



## Ο αισθητήρας

### Σκοπός

Στο κεφάλαιο αυτό παρουσιάζονται λεπτομερώς τα πρώτα τεχνητά νευρωνικά δίκτυα που αναπτύχθηκαν τη δεκαετία του πενήντα. Δίνεται η δομή τους, η λειτουργία τους, ο τρόπος εκπαίδευσής τους, καθώς επίσης και τα προβλήματα που μπορούν τα δίκτυα αυτά να επιλύσουν. Παρουσιάζονται οι περιορισμοί τους και οι λόγοι για τους οποίους ουσιαστικά τα δίκτυα αυτά εγκαταλήφθηκαν.

### Προσδοκώμενα Αποτελέσματα

Όταν ολοκληρώσετε τη μελέτη του κεφαλαίου αυτού, θα είστε σε θέση να:

- σχεδιάσετε δίκτυα της μορφής του αισθητήρα, ανάλογα με τις ανάγκες του προβλήματος που σας παρουσιάζεται
- εκπαιδεύσετε ένα τέτοιο δίκτυο στα δεδομένα του προβλήματος και να επαληθεύσετε ότι η εκπαίδευση είναι σωστή
- αναγνωρίσετε ποια προβλήματα έχουν την ικανότητα τα δίκτυα αυτά να λύνουν και ποια δεν μπορούν να λύσουν.

### Λέξεις κλειδιά

- Είσοδος δικτύου
- έξοδος δικτύου
- βάρη συνδέσεων
- κατώφλι
- γραμμικώς διαχωρίσιμο (και μη διαχωρίσιμο) πρόβλημα
- πρόβλημα X-OR
- κυρτές περιοχές
- ικανότητα αποθηκεύσης
- εκπαίδευση δικτύου
- κανόνας Δέλτα

- ρυθμός εκπαίδευσης
- σφάλμα, στόχος
- *adaline*
- *madaline*
- διόρθωση βαρών
- δυσκολίες εκπαίδευσης

### Εισαγωγικές Παρατηρήσεις

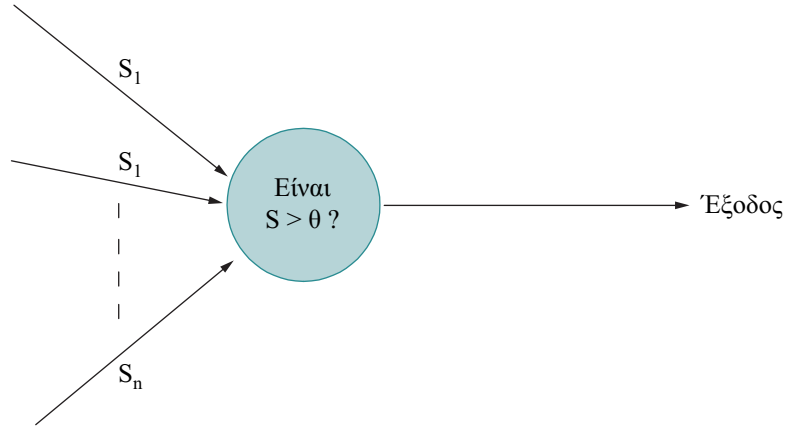
Στο κεφάλαιο αυτό ασχολούμαστε για πρώτη φορά με ένα νευρωνικό δίκτυο συγκεκριμένης μορφής και καλύπτουμε λεπτομερώς όλη την διαδικασία εκπαίδευσης ενός τέτοιου δικτύου. Τόσο τα δίκτυα αυτά όσο και οι διαδικασίες είναι απλής μορφής, αλλά είναι απαραίτητο να τα κατανοήσουμε όσο το δυνατόν καλύτερα, γιατί θα μας είναι χρήσιμα στα επόμενα κεφάλαια.

### 3.1 Τα πρώτα πρότυπα νευρωνικών δικτύων και ο αισθητήρας

Το μοντέλο του αισθητήρα (perceptron) είναι από τα πρώτα μοντέλα νευρωνικών δικτύων που αναπτύχθηκαν την δεκαετία του πενήντα και έδωσαν στην περιοχή αυτή μεγάλη ώθηση χάρη στις επιτυχίες που είχε από την αρχή. Πολλά δίκτυα που αναπτύχθηκαν αργότερα, κατά πολύ πιο περίπλοκα, ξεκίνησαν από την βάση του αισθητήρα. Προτάθηκε το 1958 από τον Rosenblatt [1–3], ο οποίος ήταν ψυχολόγος. Βέβαια καθόσον οι γνώσεις μας για το νευρικό σύστημα του ανθρώπου προόδευαν, οι πρώτες αυτές προσπάθειες φαίνονται τώρα πια ότι ήταν πολύ απλοϊκές. Οι Minsky–Papert έδειξαν [4] το 1969 ότι το πρώτο αυτό πρότυπο έχει πολλούς περιορισμούς. Σήμερα, υπάρχουν πολλές παραλλαγές νευρωνικών δικτύων που βασίζονται στον αισθητήρα και έχουν διαφορετικές δομές, άλλες απλές και άλλες πιο περίπλοκες. Η πιο απλή μορφή είναι ο λεγόμενος στοιχειώδης αισθητήρας (elementary perceptron), Σχήμα 3.1, γιατί αποτελείται από ένα μόνο νευρώνα και είναι το πιο απλό, αυτοδύναμο σύστημα που υπάρχει και επιτελεί μία ορισμένη διεργασία. Το πολύ απλό αυτό πρότυπο μπορεί να κάνει διάφορα χρήσιμα πράγματα, όπως θα δούμε παρακάτω. Ανεβαίνοντας ως προς την πολυπλοκότητα, έχουμε νευρωνικά δίκτυα τα οποία έχουν πολλούς νευρώνες, όπως στο Σχήμα 3.2, οι οποίοι είναι οργανωμένοι σε δύο επίπεδα, ένα επίπεδο στο οποίο εισέρχονται τα σήματα (επίπεδο εισόδου) και ένα επίπεδο όπου βγαίνει το αποτέλεσμα του νευρωνικού δικτύου (επίπεδο εξόδου). Θα δούμε αργότερα και νευρωνικά δίκτυα τα οποία είναι ακόμα πιο περίπλοκα και εκτός των δύο αυτών επιπέδων έχουν και άλλα επίπεδα, τα οποία βρίσκονται συνήθως ανάμεσα σε αυτό της εισόδου και αυτό της εξόδου.

Στον στοιχειώδη αισθητήρα, ο μοναδικός νευρώνας του συστήματος έχει έναν ορισμένο αριθμό συνδέσεων που προέρχονται από άλλους νευρώνες (τους οποίους όμως δεν εξετάζουμε), όπως φαίνονται στο Σχήμα 3.1. Ο νευρώνας είναι ο κύκλος και οι συνδέσεις του είναι οι διάφορες γραμμές. Έχει ένα ορισμένο αριθμό εισόδων αλλά μία μόνο έξοδο. Αυτό σημαίνει ότι η μονάδα αυτή δέχεται πολλές εισόδους,  $s_1, s_2, s_3$  κτλ. αλλά παράγει μία μόνο έξοδο, που όπως φαίνεται στο Σχήμα είναι στα δεξιά του κύκλου. Στο σημείο αυτό χρειάζεται προσοχή: μπορεί ένας νευρώνας να έχει πολλές εξόδους και άρα πολλές γραμμές δεξιά του κύκλου, αλλά όλες οι εξοδοί αυτές πάντα έχουν την ίδια τιμή. Αν, λοιπόν, υπάρχουν πολλές γραμμές–έξοδοι, ποτέ δεν μπορούν να έχουν διαφορετικές τιμές. Αυτό λοιπόν εννοούμε όταν λέμε ότι ένας νευρώνας έχει μία μόνο έξοδο.

Κάθε εισερχόμενο σήμα,  $s_i$ , συνδέεται με τον κεντρικό νευρώνα με ένα βάρος  $w_i$ . Το βάρος  $w$  μας δείχνει κατά κάποιο τρόπο την αλληλεπίδραση μεταξύ των δύο νευρώνων τους οποίους συνδέει. Τώρα που εξετάζουμε εδώ την απλή περίπτωση του ενός νευρώνα, μπορούμε να πούμε ότι το  $w$  είναι η επίδραση του εισερχόμενου σήμα-

**Σχήμα 3.1**

Ο στοιχειώδης αισθητήρας  
(perceptron)

τος με τον νευρώνα αυτό. Αυτό που έχει σημασία δεν είναι η τιμή του βάρους  $w$  από μόνη της ούτε η τιμή του σήματος  $s$ , αλλά είναι το γινόμενο  $s_i w_i$ . Έτσι κάθε  $s_i$  πολλαπλασιάζεται επί το βάρος  $w_i$  που έχει η σύνδεση  $i$  και τελικά αυτό που παρουσιάζεται στον νευρώνα από κάθε εισερχόμενο σήμα είναι το γινόμενο  $s_i w_i$ . Ο αισθητήρας κατόπιν αθροίζει τα γινόμενα αυτά για όλους τους  $n$  όρους (όπου  $n$  είναι ο αριθμός των εισόδων) και θεωρούμε, λοιπόν, ότι λαμβάνει ένα συνολικό σήμα με τιμή:

$$S = \sum_i^n s_i w_i \quad (3.1)$$

Μερικές φορές θεωρούμε ότι, εκτός από τα εισερχόμενα σήματα και τα αντίστοιχα βάρη  $w$ , ο νευρώνας έχει και ένα εσωτερικό βάρος που τον χαρακτηρίζει, και το οποίο πρέπει να ληφθεί υπόψη στην εξίσωση 3.1. Το εσωτερικό αυτό βάρος λέγεται «bias»,  $b$ , ή αλλιώς προδιάθεση ή παράγων προδιάθεσης του νευρώνα. Το βάρος αυτό είναι τελείως ξεχωριστό από τα άλλα βάρη, αλλά δρα με τον ίδιο τρόπο όπως τα άλλα βάρη  $w$  που είδαμε μέχρι τώρα. Θεωρούμε πάντοτε ότι η τιμή του σήματος που περνάει σε όλα τα εσωτερικά βάρη είναι 1. Έτσι, λοιπόν, οι μονάδες του  $b$  θα είναι οι ίδιες με τις μονάδες του γινομένου ( $s \cdot w$ ), ώστε η εξίσωση 3.1 στην πιο γενική της μορφή γίνεται τώρα:

$$S = b + \sum_i^n s_i w_i \quad (3.2)$$

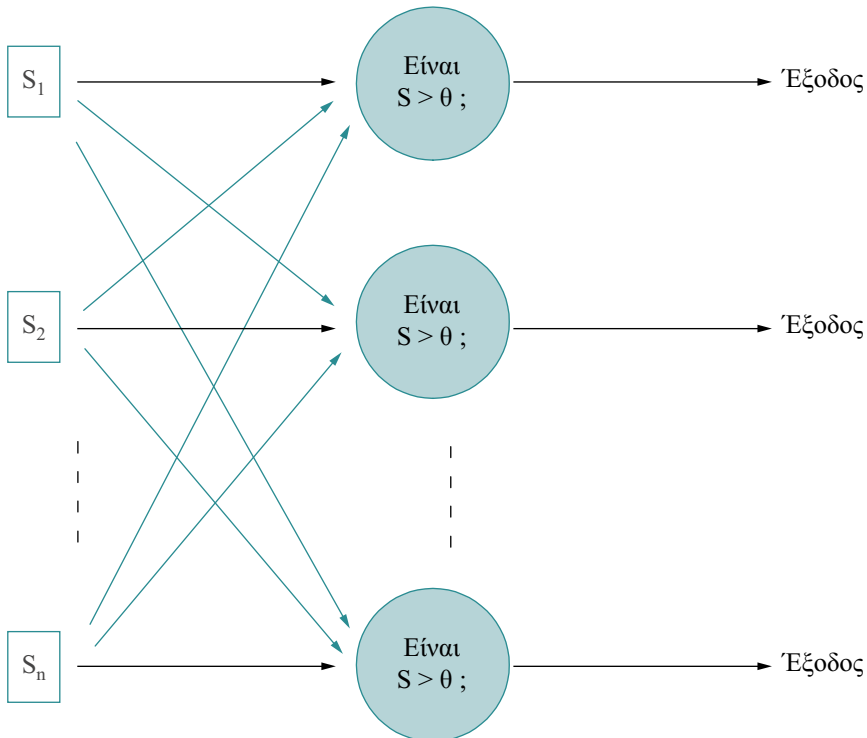
Ο όρος  $b$  δεν έχει καμία φυσική σημασία και δεν πρέπει να αποδίδεται στο εσωτερικό του νευρώνα στον οποίο καταλήγει. Μερικές φορές θεωρείται ότι είναι ένα εξωτερικό ερέθισμα, το οποίο προστίθεται στο υπόλοιπο άθροισμα για να δώσει το σωστό  $S$ . Όλα τα προβλήματα δεν αντιμετωπίζονται με τον ίδιο τρόπο και έτσι άλλες φορές χρησιμοποιείται η εξίσωση 3.1 (χωρίς τον όρο  $b$ ) και άλλες η εξίσωση 3.2 (με

τον όρο  $b$ ), χωρίς να υπάρχει ένας γενικός κανόνας. Ακολουθώς εφαρμόζουμε την συνάρτηση κατωφλίου Heaviside, με μία συγκεκριμένη τιμή του κατωφλίου,  $\theta$ . Η συνάρτηση αυτή έχει δοθεί στο Σχήμα 2.7. Συγκρίνουμε το  $\theta$  με το άθροισμα  $S$ . Εάν  $S > \theta$ , τότε ο αισθητήρας ενεργοποιείται και θεωρούμε ότι πυροδοτεί. Εάν  $S < \theta$ , τότε το άθροισμα  $S$  μηδενίζεται και ο αισθητήρας παραμένει αδρανής. Αυτό συνοψίζεται ως:

$$\text{Εάν } S > \theta \quad \text{τότε η τιμή της εξόδου} = 1 \quad (3.3)$$

$$\text{Εάν } S < \theta \quad \text{τότε η τιμή της εξόδου} = 0 \quad (3.4)$$

Αντιλαμβανόμαστε λοιπόν ότι η ενεργητικότητα του αισθητήρα εξαρτάται από τρεις παραμέτρους: τα βάρη των συνδέσεων, τις τιμές των εισόδων και την τιμή του κατωφλίου. Θεωρούμε ότι αυτό που μαθαίνει το σύστημά μας αποθηκεύεται στα βάρη των συνδέσεων, τα οποία, όπως θα δούμε και παρακάτω, μεταβάλλονται συνεχώς κατά την διάρκεια που το σύστημα «μαθαίνει» κάποια πληροφορία.



**Σχήμα 3.2**

Ο αισθητήρας με  $n$  νευρώνες

Ξεκινώντας από το πρότυπο του στοιχειώδους αισθητήρα μπορούμε να αναπτύξουμε πιο προχωρημένα πρότυπα που περιέχουν αναγκαστικά περισσότερους του ενός νευρώνες. Ένας αισθητήρας με περίπλοκη δομή δίνεται στο Σχήμα 3.2. Στο Σχήμα αυτό οι κόμβοι εισόδου παρίστανται με ένα τετράγωνο. Στους κόμβους αυτούς δεν

γίνεται καμία επεξεργασία του σήματος, αλλά χρησιμεύουν μόνο για να δέχονται το σήμα. Οι υπόλοιποι κόμβοι στους οποίους γίνεται επεξεργασία του σήματος δηλώνονται με ένα κύκλο. Εδώ έχουμε  $n$  νευρώνες, αντί για έναν μόνο που έχει ο στοιχειώδης αισθητήρας. Στην πιο γενική περίπτωση έχουμε πλήρη συνδεσμολογία, δηλ. κάθε εισερχόμενο σήμα  $s_i$  παρουσιάζεται και στους  $n$  νευρώνες, με διαφορετικό βάρος κάθε φορά. Η διαδικασία σύγκρισης με το κατώφλι  $\theta$  είναι η ίδια όπως και στο απλό μοντέλο, αλλά εδώ έχουμε μια πλειάδα από εξόδους των οποίων ο αριθμός είναι  $n$ , όσο δηλ. και ο αριθμός των νευρώνων. Υπάρχουν και διάφορα άλλα παρόμοια μοντέλα, που ονομάζονται επίσης συλλογικά αισθητήρες, και μερικά είναι πιο περίπλοκα από τα παραπάνω, αλλά ο μηχανισμός λειτουργίας τους είναι ο ίδιος όπως αυτός που είδαμε [5].

### Άσκηση Αυτοαξιολόγησης 3.1

Η ενεργητικότητα του αισθητήρα εξαρτάται μόνο από τις τιμές στα:

- (α) βάρη των συνδέσεων, τιμές εισόδου, τιμή κατωφλίου, σφάλμα στην έξοδο
- (β) βάρη των συνδέσεων, τιμές εισόδου, τιμή κατωφλίου
- (γ) βάρη των συνδέσεων, τιμές εισόδου
- (δ) βάρη των συνδέσεων, τιμές κατωφλίου
- (ε) κανένα από τα παραπάνω

### Άσκηση Αυτοαξιολόγησης 3.2

(Σωστό ή Λάθος). Το δίκτυο του Σχήματος 3.1 έχει 3 νευρώνες εισόδου και  $\theta = 0$ . Το διάνυσμα εισόδου είναι  $s = (1, 0,7, 1,6)$  και τα αντίστοιχα βάρη είναι  $w = (0,5, 1,5, -1,0)$ . Η είσοδος και η έξοδος του δικτύου είναι αντίστοιχα:

	Είσοδος	Έξοδος	Σωστό	Λάθος
α)	1	0		
β)	3,3	0		
γ)	-0,05	1		
δ)	1	1		
ε)	-0,05	0		
στ)	0,05	0		

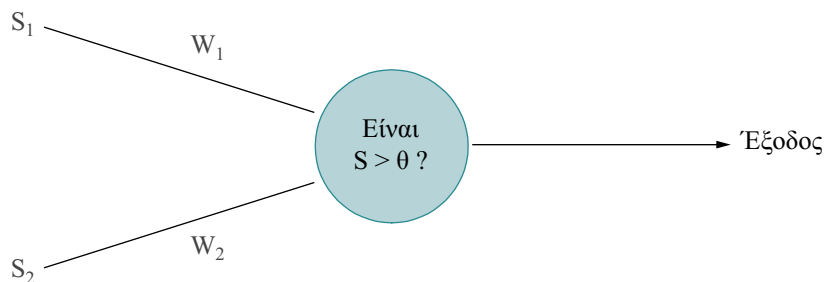
### 3.2 Το πρόβλημα της αποκλειστικής διάζευξης

Ένα από τα πιο γνωστά προβλήματα που επιλύονται με νευρωνικά δίκτυα είναι αυτό της εκμάθησης της συνάρτησης X-OR (exclusive-or), δηλ. της συνάρτησης της αποκλειστικής διάζευξης, όπως λέγεται. Η συνάρτηση αυτή δέχεται δύο εισόδους και δίνει μία έξοδο. Οι εισόδους και η έξοδος μπορεί να είναι 0 ή 1 μόνον και ισχύει ο εξής περιορισμός: Εάν και οι δύο εισόδους είναι ίδιες, τότε η έξοδος είναι 0, εάν είναι διαφορετικές, τότε η έξοδος είναι 1. Οι όροι αυτοί συνοψίζονται στον Πίνακα 3.1, που ονομάζεται και «πίνακας αλήθειας» της συνάρτησης.

