



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

Συστήματα Υπολογιστικής Ευφυΐας

Μοντελοποίηση Πραγματικού Συστήματος Με Χρήση Προσαρμοστικού Νευροασαφούς Συστήματος

1η Εργασία
Εμμανουήλ Παπαδημητρίου
AM: mcse19021

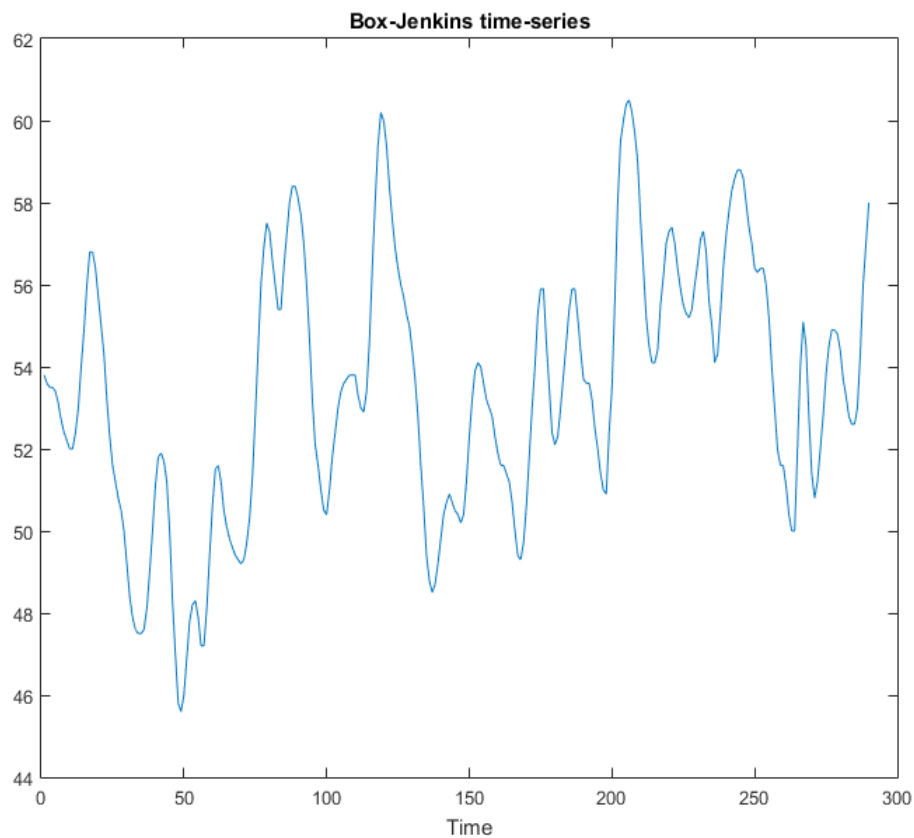
Μεταπτυχιακό Πρόγραμμα "Επιστήμη και Τεχνολογία της Πληροφορικής και των Υπολογιστών"

2 Inputs

Θα αναλυθούν οι τρεις περιπτώσεις που αφορούν τις δύο εισόδους:

- 2 inputs
- 2 inputs no feedback
- 2 feedback inputs

Αρχικά έχουμε το Box-Jenkins γράφημα, που και στις τρεις περιπτώσεις είναι το ίδιο. Αυτό συμβαίνει καθώς το αποτέλεσμα που ελέγχεται στην τελευταία στήλη του **BJ_data.dat**, είναι κοινό για όλα. Είναι το αποτέλεσμα που ζητάμε.



Οι πρώτες δοκιμές γίνανε με τις default τιμές που δόθηκαν:

- fuzzySets = 2;
- epoch_number = 50;
- initial_learning_rate = 0.1;
- mfType = 'gauss2mf';
- outType = 'linear';

Αρχικά παρατηρούμε το training error:

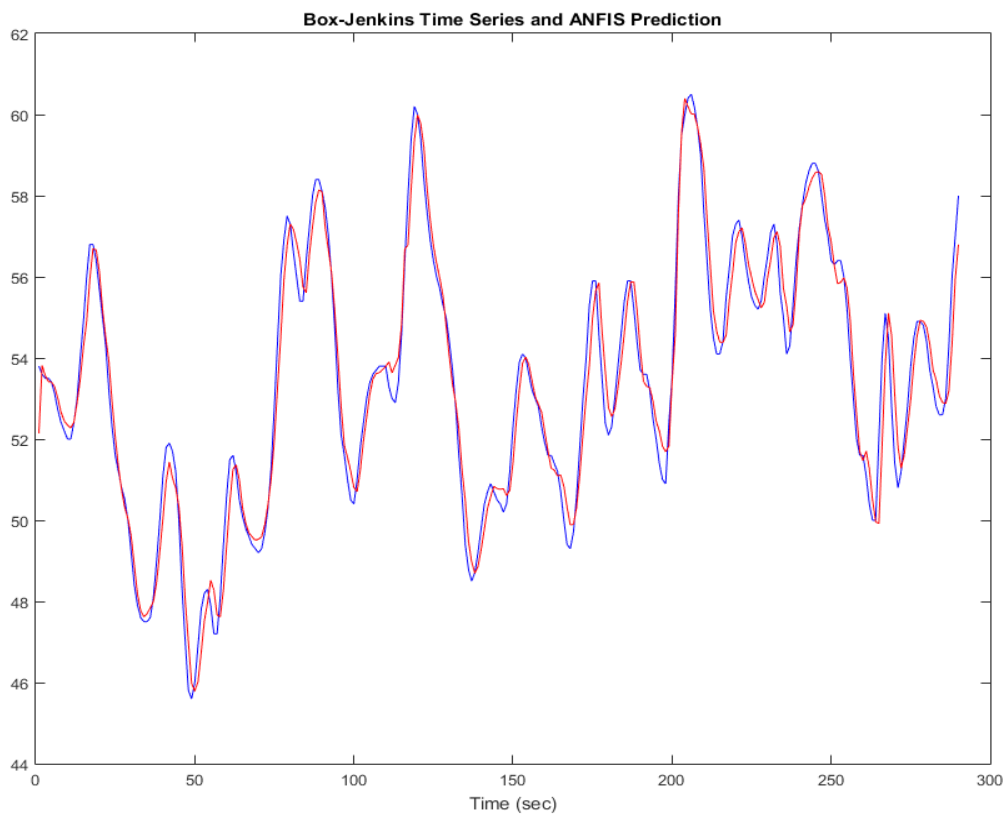
- Για 2 inputs βλέπουμε ότι κυμαίνεται από ~ 0.652 έως ~ 0.640
- Για 2 input no feedback βλέπουμε ότι κυμαίνεται από ~ 2.240 έως ~ 2.205
- Για 2 feedback input κυμαίνεται από ~ 0.406 έως ~ 0.392

Το μικρότερο error το έχει το input με feedback. Είναι λογικό καθώς οι δύο είσοδοι περιέχουν feedback, αυτό σημαίνει τιμές πιο κοντά στις αναμενόμενες.

