Μάθημα 4

Κανόνες Εκμάθησης Νευρώνα (Μέρος 2)

Κανόνας ΔΕΛΤΑ (DELTA Rule)

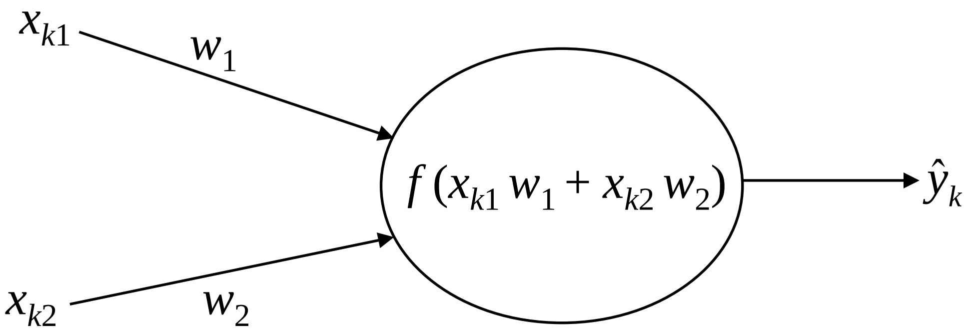
Μηχανική Μάθηση Νευρώνα: Παράδειγμα (Λογικό Ή)

Δεδομένα Εισόδου-Εξόδου

	X		*Y*
	x_1	x_2	y
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	1

k

N=4, p=2



Συνάρτηση Ενεργοποίησης: $f(a)=\begin{cases}1, & \alpha\nu\ a\ge 0.5\\ 0, & \alpha\lambda\lambda\iota\acute{\omega}\varsigma\end{cases}$

Όπου $\alpha = x_{k1}w_1 + x_{k2}w_2$

Άρα: $$\boxed{\hat{y}_k = f\left(x_{k1}w_1 + x_{k2}w_2\right)}$$

Παρατήρηση: Από τον νευρώνα θα περνάει κάθε δυάδα (x_{k1}, x_{k2}) για k=1, 2, 3, 4. Για κάθε μία από αυτές ο νευρώνας θα δώσει (εκτιμήσει) μία αντίστοιχη έξοδο. Έτσι θα έχουμε 4 εκτιμώμενες εξόδους $\hat{y}_1, \hat{y}_2, \hat{y}_3$ και $\hat{y}_4$. Έτσι θα υπολογίσουμε τα παρακάτω 4 επιμέρους σφάλματα

$$E_1 = |y_1 - \hat{y}_1|,\ E_2 = |y_2 - \hat{y}_2|,\ E_3 = |y_3 - \hat{y}_3|,\ \text{και}\ \ E_4 = |y_4 - \hat{y}_4|$$

Συνεπώς, το συνολικό σφάλμα για όλα τα δεδομένα εισόδου-εξόδου είναι:

$$E = E_1 + E_2 + E_3 + E_4 = \sum_{k=1}^{4} E_k \quad \text{ή ισοδύναμα} \quad E = \sum_{k=1}^{4} |y_k - \hat{y}_k| = \sum_{k=1}^{4} \left|y_k - f\left(x_{k1}w_1 + x_{k2}w_2\right)\right|$$

Στόχος: Η τιμή της συνάρτησης σφάλματος να γίνει όσο το δυνατόν πιο μικρή (κοντά στο μηδέν (ή στην καλύτερη κατάσταση ίση με μηδέν)

Μηχανική Μάθηση Νευρώνα: Παράδειγμα (Λογικό Ή)

Δεδομένα Εισόδου-Εξόδου

X

	x_1	x_2
1	0	0
2	0	1
3	1	0
4	1	1

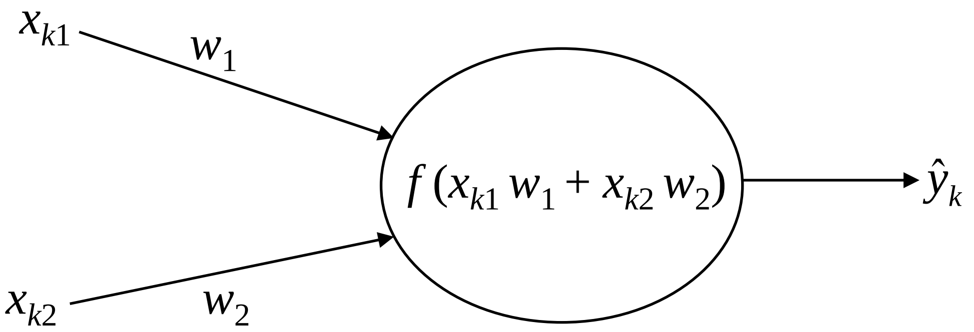
k

Y

y
0
1
1
1

$N=4,\ p=2$

Συνάρτηση Ενεργοποίησης: $f(a)=\begin{cases}1, & \alpha\nu\ a \geq 0.5\\ 0, & \alpha\lambda\lambda\iota\acute{\omega}\varsigma\end{cases}$



