

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ	3
ΚΕΦΑΛΑΙΟ 1: Ανασκόπηση βιβλιογραφίας.....	5
1.1 Εισαγωγή.....	5
1.2 Μελέτες περιπτώσεων	6
1.3 Πληροφοριακά συστήματα πρόβλεψης ταχύτητας του ανέμου.....	12
ΚΕΦΑΛΑΙΟ 2: Παρουσίαση των μοντέλων AR, ARMA και ANFIS.....	17
2.1 Μοντέλο AR.....	17
2.2 Μοντέλο ARMA.....	19
2.3 ANFIS	20
ΚΕΦΑΛΑΙΟ 3: Μέθοδοι για προσδιορισμό συστήματος.....	24
3.1 Προσδιορισμός συστήματος: Μια εισαγωγή	25
3.2 Εκτιμητής ελαχίστων τετραγώνων (Least-squares estimator)	27
3.3 Οπισθοδρόμηση για πρωσοτροφοδοτούμενα δίκτυα	32
3.4 Υβριδικός αλγόριθμος εκμάθησης: Συνδυασμός βαθμωτής ελαχιστοποίησης και εκτιμητή ελαχίστων τετραγώνων	39
ΚΕΦΑΛΑΙΟ 4: Παρουσίαση του συγκεκριμένου μοντέλου ANFIS	43
4.1 Εισαγωγή.....	43
4.2 Περιγραφή του μοντέλου	45
ΚΕΦΑΛΑΙΟ 5: Παρουσίαση αποτελεσμάτων μοντέλου ANFIS	45
4.3 Αποτελέσματα	49
4.4 Συμπεράσματα.....	50
ΠΑΡΑΡΤΗΜΑ	52
<u>ΒΙΒΛΙΟΓΡΑΦΙΑ.....</u>	79

ΠΕΡΙΛΗΨΗ

Είναι γνωστό ότι η διαδικασία πρόβλεψης της ταχύτητας του ανέμου, είναι πολύ διαδεδομένη σε πολλούς επιστήμονες, οι οποίοι ασχολούνται με τις ανανεώσιμες πηγές ενέργειας. Η παρούσα διπλωματική είναι ουσιαστικά μια προσπάθεια πρόβλεψης της ενέργειας του ανέμου, με τη βοήθεια προσαρμοστικών νευρωνικών δικτύων με νευρο-ασαφή λογική. Συγκεκριμένα, γίνεται χρήση του μοντέλου ANFIS το οποίο το δοκιμάζουμε με δυο διαφορετικές εισόδους: τη μέση ταχύτητα του ανέμου και τη μέγιστη ταχύτητα του ανέμου. Η έξοδος του μοντέλου είναι η τιμή της ενέργειας του ανέμου που προβλέπεται ότι θα πάρουμε από το μοντέλο μας, έτσι ώστε να μπορούμε να προγραμματίζουμε καλύτερα την ημερήσια παραγωγή ρεύματος από τον άνεμο. Σε γενικές γραμμές το μοντέλο ANFIS χρησιμοποιεί ένα συνδυασμό της μεθόδου εκτίμησης των ελαχίστων τετραγώνων και της οπισθοδρόμησης για προσωποδοτούμενα δίκτυα, έτσι ώστε να εκτιμήσει το βέλτιστο προβλεπόμενο φορτίο για κάθε μέρα (εκτενέστερη ανάλυση για το προσδιορισμό του συστήματος και τις παραπάνω μεθόδους θα ακολουθήσει σε επόμενο κεφάλαιο της διπλωματικής). Προφανώς για τον έλεγχο της ακρίβειας των αποτελεσμάτων θα χρησιμοποιηθούν κάποια μέτρα υπολογισμού σφάλματος, με σκοπό να βρεθεί το ελάχιστο δυνατό σφάλμα το οποίο δεν θα αλλοιώνει τη ποιότητα της εξόδου του μοντέλου μας. Μάλιστα το προτεινόμενο μοντέλο δοκιμάστηκε σε πρακτικό επίπεδο σε ένα αιολικό πάρκο στην Εύβοια με ομολογουμένως μεγάλη επιτυχία. Επίσης, τα αποτελέσματα του νευρο-ασαφούς μοντέλου, συγκρίθηκαν με τα αποτελέσματα και άλλων μοντέλων πρόβλεψης που χρησιμοποιούν χρονοσειρές, όπως τα AR (autoregressive) και ARMA (autoregressive with moving average).

Η παρούσα διπλωματική εργασία περιλαμβάνει, ανά κεφάλαιο, τα εξής:

Στο **πρώτο** κεφάλαιο γίνεται μια σύντομη ιστορική αναδρομή και μια αρκετά προσεκτική ανασκόπηση της βιβλιογραφίας, που έχει να κάνει με θέματα και μελέτες περιπτώσεων που αφορούν την πρόβλεψη του ανέμου με διάφορες μεθόδους.

Στο **δεύτερο** κεφάλαιο γίνεται μια λεπτομερής ανάλυση των μοντέλων πρόβλεψης AR και ARMA καθώς και μια γενική περιγραφή του μοντέλου ANFIS.

Στο **τρίτο** κεφάλαιο γίνεται παρουσίαση και ανάλυση των μεθόδων που χρησιμοποιεί το ANFIS για να προσδιορίσει τις παραμέτρους ενός άγνωστου συστήματος από τα δεδομένα εισόδου-εξόδου.

Στο **τέταρτο** κεφάλαιο γίνεται η παρουσίαση του μοντέλου ANFIS που χρησιμοποιήθηκε και η μεθοδολογία που ακολουθήσαμε.

Στο **πέμπτο** και τελευταίο κεφάλαιο γίνεται η σύγκριση των αποτελεσμάτων που προκύπτουν από το μοντέλο ANFIS με τα μοντέλα πρόβλεψης AR και ARMA , καθώς και τα συμπεράσματα που προκύπτουν από την παρούσα έρευνα.

Στο **τέλος** ακολουθεί το παράρτημα με τη δομή των αλγορίθμων που χρησιμοποιήθηκαν κατά την εκπόνηση αυτής της διπλωματικής εργασίας, καθώς και η απαραίτητη βιβλιογραφία.

ΚΕΦΑΛΑΙΟ 1: ΑΝΑΣΚΟΠΗΣΗ ΒΙΒΛΙΟΓΡΑΦΙΑΣ

1.1 ΕΙΣΑΓΩΓΗ

Μέσα σε λίγο καιρό η παραγωγή ενέργειας από τον άνεμο έγινε αποδεκτή από την κοινωνία, την βιομηχανία και την πολιτική ηγεσία του τόπου σαν καθαρή, πρακτική οικονομική και φιλική προς το περιβάλλον εναλλακτική. Μετά την πετρελαϊκή κρίση του 1973 οι ανανεώσιμες πηγές ενέργειας άρχισαν να εμφανίζονται στην ατζέντα των ισχυρών χωρών της ευρωπαϊκής ένωσης και συγκεκριμένα η παραγωγή ενέργειας με τη βοήθεια του ανέμου αποκτούσε ολοένα και περισσότερο ενδιαφέρον. Οι Ανανεώσιμες Πηγές Ενέργειας (Α.Π.Ε) είναι η πρώτη μορφή ενέργειας που χρησιμοποιήθηκε από τον άνθρωπο και αποτελούσε σχεδόν αποκλειστική πηγή ενέργειας μέχρι και τις αρχές του προηγούμενου αιώνα, όπου ο άνθρωπος στράφηκε έντονα στην χρήση ορυκτών καυσίμων και υδρογονανθράκων. Οι γνωστές σε εμάς έως τώρα ανανεώσιμες πηγές ενέργειας είναι : ηλιακή , γεωθερμική, βιομάζα , υδροηλεκτρική, αιολική. Τεχνολογικά επιτεύγματα όπως οι υδροηλεκτρικοί σταθμοί, τα ηλιακά συστήματα, οι ατμοπαραγωγοί και τελικά τα αιολικά συστήματα βρήκαν εφαρμογή στην δέσμευση ενέργειας από ΑΠΕ, ενώ παράλληλα βοήθησαν στην αποτελεσματικότερη αξιοποίηση της παρεχόμενης ενέργειας. Η οικονομική τους βιωσιμότητα έρχεται να επιστεγάσει το σύνολο των εμπορικών εφαρμογών τους εδώ και δεκαετίες. Παρόλες τις διαφοροποιήσεις που συναντούμε ανάμεσα στις ήδη υπάρχουσες μορφές ανανεώσιμων πηγών, διακρίνουμε και ορισμένα κοινά χαρακτηριστικά, όπως: Είναι πρακτικά ανεξάντλητες, έχουν χαμηλό λειτουργικό κόστος, βοηθούν στην οικονομική και κοινωνική αναβάθμιση στο τόπο εγκατάστασης, βοηθούν στην ορθολογικότερη διαχείριση των ενεργειακών πόρων, είναι φιλικές προς το περιβάλλον και στον άνθρωπο, συνεισφέρουν στην ενίσχυση της ενεργειακής ανεξαρτησίας λόγω της τοπικής τους εμφάνισης, καλύπτουν τις ευρύτερες ενεργειακές ανάγκες σε τοπικό και περιφερειακό επίπεδο. Όσο αναφορά την αιολική ενέργεια πρέπει να σημειωθεί ότι οι άνεμοι δημιουργούνται καθώς ηλιακή ενέργεια φθάνει και απορροφάται ανομοιογενώς από τις διάφορες επιφάνειες της γης. Έτσι εφαρμόζονται διαφορές θερμοκρασίας, πυκνότητας και πίεσης στην ατμόσφαιρα της, οι οποίες δημιουργούν την κίνηση του αέρα. Σε ευρύτερη κλίμακα αυτά τα ρεύματα αέρος λειτουργούν σαν τεράστια μέσα μεταφοράς ενέργειας. Το δυναμικό της αιολικής ενέργειας μιας περιοχής χαρακτηρίζεται συνήθως από την μέση ετήσια ταχύτητα ανέμου (σε m/s) παρουσιάζοντας αποκλίσεις ανά χρονικά διαστήματα. Η ταχύτητα ανέμου επηρεάζεται από το γεωγραφικό ύψος και από τις κλιματολογικές συνθήκες. Συνήθως μετρήσεις λαμβάνονται κοντά στην περιοχή ύψους των 10 μέτρων αλλά υπάρχουν εμπειρικές σχέσεις διόρθωσης της ταχύτητας ανέμου συναρτήσεως του ενδιαφερομένου ύψους. Οι εφαρμογές της αιολικής ενέργειας διακρίνονται ως προς την ενεργειακή τους χρήση σε : μηχανική, θερμική και ηλεκτρική ενέργεια. Σαν αποτέλεσμα εντατικών ερευνών σε αυτό το τομέα, η αιολική ενέργεια άρχισε να έχει εφαρμογές σε ποικίλες βιομηχανίες. Σκοπός της παρούσας εργασίας είναι η πρόβλεψη παραγωγής ενέργειας του ανέμου με χρήση νευρο-ασαφών τεχνικών οι οποίες βρίσκουν πρακτικές εφαρμογές σε πλήθος επιστημονικών πεδίων όπως η ανάλυση πολύπλοκων γραμμικών και μη γραμμικών συστημάτων.

Επιπλέον εξετάζουμε και συγκρίνουμε διάφορες τεχνικές τεχνητής νοημοσύνης βασισμένες σε προσεγγίσεις που χρησιμοποιούν χρονολογικές σειρές, δηλαδή οι είσοδοι του μοντέλου να είναι προηγούμενοι έξοδοί του, πράγμα αρκετά δύσκολο για να προβλεφθεί. Όπως λοιπόν καταλαβαίνει κανείς η πρόβλεψη της ταχύτητας του ανέμου που ουσιαστικά είναι το βασικότερο κριτήριο για την πρόβλεψη παραγωγής ενέργειας του ανέμου είναι ένα βασικό εργαλείο για όλους τους παραγωγούς ενέργειας που ασχολούνται με αιολικά πάρκα, έτσι ώστε να τους βοηθήσει να πάρουν τις καλύτερες αποφάσεις όσο αναφορά τη μετέπειτα πώληση της ενέργειας που έχει παραχθεί. Αυτές οι αποφάσεις είναι μείζονος σημασίας στην αγορά ηλεκτρικής ενέργειας εξαιτίας των οικονομικών οφελών που αποκομίζεται καθώς και της αυξημένης κερδοφορίας που επαφίεται σε αυτές. Επιπλέον όλοι όσοι ασχολούνται με αυτό το χώρο γνωρίζουν ότι η ταχύτητα του ανέμου αλλάζει με το χρόνο και δεν βρίσκεται συνεχώς στην ίδια περιοχή κατά τη διάρκεια του έτους. Για τον λόγο αυτό στο σχεδιασμό, στο προγραμματισμό και στη λειτουργία και συντήρηση των αιολικών πάρκων η μεταβολές της ταχύτητας κατά διαστήματα είναι ένα σημαντικό μέγεθος που πρέπει να προβλεφθεί. Αν και οι μελέτες για την παραγωγή ενέργειας με τη βοήθεια του ανέμου είναι πάρα πολλές σε αριθμό, δεν συμβαίνει το ίδιο και στο επιστημονικό πεδίο της δημιουργίας μοντέλων πρόβλεψης και εκτίμησης της ταχύτητας του ανέμου. Βέβαια οι χάρτες ταχύτητας του ανέμου παρέχουν μια κοινή βάση για περιφερειακές εκτιμήσεις και αλληλεπιδράσεις χωρίς όμως περιφερειακές προβλέψεις. Η δυσκολία τελικά στο να προβλέψει κανείς αυτό το μετεωρολογικό φαινόμενο βρίσκεται στο γεγονός ότι είναι αποτέλεσμα πολύπλοκων αλληλεπιδράσεων μεταξύ ισχυρών μηχανισμών μεγάλης κλίμακας όπως είναι οι διαφορές θερμοκρασίας και πίεσης, η περιστροφή της γης, τα τοπικά χαρακτηριστικά του επιπέδου της περιοχής που πρόκειται να εγκατασταθεί το αιολικό πάρκο.

Ακόμα είναι γνωστό από στοιχεία της ευρωπαϊκής ένωσης ότι η εως τώρα χρησιμοποιούμενη ενέργεια από τον άνεμο είναι περίπου 20GW ενώ σύμφωνα με τα όσα έχουν υπογραφεί στο πρωτόκολλο του Κιότο και τις ντιρεκτίβες της ευρωπαϊκής ένωσης θα πρέπει η παραγωγή ενέργειας από τον άνεμο να φθάσει τα 40-60GW. Προς αυτή τη κατεύθυνση θα βοηθήσει προφανώς και η απελευθέρωση της αγοράς ενέργειας με την είσοδο ανεξάρτητων παραγωγών που όπως φαίνεται έχουν στη διάθεσή τους πολλά καινοτόμα τεχνολογικά εργαλεία για την πρόβλεψη του ανέμου όπως θα διαπιστώσουμε και στην συνέχεια.

1.2 Μελέτες περιπτώσεων και αντίστοιχη βιβλιογραφία

Αρχικά καλό θα ήταν να δούμε τέσσερις βασικές προσεγγίσεις μοντέλων πρόβλεψης για τη βραχυπρόθεσμη πρόβλεψη της ταχύτητας του ανέμου: Persistence, ARIMA, Moving averages and artificial neural networks. Στο σημείο αυτό καλό θα ήταν να σημειώσουμε ότι για κάθε μοντέλο υπολογίζουμε τη ρίζα του μέσου τετραγωνικού σφάλματος του, ποσότητα η οποία αποτελεί ένα μέτρο αξιολόγησης της ακρίβειας του μοντέλου που εξετάζεται κάθε φορά. 1) **Persistence** : αυτό το μοντέλο πρόβλεψης είναι το πιο βασικό από τα τέσσερα για μια βραχυχρόνια πρόβλεψη της ταχύτητας του ανέμου. Προυποθέτει ότι το μέλλον έχει την τάση να προσαρμόζεται στα δεδομένα του παρόντος και για αυτό το λόγο είναι πολύ εύκολο και οικονομικό να εφαρμοσθεί με πολύ καλά μάλιστα αποτελέσματα στη πρόβλεψη της ταχύτητας του ανέμου σε διάστημα 1-5 ωρών.

2) **Box-Jenkins (ARIMA)**: πρόκειται για την γνωστή οικογένεια αυτοπαλίνδρομων μοντέλων πρόβλεψης με κινούμενους μέσους όρους. Δυο παραλλαγές αυτών των

μοντέλων είναι πάρα πολύ διαδεδομένες στο χώρο της πρόβλεψης της παραγωγής του ανέμου. Το ARIMA (2,0,2) και το ARIMA (2,1,2)

3) **Moving average** : τα μοντέλα κινούμενου μέσου όρου είναι πολύ διαδεδομένα σε πλήθος εφαρμογών πρόβλεψης όπως η πρόβλεψη αποθέματος σε μια αποθήκη σε σχέση με τη ζήτηση που υπάρχει στην αγορά. Σε αυτό το μοντέλο γίνεται μια εκτίμηση των περιόδων που εξετάζονται και κατά το τελικό στάδιο της πρόβλεψης κάνουμε εκτίμηση με το μέσο όρο των περιόδων που μας ενδιαφέρουν.

4) **artificial neural networks** : τα τεχνητά νευρωνικά δίκτυα είναι πάρα πολύ διαδεδομένα στον ευρύτερο χώρο της πρόβλεψης. Για την πρόβλεψη της παραγωγής του ανέμου χρησιμοποιούμε συγκεκριμένα νευρωνικά δίκτυα με πέντε στρώματα επειδή παρουσιάζουν μεγαλύτερη ακρίβεια στην πρόβλεψη συγκρίνοντας τα με νευρωνικά δίκτυα με δυο στρώματα. Η δομή του μοντέλου είναι η παρακάτω :

Layer	Layer	No. of	Activation
	Type	Nodes	Function
1	Input	3	Linear
2	Hidden	2	Sigmoid
3	Hidden	2	Gaussian
4	Output	1	Sigmoid

Στους πίνακες που ακολουθούν γίνεται η σύγκριση των αναφερθέντων μοντέλων σε διάφορες χρονικές περιόδους μέσω του υπολογισμού της ρίζας του μέσου τετραγωνικού σφάλματος.

	Actual	Persistence	ARIMA1	ARIMA2	MA	ANN
1 Hr Ave	3.78	2.3	3.21	3.51	3.60	3.76
RMSE	-	1.48	0.46	0.26	0.18	0.003

πίνακας 1: αποτελέσματα με χρονικό ορίζοντα πρόβλεψης μιας ώρας.

	Persistence	ARIMA1	ARIMA2	MA	ANN
2 Hour RMSE	1.07	0.37	0.44	0.25	0.30
5 Hour RMSE	1.43	0.28	0.35	0.25	0.29
12 Hour RMSE	2.31	0.92	1.37	0.28	0.39

πίνακας 2 : αποτελέσματα με χρονικούς ορίζοντες πρόβλεψης 2,5,12 ωρών

Εάν ανατρέξουμε στη βιβλιογραφία μπορούμε να βρούμε και άλλα πολύ βασικά μοντέλα πρόβλεψης που σχετίζονται με τη πρόβλεψη της ταχύτητας του ανέμου. Μερικά από αυτά παρουσιάζονται στη συνέχεια. 1) **Feed-forward neural networks** : Αυτά τα νευρωνικά δίκτυα δεν έχουν κρυμμένα στρώματα και η συνάρτηση μεταφοράς του στρώματος της εξόδου είναι γραμμική. Τα βάρη και οι διαφοροί άλλοι βασικοί παράγοντες των δικτύων αυτών υπολογίζονται με μια παραλλαγή της μεθόδου των ελαχίστων τετραγώνων. Παράγουν κυρίως γραμμικές προσεγγίσεις.

(2) **Radial basis function network** : οι παράμετροι αυτών των νευρωνικών δικτύων υπολογίζονται σε τρία βήματα. Α) οι κόμβοι βρίσκονται χρησιμοποιώντας αλγορίθμους ομαδοποίησης. Β) τα πλάτη βρίσκονται χρησιμοποιώντας τη μέθοδο του πλησιέστερου γείτονα. Γ) τα βάρη στο τρίτο στρώμα βρίσκονται ελαχιστοποιώντας το σύνολο των μέσων τετραγωνικών σφαλμάτων μεταξύ της εξόδου και των πραγματικών δεδομένων.

(3) **Elman recurrent network** : αυτό το είδος νευρωνικών δικτύων εισήχθει από τον ELMAN , ο οποίος πρόσθεσε επιπλέον νευρώνες εισόδου.

(4) **ANFIS models** : είναι ουσιαστικά το μοντέλο με το οποίο θα κάνουμε τη πρόβλεψη που θα ακολουθήσει παρακάτω. Πρόκειται για προσαρμοστικά νευρο-ασαφή δίκτυα που δουλεύουν χρησιμοποιώντας επαγωγικό τρόπο προσέγγισης των εκάστοτε συμπερασμάτων. Χρησιμοποιούν λογικούς κανόνες κατά τη λειτουργία τους ,τους οποίους έχει τη δυνατότητα να αλλάξει ο χρήστης ανάλογα με τις απαιτήσεις του.

(5) **Neural logic network (NLN)** : πρόκειται για μια εξελιγμένη μορφή νευρωνικών δικτύων τα οποία προσπαθούν να μιμηθούν ανθρώπινες συμπεριφορές όπως είναι η λογική και η διαίσθηση.

Στη συνέχεια θα προβούμε σε σύγκριση όλων αυτών των μεθόδων χρησιμοποιώντας το γνωστό μέτρο σύγκρισης της ρίζας του μέσου τετραγωνικού σφάλματος. Τα δεδομένα μας προήλθαν από το νησί της Κρήτης. Είναι δεδομένα που καλύπτουν ένα μήνα μετρήσεων(Μάρτιος 1996) και εκτιμούν τη μέση ταχύτητα του ανέμου ανά ώρα(σύνολο παρατηρήσεων 774ώρες) .Όπως θα παρατηρήσουμε η ομάδα πέντε των νευρωνικών δικτύων παρουσιάζει το μικρότερο σφάλμα.

Model	RMS (m s ⁻¹)
Persistent	1.2587
AR(2)	1.2383
ARIMA 1	1.2303
ARIMA 2	1.2607
Linear NN	1.2162
BP NN (back propagation)	1.2111
LM NN (Levenberg Marquardt)	1.1999
RBF (radial basis function)	1.2121
ELM (Elman algorithm)	1.2250
ANFIS	1.2062
NLN	1.2048
NLN+LR(logic rules)*	1.1971

Στη συνέχεια της επισκόπησης μας στις μεθόδους πρόβλεψης της ταχύτητας του ανέμου θα αναφερθούμε σε πραγματικές ενδιαφέρουσες μελέτες(case study) από όλο τον κόσμο έτσι ώστε να έχουμε την πρακτική διάσταση του προβλήματος που μας ενδιαφέρει.

