

Κεφάλαιο 10

Μάθηση και Γενίκευση

10.1 Ο στόχος της μάθησης

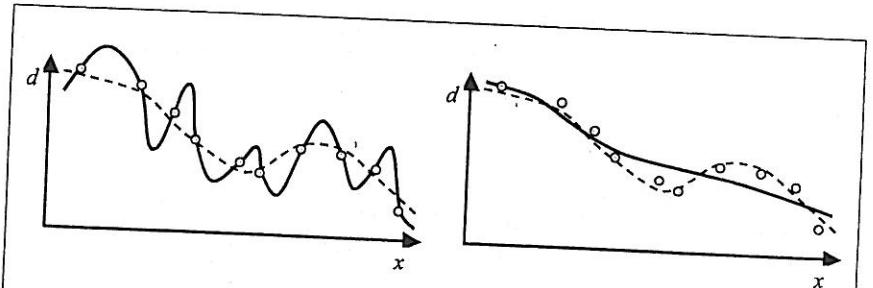
Μάθηση είναι η διαδικασία αυτοπροσαρμογής ενός συστήματος με στόχο να βελτιστοποιήσει κάποιο κριτήριο καταλληλότητας για την επίλυση ενός συγκεκριμένου προβλήματος. Ένα νευρωνικό δίκτυο λέμε ότι μαθαίνει ή εκπαιδεύεται όταν αλλάζει τις εσωτερικές του παραμέτρους (συνήθως τα συναπτικά του βάρη) με στόχο να ελαχιστοποιήσει ένα κριτήριο όπως για παράδειγμα, το μέσο τετραγωνικό σφάλμα. Αυτό δεν ισχύει μόνο για τα νευρωνικά δίκτυα αλλά και για κάθε μορφής σύστημα το οποίο μπορεί να βασίζεται σε συμβολικούς κανόνες ή σε κάποιο μη-νευρωνικό μαθηματικό μοντέλο. Έτσι η έννοια της μάθησης είναι πολύ γενικότερη και χαρακτηρίζει τα λεγόμενα «ευφυή συστήματα». Η ικανότητα μάθησης και της αυτοπροσαρμογής θεωρείται, δικαίως, ένα από τα βασικά συστατικά της ευφυΐας. Γενικά, ένα αυτόματο σύστημα που μαθαίνει ή εκπαιδεύεται χρησιμοποιώντας κάποια πρότυπα εισόδου αποκαλείται **μηχανή μάθησης**. Τα νευρωνικά δίκτυα είναι τυπικές περιπτώσεις μηχανών μάθησης.

Όπως είδαμε στο Κεφάλαιο 1.3 υπάρχουν δύο βασικά μοντέλα εκπαίδευσης: (α) η εκπαίδευση με επίβλεψη και (β) η εκπαίδευση χωρίς επίβλεψη.

Στην πρώτη περίπτωση για κάθε πρότυπο εισόδου $x^{(p)}$ δίνεται ένα διάνυσμα στόχου $d^{(p)}$ ενώ ορίζεται μια συνάρτηση κόστους $J(d^{(1)}, x^{(1)}, \dots, d^{(P)}, x^{(P)}; w)$ που κρίνει την καταλληλότητα του διανύσματος παραμέτρων w στην επίλυση του προβλήματος. Στόχος της εκπαίδευσης είναι η εύρεση του κατάλληλου διανύσματος w έτσι ώστε να ελαχιστοποιηθεί η συνάρτηση κόστους.

Στη δεύτερη περίπτωση δίνεται κάποιο κριτήριο καταλληλότητας το οποίο βελτιστοποιείται από κάποιον κανόνα μάθησης. Για παράδειγμα, στο δίκτυο SOM το κριτήριο είναι η ομοιότητα του νικητή νευρώνα και της γειτονιάς του με το διάνυσμα εισόδου. Η ομοιότητα αυτή αυξάνεται με τον αναδρομικό κανόνα του Kohonen. Ένα άλλο παράδειγμα είναι τα δίκτυα PCA, όπου το κριτήριο είναι η μεγιστοποίηση της στατιστικής διασποράς των εξόδων με τον περιορισμό ότι οι έξοδοι είναι στατιστικά ασυγχέτιστες μεταξύ

τους. Παρόμοια κριτήρια υπάρχουν σε κάθε μοντέλο μη επιβλεπόμενης μάθησης.



Σχήμα 74. Τα δύο είδη σφαλμάτων κατά τη μάθηση. Η εκτιμώμενη καμπύλη (συνεχής γραμμή) επιλέγεται μέσα από μια οικογένεια συναρτήσεων χρησιμοποιώντας ένα πεπερασμένο πλήθος δειγμάτων με θόρυβο (κουκίδες) τα οποία προέρχονται από κάποια πραγματική καμπύλη $d(x)$ (διακεκομένη γραμμή). Αριστερά: σφάλμα υπο-μοντελοποίησης. Η οικογένεια συναρτήσεων που επιλέχεται είναι πιο πολύπλοκη απ' ότι χρειάζεται. Τα δείγματα εκτιμώνται άριστα αλλά υπάρχει κακή εκτίμηση της πραγματικής καμπύλης μακριά από τα σημεία αντά. Δεξιά: σφάλμα υπο-μοντελοποίησης. Επιλογή απλούστερης οικογένειας συναρτήσεων από την ενδεδειγμένη. Ότε η εκτίμηση των δειγμάτων είναι ιδιαίτερα καλή ούτε το μοντέλο γενικεύει καλά μακριά από τα σημεία αντά.

Στο κεφάλαιο αυτό θα μας απασχολήσει κυρίως το θέμα των μηχανών μάθησης με επίβλεψη είτε αυτές είναι νευρωνικά μοντέλα είτε όχι. Ωστόσο, η χρησιμότητα της μάθησης είναι περιορισμένη αν απλώς προσφέρει τη δυνατότητα καλής προσέγγισης των στόχων μόνο για τα συγκεκριμένα πρότυπα που χρησιμοποιήθηκαν στην εκπαίδευση και δεν προσφέρει στο μοντέλο την ικανότητα να εκτιμά με επιτυχία του στόχους για πρότυπα που δεν έχει ξαναδεί. Η ικανότητα αυτή καλείται **γενίκευση** και είναι ακριβώς αυτή που διακρίνει τα ευφυή συστήματα από τα απλά συστήματα απομνημόνευσης όπως, πχ. μια απλή μνήμη RAM. Πράγματι, για ποιο λόγο να χρησιμοποιήσουμε ένα νευρωνικό δίκτυο αν πρόκειται απλώς να αποθηκεύσουμε στη μνήμη κάποια ζεύγη προτύπων-στόχων: $(x^{(1)}, d^{(1)})$, ..., $(x^{(P)}, d^{(P)})$;

Στο Σχήμα 74 περιγράφεται το πρόβλημα της γενίκευσης και τα σφάλματα στα οποία μπορούμε να υποπέσουμε κατά τη μάθηση. Συγκεκριμένα, στο μοντέλο της μάθησης με επίβλεψη προσπαθούμε να προσεγγίσουμε τις τις στόχους d για κάποιες τιμές εισόδου x που μας δίνονται (γκρι κουκίδες στο Σχήμα). Για παράδειγμα, ανάλογα με το πρόβλημά που μελετάμε, το d θα μπορούσε να είναι η ισχύς ενός κινητήρα και x οι στροφές του κινητήρα, είτε θα μπορούσε να είναι d η τιμή κλεισίματος μιας μετοχής στο χρηματι-

στήριο και χ ο χρόνος, κλπ. Προφανώς η ποικιλία των προβλημάτων είναι τεράστια όπως και τα πιθανά πεδία εφαρμογής. Το κοινό σημείο είναι ότι τηρούμενος πιθανά πεδία εφαρμογής. Το κοινό σημείο είναι ότι τηρούμενος πιθανά πεδία εφαρμογής. Το κοινό σημείο είναι ότι τηρούμενος πιθανά πεδία εφαρμογής. Το κοινό σημείο είναι ότι τηρούμενος πιθανά πεδία εφαρμογής.

Αν επιθυμούμε επιτυχημένη εκτίμηση της $f(x)$ τότε θα πρέπει να επιλέξουμε μια συνάρτηση που να την προσεγγίζει. Δυστυχώς όμως έχουμε μόνο τα πιθανά πεδία εφαρμογής. Το αν πετύχαμε καλή συγκεκριμένα σημεία κουκίδες στη διάθεσή μας. Το αν πετύχαμε καλή συγκεκριμένα σημεία κουκίδες στη διάθεσή μας. Το αν πετύχαμε καλή συγκεκριμένα σημεία κουκίδες στη διάθεσή μας. Το αν πετύχαμε καλή συγκεκριμένα σημεία κουκίδες στη διάθεσή μας.

(a) να επιλέξουμε μια συνάρτηση ιδιαίτερα πολύπλοκη για να προσεγγίζει τα δεδομένα που διαθέτουμε (Σχήμα 74-αριστερά). Το σφάλμα αυτό καλείται υπερ-μοντελοποίηση (over-modeling). Στην περίπτωση αυτή θα έχουμε κακή γενίκευση αν και μπορεί να έχουμε τέλεια προσέγγιση των διαθέσιμων δεδομένων (κουκίδες).

(β) να επιλέξουμε μια συνάρτηση πολύ απλή (Σχήμα 74-δεξιά). Το σφάλμα αυτό καλείται υπο-μοντελοποίηση (under-modeling). Στην περίπτωση αυτή θα έχουμε φτωχή προσέγγιση των διαθέσιμων δεδομένων και μετρια εκτίμηση των μη διαθέσιμων δεδομένων (γενίκευση).

Έτσι η επιλογή της κατάλληλης τάξης (ή αλλιώς της πολυπλοκότητας) του μοντέλου είναι κεφαλαιώδους σημασίας για την επιτυχημένη γενίκευση. Συνήθως η τάξη του μοντέλου είναι ανάλογη του πλήθους των ελεύθερων παραμέτρων. Σ' ένα νευρωνικό δίκτυο για παράδειγμα, η τάξη είναι ανάλογη του πλήθους των συναπτικών βαρών κι αυτό με τη σειρά του είναι ανάλογο του πλήθους των νευρώνων. Σύμφωνα με τα παραπάνω λοιπόν, τίθεται το θέμα πόσα συναπτικά βάρη ή πόσους νευρώνες πρέπει να έχει το δίκτυό μας. Πώς είμαστε σίγουροι, με τα πεπερασμένα δεδομένα που έχουμε, ότι μας κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου; Τέτοιους είδους κάναμε την κατάλληλη επιλογή της τάξης του μοντέλου;

σης και της γενίκευσης θα μας απασχολήσουν στην συνέχεια αυτού του κεφαλαίου.

10.1.1 Μαθαίνοντας να διακρίνουμε κλάσεις

Η πιο απλή περίπτωση προβλήματος ταξινόμησης είναι η διάκριση μεταξύ δύο κλάσεων προτύπων C_0 και C_1 . Η επίλυση του προβλήματος απαιτεί την εύρεση μιας διαχωριστικής συνάρτησης $g(x)$ έτσι ώστε αν το πρότυπο x ανήκει στην κλάση C_0 , η τιμή της g είναι 0, ενώ αν το πρότυπο x ανήκει στην κλάση C_1 , η τιμή της g είναι 1:

$$g(x) = \begin{cases} 0, & \text{αν } x \in C_0 \\ 1, & \text{αν } x \in C_1 \end{cases}$$

Η αναζήτηση της επιθυμητής συνάρτησης g είναι μάταιη αν δεν έχουμε ένα συγκεκριμένο πλαίσιο αναζήτησης καθώς όχι μόνο το πλήθος όλων των δυνατών συναρτήσεων όλων των δυνατών μορφών είναι υπερβολικά μεγάλο αλλά η συστηματική αναζήτηση της βέλτιστης συνάρτησης μέσα σε ένα τόσο χαώδες πλαίσιο είναι πρακτικά αδύνατη. Πρέπει τουλάχιστο, να ξέρουμε τη μορφή της συνάρτησης, ώστε η αναζήτηση να γίνεται συστηματικά. Για παράδειγμα, η μορφή μπορεί να είναι

- γραμμική:

$$g(x) = \mathbf{w}^T \mathbf{x} + w_0 \quad \text{ή, ισοδύναμα,}$$

$$g = \sum_{i=1}^n w_i x_i + w_0$$

- τετραγωνική:

$$g(x) = \mathbf{x}^T \mathbf{V} \mathbf{x} + \mathbf{w}^T \mathbf{x} + w_0 \quad \text{ή, ισοδύναμα,}$$

$$g = \sum_{i,j=1}^n x_i v_j x_j + \sum_{i=1}^n w_i x_i + w_0$$

- άθροισμα στιγμοειδών συναρτήσεων:

$$g = \sum_{i=1}^n \alpha_i f(\mathbf{w}_i^T \mathbf{x} + w_{i,0})$$

- κλπ.

Μέσα σ' ένα τέτοιο συγκεκριμένο πλαίσιο η διαδικασία της μάθησης είναι η επιλογή των κατάλληλων παραμέτρων της συνάρτησης. Πχ., για τη γραμμική συνάρτηση που είδαμε παραπάνω, οι παράμετροι αυτές είναι οι w_i , ενώ για τη τετραγωνική συνάρτηση οι παράμετροι είναι οι v_{ij} , w_i , κλπ. Η επιλογή γίνεται έτσι ώστε να ικανοποιηθεί το συγκεκριμένο κριτήριο μάθησης. Καθώς ο στόχος μας είναι ο διαχωρισμός των δύο κλάσεων, κριτήριο μάθησης θα μπορούσε να είναι, για παράδειγμα, η μεγιστοποίηση του πλήθους των προτύπων που ταξινομούνται σωστά.

