



Πολυτεχνείο Κρήτης

Τμήμα Μηχανικών Περιβάλλοντος

Π.Μ.Σ. «Περιβαλλοντική Μηχανική»

«Διαχείριση Νερού και Υγρών Αποβλήτων»

Μεταπτυχιακή Διπλωματική Εργασία

«Χρήση τεχνητών νευρωνικών δικτύων για τον προσδιορισμό της στάθμης του υδροφόρου, στην περιοχή της λεκάνης απορροής του Δούναβη»

Δημήτριος Αρ. Γαλάνης

Εξεταστική Επιτροπή

Καρατζάς Γεώργιος (Επιβλέπων)

Νικολαΐδης Νικόλαος

Τριχάκης Ιωάννης

Χανιά, Απρίλιος 2021

"Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για μη κερδοσκοπικό σκοπό, εκπαιδευτικού ή ερευνητικού χαρακτήρα, με την προϋπόθεση να αναφέρεται η πηγή προέλευσης. Ερωτήματα που αφορούν τη χρήση της εργασίας για άλλη χρήση θα πρέπει να απευθύνονται προς το συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πολυτεχνείου Κρήτης".

"Those who can imagine anything, can create the impossible"
Alan Turing

ΕΥΧΑΡΙΣΤΙΕΣ

Ολοκληρώνοντας την παρούσα μεταπτυχιακή διπλωματική εργασία καθώς και τον κύκλο μεταπτυχιακών σπουδών θα ήθελα να ευχαριστήσω όλους εκείνους που με στήριξαν με υλικό ή ηθικό τρόπο κατά την διάρκεια αυτής της πορείας.

Αρχικά θα ήθελα να ευχαριστήσω θερμά, τον κύριο Γεώργιο Καρατζά, Καθηγητή του Τμήματος Μηχανικών Περιβάλλοντος, για την άψογη συνεργασία και την πολύτιμη βοήθεια που μου προσέφερε σε όλη την διάρκεια αυτού του κύκλου σπουδών καθώς και την άψογη συνεργασία στην εκπόνηση της διπλωματικής αυτής εργασίας.

Ένα μεγάλο ευχαριστώ οφείλω και στον Ιωάννη Τριχάκη, που διέθεσε αρκετό από το χρόνο του, μεταλαμπαδεύοντας την πολύτιμη, για την εργασία αυτή εμπειρία και γνώση του, στον τομέα της τεχνητής νοημοσύνης και συγκεκριμένα στα τεχνητά νευρωνικά δίκτυα.

Ευχαριστώ επίσης τον Νικόλαο Νικολαΐδη για την παρουσία του στην τριμελή μου επιτροπή και για το χρόνο που διέθεσε για την αξιολόγηση της παρούσας εργασίας.

Ιδιαίτερη μνεία στους Παναγιώτη Γιαννακούλη και Νικόλαο Ξυνίδη για την βοήθεια που μου προσέφεραν κατά την διάρκεια αυτού του μεταπτυχιακού κύκλου σπουδών, χωρίς την οποία δεν θα ήταν εφικτή η ολοκλήρωση του.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς μου και την οικογένεια μου, για την ηθική και ψυχολογική τους υποστήριξη καθώς και σε όλα τους φίλους μου, που σταθήκαν δίπλα μου σε αυτό το εγχείρημα.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΕΥΧΑΡΙΣΤΙΕΣ	III
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ.....	IV
ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ.....	VI
ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ	VII
ΕΥΡΕΤΗΡΙΟ ΔΙΑΓΡΑΜΜΑΤΩΝ	VIII
ΠΕΡΙΛΗΨΗ	XI
ABSTRACT.....	XII
ΕΙΣΑΓΩΓΗ	1
ΝΕΡΟ	1
ΥΔΡΟΛΟΓΙΚΟΣ ΚΥΚΛΟΣ.....	3
ΥΠΟΓΕΙΑ ΥΔΑΤΑ.....	4
ΜΟΝΤΕΛΑ	6
ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ	7
ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ	7
Ο ΑΝΘΡΩΠΙΝΟΣ ΕΓΚΕΦΑΛΟΣ	8
ΤΕΧΝΗΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ.....	10
Η Γένεση.....	10
Οι Πρώτες Προσπάθειες	11
Το Perceptron.....	12
Το Adaline, ο Κανόνας <i>Widrow-Hoff</i> και ο Αλγόριθμος LMS	12
Συναρτήσεις Ενεργοποίησης	15
Αλγόριθμος της Απότομης Καθόδου	19
Ρυθμός Μάθησης	23
Ο Παράγοντας της Ορμής.....	24
Μεταβλητός Ρυθμός Μάθησης.....	25
Η Μεθοδος Back Propagation	26
Γενίκευση του Σφάλματος	28
Βαθιά Μάθηση.....	32
Προηγμένοι Μέθοδοι Σύγκλισης	33
ΠΕΡΙΟΧΗ ΜΕΛΕΤΗΣ	42
Ο ΔΟΥΝΑΒΗΣ	42
ΠΕΙΡΑΜΑΤΙΚΗ ΔΙΑΔΙΚΑΣΙΑ	44
ΠΡΟΕΤΟΙΜΑΣΙΑ ΔΕΔΟΜΕΝΩΝ	44

Διαλογή Δεδομένων.....	44
Ανάγνωση Χρονοσειρών	47
Ενσωμάτωση Μετεωρολογικών Δεδομένων	49
ΧΡΗΣΗ ΤΟΥ <i>MATLAB</i>.....	50
Περιγραφή του Εργαλείου <i>nntool</i>	51
Παραμετροποίηση Αλγορίθμων	52
Δημιουργία Ρηχών T.N.Δ	58
Δημιουργία T.N.Δ με Δύο Κρυφά Επίπεδα.....	72
Σύγκριση Αποτελεσμάτων.....	86
Η Ιδιαίτερη Περίπτωση της Ανθεκτικής Διάδοσης	88
ΠΕΡΕΤΑΙΡΩ ΔΟΚΙΜΕΣ ΤΗΣ ΜΕΘΟΔΟΥ <i>LEVENBERG-MARQUARDT</i>.....	90
1 ^ο Σενάριο	93
2 ^ο Σενάριο	95
3 ^ο Σενάριο	97
4 ^ο Σενάριο	99
5 ^ο Σενάριο	101
ΕΛΕΓΧΟΣ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	103
ΑΠΟΔΟΣΗ ΑΛΓΟΡΙΘΜΩΝ ΣΕ ΠΗΓΑΔΙΑ ΠΟΥ ΣΥΜΜΕΤΕΙΧΑΝ ΣΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΩΝ T.N.Δ.	103
ΑΠΟΔΟΣΗ ΑΛΓΟΡΙΘΜΩΝ ΣΤΟΥΣ ΣΤΑΘΜΟΥΣ ΤΗΣ ΟΥΓΓΑΡΙΑΣ	109
ΑΠΟΔΟΣΗ ΑΛΓΟΡΙΘΜΩΝ ΣΤΟΥΣ ΣΥΝΟΡΙΑΚΟΥΣ ΣΤΑΘΜΟΥΣ ΤΗΣ ΟΥΓΓΑΡΙΑΣ	114
ΣΥΜΠΕΡΑΣΜΑΤΑ – ΠΡΟΤΑΣΕΙΣ	120
ΒΙΒΛΙΟΓΡΑΦΙΑ	122
ΠΑΡΑΡΤΗΜΑΤΑ	129
ΑΛΓΟΡΙΘΜΟΙ.....	130
Python	130
MATLAB.....	137

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ

Εικόνα 1: Κατανομή του νερού της Γης.....	1
Εικόνα 2: Περιοχές που αντιμετωπίζουν φυσική ή οικονομική λειψυδρία.....	2
Εικόνα 3: Τα πέντε κομμάτια του κλιματικού συστήματος και οι αλληλεπιδράσεις τους.....	3
Εικόνα 4: Παγκόσμιες υδρολογικές ροές και αποθήκες νερού	4
Εικόνα 5: Σχηματική απεικόνιση των σχέσεων μεταξύ των κλιματικών παραμέτρων και των επιδράσεων στην ποσότητα των υπογείων υδάτων	5
Εικόνα 6: Σχηματική απεικόνιση δύο βιολογικών νευρώνων	8
Εικόνα 7: Οι γραμμές συμβολίζουν το άθροισμα των εκφορτίσεων σε καθένα από τα συστατικά του σημείου B και την επακόλουθη ισχύ του συνδυασμού επιρροών με τις οποίες αφυπνίζεται το σύνολο του B.	9
Εικόνα 8: Ένας από τους νευρώνες του Sparc.....	10
Εικόνα 9: Σχεδιάγραμμα της 1 ^{ης} ψηφιοποίησης ενός αυτοοριζόμενου δικτύου 8 στοιχείων	11
Εικόνα 10: Οργάνωση ενός Perceptron	12
Εικόνα 11: Ένας προσαρμοστικός νευρώνας	13
Εικόνα 12: Οι συνάρτησεις ενεργοποίησης Heaviside & Signum	15
Εικόνα 13: Οι συναρτήσεις ενεργοποίησης της συμμετρικής Heaviside και η γραμμική συνάρτηση.....	16
Εικόνα 14: Η συμμετρική συνάρτηση κορεσμού και η θετικά γραμμική	17
Εικόνα 15: Η λογιστική συνάρτηση ενεργοποίησης και η συνάρτηση ενεργοποίησης υπερεφαπτομένης.....	18
Εικόνα 16: Γραφική παράσταση της συνάρτησης σφάλματος για δισδιάστατη περίπτωση.	20
Εικόνα 17: Αποτύπωση του κώνου σφάλματος της εικόνας 16 με ισοβαρείς καμπύλες σφάλματος.....	21
Εικόνα 18: Επομένως η μεγαλύτερη απόσταση θα διανυθεί, έναν η μετακίνηση γίνει κάθετα στην εφαπτομένη	21
Εικόνα 19: Γραφικές παραστάσεις του ρυθμού μάθησης.....	23
Εικόνα 20: Επίδραση της ορμής	24
Εικόνα 21: Ένα πολυεπίπεδο T.N.Δ. με N_i μεταβλητές εισόδου, N_o μεταβλητές εξόδου και 1 επίπεδα	26
Εικόνα 22: Κλασικά παραδείγματα στα οποία παρατηρείται το φαινόμενο της υπερεκπαίδευσης.....	28
Εικόνα 23: Εφαρμογή της «εγκατάλειψης».....	29
Εικόνα 24: Το κριτήριο του πρόωρου τερματισμού	31
Εικόνα 25: Διαφορά ενός ρηχού με ένα βαθύ νευρωνικό δίκτυο	32
Εικόνα 26: Λεκάνη απορροής του Δούναβη, διακρίνεται ο ποταμός και οι παραπόταμοι του, οι χώρες που διαπερνά αυτός και οι παραπόταμοι του	41
Εικόνα 27: Χάρτες μέσης θερμοκρασίας και μέσης βροχόπτωσης.....	42
Εικόνα 28: Η περιοχή μελέτης με τις χώρες, τον Δούναβη και τους κύριους παραποτάμους του, όπως εισήχθη στο πρόγραμμα ArcGIS..	43
Εικόνα 29: Η θέση των πηγαδιών που χρησιμοποιήθηκαν για την εκπαίδευση των τεχνητών νευρωνικών δικτύων.	46
Εικόνα 30: Αρχεία χρονοσειρών της Αυστρίας, είναι εμφανής η διαφοροποίηση στην έναρξη των έγγραφων του εκάστοτε σταθμού, καθώς επίσης είναι εμφανής και η ύπαρξη αλφαριθμητικών χαρακτήρων, ανάμεσα στα δεδομένα.	47

Εικόνα 31: Αρχεία χρονοσειρών της Γερμανίας, στην πρώτη περίπτωση φαίνονται οι μηδενικές καθώς και οι διπλές εγγραφές, ενώ στην δεύτερη περίπτωση εμφανίζονται κάποιες μετρήσεις που ο μετρητής μετατρέπει τα μέτρα σε χιλιοστά	48
Εικόνα 32: Εισαγωγή δεδομένων στο MATLAB για εκπαίδευση νευρωνικών δικτύων.....	50
Εικόνα 33: Αριστερά εμφανίζεται το κεντρικό μενού του εργαλείου nntool, δεξιά εμφανίζονται οι επιλογές δημιουργίας νέου T.N.Δ.	51
Εικόνα 34: Παραμετροποίηση της μεθόδου της απότομης καθόδου	52
Εικόνα 35: Παραμετροποίηση της μεθόδου της απότομης καθόδου με ορμή	53
Εικόνα 36: Παραμετροποίηση της μεθόδου της απότομης καθόδου με μεταβλητό ρυθμό μάθησης	53
Εικόνα 37: Παραμετροποίηση της μεθόδου της απότομης καθόδου με μεταβλητό ρυθμό μάθησης	54
Εικόνα 38: Παράθυρο με τις επιλογές παραμετροποίησης των μεθόδων συζυγούς κλίσης με επανεκκινήσεις.....	55
Εικόνα 39: Παράθυρο με τις επιλογές παραμετροποίησης των μεθόδων της οικογένειας Quasi-Newton	56
Εικόνα 40: Παραμετροποίηση της μεθόδου Κλιμακωτής Συζυγούς Κλίσης.....	56
Εικόνα 41: Παραμετροποίηση των μεθόδων Levenberg-Marquardt.....	57
Εικόνα 42: Παραμετροποίηση της μεθόδου Ανθεκτικής Διάδοσης.....	58
Εικόνα 43: Φωτογραφία από μοντέλο με ένα επίπεδο που έτρεξε σε υπολογιστή του εργαστηρίου	90
Εικόνα 44: Τα πηγάδια που χρησιμοποιήθηκαν για τον έλεγχο της απόδοσης των ΤΝΔ σε περιοχή των συνόρων της Ουγγαρίας.....	114
Εικόνα 45: Δημιουργία λίστας των χρονοσειρών με χρήση MS-DOS.....	130
Εικόνα 46: Λίστα των χρονοσειρών με ενσωματωμένη πληροφορία των συντεταγμένων του εκάστοτε σταθμού.....	130

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

Πίνακας 1: Οι χώρες που διέρχεται ο Δούναβης, αριθμός πηγαδιών, αριθμός διαθέσιμων χρονοσειρών και αριθμός σταθμών που χρησιμοποιήθηκαν.....	45
Πίνακας 2: Διαφορές μεταξύ των δύο εργαλείων του υπολογιστικού περιβάλλοντος της MATLAB.....	50
Πίνακας 3: Συνοπτικά αποτελέσματα της εκπαίδευσης ενός T.N.Δ. για την περιοχή μελέτης με χρήση των διαθέσιμων αλγορίθμων του MATLAB. Παρουσιάζεται η απόδοση των αλγορίθμων σε MSE και απαιτούμενο χρόνο εκπαίδευσης.....	86
Πίνακας 4: Συγκεντρωτικά αποτελέσματα της χρήσης του αλγορίθμου ανθεκτικής διάδοσης	88
Πίνακας 5: Δοκιμές διαφορών αρχιτεκτονικών T.N.Δ. με 25000 ζεύγη δεδομένων. Με έντονο χρωματισμό οι 3 καλύτερες μέθοδοι σε μέσο όρο.....	91

ΕΥΡΕΤΗΡΙΟ ΔΙΑΓΡΑΜΜΑΤΩΝ

Διάγραμμα 1: Αποτελέσματα του αλγόριθμου απότομης καθόδου.....	59
Διάγραμμα 2: Αποτελέσματα του αλγόριθμου απότομης καθόδου με ορμή	60
Διάγραμμα 3: Αποτελέσματα του αλγόριθμου απότομης καθόδου με μεταβλητό ρυθμό μάθησης	61
Διάγραμμα 4: Αποτελέσματα του αλγόριθμου απότομης καθόδου με μεταβλητό ρυθμό μάθησης και ορμή	62
Διάγραμμα 5: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου Beale-Powell	63
Διάγραμμα 6: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου Fletcher-Reeves.....	64
Διάγραμμα 7: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου Polak-Ribiere	65
Διάγραμμα 8: Αποτελέσματα του αλγόριθμου βαθμισμένης συζυγούς κλίσης	66
Διάγραμμα 9: Αποτελέσματα του αλγόριθμου Quasi-Newton.....	67
Διάγραμμα 10: Αποτελέσματα του αλγόριθμου One-Step Secant	68
Διάγραμμα 11: Αποτελέσματα του αλγόριθμου Levenberg-Marquardt.....	69
Διάγραμμα 12: Αποτελέσματα του αλγόριθμου Bayesian Regularization.....	70
Διάγραμμα 13: Αποτελέσματα του αλγόριθμου Ανθεκτικής Διάδοσης (RProp)	71
Διάγραμμα 14: Αποτελέσματα του αλγόριθμου απότομης καθόδου και δύο κρυφά επίπεδα.	72
Διάγραμμα 15: Αποτελέσματα του αλγόριθμου απότομης καθόδου με ορμή και δύο κρυφά επίπεδα	73
Διάγραμμα 16: Αποτελέσματα του αλγόριθμου απότομης καθόδου με μεταβλητό ρυθμό μάθησης και δύο κρυφά επίπεδα.....	74
Διάγραμμα 17: Αποτελέσματα του αλγόριθμου καθόδου με μεταβλητό ρυθμό μάθησης, ορμή και δύο κρυφά επίπεδα (1 ^η δοκιμή)	75
Διάγραμμα 18: Αποτελέσματα του αλγόριθμου καθόδου με μεταβλητό ρυθμό μάθησης, ορμή και δύο κρυφά επίπεδα (2 ^η δοκιμή)	76
Διάγραμμα 19: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου Beale-Powell και δύο κρυφά επίπεδα	77
Διάγραμμα 20: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου Fletcher-Reeves και δύο κρυφά επίπεδα.....	78
Διάγραμμα 21: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου Polak-Ribiere και δύο κρυφά επίπεδα	79
Διάγραμμα 22: Αποτελέσματα του αλγόριθμου βαθμισμένης συζυγούς κλίσης με δύο κρυφά επίπεδα	80
Διάγραμμα 23: Αποτελέσματα του αλγόριθμου Quasi-Newton με δύο κρυφά επίπεδα	81
Διάγραμμα 24: Αποτελέσματα του αλγόριθμου One-Step Secant με δύο κρυφά επίπεδα....	82
Διάγραμμα 25: Αποτελέσματα του αλγόριθμου Levenberg-Marquardt με δύο κρυφά επίπεδα	83
Διάγραμμα 26: Αποτελέσματα του αλγόριθμου Bayesian Regularization με δύο κρυφά επίπεδα	84
Διάγραμμα 27: Αποτελέσματα του αλγόριθμου Ανθεκτικής Διάδοσης με δύο κρυφά επίπεδα	85
Διάγραμμα 28: Συγκριτικό διάγραμμα των αλγορίθμων με κριτήριο την απόδοση	87
Διάγραμμα 29: Συγκριτικό διάγραμμα των μεθόδων με κριτήριο το χρόνο σε λεπτά	87

Διάγραμμα 30: Σύγκριση αποτελεσμάτων εκπαίδευσης ως προς το χρόνο, T.N.Δ. που εκπαιδεύτηκαν με τον αλγόριθμο της ανθεκτικής διάδοσης	89
Διάγραμμα 31: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 1 ^ο σενάριο.....	93
Διάγραμμα 32: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 1 ^ο σενάριο ...	94
Διάγραμμα 33: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 2 ^ο σενάριο.....	95
Διάγραμμα 34: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 2 ^ο σενάριο ...	96
Διάγραμμα 35: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 3 ^ο σενάριο.....	97
Διάγραμμα 36: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 3 ^ο σενάριο ...	98
Διάγραμμα 37: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 4 ^ο σενάριο.....	99
Διάγραμμα 38: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 4 ^ο σενάριο ...	100
Διάγραμμα 39: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 5 ^ο σενάριο.....	101
Διάγραμμα 40: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 5 ^ο σενάριο ...	102
Διάγραμμα 41: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 1 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.	104
Διάγραμμα 42: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 1 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.	104
Διάγραμμα 43: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 2 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.	105
Διάγραμμα 44: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 2 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.	105
Διάγραμμα 45: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 3 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.	106
Διάγραμμα 46: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 3 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.	106
Διάγραμμα 47: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 4 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.	107
Διάγραμμα 48: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 4 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.	107
Διάγραμμα 49: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 5 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.	108
Διάγραμμα 50: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 5 ^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.	108
Διάγραμμα 51: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 1 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	109
Διάγραμμα 52: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 2 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	110
Διάγραμμα 53: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 3 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	111

Διάγραμμα 54: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 4 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	112
Διάγραμμα 55: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 5 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	113
Διάγραμμα 56: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 1 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	115
Διάγραμμα 57: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 2 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	116
Διάγραμμα 58: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 3 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	117
Διάγραμμα 59: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 4 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	118
Διάγραμμα 60: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 5 ^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization).....	119

ΠΕΡΙΛΗΨΗ

Η παρακολούθηση των υπογείων υδάτων, όσον αφορά την ποσότητα τους καθώς και την ποιότητα τους, είναι υποχρεωτική με βάση την ελληνική αλλά και την ευρωπαϊκή νομοθεσία. Η ακριβής παρακολούθηση τους όμως δεν είναι εφικτή, καθώς δεν είναι δυνατή η μέτρηση της στάθμης τους και των ποιοτικών χαρακτηριστικών τους, σε κάθε σημείο της εκάστοτε περιοχής ενδιαφέροντος. Αντ' αυτού σχεδιάζονται κατάλληλα υπολογιστικά μοντέλα που μπορούν να δώσουν ακριβή εκτίμηση τόσο της ποσότητας, όσο και της ποιότητας τους.

Τα τελευταία χρόνια αντί της χρησιμοποίησης των παραδοσιακών αριθμητικών μοντέλων, που βασίζονται στην προσεγγιστική επίλυση της εξίσωσης της υπόγειας ροής, ολοένα κερδίζει έδαφος η χρήση της τεχνητής νοημοσύνης και συγκεκριμένα των τεχνητών νευρωνικών δικτύων (T.N.Δ.), για την εκτίμηση τόσο των ποσοτικών όσο και των ποιοτικών χαρακτηριστικών των υπογείων υδάτων.

Στην παρούσα εργασία, πραγματοποιείται μια σύντομη καταγραφή των σημαντικότερων καινοτομιών στον τομέα των T.N.Δ., καθώς και όλων των εξελίξεων που τα καθιέρωσαν ως το βασικότερο εργαλείο χρήσης της τεχνητής νοημοσύνης σε διάφορους τομείς της επιστήμης. Στην συνέχεια αναλύονται οι μέθοδοι εκπαίδευσης που χρησιμοποιούνται στην μηχανική εκμάθηση καθώς και καταγραφή των βασικών εξισώσεων που χρησιμοποιούνται από τους εκάστοτε αλγόριθμους εκπαίδευσης.

Αφού δοθούν κάποια γενικά στοιχεία για την περιοχή της λεκάνης απορροής του Δούναβη (η οποία χρησιμοποιήθηκε ως περιοχή μελέτης στην παρούσα εργασία), περιγράφεται αναλυτικά ο τρόπος με τον οποίο έγινε η οργάνωση των δεδομένων, ώστε να μπορούν να χρησιμοποιηθούν για μηχανική εκμάθηση. Στην συνέχεια περιγράφεται ο τρόπος με τον οποίο μπορεί να γίνει χρήση T.N.Δ. μέσω του MATLAB και γίνεται κάποιος έλεγχος ώστε να βρεθεί ο κατάλληλος αλγόριθμος για την εκπαίδευση του T.N.Δ. καθώς και η επιλογή της κατάλληλης αρχιτεκτονικής του.

Με την επιλογή και χρήση του κατάλληλου αλγορίθμου, εξετάζονται κάποια σενάρια ώστε να μπορεί να επιτευχθεί σωστή εκτίμηση του υδροφόρου ορίζοντα σε όλη την περιοχή μελέτης. Το ερευνητικό κομμάτι ολοκληρώνεται με την σύγκριση των εκτιμήσεων των μοντέλων με πραγματικές τιμές, ώστε να ελεγχθεί η εγκυρότητα των αποτελεσμάτων τους.

Τέλος γίνεται μια ανασκόπηση της εργασίας, παρατηρήσεις από τα πειραματικά αποτελέσματα, καθώς προτάσεις αλλαγών και σημεία αποφυγής για μελλοντικές μελέτες ώστε να βελτιωθεί σημαντικά η εκτίμηση των T.N.Δ.

ABSTRACT

The monitoring of groundwater, in terms of their quantity and quality, is mandatory based on Greek and European legislation. However, their accurate monitoring of their level as well as their quality characteristics, is not possible, in every point of the respective study area. Instead, suitable computer models are designed that can give an accurate estimate of both quantity and quality.

In recent years, instead of using traditional arithmetic models, based on the approximate solution of the underground flow equation, the use of artificial intelligence and in particular artificial neural networks (A.N.N.), for the estimation of both quantitative and qualitative characteristics of groundwater is gaining advantage.

In the present work, a brief record of the most important innovations in the field of A.N.N., as well as all the developments that have established them as the main tool for the use of artificial intelligence in various fields of science, is listed. Afterwards, an analysis of the training methods used in machine learning is been made, followed up with the analysis of the basic equations used by the respective training algorithms.

In the next part of the study, a brief report of the Danube river basin (which was used as study area) is been made. While the method used to organize the data in order to be able to be used in machine learning, is been described in detail. Followed up with a description of the MATLAB A.N.N. tool and the selection of the appropriate algorithm as well as the proper architecture of the A.N.N.

A number of scenarios are been examined, in order to achieve a correct simulation of the groundwater levels in the whole study area. Followed up, by a validation part, in which a comparison of the predicted data with unused real data is been made.

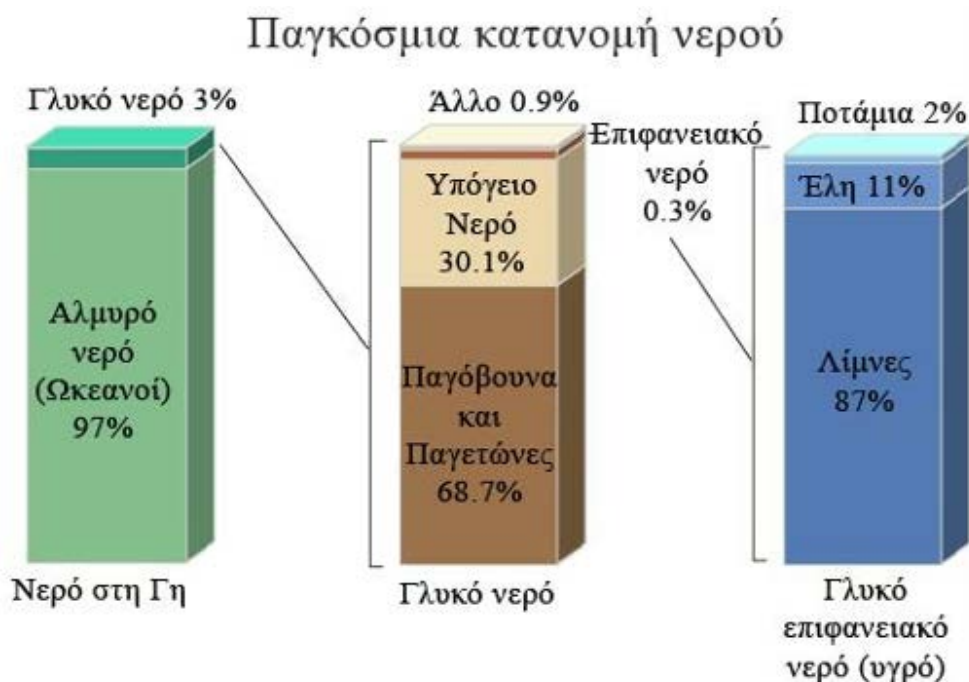
In conclusion, there is a review of this study. Including observations from the experimental results, as well as suggestions for changes and avoidance points for future studies in order to significantly improve the assessment of A.N.N.

ΕΙΣΑΓΩΓΗ

ΝΕΡΟ

Το νερό είναι ένα από τα δύο βασικά στοιχεία (το άλλο είναι το οξυγόνο) τα οποία είναι απαραίτητα για την διαβίωση του ανθρώπου αλλά και σχεδόν όλων των ζωντανών οργανισμών. Δυστυχώς όμως τα τελευταία χρόνια σε αρκετές περιοχές του πλανήτη, ανάμεσα σε αυτές και ορισμένα κομμάτια της Ελλάδας, όπως η ανατολική Κρήτη και άλλα νησιά του Αιγαίου, είτε συνέχεια, είτε εποχιακά παρατηρείτε έλλειψη νερού (Εικόνα 2) (Tzanakakis, et al., 2020; Grillakis, 2019; Koutroulis, 2019). Σύμφωνα με το Παγκόσμιο Οικονομικό Φόρουμ, η λειψυδρία είναι η 8ή πιο πιθανή αιτία για δημιουργία οικονομικής κρίσης και βρίσκεται στην κορυφή της λίστας του οικονομικού αντίκτυπου που θα προκαλέσει (W.E.F., 2015; Mekonnen & Hoekstra, 2016).

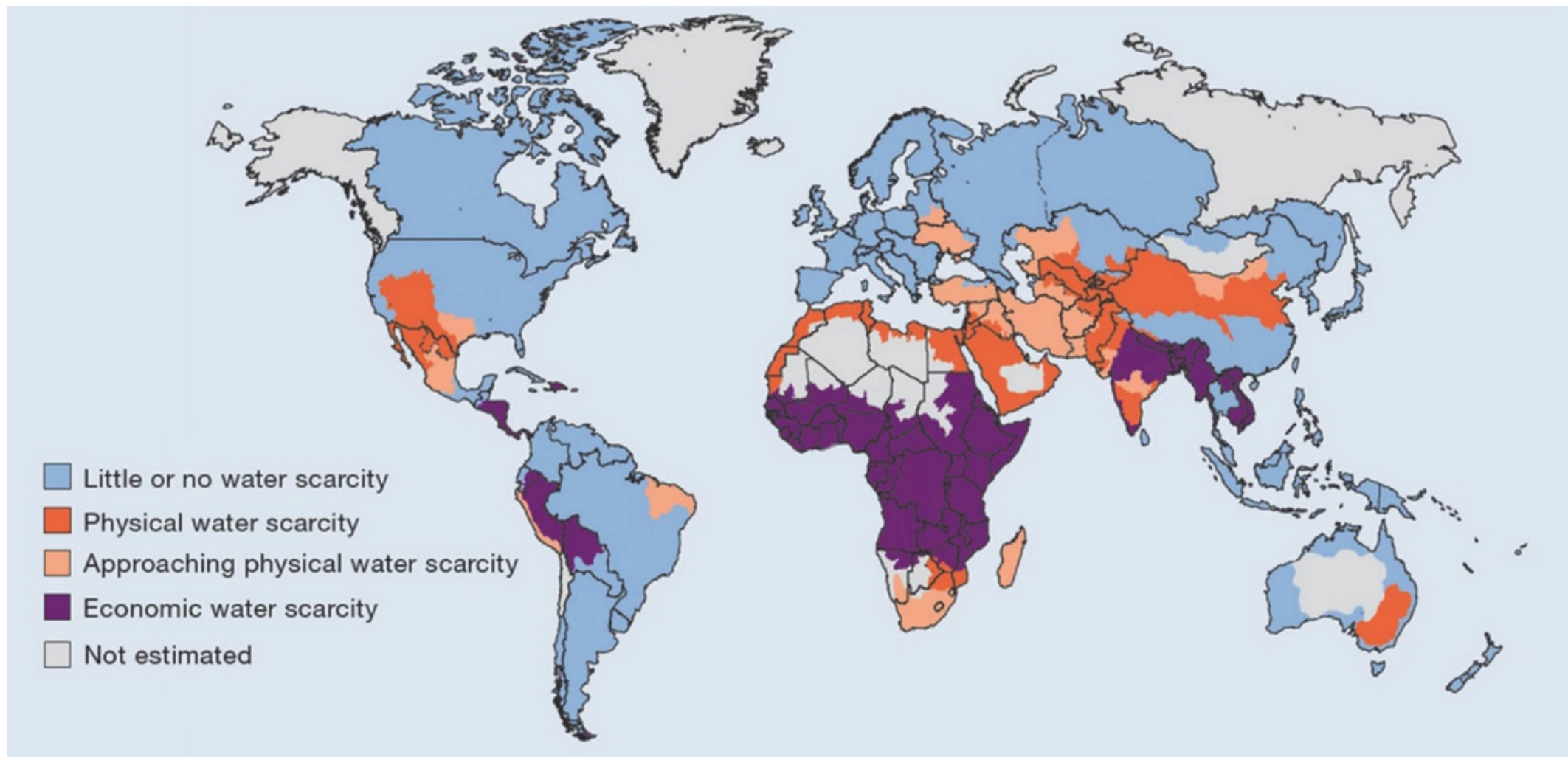
Κύρια αιτία για την λειψυδρία αποτελεί ο τριπλασιασμός της παγκόσμιας κατανάλωσης που έχει πραγματοποιηθεί τα τελευταία 70 χρόνια. Με τις πιέσεις να εντοπίζονται κυρίως λόγω της αύξησης του παγκόσμιου πληθυσμού, την αναβάθμιση του βιοτικού επιπέδου καθώς και στην αύξηση των καλλιεργήσιμων εκτάσεων (Postel S. L., 2000).



Εικόνα 1: Κατανομή του νερού της Γης (Gleick, 1993)

Με την Γη να χαρακτηρίζεται ως «Γαλάζιος Πλανήτης» θα περίμενε κανείς να υπάρχει αφθονία του συγκεκριμένου στοιχείου. Ωστόσο, ενώ το 71% της επιφάνειας του πλανήτη καλύπτεται από νερό, λιγότερο από το 3% μπορεί να χαρακτηριστεί ως «γλυκό». Επιπλέον ένα ποσοστό της τάξης του 70% από το «γλυκό» νερό, είναι «μη διαθέσιμο», όντας εγκλωβισμένο είτε στους παγετώνες, είτε στο υπέδαφος σε βάθη που δεν μπορεί να αξιοποιηθεί (Εικόνα 1) (Gleick, 1993).

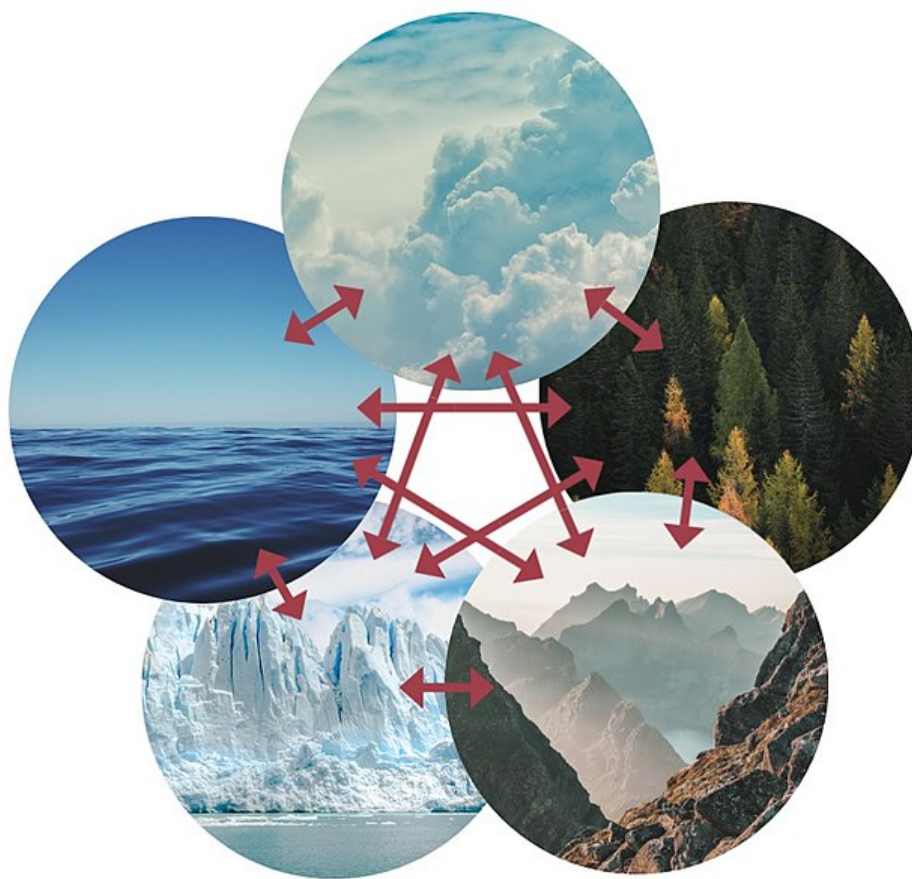
Ως εκ τούτου, κρίνεται αναγκαίο να γίνεται έλεγχος τόσο για την διατήρηση των αποθεμάτων όσο και για την δυνατότητα εκμετάλλευσης μεγαλύτερων αποθεμάτων νερού.



Εικόνα 2: Περιοχές που αντιμετωπίζουν φυσική ή οικονομική λειψυδρία (IWMI, 2017)

ΥΔΡΟΛΟΓΙΚΟΣ ΚΥΚΛΟΣ

Το κλίμα μιας περιοχής, με βάση την **Διακυβερνητική Επιτροπή για την Αλλαγή του Κλίματος** (*Intergovernmental Panel on Climate Change* - I.P.C.C.), ορίζεται η στατιστική περιγραφή με μέσο όρο και εύρος τιμών διάφορων σχετικών ποσοτήτων όπως η θερμοκρασία, η υγρασία, η ατμοσφαιρική πίεση, η ατμοσφαιρική κατακρήμνιση και η ένταση του ανέμου για μια συγκεκριμένη χρονική περίοδο (IPCC, 2013). Με βάση των Παγκόσμιο Μετεωρολογικό Οργανισμό (W.M.O.), η περίοδος των 30 ετών είναι το χαρακτηριστικό διάστημα για την στατιστική μελέτη αυτών των παραμέτρων, παρόλα αυτά διαφορετικά χρονικά διαστήματα μπορούν να χρησιμοποιηθούν (IPCC, 2012). Το κλιματικό σύστημα αποτελείται από πέντε μέρη, την ατμόσφαιρα, την υδρόσφαιρα, την κρυόσφαιρα, την λιθόσφαιρα και την βιόσφαιρα καθώς και από τις αλληλεπιδράσεις των συστημάτων μεταξύ τους καθώς και την αντίδραση τους σε εξωτερικές επιδράσεις (Εικόνα 3) (IPCC, 2001).



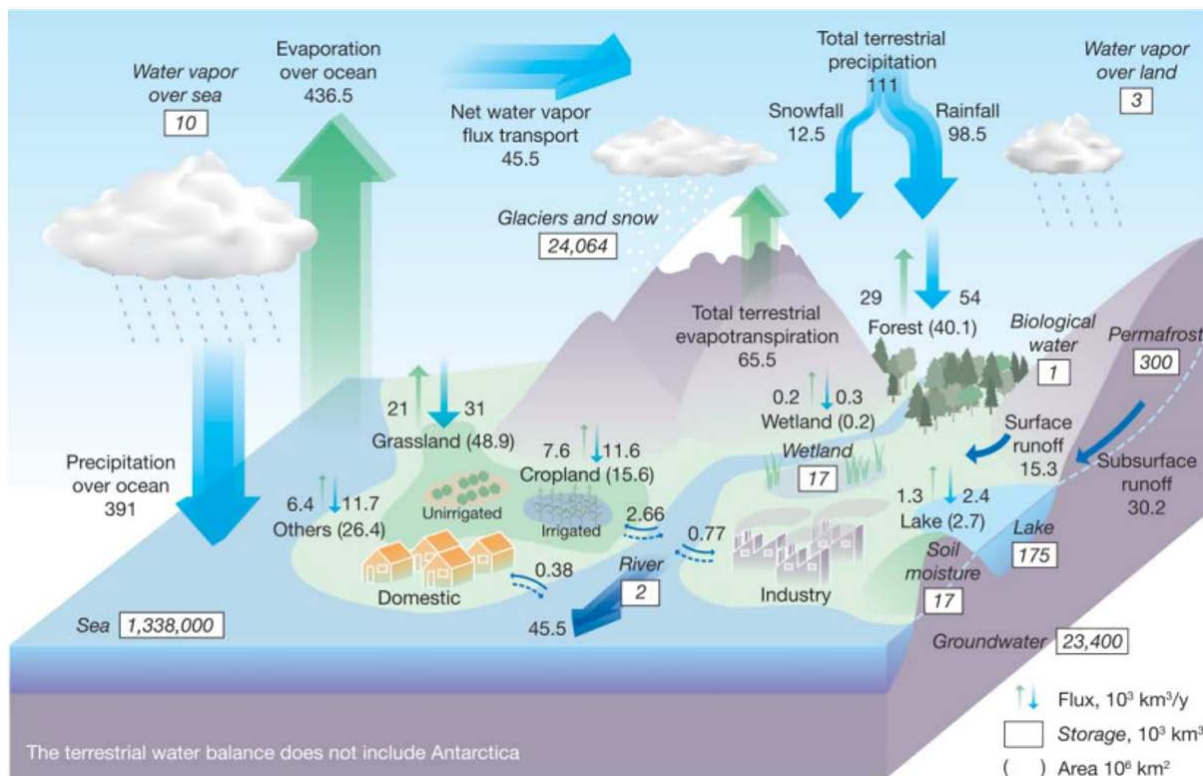
Εικόνα 3: Τα πέντε κομμάτια του κλιματικού συστήματος και οι αλληλεπιδράσεις τους¹

Βασικός μηχανισμός αλληλεπίδρασης αυτών των πέντε κομματιών είναι ο λεγόμενος «κύκλος του νερού». Με αυτόν τον όρο περιγράφεται το σύνολο των διαδικασιών μέσω των οποίων τα μόρια του νερού μεταφέρονται από το ένα μέρος του κλιματικού συστήματος σε ένα άλλο καθώς και οι μηχανισμοί μέσω των οποίων πραγματοποιείτε αυτή η μετακίνηση (Εικόνα 4).

¹ Η εικόνα λήφθηκε από τον διαδικτυακό ιστότοπο [Wikipedia](https://en.wikipedia.org/wiki/Climate_system)

ΥΠΟΓΕΙΑ ΥΔΑΤΑ

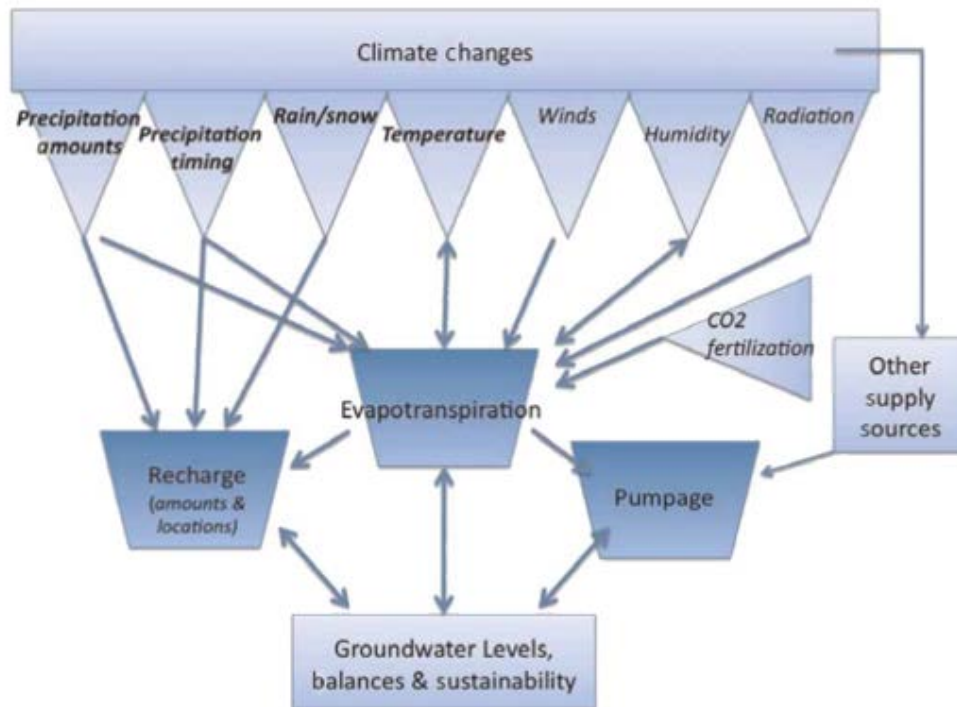
Ένα σημαντικό μέρος του κύκλου του νερού, αποτελεί τα υπόγεια ύδατα. Με τον συγκεκριμένο όρο περιγράφεται το νερό που βρίσκεται στην λιθόσφαιρα, είτε αυτό αφορά το νερό που βρίσκεται εντός της ζώνης του ριζικού συστήματος, είτε για την βαθύτερη ακόρεστη ζώνη είτε ελεύθερων ή περιορισμένων υδροφορέων (Green, et al., 2011). Παρόλο που τα υπόγεια νερά, αποτελούν το 1.69% του συνολικά εκτιμώμενου νερού που υπάρχει στην Γη, περιλαμβάνουν πάνω από το 30% του «γλυκού νερού» την στιγμή που το 68.7% βρίσκεται εγκλωβισμένο στους παγετώνες (Oki & Kanae, 2006). Εκτιμάται ότι πάνω από το 50% της παγκόσμιας κατανάλωσης νερού προέρχεται από τα υπόγεια νερά (United Nations, 2003) ενώ τα ποσοστά που αφορούν την άρδευση είναι ακόμα μεγαλύτερα (S. Siebert, et al., 2010).



Εικόνα 4: Παγκόσμιες υδρολογικές ροές και αποθήκες νερού. Στην εικόνα διακρίνονται τόσο οι φυσικές όσο και οι ανθρώπινες επιδράσεις. Ενώ με τα κάθετα βέλη, αναπαριστάτε η βροχοπτώση και η εξατμισοδιαπνοή (Oki & Kanae, 2006)

Τα υπόγεια νερά έχουν την δυνατότητα να προσαρμοστούν και να απορροφήσουν τις κλιματολογικές αλλαγές πολύ καλύτερα από ότι τα επιφανειακά ύδατα. Οι υδροφορείς έχουν την δυνατότητα να αποθηκεύσουν σημαντικές ποσότητες νερού, οι οποίες δεν μεταβάλλονται κατά την διάρκεια του έτους. Έχουν την δυνατότητα να αποθηκεύσουν επιπλέον νερό κατά την διάρκεια καταιγίδων, να μειώσουν τις απώλειες λόγω εξατμισοδιαπνοής και να προστατέψουν την ποιότητα του νερού. Παρόλα αυτά οι συνεχείς πιέσεις λόγω της κλιματικής αλλαγής καθώς και η ανθρώπινη υπερεκμετάλλευση μέσω της υπεράντλησης δημιουργεί αρνητικό αντίκτυπο (Clifton, et al., 2010).

Οι περιβαλλοντικοί παράμετροι που μπορούν να επηρεάσουν τις ροές προς και από τα υπόγεια ύδατα είναι οι ποσότητες, το είδος (βροχή, χαλάζι ή χιόνι), η διάρκεια και η συχνότητα των ατμοσφαιρικών κατακρημνίσεων, η θερμοκρασία, η ένταση του αέρα, η υγρασία και η ηλιακή ακτινοβολία (Εικόνα 5) (Earman & Dettinger, 2011).



Εικόνα 5: Σχηματική απεικόνιση των σχέσεων μεταξύ των κλιματικών παραμέτρων και των επιδράσεων στην ποσότητα των υπογείων υδάτων. (Earman & Dettinger, 2011)

Η οδηγία πλαίσιο 2000/60 ΕΚ και η ενσωμάτωση αυτής στην νομοθεσία με τον Ν. 3199/2006 προβλέπουν την παρακολούθηση των υπογείων υδάτων που χρησιμοποιούνται ή αναμένεται να χρησιμοποιηθούν για ύδρευση, με σκοπό τη διατήρηση της καλής ποσοτικής και ποιοτικής κατάστασης. Για να μπορεί να γίνει σωστή παρακολούθηση θα πρέπει να γίνει υπολογισμός της της ροής των υπογείων υδάτων με βάση την εξίσωση του υδραυλικού ισοζυγίου

$$\Delta S = I - O + P + EPT \pm Q \quad (1)$$

Όπου

ΔS είναι η μεταβολή στην ποσότητα του νερού που αποθηκεύεται στον υδροφόρο

I είναι οι εισροές στον υδροφόρο

O είναι οι εισροές στον υδροφόρο

EPT είναι η εξατμισοδιαπνοή που λαμβάνει χώρα στην περιοχή

Q είναι η άντληση/εμπλουτισμός που λαμβάνει χώρα στην περιοχή

Για τις εισροές και εκροές του υδροφόρου μπορεί να γίνει κάποια εκτίμηση, αλλά είναι αδύνατο να υπολογιστεί ακριβώς. Ενώ για την εξατμισοδιαπνοή, υπάρχουν αρχεία χρονοσειρών δεκαετιών στους μετεωρολογικούς σταθμούς. Παρόλο όμως που η μέτρηση και η βραχυπρόθεσμη πρόβλεψη της είναι εφικτή, είναι αδύνατη η άμεση συσχέτισή της με την στάθμη του υδροφόρου ορίζοντα καθώς είναι πολύ δύσκολο να προσδιοριστούν με ακρίβεια άλλες παράμετροι που είναι απαραίτητες ώστε να μπορεί να επιλυθεί η παραπάνω εξίσωση. Για το λόγο αυτό για την εκτίμηση της στάθμης των υπογείων υδάτων, χρησιμοποιούνται διάφορα μοντέλα προσομοίωσης της υπόγειας ροής.

MONTEΛΑ

Το μοντέλο είναι ένα σύγχρονο εργαλείο που απεικονίζει μια προσέγγιση ενός πραγματικού συστήματος και της προσομοίωσης της συμπεριφοράς του σε διάφορες φυσικές ή ανθρωπογενείς μεταβολές και πιέσεις. Γενικά, το μοντέλο είναι μια θεμελιώδης περιγραφή ή προσέγγιση που περιγράφει φυσικά συστήματα χρησιμοποιώντας μαθηματικές εξισώσεις. Δεν μπορεί να θεωρηθεί ως ακριβής αναπαράσταση του πραγματικού κόσμου επειδή δεν είναι εφικτή η μεταφορά της πολυπλοκότητας ενός φυσικού υδρογεωλογικού συστήματος σε ένα μοντέλο (Ghosh & Sharma, 2006).

Μέχρι τις αρχές τις νέας χιλιετίας, ο κυρίως τρόπος για την προσομοίωση των υπογείων υδάτων ήταν η χρήση των αριθμητικών μοντέλων. Τα αριθμητικά μοντέλα, είναι αλγόριθμοι οι οποίοι χρησιμοποιούν κάποια αριθμητική μέθοδο, όπως τα πεπερασμένα στοιχεία ή οι πεπερασμένες διαφορές, για την επίλυση των διαφορικών εξισώσεων που προκύπτουν από το ισοζύγιο μάζας της [εξίσωσης 1](#) (Anderson & Woessner, 1992). Για την ορθή χρήση των αριθμητικών μοντέλων απαιτείται η ακριβή γνώση των γεωλογικών και υδρογεωλογικών συνθηκών που επικρατούν σε ολόκληρη την έκταση. Η φυσική μεταβλητότητα, η μοναδικότητα των λεκανών απορροής, οι αστοχίες των γεωλογικών σχηματισμών (ρήγματα) σε συνάρτηση του κόστους απόκτησης αυτών των δεδομένων μπορεί να καταστήσει αδύνατη την χρήση αυτών των μοντέλων, ή πιο συχνά να δημιουργηθούν κάποια μοντέλα με ελλείψεις πληροφορίες που δεν μπορούν να περιγράψουν με ικανοποιητική ακρίβεια το σύστημα της υπόγειας ροής (Abrahart, Kneale, & See, 2004).

Τα τελευταία είκοσι χρόνια όμως, υπάρχει η εναλλακτική της χρήσης των μοντέλων που βασίζονται στην τεχνητή νοημοσύνη και στα τεχνητά νευρωνικά δίκτυα. Τα μοντέλα αυτά δεν προσπαθούν να επιλύσουν την [εξίσωση 1](#), αλλά χρησιμοποιώντας διάφορα δεδομένα, εκπαιδεύονται με στόχο να μπορούν να απεικονίσουν το πραγματικό σύστημα. Η επιλογή των παραμέτρων εισόδου όσο και της εξόδου του μοντέλου επηρεάζει και την ακρίβεια του μοντέλου. Για αυτό θα πρέπει να γίνεται πολύ προσεκτικά η επιλογή των παραμέτρων εισόδου καθώς το μοντέλο, είναι εφικτό να «βρει συσχέτιση» μεταξύ παραμέτρων που δεν έχουν κάποια φυσική σημασία ενώ παράλληλα καθώς ο όγκος των δεδομένων αυξάνεται, θα αυξηθεί σημαντικά και ο χρόνος εκπαίδευσης του τεχνητού νευρωνικού δικτύου (Abrahart, Kneale, & See, 2004).

Παρόλο που διάφοροι συγγραφείς περιγράφουν τα τεχνητά νευρωνικά δίκτυα, ως μαύρα κουτιά και τα απορρίπτουν ως κατώτερα, εάν χρησιμοποιηθούν οι κατάλληλοι παράμετροι και δοθούν επαρκή δεδομένα για την εκπαίδευσή τους, μπορούν να δώσουν ικανοποιητική προσέγγιση της υπόγειας ροής (Abrahart, Kneale, & See, 2004).

ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

Για αιώνες ο άνθρωπος προσπαθεί να αναλύσει την πηγή της υπεροχής του έναντι των υπόλοιπων οργανισμών στη Γη, πως δηλαδή ο ανθρώπινος εγκέφαλος, μπορεί να αντιληφθεί να κατανοήσει, να προβλέψει και να χειριστεί ένα κόσμο πολύ μεγαλύτερο και πιο περίπλοκο από τον δικό του. Το πεδίο της τεχνητής νοημοσύνης (*Artificial Intelligence – AI*) δεν περιλαμβάνει μόνο την κατανόηση αυτής της διαδικασίας, αλλά την δημιουργία νοήμονων οντοτήτων (μηχανών) που θα μπορούν να δρουν αποτελεσματικά και με ασφάλεια σε μια ευρεία ποικιλία νέων καταστάσεων (Russel & Norvig, 2020).

Από το βρετανικό bombe που φτιάχτηκε για να σπάσει την κρυπτογράφηση του ENIGMA, μέχρι τους συγχρόνους ηλεκτρονικούς υπολογιστές, ένα πράγμα έχει παραμείνει σταθερό, η ικανότητα των μηχανών να επιλύουν πράξεις σε πολύ μεγαλύτερη ταχύτητα από τον άνθρωπο. Στην πράξη οι εφαρμογές των υπολογιστών μέχρι πριν κάποια χρόνια, ήταν η εκτέλεση κάποιων αριθμητικών πράξεων σε πολύ μεγάλες ταχύτητες. Όμως αυτό δεν είναι νοημοσύνη, ο υπολογιστής απλά ακολουθεί κάποιες βασικές οδηγίες, απλά το κάνει με πολύ μεγάλη ταχύτητα. Η υπολογιστική ισχύς που είχαν, μέχρι πριν λίγα χρόνια, δεν μπορούσε να ανταγωνιστεί την αντίληψη που διαθέτουν τα έμβια όντα. Για παράδειγμα οι νυχτερίδες έχουν την δυνατότητα μέσω της ακοής να μπορούν να εντοπίζουν τους στόχους τους, να υπολογίζουν την απόσταση μεταξύ τους ενώ πρόσφατες έρευνες έχουν δείξει ότι έχουν την δυνατότητα να προβλέπουν τις μελλοντικές θέσεις των στόχων τους (Salles, Diebold, & Moss, 2020).

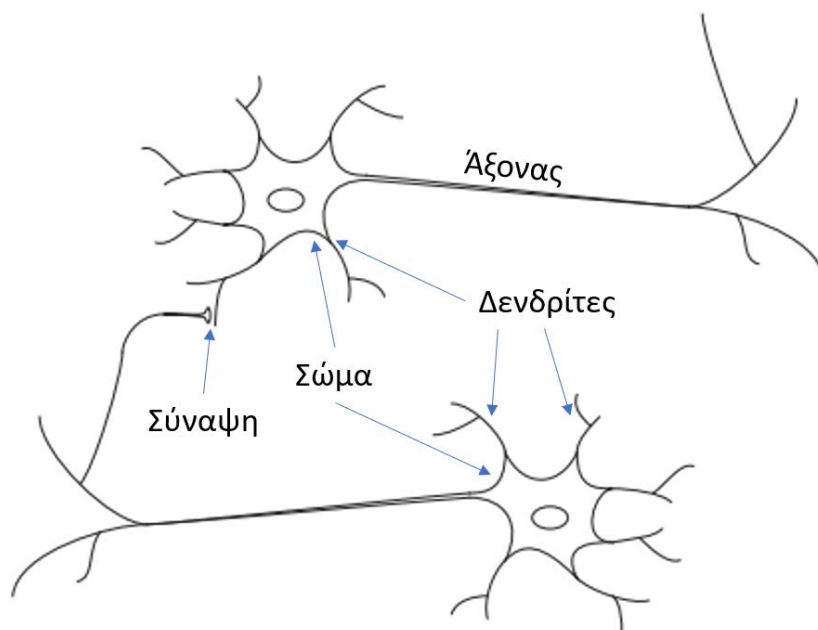
Μπορούν όμως οι μηχανές στην πραγματικότητα να σκεφτούν; Με βάση τον Alan Turing για να θεωρήσουμε ότι μια μηχανή σκέφτεται θα πρέπει να περάσει την δοκιμασία του παιχνιδιού της μίμησης, κατά την οποία μια μηχανή υποκρίνεται τον άνθρωπο και ένας ουδέτερος παρατηρητής θα πρέπει να μαντέψει εάν πρόκειται για άνθρωπο ή για μηχανή (Turing, 1950). Για να μπορέσει λοιπόν ένας υπολογιστής να περάσει αυτή την δοκιμασία θα πρέπει να έχει τις εξής δυνατότητες, δυνατότητα επικοινωνίας με χρήση της ανθρώπινης γλώσσας, την ικανότητα να αποθηκεύσει την γνώση που έχει ή που λαμβάνει από το περιβάλλον, την δυνατότητα να απαντά ερωτήσεις και να εξάγει νέα συμπεράσματα καθώς και την μηχανική εκμάθηση (*machine learning*) με βάση την οποία θα μπορεί να προσαρμόζεται σε νέες συνθήκες και να εντοπίζει και να αναλύει μοτίβα (Russel & Norvig, 2020). Με τον όρο μηχανική εκμάθηση περιγράφεται λοιπόν ένα υποσύνολο της τεχνητής νοημοσύνης, η οποία πραγματεύεται την ικανότητα των μηχανών να βελτιώνουν τις επιδόσεις τους, βάση της εμπειρίας τους (Russel & Norvig, 2020).

Για να δημιουργηθούν όμως μηχανές που να μπορούν να σκέφτονται σαν τον άνθρωπο, θα πρέπει να αναλυθεί ο τρόπος με τον οποίο λειτουργεί ο ανθρώπινος εγκέφαλος. Και στην συνέχεια να δημιουργηθούν οι κατάλληλοι αλγόριθμοι¹, οι οποίοι θα προσπαθούν να προσομοιώσουν την λειτουργία του (Russel & Norvig, 2020).

¹ Ο όρος αλγόριθμος, προέρχεται από τον μαθηματικό του 9^{ου} αιώνα Muhammad ibn Musa al-Khwarizmi, του οποίου τα κείμενα ήταν αυτά που εισήγαγαν τους Αραβικούς αριθμούς και την άλγεβρα στην Ευρώπη. Όμως η ιστορία των αλγόριθμων είναι πιο παλιά, με τον αλγόριθμο του Ευκλείδη για τον υπολογισμό του μεγαλύτερου κοινού διαιρέτη να αποτελεί την πρώτη καταγραφή (Russel & Norvig, 2020).

Ο ΑΝΘΡΩΠΙΝΟΣ ΕΓΚΕΦΑΛΟΣ

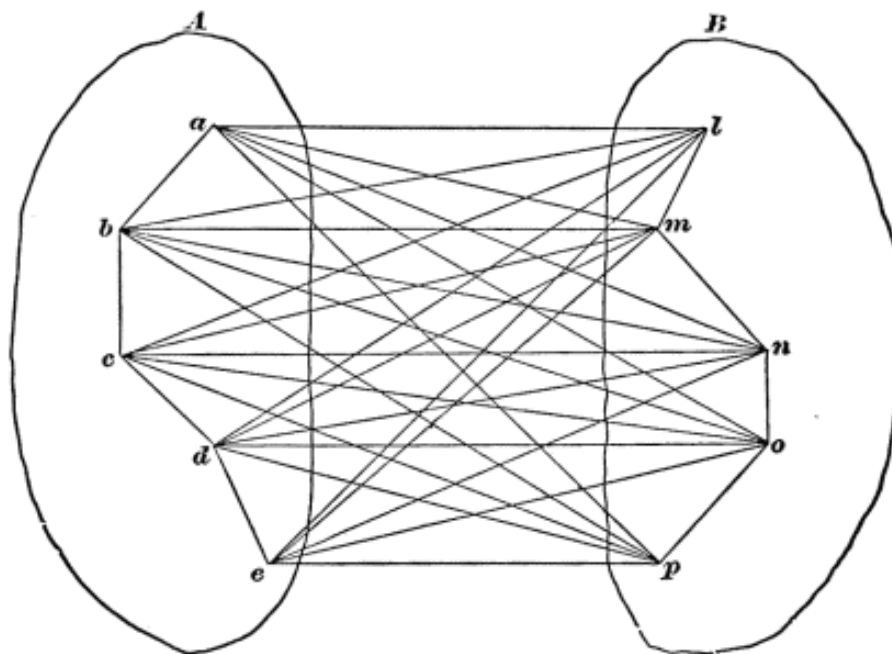
Ο ανθρώπινος εγκέφαλος αποτελείται από σχεδόν 10^{11} στοιχεία καθένα από τα οποία έχει περίπου 10^4 συνδέσεις, τους λεγόμενους νευρώνες (Hagan, et al., 2014). Οι νευρώνες ή νευρωνικά κύτταρα, αποτελούνται από τρία κομμάτια, το σώμα, τον νευράξονα και τους δενδρίτες (Εικόνα 6). Το μήκος του νευράξονα είναι πολύ μεγαλύτερο από ότι η εικόνα αποδίδει, τυπικά ένας νευράξονας έχει μήκος 1cm, μήκος που ισούται με περίπου 100 φορές την διάμετρο του σώματος του νευρώνα, ενώ υπάρχουν περιπτώσεις που μπορεί να φτάσει έως και ένα μέτρο. Ο κάθε νευρώνας έχει πολλούς δενδρίτες με πολλές διακλαδώσεις οι οποίοι μπορούν να δημιουργήσουν έως και 100,000 ενώσεις με τις καταλήξεις που υπάρχουν στους άξονες άλλων νευρώνων, δημιουργώντας με αυτόν τον τρόπο ένα δίκτυο με το οποίο ο νευρώνας προσλαμβάνει σήματα από άλλα κύτταρα. Οι ενώσεις αυτές καλούνται συνάψεις και είναι ο τρόπος με τον οποίο συνδέονται, διακλαδίζονται και συνεργάζονται μεταξύ τους τα νευρικά κύτταρα κατά ομάδες, μέσω πολύπλοκων ηλεκτροχημικών αντιδράσεων. Τα σήματα, βραχυπρόθεσμα, ελέγχουν την δραστηριότητα του εγκεφάλου επιτρέποντας την διάδοση ερεθισμάτων (σημάτων) από τον εγκέφαλο προς τα όργανα και το αντίστροφο και μακροπρόθεσμα καθορίζουν την συνδεσιμότητα των νευρώνων (Russel & Norvig, 2020). Ο εγκέφαλος είναι ένα πολύ περίπλοκο, μη γραμμικό, παράλληλο σύστημα επεξεργασίας πληροφοριών. Οι αισθήσεις στην πραγματικότητα, είναι διεργασίες επεξεργασίας πληροφοριών, είτε αυτό αφορά την αναγνώριση ενός φιλικού προσώπου μέσω της όρασης, ή ενός φαγητού μέσω της μυρωδιάς ή την ύπαρξη κινδύνου μέσω της ακοής. Οι διεργασίες αυτές εκτελούνται σε απειροελάχιστο χρόνο από τα έμβια όντα, της τάξης των δέκατων του δευτερολέπτου (Haykin, 2009).



Εικόνα 6: Σχηματική απεικόνιση δύο βιολογικών νευρώνων (Hagan, et al., 2014)

Ένα μέρος αυτού του νευρωνικού δικτύου δημιουργείτε κατά την γέννηση ενός ατόμου, ενώ άλλα μέρη δημιουργούνται και αναπτύσσονται στην υπόλοιπη ζωή του, καθώς μέσω της μάθησης νέες συνδέσεις δημιουργούνται ενώ άλλες απενεργοποιούνται. Οι μεταγενέστερες αλλαγές τείνουν να συνίστανται κυρίως στην ενίσχυση ή την αποδυνάμωση των συνοπτικών συνδέσεων. Η δημιουργία νέων αναμνήσεων είναι ένα παράδειγμα τροποποίησης των συνοπτικών βαρών (Hagan, et al., 2014).

Καθοριστικό ρόλο στην κατανόηση της λειτουργίας των νευρώνων καθώς και στην μετέπειτα λειτουργία των τεχνητών νευρωνικών δικτύων, έπαιξε η ερευνα του ψυχολόγου *William James*, ο οποίος παρουσίασε ένα λεπτομερές μηχανιστικό μοντέλο συσχέτισης το οποίο είναι σχεδόν πανομοιότυπο με τα νευρωνικά δίκτυα. Ο *James* περιέγραψε την πρώτη αρχή της λειτουργίας του εγκεφάλου ως, «όταν δύο στοιχειώδεις εγκεφαλικές διεργασίες ήταν ενεργές μαζί ή σε άμεση διαδοχή, αν ενεργοποιηθεί εκ νέου η μία από αυτές, τείνει να διαδίδει τον ενθουσιασμό της στην άλλη» (*James, 1892*), αν αντικατασταθεί η «εγκεφαλική διεργασία» με τον νευρώνα, προκύπτει ένας κανόνας μάθησης, πανομοιότυπος με την θεωρία του *Donald Hebb* για τις συνάψεις, η οποία αναφέρει ότι: «όταν ένας άξονας του κυττάρου *A* είναι αρκετά κοντά για να διεγείρει ένα κύτταρο *B* και επανειλημμένα ή επίμονα συμμετέχει στην διέγερση του, τότε πραγματοποιείται μια μεταβολική αλλαγή ή κάποια διαδικασία ανάπτυξης ένα ή και στα δύο κύτταρα έτσι ώστε να αυξηθεί η αποδοτικότητα του *A* ως ένα κύτταρο που ενεργοποιεί το *B*» (*Hebb, 1949*). Η δεύτερη αρχή που όρισε ο *James* ήταν ότι «το ποσό της δραστηριότητας σε οποιοδήποτε σημείο του φλοιού του εγκεφάλου, είναι το άθροισμα των εκφορτίσεων των υπολοίπων σημείων σε αυτό. Οι τάσεις αυτές είναι ανάλογες με τον αριθμό των φορών που αντίδραση δύο σημείων συνοδεύει εκείνο το σημείο, στις εντάσεις των ενθουσιασμών αυτών καθώς και στην απουσία κάποιου αντίπαλου σημείου, που δεν συνδέεται με το αρχικό, το οποίο θα μπορούσε να απορροφήσει κάποιες από τις εκφορτίσεις» (Εικόνα 7) (*James, 1892*) Εάν σε αυτήν την αρχή, αντικατασταθεί το «σημείο του φλοιού του εγκεφάλου» με το «νευρώνας» ή το «στοιχείο», προκύπτει ένα μοντέλο το οποίο καθορίζει την δράση του νευρώνα ως το άθροισμα των εισόδων, βαθμισμένο με βάση τον βαθμό της σύνδεσης που προκύπτει από τις προηγούμενες συσχετίσεις, τον τρέχων ενθουσιασμό των υπόλοιπων νευρώνων καθώς και με έναν ανασταλτικό μηχανισμό. Αυτή η δομή είναι παραπλήσια ενός αλγόριθμου που χρησιμοποιεί την τροποποίηση συνάψεων *Hebbian* και την γραμμική άθροιση των συνοπτικών εισόδων (*Anderson & Rosenfeld, 1988*).

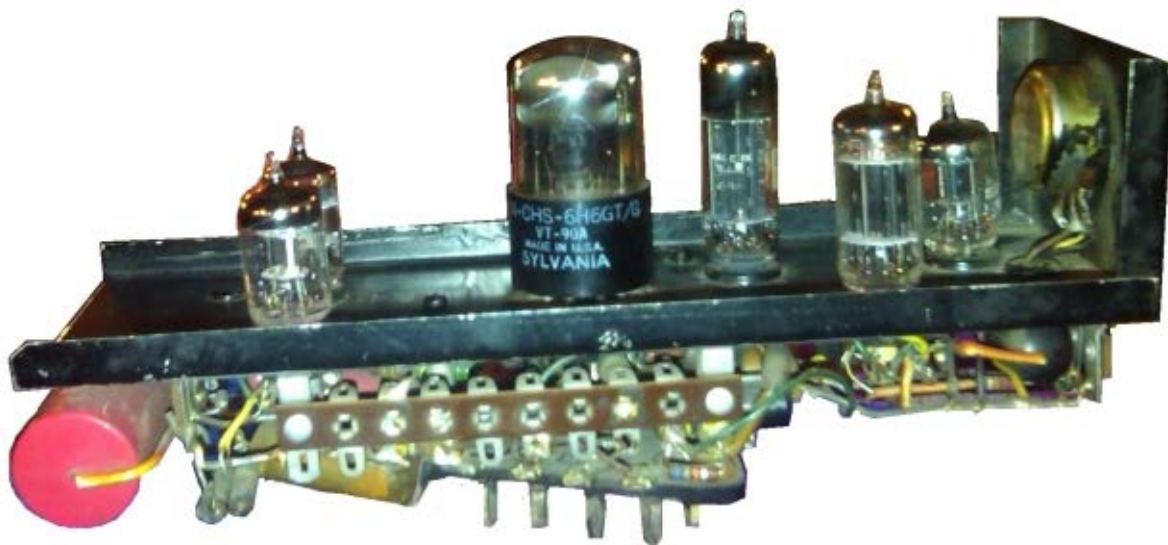


Εικόνα 7: Οι γραμμές συμβολίζουν το άθροισμα των εκφορτίσεων σε καθένα από τα συστατικά του σημείου B και την επακόλουθη ισχύ του συνδυασμού επιρροών με τις οποίες αφυπνίζεται το σύνολο του B. Η εικόνα αυτή είναι πανομοιότυπη με ένα τυπικό τεχνητό νευρωνικό δίκτυο (*James, 1892*).

ΤΕΧΝΗΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Η ΓΕΝΕΣΗ

Μια πρωτοποριακή για την εποχή της εργασία, η οποία πλέον αναγνωρίζεται ως ΑΙ και έθεσε τις βάσεις για τα τεχνητά νευρωνικά δίκτυα (Τ.Ν.Δ.), ήταν αυτή του νευροψυχολόγου *Warren McCulloch* και μιας μαθηματικής διάνοιας όπως ήταν ο *Walter Pitts* το 1943. Στην εργασία τους παρουσίασαν το πρώτο μαθηματικό μοντέλο ενός Τ.Ν.Δ., στο οποίο κάθε νευρώνας θα μπορούσε να χαρακτηριστεί ως «ανοιχτός» ή «κλειστός» με την μετατροπή σε «ανοιχτό» να πραγματοποιείται ως αντίδραση στην ενεργοποίηση ενός ικανού αριθμού γειτονικών νευρώνων. Έδειξαν, για παράδειγμα, ότι οποιαδήποτε υπολογιστική λειτουργία θα μπορούσε να υπολογιστεί από κάποιο δίκτυο συνδεδεμένων νευρώνων και ότι όλα τα λογικά συνδετικά (*AND*, *OR*, *NOT* κ.λπ.) θα μπορούσαν να εφαρμοστούν από απλές δομές δικτύου. Ο απλός τυποποιημένος νευρώνας που χρησιμοποίησαν είναι το πρότυπο αναφοράς στον τομέα των νευρωνικών δικτύων και συχνά ονομάζεται νευρώνας *McCulloch-Pitts* ή λογική μονάδα κατωφλίου (*Threshold logic unit – TLU*) ([Gurney, 1997](#)). Οι *McCulloch & Pitts* πρότειναν επίσης ότι κατάλληλα καθορισμένα δίκτυα θα μπορούσαν να εκπαιδευτούν ([McCulloch & Pitts, 1943](#)). Η εργασία αυτή ήταν συνέχεια των εργασιών του *Pitt* πάνω στα Τ.Ν.Δ. και στην προσπάθεια δημιουργίας μιας μηχανής *Tuning* ([Pitts, 1942a,b, 1943a](#)) καθώς και στην θεωρία της μάθησης ([Pitts, 1943b,c](#)).

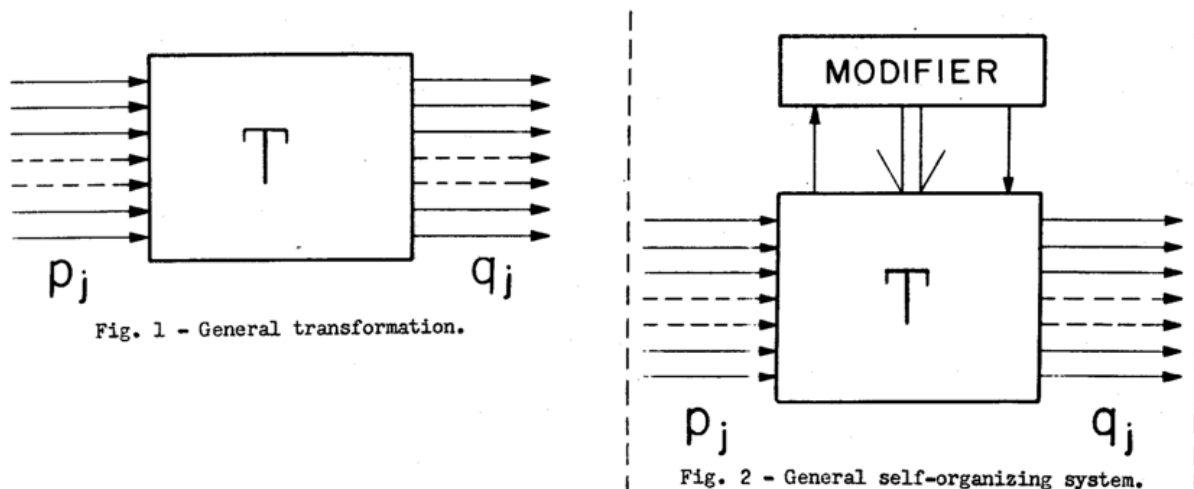


Εικόνα 8: Ένας από τους νευρώνες του *Snarc*¹

¹ Η φωτογραφία αυτή λήφθηκε από τον *G.Loan* σε μια επίσκεψη του στον *Marvin Minsky*. Κατά τον *Loan*, ο *SNARK* αποσυναρμολογήθηκε από κάποιους μεταγενέστερους φοιτητές και αυτός ο νευρώνας ήταν ο μόνος που έμεινε στην κατοχή του *M.Minsky*. Η φωτογραφία ανακτήθηκε από τον ιστότοπο [Cyberneticzoo](#).

ΟΙ ΠΡΩΤΕΣ ΠΡΟΣΠΑΘΕΙΕΣ

Ο πρώτη μηχανή που εξομοίωνε τα Τ.Ν.Δ. ήταν ο SNARC (*Stochastic Neural Analog Reinforcement Computer*) (Εικόνα 8), ο οποίος δημιουργήθηκε από δύο φοιτητές του Harvard, τους Marvin Minsky και Dean Edmonds, το 1950. Η μηχανή αποτελούνταν από 3000 ηλεκτρονικές λυχνίες και ένα μηχανισμό αυτόματου πιλότου από ένα βομβαρδιστικό B-24 που δημιουργούσαν ένα δίκτυο 40 νευρώνων και συνάψεις που μπορούσαν να αλλάξουν τα βάρη τους, με βάση την λογική της μάθησης *Hebbian* (αλλαγή των βαρών με βάση την δυνατότητα ολοκλήρωσης ενός συγκεκριμένου έργου) και μπορούσε να ολοκληρώσει με επιτυχία την εύρεση της εξόδου ενός λαβύρινθου¹. Ιστορική παραμένει και η απάντηση του Minsky στην ερώτηση της επιτροπής του, εάν αυτό είναι Μαθηματικά, «εάν δεν είναι τώρα, θα είναι κάποια στιγμή» (Russel & Norvig, 2020).



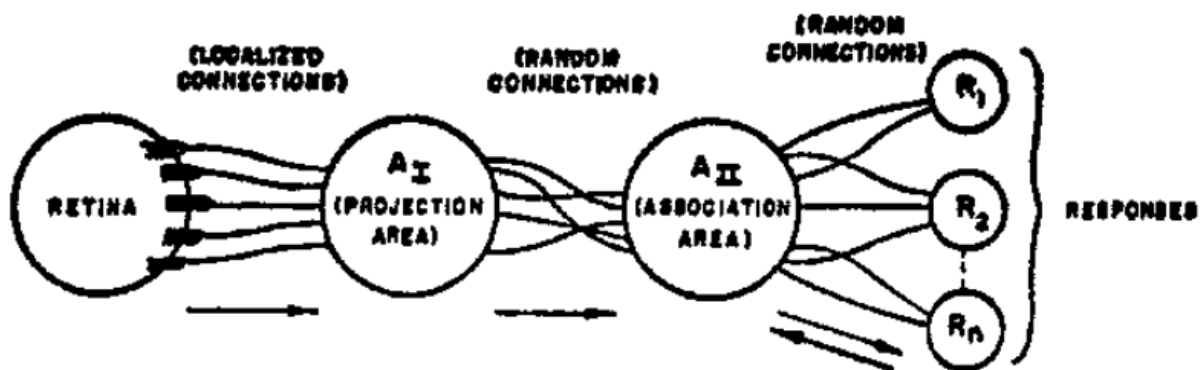
Εικόνα 9: Σχεδιάγραμμα της 1^{ης} ψηφιοποίησης ενός αυτοοριζόμενου δικτύου 8 στοιχείων (Farley & Clark, 1954).

