



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Διπλωματική Εργασία

**Αποτίμηση Μηχανών Ακραίας Μάθησης σε Εφαρμογές Ταξινόμησης και
Συναρτησιακής Προσέγγισης σε Περιβάλλον MATLAB**

Ευκαρπίδης Γεώργιος – Μιχαήλ
A.M. : 711 – 131105

Εισηγητής / Επιβλέπων Καθηγητής: Βασιλάς Νικόλαος

Αθήνα, Ελλάδα, Μάρτιος 2024

Διπλωματική Εργασία

**Αποτίμηση Μηχανών Ακραίας Μάθησης σε Εφαρμογές Ταξινόμησης και
Συναρτησιακής Προσέγγισης σε Περιβάλλον MATLAB**

**Ευκαρπίδης Γεώργιος – Μιχαήλ
Α.Μ. : 711 – 131105**

Εισηγητής / Επιβλέπων Καθηγητής:

Βασιλάς Νικόλαος, Καθηγητής

Εξεταστική Επιτροπή:

**Μπαρδής Γεώργιος,
Επίκουρος Καθηγητής**

**Γεωργουλάκη Χριστίνα,
Εργαστηριακό Διδακτικό Προσωπικό**

Ημερομηνία Εξέτασης 20/3/2024

Η εν λόγω διπλωματική εργασία υποβλήθηκε στο τμήμα Μηχανικών Πληροφορικής και Υπολογιστών του Πανεπιστημίου Δυτικής Αττικής στα πλαίσια του πενταετούς (5) προπτυχιακού προγράμματος σπουδών.

Αθήνα, Ελλάδα, Μάρτιος 2024

Δήλωση Συγγραφέα Μεταπτυχιακής Εργασίας

Ο κάτωθι υπογεγραμμένος **Ευκαρπίδης Γεώργιος – Μιχαήλ** του Κωνσταντίνου και της Ελένης με Α.Μ. **711 – 131105** φοιτητής του τμήματος Μηχανικών Πληροφορικής και Υπολογιστών της Σχολής Μηχανικών του Πανεπιστημίου Δυτικής Αττικής, δηλώνει υπεύθυνα ότι:

«Είμαι ο συγγραφέας αυτής της διπλωματικής εργασίας και ότι κάθε αρωγή που είχα και ήταν αναγκαία για την εκπόνηση και δημιουργία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Ακόμη, όλες οι πηγές από τις οποίες αντλήθηκαν σκέψεις, ιδέες, τεχνικές, μέθοδοι και γνώσεις έχουν αποτυπωθεί στο σύνολό τους με πλήρη αναφορά στους συγγραφείς που ανήκουν. Επίσης, βεβαιώνεται ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά καθώς και ότι αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του ιδρύματος.»

Ο Δηλών

Ευκαρπίδης Γεώργιος – Μιχαήλ

Ευχαριστίες

Η συγκεκριμένη διπλωματική εργασία ολοκληρώθηκε μετά από επίμονες προσπάθειες, με απώτερο σκοπό την ενασχόληση, κατανόηση και εξέταση του αντικειμένου των Τεχνητών Νευρωνικών Δικτύων.

Ιδιαίτερη μνεία θα ήθελα να αποδώσω σε αυτό το σημείο στον καθηγητή κ. Βασιλά Νικόλαο, όπου ως επιβλέπων της συγκεκριμένης διπλωματικής εργασίας μου έδειξε εμπιστοσύνη και με καθοδήγησε στην επίτευξη των στόχων που είχαμε θέσει.

Τέλος, άξιοι επαίνων και συγχαρητηρίων είναι οι γονείς μου και όλοι οι φίλοι μου, που με υποστήριξαν και μου συμπαράστηκαν σε όλη τη διαδρομή για την επιτυχή διεκπεραίωση των σπουδών μου.

Περίληψη

Στην κάτωθι διπλωματική εργασία εξετάζεται η αποτελεσματικότητα, η ακρίβεια, η ταχύτητα και η απόδοση του νευρωνικού αλγόριθμου ELM (Extreme Learning Machine) ή αλλιώς Ακραία Μηχανή Μάθησης, ο οποίος προτάθηκε από τον Guang-Bin Huang και τους συνεργάτες του Qin-Yu Zhu και Chee-Kheong Siew το 2006. Το πλαίσιο στο οποίο εξετάζεται είναι τα προβλήματα συναρτησιακής παλινδρόμησης, καθώς και ένα σύνολο από προβλήματα ταξινόμησης που αποσκοπούν στην κατανόηση της συμπεριφοράς και των επιδόσεων του ELM.

Βασικός ανταγωνιστής στη σύγκριση για την αποτίμηση αποτελεί ο Backpropagation σε διάφορες παραλλαγές του, για τον οποίο έχει εκφραστεί η κοινότητα των νευρωνικών δικτύων πως πρόκειται κατά τα φαινόμενα για τον πιο επιδραστικό, αποτελεσματικό και εύκολα προσαρμόσιμο αλγόριθμο για τα νευρωνικά δίκτυα. Δημοφιλείς παραλλαγές του Backpropagation που εξετάζονται είναι αυτές του Levenberg – Marquardt, ο Resilient Backpropagation και η Scaled Conjugate Gradient.

Στο πλαίσιο της διπλωματικής εργασίας αναλύονται τα Τεχνητά Νευρωνικά Δίκτυα ιστορικά και δομικά, καθώς και ένα σύνολο από κανόνες, νευρωνικά μοντέλα και αλγορίθμους που έχουν αναπτυχθεί από το πρόσφατο παρελθόν ως και σήμερα, και έχουν συμβάλει τα μέγιστα στην άνθιση της Τεχνητής Νοημοσύνης και των νευρωνικών δικτύων.

Οι προσομοιώσεις των προβλημάτων ταξινόμησης και λειτουργικής παλινδρόμησης πραγματοποιούνται στο περιβάλλον MATLAB και στο τέλος το συμπέρασμα της γενικευμένης απόδοσης του ELM αποκαλύπτεται.

Επιστημονική Περιοχή: Τεχνητά Νευρωνικά Δίκτυα, Βάσεις Δεδομένων

Λέξεις Κλειδιά: ELM, Backpropagation, MATLAB, MINST, RMSE, Παλινδρόμηση, Ταξινόμηση, Νευρωνικό Δίκτυο

Abstract

The following Thesis based on the examination of the efficiency, accuracy, speed and performance of ELM neural algorithm (Extreme Learning Machine), which was proposed by the Guang-Bin Huang and his colleagues Qin-Yu Zhu and Chee -Kheong Siew in 2006. The algorithm is examined in functional regression problems, as well as a set of classification problems which are intended at the behavior and performance of ELM.

The main competitor at the comparison of evaluation is the Backpropagation in different variations, which is the most influential, efficient and easily adaptive neural network based on expressions of neural network community. The popular variations of Backpropagation which examined are Levenberg–Marquardt, Resilient Backpropagation, and Scaled Conjugate Gradient.

In the context of the thesis, Artificial Neural Networks are analyzed structurally and historically, as well as rules, neural network models and algorithms that have been developed in the recent past until today, and have contributed the most to the flourishing of Artificial Intelligence and neural networks.

The simulations of classification and functional regression problems are conducted in the MATLAB environment, and at the end the conclusion of the generalized performance of ELM is revealed.

Scientific Area: Artificial Neural Networks, Databases

Περιεχόμενα

| | |
|---|----|
| Ευχαριστίες | 6 |
| Περίληψη | 9 |
| Abstract | 11 |
| Περιεχόμενα | 13 |
| Κατάλογος Εικόνων | 18 |
| Κατάλογος Πινάκων | 19 |
| Συντομογραφίες – Αρκτικόλεξο | 21 |
| Κεφάλαιο 1 Τα Τεχνητά Νευρωνικά Δίκτυα | 22 |
| 1.1 Ο Νευρώνας | 22 |
| 1.1.1 Η λειτουργία του βιολογικού νευρώνα | 23 |
| 1.1.2 Στοιχεία των εγκεφάλων | 23 |
| 1.1.3 Το μοντέλο McCulloch – Pitts | 24 |
| 1.1.4 Άλλα διαδεδομένα μοντέλα | 25 |
| 1.2 Εκπαίδευση νευρωνικών δικτύων | 26 |
| Κεφάλαιο 2 Οι Νευρωνικοί Αλγόριθμοι | 28 |
| 2.1 Ταξινόμηση νευρωνικών αλγορίθμων | 28 |
| 2.2 Νευρωνικοί αλγόριθμοι με εκπαιδευόμενα βάρη με επίβλεψη | 28 |
| 2.2.1 Perceptron | 29 |
| 2.2.1.1 Παράμετροι του Perceptron | 30 |
| 2.2.1.2 Ο αλγόριθμος εκπαίδευσης του Perceptron | 31 |
| 2.2.2 ADALINE | 31 |
| 2.2.2.1 Η επανάληψη της εκπαίδευσης του δικτύου ADALINE | 32 |
| 2.2.2.2 Ο αλγόριθμος εκπαίδευσης του ADALINE | 32 |
| 2.2.2.3 MADALINE | 33 |
| 2.2.3 Backpropagation | 33 |
| 2.2.3.2 Πλεονεκτήματα και Μειονεκτήματα των Backpropagation αλγορίθμων | 34 |
| 2.2.3.3 Σχηματική απεικόνιση δικτύου με Backpropagation εκμάθηση | 35 |
| 2.2.4 Αναδρομικά δίκτυα Backpropagation | 36 |
| 2.2.4.1 Σχηματική απεικόνιση | 36 |
| 2.2.5 Δίκτυα RBF | 37 |
| 2.2.5.1 Ιστορική αναφορά | 37 |
| 2.2.5.2 Σύγκριση RBF – MLP | 37 |
| 2.2.5.3 Εκπαίδευση δικτύου RBF | 38 |
| 2.2.5.4 Δημοφιλείς συναρτήσεις ακτινικού τύπου | 38 |
| 2.2.6 Μηχανές Διανυσμάτων Υποστήριξης (SVM) | 39 |
| 2.2.6.1 Προέλευση και Ανάπτυξη των SVMs | 39 |
| 2.2.6.2 Πλεονεκτήματα – Μειονεκτήματα των SVMs και η εφαρμογή τους | 39 |
| 2.2.6.3 Μαθηματική προσέγγιση των Μηχανών Διανυσμάτων Υποστήριξης | 40 |
| 2.2.7 Στοχαστικές Μηχανές | 41 |
| 2.2.7.1 Μηχανή Boltzmann | 41 |
| 2.2.7.2 Πλεονεκτήματα, Μειονεκτήματα και Σύγκριση με Ντετερμινιστικά Νευρωνικά Δίκτυα | 42 |
| 2.2.7.3 Μαρκοβιανές αλυσίδες | 43 |
| 2.3 Νευρωνικοί αλγόριθμοι με εκπαιδευόμενα βάρη χωρίς επίβλεψη | 43 |
| 2.3.1 Συσχετιστικά Μοντέλα | 44 |
| 2.3.2 Κανόνας Hebb | 45 |
| 2.3.3 Ανάλυση Κυρίων Συνιστωσών (Principal Component Analysis) | 45 |
| 2.3.3.1 Η εφαρμογή της PCA, τα πλεονεκτήματα και τα μειονεκτήματά της | 46 |

| | | |
|-------------------|---|-----------|
| 2.3.3.2 | Επιπλέον είδη κανόνων | 46 |
| 2.3.4 | Ανάλυση Ανεξαρτήτων Συνιστωσών (Independent Component Analysis) | 46 |
| 2.3.4.1 | Ιστορική αναδρομή και ανάλυση | 47 |
| 2.3.4.2 | Πλεονεκτήματα και Μειονεκτήματα της μεθόδου | 47 |
| 2.3.5 | Ανταγωνιστικά Μοντέλα (Generative Adversarial Networks) | 48 |
| 2.3.6 | Δίκτυο Kohonen (SOM) - LVQ | 49 |
| 2.3.6.1 | Οργάνωση και διεξαγωγή των δικτύων Kohonen | 49 |
| 2.3.6.2 | Κανόνας εκπαίδευσης Kohonen | 50 |
| 2.3.6.3 | Τοπολογία SOM | 51 |
| 2.3.6.4 | Learning Vector Quantization (LVQ) | 51 |
| 2.3.6.5 | Αρχιτεκτονική Learning Vector Quantization | 51 |
| 2.3.7 | Το δίκτυο ART | 52 |
| 2.3.7.1 | Πλεονεκτήματα των δικτύων ART | 53 |
| 2.4 | Νευρωνικοί αλγόριθμοι με σταθερά βάρη | 54 |
| 2.4.1 | Το δίκτυο Hopfield | 54 |
| 2.4.1.1 | Αρχιτεκτονική του δικτύου Hopfield και η εκπαίδευση | 54 |
| 2.4.1.2 | Λοιπές πληροφορίες για τα μοντέλα Hopfield | 55 |
| 2.4.2 | Συσχετιστικές Μνήμες | 55 |
| 2.4.2.1 | Ταξινόμηση των Συσχετιστικών Μνημών | 56 |
| 2.4.2.2 | Τρόπος λειτουργίας των Συσχετιστικών Μνημών | 56 |
| 2.4.2.3 | Πλεονεκτήματα – Μειονεκτήματα και Εφαρμογές των Συσχετιστικών Μνημών | 56 |
| 2.4.2.4 | Παραδείγματα Αυτοσυσχετιστικών και Ετεροσυσχετιστικών Μνημών | 57 |
| 2.4.3 | Brain State in a Box | 58 |
| 2.4.3.1 | Ιστορική Αναδρομή και Περιγραφή | 58 |
| Κεφάλαιο 3 | Αποσαφήνιση Εννοιών | 59 |
| Κεφάλαιο 4 | Backpropagation και Παραλλαγές | 63 |
| 4.1 | Θεώρημα Kolmogorov | 63 |
| 4.2 | Κατηγορίες του Backpropagation | 63 |
| 4.2.1 | Βασικές κατηγορίες του Backpropagation | 64 |
| 4.2.1.1 | Στατικός Backpropagation (Static Backpropagation) | 64 |
| 4.2.1.2 | Επαναλαμβανόμενος Backpropagation (Recurrent Backpropagation) | 64 |
| 4.2.1.3 | Ελαστικός Backpropagation (Resilient Backpropagation) | 65 |
| 4.2.1.4 | Backpropagation μέσω χρόνου (Backpropagation through time) | 66 |
| 4.3 | Παραλλαγές του Backpropagation | 66 |
| 4.4 | Levenberg – Marquardt Backpropagation | 68 |
| 4.5 | Resilient Backpropagation | 69 |
| 4.5.1 | Μαθηματική προσέγγιση του Resilient Backpropagation | 69 |
| 4.5.2 | Προσέγγιση με Ψευδοκώδικα του Resilient Backpropagation | 70 |
| 4.6 | Scaled Conjugate Gradient | 71 |
| 4.6.1 | Μαθηματική προσέγγιση του Scaled Conjugate Gradient | 71 |
| Κεφάλαιο 5 | Extreme Learning Machine | 73 |
| 5.1 | Ανάλυση και μαθηματική προσέγγιση του ELM | 73 |
| 5.1.1 | Θεωρητικό υπόβαθρο βαρών, πολώσεων και συναρτήσεων ενεργοποίησης | 74 |
| 5.1.2 | Μαθηματική μοντελοποίηση της δομής του ELM | 74 |
| 5.1.3 | Περιγραφική διατύπωση των βημάτων του αλγορίθμου ELM | 76 |
| 5.2 | Αποτίμηση χρήσης του ELM σε σχέση με τους συμβατικούς SLFN αλγόριθμους | 76 |
| 5.2.1 | Πλεονεκτήματα ELM | 76 |
| 5.2.2 | Μειονεκτήματα – Περιορισμοί ELM | 77 |
| 5.3 | Τοπολογία και παραλλαγές του ELM | 77 |

| | | |
|-------------------|--|-----------|
| 5.3.1 | Τοπολογία του ELM | 77 |
| 5.3.2 | Παραλλαγές του ELM | 78 |
| 5.4 | Η προσέγγιση των Widrow – Krause | 78 |
| 5.4.1 | Ο No-Prop αλγόριθμος του Widrow | 78 |
| 5.4.2 | Τα πειράματα για τον No-Prop από τον Widrow | 79 |
| 5.4.3 | Τα πειράματα του Krause για τον No-Prop | 80 |
| Κεφάλαιο 6 | Ανάλυση Προβλημάτων και Πειραμάτων | 81 |
| 6.1 | Το πρόβλημα της Συναρτησιακής Παλινδρόμησης | 81 |
| 6.2 | Τα προβλήματα Ταξινόμησης | 82 |
| 6.2.1 | Η αναγνώριση του φυτού “IRIS” | 83 |
| 6.2.2 | Το πρόβλημα του διαχωρισμού αντικειμένων “IMAGE SEGMENT” | 84 |
| 6.2.3 | Η αναγνώριση της μορφολογίας του εδάφους από δορυφόρο “SATELLITE IMAGES” | 85 |
| 6.2.4 | Η βάση δεδομένων “MNIST” | 86 |
| Κεφάλαιο 7 | Α’ Πρακτικό – Πειραματικό σκέλος – Συναρτησιακή Παλινδρόμηση | 87 |
| 7.1 | Αποτύπωση αποτελεσμάτων Συναρτησιακής Παλινδρόμησης μέσω Backpropagation | 87 |
| 7.1.1 | Απόδοση αποτελεσμάτων του Backpropagation με “αμόλυντα δεδομένα” | 87 |
| 7.1.1.1 | Παραλλαγή 1.1: 75% Training dataset – 25% Testing dataset | 87 |
| 7.1.1.2 | Παραλλαγή 1.2: 50% Training dataset – 50% Testing dataset | 89 |
| 7.1.1.3 | Παραλλαγή 1.3: 25% Training dataset – 75% Testing dataset | 90 |
| 7.1.2 | Απόδοση αποτελεσμάτων του Backpropagation με “μολυσμένα δεδομένα” | 90 |
| 7.1.2.1 | Παραλλαγή 2.1: 75% Training dataset – 25% Testing dataset | 91 |
| 7.1.2.2 | Παραλλαγή 2.1: 50% Training dataset – 50% Testing dataset | 92 |
| 7.1.2.3 | Παραλλαγή 2.3: 25% Training dataset – 75% Testing dataset | 93 |
| 7.1.3 | Αποτελέσματα του Backpropagation με “αποκοπή διαστημάτων” | 93 |
| 7.1.3.1 | Παραλλαγή 3.1: 700 / 700 τιμές του πεδίου ορισμού [-5, +5) | 94 |
| 7.1.3.2 | Παραλλαγή 3.2: 350 / 700 τιμές του πεδίου ορισμού [-5, +5) | 95 |
| 7.1.3.3 | Παραλλαγή 3.3: 1400 / 1400 τιμές του πεδίου ορισμού [-10, +10) | 96 |
| 7.1.3.4 | Παραλλαγή 3.4: 700 / 1400 τιμές του πεδίου ορισμού [-10, +10) | 97 |
| 7.1.4 | Αποτίμηση του LM με “μολυσμένα δεδομένα” και “αποκοπή διαστημάτων” | 98 |
| 7.2 | Αποτύπωση αποτελεσμάτων Συναρτησιακής Παλινδρόμησης μέσω ELM | 100 |
| 7.2.1 | Αποτελεσματικότητα του ELM με “αμόλυντα” συναρτησιακά δεδομένα | 100 |
| 7.2.1.1 | Παραλλαγή 1.1: Καθαρά δεδομένα του πεδίου ορισμού [-5, +5), αναλογία 75% - 25% | 100 |
| 7.2.1.2 | Παραλλαγή 1.2: Καθαρά δεδομένα του πεδίου ορισμού [-5, +5), αναλογία 50% - 50% | 101 |
| 7.2.1.3 | Παραλλαγή 1.3: Καθαρά δεδομένα του πεδίου ορισμού [-5, +5), αναλογία 25% - 75% | 101 |
| 7.2.2 | Αποτελεσματικότητα του ELM με “μολυσμένα” συναρτησιακά δεδομένα | 102 |
| 7.2.2.1 | Παραλλαγή 2.1: Ομοιόμορφα κατανεμημένος θόρυβος [-0.1, +0.1], 75% - 25% | 102 |
| 7.2.2.2 | Παραλλαγή 2.2: Ομοιόμορφα κατανεμημένος θόρυβος [-0.1, +0.1], 50% - 50% | 103 |
| 7.2.2.3 | Παραλλαγή 2.3: Ομοιόμορφα κατανεμημένος θόρυβος [-0.1, +0.1], 25% - 75% | 103 |
| 7.2.2.4 | Τελική αποτίμηση των προσομοιώσεων με θόρυβο στα συναρτησιακά δεδομένα | 104 |
| 7.2.3 | Αποτελεσματικότητα του ELM με “αποκοπή διαστημάτων” | 105 |
| 7.2.3.1 | Παραλλαγή 3.1: 700 / 700 τιμές του πεδίου ορισμού [-5, +5) | 105 |
| 7.2.3.2 | Παραλλαγή 3.2: 350 / 700 τιμές του πεδίου ορισμού [-5, +5) | 106 |
| 7.2.3.3 | Παραλλαγή 3.3: 1400 / 1400 τιμές του πεδίου ορισμού [-10, +10) | 107 |
| 7.2.3.4 | Παραλλαγή 3.4: 700 / 1400 τιμές του πεδίου ορισμού [-10, +10) | 108 |
| 7.2.4 | Αποτελεσματικότητα του ELM με “μολυσμένα δεδομένα” και “αποκοπή διαστημάτων” | 109 |
| 7.3 | Αποτίμηση – Σύγκριση αποτελεσμάτων Συναρτησιακής Παλινδρόμησης | 110 |
| 7.3.1 | Σύνολα δεδομένων με καθαρές τιμές | 111 |
| 7.3.2 | Σύνολα δεδομένων με μολυσμένες τιμές | 112 |
| 7.3.3 | Αποκοπή υποδιαστημάτων από το πεδίο ορισμού | 113 |
| 7.3.4 | Αποκοπή υποδιαστημάτων από το πεδίο ορισμού με μολυσμένα δεδομένα | 114 |

| | | |
|-------------------|---|------------|
| 7.3.5 | Σύνοψη των αλγορίθμων στο πρόβλημα της Συναρτησιακής Παλινδρόμησης | 116 |
| Κεφάλαιο 8 | Β' Πρακτικό – Πειραματικό σκέλος – Ταξινόμηση | 117 |
| 8.1 | Προσομοιώσεις σε προβλήματα Ταξινόμησης μέσω Backpropagation | 117 |
| 8.1.1 | Προσομοιώσεις στο πρόβλημα “IRIS” μέσω Backpropagation | 117 |
| 8.1.1.1 | Παραλλαγή 1.1: Αναλογία συνόλων 70/30 | 118 |
| 8.1.1.2 | Παραλλαγή 1.2: Αναλογία συνόλων 50/50 | 119 |
| 8.1.1.3 | Σύνοψη IRIS για τον BP | 119 |
| 8.1.2 | Προσομοιώσεις στο πρόβλημα “IMAGE SEGMENT” μέσω Backpropagation | 120 |
| 8.1.2.1 | Παραλλαγή 2.1: Αναλογία συνόλων 80/20 | 120 |
| 8.1.2.2 | Παραλλαγή 2.2: Αναλογία συνόλων 65/35 | 121 |
| 8.1.2.3 | Παραλλαγή 2.3: Αναλογία συνόλων 50/50 | 121 |
| 8.1.2.4 | Σύνοψη Image Segment για τον BP | 122 |
| 8.1.3 | Προσομοιώσεις στο πρόβλημα “SATELLITE IMAGES” μέσω Backpropagation | 123 |
| 8.1.3.1 | Παραλλαγή 3.1: Αναλογία συνόλων 80/20 | 123 |
| 8.1.3.2 | Παραλλαγή 3.2: Αναλογία συνόλων 68/32 | 124 |
| 8.1.3.3 | Παραλλαγή 3.3: Αναλογία συνόλων 50/50 | 125 |
| 8.1.3.4 | Σύνοψη Satellite Images για τον BP | 126 |
| 8.1.4 | Προσομοιώσεις στο πρόβλημα “MNIST” μέσω Backpropagation | 126 |
| 8.1.4.1 | Παραλλαγή 4.1: 10000 training – 10000 testing datasets | 126 |
| 8.1.4.2 | Παραλλαγή 4.2: 30000 training – 10000 testing datasets | 127 |
| 8.1.4.3 | Παραλλαγή 4.3: 60000 training – 10000 testing datasets | 128 |
| 8.1.4.4 | Σύνοψη MNIST για τον BP | 129 |
| 8.2 | Προσομοιώσεις σε προβλήματα Ταξινόμησης μέσω ELM | 129 |
| 8.2.1 | Προσομοιώσεις στο πρόβλημα “IRIS” μέσω ELM | 129 |
| 8.2.1.1 | Παραλλαγή 1.1: Αναλογία συνόλων 70/30 | 130 |
| 8.2.1.2 | Παραλλαγή 1.2: Αναλογία συνόλων 50/50 | 130 |
| 8.2.2 | Προσομοιώσεις στο πρόβλημα “IMAGE SEGMENT” μέσω ELM | 131 |
| 8.2.2.1 | Παραλλαγή 2.1: Αναλογία συνόλων 80/20 | 131 |
| 8.2.2.2 | Παραλλαγή 2.2: Αναλογία συνόλων 65/35 | 132 |
| 8.2.2.3 | Παραλλαγή 2.3: Αναλογία συνόλων 50/50 | 132 |
| 8.2.2.4 | Σύνοψη Image Segment για τον ELM | 133 |
| 8.2.3 | Προσομοιώσεις στο πρόβλημα “SATELLITE IMAGES” μέσω ELM | 133 |
| 8.2.3.1 | Παραλλαγή 3.1: Αναλογία συνόλων 80/20 | 134 |
| 8.2.3.2 | Παραλλαγή 3.2: Αναλογία συνόλων 68/32 | 134 |
| 8.2.3.3 | Παραλλαγή 3.3: Αναλογία συνόλων 50/50 | 135 |
| 8.2.3.4 | Σύνοψη Satellite Images για τον ELM | 135 |
| 8.2.4 | Προσομοιώσεις στο πρόβλημα “MNIST” μέσω ELM | 136 |
| 8.2.4.1 | Παραλλαγή 4.1: 10000 training – 10000 testing datasets | 136 |
| 8.2.4.2 | Παραλλαγή 4.2: 30000 training – 10000 testing datasets | 137 |
| 8.2.4.3 | Παραλλαγή 4.3: 60000 training – 10000 testing datasets | 137 |
| 8.3 | Αποτίμηση – Σύγκριση αποτελεσμάτων Ταξινόμησης | 138 |
| 8.3.1 | Το πρόβλημα ταξινόμησης των παρατηρήσεων της Ίριδας | 138 |
| 8.3.1.1 | Παραλλαγή 1.1: Σύγκριση ELM και BP - IRIS - 70/30 | 139 |
| 8.3.1.2 | Παραλλαγή 1.2: Σύγκριση ELM και BP - 50/50 | 139 |
| 8.3.2 | Το πρόβλημα ταξινόμησης με αναγνώριση αντικειμένων | 140 |
| 8.3.2.1 | Παραλλαγή 2.1: Σύγκριση ELM και BP - IMAGE SEGMENT - 80/20 | 140 |
| 8.3.2.2 | Παραλλαγή 2.2: Σύγκριση ELM και BP - IMAGE SEGMENT - 65/35 | 141 |
| 8.3.2.3 | Παραλλαγή 2.3: Σύγκριση ELM και BP - IMAGE SEGMENT - 50/50 | 141 |

| | | |
|--------------------------------|--|------------|
| 8.3.3 | Η αναγνώριση της μορφολογίας του εδάφους από δορυφορικές λήψεις | 142 |
| 8.3.3.1 | Παραλλαγή 3.1: Σύγκριση ELM και BP - SATELLITE IMAGES - 80/20 | 142 |
| 8.3.3.2 | Παραλλαγή 3.2: Σύγκριση ELM και BP - SATELLITE IMAGES - 68/32 | 143 |
| 8.3.3.3 | Παραλλαγή 3.3: Σύγκριση ELM και BP - SATELLITE IMAGES - 50/50 | 143 |
| 8.3.4 | Η αναγνώριση των χειρόγραφων προτύπων ψηφίων | 144 |
| 8.3.4.1 | Παραλλαγή 4.1: Σύγκριση ELM και BP - MNIST – 10000 | 144 |
| 8.3.4.2 | Παραλλαγή 4.2: Σύγκριση ELM και BP - MNIST – 30000 | 144 |
| 8.3.4.3 | Παραλλαγή 4.3: Σύγκριση ELM και BP - MNIST – 60000 | 145 |
| Κεφάλαιο 9 Συνοψίζοντας | | 146 |
| Βιβλιογραφία | | 148 |

Κατάλογος Εικόνων

| | | |
|-----------|--|-----|
| Εικόνα 1 | Ο βιολογικός Νευρώνας | 23 |
| Εικόνα 2 | Το μοντέλο των McCulloch και Pitts για τον νευρώνα | 25 |
| Εικόνα 3 | Perceptron: Το πιο απλό μοντέλο τεχνητού νευρώνα | 29 |
| Εικόνα 4 | ADALINE | 32 |
| Εικόνα 5 | MADALINE | 33 |
| Εικόνα 6 | Πολυστρωματικό νευρωνικό δίκτυο με χρήση Backpropagation αλγορίθμου εκμάθησης | 35 |
| Εικόνα 7 | RNN - FFNN | 36 |
| Εικόνα 8 | RBF | 37 |
| Εικόνα 9 | SVM margin | 41 |
| Εικόνα 10 | Boltzmann Machine | 42 |
| Εικόνα 11 | Associative Neural Networks (ASNN) | 44 |
| Εικόνα 12 | Μετατροπή πολυπλεγμένου σήματος στο σήμα πηγής με ICA μέθοδο | 48 |
| Εικόνα 13 | Τοπολογία δικτύου SOM (Self-Organizing Map) | 51 |
| Εικόνα 14 | Αρχιτεκτονική LVQ | 51 |
| Εικόνα 15 | Εφαρμογές του δικτύου ART | 53 |
| Εικόνα 16 | Τοπολογία δικτύου ενός επιπέδου Hopfield με n νευρώνες | 55 |
| Εικόνα 17 | Συσχετιστικές Μνήμες A | 57 |
| Εικόνα 18 | Συσχετιστικές Μνήμες B | 57 |
| Εικόνα 19 | Τοπολογία ELM | 77 |
| Εικόνα 20 | Νευρωνικό δίκτυο τριών στρωμάτων | 79 |
| Εικόνα 21 | Η μορφή Κινέζικου χαρακτήρα με – χωρίς θόρυβο (No – Prop) | 80 |
| Εικόνα 22 | Η μορφή της $y = \sin(x)$ στο διάστημα $[-5, +5]$ | 88 |
| Εικόνα 23 | Rprop & LM με καθαρά δεδομένα | 90 |
| Εικόνα 24 | Σύνολα τιμών με μολυσμένα δεδομένα | 91 |
| Εικόνα 25 | Rprop & LM με μολυσμένα δεδομένα | 93 |
| Εικόνα 26 | Σύνολα δεδομένων με αποκομμένα διαστήματα εκπαίδευσης | 94 |
| Εικόνα 27 | Παραλλαγή 3.1: Rprop - LM με αποκομμένα διαστήματα εκπαίδευσης | 95 |
| Εικόνα 28 | Παραλλαγή 3.2: Rprop - LM με αποκομμένα διαστήματα εκπαίδευσης | 96 |
| Εικόνα 29 | Παραλλαγή 3.3: Rprop - LM με αποκομμένα διαστήματα εκπαίδευσης | 97 |
| Εικόνα 30 | Παραλλαγή 3.3: Rprop - LM με αποκομμένα διαστήματα εκπαίδευσης | 98 |
| Εικόνα 31 | Σύνολα τιμών με μολυσμένα δεδομένα και αποκομμένα διαστήματα | 99 |
| Εικόνα 32 | Αποτελέσματα LM με μολυσμένα δεδομένα και αποκοπή διαστημάτων | 99 |
| Εικόνα 33 | ELM με καθαρά δεδομένα | 102 |
| Εικόνα 34 | ELM με μολυσμένα δεδομένα | 105 |
| Εικόνα 35 | ELM με αποκοπή διαστημάτων στο πεδίο ορισμού $[-5, +5]$ | 107 |
| Εικόνα 36 | ELM με αποκοπή διαστημάτων στο πεδίο ορισμού $[-10, +10]$ | 108 |
| Εικόνα 37 | Διαγράμματα με μολυσμένα δεδομένα και μη και αποκομμένα διαστήματα | 109 |
| Εικόνα 38 | ELM με αποκοπή διαστημάτων και δεδομένα με θόρυβο | 110 |
| Εικόνα 39 | Συγκριτικές Απεικονίσεις BP - ELM με Καθαρά Δεδομένα | 112 |
| Εικόνα 40 | Συγκριτικές Απεικονίσεις BP - ELM με Μολυσμένα Δεδομένα | 113 |
| Εικόνα 41 | Συγκριτικές Απεικονίσεις BP - ELM με Αποκοπή Υποδιαστημάτων | 114 |
| Εικόνα 42 | Συγκριτικές Απεικονίσεις BP - ELM με Αποκοπή Υποδιαστημάτων σε Μολυσμένα Δεδομένα | 115 |

Κατάλογος Πινάκων

| | |
|--|-----|
| Πίνακας 1 Συνάψεις εγκεφάλων | 24 |
| Πίνακας 2 Συναρτήσεις Ακτινικής Βάσης (RBF) | 39 |
| Πίνακας 3 Μαρκοβιανής Αλυσίδας | 43 |
| Πίνακας 4 Jacobian | 68 |
| Πίνακας 5 Διακύμανση Τιμών RMSE | 87 |
| Πίνακας 6 1.1 Αποτελέσματα BP με Καθαρά Δεδομένα | 89 |
| Πίνακας 7 1.2 Αποτελέσματα BP με Καθαρά Δεδομένα | 89 |
| Πίνακας 8 1.3 Αποτελέσματα BP με Καθαρά Δεδομένα | 90 |
| Πίνακας 9 2.1 Αποτελέσματα BP με Μολυσμένα Δεδομένα | 92 |
| Πίνακας 10 2.2 Αποτελέσματα BP με Μολυσμένα Δεδομένα | 92 |
| Πίνακας 11 2.3 Αποτελέσματα BP με Μολυσμένα Δεδομένα | 93 |
| Πίνακας 12 3.1 Αποτελέσματα BP με Αποκοπή Υποσυνόλων | 94 |
| Πίνακας 13 3.2 Αποτελέσματα BP με Αποκοπή Υποσυνόλων | 95 |
| Πίνακας 14 3.3 Αποτελέσματα BP με Αποκοπή Υποσυνόλων | 97 |
| Πίνακας 15 3.4 Αποτελέσματα BP με Αποκοπή Υποσυνόλων | 97 |
| Πίνακας 16 4.1 Αποτελέσματα BP με Αποκοπή Υποσυνόλων σε Μολυσμένα δεδομένα | 99 |
| Πίνακας 17 1.1 Αποτελέσματα ELM με Καθαρά Δεδομένα | 100 |
| Πίνακας 18 1.2 Αποτελέσματα ELM με Καθαρά Δεδομένα | 101 |
| Πίνακας 19 1.3 Αποτελέσματα ELM με Καθαρά Δεδομένα | 101 |
| Πίνακας 20 2.1 Αποτελέσματα ELM με Μολυσμένα Δεδομένα | 103 |
| Πίνακας 21 2.2 Αποτελέσματα ELM με Μολυσμένα Δεδομένα | 103 |
| Πίνακας 22 2.3 Αποτελέσματα ELM με Μολυσμένα Δεδομένα | 104 |
| Πίνακας 23 3.1 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων | 106 |
| Πίνακας 24 3.2 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων | 106 |
| Πίνακας 25 3.3 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων | 107 |
| Πίνακας 26 3.4 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων | 109 |
| Πίνακας 27 4.1 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων σε Μολυσμένα δεδομένα | 109 |
| Πίνακας 28 Συγκριτικά Αποτελέσματα BP - ELM με Καθαρά Δεδομένα | 111 |
| Πίνακας 29 Συγκριτικά Αποτελέσματα BP - ELM με Μολυσμένα Δεδομένα | 112 |
| Πίνακας 30 Συγκριτικά Αποτελέσματα BP - ELM με Αποκοπή Υποδιαστημάτων | 114 |
| Πίνακας 31 Συγκριτικά Αποτελέσματα BP - ELM με Αποκοπή Υποδιαστημάτων | 115 |
| Πίνακας 32 1.1 Αποτελέσματα BP - IRIS - 70/30 | 118 |
| Πίνακας 33 1.2 Αποτελέσματα BP - IRIS - 50/50 | 119 |
| Πίνακας 34 2.1 Αποτελέσματα BP - IMAGE SEGMENT - 80/20 | 120 |
| Πίνακας 35 2.2 Αποτελέσματα BP - IMAGE SEGMENT - 65/35 | 121 |
| Πίνακας 36 2.3 Αποτελέσματα BP - IMAGE SEGMENT - 50/50 | 122 |
| Πίνακας 37 3.1 Αποτελέσματα BP - SATELLITE IMAGES - 80/20 | 123 |
| Πίνακας 38 3.2 Αποτελέσματα BP - SATELLITE IMAGES - 68/32 | 124 |
| Πίνακας 39 3.3 Αποτελέσματα BP - SATELLITE IMAGES - 50/50 | 125 |
| Πίνακας 40 4.1 Αποτελέσματα BP - MNIST - 10000 | 127 |
| Πίνακας 41 4.2 Αποτελέσματα BP - MNIST - 30000 | 128 |
| Πίνακας 42 4.3 Αποτελέσματα BP - MNIST - 60000 | 128 |
| Πίνακας 43 1.1 Αποτελέσματα ELM - IRIS - 70/30 | 130 |
| Πίνακας 44 1.2 Αποτελέσματα ELM - IRIS - 50/50 | 130 |

| | | |
|-------------------|--|-----|
| Πίνακας 45 | 2.1 Αποτελέσματα ELM – IMAGE SEGMENT – 80/20 | 131 |
| Πίνακας 46 | 2.2 Αποτελέσματα ELM – IMAGE SEGMENT – 65/35 | 132 |
| Πίνακας 47 | 2.3 Αποτελέσματα ELM – IMAGE SEGMENT – 50/50 | 133 |
| Πίνακας 48 | 3.1 Αποτελέσματα ELM – SATELLITE IMAGES – 80/20 | 134 |
| Πίνακας 49 | 3.2 Αποτελέσματα ELM – SATELLITE IMAGES – 68/32 | 135 |
| Πίνακας 50 | 3.3 Αποτελέσματα ELM – SATELLITE IMAGES – 50/50 | 135 |
| Πίνακας 51 | 4.1 Αποτελέσματα BP – MNIST - 10000 | 136 |
| Πίνακας 52 | 4.2 Αποτελέσματα BP – MNIST - 30000 | 137 |
| Πίνακας 53 | 4.3 Αποτελέσματα BP – MNIST - 60000 | 137 |
| Πίνακας 54 | 1.1 Σύγκριση ELM και BP - IRIS - 70/30 | 139 |
| Πίνακας 55 | 1.2 Σύγκριση ELM και BP - IRIS - 50/50 | 139 |
| Πίνακας 56 | 2.1 Σύγκριση ELM και BP - IMAGE SEGMENT - 80/20 | 140 |
| Πίνακας 57 | 2.2 Σύγκριση ELM και BP - IMAGE SEGMENT - 65/35 | 141 |
| Πίνακας 58 | 2.3 Σύγκριση ELM και BP - IMAGE SEGMENT - 50/50 | 141 |
| Πίνακας 59 | 3.1 Σύγκριση ELM και BP - SATELLITE IMAGES - 80/20 | 142 |
| Πίνακας 60 | 3.2 Σύγκριση ELM και BP - IMAGE SEGMENT - 68/32 | 143 |
| Πίνακας 61 | 3.3 Σύγκριση ELM και BP - SATELLITE IMAGES - 50/50 | 143 |
| Πίνακας 62 | 4.1 Σύγκριση ELM και BP - MNIST - 10000 | 144 |
| Πίνακας 63 | 4.2 Σύγκριση ELM και BP - MNIST - 30000 | 145 |
| Πίνακας 64 | 4.3 Σύγκριση ELM και BP - MNIST - 60000 | 145 |

Συντομογραφίες – Αρκτικόλεξο

| | |
|------------------|---|
| ADALINE | AD Aptive LI Near E lement |
| MADALINE | M any AD Aptive LI Near E lement |
| RBP | R ecursive B ack p ropagation |
| RBF | R adial B asis F unction Ακτινικής Βάσης Συνάρτηση |
| MLP | M ultiple L ayer P erceptron |
| SVM | S upport V ector M achine |
| ANN – TNA | A rtificial N eural N etwork - Τεχνητά Νευρωνικά Δίκτυα |
| ASNN | A ssociative N eural N etworks |
| FFNN | F eed- F orward N eural N etworks |
| KNN | K - N earest N eighbor T echnique |
| PCA | P rincipal C omponent A nalysis |
| APEX | A daptive P rincipal component E Xtraction |
| GHA | G eneralized H ebbian A lgorithm |
| ICA | I ndependent C omponent A nalysis |
| GAN | G enerative A dversarial N etworks |
| SOM | S elf- O rganizing M ap |
| LVQ | L earning V ector Q uantization |
| ART | A daptive R esonance T heory |
| AM | A ssociative M emories Συσχετιστικές Μνήμες |
| CAM | C ontent A ssociative M emory |
| BSB | B rain S tate i n a B ox |
| MSE | M ean S quared E rror Μέσο Τετραγωνικό Σφάλμα |
| RMSE | R oot M ean S quare E rror Ρίζα Μέσου Τετραγωνικού Σφάλματος |
| LR | L earning R ate Ρυθμός Εκμάθησης |
| RNN | R ecursive N eural N etwork |
| BP | B ack p ropagation |
| Rprop | R esilient B ack p ropagation |
| LM | L evenberg – M arquardt |
| SCG | S caled C onjugate G radient |
| ELM | E xtrême L earning M achine Ακράιες Μηχανές Μάθησης |
| AF | A ctivation F unction Συνάρτηση Ενεργοποίησης |
| Tr. | T raining |
| Ts. | T esting |
| Acc. | A ccuracy Ακρίβεια – Ευστοχία |
| H - EML | H uang – E xtrême L earning M achine |
| H - BP | H uang - B ack p ropagation |

Κεφάλαιο 1

Τα Τεχνητά Νευρωνικά Δίκτυα

Η έρευνα η οποία σχετίζεται με τα **Τεχνητά Νευρωνικά Δίκτυα (ΤΝΔ)** έχει προέλθει από την έμπνευση πάνω στη συγκρότηση και τη λειτουργία του ανθρώπινου εγκεφάλου. Το στοιχείο που επικρατεί στον ανθρώπινο εγκέφαλο είναι οι νευρώνες ή αλλιώς τα νευρωνικά κύτταρα, τα οποία συνεργάζονται μεταξύ τους και είναι υπεύθυνα για τις δομικές και λειτουργικές διεργασίες ολόκληρου του νευρωνικού συστήματος. Το βασικότερο κίνητρο που υπήρξε ήταν η υλοποίηση νευρωνικών δικτύων μέσω ενός υπολογιστικού μοντέλου, το οποίο θα λειτουργεί στα πρότυπα του βιολογικού μοντέλου, αλλά θα απαρτίζεται από τα δομικά υλικά ενός υπολογιστή με απώτερο σκοπό τη μάθηση, τη μνήμη, τη γενίκευση, την ομαδοποίηση, την κατανόηση κτλ.

Τα Τεχνητά Νευρωνικά Δίκτυα κατά συνέπεια διατηρούν μόνο τα κεντρικά χαρακτηριστικά του βιολογικού νευρωνικού δικτύου και έχουν ως σκοπό τη μάθηση, τον παραλληλισμό και την κατανομή της πληροφορίας. Ο σημαντικότερος παράγοντας για να μπορέσει να λειτουργήσει ένα ΤΝΔ είναι η εύρεση αλγορίθμου για την εκπαίδευσή του, καθώς και ο καθορισμός τους στόχου και των κριτηρίων για να κατορθώσει μέσα από τις διεργασίες που θα κάνει να προσφέρει τα επιθυμητά αποτελέσματα.

1.1 Ο Νευρώνας

Το **νευρωνικό κύτταρο** ή **νευρώνας** (βλ. Σχήμα 1) είναι το βασικό δομικό συστατικό τόσο του ανθρώπινου όσο και του ζωικού εγκεφάλου. Πρόκειται για ένα μεγάλο κύτταρο το οποίο κατηγοριοποιείται σε 4 βασικές κατηγορίες οι οποίες είναι:

- α. το σώμα
- β. τους δενδρίτες
- γ. τον άξονα
- δ. τις συνάψεις, που συνδέουν τις διακλαδώσεις του άξονα με τους δενδρίτες άλλων νευρώνων δημιουργώντας έτσι ένα νευρωνικό δίκτυο.

Η κάθε κατηγορία – τμήμα παίζει ξεχωριστό ρόλο στη λειτουργία του νευρωνικού συστήματος:

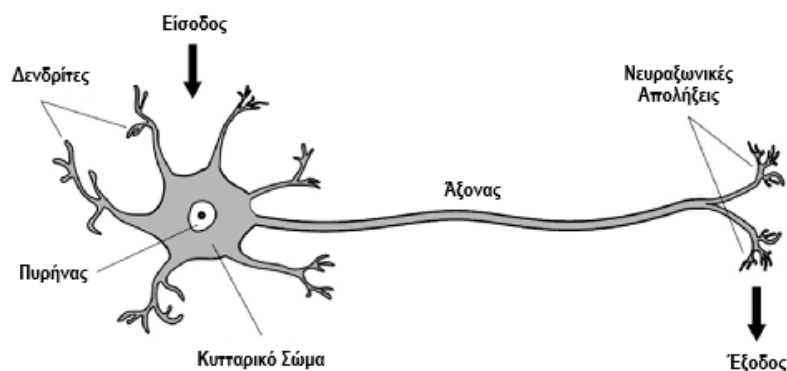
- Οι **δενδρίτες** είναι το σημείο εισόδου του νευρώνα και δέχονται ηλεκτρικά σήματα από άλλους νευρώνες.
- Ο **άξονας** είναι το σημείο εξόδου του νευρώνα, το οποίο μοιάζει με μία μακρόστενη κλωστή η οποία μπορεί να έχει μήκος από κάποια χιλιοστά μέχρι και το 1m. Περιβάλλεται από τη μυελίνη η οποία είναι μία λιπώδη ουσία που παίζει τον ρόλο μία μεμβράνης και είναι υπεύθυνη για την προστασία των νευρωνικών ιών [1, 2]. Ο άξονας είναι υπεύθυνος να μεταφέρει το σήμα του νευρωνικού κυττάρου (ηλεκτρικός παλμός) από μία περιοχή σε μία άλλη. Τέλος, η μυελίνη προσφέρει αρωγή για την ταχύτερη μεταφορά του σήματος ηλεκτρονικού παλμού μεταξύ των νευρώνων.
- Οι **συνάψεις** είναι τα σημεία ένωσης μεταξύ διακλαδώσεων του άξονα ενός νευρώνα με τους δενδρίτες από άλλους νευρώνες φέρνοντάς τα σε πολύ κοντινή απόσταση. Πρόκειται για κύστες οι οποίες περιέχουν ηλεκτροχημικά υλικά, με τα βασικότερα να είναι το Κάλιο και το Νάτριο (Ka^+ , Na^+). Το υλικό αυτό μεταφέρει την πληροφορία από τον αποστολέα – άξονα στους παραλήπτες – δενδρίτες η οποία διαδικασία ονομάζεται ηλεκτρική δραστηριότητα. Οι συνάψεις αποτελούνται από τρία μέρη:

- Το προσυναπτικό άκρο που περιέχει νευροδιαβιβαστές
- Τη συναπτική σχισμή μεταξύ των δύο νευρικών κυττάρων
- Το μετασυναπτικό άκρο που περιέχει θέσεις υποδοχέων

Ταυτόχρονα υπάρχουν δύο τύποι συνάψεων:

- Χημικές συνάψεις
- Ηλεκτρικές συνάψεις

Το ποσοστό της ηλεκτρικής δραστηριότητας που μεταδίδεται τελικά στον δενδρίτη ονομάζεται **συναπτικό βάρος**. Οι συνάψεις χωρίζονται επίσης σε **ανασταλτικές** (inhibitory) και **ενισχυτικές** (excitatory) ανάλογα με τον αν το φορτίο που εκλύεται από τη σύναψη αν τον εμποδίζει να παράγει παλμούς ή αν τον ερεθίζει να παράξει παλμούς με μεγαλύτερη συχνότητα [3].



Εικόνα 1 Ο βιολογικός Νευρώνας

1.1.1 Η λειτουργία του βιολογικού νευρώνα

Στους βιολογικούς νευρώνες για να μπορέσει να μεταφερθεί η πληροφορία αξιοποιούνται οι ηλεκτρικοί παλμοί που ταξιδεύουν στον άξονα κάθε νευρώνα και μέσω των συνάψεων διαδίδονται μέσα στους δενδρίτες των παραληπτών νευρώνων. Ο καθένας νευρώνας συγκεντρώνει όλο το ηλεκτρικό φορτίο που δέχεται από κάθε σύναψη στους δενδρίτες του ζυγίζοντας το εισερχόμενο φορτίο με το αντίστοιχο συναπτικό βάρος. Έτσι, όσο πιο ισχυρή είναι η συναπτική ζεύξη τόσο περισσότερο συμμετέχει το συγκεκριμένο φορτίο στο συνολικό άθροισμα εισόδου. Αν το συνολικό φορτίο ξεπερνάει κάποιο κατώφλι, τότε ο άξονας ξεκινάει να παράγει ηλεκτρικούς παλμούς με μεγάλη συχνότητα. Αν όμως το συνολικό φορτίο δεν περνάει το κατώφλι τότε παράγει πολύ πιο αραιά παλμούς και μένει αδρανής. Ο κάθε παλμός έχει συγκεκριμένο χρονικό πλάτος t_p και μετά από κάθε παλμό ο νευρώνας χρειάζεται συγκεκριμένο χρόνο ανάπαυσης t_r . Τελικά οι παλμοί που παράγονται ταξιδεύουν κατά μήκος του άξονα και τροφοδοτούν άλλους νευρώνες οι οποίοι είναι συνδεδεμένοι με τον άξονα.

1.1.2 Στοιχεία των εγκεφάλων

Για να μπορέσουμε να κατανοήσουμε περαιτέρω το πόσο μεγάλο και συμπακνωμένο ταυτόχρονα είναι το νευρωνικό σύστημα ενός ανθρώπου, αρκεί να αναφέρουμε ότι το πλήθος των νευρώνων στον ανθρώπινο εγκέφαλο είναι $10^{11} - 10^{12}$ περίπου, δηλαδή μπορεί να φτάσει το 1 τρισεκατομμύριο! Επιπρόσθετα, οι συνάψεις που μπορεί να έχει ο κάθε νευρώνας κυμαίνονται από $10^3 - 10^4$, δηλαδή ο

κάθε νευρώνας στέλνει την έξοδό του σε 1000 έως 10.000 νευρώνες που είναι γειτονικοί σε αυτόν. Ο μέσος όρος μετάδοσης των παλμών είναι 1 – 10 ms, με την ταχύτητα του εγκεφάλου να φτάνει περίπου στις 10^{17} συνάψεις/sec. Τέλος, για να μπορέσει να αναγνωρίσει μία σκηνή χρειάζονται περίπου 100 ms κάτι που συνεπάγεται πως απαιτούνται λιγότερα από 100 βήματα για την εξαγωγή συμπερασμάτων.

Μερικές χρήσιμες συγκρίσεις έχουν να κάνουν σχέση με τη διαφορά του όγκο των συνάψεων και της ταχύτητας που έχει ο ανθρώπινος εγκέφαλος σε σχέση ορισμένα μικρά ζώα όπως φαίνεται και στο πιο κάτω πίνακα.

| | Βδέλλα | Σκουλήκι | Μύγα | Απλυσία | Κατσαρίδα | Μέλισσα | Ανθρώπος |
|---------------------------|---------|----------|--------|------------|------------|------------|-----------|
| Αριθμός Συνάψεων | $<10^3$ | $<10^4$ | 10^8 | $<10^9$ | 10^9 | $<10^{10}$ | 10^{15} |
| Συνάψεις ανά Δευτερόλεπτο | $>10^4$ | $>10^5$ | 10^9 | $>10^{10}$ | $>10^{10}$ | $>10^{11}$ | 10^{17} |

Πίνακας 1 Συνάψεις εγκεφάλων

1.1.3 Το μοντέλο McCulloch – Pitts

Κατά τη δεκαετία του 1940 ξεκίνησε ένα ρεύμα προς τη δραστηριοποίηση στη μελέτη των βιολογικών νευρωνικών δικτύων και τη μαθηματική υλοποίησή τους. Οι δύο επιστήμονες που επέμειναν και επικεντρώθηκαν παραπάνω σε αυτή τη μελέτη ήταν ο νευροψυχίατρος Warren McCulloch (1898 -1969) και ο μαθηματικός Walter Pitts (1923 – 1969) οι οποίοι παρουσίασαν το 1943 τις πρώτες εφαρμογές ενός λογικού λογισμού στα στοιχεία του βιολογικού συστήματος. Βασισμένο στον κανόνα “all – or – none”, δηλαδή «όλα ή κανένα» της νευρικής δραστηριότητας, κατασκεύασαν μια λογική πύλη BOOLEAN, για να περιγράψουν τα νευρικά συμβάντα και τις σχέσεις τους. Η ιδέα αυτή τους οδήγησε να εξιδανικεύσουν τους νευρώνες σαν συσκευές οι οποίες είναι “on – off ” που σημαίνει ότι είτε μπορούν να πυροδοτήσουν ή να μην κάνουν απολύτως τίποτα. Φέρνοντας τις συγκεκριμένες καταστάσεις σε αντιπαράθεση με τη φύση των προτάσεων λογικής «αληθής – ψευδής», οι McCulloch και Pitts κατασκεύασαν υποθετικά δίκτυα διεγερτικών και ανασταλτικών νευρώνων, με ποικίλα μοτίβα συνδέσεων και κατέδειξαν ένα ισομορφισμό μεταξύ αυτών των υποθετικών διατάξεων των νευρωνίων και τη λογική των προτάσεων [4].

Η έννοια του δικτύου του McCulloch – Pitts έχει μείνει στην ιστορία της κυβερνητικής ως ένα ορόσημο, με απώτερο σκοπό εκείνη την εποχή να είναι η εύρεση κοινών στοιχείων μεταξύ της λειτουργίας των μηχανών και των ζώων. Οι εργασίες οι οποίες έκαναν είχε εννοιολογικές συνδέσεις και με τη μηχανή του Alan Turing (1912 – 1954) η οποία και αυτή με τη σειρά της ήταν εξίσου σημαντική στη μεταγενέστερη δουλειά του John von Neumann (1903 – 1957).

Σχετικά με τον νευρώνα των McCulloch – Pitts η κατάσταση του περιγράφεται από ένα δυαδικό y ,

$$\begin{aligned} y = 0 &\rightarrow \text{ο νευρώνας είναι αδρανής (δεν πυροδοτεί)} \\ y = 1 &\rightarrow \text{ο νευρώνας πυροδοτεί στη μέγιστη συχνότητα} \end{aligned}$$

Οι συνάψεις περιγράφονται από τα **συναπτικά βάρη** (*synaptic weights*) w_i που είναι πραγματικοί αριθμοί, θετικοί για τις ενισχυτικές συνάψεις και αρνητικοί για τις κατασταλτικές συνάψεις. Αν x_1, x_2, \dots, x_n είναι οι είσοδοι του νευρώνα τότε το άθροισμα u του φορτίου που δέχεται ο νευρώνας είναι απλά

$$u = \sum_{i=1}^n w_i x_i \quad (1)$$

Αν το άθροισμα u είναι μεγαλύτερο από το **κατώφλι** (*threshold*) θ τότε ο νευρώνας πυροδοτεί, διαφορετικά παραμένει αδρανής. Χρησιμοποιώντας μαθηματικά γράφουμε

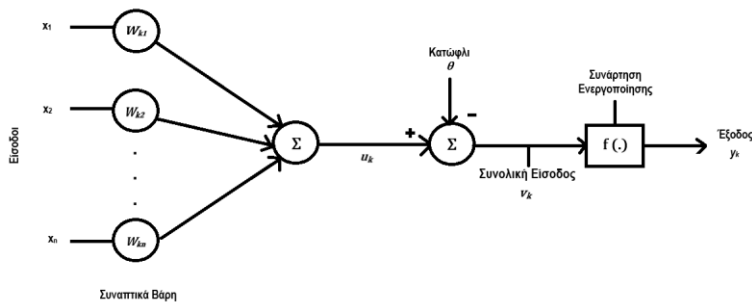
$$y = f(u - \theta)$$

όπου $f(\cdot)$ είναι η λεγόμενη **βηματική συνάρτηση**

Βηματική συνάρτηση 0/1 (step function 0/1)

$$f(x) = \begin{cases} 0, & u \leq 0 \\ 1, & u > 0 \end{cases}$$

Το παραπάνω μοντέλο περιγράφεται σχηματικά ακολούθως, το οποίο καταδεικνύει το μαθηματικό μοντέλο από έναν αθροιστή σε ένα μη – γραμμικό μετασχηματισμό f (βλ. Σχήμα 1.2)



Εικόνα 2 Το μοντέλο των McCulloch και Pitts για τον νευρώνα

Το κατώφλι θ είναι ένα πραγματικός αριθμός (θετικός ή αρνητικός) όπως είναι και τα συναπτικά βάρη $w_{k1}, w_{k2}, \dots, w_{kn}$. Κατ' αυτή τη συνθήκη το κατώφλι μπορεί να θεωρηθεί ως ένα επιπλέον συναπτικό βάρος συνδεδεμένο με μία σταθερή είσοδο x_0 η οποία έχει πάντα την τιμή -1 [3]. Έτσι θα μπορούσαμε να γράψουμε

$$u = \sum_{i=1}^n w_i x_i - \theta = \sum_{i=0}^n w_i x_i \quad (2)$$

όπου $w_0 = \theta$ και $x_0 = -1$

1.1.4 Άλλα διαδεδομένα μοντέλα

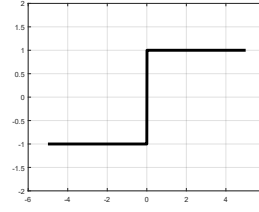
Πέρα από το απλό και διαδεδομένο μοντέλο για τον νευρώνα των McCulloch – Pitts υπάρχουν και άλλα μοντέλα εξίσου σημαντικά τα οποία αναπτύχθηκαν και εξελίχθηκαν μέσα στα χρόνια. Η πιο σημαντική διαφορά που έχουν αυτά τα μοντέλα σε σχέση με το McCulloch – Pitts είναι η μορφή της μη γραμμικής συνάρτησης $f(\cdot)$ που χρησιμοποιείται στην έξοδο. Η συνάρτηση αυτή που καλείται και **συνάρτηση ενεργοποίησης του νευρώνα (neuron activation function)** έχει τη δυνατότητα να πάρει της παρακάτω μορφές:

Βηματική -1/1 (step function -1/1)

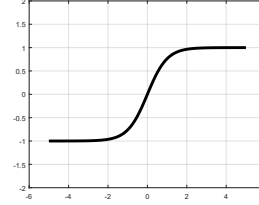
$$f(u) = -1, \text{ αν } u \leq 0$$

$$f(u) = 1, \text{ αν } u > 0$$

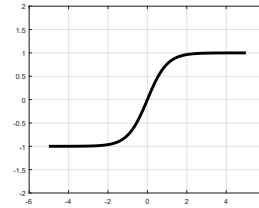
Η συνάρτηση αυτή συμβολίζεται επίσης με sgn από το λατινικό “signum” = “πρόσημο”

Σιγμοειδής (sigmoid)

$$f(u) = -1 / (1 + e^{-u})$$

Υπερβολική Εφαπτομένη (hyperbolic tangent)

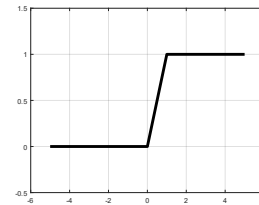
$$f(u) = \tanh(u) = (1 - e^{-u}) / (1 + e^{-u})$$

Συνάρτηση Κατωφλίου (threshold function)

$$f(u) = 0, \text{ αν } u \leq 0$$

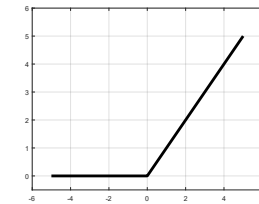
$$f(u) = u, \text{ αν } 0 < u < 1$$

$$f(u) = 1, \text{ αν } u \geq 1$$

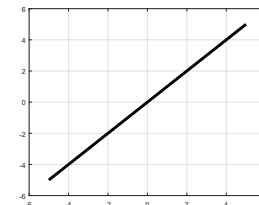
Συνάρτηση Ράμπας (ramp function)

$$f(u) = 0, \text{ αν } u \leq 0$$

$$f(u) = u, \text{ αν } u > 0$$

Γραμμική (linear)

$$f(u) = u$$



1.2 Εκπαίδευση νευρωνικών δικτύων

Ένας ανθρώπινος εγκέφαλος απαρτίζεται από 100 δισεκατομμύρια νευρώνες για τους οποίους αντιστοιχούν 1000 συνάψεις στον καθένα. Λόγω της μεγάλης πολυπλοκότητας και των πολλαπλών συνδέσεων που έχουν τα νευρώνια μεταξύ τους, μπορούν να φέρουν εις πέρας πολλές λειτουργίες και διεργασίες ταυτόχρονα με σκοπό να επιτευχθεί η **νοημοσύνη**. Οι λειτουργίες ποικίλουν και μπορούν να αφορούν πολύ απλές και προφανείς λειτουργίες, ως και συνδυασμό των απλούστερων λειτουργιών για την επίτευξη μια πιο δύσκολης λειτουργίας ύστερα από μεθοδική ακολουθία βημάτων. Κάποιες από τις λειτουργίες είναι οι ακόλουθες:

- Η αναγνώριση γνώριμων ή σχετικών εικόνων (πρόσωπα, αντικείμενα, παραστάσεις κτλ.).
- Η μνήμη, η οποία χρησιμοποιείται για να γίνει ανάκληση παλαιότερων καταστάσεων, συνθηκών, εικόνων κτλ.
- Η αναγνώριση ήχων, είτε πρόκειται για την κατανόηση είτε για την παραγωγή και την επικοινωνία.
- Η λήψη αποφάσεων.
- Η συλλογιστική και κρίση σε διάφορες καταστάσεις ή συνθήκες.
- Η μεθοδική σκέψη για την επίλυση κάποιου προβλήματος ή της αντιμετώπισης κάποιας κατάστασης.
- Η μάθηση και η αυτοπροσαρμογή σε κάποιον νέο περιβάλλον ή νέες καταστάσεις που δεν είναι γνώριμες.

Η μάθηση ουσιαστικά αποτελεί το πιο σημαντικό κομμάτι όλων των προηγούμενων, καθώς είναι αυτή που σταδιακά κατακτιέται και επαναχρησιμοποιείται ώστε να κατακτηθεί κάποια μεταγενέστερη γνώση. Ωστόσο τη δυνατότητα της μάθησης δεν την έχουν μόνο οι άνθρωποι αλλά όλα τα ζώα. Εκτός της μάθησης που είναι το πιο σημαντικό κομμάτι για τη νοημοσύνη, όλες οι υπόλοιπες λειτουργίες είναι προϊόν κατανόησης κατά την ανάπτυξη των οργανισμών από την εμβρυακή έως και την παιδική ηλικία, οι οποίες με το πέρασμα των χρόνων βελτιστοποιούνται. Το σημαντικότερο σημείο για την Τεχνητή Νοημοσύνη (TN) είναι η ύπαρξη κάποιας βάσης πάνω στην οποία μπορούν να διεκπεραιωθούν όλες οι παραπάνω λειτουργίες. Για τον άνθρωπο και για τα ζώα αυτή η βάση είναι οι νευρώνες καθώς και η δομή μαζί με πολυπλοκότητα που έχουν. Για να μπορέσουμε να μεταβούμε στο κομμάτι της Τεχνητής Νοημοσύνης των Δικτύων στους υπολογιστές, θα χρειαστούν κάποια υλικά τεχνητά, όπως για παράδειγμα ημιαγωγή, πυρίτιο για την υλοποίηση τρανζίστορ, πυκνωτές κτλ. Συνεπώς για να εφαρμόσουμε την Τεχνητή Νοημοσύνη πάνω στους υπολογιστές χρειάζεται να ικανοποιήσουμε δύο ξεχωριστές συνθήκες.

Η πρώτη συνθήκη είναι η ανάπτυξη της βάσης πάνω στην οποία θα διεκπεραιωθούν οι λειτουργίες, όπως στον άνθρωπο είναι ο νευρώνας. Άρα όπως προαναφέρθηκε τα υλικά αντικείμενα των υπολογιστών παίζουν αυτόν ακριβώς τον ρόλο σαν τους νευρώνες.

Η δεύτερη συνθήκη είναι η ανακάλυψη, δόμηση και εκτέλεση αλγορίθμων που θα αξιοποιήσουν το υλικό που υφίσταται για την εκπαίδευση του δικτύου. Ο κατάλληλος αλγόριθμος για να μπορέσει να ολοκληρώσει την εκπαίδευση θα πρέπει να έχει οδηγήσει το δίκτυο στην κατανόηση των καταστάσεων με τέτοιο τρόπο που να μπορεί να αποκτήσει την απαραίτητη πείρα. Έτσι με τη σειρά του το δίκτυο θα μπορέσει να ανταπεξέλθει σε παρόμοιες καταστάσεις και να λύσει τα προβλήματα που θα του προκύψουν, ακολουθώντας τη συλλογιστική των βημάτων της “νοημοσύνης” που έχει αποκτήσει.

Εν κατακλείδι, το αντικείμενο με το οποίο ασχολούνται τα Τεχνητά Νευρωνικά Δίκτυα είναι η ανάπτυξη και η μελέτη μαθηματικών αλγορίθμων βασισμένα στην αντίστοιχη αρχιτεκτονική και πρότυπα όπως και τα βιολογικά νευρωνικά δίκτυα.

Κεφάλαιο 2

Οι Νευρωνικοί Αλγόριθμοι

Όπως είδαμε στο προηγούμενο κεφάλαιο, για να μπορέσει το Τεχνητό Νευρωνικό δίκτυο να αποκτήσει νοημοσύνη, αναγκαίο συστατικό είναι η ανάπτυξη αλγορίθμων που θα το βοηθήσουν να εκπαιδευτεί, να προσαρμόζεται στο πρόβλημα, να κατανοεί πότε πρέπει να αντιδρά και πότε να μένει αδρανές και τέλος εφόσον έχει αποκτήσει γνώση να εκτελεί τις διεργασίες που του αναθέτονται με επιτυχία.

Σε αυτό το κεφάλαιο θα αναφερθούμε συντόμως στις γενικές κατηγορίες των αλγορίθμων εξηγώντας τη χρησιμότητά τους, τη δομή τους και τις λειτουργίες τους.

2.1 Ταξινόμηση νευρωνικών αλγορίθμων

Οι νευρωνικοί αλγόριθμοι χωρίζονται σε κατηγορίες ανάλογα με το αν εκπαιδεύονται με σταθερά ή εκπαιδευόμενα βάρη και με επίβλεψη ή χωρίς επίβλεψη [3]. Η κατηγοριοποίηση γίνεται ως εξής:

- Εκπαιδευόμενα βάρη
 - Με επίβλεψη
 - ❖ Perceptron
 - ❖ ADALINE
 - ❖ Backpropagation
 - ❖ Αναδρομικά δίκτυα Backpropagation
 - ❖ Δίκτυα RBF
 - ❖ Μοντέλα SVM
 - ❖ Στοχαστικές Μηχανές
 - Χωρίς επίβλεψη
 - ❖ Συσχετιστικά μοντέλα
 - Δίκτυα PCA
 - Δίκτυα ICA
 - ❖ Αναγνωστικά μοντέλα
 - Δίκτυο Kohonen (SOM)
 - Learning VQ
 - ART
- Σταθερά βάρη
 - Δίκτυο Hopfield
 - Συσχετισμός μνήμες
 - Brain State in a Box

2.2 Νευρωνικοί αλγόριθμοι με εκπαιδευόμενα βάρη με επίβλεψη

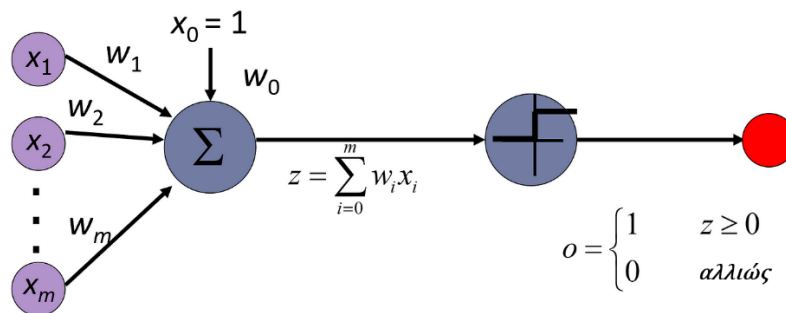
Όπως είδαμε στην προηγούμενη ενότητα 2.1 στην ταξινόμηση των αλγορίθμων των νευρωνικών δικτύων, τα συναπτικά βάρη μπορούν να είναι μεταβαλλόμενα κατά τη διάρκεια εκπαίδευσης ή σταθερά. Σε αυτή την ενότητα και τις υποενότητές της θα ασχοληθούμε με τους αλγόριθμους που αναδιατάσσουν τα βάρη τους κατά τη διάρκεια της εκπαίδευσης με σκοπό να εξασφαλίσουν τα καλύτερα δυνατά βάρη για τις προσομοιώσεις. Πρακτικά η διαδικασία που γίνεται είναι ότι κατά την

επαναληπτική διαδικασία της εκμάθησης, εξετάζονται αν τα βάρη που υπήρχαν στην προηγούμενη επανάληψη είναι ικανοποιητικά ή χρειάζεται να γίνουν διεργασίες διόρθωσής τους, για να μπορέσει το δίκτυο να αποκτήσει καλύτερη κατανόηση των δεδομένων και του προβλήματος κατά την εκπαίδευσή του.

Η μάθηση η οποία οφείλεται σε αλγορίθμους νευρωνίων **με επίβλεψη (supervised learning)** που χρησιμοποιούν σύνολα δεδομένων με ετικέτα για να εκπαιδευτούν τα νευρωνικά δίκτυα στην πρόβλεψη των αποτελεσμάτων και την αναγνώριση των μοτίβων. Σε αυτή τη κατηγορία αλγορίθμων προσφέρονται ετικέτες στους αλγορίθμους και κατ' επέκταση στα δίκτυα με σκοπό την αναγνώριση των πιθανών ή μη σχέσεων που έχουν τα δεδομένα εισόδου στο δίκτυο όπως και τα δεδομένα εξόδου. Έτσι λοιπόν μπορούν να δημιουργηθούν μοντέλα τα οποία μπορούν να κάνουν ακριβείς προβλέψεις και να εφαρμοστούν σε διάφορους κλάδους και τομείς, όπως είναι η υγεία, τα χρηματοοικονομικά, το μάρκετινγκ κτλ. Συνεπώς, γίνεται μία κατανόηση των δεδομένων από τα δίκτυα με τη βοήθεια των αλγορίθμων με επίβλεψη και δημιουργείται το μοντέλο πάνω στο οποίο θα δουλέψει σε νέα παραδείγματα.

2.2.1 Perceptron

Το πιο απλοϊκό και εύκολο δίκτυο να μελετηθεί είναι αυτό που αποτελείται από έναν και μόνο νευρώνα. Οι μόνες συνδέσεις που υπάρχουν είναι αυτές μεταξύ των εισόδων και του νευρώνα (βλ. Εικόνα 2.1)



Εικόνα 3 Perceptron: Το πιο απλό μοντέλο τεχνητού νευρώνα

Ο Perceptron λοιπόν ανακαλύφθηκε από τον Frank Rosenblatt (1928 – 1971) ο οποίος ήταν Αμερικανός ψυχολόγος διακεκριμένος στον τομέα της Τεχνητής Νοημοσύνης. Με τη χρήση ενός IBM 704 υπολογιστή στο Εργαστήριο Αεροναυπηγικής του Κόρνελ, το 1957 ξεκίνησε την ανάπτυξη του Perceptron. Οι επιστήμονες είχαν ανακαλύψει πως τα κελιά του εγκεφάλου, αποκαλούμενα και Νευρώνες, μπορούν να δεχτούν εισόδους από τις αισθήσεις μας μέσω ηλεκτρικών σημάτων. Για να μπορέσουν να αποθηκεύσουν την πληροφορία χρησιμοποιούν ηλεκτρικά σήματα και παίρνουν αποφάσεις βασισμένα στις προηγούμενες εισόδους. Έχοντας αυτά σαν γνώση ο Frank είχε την ιδέα να δημιουργήσει ένα δίκτυο από Perceptrons, που θα έχει ως στόχο να προσομοιώνει την ικανότητα μάθησης και λήψης αποφάσεων σαν τον ανθρώπινο εγκέφαλο [3].