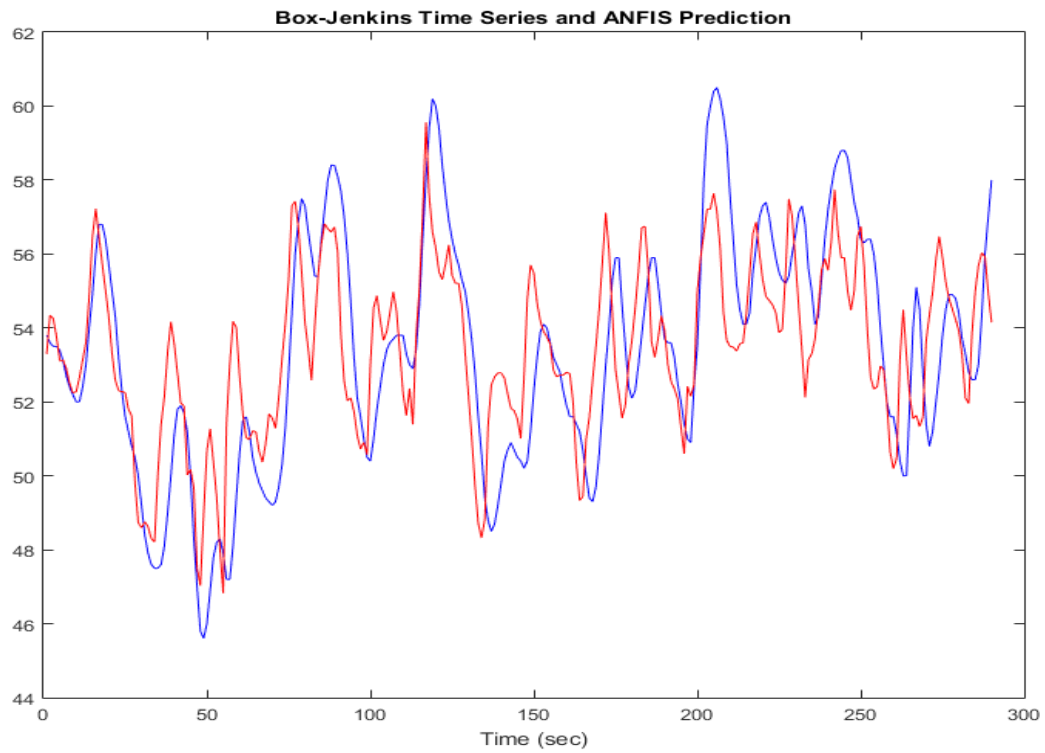
Το μεγαλύτερο error το έχει το input χωρίς feedback, καθώς οι είσοδοι σε αυτήν την περίπτωση έχουν μεγαλύτερη διαφορά με τις αναμενόμενες.

Στη συνέχεια θα μελετήσουμε τα αποτελέσματα από το ANFIS με τις πραγματικές εξόδους

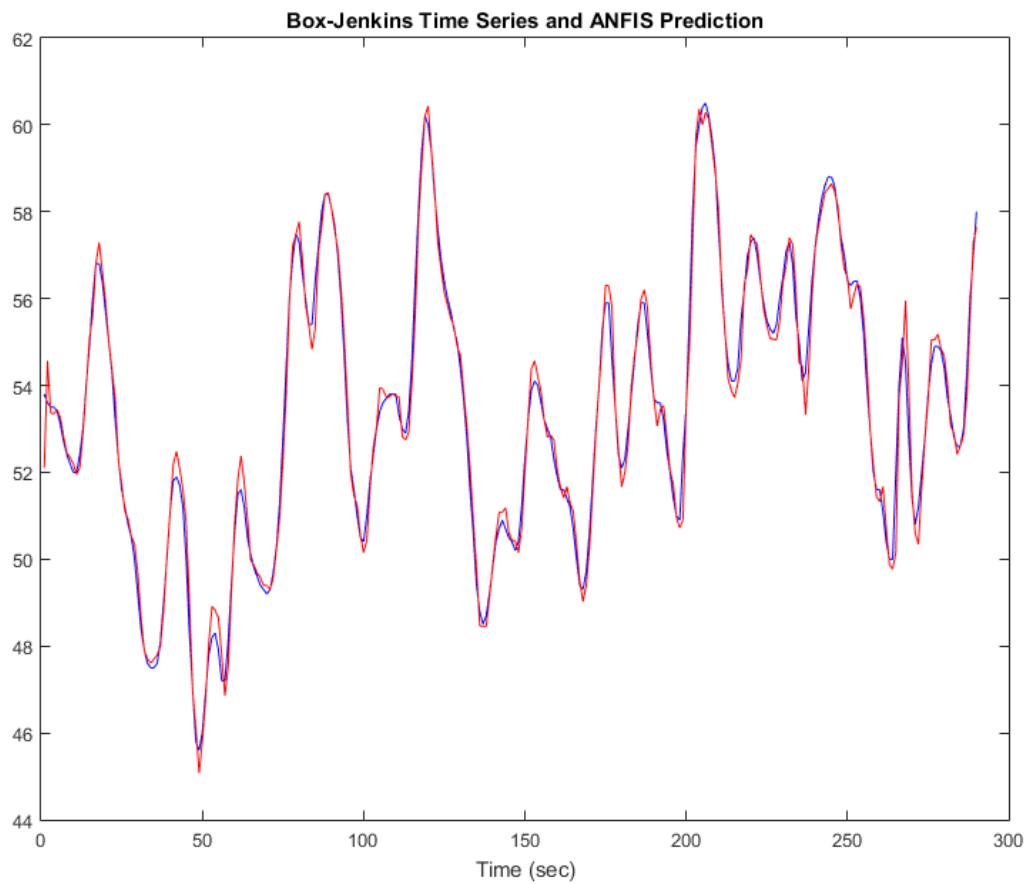
2 inputs



2 inputs no feedback



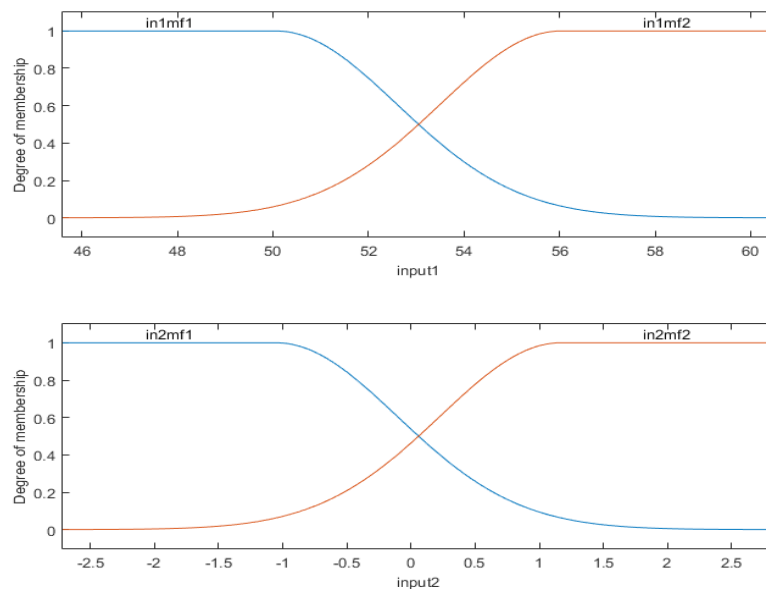
2 feedback inputs



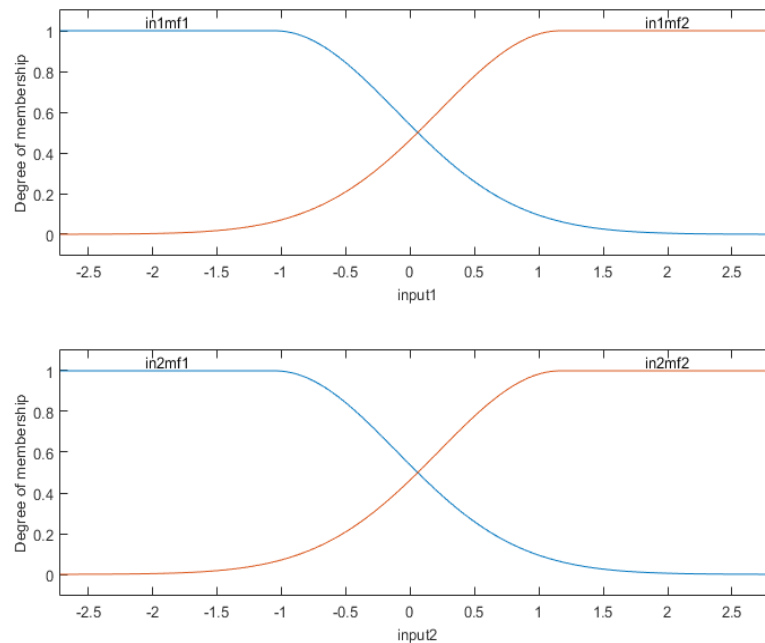
Όπως μας φανέρωσαν και οι τιμές του training error σε κάθε περίπτωση, έτσι και στα γραφήματα σύγκρισης των αποτελεσμάτων του ANFIS με τις πραγματικές εξόδους, ότι η τρίτη περίπτωση (2 feedback inputs) πλησιάζει πιο πολύ στις πραγματικές τιμές. Ενώ, είναι εμφανές ότι η περίπτωση των δύο εισόδων χωρίς feedback, απέχει σε πάρα πολλά σημεία από τις πραγματικές τιμές.