Το πρώτο ψηφιακό Τ.Ν.Δ. ήρθε με τις προσπάθειες των Farley & Clark, οι οποίοι σε ένα πειραματικό υπολογιστή MTC (*Memory Test Computer*) με μνήμη 16bit, έτρεξαν ένα αυτοοριζόμενο δίκτυο 8 στοιχείων (Εικόνα 9) (Farley & Clark, 1954). Ενώ δύο χρόνια αργότερα υπήρξε η πρώτη ψηφιοποίηση δικτύου βασισμένο στην θεωρία του Hebb, από την ερευνητική ομάδα της IBM. Αυτή η προσπάθεια αντιπροσωπεύει την πρώτη δοκιμή για προσομοίωση σε υπολογιστή μιας λεπτομερής, καλά διαμορφωμένης θεωρίας νευρωνικών δικτύων. Παρόλο που η πρώτη προσπάθεια απέτυχε, στην δεύτερη δοκιμή οι ερευνητές επανήλθαν με δύο σημαντικές τροποποιήσεις, η πρώτη αφορούσε το εύρος τιμών των συνάψεων, οι οποίες στην πρώτη δοκιμή μπορούσαν να πάρουν τις τιμές από 0 έως 1, ενώ στην δεύτερη από -1 έως 1 με αποτέλεσμα να υπάρχει καλύτερη σύγκλιση. Και η δεύτερη, ορίσαν ότι το άθροισμα των συναπτικών βαρών παραμένει σταθερό, σε αντίθεση με την θεωρία του Hebb όπου τα βάρη των συνάψεων αυξάνονται με το χρόνο χωρίς όριο (Rochester, et al., 1956). Έτσι στην δεύτερη δοκιμή, με βάση την κανονιστική νόρμα που έθεσαν, ορισμένα συνοπτικά βάρη αυξάνονταν εις βάρος άλλων. Αυτή ήταν μια σημαντική βελτίωση του αλγόριθμου εκμάθησης του Hebb και διάφορες παραλλαγές της χρησιμοποιούνται ακόμα και μέχρι σήμερα (Russel & Norvig, 2020).

¹ αν και έχει επικρατήσει ότι ήταν η μοντελοποίηση της αναζήτησης ενός αρουραίου για φαγητό σε έναν λαβύρινθο, λόγω του προσωνύμιου που είχαν δώσει οι ερευνητές στα φώτα των λυχνιών που ανάβαν σε σειρά (Anderson & Rosenfeld, 1988).

ΤΟ PERCEPTRON

Το *perceptron* (Εικόνα 10) δημιούργησε μια αίσθηση όταν περιγράφηκε για πρώτη φορά. Ήταν το πρώτο καθορισμένο με ακρίβεια, προσανατολισμένο υπολογιστικό Τ.Ν.Δ., και είχε σημαντικό αντίκτυπο ταυτόχρονα σε πολλούς τομείς της επιστήμης. Ήταν μια μηχανή εκμάθησης που ήταν δυνητικά ικανή για περίπλοκη προσαρμοστική συμπεριφορά. Η βασική διαφοροποίηση του μοντέλου ήταν ότι σχεδιάστηκε με όρους της θεωρίας πιθανοτήτων αντί της συμβολικής λογικής που χρησιμοποιούνταν μέχρι τότε. Η βασική αρχιτεκτονική δομή παραπέμπει σε ένα σύγχρονο Τ.Ν.Δ. με ένα «επίπεδο εισόδου» → «περιοχή προβολής» (*input layer / Projection Area*), ένα «κρυφό επίπεδο» → «στρώμα αλληλεπίδρασης» (*hidden layer / Association area*) και ένα «επίπεδο εξόδου» → «μονάδες απόκρισης» (*output layer / response units*), ενώ σχεδίασε και μια δομή με δύο κρυφά επίπεδα (Rosenblatt, 1958). Το *perceptron* επέδειξε ανεκτίμητη ικανότητα γενίκευσης. Δηλαδή, θα μπορούσε να ανταποκριθεί κατάλληλα σε μοτίβα που δεν είχε δει ποτέ, επειδή ήταν παρόμοια με μοτίβα που είχε δει (Anderson & Rosenfeld, 1988).



Εικόνα 10: Οργάνωση ενός Perceptron (Rosenblatt, 1958)

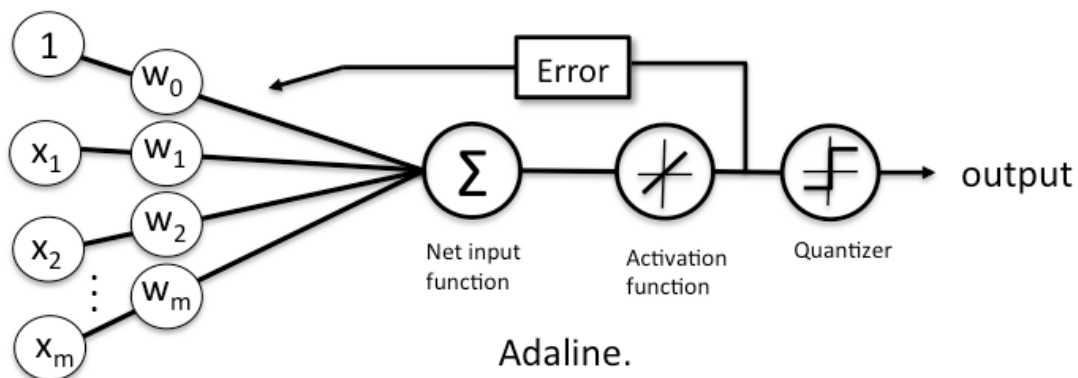
ΤΟ ADALINE, Ο ΚΑΝΟΝΑΣ WIDROW-HOFF ΚΑΙ Ο ΑΛΓΟΡΙΘΜΟΣ LMS

Παρόλο που τα perceptron είχαν δυνατότητα να επιλύουν προβλήματα κατηγοριοποίησης με επιτυχία, είχαν ένα βασικό μειονέκτημα, ότι οι περισσότερες μέθοδοι μάθησης που χρησιμοποιούσαν χρειαζόντουσαν παρά πολύ χρόνο για να επιλύσουν ακόμα και απλά προβλήματα, ενώ ακόμα και αν έβρισκαν κάποια λύση, δεν υπήρχε κάποια ένδειξη για το αν αυτή ήταν η ιδανική. Την απάντηση σε αυτό το πρόβλημα την έδωσαν οι *Bernard Widrow & Marcian E. Hoff* με το ADALINE (*ADAPTIVE LINEAR*)¹ (ANDERSON & ROSENFELD, 1988).

Το ADALINE εμπεριείχε τρεις σημαντικές καινοτομίες, η πρώτη αφορά την εισαγωγή της έννοιας της επιβλεπόμενης μάθησης (*supervised learning*). Με αυτόν τον όρο περιγράφεται η μηχανική εκμάθηση μιας συνάρτησης που χαρτογραφεί μια είσοδο σε μια συγκεκριμένη έξοδο με βάση παραδείγματα ζευγών εισόδου-εξόδου (Russel & Norvig, 2020). Στην συνέχεια ο αλγόριθμος έχει την δυνατότητα να πραγματοποιεί προβλέψεις για τις εισόδους στις οποίες δεν είναι γνωστή η τιμή της εξόδου (Mohri, Rostamizadeh, & Talwalkar, 2018). Η δεύτερη

¹ Αν και επικράτησε ως επεξήγηση αρχικά το **ADAPTIVE LINEAR NEURON** και μετά την πτώση δημοτικότητας των Τ.Ν.Δ. το **ADaptive LINear Element** (Anderson & Rosenfeld, 1988; Krose & van de Smagt, 1996)

αφορά στην εισαγωγή του κανόνα *Widrow-Hoff* γνωστός και ως κανόνας *Delta* ή κανόνας *Adaline* (Kim, 2017). Οι ερευνητές χρησιμοποιώντας τον αλγόριθμο της απότομης καθόδου (*gradient descent*) δημιούργησαν έναν αλγόριθμο για την ελάττωση του τετραγώνου του σφάλματος, γνωστός και ως αλγόριθμος *LMS* (*Least Mean Square*). Η χρήση του τετραγώνου αντί της τιμής του σφάλματος, γίνεται για τον απλούστατο λόγο ότι η τιμή που δίνει το σύστημα (y_p) μπορεί να είναι είτε μεγαλύτερη είτε μικρότερη από την πραγματική (d_p), ως εκ τούτου το σφάλμα (ε_p) θα μπορεί να λαμβάνει είτε θετικές τιμές είτε αρνητικές. Επομένως, αν υπολογιστεί το συνολικό σφάλμα, αθροίζοντας απλά τα επιμέρους σφάλματα, ελλοχεύει ο κίνδυνος να υπάρξει συμψηφισμός των αρνητικών τιμών με τις θετικές, δίνοντας παραπλανητικά αποτελέσματα. Η τρίτη καινοτομία αφορά την χρήση γραμμικής συνάρτησης ενεργοποίησης σε αντίθεση με τις συναρτήσεις Heaviside (*unit step function* ή *Heaviside step function*) που χρησιμοποιούντουσαν μέχρι τότε (Hagan, et al., 2014). Στο ADALINE οι μεταβλητές εξόδου θα έπρεπε να πάρουν είτε την τιμή 1 είτε την τιμή -1, το σύστημα κατηγοριοποιούσε τα αρνητικά ως -1 και τα θετικά ως 1. Όμως ακόμα και αν κάποιο ήταν θετικό, στην περίπτωση που δεν ήταν 1, υπήρχε ένα σφάλμα, το σύστημα υπολόγιζε αυτά τα σφάλματα και προσπαθούσε να τα ελαχιστοποιήσει με χρήση του αλγόριθμου LMS, με αυτόν τον τρόπο ακόμα και αν ήταν σωστή η κατηγοριοποίηση, σε περίπτωση που δεν υπήρχε ταύτιση 100%, μπορούσε η σύναψη να «βελτιωθεί». Σε αντίθεση με τις μεθόδους που χρησιμοποιούντουσαν πριν το ADALINE, όπου τα συνοπτικά βάρη αλλάζαν μόνο εάν το αποτέλεσμα ήταν λάθος (Εικόνα 11) (Widrow & Hoff, 1960).



Εικόνα 11: Ένας προσαρμοστικός νευρώνας (Widrow & Hoff, 1960)

Ο αλγόριθμος LMS μπορεί να συνοψιστεί στις εξισώσεις

$$\Delta_p w_j = \mu \cdot \varepsilon^p \cdot x_j \quad (2)$$

$$\varepsilon^p = d^p - y^p \quad (3)$$

Όπου

ε^p το σφάλμα που προκύπτει από το πρότυπο p

w_j το βάρος της σύναψης από τη μεταβλητή εισόδου j ($j \in [0, m] | j \in \mathbb{N}$), του προτύπου p

x_j η τιμή της μεταβλητής εισόδου j , του προτύπου p

d_p είναι η επιθυμητή τιμή για το πρότυπο p

y_p είναι η τιμή εξόδου που προκύπτει το πρότυπο p

μ είναι μια σταθερά που ονομάζεται ρυθμός μάθησης, προκύπτει από την μέθοδο της ελάττωση της παραγώγου και ανήκει στο \mathbb{R}_+ . Θα πρέπει να έχει αρκετά μικρή τιμή ώστε $\delta y \approx \Delta y$ (Gurney, 1997), αρχικά είχε προταθεί το $\mu = 1/(m+1)$, όπου m το σύνολο των μεταβλητών εισόδου, πρακτικά όμως η σύγκλιση με αυτή την τιμή είναι πολύ δύσκολη και γενικά χρησιμοποιείτε μια μικρότερη τιμή (Picton, 1994).

Στο ADALINE καθώς και σε κάθε νευρώνα, όταν εξετάζονται συστήματα με περισσότερους νευρώνες, υπολογίζεται μια επιπλέον τιμή ίση με την μονάδα, η x_0 , έτσι ώστε να εξασφαλιστεί ότι η είσοδος στον νευρώνα θα είναι μη μηδενική, ακόμα και αν όλες οι εισοδοί είναι μηδενικές. Για να έχει την ίδια διάσταση το διάνυσμα για τις τιμές εισόδου με αυτό του διανύσματος των βαρών προστίθεται και η τιμή του w_0 , η οποία υπολογίζεται κανονικά όπως τα υπόλοιπα βάρη (Russel & Norvig, 2020).

Οι παραπάνω εξισώσεις μπορούν να γραφούν και με διανυσματική μορφή:

$$\mathbf{W}_{p+1} = \mathbf{W}_p + \mu \cdot \varepsilon_p \cdot \mathbf{X}_p \quad (4)$$

$$\varepsilon^p = d^p - y^p \quad (5)$$

$$y_p = \mathbf{W}_p^T \cdot \mathbf{X}_p \quad (6)$$

Όπου

\mathbf{W}_p το διάνυσμα για τα βάρη των συνάψεων, για το πρότυπο p

\mathbf{X}_p το διάνυσμα για τις τιμές του προτύπου p

Στο διάνυσμα \mathbf{W} συμπεριλαμβάνεται η τιμή του w_0 , ενώ σε αυτό του \mathbf{X} , συμπεριλαμβάνεται το x_0 .

Η παραπάνω μορφή όμως ισχύει αποκλειστικά για δίκτυα στα οποία η συνάρτηση ενεργοποίησης του νευρώνα, είναι γραμμική, για την περιγραφή δικτύων με ένα επίπεδο με μη γραμμικές συναρτήσεις ενεργοποίησης, ο αλγόριθμος LMS παίρνει την εξής μορφή:

$$\Delta_p w_j = \mu \cdot \varepsilon^p \cdot f' \cdot x_j \quad (7)$$

Όπου $f(\cdot)$ είναι η συνάρτηση ενεργοποίησης του νευρώνα και $f'(\cdot)$ η παράγωγος.

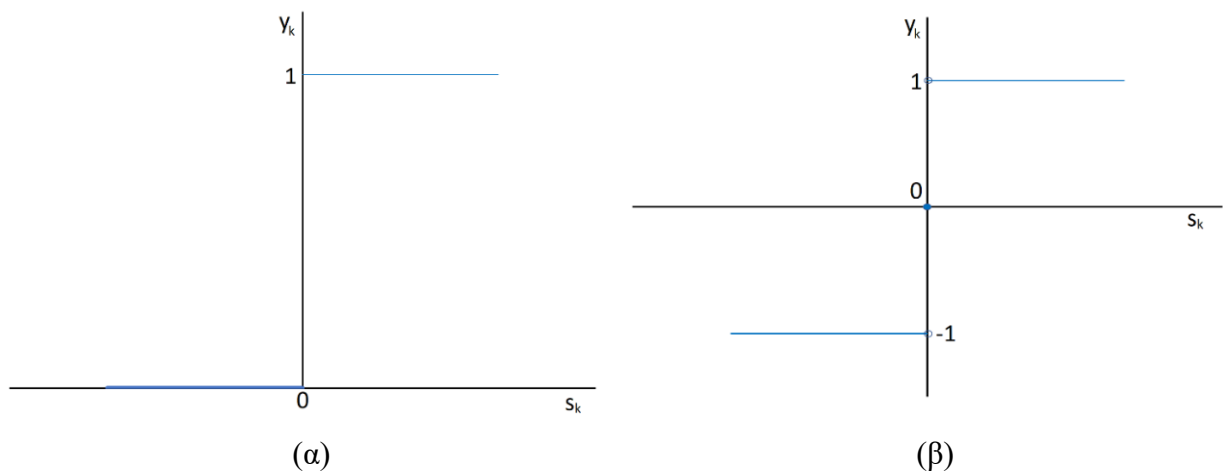
Εξέλιξη του ADALINE ήταν το MADALINE (MULTIPLE ADAPTIVE LINEAR Elements), που ήταν ένας συνδυασμός πολλαπλών ADALINE (Widrow, 1962). Το MADALINE, ήταν το πρώτο Τ.Ν.Δ. που χρησιμοποιήθηκε για την αντιμετώπιση ενός πραγματικού προβλήματος, καθώς αξιοποιήθηκε ως προσαρμοζόμενο φίλτρο που εξαφανίζει την ηχώ στις τηλεφωνικές κλήσεις (Lucky, 1965). Ενώ τα επόμενα χρόνια αξιοποιήθηκαν για την αναγνώριση προτύπων και ομιλίας, την πρόγνωση καιρού, την επεξεργασία σεισμικών σημάτων, την ακύρωση θορύβου σε τηλεφωνικές συσκευές και δρομολογητών (router) (Widrow & Lehr, 1990).

ΣΥΝΑΡΤΗΣΕΙΣ ΕΝΕΡΓΟΠΟΙΗΣΗΣ

Όπως αναφέρθηκε οι ADALINE χρησιμοποίησαν διαφορετική συνάρτηση ενεργοποίησης σε σχέση με τα προγενέστερα νευρωνικά δίκτυα. Η συνάρτηση ενεργοποίησης (στην βιβλιογραφία μπορεί να περιγράφει με τους όρους *activation function* ή *transfer function* ή και *output function*) ομαλοποιεί την είσοδο ενός νευρώνα και καθορίζει την έξοδο του νευρώνα η οποία στην συνέχεια περνάει στο επόμενο επίπεδο. Με την χρήση των συναρτήσεων ενεργοποίησης, μπορεί να προστεθεί μη-γραμμικότητα στο νευρωνικό δίκτυο και με αυτόν τον τρόπο να αποκτήσουν την δυνατότητα επίλυσης μη γραμμικών προβλημάτων (Haykin, 2009). Στην βιβλιογραφία υπάρχουν διάφοροι τύποι συναρτήσεων ενεργοποίησης, ανάλογα με την φύση του προβλήματος και την κατηγορία του νευρωνικού δικτύου υπάρχουν διάφορες επιλογές. Παρακάτω παρατίθενται οι κύριες επιλογές.

Οι συναρτήσεις Heaviside

Οι συναρτήσεις γνωστές και ως συναρτήσεις κατωφλιού ή βηματικές συναρτήσεις (Εικόνα 12α), αν και χρησιμοποιήθηκαν αρχικά στην ηλεκτρομαγνητική για την μετάδοση σημάτων (Heaviside, 1899), ήταν ο τύπος συνάρτησης ενεργοποίησης που χρησιμοποιήθηκε στο μοντέλο των McCulloch & Pitts και χρησιμοποιείται ευρέως έως και σήμερα σε νευρωνικά δίκτυα κατηγοριοποίησης (Hagan, et al., 2014). Η τιμή κάθε νευρώνα με βάση αυτές της συναρτήσεις διαμορφώνεται ως εξής:



Εικόνα 12: α) Η συνάρτηση ενεργοποίησης Heaviside
β) Η συνάρτηση ενεργοποίησης Signum

$$y_k = f(s_k) = \begin{cases} 1 & \forall s_k \geq 0 \\ 0 & \forall s_k < 0 \end{cases} \quad (8)$$

$$s_k^p = \sum_j w_{jk} \cdot y_j^p + \theta_k \quad (9)$$

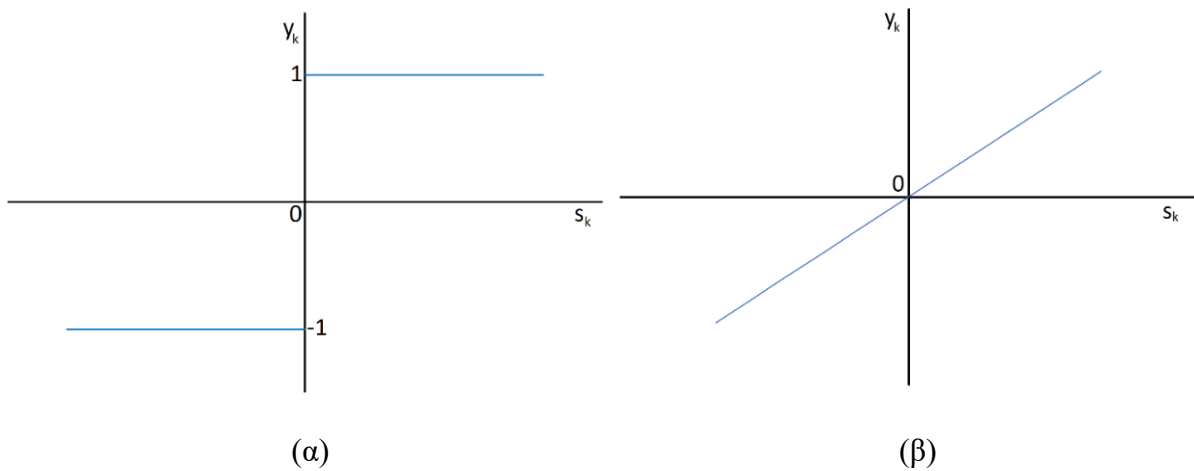
$$y_k' = f'(s_k) = 0 \quad \forall s_k \in \mathbb{R}^* \quad (10)$$

Όπου s_k^p είναι το βαθμισμένο άθροισμα των εισόδων του νευρώνα k από την είσοδο p , w_{jk} είναι το βάρος της σύναψης μεταξύ των νευρώνων j και k και θ_k η τιμή κατωφλιού του νευρώνα. Να σημειωθεί ότι δεν ορίζεται τιμή για την παράγωγο $f'(s_k)$ όταν $s_k = 0$.

Ειδική περίπτωση της συνάρτησης *Heaviside*, είναι η συνάρτηση ενεργοποίησης *signum* (ή απλά *sign*) (Εικόνα 12β), η οποία ορίζεται ως εξής

$$y_k = f(s_k) = \begin{cases} 1 & \forall s_k > 0 \\ 0 & s_k = 0 \\ -1 & \forall s_k < 0 \end{cases} \quad \text{ή} \quad y_k = f(s_k) = \begin{cases} \frac{s_k}{|s_k|} & \forall s_k \neq 0 \\ 0 & s_k = 0 \end{cases} \quad (11)$$

Για την παράγωγο ισχύει το ίδιο με την παράγωγο της *Heaviside*, είναι μηδενική σε όλο το \mathbb{R} , εκτός από το μηδέν στο οποίο δεν ορίζεται. Μια άλλη τροποποίηση της *Heaviside*, είναι αντί να δίνετε η τιμή 0 στις αρνητικές τιμές του s_k , δίνεται η τιμή -1. Η συνάρτηση αυτή ονομάζεται συμμετρική *Heaviside* (Εικόνα 13α), η διαφορά της με την *signum* είναι ότι για $s_k = 0$ η συνάρτηση λαμβάνει την τιμή 1, όπως και στην ,απλή *Heaviside*. Η παράγωγος της μπορεί να περιγραφεί επίσης από την εξίσωση 10 (Hagan, et al., 2014).



Εικόνα 13: α) Η συμμετρική *Heaviside*
β) Η γραμμική συνάρτηση ενεργοποίησης

Οι γραμμικές συναρτήσεις

Όπως αναφέρθηκε χρησιμοποιήθηκαν από τους *Widrow-Hoff* στο σύστημα *ADALINE*, τα κύρια πλεονεκτήματα τους ήταν η δυνατότητα να δίνουν τιμές σε όλο το διάστημα $[-1, 1]$, καθώς και το γεγονός ότι σε αντίθεση με τις συναρτήσεις *Heaviside* είναι παραγωγίσιμες σε όλο το \mathbb{R} , κάτι που ήταν πολύ βασικό στην εξέλιξη των αλγόριθμων εκπαίδευσης.

Η γραμμική συνάρτηση (Εικόνα 13β) περιγράφεται από τις εξής εξισώσεις

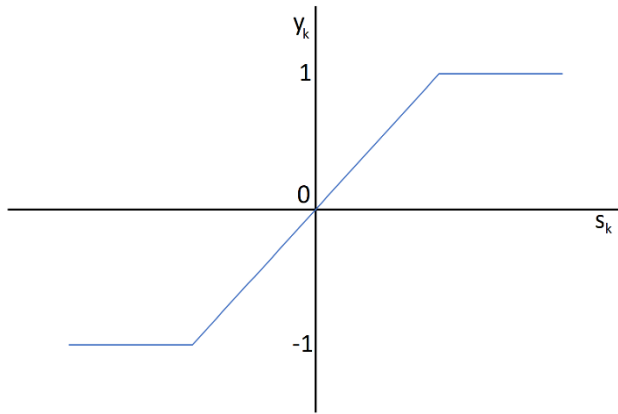
$$y_k = f(s_k) = s_k \quad \forall s_k \in \mathbb{R} \quad (12)$$

$$y'_k = f'(s_k) = 1 \quad \forall s_k \in \mathbb{R} \quad (13)$$

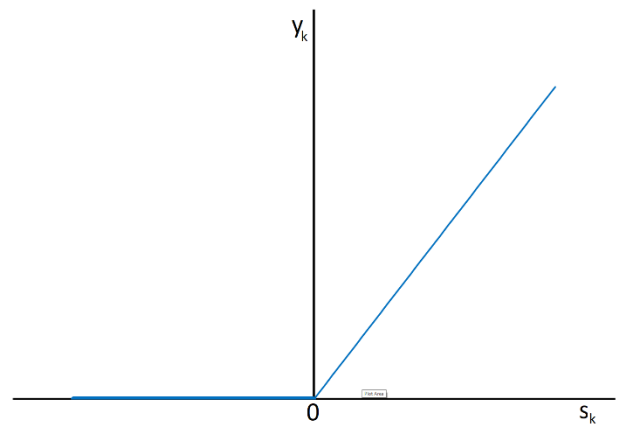
Ειδικές περιπτώσεις της γραμμικής συνάρτησης είναι η συμμετρική συνάρτηση κορεσμού (*symmetric saturating Linear*, στην βιβλιογραφία συναντάται και ως τμηματικά γραμμική συνάρτηση ή ημιγραμμική (*semi-linear*) (Krose & van de Smagt, 1996)(Εικόνα 14α) η οποία περιγράφεται από τις παρακάτω εξισώσεις

$$y_k = f(s_k) = \begin{cases} 1 & \forall s_k \in (1, +\infty) \\ s_k & \forall s_k \in [-1, 1] \\ -1 & \forall s_k \in (-\infty, -1) \end{cases} \quad (14)$$

$$y'_k = f'(s_k) = \begin{cases} 1 & \forall s_k \in [-1, 1] \\ 0 & \forall s_k \in (-\infty, -1) \cup (1, +\infty) \end{cases} \quad (15)$$



(α)



(β)

Εικόνα 14: α) Η συμμετρική συνάρτηση κορεσμού
β) Η θετικά γραμμική

Και η θετικά γραμμική (*Positive Linear*) γνωστή και ως *ReLU* (*Rectified Linear Unit*) (Εικόνα 14β) η οποία ακολουθεί την γραμμική συνάρτηση για τις θετικές τιμές του s_k , ενώ για όλες τις αρνητικές παίρνει την τιμή μηδέν. Το γεγονός αυτό τις καθιστά άκρως ελκυστικές συναρτήσεις, καθώς ένας μεγάλος αριθμός συνάψεων θα λαμβάνει μηδενικές τιμές, με αποτέλεσμα την πιο γρήγορη εκπαίδευση ενός νευρωνικού δικτύου, παράλληλα η απλή μορφή της παραγώγου, που μπορεί να λάβει μόνο τις τιμές 0 και 1 βοηθάει στην διευθέτηση του προβλήματος με την εξαφανιζόμενη ή την εκρηγνύουσα παράγωγο (Witten, et al., 2017). Για την περιγραφή της, χρησιμοποιούνται οι εξισώσεις που ακολουθούν. εξαφανιζόμενη ή την εκρηγνύουσα παράγωγο (Witten, et al., 2017). Για την περιγραφή της, χρησιμοποιούνται οι εξισώσεις που ακολουθούν.

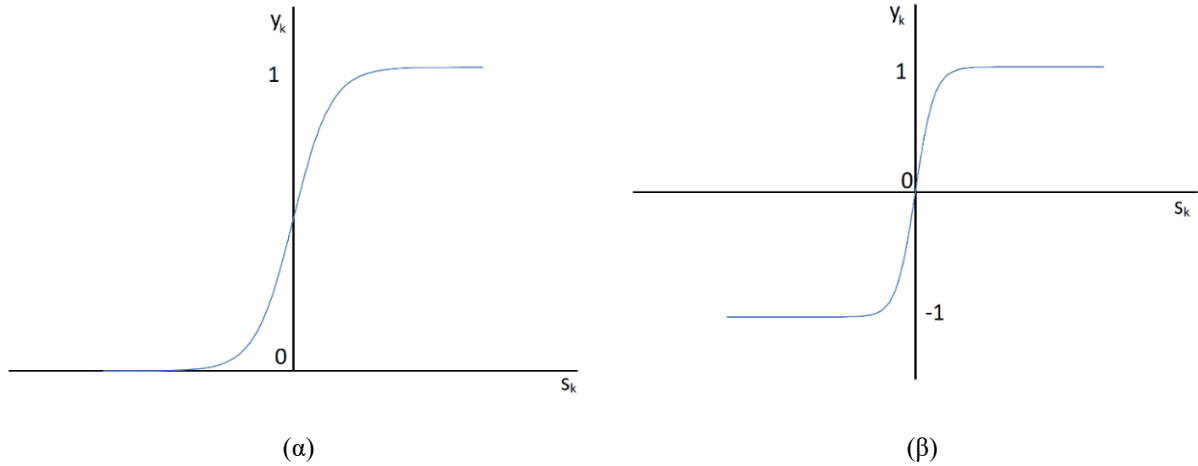
$$y_k = f(s_k) = \max(0, s_k) \quad \forall s_k \in \mathbb{R} \quad (16)$$

$$y'_k = f'(s_k) = \begin{cases} 1 & \forall s_k \geq 0 \\ 0 & \forall s_k < 0 \end{cases} \quad (17)$$

Η επιτυχία των *ReLU*, οδήγησε στην δημιουργία μιας οικογένειας συναρτήσεων ενεργοποιήσεως, όπως η *Leaky ReLU* (*Leaky Rectified Linear Unit*), η *GELU* (*Gaussian Error Linear Unit*), η *SiLU* (*Sigmoid Linear Unit*), η *ELU* (*Exponential Linear Units*) και η *Softplus* (ή *SmoothReLU*). Οι παραπάνω συναρτήσεις ενεργοποίησης έχουν εφαρμογές κυρίως σε νευρωνικά δίκτυα με πάρα πολλά επίπεδα (βαθιά νευρωνικά δίκτυα), που ξεφεύγουν από την παρούσα εργασία, και για αυτό το λόγο δεν θα αναλυθούν περαιτέρω.

Οι σιγμοειδής συναρτήσεις

Εκτός από τις συναρτήσεις που ανήκουν στην οικογένεια της *ReLU*, οι πιο διαδεδομένες συναρτήσεις και αυτές που είχαν κυριαρχήσει μέχρι την χρησιμοποίηση των *ReLU*, είναι οι σιγμοειδής συναρτήσεις. Οι σιγμοειδής συναρτήσεις, έχουν πάρει το όνομα τους από την γραφική αναπαράστασή τους, που θυμίζει το τελικό ς (σίγμα) και χρησιμοποιήθηκαν αρκετά τα προηγούμενα τριάντα χρόνια. Χωρίζονται σε δύο μεγάλες κατηγορίες, τις λογαριθμικές (ή λογιστικές) σιγμοειδής συναρτήσεις (*logistic* ή *logsis*) (Εικόνα 15α) και τις συναρτήσεις υπερεφαπτομένης (*tanh*) (Εικόνα 15β).



Εικόνα 15: α) Η λογιστική συνάρτηση ενεργοποίησης
β) Η συνάρτηση ενεργοποίησης υπερεφαπτομένης

Οι συναρτήσεις αυτές μπορούν να περιγράψουν από τις ακόλουθες εξισώσεις

Η λογιστική από

$$y_k = f(s_k) = \frac{1}{1 + e^{-s_k}} \quad \forall s_k \in \mathbb{R} \quad (18)$$

$$y'_k = f'(s_k) = f(s_k) \cdot (1 - f(s_k)) = \left(\frac{1}{1 + e^{-s_k}} \right) \left(1 - \frac{1}{1 + e^{-s_k}} \right) \quad \forall s_k \in \mathbb{R} \quad (19)$$

Και η συνάρτηση υπερεφαπτομένης από

$$y_k = f(s_k) = \frac{e^{s_k} - e^{-s_k}}{e^{s_k} + e^{-s_k}} \quad \forall s_k \in \mathbb{R} \quad (20)$$

$$y'_k = f'(s_k) = 1 - f(s_k)^2 = \frac{4}{(e^{s_k} + e^{-s_k})^2} \quad \forall s_k \in \mathbb{R} \quad (21)$$

Οι σιγμοειδής συναρτήσεις είναι συνεχείς και μονοτονικές, ενώ παραμένουν πεπερασμένες ακόμα και όταν το s_k πλησιάζει το $\pm\infty$. Το ότι είναι μονοτονικές βοηθάει στην εκπαίδευση του νευρωνικού, ενώ οι κλίσεις της συνάρτησης συμπεριφέρονται καλά ως συνάρτηση του s_k (Baughman & Liu, 1995). Το μεγαλύτερο πλεονέκτημα των σιγμοειδών συναρτήσεων, όπως φαίνεται και από τις εξισώσεις 19 και 21, είναι ότι από την στιγμή που υπολογιστεί η τιμή της συνάρτησης y_k , ο υπολογισμός της παραγώγου είναι εξαιρετικά απλός, κάτι που βελτιστοποιεί

τους αλγόριθμους εκπαίδευσης. Οι λογιστικές επειδή παίρνουν τιμές μεταξύ του $[0,1]$, είναι οι ιδανικές συναρτήσεις για μοντέλα που υπολογίζουν πιθανότητες, αφού και αυτές έχουν το ίδιο πεδίο τιμών. Από την άλλη, οι υπερβολικές συναρτήσεις επειδή έχουν ως κέντρο το μηδέν και πιέζουν τις τιμές εισόδου στο διάστημα $[-1,1]$ είναι πιο εύκολο να εκπαιδευτούν σε σχέση με τις λογιστικές. Το μεγαλύτερο μειονέκτημα και των δύο, είναι ότι για πολύ μεγάλες ή πολύ μικρές τιμές, η γραφική τους τείνει ασυμπτωτικά στο 1 και είτε στο 0 (λογιστική) είτε στο -1 (υπερεφαπτομένης), ως εκ τούτου η κλίση είναι πολύ μικρή σε εκείνα τα σημεία και η εκπαίδευση εξαιρετικά αργή (Witten, et al., 2017).

ΑΛΓΟΡΙΘΜΟΣ ΤΗΣ ΑΠΟΤΟΜΗΣ ΚΑΘΟΔΟΥ

Σε όλες τις μεθόδους εκπαίδευσης χρησιμοποιείτε μια συνάρτηση κόστους $K(\mathbf{W})$, $\mathbf{W} \in \mathbb{R}^l$, (όπου l το σύνολο των κόμβων) με την ελαχιστοποίηση της οποίας γίνεται και η ελαχιστοποίηση του σφάλματος του νευρωνικού δικτύου. Στον κανόνα *Delta*, η εκπαίδευση γίνεται χρησιμοποιώντας τον αλγόριθμο της απότομης καθόδου¹ (*gradient descent*) για την ελαχιστοποίηση του αθροίσματος των τετραγώνων του σφάλματος. Εναλλακτικά από την ελαχιστοποίηση του αθροίσματος των τετραγώνων του σφάλματος, μπορούν να χρησιμοποιηθεί το άθροισμα της απόλυτης τιμής του σφάλματος ή η συνάρτηση 0/1 *loss* κατά την οποία $L_{0/1}(d, y) = 0$ εάν και μόνο αν η εκτιμώμενη τιμή ταυτίζεται με την επιθυμητή τιμή, ενώ διαφορετικά λαμβάνει την τιμή της μονάδας (Russel & Norvig, 2020)

Φθίνουσες διευθύνσεις

Οι φθίνουσες διευθύνσεις, είναι οι διευθύνσεις που παράγονται από τις μεθόδους ελαχιστοποίησης που χρησιμοποιούν γραμμική αναζήτηση. Η δυνατή διεύθυνση (διάνυσμα) \mathbf{S} θα ονομάζεται φθίνουσα διεύθυνση εάν η τιμή της συνάρτησης μειώνεται κατά μήκος της, δηλαδή εάν:

$$K(\mathbf{W} + \mu \cdot \mathbf{S}) < K(\mathbf{W}) \quad \mu > 0 \quad (22)$$

Χρησιμοποιώντας την σειρά Taylor 1^{ης} τάξης προκύπτει ότι

$$K(\mathbf{W} + \mu \cdot \mathbf{S}) \approx K(\mathbf{W}) + \mu \cdot \mathbf{g}^T \cdot \mathbf{S} \quad (23)$$

$$\text{Όπου } \mathbf{g} = \nabla K(\mathbf{W}) = \frac{\partial K(\mathbf{W})}{\partial \mathbf{W}}$$

Επομένως θα πρέπει

$$K(\mathbf{W} + \mu \cdot \mathbf{S}) < K(\mathbf{W}) \Leftrightarrow$$

$$\cancel{K}(\mathbf{W}) + \mu \cdot \mathbf{g}^T \cdot \mathbf{S} - \cancel{K}(\mathbf{W}) < 0 \stackrel{\mu > 0}{\Leftrightarrow}$$

$$\mathbf{g}^T \cdot \mathbf{S} < 0 \quad (24)$$

Το διάνυσμα \mathbf{S} , θα ονομάζεται διεύθυνση της πιο απότομης καθόδου, εάν η συνάρτηση παρουσιάζει κατά μήκος του, την μέγιστη μείωση της (Theodoridis, 2020).

¹ Στην ελληνική βιβλιογραφία, μπορεί να αποδοθεί και ως αλγόριθμος καθοδικής κλίσης

Η μέθοδος της απότομης καθόδου

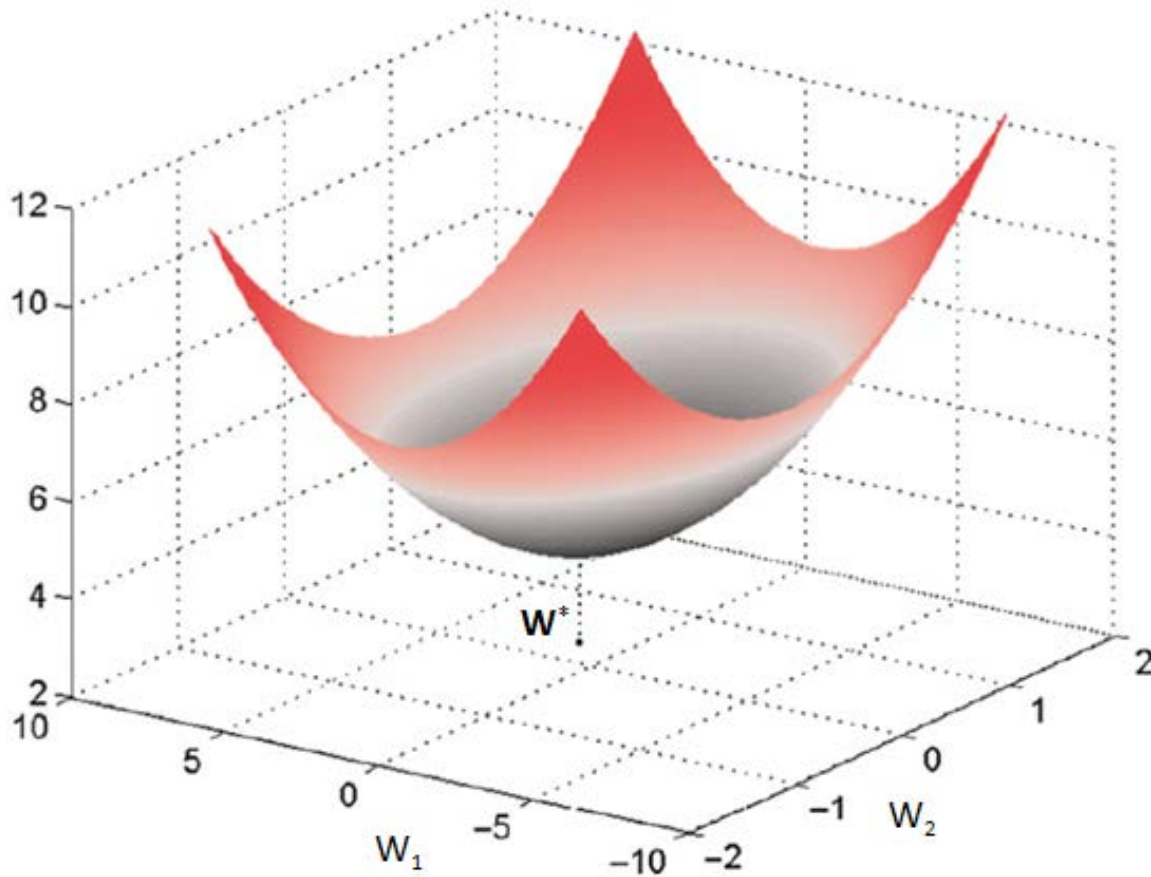
Στην συνάρτηση ελαχιστοποίησης σφάλματός, η κατεύθυνση \mathbf{S} ισούται με την διαφοροποίηση των βαρών των συνάψεων. Δηλαδή $\mathbf{S} = \Delta \mathbf{W}$. Όλοι οι αλγόριθμοι ελαχιστοποίησης της συνάρτησης σφάλματος ξεκινούν από μια αρχική εκτίμηση $\mathbf{W}(0)$ και δημιουργούν μια ακολουθία $\mathbf{W}^{(i)}, i = 1, 2, \dots$ τέτοια ώστε

$$\mathbf{W}^{(i)} = \mathbf{W}^{(i-1)} + \mu_i \cdot \Delta \mathbf{W}^{(i)} \quad i \in \mathbb{N}^* \quad (25)$$

Με i συμβολίζονται οι εποχές (*epoch*) του συστήματος. Με αυτόν τον όρο περιγράφεται ο υπολογισμός νέων τιμών για τα βάρη του συστήματος. Η διαφοροποίηση των μεθόδων έχει να κάνει με την επιλογή του ρυθμού μάθησης (η μέγεθος βήματος) μ_i καθώς και της κατεύθυνση ερευνάς $\Delta \mathbf{W}^{(i)}$

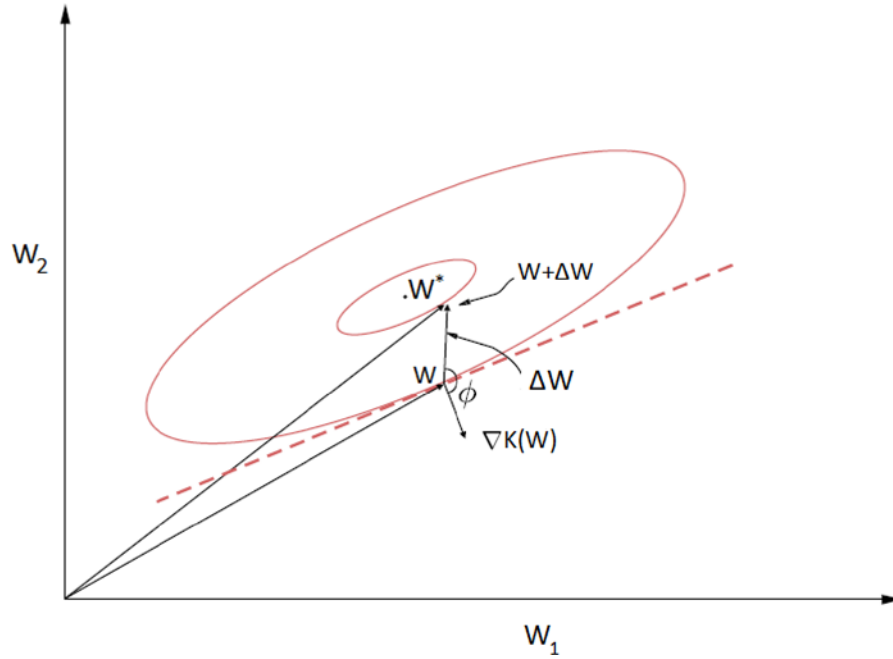
Αντικαθιστώντας στην εξίσωση 23 προκύπτει

$$K(\mathbf{W}^{(i)}) \simeq K(\mathbf{W}^{(i-1)} + \mu_i \cdot \Delta \mathbf{W}^{(i)}) \simeq K(\mathbf{W}^{(i-1)}) + \mu_i \cdot \mathbf{g}^T \cdot \Delta \mathbf{W}^{(i)} \quad (26)$$

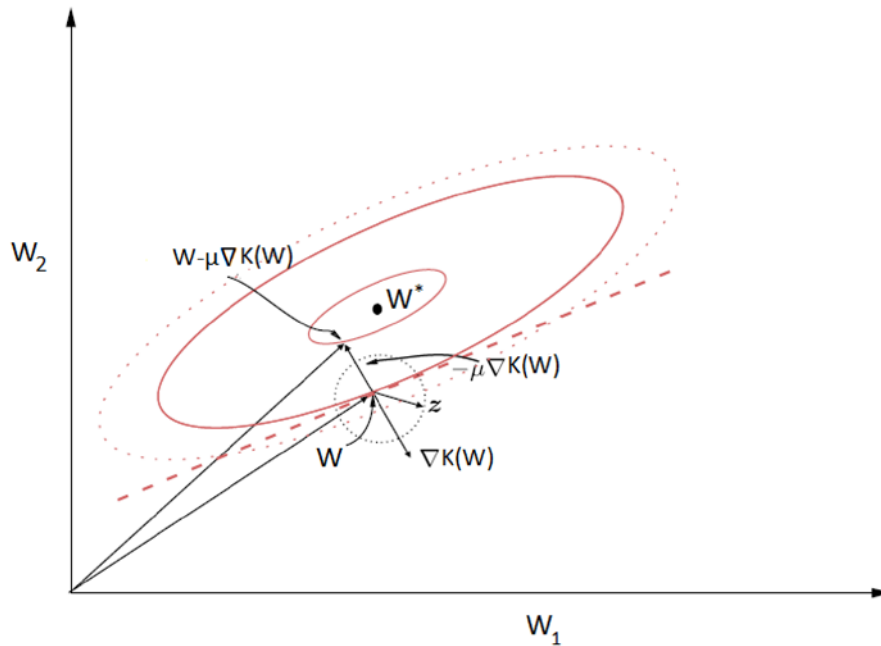


Εικόνα 16: Γραφική παράσταση της συνάρτησης σφάλματος για δισδιάστατη περίπτωση (Theodoridis, 2020)

Στην μέθοδο της απότομης καθόδου, το $\Delta \mathbf{W}^{(i)}$ επιλέγεται έτσι ώστε $K(\mathbf{W}^{(i)}) < K(\mathbf{W}^{(i-1)})$, ώσπου να βρεθεί ένα ελάχιστο \mathbf{W}^* , τέτοιο ώστε $K(\mathbf{W}^*) \leq K(\mathbf{W}^{(i)}), \forall \mathbf{W} \in \mathbb{R}^l$ (Εικόνα 16). Τα $\Delta \mathbf{W}$ που επιλέγονται θα πρέπει να πληρούν τις προϋποθέσεις της εξίσωσης 24, δηλαδή $\nabla K(\mathbf{W}^{(i-1)})^T \cdot \Delta \mathbf{W}^{(i)} < 0$



Εικόνα 17: Αν γίνει αποτύπωση του κώνου που εμφανίζεται στην εικόνα 16 με ισοβαρείς καμπύλες σφάλματος θα προκύψει η παραπάνω εικόνα. Όπως φαίνεται στην εικόνα, το διάνυσμα της παραγωγών των βαρών, θα είναι κάθετο στην εφαπτομένη της ισοβαρής καμπύλης, επομένως το επόμενο βήμα προς οποιαδήποτε κατεύθυνση θα είναι το γινόμενο της παραγωγού επί το ρυθμό μάθησης, επί το συνημίτονο της γωνίας $(180^\circ - \phi)$ (Theodoridis, 2020)



Εικόνα 18: Επομένως η μεγαλύτερη απόσταση θα διανυθεί, έναν η μετακίνηση γίνει κάθετα στην εφαπτομένη (Theodoridis, 2020)

Όπως αναλύθηκε σχηματικά στις [εικόνες 17 & 18](#), η διεύθυνση της πιο απότομης καθόδου, θα ταυτίζεται με το αρνητικό διάνυσμα της παραγωγού, επομένως η [εξίσωση 25](#) μπορεί να γραφεί ως

$$\mathbf{W}^{(i)} = \mathbf{W}^{(i-1)} - \mu_i \cdot \nabla K(\mathbf{W}^{(i)}) \quad i \in \mathbb{N}^* \quad (27)$$

Ο κανόνας *Delta*, όπως αναφέρθηκε προσπαθεί να μειώσει το άθροισμα του τετραγώνου του σφάλματος, επομένως η συνάρτηση κόστους διαμορφώνεται σε

$$K(\mathbf{W}) = (d - \mathbf{W}^T \cdot \mathbf{X})^2 = d^2 - 2 \cdot \mathbf{W}^T \cdot \mathbf{P} + \mathbf{W}^T \cdot R \cdot \mathbf{W} \quad (28)$$

με $\mathbf{P} \triangleq d \cdot \mathbf{X}$ και $R \triangleq \mathbf{X} \cdot \mathbf{X}^T$

ενώ η παράγωγος της, θα είναι

$$\nabla K(\mathbf{W}) = -2 \cdot \mathbf{P} + 2 \cdot R \cdot \mathbf{W} \quad (29)$$

Αντικαθιστώντας στην [εξίσωση 26](#) προκύπτει

$$\mathbf{W}^{(i)} = \mathbf{W}^{(i-1)} - \mu_i \cdot (\mathbf{P} - R \cdot \mathbf{W}^{(i-1)}) \Leftrightarrow$$

$$\mathbf{W}^{(i)} = \mathbf{W}^{(i-1)} - \mu_i \cdot (d \cdot \mathbf{X} - \mathbf{X} \cdot \mathbf{X}^T \cdot \mathbf{W}^{(i-1)}) \Leftrightarrow$$

$$\mathbf{W}^{(i)} = \mathbf{W}^{(i-1)} - \mathbf{X} \cdot (d - (\mathbf{X}^T \cdot \mathbf{W}^{(i-1)})) \Leftrightarrow$$

$$\mathbf{W}^{(i)} = \mathbf{W}^{(i-1)} + 2 \cdot \mu_i \cdot (d - (\mathbf{X}^T \cdot \mathbf{W}^{(i-1)})) \cdot \mathbf{X}$$

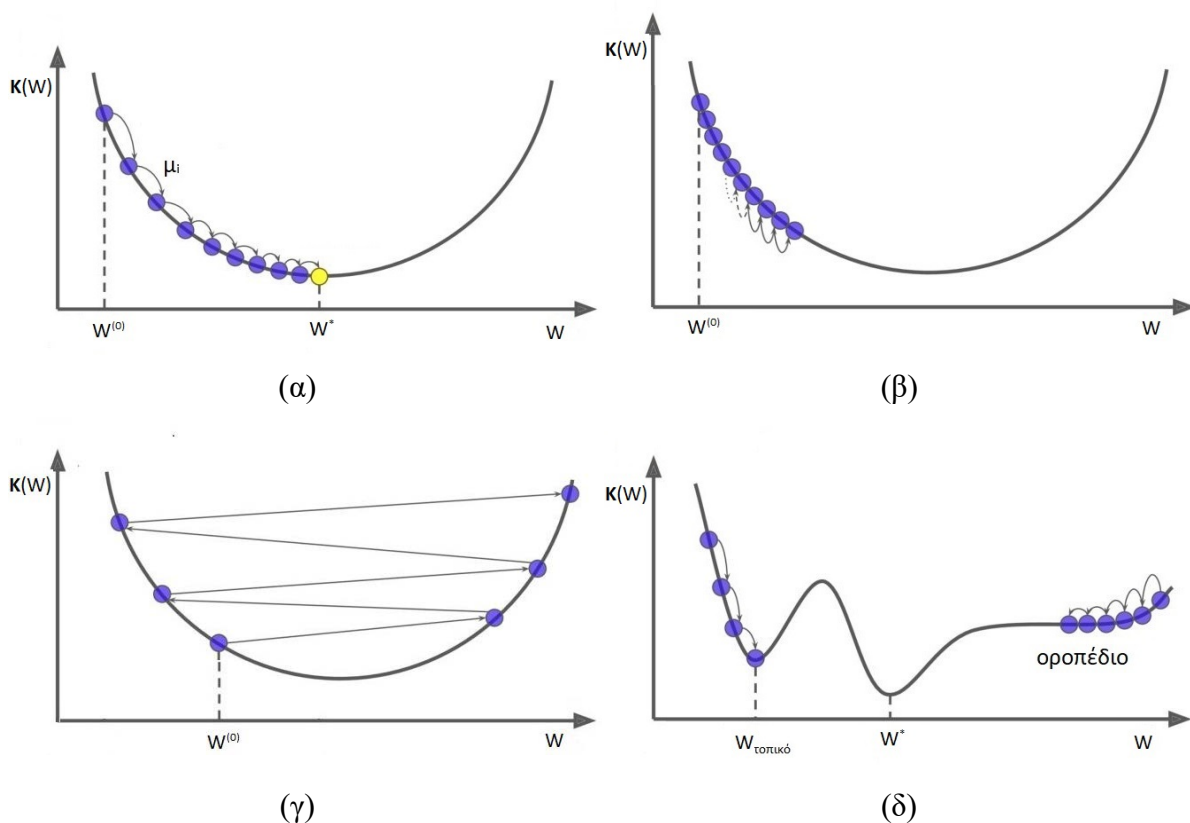
που είναι ισοδύναμη με την [εξίσωση 4](#).

Διαφοροποιήσεις αλγόριθμου απότομης καθόδου

Υπάρχουν τρεις τύποι αλγόριθμων απότομης καθόδου. Ο πρώτος που είναι η απλή εκδοχή του αλγορίθμου, ονομάζεται συνολική απότομη κάθοδος (batch gradient descent), χρησιμοποιεί το σύνολο των διαθέσιμων δεδομένων για κάθε εποχή. Λόγω του ότι χρησιμοποιεί όλα τα δεδομένα, είναι πιο πιθανό να συγκλίνει στο \mathbf{W}^* . Ωστόσο για αυτόν ακριβώς το λόγο η εκπαίδευση νευρωνικών δικτύων με χρήση αλγορίθμων αυτού του τύπου είναι εξαιρετικά αργή, ειδικά όταν χρησιμοποιείται μεγάλος αριθμός δεδομένων. Για το λόγο αυτό αναπτύχθηκαν οι επόμενοι δύο τύποι απότομης καθόδου. Στην στοχαστική απότομη κάθοδο (Stochastic Gradient Descent - **SGD**) σε κάθε εποχή, χρησιμοποιείτε μόνο ένα ζεύγος τιμών, κάτι που την καθιστά εξαιρετικά γρήγορη μέθοδο. Παρόλα αυτά είναι σχεδόν απίθανο το σύστημα να συγκλίνει στο \mathbf{W}^* . Στις περισσότερες περιπτώσεις το σύστημα θα συγκλίνει σε μια περιοχή κοντά στο \mathbf{W}^* , αλλά υπάρχει ο κίνδυνος να σταματήσει σε κάποιο τοπικό ακρότατο. Οι αδυναμίες των παραπάνω μεθόδων οδήγησαν στην δημιουργία μιας τρίτης εκδοχής, αυτή της Σταδιακής Καθόδου Μικρών Δεσμών (Mini-Batch Gradient Descent). Σε αυτού του τύπου τους αλγόριθμους τα δεδομένα μοιράζονται σε μικρότερα διαχειρίσιμα σύνολα δεδομένων. Σε κάθε εποχή ο αλγόριθμος διαχειρίζεται ένα από αυτά τα σύνολα. Αυτό βελτιώνει αρκετά την ταχύτητα του, ενώ παράλληλα αποφεύγονται μεγάλες διακυμάνσεις στη συνάρτηση κόστους και καθώς και συγκλίσεις σε διαφορετικά τοπικά ελάχιστα ([Bisong, 2019](#)).

ΡΥΘΜΟΣ ΜΑΘΗΣΗΣ

Το πρόβλημα που δημιουργείτε με την χρήση του αλγόριθμου της απότομης καθόδου, έχει να κάνει με την κατάλληλη επιλογή του ρυθμού μάθησης, μ_i . Το μ_i , είναι μια υπερπαράμετρος¹, μπορεί να παραμένει σταθερό ή να αλλάζει ανά εποχή. Από την [εξίσωση 27](#), είναι εμφανές ότι το μ_i μπορεί να καθορίσει την επίδραση της παραγώγου στα νέα βάρη. Επομένως, η επιλογή του σωστού ρυθμού μάθησης μ_i , μπορεί να οδηγήσει στην εύκολη εύρεση του \mathbf{W}^* ([Εικόνα 19α](#)). Στην περίπτωση που επιλεγθεί πολύ μικρή τιμή μ_i , το σύστημα να μην θα βρει το \mathbf{W}^* , αλλά η εκπαίδευση θα είναι εξαιρετικά αργή ([Εικόνα 19β](#)). Στον αντίποδα, η επιλογή μεγάλης τιμής για το μ_i , μπορεί να καταστήσει αδύνατη την εκπαίδευση του νευρωνικού δικτύου, καθώς λόγω των μεγάλων βημάτων που θα κάνει ο αλγόριθμος, θα περνάει το \mathbf{W}^* και θα πηγαίνει στην απέναντι πλευρά της καμπύλης και ενδεχομένως η επόμενη πρόβλεψη να είναι χειρότερη από την προηγούμενη. Μια τέτοια συμπεριφορά μπορεί να κάνει τον αλγόριθμο να αποκλίνει, λαμβάνοντας όλο και μεγαλύτερες τιμές αποτυγχάνοντας να βρει μια αποδεκτή λύση ([Εικόνα 19γ](#)). Η επιλογή του κατάλληλου μ_i , έχει να κάνει με το πρόβλημα, καθώς και το σύνολο των δεδομένων ([Géron, 2017](#)).



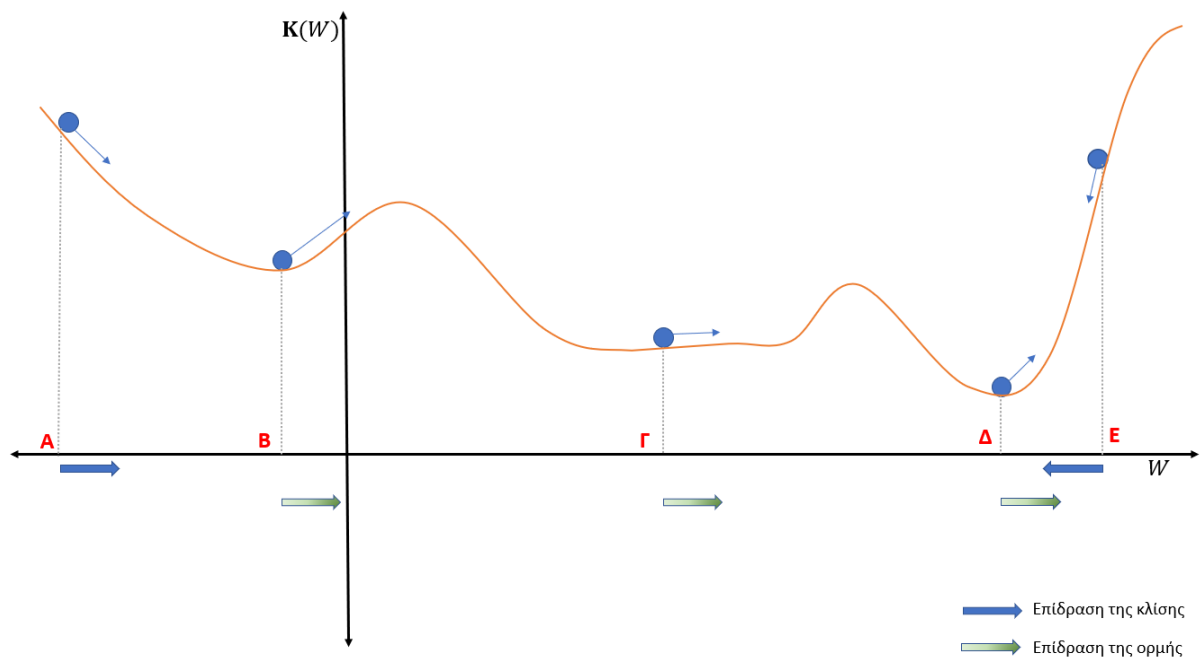
Εικόνα 19 α) Με την επιλογή κατάλληλου ρυθμού μάθησης μπορεί να βρεθεί το \mathbf{W}^*
 β) Με την επιλογή πολύ μικρού ρυθμού μάθησης, μπορεί να μην βρεθεί το \mathbf{W}^* γρήγορα
 γ) Με την επιλογή μεγάλου ρυθμού μάθησης, αντί να μειώνεται η συνάρτηση κόστους, θα αυξάνεται
 δ) Αν η παράγωγος της συνάρτησης κόστους, μηδενίζεται σε πολλά σημεία, υπάρχει κίνδυνος να μην βρεθεί το \mathbf{W}^* (Géron, 2017)

¹ Με τον όρο αυτό περιγράφονται οι παράμετροι, για τις οποίες η τιμή μπορεί να επιλεγεί χειροκίνητα από τον χρήστη.

Ο ΠΑΡΑΓΟΝΤΑΣ ΤΗΣ ΟΡΜΗΣ

Η μέθοδος της απότομης καθόδου μπορεί να δώσει λύση σε ένα μεγάλο αριθμό προβλημάτων, αλλά έχει μια σημαντική αδυναμία. Στις περιπτώσεις που η τιμή της παραγώγου της συνάρτησης του κόστους μηδενίζεται σε πολλά σημεία (άρα η συνάρτηση κόστους έχει παραπάνω από ένα ακρότατο), ανεξάρτητα από τον ρυθμό μάθησης που επιλεγεί, το σύστημα ενδεχομένως να μην καταφέρει να βρει το W^* , αλλά να σταματήσει σε κάποιο τοπικό ακρότατο ή σε κάποιο «οροπέδιο»¹(Εικόνα 19δ) (Géron, 2017).

Την λύση στο παραπάνω πρόβλημα την έδωσε ο *Boris Polyak* το 1964 (Polyak, 1964), ο οποίος πρότεινε μια τροποποίηση του αλγορίθμου της απότομης καθόδου, με την προσθήκη ενός συντελεστή που ονομάστηκε ορμή. Έτσι όπως στην κλασική φυσική ένα σώμα που έχει αναπτύξει μια συγκεκριμένη ταχύτητα, η ορμή είναι αυτή που αντιστέκεται στην αλλαγή της κατεύθυνσης της κίνησης, έτσι και στην περίπτωση της απότομης καθόδου, αν ο αλγόριθμος βελτιστοποίησης κινείται σε μια γενική κατεύθυνση, η ορμή τον αναγκάζει να «αντισταθεί» σε τυχόν αλλαγές κατεύθυνσης καθώς και στην αποφυγή τυχόν τοπικών ακροτάτων ή οροπεδίων. Έτσι αν για παράδειγμα γίνει μια αναζήτηση όπως εμφανίζεται στην Εικόνα 20, το σύστημα αρχικά θα αποκτήσει ταχύτητα λόγω της κλίσης (σημείο Α), στην συνέχεια θα αποφύγει το τοπικό ακρότατο που εμφανίζεται στο σημείο Β, λόγω της ορμής που έχει αποκτήσει. Με τον ίδιο τρόπο θα αποφύγει και το οροπέδιο στο σημείο Γ και θα φτάσει στο σημείο Δ, εκεί παρόλο που είναι ολικό ελάχιστο το σύστημα λόγω ορμής θα συνεχίσει για λίγο λόγω ορμής και θα φτάσει στο σημείο Ε, αλλά θα επιστρέψει ξανά και θα σταματήσει στο σημείο Δ. Αν δεν υπήρχε η ορμή το σύστημα θα σταματούσε στο σημείο Β.



Εικόνα 20: Επίδραση της ορμής στην εύρεση του W^*

Σημείο Α: Τυχαίο σημείο έναρξης της αναζήτησης, το σύστημα έχει μηδενική ορμή, αλλά μεγάλη κλίση

Σημείο Β: Τοπικό ελάχιστο, το σύστημα έχει αρνητική κλίση, αλλά μεγάλη κλίση

Σημείο Γ: Οροπέδιο, το σύστημα έχει μηδενική κλίση, αλλά μεγάλη ορμή

Σημείο Δ: Ολικό ελάχιστο (W^*), το σύστημα έχει μηδενική κλίση, αλλά μεγάλη ορμή

Σημείο Ε: Το σύστημα έχει μηδενική ορμή, αλλά μεγάλη κλίση

¹ Σημείο της συνάρτησης στο οποίο η κλίση είναι μηδενική ή σχεδόν μηδενική.

Οι εξισώσεις που περιγράφουν την προσθήκη της ορμής στον υπολογισμό των νέων βαρών του συστήματος, είναι:

$$\mathbf{m}^{(i)} = \beta \cdot \mathbf{m}^{(i-1)} + \mu_i \cdot \nabla K(\mathbf{W}^{(i-1)}) \quad i \in \mathbb{N}^* \quad (30)$$

$$\mathbf{W}^{(i)} = \mathbf{W}^{(i-1)} - \mathbf{m} \quad i \in \mathbb{N}^* \quad (31)$$

Όπου με m συμβολίζεται η ορμή, ενώ ο όρος β είναι μια δεύτερη υπερπαράμετρος, η οποία αντιστοιχεί στον «συντελεστή τριβής» του συστήματος. Έτσι σε συστήματα με υψηλή τριβή παίρνει την τιμή 0 (σε αυτή την περίπτωση ουσιαστικά το σύστημα λαμβάνει την μορφή της απλής μορφής της απότομης καθόδου), ενώ σε συστήματα χωρίς τριβή την τιμή 1. Η πιο συνηθισμένη και ευρέως αποδεκτή τιμή για τον συντελεστή τριβής είναι το 0.9 (Qian, 1999).

ΜΕΤΑΒΛΗΤΟΣ ΡΥΘΜΟΣ ΜΑΘΗΣΗΣ

Η επιλογή του κατάλληλου σταθερού ρυθμού μάθησης είναι δύσκολη έως αδύνατη. Γενικά συνίσταται η επιλογή ενός σχετικά μεγάλου μ_i αρχικά και σταδιακή μείωση του όσο πλησιάζει στο \mathbf{W}^* (Murata, 1998).

Μια απλή εφαρμογή είναι

$$\mu_i = \begin{cases} \mu_0 & t < t^* \\ \frac{\mu_{t-1}}{\mu_{t-1} + \mu_0} \cdot \mu_0 & t \geq t^* \end{cases} \quad (32)$$

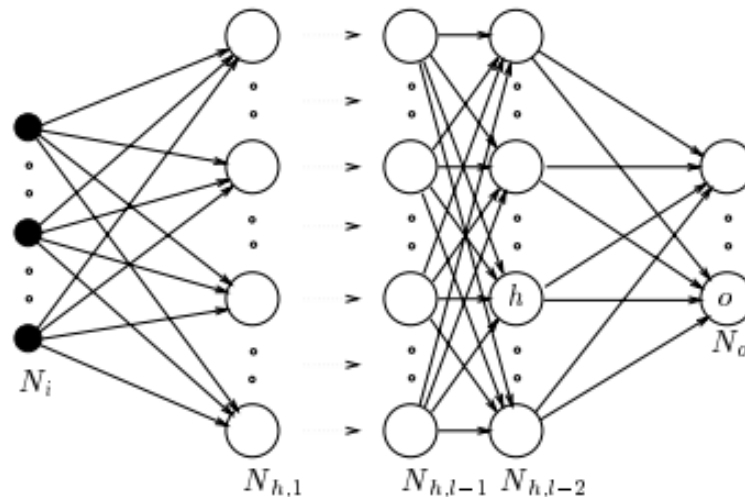
Όπου το t^* , είναι η χρονική στιγμή αλλαγής και μ_0 είναι ο αρχικός ρυθμός μάθησης. Όμως ο καθορισμός ενός κατάλληλου t^* είναι δύσκολο να εκτιμηθεί (Murata, 1998).

Η λύση σε αυτό είναι να συσχετιστεί με το σφάλμα. Οι κανόνες προσαρμογής του ρυθμού μάθησης μπορούν να συνοψιστούν σε

1. Εάν το μέσο τετραγωνικό σφάλμα (mean square error - MSE) αυξηθεί περισσότερο από ένα ποσοστό ζ (τυπικό εύρος τιμών είναι από 1%–5%) μετά από την εφαρμογή ενός νέου διανύσματος \mathbf{W}_i , το \mathbf{W}_i αγνοείται και το σύστημα κρατάει το \mathbf{W}_{i-1} . Και ο ρυθμός μάθησης θα προκύπτει από την σχέση $\mu_{i+1} = \rho \cdot \mu_i$, όπου ρ μια σταθερά (τυπική τιμή $\rho = 0.7$, γενικά θα πρέπει $\rho \in (0,1)$) προκειμένου να μειωθεί και ο συντελεστής ορμής (εάν χρησιμοποιείτε ορμή) μηδενίζεται.
2. Εάν το MSE μειώνεται μετά από μια ενημέρωση του διανύσματος βαρών, τότε αυτή γίνεται αποδεκτή και ο ρυθμός μάθησης θα προκύπτει από την σχέση $\mu_{i+1} = \eta \cdot \mu_i$, όπου η μια σταθερά (τυπική τιμή $\eta = 1.05$ γενικά θα πρέπει $\eta > 1$, Εάν χρησιμοποιείτε ορμή και έχει μηδενιστεί, τότε γίνεται επαναφορά στην αρχική της τιμή).
3. Εάν το MSE αυξάνεται μετά από μια ενημέρωση του διανύσματος βαρών, αλλά σε ποσοστό μικρότερο από ζ , τότε αυτή γίνεται αποδεκτή, αλλά δεν υπάρχει αλλαγή του ρυθμού μάθησης στην επόμενη εποχή. Εάν χρησιμοποιείτε ορμή και έχει μηδενιστεί, τότε γίνεται επαναφορά στην αρχική της τιμή (Hagan, et al., 2014).

Η ΜΕΘΟΔΟΣ BACK PROPAGATION

Ο κανόνας των *Widrow & Hoff* ήταν απλός και αποτελεσματικός και στην ουσία εφαρμόστηκε ευρέως τα επόμενα χρόνια σε περιπτώσεις γραμμικού διαχωρισμού. Όμως είχε ένα βασικό μειονέκτημα, μπορούσε να υπολογίσει το σφάλμα μόνο όταν το Τ.Ν.Δ. είχε ένα επίπεδο (Kim, 2017). Την ίδια στιγμή παρόλο που ο *Rosenblatt* ισχυρίστηκε ότι τα *Perceptron* (όπως και τα *ADALINE*) μπορούν να επιλύσουν οτιδήποτε μπορεί να προβληθεί σε ένα επίπεδο (Rosenblatt, 1961), αποδείχθηκε ότι αδυνατούσαν να επιλύσουν απλά προβλήματα που ήταν μην γραμμικώς διαχωρίσιμα, καθώς η επεξεργασία τους απαιτεί την ύπαρξη πολλαπλών επιπέδων (Crane, 1965) (Minsky & Papert, 1969). Παρόλο που τα *MADALINE*, ήταν ένα είδος πολυεπίπεδο Τ.Ν.Δ., η εκπαίδευση τους ήταν αργή και όχι αρκετά αποδοτική για την γενική εφαρμογή σε προβλήματα πολυεπίπεδων Τ.Ν.Δ. (Εικόνα 21), ενώ η μικρή υπολογιστική ισχύ των διαθέσιμων τερματικών την δεκαετία του 60', είχε περιορισμούς στην ανάπτυξη άλλων μεθόδων εκπαίδευσης Τ.Ν.Δ. με πολλαπλά επίπεδα. Έτσι οι εφαρμογές των νευρωνικών δικτύων παρέμειναν περιορισμένες, κάτι που αναπόφευκτα προκάλεσε μείωση της χρηματοδότησης για περαιτέρω ερευνα και ως εκ τούτου πάγωμα της ερευνάς πάνω στα Τ.Ν.Δ. (Graupe, 2013).



Εικόνα 21: Ένα πολυεπίπεδο Τ.Ν.Δ. με N_i μεταβλητές εισόδου, N_o μεταβλητές εξόδου και l επίπεδα (Picton, 1994)

Στις αρχές τις επόμενης δεκαετίας, στην διδακτορική του εργασία του *P. Werbos* προχώρησε στην γενίκευση του κανόνα των *Widrow & Hoff* αναπτύσσοντας έναν αλγόριθμο οπισθοδιάδοσης (*back propagation training algorithm*) (Werbos, 1974). Όμως εξαιτίας της μείωσης του ενδιαφέροντος στον συγκεκριμένο τομέα, η δουλειά του παρέμεινε σχετικά άγνωστη. Στα μέσα της δεκαετίας του 80' και με την αύξηση της υπολογιστικής ισχύς των τερματικών, υπήρξε εκ νέου ενδιαφέρον στην χρήση των νευρωνικών δικτύων, κάτι που οδήγησε στην ξανά-εύρεση του αλγόριθμου οπισθοδιάδοσης, σχεδόν ταυτόχρονα από τρεις διαφορετικές ερευνητικές ομάδες. και χρειάστηκε να ξανα-ευρεθεί σχεδόν μια δεκαετία αργότερα από τρεις διαφορετικές ομάδες, όπου παρουσίασαν παρόμοιες μεθόδους εκπαίδευσης Τ.Ν.Δ. με πολλαπλά επίπεδα (Rumelhart, Hinton, & Williams, 1985; Le Cun, 1986; Parker, 1985), η δημοσίευση της πρώτης ομάδας ήταν αυτή που έγινε ευρέως γνωστή και ουσιαστικά έδωσε την ώθηση που χρειαζόταν για την περαιτέρω ανάπτυξη των Τ.Ν.Δ.. Η γενικευμένη μορφή του κανόνα *Delta* που παρουσίασαν βασίζετε στον υπολογισμό της αναστροφής διάδοσης του σφάλματος.

Η μέθοδος της οπισθοδιάδοσης, έχει δύο φάσεις. Κατά την πρώτη φάση υπολογίζονται οι τιμές εξόδου του κάθε νευρώνα, με χρήση των βαρών που είχαν υπολογιστεί από την προηγούμενη εποχή (ή τα αρχικά βάρη αν είναι η πρώτη εποχή). Ενώ στην διάρκεια της δεύτερης εποχής η κλίση υπολογίζεται οπισθοδρομικά, δίνοντας την δυνατότητα να γίνουν οι απαραίτητοι υπολογισμοί του σφάλματος με χρήση του κανόνα της αλυσίδας, κάτι που οδηγεί στην επανεκτίμηση των τιμών των βαρών των συνάψεων. Με αυτόν τον τρόπο η διαφοροποίηση των βαρών ξεκινά να υπολογίζεται από το τελευταίο επίπεδο, και με την γνώση του σφάλματος του κάθε επιπέδου, υπολογίζεται το σφάλμα και τα νέα βάρη για το επόμενο. Η γενική εξίσωση υπολογισμού των βαρών είναι η

$$\Delta_p w_{jk} = \mu \cdot \varepsilon_k^p \cdot y_j^p \quad (33)$$

Όπου w_{jk} , είναι το βάρος της σύνταξης μεταξύ των νευρώνων j του επιπέδου l και k του επιπέδου $l-1$. Γνωρίζοντας ότι το σφάλμα στο τελευταίο επίπεδο δίνεται από την σχέση

$$\varepsilon_o^p = (d_o^p - y_o^p) f'(s_o^p) \quad (34)$$

Εάν δεν είναι ο τελευταίος κόμβος αλλά ο κόμβος του προτελευταίου επιπέδου, η εξίσωση τροποποιείται σε:

$$\varepsilon_{o+1}^p = f'(s_{o+1}^p) \cdot \sum_{o=1}^{N_o} \varepsilon_o^p \cdot w_{o+1,o} \quad (35)$$

Όπου N_o το σύνολο των μεταβλητών εξόδου.

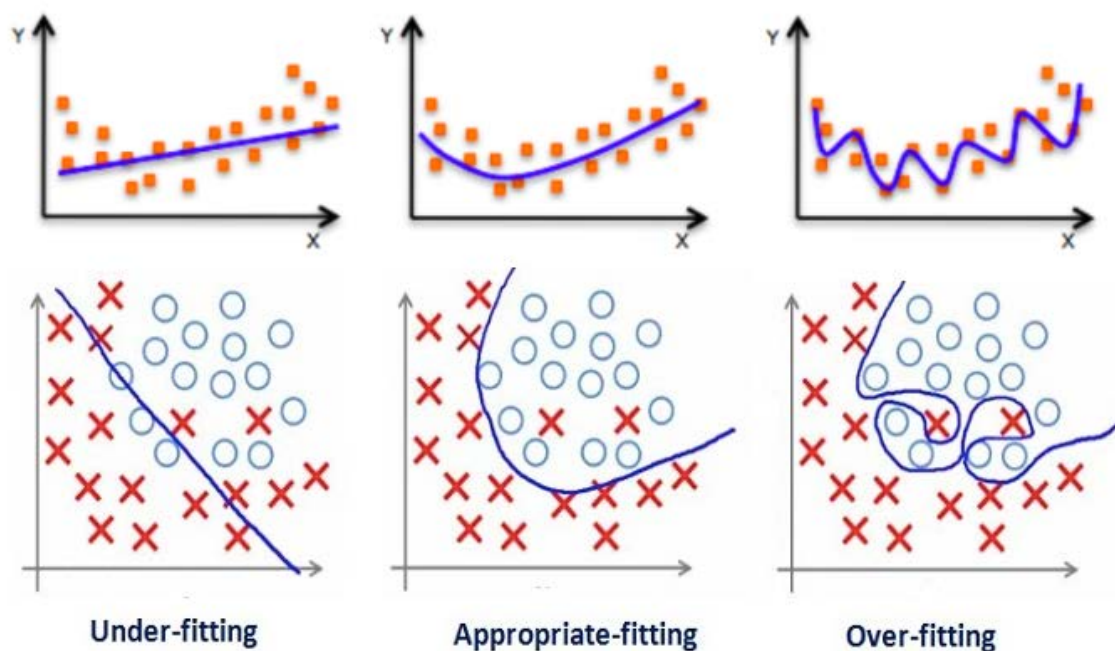
Η παραπάνω εξίσωση μπορεί να γενικευτεί για τυχαίο επίπεδο h , όπου το σφάλμα του θα υπολογίζεται με βάση το προηγούμενο επίπεδο $h-1$

$$\varepsilon_h^p = f'(s_h^p) \cdot \sum_{(h-1)=1}^{N_{(h-1)}} \varepsilon_{h-1}^p \cdot w_{h,h-1} \quad (36)$$

Όπου $N_{(h-1)}$ είναι το σύνολο νευρώνων στο επόμενο επίπεδο (Picton, 1994).