Όπου $\alpha = x_{k1}w_1 + x_{k2}w_2$

Άρα: $\boxed{\hat{y}_k = f\left(x_{k1}w_1 + x_{k2}w_2\right)}$

Αρχικοποίηση: $w_1=w_2=0$. Βήμα αύξησης 0.25

Βήμα 1)

$$x_{11}=0,\ x_{12}=0.\ \text{Άρα},\ \alpha = x_{11}w_1 + x_{12}w_2 = 0\cdot 0 + 0\cdot 0 = 0 \rightarrow \hat{y}_1 = f(a) = 0 \rightarrow E_1 = |0-0| = 0$$

$$x_{21}=0,\ x_{22}=0.\ \text{Άρα},\ \alpha = x_{21}w_1 + x_{22}w_2 = 0\cdot 0 + 1\cdot 0 = 0 \rightarrow \hat{y}_2 = f(a) = 0 \rightarrow E_2 = |1-0| = 1$$

$$x_{31}=0,\ x_{32}=0.\ \text{Άρα},\ \alpha = x_{31}w_1 + x_{32}w_2 = 1\cdot 0 + 0\cdot 0 = 0 \rightarrow \hat{y}_3 = f(a) = 0 \rightarrow E_3 = |1-0| = 1$$

$$x_{41}=0,\ x_{42}=0.\ \text{Άρα},\ \alpha = x_{41}w_1 + x_{42}w_2 = 1\cdot 0 + 1\cdot 0 = 0 \rightarrow \hat{y}_4 = f(a) = 0 \rightarrow E_4 = |1-0| = 1$$

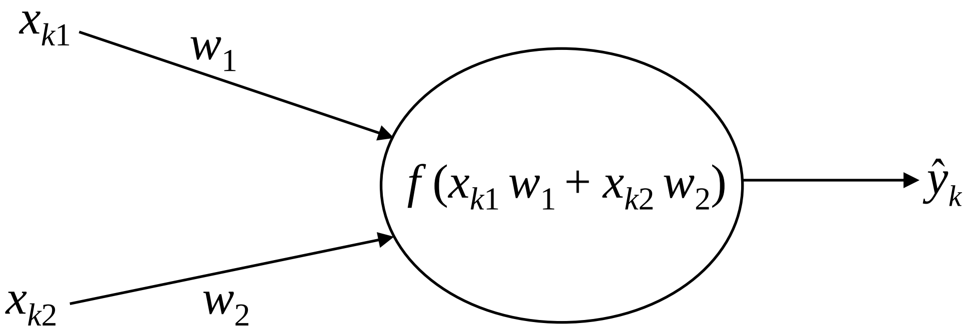
$$E = \sum_{k=1}^{4} E_k = 0+1+1+1 = 3$$

Μηχανική Μάθηση Νευρώνα: Παράδειγμα (Λογικό Ή)

Δεδομένα Εισόδου-Εξόδου

k	X		Y
	x_1	x_2	y
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	1

N=4, p=2



Συνάρτηση Ενεργοποίησης: $f\,(a)=\begin{cases}1, & \alpha\nu\ a\geq 0.5\\ 0, & \alpha\lambda\lambda\iota\acute{\omega}\varsigma\end{cases}$

Όπου $\alpha = x_{k1}\,w_1 + x_{k2}\,w_2$

Άρα: $\boxed{\hat{y}_k = f\left(x_{k1}\,w_1 + x_{k2}\,w_2\right)}$

Βήμα 2) $w_1(new) = w_1(old) + 0.25 = 0 + 0.25 = 0.25$ $\quad w_2(new) = w_2(old) + 0.25 = 0 + 0.25 = 0.25$

$$x_{11} = 0,\ x_{12} = 0.\ \text{Άρα},\ \alpha = x_{11}w_1 + x_{12}w_2 = 0\cdot 0.25 + 0\cdot 0.25 = 0.00 \rightarrow \hat{y}_1 = f\,(a) = 0 \rightarrow E_1 = |0-0| = 0$$