Στην παρακάτω μελέτη χρησιμοποιήθηκαν δεδομένα πρόβλεψης ανά δέκα λεπτά έτσι ώστε να μπορέσουμε να πάρουμε τα αποτελέσματά μας με χρονικό ορίζοντα

πρόβλεψης μιας ώρας .Η μέθοδος που χρησιμοποιήθηκε είναι ο μέσος όρος ακολουθίας. Χρησιμοποιούνται δυο ομάδες δεδομένων οι οποίες παρουσιάζονται στο παρακάτω πίνακα με τη μορφή των δεδομένων πρόβλεψης και των πραγματικών δεδομένων. Η πρώτη ομάδα δεδομένων καλύπτει τη περίοδο ενός μήνα(Μάρτιος 1998) , στο νοτιότερο μέρος του νησιού της Κρήτης. Η δεύτερη ομάδα δεδομένων καλύπτει και αυτή τη περίοδο ενός μήνα(Ιούνιος 1997), έτσι ώστε τα μεγέθη να μπορούν να είναι άμεσα συγκρίσιμα.. Τα δεδομένα προέρχονται από τη Μεγάλη Βρετανία(Oxfordshire) και συγκεκριμένα από το εργαστήριο RAL (Rutherford Appleton Laboratory). Τα δεδομένα και των δυο ομάδων χωρίστηκαν σε δεδομένα πρόβλεψης και εκτίμησης. Παρακάτω φαίνεται η σύγκριση των δυο ομάδων δεδομένων.

Model	RMS Error (m/s)
Iterative	1.0173
Simultaneous	1.0538

Πίνακας 1 : αποτελέσματα που αναφέρονται στην πρώτη ομάδα δεδομένων με χρονικό ορίζοντα πρόβλεψης τη μια ώρα

Model	RMS Error (m/s)
Iterative	0.6648
Simultaneous	0.6650

Πίνακας 2: αποτελέσματα που αναφέρονται στη δεύτερη ομάδα δεδομένων με χρονικό ορίζοντα πρόβλεψης τη μια ώρα

Όπως φαίνεται και από το παρακάτω case study η πρόβλεψη ανέμου με νευρωνικά δίκτυα με δυο ομάδες δεδομένων και η μετέπειτα σύγκριση μεταξύ τους είναι μια προσφιλή τακτική μεταξύ των ερευνητών. Η πρώτη ομάδα δεδομένων έχει προκύψει από μια όχι τόσο καλή περιοχή από απόψεως καλού αιολικού δυναμικού. Η δεύτερη ομάδα δεδομένων έχει συλλεχθεί από ένα πραγματικό αιολικό πάρκο στη βόρεια Ισπανία, το οποίο χαρακτηρίζεται από υψηλές ταχύτητες ανέμου. Ο αλγόριθμος που πρόκειται να χρησιμοποιηθεί βασίζεται σε ένα από τα προηγούμενα μοντέλα που

αναπτύξαμε παραπάνω και συγκεκριμένα το μοντέλο feed-forward, το οποίο ενσωματώθηκε για τις ανάγκες της έρευνας μας στο hardware των ανεμογεννητριών. Έτσι αρχικά ορίζουμε τον τρόπο με τον οποίο θα πραγματοποιηθεί η μάθηση του μοντέλου μας καθώς και τη συνάρτηση μεταφοράς που το διατρέπει. Τελικά θα χρησιμοποιηθεί γραμμική συνάρτηση μεταφοράς που θα μετρά τις εισόδους από τους εκάστοτε νευρώνες και θα τους προσθέτει ανάλογα με τα βάρη που θα τους αποδοθούν. Συγκεκριμένα προτιμήθηκε η σιγμοειδής συνάρτηση μιας και είναι παραγωγίσιμη και το σημείο καμπής της ταυτίζεται με το σημείο καμπής παραγωγής ενέργειας των ανεμογεννητριών. Εδώ ως μέτρο ευαισθησίας για τη μέτρηση της ακρίβειας του αλγορίθμου χρησιμοποιούμε το τετραγωνικό σφάλμα. Στη συνέχεια δίνονται τα αποτελέσματα των δυο ομάδων δεδομένων καθώς και το παρατηρούμενο όπως και το επιτρεπτό σφάλμα.

<i>Test</i>	<i>training finished</i>	<i>permitted minimum</i>	<i>obtained error</i>
<i>Number</i>	<i>(cycle)</i>	<i>quadratic error</i>	
1	1008	0.005	0.102106
2	1368	0.004	0.086384
3	1040	0.003	0.071022
4	3574	0.002	0.046677
5	Training not finished	0.001	—
6	5927	0.00108	0.025341
7	10130	0.00105	0.025268

<i>Test</i>	<i>training finished</i>	<i>permitted minimum</i>	<i>obtained error</i>
<i>Number</i>	<i>(cycle)</i>	<i>quadratic error</i>	
1	1053	0.005	0.108065
2	1276	0.004	0.097511
3	1613	0.003	0.084982
4	2748	0.0025	0.06955
5	3841	0.0023	0.066133
4	13512	0.002	0.059594
5	45917	0.0019	0.057154
7	Training not finished	0.0015	—

Στο επόμενο case study επιχειρήθηκε να γίνει η πρόβλεψη της ταχύτητας του ανέμου σε μακροχρόνιο επίπεδο από μια ομάδα ελλήνων επιστημόνων. Το πείραμα έλαβε χώρα στην ανατολική Κρήτη κατά το διάστημα της πρώτης Απριλίου το 2000 έως 31 Δεκεμβρίου του 2000. Τα δεδομένα μας είναι η ωριαία ταχύτητα του ανέμου κατά το διάστημα αυτό και η ακρίβεια του αλγορίθμου μετράται μέσω του μέσου τετραγωνικού σφάλματος. Ο αλγόριθμος που αναπτύχθηκε περιείχε πολλά

πρωτοποριακά στοιχεία όπως το γεγονός ότι χρησιμοποιούσαν on-line αλγορίθμους μάθησης και εκπαίδευσης του μοντέλου και το μοντέλο ανήκε ουσιαστικά στην ομάδα των RPE (περιοδικά επαναλαμβανόμενο σφάλμα) αλγορίθμων. Τελικά προτείνονται τρεις επιμέρους αλγόριθμοι μάθησης [1] DRPE algorithm (Decoupled RPE), 2) NLM-RPE (Neuron Linear Model RPE), 3) NNM-RPE (Neuron Non-linear Model RPE)], οι οποίοι χωρίζουν το συνολικό πρόβλημα σε υπό-προβλήματα, μειώνοντας έτσι τη πολυπλοκότητα του αρχικού αλγορίθμου. Ο χρονικός ορίζοντας της πρόβλεψης του ανέμου μπορεί να είναι από 2-3 μέρες έως και πολύ περισσότερο.

Προσπάθειες για πρόβλεψη της ταχύτητας του ανέμου με τη δημιουργία μοντέλων πρόβλεψης βλέπουμε κι σε αναπτυσσόμενες χώρες όπως η Ινδία. Σε αυτή τη μελέτη που θα αναφερθούμε γίνεται χρήση νευρωνικών δικτύων για τη πρόβλεψη του ανέμου σε ημερήσιο, εβδομαδιαίο και μηνιαίο επίπεδο σε δυο παραθαλάσσιες περιοχές αυτής της χώρας. Χρησιμοποιούνται κυρίως τα νευρωνικά δίκτυα της μορφής feed forward καθώς κι recurrent. Γίνεται χρήση χρονολογικών σειρών για την εκπαίδευση του μοντέλου καθώς και σύγκριση των νευρωνικών δικτύων με παραδοσιακές μεθόδους πρόβλεψης, δείχνοντας για μια ακόμη φορά τη υπεροχή των πρώτων έναντι των δεύτερων. Τέλος όσο αναφορά τα δεδομένα αυτά λαμβάνονταν σε ημερήσια βάση για δώδεκα χρόνια από το 1989-2000 από τις δυο παράκτιες περιοχές και τελικά συγκεντρώθηκαν σε μια ενιαία βάση δεδομένων από τη μετεωρολογική υπηρεσία της Ινδίας (IMD). 1) *μηνιαίες προβλέψεις* : τα δεδομένα των πρώτων δέκα χρόνων χρησιμοποιήθηκαν για την εκπαίδευση του μοντέλου και τα υπόλοιπα δυο για την αξιολόγησή του. Παρόμοια αποτελέσματα έδωσαν και οι εβδομαδιαίες όσο και οι ημερήσιες προβλέψεις.

Forecast	Scheme used	Network type	Training algorithm	Mean % error
Month	Neural network	Feed forward	BP	4.7
			CC	4.5
			JE	14.3
	Time series	ARIMA	ARIMA (2,12,2)	5.9
Week	Neural network	Feed forward	BP	6.0
			CC	5.4
			JE	6.1
	Time series	ARIMA	ARIMA (1,52,1)	8.6
Daily	Neural network	Feed forward	BP	7.0
			CC	6.3
			JE	7.5
	Time series	ARIMA	ARIMA (2,365,2)	11.5

Σημείωση : BP=back-propagation; CC=cascade correlation; JE=Jordan Elman; ARIMA=auto-regressive integrated moving average; r=correlation coefficient.

Ακόμα μια ενδιαφέρουσα μελέτη έγινε στη γειτονική Τουρκία με δεδομένα ανέμου των τελευταίων πενήντα χρόνων, τα οποία ελήφθησαν από τη μετεωρολογική τουρκική υπηρεσία (WTMS). Και εδώ χρησιμοποιήθηκε το μέσο σφάλμα το οποίο ποίκιλε από 3%-6%. Επίσης χρησιμοποιήθηκε και ένα έτοιμο πληροφοριακό σύστημα από τη Δανία που χρησιμοποιείται στα μετεωρολογικά τους εργαστήρια. Για αυτά τα πληροφοριακά συστήματα τα οποία είναι γνωστά ως συστήματα διαχείρισης ενέργειας θα μιλήσουμε εκτενώς παρακάτω. Αυτό που πρέπει να πούμε

για το σύστημα που αναφέρθηκε είναι ότι έδωσε καλά αποτελέσματα, χρησιμοποιώντας διαφορετικούς συντελεστές τραχύτητας σε κάθε επίπεδο από τα 10-200 μέτρα υψόμετρο.

Επιπλέον μια ερευνητική ομάδα από το Αριστοτέλειο πανεπιστήμιο πραγματοποίησε μετρήσεις του αιολικού δυναμικού στο νότιο και κεντρικό Αιγαίο πέλαγος με σκοπό τη βραχυχρόνια πρόβλεψη ανέμου σε χρονικό ορίζοντα 10 λεπτών έως μιας ώρας. Τα δεδομένα συλλέγονταν για επτά χρόνια ώστε να έχουμε όσο το δυνατόν καλύτερη πρόβλεψη. Μάλιστα πρέπει να σημειωθεί ότι το νότιο Αιγαίο σύμφωνα με διεθνής μελέτες έχει αποδειχθεί ότι έχει από τα καλύτερα αιολικά δυναμικά. Τέλος τα όργανα που χρησιμοποιήθηκαν ήταν ένα ανεμόμετρο ένα ποτενσιόμετρο και αισθητήρες πίεσης και θερμοκρασίας. Οι διάφορες ταχύτητες του ανέμου που μετρήθηκαν κατά μέσο όρο ήταν η είσοδος του μοντέλου μας.

Τελευταίο case study που αξίζει να σημειωθεί είναι αυτό το οποίο πραγματοποιήθηκε στην Εύβοια τη περίοδο 2005-2006. Το νευρο-ασαφές μοντέλο που χρησιμοποιήθηκε είναι γνωστό ως το μοντέλο ANFIS (adapted neuro-fuzzy inference system) . Το μοντέλο είχε τέσσερις εισόδους και έξοδο την τιμή της ενέργειας που παράγεται. Το 82% των δεδομένων χρησιμοποιήθηκε για την εκπαίδευση του μοντέλου ενώ το υπόλοιπο ποσοστό για την αξιολόγησή του. Επιπλέον χρησιμοποιήθηκε η μέθοδος δοκιμής και σφάλματος για να βρεθεί το είδος της βέλτιστης συνάρτησης συμμετοχής, η οποία βελτιστοποιεί το σφάλμα που προκύπτει. Έτσι επιλέχθηκαν δυο συναρτήσεις συμμετοχής με καμπανοειδή μορφή. Οι επαναλήψεις στις οποίες αξιολογήθηκε το μοντέλο είναι 125. Αυτό προέκυψε έπειτα από διάφορες δοκιμές. Η ανάλυση της ευαισθησίας του μοντέλου έγινε βάση τεσσάρων σφαλμάτων RMSE, MSE, MAE, MAPE (ρίζα μέσου τετραγωνικού σφάλματος, μέσο τετραγωνικό σφάλμα, μέσο απόλυτο σφάλμα, μέσο απόλυτο πολυωνυμικό σφάλμα. Η σύγκριση των αποτελεσμάτων φαίνεται στον παρακάτω πίνακα.

<i>Epochs</i>	<i>MSE</i>	<i>RMSE</i>	<i>MAE</i>	<i>MAPE</i>
10	6.78E+03	82.3103	48.6556	5.8002
20	6.17E+03	78.5419	47.1426	2.8315
25	7.06E+03	84.0425	49.5851	4.2921
50	5.80E+03	76.1339	47.192	4.1167
100	5.28E+03	72.6737	45.6944	4.2166
125	5.27E+03	72.5876	45.144	4.5771
250	5.88E+03	76.6534	48.4579	4.3886
500	6.20E+03	78.7459	49.5796	4.9447

1.3 Πληροφοριακά συστήματα πρόβλεψης ταχύτητας του ανέμου

Στο σημείο αυτό πρόκειται να ασχοληθούμε με διάφορα πληροφοριακά συστήματα με τα οποία μπορούμε να κάνουμε απευθείας πρόβλεψης της ταχύτητας του ανέμου εισάγοντας μόνο τα δεδομένα μας. Τα πληροφοριακά αυτά συστήματα χωρίζονται σε διάφορες κατηγορίες εξετάζοντας κάποια κριτήρια όπως :

Α) το είδος των δεδομένων που εφαρμόζονται π.χ χρονολογικές σειρές κ.ο.κ

- B) το συγκεκριμένο διάστημα και την κλίμακα που καλύπτουν τα δεδομένα π.χ αιολικά πάρκα, μεμονωμένες ανεμογεννήτριες κ.ο.κ
- Γ) τα γενικότερα συστήματα διαχείρισης της ενέργειας που υπάρχουν.

Παρακάτω αναφέρουμε τα βασικότερα από αυτά τα οποία γνωρίζουμε ότι έχουν ευρεία εφαρμογή στον χώρο της Ευρώπης.

Το πρώτο βασικό μοντέλο που εξετάζουμε είναι ένα μοντέλο που αναπτύχθηκε στο πανεπιστήμιο του Oldenburg. Ονομάστηκε Previento. Για τη δημιουργία του χρησιμοποιήσαν ένα από τα γνωστά μοντέλα πρόβλεψης που υπάρχουν στη γερμανική μετεωρολογική υπηρεσία της Γερμανίας καθώς και το γνωστό σε όλους στο χώρο της Ευρώπης μοντέλο NWP(numerical weather prediction), το οποίο προσομοιώνει τις ατμοσφαιρικές διεργασίες σε ένα υπολογιστή με σκοπό τη μελλοντική εξέλιξη της κατάστασης της ατμόσφαιρας βασισμένης σε πραγματικά δεδομένα που λαμβάνονται σε τακτά χρονικά διαστήματα. Ο ερευνητής Monnich έκανε μια πολύ σοβαρή επισκόπηση των παραμέτρων και των μοντέλων που χρησιμοποιούνται σε αυτές τις περιπτώσεις και με αυτό τον τρόπο μπόρεσε να βελτιστοποιήσει τα αποτελέσματα της πρόβλεψης στη περίπτωση που έχουμε βραχυχρόνιο ορίζοντα πρόβλεψης. Τελικά από όλες τις ομάδες μοντέλων και παραμέτρων που εξέτασε επέλεξε εκείνη που μοντελοποιεί την ατμοσφαιρική σταθερότητα. Επίσης από την συγκεκριμένη ερευνητική ομάδα προτάθηκε στο μοντέλο της NWP να χρησιμοποιηθεί μια επαναληπτική τεχνική έτσι ώστε να έχουμε καλύτερα αποτελέσματα. Επίσης διεξοδικά εξετάστηκε και η καμπύλη ισχύος, γιατί όπως αποδείχθηκε η θεωρητική καμπύλη ισχύος που δινόταν από τους κατασκευαστές και η τελική πραγματική καμπύλη ισχύος μπορούσαν να διαφέρουν σημαντικά. Ακόμα και η πραγματική καμπύλη ισχύος μπορούσε να διαφέρει από την αρχική εκτίμησης της καθώς περνούσαν τα χρόνια. Αυτό συνέβαινε λόγω των επισκευών που γίνονταν ανά τακτά χρονικά διαστήματα στις τουρμπίνες των ανεμογεννητριών. Όπως αποδείχθηκε αργότερα τέτοιου είδους σφάλματα εμφανίζονταν πολύ συχνά στο μοντέλο NWP.

LocalPred και RegioPred είναι δυο άλλα μοντέλα πληροφοριακών συστημάτων που αναπτύχθηκαν από τον Martí Perez. Το διαφορετικό σε αυτή την οικογένεια των μοντέλων είναι ότι περιλαμβάνουν τη προσαρμοστική βελτιστοποίηση της εισόδου του μοντέλου NWP, μοντελοποίηση χρονοσειρών και μοντελοποίηση καμπύλων ισχύος.

Στη συνέχεια έχουμε ένα άλλο μοντέλο που αναπτύχθηκε από τον Jørgensen *et al* και ονομάστηκε HIRPOM (HIRlam POver prediction Model). Αυτός επικεντρώθηκε στην εξομάλυνση που χρειαζόταν το μοντέλο NWP για να δώσει καλύτερα αποτελέσματα πρόβλεψης. Η εξομάλυνση του μοντέλου έγινε σε διάφορα επίπεδα κι δοκιμάστηκε σε πολλά δεδομένα όπως 30χμ, 15χμ, 5χμ και 1.4χμ για δυο μήνες τον Ιανουάριο του 2001. Τελικά παρατηρήθηκε ότι καθώς αυξάνουμε την εξομάλυνση πετυχαίνουνε ελάχιστη περισσότερη ακρίβεια στα αποτελέσματα της πρόβλεψης σε σχέση με αυτά που θα περιμέναμε να πετύχουμε. Παρόλα αυτά οι υψηλότερες ταχύτητες ανέμου παρατηρήθηκαν σε υψηλά επίπεδα εξομάλυνσης. Επιπλέον για τις υψηλότερες τιμές εξομάλυνσης αποδείχθηκε ότι είναι καλύτερο το μοντέλο στρώματος κοντά στο έδαφος παρά σε μια πολύ τραχεία επιφάνεια. Έτσι ο ερευνητής κατέληξε και σε άλλα χρήσιμα επιμέρους συμπεράσματα τα οποία βοήθησαν αρκετά στο να βελτιωθεί το NWP μοντέλο. Για τη πειραματική επαλήθευση των συμπερασμάτων του άντλησε δεδομένα από 25 άσχημες μέρες από πλευράς πρόβλεψης σε διάστημα δυο μηνών από την δανέζικη μετεωρολογική υπηρεσία.

Εκτός από τα αναφερθέντα μοντέλα υπάρχουν και άλλα που χρησιμοποιούνται. Ήδη από το 1990 ο Landberg δημιούργησε ένα μοντέλο πρόβλεψης της ταχύτητας του ανέμου με βραχυχρόνιο ορίζοντα βασισμένο σε φυσικά μεγέθη όπως και ο ευρωπαϊκός Άτλας ανέμου. Βρήκε επίσης ότι χρειαζόταν τουλάχιστον 4 μηνών δεδομένα για να έχει μια σχετικά καλή ακρίβεια στα αποτελέσματα της πρόβλεψης. Επίσης εάν είμαστε σε ένα υψηλό επίπεδο ατμοσφαιρικής πίεσης όπως το εννοεί φυσικά το μοντέλο NWP τότε ακολουθείται η παρακάτω διαδικασία : από την ταχύτητα του ανέμου που παράγεται λόγω περιστροφής της γης και την τοπική τραχύτητα του εδάφους μπορούμε να υπολογίσουμε την ταχύτητα αντίστασης του ανέμου. Μετά χρησιμοποιούμε το προφίλ λογαριθμικής κλίμακας με τη τοπική τραχύτητα και αν βρούμε το ύψος της κλίμακας γύρω στα δέκα μέτρα τότε μπορούμε να είμαστε σίγουροι για την ορθότητα της μεθόδου μας.

Ένα πολύ ενδιαφέρον επίσης μοντέλο που έχει αναπτυχθεί είναι το WPPT (Wind Power Prediction Tool). Αναπτύχθηκε από το ινστιτούτο μοντελοποίησης πληροφορικής και μαθηματικών στο πολυτεχνείο της Δανίας. Το μοντέλο αυτό χρησιμοποιήθηκε στη δυτική Δανία το 1994 και στο ανατολικό της κομμάτι το 1999. Αρχικά χρησιμοποιούσαν την εκτίμηση ελαχίστων τετραγώνων με περιοδική προσαρμογή με ορίζοντα πρόβλεψης από 0.5-36 ώρες (βραχυχρόνιος χρονικός ορίζοντας πρόβλεψης). Εξαιτίας όμως της κακής ποιότητας αποτελεσμάτων που είχε το μοντέλο σε αυτό το χρονικό διάστημα χρησιμοποιήθηκε κυρίως για χρονικό ορίζοντα πρόβλεψης πάνω από 12 ώρες. Το WPPT είναι ουσιαστικά ένα σύστημα μοντελοποίησης της πρόβλεψης της συνολικής παραγωγής ισχύος του ανέμου από συγκεκριμένα επιλεγμένα αιολικά πάρκα, on-line μετρήσεις ισχύος από όλες τις ανεμογεννήτριες της περιοχής και αριθμητικές προβλέψεις καιρού της ταχύτητας και της κατεύθυνσης του ανέμου. Εάν κριθεί απαραίτητο η συνολική περιοχή χωρίζεται σε επιμέρους. Ο τρόπος πρόβλεψης της συνολικής περιοχής γίνεται τελικά με μια προσέγγιση δυο κλάδων. Αρχικά η πρόβλεψη της ισχύος του ανέμου γίνεται με on-line μετρήσεις της παραγωγής ισχύος από τις επιλεγθείσες τοποθεσίες. Σαν είσοδο του μοντέλου μας έχουμε τις αριθμητικές προβλέψεις του καιρού. Σε μια δεύτερη φάση μαζεύουμε όλα τα αποτελέσματα των επιμέρους προβλέψεων για κάθε τοποθεσία και τα ενοποιούμε σε ένα κοινό τόπο(σταθμισμένος μέσος όρος του συνόλου των προβλέψεων). Βασικό επίσης στοιχείο είναι ότι στο συγκεκριμένο πληροφοριακό σύστημα κάνουμε χρήση στατιστικών μοντέλων για καλύτερα αποτελέσματα. Τέλος αποδείχθηκε ότι τα πιο πρόσφατα παραμετρικά μοντέλα έδωσαν καλύτερα αποτελέσματα σε σχέση με άλλα που εφαρμόζονταν. Μιλώντας κάποιος για παραμετρικά μοντέλα εννοεί όλα εκείνα τα συστήματα που παρουσιάζουν μια μη-γραμμική συμπεριφορά αλλά τελικά μοντελοποιούνται ως γραμμικά, αντικαθιστώντας τις εκάστοτε μη γραμμικές παραμέτρους με διάφορες λείες συναρτήσεις οι οποίες αν και άγνωστες ποικίλουν σε πολύ μεγάλο βαθμό. Οι συναρτήσεις αυτές καλούνται συντελεστές συνάρτησης. Για on-line εφαρμογές είναι καλύτερο να χρησιμοποιούμε αυτούς τους συντελεστές ως δεδομένους για να έχουμε καλύτερα αποτελέσματα. Επιπλέον καθώς περνά ο καιρός και προσθέτουμε και άλλα δεδομένα θα πρέπει τα παλαιότερα να τα σταθμίζουμε έτσι ώστε να μην δημιουργηθεί κανενός είδους περιπλοκή. Έτσι επιτυγχάνουμε την χρονική προσαρμογή και την περιοδική αξιολόγηση των δεδομένων μας. Κάτι τέτοιο είναι πάρα πολύ σημαντικό γιατί ως γνωστόν τα δεδομένα μας μεταβάλλονται με τον καιρό αφού γίνονται διάφορες επισκευές στις ανεμογεννήτριες και όποτε είναι αναγκαίο προσθέτουμε ή αφαιρούμε από τα αιολικά πάρκα ανεμογεννήτριες ανάλογα με τις ανάγκες που έχουμε ως στόχο να καλύψουμε.

