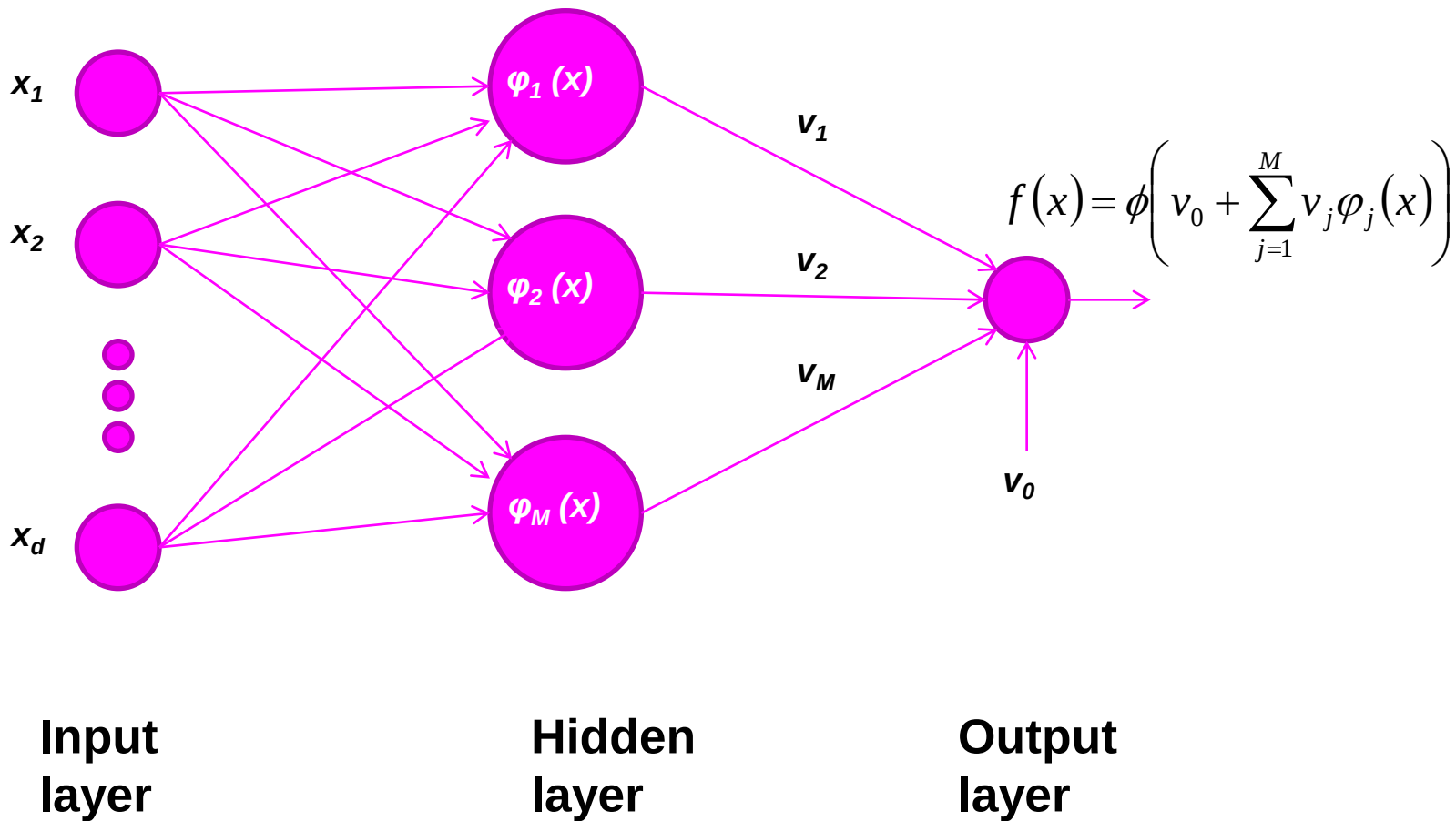
[4]. From MLP to Radial Basis Functions (RBF)