Έτσι, η μάθηση ισοδυναμεί με την αναζήτηση της καταλληλότερης συνάρτησης μέσα σε μια οικογένεια συναρτήσεων $G = \{g(\mathbf{x}, \mathbf{w}) : \mathbf{w} \in W\}$ όπου W είναι το σύνολο όλων των δυνατών τιμών που μπορούν να πάρει το διάνυσμα των παραμέτρων \mathbf{w} . Όλες οι συναρτήσεις g μέσα στην οικογένεια G είναι της ίδιας μορφής. Αυτό που αλλάζει και που διαφοροποιεί τα μέλη της οικογένειας μεταξύ τους είναι η τιμή του διανύσματος \mathbf{w} .

Παράδειγμα 26: Η οικογένεια των γραμμικών συναρτήσεων n -εισόδων, 1-εξόδου

Μια από τις πιο απλές μορφές συναρτήσεων είναι η γραμμική, η οποία όπως είδαμε δίνεται από τον τύπο:

$$g(\mathbf{x}, \mathbf{w}) = w_0 + w_1x_1 + w_2x_2 + \dots + w_nx_n$$

Αν, ειδικότερα, υποθέσουμε ότι οι παράμετροι w_i και οι είσοδοι x_i είναι πραγματικοί αριθμοί τότε το διάνυσμα των παραμέτρων $\mathbf{w} = [w_0, w_1, w_2, \dots, w_n]^T$ ανήκει στο σύνολο $W = \mathbb{R}^{n+1}$, ενώ το διάνυσμα εισόδου $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ ανήκει στο σύνολο $X = \mathbb{R}^n$. Έχουμε έτσι την οικογένεια των γραμμικών συναρτήσεων με πραγματικές εισόδους και πραγματικές παραμέτρους

$$G_R = \{g(\mathbf{x}, \mathbf{w}) = w_0 + w_1x_1 + w_2x_2 + \dots + w_nx_n : \mathbf{w} \in \mathbb{R}^{n+1}, \mathbf{x} \in \mathbb{R}^n\}$$

Μια άλλη συνηθισμένη οικογένεια είναι αυτή στην οποία οι παράμετροι και οι είσοδοι είναι μιγαδικοί αριθμοί,

$$G_C = \{g(\mathbf{x}, \mathbf{w}) = w_0 + w_1x_1 + w_2x_2 + \dots + w_nx_n : \mathbf{w} \in C^{n+1}, \mathbf{x} \in C^n\}$$

Πόσο δύσκολο είναι να ψάξουμε μια οικογένεια συναρτήσεων G έτσι ώστε να βρούμε την καταλληλότερη συνάρτηση-μέλος της οικογένειας που να διακρίνει την κλάση C_0 από την κλάση C_1 ; Ποιες είναι οι διαχωριστικές 1-

Κεφάλαιο 10: Μάθηση και Γενίκευση

κανότητες της G ; Μπορεί η συγκεκριμένη οικογένεια συναρτήσεων να λύσει το πρόβλημά μας; Αν ναι, τότε πόσο χρόνο θα μας πάρει η αναζήτηση και αν όχι, τότε πόσο κοντά μπορούμε να φτάσουμε στη λύση; Είναι προφανές ότι όλα τα παραπάνω είναι γενικά ερωτήματα που εξαρτώνται από την περιγραφή της οικογένειας G , και από το συγκεκριμένο πρόβλημα που καλούμαστε να λύσουμε.

10.1.2 Γενίκευση και πολυπλοκότητα

Ένα από τα βασικότερα ερωτήματα που απασχολεί την θεωρία της μάθησης είναι πόσο κοντά έχει φτάσει μια μηχανή μάθησης στην πραγματική καμπύλη που παράγει τα δεδομένα. Η μηχανή μάθησης έχει εκπαιδευτεί χρησιμοποιώντας πεπερασμένο πλήθος προτύπων P , αλλά η εγγύτητά της με την πραγματική καμπύλη κρίνεται με βάση την σωστή εκτίμηση του στόχου για όλες τις τιμές εισόδου, οι οποίες είναι γενικά άπειρες. Το μέτρο της απόστασης αυτής αποκαλείται **ρίσκο (risk)** και έχει την έννοια της πιθανότητας να κάνουμε λάθος ταξινόμηση γενικά στα πρότυπα του προβλήματος, είτε αυτά τα έχουμε δει κατά τη διάρκεια της εκπαίδευσης είτε όχι.

Για να διατυπώσουμε αυτό το ρίσκο με μαθηματικό τρόπο ας υποθέσουμε ότι $f(\mathbf{x}; \theta)$ είναι η οικογένεια των συναρτήσεων που μπορεί να αναπαραστήσει μια μηχανή μάθησης, όπου \mathbf{x} είναι το διάνυσμα εισόδου και θ είναι το διάνυσμα των παραμέτρων που εκπαιδεύονται. Υποθέτουμε ότι η μηχανή βγάζει μια μόνο τιμή στην έξοδο (όχι δηλαδή διάνυσμα) και η τιμή αυτή είναι 1 ή -1 . Για παράδειγμα, αν η μηχανή μας είναι ένα δίκτυο Perceptron, τότε $f(\mathbf{x}; \theta) = \text{sign}(\mathbf{w}^T \mathbf{x} + w_0)$, με $\theta = [\mathbf{w}, w_0]$. Αν πάλι η μηχανή μας είναι ένα δίκτυο MLP με τόσες εισόδους όση η διάσταση του διανύσματος \mathbf{x} , με L κρυφά στρώματα, και με μια έξοδο, τότε η οικογένεια συναρτήσεων είναι σαφώς πιο πολύπλοκη και το διάνυσμα θ αποτελείται από τα συναπτικά βάρη και τα κατώφλια όλων των νευρώνων σε όλα τα στρώματα του δικτύου.

Για τον υπολογισμό του ρίσκου απαιτείται η περιγραφή του σφάλματος ταξινόμησης από κάποια συνάρτηση $L(f(\mathbf{x}; \theta), d)$, όπου $d \in \{-1, 1\}$ είναι ο δείκτης της κλάσης του \mathbf{x} , δηλαδή ο στόχος για το πρότυπο αυτό. Χρησιμοποιώντας τη λογική Boole, έχουμε σφάλμα (δηλαδή έχουμε $L = 1$) αν $d = 1$ και $f(\mathbf{x}; \theta) = -1$ ή $d = -1$ και $f(\mathbf{x}; \theta) = 1$, ενώ δεν έχουμε σφάλμα (δηλαδή έχουμε $L = 0$) αν $d = 1$ και $f(\mathbf{x}; \theta) = 1$ ή $d = -1$ και $f(\mathbf{x}; \theta) = -1$. Όλη αυτή η λογική συμπυκνώνεται στην παρακάτω συνάρτηση

$$L(f(\mathbf{x}; \theta), d) = \text{step}(d \cdot f(\mathbf{x}; \theta)) \quad (224)$$

όπου step είναι η βηματική συνάρτηση 0/1. Με βάση αυτά, για ένα δεδομένο διάνυσμα παραμέτρων θ , το ρίσκο R είναι η αναμενόμενη τιμή του σφάλματος

$$R[f] = E\{L(f(\mathbf{x}; \theta), d)\}. \quad (225)$$

Δυστυχώς το παραπάνω ρίσκο δεν μπορεί να υπολογιστεί καθώς δε γνωρίζουμε την κατανομή πιθανότητας των \mathbf{x}, d . Αυτό που μπορούμε να υπολογίσουμε είναι το **εμπειρικό ρίσκο** R_{emp} που υπολογίζεται πρακτικά από τη μέση τιμή του σφάλματος πάνω σε κάποιο συγκεκριμένο σύνολο προτύπων με τους αντίστοιχους στόχους $T = \{(\mathbf{x}^{(1)}, d^{(1)}), \dots, (\mathbf{x}^{(P)}, d^{(P)})\}$:

$$R_{\text{emp}}[f] = \frac{1}{P} \sum_{i=1}^P L(f(\mathbf{x}^{(i)}; \theta), d^{(i)}). \quad (226)$$

Η διαφορά μεταξύ των δύο ρίσκων, πραγματικού και εμπειρικού, είναι ότι το πρώτο βασίζεται σε όλα τα πρότυπα, ακόμα και αν είναι άπειρα στο πλήθος, ενώ το δεύτερο βασίζεται σε ένα συγκεκριμένο σύνολο T με P πρότυπα. Το πραγματικό ρίσκο $R[f]$ είναι το μέτρο της ικανότητας γενίκευσης της μηχανής μας. Αυτό είναι που πραγματικά επιθυμούμε να ελαχιστοποιήσουμε. Είναι λογικό να υποθέσουμε ότι για μεγάλες τιμές του P τα δύο ρίσκα πλησιάζουν μεταξύ τους, αλλά για μικρές τιμές του P είναι πιθανό να υπάρχουν μεγάλες αποκλίσεις. Ωστόσο οι έννοιες του μεγάλου και του μικρού είναι σχετικές. Πόσο μεγάλο πρέπει να είναι το P για το πρόβλημα που έχουμε στα χέρια μας; Εξαρτάται το P από την οικογένεια των συναρτήσεων ή μόνο από τα πρότυπα και τους στόχους τους;

Οι Vapnik και Chervonenekis έδωσαν μια μερική απάντηση στα παραπάνω ερωτήματα υπολογίζοντας ένα θεωρητικό άνω όριο της διαφοράς μεταξύ των ρίσκων, $R[f] - R_{\text{emp}}[f]$. Το όριο αυτό ονομάζεται **διάστημα εμπιστοσύνης** (**confidence interval**) και είναι συνάρτηση της πολυπλοκότητας της οικογένειας \mathcal{G} από την οποία αντλείται η f . Το μέτρο αυτής της πολυπλοκότητας είναι ένας αριθμός που υπεισέρχεται στις εξισώσεις και καλείται **διάσταση Vapnik-Chervonenekis** ή **διάσταση-VC** της οικογένειας \mathcal{G} . Σε γενικές γραμμές, όσο πιο «πλούσια» είναι μια οικογένεια συναρτήσεων, δηλαδή όσο πιο πολύπλοκες καμπύλες περιέχει, τόσο πιο μεγάλη είναι η διάσταση VC. Λεπτομερής ορισμός δίνεται στην ενότητα 10.1.3 όπου γίνεται και εκτεταμένη συζήτηση γύρω από τη διάσταση VC. Το βασικό αποτέλεσμα δίνεται στο παρακάτω Θεώρημα [226,227]

Θεώρημα 9.

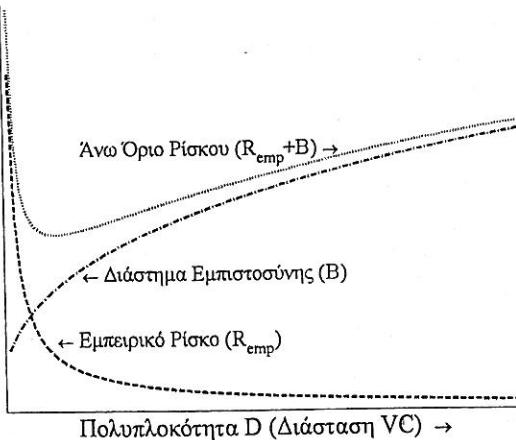
Έστω D η διάσταση VC της οικογένειας συναρτήσεων \mathcal{G} . Τότε με πιθανότητα τουλάχιστον $1 - \delta$ ισχύει ότι για κάθε συνάρτηση $f \in \mathcal{G}$ και για κάθε πλήθος προτύπων $P > D$

$$R[f] - R_{\text{emp}}[f] \leq B$$

$$\text{όπου } B = \sqrt{\frac{D[1 + \ln(\frac{2P}{D})] - \ln(\delta/4)}{P}} \quad (227)$$

Εποι, μπορεί να μην ξέρουμε ακριβώς την τιμή $R[f]$ για μια συγκεκριμένη συνάρτηση f , ξέροντας όμως το διάστημα εμπιστοσύνης B , ξέρουμε ότι η καλύτερη δυνατή συνάρτηση δίνει ρίσκο $R[f] \leq R_{\text{emp}}[f] + B$. Από την (227) προκύπτουν οι εξής παρατηρήσεις:

- Το διάστημα εμπιστοσύνης B είναι φθίνουσα συνάρτηση του πλήθους των προτύπων P . Αυτό είναι λογικό, καθώς μας λέει ότι όσο πιο πολλά πρότυπα έχουμε τόσο πιο καλή εκτίμηση του πραγματικού ρίσκου μας δίνει το εμπειρικό ρίσκο.
- Το διάστημα εμπιστοσύνης B είναι αύξουσα συνάρτηση του D , δηλαδή της διάστασης-VC. Αυτό σημαίνει ότι όσο πιο πολύπλοκη οικογένεια συναρτήσεων \mathcal{G} επιλέγουμε τόσο πιο μεγάλο διάστημα εμπιστοσύνης B παίρνουμε. Βέβαια όσο πιο πολύπλοκη είναι η οικογένεια \mathcal{G} τόσο πιο μικρό εμπειρικό ρίσκο επιτυγχάνουμε. Έτσι το άθροισμα $R_{\text{emp}}[f] + B$ που είναι το άνω όριο του πραγματικού ρίσκου ενώ στην αρχή μειώνεται καθώς αυξάνει το D , από ένα σημείο και μετά αρχίζει να αυξάνεται (βλ. Σχήμα 75). Αυτό σημαίνει ότι όταν η διάσταση VC ξεπεράσει την τιμή αυτή επιτυγχάνουμε οριακή βελτίωση του εμπειρικού ρίσκου με ταυτόχρονη σημαντική αύξηση του διαστήματος εμπιστοσύνης. Με άλλα λόγια μαθαίνουμε πολύ καλά τα πρότυπα που μας δόθηκαν για εκπαίδευση αλλά έχουμε πολλή αυξανόμενη ασφαλεία σε σχέση με το πραγματικό ρίσκο.



