

ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ (LINEAR CLASSIFIERS)

❖ **Πρόβλημα:** Θεωρείστε ένα πρόβλημα δύο κλάσεων ω_1, ω_2

➤ $g(\underline{x}) = \underline{w}^T \underline{x} + w_0 = 0 =$
 $w_1 x_1 + w_2 x_2 + \dots + w_l x_l + w_0$

➤ Έστω $\underline{x}_1, \underline{x}_2$ πάνω στο υπερεπίπεδο απόφασης:

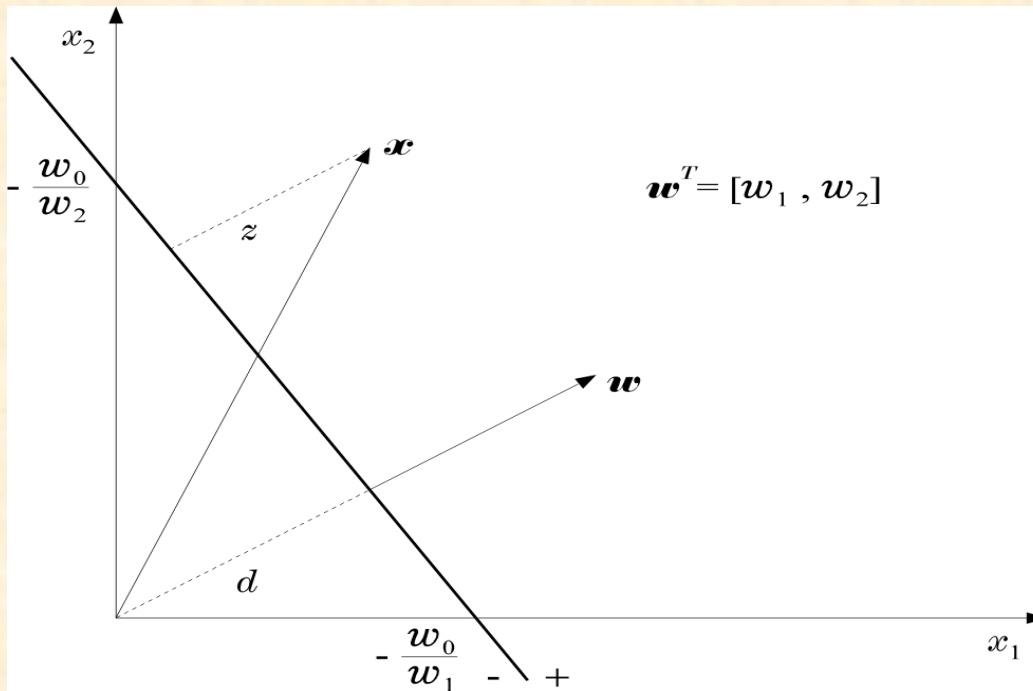
$$0 = \underline{w}^T \underline{x}_1 + w_0 = \underline{w}^T \underline{x}_2 + w_0 \Rightarrow$$

$$\underline{w}^T (\underline{x}_1 - \underline{x}_2) = 0 \quad \forall \underline{x}_1, \underline{x}_2$$

➤ Έτσι:

$\underline{w} \perp$ στο υπερπίπεδο

$$g(\underline{x}) = \underline{w}^T \underline{x} + w_0 = 0$$



$$d = \frac{|w_0|}{\sqrt{w_1^2 + w_2^2}}, \quad z = \frac{|g(\underline{x})|}{\sqrt{w_1^2 + w_2^2}}$$

❖ Ο αλγόριθμος Perceptron

➤ Έστω ότι οι κλάσεις είναι γραμμικώς διαχωρίσιμες, δηλ.,

$$\exists \underline{w}^* : \underline{w}^{*T} \underline{x} > 0 \quad \forall \underline{x} \in \omega_1$$

$$\underline{w}^{*T} \underline{x} < 0 \quad \forall \underline{x} \in \omega_2$$