$$x_{21} = 0,\ x_{22} = 0.\ \text{Άρα},\ \alpha = x_{21}w_1 + x_{22}w_2 = 0\cdot 0.25 + 1\cdot 0.25 = 0.25 \rightarrow \hat{y}_2 = f\,(a) = 0 \rightarrow E_2 = |1-0| = 1$$

$$x_{31} = 0,\ x_{32} = 0.\ \text{Άρα},\ \alpha = x_{31}w_1 + x_{32}w_2 = 1\cdot 0.25 + 0\cdot 0.25 = 0.25 \rightarrow \hat{y}_3 = f\,(a) = 0 \rightarrow E_3 = |1-0| = 1$$

$$x_{41} = 0,\ x_{42} = 0.\ \text{Άρα},\ \alpha = x_{41}w_1 + x_{42}w_2 = 1\cdot 0.25 + 1\cdot 0.25 = 0.50 \rightarrow \hat{y}_4 = f\,(a) = 1 \rightarrow E_4 = |1-1| = 0$$

$$E = \sum_{k=1}^{4} E_k = 0 + 1 + 1 + 0 = 2$$

Μηχανική Μάθηση Νευρώνα: Παράδειγμα (Λογικό Ή)

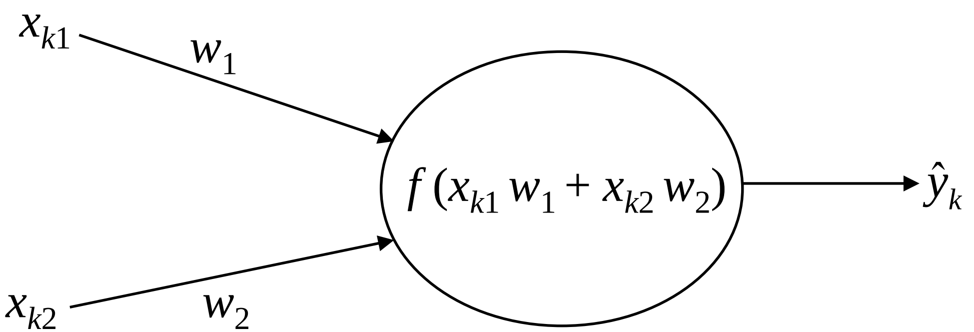
Δεδομένα Εισόδου-Εξόδου

k

	X		**Y**
	x_1	x_2	y
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	1

N=4, p=2

Συνάρτηση Ενεργοποίησης: $f(a)=\begin{cases}1, & \alpha\nu\ a\geq 0.5\\ 0, & \alpha\lambda\lambda\iota\acute{\omega}\varsigma\end{cases}$



Όπου $\alpha = x_{k1}w_1 + x_{k2}w_2$

Άρα: $\boxed{\hat{y}_k = f\left(x_{k1}w_1+x_{k2}w_2\right)}$

Βήμα 3) $w_1(new)=w_1(old)+0.25=0.25+0.25=0.5$ $w_2(new)=w_2(old)+0.25=0.25+0.25=0.5$

$x_{11}=0,\ x_{12}=0.$ Άρα, $\alpha = x_{11}w_1+x_{12}w_2=0\cdot 0.5+0\cdot 0.5=0.0\rightarrow \hat{y}_1 = f(a)=0\rightarrow E_1=|0-0|=0$

$x_{21}=0,\ x_{22}=0.$ Άρα, $\alpha = x_{21}w_1+x_{22}w_2=0\cdot 0.5+1\cdot 0.5=0.5\rightarrow \hat{y}_2 = f(a)=1\rightarrow E_2=|1-1|=0$

$x_{31}=0,\ x_{32}=0.$ Άρα, $\alpha = x_{31}w_1+x_{32}w_2=1\cdot 0.5+0\cdot 0.5=0.5\rightarrow \hat{y}_3 = f(a)=1\rightarrow E_3=|1-1|=0$

$x_{41}=0,\ x_{42}=0.$ Άρα, $\alpha = x_{41}w_1+x_{42}w_2=1\cdot 0.5+1\cdot 0.5=1.0\rightarrow \hat{y}_4 = f(a)=1\rightarrow E_4=|1-1|=0$