Όπως φαίνεται και στην παραπάνω εικόνα (Εικόνα 3) τα βασικά συστατικά ενός Perceptron είναι τα εξής:

1. **Επίπεδο εισόδου:** Το επίπεδο εισόδου απαρτίζεται είτε από έναν είτε από περισσότερους νευρώνες εισόδου, οι οποίοι λαμβάνουν σήματα από το περιβάλλον (εξωτερικός κόσμος) ή

από άλλα στρώματα του νευρωνικού δικτύου, στην περίπτωση που παρατάζουμε πολλές συστοιχίες Perceptron.

2. **Συναπτικά βάρη:** Κάθε νευρώνας της εισόδου συνδέεται με ένα συναπτικό βάρος, το οποίο αντιπροσωπεύει την ισχύ της σύνδεσης μεταξύ του νευρώνα εισόδου και του νευρώνα εξόδου.
3. **Πόλωση – Κατώφλι (Bias – Threshold):** Είναι ένας επιπλέον όρος που προστίθεται στο στρώμα εισόδου, ο οποίος έχει μοναδιαία τιμή. Αν το σύνολο του αθροίσματος των τιμών των υπόλοιπων τιμών είναι μεγαλύτερο αυτήν, τότε ο νευρώνας πυροδοτεί.
4. **Συνάρτηση ενεργοποίησης:** Η συνάρτηση ενεργοποίησης καθορίζει την έξοδο του Perceptron βάσει του αθροίσματος που προκύπτει από τις εισόδους και την πόλωση. Οι πιο κοινές συναρτήσεις που χρησιμοποιούνται στον Perceptron είναι η βηματική συνάρτηση και η σιγμοειδής συνάρτηση.
5. **Έξοδος:** Η έξοδος του Perceptron είναι μία δυαδική τιμή 0 ή 1 η οποία υποδεικνύει την κλάση ή την κατηγορία στην οποία ανήκουν τα δεδομένα.
6. **Αλγόριθμος εκμάθησης:** Για την εκπαίδευση του νευρώνα Perceptron συνήθως χρησιμοποιείται ένας αλγόριθμος εκμάθησης Backpropagation. Την χρονική περίοδο που εκπαιδεύεται ο νευρώνας τα βάρη και οι πολώσεις προσαρμόζονται ανάλογα ώστε να επιτευχθεί το ελάχιστο τετραγωνικό σφάλμα ρίζας, για να έχουμε τις ελάχιστες δυνατές απώλειες και την καλύτερη δυνατή μάθηση και κατανόηση.

2.2.1.1 Παράμετροι του Perceptron

Η συνάρτηση μεταφοράς που υλοποιεί ο απλοποιημένος νευρώνας **Perceptron** είναι η εξής:

$$u = \sum_{i=1}^n w_i x_i - \theta \quad (3\alpha)$$

$$y = f(u) \quad (3\beta)$$

Η συνάρτηση μεταφοράς απεικονίζει το **διάνυσμα εισόδου** $x = [x_1, x_2, \dots, x_n]^T$ στην έξοδο y . Οι παράμετροι w_1, w_2, \dots, w_n είναι τα **συναπτικά βάρη** του νευρώνα ενώ η παράμετρος θ ονομάζεται **κατώφλι** ενεργοποίησης. Ο όρος αυτός εξηγείται από το γεγονός ότι η **διέγερση** u του νευρώνα η οποία προκύπτει από τις εισόδους είναι θετική αν το άθροισμα ξεπεράσει το όριο θ :

$$u > 0, \text{ αν } \sum_{i=1}^n w_i x_i > \theta \quad (4\alpha)$$

$$u = 0, \text{ αν } \sum_{i=1}^n w_i x_i = \theta \quad (4\beta)$$

$$u < 0, \text{ αν } \sum_{i=1}^n w_i x_i < \theta \quad (4\gamma)$$

Η **συνάρτηση ενεργοποίησης** $f(\cdot)$ τροφοδοτείται από το αποτέλεσμα που προκύπτει από το u και δίνει την έξοδο y του νευρώνα. Η συνάρτηση ενεργοποίησης είναι μη γραμμική και για τον Perceptron μπορεί να έχει μία από τις ακόλουθες μορφές.

$$f(u) = \begin{cases} 1, & \text{αν } u > 0 \\ 0, & \text{αν } u \leq 0 \end{cases} \quad (5\alpha)$$

$$f(u) = \begin{cases} 1, & \text{αν } u > 0 \\ -1, & \text{αν } u \leq 0 \end{cases} \quad (5\beta)$$

Συνεπώς η έξοδος y είναι ένας δυαδικός αριθμός που μπορεί να πάρει είτε το 0/1 είτε το -1/1. Οι παράμετροι που θα ρυθμίσουν τη συμπεριφορά του νευρώνα είναι το **διάνυσμα συναπτικών βαρών** $\mathbf{w} = [w_1, w_2, \dots, w_n]^T$ και το κατώφλι θ .

Σε κάθε επανάληψη που γίνεται λοιπόν, γίνεται εκ νέου αναπροσαρμογή στα συναπτικά βάρη για να πετύχει το δίκτυο το ελάχιστο δυνατό τετραγωνικό σφάλμα ρίζας, εφόσον κριθεί ότι κάποιο ή κάποια από τα υπάρχοντα βάρη δεν ικανοποιούν τις συνθήκες που απαιτούνται και πρέπει να αλλαχθούν.

2.2.1.2 Ο αλγόριθμος εκπαίδευσης του Perceptron

Για να μπορέσει να εκπαιδευτεί ο Perceptron χρειάζεται μία σειρά βημάτων, τα οποία αποτυπώνονται στη συνέχεια, για ένα συγκεκριμένο σύνολο επαναλήψεων – εποχών. Τα βήμα που πρέπει να ακολουθήσει επομένως είναι:

1. Αρχικοποίηση συναπτικών βαρών και κατωφλίων με τυχαίους και μικρούς αριθμούς.
2. Παρουσίαση νέων εισόδων x_1, x_2, \dots, x_n και επιθυμητής εξόδου y .
3. Υπολογισμός ενεργής εξόδου:

$$y(t) = f\left(\sum_{i=0}^n w_i(t)x_i(t)\right) \quad (6)$$

4. Προσαρμογή συναπτικών βαρών:

$$w_i(t+1) = w_i(t) + \varepsilon [d(t) - y(t)] x_i(t) \quad \forall i \leq n \quad (7)$$

όπου $\mathbf{d}(t)$ είτε με 0 ή 1 είτε με -1 ή 1 ανάλογα με το αν το $x(t)$ ανήκει στην κατηγορία A ή B και ο ρυθμός εκμάθησης επιλέγεται στο διάστημα $0 < \varepsilon < 1$.

5. Πήγαινε στο βήμα 2.

2.2.2 ADALINE

Ο όρος **ADALINE** αποτελεί ακρωνύμιο των αρχικών λέξεων **AD**aptive **LI**near **E**lement (Αυτοπροσαρμοζόμενο Γραμμικό Στοιχείο) [3] [5]. Τον όρο και κατ' επέκταση τον νευρώνα τον ανέπτυξαν οι **Widrow** και **Hoff** το 1960. Πρόκειται για ένα δίκτυο που έχει μια γραμμική μονάδα και από το οποίο λείπει η μη γραμμική συνάρτηση f . Έτσι ο γραμμικός νευρώνας ικανοποιείται και εκτελεί την ακόλουθη σχέση μεταξύ της εισόδου και της εξόδου:

$$u = \sum_{i=1}^n w_i x_i - \theta \quad (8)$$

Αν θεωρήσουμε ότι στην παραπάνω σχέση 3, ότι το κατώφλι θ είναι ένα επιπλέον συναπτικό βάρος w_0 και ορίζουμε την είσοδο $x_0 = 1$, τότε μπορούμε να γράψουμε τη σχέση ως εξής:

$$u = \sum_{i=1}^n w_i x_i \quad (9)$$

ή

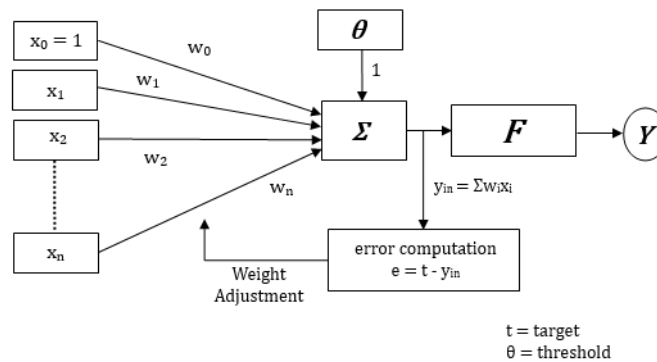
$$u = \mathbf{w}^T \mathbf{x} \quad (10)$$

όπου $\mathbf{w} = [w_0, w_1, w_2, \dots, w_n]^T$, $\mathbf{x} = [x_0, x_1, x_2, \dots, x_n]^T$ είναι τα διανύσματα βάρους και εισόδου αντίστοιχα.

Η διαφορά με το μη γραμμικό νευρώνα των McCulloch – Pitts παρουσιάζεται στο γεγονός ότι η έξοδος u μπορεί να πάρει τιμές από $-\infty$ έως το $+\infty$ και ότι η έξοδος δεν λαμβάνει τις διακριτές τιμές 0/1 ή -1/1. Αυτό σημαίνει ότι ενδέχεται και οι στόχοι να παίρνουν τιμές από το σύνολο των πραγματικών αριθμών \mathbb{R} . Το πρόβλημα το οποίο όμως αντιμετωπίζεται είναι πως οι τιμές των στόχων $\mathbf{d}^{(p)}$ δεν είναι προφανείς αν δεν προέρχονται από κάποιο διακριτό σύνολο. Συνεπώς, καταλήγουμε να χρησιμοποιούμε τις κοινότερες τιμές 0 / 1.

2.2.2.1 Η επανάληψη της εκπαίδευσης του δικτύου ADALINE

Παρακάτω ακολουθεί ένα σχεδιάγραμμα του κύκλου επανάληψης που κάνει ένα δίκτυο ADALINE:



Εικόνα 4 ADALINE

2.2.2.2 Ο αλγόριθμος εκπαίδευσης του ADALINE

Όπως κάθε Τεχνητό Νευρωνικό Δίκτυο έτσι και το ADALINE ακολουθεί μία σειρά από βήματα για να μπορέσει να εκπαιδευτεί. Τα βήματα τα οποία είναι απαραίτητα και πρέπει ν' ακολουθήσει είναι τα εξής:

1. Αρχικοποίηση των βαρών και των κατωφλίων με μικρούς τυχαίους αριθμούς.
2. Παρουσίαση νέας εισόδου $\mathbf{x}_k = x_{k0}, x_{k1}, x_{k2}, \dots, x_{kn}$ και επιθυμητής εξόδου u_k .
3. Υπολογισμός συνολικής εισόδου

$$u_k = \sum_{i=0}^n w_i(k) x_{ik} \quad (11)$$

4. Προσαρμογή συναπτικών βαρών με τον κανόνα Widrow – Hoff:

$$w_i(k+1) = w_i(k) + a(k)[d_k - u_k]x_{ik}, \forall 0 \leq i \leq n \quad (12)$$

όπου $u_k = 1$ ή -1 ανάλογα με την κατηγορία στην οποία ανήκει το x_k και ο ρυθμός εκμάθησης επιλέγεται: $0 < a(k) < 1$.

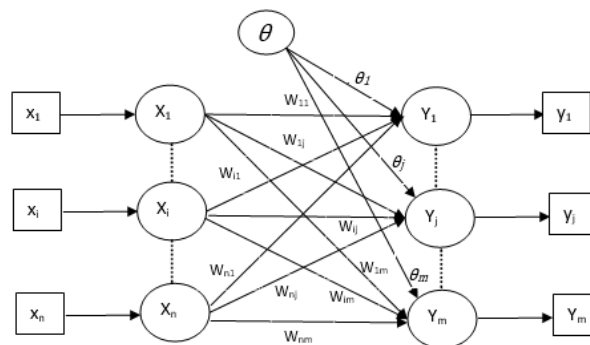
5. Πήγαινε στο βήμα 2 μέχρι να σταματήσουν να αλλάζουν τα βάρη ή να συμπληρωθεί ο μέγιστος αριθμός των επαναλήψεων.

Οι ιδιότητες τις οποίες έχει ο ADALINE είναι οι εξής:

- Συγκλίνει ακόμη κι αν το πρόβλημα δεν είναι γραμμικά διαχωρίσιμο.
- Δεν εγγυάται το διαχωρισμό των κλάσεων όταν το πρόβλημα είναι γραμμικά διαχωρίσιμο. Στον αλγόριθμο του Perceptron τέτοιο πρόβλημα δεν υφίσταται.

2.2.2.3 MADALINE

Το **MADALINE** είναι ένα δίκτυο το οποίο προκύπτει από την παράλληλη σύνδεση πολλαπλών ADALINEs εξού και το **Multiple ADALINEs = MADALINE**. Με το MADALINE έχουμε τη δυνατότητα να συσχετίσουμε διανύσματα εισόδου με διανύσματα εξόδου και όχι με μία μόνο τιμή.



Εικόνα 5 MADALINE

2.2.3 Backpropagation

Ο **Backpropagation (BP)** είναι ένας αλγόριθμος οπίσθιας μετάδοσης των σφαλμάτων (συντομογραφία του “Backward Propagation”) και αποτελεί την πιο γνωστή και διαδεδομένη μέθοδο εκπαίδευσης Τεχνητών Νευρωνικών Δικτύων τα οποία είναι πολυστρωματικά. Έχει χρησιμοποιηθεί στις περισσότερες εφαρμογές δικτύων έως σήμερα, αποτελώντας μία βασική σταθερά για τα προβλήματα που αναζητούν δίκτυα με εκπαίδευση από αλγόριθμους με επίβλεψη. Πρόκειται για ένα εργαλείο που βελτιώνει σημαντικά τα σφάλματα των εκτελέσεων, προσδοκώντας την καλύτερη δυνατή ακρίβεια στη λύση των προβλημάτων.

Γενικά η μέθοδος η οποία ακολουθεί βασίζεται στο σκεπτικό της επανατροφοδότησης των εξόδων σε εισόδους, εφόσον γίνεται αναπροσαρμογή των βαρών, και επανάληψη της διαδικασίας για όσο χρειαστεί μέχρι να περιοριστεί το σφάλμα στο μεγαλύτερο βαθμό ή να τελειώσει το πλήθος των εποχών εκμάθησης (επαναλήψεων εκμάθησης). Από μαθηματικής απόψεων διαχειρίζεται με γρήγορες

μεθόδους τον υπολογισμό των παραγώγων σε ένα νευρωνικό δίκτυο, οι οποίες είναι αλλαγές στην έξοδο λόγω συντονισμού και προσαρμογών.

Μεγαλύτερη ανάγκη για χρήση του Backpropagation προέκυψε όταν επανεμφανίστηκε η ανάγκη για χρήση δικτύων βαθιάς μάθησης (Deep Learning Networks) για την αναγνώριση εικόνων και ομιλίας. Θεωρείτε ένας από τους πιο αποτελεσματικούς αλγορίθμους για τις σύγχρονες υλοποιήσεις συστημάτων και πλέον για να βελτιστοποιήσει περαιτέρω τις δυνατότητες του αξιοποιεί πλήρως και τις GPU των συστημάτων που τον εκτελούν.

Παρακάτω θα κάνουμε μία σύντομη αναφορά για τον Backpropagation στην γενική δομή του και στην εφεύρεσή του και σε μετέπειτα κεφάλαιο (**Κεφάλαιο 4**) θα προχωρήσουμε σε εκτενέστερη ανάλυση της δομής, της μαθηματικής προσέγγισης και των βασικών παραλλαγών που χρησιμοποιήθηκαν και στο πρακτικό κομμάτι.

2.2.3.1 Ιστορική ανασκόπηση

Ο αλγόριθμος του Backpropagation ανακαλύφθηκε στην δεκαετία του 1970 ως μια γενική μέθοδο βελτιστοποίησης για την εκτέλεση αυτόματης διαφοροποίησης πολλαπλών εμφωλευμένων συναρτήσεων. Ωστόσο, το 1986 με τη δημοσίευση που έκαναν για μία επιστημονική εργασία οι Rumelhart, Hinton και Williams, με τίτλο “**Learning Representations by Back-Propagating Errors**” (“**Μάθηση Αναπαραστάσεων με Λάθη Οπίσθιας Διάδοσης**”), η σημασία του αλγορίθμου εκτιμήθηκε από την ευρύτερη κοινότητα μηχανικής μάθησης.

Στη κοινότητα των ερευνητών αιωρούνταν για μεγάλο χρονικό διάστημα η ανάγκη για την εκπαίδευση Τεχνητών Νευρωνικών Δικτύων τα οποία δε θα είναι μονοστρωματικά, αλλά θα αποτελούνται από πολλαπλά στρώματα νευρώνων. Τα δίκτυα αυτά θα είχαν τη δυνατότητα να μπορούν να αναγνωρίσουν εσωτερικά μοτίβα και να κάνουν τη μάθηση πιο εύκολη αλλά και να έχει μεγαλύτερη ακρίβεια (ευστοχία – accuracy). Με δεδομένο ότι η ενεργοποίηση ενός κόμβου εξαρτάται από τα εισερχόμενα βάρη αλλά και την πόλωση, οι ερευνητές ισχυρίζονται ότι ένας κόμβος έχει μάθει ένα χαρακτηριστικό αν τα βάρη και η πόλωση προκαλούν την ενεργοποίηση του κόμβου όταν το χαρακτηριστικό αυτό είναι παρόν στην είσοδό του.

Ο Backpropagation ήταν μία από τις πρώτες μεθόδους που μπόρεσαν να αποδείξουν ότι τα τεχνητά νευρωνικά δίκτυα μπορούσαν να μάθουν μοτίβα σε ικανοποιητικό βαθμό. Οι ειδικοί εξέτασαν τα πολυστρωματικά εμπροστροφοδοτούμενα δίκτυα που εκπαιδεύτηκαν με τον Backpropagation και διαπίστωσαν πως στην πραγματικότητα πολλοί κόμβοι έμαθαν χαρακτηριστικά παρόμοια με αυτά που σχεδιάστηκαν από το ειδικούς χειροκίνητα αλλά και αυτά που βρήκαν οι νευροεπιστήμονες μέσα από έρευνες βιολογικών νευρωνικών δικτύων από θηλαστικά ζώα. Έτσι μετά τη διαπίστωση από τους ειδικούς ότι μπορεί ο Backpropagation να αναγνωρίσει τα πρότυπα χαρακτηριστικά, μπόρεσε να ενταχθεί και σε πολυστρωματικά τεχνητά νευρωνικά δίκτυα, καθώς μεταγενέστερα και σε εφαρμογές που απαιτούσαν δίκτυα βαθιάς μάθησης.

2.2.3.2 Πλεονεκτήματα και Μειονεκτήματα των Backpropagation αλγορίθμων

Τα πλεονεκτήματα τα οποία έχουν οι αλγόριθμοι του Backpropagation είναι αρκετά αλλά έχουν και κάποια μειονεκτήματα που δημιουργούν προβλήματα κατά την εκτέλεση των νευρωνικών δικτύων στις εφαρμογές.

Τα πλεονεκτήματα είναι τα εξής:

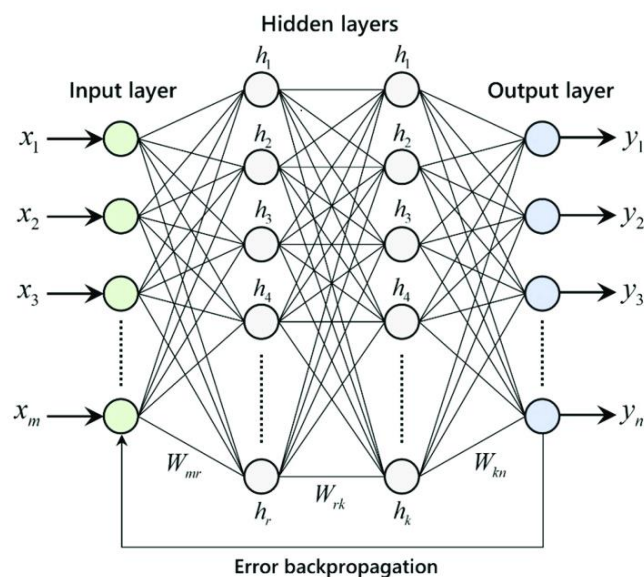
- Δεν έχουν καμιά παράμετρο για τον συντονισμό εκτός από τον αριθμό των εισόδων.
- Η προσαρμοστικότητά τους είναι ιδιαίτερα καλή και έχουν καλή αποτελεσματικότητα χωρίς να απαιτούν προηγούμενη σχετικά με το δίκτυο.
- Χρησιμοποιούν μία τυπική διαδικασία που συνήθως λειτουργεί καλά.
- Είναι φιλικά για τον χρήστη, γρήγορα και εύκολα ως προς τον προγραμματισμό τους.
- Οι χρήστες δεν είναι απαραίτητο να μάθουν ειδικές λειτουργίες.

Τα μειονεκτήματά τους είναι τα εξής:

- Προτιμούν μια προσέγγιση matrix – based παρά mini – batch.
- Η εξόρυξη των δεδομένων και των αποτελεσμάτων γενικότερα είναι ευαίσθητη σε δεδομένα τα οποία έχουν μολυνθεί με θόρυβο και σε άλλες παρατυπίες. Με αυτό τον τρόπο δημιουργούν πρόβλημα στην εκπαίδευση του δικτύου με Backpropagation άρα και στην αποτελεσματικότητά του.
- Η απόδοσή του εξαρτάται σε μεγάλο βαθμό από την ποιότητα των δεδομένων εισόδου.
- Η εκπαίδευση απαιτεί χρόνο και πόρους.
- Ο αλγόριθμος μπορεί να κολλήσει σε τοπικά ελάχιστα καταναλώνοντας επιπρόσθετο χρόνο ή ακόμα και να μην καταφέρει να τελειώσει με επιτυχία τις προσομοιώσεις.
- Όταν ο **Ρυθμός Εκμάθησης (Learning Rate - LR)** είναι πολύ μικρός, η σύγκλιση είναι πολύ αργή.
- Όταν το δίκτυο που εκπαιδεύεται με BP, εκπαιδεύεται περισσότερο του αναμενομένου – αναγκαίου (over – trained), αποκτά χειρότερη γενικευμένη απόδοση.
- Η μέθοδος Gradient – Based είναι χρονοβόρα σε μεγάλες εφαρμογές.

2.2.3.3 Σχηματική απεικόνιση δικτύου με Backpropagation εκμάθηση

Παρακάτω ακολουθεί μία εικόνα που αντιπροσωπεύει την σχηματική απεικόνιση ενός νευρωνικού δικτύου το οποίο έχει εκπαιδευτεί με τη χρήση του Backpropagation:



Εικόνα 6 Πολυστρωματικό νευρωνικό δίκτυο με χρήση Backpropagation αλγορίθμου εκμάθησης

2.2.4 Αναδρομικά δίκτυα Backpropagation

Ο Backpropagation χωρίζεται σε δύο βασικούς τύπους δικτύων τον **στατικό** Backpropagation και τον **αναδρομικό** Backpropagation.

- Ο στατικός Backpropagation (**Static Backpropagation**) είναι ένα αλγόριθμος ο οποίος είναι σχεδιασμένος να διαχειρίζεται και να χαρτογραφεί δεδομένα εισόδων και εξόδων που οι τιμές τους είναι στατικές.
- Ο αναδρομικός Backpropagation (**Recursive Backpropagation – RBP**) είναι ένα αλγόριθμος που χρησιμοποιείται για την εκμάθηση ενός σταθερού σημείου. Ουσιαστικά γίνεται επαναλαμβανόμενη αντίστροφη τροφοδότηση ώστε να επιτευχθεί μία σταθερή τιμή.

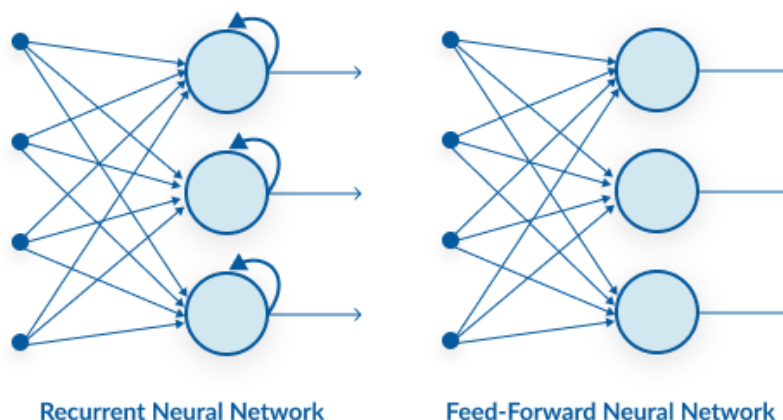
Ο λόγος για τον οποίο χρησιμοποιείται ο αναδρομικός Backpropagation οφείλεται στο γεγονός ότι είναι αναγκαία η βελτιστοποίηση των βαρών βασισμένος στις μεθόδους Gradient Descent (Απότομης Κατάβασης). Ένα δίκτυο το οποίο εκπαιδεύεται με RBP δεν απαιτεί την ύπαρξη εποχών για την εκπαίδευσή του ή προσεγγιστικούς συντελεστές.

Μία από τις βασικές ικανότητες τους είναι ότι μπορούν να μοντελοποιήσουν και να αναπαραστήσουν χρονικά μεταβαλλόμενα μοντέλα, συνεπώς με αυτόν τον τρόπο γίνονται δυναμικά καθώς μπορούν να προσαρμόζονται σε διαφορετικές χρονικές στιγμές.

Η χρησιμότητά τους παρατηρείται στη βαθιά μάθηση και ο λόγος είναι ότι προσπαθούν να προσομοιώσουν τις λειτουργίες τους ανθρώπινου εγκεφάλου, κάτι το οποίο απαιτεί μεγάλη σχολαστικότητα και επαναληπτικότητα. Μπορούμε να δούμε μοντέλα που χρησιμοποιούνται στην αναγνώριση ομιλίας, στις τηλεπικοινωνίες, στην ανάλυση δεδομένων από video, στη μετάφραση κειμένου και στην αυτοματοποιημένη συγγραφή ενός κειμένου ή κώδικα [6].

2.2.4.1 Σχηματική απεικόνιση

Παρακάτω απεικονίζεται σχηματικά η ροή της λειτουργίας των αναδρομικών δικτύων με Backpropagation συγκριτικά με τα μοντέλα εμπρόσθιας τροφοδότησης, καθώς όπως παρατηρείται η έξοδός τους επανατροφοδοτεί τους νευρώνες του κρυφού στρώματος, επαναλαμβάνοντας αυτή τη διαδικασία έως ότου να επιτευχθεί η σταθερά που είναι αναγκαία.



Εικόνα 7 RNN - FFNN

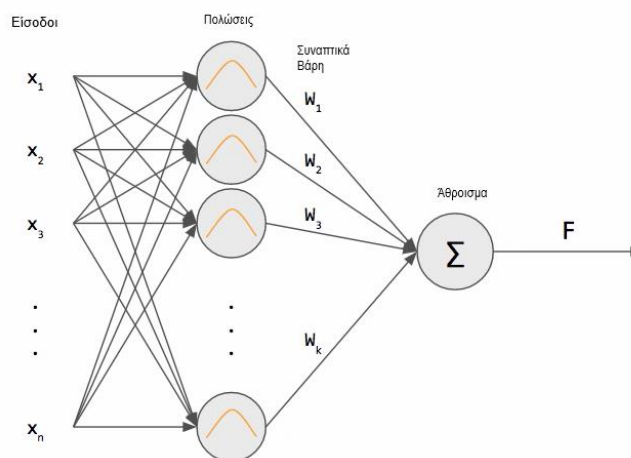
2.2.5 Δίκτυα RBF

Τα δίκτυα **RBF** (Δίκτυα Ακτινικής Βάσης) έχουν μία αρχιτεκτονική διαφοροποιημένη από τα περισσότερα νευρωνικά δίκτυα, η οποία έγκειται στο γεγονός ότι αποτελούνται μόνο από ένα στρώμα εισόδου, ένα κρυφό στρώμα και ένα στρώμα εξόδου. Τα περισσότερα νευρωνικά δίκτυα αποτελούνται από πολλαπλά στρώματα που εισάγοντας τη μη γραμμικότητα από ένα σύνολο μη γραμμικών συναρτήσεων. Το γεγονός ότι τα RBF δίκτυα αποτελούνται από δύο στρώματα και δε χρειάζονται παραπάνω, βασίζεται στο γεγονός ότι δεν υπάρχει κάποιος αλγόριθμος αξιόλογος που να μπορεί να διαχειριστεί δίκτυα με τρία ή περισσότερα στρώματα, ωστόσο είναι και άκρως επαρκές το σύνολο των δύο στρωμάτων.

Η ανάκληση σε ένα δίκτυο το οποίο είναι RBF είναι απολύτως όμοια με ένα δίκτυο MLP (Multiple Layer Perceptron) . Εδώ μάλιστα μέσω των δύο στρωμάτων υπάρχει και μία απλούστερη διαδικασία, καθώς το κρυφό στρώμα περιέχει τη συνάρτηση ακτινικής βάσης f ενώ το στρώμα εξόδου είναι γραμμικό. Από την άλλη το στρώμα εισόδου δεν είναι ένα στρώμα στο οποίο γίνονται υπολογισμοί, απλά είναι ένας αγωγός – εξάγγελος για τα δεδομένα ώστε να καταλήξουν στο κρυφό στρώμα [3].

2.2.5.1 Ιστορική αναφορά

Όπως στα MLP έτσι και στα RBF υπάρχει αντίστοιχο θεώρημα καθολικού προσεγγιστή. Το θεώρημα αυτό αποδείχθηκε από τους Park και Sandberg το 1991 στο οποίο αποδεικνύεται πως οι συναρτήσεις ακτινικού τύπου είναι βάση του χώρου των συνεχών συναρτήσεων, συνεπώς μπορούμε να προσεγγίσουμε οποιαδήποτε συνεχή συνάρτηση. Η πρώτη όμως αναφορά γι' αυτά τα δίκτυα έγινε σε μία εργασία από τους Broomhead και Lowe που ήταν ερευνητές στο Royal Signals and Radar Establishment, το 1988.



Εικόνα 8 RBF

2.2.5.2 Σύγκριση RBF – MLP

- Τα δίκτυα RBF έχουν δύο στρώματα ενώ τα MLP περισσότερα από δύο στρώματα.
- Τα MLP μοιάζουν περισσότερο με τους πραγματικούς νευρώνες λόγω των πολλών στρωμάτων τους.
- Εκπαιδεύονται και τα δύο δίκτυα με επίβλεψη.

- Το δίκτυο RBF μπορεί να οργανώσει μόνο του το κρυφό στρώμα.
- Και τα δύο είναι παγκόσμιοι προσεγγιστές, που σημαίνει ότι μπορούμε να προσεγγίσουμε το ένα είδος δικτύου χρησιμοποιώντας το άλλο και αντίστροφα.
- Είναι ισοδύναμα μεταξύ τους και στις μηχανές ασαφούς λογικής (fuzzy logic).

2.2.5.3 Εκπαίδευση δικτύου RBF

Σχετικά με την εκπαίδευση των δικτύων νευρωνίων με RBF, το πρώτο (κρυφό) στρώμα δεν υπολογίζει τα εσωτερικά γινόμενα με τις εισόδους, αλλά την **Ευκλείδεια Απόσταση** του διανύσματος βαρών κάθε νευρώνιου από το διάνυσμα εισόδων:

$$u_i = \|x - w_i\| = \sqrt{(x_1 - t_{i1})^2 + \dots + (x_n - t_{in})^2} \quad (13)$$

όπου $w_i = (t_{i1}, t_{i2}, \dots, t_{in})$ είναι τα συναπτικά βάρη του νευρώνα i με $1 \leq i \leq n$.

Η έξοδος $f(u_i)$ των νευρωνίων του κρυφού στρώματος υπολογίζεται ως μία μη γραμμική συνάρτηση ακτινικής βάσης του u_i και συνήθως περιγράφεται από την γκαουσιανή συνάρτηση:

$$f(u_i) = e^{-\frac{u_i^2}{\sigma_i^2}} \quad (14)$$

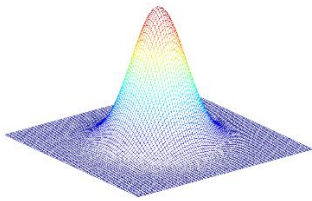
ή ισοδύναμα, επειδή u_i εξαρτάται από x , από την

$$f_i(x) = e^{-\frac{\|x - t_i\|^2}{\sigma_i^2}} \quad (15)$$

όπου η “μέση τιμή” του w_i και η “τυπική απόκλιση” σ_i της γκαουσιανής είναι οι παράμετροι που καθορίζουν σε ποια περιοχή του χώρου εισόδων εστιάζεται το νευρόνιο i .

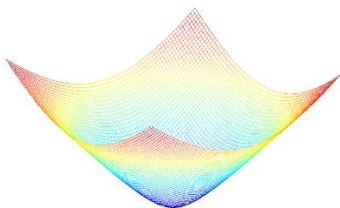
2.2.5.4 Δημοφιλείς συναρτήσεις ακτινικού τύπου

Κάποιες από τις πιο δημοφιλείς συναρτήσεις ακτινικού τύπου είναι οι εξής [7]:



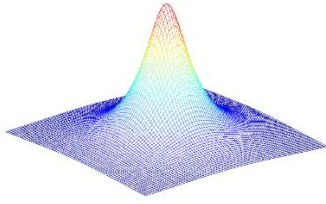
Γκαουσιανή
(Gaussian)

$$f(x) = e^{-\frac{\|x - c\|^2}{\sigma^2}}$$



Ανάστροφη
Πολυτεταγωνική
(Inverse
Multiquadratic)

$$f(x) = (\|x - c\|^2 + \sigma^2)^{\frac{1}{2}}$$



Cauchy

$$f(x) = \frac{(\|x - c\|^2 + \sigma^2)^{-1}}{\sigma}$$

Πίνακας 2 Συναρτήσεις Ακτινικής Βάσης (RBF)

2.2.6 Μηχανές Διανυσμάτων Υποστήριξης (SVM)

Η προέλευση και η εξέλιξη του όρου “**Support Vector Machine**” έχει συνυφαστεί με τους πρωτοπόρους ερευνητές πάνω στην εξέλιξη της Τεχνητής Νοημοσύνης και της Μηχανικής μάθησης εδώ και πολλά χρόνια. Η έναρξη της χρήσης τους σηματοδότησε μία μεγάλη εξέλιξη στην ανάλυση των δεδομένων, οδηγώντας σε καινοτομίες που διαμόρφωσαν το σύγχρονο τοπίο της Τεχνητής Νοημοσύνης.

2.2.6.1 Προέλευση και Ανάπτυξη των SVMs

Οι Μηχανές Διανυσμάτων Υποστήριξης σαν εννοιολογικός όρος πρωτοεμφανίστηκε τη δεκαετία του 1960. Ωστόσο μέχρι και τη δεκαετία του 1990 δεν υπήρχε κάποια ιδιαίτερη εξέλιξη, κάτι το οποίο έμελλε να αλλάξει, καθώς ο Vladimir Vapnik και η Corina Cortes πέτυχαν αξιόλογες προόδους στην τελειοποίηση των αλγορίθμων και των απαραίτητων Μαθηματικών που βασιζόνταν. Έτσι ανέδειξαν τις ικανότητές τους πάνω στην ταξινόμηση, την παλινδρόμηση και τις ανίχνευσης ακραίων σημείων μέσω των οποίων μπορούσε να γίνει εξόρυξη και πρόγνωση νέων δεδομένων.

Σταδιακά οι Μηχανές Διανυσμάτων Υποστήριξης ξεκίνησαν να γίνονται πιο δημοφιλείς και αυτό οφείλεται κατά κύριο λόγο στο γεγονός ότι υπήρχε ευελιξία στην θεωρητική ανάλυσή τους, στην προσαρμοστικότητά του σε πολλές εφαρμογές και στην προβλεπτική ικανότητα που έχουν.

Τα κυριότερα προβλήματα τα οποία λύνουν είναι τα δυαδικά, δηλαδή τα προβλήματα στα οποία υπάρχουν δύο τάξεις – κατηγορίες. Ταυτόχρονα υπάρχουν δύο κατηγορίες που χωρίζονται τα SVMs. Είναι η πρωταρχική και η διπλή μορφή. Η κάθε μία έχει όπως είναι προφανές τα πλεονεκτήματά της και τα μειονεκτήματά της. Η πρωταρχική μορφή είναι η παλαιότερη από τις δύο και είναι πιο εύκολη στη χρήση της, αλλά ταυτόχρονα και την λιγότερο ισχυρή. Τουναντίον, η διπλή μορφή έχει ανάγκη από περισσότερη ισχύ για να μπορέσει να λύσει προβλήματα, όμως αποτελεί τον πυρήνα των SVM μηχανών. Η επέκταση των SVMs έγινε με το Kernel SVM το οποίο είναι μία εξελιγμένη παραλλαγή των παλαιών, μέσω του οποίου επιτεύχθηκαν οι μεγαλύτερες και πιο σημαντικές επιτυχίες στη μηχανική μάθησης.

2.2.6.2 Πλεονεκτήματα – Μειονεκτήματα των SVMs και η εφαρμογή τους

Κάποια από τα σημαντικότερα πλεονεκτήματα των Μηχανών Διανυσμάτων Υποστήριξης είναι τα ακόλουθα:

- Έχουν πολύ καλή απόδοση όταν δεν έχουμε καλή εικόνα ή ακόμα δε γνωρίζουμε τις ιδιαιτερότητες των δεδομένων μας.

- Λειτουργούν πολύ καλά και με δεδομένα τα οποία δεν έχουν πλήρη συγκρότηση και οργάνωση όμως είναι τα δέντρα, τα κείμενα ή οι εικόνες.
- Η πραγματική δύναμη των SVMs είναι ο πυρήνας τους, μέσω του οποίου εφόσον υπάρχει κάποια στοχευμένη και κατάλληλη συνάρτηση, μπορούν να λύσουν οποιοδήποτε περίπλοκο πρόβλημα.
- Έχουν πολύ καλή απόδοση σε δεδομένα υψηλής κλίμακας.
- Έχουν μικρότερο βαθμό κινδύνου για over-fitting.
- Στη σύγκρισή τους με τα ANN, προσφέρουν καλύτερη αποτελεσματικότητα.

Στον αντίποδα τα σημαντικότερα μειονεκτήματά τους είναι τα ακόλουθα:

- Η εύρεση κατάλληλης συνάρτησης πυρήνα δεν είναι εύκολη υπόθεση, συνεπώς δεν έχουμε πάντα προσομοιώσεις με τις βέλτιστες συνθήκες.
- Καταναλώνουν μεγάλη χρονική διάρκεια για να μπορέσουν να εκπαιδευτούν σε δεδομένα μεγάλης κλίμακας.
- Είναι δύσκολη η κατανόηση του τελικού μοντέλου και των μεταβλητών των βαρών.

Κάποιες από τις πιο γνωστές εφαρμογές των Μηχανών Διανυσμάτων Υποστήριξης είναι:

- Η πρόβλεψη της δομής των πρωτεϊνών.
- Η αναγνώριση των χειρόγραφων εγγράφων.
- Η ανίχνευση στεγανογραφίας μέσα σε εικόνες, δηλαδή πληροφορίες τις οποίες θέλει ο αποστολέας να παραδώσει έχοντας αποκρύψει σημαντικά κομμάτια από αυτές.
- Στη διάγνωση του καρκίνου του μαστού.

2.2.6.3 Μαθηματική προσέγγιση των Μηχανών Διανυσμάτων Υποστήριξης

Έστω ότι έχουμε ένα πρόβλημα ταξινόμησης δύο κλάσεων οι οποίες είναι οι A_0 και A_1 , οι οποίες είναι γραμμικά διαχωρίσιμες. Αυτό σημαίνει ότι υπάρχει ένα διάνυσμα w και ένα κατώφλι w_0 για τα οποία ισχύει:

$$w^T x + w_0 \begin{cases} < 0, & \text{αν } x \in A_0 \\ > 0, & \text{αν } x \in A_1 \end{cases} \quad (16)$$

Είναι σαφές πως δεν υπάρχει μία λύση για την επίλυση του συγκεκριμένου προβλήματος, καθώς είναι άπειρα τα ζεύγη (w, w_0) που μπορούν να ικανοποιούν τη συνθήκη. Συνεπώς, για να μπορέσει να επιλεγεί η πιο σωστή – βέλτιστη επιλέγεται κάποιο κριτήριο το οποίο ονομάζεται **περιθώριο ταξινόμησης (margin)** γ , μεταξύ των δύο κλάσεων το οποίο ορίζεται ως το άθροισμα

$$\gamma = \gamma_0 + \gamma_1 \quad (17)$$

όπου το γ_0 ανήκει στην κλάση A_0 και αντίστοιχα το γ_1 ανήκει στην κλάση A_1 για τα οποία ισχύει ότι:

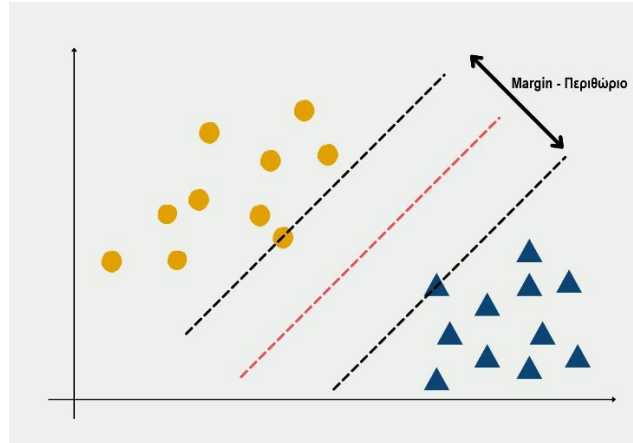
$$\gamma_0 = \min_{x \in A_0} \frac{|w^T x + w_0|}{\|w\|} = \min_{x \in A_0} - \frac{w^T x + w_0}{\|w\|} \quad (18)$$

$$\gamma_1 = \min_{x \in A_1} \frac{|w^T x + w_0|}{\|w\|} = \min_{x \in A_1} - \frac{w^T x + w_0}{\|w\|} \quad (19)$$

Αρα ψάχνουμε να βρούμε τις ελάχιστες τιμές ώστε να ικανοποιείται η παρακάτω εξίσωση της γραμμής:

$$w^T x + w_0 = 0 \quad (20)$$

Τα πρότυπα x των δύο αυτών κλάσεων που προσπαθούν να ικανοποιήσουν τις παραπάνω σχέσεις, ονομάζονται **διανύσματα υποστήριξης (support vectors)**.



Εικόνα 9 SVM margin

Η ανάλυση των SVMs δε σταματάει σε αυτά που έχουμε παραθέσει μέχρι στιγμής. Αύτη είναι μία επιφανειακή αναφορά σχετικά με το πως εξελίχθηκαν και πως μπορούν να αξιοποιηθούν. Ο Huang, που πρότεινε τον αλγόριθμο ELM που θα δούμε και παρακάτω, χρησιμοποίησε αλγόριθμο SVM για την επίλυση των προβλημάτων που είχε επιλέξει να εξετάσει και να συγκρίνει τ' αποτελέσματά τους. Όμως, επειδή στη συγκεκριμένη διπλωματική δε χρησιμοποιήσαμε και εμείς SVM, κάνουμε μία απλή αναφορά στην ύπαρξή τους και δεν επεκτεινόμαστε περαιτέρω.

2.2.7 Στοχαστικές Μηχανές

Οι **Στοχαστικές Μηχανές** και κατ' επέκταση τα στοχαστικά νευρωνικά δίκτυα είναι εκείνα τα οποία ο αλγόριθμος εκμάθησης αντί να παρέχει στο εκάστοτε νευρώνιο του δικτύου ντετερμινιστικές τιμές, εκχωρεί πιθανότητες. Συνεπώς, αντί το κατώφλι να είναι ένας πραγματικός αριθμός πάνω από τον οποίο πυροδοτούν οι νευρώνες του δικτύου, είναι ένας ακέραιος μη αρνητικός αριθμός, δηλαδή πιθανότητα που όπως και σε όλα τα υπόλοιπα νευρωνικά δίκτυα, αν η τιμή εξόδου από τον νευρώνα είναι μεγαλύτερη αυτού, πυροδοτεί ο νευρώνας. Κατά την κατασκευή τους δίνονται τυχαία στοχαστικά συναπτικά βάρη στις συνάψεις [8].

Η χρησιμότητά τους είναι σημαντική σε εφαρμογές οι οποίες που χρίζουν βελτιστοποίησης, με βασικό προτέρημα ότι οι τυχαίες διακυμάνσεις στις τιμές τους βοηθούν στην αποφυγή από τα τοπικά ελάχιστα που έχουν τα ANN δίκτυα και αποτελεί ένα πολύ σοβαρό πρόβλημα.

2.2.7.1 Μηχανή Boltzmann

Η Μηχανή Boltzmann η οποία πήρε το όνομά της από τον Ludwig Boltzmann (1844 – 1906) που ήταν Αυστριακός φυσικός και φιλόσοφος, είναι ένα μοντέλο που η αρχιτεκτονική του και η δομή του βασίζονται στην ενέργεια. Πρόκειται για ένα στοχαστικό νευρωνικό δίκτυο που εφαρμόζεται στα

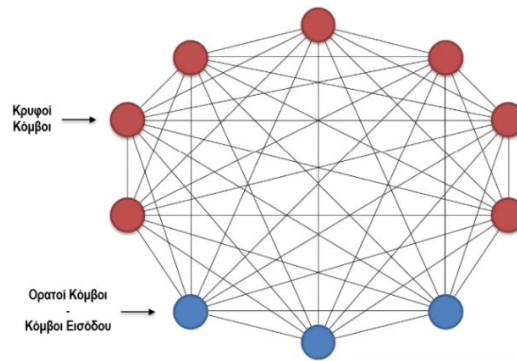
πλαίσια γνωστικής επιστήμης και μπορούν να εκπαιδευτούν είτε με επίβλεψη είτε χωρίς επίβλεψη αναλόγως τη διεργασία.

Απαρτίζεται από κρυφούς και φανερούς κόμβους και δεν έχει κόμβους εξόδου με το δεύτερο βασικό χαρακτηριστικό της δομής τους να είναι η απευθείας σύνδεση κάθε κόμβου με όλους τους υπόλοιπους. Γίνεται τροφοδότηση των κρυφών με τα δεδομένα εισόδου και μετέπειτα αναπροσαρμογή των πιθανοτήτων των συναπτικών βαρών αναμεταξύ τους. Πέρα από τα δεδομένα που μπορεί να δεχθεί από τους κόμβους εισόδου, μπορεί να παράξει και τα δικά του δεδομένα εσωτερικά των κρυφών κόμβων χωρίς να περιμένει εξωτερικό ερέθισμα [3] [9].

Η βάση του μοντέλου έχει την Ενέργεια ως κυρίως χαρακτηριστικό και η βασική αρχή λειτουργίας τους αφορά την κατανομή πιθανότητας Boltzmann $p(x)$ η οποία θα πρέπει να ικανοποιεί τη σχέση $\mathbf{0} \leq p(x) \leq \mathbf{1}$. Το σύνολο των πιθανοτήτων όλων των τιμών θα πρέπει να είναι 1.

$$\sum_{i=1}^K p(x_i) = 1 \quad (21)$$

Μια τοπολογία που απεικονίζει μία μηχανή Boltzmann είναι η ακόλουθη:



Εικόνα 10 Boltzmann Machine

2.2.7.2 Πλεονεκτήματα, Μειονεκτήματα και Σύγκριση με Ντετερμινιστικά Νευρωνικά Δίκτυα

Όπως σε όλα τα είδη νευρωνικών δικτύων έτσι και στις Στοχαστικές Μηχανές υπάρχουν κάποια πλεονεκτήματα που κάνουν την επιλογή τους ελκυστική ή και αναγκαία, αλλά και κάποια μειονεκτήματα από την επιλογή και χρήση τους σε σύγκριση με τα Ντετερμινιστικά Μοντέλα Νευρωνικών Δικτύων. Με τη χρήση λοιπόν των Στοχαστικών Μηχανών κερδίζουμε τα εξής:

- Μπορούμε να αντικατοπτρίσουμε οικονομικά σενάρια του πραγματικού κόσμου, εξασφαλίζοντας πραγματικές πιθανότητες για το καθένα, άρα και πλήρη κατανόηση των επιλογών μας
- Μπορούμε να εξασφαλίσουμε μελλοντικά στοιχεία για τις εκτιμήσεις που πρόκειται να παρουσιαστούν, καθώς κάνουν χιλιάδες υπολογισμούς με διαφορετικές παραμέτρους παρουσιάζοντας τις θετικές και αρνητικές πτυχές της κάθε επιλογής.
- Υπολογιστικά είναι πολύ γρήγορα καθώς μόνο ένα δείγμα υφίσταται επεξεργασία κάθε φορά.
- Δεν περιορίζεται σε τοπικά ελάχιστα η εκτέλεση τους όπως με τα ντετερμινιστικά μοντέλα.
- Επειδή δε χρειάζεται να προσθέσουμε όλο το σετ εκπαίδευσης, αλλά κάθε φορά εκχωρείτε ένα δε μπορούμε να ξεφύγουμε από τα όρια της μνήμης, όπως στα ντετερμινιστικά μοντέλα.

Μολονότι από τη χρήση τους μπορούμε να εξασφαλίσουμε σημαντικά θετικά στοιχεία, έχουν και αρνητικές επιπτώσεις τις οποίες θα πρέπει να λάβουμε υπόψιν κατά την επιλογή τους.

- Λόγω των συχνών ενημερώσεων τα βήματα προς τα ελάχιστα είναι πολύ θορυβώδη και μπορεί να οδηγήσουν σε λανθασμένες κατευθύνσεις της αποτελεσματικότητά τους.
- Χάνουν τα πλεονεκτήματα των διανυσματικών πράξεων, καθώς κάθε φορά ασχολούνται μόνο με ένα δείγμα.
- Δεν έχουν την ίδια σταθερή σύγκλιση κλίσης όπως τα ντετερμινιστικά νευρωνικά δίκτυα.
- Εξαιτίας της χρήσης όλων των πόρων για την επεξεργασία ενός δείγματος δεν είναι υπολογιστικά αποδοτικά, σε αντίθεση με τα ντετερμινιστικά που ισοκατανέμουν τους πόρους του συστήματος σε όλο το δείγμα δεδομένων.

2.2.7.3 Μαρκοβιανές αλυσίδες

Κάποιες εξίσου σημαντικές πληροφορίες σχετικά με τις Στοχαστικές Μηχανές αφορούν της Μαρκοβιανές αλυσίδες, οι οποίες αφορούν μία σημαντική παραλλαγή των Στοχαστικών Δικτύων. Η βασική τους δομή στην ιδέα της δημιουργίας τους είναι ότι για να μπορέσουμε να προβλέψουμε μελλοντικές καταστάσεις και να προχωρήσουμε στην στοχαστική ανέλιξη, κρατάμε από τις παρελθοντικές καταστάσεις μόνο αυτή που είναι πιο πρόσφατη η οποία μας καθορίζει το δρόμο για τις μελλοντικές.

Έτσι έστω ότι υπάρχει μία στοχαστική ανέλιξη $\{X_i, i = 0, 1, \dots\}$ με τιμές στο $\{0, 1, \dots\}$ οι οποίες τιμές καλούνται καταστάσεις (αν π.χ. $X_n = 1$) τότε στον χρόνο n η ανέλιξη βρίσκεται στην κατάσταση i .

Αν η $\{X_i, i = 0, 1, \dots\}$ ικανοποιεί την ιδιότητα:

$$P(X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \dots, X_0 = i_0) = P(X_{n+1} = j | X_n = i) = P_{ij} \quad (22)$$

για όλες τις καταστάσεις $j, i, i_{n-1}, \dots, i_0$ και για όλους τους χρόνους n καλείται Μαρκοβιανή αλυσίδα διακριτού χρόνου.

Η στοχαστική συμπεριφορά της μαρκοβιανής αλυσίδας περιγράφεται από τον παρακάτω πίνακα πιθανοτήτων, για τον οποίον όλες οι γραμμές έχουν άθροισμα 1:

$$P = \begin{bmatrix} p_{00} & p_{01} & \dots & p_{0j} & \dots \\ p_{10} & p_{11} & & p_{1j} & \\ \vdots & & \ddots & & \\ p_{i0} & p_{i1} & & p_{ij} & \\ \vdots & & & & \ddots \end{bmatrix}$$

Πίνακας 3 Μαρκοβιανής Αλυσίδας

2.3 Νευρωνικοί αλγόριθμοι με εκπαιδευόμενα βάρη χωρίς επίβλεψη

Η μάθηση χωρίς επίβλεψη (**unsupervised learning**) είναι η μέθοδος εκμάθησης των νευρωνικών συστημάτων, κατά την οποία τα δίκτυα εκπαιδεύονται αποκλειστικά με δεδομένα χωρίς ετικέτα σε μοτίβα. Τα δεδομένα εισόδου δεν του προσφέρουν υποδείξεις για το πρόκειται να δει, που σημαίνει ότι δεν γνωρίζουν αυτά τα δίκτυα να ξεχωρίζουν π.χ. τα αντικείμενα των εικόνων, αλλά μπορούν να κατανοήσουν ότι ένα νέο δεδομένο δεν αντιστοιχεί στα ήδη υπάρχοντα που έχουν εκπαιδευτεί. Άρα

πρόκειται για μια διαδικασία μέσω της μίμησης για τα νευρωνικά δίκτυα, μία μέθοδο που τη χρησιμοποιούν και οι άνθρωποι.

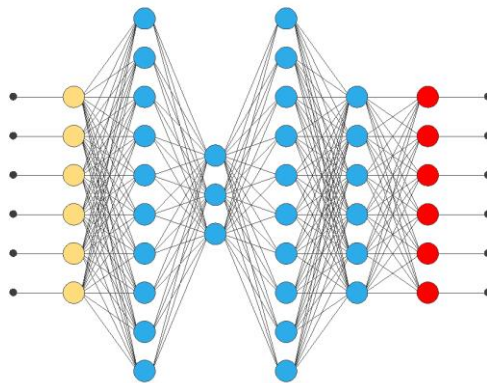
Συνεπώς, στην εκπαίδευση χωρίς επίβλεψη δε μας απασχολεί η κατηγοριοποίηση ή τοποθέτηση μιας παρατήρησης στη σωστή “θέση – σημείο”, αλλά ο διαχωρισμός των παρατηρήσεων μεταξύ τους. Εξού και ο λόγος ότι δεν αποτελεί κομμάτι της εκπαίδευσής των δικτύων χωρίς επίβλεψη η αναγνώριση των δεδομένων.

2.3.1 Συσχετιστικά Μοντέλα

Τα **Συσχετιστικά Μοντέλα** τα οποία καλούνται και ως **ASNN (Associative Neural Networks)** είναι ένα είδος δικτύων μνήμης. Δομικά προκύπτουν από τον συνδυασμό των **FFNN (Feed-Forward Neural Networks)** και των **KNN (K-Nearest Neighbor Technique)** δικτύων που αλληλοεπιδρούν με τα γειτονικά δίκτυα. Η μεθοδολογία του αφορά την σύγκριση των απαντήσεων του δικτύου σε σύνολο σε σχέση με την τεχνική του πλησιέστερου γείτονα, με σκοπό την διόρθωση της συνολικής πρόβλεψης του δικτύου σε σχέση με την εξέταση ενός κομματιού από αυτό. Άρα στην πραγματικότητα βελτιώνεται η καθολική προκατάληψη όλου του δικτύου για μια καθορισμένη περίπτωση, μέσω της εξέτασης αυτής της περίπτωσης από ένα κομμάτι του δικτύου, με σκοπό να αλλάξει συμπεριφορά γι’ αυτή τη περίπτωση όλο το δίκτυο [10].

Αυτό βοηθάει και στην προσέγγιση νέων δεδομένων όταν ενταχθούν στο δίκτυο, καθώς δε θα χρειαστεί να επαναληφθεί η εκπαίδευση για όλο το σύνολο του δικτύου, αλλά θα γίνει άμεση πρόσβαση μόνο στο σημείο του δικτύου που απασχολείται με τα αντίστοιχα δεδομένα, όπως γίνεται και στις μνήμες των υπολογιστών.

Τα πολυστρωματικά νευρωνικά δίκτυα (MLP) δεν έχουν μνήμη, συνεπώς δε μπορούν να κάνουν τις αντίστοιχες προσεγγίσεις που κάνουν τα συσχετιστικά μοντέλα. Αυτό σημαίνει ότι μετά την εκπαίδευσή τους όλες οι πληροφορίες αποθηκεύονται στα συναπτικά βάρη για τα δεδομένα εισόδου και δε μπορούν ν’ αλλάξουν. Έτσι λοιπόν το δίκτυο καταλήγει στην ετυμηγορία του και μετά δε μπορεί να διορθώσει κάποια λάθη που ενδεχομένως έχουν προκύψει από την εκπαίδευση. Για να το κάνει αυτό θα πρέπει να επαναλάβει τη διαδικασία εκπαίδευσης από την αρχή. Τα συσχετιστικά μοντέλα όμως, επειδή λειτουργούν σε μνήμες, χωρίς να χρειαστεί να επανεκπαιδεύσουν το νευρωνικό δίκτυο διορθώνουν το πρόβλημα μέσω των γειτονικών νευρώνων, προωθώντας τη βελτιωμένη εκδοχή σε όλο το δίκτυο και κάνοντάς την καθολική.



Εικόνα 11 Associative Neural Networks (ASNN)

2.3.2 Κανόνας Hebb

Το έργο του Donald Olding Hebb (1904 – 1985) το οποίο δημοσιεύτηκε κατά τα τέλη της δεκαετίας του 1940, ήταν η εναρκτήρια σπίθα για τη μεταγενέστερη μελέτη των δικτύων μάθησης που βασίζονται στην αλληλεπίδραση των γειτονικών τοπικών μοντέλων. Ουσιαστικά έγινε γνωστός για τη **Χεμπιανή Μάθηση (Hebbian Learning)** την οποία εισήγαγε το 1949.

Η βάση της θεωρίας του λέει ότι *“Αν ο άξονας του κυττάρου A βρίσκεται αρκετά κοντά έτσι ώστε να διεγείρει το κύτταρο B και επίμονα και επανειλημμένα λαμβάνει μέρος στη διεγερσή του B τότε λαμβάνει χώρα κάποια διαδικασία ανάπτυξης ή κάποια αλλαγή στο μεταβολισμό στο ένα από τα δύο ή και στα δύο κύτταρα έτσι ώστε η αποτελεσματικότητα του κυττάρου A στο να διεγείρει το κύτταρο B να αυξάνεται.”* [3] [7]

Παρά το γεγονός ότι δεν πρόλαβε να το αποδείξει επιστημονικά, μεταγενέστερα αποδείχθηκε ότι κάποιοι νευρώνες έχουν πράγματι αυτή τη συμπεριφορά, κάτι το οποίο επηρέασε τις θεωρίες και τις έρευνες της Νευροεπιστήμης.

Σύμφωνα με τον παραπάνω κανόνα, το συναπτικό βάρος μεταξύ του νευρώνα A και του νευρώνα B είναι ανάλογο με το γινόμενο των εξόδων τους που εκφράζεται και στην παρακάτω σχέση:

$$w = w + \beta \cdot a \cdot b \quad (23)$$

Το συναπτικό βάρος αυξάνεται όταν υπάρχει συμφωνία μεταξύ των νευρώνων, ενώ αν είναι αντίθετοι μεταξύ τους και δε μπορούν να συμφωνήσουν τότε μειώνεται το αρνητικό βάρος.

Η κύρια μαθηματική διατύπωση του κανόνα του Hebb είναι η ακόλουθη:

$$\Delta w_{ij}(t_n) = F(y_k(n), x_j(n)) \quad (24)$$

όπου $F(x, y)$ είναι η συνάρτηση των σημάτων εισόδου και εξόδου αντίστοιχα ή αλλιώς των προσυναπτικών και μετασυναπτικών σημάτων.

2.3.3 Ανάλυση Κυρίων Συνιστωσών (Principal Component Analysis)

Η **Ανάλυση Κύριων Συνιστωσών (Principal Component Analysis) – PCA**, είναι μία στατιστική μέθοδος που είναι ισοδύναμη με τον Μετασχηματισμό Karhunen – Loève, που χρησιμοποιείται στη μονοδιάστατη και δισδιάστατη επεξεργασία σήματος. Πρωτοεμφανίστηκε το 1901 από τον Karl Pearson (1857 – 1936) ως μία μέθοδο γραμμικής παλινδρόμησης ελαχίστων τετραγώνων σε n διαστάσεις. Μετά από περίπου 22 χρόνια ο Harold Hotelling το 1933 ανέπτυξε με τη σειρά του τη συγκεκριμένη θεωρία και πρότεινε τη μέθοδο διασποράς πολυδιάστατων τυχαίων μεταβλητών, αποκτώντας και τον τίτλο του εφευρέτη της μεθόδου PCA.

Ο κεντρικός άξονας της μεθόδου είναι η μείωση της διάστασης ενός συνόλου δεδομένων. Πιο συγκεκριμένα έστω ότι ένα σύνολο δεδομένων έχει ένα μεγάλο όγκο γνωρισμάτων (παραγόντων). Μέσα από αυτά τα γνωρίσματα και των μεταβλητών που τα περιγράφουν, υπάρχει ένα σύνολο από μη συσχετιστικές μεταβλητές, οι οποίες δεν είναι απαραίτητες για να μπορέσουμε να περιγράψουμε τα γνωρίσματα του προβλήματος. Αυτές λοιπόν οι μη συσχετιστικές μεταβλητές αποκόπονται από το βασικό σύνολο και παραμένουν όλες εκείνες οι οποίες είναι αναγκαίες για να μπορέσουμε να περιγράψουμε το πρόβλημα.

Μέσα από αυτή τη διαδικασία μειώνουμε τον όγκο των δεδομένων, αλλά και κατ' επέκταση τον χρόνο που απαιτείται για να μπορέσουμε να οδηγηθούμε στη λύση του προβλήματος. Μολονότι μειώνεται η διάσταση του διανύσματος οι πιο σημαντικές σχέσεις ή τα μοτίβα των μεταβλητών παραμένουν ανεπηρέαστα βοηθώντας απρόσκοπτα στην επίλυση του ζητήματος.

Η μέθοδος αυτή μπορεί να χρησιμοποιηθεί τόσο σε προβλήματα παλινδρόμησης όσο και σε προβλήματα ταξινόμησης [11].

2.3.3.1 Η εφαρμογή της PCA, τα πλεονεκτήματα και τα μειονεκτήματά της

Οι σημαντικότερες εφαρμογές της Ανάλυσης Κυρίων Συνιστωσών είναι οι εξής:

- Στην οπτικοποίηση πολυδιάστατων δεδομένων.
- Στο να μπορέσει να γίνει αλλαγή του μεγέθους μία εικόνας
- Στα δεδομένα που αφορούν την υγειονομική περίθαλψη, με σκοπό την καλύτερη κατανόηση των παραγόντων που οφείλονται στη αύξηση του κινδύνου στις χρόνιες νόσους.

Ορισμένα από τα βασικά πλεονεκτήματα της μεθόδου είναι που καθιστούν σημαντική την επιλογή της είναι τα εξής:

- Αυξάνει την απόδοση των αλγορίθμων καθώς απαλείφει τις περιττές μεταβλητές.
- Βοηθάει στην μείωση του θορύβου, όπου δε μπορεί να γίνει αυτόματα.
- Προσφέρει αύξηση της ταχύτητας των διαδικασιών και των αλγορίθμων.

Σε αντιπαράθεση οι αρνητικές επιπτώσεις που έχει η χρήση της μεθόδου Ανάλυσης Κυρίων Συνιστωσών είναι οι παρακάτω:

- Σε κάποιες περιπτώσεις ακόμα και μετά τον προσδιορισμό των πιο σημαντικών χαρακτηριστικών είναι δύσκολο να ερμηνεύσουμε ποια είναι αυτά.
- Ενδέχεται τα επεξεργασμένα δεδομένα να είναι πιο δύσκολα κατανοήσιμα σε σχέση με τ' αρχικά δεδομένα.

2.3.3.2 Επιπλέον είδη κανόνων

Χωρίς να επιμείνουμε στη μαθηματική ανάλυση στη συνέχεια θα παραθέσουμε ονομαστικά κάποια επιπλέον είδη κανόνων που προσφέρουν εναλλακτικές επιλογές. Κάποια από αυτά είναι:

- Ο κανονικοποιημένος κανόνας του Hebb
- Ο κανόνας Oja
- Ο Γενικευμένος Αλγόριθμος του Hebb (Generalized Hebbian Algorithm, GHA)
- Ο κανόνας APEX (Adaptive Principal component EXtraction)
- Ο κανόνας του Földiák
- Ο κανόνας του υποχώρου (subspace rule)
- Το μοντέλο Rubner
- Τα μοντέλα του Οπτικού Φλοιού (Visual Cortex), Μοντέλο Linsker

2.3.4 Ανάλυση Ανεξαρτήτων Συνιστωσών (Independent Component Analysis)

Επειδή μέσω της Ανάλυσης Κυρίων Συνιστωσών δημιουργούνται στατικώς ασυσχέτιστα σήματα, για να μπορέσει να βελτιωθεί αυτή η παθογένεια, επεκτάθηκε η μέθοδος PCA και μετατράπηκε σε ICA. Η

ICA (Independent Component Analysis) ή αλλιώς **Ανάλυση Ανεξαρτήτων Συνιστωσών** χρησιμοποιεί στατικές συναρτήσεις μεγάλου βαθμού, μέσω της οποίας γίνεται ανάκτηση ανεξάρτητων σημάτων όπου καλείται και **Τυφλός Διαχωρισμός Πηγών (Blind Source Separation)**.

2.3.4.1 Ιστορική αναδρομή και ανάλυση

Το ICA εισήχθη για πρώτη φορά τη δεκαετία του 1980 και πιο συγκεκριμένα το 1985 από τους Jeanny Héroult and Bernard Ans και εξελίχθηκε περισσότερο από τον Christian Jutten το 1985 και το 1986, με μία επιπλέον βελτίωση από τον Pierre Comon το 1991. Μέχρι όμως το 1994 που δημοσιεύτηκε το paper του [12] ήταν άγνωστο ο όρος αλλά και η μέθοδος.

Ο βασικός στόχος της μεθόδου είναι να καταφέρει να εξάγει χρήσιμες πληροφορίες ή σήματα πηγής μέσα από τα δεδομένα. Η μορφή που μπορεί να έχουν αυτά τα δεδομένα είναι εικόνων, χρηματιστηρίων, ήχων. Κατά κύριο λόγο όμως χρησιμοποιήθηκε στις εφαρμογές ιατρικών σημάτων, βιολογικών διαδικασιών και ηχητικών σημάτων. Πάραυτα παραμένει ένας αλγόριθμος μείωσης διαστάσεων όπως το PCA.

Υπάρχουν διάφοροι αλγόριθμοι που μπορούν να ικανοποιήσουν την ICA μέθοδο με κυριότερους τους FastICA και τον Infomax [13].

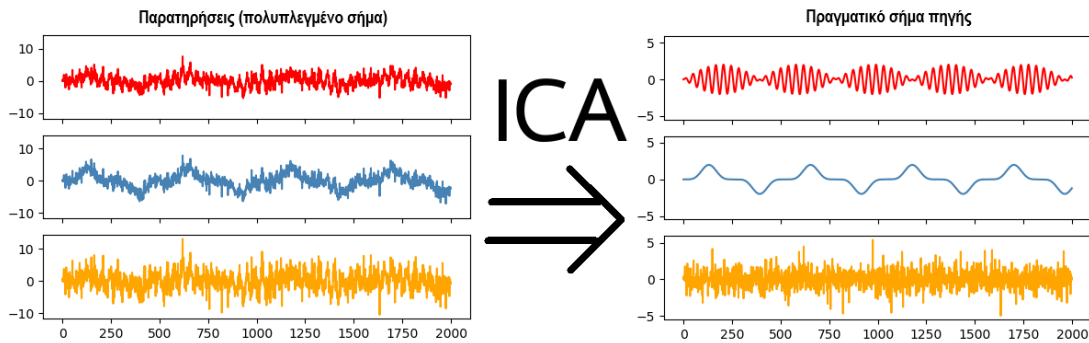
2.3.4.2 Πλεονεκτήματα και Μειονεκτήματα της μεθόδου

Μολονότι μπορεί να βοηθήσει σε πολλές πτυχές τη βελτιστοποίηση των εξαγόμενων αποτελεσμάτων, έχει και αυτή όπως και όλες οι προαναφερθείσες μέθοδοι κάποια μελανά σημεία.

- Πρώτο και κύριο πρόβλημα είναι η ελλιπής κατάρτιση της ICA ή η υπερπλήρης ICA, οι οποίες δημιουργούν πρόβλημα στη διεξαγωγή της μεθόδου, καθώς παρουσιάζουν προβληματικά αποτελέσματα είτε λόγω έλλειψης σημαντικών δεδομένων είτε γιατί μπορεί να έχουν εκπαιδευτεί άνω του αναμενομένου και αν λοξοδρομούν από τις πραγματικές αποφάσεις που πρέπει να πάρουν.
- Σε πολλές μελέτες χρησιμοποιείται η τεχνική χωρίς να υπάρχει πλήρη κατανόηση των λεπτομερειών της, με αποτέλεσμα είτε να μην γνωρίζουμε ποια είναι τα βασικά γνωρίσματα με τα οποία έπρεπε να εμπλουτιστεί είτε να μην κατανοούμε τα αποτελέσματα τα οποία κατάφερε.
- Σε προβλήματα του πραγματικού κόσμου τα οποία έχουν μεγάλο όγκο δεδομένων, η τεχνική του ICA μπορεί να αποδειχθεί υπολογιστικά ασύμφορη, αφού χρειάζεται μεγάλο όγκο πόρων για να μπορέσει αν φέρει εις πέρας τα ογκώδη σύνολα δεδομένων.

Σαφέστατα όμως υπάρχουν και θετικά αποτελέσματα από τη χρήση της, τα οποία είναι τα παρακάτω:

- Έχει τη δυνατότητα να ξεχωρίσει μία μίξη σημάτων στα ανεξάρτητα στοιχεία τους.
- Λόγω του γεγονότος ότι είναι μία μέθοδος χωρίς επίβλεψη, μπορεί να εφαρμοστεί σε δεδομένα τα οποία δεν έχουν ετικέτα, στην περίπτωση που δεν έχουμε τέτοια δεδομένα.
- Ένα επιπλέον πλεονέκτημα είναι η δυνατότητα της εξαγωγής γνωρισμάτων για τα δεδομένα τα οποία έχουμε, που σημαίνει ότι μπορούν τα δεδομένα με τα χαρακτηριστικά που προέκυψαν, να χρησιμοποιηθούν σε άλλες μεθόδους όπως είναι η ταξινόμηση.



Εικόνα 12 Μετατροπή πολυπλεγμένου σήματος στο σήμα πηγής με ICA μέθοδο

2.3.5 Ανταγωνιστικά Μοντέλα (Generative Adversarial Networks)

Τα **Ανταγωνιστικά Μοντέλα** νευρωνικών δικτύων (**Generative Adversarial Network**) – **GAN**, είναι η δεύτερη κατηγορία δικτύων μετά τα Συσχετιστικά Μοντέλα που υπάρχουν στα Τεχνητά Νευρωνικά Δίκτυα Χωρίς Επίβλεψη (Unsupervised Neural Networks). Η κυριότερη τους διαφορά σε σχέση με τα χεμπιανά είναι ότι στα χεμπιανά οι νευρώνες που αφορούν την έξοδο μπορούν να δουλεύουν συνεργατικά και να ενεργοποιούνται ταυτόχρονα πολλού μαζί. Αντίθετα στα Ανταγωνιστικά Μοντέλα αυτή η λειτουργία δεν πραγματοποιείται, καθώς κατά την έξοδο υπάρχει ανταγωνισμός των νευρώνων που έχουν πυροδοτηθεί, με αποτέλεσμα ένα μόνο απ' όλους τους ενεργοποιημένους νευρώνες να καταφέρνει κάθε χρονική στιγμή να εμφανίζεται στην τελική έξοδο [14].