➤ Η περίπτωση $\underline{w}^{*T} \underline{x} + w_0^*$ εμπίπτει στην παραπάνω διατύπωση, αφού

$$\bullet \quad \underline{w}' \equiv \begin{bmatrix} \underline{w}^* \\ w_0^* \end{bmatrix}, \quad \underline{x}' = \begin{bmatrix} \underline{x} \\ 1 \end{bmatrix}$$

$$\bullet \quad \underline{w}^{*T} \underline{x} + w_0^* = \underline{w}'^T \underline{x}' = 0$$

➤ **Στόχος:** Υπολογισμός λύσης, δηλ., προσδιορισμός υπερεπιπέδου \underline{w} , έτσι ώστε

$$\underline{w}^T \underline{x} (> <) 0 \quad x \in \begin{cases} \omega_1 \\ \omega_2 \end{cases}$$

- **Βήματα**

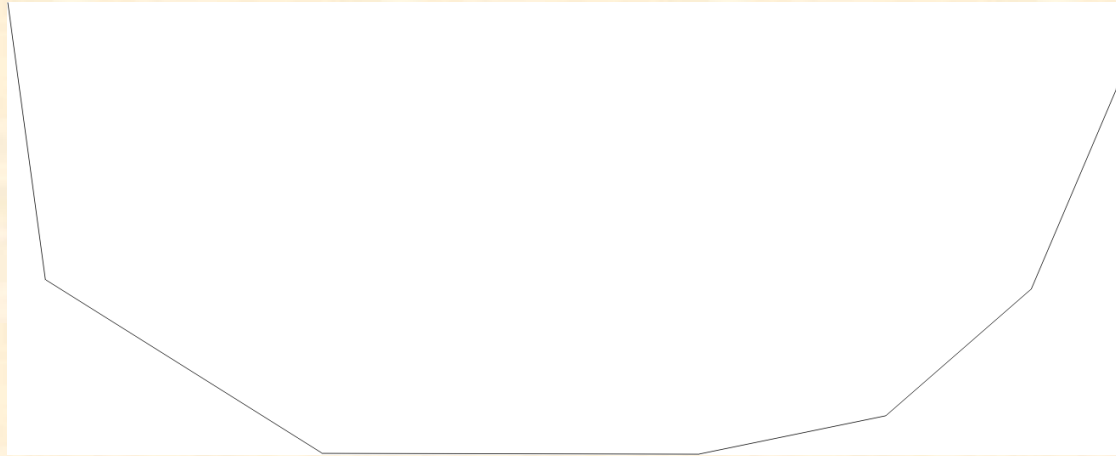
- Ορισμός συνάρτησης κόστους που θα ελαχιστοποιηθεί
- Επιλογή αλγορίθμου για την ελαχιστοποίηση της συνάρτησης κόστους
- Το ελάχιστο αντιστοιχεί σε μία λύση.

➤ Η συνάρτηση κόστους

$$J(\underline{w}) = \sum_{\underline{x} \in Y} (\delta_x \underline{w}^T \underline{x})$$