Τελικές τιμές των συναπτικών βαρών: **w_1=w_2=0.5**

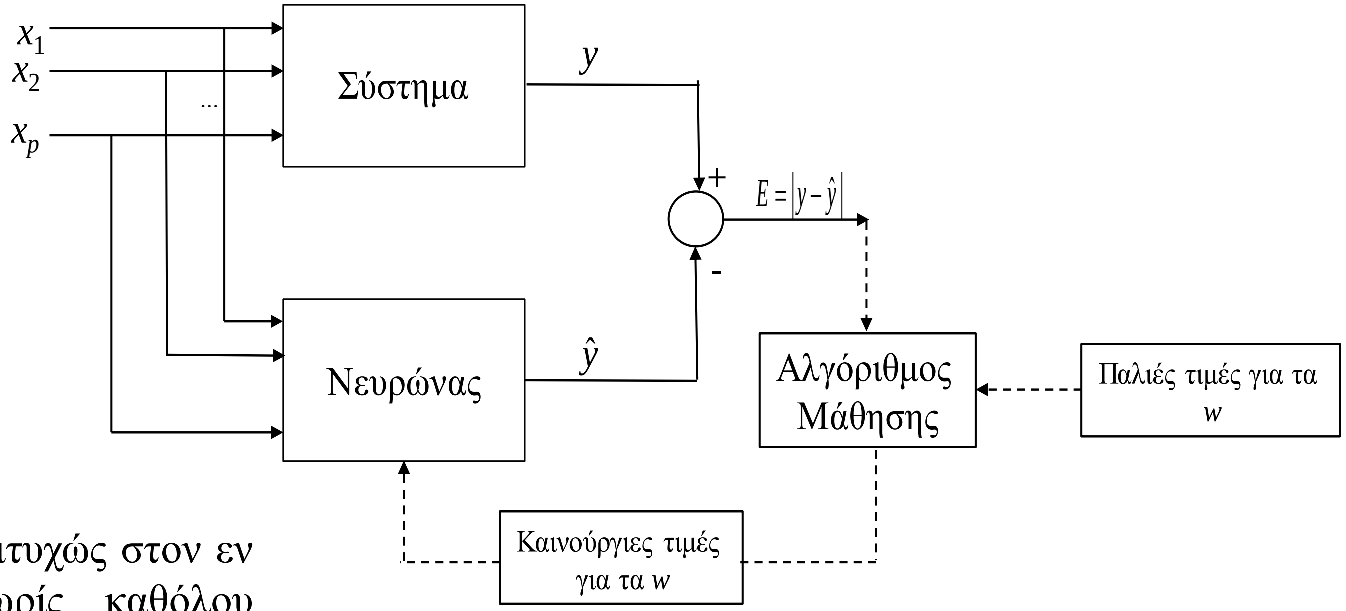
$$E=\sum_{k=1}^{4}E_k=0+0+0+0=0$$

Μηχανική Μάθηση Νευρώνα: Παράδειγμα (Λογικό Ή)

Παρατηρήσεις για τον παραπάνω αλγόριθμο

1). Οι καινούργιες τιμές των συναπτικών βαρών υπολογίζονταν με το να προσθέσουμε μία τιμή στις προηγούμενες τιμές τους. Συγκεκριμένα ο κανόνας μηχανικής μάθησης είχε την παρακάτω μορφή:

$$w_i = w_i + 0.25 \rightarrow w_i^{new} = w_i^{old} + 0.25$$



2). Η στρατηγική αυτή είναι αυθαίρετη, παρόλο που δούλεψε επιτυχώς στον εν λόγω παράδειγμα, το οποίο είναι πάρα πολύ απλό χωρίς καθόλου πολυπλοκότητα. Σε γενικές γραμμές, μία τέτοια στρατηγική θα αποτύχει για πολυπλοκότερα συστήματα με πιο πολλές μεταβλητές εισόδου και πιο πολλά δεδομένα (δηλ. πλειάδες).

3). Στη συνήθη περίπτωση, για κάποια συναπτικά βάρη απαιτείται η τιμή τους να αυξάνεται και για άλλα να μειώνεται. Αυτό σημαίνεί ότι είναι αρκετά πιθανό κάποια συναπτικά βάρη να έχουν, στο τέλος, θετικές τιμές, ενώ κάποια άλλα αρνητικές τιμές. Επίσης η διαφορές των τιμών μεταξύ δυάδων συναπτικών βαρών σε κάποιες περιπτωσεις μπορεί να είναι πολύ μεγάλες ενώ σε άλλες πολύ μικρές.

4). Τέλος, για κάθε αλλαγή στις τιμές των συναπτικών βαρών, λαμβάνονται υπόψη όλα τα δεδομένα. Η τεχνική αυτή ονομάζεται Batch Machine Learning. Μία δεύτερη τεχνική για κάθε δεδομένο αλλάζει και τα συναπτικά βάρη. Η τεχνική αυτή ονομάζεται Sequential Machine Learing (Σειριακή Μηχανική Μάθηση).