Ιστορικά η εξέλιξη των GANs παρά ότι σαν ιδέα υπάρχει από τη δεκαετία του 1950, όταν ακόμα δεν υπήρχε ο υπολογιστής στη μορφή που τον ξέρουμε σήμερα, με τον Alan Turing (1912 – 1954) με τη Μηχανή Turing που είχε δημιουργήσει για την αποκρυπτογράφηση της γερμανικής κρυπτογραφικής συσκευής Enigma, ήταν τα πρώτα ψήγματα για τη δημιουργία νέων δεδομένων, παρά ότι δεν γνώριζε κανείς αυτό το εύρημα τότε. Με την πάροδο των χρόνων και τις ανάγκες για τη δημιουργία νέων δεδομένων φτάσαμε το 2014 μέσω του Ian Goodfellow και των συναδέλφων του για να γίνει πραγματικότητα η υλοποίησή τους. Η μεθοδολογία που λειτουργούν είναι μέσω δύο ξεχωριστών δικτύων, όπου το ένα παράγει πλαστά δεδομένα και το άλλο να διαχωρίζει τα πλαστά από τα διακριτά και πραγματικά δεδομένα. Αυτά τα δύο δίκτυα είναι ο Δημιουργός (Generator) και ο Διαχωριστής – Διευκρινιστής (Discriminator). Ο διαχωρισμός των υποχρεώσεων που έχουν αυτά τα δύο δίκτυα είναι οι εξής:

- **Είσοδος**
 - Ο Generator, εισάγει διανύσματα τυχαίων αριθμών.
 - Ο Discriminator, εισάγει δεδομένα από δύο πηγές:
 - Τα πραγματικά δεδομένα από ένα training dataset.
 - Τα ψευδή δεδομένα από τον Generator.
- **Έξοδος**
 - Ο Generator, εξάγει πραγματικά δεδομένα όσο το δυνατόν πιο πειστικά.
 - Ο Discriminator, προβλέπει την πιθανότητα τα δεδομένα του να είναι πραγματικά.
- **Στόχος**
 - Ο Generator, έχει ως στόχο τη δημιουργία πλαστών δεδομένων όσο το δυνατόν πιο δυσδιάκριτα και όμοια με τα πραγματικά δεδομένα εκμάθησης
 - Ο Discriminator, έχει ως στόχο τη διάκριση των πραγματικών δεδομένων σε σχέση με τα πλαστά του Generator.

Απώτερος στόχος από αυτή τη μέθοδο είναι να βρεθεί αυτή η πιθανότητα η οποία θα είναι η ελάχιστη από τον Discriminator, ώστε τα πλαστά δεδομένα του Generator, να μη μπορούν να επηρεάσουν το δίκτυο στο σύνολό του. Τότε, μπορούμε να χρησιμοποιήσουμε το παραγόμενο δίκτυο για την εξάλειψη τυχόν σφαλμάτων σε άλλα δίκτυα κ.τ.λ.

Πάρα το γεγονός ότι είναι ένα πρώιμο εργαλείο για τα μοντέλα της Τεχνητής Νοημοσύνης και των Νευρωνικών Δικτύων, έχουν υπάρξει ήδη αρκετοί τομείς όπως τα βιντεοπαιχνίδια, η επιστήμη, οι εφαρμογές που ανακαλύπτουν κακόβουλα προγράμματα αλλά και βιομηχανικές αλυσίδες που τα χρησιμοποιούν αυτά τα μοντέλα για τη βελτιστοποίηση των αναγκών τους

2.3.6 Δίκτυο Kohonen (SOM) - LVQ

Το μεγαλύτερο ποσοστό νευρωνικών αλγορίθμων που υπάρχει και έχουμε δει μέχρι τώρα έχει ο ως πρωτεύων δόγμα την έννοια του συναπτικού βάρους. Που σημαίνει ότι η πληροφορία που είναι αποθηκευμένη στις συνάψεις για να μπορέσει να αναπαρασταθεί με κάποια συνάρτηση εισόδου η εξόδου – εξόδου έχει ανάγκη από τα συναπτικά βάρη. Σε κάποιες όμως περιπτώσεις στα νευρωνικά δίκτυα του εγκεφάλου δεν παίζει πρωταρχικό ρόλο ο τρόπος που θα αναπαρασταθεί η πληροφορία για να παρουσιαστεί, αλλά για μπορέσει να συλλεχθεί, να υποστεί επεξεργασία και να παρουσιαστεί το σημαντικότερο είναι το σημείο στο οποίο θα βρίσκεται το νευρώνιο.

Όπως έχει ερευνηθεί πληροφορίες που αφορούν τον ήχο, την εικόνα, την αφή κ.τ.λ. τα διαχειρίζονται νευρώνια του εγκεφάλου που βρίσκονται σε συγκεκριμένο σημείο μέσα σε αυτόν. Συνεπώς, σε πολλές περιπτώσεις κυρίαρχο ρόλο παίζει η διάταξη των νευρώνων και όχι η δομή για την τροποποίηση της πληροφορίας. Αυτές οι δομές – συστοιχίες ονομάζονται **χάρτες**. Οι κατηγορίες οι οποίες χωρίζονται οι χάρτες είναι οι ακόλουθες:

- **Σωματοτοπικοί χάρτες**, στους οποίους καταλήγουν οι πληροφορίες που αφορούν τον πόνο, τη θερμοκρασία, την πίεση, τις δονήσεις κ.τ.λ. και γενικότερο το σύνολο των πληροφοριών που σχετίζονται με τη σωματική αίσθηση.
- **Ακουστοτοπικοί χάρτες (κοχλιοτοπικοί χάρτες)** οι οποίοι σχετίζονται με το σύνολο των πληροφοριών που αφορούν τα ηχητικά κύματα, τις συχνότητες και οτιδήποτε αφορά τον ήχο, όπου καταλήγουν στο τύμπανο για να γίνει αναγνώρισή τους.
- **Ρετινοτοπικοί χάρτες** που σχετίζονται με τη διάταξη των κυττάρων των νευρώνων στα οποία καταλήγουν πληροφορίες σχετικές με την όραση.

Ένα από τα μοντέλα που έχουν αντίστοιχη συμπεριφορά και διάταξη με τους χάρτες είναι το δίκτυο του **Kohonen (SOM)**. Ο **Teuvo Kohonen** (1934 –) είναι Φινλανδός καθηγητής ο οποίος τη δεκαετία του 1980, εισήγαγε την ορολογία **SOM (Self-Organizing Map)** Αυτό-Οργανωμένοι Χάρτες ή αλλιώς Αυτό-Οργανούμενα Δίκτυα, τα οποία είναι δίκτυα χωρίς επίβλεψη [3].

2.3.6.1 Οργάνωση και διεξαγωγή των δικτύων Kohonen

Τα δίκτυα του Kohonen πραγματοποιούν τις διεργασίες τους σε δύο ξεχωριστές φάσεις:

- Η πρώτη φάση ονομάζεται “**Φάση Εκμάθησης**”, όπου χτίζεται ο χάρτης και το δίκτυο οργανώνει την ανταγωνιστική διαδικασία χρησιμοποιώντας τα σύνολο των δεδομένων εκπαίδευσης.

- Η δεύτερη φάση ονομάζεται “**Φάση Πρόβλεψης**”, όπου στα νέα διανύσματα δίνεται η τοποθεσία πάνω στον συγκλειόμενο χάρτη, ώστε να μπορούν τα νέα δεδομένα να κατηγοριοποιηθούν ή να ταξινομηθούν ευκολότερα.

Η διάταξη που μπορεί να έχει το δίκτυο είναι μονοδιάστατη, δυσδιάστατη ή τρισδιάστατη ανάλογα τη συστοιχία που έχουν τα ανταγωνιστικά νευρώνια. Όπως έχουμε αναφέρει και προηγουμένως, η συσχέτιση ενός κόμβου με τους γειτονικούς παίζει έναν σπουδαίο ρόλο, καθώς μέσω της αλληλεπίδρασής τους έρχονται σε ανταγωνιστική κατάσταση για να αποφανθεί ποιος θα κυριαρχήσει. Ταυτόχρονα η αλληλοσυσχέτιση παίζει ρόλο και στην ενεργοποίηση των νευρώνων [9] [3].

Με την εισαγωγή των δεδομένων από το διάνυσμα εισόδου και μετά την “σύγκρουση” ένας νευρώνας ανακηρύσσεται ένας νευρώνας νικητής, καθώς σε αυτόν ταιριάζει περισσότερο το διάνυσμα εισόδου. Συνεπώς, πυροδοτείται η έξοδος που αντιστοιχεί ο συγκεκριμένος νευρώνας και οι άλλοι μένουν ανενεργοί. Σκοπός είναι να δημιουργηθούν κλάσεις – συστοιχίες ώστε τα αντίστοιχα πρότυπα στο μέλλον να αναδεικνύονται από τον συγκεκριμένο νευρώνα, διασφαλίζοντας ότι οι άλλοι νευρώνες της γειτονιάς θα τον ανακηρύξουν νικητή κάθε φορά, αλλά και θα εκπαιδευτούν για να το επιτυγχάνουν αυτό πάντα.

Για να μπορέσει να πραγματοποιηθεί η αντιστοίχιση του προτύπου εισόδου με τον νευρώνα – νικητή θα πρέπει να χρησιμοποιήσουμε ένα κριτήριο που αφορά την απόσταση. Το πιο σύνθητες κριτήριο είναι η Ευκλείδεια απόσταση του διανύσματος εισόδου και του διανύσματος βαρών $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$, $\mathbf{w} = [w_1, w_2, \dots, w_n]^T$ του νευρώνα αντίστοιχα:

$$\mathbf{d} = \|\mathbf{w} - \mathbf{x}\| \quad (25)$$

Οι νευρώνες ανταγωνίζονται μεταξύ τους και στο τέλος κερδίζει αυτός που έχει τη μικρότερη ευκλείδεια απόσταση από το πρότυπο εισόδου:

$$d_{i^*} = \min_i d_i \quad (26)$$

2.3.6.2 Κανόνας εκπαίδευσης Kohonen

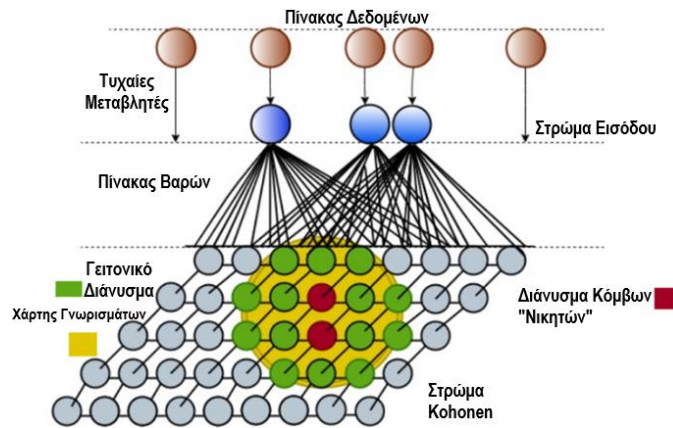
$$w_i(k+1) = w_i(k), \text{ αν } i \notin N \text{ (δεν εκπαιδεύεται)}$$

$$w_i(k+1) = w_i(k) + a(k)[x(k) - w_i(k)], \text{ αν } i \in N$$

Το αποτέλεσμα που προκύπτει είναι ότι την επόμενη φορά που θα εμφανιστεί στην είσοδο το x , τα διανύσματα $w_i(k)$ της γειτονιάς N θα είναι ακόμη πιο όμοια με το x και άρα ακόμη πιο πιθανά να νικήσουν τον ανταγωνισμό.

$$\|w_i(k+1) - x(k)\| < \|w_i(k) - x(k)\| \quad (27)$$

2.3.6.3 Τοπολογία SOM



Εικόνα 13 Τοπολογία δικτύου SOM (Self-Organizing Map)

Στην παραπάνω εικόνα παρουσιάζονται σαν σχεδιάγραμμα τα δεδομένα εισόδου, ο χάρτης Kohonen, οι συσχετίσεις μεταξύ των γειτονικών νευρώνων καθώς και οι κόμβοι νικητές

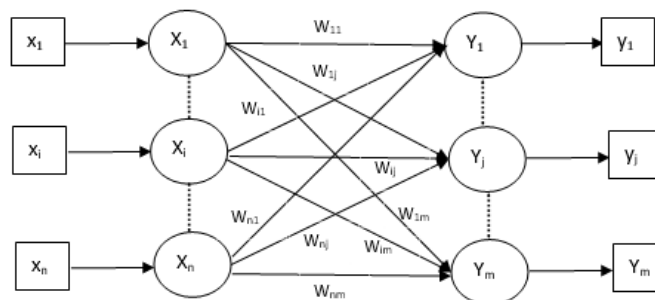
2.3.6.4 Learning Vector Quantization (LVQ)

Η εκπαίδευση μέσω του **Learning Vector Quantization (LVQ)**, δηλαδή της μάθησης κβαντικού διανύσματος, αποτελεί μία μέθοδος παρόμοια με αυτή του SOM η οποία χρησιμοποιείται μηχανική μάθησης για τη διασταύρωση της ομαδοποίησης και της ταξινόμησης και είναι ο πρόδρομος των SOM.

Βασικότερος λόγος που αναπτύχθηκε από τον **Teuvo Kohonen** ήταν για την καλύτερη κατανόηση των αλγορίθμων ταξινόμησης. Αποτελείται και αυτό από δύο φάσης – επίπεδα. Το πρώτο επίπεδο ονομάζεται ανταγωνιστικό επίπεδο, ενώ το δεύτερο επίπεδο ονομάζεται γραμμικό στρώμα.

2.3.6.5 Αρχιτεκτονική Learning Vector Quantization

Έστω ότι έχουμε το παρακάτω σχήμα:



Εικόνα 14 Αρχιτεκτονική LVQ

για τα οποία έχουμε n πλήθος μονάδων εισόδου και m πλήθος μονάδων εξόδου.

- x = το διάνυσμα εκπαίδευσης
- T = κλάση για το διάνυσμα εκπαίδευσης x

- w_j = το διάνυσμα βαρών για j-οστή εξόδου
- c_j = η γειτονική κλάση με την j-οστή

Για τον αλγόριθμο εκμάθησης ακολουθούνται τα εξής βήματα:

1. Αρχικοποιούμε τα διανύσματα αναφοράς, τα οποία γίνονται ως εξής:
 - α. Από το σύνολο δεδομένων εκπαίδευσης, παίρνουμε το πρώτο σύνολο συστοιχιών m .
 - β. Καταχωρούμε τ' αρχικά βάρη τυχαία όπως και την ταξινόμηση.
 - γ. Εφαρμόζουμε τη μέθοδο ομαδοποίησης K-means.
2. Αρχικοποιούμε το διάνυσμα αναφοράς a για το ρυθμό εκμάθησης.
3. Συνεχίζουμε με τα βήματα 4 – 9 αν η συνθήκη για τη διακοπή του αλγορίθμου δεν πληρείται.
4. Ακολουθούμε τα βήματα 5 – 6 για κάθε διάνυσμα εισόδου εκπαίδευσης του x .
5. Υπολογίζουμε το τετράγωνο της Ευκλείδειας απόστασης για τα $j = 1$ έως m και $i = 1$ έως n

$$D(j) = \sum_{i=1}^n \sum_{j=1}^m (x_i - w_{ij})^2 \quad (28)$$

6. Βρίσκουμε και αποκτάμε τον νικητή J όπου η απόσταση D_j είναι η μικρότερη.
7. Υπολογίζουμε τα νέα βάρη του νικητή:
 - α. αν $T = C_j$ τότε $w_j(new) = w_j(old) + a[x - w_j(old)]$
 - β. αν $T \neq C_j$ τότε $w_j(new) = w_j(old) - a[x - w_j(old)]$
8. Μείωση τον ρυθμό εκμάθησης a .
9. Ελέγχουμε το κριτήριο διακοπής:
 - α. Είτε μπορεί να έχουμε φτάσει το μέγιστο αριθμό εποχών.
 - β. Είτε να έχει μειωθεί τόσο πολύ ο ρυθμός εκμάθησης a σε μια αμελητέα τιμή.

2.3.7 Το δίκτυο ART

Το δίκτυο **ART (Adaptive Resonance Theory)** βασίζεται στην Θεωρία του Αυτοπροσαρμοζόμενου Συντονισμού, η οποία θεωρία αναπτύχθηκε από τον **Stephen Grossberg** (1939 -) και την **Gail Carpenter** (1948 -) περίπου το 1987 και πραγματεύεται τις διαδικασίες που εκτελεί ο εγκέφαλος για να επεξεργαστεί τις πληροφορίες που διαθέτει.

Βασική πτυχή της ανάπτυξής τους είναι η κατανόηση του πως τα βιολογικά νευρώνια διαθέτουν την ικανότητα της **πλαστικότητας**, δηλαδή την αρετή να μαθαίνουν νέα και σημαντικά ερεθίσματα όντας προσαρμοσμένα, αλλά και της **σταθερότητας** που σημαίνει ότι δεν είναι ασταθή στη συχνή τροφοδοσία από επαναληπτικά αλλά μη σχετικά ερεθίσματα αλλάζοντας τη συμπεριφορά τους.

Οι Grossberg και Carpenter ανέπτυξαν τα ART-1, ART-2 και ART-3 για να ανταπεξέλθουν σε αυτόν το προβληματισμό και να τον κατανοήσουν πλήρως. Η αρχιτεκτονική που έχουν αυτά τα δίκτυα τους δίνει τη δυνατότητα να μπορούν να ανταπεξέρχονται στην εκμάθηση νέων μοτίβων χωρίς να ξεχνάνε τα προϋπάρχοντα [15].

Η μεθοδολογία που δουλεύουν είναι με την ομαδοποίηση των δεδομένων. Έτσι κάθε κλάση αντιπροσωπεύεται από το διάνυσμα των συναπτικών βαρών ενός νευρώνα. Αυτό διάνυσμα αποκαλείται και αρχέτυπο γι' αυτήν την κλάση. Κάθε διάνυσμα για να μπορέσει να γίνει αποδεκτό σε μία κλάση θα πρέπει να ταιριάζει με το αρχέτυπο διάνυσμα της κλάσης. Στην περίπτωση που ταιριάζει σε κάποια

κλάση τότε λέμε ότι το νέο διάνυσμα συντονίζεται. Στην περίπτωση που δε συντονίζεται με κάποια κλάση, τότε δημιουργείται εκ νέου μία κλάση με αρχέτυπο το νέο ασυντόνιστο διάνυσμα με σκοπό στο μέλλον να μπορέσει να προσεγγίσει κι άλλα αντίστοιχα διανύσματα.

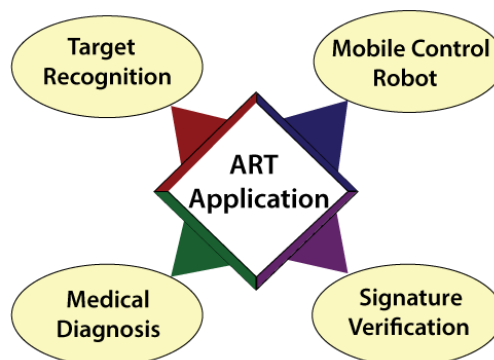
Κάποια από τα πιο γνωστά και αξιομημονένυτα μοντέλα ART που αναπτύχθηκαν τα τελευταία 20 χρόνια από τον Grossberg και την Carpenter αναφέρονται παρακάτω επιγραμματικά:

1. **ART-1**: Η πρωταρχική μορφή των ART δικτύων και πιο παλιά που διαθέτει την πιο απλή και βασική αρχιτεκτονική.
2. **ART-2**: Είναι η εξελιγμένη μορφή της ART-1 που δίνει στη δυνατότητα στα δίκτυα να ομαδοποιούν τα δεδομένα εισόδου.
3. **Fuzzy ART**: Η επαύξηση της ασαφούς λογικής και της ART.
4. **ARTMAP**: Πρόκειται για μία μορφή της ART λογικής αλλά με επίβλεψη (εποπτευομένη) όπου ένα δίκτυο ART μαθαίνει από ένα προηγούμενο δίκτυο ART. Εναλλακτικά αποκαλείται και predictive ART (προγνωστική ART).
5. **FARTMAP**: Αφορά μία αρχιτεκτονική ART που συμπεριλαμβάνει και την ασαφή λογική.

2.3.7.1 Πλεονεκτήματα των δικτύων ART

Η επιλογή της χρήσης των δικτύων που βασίζονται στην Θεωρία του Αυτοπροσαρμοζόμενου Συντονισμού μπορούν να προσφέρουν κάποια σημαντικά πλεονεκτήματα, τα οποία κάτωθι αναφέρονται συνοπτικά:

- Τα δίκτυα ART έχουν τη δυνατότητα να συντονιστούν και χρησιμοποιηθούν με διαφορετικές τεχνικές για να μπορέσουν να δώσουν ακριβή αποτελέσματα σε διάφορα προβλήματα που καλούνται να εφαρμοστούν.
- Μπορούν να χρησιμοποιηθούν στη ρομποτική, στην αναγνώριση προσώπων, στις ιατρικές διαγνώσεις, στην αυθεντικοποίηση και επαλήθευση υπογραφών αλλά και πολλές άλλες εφαρμογές.
- Όπως προαναφέρθηκε διαθέτουν σταθερότητα και επηρεάζονται από τις αστάθειες και τη μεγάλη κλίμακα των δεδομένων που μπορεί να δεχθούν στην είσοδο.
- Διαθέτει την ικανότητα της ευελιξίας στην προσαρμογή σε νέες κλάσεις όταν κρίνεται απαραίτητο, κάτι το οποίο δε διαθέτει η ανταγωνιστική μάθηση.



Εικόνα 15 Εφαρμογές του δικτύου ART

2.4 Νευρωνικοί αλγόριθμοι με σταθερά βάρη

Σε αυτή την ενότητα θα δούμε νευρωνικούς αλγορίθμους και κατ' επέκταση νευρωνικά δίκτυα, τ' οποία κατά τη διάρκεια εκπαίδευσής τους δεν αναπροσαρμόζουν τα βάρη τους αλλά παραμένουν σταθερά. Τα περισσότερα νευρωνικά δίκτυα είτε πρόκειται για δίκτυα με επίβλεψη είτε για δίκτυα χωρίς επίβλεψη, κατά τη διάρκεια της διαδικασίας εκπαίδευσης αναπροσαρμόζουν τα βάρη τους με σκοπό να φτάσουν στο ελάχιστον δυνατό σφάλμα κατά τις προσομοιώσεις τους. Τα δίκτυα που χαρακτηρίζονται με σταθερά βάρη, υπολογίζουν μία και μόνο μία φορά το διάνυσμα βαρών κατά την αποθήκευση και στη συνέχεια παραμένει ανεπηρέαστο για όλη την υπόλοιπη διαδικασία.

Τα δίκτυα τα οποία θα δούμε σε αυτή την ενότητα είναι το δίκτυο Hopfield, οι Συσχετιστικές Μνήμες και το Brain State in a Box δίκτυο, που πρόκειται για 3 δίκτυα τα οποία κρατάνε σταθερά τα συναπτικά τους βάρη, χωρίς να τα αναπροσαρμόζουν σε κάθε εποχή.

2.4.1 Το δίκτυο Hopfield

Το νευρωνικό δίκτυο του Hopfield είναι εμπνευσμένο από τη Στατιστική Φυσική (spin glasses), που προτάθηκε από τον **John Joseph Hopfield** (1933 -) ο οποίος είναι Πολωνός φυσικός. Αποτελείται από ένα στρώμα n συνδεδεμένων επαναλαμβανόμενων νευρώνων. Χρησιμοποιείται κατά κύριο λόγο σε καταστάσεις που χρίζουν αντιμετώπισης μέσω αυτόματης συσχέτισης και βελτιστοποίησης και εντάσσεται στις μη γραμμικές μνήμες αυτοσυσχέτισης [3] [13].

2.4.1.1 Αρχιτεκτονική του δικτύου Hopfield και η εκπαίδευση

Τα δίκτυα αυτού του τύπου διαθέτουν ένα μόνο επίπεδο για το οποίο οι συνδέσεις των νευρώνων είναι αμφίδρομες και όλοι οι νευρώνες λειτουργούν ταυτόχρονα σαν είσοδοι αλλά και σαν έξοδοι. Τα δίκτυα αυτά έχουν τόσα νευρώνια όσες είναι και οι είσοδοί τους.

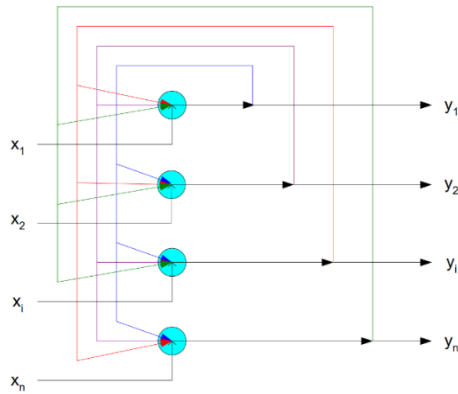
Τα διανύσματα εισόδου πρέπει να είναι διπολικά, δηλαδή με στοιχεία στο $\{-1, +1\}$. Ταυτόχρονα είναι επανατροφοδοτούμενα γιατί το κάθε νευρώνιο είναι συνδεδεμένο με όλα τα άλλα μέσω συναπτικών βαρών εκτός από τον εαυτό του.

Η εκπαίδευση του δικτύου Hopfield γίνεται μέσω του κανόνα του Hebb (ή κανόνα εξωτερικών γινομένων):

$$w_{ij} = \sum_{p=1}^m x_j^{(p)} x_i^{(p)} \quad \forall j \neq i, \text{ με } w_{ii} = 0 \quad (29)$$

όπου w_{ji} είναι το συναπτικό βάρος ανάμεσα στο νευρώνιο i και στον νευρώνα j , $x_i^{(p)} \in \{-1, +1\}$ είναι το i στοιχείο του p -οστού προτύπου διανύσματος $x^{(p)}$ και M είναι το πλήθος των προτύπων διανυσμάτων που θα αποθηκευτούν στο δίκτυο [9].

Η χωρητικότητα της συσχετιστικής μνήμης έχει υπολογιστή στο $0.15N$, όπου N είναι το πλήθος των νευρώνων του δικτύου. Δηλαδή αν έχουμε ένα δίκτυο με $N = 300$ τότε περίπου μπορούν να αποθηκευτούν 45 πρότυπα. Η χωρητικότητα αυτή είναι μικρότερη από τη μνήμη CMM ή της μνήμης του γενικευμένου ανάστροφου.



Εικόνα 16 Τοπολογία δικτύου ενός επιπέδου Hopfield με n νευρώνες

2.4.1.2 Λοιπές πληροφορίες για τα μοντέλα Hopfield

Τα δίκτυα Hopfield διακρίνονται σε διακριτά και σε συνεχή μοντέλα. Τα διακριτά μοντέλα διαχωρίζονται σε επιμέρους διπολικά και δυαδικά μοντέλα, ενώ τα συνεχή μοντέλα δεν έχουν κάποια επιπλέον κατηγοριοποίηση.

Το διακριτό μοντέλο χρησιμοποιεί συναρτήσεις προσήμου, όπως είναι η βηματική συνάρτηση. Αντίθετα το συνεχές χρησιμοποιεί τη σιγμοειδή συνάρτηση. Αυτό δίνει τη δυνατότητα στο συνεχές δίκτυο να λαμβάνει τιμές μεταξύ του $[0, 1]$ και όχι μόνο διακριτές τιμές.

Στα πλεονεκτήματα των δικτύων Hopfield είναι ο εύκολος σχεδιασμός τους καθώς και η εφαρμογής τους. Η χρησιμότητά τους έγκειται σε εφαρμογές βελτιστοποίησης, αναγνώρισης προσώπων κ.τ.λ. Υπάρχει ένας προβληματισμός ως προς τον χρόνο για την εκτέλεσή τους καθώς πρόκειται για πολύ αργές μεθόδους εκτέλεσης μιας και μπορούν να αναγνωρίσουν ένα πρότυπο τη φορά. Επιπροσθέτως, εφαρμογές με μεγάλο όγκο στοιχείων και μεγάλα πρότυπα δε μπορούν να αποθηκευτούν στις μνήμες τους και τα καθιστούν πρακτικά αδύνατα.

Υπάρχουν και κάποιες άλλες εφαρμογές των δικτύων Hopfield όπου κάποιες από αυτές αναφέρονται ονομαστικά και είναι:

- Δίκτυο Hopfield σε συσχετιστική μνήμη.
- Δίκτυο Hopfield σε μηχανή βελτιστοποίησης.
- Το δίκτυο MAXNET, που είναι μία ειδική κατηγορία του δικτύου Hopfield.

2.4.2 Συσχετιστικές Μνήμες

Οι **Συσχετιστικές Μνήμες (Associative Memories)** είναι συστήματα αποθήκευσης και ανάκτησης προτύπων εμπνευσμένα από την ομώνυμη ψυχολογική έννοια. Όταν δεχόμαστε κάποια ερεθίσματα που ενεργοποιούν τις Συσχετιστικές Μνήμες, τότε μπορεί να ανακληθούν μνήμες σχετικές με αυτές τις εισόδους – ερεθίσματα. Επίσης, είναι γνωστές και με τον όρο CAM (Content Associative Memory) και έχει ως βασικό στόχο τη βελτιστοποιημένη αναζήτηση μέσω δεδομένων, σε αντίθεση με την παροχή μιας απλής πρόσβασης στα δεδομένα βάσης της διεύθυνσης.

Πρόκειται για γραμμικά νευρωνικά δίκτυα και στα μοντέλα που εκτελείται δεν υπάρχει κάποιο σχετικό κατώφλι, όπως π.χ. στο Perceptron για το οποίο πυροδοτείται ο νευρώνας. Οι μνήμες αυτές ανήκουν στα επιβλεπόμενα δίκτυα μάθησης.

2.4.2.1 Ταξινόμηση των Συσχετιστικών Μνημών

Υπάρχουν διαφορετικά είδη Συσχετιστικών Μνημών και για την ακρίβεια πρόκειται για τρεις διαφορετικές κατηγορίες που οι κάθε μία διαθέτει δύο ξεχωριστές εξισώσεις που την επαληθεύουν. Αυτές οι εξισώσεις είναι η “Μνήμη Γενικευμένου Αντιστρόφου” (GIM) και η “Μνήμη Πίνακα Συσχέτισης” (CMM).

1. **Μνήμες Αυτοσυσχέτισης:** Σε αυτή την κατηγορία η κάθε είσοδος συσχετίζεται με τον εαυτό της, δηλαδή $Y = X$.
 - Μνήμη Γενικευμένου Αντιστρόφου: $W = XX^+$
 - Μνήμη Πίνακα Συσχέτισης: $W = XX^T$
2. **Μνήμες Ετεροσυσχέτισης:** Σε αυτήν την κατηγορία οι εισοδοί είναι διαφορετικές από τις εξόδους, $Y \neq X$.
 - Μνήμη Γενικευμένου Αντιστρόφου: $W = YX^+$
 - Μνήμη Πίνακα Συσχέτισης: $W = YX^T$
3. **Μνήμες Ταξινόμησης:** Σε αυτή την κατηγορία η κ-ιοστή είσοδος συσχετίζεται με την κ-ιοστή στήλη του I (το μοναδιαίο διάνυσμα e_k), δηλαδή $Y = I$.
 - Μνήμη Γενικευμένου Αντιστρόφου: $W = X^+$
 - Μνήμη Πίνακα Συσχέτισης: $W = X^T$

2.4.2.2 Τρόπος λειτουργίας των Συσχετιστικών Μνημών

Στις συμβατικές μνήμες του υπολογιστή τα δεδομένα αποθηκεύονται σε συγκεκριμένες τοποθεσίες, οι οποίες ονομάζονται διευθύνσεις, και ανακτώνται με αναφορές που γίνονται στις διευθύνσεις αυτών. Σε μία συσχετιστική μνήμη, τα δεδομένα αποθηκεύονται όλα μαζί με πρόσθετες ετικέτες ή μεταδεδομένα (δηλαδή περιγραφές που αναφέρονται στα περιεχόμενα των δεδομένων). Όταν θα εκτελεστεί μία αναζήτηση, η συσχετιστική μνήμη θα συγκρίνει το ερώτημα αναζήτησης με τις ετικέτες όλων των αποθηκευμένων δεδομένων και θα ανακτήσει τα περιεχόμενα αυτών των δεδομένων που ταιριάζουν καλύτερα στο πρότυπο της εισόδου [10].

2.4.2.3 Πλεονεκτήματα – Μειονεκτήματα και Εφαρμογές των Συσχετιστικών Μνημών

Τα βασικότερα πλεονεκτήματα των Συσχετιστικών Μνημών είναι:

- Είναι πολύ χρήσιμες όταν χρειάζεται να πραγματοποιηθούν παράλληλες αναζητήσεις.
- Βοηθάει στην επιτάχυνση βάσεων δεδομένων με μεγάλο όγκο και περιεχόμενο δεδομένων.
- Χρησιμεύουν σε περιπτώσεις που ο χρόνος των αναζητήσεων είναι μεγάλος και πρέπει να μειωθεί ώστε να επιταχυνθούν οι διαδικασίες.

Τα κυριότερα μειονεκτήματα τους είναι:

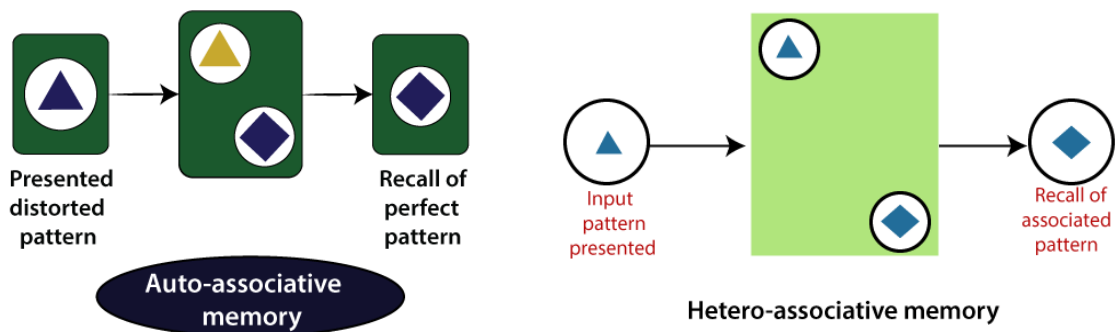
- Έχουν μεγαλύτερο κόστος σε σχέση με τις μνήμες RAM.
- Είναι αναγκαίο κάθε κελί να έχει τα κατάλληλα εργαλεία για να μπορέσει να πραγματοποιήσει τις κατάλληλες συγκρίσεις με τα δεδομένα που εισέρχονται.

Κάποιες από τις σημαντικότερες εφαρμογές των Συσχετιστικών Μνημών είναι:

- Η εφαρμογή τους στην υγειονομική περίθαλψη, μέσω της ανάλυσης σύνθετων ιατρικών δεδομένων, για να μπορέσουν να ανιχνευτούν σοβαρές ασθένειες.
- Στα χρηματοοικονομικά, μέσω της ανακάλυψης μοτίβων που ελλοχεύουν για την εξαπάτηση πελατών.
- Στο λιανεμπόριο, με τη μορφή προτύπων αγοράς που θα βοηθήσουν στην καλύτερη εμπειρία αγοράς των πελατών.
- Σε μεγάλες βάσεις δεδομένων, ώστε να μπορέσει να γίνει η προσπέλαση πιο γρήγορη για την ανάκτηση αναγκαίων δεδομένων.

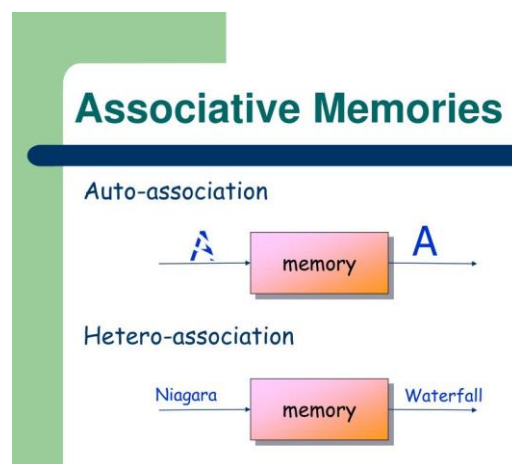
2.4.2.4 Παραδείγματα Αυτοσυσχετιστικών και Ετεροσυσχετιστικών Μνημών

Παράδειγμα Α



Εικόνα 17 Συσχετιστικές Μνήμες Α

Παράδειγμα Β



Εικόνα 18 Συσχετιστικές Μνήμες Β

2.4.3 Brain State in a Box

Το **Brain State in a Box** ή αλλιώς **BSB**, είναι ένα νευρωνικό δίκτυο που ανήκει στην κατηγορία των μη γραμμικών αυτό-συσχετιστικών νευρωνικών δικτύων που αποτελείται από ένα επίπεδο, αλλά μπορεί να μετατραπεί και σε ετεροσυσχετιστικό μοντέλο με δύο ή και περισσότερα επίπεδα.

2.4.3.1 Ιστορική Αναδρομή και Περιγραφή

Οι πρώτοι που το πρότειναν ήταν οι **J.A Anderson, J.W. Silverstein, S.A Ritz** και **R.S. Jones** το 1977 ως ένα μοντέλο συσχετιστικής μνήμης το οποίο είναι παρόμοιο με το μοντέλο Hopfield.

Στα κύρια χαρακτηριστικά του είναι ότι όλοι οι κόμβοι του ενημερώνονται ταυτόχρονα και οι νευρώνες του παίρνουν τιμές στο διάστημα $[-1, +1]$.

Επιπρόσθετα οι νευρώνες περιέχουν τη μη γραμμική συνάρτηση ενεργοποίησης f_{box} και οι έξοδοί του ανατροφοδοτούνται στην είσοδο. Η τροχιά του δικτύου περιορίζεται να είναι στον υπερκύβο $H_n = [-1, +1]^n$ εξού και το όνομά του. Με αυτό τον τρόπο το δίκτυο δεν πρόκειται ποτέ να αποκλίνει στο άπειρο [16].

Οι παράμετροι του δικτύου συγκεντρώνονται στην ακόλουθη συνάρτηση [3]:

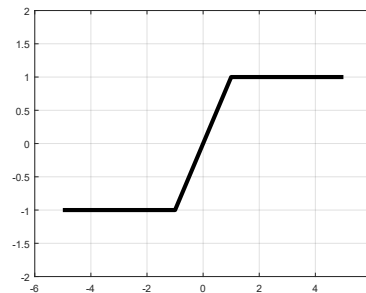
$$x(k+1) = g(x(k) + aWx(k)), x(0) = x_0 \quad (30)$$

όπου

- x_0 : αρχική κατάσταση
- $x(k) \in \mathbb{R}^n$: η κατάσταση που βρίσκεται το BSB δίκτυο τη χρονική στιγμή k
- $a \in \mathbb{R}$: βήμα – ρυθμός εκμάθησης
- $W \in \mathbb{R}^{n \times n}$: ο συμμετρικός πίνακας βαρών
- $g: \mathbb{R}^n \rightarrow \mathbb{R}^n$: συνάρτηση ενεργοποίησης (γραμμική συνάρτηση κορεσμού)

Η γραμμική συνάρτηση έχει τις εξής μορφές:

$$(g(x))_i = (sat(x))_i = \begin{cases} 1 & \text{αν } x_i \geq 1 \\ x_i & \text{αν } -1 < x_i < 1 \\ -1 & \text{αν } x_i \leq -1 \end{cases} \quad (31)$$



όπου x_i το i -οστό στοιχείο του διανύσματος x .

Κεφάλαιο 3

Αποσαφήνιση Εννοιών

Σε αυτό το κεφάλαιο θα αποσαφηνίσουμε κάποιες έννοιες και θα αναπτύξουμε τους ορισμούς σε κάποιες ορολογίες που είτε έχουν αναφερθεί είτε θα αναφερθούν στα κεφάλαια αυτής της διπλωματικής. Σκοπός είναι η καλύτερη κατανόηση αυτών μέσω μίας περιεκτικής αναφοράς τους.

Ντετερμινιστικό μοντέλο

Το μοντέλο το οποίο έχει ως κύριο σκοπό την προσπάθεια της μοντελοποίησης ενός φαινομένου – κατάστασης μέσω κάποιας αναλυτικής συνάρτησης για την οποία η κατάσταση είναι καθορισμένη εξ' αρχής και δεν πρόκειται να αλλάξει μελλοντικά, ονομάζεται **Ντετερμινιστικό μοντέλο**. Οι άγνωστοι όροι του μοντέλου που υπάρχουν πρέπει να εκτιμηθούν και να βρεθεί η βέλτιστη λύση για τις άγνωστες παραμέτρους που υπάρχουν.

Στοχαστικό μοντέλο

Ένα μοντέλο το οποίο για την ερμηνεία του φαινομένου που μελετάει έχει ως ανάγκη την προσέγγιση μέσω στατιστικής διαδικασίας, για το οποίο ενδέχεται να υπάρχουν περισσότερες από μια μελλοντικές καταληκτικές καταστάσεις, τότε το μοντέλο αυτό ονομάζεται **Στοχαστικό μοντέλο**. Κατά κύριο κανόνα τα Στοχαστικά Μοντέλα ακολουθούν την κατανομή του Gauss και η συνάρτηση που τα ερμηνεύει ονομάζεται εμπειρική συνάρτηση.

Μέση τιμή (Mean)

Ως **Μέση Αριθμητική Τιμή** ή **Μέσος Όρος** καλείται το πηλίκο της διαίρεσης από το άθροισμα των επιμέρους τιμών μίας μεταβλητής ως προς το συνολικό πλήθος των επιμέρους τιμών. Ο συμβολισμός της είναι με \bar{x} και ο τύπος της περιγράφεται ως εξής:

$$\bar{x} = \frac{\sum x_i}{N} \quad (32)$$

όπου x_i με $i = 0, 1, 2, \dots, N$ η εκάστοτε τιμή μια μεταβλητής και N το πλήθος των τιμών.

Η παραπάνω μέση τιμή ονομάζεται **Μέση τιμή διακριτής μεταβλητής**.

Όταν σε μία κατανομή υπάρχουν κλάσεις που ομαδοποιούν τις τιμές τώρα ο τύπος γίνεται:

$$\bar{x} = \frac{\sum x_i \cdot v_i}{\sum v_i} \quad (33)$$

όπου x_i είναι το μέσο του κέντρου της κλάσης (πχ αν το κάτω άκρο μία κλάσης είναι το 10 και το άνω άκρο το 20 τότε το $x_i = 15$) και το v_i αντιπροσωπεύει το πλήθος των τιμών που ανήκουν σε αυτή την κλάση.

Αυτή η μέση τιμή ονομάζεται **Μέση τιμή συνεχούς μεταβλητής**.

Τυπική Απόκλιση (Standard Deviation)

Η **απόκλιση** s^2 από ένα σύνολο παρατηρήσεων είναι ο μέσος όρος των τετραγώνων των αποκλίσεων των παρατηρήσεων από το μέσο όρο τους. Συμβολικά, η απόκλιση από n παρατηρήσεις x_1, x_2, \dots, x_n είναι

$$s^2 = \frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + \dots + (x_n - \bar{x})^2}{n - 1} \quad (34)$$

Εναλλακτικά,

$$s^2 = \frac{1}{n - 1} \sum (x_i - \bar{x})^2 \quad (35)$$

Η **τυπική απόκλιση** s είναι η τετραγωνική ρίζα της απόκλισης s^2

$$s = \sqrt{\frac{1}{n - 1} \sum (x_i - \bar{x})^2} \quad (36)$$

Μέσο Τετραγωνικό Σφάλμα (Mean Squared Error – MSE)

Το **Μέσο Τετραγωνικό Σφάλμα** ορίζεται ως η μέση τιμή του τετραγώνου της διαφοράς των εκτιμήσεων από τις πραγματικές τιμές, προς το πλήθος των πραγματικών τιμών.

$$MSE = \frac{1}{N} \sum_{i=1}^N (Y_i - \bar{Y}_i)^2 \quad (37)$$

Ρίζα Μέσου Τετραγωνικού Σφάλματος (Root Mean Square Error – RMSE)

Ρίζα Μέσου Τετραγωνικού Σφάλματος καλείται η τετραγωνική ρίζα του μέσου όρου των τετραγώνων των σφαλμάτων των παρατηρήσεων. Επειδή βρίσκεται στην ίδια κλίμακα με τα δεδομένα έχει πλεονέκτημα σε σχέση με το MSE.

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (Y_i - \bar{Y}_i)^2} \quad (38)$$

Το RMSE είναι ένα μέγεθος που θα το χρησιμοποιήσουμε κατά κόρον στις παρατηρήσεις που θα κάνουμε στο πειραματικό κομμάτι.

Ακρίβεια – Ευστοχία (Accuracy)

Με τον όρο **Ακρίβεια – Ευστοχία (Accuracy)** το πηλίκο της διαίρεσης του πλήθους των σωστών προβλέψεων ως προς το πλήθος των προβλέψεων που ένα μοντέλο καταφέρνει ως αποτέλεσμα.

$$Ακρίβεια = \frac{Πλήθος σωστών προβλέψεων}{Πλήθος προβλέψεων} \quad (39)$$

Το Accuracy θα το χρησιμοποιήσουμε κατά κόρον στο σκέλος των προβλημάτων της ταξινόμησης με το μορφή του % στο πειραματικό κομμάτι ως μέγεθος αποτελεσματικότητας.

Regression (Παλινδρόμηση)

Regression (Παλινδρόμηση) είναι η μέθοδος κατά την οποία ο στόχος του αποτελέσματος είναι να βρεθεί ένα πραγματικό διάνυσμα το οποίο θα συνδέσει τις εξαρτημένες μεταβλητές (μεταβλητές στόχους) με μία ή περισσότερες ανεξάρτητες μεταβλητές (προβλεπτικές μεταβλητές).

Υπάρχουν δύο είδη παλινδρόμησης, η γραμμική και η λογιστική παλινδρόμηση. Η γραμμική παλινδρόμηση, η οποία ονομάζεται και απλή γραμμική παλινδρόμηση (simple linear regression), είναι μια μορφή παλινδρόμησης κατά την οποία υπάρχει μόνο μία ανεξάρτητη μεταβλητή (input – independent variable) X και η εξαρτημένη μεταβλητή (response dependent variable) Y οι οποίες συνδέονται με μία γραμμική συνάρτηση.

Στη λογιστική παλινδρόμηση (logistic regression) αντί να χρησιμοποιηθεί το γραμμικό μοντέλο προσέγγισης, χρησιμοποιείται η θεωρία των πιθανοτήτων (μη γραμμικό μοντέλο), κατά την οποία ψάχνουμε να βρούμε την πιθανότητα ένα γεγονός να εμφανίζεται ή όχι στα δεδομένα που έχουμε. Σε αυτήν η εξαρτημένη μεταβλητή είναι δυαδική με $Y = 0$ ή $Y = 1$ που αντιπροσωπεύουν την ύπαρξη ή μη του γεγονότος μέσα στα δεδομένα [13].

Classification (Ταξινόμηση)

Classification (Ταξινόμηση) είναι η μέθοδος κατά την οποία γίνεται αντιστοίχιση των εισαγόμενων δεδομένων στον στόχο που υπάρχει, όπου για το κάθε δεδομένο το διάνυσμα που το περιγράφει το κατατάσσει σε μία κλάση – κατηγορία – ετικέτα (label). Σε αυτή την περίπτωση λέμε ότι είναι ταξινόμηση μονής ετικέτας (single label classification).

Στην περίπτωση όπου το διάνυσμα εισόδου του κάθε δεδομένου αντιστοιχίζεται με περισσότερες από μία ετικέτες, δηλαδή μία ομάδα – υποσύνολο ετικετών, τότε αυτή η ταξινόμηση ονομάζεται πολλαπλών ετικετών (multi-label classification).

Μάθηση Με Επίβλεψη (Supervised Learning)

Τα δίκτυα για τα οποία είναι γνωστή εκ των προτέρων η κατηγορία στην οποία ανήκουν τα δεδομένα, δηλαδή για κάθε είσοδο που τροφοδοτείται το δίκτυο γνωρίζουμε σε ποια κατηγορία πρόκειται ν' ανήκει και δεν υπάρχουν αμφιβολίες γι' αυτό, η μάθηση που περιγράφει αυτή τη μέθοδο ονομάζεται **μάθηση με επίβλεψη** ή αλλιώς **επιβλεπόμενη μάθηση (supervised learning)**. Τα βάρη και οι πολώσεις του δικτύου προσαρμόζονται με τέτοιο τρόπο ώστε να τείνουν ή ακόμα καλύτερα να φτάνουν στο επιθυμητό αποτέλεσμα. Συνεπώς, για τα δίκτυα αυτά υπάρχει έξοδος η οποία είναι και η ετικέτα για το κάθε δεδομένο ξεχωριστά.

Μάθηση Χωρίς Επίβλεψη (Unsupervised Learning)

Τα δίκτυα για τα οποία η μάθηση η οποία ακολουθείται έχει ως σκοπό να την ομαδοποίηση και/ή την εξαγωγή χαρακτηριστικών για τα ομαδοποιημένα δεδομένα, ονομάζεται **μάθηση χωρίς επίβλεψη** ή αλλιώς **μη επιβλεπόμενη μάθηση (unsupervised learning)**. Με τη μάθηση αυτή τα δίκτυα δημιουργούν πρότυπα ή αλλιώς μοτίβα, τα οποία δεν είναι γνωστό αν είναι υπαρκτά ή μη. Και αυτού του είδους η μάθηση πραγματοποιείται για τα δίκτυα στα οποία τα δεδομένα που διαχειρίζονται δεν έχουν ετικέτα, δηλαδή δεν ανήκουν σε κάποια κατηγορία, συνεπώς σκοπός είναι η αναγνώριση των συσχετίσεων που έχουν μεταξύ τους τα δεδομένα από το δίκτυο.

Ρυθμός Εκμάθησης (Learning Rate – LR)

Ο **ρυθμός εκμάθησης (learning rate – LR)** είναι το μέγεθος το οποίο εφαρμόζεται στα νευρωνικά δίκτυα που έχουν εκπαιδευόμενα βάρη, με τον κυριότερο σκοπό να είναι η αναπροσαρμογή των διανυσμάτων των βαρών για να μπορέσουν να βελτιστοποιηθούν αυτά, και κατ' επέκταση να αποκτήσει καλύτερη γνώση μέσα από την εκπαίδευσή του το νευρωνικό δίκτυο. Ο ρυθμός εκμάθησης είναι μία μονάδα που συμβολίζεται με το α και οι τιμές που μπορεί να πάρει είναι $[0, 1]$. Όσο πιο κοντά είναι το α στο 0 τόσο πιο μικρό θα είναι το βήμα βελτίωσης και αναπροσαρμογής των βαρών, με συνέπεια την καλύτερη αλλά και πιο αργή εκπαίδευση. Αντίθετα, όσο πιο κοντά είναι στο 1 το α τόσο πιο μεγάλο θα είναι το βήμα βελτίωσης των βαρών, άρα και λιγότερο εξονυχιστική η εκπαίδευση. Μολονότι με μικρές τιμές του ρυθμού εκμάθησης μπορούμε να πάρουμε ένα πιο εκπαιδευμένο δίκτυο, αυτό δεν είναι και αυτοσκοπός, καθώς μπορεί να δημιουργήσει overfitting η αργή εκμάθηση. Αντιθέτως, με μεγάλο ρυθμό εκμάθησης μπορεί το δίκτυό να μην καταφέρει να εκπαιδευτεί σωστά.

Overfitting (Υπερπροσαρμογή)

Το **overfitting (υπερπροσαρμογή)** είναι το φαινόμενο το οποίο παρατηρείται στην εκπαίδευση των νευρωνικών δικτύων, τα οποία καταλήγουν σε υπερβολική προσαρμογή στο μοντέλο που εξετάζουν, με αποτέλεσμα να αποκλίνουν από τα επιθυμητά αποτελέσματα και κατ' επέκταση τη σωστή αποτελεσματικότητα. Η συνέπεια η οποία έχει το overfitting σε ένα νευρωνικό δίκτυο είναι η υπερβολική γνώση για τις λεπτομέρειες των δεδομένων που εκπαιδεύεται, με αποτέλεσμα να μη μπορεί να προσεγγίσει νέα δεδομένα στα οποία θα κληθεί να προσομοιώσει.

Σύγκλιση (Convergence)

Η **σύγκλιση (convergence)** είναι η κατάσταση στην οποία ένα νευρωνικό δίκτυο φτάνει καταφέροντας να προσεγγίσει τα επιθυμητά αποτελέσματα, μέσα από τα πρότυπα που έχει δημιουργήσει από τα δεδομένα που έχει εκπαιδευτεί. Η σύγκλιση μπορεί να αφορά ένα τοπικό ελάχιστον για έναν αλγόριθμο εκμάθησης, το οποίο δεν εγγυάται και τη σύγκλιση όλο του συστήματος.

Απόκλιση (Divergence)

Η **απόκλιση (divergence)** είναι η κατάσταση κατά την οποία ένα νευρωνικό δίκτυο φτάνει όταν δεν καταφέρει να προσεγγίσει τα επιθυμητά δεδομένα καθώς απομακρύνονται οι αποφάσεις του από αυτές που θα έπρεπε να πάρει. Είναι το αντίθετο φαινόμενο από τη σύγκλιση και για να μπορέσει να αντιμετωπιστεί είναι αναγκαίο το σφάλμα που έχει προκύψει από τις προβλέψεις, να μπορέσει να βελτιστοποιηθεί και να μειωθεί στο ελάχιστο δυνατό, ώστε να εξασφαλίσουμε τα σωστά αποτελέσματα.

Κεφάλαιο 4

Backpropagation και Παραλλαγές

Ένας από τους πιο σημαντικούς αλγόριθμους εκμάθησης των νευρωνικών δικτύων, αν όχι ο πιο σημαντικός, αλλά και διαδεδομένος είναι ο αλγόριθμος Backpropagation. Η σημαντικότητα του αυτή προκύπτει από το γεγονός ότι είναι πολύ ακριβής, το δίκτυο που χρειάζεται για να εκτελεστεί είναι πολύ συμπαγές και όχι άσκοπα μεγάλο με μεγάλο όγκο νευρώνων και η εκπαίδευσή του δεν καταναλώνει πολύ χρόνο ώστε να μπορέσει να μας παραδώσει πολύ ικανοποιητικά αποτελέσματα.

Όπως αναφέραμε και στην **Ενότητα 2.2.3**, ο Backpropagation είναι ένα οπισθοτροφοδοτούμενο νευρωνικό δίκτυο, που σημαίνει ότι επανατροφοδοτεί τις εισόδους από το feedback των εξόδων για να αναπροσαρμόσει τα συναπτικά βάρη, μειώνοντας το σφάλμα από τις προσεγγίσεις που είχαν γίνει και φτάνοντας εν τέλει σε καλύτερη προσέγγιση – πρόβλεψη.

Όπως προαναφέραμε στην **Ενότητα 2.2.3**, ο αλγόριθμος αυτός προτάθηκε **Paul Werbos** (1947 –) τη δεκαετία του 1970 οποίος είναι Αμερικανός επιστήμονας και πρωτοπόρος στη μηχανική μάθηση. Τη δεκαετία του 1980 η μέθοδος αυτή έγινε αντιληπτή ότι μπορεί να εφαρμοστεί και στα δίκτυα πολλαπλών στρωμάτων, και έγινε η πιο γνωστή και διαδεδομένη. Πρόκειται για έναν αλγόριθμο που ανήκει στην κατηγορία των αλγορίθμων με επίβλεψη, καθώς έχει ως σκοπό την εύρεση της κατάλληλης εξόδου από τα δεδομένα εισόδου χρησιμοποιώντας το πρότυπο μαθαίνει και όχι της αλληλοσυσχέτισης των εισαγόμενων δεδομένων μεταξύ τους.

Στην ενότητα αυτή θα αναπτύξουμε κάποιες μεθόδους και θεωρήματα που χρησιμοποιήσαμε στο πειραματικό κομμάτι στη σύγκριση και αποτίμηση των Backpropagation παραλλαγών με τον ELM.

4.1 Θεώρημα Kolmogorov

Ο **Andrey Nikolaevich Kolmogorov** (1903 – 1987) ήταν ένας Σοβιετικός μαθηματικός, ο οποίος διέπρεψε στον τομέα της θεωρίας των πιθανοτήτων, στην κλασσική μηχανική, στους αλγόριθμους των υπολογιστών αλλά και στην πολυπλοκότητα τους.

Ανέπτυξε ένα θεώρημα σημαντικό για τα νευρωνικά δίκτυα στο οποίο λέει ότι:

«Εστω η συνεχής συνάρτηση $g: [0, 1]^n \rightarrow \mathbb{R}^m$, $g(x) = y$. Η g μπορεί να υλοποιηθεί επακριβώς από ένα νευρωνικό δίκτυο τριών στρωμάτων εμπρόσθιας τροφοδοσίας με n εισόδους στο πρώτο στρώμα, $2n + 1$ νευρώνια στο κρυφό στρώμα και m νευρώνια στο τελευταίο στρώμα (στρώμα εξόδου)»

Ωστόσο, για το παραπάνω δίκτυο το οποίο ισχυρίζεται ότι υπάρχει σαν αρχιτεκτονική, δε μας λέει πως να υπολογίσουμε τις παραμέτρους του [17].

4.2 Κατηγορίες του Backpropagation

Για την εκπαίδευση του Backpropagation υπάρχουν ποικίλες μέθοδοι οι οποίες μπορούν να προσεγγίσουν τους στόχους ενός δικτύου το οποίο καλείται να εκπαιδευτεί με αυτόν. Κατά τη διάρκεια των χρόνων, όπως όλοι οι αλγόριθμοι έτσι και ο Backpropagation έχει δεχθεί πολλές βελτιώσεις, πάντα με γνώμονα την επίτευξη του τελικού στόχου με όσο το δυνατόν λιγότερες αποκλίσεις από την πραγματικότητα. Ορισμένες από τις πιο αξιόλογες και γνώστες μεθόδους που κατά τα χρόνια

δημιουργήθηκαν, αναπτύχθηκαν, διαδόθηκαν αλλά και έγιναν αποδεκτές, παρουσιάζονται στην αμέσως επόμενη υπό-ενότητα.

4.2.1 Βασικές κατηγορίες του Backpropagation

Υπάρχουν τέσσερις βασικές κατηγορίες στις οποίες χωρίζεται ο Backpropagation και είναι οι εξής:

1. Στατικός Backpropagation (Static Backpropagation)
2. Επαναλαμβανόμενος Backpropagation (Recurrent Backpropagation)
3. Ελαστικός Backpropagation (Resilient Backpropagation)
4. Backpropagation μέσω χρόνου (Backpropagation through time)

4.2.1.1 Στατικός Backpropagation (Static Backpropagation)

Ο **Στατικός Backpropagation** έχει ως στόχο τον υπολογισμό της συνάρτησης σφάλματος, ώστε να μπορέσει να προσαρμόσει σε πρώτη φάση τα αρχικά βάρη και στη συνέχεια τα αναπροσαρμοσμένα, με σκοπό να εξασφαλίσει το ελάχιστο δυνατό τετραγωνικό σφάλμα για την βελτιστοποίηση τόσο των βαρών όσο και την απόδοση και του δικτύου.

Κάποια από τα βασικά πλεονεκτήματά του είναι:

- Μπορεί να διαχειριστεί τις αλληλοσυσχετίσεις μεταξύ των εισόδων και των εξόδων, ακόμα και αν αυτές είναι πολύπλοκες και μη γραμμικές.
- Εφαρμόζεται σε προβλήματα που έχουν σχέση με την επεξεργασία της φυσικής γλώσσας, αναγνώρισης εικόνων και ομιλίας.
- Έχει την ικανότητα να εκπαιδεύσει ένα νευρωνικό δίκτυο για το οποίο η εκπαίδευση με επίβλεψη είναι το ζητούμενο, καθώς τα δεδομένα έχουν ετικέτες.

Τα βασικότερα μειονεκτήματα που το επηρεάζουν είναι:

- Σε μεγάλα και πολύπλοκα δίκτυα το κόστος από τη χρήση του να είναι ιδιαίτερα μεγάλο.
- Χρειάζεται ένα μεγάλο πλήθος από δεδομένα με ετικέτα για να μπορέσει να εκπαιδευτεί ικανοποιητικά.
- Η συνάρτηση του σφάλματος έχει την ιδιορρυθμία να παραμένει σε τοπικά ελάχιστα, μη έχοντας τη δυνατότητα να ξεφύγει από αυτά επειδή παραμένει σε ακατάλληλες τιμές.

Η εξίσωση που υλοποιεί τον Στατικό Backpropagation είναι η εξής:

$$\frac{\partial E}{\partial w} = \frac{\partial E}{\partial y} * \frac{\partial y}{\partial s} * \frac{\partial s}{\partial w} \quad (40)$$

όπου το E είναι η συνάρτηση σφάλματος, το w είναι ένα βάρος που εφαρμόζεται στο δίκτυο και s μία ενδιάμεση μεταβλητή.

4.2.1.2 Επαναλαμβανόμενος Backpropagation (Recurrent Backpropagation)

Ο **Επαναλαμβανόμενος Backpropagation** είναι μία από τις πιο δημοφιλείς τεχνικές εκπαίδευσης των Τεχνητών Νευρωνικών Δικτύων, ο οποίος υπολογίζει τις κλίσεις (gradients) της συνάρτησης απώλειας σε σχέση με τις παραμέτρους του, δικτύου και στην συνέχεια προβαίνει σε αλλαγές στις παραμέτρους με σκοπό την ελαχιστοποίηση της απώλειας αλλά και της βελτίωσης του δικτύου.

Τα δίκτυα τα οποία εκπαιδεύονται με τον Επαναλαμβανόμενο Backpropagation στο εσωτερικό τους διαθέτουν μία επαναλαμβανόμενη ανάδραση, η οποία επιτρέπει τη μεταφορά των δεδομένων από τη μία κατάσταση στην άλλη. Επειδή οι παράμετροι του δικτύου μοιράζονται ισομερώς στα χρονικά βήματα, οι κλίσεις είναι απαραίτητο να συνδυαστούν σωστά στο κάθε βήμα, για να γίνει ορθά η ενημέρωση των παραμέτρων του νευρωνικού δικτύου.

Μέσω αυτής της τεχνικής του Backpropagation μπορούμε να εξασφαλίσουμε κάποια οφέλη όπως είναι:

- Επίλυση ακολουθιακών προβλημάτων.
- Η δυνατότητα ανάκλησης προηγούμενων εισόδων.
- Η αξιοποίηση διαδοχικών δεδομένων σε πραγματικό χρόνο.

Αντίθετα η χρήση αυτή της τεχνικής μας περιορίζει στα καθώς:

- Λόγω του γεγονότος ότι επεξεργάζονται ακολουθίες διαφορετικών μηκών, χρειάζονται αρκετοί πόροι για να το καταφέρουν.
- Όταν μία ακολουθία έχει πολύ μεγάλο μήκος, μπορεί να μην καταφέρουν οι κλίσεις να εμφανιστούν ή ακόμα και να εκραγούν, δηλαδή όταν η κλίση συνεχίζει να μεγαλώνει, προκαλεί και μεγάλη ενημέρωση των βαρών, άρα το αποτέλεσμα είναι να αποκλίνει η μέθοδος από το επιθυμητό.

Η συνάρτηση που εφαρμόζεται στην τεχνική του Επαναλαμβανόμενου Backpropagation είναι:

$$\Delta w_{ij}(n) = -\eta \frac{\partial E(n)}{\partial w_{ij}(n)} - \lambda \frac{\partial E(n-1)}{\partial w_{ij}(n)} - \lambda^2 \frac{\partial E(n-2)}{\partial w_{ij}(n)} - \dots - \lambda^{n-1} \frac{\partial E(1)}{\partial w_{ij}(n)} \quad (41)$$

όπου $\Delta w_{ij}(n)$ είναι η ενημέρωση του βάρους για τον j -οστό νευρώνα στο χρονικό βήμα n , το η είναι ο ρυθμός εκμάθησης, το λ είναι ο συντελεστής απόσβεσης, $\frac{\partial E(t)}{\partial w_{ij}(n)}$ είναι η μερική παράγωγος του σφάλματος σε σχέση με το βάρος w_{ij} στο χρονικό βήμα t .

4.2.1.3 Ελαστικός Backpropagation (Resilient Backpropagation)

Ο **Ελαστικός Backpropagation** είναι ένας τύπος αλγορίθμου εκμάθησης, ο οποίος προσαρμόζει τα συναπτικά βάρη και τις πολώσεις δίνοντας σημασία μόνο στο πρόσημο της κλίσης. Η βασική της αρχή είναι η τροποποίηση του μεγέθους του βήματος με το οποίο ενημερώνονται τα συναπτικά βάρη με δυναμικό τρόπο, λαμβάνοντας υπόψιν τις ακριβώς προηγούμενες ενημερώσεις βαρών στην προηγούμενη επανάληψη.

Ο συγκεκριμένος αλγόριθμος έχει κατασκευαστεί ώστε να μπορεί να είναι μεθοδικός και αποτελεσματικός όταν στο σύνολο των δεδομένων υπάρχουν κάποια τα οποία είναι είτε περιττά είτε έχουν μολυνθεί με θόρυβο ή δεν είναι κατάλληλα.

Γι' αυτό το λόγο με τη χρήση του μπορούμε να έχουμε τα εξής πλεονεκτήματα:

- Δεν απαιτεί μεγάλη επεξεργασία για την εκτέλεσή της σε σχέση με άλλες τεχνικές.
- Επειδή δεν είναι επιρρεπής στις παραπάνω αντιξοότητες, μπορεί να συγκλίνει πολύ πιο γρήγορα σε σχέση με άλλες τεχνικές.

Αντίθετα, η χρήση της έχει και κάποιες μελανές στιγμές όπως:

- Μπορεί να χρειαστεί περισσότερη προσαρμογή στις υπέρ-παραμέτρους, σε σχέση με άλλες τεχνικές κατάβασης κλίσης.

- Δεν λαμβάνει υπόψιν τη δυναμική της κλίσης, με αποτέλεσμα να κολλήσει σε τοπικά ελάχιστα.

Η συνάρτηση η οποία υλοποιεί αυτή την κατηγορία του Backpropagation είναι:

$$\Delta w_{ij}(t) = - \text{sign}(g_{ij}(t) * \Delta w_{ij}(t - 1)) \quad (42)$$

όπου $g_{ij}(t)$ είναι η κλίση της συνάρτησης του σφάλματος λαμβάνοντας υπόψιν το βάρος w_{ij} τη χρονική στιγμή t , $\text{sign}()$ είναι η συνάρτηση του προσήμου και $\Delta w_{ij}(t)$ είναι η μεταβολή του βάρους τη χρονική στιγμή t .

4.2.1.4 Backpropagation μέσω χρόνου (Backpropagation through time)

Η κατηγορία του **Backpropagation μέσω χρόνου** χρησιμοποιείται σε επαναλαμβανόμενα νευρωνικά δίκτυα (RNNs), για τον υπολογισμό των κλίσεων σε κάθε βάρους στο δίκτυο. Σε αντίθεση με άλλα δίκτυα διατηρεί στη “μνήμη” τις προηγούμενες εισόδους στην ακολουθία, κάτι το οποίο είναι ιδιαίτερα χρήσιμο για δεδομένα χρονοσειρών.

Έχουμε το πλεονέκτημα ότι μπορούν να χρησιμοποιηθούν σε πολλά frameworks βαθιάς μάθησης, όπως το TensorFlow και το PyTorch. Αντίθετα, λόγω της αναδρομής στον χρόνο χρειάζονται πολλούς υπολογιστικούς πόρους και μεγάλη ισχύ από τη CPU [18].

Ο Backpropagation μέσω χρόνου βασίζεται στην παρακάτω εξίσωση:

$$\frac{\partial h(n)}{\partial w_h(n)} = \frac{\partial f(x_t, h_{t-1}, w_h)}{\partial w_h} + \frac{\partial f(x_t, h_{t-1}, w_h)}{\partial w_{h-1}} \cdot \frac{\partial h_{t-1}}{\partial w_h} \quad (43)$$

Από τις παραπάνω τέσσερις κατηγορίες τύπων του Backpropagation, ο Στοχαστικός και ο Ελαστικός Backpropagation είναι οι πιο βασικοί τύποι, με τους άλλους δύο Επαναληπτικός και Μέσω Χρόνου να είναι δύο τύποι πιο εναλλακτικοί, που μπορούν να προκύψουν και με ευρύτερες παραλλαγές των πρώτων δύο. Κάποιες από τις παραλλαγές θα τις αναφέρουμε επιγραμματικά στην επόμενη ενότητα.

4.3 Παραλλαγές του Backpropagation

Ο αλγόριθμος του Backpropagation έχει παραλλαχθεί με διαφορετικές τεχνικές, ώστε να μπορέσει να προσαρμοστεί στις ανάγκες του κάθε προβλήματος. Είναι γεγονός ότι με μία παραλλαγή του Backpropagation δεν υπάρχει εγγύηση ότι μπορεί να φέρει τα κατάλληλα αποτελέσματα σε οποιοδήποτε πρόβλημα. Χαρακτηριστικό είναι ότι σε κάποιο πρόβλημα αν ο όγκος των δεδομένων που έχουμε να επεξεργαστούμε είναι πολύ μεγάλος, τότε μπορεί μία τεχνική που είναι πολύ ακριβής με εξαιρετικά ποσοστά επιτυχίας στις προβλέψεις, να μην προτιμηθεί σε σχέση με μία δεύτερη η οποία μπορεί να στερείται ένα μικρό ποσοστό επιτυχίας σε σχέση με την πρώτη, αλλά είναι αρκετές φορές πιο γρήγορη ή να καταναλώνει λιγότερους επεξεργαστικούς πόρους σε σχέση με την πρώτη.

Έτσι κατά καιρούς έχουν δημιουργηθεί πολλές παραλλαγές οι οποίες είναι αξιόλογες, αφήνοντας στην κρίση του κάθε χρήστη να διαλέξει ποια ή ποιες είναι καταλληλότερες για το πρόβλημα που έχει να διαχειριστεί. Στην δική μας περίπτωση για τα προβλήματα που επιλέξαμε να κάνουμε τη σύγκριση και την αποτίμηση με τον ELM, επιλέξαμε τη μέθοδο του Levenberg – Marquardt (LM), τον Resilient Backpropagation (Rprop) και τη μέθοδο Scaled Conjugate Descent (SCG) με την τελευταία να λαμβάνει συμμετοχή στα προβλήματα της Ταξινόμησης.

Ορισμένες από τις παραλλαγές [19] που προκύπτουν από τις παραπάνω κατηγορίες που Backpropagation, είναι οι ακόλουθες:

- Πέρα από τη Στοχαστική μέθοδο μέσω του Gradient Descent, υπάρχει και η **Mini-Batch Gradient Descent**, που θεωρείται μία μέθοδος που διασταυρώνει την Stochastic Gradient Descent με την Full-Batch Gradient Descent, η υποδιαιρεί το γενικό σύνολο των δεδομένων εκπαίδευσης σε μικρότερα σύνολα με σκοπό τον πιο άμεσο και γρήγορο υπολογισμό των διαβαθμίσεων.
- Η **Gradient Descent with Momentum** αποτελεί μία ακόμη παραλλαγή του Backpropagation, η οποία εντάσσει ένα κλάσμα από τον υπολογισμό την ενημέρωσης του προηγούμενου βάρους, στην τρέχουσα ενημέρωση βάρους, έτσι ώστε ο αλγόριθμος να μπορεί να αποκτήσει ένα momentum (δυναμική – ορμή) κατά την κατάβαση κλίσης για τον υπολογισμό της συνάρτησης κόστους. Αυτό βοηθάει στην επιτάχυνση της κλίσης του αλγορίθμου, στην μείωση της εμφάνισης τοπικών ελαχίστων, στη μείωση του overfitting κ.τ.λ.
- Η παραλλαγή γνωστή ως **Weight Regularization** είναι μία τεχνική που εφαρμόζεται από τις μεθόδους L1 και L2, όπου έχει ως γενική στόχευση τις μικρές τροποποιήσεις στα δεδομένα των πειραμάτων ή σε μη ορατά δεδομένα, κάτι το οποίο προσφέρει καλύτερη γενίκευση στη λύση του προβλήματος. Βοηθάει στην βελτιστοποίηση της κακής προσαρμογή των μοντέλων αλλά και στη γενικότερη απόδοση του δικτύου. Η L1 τεχνική βασίζεται στον υπολογισμό του αθροίσματος των τετραγώνων των βαρών για την ενημέρωση των νέων βαρών, επιδιώκοντας τα βάρη να έχουν τιμές της μορφής 0,0. Αντίθετα η L2 βασίζεται στον υπολογισμό των απόλυτων τιμών των βαρών για την ενημέρωση των επιμέρους βαρών, με γενική στόχευση την μείωση των βαρών που η τιμή τους είναι πολύ μεγάλη συγκριτικά με τις τιμές των υπόλοιπων βαρών. Οι παραπάνω μέθοδοι L1 και L2 μπορούν να συνδυαστούν και με την **Bayesian Regularization** μέθοδο, βασιζόμενες στον υπολογισμό του β διανύσματος, που αφορά την ελαχιστοποίηση του τετραγωνικού σφάλματος των τιμών των παρατηρήσεων.
- Μία ακόμη παραλλαγή είναι η **Adaptive Learning Rates**, μέσω της οποίας κατά τη διάρκεια εκπαίδευσης του νευρωνικού δικτύου προσαρμόζεται ο ρυθμός εκμάθησης για κάθε παράμετρο. Ο τρόπος με τον οποίον προσαρμόζουν τον ρυθμό εκμάθησης γίνεται εφόσον πρώτα ληφθούν υπόψιν οι προηγούμενες κλίσεις του δικτύου και αναλόγως γίνονται οι σχετικές ενημερώσεις. Ο Adagrad, ο Adadelat, ο RMSprop και ο Adam είναι κάποιοι από τους αλγορίθμους που διεκπεραιώνουν την εκπαίδευση των νευρωνικών δικτύων με δυναμική αναπροσαρμογή του ρυθμού εκμάθησης. Η παραλλαγή αυτή βοηθάει στις περιπτώσεις που υπάρχει θόρυβος στα δεδομένα εκμάθησης και δοκιμών ώστε να υπάρξει καλύτερη σύγκλιση στα αποτελέσματα και πιο γρήγορα.

Κάποιες επιπλέον γνωστές παραλλαγές είναι οι Hessian – Free Optimization, η Long Short – Term Memory (LSTM), τα Residual Networks (ResNets), η Nesterov Accelerated Gradient (NAG) κ.τ.λ.