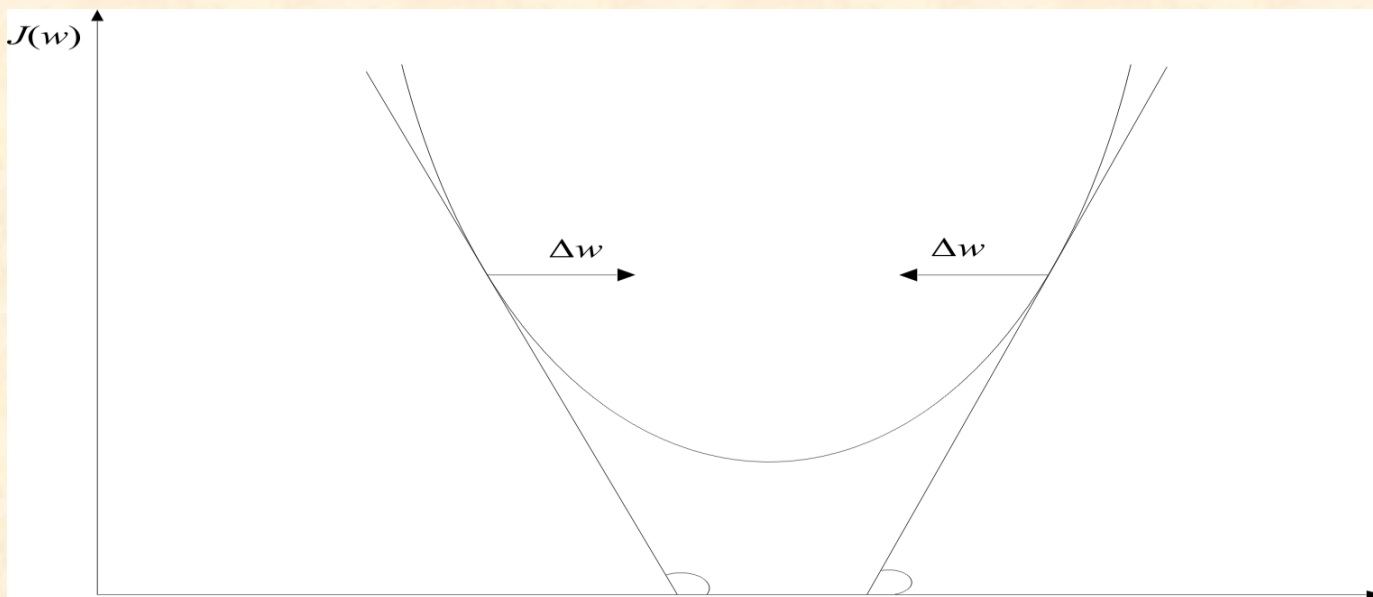
- όπου Y το υποσύνολο των **λανθασμένα** ταξινομημένων διανυσμάτων από το \underline{w} . Όταν $Y = \emptyset$ έχουμε μία λύση και είναι $J(\underline{w}) = 0$
- $\delta_x = -1$ if $\underline{x} \in Y$ and $\underline{x} \in \omega_1$
 $\delta_x = +1$ if $\underline{x} \in Y$ and $\underline{x} \in \omega_2$
- Γενικά, $J(\underline{w}) \geq 0$

- Η $J(\underline{w})$ είναι τμηματικά γραμμική (Γιατί?)



➤ Ο Αλγόριθμος

- Υιοθετείται η φιλοσοφία της σταδιακής καθόδου κατά την κλίση (gradient descent).



$$\underline{w}(\text{new}) = \underline{w}(\text{old}) + \Delta \underline{w}$$

$$\Delta \underline{w} = -\mu \frac{\partial J(\underline{w})}{\partial \underline{w}} \Big|_{\underline{w} = \underline{w}(\text{old})}$$

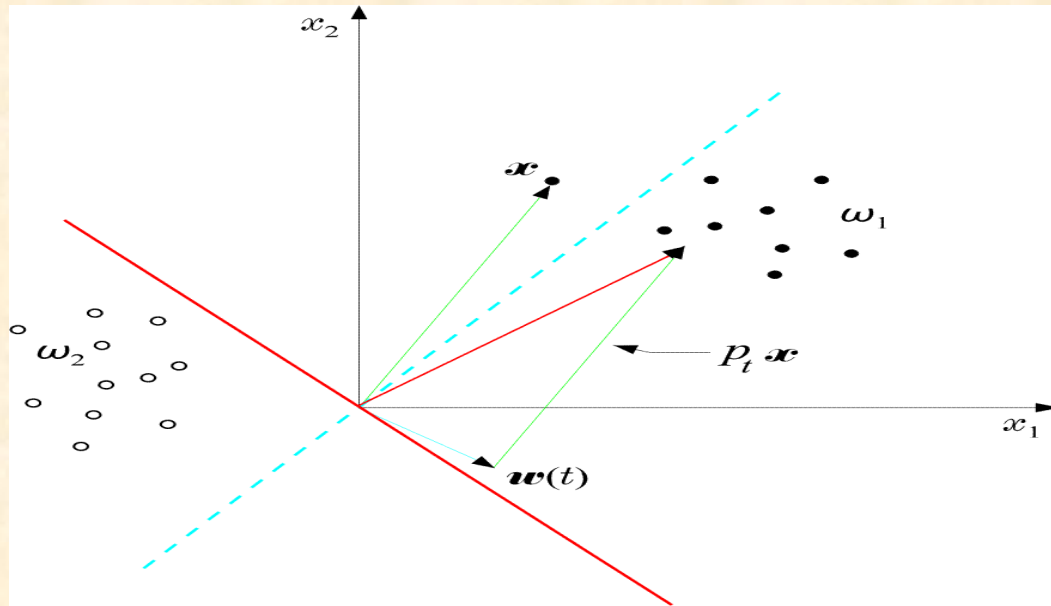
- Όπου είναι νόμιμο (valid)