ΓΕΝΙΚΕΥΣΗ ΤΟΥ ΣΦΑΛΜΑΤΟΣ

Η αρχή της επιβλεπόμενης μάθησης είναι ότι με χρήση γνωστών διανυσμάτων εισόδου και των αντίστοιχων διανυσμάτων εξόδου, να υπολογιστεί ο πίνακας των βαρών των νευρώνων, έτσι ώστε το σφάλμα εκπαίδευσης να ελαχιστοποιείται. Όμως αυτό που είναι επιθυμητό είναι η γενίκευση της λειτουργίας των Τ.Ν.Δ., ώστε να μπορούν να δίνουν αποτελέσματα σε διανύσματα εισόδου, στα οποία δεν υπάρχει γνωστό διάνυσμα εξόδου, με μικρό σφάλμα (Prechelt, 1998). Για το λόγο αυτό, το σύνολο των δεδομένων χωρίζεται σε δύο υποσύνολα, το πρώτο χρησιμοποιείται για την εκπαίδευση του νευρωνικού και ονομάζεται υποσύνολο εκπαίδευσης (*training set*), ενώ το δεύτερο χρησιμοποιείται για να ελεγχθεί κατά πόσο το Τ.Ν.Δ. μπορεί να προβλέψει άγνωστες καταστάσεις και ονομάζεται υποσύνολο ελέγχου (*test set*). Καθώς το υποσύνολο ελέγχου δεν χρησιμοποιείται στους υπολογισμούς, το σφάλμα που παρουσιάζει, ταυτίζεται ή είναι αρκετά κοντά στο πραγματικό σφάλμα του συστήματος (Thaler & Furrer, 2005; Russel & Norvig, 2020).

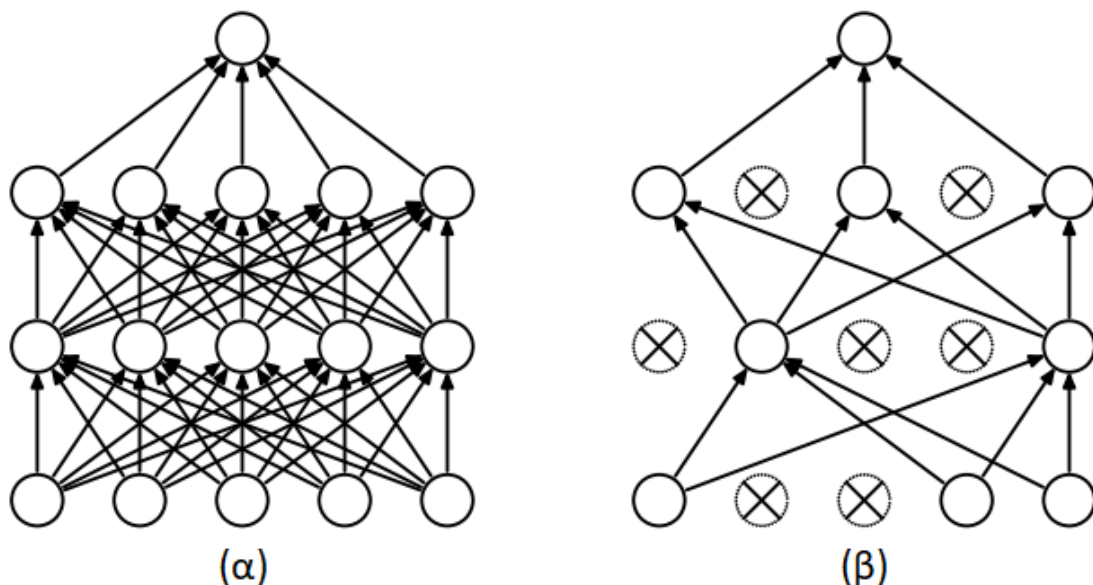


Εικόνα 22: Κλασικά παραδείγματα στα οποία παρατηρείται το φαινόμενο της υπερεκπαίδευσης.

Στην πρώτη γραμμή είναι ένα πρόβλημα στο οποίο αναζητάτε η αντιπροσωπευτική συνάρτηση του προβλήματος. Στην πρώτη περίπτωση η γραμμική απεικόνιση, δεν είναι αντιπροσωπευτική όλων των αποτελεσμάτων και παρουσιάζει πολύ μεγάλες αποκλίσεις από τα πειραματικά δεδομένα, σε αυτή την περίπτωση η ευθεία γραμμή υποπροσαρμόζει τα δεδομένα. Αντιθέτως, στην τρίτη περίπτωση, η χρήση μιας πολύπλοκης απεικόνισης με κάποιο πολυώνυμο μεγάλου βαθμού, μπορεί να παρουσιάζει μικρές αποκλίσεις από τις μετρήσιμες τιμές, αλλά υπερπροσαρμόζοντας τα δεδομένα, αστοχεί να προβάλει μια αντιπροσωπευτική τάση της συνάρτησης, όπως γίνεται στην δεύτερη περίπτωση. Αντίστοιχα παρουσιάζονται παραδείγματα σε πρόβλημα κατηγοριοποίησης παρουσιάζονται στην δεύτερη γραμμή (Russel & Norvig, 2020)

Τα νευρωνικά δίκτυα εξ ορισμού, είναι «μαύρα κουτιά», οι τιμές που λαμβάνουν τα βάρη στους νευρώνες δεν έχουν κάποια φυσική σημασία και απλά είναι κάποιες τιμές που ελαχιστοποιούν τις συγκεκριμένες συναρτήσεις κόστους, μέσω μιας διαδικασίας μάθησης. Ακριβώς για αυτό το λόγο υπάρχει ο κίνδυνος ένα Τ.Ν.Δ. ενώ το σφάλμα εκπαίδευσης να δείχνει να μειώνεται, η ανταπόκριση του μοντέλου σε τιμές που δεν έχουν χρησιμοποιηθεί να γίνεται χειρότερη. Αυτό οφείλεται στο γεγονός ότι τα πειραματικά δεδομένα παρουσιάζουν κάποιο «θόρυβο», μια μικρή απόκλιση από την ιδεατή θέση στην οποία θα έπρεπε να είναι.

Καθώς το Τ.Ν.Δ προσπαθεί να βρει τις βέλτιστες λύσεις για της συγκεκριμένες εξισώσεις και τα συγκεκριμένα ζεύγη δεδομένων, προσπαθεί να μάθει να ενσωματώνει και αυτόν τον «θόρυβο» με αποτέλεσμα εντέλει να αποκλίνει από την ιδεατή κατάσταση, δημιουργώντας λύσεις που περιλαμβάνουν αυτές τις μικρές αστοχίες και όσο περισσότερο μειώνεται το σφάλμα για τις συγκεκριμένες λύσεις, να μεγαλώνει η απόκλιση από την ιδεατή κατάσταση του συστήματος, άρα να μεγαλώνει και το γενικό σφάλμα του μοντέλου (Prechelt, 1998). Το φαινόμενο αυτό ονομάζεται υπερεκπαίδευση (*overtraining*) ή υπερπροσαρμογή (*overfitting*) (Εικόνα 22). Η υπερεκπαίδευση μπορεί να προκληθεί από διάφορες αιτίες. Οι τρεις πιο συνηθισμένες περιπτώσεις είναι η πολυπλοκότητα του μοντέλου, η έλλειψη δεδομένων και η χρήση υπερβολικού αριθμού μεταβλητών εισόδου. Όπως φαίνεται στην Εικόνα 22, η επιλογή μιας πολυωνυμικής εξίσωσης μεγάλου βαθμού, μπορεί μεν να έχει μικρότερο σφάλμα, αλλά παύει να είναι αντιπροσωπευτική του προβλήματος. Αντίστοιχα στα Τ.Ν.Δ. αυξάνοντας σημαντικά τον αριθμό των κόμβων ενός επιπέδου ή τον αριθμό των επιπέδων, μπορεί να οδηγήσει σε υπερεκπαίδευση. Ανάλογα των συνάψεων και των συνολικών κόμβων που υπάρχουν σε ένα σύστημα, απαιτείται και το αντίστοιχο σύνολο δεδομένων. Τα πολύπλοκα Τ.Ν.Δ. έχουν μεγάλες δυνατότητες προσομοίωσης, όμως αν τα δεδομένα δεν επαρκούν για την ορθή εκπαίδευση τους, η γενίκευση της λύσης δεν θα είναι δυνατή. Γενικά υπάρχει ο «κανόνας του θείου Bernie»¹, με βάση τον οποίο, το σύνολο των συνάψεων και των κόμβων πρέπει να είναι λιγότερο από το 10% των διαθέσιμων δεδομένων για εκπαίδευση (Widrow, 1987). Τέλος, υπερεκπαίδευση μπορεί να συμβεί και όταν οι μεταβλητές εισόδου είναι πολύ περισσότερες από αυτές που απαιτεί το πρόβλημα, καθώς ακόμα και αν μια μεταβλητή δεν έχει άμεση φυσική σχέση με την μεταβλητή εξόδου, το νευρωνικό δίκτυο θα βρει κάποια σχέση (έστω με μεγάλο σφάλμα) που θα την προσθέσει στο τελικό αποτέλεσμα, οδηγώντας σε απόκλιση από την ιδεατή κατάσταση. Για αυτό είναι κρίσιμη η σωστή επιλογή των παραμέτρων οι οποίες χρησιμοποιούνται για την εκπαίδευση του Τ.Ν.Δ (Russel & Norvig, 2020).



Εικόνα 23: Εφαρμογή της «εγκατάλειψης», στην εικόνα (α) εμφανίζεται ένα Τ.Ν.Δ. πριν την χρήση της εγκατάλειψης και στη (β) εμφανίζεται η αρχιτεκτονική του, μετά την αφαίρεση των κόμβων και των μεταβλητών εισόδου που δημιουργούσαν υπερεκπαίδευση (Srivastava, et al., 2014).

¹ Ο κανόνας του θείου Bernie (Uncle's Bernie Rule) από το μικρό όνομα του Widrow, Bernard, που τον πρότεινε το 1987 (Bourlard & Morgan, 1994).

Κατά καιρούς έχουν προταθεί διάφορες μέθοδοι για να αντιμετωπιστεί το πρόβλημα της υπερεκπαίδευσης, αλλά πλέον τρεις είναι αυτές που χρησιμοποιούνται ευρέως. Η εγκατάλειψη (*dropout*), η κανονικοποίηση (*regularization*) και η μέθοδος του προώρου τερματισμού (*early stopping*). Η **εγκατάλειψη**, είναι η πιο πρόσφατη από αυτές, η μέθοδος αυτή ξεκινάει με πολύ σύνθετες μορφές δικτύων με παρά πολλούς κόμβους, αρκετά επίπεδα καθώς και πλειάδα μεταβλητών εισόδου (Εικόνα 23). Κατά την διάρκεια της εκπαίδευσης, αφαιρούνται κάποιοι κόμβοι ή και κάποιες μεταβλητές εισόδου, καθώς γίνεται έλεγχος όλων των δυνατών συνδυασμών και επιλέγεται ο συνδυασμός με το καλύτερο αποτέλεσμα (Srivastava, et al., 2014).

Μια άλλη μέθοδος για την αποφυγή της υπερεκπαίδευσης είναι η **κανονικοποίηση**. Ο στόχος της, είναι να περιορίσει την λύση του προβλήματος ανακατασκευής υπερεπιφάνειας σε συμπαγή υποσύνολα, ελαχιστοποιώντας την επαυξημένη συνάρτηση κόστους (Haykin, 2009). Στην μηχανική μάθηση, χρησιμοποιούνται τρεις κατηγορίες κανονικοποίησης, η κανονικοποίηση L1 (γνωστή και ως κανονικοποίηση *Lasso*), η κανονικοποίηση L2 (γνωστή και ως κανονικοποίηση *Ridge*) καθώς και ο συνδυασμός αυτών των δύο. Το όνομα τους προκύπτει από την χρήση των νορμών L1 και L2 αντίστοιχα. Η γενική μορφή των νορμών είναι:

$$\|\mathbf{W}\|_p = \left(\sum_{i=1}^N |w_i|^p \right)^{\frac{1}{p}} = \left(|w_1|^p + |w_2|^p + \dots + |w_N|^p \right)^{\frac{1}{p}} \quad (37)$$

Επομένως για τις L1 και L2 προκύπτει αντίστοιχα

$$L1 \quad \|\mathbf{W}\|_1 = \left(\sum_{i=1}^N |w_i|^1 \right)^{\frac{1}{1}} = \sum_{i=1}^N |w_i|$$

$$L2 \quad \|\mathbf{W}\|_2 = \left(\sum_{i=1}^N |w_i|^2 \right)^{\frac{1}{2}} = \left(\sum_{i=1}^N w_i^2 \right)^{\frac{1}{2}}$$

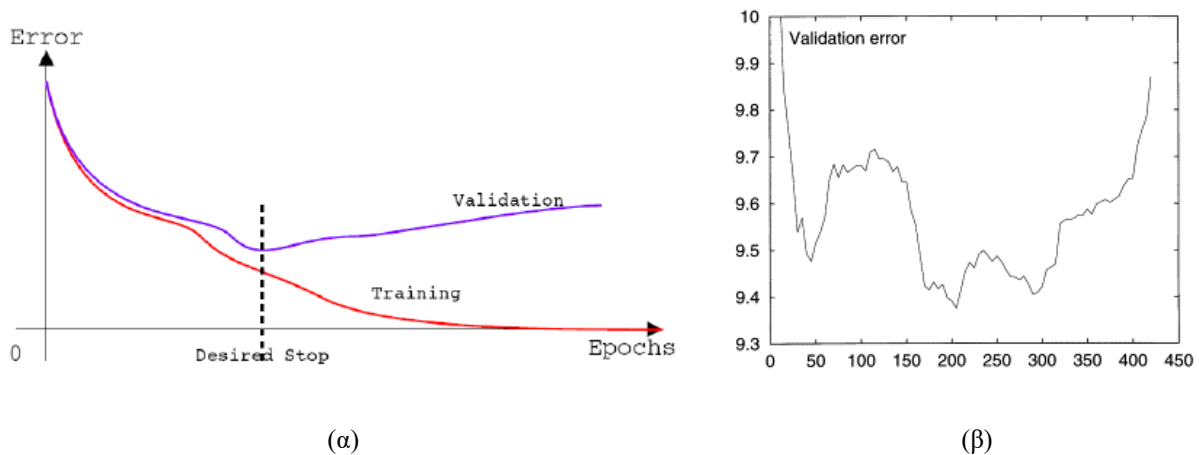
Και οι δύο προσθέτουν έναν επιπλέον όρο, ως ποινή, στην συνάρτηση κόστους που περιγράφηκε στην **εξίσωση 27**. Επομένως προκύπτουν αντίστοιχα

$$K(\mathbf{W}^{(i)}) = K(\mathbf{W}^{(i-1)}) + \mu_i \cdot \nabla K(\mathbf{W}^{(i-1)})^T \cdot \Delta \mathbf{W}^{(i)} + \lambda \cdot \sum_{i=1}^N |w_i| \quad (38)$$

$$K(\mathbf{W}^{(i)}) = K(\mathbf{W}^{(i-1)}) + \mu_i \cdot \nabla K(\mathbf{W}^{(i-1)})^T \cdot \Delta \mathbf{W}^{(i)} + \lambda \cdot \left(\sum_{i=1}^N w_i^2 \right) \quad (39)$$

Όπου λ ($\lambda \geq 0$) είναι μια νέα υπερπαράμετρος που καλείται παράμετρος κανονικοποίησης και εκφράζει το μέγεθος της ποινής. Πολύ μεγάλες τιμές του λ , οδηγούν τα μοντέλα σε υποπροσαρμογή, ενώ όταν το λ ισούται με μηδέν, δεν υπάρχει κανονικοποίηση. Με την μείωση των απόλυτων τιμών των συντελεστών w , η κανονικοποίηση L1 κάνει αντίστοιχη δουλειά με την εγκατάλειψη, ωθεί το σύστημα στην μείωση των συνάψεων. Ενώ η L2, με την μείωση του τετράγωνου των συντελεστών w , δεν μειώνει τον αριθμό των συνάψεων, όσο την επίδραση που έχει κάθε στοιχείο στο μοντέλο καθώς μειώνει τις τιμές των εκάστοτε συντελεστών, για αυτό είναι γνωστή και ως «φθορά των βαρών» (*weight decay*) (Hastie, Tibshirani, & Friedman, 2008).

Η πιο ευρέως διαδεδομένη μέθοδος για την αποφυγή της υπερεκπαίδευσης είναι ο **πρόωρος τερματισμός**. Η αρχή λειτουργίας της συγκεκριμένης μεθόδου είναι ότι το σύνολο των δεδομένων χωρίζεται σε τρία υποσύνολα, αντί για δύο, με το τρίτο να ονομάζεται υποσύνολο επικύρωσης (*validation set*). Το συγκεκριμένο κριτήριο δεν χρησιμοποιείται για την εκπαίδευση, αλλά για τον έλεγχο κατά πόσο το δίκτυο ανταποκρίνεται σε τιμές που δεν χρησιμοποιούνται για την εκπαίδευση του. Διαφέρει από το υποσύνολο ελέγχου (*test set*), το οποίο δεν χρησιμοποιείται καθόλου στους υπολογισμούς. Η εκπαίδευση του δικτύου γίνεται κανονικά με βάση το υποσύνολο εκπαίδευσης και όταν το σφάλμα που υπολογίζεται για το υποσύνολο επικύρωσης μεγαλώνει την ώρα που το σφάλμα της εκπαίδευσης μειώνεται, θεωρείται ότι το Τ.Ν.Δ. αρχίζει να προσαρμόζεται στο θόρυβο, αντί στο σήμα, επομένως το Τ.Ν.Δ. οδηγείτε σε υπερεκπαίδευση (Prechelt, 1998) (Piotrowski & Napiorkowski, 2013).

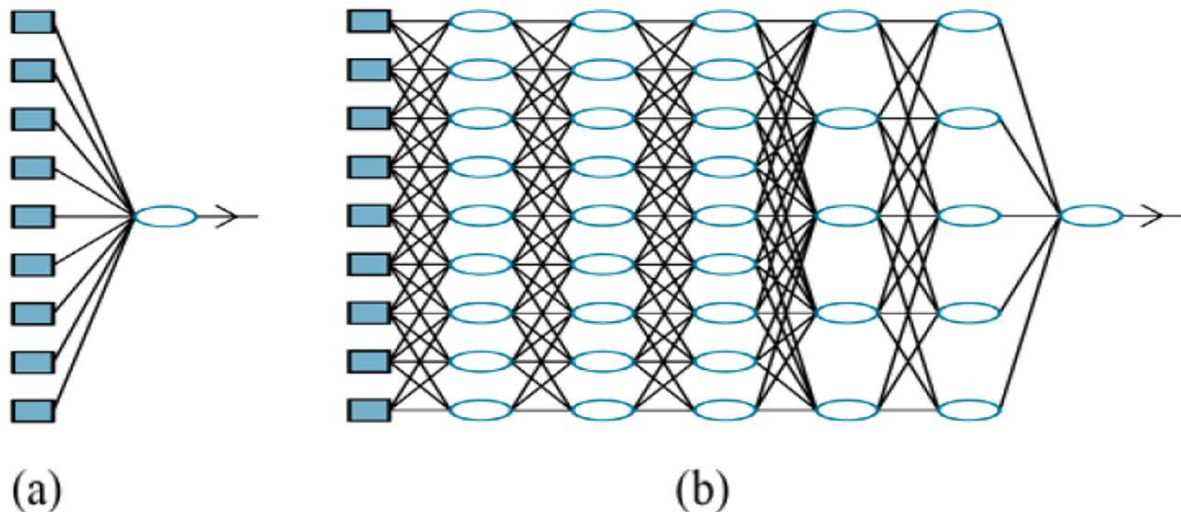


Εικόνα 24: α) Το κριτήριο του πρόωρου τερματισμού (Arvetti, Gini, & Folgheraiter, 2007)
β) Η γραφική παράσταση του σφάλματος επικύρωσης (Prechelt, 1998)

Η παραπάνω εικόνα είναι μια ιδεατή κατάσταση και σπανία μπορεί να βρεθεί σε πραγματικά προβλήματα, καθώς η γραφική συνάρτηση του σφάλματος του υποσυνόλου της επικύρωσης ενδέχεται να παρουσιάζει παραπάνω από ένα ακρότατα (Εικόνα 24β). Ο άμεσος τερματισμός της εκπαίδευσης δεν συνίσταται, καθώς κάτι τέτοιο θα οδηγούσε σε υποπροσαρμογή. Για αυτό το λόγο επιλέγεται συνήθως ένας συγκεκριμένος αριθμός εποχών ως περιθώριο, στις οποίες θα πρέπει να αυξάνεται συνεχόμενα το σφάλμα στα δεδομένα επικύρωσης, ώστε να διακοπή η διαδικασία εκπαίδευσης του νευρωνικού. Στην συνέχεια, το σύστημα επιστρέφει στις παραμέτρους που αντιστοιχούσαν στο μικρότερο σφάλμα των δεδομένων επαλήθευσης. Με αυτόν τον τρόπο αποφεύγονται αρκετά τοπικά ακρότατα. Ενδεχομένως όμως και πάλι η εκπαίδευση να μην φτάσει ποτέ σε κάποιο ολικό ακρότατο, αν μεσολαβούν πολλές εποχές. Δυστυχώς δεν υπάρχει κάποια γενικευμένη μορφή για την γραφική του σφάλματος της επικύρωσης, καθώς σε διαφορετικά προβλήματα θα υπάρχουν διαφορετικές καμπύλες, με διαφορετικά ελάχιστα. Το μόνο που παρουσιάζεται κοινό σε όλες τις καμπύλες σφάλματος, είναι ότι τα τοπικά ακρότατα, μεταξύ τους δεν παρουσιάζουν μεγάλες διαφορές (Prechelt, 1998).

ΒΑΘΙΑ ΜΑΘΗΣΗ

Η βαθιά μάθηση (*Deep Learning*) είναι μια ευρεία οικογένεια τεχνικών για τη μηχανική μάθηση στην οποία οι υποθέσεις έχουν την μορφή σύνθετων αλγεβρικών κυκλωμάτων με ρυθμιζόμενα βάρη συνδέσεων. Η λέξη βαθιά, αναφέρεται στο γεγονός ότι τα συστήματα αυτά, οργανώνονται σε πολλά επίπεδα με πολλά κρυφά στρώματα, κάτι που μεταφράζεται στο γεγονός ότι οι διαδρομές υπολογισμού από εισόδους σε εξόδους, έχουν πολλά βήματα. Η βαθιά μάθηση είναι αυτή τη στιγμή η πιο ευρέως χρησιμοποιούμενη προσέγγιση για εφαρμογές όπως: η οπτική αναγνώριση αντικειμένων, η μηχανική μετάφραση, η αναγνώριση ομιλίας, η σύνθεση ομιλίας και η σύνθεση εικόνας (Russel & Norvig, 2020).



Εικόνα 25 α) Ένα απλό (ρηχό) νευρωνικό δίκτυο, όπως αυτό που περιγράφηκε από τους McCulloch-Pitts
β) Ένα βαθύ νευρωνικό δίκτυο, σε σχέση με το ρηχό, στο βαθύ T.N.Δ. υπάρχουν μακρύτερα υπολογιστικά μονοπάτια που επιτρέπουν την αλληλεπίδραση μεταξύ των μεταβλητών (Russel & Norvig, 2020)

Νευρωνικά δίκτυα όπως τα ADALINE, τα MADALINE ή τα PERCEPTRON, παρόλο που ανταποκρίθηκαν σε διάφορα προβλήματα γραμμικής και λογιστικής παλινδρόμησης και είχαν την δυνατότητα να χειριστούν αρκετές μεταβλητές εισόδου, είχαν πολύ μικρές διαδρομές υπολογισμού (Εικόνα 25α). Κάθε τιμή εισόδου πολλαπλασιαζόταν με ένα βάρος, στην συνέχεια τα αποτελέσματα αθροίζονταν και πρόκυπτε η τιμή εξόδου. Η συμβολή της κάθε μεταβλητής στο αποτέλεσμα ήταν ανεξάρτητη χωρίς να υπάρχει αλληλεπίδραση με τις υπόλοιπες. Αυτό περιορίζει σημαντικά τις δυνατότητες αυτών των μοντέλων, περιορίζοντας τα σε απλά προβλήματα ή σε εικονικές καταστάσεις (Russel & Norvig, 2020). Τα ρηχά νευρωνικά δίκτυα (T.N.Δ. με ένα κρυφό επίπεδο) παρόλο που έχουν κάποια πλεονεκτήματα, όπως η δυνατότητα χρήσης κυρτών συναρτήσεων κόστους, είναι και αυτά απλές δομές με συγκεκριμένες αδυναμίες και περιορισμούς στην αποτελεσματικότητα αναπαράστασης ορισμένων προβλημάτων, όπως η δυνατότητα απεικόνισης συνθετικών συναρτήσεων. Παρόλο που ορισμένα θεωρήματα δείχνουν ότι ορισμένες ρηχές αρχιτεκτονικές μπορούν να προσεγγίσουν οποιαδήποτε συνάρτηση με αυθαίρετη ακρίβεια, δεν υπάρχει κάποια εγγύηση ως προς την αποτελεσματικότητα της αναπαράστασης. Αντίθετα, οι βαθιές αρχιτεκτονικές μπορούν, να αντιπροσωπεύουν αποτελεσματικότερα (και με καλύτερες ιδιότητες κλιμάκωσης) ορισμένες οικογένειες συναρτήσεων, καθώς η αύξηση των κρυφών στρωμάτων οδηγεί σε αύξηση της εκφραστικότητας και της δύναμης του μοντέλου (Εικόνα 25β). Για το λόγο αυτό. Τα βαθιά T.N.Δ., μπορούν να χρησιμοποιηθούν για να περιγράψουν ακόμα και τις πιο περίπλοκες έννοιες του πραγματικού κόσμου (Bengio & LeCun, 2007).

ΠΡΟΗΓΜΕΝΟΙ ΜΕΘΟΔΟΙ ΣΥΓΚΛΙΣΗΣ

Η μέθοδος της απότομης καθόδου δρα στη βάση της γραμμικής προσέγγισης της συνάρτησης κόστους. Για να το κάνει αυτό, βασίζεται στο διάνυσμα της κλίσης ως την μοναδική πηγή μιας πληροφορίας 1^{ης} τάξης της επιφάνειας του σφάλματος. Αυτό όμως έρχεται με το μειονέκτημα της αργής σύγκλισης, κάτι που είναι ανασταλτικός παράγοντας της χρήσης πολύπλοκων αρχιτεκτονικών δομών. Η ενσωμάτωση της ορμής, είναι μια απόπειρα χρήσης πληροφοριών 2^{ης} τάξης για την επιφάνεια του σφάλματος, η οποία βελτιώνει τις επιδόσεις της μεθόδου. Ωστόσο, η χρήση της ορμής, κάνει την διαδικασία εκπαίδευσης πιο ευαίσθητη στην διαχείριση της, προσθέτοντας μια επιπλέον υπερπαράμετρο. Για το λόγο αυτό προτάθηκε η χρήση μεθόδων μεγαλύτερης τάξης στην διαδικασία της εκπαίδευσης (Haykin, 2009).

Μέθοδος Newton

Η συνάρτηση κόστους $K(\mathbf{W})$, αν χρησιμοποιηθεί η σειρά *Taylor* μέχρι την 2^η τάξη διαμορφώνεται ως εξής:

$$K(\mathbf{W}^{(i)} + \mathbf{p}) = K(\mathbf{W}^{(i-1)}) + \mathbf{g}^T \cdot \mathbf{p} + \frac{1}{2} \cdot \mathbf{p}^T \cdot \mathbf{H} \cdot \mathbf{p} \quad (40)$$

Όπου

$$\mathbf{g} = \nabla K(\mathbf{W}^{i-1}) = \left. \frac{\partial K(\mathbf{W})}{\partial \mathbf{W}} \right|_{\mathbf{W}=\mathbf{W}^{(i-1)}} \quad \text{είναι η τοπική βαθμίδα κλίσης}$$

$$\mathbf{H} = \nabla^2 K(\mathbf{W}^{i-1}) = \left. \frac{\partial^2 K(\mathbf{W})}{\partial \mathbf{W}^2} \right|_{\mathbf{W}=\mathbf{W}^{(i-1)}} \quad \text{είναι ο πίνακας Hessian, που αντιπροσωπεύει την καμπυλότητα της επιφάνειας σφάλματος}$$

$$\mathbf{p} = \Delta \mathbf{W}^{(i)}$$

Το ελάχιστο της εξίσωσης 40, θα επιτυγχάνεται όταν ελαχιστοποιείτε η τετραγωνική συνάρτηση

$$\Phi(\mathbf{p}) = \mathbf{g}^T \cdot \mathbf{p} + \frac{1}{2} \cdot \mathbf{p}^T \cdot \mathbf{H} \cdot \mathbf{p} \quad (41)$$

Κάτι που συμβαίνει όταν

$$\mathbf{p}^* = -\mathbf{H}^{-1} \cdot \mathbf{g} \quad (42)$$

Όπου \mathbf{H}^{-1} είναι ο αντίστροφος πίνακας του Hessian, υποθέτοντας ότι αυτός υπάρχει. Ένας αλγόριθμος ελαχιστοποίησης, που το \mathbf{p}^* προκύπτει από υπολογισμό της εξίσωσης 42 χαρακτηρίζεται ως μέθοδος Newton και το \mathbf{p}^* καλείται κατεύθυνση *Newton* (Gill, Murray, & Wright, 1981). Εάν ο πίνακας \mathbf{H} υπάρχει και είναι θετικά ορισμένος, αρκεί μια επανάληψη για να βρεθεί το ελάχιστο. Αλλά τις περισσότερες φορές ειδικά σε προβλήματα που περιλαμβάνουν κρυφά επίπεδα, ο πίνακας \mathbf{H} δεν είναι θετικά ορισμένος με αποτέλεσμα να είναι αδύνατη η εκπαίδευση ενός T.N.Δ. με χρήση αυτής της μεθόδου (Battiti, 1992).

Μέθοδος Quasi-Newton

Η σειρά Taylor μπορεί να χρησιμοποιηθεί και στον υπολογισμό της παραγώγου. Έτσι

$$\mathbf{g}\left(\mathbf{W}^{(i)}\right)=\mathbf{g}\left(\mathbf{W}^{(i-1)}+\mathbf{p}\right)=\mathbf{g}\left(\mathbf{W}^{(i-1)}\right)+\mathbf{H}^T \cdot \mathbf{p}+\ldots \quad (43)$$

Η καμπυλότητα της συνάρτησης K στην περιοχή του \mathbf{p} μπορεί να προσεγγιστεί χρησιμοποιώντας πρώτης τάξης σειρά *Taylor*, επομένως

$$\mathbf{p}^T \cdot \mathbf{H} \cdot \mathbf{p} \approx \mathbf{p}^T \cdot \mathbf{B} \cdot \mathbf{p}=\left(\mathbf{g}\left(\mathbf{W}^{(i-1)}+\mathbf{p}\right)-\mathbf{g}\left(\mathbf{W}^{(i-1)}\right)\right)^T \cdot \mathbf{p} \quad (44)$$

Όπου ο πίνακας \mathbf{B} είναι μια προσέγγιση του πίνακα *Hessian*. Με την χρήση αυτού του πίνακα, αντί για τον πίνακα *Hessian*, η [εξίσωση 42](#) μετατρέπεται σε

$$\mathbf{p}^*=-\mathbf{B}^{-1} \cdot \mathbf{g} \quad (45)$$

Οι μέθοδοι που χρησιμοποιούν την παραπάνω προσέγγιση, καλούνται μέθοδοι *Quasi-Newton*. Ως αρχική προσέγγιση $\mathbf{B}^{(0)}$ του πίνακα *Hessian*, συνήθως λαμβάνεται ο ταυτοτικός πίνακας (εάν δεν είναι διαθέσιμα περισσότερα στοιχεία). Με την επιλογή του ταυτοτικού πίνακα ως αρχική συνθήκη, ο αλγόριθμος την πρώτη εποχή ουσιαστικά ταυτίζεται με τον αλγόριθμο της απότομης καθόδου ([Gill, Murray, & Wright, 1981](#)).

Η πιο διαδεδομένη σχέση για την ενημέρωση των πινάκων \mathbf{B} , είναι η σχέση *Broyden-Fletcher-Goldfarb-Shanno (BFGS)* ([Fletcher, 1987](#))

$$\mathbf{B}^{(i)}=\mathbf{B}^{(i-1)}+\frac{\mathbf{y}^{(i-1)} \cdot\left(\mathbf{y}^{(i-1)}\right)^T}{\left(\mathbf{y}^{(i-1)}\right)^T \cdot \mathbf{p}^{(i-1)}}-\frac{\mathbf{B}^{(i-1)} \cdot \mathbf{p}^{(i-1)} \cdot\left(\mathbf{p}^{(i-1)}\right)^T \cdot\left(\mathbf{B}^{(i-1)}\right)^T}{\left(\mathbf{p}^{(i-1)}\right)^T \cdot \mathbf{B}^{(i-1)} \cdot \mathbf{p}^{(i-1)}} \quad (46)$$

$$\text{Όπου } \mathbf{y}^{(i-1)}=\nabla K\left(\mathbf{W}^{(i)}\right)-\nabla K\left(\mathbf{W}^{(i-1)}\right) \Rightarrow \mathbf{y}^{(i-1)}=\mathbf{B}^{(i)} \cdot \mathbf{p}^{(i-1)}$$

Μέθοδος One-Step Secant (OSS)

Η μέθοδος αυτή είναι μια μίξη της μεθόδου *Newton* και της μεθόδου συζυγούς κλίσης που θα αναφερθεί στην συνέχεια. Η *OSS*, ψάχνει την νέα διεύθυνση με την χρήση της εξίσωσης

$$\mathbf{p}^{(i)}=-\mathbf{g}^{(i-1)}+\mathbf{A}^{(i-1)} \cdot \mathbf{p}^{(i-1)}+\mathbf{B}^{(i-1)} \cdot \mathbf{y}^{(i-1)} \quad (47)$$

$$\text{Όπου } \mathbf{A}^{(i-1)}=-\left(1+\frac{\left(\mathbf{y}^{(i-1)}\right)^T \cdot \mathbf{y}^{(i-1)}}{\left(\mathbf{p}^{(i-1)}\right)^T \cdot \mathbf{y}^{(i-1)}}\right) \cdot \mathbf{B}^{(i-1)}+\frac{\left(\mathbf{y}^{(i-1)}\right)^T \cdot \mathbf{g}^{(i-1)}}{\left(\mathbf{p}^{(i-1)}\right)^T \cdot \mathbf{y}^{(i-1)}} \quad \text{και} \quad \mathbf{B}^{(i-1)}=\frac{\left(\mathbf{p}^{(i-1)}\right)^T \cdot \mathbf{g}^{(i-1)}}{\left(\mathbf{p}^{(i-1)}\right)^T \cdot \mathbf{y}^{(i-1)}}$$

([Battiti, 1992](#))

Μέθοδος Levenberg-Marquardt

Η μέθοδος αυτή είναι μια τροποποίηση της μεθόδου *Newton* και σχεδιάστηκε για να ελαχιστοποιεί συναρτήσεις οι οποίες είναι άθροισμα τετραγώνων άλλων μη γραμμικών συναρτήσεων και για αυτό το λόγο χρησιμοποιείται ευρέως στην εκπαίδευση των Τ.Ν.Δ. που προσπαθούν να μειώσουν το MSE (Hagan, et al., 2014).

Οι *Levenberg* και *Marquardt* (Marquardt, 1963; Levenberg, 1944) πρότειναν έναν πολύ κομψό τρόπο για να βρεθεί το ελάχιστο της συνάρτησης κόστους. Αρχικά παρατήρησαν ότι η συνάρτηση κόστους ήταν το άθροισμα τετραγώνων άλλων συναρτήσεων. Όπως αναφέρθηκε στην [εξίσωση 28](#) η συνάρτηση κόστους ισούται με

$$K(\mathbf{W}) = (d - \mathbf{W}^T \cdot \mathbf{X})^2$$

Θέτοντας ως

$$\mathbf{v}(\mathbf{W}) = d - \mathbf{W}^T \cdot \mathbf{X}$$

Προκύπτει

$$K(\mathbf{W}) = (\mathbf{v}(\mathbf{W}))^2 = (\mathbf{v}(\mathbf{W}))^T \cdot \mathbf{v}(\mathbf{W}) = \|\mathbf{v}(\mathbf{W})\|^2 \quad (48)$$

Την ώρα που τα νέα βάρη θα υπολογίζονται από την σχέση

$$\mathbf{W}^{(i)} = \mathbf{W}^{(i-1)} - \nabla^2 K(\mathbf{W}^{i-1}) \cdot \nabla K(\mathbf{W}^{i-1}) \quad (49)$$

Στην συνέχεια υπολόγισαν την παράγωγο για το στοιχείο j , η οποία θα δίνεται από την σχέση

$$[\nabla K(\mathbf{W})]_j = \frac{\partial K(\mathbf{W})}{\partial w_j} = 2 \cdot \sum_{i=1}^N v_i(\mathbf{W}) \cdot \frac{\partial v_i(\mathbf{W})}{\partial w_j}$$

Η οποία με χρήση της Ιακωβιανής Μήτρας (*Jacobian Matrix*), \mathbf{J}

$$\mathbf{J}(\mathbf{W}) = \begin{bmatrix} \frac{\partial v_1(\mathbf{W})}{\partial w_1} & \frac{\partial v_1(\mathbf{W})}{\partial w_2} & \dots & \frac{\partial v_1(\mathbf{W})}{\partial w_n} \\ \frac{\partial v_2(\mathbf{W})}{\partial w_1} & \frac{\partial v_2(\mathbf{W})}{\partial w_2} & \dots & \frac{\partial v_2(\mathbf{W})}{\partial w_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial v_N(\mathbf{W})}{\partial w_1} & \frac{\partial v_N(\mathbf{W})}{\partial w_2} & \dots & \frac{\partial v_N(\mathbf{W})}{\partial w_n} \end{bmatrix}$$

Μπορεί να γραφεί με διανυσματική μορφή ως

$$\nabla K(\mathbf{W}) = 2 \cdot \mathbf{J}^T(\mathbf{W}) \cdot \mathbf{v}(\mathbf{W}) \quad (50)$$

Αντίστοιχα ο υπολογισμός του k, j στοιχείου του πίνακα *Hessian* θα προκύπτει από την σχέση

$$\left[\nabla^2 K(\mathbf{W}) \right]_{k,j} = \frac{\partial^2 K(\mathbf{W})}{\partial w_k \partial w_j} = 2 \cdot \sum_{i=1}^N \left\{ \frac{\partial v_i(\mathbf{W})}{\partial w_k} \cdot \frac{\partial v_i(\mathbf{W})}{\partial w_j} + v_i(\mathbf{W}) \cdot \frac{\partial^2 v_i(\mathbf{W})}{\partial w_k \partial w_j} \right\}$$

Κάτι που μπορεί να γραφεί με διανυσματική μορφή ως

$$\mathbf{H} = \nabla^2 K(\mathbf{W}) = 2 \cdot \mathbf{J}^T(\mathbf{W}) \cdot \mathbf{J}(\mathbf{W}) + 2 \cdot \mathbf{R}(\mathbf{W}) \quad (51)$$

$$\text{Όπου } \mathbf{R}(\mathbf{W}) = \sum_{i=1}^N v_i(\mathbf{W}) \cdot \nabla^2 v_i(\mathbf{W})$$

Επομένως για μικρές τιμές του $\mathbf{R}(\mathbf{W})$, ο πίνακας *Hessian* θα μπορούσε να προσεγγιστεί ως

$$\mathbf{H} = \nabla^2 K(\mathbf{W}) \simeq 2 \cdot \mathbf{J}^T(\mathbf{W}) \cdot \mathbf{J}(\mathbf{W}) \quad (52)$$

Αντικαθιστώντας τις 49 και 51 στην 48 προκύπτει η μέθοδος *Gauss-Newton*

$$\begin{aligned} \mathbf{W}^{(i)} &= \mathbf{W}^{(i-1)} - \left[2 \cdot \mathbf{J}^T(\mathbf{W}^{(i-1)}) \cdot \mathbf{J}(\mathbf{W}^{(i-1)}) \right]^{-1} \cdot 2 \cdot \mathbf{J}^T(\mathbf{W}^{(i-1)}) \cdot \mathbf{v}(\mathbf{W}^{(i-1)}) \Leftrightarrow \\ \mathbf{W}^{(i)} &= \mathbf{W}^{(i-1)} - \left[\mathbf{J}^T(\mathbf{W}^{(i-1)}) \cdot \mathbf{J}(\mathbf{W}^{(i-1)}) \right]^{-1} \cdot \mathbf{J}^T(\mathbf{W}^{(i-1)}) \cdot \mathbf{v}(\mathbf{W}^{(i-1)}) \end{aligned} \quad (53)$$

Το μεγαλύτερο πλεονέκτημα αυτής της μεθόδου, είναι ότι δεν είναι απαραίτητος ο υπολογισμός των δεύτερων παραγώγων της συνάρτησης κόστους (Hagan, et al., 2014; Moré, 1978). Ενώ το μεγαλύτερο μειονέκτημα της μεθόδου *Gauss-Newton*, είναι ότι ο πίνακας *Hessian*, μπορεί να μην είναι αναστρέψιμος. Αυτό μπορεί να παρακαμφθεί, χρησιμοποιώντας την παρακάτω προσέγγιση

$$\mathbf{G} = \mathbf{H} + \nu \cdot \mathbf{I} \quad (54)$$

Όπου το ν μια σταθερά και ο \mathbf{I} είναι ο ταυτοτικός πίνακας. Για να είναι ο πίνακας \mathbf{G} αναστρέψιμος θα πρέπει οι ιδιοτιμές του να είναι μη μηδενικές. Έστω ότι οι ιδιοτιμές και τα ιδιοδιανύσματα του πίνακα *Hessian* είναι $\{\lambda_1, \lambda_2, \dots, \lambda_n\}$ και $\{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n\}$ αντίστοιχα. Επομένως

$$\begin{aligned} \mathbf{G} \cdot \mathbf{z}_j &= \mathbf{H} \cdot \mathbf{z}_j + \nu \cdot \mathbf{z}_j \Leftrightarrow \\ \mathbf{G} \cdot \mathbf{z}_j &= (\lambda_j + \nu) \cdot \mathbf{z}_j \end{aligned} \quad (55)$$

Αρά με βάση την εξίσωση 55, εάν η ποσότητα $(\lambda_j + \nu)$ είναι θετική για κάθε j , τότε ο \mathbf{G} θα είναι αναστρέψιμος και θετικά ορισμένος. Κάτι που θα ισχύει για ικανοποιητικά μεγάλο ν , ο \mathbf{G} θα είναι θετικά ορισμένος και αναστρέψιμος. Επομένως η εξίσωση 53 μπορεί να γραφεί ως

$$\Delta \mathbf{W}^{(i)} = - \left[\mathbf{J}^T(\mathbf{W}^{(i-1)}) \cdot \mathbf{J}(\mathbf{W}^{(i-1)}) + \nu_{i-1} \cdot \mathbf{I} \right]^{-1} \cdot \mathbf{J}^T(\mathbf{W}^{(i-1)}) \cdot \mathbf{v}(\mathbf{W}^{(i-1)})$$

Έτσι όσο το ν αυξάνεται, ο αλγόριθμος πλησιάζει την μέθοδο της απότομης καθόδου με μικρό συντελεστή μάθησης, έτσι για μεγάλο ν

$$\mathbf{W}^{(i)} = \mathbf{W}^{(i-1)} - \frac{1}{\nu_{i-1}} \mathbf{J}^T \left(\mathbf{W}^{(i-1)} \right) \cdot \mathbf{v} \left(\mathbf{W}^{(i-1)} \right) = \mathbf{W}^{(i-1)} - \frac{1}{2 \cdot \nu_{i-1}} \nabla K \left(\mathbf{W}^{(i-1)} \right) \quad (56)$$

Ενώ όταν το ν πλησιάζει το μηδέν, η μέθοδος ταυτίζεται με την *Gauss-Newton* (εξίσωση 53).

Η αρχική προσέγγιση του *Levenberg*, προέβλεπε την χρήση μιας μεγάλης τιμής του ν , ώστε το σύστημα να κινείται στην κατεύθυνση της απότομης καθόδου και να έχει πάντα κάποια λύση (*Levenberg, 1944*). Η βελτίωση του αλγορίθμου προέκυψε από τον *Marquardt*, ο οποίος άλλαξε την στρατηγική επιλογής του ν βελτιώνοντας σημαντικά την απόδοση του αλγορίθμου. Αντί να χρησιμοποιήσει μια μεγάλη τιμή, όπως ο *Levenberg*, πρότεινε την αρχική επιλογή μιας σχετικά μικρής τιμής (πχ $\nu = 0.01$). Εάν η επόμενη τιμή δεν είναι μικρότερη του αρχικού σφάλματος, το βήμα επαναλαμβάνεται χρησιμοποιώντας ένα μεγαλύτερο ν . Για να γίνει αυτό, το ν πολλαπλασιάζεται με μία σταθερά ρ , μεγαλύτερη της μονάδας (πχ $\rho = 10$). Μεγαλώνοντας το ν , ο αλγόριθμος αρχίζει να συμπεριφέρεται σαν μέθοδος απότομης καθόδου, επομένως κάποια στιγμή, όταν το βήμα μικρύνει επαρκώς, το σύστημα θα κινηθεί προς τη διεύθυνση της πιο απότομης καθόδου. Στην συνέχεια το ν θα πρέπει να μεγαλώσει, ώστε το σύστημα να αποκτήσει την ταχύτητα της *Gauss-Newton*. Για αυτό και αυτή τη φορά το ν διαιρείται με την σταθερά ρ , ώστε να μικρύνει και να αποκτήσει η μέθοδος γρηγορότερη σύγκλιση (*Marquardt, 1963*). Με αυτόν τον τρόπο, ο *Marquardt*, συνδύασε την ταχύτητα της μεθόδου *Gauss-Newton* και την βεβαιότητα σύγκλισης της μεθόδου της απότομης καθόδου (*Scales, 1985*).

Η μέθοδος *Levenberg-Marquardt*, μπορεί να συνδυαστεί με την *ridge* (κανονικοποίηση L2), ώστε να προσφέρει καλύτερη γενίκευση του σφάλματος. Από την στιγμή που είναι ήδη γνωστή η τιμή του πίνακα *Hessian*, είναι εύκολο να συνδυαστεί με μια μέθοδο **κανονικοποίησης Bayesian** (τροποποίηση της κανονικοποίησης L2). Η νέα συνάρτηση κόστους διαμορφώνεται ως εξής

$$BR(\mathbf{W}) = \alpha \cdot K(\mathbf{W}) + \beta \cdot \sum_{j=1}^N (w_j^2) \quad (57)$$

Όπου $K(\mathbf{W})$, η συνάρτηση κόστους της εξίσωσης 48. Οι συντελεστές α , β λαμβάνουν κάποιες αρχικές τιμές (πχ $\alpha=0$, $\beta=1$) και στην συνέχεια υπολογίζονται από τις σχέσεις

$$\alpha^{(i+1)} = \frac{\gamma^{(i)}}{2 \cdot \left(\sum_{j=1}^N (w_j^2) \right)^{(i)}} \text{ και } \beta^{(i+1)} = \frac{\gamma^{(i)}}{2 \cdot K(W)^{(i)}}$$

Ο συντελεστής γ καλείται αποτελεσματικός αριθμός παραμέτρων, λαμβάνει τιμές από 0 έως N (όπου N ο ολικός αριθμός των παραμέτρων του δικτύου) και υπολογίζεται από την σχέση

$$\gamma^{(i)} = N^{(i)} - 2 \cdot a^{(i)} \cdot \text{trace}(\mathbf{H}^{(i)})^{-1}$$

Όπου $\text{trace}(\mathbf{H})$, είναι το άθροισμα της διαγωνίου, του πίνακα *Hessian* (*Dan Foresee & Hagan, 1997; Burden & Winkler, 2008*).

Μέθοδος Ανθεκτικής Διάδοσης (*RProp*)

Η μέθοδος της ανθεκτικής διάδοσης (*Resilient Propagation* – *RProp*), ακολουθεί μια διαφορετική λογική από τις προηγούμενες μεθόδους. Η ενημέρωση των βαρών γίνεται με χρήση πληροφορίας της τοπικής κλίσης, χωρίς όμως να υπολογίζει την συμπεριφορά της. Ο κανόνας μάθησης που χρησιμοποιείται είναι ο εξής, αρχικά υπολογίζονται οι τιμές μιας νέας παραμέτρου, η οποία καθορίζει το μέγεθος της αλλαγής του βάρους κάθε σύναψης, οι τιμές της για όλες της εποχές πέρα από την πρώτη, προκύπτουν από την εξίσωση

$$\Delta_{k,j}^{(i)} = \begin{cases} \eta^+ \cdot \Delta_{k,j}^{(i-1)} & \frac{\partial K^{(i-1)}(\mathbf{w})}{\partial w_{k,j}} \cdot \frac{\partial K^{(i)}(\mathbf{w})}{\partial w_{k,j}} > 0 \\ \eta^- \cdot \Delta_{k,j}^{(i-1)} & \frac{\partial K^{(i-1)}(\mathbf{w})}{\partial w_{k,j}} \cdot \frac{\partial K^{(i)}(\mathbf{w})}{\partial w_{k,j}} < 0 \\ \Delta_{k,j}^{(i-1)} & \frac{\partial K^{(i-1)}(\mathbf{w})}{\partial w_{k,j}} \cdot \frac{\partial K^{(i)}(\mathbf{w})}{\partial w_{k,j}} = 0 \end{cases} \quad (58)$$

Για να αποφευχθούν φαινόμενα που ο αλγόριθμος θα μεταβάλλει τα βάρη με πολύ μικρή ταχύτητα ή με πολύ μεγάλη, είναι απαραίτητο να δοθεί κάποια μέγιστη και κάποια ελάχιστη τιμή για τα Δ , ενδεικτικά $\Delta_{(k,j)}^{MAX} = 50$ και $\Delta_{(k,j)}^{Min} = 10^{-6}$ (Riedmiller & Braun, 1993). Επίσης θα πρέπει να δοθεί μια αρχική τιμή για το $\Delta_{k,j}$, αυτή θα πρέπει να είναι ανάλογη με τα συνολικά βάρη, με το 0.1 να αποτελεί μια καλή επιλογή (Riedmiller & Braun, 1993). Οι παράμετροι η^+ και η^- καθορίζονται επίσης πριν την έναρξη της εκπαίδευσης και θα πρέπει να πληρούν το εξής κριτήριο $0 < \eta^- < 1 < \eta^+$. Συνοπτικά, η μέθοδος λειτουργεί ως εξής: κάθε φορά που η μερική παράγωγος των βαρών αλλάζει πρόσημο στο $w_{k,j}$ (κάτι που υποδηλώνει ότι το τελευταίο βήμα ήταν πολύ μεγάλο και ο αλγόριθμος έκανε άλμα στην άλλη πλευρά της καμπύλης), η νέα παράμετρος μειώνεται με την χρήση του συντελεστή η^- . Εάν η παράγωγος διατηρεί το ίδιο πρόσημο, η μεταβλητή $\Delta_{k,j}$, αυξάνεται ώστε να επιταχυνθεί η σύγκλιση (Riedmiller & Braun, 1993).

Αφού καθοριστεί ο τρόπος που θα αλλάξουν τα βάρη, η αλλαγή των βαρών ακολουθεί τον εξής απλό κανόνα. Εάν η παράγωγος έχει θετική τιμή (κάτι που υποδηλώνει την αύξηση του σφάλματος), τα βάρη μειώνονται με βάση το $\Delta_{k,j}$, ενώ αν η παράγωγος είναι αρνητική, το $\Delta_{k,j}$, προστίθεται. Εξαίρεση στο παραπάνω αποτελεί η περίπτωση που η παράγωγος αλλάξει πρόσημο, σε αυτή την περίπτωση ο αλγόριθμος επανέρχεται σε μια προγενέστερη κατάσταση. Το παραπάνω περιγράφεται από την ακόλουθη εξίσωση

$$w_{k,j}^{(i+1)} = \begin{cases} w_{k,j}^{(i)} - \Delta_{k,j}^{(i)} & \frac{\partial K^{(i)}(\mathbf{w})}{\partial w_{k,j}} > 0 \wedge \frac{\partial K^{(i-1)}(\mathbf{w})}{\partial w_{k,j}} \cdot \frac{\partial K^{(i)}(\mathbf{w})}{\partial w_{k,j}} \geq 0 \\ w_{k,j}^{(i)} + \Delta_{k,j}^{(i)} & \frac{\partial K^{(i)}(\mathbf{w})}{\partial w_{k,j}} < 0 \wedge \frac{\partial K^{(i-1)}(\mathbf{w})}{\partial w_{k,j}} \cdot \frac{\partial K^{(i)}(\mathbf{w})}{\partial w_{k,j}} \geq 0 \\ w_{k,j}^{(i)} & \frac{\partial K^{(i)}(\mathbf{w})}{\partial w_{k,j}} = 0 \\ w_{k,j}^{(i-1)} \wedge \frac{\partial K^{(i-1)}(\mathbf{w})}{\partial w_{k,j}} \equiv 0 & \frac{\partial K^{(i-1)}(\mathbf{w})}{\partial w_{k,j}} \cdot \frac{\partial K^{(i)}(\mathbf{w})}{\partial w_{k,j}} < 0 \end{cases} \quad (59)$$

Προκειμένου να αποφευχθεί η διπλή τιμωρία της τιμής ενημέρωσης, δεν πρέπει να υπάρξει προσαρμογή του $\Delta_{k,j}$, στο επόμενο βήμα. Πρακτικά αυτό επιτυγχάνεται μηδενίζοντας την τιμή της παραγώγου στο προηγούμενο βήμα (Riedmiller & Braun, 1993). Η μέθοδος *RProp*, είναι αρκετά γρήγορη μέθοδος, καθώς είναι μέθοδος 1^{ης} τάξης και δεν χρειάζεται ο υπολογισμός της 2^{ης} παραγώγου την στιγμή που καταφέρνει να είναι πιο αποδοτική από άλλες μεθόδους όπως οι μέθοδοι συζυγούς κλίσης ή η μέθοδος απότομης κάθοδου. (Prechelt, 1998; Rönngvaldsson, 1998).

Μέθοδοι συζυγούς κλίσης

Οι μέθοδοι της συζυγούς κλίσης (*Conjugate Gradient*) μελετούν την ελαχιστοποίηση της συνάρτησης

$$K(\mathbf{W}) = \frac{1}{2} \mathbf{W}^T \cdot \mathbf{A} \cdot \mathbf{W} + \mathbf{b}^T \cdot \mathbf{W} + c \quad (60)$$

Όπου ο \mathbf{A} , είναι ένας $(m \times m)$ συμμετρικός, θετικά ορισμένος πίνακας, το \mathbf{b} , είναι ένα διάνυσμα $(m \times 1)$, και το c ένας σταθερός όρος

Η ελαχιστοποίηση αυτής της συνάρτησης γίνεται για $\mathbf{W}^* = \mathbf{A}^{-1} \cdot \mathbf{b}$.

Για δεδομένο πίνακα \mathbf{A} , ορίζονται ως \mathbf{A} -συζυγή (*A-conjugate*) το σύνολο μη μηδενικών διανυσμάτων $\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_{m-1}$, που ικανοποιούν την συνθήκη

$$\mathbf{s}_n^T \cdot \mathbf{A} \cdot \mathbf{s}_j = 0 \quad \forall n \neq j \mid n, j \in \mathbb{N} \quad (61)$$

Η πρώτη και η δεύτερη παράγωγος της συνάρτησης που παρουσιάζεται στην [εξίσωση 60](#) θα είναι

$$\mathbf{g} = \nabla K(\mathbf{W}) = \mathbf{A} \cdot \mathbf{W} + \mathbf{b} \quad (62)$$

$$\mathbf{H} = \nabla^2 K(\mathbf{W}) = \mathbf{A} \quad (63)$$

Έχει αποδειχθεί ότι αν γίνουν m επαναλήψεις σε όλα τα σύνολα $\{\mathbf{s}_0, \mathbf{s}_1, \dots, \mathbf{s}_{m-1}\}$, θα βρεθεί το ελάχιστο της συνάρτησης (Hagan, et al., 2014). Επομένως η αλλαγή στην κλήση στην επανάληψη $k+1$ θα δίνεται

$$\Delta \mathbf{g}_k = \mathbf{g}_{k+1} - \mathbf{g}_k = (\mathbf{A} \cdot \mathbf{W}_{k+1} + \mathbf{b}) - (\mathbf{A} \cdot \mathbf{W}_k + \mathbf{b}) = \mathbf{A} \cdot \Delta \mathbf{W}_k \quad (64)$$

Επομένως, έστω \mathbf{s}_k η διεύθυνση που ελαχιστοποιεί την συνάρτηση. Η διαφοροποίηση στα βάρη θα είναι $\Delta \mathbf{W}_k = \mathbf{W}_{k+1} - \mathbf{W}_k = \mu_k \cdot \mathbf{s}_k$, ενώ η [εξίσωση 61](#), θα πάρει την εξής μορφή $\mathbf{s}_k^T \cdot \mathbf{A} \cdot \mathbf{s}_j = 0 \Leftrightarrow \Delta \mathbf{W}_k^T \cdot \mathbf{A} \cdot \mathbf{s}_j = 0 \Leftrightarrow$

$$\Delta \mathbf{g}_k^T \cdot \mathbf{s}_j = 0 \quad (65)$$

Με αυτόν τον τρόπο η αναζήτηση της λύσης δεν χρειάζεται τον υπολογισμό του πίνακα *Hessian* (Hagan, et al., 2014).

Η αναζήτηση της λύσης μπορεί να ξεκινήσει, θέτοντας $\mathbf{s}_0 = -\mathbf{g}_0$. Έχει αποδειχθεί ότι η αναζήτηση των επόμενων \mathbf{s} , μπορεί να απλοποιηθεί με βάση την εξίσωση

$$\mathbf{s}_k = -\mathbf{g}_k + \beta_k \cdot \mathbf{s}_{k-1} \quad (66)$$

Ανάλογα με ποιον τρόπο υπολογίζονται οι βαθμίδες β_k , προκύπτουν και οι διάφορες παραλλαγές της μεθόδου. Οι πιο διαδεδομένες παραλλαγές είναι

- Μέθοδος Συζυγούς Κλίσης με επανεκκινήσεις τύπου *Fletcher-Reeves*

$$\beta_k = \frac{\mathbf{g}_k^T \cdot \mathbf{g}_k}{\mathbf{g}_{k-1}^T \cdot \mathbf{g}_{k-1}} \quad (67)$$

- Μέθοδος Συζυγούς Κλίσης με επανεκκινήσεις τύπου *Polak-Ribiere*

$$\beta_k = \frac{\Delta \mathbf{g}_{k-1}^T \cdot \mathbf{g}_k}{\mathbf{g}_{k-1}^T \cdot \mathbf{g}_{k-1}} \quad (68)$$

- Μέθοδος Συζυγούς Κλίσης με επανεκκινήσεις τύπου *Beale-Powell*

Η μέθοδος αυτή διαφοροποιείται σε σχέση με τις προηγούμενες
Ο *Beale* όρισε την σχέση

$$\mathbf{d}_k = -\mathbf{g}_k + \beta_k \cdot \mathbf{d}_{k-1} + \gamma_k \cdot \mathbf{d}_t \quad \forall k > t \quad (69)$$

$$\text{Με } \beta_k = \frac{\mathbf{g}_{k-1}^T \cdot \mathbf{y}_{k-1}}{\mathbf{d}_{k-1}^T \cdot \mathbf{y}_{k-1}} \text{ και } \gamma_k = \frac{\mathbf{g}_k^T \cdot \mathbf{y}_t}{\mathbf{d}_t^T \cdot \mathbf{y}_t}$$

Η μέθοδος αυτή δεν είχε καλή απόδοση, έως ότου ο *Powell* ορίσει το κριτήριο

$$\left| \mathbf{g}_k^T \cdot \mathbf{g}_{k-1} \right| \leq c \cdot \left\| \mathbf{g}_k \right\|^2 \quad (70)$$

Όπου c είναι μια σταθερά που ανήκει στο διάστημα $(0,1)$ ([Dai, Liao, & Li, 2004](#)).

Εκτός από τις παραπάνω περιπτώσεις, ιδιαίτερη αναφορά θα πρέπει να γίνει στην μέθοδο της **Κλιμακωτής Συζυγούς Κλίσης**. Οι προηγούμενες εκδοχές των συζυγών κλίσεων, καθώς και αυτές των *Quasi-Newton*, απαιτούν την γραμμική αναζήτηση σε κάθε εποχή, κάτι που αυξάνει το υπολογιστικό κόστος. Η μέθοδος αυτή, σχεδιάστηκε για να αποφύγει αυτήν την χρονοβόρα διαδικασία. Αντί για την χρήση μεθόδου αναζήτησης, προσπαθεί να πραγματοποιήσει μια μη συμμετρική προσέγγιση του \mathbf{s}_k , ενσωματώνοντας την ικανότητα του αλγορίθμου *Levenberg-Marquardt* να προσεγγίζει με βάση την περιοχή εμπιστοσύνης ([Beale, Hagan, & Demuth, 2017](#)). Το \mathbf{s}_k υπολογίζεται από την εξίσωση

$$\mathbf{s}_k = \frac{\nabla K(\mathbf{W}_k + \sigma_k \cdot \mathbf{s}_{k-1}) - \nabla K(\mathbf{W}_k)}{\sigma_k} + \lambda_k \cdot \mathbf{s}_{k-1} \quad 0 < \sigma_k \ll 1 \quad (71)$$

Ο τρόπος που μεταβάλετε το λ_k , μπορεί να μην έχει το υπολογιστικό κόστος, αλλά περιλαμβάνει αρκετές εξισώσεις, η ανάλυση των οποίων ξεπερνά τους στόχους της παρούσας εργασίας. Επιγραμματικά, η μεταβολή του λ_k πραγματοποιείται με παραπλήσιο τρόπο, με την μεταβολή του Δ στην μέθοδο *RProp*. Όταν η ποσότητα $\mathbf{s}_k^T \cdot \mathbf{A} \cdot \mathbf{s}_k$, είναι μεγαλύτερη από το μηδέν, το λ_k μειώνεται, αντιθέτως όταν είναι αρνητική, το λ_k αυξάνεται. Η αρχική τιμή του λ_k θα πρέπει επίσης να είναι αρκετά μικρότερη της μονάδας, όπως και το σ_k ([Møller, 1993](#); [Chel, Majumder, & Nandi, 2011](#)).



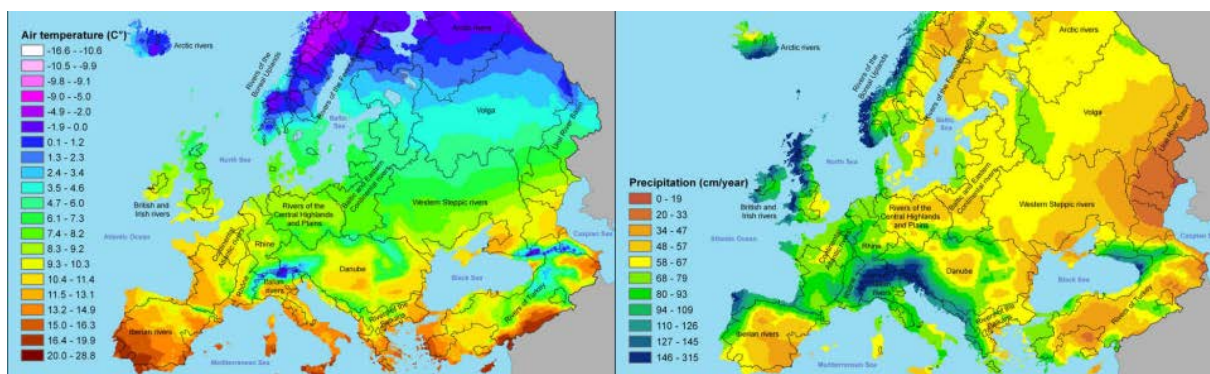
Εικόνα 26: Λεκάνη απορροής του Δούναβη, διακρίνεται ο ποταμός και οι παραπόταμοι του, οι χώρες που διαπερνά αυτός και οι παραπόταμοι του (Sommerwerk, et al., 2009)

ΠΕΡΙΟΧΗ ΜΕΛΕΤΗΣ

Ο ΔΟΥΝΑΒΗΣ

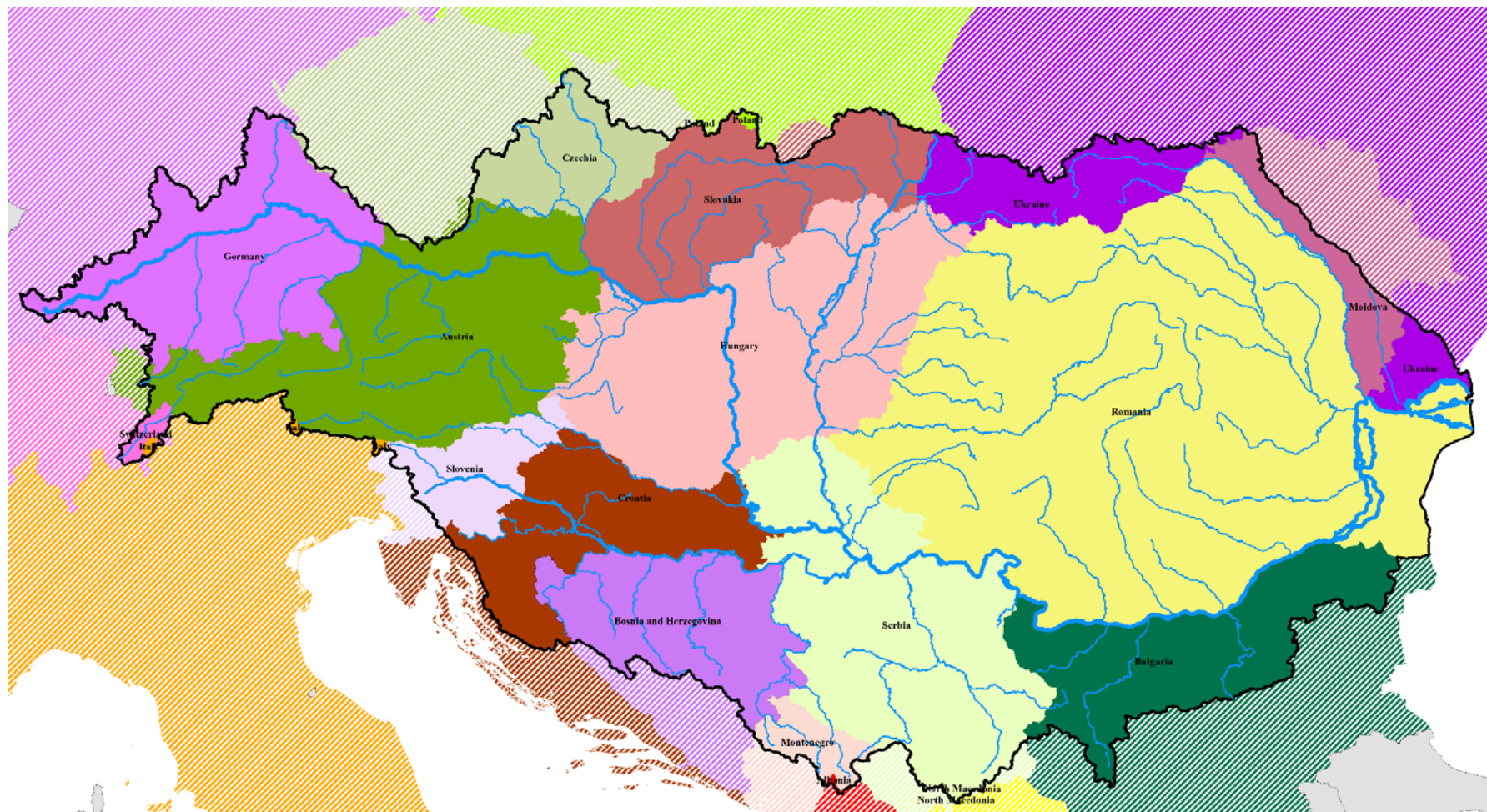
Ο Δούναβης, είναι ο δεύτερος μεγαλύτερος ποταμός της Ευρώπης (μετά τον Βόλγα) και ο μεγαλύτερος ποταμός που διαχέει χώρες της Ευρωπαϊκής Ένωσης με συνολικό μήκος από τις πηγές του στην Γερμανία μέχρι το δέλτα του στην μαύρη θάλασσα να είναι 2857km (Pavlović, et al., 2016; Vignati, et al., 2013). Συνολικά διατρέχει 10 διαφορετικές χώρες (Γερμανία, Αυστρία, Σλοβακία, Ουγγαρία, Κροατία, Σερβία, Βουλγαρία, Ρουμανία, Μολδαβία, και Ουκρανία) την ώρα που η λεκάνη απορροής του εκτείνεται σε 817,000km² (Vignati, et al., 2013) και σε ακόμα 9 χώρες (Αλβανία, Βόρεια Μακεδονία, Βοσνία-Ερζεγοβίνη, Ελβετία, Ιταλία, Μαυροβούνιο Πολωνία, Σλοβενία και Τσεχία) (Εικόνα 26)(Schmedtje, et al., 2005; Pavlović, et al., 2016). Ενώ μέσω καναλιών συνδέεται με τον Ρήνο με αποτέλεσμα να μπορεί να συνδέει το Ρότερνταμ με την Μαύρη Θάλασσα (Trans European Transport Network – TEN-T) (Mihic, Golusin, & Mihajlovic, 2011).

Λόγο της έκτασης της λεκάνης απορροής, του διακριτικού προσανατολισμού της Δύσης ανατολής καθώς και της τεράστιας ποσότητας νερού του ίδιου του ποταμού, ο Δούναβης παρουσιάζει μεγάλη κλιματική ετερογένεια και δεν μπορεί να περιγράψει από ένα κλίμα. Η κλιματική περιγραφή του μπορεί να γίνει τμηματικά ως εξής: η περιοχή του Άνω Δούναβη, έχει σημαντική επίδραση από το κλίμα του Ατλαντικού, παρατηρούνται υψηλά ποσοστά βροχόπτωσης και ήπιοι χειμώνες. Η ανατολική πλευρά της λεκάνης του Δούναβη, έχουν μεγάλη επιρροή από το ηπειρωτικό κλίμα και χαρακτηρίζονται από χαμηλές βροχοπτώσεις και ξηρούς κρύους χειμώνες. Ενώ η νοτιοδυτική περιοχή της λεκάνης απορροής που βρίσκεται σε βόρεια Ιταλία, Σλοβενία και Κροατία, έχει σημαντική επίδραση από το Μεσογειακό κλίμα. Η βροχόπτωση κυμαίνεται από 500 έως 2000mm, με το μέγιστο να προκύπτει τον Ιούλιο στην περιοχή του Άνω Δούναβη, τον Μάιο-Ιούνιο στο νοτιοανατολικό κομμάτι ενώ στη μεσογειακή πλευρά της λεκάνης η μέγιστη βροχόπτωση παρατηρείται το φθινόπωρο (Εικόνα 27) (Sommerwerk, et al., 2009).



Εικόνα 27: Χάρτες μέσης θερμοκρασίας και μέσης βροχόπτωσης (Sommerwerk, et al., 2009)

Αποτελεί πηγή πόσιμου νερού για αρκετές πόλεις ενώ τρεις πρωτεύουσες (Βουδαπέστη, Βελιγράδι και Μπρατισλάβα), καλύπτουν πάνω από το 50% των αναγκών από τον ποταμό, την ώρα που εμπλουτίζει πηγάδια που χρησιμοποιούνται για την ύδρευση πόλεων στην Σλοβακία την Ουγγαρία και την Γερμανία (Ascott, et al., 2016). Ενώ το δέλτα του ποταμού είναι ένας από τους σημαντικότερους υδροβιότοπους της Ευρώπης και έχει χαρακτηριστεί Μνημείο Παγκόσμιας Κληρονομιάς της UNESCO (Pavlović, et al., 2016).



Εικόνα 28: Η περιοχή μελέτης με τις χώρες, τον Δούναβη και τους κύριους παραποτάμους του, όπως εισήχθη στο πρόγραμμα ArcGIS. Η λήψη των δεδομένων πραγματοποιήθηκε από τους οργανισμούς ICPDR (International Commission for the Protection of the Danube River) και από την Eurostat¹.

¹ © EuroGeographics for the administrative boundaries

ΠΕΙΡΑΜΑΤΙΚΗ ΔΙΑΔΙΚΑΣΙΑ

ΠΡΟΕΤΟΙΜΑΣΙΑ ΔΕΔΟΜΕΝΩΝ

Λόγο των πολλών υπηρεσιών που είναι υπεύθυνες για την περιοχή μελέτης, τα δεδομένα ήταν μεν δομημένα, αλλά σε διαφορετικού είδους αρχεία. Αυτό είχε ως αποτέλεσμα να χρειαστεί η επεξεργασία των δεδομένων, ώστε να μπορούν οι πληροφορίες να οργανωθούν με μια συγκεκριμένη τυποποίηση ώστε να μπορούν στην συνέχεια να αξιοποιηθούν σε κάποιο μοντέλο. Οι πληροφορίες που υπήρχαν για την περιοχή μελέτης μπορούν να χωριστούν σε τέσσερις κατηγορίες. Αρχικά υπήρχαν οι θέσεις των σταθμών σε ψηφιακά αρχεία τύπου *shapefile*¹, ενώ τα μετεωρολογικά δεδομένα καθώς και το υψόμετρο της περιοχής, υπήρχαν σε αρχεία τύπου *raster*². Για την εξαγωγή των πληροφοριών από αυτά τα αρχεία σε αρχεία τύπου υπολογιστικών φύλλων (*spreadsheet*) ή σε αρχεία κειμένου με δομημένα δεδομένα (*comma separated values – csv*) ώστε να μπορούν να χρησιμοποιηθούν οι τιμές τους, ήταν απαραίτητη η χρήση του προγράμματος ArcGIS (Εικόνα 28). Στην συνέχεια υπήρχαν, τα υδρογεωλογικά δεδομένα από έξι χώρες (Βουλγαρία, Ουγγαρία, Ρουμανία, Σερβία, Σλοβενία, Τσεχία) που με εξαίρεση την Ουγγαρία, ήταν ένα μικρό ποσοστό των συνολικών δεδομένων, ήταν ήδη επεξεργασμένα και κατηγοριοποιημένα σε υπολογιστικά φύλλα. Ενώ τέλος τα υδρογεωλογικά δεδομένα από Αυστρία, Γερμανία και Κροατία, ήταν σε ξεχωριστά αρχεία χρονοσειρών και χρειαζόντουσαν μια επεξεργασία ώστε να μπορούν να αξιοποιηθούν σε αυτή την εργασία.

ΔΙΑΛΟΓΗ ΔΕΔΟΜΕΝΩΝ

Αρχικά πραγματοποιήθηκε μια διαδικασία διαλογής δεδομένων, καθώς μετά από έλεγχο, κάποιοι σταθμοί βρέθηκε ότι είχαν τις ίδιες συντεταγμένες. Αυτό μπορεί είτε να έγινε από σφάλμα εισαγωγής των δεδομένων στο *shapefile* είτε επειδή οι σταθμοί ήταν σε σχετικά κοντινή απόσταση, υπάρχει το ενδεχόμενο να μην γινόντουσαν ακριβές μετρήσεις από τις αρμόδιες υπηρεσίες και να δινόταν και στους δύο (ή τρεις σε ορισμένες περιπτώσεις) σταθμούς, οι ίδιες συντεταγμένες. Σε κάθε περίπτωση, τυχών ενσωμάτωση αυτών των σταθμών θα οδηγούσε να υπάρχουν ίδια διανύσματα εισόδου με διαφορετικά διανύσματα εξόδου, για το λόγο αυτό αποφασίστηκε να μην συμπεριληφθούν σταθμοί που έχουν κοινές συντεταγμένες, καθώς θα περιπλεκόταν η διαδικασία της εκπαίδευσης. Εξαιτίας της ταύτισης των συντεταγμένων, αποκλείστηκαν συνολικά 48 χρονοσειρές δεδομένων, εκ των οποίων οι 23 αφορούν σταθμούς που βρίσκονται στην Γερμανία, 15 που βρίσκονται στην Αυστρία, 8 που βρίσκονται στην Κροατία και 2 που βρίσκονται στην Σερβία. Οι σταθμοί που αποκλείστηκαν, σε σχέση με αυτούς που προτιμήθηκαν, παρουσίαζαν μικρές αποκλίσεις, πλην ελάχιστων εξαιρέσεων, στην στάθμη του υδροφόρου ορίζοντα. Για το λόγο αυτό η επιλογή των σταθμών έγινε με βάση το πλήθος μετρήσεων. Στις περιπτώσεις που παρατηρήθηκε σημαντική απόκλιση της υδραυλικής στάθμης των σταθμών, επιλέχθηκε το πηγάδι με την μεγαλύτερη στάθμη.

¹ Η μορφή *shapefile* είναι μια μορφή γεωχωρικών διανυσματικών δεδομένων για λογισμικό συστήματος γεωγραφικών πληροφοριών (G.I.S.)