Όλες οι παραλλαγές του Backpropagation έχουν ως γενικότερο σκοπό την καλύτερη προσέγγιση των συγκλίσεων των δικτύων, μέσω της κατανόησης των προτύπων που δημιουργούνται από τα δεδομένα. Προσπαθούν να βελτιστοποιήσουν τα σφάλματα από τις προσεγγίσεις των δεδομένων για να γίνουν πιο αποτελεσματικά και να προσφέρουν ένα καλύτερο ποσοστό επιτυχίας.

4.4 Levenberg – Marquardt Backpropagation

Ο **Kenneth Levenberg (1919 – 1973)** και ο **Donald Marquardt (1929 – 1997)** υπήρξαν Αμερικανοί στατιστικολόγοι, όπου έγιναν ευρέως γνωστοί από τη μέθοδο ταιριάσματος μη-γραμμικών ελαχίστων τετραγώνων του ομώνυμου αλγορίθμου που χρησιμοποιείται ως και σήμερα εκτενώς, του Levenberg – Marquardt για τα τεχνητά νευρωνικά δίκτυα.

Ο αλγόριθμος του **Levenberg – Marquardt (LM)** [20] είναι ένας από τους 3 αλγόριθμους που χρησιμοποιήσαμε στα πειράματα με τα προβλήματα συναρτησιακής παλινδρόμησης και ταξινόμησης όπως θα δούμε στο **Κεφάλαιο 7**. Πρόκειται για τον ίσως πιο αποτελεσματικό αλγόριθμο του Backpropagation, ο οποίος συνδυάζει τα πλεονεκτήματα της μεθόδου **Gradient Descent** και της μεθόδου **Gauss – Newton**. Τα βήματα που ακολουθεί είναι ένας γραμμικός συνδυασμός των δύο προαναφερθεισών μεθόδων βασιζόμενα σε προσαρμοστικούς κανόνες. Αρχικώς κυριαρχούν τα βήματα της Gradient Descent μεθόδου έως ότου να χρειαστεί να ακολουθήσουν τα βήματα της Gauss – Newton.

Στην εκπαίδευση μέσω του MATLAB [21] για νευρωνικά δίκτυα μέσω του LM, για τους υπολογισμούς χρησιμοποιείται ο Jacobian πίνακας, όπου η απόδοση εκρέει από τα ελάχιστα ή το άθροισμα των σφαλμάτων των τετραγώνων. Επομένως, τα δίκτυα εκπαιδεύονται με αυτή την τεχνική και για την απόδοση λαμβάνονται υπόψιν είτε το Ελάχιστο Τετραγωνικό Σφάλμα (MSE) είτε το Σφάλμα Αθροίσματος Τετραγωνικού (SSE).

Ο πίνακας Jacobian [22] είναι ένας πίνακας που συνθέτει της πρώτης τάξης επιμέρους παραγώγους μίας συνάρτησης με πολλαπλές μεταβλητές.

Για μία συνάρτηση $f: \mathbb{R}^3 \rightarrow \mathbb{R}$, η παράγωγος του J για το διάνυσμα μίας γραμμής ορίζεται ως:

$$\left(\frac{\partial(f)}{\partial x_1}(J), \frac{\partial(f)}{\partial x_2}(J), \dots, \frac{\partial(f)}{\partial x_n}(J) \right) \quad (44)$$

Συνεπώς, ο πίνακας Jacobian γράφεται ως:

$$\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_m} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_m} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \dots & \frac{\partial f_3}{\partial x_m} \end{bmatrix}$$

Πίνακας 4 Jacobian

Ο αλγόριθμος ξεκινάει με μία αρχική πρόβλεψη για το \mathbf{x}_0 . Το \mathbf{x} μπορεί να προσαρμοστεί από το δ μόνο για τα βήματα κατάβασης. Η εξίσωση που διαμορφώνεται είναι η ακόλουθη:

$$(J^T J + \lambda I)\delta = J^T r \quad (45)$$

όπου το J είναι ο πίνακας Jacobian από τις παραγώγους των υπολοίπων σε σχέση με τις παραμέτρους, το λ είναι η παράμετρος απόσβεσης και r το διάνυσμα των υπολοίπων.

4.5 Resilient Backpropagation

Η δεύτερη υλοποίηση Backpropagation αλγορίθμου που χρησιμοποιήσαμε στο πειραματικό κομμάτι είναι αυτή του Resilient Backpropagation. Όπως προαναφέραμε και στην *υπό-ενότητα 4.2.1.3* αυτό ο τύπος αλγορίθμου δίνει βάση στην αλλαγή του προσήμου της κλίσεις, για να προσαρμόσει ανάλογα τις νέες τιμές στα συναπτικά βάρη και τις πολώσεις. Μολονότι δίνεται βάση στο πρόσημο της παραγώγου, το μέγεθος της παραγώγου δεν επηρεάζει τον αλγόριθμο στην ενημέρωση των παραμέτρων. Αν η παράγωγος μηδενιστεί τότε δεν αλλάζει η τιμή της εκάστοτε ενημέρωσης. Σε περίπτωση που αλλάξει η παράγωγος πρόσημο σε σχέση με αυτό που έχει το βάρος, τότε ακολουθείται μία συγκεκριμένη αύξηση ή μείωση (αντίστοιχη λογική με τον ρυθμό εκμάθησης), ίδιου προσήμου με αυτό που έχει η παράγωγος. Επιπρόσθετα, αν για ένα βάρος η παράγωγος παραμείνει ίδια για δύο συνεχόμενες φορές, αυξάνεται το βήμα ενημέρωσης. Με αυτόν τον τρόπο τα διαφορετικά βάρη θα χρειαστούν διαφορετικού μεγέθους βήματα κατά τη διάρκεια εκπαίδευσης για να καταφέρουν να προσαρμοστούν σωστά [23] [24].

4.5.1 Μαθηματική προσέγγιση του Resilient Backpropagation

Για να εξηγήσουμε μαθηματικά την παραπάνω διαδικασία έχουμε τις εξής συνθήκες:

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^+ \cdot \Delta_{ij}^{(t-1)}, \text{ αν } \frac{\partial E^{(t-1)}}{\partial w_{ij}} \cdot \frac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ \eta^- \cdot \Delta_{ij}^{(t-1)}, \text{ αν } \frac{\partial E^{(t-1)}}{\partial w_{ij}} \cdot \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ \Delta_{ij}^{(t-1)}, \text{ αλλιώς} \end{cases} \quad (46)$$

όπου $0 < \eta^- < 1 < \eta^+$, w_{ij} ένα συγκεκριμένο βάρος και Δ_{ij} η μεταβολή του βάρους.

Οι προτεινόμενες τιμές που μπορούν να έχουν οι παράμετροι είναι:

- $\Delta_{ij}^{(0)} = 0.1$
- $\Delta_{min} = 10^{-6}$
- $\Delta_{max} = 50$
- $\eta^- = 0.5$
- $\eta^+ = 1.2$

Οι αναπροσαρμογές των βαρών πραγματοποιούνται σύμφωνα με τις ακόλουθες συνθήκες:

$$\Delta w_{ij}^{(t)} = \begin{cases} -\Delta_{ij}^{(t)}, \text{ αν } \frac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ +\Delta_{ij}^{(t)}, \text{ αν } \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ 0, \text{ αλλιώς} \end{cases} \quad (47)$$

όπου ο υπολογισμός των σφαλμάτων από τις κλίσεις υπολογίζεται μέσω όλων των δεδομένων εκπαίδευσης (μάθηση μέσω εποχών), έχοντας ως πλεονέκτημα ότι το μέγεθος του βήματος υπολογίζεται δυναμικά.

4.5.2 Προσέγγιση με Ψευδοκώδικα του Resilient Backpropagation

$$\forall i, j : \Delta_{ij}(t) = \Delta_0$$

$$\forall i, j : \frac{\partial E}{\partial w_{ij}}(t-1) = \Delta_0$$

Repeat

Compute Gradient $\frac{\partial E}{\partial w_{ij}}(t)$

For all weights and biases {

if $\left(\frac{\partial E}{\partial w_{ij}}(t-1) \cdot \frac{\partial E}{\partial w_{ij}}(t) > 0\right)$ then {

$$\Delta_{ij}(t) = \text{minimum}(\Delta_{ij}(t-1) \cdot \eta^+, \Delta_{max})$$

$$\Delta w_{ij}(t) = -\text{sign}\left(\frac{\partial E}{\partial w_{ij}}(t)\right) \cdot \Delta_{ij}(t)$$

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t)$$

$$\frac{\partial E}{\partial w_{ij}}(t-1) = \frac{\partial E}{\partial w_{ij}}(t)$$

}

else if $\left(\frac{\partial E}{\partial w_{ij}}(t-1) \cdot \frac{\partial E}{\partial w_{ij}}(t) < 0\right)$ then {

$$\Delta_{ij}(t) = \text{maximum}(\Delta_{ij}(t-1) \cdot \eta^-, \Delta_{min})$$

$$\frac{\partial E}{\partial w_{ij}}(t-1) = 0$$

}

else if $\left(\frac{\partial E}{\partial w_{ij}}(t-1) \cdot \frac{\partial E}{\partial w_{ij}}(t) = 0\right)$ then {

$$\Delta w_{ij}(t) = -\text{sign}\left(\frac{\partial E}{\partial w_{ij}}(t)\right) \cdot \Delta_{ij}(t)$$

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t)$$

$$\frac{\partial E}{\partial w_{ij}}(t-1) = \frac{\partial E}{\partial w_{ij}}(t)$$

}

}

Until (converged)

$$\text{sign} = \begin{cases} \mathbf{1}, & x > 0 \\ \mathbf{0}, & x = 0 \\ -\mathbf{1}, & x < 0 \end{cases}$$

4.6 Scaled Conjugate Gradient

Ο **Scaled Conjugate Gradient (SCG)** είναι μία από τις πολλές παραλλαγές του Backpropagation και πρόκειται για ένα αλγόριθμο ο οποίος αναπτύχθηκε από το **Martin Fodslette Møller** το 1991. Η αφορμή για τον Møller ν' αναπτύξει τον συγκεκριμένο αλγόριθμο βασίζεται σε βασικά προβλήματα που είχαν οι κλασσικοί αλγόριθμοι μέσω gradient descent του Backpropagation. Ένα πρόβλημα που υπάρχει είναι η ανάγκη για τον καθαρισμό κρίσιμων παραμέτρων από τον χρήστη οι οποίες μάλιστα μπορούν να εξασφαλίσουν την επιτυχία ή μη του αλγορίθμου. Από αυτές τις παραμέτρους προκύπτει και το πρόβλημα τη αδυναμίας σύγκλισης ενός αλγορίθμου.

Ένα άλλο πρόβλημα είναι ότι οι κοινοί έως τότε αλγόριθμοι του Backpropagation μέσω gradient descent ήταν προβληματικοί με δεδομένα μεγάλης κλίμακας όπως είναι ο κλασσικός Backpropagation των Rumelhart, Hinton και Williams που δημιουργήθηκε το 1986. Το κύριο σημείο που οδηγεί στην αδυναμία του είναι ότι εξαρτάται από παραμέτρους που πρέπει να δοθούν από τον χρήστη χωρίς να έχει γνώση αυτός του θεωρητικού υποβάθρου για την επιλογή αυτών.

Σκοπός αυτού του αλγορίθμου είναι αν εξασφαλίσει τη βέλτιστη συνάρτηση που θα ελαχιστοποιεί το σφάλμα των παραμέτρων, χωρίς να υποκύπτει στα προαναφερθέντα προβλήματα. Έτσι βασιζόμενος ο Møller στην άποψη κάποιων ειδικών της κοινότητας που ασχολούνται με την αριθμητική ανάλυση, η οποία ισχυρίζονταν πως οι μέθοδοι που αποκαλούνται “*Μέθοδοι Συζευγμένης Κλίσης*”, μπορούν να εφαρμοστούν σε προβλήματα μεγάλης κλίμακας και να επιλύσουν. Με αυτή τη λογική ως βάση αναπτύχθηκε ο συγκεκριμένος αλγόριθμος μάθησης με επίβλεψη, καθώς και στο γεγονός της πολυπλοκότητας κάποιων παραλλαγών που βασίζονται στην αναζήτηση γραμμής σε κάθε επαναληπτική διαδικασία κατά την εκμάθηση [25] [26].

4.6.1 Μαθηματική προσέγγιση του Scaled Conjugate Gradient

Στον συγκεκριμένο αλγόριθμο το μέγεθος του βήματος για την αναπροσαρμογή των παραμέτρων είναι μία συνάρτηση τετραγωνικής προσέγγισης συναρτήσει της συνάρτησης σφάλματος, η οποία καθίσταται πιο ισχυρή και ανεξάρτητη από τις παραμέτρους που ορίζουν οι χρήστες σε άλλους αλγορίθμους. Το μέγεθος του βήματος υπολογίζεται χρησιμοποιώντας διαφορετική προσέγγιση. Ο όρος δεύτερης τάξης υπολογίζεται από την ακόλουθη συνάρτηση:

$$\bar{\sigma}_k = \frac{E' \cdot (\bar{w}_k + \sigma_k \cdot \bar{p}_k) - E'(\bar{w}_k)}{\sigma_k} + \lambda_k \cdot \bar{p}_k \quad (48)$$

όπου το λ_k είναι βαθμωτό και προσαρμόζεται κάθε φορά από το πρόσημο του δ_k .

Το μέγεθος του βήματος υπολογίζεται από την ακόλουθη εξίσωση:

$$\alpha_k = \frac{\mu_k}{\delta_k} = \frac{-p_j^{-T} \cdot E'_{q_w}(\bar{y}_1)}{p_j^{-T} \cdot E'' \cdot (\bar{w}) \cdot \bar{p}_j} \quad (49)$$

όπου το $\bar{w} \rightarrow$ είναι το διάνυσμα βαρών στο \mathbb{R}^n ,

$E(\bar{w}) \rightarrow$ είναι το καθολικό σφάλμα της συνάρτησης,

$E'(\bar{w}) \rightarrow$ είναι η κλίση από το σφάλμα,

$E'_{q_w}(\bar{y}_1) \rightarrow$ είναι η τετραγωνική προσέγγιση από τη συνάρτηση σφάλματος,

$\bar{p}_1, \bar{p}_2, \dots, \bar{p}_n \rightarrow$ είναι το διάνυσμα των μη-μηδενικών βαρών,

$\lambda_k \rightarrow$ πρέπει να ενημερωθεί έτσι ώστε,

$$\bar{\lambda}_k = 2 \cdot \left(\lambda_k - \frac{\delta_k}{|\bar{p}_k|^2} \right) \quad (50)$$

Αν $\Delta_k > 0.75$, τότε $\lambda_k = \frac{\lambda_k}{4}$,

Αν $\Delta_k < 0.25$, τότε $\lambda_k = \lambda_k + \frac{\delta_k \cdot (1 - \Delta_k)}{|\bar{p}_k|^2}$

όπου Δ_k είναι η παράμετρος σύγκρισης και δίνεται από τον τύπο:

$$\Delta_k = \frac{2 \cdot \delta_k \cdot [E(\bar{w}_k) - E(\bar{w}_k + \alpha_k \cdot \bar{p}_k)]}{\mu_k^2} \quad (51)$$

Αρχικά οι τιμές έχουν αρχικοποιηθεί ως εξής:

$$0 < \sigma \leq 10^{-4}, 0 < \lambda_i \leq 10^{-6} \text{ και } \bar{\lambda}_i = 0$$

Η διαδικασία της εκπαίδευσης θα σταματήσει όταν έστω οποιαδήποτε από τις παρακάτω συνθήκες εμφανιστεί:

- Όταν η απόδοση της κλίσης πέσει χαμηλότερα από το mid-grad (είναι η ελάχιστη αποδεκτή τιμή που μπορεί να έχει η κλίση – χαμηλό κατώφλι κλίσης).
- Η απόδοση μικραίνει βάση του στόχου.
- Φτάσει ο αλγόριθμος στον μέγιστο αριθμό των εποχών που έχει οριστεί εξ' αρχής.
- Υπερβεί ο αλγόριθμος τον μέγιστο χρόνο που έχει τεθεί εξ' αρχής.

Κεφάλαιο 5

Extreme Learning Machine

Ο **Extreme Learning Machine (ELM)**, και στην ελληνική εκδοχή του **Μηχανή Ακραίας Μάθησης**, είναι ένας αλγόριθμος που απευθύνεται σε μονοστρωματικά νευρωνικά δίκτυα εμπρόσθιας τροφοδοσίας (**Single Layer Feedforward Neural Networks - SLFNs**) ο οποίος αναπτύχθηκε από τους **Guang-Bin Huang, Qin-Yu Zhu** και **Chee-Kheong Siew** το **2006** στη σχετική εισήγηση που έκαναν με τίτλο “**Extreme Learning Machine: Theory and applications**” [27] [28].

Στην προτεινόμενη εισήγηση την οποία έκαναν ο Huang, ο Zhu και ο Siew παρουσίασαν τη δομή, το μαθηματικό υπόβαθρο αλλά και πειράματα ενός αλγορίθμου για SLNFs, ο οποίος ανήκει στην κατηγορία των αλγορίθμων με επίβλεψη, και είχε ως στόχο τη αντιμετώπιση των προβλημάτων των SLFNs όταν εκπαιδεύονταν με τους έως τότε γνωστούς από την κοινότητα νευρωνικούς αλγόριθμους. Τα βασικότερα προβλήματα που αντιμετωπίζουν οι παραδοσιακοί αλγόριθμοι είναι α) χρησιμοποιούνται εκτενώς οι αλγόριθμοι εκμάθησης που είναι βασισμένη στην κλίση, παρά το γεγονός ότι είναι αργοί στην εκτέλεσή τους, με τις περισσότερες φορές περισσότερο του αναμενομένου, και β) ότι όλες οι παράμετροι που αφορούν το δίκτυο πρέπει να προσαρμοστούν επαναληπτικά για να μπορέσει το δίκτυο να εκπαιδευτεί.

Οι τρεις προαναφερόμενοι εφευρέτες με βασικό εισηγητή τον **Huang**, θέλησαν να δημιουργήσουν και αν προτείνουν τον ELM ως έναν αλγόριθμο που θα λύσει τις προηγούμενες παθογένειες των παραδοσιακών αλγορίθμων εκμάθησης των νευρωνικών δικτύων σε πολύ πιο γρήγορο χρόνο, αλλά ταυτόχρονα πετυχαίνοντας και καλύτερη γενίκευση των αποτελεσμάτων σε σχέση με τους παραδοσιακούς.

5.1 Ανάλυση και μαθηματική προσέγγιση του ELM

Πολλοί ερευνητές έχουν εξερευνήσει τη καθολική προσέγγιση των δυνατοτήτων των κλασικών πολυστρωματικών νευρωνικών δικτύων εμπρόσθιας τροφοδοσίας. Ο Hornik απέδειξε ότι αν υπάρχει μία συνάρτηση ενεργοποίησης συνεχής, φραγμένη και μη σταθερή, τότε οι συνεχείς συσχετίσεις μπορούν να προσδιοριστούν μέσω ενός συμπαγούς και πεπερασμένου συνόλου δεδομένων εισόδου από ένα νευρωνικό δίκτυο.

Επιπρόσθετα ο Leshno, επέκτεινε αυτή τη θεωρία του Hornik και απέδειξε ότι τα εμπροστοτροφοδοτούμενα νευρωνικά δίκτυα με μία μη-πολυωνυμική συνάρτηση ενεργοποίησης, μπορούν να προσεγγίσουν συνεχείς συναρτήσεις.

Γνωρίζοντας τα παραπάνω, ο Huang με τον Babri απέδειξαν ότι σε ένα SLNF με N κρυφούς κόμβους, μπορεί να μάθει N διακριτές παρατηρήσεις σχεδόν με οποιαδήποτε συνάρτηση ενεργοποίησης που είναι μη-γραμμική. Για να μπορέσει να πραγματοποιηθεί ο παραπάνω ισχυρισμός, είναι αναγκαίο να προσαρμοστούν τα βάρη και το στρώμα των πολώσεων, όπως ακριβώς γίνεται με όλους τους αλγορίθμους εκμάθησης των νευρωνικών δικτύων.

5.1.1 Θεωρητικό υπόβαθρο βαρών, πολώσεων και συναρτήσεων ενεργοποίησης

Για αρκετές δεκαετίες η κύρια μέθοδος των αλγορίθμων των νευρωνικών δικτύων υπήρξε η μέθοδος gradient descent (καθοδικής κλίσης), η οποία στα πειράματα που εφαρμοζόταν μέσω των νευρωνικών αλγορίθμων εκμάθησης είχε ως αποτέλεσμα την αργοπορία στην παρουσίαση των αποτελεσμάτων της. Πέραν τούτου οι κοινές μέθοδοι είναι επιρρεπείς στο να συγκλίνουν σε τοπικά ελάχιστα.

Ο Huang με τους συνεργάτες του απέδειξαν ότι μέσω του ELM τον οποίο εφάρμοσαν τόσο σε τεχνητές αλλά και σε μεγάλες και πραγματικές εφαρμογές, πως αν στο κρυφό στρώμα του δικτύου υπάρχει μία συνάρτηση ενεργοποίησης απείρως παραγωγίσιμη (ή διαφορίσιμη), τότε μπορούν οι αρχικές αναθέσεις των διανυσμάτων του βάρους και των πολώσεων να αποδοθούν τυχαία και με τυχαίες τιμές στις παραμέτρους εισόδου. Βασιζόμενοι στο θεώρημα του Barlett για τα SLFN, το οποίο αναφέρει πως όσο μικρότερη είναι η νόρμα του σφάλματος των βαρών, τόσο καλύτερη γενικευμένη απόδοση μπορεί να έχει το δίκτυο, ο Huang και οι συνεργάτες του κατάφεραν να εντάξουν στον ELM την ικανότητα να φτάνει σε πολύ χαμηλά σφάλματα βαρών με μεγάλη ταχύτητα και ακρίβεια σε αντίθεση με τους ως τότε πασίγνωστους αλγορίθμους εκμάθησης των SLNFs.

5.1.2 Μαθηματική μοντελοποίηση της δομής του ELM

Για N τυχαίες και διακριτά δείγματα $\mathbf{x}_i, \mathbf{t}_i$, όπου $\mathbf{x}_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in \mathbb{R}^n$ και $\mathbf{t}_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in \mathbb{R}^m$ στα κανονικά SLFNs με \tilde{N} κρυφούς κόμβους και συνάρτηση ενεργοποίησης την $g(x)$ έχουμε:

$$\sum_{i=1}^{\tilde{N}} \beta_i g_i(x_j) = \sum_{i=1}^{\tilde{N}} \beta_i g(w_i \cdot x_j + b_i) = \mathbf{o}_j, j = 1, 2, \dots, N \quad (52)$$

όπου $\mathbf{w}_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ είναι το διάνυσμα των βαρών που συνδέουν τον i -οστό κρυφό κόμβο και τους κόμβους εισόδου, $\boldsymbol{\beta}_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ είναι το διάνυσμα βαρών που συνδέει τον i -οστό κρυφό κόμβο με τους κόμβους εξόδου και b_i το διάνυσμα των πολώσεων του i -οστού.

Αυτά τα κανονικά SLFNs με \tilde{N} κρυφούς κόμβους και συνάρτηση ενεργοποίησης την $g(x)$ μπορεί να προσεγγίσει αυτά τα N δείγματα με μηδενικό σφάλμα που σημαίνει ότι $\sum_{j=1}^{\tilde{N}} \|\mathbf{o}_j - \mathbf{t}_j\| = 0$ και για τα οποία υπάρχουν β_i, w_i και b_i τέτοια ώστε:

$$\sum_{i=1}^{\tilde{N}} \beta_i g(w_i \cdot x_j + b_i) = \mathbf{t}_j, j = 1, 2, \dots, N \quad (53)$$

Οι παραπάνω N συναρτήσεις μπορούν να γραφτούν ως εξής:

$$\mathbf{H} \cdot \boldsymbol{\beta} = \mathbf{T} \quad (54)$$

για τη οποία συνάρτηση ισχύουν και συνεπάγονται τα:

$$\mathbb{H}(w_1, w_2, \dots, w_{\tilde{N}}, b_1, b_2, \dots, b_{\tilde{N}}, x_1, x_2, \dots, x_{\tilde{N}}) = \begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \cdots & g(w_{\tilde{N}} \cdot x_1 + b_{\tilde{N}}) \\ \vdots & \cdots & \vdots \\ g(w_1 \cdot x_N + b_1) & \cdots & g(w_{\tilde{N}} \cdot x_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}} \quad (55)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{N \times \tilde{N}} \quad \text{και} \quad T = \begin{bmatrix} t_1^T \\ \vdots \\ t_{\tilde{N}}^T \end{bmatrix}_{N \times \tilde{N}} \quad (56)$$

Ο πίνακας \mathbb{H} αντιπροσωπεύει το κρυφό στρώμα του πίνακα εξόδου του νευρωνικού δικτύου, η i -οστή στήλη του \mathbb{H} είναι ο i -οστός κρυφός κόμβος εξόδου σε σχέση με τις εισόδους x_1, x_2, \dots, x_N .

Με τις προηγούμενες εξισώσεις (52 – 56) μπορεί να υποστηριχθεί ένα νευρωνικό δίκτυο το οποίο αποτελείται από ένα κρυφό στρώμα, εμπρόσθιας τροφοδοσίας με τυχαίους κρυφούς. Ο Huang με τους συνεργάτες του πάνω σε αυτή τη βάση και με τα αμέσως επόμενα δύο θεωρήματα που ανέπτυξαν και θα αναφέρουμε μπόρεσαν να συνθέσουν τη μαθηματική μοντελοποίηση του ELM. [27]

Θεώρημα 1

«Έστω ότι ένα κανονικό SLFN με N τυχαίους κόμβους και συνάρτηση ενεργοποίησης την $g: \mathbb{R} \rightarrow \mathbb{R}$ η οποία είναι απείρως παραγωγίσιμη σε κάθε διάστημα, για N τυχαία και διακριτά δείγματα (x_i, t_i) , όπου $x_i \in \mathbb{R}^n$ και $t_i \in \mathbb{R}^m$, για κάθε w_i και b_i τα οποία έχουν επιλεγθεί τυχαία στα διαστήματα \mathbb{R}^n και \mathbb{R} , αντίστοιχα, σύμφωνα με οποιαδήποτε συνεχή κατανομή πιθανοτήτων, ο πίνακας εξόδου του κρυφού στρώματος \mathbb{H} του SLFN είναι αναστρέψιμος και ισχύει ότι: $\|\mathbb{H} \cdot \beta - T\| = 0$. » [27]

Θεώρημα 2

«Δίνεται μία μικρή πιθανότητα με τιμή $\varepsilon > 0$ και συνάρτηση ενεργοποίησης τη $g: \mathbb{R} \rightarrow \mathbb{R}$ η οποία είναι απείρως παραγωγίσιμη σε κάθε διάστημα, για την οποία υπάρχει $\tilde{N} \leq N$ τέτοιο ώστε για τα N διακριτά δείγματα (x_i, t_i) , όπου $x_i \in \mathbb{R}^n$ και $t_i \in \mathbb{R}^m$, για κάθε w_i και b_i τα οποία έχουν επιλεγθεί τυχαία στα διαστήματα \mathbb{R}^n και \mathbb{R} , αντίστοιχα, σύμφωνα με οποιαδήποτε συνεχή κατανομή πιθανοτήτων, τότε με πιθανότητα ίση με 1 ισχύει ότι: $\|\mathbb{H}_{N \times \tilde{N}} \cdot \beta_{\tilde{N} \times m} - T_{\tilde{N} \times m}\| < \varepsilon$. » [27]

Γι' αυτά τα δύο θεωρήματα ο Huang στην εισήγησή του αναφέρει ρητά και την απόδειξή τους, κάτι το οποίο παραλείπουμε εμείς.

Με τις προηγούμενες εξισώσεις και τα θεώρημα 1 και θεώρημα 2 η συμβατική λύση των SLFNs μέσω της καθοδικής κλήσης γίνεται ως εξής:

$$\begin{aligned} & \mathbb{H}(\widehat{w}_1, \widehat{w}_2, \dots, \widehat{w}_{\tilde{N}}, \widehat{b}_1, \widehat{b}_2, \dots, \widehat{b}_{\tilde{N}}) \hat{\beta} - T \\ & = \min_{w_i, b_i, \beta} \|\mathbb{H}(\widehat{w}_1, \widehat{w}_2, \dots, \widehat{w}_{\tilde{N}}, \widehat{b}_1, \widehat{b}_2, \dots, \widehat{b}_{\tilde{N}}) \hat{\beta} - T\| \end{aligned} \quad (57)$$

Η παραπάνω εξίσωση μπορεί να γραφεί ως:

$$E = \sum_{j=1}^N \left(\sum_i^{\tilde{N}} \beta_i \cdot g(w_i \cdot x_j + b_i) - t_j \right)^2 \quad (58)$$

Για να μπορέσει να γίνει αποδοτικός ο αλγόριθμος απαραίτητη προϋπόθεση είναι να βρεθεί η μικρότερη δυνατή νόρμα των ελαχίστων τετραγώνων των SLFNs, για την οποία ο Huang απέδειξε πως μπορεί να πραγματοποιηθεί από την παρακάτω εξίσωση:

$$\hat{\beta} = \mathbb{H}^+ \mathbf{T} \quad (59)$$

όπου το \mathbb{H}^+ είναι ο γενικευμένος ανάστροφος του Moore – Penrose του πίνακα \mathbb{H} .

5.1.3 Περιγραφική διατύπωση των βημάτων του αλγορίθμου ELM

Έχοντας ως παρακαταθήκη τη μαθηματική μοντελοποίηση της δομής του ELM που ανέπτυξε ο Huang με τους συνεργάτες του, παρακάτω συγκεντρώνονται λεκτικά τα βήματα που ακολουθεί ο αλγόριθμος για την εκπαίδευση του δικτύου:

1. Τροφοδοτείται το στρώμα με τους κόμβους εισόδου με τις επιθυμητές εισόδους, δηλαδή το διάνυσμα εισόδων \mathbf{x} , του δικτύου.
2. Στους κόμβους εισόδου επισυνάπτονται τυχαία και διακριτά συναπτικά βάρη μέσω του διανύσματος \mathbf{w} και οι πολώσεις με το διάνυσμα \mathbf{b} .
3. Υπολογίζεται μέσω της συνάρτησης ενεργοποίησης \mathbf{g} ο πίνακας \mathbb{H} που συνδέει το διάνυσμα εισόδου, με το διάνυσμα βαρών εισόδου και το διάνυσμα των πολώσεων εισόδου.
4. Ακολουθεί ο υπολογισμός του γενικευμένου ανάστροφου των Moore – Penrose \mathbb{H}^+ από τον πίνακα \mathbb{H} .
5. Μέσω του γινομένου του πίνακα του γενικευμένου ανάστροφου των Moore – Penrose \mathbb{H}^+ με το διάνυσμα στόχων \mathbf{t} , υπολογίζεται το διάνυσμα βαρών εξόδων $\hat{\beta}$.
6. Τέλος, μέσω του γινομένου του διανύσματος βαρών εξόδου $\hat{\beta}$ με τον πίνακα εξόδου \mathbb{H} από τη συνάρτηση \mathbf{g} , καταλήγουμε στο διάνυσμα προβλέψεων του αλγορίθμου \mathbf{o} .
7. Το σφάλμα του δικτύου προκύπτει από τη διαφορά του \mathbf{t} διανύσματος στόχων με το \mathbf{o} διάνυσμα προβλέψεων. Όσο πιο μικρή είναι η διαφορά των τετραγώνων των δύο διανυσμάτων, τόσο πιο καλή γενίκευση έχει ο αλγόριθμος.

5.2 Αποτίμηση χρήσης του ELM σε σχέση με τους συμβατικούς SLFN αλγόριθμους

Σε όλες τις περιπτώσεις των αλγορίθμων των νευρωνικών δικτύων, κατά την επιλογή ενός αλγορίθμου για την εκπαίδευση και τις προσομοιώσεις μέσω των συνόλου των παρατηρήσεων, εμφανίζεται ένας διπλός αντίκτυπος κατά τη διεξαγωγή των διεργασιών. Αυτός ο διπλός αντίκτυπος διχοτομείται σε θετικό και αρνητικό, όπου αντιπροσωπεύονται τα πλεονεκτήματα και τα μειονεκτήματα αντίστοιχα.

5.2.1 Πλεονεκτήματα ELM

Ο προτεινόμενος αλγόριθμος του ELM έχει κάποια πολύ σημαντικά πλεονεκτήματα σε σχέση με τους υπόλοιπους αλγόριθμους SLFN που εφαρμόζονται στα νευρωνικά δίκτυα. Τα βασικότερα εξ' αυτών είναι τα ακόλουθα:

- Ο αλγόριθμος ELM δεν προβληματίζεται από τον ρυθμό εκμάθησης η όπως οι υπόλοιποι SLFN αλγόριθμοι νευρωνικών δικτύων καθώς δεν έχει. Για εκείνους όσο το η είναι μικρό τόσο πιο πολύ καθυστερούν να συγκλίνουν, ενώ αν το η είναι πολύ μεγάλο, τότε μπορεί να αποκλίνει ο αλγόριθμος εκμάθησης και να γίνει ασταθές το δίκτυο.
- Ο ELM δεν έχει την ιδιορρυθμία της διακοπής της εκπαίδευσης λόγω τοπικών ελαχίστων.

- Ένα νευρωνικό δίκτυο μπορεί να εκπαιδευτεί περισσότερο από το επαρκές και αποδεκτό μέσω του Backpropagation, με αποτέλεσμα να αποκτήσει χειρότερη γενίκευση ή και να μην συγκλίνει.
- Η μέθοδος της καθοδικής κλίσης είναι χρονοβόρα ειδικότερα σε πολύ μεγάλα προβλήματα και εφαρμογές.

5.2.2 Μειονεκτήματα – Περιορισμοί ELM

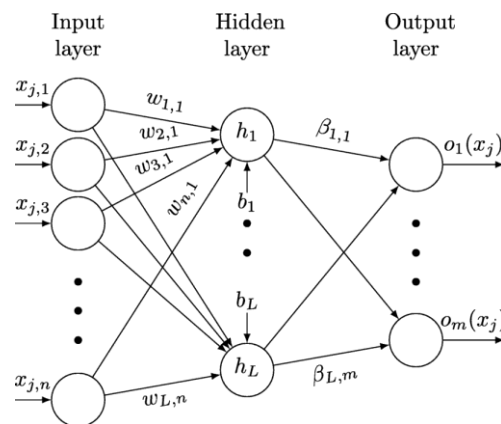
Σε αυτό το σημείο παραθέτουμε κάποια μειονεκτήματα – περιορισμούς της μεθόδου ELM τα οποία είναι τ' ακόλουθα:

- Μέσω του ELM αλγορίθμου οι προσεγγίσεις που γίνονται βασίζονται στην τυχαιότητα του μετασχηματισμού των δεδομένων. Επειδή δεν υπάρχει ακρίβεια στις προσαρμογές που κάνει εξ' αρχής αλλά βασίζεται στις τυχαίες μετατροπές που κάνει, ενδέχεται για προβλήματα που χρειάζονται την ακρίβεια για να λυθούν στη προσαρμογή των παραμέτρων, να μην είναι αρμοστός.
- Ένας επιπλέον περιορισμός του αλγορίθμου είναι ότι για την εκμάθηση ζητάει το σύνολο των δεδομένων εξ' αρχής, κάτι το οποίο δε βοηθάει σε καταστάσεις που τα προβλήματα χρειάζονται μία μελλοντική ή διαδοχική εκχώρηση δεδομένων για να μπορέσουν να ανταποκριθούν.

5.3 Τοπολογία και παραλλαγές του ELM

5.3.1 Τοπολογία του ELM

Μια σχετική τοπολογία για τη σχεδιαστική απεικόνιση της δομής ενός νευρωνικού δικτύου με τη χρήση του ELM, παρουσιάζεται στην αμέσως επόμενη εικόνα και δείχνει τη σύνδεση των εισόδων – εξόδων με τα βάρη εισόδου, βάρη εξόδου, τις πολώσεις και των πίνακα εξόδου \mathbb{H} του κρυφού στρώματος.



Εικόνα 19 Τοπολογία ELM

Όπως φαίνεται στην Εικόνα 19 το διάνυσμα εισόδων \mathbf{x} προηγείται και απαρτίζει το στρώμα εισόδου. Στη συνέχεια επισυνάπτεται το διάνυσμα συναπτικών βαρών \mathbf{w} και το διάνυσμα πολώσεων \mathbf{b} τα οποία υπολογίζουν μέσω της \mathbf{g} τον πίνακα \mathbb{H} . Τέλος, εφόσον βρεθεί ο γενικευμένος ανάστροφος πίνακας \mathbb{H}^+ υπολογίζεται το διάνυσμα βαρών εξόδου β , για καταλήξει το δίκτυο στην προσέγγιση του \mathbf{o} διανύσματος, που είναι και η τελική έξοδος του δικτύου.

5.3.2 Παραλλαγές του ELM

Ο ELM όπως και το σύνολο των αλγορίθμων κατά το πέρας των χρόνων μετά την εισήγησή τους, έχουν αναπτυχθεί και τροποποιηθεί σε διάφορες παραλλαγές οι οποίες βοηθάνε σε ορισμένους τομείς για να εξασφαλιστεί καλύτερη αποτελεσματικότητα. Παρακάτω αναφέρονται ονομαστικά κάποιες από τις παραλλαγές του ELM [29], χωρίς να γίνεται κάποια επιπλέον ανάλυση σε αυτές τις υποκατηγορίες:

1. Incremental ELM (**I-ELM**, **CI-ELM**, **EI-ELM**)
2. Pruning ELM (**P-ELM**)
3. Error-Minimized ELM (**EM-ELM**)
4. Two-Stage ELM (**TS-ELM**)
5. Online sequential ELM (**OS-ELM**, **EOS-ELM**, **FOS-ELM**)
6. Evolutionary ELM (**E-ELM**)
7. Voting-based ELM (**V-ELM**)
8. Ordinal ELM
9. Fully complex ELM (**C-ELM**)
10. Symmetric ELM (**S-ELM**)

5.4 Η προσέγγιση των Widrow – Krause

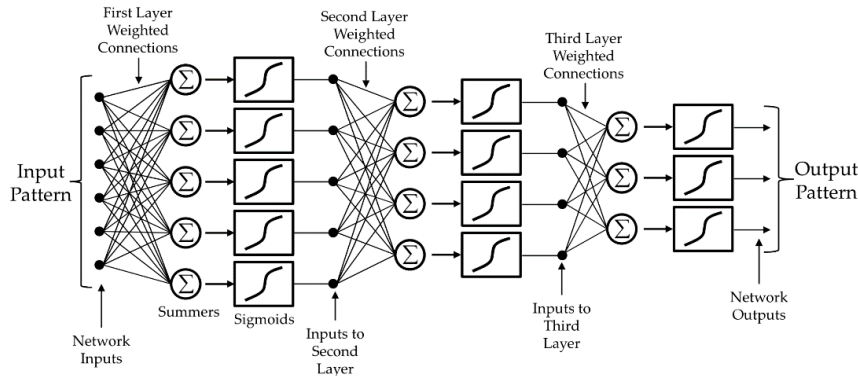
Ο **Bernard Widrow** (1924 –) είναι Αμερικανός καθηγητής Ηλεκτρολόγος Μηχανικός ο οποίος το 1960 μαζί με τον Ted Hoff ανακάλυψαν το ADALINE ή αλλιώς όπως αποκαλείται “delta rule”, που ουσιαστικά είναι μέθοδος LMS (Least Mean Square) κατά την οποία γίνεται η αναπροσαρμογή των συναπτικών βαρών, μέσω της οποίας διορθώνεται σταδιακά το σφάλμα και ελαχιστοποιείται. Πρακτικά, είναι μία από τις βασικές αρχές που ακολουθούν το σύνολο των αλγορίθμων εκμάθησης που αναπτύχθηκαν κατά τα χρόνια, εξακολουθώντας να συμβάλει τα μέγιστα και στις νέες μεθόδους που αναπτύσσονται ακόμη και σήμερα [30].

5.4.1 Ο No-Prop αλγόριθμος του Widrow

Ο Bernard Widrow το **2013** σε μία επιστημονική εργασία (paper) που δημοσίευσε, για την οποία εργάστηκε μαζί με τους Aaron Greenblatt, Youngsik Kim και Dookun Park, εισήγαγε και πρότεινε έναν νέο αλγόριθμο τον οποίον ονόμασε No-Prop. Ο τίτλος της επιστημονικής εργασίας ήταν “The No-Prop algorithm: A new learning algorithm for multilayer neural networks”.

Τα βασικά σημεία του προτεινόμενου αλγορίθμου είναι ότι γίνεται εκπαίδευση μόνο στα συναπτικά βάρη των νευρώνων του επιπέδου εξόδου τα οποία έχουν αποδοθεί με τυχαίο τρόπο στους κρυφούς νευρώνες. Η εκπαίδευσή τους πραγματοποιείται μέσω της κλίσης για την ελαχιστοποίησή τους μέσω του αλγορίθμου LMS του Widrow και Hoff. Όταν το νευρωνικό δίκτυο μέσω του νέου αλγορίθμου μπορεί να διακρίνει ένα σύνολο από διακριτά μοτίβα χωρίς σφάλμα (LMS Capacity), τα οποία δεν είναι λιγότερα από το σύνολο των μοτίβων με τα οποία εκπαιδεύεται το δίκτυο, η γενίκευση και η εκπαίδευση του δικτύου είναι αντίστοιχη με αυτή της περίπτωσης του Backpropagation. Στην περίπτωση που ισχύει το αντίθετο, δηλαδή το πλήθος των μοτίβων ξεπεράσει το LMS Capacity, ο BP φαίνεται να είναι πιο αποδοτικός. Ωστόσο αυτό μπορεί ν’ αλλάξει με την αύξηση των νευρωνίων του κρυφού στρώματος. Σε κάθε περίπτωση όμως ο προτεινόμενος αλγόριθμος συγκλίνει κατά πολύ γρηγορότερα σε σχέση με τον BP [31].

Στην εισήγησή του για τον νέο αλγόριθμο ο Widrow χρησιμοποίησε στο paper το σχήμα της παρακάτω εικόνας (βλ. **Εικόνα 20**), στο οποίο απεικονίζεται ένα δίκτυο που αποτελείται από 3 κρυφά στρώματα, στα οποία εφαρμόζεται η σιγμοειδής συνάρτηση ενεργοποίησης και έχουν αποδοθεί τυχαία τα βάρη σε κάθε είσοδο. Πάνω σε αυτό το σχήμα, αναφέρει πως χωρίς να προβεί ο αλγόριθμος στην προσαρμογή των βαρών των δύο πρώτων στρωμάτων, παρά μόνο σε αυτά του 3^{ου} στρώματος, το δίκτυο μπορεί μόνο μέσω του επιπέδου εξόδου να έχει προσεγγίσει σωστά και χωρίς σφάλματα όλα τα δυνατά μοτίβα τα οποία είναι άγνωστα από τα δύο πρώτα επίπεδα. Οι εισοδοί συνεπώς του τρίτου στρώματος θα είναι διακριτές και γραμμικά ανεξάρτητες.



Εικόνα 20 Νευρωνικό δίκτυο τριών στρωμάτων

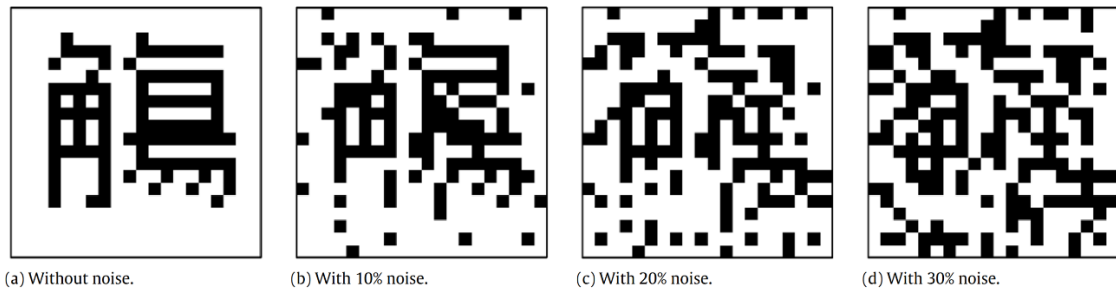
5.4.2 Τα πειράματα για τον No-Prop από τον Widrow

Μέσα από τα πειράματα και τις αξιολογήσεις τις οποίες προέβη ο Widrow, αποσαφηνίζει πως είναι σε πρώιμο στάδιο ακόμα ο αλγόριθμος, αναφέροντας πως δε γίνεται να εξασφαλιστεί η επιλογή του σε κάποιο πρόβλημα σε σχέση με τον BP, αλλά και ότι η δουλειά που γίνεται είναι ακόμη σε εξέλιξη.

Για να εφαρμόσει τη θεωρία που ανέλυσε για τον αλγόριθμο, προχώρησε σε μία σειρά από πειράματα μέσω του MATLAB, για τα οποία αναφέρει τα αποτελέσματα τα οποία είχε σε σχέση με τον Backpropagation, τόσο στον βαθμό της εύρεσης των μοτίβων, όσο και χρονικά τη σύγκλιση και την ταχύτητα που έχουν οι δύο μέθοδοι.

Για ένα πρόβλημα ταξινόμησης το οποίο αποτελείται από 20.000 κινέζικους χαρακτήρες, οι οποίοι απεικονίζονται σε εικόνες 20 x 20 pixels (μαύρο για τον χαρακτήρα, άσπρο για την υπόλοιπη εικόνα), ένταξε τυχαίο θόρυβο στις εικόνες της τάξεως του 5% έως 30%, για ν' αλλοιώσει την αρχική εικόνα (βλ. **Εικόνα 21**). Αρκετοί λοιπόν από αυτούς τους χαρακτήρες επιλέχθηκαν για εκπαίδευση και εξέταση. Για την εκπαίδευση χρησιμοποιήθηκαν 200 χαρακτήρες, που ο καθένας περιγράφεται από 400 στοιχεία – γνωρίσματα (αφού 20 x 20 pixels = 400 pixels), για τους οποίους 200 χαρακτήρες υπάρχουν 50 διαφορετικές εκδοχές τους που συμπεριλαμβάνουν τυχαίο θόρυβο, δομώντας το δίκτυο με 400 κρυφούς νευρώνες και 200 νευρώνες εξόδου. Για τα πειράματα χρησιμοποιήθηκαν 100 χαρακτήρες με θόρυβο. Ανάλογα το επίπεδο του θορύβου προστέθηκε θόρυβος ή όχι και στα δεδομένα εκπαίδευσης. Τέλος, η εκπαίδευση σταμάτησε στις 10.000 επαναλήψεις.

Οι προσομοιώσεις καταλήγουν με τους δύο αλγορίθμους να είναι πολύ κοντά στα επίπεδα σφαλμάτων, ιδιαίτερα όταν ο θόρυβος αυξάνεται και φτάνει στο 25% - 30%, με τον **No-Prop να φτάνει στο 2.92% σφάλμα και τον Backpropagation στο 2.75%**. Ωστόσο η σύγκλιση του No-Prop γίνεται πολύ πιο γρήγορα με τον ίδιο να συγκλίνει σε καθολικά ελάχιστα και όχι σε τοπικά όπως ο Backpropagation.



Εικόνα 21 Η μορφή Κινέζικου χαρακτήρα με – χωρίς θόρυβο (No – Prop)

Εν κατακλείδι, ολοκληρώνει την εισήγησή του ο Widrow και η συνεργάτες του, συμπεραίνοντας ότι ο No – Prop μπορεί να εκπαιδεύσει ένα δίκτυο, μόνο προσαρμόζοντας τα βάρη του τελευταίου κρυφού επιπέδου σε ένα πολυστρωματικό νευρωνικό δίκτυο, με την ικανότητα να το καταφέρνει αυτό έχοντας στο κρυφό στρώμα περισσότερους ή ίσους νευρώνες από το πλήθος των προτύπων που πρόκειται να αναγνωρίσει [31].

5.4.3 Τα πειράματα του Krause για τον No-Prop

Ο André Frank Krause βασισμένος στον αλγόριθμο No-Prop που πρότεινε ο Widrow, επιχείρησε να πειραματιστεί μαζί του για να εξακριβώσει την αξιοπιστία του και την αποτελεσματικότητά του. Για να το πετύχει αυτό έκανε μία σειρά από πειράματα, με κυριότερο το πείραμα με το πρόβλημα MNIST. Το MNIST (το οποίο το χρησιμοποιούμε και εμείς στα πειράματά μας) είναι μία βάση από χειρόγραφες παραστάσεις των ψηφίων (0 – 9), τα οποία έχουν γίνει ψηφιακές εικόνες 28 x 28 pixels. Αυτές οι εικόνες αντιπροσωπεύουν μόνο ένα ψηφίο όπως είναι προφανές, και πρόκειται για ένα πείραμα που έχει μεγάλο μέγεθος σε σύνολο δεδομένων και το οποίο ακολουθούν πολλοί για εξακριβώσουν την επιτυχία κάποιου νευρωνικού αλγορίθμου και του δικτύου τους.

Ο Krause έφτιαξε ένα δίκτυο με 4000 νευρώνες στο κρυφό στρώμα, το οποίο έφτασε στην ολοκλήρωση της εκπαίδευσης σε 100 sec με τη χρήση του No-Prop-Fast σε σχέση με τα 1000 sec που ήθελε ο πιο γρήγορος BP μέσω του MATLAB toolbox, ο οποίος είναι ο Scaled Conjugate Gradient. Η καλύτερη επιτυχία που είχε ο No-Prop-Fast ήταν το 97.7% με ένα δίκτυο που αποτελείται από 8000 νευρώνες, σε αντίθεση με το 99.6% που μπορεί να φτάσει ο BP. Φυσικά ο χρόνος που χρειάζεται ο ένας σε σχέση με τον άλλο διαφέρει κατά πολύ.

Επιπρόσθετα, επιχείρησε κι άλλα πειράματα με τα λεγόμενα eye – tracking, τα οποία βασίζονταν πάνω σε εικονογραφημένα cartoons, για την αναγνώριση του φύλου ή της ηλικίας των cartoons είτε αυτά περιείχαν κάποιο κείμενο είτε όχι. Και για αυτά υπάρχουν αναλυτικά τα οι μετρήσεις όπως και σχετικές εικόνες για την καλύτερη αποσαφήνιση των προβλημάτων.

Στο paper που δημοσίευσε ο Krause, αναφέρει χαρακτηριστικά τις μετρήσεις σε επιτυχία αλλά και ταχύτητα εκμάθησης, δείχνοντας ότι ο αλγόριθμος έχει τις δυνατότητες να βελτιωθεί και να αποδειχθεί ένας πολύ αξιόλογος και αξιόπιστος αλγόριθμος [31] [30].

Κεφάλαιο 6

Ανάλυση Προβλημάτων και Πειραμάτων

6.1 Το πρόβλημα της Συναρτησιακής Παλινδρόμησης

Για τη σύγκριση και την αποτίμηση του ELM σε σχέση με τον Backpropagation, βασιστήκαμε και στις δύο κατηγορίες προβλημάτων, την **Παλινδρόμηση (Regression)** και την **Ταξινόμηση (Classification)**. Ως αφετηρία των πειραμάτων μας είχαμε την Παλινδρόμηση και συγκεκριμένα τη **Συναρτησιακή Παλινδρόμηση** χρησιμοποιώντας τη συνάρτηση $y = \text{sinc}(x)$.

Καταρχάς να διευκρινίσουμε ότι η $y = \text{sinc}(x)$ είναι όμοια με τη $y = \sin(x) / x$ με $x = 0 \rightarrow y(0) = 1$, καθώς η πρώτη μεταφράζεται ως $y = \frac{\sin \pi x}{\pi x}$, η οποία πρακτικά ικανοποιεί την ίδια κυματομορφή με τη διαφορά ότι λόγω του π , έχει περίπου 3.14 πιο γρήγορη συχνότητα. Αυτό σημαίνει ότι η $y = \text{sinc}(x)$ και η $y = \sin(x) / x$ για $x = 0 \rightarrow y(0) = 1$, αλλά η πρώτη θα μηδενίζεται για τις ακέραιες τιμές του x ενώ η δεύτερη για τα πολλαπλάσια του π . Έχοντας αυτό ως προϋπόθεση, στα δικά μας πειράματα **επιλέξαμε** τη **sinc** σε σχέση με τον **Huang** που επέλεξε την $\frac{\sin x}{x}$, κάτι το οποίο δε θα επηρεάσει επ' ουδενί τα πειράματα μας, αλλά θα μας βοηθήσει ώστε να είναι πιο συμπτυκνωμένα τα αποτελέσματά μας.

Όσον τα πειράματα αυτά καθαυτά, επιλέξαμε να τρέξουμε πειράματα χωρίζοντας τα δεδομένα σε σύνολα με τις αναλογίες να είναι **75% - 25%**, **50% - 50%** και **75% - 25% training dataset – testing dataset** αντίστοιχα. Αυτό σημαίνει ότι ξεκινήσαμε με περισσότερα δεδομένα για εκπαίδευση και λιγότερα για την εξέταση, όπου σταδιακά αντιστράφηκε αυτή η συνθήκη. Αυτό έγινε στα σύνολα δεδομένων και των δύο εξεταζόμενων αλγορίθμων.

Για να εξάγουμε τα δεδομένα, χρησιμοποιήσαμε τα **διαστήματα [-5, +5]** και **[-10, +10]** με το **βήμα** να είναι το **0.01** για να πάρουμε τις διακριτές τιμές των συνόλων. Αυτή την αρχή κρατήσαμε σε όλα τα πειράματα που τρέξαμε ανεξαρτήτως των παραλλαγών ή της επιλογής των διαστημάτων που θα διεξάγουμε τις προσομοιώσεις για κάθε παραλλαγή.

Οι **παραλλαγές** των πειραμάτων για την Συναρτησιακή Παλινδρόμηση είναι **3**. Δηλαδή με βάση τη συνάρτηση $y = \text{sinc}(x)$ υλοποιήσαμε τρεις διαφορετικές φιλοσοφίες πειραμάτων, όπου στην πρώτη τα δεδομένα είναι καθαρά και για τα δύο σύνολα προσομοιώσεων. Στη δεύτερη εντάσσουμε θόρυβο στα στοιχεία των διανυσμάτων y . Στην τρίτη αφαιρούμε μέρη από το γενικό σύνολο τιμών, εκπαιδεύοντας το δίκτυο με ελλιπές σύνολο τιμών, αλλά εξετάζουμε την αποτελεσματικότητά του όχι μόνο στα υποσύνολα που εκπαιδεύτηκε αλλά και σε αυτά που δεν έχει εκπαιδευτεί. Μάλιστα σε αυτή την περίπτωση αυξήσαμε ακόμη περισσότερο την προβληματικότητα στα δεδομένα, εντάσσοντας και θόρυβο στα διανύσματα εξόδων. Τέλος, σε όλα τα πειράματα για την παλινδρόμηση χρησιμοποιήσαμε τις **Levenberg – Marquardt** και **Resilient Backpropagation** μεθόδους για τον BP.

Βασικό κριτήριο της αξιολόγησης των αποτελεσμάτων είναι το RMSE στην εκπαίδευση και στην εξέταση του δικτύου, όπως επίσης και η γραφικές απεικονίσεις σε σύγκριση με την πραγματική γραφική παράσταση της $y = \text{sinc}(x)$. Να αναφέρουμε δε ότι τα διανύσματα τιμών που δημιουργούμε, στις δύο πρώτες παραλλαγές χρησιμοποιούνται για εκπαίδευση και τυχαίες τιμές πλήθους ίσο με το πλήθος των τιμών του αρχικού συνόλου, παρμένες μέσα από το αρχικό σύνολο τιμών με ακρίβεια 0.0001 επιλέγονται για να δημιουργηθεί το διάνυσμα τιμών για το σκέλος εξέτασης. Αντίθετα στην τρίτη παραλλαγή, αλλάζει αυτή η φιλοσοφία και δίνεται το διάστημα το αρχικό για εξέταση και όχι εκπαίδευση, με σκοπό να διερευνήσουμε αν στα διαστήματα που λείπουν από το αρχικό διάστημα το δίκτυο μπορεί να αναγνωρίσει τη συνέχεια της συνάρτησης και να παρουσιάσει τις προσεγγίσεις που θα είναι ίδιες ή κοντά στις πραγματικές τιμές της συνάρτησης που εξετάζουμε.

Πιο αναλυτικά στην πρώτη περίπτωση όπου τα σύνολα τιμών δεν έχουν υποστεί κάποια επεξεργασία, τα σύνολα τιμών ήταν τα **[-5, +5]** και **[-10, +10]**, για τα οποία το βήμα των διακριτών τιμών ήταν **0.01**. Στο πλήθος των **νευρώνων** είχαμε από **2 – 20** για των ELM και για το πρώτο σύνολο τιμών, ενώ για το

δεύτερο το πλήθος των νευρώνων κυμάνθηκε από **10 – 40**. Για τα πειράματα με BP, εντάχθηκε και **Ρυθμός Εκμάθησης = 0.3 – 1** και οι νευρώνες στο κρυφό στρώμα από **5 – 60**. Τα διαστήματα είναι τα ίδια. Και σε ό,τι αφορά τις εποχές αφήνουμε το δίκτυο να τρέχει μέχρι **1000 εποχές** αν χρειαστεί σταματώντας σε περίπτωση που τις υπερβεί. Συνεπώς, μέσω των αλγορίθμων BP το δίκτυο έχει πλήρη ελευθερία για το πότε θα σταματήσει την εκπαίδευση εφόσον έχει προσαρμόσει σε σωστό βαθμό τα συναπτικά του βάρη, χωρίς να το περιορίζουμε εμείς για το πότε θα το επιχειρήσει ή να σταματάμε το δίκτυο από το να εκπαιδευτεί πλήρως.

Στη δεύτερη παραλλαγή κρατήσαμε τα ίδια διαστήματα τιμών, το ίδιο βήμα και προσθέσαμε στα διανύσματα εξόδου θόρυβο ομοιόμορφα κατανεμημένο για όλα τα στοιχεία σε όλο το διάστημα ο οποίος κυμαίνεται στο διάστημα **[-0.1, +0.1]**. Όπου χρειάστηκε αυξήσαμε το πλήθος των νευρώνων και στους δύο αλγορίθμους ώστε να πάρουμε τις καλύτερες δυνατές προσεγγίσεις.

Στην τρίτη και πιο δύσκολη παραλλαγή εξετάσεων, προχωρήσαμε στην αφαίρεση διακριτών και μη συνεχόμενων διαστημάτων από τα σύνολα τιμών. Διαχωρίσαμε ανάλογα με το μέγεθος του διαστήματος το μήκος του διανύσματος που θα αφαιρέσουμε, καθώς και το πλήθος των διανυσμάτων που θα αφαιρεθούν. Χαρακτηριστικά για το διάστημα τιμών **[-5, +5]** αφαιρέσαμε **3 υποδιαστήματα** που το καθένα είχε **μήκος 1 μονάδα**. Συνολικά, **αφαιρέσαμε 3 από τις 10 μονάδες** του αρχικού συνόλου, δημιουργώντας ένα νέο σύνολο τιμών μέσα από το οποίο θα επιλεγθούν τυχαία κάποιες τιμές. Ομοίως, για το **[-10, +10]** αφαιρέσαμε **4 υποδιαστήματα 1.5 μονάδας** το καθένα. Άρα συνολικά **αφαιρούμε το 30%** σε κάθε περίπτωση, όπου επιλέγονται τυχαία τιμές στα νέα διαστήματα οι οποίες όμως είναι ίσες ή λιγότερες από το πλήθος των στοιχείων του αρχικού διαστήματος αφαιρώντας το πλήθος των στοιχείων από τα διανύσματα τιμών που αποκόπηκαν. Συνεπώς, π.χ. στο διάστημα **[-5, +5]** με βήμα **0.01** δημιουργείται ένα διάνυσμα x το οποίο περιέχει 1000 τιμές διακριτές. Το 30% του διαστήματος αυτού θα αφαιρεθεί και θα μείνουν **700 διακριτές τιμές** για τα x . Έτσι στο νέο μη-συνεχές διάστημα θα πρέπει να επιλεγθούν μέχρι 700 διακριτές τιμές οι οποίες θα χρησιμοποιηθούν για εκπαίδευση. Το αρχικό διάστημα τιμών με το διάνυσμα τιμών που δημιουργήσαμε εξ' αρχής, χρησιμοποιείται πλέον για τις προσομοιώσεις εξέτασης και όχι για την εκπαίδευση όπως κάναμε στις προηγούμενες δύο παραλλαγές.

Σαν δεύτερο σκέλος αυτής της παραλλαγής είναι η ένταξη θορύβου αντίστοιχου με την προηγούμενη παραλλαγή, δηλαδή ομοιόμορφα κατανεμημένου σε όλα τα στοιχεία των διανυσμάτων εξόδου ο οποίο είναι επιλεγμένος μέσα από το διάστημα **[-0.1, +0.1]**.

Περισσότερες πληροφορίες για το κάθε πείραμα ξεχωριστά θα διατυπώνονται στο **Κεφάλαιο 7** όπου αναπτύσσουμε το κάθε πρόβλημα και την κάθε περίπτωση παραμέτρων και παραλλαγών ξεχωριστά.

6.2 Τα προβλήματα Ταξινόμησης

Τα προβλήματα **Ταξινόμησης (Classification)** είναι η δεύτερη κατηγορία προβλημάτων και πιο αντιπροσωπευτική, καθώς έχουμε να αντιμετωπίσουμε προβλήματα που έχουν τιμές με ετικέτα, συνεπώς θα πρέπει να τοποθετηθούν σε κάποιο σύνολο ή κατηγορία ανάλογα με την περιγραφή των τιμών των παρατηρήσεων.

Καταρχάς σε αυτά τα προβλήματα εξετάσαμε πέρα από τον Levenberg – Marquardt και τον Resilient Backpropagation που εμφανίστηκαν στο πρόβλημα της συναρτησιακής παλινδρόμησης, και τον **Scaled Conjugate Gradient Backpropagation**. Με αυτόν τον τρόπο προσθέσαμε μία επιπλέον επιλογή νευρωνικού αλγορίθμου, ο οποίος θα συγκριθεί με τους δύο που προϋπάρχουν αλλά και με τον ELM. Ένας σημαντικός λόγος που επιλέξαμε μία επιπλέον παραλλαγή του BP, είναι γιατί πλέον ο χρόνος στις περιπτώσεις που το πρόβλημα είναι ογκώδες ή περίπλοκο, ξεκινάει και παίζει σημαντικό ρόλο, καθώς ένας αλγόριθμος δεν πρέπει να είναι μόνο αποτελεσματικός, αλλά να μπορεί να μπορεί να περατώνει τις προσομοιώσεις σε ικανοποιητικό χρονικό διάστημα. Τέλος, επειδή δε γνωρίζουμε ποια ή ποιες υλοποιήσεις του BP εξέτασε ο Huang, επιλέξαμε τις κατά τα φαινόμενα πιο αποδοτικές για να είμαστε τόσο καλυμμένοι όσο και να αποκτήσουμε καλύτερη άποψη για τα αποτελέσματα.

Το πλήθος των προβλημάτων που επιλέξαμε είναι τέσσερα. Τα δύο πρώτα προβλήματα είναι από τα πιο γνωστά στην κοινότητα, τα οποία χρησιμοποιούνται κατά κόρον για τους πειραματισμούς στα Τεχνητά Νευρωνικά Δίκτυα, τα οποία είναι το **IRIS** και το **MNIST**. Όμως, αυτά τα δύο προβλήματα δεν συμπεριλαμβάνονται και στα πειράματα που παρουσιάζει ο Huang στην επιστημονική εργασία για την εισήγηση του ELM. Γι' αυτό το λόγω επιλέξαμε δύο προβλήματα για τα οποία έκανε πειράματα και παρουσίασε τις μετρήσεις τόσο για τον ELM όσο και για τον BP ο Huang. Αυτά τα δύο προβλήματα είναι το **IMAGE SEGMENT** και το **SATELLITE IMAGES**. Σε αυτή την ενότητα θα αναλύσουμε τη δομή του καθενός καθώς και τις επιλογές που κάναμε για να τρέξουμε τα πειράματα και εμείς.

Το κάθε σύνολο παραμέτρων που επιλέγουμε το επαναλαμβάνουμε για περισσότερες από μία φορές, ώστε να βγάλουμε τον Μ.Ο. των μεγεθών που εξετάζουμε για την αποτελεσματικότητα του αλγορίθμου.

Τα μεγέθη που εξετάζουμε είναι:

- Ποσοστό Ακρίβειας – Ευστοχίας Εκπαίδευσης (Testing Accuracy)
- Ποσοστό Ακρίβειας – Ευστοχίας Εξέτασης (Testing Accuracy)
- Χρόνος Εκτέλεσης (ατομικής εκτέλεσης και συνολικών εκτελέσεων)
- RMSE Training
- RMSE Testing

Μεγαλύτερη σημασία δίνουμε στα ποσοστά ακρίβειας και στον χρόνο εκτέλεσης και δευτερευόντως στα RMSE, καθώς στα προβλήματα Ταξινόμησης βασιζόμαστε στην ακρίβεια που δικτύου να τοποθετήσει στη σωστή τάξη – κλάση – κατηγορία τις παρατηρήσεις, σε σχέση με το μέσο τετραγωνικό σφάλμα όπως στα προβλήματα συναρτησιακής παλινδρόμησης.

Τα πειράματα έγιναν σε ένα σύστημα **8 πυρήνων με hyperthreading** και **16 εικονικούς πυρήνες**. Ο **επεξεργαστής (CPU)** είναι ο **AMD Ryzen 7 3700x** ο οποίος έχει **base clock στα 3.6GHz** και **boost clock στα 4.4GHz**. Η έκδοση του **MATLAB** είναι η **R2023a**. **Οι προσομοιώσεις για τα προβλήματα Ταξινόμησης έγιναν με τη βοήθεια του παράλληλου υπολογισμού**, εργαλείο το οποίο προσφέρεται από το ίδιο το MATLAB για την γρηγορότερη εκτέλεση των προσομοιώσεων.

6.2.1 Η αναγνώριση του φυτού “IRIS”

Το πρόβλημα **IRIS** είναι ένα από τα πιο γνωστά προβλήματα ταξινόμησης, το οποίο χρησιμοποιείται για την αποτίμηση ενός αλγορίθμου εκμάθησης νευρωνικών δικτύων μέσω του οποίου αποτιμάται η αποτελεσματικότητα ενός νευρωνικού αλγορίθμου και κατ' επέκταση και η αποδοτικότητα του ίδιου του δικτύου.

Το πρόβλημα βασίζεται σε παρατηρήσεις οι οποίες αφορούν **το φυτό της Ίριδας** και περιγράφεται από τέσσερα διαφορετικά γνωρίσματα. Αυτά είναι το μήκος και το πλάτος του πέταλου όπως και το μήκος και πλάτος του σέπαλου. Αυτά οι τέσσερις μετρήσεις ή γνωρίσματα περιγράφουν μία παρατήρηση. Αυτή η παρατήρηση μπορεί να ενταχθεί σε μία από τις τρεις διαφορετικές κατηγορίες. Απαραίτητη προϋπόθεση κάθε παρατήρηση να ανήκει σε μία μόνο κατηγορία. Οι τρεις κατηγορίες για την ιστορία που κατηγοριοποιείται η παρατήρηση είναι οι *Setosa*, *Virginica* και *Versicolor*. Ωστόσο στο τέλος δε μας απασχολεί το πως ονομάζεται η κάθε κατηγορία. Μας ενδιαφέρει το πλήθος αυτών αλλά και η δυαδική αναπαράσταση αυτών. Συνεπώς, εφόσον είναι τρεις κατηγορίες σημειώνουμε με το bit “**1**” τη θέση του δυαδικού ψηφίου το οποίο πραγματικά ανήκει η παρατήρηση και τα υπόλοιπα δύο bits έχουν την τιμή “**0**”. Άρα οι τρεις κατηγορίες περιγράφονται ως “**001**”, “**010**” και “**100**”.

Η συγκεκριμένη βάση με παρατηρήσεις αποτελείται από **150**. Αυτές τις 150 παρατηρήσεις τις χωρίζουμε σε **50% - 50%** και **70% - 30%** αναλογίες για training – testing datasets. Αυτός ο διαχωρισμός γίνεται τυχαία. Βασικός κανόνας για την επιλογή είναι ότι από το σύνολο των παρατηρήσεων της εκάστοτε κατηγορίας θα παρθεί το αντίστοιχο ποσοστό που έχουμε επιλέξει για την εκπαίδευση και αντίστοιχο για την εκπαίδευση. Συνεπώς, στις 150 παρατηρήσεις που ανήκουν σε τρεις

κατηγορίες, αν οι 50 ανήκουν στην κατηγορία 1, τότε αν το ποσοστό για εκπαίδευση είναι 70%, οι 35 παρατηρήσεις της κατηγορίας 1 θα χρησιμοποιηθούν για εκπαίδευση και οι υπόλοιπες 15 από τις 50 που απαρτίζουν το 30% θα χρησιμοποιηθούν για εξέταση. Το ίδιο θα συμβεί για όλες τις άλλες παρατηρήσεις των κατηγοριών. Με αυτό τον τρόπο, εφαρμόζουμε τις αναλογίες ξεχωριστά σε κάθε κατηγορία και όχι στο σύνολο των παρατηρήσεων, εξασφαλίζοντας ότι το δίκτυο εκπαιδεύει και θα πειραματιστεί δίκαια. Σε αντίθετη περίπτωση, τα σύνολα μπορεί να προκύπτανε μεροληπτικά και το δίκτυο να είχε καλύτερη προσέγγιση σε ένα σύνολο παρατηρήσεων σε σχέση με ένα άλλο. Η συγκεκριμένη προϋπόθεση έχει εξασφαλιστεί για όλα τα προβλήματα ταξινόμησης που θα συζητήσουμε και θα λέμε για πιο άμεση κατανόηση ότι **“ο διαχωρισμός των παρατηρήσεων έχει γίνει με αμερόληπτο τρόπο ως προς τα ποσοστά αναλογίας των συνόλων”** ή πιο απλά ότι **“τα σύνολα παρατηρήσεων είναι αμερόληπτα”**.

Τέλος, για να μπορέσουμε να εξασφαλίσουμε τη μεγαλύτερη δυνατή αμεροληψία στα αποτελέσματα των προσομοιώσεων, επαναλάβαμε το κάθε ξεχωριστό σύνολο παραμέτρων για ένα πείραμα, 10 συνεχόμενες φορές. Αυτό σημαίνει ότι π.χ. αν επιλέξουμε για μία ορισμένη αναλογία ποσοστών των συνόλων ένα συγκεκριμένο πλήθος κρυφών νευρώνων, μία συνάρτηση ενεργοποίησης ή ρυθμό εκμάθησης και παραλλαγή αλγορίθμου (ανάλογα τον αλγόριθμο που εξετάζουμε) θα επαναλάβουμε για τις ίδιες παραμέτρους 10 φορές τις προσομοιώσεις, παίρνοντας στο τέλος τον μέσο όρο των μεγεθών που εξετάζουμε και όχι την καλύτερη δυνατή εκτέλεση.

6.2.2 Το πρόβλημα του διαχωρισμού αντικειμένων “IMAGE SEGMENT”

Το **Image Segmentation** ή αλλιώς **Image Segment** είναι ένα σύνολο παρατηρήσεων που υλοποιήθηκαν για τους λόγους εξέτασης των νευρωνικών δικτύων και των αλγορίθμων στα προβλήματα ταξινόμησης. Σαν μέθοδος χρησιμοποιείται ευρέως και συνέχεια όχι μόνο για σκοπούς αξιολόγησης αλλά και σαν εργαλείο κατανόησης, παρατηρήσεων κ.τ.λ. σε άλλους τομείς που είναι χρήσιμο.

Πρόκειται για το ένα από τα δύο προβλήματα ταξινόμησης που επιλέξαμε να επαναλάβουμε τις προσομοιώσεις, για το οποίο έχει και ο Huang εκτελέσει και παρουσιάσει τα αποτελέσματα στην επιστημονική του εργασία.

Το σύνολο παρατηρήσεων που απαρτίζουν το πρόβλημα Image Segment, έχει προέλθει από μία **βάση 7 εικόνων εξωτερικού χώρου** που έχουν σχεδιαστεί τυχαία. Αυτές οι εικόνες δημιουργούν instances (παραδείγματα) 3 x 3 pixels, όπου το καθένα αντιπροσωπεύει ένα κομμάτι από τα αντικείμενα που έχουν δημιουργηθεί. Έτσι δημιουργούνται πολλές παρατηρήσεις από το κάθε αντικείμενο των εικόνων με σκοπό να αναγνωριστούν μέσα από αυτά τα παραδείγματα – παρατηρήσεις τα αντικείμενα [32].

Τα παραδείγματα – παρατηρήσεις περιγράφονται από ένα σύνολο **19 γνωρισμάτων** τα οποία αφορούν μετρήσεις που περιγράφουν πλήρως τις υπό-εικόνες **3 x 3 pixels**. Η πλήρη επεξήγηση του κάθε γνωρίσματος δεν είναι αντικείμενο αυτής της διπλωματικής. Για περισσότερες πληροφορίες μπορείτε να επισκεφθείτε τον ακόλουθο σύνδεσμο <https://archive.ics.uci.edu/dataset/50/image+segmentation> όπου διατυπώνονται αναλυτικά οι περιγραφές των μεταβλητών, αλλά μπορείτε να αποκτήσετε πρόσβαση και αν κατεβάσετε το αρχείο με τις παρατηρήσεις.