Παρακάτω βλέπουμε τα γραφήματα για την παραγωγή ενός ασαφούς interface που θα χρησιμοποιηθεί για την εκπαίδευση του ANFIS.

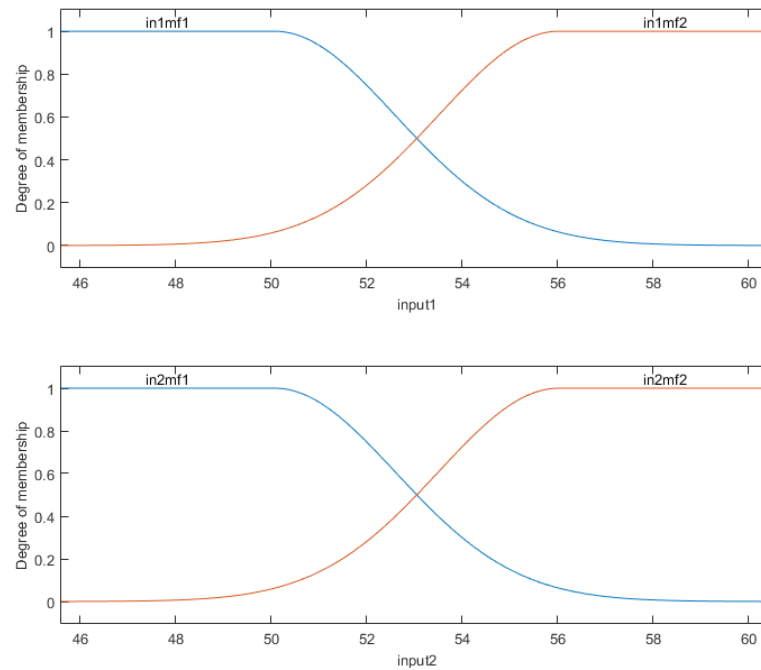
2 inputs



2 inputs no feedback



2 feedback inputs



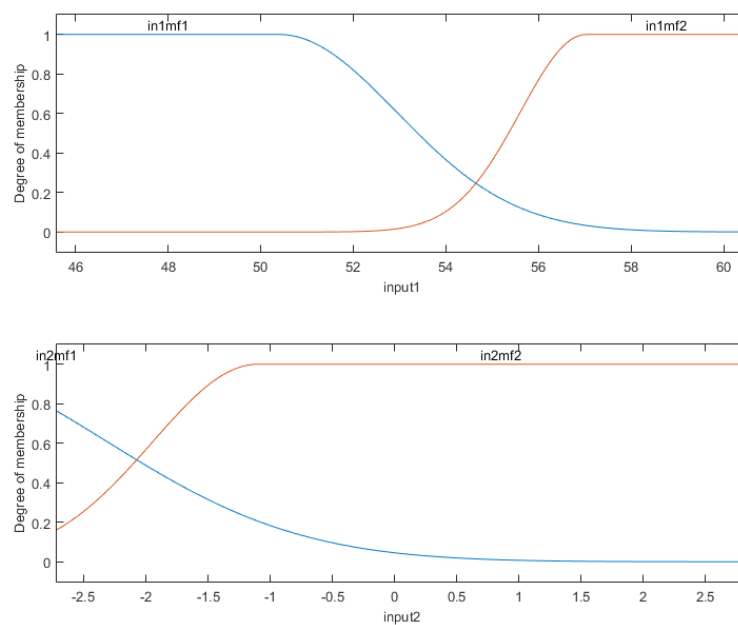
Στην δημιουργία του ασαφούς νευρωνικού δικτύου έχουν καθοριστικό ρόλο οι παρακάτω μεταβλητές:

- fuzzySets
- mfType
- outType

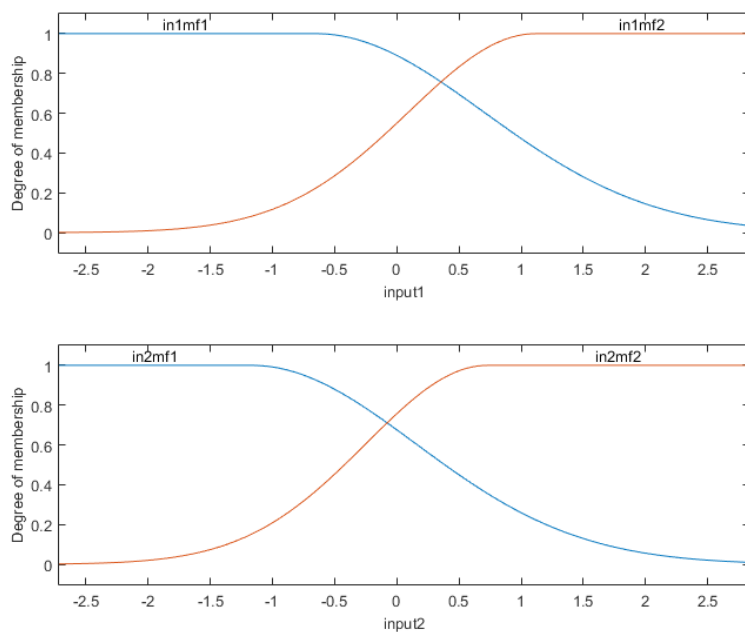
Όπως φαίνεται από τα γραφήματα, έχει γίνει η χρήση δύο ασαφή συνόλων (fuzzySets) για κάθε είσοδο. Ακόμη, με το είδος συνάρτησης συμμετοχής σαν `gaus2mf` (mfType) δηλώνουμε ότι χρειαζόμαστε σαν αποτέλεσμα μια γκαουσιανή ασαφής συνάρτηση συμμετοχής δύο συνδυασμένων γκαουσιανών. Τέλος, έχει οριστεί ως γραμμική (linear) το είδος εξόδου του κανόνα.

Παρουσιάζονται παρακάτω, τα γραφήματα των αποτελεσμάτων του αλγόριθμου εκπαίδευσης ANFIS.