Σχήμα 75. Όσο αυξάνει η πολυπλοκότητα της οικογένειας των συναρτήσεων (α) μειώνεται το εμπειρικό ρίσκο και (β) αυξάνεται το διάστημα εμπιστοσύνης. Το άθροισμα των δύο αποτελεί και το άνω όριο του πραγματικού ρίσκου. Αντό μειώνεται στην αρχή αλλά μετά από κάποιο σημείο αυξάνεται. Μετά από αυτό το σημείο χρησιμοποιούμε περισσότερη πολυπλοκότητα απ' όση χρειάζεται (overfitting).

Με βάση τα παραπάνω οι Vapnik και Chervonenkis [226,227] ανέπτυξαν τη θεωρία της **Ελαχιστοποίησης του Δομικού Ρίσκου (Structural Risk Minimization – SRM)**. Σύμφωνα με αυτή, εκπαιδεύουμε μια σειρά από N μηχανές μάθησης. Κάθε μηχανή M_i επιλέγει την καλύτερη συνάρτηση από μια οικογένεια συναρτήσεων G_i με $\text{VCdim}(G_i) = D_i$. Έτσι ώστε να ελαχιστοποιείται το R_{emp} , δηλαδή το σφάλμα εκπαίδευσης. Οι οικογένειες είναι διατεταγμένες κατά αύξουσα σειρά διαστάσεων VC: $D_1 < D_2 < D_3 < \dots < D_N$, οπότε οι μηχανές μάθησης είναι όλοι και πιο πολύπλοκες. Το ζητούμενο είναι να επιλέξουμε τη μηχανή εκείνη που επιτυγχάνει το μικρότερο δυνατό σφάλμα γενίκευσης. Καθώς το άθροισμα $R_{emp} + B$ είναι η καλύτερη εκτίμηση που έχουμε για το πραγματικό ρίσκο επιλέγουμε τη μηχανή εκείνη που ελαχιστοποιεί το άθροισμα αυτό.

10.1.3 Η διάσταση Vapnik-Chervonenkis

Όπως είδαμε στη θεωρία SRM, η διαφορά μεταξύ εμπειρικού και πραγματικού ρίσκου είναι συνάρτηση ενός αριθμού ενδεικτικού της πολυπλοκότητας της οικογένειας συναρτήσεων G . Ο αριθμός αυτός λέγεται **διάσταση Vapnik-Chervonenkis (διάσταση VC)** πήρε το όνομά της από τους εισηγητές της σχετικής θεωρίας Ρώσους μαθηματικούς Vladimir Vapnik και Alexey Chervonenkis [228,229,226]. Για να ορίσουμε, αλλά και για να κατανοήσουμε τη διάσταση VC χρειάζεται να δώσουμε τους παρακάτω ορισμούς:

- **Διχοτόμηση** N σημείων στο χώρο \mathbb{R}^n είναι ο διαχωρισμός τους σε 2 ομάδες: στα σημεία της πρώτης ομάδας δίνουμε την ετικέτα 0 ενώ στα σημεία της άλλης ομάδας δίνουμε την ετικέτα 1.
- **Κατακερματισμός.** Λέμε ότι μια οικογένεια συναρτήσεων G κατακερματίζει ένα σύνολο N σημείων στο χώρο \mathbb{R}^n αν για κάθε πιθανή διχοτόμηση των σημείων αυτών υπάρχει μια συνάρτηση μέλος της οικογένειας που διακρίνει τα σημεία της μιας ομάδας από τα σημεία της άλλης.

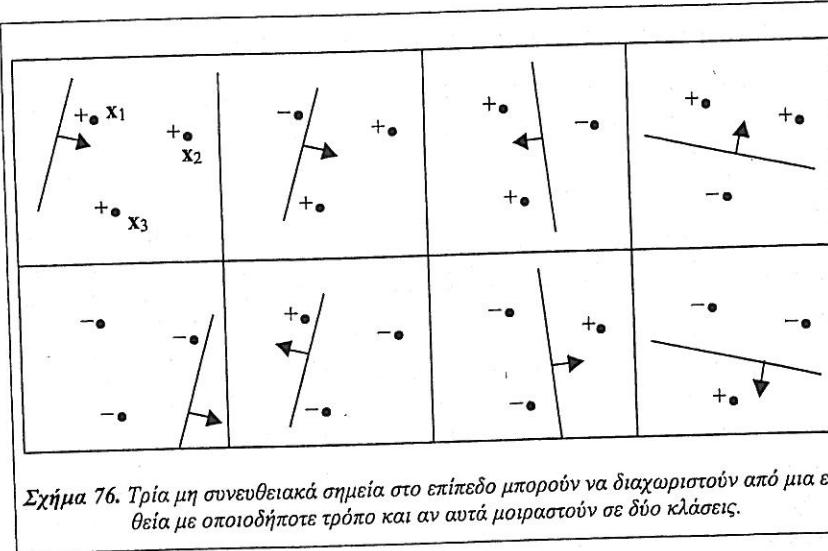
Παράδειγμα 27: Κατακερματισμός τριών σημείων από ευθείες στο δισδιάστατο επίπεδο

Θεωρήστε την οικογένεια F_2 των ευθειών στο δισδιάστατο επίπεδο \mathbb{R}^2 η οποία περιγράφεται από τον γενικό τύπο:

$$g(x, w) = w_0 + w_1x_1 + w_2x_2$$

με ελεύθερες παραμέτρους w_0, w_1, w_2 . Θεωρήστε επίσης οποιαδήποτε τρία σημεία x_1, x_2, x_3 , στο ίδιο επίπεδο (Σχ. 76) τα οποία δεν βρίσκονται στην ίδια ευθεία. Όπως φαίνεται στο Σχ. 76, οποιονδήποτε συνδυασμό επικετών και αν δώσουμε στα σημεία αυτά υπάρχει πάντα κάποια ευθεία που διαχωρίζει τα σημεία της μιας κατηγορίας από τα σημεία της άλλης. Συνεπώς η οικογένεια F_2 κατακερματίζει 3 σημεία στο \mathbb{R}^2 .

Είναι προφανές ότι δεν ισχύει το ίδιο για 4 σημεία. Για παράδειγμα, είναι γνωστό ότι ο συνδυασμός επικετών που αντιστοιχεί στο πρόβλημα XOR δεν διαχωρίζεται από καμία ευθεία.



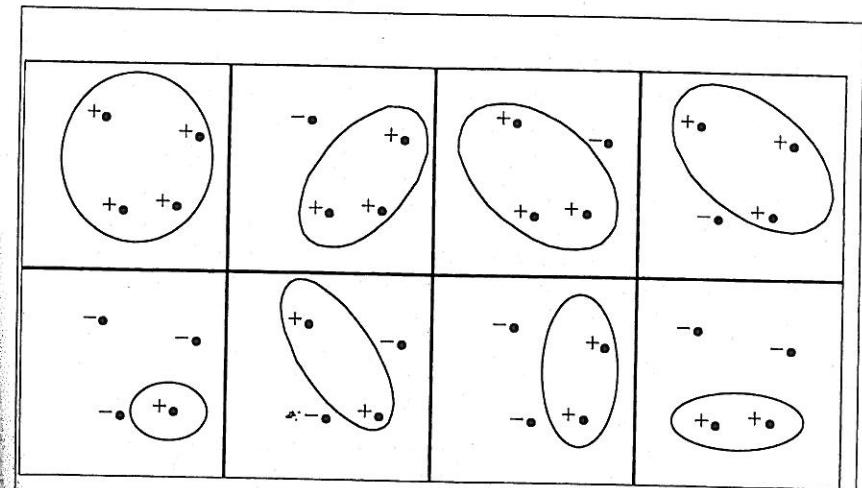
Η διάσταση VC μιας οικογένειας συναρτήσεων G που απεικονίζουν το χώρο \mathbb{R}^n στο χώρο \mathbb{R}^m είναι ίση με D αν η G μπορεί να κατακερματίσει ένα σύνολο D σημείων στο χώρο \mathbb{R}^m . Συμβολισμός: $\text{VCDim}(G) = D$.

Παράδειγμα 28: Κατακερματισμός τεσσάρων σημείων από ελλείψεις στο δισδιάστατο επίπεδο

Θεωρήστε την οικογένεια G_e των ελλείψεων στο δισδιάστατο επίπεδο \mathbb{R}^2 . Όπως φαίνεται στο Σχήμα 77, η οικογένεια αυτή μπορεί να κατακερματίσει τέσσερα σημεία. Συνεπώς η διάσταση VC είναι τουλάχιστον 4.

Παράδειγμα 29: Οικογένεια συναρτήσεων με άπειρη διάσταση VC

Καθώς η διάσταση VC είναι ένα μέτρο της πολυπλοκότητας μιας οικογένειας συναρτήσεων μπορούμε να υποπέσουμε στο σφάλμα να θεωρήσουμε ότι η διάσταση αυτή είναι ίση με το πλήθος των ελεύθερων παραμέτρων της οικογένειας. Η εσφαλμένη αυτή εντύπωση ενισχύεται από το γεγονός ότι στις γραμμικές συναρτήσεις αυτό όντως ισχύει, δηλαδή η διάσταση VC είναι $n+1$ δηλαδή πράγματι ίση με το πλήθος των ελεύθερων παραμέτρων.



Σχήμα 77. Τέσσερα σημεία στο επίπεδο μπορούν να χωριστούν από μια έλλειψη με οποιοδήποτε τρόπο και αν αυτά μοιραστούν σε δύο κλάσεις.

Όπως θα δούμε όμως αμέσως παρακάτω η οικογένεια συναρτήσεων

$$G_{\sin} = \{ g(x; \omega) = \text{sign}(\sin(\omega x)), \text{ για όλα } \omega \in \mathbb{R} \}$$

έχει άπειρη διάσταση VC παρόλο που έχει μόνο μια ελεύθερη παράμετρο ω . Στον παραπάνω ορισμό η συνάρτηση $\text{sign}(\cdot)$ είναι η συνάρτηση πρόστημου. Πράγματι, ας πάρουμε οποιοδήποτε πλήθος σημείων D , και ας πάρουμε τα σημεία

$$x^{(i)} = 10^{-i}, \quad i = 1, \dots, D$$

τότε όποιες και αν είναι οι ετικέτες $d^{(i)} \in \{-1, 1\}$, $i = 1, \dots, D$, μπορούμε να διαχωρίσουμε τα σημεία αυτά θέτοντας

$$\omega = \frac{\pi}{2} \sum_{i=1}^D (1 - d^{(i)}) 10^i + \pi$$

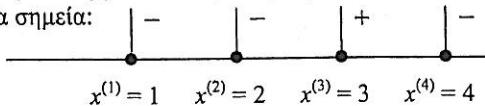
οπότε

$$\omega x_i = (1 - d^{(i)}) \frac{\pi}{2} + \frac{\pi}{2} \sum_{j=0}^{i-1} (1 - d^{(j)}) 10^{j-i} + \frac{\pi}{2} \sum_{j=i+1}^D (1 - d^{(j)}) 10^{j-i} + \pi 10^{-i} \quad (228)$$

Ο δεύτερος όρος στο δεξί μέλος της (228) είναι πολλαπλάσιο του 10π , συνεπώς, δεν επηρεάζει το πρόστημα του ημιτόνου. Επίσης το άθροι-

σμα του τρίτου και τέταρτου όρου στο δεξί μέλος της (228) είναι μικρότερο από $\pi/2$, συνεπώς και αυτό δεν επηρεάζει το πρόσημο του ημιτόνου. Άρα το πρόσημο του ημιτόνου του $(\omega x^{(i)})$ είναι ίσο με το πρόσημο του ημιτόνου του πρώτου όρου $(1-d^{(i)})\pi/2$, δηλαδή είναι -1 αν $d^{(i)} = -1$ και 1 αν $d^{(i)} = 1$. Συνεπώς τα D σημεία ταξινομούνται σωστά σύμφωνα με την κλάση στην οποία ανήκουν.

Είναι ενδιαφέρον να σημειώσουμε ότι παρ' όλο που η οικογένεια αυτή μπορεί να κατακερματίσει απεριόριστο πλήθος σημείων αν αυτά βρίσκονται στη διάταξη $x^{(i)} = 10^{-i}$, δεν μπορεί να διαχωρίσει τα παράκατω τέσσερα σημεία:



Σε κάποιες περιπτώσεις είναι εύκολο να υπολογιστεί η διάσταση VC σε σχέση με το πλήθος των ελεύθερων παραμέτρων της οικογένειας των συναρτήσεων. Για παράδειγμα, το απλό δίκτυο Perceptron με n εισόδους και 1 έξοδο έχει διάσταση $n+1$ όπως και οι γραμμικές συναρτήσεις. Στις περισσότερες ωστόσο περιπτώσεις, η διάσταση VC μπορεί να υπολογιστεί μόνο κατά προσέγγιση. Για παράδειγμα, οι Koiran και Sontag [135] απέδειξαν ότι η διάσταση VC της οικογένειας των συναρτήσεων που υλοποιούνται από ένα δίκτυο MLP είναι της τάξεως $O(N^2)$ όπου N είναι το πλήθος των ελεύθερων παραμέτρων του δικτύου, δηλαδή το συνολικό πλήθος των συναπτικών βαρών και των κατωφλίων.