Επιπλέον οι ARMINES και RAL ανέπτυξαν μια πολύ σημαντική δουλειά όσο αναφορά τη βραχυχρόνια πρόβλεψη παραγωγής ενέργειας από τον άνεμο από το 1993. Ο χρονικός ορίζοντας της πρόβλεψης είναι 6-10 ώρες και γίνεται χρήση χρονοσειρών με την βοήθεια των οποίων προσπαθούμε να προβλέψουμε την έξοδο του αιολικού πάρκου. Τα μοντέλα τα οποία τελικά δημιουργήθηκαν, ενσωματώθηκαν στο λογισμικό EMS που δημιούργησε η ενεργειακή εταιρεία AMBER S.A και δοκιμάστηκε σε μια γραμμή παραγωγής στο νησί της Λήμνου. Στη συνέχεια πολλές άλλες παραλλαγές και προσεγγίσεις διαφόρων μοντέλων δοκιμάστηκαν όπως ARMA, neural networks of various types (backpropagation, RHONN etc), fuzzy neural networks, wavelet networks etc. Τελικά αποδείχθηκε ότι την καλύτερη απόδοση από όλα αυτά τα μοντέλα παρουσίασαν τα ασαφή νευρωνικά δίκτυα, γεγονός το οποίο πρόκειται να αποδείξουμε και στη συνέχεια.

Ακόμα το γερμανικό Ινστιτούτο ενέργειας ISET (Institut für Solare Energieversorgungstechnik) ασχολήθηκε από το 2000 με τη βραχυχρόνια πρόβλεψη της ταχύτητας του ανέμου, χρησιμοποιώντας παραλλαγές διάφορες νευρωνικών δικτύων. Οι παραπάνω ερευνητικές προτάσεις προέκυψαν από το ερευνητικό γερμανικό πρόγραμμα WMEP (Wissenschaftliches Mess- und EvaluierungsProgramm), το οποίο ασχολήθηκε σε πάρα πολύ μεγάλο βαθμό με τη αιολική ενέργεια. Μάλιστα ο πρώτος τους βασικός πελάτης υπήρξε η ευρωπαϊκή ένωση, η οποία παρατήρησε ότι υπήρχε έλλειψη παραγωγής ενέργειας από τον άνεμο και για αυτό τον λόγο ήθελε να δημιουργήσει ισχυρά εργαλεία διαχείρισης και πρόβλεψής του. Το τελικό μοντέλο που αναπτύχθηκε ονομάστηκε Advanced Wind Power Prediction Tool (AWPT).

Επίσης ένα πολύ καλό εργαλείο πρόβλεψης του ανέμου υπήρξε το αμερικάνικο EWind το οποίο κατασκευάστηκε από την True Wind, Inc. Το συγκεκριμένο μοντέλο δεν χρησιμοποιούσε μερική ή ολική παραμετροποίηση των τοπικών επιδράσεων αλλά το χρησιμοποιούσαν ως μέσης κλίμακας μοντέλο με άνω και κάτω φράγματα με δεδομένα από ένα τοπικό μοντέλο. Με αυτό τον τρόπο εφαρμόζονταν πιο φυσικές διεργασίες και οι προβλέψεις έδιναν καλύτερα αποτελέσματα. Μάλιστα πριν τη χρήση αυτού του μοντέλου με την ίδια λογική επεξεργασία δεδομένων χρησιμοποιούσαν το μοντέλο MASS (Mesoscale Atmospheric Simulation System). Παρόμοια δουλεύουν και τα μοντέλα ForeWind, MM5, WRF, COAMPS, workstation-ETA and OMEGA. Για να ελαχιστοποιήσουν όσο το δυνατόν περισσότερο τα συστηματικά σφάλματα αυτά τα μοντέλα χρησιμοποιούν προσαρμοστικές στατιστικές μέτρησης των σφαλμάτων ή παλινδρομικά γραμμικά μοντέλα ή μπευνζιανά νευρωνικά δίκτυα. Ο ορίζοντας πρόβλεψης είναι 48 ώρες. Στην Καλιφόρνια και στην Πενσυλβανία που χρησιμοποιήθηκαν είχαν 48% καλύτερο RMSE (ρίζα μέσου τετραγωνικού σφάλματος). Το μοντέλο εφαρμόστηκε σε δυο μεγάλα αιολικά πάρκα. Το πρώτο με 900 ανεμογεννήτριες και απόδοση 90 MW στο Altamont Pass και το άλλο 111 ανεμογεννήτριες απόδοσης 66.6 MW στο San Gorgonio Pass. Τέλος παρουσίασαν μέσο απόλυτο σφάλμα τις πρώτες 28 μέρες λειτουργίας 10.8% ενώ τις υπόλοιπες 11.7%.

Τελευταίο μοντέλο που πρόκειται να εξετάσουμε είναι αυτό που ανέπτυξε η ισπανική ενεργειακή εταιρεία Red Eléctrica de España (the Spanish TSO). Το μοντέλο ονομάστηκε Sipleólico και αναπτύχθηκε στο πανεπιστήμιο University Carlos III of Madrid. Ουσιαστικά είναι μια οικογένεια από 9 διαφορετικά μοντέλα τα οποία εφαρμόζονται ανάλογα με τη διαθεσιμότητα των δεδομένων μας. Βασίζονται κυρίως σε εισόδους που αποτελούνται από χρονολογικές σειρές και όχι απλά σε αριθμητικές μετεωρολογικές προβλέψεις. Στα πρώτα τρία μοντέλα λαμβάνονται μεγαλύτεροι περίοδοι πρόβλεψης ενώ στα υπόλοιπα τρία χρησιμοποιείται ως

δεδομένο και η κατεύθυνση κάθε φορά του ανέμου. Τα υπόλοιπα είναι συνδυασμός των ανωτέρω με τη μόνη διαφορά ότι συμπεριλαμβάνουν μη παραμετρικούς όρους στο σύνολο της περιόδου πρόβλεψης. Η αξιολόγηση των μοντέλων γίνεται με τους εξής αλγορίθμους : a Recursive Least Squares (RLS) algorithm or a Kalman Filter. Συμπερασματικά, μπορούμε να σημειώσουμε ότι ο συνδυασμός αυτών των μοντέλων βοηθά στη βέλτιστη απόδοση της πρόβλεψης του ανέμου τόσο σε πρακτικό όσο και σε θεωρητικό επίπεδο με σκοπό τη μελλοντική εξέλιξη τους για πιο ακριβή αποτελέσματα.

ΚΕΦΑΛΑΙΟ 2: ΠΑΡΟΥΣΙΑΣΗ ΤΩΝ ΜΟΝΤΕΛΩΝ AR, ARMA ΚΑΙ ANFIS

2.1 Μοντέλο AR

Ένα στοχαστικό μοντέλο που μπορεί να είναι εξαιρετικά χρήσιμο στην αναπαράσταση ορισμένων πρακτικά εμφανιζόμενων σειρών είναι το αποκαλούμενο αυτοπαλινδρόμενο μοντέλο. Στο μοντέλο αυτό, η τρέχουσα τιμή της διαδικασίας εκφράζεται σαν ένα πεπερασμένο, γραμμικό άθροισμα των προηγούμενων τιμών της διαδικασίας και μια διαταραχή a_t . Ας θεωρήσουμε τις τιμές της διαδικασίας σε ίσα χρονικά διαστήματα $t, t-1, t-2, \dots$ ως $z_t, z_{t-1}, z_{t-2}, \dots$. Επίσης έστω ότι $\tilde{z}_t, \tilde{z}_{t-1}, \tilde{z}_{t-2}, \dots$ είναι αποκλίσεις από το m (για παράδειγμα $\tilde{z}_t = z_t - m$). Τότε το

$$\tilde{z}_t = f_1 \tilde{z}_{t-1} + f_2 \tilde{z}_{t-2} + \dots + f_r \tilde{z}_{t-r} + a_t \quad (2.1.1)$$

ονομάζεται αυτοπαλινδρόμενη (AR) διαδικασία τάξης p . Ο λόγος για το όνομα αυτό είναι ότι ένα γραμμικό μοντέλο

$$\tilde{z} = f_1 \tilde{x}_1 + f_2 \tilde{x}_2 + \dots + f_r \tilde{x}_r + a$$

που σχετίζει μια εξαρτημένη μεταβλητή z με ένα σετ ανεξάρτητων μεταβλητών x_1, x_2, \dots, x_r , συν ένα σφάλμα a , αναφέρεται συχνά σαν ένα παλινδρόμενο μοντέλο, και το z λέγεται ότι παλινδρομείται στα x_1, x_2, \dots, x_r . Στην (2.1.1) η μεταβλητή z παλινδρομείται σε προηγούμενες τιμές του εαυτού της, συνεπώς το μοντέλο είναι αυτοπαλινδρόμενο. Αν ορίσουμε ένα αυτόπαλινδρόμενο τελεστή τάξης p ως

$$f(B) = 1 - f_1 B - f_2 B^2 - \dots - f_r B^r$$

τότε το αυτοπαλινδρόμενο μοντέλο μπορεί να γραφτεί οικονομικά ως

$$f(B)\tilde{z}_t = a_t$$

Το μοντέλο περιλαμβάνει $p+2$ άγνωστες παραμέτρους $m, f_1, f_2, \dots, f_r, S_a^2$ που στην πράξη πρέπει να υπολογιστούν από τα δεδομένα. Η επιπλέον παράμετρος S_a^2 είναι η διασπορά της διαδικασίας λευκού θορύβου a_t .

Δεν είναι δύσκολο να δούμε ότι το αυτοπαλινδρόμενο μοντέλο είναι μια ειδική περίπτωση του γραμμικού μοντέλου φίλτρου (linear filter model) του (2.1.1). Για παράδειγμα μπορούμε να απαλείψουμε τον \tilde{z}_{t-1} από το δεξιό μέρος της (2.1.1) μετασχηματίζοντάς τη ως εξής

$$\tilde{z}_{t-1} = f_1 \tilde{z}_{t-2} + f_2 \tilde{z}_{t-3} + \dots + f_r \tilde{z}_{t-r-1} + a_{t-1}$$

Ομοίως μπορούμε να μετασχηματίσουμε για τον \tilde{z}_{t-2} κλπ.

Συμβολικά έχουμε ότι το

$$f(B)\tilde{z}_t = a_t$$

είναι ισοδύναμο με το

$$\tilde{z}_t = y(B)a_t$$

με

$$y(B) = f^{-1}(B)$$

Οι αυτοπαλινδρόμενες διαδικασίες μπορεί να είναι στάσιμες και μη στάσιμες. Για να είναι στάσιμες πρέπει τα ϕ να επιλέγονται έτσι ώστε τα βάρη y_1, y_2, \dots στην $y(B) = f^{-1}(B)$ να σχηματίζουν συγκλίνουσα σειρά.

2.2 Μοντέλο ARMA

Για να επιτύχουμε μεγαλύτερη ευκαμψία στην προσαρμογή κανονικών χρονοσειρών, είναι μερικές φορές καλύτερο να χρησιμοποιούμε όρους και

αυτοπαλινδρόμενους και κινούμενου μέσου στο μοντέλο. Αυτό οδηγεί στο μοντέλο ARMA (mixed autoregressive-moving average)

$$\tilde{z}_t = f_1 \tilde{z}_{t-1} + f_2 \tilde{z}_{t-2} + \dots + f_p \tilde{z}_{t-p} + a_t - q_1 a_{t-1} - q_2 a_{t-2} - \dots - q_q a_{t-q} \quad (2.2.1)$$

ή

$$f(B)\tilde{z}_t = q(B)a_t$$

που έχει $p + q + 2$ άγνωστες παραμέτρους $m; f_1, \dots, f_p; q_1, \dots, q_q; S_a^2$, οι οποίες υπολογίζονται από τα δεδομένα.

Στην πράξη είναι συνήθως αλήθεια ότι επαρκής αναπαράσταση των πραγματικά εμφανιζόμενων στάσιμων χρονοσειρών μπορεί να παρθεί με αυτοπαλινδρόμενα, κινούμενου μέσου, ή μεικτών μοντέλων, στα οποία τα p και q δεν είναι μεγαλύτερα από 2 και συχνά είναι μικρότερα από 2.

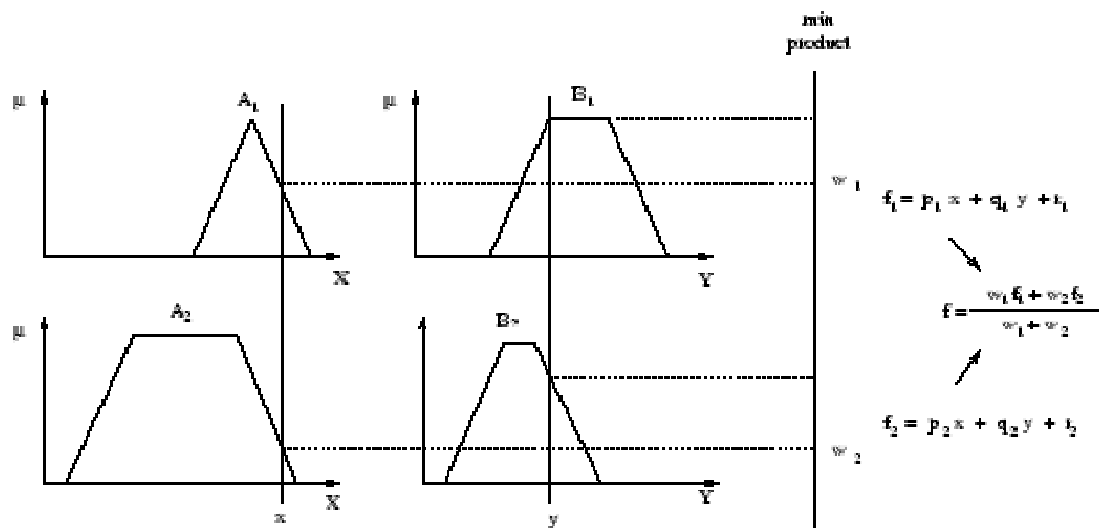
2.3 ANFIS

Ένα νευρο-ασαφές σύστημα ορίζεται ως ένας συνδυασμός τεχνητών νευρωνικών δικτύων (ANN) και Ασαφούς Συστήματος Συμπερασμού (Fuzzy Inference System, FIS) κατά τέτοιο τρόπο ώστε ο αλγόριθμος εκμάθησης νευρωνικών δικτύων χρησιμοποιείται για να καθορίσει τις παραμέτρους του FIS. Το Προσαρμοστικό Νευροασαφές σύστημα συμπερασμού (Adaptive Neural Fuzzy Inference System, ANFIS) είναι ένα σύστημα που ανήκει στην νευρο-ασαφή κατηγορία.

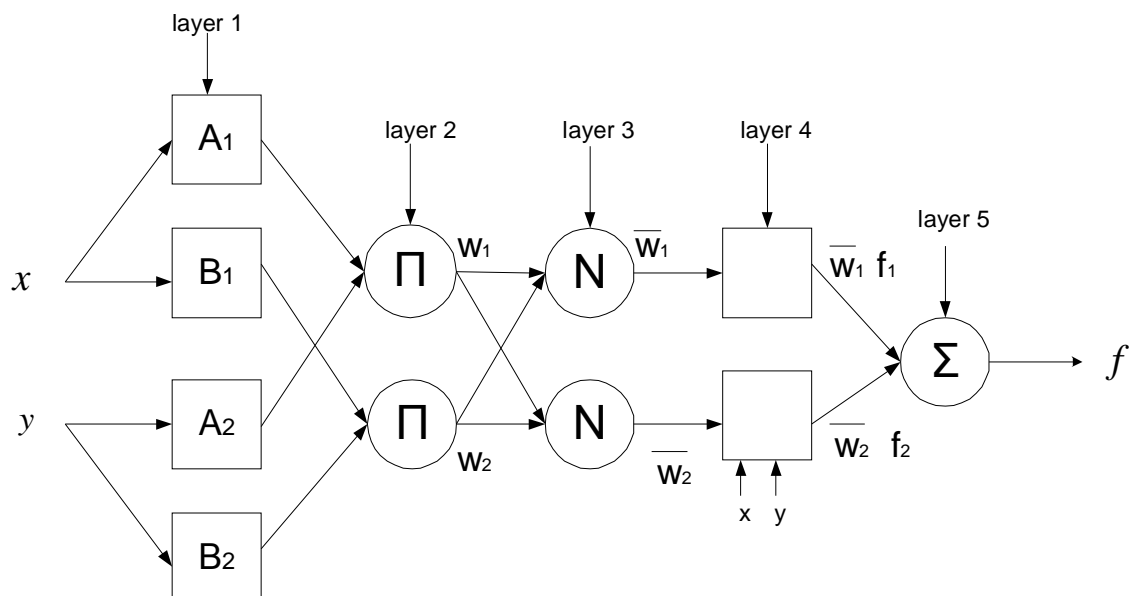
Λειτουργικά, δεν υπάρχει σχεδόν κανένας περιορισμός στις συναρτήσεις κόμβων ενός προσαρμόσιμου (adaptive) δικτύου εκτός από τμήματα που μπορούν να διαφοριστούν (piecewise differentiability). Δομικά, ο μόνος περιορισμός της διαμόρφωσης δικτύων είναι ότι πρέπει να είναι τύπου πρόσθιας τροφοδότησης. Λόγω αυτού του μικρού περιορισμού, οι εφαρμογές του προσαρμόσιμου δικτύου είναι

άμεσες και απέραντες σε διάφορους τομείς. Σε αυτό το τμήμα, προτείνουμε μια κατηγορία προσαρμόσιμων δικτύων, τα οποία είναι λειτουργικά ισοδύναμα με τα ασαφή συστήματα συμπερασμού.

Ο ασαφής μηχανισμός συμπερασμού παρουσιάζεται στο επόμενο σχήμα:



Σχήμα 1: Ασαφής μηχανισμός συμπερασμού



Σχήμα 2: Αρχιτεκτονική του ANFIS

Για ευκολία, υποθέτουμε ότι το ασαφές σύστημα συμπερασμού υπό εξέταση έχει δύο εισόδους x και y και μία έξοδο z . Υποθέτουμε ότι η βάση κανόνων περιέχει δύο ασαφείς κανόνες τύπου Takagi και Sugeno:

Κανόνας 1:

$$\text{Αν } x \text{ είναι } A_1 \text{ και } y \text{ είναι } B_1 \text{ τότε } f_1 = p_1 \cdot x + q_1 \cdot y + r_1$$

Κανόνας 2:

$$\text{Αν } x \text{ είναι } A_2 \text{ και } y \text{ είναι } B_2 \text{ τότε } f_2 = p_2 \cdot x + q_2 \cdot y + r_2$$

Η αρχιτεκτονική του ANFIS παρουσιάζεται στο σχήμα 2. Οι συναρτήσεις κόμβων στο ίδιο επίπεδο είναι της ίδιας οικογένειας συναρτήσεων όπως περιγράφουμε παρακάτω:

Επίπεδο 1 Κάθε κόμβος i σε αυτό το επίπεδο είναι ένας προσαρμόσιμος κόμβος με μία συνάρτηση κόμβου.

$$O_i^1(x) = m_{A_i}(x)$$

όπου x – η είσοδος στον κόμβο i

A_i - η γλωσσική ταμπέλα (small, large, κλπ.) που σχετίζεται με αυτή τη συνάρτηση του κόμβου.

Με άλλα λόγια, το O_i^1 είναι ο βαθμός συμμετοχής του A_i και καθορίζει το βαθμό στον οποίο η είσοδος x ικανοποιεί τον ποσοτικοποιητή A_i . Συνήθως επιλέγεται το $m_{A_i}(x)$ σε σχήμα καμπάνας με το μέγιστο ίσο με 1 και το ελάχιστο ίσο με 0, όπως η γενικευμένη καμπανοειδής συνάρτηση

$$m_{A_i}(x) = \frac{1}{1 + \left[\left(\frac{x - c_i}{a_i} \right)^2 \right]^{b_i}}$$

ή η γκαουσιανή συνάρτηση

$$m_{A_i}(x) = e^{-\left(\frac{x-c_i}{a_i}\right)^2}$$

όπου a_i, b_i, c_i είναι το σύνολο παραμέτρων.

Καθώς οι τιμές αυτών των παραμέτρων αλλάζουν, οι συναρτήσεις ποικίλλουν ανάλογα, παρουσιάζοντας έτσι διάφορες μορφές της συνάρτησης συμμετοχής για τη γλωσσική ταμπέλα A_i . Οι παράμετροι σε αυτό το επίπεδο αναφέρονται ως *αρχικοί παράμετροι (premise parameters)*.

Επίπεδο 2 Κάθε κόμβος σε αυτό το επίπεδο είναι ένας σταθερός (fixed) κόμβος Π, του οποίου η έξοδος είναι το γινόμενο όλων των εισερχόμενων σημάτων:

Επίπεδο 3 Κάθε κόμβος σε αυτό το επίπεδο είναι ένας σταθερός κόμβος Ν. Ο *i-οστος* κόμβος υπολογίζει το λόγο του βαθμού ενεργοποίησης (firing strength) του *i-οστού* κανόνα στο άθροισμα των βαθμών ενεργοποίησης όλων των κανόνων:

$$\bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i=1,2$$

Για ευκολία, οι έξοδοι αυτού του επιπέδου ονομάζονται κανονικοποιημένοι βαθμοί ενεργοποίησης (normalized firing strengths).

Επίπεδο 4 Κάθε κόμβος i σε αυτό το επίπεδο είναι ένας προσαρμόσιμος κόμβος με μία συνάρτηση κόμβου.

$$O_{4,i} = \bar{w}_i * f_i = \bar{w}_i * (p_1 * x + q_i * y + r_i)$$

όπου: \bar{w}_i - η έξοδος του επιπέδου 3

$\{p_i, q_i, r_i\}$ - το σύνολο παραμέτρων. Οι παράμετροι σε αυτό το επίπεδο αναφέρονται ως *επακόλουθοι (consequent parameters)*.

Επίπεδο 5 Ο μοναδικός κόμβος σε αυτό το επίπεδο είναι ένας σταθερός κόμβος Σ που υπολογίζει τη συνολική έξοδο σαν το ολικό άθροισμα όλων των εισερχόμενων σημάτων:

$$overall\ output = O_{5,i} = \sum_i \bar{w}_i * f_i = \frac{\sum_i w_i * f_i}{\sum_i w_i}$$

ΚΕΦΑΛΑΙΟ 3: Μέθοδοι για προσδιορισμό συστήματος

3.1 Προσδιορισμός συστήματος: Εισαγωγή

Το πρόβλημα προσδιορισμού ενός μαθηματικού μοντέλου για ένα άγνωστο σύστημα (το οποίο αναφέρεται επίσης και ως σύστημα στόχος) παρατηρώντας τα ζευγάρια δεδομένων εισόδου-εξόδου του αναφέρεται γενικά ως αναγνώριση συστήματος. Οι στόχοι του προσδιορισμού συστήματος είναι πολλαπλοί:

- Για να προβλέψει μια συμπεριφορά του συστήματος, όπως στη πρόβλεψη χρονοσειρών και στη πρόβλεψη καιρού.
- Για να εξηγήσει τις αλληλεπιδράσεις και τις σχέσεις μεταξύ των εισόδων και των εξόδων ενός συστήματος. Για παράδειγμα, ένα μαθηματικό μοντέλο μπορεί να χρησιμοποιηθεί για να εξετάσει εάν η ζήτηση ποικίλλει πράγματι ανάλογα με την προσφορά σε ένα οικονομικό σύστημα.
- Για να σχεδιάσει έναν ελεγκτή βασισμένο στο μοντέλο ενός συστήματος, όπως στον έλεγχο αεροσκαφών και σκαφών. Επίσης για να κάνουμε την προσομοίωση υπολογιστών του συστήματος υπό έλεγχο, χρειαζόμαστε ένα μοντέλο του συστήματος.