$$\frac{\partial J(\underline{w})}{\partial \underline{w}} = \frac{\partial}{\partial \underline{w}} \left(\sum_{\underline{x} \in Y} \delta_x \underline{w}^T \underline{x} \right) = \sum_{\underline{x} \in Y} \delta_x \underline{x}$$

- $$\underline{w}(t+1) = \underline{w}(t) - \rho_t \sum_{\underline{x} \in Y} \delta_x \underline{x}$$

Αυτός είναι ο γνωστός αλγόριθμος Perceptron.

➤ Ένα παράδειγμα



$$\begin{aligned}\underline{w}(t+1) &= \underline{w}(t) + \rho_t \underline{x} \\ &= \underline{w}(t) - \rho_t \delta_x \underline{x} \quad (\delta_x = -1)\end{aligned}$$

- Ο αλγόριθμος perceptron **συγκλίνει** σε **πεπερασμένο** αριθμό επαναλήψεων σε λύση αν

$$\lim_{t \rightarrow \infty} \sum_{k=0}^t \rho_k \rightarrow \infty$$

$$\lim_{t \rightarrow \infty} \sum_{k=0}^t \rho_k^2 < +\infty$$

e.g.: $\rho_t = \frac{c}{t}$

❖ Μία χρήσιμη παραλλαγή του αλγορίθμου perceptron

$$\underline{w}(t+1) = \underline{w}(t) + \rho \underline{x}_{(t)}, \quad \begin{array}{l} \underline{w}^T(t) \underline{x}_{(t)} \leq 0 \\ \underline{x}_{(t)} \in \omega_1 \end{array}$$

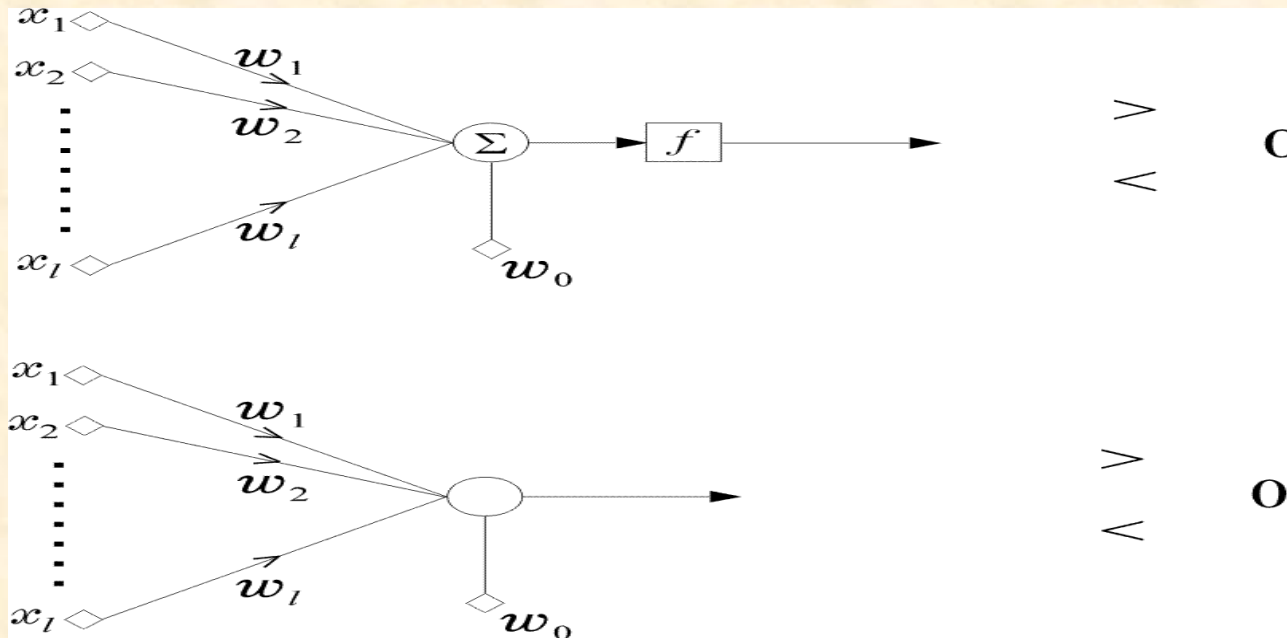
$$\underline{w}(t+1) = \underline{w}(t) - \rho \underline{x}_{(t)}, \quad \begin{array}{l} \underline{w}^T(t) \underline{x}_{(t)} \geq 0 \\ \underline{x}_{(t)} \in \omega_2 \end{array}$$