**Πίνακας 3.1**

*Η συνάρτηση X-OR*

Είσοδος 1	Είσοδος 2	Έξοδος
0	0	0
0	1	1
1	0	1
1	1	0



**Σχήμα 3.3**

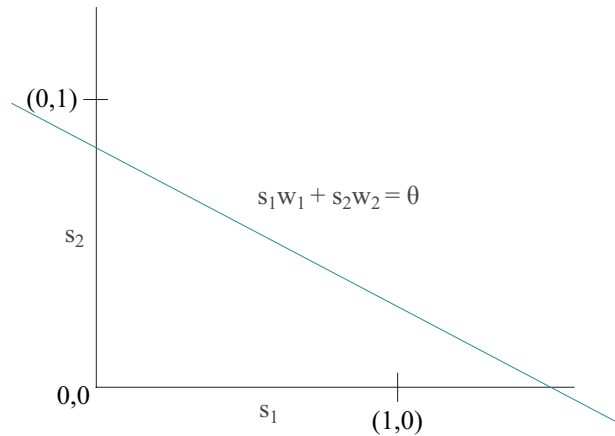
Ο αισθητήρας με δύο εισόδους για το πρόβλημα XOR

Εάν χρησιμοποιήσουμε τον στοιχειώδη αισθητήρα με δύο εισόδους και μία έξοδο τότε έχουμε το Σχήμα 3.3 που είναι μία ειδική περίπτωση του Σχήματος 3.1. Χρησιμοποιώντας το απλό αυτό δίκτυο όλοι οι δυνατοί συνδυασμοί αναπαρίστανται στο διάγραμμα του Σχήματος 3.4, στο επίπεδο  $x-y$ , όπου οι δύο άξονες είναι οι δύο εισόδους  $s_1$  και  $s_2$ . Στο δίκτυο του Σχήματος 3.3 κάθε φορά που έρχονται οι δύο εισόδους  $s_1$  και  $s_2$  τίθεται το ερώτημα της σύγκρισης μεταξύ του  $S$  και του  $\theta$ . Θέλουμε το δίκτυο να δίνει έξοδο = 0, όταν το  $S$  είναι  $S < 0,5$  και να δίνει έξοδο = 1, όταν  $S > 0,5$ . Βλέπουμε όμως καθαρά ότι δεν υπάρχει κανένας συνδυασμός τιμών των  $w_1$  και  $w_2$  που να παράγει τις σχέσεις που περιλαμβάνονται στον Πίνακα 3.1. Ας θεωρήσουμε ότι το κατώφλι  $\theta = 0,5$ . Η αλγεβρική σχέση γίνεται:

$$s_1 w_1 + s_2 w_2 = 0,5 \quad (3.5)$$

**Σχήμα 3.4**

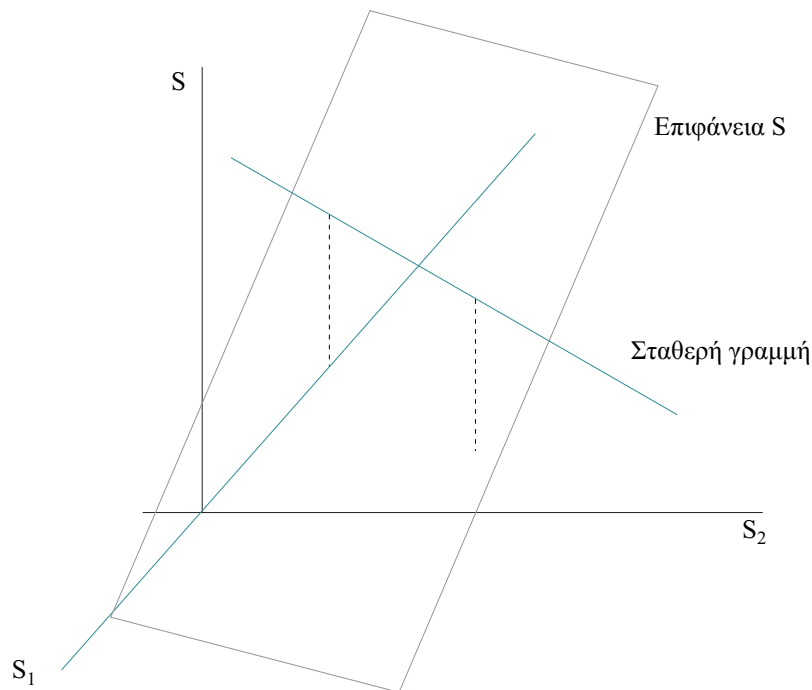
Το πρόβλημα της συνάρτησης  
X-OR σε αναπαράσταση  
στο επίπεδο  $x$ - $y$ .



και περιγράφει το δίκτυό μας. Η εξίσωση αυτή είναι γραμμική ως προς  $s_1$  και  $s_2$ . Αυτό σημαίνει ότι όλες οι τιμές των  $s_1$  και  $s_2$  που ικανοποιούν την εξίσωση αυτή θα βρίσκονται σε μία ευθεία γραμμική στο επίπεδο  $x$ - $y$ , όπως λ.χ. η ευθεία του Σχήματος 3.4. Οποιαδήποτε τιμή αν έχουν τα  $s_1$  και  $s_2$  πάνω στην γραμμή αυτή, θα δώσουν  $S = 0,5$ . Εάν τα  $s_1$  και  $s_2$  βρίσκονται στην μία πλευρά της γραμμής, τότε το  $S > \theta$ , και έξοδος = 1. Αν τα  $s_1$  και  $s_2$  βρίσκονται στην άλλη πλευρά της γραμμής, τότε  $S < \theta$  και έξοδος = 0. Αλλάζοντας τις τιμές  $w_1$  και  $w_2$  καθώς και την τιμή του  $\theta$ , θα αλλάξει η κλίση και η θέση της γραμμής αυτής. Αυτό που θέλουμε εμείς όμως είναι τα σημεία  $(0,0)$  και  $(1,1)$  να βρίσκονται από τη μία πλευρά της ευθείας και τα σημεία  $(0,1)$  και  $(1,0)$  να βρίσκονται από την άλλη. Μόνον τότε το δίκτυο θα δίνει την σωστή απάντηση. Βλέπουμε ότι δεν υπάρχει κανένας τρόπος να τραβήξουμε μία ευθεία γραμμή, με οποιαδήποτε κλίση, που να ικανοποιεί την συνθήκη αυτή. Καταλήγουμε λοιπόν στο συμπέρασμα ότι το δίκτυο του Σχήματος 3.3, ανεξάρτητα από τιμές  $w_1, w_2$ , και  $\theta$ , δεν μπορεί να λύσει το πρόβλημα της συνάρτησης X-OR.

Σε μία διαφορετική απεικόνιση, στο Σχήμα 3.5, έχουμε το επίπεδο  $x$ - $y$ , με άξονες τα δύο σήματα εισόδου  $s_1$  και  $s_2$ , όπως και πριν, και το κεφαλαίο  $S$  στον άξονα των  $z$ . Δημιουργείται έτσι η επιφάνεια  $S$ , κάθε σημείο της οποίας είναι πάνω από το αντίστοιχο σημείο του  $x$ - $y$  επιπέδου και σε απόσταση η οποία δίδεται από την τιμή του  $S$  στο σημείο αυτό. Η κλίση της επιφάνειας αυτής είναι φυσικά σταθερή σε όλο το  $x$ - $y$  επίπεδο. Τα σημεία εκείνα τα οποία δίδουν τιμή του  $S$  ίση με την τιμή κατωφλίου  $\theta$  θα έχουν προβολή μια σταθερή γραμμή στην  $S$  επιφάνεια. Όλα τα σημεία από την μία πλευρά της γραμμής κατωφλίου θα έχουν προβολή στην  $S$  επιφάνεια σε μεγαλύτερες τιμές από την σταθερή γραμμή, ενώ τα σημεία από την άλλη πλευρά θα έχουν προβολή σε μικρότερες τιμές. Έτσι, η γραμμή κατωφλίου διαιρεί το  $x$ - $y$  επίπεδο σε δύο περιοχές. Όλα τα σημεία από την μία πλευρά της γραμμής δίνουν έξοδο = 0 και τα σημεία από την άλλη πλευρά δίνουν έξοδο = 1. Καταλήγουμε, λοιπόν, στο ίδιο συμπέ-





**Σχήμα 3.5**  
Η επιφάνεια  $S$

ρασμα όπως και προηγουμένως με την θεώρηση στις 2 διαστάσεις, ότι δηλαδή ένας απλός αισθητήρας δεν μπορεί να λύσει σωστά το πρόβλημα X-OR.

### 3.3 Γραμμική διαχωρισιμότητα

Υπάρχουν πολλές συναρτήσεις, παρόμοιες με την συνάρτηση του X-OR, οι οποίες δεν μπορούν να παρασταθούν με ένα δίκτυο ενός μόνο νευρώνα. Για όλες αυτές τις συναρτήσεις λέμε ότι είναι γραμμικά μη διαχωρίσιμες.

Είδαμε ότι στην περίπτωση που έχουμε δύο εισόδους, τότε ο διαχωρισμός γίνεται από μία ευθεία γραμμή. Αν το πρόβλημά μας είχε τρεις εισόδους, τότε ο διαχωρισμός θα γινόταν από ένα επίπεδο που θα έτεμνε τον τρισδιάστατο χώρο. Για την περίπτωση τεσσάρων και πάνω εισόδων, πρέπει να δημιουργήσουμε έναν υπερ-χώρο  $n$  διαστάσεων που θα τέμνεται από ένα υπερ-επίπεδο, όπου υπερεπίπεδο θεωρούμε ένα γεωμετρικό σχήμα που διαιρεί τον χώρο σε τέσσερις ή παραπάνω διαστάσεις.

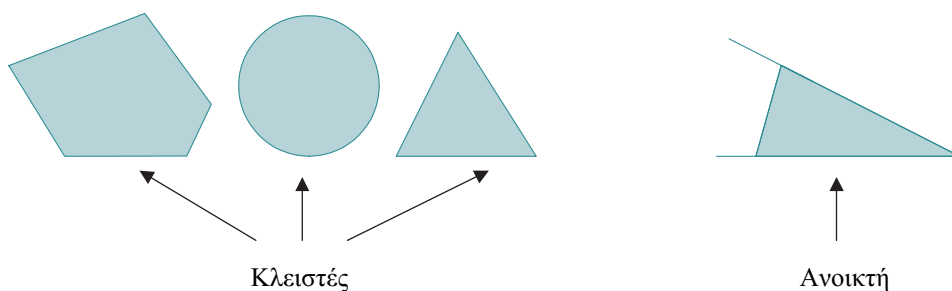
Δεν υπάρχει κανένας απλός τρόπος να ξέρουμε εκ των προτέρων εάν η συνάρτηση που μας παρουσιάζεται είναι γραμμικά διαχωρίσιμη, ειδικά όταν ο αριθμός των μεταβλητών είναι μεγάλος. Ένας νευρώνας με  $n$  εισόδους μπορεί να έχει  $2^n$  διαφορετικούς συνδυασμούς από 0 και 1. Καθόσον κάθε συνδυασμός μπορεί να δώσει δύο διαφορετικές εξόδους (0 ή 1), υπάρχουν  $2^{2^n}$  διαφορετικές συναρτήσεις  $n$  μεταβλητών. Καταλαβαίνουμε, λοιπόν, ότι η πιθανότητα να είναι μία συνάρτηση γραμμικά διαχωρίσιμη είναι πολύ μικρή, όταν μάλιστα υπάρχουν πολλές εισοδοί.

Ήδη από τα πρώτα χρόνια της ανάπτυξης των νευρωνικών δικτύων και της εμφάνισης του προτύπου του αισθητήρα έγινε κατανοητό το πρόβλημα αυτό της γραμμικής διαχωρισιμότητας και οι περιορισμοί τους οποίους εισάγει. Η αδυναμία αυτή του προτύπου του αισθητήρα να λύσει τόσο απλά προβλήματα είναι το μεγαλύτερο μειονέκτημά του. Τα πιο πολλά προβλήματα δεν είναι γραμμικά διαχωρίσιμα και τα λίγα προβλήματα που είναι, έχουν λυθεί πιο εύκολα με άλλους τρόπους. Αυτό ισχύει για προβλήματα κάθε φύσης, τεχνικά και μη. Η αναγνώριση αυτής της δυσκολίας και η ανικανότητα του αισθητήρα να την λύσει, πραγματικά σταμάτησαν την έρευνα στην περιοχή αυτή με τέτοιου είδους πρότυπα, ήδη από την δεκαετία των εξήντα.

Φυσική προέκταση του απλού μοντέλου ήταν να προταθεί ένα πιο περίπλοκο δίκτυο το οποίο να περιέχει περισσότερους νευρώνες, αντί για τις απλές μορφές που είδαμε μέχρι τώρα. Όπως είπαμε, οι νευρώνες αυτοί κατατάσσονται σε επίπεδα. Έτσι έχουμε δίκτυα με δύο επίπεδα νευρώνων αντί για ένα που έχει το στοιχειώδες μοντέλο. Ο αριθμός των επιπέδων μερικές φορές περιλαμβάνει την είσοδο και μερικές όχι. Στα επόμενα Σχήματα στο κεφάλαιο αυτό δεν περιλαμβάνεται το επίπεδο της εισόδου. Αυτό όμως δεν συμβαίνει πάντα, αλλά όπως καταλαβαίνουμε είναι θέμα ονοματολογίας και μόνον. Το μοντέλο δύο επιπέδων μπορεί να ξεχωρίσει σημεία που περιλαμβάνονται σε ανοιχτές ή κλειστές κυρτές περιοχές. Κυρτή περιοχή είναι η περιοχή στην οποία οποιαδήποτε δύο σημεία στην περιοχή αυτή μπορούν να ενωθούν από μία ευθεία γραμμή, η οποία ανήκει εξ ολοκλήρου στην περιοχή αυτή. Κλειστή κυρτή περιοχή είναι μία περιοχή στην οποία όλα τα σημεία περιέχονται μέσα στα όρια της (λ.χ. ο κύκλος), ενώ ανοιχτή είναι η περιοχή στην οποία μερικά σημεία βρίσκονται έξω από τα προκαθορισμένα όρια (λ.χ. περιοχή μεταξύ δύο παράλληλων γραμμών). Διάφορα παραδείγματα με τέτοιες περιοχές φαίνονται στο Σχήμα 3.6.

Θεωρούμε ένα νευρωνικό δίκτυο με δύο επίπεδα και με δύο εισόδους που έρχονται στους δύο νευρώνες του πρώτου επιπέδου, όπως στο Σχήμα 3.7. Οι δύο νευρώνες του πρώτου επιπέδου συνδέονται με ένα νευρώνα του δεύτερου επιπέδου. Ας υποθέσουμε ότι  $\theta =$

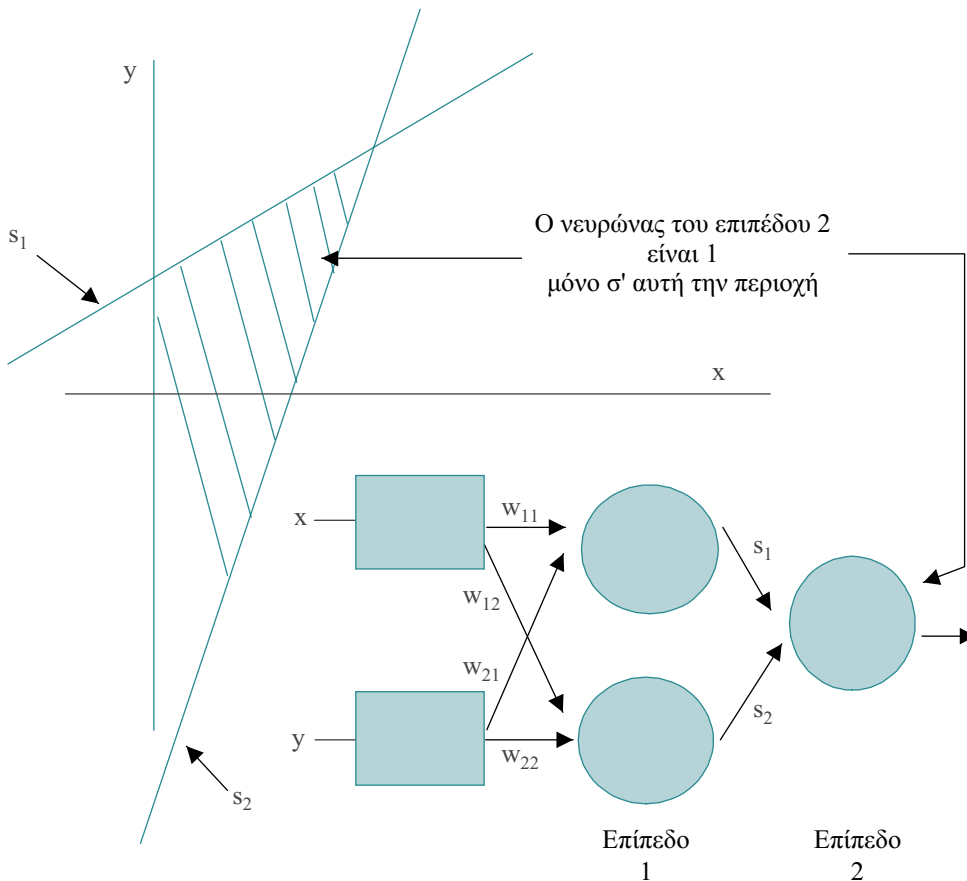
### ΚΥΡΤΕΣ ΠΕΡΙΟΧΕΣ



**Σχήμα 3.6**

Κυρτές περιοχές,  
ανοιχτές και κλειστές

0,75 για τον νευρώνα του δεύτερου επιπέδου και ότι τα βάρη είναι και τα δύο ίσα προς 0,5. Στην περίπτωση αυτή, εάν η έξοδος των δύο νευρώνων του πρώτου επιπέδου είναι 1, τότε παράγεται ως έξοδος το 1 και απο τον νευρώνα του δεύτερου επιπέδου. Αυτό όμως είναι ανάλογο ως ο νευρώνας εξόδου να εκτελεί την συνάρτηση για την λογική σύζευξη. Στο Σχήμα 3.7 θεωρούμε ότι κάθε νευρώνας του πρώτου επιπέδου διαιρεί το  $x$ - $y$  επίπεδο με τέτοιο τρόπο ώστε ο πρώτος από τους δύο νευρώνες να δίδει έξοδο = 1 για εισόδους κάτω από την πάνω γραμμή, και ο άλλος νευρώνας να δίδει έξοδο = 1 για εισόδους πάνω από την κάτω γραμμή. Μετά από αυτόν τον διπλό χωρισμό παρατηρούμε ότι η τελική έξοδος του δικτύου είναι 1 μόνον μέσα στην σκιασμένη περιοχή σχήματος V. Αν είχαμε χρησιμοποιήσει τρεις νευρώνες στο επίπεδο εισόδου, τότε θα είχαμε τρεις ευθείες τεμνόμενες γραμμές, οι οποίες δίνουν μια περιοχή σε σχήμα τριγώνου. Για περισσότερους νευρώνες θα δημιουργείται ένα πολύγωνο με ανάλογο αριθμό πλευρών. Όλα τα πολύγωνα αυτά είναι κυρτά επειδή σχηματίζονται από τις περιοχές της λογικής σύζευξης και έτσι περιοχές που είναι κυρτές μπορούν να περιληφθούν. Σημεία που δεν μπορούν να περιληφθούν σε μία κυρτή περιοχή δεν μπορούν να διαχωριστούν από τα άλλα σημεία στο επίπεδο από ένα δίκτυο μόνο με νευρώνες εισόδου-εξόδου.

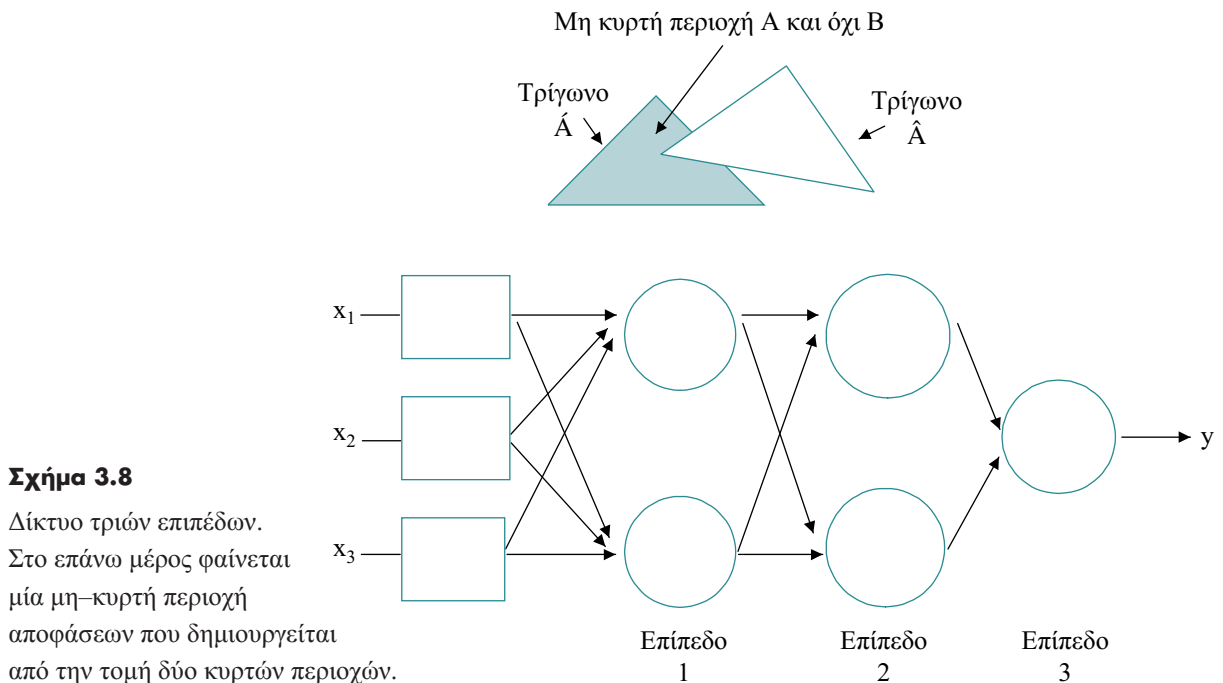


**Σχήμα 3.7**  
Κυρτή περιοχή απόφασης που παράγεται από σύστημα δύο επιπέδων

Ο νευρώνας του δεύτερου επιπέδου μπορεί να αναπαραστήσει και άλλες συναρτήσεις, εκτός από την λογική διάζευξη. Για να γίνει αυτό θα πρέπει τα  $w$  και τα  $\theta$  να επιλεγούν σωστά. Για να αναπαραστήσει την λογική διάζευξη θα πρέπει τουλάχιστον ένας από τους νευρώνες του πρώτου επιπέδου να έχει έξοδο  $= 1$ . Υπάρχουν 16 δυαδικές συναρτήσεις δύο παραμέτρων. Αν επιλεγούν σωστά τα  $\theta$  και  $w$ , ένα δίκτυο με δύο επίπεδα μπορεί να αναπαραστήσει τις 14 από αυτές, δηλ. όλες εκτός από το X-OR και το X-NOR. Οι 14 αυτές είναι γραμμικά διαχωρίσιμες, ενώ οι δύο είναι μη διαχωρίσιμες.