Ο προσδιορισμός συστημάτων περιλαμβάνει γενικά δύο βήματα:

Προσδιορισμός δομής Σε αυτό το βήμα, πρέπει να εφαρμόσουμε την *a priori* γνώση για το σύστημα στόχο για να καθορίσουμε μια κατηγορία μοντέλων μέσα στην οποία

πρόκειται να διεξαχθεί η αναζήτηση του καταλληλότερου μοντέλου. Συνήθως αυτή η κατηγορία μοντέλων δείχνεται από μια παραμετρική συνάρτηση $y = f(\mathbf{u}; \boldsymbol{\theta})$, όπου y είναι η έξοδος του μοντέλου, \mathbf{u} είναι το διάνυσμα εισόδου και $\boldsymbol{\theta}$ είναι το διάνυσμα παραμέτρου. Ο προσδιορισμός της συνάρτησης f είναι εξαρτώμενο πρόβλημα και η συνάρτηση βασίζεται στην εμπειρία και τη διαίσθηση του σχεδιαστή και τους νόμους της φύσης που κυβερνούν το σύστημα στόχο.

Προσδιορισμός παραμέτρου Στο δεύτερο βήμα, η δομή του μοντέλου είναι γνωστή και το μόνο που χρειάζεται να κάνουμε είναι να εφαρμόσουμε τεχνικές βελτιστοποίησης για να καθορίσουμε το διάνυσμα παραμέτρου $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$ τέτοιο ώστε το προκύπτον μοντέλο $\hat{y} = f(\mathbf{u}; \hat{\boldsymbol{\theta}})$ να μπορεί να περιγράψει το σύστημα κατάλληλα.

Εάν δεν έχουμε οποιαδήποτε a priori γνώση για το σύστημα στόχο, τότε ο προσδιορισμός δομής γίνεται ένα δύσκολο πρόβλημα και πρέπει να επιλέξουμε τη δομή με δοκιμή και σφάλμα. Ευτυχώς, ξέρουμε πάρα πολλά για τις δομές των περισσότερων συστημάτων εφαρμοσμένης μηχανικής και βιομηχανικών διαδικασιών και έτσι είναι συνήθως δυνατό να παραχθεί μια συγκεκριμένη κατηγορία μοντέλων—συγκεκριμένα, μια παραμετρική συνάρτηση-που μπορεί καλύτερα να περιγράψει το σύστημα στόχο. Συνεπώς, το πρόβλημα προσδιορισμού συστημάτων περιορίζεται συνήθως σε αυτό του προσδιορισμού παραμέτρου.

Γενικά, ο προσδιορισμός συστημάτων δεν είναι μια διαδικασία ενός περάσματος αλλά πρέπει να κάνουμε προσδιορισμό δομής και παραμέτρου επανειλημμένα έως ότου να βρούμε ένα ικανοποιητικό πρότυπο ως εξής:

1. Καθορίζουμε και παραμετρικοποιούμε μια κατηγορία μαθηματικών μοντέλων που αντιπροσωπεύουν το σύστημα που προσδιορίζεται.
2. Εκτελούμε τον προσδιορισμό παραμέτρου για να επιλέξουμε τις παραμέτρους που ταιριάζουν καλύτερα στο σύνολο δεδομένων εκπαίδευσης.
3. Πραγματοποιούμε δοκιμές επικύρωσης για να δούμε εάν το μοντέλο που προσδιορίζεται ανταποκρίνεται σωστά σε ένα άγνωστο σύνολο δεδομένων. (Αυτό το σύνολο στοιχείων είναι χωρισμένο από το σύνολο δεδομένων

εκπαίδευσης και αναφέρεται ως **σύνολο δεδομένων δοκιμής, επικυρώσης ή ελέγχου.**)

4. Ολοκληρώνουμε τη διαδικασία μόλις τα αποτελέσματα της δοκιμής επικύρωσης είναι ικανοποιητικά. Διαφορετικά, μια άλλη κατηγορία μοντέλων επιλέγεται και τα βήματα 2 μέχρι 4 επαναλαμβάνονται.

Το ANFIS χρησιμοποιεί έναν υβριδικό αλγόριθμο εκμάθησης για να προσδιορίσει τις παραμέτρους των ασαφών συστημάτων τύπου Sugeno. Εφαρμόζει έναν συνδυασμό της μεθόδου ελαχίστων τετραγώνων (least-squares) και της μεθόδου οπισθόδρομης βαθμωτής ελαχιστοποίησης (backpropagation gradient descent) για την εκπαίδευση παραμέτρων των συναρτήσεων συμμετοχής τύπου FIS ώστε να μιμηθεί ένα δοσμένο σύνολο δεδομένων εκπαίδευσης. Στη συνέχεια, θα περιγράψουμε τη μέθοδο ελαχίστων τετραγώνων και τη μέθοδο οπισθόδρομης βαθμωτής ελαχιστοποίησης και έπειτα θα δούμε πως το ANFIS συνδυάζει αυτές τις δύο μεθόδους προκειμένου να βρεθούν γρηγορότερα και καλύτερα αποτελέσματα.

3.2 Εκτίμηση ελαχίστων τετραγώνων (Least-squares estimator)

Στο γενικό πρόβλημα ελαχίστων τετραγώνων, η έξοδος του γραμμικού μοντέλου y δίνεται από τη γραμμικά παραμετροποιημένη έκφραση

$$y = q_1 f_1(\mathbf{u}) + q_2 f_2(\mathbf{u}) + \dots + q_n f_n(\mathbf{u}), \quad (3.2.1)$$

όπου $\mathbf{u} = [u_1, \dots, u_p]^T$ είναι το διάνυσμα εισόδου του μοντέλου, f_1, \dots, f_n είναι γνωστές συναρτήσεις του \mathbf{u} , και q_1, \dots, q_n είναι άγνωστες παράμετροι που θα υπολογιστούν. Στη στατιστική, το να βάλεις δεδομένα χρησιμοποιώντας ένα γραμμικό μοντέλο ανφέρεται σαν γραμμική παλινδρόμηση. Κατά συνέπεια η εξίσωση (3.2.1) καλείται επίσης συνάρτηση παλινδρόμησης, και τα q_i ονομάζονται συντελεστές παλινδρόμησης.

Για να προσδιορίσουμε τις άγνωστες παραμέτρους q_i , συνήθως πρέπει να εκτελέσουμε πειράματα για να λάβουμε ένα σύνολο δεδομένων εκπαίδευσης που αποτελείται από τα ζευγάρια δεδομένων $\{(\mathbf{u}_i, y_i), i=1, \dots, m\}$ τα οποία αντιπροσωπεύουν τα επιθυμητά ζευγάρια εισόδου-εξόδου του συστήματος στόχου που θα μοντελοποιηθεί. Η αντικατάσταση κάθε ζευγαριού στοιχείων στην εξίσωση (3.2.1) παράγει ένα σύνολο γραμμικών εξισώσεων m :

$$\begin{aligned} f_1(\mathbf{u}_1)q_1 + f_2(\mathbf{u}_1)q_2 + \dots + f_n(\mathbf{u}_1)q_n &= y_1 \\ f_1(\mathbf{u}_2)q_1 + f_2(\mathbf{u}_2)q_2 + \dots + f_n(\mathbf{u}_2)q_n &= y_2 \\ &\mathbf{M} \\ f_1(\mathbf{u}_m)q_1 + f_2(\mathbf{u}_m)q_2 + \dots + f_n(\mathbf{u}_m)q_n &= y_m \end{aligned} \quad (3.2.2)$$

Χρησιμοποιώντας τη μορφή πινάκων, μπορούμε να ξαναγράψουμε τις προηγούμενες εξισώσεις σε μια συνοπτική μορφή:

$$\mathbf{A}\boldsymbol{\theta} = \mathbf{y} \quad (3.2.3)$$

όπου \mathbf{A} είναι ένας $m \times n$ πίνακας (μερικές φορές λέγεται πίνακας σχεδιασμού):

$$\mathbf{A} = \begin{bmatrix} f_1(\mathbf{u}_1) & \mathbf{L} & f_n(\mathbf{u}_1) \\ \mathbf{M} & \mathbf{M} & \mathbf{M} \\ f_1(\mathbf{u}_m) & \mathbf{L} & f_n(\mathbf{u}_m) \end{bmatrix},$$

$\boldsymbol{\theta}$ είναι ένα $n \times 1$ διάνυσμα άγνωστων παραμέτρων:

$$\boldsymbol{\theta} = \begin{bmatrix} q_1 \\ \mathbf{M} \\ q_n \end{bmatrix},$$

και \mathbf{y} είναι ένα $m \times 1$ διάνυσμα εξόδου:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \mathbf{M} \\ y_n \end{bmatrix}.$$

Η *i-οστή* σειρά του ενωμένου πίνακα δεδομένων $[\mathbf{A}\mathbf{y}]$, που δηλώνεται με $[\mathbf{a}_i^T \mathbf{y}_i]$, σχετίζεται με το *i-οστό* ζευγάρι δεδομένων $(\mathbf{u}_i; y_i)$ μέσω της

$$\mathbf{a}_i^T = [f_1(\mathbf{u}_i), \mathbf{K}, f_n(\mathbf{u}_i)].$$

Εφόσον το μεγαλύτερο μέρος των υπολογισμών μας βασίζεται στους πίνακες \mathbf{A} και \mathbf{y} , μερικές φορές αναφερόμαστε στο $(\mathbf{a}_i^T; y_i)$ σαν το *i-οστό* ζευγάρι δεδομένων του συνόλου δεδομένων εκπαίδευσης.

Για να προσδιορίσουμε μεμονωμένα το άγνωστο διάνυσμα $\boldsymbol{\theta}$, είναι απαραίτητο να ισχύει $m \geq n$. Αν ο \mathbf{A} είναι τετραγωνικός ($m = n$) και αντιστρέψιμος, τότε μπορούμε να λύσουμε ως προς τον άγνωστο \mathbf{x} από την εξίσωση (3.2.3) και έχουμε

$$\boldsymbol{\theta} = \mathbf{A}^{-1}\mathbf{y} \quad . \quad (3.2.4)$$

Εντούτοις, ο m είναι συνήθως μεγαλύτερος από τον n , δείχνοντας έτσι ότι έχουμε περισσότερα ζευγάρια στοιχείων από τις παραμέτρους. Σε αυτήν την περίπτωση, μια ακριβής λύση που να ικανοποιεί όλες τις m εξισώσεις δεν είναι πάντα δυνατή, δεδομένου ότι τα στοιχεία μπορεί να μολυνθούν από θόρυβο, ή το μοντέλο μπορεί να μην είναι κατάλληλο για την περιγραφή του συστήματος στόχου. Κατά συνέπεια η εξίσωση (3.2.3) πρέπει να τροποποιηθεί με την ενσωμάτωση ενός διανύσματος λάθους \mathbf{e} για να αποτελέσει το τυχαίο λάθος θορύβου ή το τυχαίο λάθος διαμόρφωσης ως εξής:

$$\mathbf{A}\boldsymbol{\theta} + \mathbf{e} = \mathbf{y} \quad . \quad (3.2.5)$$

Τώρα, αντί της εύρεσης της ακριβούς λύσης στην εξίσωση (3.2.3), θέλουμε να ψάξουμε για το $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$ που ελαχιστοποιεί το άθροισμα του τετραγωνικού σφάλματος και ορίζεται ως

$$E(\boldsymbol{\theta}) = \sum_{i=1}^m (y_i - \mathbf{a}_i^T \boldsymbol{\theta})^2 = \mathbf{e}^T \mathbf{e} = (\mathbf{y} - \mathbf{A}\boldsymbol{\theta})^T (\mathbf{y} - \mathbf{A}\boldsymbol{\theta}) \quad (3.2.6)$$

όπου $\mathbf{e} = \mathbf{y} - \mathbf{A}\boldsymbol{\theta}$ είναι το διάνυσμα λάθους που παράγεται από μια συγκεκριμένη επιλογή του $\boldsymbol{\theta}$. Σημειώστε ότι το $E(\boldsymbol{\theta})$ είναι σε τετραγωνική μορφή και έχει ένα μοναδικό ελάχιστο $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$. Το ακόλουθο θεώρημα δηλώνει έναν απαραίτητο όρο που ικανοποιείται από τον εκτιμητή ελαχίστων τετραγώνων $\hat{\boldsymbol{\theta}}$.

Θεώρημα 3.2.1 Εκτιμητής ελαχίστων τετραγώνων

Το τετραγωνικό σφάλμα στην εξίσωση (3.2.6) ελαχιστοποιείται όταν $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$, ο οποίος καλείται εκτιμητής ελαχίστων τετραγώνων (LSE για συντομία) και ο οποίος ικανοποιεί την κανονική εξίσωση

$$\mathbf{A}^T \mathbf{A} \hat{\boldsymbol{\theta}} = \mathbf{A}^T \mathbf{y} \quad (3.2.7)$$

Αν ο $\mathbf{A}^T \mathbf{A}$ είναι αντιστρέψιμος, ο $\hat{\boldsymbol{\theta}}$ είναι μοναδικός και δίνεται από την

$$\hat{\boldsymbol{\theta}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y} \quad (3.2.8)$$

Απόδειξη: Υπάρχουν διάφορες μέθοδοι διαθέσιμες στη βιβλιογραφία για την εύρεση του εκτιμητή ελαχίστων τετραγώνων για την εξίσωση (3.2.3). Μια απλή προσέγγιση είναι να τεθεί το παράγωγο του $E(\boldsymbol{\theta})$ ως προς $\boldsymbol{\theta}$ ίσο σε μηδέν. Σημειώνοντας ότι ο $\boldsymbol{\theta}^T \mathbf{A}^T \mathbf{y} = \mathbf{y}^T \mathbf{A} \boldsymbol{\theta}$ είναι βαθμωτός, μπορούμε να επεκταθούμε για $E(\boldsymbol{\theta})$:

$$E(\boldsymbol{\theta}) = (\mathbf{y}^T - \boldsymbol{\theta}^T \mathbf{A}^T)(\mathbf{y} - \mathbf{A}\boldsymbol{\theta}) = \boldsymbol{\theta}^T \mathbf{A}^T \mathbf{A} \boldsymbol{\theta} - 2\mathbf{y}^T \mathbf{A} \boldsymbol{\theta} + \mathbf{y}^T \mathbf{y} \quad (3.2.9)$$

Τότε η παράγωγος του $E(\boldsymbol{\theta})$ είναι

$$\frac{\partial E(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = 2\mathbf{A}^T \mathbf{A} \boldsymbol{\theta} - 2\mathbf{A}^T \mathbf{y}. \quad (3.2.10)$$

Θέτοντας $\frac{\partial E(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = 0$ στο $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$, παίρνουμε την κανονική εξίσωση

$$\mathbf{A}^T \mathbf{A} \hat{\boldsymbol{\theta}} = \mathbf{A}^T \mathbf{y}. \quad (3.2.11)$$

Αν ο $\mathbf{A}^T \mathbf{A}$ είναι αντιστρέψιμος, τότε ο $\hat{\boldsymbol{\theta}}$ μπορεί να επιλυθεί μοναδικά:

$$\hat{\boldsymbol{\theta}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y}. \quad (3.2.12)$$

Εφόσον το $E(\boldsymbol{\theta})$ είναι μια τετραγωνική συνάρτηση $\boldsymbol{\theta}$, το σφάλμα ελαχίστων τετραγώνων που επιτυγχάνεται για $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$ μπορεί να βρεθεί να είναι

$$E(\hat{\boldsymbol{\theta}}) = (\mathbf{y} - \mathbf{A}\hat{\boldsymbol{\theta}})^T (\mathbf{y} - \mathbf{A}\hat{\boldsymbol{\theta}}) = \mathbf{y}^T \mathbf{y} - \mathbf{y}^T \mathbf{A} (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y}. \quad (3.2.13)$$

Όμως, αν ο $\mathbf{A}^T \mathbf{A}$ είναι μη αντιστρέψιμος, τότε ο LSE δεν είναι μοναδικός και πρέπει να εφαρμόσουμε την έννοια του γενικευμένου αντιστρόφου για να βρούμε τον $\hat{\boldsymbol{\theta}}$. Χωρίς απώλεια γενικότητας, θα υποθέσουμε ότι ο $\mathbf{A}^T \mathbf{A}$ είναι αντιστρέψιμος.

Η προηγούμενη παραγωγή είναι βασισμένη στην υπόθεση ότι κάθε στοιχείο του διανύσματος λάθους \mathbf{e} έχει το ίδιο βάρος προς το γενικό τετραγωνικό σφάλμα. Μια περαιτέρω γενίκευση είναι να αφήσουμε κάθε όρο σφάλματος να σταθμιστεί διαφορετικά. Συγκεκριμένα, ας πούμε ότι ο \mathbf{W} είναι ο επιθυμητός πίνακας βαρύτητας, ο οποίος είναι συμμετρικός και θετικά ορισμένος. Τότε το σταθμισμένο τετραγωνικό σφάλμα είναι

$$E_{\mathbf{W}}(\boldsymbol{\theta}) = (\mathbf{y} - \mathbf{A}\boldsymbol{\theta})^T \mathbf{W} (\mathbf{y} - \mathbf{A}\boldsymbol{\theta}). \quad (3.2.14)$$

Ελαχιστοποιώντας το $E_{\mathbf{W}}(\boldsymbol{\theta})$ ως προς $\boldsymbol{\theta}$ παράγουμε το σταθμισμένο εκτιμητή ελαχίστων τετραγώνων $\hat{\boldsymbol{\theta}}_{\mathbf{W}}$:

$$\hat{\theta}_w = (A^T W A)^{-1} A^T W y \quad (3.2.15)$$

Προφανώς, ο $\hat{\theta}_w$ μειώνεται σε $\hat{\theta}$ όταν ο W επιλέγεται σαν ένας μοναδιαίος πίνακας.

3.3 Οπισθοδρόμηση για πρωστοτροφοδοτούμενα δίκτυα

Η ενότητα αυτή παρουσιάζει ένα βασικό κανόνα εκμάθησης για προσαρμόσιμα δίκτυα, που είναι στην ουσία η πιο απλή μέθοδος βαθμωτής ελαχιστοποίησης. Το κεντρικό μέρος αυτού του κανόνα εκμάθησης αφορά πως να πάρουμε επαναληπτικά ένα διάνυσμα κλίσης στο οποίο κάθε στοιχείο ορίζεται ως την παράγωγο ενός μέτρου σφάλματος ως προς μια παράμετρο. Αυτό γίνεται με τη βοήθεια του κανόνα αλυσίδας, ενός βασικού τύπου για το διαφορισμό σύνθετων συναρτήσεων ο οποίος αναλύεται σε κάθε εγχειρίδιο μαθηματικών. Η διαδικασία εύρεσης ενός διανύσματος κλίσης σε ένα δίκτυο αναφέρεται γενικά σαν οπισθοδρόμηση (backpropagation) επειδή το διάνυσμα κλίσης υπολογίζεται σε κατεύθυνση αντίθετη από τη ροή της εξόδου κάθε κόμβου. Μόλις πάρουμε την κλίση, διάφορες τεχνικές βελτιστοποίησης και παλινδρόμησης βασισμένες στις παραγώγους είναι διαθέσιμες για την ενημέρωση των παραμέτρων. Ειδικότερα, εάν χρησιμοποιούμε το διάνυσμα κλίσης σε μια απλή μέθοδο βαθμωτής ελαχιστοποίησης, το προκύπτον παράδειγμα εκμάθησης αναφέρεται συχνά ως κανόνας οπισθόδρομης εκμάθησης. Θα αναλύσουμε αυτόν τον κανόνα εκμάθησης στην συνέχεια αυτής της ενότητας.

Υποθέστε ότι δεδομένο πρωστοτροφοδοτούμενο προσαρμόσιμο δίκτυο στην αναπαράστασή του σε επίπεδα έχει L επίπεδα και το επίπεδο l (όπου $l = 0, 1, \dots, L; l = 0$ αντιπροσωπεύει το επίπεδο εισαγωγής) έχει $N(l)$ κόμβους. Τότε η έξοδος και η συνάρτηση του κόμβου i [$i = 1, \dots, N(l)$] στο επίπεδο l μπορούν να αναπαρασταθούν σαν $x_{l,i}$ και $f_{l,i}$, αντίστοιχα. Χωρίς απώλεια γενικότητας, υποθέτουμε ότι δεν υπάρχει καμία σύνδεση μεταξύ μη συνεχόμενων στρωμάτων. Δεδομένου ότι η έξοδος ενός κόμβου εξαρτάται από τα εισερχόμενα σήματα και το σύνολο παραμέτρων του κόμβου, έχουμε την ακόλουθη γενική έκφραση για τη συνάρτηση των κόμβων $f_{l,i}$:

$$x_{l,i} = f_{l,i}(x_{l-1,1}, \dots, x_{l-1,N(l-1)}, a, b, g, \dots), \quad (3.3.1)$$

όπου a, b, g κλπ. είναι οι παράμετροι αυτού του κόμβου.

Υποθέτοντας ότι το δοσμένο σύνολο δεδομένων εκπαίδευσης έχει P καταχωρήσεις, μπορούμε να ορίσουμε ένα μέτρο σφάλματος για την p -οστή ($1 \leq p \leq P$) καταχώρηση των δεδομένων εκπαίδευσης σαν το άθροισμα των τετραγωνικών σφαλμάτων:

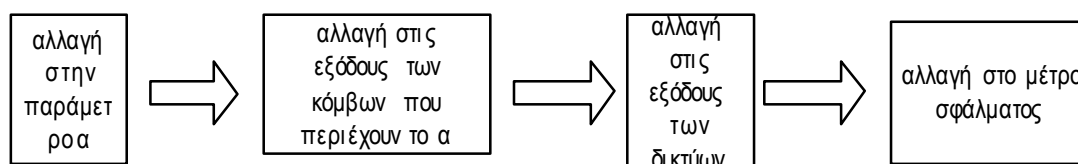
$$E_p = \sum_{k=1}^{N(l)} (d_k - x_{L,k})^2, \quad (3.3.2)$$

όπου d_k είναι το k -οστό συστατικό του p -οστού επιθυμητού διανύσματος εξόδου και $x_{L,k}$ είναι το k -οστό συστατικό του πραγματικού διανύσματος εξόδου που παράγεται με την παρουσίαση του p -οστού διανύσματος εισόδου στο δίκτυο. (Για σημειογραφική απλότητα, παραλείπουμε το δείκτη p και από το d_k και από το $x_{L,k}$). Προφανώς, όταν το E_p είναι ίσο με το μηδέν, το δίκτυο είναι ικανό να αναπαράγει ακριβώς το επιθυμητό διάνυσμα εξόδου στο p -οστό ζευγάρι δεδομένων εκπαίδευσης. Κατά συνέπεια ο στόχος μας εδώ είναι να ελαχιστοποιήσουμε ένα συνολικό μέτρο σφάλματος, που ορίζεται ως $E = \sum_{p=1}^P E_p$.