Το **σύνολο των παρατηρήσεων είναι 2310** όπου κάθε παρατήρηση μπορεί να κατηγοριοποιηθεί σε μία από τις **7 διαθέσιμες κλάσεις**, που έχουν προκύψει από τα αντικείμενα των εικόνων και είναι:

1. ουρανός
2. φύλλωμα
3. γρασίδι
4. μονοπάτι – δρόμος
5. τούβλο
6. τσιμέντο
7. παράθυρο

Οι κατηγορίες περιγράφονται από 7-bit εξόδους δυαδικές, όπως και στα προηγούμενα προβλήματα, με τη θέση του bit που είναι ίσο με 1 να αποτελεί την κλάση στην οποία ανήκει. Να σημειώσουμε δε ότι

το ποια κλάση και με ποιο 7-bit συνδυασμό έχει αντιπροσωπευτεί, είναι τελείως τυχαία και δεν επηρεάζει το τελικό αποτέλεσμα.

Ως προς τη διεξαγωγή των προσομοιώσεων δημιουργήσαμε 3 διαφορετικά ζεύγη συνόλων για εκπαίδευση και για εξέταση, τα οποία είναι **50% - 50%**, **65% - 35%** και **80% - 20%** **training – testing** αντίστοιχα. **Ο διαχωρισμός των παρατηρήσεων έγινε με αμερόληπτο τρόπο** που σημαίνει ότι βάσει του εκάστοτε ποσοστού που αντιστοιχεί στο κάθε σύνολο, έγινε τυχαία επιλογή από το άθροισμα των παρατηρήσεων της κάθε κατηγορίας, ούτως ώστε να μην ευνοηθεί κάποια κατηγορία περισσότερο από κάποια άλλη από την τυχαία επιλογή. Το **65% - 35%** είναι το ποσοστό που επέλεξε και ο Huang να παρουσιάσει στα αποτελέσματά του, συνεπώς το επιλέξαμε και εμείς για να κάνουμε σύγκριση και με τα δικά του αποτελέσματα. Αναμφίβολα δε γνωρίσουμε τον τρόπο που διαχώρισε τις παρατηρήσεις του ο ίδιος, παρά μόνο πόσες χρησιμοποίησε για το κάθε σύνολο. Επίσης, δε γνωρίζουμε αν επέλεξε να επεξεργαστεί με κάποιον τρόπο τις μετρήσεις παρατηρήσεων όπως π.χ. κανονικοποίηση οπότε η σύγκριση δεν είναι απολύτως ξεκάθαρη.

Τέλος, όπως στα προηγούμενα προβλήματα έτσι και εδώ κάθε συνδυασμός παραμέτρων που τρέχουμε για μια προσομοίωση στο κάθε αλγόριθμο, επαναλαμβάνεται **10 φορές** και παρουσιάζονται ως τελικές μετρήσεις των αποτελεσμάτων του κάθε μεγέθους που εξετάζουμε ο **μέσος όρος** από όλες τις εκτελέσεις, παρουσιάζοντας τη γενικευμένη εικόνα των προσομοιώσεων και όχι την καλύτερη.

6.2.3 Η αναγνώριση της μορφολογίας του εδάφους από δορυφόρο “SATELLITE IMAGES”

Το τελευταίο από τα τέσσερα εξεταζόμενα προβλήματα ταξινόμησης και δεύτερο από αυτά που επιλέξαμε να επαναλάβουμε τις προσομοιώσεις, για το οποίο έχει εκτελέσει και παρουσιάσει τα αποτελέσματά του και ο ίδιος ο Huang στην επιστημονική του εργασία, είναι το πρόβλημα “SATELLITE IMAGES”.

Το Satellite Images είναι ένα πρόβλημα που αποτελείται από τα δορυφορικά δεδομένα Landsat τα οποία ανήκουν στα Συστήματα Παρατήρησης της Γης από τη NASA. Ένα καρτέ εικόνων Landsat MSS αποτελείται από τέσσερις ψηφιακές εικόνες της ίδιας σκηνής σε διαφορετικές φασματικές ζώνες. Δύο από αυτές βρίσκονται στη υπέρυθη περιοχή και δύο από αυτές στην ορατή (πράσινες και κόκκινες περιοχές). Κάθε εικόνα περιέχει 2340 x 3380 pixels, όπου το κάθε pixel αντιπροσωπεύει περιοχή 80m x 80m. Η βάση δεδομένων περιέχει μία υπό-περιοχή πολύ μικρή της συνολικής σκηνής που αποτελείται από 82 x 100 pixels. Κάθε γραμμή δεδομένων αντιστοιχεί σε μία μικρή τετραγωνική γειτονιά που 3 x 3 pixels και περιέχεται πλήρως στην υπό-περιοχή 82 x 100 pixels [33]. Περισσότερες πληροφορίες σχετικές με τη βάση δεδομένων υπάρχουν στον παρακάτω σύνδεσμο <https://archive.ics.uci.edu/dataset/146/statlog+landsat+satellite> .

Το σύνολο των στοιχείων αποτελείται από **6430 παραδείγματα – παρατηρήσεις**, όπου κάθε παρατήρηση **περιγράφεται από 36 γνωρίσματα – μετρήσεις**. Κάθε παρατήρηση μπορεί να κατηγοριοποιηθεί σε μία μόνο από τις **7 κατηγορίες – κλάσεις** οι οποίες είναι:

1. κόκκινο χρώμα
2. καλλιέργεια βαμβακιού
3. γκρίζο χρώμα
4. υγρό γκρίζο χρώμα
5. χρώμα με καλάμια βλάστησης
6. πολύ υγρό γκρίζο έδαφος
7. όλες οι κατηγορίες μαζί

Για την τελευταία κατηγορία δεν υπάρχει κάποια παρατήρηση. Συνεπώς, το πρακτικό πλήθος κατηγοριών είναι 6, άρα και περιγραφή της κάθε κατηγορίας γίνεται από 6-bit δυαδικές σειρές, με τη θέση του bit που είναι ίση με 1 να αντιπροσωπεύει τη κλάση που αντιστοιχεί η εκάστοτε παρατήρηση.

Όσον αφορά τις προσομοιώσεις, διαχωρίσαμε το σύνολο των παρατηρήσεων σε 3 διαφορετικά ζευγάρια συνόλων ανάλογα το ποσοστό τους. Αυτά είναι **50% - 50%**, **68% - 32%** και **80% - 20%** **training – testing** αντίστοιχα. Το **68% - 32%** είναι περίπου το ποσοστό που χρησιμοποίησε για να

διαχωρίζει τις παρατηρήσεις του και ο **Huang**. Τέλος, επαναλαμβάνουμε **10 φορές** τις προσομοιώσεις για το κάθε ζευγάρι παραμέτρων, παίρνοντας το μέσο όρο από τα αποτελέσματα των μετρήσεων.

6.2.4 Η βάση δεδομένων “MNIST”

Ένα από τα πιο σημαντικά προβλήματα που βοηθάνε στην εύρεση της αξιοπιστίας και της επιτυχίας των αλγορίθμων των Τεχνητών Νευρωνικών Δικτύων, είναι το πρόβλημα **MNIST**. Η **MNIST (Modified National Institute of Standards and Technology)** είναι μία βάση δεδομένων που περιέχει χειρόγραφες αναπαραστάσεις ψηφίων, που χρησιμοποιούνται για την εκπαίδευση διαφόρων συστημάτων επεξεργασίας και αναγνώρισης εικόνων [34].

Η δημιουργία της βάσης ξεκίνησε το 1994 όπου αρχικώς οι εικόνες ήταν δυαδικές 128 x 128 pixels και μεταγενέστερα δέχθηκαν επεξεργασία για να γίνουν 28 x 28 στην κλίμακα της γκρι απόχρωσης (grayscale). Στην grayscale τα pixels με 0 είναι αυτά που έχουν το λευκό χρώμα και αυτά με το 255 είναι αυτά που έχουν το μαύρο χρώμα. Σχετικά με το πλήθος των εικόνων έχουν επιλεγεί 60.000 εικόνες για εκπαίδευση και 10.000 για το πειραματικό σκέλος.

Στα δικά μας παραδείγματα εμείς δημιουργήσαμε 3 διαφορετικές κατηγορίες πειραμάτων, όπου όλες στο σκέλος της εξέτασης περιλαμβάνουν τα 10.000 διαθέσιμα δείγματα ψηφίων που υπάρχουν για το testing dataset. Αντίθετα, για το training dataset έχουν επιλέξει 3 διαφορετικές κλίμακες όπου πρώτα επιλέγουμε 10.000 δείγματα, στη συνέχεια 30.000 δείγματα και στο τέλος και τα 60.000 δείγματα για να κάνουμε τις προσομοιώσεις μας. Αυτή η διαβάθμιση επιλέχθηκε ώστε να διακρίνουμε αν μπορούν οι αλγόριθμοι να ανταπεξέλθουν με λιγότερα δείγματα από τα 60.000 για εκπαίδευση, προσφέροντας ικανοποιητικά αποτελέσματα.

Δημιουργώντας τα 3 διαφορετικά ζευγάρια συνόλων, κάναμε τις προσομοιώσεις για τις τρεις διαφορετικές παραλλαγές του BP αλλά και για τον ELM. Εξαιρώντας την παραλλαγή των Levenberg – Marquardt για την οποία δεν τρέξαμε πειράματα για 60.000 εικόνες και ταυτόχρονα χρησιμοποιήσαμε περιορισμένα δίκτυα με λίγους νευρώνες, διεξήγαμε μόνο μία προσομοίωση με τις παραμέτρους που επιλέξαμε. Για όλες τις υπόλοιπες υποπεριπτώσεις προσομοιώσεων, επαναλάβαμε το πείραμα για κάθε ξεχωριστό ζευγάρι παραμέτρων τουλάχιστον 3 φορές και κατά κύριο λόγο 5 φορές, λαμβάνοντας τον μέσο όρο των μεγεθών που εξετάζουμε, όπως προαναγγείλαμε ότι κάναμε και στο IRIS. Ο λόγος που γίνεται αυτό είναι για την καλύτερη εικόνα με πιο αμερόληπτα αποτελέσματα, αφού αν εκτελέσουμε για μία φορά κάποιο πείραμα, μπορεί να έχουμε πλασματική εικόνα των αποτελεσμάτων είτε θετική είτε αρνητική. Επειδή όμως η μέθοδος LM είναι πολύ αργή (αν και ακριβής) ειδικότερα με μεγάλα σύνολα παρατηρήσεων, γιατί ο τρόπος που εκτελεί τους υπολογισμούς με τον Jacobian πίνακα που επιλύει είναι εξαιρετικά πολλοί και περίπλοκοι, η αύξηση τόσο των παρατηρήσεων και των γνωρισμάτων αυτών σε συνδυασμό με την αύξηση και του πλήθους των νευρώνων καταναλώνουν υπερβολικό χρόνο για την περάτωση.

Επιπροσθέτως, η δομή της εκάστοτε παρατήρησης είναι ένα σύνολο από 768 γνωρίσματα κι αυτό διότι έχουμε διαμορφώσει την 28 x 28 pixels εικόνα σε ένα μονοδιάστατο πίνακα, που κάθε γραμμή έχει τοποθετηθεί στο τέλος της προηγούμενης, για να σχηματιστεί στο τέλος ο 1 X 768 πίνακας. Κάθε στοιχείο του πίνακα αντιπροσωπεύει και ένα pixel της εικόνας. Οι κατηγορίες που μπορεί να ανήκει ένα ψηφίο είναι διακριτές και είναι 10, καθώς τα ψηφία είναι από το 0 – 9. Τέλος, για την δυαδική αναπαράσταση των κατηγοριών που ανήκει η κάθε παρατήρηση δημιουργήθηκε μία έξοδος 10-bit, όπου το **0** περιγράφεται ως “**0 0 0 0 0 0 0 0 0 1**” και το **9** περιγράφεται ως “**1 0 0 0 0 0 0 0 0 0**”. Με αντίστοιχο τρόπο περιγράφονται όλα τα υπόλοιπα στοιχεία, όπου η θέση του bit που είναι ίσο με 1 αντιπροσωπεύει το προηγούμενο αριθμητικά ψηφίο. Συνεπώς, αν $\theta = \text{θέση ψηφίου που είναι ίσο } 1$, τότε $\theta - 1 =$ είναι το ψηφίο που αναγνωρίστηκε και αντιπροσωπεύει η έξοδος αυτή.

Στο **Κεφάλαιο 7** περιγράφονται αναλυτικά τα αποτελέσματα και η αποτίμηση της σύγκρισης των αλγορίθμων. Αυτό είναι το δεύτερο πρόβλημα που επιλέξαμε για να αξιολογήσουμε τους αλγορίθμους, το οποίο δε συμπεριλαμβάνεται στην επιστημονική εργασία στην οποία πρότεινε τον ELM ο Huang και οι συνεργάτες του. Οι επόμενες δύο υπό-ενότητες είναι αφιερωμένες σε δύο προβλήματα από αυτά που παρουσίασε ο Huang, τα οποία επιλέξαμε και εμείς για να συγκρίνουμε τους νευρωνικούς αλγορίθμους.

Α' Πρακτικό – Πειραματικό σκέλος – Συναρτησιακή Παλινδρόμηση

7.1 Αποτύπωση αποτελεσμάτων Συναρτησιακής Παλινδρόμησης μέσω Backpropagation

Το πρώτο σκέλος παραδειγμάτων που εξετάσαμε για να δούμε πως ανταποκρίνεται ο ELM αλλά και οι Backpropagation παραλλαγές, προτού προχωρήσουμε στην αποτίμηση, αφορά τη συναρτησιακή παλινδρόμηση. Όπως εξηγήσαμε και στην *υπό-ενότητα 6.1* η συνάρτηση που χρησιμοποιήσαμε για τα πειράματα είναι η $y = \text{sinc}(x)$. Μέσω αυτής της συνάρτησης θα παρουσιάσουμε σε αυτή την υπό-ενότητα την επιτυχία και τις δυσκολίες που παρουσιάστηκαν από τη σειρά των προβλημάτων που διεξήγαμε, αρχής γενομένης με τον **Backpropagation**.

Στα πειράματα που προχωρήσαμε τις προσομοιώσεις, οι δύο παραλλαγές που επιλέξαμε για τον Backpropagation είναι η **Levenberg – Marquardt** και η **Resilient Backpropagation** οι οποίες μέσω του "**net**" **toolbox** του **MATLAB** ορίσαμε και δημιουργήσαμε τα νευρωνικά δίκτυα μας. Οι συναρτήσεις αυτών είναι η **LM** και **Rprop** που σε συνδυασμό με το **net** αλλά και τις παραμέτρους του **πλήθους των νευρώνων** και του **ρυθμού εκμάθησης (LR)** απαρτίζουν τις διαφοροποιημένες παραμετρικές προσομοιώσεις.

Για την καλύτερη κατανόηση των αποτελεσμάτων πέρα από την εκτενής ανάλυση που θα υπάρχει σε κάθε περίπτωση, παρουσιάζονται τα πιο βασικά στοιχεία από τις μετρήσεις σε πίνακες, καθώς επίσης θα υπάρξουν και σχετικές εικόνες για την σχηματική απεικόνιση.

Το βασικό εξεταζόμενο μέγεθος είναι το **RMSE (Root Mean Square Error)** για το οποίο υπάρχει μία γενική διαβάθμιση στις τιμές του, όπου φανερώνεται πότε είναι αποδεκτό και πότε όχι και σε πιο βαθμό. Αυτό παρουσιάζεται στον ακόλουθο πίνακα, όπου τις τιμές αυτές θα τις ακολουθήσουμε για όλες τις προσομοιώσεις:

| Εξαιρετικό | Πολύ Καλό | Αποδεκτό | Μη Αποδεκτό |
|------------------|----------------------|-------------------|-------------|
| $RMSE \leq 0.75$ | $0.75 < RMSE \leq 1$ | $1 < RMSE \leq 2$ | $RMSE > 2$ |

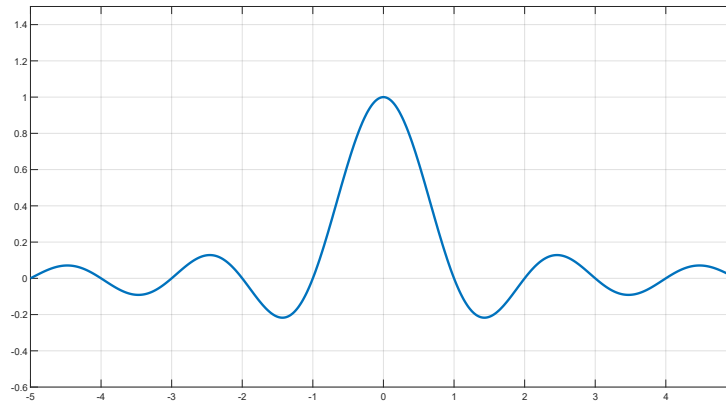
Πίνακας 5 Διακύμανση Τιμών RMSE

7.1.1 Απόδοση αποτελεσμάτων του Backpropagation με “αμόλυντα δεδομένα”

7.1.1.1 Παραλλαγή 1.1: 75% Training dataset – 25% Testing dataset

Η πρώτη παραλλαγή των συναρτησιακών παραδειγμάτων που επιχειρήσαμε αφορά καθαρά δεδομένα και στα δύο σύνολα δεδομένων (training dataset – testing dataset).

Το σημείο αφετηρίας για τις προσομοιώσεις μας είναι με ρυθμό εκμάθησης $LR = 0.3$ και για κρυφούς νευρώνες $N = 5$. Τα σύνολα τιμών για τις τιμές του x δημιουργήθηκαν με **βήμα** = **0.01**, επιλέγοντας 3 διαφορετικές διακυμάνσεις στον διαχωρισμό των δύο συνόλων εκπαίδευσης και εξέτασης, όπου ξεκινήσαμε λαμβάνοντας το **75% τυχαίων τιμών** από αυτές του συνόλου τιμών για εκπαίδευση και ακολουθήσε το **50%** και **25%**. Το σύνολο τιμών είναι $[-5, +5]$. Τα ποσοστά τυχαίων τιμών που υπολείπονται για να φτάσουμε το 100% χρησιμοποιούνται για το στάδιο της εξέτασης του δικτύου.



Εικόνα 22 Η μορφή της $y = \text{sinc}(x)$ στο διάστημα $[-5, +5]$

Στην περίπτωση που το σύστημά μας έχει **5 κρυφούς νευρώνες**, με συνάρτηση εκμάθησης την **Rprop** και με **0.3 ρυθμό εκμάθησης**, στο διάστημα $[-5, +5]$ με το **75%** των **τυχαίων τιμών** να χρησιμοποιείται για **εκπαίδευση** και το **25%** για **εξέταση**, παρά το γεγονός ότι αριθμητικά έχει μία πολύ καλή προσέγγιση στο **RMSE**, ως προς πραγματική προσέγγιση μέσω και της γραφικής παράστασης, η πρόβλεψη των αποτελεσμάτων της γραφικής παράστασης είναι διαφορετική στο σύνολό της σε σχέση με την πραγματική γραφική παράσταση. Γι' αυτό το λόγω διπλασιάσαμε το πλήθος των **κρυφών νευρώνων** σε **10** και πράγματι υπάρχει μία βελτίωση στην πρόγνωση των πραγματικών δεδομένων με το **RMSE** να παραμένει χαμηλά. Επειδή είχαμε τα καλύτερα δυνατά αποτελέσματα **αυξήσαμε** και μία δεύτερη φορά τους **κρυφούς νευρώνες** φτάνοντας στους **20**. Εδώ λοιπόν το δίκτυο εκπαιδεύεται πολύ καλύτερα και μπορεί να προβλέψει με μεγάλη ακρίβεια τα πραγματικά δεδομένα.

Ωστόσο με τη χρήση του **LM** φανερώνεται η καλύτερη γενίκευση που έχει ο αλγόριθμος από τους **5 κρυφούς νευρώνες** με αποτέλεσμα στους **10 κρυφούς νευρώνες** να μην έχει ανάγκη το δίκτυο και επιπλέον προσαρμογές. Αυτό συμβαίνει γιατί μηδένισε τα σφάλματα και η γραφική του προσέγγιση είναι αντίστοιχη με τα πραγματικά δεδομένα.

Επαναλάβαμε το ίδιο σετ δοκιμών τόσο για **ρυθμό εκμάθησης = 0.7** όσο και για **ρυθμό εκμάθησης = 1**. Τα αποτελέσματα τ' οποία πήραμε είχαν ακριβώς την ίδια φιλοσοφία. Πιο συγκεκριμένα ο αλγόριθμος **Rprop** δεν καταφέρνει να κάνει το δίκτυο να λειτουργήσει τόσο καλά και αποδοτικά όσο ο **LM**, γιατί έχει μεγαλύτερα **RMSE** σε όλο το φάσμα των δοκιμών με την αύξηση των κρυφών νευρώνων, η πρόβλεψη των δεδομένων του απέχει κατά πολύ από την πραγματικότητα αλλά και από τον **LM** και τέλος χρειάζεται μεγαλύτερο πλήθος κρυφών νευρώνων για να αποφέρει ένα ικανοποιητικό αποτέλεσμα.

Αντ' αυτού ο αλγόριθμος των **Levenberg – Marquardt**, χρειάζεται μόλις **10 νευρώνες στο κρυφό στρώμα** για να καταφέρει να τελειοποιήσει την απόδοσή του και να φτάσει σε ταύτιση με τα πραγματικά δεδομένα.

Συνοπτικά η διακύμανση των αποτελεσμάτων (**ανεξαρτήτως του LR**) για το πεδίο ορισμού $[-5, +5]$ του **RMSE** τόσο στην εκπαίδευση όσο και στην εξέταση ανάλογα το πλήθος των νευρώνων, χρησιμοποιώντας το **75%** των **τυχαίων τιμών για εκπαίδευση** και το **25%** των **υπόλοιπων για εξέταση** παρουσιάζεται στον ακόλουθο πίνακα:

| 1.1 | LR = {0.3, 0.7, 1.0} Training Dataset – Testing Dataset = {75% - 25%} Range = [-5, +5) | | | | | |
|-------|--|--------------------|--------------------|--------------------|--------------------|--------------------|
| | Neurons = 5 | | Neurons = 10 | | Neurons = 20 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.0773 – 0.0882 | 0.0799 – 0.0884 | 0.0411 – 0.0529 | 0.0386 – 0.0571 | 0.0146 - 0.0331 | 0.0136 – 0.0353 |
| LM | 0 - 0.0691 | 0 - 0.0710 | 0 | 0 | - | - |

Πίνακας 6 1.1 Αποτελέσματα BP με Καθαρά Δεδομένα

Ο χρόνος για την περάτωση της κάθε προσομοίωσης δεν ξεπέρασε το **0.5"**. Γι' αυτό το λόγω δεν τον συμπεριλαμβάνουμε στον πίνακα.

7.1.1.2 Παραλλαγή 1.2: 50% Training dataset – 50% Testing dataset

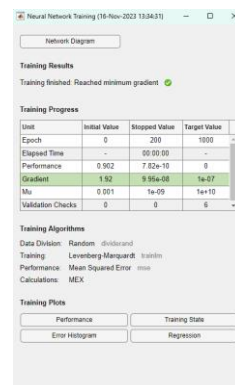
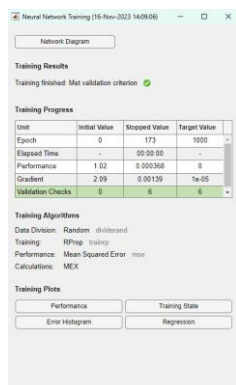
Αλλάζοντας την αναλογία των τυχαίων τιμών μέσα στα δύο σύνολα, επαναλάβαμε τα πειράματα για 50% τυχαίες τιμές στο training dataset και 50% τυχαίες τιμές στο testing dataset. Συνεπώς, τα δύο σύνολα περιέχουν τον ίδιο αριθμό τιμών.

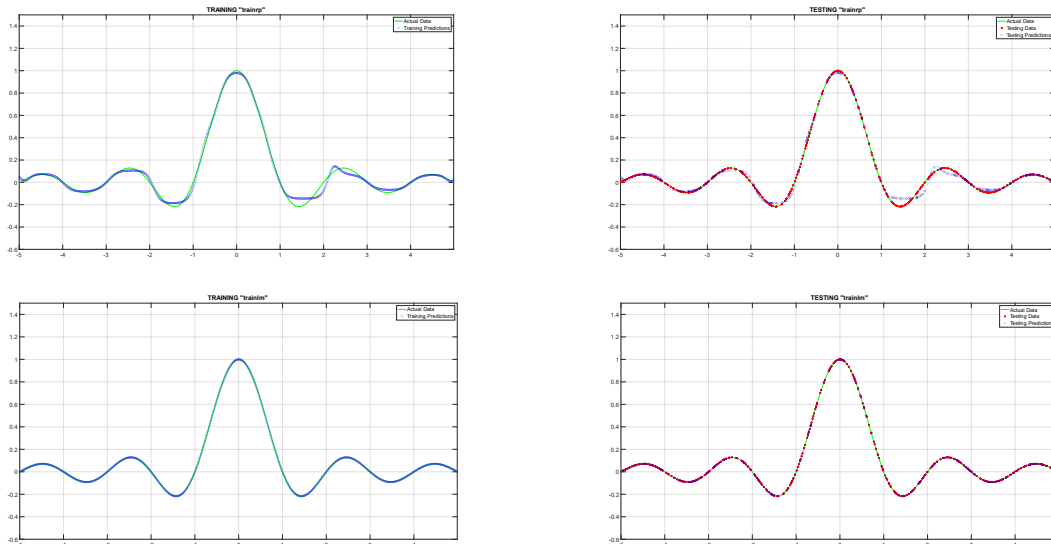
Σε όλα τα πειράματα που κάναμε πάντα ο **LM** αλγόριθμος είναι πιο αποδοτικός, έχει καλύτερη ακρίβεια αλλά ταυτόχρονα προσφέρει ένα πιο συμπαγές δίκτυο, καθώς δεν είναι αναγκαίο να χρησιμοποιήσει έναν μεγάλο όγκο από νευρώνες για να τον βοηθήσουν να αποφέρει τα κατάλληλα αποτελέσματα σε σχέση με τον **Rprop**. Πιο αναλυτικά μπορούμε να δούμε τη διακύμανση των αποτελεσμάτων του RMSE στον πίνακα που ακολουθεί:

| 1.2 | LR = {0.3, 0.7, 1.0} Training Dataset – Testing Dataset = {50% - 50%} Range = [-5, +5) | | | | | |
|-------|--|--------------------|--------------------|--------------------|--------------------|--------------------|
| | Neurons = 5 | | Neurons = 10 | | Neurons = 20 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.0737 – 0.0880 | 0.0762– 0.0898 | 0.0399 – 0.0672 | 0.0369 - 0.0699 | 0.0192 - 0.0253 | 0.0195 - 0.0249 |
| LM | 0.0142 - 0.0520 | 0.0148 - 0.0493 | 0 | 0 | - | - |

Πίνακας 7 1.2 Αποτελέσματα BP με Καθαρά Δεδομένα

Όπως και στην προηγούμενη περίπτωση έτσι και σε αυτή που είναι 50-50 τα σύνολα για τις προσομοιώσεις, τα καλύτερα δυνατά αποτελέσματα τα λάβαμε μέσω του **Levenberg – Marquardt** και για 10 κρυφούς νευρώνες. Αυτή η διαφορά μπορεί να αποτυπωθεί και στις παρακάτω εικόνες, όπου φαίνεται η βαθμιαία βελτίωση του δικτύου με την αύξηση των νευρώνων και στους δύο αλγόριθμους τους BP, αλλά μέσω Levenberg – Marquardt έχουμε καλύτερη γενίκευση. Οι εικόνες που παρουσιάζονται έχουν εξαχθεί από δοκιμές με **20 νευρώνες για Rprop** και **10 νευρώνες για LM**, για 0.7 ρυθμό εκμάθησης, όπως επίσης παρουσιάζεται και η πρόοδος του δικτύου σε κάθε περίπτωση.





Εικόνα 23 Rprop & LM με καθαρά δεδομένα

Όπως φαίνεται και στις παραπάνω εικόνες η μέθοδος του Levenberg – Marquardt γενικεύει πολύ καλύτερα και φτάνει σε μηδενικό σφάλμα για το συγκεκριμένο παράδειγμα. Μάλιστα ο χρόνος δεν ξεπέρασε το **1 δευτερόλεπτο** για την περάτωση της εκάστοτε προσομοίωσης.

7.1.1.3 Παραλλαγή 1.3: 25% Training dataset – 75% Testing dataset

Θέλοντας να διαπιστώσουμε αν το δίκτυο μπορεί να κατανοήσει τη συνάρτηση όταν εκπαιδεύεται με λιγότερα δεδομένα απ’ όσα θα εξεταστεί, αλλάξαμε για μία τελευταία φορά τα σύνολα δίνοντας το **25%** από τυχαία δεδομένα στο **training dataset** και το υπόλοιπο **75%** στο **testing dataset**. Και σε αυτή την περίπτωση ο χρόνος δεν υπερβαίνει το **1 δευτερόλεπτο** για κάθε προσομοίωση. Όπως θα παρατηρήσουμε στον ακόλουθο πίνακα δεν αντιμετωπίζει κάποιο πρόβλημα το δίκτυο, με τις προσεγγίσεις που κάνει να είναι ανάλογες όπως στα προηγούμενα δύο παραδείγματα:

| 1.3 | LR = {0.3, 0.7, 1.0} Training Dataset – Testing Dataset = {25% - 75%} Range = [-5, +5] | | | | | |
|-------|--|-----------------|-----------------|-----------------|-----------------|-----------------|
| | Neurons = 5 | | Neurons = 10 | | Neurons = 20 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.0719 - 0.0856 | 0.0717 - 0.0857 | 0.0282 - 0.0594 | 0.0283 - 0.0585 | 0.0186 - 0.0218 | 0.0189 - 0.0208 |
| LM | 0.0510 - 0.0534 | 0.0497 - 0.0546 | 0 | 0 | - | - |

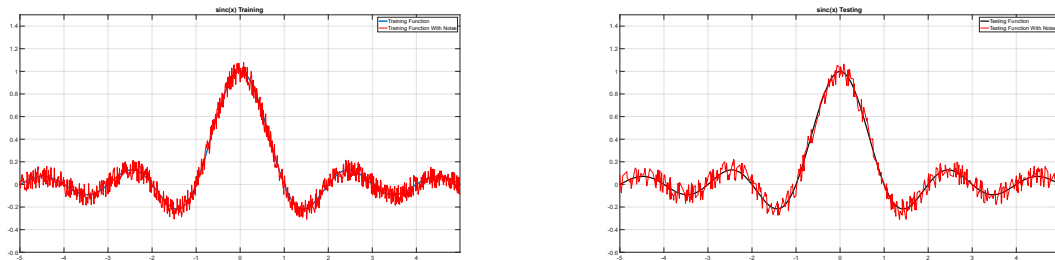
Πίνακας 8 1.3 Αποτελέσματα BP με Καθαρά Δεδομένα

7.1.2 Απόδοση αποτελεσμάτων του Backpropagation με “μολυσμένα δεδομένα”

Σε αυτή την ενότητα θα αναπτύξουμε την αποδοτικότητα του Backpropagation στις δύο παραλλαγές που ήδη έχουμε συζητήσει, ακολουθώντας τις παραπάνω τεχνικές για τις προσομοιώσεις, με τη διαφορά ότι θα προσθέσουμε θόρυβο στα δεδομένα εξόδου των συνόλων τιμών. Ο θόρυβος αυτός είναι ομοιόμορφα κατανομημένος, που σημαίνει ότι κάθε στοιχείο εξόδου τον δέχεται, είναι τυχαίος και διαλέγεται μέσα στο διάστημα $[-0.1, +0.1]$.

Θα αναπτύξουμε μία αντίστοιχη σειρά όπως και πριν από προσομοιώσεις, με το σύνολο τιμών – πεδίο ορισμού της συνάρτησης $y = \text{sinc}(x)$ να είναι το $[-5, +5]$ (Εικόνα 22) και θα χωρίσουμε στις ίδιες αναλογίες την τυχαία επιλογή των δεδομένων μας, τα οποία έχουν δημιουργηθεί με βήμα = 0.01.

Ένα δείγμα για το πως μολύνονται τα δεδομένα του συνόλου και κατ' επέκταση αλλοιώνεται η συνάρτηση φαίνεται στην ακόλουθη εικόνα, όπου με τις κόκκινες γραμμές αναπαρίστανται τα δεδομένα που έχουν μολυνθεί, ενώ οι καθαρές κυματομορφές τις $y = \text{sinc}(x)$ φανερώνονται με μπλε και μαύρο χρώνο αριστερά για το training και δεξιά για το testing αντίστοιχα.



Εικόνα 24 Σύνολα τιμών με μολυσμένα δεδομένα

7.1.2.1 Παραλλαγή 2.1: 75% Training dataset – 25% Testing dataset

Εκκινώντας για **5 κρυφούς νευρώνες** και για **ρυθμό εκμάθησης = 0.3**, το δίκτυό μας έχει καλή πρόβλεψη για τα δεδομένα τα οποία βρίσκονται κοντά στην αρχή των αξόνων μέσω του **Rprop**. Για τα υπόλοιπα δεδομένα δεν έχει κάνει καμιά πρόβλεψη και αδυνατεί να τα προσεγγίσει. Το **RMSE πάντα είναι χαμηλό και με την αύξηση των νευρώνων βελτιώνεται**. Συνεπώς, από τις τιμές του τόσο στο training όσο και στο testing, δε μπορούμε να βγάλουμε κάποιο ασφαλές συμπέρασμα. Το κριτήριό μας θα πρέπει να είναι οι γραφικές παραστάσεις που προκύπτουν από τα σημεία των προβλέψεών του.

Με την αύξηση στους **15 κρυφούς νευρώνες**, σύστημα απέκτησε μία καλύτερη προσεγγιστική ικανότητα, αλλά δεν καταφέρνει σε καλό βαθμό να είναι αρμονικό το αποτέλεσμα. Γι' αυτό το σκοπό προχωρήσαμε στην περαιτέρω αύξηση των **νευρώνων** σε **30**. Εδώ το δίκτυο έχει την καλύτερη δυνατή προσέγγιση από τα δεδομένα που έχει για την εκμάθησή του και προσεγγίζει σε καλύτερο βαθμό και την περιοδικότητα της συνάρτησης με το σφάλμα ελαχιστοποιείται.

Σε αντίθεση με τον Rprop, ο **LM** έχει καλύτερη απόδοση σε κάθε ξεχωριστή βαθμίδα για το πλήθος των κρυφών νευρώνων. Είναι χαρακτηριστικό πως με τους **15 κρυφούς νευρώνες** το δίκτυο αποκτάει πολύ καλή προβλεπτική ικανότητα, χωρίς να χρειάζεται να προσθέσουμε περισσότερους νευρώνες.

Μάλιστα επαναλάβαμε και για τους δύο αλγόριθμους όλες τις ξεχωριστές προσομοιώσεις τόσο για **ρυθμό εκμάθησης = 0.7** όσο και για **ρυθμό εκμάθησης = 1**. Οι διαφορές στα αποτελέσματα είναι σχεδόν μηδενικές, δείγμα του ότι δε χρειάζεται ιδιαίτερη προσπάθεια στους πειραματισμούς μας για τις παραμέτρους, καθώς το πρόβλημα δεν είναι πολύπλοκο.

Στον παρακάτω πίνακα συγκεντρώνονται όλες οι μετρήσεις του RMSE που υποδεικνύει που έχει την καλύτερη προσέγγιση και με ποιες παραμέτρους. Και εδώ βέβαια είναι χαρακτηριστικό πως χρονικά είναι πολύ γρήγορες οι προσομοιώσεις, καθώς ένα τέτοιου είδους πρόβλημα δεν καταναλώνει χρόνο για το δίκτυο γιατί δεν είναι πολύπλοκο και ογκώδες. **Ο χρόνος δεν ξεπερνάει το 0.5 του δευτερολέπτου.**

| 2.1 | LR = {0.3, 0.7, 1.0} Training Dataset – Testing Dataset = {25% - 75%} Range = [-5, +5] | | | | | |
|-------|--|--------------------|--------------------|--------------------|--------------------|--------------------|
| | Neurons = 5 | | Neurons = 15 | | Neurons = 30 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.0952 | 0.0895 | 0.0656 - 0.0685 | 0.0637 - 0.0710 | 0.0610 - 0.0705 | 0.0577 - 0.0700 |
| LM | 0.0760 - 0.0970 | 0.0760 - 0.0970 | 0.0558 - 0.0571 | 0.0567 | - | - |

Πίνακας 9 2.1 Αποτελέσματα BP με Μολυσμένα Δεδομένα

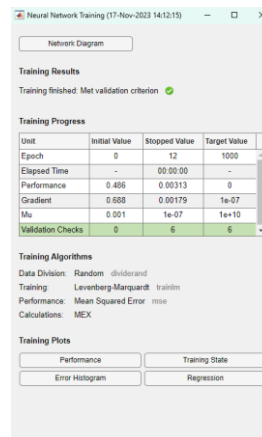
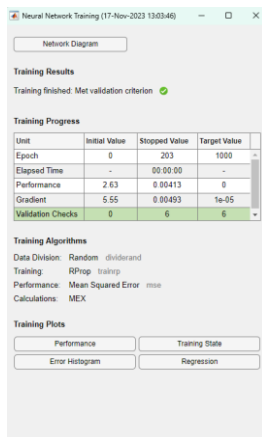
7.1.2.2 Παραλλαγή 2.1: 50% Training dataset – 50% Testing dataset

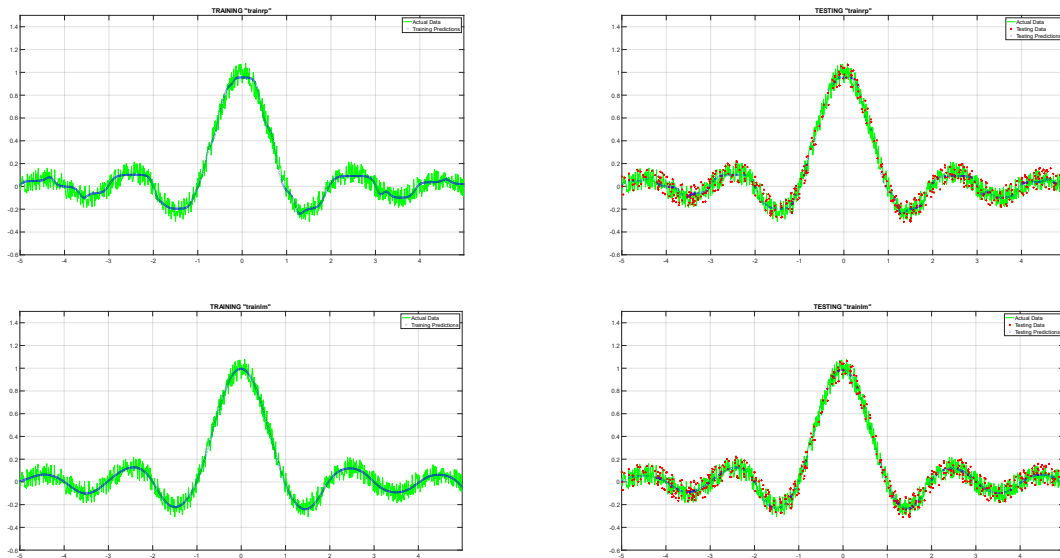
Όπως και προηγουμένως στην υπό-ενότητα 7.1.2.1, επαναλάβαμε τα ίδια ακριβώς πειράματα για το 50% αλλά και για 25% ποσοστό τυχαίων δεδομένων για το σύνολο δεδομένων της εκπαίδευσης σε σχέση με το πεδίο ορισμού, με το 50% και 75% αντίστοιχα που υπολείπεται για το 100% να αποτελεί το πλήθος τυχαίων δεδομένων των συνόλων εξέτασης. Η εξέλιξη στην αναβάθμιση του δικτύου και στην απόδοσή του είναι σε παρόμοια επίπεδα. Όπως θα φανεί και στον Πίνακα 10 και στον Πίνακα 11 που παρουσιάζονται οι τιμές του RMSE συγκεντρωτικά, η αύξηση των νευρώνων δίνει καλύτερες προσεγγίσεις στο δίκτυο. Ταυτόχρονα, ο ρυθμός εκμάθησης δεν παίζει κάποιον ρόλο καθώς μιλάμε για σχεδόν ίδιες προβλέψεις, όπου οι απειροελάχιστες διαφορές είναι αναμενόμενο να συμβούν, αφού πρόκειται για ένα νευρωνικό δίκτυο που σημαίνει ότι η απόδοσή του είναι δυναμική. Και στις δύο περιπτώσεις ο Levenberg – Marquardt είναι πιο αποδοτικός από τον Resilient Backpropagation. Τα αποτελέσματα για τα ισόποσα σύνολα δεδομένων (50% - 50%) είναι τα ακόλουθα:

| 2.2 | LR = {0.3, 0.7, 1.0} Training Dataset – Testing Dataset = {50% - 50%} Range = [-5, +5] | | | | | |
|-------|--|--------------------|--------------------|--------------------|--------------------|--------------|
| | Neurons = 5 | | Neurons = 15 | | Neurons = 30 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.0915 - 0.1033 | 0.0927 - 0.1071 | 0.0639 - 0.0752 | 0.0654 - 0.0797 | 0.0636 - 0.0655 | 0.0663 |
| LM | 0.0599 - 0.0777 | 0.0621- 0.0816 | 0.0565 - 0.0586 | 0.0599 - 0.0611 | - | - |

Πίνακας 10 2.2 Αποτελέσματα BP με Μολυσμένα Δεδομένα

Στις ακόλουθες εικόνες στην αριστερή στείλει παρουσιάζονται η πρόοδος του δικτύου μέσω του Resilient Backpropagation, ενώ στη δεξιά μέσω του Levenberg – Marquardt. Η πρώτη γραμμή αφορά την πρόοδο του δικτύου, η δεύτερη το στάδιο εκπαίδευσης και η τρίτη την εξέταση σε νέα δεδομένα.





Εικόνα 25 Rprop & LM με μολυσμένα δεδομένα

7.1.2.3 Παραλλαγή 2.3: 25% Training dataset – 75% Testing dataset

Σε αυτή την υπό-ενότητα παρουσιάζεται και ο πίνακας που συγκεντρώνει τ' αποτελέσματα βασισμένα στο **RMSE**, από τις προσομοιώσεις με **25% - 75%** σύνολα τιμών βάση του αρχικού πεδίου ορισμού για εκπαίδευση και εξέταση αντίστοιχα. Ο χρόνος για να περατώσουν μία προσομοίωση τα δίκτυα δεν ξεπέρασε τα 2 δευτερόλεπτα.

| 2.3 | LR = {0.3, 0.7, 1.0} Training Dataset – Testing Dataset = {50% - 50%} Range = [-5, +5] | | | | | |
|--------------|--|-----------------|-----------------|-----------------|-----------------|-----------------|
| | Neurons = 5 | | Neurons = 15 | | Neurons = 30 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.1035 | 0.0945 - 0.1048 | 0.0598 - 0.0728 | 0.0585 - 0.0735 | 0.0645 - 0.0683 | 0.0631 - 0.0691 |
| LM | 0.0661 - 0.0989 | 0.0661 - 0.1018 | 0.0577 | 0.0576 | - | - |

Πίνακας 11 2.3 Αποτελέσματα BP με Μολυσμένα Δεδομένα

Οι προσομοιώσεις εδώ του **Rprop** δεν είναι πολύ καλές και αποδοτικές, καθώς φαίνεται ότι δυσκολεύεται να προσεγγίσει τα νέα δεδομένα χωρίς την κατάλληλη εκπαίδευση. Για την κατάλληλη εκπαίδευση επομένως, έχει ανάγκη από ένα περιεκτικό σύνολο δεδομένων στην εκπαίδευση του δικτύου, για να μπορέσει να μας εξασφαλίσει καλές προσομοιώσεις. Κάτι τέτοιο δε μπορεί να το καταφέρει έχοντας μόνο το 25% των διαθέσιμων δεδομένων στο training dataset. Αντίθετα ο **LM** δείχνει ότι και με αυτά τα λιγότερα δεδομένα, μπορεί να κατανοήσει τα μοτίβα, προσφέροντας στο δίκτυο ορθή γενίκευση.

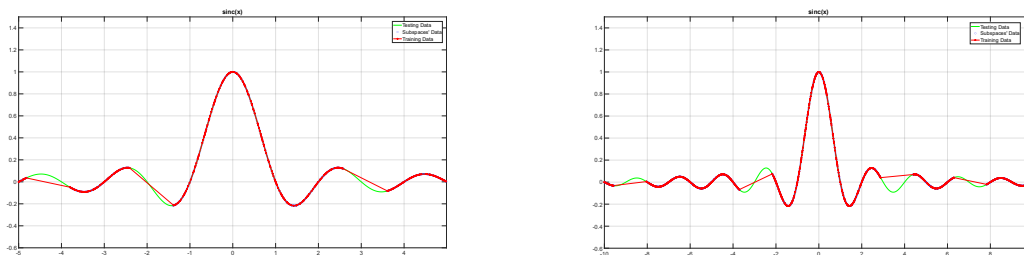
7.1.3 Αποτελέσματα του Backpropagation με “αποκοπή διαστημάτων”

Για να εξετάσουμε αν το νευρωνικό δίκτυο όταν είναι εκπαιδευμένο με Backpropagation μπορεί ν' ανταποκριθεί και σε πιο δύσκολες συνθήκες, αποφασίσαμε να δοκιμάσουμε κάποια πειράματα με αφαίρεση υποδιαστημάτων, από το αρχικό διάστημα ορισμού των σημείων, δίνοντας αυτό το ελλειπές διάστημα που απομένει ως σύνολο εκπαίδευσης στο δίκτυο. Για τον σκοπό αυτόν σε πρώτη φάση ορίσαμε ένα αρχικό διάστημα ορισμού [-5, +5) για τιμές του x όπως και στα προηγούμενα

παραδείγματα, με το βήμα των τιμών του διανύσματος να είναι $\beta\eta\mu\alpha = 0.1$. Σε αυτό το διάστημα αφαιρέσαμε 3 μη-διαδοχικά διαστήματα μήκους 1 μονάδες το καθένα. Συνεπώς, στο νέο μη-συνεχές υποδιάστημα υπάρχουν 700 από τις 1000 τιμές του x , σε σχέση με το αρχικό σύνολο τιμών. Αρχικά επιλέξαμε να δώσουμε στο δίκτυο και τις 700 τιμές και στη συνέχεια 350 τυχαίες τιμές από τις 700. Αυτές οι τιμές αποτελούν τα δεδομένα του συνόλου εκπαίδευσης ενώ το αρχικό σύνολο τιμών αποτελεί το σύνολο δεδομένων της εξέτασης. Με αυτό τον τρόπο θέλουμε να δούμε αν στα πεδία τα οποία έχουμε αφαιρέσει και δεν έχει πάρει από αυτά τιμές για να εκπαιδευτεί το δίκτυο, μπορεί να ανταπεξέλθει σε τυχαία δεδομένα που θα το προσφερθούν και να προσεγγίσει τα αντίστοιχα y της συνάρτησης $y = \text{sinc}(x)$. Αυτή τη μεθοδολογία θα την ακολουθήσουμε για όλο το σύνολο πειραμάτων με αποκοπή διαστημάτων για όλους τους αλγορίθμους.

Στην **Ενότητα 6.1** έχουμε αναπτύξει ως ένα βαθμό τη σχετική φιλοσοφία γι' αυτού του είδους το πρόβλημα παλινδρόμησης που εφαρμόζουμε στις εξετάσεις των αλγορίθμων και των δικτύων.

Στην εικόνα που ακολουθεί φαίνονται δύο απεικονίσεις από τη δημιουργία των συνόλων (αριστερά με 3 και δεξιά με 4 υποδιαστήματα να λείπουν), όπου με την πράσινη γραμμή απεικονίζεται το σύνολο των στοιχείων που θα εξεταστεί το δίκτυο, με τα μπλε κυκλάκια παρουσιάζονται όλες τα δυνατά δεδομένα που απομένουν μετά την αποκοπή των υπό-διαστημάτων, τα οποία ταυτίζονται με αυτά της πράσινης γραμμής, ενώ τέλος με την κόκκινη γραμμή με αστεράκια αφορούν τα τυχαία δεδομένα που θα εκπαιδεύσουμε το δίκτυο:



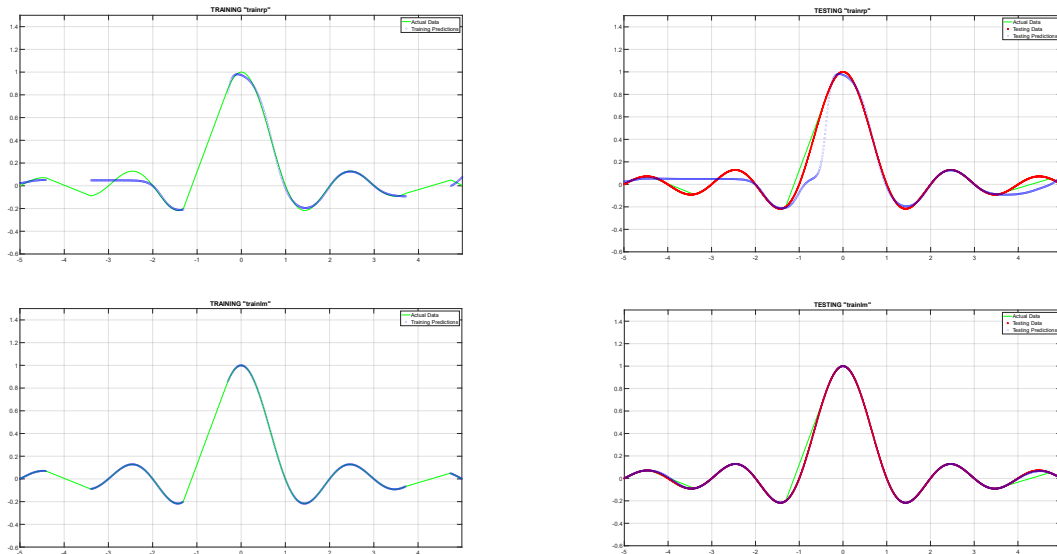
Εικόνα 26 Σύνολα δεδομένων με αποκομμένα διαστήματα εκπαίδευσης

7.1.3.1 Παραλλαγή 3.1: 700 / 700 τιμές του πεδίου ορισμού [-5, +5)

Έχοντας τα παραπάνω ως γνώση τρέξαμε για ρυθμός εκμάθησης 0.2, για ένα σύνολο 10 κρυφών νευρώνων αρχικά και περισσότερων στην συνέχεια τόσο τον Resilient Backpropagation όσο και τον Levenberg – Marquardt. Όπως θα δούμε ο Resilient δε μπορεί να ανταποκριθεί σε αυτού του είδους το πρόβλημα σε όλες τις εκφάνσεις του και για όλες τις παραλλαγές των πειραμάτων, ενώ ο Levenberg – Marquardt παρουσιάζει μια πολύ καλή γενίκευση σχεδόν σε όλες τις περιπτώσεις.

| 3.1 | LR = {0.2} Training Dataset = {700 / 700} Range = [-5, +5) | | | | | |
|-------|--|--------------|---------------|--------------|---------------|--------------|
| | Neurons = 10 | | Neurons = 20 | | Neurons = 40 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.0511 | 0.0930 | - | - | 0.0213 | 0.2021 |
| LM | 0 | 0.0009 | 0.0001 | 0.0127 | - | - |

Πίνακας 12 3.1 Αποτελέσματα BP με Αποκοπή Υποσυνόλων



Εικόνα 27 Παραλλαγή 3.1: Rprop - LM με αποκομμένα διαστήματα εκπαίδευσης

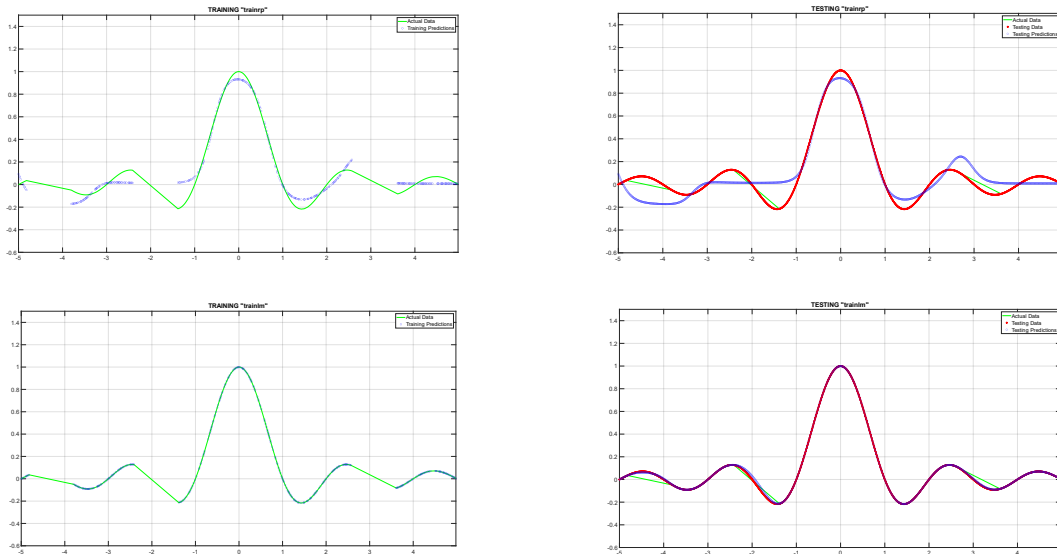
7.1.3.2 Παραλλαγή 3.2: 350 / 700 τιμές του πεδίου ορισμού [-5, +5)

Σε αυτή την παραλλαγή το **μόνο** που αλλάξαμε σε σχέση με την προηγούμενη είναι ότι μειώσαμε τα διαθέσιμα στοιχεία που μπορούσε να πάρει το δίκτυο για να εκπαιδευτεί σε **350 / 700**. Στις εικόνες παρατηρούμε πως δεν αλλάζει η αποτελεσματικότητα στη σύγκριση των δύο αλγορίθμων. Επίσης, να αναφέρουμε ότι δε μπορούμε να αυξήσουμε κατά πολύ τους νευρώνες του δικτύου, αναμένοντας καλύτερα αποτελέσματα, γιατί για περισσότερους από 20 κρυφούς νευρώνες το δίκτυο παρουσιάζει **overfitting** και αντί να συγκλίνει και να βελτιώνεται αποκλίνει.

| 3.2 | LR = {0.2} Training Dataset = {350 / 700} Range = [-5, +5) | | | | | |
|-------|--|--------------|---------------|--------------|---------------|--------------|
| | Neurons = 10 | | Neurons = 20 | | Neurons = 40 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.0671 | 0.0861 | - | - | 0.0252 | 0.1122 |
| LM | 0 | 0.0007 | - | - | - | - |

Πίνακας 13 3.2 Αποτελέσματα BP με Αποκοπή Υποσυνόλων

Ενώ οι τιμές των RMSE είναι σε πολύ ικανοποιητικό επίπεδο, οι γραφικές παραστάσεις δεν λένε το ίδιο πράγμα και όπως θα δούμε στις επόμενες εικόνες που παρουσιάζονται για **ρυθμό εκμάθησης = 0.2** ο LM καταφέρνει να προσεγγίσει όλη τη συνάρτηση ακόμα και στα σημεία που δεν έχει εκπαιδευτεί (**δεύτερη σειρά**), ενώ ο Rprop δεν προσεγγίζει τη συνάρτηση σωστά (**πρώτη σειρά**). Στην πρώτη στήλη παρουσιάζονται τα αποτελέσματα εκπαίδευσης ενώ στη δεύτερη τα αποτελέσματα από τις εξετάσεις των προσομοιώσεων:



Εικόνα 28 Παραλλαγή 3.2: Rprop - LM με αποκομμένα διαστήματα εκπαίδευσης

Είναι εμφανής η διαφορά που έχει προκύψει στις προβλέψεις των δύο διαφορετικών παραλλαγών του Backpropagation, με τον LM να είναι πολύ πιο αποδοτικός σε σχέση με τον RP. Χρονικά δεν υπήρχε κάποιο αξιοσημείωτο συμβάν για να σχολιάσουμε, καθώς και εδώ όπως σε όλα τα προηγούμενα αλλά και στα επόμενα πειράματα συναρτησιακής παλινδρόμησης, ο χρόνος για την περάτωση της κάθε προσομοίωσης είναι της τάξεως **0.5 – 3 δευτερολέπτων**.

7.1.3.3 Παραλλαγή 3.3: 1400 / 1400 τιμές του πεδίου ορισμού [-10, +10)

Σε αυτή και την επόμενη παραλλαγή θα παρουσιάσουμε τα αποτελέσματα των αλγορίθμων, αλλά με αυξημένο διάστημα τιμών. Για το σκοπό αυτό το νέο διάστημα είναι πλέον **[-10, +10)** κάτι το οποίο είχε επίπτωση και στην αύξηση των υποδιαστημάτων που αφαιρούμε σε **4** αλλά και το μήκος του εκάστοτε υποδιαστήματος σε **1.5 μονάδα**. Στα πειράματα τα οποία έγιναν χρησιμοποιήθηκαν **ρυθμοί εκμάθησης 0.2 και 0.8** και **πλήθος νευρώνων ίσο με 10, 30 και 40** ανάλογα την περίπτωση. Παρόλα αυτά θα παρουσιάσουμε τις καλύτερες δυνατές προσομοιώσεις, οπότε θα αρκεστούμε στις προσομοιώσεις με ρυθμό εκμάθησης **0.2**.

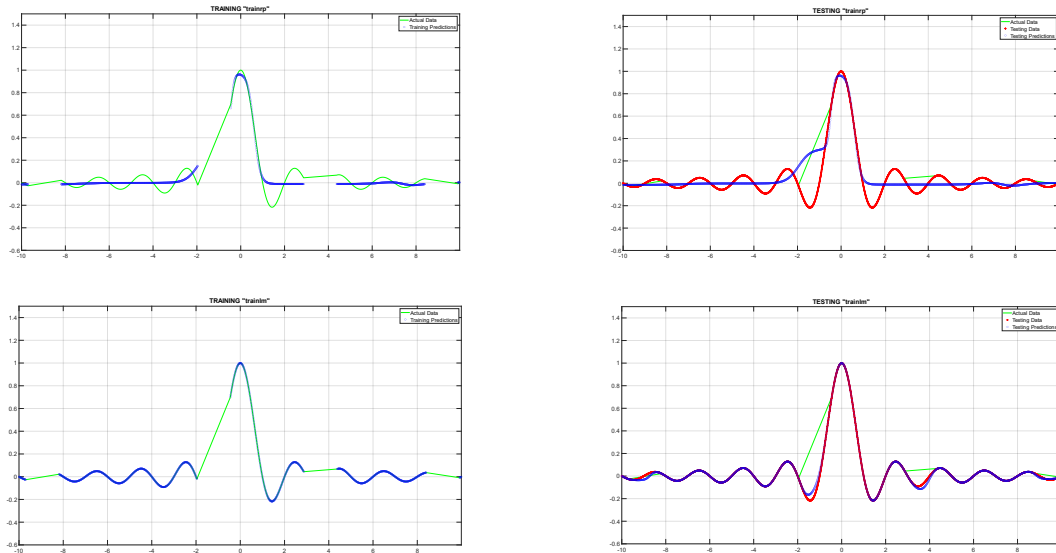
Διεξάγοντας αντίστοιχες προσομοιώσεις όπως και στις προηγούμενες υποενότητες, με τη διαφοροποίηση να έγκειται στο πεδίο ορισμού σύνολο αφού πλέον είναι το [-10, +10) λαμβάνουμε τα 1400 διαθέσιμα ζευγάρια τιμών για την $y = \text{sinc}(x)$, που έχουν απομείνει μετά την αποκοπή των υποδιαστημάτων.

Πιο συγκεκριμένα ο Levenberg – Marquardt είναι πολύ αποδοτικός για το δίκτυο και καταφέρνει να το εκπαιδεύσει σε αντίθεση με τον Resilient Backpropagation που δεν τα καταφέρνει στον ίδιο βαθμό. Από πλευράς **RMSE** είναι αποδεκτές σε εξαιρετικό βαθμό όλες οι προσομοιώσεις, αλλά αυτό δεν αντικατοπτρίζεται στις προβλέψεις των γραφικών παραστάσεων όπως θα δούμε στις εικόνες που ακολουθούν.

Ξεκινώντας με τον πίνακα των αποτελεσμάτων έχουμε τα εξής:

| 3.3 | LR = {0.2} Training Dataset = {1400 / 1400} Range = [-10, +10) | | | | | |
|-------|--|--------------|---------------|--------------|---------------|--------------|
| | Neurons = 10 | | Neurons = 30 | | Neurons = 40 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.0315 | 0.0741 | - | - | 0.0154 | 0.1349 |
| LM | 0.0274 | 0.0410 | 0 | 0.0129 | - | - |

Πίνακας 14 3.3 Αποτελέσματα BP με Αποκοπή Υποσυνόλων



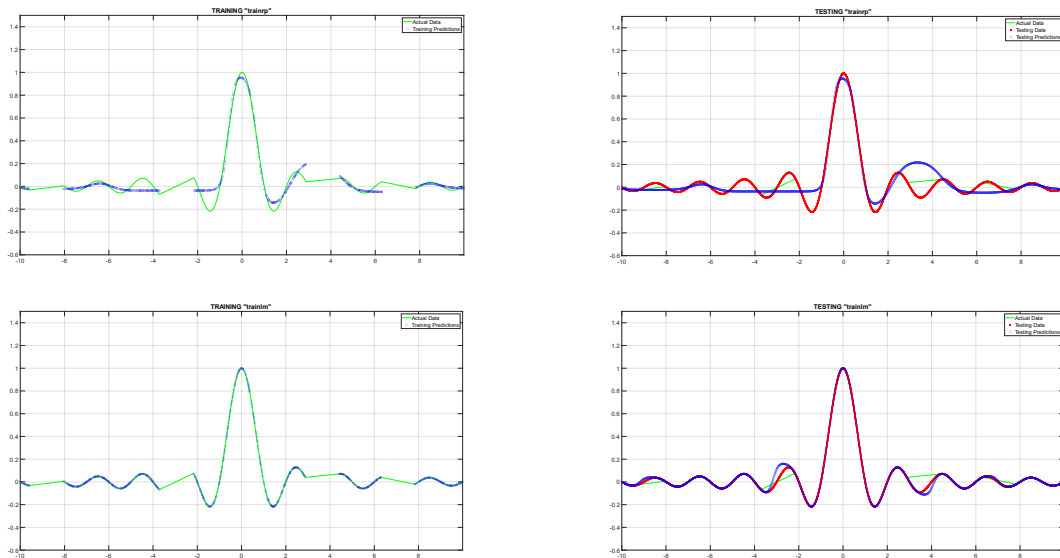
Εικόνα 29 Παραλλαγή 3.3: Rprop - LM με αποκομμένα διαστήματα εκπαίδευσης

7.1.3.4 Παραλλαγή 3.4: 700 / 1400 τιμές του πεδίου ορισμού [-10, +10)

Προσφέροντας στο σύνολο δεδομένων εκπαίδευσης πλέον τα 700 από τα 1400 διαθέσιμα δεδομένα από το μη-συνεχές διάστημα που προέκυψε από τις αφαιρέσεις των υποδιαστημάτων, η υπεροχή του Levenberg – Marquardt συνεχίζει να υφίσταται, με τον Resilient να μην καταφέρνει να κάνει καλές προσεγγίσεις, δείγμα του ότι δεν είναι ο κατάλληλος αλγόριθμος γι' αυτού του είδους το πρόβλημα. Αρχικά παρουσιάζονται στον ακόλουθο πίνακα τα αποτελέσματα στα RMSE των προσομοιώσεων και στη συνέχεια οι αντίστοιχες γραφικές αναπαραστάσεις από τις προβλέψεις των αλγορίθμων. Μέσα από αυτές γίνεται πιο ξεκάθαρη η δυνατότητα του ενός αλγορίθμου σε σχέση με τον άλλο.

| 3.4 | LR = {0.2} Training Dataset = {700 / 1400} Range = [-10, +10) | | | | | |
|-------|---|--------------|---------------|--------------|---------------|--------------|
| | Neurons = 10 | | Neurons = 30 | | Neurons = 40 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| Rprop | 0.0608 | 0.0801 | - | - | 0.0172 | 0.1114 |
| LM | 0.0258 | 0.0537 | 0 | 0.0219 | - | - |

Πίνακας 15 3.4 Αποτελέσματα BP με Αποκοπή Υποσυνόλων



Εικόνα 30 Παραλλαγή 3.3: Rprop - LM με αποκομμένα διαστήματα εκπαίδευσης

Όπως φαίνεται στα παραπάνω σχήματα ο Levenberg – Marquardt καταφέρνει να προσεγγίσει τη συνάρτηση ενώ ο Resilient Backpropagation δεν τα καταφέρνει, κι αυτό δε μπορεί να διορθωθεί με την αύξηση των νευρώνων καθώς “πέφτει” σε κατάσταση **overfitting**.

7.1.4 Αποτίμηση του LM με “μολυσμένα δεδομένα” και “αποκοπή διαστημάτων”

Μία τελευταία παραλλαγή του προβλήματος με την αποκοπή διαστημάτων, είναι αυτή στην οποία προσθέτουμε και θόρυβο στα δεδομένα εξόδου. Με αυτό τον τρόπο προσπαθούμε να πείσουμε το δίκτυο που εκπαιδεύεται με τον Backpropagation να κατανοήσει τα μοτίβα σε δύσκολες συνθήκες. Όμως σε αυτό το στάδιο δε θα εξετάσουμε τον Resilient Backpropagation καθώς ήδη από τις προηγούμενες παραλλαγές αποκοπής διαστημάτων δε μπορούσα να ανταποκριθεί σωστά. Συνεπώς, ελέγχουμε μόνο τον Levenberg – Marquardt.

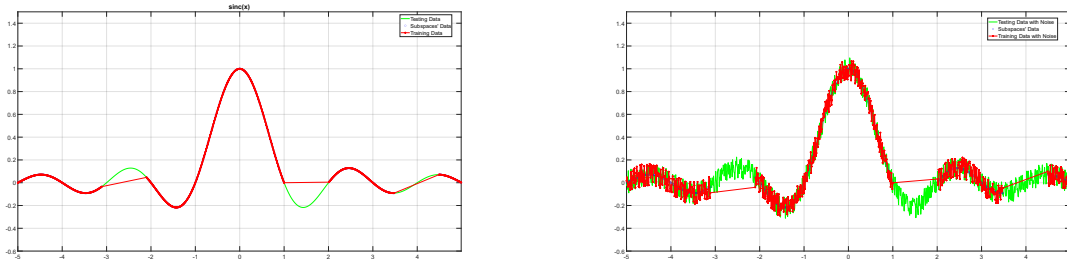
Ο Levenberg – Marquardt καταφέρνει να αποδώσει στο δίκτυο την εκπαίδευση που πρέπει, αλλά αυτό μπορεί να το κάνει σε ένα πεδίο ορισμού $[-5, +5]$ με βήμα = **0.01**. Σε μεγαλύτερα πεδία ορισμού δε μπορεί να ανταποκριθεί και αποκλίνει και αυτός. Από αυτό το πεδίο ορισμού αφαιρούνται **3 υποδιαστήματα τυχαία που το καθένα έχει μήκος 1 μονάδας**. Ταυτόχρονα λαμβάνονται για την εκπαίδευση του δικτύου όλα τα δεδομένα από το νέο μη-συνεχές διάστημα τιμών τα οποία είναι **700**.

Στις αμέσως δύο επόμενες εικόνες παρουσιάζονται αριστερά τα δεδομένα καθαρά, όπως έχουμε δείξει και σε προηγούμενες περιπτώσεις, ενώ δεξιά παρουσιάζονται τα πραγματικά δεδομένα που θα διαχειριστεί το δίκτυο, για τα οποία έχουν υποστεί μόλυνση με έναν ομοιόμορφα κατανομημένο τυχαίο θόρυβο μέσα από το διάστημα $[-0.1, +0.1]$.

| 4.1 | LR = {0.2} Training Dataset = {700 / 700} Range = [-5, +5] | | | | | |
|-----|--|--------------|---------------|--------------|---------------|--------------|
| | Neurons = 10 | | Neurons = 30 | | Neurons = 40 | |
| | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| LM | 0.0604 | 0.0723 | 0.0588 | 0.1187 | 0.0576 | 0.3288 |

Πίνακας 16.4.1 Αποτελέσματα BP με Αποκοπή Υποσυνόλων σε Μολυσμένα δεδομένα

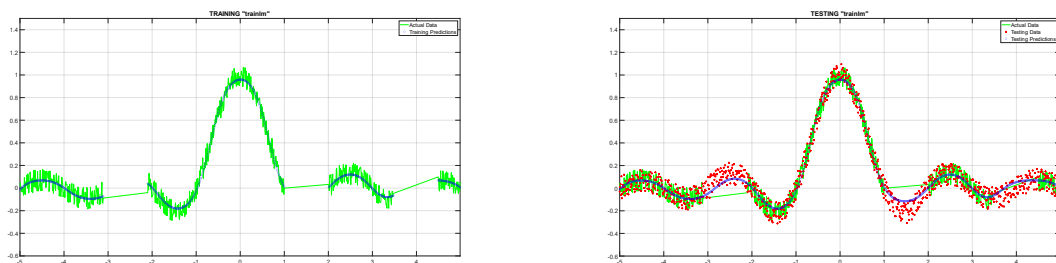
Από τ' αποτελέσματα του πίνακα με την αύξηση των νευρώνων προκύπτει κλιμάκωση σε **overfitting** και ενώ στους 10 νευρώνες ανταποκρίνεται πολύ καλά το δίκτυο με την εκπαίδευση του LM, με περισσότερους νευρώνες το δίκτυο όχι μόνο δε βελτιώνει την απόδοσή του αλλά αποκλίνει κιάλας.



Εικόνα 31 Σύνολα τιμών με μολυσμένα δεδομένα και αποκομμένα διαστήματα

Ο αλγόριθμος και εδώ μπορεί να βοηθήσει το δίκτυο να εκτελέσει τις προσομοιώσεις πολύ γρήγορα, γι' αυτό το λόγο δε γίνεται και κάποια νύξη σε αυτό.

Τα αποτελέσματα του δικτύου που εκπαιδεύεται με τα μολυσμένα δεδομένα (αριστερά) αλλά και οι προβλέψεις του στην εξέταση (δεξιά) παρουσιάζονται στις επόμενες δύο εικόνες.



Εικόνα 32 Αποτελέσματα LM με μολυσμένα δεδομένα και αποκοπή διαστημάτων

Εδώ τελειώσαμε με όλα τα δυνατά πειράματα που χρειαζόταν να κάνουμε για να δούμε και την απόδοση του Backpropagation, σε ένα πρόβλημα συναρτησιακό παλινδρόμησης. Όπως διαπιστώσαμε ο Backpropagation είναι ένας πολύ καλός και αξιόπιστος αλγόριθμος για τα νευρωνικά δίκτυα σε συναρτησιακές εφαρμογές, ο οποίο αν τροφοδοτηθεί με τα σωστά δεδομένα και σε συνθήκες που είναι αποδεκτές, μπορεί να προσφέρει σωστές προσομοιώσεις. Είδαμε την παραλλαγή μέσω Levenberg – Marquardt να αποδίδει πολύ καλύτερα σε σχέση με τον Resilient Backpropagation αλλά και με μεγαλύτερη ανοχή στις αντίξοες συνθήκες.

Στην επόμενη ενότητα θα προχωρήσουμε και με τις δοκιμές για το ίδιο πρόβλημα συναρτησιακής παλινδρόμησης και για τον ELM αλγόριθμο, προτού προχωρήσουμε στη τελική αποτίμηση της συναρτησιακής εφαρμογής του ELM και Backpropagation.

7.2 Αποτύπωση αποτελεσμάτων Συναρτησιακής Παλινδρόμησης μέσω ELM

Σε αυτή την ενότητα θα επαναλάβουμε όλες τις παραλλαγές του προβλήματος της συναρτησιακής παλινδρόμησης, τις οποίες εκτελέσαμε στην **Ενότητα 7.1**, αλλά πλέον ο εξεταζόμενος νευρωνικός αλγόριθμος θα είναι ο **ELM**. Βασικός σκοπός είναι η αποτύπωση των γραφικών παραστάσεων την $y = \text{sin}(x)$ για την οποία και ο ίδιος ο Huang έτρεξε ένα σύνολο από πειράματα για να συγκρίνει με τον BP τα αποτελέσματά τους, ώστε να διαπιστώσει αν αυτός ο αλγόριθμος μπορεί να προσφέρει στο σύστημα καλύτερη προβλεπτική ικανότητα.

Εξ' αρχής να επισημάνουμε πως ο χρόνος για τη διεκπεραίωση της κάθε προσομοίωσης είναι εξαιρετικά μικρός, και παρόλο ότι οι προσομοιώσεις με τον BP ήταν πολύ γρήγορες εδώ υπάρχουν ακόμη πιο άμεσες προσομοιώσεις. Παρόλα αυτά εμείς θα επικεντρωθούμε και πάλι στο RMSE και στις γραφικές απεικονίσεις των προβλέψεων σε training και testing.