Δεν είναι απαραίτητο οι είσοδοι να έχουν δυαδικές τιμές. Ένα διάνυσμα εισόδων με συνεχείς τιμές μπορεί να αναπαραστήσει ένα σημείο οπουδήποτε στο  $x$ - $y$  επίπεδο. Στην περίπτωση αυτή το δίκτυο υποδιαιρεί το επίπεδο σε συνεχείς περιοχές, κατ' αντίθεση στο να βγάξει ως έξοδο 0 ή 1. Η γραμμική διαχωρισιμότητα όμως σε κάθε περίπτωση απαιτεί ότι η έξοδος του νευρώνα στο δεύτερο επίπεδο περιέχεται σε ένα τμήμα του  $x$ - $y$  επιπέδου που περικλείεται από ένα κυρτό πολύγωνο. Αν έχουμε δηλ. δύο περιοχές  $P$  και  $Q$  που πρέπει να διαχωρισθούν, τότε όλα τα σημεία της περιοχής  $P$  πρέπει να περιέχονται σε ένα κυρτό πολύγωνο το οποίο δεν περιέχει κανένα σημείο του  $Q$ , και αντιθέτως.

Στο Σχήμα 3.8 έχουμε ένα δίκτυο με τρία επίπεδα. Οι ικανότητές του εξαρτώνται από τον αριθμό των νευρώνων και από τα βάρη  $w$ . Εδώ δεν υπάρχουν περιορισμοί



κυρτότητας. Ο νευρώνας του τρίτου επιπέδου δέχεται ως είσοδο μια ομάδα κυρτών πολυγώνων και ο λογικός συνδυασμός τους δεν είναι απαραίτητο να είναι κυρτός. Στο Σχήμα αυτό βλέπουμε δύο τρίγωνα, το Α και το Β. Τα δύο αυτά τρίγωνα συνδέονται με την συνάρτηση «Α και όχι Β» και έτσι ορίζεται μια μη-κυρτή περιοχή. Όταν προσθέτουμε περισσότερους νευρώνες, ο αριθμός των πλευρών των πολυγώνων αυξάνεται χωρίς περιορισμό. Αυτό έχει ως αποτέλεσμα να μπορούμε να περικλείουμε μια περιοχή οποιουδήποτε σχήματος, με όσο μεγάλη ακρίβεια θέλουμε. Επιπλέον, δεν είναι απαραίτητο να τέμνονται (δηλ. να έχουν κοινά σημεία) οι περιοχές εξόδου του δεύτερου επιπέδου. Είναι δυνατόν να περικλείουμε διάφορες περιοχές, κυρτές και μη, και να δίνει το δίκτυο έξοδο = 1 για κάθε περίπτωση που το διάλυσμα εισόδου βρίσκεται μέσα σε κάποια από αυτές.

Βλέπουμε, λοιπόν, από τα παραπάνω ότι έχει μεγάλη σημασία να ξεφύγουμε από την περίπτωση του ενός νευρώνα του στοιχειώδους αισθητήρα προς τα μοντέλα που έχουν πολλά επίπεδα. Παρόλα αυτά για πολλά χρόνια δεν υπήρχε κανένας επιτυχής αλγόριθμος εκπαίδευσης τέτοιων δικτύων και, επομένως, για ένα μεγάλο χρονικό διάστημα τα πράγματα που μπορούσαν να κάνουν τα δίκτυα αυτά ήταν και πάλι περιορισμένα, καθώς επίσης και τα προβλήματα που έλυναν με επιτυχία.

### 3.4 Ικανότητα αποθήκευσης

Μία απλή σκέψη που δημιουργήθηκε από την αρχή που αναπτύχθηκαν τα νευρωνικά δίκτυα είναι το κατά πόσο θα μπορούσαν να χρησιμοποιηθούν ως στοιχεία αποθήκευσης, όπως είναι λ.χ. η μνήμη των ηλεκτρονικών υπολογιστών. Η αποθήκευση αυτή θα γίνονταν στα  $w$  (δηλ. στα βάρη) που ενώνουν τους νευρώνες, με το να πάρουν τα  $w$  τις κατάλληλες τιμές. Δημιουργήθηκε η σκέψη μήπως ο αριθμός των bits που απαιτώνται για να αποθηκεύσουμε μία πληροφορία στα  $w$  του αισθητήρα είναι πολύ μικρότερος από αυτόν της συνηθούς μνήμης του υπολογιστή. Από το βιβλίο όμως των Minsky–Pappert φάνηκε ότι ο αριθμός αυτός αυξάνεται πάρα πολύ γρήγορα, ταχύτερα από εκθετικά, με το μέγεθος του προβλήματος. Αυτό απαιτεί υπερβολικά μεγάλη μνήμη. Το συμπέρασμα τότε αναγκαστικά είναι ότι τα συστήματα αυτά περιορίζονται σε προβλήματα μικρού μεγέθους. Πάντως δεν υπάρχει ποσοτική σχέση για τα νευρωνικά δίκτυα που να συσχετίζει τις παραμέτρους αυτές.

### 3.5 Η εκπαίδευση του αισθητήρα

Το γεγονός ότι τα νευρωνικά δίκτυα έχουν την ικανότητα να μαθαίνουν και να εκπαιδεύονται, είναι ίσως το πιο σημαντικό τους χαρακτηριστικό. Όπως και τα βιολογικά δίκτυα, έτσι και τα τεχνητά δίκτυα μεταβάλλονται από την εμπειρία που αποκτούν στην προσπάθειά τους να δώσουν ως έξοδο το ζητούμενο σωστό αποτέλεσμα.

Όπως είδαμε νωρίτερα, μπορούμε να αποφανθούμε εάν ένα δίκτυο ενός επιπέδου μπορεί να αναπαραστήσει μία συνάρτηση, χρησιμοποιώντας το κριτήριο της γραμμικής διαχωρισιμότητας. Όταν η απάντηση είναι θετική, τότε μόνον το δίκτυο μπορεί να εκπαιδευθεί. Η εκπαίδευση του δικτύου συνίσταται στο να μπορέσουμε να βρούμε τις κατάλληλες τιμές των  $w$  και  $\theta$  και τότε το δίκτυο θα μπορεί να αναγνωρίζει τα πρότυπα στα οποία έχει εκπαιδευθεί. Μία τέτοια διαδικασία που να δίνει τα σωστά  $w$  και  $\theta$  προτάθηκε για πρώτη φορά από τον Rosenblatt.

Όπως είδαμε στο προηγούμενο κεφάλαιο, η εκπαίδευση μπορεί να είναι είτε εποπτευόμενη είτε μη εποπτευόμενη. Η εκπαίδευση του αισθητήρα είναι εποπτευόμενου τύπου. Ο αλγόριθμος για την εκπαίδευση μπορεί να δημιουργηθεί με πρόγραμμα στον υπολογιστή. Έχει την εξής διαδικασία: Στην αρχή της μάθησης το σύστημα δεν έχει καμία προηγούμενη γνώση. Τα βάρη  $w_i$  πρέπει να έχουν τυχαίες τιμές, λ.χ. έχουν τιμές οι οποίες δίνονται από μία κατανομή ψευδοτυχαίων αριθμών, και είναι όλα λ.χ. στο διάστημα  $0 < w < 1$ . Όταν παρουσιάζουμε τα πρότυπα στο νευρωνικό δίκτυο, τότε το σύστημα μαθαίνει την πληροφορία που έχει μέσα του κάθε πρότυπο με το να μεταβάλλει τα βάρη του προς την σωστή κατεύθυνση. Η διαδικασία αυτή είναι ανάλογη με την εξάσκηση που υφίσταται ένα βιολογικό σύστημα όταν μαθαίνει μια διεργασία. Η μεταβολή των βαρών συνεχίζεται μέχρις ότου το σύστημα μάθει το σήμα που του δόθηκε. Όταν συμβεί αυτό τότε σταματάει η μεταβολή των  $w$  και οι τελικές τιμές τους αποθηκεύονται και χρησιμοποιούνται περαιτέρω. Στο σημείο αυτό λέμε ότι το δίκτυο έχει εκπαιδευθεί και έχει μάθει τα πρότυπα που του διδάξαμε. Η διαδικασία αυτή, όπως παρουσιάστηκε, είναι γενική και αποτελεί τον συνηθισμένο τρόπο εκπαίδευσης των νευρωνικών δικτύων. Αυτό όμως που διαφέρει είναι η τεχνική με την οποία αλλάζουμε τα  $w$ . Ανάλογα με την δομή των δικτύων έχουν αναπτυχθεί διάφορες τεχνικές. Αμέσως παρακάτω θα δούμε την περίπτωση της μεθόδου του κανόνα Δέλτα.

### 3.6 Η διαδικασία εκπαίδευσης

Όλα τα νευρωνικά δίκτυα, συμπεριλαμβανομένου και του στοιχειώδους αισθητήρα, τα οποία υποβάλλονται σε μία διαδικασία εκπαίδευσης, ξεκινούν από μία κατάσταση κατά την οποία δεν έχουν καμία απολύτως γνώση για το πρόβλημα το οποίο θα μελετήσουν. Κατά την διάρκεια της εκπαίδευσης παρουσιάζονται τα διάφορα πρότυπα, παραδείγματα (patterns), τα οποία ο αισθητήρας πρέπει να μάθει να αναγνωρίζει. Τα παραδείγματα αυτά αποτελούν την ομάδα παραδειγμάτων εκπαίδευσης. Κάθε ομάδα αποτελείται από δύο τμήματα: Πρώτα είναι το τμήμα που περιλαμβάνει τα σήματα εισόδου, δηλ. οι τιμές  $s_1, s_2, s_3$  κτλ., τα οποία είναι τα σήματα

που παρουσιάζονται στο νευρωνικό δίκτυο στην είσοδο του, δηλ. στο πρώτο επίπεδο νευρώνων. Κατόπιν δίνεται το τμήμα που περιλαμβάνει τους στόχους εκπαίδευσης, αυτό δηλαδή το οποίο είναι το επιθυμητό αποτέλεσμα, και είναι τα σήματα εξόδου. Σε κάθε ομάδα εισερχομένων σημάτων αντιστοιχεί ένας μόνον στόχος, δηλ. υπάρχει μια μόνο σωστή απάντηση, και για όλα τα σήματα υπάρχει αντιστοίχια εισόδων-εξόδων.

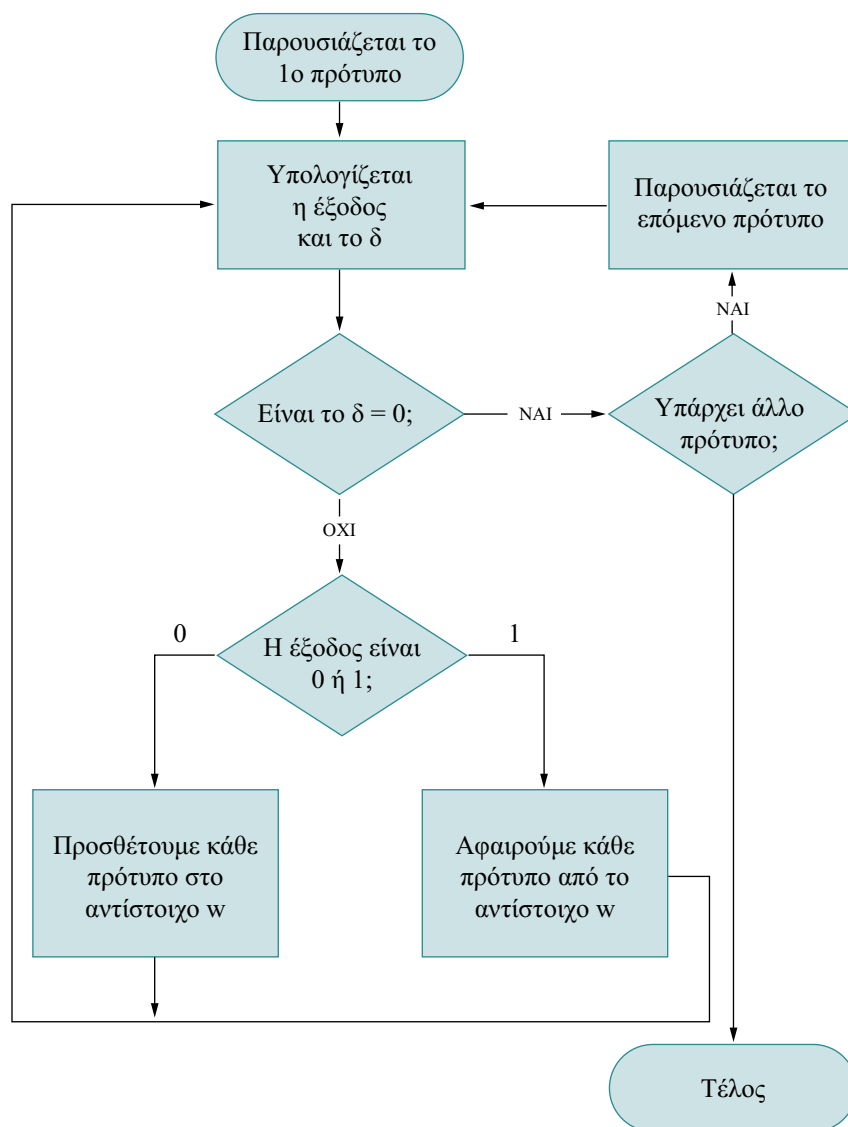
Ένα παράδειγμα τέτοιων σημάτων δίδεται στο Σχήμα 3.9. Κάθε πρότυπο είναι ένα ζεύγος διανυσμάτων που αποτελείται από το διάνυσμα εισόδου και το διάνυσμα εξόδου. Το επίπεδο εισόδου έχει 8 νευρώνες, ενώ το επίπεδο εξόδου έχει 5 νευρώνες. Στο πρόβλημα αυτό υπάρχουν 5 πρότυπα (patterns) προς εκπαίδευση. Στο πρώτο πρότυπο η τιμή (11000000) στην είσοδο πρέπει να δίνει ως αποτέλεσμα στην έξοδο (στόχο) την τιμή (10000) κοκ. Κάθε φορά που παρουσιάζουμε ένα πρότυπο στην είσοδο ακολουθείται η γνωστή διαδικασία. Οι νευρώνες υπολογίζουν το άθροισμα  $S_i$  το συγκρίνουν με το κατώφλι  $\theta$  και δίνουν στην έξοδο την πέπουσα τιμή. Η τελική έξοδος του δικτύου συγκρίνεται με τη σωστή έξοδο, αυτή δηλ. που πρέπει να δίνει το δίκτυο, η οποία είναι γνωστή εκ των προτέρων, τον στόχο. Υπολογίζεται η διαφορά των δύο και το δίκτυο χρησιμοποιεί την διαφορά αυτή για να διορθώσει τις τιμές των  $w$ . Η διόρθωση γίνεται με τέτοιο τρόπο ώστε το δίκτυο συνολικά να δώσει ως έξοδο την επόμενη φορά μια τιμή που είναι πιο κοντά στην επιθυμητή.

ΕΙΣΟΔΟΣ	ΕΠΙΘΥΜΗΤΗ ΕΞΟΔΟΣ	
<div> <div>1</div> <div>1</div> <div>0</div> <div>0</div> <div>0</div> <div>0</div> <div>0</div> <div>0</div> </div>	<div> <div>1</div> <div>0</div> <div>0</div> <div>0</div> <div>0</div> </div>	Κατηγορία 1
<div> <div>0</div> <div>0</div> <div>1</div> <div>1</div> <div>1</div> <div>0</div> <div>0</div> <div>0</div> </div>	<div> <div>0</div> <div>1</div> <div>0</div> <div>0</div> <div>0</div> </div>	Κατηγορία 2
<div> <div>0</div> <div>0</div> <div>0</div> <div>0</div> <div>1</div> <div>1</div> <div>1</div> <div>1</div> </div>	<div> <div>0</div> <div>0</div> <div>1</div> <div>0</div> <div>0</div> </div>	Κατηγορία 3
<div> <div>1</div> <div>0</div> <div>1</div> <div>0</div> <div>1</div> <div>0</div> <div>1</div> <div>0</div> </div>	<div> <div>0</div> <div>0</div> <div>0</div> <div>1</div> <div>0</div> </div>	Κατηγορία 4
<div> <div>1</div> <div>1</div> <div>0</div> <div>0</div> <div>1</div> <div>1</div> <div>0</div> <div>0</div> </div>	<div> <div>0</div> <div>0</div> <div>0</div> <div>0</div> <div>1</div> </div>	Κατηγορία 5

**Σχήμα 3.9**  
Πρότυπο και  
ο αντίστοιχος στόχος

Ο σκοπός της εκπαίδευσης είναι να βρεθεί μία μοναδική ομάδα τιμών των  $w$ , που όταν βρεθεί και χρησιμοποιηθεί, τότε το δίκτυο θα βρίσκει την σωστή τιμή για κάθε πρότυπο. Μετά την εκπαίδευση τα  $w$  δεν αλλάζουν καθόλου.

Η εκπαίδευση με την μέθοδο του κανόνα Δέλτα ακολουθεί το διάγραμμα ροής στο Σχήμα 3.10. Ο αλγόριθμος αυτός όπως φαίνεται στο Σχήμα, περιλαμβάνει ένα μεγάλο αριθμό κύκλων. Ως ένα κύκλο θεωρούμε όλη την διεργασία που ακολουθούμε από την παρουσίαση των τιμών όλων των προτύπων στην είσοδο μέχρι τον στόχο στην έξοδο, και ακολούθως την διόρθωση των τιμών των  $w$  με τον κατάλληλο τρόπο. Όταν ακολουθηθεί ο αλγόριθμος αυτός μετά από ορισμένους κύκλους (περάσματα) το δίκτυο θα μάθει να αναγνωρίζει τα πρότυπα και να δίνει κάθε φορά τη σωστή απάντηση.



**Σχήμα 3.10**

Αλγόριθμος  
εκπαίδευσης  
του δικτύου



Σύμφωνα με τον κανόνα Δέλτα, ορίζουμε ως παράμετρο  $\delta$  τη διαφορά εξόδου και στόχου δηλ.

$$\delta = t - o \quad (3.6)$$

όπου « $t$ » είναι ο στόχος (αρχικό γράμμα από τη λέξη target) και « $o$ » η έξοδος (αρχικό γράμμα από την λέξη output), που δίδεται μια δεδομένη στιγμή. Υπολογίζεται πρώτα το  $\delta$  σύμφωνα με την εξίσωση (3.6). Εάν  $\delta = 0$ , τότε η έξοδος είναι σωστή και δεν γίνεται καμία διόρθωση (αυτό αντιστοιχεί σε απάντηση «Ναι» στο διάγραμμα ροής). Εάν  $\delta > 0$  ή  $\delta < 0$ , τότε θα γίνει διόρθωση (οι περιπτώσεις αυτές αντιστοιχούν σε απάντηση «Όχι» στο ερώτημα). Ακολούθως τίθεται το ερώτημα εάν η έξοδος είναι 0 ή 1. Εάν είναι 0, τότε η περίπτωση αντιστοιχεί σε  $\delta > 0$ , οπότε προσθέτουμε την τιμή κάθε εισόδου στο αντίστοιχο  $w$ . Εάν η έξοδος είναι 1, τότε έχουμε  $\delta < 0$  και αφαιρούμε την τιμή κάθε εισόδου από το αντίστοιχο  $w$ . Υπολογίζουμε τώρα την ποσότητα  $\Delta$ .

$$\Delta_i = \eta \delta x_i \quad (3.7)$$

όπου  $x_i$  είναι η τιμή του σήματος εισόδου και  $\eta$  είναι μια σταθερά που δίδει τον «ρυθμό εκπαίδευσης». Ακολούθως:

$$w_i(n+1) = w_i(n) + \Delta_i \quad (3.8)$$

όπου  $w_i(n)$  είναι η τιμή του βάρους πριν την διόρθωση στο βήμα  $n$ ,  $w_i(n+1)$  είναι η τιμή του βάρους μετά την διόρθωση στο βήμα  $n+1$ , δηλ. η διορθωμένη τιμή από το βήμα  $n$  στο  $n+1$  και  $\Delta$  είναι το ποσό της διόρθωσης. Ο κανόνας αυτός μεταβάλλει ένα βάρος  $w_i$  μόνον αν το σήμα  $x_i = 1$ , αλλά δεν το μεταβάλλει αν  $x_i = 0$ , διότι τότε  $\Delta_i = 0$ . Επίσης, θα πρέπει  $\eta \neq 0$ , για να γίνει οποιαδήποτε μεταβολή. Η τιμή του  $\eta$  είναι συνήθως  $0 < \eta < 1$ . Ο χρόνος εκπαίδευσης είναι μεγάλος αν το  $\eta$  είναι μικρό, ενώ μικραίνει όταν το  $\eta$  είναι μεγαλύτερο.

Όταν παρουσιάσουμε στο δίκτυο όλα τα πρότυπα παρατηρούμε ότι το δίκτυο αρχίζει να εκπαιδεύεται και λέμε ότι το δίκτυο μαθαίνει σταδιακά τα πρότυπα τα οποία του παρουσιάζονται. Χρησιμοποιούμε τον όρο «μαθαίνει» ως συνώνυμο του «εκπαιδεύεται», εννοώντας ότι το δίκτυο αποκτά την ικανότητα να λύνει κάποιο πρόβλημα. Η εκπαίδευση αυτή δεν γίνεται σε ένα βήμα, αλλά ακολουθεί μία διαδικασία πολλών κύκλων, μια διαδικασία η οποία επαναλαμβάνεται πολλές φορές και κατά την οποία το δίκτυο καλυτερεύει συνεχώς τις τιμές των βαρών του. Μετά από μερικούς κύκλους, που μπορεί και να είναι πολλές χιλιάδες, το δίκτυο έχει ήδη βρεί τις κατάλληλες τιμές των  $w$  και έτσι έχει αναπτύξει τις ικανότητες του. Ακολούθως όμως η πρόοδος αυτή σταματάει και λέμε ότι το δίκτυο έχει πλέον συγκλίνει. Αυτό σημαίνει ότι οι τιμές των  $w$  δεν αλλάζουν πλέον και αν όλα έχουν προχωρήσει καλά με τη

σύγκλιση, το δίκτυο έχει εκπαιδευθεί σωστά. Υπάρχει όμως και η περίπτωση στο σημείο αυτό να μην έχει επέλθει η εκπαίδευση. Θα δούμε αργότερα τους λόγους για τους οποίους ένα νευρωνικό δίκτυο δεν εκπαιδεύεται σωστά.