Θυμηθείτε ότι ο ορισμός του E_p στην εξίσωση (3.3.2) δεν είναι καθολικός αλλά είναι δυνατός και άλλος ορισμός για το E_p για συγκεκριμένες καταστάσεις ή εφαρμογές. Επομένως, θα αποφύγουμε μια ρητή έκφραση για το μέτρο σφάλματος E_p για να δώσουμε έμφαση στη γενικότητα. Επιπλέον, υποθέτουμε ότι το E_p εξαρτάται μόνο από τους κόμβους εξόδου.

Για να χρησιμοποιήσουμε τη βαθμωτή ελαχιστοποίηση ώστε να ελαχιστοποιήσουμε το μέτρο σφάλματος, πρέπει πρώτα να βρούμε το διάνυσμα κλίσης. Πριν

υπολογίσουμε το διάνυσμα κλίσης, πρέπει να παρατηρήσουμε τις ακόλουθες αιτιώδεις σχέσεις:



όπου τα βέλη \Rightarrow δείχνουν τις αιτιώδεις σχέσεις. Με άλλα λόγια, μια μικρή αλλαγή σε μια παράμετρο α θα επηρεάσει την έξοδο του κόμβου που περιέχει το α . Αυτό με τη σειρά του θα επηρεάσει την έξοδο του τελευταίου επιπέδου και συνεπώς το μέτρο σφάλματος. Επομένως, η βασική αρχή στον υπολογισμό του διανύσματος κλίσης είναι να περάσουμε μια σειρά από πληροφορίες παραγώγων ξεκινώντας από το επίπεδο εξόδου και πηγαίνοντας ανάποδα από επίπεδο σε επίπεδο έως ότου φθάσουμε στο επίπεδο εισόδου.

Για να διευκολύνει τη συζήτηση, ορίζουμε το σήμα σφάλματος $\epsilon_{l,i}$ σαν τη παράγωγο του μέτρου σφάλματος E_p ως προς την έξοδο του κόμβου i στο επίπεδο l , λαμβάνοντας υπόψη και τις άμεσες και τις έμμεσες πορείες. Σε σύμβολα,

$$\epsilon_{l,i} = \frac{\partial^+ E_p}{\partial x_{l,i}} \quad (3.3.3)$$

Η έκφραση αυτή ονομάστηκε διατεταγμένη παράγωγος από τον Werbos. Η διαφορά μεταξύ της διατεταγμένης παραγώγου και της συνηθισμένης μερικής παραγώγου βρίσκεται στον τρόπο με τον οποίο βλέπουμε τη συνάρτηση να διαφορίζεται. Για την έξοδο ενός εσωτερικού κόμβου $x_{l,i}$ (όπου $l \neq L$), η μερική

παράγωγος $\frac{\partial E_p}{\partial x_{l,i}}$ είναι ίση με το μηδέν, εφόσον το E_p δεν εξαρτάται από το $x_{l,i}$

έμμεσα, αφού μια αλλαγή στο $x_{l,i}$ θα διαδοθεί μέσω έμμεσων πορειών στο επίπεδο εξόδου και συνεπώς θα προκαλέσει μια αντίστοιχη αλλαγή στην τιμή του E_p . Επομένως, το $\epsilon_{l,i}$ μπορεί να υπολογιστεί ως μια αναλογία αυτών των δύο αλλαγών όταν αυτές γίνονται απειροελάχιστες.

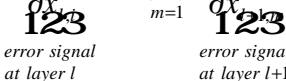
Το σήμα σφάλματος για τον i -οστό κόμβο εξόδου (στο επίπεδο L) μπορεί να υπολογιστεί απευθείας:

$$\epsilon_{L,i} = \frac{\partial^+ E_p}{\partial x_{L,i}} = \frac{\partial E_p}{\partial x_{L,i}} \quad (3.3.4)$$

Αυτό ισούται με $\epsilon_{L,i} = -2(d_i - x_{L,i})$ αν το E_p ορίζεται όπως στην εξίσωση (3.3.2).

Για τον εσωτερικό κόμβο στην i -οστή θέση του επιπέδου l , το σήμα σφάλματος μπορεί να βρεθεί από τον κανόνα αλυσίδας:

$$\epsilon_{l,i} = \frac{\partial^+ E_p}{\partial x_{l,i}} = \sum_{m=1}^{N(l+1)} \frac{\partial^+ E_p}{\partial x_{l+1,m}} \frac{\partial f_{l+1,m}}{\partial x_{l,i}} = \sum_{m=1}^{N(l+1)} \epsilon_{l+1,m} \frac{\partial f_{l+1,m}}{\partial x_{l,i}}, \quad (3.3.5)$$



όπου $0 \leq l \leq L-1$. Δηλαδή το σήμα σφάλματος ενός εσωτερικού κόμβου στο στρώμα l μπορεί να εκφραστεί ως γραμμικός συνδυασμός των κόμβων στο στρώμα $l+1$. Επομένως, για οποιουδήποτε l και i [και $1 \leq i \leq N(l)$], μπορούμε να βρούμε

το $\epsilon_{l,i} = \frac{\partial^+ E_p}{\partial x_{l,i}}$ εφαρμόζοντας πρώτα την εξίσωση (3.3.4) μία φορά για να πάρουμε τα

σήματα σφάλματος στο επίπεδο εξόδου, και έπειτα εφαρμόζοντας την εξίσωση (3.3.5) επαναληπτικά έως ότου φθάσουμε στο επιθυμητό επίπεδο l . Η διαδικασία αυτή καλείται οπισθοδρόμηση δεδομένου ότι τα σήματα σφάλματος λαμβάνονται διαδοχικά από το επίπεδο εξόδου προς το επίπεδο εισόδου.

Το διάνυσμα κλίσης ορίζεται ως την παράγωγο του μέτρου σφάλματος ως προς κάθε παράμετρο, έτσι πρέπει να εφαρμόσουμε τον κανόνα αλυσίδας ξανά για να βρούμε το διάνυσμα κλίσης. Εάν a είναι μια παράμετρος του i -οστού κόμβου στο επίπεδο l , έχουμε:

$$\frac{\partial^+ E_p}{\partial a} = \frac{\partial^+ E_p}{\partial x_{l,i}} \frac{\partial f_{l,i}}{\partial a} = \epsilon_{l,i} \frac{\partial f_{l,i}}{\partial a}. \quad (3.3.6)$$

Σημειώστε ότι αν επιτρέψουμε στην παράμετρο a να μοιράζεται μεταξύ διαφορετικών κόμβων, τότε η εξίσωση (3.3.6) πρέπει να αλλαχθεί σε μια πιο γενική μορφή:

$$\frac{\partial^+ E_p}{\partial a} = \sum_{x^* \in S} \frac{\partial^+ E_p}{\partial x^*} \frac{\partial f^*}{\partial a}, \quad (3.3.7)$$

όπου S είναι το σύνολο των κόμβων που περιέχουν το a σαν παράμετρο ενώ x^* και f^* είναι η έξοδος και η συνάρτηση, αντίστοιχα, ενός γενικού κόμβου στο S .

Η παράγωγος του γενικού μέτρου σφάλματος E ως προς το a είναι

$$\frac{\partial^+ E}{\partial a} = \sum_{p=1}^P \frac{\partial^+ E_p}{\partial a}. \quad (3.3.8)$$

Συνεπώς, για την απλούστερη βαθμωτή ελαχιστοποίηση χωρίς ελαχιστοποίηση γραμμών, ο τύπος για τη γενική παράμετρο a είναι

$$\Delta a = -h \frac{\partial^+ E_p}{\partial a}, \quad (3.3.9)$$

όπου το h είναι ο ρυθμός εκμάθησης, το οποίο μπορεί να εκφραστεί περαιτέρω ως

$$h = \frac{k}{\sqrt{\sum_a \left(\frac{\partial E}{\partial a} \right)^2}}, \quad (3.3.10)$$

όπου το k είναι το μέγεθος βήματος, το μήκος δηλαδή κάθε μετάβασης κατά μήκος της κατεύθυνσης κλίσης στο διάστημα παραμέτρου. Συνήθως μπορούμε να αλλάξουμε το μέγεθος βήματος για να μεταβάλλουμε την ταχύτητα της σύγκλισης.

Όταν ένα πρωστοτροφοδοτούμενο δίκτυο n -κόμβων αναπαριστάται στην τοπολογική του διάταξη, μπορούμε να υπολογίσουμε το μέτρο σφάλματος E_p σαν την έξοδο ενός επιπλέον κόμβου με ένδειξη $n+1$, του οποίου η συνάρτηση κόμβου f_{n+1} μπορεί να οριστεί από τις εξόδους κάθε κόμβου με μικρότερη ένδειξη. (Επομένως, το E_p μπορεί να εξαρτάται άμεσα από οποιουσδήποτε κόμβους.) Εφαρμόζοντας πάλι τον κανόνα αλυσίδας, έχουμε τον ακόλουθο συνοπτικό τύπο για τον υπολογισμό του σήματος σφάλματος $\epsilon_i = \frac{\partial E_p}{\partial x_i}$:

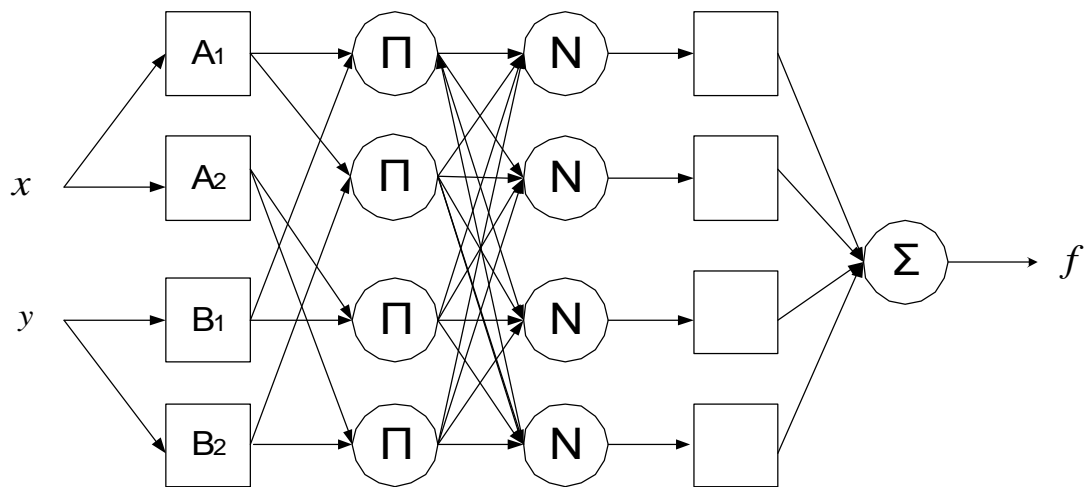
$$\frac{\partial^+ E_p}{\partial x_i} = \frac{\partial f_{n+1}}{\partial x_i} + \sum_{i < j \leq n} \epsilon_j \frac{\partial^+ E_p}{\partial x_j} \frac{\partial f_j}{\partial x_i}, \quad (3.3.11)$$

ή

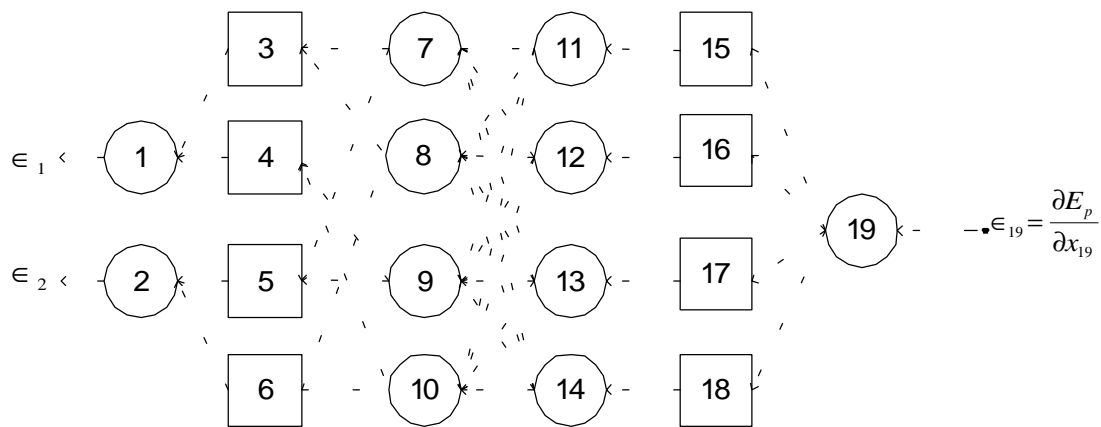
$$\epsilon_i = \frac{\partial f_{n+1}}{\partial x_i} + \sum_{i < j \leq n} \epsilon_j \frac{\partial f_j}{\partial x_i}, \quad (3.3.12)$$

όπου ο πρώτος όρος δείχνει μια άμεση επίδραση του x_i στο E_p μέσω της άμεσης διαδρομής από τον κόμβο i στον κόμβο $n+1$ και κάθε όρος παραγώγου στο άθροισμα δείχνει την έμμεση επίδραση του x_i στο E_p . Μόλις βρούμε το σήμα σφάλματος για κάθε κόμβο, τότε το διάνυσμα κλίσης για τις παραμέτρους παράγεται όπως πριν.

Ένας άλλος συστηματικός τρόπος να υπολογιστούν τα σήματα σφάλματος είναι μέσω της αναπαράστασης του δικτύου διάδοσης σφάλματος (ή του μοντέλου ευαισθησίας), το οποίο λαμβάνεται από το αρχικό προσαρμόσιμο δίκτυο με την αντιστροφή των συνδέσεων και την παροχή των σημάτων σφάλματος στο επίπεδο εξόδου ως είσοδοι στο νέο δίκτυο. Το ακόλουθο σχήμα δείχνει το μοντέλο ANFIS και το δίκτυο διάδοσης σφάλματος του:



Σχήμα 1(a): Το προσαρμόσιμο δίκτυο



Σχήμα 1(b): Το δίκτυο διάδοσης σφάλματος

Τώρα θα υπολογίσουμε τα σήματα σφάλματος στους εσωτερικούς κόμβους. Χρησιμοποιούμε f_i και x_i για να δείξουμε τη συνάρτηση και την έξοδο του κόμβου i . Η έξοδος του κόμβου i είναι το σήμα σφάλματος αυτού του κόμβου στο πραγματικό προσαρμόσιμο δίκτυο. Σε σύμβολα, αν επιλέξουμε το τετραγωνικό μέτρο σφάλματος για E_p , τότε έχουμε το ακόλουθο:

$$\epsilon_{19} = -2(d_{19} - x_{19}),$$

Αυτό είναι επειδή ο κόμβος 19 είναι μόνο ένας κόμβος προσωρινής αποθήκευσης στο δίκτυο διάδοσης σφάλματος. Για τους κόμβους 15, 16, 17 και 18 έχουμε:

$$\begin{aligned}\epsilon_{18} &= \frac{\partial^+ E_p}{\partial x_{18}} = \frac{\partial^+ E_p}{\partial x_{19}} \frac{\partial f_{19}}{\partial x_{18}} = \epsilon_{19} \frac{\partial f_{19}}{\partial x_{18}}, \\ \epsilon_{17} &= \frac{\partial^+ E_p}{\partial x_{17}} = \frac{\partial^+ E_p}{\partial x_{19}} \frac{\partial f_{19}}{\partial x_{17}} = \epsilon_{19} \frac{\partial f_{19}}{\partial x_{17}}, \\ \epsilon_{16} &= \frac{\partial^+ E_p}{\partial x_{16}} = \frac{\partial^+ E_p}{\partial x_{19}} \frac{\partial f_{19}}{\partial x_{16}} = \epsilon_{19} \frac{\partial f_{19}}{\partial x_{16}}, \\ \epsilon_{15} &= \frac{\partial^+ E_p}{\partial x_{15}} = \frac{\partial^+ E_p}{\partial x_{19}} \frac{\partial f_{19}}{\partial x_{15}} = \epsilon_{19} \frac{\partial f_{19}}{\partial x_{15}}.\end{aligned}$$

Αυτό είναι επειδή όλοι αυτοί οι κόμβοι εξαρτώνται από τον κόμβο 19. Αυτό ισχύει και για τους κόμβους 11, 12, 13 και 14 με τη διαφορά ότι κάθε ένας από αυτούς εξαρτάται από διαφορετικό κόμβο:

$$\begin{aligned}\epsilon_{14} &= \frac{\partial^+ E_p}{\partial x_{14}} = \frac{\partial^+ E_p}{\partial x_{18}} \frac{\partial f_{18}}{\partial x_{14}} = \epsilon_{18} \frac{\partial f_{18}}{\partial x_{14}}, \\ \epsilon_{13} &= \frac{\partial^+ E_p}{\partial x_{13}} = \frac{\partial^+ E_p}{\partial x_{17}} \frac{\partial f_{17}}{\partial x_{13}} = \epsilon_{17} \frac{\partial f_{17}}{\partial x_{13}}, \\ \epsilon_{12} &= \frac{\partial^+ E_p}{\partial x_{12}} = \frac{\partial^+ E_p}{\partial x_{16}} \frac{\partial f_{16}}{\partial x_{12}} = \epsilon_{16} \frac{\partial f_{16}}{\partial x_{12}}, \\ \epsilon_{11} &= \frac{\partial^+ E_p}{\partial x_{11}} = \frac{\partial^+ E_p}{\partial x_{15}} \frac{\partial f_{15}}{\partial x_{11}} = \epsilon_{15} \frac{\partial f_{15}}{\partial x_{11}}.\end{aligned}$$

Αντιθέτως, οι κόμβοι 7, 8, 9 και 10 εξαρτώνται από τέσσερις διαφορετικούς κόμβους όπως φαίνεται στο σχήμα 1. Έτσι, έχουμε:

$$\begin{aligned}\epsilon_{10} &= \frac{\partial^+ E_p}{\partial x_{10}} = \epsilon_{14} \frac{\partial f_{14}}{\partial x_{10}} + \epsilon_{13} \frac{\partial f_{13}}{\partial x_{10}} + \epsilon_{12} \frac{\partial f_{12}}{\partial x_{10}} + \epsilon_{11} \frac{\partial f_{11}}{\partial x_{10}}, \\ \epsilon_9 &= \frac{\partial^+ E_p}{\partial x_9} = \epsilon_{14} \frac{\partial f_{14}}{\partial x_9} + \epsilon_{13} \frac{\partial f_{13}}{\partial x_9} + \epsilon_{12} \frac{\partial f_{12}}{\partial x_9} + \epsilon_{11} \frac{\partial f_{11}}{\partial x_9}, \\ \epsilon_8 &= \frac{\partial^+ E_p}{\partial x_8} = \epsilon_{14} \frac{\partial f_{14}}{\partial x_8} + \epsilon_{13} \frac{\partial f_{13}}{\partial x_8} + \epsilon_{12} \frac{\partial f_{12}}{\partial x_8} + \epsilon_{11} \frac{\partial f_{11}}{\partial x_8},\end{aligned}$$

$$\epsilon_7 = \frac{\partial^+ E_p}{\partial x_7} = \epsilon_{14} \frac{\partial f_{14}}{\partial x_7} + \epsilon_{13} \frac{\partial f_{13}}{\partial x_7} + \epsilon_{12} \frac{\partial f_{12}}{\partial x_7} + \epsilon_{11} \frac{\partial f_{11}}{\partial x_7}.$$

Οι κόμβοι 3, 4, 5 και 6 εξαρτώνται από δύο διαφορετικούς κόμβους και σαν αποτέλεσμα έχουμε τις ακόλουθες σχέσεις:

$$\epsilon_6 = \frac{\partial^+ E_p}{\partial x_6} = \frac{\partial^+ E_p}{\partial x_{10}} \frac{\partial f_{10}}{\partial x_6} + \frac{\partial^+ E_p}{\partial x_8} \frac{\partial f_8}{\partial x_6} = \epsilon_{10} \frac{\partial f_{10}}{\partial x_6} + \epsilon_8 \frac{\partial f_8}{\partial x_6},$$

$$\epsilon_5 = \frac{\partial^+ E_p}{\partial x_5} = \frac{\partial^+ E_p}{\partial x_9} \frac{\partial f_9}{\partial x_5} + \frac{\partial^+ E_p}{\partial x_7} \frac{\partial f_7}{\partial x_5} = \epsilon_9 \frac{\partial f_9}{\partial x_5} + \epsilon_7 \frac{\partial f_7}{\partial x_5},$$

$$\epsilon_4 = \frac{\partial^+ E_p}{\partial x_4} = \frac{\partial^+ E_p}{\partial x_{10}} \frac{\partial f_{10}}{\partial x_4} + \frac{\partial^+ E_p}{\partial x_9} \frac{\partial f_9}{\partial x_4} = \epsilon_{10} \frac{\partial f_{10}}{\partial x_4} + \epsilon_9 \frac{\partial f_9}{\partial x_4},$$

$$\epsilon_3 = \frac{\partial^+ E_p}{\partial x_3} = \frac{\partial^+ E_p}{\partial x_8} \frac{\partial f_8}{\partial x_3} + \frac{\partial^+ E_p}{\partial x_7} \frac{\partial f_7}{\partial x_3} = \epsilon_8 \frac{\partial f_8}{\partial x_3} + \epsilon_7 \frac{\partial f_7}{\partial x_3}.$$

Τέλος, οι κόμβοι 1 και 2 εξαρτώνται από δύο κόμβους και έτσι έχουμε:

$$\epsilon_2 = \frac{\partial^+ E_p}{\partial x_2} = \frac{\partial^+ E_p}{\partial x_6} \frac{\partial f_6}{\partial x_2} + \frac{\partial^+ E_p}{\partial x_5} \frac{\partial f_5}{\partial x_2} = \epsilon_6 \frac{\partial f_6}{\partial x_2} + \epsilon_5 \frac{\partial f_5}{\partial x_2},$$

$$\epsilon_1 = \frac{\partial^+ E_p}{\partial x_1} = \frac{\partial^+ E_p}{\partial x_4} \frac{\partial f_4}{\partial x_1} + \frac{\partial^+ E_p}{\partial x_3} \frac{\partial f_3}{\partial x_1} = \epsilon_4 \frac{\partial f_4}{\partial x_1} + \epsilon_3 \frac{\partial f_3}{\partial x_1}$$

3.4 Υβριδικός αλγόριθμος εκμάθησης: Συνδυασμός βαθμωτής ελαχιστοποίησης και εκτιμητή ελαχίστων τετραγώνων

Αν και μπορούμε να εφαρμόσουμε οπισθοδρόμηση ή την βαθμωτή ελαχιστοποίηση για να προσδιορίσουμε τις παραμέτρους σε ένα προσαρμόσιμο δίκτυο, αυτή η απλή μέθοδος βελτιστοποίησης παίρνει συνήθως πολύ χρόνο προτού να συγκλίνει. Μπορούμε να παρατηρήσουμε, όμως, ότι η έξοδος ενός προσαρμόσιμου δικτύου είναι γραμμική σε μερικές από τις παραμέτρους του δικτύου.