Μηχανική Μάθηση Νευρώνα: Κανόνας ΔΕΛΤΑ (DELTA Rule)

Δεδομένων Εισόδου-Εξόδου

X

j

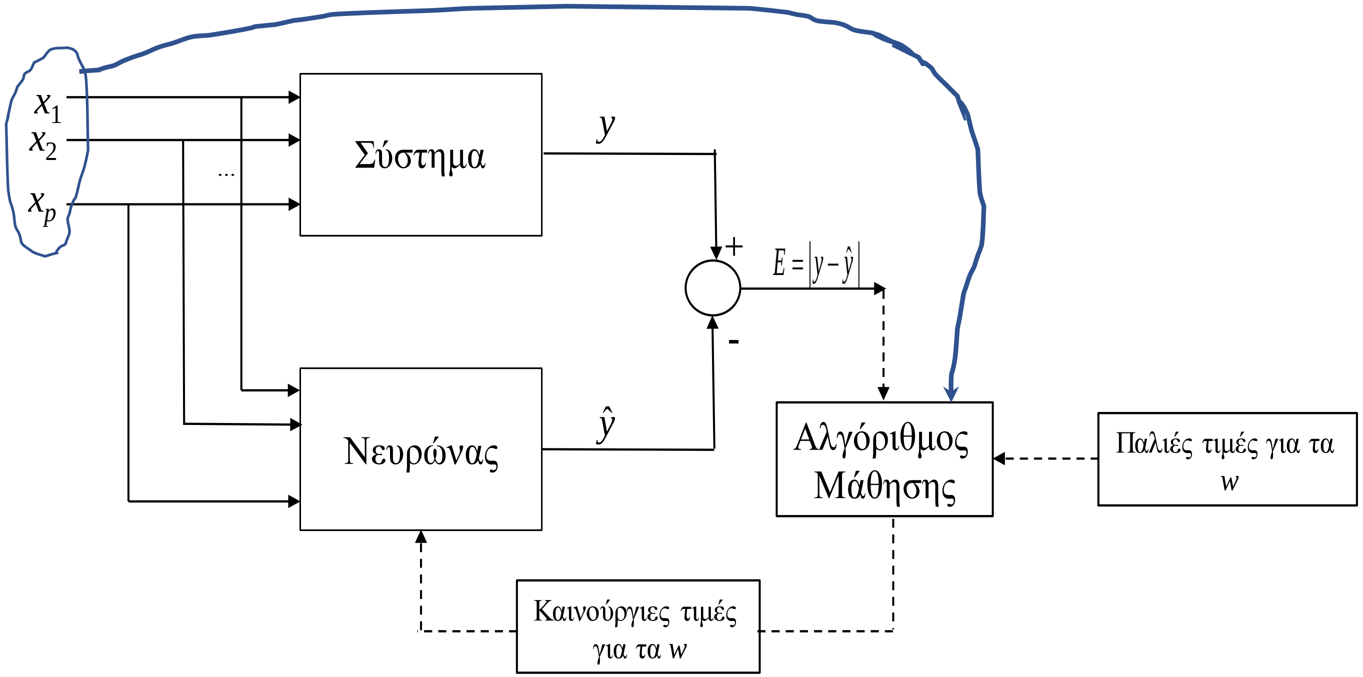
k

	x_1	x_2	…	x_j	…	x_p
1	x_{11}	x_{12}	…	x_{1j}	…	x_{1p}
2	x_{21}	x_{22}	…	x_{2j}	…	x_{2p}
…	…	…	…	…	…	…
k	x_{k1}	x_{k2}	…	x_{kj}	…	x_{kp}
…	…	…	…	…	…	…
N	x_{N1}	x_{N2}	…	x_{Nj}	…	x_{Np}

Y

k

	y
1	y_1
2	y_2
…	…
k	y_k
…	…
N	y_N



x_1

x_2

…

x_p

Σύστημα

y

+

$E = |y - \hat{y}|$

-

Νευρώνας

$\hat{y}$

Αλγόριθμος Μάθησης

Παλιές τιμές για τα w

Καινούργιες τιμές για τα w

Μηχανική Μάθηση Νευρώνα: Κανόνας ΔΕΛΤΑ (DELTA Rule)