### Δραστηριότητα 3.1

Συγκρίνετε την μέθοδο εκπαίδευσης του νευρωνικού δικτύου του αισθητήρα με αυτήν του δικτύου του Hebb, από το κεφάλαιο 1. Δώστε τις ομοιότητες και διαφορές. Ποιά μέθοδος πιστεύετε ότι είναι πιο ικανοποιητική από τις δύο;

### Δραστηριότητα 3.2

Εκτός από τις δύο αυτές μεθόδους της Δραστηριότητας 3.1 υπάρχει και μία άλλη παρεμφερής μέθοδος που ονομάζεται μέθοδος LMS (least mean square). Η μέθοδος αυτή παρουσιάζεται λεπτομερώς στο βιβλίο [5], αλλά και σε πολλά άλλα βιβλία (λ.χ. βιβλίο [2] στην βιβλιογραφία της εισαγωγής). Βρείτε από την βιβλιογραφία ποιό είναι το κριτήριο εκπαίδευσης της μεθόδου LMS και συγκρίνετε το με αυτά της Δραστηριότητας 3.1.

### Δραστηριότητα 3.3

Έχουμε μία ομάδα από 8 σημεία (τιμές του  $x$ ) και 8 αντίστοιχες τιμές του  $y$ , οι οποίες είναι οι τιμές του στόχου. Θεωρούμε ότι τα σημεία αυτά είναι δεδομένα τα οποία πέφτουν πάνω σε μία ευθεία γραμμή. Κάνετε την γραφική παράσταση των σημείων αυτών, και χαράζετε μερικές ευθείες γραμμές. Υπολογίστε τις εξισώσεις  $y = mx + c$ , μετρώντας την κλίση και την τέμνουσα κάθε γραμμής. Για κάθε γραμμή μπορείτε να υπολογίσετε το σφάλμα με την γνωστή μέθοδο των ελαχίστων τετραγώνων. Πρώτα κάντε το αυτό χρησιμοποιώντας τους γνωστούς τύπους της μεθόδου των ελαχίστων τετραγώνων. Ακολούθως χρησιμοποιείστε ένα νευρωνικό δίκτυο όπως στο Σχήμα 3.3 και βρείτε το ίδιο αποτέλεσμα αλλά χωρίς τον τύπο τώρα. *Υπόδειξη:* Το δίκτυο που θα σχεδιάσετε θα μπορούσε να έχει τα εξής δεδομένα: Η μία είσοδος θα είναι πάντα το  $x$ , η άλλη είσοδος θα είναι πάντα η μονάδα (1), δηλ. έχουμε την περίπτωση του εσωτερικού βάρους  $b$ . Ο πρώτος νευρώνας (με είσοδο  $x$ ) συνδέεται με την έξοδο με βάρος  $m$ , ο δεύτερος συνδέεται με την έξοδο με βάρος  $c$ . Η έξοδος είναι το  $y$ . Έχουμε 8 πρότυπα, όπως είναι στον Πίνακα 3.2. Χρησιμοποιήστε ως σταθερά εκπαίδευσης  $\eta = 0,1$ . Πόσους κύκλους χρειάζεται το δίκτυο αυτό

για να εκπαιδευθεί; (Παρόμοια προεργασία για την άσκηση αυτή μπορείτε να βρείτε στο βιβλίο του Callan, [9] στον Πρόλογο).

**Πίνακας 3.2**

X	Στόχος (y)
0,30	1,60
0,35	1,40
0,40	1,40
0,50	1,60
0,60	1,70
0,80	2,00
0,95	1,70
1,10	2,10

### Άσκηση Αυτοαξιολόγησης 3.3

Ο κανόνας Δέλτα δίνει την ποσότητα  $\Delta$  ως:

(α)  $\Delta = \text{σταθερά}(\text{στόχος} - \text{έξοδος})\text{στόχος}$

(β)  $\Delta = \text{σταθερά}(\text{στόχος} - \text{έξοδος})\text{σήμα εξόδου}$

(γ)  $\Delta = \text{σταθερά}(\text{στόχος} - \text{έξοδος})\text{σήμα εισόδου}$

(α)  $\Delta = \text{στόχος} - \text{έξοδος}$

(ε) κανένα από τα παραπάνω

### 3.7 Πρότυπα Adaline και Madaline

Τα δύο αυτά πρότυπα ανήκουν στην κατηγορία των προτύπων του αισθητήρα. Το adaline βγαίνει από το adaptive linear neuron (ada-line), ενώ το madaline από το multilayer adaline (m-adaline). Παρουσιάστηκαν [6,7] από τον B. Widrow το 1959. Στο πρώτο πρότυπο, το adaline, έχουμε ένα δίκτυο με πολλές εισόδους και μία έξοδο, (Σχήμα 3.11). Κάθε είσοδος έχει το δικό της βάρος. Υπάρχει επίσης ο στόχος και έτσι η έξοδος συγκρίνεται κάθε φορά με τον στόχο, ώστε να βρεθεί η τιμή του σφάλματος. Οι νευρώνες εδώ έχουν σήμα με τιμές +1 και -1. Επίσης η έξοδος του δικτύου πρέπει να είναι +1 ή -1, κατ' αντίθεση με τον απλό αισθητήρα, όπου οι δυαδικές τιμές ήταν 0 ή 1. Παράγεται το άθροισμα της εξίσωσης 3.1. Το πρώτο σήμα που εισέρχεται στην είσοδο έχει πάντα (εξ ορισμού) την τιμή +1, δηλ.  $s_0 = 1$ . Το αντίστοιχο βάρος  $w$  μεταβάλλεται όπως και τα άλλα βάρη. Μετά την άθροιση γίνεται η σύγκριση με το  $\theta$ , και η έξοδος είναι:

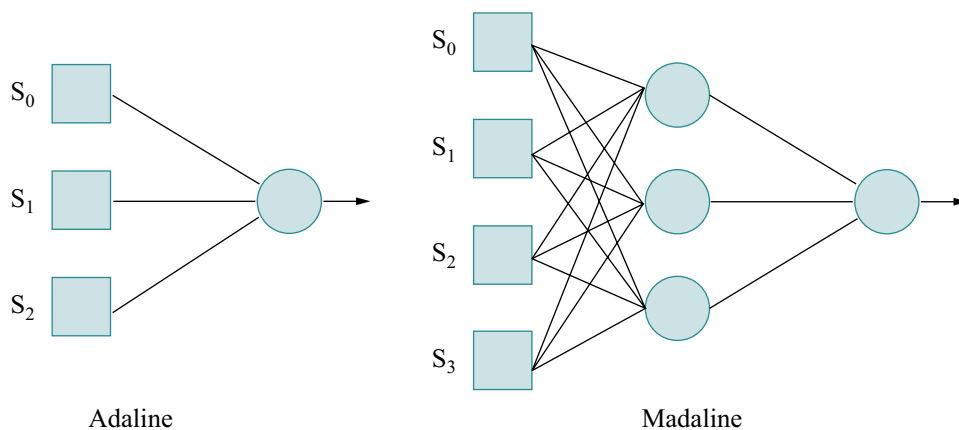
$$\text{Εάν } S \geq \theta \quad \text{τότε έξοδος} = 1 \quad (3.9)$$

$$\text{Εάν } S < \theta \quad \text{τότε έξοδος} = -1 \quad (3.10)$$

Δηλ. εδώ έχουμε  $\theta = 0$ . Κατά τα άλλα η διαδικασία είναι η ίδια όπως και με τον κανόνα Δέλτα.

**Σχήμα 3.11**

Τα δύο συγγενή  
πρότυπα Adaline  
και Madaline



Στο πρότυπο madaline έχουμε ένα επίπεδο μονάδων adaline (στο Σχήμα 3.11 οι μονάδες αυτές είναι 3) που ενώνονται με μία μονάδα εξόδου, σχηματίζοντας έτσι μία μονάδα madaline. Καθόσον έχουμε εδώ 3 επίπεδα, οι μεταβολές στα  $w$  δεν γίνονται σε όλα, αλλά μόνο σε αυτά από την είσοδο (4 νευρώνες) στο μεσαίο επίπεδο (3 νευρώνες). Χρησιμοποιούμε, δηλαδή, μία διαφορετική συνάρτηση μεταφοράς εδώ. Από το μεσαίο επίπεδο προς την έξοδο η απόφαση λαμβάνεται πλειοψηφικά (ή μπορεί να χρησιμοποιηθεί και άλλη λογική συνάρτηση). Αν περισσότεροι από τους μισούς

νευρώνες δίνουν  $+1$  (ή  $-1$ ), τότε η έξοδος madaline είναι επίσης  $+1$  (ή  $-1$ ). Μπορεί όμως να χρησιμοποιηθεί και άλλη λογική συνάρτηση για την λήψη της απόφασης. Η διαδικασία εκπαίδευσης έχει επίσης εδώ την ίδια μορφή. Η έξοδος συγκρίνεται με τον στόχο. Από την σύγκριση προκύπτει ένα σφάλμα, το οποίο χρησιμοποιείται για την μεταβολή των  $w$ . Σε μία δεδομένη στιγμή μόνο μία μονάδα adaline μεταβάλλει τα βάρη της.

### 3.8 Προβλήματα κατά την εκπαίδευση

Οι διαδικασίες εκπαίδευσης που είδαμε στα προηγούμενα τμήματα, ενώ καταρχήν φαίνονται σωστές, εντούτοις μπορεί να έχουν αρκετά προβλήματα, τα οποία ακόμη και σήμερα δεν έχουν απαντηθεί ικανοποιητικά. Είναι φανερό ότι το δίκτυο πρέπει να μαθαίνει όλο το σύνολο των προτύπων που του παρουσιάζονται. Το ερώτημα είναι πώς πρέπει να παρουσιάζονται τα πρότυπα, δηλ. σε μια δεδομένη σειρά, που επαναλαμβάνεται συνεχώς ή πρέπει να επιλέγονται με τυχαίο τρόπο; Δεν υπάρχει σωστή θεωρητική απάντηση σε αυτό, απάντηση που να καλύπτει όλους τους τύπους των δικτύων και όλες τις μεθόδους εκπαίδευσης. Πόσους κύκλους χρειάζεται ένα νευρωνικό δίκτυο για να εκπαιδευθεί; Και εδώ δεν υπάρχει σαφής απάντηση. Ποιες πρέπει να είναι οι τιμές του  $\eta$ ; Σε όλα αυτά τα ερωτήματα οι απαντήσεις είναι εμπειρικές και συνήθως δίνονται με την μέθοδο trial-and-error, δηλ. την μέθοδο κατά την οποία δοκιμάζουμε κάποιες λογικές τιμές, τις οποίες μεταβάλλουμε ανάλογα με τα αποτελέσματα που παίρνουμε, μέχρις ότου αυτά είναι ικανοποιητικά. Το βασικό μειονέκτημα όλων των προτύπων του τύπου του αισθητήρα είναι ότι δεν επιτρέπουν περισσότερα από ένα επίπεδα στα οποία να μεταβάλλονται τα βάρη. Δεν υπήρχε (τουλάχιστον τα πρώτα χρόνια) μαθηματικός τρόπος που να μεταφέρει τις αλλαγές των  $w$  επίπεδο προς επίπεδο. Αυτό έγινε αργότερα με την μέθοδο της οπισθοδιάδοσης (κεφάλαιο 4).

### Άσκηση Αυτοαξιολόγησης 3.4

Σωστό ή Λάθος. Το πρότυπο του αισθητήρα ως τεχνητό νευρωνικό δίκτυο λύνει επιτυχώς τα περισσότερα προβλήματα που χρησιμοποιούν νευρωνικά δίκτυα.

## Σύνοψη

*Στο κεφάλαιο αυτό παρουσιάσαμε την πιο απλή μορφή νευρωνικού δικτύου που υπάρχει και που δημιουργήθηκε στα πρώτα βήματα της ανάπτυξης της περιοχής αυτής. Παρόλο που τα δίκτυα αυτά έχουν πολύ περιορισμένες εφαρμογές και δυνατότητες, εν τούτοις είναι πολύ σημαντικό να τα κατανοήσουμε λεπτομερώς, γιατί οι ίδιες διεργασίες θα μας χρησιμεύσουν σε πιο περίπλοκα δίκτυα σε μελλοντικά κεφάλαια. Παρουσιάσαμε την δομή τέτοιων δικτύων αρχίζοντας από τον στοιχειώδη αισθητήρα, που έχει ένα (1) μόνο νευρώνα, μέχρι τις δομές δύο ή περισσότερων επιπέδων με πολλούς νευρώνες το κάθε ένα. Παρουσιάσαμε τον κανόνα Δέλτα, που είναι μια μαθηματική διεργασία εκπαίδευσης του δικτύου. Εξηγήσαμε τι ακριβώς συμβαίνει σε ένα δίκτυο όταν εκπαιδεύεται.*

## Βιβλιογραφικές παραπομπές

- [1] F. Rosenblatt, Principles of Neurodynamics, Spartan Books, Washington DC (1962).
- [2] F. Rosenblatt, «The perceptron: A probabilistic model for information storage and organization in the brain», Psychological Review, **65**,386(1958).
- [3] H. D. Block, «The perceptron: A model for brain functioning», Reviews of Modern Physics, **34**,123(1962).
- [4] M. Minsky and S. Pappert, Perceptrons, MIT Press (1969)
- [5] S. Haykin, Neural Networks: A Comprehensive Foundation, Second Edition, Prentice Hall, Upper Saddle Point (1999).
- [6] B. Widrow and M. A. Lehr, «30 years of adaptive Neural Networks: Perceptron, Madaline, and Backpropagation.», Proceedings of the IEEE, 78(9)1415(1990).
- [7] B. Widrow and S. D. Stearns, Adaptive Signal Processing, Prentice Hall, Englewood Cliffs (NJ), 1985.

## Μέθοδος οπισθοδιάδοσης του λάθους

### Σκοπός

Στο κεφάλαιο αυτό παρουσιάζεται η πιο διαδεδομένη μέθοδος για την εκπαίδευση ενός νευρωνικού δικτύου, η μέθοδος της οπισθοδιάδοσης του λάθους. Παρουσιάζεται η μαθηματική υποδομή και όλες οι λεπτομέρειες της τεχνικής που χρησιμοποιείται για την εκπαίδευση του δικτύου, που είναι η τεχνική της πλέον απότομης καθόδου (steepest descent). Δίδεται η κατάλληλη δομή των δικτύων αυτών και εξηγείται για ποιους λόγους είναι απαραίτητη μια δομή με κρυμμένα επίπεδα. Χρησιμοποιούμε το παράδειγμα του προβλήματος  $X-OR$ , για το οποίο παρουσιάζουμε την λύση με μορφή προσομοίωσης.

### Προσδοκώμενα Αποτελέσματα

Όταν ολοκληρώσετε τη μελέτη του κεφαλαίου αυτού, θα είστε σε θέση να:

- περιγράψετε την τεχνική της εκπαίδευσης δικτύων με την μέθοδο της οπισθοδιάδοσης του λάθους σε 6 βήματα
- διακρίνετε τουλάχιστον δύο κατηγορίες προβλημάτων και να προτείνετε την κατάλληλη δομή δικτύου για καθεμία από αυτές
- εκπαιδεύσετε το δίκτυο που προτείνετε στα πρότυπα που δίνει το πρόβλημα
- σχεδιάσετε και να κατασκευάσετε έναν αλγόριθμο προσομοίωσης για το πρόβλημα  $X-OR$

### Λέξεις κλειδιά

- Τεχνική της πλέον απότομης καθόδου
- οπισθοδιάδοση του λάθους
- κρυμμένο επίπεδο
- στιγμοειδής συνάρτηση
- βάρος
- εσωτερικό βάρος
- διόρθωση βαρών

- κύκλος διόρθωσης
- σφάλμα
- στόχος
- γενικευμένος κανόνας Δέλτα
- πρόβλημα  $X-OR$
- τοπικό ελάχιστο
- ολικό ελάχιστο.

### Εισαγωγικές Παρατηρήσεις

Για πολλά χρόνια δεν υπήρχε τρόπος εκπαίδευσης δικτύων με πολλά επίπεδα και οι γνώσεις μας περιοριζόνταν σε δίκτυα με ένα ή δύο επίπεδα μόνον. Φυσικά, τα προβλήματα που μπορούσαν να λύσουν τα απλά δίκτυα ήταν λίγα και οι περιορισμοί που υπεισέρχονταν τα καθιστούσαν όχι και τόσο χρήσιμα. Έτσι, έγινε γρήγορα αντιληπτό ότι ήταν απαραίτητο να βρούμε ένα τρόπο να εκπαιδεύσουμε νευρωνικά δίκτυα που μπορούν να αναπαραστήσουν πιο περίπλοκες διεργασίες, όπως λ.χ. το πρόβλημα  $X-OR$ . Είδαμε στα προηγούμενα κεφάλαια ότι το πρόβλημα αυτό δεν λύνεται με τις γνωστές ως τώρα τεχνικές. Το κενό αυτό ήρθε να καλύψει η μέθοδος εκπαίδευσης με την οπισθοδιάδοση του λάθους (*error backpropagation* ή σε συντομία *backprop*), που αναπτύχθηκε και έγινε ευρύτατα γνωστή την δεκαετία των ογδόντα, και σύντομα έγινε η υπ' αριθμόν 1 διαδικασία στα τεχνητά νευρωνικά δίκτυα. Με μεγάλη διαφορά είναι η πιο συχνά συναντώμενη τεχνική σχεδόν σε όλες τις εφαρμογές που χρησιμοποιούνται σήμερα. Τα δύο βασικά χαρακτηριστικά στην μέθοδο αυτή είναι τα εξής: Πρώτον, απαραίτητη προϋπόθεση είναι εκτός από τα επίπεδα εισόδου/εξόδου η ύπαρξη περισσότερων επιπέδων, των λεγόμενων κρυμμένων επιπέδων, που βρίσκονται μετά την είσοδο και πριν την έξοδο. Δεύτερον, η συνάρτηση μεταφοράς πρέπει να είναι οπωσδήποτε μία σιγμοειδής μη-γραμμική συνάρτηση.



## 4.1 Οι πρώτες ιδέες

Η μέθοδος οπισθοδιάδοσης του λάθους (error backpropagation) είναι η πιο δημοφιλή μέθοδος σήμερα για την εκπαίδευση ενός δικτύου που αποτελείται από πολλά επίπεδα και έχει χρησιμοποιηθεί στις πιο πολλές εφαρμογές. Ιστορικά, πρώτα αναπτύχθηκαν δίκτυα ενός και δύο επιπέδων, όπως ο στοιχειώδης αισθητήρας που είδαμε στο προηγούμενο κεφάλαιο. Τα δίκτυα όμως αυτά γρήγορα φάνηκε ότι έχουν μεγάλους περιορισμούς ως προς τις ικανότητες που τους έχουν και έτσι σύντομα εγκαταλείφθηκαν. Έτσι φυσιολογικά ακολούθησαν τα δίκτυα πολλών επιπέδων που αναπτύχθηκαν αργότερα και για τα οποία αρχικά δεν υπήρχαν θεωρητικοί τρόποι για την εκπαίδευσή τους, μέχρι που εμφανίστηκε η μέθοδος οπισθοδιάδοσης. Η μέθοδος αυτή αναπτύχθηκε ανεξάρτητα σε διάφορες παραλλαγές από τους Bryson και Ho [1], Werbos [2, 3], Parker [4], αλλά διαφημίστηκε πολύ και προωθήθηκε από το έργο «Parallel Distributed Processing: Explorations in the Microstructure of Cognition» των Rumelhart και McClelland [5], το οποίο άνοιξε πολλές εφαρμογές και νέα πεδία, ανακινώντας μεγάλο ενδιαφέρον σε όλη την περιοχή των νευρωνικών δικτύων. Ως τεχνική βασίζεται σε καθαρά μαθηματική θεώρηση με αυστηρά τεκμηριωμένες αποδείξεις. Το νευρωνικό δίκτυο στο οποίο εφαρμόζεται είναι αρκετά πιο περίπλοκο από τον αισθητήρα. Είναι ένα δίκτυο πολλαπλών επιπέδων και κάθε επίπεδο έχει (ή μπορεί να έχει) πολλούς νευρώνες. Οι νευρώνες μέσα στο ίδιο επίπεδο δεν συνδέονται μεταξύ τους, αλλά οι νευρώνες που ανήκουν σε διαφορετικά επίπεδα συνδέονται ως συνήθως με τις γνωστές συνάψεις. Υπάρχουν λοιπόν πολλές σειρές με τα βάρη  $w$  μεταξύ των επιπέδων αυτών και όχι μία μόνο σειρά που είδαμε στο κεφάλαιο 3. Η καινοτομία που εισάγεται στα δίκτυα αυτά είναι ότι μπορούμε να επιφέρουμε τις κατάλληλες μεταβολές στα βάρη στα ενδιάμεσα επίπεδα, εκεί όπου δεν υπάρχει στόχος και άρα δεν μπορεί να χρησιμοποιηθεί μια απλή τεχνική, όπως είναι λ.χ. ο κανόνας Δέλτα.