10.2 Πρακτικές μέθοδοι βελτίωσης της ικανότητας γενίκευσης

Η θεωρία SRM των Vapnik-Chervonenkis αν και πολύ σημαντική σε θεωρητικό επίπεδο, δεν προσφέρει ιδιαίτερα πρακτικές μεθόδους αναζήτησης της βέλτιστης μηχανής μάθησης ως προς την ικανότητα γενίκευσης. Η δυσκολία προκύπτει από την αδυναμία μας να υπολογίσουμε τη διάσταση VC επακριβώς για τις περισσότερες οικογένειες συναρτήσεων που μας ενδιαφέρουν στην πράξη. Για παράδειγμα, αν έχουμε την οικογένεια των MLP η καλύτερη προσέγγιση της διάστασης VC είναι $D = \alpha N^2 + [\text{όροι μικρότερης τάξης από } N^2]$ όπου ούτε η παράμετρος α ούτε οι όροι μικρότερης τάξης από N^2 είναι γνωστοί.

10.2.1 Η μέθοδος Cross-Validation

Ένας τρόπος για να βελτιώσουμε την ικανότητα γενίκευσης μιας μηχανής μάθησης είναι να εκπαιδεύσουμε πολλά μοντέλα πάνω σε διαφορετικές οικογένειες συναρτήσεων, να εκτιμήσουμε την ικανότητα γενίκευσης της τελικής συναρτησης που έμαθε κάθε μοντέλο, και τέλος να επιλέξουμε εκείνη την συναρτηση-μοντέλο που επιτυγχάνει την καλύτερη γενίκευση. Επειδή στην πράξη διαθέτουμε συνήθως ένα πεπερασμένο και ενδεχομένως μικρό σύνολο προτύπων τίθεται το ερώτημα της εκτίμησης της ικανότητας γενίκευσης ενός μοντέλου χρησιμοποιώντας πεπερασμένα δείγματα.

Η μέθοδος cross-validation [233] κάνει ακριβώς αυτό. Έστω ότι έχουμε επιλέξει το μοντέλο που θα εκπαιδευτεί (πχ. ένα δίκτυο MLP δύο στρωμάτων με $n-N_h-m = [\text{νευρώνες εισόδου}]-[\text{κρυφούς νευρώνες}]-[\text{νευρώνες εξόδου}]$). Έστω, επίσης, ότι διαθέτουμε ένα συγκεκριμένο πλήθος από ζεύγη προτύπων-στόχων: $(x^{(i)}, d^{(i)})$, $i=1, \dots, P$. Για να εκτιμήσουμε την ικανότητα γενίκευσης του μοντέλου μας, το αφήνουμε να εκπαιδευτεί αφήνοντας έξω από την εκπαίδευση ένα ποσοστό S των προτύπων. Για παράδειγμα, αν επιλέξουμε $S = 0.1$, ενώ διαθέτουμε $P=1000$ πρότυπα, θα εκπαιδεύσουμε το δίκτυο χρησιμοποιώντας 900 από αυτά και θα κρατήσουμε τα υπόλοιπα 100 για έλεγχο του δικτύου μετά από την εκπαίδευση. Το σύνολο των προτύπων που χρησιμοποιούνται στην εκπαίδευση καλείται σύνολο εκπαίδευσης (training set) ενώ το σύνολο που χρησιμοποιείται για τον έλεγχο της ικανότητας γενίκευσης λέγεται σύνολο ελέγχου (test set). Ας ονομάσουμε I_{train} και I_{test} το σύνολο των δεικτών i των προτύπων που χρησιμοποιούνται στην εκπαίδευση και στον έλεγχο, αντίστοιχα. Αν εκπαιδεύσουμε το δίκτυο χρησιμοποιώντας τα ζεύγη $(x^{(1)}, d^{(1)})$, ..., $(x^{(900)}, d^{(900)})$, κρατώντας τα ζεύγη $(x^{(901)}, d^{(901)})$, ..., $(x^{(1000)}, d^{(1000)})$ για έλεγχο, τότε $I_{train} = \{1, 2, \dots, 900\}$ και $I_{test} = \{901, 902, \dots, 1000\}$.

Με την ολοκλήρωση της εκπαίδευσης, το μοντέλο μαθαίνει κάποια συνάρτηση $f(x; \theta)$, όπου θ είναι το διάνυσμα των παραμέτρων του μοντέλου. Με βάση τον παραπάνω διαχωρισμό των προτύπων, μπορούμε να μιλάμε για δύο είδη σφαλμάτων σχετικών με την $f(x; \theta)$:

(a) το σφάλμα εκπαίδευσης:

$$J_{train} = \frac{1}{|I_{train}|} \sum_{i \in I_{train}} \|d^{(i)} - f(x^{(i)}; \theta)\|^2 \quad (229)$$

(β) και το σφάλμα ελέγχου:

$$J_{test} = \frac{1}{|I_{test}|} \sum_{i \in I_{test}} \left\| \mathbf{d}^{(i)} - f(\mathbf{x}^{(i)}; \theta) \right\|^2. \quad (230)$$

Το πρώτο σφάλμα δείχνει πόσο καλά το μοντέλο προσεγγίζει τα πρότυπα με τα οποία εκπαιδεύτηκε, ενώ το δεύτερο σφάλμα δείχνει πόσο καλά γενικεύει επάνω στα συγκεκριμένα πρότυπα ελέγχου τα οποία κρατήθηκαν κρυφά κατά την εκπαίδευση.

Επειδή ο διαχωρισμός των προτύπων σε πρότυπα εκπαίδευσης και ελέγχου είναι αυθαίρετη, η εκτίμηση της ικανότητας γενίκευσης του μοντέλου με βάση το σφάλμα J_{test} ενέχει αρκετό ρίσκο. Αν επιλέγαμε άλλο σύνολο εκπαίδευσης και συνεπώς, άλλο σύνολο ελέγχου, πιθανώς να πάρναμε τελείως διαφορετικά σφάλματα J_{train} και J_{test} . Έτσι για να πάρουμε καλύτερη εκτίμηση της γενίκευσης του μοντέλου χωρίζουμε το σύνολο προτύπων σε N ομάδες με S πρότυπα·η κάθε μία, και εκπαίδευσυμε το μοντέλο N φορές. Δεν είναι δύσκολο να δούμε ότι $N=1/S$. Για παράδειγμα, αν $S=0.05$, τότε $N=20$, ενώ αν $S=0.1$, τότε $N=10$, κλπ. Στην ακραία περίττωση, γνωστή ως μέθοδος *leave-one-out*, έχουμε $S=1/P$, οπότε κάθε ομάδα έχει ένα μόνο πρότυπο ενώ έχουμε συνολικά $N=P$ ομάδες. Σε κάθε περίπτωση, την i -στή φορά που εκπαίδευσυμε το δίκτυο θέτουμε την i -στή ομάδα ως σύνολο ελέγχου. Αφού υπολογίσουμε τα αντίστοιχα κόστη $J_{train}(i)$ και $J_{test}(i)$, εκτιμάμε την ικανότητα γενίκευσης του μοντέλου από το μέσο σφάλμα ελέγχου:

$$\bar{J}_{test} = \frac{1}{N} \sum_{i=1}^N J_{test}(i). \quad (231)$$

Ο Αλγόριθμος 22 περιγράφει συνολικά την μέθοδο Cross-Validation. Το πλεονέκτημα της μεθόδου είναι η καλή εκτίμηση της ικανότητας γενίκευσης του μοντέλου που εξετάζουμε. Το μη-αμελητέο μειονέκτημα της μεθόδου είναι ο πολύ μεγάλος υπολογιστικός χρόνος που απαιτείται ο οποίος είναι ανάλογος του N . Αν και τυπικές τιμές είναι κοντά στο 10, ο χρόνος αυτός μπορεί, σε ορισμένες περιπτώσεις, να είναι απαγορευτικός.

10.2.2 Η μέθοδος κανονικοποίησης

Αλγόριθμος 22: Cross-Validation

Είσοδος:

P πρότυπα $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(P)}$, με στόχους $\mathbf{d}^{(1)}, \dots, \mathbf{d}^{(P)}$

Ένα συγκεκριμένο νευρωνικό μοντέλο

Εξόδος:

Εκτίμηση \bar{J}_{test} της ικανότητας γενίκευσης του μοντέλου

Μέθοδος:

Χώρισε τα πρότυπα σε N ίσες ομάδες, G_1, G_2, \dots, G_N

Για κάθε επανάληψη $i=1, \dots, N$ {

Εκπαίδευσε το μοντέλο χρησιμοποιώντας τα πρότυπα από όλες τις ομάδες εκτός από την G_i

Υπολόγισε το σφάλμα ελέγχου $J_{test}(i)$ σύμφωνα με την Εξ. (230) για τα πρότυπα της ομάδας G_i

}

Υπολόγισε το \bar{J}_{test} από το μέσο όρο των $J_{test}(i)$ (Εξ. (231)).

Η θεωρία της κανονικοποίησης (regularization theory) προτάθηκε το 1963 από τον Tikhonov για την επίλυση κακώς ορισμένων προβλημάτων (*ill-posed problems*) (βλ. [218]). Ένα πρόβλημα καλείται κακώς ορισμένο αν συμβαίνει ένα από τα παρακάτω:

- (α) το πρόβλημα δεν έχει λύση,
- (β) έχει περισσότερες από μια λύσεις, ενδεχομένως άπειρες το πλήθος, ή
- (γ) έχει λύση αλλά η λύση αλλάζει πάρα πολύ με μικρές αλλαγές των παραμέτρων του προβλήματος.

Χαρακτηριστικό παράδειγμα είναι η εκτίμηση μιας κρυμμένης καμπύλης μέσα από πεπερασμένο πλήθος δειγμάτων. Όπως είδαμε και προηγουμένως, υπάρχουν άπειρες λύσεις στο πρόβλημα αφού υπάρχουν άπειρες καμπύλες

που περνούν από τα δείγματα αυτά. Ποια λοιπόν από όλες αυτές είναι η καμπύλη που ζητάμε;

Η προσέγγιση που προτείνεται από την μέθοδο της κανονικοποίησης είναι η μετατροπή της συνάρτησης κόστους J σε μια νέα συνάρτηση κόστους με την εισαγωγή ενός επί πλέον όρου λJ_{reg} ο οποίος τιμωρεί λύσεις με μεγάλες τιμές παραγώγων, δηλαδή λύσεις με μεγάλη καμπυλότητα (όχι ομαλές). Για παράδειγμα, έστω ότι έχουμε ένα δίκτυο MLP που περιγράφεται από την συνάρτηση $y=f(\mathbf{x}; \mathbf{w})$ όπου \mathbf{x} είναι η είσοδος και \mathbf{w} είναι το συνολικό διάνυσμα των παραμέτρων (τα συναπτικά βάρη όλων των στρωμάτων). Επιθυμούμε να εκπαιδεύσουμε το δίκτυο χρησιμοποιώντας P πρότυπα εισόδου $\mathbf{x}^{(p)}$ και τους αντίστοιχους στόχους $d^{(p)}$, $p=1, \dots, P$. Η συνάρτηση κόστους, χωρίς κανονικοποίηση, είναι το γνωστό μας μέσο τετραγωνικό σφάλμα

$$J_{mse}(\mathbf{w}) = \frac{1}{P} \sum_{p=1}^P [d^{(p)} - f(\mathbf{x}^{(p)}; \mathbf{w})]^2.$$

Η ελαχιστοποίηση του J χρησιμοποιώντας, πχ., τον αλγόριθμο Back-Propagation θα μας δώσει διαφορετικές λύσεις ανάλογα με το σημείο εκκίνησης του αλγορίθμου.

Η κανονικοποιημένη συνάρτηση κόστους

$$J_{new}(\mathbf{w}) = J_{mse}(\mathbf{w}) + \lambda J_{reg}(\mathbf{w}) \quad (232)$$

περιέχει τον όρο της κανονικοποίησης J_{reg} ο οποίος κλιμακώνεται με μια σταθερά λ . Η σταθερά αυτή, γνωστή ως **παράμετρος κανονικοποίησης (regularization parameter)** ζυγίζει την επιρροή του όρου κανονικοποίησης στο συνολικό κόστος. Τώρα τιμωρούνται λύσεις με μεγάλες τιμές του J_{reg} και συνεπώς περιορίζεται το πλήθος των πιθανών λύσεων, οπότε το πρόβλημα γίνεται καλύτερα ορισμένο από πριν. Η ελαχιστοποίηση του J_{new} μπορεί να γίνει με οποιαδήποτε μέθοδο, πχ., την κατάβαση δυναμικού, αρκεί κάποιος να υπολογίσει τις αντίστοιχες παραγώγους.

Ένα παράδειγμα του όρου κανονικοποίησης είναι

$$J_{reg} = \frac{1}{2} \sum_{p=1}^P \left(f_i''(\mathbf{x}^{(p)}; \mathbf{w}) \right)^2,$$

$$f_i''(\mathbf{x}; \mathbf{w}) = \frac{\partial^2 f(\mathbf{x}; \mathbf{w})}{\partial x_i^2}.$$

Κεφάλαιο 10: Μάθηση και Γενίκευση

Ο συγκεκριμένος όρος τιμωρεί συναρτήσεις f με μεγάλη δεύτερη παράγωγο. Καθώς μόνο οι επίπεδες επιφάνειες έχουν μηδενική δεύτερη παράγωγο, γινέται, τις επίπεδες επιφάνειες.