$$\underline{w}(t+1) = \underline{w}(t) \quad \text{διαφορετικά}$$

➤ Είναι ένας αλγόριθμος του τύπου

επιβράβευσης και τιμωρίας

❖ Το perceptron



w_i 's συνάψεις ή συναπτικά βάρη

w_0 κατώφλι

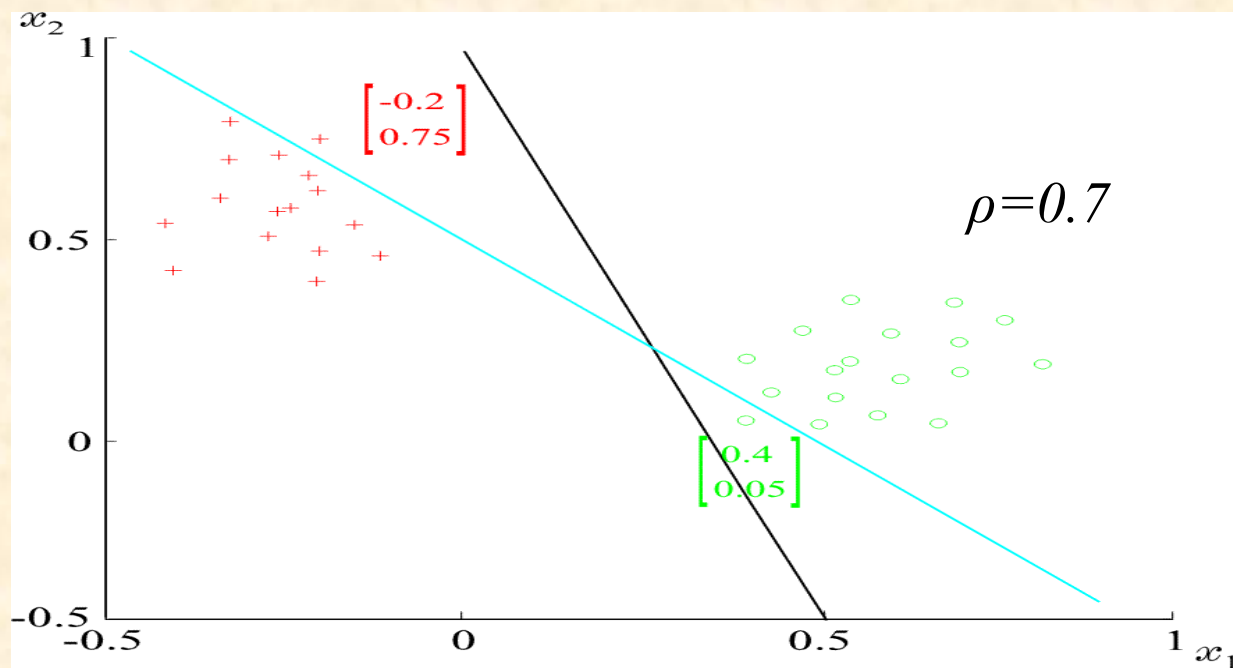
- Το δίκτυο καλείται **perceptron** ή **νευρώνας (neuron)**
- Είναι μία **μηχανή μάθησης** που **μαθαίνει** από τα **διανύσματα εκπαίδευσης** μέσω του αλγορίθμου perceptron.