Έτσι μπορούμε να προσδιορίσουμε αυτές τις γραμμικές παραμέτρους με τη γραμμική μέθοδο ελαχίστων τετραγώνων που περιγράψαμε παραπάνω. Αυτή η προσέγγιση οδηγεί σε ένα υβριδικό κανόνα εκμάθησης που συνδυάζει τη βαθμωτή ελαχιστοποίηση (SD) και τον εκτιμητή ελαχίστων τετραγώνων (LSE) για γρήγορο προσδιορισμό των παραμέτρων. Στη συνέχεια θα δούμε πως αυτό λειτουργεί.

Το προσαρμόσιμο δίκτυό μας έχει μία έξοδο που αναπαριστάται ως

$$o = F(i, S), \quad (3.4.1)$$

όπου i είναι το διάνυσμα των μεταβλητών εισόδου, S είναι το σύνολο των παραμέτρων, και F είναι η συνολική συνάρτηση που εφαρμόζεται από το προσαρμόσιμο δίκτυο. Αν υπάρχει μια συνάρτηση H τέτοια ώστε η σύνθετη συνάρτηση $H \circ F$ είναι γραμμική σε μερικά από τα στοιχεία του S , τότε αυτά τα στοιχεία μπορούν να προσδιοριστούν από τη μέθοδο ελαχίστων τετραγώνων. Πιο συγκεκριμένα, αν το σύνολο παραμέτρων S μπορεί να χωριστεί σε δύο σύνολα

$$S = S_1 \oplus S_2, \quad (3.4.2)$$

(όπου \oplus αντιπροσωπεύει άμεσο άθροισμα) τέτοια ώστε η $H \circ F$ είναι γραμμική στα στοιχεία του S_2 , τότε εφαρμόζοντας την H στην εξίσωση (3.4.1), έχουμε

$$H(o) = H \circ F(Bi, S), \quad (3.4.3)$$

που είναι γραμμική στα στοιχεία του S_2 . Τώρα με δεδομένα στοιχεία του S_1 , μπορούμε να βάλουμε P δεδομένα εκπαίδευσης στην εξίσωση (3.4.2) και να πάρουμε μια εξίσωση πινάκων:

$$A\theta = y \quad (3.4.4)$$

όπου θ είναι ένα άγνωστο διάνυσμα του οποίου τα στοιχεία είναι παράμετροι στο S_2 . Αυτό είναι ένα κλασσικό γραμμικό πρόβλημα ελαχίστων τετραγώνων, και η

καλύτερη λύση για το $\boldsymbol{\theta}$, που ελαχιστοποιεί το $\|\mathbf{A}\boldsymbol{\theta} - \mathbf{y}\|^2$, είναι ο εκτιμητής ελαχίστων τετραγώνων (LSE) $\boldsymbol{\theta}^*$:

$$\boldsymbol{\theta}^* = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{y}, \quad (3.4.5)$$

όπου \mathbf{A}^T είναι ο ανάστροφος του \mathbf{A} και $(\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T$ είναι ο ψευδοαντίστροφος του \mathbf{A} αν $\mathbf{A}^T \mathbf{A}$ είναι αντιστρέψιμος.

Τώρα μπορούμε να συνδυάσουμε τη βαθμωτή ελαχιστοποίηση και τον εκτιμητή ελαχίστων τετραγώνων για να ενημερώσουμε τις παραμέτρους σε ένα προσαρμόσιμο δίκτυο. Για να εφαρμοστεί υβριδική εκμάθηση σε μια παρτίδα επεξεργασίας (batch mode), κάθε εποχή αποτελείται από ένα πέρασμα προς τα μπρος και ένα πέρασμα προς τα πίσω. Στο πέρασμα προς τα μπρος, αφού παρουσιάζεται ένα διάνυσμα εισόδου, υπολογίζουμε τις εξόδους των κόμβων στο δίκτυο από επίπεδο σε επίπεδο μέχρι να βρούμε μια αντίστοιχη σειρά στους πίνακες \mathbf{A} και \mathbf{y} στην εξίσωση (3.4.4). Αυτή η διαδικασία επαναλαμβάνεται για όλα τα ζευγάρια δεδομένων εκπαίδευσης για να σχηματίσουμε τους ολοκληρωμένους \mathbf{A} και \mathbf{y} . Έπειτα οι παράμετροι στο S_2 προσδιορίζονται από τον ψευδοαντίστροφο τύπο στην εξίσωση (3.4.5). Αφού προσδιορίζονται οι παράμετροι στο S_2 , μπορούμε να υπολογίσουμε το μέτρο σφάλματος για κάθε ζευγάρι δεδομένων εκπαίδευσης. Στο πέρασμα προς τα πίσω, τα σήματα σφάλματος [η παράγωγος του μέτρου σφάλματος ως προς την έξοδο κάθε κόμβου, βλέπε εξισώσεις (3.3.4) και (3.3.5)] διαδίδονται από το άκρο της εξόδου ως το άκρο της εισόδου. Το διάνυσμα κλίσης αυξάνεται για κάθε είσοδο δεδομένων εκπαίδευσης. Στο τέλος του περάσματος προς τα πίσω για όλα τα δεδομένα εκπαίδευσης, οι παράμετροι στο S_1 ενημερώνονται από τη βαθμωτή ελαχιστοποίηση στην εξίσωση (3.3.9).

Για δεδομένες σταθερές τιμές των παραμέτρων στο S_1 , οι παράμετροι στο S_2 που βρίσκονται είναι εγγυημένο ότι είναι στο ολικό βέλτιστο σημείο του διαστήματος παραμέτρων του S_2 εξαιτίας της επιλογής του τετραγωνικού μέτρου σφάλματος. Αυτός ο υβριδικός κανόνας εκμάθησης μπορεί όχι μόνο να μειώσει τη διάσταση του διαστήματος εξερεύνησης που εξερευνείται από την πραγματική μέθοδο βαθμωτής

ελαχιστοποίησης, αλλά, γενικά, θα μειώσει σημαντικά το χρόνο που απαιτείται για να φθάσει στη σύγκλιση.

Από την αρχιτεκτονική του ANFIS που δείξαμε προηγουμένως, παρατηρούμε ότι όταν οι τιμές των αρχικών παραμέτρων (premise parameters) είναι σταθερές, η συνολική έξοδος μπορεί να εκφραστεί σαν γραμμικός συνδυασμός των επακόλουθων παραμέτρων (consequent parameters). Σε σύμβολα, η έξοδος f μπορεί να ξαναγραφτεί ως:

$$\begin{aligned} f &= \frac{w_1}{w_1 + w_2 + w_3 + w_4} f_1 + \frac{w_2}{w_1 + w_2 + w_3 + w_4} f_2 + \frac{w_3}{w_1 + w_2 + w_3 + w_4} f_3 + \frac{w_4}{w_1 + w_2 + w_3 + w_4} f_4 \\ &= \bar{w}_1(p_1x + q_1y + r_1) + \bar{w}_2(p_2x + q_2y + r_2) + \bar{w}_3(p_3x + q_3y + r_3) + \bar{w}_4(p_4x + q_4y + r_4) \\ &= (\bar{w}_1x)p_1 + (\bar{w}_1y)q_1 + (\bar{w}_1)r_1 + (\bar{w}_2x)p_2 + (\bar{w}_2y)q_2 + (\bar{w}_2)r_2 + (\bar{w}_3x)p_3 + (\bar{w}_3y)q_3 + (\bar{w}_3)r_3 \\ &\quad + (\bar{w}_4x)p_4 + (\bar{w}_4y)q_4 + (\bar{w}_4)r_4 \end{aligned}$$

που είναι γραμμική στις επακόλουθες παραμέτρους (consequent parameters)

$p_1, q_1, r_1, p_2, q_2, r_2, p_3, q_3, r_3, p_4, q_4$ και r_4 . Από την παρατήρηση αυτή έχουμε

S = σύνολο συνολικών παραμέτρων,

S_1 = σύνολο αρχικών παραμέτρων (μη γραμμικών),

S_2 = σύνολο επακόλουθων παραμέτρων (γραμμικών)

στην εξίσωση (3.4.2) ενώ $H(\cdot)$ και $F(\cdot, \cdot)$ είναι η μοναδιαία συνάρτηση και η συνάρτηση του ασαφούς συστήματος συμπερασμού (fuzzy inference system), αντίστοιχα στην εξίσωση (3.4.3). Επομένως, ο υβριδικός αλγόριθμος εκμάθησης που αναπτύσσεται παραπάνω μπορεί να εφαρμοστεί άμεσα. Πιο συγκεκριμένα, στο πέρασμα προς τα μπρος του υβριδικού αλγορίθμου, οι έξοδοι κόμβων πηγαίνουν προς τα εμπρός μέχρι το επίπεδο 4 και οι επακόλουθες παράμετροι προσδιορίζονται με τη μέθοδο ελαχίστων τετραγώνων. Στο πέρασμα προς τα πίσω, τα σήματα σφάλματος διαδίδονται προς τα πίσω και οι αρχικές παράμετροι ενημερώνονται από την βαθμωτή ελαχιστοποίηση. Ο πίνακας 1 συνοψίζει τις δραστηριότητες που πραγματοποιούνται σε κάθε πέρασμα.

	Πέρασμα προς τα μπρος	Πέρασμα προς τα πίσω
Αρχικές παράμετροι	Σταθερές	Βαθμωτή ελαχιστοποίηση

Επακόλουθες παράμετροι	Εκτιμητής ελαχίστων τετραγώνων	Σταθερές
Σήματα	Έξοδοι κόμβων	Σήματα σφάλματος

Πίνακας 1: Δύο περάσματα στην υβριδική διαδικασία εκμάθησης για το ANFIS

ΚΕΦΑΛΑΙΟ 4: Παρουσίαση της χρήσης του μοντέλου ANFIS και χρησιμοποιούμενη μεθοδολογία

Εισαγωγή

Στο κεφάλαιο αυτό θα ασχοληθούμε με τη παρουσίαση του μοντέλου μας, ξεκαθαρίζοντας την χρησιμοποίηση διαφόρων παραγόντων, τόσο για την εκπαίδευση του μοντέλου μας όσο και για τον έλεγχο των αποτελεσμάτων μας. Στόχος της παρούσας μελέτης είναι προφανώς να ελαχιστοποιήσουμε όσο το δυνατόν περισσότερο το σφάλμα πρόβλεψης για να πετύχουμε καλύτερα αποτελέσματα.

Περιγραφή μοντέλου

Όπως έχει ειπωθεί το μοντέλο με το οποίο κάνουμε την πρόβλεψη της ενέργειας του ανέμου, είναι το μοντέλο ANFIS, το οποίο χρησιμοποιεί την Sugeno-type FIS μορφή. Το μοντέλο μας προβλέπει την ημερήσια παραγωγή ενέργειας από τον άνεμο για την επόμενη ημέρα. Επίσης η μέθοδος δοκιμής και σφάλματος χρησιμοποιήθηκε για να αποφασιστεί ο τύπος της συνάρτησης συμμετοχής που περιγράφει καλύτερα το μοντέλο μας και μας δίνει το μικρότερο δυνατό σφάλμα. Τελικά η τραπεζοειδής συνάρτηση μεταφοράς μας έδωσε καλύτερα αποτελέσματα από τη γκαουσιανή και της παραλλαγές της καθώς και από τη κωδωνοειδή και την τριγωνική. Μετά από αρκετές δοκιμές κρίθηκε σκόπιμο να χρησιμοποιήσουμε στο μοντέλο πρόβλεψης που σχεδιάσαμε δυο συναρτήσεις μεταφοράς τραπεζοειδούς σχήματος που υπακούουν στον παρακάτω τύπο:

$$\text{trapmf}(x, a, b, c, d) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right)$$

Όπως ίσως είναι εύκολα αντιληπτό το αναπτυχθέν μοντέλο ANFIS χρησιμοποιεί σαν ‘και’, δηλαδή σαν διαδικασία συγκερασμού δυο γεγονότων το παραγόμενο προϊόν, που στη προκειμένη περίπτωση είναι η τελική ποσότητα παραγωγής ρεύματος από τον άνεμο. Επίσης σαν ‘ή’ δηλαδή σαν διαδικασία αποδοχής ή απόρριψης του ενός ή του άλλου πράγματος τη μέγιστη τιμή της ταχύτητας του ανέμου. Επιπλέον το μοντέλο μας χρησιμοποιεί ένα υβριδικό αλγόριθμο εκπαίδευσης, έτσι ώστε να αναγνωρίζει τις παραμέτρους του μοντέλου ANFIS. Το υβρίδιο αυτό είναι ένας συνδυασμός της μεθόδου των ελαχίστων τετραγώνων(για τον ορισμό κυρίως των κανόνων που διέπουν το μοντέλο) και της μεθόδου οπισθοδρόμησης για

προσωποτροφοδοτούμενα μοντέλα(για την εκπαίδευση των παραμέτρων του νευρο-ασαφούς μοντέλου, κυρίως την συνάρτηση μεταφοράς, δοθέντος ενός σετ δεδομένων). Τέλος, είναι πολύ βασικό να αναφέρουμε ότι χρησιμοποιούμε και ένα σετ δεδομένων τα οποία θα κάνουν μια επαλήθευση των δεδομένων που θα μας δώσουν τα αποτελέσματα της πρόβλεψης, τα λεγόμενα checking data.

Στον παρακάτω πίνακα βλέπουμε ότι αρχικά δημιουργήθηκαν τέσσερα μοντέλα ANFIS με τέσσερις διαφορετικές εισόδους για κάθε ένα από αυτά τα μοντέλα. Έτσι έχουμε έναν συγκεντρωτικό πίνακα με τα μοντέλα και τις υποψήφιες εισόδους. Χρησιμοποιούμε τρεις χρονικές περιόδους για να ορίσουμε τις εισόδους μας την (κ-1),(κ-2),(κ-3). Στην πραγματικότητα αντιπροσωπεύουν την μέση ταχύτητα του ανέμου μια ημέρα πριν και δυο ή τρεις ακόμα ημέρες νωρίτερα. Προφανώς η έξοδος του μοντέλου μας είναι η παραγωγή ενέργειας από το άνεμο για την επόμενη όμως ημέρα.

ΠΙΝΑΚΑΣ1.Παραλλαγές εισόδων για κάθε μοντέλο

Models	Περιγραφή των εισόδων	No of MFs
ANFIS 1	M_S AVG: μέση ταχύτητα του ανέμου για k-1	2
	M_S AVG: μέση ταχύτητα του ανέμου για k-2	2
ANFIS 2	M_S AVG: μέση ταχύτητα του ανέμου για k-1	2
	M_S AVG: μέση ταχύτητα του ανέμου για k-3	2
ANFIS 3	M_S MAX: μέγιστη ταχύτητα του ανέμου για k-1	2
	M_S AVG: μέγιστη ταχύτητα του ανέμου για k-2	2
ANFIS 4	M_S MAX: μέγιστη ταχύτητα του ανέμου για k-1	2
	M_S AVG: μέγιστη ταχύτητα του ανέμου για k-3	2

Κάθε είσοδος χαρακτηρίζεται από μια χαμηλή και μια υψηλή περιοχή. Αφού στη συνέχεια εκπαιδεύσαμε το μοντέλο μας, προέκυψαν αυτόματα οι κανόνες στους οποίους αυτό υπακούει και οι οποίοι έχουν την παρακάτω μορφή:

R1: εάν x είναι A_1 και y είναι B_1 τότε $f_1 = p_1 \times x + q_1 \times y + r_1$

R2: εάν x είναι A_2 και y είναι B_2 τότε $f_2 = p_2 \times x + q_2 \times y + r_2$

Στον πίνακα 2.φαίνονται ξεκάθαρα οι τέσσερις κανόνες που το μοντέλο ANFIS1 χρησιμοποιεί για να επιλέξει την βέλτιστη στρατηγική ελέγχου. Το μοντέλο ANFIS2 όπως φαίνεται υπακούει στους ίδιους κανόνες με τη διαφορά ότι χρησιμοποιεί ως ορίζοντα χρονικό τις τρεις προηγούμενες ημέρες. Το μοντέλο ακόμα ANFIS3 είναι πανομοιότυπο με το πρώτο ενώ το ANFIS4 είναι ίδιο με το τρίτο χρησιμοποιώντας όμως διαφορετικό χρονικό ορίζοντα. Αναλυτικά τα αποτελέσματα έχουν ως εξής:

ΠΙΝΑΚΑΣ2.Οι κανόνες του ANFIS 1 μοντέλου

κ	Περιγραφή κανόνα
---	------------------

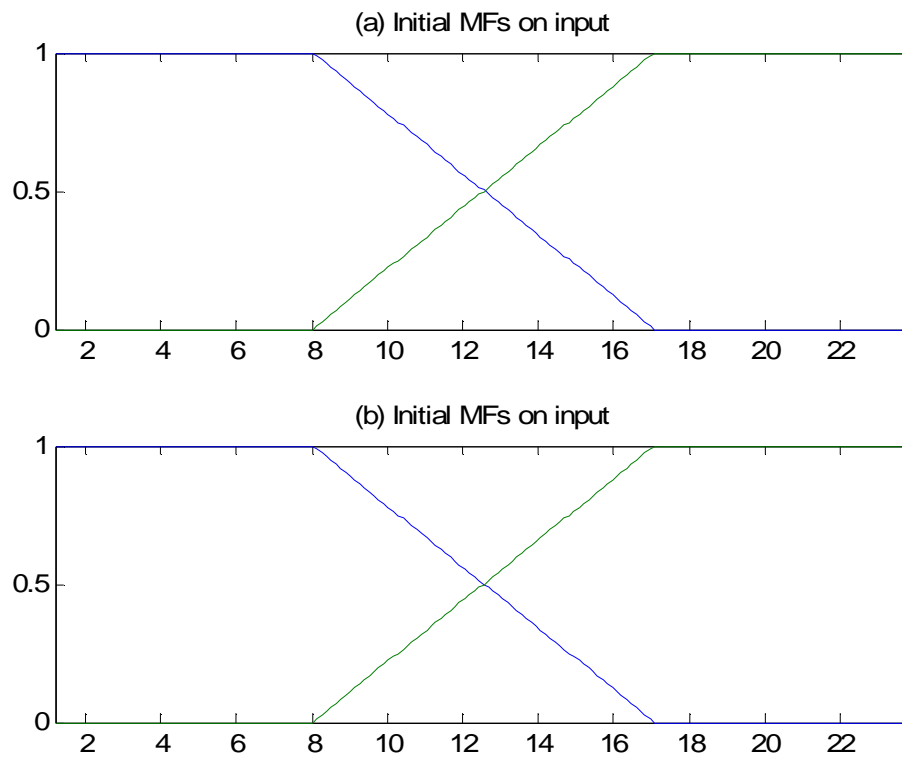
ανόνας

- R1: εάν $M_S\ AVG\ (k-1)$ είναι χαμηλή και $M_S\ AVG\ (k-2)$ είναι χαμηλή τότε η έξοδος είναι η out1mf1
- R2: εάν $M_S\ AVG\ (k-1)$ είναι χαμηλή και $M_S\ AVG\ (k-2)$ είναι υψηλή τότε η έξοδος είναι out1mf2
- R3: εάν $M_S\ AVG\ (k-2)$ είναι υψηλή και $M_S\ AVG\ (k-1)$ είναι χαμηλή τότε η έξοδος είναι out1mf3
- R4: εάν $M_S\ AVG\ (k-2)$ είναι υψηλή και $M_S\ AVG\ (k-1)$ είναι υψηλή τότε η έξοδος είναι out1mf4
-

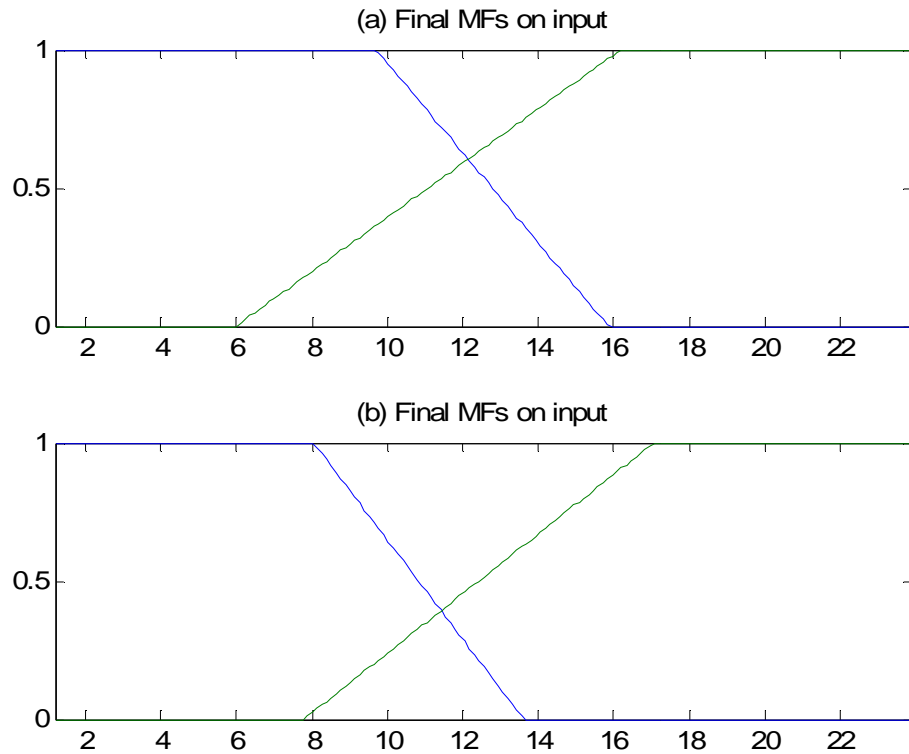
Για να έχουμε τη δυνατότητα να συγκρίνουμε τα μοντέλα ANFIS AR και ARX είναι απαραίτητο και σκόπιμο να χρησιμοποιήσουμε τα ίδια σετ δεδομένων ώστε να μην υπάρχει καμιά απολύτως αμφιβολία για την αξιοπιστία των αποτελεσμάτων μας .