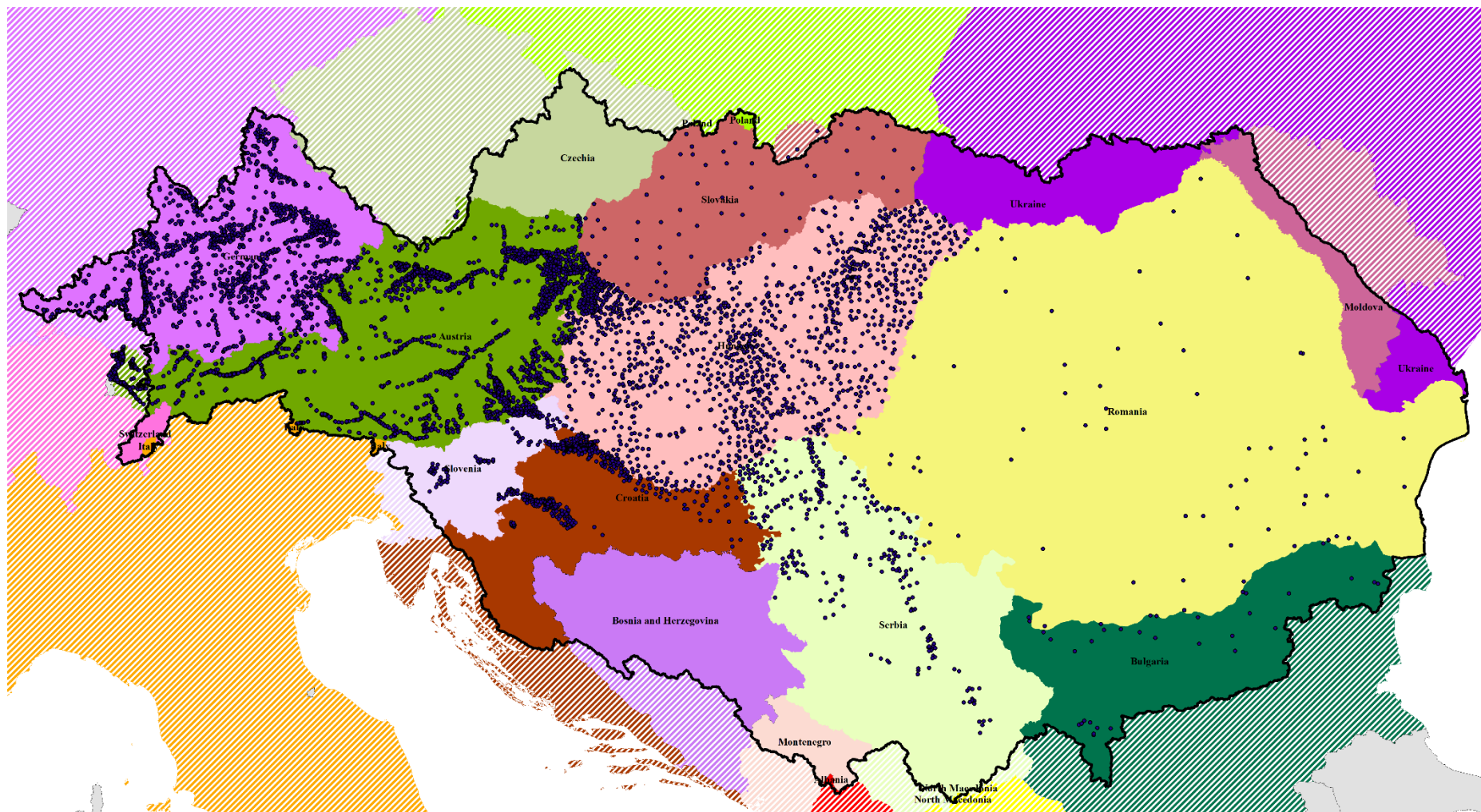
² Ένα αρχείο εικόνας *raster* είναι ένας ορθογώνιος πίνακας με τιμές που λαμβάνονται συνήθως ως δείγμα, γνωστές ως *pixel* (στοιχείο εικόνας). Κάθε *pixel* έχει έναν ή περισσότερους αριθμούς που σχετίζονται με αυτό, καθορίζοντας ένα χρώμα στο οποίο θα πρέπει να εμφανίζεται το *pixel*.

Στην συνέχεια έγινε ένας έλεγχος για το πόσοι σταθμοί είχαν διαθέσιμη πληροφορία τόσο για την τοποθεσία τους, όσο και την εξέλιξη της στάθμης του υδροφόρου ορίζοντα, καθώς υπήρχαν αρκετοί σταθμοί για τους οποίους δεν μπόρεσε να βρεθεί κάποια χρονοσειρά δεδομένων, καθώς και κάποιες χρονοσειρές, που δεν υπήρχαν συντεταγμένες για τους σταθμούς που αναφέρονταν. Το σύνολο των σταθμών που διέθεταν όλα τα δεδομένα ξεπερνούσε τις επτάμιση χιλιάδες (για την ακρίβεια ήταν 7718) (Εικόνα 29 ; Πίνακας 1) την ώρα που το σύνολο των εγγραφών για υδραυλικά ύψη να ξεπερνά τα πέντε εκατομμύρια.

Επειδή αρκετοί σταθμοί είχαν μικρές αποστάσεις μεταξύ τους, ακόμα και χρησιμοποιώντας ένα ποσοστό ως υποσύνολο ελέγχου, υπήρχε ο κίνδυνος ένα μέρος αυτού του υποσυνόλου, ουσιαστικά να είχε ταυτόσημες τιμές με πηγάδια που χρησιμοποιήθηκαν για την εκπαίδευση των νευρωνικών. Για το λόγο αυτό και για να μειωθεί ο όγκος των δεδομένων εκπαίδευσης, τα δεδομένα της Ουγγαρίας εξαιρέθηκαν από την εκπαίδευση των Τ.Ν.Δ και κρατήθηκαν για να γίνει έλεγχος των προτεινόμενων μοντέλων στο τέλος. Σε αυτή την απόφαση συντέλεσε το γεγονός ότι η Ουγγαρία βρίσκεται στο κέντρο της περιοχής μελέτης, επομένως ακόμα και αν δεν υπάρχουν καθόλου δεδομένα για αυτήν στην εκπαίδευση ενός Τ.Ν.Δ, ένα καλά εκπαιδευμένο Τ.Ν.Δ. θα πρέπει να μπορεί να γενικεύει την λειτουργία του σε αυτήν, χωρίς ιδιαίτερη αύξηση του σφάλματος. Έτσι ο συνολικός αριθμός των σταθμών που χρησιμοποιήθηκαν για εκπαίδευση των Τ.Ν.Δ., έπεσε στους 6487 σταθμούς.

Πίνακας 1: Οι χώρες που διέρχεται ο Δούναβης, αριθμός πηγαδιών, αριθμός διαθέσιμων χρονοσειρών και αριθμός σταθμών που χρησιμοποιήθηκαν.

Χώρα	Πηγάδια με Συντεταγμένες	Δεδομένα Χρονοσειρών	Σταθμοί που χρησιμοποιήθηκαν	Είδος εγγραφών
Αυστρία	3399	3409	3384	Μηνιαία
Βουλγαρία	36	36	36	Μηνιαία
Γερμανία	1927	1928	1895	Ανάλογα τον σταθμό
Κροατία	696	699	688	Δύο φορές την εβδομάδα
Ουγγαρία	1431	1166	1166	Μηνιαία
Ρουμανία	65	65	65	Ετήσια
Σερβία	300	300	298	Ανά 5 ημέρες
Σλοβακία	71	71	71	Εβδομαδιαία
Σλοβενία	115	115	115	Συνήθως ανά 5 ημέρες
ΣΥΝΟΛΟ	8040	7789	7718	



Εικόνα 29: Η θέση των πηγαδιών που χρησιμοποιήθηκαν για την εκπαίδευση των τεχνητών νευρωνικών δικτύων.

ΑΝΑΓΝΩΣΗ ΧΡΟΝΟΣΕΙΡΩΝ

Όπως αναφέρθηκε τα υδρολογικά δεδομένα των έξι χωρών, ήταν ήδη διαμορφωμένα και δεν χρειαζόντουσαν επεξεργασία για την μετέπειτα χρήση τους. Αντιθέτως για τις τρεις υπόλοιπες χώρες χρειαζόταν να γίνει επεξεργασία των αρχικών χρονοσειρών ώστε να προκύψουν αρχεία σε κατάλληλη μορφή ώστε να μπορούν να χρησιμοποιηθούν για την εκπαίδευση των Τ.Ν.Δ.. Όπως φαίνεται και στον Πίνακα 1, υπήρχε αναντιστοιχία των πηγαδιών για τα οποία υπήρχε γνώση του γεωγραφικού στίγματος τους, με αυτά που υπήρχε γνώση των χρονοσειρών τους. Πέρα από αυτό, υπήρχε διαφοροποίηση στο όνομα με το οποίο ήταν αποθηκευμένες οι χρονοσειρές σε κάθε χώρα, με αυτό που ήταν καταχωρημένοι οι σταθμοί στο αρχείο με τα δεδομένα. Σαν πρώτο βήμα λοιπόν, έπρεπε να δημιουργηθεί μια λίστα με τα πηγάδια στα οποία υπήρχαν διαθέσιμα δεδομένα χρονοσειρών και συντεταγμένες, καθώς και αντιστοίχιση των ονομάτων των σταθμών με τα αρχεία χρονοσειρών.

Το πλήθος των χρονοσειρών των υπόλοιπων τριών χωρών (σχεδόν 6000) καθιστούσε αδύνατη την χειροκίνητη διαχείριση τους, καθώς ήταν σε διαφορετικού τύπου αρχεία, με διαφορετική κωδικοποίηση και διαμόρφωση. Για το λόγο αυτό θα έπρεπε να δημιουργηθεί κάποιος αλγόριθμος που να μπορούσε να εκτελέσει αυτή την διαδικασία. Λόγω της διαφοροποίησης της διαμόρφωσης των δεδομένων σε κάθε χώρα, χρησιμοποιήθηκαν διάφορες παραλλαγές του ίδιου αλγορίθμου, προσαρμοσμένες στις ανάγκες της εκάστοτε χώρας. Στο παράρτημα παρατίθεται ο αλγόριθμος που χρησιμοποιήθηκε για την Αυστρία, ο οποίος με κάποιες τροποποιήσεις όσον αφορά την ανάγνωση των δεδομένων, μπορεί να χρησιμοποιηθεί και για τις υπόλοιπες χώρες.

1	Messstelle:	;Achau, Br	1	Messstelle:	;Leobersdorf, Bl 451
2	HZB-Nummer:	;300012	2	HZB-Nummer:	;300699
3	HD-Nummer:	;	3	HD-Nummer:	;
4	DBMS-Nummer:	;3002000	4	DBMS-Nummer:	;3002048
5	PorenGW-Gebiet:	;250 Südliches Wiener Becken	5	PorenGW-Gebiet:	;250 Südliches Wiener Becken
6	Grundwasserkörper:	;Südl. Wiener Becken [DUJ]	6	Grundwasserkörper:	;Südl. Wiener Becken [DUJ]
7	Sachgebiet:	;UWQ	7	Sachgebiet:	;UWQ
8	Arbeitsgebiet:	;GW	8	Arbeitsgebiet:	;GW
9	Dienststelle:	;HD-Niederösterreich	9	Dienststelle:	;HD-Niederösterreich
10	Messstellenbetreiber:	;Hydrographischer Dienst	10	Messstellenbetreiber:	;Hydrographischer Dienst
11	Höhenangaben [m ü.A.]:	;gültig seit: ;05.07.1999	11	Höhenangaben [m ü.A.]:	;gültig seit: ;01.07.1996
12	Geländehöhe:	;172.65	12	Geländehöhe:	;266.18
13	Messpunkthöhe:	;172.79	13	Messpunkthöhe:	;266.88
14	Schllage:	;168.90	14	Schllage:	;259.58
15	T-Messtiefe u.GOK:	;	15	T-Messtiefe u.GOK:	;
16	Geographische Koordinaten (Referenzellipsoid: Bessel 1841):		16	Geographische Koordinaten (Referenzellipsoid: Bessel 1841):	
17	gültig seit:	;Länge (Grad,Min,Sek): ;Breite (Grad,Min,Sek):	17	gültig seit:	;Länge (Grad,Min,Sek): ;Breite (Grad,Min,Sek):
18	03.05.1940	;16 23 15 ;48 04 54	18	11.10.1963	;16 13 02 ;47 55 53
19	01.01.2009	;16 23 13 ;48 04 54	19	01.07.1996	;16 13 01 ;47 55 52
20	Exportzeitreihe:	;GWS,I,Mit,M,1,A,Z,0,,,	20	01.01.2009	;16 13 01 ;47 55 50
21	Exportqualität:	;MAXQUAL	21	Exportzeitreihe:	;GWS,I,Mit,M,1,A,Z,0,,,
22	Einheit:	;m ü.A.	22	Exportqualität:	;MAXQUAL
23	Exportzeitraum:	;01.01.1900 00:00; bis ;31.12.2013 00:00	23	Einheit:	;m ü.A.
24	Der Intervallwert gilt bis	zum nächsten Zeitpunkt mit einem Wert oder Lücke	24	Exportzeitraum:	;01.01.1900 00:00; bis ;31.12.2013 00:00
25	Werteformat:	;Fehlertoleranz	25	Der Intervallwert gilt bis	zum nächsten Zeitpunkt mit einem Wert oder Lücke
26	Werte:		26	Werteformat:	;Fehlertoleranz
27	01.12.1940 00:00:00	; 171,4200	27	Werte:	
28	01.01.1941 00:00:00	; 171,4700	28	01.01.1964 00:00:00	; 262,2900
29	01.02.1941 00:00:00	; 171,9400	29	01.02.1964 00:00:00	; 262,4700
30	01.03.1941 00:00:00	; 172,1500	30	01.03.1964 00:00:00	; 262,6500
31	01.04.1941 00:00:00	; 172,0800	31	01.04.1964 00:00:00	; 262,8400
32	01.05.1941 00:00:00	; 171,8300	32	01.05.1964 00:00:00	; 262,7800
33	01.06.1941 00:00:00	; 171,5600	33	01.06.1964 00:00:00	; 262,7700
34	01.07.1941 00:00:00	; 171,5600	34	01.07.1964 00:00:00	; 262,8100
35	01.08.1941 00:00:00	; 171,4000	35	01.08.1964 00:00:00	; 262,7900
36	01.09.1941 00:00:00	; 171,2200	36	01.09.1964 00:00:00	; 262,7800
37	01.10.1941 00:00:00	; 171,1400	37	01.10.1964 00:00:00	; 262,7500
38	01.11.1941 00:00:00	;Lücke	38	01.11.1964 00:00:00	; 262,6700
39	01.11.1942 00:00:00	; 170,9900	39	01.12.1964 00:00:00	; 262,7100
40	01.12.1942 00:00:00	; 170,9800	40	01.01.1965 00:00:00	; 262,6800

Εικόνα 30: Αρχεία χρονοσειρών της Αυστρίας, είναι εμφανής η διαφοροποίηση στην έναρξη των εγγραφών του εκάστοτε σταθμού, καθώς επίσης είναι εμφανής και η ύπαρξη αλφαριθμητικών χαρακτήρων, ανάμεσα στα δεδομένα.

Εκτός των προαναφερθέντων, οι χρονοσειρές των τριών χωρών παρουσίαζαν διάφορες ανωμαλίες οι οποίες θα έπρεπε να προβλεφθούν ώστε να γίνει σωστά η ανάγνωση των χρονοσειρών και η δημιουργία κατάλληλων αρχείων για εκπαίδευση Τ.Ν.Δ.. Ενδεικτικά στην Αυστρία, στα αρχεία που ήταν αποθηκευμένες οι χρονοσειρές, πριν την έναρξη των δεδομένων, υπήρχαν κάποιες γενικές πληροφορίες, όπως η θέση του σταθμού, η ημερομηνία έναρξης των μετρήσεων, η ονομασία του σταθμού, η περιοχή που βρίσκεται ο σταθμός κ.α.. Το πρόβλημα έγκειται στο γεγονός ότι αυτό το κείμενο δεν ήταν σταθερού μεγέθους, αλλά θα

μπορούσε να είναι 25, 26, 27 ή και παραπάνω γραμμές πριν τα δεδομένα. Επομένως ο αλγόριθμος που θα δημιουργούνταν, θα έπρεπε να είχε πρόβλεψη για αυτό, καθώς δεν αρκούσε να ξεκινήσει την ανάγνωση του αρχείου από ένα συγκεκριμένο σημείο και μετά (Εικόνα 30). Αντίστοιχο πρόβλημα υπήρχε και στα αρχεία των υπόλοιπων χωρών, μικρότερου μεγέθους όμως, καθώς εκεί απλά υπήρχε το ενδεχόμενο να ξεκινάνε άμεσα οι καταγραφές, χωρίς κάποια τιτλοφόρηση στην πρώτη γραμμή.

Ένα άλλο πρόβλημα των συγκεκριμένων αρχείων, ήταν ότι όταν ο μετρητής λόγο σφάλματος, δεν έπαιρνε μέτρηση, στην καταγραφή εμφανιζόταν η λέξη «*Lücke*» (μετ. «κενό»), έτσι ενώ το σύστημα λάμβανε μια αλφαριθμητική τιμή, ενώ προσπαθούσε να διαβάσει μια αριθμητική, με αποτέλεσμα να μην μπορεί να συνεχίσει την ανάγνωση αρχείων και να τερματίζεται.

Στην Γερμανία, σε κάποιους σταθμούς, υπήρχαν κάποιες περίοδοι, που το σύστημα κατέγραφε δύο ημερήσιες τιμές. Καθώς η ώρα των μετρήσεων δεν αναφερόταν κάπου, συνυπολογίζοντας το γεγονός ότι υπήρχαν διαθέσιμα ημερήσια και όχι αναλυτικότερα μετεωρολογικά δεδομένα, στις περιπτώσεις που υπήρχαν παραπάνω από μια καταγραφές, χρησιμοποιούνταν μόνο η πρώτη. Ένα άλλο πρόβλημα που παρουσίασαν οι χρονοσειρές στους γερμανικούς σταθμούς, ήταν ότι υπήρχαν καταγραφές στις οποίες λόγο σφάλματος του μετρητή, μετατρέποντουσαν τα μέτρα σε χιλιοστά, επομένως ο αλγόριθμος διαμορφώθηκε έτσι ώστε να αγνοεί τις συγκεκριμένες καταγραφές (Εικόνα 31).

900	899, BW01136660,	7001, 2001-11-10,	526.680	272	271, BW00525701,	7001, 2002-08-05,	635.200
901	900, BW01136660,	7001, 0-00-00,	0.000	273	272, BW00525701,	7001, 2002-08-19,	635.170
902	901, BW01136660,	7001, 2001-11-11,	526.690	274	273, BW00525701,	7001, 2002-09-02,	0.636
903	902, BW01136660,	7001, 0-00-00,	0.000	275	274, BW00525701,	7001, 2002-09-16,	636.030
904	903, BW01136660,	7001, 2001-11-12,	526.690	276	275, BW00525701,	7001, 2002-10-07,	635.840
905	904, BW01136660,	7001, 0-00-00,	0.000	277	276, BW00525701,	7001, 2002-10-21,	635.860
906	905, BW01136660,	7001, 2001-11-12,	526.700	278	277, BW00525701,	7001, 2002-11-04,	635.950
907	906, BW01136660,	7001, 0-00-00,	0.000	279	278, BW00525701,	7001, 2002-11-18,	635.990
908	907, BW01136660,	7001, 2001-11-13,	526.700	280	279, BW00525701,	7001, 2002-12-02,	636.020
909	908, BW01136660,	7001, 0-00-00,	0.000	281	280, BW00525701,	7001, 2002-12-16,	636.090
910	909, BW01136660,	7001, 2001-11-13,	526.690	282	281, BW00525701,	7001, 2003-01-06,	636.080
911	910, BW01136660,	7001, 0-00-00,	0.000	283	282, BW00525701,	7001, 2003-01-20,	636.060
912	911, BW01136660,	7001, 2001-11-14,	526.680	284	283, BW00525701,	7001, 2003-02-03,	636.040
913	912, BW01136660,	7001, 0-00-00,	0.000	285	284, BW00525701,	7001, 2003-02-10,	0.636
914	913, BW01136660,	7001, 2001-11-14,	526.670	286	285, BW00525701,	7001, 2003-03-03,	0.636
915	914, BW01136660,	7001, 0-00-00,	0.000	287	286, BW00525701,	7001, 2003-03-17,	635.920
916	915, BW01136660,	7001, 2001-11-15,	526.670	288	287, BW00525701,	7001, 2003-04-07,	635.320
917	916, BW01136660,	7001, 0-00-00,	0.000	289	288, BW00525701,	7001, 2003-04-21,	635.420
918	917, BW01136660,	7001, 2001-11-16,	526.670	290	289, BW00525701,	7001, 2003-05-05,	635.590
919	918, BW01136660,	7001, 0-00-00,	0.000	291	290, BW00525701,	7001, 2003-05-19,	635.660
920	919, BW01136660,	7001, 2001-11-16,	526.660	292	291, BW00525701,	7001, 2003-06-02,	635.630
921	920, BW01136660,	7001, 0-00-00,	0.000	293	292, BW00525701,	7001, 2003-06-16,	635.580
922	921, BW01136660,	7001, 2001-11-17,	526.660	294	293, BW00525701,	7001, 2003-07-07,	635.560

Εικόνα 31: Αρχεία χρονοσειρών της Γερμανίας, στην πρώτη περίπτωση φαίνονται οι μηδενικές καθώς και οι διπλές εγγραφές, ενώ στην δεύτερη περίπτωση εμφανίζονται κάποιες μετρήσεις που ο μετρητής μετατρέπει τα μέτρα σε χιλιοστά

Τέλος, υπήρχαν κάποιες μηδενικές καταγραφές, που θα έπρεπε ο αλγόριθμος να αγνοήσει. Όσον αφορά τις χρονοσειρές της Κροατίας δεν παρουσίασαν κάτι διαφορετικό από τις άλλες δύο, πέρα από ότι σε κάποιους σταθμούς μπορεί ο μετρητής να μην έκανε κάποιες καταγραφές και να εμφάνιζε κενό, κάτι που εύκολα ο αλγόριθμος μπορούσε να παρακάμψει.

Αν και υπήρχε η δυνατότητα ο αλγόριθμος να δημιουργήσει απευθείας ένα αρχείο με τα δεδομένα όλων των χρονοσειρών κάθε χώρας, εν τέλει προτιμήθηκε αρχικά να δημιουργηθούν ξεχωριστά διαμορφωμένα αρχεία για κάθε σταθμό. Στην συνέχεια η ένωση όλων των αρχείων, από την στιγμή που είχαν την απαιτούμενη διαμόρφωση, μπορούσε να ολοκληρωθεί με μια απλή εντολή του λειτουργικού συστήματος. Διαφορετικά ο αλγόριθμος θα έπρεπε να είχε στην μνήμη του πίνακες με πλήθος εγγραφών που θα ξεπερνούσε το ένα εκατομμύριο, κάτι που θα καθυστέρουσε σημαντικά την ολοκλήρωση της διαδικασίας.

ΕΝΣΩΜΑΤΩΣΗ ΜΕΤΕΩΡΟΛΟΓΙΚΩΝ ΔΕΔΟΜΕΝΩΝ

Με την χρήση του αλγορίθμου τα υδρολογικά καθώς και τα γεωγραφικά δεδομένα των τριών χωρών οργανώθηκαν σε αρχεία τύπου *csv*. Επομένως για να ολοκληρωθεί η προετοιμασία των δεδομένων θα έπρεπε να γίνει η ενσωμάτωση των μετεωρολογικών δεδομένων. Η εξαγωγή των δεδομένων αυτών, όπως αναφέρθηκε έγινε από αρχεία τύπου *raster*, και με την βοήθεια του Δρ. *Ιωάννη Τριχάκη*, αυτά οργανώθηκαν σε μορφή πινάκων, όπου οι στήλες αντιπροσώπευαν τους σταθμούς μελέτης και οι γραμμές τις ημέρες, με την 31/12/1989 να ορίζεται ως το σημείο μηδέν και τελική ημερομηνία την 31/10/2014. Το αποτέλεσμα ήταν να δημιουργηθούν πέντε πίνακες μεγέθους 8040×9070 που ενσωμάτωναν την πληροφορία για την βροχόπτωση, την μέση θερμοκρασία της ημέρας, την εξάτμιση από τις υδάτινες επιφάνειες, την δυνητική εξατμισοδιαπνοή και την εξατμισοδιαπνοή που προκαλείτε από την βλάστηση και το έδαφος. Καθώς και ένα διάνυσμα που ενσωμάτωνε την πληροφορία για το υψόμετρο της περιοχής που βρισκόταν ο σταθμός. Λόγο του ότι αυτή η πληροφορία αφορά ολόκληρη την περιοχή και όχι το σημείο που βρίσκεται ο σταθμός, η συγκεκριμένη τιμή αναμένετε να έχει μεγάλες αποκλίσεις από την πραγματική τιμή του υψόμετρου της περιοχής.

Επομένως θα έπρεπε να δημιουργηθεί ένας δεύτερος αλγόριθμος, ο οποίος θα έπρεπε να εκτελέσει δύο λειτουργίες. Αρχικά θα έπρεπε να διαβάζει τις εγγραφές του προηγούμενου πίνακα και αφού πραγματοποιούσε έλεγχο να συγκρατούσε μόνο τις τιμές οι οποίες ήταν μεταγενέστερες της ημερομηνίας μηδέν και προγενέστερες από την τελική ημερομηνία. Στην συνέχεια αφού η εγγραφή ήταν εντός αυτών των ορίων, να έβρισκε τις τιμές των αντίστοιχων μεταβλητών από τους αντίστοιχους πίνακες/διανύσματα και να αποθηκευόταν σε ένα ξεχωριστό αρχείο. Επειδή τα υπόγεια ύδατα παρουσιάζουν μια ανοχή στις περιβαλλοντικές αλλαγές και η επίδραση των κλιματικών συνθηκών ενδεχομένως να εμφανιστεί μετά από κάποιο χρονικό διάστημα, ο αλγόριθμος πέρα από τις τιμές των συγκεκριμένων μεταβλητών την ημέρα που υπήρχε κάποια εγγραφή για την στάθμη κάποιου πηγαδιού, κρατούσε στην μνήμη του και τις τιμές που είχαν οι αντίστοιχες μεταβλητές και τις δύο προηγούμενες μέρες.

Με αυτόν τον τρόπο δημιουργήθηκε ένας πίνακας μεγέθους 4427473×21. Όπου το πρώτο νούμερο που πλησιάζει τα τεσσαράμισι εκατομμύρια, αντιπροσωπεύει τα δείγματα που θα χρησιμοποιηθούν για την εκπαίδευση των Τ.Ν.Δ. και το 21 τις μεταβλητές εισόδου και εξόδου του συστήματος οι οποίες ήταν οι εξής

- Μια μεταβλητή που περιείχε πληροφορία για το έτος, ώστε να μπορούν να προβλεφθούν σύντομες επιδράσεις τις κλιματικής αλλαγής
- Μια μεταβλητή που περιείχε πληροφορία για την ημέρα του έτους, ώστε να υπάρχει διαφοροποίηση όσον αφορά τις εποχές. Η μεταβλητή αυτή προτιμήθηκε από το να μπουν ξεχωριστά οι ημέρες και ο μήνας, καθώς το πρώτο δεν θα είχε κάποια ιδιαίτερη σημασία όταν για παράδειγμα μια εγγραφή που θα αφορούσε την 20/7 θα είχε την ίδια τιμή με μια εγγραφή που αφορούσε την 20/12
- Μια μεταβλητή για κάθε συντεταγμένη X, Y, Z (γεωγραφικό πλάτος, γεωγραφικό μήκος, υψόμετρο)
- Μια μεταβλητή για την στάθμη του πηγαδιού

Και τρεις μεταβλητές για κάθε μια μετεωρολογική παράμετρο (βροχόπτωση, μέση θερμοκρασία της ημέρας, εξάτμιση από τις υδάτινες επιφάνειες, δυνητική εξατμισοδιαπνοή και εξατμισοδιαπνοή που προκαλείτε από την βλάστηση και το έδαφος)

ΧΡΗΣΗ ΤΟΥ MATLAB

Από την στιγμή που τα δεδομένα ήταν αποθηκευμένα σε αρχεία τύπου csv, η εισαγωγή τους στο περιβάλλον του *MATLAB*, ήταν απλή διαδικασία, χρησιμοποιώντας απλά την επιλογή της εισαγωγή δεδομένων (*import data*). Αφού φορτώσει τα δεδομένα το *MATLAB*, υπάρχει η δυνατότητα να γίνει τροποποίηση είτε στο όνομα της μεταβλητής είτε ακόμα και στον τύπο της (Εικόνα 32).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
	YEAR	DayOffYear	X	Y	Z	EO	ES	ET	PCP	Tavg	xEO	xES	xET	xPCP	xTavg	xEO	xES	xET	xPCP	xTavg	Z.D	Depth
1	YEAR	DayOffYear	X	Y	Z	EO	ES	ET	PCP	Tavg	xEO	xES	xET	xPCP	xTavg	xEO	xES	xET	xPCP	xTavg	Z.D	Depth
2	1990	32	4798361.33	2794738.23	174.81300030	0.11892300	0.13347000	0.13201300	0.00022297	3.88837000	0.49570500	0.54202900	0.58835200	0.00201759	3.36925000	0.54549900	0.60583800	0.66621700	0.00054892	5.51960000	3.58770300	171.2335
3	1990	32	4804828.51	2799168.22	182.14198800	0.11565000	0.13100000	0.14653500	0.00771342	4.08285000	0.38143300	0.41639800	0.44764900	0.01466050	4.02182000	0.48884100	0.53829100	0.58674000	0.01162160	5.21896000	26.24728830	155.8947
4	1990	32	4779649.68	2795653.38	152.35400090	0.10534000	0.11782500	0.13031100	0.00000000	2.73244000	0.58487700	0.64674200	0.70880700	0.00000000	4.28993000	0.61567600	0.68996100	0.75424700	0.00000000	6.93936000	33.90180390	318.4322
5	1990	32	4826235.06	2791367.52	181.66299440	0.28562100	0.32093200	0.35624400	0.00020461	4.80358000	0.53741900	0.59082200	0.64427500	0.00000032	4.90261000	0.69631800	0.76961200	0.84290700	0.00000000	6.39375000	6.67719440	151.9858
6	1990	32	4828443.74	2790705.98	154.84800510	0.28562100	0.32093200	0.35624400	0.00020461	4.80358000	0.53741900	0.59082200	0.64427500	0.00000032	4.90261000	0.69631800	0.76961200	0.84290700	0.00000000	6.39375000	2.35220510	152.5968
7	1990	32	4804640.29	2779843.11	194.58799740	0.39804000	0.44855020	0.49914400	0.00000000	3.94372000	0.64843900	0.72542600	0.80324100	0.00152162	4.63581000	1.13651000	1.25861000	1.38071000	0.00000000	7.94218000	1.41279740	193.1732
8	1990	32	4806608.82	2791775.54	180.05700680	0.12932000	0.14622600	0.16113300	0.00107796	4.08630000	0.39048800	0.42187700	0.45326700	0.00227820	3.99840000	0.52821700	0.57920300	0.63018900	0.00180410	3.44010000	8.20130680	170.8557
9	1990	32	4807596.78	2791543.32	172.08399960	0.12932000	0.14622600	0.16113300	0.00107796	4.08630000	0.39048800	0.42187700	0.45326700	0.00227820	3.99840000	0.52821700	0.57920300	0.63018900	0.00180410	3.44010000	2.59729960	169.4867
10	1990	32	4798271.83	2781857.46	201.38705040	0.22975400	0.26016800	0.29058200	0.00000000	3.71403000	0.63346600	0.70077400	0.76796200	0.00283400	4.72471000	0.86306200	0.95782000	1.05258000	0.00000000	6.88119000	4.19959440	197.1675
11	1990	32	4796749.31	2783991.29	197.68090730	0.22975400	0.26016800	0.29058200	0.00000000	3.71403000	0.63346600	0.70077400	0.76796200	0.00283400	4.72471000	0.86306200	0.95782000	1.05258000	0.00000000	6.88119000	1.49167790	196.2134
12	1990	32	4790296.00	2783041.68	198.675999510	0.22975400	0.26016800	0.29058200	0.00000000	3.71403000	0.63346600	0.70077400	0.76796200	0.00283400	4.72471000	0.86306200	0.95782000	1.05258000	0.00000000	6.88119000	2.87469510	196.7453
13	1990	32	4797772.15	2791546.70	200.004999910	0.22975400	0.26016800	0.29058200	0.00000000	3.71403000	0.63346600	0.70077400	0.76796200	0.00283400	4.72471000	0.86306200	0.95782000	1.05258000	0.00000000	6.88119000	6.27889910	200.1671
14	1990	32	4788344.28	2780778.90	200.004999910	0.22975400	0.26016800	0.29058200	0.00000000	3.71403000	0.63346600	0.70077400	0.76796200	0.00283400	4.72471000	0.86306200	0.95782000	1.05258000	0.00000000	6.88119000	-0.77440460	200.7826
15	1990	32	4797713.09	2780900.20	202.81399540	0.22975400	0.26016800	0.29058200	0.00000000	3.71403000	0.63346600	0.70077400	0.76796200	0.00283400	4.72471000	0.86306200	0.95782000	1.05258000	0.00000000	6.88119000	0.55069540	202.2833
16	1990	32	4797737.45	2780073.28	205.04100040	0.22975400	0.26016800	0.29058200	0.00000000	3.71403000	0.63346600	0.70077400	0.76796200	0.00283400	4.72471000	0.86306200	0.95782000	1.05258000	0.00000000	6.88119000	1.87570040	203.1658
17	1990	32	4796992.10	2782325.08	198.25999670	0.22975400	0.26016800	0.29058200	0.00000000	3.71403000	0.63346600	0.70077400	0.76796200	0.00283400	4.72471000	0.86306200	0.95782000	1.05258000	0.00000000	6.88119000	2.3099670	195.9520
18	1990	32	4812280.33	2795774.25	181.06300510	0.12758300	0.14428100	0.16088800	0.00412213	4.13037000	0.37952300	0.40977500	0.43997700	0.00819858	3.98779000	0.50562200	0.55484600	0.60366600	0.01536740	1.76022000	3.23790310	158.7335
19	1990	32	4812872.32	2795763.69	159.15400700	0.12758300	0.14428100	0.16088800	0.00412213	4.13037000	0.37952300	0.40977500	0.43997700	0.00819858	3.98779000	0.50562200	0.55484600	0.60366600	0.01536740	1.76022000	1.48710700	157.6569
20	1990	32	4781075.37	2762363.90	302.17001340	0.13200600	0.14724800	0.16249100	0.00000000	2.91161000	0.82218800	0.89505800	0.96792300	0.00000000	4.82500000	0.67351300	0.76002500	0.84655600	0.00000000	6.26959000	26.36591340	275.8041
21	1990	32	4811934.83	2787982.32	175.09399410	0.21871000	0.24727400	0.27583900	0.00000000	4.12674000	0.35709100	0.60870700	0.66632400	0.01059540	4.45051000	0.76303000	0.84079100	0.91733100	0.00000000	6.32806000	4.62079410	171.0732
22	1990	32	4808173.83	2786998.77	174.97900390	0.20399900	0.23075600	0.25752500	0.00000000	4.06996000	0.34423900	0.59479800	0.64471600	0.00293457	4.42656000	0.73167000	0.80759900	0.88191000	0.00000000	6.20271000	-0.21399610	175.1930
23	1990	32	4804176.72	2780344.84	184.43200480	0.18613400	0.21096100	0.23599900	0.00000000	3.95959000	0.54891800	0.60155100	0.65323700	0.00081170	4.42308000	0.72089400	0.79448900	0.86952100	0.00000000	6.17906000	4.49570440	179.0760
24	1990	32	4788910.46	2778911.78	244.23500060	0.18136000	0.19190800	0.21478600	0.00000000	3.90223000	0.67679900	0.74888400	0.82099100	0.01123950	4.87022000	0.71708400	0.80310800	0.89391100	0.00000000	6.88044000	5.23930960	238.9957
25	1990	32	4790638.32	2795501.38	172.55000490	0.07391940	0.08451790	0.09411570	0.00238465	3.79732000	0.43681000	0.48007900	0.52321400	0.00631572	4.38940000	0.45400400	0.50681800	0.55772300	0.00457900	5.18475000	4.95180450	167.6374
26	1990	32	4790962.03	2796498.70	169.25999780	0.07391940	0.08451790	0.09411570	0.00238465	3.79732000	0.43681000	0.48007900	0.52321400	0.00631572	4.38940000	0.45400400	0.50681800	0.55772300	0.00457900	5.18475000	4.48849780	164.7675

Εικόνα 32: Εισαγωγή δεδομένων στο MATLAB για εκπαίδευση νευρωνικών δικτύων

Στην συνέχεια ο χρήστης έχει δύο επιλογές, είτε θα χρησιμοποιήσει την εντολή *nftool* η οποία θα του δώσει κάποιες βασικές επιλογές για ρηχά νευρωνικά δίκτυα ή θα χρησιμοποιήσει την εντολή *nntool*, η οποία θα δώσει στην χρήση την δυνατότητα να μπορεί να πραγματοποιήσει αρκετές περισσότερες επιλογές όσον αφορά την δημιουργία αλλά και την λειτουργία του νευρωνικού δικτύου. Οι διαφορές μεταξύ των δύο εργαλείων συνοψίζονται στον Πίνακα 2.

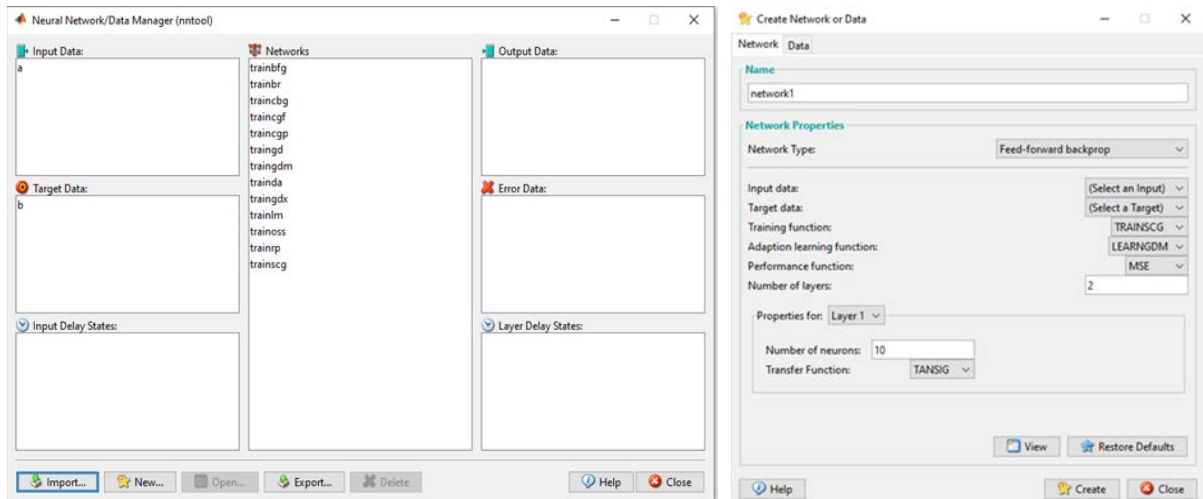
Πίνακας 2: Διαφορές μεταξύ των δύο εργαλείων του υπολογιστικού περιβάλλοντος της MATLAB

	nntool	nftool
Επιλογή αριθμού κόμβων	✓	✓
Επιλογή αριθμού επιπέδων	✓	✗
Επιλογή συνάρτησης ενεργοποίησης	✓	✗
Επιλογή ποσοστού υποσυνόλων	✗	✓
Τροποποίηση παραμέτρων αλγορίθμου	✓	✗
Διαθέσιμοι αλγόριθμοι για εκπαίδευση	14	3

ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΕΡΓΑΛΕΙΟΥ NNTOOL

Το μοναδικό πλεονέκτημα του *nnftool*, δεν θα είχε πρακτική χρήση, αφού το πλήθος των δεδομένων ήταν τέτοιο που επαρκούσε τόσο για την εκπαίδευση όσο και τον έλεγχο των αποτελεσμάτων. Για το λόγο αυτό, παρόλο που κάποια σενάρια

Η αρχική εικόνα του εργαλείου *nnftool*, είναι αρκετά λιτή. Ο χρήστης έχει την δυνατότητα να εισάγει τα δεδομένα σε μορφή πινάκων, να δημιουργήσει νέα ή να εισάγει ήδη εκπαιδευμένα δίκτυα καθώς και να εξάγει τα αποτελέσματα σε αρχεία, η στο κεντρικό παράθυρο του *MATLAB* για περεταίρω επεξεργασία (Εικόνα 33).



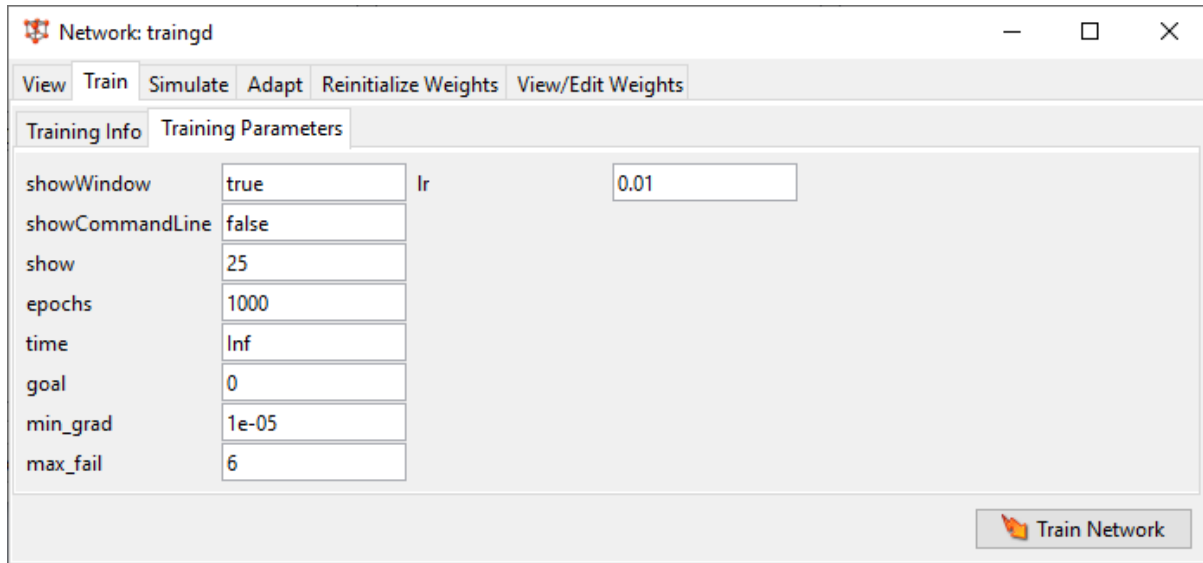
Εικόνα 33: Αριστερά εμφανίζεται το κεντρικό μενού του εργαλείου *nnftool*, δεξιά εμφανίζονται οι επιλογές δημιουργίας νέου Τ.Ν.Δ.

Στην δημιουργία νέου δικτύου, ο χρήστης έχει διάφορες δυνατότητες, όπως η επιλογή του είδους του δικτύου (υπάρχει μια ποικιλία τύπων Τ.Ν.Δ., στην παρούσα εργασία επιλέχθηκαν αποκλειστικά δίκτυα εμπροσθοδιάδοσης), τον αλγόριθμο εκπαίδευσης του δικτύου (θα αναφερθούν ξεχωριστά οι επιλογές στην συνέχεια), τον τρόπο υπολογισμού της απόδοσης (δίνεται η δυνατότητα υπολογισμού του συνολικού τετραγωνικού σφάλματος (*SSE*), η επιλογή του μέσου τετραγωνικού σφάλματος (*MSE*) και η επιλογή του αθροίσματος του μέσου τετραγωνικού σφάλματος με το μέσο τετράγωνο των βαρών του συστήματος (*MSEREG*)). Στην εργασία αυτή σε όλα τα νευρωνικά δίκτυα, επιλέχθηκε το *MSE*, ώστε να μπορούν να συγκριθούν διαφορετικές αρχιτεκτονικές και επιλογές διανυσμάτων εισόδου μεταξύ τους καθώς και για να μπορεί να γίνει σύγκριση των αποτελεσμάτων με προηγούμενες μελέτες, που έχουν γίνει πάνω στο ίδιο πρόβλημα. Άλλες παράμετροι που μπορούν να καθοριστούν από τον χρήστη είναι ο αριθμός των επιπέδων και των κόμβων του Τ.Ν.Δ.. Τέλος, ο χρήστης μπορεί να διαλέξει μεταξύ τριών συναρτήσεων ενεργοποίησης, της λογιστικής, της συνάρτησης υπερεφαπτομένης καθώς και της γραμμικής συνάρτησης ενεργοποίησης. Για λόγους συμβατότητας με όλες τις μεθόδους, καθώς και επειδή προσφέρει καλύτερη σύγκλιση σε σχέση με την λογιστική, επιλέχθηκε η συνάρτηση της υπερεφαπτομένης.

ΠΑΡΑΜΕΤΡΟΠΟΙΗΣΗ ΑΛΓΟΡΙΘΜΩΝ

Το *MATLAB*, μέσω του εργαλείου του *nntool*, δίνει την δυνατότητα χρήσης 14 διαφορετικών αλγόριθμων εκπαίδευσης Τ.Ν.Δ. Στην παρούσα εργασία χρησιμοποιήθηκαν οι 13 από αυτούς. Ο μόνος αλγόριθμος που δεν χρησιμοποιήθηκε, ήταν αυτός του τυχαίου περιπάτου, καθώς είναι εξαιρετικά χρονοβόρος και χωρίς τα αντίστοιχα αποτελέσματα.

Αλγόριθμος απότομης καθόδου

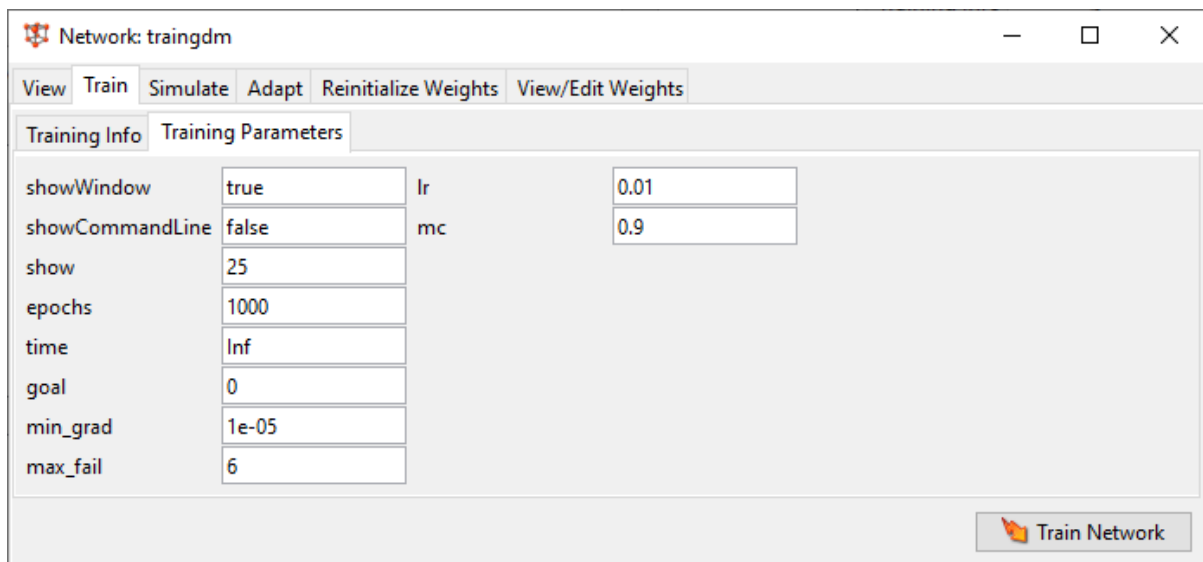


Εικόνα 34: Παραμετροποίηση της μεθόδου της απότομης καθόδου

Η μέθοδος της απότομης καθόδου, είναι μια απλή μέθοδος και ως εκ τούτου, δεν δίνονται πολλές επιλογές διαμόρφωσης της μεθόδου. Αρχικά δίνονται οι επιλογές τερματισμού της εκπαίδευσης. Ο χρήστης έχει την δυνατότητα να επιλέξει ένα μέγιστο όριο εποχών (προεπιλογή είναι οι χίλιες εποχές), το μέγιστο χρονικό διάστημα που θα τρέχει ο αλγόριθμος εκπαίδευσης (η επιλογή αυτή είναι απενεργοποιημένη αρχικά), ο στόχος σύγκλισης (αρχικός στόχος είναι η ταύτιση των αποτελεσμάτων), την ελάχιστη κλίση για την οποία θα τρέχει ο αλγόριθμος καθώς και τα λάθη επαλήθευσης. Η μοναδική υπερπαράμετρος που ο χρήστης έχει επιλογή να μεταβάλει είναι ο ρυθμός μάθησης, με προεπιλεγμένη τιμή το 0.01 (Εικόνα 34).

Αλγόριθμος απότομης καθόδου με ορμή

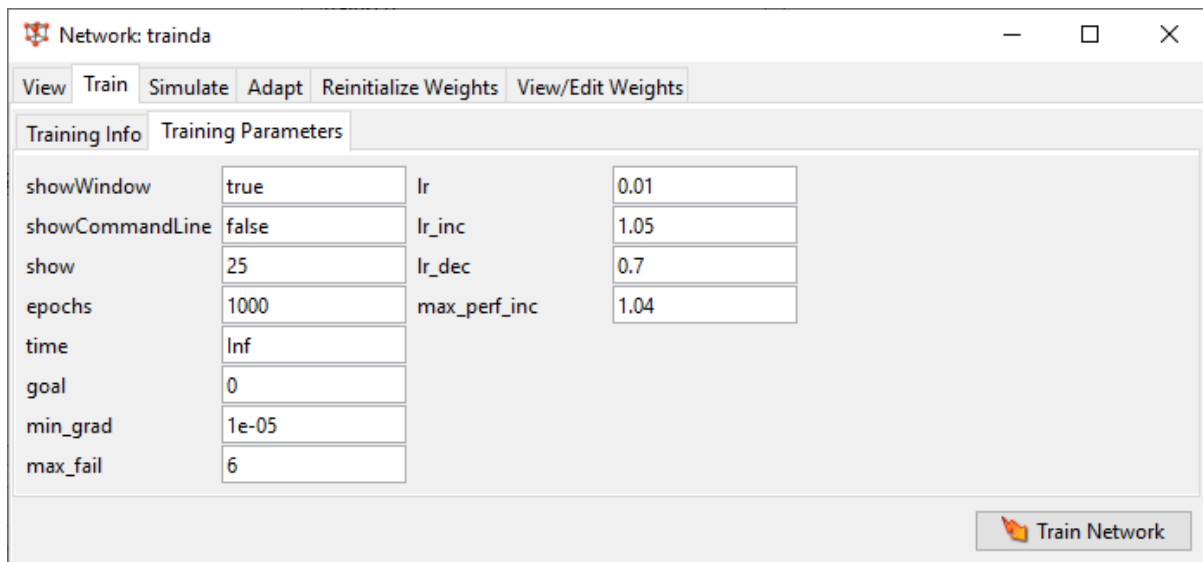
Όπως είναι γνωστό από την θεωρία, δεν υπάρχουν μεγάλες διαφορές στην μέθοδο αυτή με την μέθοδο της απότομης καθόδου, για αυτό το λόγο η μοναδική τους διαφορά στις επιλογές παραμετροποίησης, είναι η δυνατότητα επιλογής του συντελεστή «τριβής», ο οποίος σαν προεπιλογή έχει την τιμή 0.9, που όπως έχει αναφερθεί στην θεωρία, είναι και η συνιστώμενη τιμή στην διεθνή βιβλιογραφία (Εικόνα 35).



Εικόνα 35: Παραμετροποίηση της μεθόδου της απότομης καθόδου με ορμή

Αλγόριθμος απότομης καθόδου με μεταβλητό ρυθμό μάθησης

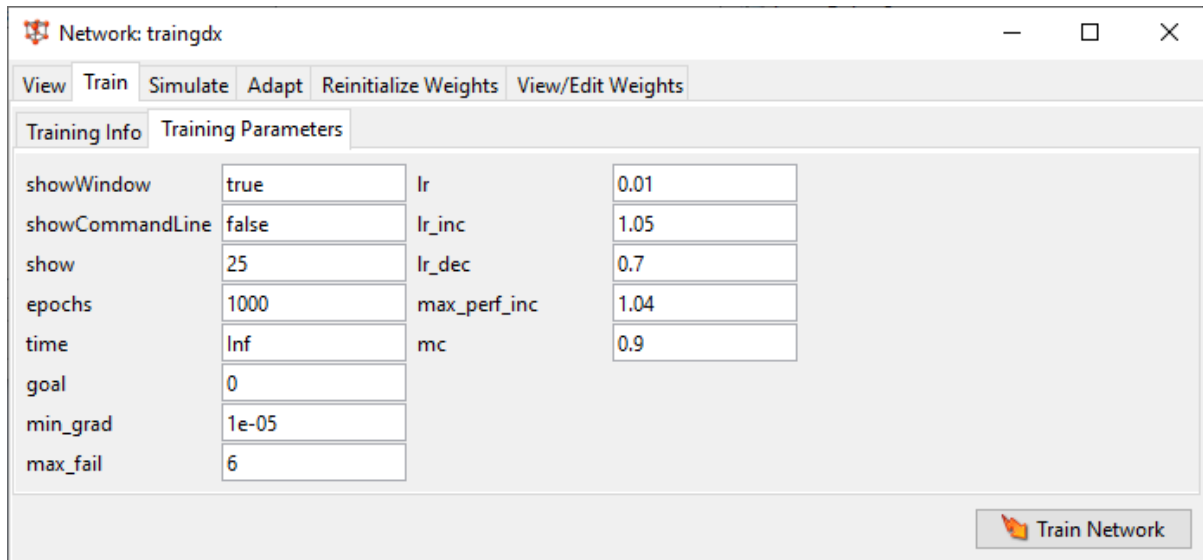
Εκτός των κριτηρίων τερματισμού και του ρυθμού μάθησης, στην συγκεκριμένη μέθοδο δίνετε η δυνατότητα καθορισμού των τριών υπερπαραμέτρων που αναφέρονται στο θεωρητικό κομμάτι της μεθόδου αυτής. Οι προεπιλεγμένες τιμές, είναι οι τυπικές τιμές που δίνονται στην διεθνή βιβλιογραφία, ο χρήστης μπορεί να αλλάξει την σταθερά μείωσης (lr_dec) και να επιλέξει μια άλλη τιμή από το διάστημα (0,1), να αυξήσει την σταθερά αύξησης (lr_inc) ή να αλλάξει το μέγιστο ποσοστό στο οποίο δεν γίνεται μείωση του ρυθμού μάθησης (max_perf_inc) (Εικόνα 36).



Εικόνα 36: Παραμετροποίηση της μεθόδου της απότομης καθόδου με μεταβλητό ρυθμό μάθησης

Αλγόριθμος απότομης καθόδου με μεταβλητό ρυθμό μάθησης και ορμή

Η μέθοδος αυτή, είναι συνδυασμός των δύο προηγούμενων μεθόδων, επομένως είναι φυσικό επακόλουθο να δίνεται δυνατότητα τροποποίησης των υπερπαραμέτρων που εμφανίζονται σε αυτές, με τις ίδιες προεπιλεγμένες τιμές (Εικόνα 37).



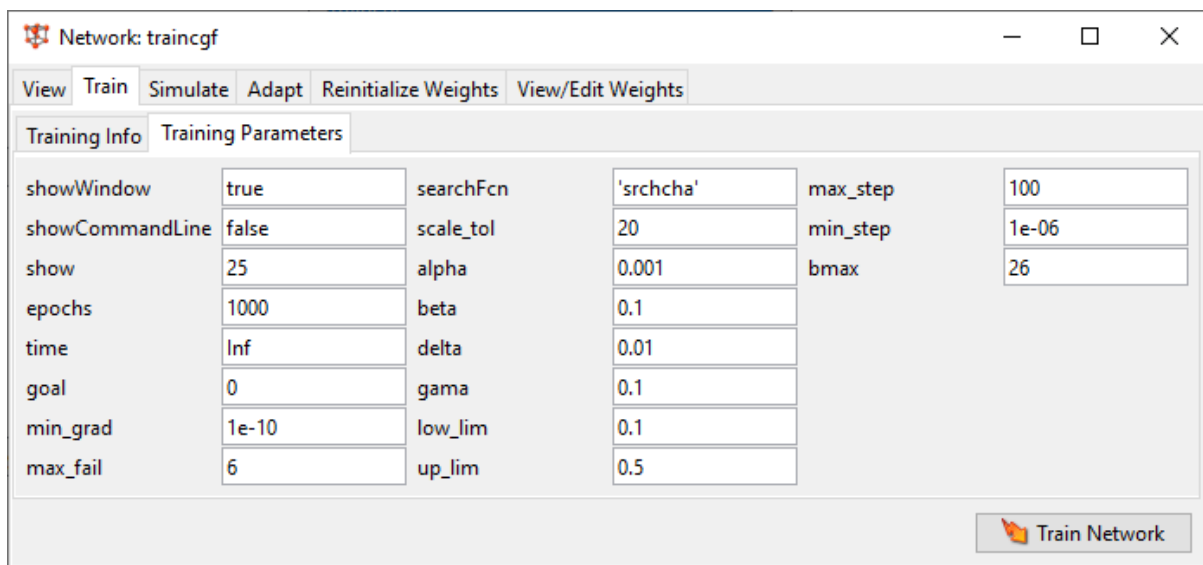
Εικόνα 37: Παραμετροποίηση της μεθόδου της απότομης καθόδου με μεταβλητό ρυθμό μάθησης

Αλγόριθμος συζυγούς κλίσης με επανεκκινήσεις

Αυτές οι μέθοδοι ανήκουν σε διαφορετική οικογένεια, από τις προηγούμενες τέσσερις. Αρχικά αυτό που παρατηρείτε είναι ότι λόγω της χρήσης 2^{ης} παραγώγου μπορεί να τρέξει και αποδοτικά και σε προβλήματα με μικρή κλίση, για αυτό και στα κριτήρια τερματισμού η προεπιλεγμένη ελάχιστη κλίση είναι πέντε τάξεις μεγέθους πιο μικρή.

Όπως αναφέρθηκε και στο θεωρητικό μέρος, ανάλογα τον τρόπο υπολογισμού της βαθμίδας β_k , προκύπτει και διαφορετικός αλγόριθμος εκπαίδευσης. Το περιβάλλον του *MATLAB* έχει συμπεριλάβει τρεις διαφορετικούς αλγόριθμους (μέθοδος συζυγούς κλίσης με επανεκκινήσεις τύπου *Fletcher-Reeves*, μέθοδος συζυγούς κλίσης με επανεκκινήσεις τύπου *Polak-Ribiere* και μέθοδος συζυγούς κλίσης με επανεκκινήσεις τύπου *Beale-Powell*). Και στις τρεις περιπτώσεις δίνεται το ίδιο παράθυρο παραμετροποίησης (Εικόνα 38). Όπως είναι γνωστό από την θεωρία, και οι τρεις δεν έχουν κάποια υπερπαραμέτρο ενσωματωμένη, αλλά απαιτούν την χρήση κάποιας συνάρτησης αναζήτησης. Η προεπιλεγμένη συνάρτηση είναι η αναζήτηση *Charalambous* (Charalambous, 1992), η οποία έχει αποδειχθεί πιο αποδοτική σε συνδυασμό με τις μεθόδους συζυγούς κλίσης, σε σχέση με υπόλοιπες τέσσερις που δίνονται ως επιλογές (Beale, Hagan, & Demuth, 2017), οι οποίες είναι η αναζήτηση προς τα πίσω (*backtracking*) (Dennis & Schnabel, 1983), η αναζήτηση με χρήση υβριδικής μεθόδου διχοτόμησης με κυβική παρεμβολή (Scales, 1985), η αναζήτηση χρυσού τμήματος (Hagan, et al., 2014) και η αναζήτηση *Brent* (Brent, 1973). Πέρα από αυτές τις επιλογές, ο χρήστης έχει την δυνατότητα να χρησιμοποιήσει κάποια δική του συνάρτηση γραμμικής αναζήτησης. Οι υπόλοιπες παράμετροι, επηρεάζουν την λειτουργία της εκάστοτε συνάρτησης έρευνας. Δεν χρησιμοποιούνται όλες κάθε φορά, αλλά ανάλογα την μέθοδο αναζήτησης θα χρησιμοποιηθούν κάποιες (ή όλες) από αυτές. Η παράμετρος *delta*, είναι το αρχικό βήμα. Η

παράμετρος *scale_tol*, συνδυάζει την ανοχή ακρίβειας με το αρχικό βήμα *delta*. Η παράμετρος *alpha*, είναι συντελεστή κλίμακας, ο οποίος καθορίζει ότι θα υπάρχει επαρκής μείωση του σφάλματος. Η παράμετρος *beta*, είναι συντελεστής κλίμακας, ο οποίος καθορίζει την ύπαρξη αρκετά μεγάλου μέγεθος βήματος. Η παράμετρος *gama*, φροντίζει ώστε να μην υπάρχουν μικρές πτώσεις στην απόδοση και η προτεινόμενη τιμή της από την βιβλιογραφία, είναι το 0.1, που είναι και η τιμή που έχει οριστεί και ως προεπιλογή. Οι παράμετροι *low_lim* και *up_lim*, θέτουν κάποια ανώτερα και κατώτατα όρια στην αλλαγή του βήματος την ώρα που οι παράμετροι *max_step* και *min_step* θέτουν κάποια όρια στο πεδίο τιμών του βήματος. Και η μεταβλητή *bmax* θέτει μια μέγιστη τιμή βήματος. Η αναζήτηση Charalambous, χρησιμοποιεί μόνο τις μεταβλητές *alpha*, *beta*, *gama*, *delta* και *scale_tol* (Beale, Hagan, & Demuth, 2017),



Εικόνα 38: Παράθυρο με τις επιλογές παραμετροποίησης των μεθόδων συζυγούς κλίσης με επανεκκινήσεις. Και οι τρεις μέθοδοι, εμφανίζουν το ίδιο μενού. Στην εικόνα εμφανίζεται το παράθυρο επιλογών της μεθόδου συζυγούς κλίσης με επανεκκινήσεις τύπου Fletcher-Reeves.

Αλγόριθμοι της οικογένειας Newton

Από την κατηγορία αυτή, το MATLAB προσφέρει δύο επιλογές, την μέθοδο *BFGS-Quasi-Newton* και την *One-Step Secant*. Οι δύο μέθοδοι από την φύση τους, δεν έχουν κάποια υπερπαράμετρο, αλλά και αυτές, όπως και οι μέθοδοι συζυγούς κλίσης, έχουν την ανάγκη χρήσης μιας επιπλέον συνάρτησης αναζήτησης. Οι ίδιες επιλογές είναι διαθέσιμες και σε αυτήν την περίπτωση, με την μόνη διαφοροποίηση να είναι ότι η προεπιλεγμένη συνάρτηση αναζήτησης είναι η αναζήτηση προς τα πίσω, αφού είναι η πιο κατάλληλη για χρήση με της μεθόδους *Quasi-Newton* (Beale, Hagan, & Demuth, 2017). Εκτός των παραμέτρων που αφορούν την συνάρτηση αναζήτησης, στις επιλογές της *BFGS-Quasi-Newton*, δίνεται η δυνατότητα να κατακερματιστούν τα δεδομένα σε παρτίδες. Αυτό γίνεται με την προσθήκη μιας παραμέτρου με την ονομασία *batch_frag*. Όταν η μεταβλητή αυτή έχει την τιμή μηδέν, δεν υπάρχει επίδραση στο σύστημα. Εάν όμως της δοθεί κάποια μη μηδενική τιμή, τότε τα δεδομένα χωρίζονται σε παρτίδες και οι συνθήκες του τελικού επιπέδου μιας προηγούμενης εποχής, χρησιμοποιούνται ως αρχικές συνθήκες για την επόμενη εποχή (Εικόνα 39) (Beale, Hagan, & Demuth, 2017).

Network: trainbfg

View Train Simulate Adapt Reinitialize Weights View/Edit Weights

Training Info Training Parameters

showWindow	true	searchFcn	'srchbac'	max_step	100
showCommandLine	false	scale_tol	20	min_step	1e-06
show	25	alpha	0.001	bmax	26
epochs	1000	beta	0.1	batch_frag	0
time	Inf	delta	0.01		
goal	0	gamma	0.1		
min_grad	1e-06	low_lim	0.1		
max_fail	6	up_lim	0.5		

Train Network

Εικόνα 39: Παράθυρο με τις επιλογές παραμετροποίησης των μεθόδων της οικογένειας Quasi-Newton. Οι επιλογές είναι ίδιες με αυτές που έχουν οι μέθοδοι συζυγούς κλίσης με επανεκκινήσεις, με εξαίρεση την χρήση διαφορετικής συνάρτησης αναζήτησης και την ύπαρξη της παραμέτρου *batch_frag*. Στην εικόνα εμφανίζεται το παράθυρο επιλογών της μεθόδου BFGS-Quasi-Newton

Αλγόριθμος Κλιμακωτής Συζυγούς Κλίσης

Network: trainscg

View Train Simulate Adapt Reinitialize Weights View/Edit Weights

Training Info Training Parameters

showWindow	true	sigma	5e-05
showCommandLine	false	lambda	5e-07
show	25		
epochs	1000		
time	Inf		
goal	0		
min_grad	1e-06		
max_fail	6		

Train Network

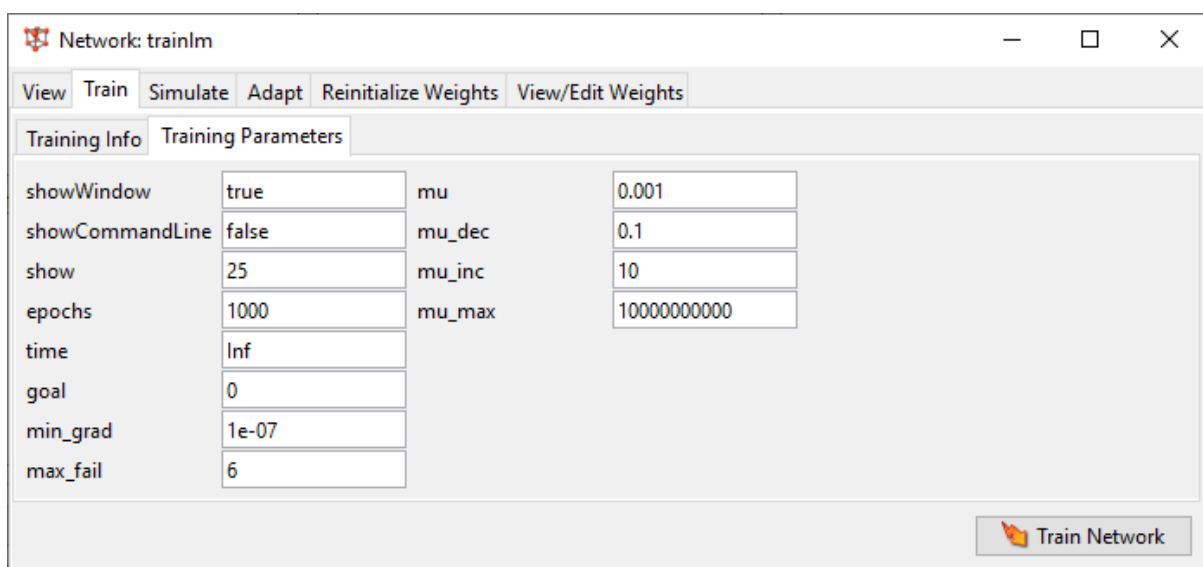
Εικόνα 40: Παραμετροποίηση της μεθόδου Κλιμακωτής Συζυγούς Κλίσης

Η μέθοδος αυτή όπως φαίνεται και στην εξίσωση 69, έχει δύο παραμέτρους που οι αρχικές τους τιμές πρέπει να καθοριστούν πριν την εκπαίδευση, με το *sigma* να μην αλλάζει τιμές κατά την διάρκεια της εκπαίδευσης, την ώρα που το *lambda* (γνωστό και ως πολλαπλασιαστής *Lagrange*) υπολογίζεται εκ νέου σε κάθε επανάληψη (Εικόνα 40) (Møller, 1993). Το *sigma* εξ ορισμού θα πρέπει να έχει πολύ μικρότερη τιμή από τη μονάδα, εξ ου και η προεπιλεγμένη τιμή του 10^{-5} , τιμή που απαντάται και στην βιβλιογραφία (Chel, Majumder, & Nandi, 2011), η τιμή του *lambda* θα πρέπει να είναι επίσης αρκετά μικρότερη από τη μονάδα αν και συστήνονται αρκετά μικρότερες τιμές ($0 < \lambda < 10^{-6}$) (Møller, 1993).

Αλγόριθμοι Levenberg-Marquardt

Ο αλγόριθμος *Levenberg-Marquardt*, είναι από τους πιο ευρέως χρησιμοποιούμενους στην μηχανική μάθηση καθώς συνδυάζει ταχύτητα και επιδόσεις. Έχει όμως ένα μεγάλο μειονέκτημα καθώς επιβαρύνει την μνήμη RAM, του υπολογιστή. Λόγω όμως της αποτελεσματικότητάς του, στο MATLAB δίνονται δύο εκδοχές του, μια που χρησιμοποιεί την μέθοδο της πρόωρης διακοπής και φέρει το όνομα του αλγορίθμου και μία δεύτερη η οποία χρησιμοποιεί μια μέθοδο κανονικοποίησης L2 και ονομάζεται *Bayesian Regularization*.

Και οι δύο μέθοδοι έχουν το ίδιο παράθυρο παραμετροποίησης (Εικόνα 41), αν και δεν χρησιμοποιούνται όλες οι παράμετροι και από τις δύο μεθόδους, καθώς λόγω χάρη, η *Bayesian Regularization*, δεν χρησιμοποιεί την μέθοδο της πρόωρης διακοπής, επομένως δεν υπάρχει περίπτωση να σταματήσει στην περίπτωση που το T.N.Δ. θα κάνει υπερεκπαίδευση. Το γεγονός αυτό ουσιαστικά εξουδετερώνει την παράμετρο του *max_fail*.



Εικόνα 41: Παραμετροποίηση των μεθόδων *Levenberg-Marquardt*. Το μενού είναι ακριβώς ίδιο και στις δύο εκδοχές του αλγορίθμου, παρόλο που δεν χρησιμοποιούνται και όλες οι παράμετροι και από τις δύο εκδοχές. Στην εικόνα εμφανίζεται το παράθυρο επιλογών του αλγορίθμου που φέρει την ονομασία της μεθόδου.

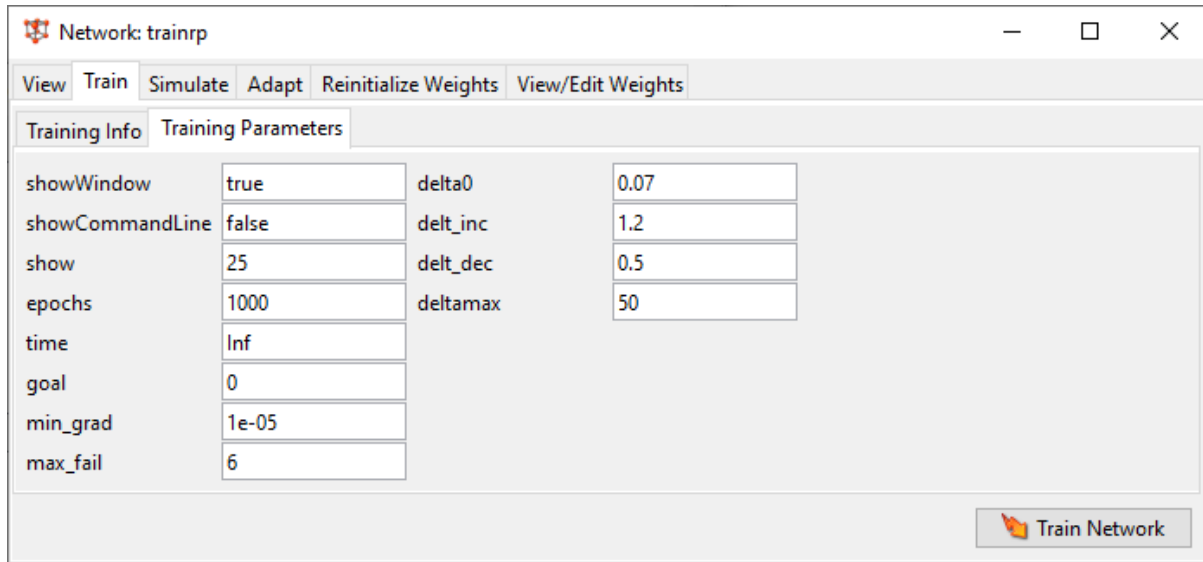
Οι παράμετροι που υπάρχουν στη δεύτερη στήλη, είναι το ν που περιγράφηκε στην εξίσωση 55. Όπως αναφέρθηκε και στο θεωρητικό κομμάτι, αρχικά θα πρέπει να δοθεί μια πολύ μικρή τιμή, ώστε ο αλγόριθμος να πηγαίνει προς την συμπεριφορά της μεθόδου της απότομης καθόδου, για αυτό και η προεπιλεγμένη τιμή του μ είναι το 0.001. Ενώ οι συντελεστές μ_{dec} και μ_{inc} παίζουν τον ρόλο της παραμέτρου ρ , με την διαφορά να είναι ότι στην περιγραφή της μεθόδου χρησιμοποιούνταν το ρ και το $1/\rho$, την ώρα που στον αλγόριθμο δεν φαίνεται να υπάρχει αυτός ο περιορισμός. Τέλος, προστίθεται ως κριτήριο τερματισμού η περίπτωση που το ρ ξεπεράσει κάποιο όριο, με την χρήση της παραμέτρου μ_{max} .

Αλγόριθμος Ανθεκτικής Διάδοσης

Το MATLAB, ενσωματώνει αυτή την μέθοδο, στον αλγόριθμο με την ονομασία *RProp*. Η μέθοδος αυτή, δεν χρησιμοποιεί 2^η παράγωγο για αυτό και είναι και πιο γρήγορη από τις υπόλοιπες μεθόδους. Αυτό όμως την κάνει πιο ευαίσθητη στις μικρές κλίσεις. Γεγονός που

αποτυπώνεται και στα προεπιλεγμένα κριτήρια τερματισμού της, όπου έχει προεπιλεχθεί αντίστοιχο όριο με αυτό που είχαν οι μέθοδοι απότομης καθόδου.

Όπως αναφέρθηκε και στην θεωρία, η μέθοδος αυτή έχει τρεις παραμέτρους, οι οποίες θα πρέπει να καθοριστούν πριν την έναρξη της. Αυτές είναι μια αρχική τιμή για το *delta*, όπου η προεπιλογή είναι αρκετά κοντά στην τιμή που αναγράφεται και στην βιβλιογραφία. Ακολουθούν οι παράμετροι η^+ και η^- , που στον αλγόριθμο αναφέρονται ως *delta_inc* και *delta_dec*, αντίστοιχα. Τέλος η παράμετρος *deltamax*, θέτει ένα ανώτατο όριο για το *delta*, ώστε να μην γίνονται μεγάλα άλματα. Απουσιάζει όμως κάποια επιλογή για ελάχιστη τιμή της παραμέτρου (Εικόνα 42).

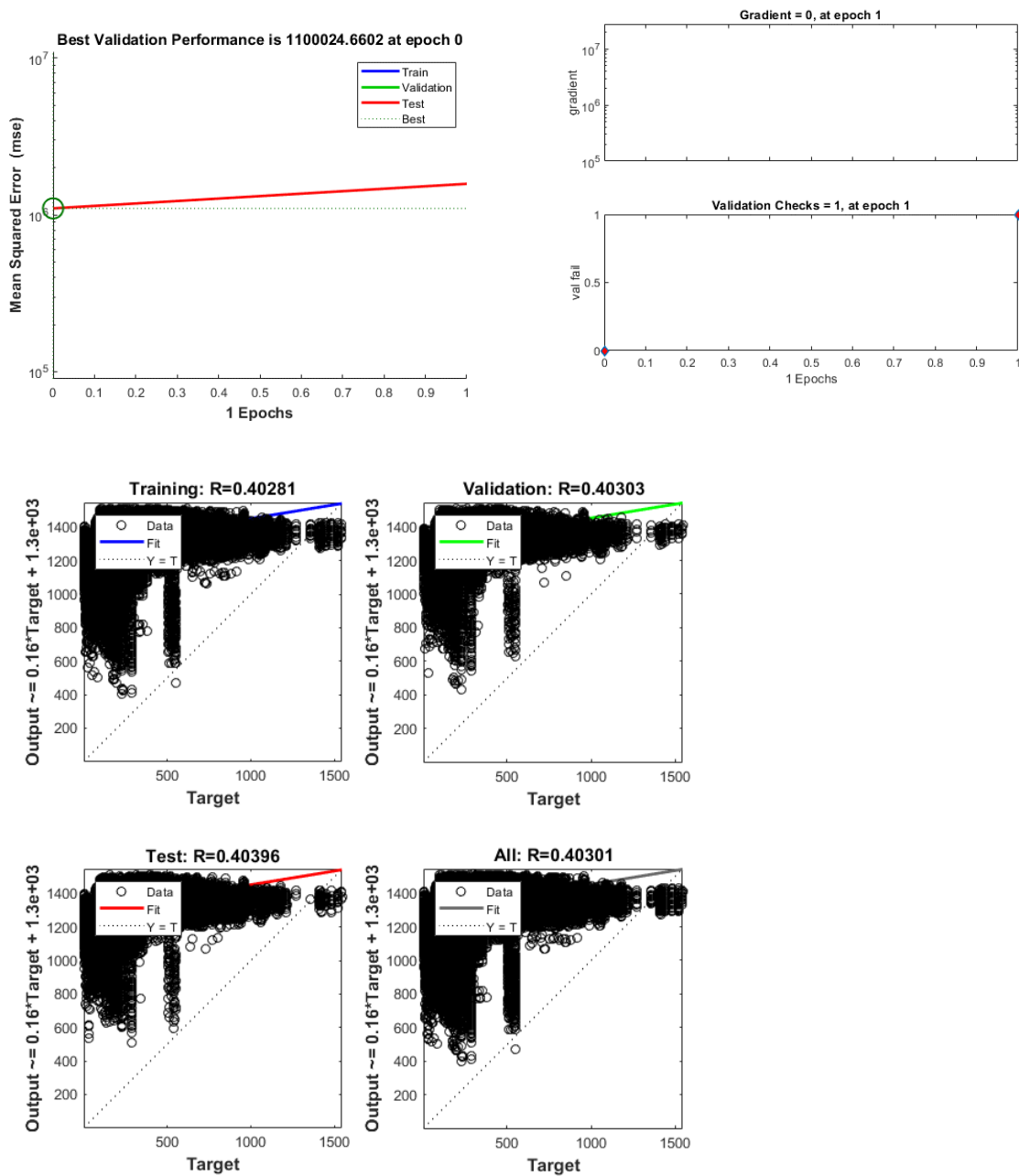


Εικόνα 42: Παραμετροποίηση της μεθόδου Ανθεκτικής Διάδοσης

ΔΗΜΙΟΥΡΓΙΑ ΡΗΧΩΝ Τ.Ν.Δ

Αρχικά δημιουργήθηκαν διαφορά ρηχά Τ.Ν.Δ., για να ελεγχθεί η αποτελεσματικότητα κάθε αλγόριθμου. Προκειμένου να γίνει έλεγχος όλων των διαθέσιμων αλγόριθμων και να αποφευχθούν φαινόμενα υπερεκπαίδευσης, τα Τ.Ν.Δ. που δημιουργήθηκαν περιορίστηκαν στους 20 κόμβους. Ως συνάρτηση ενεργοποίησης επιλέχθηκε η συνάρτηση υπερεφαπτομένης καθώς έχει καλύτερη σύγκλιση από την λογιστική, ενώ είναι εύκολο να υπολογιστεί και η δεύτερη παράγωγος της. Από τις μεταβλητές που ήταν διαθέσιμες για το διάνυσμα εισόδου, δεν χρησιμοποιήθηκε το υψόμετρο, αφού θα μπορούσε το νευρωνικό να εστιάσει την εκπαίδευση του με βάση αυτό και να μην υπολογίσει τις υπόλοιπες παραμέτρους. Στην παρουσίαση των αποτελεσμάτων λαμβάνεται υπόψιν και ο απαιτούμενος χρόνος για την εκπαίδευση τους, παρόλο που δυστυχώς αυτός δεν εμφανίζεται στα διαγράμματα. Θα μπορούσαν να ενσωματωθούν τα αντίστοιχα στιγμιότυπα οθόνης, αλλά αυτό θα αύξανε σημαντικά τον όγκο της παρούσας εργασίας. Για το λόγο αυτό προτιμήθηκε οι χρόνοι να μουν συγκεντρωτικά στον Πίνακα 3.