Ένα θεμελιώδες αποτέλεσμα της θεωρίας κανονικοποίησης είναι ότι αν ο όρος της κανονικοποίησης είναι της μορφής

$$J_{reg} = \frac{1}{2} \int \|Df(\mathbf{x}; \mathbf{w})\|^2 d\mathbf{x}$$

όπου D είναι κάποιος γραμμικός διαφορικός τελεστής, τότε η βέλτιστη συνάρτηση f είναι της μορφής

$$f(\mathbf{x}; \mathbf{w}) = \sum_{p=1}^P w_p G(\|\mathbf{x} - \mathbf{x}^{(p)}\|) \quad (233)$$

όπου η συνάρτηση G καλείται **συνάρτηση Green**. Οι παράμετροι w_p βρίσκονται αν λύσουμε το παρακάτω σύστημα γραμμικών εξισώσεων για $r=1, \dots, P$:

$$\sum_{p=1}^P G(\|\mathbf{x}^{(r)} - \mathbf{x}^{(p)}\|) w_p + \lambda w_r = d^{(r)}. \quad (234)$$

Το ενδιαφέρον είναι ότι η συνάρτηση Green είναι τύπου ακτινικής βάσης και συνεπώς η (233) περιγράφει την έξοδο ενός δικτύου RBF με p κρυφούς νευρώνες τύπου Green και με συναπτικά βάρη εξωτερικού στρώματος τις τιμές w_p . Μάλιστα για μια συγκεκριμένη τιμή του τελεστή D η συνάρτηση Green δεν είναι άλλη από την γνωστή μας Γκαουσιανή συνάρτηση $G(x) = \exp\{-x^2/(2\sigma^2)\}$. Αυτή την σχέση μεταξύ κανονικοποίησης και δικτύων RBF υπέδειξαν πρώτοι οι Poggio και Girosi [190].

Το βασικό πρόβλημα της μεθόδου είναι η κατάλληλη επιλογή της παραμέτρου κανονικοποίησης. Αν επιλέξουμε πολύ μεγάλη τιμή για το λ τότε το δίκτυο δίνει πολύ μεγάλη έμφαση στην ομαλότητα της λύσης παραβλέποντας ενδεχομένως πόσο καλά ταιριάζει η λύση αυτή στα δεδομένα. Αυτό είναι αντίστοιχο με το σφάλμα υπο-μοντελοποίησης που περιγράψαμε στην αρχή του κεφαλαίου. Αν, αντίθετα, επιλέξουμε πολύ μικρή τιμή για το λ , τότε θα έχουμε πολύ καλή εφαρμογή στα δεδομένα αλλά κινδυνεύουμε η καμπύλη μας να είναι πολύ «κατσαρή» οπότε να έχουμε φτωχή γενίκευση. Αυτό είναι αντίστοιχο με το σφάλμα υπερ-μοντελοποίησης. Η κατάλληλη επιλογή της παραμέτρου γίνεται συνήθως με τη μέθοδο της δοκιμής και

σφάλματος και εξαρτάται πολύ από το συγκεκριμένο πρόβλημα, από το σύνολο των δεδομένων που διαθέτουμε, και από τον συγκεκριμένο όρο κανονικοποίησης που χρησιμοποιούμε.

10.2.3 Εξασθένιση βαρών (Weight decay)

Η μέθοδος της εξασθένισης των βαρών [88,141] εφαρμόζεται σε δίκτυα MLP. Μοιάζει με την μέθοδο της κανονικοποίησης στο γεγονός ότι εισάγεται ένας έξτρα όρος στην συνάρτηση κόστους ο οποίος είναι της μορφής

$$J_e = \frac{1}{2} \sum_{i,j} w_{i,j}^2$$

όπου $w_{i,j}$ είναι τα συναπτικά βάρη του δικτύου. Ο όρος αυτός «τιμωρεί» τα συναπτικά βάρη με μεγάλη απόλυτη τιμή όποτε τα βάρη πιέζονται προς το μηδέν. Η λογική είναι ότι αν η απεικόνιση εισόδου-εξόδου που επιδιώκουμε να εκπαιδεύσουμε απαιτεί κάποια βάρη να είναι ισχυρά τότε αυτά θα επιζήσουν. Τα βάρη που θα εξασθενήσουν προς το μηδέν και ίσως θα εξαφανιστούν θα είναι τα «περιττά» βάρη, αντά δηλαδή που δεν συνεισφέρουν στο τελικό αποτέλεσμα. Η νέα συνάρτηση κόστους που ελαχιστοποιείται είναι

$$J_{wd} = J_{mse} + \lambda J_e.$$

Είναι σχετικά εύκολο να υπολογίσουμε τον κανόνα ενημέρωσης των βαρών Back-Propagation υπολογίζοντας την παράγωγο

$$\frac{\partial J_{wd}}{\partial w_{ij}} = \frac{\partial J_{mse}}{\partial w_{ij}} + \lambda w_{ij}$$

$$\Delta w_{ij} = -\beta \frac{\partial J_{mse}}{\partial w_{ij}} - \beta \lambda w_{ij}$$

εργάσοντας το weight

decay μπορεί να ακολουθήσει

weight elimination (ή/και node elimination).

Η ελάττωση των βαρών μειώνει το πλήθος των ελεύθερων παραμέτρων του δικτύου. Αυτό δημιουργεί ένα απλούστερο δίκτυο – τόσο απλό όσο απαιτεί η εφαρμογή. Με άλλα λόγια το δίκτυο λύνει αυτόματα το πρόβλημα της υπερ-ή υπο-μοντελοποίησης: αυτοοργανώνεται ώστε να κόψει ακριβώς τα βάρη που δεν χρειάζονται.

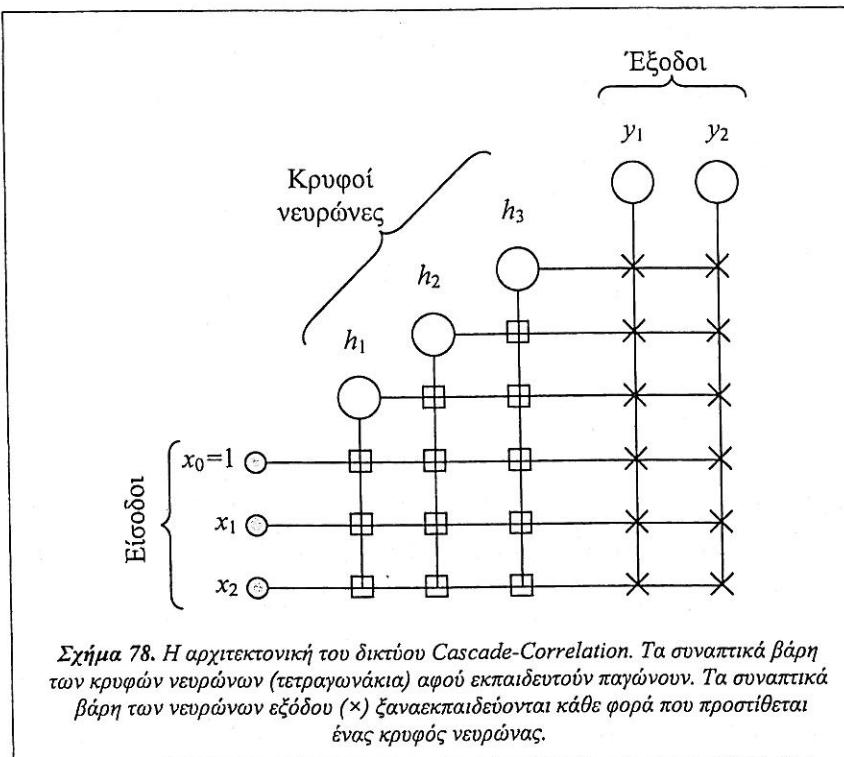
10.2.4 Επαύξηση (Growing)

Οι μέθοδοι επαύξησης εκκινούν από ένα μικρό δίκτυο και προσθέτουν σταδιακά νευρώνες έτσι ώστε να επιτευχθεί η κατάλληλη πολυπλοκότητα του δικτύου. Για παράδειγμα, μπορούμε να εκκινήσουμε με ένα δίκτυο MLP δύο στρωμάτων με M κρυφούς νευρώνες. Αφού το εκπαιδεύσουμε πάνω σε επάνω σε ένα σε δεδομένων επολογίζουμε το σφάλμα γενίκευσης J_{test} νοποιητικό τότε σταματάμε. Αν το σφάλμα γενίκευσης είναι ικανοποιητικό τότε σταματάμε. Αν όχι, προσθέτουμε έναν ή περισσότερους σφάλμα γενίκευσης καθώς αυξάνουμε τους κρυφούς νευρώνες συνεχίζουμε, αλλιώς σταματάμε.

Το δίκτυο Cascade correlation

Ένας λόγος που τα νευρωνικά μοντέλα, όπως το back-propagation, είναι τόσο αργά στην εκπαίδευση είναι το λεγόμενο πρόβλημα των κινούμενου στόχου: επειδή τα βάρη του δικτύου αλλάζουν ταυτόχρονα, κάθε κρυφός νευρώνας βλέπει ένα συνεχώς μεταβαλλόμενο περιβάλλον. Έτσι, αντί τα βάρη να αναλαμβάνουν γρήγορα τις τιμές που πρέπει, πολὺς χρόνος χάνεται σε άσκοπες κινήσεις.

Το μοντέλο *Cascade Correlation* [53] προσπαθεί να λύσει το παραπάνω πρόβλημα εφαρμόζοντας την ιδέα της επαύξησης. Αρχικά δεν περιέχει καθείς μέσα από συναπτικά βάρη ("X" στο Σχ. 78). Οι νευρώνες εμπεριέχουν την σιγμοειδή συνάρτηση. Τα βάρη μπορούν να εκπαιδευτούν με οποιοδήποτε αλγόριθμο κατάλληλο για ένα δίκτυο με ένα στρώμα. Για παράδειγμα, προστίθεται ένας κρυφός νευρώνας και εκπαιδεύονται τα βάρη τόσο του κρυφού νευρώνα όσο και των νευρώνων εξόδου. Συνεχίζουμε να προσθέτουμε κρυφούς νευρώνες μέχρι η επίδοση του δικτύου, δηλαδή το σφάλμα, να πέσει σε αποδεκτά επίπεδα. Κάθε κρυφός νευρώνας δέχεται είσοδο από τις εισόδους του δικτύου και από τους κρυφούς νευρώνες που προστέθηκαν πριν από αυτόν, δημιουργώντας έτσι την διάταξη των κρυφών νευρώνων (*cascade*) που φαίνεται στο Σχήμα 78.



Τα βάρη του κρυφού νευρώνα που μόλις προσθέσαμε εκπαιδεύονται έτσι ώστε να μεγιστοποιούν την συσχέτιση S μεταξύ της εξόδου h του νευρώνα αυτού

$$h = f\left(\sum_{i=0}^n w_i x_i\right), \quad f = \text{συγμοειδής συνάρτηση}$$

και του σφάλματος e_j των εξόδων $j=1, \dots, m$

$$e_j = (y_j - d_j)$$

Η συσχέτιση υπολογίζεται αθροίζοντας για όλες τις εξόδους j και για όλα τα πρότυπα εκπαίδευσης $p=1, \dots, P$:

Κεφάλαιο 10: Μάθηση και Γενίκευση

$$S = \sum_{j=1}^m |v_j|$$

$$v_j = \sum_{p=1}^P (h^{(p)} - \bar{h})(e_j^{(p)} - \bar{e}_j)$$

Οι ποσότητες \bar{h} και \bar{e}_j συμβολίζουν τις μέσες τιμές των $h^{(p)}$ και $e_j^{(p)}$ για όλα τα πρότυπα. Το σκεπτικό πίσω από την χρήση της συσχέτισης είναι ότι ένας κρυφός νευρώνας που δεν είναι συσχετισμένος με το σφάλμα εξόδου θα χρησιμεύσει ελάχιστα στην μείωση του σφάλματος αυτού. Εάντοτε, καθώς την μέγιστη συσχέτιση ελπίζοντας ότι μετά την εκπαίδευση των εξωτερικών νευρώνων θα επιτύχουμε τη μέγιστη δυνατή βελτίωση του σφάλματος.

Για την εκπαίδευση των βαρών χρησιμοποιείται η μέθοδος της ανάβασης δυναμικού και ο αλγόριθμος Quickeprop όπου γίνεται χρήση της παραγώγου

$$\frac{\partial S}{\partial w_i} = \sum_{p,j} \sigma_j(e_j^{(p)} - \bar{e}_j) f' x_i^{(p)}$$

Η ποσότητα σ_j είναι το πρόσημο του v_j , f' είναι η παράγωγος της σιγμοειδούς συνάρτησης και $x_i^{(p)}$ είναι η i -στή είσοδος για το πρότυπο p . Αφού εκπαιδεύονται τα βάρη ενός κρυφού νευρώνα, στη συνέχεια παγώνουν και δεν θα εκπαιδεύονται στο μέλλον ξανά. Μόνο τα βάρη του στρώματος εξόδου ξαναεκπαιδεύεται όταν προστίθεται ένας κρυφός νευρώνας. Έτσι η εκπαίδευση είναι ταχύτατη αφού κάθε φορά εκπαιδεύεται ένα μόνο στρώμα.

Παράδειγμα 30. Το πρόβλημα “parity”.

Οι Fahlman και Lebiere [53] εφάρμοσαν τον αλγόριθμο Cascade Correlation στο κλασικό πρόβλημα “parity”. Η συνάρτηση parity δέχεται δυαδικές εισόδους και παράγει έξοδο 1 αν το πλήθος των εισόδων που είναι ίσες με 1 είναι περιττό, ενώ η έξοδος είναι 0 αν το πλήθος των 1 στην είσοδο είναι άρτιο. Για $n=2$, η συνάρτηση parity ταυτίζεται με την συνάρτηση XOR. Επιδιώκουμε το δίκτυο να μάθει αυτήν την συνάρτηση.