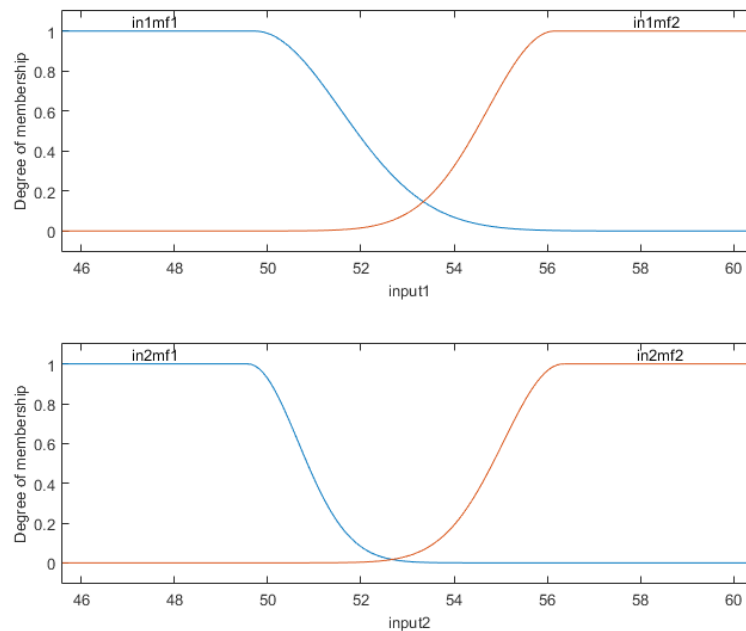
2 inputs



2 inputs no feedback



2 feedback inputs



Στον αλγόριθμο εκπαίδευσης ANFIS οι καθοριστικές μεταβλητές είναι:

- `trnData`: Περιέχει τα δεδομένα του αρχείου `BJ_data.dat`
- `fismat`: Ασαφές Interface
- `epoch_number`: Αριθμός επαναλήψεων εκπαίδευσης
- `initial_learning_rate`: Αρχικός ρυθμός εκμάθησης

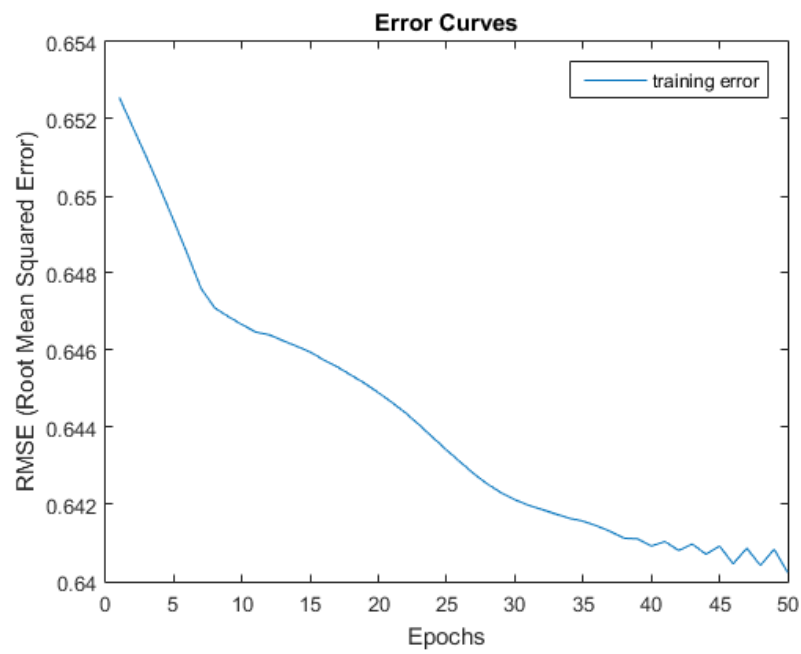
Οι δύο μεταβλητές που συναντάμε τώρα είναι ο αριθμός επαναλήψεων (`epoch_number`) και ο αρχικός ρυθμός εκμάθησης (`initial_learning_rate`).

Ο αριθμός επαναλήψεων σε αυτήν την περίπτωση έχει ορισθεί ως 50, το οποίο είναι εμφανές και στην εμφάνιση του `training error` που έχει 50 αποτελέσματα, κάθε ένα για κάθε επανάληψη. Με κάθε επανάληψη, το `training error` μειώνεται και το σύστημα βελτιώνεται. Άρα είναι εμφανές, ότι όσες περισσότερες επαναλήψεις έχουμε τόσο πιο βέλτιστα αποτελέσματα θα έχουμε. Όμως, το μεγαλύτερο πλήθος επαναλήψεων χρειάζεται και περισσότερους πόρους από το υπολογιστικό σύστημα που εκτελείται η εκπαίδευση. Είναι πολύ σημαντικός παράγοντας αυτός για το αν θα αποδεχθούμε τον αριθμό αυτόν των επαναλήψεων. Άλλος ένας παράγοντας είναι, το πόσο βέλτιστα αποτελέσματα έχουμε με τις επαναλήψεις. Ένας κανόνας είναι, όταν φτάσουμε σε συνεχή αποτελέσματα με ίδιες τιμές τότε σημαίνει ότι πρέπει να σταματήσουμε την αύξηση των αριθμών των επαναλήψεων.

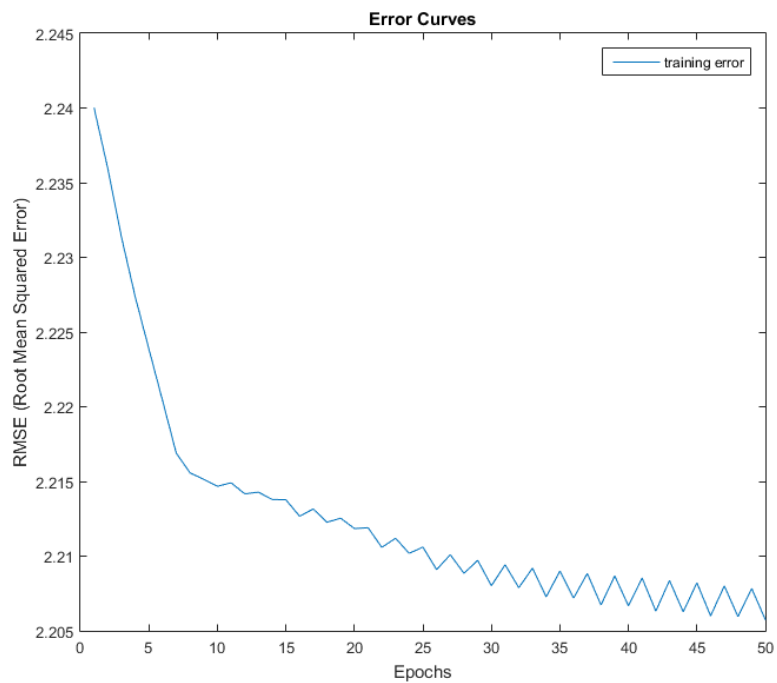
Ο αρχικός ρυθμός εκμάθησης έχει ορισθεί ως 0.1. Όσο πιο μικρά είναι τα βήματα, τόσο περισσότερο χρόνο θα χρειαστεί για να ολοκληρωθεί η εκπαίδευση.

Θα παρουσιαστούν τα διαγράμματα για το training error.