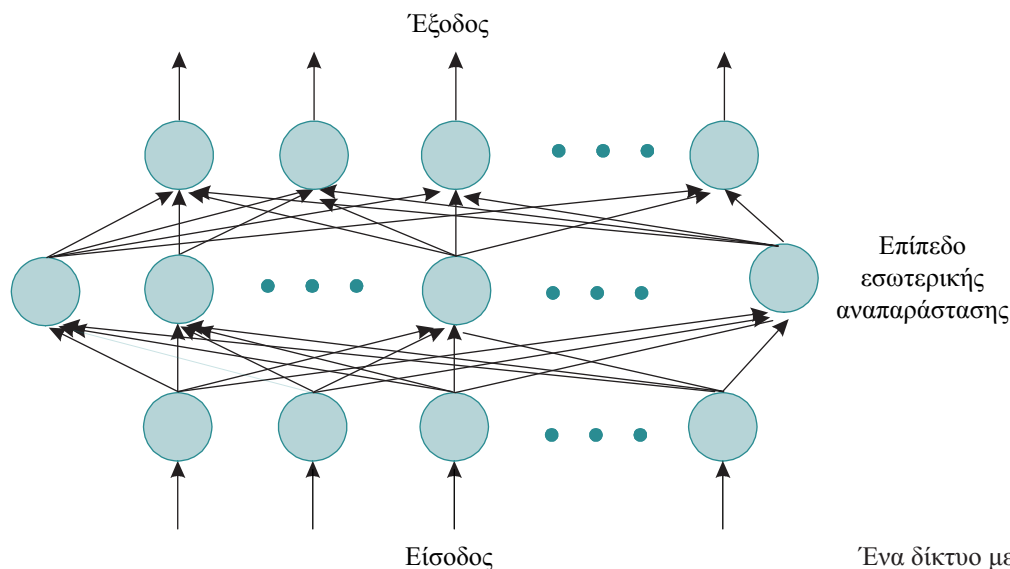
### Άσκηση Αυτοαξιολόγησης 4.1

Γιατί δεν εφαρμόζεται ο κανόνας Δέλτα εκεί όπου το πρόβλημα δεν έχει στόχο;

Η κεντρική ιδέα της δομής και λειτουργίας τέτοιων δικτύων είναι σχετικά απλή: ένα δίκτυο ξεκινά την διαδικασία μάθησης από τυχαίες τιμές των βαρών του. Εάν δώσει λάθος απάντηση (που είναι και το πιο πιθανό), τότε τα βάρη διορθώνονται έτσι ώστε το λάθος να γίνει μικρότερο. Η ίδια διαδικασία επαναλαμβάνεται πολλές φορές έτσι ώστε σταδιακά το λάθος ελαττώνεται μέχρις ότου γίνει πολύ μικρό και ανεκτό. Στο

σημείο αυτό λέμε ότι το δίκτυο έχει μάθει τα παραδείγματα που του διδάξαμε με την ακρίβεια που θέλαμε να μάθει.

Έχουμε δει λεπτομερώς την δομή του μοντέλου του αισθητήρα, όπου τα εισερχόμενα σήματα στο δίκτυο φθάνουν στο επίπεδο εισόδου, επεξεργάζονται στους νευρώνες και από εκεί οδηγούνται κατευθείαν προς στο επίπεδο εξόδου. Τέτοια δίκτυα δεν έχουν εσωτερική αναπαράσταση. Αυτό σημαίνει ότι οποιαδήποτε κωδικοποίηση δίνεται στο σήμα εισόδου, ότι είναι αρκετή, καθόσον τα πρότυπα που εισάγονται στην είσοδο και αυτά που παράγονται στην έξοδο είναι του ίδιου τύπου. Αυτό επιτρέπει στα δίκτυα αυτά να κάνουν λογικές γενικεύσεις και να βρίσκουν πρότυπα τα οποία ποτέ δεν έχουν δει. Ο περιορισμός όμως του ότι οι εισοδοί και εξοδοί πρέπει να είναι του ίδιου τύπου δεν τους επιτρέπει να λύσουν πιο γενικά ή πιο περίπλοκα προβλήματα. Στο γνωστό πρόβλημα του X-OR βλέπουμε ότι δύο πρότυπα που είναι τελείως διαφορετικά πρέπει να δώσουν ίδια απάντηση. Η λύση στην δυσκολία αυτή βρίσκεται με το να δώσουμε στο δίκτυο μια διαφορετική δομή και να αποκτήσει έτσι μία καινούρια ικανότητα. Προσθέτουμε τώρα και ένα τρίτο επίπεδο, μεταξύ του επιπέδου εισόδου και εξόδου, που ονομάζεται κρυμμένο επίπεδο [6] και το οποίο τώρα μπορεί να δημιουργήσει την εσωτερική αναπαράσταση των σημάτων εισόδου. Ο λόγος που ονομάζουμε το επίπεδο αυτό κρυμμένο επίπεδο είναι ότι το επίπεδο αυτό δεν «βλέπει» κατευθείαν ούτε την είσοδο ούτε την έξοδο του δικτύου αλλά μόνον το εσωτερικό του.

**Σχήμα 4.1**

Ένα δίκτυο με πολλαπλά επίπεδα

Μετά τις πολλές εργασίες που έγιναν με το μοντέλο του αισθητήρα φάνηκε ότι όταν υπάρχει ένα κρυμμένο επίπεδο τότε δημιουργείται πάντοτε ένας τρόπος αναπαράστασης στο κρυμμένο αυτό επίπεδο, το οποίο τώρα μπορεί να ξεπεράσει τον περιορισμό που υπήρχε προηγουμένως περί της ομοιότητας εισόδου–εξόδου. Αρκεί να έχουμε αρκετές μονάδες (νευρώνες) στο κρυμμένο επίπεδο και να βρούμε τα σωστά βάρη  $w$  με μια κατάλληλη διαδικασία. Ένα τέτοιο δίκτυο πολλαπλών επιπέδων φαίνεται στο Σχήμα 4.1. Ως συντομογραφία ενός πολυεπιπέδου νευρωνικού δικτύου συχνά χρησιμοποιείται ο εξής:  $p-m_1-m_2-\dots-m_n-n$ , όπου  $p$  είναι ο αριθμός των εισόδων,  $n$  είναι ο αριθμός των εξόδων,  $m$  ο αριθμός των κρυμμένων επιπέδων με  $m_1$  κόμβους το πρώτο,  $m_2$  κόμβους το δεύτερο,... και  $m_n$  το τελευταίο.

Πρώτα, υπάρχει ένα επίπεδο εισόδου το οποίο αποτελείται από μία ομάδα νευρώνων οι οποίοι δεν κάνουν ουσιαστικά τίποτα άλλο παρά να δέχονται το σήμα εισόδου. Κατόπιν υπάρχει ένας αριθμός εσωτερικών επιπέδων, καθένα από τα οποία έχει έναν αριθμό νευρώνων, και τα οποία δέχονται το σήμα από το επίπεδο εισόδου, το επεξεργάζονται και κατόπιν το προωθούν προς την έξοδο. Στο Σχήμα 4.1 υπάρχει ένα μόνο κρυμμένο επίπεδο, αλλά θα μπορούσε να ήταν δύο, τρία ή οποιοσδήποτε άλλος αριθμός επιπέδων. Τέλος, υπάρχει ένα επίπεδο εξόδου που έχει επίσης έναν αριθμό νευρώνων, οι οποίοι δέχονται σήμα από τα εσωτερικά επίπεδα και το προωθούν προς την έξοδο του δικτύου. Γενικά, δεν υπάρχει κανόνας ως προς τον αριθμό τόσο των εσωτερικών επιπέδων όσο και ως προς τον αριθμό των νευρώνων που περιλαμβάνει κάθε επίπεδο (εισόδου, εξόδου ή εσωτερικό). Η απάντηση σ' αυτό είναι διαφορετική σε κάθε πρόβλημα. Για τον αριθμό νευρώνων στην είσοδο και έξοδο το

πρόβλημα είναι κάπως ευκολότερο, γιατί ο αριθμός αυτός θα παρέχεται άμεσα από τα δεδομένα του προβλήματος. Εάν, λ.χ. θέλουμε να αναπαραστήσουμε μία συνάρτηση που ορίζεται από 256 σημεία, τότε η είσοδος θα πρέπει να έχει 256 μονάδες. Αλλά για τον αριθμό μονάδων στο κρυμμένο επίπεδο δεν υπάρχει ούτε τέτοιου είδους υπόδειξη. Στην βιβλιογραφία αναφέρεται ότι τέτοιες απαντήσεις βγαίνουν ακόμη και με «μαύρη τέχνη». Πολλές φορές αναγκάζομαστε και καταφεύγουμε στην μέθοδο των δοκιμών (trial and error) και με τον τρόπο αυτό βρίσκουμε μία απάντηση που σίγουρα δουλεύει, αλλά είναι επίπονη και χρονοβόρα. Ανάλογα με το πρόβλημα υπάρχουν πολλοί εμπειρικοί κανόνες που βάζουν κάποια όρια στην αρχιτεκτονική του δικτύου που θα χρησιμοποιηθεί σε μία πρακτική εφαρμογή. Έχει αποδειχθεί λ.χ. ότι ένα δίκτυο δεν μπορεί να μάθει περισσότερα παραδείγματα από το διπλάσιο του αριθμού των βαρών του.

Όπως φαίνεται και στο Σχήμα 4.1 οι νευρώνες των διαφορετικών επιπέδων είναι συνδεδεμένοι μεταξύ τους με μία γραμμή. Στο σημείο αυτό δεν υπάρχει ένας γενικός κανόνας, δηλ. πόσοι και ποιοι νευρώνες είναι συνδεδεμένοι με ποιους. Σε μία περίπτωση θα μπορούσε κάθε νευρώνας να είναι συνδεδεμένος με όλους τους άλλους νευρώνες, όλων των επιπέδων (μέγιστος αριθμός συνδέσεων). Σε άλλη περίπτωση θα μπορούσε κάθε νευρώνας να συνδέεται με έναν μόνο άλλο νευρώνα (ο ελάχιστος αριθμός των συνδέσεων που μπορεί να έχει). Αρκετά συνηθισμένες είναι οι ενδιάμεσες περιπτώσεις, όπου συνήθως υπάρχουν μερικές συνδέσεις μεταξύ των νευρώνων. Όπως είναι προφανές ο αριθμός των συνδέσεων, ιδίως για την πλήρη συνδεσμολογία είναι πολύ μεγάλος. Αν έχουμε  $N$  νευρώνες, τότε ο αριθμός των συνδέσεων σε πλήρη συνδεσμολογία είναι  $N(N-1)/2$ .

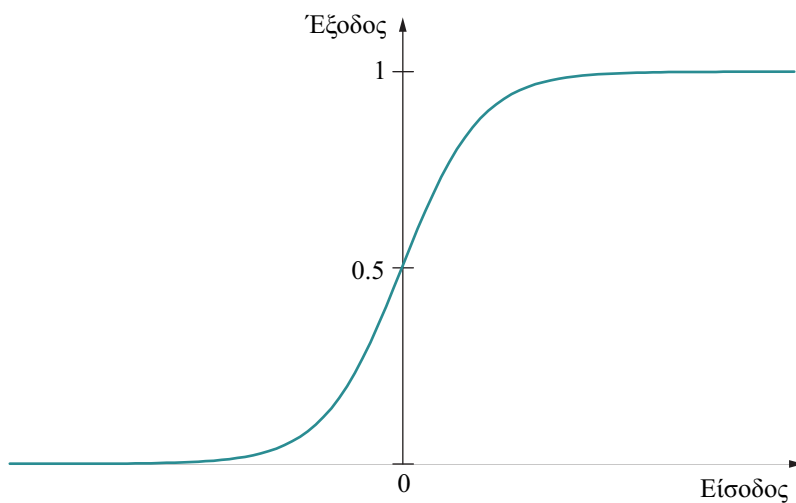
## Άσκηση Αυτοαξιολόγησης 4.2

Βρείτε γιατί σε δίκτυο  $N$  νευρώνων με πλήρη συνδεσμολογία ο αριθμός των συνδέσεων δίδεται από την σχέση  $N(N-1)/2$

Η διαδικασία εκπαίδευσης έχει την ίδια φιλοσοφία με αυτή του αισθητήρα, αλλά έχει μερικές ουσιώδεις διαφορές. Το σήμα  $s$  έρχεται σε κάθε νευρώνα του επιπέδου εισόδου (το πρώτο επίπεδο). Πολλαπλασιάζεται επί το αντίστοιχο βάρος  $w$  κάθε σύναψης (και στα τεχνητά δίκτυα μπορούμε ελεύθερα να χρησιμοποιήσουμε τον όρο σύναψη από τα βιολογικά δίκτυα για να υποδηλώσουμε την σύνδεση μεταξύ δύο νευρώνων). Σε κάθε νευρώνα αθροίζονται τα γινόμενα  $s_i w_i$ , με  $i = 1, \dots, n$ , όπου  $n$  το πλήθος των συνδέσεων, τα οποία έρχονται ως είσοδος, και υπολογίζεται το  $S$ , όπως

και στο μοντέλο του αισθητήρα. Εδώ όμως υπάρχει μία ουσιαστική διαφορά. Ενώ στον αισθητήρα το άθροισμα συγκρίνεται με το  $\theta$  και συνήθως έχουμε συνάρτηση μεταφοράς με δυαδική μορφή, εδώ είναι απαραίτητο να χρησιμοποιήσουμε μία συνάρτηση με σιγμοειδή μορφή. Έστω ότι η τιμή της εξόδου θα είναι  $o$  (προσοχή εδώ, το «ο» είναι το γράμμα όμικρον, και όχι το μηδέν, 0). Μία συχνά χρησιμοποιούμενη τέτοια συνάρτηση είναι η

$$o = \frac{1}{1 + e^{-s}} \quad (4.1)$$



**Σχήμα 4.2**

Η σιγμοειδής καμπύλη

Η συνάρτηση αυτή φαίνεται στο Σχήμα 4.2 και έχει τα εξής χαρακτηριστικά. Η τιμή του  $o$  περιορίζεται πάντοτε στο διάστημα  $0 < o < 1$ , για οποιαδήποτε τιμή της εισόδου  $S$ . Αυτό είναι σημαντικό, διότι έτσι είμαστε βέβαιοι ότι δεν θα υπάρχουν περιπτώσεις που η έξοδος παίρνει μεγάλες τιμές ή απειρίζεται. Η καμπύλη αυτή ονομάζεται σιγμοειδής, λόγω του σχήματος που έχει και μοιάζει με ένα τελικό σίγμα. Είναι ιδανική συνάρτηση, γιατί συμπεριφέρεται καλά για όλα τα μεγέθη τιμών. Για μικρές τιμές του  $S$  η κλίση είναι μεγάλη και έτσι η έξοδος δεν είναι σχεδόν 0. Ανάλογα, για μεγάλες τιμές του  $S$  η κλίση είναι κανονική, ούτως ώστε να μην μπορεί το δίκτυο να δώσει πολύ μεγάλες τιμές ή άπειρο στην έξοδο του. Μία άλλη ονομασία της συνάρτησης  $o$  είναι «συμπιέζουσα συνάρτηση», διότι συμπιέζει οποιαδήποτε τιμή του  $S$ , όσο μεγάλη και αν είναι, στο διάστημα μεταξύ 0 και 1. Παρατηρούμε επίσης ότι η συνάρτηση αυτή είναι μη γραμμική, μία απαραίτητη προϋπόθεση για να μπορεί το δίκτυο να δημιουργήσει αναπαράσταση των σημάτων.

### Άσκηση Αυτοαξιολόγησης 4.3

Για την Άσκηση Αυτοαξιολόγησης 3.2, με τα ίδια δεδομένα, υπολογίστε την έξοδο του δικτύου χρησιμοποιώντας τώρα για συνάρτηση ενεργοποίησης την σιγμοειδή συνάρτηση, εξίσωση 4.1, και συγκρίνετε τις δύο περιπτώσεις.

Κάτι άλλο για την σιγμοειδή συνάρτηση που επίσης είναι απαραίτητο στην διαδικασία εκπαίδευσης είναι ότι πρέπει και η παράγωγός της να συμπεριφέρεται επίσης καλά, δηλ. να έχει τις ίδιες ιδιότητες που είδαμε παραπάνω. Εύκολα δείχνουμε ότι η παράγωγος αυτή είναι:

$$\frac{\partial o}{\partial S} = o(1 - o) \quad (4.2)$$

Ο υπολογισμός της παραγώγου της σιγμοειδούς συνάρτησης απευθείας από την ίδια την συνάρτηση, έχει σημαντικά υπολογιστικά πλεονεκτήματα και διευκολύνει την υλοποίηση σε hardware.

### Άσκηση Αυτοαξιολόγησης 4.4

Αποδείξτε την εξίσωση (4.2) χρησιμοποιώντας την εξίσωση (4.1)

### Δραστηριότητα 4.1

Εκτός από την συχνά χρησιμοποιούμενη συνάρτηση της εξίσωσης 4.1, όπως αναφέραμε παραπάνω υπάρχουν και μερικές άλλες παρόμοιες συναρτήσεις, όπως είναι οι εξής:

$$\varphi(v) = \frac{1 - \exp(-av)}{1 + \exp(-av)} = \tanh\left(\frac{av}{2}\right)$$

$$\varphi(v) = \frac{v}{\sqrt{1 + v^2}}$$

Βρείτε τα όρια των δύο αυτών συναρτήσεων. Δείξτε ότι οι παράγωγοί τους είναι οι εξής:

$$\frac{d\varphi}{dv} = \frac{a}{2} [1 - \varphi^2(v)]$$

$$\frac{d\phi}{dv} = \frac{\phi^3(v)}{v^3}$$

Ποια τιμή παίρνουν οι παράγωγοι αυτοί στην αρχή των αξόνων;

Η συνολική διαδικασία εκπαίδευσης συνοψίζεται στα εξής 6 βήματα:

- Παίρνουμε ένα πρότυπο από τα πολλά που έχει το πρόβλημα μας. Το εισάγουμε στο επίπεδο εισόδου.
- Υπολογίζουμε την έξοδο χρησιμοποιώντας την σιγμοειδή συνάρτηση.
- Προωθούμε την έξοδο του πρώτου επιπέδου στο επόμενο επίπεδο (το κρυμμένο) και ακολούθως με τον ίδιο τρόπο σε όλα τα επίπεδα μέχρι το τελικό επίπεδο εξόδου.
- Στην έξοδο υπολογίζουμε το σφάλμα.
- Ανάλογα με το σφάλμα που προκύπτει μεταβάλλουμε τα βάρη, ένα-ένα, και επίπεδο-προς-επίπεδο, επιστρέφοντας από την έξοδο μέχρι την είσοδο
- Προχωρούμε στο επόμενο πρότυπο και ακολούθουμε την ίδια διαδικασία για όλα τα πρότυπα.

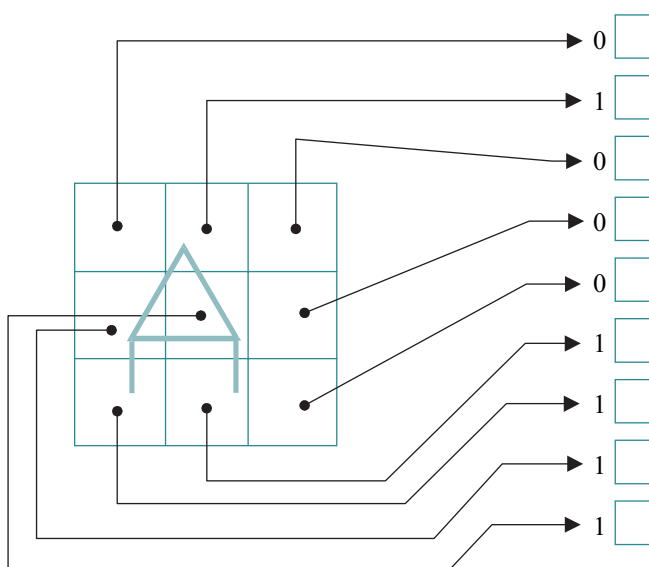
Τα έξι αυτά βήματα αποτελούν ένα κύκλο, δηλ. ένα πέρασμα από την είσοδο μέχρι την έξοδο, μέσω των κρυμμένων επιπέδων, και από την έξοδο πίσω στην είσοδο. Μετά το τέλος ενός κύκλου διόρθωσης των  $w$  επαναλαμβάνουμε την διαδικασία για πολλούς κύκλους, όσους χρειάζεται, έως ότου διαδοχικά το σφάλμα φθάσει να είναι αρκετά μικρό. Η ανοχή για το σφάλμα δίδεται εκ των προτέρων και τυπικές τιμές είναι μερικές % μονάδες, όπως λ.χ. 2 ή 5 %. Αυτό που ακόμη δεν αναφέραμε είναι το πως διορθώνουμε τα βάρη, αλλά η διαδικασία αυτή θα παρουσιασθεί λεπτομερώς στις επόμενες ενότητες του κεφαλαίου.

Ένα παράδειγμα ζεύγους προτύπου-στόχου δίδεται στο Σχήμα 4.3, όπου το γράμμα Α έχει σχεδιασθεί σε ένα πλέγμα. Αν οποιαδήποτε γραμμή ή τμήμα του γράμματος περνάει μέσα σε ένα τετραγωνάκι, τότε η είσοδος στον αντίστοιχο νευρώνα είναι 1. Διαφορετικά η είσοδος είναι 0. Ως έξοδος μπορεί να είναι ένας αριθμός που παριστάνει το Α, ή ένα άλλο σύνολο από 0 και 1. Για ολόκληρο το αλφάβητο θα χρειάζομασταν 24 ζεύγη εκπαίδευσης του δικτύου, ένα ζεύγος για κάθε γράμμα.