MLP δίκτυο με M
σιγμοειδής νευρώνες



Εναλλακτική μορφή MLP δικτύου

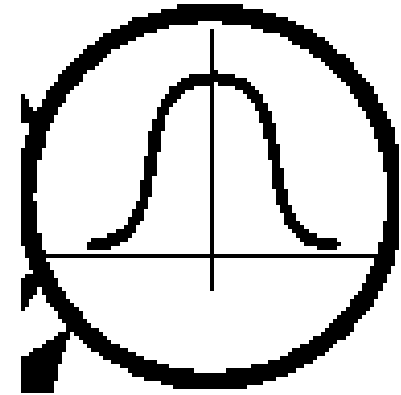
$$\varphi_j(x) = \frac{1}{1 + e^{-\left(w_{j0} + \sum_{k=1}^d w_{jk} x_k\right)}}$$



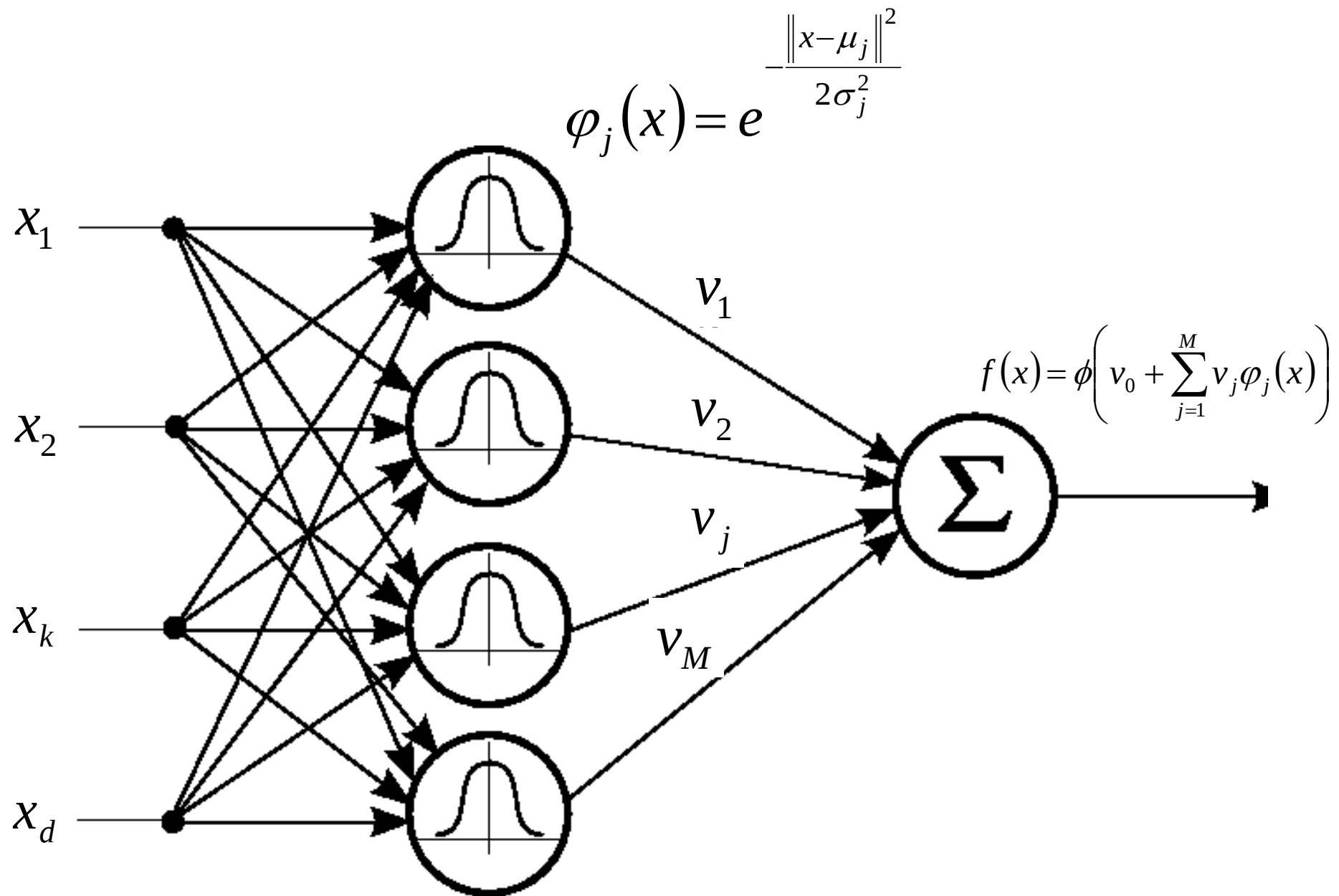
RBF Neural Networks

- Συνάρτηση ενεργοποίησης: ακτινική συνάρτηση βάσης (radial basis function – RBF)