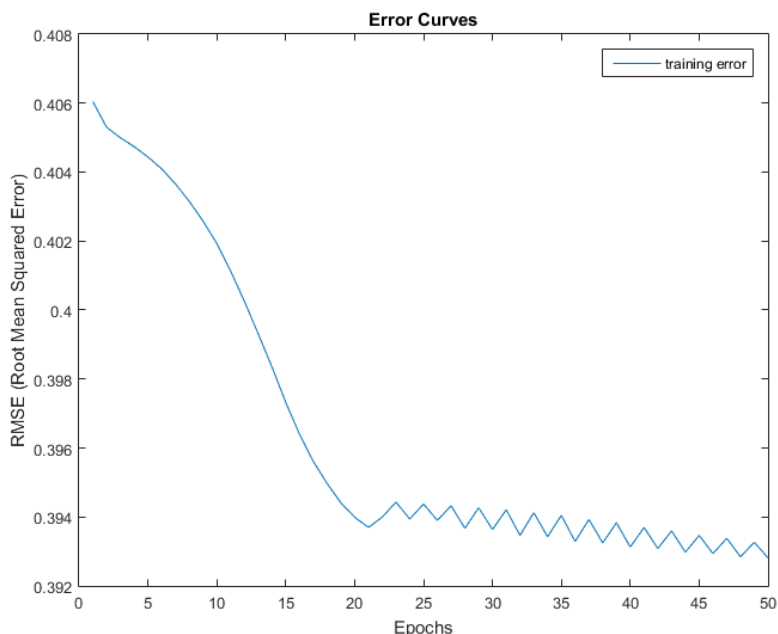
2 inputs



2 inputs no feedback



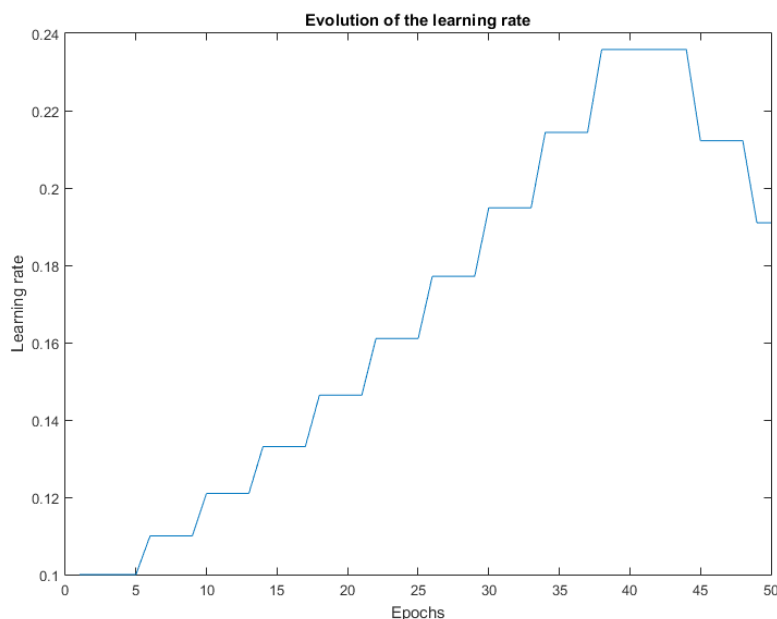
2 feedback inputs



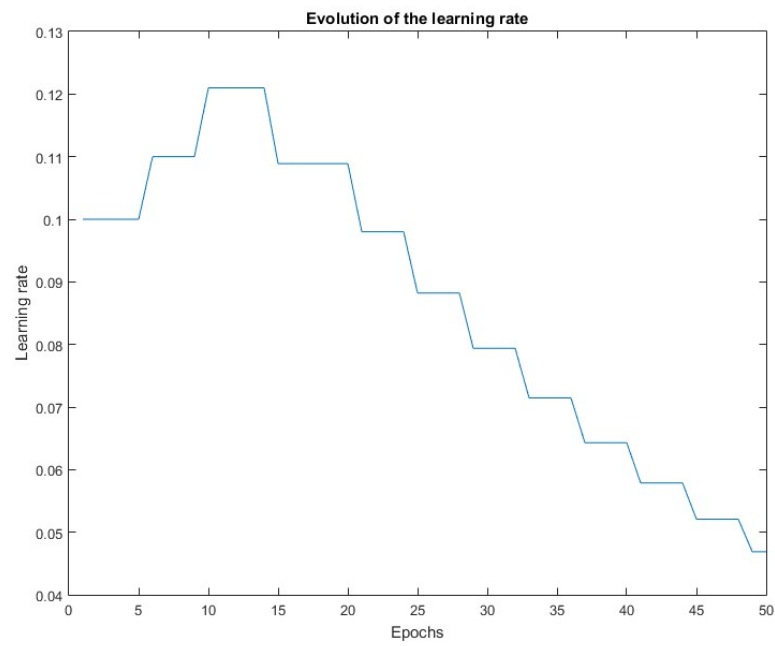
Το διάνυσμα `training_error` το λαμβάνουμε ως αποτέλεσμα από τον αλγόριθμο εκπαίδευσης ANFIS. Είναι το μέσο τετραγωνικό σφάλμα για κάθε επανάληψη εκπαίδευσης. Όπως βλέπουμε, με τις τιμές αυτές, η καμπύλη μέχρι ένα σημείο μειώνεται και στην συνέχεια σχηματίζει κορυφές. Αυτό συμβαίνει γιατί ίσως τα δεδομένα να μην χαρακτηρίζουν επαρκώς τα κυριότερα χαρακτηριστικά των μη εκπαιδευμένων δεδομένων λόγω σφάλματος μέτρησης, παρεμβολών κτλ. Η εξομάλυνση του σφάλματος πετυχαίνεται με την αύξηση του αριθμού επανάληψης εκπαίδευσης και τη μείωση του αρχικού ρυθμού εκμάθησης.

Θα παρουσιαστούν τα διαγράμματα για το learning rate.