Η μέθοδος εκπαίδευσης της οπισθοδιάδοσης του σφάλματος χρησιμοποιεί τις ίδιες γενικές αρχές όπως και ο κανόνας Δέλτα. Το σύστημα πρώτα παίρνει τις εισόδους

του πρώτου προτύπου και με την διαδικασία που περιγράφηκε προηγουμένως παράγει την έξοδο. Την τιμή εξόδου την συγκρίνει με την τιμή του στόχου. Εάν δεν υπάρχει διαφορά μεταξύ των δύο, δεν συμβαίνει τίποτα και προχωράμε στο επόμενο πρότυπο. Εάν υπάρχει διαφορά (που είναι το πιο συνηθισμένο), τότε αλλάζουμε τις τιμές των  $w$  με τέτοιο τρόπο ώστε η διαφορά αυτή να ελαττωθεί.

Η καλύτερη πηγή για την μαθηματική ανάπτυξη των εξισώσεων που οδηγούν στην εκπαίδευση του δικτύου είναι στο Κεφάλαιο 8 του βιβλίου των Rumelhart–McClelland [7]. Άλλες πηγές είναι [8,9].



**Σχήμα 4.3**

Αναγνώριση προτύπου

### Άσκηση Αυτοαξιολόγησης 4.5

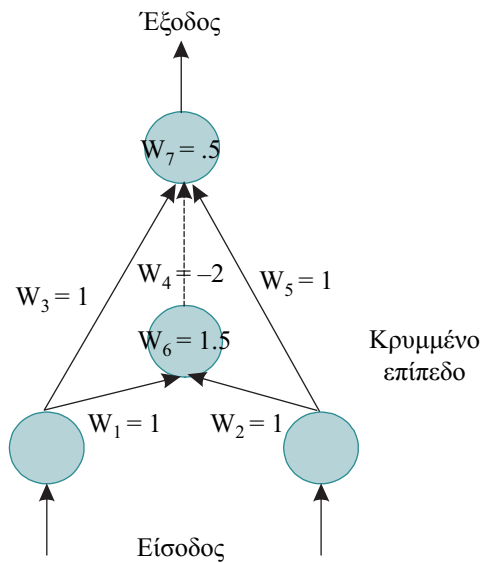
Σωστό ή Λάθος. Η σιγμοειδής συνάρτηση χρησιμοποιείται συχνά στην εκπαίδευση των νευρωνικών δικτύων γιατί έχει την ιδιότητα ότι δεν απειρίζεται ποτέ.

## 4.2 Η μέθοδος εκπαίδευσης για γραμμικούς νευρώνες

Με την μέθοδο αυτή μπορούμε να πετύχουμε αυτό που δεν κατορθώνουμε να κάνουμε με ένα απλό αισθητήρα, δηλ. να λύσουμε περίπλοκα προβλήματα όπως είναι τα γραμμικώς μη-διαχωρίσιμα. Όπως αναφέραμε παραπάνω το κλειδί στο σημείο αυτό είναι ότι πρέπει να υπάρχει ένα (τουλάχιστον) κρυμμένο επίπεδο. Με τον τρόπο αυτό δημιουργείται μία εσωτερική αναπαράσταση των προτύπων που παρουσιάζονται στην είσοδο προς τους νευρώνες του κρυμμένου επιπέδου και με τους οποίους η



ομοιότητα των προτύπων στους νευρώνες του κρυμμένου επιπέδου θα μπορεί να υποστηρίξει την απαιτούμενη αναπαράσταση (ή απεικόνιση) από την είσοδο στην έξοδο. Εάν λοιπόν έχουμε τις σωστές συνδέσεις και αρκετά μεγάλο αριθμό κρυμμένων μονάδων, θα μπορούμε πάντοτε να βρίσκουμε την αναπαράσταση αυτή.



**Σχήμα 4.4**

Ένα δίκτυο, στο οποίο εφαρμόζεται η εκπαίδευση με τη μέθοδο της οπισθοδιάδοσης

Στο Σχήμα 4.4 παρουσιάζουμε ένα τέτοιο δίκτυο, με ένα κρυμμένο επίπεδο που περιέχει ένα νευρώνα μόνον. Οι αριθμοί στις συνάψεις είναι οι τιμές των βαρών. Οι αριθμοί που είναι μέσα στους κύκλους είναι οι τιμές του εσωτερικού βάρους (κατωφλίου) του αντίστοιχου νευρώνα. Δηλαδή, εσωτερικά βάρη έχουμε μόνο τα  $w_6$  και  $w_7$ , τα οποία για να παράγουν το ζητούμενο γινόμενο ( $s \cdot w$ ) πολλαπλασιάζονται επί 1. Η τιμή  $w_4 = -2$  από τον κρυμμένο νευρώνα στον νευρώνα εξόδου καθιστά τον νευρώνα εξόδου μη-ενεργό όταν και οι δύο εισόδοι ταυτόχρονα είναι ενεργοί. Στον νευρώνα του κρυμμένου επιπέδου έχουμε  $\theta = 1,5$  διότι έτσι ο νευρώνας αυτός θα πυροδοτεί μόνον όταν και οι δύο νευρώνες του πρώτου επιπέδου είναι ενεργοί. Η τιμή  $\theta = 0,5$  στον νευρώνα εξόδου καθιστά τον νευρώνα αυτόν ενεργό μόνον όταν λαμβάνει θετικό σήμα μεγαλύτερο από 0,5. Από την πλευρά του νευρώνα εξόδου ο νευρώνας του κρυμμένου επιπέδου φαίνεται ως μια ακόμα μονάδα εισόδου. Τον βλέπει δηλαδή σαν να υπήρχαν τρεις τιμές εισόδου. Σε ένα τέτοιο δίκτυο θα αναπτύξουμε την μέθοδο της οπισθοδιάδοσης αμέσως παρακάτω.

**Πίνακας 4.1***Σύμβολα στις εξισώσεις της μεθόδου οπισθοδιάδοσης*

$w_{ij}$	Το βάρος που συνδέει τους νευρώνες $i$ και $j$
$\Delta_p w_{ji}$	η αλλαγή στο βάρος $w$ το οποίο συνδέει τους νευρώνες $i$ και $j$ , μετά από παρουσίαση του προτύπου $p$
$E_p$	$E_p$ είναι το σφάλμα (διαφορά εισόδου–εξόδου) στο πρότυπο $p$
$t_{pj}$	ο στόχος του νευρώνα $j$ για το πρότυπο $p$
$o_{pj}$	η έξοδος του νευρώνα $j$ για το πρότυπο $p$
$x_{pi}$	το σήμα εισόδου στον νευρώνα $i$ για το πρότυπο $p$
$\delta_{pj}$	η διαφορά $(t_{pj} - o_{pj})$

Η μέθοδος αυτή βασίζεται στην μαθηματική μέθοδο της ελαχιστοποίησης του σφάλματος με την τεχνική της πλέον απότομης καθόδου (steepest descent technique) στην επιφάνεια του σφάλματος, ένα πρόβλημα που ανήκει στη γενικότερη κατηγορία προβλημάτων επικλινούς καθόδου (gradient descent), που έχουν αναπτυχθεί για προβλήματα Μαθηματικής Φυσικής. Αυτό που επιτελεί είναι να ελαχιστοποιεί το τετράγωνο της διαφοράς μεταξύ του σήματος που λαμβάνεται στην έξοδο και της επιθυμητής τιμής (στόχος), για όλους τους νευρώνες εξόδου και για όλα τα πρότυπα. Αυτό σημαίνει ότι η παράγωγος του σφάλματος ως προς κάθε βάρος  $w$  είναι ανάλογος προς την μεταβολή της τιμής του βάρους, όπως δίνεται από τον κανόνα Δέλτα, με αρνητική σταθερά αναλογίας. Αυτό είναι ανάλογο με την διαδικασία της πιο απότομης καθόδου (steepest descent) πάνω στην επιφάνεια που βρίσκεται μέσα στον χώρο των βαρών και στον οποίο χώρο το ύψος είναι ίσο με την τιμή του σφάλματος. Τα παραπάνω ισχύουν για γραμμικές μονάδες νευρώνων. Έτσι έχουμε:

$$E_p = \frac{1}{2} \sum_j (t_{pj} - o_{pj})^2. \quad (4.3)$$

όπου  $E_p$  είναι το σφάλμα (διαφορά εισόδου–εξόδου) στο πρότυπο  $p$ ,  $t_{pj}$  και  $o_{pj}$  είναι ο στόχος και η έξοδος του νευρώνα  $j$  για το πρότυπο  $p$ . Το συνολικό σφάλμα  $E$  είναι το άθροισμα των σφαλμάτων όλων των προτύπων:

$$E = \sum_p E_p \quad (4.4)$$

Παρατηρούμε ότι παίρνουμε το τετράγωνο της διαφοράς και όχι την διαφορά, και επίσης το  $\frac{1}{2}$  της ποσότητας αυτής. Ο λόγος είναι ότι χρειαζόμαστε την απόλυτη τιμή του σφάλματος και όχι αν το σφάλμα είναι θετικό ή αρνητικό. Ο παράγων  $\frac{1}{2}$  είναι

μία αυθαίρετη σταθερά που δεν επηρεάζει την ανάπτυξη. Για γραμμικές μονάδες εφαρμόζουμε τον κανόνα Δέλτα και ουσιαστικά έχουμε μία επικλινή κάθοδο (gradient descent) στο  $E$ . Θα δείξουμε ότι:

$$-\frac{\partial E_p}{\partial w_{ji}} = \delta_{pj} x_{pi} \quad (4.5)$$

που είναι ποσότητα ανάλογη του  $\Delta_p w_{ji}$  ( $\Delta_p w_{ji}$  είναι η αλλαγή που θα γίνει στο βάρος  $w$  το οποίο συνδέει τους νευρώνες  $i$  και  $j$ , μετά από παρουσίαση του προτύπου  $p$ ). Όταν δεν υπάρχουν κρυμμένες μονάδες, τότε η παράγωγος υπολογίζεται αμέσως. Χρησιμοποιούμε τον κανόνα αλυσίδας και γράφουμε την παράγωγο ως γινόμενο δύο άλλων παραγώγων: μία παράγωγο του σφάλματος ως προς την έξοδο του νευρώνα επί μία παράγωγο της εξόδου ως προς το βάρος.

$$\frac{\partial E_p}{\partial w_{ji}} = \frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial w_{ji}} \quad (4.6)$$

Η πρώτη παράγωγος μας λέει πως αλλάζει το σφάλμα ως προς την έξοδο του  $j$  νευρώνα, ενώ το δεύτερο τμήμα μας λέει πόσο η μεταβολή του  $w_{ji}$  αλλάζει αυτήν την έξοδο. Έτσι υπολογίζουμε κατευθείαν τις παραγώγους:

$$\frac{\partial E_p}{\partial o_{pj}} = (t_{pj} - o_{pj}) = -\delta_{pj} \quad (4.7)$$

Η συνεισφορά του νευρώνα  $j$  στο σφάλμα είναι ανάλογη του  $\delta_{pj}$ . Αφού έχουμε γραμμικές μονάδες:

$$o_{pj} = \sum_i w_{ji} x_{pi} \quad (4.8)$$

καταλήγουμε ότι:

$$\frac{\partial o_{pj}}{\partial w_{ji}} = x_{pi} \quad (4.9)$$

Αντικαθιστώντας στην εξίσωση (4.6) βλέπουμε ότι:

$$-\frac{\partial E}{\partial w_{ji}} = \delta_{pj} x_{pi} \quad (4.10)$$

όπως ακριβώς θέλουμε. Συνδυάζοντας την τελευταία αυτή εξίσωση με την παρατήρηση ότι

$$\frac{\partial E}{\partial w_{ji}} = \sum_p \frac{\partial E_p}{\partial w_{ji}} \quad (4.11)$$

οδηγούμαστε στο συμπέρασμα ότι η μεταβολή στο  $w_{ji}$  μετά από ένα πλήρη κύκλο, όπου παρουσιάζουμε όλα τα πρότυπα, είναι ανάλογη προς στην παράγωγο αυτή και ως εκ τούτου ο κανόνας Δέλτα εφαρμόζει μία επικλινή κάθοδο στο  $E$ . Κανονικά τα βάρη  $w$  δεν πρέπει να αλλάζουν κατά την διάρκεια του κύκλου που παρουσιάζουμε τα διάφορα πρότυπα, ένα-ένα, αλλά μόνο στο τέλος του κύκλου. Αν όμως ο ρυθμός εκπαίδευσης είναι μικρός, δεν δημιουργείται μεγάλο σφάλμα και ο κανόνας Δέλτα δουλεύει σωστά. Τελικά με τον τρόπο αυτό θα βρούμε τις τιμές των  $w$  που ελαχιστοποιούν την συνάρτηση σφάλματος.

### 4.3 Η μέθοδος εκπαίδευσης για μη-γραμμικούς νευρώνες

Δείξαμε πως ο κανόνας Δέλτα επιφέρει επικλινή κάθοδο στο τετράγωνο του αθροίσματος του σφάλματος για γραμμικές συναρτήσεις ενεργοποίησης. Στην περίπτωση που δεν έχουμε κρυμμένα επίπεδα, η επιφάνεια σφάλματος είναι σαν μία κοιλάδα με ένα μόνο ελάχιστο και έτσι η επικλινή κάθοδος πάντοτε θα βρίσκει τις καλύτερες τιμές για τα βάρη  $w$ . Στην περίπτωση όμως με τα κρυμμένα επίπεδα δεν είναι προφανές πως υπολογίζονται οι παράγωγοι. Η επιφάνεια σφάλματος δεν είναι κοίλη προς τα πάνω και έτσι υπάρχει η πιθανότητα να βρεθούμε σε ένα τοπικό ελάχιστο. Θα δείξουμε παρακάτω ότι υπάρχει ένας αποτελεσματικός τρόπος για τον υπολογισμό των παραγώγων, καθώς επίσης και ότι το πρόβλημα των τοπικών ελαχίστων συνήθως δεν είναι καταστροφικό, αφού πάντα έχουμε τρόπους να το ξεπεράσουμε και τελικά να πετύχουμε την εκπαίδευση του δικτύου.

Χρησιμοποιούμε εδώ δίκτυα με δομές πολλαπλών επιπέδων και στα οποία το σήμα διαδίδεται πάντοτε στην ίδια κατεύθυνση, από το επίπεδο εισόδου προς το επίπεδο εξόδου (feedforward). Το σήμα έρχεται στο επίπεδο εισόδου, στο πιο χαμηλό επίπεδο, επεξεργάζεται από το δίκτυο και προωθείται στα κρυμμένα επίπεδα. Τα κρυμμένα επίπεδα το επεξεργάζονται και το προωθούν στο επίπεδο εξόδου. Η επεξεργασία γίνεται πάντοτε επίπεδο προς επίπεδο, σε κάθε νευρώνα χωριστά. Υπολογίζεται σε κάθε νευρώνα η συνάρτηση ενεργοποίησης, χρησιμοποιώντας την μη-γραμμική σιγμοειδή συνάρτηση, παίρνοντας ως είσοδο την έξοδο του προηγούμενου επιπέδου και δίνοντας ως έξοδο προς το παραπάνω επίπεδο την υπολογιζόμενη τιμή. Για μια τέτοια, μη γραμμική συνάρτηση η έξοδος είναι:

$$S_{pj} = \sum_i w_{ji} o_{pi} \quad (4.12)$$

όπου  $o_{pi}$  είναι το σήμα εισόδου του νευρώνα  $i$ . Έτσι θα πρέπει:

$$o_{pj} = f_j(S_{pj}) \quad (4.13)$$

όπου η  $f$  είναι διαφορίσιμη και αύξουσα συνάρτηση. Γραμμικές συναρτήσεις εδώ δεν επαρκούν, διότι η παράγωγός τους είναι άπειρη στο κατώφλι και μηδέν στα άλλα σημεία. Θεωρούμε λοιπόν ότι:

$$\Delta_p w_{ji} \sim -\frac{\partial E_p}{\partial w_{ji}} \quad (4.14)$$

όπου  $\Delta_p w_{ji}$  είναι η αλλαγή που θα γίνει στο βάρος  $w$  το οποίο συνδέει τους νευρώνες  $i$  και  $j$ , μετά από παρουσίαση του προτύπου  $p$ . Επίσης,  $E$  είναι η συνάρτηση σφάλματος (άθροισμα τετραγώνων). Θέτουμε και εδώ την παράγωγο αυτή ως γινόμενο δύο παραγώγων: μία που δίνει την μεταβολή του σφάλματος ως προς την μεταβολή στην τιμή εισόδου και μία που δίνει την μεταβολή στην τιμή εισόδου ως προς την μεταβολή του βάρους. Έτσι:

$$\frac{\partial E_p}{\partial w_{ji}} = \frac{\partial E_p}{\partial S_{pj}} \frac{\partial S_{pj}}{\partial w_{ji}} \quad (4.15)$$

Με την εξίσωση (4.12) βλέπουμε ότι:

$$\frac{\partial S_{pj}}{\partial w_{ji}} = \frac{\partial}{\partial w_{ji}} \sum_k w_{jk} o_{pk} = o_{pi} \quad (4.16)$$

Ορίζουμε ότι:

$$\delta_{pj} = -\frac{\partial E_p}{\partial S_{pj}} \quad (4.17)$$

Ο ορισμός αυτός θα μπορούσε να θεωρηθεί αυστηρά ως αυθαίρετος, αλλά αν προσέξουμε λίγο βλέπουμε ότι είναι ανάλογος με τον ορισμό της εξίσωσης (4.7), όπου  $\delta_{pj} = (o_{pj} - t_{pj})$ , καθ' όσον  $o_{pj} = S_{pj}$  όταν οι νευρώνες είναι γραμμικοί. Η εξίσωση λοιπόν (4.15) γίνεται τώρα:

$$-\frac{\partial E_p}{\partial w_{ji}} = \delta_{pj} o_{pi} \quad (4.18)$$

Αυτό δηλώνει ότι για να εφαρμόσουμε την επικλινή κάθοδο ως προς  $E$  θα πρέπει να κάνουμε τις αλλαγές στα  $w$  ως εξής:

$$\Delta_p w_{ji} = \eta \delta_{pj} o_{pi} \quad (4.19)$$

όπως ακριβώς και στον συνήθη κανόνα Δέλτα. Η μορφή της εξίσωσης (4.19) δίνει τον γενικευμένο «κανόνα Δέλτα», όπου  $\Delta_p$  είναι το  $\Delta$  του προτύπου  $p$ . Τώρα πρέπει να υπολογίσουμε τα σωστά  $\delta_{pj}$  για κάθε νευρώνα του δικτύου. Θα αποδείξουμε τώρα μία αναδρομική σχέση για αυτά τα  $\delta$ , με την οποία μπορούμε να προωθήσουμε το σφάλμα προς τα πίσω, δηλ. από την έξοδο προς την είσοδο. Θέτουμε και εδώ την παράγωγο αυτή ως γινόμενο δύο παραγώγων: μία που δίνει την μεταβολή του σφάλματος ως συνάρτηση της εξόδου και μία που δίνει την μεταβολή της εξόδου ως συνάρτηση της μεταβολής της εισόδου. Έτσι έχουμε:

$$\delta_{pj} = -\frac{\partial E_p}{\partial S_{pj}} = -\frac{\partial E_p}{\partial o_{pj}} \frac{\partial o_{pj}}{\partial S_{pj}} \quad (4.20)$$

Αλλά από την εξίσωση (4.13) έχουμε ότι:

$$\frac{\partial o_{pj}}{\partial S_{pj}} = f'_j(S_{pj}) \quad (4.21)$$

που είναι η παράγωγος της συνάρτησης ενεργοποίησης για τον νευρώνα  $j$ , υπολογιζόμενη στο σήμα εισόδου  $S_{pj}$  στο νευρώνα αυτό. Τώρα υπολογίζουμε την πρώτη παράγωγο στην εξίσωση του  $\delta_{pj}$ . Εδώ χρειάζεται προσοχή, είναι το πιο λεπτό σημείο όλης της μεθόδου. Τον παράγοντα αυτόν τον υπολογίζουμε διαφορετικά αν ο νευρώνας είναι στο επίπεδο εξόδου ή εσωτερικός. Στην περίπτωση που είναι στο επίπεδο εξόδου τότε:

$$\frac{\partial E_p}{\partial o_{pj}} = -(t_{pj} - o_{pj}) \quad (4.22)$$

που είναι το ίδιο αποτέλεσμα όπως με τον συνήθη κανόνα Δέλτα. Αντικαθιστώντας τους δύο παράγοντες στην εξίσωση (4.20) παίρνουμε:

$$\delta_{pj} = (t_{pj} - o_{pj}) f'_j(S_{pj}) \quad (4.23)$$

για νευρώνες που είναι στο επίπεδο εξόδου. Για νευρώνες που είναι εσωτερικοί υπάρχει το πρόβλημα ότι δεν έχουμε κανένα  $t_{pj}$ , δηλ. δεν έχουμε τιμές των στόχων. Στην περίπτωση αυτή χρησιμοποιούμε και πάλι τον κανόνα αλυσίδας και έχουμε:

$$\begin{aligned} \sum_k \frac{\partial E_p}{\partial (S_{pk})} \frac{\partial (S_{pk})}{\partial o_{pj}} &= \sum_k \frac{\partial E_p}{\partial (S_{pk})} \frac{\partial}{\partial o_{pj}} \sum_i w_{ki} o_{pi} = \\ &= \sum_k \frac{\partial E_p}{\partial (S_{pk})} w_{kj} = -\sum_k \delta_{pk} w_{kj} \end{aligned} \quad (4.24)$$

Αντικαθιστώντας παρομοίως στην εξίσωση (4.20) παίρνουμε:

$$\delta_{pj} = f'_j(S_{pj}) \sum_k \delta_{pk} w_{kj} \quad (4.25)$$

η οποία εξίσωση αφορά τώρα νευρώνες που δεν είναι στην έξοδο αλλά σε εσωτερικό επίπεδο. Οι εξισώσεις (4.23) και (4.25) δίνουν τον τρόπο με τον οποίο υπολογίζονται όλα τα  $\delta$ , για όλους τους νευρώνες στο δίκτυο, και τα οποία χρησιμοποιούνται για να υπολογίσουμε την μεταβολή στα  $w$  σε όλο το δίκτυο. Η διαδικασία αυτή θεωρείται ότι είναι ένας γενικευμένος κανόνας Δέλτα, για μη-γραμμικούς νευρώνες.