$$\varphi_j(x) = e^{-\frac{\|x - \mu_j\|^2}{2\sigma_j^2}}$$



- Διαμερισμός του χώρου σε M σφαιρικές περιοχές.
- Οι παράμετροι $\{\mu_j, \sigma_j^2\}$ υποθέτουμε ότι είναι γνωστοί.
- π.χ. μπορούν να υπολογιστούν εκτελώντας τον αλγόριθμο k-means για M αριθμό ομάδων



- Έξοδος δικτύου: $f(x) = \phi \left(v_0 + \sum_{j=1}^M v_j \varphi_j(x) \right) \quad \varphi_j(x) = e^{-\frac{\|x - \mu_j\|^2}{2\sigma_j^2}}$
- Συνάρτηση σφάλματος: $E(\theta) = \frac{1}{2} \sum_{n=1}^N (f(x_n) - y_n)^2$
- Άγνωστοι παράμετροι (βάρη): $\theta = \{v_j\}_{j=0}^M$
- Μερικοί παράγωγοι των βαρών:

$$\frac{\partial E(w)}{\partial v_j} = \sum_{n=1}^N \delta_n \phi(x_n) (1 - \phi(x_n)) \begin{cases} 1 & j = 0 \\ \varphi_j(x_n) & j > 0 \end{cases}$$

$$(\delta_n = f(x_n) - y_n)$$

Εκπαίδευση ενός RBF Νευρωνικού Δικτύου

- Αρχικοποίηση παραμέτρων $\{ v_j \}_{j=0,1,2, \dots, M}$
- Επαναληπτικά μέχρι σύγκλιση

$$v_0^{(new)} = v_0^{(old)} - \eta \sum_{n=1}^N \delta_n \phi(x_n)(1 - \phi(x_n))$$

$$v_j^{(new)} = v_j^{(old)} - \eta \sum_{n=1}^N \delta_n \phi(x_n)(1 - \phi(x_n)) \varphi_j(x_n)$$

$$\delta_n = f(x_n) - y_n$$

[5]. Περίπτωση Δικτύου RBF με άγνωστα $\{\mu_j, \sigma_j^2\}$

- Εναλλακτικά μπορούμε να θεωρήσουμε και τις παραμέτρους των M RBF νευρώνων $\{\sigma_j^2, \mu_j\}$ ως αγνώστους :

$$\theta = v_0, \{v_j, \mu_j, \sigma_j^2\}_{j=1}^M$$

- τις οποίες μπορούμε να εκτιμήσουμε ελαχιστοποιώντας την συνάρτηση σφάλματος:

$$E(\theta) = \frac{1}{2} \sum_{n=1}^N (f(x_n) - y_n)^2$$

$$f(x_n) = \phi \left(v_0 + \sum_{j=1}^M v_j \varphi_j(x_n) \right) \quad \varphi_j(x_n) = e^{-\frac{\|x_n - \mu_j\|^2}{2\sigma_j^2}}$$

Εκπαίδευση RBF με άγνωστα $\{\mu_j, \sigma_j^2\}$ (συν.)

- Μερικές παράγωγοι : $\mathcal{G}_j = \{\mu_j \text{ or } \sigma_j^2\}$

$$\frac{\partial E(\theta)}{\partial \mathcal{G}_j} = \sum_{n=1}^N \delta_n \phi(x_n) (1 - \phi(x_n)) v_j \frac{\partial \phi_j(x_n)}{\partial \mathcal{G}_j}$$

$$\frac{\partial \phi_j(x_n)}{\partial \mu_j} = \phi_j(x_n) \frac{x_n - \mu_j}{\sigma_j^2}$$

$$\frac{\partial \phi_j(x_n)}{\partial \sigma_j} = \phi_j(x_n) \frac{\|x_n - \mu_j\|^2}{\sigma_j^3}$$

Εκπαίδευση RBF με άγνωστα $\{\mu_j, \sigma_j^2\}$ (συν.)