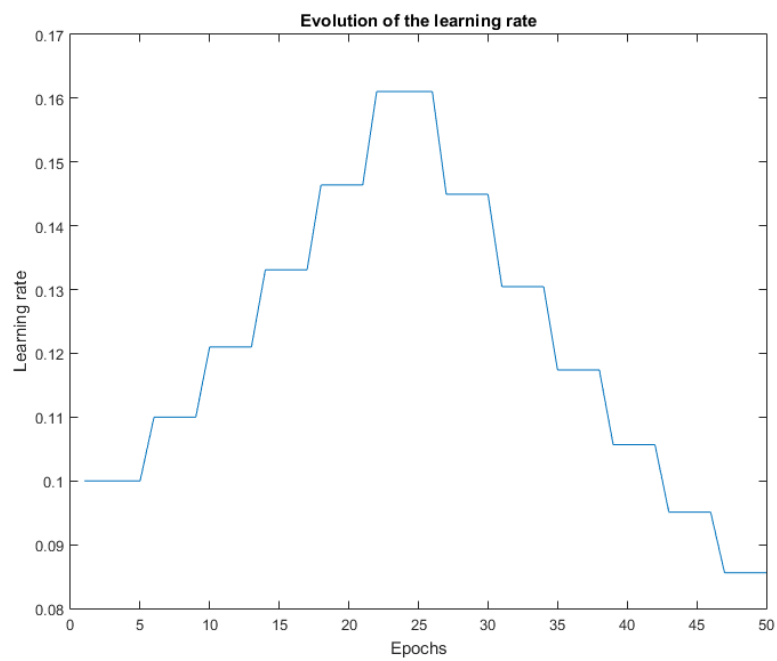
2 inputs



2 inputs no feedback



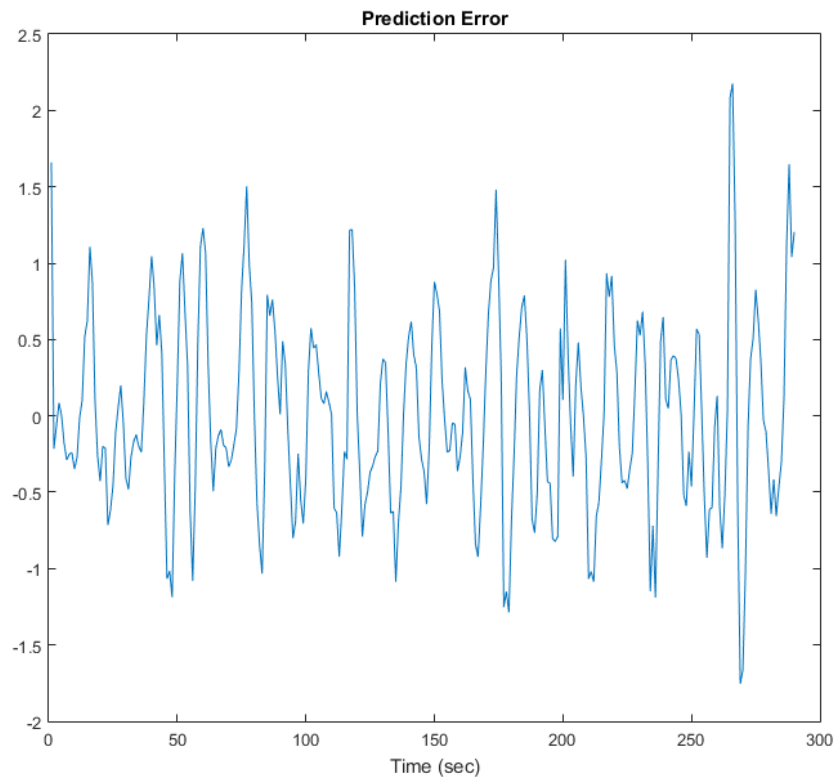
2 feedback inputs



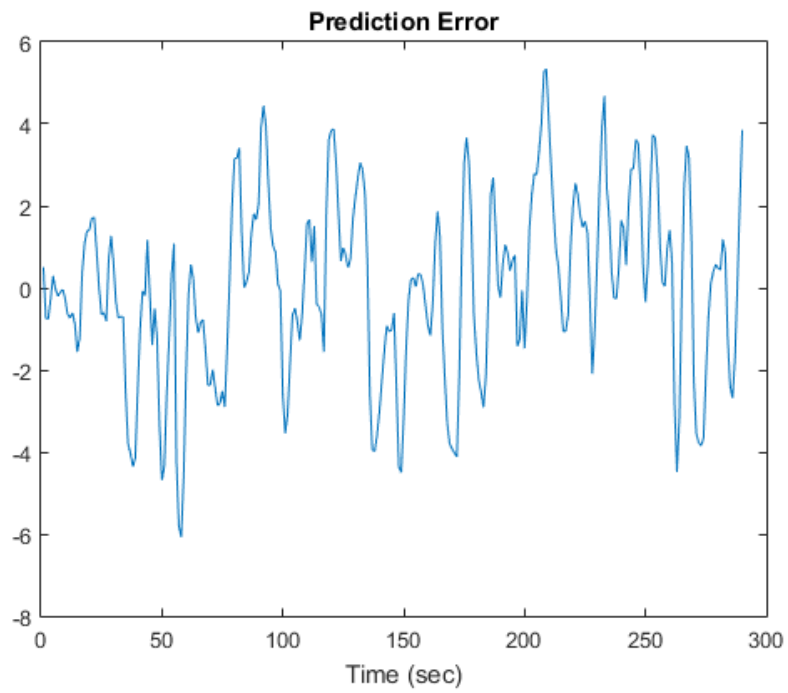
Το διάνυσμα `learning_rate` το λαμβάνουμε ως αποτέλεσμα από τον αλγόριθμο εκπαίδευσης ANFIS. Είναι το μέγεθος του βήματος εκπαίδευσης για κάθε αριθμό επαναλήψεων εκπαίδευσης. Αρχικά, το βήμα αυξάνεται και μόλις φτάσει στο μέγιστο, αρχίζει και μειώνεται για το υπόλοιπο της εκπαίδευσης. Αυτό είναι εμφανές από τα διαγράμματα, με αρχικό βήμα το 0.1, αυξάνεται το βήμα όπως βλέπουμε την καμπύλη, και στην συνέχεια φτάνει ένα μέγιστο και από εκεί αρχίζει και μειώνεται το βήμα μέχρι το πέρας των επαναλήψεων. Το να φτάσει το μέγιστο δεν συμβαίνει πάντα, εξαρτάται από το βήμα που έχουμε θέσει και των αριθμό των επαναλήψεων εκπαίδευσης. Για να αποδεχθούμε την τιμή που θέσαμε ως αρχικό ρυθμό εκμάθησης (`initial_learning_rate`), πρέπει το `learning rate` να αυξάνεται, να βρίσκει το μέγιστο και να μειώνεται, αν δεν το κάνει αυτό σημαίνει ότι δεν είναι το ιδανικό.

Θα παρουσιαστούν τα διαγράμματα για το `learning rate`.

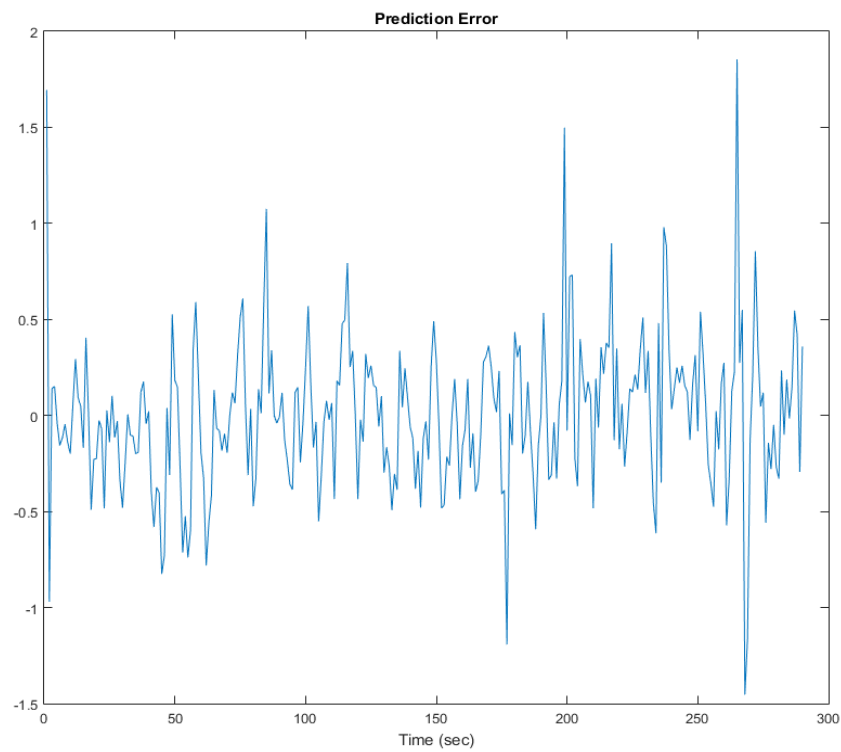
2 inputs



2 inputs no feedback



2 feedback inputs



Στα διαγράμματα αυτά βλέπουμε την διαφορά των πραγματικών τιμών με των τιμών που βγήκαν από τον αλγόριθμο εκπαίδευσης FIS. Όσο μικρότερος είναι ο άξονας των y, τόσο καλύτερες τιμές έχουμε. Όπως φαίνεται στο διάγραμμα 3 του 2 feedback inputs. Το ιδανικό θα ήταν μια συνεχής γραμμή.

Δοκιμές με τιμές

Δομή Ασαφούς Νευρωνικού Δικτύου

Θα μελετηθεί η επίδραση των μεταβλητών του της δομής του νευρωνικού δικτύου μετά από δοκιμές, οι μεταβλητές αυτές είναι:

- fuzzySets
- mfType
- outType

Στις δοκιμές, παραμένουν σταθερές οι τιμές των μεταβλητών του αλγορίθμου εκπαίδευσης ANFIS (epoch_number και initial_learning_rate), καθώς η συμπεριφορά τους θα μελετηθεί αργότερα.