➤ **Παράδειγμα:** Σε κάποια επανάληψη t ο αλγόριθμος perceptron βρίσκεται στην κατάσταση

$$w_1 = 1, w_2 = 1, w_0 = -0.5$$

$$x_1 + x_2 - 0.5 = 0$$

Το αντίστοιχο υπερεπίπεδο είναι



$$\underline{w}(t+1) = \begin{bmatrix} 1 \\ 1 \\ -0.5 \end{bmatrix} - 0.7(-1) \begin{bmatrix} 0.4 \\ 0.05 \\ 1 \end{bmatrix} - 0.7(+1) \begin{bmatrix} -0.2 \\ 0.75 \\ 1 \end{bmatrix} = \begin{bmatrix} 1.42 \\ 0.51 \\ -0.5 \end{bmatrix}$$

❖ Μέθοδοι ελαχίστων τετραγώνων

- Αν οι κλάσεις είναι γραμμικώς διαχωρίσιμες, η έξοδος του perceptron είναι ± 1
- Αν οι κλάσεις ΔΕΝ είναι γραμμικώς διαχωρίσιμες, θα υπολογίσουμε τα βάρη w_1, w_2, \dots, w_0

έτσι ώστε η **διαφορά** μεταξύ

- Της πραγματικής εξόδου του ταξινομητή, $\underline{w}^T \underline{x}$, και
- Των επιθυμητών εξόδων, π.χ.

$$+1 \text{ αν } \underline{x} \in \omega_1$$

$$-1 \text{ αν } \underline{x} \in \omega_2$$

να είναι **ΜΙΚΡΗ**

➤ ΜΙΚΡΗ, ως προς το μέσο τετραγωνικό σφάλμα (mean square error), σημαίνει επιλογή του \underline{w} έτσι ώστε η συνάρτηση κόστους

- $J(\underline{w}) \equiv E[(y - \underline{w}^T \underline{x})^2]$ να ελαχιστοποιηθεί
- $\hat{\underline{w}} = \arg \min_{\underline{w}} J(\underline{w})$
- y οι αντίστοιχες επιθυμητές αποκρίσεις

➤ Ελαχιστοποίηση της

$J(\underline{w})$ ως προς \underline{w} δίνει :

$$\begin{aligned}\frac{\partial J(\underline{w})}{\partial \underline{w}} &= \frac{\partial}{\partial \underline{w}} E[(y - \underline{w}^T x)^2] = 0 \\ &= 2E[\underline{x}(y - \underline{x}^T \underline{w})] \Rightarrow \\ E[\underline{x}\underline{x}^T] \underline{w} &= E[\underline{x}y] \Rightarrow\end{aligned}$$

$$\underline{\hat{w}} = R_x^{-1} E[\underline{x}y]$$

όπου R_x είναι ο πίνακας αυτοσυσχέτισης (autocorrelation matrix)

$$R_x \equiv E[\underline{x}\underline{x}^T] = \begin{bmatrix} E[x_1 x_1] & E[x_1 x_2] \dots & E[x_1 x_l] \\ \dots\dots\dots & \dots\dots\dots & \dots\dots\dots \\ E[x_l x_1] & E[x_l x_2] \dots & E[x_l x_l] \end{bmatrix}$$

και $E[\underline{x}y] = \begin{bmatrix} E[x_1 y] \\ \dots \\ E[x_l y] \end{bmatrix}$ το διάνυσμα ετεροσυσχέτισης (crosscorrelation vector)

- ❖ ΜΙΚΡΗ, ως προς το άθροισμα των τετραγωνικών σφαλμάτων, σημαίνει

- $$J(\underline{w}) = \sum_{i=1}^N (y_i - \underline{w}^T \underline{x}_i)^2$$

(y_i, \underline{x}_i) : ζεύγη εκπ/σης δηλαδή, το διαν. εισ. x_i και η αντίστοιχη **ετικέτα κλάσης** $y_i (\pm 1)$.