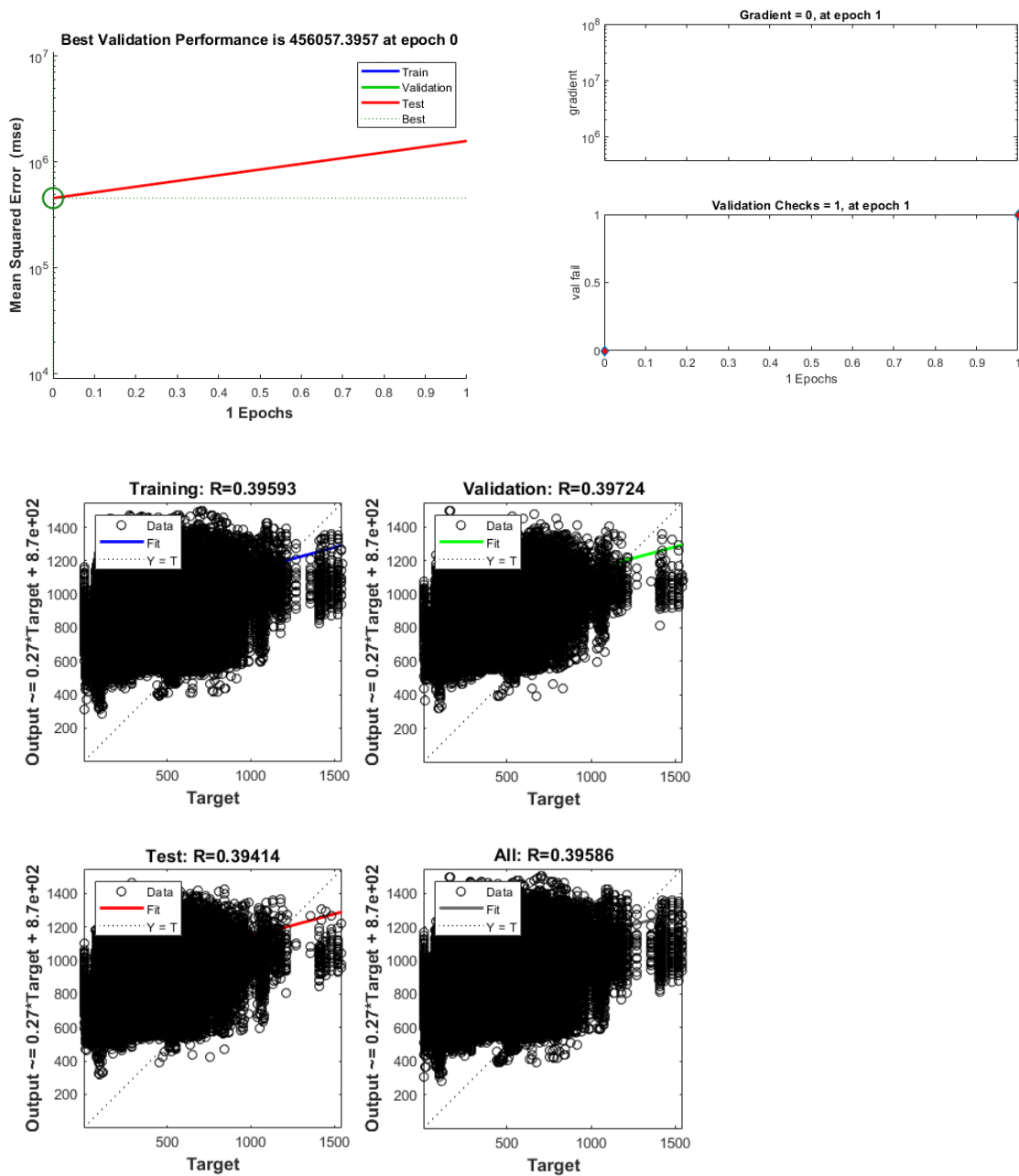
Αλγόριθμος απότομης καθόδου



Διάγραμμα 1: Αποτελέσματα του αλγόριθμου απότομης καθόδου

Η μέθοδος της απότομης καθόδου είναι η πιο απλή μέθοδος και θα ήταν έκπληξη εάν θα μπορούσε να δώσει κάποιο αποτέλεσμα. Μόλις στην 1^η εποχή η παράγωγος μηδενίστηκε και το σύστημα εγκλωβίστηκε σε κάποιο τοπικό μέγιστο (μηδενική παράγωγος). Η εκπαίδευση τερματίστηκε καθώς αυτό αποτελούσε κριτήριο τερματισμού. Καθώς η πρόβλεψη στην 1^η εποχή ήταν χειρότερη από την αρχική, καλύτερη πρόβλεψη του μοντέλου, ήταν τα τυχαία βάρη που είχαν δοθεί ως αρχικές συνθήκες (Διάγραμμα 1).

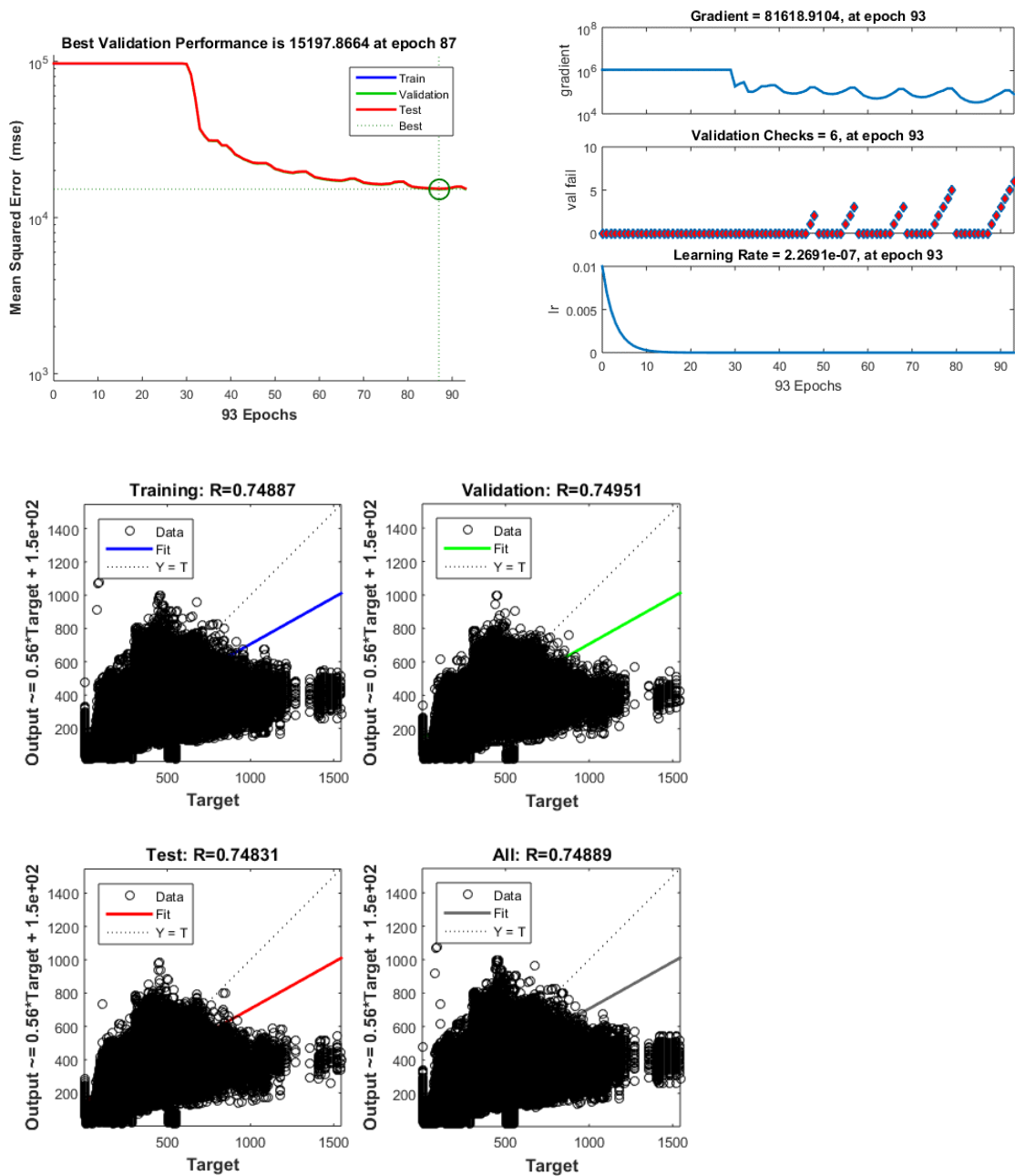
Αλγόριθμος απότομης καθόδου με ορμή



Διάγραμμα 2: Αποτελέσματα του αλγόριθμου απότομης καθόδου με ορμή

Αντίστοιχα προβλήματα παρουσιάζει και η μέθοδος της απότομης καθόδου με ορμή. Και σε αυτήν την περίπτωση το σύστημα εγκλωβίστηκε σε κάποιο τοπικό μέγιστο, αδυνατώντας να συνεχίσει την εκπαίδευση (Διάγραμμα 2).

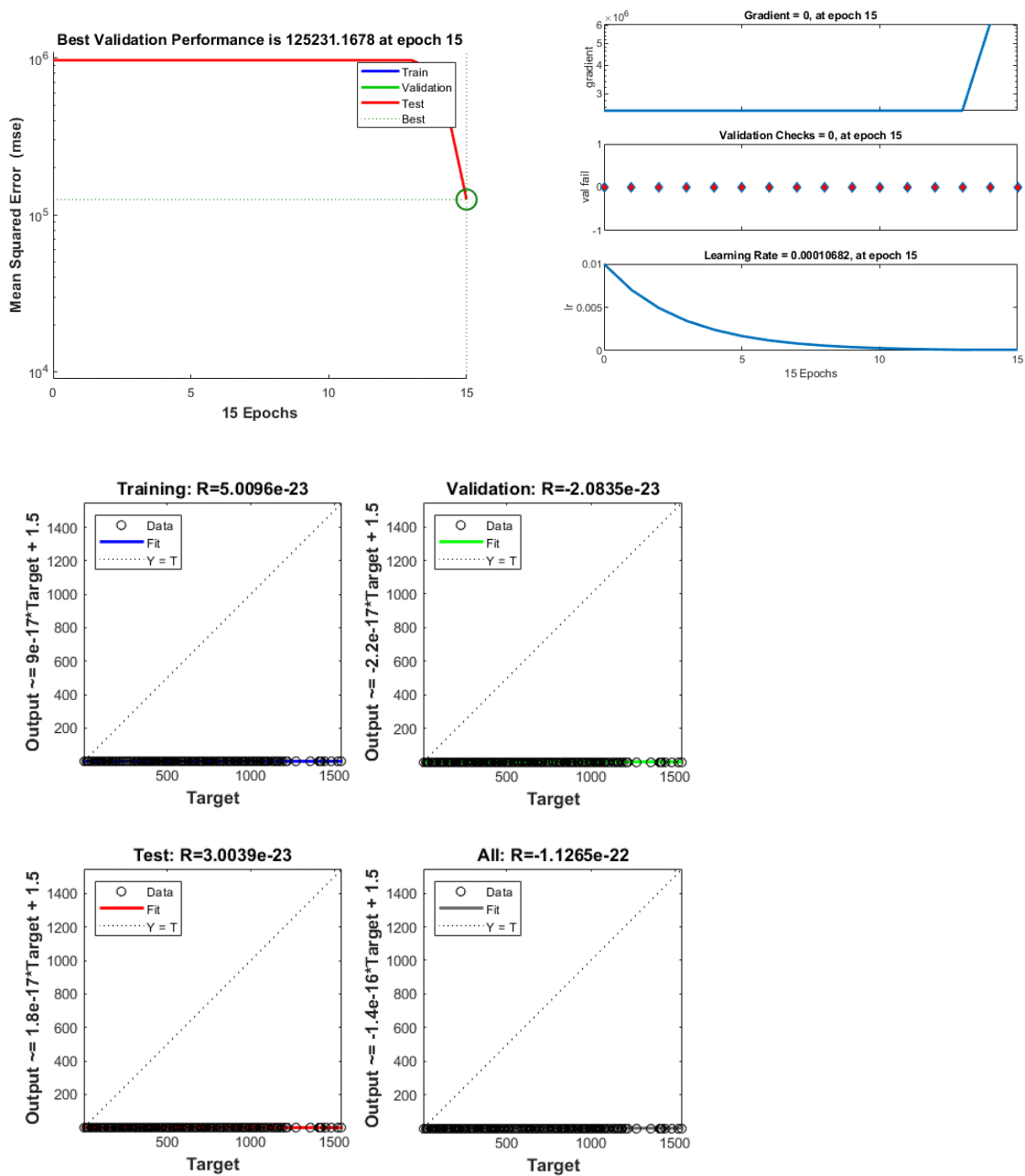
Αλγόριθμος απότομης καθόδου με μεταβλητό ρυθμό μάθησης



Διάγραμμα 3: Αποτελέσματα του αλγόριθμου απότομης καθόδου με μεταβλητό ρυθμό μάθησης

Σε αυτήν την περίπτωση το σύστημα δεν βρήκε κάποιο τοπικό μέγιστο τις πρώτες εποχές με αποτέλεσμα να πραγματοποιήσει κάποια εκπαίδευση των νευρώνων. Λόγω του μεταβλητού ρυθμού μάθησης κατάφερε να αποφύγει κάποια μικρά τοπικά ελάχιστα. Όμως πάλι εγκλωβίστηκε σχετικά νωρίς, χωρίς να καταφέρει να εκπαιδεύσει ικανοποιητικά το μοντέλο (Διάγραμμα 3).

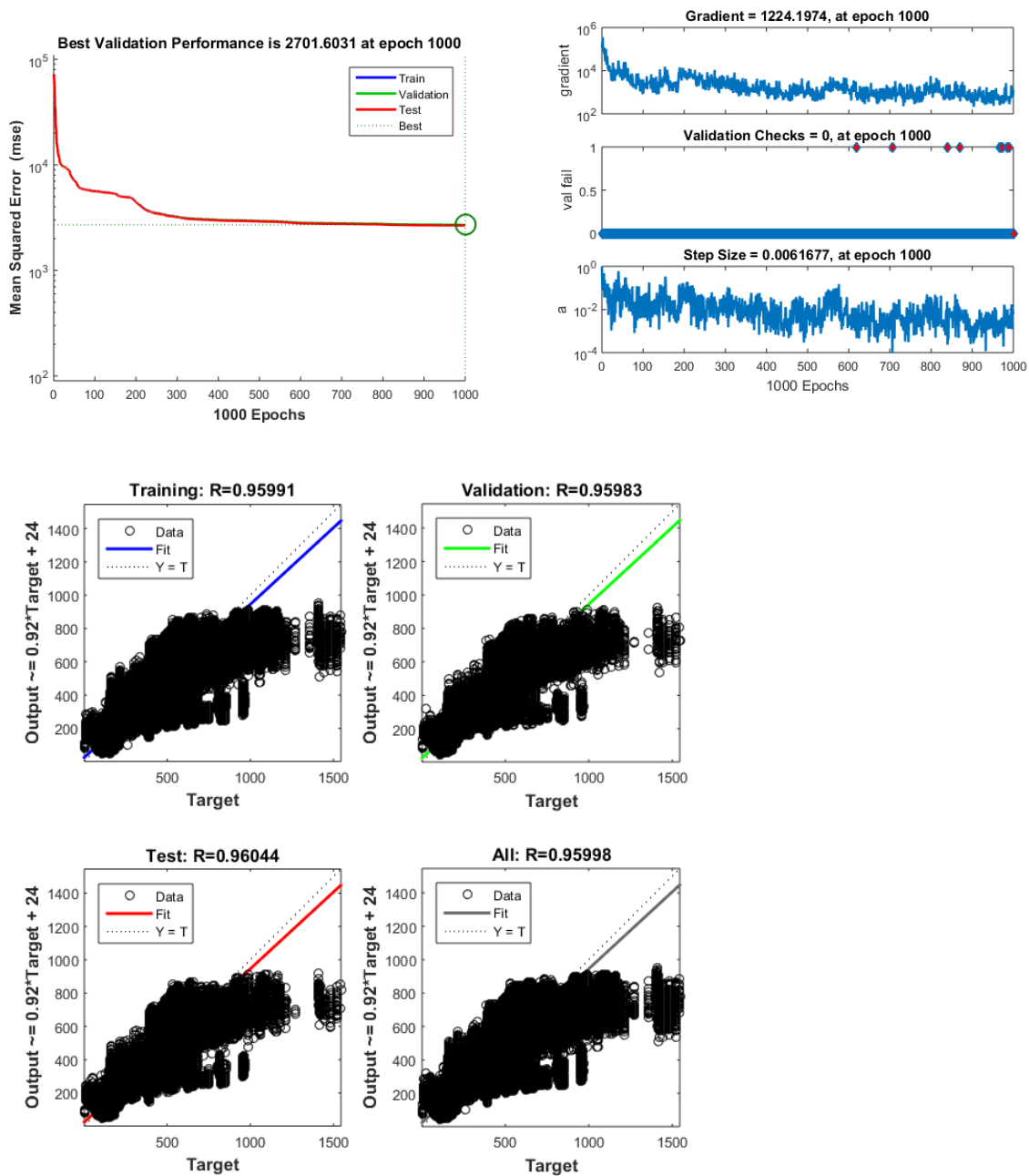
Αλγόριθμος απότομης καθόδου με μεταβλητό ρυθμό μάθησης και ορμή



Διάγραμμα 4: Αποτελέσματα του αλγόριθμου απότομης καθόδου με μεταβλητό ρυθμό μάθησης και ορμή

Και σε αυτή την περίπτωση το σύστημα δεν κατάφερε να εκπαιδευτεί. Τις πρώτες εποχές το σύστημα δεν μπορούσε να βρει καλύτερες λύσεις και όταν το έκανε έπεσε σε τοπικό ελάχιστο και μηδενίστηκε η παράγωγος του. Κάτι που οδήγησε στον τερματισμό της εκπαίδευσης (Διάγραμμα 4).

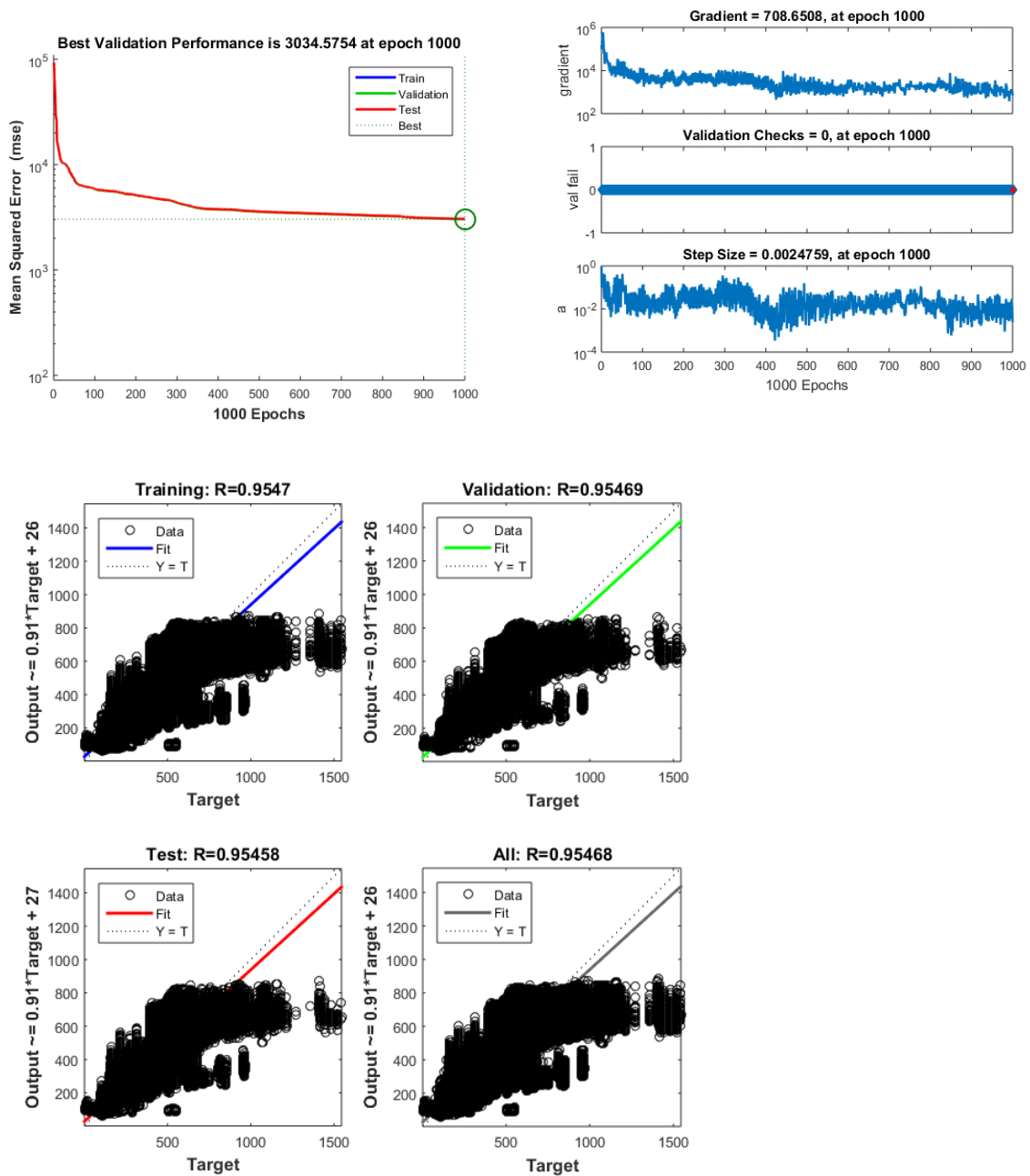
Αλγόριθμος συζυγούς κλίσης με επανεκκινήσεις τύπου *Beale-Powell*



Διάγραμμα 5: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου *Beale-Powell*

Οι μέθοδοι συζυγούς κλίσης, χρησιμοποιούν 2ή τάξη σειράς Taylor, γεγονός που τους βοηθάει ώστε να αποκτούν μεγαλύτερη ταχύτητα σύγκλισης. Η μέθοδος αυτή έδωσε τα καλύτερα αποτελέσματα εν συγκρίσει με τις υπόλοιπους μεθόδους των συζυγών κλίσεων με επανεκκινήσεις, Όμως η εκπαίδευση του Τ.Ν.Δ. όπως φαίνεται δεν έχει ολοκληρωθεί, αφού ακόμα οι τιμές δημιουργούν ένα σύννεφο γύρω από την διαγώνιο, ενώ σε μια ιδανική εκπαίδευση θα έπρεπε να είναι πάνω σε αυτήν (Διάγραμμα 5).

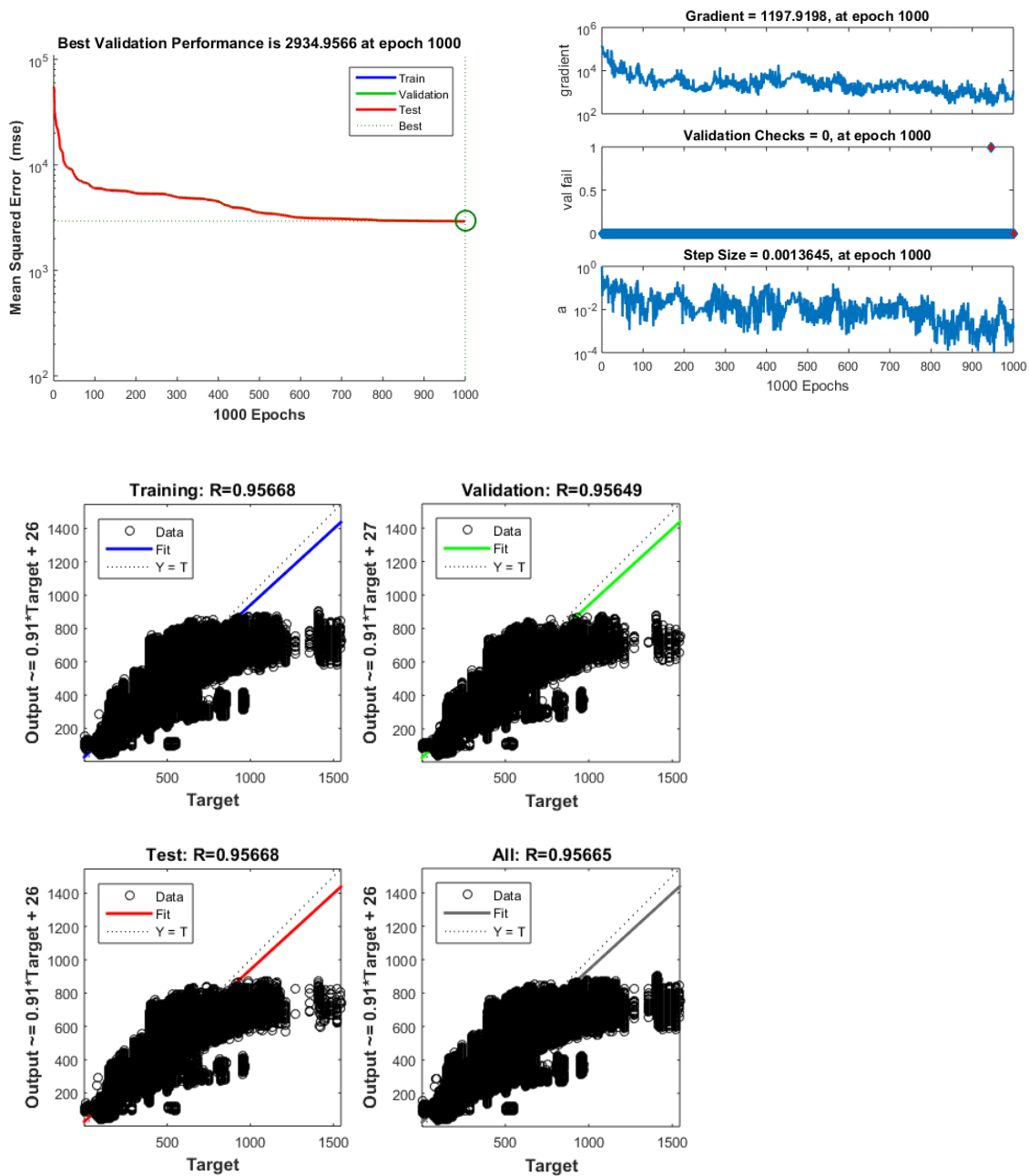
Αλγόριθμος συζυγούς κλίσης με επανεκκινήσεις τύπου *Fletcher-Reeves*



Διάγραμμα 6: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου *Fletcher-Reeves*

Αντίστοιχη εικόνα προκύπτει και για τον αλγόριθμο αυτό. Κάτι που αναμενόταν καθώς η διαφοροποίησή τους, είναι μόνο στο πως προκύπτει το β_k (Διάγραμμα 6). Η διαφορά στην απόδοσή τους δεν ήταν τόσο μεγάλη και ο συγκεκριμένος αλγόριθμος χρειάστηκε περίπου το 75% του χρόνου που απαιτήθηκε για τον προηγούμενο αλγόριθμο, δίνοντας ελάχιστα χειρότερη πρόβλεψη.

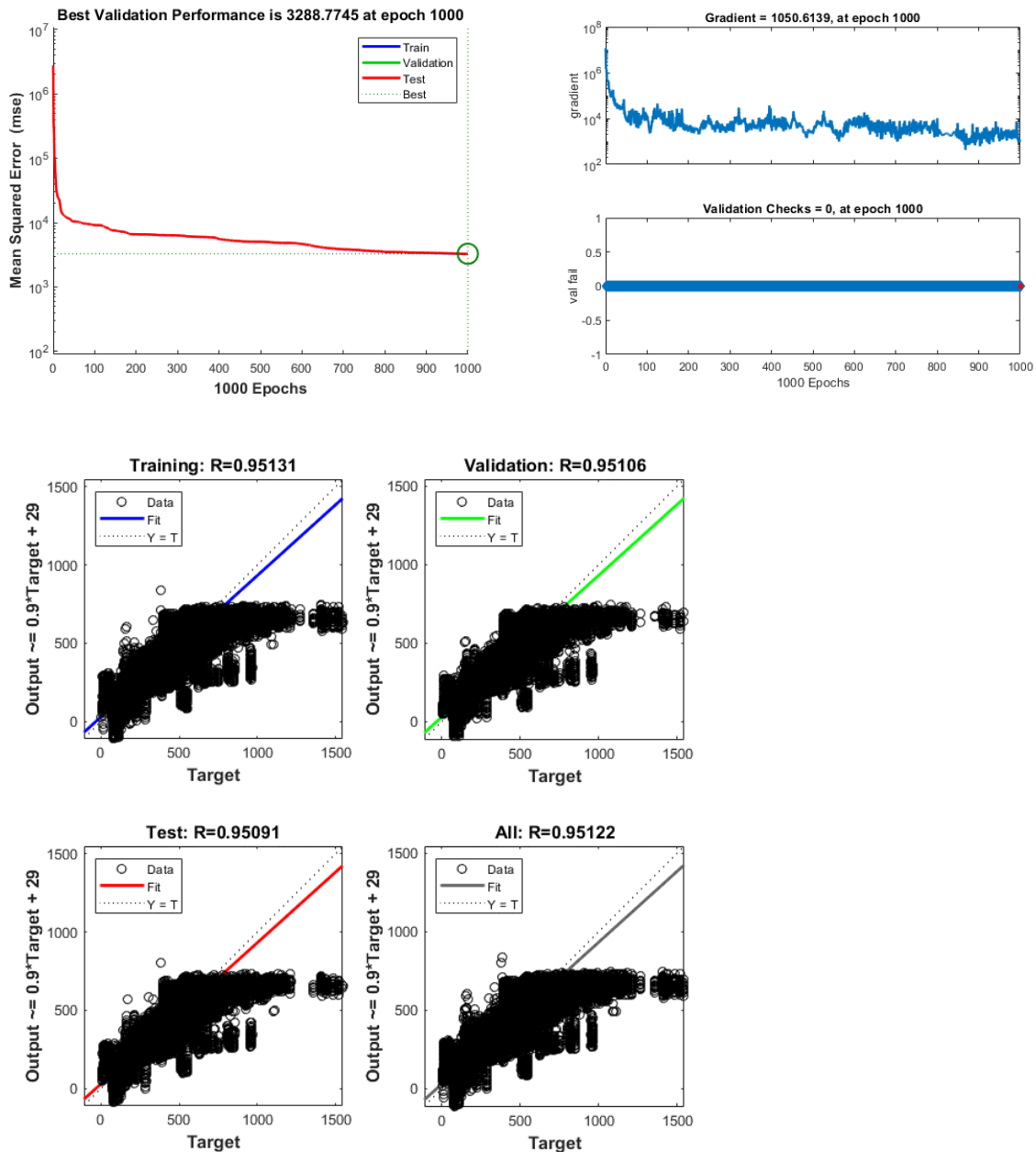
Αλγόριθμος συζυγούς κλίσης με επανεκκινήσεις τύπου *Polak-Ribiere*



Διάγραμμα 7: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου *Polak-Ribiere*

Η *Polak-Ribiere* έδωσε τα καλύτερα αποτελέσματα από τους τρεις αυτούς αλγόριθμους, λαμβάνοντας υπόψιν την απόδοση σε σχέση με το χρόνο που απαιτήθηκε, με την *Fletcher-Reeves* να είναι πολύ κοντά. Ο αλγόριθμος *Beale-Powell* να μην έδωσε λίγο καλύτερα αποτελέσματα, αλλά εξαιτίας της χρήσης των επιπλέον συναρτήσεων που έχει αυτή η μέθοδος αυξήθηκε και το υπολογιστικό κόστος, δυσανάλογα σε σχέση με την απόδοση της (Διάγραμμα 7).

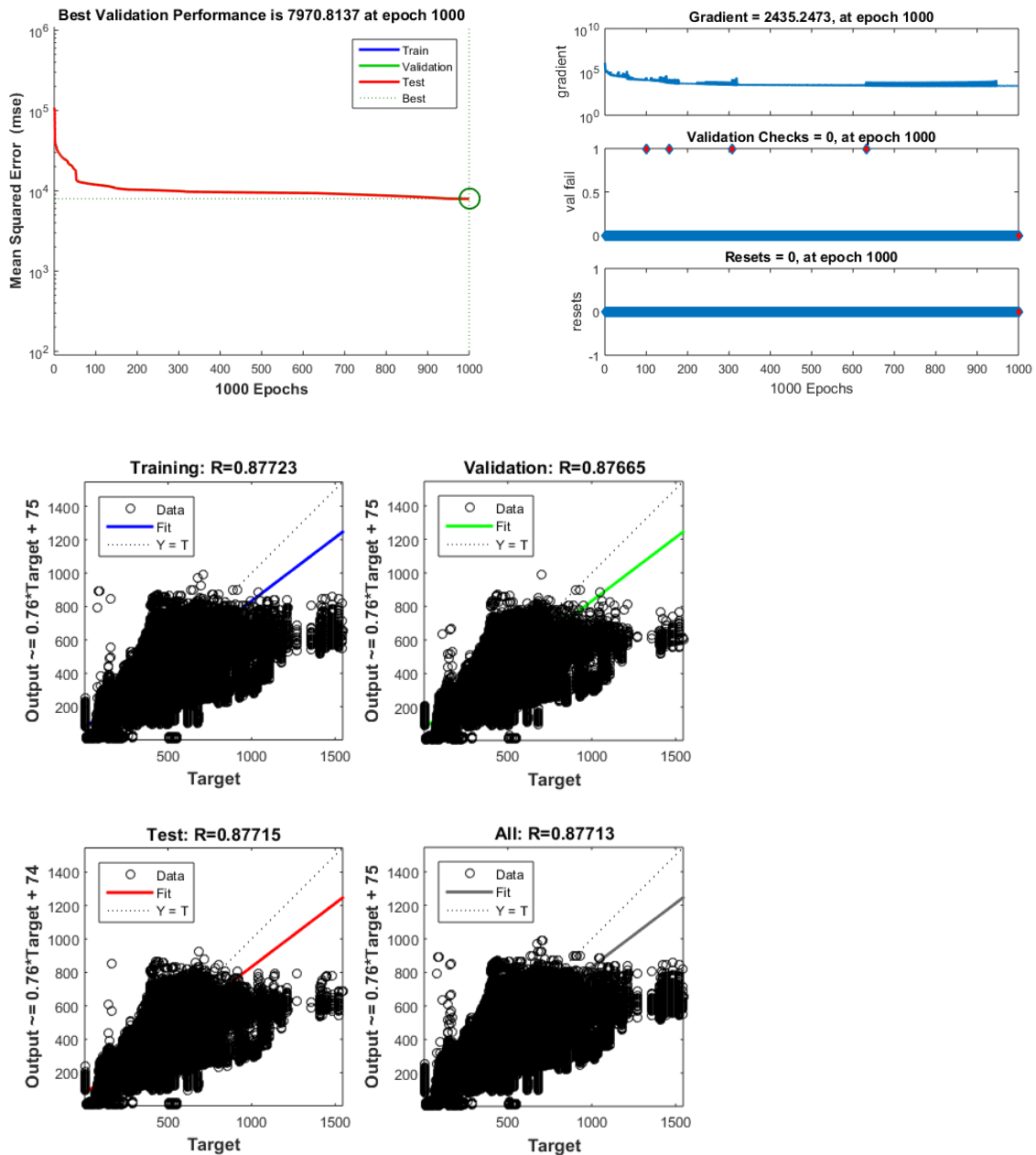
Αλγόριθμος βαθμισμένης συζυγούς κλίσης



Διάγραμμα 8: Αποτελέσματα του αλγόριθμου βαθμισμένης συζυγούς κλίσης

Παρόλο που διαφέρει αρκετά ο τρόπος υπολογισμού του β_k και το γεγονός ότι η συγκεκριμένη μέθοδος δεν χρησιμοποιεί κάποια συνάρτηση γραμμικής ερευνάς, ήταν σχεδόν το ίδιο αποτελεσματική με τις άλλες μεθόδους που χρησιμοποιούν επανεκκινήσεις. Όμως ήταν πολύ πιο γρήγορη καθώς ναι μεν έδωσε λίγο χειρότερα αποτελέσματα εκπαίδευσης, αλλά χρειάστηκε μόνο το $\frac{1}{3}$ του χρόνου που απαιτούσαν οι *Polak-Ribiere* και *Fletcher-Reeves* και το $\frac{1}{4}$ του απαιτούμενου χρόνου εκπαίδευσης του *Beale-Powell* (Διάγραμμα 8).

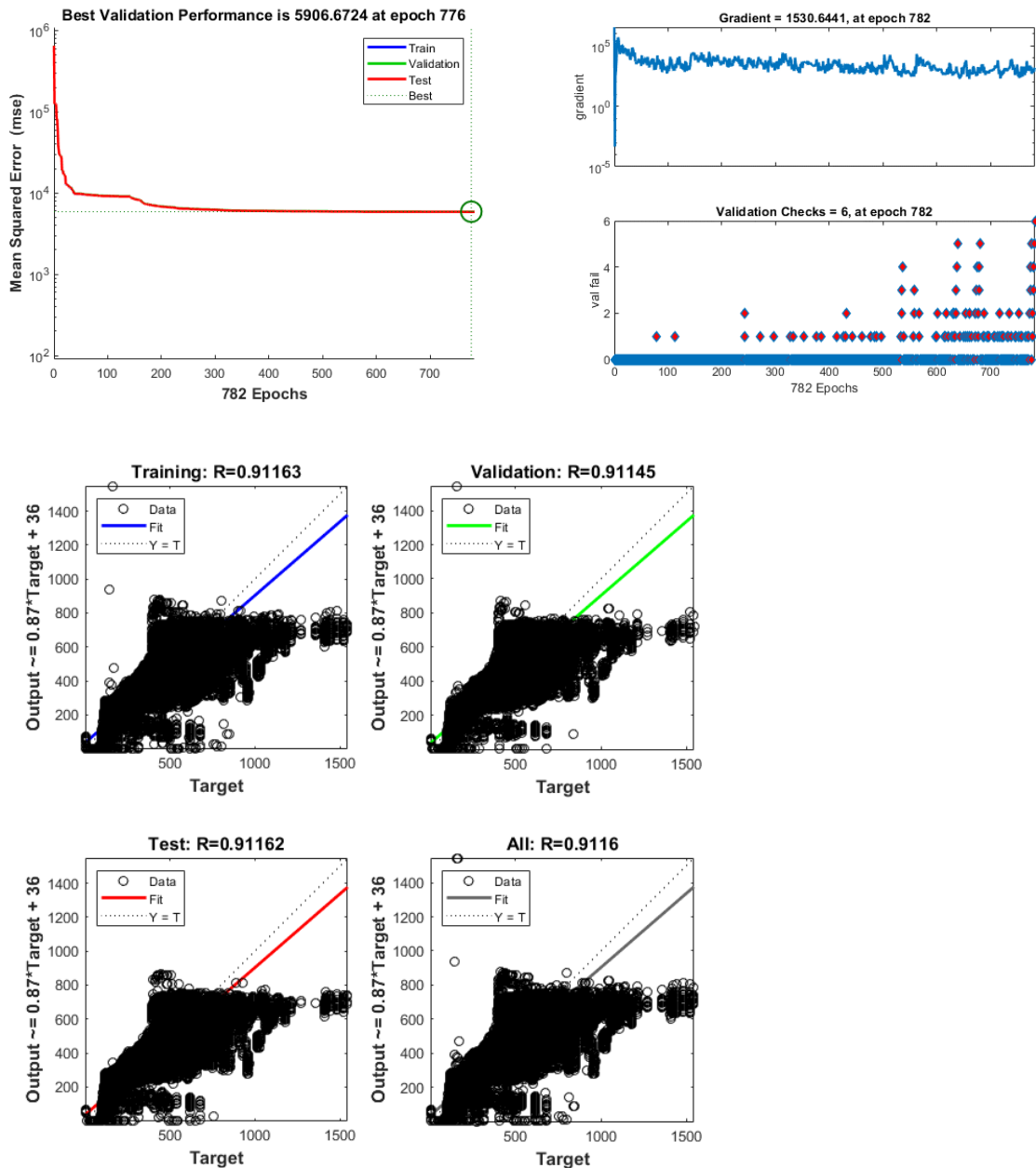
Αλγόριθμος *BFGS-Quasi-Newton*



Διάγραμμα 9: Αποτελέσματα του αλγόριθμου *Quasi-Newton*

Όπως αναμενόταν η μέθοδος *BFGS-Quasi-Newton*, θα είχε αργή εκτέλεση (η διάρκεια της εκπαίδευσης ήταν 26 ώρες) καθώς σε κάθε επανάληψη θα απαιτούσε τον προσεγγιστικό υπολογισμό του πίνακα *Hessian*. Ενώ και τα αποτελέσματα της δεν ήταν και τα καλύτερα δυνατά, καθώς έδωσε πολύ χειρότερα αποτελέσματα σε σχέση με τις μεθόδους συζυγούς κλίσης που χρειάστηκαν περίπου 4,5 ώρες για την εκπαίδευση του ιδίου Τ.Ν.Δ. (Διάγραμμα 9).

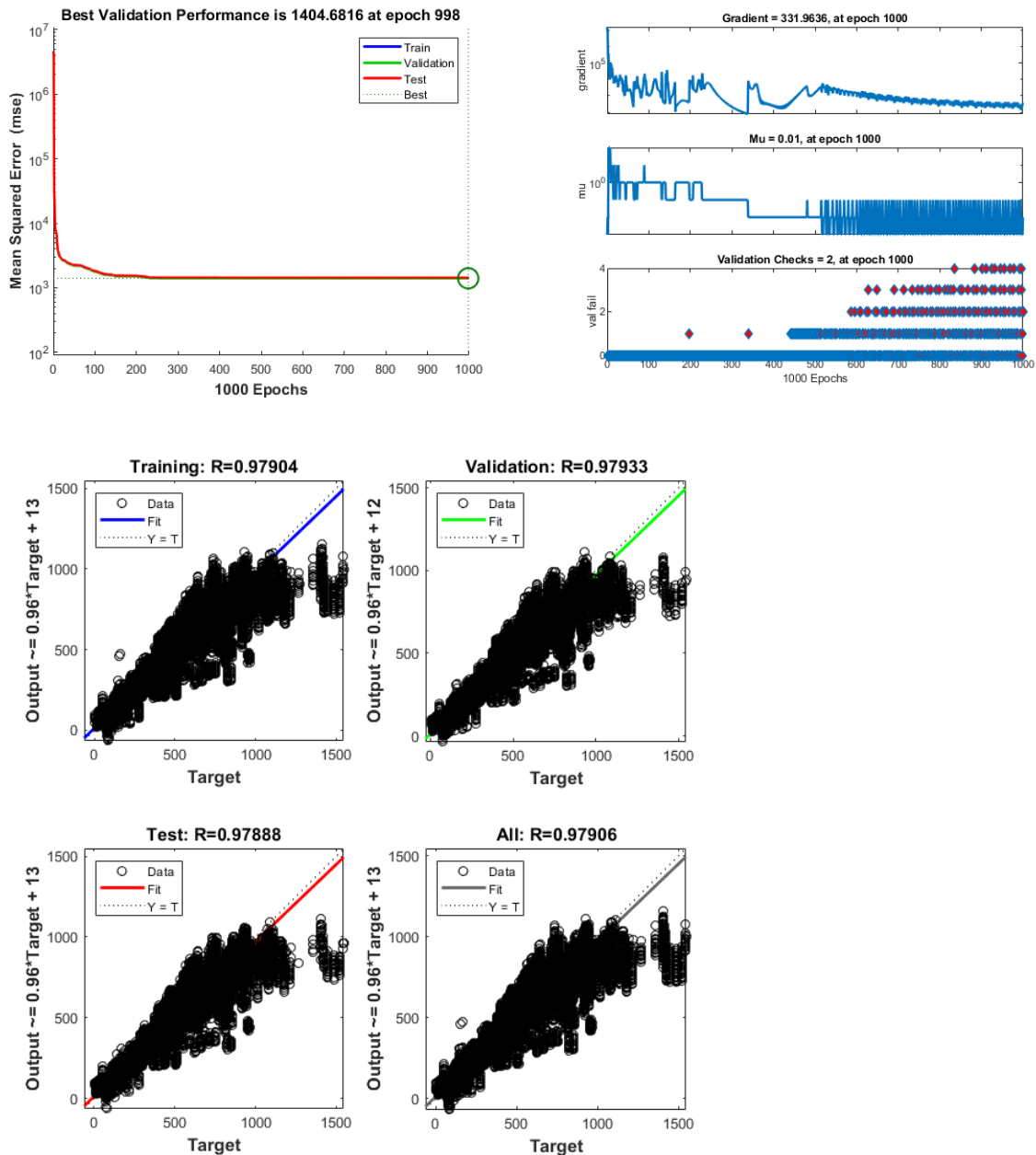
Αλγόριθμος *One-Step Secant*



Διάγραμμα 10: Αποτελέσματα του αλγόριθμου *One-Step Secant*

Όπως αναφέρθηκε στην θεωρία αυτή η *One-Step Secant*, είναι κάτι ενδιάμεσο μεταξύ των μεθόδων συζυγούς κλίσης και των μεθόδων *Quasi-Newton*. Αντίστοιχα και τα αποτελέσματά της, βρέθηκαν ανάμεσα τους. Η *One-Step Secant*, μπορεί να απέδωσε καλύτερα από την *BFGS-Quasi-Newton* και να χρειάστηκε μόνο το 25% του χρόνου της πρώτης. Όμως έδωσε πολύ χειρότερες προβλέψεις από τις μεθόδους συζυγούς κλίσης, ξεπερνώντας σε χρόνο εκπαίδευσης την *Beale-Powell*, την ώρα που τερματίστηκε πρόωρα λόγω ελέγχων επικύρωσης (Διάγραμμα 10).

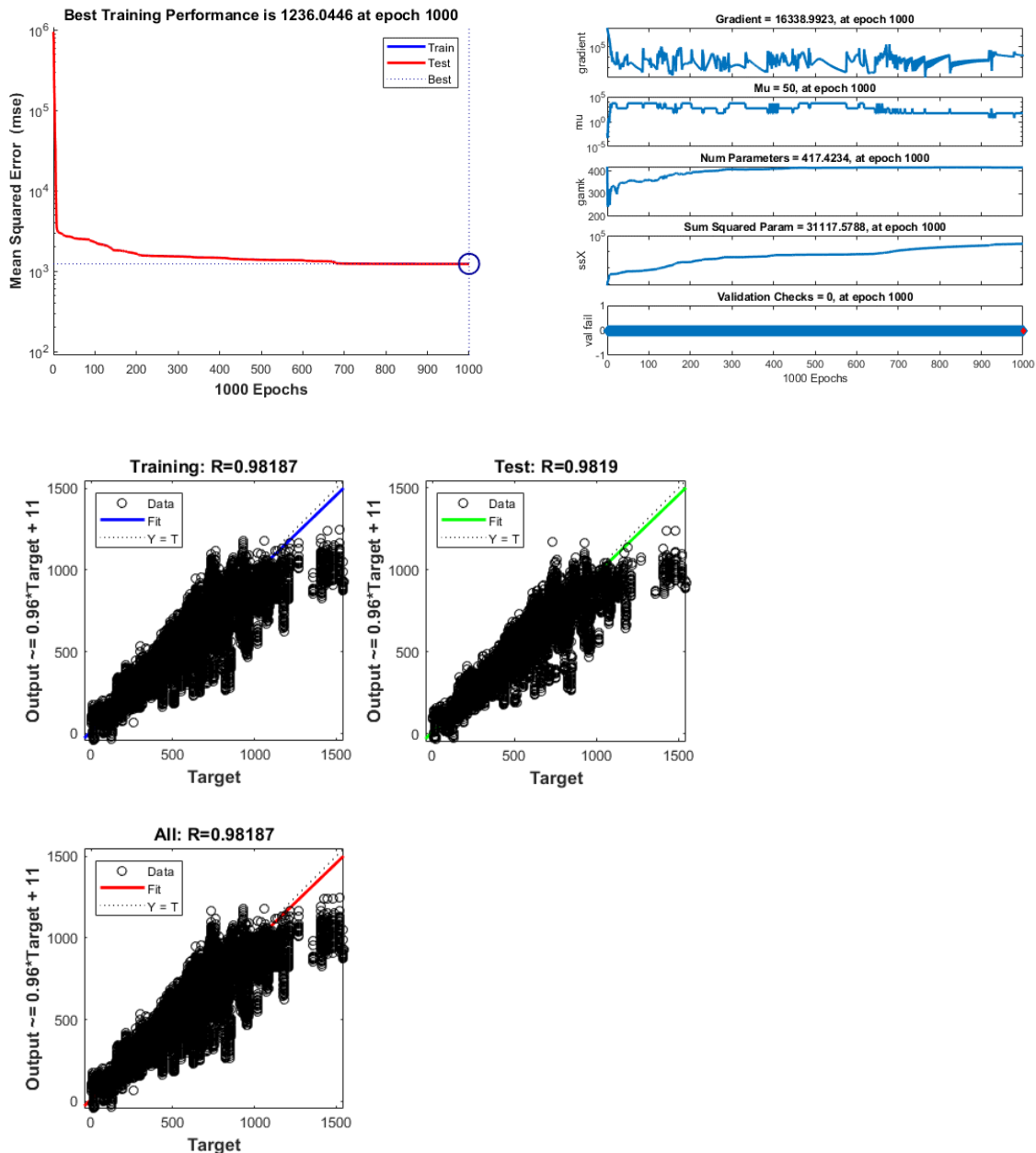
Αλγόριθμος *Levenberg-Marquardt*



Διάγραμμα 11: Αποτελέσματα του αλγόριθμου *Levenberg-Marquardt*

Όπως αναμενόταν από την θεωρία, ο αλγόριθμος αυτός θα έδινε πολύ καλύτερα αποτελέσματα εκπαίδευσης. Η απόδοση του σε σχέση με τις μεθόδους συζυγούς κλίσης ήταν πολύ καλύτερη, αλλά με το μειονέκτημα ότι χρειάστηκε τον τετραπλάσιο χρόνο εκπαίδευσης από τις *Polak-Ribiere* και *Fletcher-Reeves* και τον τριπλάσιο από την *Beale-Powell*. Παρόλα αυτά ακόμα και αυτή η μέθοδος δεν έδωσε ένα καλά εκπαιδευμένο Τ.Ν.Δ. (Διάγραμμα 11).

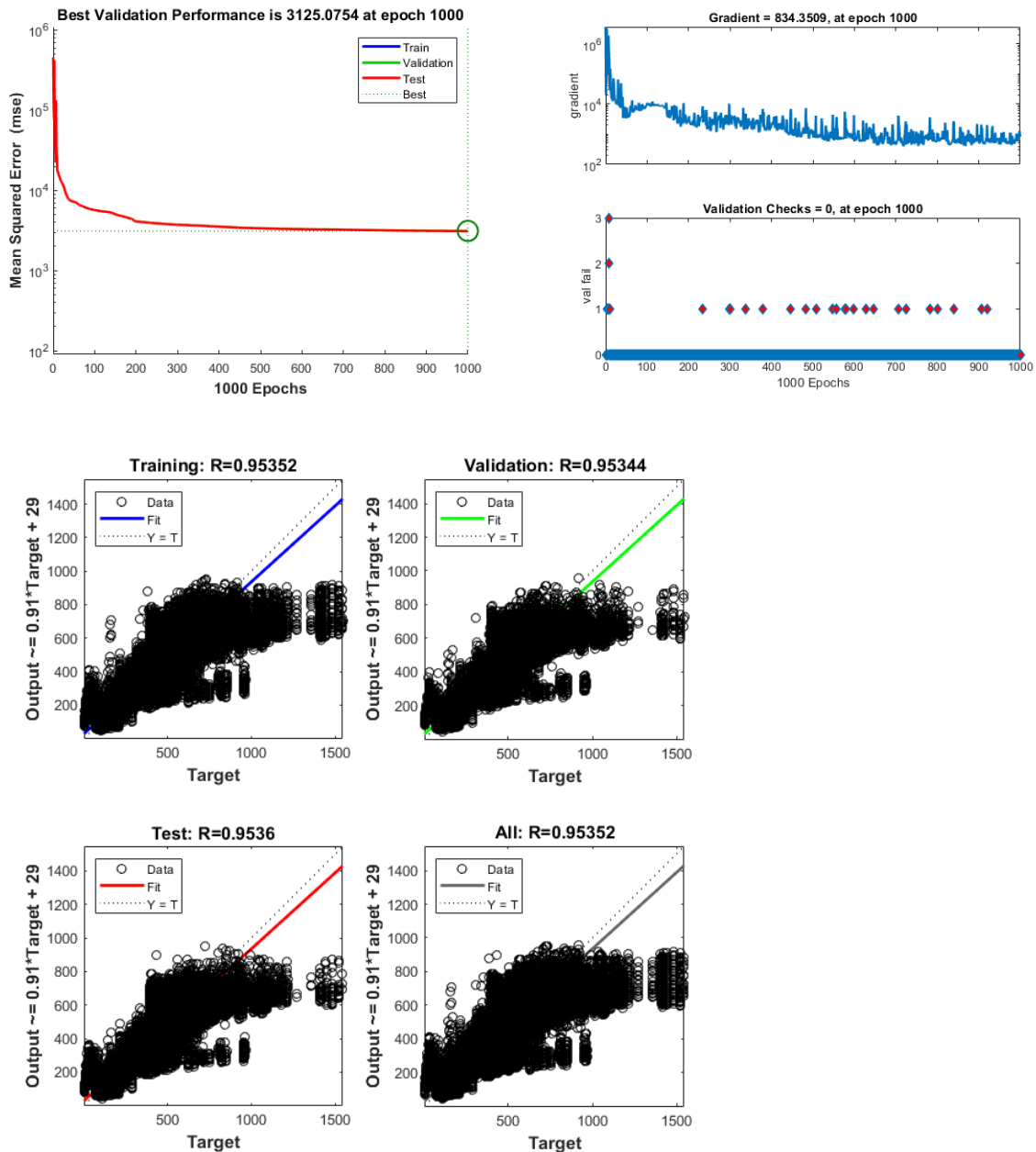
Αλγόριθμος *Bayesian Regularization*



Διάγραμμα 12: Αποτελέσματα του αλγόριθμου *Bayesian Regularization*

Από την στιγμή που πρόκειται για ίδιο αλγόριθμο εκπαίδευσης με την *Levenberg-Marquardt* αναμένονταν αντίστοιχα αποτελέσματα. Η εκπαίδευση χρειάστηκε λίγο περισσότερο χρόνο σε σχέση με την *Levenberg-Marquardt* (περίπου 10% παραπάνω) κάτι που ενδεχομένως να οφείλεται στον υπολογισμό του αθροίσματος των τετραγώνων των βαρών σε κάθε εποχή. Χαρακτηριστικό της μεθόδου αυτής, είναι η μη χρήση επικύρωσης, κάτι που οδηγεί και σε ένα λιγότερο διάγραμμα παλινδρόμησης στα αποτελέσματα (Διάγραμμα 12).

Αλγόριθμος Ανθεκτικής Διάδοσης (RProp)



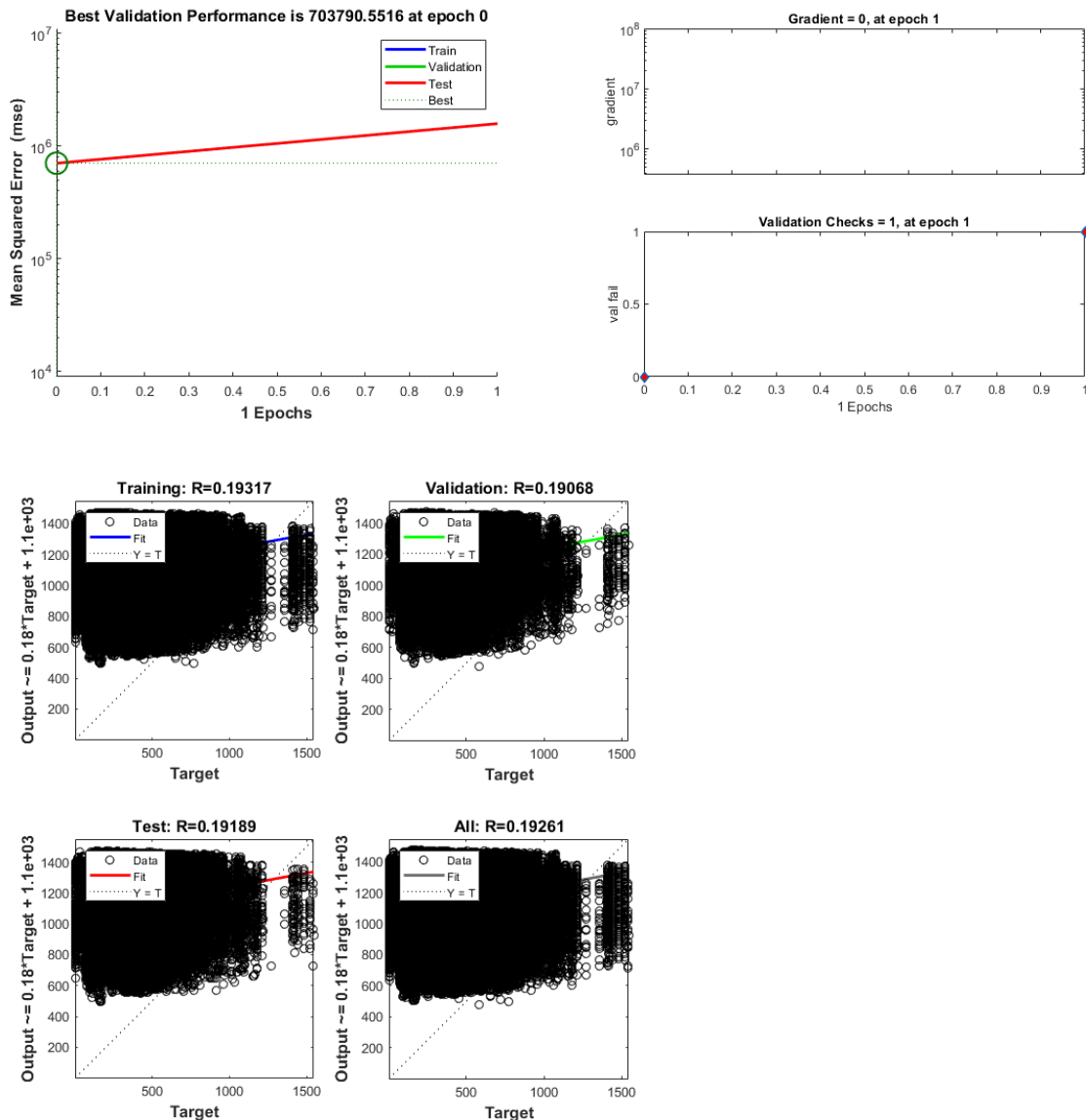
Διάγραμμα 13: Αποτελέσματα του αλγόριθμου Ανθεκτικής Διάδοσης (RProp)

Η μέθοδος αυτή όπως αναφέρθηκε στην θεωρία δεν χρησιμοποιεί πληροφορίες 2^{ης} τάξης κάτι που την κάνει αρκετά πιο γρήγορη από τις υπόλοιπες μεθόδους (με εξαίρεση της μεθόδους της απότομης καθόδου) χωρίς όμως να γίνονται ιδιαίτερες θυσίες όσον αφορά την απόδοση. Τα αποτελέσματα που έδωσε, είναι αντίστοιχα των μεθόδων συζυγούς κλίσης, αλλά ο χρόνος εκπαίδευσης της ήταν πολύ μικρότερος, αφού η εκπαίδευση του Τ.Ν.Δ. είχε ολοκληρωθεί σε 45 λεπτά (Διάγραμμα 13).

ΔΗΜΙΟΥΡΓΙΑ Τ.Ν.Δ ΜΕ ΔΥΟ ΚΡΥΦΑ ΕΠΙΠΕΔΑ

Τα αποτελέσματα της εκπαίδευσης με χρήση ενός κρυφού επιπέδου, δεν ήταν ικανοποιητικά, κάτι που αναμενόταν, καθώς λόγω της πολυπλοκότητας του προβλήματος θα απαιτούνταν μια αρκετά πιο πολύπλοκη αρχιτεκτονική. Για το λόγο αυτό, οι εκπαιδεύσεις επαναλήφθηκαν με την προσθήκη ενός επιπλέον κρυφού επιπέδου.

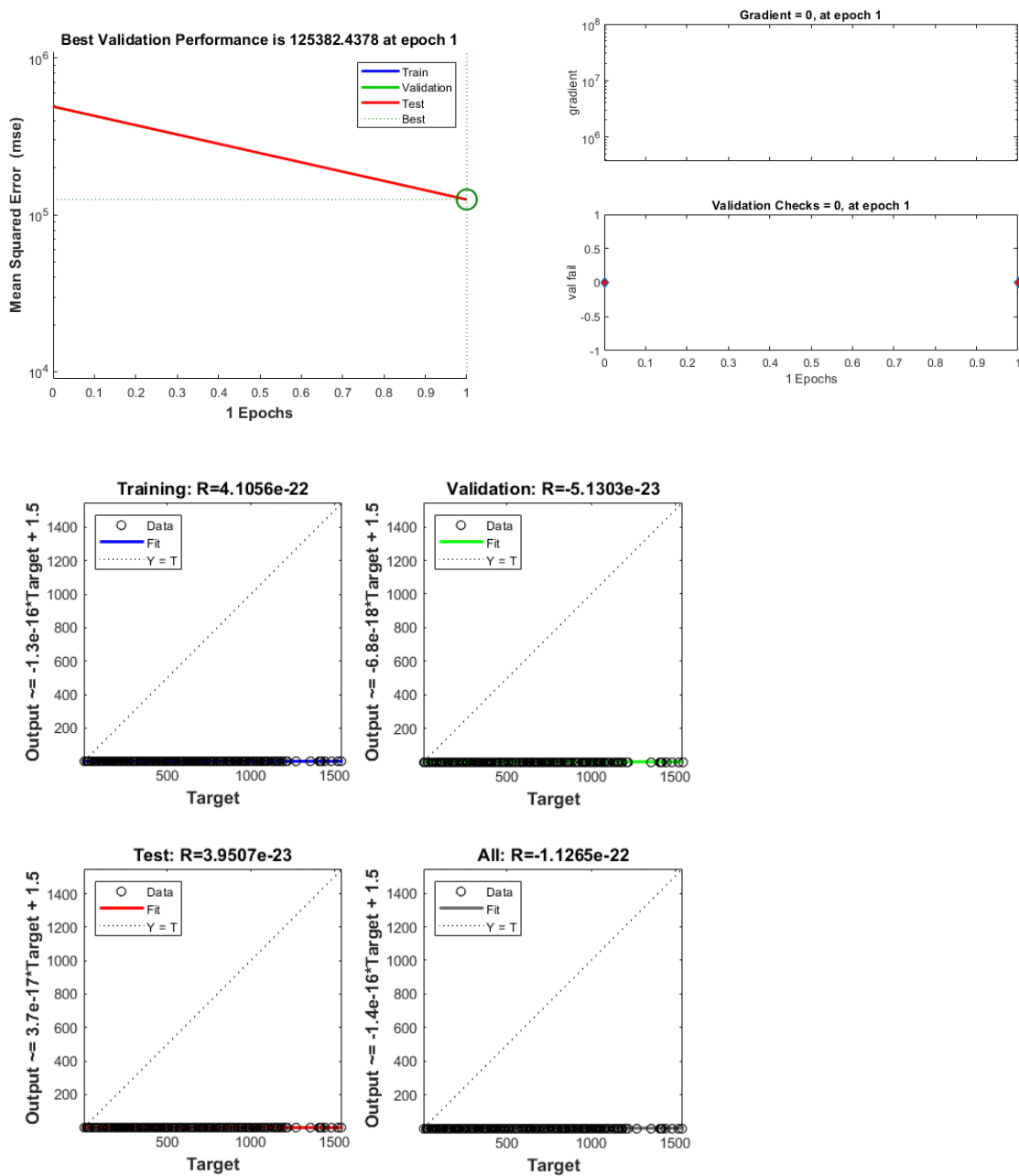
Αλγόριθμος απότομης καθόδου με δύο κρυφά επίπεδα



Διάγραμμα 14: Αποτελέσματα του αλγόριθμου απότομης καθόδου και δύο κρυφά επίπεδα.

Από την στιγμή που η μέθοδος της απότομης καθόδου είχε αποτύχει στην απλή δομή του ενός κρυφού επιπέδου, ήταν σχεδόν βέβαιο ότι δεν θα μπορούσε να ανταποκριθεί στις απαιτήσεις της εκπαίδευσης ενός Τ.Ν.Δ. με δύο κρυφά επίπεδα, κάτι που επιβεβαιώθηκε από τα αποτελέσματα της δοκιμής (Διάγραμμα 14).

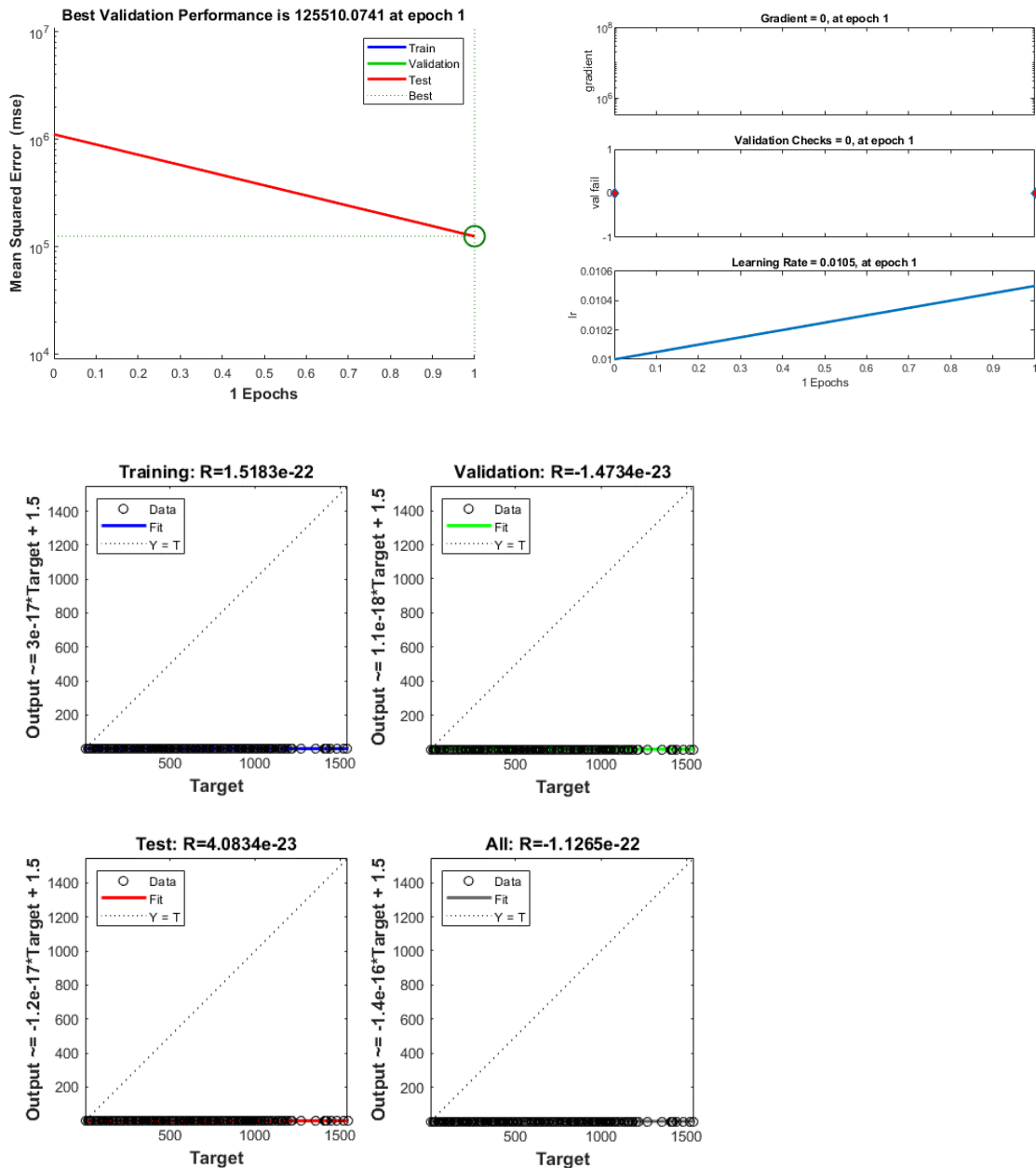
Αλγόριθμος απότομης καθόδου με ορμή και δύο κρυφά επίπεδα



Διάγραμμα 15: Αποτελέσματα του αλγόριθμου απότομης καθόδου με ορμή και δύο κρυφά επίπεδα

Ίδια εικόνα για την μέθοδο της απότομης καθόδου με ορμή. Το σύστημα φτάνει μεν σε κάποιο τοπικό ακρότατο μετά από μία μόνο επανάληψη, αλλά ο μηδενισμός της παραγώγου οδηγεί στην ολοκλήρωση της εκπαίδευσης, οδηγώντας σε υποπροσαρμογή (Διάγραμμα 15).

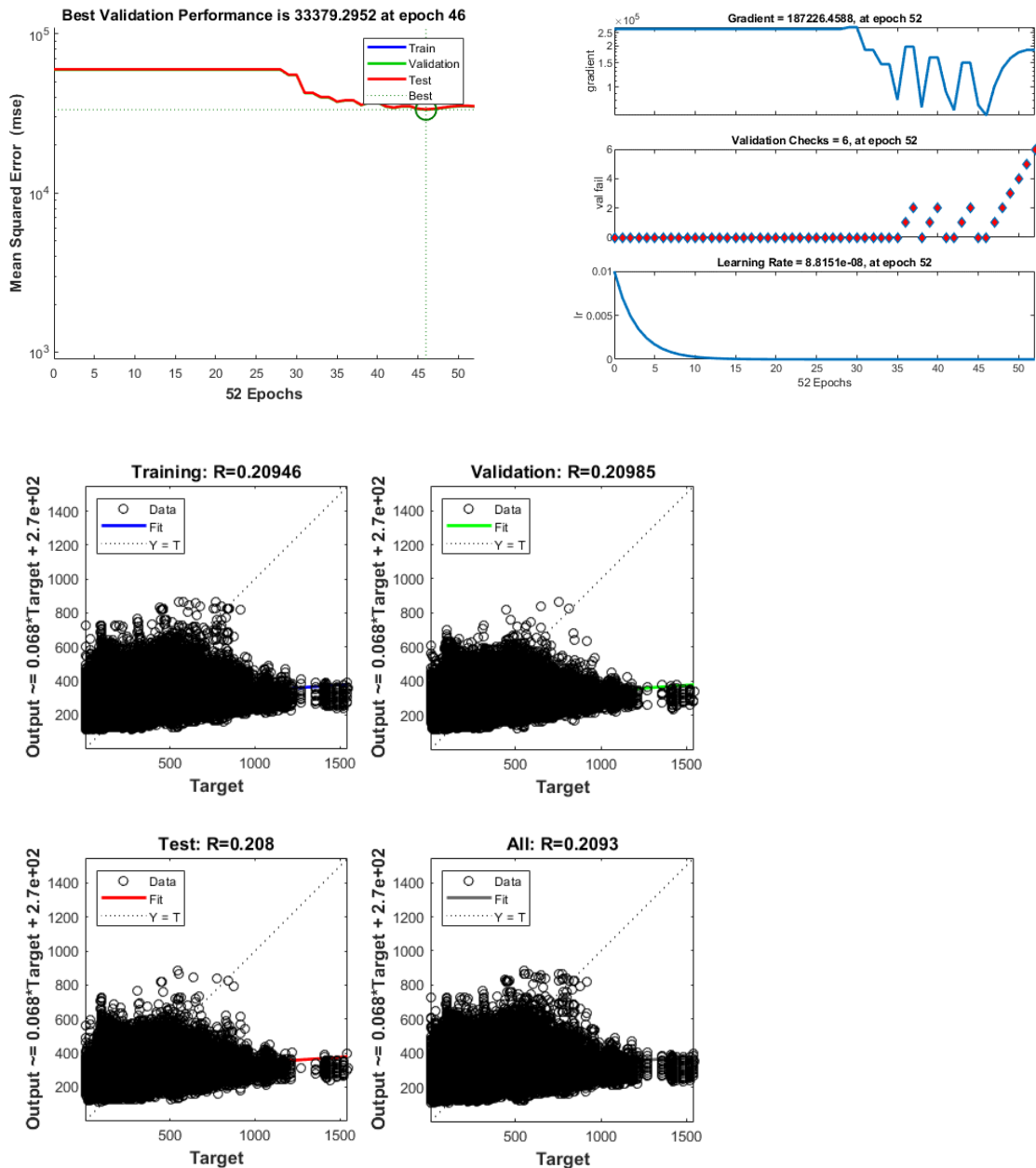
Αλγόριθμος απότομης καθόδου με M.P.M. και δύο κρυφά επίπεδα



Διάγραμμα 16: Αποτελέσματα του αλγόριθμου απότομης καθόδου με μεταβλητό ρυθμό μάθησης και δύο κρυφά επίπεδα

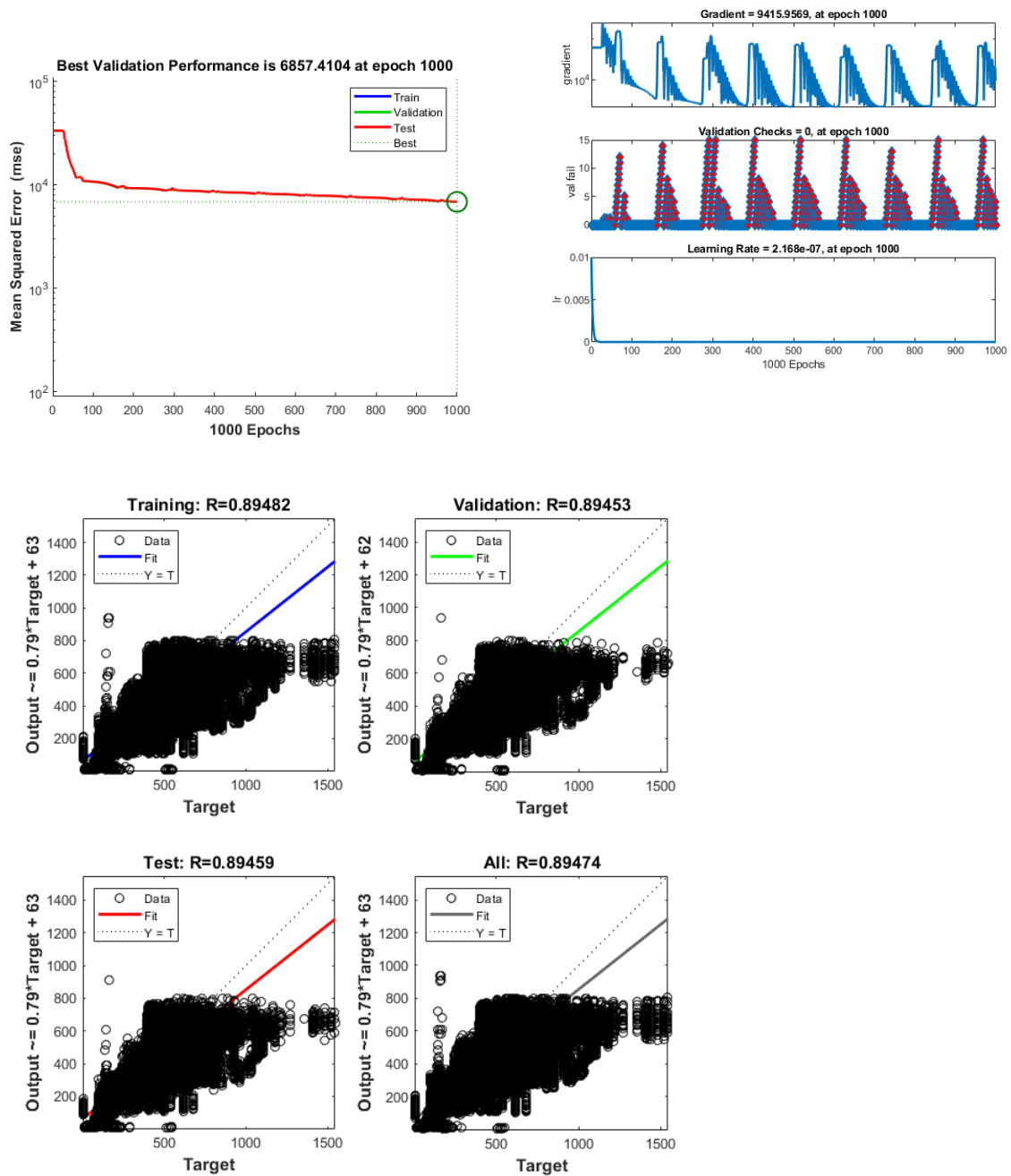
Παρόλο που με ένα επίπεδο, η μέθοδος αυτή είχε καταφέρει να δώσει κάποια αποτελέσματα, στην δοκιμή με χρήση δύο κρυφών επιπέδων, ο τερματισμός της ήταν άμεσος καθώς εγκλωβίστηκε σε ένα τοπικό ελάχιστο (Διάγραμμα 16).