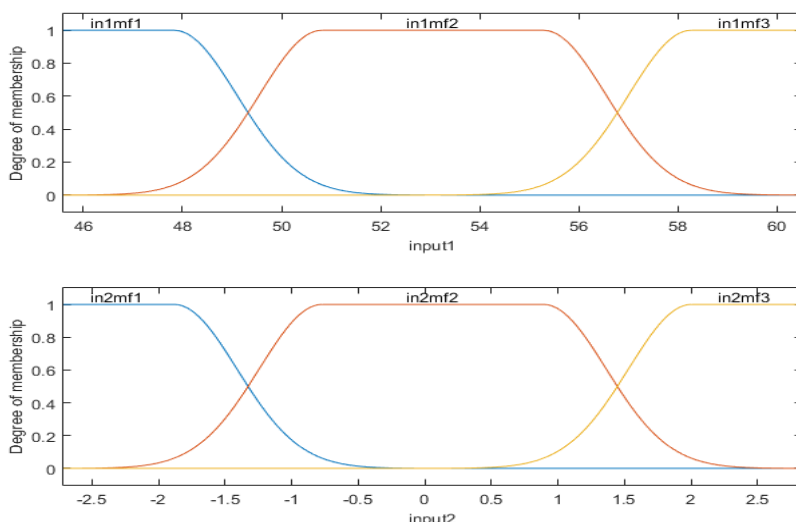
Αρχικά, θα μελετηθεί η επίδραση διαφορετικών τιμών για ασαφή σύνολα ανά είσοδο (fuzzySets). Στις προηγούμενες σελίδες, αναλύθηκε το παράδειγμα με 2 ασαφή σύνολα για κάθε είσοδο, τώρα ελέγχουμε με **3 ασαφή σύνολα**.

Απευθείας παρατηρούμε ότι, το training error μειώθηκε και στις τρεις περιπτώσεις:

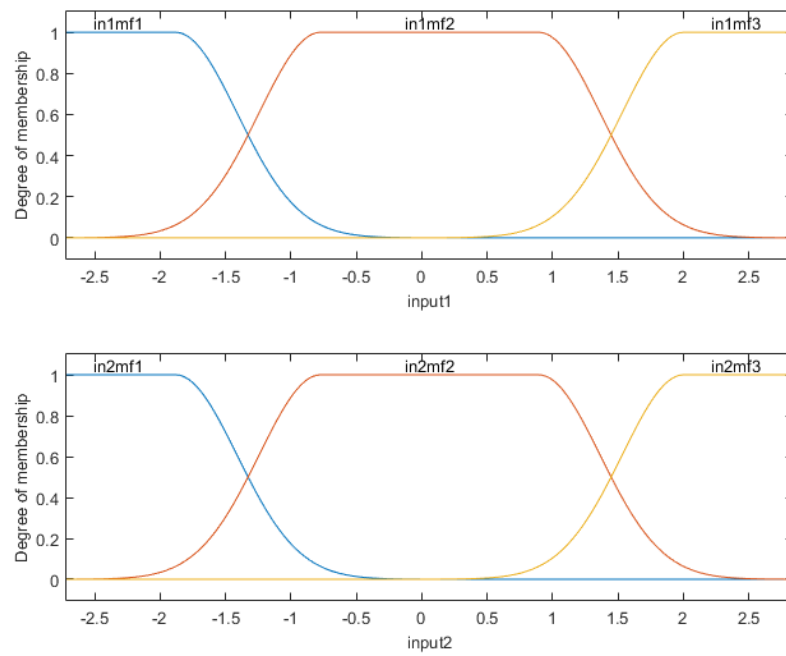
- 2 inputs: από 0.64023 σε 0.62026
- 2 no feedback inputs: από 2.20573 σε 2.16117
- 2 feedback inputs: από 0.040706 σε 0.39234

Διαγράμματα συναρτήσεων συμμετοχής

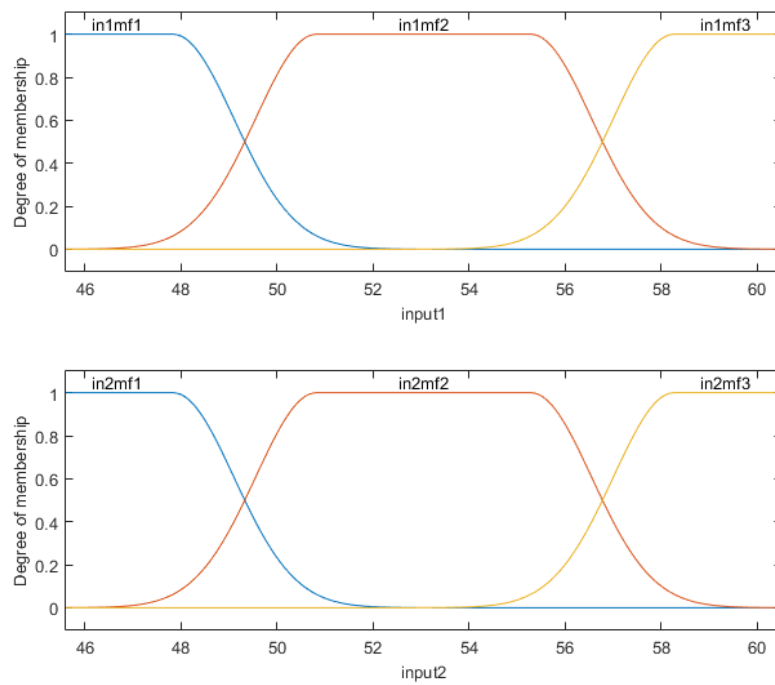
2 inputs



2 inputs no feedback



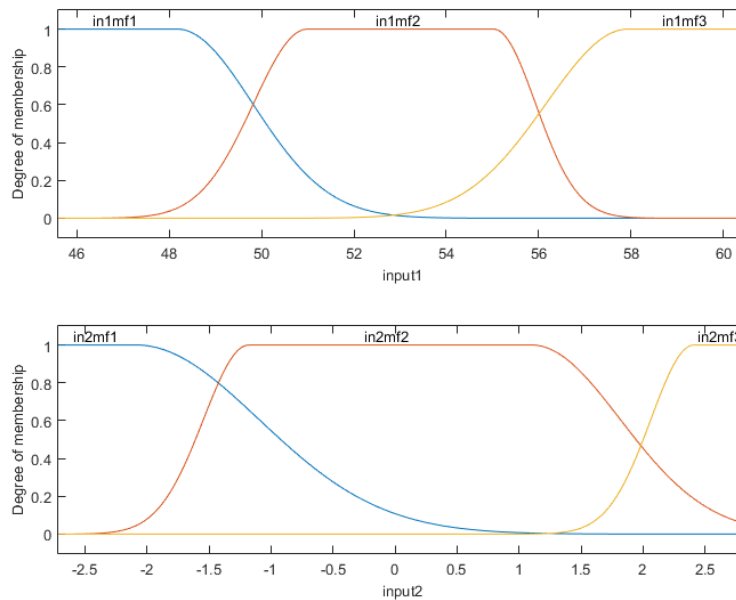
2 feedback inputs



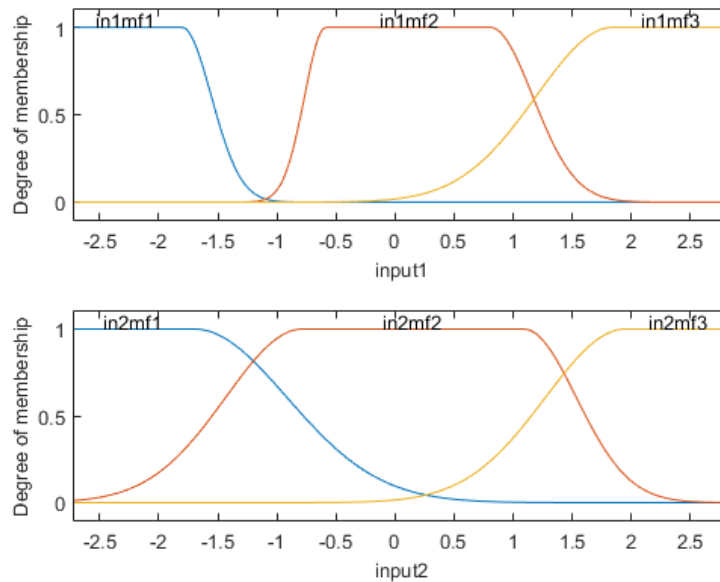
Όπως παρατηρούμε στα διαγράμματα των αρχικών συναρτήσεων συμμετοχής που καλύπτουν ομοιόμορφα τον χώρο εισόδου, έχουμε χρησιμοποιήσει 3 ασαφή σύνολα για κάθε είσοδο. Αυτό οδηγεί, στη διαφοροποίηση με την προηγούμενη περίπτωση που είχαμε 2 ασαφή σύνολα, καθώς τώρα η δομή του νευρωνικού δικτύου καλύπτει παραπάνω περιπτώσεις, έτοιμες για εκπαίδευση. Ακόμη, η εκπαίδευση έκανε περισσότερο χρόνο να ολοκληρωθεί από την προηγούμενη περίπτωση, καθώς τώρα έχουμε περισσότερα δεδομένα για κάθε είσοδο, άρα και περισσότερους υπολογισμούς.