7.2.1 Αποτελεσματικότητα του ELM με “αμόλυντα” συναρτησιακά δεδομένα

Πρώτη απόπειρα λοιπόν είναι με αμόλυντα δεδομένα εξόδου τόσο για το training dataset όσο και για το testing dataset. Αρχικά το πεδίο ορισμού που ορίζουμε για να επιλέξουμε τις τιμές του x είναι το $[-5, +5]$ για το οποίο όπως πάντα το **βήμα** = **0.01**. Για τη δημιουργία των συνόλων διαχωρίσαμε με τυχαία επιλογή τα στοιχεία του πεδίου ορισμού και δημιουργήσαμε τρεις ξεχωριστές αναλογίες ποσοστών οι οποίες είναι **75% - 25%**, **50% - 50%** και **25% - 75%** για **training dataset – testing dataset**. Για όλες τις περιπτώσεις χρησιμοποιούμε τρεις διαφορετικές κλίμακες κρυφών νευρώνων οι οποίες είναι **2 – 10 – 20** όπως και τρεις διαφορετικές συναρτήσεις ενεργοποίησης οι οποίες είναι **sigmoid – cosine – tangent**. Στους πίνακες που θα παρουσιάσουμε θα συμπεριλάβουμε τις καλύτερες προσομοιώσεις για κάθε κλίμακα κρυφών νευρώνων, ώστε να έχουμε πιο συμπυκνωμένα τα δεδομένα και τη κλιμακωτή διαβάθμιση των αποτελεσμάτων.

7.2.1.1 Παραλλαγή 1.1: Καθαρά δεδομένα του πεδίου ορισμού $[-5, +5]$, αναλογία 75% - 25%

Η πρώτη παραλλαγή αφορά σύνολα δεδομένων που το **75%** από τα δεδομένα του πεδίου ορισμού καταχωρείτε στο **training dataset** και το **25%** στο **testing dataset**. Όπως φανερώνεται στον παρακάτω πίνακα το δίκτυο για να μπορέσει να ανταποκριθεί σωστά έχει ανάγκη από **20 κρυφούς νευρώνες**. Με λιγότερους είτε δε μπορεί να προβλέψει σωστά είτε δεν καταφέρνει καθόλου. Η παράμετρος **AF** αφορά τη συνάρτηση ενεργοποίησης μέσω τις οποίας πήραμε τα καλύτερα αποτελέσματα

| Training Dataset – Testing Dataset = {75% / 25%} Range = [-5, +5] | | | | | |
|---|--------------|-------------------------------------|--------------|-------------------------------------|---------------|
| Neurons = 2 AF = TANGENT | | Neurons = 10 AF = TANGENT | | Neurons = 20 AF = SIGMOID | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.2303 | 0.2257 | 0.0675 | 0.0717 | 0.0001 | 0.0001 |

Πίνακας 17 1.1 Αποτελέσματα ELM με Καθαρά Δεδομένα

Βλέπουμε από τον παραπάνω πίνακα ότι το δίκτυο μέσω του ELM καταφέρνει με τη χρήση **20 νευρώνων στο κρυφό στρώμα** να φτάσει σε πλήρη κατανόηση της συνάρτησης.

7.2.1.2 Παραλλαγή 1.2: Καθαρά δεδομένα του πεδίου ορισμού [-5, +5), αναλογία 50% - 50%

Σε αυτή την παραλλαγή επιχειρήσαμε τα ίδια ακριβώς πειράματα με πριν με μόνη διαφοροποίηση το μέγεθος των συνόλων των δεδομένων. Πλέον **δώσαμε και στα δύο σύνολα 50% από το πλήθος των αρχικών δεδομένων με τυχαίο τρόπο**. Όπως θα παρατηρήσουμε το δίκτυο ανταποκρίνεται ακριβώς το ίδιο με πριν και προσαρμόζεται εξαιρετικά όταν έχει στη διάθεσή του **20 νευρώνες στο κρυφό στρώμα**.

| Training Dataset – Testing Dataset = {50% / 50%} Range = [-5, +5) | | | | | |
|---|--------------|------------------------------|--------------|------------------------------|--------------|
| Neurons = 2 AF = TANGENT | | Neurons = 10 AF = TANGENT | | Neurons = 20 AF = SIGMOID | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.1908 | 0.1900 | 0.0781 | 0.0815 | 0.0001 | 0.0001 |

Πίνακας 18 1.2 Αποτελέσματα ELM με Καθαρά Δεδομένα

7.2.1.3 Παραλλαγή 1.3: Καθαρά δεδομένα του πεδίου ορισμού [-5, +5), αναλογία 25% - 75%

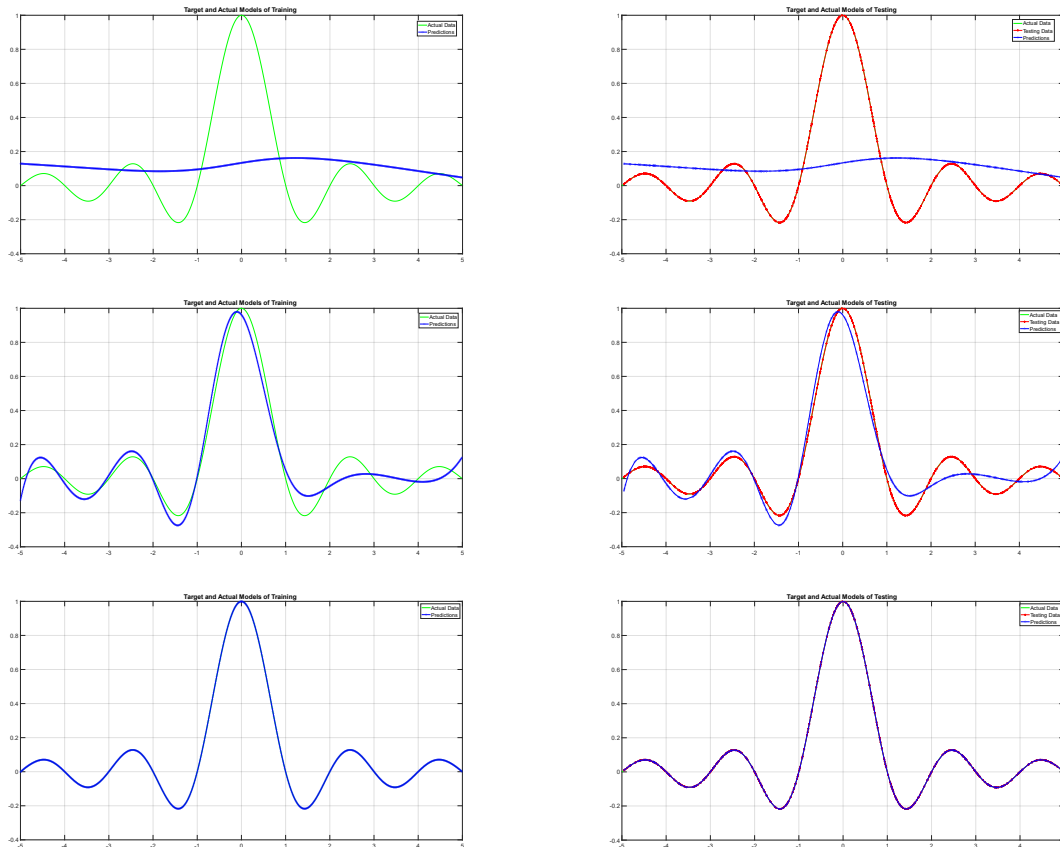
Η τελευταία παραλλαγή έχει ως διαφοροποίηση στην αναλογία των ποσοστών που δίνουμε στα δεδομένα, πως τα δεδομένα εκπαίδευσης είναι λιγότερα από τα δεδομένα εξέτασης. Αυτό γίνεται με απώτερο σκοπό να διαπιστώσουμε αν το δίκτυο μπορεί να εκπαιδευτεί έστω και με λίγα δεδομένα, αλλά παρόλα αυτά να μπορέσει να αναγνωρίσει τα μοτίβα. Θα δούμε και εδώ πως δεν αντιμετωπίζει κάποιο πρόβλημα το δίκτυο και μάλιστα η προσεγγιστική ικανότητά του είναι ίδια με της προηγούμενες παραλλαγές. Ο παρακάτω πίνακας το με τις μετρήσεις του **RMSE** το αποδεικνύει αυτό σε πρώτη φάση και δευτερευόντως οι εικόνες που παρουσιάζονται το καθιστούν ξεκάθαρο. Μάλιστα οι εικόνες αυτές αντιπροσωπεύουν και τις τρεις παραλλαγές που εξετάσαμε με καθαρά δεδομένα, καθώς ο αλγόριθμος δείχνει την ίδια ακριβώς αντιμετώπιση με το ίδιο πλήθος νευρώνων.

| Training Dataset – Testing Dataset = {25% / 75%} Range = [-5, +5) | | | | | |
|---|--------------|------------------------------|--------------|------------------------------|--------------|
| Neurons = 2 AF = COSINE | | Neurons = 10 AF = TANGENT | | Neurons = 20 AF = SIGMOID | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.2528 | 0.2431 | 0.0944 | 0.0979 | 0.0002 | 0.0002 |

Πίνακας 19 1.3 Αποτελέσματα ELM με Καθαρά Δεδομένα

Όπως προαναφέραμε οι παρακάτω εικόνες έχουν προκύψει από προσομοιώσεις με **50% - 50%** σε training – testing dataset σε σχέση με τα αρχικά δεδομένα από το πεδίο ορισμού. Όμως η απόδοση των δικτύων είναι ανάλογη στους 2 νευρώνες για όλες τις παραλλαγές, ομοίως στους 10 νευρώνες αλλά και στους 20 νευρώνες ανεξαρτήτως της συνάρτησης ενεργοποίησης. Η μόνη επισήμανση είναι ότι πάντα στους 20 νευρώνες η καλύτερη προσομοίωση ανεξαρτήτως συνόλων γίνεται μέσω της **sigmoid** συνάρτησης καθώς φτάνει στην απόλυτη προσέγγιση.

Συνεπώς στις ακόλουθες εικόνες η αριστερή στήλη αφορά τις προσεγγίσεις από τις προβλέψεις στην εκπαίδευση και η δεξιά τις αντίστοιχες των εξετάσεων με άγνωστα δεδομένα. Επιπρόσθετα η πρώτη σειρά αφορά το δίκτυο με 2 κρυφούς νευρώνες, η δεύτερη με 10 και τέλος η τρίτη με τους 20:



Εικόνα 33 ELM με καθαρά δεδομένα

7.2.2 Αποτελεσματικότητα του ELM με “μολυσμένα” συναρτησιακά δεδομένα

Το δεύτερο σύνολο δοκιμών που κάναμε με τον ELM έγινε με προσθήκη θορύβου στα συναρτησιακά δεδομένα των εξόδων. Η προσθήκη του θορύβου έγινε σε όλα τα δεδομένα εξόδου ομοιόμορφα κατανομημένος ο οποίος επιλέχθηκε τυχαία για το κάθε σημείο μέσα από το διάστημα $[-0.1, +0.1]$. Σκοπός της προσθήκης του θορύβου είναι η δυσκολία στην ανάγνωση των προτύπων από το δίκτυο μέσω του νευρωνικού αλγορίθμου, καθώς και ικανότητά του δικτύου να μπορεί να προσομοιώνει σε νέα τυχαία δεδομένα.

Για τη διεξαγωγή αυτού του προβλήματος δημιουργήθηκαν τρία διαφορετικά ζευγάρια συνόλων, επιλέγοντας τυχαία τα δεδομένα από το αρχικό πεδίο ορισμού $[-5, +5]$ με τα μολυσμένα δεδομένα. Οι αναλογίες των συνόλων είναι **75% - 25%**, **50% - 50%** και **25% - 75%** για **training dataset – testing dataset**. Επιπρόσθετα για τη κάθε υποπερίπτωση έγιναν προσομοιώσεις με τρεις διαφορετικές συναρτήσεις ενεργοποίησης οι οποίες είναι **sine – tangent – RBF**. Τέλος, υπάρχει και κλιμακωτή αύξηση του **πλήθους των κρυφών νευρώνων** σε **5 – 15 – 40**.

Παρακάτω θα παρουσιάσουμε σε κάθε παραλλαγή τα καλύτερα αποτελέσματα ανάλογα το πλήθος των νευρώνων (όπως κάναμε και στην **Ενότητα 7.2.1**) και τελικώς θα ακολουθήσουν οι εικόνες από τις καλύτερες προβλέψεις του αλγορίθμου.

.2.2.1 Παραλλαγή 2.1: Ομοιόμορφα κατανομημένος θόρυβος $[-0.1, +0.1]$, 75% - 25%

Πρώτες προσομοιώσεις που κάναμε σε αυτή την κατηγορία έχει σύνολα **75% - 25%** σε training – testing αντίστοιχα τα οποία έχουν επιλεγεί όπως προαναφέραμε τυχαία. Το δίκτυο από άποψη

αποτελεσματικότητας αν εξετάσουμε το RMSE είναι πολύ αποδοτικό σε όλες τις προσομοιώσεις. Παρόλα αυτά οι 5 νευρώνες στο κρυφό στρώμα δεν μπορούν να ωφελήσουν το δίκτυο καθώς οι προβλέψεις στις απεικονίσεις φανερώνουν ότι δεν έχει κατανοήσει τα πρότυπα. Γι' αυτό το λόγω αυξήσαμε και τους νευρώνες στους 15 αλλά και στους 40, με τα αποτελέσματα να είναι πανομοιότυπα. Όμως λόγω του γεγονότος ότι με 15 νευρώνες το δίκτυο προσφέρει την ίδια αποτελεσματικότητα και είναι πιο συμπαγές είναι η καλύτερη δυνατή επιλογή. Στον πίνακα που ακολουθεί παρουσιάζουμε τις καλύτερες προσομοιώσεις μέσω του ELM:

| Training Dataset – Testing Dataset = {75% / 25%} Range = [-5, +5] Noise = [-0.1, +0.1] | | | | | |
|--|--------------|------------------------------|--------------|------------------------------|--------------|
| Neurons = 5 AF = TANGENT | | Neurons = 15 AF = TANGENT | | Neurons = 40 AF = TANGENT | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.1224 | 0.1181 | 0.0581 | 0.0576 | 0.0578 | 0.0581 |

Πίνακας 20 2.1 Αποτελέσματα ELM με Μολυσμένα Δεδομένα

7.2.2.2 Παραλλαγή 2.2: Ομοιόμορφα κατανεμημένος θόρυβος [-0.1, +0.1], 50% - 50%

Η δεύτερη παραλλαγή έγινε με προσομοιώσεις οι οποίες έχουν σύνολα δεδομένων με μοιρασμένα δεδομένα από 50% τυχαία καταχωρημένα σε αυτά. Ακολουθώντας την ίδια μεθοδολογία στην κλιμακωτή αύξηση των νευρώνων, παρατηρούμε την ίδια εικόνα στην εξέλιξη και βελτίωση του δικτύου. Αρχικώς οι 5 νευρώνες δεν είναι αρκετοί για το δίκτυο, με τους 15 και τους 40 να είναι ισάξιοι στην αποδοτικότητα του δικτύου. Ο παρακάτω πίνακας συγκεντρώνει και επιβεβαιώνει τα παραπάνω:

| Training Dataset – Testing Dataset = {50% / 50%} Range = [-5, +5] Noise = [-0.1, +0.1] | | | | | |
|--|--------------|--------------------------|--------------|------------------------------|--------------|
| Neurons = 5 AF = TANGENT | | Neurons = 15 AF = RBF | | Neurons = 40 AF = TANGENT | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.1979 | 0.1965 | 0.0581 | 0.0594 | 0.0569 | 0.0597 |

Πίνακας 21 2.2 Αποτελέσματα ELM με Μολυσμένα Δεδομένα

7.2.2.3 Παραλλαγή 2.3: Ομοιόμορφα κατανεμημένος θόρυβος [-0.1, +0.1], 25% - 75%

Η τελευταία παραλλαγή αφορά σύνολα δεδομένων με 25% από το αρχικό πεδίο ορισμού να έχει δοθεί με τυχαίο τρόπο στο training dataset, ενώ το υπόλοιπο 75% στο testing dataset. Μέσω αυτής της παραλλαγής θέλουμε να διαπιστώσουμε αν ο αλγόριθμος είναι ικανός να εκπαιδεύσει το δίκτυο με δεδομένα που όχι μόνο έχουν θόρυβο αλλά δεν είναι και αρκετά. Όπως διαπιστώνεται, ο αλγόριθμος τα καταφέρνει στο ίδιο βαθμό όπως και στις προηγούμενες περιπτώσεις των παραλλαγών, με τις τιμές του RMSE να είναι ανάλογες με τις προηγούμενες παραλλαγές για το ίδιο πλήθος νευρώνων.

| Training Dataset – Testing Dataset = {25% / 75%} Range = [-5, +5] Noise = [-0.1, +0.1] | | | | | |
|--|--------------|--------------------------|--------------|------------------------------|--------------|
| Neurons = 5 AF = RBF | | Neurons = 15 AF = RBF | | Neurons = 40 AF = TANGENT | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.0973 | 0.0901 | 0.0581 | 0.0577 | 0.0575 | 0.0579 |

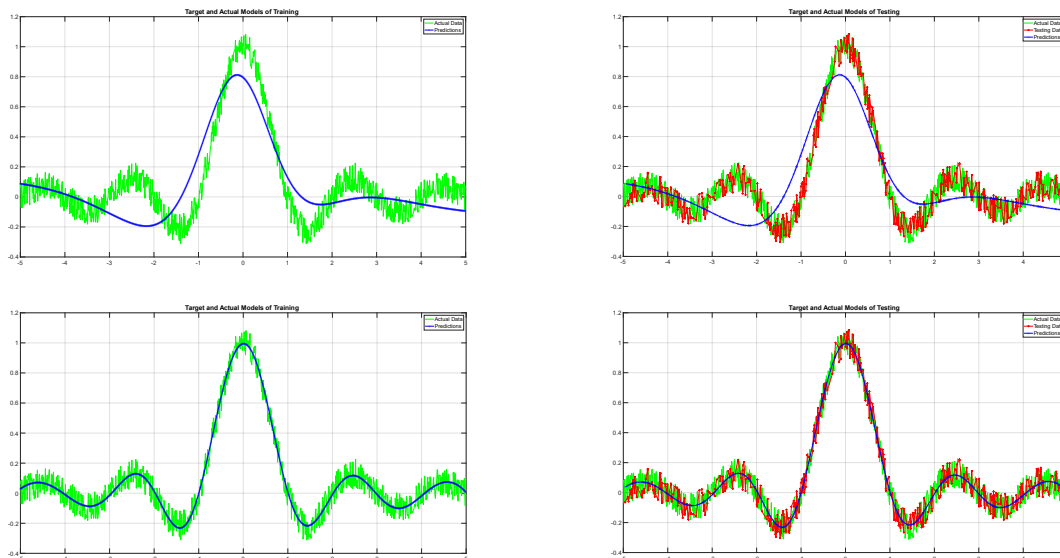
Πίνακας 22 2.3 Αποτελέσματα ELM με Μολυσμένα Δεδομένα

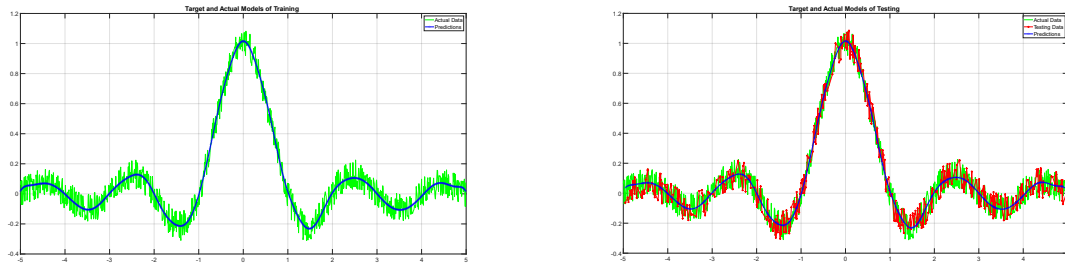
7.2.2.4 Τελική αποτίμηση των προσομοιώσεων με θόρυβο στα συναρτησιακά δεδομένα

Όπως φαίνεται στους παραπάνω πίνακες ο αλγόριθμος καταφέρνει να εκπαιδεύσει το δίκτυο εφόσον το πλήθος των νευρώνων είναι πάνω από 15. Έχει σημαντική συνέπεια σε όλες τις προσομοιώσεις ανεξαρτήτως του πλήθους των συνόλων εκπαίδευσης και εξέτασης. Διατηρεί τις ίδιες τιμές στο RMSE οι οποίες συμβαδίζουν και στις τρεις παραλλαγές που εκτελέσαμε.

Επιπρόσθετα, χρονικά ο αλγόριθμος είναι εξαιρετικά γρήγορος και πολλές φορές παρατηρήσαμε μηδενικές χρονικές τιμές στις προσομοιώσεις, καθώς οι χρονικές διάρκειες είναι πολύ μικρότερες από το ένα χιλιοστό.

Ακολούθως, παρουσιάζονται και οι γραφικές παραστάσεις από τις προβλέψεις του δικτύου. Επειδή τα αποτελέσματα είναι πανομοιότυπα για κάθε προσομοίωση με τους ίδιους νευρώνες ανεξαρτήτως αναλογίας ποσοτών στα σύνολα δεδομένων, επιλέξαμε να παρουσιάσουμε τις καλύτερες εκδοχές για 5 – 15 – 40 νευρώνες από σύνολα τιμών που είναι ισάξια στο 50%. Η πρώτη σειρά εικόνων αντιπροσωπεύει τα αποτελέσματα από τις εκτελέσεις με 5 νευρώνες στο δίκτυο μέσω του ELM, η δεύτερη για τους 15 και η τρίτη για 40. Τέλος, η αριστερή στήλη αντιπροσωπεύει τις προβλέψεις που αφορούν την εκπαίδευση ενώ η δεξιά αυτές των εξετάσεων σε νέα δεδομένα:





Εικόνα 34 ELM με μολυσμένα δεδομένα

7.2.3 Αποτελεσματικότητα του ELM με “αποκοπή διαστημάτων”

Μία από τις πιο δύσκολες δοκιμασίες στο πρόβλημα της συναρτησιακής προσέγγισης που εκτελέσαμε, είναι αυτή που θέλουμε να εξετάσουμε τη συμπεριφορά του δικτύου από την εκπαίδευση του με αποκομμένα διαστήματα από το πεδίο ορισμού. Αυτό σημαίνει ότι η εκπαίδευση του δικτύου μέσω του εξεταζόμενου αλγορίθμου ELM, θα γίνει έχοντας αφαιρέσει μη-διαδοχικά άλλα συνεχή υποδιαστήματα από το πεδίο ορισμού, για τα οποία τα δεδομένα που υπάρχουν θα δοθούν για το στάδιο της εξέτασης.

Σε αυτή την περίπτωση έχουμε δύο διαφορετικά πεδία ορισμού $[-5, +5]$ και $[-10, +10]$ μέσα από τα οποία έχουμε δημιουργήσει τα διανύσματα του x με βήμα $= 0.01$. Για το πρώτο πεδίο ορισμού αφαιρούμε 3 υποδιαστήματα από 1 μονάδα μήκους το καθένα, άρα 100 τιμές του x από το καθένα, και για το δεύτερο 4 υποδιαστήματα από 1.5 μονάδα μήκους το καθένα, άρα 150 τιμές του x , από το καθένα. Από τα νέα υποδιαστήματα που μένουν εκπαιδεύουμε το δίκτυο μέσω του ELM παίρνοντας τις μισές διαθέσιμες τιμές των διανυσμάτων του x και στη συνέχεια σε δεύτερο χρόνο όλες τις διαθέσιμες τιμές. Οι προσομοιώσεις που κάναμε έχουν ως συναρτήσεις ενεργοποίησης τις **Tangent** και **RBF** και εξετάζουμε την απόδοση του δικτύου μέσω του RMSE σε training και testing όσο και τις γραφικές παραστάσεις που καταφέρνει το δίκτυο μέσω των προβλέψεών του.

7.2.3.1 Παραλλαγή 3.1: 700 / 700 τιμές του πεδίου ορισμού $[-5, +5]$

Η πρώτη απόπειρα που κάναμε είναι στο πεδίο ορισμού $[-5, +5]$ για το οποίο όλο το σύνολο τιμών του το χρησιμοποιούμε για το στάδιο εξέτασης. Από αυτό πεδίο αφαιρούμε 3 υποδιαστήματα με 100 συνεχείς τιμές και δημιουργούμε ένα νέο διάστημα το οποίο όμως δεν είναι συνεχές, το οποίο το προσφέρουμε όλο στο δίκτυο για να εκπαιδευτεί. Αρχικά εκπαιδεύουμε το δίκτυο πρώτα για 5, στη συνέχεια για 10 και τέλος για 25 κρυφούς νευρώνες, με απώτερο σκοπό να αποφανθούμε αν το δίκτυο μπορεί να κατανοήσει τα μοτίβα και να προβλέψει στα διαστήματα που έχουν αποκοπεί σωστά τα δεδομένα. Το αποτέλεσμα είναι ότι το δίκτυο αρχικά δε μπορεί να προβλέψει και να κατανοήσει σωστά τα μοτίβα με τους 5 κρυφούς νευρώνες. Σε δεύτερο στάδιο με τους 10 νευρώνες μολονότι ανταποκρίνεται καλύτερα, δεν προσφέρει τις καλύτερες προβλέψεις. Για να φτάσουμε στο τέλος στις προσομοιώσεις με 25 νευρώνες και να πάρουμε αποτελέσματα που έχουν μηδενικό **RMSE**, δηλαδή το ο ELM καταφέρνει να εκπαιδεύσει το δίκτυο όχι μόνο στο να κατανοήσει πλήρως τα δεδομένα αλλά και να ανταποκριθεί σε νέα δεδομένα για τα οποία δεν έχει γνώση σε ορισμένα σημεία του πεδίου ορισμού. Παρακάτω ακολουθεί πίνακας που συγκεντρώνει τα καλύτερα αποτελέσματα των προσομοιώσεων για κάθε επίπεδο των κρυφών νευρώνων:

| Training Dataset = {700 / 700} Range = [-5, +5) | | | | | |
|---|--------------|--------------------------|--------------|--------------------------|--------------|
| Neurons = 5 AF = RBF | | Neurons = 10 AF = RBF | | Neurons = 25 AF = RBF | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.0958 | 0.1663 | 0.0338 | 0.0578 | 0 | 0 |

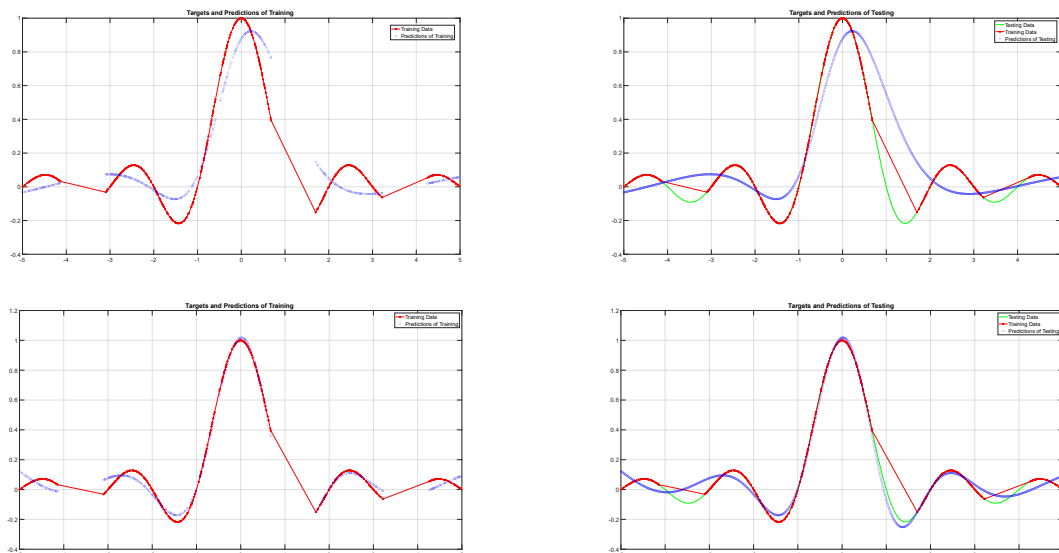
Πίνακας 23 3.1 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων

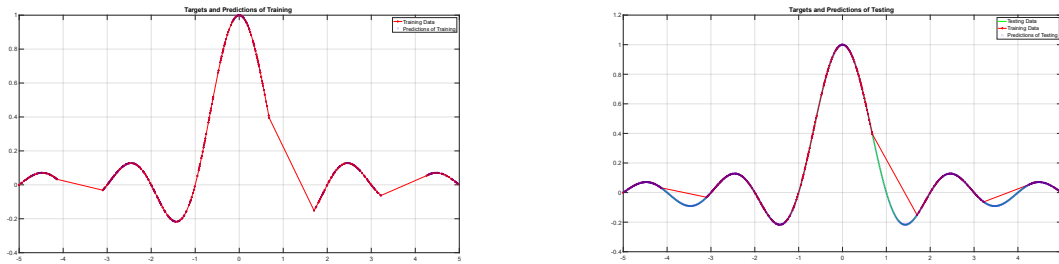
7.2.3.2 Παραλλαγή 3.2: 350 / 700 τιμές του πεδίου ορισμού [-5, +5)

Στην υποενότητα αυτή επαναλάβουμε τις προσομοιώσεις της προηγούμενης *υπό-ενότητας 7.2.3.1*, με μόνη διαφοροποίηση ότι προσφέραμε στο δίκτυο τα μισά δεδομένα από αυτά τα οποία περιέχει το νέο διάστημα τιμών με τα αποκομμένα υποδιαστήματα. Ο λόγος που το κάναμε αυτό είναι για να εξετάσουμε αν μπορεί το δίκτυο να ανταποκριθεί στην εκπαίδευση του ELM με λιγότερα δεδομένα από αυτά εν τέλει που θα κληθεί να εξεταστεί. Όπως θα δούμε παρά το γεγονός ότι εκπαιδεύτηκε με 350 τυχαίες τιμές ανταπεξήλθε στον ίδιο βαθμό με πριν, κάτι το οποίο αποτυπώνεται και στον πίνακα που ακολουθεί, αλλά και σχετικές εικόνες που απεικονίζουν τις προβλέψεις του δικτύου.

| Training Dataset = {350 / 700} Range = [-5, +5) | | | | | |
|---|--------------|--------------------------|--------------|--------------------------|--------------|
| Neurons = 5 AF = TANGENT | | Neurons = 10 AF = RBF | | Neurons = 25 AF = RBF | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.1441 | 0.1962 | 0.0320 | 0.0371 | 0 | 0 |

Πίνακας 24 3.2 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων





Εικόνα 35 ELM με αποκοπή διαστημάτων στο πεδίο ορισμού [-5, +5)

Στις προηγούμενες εικόνες η αριστερή στήλη περιέχει όλες τις προβλέψεις του δικτύου μετά την εκπαίδευση του δικτύου και στη δεξιά οι αντίστοιχες μετά τις προσομοιώσεις στα άγνωστα δεδομένα. Αντίστοιχα η κάθε σειρά αντιπροσωπεύει τις ξεχωριστές κλίμακες του πλήθους των κρυφών νευρώνων που εφαρμόζονται στο δίκτυο, αρχικά με 5, στη συνέχεια με 10 και τέλος με 25.

Σε αυτό το σημείο να επισημάνουμε δε ότι οι γραφικές απεικονίσεις από την παραλλαγή με τις 700 / 700 τιμές που δίνουμε στο δίκτυο της υπό-ενοτήτας 7.2.3.1 είναι αντίστοιχες με αυτές της Εικόνας 35, γι' αυτό το λόγο επιλέξαμε να παρουσιάσουμε την πιο "δύσκολη" κατά τα φαινόμενα περίπτωση με ελλιπές σύνολο δεδομένων εκπαίδευσης. Τέλος, η περάτωση κάθε προσομοίωσης είναι ακαριαία σε βαθμό που οι τιμές του χρόνου είναι κάτω από το 1 χιλιοστό του δευτερολέπτου.

7.2.3.3 Παραλλαγή 3.3: 1400 / 1400 τιμές του πεδίου ορισμού [-10, +10)

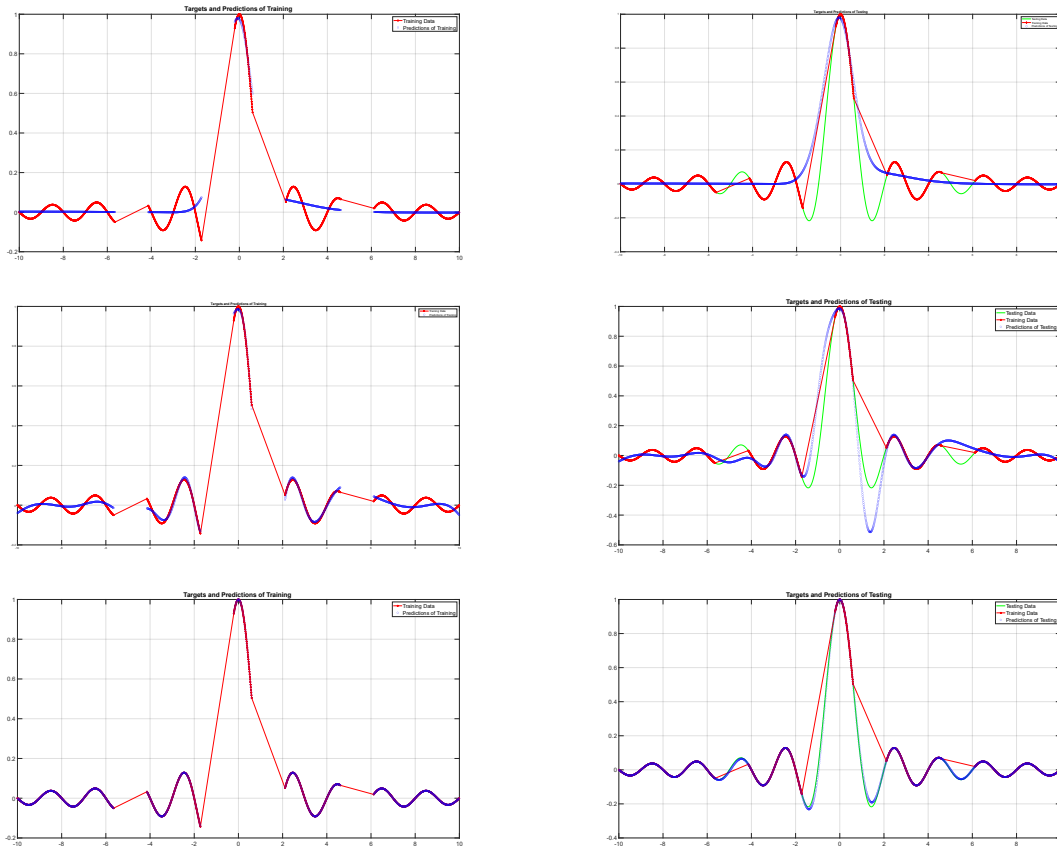
Θέλοντας να ελέγξουμε σε πιο εκτενή βαθμό την αποτελεσματικότητα του ELM σε αυτού του είδους την παραλλαγή του προβλήματος συναρτησιακής παλινδρόμησης, εκτελέσαμε τις προσομοιώσεις των προηγούμενων δύο υπό-ενοτήτων αλλά για το πεδίο ορισμού [-10, +10). Σε αυτό το πεδίο ορισμού αφαιρέσαμε 4 συνεχή υποσύνολα μήκους 1.5 μονάδας το καθένα. Το βήμα και εδώ είναι 0.01, το οποίο μεταφράζεται σε 150 στοιχεία για την $y = \text{sinc}(x)$ για το κάθε αποκομμένο διάστημα. Συνεπώς, έχουμε 1400 σημεία δεδομένων για να εκπαιδεύσουμε το δίκτυο μέσω του ELM. Χρησιμοποιώντας το πεδίο ορισμού ως σύνολο δεδομένων για το στάδιο της εξέτασης εφαρμόσαμε για 5 – 20 – 40 νευρώνες στον κρυφό στρώμα τρέξαμε μια σειρά από προσομοιώσεις για να διαλευκάνουμε αν ο αλγόριθμος μπορεί να εκπαιδεύσει σωστά το δίκτυο.

Όπως σε όλες τις προηγούμενες προσομοιώσεις των παραλλαγών παρουσιάζονται στο επόμενο πίνακα τα αποτελέσματα μέσα από τις μετρήσεις του RMSE, τα οποία όμως δεν αποσαφηνίζουν με σωστό τρόπο την πραγματική εικόνα. Αυτό φαίνεται καλύτερα στις εικόνες με τις οποίες απεικονίζονται οι προβλέψεις που έκανε το δίκτυο για κάθε κλίμακα νευρώνων.

| Training Dataset = {1400 / 1400} Range = [-10, +10) | | | | | |
|---|--------------|--------------------------|--------------|--------------------------|--------------|
| Neurons = 5 AF = TANGENT | | Neurons = 20 AF = RBF | | Neurons = 40 AF = RBF | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.0800 | 0.2240 | 0.0189 | 0.0446 | 0.0001 | 0.0079 |

Πίνακας 25 3.3 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων

Όπως είναι το ξεκάθαρο με **40 κρυφούς νευρώνες** το δίκτυο καταφέρνει να εκπαιδευτεί εξαιρετικά και να ανταπεξέλθει στα δεδομένα από τα υποσύνολα που δεν έχει εκπαιδευτεί. Ακολουθούν εικόνες που κάνουν ξεκάθαρο αυτό το απόφθεγμα.



Εικόνα 36 ELM με αποκοπή διαστημάτων στο πεδίο ορισμού [-10, +10)

Στις προηγούμενες εικόνες η κάθε σειρά αντιπροσωπεύει τις ξεχωριστές κλίμακες του πλήθους των κρυφών νευρώνων που εφαρμόζονται στο δίκτυο, αρχικά με 5, στη συνέχεια με 10 και τέλος με 25. Η αριστερή στήλη περιέχει όλες τις προβλέψεις του δικτύου μετά το training του δικτύου και στη δεξιά οι αντίστοιχες μετά τις προσομοιώσεις στα άγνωστα δεδομένα στο testing.

7.2.3.4 Παραλλαγή 3.4: 700 / 1400 τιμές του πεδίου ορισμού [-10, +10)

Για να προσθέσουμε ένα επίπεδο δυσκολίας στις προσομοιώσεις, μειώσαμε τα δεδομένα στο σύνολο εκπαίδευσης στα **700**, το οποίο μεταφράζεται στο **35%** από τα δεδομένα του πεδίου ορισμού, με το υπόλοιπο **65%** να είναι άγνωστο στο δίκτυο και να καλείται να τα αναγνωρίσει με σωστές προβλέψεις. Τα δεδομένα επιλέγονται με τυχαίο τρόπο, συνεπώς όχι μόνο έχει άγνοια το δίκτυο και υποσύνολα του πεδίου ορισμού που θα εξεταστεί, αλλά ταυτόχρονα υπάρχουν και ελλείψεις στο πεδίο που εκπαιδεύεται.

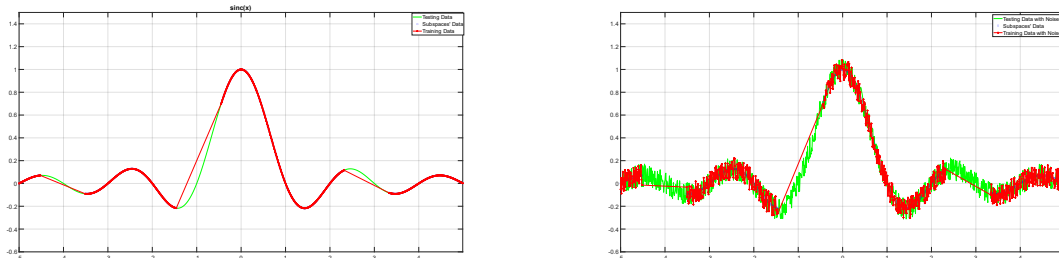
Όπως θα δούμε στον πίνακα που ακολουθεί το δίκτυο έχει ακριβώς την ίδια εικόνα στις προσομοιώσεις με τις προσομοιώσεις που κάναμε στην **υποενότητα 7.2.3.3** και οι προβλέψεις είναι αντίστοιχες με αυτές που απεικονίζει το δίκτυο στο σύμπλεγμα **Εικόνας 36**.

| Training Dataset = {700 / 1400} Range = [-10, +10) | | | | | |
|--|--------------|--------------------------|--------------|--------------------------|--------------|
| Neurons = 5 AF = TANGENT | | Neurons = 20 AF = RBF | | Neurons = 40 AF = RBF | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.0817 | 0.0764 | 0.0199 | 0.0343 | 0 | 0.0021 |

Πίνακας 26 3.4 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων

7.2.4 Αποτελεσματικότητα του ELM με “μολυσμένα δεδομένα” και “αποκοπή διαστημάτων”

Η πιο δύσκολη παραλλαγή του συναρτησιακού προβλήματος παλινδρόμησης για την πρόβλεψη της $y = \text{sinc}(x)$, είναι αυτό στο οποίο όχι μόνο αφαιρούμε υποδιαστήματα από το πεδίο ορισμού, αλλά προσθέτουμε και ομοιόμορφα καταναμημένο θόρυβο στα δεδομένα ο οποίο επιλέγεται τυχαία μέσα από το $[-0.1, +0.1]$.



Εικόνα 37 Διαγράμματα με μολυσμένα δεδομένα και μη και αποκομμένα διαστήματα

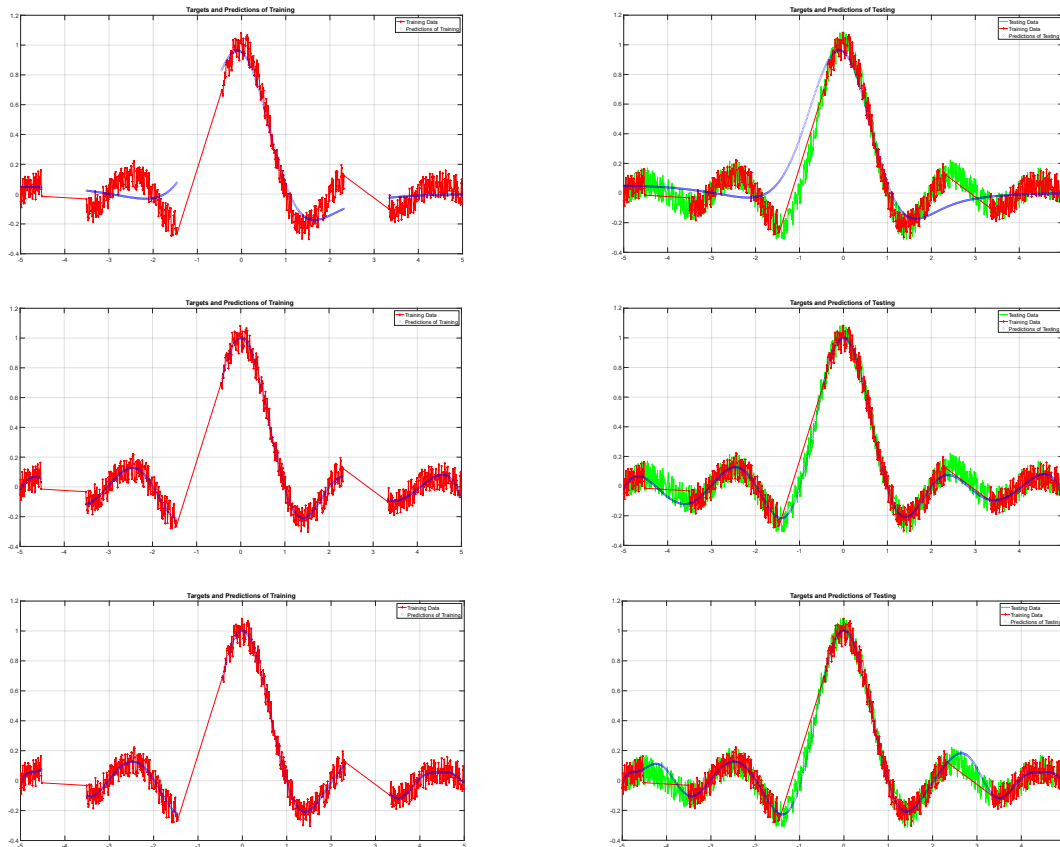
Όπως φαίνεται στις προηγούμενες εικόνες μέσα από το διάστημα $[-5, +5]$, στο αριστερό σχήμα φαίνεται πως είναι η συνάρτηση με καθαρά τα δεδομένα και στα δεξιά εικόνα τα δεδομένα με θόρυβο, η οποία θα δοθεί θα χρησιμοποιηθεί και για τις προσομοιώσεις.

Ο αλγόριθμος τους ELM καταφέρνει να εκπαιδεύσει σε πολύ καλό βαθμό το δίκτυο, ειδικότερα όταν τροφοδοτείται με τον κατάλληλο αριθμό νευρώνων στο κρυφό στρώμα. Όπως φαίνεται στον ακόλουθο πίνακα οι τιμές του RMSE είναι σε πολύ χαμηλό επίπεδο, κάτι το οποίο γίνεται πιο ξεκάθαρο και από τις προβλέψεις του δικτύου που αποτυπώνονται στις εικόνες που ακολουθούν τον πίνακα.

| Training Dataset = {700 / 700} Range = [-5, +5) | | | | | |
|---|--------------|--------------------------|--------------|--------------------------|--------------|
| Neurons = 5 AF = RBF | | Neurons = 20 AF = RBF | | Neurons = 40 AF = RBF | |
| Training RMSE | Testing RMSE | Training RMSE | Testing RMSE | Training RMSE | Testing RMSE |
| 0.1427 | 0.1545 | 0.0579 | 0.0658 | 0.0570 | 0.0624 |

Πίνακας 27 4.1 Αποτελέσματα ELM με Αποκοπή Υποσυνόλων σε Μολυσμένα δεδομένα

Το παρακάτω σύνολο εικόνων αφορούν τα αποτελέσματα των προβλέψεων του δικτύου με 5 – 20 – 40 νευρώνες (αντίστοιχα στην κάθε σειρά), η αριστερή στήλη τις προβλέψεις από τις εκπαιδεύσεις και δεξιά οι προβλέψεις στα άγνωστα σύνολα δεδομένων.



Εικόνα 38 ELM με αποκοπή διαστημάτων και δεδομένα με θόρυβο

7.3 Αποτίμηση – Σύγκριση αποτελεσμάτων Συναρτησιακής Παλινδρόμησης

Σε αυτή την ενότητα θα συγκρίνουμε τα αποτελέσματα που πήραμε από τις προσομοιώσεις στο συναρτησιακό πρόβλημα παλινδρόμησης, μεταξύ του ELM και του Backpropagation. Στις συγκρίσεις τις οποίες θα κάνουμε θα παρουσιάσουμε την καλύτερη δυνατή προσομοίωση για τον κάθε νευρωνικό αλγόριθμο για κάθε παραλλαγή του προβλήματος. Συνεπώς, ανάλογα την παραλλαγή του προβλήματος καθώς και το πεδίο ορισμού που θα ορίσουμε για κάθε περίπτωση θα αποτελεί μία ξεχωριστή σύγκριση.

Αρχικά θα συγκρίνουμε τους 3 ξεχωριστούς αλγόριθμους για τις προβλεπτικές τους ικανότητες πάνω στα καθαρά δεδομένα για τη συνάρτηση $y = \text{sinc}(x)$, θα ακολουθήσει αυτή με τα μολυσμένα και τέλος τα πειράματα με τα αποκομμένα υποδιαστήματα από το πεδίο ορισμού. Τα βασικά σημεία που θα συγκρίνουμε τους 3 αλγόριθμους είναι η αποδοτικότητα τους μέσω του RMSE – των γραφικών παραστάσεων από την πρόβλεψη των δεδομένων, ο όγκος του δικτύου που χρειάζονται για να το επιτύχουν και τέλος ο χρόνος για την περάτωση μιας προσομοίωσης και όλα αυτά για τη βέλτιστη προσπάθεια της κάθε περίπτωσης.

7.3.1 Σύνολα δεδομένων με καθαρές τιμές

Η πρώτη περίπτωση λοιπόν που εξετάζουμε είναι αυτή που έχει καθαρές τιμές στα δεδομένα των δύο συνόλων δεδομένων (training – testing dataset). Όπως είδαμε όμως επειδή οι αλγόριθμοι μπορούν να εκπαιδεύσουν σε εξαιρετικό βαθμό το δίκτυο σε αυτή την παραλλαγή του προβλήματος, επιλέξαμε να συγκρίνουμε την περίπτωση στην οποία οι αλγόριθμοι **καλούνται να εκπαιδευτούν έχοντας στη διάθεσή τους το 25% από τα δεδομένα του πεδίου ορισμού, με το υπόλοιπο 75% να δίνεται στο δίκτυο για το πειραματικό στάδιο με άγνωστα δεδομένα**, δηλαδή στο **testing**. Όπως είδαμε στις προηγούμενες υποενότητες (*υποενότητες 7.1.1 και 7.2.1*) το εξεταζόμενο πεδίο ορισμού είναι το $[-5, +5]$ και το βήμα για δημιουργία του x διανύσματος είναι το **0.01**, όπως και σε όλες τις περιπτώσεις. Γι' αυτό το λόγο δε θα χρειαστεί να επισημάνουμε το βήμα.

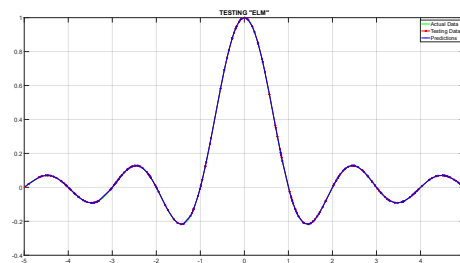
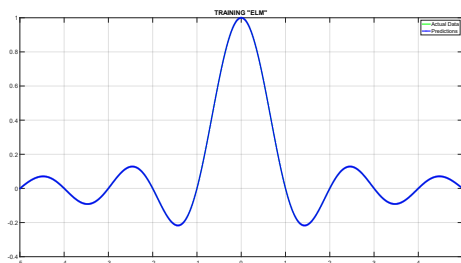
Στον πίνακα τον οποίο ακολουθεί παρουσιάζονται οι παράμετροι οι οποίοι χρειάζονταν οι αλγόριθμοι για να προσφέρουν την καλύτερη δυνατή απόδοση στο δίκτυο και ακολούθως η ερμηνεία αυτών. Να επισημάνουμε ότι το **AF** αφορά τη **Συνάρτηση Ενεργοποίησης** την οποία χρησιμοποιούμε στον **ELM**, ενώ το **LR** αφορά τον **Ρυθμό Εκμάθησης** που χρησιμοποιούμε στον **Backpropagation**. Επίσης, το **TR** αντιπροσωπεύει τιμή από το **Training** σκέλος ενώ το **TS** από το **Testing** σκέλος.

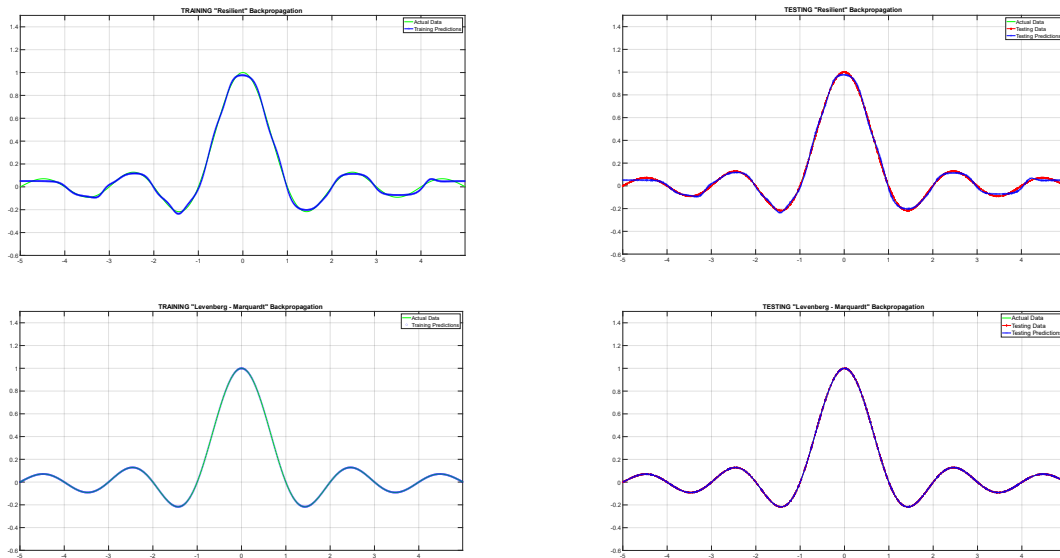
| Neurons | | AF | Algorithm | Neurons | | LR | Algorithm | Neurons | | LR | Algorithm |
|---------------|---------------|---------|-----------|---------------|---------------|---------|-----------|---------------|---------------|---------|-----------|
| 20 | | Sigmoid | ELM | 20 | | 0.7 | BP RP | 10 | | 0.3 | BP LM |
| TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS | TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS | TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS |
| 0.0156 | 0 | 0.0002 | 0.0002 | 0.3906 | 0 | 0.0218 | 0.0208 | 0.5625 | 0 | 0 | 0 |

Πίνακας 28 Συγκριτικά Αποτελέσματα BP - ELM με Καθαρά Δεδομένα

Όπως φαίνεται από τον παραπάνω πίνακα, η καλύτερη εκτέλεση απ' όλες σχετικά με τον χρόνο είναι αυτή του ELM. Σχετικά με το μέγεθος του δικτύου ο καλύτερος αλγόριθμος είναι αυτός του BP LM, δηλαδή του Levenberg – Marquardt. Τέλος, αποδοτικά ο Resilient Backpropagation υστερεί σε σχέση με τους άλλους 2, μολονότι αποδίδει εξαιρετικά. Ο ELM και ο BP LM είναι ισάξιοι κάτι το οποίο παρουσιάζεται και στις ακόλουθες εικόνες, όπου φανερώνουν ότι τα δίκτυα με τις παραπάνω παραμέτρους καταφέρνουν να κατανοήσουν πλήρως τη συνάρτηση $y = \sin(x)$ παρά το γεγονός ότι δεν εκπαιδεύονται με μεγάλο πλήθος δεδομένων αλλά με τυχαία και λιγοστά δεδομένα.

Στις εικόνες που ακολουθούν στην αριστερή στήλη περιέχονται όλες οι προβλέψεις από τους αλγόριθμους αναφορικά με την εκπαίδευση ενώ δεξιά σχετικά με την εξέταση. Επιπρόσθετα στην πρώτη σειρά εμπεριέχονται οι εικόνες από τις καλύτερες προσομοιώσεις του ELM, στη δεύτερη από τον Resilient Backpropagation ενώ στην τρίτη από τον Levenberg – Marquardt. Αυτή θα είναι η δομή για όλα τα συμπλέγματα εικόνων που θα δούμε και παρακάτω.





Εικόνα 39 Συγκριτικές Απεικονίσεις BP - ELM με Καθαρά Δεδομένα

Συνεπώς, ο ELM είναι πιο γρήγορος κατά **36 φορές** από τους **BP**, μολονότι αποδεδειγμένα και οι **3 αλγόριθμοι** είναι εξαιρετικά γρήγοροι, αλλά ο **BP LM** μπορεί να αποδώσει και σε πιο συμπαγές δίκτυο από νευρώνες, καθώς θέλει **50% λιγότερους νευρώνες**.

7.3.2 Σύνολα δεδομένων με μολυσμένες τιμές

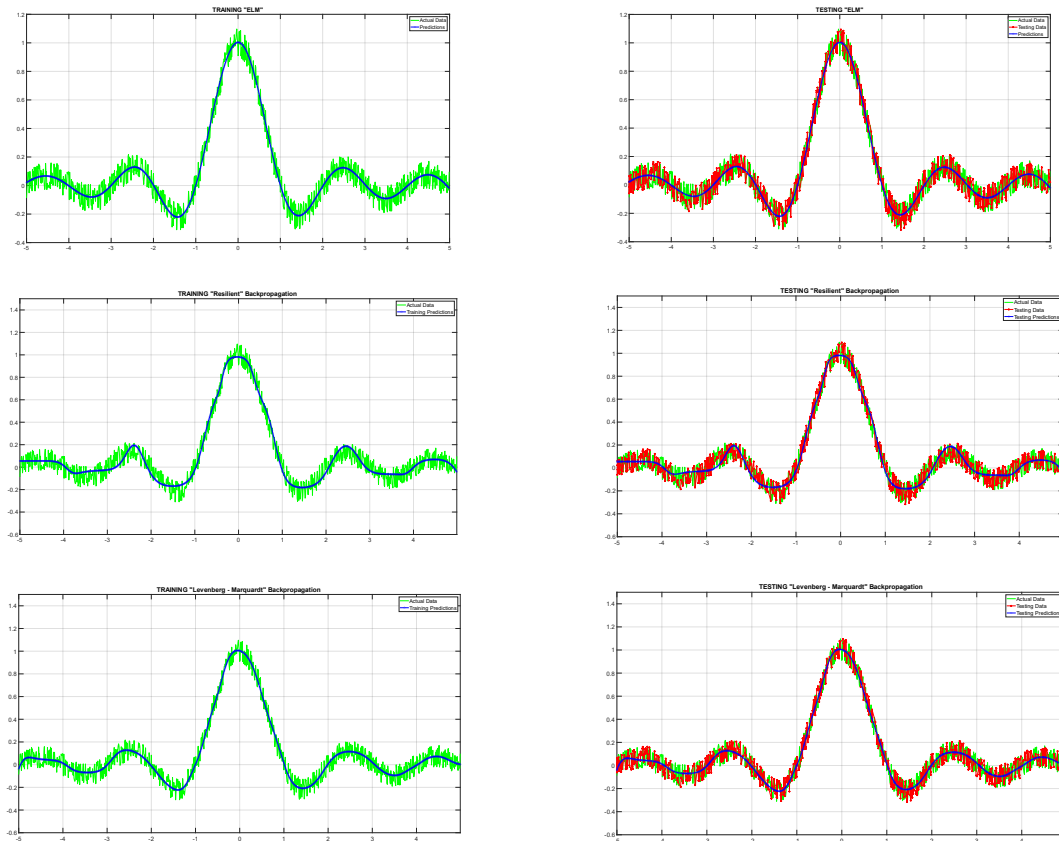
Σε αυτή την υποενότητα θα εξετάσουμε την αποτελεσματικότητα των νευρωνικών αλγορίθμων σε σύνολα δεδομένων που οι διανυσματικές τιμές των εξόδων y έχουν δεχθεί την προσθήκη ομοιόμορφα κατανομημένου θορύβου. Αυτός ο θόρυβος όπως είδαμε στις **υποενότητες 7.1.2** και **7.2.2** υπάρχει σε όλες τις τιμές των y και είναι επιλεγμένος τυχαία μέσα από το διάστημα $[-0.1, +0.1]$. Και σε αυτή την παραλλαγή του συναρτησιακού προβλήματος επιλέξαμε να συγκρίνουμε την αποδοτικότητα των αλγορίθμων στις πιο δύσκολες συνθήκες οι οποίες είναι στο πεδίο ορισμού $[-5, +5]$, όπου λαμβάνουμε το **25%** από τα δεδομένα που διαθέτει με τυχαίο τρόπο για να δημιουργήσουμε το **σύνολο δεδομένων για την εκπαίδευση** των δικτύων. Το υπόλοιπο **75%** χρησιμοποιείται στο **testing dataset για το πειραματικό σκέλος**.

Αρχικώς παρουσιάζουμε τα συγκεντρωτικά αποτελέσματα των καλύτερων προσομοιώσεων των αλγορίθμων για τα δίκτυα, τα οποία αποτυπώνονται στον ακόλουθο πίνακα:

| Neurons | | AF | Algorithm | Neurons | | LR | Algorithm | Neurons | | LR | Algorithm |
|---------------|---------------|---------|-----------|---------------|---------------|---------|-----------|---------------|---------------|---------|-----------|
| 15 | | RBF | ELM | 15 | | 0.3 | BP RP | 15 | | 0.3 | BP LM |
| TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS | TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS | TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS |
| 0 | 0 | 0.0581 | 0.0577 | 0.4688 | 0 | 0.0694 | 0.0753 | 0.4531 | 0 | 0.0577 | 0.0576 |

Πίνακας 29 Συγκριτικά Αποτελέσματα BP - ELM με Μολυσμένα Δεδομένα

Όπως φαίνεται από τον παραπάνω πίνακα οι αλγόριθμοι είναι πολύ κοντά μεταξύ τους. Σχετικά με τον όγκο του δικτύου και οι 3 αλγόριθμοι κατάφεραν τις καλύτερες προσομοιώσεις τους με 15 νευρώνες στο κρυφό στρώμα. Χρονικά προηγείται και πάλι ο ELM συγκριτικά με τους άλλους δύο. Τέλος, σε επίπεδο καθαρά απόδοσης του μέσου τετραγωνικού σφάλματος ρίζας ο ELM και ο Levenberg – Marquardt Backpropagation έχουν το ίδιο απόδοση η οποία είναι καλύτερη από τον Resilient Backpropagation και αυτό θα διαπιστωθεί στις ακόλουθες εικόνες.



Εικόνα 40 Συγκριτικές Απεικονίσεις BP - ELM με Μολυσμένα Δεδομένα

7.3.3 Αποκοπή υποδιαστημάτων από το πεδίο ορισμού

Η τρίτη κατά σειρά περίπτωση που συγκρίνουμε τους τρεις αλγορίθμους αφορά την εκπαίδευσή τους σε σύνολα τιμών για τα οποία δεν υπάρχουν δεδομένα για κάποια υποδιαστήματα. Όπως είδαμε και στις επιμέρους προσομοιώσεις στις *υποενότητες 7.1.3* και *7.2.3*, από το αρχικό πεδίο ορισμού αφαιρούμε 3 ή 4 υποδιαστήματα (ανάλογα το μέγεθος του πεδίου ορισμού) τα οποία έχουν μήκος 1 ή 1.5 μονάδα. Αυτό σημαίνει ότι τα σημεία της συνάρτησης $y = \text{sinc}(x)$ που περιέχονται σε αυτά τα υποδιαστήματα, χρησιμοποιούνται αποκλειστικά για το πειραματικό στάδιο, χωρίς να έχει εκπαιδευτεί το δίκτυο σε αυτά. Παράλληλα από το νέο πεδίο ορισμού επιλέγονται τυχαία οι μισές τιμές ή επιλέγονται όλες για να εκπαιδευτεί το δίκτυο.

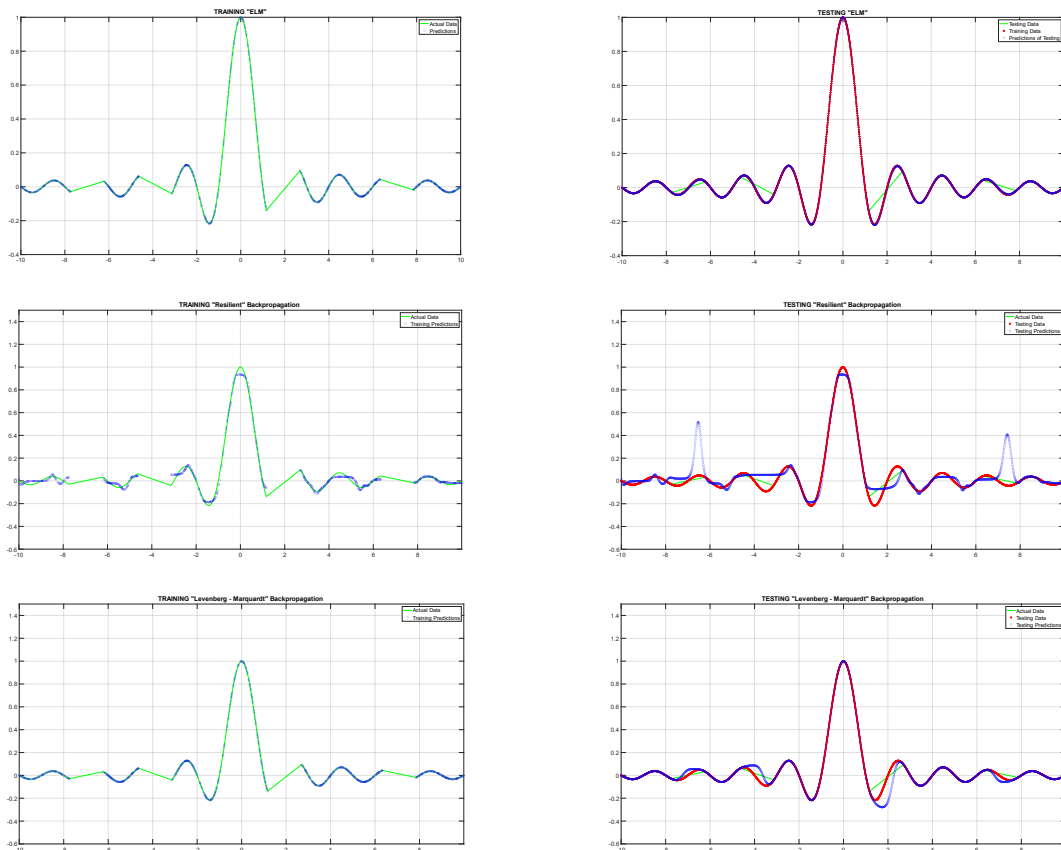
Στις προσομοιώσεις που θα συγκρίνουμε τους αλγορίθμους, θα επιλέξουμε το πεδίο ορισμού $[-10, +10]$, από το οποίο θα αφαιρέσουμε συνολικά 4 υποδιαστήματα μήκους 1.5 μονάδας με συνεχείς τιμές, τα οποία δεν είναι συνεχή μεταξύ τους. Τα δύο υποδιαστήματα αφαιρούνται από τις αρνητικές τιμές του διανύσματος x και υπόλοιπα δύο από τις θετικές. Από τα σημεία που απομένουν επιλέγονται τυχαία τα 700 από τα 1400. Απώτερος σκοπός είναι να διαπιστώσουμε αν μπορεί το δίκτυο με περιορισμένο αριθμό δεδομένων και χωρίς να έχει εκπαιδευτεί σε ορισμένα υποσύνολα του πεδίου ορισμού, να προβλέψει όλη τη συνάρτηση του $y = \text{sinc}(x)$.

Στον πίνακα που ακολουθεί αποτυπώνονται οι καλύτερες δυνατές προσομοιώσεις βάσει των **RMSE** και για τους τρεις αλγορίθμους:

| Neurons | | AF | Algorithm | Neurons | | LR | Algorithm | Neurons | | LR | Algorithm |
|---------------|---------------|---------|-----------|---------------|---------------|---------|-----------|---------------|---------------|---------|-----------|
| 40 | | RBF | ELM | 40 | | 0.2 | BP RP | 30 | | 0.2 | BP LM |
| TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS | TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS | TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS |
| 0.0156 | 0 | 0.0001 | 0.0006 | 0.3281 | 0 | 0.0221 | 0.0890 | 2.5469 | 0.0312 | 0 | 0.0389 |

Πίνακας 30 Συγκριτικά Αποτελέσματα BP - ELM με Αποκοπή Υποδιαστημάτων

Όπως φαίνεται από τον πίνακα παραπάνω, ο ELM είναι πολύ πιο αποδοτικός σε χρόνο περάτωσης αλλά και πιο αποτελεσματικός, καθώς οι τιμές του RMSE είναι σχεδόν μηδενικές. Αντίθετα οι Backpropagation υλοποιήσεις υστερούν κυρίως όμως ο Resilient Backpropagation σε αποτελεσματικότητα. Αυτό θα φανεί πιο καθαρά στις ακόλουθες εικόνες, αφού δε μπορεί να προβλέψει με σωστό τρόπο τα δεδομένα που υπάρχουν στα διαστήματα που δεν έχει εκπαιδευτεί στη συνάρτηση $y = \text{sinc}(x)$. Ο Levenberg – Marquardt από την άλλη πλευρά είναι πιο αποτελεσματικός και όχι στο επίπεδο του ELM. Μάλιστα χρονικά είναι πιο αργός και από τους δύο όντας 8 φορές πιο αργός από τον Rprop και 160 φορές (!) πιο αργός από τον ELM.



Εικόνα 41 Συγκριτικές Απεικονίσεις BP - ELM με Αποκοπή Υποδιαστημάτων

7.3.4 Αποκοπή υποδιαστημάτων από το πεδίο ορισμού με μολυσμένα δεδομένα

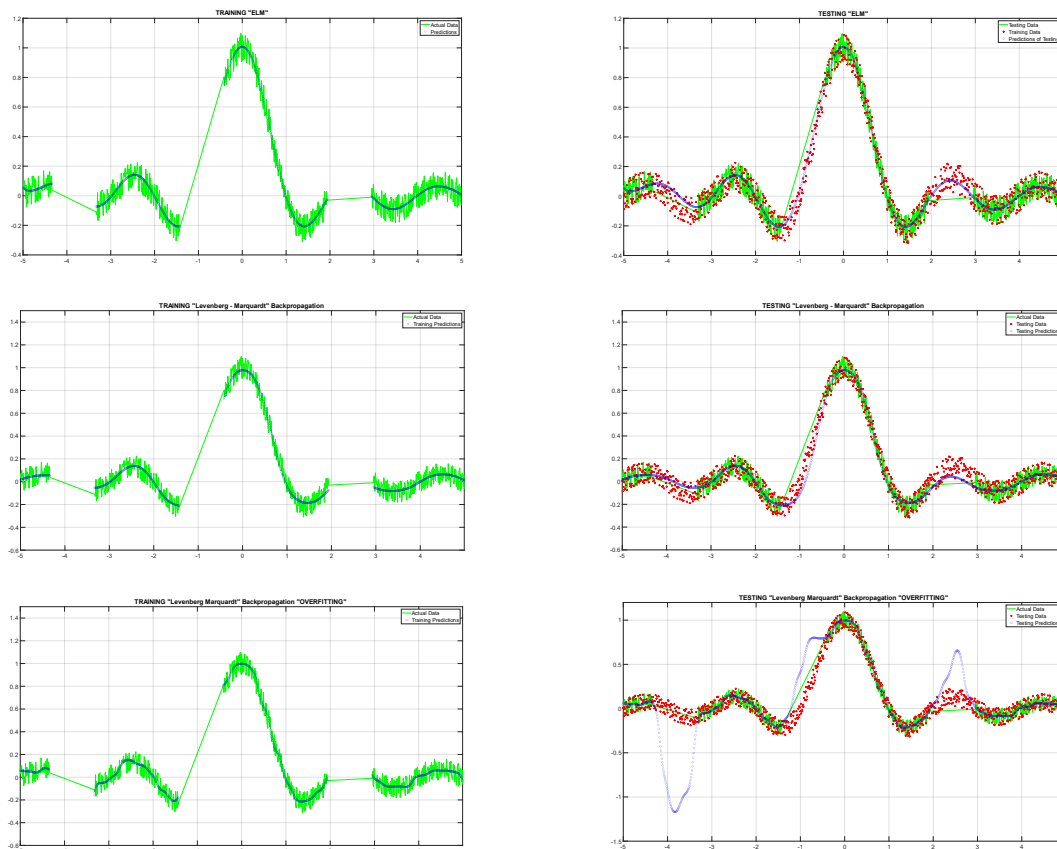
Η τελευταία σύγκριση που κάναμε είναι αυτή στην οποία όχι μόνο έχουμε αποκόψει υποδιαστήματα από το συνολικό πεδίο ορισμού, μέσα από τα οποία δεν υπάρχουν δεδομένα για να εκπαιδευτεί το δίκτυο μέσω των νευρωνικών αλγορίθμων, αλλά έχουμε μολύνει με θόρυβο τα δεδομένα του y ο οποίος είναι ομοίμορφα κατανομημένος και τυχαία επιλεγμένος μέσα από το διάστημα $[-0.1, +0.1]$. Το πεδίο ορισμού είναι το $[-5, +5]$ και λαμβάνουμε όλα τα δεδομένα από το νέο διάστημα που προκύπτει.

Επομένως, εξετάζουμε τα αποτελέσματα των δικτύων στο πειραματικό στάδιο για τα δεδομένα στα υποδιαστήματα που δεν εκπαιδεύτηκε το δίκτυο από τους αλγορίθμους. Να σημειώσουμε πως δεν επιχειρήσαμε προσομοιώσεις με τον Resilient Backpropagation σε αυτή την παραλλαγή, καθώς η αποτελεσματικότητά του δεν ήταν ικανοποιητική από το προηγούμενο πείραμα που εξετάζαμε τα αποκομμένα διαστήματα χωρίς την προσθήκη θορύβου.

| Neurons | | AF | Algorithm | | Neurons | | LR | Algorithm | | Neurons | | LR | Algorithm |
|---------------|---------------|---------|-----------|---------------|---------------|---------|---------|---------------|---------------|---------|---------|-----|-----------|
| 25 | | SIGMOID | ELM | | - | | - | - | | 25 | | 0.2 | BP LM |
| TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS | TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS | TR Time (sec) | TS Time (sec) | RMSE TR | RMSE TS | | |
| 0.0156 | 0 | 0.0580 | 0.0622 | - | - | - | - | 0.3438 | 0.0312 | 0.0604 | 0.0723 | | |

Πίνακας 31 Συγκριτικά Αποτελέσματα BP - ELM με Αποκοπή Υποδιαστημάτων

Από τα παραπάνω η αποτελεσματικότητα του δικτύου είναι παραπλήσια και με τους δύο αλγορίθμους με τον ELM να έχει μία ελαφριά υπεροχή. Αυτό που κάνει τη διαφορά είναι ότι αν προσθέσουμε περισσότερους νευρώνες στο δίκτυο, στην περίπτωση του ELM δε θα πάρουμε πιο βελτιωμένα αποτελέσματα αλλά θα μείνουμε στα ίδια. Αντίθετα με τον Levenberg – Marquardt Backpropagation το δίκτυο θα αποκλίνει από τη λύση και θα εμφανίσει την κατάσταση του **overfitting**. Συνεπώς, ο LM είναι πιο ευάλωτος στην άσκοπη αύξηση του όγκου του δικτύου.