ΚΕΦΑΛΑΙΟ 5: ΠΑΡΟΥΣΙΑΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

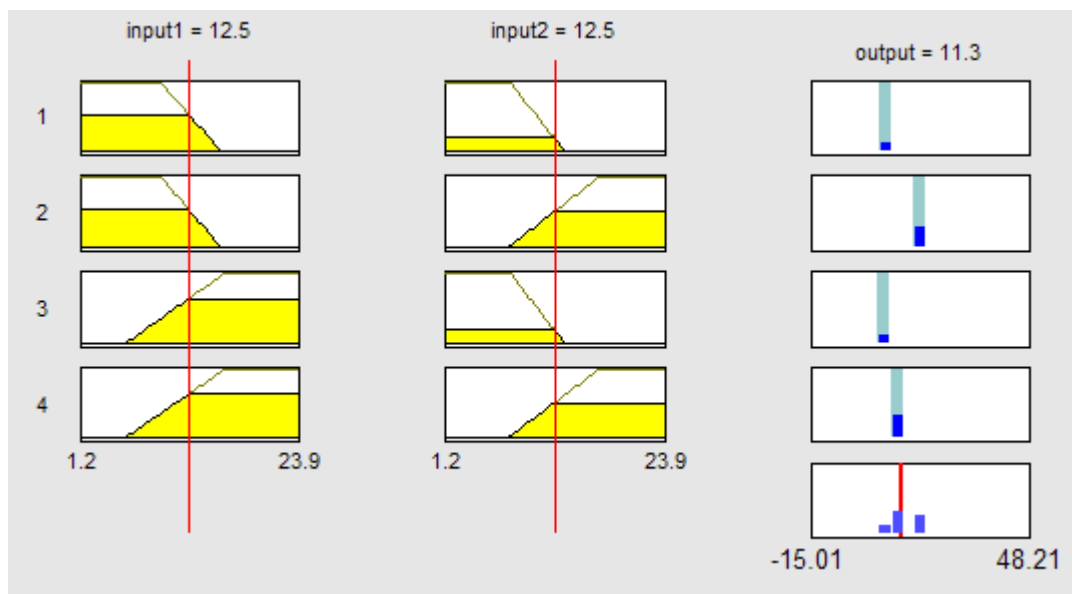
Τα δεδομένα που χρησιμοποιούμε για να εφαρμόσουμε το μοντέλο μας είναι πραγματικά και έχουν παρθεί από ένα αιολικό πάρκο στην Εύβοια. Συνολικά καταφέραμε να συλλέξουμε 365 δείγματα, από την περίοδο 9 Μαρτίου 2005 έως 22 Μαρτίου 2006. Το 68.5% των συνολικών δεδομένων δηλαδή τα 248 δείγματα χρησιμοποιήθηκαν για την εκπαίδευση του μοντέλου μας, ενώ το 31.5% των υπολοίπων δεδομένων, δηλαδή τα 113 υπόλοιπα δείγματα για να ελέγξουμε την ορθότητα των αποτελεσμάτων μας. Πολλές διαφορετικές παραλλαγές του γνωστού μοντέλου ANFIS εφαρμόστηκαν και ελέγχθηκαν με βάση τα αποτελέσματα τα οποία έδωσαν. Αρχικά ελέγξαμε το ANFIS1, το οποίο είχε ως εισόδους την μέση ταχύτητα του ανέμου μια και δυο μέρες πριν. Αργότερα ελέγξαμε το ANFIS2 που είχε ως εισόδους την μέση ταχύτητα του ανέμου μια αλλά και τρεις μέρες νωρίτερα. Το ANFIS3 είχε ως εισόδους τη μέγιστη ταχύτητα του ανέμου μια και δυο μέρες πριν, ενώ το ANFIS4 είχε ως εισόδους την μέγιστη ταχύτητα του ανέμου μια και τρεις μέρες νωρίτερα. Όπως ίσως έχει γίνει αντιληπτό και από την έως τώρα ανάλυση του μοντέλου μας οι διαδικασίες της εκπαίδευσης και τσεκαρίσματος των δεδομένων μας είναι και ο βασικός παράγοντας που κρίνει τελικά την ορθότητα των αποτελεσμάτων μας και μπορεί να δώσει την πρέπουσα αξιοπιστία στο μοντέλο μας. Οι δυο αυτές διαδικασίες έχουν ορισθεί επαρκώς στον αλγόριθμο μας. Η παράμετρος κ συμβολίζει το χρόνο που μεταβάλλεται ανάλογα την παραλλαγή του μοντέλου που χρησιμοποιείται κάθε φορά. Στα παρακάτω σχήμα (σχήμα3) φαίνονται παραστατικά οι αρχικές συναρτήσεις μεταφοράς για κάθε είσοδο πριν όμως την εκπαίδευση του μοντέλου μας. Στο σχήμα4. φαίνεται η τελική συνάρτηση μεταφοράς η οποία και προέκυψε μετά από πάρα πολλές δοκιμές.



Σχήμα 3. Συναρτήσεις μεταφοράς πριν την εκπαίδευση



Σχήμα 4. Συναρτήσεις μεταφοράς μετά την εκπαίδευση



Σχήμα 5. Σύνοψη των κανόνων και του μηχανισμού αποφάσεων

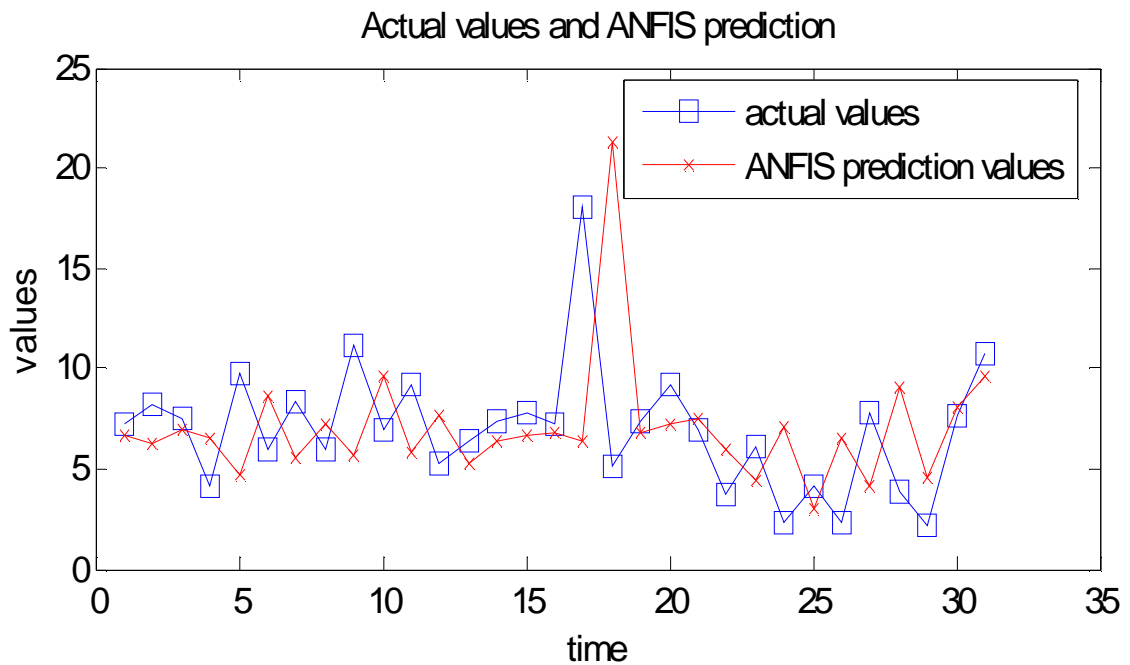
Για να μετρήσουμε ποιοτικότερα την αξιοπιστία των αποτελεσμάτων μας και να μειώσουμε όσο το δυνατόν περισσότερο την απόκλιση τους από την πραγματικότητα χρησιμοποιήσαμε κάποια μέτρα μέτρησης τους σφάλματος όπως είναι εύκολα αντιληπτό από τον παρακάτω πίνακα. Κάποια από αυτά είναι ενδεικτικά το μέσο τετραγωνικό σφάλμα, η ρίζα του μέσου τετραγωνικού σφάλματος και το μέσο απόλυτο σφάλμα. Επίσης στον πίνακα που ακολουθεί φαίνεται ξεκάθαρα η σύγκριση των αποτελεσμάτων του μοντέλου ANFIS στη παραλλαγή του. Όταν δηλαδή έχω δυο τραπεζοειδής συναρτήσεις μεταφοράς και κάποιων άλλων μοντέλων πρόβλεψης όπως είναι τα μοντέλα AR και ARMA.

Πίνακας 3. Αποτελέσματα πρόβλεψης

Errors	ANFIS 1	ANFIS 2	ANFIS 3	ANFIS 4	AR	ARMA
MSE	15.79	24.95	91.27	65.81	17.07	16.04
RMSE	3.97	4.99	9.95	8.11	4.13	4.05
MAE	2.73	3.28	6.84	6.13	2.82	2.79
MAPE	36.04	41.33	36.98	37.00	37.61	37.17

Όπως είναι εύκολα αντιληπτό από την παραπάνω σύγκριση των αποτελεσμάτων, το μοντέλο με το μικρότερο δυνατό σφάλμα είναι το ANFIS1, το οποίο έχει ως εισόδους την μέση τιμή της ταχύτητας του ανέμου για μια και δυο μέρες πριν.

Στη συνέχεια ακολουθεί μια γραφική σύγκριση των πραγματικών δεδομένων αλλά και αυτών που εκτιμήθηκαν για το ANFIS1. Με την μπλε γραμμή που έχει ως κορυφές τετράγωνα είναι τα πραγματικά δεδομένα, ενώ με την κόκκινη που καταλήγει σε αστερίσκους είναι αυτά που έχουν εκτιμηθεί.



ΕΠΙΛΟΓΟΣ-ΣΥΜΠΕΡΑΣΜΑΤΑ

Η παρούσα διπλωματική έχει ως σκοπό να μπορούμε να προβλέψουμε την παραγωγή ενέργειας από τον άνεμο για το διάστημα της επόμενης ημέρας. Η ακρίβεια των αποτελεσμάτων μετρώνται με συγκεκριμένα είδη σφάλματος (MSE, RMSE, MAE, MAPE) και συγκρίνονται και με τα αποτελέσματα άλλων μοντέλων πρόβλεψης τα οποία όμως είναι διαφορετικής φιλοσοφίας από το προτεινόμενο μοντέλο ANFIS. Πιο συγκεκριμένα, τα μοντέλα AR και ARMA ανήκουν στην κατηγορία των παραμετρικών συστημάτων, ενώ το μοντέλο ANFIS ανήκει στα λεγόμενα νευρο-ασαφή συστήματα της κατηγορίας των μη παραμετρικών συστημάτων. Επιπλέον, το προτεινόμενο μοντέλο ANFIS είναι ένα αρκετά εύκολο μοντέλο στην χρήση του, και σε σχέση με άλλα παραδοσιακά μοντέλα πρόβλεψης, απαιτείται μικρότερη εκπαίδευση, έτσι ώστε να πραγματοποιηθούν οι απαραίτητοι υπολογισμοί πρόβλεψης της ενέργειας από τον άνεμο. Ακόμα το προτεινόμενο μοντέλο ANFIS δίνει πολύ καλύτερα αποτελέσματα από παρόμοια με αυτό μοντέλα στην επίλυση μη γραμμικών προβλημάτων μιας και δεν βασίζεται σε πρότυπα προβλήματα μοντελοποίησης, τα οποία θα πρέπει κάθε φορά να τροποποιούνται κατάλληλα, έτσι ώστε να ταιριάζουν απόλυτα με το πρόβλημα προς επίλυση. Τέλος η δομή των κανόνων που διέπουν τη λειτουργία του μοντέλου ANFIS είναι ιδιαίτερα εμφανής στο χρήστη, πράγμα που είναι ιδιαίτερα σημαντικό μιας και μπορεί να εξάγει ευκολότερα πληροφορίες για την εμπειρική σχέση μεταξύ της μέσης ταχύτητας του ανέμου και της παραγόμενης ενέργειας κάθε φορά ανάμεσα στο χρονικό διάστημα που μας ενδιαφέρει. Παρόλα βέβαια τα παραπάνω εμφανή

πλεονεκτήματα που φαίνεται να έχει το προτεινόμενο μοντέλο ANFIS πρέπει η εφαρμογή του να γίνεται με πολύ προσεχτικό τρόπο. Ο ελάχιστος αριθμός των δειγμάτων που θα πρέπει κάποιος να συλλέξει δεν θα πρέπει να είναι κάτω του 150. Επίσης ο αριθμός των παραμέτρων του μοντέλου δεν θα πρέπει να υπερβαίνει το ένα τέταρτο του αριθμού των δειγμάτων κατά την εκπαίδευση του μοντέλου, έτσι ώστε να αποφύγουμε την άσκοπη μεγιστοποίηση της πολυπλοκότητας του αλγορίθμου μας αλλά και την έλλειψη γενικότητας που θα πρέπει να χαρακτηρίζει και ένα μοντέλο που ανήκει στην ομάδα των νευρο-ασαφών συστημάτων.

ΠΑΡΑΡΤΗΜΑ

% PREDICTION USING ---ANFIS1--- MODEL

```
close all %clean the workspace
clear
clc
```

```
mydata=xlsread ('27738-WT1fix', 'd2:d366') % retrieves the E column data of range 1
to 252 from the data file 'atermonddata'
```

```
% load mydata
tic
figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')
```

```
% Estimation of an ANFIS model
% prepare training data
% input data
tr=mydata(1:250) % TRAINING DATA
% input (k-2)
train=tr
train(length(train))=[] % removes the last row
train(length(train))=[]; % removes the second last row
length(train)
```

```
% input k-1
```

```
train1=tr % first input
train1(length(train1))=[] % removes the last row
train1(1)=[] % removes the first row
length(train1)
```

```
% output k
train2=tr % second input
train2(1)=[] % removes the first row
train2(1)=[] % removes the second row
length(train2)
```

```
trn_data=[train train1 train2 ] % (k-2) (k-1) (k) training data
```

```

%preparing the evaluation(test) data
ev=mydata(251:end)% TESTING DATA

% input (k-2)
eval=ev
eval(length(eval))=[]; eval(length(eval))=[];
length(eval)

% input (k-1)
eval1=ev
eval1(length(eval))=[];
eval1(1)=[];
length(eval)

% input (k)
eval2=ev
eval2(1)=[]
eval2(1)=[];
length(eval2)

evaldata=[eval eval1] %input (k-2) and (k-1)

y2=eval2 % (k) output data for testing


% generate FIS matrix
epoch_n=300
mf_n=2;

%mf_type='gbellmf'; %type of membership function
%mf_type='trimf'; %the parameter b>c
%mf_type='gauss2mf'
mf_type='gaussmf' %
%mf_type='smf' % unsupported
%mf_type='trapmf' % the parameter b>c
%mf_type='zmf' % unsupported
%mf_type='pimf' % run problem
in_fismat=genfis1(trn_data, mf_n, mf_type);

% start training
ss=0.1;

[m_anfis trn_error step_size ] = ...
    anfis(trn_data, in_fismat, [epoch_n nan ss nan nan], [1,1,1,1]);

```

```

figure('name',...
      ['ANFIS: time series prediction (file: ' mfilename ')'],...
      'number', 'off');

subplot(211);
tmp=[trn_error ];
plot(tmp);
title('Error Curves');
axis([0 epoch_n min(tmp(:)) max(tmp(:))]);
xlabel('epochs')
ylabel('RMSE')
legend('Training Error');

subplot(212);
plot(step_size);
xlabel('epochs')
title('Step Size');

input=evaldata
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%prediction by ANFIS model out of sample

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ANFIS EVALUATION
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
yhat_anfis=evalfis(input, m_anfis);

adapt_input=y2;
length(adapt_input);
length(yhat_anfis);
result=[adapt_input yhat_anfis (adapt_input-yhat_anfis)];
er_anfis=adapt_input-yhat_anfis; %error

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure (20)
plot(adapt_input(end-30:end), 'b-s'), hold, plot(yhat_anfis(end-30:end), 'r-x');
legend('actual values','ANFIS prediction values')
xlabel('time')
ylabel('values')
title('Actual values and ANFIS prediction')

```



```

figure (30)
plot(adapt_input-yhat_anfis)
xlabel('time')
ylabel('error')
title('Prediction errors')

```

```

%%%%%%%%%%%%%% *****
**

```

```

% plot the initial membership functons
figure (60)
subplot(2,1,1)
[mfx, mfy]=plotmf(in_fismat, 'input', 1);
plot(mfx, mfy);
title('(a) Initial MFs on input')
axis([-inf inf -inf inf]);
subplot(2,1,2)
[mfx, mfy]=plotmf(in_fismat, 'input', 2);
plot(mfx, mfy);
title('(b) Initial MFs on input')
axis([-inf inf -inf inf]);

```

```

% plot final MF's on x,y,z,u

```

```

figure (70)
subplot(2,1,1)
[mfx, mfy]=plotmf(m_anfis, 'input', 1);
plot(mfx, mfy);
title('(a) Final MFs on input')
axis([-inf inf -inf inf]);
subplot(2,1,2)
[mfx, mfy]=plotmf(m_anfis, 'input', 2);
plot(mfx, mfy);
title('(b) Final MFs on input')
axis([-inf inf -inf inf]);

```

```

%%%%%%%%%% Ploting TRAINING data as a scater plot%%%%%%%%%%
figure(80)
subplot(1,2,1)
plot (train, train2, 'o')
xlabel ('input (k-2) ')
ylabel('ouput (k)')
title('Training data')
axis equal; axis square

```

```

subplot(1,2,2)
plot (train1, train2, 'o')
xlabel ('input (k-1) ')
ylabel('ouput (k)')
title('Training data')
axis equal; axis square

%%%%%% Ploting EVALUATION data as a scater plot%%%%%%%%
figure(90)
subplot(1,2,1)
plot (eval, eval2, 'o')
xlabel ('input (k-2)')
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square

subplot(1,2,2)
plot (eval1, eval2, 'o')
xlabel ('input (k-1)')
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square

figure(100)
plotfis(in_fismat)

figure(110)
plotfis(m_anfis)

showrule(m_anfis)

anfisedit (m_anfis)
surfview(m_anfis)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error measures calculation

MSE_anfis=(1/length(yhat_anfis))*norm(er_anfis)^2

%Root Mean Square Error (RMSE)
RMSE_anfis=sqrt(norm(er_anfis)^2/length(er_anfis))

%Mean Absolute Error (MAE)
MAE_anfis=(1/length(er_anfis))*sum(abs(er_anfis))

```

```
%Mean Absolute percentage Error (MAPE)
MAPE_anfis=(100/length(er_anfis))*sum(abs(er_anfis)./abs(adapt_input))

toc % the calculation time in seconds
runing_minute_time=toc/60
%end
```

% PREDICTION USING ---ANFIS2--- MODEL

```
close all %clean the workspace
clear
clc
```

```
mydata=xlsread ('27738-WT1fix', 'd2:d366') % retrieves the E column data of range 1
to 252 from the data file 'atermonddata'
```

```
% load mydata
tic
figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')
```

```
% Estimation of an ANFIS model
% prepare training data
% input data
tr=mydata(1:250) % TRAINING DATA
```

```
% input (k-3)
train=tr;
train(length(train))=[] % removes the last row
train(length(train))=[]; %removes the second last row
train(length(train))=[];
length(train)
```

```
% input (k-2)
train1=tr;
train1(length(train1))=[];% removes the last row
train1(length(train1))=[]; %removes the second last row
train1(1)=[]; % removes the first row
length(train1)
```

```
% input k-1
```

```
train2=tr; % first input
train2(length(train2))=[];% removes the last row
train2(1)=[];% removes the first row
train2(1)=[];
length(train2)
```

```
% output k
train3=tr; %second input
train3(1)=[]; % removes the first row
```

```

train3(1)=[]; %removes the second row
train3(1)=[];
length(train3)

trn_data=[train train1 train3] %(k-2) (k-1) (k) training data

%preparing the evaluation(test) data
ev=mydata(251:end)% TESTING DATA

% input (k-3)
eval=ev
eval(length(eval))=[]; eval(length(eval))=[];eval(length(eval))=[];
length(eval)
% input (k-2)
eval1=ev
eval1(length(eval1))=[]; eval1(length(eval1))=[];eval1(1)=[];
length(eval1)

% input (k-1)
eval2=ev
eval2(length(eval2))=[];
eval2(1)=[];eval2(1)=[];
length(eval2)

% input (k)
eval3=ev
eval3(1)=[];
eval3(1)=[];
eval3(1)=[];
length(eval3)

evaldata=[eval eval2] %input (k-2) and (k-1)

y2=eval3 %(k) output data for testing


% generate FIS matrix
epoch_n=300
mf_n=2;

%mf_type='gbellmf'; %type of membership function
%mf_type='trimf'; %the parameter b>c
%mf_type='gauss2mf'

```

```

mf_type='gaussmf' %
%mf_type='smf' % unsupported
%mf_type='trapmf' % the parameter b>c
%mf_type='zmf' % unsupported
%mf_type='pimf' % run problem
in_fismat=genfis1(trn_data, mf_n, mf_type);

% start training
ss=0.1;

[m_anfis trn_error step_size ] = ...
    anfis(trn_data, in_fismat, [epoch_n nan ss nan nan], [1,1,1,1]);

figure('name',...
    ['ANFIS: time series prediction (file: ' mfilename ')'],...
    'number', 'off');

subplot(211);
tmp=[trn_error ];
plot(tmp);
title('Error Curves');
axis([0 epoch_n min(tmp(:)) max(tmp(:))]);
xlabel('epochs')
ylabel('RMSE')
legend('Training Error');

subplot(212);
plot(step_size);
xlabel('epochs')
title('Step Size');

input=evaldata
%%%%%%%%%%%%%%
%%%%%%%%%%%%%%

%prediction by ANFIS model out of sample

%%%%%%%%%%%%%% ANFIS EVALUATION
%%%%%%%%%%%%%%
yhat_anfis=evalfis(input, m_anfis);

adapt_input=y2;
length(adapt_input);

```

```

length(yhat_anfis);
result=[adapt_input yhat_anfis (adapt_input-yhat_anfis)];
er_anfis=adapt_input-yhat_anfis; %error

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure (20)
plot(adapt_input(end-30:end), 'b-s'), hold, plot(yhat_anfis(end-30:end), 'r-x');
legend('actual values','ANFIS prediction values')
xlabel('time')
ylabel('values')
title('Actual values and ANFIS prediction')

figure (30)
plot(adapt_input-yhat_anfis)
xlabel('time')
ylabel('error')
title('Prediction errors')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
**

% plot the initial membership functions
figure (60)
subplot(2,1,1)
[mfx, mfy]=plotmf(in_fismat, 'input', 1);
plot(mfx, mfy);
title('(a) Initial MFs on input')
axis([-inf inf -inf inf]);
subplot(2,1,2)
[mfx, mfy]=plotmf(in_fismat, 'input', 2);
plot(mfx, mfy);
title('(b) Initial MFs on input')
axis([-inf inf -inf inf]);

% plot final MF's on x,y,z,u

figure (70)
subplot(2,1,1)
[mfx, mfy]=plotmf(m_anfis, 'input', 1);
plot(mfx, mfy);
title('(a) Final MFs on input')
axis([-inf inf -inf inf]);
subplot(2,1,2)
[mfx, mfy]=plotmf(m_anfis, 'input', 2);

```

```

plot(mfx, mfy);
title('(b) Final MFs on input')
axis([-inf inf -inf inf]);

```

```

%%%%%%%% Plotting TRAINING data as a scater plot%%%%%%%%
figure(80)
subplot(1,2,1)
plot (train, train2, 'o')
xlabel ('input (k-2) ')
ylabel('ouput (k)')
title('Training data')
axis equal; axis square

```

```

subplot(1,2,2)
plot (train1, train2, 'o')
xlabel ('input (k-1) ')
ylabel('ouput (k)')
title('Training data')
axis equal; axis square

```

```

%%%%%%%% Plotting EVALUATION data as a scater plot%%%%%%%%
figure(90)
subplot(1,2,1)
plot (eval, eval2, 'o')
xlabel ('input (k-2)')
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square

```

```

subplot(1,2,2)
plot (eval1, eval2, 'o')
xlabel ('input (k-1)')
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square

```

```

figure(100)
plotfis(in_fismat)

```

```

figure(110)
plotfis(m_anfis)

```

```

showrule(m_anfis)

```

```

anfisedit (m_anfis)
surfview(m_anfis)

```



```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error measures calculation

MSE_anfis=(1/length(yhat_anfis))*norm(er_anfis)^2

%Root Mean Square Error (RMSE)
RMSE_anfis=sqrt(norm(er_anfis)^2/length(er_anfis))

%Mean Absolute Error (MAE)
MAE_anfis=(1/length(er_anfis))*sum(abs(er_anfis))

%Mean Absolute percentage Error (MAPE)
MAPE_anfis=(100/length(er_anfis))*sum(abs(er_anfis)./abs(adapt_input))

toc % the calculation time in seconds
runing_minute_time=toc/60
%end

```

% PREDICTION USING ---ANFIS3--- MODEL

```
close all %clean the workspace
clear
clc
```

```
mydata=xlsread ('27738-WT1fix', 'e2:e366') % retrieves the E column data of range 1
to 252 from the data file 'atermondata'
```

```
%load mydata
tic
figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')
```

```
% Estimation of an ANFIS model
% prepare training data
%input data
tr=mydata(1:250) % TRAINING DATA
% input (k-2)
train=tr
train(length(train))=[] % removes the last row
train(length(train))=[]; %removes the second last row
length(train)
```

```
%input k-1
```

```
train1=tr % first input
train1(length(train1))=[] % removes the last row
train1(1)=[] % removes the first row
length(train1)
```

```
% output k
train2=tr %second input
train2(1)=[] % removes the first row
train2(1)=[] %removes the second row
length(train2)
```

```
trn_data=[train train1 train2 ] %(k-2) (k-1) (k) training data
```

```
%preparing the evaluation(test) data
ev=mydata(251:end)% TESTING DATA
```

```
% input (k-2)
```

```

eval=ev
eval(length(eval))=[]; eval(length(eval))=[];
length(eval)

% input (k-1)
eval1=ev
eval1(length(eval))=[];
eval1(1)=[];
length(eval)

% input (k)
eval2=ev
eval2(1)=[]
eval2(1)=[];
length(eval2)

evaldata=[eval eval1] %input (k-2) and (k-1)

y2=eval2 % (k) output data for testing


% generate FIS matrix
epoch_n=300
mf_n=2;

% mf_type='gbellmf'; %type of membership function
% mf_type='trimf'; %the parameter b>c
% mf_type='gauss2mf'
mf_type='gaussmf' %
% mf_type='smf' % unsupported
% mf_type='trapmf' % the parameter b>c
% mf_type='zmf' % unsupported
% mf_type='pimf' % run problem
in_fisimat=genfis1(trn_data, mf_n, mf_type);

% start training
ss=0.1;

[m_anfis trn_error step_size ] = ...
    anfis(trn_data, in_fisimat, [epoch_n nan ss nan nan], [1,1,1,1]);

figure('name',...
    ['ANFIS: time series prediction (file: ' mfilename ')'],...