Τα αποτελέσματα των πειραμάτων τους για διάφορα πλήθη εισόδων συνοψίζονται στον παρακάτω πίνακα. Έγιναν πέντε επαναλήψεις του αλγορίθμου για κάθε περίττωση ώστε να εξαχθούν στατιστικά συμπεράσματα. Τα αποτελέσματα μπορούν να συγκριθούν με τα αντίστοιχα χρησιμοποιώντας τον απλό αλγόριθμο Back-Propagation. Για παρά-

δειγμα, για $n=8$, ο αλγόριθμος BP απαιτεί περίπου 2000 εποχές και 16 κρυφούς νευρώνες. Η υπεροχή του Cascade Correlation τόσο σε πλήθος εποχών όσο και πλήθος κρυφών νευρώνων είναι προφανής.

n	Πρότυπα $P=2^n$	Κρυφοί νευρώνες	Μέσο πλήθος εποχών
2	4	1	24
3	8	1	32
4	16	2	66
5	32	2-3	142
6	64	3	161
7	128	4-5	192
8	256	4-5	357

Επιπλέον, επειδή το μοντέλο CC βρίσκει από μόνο του το απαραίτητο πλήθος των κρυφών νευρώνων και άρα επιλέγει την πολυπλοκότητα του δικτύου, γενικεύει καλά. Οι ίδιοι ερευνητές έκαναν πειράματα για το ίδιο πρόβλημα με $n=10$ εισόδους (και άρα $P=1024$ πρότυπα) όπου ένα ποσοστό μόνο των προτύπων χρησιμοποιήθηκε στην εκπαίδευση ενώ το υπόλοιπο χρησιμοποιήθηκε για να γίνει έλεγχος της ικανότητας γενίκευσης. Βρέθηκε ότι όταν εκπαιδεύονταν με τα μισά πρότυπα η επιτυχία στην γενίκευση ανερχόταν στο 96% ενώ η εκπαίδευση μόνο στα 25% των προτύπων οδηγούσε σε γενίκευση 90%. Το πλήθος των κρυφών νευρώνων κυμαίνοταν από 4 έως 7 ενώ οι εποχές από 276 ως 551.

10.2.5 Κλάδεμα (Pruning)

Το κλάδεμα ακολουθεί την αντίστροφη διαδικασία από την επαύξηση: ξεκινάμε από ένα μεγάλο δίκτυο και το μειώνουμε είτε αφαιρώντας μη απαραίτητους νευρώνες είτε μη απαραίτητες συνάψεις έτσι ώστε να πετύχουμε την κατάλληλη πολυπλοκότητα του δικτύου.

Κλάδεμα νευρώνων

Οι Mozer και Smolenksy [171] πρότειναν έναν αλγόριθμο αφαίρεσης νευρώνων σε ένα δίκτυο MLP βασισμένο στον υπολογισμό της **σημαντικότητας (saliency)** του κάθε νευρώνα. Για να μετρηθεί η σημαντικότητα ενός νευρώνα υπολογίζεται η διαφορά του σφάλματος όταν το δίκτυο δεν περιέχει τον νευρώνα αυτό μείον το σφάλμα όταν τον περιέχει

$$\rho_i = J(\text{χωρίς τον νευρώνα } i) - J(\text{με τον νευρώνα } i).$$

Μετά από κάποιες εποχές εκπαίδευσης αφαιρούμε τους νευρώνες με τις μικρότερες σημαντικότητες και συνεχίζουμε επαναλαμβάνοντας τη διαδικασία εκπαίδευσης – αφαίρεσης νευρώνων τόσες φορές όσες χρειάζονται για να επιτευχθεί το επιθυμητό σφάλμα. Το πρόβλημα με τη μέθοδο αυτή είναι ο πολύ μεγάλος υπολογιστικός φόρτος αφού πρέπει να κάνουμε ανάκληση του δικτύου 2 φορές για όλα τα πρότυπα και αυτό πρέπει να επαναληφθεί για όλους τους νευρώνες i .

Μια εναλλακτική πρόταση είναι να αντιστοιχίσουμε μια έξτρα μεταβλητή γ στον νευρώνα i έτσι ώστε η έξοδος του να είναι 0 όταν $\gamma=0$, ενώ θα μένει ανεπτρέαστη όταν $\gamma=1$

$$a_i(l) = f\left(\gamma_i \sum_{j=0}^{N(l-1)} w_{ij} a_j(l-1)\right).$$

Τώρα η σημαντικότητα είναι

$$\rho_i = J(\gamma=0) - J(\gamma=1)$$

και μπορεί να προσεγγιστεί από την παράγωγο

$$\tilde{\rho}_i = -\frac{\partial J}{\partial \gamma_i} \Big|_{\gamma_i=1}.$$

Για τον υπολογισμό της παραγώγου μπορεί να χρησιμοποιηθεί μια επέκταση του αλγορίθμου Back-Propagation. Το νέο μοντέλο ονομάστηκε δίκτυο **σκελετός (skeleton network)**. Στην πράξη επειδή η παράγωγος μεταβάλλεται απότομα από επανάληψη σε επανάληψη, χρησιμοποιείται ο εξομαλυντικός κανόνας

$$\tilde{\rho}_i(k+1) = 0.8 \tilde{\rho}_i(k) - 0.2 \frac{\partial J(k)}{\partial \gamma_i}.$$

Παράδειγμα 31. Το πρόβλημα του πολυπλέκτη.

Οι Mozer και Smolensky [171] εφάρμοσαν το μοντέλο τους σε διάφορες εφαρμογές όπως λογικούς κανόνες και προβλήματα βελτίωσης της μάθησης. Χαρακτηριστικό παράδειγμα αποτελεί το πρόβλημα της πολυπλέξης (*multiplexing*) 4 δυαδικών μεταβλητών A, B, C, D , χρησιμοποιώντας δύο δυαδικές μεταβλητές ελέγχου M_1, M_2 . Το δίκτυο έχει 6 εισόδους $\{A, B, C, D, M_1, M_2\}$ και μια έξοδο Y . Καλείται να βγάλει στην έξοδο

- $Y=A$ αν $(M_1, M_2)=(0,0)$
- $Y=B$ αν $(M_1, M_2)=(0,1)$
- $Y=C$ αν $(M_1, M_2)=(1,0)$
- $Y=D$ αν $(M_1, M_2)=(1,1)$

δηλαδή καλείται να υλοποιήσει την δυαδική συνάρτηση

$$Y = \overline{M_1} \overline{M_2} A + \overline{M_1} M_2 B + M_1 \overline{M_2} C + M_1 M_2 D.$$

Έγινε σύγκριση μεταξύ ενός απλού δίκτυου Back-Propagation με 4 κρυφούς νευρώνες και ενός δίκτυου σκελετού το οποίο ξεκινούσε με 8 κρυφούς νευρώνες που σταδιακά μειώνονταν στους 4. Έγιναν 100 πειράματα για συλλογή στατιστικών συμπερασμάτων. Το απλό δίκτυο δεν συνέκλινε στο 17% των περιπτώσεων ενώ το δίκτυο σκελετού συνέκλινε και τις 100 φορές. Και στις δύο περιπτώσει χρησιμοποιήθηκε το ίδιο κριτήριο τερματισμού. Το μέσο πλήθος εποχών για τη σύγκλιση του δίκτυου σκελετού με 8 κρυφούς νευρώνες ήταν 25 ενώ αντίστοιχα για το απλό δίκτυο ήταν 52 εποχές. Γενικά παρατηρήθηκε επιτάχυνση της σύγκλισης ενώ το κλάδεμα των περιττών νευρώνων δεν μείωσε την επίδοση του συστήματος.

Κλάδεμα βαρών (Optimal Brain Damage)

Η έννοια της σημαντικότητας των νευρώνων μπορεί να μεταφερθεί και στα συναπτικά βάρη. Το μέτρο της σημαντικότητας ενός βάρους w_{ij} θα μπορούσε να ήταν το πόσο επηρεάζεται το κόστος J από την μεταβολή του w_{ij} . Ο Le Cun, και συνεργάτες [149] παίρνουν ως μέτρο την δεύτερη παράγωγο του κόστους πολλαπλασιασμένη επί το τετράγωνο του βάρους

Κεφάλαιο 10: Μάθηση και Γενίκευση

$$\rho_{ij} = \frac{1}{2} H_{ij} w_{ij}^2,$$

$$H_{ij} = \frac{\partial^2 J}{(\partial w_{ij})^2}.$$

Η μέθοδος αυτή ονομάζεται **βέλτιστη εγκεφαλική βλάβη** (*optimal brain damage*). Ο υπολογισμός της δεύτερης παραγώγου μοιάζει πολύ με τον αλγόριθμο Back-Propagation. Ο αναγνώστης μπορεί να βρει λεπτομέρειες στο [149]. Ο Αλγόριθμος 23 περιγράφει τη μέθοδο.

Μια παραλλαγή της μεθόδου είναι ο **βέλτιστος εγκεφαλικός χειρούργος** (*optimal brain surgeon*) [76] όπου χρησιμοποιείται ολόκληρος ο πίνακας $H=[h_{ij}]$ της δεύτερης παραγώγου γνωστός ως **πίνακας του Hess** (Hessian Matrix):

$$h_{kl} = \frac{\partial^2 J}{\partial \omega_k \partial \omega_l}$$

όπου οι ποσότητες $\omega_1, \omega_2, \dots$, είναι τα βάρη w_{ij} , ταξινομημένα σε μονοδιάστατη σειρά. Η μέθοδος αναπτύσσεται λεπτομερώς στο [79, Κεφ. 4].

Αλγόριθμος 23: Optimal brain damage

Είσοδοι: P πρότυπα εκπαίδευσης με στόχους. Ένα μεγάλο αρχικό δίκτυο MLP.

Εξοδοί: Ένα μικρότερο δίκτυο MLP με αρκετά βάρη μηδενισμένα

Μέθοδος:

Βήμα 1: Επέλεξε ένα σχετικά μεγάλο αρχικό δίκτυο MLP

Βήμα 2: Εκπαίδευσε το δίκτυο μέχρι να ικανοποιηθεί κάποιο κριτήριο τερματισμού

Βήμα 3: Υπολόγισε τις δεύτερες παραγώγους H_{ij} και από αυτές τις σημαντικότερες ρ_{ij} .

Βήμα 4: Ταξινόμησε τις σημαντικότητες κατά φθίνουσα σειρά και σβήσε κάποια από τα βάρη με τη χαμηλότερη σημαντικότητα

Βήμα 5: Πήγαινε στο Βήμα 2 και επανέλαβε μέχρι να ικανοποιηθεί ένα ολικό κριτήριο τερματισμού.

10.2.6 Επιτροπές (Committee Machines)

Όπως είδαμε στην ενότητα 10.2.1, μια συνηθισμένη προσέγγιση για την βελτίωση της ικανότητας γενίκευσης του τελικού εκτιμητή είναι να εκπαιδεύσουμε μια ομάδα με N διαφορετικά μοντέλα χρησιμοποιώντας τα ίδια δεδομένα εκπαίδευσης και κατόπιν να επιλέξουμε εκείνο το μοντέλο που γενικεύει καλύτερα. Το βασικό πρόβλημα της προσέγγισης αυτής είναι ότι σπαταλιέται πολύς υπολογιστικός χρόνος αφού από τα N μοντέλα μόνο ένα θα χρησιμοποιηθεί τελικά. Μια εναλλακτική προσέγγιση [185] είναι να χρησιμοποιήσουμε τα αποτελέσματα όλων των μοντέλων συνδυάζοντας τις μεμονωμένες εκτιμήσεις $f_i(\mathbf{x}; \theta_i)$ του κάθε μοντέλου σε ένα μοναδικό εκτιμητή

$$F(\mathbf{x}) = \sum_{i=1}^N c_i f_i(\mathbf{x}; \theta_i). \quad (235)$$

Ελπίζουμε ότι με τον κατάλληλο συνδυασμό τους θα επιτύχουμε καλύτερη τελική εκτίμηση σε σχέση με τους ξεχωριστούς μεμονωμένους εκτιμητές. Η ομάδα των μοντέλων καλείται **επιτροπή** (committee). Η ιδέα μπορεί να εφαρμοστεί γενικότερα με κάθε είδους μοντέλα μάθησης και φυσικά με νευρωνικά δίκτυα.

Υπάρχουν εν γένει, δύο ειδών προβλήματα που μπορούν να αντιμετωπίσουν με τη χρήση επιτροπών. Το πρώτο είναι η προσέγγιση μιας άγνωστης καμπύλης χρησιμοποιώντας πεπερασμένο πλήθος δειγμάτων. Το πρόβλημα αυτό είναι γνωστό στη στατιστική ως παλινδρόμηση (regression). Στην περίπτωση αυτή οι στόχοι είναι πραγματικοί αριθμοί. Το δεύτερο πρόβλημα είναι η ταξινόμηση των προτύπων, συνήθως σε δύο κλάσεις. Στην περίπτωση αυτή οι στόχοι είναι δυαδικοί αριθμοί, πχ. -1/1.

Παράδειγμα 32.