Αλγόριθμος απότομης καθόδου με M.P.M., ορμή και δύο κρυφά επίπεδα



Διάγραμμα 17: Αποτελέσματα του αλγόριθμου καθόδου με μεταβλητό ρυθμό μάθησης, ορμή και δύο κρυφά επίπεδα (1^η δοκιμή)

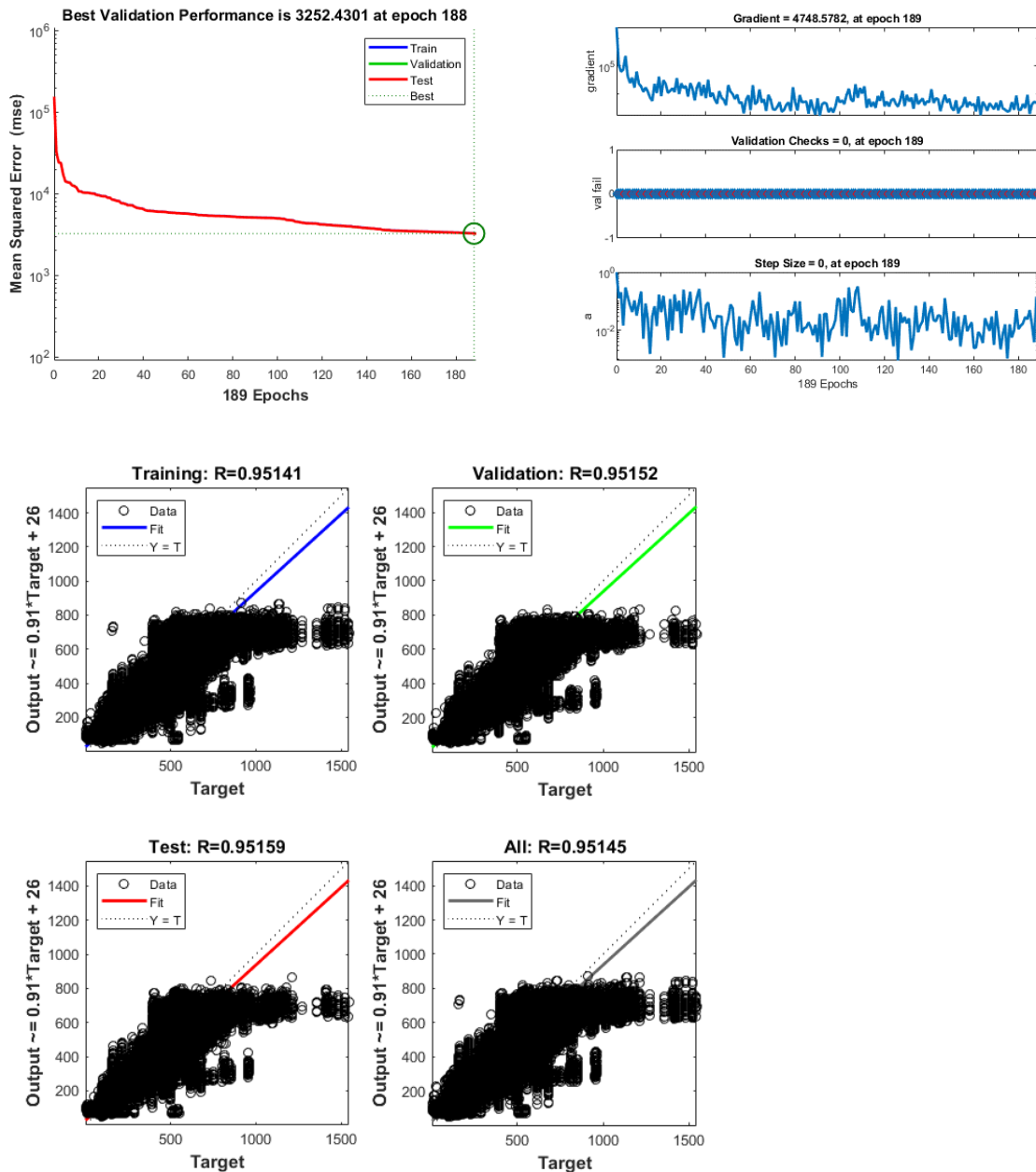
Σε αυτή την περίπτωση, ήταν η πρώτη φορά που μια μέθοδος απότομης καθόδου, τερματίστηκε για κάποιο άλλο λόγο πέρα από τον μηδενισμό της κλίσης (Διάγραμμα 17). Για το λόγο αυτό, η δοκιμή επαναλήφθηκε, τροποποιώντας την μεταβλητή για τα σφάλματα επικύρωσης.



Διάγραμμα 18: Αποτελέσματα του αλγόριθμου καθόδου με μεταβλητό ρυθμό μάθησης, ορμή και δύο κρυφά επίπεδα (2^η δοκιμή)

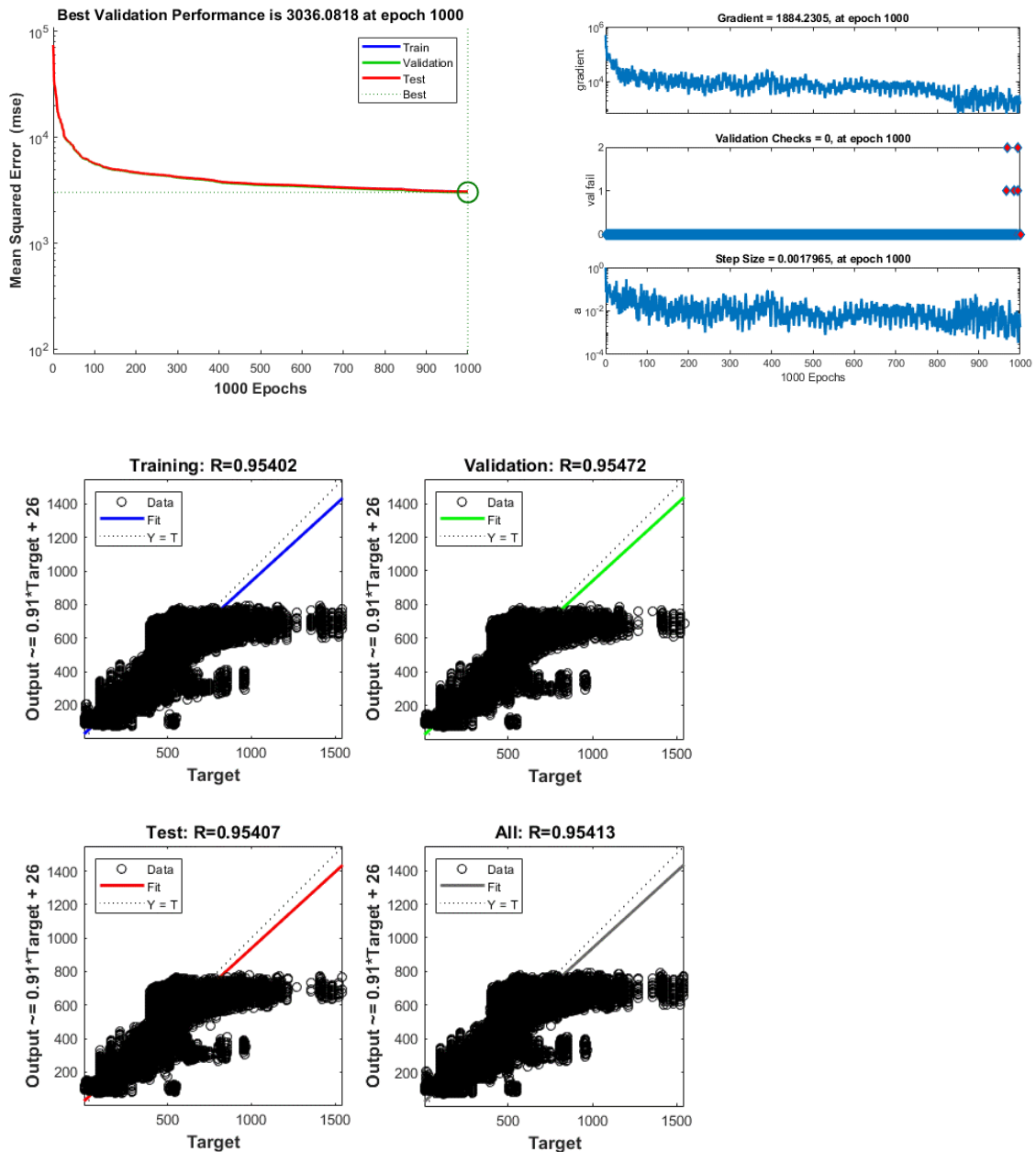
Δίνοντας μεγαλύτερο περιθώριο για σφάλματα επικύρωσης, η μέθοδος αυτή κατάφερε να ολοκληρώσει την εκπαίδευση του Τ.Ν.Δ. σε σχετικά σύντομο χρόνο. Όμως το αποτέλεσμα της εκπαίδευσης της με δύο κρυφά επίπεδα ήταν εμφανώς χειρότερο σχεδόν από όλες τις εκπαίδευσης με ένα κρυφό (μόνο η *Quasi-Newton* είχε ελάχιστα χειρότερο αποτέλεσμα με ένα κρυφό επίπεδο) (Διάγραμμα 18).

Αλγόριθμος Σ.Κ. με επανεκκινήσεις τύπου *Beale-Powell* και δύο κρυφά επίπεδα



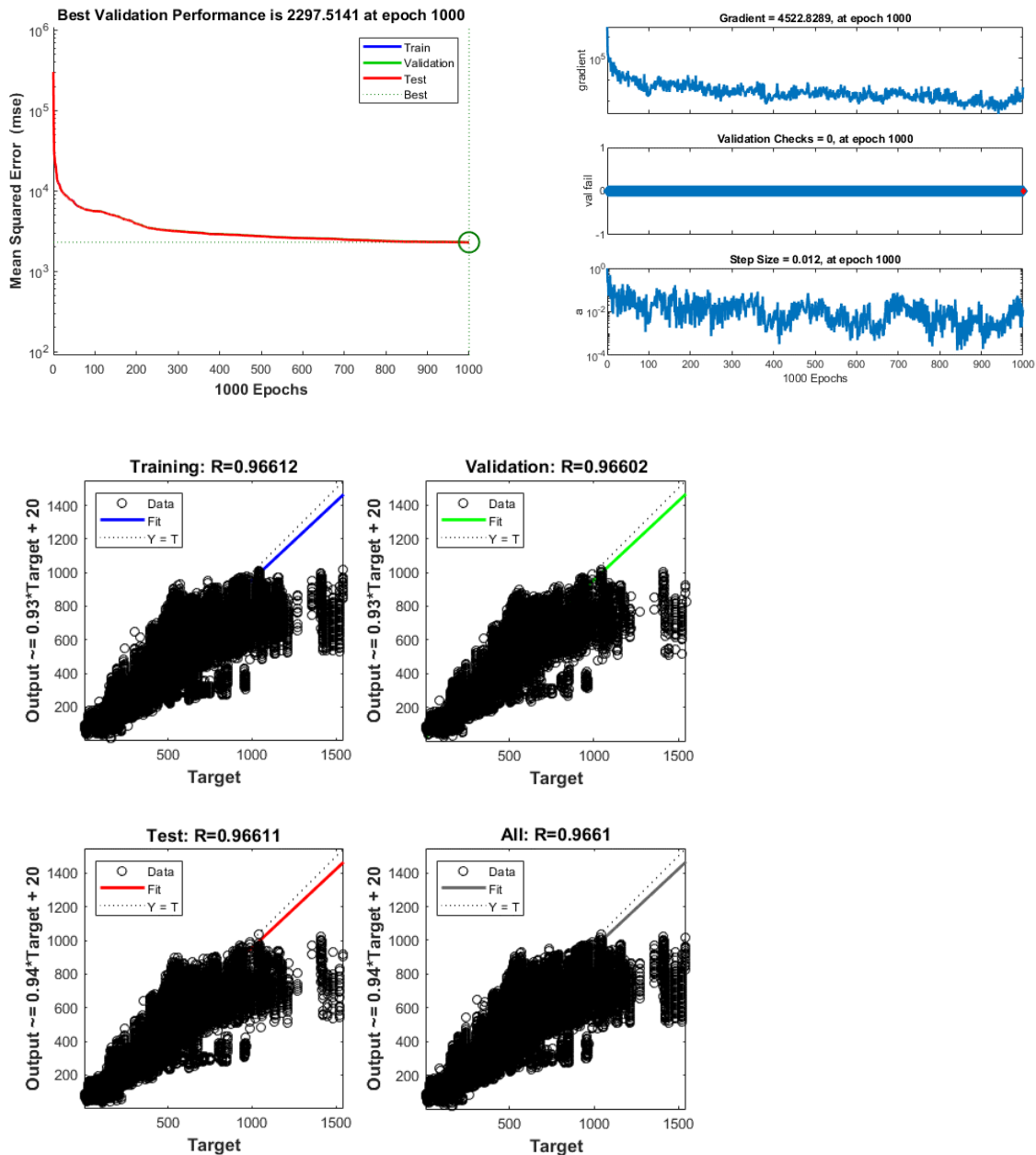
Διάγραμμα 19: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου *Beale-Powell* και δύο κρυφά επίπεδα

Λόγω του ότι εγκλωβίστηκε σε κάποιο τοπικό ελάχιστο, η εκπαίδευση σταμάτησε αρκετά νωρίς λόγω πρόωρου τερματισμού. Παρόλα αυτά η προσθήκη ενός ακόμα επιπέδου, βελτίωσε αρκετά την απόδοση της μεθόδου, αφού χωρίς να έχει ολοκληρωθεί η εκπαίδευση και στο ¼ του χρόνου που χρειάστηκε στην αντίστοιχη δοκιμή με ένα κρυφό επίπεδο, η εκπαίδευση του Τ.Ν.Δ., έφτασε τα ίδια επίπεδα (Διάγραμμα 19).

Αλγόριθμος Σ.Κ. με επανεκκινήσεις τύπου *Fletcher-Reeves* και δύο κρυφά επίπεδα

Διάγραμμα 20: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου *Fletcher-Reeves* και δύο κρυφά επίπεδα

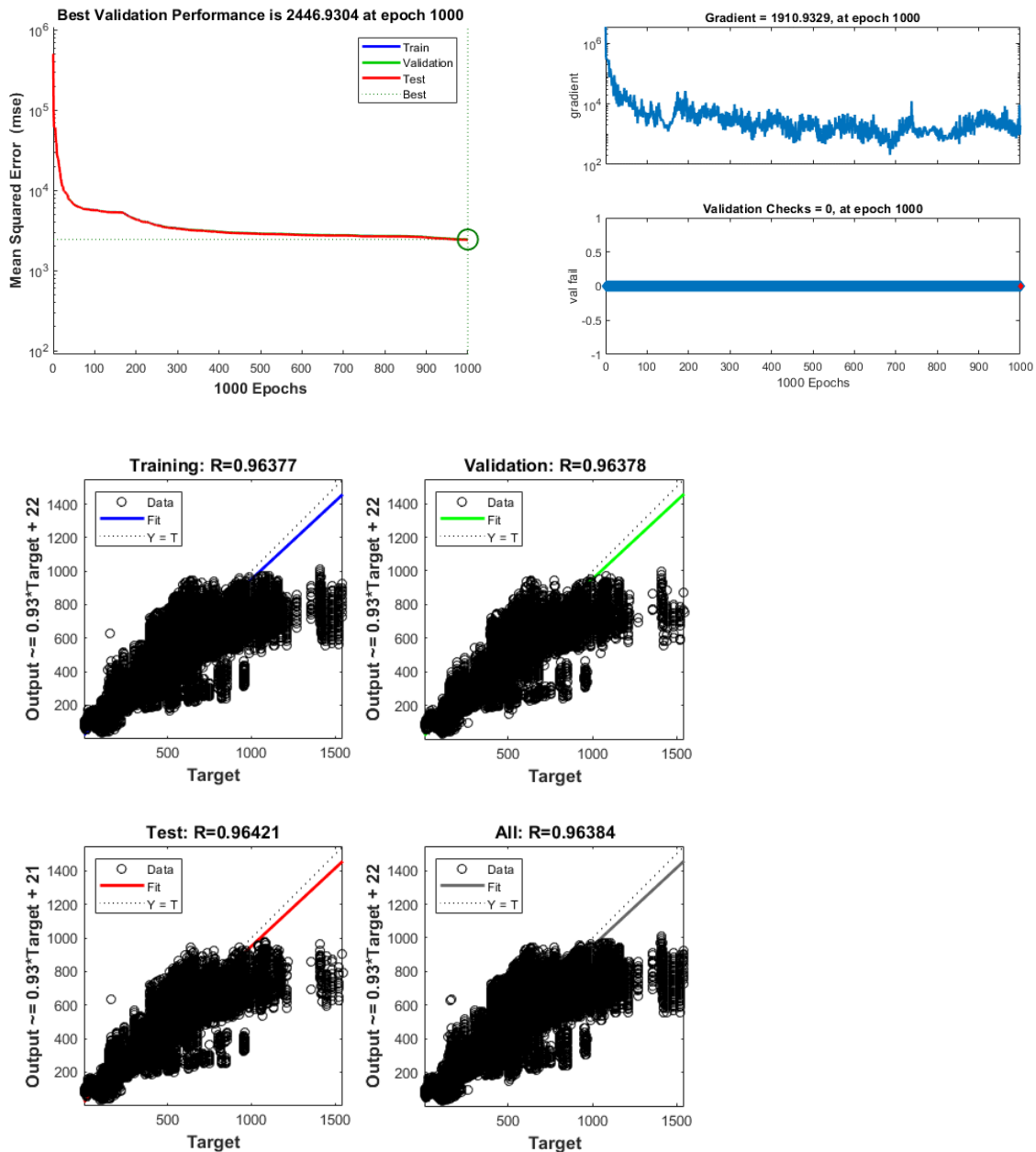
Απογοητευτικά ήταν τα αποτελέσματα της συγκεκριμένης μεθόδου με την προσθήκη ενός ακόμα κρυφού επιπέδου καθώς δεν υπήρξε καμία βελτίωση στην ικανότητα του μοντέλου. Ενώ και σε αυτή την περίπτωση, υπήρχαν αρκετά σφάλματα επικύρωσης προς το τέλος της εκπαίδευσης. Κάτι που αποτελεί ένδειξη ότι ο αλγόριθμος ενδεχομένως σύντομα να εγκλωβιζόταν σε κάποιο τοπικό ελάχιστο, εάν συνεχιζόταν η εκπαίδευση (Διάγραμμα 20).

Αλγόριθμος Σ.Κ. με επανεκκινήσεις τύπου *Polak-Ribiere* και δύο κρυφά επίπεδα

Διάγραμμα 21: Αποτελέσματα του αλγόριθμου συζυγούς κλίσης με επανεκκινήσεις τύπου *Polak-Ribiere* και δύο κρυφά επίπεδα

Σαφή βελτίωση των αποτελεσμάτων της παρουσίασε αυτή η εκδοχή της μεθόδου των συζυγών κλίσεων. Με την προσθήκη του 2^{ου} κρυφού επιπέδου υπήρξε μια βελτίωση πάνω από 20% της απόδοσης του αλγορίθμου την ώρα που η διάρκεια εκπαίδευσης δεν ήταν μεγαλύτερη από αυτήν που χρειάστηκε ο αλγόριθμος *Beale-Powell* με ένα κρυφό επίπεδο. Αν δεν υπήρχε ο αλγόριθμος *Levenberg-Marquardt* θα αποτελούσε αντικείμενο για περεταίρω διερεύνηση (Διάγραμμα 21).

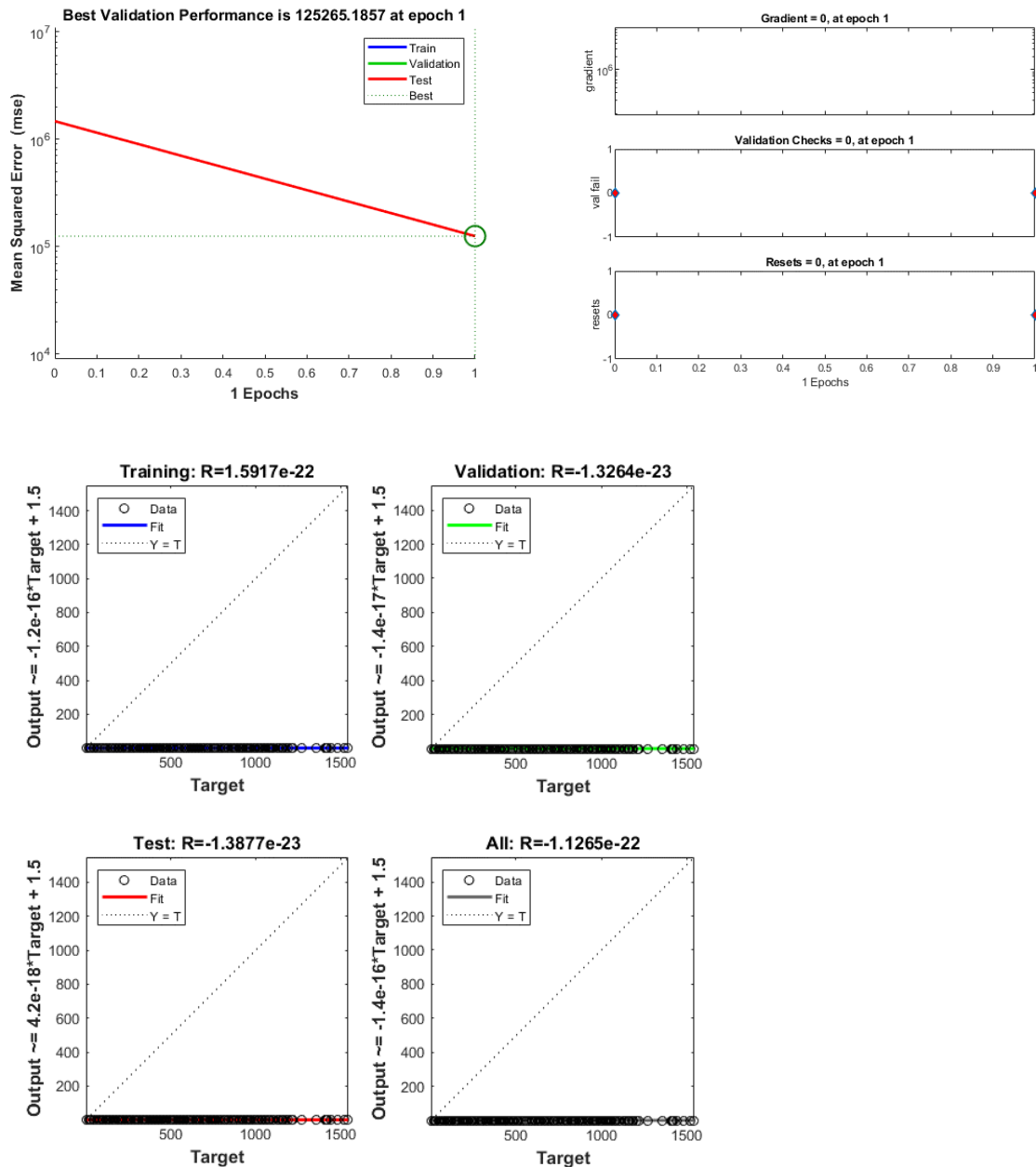
Αλγόριθμος βαθμισμένης Σ.Κ. με δύο κρυφά επίπεδα



Διάγραμμα 22: Αποτελέσματα του αλγόριθμου βαθμισμένης συζυγούς κλίσης με δύο κρυφά επίπεδα

Και αυτή η μέθοδος παρουσίασε αντίστοιχη βελτίωση με την προσθήκη ενός δεύτερου κρυφού επιπέδου. Το μόνο μειονέκτημα της ήταν ότι το κόστος υπολογισμού αυξήθηκε σημαντικά σε σχέση με την χρήση ενός επιπέδου. Γενικά σε όλες τις μεθόδους συζυγούς κλίσης, υπήρξε αύξηση του υπολογιστικού κόστους, αλλά η αύξηση του απαιτούμενου χρόνου εκπαίδευσης, ήταν της τάξης του 25%, στην περίπτωση της βαθμισμένης συζυγούς κλίσης η αντίστοιχη αύξηση πλησίαζε το 50% (Διάγραμμα 22).

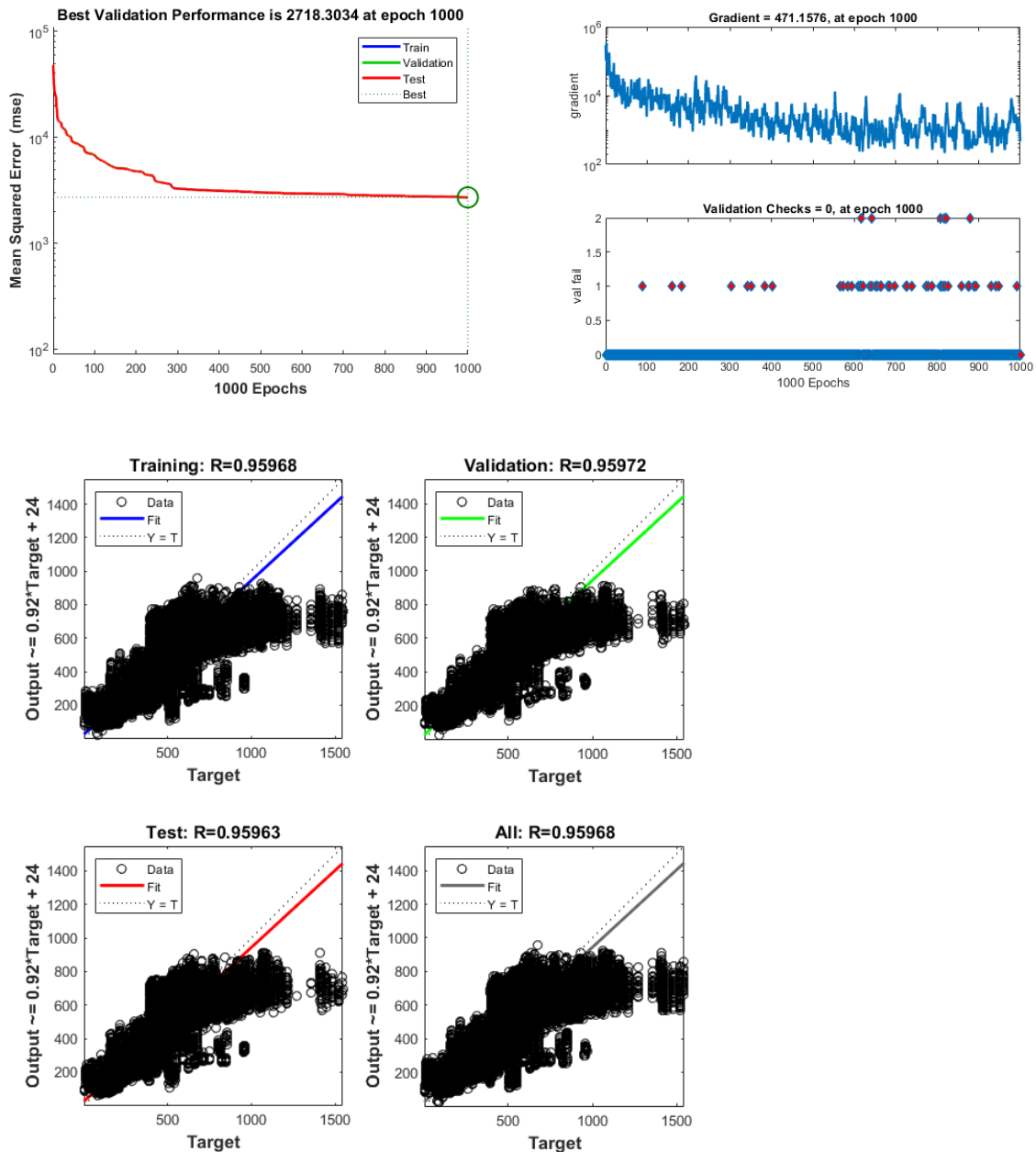
Αλγόριθμος *Quasi-Newton* με δύο κρυφά επίπεδα



Διάγραμμα 23: Αποτελέσματα του αλγόριθμου *Quasi-Newton* με δύο κρυφά επίπεδα

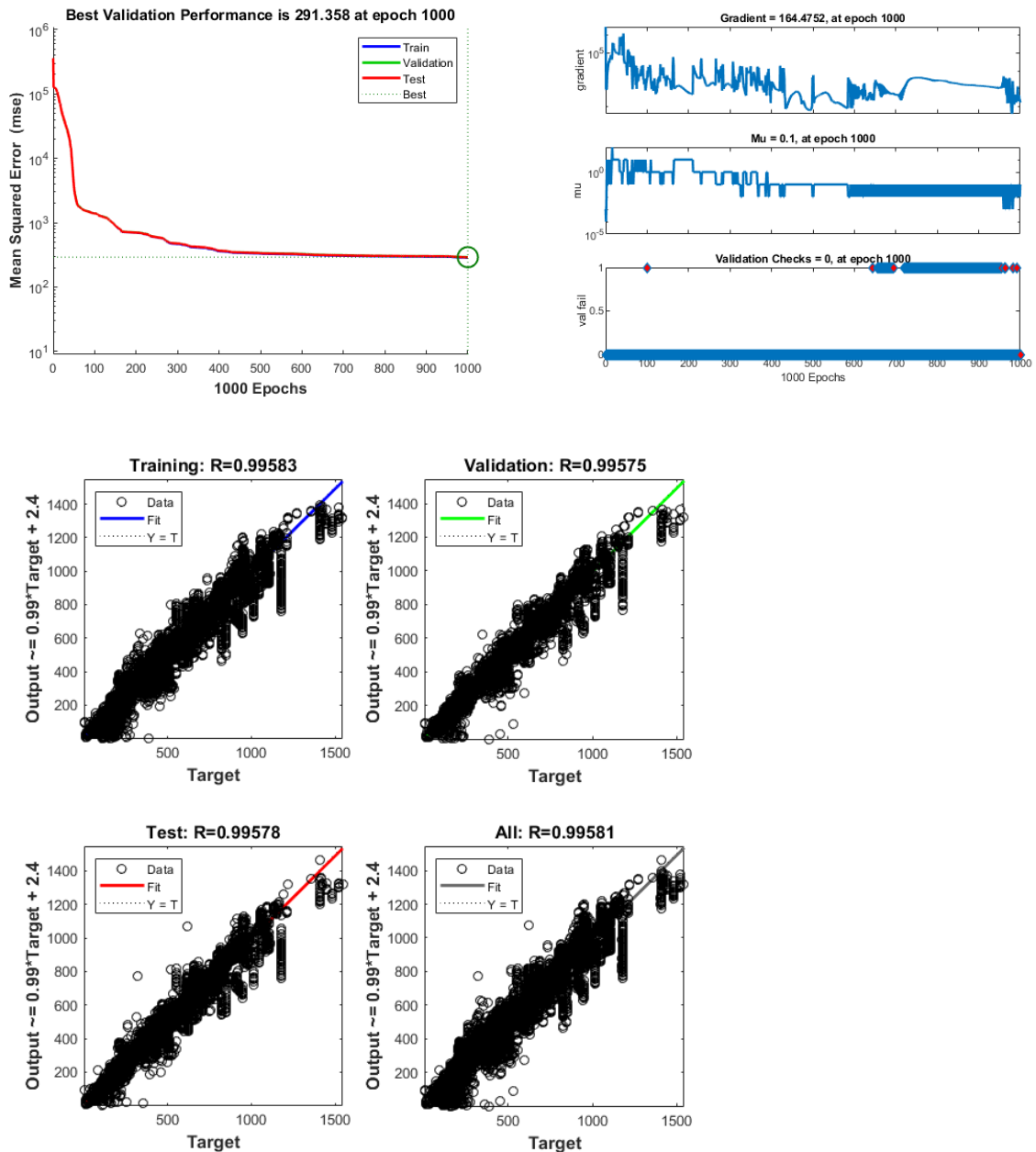
Απογοητευτική ήταν και η απόδοση αυτού του αλγορίθμου, ο οποίος εγκλωβίστηκε σε τοπικό ελάχιστο στην πρώτη εποχή, οδηγώντας σε τερματισμό της εκπαίδευσης λόγω μηδενισμού της κλίσης (Διάγραμμα 23).

Αλγόριθμος *One-Step Secant* με δύο κρυφά επίπεδα

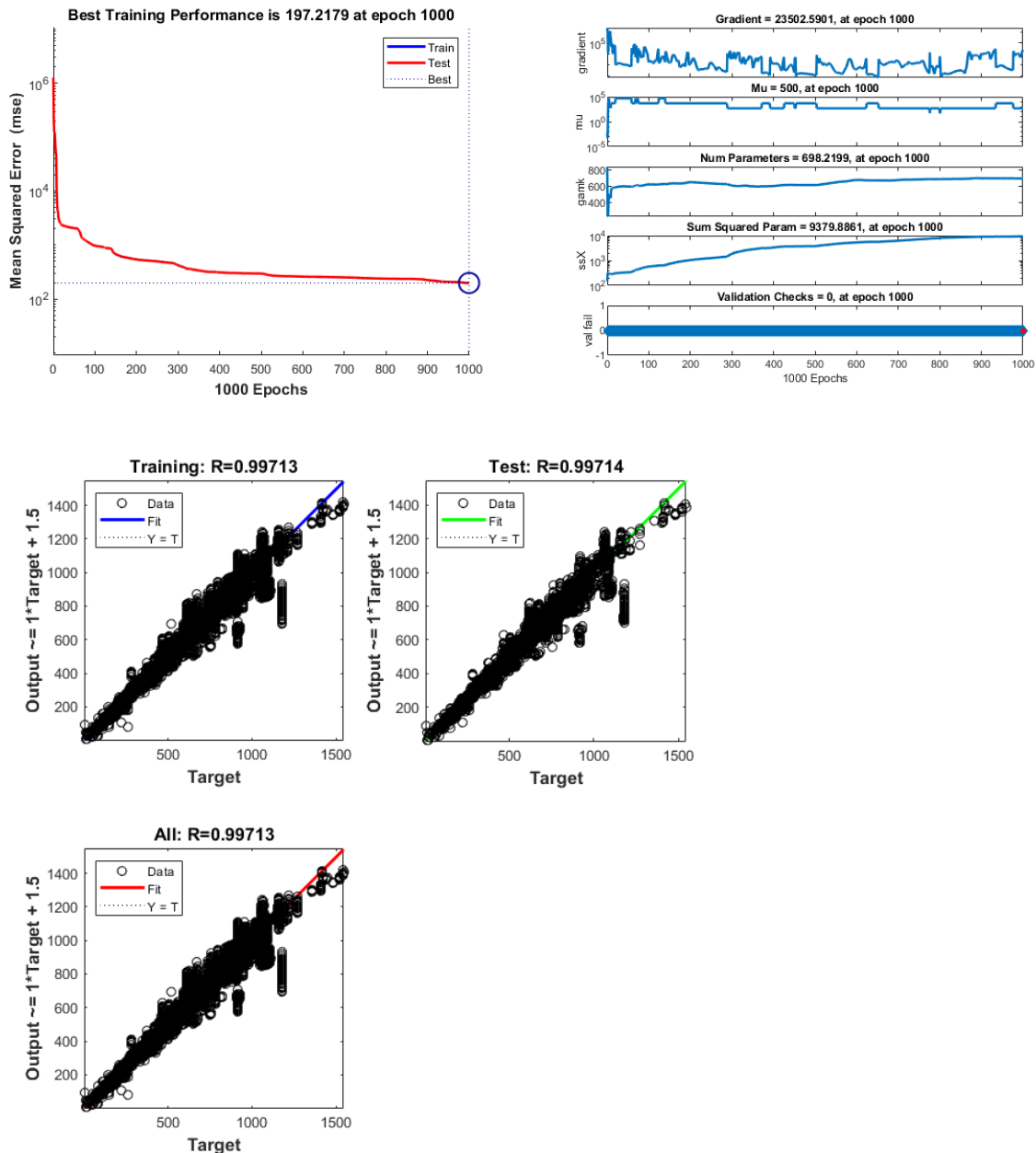


Διάγραμμα 24: Αποτελέσματα του αλγόριθμου *One-Step Secant* με δύο κρυφά επίπεδα

Εμφανής η σημαντική βελτίωση της απόδοσης και αυτού του αλγορίθμου με την προσθήκη ενός δεύτερου επιπέδου, αφού πλησίασε αρκετά της μεθόδου των συζυγών κλίσεων. Το υπολογιστικό κόστος όμως ήταν αρκετά υψηλό, αφού χρειάστηκε τον διπλάσιο χρόνο εκπαίδευσης από την *Polak-Ribiere* και τον τετραπλάσιο από την μέθοδο βαθμισμένης συζυγούς κλίσης (Διάγραμμα 24).

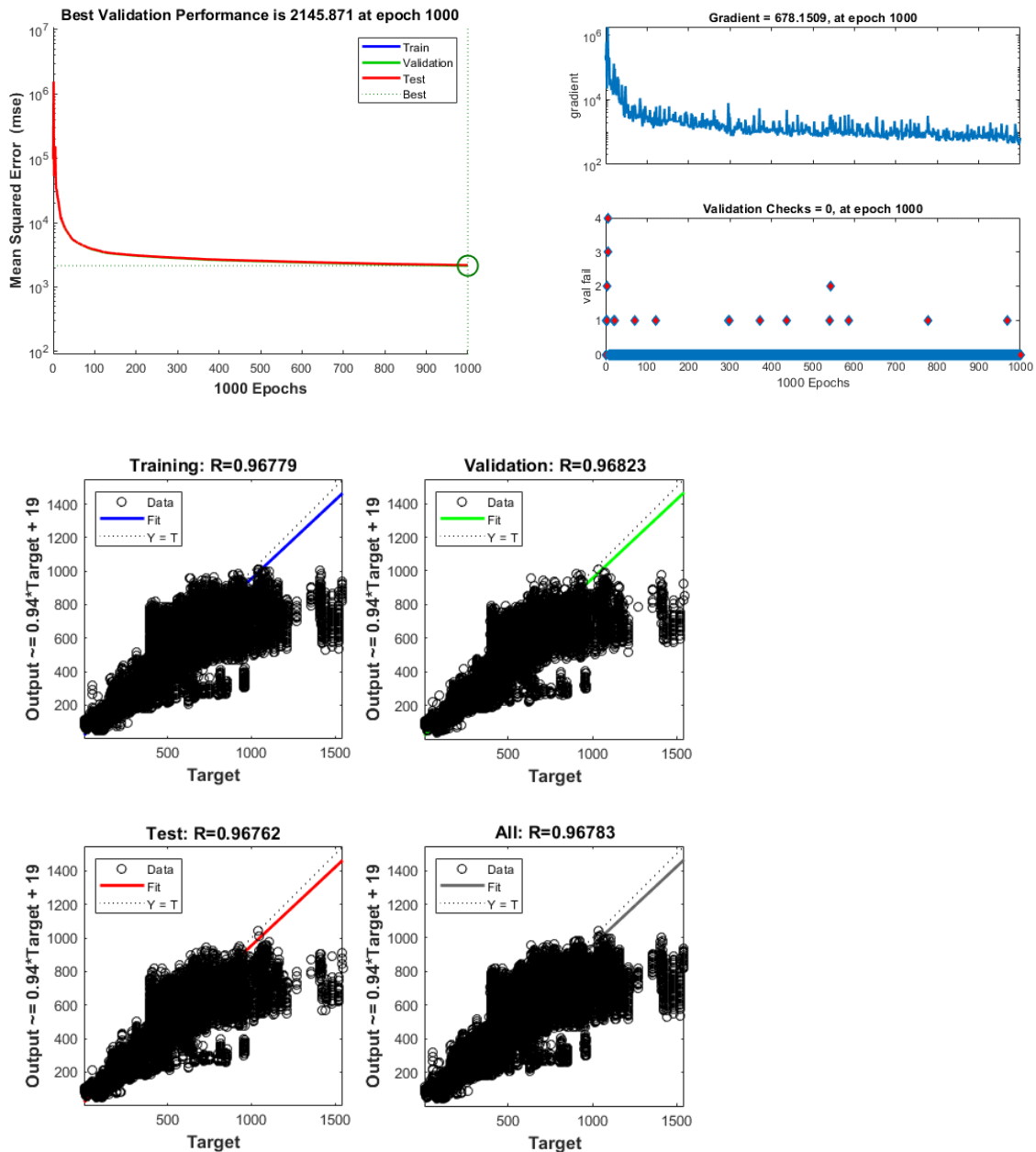
Αλγόριθμος Levenberg-Marquardt με δύο κρυφά επίπεδα*Διάγραμμα 25: Αποτελέσματα του αλγόριθμου Levenberg-Marquardt με δύο κρυφά επίπεδα*

Ο συγκεκριμένος αλγόριθμος επιβεβαίωσε ότι θα δώσει τα καλύτερα αποτελέσματα, επίσης επιβεβαιώνει την ανάγκη προσθήκης περισσότερων κρυφών επιπέδων, αφού με την προθήκη ενός μόνο επιπλέον επιπέδου υπήρξε βελτίωση της τάξης του 80%. Πλέον είναι εμφανές ότι τα αποτελέσματα τείνουν να μπουν στην διαγώνιο, μειώνοντας το χαρακτηριστικό σύννεφο που παρατηρούνταν στα προηγούμενα διαγράμματα. Στα αρνητικά της προσθήκης δεύτερου κρυφού επιπέδου, ήταν ότι ο χρόνος εκπαίδευσης υπερτριπλασιάστηκε (Διάγραμμα 25).

Αλγόριθμος *Bayesian Regularization* με δύο κρυφά επίπεδαΔιάγραμμα 26: Αποτελέσματα του αλγόριθμου *Bayesian Regularization* με δύο κρυφά επίπεδα

Αντίστοιχη βελτίωση παρουσίασε και η δεύτερη εκδοχή του αλγορίθμου *Levenberg-Marquardt*. Το γεγονός ότι δεν είναι απαραίτητο να υπάρχει υποσύνολο επικύρωσης, αφού η γενίκευση του σφάλματος γίνεται μέσω κανονικοποίησης, δημιούργησε μεγαλύτερο υποσύνολο εκπαίδευσης, κάτι που μεταφράζεται και μικρή βελτίωση των αποτελεσμάτων σε σχέση με την εκδοχή του αλγορίθμου με χρήση πρόωρου τερματισμού (Διάγραμμα 26).

Αλγόριθμος *RProp* με δύο κρυφά επίπεδα



Διάγραμμα 27: Αποτελέσματα του αλγόριθμου Ανθεκτικής Διάδοσης με δύο κρυφά επίπεδα

Σημαντική βελτίωση υπήρξε και σε αυτόν τον αλγόριθμο, που σε βάση αποτελεσμάτων ήταν ο καλύτερος μετά τις εκδοχές του *Levenberg-Marquardt*. Βέβαια σημαντική ήταν και η αύξηση του υπολογιστικού κόστους, αφού ο χρόνος εκπαίδευσης διπλασιάστηκε (Διάγραμμα 27).

ΣΥΓΚΡΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

Συνοπτικά τα αποτελέσματα των αλγορίθμων παρουσιάζονται στον παρακάτω πίνακα

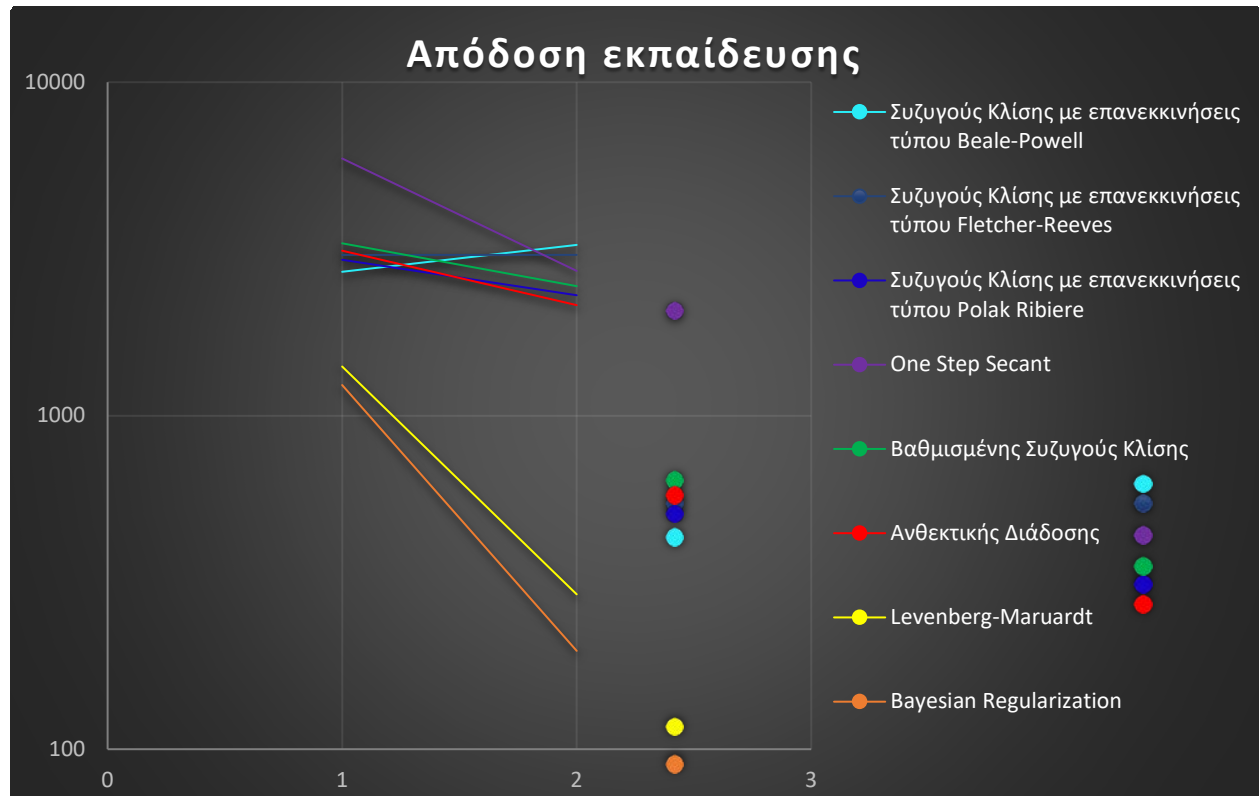
Πίνακας 3: Συνοπτικά αποτελέσματα της εκπαίδευσης ενός Τ.Ν.Α. για την περιοχή μελέτης με χρήση των διαθέσιμων αλγορίθμων του MATLAB. Παρουσιάζεται η απόδοση των αλγορίθμων σε MSE και απαιτούμενο χρόνο εκπαίδευσης.

Μέθοδος	Ένα κρυφό επίπεδο (MSE)	Δύο κρυφά επίπεδα (MSE)	Ένα κρυφό επίπεδο (χρόνος- min)	Δύο κρυφά επίπεδα (χρόνος- min)
Απότομη Κάθοδος	1100025	703791	0	0
Α.Κ. με Ορμή	456057	125382	0	0
Α.Κ. με M.P.M.	15198	125510	6	0
Α.Κ. με Ορμή και M.P.M.	125231	6857	1	98
Σ. Κ. με επαν. τ. <i>Beale-Powell</i>	2702	3252	354	77
Σ. Κ. με επαν. τ. <i>Fletcher-Reeves</i>	3035	3036	268	347
Σ. Κ. με επαν. τ. <i>Polak Ribiere</i>	2935	2298	277	399
Βαθμισμένης Συζυγούς Κλίσης	3289	2447	69	184
<i>BFGS-Quasi-Newton</i>	7971	125265	1538	1
<i>One Step Secant</i>	5907	2718	371	724
<i>Levenberg-Marquardt</i>	1405	291	1150	4319
<i>Bayesian Regularization</i>	1236	197	1277	4549
Ανθεκτικής Διάδοσης	3125	2146	46	92

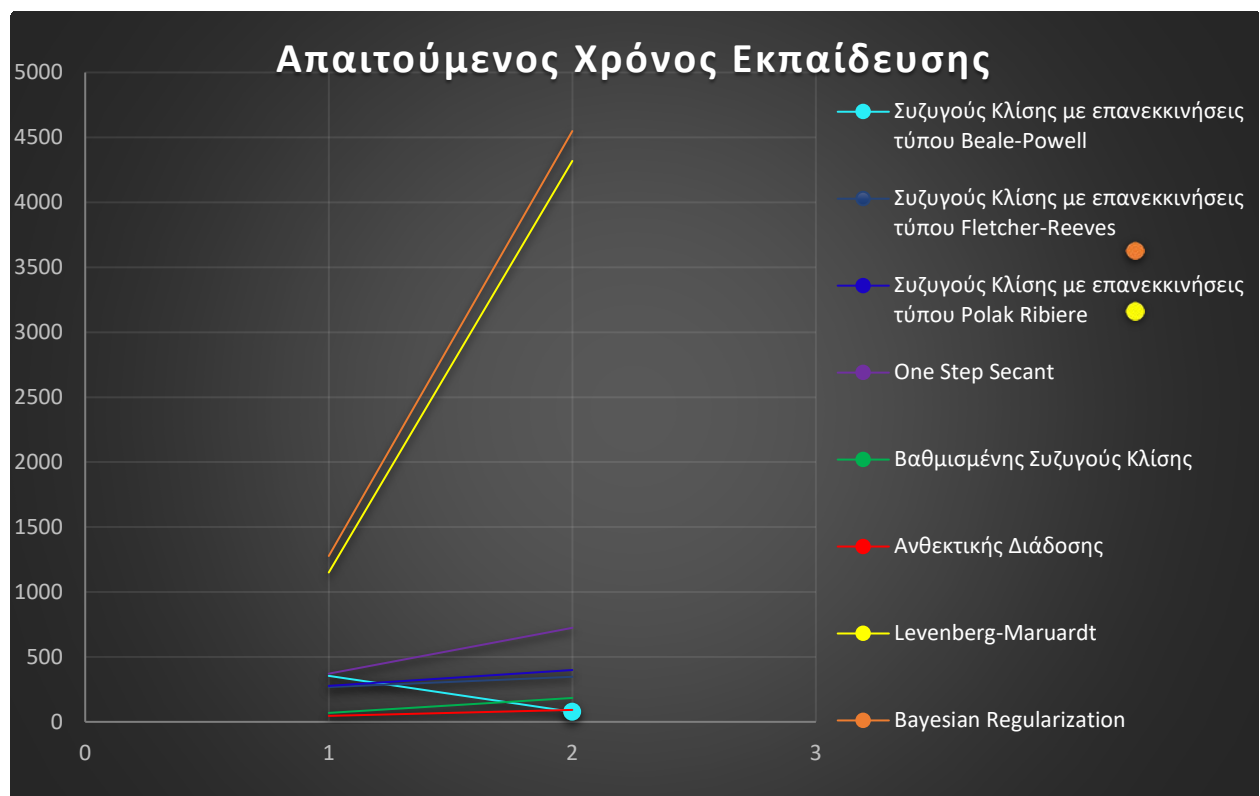
Όπως αναφέρθηκε και στην παρουσίαση των αποτελεσμάτων, αλλά και είναι εμφανές στο [Διάγραμμα 28](#), οι αλγόριθμοι που χρησιμοποιούν την μέθοδο *Levenberg-Marquardt*, έδωσαν πολύ καλύτερα αποτελέσματα σε σχέση με τους υπόλοιπους αλγόριθμους. Επίσης είχαν και συγκριτικά την μεγαλύτερη βελτίωση με την προσθήκη ενός δεύτερου κρυφού επιπέδου. Κάτι που ευνόησε σχεδόν όλους τους αλγόριθμους εκπαίδευσης.

Από την άλλη, σημαντικό μειονέκτημα των μεθόδων *Levenberg-Marquardt*, ήταν οι απαιτήσεις τους σε χρόνο εκπαίδευσης ([Διάγραμμα 29](#)). Με την προσθήκη του δεύτερου κρυφού επιπέδου, οι χρόνοι εκπαίδευσης τους, σχεδόν τριπλασιάστηκαν, φτάνοντας τις τρεις μέρες. Αν και ο χρόνος, σαν τιμή, δεν μπορεί να αποτελέσει ασφαλές κριτήριο, καθώς καθοριστικός παράγοντας στην απαίτηση του χρόνου είναι και η υπολογιστική δύναμη του τερματικού, έτσι ένας υπερυπολογιστής σε αυτό το πρόβλημα θα χρειαζόταν κάποια λεπτά για την ολοκλήρωση του. Όμως η σύγκριση της απόδοσης του αλγορίθμου με τον απαιτούμενο χρόνο, μπορεί να δώσει σημαντικές πληροφορίες για το ποιος αλγόριθμος μπορεί να αποδώσει καλύτερα σε δεδομένο πρόβλημα με βάση τις δυνατότητες του εκάστοτε τερματικού. Με αυτή την λογική εκτός του αλγορίθμου *Levenberg-Marquardt*, η μέθοδος της Ανθεκτικής διάδοσης

ήταν η πιο ενδιαφέρουσα από τις υπόλοιπες, καθώς κατάφερε να έχει την τρίτη καλύτερη όσον αφορά το MSE και την πρώτη όσον αφορά τον απαιτούμενο χρόνο εκπαίδευσης.



Διάγραμμα 28: Συγκριτικό διάγραμμα των αλγορίθμων με κριτήριο την απόδοση



Διάγραμμα 29: Συγκριτικό διάγραμμα των μεθόδων με κριτήριο το χρόνο σε λεπτά

Η ΙΔΙΑΙΤΕΡΗ ΠΕΡΙΠΤΩΣΗ ΤΗΣ ΑΝΘΕΚΤΙΚΗΣ ΔΙΑΔΟΣΗΣ

Ο αλγόριθμος της ανθεκτικής διάδοσης αποδείχθηκε αρκετά γρήγορος και έδινε αρκετά ικανοποιητικά αποτελέσματα εν συγκρίσει του απαιτούμενου χρόνου εκπαίδευσης. Για το λόγο, υπήρχε η δυνατότητα να δοκιμαστούν διάφορες αρχιτεκτονικές για μελετηθεί περεταίρω. Συγκεκριμένα έγιναν δοκιμές για χρήση της με τέσσερα, έξι, οκτώ και δεκαέξι κρυφά επίπεδα. Ενώ δοκιμάστηκε η συμπεριφορά της και σε περισσότερες εποχές αφού έτρεξε μέχρι και για 16,000 εποχές¹. Επίσης δοκιμάστηκε και η απόδοση της αν διπλασιαστούν οι κόμβοι στα επίπεδα. Τα αποτελέσματα των δοκιμών αναφέρονται στον Πίνακα 4.

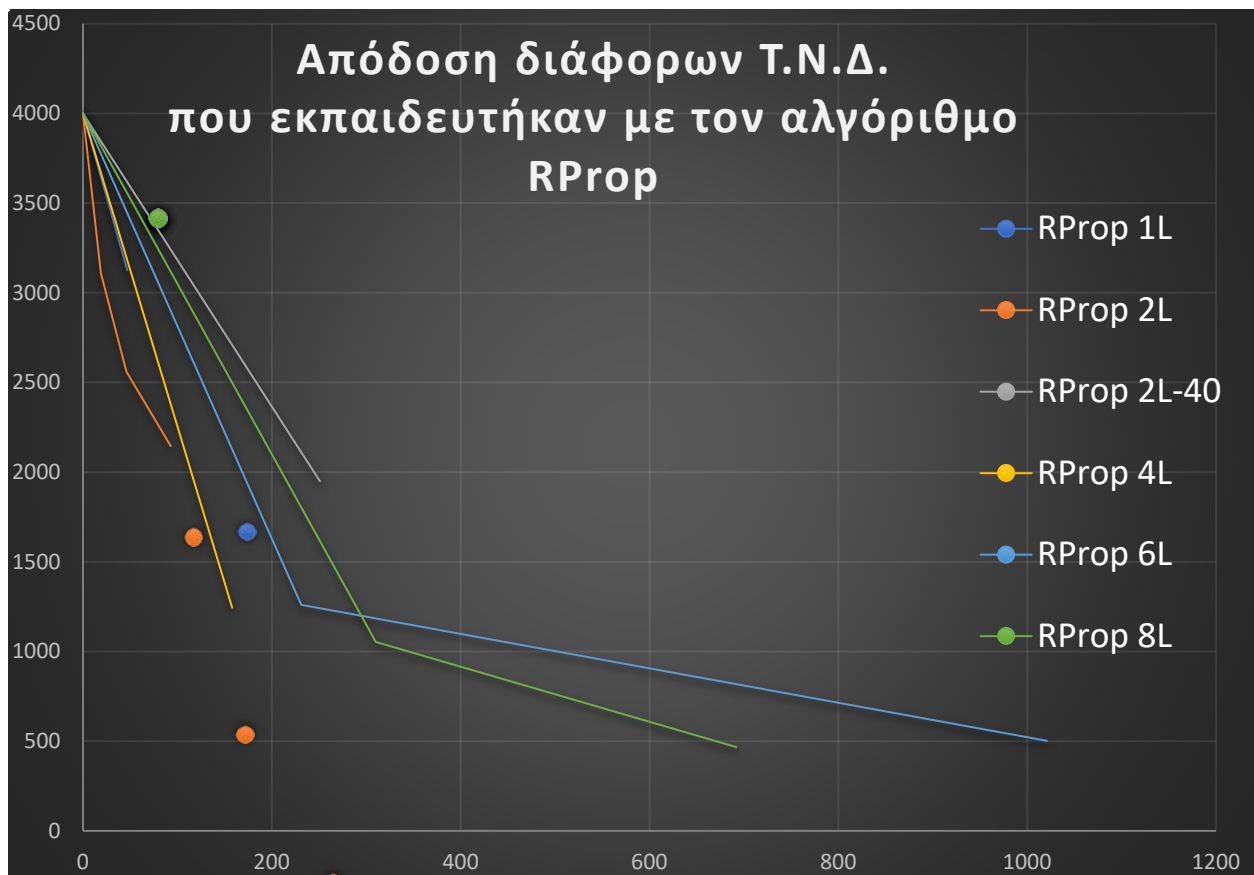
Πίνακας 4: Συγκεντρωτικά αποτελέσματα της χρήσης του αλγορίθμου ανθεκτικής διάδοσης

Δοκιμή	MSE	Χρόνος (min)
1L 1000 epoch	3125	47
2L 1000 epoch	2145	93
2L 1000 epoch - 40	1949	251
4L 1000 epoch	1244	158
6L 4000 epoch	502	1021
8L 1000 epoch	1054	310
8L 2000 epoch	467	692
8L 16000 epoch	160	5603

Με χρήση των παραπάνω στοιχείων καθώς και με διάφορα ζεύγη τιμών που είχαν κρατηθεί κατά την διάρκεια της εκπαίδευσης των νευρωνικών, δημιουργήθηκε ένα υποτυπώδες διάγραμμα απόδοσης της μεθόδου αυτής ως προς το χρόνο (Διάγραμμα 30).

Ως γενική τάση φαίνεται ότι η αύξηση των επιπέδων ευνοεί την καλύτερη εκπαίδευση στον ίδιο χρόνο. Την ώρα που η αύξηση των κόμβων στο επίπεδο, βελτιώνει την εκπαίδευση, αλλά όχι τόσο συγκριτικά με το υπολογιστικό κόστος. Συγκεκριμένα, συγκρίνοντας την δοκιμή με 2L και 20 κόμβους στο επίπεδο, με την δοκιμή με 40 κόμβους στο επίπεδο, ο διπλασιασμός των κόμβων στο ίδιο επίπεδο αύξησε τον χρόνο εκπαίδευσης κατά 170%, την ώρα που το MSE μειώθηκε 9%. Αντίθετα, ο διπλασιασμός των επιπέδων από δύο σε τέσσερα, προκάλεσε αύξηση του χρόνου εκπαίδευσης κατά 70% προκαλώντας όμως μείωση αντίστοιχου μεγέθους στο MSE, το οποίο μειώθηκε κατά 40%.

¹ Η δοκιμή με 16 επίπεδα εγκαταλείφθηκε πρόωρα, καθώς η απόδοση του αλγορίθμου δεν ήταν καλύτερη από την απόδοση με 8 επίπεδα, την ώρα που ο απαιτούμενος χρόνος εκπαίδευσης ήταν υπερδιπλάσιος.



Διάγραμμα 30: Σύγκριση αποτελεσμάτων εκπαίδευσης ως προς το χρόνο, T.N.Δ. που εκπαιδεύτηκαν με τον αλγόριθμο της ανθεκτικής διάδοσης¹. Επειδή δεν υπάρχουν στοιχεία για το σφάλμα κατά την έναρξη της εκπαίδευσης, δόθηκε σε όλους τους αλγορίθμους η ίδια αυθαίρετη τιμή.

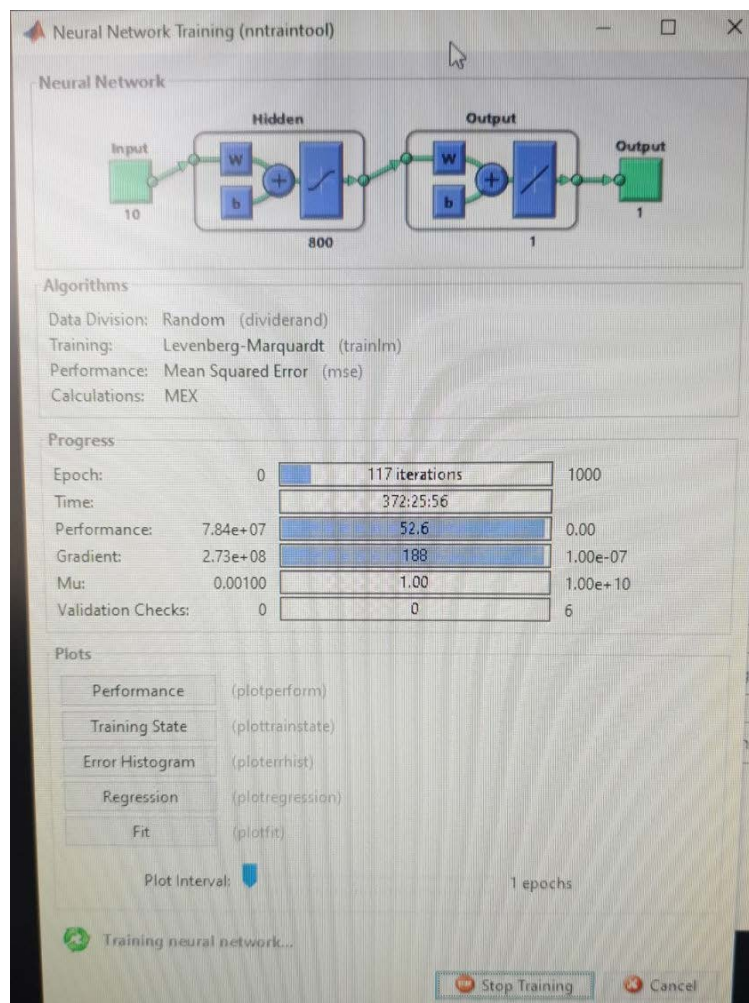
Η τελευταία δοκιμή με τα 8 επίπεδα για 16,000 εποχές, έδωσε καλύτερα αποτελέσματα ακόμα και από τις μεθόδους *Levenberg-Marquardt*. Όμως ο απαιτούμενος χρόνος για την εκπαίδευση του συγκεκριμένου νευρωνικού δικτύου, ήταν μεγαλύτερος από τον απαιτούμενο χρόνο για την εκπαίδευση των νευρωνικών δικτύων με τους αλγορίθμους *Levenberg-Marquardt*, καθώς ξεπέρασε τις 93 ώρες, την ώρα που ο αλγόριθμος με κανονικοποίηση *Bayesian*, χρειάστηκε λιγότερες από 76 ώρες για να δώσει παραπλήσια αποτελέσματα.

Εκτός του παραπάνω, δύο άλλοι παράγοντες οδήγησαν στο να μην εξερευνηθεί περεταίρω αυτή η μέθοδος. Ο πρώτος αφορά το γεγονός ότι ο συγκεκριμένος αλγόριθμος αδυνατούσε να συνεχίσει την εκπαίδευση ενός νευρωνικού αν αυτή σταματούσε, κάτι που θα εξηγηθεί παρακάτω πόσο σημαντικό είναι και ο δεύτερος αφορά την εξάντληση των αρχιτεκτονικών που θα μπορούσαν να βελτιώσουν την απόδοση του, όπως φάνηκε ακόμα και τα οκτώ επίπεδα είναι πολύ κοντά σε απόδοση με τα έξι, ενώ τα δεκαέξι ήταν εμφανώς λιγότερο αποτελεσματικά από τα οκτώ. Επομένως ακόμα και αν υπάρχει κάποιος αριθμός επιπέδων ενδιάμεσα που να δίνει καλύτερα αποτελέσματα, αυτά δεν θα είναι σημαντικά καλύτερα. Ενώ όπως φάνηκε με την αύξηση των κόμβων στο ίδιο επίπεδο, δεν θα υπήρχε σημαντική μείωση του MSE, αν αυξανόντουσαν οι κόμβοι στα οκτώ επίπεδα.

¹ Στο διάγραμμα δεν συμπεριλήφθηκε η τιμή για την εκπαίδευση T.N.Δ. με οκτώ επίπεδα για 16,000 εποχές, ώστε να αποτυπώνονται καλύτερα οι διαφορές μεταξύ των μεθόδων.

ΠΕΡΕΤΑΙΡΩ ΔΟΚΙΜΕΣ ΤΗΣ ΜΕΘΟΔΟΥ *LEVENBERG-MARQUARDT*

Όπως φάνηκε από τα προηγούμενα αποτελέσματα, η χρήση των αλγορίθμων που ακολουθούν την μέθοδο *Levenberg-Marquardt*, είναι η ενδεδειγμένη. Με βάση το παγκόσμιο θεώρημα προσέγγισης (*universal approximation theory*) των νευρωνικών δικτύων, κάθε συνεχής συνάρτηση μπορεί να προσεγγιστεί με χρήση ενός νευρωνικού δικτύου με δύο κρυφά επίπεδα, φθάνοντας οποιαδήποτε επιθυμητή προσέγγιση. Το πρόβλημα είναι ότι οι απαιτούμενοι κόμβοι σε μια τέτοια περίπτωση αυξάνονται εκθετικά, οδηγώντας με την σειρά τους σε εκθετική αύξηση και των απαιτήσεων σε υπολογιστικά συστήματα καθώς και σε χρόνο (Εικόνα 43) (Russel & Norvig, 2020).



Εικόνα 43: Φωτογραφία από μοντέλο με ένα επίπεδο που έτρεξε σε υπολογιστή του εργαστήριου, σε αυτόν με την μεγαλύτερη διαθέσιμη RAM, καθώς οι υπόλοιποι δεν είχαν δυνατότητα εκπαίδευσης με τέτοιο συνολικό αριθμό κόμβων. Είναι εμφανές ότι η ταχύτητα εκπαίδευσης είναι πολύ μικρή. Η εικόνα λήφθηκε μετά από 16 ημέρες εκτέλεσης του μοντέλου, ενώ υπολογιζόταν ότι για να φτάσει τις 1000 εποχές το μοντέλο θα χρειαζόταν περίπου 5 με 6 μήνες. Παρόλα αυτά κρίνοντας από το τελικό αποτέλεσμα αρχιτεκτονικών με περισσότερα επίπεδα, για να μπορούσε το συγκεκριμένο μοντέλο να παρουσιάσει αντίστοιχες επιδόσεις θα χρειαζόταν πολύ μεγαλύτερο αριθμό από 1000 εποχές, με αποτέλεσμα να είναι πρακτικά μη εφαρμόσιμο.

Από τα παραπάνω αυτό που προκύπτει είναι η ανάγκη δημιουργίας νευρωνικών δικτύων με αρκετά πιο πολύπλοκες δομές. Όμως πάλι τα προηγούμενα αποτελέσματα έδειξαν ότι μικρή αύξηση της πολυπλοκότητας της αρχιτεκτονικής τους θα οδηγούσε σε πολλαπλασιασμό του

απαιτούμενου χρόνου για την εκπαίδευση τους. Για να μπορεί να ερευνηθεί ποια αρχιτεκτονική θα ήταν πιο αποτελεσματική για το συγκεκριμένο πρόβλημα, λήφθηκε ένα μικρό υποσύνολο των δεδομένων (25000 ζεύγη δεδομένων), ώστε να εξεταστεί η απόδοση δέκα διαφορετικών αρχιτεκτονικών δόμων, όσον αφορά την μείωση του σφάλματος. Στην συνέχεια οι τρεις πιο αποδοτικές δομές, εξετάστηκαν με ένα μεγαλύτερο υποσύνολο των δεδομένων, ώστε να βρεθεί ποια από τις τρεις ήταν η πιο αποδοτική για το δεδομένο πρόβλημα. Στους ελέγχους αυτούς χρησιμοποιήθηκε ο αλγόριθμος που ενσωματώνει την λειτουργία του πρόωρου τερματισμού, καθώς η δεύτερη εκδοχή που χρησιμοποιεί κανονικοποίηση, είναι πιο εύκολο να προκαλέσει υπερεκπαίδευση, όταν τα δεδομένα που χρησιμοποιούνται για εκπαίδευση δεν είναι επαρκή.

Λόγω του μικρού δείγματος των συνολικών δεδομένων, η εκπαίδευση των T.N.Δ. στα νευρωνικά δίκτυα, σταματούσε λόγω πρόωρου τερματισμού. Για να μην «χαθεί» κάποια πιθανή αρχιτεκτονική που θα μπορούσε να λειτουργήσει πιο αποδοτικά, οι δοκιμές για κάθε δομή πραγματοποιήθηκαν δύο φορές. Στον Πίνακα 5 που ακολουθεί, συνοψίζονται τα αποτελέσματα των δοκιμών αυτών.

Πίνακας 5: Δοκιμές διαφόρων αρχιτεκτονικών T.N.Δ. με 25000 ζεύγη δεδομένων. Με έντονο χρωματισμό οι 3 καλύτερες μέθοδοι σε μέσο όρο

Αρχιτεκτονική	1 ^η δοκιμή	2 ^η δοκιμή	M.O.
100	2493.9689	2363.2928	2428.631
20-20-20	533.1004	2848.126	1690.613
25-25	837.5204	1070.411	953.9656
25-25-25	438.4029	548.8165	493.6097
25-25-25-25	786.5355	316.2257	551.3806
27-27-27	442.7395	1373.308	908.0236
30-30	891.6503	3391.134	2141.392
30-30-30	948.5639	487.1554	717.8597
40-25	2084.196	6561.091	4322.643
40-40	2661.946	840.1973	1751.072

Από τις αρχιτεκτονικές που δοκιμάστηκαν, οι τρεις πιο αποδοτικές ήταν η χρήση τριών επιπέδων είτε με 25 κόμβους σε κάθε επίπεδο, είτε με 30 κόμβους σε κάθε επίπεδο και η χρήση τεσσάρων επιπέδων με 25 κόμβους ανά επίπεδο. Στην συνέχεια έγινε μια δοκιμή με χρήση ενός μεγαλύτερου ποσοστού των δεδομένων ώστε να είναι πιο αντιπροσωπευτικό το

αποτέλεσμα, για το λόγο αυτό χρησιμοποιήθηκαν 250,000 δεδομένα. Λόγο όμως του αυξημένου υπολογιστικού κόστους, έγινε μόνο μια δοκιμή ανά αλγόριθμο.

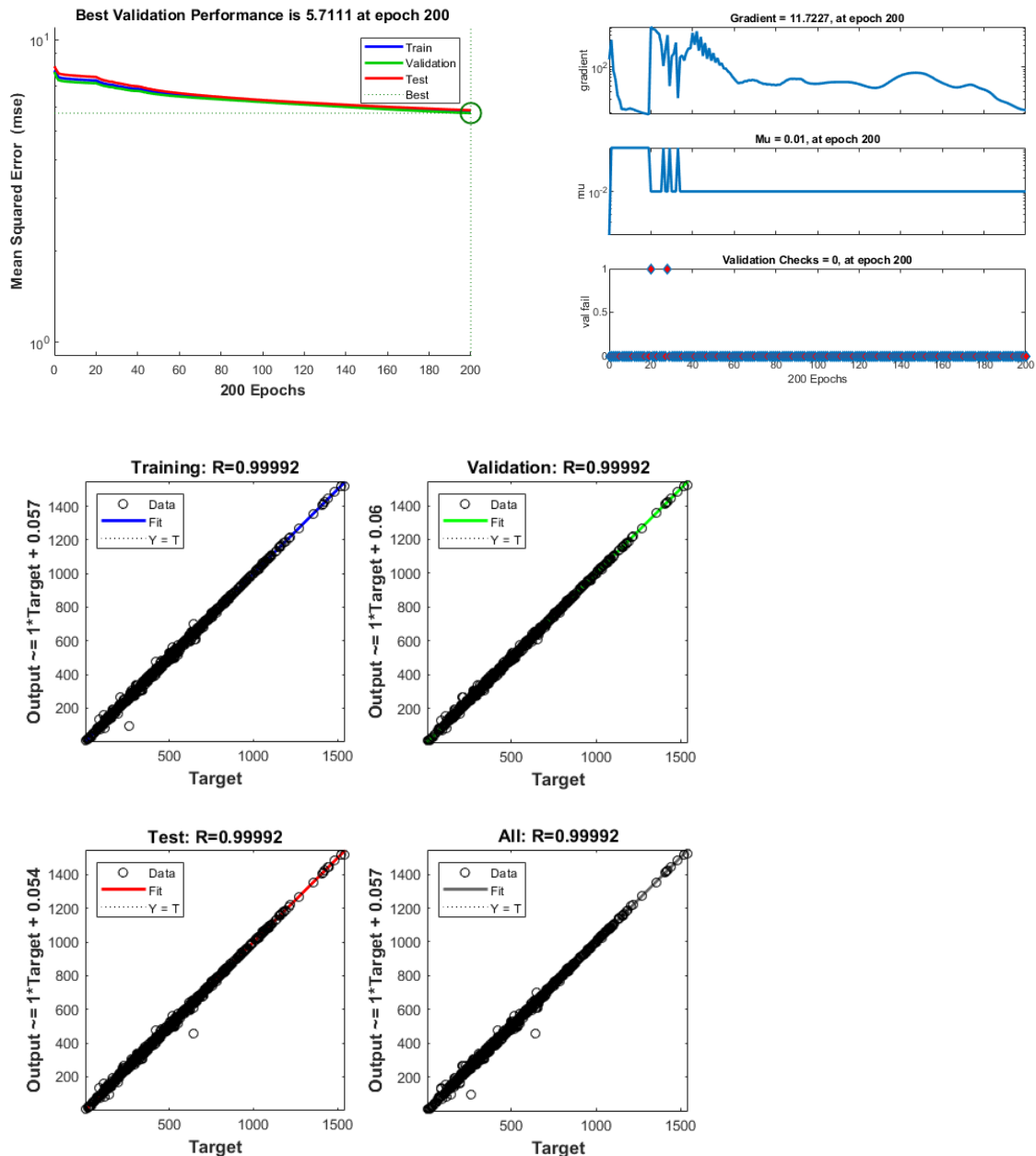
Από τις δοκιμές που έγιναν προέκυψε ότι παρόλο το υπολογιστικό της κόστος, η πιο αποδοτική αρχιτεκτονική, ήταν η χρήση τεσσάρων επιπέδων με 25 νευρώνες το καθένα, η οποία κατάφερε να έχει MSE 25, την ώρα που το T.N.Δ με τρία επίπεδα και 25 νευρώνες που είχε MSE 40 ενώ αυτό με τα τρία επίπεδα και 30 νευρώνες ανά επίπεδο είχε πρόωρο τερματισμό μετά από μόλις 100 εποχές και κατέληξε με MSE 1978.

Ως εκ τούτου αποφασίστηκε η περεταίρω εξερεύνηση της αρχιτεκτονικής με τέσσερα επίπεδα με 25 νευρώνες το κάθε ένα από αυτά. Η συγκεκριμένη αρχιτεκτονική δοκιμάστηκε σε τέσσερα διαφορετικά σενάρια με στόχο την δυνατότητα ενσωμάτωσης των μετεωρολογικών διακυμάνσεων.

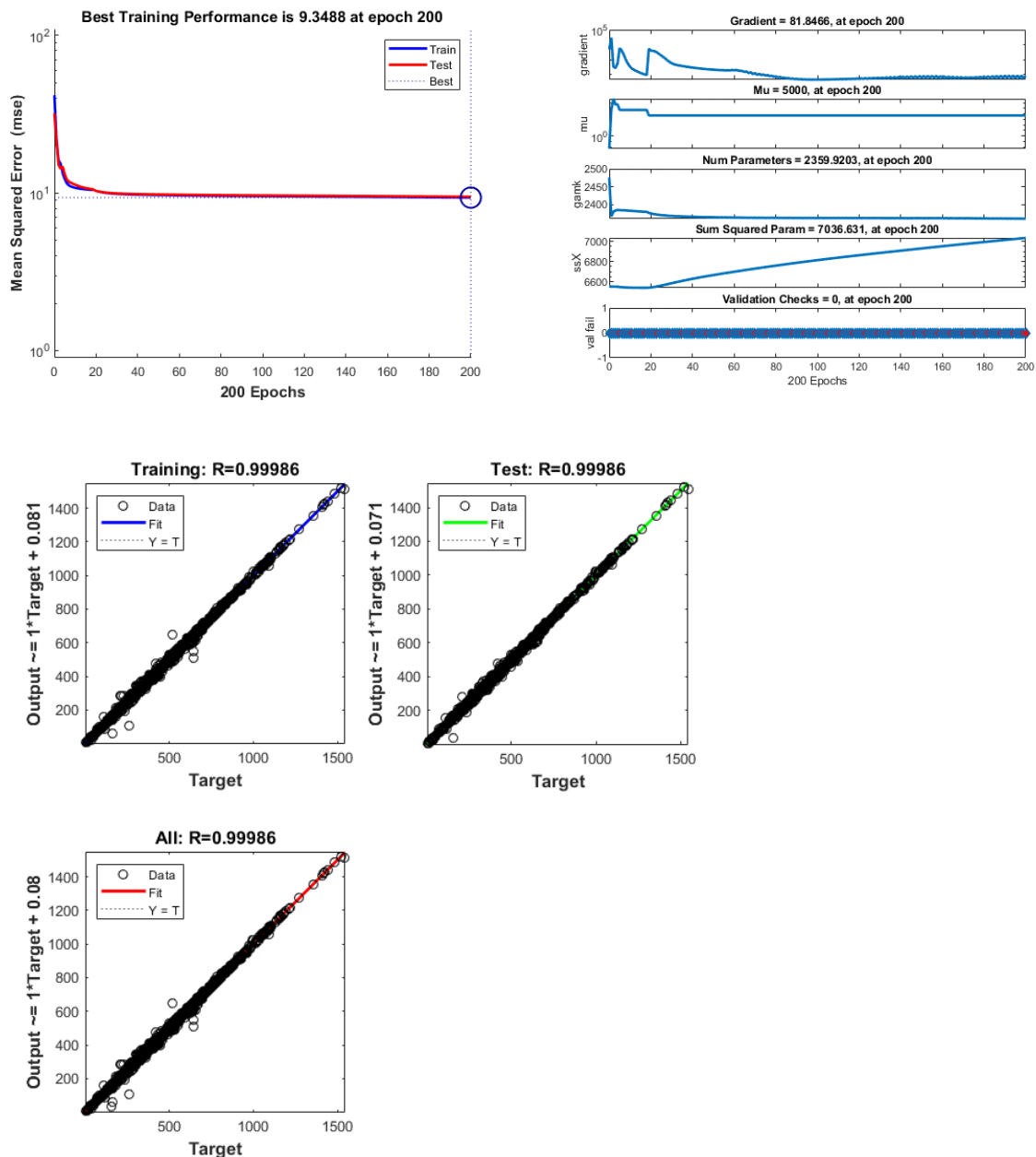
- Στο πρώτο σενάριο, τα δεδομένα εισόδου περιλαμβάναν όλες τις γνωστές μεταβλητές και σαν ζητούμενο ήταν η στάθμη του υδροφόρου ορίζοντα σε σχέση με την επιφάνεια της θάλασσας.
- Στο δεύτερο σενάριο, πάλι τα δεδομένα εισόδου περιλαμβάναν όλες τις γνωστές μεταβλητές και σαν ζητούμενο ήταν η στάθμη του υδροφόρου ορίζοντα σε σχέση με την επιφάνεια του εδάφους.
- Στο τρίτο σενάριο, τα δεδομένα εισόδου περιλαμβάναν όλες τις γνωστές μεταβλητές εκτός του υψομέτρου και σαν ζητούμενο ήταν η στάθμη του υδροφόρου ορίζοντα σε σχέση με την επιφάνεια της θάλασσας.
- Στο τέταρτο σενάριο, τα δεδομένα εισόδου περιλαμβάναν όλες τις γνωστές μεταβλητές εκτός του υψομέτρου και σαν ζητούμενο ήταν η στάθμη του υδροφόρου ορίζοντα σε σχέση με την επιφάνεια του εδάφους.