Ως περίληψη, η παραπάνω διαδικασία μπορεί να συνοψισθεί σε τρεις εξισώσεις. Πρώτα, εφαρμόζουμε τον γενικευμένο κανόνα Δέλτα με τον ίδιο τρόπο όπως και τον γενικό κανόνα. Το  $w$  σε κάθε επίπεδο αλλάζει κατά μία ποσότητα που είναι ανάλογη του σήματος σφάλματος  $\delta$ , και ανάλογος επίσης της εξόδου  $o$ . Δηλαδή,

$$\Delta_p w_{ji} = \eta \delta_{pj} o_{pi} \quad (4.26)$$

Οι άλλες δύο εξισώσεις δίδουν το σήμα του σφάλματος. Η διαδικασία του υπολογισμού του σήματος αυτού είναι μία κυκλική διαδικασία που ξεκινάει από το επίπεδο εξόδου. Για ένα νευρώνα στο επίπεδο εξόδου το σφάλμα είναι:

$$\delta_{pj} = (t_{pj} - o_{pj}) f'_j(S_{pj}) \quad (4.27)$$

όπου  $f'_j(S_{pj})$  είναι η παράγωγος της συνάρτησης ενεργοποίησης. Για νευρώνες στα κρυμμένα επίπεδα δίδεται από:

$$\delta_{pj} = f'_j(S_{pj}) \sum_k \delta_{pk} w_{kj} \quad (4.28)$$

Οι τρεις αυτές εξισώσεις αποτελούν έναν κύκλο για την εκπαίδευση του δικτύου και επιφέρουν μία αλλαγή μόνο σε κάθε  $w$ . Το σύστημα ακολούθως επαναλαμβάνει τόσους κύκλους όσοι του χρειάζονται για να εκπαιδευτεί.

#### 4.4 Προσομοίωση του προβλήματος X-OR

Θα παρουσιάσουμε τώρα ως παράδειγμα την λεπτομερή λύση για το γνωστό πρόβλημα του X-OR, το οποίο συζητήσαμε και στο κεφάλαιο 3, και είδαμε ότι δεν μπορούμε να λύσουμε με ένα απλό δίκτυο ενός ή δύο επιπέδων. Θα χρειασθεί να προτείνουμε μία δομή που περιέχει ένα κρυμμένο επίπεδο, όπως περιγράφηκε παραπάνω. Μία τέτοια δομή φαίνεται στο Σχήμα 4.4. Η δομή του δικτύου περιλαμβάνει δύο νευρώνες στο επίπεδο εισόδου, ένα νευρώνα στο κρυμμένο επίπεδο και ένα στο επί-

πεδο εξόδου. Οι συνδέσεις είναι όπως στο Σχήμα. Είναι απαραίτητο οι εισόδοι να πηγαίνουν και στο κρυμμένο επίπεδο και στην έξοδο κατευθείαν. Το πρόβλημα αυτό έχει διδακτική σημασία, καθόσον η λύση του περιλαμβάνει όλες τις λεπτομέρειες της τεχνικής της οπισθοδιάδοσης και γι' αυτό θα σκιαγραφήσουμε την λύση του με μέθοδο προσομοίωσης, ακολουθώντας ένα-ένα τα βήματα και τις εξισώσεις.

Χρησιμοποιούμε την εξίσωση (4.1) για τον υπολογισμό των εξόδων από κάθε νευρώνα, την στιγμοειδή συνάρτηση, ως εξής:

$$o_{pj} = \frac{1}{1 + e^{-(\sum_i w_{ji} o_{pi} + \theta_j)}} \quad (4.29)$$

όπου  $\theta_j$  είναι η παράμετρος προδιάθεσης (ή προδιάθεση) και που παίζει κατά κάποιο τρόπο τον ρόλο του κατωφλίου, ή του εσωτερικού βάρους του νευρώνα. Οι τιμές της προδιάθεσης,  $\theta_j$ , θα διδαχθούν στο δίκτυο, όπως και οι τιμές των άλλων βαρών  $w$ . Παρατηρούμε δηλ. ότι εκτός από τα πέντε βάρη που συνδέουν νευρώνες μεταξύ τους ( $w_1$  ως  $w_5$ ), έχουμε και άλλα δύο βάρη, τα  $w_6$  και  $w_7$ , τα οποία θεωρούμε ότι είναι τα «εσωτερικά  $w$ » των δύο νευρώνων, του κρυμμένου επιπέδου και της εξόδου, και στην εξίσωση (4.29) έχουν τον ρόλο του  $\theta$ . Εδώ το  $\theta$  αυτό στην εξίσωση δεν πολλαπλασιάζεται επί την τιμή σήματος  $s$ , όπως γίνεται συνήθως, ή μπορούμε να πούμε ότι πολλαπλασιάζεται επί το  $s$  με τιμή  $s = 1$ . Με άλλα λόγια μπορούμε να πούμε ότι το σήμα  $S$  για τα εσωτερικά  $w$  είναι πάντα  $S = 1$ . Οι δύο νευρώνες της εισόδου δεν έχουν εσωτερικά βάρη, και έτσι δεν υπάρχει ο όρος αυτός στην εξίσωση (4.29).

Η εξίσωση της παραγώγου είναι ίδια με την εξίσωση (4.2). Προκύπτει λοιπόν ότι:

$$\frac{\partial o_{pj}}{\partial S_{pj}} = o_{pj}(1 - o_{pj}) \quad (4.30)$$

Επίσης οι εξισώσεις των  $\delta$  είναι οι ίδιες. Και πάλι χωρίζουμε τις δύο διαφορετικές περιπτώσεις και για μονάδες εξόδου έχουμε:

$$d_{pj} = (t_{pj} - o_{pj}) o_{pj}(1 - o_{pj}) \quad (4.31)$$

ενώ για εσωτερικές μονάδες έχουμε ότι:

$$\delta_{pj} = o_{pj}(1 - o_{pj}) \sum_k \delta_{pk} w_{kj} \quad (4.32)$$

Η παράγωγος  $o_{pj}(1 - o_{pj})$  έχει μέγιστο για  $o_{pj} = 0,5$  και πλησιάζει το ελάχιστό της όταν το  $o_{pj}$  πλησιάζει το 0 ή 1, καθ' ότι  $0 \leq o_{pj} \leq 1$ . Αυτό συμβαίνει διότι η παράγωγος είναι η κλίση της συνάρτησης και όπως βλέπουμε στην γραφική παράσταση της συνάρτησης στο Σχήμα 4.2 την μεγαλύτερη κλίση η καμπύλη του σχήματος την έχει εκεί



όπου η έξοδος είναι 0,5. Ανάλογα, την μικρότερη κλίση (σχεδόν 0) την έχει στα δύο άκρα της καμπύλης, δεξιά και αριστερά. Η μεταβολή σε ένα  $w$  είναι ανάλογη με την παράγωγο αυτή και έτσι τα  $w$  θα μεταβάλλονται περισσότερο για την περίπτωση που έχουμε μια μεσαία τιμή, δηλ. κοντά στο 0,5 (και όχι κοντά στο 0 ή στο 1) ενώ ακόμη δεν έχει αποφασισθεί αν ο νευρώνας θα είναι ενεργός ή μη-ενεργός. Αυτό το χαρακτηριστικό δίνει την σταθερότητα της λύσης του συστήματος.

Πρέπει επίσης να σημειώσουμε ότι όταν ελέγχουμε για τιμές εξόδου 0 ή 1, είναι αδύνατο στην προσομοίωση να πάρουμε ακριβώς τις τιμές αυτές, παρά μόνο αν θεωρήσουμε άπειρο αριθμό κύκλων ή αν έχουμε  $w$  που τείνουν στο άπειρο. Συνήθως είναι αρκετό όταν παίρνουμε κάποια συμφωνία της τάξης του 10% με τις τιμές αυτές, δηλ. όταν παίρνουμε 0,1 και 0,9 αντί για 0 και 1, αντίστοιχα, η ακρίβεια αυτή είναι ικανοποιητική.

Είδαμε ότι οι μεταβολές που επιφέρονται στα βάρη είναι ανάλογες της ποσότητας  $\partial E_p / \partial w$ . Στην σωστή μέθοδο με επικλινή κάθοδο (gradient descent) τα βήματα πρέπει να είναι πολύ μικρά και έτσι στην εξίσωση της μεταβολής των  $w$  εισέρχεται μία σταθερά, το  $\eta$ , που αντιπροσωπεύει τον ρυθμό εκπαίδευσης του δικτύου. Όσο μεγαλύτερο είναι το  $\eta$ , τόσο μεγαλύτερες είναι οι μεταβολές στα  $w$  και τόσο γρηγορότερα το δίκτυο εκπαιδεύεται. Αν όμως το  $\eta$  γίνει πολύ μεγάλο, τότε αυτό οδηγεί σε ταλαντώσεις και έτσι αναγκάζομαστε να μην μπορούμε να το αυξήσουμε πολύ. Ένας τρόπος να αυξήσουμε τον ρυθμό εκπαίδευσης και να αποφύγουμε τις ταλαντώσεις είναι να περιλάβουμε και έναν όρο ακόμα που δηλώνει την ορμή του συστήματος. Έτσι η νέα εξίσωση της μεταβολής των βαρών τώρα γίνεται:

$$\Delta w_{ji}(n+1) = \eta \delta_{pj} o_{pi} + \alpha \Delta w_{ji}(n) \quad (4.33)$$

όπου το  $n$  δηλώνει τον κύκλο,  $\eta$  τον ρυθμό εκπαίδευσης και  $\alpha$  είναι η σταθερά που λαμβάνει υπόψη τις προηγούμενες μεταβολές των  $w$  όταν υπολογίζει την νέα μεταβολή. Αυτή είναι μία μορφή ορμής του συστήματος που ουσιαστικά φιλτράρει μεταβολές υψηλής συχνότητας στην επιφάνεια σφάλματος. Συνήθως παίρνουμε μία τιμή του  $\alpha$  που είναι  $\alpha = 0,9$ .

## Άσκηση Αυτοαξιολόγησης 4.6

Όπως αναφέρθηκε πολλές φορές η δομή του δικτύου οπισθοδιάδοσης που παρουσιάστηκε με ένα νευρώνα στο κρυμμένο επίπεδο (Σχήμα 4.4) δεν είναι η μοναδική. Σχεδιάστε ένα δίκτυο με μία άλλη δομή που να έχει δύο νευρώνες στην είσοδο, δύο στο κρυμμένο επίπεδο και ένα στην έξοδο (2–2–1). Υπάρχει πλήρης συν-

δεσμολογία, αλλά μόνο από επίπεδο σε επίπεδο. Δίδονται είσοδοι 0,1 και 0,9 και ο στόχος στην έξοδο είναι 0,9. Τα βάρη αρχικά είναι όλα 0,3 από την είσοδο στο κρυμμένο επίπεδο και 0,2 από το κρυμμένο επίπεδο στην έξοδο. Επίσης οι τιμές της προδιάθεσης (δηλ. τα εσωτερικά βάρη) είναι όλες 0,4 και  $\eta = 0,25$ . Θα χρησιμοποιήσουμε ένα δίκτυο οπισθοδιάδοσης, παρόμοιο με αυτό της ενότητας 4.4, και ως συνάρτηση μεταφοράς την γνωστή μας σιγμοειδή συνάρτηση. Βρείτε τις τιμές των βαρών μετά από ένα πέρασμα οπισθοδιάδοσης.

## Δραστηριότητα 4.2

Για ένα δίκτυο παρόμοιο με αυτό της Άσκησης Αυτοαξιολόγησης 4.6 δώστε ως σήμα στην είσοδο το πρότυπο (0,1, 0,9), με στόχο στην έξοδο το 0,9, και τιμές στα αρχικά βάρη τα (0,1, -0,2, 0,1) (0,1, -0,1, 0,3) για το πρώτο επίπεδο, και τα (0,2, 0,2, 0,3) για το δεύτερο επίπεδο. Πραγματοποιήστε έναν ολόκληρο κύκλο του σήματος, δηλ. την μετάδοση από την είσοδο στην έξοδο και την επιστροφή-διόρθωση από την έξοδο στην είσοδο.

Για τις αρχικές τιμές των  $w$  μπορούμε να ακολουθήσουμε διαφορετικούς δρόμους. Δεν υπάρχει κάποιος κανόνας που να μας υποδείξει τις αρχικές αυτές τιμές. Μία πρώτη περίπτωση είναι να θεωρούμε αρχικά τις τιμές όπως είδαμε στο Σχήμα 4.4, όπου τα βάρη  $w_1 = w_2 = w_3 = w_5 = 1$  κτλ.

### Πίνακας 4.2

$H$	$\alpha$	Αριθμός κύκλων
0,1	0	$82000 \pm 24000$
0,9	0	$8000 \pm 2200$
0,1	0,1	$8100 \pm 2200$
0,9	0,9	$870 \pm 230$

Μία άλλη περίπτωση είναι να ξεκινήσουμε με τυχαίες τιμές των  $w$ , ας πούμε ότι διαλέγουμε τιμές που κυμαίνονται στο διάστημα  $-0,3 < x < 0,3$ . Τις τιμές αυτές τις παίρνουμε από μία γεννήτρια τυχαίων αριθμών που υπάρχει σε κάθε υπολογιστή. Χρησιμοποιούμε ομαλή κατανομή των αριθμών, δηλ. κάθε αριθμός στο διάστημα αυτό έχει την ίδια πιθανότητα. Αφού το δίκτυο εκπαιδευθεί, σημειώνουμε τον αριθμό κύκλων που χρειάστηκε για να γίνει αυτό. Ακολουθώντας επαναλαμβάνουμε την ίδια

διαδικασία με άλλες αρχικές τυχαίες τιμές. Επειδή τα αρχικά  $w$  τώρα είναι διαφορετικά, παίρνουμε διαφορετικό αριθμό κύκλων που απαιτούνται για την ίδια εκπαίδευση του δικτύου. Πραγματοποιούμε έτσι ένα αριθμό από επαναλήψεις και τέλος παίρνουμε τον μέσο όρο από όλες αυτές τις πραγματοποιήσεις. Τα αποτελέσματα του Πίνακα 4.2 είναι ο μέσος όρος από 10 τέτοιες πραγματοποιήσεις. Παρατηρούμε ότι η τυπική απόκλιση έχει μεγάλη τιμή, πράγμα που δείχνει ότι η λύση είναι άμεσα εξαρτώμενη από τα αρχικά  $w$ . Επίσης παρατηρούμε ότι, όταν το  $\eta$  έχει μεγάλη τιμή ( $\eta = 0,9$ ), το δίκτυο εκπαιδεύεται 10 φορές γρηγορότερα από όταν έχει μικρή τιμή ( $\eta = 0,1$ ), πράγμα αναμενόμενο όπως είδαμε παραπάνω. Υπάρχουν επίσης πιο εξελιγμένοι αλγόριθμοι στους οποίους το βήμα αυτό είναι αυτοπροσαρμοζόμενο, αλλά δεν θα τους εξετάσουμε εδώ.

Τέλος, υπάρχει μία σειρά από ερωτήματα κατά την διαδικασία παρουσίασης των προτύπων που καλό είναι να αναφέρουμε. Πρώτα είναι η σειρά παρουσίασης των προτύπων. Πρέπει να παρουσιάζονται τα πρότυπα στην είσοδο με μία δεδομένη σειρά, ή μπορούμε να τα εναλλάσσουμε με τυχαίο τρόπο; Και οι δύο τρόποι χρησιμοποιούνται σήμερα και επομένως η απάντηση εδώ είναι ότι εξαρτάται από το συγκεκριμένο πρόβλημα. Υπάρχουν, επίσης, δύο τρόποι σχετικά με το πότε γίνεται η διόρθωση των βαρών κατά την διαδικασία εκπαίδευσης. Ο πρώτος τρόπος είναι να γίνεται μετά από κάθε πέρασμα για κάθε πρότυπο (sequential mode of training) και επομένως μόλις παρουσιάσουμε το δεύτερο πρότυπο θα χρησιμοποιήσουμε τα ήδη αλλαγμένα  $w$  από το πρώτο πρότυπο κοκ. Ο δεύτερος τρόπος είναι να κάνουμε τις αλλαγές στα  $w$  μετά την παρουσίαση όλων των προτύπων (batch mode of training). Στο παράδειγμα του X-OR θα πρέπει να παρουσιάσουμε και τα 4 πρότυπα στην είσοδο, και μετά να κάνουμε τις αλλαγές στα  $w$ . Από τους δύο αυτούς τρόπους ο πρώτος συνήθως είναι προτιμότερος, γιατί συγκλίνει γρηγορότερα και χρειάζεται μικρότερη μνήμη για αποθήκευση. Επιπλέον είναι περισσότερο στοχαστικός και επομένως λιγότερο πιθανό το δίκτυο να πέσει σε τοπικό ελάχιστο. Ο δεύτερος όμως τρόπος είναι ευκολότερο να γίνει με παράλληλη επεξεργασία και επίσης μπορούμε να κάνουμε θεωρητική πρόβλεψη για το πότε το δίκτυο θα συγκλίνει. Παρόλα αυτά ο πρώτος τρόπος χρησιμοποιείται πιο συχνά.

Συνοψίζοντας λοιπόν την λύση του προβλήματος αυτού δίνουμε το διάγραμμα ροής, καθώς και τις εξισώσεις στα Σχήματα 4.5 και 4.6, παρακάτω.

**Υπολογισμός των  $\delta$** 

$$\delta[4] = \text{der}(o[4]) \cdot (t - o[4])$$

$$\delta[3] = \text{der}(o[3]) \cdot (\delta[4] \cdot w[4])$$

$$\delta[2] = \text{der}(o[2]) \cdot (\delta[4] \cdot w[5] + \delta[3] \cdot w[2])$$

$$\delta[1] = \text{der}(o[1]) \cdot (\delta[4] \cdot w[3] + \delta[3] \cdot w[1])$$

όπου:  $\text{der}(o[i])$  η παράγωγος της αντίστοιχης εξόδου

**Υπολογισμός των  $\Delta w$** **Αλλαγή των  $w$** 

$$\Delta w[1] = \eta \cdot \delta[3] \cdot o[1]$$

$$w_i[n+1] = w_i[n] + \Delta w_i[n] + p \cdot \Delta w_i[n-1]$$

$$\Delta w[2] = \eta \cdot \delta[3] \cdot o[2]$$

όπου  $i = 1, \dots, 7$

$$\Delta w[3] = \eta \cdot \delta[4] \cdot o[1]$$

$$\Delta w[4] = \eta \cdot \delta[4] \cdot o[3]$$

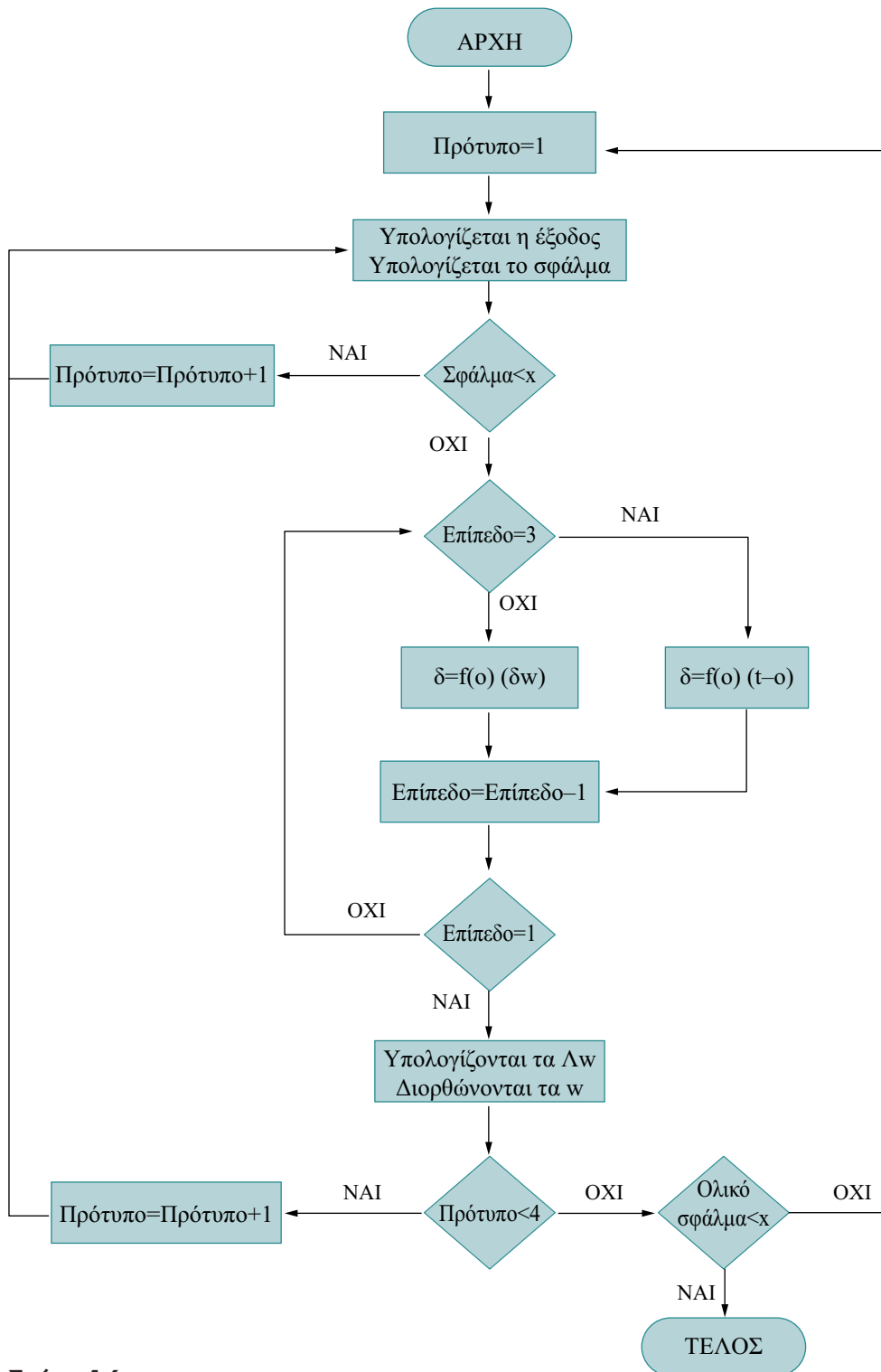
$$\Delta w[5] = \eta \cdot \delta[4] \cdot o[2]$$

$$\Delta w[6] = \delta[3]$$

$$\Delta w[7] = \delta[4]$$

**Σχήμα 4.5**

Περίληψη εξισώσεων εκπαίδευσης δικτύου

**Σχήμα 4.6**

Διάγραμμα ροής του αλγορίθμου επίλυσης του XOR

### Δραστηριότητα 4.3

#### Προσομοίωση προβλήματος X-OR

Η δραστηριότητα αυτή είναι ίσως το πιο σημαντικό τμήμα αυτού του κεφαλαίου, γιατί περιλαμβάνει την εκτέλεση ολόκληρης της διαδικασίας οπισθοδιάδοσης του σφάλματος. Αν καταφέρετε να την εκτελέσετε ολόκληρη, σε όλα τα βήματα, και να εκπαιδεύσετε ένα δίκτυο στο πρόβλημα X-OR, τότε θα έχετε καταλάβει πλήρως το πως λειτουργεί η πιο δημοφιλής μέθοδος στα νευρωνικά δίκτυα σήμερα. Συνίσταται λοιπόν να δώσετε προσοχή στα επόμενα βήματα.