Δεδομένων Εισόδου-Εξόδου

X

$j \rightarrow$

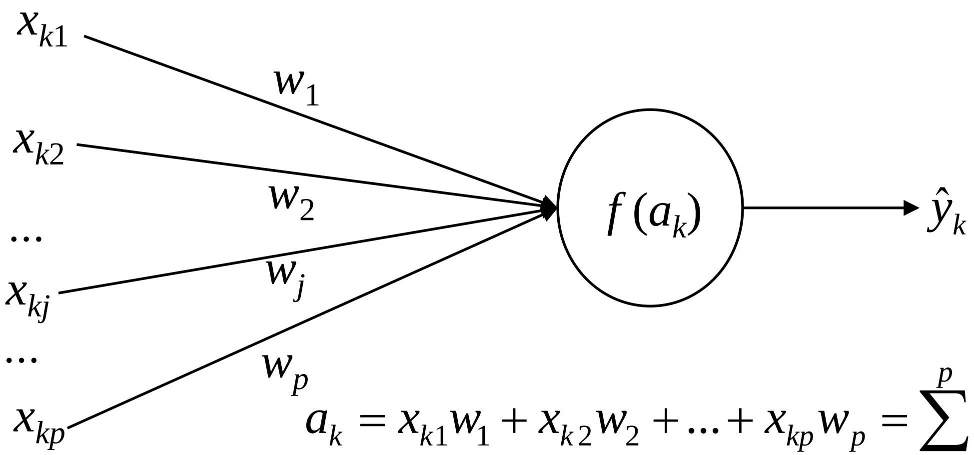
$k \downarrow$

	x_1	x_2	…	x_j	…	x_p
1	x_{11}	x_{12}	…	x_{1j}	…	x_{1p}
2	x_{21}	x_{22}	…	x_{2j}	…	x_{2p}
…	…	…	…	…	…	…
k	x_{k1}	x_{k2}	…	x_{kj}	…	x_{kp}
…	…	…	…	…	…	…
N	x_{N1}	x_{N2}	…	x_{Nj}	…	x_{Np}

Y

$k \downarrow$

	y
1	y_1
2	y_2
…	…
k	y_k
…	…
N	y_N



$$a_k = x_{k1}w_1 + x_{k2}w_2 + \ldots + x_{kp}w_p = \sum_{j=1}^{p} x_{kj}w_j$$

- Ο κανόνας ΔΕΛΤΑ (DELTA rule) είναι ένας κανόνας σειριακής μηχανικής μάθησης
  - Αρχικά οι τιμές των συναπτικών βαρών επιλέγονται τυχαία και για αυτές τις τιμές υπολογίζουμε την έξοδο του νευρώνα και για τα N διανυσματικά δεδομένα εισόδου (είναι μία αρχική εκτίμηση).
  - Πρώτα εισέρχεται το πρώτο (διανυσματικό) δεδομένο εισόδου (δηλ. η πρώτη πλειάδα $x_{11}, x_{12},..., x_{1p}$ και αλλάζουν οι τιμές των συναπτικών βαρών, με βάση τις αρχικές τιμές τους. Υπολογίζεται η πρώτη εκτιμώμενη έξοδος $\hat{y}_1$
  - Μετά εισέρχεται το δεύτερο (διανυσματικό) δεδομένο εισόδου (δηλ. η δεύτερη πλειάδα $x_{21}, x_{22},..., x_{2p}$ και αλλάζουν οι τιμές των συναπτικών βαρών, με βάση τις αρχικές τιμές τους. Υπολογίζεται η δεύτερη εκτιμώμενη έξοδος $\hat{y}_2$
  - ……
  - Μετά εισέρχεται το kοστό (διανυσματικό) δεδομένο εισόδου (δηλ. η kοστή πλειάδα $x_{k1}, x_{k2},..., x_{kp}$ και αλλάζουν οι τιμές των συναπτικών βαρών, με βάση τις αρχικές τιμές τους. Υπολογίζεται η kοστή εκτιμώμενη έξοδος $\hat{y}_k$
  - ……..
  - Μετά εισέρχεται το Nοστό (διανυσματικό) δεδομένο εισόδου (δηλ. η Nοστή πλειάδα $x_{N1}, x_{N2},..., x_{Np}$ και αλλάζουν οι τιμές των συναπτικών βαρών, με βάση τις αρχικές τιμές τους. Υπολογίζεται η Nοστή εκτιμώμενη έξοδος $\hat{y}_N$
  - Υπολογίζεται το συνολικό σφάλμα (E) μεταξύ των πραγματικών και των εκτιμώμενων εξόδων. Αν το E είναι πολύ-πολύ μικρό τότε ο αλγόριθμος σταματάει, αλλιώς ξαναρχίζουμε την παραπάνω διαδικασία από το πρώτο δεδομένο.