Τέλος εξετάστηκε και η απόδοση των συγκεκριμένων αλγορίθμων σε ένα σενάριο που τους δινόταν μόνο τα μετεωρολογικά δεδομένα της ίδιας μέρας και σαν ζητούμενο ήταν η στάθμη του υδροφόρου ορίζοντα σε σχέση με την επιφάνεια της θάλασσας. Όστε να εκτιμηθεί αν η παροχή επιπλέον πληροφορίας, βοηθάει στην πρόβλεψη της στάθμης. Επειδή οι μεταβλητές εισόδου σε αυτό το σενάριο ήταν δέκα αντί για είκοσι, χρησιμοποιήθηκε ένα T.N.Δ. με λιγότερους νευρώνες, συγκεκριμένα, ένα T.N.Δ. με τέσσερα επίπεδα και είκοσι κόμβους ανά επίπεδο.

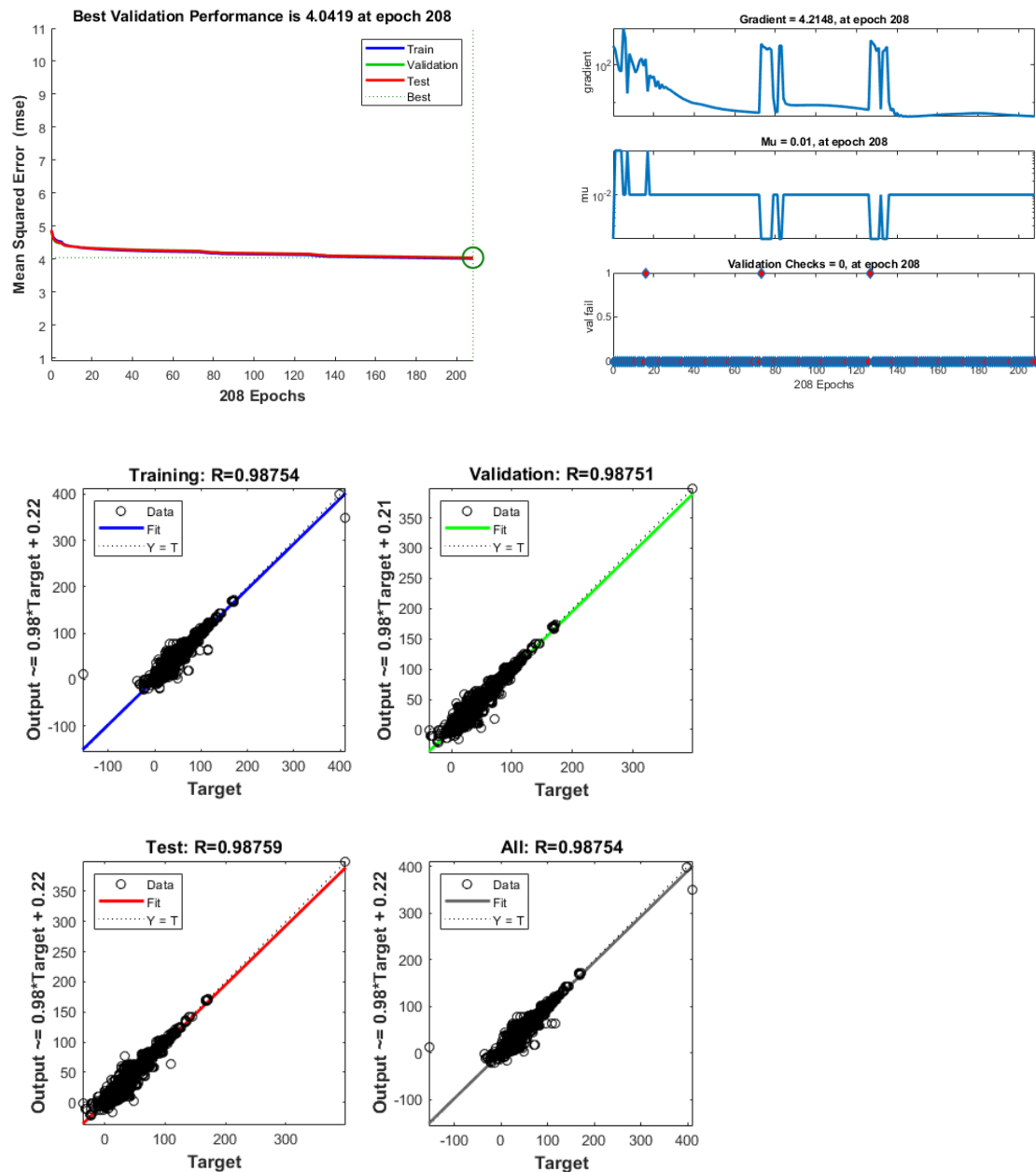
Επειδή οι συγκεκριμένες αρχιτεκτονικές ήταν αρκετά πολύπλοκες και τα τερματικά που ήταν διαθέσιμα δεν είχαν την απαιτούμενη υπολογιστική ισχύ, ήταν αδύνατο να γίνει εκπαίδευση των νευρωνικών από το μηδέν. Για το λόγο αυτό, χρησιμοποιήθηκαν αρχικά κάποια υποσύνολα ώστε να προκύψουν κάποιες αρχικές τιμές στα βάρη των νευρώνων που να είναι σχετικά κοντά στην βέλτιστη λύση και στην συνέχεια έγινε επανεκπαίδευση των δικτύων με το σύνολο των δεδομένων, για διακόσιες εποχές αντί για χίλιες.

1^ο ΣΕΝΑΡΙΟΑλγόριθμος Levenberg-MarquardtΔιάγραμμα 31: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 1^ο σενάριο

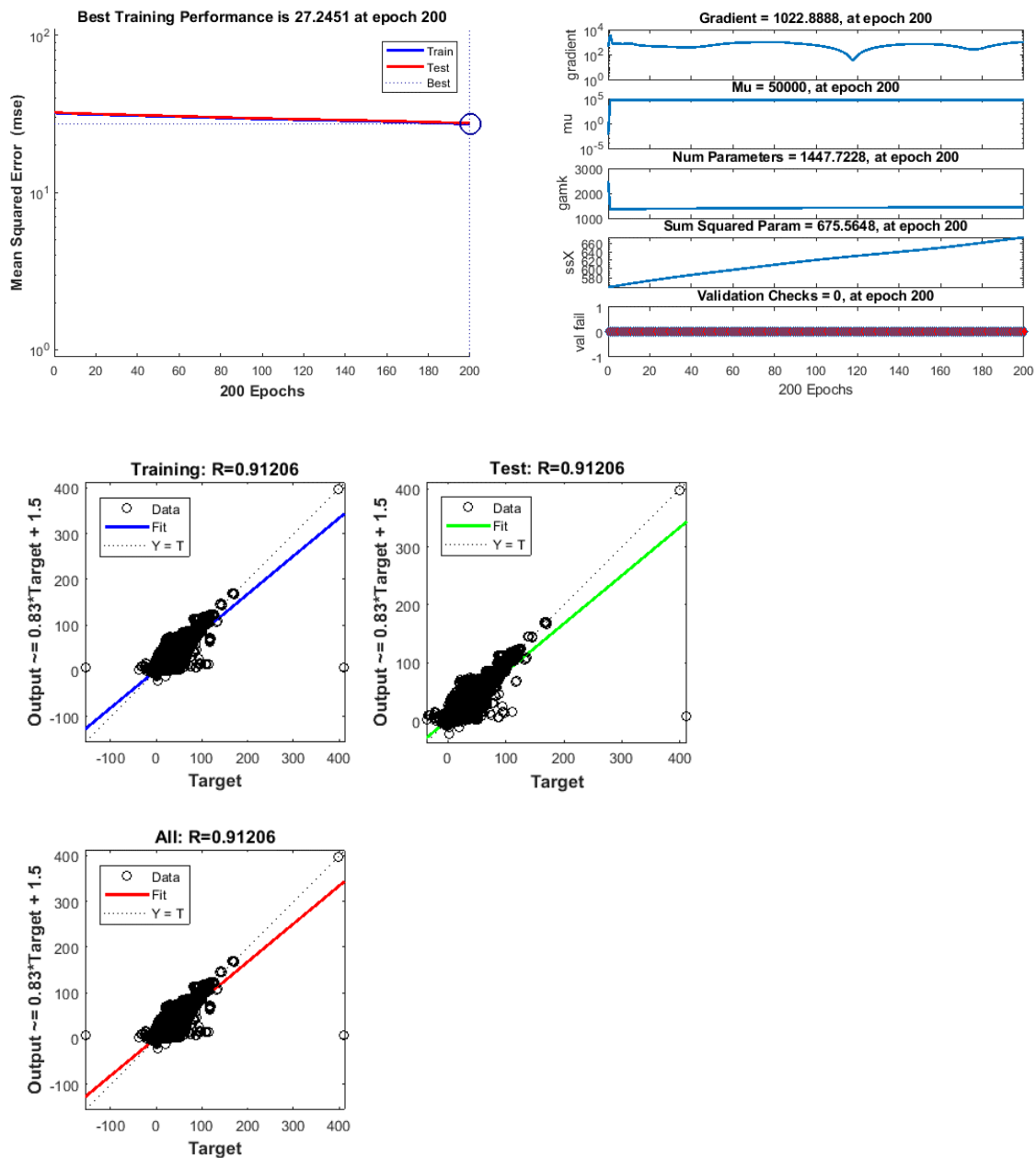
Από την αρχική εικόνα που δίνει το *MATLAB*, φαίνεται ότι η εκπαίδευση του T.N.Δ. ήταν πολύ καλή. Η χρήση των αποτελεσμάτων από προηγούμενες εκπαιδεύσεις με λιγότερα δεδομένα (αρχικά χρησιμοποιήθηκε ένα σετ με 250,000 δεδομένα για 1115 εποχές, έχοντας *MSE* 6.02) βοήθησε τον αλγόριθμο να ξεκινήσει σχετικά κοντά στην λύση (από 7.9) και μετά από 200 εποχές (και 120 ώρες) να έχει μια απόδοση 5.7 *MSE* (Διάγραμμα 31).

Αλγόριθμος Bayesian RegularizationΔιάγραμμα 32: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 1^ο σενάριο

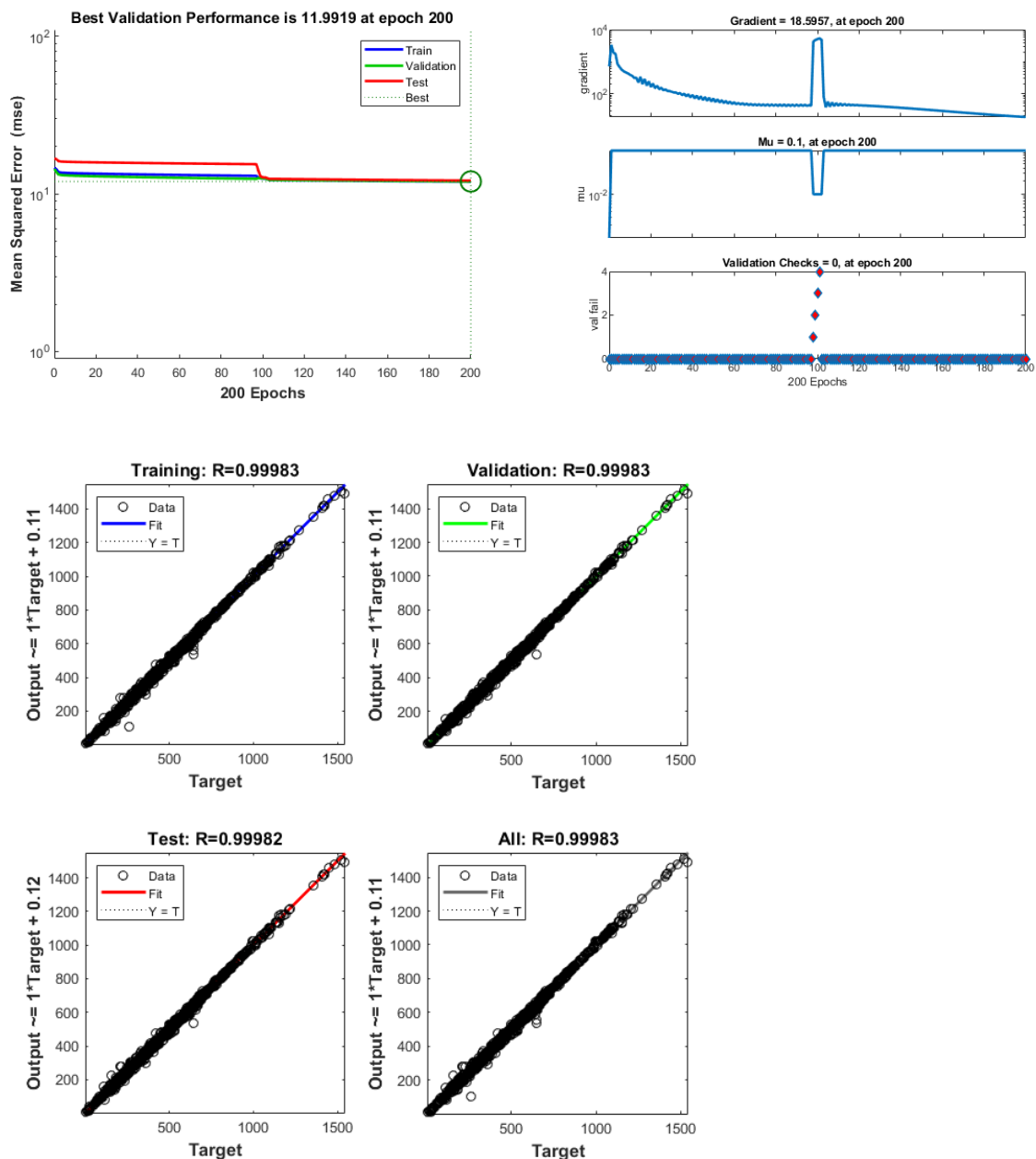
Αρχικά χρησιμοποιήθηκε ένα μικρό υποσύνολο με 25,000 δεδομένα ώστε να γίνει μια προ-εκπαίδευση για 2000 εποχές. Το MSE κατέβηκε στο 8.6, αλλά στην συνέχεια όταν άρχισε η επανεκπαίδευση το MSE εκτινάχθηκε στο 42, αυτό σημαίνει ότι αρχικά το T.N.Δ. είχε υπερπροσαρμοστεί στο μικρό υποσύνολο και ενσωμάτωνε θόρυβο. Μετά από λίγες εποχές το νευρωνικό βελτίωσε την λύση του και έφτασε στα ίδια επίπεδα MSE με αυτά της προ-εκπαίδευσης (Διάγραμμα 32).

2^ο ΣΕΝΑΡΙΟΑλγόριθμος Levenberg-MarquardtΔιάγραμμα 33: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 2^ο σενάριο

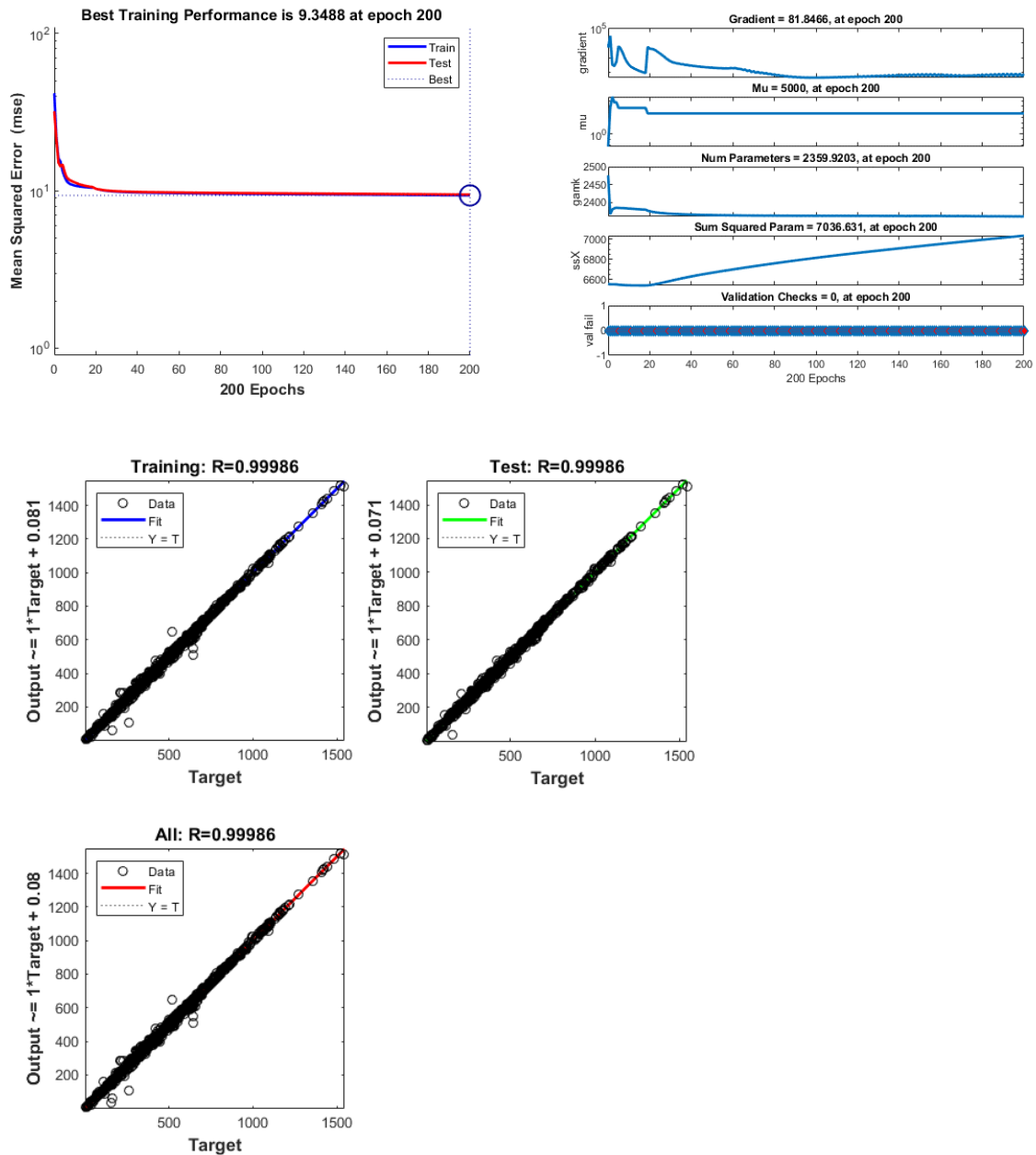
Με την χρήση ενός υποσυνόλου που είχε 250,000 δεδομένα και έτρεξε για 1000 εποχές έγινε αρκετά καλή εκτίμηση των συναπτικών βαρών, έτσι το T.N.Δ ξεκίνησε από σφάλμα κάτω από 5 και έφτασε στο 4.04 μετά από 200 εποχές εκπαίδευσης (Διάγραμμα 33).

Αλγόριθμος Bayesian RegularizationΔιάγραμμα 34: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 2^ο σενάριο

Αρχικά το T.N.Δ. εκπαιδεύτηκε για 1000 εποχές με την χρήση 250,000 δεδομένων κάτι που κατέβασε το MSE στο 31.89. Στην συνέχεια με την χρήση όλων των δεδομένων για 200 εποχές, έγινε επανεκπαίδευση του T.N.Δ. με αποτέλεσμα να κατέβει το MSE στο 27.24. Αυτή η δοκιμή ήταν και η μόνη που τα αποτελέσματα των δύο αλγορίθμων με αυτή την αρχιτεκτονική είχαν τόσο μεγάλη διαφορά (Διάγραμμα 34).

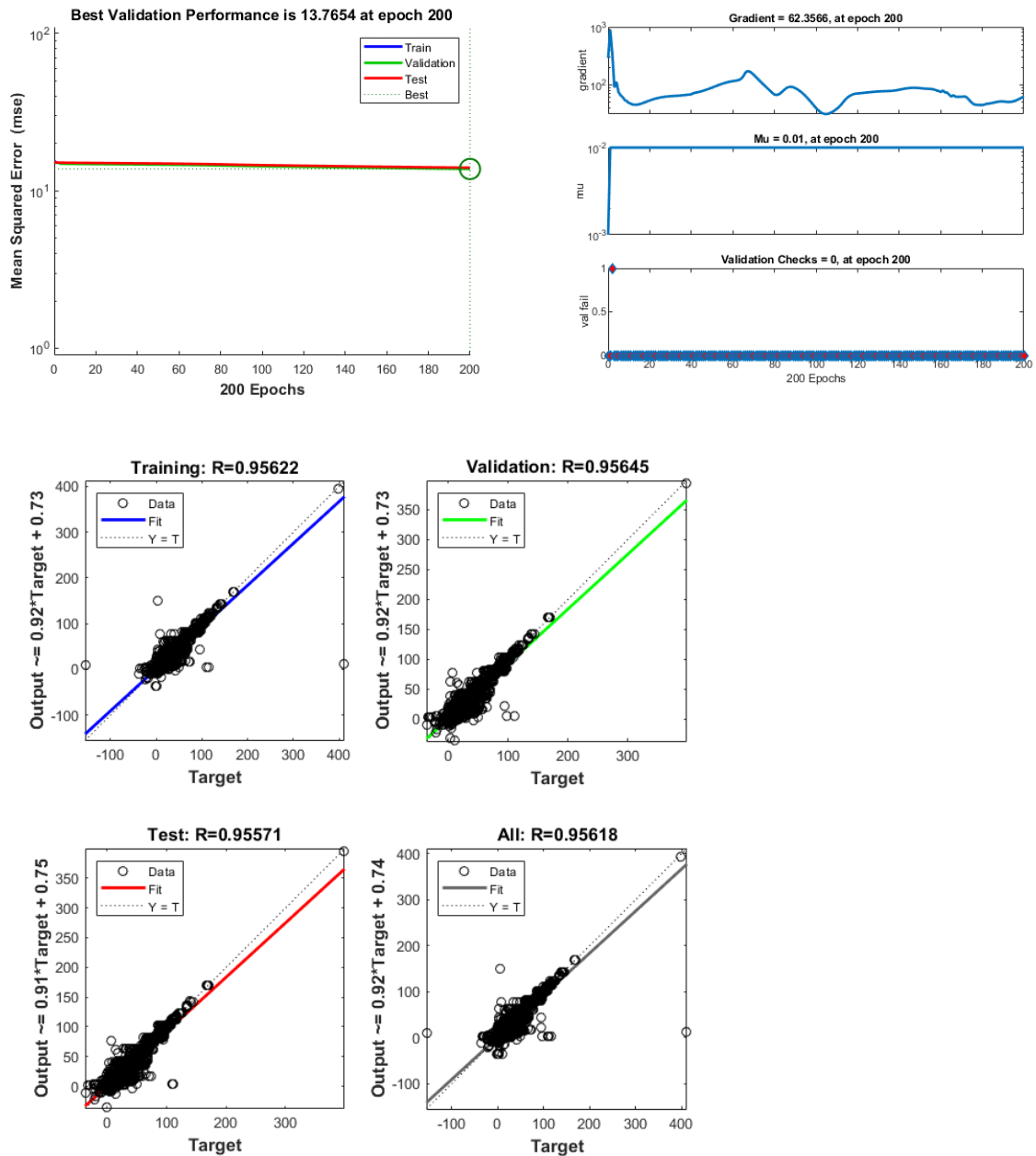
3^ο ΣΕΝΑΡΙΟΑλγόριθμος Levenberg-MarquardtΔιάγραμμα 35: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 3^ο σενάριο

Για την εκπαίδευση αυτού του T.N.Δ. πραγματοποιήθηκαν δύο προ-εκπαιδεύσεις και στην συνέχεια έγινε επανεκπαίδευση με όλα τα δεδομένα. Στην πρώτη προ-εκπαίδευση, χρησιμοποιήθηκαν 25,000 δεδομένα για 564 εποχές (ο αλγόριθμος είχε πρόωρο τερματισμό) μόνο κατεβάζοντας το *MSE* στο 25.33, στην συνέχεια έγινε επανεκπαίδευση δεκαπλασιάζοντας τα δεδομένα, αυτό είχε ως αποτέλεσμα την περαιτέρω μείωση του *MSE* στα 13.57. Έτσι με την χρησιμοποίηση όλων των δεδομένων, το *MSE* έφτασε οριακά κάτω από 12 (Διάγραμμα 35).

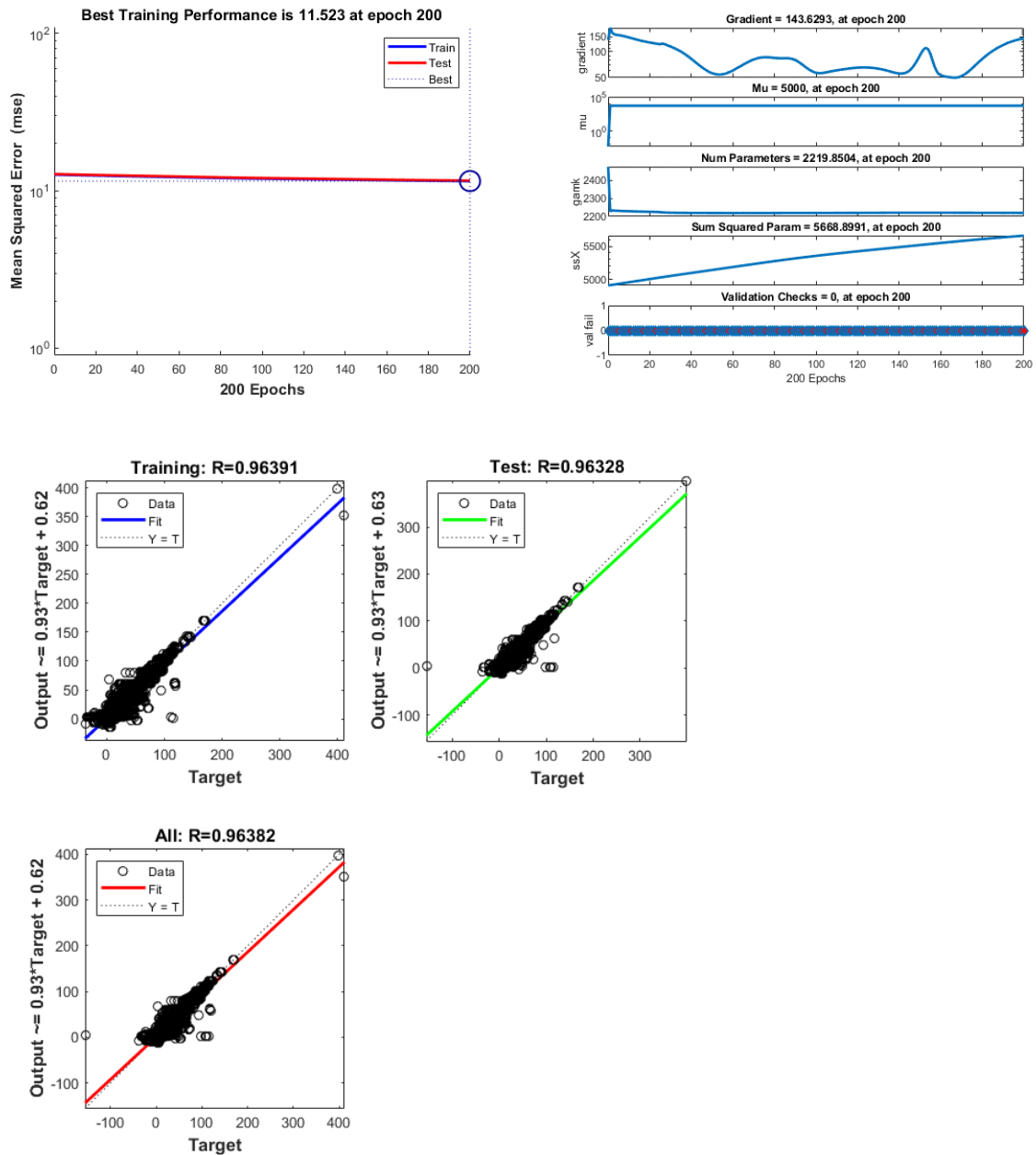
Αλγόριθμος Bayesian Regularization

Διάγραμμα 36: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 3^ο σενάριο

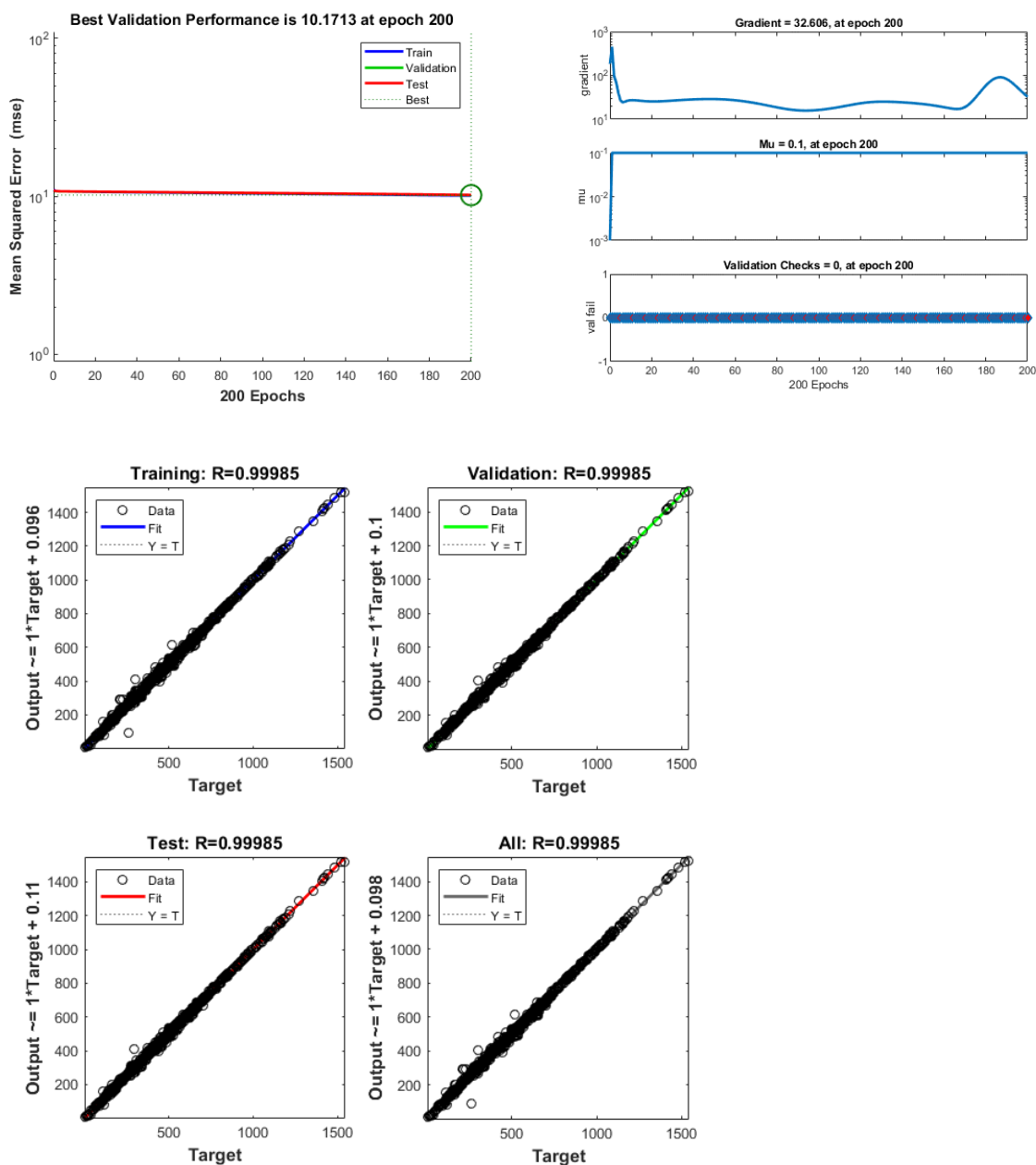
Σε αυτήν την περίπτωση έγινε μια προ-εκπαίδευση για 2000 εποχές με χρήση 25000 δεδομένων και στην συνέχεια χρησιμοποιήθηκε όλο το σύνολο των δεδομένων, με αυτόν τον τρόπο το *MSE* κατέβηκε στο 9.3 (Διάγραμμα 36).

4^ο ΣΕΝΑΡΙΟΑλγόριθμος Levenberg-MarquardtΔιάγραμμα 37: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 4^ο σενάριο

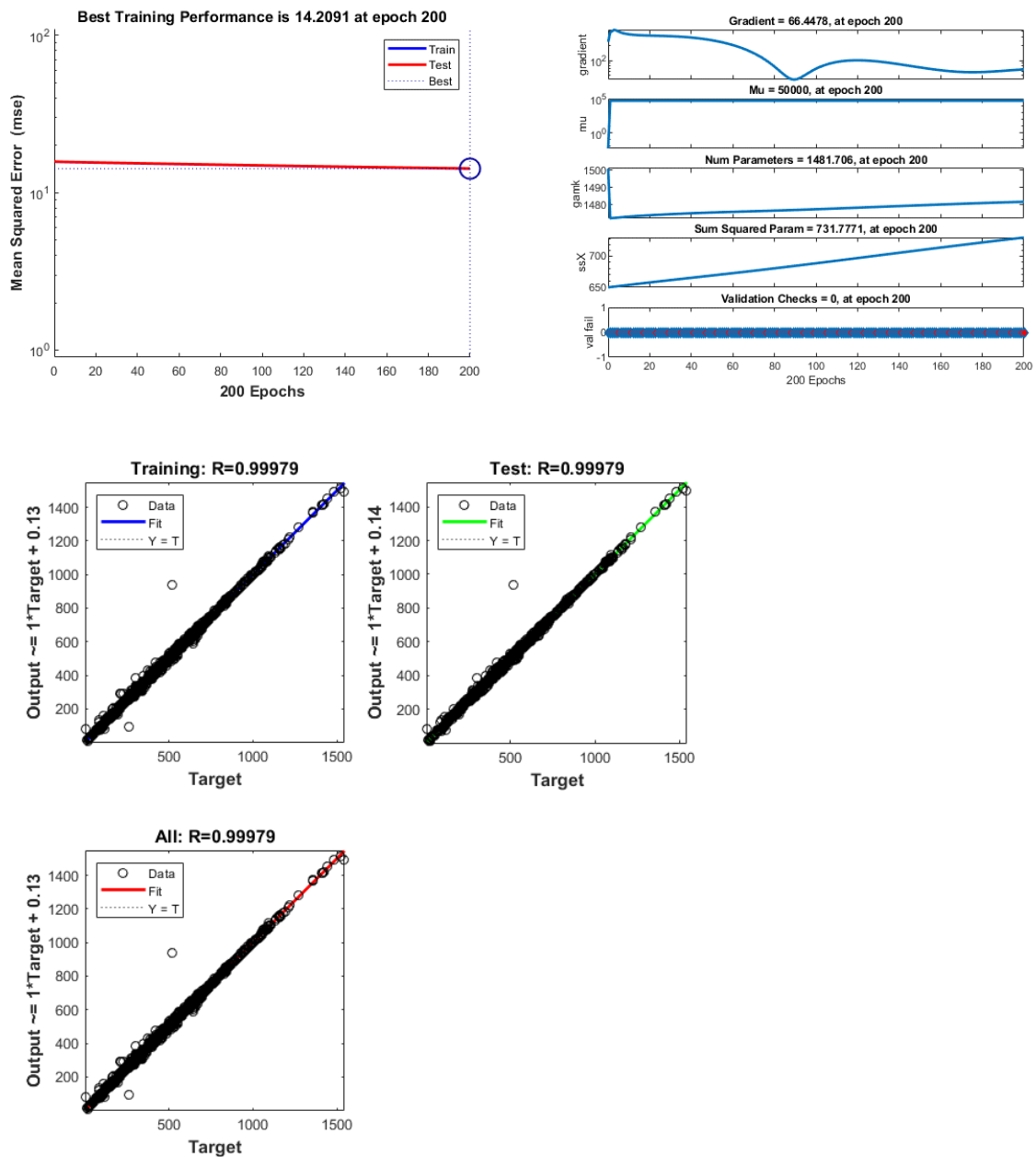
Με την προ-εκπαίδευση 25,000 δεδομένων για 1000 εποχές το *MSE* έπεσε στο 15.6. Στην συνέχεια με την χρήση όλου του σετ κατέβηκε στα 13.76 (Διάγραμμα 37).

Αλγόριθμος Bayesian RegularizationΔιάγραμμα 38: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 4^ο σενάριο

Με την προ-εκπαίδευση 25,000 δεδομένων για 2000 εποχές το *MSE* έπεσε στο 12.13. Στην συνέχεια με την χρήση όλου του σετ κατέβηκε στα 11.52 (Διάγραμμα 38).

5^ο ΣΕΝΑΡΙΟΑλγόριθμος Levenberg-MarquardtΔιάγραμμα 39: Αποτελέσματα αλγόριθμου Levenberg-Marquardt για το 5^ο σενάριο

Με την χρήση προ-εκπαίδευσης με 25,000 δεδομένα για 787 εποχές (η εκπαίδευση σταμάτησε λόγω πρόωρου τερματισμού) και στην συνέχεια με την χρήση του συνόλου των δεδομένων, η εκπαίδευση ολοκληρώθηκε μετά από 200 εποχές με MSE 10.17 (Διάγραμμα 39).

Αλγόριθμος Bayesian RegularizationΔιάγραμμα 40: Αποτελέσματα του αλγόριθμου Bayesian Regularization για το 5^ο σενάριο

Με χρήση ενός υποσυνόλου των δεδομένων έγινε μια προεκπαίδευση του δικτύου, κατεβάζοντας το *MSE* μετά από 1500 εποχές, ελάχιστα κάτω από το 15. Στην συνέχεια η εκπαίδευση με ολόκληρο το σετ δεδομένων, απλά επικύρωσε την προηγούμενη εκπαίδευση, αφού η διόρθωση του σφάλματος ήταν πολύ μικρή (Διάγραμμα 40).

ΕΛΕΓΧΟΣ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

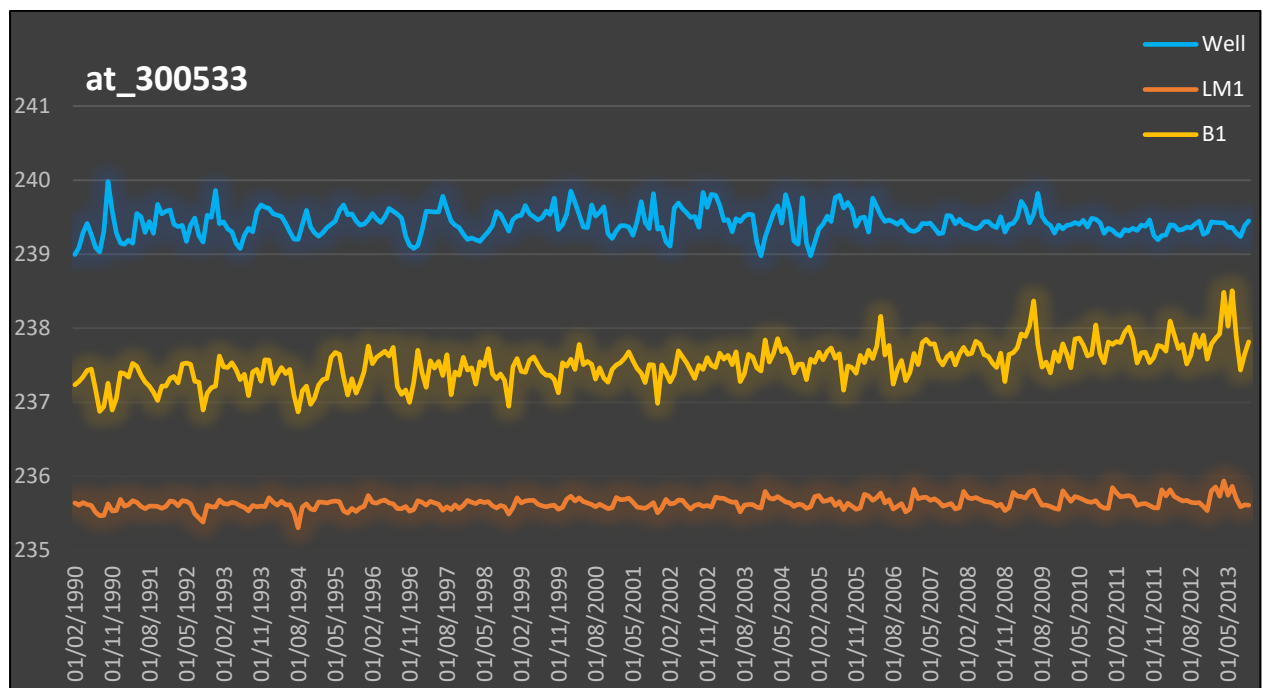
Προκειμένου να ελεγχθεί η αξιοπιστία των Τ.Ν.Δ. έγινε σύγκριση των αποτελεσμάτων που παρήγαγαν με τα δεδομένα που ήδη ήταν γνωστά. Ο έλεγχος αυτός πραγματοποιήθηκε σε δύο στάδια, αρχικά ελέγχθηκε η συμπεριφορά των αλγορίθμων σε πηγάδια που είχαν συμπεριληφθεί στο σύνολο της εκπαίδευσης (αν και δεν ήταν δεδομένο ότι όλες οι τιμές τους θα είχαν χρησιμοποιηθεί για εκπαίδευση καθώς κάποιες από αυτές θα μπορούσαν να είχαν συμπεριληφθεί είτε στο υποσύνολο ελέγχου και των δύο αλγορίθμων, είτε στο υποσύνολο επικύρωσης του αλγόριθμου *Levenberg-Marquardt*). Στο δεύτερο στάδιο, έγινε έλεγχος των πηγαδιών στο πως μπορούν να προβλέψουν την κατάσταση σε σημεία που δεν έχουν δει καθόλου στην εκπαίδευση τους, όπως έγινε με τα πηγάδια που βρίσκονται στην Ουγγαρία.

ΑΠΟΔΟΣΗ ΑΛΓΟΡΙΘΜΩΝ ΣΕ ΠΗΓΑΔΙΑ ΠΟΥ ΣΥΜΜΕΤΕΙΧΑΝ ΣΤΗΝ ΕΚΠΑΙΔΕΥΣΗ ΤΩΝ Τ.Ν.Δ.

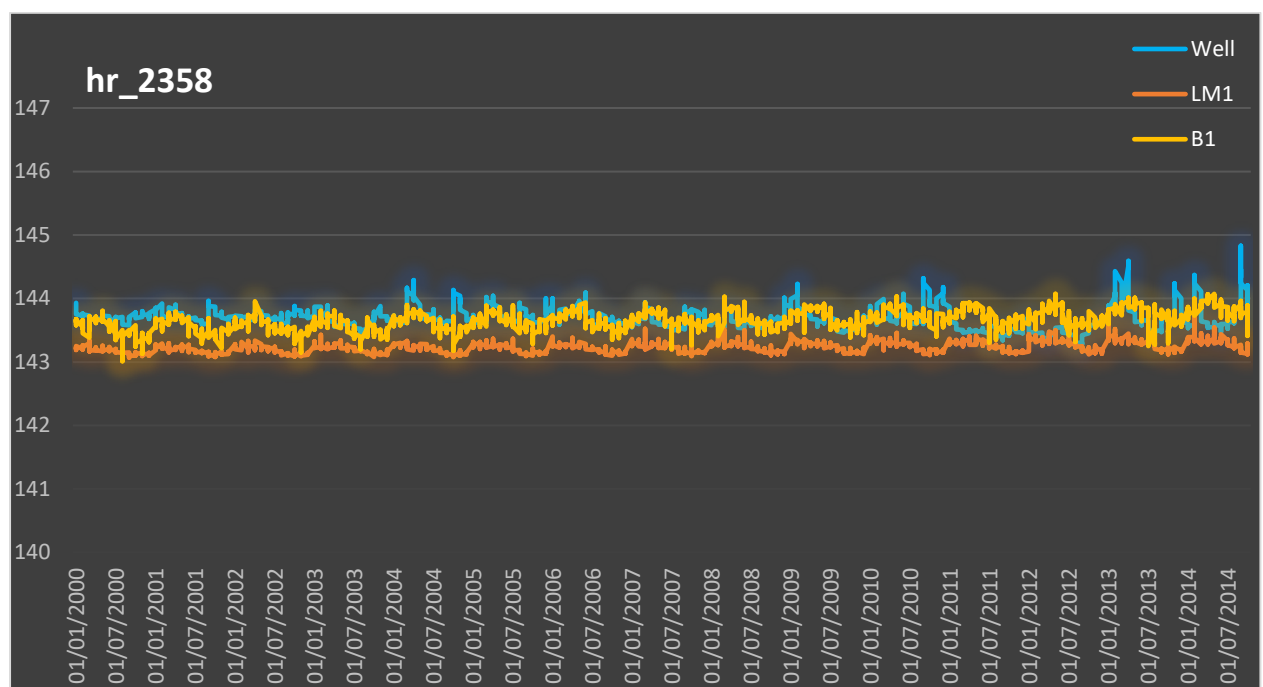
Λόγω των πολλών χωρών που υπάρχουν στην περιοχή μελέτης, θα ήταν υπερβολικό να εξεταστεί ξεχωριστά η απόδοση των αλγορίθμων σε κάθε μια από αυτές. Ειδικά από την στιγμή που το σημαντικότερο κριτήριο για την αξιοπιστία των αλγορίθμων, θα είναι η απόδοση τους στην περιοχή της Ουγγαρίας. Ως εκ τούτου, η απόδοση των αλγορίθμων θα εξεταστεί σε δύο διαφορετικά πηγάδια ένα ανάντη της Ουγγαρίας και ένα κατάντι αυτής. Τα πηγάδια που επιλέχθηκαν είναι ένα στην Αυστρία (*at_300533*) και ένα στην Κροατία (*hr_2358*) και τα αποτελέσματα των αλγορίθμων εμφανίζονται ανά σενάριο στα επόμενα δέκα γραφήματα (*Διάγραμμα 41* έως *Διάγραμμα 50*). Τα γραφήματα με περιττό αριθμό αφορούν τον αλγόριθμο *Levenberg-Marquardt*, ενώ αυτά με τον ζυγό, τον αλγόριθμο *Bayesian Regularization*.

Ως γενικότερο συμπέρασμα από αυτά τα γραφήματα προκύπτει ότι ναι μεν κατάφεραν να δώσουν μια γενικότερη πρόβλεψη για την στάθμη του υδροφόρου, αλλά απέτυχαν να αποτυπώσουν τις διακυμάνσεις. Οι προσπάθειες που έκαναν φαίνεται ότι σχετίζονται περισσότερο με το χρόνο, για αυτό και παρατηρούνται κάποια επαναλαμβανόμενα μοτίβα αυξομειώσης. Παρά με τις περιβαλλοντικές συνθήκες. Γενικότερα οι αλγόριθμοι φαίνεται ότι προσομοίωσαν καλύτερα το πηγάδι της Κροατίας, από αυτό της Αυστρίας. Αυτό ενδεχομένως να οφείλεται στην μικρότερη απόσταση που έχει ο υδροφόρος ορίζοντας με την επιφάνεια του εδάφους, στο πηγάδι της Κροατίας σε σχέση με αυτό της Αυστρίας. Επομένως οι αλγόριθμοι που λαμβάνουν το υψόμετρο ως μεταβλητή εισόδου (σενάρια 1, 2 και 5), να προσαρμόστηκαν πάνω σε αυτό. Την στιγμή που και οι αλγόριθμοι που προσπαθούσαν να προσδιορίσουν την συγκεκριμένη απόσταση (σενάρια 2 και 4) να ευνοήθηκαν από αυτό.

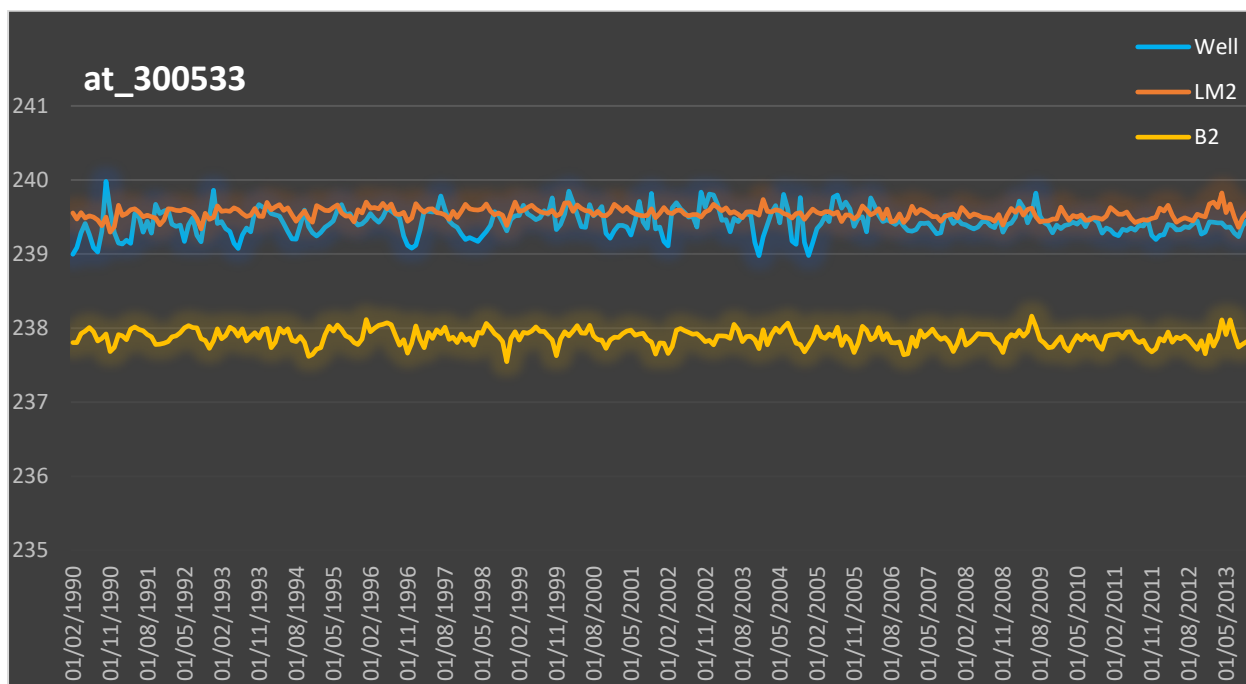
Τις μεγαλύτερες αποκλείσεις στον σταθμό της Αυστρίας, παρουσίασε ο αλγόριθμος *Levenberg-Marquardt* στο 1^ο σενάριο, ενώ τις μικρότερες αποκλείσεις φαίνεται να τις έδωσε ο ίδιος αλγόριθμος στο 3^ο σενάριο. Στον σταθμό της Κροατίας οι μεγαλύτερες αποκλείσεις προέκυψαν από τους αλγόριθμους του 5^{ου} σεναρίου, ενώ τις λιγότερες αποκλείσεις φαίνεται να τις έδωσε ο αλγόριθμος *Bayesian Regularization* για το 1^ο σενάριο.

1^ο Σενάριο

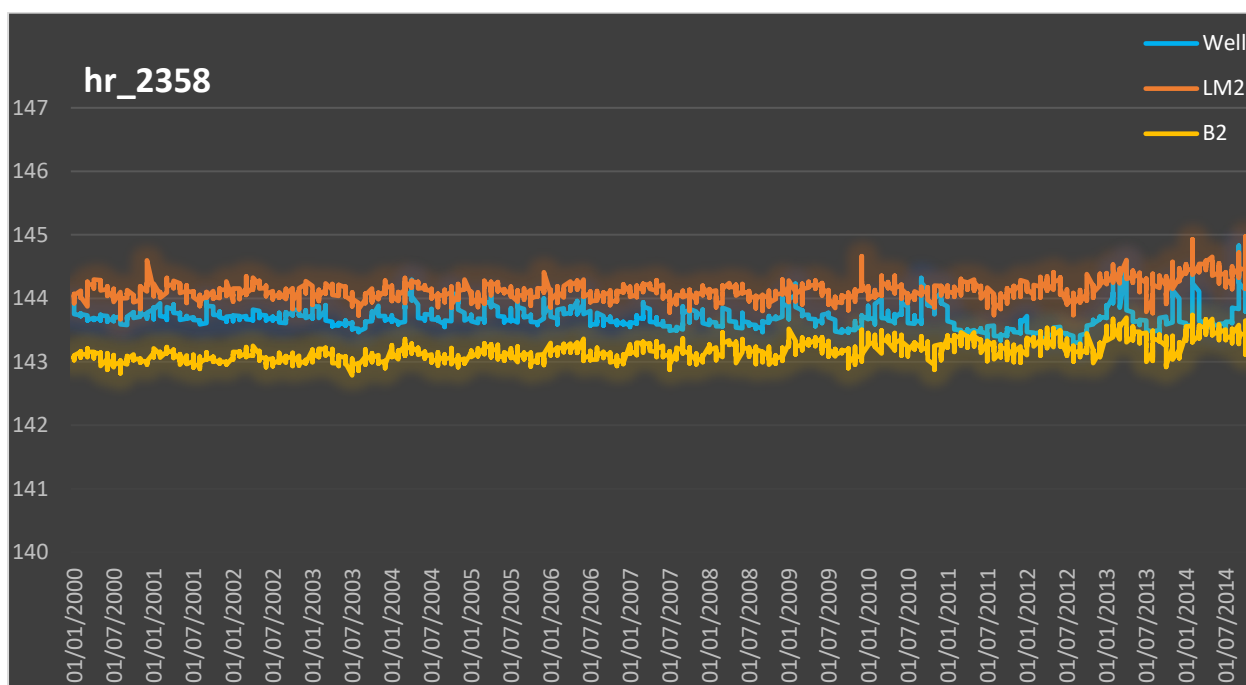
Διάγραμμα 41: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 1^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.



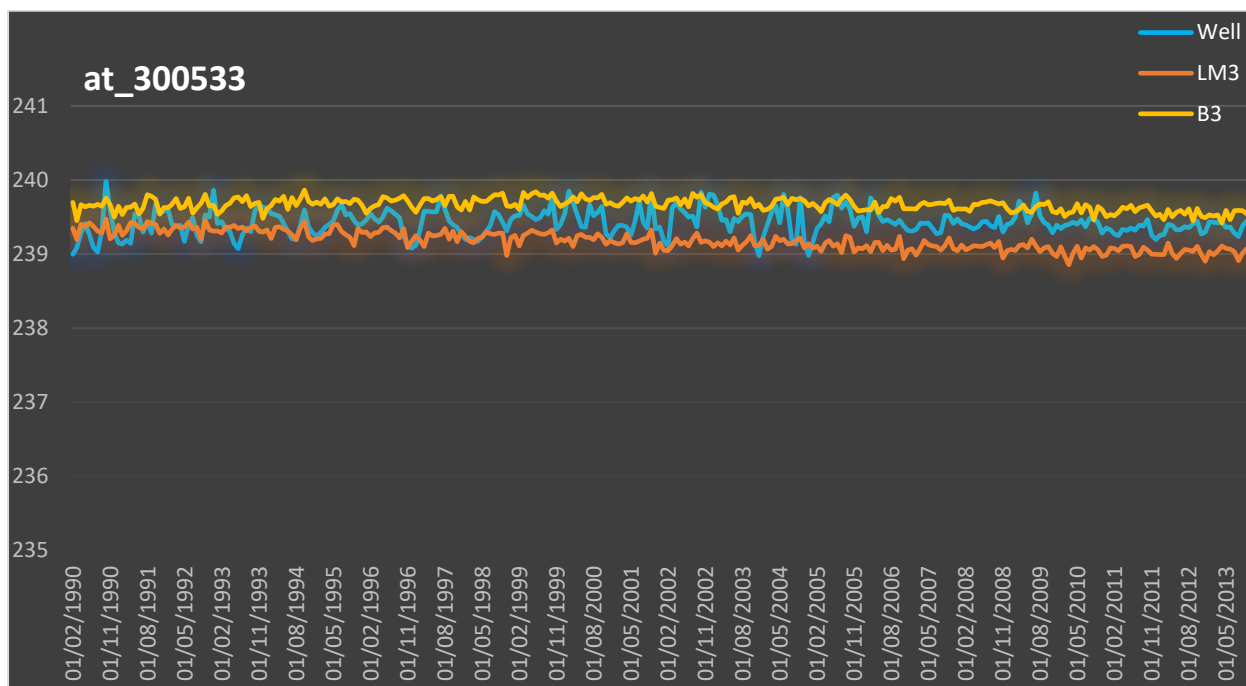
Διάγραμμα 42: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 1^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.

2^ο Σενάριο

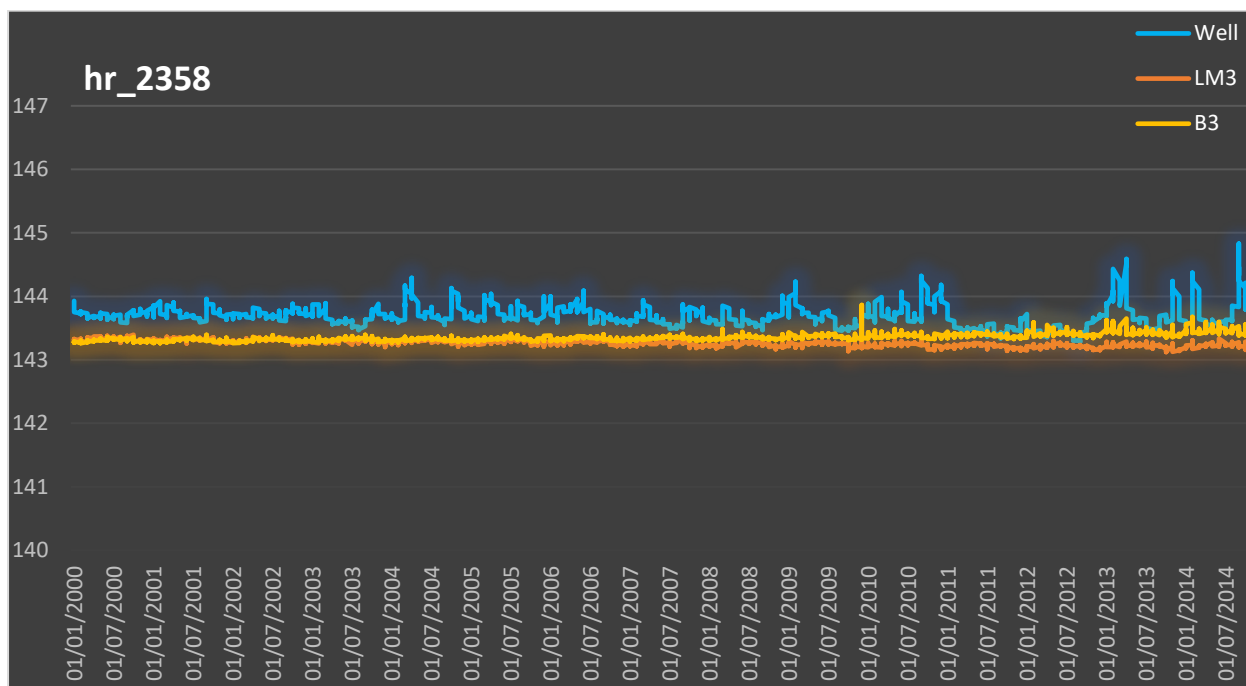
Διάγραμμα 43: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 2^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.



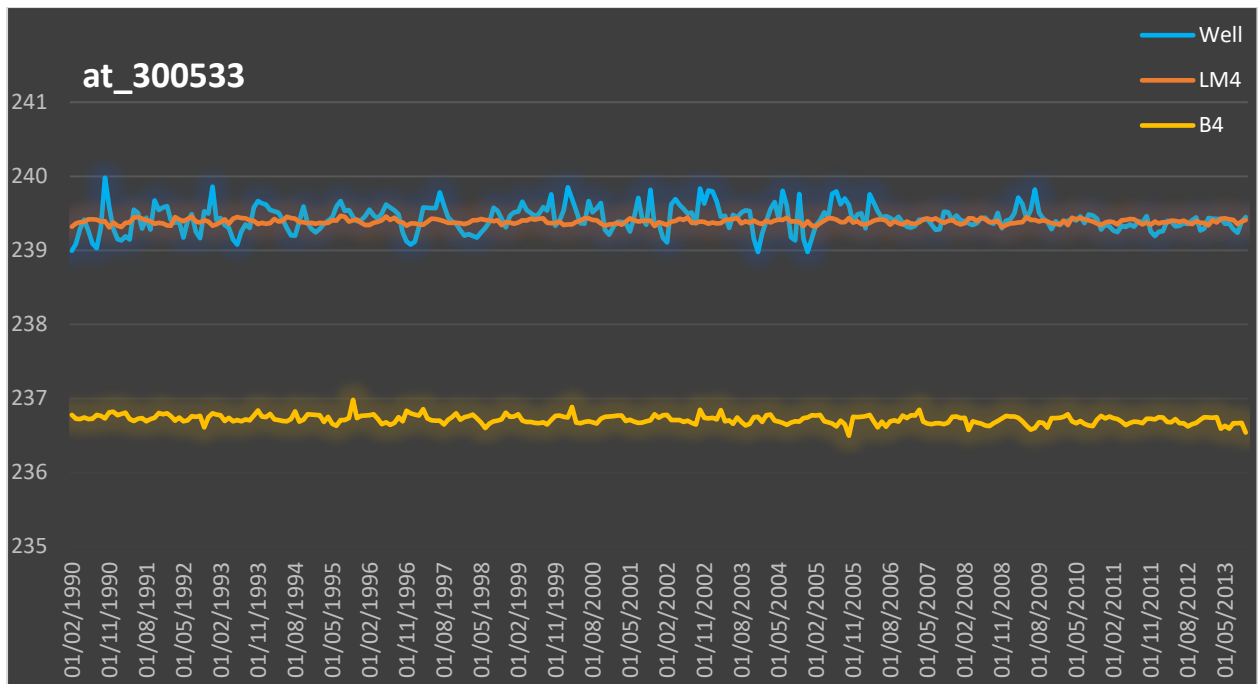
Διάγραμμα 44: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 2^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.

3^ο Σενάριο

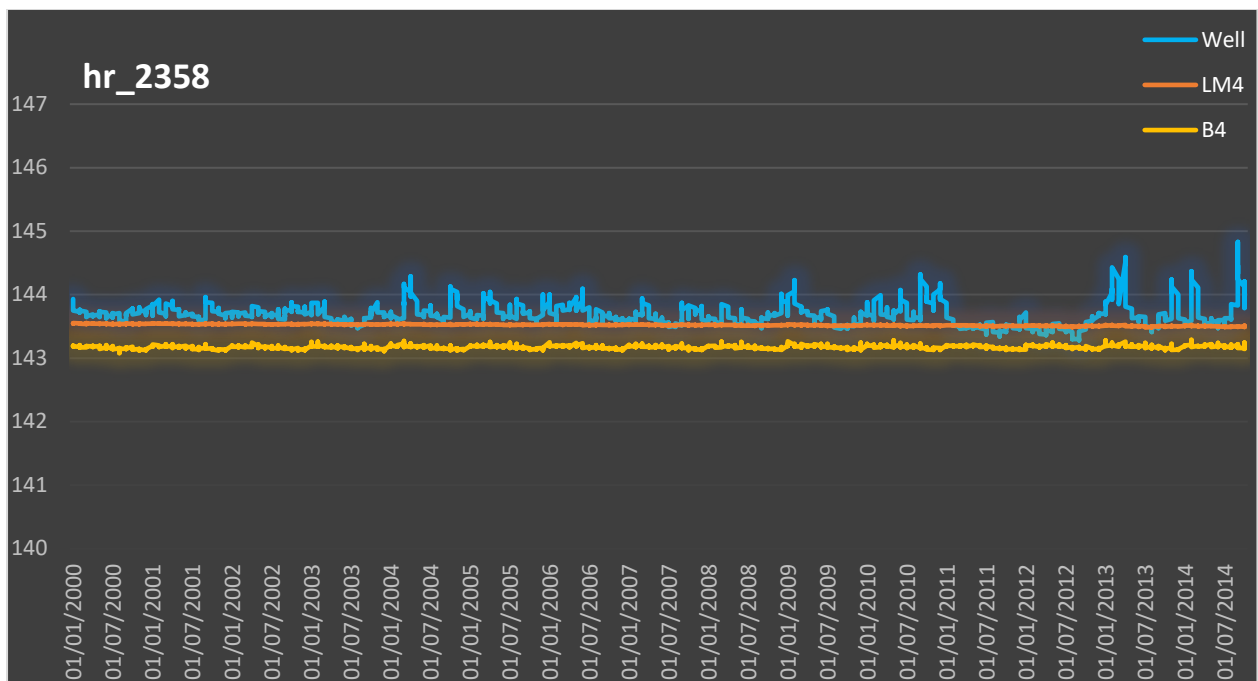
Διάγραμμα 45: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 3^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.



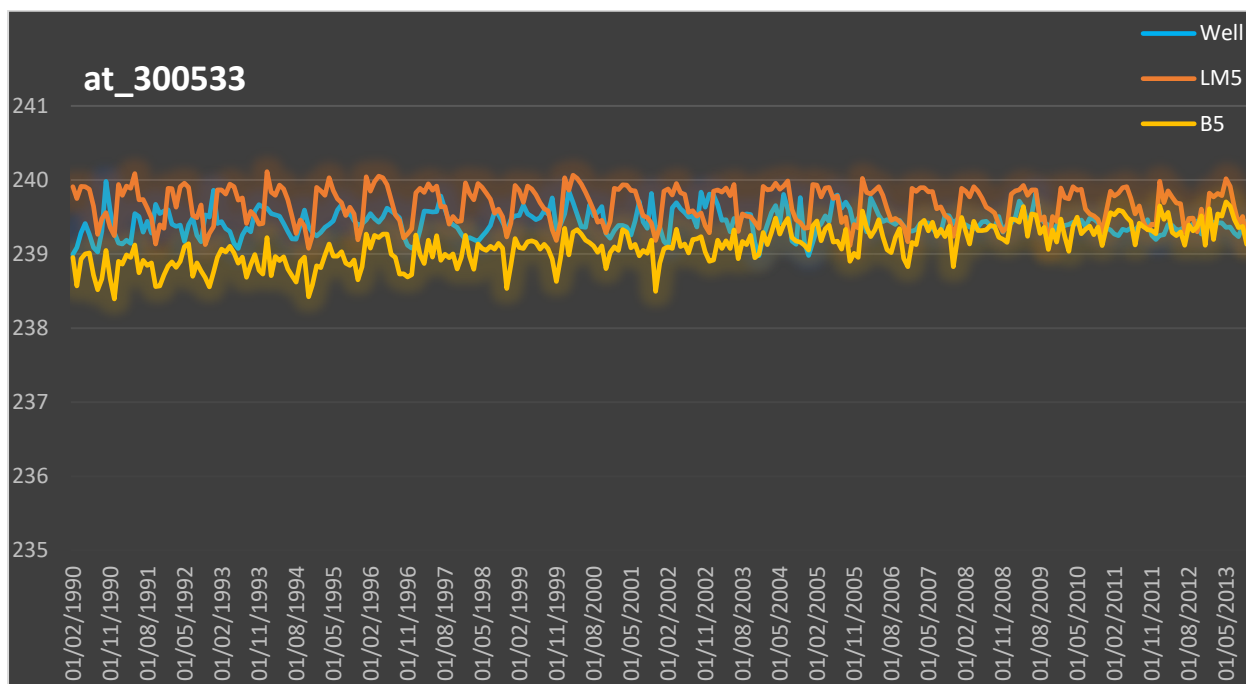
Διάγραμμα 46: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 3^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.

4^ο Σενάριο

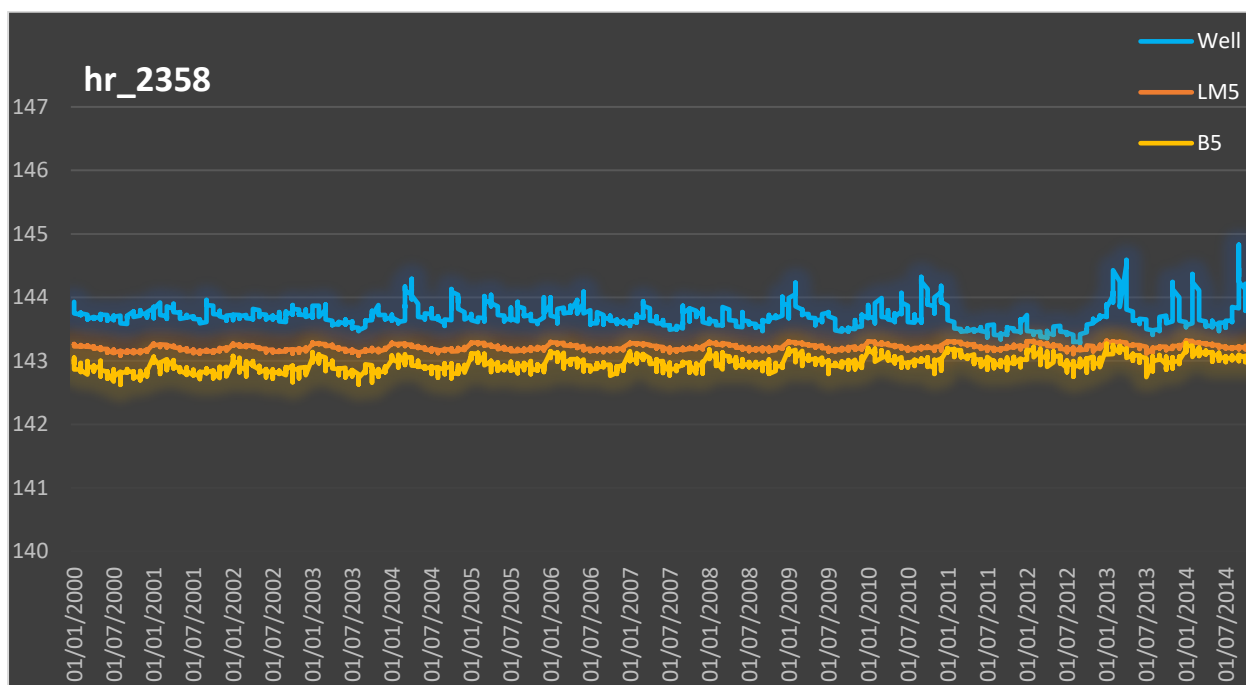
Διάγραμμα 47: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 4^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.



Διάγραμμα 48: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 4^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.

5^ο Σενάριο

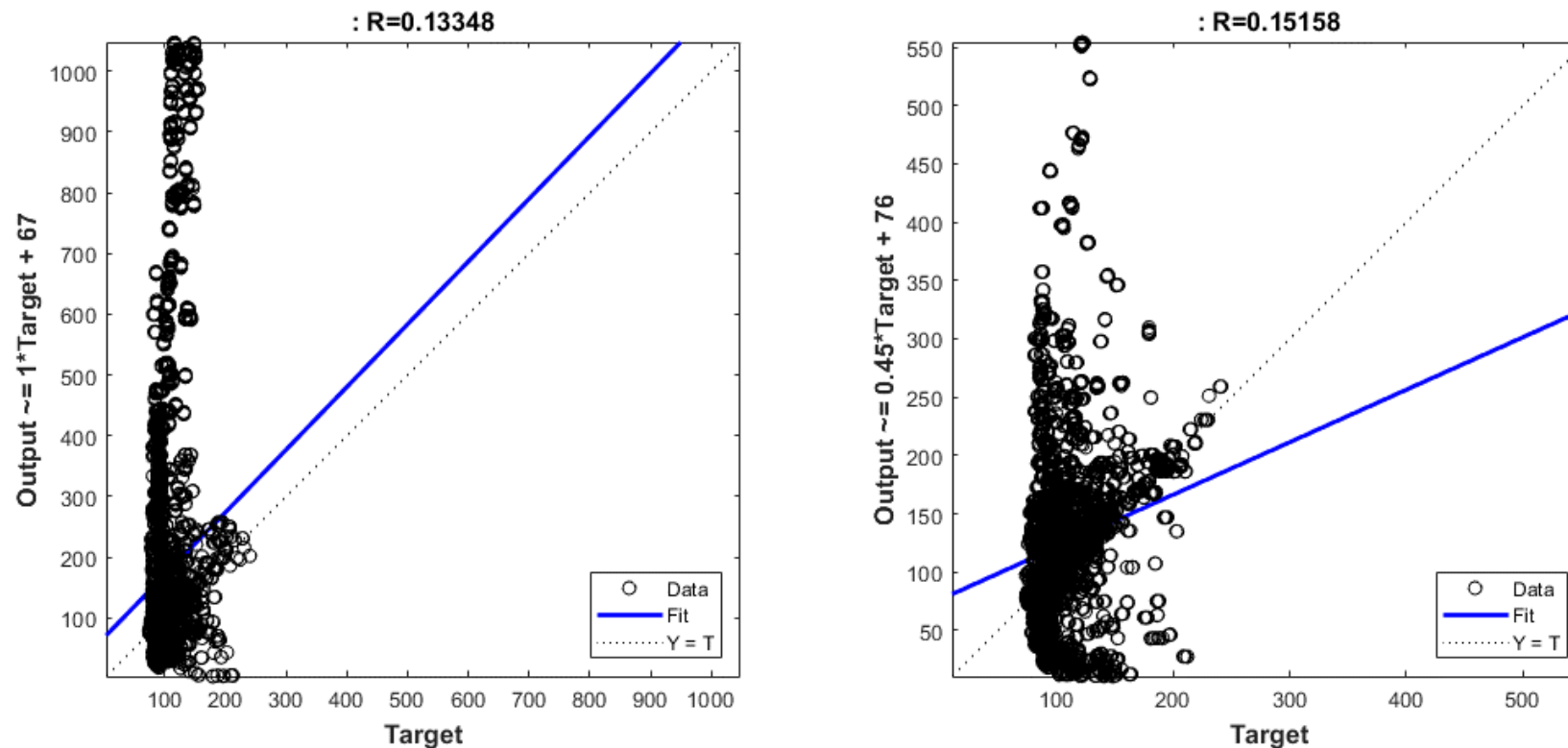
Διάγραμμα 49: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 5^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Αυστρίας.



Διάγραμμα 50: Αντιπαραβολή των αποτελεσμάτων των νευρωνικού δικτύων που προέκυψαν από το 5^ο σενάριο με τις πραγματικές τιμές ενός τυχαίου πηγαδιού της Κροατίας.