- 1) Σχεδιάστε ένα δίκτυο το οποίο θα χρησιμοποιήσετε για να κάνετε μία πλήρη προσομοίωση του προβλήματος αυτού. Το δίκτυό σας θα μπορούσε να είναι παρόμοιο με αυτό στο Σχήμα 4.4, αλλά όχι απαραίτητα, θα μπορούσε λ.χ. να έχει δύο νευρώνες στο κρυμμένο επίπεδο αντί για ένα που έχει το Σχήμα 4.4. Συνθέστε τον κώδικα, σε όποια γλώσσα προτιμάτε, κωδικός που θα πραγματοποιεί όλα τα στάδια της εκπαίδευσης, όπως δίνονται στο διάγραμμα ροής στα Σχήματα 4.5 και 4.6. Το ζητούμενο είναι να εκπαιδεύσετε το δίκτυο στα πρότυπα του Πίνακα 3.1, δηλ. να βρείτε τις κατάλληλες τιμές των βαρών  $w$  που λύνουν το πρόβλημα X-OR.
- 2) Χρησιμοποιήστε και τους 4 συνδυασμούς για τις τιμές των  $a$  και  $\eta$  στον Πίνακα 4.2, και κατασκευάστε και εσείς τον αντίστοιχο πίνακα. Ξεκινήστε από τυχαίες τιμές των βαρών, οι οποίες θα πρέπει να κυμαίνονται στο διάστημα  $-0,3 < w < 0,3$ . Πραγματοποιήστε τουλάχιστον 10 φορές τον ίδιο υπολογισμό, αλλά με άλλες αρχικές τιμές των βαρών, και στο τέλος θα πάρετε τον μέσο όρο του αριθμού των κύκλων μέχρι το δίκτυο να εκπαιδευθεί. Δώστε 4 τέτοιους αριθμούς κύκλων για την εκπαίδευση του δικτύου σας, όπως και στον Πίνακα 4.2, καθώς και τις αντίστοιχες τιμές της τυπικής απόκλισης.
- 3) Οι στόχοι, όπως φυσικά ξέρουμε, έχουν δύο τιμές, το 0 και το 1. Δεν είναι απαραίτητο να τρέχει το πρόγραμμα μέχρι να βρείτε 0,0000 και 1,000, αλλά αρκεί στις τιμές αυτές μία ακρίβεια της τάξης του 10%, δηλ. αντί για 1,0 θα σταματήσετε στο 0,9 και αντί για 0,0 θα σταματήσετε στο 0,1. Η ακρίβεια αυτή αρκεί.
- 4) Ως συνάρτηση μεταφοράς χρησιμοποιήστε την εξίσωση 4.1
- 5) Προαιρετικά, μετατρέψτε το πρόγραμμά σας σε μία γενική μορφή, ώστε να λύνει το ίδιο πρόβλημα αλλά για οποιοδήποτε αριθμό κρυμμένων επιπέδων, αριθμό νευρώνων, κτλ. Δηλαδή, οι αριθμοί αυτοί θα πρέπει να δίνονται ως παράμετροι έξω από το πρόγραμμα και να μπορεί ο χρήστης να τις αλλάζει οποιαδήποτε στιγμή.

## Δραστηριότητα 4.4

Δημιουργήστε ένα δίκτυο οπισθοδιάδοσης για να ερευνήσετε πόσο καλά μπορεί να μάθει να υπολογίζει τιμές συναρτήσεων. Ουσιαστικά το δίκτυο θα μάθει να κάνει αντιστοιχίες τιμών μία-προς-μία. Χρησιμοποιήστε τις παρακάτω τέσσερεις συναρτήσεις στα αντίστοιχα διαστήματα:

$$1) f(x) = 1/x \quad 1 \leq x \leq 100$$

$$2) f(x) = \log_{10} x \quad 1 \leq x \leq 10$$

$$3) f(x) = \exp(-x) \quad 1 \leq x \leq 10$$

$$4) f(x) = \sin x \quad 0 \leq x \leq \pi/2$$

Χρησιμοποιήστε ένα κρυμμένο επίπεδο. Υπολογίστε τις συναρτήσεις για ένα αριθμό σημείων και ακολούθως εκπαιδεύστε το δίκτυο στην αντιστοίχιση των τιμών αυτών. Μεταβάλλετε τον αριθμό των νευρώνων στο κρυμμένο επίπεδο και δείτε έτσι πως μεταβάλλεται η ικανότητα του δικτύου να εκπαιδευθεί. Τέλος, δώστε τα κατάλληλα βάρη στα οποία καταλήγει το δίκτυο μετά την εκπαίδευση του.

## 4.5 Μειονεκτήματα και προβλήματα

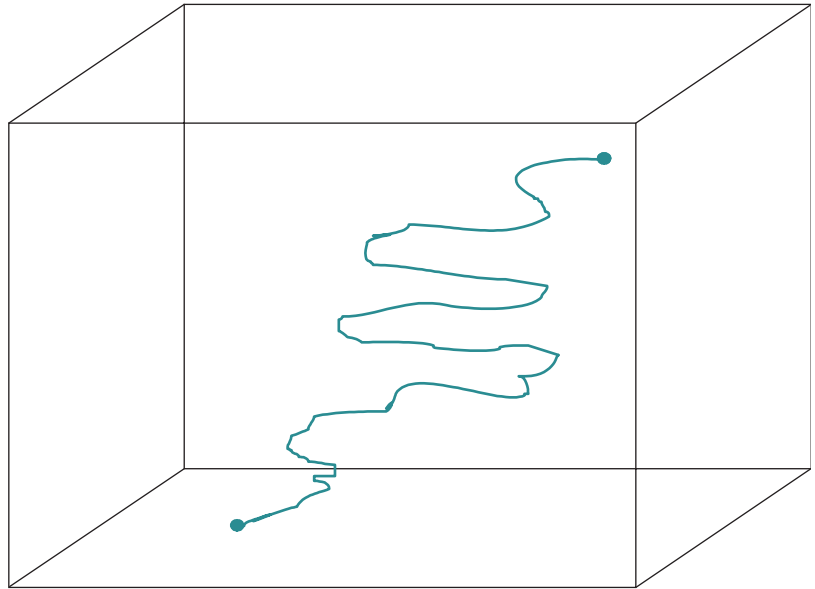
Παρά την μεγάλη επιτυχία της μεθόδου της οπισθοδιάδοσης, εν τούτοις υπάρχουν και περιπτώσεις που η μέθοδος αποτυγχάνει ή δεν δουλεύει άμεσα με επιτυχία. Σε τέτοιες περιπτώσεις συνήθως χρειάζεται να αλλάξουμε τιμές παραμέτρων, αρχικές συνθήκες κτλ., μέχρις ότου διορθωθεί το πρόβλημα.

Μερικές φορές ο χρόνος εκπαίδευσης είναι υπερβολικά μεγάλος. Χρειάζονται λ.χ. πολλά εκατομμύρια κύκλοι διόρθωσης μέχρις ότου το σύστημα συγκλίνει ή μπορεί και να μην συγκλίνει ποτέ. Σε τέτοιες περιπτώσεις πρέπει να αλλάξουμε το μέγεθος του βήματος. Αυτό συμβαίνει διότι τα βάρη μπορεί να πάρουν μεγάλες τιμές. Αυτό σημαίνει ότι πολλοί νευρώνες δίδουν μεγάλη τιμή εξόδου σε περιοχές όπου η παράγωγος της συνάρτησης εξόδου είναι πολύ μικρή. Καθόσον το σφάλμα που επιστρέφει από την έξοδο προς το κρυμμένο επίπεδο μέσα στο δίκτυο είναι ανάλογο της παραγωγού αυτής, μπορεί τότε η διαδικασία εκπαίδευσης να «κωλύσει». Τότε μικραίνουμε το μέγεθος του βήματος, αλλά αυτό έχει ως αποτέλεσμα να μεγαλώσει ο χρόνος εκπαίδευσης.

Ένα άλλο συχνό πρόβλημα είναι αυτό των τοπικών ελαχίστων. Η μέθοδος αυτή, όπως είδαμε παραπάνω, χρησιμοποιεί την μαθηματική τεχνική της επικλινούς καθόδου. Μία εικονική αναπαράσταση της καθόδου αυτής δίδεται στο Σχήμα 4.7, όπου

**Σχήμα 4.7**

Σχηματικό διάγραμμα της διαδρομής που ακολουθεί το σφάλμα κατά την εκπαίδευση του δικτύου, όπου αρχικά στο επάνω μέρος του κύβου το σφάλμα είναι μεγάλο, αλλά κατά τη διαδικασία της εκπαίδευσης σταδιακά ελαττώνεται φθάνοντας στο κάτω μέρος του κύβου.



βλέπουμε ότι το σφάλμα στην αρχή είναι μεγάλο αλλά σιγά-σιγά βρίσκει το ελάχιστο μέσα στον κύβο. Ακολουθείται η κλίση της επιφάνειας σφάλματος προς τα κάτω, μεταβάλλοντας συνεχώς τα βάρη μέχρι το σύστημα να φθάσει στο ελάχιστο. Το ελάχιστο αυτό όμως πρέπει να είναι το ολικό ελάχιστο. Η επιφάνεια μπορεί να έχει πολλά βουνά, λόφους, κοιλάδες, φαράγγια, χαράδρες κτλ. Αυτό σημαίνει ότι υπάρχουν πολλά τοπικά ελάχιστα, που είναι ψηλότερα από το ολικό ελάχιστο και στα οποία μπορεί εύκολα να παγιδευθεί το δίκτυο στην προσπάθειά του να βρει το ολικό ελάχιστο. Επειδή το σύστημα θέλει να πάει πάντα προς τα κάτω, αν πέσει σε ένα τοπικό ελάχιστο δεν έχει τρόπο να αποπαγιδευθεί μόνο του και να συνεχίσει τον δρόμο του, εκτός αν εκπαιδευτεί από την αρχή με νέα αρχικοποίηση. Συνήθως χρησιμοποιούμε στατιστικές μεθόδους εκπαίδευσης, για να αποφεύγεται το πρόβλημα αυτό (τις οποίες θα εξετάσουμε στο κεφάλαιο 5).

Το μέγεθος του βήματος επίσης παίζει σημαντικό ρόλο στην ταχύτητα εκμάθησης. Εάν είναι πολύ μικρό, τότε η εκπαίδευση αργεί υπερβολικά και πρέπει να το αυξήσουμε. Και εδώ η πιο σωστή και ιδανική λύση βρίσκεται με trial-and-error, δηλ. με πολλαπλές δοκιμές μέχρις ότου βρούμε την ιδανική τιμή.

Τέλος, θα πρέπει να θυμίσουμε ότι κατά την διαδρομή της εκπαίδευσης θα πρέπει να παρουσιάσουμε όλα τα πρότυπα, με ένα από τους δύο τρόπους που αναφέρθηκε παραπάνω, και τα πρότυπα πρέπει να παραμείνουν σταθερά. Οι αλλαγές των βαρών θα πρέπει επίσης να γίνονται στο δίκτυο μετά την παρουσίαση όλων των προτύπων. Αν όμως το δίκτυο βρίσκεται σε ένα περιβάλλον το οποίο συνεχώς αλλάζει πρότυ-



πα, τότε η εκπαίδευση του δικτύου δεν θα συγκλίνει ποτέ και το δίκτυο θα εκπαιδεύεται άσκοπα. Βλέπουμε λοιπόν στο σημείο αυτό ότι η μέθοδος αυτή δεν μιμείται τα βιολογικά συστήματα, τα οποία έχουν την ικανότητα να μαθαίνουν ακόμα και όταν αλλάζουν τα πρότυπα που παρουσιάζονται, με την ικανότητα που έχουν να τα ταξινομούν επιλεκτικά και να δίνουν διαφορετικό βάρος στα πρότυπα που τους παρουσιάζονται.

### Άσκηση Αυτοαξιολόγησης 4.7

Η μέθοδος της οπισθοδιάδοσης έχει μειονεκτήματα γιατί:

- (α) ο χρόνος εκπαίδευσης μπορεί να είναι υπερβολικά μεγάλος
- (β) το δίκτυο μπορεί να πέσει σε τοπικά ελάχιστα και να μην μπορεί να απεγκλωβισθεί
- (γ) το μέγεθος του βήματος πρέπει να επιλεγεί προσεκτικά
- (δ) αν τα πρότυπα που παρουσιάζονται αλλάζουν, το δίκτυο δεν εκπαιδεύεται
- (ε) όλα τα παραπάνω

## 4.6 Εφαρμογές

Όπως αναφέρθηκε παραπάνω η μέθοδος της οπισθοδιάδοσης είναι η πιο κοινή και ευρέως χρησιμοποιούμενη μέθοδος σήμερα για εκπαίδευση νευρωνικών δικτύων. Υπάρχουν πολλές εφαρμογές της όπως αυτές της οπτικής αναγνώρισης χαρακτήρων, της λήψης αποφάσεων κτλ.

Ένα άλλο παρόμοιο πακέτο, το NetTalk [10], αναπτύχθηκε από τους Sejnowski και Rosenberg (1987) που μετατρέπει με μεγάλη επιτυχία κείμενα Αγγλικών κατευθείαν σε ομιλία.

Υπάρχουν επίσης προσπάθειες και προγράμματα για την πιο δύσκολη διαδικασία, την αναγνώριση χειρογράφων κειμένων. Οι χαρακτήρες κανονικοποιούνται πρώτα ώστε να έχουν όλοι το ίδιο μέγεθος, μετά τοποθετούνται σε ένα πλέγμα και γίνονται οι προβολές των γραμμών στα τετράγωνα του πλέγματος. Οι προβολές αυτές είναι οι τιμές εισόδου για το δίκτυο. Η μέθοδος αυτή [11] αναπτύχθηκε από τον Burr (1987) και έχει > 99% επιτυχία.

Παρόμοιο πρόγραμμα έχει αναπτύξει και η εταιρία υπολογιστών NEC με ακρίβεια > 99%, αλλά η αναγνώριση γίνεται με άλλες μεθόδους. Το νευρωνικό δίκτυο οπι-

σθοδιάδοσης χρησιμοποιείται για να δώσει επιβεβαίωση των άλλων μεθόδων, αλλά διαπιστώθηκε ότι ο συνδυασμός αυτός έχει μεγαλύτερο ποσοστό επιτυχίας.

## Σύνοψη

Στο κεφάλαιο αυτό εξετάσαμε την μέθοδο εκπαίδευσης δικτύων με οπισθοδιάδοση. Η μέθοδος αυτή είναι εποπτευόμενη, καθότι πάντοτε δίδεται ο στόχος που πρέπει το δίκτυο να έχει ως έξοδο. Τα πρότυπα που παρουσιάζονται στο δίκτυο πρέπει να ανήκουν στην ίδια μορφή ή κατηγορία και πρέπει να είναι αρκετά τον αριθμό ώστε να λαμβάνουν υπόψη τους όλες τις πλευρές του προβλήματος. Η δομή του δικτύου πρέπει να περιέχει πάντοτε κρυμμένα επίπεδα. Έστω και ένα κρυμμένο επίπεδο θεωρητικά είναι αρκετό για πολλά προβλήματα, αλλά συνήθως χρειάζονται περισσότερα του ενός. Η συνάρτηση μεταφοράς πρέπει να έχει οπωσδήποτε μορφή σιγμοειδούς συνάρτησης για να παραμένει το δίκτυο σε πεπερασμένες τιμές. Χρησιμοποιήσαμε την πιο κοινή τέτοια συνάρτηση με την μορφή  $1 + \exp(-S)$  στον παρανομαστή κλάσματος. Αρχικά οι τιμές των βαρών είναι επιλεγμένες τυχαία και το δίκτυο εκπαιδεύεται με το να αναπροσαρμόζει τις τιμές αυτές. Η αναπροσαρμογή γίνεται με τον γενικευμένο κανόνα Δέλτα, ο οποίος ελαχιστοποιεί το τετράγωνο του σφάλματος που προκύπτει από τη διαφορά του στόχου από την εκάστοτε έξοδο του δικτύου. Η μέθοδος αυτή είναι μία μορφή της μαθηματικής μεθόδου της πλέον απότομης καθόδου. Κατά την εκπαίδευση το δίκτυο περνά από πολλές τέτοιες αναπροσαρμογές ή κύκλους μέχρις ότου το σφάλμα γίνει πολύ χαμηλό, όσο είναι ανεκτό στο πρόβλημα μας. Μετά την εκπαίδευση οι τελικές τιμές των βαρών παραμένουν σταθερές και το δίκτυο μπορεί να χρησιμοποιηθεί για να αναγνωρίσει νέα πρότυπα. Αρκετές φορές χρειάζεται μεγάλος υπολογιστικός χρόνος για την εκπαίδευση του δικτύου, πολλές χιλιάδες ή εκατομμύρια κύκλοι. Τέλος, πρέπει να τονισθεί ότι η οπισθοδιάδοση εφαρμόζεται σε δίκτυα των οποίων η δομή είναι καθορισμένη.

**Βιβλιογραφικές παραπομπές**

- [1] A. E. Bryson and Y. C. Ho, Applied Optimal Control, Blaisdell Publishing, New York 1969).
- [2] P. J. Werbos, Beyond regression: new tools for prediction and analysis in behavioral sciences, Ph.D thesis, Harvard University, Cambridge (Mass), (1974).
- [3] P. J. Werbos, Backpropagation through time: what it does and how to do it, Proceedings of the IEEE, **78**,1550(1990).
- [4] D. Parker, Learning logic. Invention report s 81–64, File 1, Office of Technology Licensing, Stanford University, Stanford, California ( 1982).
- [5] D. E. Rumelhart and J. L. McClelland, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volumes 1 and 2, MIT Press, Cambridge, Mass. (1986).
- [6] D. S. Touretzky and D. A. Pomerleau, What is hidden in the hidden layers?, Byte, **14**,227(1989).
- [7] D. E. Rumelhart G. E. Hinton, and R. J. Williams, Learning internal representations by error propagation, in Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Chapter 8, MIT Press, Cambridge, Mass. (1986).
- [8] D. E. Rumelhart G. E. Hinton, and R. J. Williams, Learning representations by back propagating errors, Nature, **323**,533(1988).
- [9] R. Hecht–Nielsen, Theory of Backpropagation Neural Networks, Proceedings of the International Joint Conference on Neural Networks (Cat. No. 89CH2765–6), **1**,593(1989).
- [10] T. J. Sejnowski and C. R. Rosenberg, Parallel networks that learn to pronounce English text, Complex Systems, **1**,145(1987).
- [11] D. J. Burr, Experiments with a Connectionist Text Reader, in Proceedings of the First International Conference on Neural networks, M. Caudill and C. Butler, eds. pp. 717, SOS Printing, San Diego (Calif.), (1987).