- $$\frac{\partial J(\underline{w})}{\partial \underline{w}} = \frac{\partial}{\partial \underline{w}} \sum_{i=1}^N (y_i - \underline{w}^T \underline{x}_i)^2 = 0 \Rightarrow$$

$$\left(\sum_{i=1}^N \underline{x}_i \underline{x}_i^T \right) \underline{w} = \sum_{i=1}^N \underline{x}_i y_i$$

❖ Ψευδοαντίστροφος πίνακας

➤ Ορίζουμε

$$X = \begin{bmatrix} \underline{x}_1^T \\ \underline{x}_2^T \\ \dots \\ \underline{x}_N^T \end{bmatrix} \quad (\text{ένας } N \times l \text{ πίνακας})$$

$$\underline{y} = \begin{bmatrix} y_1 \\ \dots \\ y_N \end{bmatrix} \quad \text{αντίστοιχες επιθυμητές αποκρίσεις}$$

➤ $X^T = [\underline{x}_1, \underline{x}_2, \dots, \underline{x}_N]$ (ένας $l \times N$ πίνακας)

$$\text{➤ } X^T X = \sum_{i=1}^N \underline{x}_i \underline{x}_i^T$$

$$\text{➤ } X^T \underline{y} = \sum_{i=1}^N \underline{x}_i y_i$$

Έτσι
$$\left(\sum_{i=1}^N \underline{x}_i^T \underline{x}_i\right) \underline{\hat{w}} = \left(\sum_{i=1}^N \underline{x}_i y_i\right)$$