Εικόνα 42 Συγκριτικές Απεικονίσεις BP - ELM με Αποκοπή Υποδιαστημάτων σε Μολυσμένα Δεδομένα

7.3.5 Σύνοψη των αλγορίθμων στο πρόβλημα της Συναρτησιακής Παλινδρόμησης

Από τα πειράματα που διεξήχθησαν και τις συγκρίσεις που κάναμε πάνω στο πρόβλημα συναρτησιακής παλινδρόμησης στη συνάρτηση $y = \sin(x)$, διαπιστώσαμε πως ο ELM μπορεί να ανταπεξέλθει σε άριστο βαθμό. Τις περισσότερες φορές είναι αντίστοιχος στην αποτελεσματικότητά του σε σχέση με το Levenberg – Marquardt και πάντα είναι πιο αποδοτικός στην εκτέλεση των προσομοιώσεων, καθώς δεν έχει επαναληπτική εκπαίδευση στα συναπτικά βάρη.

Ταυτόχρονα, ο Resilient Backpropagation μπορεί να ανταπεξέλθει ικανοποιητικά στα περισσότερα προβλήματα, αλλά η ανοχή του δεν είναι το ίδιο καλή όπως στους άλλους δύο αλγορίθμους. Σε ελλιπή δεδομένα ή σε καταστάσεις που δεν έχει εκπαιδευτεί για κάποια διαστήματα, αντιμετωπίζει σοβαρό πρόβλημα και δεν συγκλίνει εύκολα ο αλγόριθμος. Έχει την αρετή λόγω της δομής του να είναι πιο ταχύς σε σχέση με τον LM αλλά όχι τόσο αποτελεσματικός.

Τέλος, το κομμάτι στο οποίο υπερτερεί ο LM σε σχέση με τους άλλους δύο αλγορίθμους είναι η πιο συμπαγής φύση των δικτύων αναφορικά με τους νευρώνες στο κρυφό στρώμα. Σχεδόν σε όλες τις περιπτώσεις ο LM καταφέρνει να αποδώσει τα καλύτερα αποτελέσματά του με λιγότερους νευρώνες συγκριτικά με τους άλλους δύο αλγορίθμους.

Στο επόμενο κεφάλαιο θα εξετάσουμε την αποτελεσματικότητα και τη γενικευμένη απόδοση των δικτύων από τις προσομοιώσεις των νευρωνικών αλγορίθμων σε προβλήματα ταξινόμησης, τα οποία είναι πιο πρακτικά και κατανοητά σε σχέση με τα συναρτησιακά. Ταυτόχρονα, οι αλγόριθμοι έχουν να επεξεργαστούν μεγαλύτερο όγκο δεδομένων, με το μέγεθος **accuracy**, δηλαδή ακρίβεια να παίζει το σημαντικότερο ρόλο στην αποτελεσματικότητα των προβλέψεων των δικτύων.

Κεφάλαιο 8

Β' Πρακτικό – Πειραματικό σκέλος – Ταξινόμηση

Το β' πρακτικό – πειραματικό σκέλος αυτής τη διπλωματικής αφορά τα αποτίμηση των αλγορίθμων ELM και BP πάνω στα προβλήματα ταξινόμησης. Τα προβλήματα ταξινόμησης είναι αυτά όπου τα εισαγόμενα περιγράφονται με ένα διάνυσμα τιμών – μετρήσεων και σκοπός είναι αντιστοίχισή τους σε μία ή περισσότερες κλάσεις – κατηγορίες. Στις υποενότητες που ακολουθούν θα αναπτύξουμε 4 διαφορετικά προβλήματα ταξινόμησης χρησιμοποιώντας τα νευρωνικά δίκτυα και τους νευρωνικούς αλγορίθμους, για να αποφανθούμε αν ο ELM και εξεταζόμενος νευρωνικός αλγόριθμος είναι το ίδιο αποδοτικός όπως στα προβλήματα συναρτησιακής παλινδρόμησης, όπως είδαμε στο προηγούμενο **Κεφάλαιο 7**, σε σχέση με τον BP.

Θα ξεκινήσουμε τα προβλήματα με το πρόβλημα IRIS το οποίο είναι ένα από τα πιο διαδομένα προβλήματα ταξινόμησης, αν και πολύ μικρό στον όγκο του. Θα ακολουθήσουν δύο προβλήματα για τα οποία έχουμε αποτελέσματα από τα πειράματα του Huang στην επιστημονική εργασία στην οποία πρότεινε τον ELM. Αυτά τα προβλήματα είναι το Image Segment και το Satellite Images, όπου είναι πιο μεγάλα σε όγκο δεδομένων προβλήματα σε σχέση με το IRIS. Και τέλος, θα ολοκληρώσουμε την ξεχωριστή εξέταση του κάθε αλγορίθμου με το πρόβλημα των χειρόγραφων ψηφίων MNIST, προτού συγκρίνουμε τα αποτελέσματά τους και αποτιμήσουμε τις αποδόσεις τους.

8.1 Προσομοιώσεις σε προβλήματα Ταξινόμησης μέσω Backpropagation

8.1.1 Προσομοιώσεις στο πρόβλημα “IRIS” μέσω Backpropagation

Το πρώτο πρόβλημα που εξετάζουμε είναι το IRIS. Το πρόβλημα IRIS αποτελείται από 150 παρατηρήσεις, όπου για την κάθε παρατήρηση την περιγράφουν 4 γνωρίσματα και μπορεί να ενταχθεί σε 3 κλάσεις. Όπως έχουμε αναφέρει και στο **Κεφάλαιο 6**, το γνώρισμα αποτελεί μία μέτρηση για κάθε χαρακτηριστικό μιας παρατήρησης, π.χ. μέτρηση μήκους του πέταλου σε ένα λουλούδι, όπου το σύνολο των γνωρισμάτων περιγράφουν την παρατήρηση η οποία μπορεί να κατηγοριοποιηθεί σε μία κλάση – κατηγορία. Η κατηγορία περιγράφεται με δυαδικό διάνυσμα το οποίο έχει τόσα bits όσα το πλήθος των κατηγοριών – κλάσεων όλων των παρατηρήσεων. Η έξοδος η οποία τοποθετείται η κάθε παρατήρηση είναι αυτή όπου υπάρχει το bit με το ψηφίο 1 στην αντίστοιχη θέση, ενώ όλα τα άλλα bits είναι μηδενικά. Στο IRIS υπάρχουν όπως προαναφέρθηκε 3 κατηγορίες, συνεπώς οι έξοδοι περιγράφονται ως **001**, **010** και **100**.

Για το συγκεκριμένο πρόβλημα επιλέξαμε να χωρίσουμε τις παρατηρήσεις σε δύο ξεχωριστά ποσοστά αναλογίας τα σύνολα για εκπαίδευση και εξέταση. Η μία αναλογία είναι **70/30** και η δεύτερη **50/50** σύνολο **εκπαίδευσης/εξέτασης** αντίστοιχα. Επειδή η βάση παρατηρήσεων είναι μικρή επαναλάβαμε την κάθε ξεχωριστή προσομοίωση (αλγόριθμος – νευρώνες – ρυθμός εκμάθησης) για 10 φορές και λάβαμε τον μέσο όρο από αυτές τις φορές σαν τελική μέτρηση για το κάθε εξεταζόμενο μέγεθος. Οι αλγόριθμοι του Backpropagation που τρέξαμε είναι ο **Levenberg – Marquardt**, ο **Resilient Backpropagation** και ο **Scaled Conjugate Gradient**. Τέλος, οι παρατηρήσεις σε όλα τα προβλήματα έχουν επιλεγεί με **τυχαίο** αλλά και **αμερόληπτο τρόπο**, που σημαίνει ότι ανάλογα το ποσοστό που αντιστοιχεί στο κάθε σύνολο λαμβάνεται τυχαία το αντίστοιχο ποσοστό από τις συνολικές

παρατηρήσεις της κάθε κατηγορίας. Με αυτό τον τρόπο όλες οι κατηγορίες θα έχουν παρατηρήσεις και στα δύο σύνολο, αλλά ταυτόχρονα θα είναι αναλογικά επιλεγμένες ώστε να υπάρχει πλήρης ισορροπία και δικαιοσύνη.

Στους πίνακες που θα ακολουθήσουμε ο τίτλος **ALG** υποδηλώνει τον κάθε αλγόριθμο με το **SCG** να αντιπροσωπεύει τον Scaled Conjugate Gradient, το **Rprop** να αναφέρεται στον Resilient Backpropagation και τον **LM** στον Levenberg – Marquardt. Επιπρόσθετα η παράμετρος Neuron αφορά το πλήθος των νευρώνων στο κρυφό στρώμα και το **LR** να υποδηλώνει τον **ρυθμό εκμάθησης**. Οι αναφορές που γίνονται στο **Tr.** και το **Ts.** σχετίζονται με το **Training** και **Testing** αντίστοιχα. Τέλος, το **RMSE** αντιπροσωπεύει το **σφάλμα ρίζας**, το **Acc. %** το ποσοστό ακρίβειας του αλγορίθμου για το εκάστοτε στάδιο και τέλος το **Full Time** τον πλήρη χρόνο για όλες τις προσομοιώσεις που έγιναν από το δίκτυο.

8.1.1.1 Παραλλαγή 1.1: Αναλογία συνόλων 70/30

Ξεκινώντας την αποτίμηση της πρώτης παραλλαγής δηλώνεται ότι ο **ρυθμός εκμάθησης** είναι **0.01** και **0.1**, οι **βαθμίδες νευρώνων** που εξετάσαμε το πρόβλημα είναι οι **10, 20** και **40** και οι αλγόριθμοι είναι οι **SCG, Rprop** και **LM**. Τα εξεταζόμενα μεγέθη για την αποτελεσματικότητα είναι το **RMSE** και το **Accuracy %** δηλαδή το μέσο τετραγωνικό σφάλμα ρίζας και το ποσοστό ακρίβειας στις προβλέψεις του δικτύου τόσο στο training όσο και στο testing, αλλά και ο σχετικός χρόνος που χρειάστηκε το δίκτυο για την περάτωση μίας ή όλων των προσομοιώσεων.

Στον παρακάτω πίνακα συγκεντρώνονται τα αποτελέσματα δυνατά αποτελέσματα του δικτύου βάση του πλήθους των νευρώνων για τον κάθε αλγόριθμο, πάντα με γνώμονα ως **πιο αξιόλογη προσπάθεια αυτή με το καλύτερο ποσοστό ακρίβειας στο testing**. Ο λόγος που επιλέγεται αυτό το ποσοστό είναι γιατί θεωρείται και είναι πιο αποτελεσματικό ένα δίκτυο το οποίο μετά την εκπαίδευσή του μπορεί, να προσαρμοστεί και να προβλέψει νέα δεδομένα τα οποία είναι άγνωστα προς αυτό.

| 1.1 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|------------------------|-------------------|----------------|------------|-------------------|----------------|------------|-------------------|----------------|------------|
| | SCG | 10 | 0.1 | Rprop | 20 | 0.1 | LM | 10 | 0.1 |
| Tr. Time (sec) | 0.5354 | | | 0.5171 | | | 0.5153 | | |
| Tr. RMSE | 0.0607 | | | 0.0599 | | | 0.0599 | | |
| Tr. Acc. % | 92.29% | | | 92.38% | | | 92.38% | | |
| Ts. Time (sec) | 0.0046 | | | 0.0045 | | | 0.0044 | | |
| Ts. RMSE | 0.0345 | | | 0.0435 | | | 0.0490 | | |
| Ts. Acc. % | 95.78% | | | 94.67% | | | 94.00% | | |
| Full Time (sec) | 5.4002 (x10 reps) | | | 5.2156 (x10 reps) | | | 5.1970 (x10 reps) | | |

Πίνακας 32 1.1 Αποτελέσματα BP - IRIS - 70/30

Από τον παραπάνω πίνακα διαπιστώνουμε ότι ο καλύτερος αλγόριθμος του Backpropagation γι' αυτή την παραλλαγή είναι ο **Scaled Conjugate Gradient**, καθώς έχει το καλύτερο ποσοστό στην εξέταση κι από τους τρεις αλγορίθμους. Μάλιστα αυτό το καταφέρνει με **10 νευρώνες**. Αν αυξήσουμε το πλήθος των νευρώνων το δίκτυο ξεκινάει να φθίνει σε ποσοστά επιτυχίας. Χρονικά και οι τρεις αλγόριθμοι

έχουν την ίδια απόδοση, ενώ με τον **ρυθμό εκμάθησης = 0.1** αποκτούν καλύτερη γενικευμένη απόδοση σε σχέση με τον 0.01.

8.1.1.2 Παραλλαγή 1.2: Αναλογία συνόλων 50/50

Στην επόμενη παραλλαγή επιλέξαμε να χωρίσουμε τα σύνολα με ίσο πλήθος στοιχείων, ώστε να εξετάσουμε το δίκτυο αν μπορεί να ανταπεξέλθει έχοντας εκπαιδευτεί με λιγότερα δεδομένα σε σχέση με την πρώτη παραλλαγή. Επαναλάβαμε με τον ίδιο τρόπο και τις ίδιες παραμέτρους τις προσομοιώσεις όπου τα αποτελέσματα παρουσιάζονται στον παρακάτω πίνακα:

| 1.2 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|-----------------|-------------------|---------|-----|-------------------|---------|-----|-------------------|---------|------|
| | SCG | 10 | 0.1 | Rprop | 20 | 0.1 | LM | 20 | 0.01 |
| Tr. Time (sec) | 0.5329 | | | 0.5236 | | | 0.5255 | | |
| Tr. RMSE | 0.0457 | | | 0.0588 | | | 0.0446 | | |
| Tr. Acc. % | 94.40% | | | 92.80% | | | 94.53% | | |
| Ts. Time (sec) | 0.0051 | | | 0.0044 | | | 0.0047 | | |
| Ts. RMSE | 0.0381 | | | 0.0392 | | | 0.0446 | | |
| Ts. Acc. % | 95.33% | | | 95.20% | | | 94.53% | | |
| Full Time (sec) | 5.3799 (x10 reps) | | | 5.2791 (x10 reps) | | | 5.3021 (x10 reps) | | |

Πίνακας 33 1.2 Αποτελέσματα BP - IRIS - 50/50

Όπως φαίνεται από τον παραπάνω πίνακα οι αλγόριθμοι είναι όλοι ισάξιοι όσον αφορά το χρονικό διάστημα για την ολοκλήρωση μίας αλλά και συνολικά όλων των προσομοιώσεων. **Ο πιο αποτελεσματικός και αποδοτικός συνάμα αλγόριθμος** είναι και εδώ ο **Scaled Conjugate Gradient** αφού καταφέρνει το καλύτερο δυνατό ποσοστό στο testing σε σχέση με τους άλλους δύο αλγορίθμους **95.33%**, και μάλιστα με τους μίσους **νευρώνες** στο κρυφό στρώμα με **10**. Για ελάχιστα το καλύτερο ποσοστό ακρίβειας το λαμβάνει το δίκτυο για την εκπαίδευση από τον Levenberg – Marquardt που είναι ίσο με 94.53%.

8.1.1.3 Σύνοψη IRIS για τον BP

Από τις παραπάνω μετρήσεις που παρουσιάζουμε στους πίνακες, φαίνεται ότι ο πιο αποδοτικός αλλά και αποτελεσματικός αλγόριθμος από τις τρεις παραλλαγές του BP είναι ο Scaled Conjugate Gradient. Καταφέρνει καλύτερα ποσοστά επιτυχίας από όλους τους αλγορίθμους και με πιο συμπαγές δίκτυο σε σχέση με αυτούς. Να επισημάνουμε δε ότι κάθε προσομοίωση με τις παραμέτρους της επαναλήφθηκε για **10 φορές**, για να καταφτάσουμε στο τελικό αποτέλεσμα του εκάστοτε εξεταζόμενου μεγέθους. Συνεπώς, οι μετρήσεις που παρουσιάζονται στον πίνακα είναι ο Μ.Ο. απ' όλες τις επαναλήψεις. Ταυτόχρονα για το RMSE κυμαίνεται κάτω από **0.1** το οποίο επεκτείνεται μέσα από τα ποσοστά ακρίβειας που παρουσιάζονται. Τα ποσοστά ακρίβειας αποτελούν τον πιο ενδεδειγμένο τρόπο για την αποτελεσματικότητα του δικτύου κι αυτό γιατί συγκρίνεται η έξοδος – κατηγορία της παρατήρησης του δικτύου που εξέτασε και πρόβλεψε, σε σύγκριση με την πραγματική έξοδο – στόχο που έπρεπε να προβλέψει το δίκτυο. Τέλος, από τις προσομοιώσεις παρατηρείται ότι ο BP σε όλες τις παραλλαγές γι' αυτό το πρόβλημα ταξινόμησης, αν δεχθεί μεγαλύτερο αριθμό κρυφών νευρώνων από αυτών που είναι

ο βέλτιστος για να αποδώσει, οδεύει στην φθίνουσα απόδοση και εν τέλει στην απόκλιση από τα πραγματικά δεδομένα. Συνεπώς, είναι επιρρεπής σε **overfitting**.

8.1.2 Προσομοιώσεις στο πρόβλημα “IMAGE SEGMENT” μέσω Backpropagation

Σε αυτή την ενότητα εξετάζουμε το δεύτερο πρόβλημα ταξινόμησης, και πρώτο από τα δύο πρόβλημα σε σχέση με αυτά που επαναλάβαμε από την επιστημονική εργασία του Huang, γνωστό ως **Image Segment** ή **Image Segmentation**. Όπως αναφέραμε στο **Κεφάλαιο 6** πρόκειται για μία βάση δεδομένων με **2310 παρατηρήσεις**, οι οποίες δημιουργούνται από **19 ξεχωριστές μετρήσεις – γνωρίσματα** και κατηγοριοποιούνται σε **7 κλάσεις – κατηγορίες**. Αυτές οι παρατηρήσεις έχουν προέλθει από εικόνες που έχουμε τις έχουμε χωρίσει σε υπό-εικόνες – δείγματα και αντιπροσωπεύουν ένα από τα 7 αντικείμενα – κατηγορίες που απεικονίζουν.

Όπως κάναμε και στο προηγούμενο παράδειγμα χωρίζουμε με βάση το ποσοστό αναλογίας του κάθε συνόλου τις παρατηρήσεις από κάθε κατηγορία, με απώτερο σκοπό να πάρουμε με τυχαίο τρόπο τις παρατηρήσεις για κάθε σύνολο δεδομένων αλλά να διαθέτει το κάθε σύνολο δείγματα από όλες τις παρατηρήσεις. Φτιάξαμε 3 διαφορετικές αναλογίες ποσοστών **80/20**, **65/35** και **50/50** και εξετάσαμε τους αλγορίθμους. Η αναλογία **65/35** είναι η αντίστοιχη που είχε επιλέξει να παρουσιάσει και ο **Huang**, όπου στη σύγκριση των αποτελεσμάτων στην τελική αποτίμηση θα την παρουσιάσουμε. Ανάλογα και με το προηγούμενο πρόβλημα ταξινόμησης επαναλάβαμε κάθε προσομοίωση από **10 φορές** και πήραμε τον Μ.Ο. των αποτελεσμάτων για τις τελικές μετρήσεις των μεγεθών αξιολόγησης.

8.1.2.1 Παραλλαγή 2.1: Αναλογία συνόλων 80/20

Η πρώτη απόπειρα που κάναμε είχε ως ποσοστό αναλογίας από τις παρατηρήσεις για το **σύνολο εκπαίδευσης το 80% και το 20%** για το **σύνολο εξέτασης** από το σύνολο των παρατηρήσεων. Οι προσομοιώσεις έγιναν με **0.1** και **0.01** ρυθμό εκμάθησης αλλά και για **30 – 70 – 100** νευρώνες στο κρυφό στρώμα. Όπως θα δούμε από την αύξηση των νευρώνων περισσότερο επηρεάζεται ο **Levenberg – Marquardt** χρονικά σε σχέση με τους άλλους δύο αλγορίθμους. Στον παρακάτω πίνακα θα παρουσιάσουμε τις καλύτερες προσπάθειες του κάθε αλγορίθμου βάσει των παραπάνω δεδομένων που θέσαμε:

| 2.1 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|----------------|------------------------|---------|------|------------------------|---------|-----|---------------------|---------|------|
| | SCG | 100 | 0.01 | Rprop | 100 | 0.1 | LM | 70 | 0.01 |
| Tr. Time (sec) | 3.8819 sec | | | 3.2550 sec | | | 14.4862 sec | | |
| Tr. RMSE | 0.0382 | | | 0.0390 | | | 0.0193 | | |
| Tr. Acc. % | 92.85% | | | 92.69% | | | 96.40% | | |
| Ts. Time (sec) | 0.0060 | | | 0.0060 | | | 0.0054 | | |
| Ts. RMSE | 0.0574 | | | 0.0595 | | | 0.0429 | | |
| Ts. Acc. % | 89.26% | | | 88.87% | | | 91.97% | | |
| Full Time | 38.8795 sec (x10 reps) | | | 32.6105 sec (x10 reps) | | | 00:02:24 (x10 reps) | | |

Πίνακας 34 2.1 Αποτελέσματα BP – IMAGE SEGMENT - 80/20

Από τις παραπάνω προσομοιώσεις προκύπτει πως η παραλλαγή του αλγορίθμου **Levenberg – Marquardt**, είναι η καλύτερη από τις τρεις περιπτώσεις σχετικά με τα ποσοστά αποτελεσματικότητας στην ακρίβεια. Μάλιστα αυτό το επιτυγχάνει με λιγότερους νευρώνες στο κρυφό στρώμα σε σχέση με τις άλλες δύο περιπτώσεις, αφού χρειάζεται **70 νευρώνες και οι άλλες περιπτώσεις θέλουν 100**. Όμως λόγω του γεγονότος ότι χρειάζεται **0.01 ρυθμό εκπαίδευσης** για να το καταφέρει, και επειδή η δομή του αλγορίθμου τον κάνει πιο χρονοβόρο από τους άλλους δύο, **χρονικά υστερεί** έχοντας ανάγκη **2:24 λεπτά** για να καταφέρει να επαναλάβει 10 φορές τις προσομοιώσεις από ένα σύνολο παραμέτρων, τη στιγμή που **οι άλλοι δύο αλγόριθμοι χρειάζονται περίπου 40”**.

8.1.2.2 Παραλλαγή 2.2: Αναλογία συνόλων 65/35

Σε αυτή την παραλλαγή μειώσαμε τα διαθέσιμα δεδομένα για εκπαίδευση, ώστε να δυσκολέψουμε τους αλγορίθμους στην εκπαίδευση των δικτύων. Οι ρυθμοί εκμάθησης στις προσομοιώσεις δεν άλλαξαν αλλά αυξήθηκαν οι νευρώνες με τις κλίμακες να διαμορφώνονται σε **50 – 100 – 130**. Όπως και στο προηγούμενο παράδειγμα έτσι και εδώ επαναλάβαμε τις δοκιμές για 10 φορές για κάθε σετ παραμέτρων και πήραμε των μέσω όρο αυτών. Στον παρακάτω πίνακα παρουσιάζουμε τις καλύτερες δυνατές προσεγγίσεις του κάθε αλγορίθμου:

| 2.2 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|-----------------|------------------------|---------|------|------------------------|---------|-----|------------------------|---------|-----|
| | SCG | 50 | 0.01 | Rprop | 50 | 0.1 | LM | 50 | 0.1 |
| Tr. Time (sec) | 1.8424 sec | | | 1.6974 sec | | | 5.8550 sec | | |
| Tr. RMSE | 0.0477 | | | 0.0458 | | | 0.0260 | | |
| Tr. Acc. % | 91.07% | | | 91.44% | | | 95.13% | | |
| Ts. Time (sec) | 0.0061 | | | 0.0061 | | | 0.0063 | | |
| Ts. RMSE | 0.0659 | | | 0.0652 | | | 0.0521 | | |
| Ts. Acc. % | 87.67% | | | 87.80% | | | 90.26% | | |
| Full Time (sec) | 18.4845 sec (x10 reps) | | | 17.0348 sec (x10 reps) | | | 58.6133 sec (x10 reps) | | |

Πίνακας 35 2.2 Αποτελέσματα BP – IMAGE SEGMENT - 65/35

Σε αυτή την περίπτωση παρατηρούμε ότι όλοι οι αλγόριθμοι έχουν καταφέρει τις καλύτερες προσεγγίσεις για το δίκτυο με **50 νευρώνες** στο κρυφό στρώμα. Τα ποσοστά επιτυχίας των αλγορίθμων είναι πολύ κοντά τόσο στο training όσο και στο testing, όπως και τα RMSE. Όμως μέσω του αλγορίθμου **Levenberg – Marquardt** το δίκτυο έστω και για λίγο έχει καλύτερη γενίκευση σε σχέση με τους άλλους δύο αλγόριθμους. Χρονικά ωστόσο ο **Levenberg – Marquardt** είναι πιο αργός σε σχέση με τους άλλους δύο καθώς χρειάζεται έως και **6 λεπτά** αν τρέξουμε προσομοιώσεις με **130 νευρώνες στο κρυφό στρώμα**. Οι **Scaled Conjugate Gradient** αλλά και **Resilient Backpropagation** δεν ξεπερνούν το **μισό λεπτό** για να ολοκληρώσουν 10 προσομοιώσεις σε καμία από τις δοκιμές που κάναμε.

8.1.2.3 Παραλλαγή 2.3: Αναλογία συνόλων 50/50

Στην τελευταία παραλλαγή που δημιουργήσαμε γι' αυτό το πρόβλημα, κάναμε προσομοιώσεις με το δίκτυο να λαμβάνει ίσα σύνολα δεδομένων και στα δύο στάδια προσομοίωσης. Μειώνοντας το πλήθος των δεδομένων στο testing αυξήσαμε παράλληλα και την **κλίμακα των νευρώνων** σε **70 – 100 – 150**.

Με αυτόν τον τρόπο δώσαμε στους νευρώνες και κατ' επέκταση στο δίκτυο περισσότερους πόρους ώστε να μπορέσει να επεξεργαστεί τα πιο περιορισμένα δεδομένα εκπαίδευσης. Όπως και πριν παρουσιάζουμε τις καλύτερες προσπάθειες των νευρωνικών αλγορίθμων του BP στον παρακάτω πίνακα:

| 2.3 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|----------------|------------------------|---------|------|------------------------|---------|-----|---------------------|---------|------|
| | SCG | 100 | 0.01 | Rprop | 70 | 0.1 | LM | 100 | 0.01 |
| Tr. Time (sec) | 1.4916 sec | | | 1.0130 sec | | | 7.0227 sec | | |
| Tr. RMSE | 0.0502 | | | 0.0504 | | | 0.0261 | | |
| Tr. Acc. % | 90.61% | | | 90.57% | | | 95.13% | | |
| Ts. Time (sec) | 0.0073 | | | 0.0064 | | | 0.0066 | | |
| Ts. RMSE | 0.0692 | | | 0.0634 | | | 0.0553 | | |
| Ts. Acc. % | 87.05% | | | 88.15% | | | 89.66% | | |
| Full Time | 14.9884 sec (x10 reps) | | | 10.1932 sec (x10 reps) | | | 00:01:10 (x10 reps) | | |

Πίνακας 36 2.3 Αποτελέσματα BP – IMAGE SEGMENT - 50/50

Όπως προκύπτει από τον παραπάνω πίνακα, ο Resilient Backpropagation χρειάζεται λιγότερους νευρώνες για να εκτελέσει τις καλύτερες δυνατές προσομοιώσεις, με τους άλλους δύο να χρειάζονται **100 νευρώνες σε σχέση με τους 70 του Resilient Backpropagation**. Όμως ο **Levenberg – Marquardt** καταφέρνει να εκπαιδεύσει καλύτερα το δίκτυο ώστε να προσομοιώνει σε νέα δεδομένα, κι αυτό αφού φτάνει σε **89.66% επιτυχία ακρίβειας στο testing**. Ο Resilient Backpropagation είναι κατά 1.5% περίπου λιγότερο ακριβής φτάνοντας στο **88.15%**, ενώ τελευταίος με περίπου 2.5% λιγότερο ακριβείς προσπάθειες είναι ο Scaled Conjugate Gradient σε σχέση με τον Levenberg – Marquardt.

Τέλος, **χρονικά ο Levenberg – Marquardt** και εδώ είναι πιο αργός σε σχέση με τους άλλους δύο νευρωνικούς αλγόριθμους του BP, αφού μπορεί να χρειαστεί ως και **3 λεπτά** για να καταφέρει να ολοκληρώσει 10 προσομοιώσεις σε σχέση με τα κάτω από **20'' που χρειάζονται οι άλλοι δύο**.

8.1.2.4 Σύνοψη Image Segment για τον BP

Από τις παραπάνω προσομοιώσεις που κάναμε για τις 3 διαφορετικές αναλογίες συνόλων των παρατηρήσεων για τις παραλλαγές του BP, προκύπτει ότι σε αυτό το πρόβλημα ταξινόμησης πιο αποτελεσματικός είναι ο **Levenberg – Marquardt**, καθώς καταφέρνει καλύτερα ποσοστά επιτυχίας σε σχέση με τους άλλους δύο αλγόριθμους στις προβλέψεις που κάνει στο testing στάδιο στα νέα δεδομένα, ύστερα από την εκπαίδευση του.

Μολονότι είναι ο πιο αποτελεσματικός δεν είναι ο πιο αποδοτικός σε σχέση με τον όγκο του δικτύου, τους πόρους και τον χρόνο εκτέλεσης των προσομοιώσεων, κι αυτό γιατί χρειάζεται έως και 15 φορές (!) περισσότερο χρόνο για να ολοκληρώσει μία προσομοίωση σε σχέση με τον Scaled Conjugate Gradient και τον Resilient Backpropagation.

Τέλος, στο συγκεκριμένο πρόβλημα με την αύξηση των νευρώνων δε φαίνεται προκύπτει άμεσα πρόβλημα με overfitting, κάτι το οποίο δίνει περισσότερη ανοχή στο δίκτυο.

8.1.3 Προσομοιώσεις στο πρόβλημα “SATELLITE IMAGES” μέσω Backpropagation

Το **Satellite Images** είναι ένα πρόβλημα το οποίο αποτελείται από εικόνες που έχουν τραβηχτεί από δορυφόρο (για περισσότερη ανάλυση βλ. **Κεφάλαιο 6, Ενότητα 6.2.3**). Η γενική φιλοσοφία του προβλήματος είναι η τμηματοποίηση των αρχικών εικόνων σε πιο μικρές ώστε να δημιουργηθούν δείγματα. Αυτά τα δείγματα περιέχουν ένα είδος μορφολογίας τους εδάφους που αντιπροσωπεύει μία κατηγορία. Το κάθε δείγμα για να μπορέσει να περιγραφεί χρησιμοποιούνται κάποιες σχετικές μετρήσεις που το περιγράφουν. **Οι μετρήσεις – γνωρίσματα των δειγμάτων – παρατηρήσεων είναι 36 και οι κατηγορίες κλάσεις που μπορεί να ανήκουν οι παρατηρήσεις είναι 7. Μολονότι είναι 7 οι 6 κατηγορίες περιλαμβάνονται στις παρατηρήσεις**, καθώς η 7^η αποτελεί συνδυασμό όλων των άλλων κάτι το οποίο δε συμβαίνει σε καμία υπό-εικόνα – δείγμα από τις παρατηρήσεις μας.

Αυτό το πρόβλημα είναι το 3^ο κατά σειρά που εξετάζουμε και το 2^ο από που επαναλαμβάνουμε, καθώς το υπάρχει στην επιστημονική εργασία του Huang. Το πλήθος των παρατηρήσεων είναι αυξημένο σε σχέση με τα προηγούμενα δύο προβλήματα, καθώς πλέον μιλάμε για **6430 παρατηρήσεις**. Αυτές τις παρατηρήσεις θα τις χωρίσουμε σε τρεις διαφορετικές αναλογίες με τυχαίο τρόπο αλλά ομοιόμορφα καταναμημένα τα ποσοστά των συνόλων. Οι αναλογίες είναι **80/20, 68/32 και 50/50 σε training dataset / testing dataset**, όπου από την κάθε κατηγορία λαμβάνουμε το αντίστοιχο ποσοστό, ανάλογα για ποιο σύνολο δεδομένων μιλάμε, από το πλήθος των παρατηρήσεων. Έτσι αν θέλουμε να φτιάξουμε 50/50 σύνολα δεδομένων θα πάρουμε το 50% από τις παρατηρήσεις της κάθε κατηγορίας, συνεπώς στο τέλος θα έχουμε το 50% από όλες τις παρατηρήσεις όλων των κατηγοριών και ομοιόμορφα διαχωρισμένο. Το 68/32 αποτελεί την αναλογία που χώρισε τις παρατηρήσεις του και ο Huang.

Τέλος, επισημαίνεται πως και σε αυτό το πρόβλημα κάθε ξεχωριστό σύνολο παραμέτρων εκτελείται για 10 φορές και λαμβάνεται ο Μ.Ο. των μετρήσεων για κάθε μέγεθος.

8.1.3.1 Παραλλαγή 3.1: Αναλογία συνόλων 80/20

Σε αυτή την παραλλαγή ο ρυθμός εκμάθησης είναι ο αντίστοιχος με τα προηγούμενα προβλήματα, δηλαδή 0.01 και 0.1. Οι κλιμάκωση στους νευρώνες ξεκινάει με 30, αυξάνεται στους 70 και καταλήγει στους 100. Μέσω αυτών των παραμέτρων εκτελούμε τις προσομοιώσεις για τους 3 διαφορετικούς αλγορίθμους του Backpropagation, συγκεντρώνοντας τις καλύτερες προσομοιώσεις για τον καθένα στον παρακάτω πίνακα:

| 3.1 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|----------------|---------------------|---------|-----|---------------------|---------|-----|--------------------|---------|-----|
| | SCG | 70 | 0.1 | Rprop | 70 | 0.1 | LM | 70 | 0.1 |
| Tr. Time | 9.1603 sec | | | 7.7391 sec | | | 00:02:59 | | |
| Tr. RMSE | 0.0782 | | | 0.0812 | | | 0.0721 | | |
| Tr. Acc. % | 86.46% | | | 85.94% | | | 87.52% | | |
| Ts. Time (sec) | 0.0075 | | | 0.0074 | | | 0.0070 | | |
| Ts. RMSE | 0.0845 | | | 0.0878 | | | 0.0845 | | |
| Ts. Acc. % | 85.37% | | | 84.79% | | | 85.37% | | |
| Full Time | 00:01:31 (x10 reps) | | | 00:01:17 (x10 reps) | | | 00:15:58 (x5 reps) | | |

Πίνακας 37 3.1 Αποτελέσματα BP – SATELLITE IMAGES - 80/20

Από τις μετρήσεις που φαίνονται στον παραπάνω πίνακα, οι αλγόριθμοι στη συγκεκριμένη παραλλαγή αποδίδουν όλοι καλύτερα με **70 νευρώνες στο κρυφό στρώμα**. Από άποψη αποτελεσματικότητας ο Resilient Backpropagation **υστερεί κατά 0.58% σε σχέση** με τους άλλους δύο, οι οποίοι καταφέρνουν **ποσοστό ακρίβειας** στις πειραματικές προσομοιώσεις **85.37%**. Ο **Levenberg – Marquardt** καταφέρνει 87.52% ποσοστό ακρίβειας στις προβλέψεις κάτι το οποίο τον **κάνει 1% – 1.5% πιο αποτελεσματικό στην εκπαίδευση σε σχέση με τους άλλους δύο**.

Αυτό όμως που κάνει τον Scaled Conjugate Gradient πιο ελκυστική και βέλτιστη κατά κανόνα επιλογή, είναι το γεγονός ότι καταφέρνει να εκτελέσει μία προσομοίωση σε **10 δευτερόλεπτα τη στιγμή που ο Levenberg – Marquardt χρειάζεται 3 περίπου λεπτά**, που τον κάνει κάτι παραπάνω από **18 (!) φορές πιο αργό**.

Να διευκρινίσουμε όμως ότι όλοι οι αλγόριθμοι είναι σε επίπεδο αποτελεσματικότητας πολύ κοντά και ότι δεν έχουν κάποια θεματική βελτίωση με την αύξηση των νευρώνων στο κρυφό στρώμα, ούτε κάποια απόκλιση.

8.1.3.2 Παραλλαγή 3.2: Αναλογία συνόλων 68/32

Η δεύτερη παραλλαγή του προβλήματος έχει ως χαρακτηριστικό την αντίστοιχη αναλογία των συνόλων με αυτή που επέλεξε ο Huang. Με άλλα λόγια οι παρατηρήσεις που έχει το κάθε σύνολο για τις προσομοιώσεις, είναι σε πλήθος παρατηρήσεων αντίστοιχο με αυτό που είχε επιλέξει ο Huang στα datasets του. Οι ρυθμοί εκμάθησης σας παράμετροι δεν έχουν αλλάξει σε σχέση με την προηγούμενη παραλλαγή όμως έχουν αλλάξει οι νευρώνες στην κλίμακα, αφού πλέον ξεκινάμε με 50 νευρώνες τις προσομοιώσεις, προχωράμε στην αύξηση με 100 και καταλήγουμε στα πειράματα με 130 νευρώνες.

Όπως και παραπάνω τα αποτελέσματα του δικτύου για κάθε μία από τους καλύτερους συνδυασμούς παραμέτρων για τον κάθε αλγόριθμο του BP παρουσιάζονται στον παρακάτω πίνακα, προτού ερμηνεύσουμε πιο συγκεκριμένα τις μετρήσεις:

| 3.2 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|----------------|------------------------|---------|-----|------------------------|---------|-----|--------------------|---------|-----|
| | SCG | 100 | 0.1 | Rprop | 100 | 0.1 | LM | 100 | 0.1 |
| Tr. Time | 5.9215 sec | | | 5.4311 sec | | | 00:02:18 | | |
| Tr. RMSE | 0.0791 | | | 0.0819 | | | 0.0723 | | |
| Tr. Acc. % | 86.29% | | | 85.82% | | | 87.48% | | |
| Ts. Time (sec) | 0.0092 | | | 0.0088 | | | 0.0085 | | |
| Ts. RMSE | 0.0878 | | | 0.0912 | | | 0.0861 | | |
| Ts. Acc. % | 84.80% | | | 84.21% | | | 85.08% | | |
| Full Time | 59.3070 sec (x10 reps) | | | 54.3988 sec (x10 reps) | | | 00:11:33 (x5 reps) | | |

Πίνακας 38 3.2 Αποτελέσματα BP – SATELLITE IMAGES - 68/32

Από τον παραπάνω πίνακα γίνεται εμφανές ότι η απόδοση των τριών αλγορίθμων είναι παρόμοια, γιατί για να καταφέρουν τις καλύτερες δυνατές προσομοιώσεις χρησιμοποιούν το ίδιο πλήθος νευρώνων και τον ίδιο ρυθμό εκμάθησης.

Ελάχιστα πιο αποτελεσματικός είναι ο αλγόριθμος του Levenberg – Marquardt ο οποίο στο testing σε νέα δεδομένα καταφέρνει ποσοστό ακρίβειας στις προβλέψεις του **85.08%**. Η ίδια εικόνα φανερώνεται

και στην εκπαίδευση του δικτύου με τον Levenberg – Marquardt να είναι κατά **1% – 1.5%** πιο αποτελεσματικός σε σχέση με τους άλλους δύο.

Εκεί που υπάρχει μεγάλη διαφορά είναι στην χρονική διάρκεια της εκτέλεσης των προσομοιώσεων, με τον **Scaled Conjugate Gradient** και τον **Resilient Backpropagation** να χρειάζονται περίπου **7 δευτερόλεπτα** για να εκτελέσουν μία προσομοίωση ενώ τον **Levenberg – Marquardt** να χρειάζεται **2 λεπτά και 20 δευτερόλεπτα περίπου**. Η διαφορά αυτή είναι χαώδης και μεταφράζεται σε πάνω από **20 (!) φορές** πιο αργό ο ένας αλγόριθμος σε σχέση με τους άλλους δύο.

8.1.3.3 Παραλλαγή 3.3: Αναλογία συνόλων 50/50

Η τελευταία παραλλαγή που τρέξαμε είναι αυτή που και τα δύο σύνολα δεδομένων για την εκπαίδευση και τη εξέταση έχουν ίδιο αριθμό παρατηρήσεων. Συνεπώς, το δίκτυο εκπαιδεύεται με τον ίδιο αριθμό παρατηρήσεων με αυτές που εξετάζεται. Η δεύτερη διαφοροποίηση σε σχέση με την προηγούμενη παραλλαγή είναι ότι αλλάζουμε το πλήθος των νευρώνων για τις προσομοιώσεις μας, καθώς τους αυξάνουμε σε κάθε επίπεδο ξεκινώντας με 70, στη συνέχεια πάμε στους 100 και τέλος προσομοιώνουμε για 150 νευρώνες.

Όπως θα διαπιστώσουμε δεν προκύπτουν ιδιαίτερες διαφοροποιήσεις στην αποτελεσματικότητας του δικτύου και για τους τρεις αλγορίθμους. Όπως θα δούμε και στον ακόλουθο πίνακα, τα ποσοστά ακρίβειας είναι περίπου στο 84% - 86% τόσο στο training όσο και στο testing.

Αυτό που παρατηρείται και εδώ είναι οι πιο αργές προσομοιώσεις του Levenberg – Marquardt σε σχέση με τους άλλου δύο αλγορίθμους, κάτι το οποίο και εδώ μας ανάγκασε αν τρέξουμε τις επαναληπτικές προσομοιώσεις για 5 φορές σε σχέση με τις 10 που τρέχουμε στους άλλους δύο αλγορίθμους.

| 3.3 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|----------------|------------------------|---------|------|------------------------|---------|------|--------------------|---------|-----|
| | SCG | 150 | 0.01 | Rprop | 150 | 0.01 | LM | 150 | 0.1 |
| Tr. Time | 4.3116 sec | | | 3.9662 sec | | | 00:01:37 | | |
| Tr. RMSE | 0.0780 | | | 0.0802 | | | 0.0709 | | |
| Tr. Acc. % | 86.48% | | | 86.11% | | | 87.73% | | |
| Ts. Time (sec) | 0.0111 | | | 0.0120 | | | 0.0111 | | |
| Ts. RMSE | 0.0887 | | | 0.0916 | | | 0.0884 | | |
| Ts. Acc. % | 84.63% | | | 84.13% | | | 84.69% | | |
| Full Time | 43.2273 sec (x10 reps) | | | 39.7816 sec (x10 reps) | | | 00:08:05 (x5 reps) | | |

Πίνακας 39 3.3 Αποτελέσματα BP – SATELLITE IMAGES - 50/50

Στις συγκεκριμένες προσομοιώσεις τις οποίες τρέξαμε για την αναλογία μεγέθους συνόλων δεδομένων 50/50, ότι ο Levenberg – Marquardt είναι αμυδρά και αμελητέα πιο αποτελεσματικός από τον Scaled Conjugate Gradient με **84.69%**. Ο Resilient Backpropagation ακολουθεί πολύ κοντά όντας μόλις **0.50%** λιγότερο αποτελεσματικός. Ταυτόχρονα όμως ο **Resilient** και ο **SCG** χρειάζονται **περίπου 5 δευτερόλεπτα** για την ολοκλήρωση μίας προσομοίωσης, ενώ ο **LM** χρειάζεται περίπου **100 δευτερόλεπτα** κάτι το οποίο τον κάνει **20 φορές (!) πιο αργό**.

8.1.3.4 Σύνοψη Satellite Images για τον BP

Εν κατακλείδι από τις παραπάνω προσομοιώσεις στις 3 διαφορετικές παραλλαγές του προβλήματος, διαπιστώσαμε πως και οι τρεις αλγόριθμοι του BP είναι περίπου το ίδιο αποτελεσματικοί. Η αποτελεσματικότητα που έχουν είναι αντίστοιχη ανεξαρτήτως της αναλογίας του έχουν τα σύνολα των παρατηρήσεων για την εκπαίδευση και την εξέταση του δικτύου.

Με τον τριπλασιασμό των παρατηρήσεων περίπου σε σχέση με τον προηγούμενο πρόβλημα ταξινόμησης Image Segment, η διαφοροποίηση στη διάρκεια εκτέλεσης της κάθε προσομοίωσης είναι πιο εμφανής. Αυτό φαίνεται από το γεγονός ότι ο Levenberg – Marquardt χρειάζεται πολλαπλάσιες φορές περισσότερο χρόνο για να εκτελέσει την αντίστοιχη προσομοίωση σε σχέση με τους άλλους δύο αλγόριθμους του BP. Μάλιστα στον επόμενο πρόβλημα ταξινόμησης (MNIST) που θα εξετάσουμε, θα γίνει ακόμη μεγαλύτερη αυτή η διαφορά στη διάρκεια εκτέλεσης μιας προσομοίωσης, που θα κάνει τον αλγόριθμο Levenberg – Marquardt πρακτικά άσκοπο να τον χρησιμοποιήσουμε, καθώς οι χρόνοι θα είναι δυσανάλογοι από την αποτελεσματικότητά του.

8.1.4 Προσομοιώσεις στο πρόβλημα “MNIST” μέσω Backpropagation

Το τελευταίο πρόβλημα ταξινόμησης που θα εξετάσουμε για τις τρεις παραλλαγές του Backpropagation, αφορά τη βάση δεδομένων MNIST. Όπως εξηγούμε στην *υποενότητα 6.2.4* στο **Κεφάλαιο 6**, η βάση δεδομένων αποτελείται από ένα σύνολο χειρόγραφων αναπαριστάσεων για τα ψηφία 0 – 9, από ένα σύνολο πολλών και διαφορετικών χαρακτήρων. Αυτά τα χειρόγραφα ψηφία ψηφιοποιούνται σε εικόνες γκριζας κλίμακας των 28 x 28 pixels, στις οποίες αναπαρίσταται επακριβώς το χειρόγραφο ψηφίο. Επειδή είναι στην κλίμακα του γκριζου τα pixels που αφορούν το ψηφίο αναπαρίσταται με λευκό χρώμα, το οποίο έχει τιμή 0 ενώ το background έχει χρώμα μαύρο και τιμή 1. Οποιοδήποτε άλλο ψηφίο είναι ανάμεσα σε αυτά τα δύο αναπαρίσταται με κάποιο δεκαδικό νούμερο μεταξύ του (0, 1).

Από τη βάση δεδομένων μπορούμε να πάρουμε 60.000 δείγματα για την εκπαίδευση και 10000 δείγματα για λόγους εξέτασης. Εμείς θα χωρίσουμε σε 3 διαφορετικά επίπεδα τα σύνολα τιμών, όπου το testing dataset θα είναι πάντα το ίδιο με 10000 παρατηρήσεις, ενώ το training dataset θα είναι αρχικά 10000, στη συνέχεια 30000 και τέλος 60000. Ο λόγος για τον οποίο γίνεται αντίθετα η κλιμάκωση του μεγέθους των συνόλων, προκύπτει από το γεγονός ότι όσο μεγαλύτερο είναι το σύνολο παρατηρήσεων του training dataset τόσο πιο πιθανό είναι να βρεθούν ψηφία που είναι αμφιλεγόμενη η κατηγορία που ανήκουν (δηλαδή το πραγματικό ψηφίο που αντιπροσωπεύουν), άρα και πιο πιθανό να υπάρχουν σφάλματα πρόβλεψης.

Τέλος, όπως είναι προφανές οι κατηγορίες είναι 10 που μπορεί να ανήκει μία παρατήρηση, δηλαδή μία για κάθε ψηφίο, ενώ τα γνώρισμα που περιγράφουν την παρατήρηση (μετρήσεις) είναι 768. Αυτό προκύπτει από το γεγονός ότι η εικόνα είναι 28 x 28 pixels, που σημαίνει ότι τοποθετούμε σε ένα διάνυσμα αυτά τα pixels, άρα κάθε γνώρισμα αντιπροσωπεύει μία τιμή ενός pixel.

8.1.4.1 Παραλλαγή 4.1: 10000 training – 10000 testing datasets

Για το πρώτο επίπεδο πλήθους του συνόλου εκπαίδευσης, όπως και για όλα τα επόμενα ο ρυθμός εκμάθησης είναι είτε 0.1 είτε 0.01. Οι νευρώνες στο κρυφό στρώμα είναι 40 στην αρχή, 100 ακολούθως και 200 στο τέλος. Μόνο στη μια περίπτωση προσομοίωσης του Levenberg – Marquardt που εκτελέσαμε χρησιμοποιήσαμε 20 νευρώνες. Μάλιστα η προσομοίωση έτρεξε για μία φορά μόνο καθώς

ο χρόνος εκτέλεσης είναι πολύ μεγάλος με αυτόν τον αλγόριθμο για ένα τόσο μεγάλο σύνολο δεδομένων και γνωρισμάτων. Για τις προσομοιώσεις του δικτύου για τον Scaled Conjugate Gradient και για τον Resilient Backpropagation, επαναλάβαμε 5 φορές την κάθε προσομοίωση λαμβάνοντας τον Μ.Ο. των μεγεθών που εξετάζουμε ώστε να έχουμε καλύτερη αξιολόγηση των αλγορίθμων. Στον παρακάτω πίνακα παρουσιάζουμε τις καλύτερες προσομοιώσεις ως προς τον ρυθμό εκμάθησης αλλά και του πλήθους των νευρώνων, για να αποκτήσει το δίκτυο την καλύτερη προβλεπτική ικανότητα.

| 4.1 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|----------------|--------------------|---------|-----|--------------------|---------|------|-------------------|---------|------|
| | SCG | 200 | 0.1 | Rprop | 200 | 0.01 | LM | 20 | 0.01 |
| Tr. Time | 00:01:08 | | | 00:01:20 | | | 01:42:30 | | |
| Tr. RMSE | 0.0177 | | | 0.0790 | | | 0.0253 | | |
| Tr. Acc. % | 96.04% | | | 82.34% | | | 94.35% | | |
| Ts. Time (sec) | 0.1076 | | | 0.1058 | | | 0.0963 | | |
| Ts. RMSE | 0.0177 | | | 0.0790 | | | 0.0253 | | |
| Ts. Acc. % | 96.04% | | | 82.34% | | | 94.35% | | |
| Full Time | 00:05:41 (x5 reps) | | | 00:06:44 (x5 reps) | | | 01:42:30 (x1 rep) | | |

Πίνακας 40 4.1 Αποτελέσματα BP – MNIST - 10000

Όπως βλέπουμε από τις παραπάνω προσομοιώσεις, ο Resilient Backpropagation είναι πολύ πιο πίσω σε απόδοση στην ακρίβεια των προβλέψεων σε σχέση με τους άλλους δύο αλγορίθμους. Ο **Scaled Conjugate Gradient είναι ο καλύτερος και πιο ακριβής αλγόριθμος** αλλά ταυτόχρονα και ο πιο γρήγορος. Αντίθετα ο Levenberg – Marquardt είναι αποτελεσματικός αλλά πολύ αργός αλγόριθμος, και αυτό φαίνεται από το γεγονός ότι χρειάζεται **100 λεπτά για να ολοκληρώσει** μία πλήρη προσομοίωση, τη στιγμή που οι άλλοι δύο αλγόριθμοι χρειάζονται λιγότερο από **1.5 λεπτό**. Γι' αυτό το λόγο δε μπορεί να επιλεγεί ως ο καλύτερος αλγόριθμος για το συγκεκριμένο παράδειγμα.

8.1.4.2 Παραλλαγή 4.2: 30000 training – 10000 testing datasets

Στη δεύτερη παραλλαγή του προβλήματος αυξήσαμε το πλήθος του training dataset στις 30000 παρατηρήσεις. Επαναλάβαμε τις ίδιες ακριβώς προσομοιώσεις και για τις ίδιες παραμέτρους, προσθέτοντας ένα επίπεδο περισσότερων νευρώνων, τους 300. Στο πίνακα που ακολουθεί, παρουσιάζονται οι καλύτερες εκτελέσεις των αλγορίθμων για το δίκτυο. Να επισημάνουμε πως οι χρόνοι εκπαίδευσης όπως είναι προφανές αυξήθηκαν κατά πολύ σε σχέση με τις προηγούμενες προσομοιώσεις. Είναι χαρακτηριστικό δε ότι η προσομοίωση με τον Levenberg – Marquardt έφτασε τις 6 περίπου ώρες. Μολονότι χρησιμοποιήθηκαν για αυτόν 20 νευρώνες στο κρυφό στρώμα, κατάφερε να ανταποκριθεί. Όμως ο χρόνος ο οποίος απαιτείται είναι κατά πολύ πιο μεγάλος σε σχέση με τους άλλους δύο αλγορίθμους. Ο πιο αποτελεσματικός από όλους τους αλγορίθμους τόσο χρονικά όσο και στην ακρίβεια είναι και πάλι ο αλγόριθμος Scaled Conjugate Gradient. Μάλιστα κατορθώνει αυτές τις επιδόσεις με 300 νευρώνες, τη στιγμή που ο Resilient Backpropagation ξεκινάει να αποκλίνει εάν το δίκτυο αποτελείται από περισσότερους από 200 κρυφούς νευρώνες, άρα να εκδηλώνει το φαινόμενο του **overfitting**. Βέβαια στους 200 νευρώνες κατορθώνει να βελτιώσει τα ποσοστά επιτυχίας τους συγκριτικά με την προηγούμενη παραλλαγή που είχε μικρότερο πλήθος παρατηρήσεων για εκπαίδευση.

| 4.2 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|----------------|--------------------|---------|-----|--------------------|---------|------|-------------------|---------|------|
| | SCG | 300 | 0.1 | Rprop | 200 | 0.01 | LM | 20 | 0.01 |
| Tr. Time | 00:07:23 | | | 00:04:27 | | | 05:54:50 | | |
| Tr. RMSE | 0.0188 | | | 0.0633 | | | 0.0225 | | |
| Tr. Acc. % | 95.79% | | | 85.84% | | | 94.97% | | |
| Ts. Time (sec) | 0.1487 | | | 0.1120 | | | 0.1053 | | |
| Ts. RMSE | 0.0181 | | | 0.0617 | | | 0.0219 | | |
| Ts. Acc. % | 95.95% | | | 86.21% | | | 95.11% | | |
| Full Time | 00:36:56 (x5 reps) | | | 00:22:17 (x5 reps) | | | 05:54:50 (x1 rep) | | |

Πίνακας 41 4.2 Αποτελέσματα BP – MNIST - 30000

8.1.4.3 Παραλλαγή 4.3: 60000 training – 10000 testing datasets

Στην τελευταία παραλλαγή του προβλήματος τρέχουμε τις αντίστοιχες προσομοιώσεις αλλά για ένα σύνολο δεδομένων εκπαίδευσης των 60000 παρατηρήσεων. Εδώ ξεκινάμε από 100 νευρώνες να εκτελούμε τις προσομοιώσεις. Ακολουθούν 200 και 300 νευρώνες στις προσομοιώσεις. Για τον Levenberg – Marquardt δεν εκτελέσαμε κάποια προσομοίωση, κι αυτό γιατί ήδη από τις προηγούμενες παραλλαγές χρειαζόταν να καταναλώσει ένα πολύ μεγάλο χρονικό διάστημα για να ολοκληρώσει μόνο μία προσομοίωση. Ταυτόχρονα η αδυναμία να προσθέσουμε περισσότερους νευρώνες για να διευκολύνουμε τον αλγόριθμο και το δίκτυο κατ' επέκταση, εξαιτίας του πολλαπλάσιου χρόνου που θα χρειαζόταν, οδήγησε στην απόρριψη της επιλογής για προσομοίωση.

Στον πίνακα που ακολουθεί, παρουσιάζονται οι καλύτερες εκτελέσεις, και όπως θα δούμε θα είναι για το ίδιο πλήθος νευρώνων με την προηγούμενη παραλλαγή. Χρονικά όπως είναι εμφανές αυξήθηκαν σε διάρκεια οι προσομοιώσεις και για τους δύο αλγορίθμους, αλλά η αποτελεσματικότητά τους έμεινε στα ίδια επίπεδα.

| 4.3 | ALG | Neurons | LR | ALG | Neurons | LR | ALG | Neurons | LR |
|----------------|--------------------|---------|------|--------------------|---------|-----|-----|---------|----|
| | SCG | 300 | 0.01 | Rprop | 200 | 0.1 | LM | - | - |
| Tr. Time | 00:18:03 | | | 00:09:45 | | | - | | |
| Tr. RMSE | 0.0180 | | | 0.0611 | | | - | | |
| Tr. Acc. % | 95.98% | | | 86.34% | | | - | | |
| Ts. Time (sec) | 0.1448 | | | 0.1125 | | | - | | |
| Ts. RMSE | 0.0174 | | | 0.0597 | | | - | | |
| Ts. Acc. % | 96.12% | | | 86.64% | | | - | | |
| Full Time | 01:30:18 (x3 reps) | | | 00:28:46 (x3 reps) | | | - | | |

Πίνακας 42 4.3 Αποτελέσματα BP – MNIST - 60000

8.1.4.4 Σύνοψη MNIST για τον BP

Όπως προκύπτει από τις παραπάνω προσομοιώσεις που έγιναν και για τις τρεις παραλλαγές του προβλήματος MNIST, συνολικά ο Scaled Conjugate Gradient είναι η καλύτερη παραλλαγή του BP. Αυτό παρατηρείται από το γεγονός ότι μπορεί να επιτύχει καλύτερα αποτελέσματα στην ακρίβεια των προβλέψεων τόσο στην εκπαίδευση όσο και στις πειραματικές προσομοιώσεις.

Επιπρόσθετα, η χρονική διάρκεια των εκτελέσεων είναι πολύ κοντά με αυτή του Resilient Backpropagation ο οποίος είναι ο πιο γρήγορος αλγόριθμος. Βέβαια αν και πιο γρήγορος δεν είναι το ίδιο αποτελεσματικός, καθώς δεν καταφέρνει να ξεπεράσει το 87% επιτυχίας στις προβλέψεις του σε καμία παραλλαγή του προβλήματος.

Ο Scaled Conjugate Gradient, έχει ένα ποσοστό επιτυχίας στο 96% περίπου στις προσομοιώσεις που κάναμε, και το διατηρεί ανεξάρτητα από το σύνολο των δεδομένων εκπαίδευσης, εφόσον λάβει το σωστό πλήθος νευρώνων το δίκτυο που θα τον εφαρμόσουμε.

Τέλος, ο Levenberg – Marquardt παρά το γεγονός ότι με 20 νευρώνες μπορεί να εκπαιδεύσει ένα δίκτυο για προσομοιώνει με ποσοστά επιτυχίας στο 95%, δεν είναι ο κατάλληλος αλγόριθμος για να επιλεγεί καθώς χρειάζεται ένα πολύ μεγάλο χρονικό διάστημα για να περατώσει μόλις μία προσομοίωση.

8.2 Προσομοιώσεις σε προβλήματα Ταξινόμησης μέσω ELM

Σε αυτή την ενότητα θα αποτυπώσουμε τα αποτελέσματα από τις προσομοιώσεις που κάναμε στα προβλήματα ταξινόμησης, χρησιμοποιώντας τον ELM ως νευρωνικό αλγόριθμο εκπαίδευσης. Στις ακόλουθες προσομοιώσεις θα χρησιμοποιήσουμε στη θέση του ρυθμού εκμάθησης που δεν υπάρχει στον ELM, τις συναρτήσεις ενεργοποίηση οι οποίες είναι η **σιγμοειδής**, η **εφαπτομένη** και η **ακτινικής βύσης**. Παράλληλα θα εφαρμόσουμε κλιμακωτή αύξηση στους νευρώνες του δικτύου, για να διαπιστώσουμε αν βελτιώνεται ή φθίνει η απόδοση του δικτύου που δοκιμάζουμε.

Να επισημάνουμε πως τα σύνολα δεδομένων για κάθε παραλλαγή είναι ακριβώς τα ίδια με αυτά που τρέξαμε και στις προσομοιώσεις με τον BP στα αντίστοιχα προβλήματα. Έτσι υπάρχει πλήρη αμεροληψία και δικαιοσύνη καθώς το κάθε δίκτυο εκπαιδεύεται και τεστάρεται με τα ίδια δεδομένα για τους ξεχωριστούς νευρωνικούς αλγορίθμους, κρατώντας πάντα μία παράμετρο σταθερή.

8.2.1 Προσομοιώσεις στο πρόβλημα “IRIS” μέσω ELM

Αρχικά όπως κάναμε και στον Backpropagation θα ξεκινήσουμε με το μικρότερο σε όγκο πρόβλημα από άποψη παρατηρήσεων, γνωρισμάτων αλλά και κατηγοριών, το οποίο είναι το πρόβλημα του φυτού της Ίριδας.

Πάνω σε αυτό θα εκτελέσουμε 2 ξεχωριστές παραλλαγές του που σχετίζονται με την αναλογία στο μέγεθος του κάθε συνόλου. Θα ξεκινήσουμε με 70/30 αναλογία σε training / testing datasets. Κάθε προσομοίωση που διαθέτει ένα ξεχωριστό σύνολο παραμέτρων θα εκτελεστεί για 10 φορές, με απώτερο σκοπό την καλύτερη δυνατή προσέγγιση των αποτελεσμάτων. Θα εξετάσουμε σε τρεις διαφορετικές κλίμακες νευρώνων τα αποτελέσματα, οι οποίες είναι 10 – 20 – 40. Τον βασικότερο ρόλο για τον καλύτερο συνδυασμό παραμέτρων παίζει η ακρίβεια στις προβλέψεις του σταδίου εξέτασης. Τέλος, για το μέγεθος του χρόνου δε θα κάνουμε κάποια αναφορά, γιατί όπως θα δούμε οι εκτελέσεις που γίνονται χρειάζονται λιγότερο από χιλιοστά χρόνο για να ολοκληρωθούν.

8.2.1.1 Παραλλαγή 1.1: Αναλογία συνόλων 70/30

Στον πίνακα που ακολουθεί παρουσιάζονται οι καλύτερες εκτελέσεις του ELM για τον συνδυασμό συνάρτησης ενεργοποίησης – πλήθος νευρώνων.

| 1.1 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|-----------------------|---------|-----------------------|---------|-----------------------|---------|
| | Sigmoid | 40 | Tangent | 20 | RBF | 40 |
| Tr. Time | 0.0007 sec | | 0.0003 sec | | 0.0007 sec | |
| Tr. RMSE | 0.0976 | | 0.0982 | | 0.1560 | |
| Tr. Acc. % | 97.81% | | 97.81% | | 95.62% | |
| Ts. Time | 0.0001 sec | | 0.0001 sec | | 0.0001 sec | |
| Ts. RMSE | 0.1755 | | 0.1777 | | 0.2147 | |
| Ts. Acc. % | 94.44% | | 94.67% | | 92.22% | |
| Full Time | 0.0074 sec (x10 reps) | | 0.0040 sec (x10 reps) | | 0.0076 sec (x10 reps) | |

Πίνακας 43 1.1 Αποτελέσματα ELM – IRIS – 70/30

Από τον παραπάνω πίνακα γίνεται εμφανές πως τα ποσοστά επιτυχίας στην ακρίβεια είναι αντίστοιχα και για τις τρεις διαφορετικές συναρτήσεις ενεργοποίησης. Το ποσοστό ακρίβειας της εκπαίδευσης είναι το αντίστοιχο στη σιγμοειδή και την εφαπτομένη, με την εφαπτομένη να έχει ένα ελαφρύ προβάδισμα στο ποσοστό της εξέτασης αλλά και στο γεγονός ότι χρειάζεται λιγότερους νευρώνες για να πετύχει καλύτερη γενίκευση το δίκτυο.

8.2.1.2 Παραλλαγή 1.2: Αναλογία συνόλων 50/50

Αλλάζοντας την αναλογία στις παρατηρήσεις που έχουν το σύνολο δεδομένων εκπαίδευσης και το σύνολο δεδομένων εξέτασης κάνοντάς τα πλέον ισομεγέθη, επαναλάβαμε τις προσομοιώσεις και στον παρακάτω πίνακα φανερώνονται τα καλύτερα αποτελέσματα σε συνδυασμό συνάρτησης ενεργοποίησης – νευρώνες:

| 1.2 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|-----------------------|---------|-----------------------|---------|-----------------------|---------|
| | Sigmoid | 20 | Tangent | 20 | RBF | 20 |
| Tr. Time | 0.0002 sec | | 0.0004 sec | | 0.0004 sec | |
| Tr. RMSE | 0.1073 | | 0.0986 | | 0.1396 | |
| Tr. Acc. % | 97.20% | | 97.73% | | 96.27% | |
| Ts. Time | 0.0001 sec | | 0.0002 sec | | 0.0002 sec | |
| Ts. RMSE | 0.1583 | | 0.1787 | | 0.2161 | |
| Ts. Acc. % | 95.47% | | 94.53% | | 92.53% | |
| Full Time | 0.0031 sec (x10 reps) | | 0.0054 sec (x10 reps) | | 0.0061 sec (x10 reps) | |

Πίνακας 44 1.2 Αποτελέσματα ELM – IRIS – 50/50

Όπως φαίνεται από τον παραπάνω πίνακα και οι τρεις συναρτήσεις ενεργοποίησης είναι πολύ κοντά μεταξύ τους, πετυχαίνοντας τις καλύτερες προσομοιώσεις στους **20 κρυφούς νευρώνες**. Πιο ειδικευμένα το δίκτυο καταφέρνει να εκπαιδευτεί καλύτερα όταν χρησιμοποιείται η εφαπτομένη ως συνάρτηση ενεργοποίησης, ενώ μπορεί να προσομοιώνει καλύτερα σε νέα δεδομένα μετά την εκπαίδευση μέσω της σιγμοειδούς συνάρτησης.

Τέλος, αν προχωρήσουμε σε προσομοιώσεις **40 νευρώνες στο κρυφό στρώμα**, το δίκτυο αν και καταφέρνει ποσοστά πρόβλεψης στην **εκπαίδευση στο 99%**, στα νέα δεδομένα δε μπορεί να ανταποκριθεί στον ίδιο βαθμό όπως με τους 20 νευρώνες και **έχει μία πιο αδύναμη εικόνα**.

8.2.2 Προσομοιώσεις στο πρόβλημα “IMAGE SEGMENT” μέσω ELM

Στο πρόβλημα ταξινόμησης με την κατηγοριοποίηση των υπό-εικόνων στα κατάλληλη κατηγορία αντικειμένου που ανήκουν, χωρίσαμε τα δείγματά μας σε 3 διαφορετικές αναλογίες συνόλων. Αρχικά στην πρώτη παραλλαγή εξετάζουμε το δίκτυο με το 80% από τα δεδομένα που έχουμε διαθέσιμα να απαρτίζουν το σύνολο των δεδομένων εκπαίδευσης, και το υπόλοιπο 20% να χρησιμοποιείται για το τεστάρισμα του δικτύου σε νέα δεδομένα. Ακολουθεί η αναλογία 65/35 η οποία αντιστοιχεί και στην αναλογία που επέλεξε ο Huang για τις προσομοιώσεις του στο συγκεκριμένο πρόβλημα και τέλος η αναλογία 50/50 που τα σύνολα δεδομένων μας είναι ισάξια. Όπως θα δούμε και στις 3 παραλλαγές ο ELM ενώ καταφέρνει να εκπαιδεύσει το δίκτυο σε ποσοστό 95% ακρίβειας, οι προσομοιώσεις στο τεστάρισμα του δικτύου μετά βίας ξεπερνάνε το 72%. Αυτή η εικόνα είναι σταθερή και παρατηρείται σε όλες τις παραλλαγές. Σε παρακάτω υποενότητα θα συγκρίνουμε τα αποτελέσματα μαζί με τον BP και τα αντίστοιχα του Huang και θα διαπιστώσουμε πως η αδυναμία εμφανίζεται μόνο στο συγκεκριμένο πρόβλημα ταξινόμησης και για τον συγκεκριμένο αλγόριθμο.

8.2.2.1 Παραλλαγή 2.1: Αναλογία συνόλων 80/20

Σε αυτή την παραλλαγή τα τρία επίπεδα στο πλήθος του όγκου **των νευρώνων του κρυφού επιπέδου** είναι τα **100 – 200 – 500**. Οι συναρτήσεις ενεργοποίησης διαφέρουν μόνο στο γεγονός ότι **αντικαταστήσαμε την RBF με την Hardlim**, δηλαδή τη βηματική συνάρτηση. Σε κάθε ξεχωριστό συνδυασμό παραμέτρων επαναλαμβάνουμε 10 επαναλήψεις προσομοιώσεων, ώστε να λάβουμε τη μέση τιμή των αποτελεσμάτων για την καλύτερη δυνατή γενίκευση των αποτελεσμάτων, αλλά και τη μεγαλύτερη αμεροληψία. Το ίδιο συμβαίνει για όλες τις παραλλαγές των αναλογιών των συνόλων δεδομένων. Στο παρακάτω πίνακα βλέπουμε τις καλύτερες προσπάθειες ανά συνάρτηση ενεργοποίησης για το δίκτυο:

| 2.1 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|-----------------------|---------|-----------------------|---------|-----------------------|---------|
| | Sigmoid | 500 | Tangent | 500 | Hardlim | 500 |
| Tr. Time | 0.1762 sec | | 0.1812 sec | | 0.1813 sec | |
| Tr. RMSE | 0.0794 | | 0.0829 | | 0.0828 | |
| Tr. Acc. % | 96.86% | | 96.65% | | 96.64% | |
| Ts. Time | 0.0049 sec | | 0.0048 sec | | 0.0048 sec | |
| Ts. RMSE | 0.2661 | | 0.2728 | | 0.2663 | |
| Ts. Acc. % | 72.49% | | 70.93% | | 72.38% | |
| Full Time | 1.8114 sec (x10 reps) | | 1.8597 sec (x10 reps) | | 1.8619 sec (x10 reps) | |

Πίνακας 45 2.1 Αποτελέσματα ELM – IMAGE SEGMENT – 80/20

Από τα αποτελέσματα στα ποσοστά ακρίβειας παρατηρούμε πως, οι προσομοιώσεις με τη σιγμοειδή και τη βηματική συνάρτηση είναι πιο αποτελεσματικές σε σύγκριση με αυτή της εφαπτομένης. Για ένα πολύ μικρό ποσοστό της τάξεως του **0.22%** στην εκπαίδευση και για ένα ποσοστό του **0.11%**, οι προσομοιώσεις με τη σιγμοειδή συνάρτηση είναι πιο αποδοτικές σε σχέση με τη βηματική.

Το ποσοστό επιτυχίας στην ακρίβεια των προβλέψεων στα νέα δεδομένα είναι περίπου 1.5% πιο λίγο για τις προσομοιώσεις με την εφαπτομένη. Χρονικά είναι στα ίδια επίπεδα, τα οποία είναι εξαιρετικά πολύ γρήγορα, οι προσομοιώσεις και στις τρεις συναρτήσεις ενεργοποίησης, όπου ταυτόχρονα πετυχαίνουν τις καλύτερες προσομοιώσεις τους για **500 νευρώνες** στο κρυφό στρώμα.

8.2.2.2 Παραλλαγή 2.2: Αναλογία συνόλων 65/35

Επαναλαμβάνοντας τις παραπάνω προσομοιώσεις για ένα συνδυασμό δεδομένων με αναλογία 65/35 στα σύνολα τιμών, με μόνη διαφοροποίηση τους 650 νευρώνες ως μέγιστο αριθμό αντί για τους 500, διαπιστώσαμε ότι το δίκτυο έχει ακριβώς την ίδια συμπεριφορά. Αυτή μεταφράζεται στο γεγονός ότι δε μπορεί να βελτιώσει τα ποσοστά επιτυχίας στην ακρίβεια του testing σκέλους, αλλά παραμένουν στα επίπεδα του ~70%. Η μόνη διαφοροποίηση που θα δούμε στα αποτελέσματα του παρακάτω πίνακα, είναι ότι πλέον οι προσομοιώσεις με τη σιγμοειδή συνάρτηση αποδίδουν 5% καλύτερες προβλέψεις στο δίκτυο σε σχέση με τις άλλες δύο συγκρινόμενες συναρτήσεις ενεργοποίησης σε νέα και άγνωστα δεδομένα.

| 2.2 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|-----------------------|---------|-----------------------|---------|-----------------------|---------|
| | Sigmoid | 650 | Tangent | 650 | Hardlim | 100 |
| Tr. Time | 0.0968 sec | | 0.0066 sec | | 0.0070 sec | |
| Tr. RMSE | 0.0936 | | 0.2729 | | 0.2683 | |
| Tr. Acc. % | 95.76% | | 71.92% | | 72.74% | |
| Ts. Time | 0.0041 sec | | 0.0007 sec | | 0.0008 sec | |
| Ts. RMSE | 0.2760 | | 0.3033 | | 0.3007 | |
| Ts. Acc. % | 70.44% | | 65.37% | | 65.85% | |
| Full Time | 1.0089 sec (x10 reps) | | 0.0731 sec (x10 reps) | | 0.0779 sec (x10 reps) | |

Πίνακας 46 2.2 Αποτελέσματα ELM – IMAGE SEGMENT – 65/35

Με τις παραπάνω μετρήσεις γίνεται εμφανές πως το δίκτυο αποκτά καλύτερη προσέγγιση στις προβλέψεις, τόσο με στην εκπαίδευση όσο και στην εξέταση μέσω της σιγμοειδής συνάρτησης ενεργοποίησης. Μάλιστα η εφαπτομένη και η βηματική προσφέρουν περισσότερο από 20% χειρότερες εκπαιδύσεις στην ακρίβεια των προβλέψεων του δικτύου. Επίσης, να αναφέρουμε πως η αύξηση των νευρώνων δε βελτιώνει ιδιαίτερα, και σε κάποιες περιπτώσεις είναι και χειρότερη επιλογή, στην αποτελεσματικότητα των δικτύων. Τα ποσοστά ακρίβειας είναι σχεδόν αντίστοιχα στους 100, στους 200 και στους 650 κρυφούς νευρώνες. Τέλος, χρονικά όλες οι προσομοιώσεις είναι στο ίδιο επίπεδο.