Έστω ότι έχουμε ένα σύνολο προτύπων εκπαίδευσης $\mathbf{x}^{(i)}$, $i=1, \dots, P$, που ανήκουν σε δύο κλάσεις οπότε οι σχετικοί στόχοι είναι $d^{(i)}=-1$ ή 1. Χρησιμοποιούμε το σύνολο αυτό για να εκπαίδευσουμε τρία διαφορετικά νευρωνικά δίκτυα, ένα δίκτυο Perceptron, ένα δίκτυο MLP, και ένα δίκτυο RBF, παίρνοντας τις συναρτήσεις f_{PER} , f_{MLP} , f_{RBF} , που υλοποιούνται από τα μοντέλα αυτά. Το δίκτυο Perceptron χρησιμοποιεί τη σκληρή βηματική συνάρτηση -1/1, ενώ τα δίκτυα MLP και RBF δίνουν «μαλακή» έξοδο μεταξύ -1 και 1. Εισάγοντας ένα άγνωστο πρότυπο \mathbf{x} , παίρνουμε τις εξής αποκρίσεις:

Κεφάλαιο 10: Μάθηση και Γενίκευση

$$f_{\text{PER}}(\mathbf{x}) = -1, f_2(\mathbf{x}) = -0.2, f_3(\mathbf{x}) = +0.6.$$

Τα δύο από τα τρία μοντέλα (Perceptron και MLP) αποφασίζουν ότι το πρότυπο ανήκει στην κλάση -1, αν και το μοντέλο MLP δίνει χλιαρή επιμηγορία. Το τρίτο μοντέλο (RBF) αποφασίζει ότι το πρότυπο ανήκει στην κλάση 1, με αρκετή, αλλά όχι πολύ μεγάλη, βεβαιότητα. Η απόφαση της επιτροπής θα είναι ένας γραμμικός συνδυασμός των τριών αποφάσεων. Αν δεν έχουμε λόγο να πιστεύουμε ότι κάποιο μοντέλο είναι καλύτερος εκτιμητής από τα υπόλοιπα, τότε ένας απλός συνδυασμός τους είναι να χρησιμοποιήσουμε τον ίδιο συντελεστή βαρύτητας και για τα τρία $c_{\text{PER}}=c_{\text{MLP}}=c_{\text{RBF}}=1/3$, οπότε

$$F(\mathbf{x}) = 1/3(-1-0.2+0.6) = -0.2.$$

Σύμφωνα με το παραπάνω κριτήριο η απόφαση είναι ότι το πρότυπο ανήκει στην κλάση -1, κάτι που συμφωνεί με την πλειοψηφία των μοντέλων.

Αν ωστόσο, πιστεύουμε ότι, πχ., το δίκτυο RBF είναι καλύτερος εκτιμητής της κατάστασης απ' ότι τα άλλα δύο μοντέλα, τότε μπορούμε να δώσουμε μεγαλύτερη βαρύτητα σ' αυτό, θέτοντας για παράδειγμα, $c_{\text{RBF}}=0.8$, $c_{\text{PER}}=c_{\text{MLP}}=0.1$, οπότε

$$F(\mathbf{x}) = +0.36.$$

Στην περίπτωση αυτή βαράίνει η επιμηγορία του δικτύου RBF έναντι των υπολοίπων δικτύων και η απόφαση της επιτροπής είναι ότι το πρότυπο ανήκει στην κλάση 1.

Τα ερωτήματα που εύλογα γεννιούνται είναι δύο:

- (a) ποιο είναι το κίνητρο για να χρησιμοποιήσει κανείς επιτροπές και τι κερδίζει κανείς χρησιμοποιώντας τις; και
- (b) ποιες είναι οι καταλληλότερες τιμές των συντελεστών βαρύτητας ώστε να επιτύχουμε τα καλύτερα αποτελέσματα;

Η απάντηση στο πρώτο ερώτημα έχει να κάνει με τον υπολογισμό του σφάλματος που επιτυγχάνεται από την επιτροπή. Μας ενδιαφέρει το σφάλμα γενικά, σε όλη την κατανομή τιμών εισόδου \mathbf{x} και όχι το εμπειρικό ρίσκο σε κάποιο συγκεκριμένο σύνολο εκπαίδευσης. Θα θεωρήσουμε ότι η είσοδος \mathbf{x} είναι μια τυχαία μεταβλητή με στόχους $d(\mathbf{x})$ και θα υπολογίσουμε την μέση τιμή του τετραγωνικού σφάλματος της επιτροπής:

$$J_{\text{επιτροπή}} = E\{[d(\mathbf{x}) - F(\mathbf{x})]^2\} \quad (236)$$

Για να απλουστεύσουμε τα πράγματα, ας υποθέσουμε κατ' αρχήν ότι οι συντελεστές βαρύτητας c_i είναι όλοι ίσοι με $1/N$. Αν

$$J_i = E\{[d(\mathbf{x}) - f_i(\mathbf{x})]^2\} \quad (237)$$

είναι το μέσο σφάλμα του i -στού μοντέλου, τότε χρησιμοποιώντας την (235) έχουμε

$$\begin{aligned} J_{\text{επιτροπή}} &= E\left\{\left[d(\mathbf{x}) - \frac{1}{N} \sum_{i=1}^N f_i(\mathbf{x})\right]^2\right\} \\ &= E\left\{\left[\frac{1}{N} \sum_{i=1}^N (d(\mathbf{x}) - f_i(\mathbf{x}))\right]^2\right\} \end{aligned}$$

Ας ονομάσουμε

$$e_i(\mathbf{x}) = d(\mathbf{x}) - f_i(\mathbf{x})$$

το σφάλμα του i -στού μοντέλου για το πρότυπο \mathbf{x} , και ας υποθέσουμε ότι τα σφάλματα των διαφορετικών μοντέλων είναι ασυχέτιστα μεταξύ τους δηλαδή, $E\{e_i(\mathbf{x}) e_j(\mathbf{x})\} = 0$, για $i \neq j$. Τότε

$$\begin{aligned} J_{\text{επιτροπή}} &= \frac{1}{N^2} E\left\{\left[\sum_{i=1}^N e_i\right]^2\right\} \\ &= \frac{1}{N^2} \sum_{i=1}^N E\{e_i^2\} \\ &= \frac{1}{N} \left\{ \frac{1}{N} \sum_{i=1}^N J_i \right\} \end{aligned} \quad (238)$$

Το παραπάνω αποτέλεσμα είναι εντυπωσιακό και μας δίνει εν μέρει απάντηση στο πρώτο ερώτημα. Από την (238) προκύπτει ότι το μέσο τετραγωνικό σφάλμα της επιτροπής είναι N φορές μικρότερο από το μέσο όρο των μέσων τετραγωνικών σφαλμάτων των επί μέρους μοντέλων που συμμετέχουν στην επιτροπή. Συνεπώς η χρήση της επιτροπής μειώνει σημαντικά το μέσο σφάλμα εκτίμησης, υπό την προϋπόθεση ότι τα σφάλματα των επί μέρους μοντέλων να είναι μεταξύ τους ασυχέτιστα.

Στην πράξη, συνήθως, η παραπάνω προϋπόθεση δεν ισχύει. Στην περίπτωση αυτή μπορούμε να χρησιμοποιήσουμε την ανισότητα Cauchy: $[\sum_{i=1}^N e_i]^2 \leq N \sum_{i=1}^N e_i^2$, για να πάρουμε το παρακάτω σημαντικό, αν και λιγότερο εντυπωσιακό, αποτέλεσμα:

$$J_{\text{επιτροπή}} \leq \frac{1}{N} \sum_{i=1}^N J_i. \quad (239)$$

Σύμφωνα με την (239), το σφάλμα της επιτροπής δεν είναι, σε καμιά περίπτωση, χειρότερο από το μέσο σφάλμα των επί μέρους μοντέλων που την απαρτίζουν.

Είναι δυνατή η περαιτέρω βελτίωση της επίδοσης της επιτροπής αν επιλέξουμε τις βέλτιστες τιμές των συντελεστών βαρύτητας c_i . Αυτό θα μας δώσει απάντηση στο δεύτερο ερώτημα. Για τον υπολογισμό των τιμών αυτών είναι απαραίτητος ο υπολογισμός της του πίνακα \mathbf{R} που αποτελείται από τις τιμές $r_{ij} = E\{e_i e_j\}$. Αν $\Sigma = \mathbf{R}^{-1}$ τότε αποδεικνύεται ότι οι βέλτιστες τιμές των συντελεστών δίνονται από τον τόπο (βλ. [14, Κεφ. 9])

$$c_i = \frac{\sum_{n=1}^N \sigma_{i,n}}{\sum_{m=1}^N \sum_{n=1}^N \sigma_{m,n}} \quad (240)$$

όπου $\sigma_{m,n}$ είναι τα στοιχεία του πίνακα Σ . Τότε το μέσο τετραγωνικό σφάλμα J' της επιτροπής είναι μικρότερο ή ίσο από το αντίστοιχο σφάλμα της απλής επιτροπής με συντελεστές $c_i = 1/N$ [140].

Bagging

Με βάση την παραπάνω θεωρία, είναι προφανές ότι η επίδοση της επιτροπής μπορεί να βελτιωθεί σημαντικά αν τα επί μέρους μοντέλα εκπαιδευτούν έτσι ώστε η συσχέτιση μεταξύ των σφαλμάτων τους μειωθεί. Θα μπορούσαμε να ισχυριστούμε ότι τα μοντέλα είναι ασυχέτιστα μεταξύ τους αν εκπαιδεύονταν πάνω σε διαφορετικά υποσύνολα δεδομένων. Μια τέτοια ιδέα προτάθηκε από τον Breiman [18] η οποία ονομάστηκε Bagging από τα αρχικά των λέξεων *Bootstrap Aggregation*. Επειδή συνήθως έχουμε πεπερασμένο πλήθος P δειγμάτων στη διάθεσή μας, δημιουργούμε τα N διαφορετικά υποσύνολα εκπαίδευσης με τη μέθοδο **Bootstraping**: για κάθε υποσύνολο επιλέγουμε τυχαία K πρότυπα, με πιθανές επαναλήψεις, από το σύνολο των P προτύπων. Έτσι είναι δυνατόν να έχουμε το ίδιο πρότυπο περισσότερες από μια φορές στο ίδιο υποσύνολο. Αν και τα διαφορετικά υποσύνολα εκπαίδευσης δεν είναι τελείως διαφορετικά μεταξύ τους, μπορούμε να πούμε ότι μιμούμεθα την ιδανική περίπτωση.

Πειραματικά αποτελέσματα δείχνουν ότι η μέθοδος bagging έχει καλύτερη επίδοση σε σχέση με την επιτροπή που προκύπτει από τον απλό μέσο όρο των μελών της.

Ενίσχυση (Boosting)

Η μέθοδος της Ενίσχυσης (Boosting) είναι μια σειριακή μέθοδος υλοποίησης μιας επιτροπής όπου τα μέλη της εκπαιδεύονται το ένα μετά το άλλο. Η εκπαίδευση του κάθε μοντέλου εξαρτάται από την επίδοση των προηγούμενων μοντέλων. Υπάρχουν πολλές παραλλαγές της μεθόδου. Η αρχική προσέγγιση που προτάθηκε είναι γνωστή ως ενίσχυση με φίλτραρισμα (*boosting by filtering*) [208]. Σύμφωνα με αυτήν, η επιτροπή αποτελείται από τρία νευρωνικά μοντέλα. Το πρώτο μοντέλο εκπαιδεύεται πάνω σε K πρότυπα που επιλέγονται τυχαία από το σύνολο των προτύπων. Το δεύτερο μοντέλο εκπαιδεύεται πάνω σε K πρότυπα επίσης, αλλά τα πρότυπα αυτά επιλέγονται έτσι ώστε τα μισά από αυτά ταξινομήθηκαν σωστά και τα άλλα μισά ταξινομήθηκαν λάθος από το πρώτο δίκτυο. Το τρίτο δίκτυο εκπαιδεύεται πάνω σε πρότυπα για τα οποία τα δύο πρώτα μοντέλα διαφωνούν ως προς την ταξινόμησή τους. Αφού εκπαιδευτούν και τα τρία δίκτυα, η επιτροπή αποφασίζει για την ταξινόμηση ενός προτύπου κατά πλειοψηφία μετά από ψηφοφορία των τριών μελών της.

Το πρόβλημα είναι ότι η παραπάνω προσέγγιση απαιτεί πολύ μεγάλα σύνολα προτύπων εκπαίδευσης. Για παράδειγμα, απαιτείται να βρούμε K πρότυπα πάνω στα οποία τα δύο πρώτα δίκτυα να διαφωνούν. Αντό δεν είναι εύκολο αν έχουμε μικρό πλήθος δεδομένων. Τη λύση δίνει η μέθοδος της Αντοπροσαρμοστικής Ενίσχυσης (Adaptive Boosting) γνωστή ως AdaBoost [57]. Η μέθοδος συνδυάζει ιδέες τόσο από την μέθοδο Bagging όσο και από την μέθοδο Boosting. Μια από τις διάφορες παραλλαγές που έχουν προταθεί περιγράφεται στον Αλγόριθμο 24 (βλ. [18]).

Σύμφωνα με την μέθοδο αυτή τα μέλη της επιτροπής εκπαιδεύονται χρησιμοποιώντας bootstraping, όπως και στην περίπτωση του bagging. Τώρα όμως, η πιθανότητα να επιλεγεί ένα πρότυπο ώστε να γίνει μέλος του συνόλου εκπαίδευσης για ένα μοντέλο εξαρτάται από τα προηγούμενα μοντέλα. Αν ένα πρότυπο j ταξινομηθεί λάθος από ένα μοντέλο (δηλαδή αν $d(j)=1$) τότε αυξάνεται η πιθανότητά $p(j)$ του προτύπου αυτού να επιλεγεί από το επόμενο μοντέλο. Για την ακρίβεια η πιθανότητά των προτύπων που ταξινομήθηκαν λάθος πολλαπλασιάζεται επί b_i ενώ γι' αυτά που ταξινομήθηκαν σωστά πολλαπλασιάζεται επί 1. Αρχικά όλα τα πρότυπα είναι ισοπιθανά.

Επί πλέον, ο αριθμός b_i είναι μεγάλος αν το μοντέλο i έκανε συνολικά λίγες λάθος ταξινομήσεις. Στην περίπτωση αυτή οι πιθανότητες των προτύπων που ταξινομήθηκαν λάθος αυξάνουν πολύ ώστε είναι σχεδόν σύγουρο ότι θα επιλεγούν για το σύνολο εκπαίδευσης του επόμενου μοντέλου. Τέλος οι αποφάσεις των μελών της επιτροπής ζυγίζονται επί $\log(b_i)$ συνεπώς δίνουμε μεγαλύτερη βαρύτητα στα μοντέλα που είναι καλοί ταξινομητές.