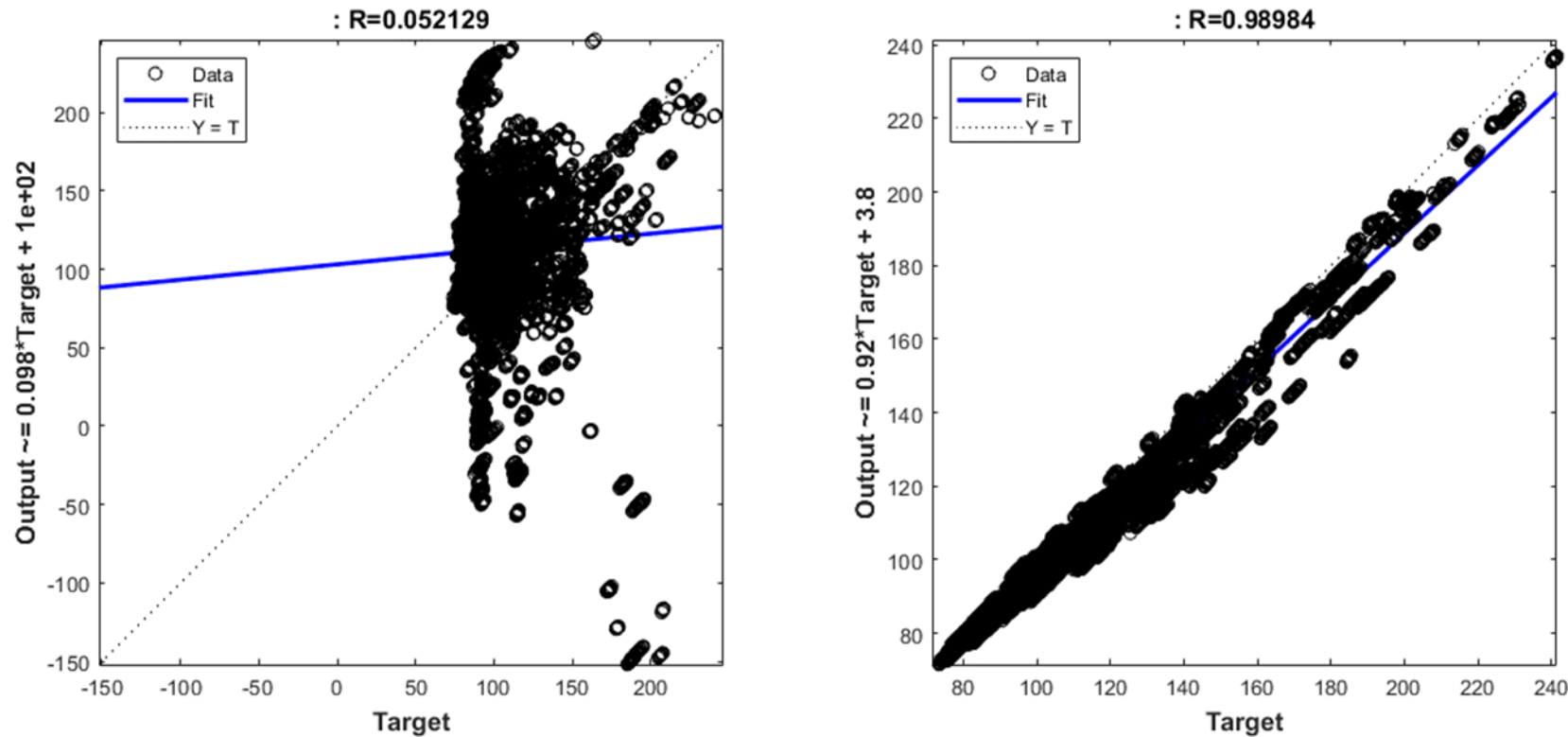
ΑΠΟΔΟΣΗ ΑΛΓΟΡΙΘΜΩΝ ΣΤΟΥΣ ΣΤΑΘΜΟΥΣ ΤΗΣ ΟΥΓΓΑΡΙΑΣ

1ο Σενάριο



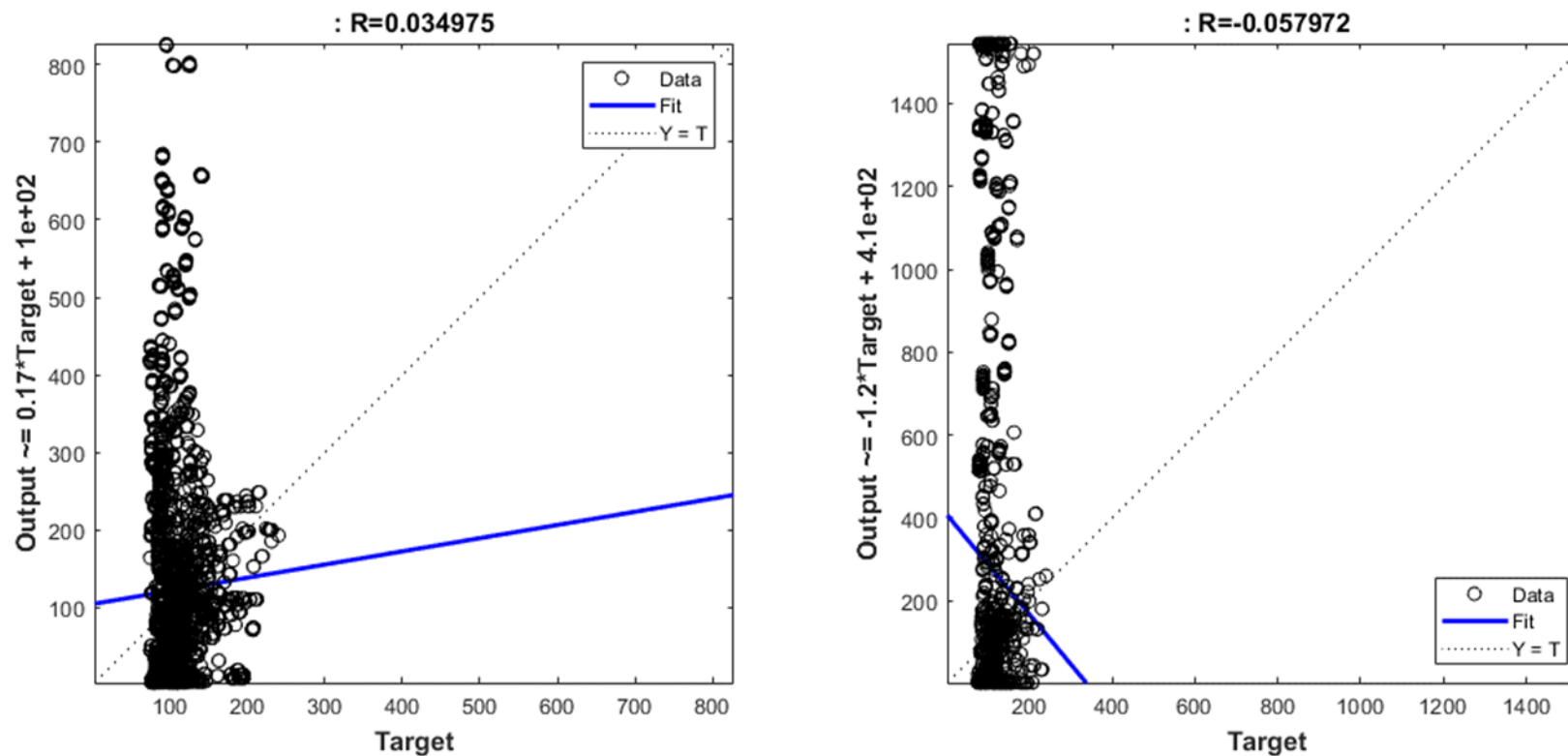
Διάγραμμα 51: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 1^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Όπως φαίνεται από τα παραπάνω διαγράμματα, η απόδοση των αλγορίθμων σε πηγάδια που δεν είχαν δει, δείχνει ότι τα δύο δίκτυα είχαν υπερπροσαρμοστεί στις τιμές των πηγαδιών που τους δόθηκαν για εκπαίδευση, με αποτέλεσμα να μην μπορούν να δώσουν ικανοποιητικές αποδόσεις στην υπόλοιπη περιοχή. Ενδεικτικά το *MSE* του αλγόριθμου *Levenberg-Marquardt* ήταν περίπου 6000 ενώ του αλγόριθμου *Bayesian Regularization* ξεπερνούσε τις 40,000 (Διάγραμμα 51).

2^ο Σενάριο

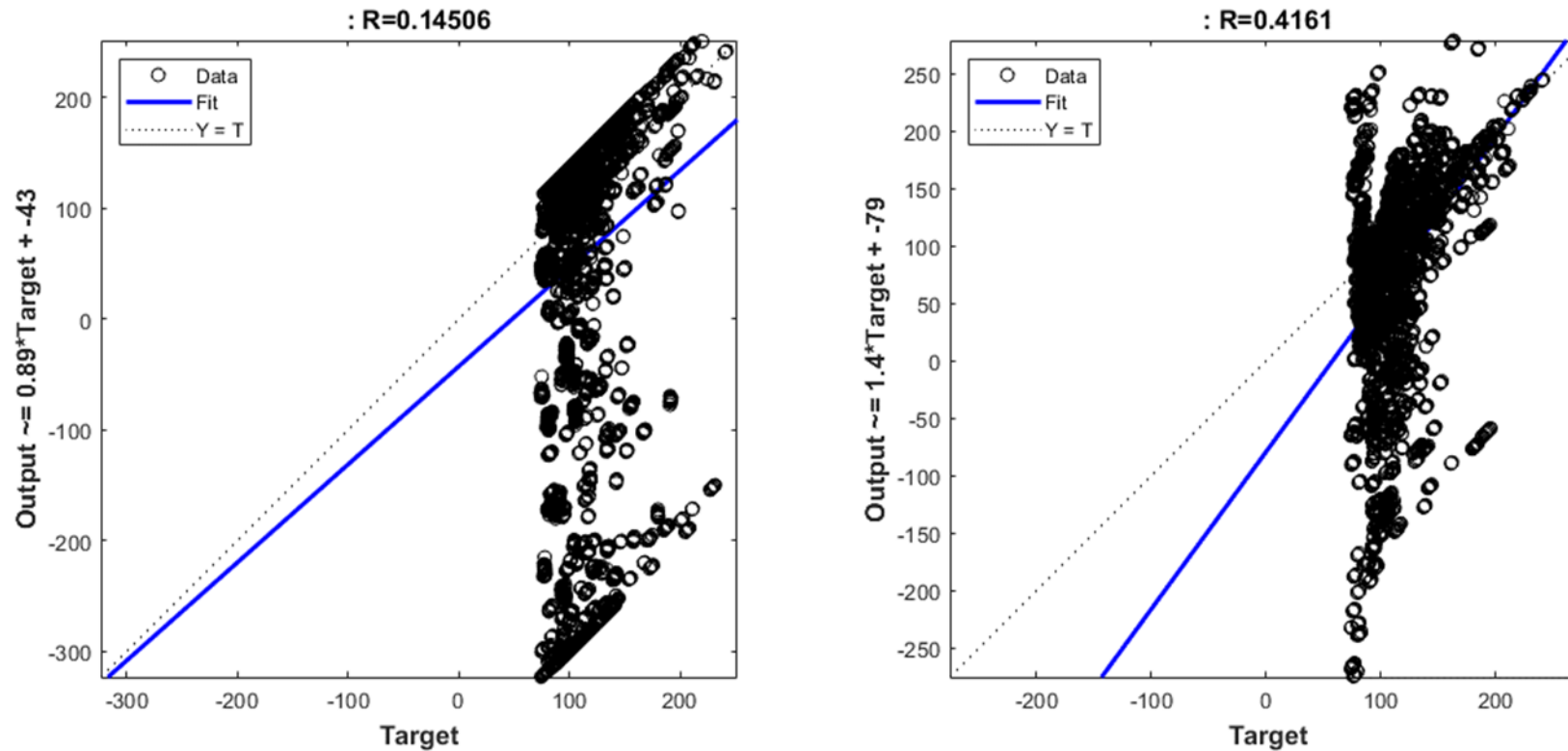
Διάγραμμα 52: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 2^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Το Τ.Ν.Δ. που εκπαιδεύτηκε με τον αλγόριθμο *Levenberg-Marquardt* απέτυχε να δώσει ικανοποιητικές λύσεις για την Ουγγαρία, έχοντας *MSE* 2700. Αντίθετα, το Τ.Ν.Δ. που εκπαιδεύτηκε από τον αλγόριθμο της *Bayesian Regularization*, είχε σχεδόν άψογη συμπεριφορά και στις τιμές που δεν είχε δει, έτσι το *MSE* που είχε στην εκπαίδευση (που ήταν το μεγαλύτερο *MSE* εκπαίδευσης από όλα τα νευρωνικά που εξετάστηκαν) ανέβηκε περίπου 6 μονάδες και από 27.4 πήγε στο 33.72 (Διάγραμμα 52).

3^ο Σενάριο

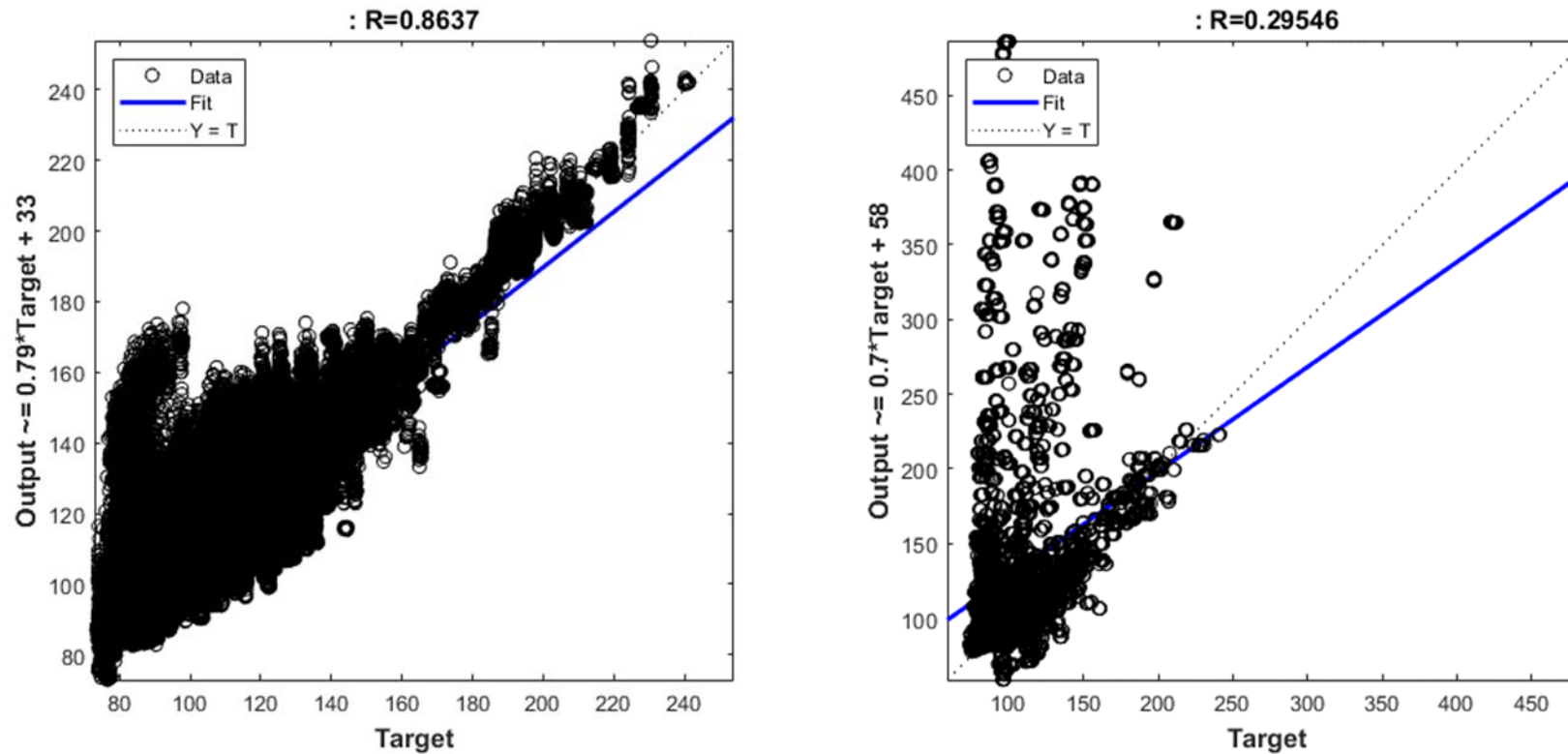
Διάγραμμα 53: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 3^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Η απόδοση και των δύο Τ.Ν.Δ. σε αυτή την δοκιμασία ήταν απογοητευτική. Κατέγραψαν 15,000 *MSE* και σχεδόν 300,000 *MSE* αντίστοιχα (Διάγραμμα 53).

4^ο Σενάριο

Διάγραμμα 54: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 4^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Οι αλγόριθμοι και σε αυτό το σενάριο απέτυχαν να πραγματοποιήσουν γενίκευση, με αποτέλεσμα να έχουν MSE πάνω από 25,000 και πάνω από 7,000 αντίστοιχα (Διάγραμμα 54).

5^ο Σενάριο

Διάγραμμα 55: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 5^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Σε αυτό το σενάριο ο αλγόριθμος Levenberg-Marquardt είχε μια μέτρια συμπεριφορά, καταφέροντας να έχει ένα κάπως καλό συντελεστή συσχέτισης R, αυτό αποτυπώνεται και στο MSE του, που ήταν λίγο κάτω από το 250 (Διάγραμμα 55).

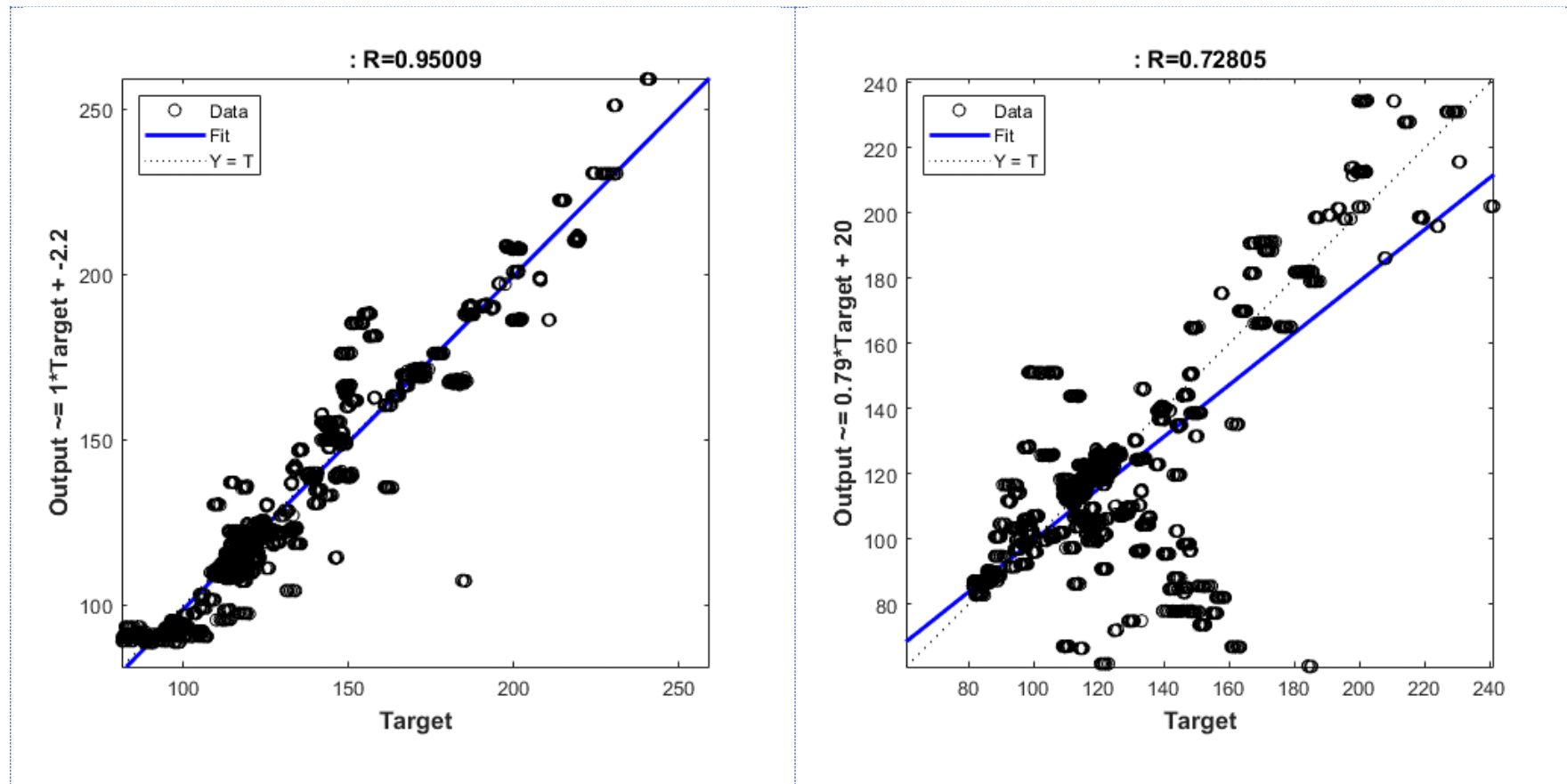
ΑΠΟΔΟΣΗ ΑΛΓΟΡΙΘΜΩΝ ΣΤΟΥΣ ΣΥΝΟΡΙΑΚΟΥΣ ΣΤΑΘΜΟΥΣ ΤΗΣ ΟΥΓΓΑΡΙΑΣ

Οι παραπάνω έλεγχοι επιβεβαίωσαν ότι η απόδοση ενός Τ.Ν.Δ. δεν θα πρέπει να κρίνεται μόνο από τα στοιχεία της εκπαίδευσης, ειδικά σε προβλήματα σαν και αυτό, στο οποίο αρκετά ζεύγη δεδομένων αφορούν το ίδιο πηγάδι με αποτέλεσμα ακόμα και αν «κρυφτούν» κάποιες τιμές από τον αλγόριθμο, αυτός να μπορεί να υπερπροσαρμοστεί σε αυτές, αφού κάποιες πολύ κοντινές ή ταυτόσημες τιμές με αυτές που είναι στο υποσύνολο επικύρωσης ή σε αυτό του υποσυνόλου ελέγχου, χρησιμοποιούνται για την εκπαίδευση των Τ.Ν.Δ.

Στην συνέχεια, προκείμενου να επιβεβαιωθεί ότι όντως οι αλγόριθμοι πραγματοποίησαν όντως υπερεκπαίδευση. Απομονώθηκαν ορισμένα πηγάδια στα σύνορα της Ουγγαρίας με Κροατία, Αυστρία και Σερβία. Γιατί ενδεχομένως να ήταν παράλογη η απαίτηση ο αλγόριθμος να αποδίδει σωστά σε όλη την Ουγγαρία, όταν σε μια τόσο μεγάλη περιοχή δεν είχε ούτε ένα σετ δεδομένων. Όμως σε μια περιοχή που είναι κοντά σε αρκετά πηγάδια που χρησιμοποιήθηκαν για την εκπαίδευση του δικτύου, αν δεν έχουν υπερεκπαίδευση τα πηγάδια, θα πρέπει να έχουν ικανοποιητική απόδοση. Για το λόγο αυτό απομονώθηκαν τα πηγάδια που εμφανίζονται στην παρακάτω εικόνα (Εικόνα 44).

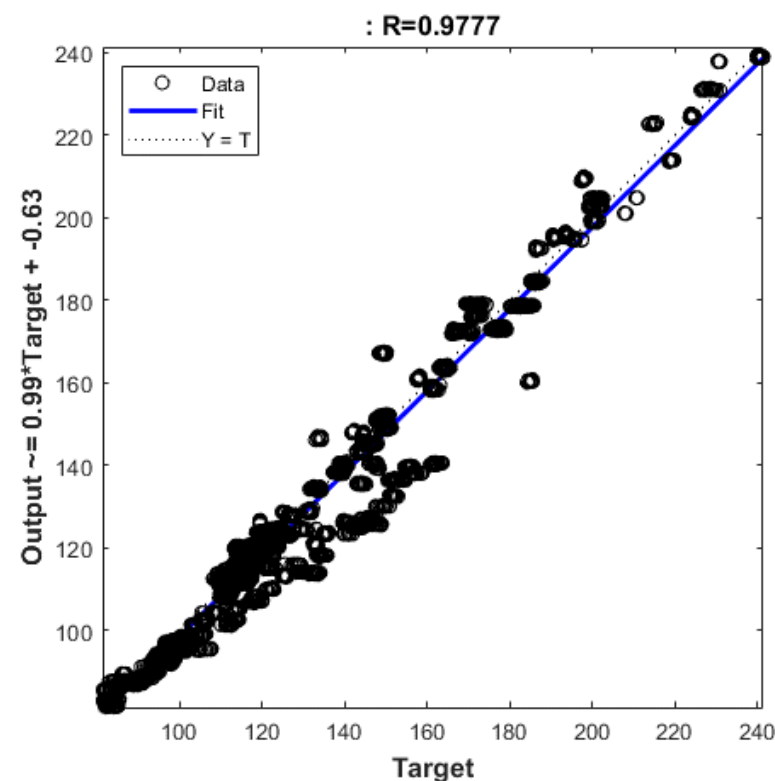
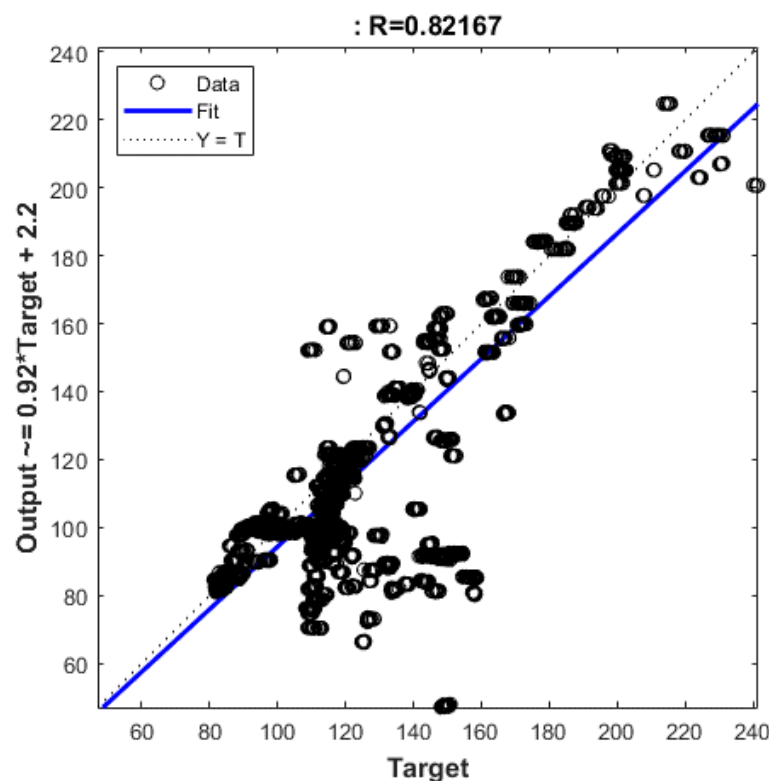


Εικόνα 44: Τα πηγάδια που χρησιμοποιήθηκαν για τον έλεγχο της απόδοσης των ΤΝΔ σε περιοχή των συνόρων της Ουγγαρίας

1^ο Σενάριο

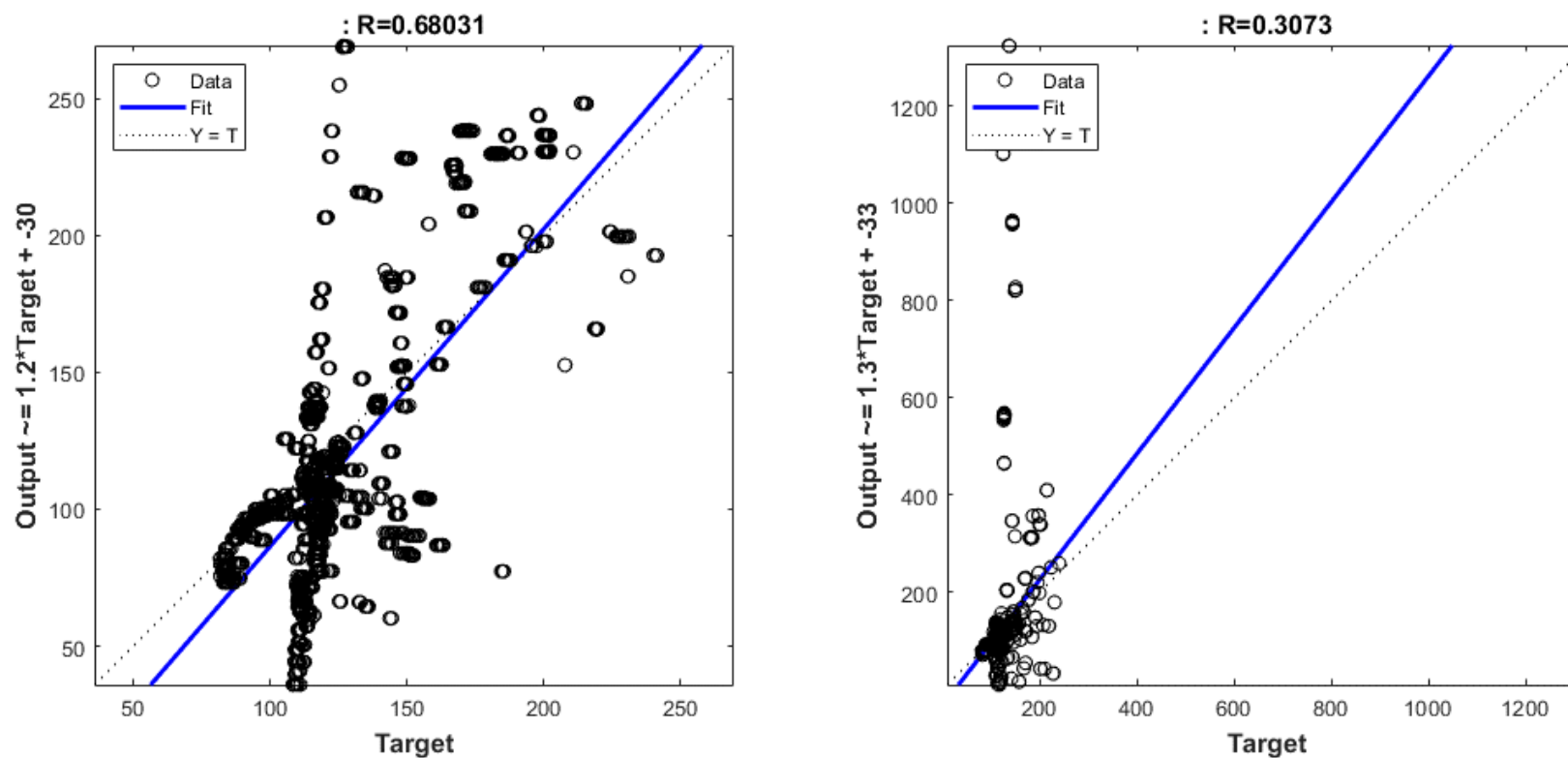
Διάγραμμα 56: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 1^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Πολύ καλύτερη εικόνα παρουσίασαν οι αλγόριθμοι στον έλεγχο που έγινε για τα πηγάδια που αφορούν μόνο την παραμεθόριο της Ουγγαρίας. Ειδικά ο αλγόριθμος Levenberg-Marquardt παρουσίασε συντελεστή συσχέτισης 0.95 και MSE 87. Βελτίωση είχε και ο αλγόριθμος Bayesian Regularization αλλά όχι σε βαθμό που να δίνει ικανοποιητικές προβλέψεις (MSE 504). Και οι δύο αλγόριθμοι παρουσίασαν μείωση του MSE , πάνω από 98% (Διάγραμμα 56).

2^ο Σενάριο

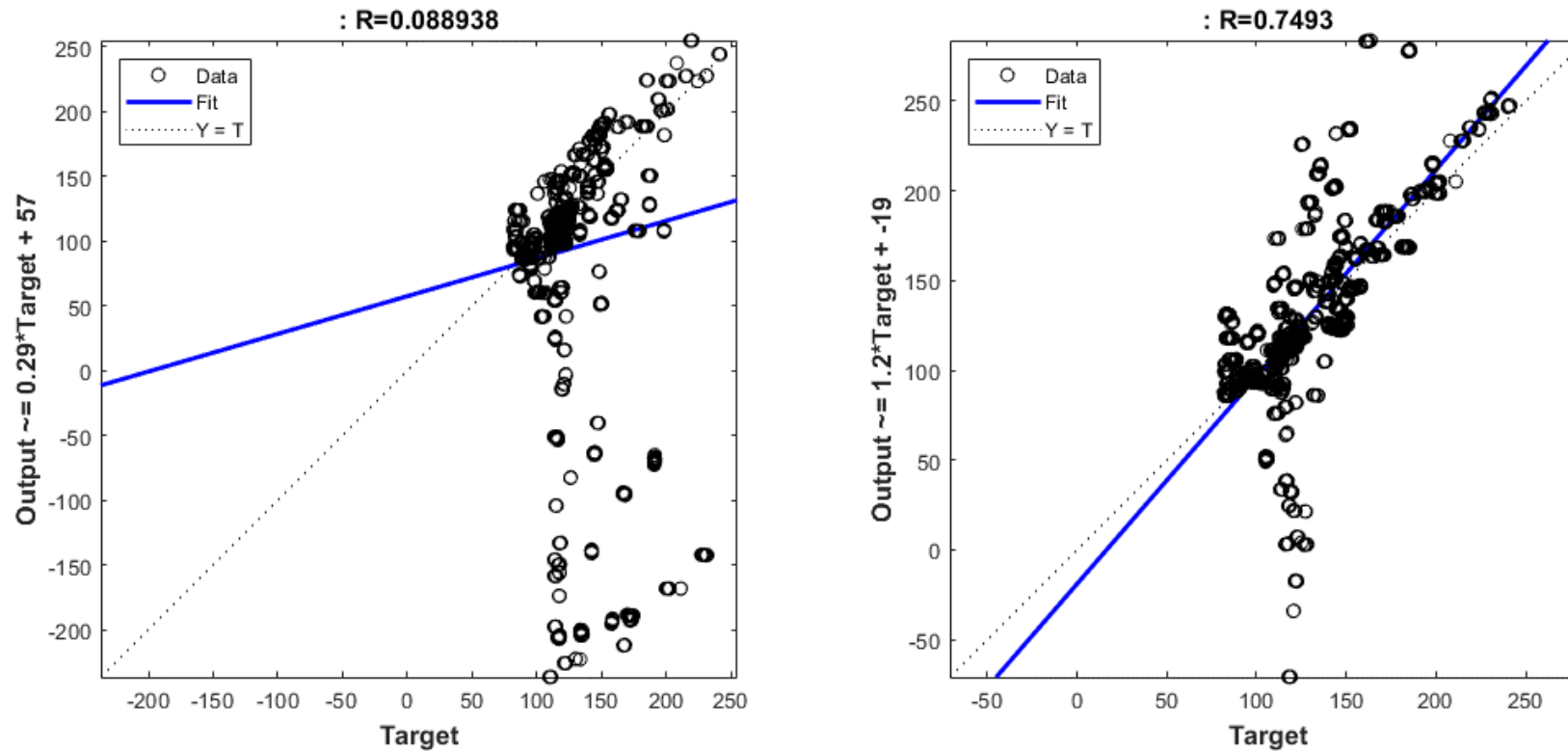
Διάγραμμα 57: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 2^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Βελτίωση παρουσίασε και σε αυτό το σενάριο ο αλγόριθμος *Levenberg-Marquardt* μειώνοντας το *MSE* κατά 85% (από 2718 σε 384), αντιθέτως ο αλγόριθμος *Bayesian Regularization* δεν παρουσίασε κάποια βελτίωση, αλλά οριακά μειώθηκε η απόδοσή του, αποδεικνύοντας ότι όντως είχε πάρα πολύ καλή γενίκευση και οι δυνατότητες του δεν περιορίζονταν κοντά στις περιοχές που είχε αρκετά δεδομένα εκπαίδευσης (Διάγραμμα 57).

3^ο Σενάριο

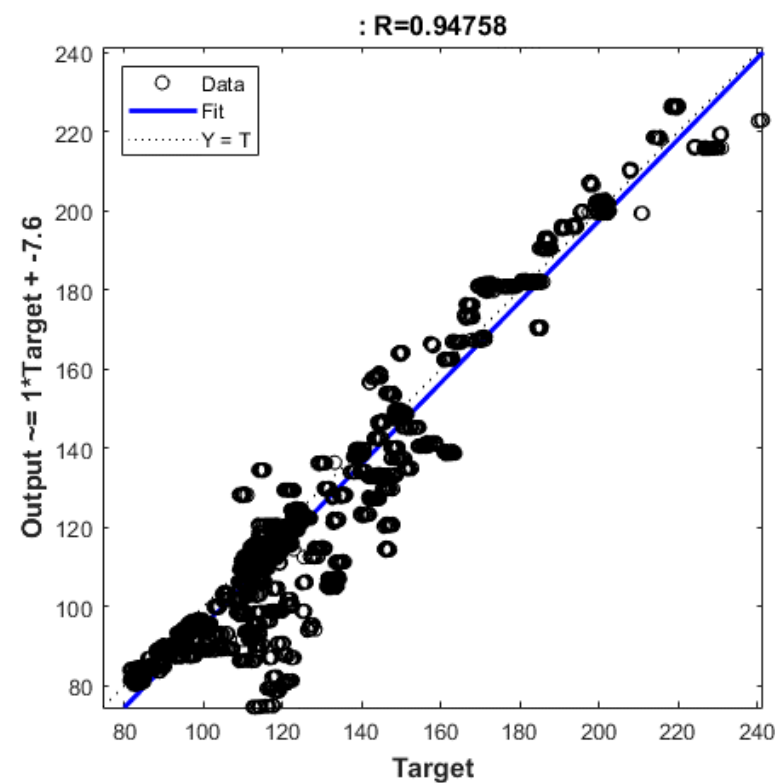
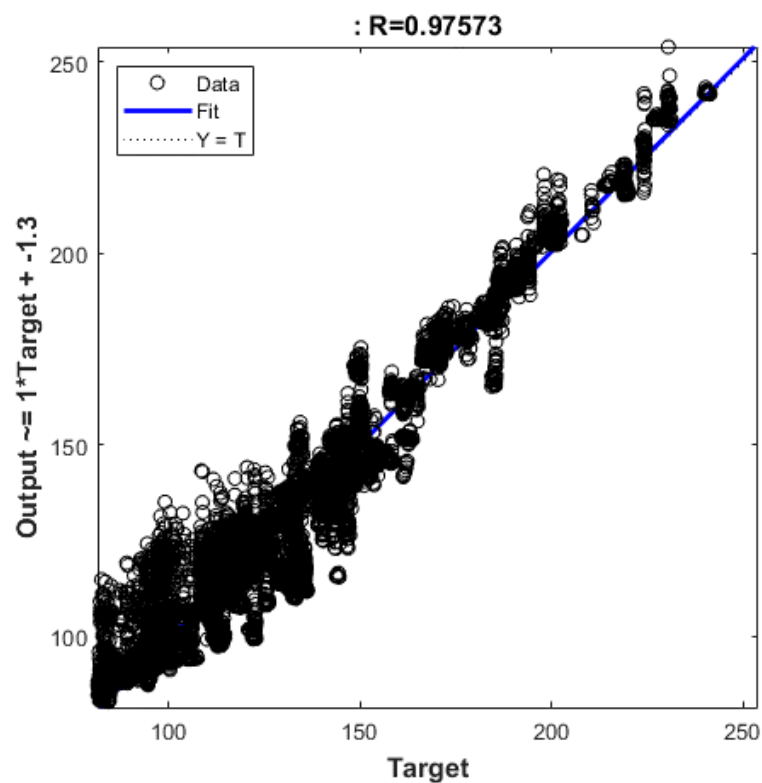
Διάγραμμα 58: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 3^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Στα σενάρια 3 και 4, οι αλγόριθμοι είχαν τις χειρότερες γενικεύσεις όπως φάνηκε στα [Διάγραμμα 53](#) [Διάγραμμα 54](#) επομένως παρόλο που μειώθηκε σχεδόν δύο τάξεις μεγέθους το *MSE* για τους αλγόριθμους στο 3^ο σενάριο, η απόδοσή τους παρέμεινε κακή ([Διάγραμμα 58](#)).

4^ο Σενάριο

Διάγραμμα 59: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 4^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Ο αλγόριθμος *Levenberg-Marquardt* είχε πολύ κακή απόδοση και σε αυτό το σενάριο, αντιθέτως ο αλγόριθμος *Bayesian Regularization* βελτίωσε αρκετά την απόδοση του, ρίχνοντας το MSE κάτω από 1000, αλλά παρόλα αυτά δεν μπορεί να ειπωθεί ότι δίνει ικανοποιητική προσέγγιση της στάθμης (Διάγραμμα 59).

5^ο Σενάριο

Διάγραμμα 60: Διάγραμμα παλινδρόμησης μεταξύ των αλγόριθμων του 5^{ου} σεναρίου με τις πραγματικές τιμές των πηγαδιών της επιλεγμένης περιοχής της Ουγγαρίας. (αριστερά είναι τα αποτελέσματα του αλγόριθμου Levenberg-Marquardt και δεξιά τα αποτελέσματα του αλγόριθμου Bayesian Regularization)

Πολύ καλή απόδοση και για τους δύο αλγόριθμους για την παραμεθόριο περιοχή της Ουγγαρίας. Οι αλγόριθμοι κατέγραψαν συντελεστές συσχέτισης 0.97 και 0.95 αντίστοιχα, πλησιάζοντας σε επιδόσεις τον αλγόριθμο Bayesian Regularization του 2^{ου} σεναρίου (Διάγραμμα 60).

ΣΥΜΠΕΡΑΣΜΑΤΑ – ΠΡΟΤΑΣΕΙΣ

Όπως αναφέρθηκε και στο θεωρητικό κομμάτι, οι υπολογιστές δεν έχουν νοημοσύνη, απλώς έχουν την ικανότητα να εκτελούν αλγόριθμους με εκπληκτική ακρίβεια και ταχύτητα. Το ότι τα Τ.Ν.Δ. δεν κατόρθωσαν να αφομοιώσουν τις μεταβολές της στάθμης του υδροφόρου ανάλογα τις περιβαλλοντικές συνθήκες, και μεν οφείλεται σε ένα βαθμό στο ότι έλειπαν αρκετά δεδομένα (πχ. Υδραυλικές αγωγιμότητες) αλλά το κυρίως πρόβλημα είναι ότι έγιναν οι λάθος ερωτήσεις προς το πρόγραμμα.

Αρχικά οι αλγόριθμοι παρουσίασαν υπερεκπαίδευση και η γενίκευση που έδωσαν ήταν σχετικά κακή. Αφού όπως αποδείχθηκε από την περίπτωση της Ουγγαρίας ότι πλην ενός αλγορίθμου οι υπόλοιποι όσο απομακρύνονταν από τα πηγάδια που χρησιμοποιήθηκαν αδυνατούσαν να δώσουν ικανοποιητικές απαντήσεις. Όμως ακόμα και με υπερεκπαίδευση οι αλγόριθμοι αδυνατούσαν να προσεγγίσουν τις διακυμάνσεις που παρουσίαζε η στάθμη των πηγαδιών. Αυτό οφείλεται στο γεγονός ότι η ερώτηση που έγινε προς τον αλγόριθμο, σε όλες τις περιπτώσεις ήταν η εύρεση της στάθμης του υδροφόρου, κάτι που απαντήθηκε από όλους τους αλγόριθμους με τετραγωνικό σφάλμα που κυμαινόταν στο 10 (τουλάχιστον στα πηγάδια ή πολύ κοντά σε αυτά). Αν γίνει η αναγωγή του σε κανονικό σφάλμα, οι αλγόριθμοι κατάφεραν να κάνουν προβλέψεις ± 3 μέτρα από την στάθμη του υδροφόρου. Την ώρα που η διακύμανση της στάθμης στην περιοχή μελέτης ήταν αρκετές εκατοντάδες μέτρα. Άρα αν παραβρεθεί η απόδοση του αλγορίθμου στην Ουγγαρία, θα έβγαине το συμπέρασμα ότι οι αλγόριθμοι έδιναν πολύ καλή πρόβλεψη για την μέση στάθμη και πολύ κακή πρόβλεψη για τις διακυμάνσεις αυτής, κάτι που φαίνεται στα [Διαγράμματα 41 έως 50](#).

Αυτό που θα πρέπει να γίνει σε μελλοντικές έρευνες είναι ζητηθεί πιο συγκεκριμένα η διακύμανση και όχι η στάθμη. Όπως πχ. Είχε κάνει ο Ιωάννης Τριχάκης ([Trichakis, Nikolos, & Karatzas, 2009](#)) που αντί να ζητάει την στάθμη του υδροφόρου, ζητούσε την ημερήσια μεταβολή. Κάτι τέτοιο δεν ήταν εφικτό στην παρούσα μελέτη καθώς δεν υπήρχε συγκεκριμένη συχνότητα των δεδομένων, επομένως δεν θα μπορούσε να εφαρμοστεί. Όμως όπως φάνηκε από τις προσομοιώσεις των σεναρίων 2 & 4 που δεν ζητούσαν το ύψος του υδροφόρου από την επιφάνεια της θάλασσας, αλλά την απόσταση του υδροφόρου από την επιφάνεια, η δεύτερη ερώτηση, επειδή το εύρος τιμών της, ήταν μικρότερο και δεν μπορούσε να συσχετιστεί άμεσα με κάποια μεταβλητή (όπως λόγου χάρη συσχετίζεται το ύψος του υδροφόρου από την επιφάνεια της θάλασσας με το υψόμετρο της περιοχής), οι γραφικές παραστάσεις από τα πηγάδια, έδειξαν ότι υπήρχε καλύτερη προσομοίωση. Η συσχέτιση που είχε το ύψος με την απόδοση των αλγορίθμων φάνηκε στην εξέταση των πηγαδιών της παραμεθόριου της Ουγγαρίας, εκεί όπου τα σεναρία 1 και 5 που λάμβαναν το υψόμετρο ως δεδομένο και έδιναν την στάθμη του υδροφόρου ως αποτέλεσμα, παρουσίασαν σαφή βελτίωση σε σχέση με τα σεναρία 3 και 4 που δεν έπαιρναν καθόλου υπόψιν τους το υψόμετρο ενώ στο σενάριο 2, ο μεν αλγόριθμος της *Bayesian Regularization* είχε πολύ καλή επίδραση σε όλη την περιοχή μελέτης και δεν μπορεί να κριθεί ο δε αλγόριθμος *Levenberg-Marquardt* δεν παρουσίασε την ίδια βελτίωση συγκριτικά με τους αλγόριθμους των σεναρίων 1 και 5.

Η ερώτηση που είχε κάνει ο Τριχάκης λοιπόν στο Τ.Ν.Δ., ήταν στην σωστή κατεύθυνση, αλλά είχε ένα βασικό μειονέκτημα. Η πρόβλεψη της μεταβολής θα μπορούσε να γίνει από τα Τ.Ν.Δ, αλλά αυτό θα μπορούσε να χρησιμοποιηθεί μόνο σε περιπτώσεις που ήταν γνώστη η προηγούμενη τιμή, κάτι που δεν είναι εφικτό να συμβαίνει σε ολόκληρη την περιοχή μελέτης. Επομένως αυτό που θα μπορούσε να γίνει είναι να υπάρχουν δύο ζητούμενα, η μία να αφορά τον μέσο όρο της στάθμης του υδροφόρου και η δεύτερη να αφορά την απόκλιση από τον μέσο

όρο. Με αυτόν τον τρόπο, το T.N.Δ. θα μπορεί να δίνει εκτίμηση για υδραυλικό ύψος και σε περιπτώσεις που δεν είναι γνωστές προηγούμενες τιμές του, ενώ και το ίδιο το T.N.Δ. θα αξιολογεί την απόδοση του με βάση και την προσομοίωση στις αποκλίσεις.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- Abrahart, R. J., Kneale, P. E., & See, L. M. (2004). *Neural Networks for Hydrological Modelling*. London, UK: Taylor & Francis Group.
- Anderson, J. A., & Rosenfeld, E. (1988). *Neurocomputing: Foundations of Reseach*. Massachusetts, USA: Massachusetts Institute of Technology.
- Anderson , M., & Woessner, W. (1992). *Applied Groundwater Modeling, Simulation of Flow and Advective Transport*. California, LA USA: Academic Press.
- Arveti, M., Gini, G., & Folgheraiter, M. (2007). Classification of EMG signals through wavelet analysis and neural networks for controlling an active hand prosthesis. *IEEE 10th International Conference on Rehabilitation Robotics*, <https://doi.org/10.1109/icorr.2007.4428476>.
- Ascott, M. J., Lapworth, D. J., Goody, D. C., Sage, R. C., & Karapanos, I. (2016). Impacts of extreme flooding on riverbank filtration water quality. *Science of The Total Environment*, <https://doi.org/10.1016/j.scitotenv.2016.02.169> .
- Battiti, R. (1992). First- and Second-Order Methods for Learning: Between Steepest Descent and Newton's Method. *Neural Computation*, <https://doi.org/10.1162/neco.1992.4.2.141>.
- Baughman, D. R., & Liu, Y. A. (1995). *Neural Networks in Bioprocessing and Chemical Engineering*. San Diego, CA: Academic Press. <https://doi.org/10.1016/B978-0-12-083030-5.50008-4>
- Beale, M. H., Hagan, M. T., & Demuth, H. B. (2017). *Neural Network Toolbox*. Natick, MA, USA: MathWorks.
- Bengio, Y., & LeCun, Y. (2007). Scaling learning algorithms towards AI. In L. Bottou, O. Chapelle, D. DeCoste, & J. Weston, *Large-Scale Kernel Machines* (pp. 321-387). Cambridge MA: MIT Press.
- Bisong, E. (2019). *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Berkeley, CA: Apress. https://doi.org/10.1007/978-1-4842-4470-8_16
- Bourlard, H. A., & Morgan, N. (1994). *Connectionist Speech Recognition, A Hybrid Approach*. Cambridge, MA: Springer.
- Brent, R. P. (1973). *Algorithms for Minimization Without Derivatives*. Englewood Cliffs, NJ, USA: Prentice-Hall.
- Burden, F., & Winkler, D. (2008). Bayesian Regularization of Neural Networks. *Artificial Neural Networks*, https://doi.org/10.1007/978-1-60327-101-1_3.
- Charalambous, C. (1992). Conjugate gradient algorithm for efficient training of artificial neural networks. *IEE Proceedings G Circuits, Devices and Systems*, <https://doi.org/10.1049/ip-g-2.1992.0050>.
- Chel, H., Majumder, A., & Nandi, D. (2011). Scaled Conjugate Gradient Algorithm in Neural Network Based Approach for Handwritten Text Recognition. *Trends in Computer*

- Science, Engineering and Information Technology*, https://doi.org/10.1007/978-3-642-24043-0_21.
- Clifton, C., Evans, R., Hayes, S., Hirji, R., Puz, G., & Pizarro, C. (2010). *Water and Climate Change: Impacts on Groundwater Resources and Adaptation Options*. Water Work. Notes. 25.
- Crane, E. B. (1965). *Artificial Intelligence Techniques*. Washington, DC.: Spartan Press.
- Dai, Y.-H., Liao, L.-Z., & Li, D. (2004). On restart procedures for the conjugate gradient method. *Numerical Algorithms*, <https://doi.org/10.1023/B:NUMA.0000021761.10993.6e>.
- Dan Foresee, F., & Hagan, M. T. (1997). Gauss-Newton Approximation to Bayesian Learning. *Proceedings of International Conference on Neural Networks*, <https://doi.org/10.1109/ICNN.1997.614194>.
- Dennis, J. E., & Schnabel, R. B. (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. New Jersey, NJ, USA: Prentice-Hall.
- Earman, S., & Dettinger, M. (2011). Potential impacts of climate change on groundwater resources – a global review. *Journal of Water and Climate Change*, 213-229 <https://doi.org/10.2166/wcc.2011.034>.
- Farley, B. G., & Clark, W. A. (1954). Simulation of self-organizing systems by digital computer. *Transactions of the IRE Professional Group on Information Theory*, <https://doi.org/10.1109/tit.1954.1057468>.
- Fletcher, R. (1987). *Practical Methods of Optimization*. New York, NY: Wiley.
- Géron, A. (2017). *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems (1st Edition)*. Sebastopol, CA: O'Reilly Media.
- Ghosh, E. N., & Sharma, K. D. (2006). *Groundwater Modelling and Management*. New Delhi: Capital Pub.
- Gill, P. E., Murray, W., & Wright, M. (1981). *Practical Optimization*. New York, NY: Academic Press.
- Gleick, P. H. (1993). *Water in Crisis: A Guide to the World's Fresh Water Resources*. New York: Oxford University Press.
- Graupe, D. (2013). *Principles of Artificial Neural Networks, Third Edition* (Τόμ. Advanced Series in Circuits and Systems - Vol.7). (W.-K. Chen, & Dieter A. Mlynski, Επιμ.) Singapore: World Scientific Publishing Co. Pte. Ltd.
- Green, T. R., Taniguchi, M., Kooi, H., Gurdak, J. J., Allen, D. M., Hiscock, K. M., . . . Aureli, A. (2011). Beneath the surface of global change: Impacts of climate change on groundwater. *Journal of Hydrology*, 532-560 <https://doi.org/10.1016/j.jhydrol.2011.05.002>.
- Grillakis, M. G. (2019). Increase in severe and extreme soil moisture droughts for Europe under climate change. *Science of the Total Environment*, <https://doi.org/10.1016/j.scitotenv.2019.01.001>.

- Gurney, K. (1997). *An Introduction to Neural Networks*. London, UK: UCL Press.
- Hagan, M. T., Demuth, H. B., Beale, M. J., & De Jesus, O. (2014). *Neural Network Design (2nd Edition)*. Amazon.
- Hagan, M. T., Demuth, H. B., Beale, M. J., & De Jesus, O. (2014). *Neural Network Design (2nd Edition)*. Boston, MA, USA: PWS.
- Hastie, T., Tibshirani, R., & Friedman, J. (2008). *The Elements of Statistical Learning, Data Mining, Inference and Prediction (2nd Edition)*. New York: Springer.
- Haykin, S. (2009). *Neural Networks and Learning Machines (3rd Edition)*. New Jersey: Pearson.
- Heaviside, O. (1899). *Electromagnetic Theory, Vol II*. London UK: The Electrician.
- Hebb, D. O. (1949). Chapter 4, The First Stage of Perception: Growth of the Assembly. In D. O. Hebb, *The Organization of Behavior* (pp. 60-78). New York, NY, USA: Wiley.
- IPCC. (2001). The Climate System: an Overview. Στο Houghton, J.T., Y. Ding, D.J. Griggs, M. Noguer, P.J. van der Linden, X. Dai, . . . C.A. Johnson (eds.), *Climate Change 2001: The Scientific Basis. Contribution of Working Group I to the Third Assessment Report of the Intergovernmental Panel on Climate Change* (σσ. 87-88). Cambridge, United Kingdom and New York, NY, USA: Cambridge University Press.
- IPCC. (2012). *Glossary of terms. In: Managing the Risks of Extreme Events and Disasters to Advance Climate Change Adaptation*. Cambridge, UK, and New York, NY, USA: Cambridge University Press.
- IPCC. (2013). Annex III: Glossary [Planton, S. (ed.)]. In: *Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change. [Stocker, T.F., D. Qin, G.-K. Plattner, M. Tignor, S.K. Allen, J. Boschung, A. Nauels, Y. Xia, V. Bex and P.M. Midgley (eds.)]*. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA., 1447-1465.
- IWMI. (2017). *Water for Food, Water for Life: A Comprehensive Assessment of Water Management in Agriculture*. London, UK: Published with the International Water Management Institute.
- James, W. (1892). Chapter XVI "Association". In J. Willian, *Prychology (Briefer Course)* (pp. 253-279). New York, NY, USA: Henry Holt and Company.
- Kim, P. (2017). *Matlab Deep Learning With Machine Learning, Neural Networks and Artificial Intelligence*. New York, USA: APress.
- Koutroulis, A. G. (2019). Dryland changes under different levels of global warming. *Science of The Total Environment*, <https://doi.org/10.1016/j.scitotenv.2018.11.215>.
- Krose, B., & van de Smagt, P. (1996). *An introduction to Neural Networks (8th edition)*. Amsterdam: The University of Amsterdam.
- Le Cun, Y. (1986). Learning Process in an Asymmetric Threshold Network. *Disordered Systems and Biological Organization*, https://doi.org/10.1007/978-3-642-82657-3_24.

- Levenberg, K. (1944). A Method For The Solution Of Certain Non-Linear Problems In Least Squares. *Quarterly of Applied Mathematics*, <https://doi.org/10.1090/qam/10666>.
- Lucky, R. W. (1965). Automatic Equalization for Digital Communication. *The Bell System Technical Journal*, <https://doi.org/10.1002/j.1538-7305.1965.tb01678.x>.
- Marquardt, D. W. (1963). An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *SIAM Journal on Applied Mathematics*, <https://doi.org/10.1137/0111030>.
- McCulloch, W. S., & Pitts, W. (1943). A Logical Calculus of the Ideas Immanent in Nervous Activity. *The Bulletin of Mathematical Biophysics*, <https://doi.org/10.1007/BF02478259>.
- Mekonnen, M. M., & Hoekstra, A. Y. (2016). Four billion people facing severe water scarcity. *Science Advances*, <https://doi.org/10.1126/sciadv.1500323>.
- Mihic, S., Golusin, M., & Mihajlovic, M. (2011). Policy and promotion of sustainable inland waterway transport in Europe – Danube River. *Renewable and Sustainable Energy Reviews*, <https://doi.org/10.1016/j.rser.2010.11.033>.
- Minsky, M., & Papert, S. (1969). *Perceptrons*. Cambridge, MA.: MIT Press.
- Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2018). *Foundations of Machine Learning (2nd Edition)*. Cambridge, Massachusetts: The MIT Press.
- Møller, M. F. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, [https://doi.org/10.1016/s0893-6080\(05\)80056-5](https://doi.org/10.1016/s0893-6080(05)80056-5).
- Moré, J. J. (1978). The Levenberg-Marquardt algorithm: Implementation and theory. *Numerical Analysis*, <https://doi.org/10.1007/bfb0067700>.
- Murata, N. (1998). A Statistical Study of On-line Learning. In D. Saad, *On-line Learning in Neural Networks* (pp. pp 63-92). Cambridge, UK: Cambridge University Press.
- Oki, T., & Kanae, S. (2006). Global Hydrological Cycles and World Water Resources. *Science*, 1068-1072 <https://doi.org/10.1126/science.1128845>.
- Parker, D. B. (1985). *Learning-Logic: Casting the Cortex of the Human Brain in Silicon. Technical Report Tr-47*. Cambridge, MA: Center for Computational Research in Economics and Management Science. MIT.
- Pavlović, P., Mitrović, M., Đorđević, D., Sakan, S., Slobodnik, J., Liška, I., . . . Paunović, M. (2016). Assessment of the contamination of riparian soil and vegetation by trace metals — A Danube River case study. *Science of The Total Environment*, <https://doi.org/10.1016/j.scitotenv.2015.06.125>.
- Picton, P. (1994). *Introduction to Neural Networks*. London, UK: Macmillan.
- Piotrowski, A. P., & Napiorkowski, J. J. (2013). A comparison of methods to avoid overfitting in neural networks training in the case of catchment runoff modelling. *Journal of Hydrology*, <https://doi.org/10.1016/j.jhydrol.2012.10.019>.
- Pitts, W. (1942). Some observations on the simple neuron circuit. *The Bulletin of Mathematical Biophysics*, <https://doi.org/doi:10.1007/bf02477942>.

- Pitts, W. (1942). The linear theory of neuron networks: The static problem. *The Bulletin of Mathematical Biophysics*, <https://doi.org/10.1007/bf02478112>.
- Pitts, W. (1943). A general theory of learning and conditioning: Part I. *Psychometrika*, <https://doi.org/10.1007/bf02288680>.
- Pitts, W. (1943). A general theory of learning and conditioning: Part II. *Psychometrika*, <https://doi.org/10.1007/bf02288697>.
- Pitts, W. (1943). The linear theory of neuron networks: The dynamic problem. *The Bulletin of Mathematical Biophysics*, <https://doi.org/10.1007/bf02478116>.
- Polyak, B. (1964). Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, [https://doi.org/10.1016/0041-5553\(64\)90137-5](https://doi.org/10.1016/0041-5553(64)90137-5).
- Postel, S. L. (2000). Enterig an era of water scarcity: the challenges ahead. *Ecological Applications*, [https://doi.org/10.1890/1051-0761\(2000\)010\[0941:EAEOWS\]2.0.CO;2](https://doi.org/10.1890/1051-0761(2000)010[0941:EAEOWS]2.0.CO;2).
- Prechelt, L. (1998). Automatic early stopping using cross validation: quantifying the criteria. *Neural Networks*, [https://doi.org/10.1016/S0893-6080\(98\)00010-0](https://doi.org/10.1016/S0893-6080(98)00010-0).
- Prechelt, L. (1998). Early Stopping — But When? Στο G. Montavon, G. B. Orr, & K.-R. Müller, *Neural Networks: Tricks of the Trade* (σσ. pp 53-67). Berlin, GE: Springer.
- Qian, N. (1999). On the momentum term in gradient descent learning algorithms. *Neural Networks*, [https://doi.org/10.1016/S0893-6080\(98\)00116-6](https://doi.org/10.1016/S0893-6080(98)00116-6).
- Riedmiller, M., & Braun, H. (1993). A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm. *IEEE International Conference on Neural Networks* (σσ. 586-591). San Francisco, CA, USA: IEEE. <https://doi.org/10.1109/ICNN.1993.298623>.
- Rochester, N., Holland, J. H., Haibt, L. H., & Duda, W. L. (1956). Tests on a cell assembly theory of the action of the brain, using a large digital computer. *IEEE Transactions on Information Theory*, <https://doi.org/10.1109/TIT.1956.1056810>.
- Rögnvaldsson, T. S. (1998). A Simple Trick for Estimating the Weight Decay Parameter. Στο G. Montavon, G. B. Orr, & K.-R. Müller, *Neural Networks: Tricks of the Trade* (σσ. 69-89). Berlin, GE: Springer.
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, <https://doi.apa.org/doiLanding?doi=10.1037/h0042519>.
- Rosenblatt, F. (1961). *Principles of neurodynamics: Perceptions and the theory of brain mechanism*. Washington, DC: Spartan Books.
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). *Learning Internal Representations by Error Propagation*. San Diego CA: University of California.
- Russel, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach (4th Edition)*. Philadelphia, PA, USA: Pearson.

- S. Siebert, J. Burke, J. M. Faures, K. Frenken, J. Hoogeveen, P. Döll, & F. T. Portmann. (2010). Groundwater use for irrigation – a global inventory. *Hydrology and Earth System Sciences*, 1863–1880 <https://doi.org/10.5194/hess-14-1863-2010>.
- Salles, A., Diebold, C., & Moss, C. F. (2020). Echolocating bats accumulate information from acoustic snapshots to predict auditory object motion. *PNAS*, <https://doi.org/10.1073/pnas.2011719117>.
- Scales, L. E. (1985). *Introduction to Non-Linear Optimization*. London, UK: Macmillan Computer Science Series.
- Schmedtje, U., Bachmann, J., Behrendt, H., Birk, S., Biza, P., D'Eugenio, J. V., . . . Zinke, A. (2005). “*Danube Basin Analysis (WFD Roof Report 2004)*”. Vienna, AT: ICPDR.
- Sommerwerk, N., Hein, T., Schneider-Jacoby, M., Baumgartner, C., Ostojić, A., Siber, R., . . . Tockner, K. (2009). The Danube River Basin. In K. Tockner, U. Uehlinger, & C. T. Robinson, *Rivers of Europe* (pp. 59-112). London UK: Academic Press. <https://doi.org/10.1016/B978-0-12-369449-2.X0001-X>
- Srivastava, N., G., Hinton, Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, (56):1929–1958.
- Thaler, S., & Furrer, D. (2005). Neural Network Modeling. *Advance Material and Processes*, 42-46.
- Theodoridis, S. (2020). *Machine Learning, A Bayesian and Optimization Perspective (2nd Edition)*. London, UK: A Bayesian and Optimization Perspective. <https://doi.org/10.1016/C2019-0-03772-7>
- Trichakis, I. C., Nikolos, I. K., & Karatzas, G. P. (2009). Optimal selection of artificial neural network parameters for the prediction of a karstic aquifer’s response. *Hydrological Processes*, <https://doi.org/10.1002/hyp.7410>.
- Turing, A. (1950). Computing Machinery and Intelligence. *Mind*, <https://doi.org/10.1093/mind/LIX.236.433>.
- Tzanakakis, V. A., Angelakis, A. N., Paranychiakis, N. V., Y. G., Dialynas, & Tchobanoglous, G. (2020). Challenges and Opportunities for Sustainable Management of Water Resources in the Island of Crete, Greece. *Water*, <https://doi.org/10.3390/w12061538>.
- United Nations. (2003). *Water for people, water for life*. Paris: UNESCO-WWAP.
- Vignati, D. A., Secrieru, D., Bogatova, Y. I., Dominik, J., Céréghino, R., Berlinsky, N. A., . . . Stanica, A. (2013). Trace element contamination in the arms of the Danube Delta (Romania/Ukraine): Current state of knowledge and future needs. *Journal of Environmental Management*, <https://doi.org/10.1016/j.jenvman.2013.04.007>.
- W.E.F. (2015). *Στο Global Risks 2015, 10th Edition*. Geneva, Switzerland: World Economic Forum.
- Werbos, P. J. (1974). *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Cambridge, Massachusetts: Ph.D. dissertation, Committee on Applied Mathematics, Harvard University.

- Widrow, B. (1962). Generalization and information storage in networks of ADALINE "neurons". Στο M. C. Yovitz, G. T. Jacobi, & G. Goldstein (Επιμ.), *Self-Organizing Systems, symposium proceedings* (σσ. 435-461). Washington, DC: Spartan Books.
- Widrow, B. (1987). ADALINE and MADALINE, Plenary Speech. *IEEE Proc. First Intl. Conf. on Neural Networks*, (σσ. pp. 143-158). San Diego, CA.
- Widrow, B., & Hoff, M. E. (1960). *Adaptive Switching Circuits*. New York: IRE WESCON Convention Record.
- Widrow, B., & Lehr, M. A. (1990). 30 Years of Adaptive Neural Networks: Perceptron, Madaline, and Backpropagation. *Proceedings of the IEEE*, <https://doi.org/10.1109/5.58323>.
- Witten, I., Frank, E., Hall, M. A., & Pal, C. J. (2017). *Data Mining, Practical Machine Learning Tools and Techniques (4th Edition)*. San Francisco,CA: Morgan Kaufmann.

ΠΑΡΑΡΤΗΜΑΤΑ

ΑΛΓΟΡΙΘΜΟΙ

ΡΥΘΗΟΝ

Η συγκεκριμένη γλώσσα προγραμματισμού επιλέχθηκε για την διαλογή και την κύρια προετοιμασία δεδομένων, καθώς ως γλώσσα προγραμματισμού έχει την ευχέρεια να αντιμετωπίζει μεταβλητές τόσο δεκαδικών, όσο και άλλων τύπων (πχ κειμένου) κάτι που διευκολύνει την διαδικασία διαλογής, όπου υπάρχουν διαφορετικοί τύποι αρχείων, διαμορφώσεις και σε πολλές περιπτώσεις κείμενα και αριθμοί μαζί.

Πριν την χρήση της python, ήταν απαραίτητο να δημιουργηθεί μια λίστα των αρχείων χρονοσειρών της εκάστοτε χώρας καθώς και αυτή η λίστα να εμπλουτιστεί με στοιχεία όπως οι συντεταγμένες των σταθμών και κάποιων κωδικών ονομασιών που καθιστούν εφικτή την ενσωμάτωση των μετεωρολογικών δεδομένων (Εικόνα 45).

```
Volume in drive C is Windows
Volume Serial Number is SerialNumber

Directory of C:\...\Austria

10/14/2020  10:19 AM    <DIR>          .
10/14/2020  10:19 AM    <DIR>          ..
06/08/2016  08:57 PM                32,110 GWS-Monmit-300012.csv
06/08/2016  08:57 PM                26,877 GWS-Monmit-300020.csv
...
06/08/2016  09:02 PM                1,204 GWS-Monmit-379735.csv
06/08/2016  09:02 PM                1,924 GWS-Monmit-379743.csv
10/14/2020  10:19 AM                 0 list.txt
3410 File(s)      45,378,702 bytes
2 Dir(s)  120,746,754,048 bytes free
```

Εικόνα 45: Δημιουργία λίστας των χρονοσειρών με χρήση MS-DOS

Η λίστα των χρονοσειρών πραγματοποιήθηκε με μια εντολή του DOS, στην συνέχεια τα δεδομένα μεταφέρθηκαν σε περιβάλλον excel όπου και ενσωματώθηκαν τα δεδομένα από το ArcGIS. Η τελική μορφή αυτών των αρχείων παρουσιάζεται στην παρακάτω εικόνα (Εικόνα 46).

	A	B	C	D	E	F
1	Station Name	ID Code	X	Y	Z	Meteo Code
2	GWS-Monmit-300012.csv	516	4796361.3358	2794738.2306	174.813	516
3	GWS-Monmit-300020.csv	517	4804828.5109	2799168.2248	182.142	517
4	GWS-Monmit-300103.csv	2934	4779649.6875	2755653.3816	352.354	2934
5	GWS-Monmit-300111.csv	518	4826220.0651	2791367.5235	161.663	518
6	GWS-Monmit-300137.csv	518	4826445.7459	2790705.9835	154.949	519
7	GWS-Monmit-300152.csv	2935	4804640.2946	2779843.1106	194.588	2935
8	GWS-Monmit-300186.csv	520	4806608.8254	2791775.3463	180.057	520
9	GWS-Monmit-300202.csv	520	4807596.7864	2791543.3214	172.084	521
10	GWS-Monmit-300228.csv	522	4798627.8311	2781857.4643	201.367	522

Εικόνα 46: Λίστα των χρονοσειρών με ενσωματωμένη πληροφορία των συντεταγμένων του εκάστοτε σταθμού.

Αρχικά δημιουργήθηκε ένα απλό κομμάτι κώδικα στο οποίο πραγματοποιείτε η επιλογή της χώρας στην οποία θα γίνει επεξεργασία των χρονοσειρών

```
# This program is a part of the thesis
# "Use of artificial neural networks to determine the groundwater levels
# in the Danube catchment area"
# The program can read data from three different format types and create
# one file which include all the data of each country

check = True
folder_path = 'C:\....'
while check:
    choice = input("""
Select country, Press
1, for Austria
2, for Germany
3, for Croatia
or type exit to terminate the program
""")
    if choice == '1':
        country = 'Austria'
        check = False
    elif choice == '2':
        country = 'Germany'
        check = False
    elif choice == '3':
        country = 'Croatia'
        check = False
    elif choice.lower() == 'exit':
        quit()
    else:
        print("Your selection isn't valid")
        print("Make a Proper selection")

country_path = folder_path+country+'\\'
# Import the list of timeseries
arguments = read_list(country_path)
for i in range(len(arguments)):
    # Read data from each timeseries and make proper transformations
    dates, values, extension = proper_reader(
        country, country_path+arguments[i][0])
# Calculations and extraction of data in csv format
write_data(country_path+"output\\" +
            arguments[i][0]+extension, arguments[i], dates, values)
merge_files(country, country_path)
```

Η πρώτη συνάρτηση που εκτελείτε, διαβάζει την λίστα που δημιουργήθηκε με τα ονόματα των αρχείων χρονοσειρών, καθώς και τους κωδικούς των σταθμών και τις συντεταγμένες τους.

```
def read_list(country_path):  
    # This function reads a csv file which contains a list of the  
    # timeseries, as long as station codes and coordinates  
    import xlrd  
    path = country_path+'list.xls'  
    workbook = xlrd.open_workbook(path, 'r')  
    workbook_sheet = workbook.sheet_by_index(0)  
    arguments = []  
    for row in range(1, workbook_sheet.nrows):  
        arguments.append([]*6)  
        for column in range(6):  
            cell_value = ((str(workbook_sheet.cell(row, column))  
                           ).split(':')[ -1]).replace("'", "")  
            arguments[row-1].append(cell_value)  
    return arguments
```

Στην συνέχεια καλείται η συνάρτηση για την επιλογή της κατάλληλης υπορουτίνας για την ανάγνωση των αρχείων χρονοσειρών της εκάστοτε χώρας, καθώς και για να διαχειριστεί την διαφορά στους τύπους αρχείων που είναι αποθηκευμένες οι εκάστοτε πληροφορίες, ώστε στο τέλος να προκύψουν αρχεία κειμένου με την ίδια διαμόρφωση.

```
def proper_reader(country, country_path):  
    # This function, choose the appropriate function to read the specific  
    # format of each country. Apart from this some modifications in the  
    # paths and filenames are made, due to the differences in the file  
    # format of each country. Finally, the function returns one vector with  
    # the dates and one with the water level at the time.  
    if country == 'Austria':  
        dates, values = read_austria_data(country_path)  
        extension = ""  
    if country == 'Germany':  
        extension = ".csv"  
        dates, values = read_germany_data(country_path+extension)  
    if country == 'Croatia':  
        dates, values = read_croatia_data(country_path)  
        extension = '.csv'  
    return dates, values, extension
```

Στην συνέχεια ανάλογα την χώρα εκτελείτε μία από τις επόμενες τρεις συναρτήσεις.

```
# The following three functions are used in order to parse the data from each
# country. Due to the fact that the files might contain a number of no-data
# lines like labels, false lines etc. these functions read the data with
# different format and transformed it, in the suitable format in order to
# carry out further data checks. Finally, the function returns the proper
# vectors of date and values.
```

```
def read_austria_data(path):
# This function is used in order to parse the data from austrian dataserries
    dates = []
    values = []
    yesterday = 'yesterday'
    input_file = open(path, 'r')
    for line in input_file.readlines():
        try:
            date, value = line.split(";")
            date = (date.split())[0].replace(".", "/")
        except:
            date = '01/01/1900'
            value = 0
        dates, values, yesterday = add_data(
            dates, values, value, date, yesterday)
    input_file.close()
    return dates, values
```

```
def read_germany_data(path):
# This function is used in order to parse the data from German dataserries
    dates = []
    values = []
    yesterday = 'yesterday'
    input_file = open(path, 'r')
    for line in input_file.readlines():
        try:
            date, value = line.split(",")[3:5]
            date = (date.split())[0]
            year, month, day = date.split('-')
            date = day+'/' +month+'/' +year
        except:
            date = '01/01/1900'
            value = 0
        dates, values, yesterday = add_data(
            dates, values, value, date, yesterday)
    input_file.close()
    return dates, values
```

```

def read_croatia_data(path):
    # This function is used in order to parse the data from Croatian dataseries
    import openpyxl
    import datetime
    dates = []
    values = []
    yesterday = 'yesterday'
    input_file = openpyxl.load_workbook(
        path+'.xlsx', read_only=True, data_only=True)
    sheet = input_file.active
    for row in range(sheet.max_row):
        date = sheet.cell(row+1, 1).value
        value = sheet.cell(row+1, 2).value
        try:
            day = str(date.day)
            month = str(date.month)
            year = str(date.year)
            date = day+'/' +month+'/' +year
        except:
            date = '01/01/1900'
        dates, values, yesterday = add_data(
            dates, values, value, date, yesterday)
    return dates, values

```

Αφού ολοκληρωθεί η ανάγνωση των δεδομένων, καλείτε μια συνάρτηση για να ελέγξει εάν τα δεδομένα πληρούν τις προϋποθέσεις για να συμπεριληφθούν στην εκπαίδευση των Τ.Ν.Δ.

```

def add_data(dates, values, value, date, yesterday):
    # this function checks if the date is within the study period and if no
    # error was made by the measurement recorder
    try:
        day, month, year = date.split('/')
    except:
        day = month = year = '1'
    date = day+'/' +month+'/' +year
    if not date == yesterday:
        if '1990' <= year < '2013' or year == '2013' and month < '11':
            if isinstance(value, int):
                value = float(value)
            if not isinstance(value, float):
                try:
                    value = float(value.replace(",", "."))
                except:
                    value = 0
            if 1 < value < 2000:
                dates.append(date)
                values.append(value)
                yesterday = date
    return dates, values, yesterday

```

Στην συνέχεια καλείται η συνάρτηση που θα αποθηκεύσει προσωρινά τα δεδομένα

```
def write_data(path, arguments, dates, values):  
    # This function exports the results in a csv file  
    station_name, id_code, x, y, z, meteo_code = arguments  
    path = path  
    output = open(path, 'w')  
    for line in range(0, len(values)):  
        days, daysofyear, year = how_many_days(dates[line])  
        output.write(f"{station_name},{id_code},{dates[line]}," +  
                    f"{meteo_code},{days},{daysofyear},{year}," +  
                    f"{x},{y},{z},{values[line]:9.4f}\n")  
    output.close()
```

Κατά την εκτέλεση της προηγούμενης συνάρτησης χρησιμοποιείτε μια συνάρτηση υπολογισμού. Η χρήση της συνάρτησης αυτής επιλέχθηκε να γίνει κατά την έγγραφη των δεδομένων, έτσι ώστε να μην υπάρχει επιβάρυνση του κώδικα με επιπλέον πίνακες που θα κατανάλωναν πόρους του συστήματος, επιβραδύνοντας την ολοκλήρωση της διαδικασίας.

```
def how_many_days(date):  
    # This function calculates the time period between 31/12/1989 and the date  
    # the measurement was taken. Also, it calculates the day of the year and  
    # the year that measurement was taken.  
    import datetime  
    date = datetime.datetime.strptime(date, '%d/%m/%Y')  
    days = (date - datetime.datetime.strptime(  
        '31/12/1989', '%d/%m/%Y')).days  
    jaar = "01/01/"+str(date.year)  
    daysofyear = (date - datetime.datetime.strptime(  
        jaar, '%d/%m/%Y')).days+1  
    return days, daysofyear, date.year
```

Σε αυτό το σημείο έχει ολοκληρωθεί η διαμόρφωση των δεδομένων στο επιθυμητό format. Αυτό που μένει για την ολοκλήρωση της προεπεξεργασίας των δεδομένων, είναι η συρραφή των αρχείων που έχουν δημιουργηθεί σε ένα αρχείο. Η διαδικασία αυτή μπορεί να πραγματοποιηθεί μέσω DOS, αλλά είναι εφικτό να γίνει και μέσω Python. Η ακόλουθη υπορουτινά πραγματοποιεί αυτή την συρραφή.


```
def merge_files(country, country_path):  
    # This function, reads all csv files in the destination folder, and  
    # then it combines them into one file, named after the country  
    import glob  
    import os  
    import pandas as pd  
    extension = 'csv'  
    os.chdir(country_path)  
    all_filenames = [i for i in glob.glob('*.{}'.format(extension))]  
    # combine all files in the list  
    country_csv = pd.concat([pd.read_csv(f, header=None)  
                             for f in all_filenames])  
  
    # export to csv  
    country_csv.to_csv(country+".csv", index=False,  
                       encoding='utf-8-sig', header=None)
```

MATLAB

Τα δεδομένα στην συνέχεια εισήχθησαν στο MATLAB όπου και έγινε η ενσωμάτωση των μετεωρολογικών δεδομένων, αφού αυτά φορτώθηκαν σε αντίστοιχους πίνακες E0, ES, ET, PCP, και Tavg.

```
%This script imports the meteorological data about Evaporation,
%Potential Evaporation, Evapotranspiration, Precipitation and Average
%Temperature, as long as the Geohydrological data and the coordinates of
%the stations, in the end it produces the input table and the output
%data for supervised training of a feedforward artificial neural network
clc
clear
%E0 is a matrix with the Evaporation data, lines stand for dates,
%columns for the geohydrological stations
delimiterIn = ' ';
E0=importdata('C:\...\E0.csv',delimiterIn);
%ES is a table with the Potential Evaporation data, lines stand for
%dates, columns for the stations
ES=importdata('C:\...\ES.csv', delimiterIn);
%ET is a table with the Evapotranspiration data, lines stand for dates,
%columns for the geohydrological stations
ET=importdata('C:\...\ET.csv', delimiterIn);
%PCP is a table with the Precipitation data, lines stand for dates,
%columns for the geohydrological stations
PCP=importdata('C:\...\PCP.csv', delimiterIn);
%Tavg is a table with the Average Temperature data, lines stand for
%dates, columns for the geohydrological stations
Tavg=importdata('C:\...\Tavg.csv', delimiterIn);
Data=importdata('C:\...\data.csv', ',');
%Data is a table with the Groundwater Levels, every line contains the
%following information geohydrological station meteorological identifier,
%days difference from 31/12/1989,year, days of the year, x coordinate,
%y coordinate, z coordinate, groundwater level
E0 = table2array(E0);
ES = table2array(ES);
ET = table2array(ET);
PCP = table2array(PCP);
Tavg = table2array(Tavg);
Data = table2array(Data(:,4:end));
[rows, columns] = size(Data);
days = 3;
line = 0;
for row = 1:rows
    if Data(row,2)>=days
        line = line+1;
        Input_data(line,1:5) = Data(row, 3 : 7);
        Output_data(line) = Data(row , 8);
        for d = 0:(days-1)
            Input_data(line,6+d*5) = E0(Data(row,2)+d,Data(row,1)+1);
            Input_data(line,7+d*5) = ES(Data(row,2)+d,Data(row,1)+1);
            Input_data(line,8+d*5) = ET(Data(row,2)+d,Data(row,1)+1);
            Input_data(line,9+d*5) = PCP(Data(row,2)+d,Data(row,1)+1);
            Input_data(line,10+d*5) = Tavg(Data(row,2)+d,Data(row,1)+1);
        end
    end
end
```