8.2.2.3 Παραλλαγή 2.3: Αναλογία συνόλων 50/50

Η τελευταία παραλλαγή γι' αυτό το πρόβλημα αφορά τις προσομοιώσεις με ίδιο πλήθος παρατηρήσεων του συνόλου δεδομένων εκπαίδευσης σε σχέση με αυτό του συνόλου εξέτασης. Όπως θα παρατηρήσουμε στα συγκεντρωτικά αποτελέσματα στον πίνακα που θα ακολουθήσει, η σταθερότητα στις προσομοιώσεις είναι αντίστοιχη με τις προηγούμενες παραλλαγές. Παρά το γεγονός ότι φτάνουμε έως και τους 800 νευρώνες στο κρυφό στρώμα, ο αλγόριθμος με τις διαφορετικές συναρτήσεις ενεργοποίησης δεν καταφέρνει να κάνει το δίκτυο να αποδώσει σε νέα δεδομένα με ποσοστό επιτυχίας πάνω από 70%. Είναι χαρακτηριστικό σε αυτή την παραλλαγή, όπως και στην πρώτη, πως ενώ τα

ποσοστά ακρίβειας στην εκπαίδευση φτάνουν στο 95% και πλέον, δε μετουσιώνονται στο σκέλος της εξέτασης.

| 2.3 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|-----------------------|---------|-----------------------|---------|-----------------------|---------|
| | Sigmoid | 800 | Tangent | 800 | Hardlim | 800 |
| Tr. Time | 0.0555 sec | | 0.0563 sec | | 0.0560 sec | |
| Tr. RMSE | 0.1072 | | 0.1042 | | 0.1054 | |
| Tr. Acc. % | 94.82% | | 95.12% | | 95.00% | |
| Ts. Time | 0.0035 sec | | 0.0036 sec | | 0.0039 sec | |
| Ts. RMSE | 0.2835 | | 0.2887 | | 0.2839 | |
| Ts. Acc. % | 69.06% | | 68.23% | | 68.95% | |
| Full Time | 0.5903 sec (x10 reps) | | 0.5991 sec (x10 reps) | | 0.5982 sec (x10 reps) | |

Πίνακας 47.2.3 Αποτελέσματα ELM – IMAGE SEGMENT – 50/50

8.2.2.4 Σύνοψη Image Segment για τον ELM

Συνοψίζοντας, για το συγκεκριμένο πρόβλημα ο ELM δεν είχε τόσο καλή προσέγγιση όσο είχε στο πρόβλημα της Ίριδας, αλλά όπως θα έχει και στα επόμενα προβλήματα ταξινόμησης που θα δούμε. Είναι χαρακτηριστικό πως τα ποσοστά επιτυχίας που είχε στην ακρίβεια, δε μετουσιώνονταν και στην εξέταση των δικτύων σε νέα δεδομένα, ανεξαρτήτως της αναλογίας των συνόλων των δεδομένων.

Αυτό που κρατάμε είναι το πλεονέκτημα της ταχύτητας και σε εντυπωσιακό βαθμό, αλλά και τη σταθερότητα που παρουσιάζει ο αλγόριθμος, χωρίς να επηρεάζεται από τις αλλαγές των παραμέτρων.

8.2.3 Προσομοιώσεις στο πρόβλημα “SATELLITE IMAGES” μέσω ELM

Το τρίτο κατά σειρά πρόβλημα ταξινόμησης που εξετάζουμε, είναι αυτό με τις υπό-εικόνες που τμηματοποιούμε από μία μεγάλη εικόνα τραβηγμένη από δορυφόρο στο διάστημα. Στη φιλοσοφία του είναι ανάλογο με το προηγούμενο, καθώς εξετάζουμε μικρές εικόνες από τα τμήματα μιας πολύ μεγαλύτερης και τις κατηγοριοποιούμε ανάλογα τη μορφολογία του εδάφους σε 6 κατηγορίες. Τα γνωρίσματα είναι 36 εδώ και οι παρατηρήσεις φτάνουν τις 6430.

Και σε αυτή την περίπτωση προβλήματος χωρίζουμε σε τρεις διαφορετικές αναλογίες τα σύνολα των δεδομένων μας ξεκινώντας με 80/20, ακολούθως 68/32 και τέλος σε 50/50. Να αναφέρουμε ότι είναι το δεύτερο πρόβλημα που χρησιμοποίησε στην επιστημονική εργασία του για τον ELM ο Huang, και το επαναλαμβάνουμε και εμείς.

Οι συναρτήσεις ενεργοποίησης που επιλέξαμε είναι η σιγμοειδής, εφαπτομένη και βηματική. Και στις τρεις παραλλαγές ξεκινάμε με 200 νευρώνες τις προσομοιώσεις μας και συνεχίζουμε στους 500. Αυτό που αλλάζει είναι το τρίτο και μέγιστο επίπεδο στους νευρώνες, που διαφοροποιείται ανάλογα τα σύνολα δεδομένων.

Τέλος, όπως έχουμε προαναφέρει τα σύνολα δεδομένων στην κάθε παραλλαγή είναι ακριβώς τα ίδια με αυτά που χρησιμοποιήσαμε και στις προσομοιώσεις μέσω του BP, συνεπώς κρατάμε την αμεροληψία στα αποτελέσματα που θα προκύψουν.

8.2.3.1 Παραλλαγή 3.1: Αναλογία συνόλων 80/20

Πρώτη παραλλαγή όπως πάντα είναι αυτή που έχει το μέγιστο πλήθος δεδομένων από τις αναλογίες που διαλέγουμε για το σύνολο δεδομένων εκπαίδευσης. Έτσι προσπαθούμε σε πρώτο χρόνο να εκπαιδύσουμε όσο το δυνατόν καλύτερα το δίκτυο και να το εξετάσουμε σε ένα σύνολο από λίγα δεδομένα.

Στον πίνακα που ακολουθεί θα παρατηρήσουμε ότι ο αλγόριθμος μπορεί να εκπαιδύσει το δίκτυο σε πολύ καλό βαθμό, αλλά ταυτόχρονα να μπορεί να προσεγγίζει και τα νέα δεδομένα με πολύ ικανοποιητικά ποσοστά ακρίβειας.

| 3.1 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|-----------------------|---------|-----------------------|---------|-----------------------|---------|
| | Sigmoid | 700 | Tangent | 700 | Hardlim | 700 |
| Tr. Time | 0.5707 sec | | 0.5837 sec | | 0.5910 sec | |
| Tr. RMSE | 0.1546 | | 0.1468 | | 0.1496 | |
| Tr. Acc. % | 91.70% | | 92.49% | | 92.34% | |
| Ts. Time | 0.0145 sec | | 0.0150 sec | | 0.0143 sec | |
| Ts. RMSE | 0.2184 | | 0.2253 | | 0.2132 | |
| Ts. Acc. % | 83.79% | | 83.52% | | 84.77% | |
| Full Time | 5.8513 sec (x10 reps) | | 5.9868 sec (x10 reps) | | 6.0528 sec (x10 reps) | |

Πίνακας 48 3.1 Αποτελέσματα ELM – SATELLITE IMAGES – 80/20

Αρχικά, η **χρονική διάρκεια** των προσομοιώσεων είναι η αντίστοιχη και για τις τρεις περιπτώσεις συναρτήσεων ενεργοποίησης, είναι εξαιρετικά γρήγορες προσομοιώσεις και μεταφράζονται σε **0.7 δευτερόλεπτα**. Οι καλύτερες προσομοιώσεις έχουν προκύψει από δίκτυα που είναι εφοδιασμένα με **700 νευρώνες** στο κρυφό στρώμα. Ελάχιστα καλύτερη ακρίβεια στην εκπαίδευση ενός δικτύου μπορεί να προσφέρει μια προσομοίωση χρησιμοποιώντας την εφαπτομένη σαν συνάρτηση ενεργοποίησης, ενώ αντίθετα η προσομοίωση που έχει ως συνάρτηση ενεργοποίησης τη βηματική προσφέρει συνολικά καλύτερες προσεγγίσεις σε νέα δεδομένα στο δίκτυο.

8.2.3.2 Παραλλαγή 3.2: Αναλογία συνόλων 68/32

Σε αυτή την παραλλαγή ακολουθήσαμε μία αναλογία συνόλων στα δεδομένα αντίστοιχη με αυτή του Huang. Έτσι μειώσαμε τα δεδομένα στο σύνολο εκπαίδευσης και προσθέσαμε περισσότερα στο σύνολο για την εξέταση του δικτύου. Η δεύτερη διαφοροποίηση είναι ότι οι μέγιστοι νευρώνες που εξετάσαμε είναι οι 800 αντί για 700.

Όπως θα παραληρήσουμε στο παρακάτω πίνακα το δίκτυο έχει αντίστοιχα ποσοστά επιτυχίας όπως και στην προηγούμενη παραλλαγή. Η ακρίβεια στις προβλέψεις της εκπαίδευσης μπορεί να φτάσει στο 92% ενώ τα νέα δεδομένα μπορεί να τα κατηγοριοποιήσει με ποσοστό επιτυχία στο 84%.

Είναι χαρακτηριστικό πως ενώ αυξάνουμε τους νευρώνες στο δίκτυο, η βελτίωση στην εκπαίδευση είναι πιο μεγάλη σε σχέση με την εξέταση καθώς με 200 νευρώνες το δίκτυο ξεκινάει με ένα ποσοστό επιτυχίας εκπαίδευσης στο 85.20% και φτάνει έως το 92% με 800 νευρώνες, ενώ αντίστοιχα στην εξέταση ξεκινάει από το 82.50% και φτάνει έως το 84.12%.

| 3.2 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|-----------------------|---------|-----------------------|---------|-----------------------|---------|
| | Sigmoid | 800 | Tangent | 800 | Hardlim | 500 |
| Tr. Time | 0.3188 sec | | 0.3377 sec | | 0.1470 sec | |
| Tr. RMSE | 0.1599 | | 0.1535 | | 0.1804 | |
| Tr. Acc. % | 91.21% | | 91.90% | | 89.16% | |
| Ts. Time | 0.0130 sec | | 0.0137 sec | | 0.0082 sec | |
| Ts. RMSE | 0.2217 | | 0.2280 | | 0.2196 | |
| Ts. Acc. % | 83.62% | | 83.08% | | 84.12% | |
| Full Time | 3.3185 sec (x10 reps) | | 3.5136 sec (x10 reps) | | 1.5515 sec (x10 reps) | |

Πίνακας 49 3.2 Αποτελέσματα ELM – SATELLITE IMAGES – 68/32

Όπως παρατηρούμε η καλύτερη δυνατή αποτελεσματικότητα μπορεί να προέλθει μέσω της χρήσης της βηματικής συνάρτησης ενεργοποίησης και για 500 νευρώνες. Επειδή, στο δίκτυο το οποίο αποτελείται από ένα πιο μικρό αριθμό νευρώνων γίνονται και λιγότεροι υπολογισμοί, εξού η γρηγορότερες προσομοιώσεις. Αν συγκρίνουμε τις προσομοιώσεις με τον ίδιο αριθμό νευρώνων στο κρυφό στρώμα δεν υπάρχει κάποια χρονική διαφοροποίηση.

8.2.3.3 Παραλλαγή 3.3: Αναλογία συνόλων 50/50

Η τελευταία παραλλαγή είναι αυτή η οποία έχει ίδιο αριθμό παρατηρήσεων και στα δύο σύνολα δεδομένων. Η επιπλέον διαφοροποίηση σε σχέση με τις άλλες παραλλαγές είναι ότι εξετάζουμε το δίκτυο και για 1000 νευρώνες στις προσομοιώσεις μας.

| 3.3 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|-----------------------|---------|-----------------------|---------|-----------------------|---------|
| | Sigmoid | 500 | Tangent | 500 | Hardlim | 500 |
| Tr. Time | 0.1347 sec | | 0.1359 sec | | 0.1376 sec | |
| Tr. RMSE | 0.1694 | | 0.1655 | | 0.1654 | |
| Tr. Acc. % | 90.19% | | 90.65% | | 90.70% | |
| Ts. Time | 0.0102 sec | | 0.0108 sec | | 0.0105 sec | |
| Ts. RMSE | 0.2258 | | 0.2317 | | 0.2258 | |
| Ts. Acc. % | 83.28% | | 82.66% | | 83.38% | |
| Full Time | 1.4491 sec (x10 reps) | | 1.4678 sec (x10 reps) | | 1.4805 sec (x10 reps) | |

Πίνακας 50 3.3 Αποτελέσματα ELM – SATELLITE IMAGES – 50/50

Όπως προκύπτει από τις παραπάνω μετρήσεις, και σε αυτή την παραλλαγή η καλύτερη προσέγγιση έρχεται με τη χρήση της βηματικής συνάρτησης στις προσομοιώσεις.

8.2.3.4 Σύνοψη Satellite Images για τον ELM

Από τις προσομοιώσεις που κάναμε γι' αυτό το πρόβλημα ταξινόμησης προκύπτει πως ο ELM καταφέρνει να κρατήσει μία σταθερή εικόνα γι' όλες τις παραλλαγές του προβλήματος. Αρχικά, παρά ότι έχουμε 3 φορές σχεδόν περισσότερες παρατηρήσεις από το προηγούμενο πρόβλημα, χρονικά είναι εξαιρετικά γρήγορος και πάλι.

Επιπρόσθετα, το ποσοστό ακρίβειας στην εκπαίδευση του δικτύου είναι σταθερά πάνω από το 90%, όπως και το ποσοστό ακρίβειας στο testing είναι κοντά ή και πάνω από το 84%. Σε κάθε ξεχωριστό συνδυασμό παραμέτρων εκτελέσαμε 10 επαναλήψεις τις ίδιες προσομοιώσεις, με σκοπό να καταφέρουμε τα καλύτερα δυνατά αποτελέσματα αλλά μέσω αξιοπιστίας στις προσομοιώσεις μας.

Τέλος, δεν αντιμετωπίσαμε κάποια φαινόμενα overfitting στις προσομοιώσεις μας, γεγονός που δίνει τη δυνατότητα ανοχής στις προσομοιώσεις μέσω του ELM, και δεν δημιουργεί ασταθή και εύκολα στην απόκλιση δίκτυα.

8.2.4 Προσομοιώσεις στο πρόβλημα “MNIST” μέσω ELM

Το τελευταίο πρόβλημα που θα εξετάσουμε μέσω του ELM, είναι το πρόβλημα ταξινόμησης MNIST. Όπως έχουμε προαναφέρει πρόκειται για ένα πρόβλημα που έχει προκύψει από χειρόγραφες αναπαραστάσεις των ψηφίων 0 – 9, οι οποίες έχουν ψηφιοποιηθεί σε εικόνες 28 x 28 pixels σε greyscale κλίμακα.

Τα επίπεδα που εξετάζουμε στις παραλλαγές για τα δεδομένα εκπαίδευσης είναι 3. Πρώτα για 10000 παρατηρήσεις, στη συνέχεια για 30000 και τέλος για 60000 παρατηρήσεις. Σε όλες τις προσομοιώσεις χρησιμοποιείται το ίδιο σύνολο δεδομένων εξέτασης το οποίο απαρτίζεται από 10000 παρατηρήσεις. Παράλληλα τα σύνολα δεδομένων εκπαίδευσης περιέχουν τις αντίστοιχες παρατηρήσεις όπως και στον BP, καθώς πρόκειται για τα ίδια σύνολα. Επισημαίνεται ότι στις προσομοιώσεις με 10000 παρατηρήσεις, κάθε συνδυασμός παραμέτρων εκτελέστηκε για 10 φορές και λάβαμε τον Μ.Ο. των εξεταζόμενων μεγεθών. Αντίστοιχα, για τις 30000 και 60000 εκτελέστηκαν 5 φορές.

Σε αυτό το πρόβλημα εκτελέσαμε ένα σύνολο με 40 και 100 νευρώνες, με συναρτήσεις ενεργοποίησης τη σιγμοειδή, τη εφαπτομένη και τη RBF. Και τις τρεις παραλλαγές πάντα οι καλύτερες προσομοιώσεις είναι αυτές με τη σιγμοειδή συνάρτηση. Γι’ αυτό το λόγο επιλέξαμε σαν παράμετρο και βασική συνάρτηση ενεργοποίησης για τις προσομοιώσεις μας τη **σιγμοειδή** και αυξάναμε το πλήθος των κρυφών νευρώνων έως και τους 5000 για να δούμε τα ποσοστά επιτυχίας του.

Στους πίνακες που θα ακολουθήσουν θα παρουσιάσουμε τις προσομοιώσεις για 40 – 400 – 5000 νευρώνες μέσω της σιγμοειδής συνάρτησης, για να δούμε την κλιμακωτή βελτίωση του δικτύου.

8.2.4.1 Παραλλαγή 4.1: 10000 training – 10000 testing datasets

| 4.1 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|-----------------------|---------|-----------------------|---------|---------------------|---------|
| | Sigmoid | 40 | Sigmoid | 400 | Sigmoid | 5000 |
| Tr. Time | 0.0192 sec | | 0.2232 sec | | 37.4167 sec | |
| Tr. RMSE | 0.2490 | | 0.1275 | | 0.0004 | |
| Tr. Acc. % | 68.85% | | 91.67% | | 100.00% | |
| Ts. Time | 0.0153 sec | | 0.0658 sec | | 0.6963 sec | |
| Ts. RMSE | 0.2490 | | 0.1275 | | 0.0004 | |
| Ts. Acc. % | 68.85% | | 91.67% | | 100.00% | |
| Full Time | 0.3452 sec (x10 reps) | | 2.8899 sec (x10 reps) | | 00:06:21 (x10 reps) | |

Πίνακας 51 4.1 Αποτελέσματα BP – MNIST - 10000

Όπως προκύπτει από τον παραπάνω πίνακα ξεκινώντας από **40 νευρώνες** στο δίκτυο τα ποσοστά επιτυχίας στην ακρίβεια είναι ίδια και στα δύο επίπεδα εξέτασης με **69.85%**. Είναι χαρακτηριστικό πως το ίδιο συμβαίνει και στους 400 αλλά και στους 5000 νευρώνες, με τα ποσοστά να φτάνουν στο **91.67%** και **100%** (!) αντίστοιχα.

Στους 5000 νευρώνες όπως παρατηρούμε το δίκτυο καταφέρνει να εκπαιδευτεί σχεδόν πλήρως, κι αυτό παρατηρείται αν δούμε το RMSE που είναι 0.0004 σε training και testing που σημαίνει ότι λανθάνει σε 4 από τις 10000 παρατηρήσεις που προσπαθεί να προβλέψει σε κάθε σκέλος!

Το εντυπωσιακό είναι ότι το καταφέρνει αυτό καταναλώνοντας 38 δευτερόλεπτα περίπου (!) κατά Μ.Ο. , για μία προσομοίωση με 5000 νευρώνες στο κρυφό στρώμα.

8.2.4.2 Παραλλαγή 4.2: 30000 training – 10000 testing datasets

| 4.2 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|----------------------|---------|----------------------|---------|--------------------|---------|
| | Sigmoid | 40 | Sigmoid | 400 | Sigmoid | 5000 |
| Tr. Time | 0.0645 sec | | 0.6188 sec | | 00:01:01 | |
| Tr. RMSE | 0.2506 | | 0.1375 | | 0.0433 | |
| Tr. Acc. % | 68.45% | | 90.33% | | 99.03% | |
| Ts. Time | 0.0181 sec | | 0.0797 sec | | 0.8530 sec | |
| Ts. RMSE | 0.2466 | | 0.1350 | | 0.0443 | |
| Ts. Acc. % | 69.46% | | 90.68% | | 98.97% | |
| Full Time | 0.8255 sec (x5 reps) | | 3.4927 sec (x5 reps) | | 00:05:12 (x5 reps) | |

Πίνακας 52 4.2 Αποτελέσματα BP – MNIST - 30000

Αντίστοιχη εικόνα παρατηρείται όταν το δίκτυο εκπαιδεύεται με 30000 παρατηρήσεις από το σύνολο δεδομένων εκπαίδευσης. Το δίκτυο καταφέρνει να προβλέψει με επιτυχία το 99.03% των παρατηρήσεων κατά την εκπαίδευση και το 98.97% όταν προσομοιώνει σε νέα δεδομένα.

Χρονικά χρειάζεται κάτι περισσότερο από 1' για να ολοκληρώσει μία προσομοίωση με 5000 νευρώνες στο κρυφό στρώμα.

8.2.4.3 Παραλλαγή 4.3: 60000 training – 10000 testing datasets

| 4.3 | AF | Neurons | AF | Neurons | AF | Neurons |
|------------|----------------------|---------|----------------------|---------|--------------------|---------|
| | Sigmoid | 40 | Sigmoid | 400 | Sigmoid | 5000 |
| Tr. Time | 0.1564 sec | | 1.2737 sec | | 00:01:29 | |
| Tr. RMSE | 0.2554 | | 0.1407 | | 0.0598 | |
| Tr. Acc. % | 67.19% | | 89.86% | | 98.16% | |
| Ts. Time | 0.0208 sec | | 0.0989 sec | | 1.0945 sec | |
| Ts. RMSE | 0.2516 | | 0.1375 | | 0.0590 | |
| Ts. Acc. % | 68.18% | | 90.32% | | 98.20% | |
| Full Time | 0.8857 sec (x5 reps) | | 6.8627 sec (x5 reps) | | 00:07:35 (x5 reps) | |

Πίνακας 53 4.3 Αποτελέσματα BP – MNIST - 60000

Όπως φαίνεται από τα παραπάνω ποσοστά επιτυχίας στις προβλέψεις, το δίκτυο με 5000 νευρώνες φτάνει να προβλέψει πάνω από 98%. Αυτό σημαίνει ότι λανθάνει στις 1100 περίπου παρατηρήσεις από τις 60000 που εκπαιδεύεται και μόλις στις 180 από τις 10000 όταν κληθεί να πειραματιστεί στα νέα δεδομένα.

8.3 Αποτίμηση – Σύγκριση αποτελεσμάτων Ταξινόμησης

Σε αυτή την ενότητα θα συγκρίνουμε τ' αποτελέσματα από τις προσομοιώσεις των νευρωνικών αλγορίθμων στα δίκτυα για τα προβλήματα ταξινόμησης. Σκοπός είναι να αποτιμήσουμε την ακρίβεια, την αποτελεσματικότητα, την ταχύτητα και την απόδοση των νευρωνικών αλγορίθμων στα 4 διαφορετικά προβλήματα που αναπτύξαμε στις προηγούμενες ενότητες, αλλά και να συγκρίνουμε τα δικά μας ευρήματα με τα αντίστοιχα αποτελέσματα που παρουσίασε ο Huang στην επιστημονική του έρευνα προτείνοντας τον ELM.

Παρακάτω θα παρουσιάσουμε σε πίνακες τις καλύτερες προσομοιώσεις για κάθε ξεχωριστή παραλλαγή των προβλημάτων, στους οποίους θα αναφερθούμε και στους χρόνους πέρα από την ακρίβεια των προβλέψεων. Μόλις ολοκληρώσουμε όλες τις συγκρίσεις θα καταλήξουμε σε μία τελική αποτίμηση και για τις δύο ξεχωριστές κατηγορίες προβλημάτων που αναλύσαμε στα τρία τελευταία κεφάλαια.

Ο διαχωρισμός των δεδομένων που ακολουθείται για όλα τα προβλήματα ταξινόμησης και όλες τις παραλλαγές είναι αμερόληπτος όπως έχουμε προαναφέρει. Αυτό ερμηνεύεται ως εφαρμογή του ποσοστού των δεδομένων του κάθε συνόλου για το σύνολο των παρατηρήσεων κάθε κατηγορίας ξεχωριστά και όχι μόνο επί του αρχικού συνόλου όλων των παρατηρήσεων. Αυτό σημαίνει ότι αν επιλέξουμε να αποδώσουμε από το σύνολο όλων των παρατηρήσεων το 70% στο σύνολο δεδομένων εκπαίδευσης, τότε θα συγκεντρωθούν οι παρατηρήσεις κάθε κατηγορίας ξεχωριστά και για κάθε κατηγορία θα εφαρμόσουμε τυχαία το 70% στις παρατηρήσεις τους. Με αυτό τον τρόπο θα πάρουμε στο τέλος το 70% από όλες τις παρατηρήσεις αλλά ταυτόχρονα και το 70% από την κάθε κατηγορία. Συνεπώς, θα είναι ομοιόμορφα κατανομημένη αυτή η τυχαία επιλογή.

Τέλος, **τα σύνολα κάθε παραλλαγής όλων των προβλημάτων δημιουργήθηκαν εξ' αρχής και χρησιμοποιήθηκαν ακριβώς τα ίδια για κάθε αλγόριθμο**. Έτσι οι αλγόριθμοι εκπαιδύσαν και έδωσαν τη γνώση για εξέταση στο δίκτυο πάνω στα ίδια σύνολα δεδομένων, άρα έχουμε μία δίκαιη σύγκριση χωρίς να ευνοείται ή να αδικείται κάποιος νευρωνικός αλγόριθμος με διαφοροποίηση των συνόλων των δεδομένων.

8.3.1 Το πρόβλημα ταξινόμησης των παρατηρήσεων της Ίριδας

Το πρόβλημα IRIS είναι το πρώτο πρόβλημα που εξετάσαμε και στους δύο αλγορίθμους για την ταξινόμηση. Όπως έχουμε προαναφέρει αρκετές φορές, πρόκειται για ένα σύνολο από 150 παρατηρήσεις οι οποίες περιέχουν 4 ξεχωριστά γνωρίσματα, δηλαδή 4 διαφορετικά είδη μέτρησης και υπάρχουν 3 διαφορετικές κατηγορίες στις οποίες μπορεί να τοποθετηθεί η κάθε παρατήρηση.

Μόνο για το συγκεκριμένο πρόβλημα η κάθε κατηγορία περιέχει από 50 παρατηρήσεις, συνεπώς πρόκειται για ισομοιρασμένο σετ δεδομένων. Για αυτή την περίπτωση χωρίσουμε με τυχαίο τρόπο αλλά και αμερόληπτο σε δύο ξεχωριστές αναλογίες συνόλων τα δεδομένα – παρατηρήσεις, και δημιουργούμε δύο αναλογίες 70/30 και 50/50 για σύνολο δεδομένων εκπαίδευσης / σύνολο δεδομένων εξέτασης.

8.3.1.1 Παραλλαγή 1.1: Σύγκριση ELM και BP - IRIS - 70/30

Οι δύο ακόλουθοι πίνακες περιέχουν τις καλύτερες προσπάθειες για το κάθε ζευγάρι αναλογίας των συνόλων δεδομένων συγκεντρωτικά:

| 1.1 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|------------|---------|-----|---------|------------|-----------|------------|-----------|--------------------------|
| ELM | Tangent | - | 20 | 0.0003 sec | 97.81% | 0.0001 sec | 94.67% | 0.0040 sec (x10 reps) |
| SCG | - | 0.1 | 10 | 0.5354 sec | 92.29% | 0.0046 sec | 95.78% | 5.4002 sec (x10 reps) |
| Rprop | - | 0.1 | 20 | 0.5171 sec | 92.38% | 0.0045 sec | 94.67% | 5.2156 sec (x10 reps) |
| LM | - | 0.1 | 10 | 0.5153 sec | 92.38% | 0.0044 sec | 94.00% | 5.1970 sec (x10 reps) |

Πίνακας 54 1.1 Σύγκριση ELM και BP - IRIS - 70/30

Από τον παραπάνω πίνακα φαίνεται ότι όλοι οι αλγόριθμοι είναι πολύ κοντά μεταξύ τους με τον Scaled Conjugate Gradient να είναι ο πιο αποτελεσματικός από όλους, καθώς καταφέρνει 95.78% επιτυχία στην ακρίβεια σε άγνωστα δεδομένα. Αντίθετα, ο ELM καταφέρνει να εκπαιδεύσει το δίκτυο και να προβλέπει με 97.81% στο στάδιο της εκπαίδευσης που είναι περισσότερο από 5% πιο αποτελεσματικός σε αυτόν τον τομέα. Τέλος, αν και χρονικά είναι όλοι οι αλγόριθμοι εξαιρετικά γρήγοροι, ο ELM χρειάζεται 4/10 του χιλιοστού για μία προσομοίωση, τη στιγμή που οι υπόλοιποι τρεις χρειάζονται περίπου μισό λεπτό.

8.3.1.2 Παραλλαγή 1.2: Σύγκριση ELM και BP - 50/50

| 1.2 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|------------|---------|------|---------|------------|-----------|------------|-----------|--------------------------|
| ELM | Sigmoid | - | 20 | 0.0002 sec | 97.20% | 0.0001 sec | 95.47% | 0.0031 sec (x10 reps) |
| SCG | - | 0.1 | 10 | 0.5329 sec | 94.40% | 0.0051 sec | 95.33% | 5.3799 sec (x10 reps) |
| Rprop | - | 0.1 | 20 | 0.5236 sec | 92.80% | 0.0044 sec | 95.20% | 5.2791 sec (x10 reps) |
| LM | - | 0.01 | 20 | 0.5255 sec | 94.53% | 0.0047 sec | 94.53% | 5.3021 sec (x10 reps) |

Πίνακας 55 1.2 Σύγκριση ELM και BP - IRIS - 50/50

Μειώνοντας τα διαθέσιμα δεδομένα για την εκπαίδευση του δικτύου και αυξάνοντας τα δεδομένα εξέτασης, το ELM προπορεύεται σε κάθε κατηγορία σύγκρισης. Προσφέρει στο δίκτυο την καλύτερη εκπαίδευση για τις προβλέψεις του, σχεδόν σε ακαριαίο χρόνο και μετουσιώνει την καλύτερη εκπαίδευση και στις προβλέψεις στα άγνωστα δεδομένα. Φυσικά όλοι οι αλγόριθμοι είναι πολύ κοντά μεταξύ τους με τον Scaled Conjugate Gradient να είναι μπορεί να αποδώσει με πιο συμπαγές δίκτυο, από τη στιγμή που χρειάζεται 10 νευρώνες στο κρυφό στρώμα σε σχέση με τους 20 που θέλουν οι υπόλοιποι.

Συνοψίζοντας

Όλοι οι αλγόριθμοι είναι πολύ κοντά μεταξύ τους έχοντας τη δυνατότητα να εκπαιδεύσουν το δίκτυο σε βαθμό που μπορεί να πετύχει άνω του 94% επιτυχία σε άγνωστα δεδομένα. Αυτό που κάνει τη διαφορά είναι όμως η απόλυτη ταχύτητα του ELM, παρά το γεγονός ότι μιλάμε και οι 4 αλγόριθμοι είναι πολύ γρήγοροι καθώς είναι πολύ μικρό το σύνολο των παρατηρήσεων.

8.3.2 Το πρόβλημα ταξινόμησης με αναγνώριση αντικειμένων

Το δεύτερο πρόβλημα που εξετάσαμε του νευρωνικούς αλγόριθμους είναι αυτό της αναγνώρισης αντικειμένων μία εικόνας, μέσα από υπό-εικόνες που τμηματοποιούν την αρχική. Σε αυτό το πρόβλημα υπάρχουν 2310 παρατηρήσεις, περιγράφονται με 19 γνωρίσματα και μπορούν να κατηγοριοποιηθούν σε 7 ξεχωριστές κατηγορίες – κλάσεις αντικειμένων.

Γι' αυτό το πρόβλημα εντάσσουμε στη σύγκριση και τα αποτελέσματα του Huang που παρουσιάζει στην επιστημονική εργασία της εισήγησης του ELM. Για τα πειράματα που έκανε όμως δε γνωρίζουμε αν διαχειρίστηκε με κάποιο τρόπο τα δεδομένα του π.χ. με κανονικοποίηση των τιμών τους ή αν επέλεξε με κάποιο όχι τυχαίο και αδιάληπτο τρόπο να δημιουργήσει τα σύνολα δεδομένων. Γνωρίζουμε μόνο το πλήθος δεδομένων του κάθε συνόλου, κάτι το οποίο το εφαρμόσαμε με ποσοστά αναλογίας εμείς. Δεν επισημαίνεται στον Backpropagation ποια παραλλαγή του χρησιμοποιήθηκε από τον Huang, συνεπώς στους πίνακες που ακολουθούν αναφέρεται σαν **H-BP**, όπως και στον ELM θα τον διακρίνουμε σε σχέση με τον δικό μας με τίτλο **H-ELM**. Τέλος, παρά το γεγονός ότι θα αναφέρουμε τους χρόνους του Huang για τις εκάστοτε προσομοιώσεις, δε μπορούμε να τους συγκρίνουμε με τους δικούς καθώς το σύστημα που έτρεξε τις προσομοιώσεις είναι διαφορετικό από το δικό μας και πολύ πιο αδύναμο. Μην ξεχνάμε ότι πρόκειται για έναν αλγόριθμο τον οποίον πρότεινε πριν από 18 χρόνια, όπου τότε η δυνατότητες ενός συμβατικού οικιακού μηχανήματος δε συγκρίνονται με αυτά της τωρινής εποχής.

Υπάρχουν τρεις ξεχωριστές παραλλαγές του προβλήματος βασισμένες στην αναλογία του πλήθους των παρατηρήσεων. Θα ξεκινήσουμε με την 80/20 training / testing datasets, θα ακολουθήσει η 65/35 που έχει χρησιμοποιήσει και ο Huang και τέλος θα κλείσουμε με την 50/50 και τα ισόποσα σύνολα δεδομένων.

8.3.2.1 Παραλλαγή 2.1: Σύγκριση ELM και BP - IMAGE SEGMENT - 80/20

Στον ακόλουθο πίνακα συγκεντρώνονται οι τέσσερις καλύτερες συστοιχίες προσομοιώσεων βασισμένες στο ποσοστό επιτυχίας της ακρίβειας στο σκέλος της εξέτασης:

| 2.1 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|------------|---------|------|---------|-------------|-----------|------------|-----------|-------------------------|
| ELM | Sigmoid | - | 500 | 0.1762 sec | 96.86% | 0.0049 sec | 72.49% | 1.8114 sec (x10 reps) |
| SCG | - | 0.01 | 100 | 3.8819 sec | 92.85% | 0.0060 sec | 89.26% | 38.8795 sec (x10 reps) |
| Rprop | - | 0.1 | 100 | 3.2550 sec | 92.69% | 0.0060 sec | 88.87% | 32.6105 sec (x10 reps) |
| LM | - | 0.01 | 70 | 14.4862 sec | 96.40% | 0.0054 sec | 91.97% | 00:02:24 sec (x10 reps) |

Πίνακας 56 2.1 Σύγκριση ELM και BP - IMAGE SEGMENT - 80/20

Στις προηγούμενες ενότητες είδαμε ότι ο ELM στις δικές μας περιπτώσεις δεν απέδωσε πολύ καλά ποσοστά επιτυχίας στην ακρίβεια των νέων δεδομένων. Ενώ όπως βλέπουμε και στον παραπάνω πίνακα είναι εξαιρετικά ακριβής στην εκπαίδευση, δεν τα καταφέρνει με τον ίδιο τρόπο και στην εξέταση όπου ο Levenberg – Marquardt είναι ο πιο αποτελεσματικός αλγόριθμος. Μολονότι συμβαίνει αυτό σε χρόνο περάτωσης προσομοιώσεων είναι 18 φορές (!) τουλάχιστον πιο γρήγορος σε σχέση με τους BP, φτάνοντας έως και 80 φορές(!) πιο γρήγορος.

8.3.2.2 Παραλλαγή 2.2: Σύγκριση ELM και BP - IMAGE SEGMENT - 65/35

Σε αυτή την παραλλαγή θα προσθέσουμε και τα αποτελέσματα από τον ELM και τον BP που εκτέλεσε και ο Huang. Ο πίνακας που ακολουθεί συγκεντρώνει τις καλύτερες προσομοιώσεις για κάθε αλγόριθμο και ακολουθεί η αποτίμηση αυτών:

| 2.2 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|------------|---------|------|---------|------------|-----------|------------|-----------|------------------------|
| ELM | Sigmoid | - | 650 | 0.0968 sec | 95.76% | 0.0041 sec | 70.44% | 1.0089 sec (x10 reps) |
| SCG | - | 0.01 | 50 | 1.8424 sec | 91.07% | 0.0061 sec | 87.67% | 18.4845 sec (x10 reps) |
| Rprop | - | 0.1 | 50 | 1.6974 sec | 91.44% | 0.0061 sec | 87.80% | 17.0348 sec (x10 reps) |
| LM | - | 0.1 | 50 | 5.8550 sec | 95.13% | 0.0063 sec | 90.26% | 58.6133 sec (x10 reps) |
| H - ELM | - | - | 200 | 1.40 sec | 97.35% | 0.07 sec | 95.01% | 1.47 sec |
| H - BP | - | - | 100 | 01:19:06 | 96.92% | 0.04 sec | 86.27% | 01:19:07 |

Πίνακας 57 2.2 Σύγκριση ELM και BP - IMAGE SEGMENT - 65/35

Από τις παραπάνω μετρήσεις γίνεται πιο ξεκάθαρη η προβληματική απόδοση του ELM που εκτελέσαμε τις δικές μας προσομοιώσεις, σε σχέση με τις αντίστοιχες του Huang αλλά και των Backpropagation παραλλαγών. Μέσα από τις προσομοιώσεις του ο Huang κατάφερε να εκπαιδεύσει ελάχιστα καλύτερα τα δίκτυά του. Στις εξετάσεις όμως πάρα το γεγονός ότι πετυχαίνει καλύτερη ακρίβεια μέσω του δικού ELM για το δίκτυό του, στη δική μας περίπτωση καταφέρνουμε καλύτερες προσομοιώσεις μέσω του BP σε σχέση με αυτές του Huang. Είναι ξεκάθαρο ότι για το συγκεκριμένο πρόβλημα ταξινόμησης ο ELM δε μπορεί να αποδώσει την κατά τα φαινόμενα εξαιρετική εκπαίδευσή του που πετυχαίνει στο δίκτυο, και στις προσομοιώσεις με νέα και άγνωστα δεδομένα από το testing dataset. Ωστόσο παραμένει εξαιρετικά ταχύς, πολλαπλάσιες φορές από τον Backpropagation.

8.3.2.3 Παραλλαγή 2.3: Σύγκριση ELM και BP - IMAGE SEGMENT - 50/50

Στην τελευταία παραλλαγή δώσαμε στα δίκτυα δύο σύνολα δεδομένων ίσα σε πλήθος παρατηρήσεων. Παρόλα αυτά δεν περιμένουμε κάποια αλλαγή στη συμπεριφορά των δικτύων σε σχέση με τις προηγούμενες παραλλαγές, καθώς ο ELM μολονότι εκπαιδεύει καλύτερα από τον BP, αλλά είναι και συντριπτικά πιο ταχύς, δε μπορεί να το μετουσιώσει στα άγνωστα δεδομένα που καλείται να ανταποκριθεί στο στάδιο της εξέτασης.

| 2.3 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|------------|---------|------|---------|------------|-----------|------------|-----------|------------------------|
| ELM | Sigmoid | - | 800 | 0.0555 sec | 94.82% | 0.0035 sec | 69.06% | 0.5903 sec (x10 reps) |
| SCG | - | 0.01 | 100 | 1.4916 sec | 90.61% | 0.0060 sec | 87.05% | 14.9884 sec (x10 reps) |
| Rprop | - | 0.1 | 70 | 1.0130 sec | 90.57% | 0.0064 sec | 88.15% | 10.1932 sec (x10 reps) |
| LM | - | 0.01 | 100 | 7.0227 sec | 95.13% | 0.0066 sec | 89.66% | 00:01:10 (x10 reps) |

Πίνακας 58 2.3 Σύγκριση ELM και BP - IMAGE SEGMENT - 50/50

Συνοψίζοντας

Για το συγκεκριμένο πρόβλημα όπως διαπιστώσαμε ο ELM στις δικές μας προσομοιώσεις, δεν αποδίδει με την ίδια ακρίβεια όπως ο Backpropagation, αλλά ούτε και με τις αντίστοιχες προσομοιώσεις του Huang στο πειραματικό σκέλος με άγνωστα σε αυτών δεδομένα. Μολονότι καταφέρνει να εκπαιδεύσει τις περισσότερες φορές καλύτερα σε σχέση με τον ανταγωνισμό ένα δίκτυο, με πολύ πιο γρήγορες προσομοιώσεις, εν τέλει η αποτελεσματικότητα της εκπαίδευσης δε μετουσιώνεται ορθά.

Όπως θα παρατηρήσουμε και στις επόμενες συγκρίσεις, είναι η μοναδική περίπτωση που δεν ανταποκρίνεται στον ίδιο βαθμό με τον ανταγωνισμό και αποτελεί την εξαίρεση.

8.3.3 Η αναγνώριση της μορφολογίας του εδάφους από δορυφορικές λήψεις

Μίας αντίστοιχης φιλοσοφίας πρόβλημα όπως το προηγούμενο με το διαχωρισμό των αντικειμένων εικόνας, είναι και αυτό του **Satellite Images** για το οποίο διαχωρίζονται σε 6 κατηγορίες οι υπό-εικόνες, από την τμηματοποίηση μεγαλύτερων εικόνων που έχουν τραβηχτεί μέσω δορυφορικών λήψεων. Αυτές οι εικόνες περιέχουν κάποιες ξεχωριστές μορφές τους χρώματος του εδάφους, οι οποίες συγκεντρώνονται σε ένα σύνολο από 6430 παρατηρήσεις, για τις οποίες χρειάζονται 36 γνωρίσματα – χαρακτηριστικά για να τις περιγράψουν.

Χωρίζουμε με τυχαίο τρόπο αλλά αμερόληπτο τα συνολικά δεδομένα στα δυο ξεχωριστά σύνολα προσομοίωσης, ένα για την εκπαίδευση και ένα για την εξέταση. Δημιουργούμε 3 διαφορετικές παραλλαγές με αναλογίες συνόλου δεδομένων εκπαίδευσης / εξέτασης τις 80/20, 68/32 και 50/50. Η αναλογία 68/32 των συνόλων είναι αντίστοιχη με αυτή που εφάρμοσε και ο Huang στις προσομοιώσεις που παρουσίασε την επιστημονική εργασία εισήγησης του ELM.

Θα ακολουθήσουν συγκεντρωτικοί πίνακες που αποτυπώνουμε τις καλύτερες μετρήσεις των αποτελεσμάτων των νευρωνικών αλγορίθμων, από τις προσομοιώσεις που τους ωθήσαμε να κάνουν προτού τις ερμηνεύσουμε.

8.3.3.1 Παραλλαγή 3.1: Σύγκριση ELM και BP - SATELLITE IMAGES - 80/20

| 3.1 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|------------|---------|-----|---------|------------|-----------|------------|-----------|--------------------------|
| ELM | Hardlim | - | 700 | 0.5910 sec | 92.34% | 0.0143 sec | 84.77% | 6.0528 sec (x10 reps) |
| SCG | - | 0.1 | 70 | 9.1603 sec | 86.46% | 0.0075 sec | 85.37% | 00:01:31 (x10 reps) |
| Rprop | - | 0.1 | 70 | 7.7391 sec | 85.94% | 0.0074 sec | 84.79% | 00:01:17 (x10 reps) |
| LM | - | 0.1 | 70 | 00:02:59 | 87.52% | 0.0070 sec | 85.37% | 00:15:58 (x5 reps) |

Πίνακας 59 3.1 Σύγκριση ELM και BP - SATELLITE IMAGES - 80/20

Από τον παραπάνω πίνακα φαίνεται το πόσο συμπαγές είναι το κάθε δίκτυο που εφαρμόζεται ο BP, καθώς χρειάζεται 70 νευρώνες για να μπορέσει να προσομοιώσει σωστά και να προβλέψει τις κατηγορίες των παρατηρήσεων. Αντίθετα ο ELM χρειάζεται 700 νευρώνες για να το καταφέρει αυτό. Παρόλα αυτά είναι 14 – 300 φορές (!) πιο γρήγορος σε σχέση με τον BP. Στην επιτυχία με την ακρίβεια τον προβλέψεων ο ELM είναι 5% – 6% πιο αποτελεσματικός στην εκπαίδευση, και περίπου 0.6% λιγότερο αποτελεσματικός στα νέα δεδομένα. Άρα βλέπουμε ότι όλοι οι αλγόριθμοι είναι πολύ κοντά μεταξύ τους σε επιδόσεις και αλλά χρονικά ο ELM είναι πολύ πιο αποδοτικός σε σχέση με τους άλλους.

8.3.3.2 Παραλλαγή 3.2: Σύγκριση ELM και BP - SATELLITE IMAGES - 68/32

| 3.2 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|------------|---------|-----|---------|------------|-----------|------------|-----------|------------------------|
| ELM | Hardlim | - | 500 | 0.1470 sec | 89.16% | 0.0082 sec | 84.12% | 1.5515 sec (x10 reps) |
| SCG | - | 0.1 | 100 | 5.9215 sec | 86.29% | 0.0092 sec | 84.80% | 59.3070 sec (x10 reps) |
| Rprop | - | 0.1 | 100 | 5.4311 sec | 85.82% | 0.0088 sec | 84.21% | 54.3988 sec (x10 reps) |
| LM | - | 0.1 | 100 | 00:02:18 | 87.48% | 0.0085 sec | 85.08% | 00:11:33 (x5 reps) |
| H - ELM | - | - | 500 | 14.92 sec | 93.52% | 0.34 sec | 89.04% | 15.26 sec |
| H - BP | - | - | 100 | 03:29:21 | 95.26% | 0.08 sec | 82.34% | 03:29:21 |

Πίνακας 60 3.2 Σύγκριση ELM και BP - IMAGE SEGMENT - 68/32

Για τη συγκεκριμένη παραλλαγή του προβλήματος φτάσαμε πιο κοντά με τις προσομοιώσεις του Huang, τόσο σε επίπεδα επιτυχίας όσο και στο όγκο των δικτύων. Χαρακτηριστικά χρειάστηκαν 500 νευρώνες για τις καλύτερες προσομοιώσεις μέσω του ELM και 100 μέσω των BP.

Σχετικά με το ποσοστό επιτυχίας στην εκπαίδευση ο Huang πετυχαίνει ποσοστά 93% – 95% ενώ εμείς 85% – 89%, που σημαίνει ότι εκπαιδεύονται καλύτερα τα δίκτυά του και για του δύο αλγορίθμους. Η επιτυχία στις προβλέψεις με άγνωστα δεδομένα μέσω του ELM φτάνει στο 84% σε εμάς και στο 89% για το Huang, ενώ αντίθετα στο 85% για εμάς με τον BP και στο 82% περίπου για τον Huang. Συνεπώς, καταφέρνουμε να έχουμε πιο όμοια σε αποτελεσματικότητα δίκτυα.

Σε ταχύτητα εκτέλεσης ο ELM είναι πολύ πιο γρήγορος στις προσομοιώσεις του σε σχέση με τον BP και αυτό γίνεται πιο εμφανές αν τον συγκρίνουμε με τον Levenberg – Marquardt όπου καταφέρνει να εκτελέσει μία προσομοίωση περισσότερες από 900 φορές (!) πιο γρήγορα.

8.3.3.3 Παραλλαγή 3.3: Σύγκριση ELM και BP - SATELLITE IMAGES - 50/50

| 3.3 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|------------|---------|-----|---------|------------|-----------|------------|-----------|------------------------|
| ELM | Hardlim | - | 500 | 0.1376 sec | 90.70% | 0.0105 sec | 83.38% | 1.4805 sec (x10 reps) |
| SCG | - | 0.1 | 150 | 4.3116 sec | 86.48% | 0.0111 sec | 84.63% | 43.2273 sec (x10 reps) |
| Rprop | - | 0.1 | 150 | 3.9662 sec | 86.11% | 0.0120 sec | 84.13% | 39.7816 sec (x10 reps) |
| LM | - | 0.1 | 150 | 00:01:37 | 87.73% | 0.0111 sec | 84.69% | 00:08:05 (x5 reps) |

Πίνακας 61 3.3 Σύγκριση ELM και BP - SATELLITE IMAGES - 50/50

Στην τελευταία παραλλαγή του προβλήματος μειώσαμε το πλήθος των παρατηρήσεων με τα οποία εκπαιδεύονται τα δίκτυα. Η διαφοροποίηση η οποία υπήρχε ότι μειώθηκαν ελάχιστα τα ποσοστά επιτυχίας των προβλέψεων τόσο στην εκπαίδευση όσο και στις προσομοιώσεις του πειραματικού σκέλους. Όμως όλοι οι αλγόριθμοι παρέμειναν πολύ κοντά στα ποσοστά επιτυχίας των δικτύων, άρα και ισάξιοι.

8.3.4 Η αναγνώριση των χειρόγραφων προτύπων ψηφίων

Το τελευταίο πρόβλημα ταξινόμησης που εξετάσαμε είναι αυτό μέσω της αναγνώρισης προτύπων των χειρόγραφων ψηφίων. Αυτά τα χειρόγραφα ψηφία έχουν μετατραπεί σε ψηφιακές εικόνες 28 x 28 pixels. Κάθε εικόνα ανάγεται σε ένα διάγραμμα γνωρισμάτων 768 στο πλήθος ώστε να δημιουργηθεί μία παρατήρηση. Υπάρχουν στο σύνολό τους 60000 παρατηρήσεις για την εκπαίδευση των δικτύων που τις χωρίζουμε σε 3 διαφορετικά σύνολα. Ένα με 10000 παρατηρήσεις, ένα με 30000 και ένα με τις 60000. Σε κάθε μία από αυτές τις παραλλαγές υπάρχει ένα σύνολο από 10000 παρατηρήσεις που χρησιμοποιείται ως σύνολο δεδομένων εξέτασης, το οποίο είναι το ίδιο και αμετάβλητο. Το πρόβλημα αυτό είναι γνωστό ως MNIST.

8.3.4.1 Παραλλαγή 4.1: Σύγκριση ELM και BP - MNIST – 10000

Οι πρώτες προσομοιώσεις που κάναμε στα δίκτυα και για τους τέσσερις αλγορίθμους είναι με 10000 παρατηρήσεις στο σύνολο των δεδομένων εκπαίδευσης. Στον ακόλουθο πίνακα συγκεντρώνουμε τις καλύτερες προσπάθειες των αλγορίθμων. Όπως αναφέραμε και στις επιμέρους αναλύσεις, επειδή ο Levenberg – Marquardt είναι πολύ αργός αλγόριθμος γι' αυτού του είδους το πρόβλημα, δεν επιχειρήσαμε να αυξήσουμε το πλήθος των νευρώνων του κρυφού στρώματος πάνω από 20. Μολονότι δέχεται λίγους νευρώνες το δίκτυο που εκπαιδεύεται με τον LM, καταφέρνει πολύ ικανοποιητικά ποσοστά ακρίβειας.

| 4.1 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|------------|---------|------|---------|-------------|-----------|------------|-----------|---------------------|
| ELM | Sigmoid | - | 5000 | 37.4167 sec | 100.00% | 0.6963 sec | 100.00% | 00:06:21 (x10 reps) |
| SCG | - | 0.1 | 200 | 00:01:08 | 96.04% | 0.1076 sec | 96.04% | 00:05:41 (x5 reps) |
| Rprop | - | 0.01 | 200 | 00:01:20 | 82.34% | 0.1058 sec | 82.34% | 00:06:44 (x5 reps) |
| LM | - | 0.01 | 20 | 01:42:30 | 94.35% | 0.0963 sec | 94.35% | 01:42:30 (x1 rep) |

Πίνακας 62 4.1 Σύγκριση ELM και BP - MNIST - 10000

Από τον παραπάνω πίνακα γίνεται ξεκάθαρο ότι στην αποτελεσματικότητα δε μπορεί ν' ακολουθήσει τους υπόλοιπους αλγορίθμους ο Rprop και μένει πίσω. Ταυτόχρονα όλοι οι αλγόριθμοι καταφέρνουν το ίδιο ακριβώς ποσοστό επιτυχίας στην εκπαίδευση αλλά και στην εξέταση με άγνωστα δεδομένα. Ο ELM καταφέρνει να προβλέψει στο 100% όλες τις παρατηρήσεις και μάλιστα το πετυχαίνει με Μ.Ο. προσομοιώσεων κάτω από τα 40 δευτερόλεπτα. Την ίδια στιγμή οι SCG και Rprop χρειάζονται 70 – 80 δευτερόλεπτα, με τον LM να ξεπερνάει τα 100 λεπτά (!). Τέλος, λόγω της ταχύτητας του ELM είχαμε τη δυνατότητα να αυξήσουμε σε πολύ μεγάλο επίπεδο το πλήθος των κρυφών νευρώνων.

8.3.4.2 Παραλλαγή 4.2: Σύγκριση ELM και BP - MNIST – 30000

Σε αυτή την παραλλαγή αυξήσαμε τον όγκο των παρατηρήσεων για την εκπαίδευση, επαναλαμβάνοντας τα παραπάνω πειράματα και για τους τρεις αλγορίθμους. Όπως θα παρατηρήσουμε ο ELM καταφέρνει να εκπαιδεύσει το δίκτυο με 4% καλύτερη ακρίβεια, αλλά και να μπορεί να προσομοιώνει σε νέα δεδομένα με 4% μεγαλύτερη επιτυχία. Όπως θα δούμε είναι πιο γρήγορος αλγόριθμος απ' όλες τις παραλλαγές του BP. Ο Rprop είναι ο πιο αδύναμος απ' όλους τους αλγορίθμους, με τον LM να είναι ο πιο αργός και με μεγάλη απόσταση από τον ELM αλλά και τους υπόλοιπους BP.

| 4.2 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|--------------|----------------|-------------|----------------|-----------------|------------------|-----------------|------------------|-----------------------|
| ELM | Sigmoid | - | 5000 | 00:01:01 | 99.03% | 0.8530 sec | 98.97% | 00:05:12 (x5 reps) |
| SCG | - | 0.1 | 300 | 00:07:23 | 95.79% | 0.1487 sec | 95.95% | 00:36:56 (x5 reps) |
| Rprop | - | 0.01 | 200 | 00:04:27 | 85.84% | 0.1120 sec | 86.21% | 00:22:17 (x5 reps) |
| LM | - | 0.01 | 20 | 05:54:50 | 94.97% | 0.1053 sec | 95.11% | 05:54:50 (x1 rep) |

Πίνακας 63 4.2 Σύγκριση ELM και BP - MNIST - 30000

8.3.4.3 Παραλλαγή 4.3: Σύγκριση ELM και BP - MNIST – 60000

Στα τελευταία πειράματα που κάναμε εφοδιάσαμε το σύνολο των δεδομένων εκπαίδευσης με όλες τις διαθέσιμες παρατηρήσεις. Αυτό είχε ως αποτέλεσμα να εκπαιδευτούν με τον καλύτερο δυνατό τρόπο τα δίκτυα. Έτσι από τα αποτελέσματα επιτυχίας που λάβαμε, η ψαλίδα μεταξύ του ELM και του SCG έκλεισε περισσότερο, με τον ELM να είναι 2% περίπου πιο αποτελεσματικός σε σχέση με τον δεύτερο.

Ωστόσο η διαφορά στην ταχύτητα και τον όγκο νευρώνων είναι χαρακτηριστική και για τους δύο, με τον ELM να πετυχαίνει καλύτερα ποσοστά επιτυχίας στην ακρίβεια των προβλέψεων τόσο στην εκπαίδευση όσο και στην εξέταση χρησιμοποιώντας 5000 νευρώνες. Από την άλλη ο SCG χρειάζεται 300 νευρώνες για να πετύχει τα καλύτερα ποσοστά ακρίβειας, όμως είναι και 12 φορές πιο αργός από τον ELM.

Προσομοιώσεις μέσω του LM δεν κάναμε γι' αυτό το πήραμε, καθώς χρονικά είναι πολύ αργός. Ταυτόχρονα η χρήση λίγων νευρώνων αν και αποδοτική ως ένα βαθμό, δεν είναι η ενδεδειγμένη για τον λόγο ότι χρειάζεται το δίκτυο και ο νευρωνικός αλγόριθμος ένα ικανοποιητικό όγκο νευρώνων για να μπορέσει να προσεγγίσει και να προβλέψει με ασφάλεια σωστά τις πιθανές κατηγορίες των παρατηρήσεων.

| 4.3 | AF | LR | Neurons | Tr. Time | Tr. Acc.% | Ts. Time | Ts. Acc.% | Full Time |
|--------------|----------------|-------------|----------------|-----------------|------------------|-----------------|------------------|-----------------------|
| ELM | Sigmoid | - | 5000 | 00:01:29 | 98.16% | 1.0945 sec | 98.20% | 00:07:35 (x5 reps) |
| SCG | - | 0.01 | 300 | 00:18:03 | 95.98% | 0.1487 sec | 96.12% | 01:30:18 (x3 reps) |
| Rprop | - | 0.1 | 200 | 00:09:45 | 86.34% | 0.1125 sec | 86.64% | 00:28:46 (x3 reps) |
| LM | - | - | - | - | - | - | - | - |

Πίνακας 64 4.3 Σύγκριση ELM και BP - MNIST - 60000

Κεφάλαιο 9

Συνοψίζοντας

Στο τελευταίο κεφάλαιο συνοψίζουμε μέσα απ' όλα τα συμπεράσματα που αποκτήσαμε μετά το πέρας των δοκιμών μας, τα προτερήματα των δύο αλγορίθμων. Ο βασικός άξονας που εξετάσαμε ήταν η αποτελεσματικότητα των αλγορίθμων, αλλά και ο συνδυασμός ταχύτητας, απόδοσης και όγκου των δικτύων για να αποδώσουν οι αλγόριθμοι.

Αρχικά, η αποτελεσματικότητα των αλγορίθμων ήταν σε πολύ υψηλό επίπεδο, σχεδόν σε όλα τα πειράματα που κάναμε τόσο μέσω της συναρτησιακής παλινδρόμησης, όσο και στα προβλήματα ταξινόμησης.

Ο ELM κατάφερε να αποδώσει τα μέγιστα σχεδόν παντού, με εξαίρεση το πρόβλημα ταξινόμησης "Image Segment", στο οποίο παρά το γεγονός ότι είχε εξαιρετικά ποσοστά επιτυχίας στις προβλέψεις της εκμάθησης, δε μπορούσε να γενικεύσει και να αποδώσει στον αντίστοιχο βαθμό στο στάδιο της εξέτασης με άγνωστα δεδομένα.

Αντιθέτως, ο BP κατάφερε να αποδώσει σε όλα τα προβλήματα. Βέβαια αυτό δε συνέβαινε πάντα με για όλες τις παραλλαγές του. Είναι χαρακτηριστικό ότι ο Rprop δε μπορούσε να γενικεύσει στον ίδιο βαθμό στα προβλήματα ταξινόμησης και συναρτησιακής παλινδρόμησης σε σχέση με τον LM. Επιπρόσθετα, μολονότι ο LM απέδιδε καλύτερα στα άγνωστα δεδομένα, στα προβλήματα ταξινόμησης όσο μεγαλύτερο όγκο δεδομένων είχε να επεξεργαστεί, τόσο πιο πολύ αργός γινόταν στην εκτέλεσή του σε σχέση με τον Rprop και το SCG. Πρακτικά δε μπορούσε να εφαρμοστεί ή για να εφαρμοστεί θα έπρεπε το δίκτυο να εφοδιαστεί με έναν πολύ μικρό αριθμό από νευρώνες στο κρυφό στρώμα. Έτσι, η απόδοσή του γινόταν ασταθής και απρόβλεπτη χωρίς να μπορεί να ληφθεί απόλυτα ορθή αυτή η μέθοδος.

Χρονικά, η απόδοση του ELM ήταν πολύ καλύτερη σε σχέση με τις παραλλαγές το BP. Όπως είδαμε η κατανάλωση χρόνου οφείλεται στην εκπαίδευση των δεδομένων από το δίκτυο. Στις προσομοιώσεις με νέα δεδομένα όλοι οι αλγόριθμοι είναι πολύ γρήγοροι και σπάνια να υπάρχει κάποια διαφορά μεταξύ τους. Στην εκπαίδευση όμως η διαφοροποίηση στις χρονικές εκτελέσεις ήταν εμφανής, με τον ELM να είναι πάντα γρηγορότερος από όλες τις παραλλαγές του BP, με τον SCG και Rprop να έχουν μία θετική εικόνα σε αυτόν τον τομέα. Ο LM από την άλλη πλευρά έχει μια δραματική χρονική διάρκεια για την εκτέλεση των εκπαιδύσεων και σε συνδυασμό με την αύξηση όγκου δεδομένων και κρυφών νευρώνων η επιλογή του είναι απευκταία, παρά το γεγονός ότι μπορεί να αποδώσει πολύ καλά.

Στη σύγκριση μεταξύ του ELM και του BP σε προβλήματα συναρτησιακής παλινδρόμησης, παρατηρήσαμε πως έχει μια καλύτερη γενίκευση ο ELM στις γραφικές παραστάσεις των συναρτήσεων. Βλέποντας τις παραστάσεις των προβλέψεων απεικονισμένες, ο ELM προσφέρει πιο κοντά στις πραγματικές παραστάσεις και πιο ομοιόμορφα ακολουθεί τα αποτελέσματα στις απεικονίσεις, μολονότι μπορεί ο BP να έχει ένα πολύ κοντινό τετραγωνικό σφάλμα. Αυτό συμβαίνει γιατί το σφάλμα που προκύπτει από τον ELM είναι ομοιόμορφα κατανομημένο σε όλα τα σημεία της γραφικής παράστασης, ενώ ο BP φτάνει στο σημείο να ταυτίζεται στα σημεία που είναι κοντά στην αρχή των αξόνων, αλλά έχει μεγαλύτερη απόκλιση όσο απομακρύνεται από αυτήν.

Ένα επιπλέον σημείο που χρήζει αναφοράς είναι το σκέλος του μεγέθους ενός δικτύου. Σε αυτό σκέλος η καλύτερη γενικευμένη απόδοση ενός δικτύου στο οποίο έχει εφαρμοστεί ο BP, έρχεται με τη χρήση λιγότερων νευρώνων στο κρυφό στρώμα. Αυτό σημαίνει ότι ο BP μπορεί να αποδώσει σε μικρότερα

και πιο συμπαγή δίκτυα. Αντίθετα, τις περισσότερες φορές η χρήση μεγαλύτερους πλήθους νευρώνων στα δίκτυα που εφαρμόστηκε ο ELM ήταν αναγκαία. Είναι γεγονός πως για να μπορέσει να αποδώσει σωστά ένα δίκτυο μετά την εκπαίδευσή του από τον ELM στο ίδιο επίπεδο με τον BP θα χρειαστεί να είναι μεγαλύτερο σε όγκο. Αυτό ανάλογα την περίπτωση μπορεί να μην μας επηρεάσει, μπορεί όμως να μην το επιθυμούμε λόγω κόστους. Βέβαια η αλόγιστη αύξηση του όγκου ενός δικτύου δεν ενδείκνυται και ειδικότερα για τον BP, καθώς όπως είδαμε στα πειράματα το εύρος του διαστήματος μέγιστης απόδοσης σε σχέση την απόκλιση ενός δικτύου είναι πολύ μικρό. Αντίθετα, σε ένα δίκτυο που έχει εκπαιδευτεί με ένα πλήθος νευρώνων μεγαλύτερο από το αναγκαίο, δεν επηρεάζεται τόσο άμεσα ώστε να αποκλίνει από τα πραγματικά δεδομένα. Συνεπώς, ο BP είναι πιο ευπαθής και είναι αναγκαίο να βρεθεί η καλύτερη δυνατή απόδοση με όσο το δυνατόν μικρότερο σε όγκο δίκτυο.

Κλείνοντας, συμπεράναμε πως ο ELM που πρότεινε ο Huang είναι ένας πολύ αξιόλογος και αποτελεσματικός αλγόριθμος. Βρίσκεται στο ίδιο επίπεδο αποτελεσματικότητας και κάποιες φορές και σε καλύτερο σε σύγκριση με τον BP. Είναι πάντοτε πιο γρήγορος από αυτόν και το πετυχαίνει καθώς εφαρμόζεται σε ένα μεγαλύτερο σε όγκο δίκτυο για να μπορεί η εφαρμοστεί η τυχαιότητα των συναπτικών βαρών και να αποδώσει. Χωρίς την επαναληπτική αναπροσαρμογή βαρών αλλά διορθώνοντας μία φορά τα συναπτικά βάρη, έχει ένα πολύ μεγάλο πλεονέκτημα σε σύγκριση με τον BP. Αυτό στη συντριπτική πλειοψηφία των περιπτώσεων τον βοηθάει στο ίδιο ή και σε ελαφρώς μεγαλύτερο βαθμό από τον BP. Συνεπώς, δεν υπάρχει πρόβλημα στην επιλογή εν τέλει του ενός ή του άλλου, αλλά στην επιλογή των κατάλληλων παραμέτρων για να γενικεύσουν στον αντίστοιχο βαθμό, καθώς και οι δύο είναι εξαιρετικά αποτελεσματικοί υπό τις σωστές συνθήκες.

Βιβλιογραφία

- [1] Κ. ΣΙΔΗΡΟΠΟΥΛΟΥ, Βασικές αρχές λειτουργίας του νευρωνικού συστήματος, Αθήνα, Ζωγράφου: Ελληνικά Ακαδημαϊκά Ηλεκτρονικά Συγγράμματα και Βοηθήματα, 2015.
- [2] Μ. Λουλάκης, ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ, Αθήνα, Ζωγράφου: Ελληνικά Ακαδημαϊκά Ηλεκτρονικά Συγγράμματα και Βοηθήματα, 2015.
- [3] Κ. Διαμαντάρας, ΤΕΧΝΗΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ, Αθήνα: Κλειδάριθμος, 2007.
- [4] T. H. ABRAHAM, THE INTELLECTUAL ORIGINS OF THE McCULLOCH – PITTS NEURAL NETWORKS, vol. 38(1), New York: John Wiley & Sons, Inc., 2002, pp. 3-25.
- [5] geeksforgeeks, «geeksforgeeks.com,» 20 4 2024. [Ηλεκτρονικό]. Available: <https://www.geeksforgeeks.org/adaline-and-madeline-network/>. [Πρόσβαση 2 3 2024].
- [6] S. M. Dionisi, «A COMPARISON OF THE BACKPROPAGATION AND RECURSIVE PREDICTION ERROR ALGORITHMS FOR TRAINING NEURAL NETWORKS,» Academic Press Limited , Sheffield UK, 1991.
- [7] Κ. Διαμαντάρας, «Δίκτυα Συναρτήσρων Βάσης Ακτινικού Τύπου,» Κώστας Διαμαντάρας, Θεσσαλονίκη, 2011.
- [8] Καραγιαννίδης Γεώργιος, Χατζηδιαμαντής Νέστωρ, ΠΙΘΑΝΟΤΗΤΕΣ ΚΑΙ ΣΤΟΧΑΣΤΙΚΕΣ ΔΙΑΔΙΚΑΣΙΕΣ, 1 επιμ., Αθήνα: ΕΚΔΟΣΕΙΣ ΤΣΙΟΛΑ, 2022.
- [9] Π. ΑΡΓΥΡΑΚΗΣ, Νευρωνικά Δίκτυα και Εφαρμογές, τόμ. ΙΙ, Πάτρα: ΠΑΝΟΣ ΑΡΓΥΡΑΚΗΣ, 2001.
- [10] I. Tetko, "Associative Neural Network," *Methods in molecular biology (Clifton, N.J.)*, pp. 187 - 199, Ιανουάριος 2009.
- [11] A. Tharwat, «Independent component analysis,» Emerald Publishing, Frankfurt, 2018.
- [12] P. Comon, «Independent component analysis,» 24 8 1994. [Ηλεκτρονικό]. Available: <http://mlsp.cs.cmu.edu/courses/fall2012/lectures/ICA.pdf>. [Πρόσβαση 2024 3 1].
- [13] S. Haykin, Νευρωνικά Δίκτυα και Μηχανική Μάθηση, 3 ed., Αθήνα: Παπασωτηρίου Εκδόσεις, 2010.
- [14] J. Brownlee, «machinelearningmastery,» 19 7 2019. [Ηλεκτρονικό]. Available: <https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/>. [Πρόσβαση 2 3 2024].
- [15] Gail A. Carpenter, Stephen Grossberg, «Adaptive Resonance Theory,» Gail A. Carpenter, Stephen Grossberg, Boston, 2016.
- [16] Cheolhwan Oh, Stefen Hui, Stanis aw H. ak, "The Generalized Brain-State-in-a-Box (gBSB) Neural Network: Model, Analysis, and Applications," Cheolhwan Oh, Stefen Hui, Stanis aw H. ak, San Diego, 2005.

- [17] D. Nualart, «Kolmogorov and Probability Theory,» 8 2004. [Ηλεκτρονικό]. Available: <https://core.ac.uk/download/pdf/268083255.pdf>. [Πρόσβαση 2 3 2024].
- [18] O. t. R. Team, «iq.opengenus,» . [Ηλεκτρονικό]. Available: <https://iq.opengenus.org/types-of-backpropagation/>. [Πρόσβαση 2 3 2024].
- [19] E. Gomed, «medium.com,» 20 8 2023. [Ηλεκτρονικό]. Available: <https://medium.com/@evertongomed/variations-of-backpropagation-enhancing-neural-network-training-ff082653c88a>. [Πρόσβαση 2 3 2024].
- [20] H. P. Gavin, «The Levenberg-Marquardt algorithm for nonlinear least squares curve-fitting problems,» Henri P. Gavin, Duke, 2022.
- [21] «mathworks.com,» mathworks, . [Ηλεκτρονικό]. Available: <https://www.mathworks.com/help/deeplearning/ref/trainlm.html>. [Πρόσβαση 2 3 2024].
- [22] U. Ijaz, «educative.io,» . [Ηλεκτρονικό]. Available: <https://www.educative.io/answers/what-is-the-jacobian-matrix>. [Πρόσβαση 2 3 2024].
- [23] Christian Igel and Michael Husken, «Improving the Rprop Learning Algorithm,» Bochum.
- [24] Christian Igel and Michael Husken, «Improving the Rprop Learning Algorithm,» Proceedings of the Second International Symposium on Neural Computation, Bochum, 2000.
- [25] M. F. Møller, "A Scaled Conjugate Gradient Algorithm," in *Neural Network*, vol. 6, Aarhus, Pergamon Press Ltd., 1991, pp. 525-533.
- [26] J. R. Shewchuk, «An Introduction to the Conjugate Gradient Method Without the Agonizing Pain,» Jonathan Richard Shewchuk, Pittsburgh, 1994.
- [27] Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, «Extreme learning machine: Theory and applications,» Elsevier B.V, Singapore, 2006.
- [28] Jian Wang, Siyuan Lu, Shui-Hua Wang, Yu-Dong Zhang, «A review on extreme learning machine,» 22 5 2021. [Ηλεκτρονικό]. Available: https://www.researchgate.net/publication/351774636_A_review_on_extreme_learning_machine . [Πρόσβαση 1 3 2024].
- [29] Shifei Ding, Xinzheng Xu, Ru Nie, «Extreme learning machine and its applications,» σε *Extreme learning machine and its*, London, Online, Springer-Verlag, 2013, p. 549–556.
- [30] André Frank Krause, Kai Essig, Martina Piefke, Thomas Schack, «No-Prop-fast - a high-speed multilayer neural network learning algorithm: MNIST benchmark and eye-tracking data classification,» Authors, Bielefeld, 2013.
- [31] Bernard Widrow, Aaron Greenblatt, Youngsik Kim, Dookun Park, «The No-Prop algorithm: A new learning algorithm for multilayer neural networks,» Elsevier Ltd, California, United States, 2012.
- [32] Ν. Η. Παπαμάρκος, Ψηφιακή επεξεργασία & ανάλυση εικόνας, 3 επιμ., Ξάνθη: Παπαμάρκος, Νικόλαος, 2013.

- [33] Rafael C. Gonzalez, Richard E. Woods, ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΕΙΚΟΝΑΣ, 4 ed., Αθήνα: Εκδόσεις ΤΖΙΟΛΑ, 2018.
- [34] J. McCaffrey, «Preparing MNIST Image Data Text Files,» Converge360, 1105 Media company, 2022. [Ηλεκτρονικό]. Available: <https://visualstudiomagazine.com/articles/2022/02/01/preparing-mnist-image-data-text-files.aspx>. [Πρόσβαση 13 2024].