```

```

    'number', 'off');

subplot(211);
tmp=[trn_error ];
plot(tmp);
title('Error Curves');
axis([0 epoch_n min(tmp(:)) max(tmp(:))]);
xlabel('epochs')
ylabel('RMSE')
legend('Training Error');

subplot(212);
plot(step_size);
xlabel('epochs')
title('Step Size');

input=evaldata
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%prediction by ANFIS model out of sample

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ANFIS EVALUATION
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
yhat_anfis=evalfis(input, m_anfis);

adapt_input=y2;
length(adapt_input);
length(yhat_anfis);
result=[adapt_input yhat_anfis (adapt_input-yhat_anfis)];
er_anfis=adapt_input-yhat_anfis; %error

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure (20)
plot(adapt_input(end-30:end), 'b-s'), hold, plot(yhat_anfis(end-30:end), 'r-x');
legend('actual values','ANFIS prediction values')
xlabel('time')
ylabel('values')
title('Actual values and ANFIS prediction')

figure (30)
plot(adapt_input-yhat_anfis)
xlabel('time')
ylabel('error')

```

```
title('Prediction errors')
```

```
%%%%%%%%%*****  
**
```

```
% plot the initial membership functions  
figure (60)  
subplot(2,1,1)  
[mfx, mfy]=plotmf(in_fismat, 'input', 1);  
plot(mfx, mfy);  
title('(a) Initial MFs on input')  
axis([-inf inf -inf inf]);  
subplot(2,1,2)  
[mfx, mfy]=plotmf(in_fismat, 'input', 2);  
plot(mfx, mfy);  
title('(b) Initial MFs on input')  
axis([-inf inf -inf inf]);
```

```
% plot final MF's on x,y,z,u
```

```
figure (70)  
subplot(2,1,1)  
[mfx, mfy]=plotmf(m_anfis, 'input', 1);  
plot(mfx, mfy);  
title('(a) Final MFs on input')  
axis([-inf inf -inf inf]);  
subplot(2,1,2)  
[mfx, mfy]=plotmf(m_anfis, 'input', 2);  
plot(mfx, mfy);  
title('(b) Final MFs on input')  
axis([-inf inf -inf inf]);
```

```
%%%%%%%%% Ploting TRAINING data as a scater plot%%%%%%%%%  
figure(80)  
subplot(1,2,1)  
plot (train, train2, 'o')  
xlabel ('input (k-2) ' )  
ylabel('ouput (k)')  
title('Training data')  
axis equal; axis square  
  
subplot(1,2,2)  
plot (train1, train2, 'o')  
xlabel ('input (k-1) ' )
```

```

ylabel('ouput (k)')
title('Training data')
axis equal; axis square

%%%%%% Ploting EVALUATION data as a scater plot%%%%%%%%
figure(90)
subplot(1,2,1)
plot (eval, eval2, 'o')
xlabel ('input (k-2)')
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square

subplot(1,2,2)
plot (eval1, eval2, 'o')
xlabel ('input (k-1)')
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square

figure(100)
plotfis(in_fismat)

figure(110)
plotfis(m_anfis)

showrule(m_anfis)

anfisedit (m_anfis)
surfview(m_anfis)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error measures calculation

MSE_anfis=(1/length(yhat_anfis))*norm(er_anfis)^2

%Root Mean Square Error (RMSE)
RMSE_anfis=sqrt(norm(er_anfis)^2/length(er_anfis))

%Mean Absolute Error (MAE)
MAE_anfis=(1/length(er_anfis))*sum(abs(er_anfis))

%Mean Absolute percentage Error (MAPE)
MAPE_anfis=(100/length(er_anfis))*sum(abs(er_anfis)./abs(adapt_input))

toc % the calculation time in seconds

```

```
runing_minute_time=toc/60  
%end
```

% PREDICTION USING ---ANFIS4--- MODEL

```
close all %clean the workspace
clear
clc
```

```
mydata=xlsread ('27738-WT1fix', 'e2:e366') % retrieves the E column data of range 1
to 252 from the data file 'atermonddata'
```

```
%load mydata
tic
figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')
```

```
% Estimation of an ANFIS model
% prepare training data
%input data
tr=mydata(1:250) % TRAINING DATA
```

```
% input (k-3)
train=tr;
train(length(train))=[] % removes the last row
train(length(train))=[]; %removes the second last row
train(length(train))=[];
length(train)
```

```
% input (k-2)
train1=tr;
train1(length(train1))=[];% removes the last row
train1(length(train1))=[]; %removes the second last row
train1(1)=[]; % removes the first row
length(train1)
```

```
%input k-1
```

```
train2=tr; % first input
train2(length(train2))=[];% removes the last row
train2(1)=[];% removes the first row
train2(1)=[];
length(train2)
```

```
% output k
train3=tr; %second input
train3(1)=[]; % removes the first row
```



```

train3(1)=[]; %removes the second row
train3(1)=[];
length(train3)

trn_data=[train train1 train3] %(k-2) (k-1) (k) training data

%preparing the evaluation(test) data
ev=mydata(251:end)% TESTING DATA

% input (k-3)
eval=ev
eval(length(eval))=[]; eval(length(eval))=[];eval(length(eval))=[];
length(eval)
% input (k-2)
eval1=ev
eval1(length(eval1))=[]; eval1(length(eval1))=[];eval1(1)=[];
length(eval1)

% input (k-1)
eval2=ev
eval2(length(eval2))=[];
eval2(1)=[];eval2(1)=[];
length(eval2)

% input (k)
eval3=ev
eval3(1)=[];
eval3(1)=[];
eval3(1)=[];
length(eval3)

evaldata=[eval eval2] %input (k-2) and (k-1)

y2=eval3 %(k) output data for testing


% generate FIS matrix
epoch_n=300
mf_n=2;

%mf_type='gbellmf'; %type of membership function
%mf_type='trimf'; %the parameter b>c
%mf_type='gauss2mf'

```

```

mf_type='gaussmf' %
%mf_type='smf' % unsupported
%mf_type='trapmf' % the parameter b>c
%mf_type='zmf' % unsupported
%mf_type='pimf' % run problem
in_fismat=genfis1(trn_data, mf_n, mf_type);

% start training
ss=0.1;

[m_anfis trn_error step_size ] = ...
    anfis(trn_data, in_fismat, [epoch_n nan ss nan nan], [1,1,1,1]);

figure('name',...
    ['ANFIS: time series prediction (file: ' mfilename ')'],...
    'number', 'off');

subplot(211);
tmp=[trn_error ];
plot(tmp);
title('Error Curves');
axis([0 epoch_n min(tmp(:)) max(tmp(:))]);
xlabel('epochs')
ylabel('RMSE')
legend('Training Error');

subplot(212);
plot(step_size);
xlabel('epochs')
title('Step Size');

input=evaldata
%%%%%%%%%%%%%%
%%%%%%%%%%%%%%

%prediction by ANFIS model out of sample

%%%%%%%%%%%%%% ANFIS EVALUATION
%%%%%%%%%%%%%%
yhat_anfis=evalfis(input, m_anfis);

adapt_input=y2;
length(adapt_input);

```

```

length(yhat_anfis);
result=[adapt_input yhat_anfis (adapt_input-yhat_anfis)];
er_anfis=adapt_input-yhat_anfis; %error

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure (20)
plot(adapt_input(end-30:end), 'b-s'), hold, plot(yhat_anfis(end-30:end), 'r-x');
legend('actual values','ANFIS prediction values')
xlabel('time')
ylabel('values')
title('Actual values and ANFIS prediction')

figure (30)
plot(adapt_input-yhat_anfis)
xlabel('time')
ylabel('error')
title('Prediction errors')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
**

% plot the initial membership functions
figure (60)
subplot(2,1,1)
[mfx, mfy]=plotmf(in_fismat, 'input', 1);
plot(mfx, mfy);
title('(a) Initial MFs on input')
axis([-inf inf -inf inf]);
subplot(2,1,2)
[mfx, mfy]=plotmf(in_fismat, 'input', 2);
plot(mfx, mfy);
title('(b) Initial MFs on input')
axis([-inf inf -inf inf]);

% plot final MF's on x,y,z,u

figure (70)
subplot(2,1,1)
[mfx, mfy]=plotmf(m_anfis, 'input', 1);
plot(mfx, mfy);
title('(a) Final MFs on input')
axis([-inf inf -inf inf]);
subplot(2,1,2)
[mfx, mfy]=plotmf(m_anfis, 'input', 2);

```

```

plot(mfx, mfy);
title('(b) Final MFs on input')
axis([-inf inf -inf inf]);

```

```

%%%%%%%% Plotting TRAINING data as a scater plot%%%%%%%%
figure(80)
subplot(1,2,1)
plot (train, train2, 'o')
xlabel ('input (k-2) ')
ylabel('ouput (k)')
title('Training data')
axis equal; axis square

```

```

subplot(1,2,2)
plot (train1, train2, 'o')
xlabel ('input (k-1) ')
ylabel('ouput (k)')
title('Training data')
axis equal; axis square

```

```

%%%%%%%% Plotting EVALUATION data as a scater plot%%%%%%%%
figure(90)
subplot(1,2,1)
plot (eval, eval2, 'o')
xlabel ('input (k-2)')
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square

```

```

subplot(1,2,2)
plot (eval1, eval2, 'o')
xlabel ('input (k-1)')
ylabel('ouput (k)')
title('Evaluating data')
axis equal; axis square

```

```

figure(100)
plotfis(in_fismat)

```

```

figure(110)
plotfis(m_anfis)

```

```

showrule(m_anfis)

```

```

anfisedit (m_anfis)
surfview(m_anfis)

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% error measures calculation

MSE_anfis=(1/length(yhat_anfis))*norm(er_anfis)^2

%Root Mean Square Error (RMSE)
RMSE_anfis=sqrt(norm(er_anfis)^2/length(er_anfis))

%Mean Absolute Error (MAE)
MAE_anfis=(1/length(er_anfis))*sum(abs(er_anfis))

%Mean Absolute percentage Error (MAPE)
MAPE_anfis=(100/length(er_anfis))*sum(abs(er_anfis)./abs(adapt_input))

toc % the calculation time in seconds
runing_minute_time=toc/60
%end

```

% PREDICTION USING ---AR--- MODEL

```
close all %clean the workspace
clear
clc
```

```
mydata=xlsread ('27738-WT1fix', 'e2:e366') % retrieves the E column data of range 1
to 252 from the data file 'atermonddata'
```

```
%load mydata
```

```
figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')
```

```
% Estimation of an AR model
```

```
y1=mydata(1:250) % TRAINING DATA
y2=mydata(251:end)% TESTING DATA
```

```
%%%%%%%%%%%%
%%%%%%%%%%%%
m_ar=ar(y1,1, 'ls') % estimation of the model
```

```
fpe_ar=fpe(m_ar) % estimation of Akaike's Final prediction Error (FPE)
aic_ar=aic(m_ar) % estimation of Akaike's Information criterion (AIC)
```

```
%%%%%%%%%%%%
%%%%%%%%%%%%
% in sample evaluation
yhat_ar_insample=predict(m_ar, y1,1) %predicts the in sample data
```

```
t1=1:(length(yhat_ar_insample))
figure(2)
plot(t1,y1,':', t1,yhat_ar_insample,'-');
legend('actual value',' ar prediction')
xlabel('time')
ylabel('value')
title('Actual and AR prediction - in sample')
```

```
figure (3) % in sample prediction and plotting
compare(y1, m_ar, 1)% it is another way to predict and plot directly
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%prediction by AR model out of sample
```

```
yhat_ar=predict(m_ar, y2,1) %predicts the unseen data
```

```
t=1:(length(yhat_ar))  
figure(4)  
plot(t,y2,'b-s', t,yhat_ar,'r-x');  
legend('actual value',' ar prediction')  
xlabel('time')  
ylabel('value')  
title('Actual and AR prediction -out of sample')
```

```
figure (5)  
compare(y2, m_ar, 1)% it is another way to predict and plot directly
```

```
[y2 yhat_ar]% prints the actual and predicted value
```

```
get(m_ar) % gives model information
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
% error measures calculation
```

```
% Mean Square Error (MSE)  
MSE_ar=(1/length(yhat_ar))*norm(y2-yhat_ar)^2
```

```
% Root Mean Square Error (RMSE)  
RMSE_ar=sqrt(norm(y2-yhat_ar)^2/length(y2-yhat_ar))
```

```
% Mean Absolute Error (MAE)  
MAE_ar=(1/length(y2-yhat_ar))*sum(abs(y2-yhat_ar))
```

```
% Mean Absolute Percentage Error (MAPE)  
MAPE_ar=(100/length(y2-yhat_ar))*sum(abs(y2-yhat_ar)./abs(y2))
```

```
%end
```

% PREDICTION USING ---ARMA--- MODEL

```
close all %clean the workspace
clear
clc
```

```
mydata=xlsread ('27738-WT1fix', 'e2:e366') % retrieves the E column data of range 1
to 252 from the data file 'atermondata'
```

```
%load mydata
```

```
figure(1) % a view of data
plot (mydata)
xlabel('time'); ylabel('values')
title('Actual values')
```

```
% Estimation of an ARMA model
```

```
y1=mydata(1:250) % TRAINING DATA
y2=mydata(251:end)% TESTING DATA
```

```
%%%%%%%%%%
%%%%%%%%%%
m_arma=arma(y1,1) % estimation of the model
```

```
fpe_arma=fpe(m_arma) % estimation of Akaike's Final prediction Error (FPE)
aic_arma=aic(m_arma) % estimation of Akaike's Information criterion (AIC)
```

```
%%%%%%%%%%
%%%%%%%%%%
```

```
% in sample evaluation
```

```
yhat_arma_insample=predict(m_arma, y1,1) %predicts the in sample data
```

```
t1=1:(length(yhat_arma_insample))
figure(2)
plot(t1,y1,':', t1,yhat_arma_insample,'-');
legend('actual value',' arma prediction')
xlabel('time')
ylabel('value')
title('Actual and ARMA prediction - in sample')
```

```
figure (3) % in sample prediction and plotting
```

```
compare(y1, m_arma, 1)% it is another way to predict and plot directly
```



```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%prediction by ARMA model out of sample
```

```
yhat_arma=predict(m_arma, y2,1) %predicts the unseen data
```

```
t=1:(length(yhat_arma))
```

```
figure(4)
```

```
plot(t,y2,'b-s', t,yhat_arma,'r-x');
```

```
legend('actual value',' arma prediction')
```

```
xlabel('time')
```

```
ylabel('value')
```

```
title('Actual and ARMA prediction -out of sample')
```

```
figure (5)
```

```
compare(y2, m_arma, 1)% it is another way to predict and plot directly
```

```
[y2 yhat_arma]% prints the actual and predicted value
```

```
get(m_arma) % gives model information
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% error measures calculation
```

```
% Mean Square Error (MSE)
```

```
MSE_arma=(1/length(yhat_arma))*norm(y2-yhat_arma)^2
```

```
% Root Mean Square Error (RMSE)
```

```
RMSE_arma=sqrt(norm(y2-yhat_arma)^2/length(y2-yhat_arma))
```

```
% Mean Absolute Error (MAE)
```

```
MAE_arma=(1/length(y2-yhat_arma))*sum(abs(y2-yhat_arma))
```

```
% Mean Absolute Percentage Error (MAPE)
```

```
MAPE_arma=(100/length(y2-yhat_arma))*sum(abs(y2-yhat_arma)./abs(y2))
```

```
%end
```

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Atsalakis, G., Wind energy production forecasting by Neural networks and Genetic Algorithms, *Working paper*, Technical University of Crete, March, (2007).
- [2] Atsalakis, G., and C. Ucenic, Electric load forecasting by Neuro-fuzzy approach, *WSEAS International Conference on Energy and Environmental Systems*, Chalkida, Evia, Greece, (2006).
- [3] Atsalakis, G., and C. Ucenic, Forecasting the Electricity Demand Using a Neuro-fuzzy Approach Versus Traditions Methods, *Journal of WSEAS Transactions on Business and Economics*, issue 1, vol. 3, pp. 9-17, (2006).
- [4] Atsalakis, G., and C. Ucenic, Forecasting the wind energy production using a Neuro-fuzzy model, *Journal of WSEAS Transactions on Environment and Development*, vol. 2, (6) pp. 823-829, (2006).
- [5] Atsalakis, G., C. Ucenic, and G. Plokamakis, Forecasting of Electricity Demand Using Neuro-fuzzy (ANFIS) Approach, *International Conference on NHIBE*, Corfu, Greece, (2005).
- [6] Bailey, B., M. C. Brower, and J. Zack, Short-Term Wind Forecasting, *Proceedings of the European Wind Energy Conference*, Nice, France, pp. 1062-1065, (1999).
- [7] Beyer, H.G., D. Heinemann, H. Mellinghoff, K. Mönnich, and H.-P. Waldl: Forecast of Regional Power Output of Wind Turbines, *Proceedings of the European Wind Energy Conference*, Nice, France, pp. 1070-1073, (1999).
- [8] Box, J., and G. Jenkins, *Time series analysis, forecasting and control*, San Francisco: Holden-Day, (1976).
- [9] Connor, J., Martin R and L. Atlas, Recurrent neural networks and robust time series prediction, *IEEE Trans Neural Networks*, vol. 5, (2), pp240-253, (1994).
- [10] Durstewitz, M., C. Ensslin, B. Hahn and M. Hoppe-Kilpper, Annual Evaluation of the Scientific Measurement and Evaluation Programme (WMEP), *Working paper*, Kassel, (2001).
- [11] Ernst, B., K. Rohrig, H. Regber and Dr. P. Schorn, Managing 3000 MW Wind Power in a Transmission System Operation Center, *Proceedings of the European Wind Energy Conference*, Copenhagen, Denmark, pp. 890-893, (2001).
- [12] Focken, U., M. Lange and H.-P. Waldl, Previento, A Wind Power Prediction System With an Innovative Upscaling Algorithm, *Proceedings of the European Wind Energy Conference*, Copenhagen, Denmark, pp. 826-829, (2001).

- [13] Gibel G., On the benefits of distributed generation of wind energy in Europe, *Working paper*, University of Oldenburg, pp. 1-2, (2000).
- [14] Gibel G., Utility wind- modeling planning meeting, *Working paper*, National Renewable Energy Laboratory, Golden, Colorado, USA, pp. 22-23, (1996).
- [15] Ho, S. L., M. Xie, and T.N. Goh, A comparative study of neural network and Box Jenkins ARIMA modeling in time series prediction, *Computers and Industrial Engineering*, Elsevier science, vol. 42, pp. 371-375, (2002).
- [16] Hush, D., and B. Horne, Progress in supervised neural networks, what is new since Lipmann?, *IEEE Signal processing magazine*, vol. 10, (1), pp. 8-39, (1993).
- [17] Jang, JS., ANFIS: Adaptive –Network –based Fuzzy Inference System, *IEEE Trans systems, Man and Cybernetics*, vol. 23, (3), pp. 665-684, (1993).
- [18] Jang, JS., and C. Sun, Predicting chaotic time series with fuzzy if-then rules, *Proceedings of the IEEE international conference on fuzzy systems*, pp. 1079-1084, (1993).
- [19] Jørgensen, J., C. Moehrlen, B., Ó. Gallaghóir, K. Sattler, and E. McKeogh, HIRPOM: Description of an operational numerical wind power prediction model for large scale integration of on- and offshore wind power in Denmark, *Poster on the Global Windpower Conference and Exhibition*, Paris, France, vol. 5, pp. (2002).
- [20] Kalogirou, SA., Applications of artificial neural networks for energy systems, *Applied Energy*, vol. 67, pp. 17–35, (2000).
- [21] Landberg, L., Short-term Prediction of Local Wind Conditions, *PhD-Thesis*, Risø-R-702 (EN), Risø National Laboratory, Roskilde, Denmark, (1994).
- [22] Martí Perez, I., Wind Forecasting Activities, *Proceedings of the First IEA Joint Action Symposium on Wind Forecasting Techniques*, Published by FOI - Swedish Defence Research Agency Norrköping, Sweden, pp. 11-20, (2002).
- [23] Mönnich.K., Vorhersage der Leistungsabgabe netzeinspeisender windkraftanlagen zur Unterstützung der Kraftwerkseinsatzplanung, *PhD-thesis*, Carl von Ossietzky Universität, Oldenburg, (2000).
- [24] Morales, G., and G. Sipreólico, Wind power prediction experience, Talk slides accompanied by the paper: Sánchez, I., J. Usaola, O. Ravelo, C. Velasco, J. Domínguez, M.G. Lobo, G. González, F. Soto, B. Díaz-Guerra and M. Alonso: Sipreólico, A wind power prediction system based on flexible combination of dynamic models, Application to the Spanish power system, *Proceedings of the First IEA Joint Action Symposium on Wind Forecasting Techniques*, Published by FOI - Swedish Defence Research Agency, Norrköping, Sweden, pp. 197-214, (2002).
- [25] Nielsen, T.S., H. Madsen, and J. Tøfting, Experiences with Statistical Methods for Wind Power Prediction, *Proceedings of the European Wind Energy Conference*, Nice, France, pp. 1066-1069, (1999).

- [26] Rnaweera, D., N. Hubele, and A. Papalexopoulos, Application of radial basis function network model for short term load forecasting, *IEEE Proceedings Generation Transmission Distribution*, vol. 142, (1), (1995).
- [27] Sfetsos, A., A comparison of various forecasting techniques applied to mean hourly wind speed time series, Renewable energy, *Published by Pergamon*, vol. 21, pp. 24, (1999).
- [28] Zadeh, L. A., Fuzzy sets, *Information and Control*, vol. 8, pp. 338-353, (1965).

ΤΕΛΟΣ