Μηχανική Μάθηση Νευρώνα: Κανόνας ΔΕΛΤΑ (DELTA Rule)

Δεδομένων Εισόδου-Εξόδου

X → *j* , ↓ *k*

	x_1	x_2	…	x_j	…	x_p
1	x_{11}	x_{12}	…	x_{1j}	…	x_{1p}
2	x_{21}	x_{22}	…	x_{2j}	…	x_{2p}
…	…	…	…	…	…	…
k	x_{k1}	x_{k2}	…	x_{kj}	…	x_{kp}
…	…	…	…	…	…	…
N	x_{N1}	x_{N2}	…	x_{Nj}	…	x_{Np}

Y ↓ *k*

	y
1	y_1
2	y_2
…	…
k	y_k
…	…
N	y_N

Κανόνας ΔΕΛΤΑ:

$$w_j = w_j + n\left(y_k - \hat{y}_k\right)x_{kj} \rightarrow w_j^{new} = w_j^{old} + n\left(y_k - \hat{y}_k^{old}\right)x_{kj}$$

Βήμα 1. Εισάγουμε του πίνακες **X** και **Y** στο πρόγραμμα

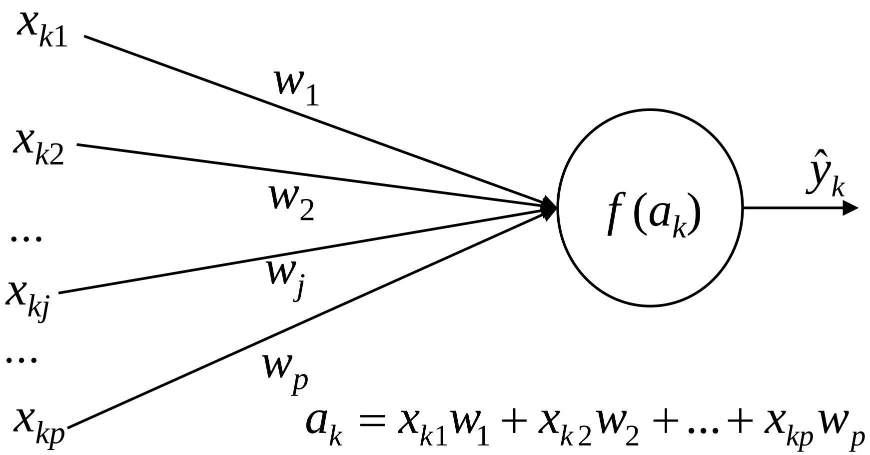
Βήμα 2. Επιλέγουμε μικρές τυχαίες αρχικές τιμές για τα w_j (j=1, 2, …, p)

$$w_j = rand \bullet 0.05$$ (το rand δίνει τυχαίες τιμές μεταξύ 0 και 1)

Υπολογίζουμε για όλες τις πλειάδες τις αντίστοιχες εξόδους.

Υπολογίζουμε και το συνολικό σφάλμα E, το οποίο αναμένεται να είναι μεγάλο γιατί τα συναπτικά βάρη επιλέχθηκαν τυχαία

```
Βήμα 3.  While $E$>0.001
    For $k$=1:$N$
        Υπολόγισε για όλα τα $j$:  $w_j = w_j + n\left(y_k - \hat{y}_k\right)x_{kj}$
        Υπολόγισε το τρέχον $\alpha_k$
        Υπολόγισε την εκτιμώμενη έξοδο: $\hat{y}_k$
    EndFor
    Υπολόγισε το συνολικό σφάλμα $E$
EndWhile
```



$$a_k = x_{k1}w_1 + x_{k2}w_2 + \ldots + x_{kp}w_p = \sum_{j=1}^{p} x_{kj}w_j$$

$$\hat{y}_k = f\left(a_k\right) = f\left(\sum_{j=1}^{p} x_{kj}w_j\right)$$

$$f\left(a\right) = \begin{cases} 1, & \alpha\nu\ a \geq \gamma \\ 0, & \alpha\lambda\lambda\iota\acute{\omega}\varsigma \end{cases}$$

Μηχανική Μάθηση Νευρώνα: Κανόνας ΔΕΛΤΑ (DELTA Rule)

Δεδομένα Εισόδου-Εξόδου

$\boldsymbol{X}$

j →

k ↓

	x_1	x_2	…	x_j	…	x_p
1	x_{11}	x_{12}	…	x_{1j}	…	x_{1p}
2	x_{21}	x_{22}	…	x_{2j}	…	x_{2p}
…	…	…	…	…	…	…
k	x_{k1}	x_{k2}	…	x_{kj}	…	x_{kp}
…	…	…	…	…	…	…
N	x_{N1}	x_{N2}	…	x_{Nj}	…	x_{Np}