$$(X^T X) \underline{\hat{w}} = X^T \underline{y} \Rightarrow$$

$$\underline{\hat{w}} = (X^T X)^{-1} X^T \underline{y}$$

$$= X^\# \underline{y}$$

$$X^\# \equiv (X^T X)^{-1} X^T$$

Ψευδοαντίστροφος του X

➤ Έστω $N=l$ \Rightarrow ο X είναι τετραγωνικός και (γενικά) αντιστρέψιμος. Τότε

$$(X^T X)^{-1} X^T = X^{-1} X^{-T} X^T = X^{-1} \Rightarrow$$

$$X^\# = X^{-1}$$

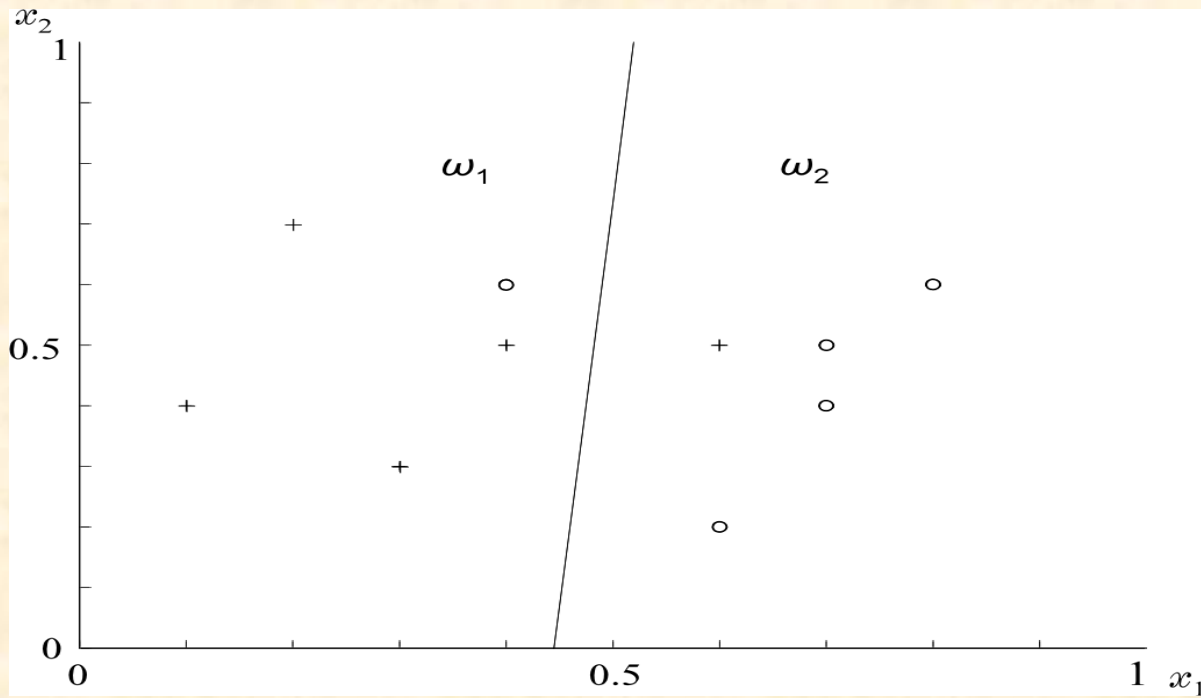
- Έστω $N > l$. Τότε, γενικά, δεν υπάρχει λύση που να ικανοποιεί όλες τις εξισώσεις ταυτόχρονα:

$$\begin{array}{l}
 \underline{x}_1^T \underline{w} = y_1 \\
 \underline{x}_2^T \underline{w} = y_2 \\
 \dots \\
 \underline{x}_N^T \underline{w} = y_N
 \end{array}
 \quad N \text{ εξισώσεις } > l \text{ άγνωστοι}$$

- Η “λύση” $\underline{w} = X^\# \underline{y}$ αντιστοιχεί στην ελάχιστη τιμή του αθροίσματος τετραγώνων (minimum sum of squares solution).

➤ Παράδειγμα: $\omega_1 : \begin{bmatrix} 0.4 \\ 0.5 \end{bmatrix}, \begin{bmatrix} 0.6 \\ 0.5 \end{bmatrix}, \begin{bmatrix} 0.1 \\ 0.4 \end{bmatrix}, \begin{bmatrix} 0.2 \\ 0.7 \end{bmatrix}, \begin{bmatrix} 0.3 \\ 0.3 \end{bmatrix}$

$\omega_2 : \begin{bmatrix} 0.4 \\ 0.6 \end{bmatrix}, \begin{bmatrix} 0.6 \\ 0.2 \end{bmatrix}, \begin{bmatrix} 0.7 \\ 0.4 \end{bmatrix}, \begin{bmatrix} 0.8 \\ 0.6 \end{bmatrix}, \begin{bmatrix} 0.7 \\ 0.5 \end{bmatrix}$



$$X = \begin{bmatrix} 0.4 & 0.5 & 1 \\ 0.6 & 0.5 & 1 \\ 0.1 & 0.4 & 1 \\ 0.2 & 0.7 & 1 \\ 0.3 & 0.3 & 1 \\ 0.4 & 0.6 & 1 \\ 0.6 & 0.2 & 1 \\ 0.7 & 0.4 & 1 \\ 0.8 & 0.6 & 1 \\ 0.7 & 0.5 & 1 \end{bmatrix} = \underline{y}$$

$$\blacktriangleright \quad X^T X = \begin{bmatrix} 2.8 & 2.24 & 4.8 \\ 2.24 & 2.41 & 4.7 \\ 4.8 & 4.7 & 10 \end{bmatrix}, \quad X^T \underline{y} = \begin{bmatrix} -1.6 \\ 0.1 \\ 0.0 \end{bmatrix}$$

$$\underline{w} = (X^T X)^{-1} X^T \underline{y} = \begin{bmatrix} -3.13 \\ 0.24 \\ 1.34 \end{bmatrix}$$