Αλγόριθμος 24: Επιτροπή με Ενίσχυση (Boosting)

Αρχικοποίησε τις πιθανότητες των δειγμάτων $p(1) = p(2) = \dots = p(P) = 1/P$
Για κάθε μοντέλο-μέλος της επιτροπής $i=1, \dots, N$ {

Πάρε K δείγματα με Bootstraping από το σύνολο των προτύπων χρησιμοποιώντας τις πιθανότητες $p(j)$ και εκπαίδευσε το μοντέλο i μ' αυτά.

Για κάθε πρότυπο $j=1, \dots, K$ {

Θέσε $d(j)=1$ αν το πρότυπο j ταξινομήθηκε λάθος
αλλιώς $d(j)=0$

}

$$\text{Θέσε } \omega_i = \sum_{j=1}^K d(j)p(j)$$

$$b_i = (1 - \omega_i)/\omega_i$$

$$C = 1 / \sum_{j=1}^K p(j)b_i^{d(j)}$$

Για κάθε πρότυπο $j=1, \dots, K$ {

Ενημέρωσε τις πιθανότητες $p(j) \leftarrow C p(j) b_i^{d(j)}$

}

}

Η απόφαση της επιτροπής δίνεται μετά από ψηφοφορία των μελών με συντελεστές βαρύτητας $c_i = \log(b_i)$.

Άλλες παραλλαγές του αλγορίθμου AdaBoost περιλαμβάνουν την χρήση βαρών για τα πρότυπα (αντί για πιθανότητες) [59], τη χρήση του AdaBoost για εκτίμηση συναρτήσεων (regression) ή ταξινόμηση προτύπων με επίπεδα εμπιστοσύνης [241], το κανονικοποιημένο (regularized) AdaBoost [196],

και το n -Arc για θορυβώδη δεδομένα [197]. Έχει επίσης διαπιστωθεί η σχέση μεταξύ Ενίσχυσης και Μηχανών Διανυσμάτων Υποστήριξης καθώς και τα δύο βρίσκουν το κατάλληλο διαχωριστικό επίπεδο σε χώρο μεγάλων διαστάσεων με μεγάλο περιθώριο [209,49].

10.2.7 Μίγματα Εμπειρογνωμόνων (Mixtures of Experts)

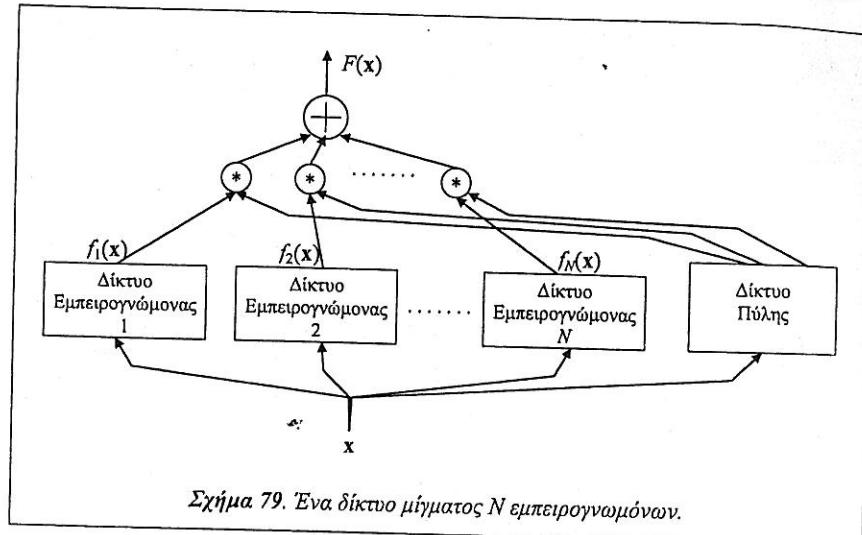
Τα μίγματα εμπειρογνωμόνων [100] είναι μοντέλα που συνδυάζουν τις εξόδους από διάφορα νευρωνικά δίκτυα, όπως οι επιτροπές, με τη διαφορά ότι ο συντελεστής βαρύτητας c_i για το δίκτυο i είναι συνάρτηση του διανύσματος εισόδου x , οπότε ο συνδυασμένος εκτιμητής είναι

$$F(x) = \sum_{i=1}^N c_i(x) f_i(x; \theta_i). \quad (241)$$

Τα μέλη της επιτροπής είναι τα N νευρωνικά δίκτυα (εμπειρογνώμονες) $f_i(x; \theta_i)$, $i=1, 2, \dots, N$. Οι συντελεστές βαρύτητας $c_i(x)$ είναι θετικοί αριθμοί που αθροίζουν στο 1 και ορίζουν ποιοι «εμπειρογνώμονες» είναι υπεύθυνοι για κάθε περιοχή του χώρου x . Οι συντελεστές αυτοί παράγονται από ένα άλλο νευρωνικό δίκτυο, το «Δίκτυο Πύλης» (Gating Network) όπως φαίνεται στο Σχήμα 79. Το δίκτυο αυτό δέχεται την ίδια είσοδο x , όπως και οι εμπειρογνώμονες, και παράγει N εξόδους γ_i από τις οποίες παράγονται οι συντελεστές βαρύτητας ως εξής:

$$c_i(x) = \frac{\exp\{\gamma_i(x)\}}{\sum_{j=1}^N \exp\{\gamma_j(x)\}}. \quad (242)$$

Η συνάρτηση (242) είναι γνωστή ως συνάρτηση softmax [19] και εξασφαλίζεται ότι οι συντελεστές είναι όλοι θετικοί και αθροίζουν στη μονάδα, οπότε δεν χρειάζεται καμία άλλη κανονικοποίηση. Τα δίκτυα εμπειρογνώμονες εκπαιδεύονται με επίβλεψη ταυτόχρονα με το δίκτυο πύλης. Για κάθε πρότυπο εισόδου $x^{(p)}$, $p=1, \dots, P$, υπάρχει ένας στόχος $d^{(p)}$ ο οποίος είναι κοινός για όλα τα δίκτυα εμπειρογνωμόνων. Έτσι μπορούμε να πούμε ότι τα δίκτυα αυτά εκπαιδεύονται ανεξάρτητα το ένα από το άλλο.



Σχήμα 79. Ένα δίκτυο μίγματος N εμπειρογνωμόνων.

Από την άλλη μεριά, το δίκτυο πύλης εκπαιδεύεται έτσι ώστε να ελαχιστοποιηθεί μια από τις παρακάτω συναρτήσεις κόστους

$$J_1 = \sum_{p=1}^P \sum_{i=1}^N c_i(x^{(p)}) [d^{(p)} - f_i(x^{(p)})]^2$$

$$\text{ή } J_2 = -\sum_{p=1}^P \log \sum_{i=1}^N c_i(x) \exp\left\{-\frac{1}{2}[d^{(p)} - f_i(x)]^2\right\}$$

Συνήθως επιλέγεται η δεύτερη συνάρτηση (J_2) διότι δίνει καλύτερα αποτελέσματα στην πράξη. Σκοπός του δικτύου πύλης είναι να διαμοιράσει το χώρο του διανύσματος εισόδου σε περιοχές όπου σε κάθε περιοχή κυριαρχεί ένας εμπειρογνώμονας διότι αυτός την περιγράφει καλύτερα. Έτσι μπορεί ο εμπειρογνώμονας i (πχ. ένα δίκτυο RBF) να κυριαρχεί στην περιοχή A ενώ ο εμπειρογνώμονας j (πχ. ένα δίκτυο MLP) να υπερισχύει στην περιοχή B . Αυτό έχει σαν αποτέλεσμα την καλύτερη προσέγγιση των δεδομένων κατά την εκπαίδευση αλλά και τη βελτίωση της ικανότητας γενίκευσης του μοντέλου. Έπισης έχουν μελετηθεί διάφορες περιπτώσεις μοντέλων εμπειρογνωμόνων, όπως

(a) Γραμμικοί εμπειρογνώμονες:

$$f_i(x) = \theta_i^T x$$

Για λεπτομερή περιγραφή της μεθόδου στην περίπτωση αυτή, βλ. [79, κεφάλαιο 7].

(β) Γκαουσιανοί εμπειρογνώμονες:

$$f_i(\mathbf{x}) = s_i \exp\left\{-\frac{1}{2\sigma_i^2(\mathbf{x})} \|d - \mu_i(\mathbf{x})\|^2\right\}.$$

Στην περίπτωση αυτή το διάνυσμα των παραμέτρων που πρέπει να εκπαίδευται είναι $\theta_i = [s_i, \sigma_i, \mu_i]^T$ (βλ. [14, κεφάλαιο 6]).

Η εκπαίδευση του δικτύου πύλης μπορεί να γίνει με την μέθοδο Back-Propagation. Για τον σκοπό αυτό, θα χρειαστούμε την κλίση (gradient) του κόστους J_2 ως προς τις εξόδους του δικτύου y_i . O Bishop [14] υπολόγισε ότι το gradient δίνεται από τον τύπο

$$\begin{aligned}\frac{\partial J_2}{\partial \gamma_i} &= \sum_{p=1}^P c_i(\mathbf{x}^{(p)}) - \pi_i^{(p)} \\ \pi_i^{(p)} &= \frac{c_i(\mathbf{x}^{(p)}) f_i(\mathbf{x}^{(p)})}{\sum_{j=1}^P c_j(\mathbf{x}^{(p)}) f_j(\mathbf{x}^{(p)})}\end{aligned}$$

Αν το δίκτυο πύλης είναι γραμμικό, δηλαδή $y_i = \mathbf{a}_i^T \mathbf{x}$, τότε τα πράγματα απλοποιούνται πολύ (βλ. [79, κεφάλαιο 7]).

Έχουν προταθεί διάφορες παραλλαγές μιγμάτων εμπειρογνωμόνων. Ο Tresp [220] μελέτησε την περίπτωση όπου και οι εμπειρογνώμονες και το δίκτυο πύλης είναι Γκαουσιανού τύπου. Το Ιεραρχικό μοντέλο Μίγματος Εμπειρογνωμόνων (Hierarchical Mixtures of Experts – HME) αναπτύχθηκε από τους Jordan και Jacobs [107] σύμφωνα με το οποίο υπάρχουν δύο στρώματα δικτύων πύλης. Στην γενική περίπτωση το μοντέλο αυτό έχει τη μορφή δένδρου όπου οι κόμβοι-φύλλα είναι δίκτυα εμπειρογνωμόνων, ενώ οι κόμβοι μη-φύλλα είναι δίκτυα πύλης (gates). Έτσι, ο χώρος του διανύσματος \mathbf{x} χωρίζεται σε περιοχές όπου κάθε περιοχή μπορεί να χωριστεί σε υποπεριοχές, κλπ., με στόχο πάντα την καλύτερη περιγραφή των δεδομένων. Εντυχώς η επί πλέον πολυπλοκότητα του μοντέλου δεν έχει συνέπεια στην ταχύτητα εκπαίδευσης καθώς μπορεί να χρησιμοποιηθεί ο αλγόριθμος EM ο οποίος είναι ταχύς (για λεπτομερή ανάλυση της μεθόδου βλ. [79, κεφάλαιο 7]). Για μια καλή σύνοψη παλαιών και νέων μεθόδων σχετικών με επιτροπές και μίγματα εμπειρογνωμόνων βλ. [221].

Κεφάλαιο 11

Στοιχεία Γραμμικής Άλγεβρας

11.1 Διανυσματικοί χώροι

Η έννοια του διανυσματικού χώρου –ή γραμμικού χώρου– είναι βασική και σχετίζεται με τις περισσότερες μαθηματικές έννοιες που αφορούν τα νευρωνικά δίκτυα. Πίσω από τον διανυσματικό χώρο βρίσκεται η έννοια του βαθμωτού πεδίου. Ένα βαθμωτό πεδίο K είναι ένα σύνολο αριθμών το οποίο είναι κλειστό ως προς τις πράξεις της πρόσθεσης και του πολλαπλασιασμού, δηλαδή τόσο το άθροισμα όσο και το γινόμενο δύο οποιονδήποτε στοιχείων του K ανήκει επίσης στο K . Τόσο η πρόσθεση όσο και ο πολλαπλασιασμός είναι πράξεις που παίρνουν δύο ορίσματα και έχουν τις παρακάτω ιδιότητες:

- Αντιμεταθετική ιδιότητα

$$x + y = y + x, \quad x \cdot y = y \cdot x$$

- Προσεταιριστική ιδιότητα

$$(x + y) + z = x + (y + z), \quad (x \cdot y) \cdot z = x \cdot (y \cdot z)$$

- Ύπαρξη ουδέτερου στοιχείου (καλείται 0 για την άθροιση και 1 για τον πολλαπλασιασμό)

$$x + 0 = x, \quad x \cdot 1 = x$$

- Επιμεριστική ιδιότητα των πολλαπλασιασμού ως προς την πρόσθεση

$$x \cdot (y + z) = x \cdot y + x \cdot z$$

Επί πλέον όλα τα στοιχεία του K έχουν ένα αντίστροφο στο K ως προς την πρόσθεση και όλα τα στοιχεία του K εκτός από το 0, έχουν ένα αντίστροφο στο K ως προς τον πολλαπλασιασμό. Όλα τα στοιχεία του K καλούνται βαθμωτοί αριθμοί ή απλά αριθμοί. Αντίστροφος ενός αριθμού x ως προς μια πράξη P καλείται ένας αριθμός y ο οποίος κάνοντας την πράξη xP δίνει σαν αποτέλεσμα το ουδέτερο στοιχείο της πράξης P .