$\boldsymbol{Y}$

k ↓

	y
1	y_1
2	y_2
…	…
k	y_k
…	…
N	y_N

Εκτιμώμενη Έξοδος

$\hat{\boldsymbol{Y}}$

k ↓

	$\hat{y}$
1	$\hat{y}_1$
2	$\hat{y}_2$
…	…
k	$\hat{y}_k$
…	…
N	$\hat{y}_N$

Επιμέρους Σφάλματα

k ↓

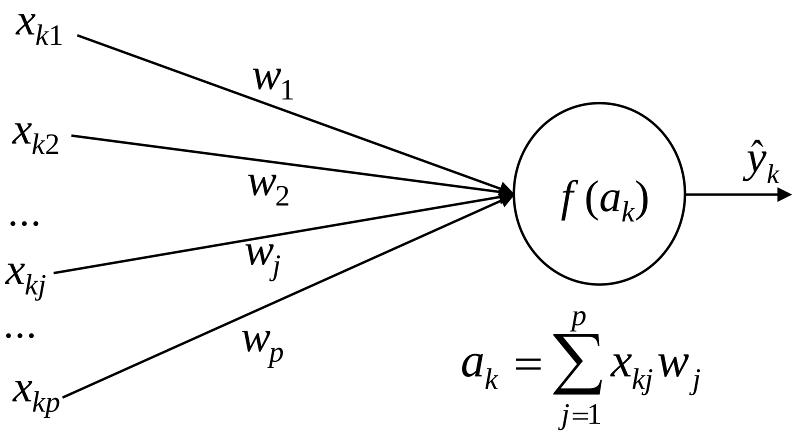
	Er
1	$Er_1 = \lvert y_1 - \hat{y}_1 \rvert$
2	$Er_2 = \lvert y_2 - \hat{y}_2 \rvert$
…	…
k	$Er_k = \lvert y_k - \hat{y}_k \rvert$
…	…
N	$Er_N = \lvert y_N - \hat{y}_N \rvert$

Συνολικό Σφάλμα

$$E = Er_1 + Er_2 + \ldots Er_k + \ldots + Er_N = \sum_{k=1}^{N} Er_k$$

$$E = \lvert y_1 - \hat{y}_1 \rvert + \lvert y_2 - \hat{y}_2 \rvert + \ldots + \lvert y_N - \hat{y}_N \rvert$$

$$E = \sum_{k=1}^{N} \lvert y_k - \hat{y}_k \rvert$$



$$a_k = \sum_{j=1}^{p} x_{kj} w_j$$

Συμπερασματικά:
Τα συναπτικά βάρη αρχικοποιούνται τυχαία και υπολογίζονται οι αντίστοιχες έξοδοι για κάθε δεδομένο εισόδου
Σε κάθε (γενική) επανάληψη, το κάθε δεδομένο εισόδου εισάγεται στον νευρώνα και εφαρμόζεται ο Κανόνας ΔΕΛΤΑ όπου για κάθε είσοδο τα συναπτικά βάρη τροποποιούνται (λαμβάνοντας ότι κάθε δεδομένο εισόδου εισέρχεται κάθε φορά) και υπολογίζεται η αντίστοιχη εκτιμώμενη έξοδος.
Τέλος σε κάθε (γενική) επανάληψη υπολογίζεται το συνολικό σφάλμα E. ΑΝ αυτό είναι μικρό, η διαδικασία σταματάει. Αλλιώς, συνεχίζεται…

Μηχανική Μάθηση Νευρώνα: Κανόνας ΔΕΛΤΑ (DELTA Rule)

```
Read x, y     % Δεδομένα εισόδου-εξόδου
p= …;  N=….;
for j=1:p
     w(j)=rand*0.05;
end
for k=1:N
     a=0;
     for j=1:p
         a=a+x(k,j)*w(j);
     end
     y_est(k)=activation(a);
     Er(k)=abs(y(k)-y_est(k));
end
E=0;
for k=1:N
      E=E+Er(k);
end
```

```
it=1;
while E>0.0001
     it=it+1
     E=0;
     for k=1:N
         for j=1:p,
              w(j)=w(j)+ni*(y(k)-y_est(k))*x(k,j);
         end
         a=0;
         for j=1:p
               a=a+x(k,j)*w(j);
          end
          y_est(k)=activation(a);
          Er(k)=abs(y(k)-y_est(k));
       end
       E=0;
       for  k=1:N
           E=E+Er(k);
       end
end
```