- Μερικές παράγωγοι : $\mathcal{G}_j = \{\mu_j \text{ or } \sigma_j^2\}$

$$\frac{\partial E(\theta)}{\partial \mathcal{G}_j} = \sum_{n=1}^N \delta_n \phi(x_n) (1 - \phi(x_n)) v_j \frac{\partial \phi_j(x_n)}{\partial \mathcal{G}_j}$$

$$\frac{\partial \phi_j(x_n)}{\partial \mu_j} = \phi_j(x_n) \frac{x_n - \mu_j}{\sigma_j^2}$$

$$\frac{\partial \phi_j(x_n)}{\partial \sigma_j} = \phi_j(x_n) \frac{\|x_n - \mu_j\|^2}{\sigma_j^3}$$

Εκπαίδευση RBF με άγνωστα $\{\mu_j, \sigma_j^2\}$ (συν.)

- Επαναληπτικοί τύποι:

$$v_0^{(new)} = v_0^{(old)} - \eta \sum_{n=1}^N \delta_n \phi(x_n) (1 - \phi(x_n))$$

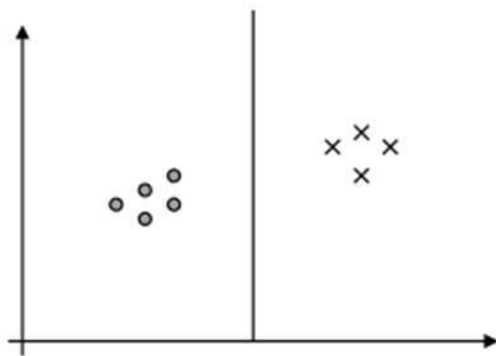
$$v_j^{(new)} = v_j^{(old)} - \eta \sum_{n=1}^N \delta_n \phi(x_n) (1 - \phi(x_n)) \varphi_j(x_n)$$

$$\mu_j^{(new)} = \mu_j^{(old)} - \eta \sum_{n=1}^N \delta_n \phi(x_n) (1 - \phi(x_n)) v_j \varphi_j(x_n) \frac{x_n - \mu_j^{(old)}}{\sigma_j^2}$$

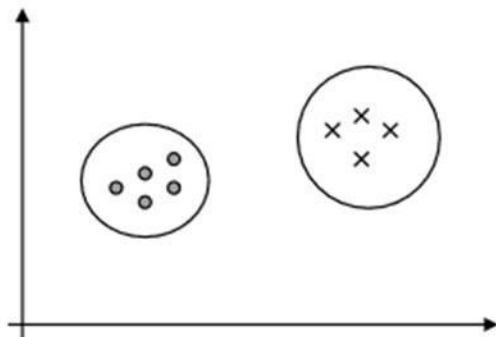
$$\sigma_j^{(new)} = \sigma_j^{(old)} - \eta \sum_{n=1}^N \delta_n \phi(x_n) (1 - \phi(x_n)) v_j \varphi_j(x_n) \frac{\|x_n - \mu_j^{(new)}\|^2}{(\sigma_j^3)^{(old)}}$$

RBF vs. MLP

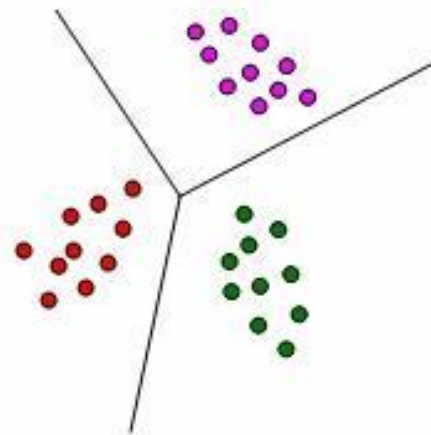
- Προβλήματα μάθησης με επίβλεψη (**Supervised learning**)



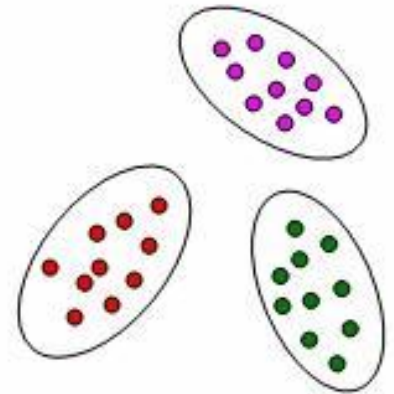
MLP



RBF



MLP



RBF