Διαγράμματα τελικών συναρτήσεων συμμετοχής

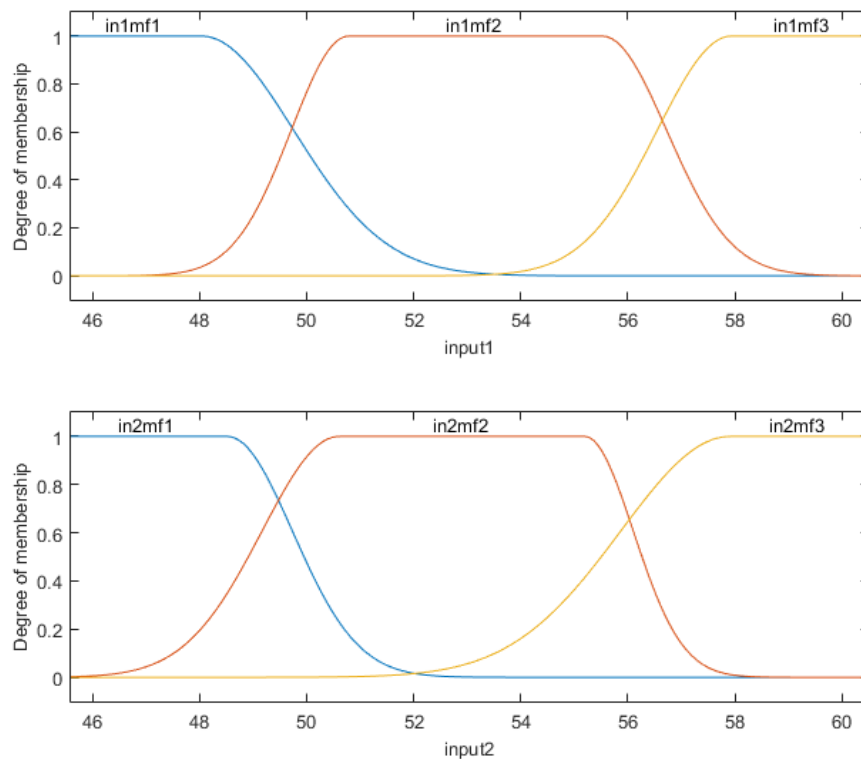
2 inputs



2 inputs no feedback



2 feedback inputs



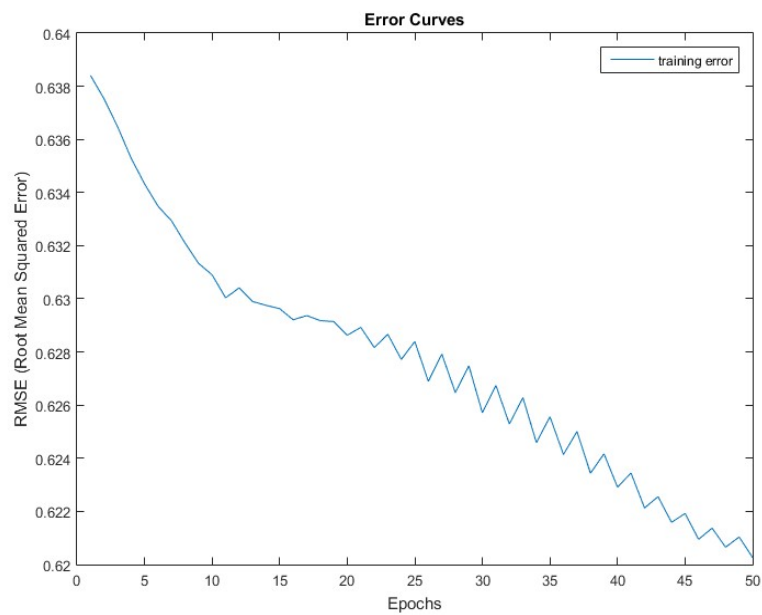
Τα παραπάνω γραφήματα, απεικονίζουν τα αποτελέσματα που εξήχθησαν από τον αλγόριθμο εκπαίδευσης ANFIS.

Παρατηρούμε ότι τα αποτελέσματα του αλγόριθμου εκπαίδευσης, είναι αρκετά διαφορετικά συγκριτικά με τα 2 ασαφή σύνολα ανά είσοδο. Τα 3 ασαφή σύνολα, φαίνεται να αντιπροσωπεύουν με μεγαλύτερη επιτυχία τα αποτελέσματα. Ο αλγόριθμος εκπαίδευσης ANFIS, σε αυτήν την περίπτωση είχε μεγαλύτερη ελευθερία ως προς την προσέγγιση των αποτελεσμάτων, αφού βλέπουμε από τα διαγράμματα ότι τα δεδομένα που παρήχθησαν καλύπτουν μεγάλο εύρος των πιθανών περιπτώσεων.

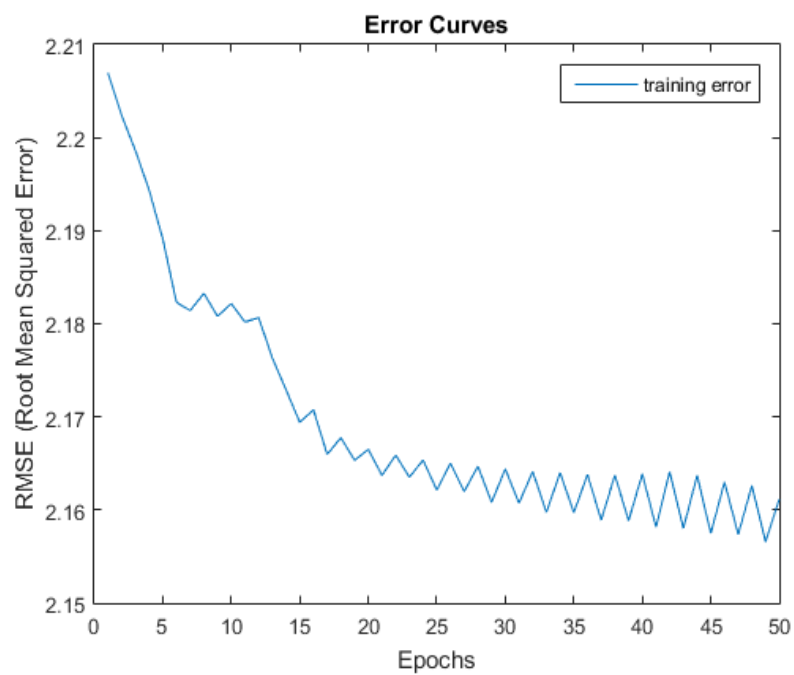
Όπως παρατηρήσαμε παραπάνω, το training error και στις τρεις περιπτώσεις έχει μειωθεί. Συνδυάζοντας την παρατήρηση μας για το σφάλμα και τα αποτελέσματα του αλγόριθμου εκπαίδευσης, βλέπουμε ότι πιθανόν όσο περισσότερα ασαφή σύνολα ανά είσοδο έχει η δομή του νευρωνικού δικτύου, τόσο καλύτερα αποτελέσματα θα έχουμε.

Όμως, δεν μας αρκεί μόνο αυτό για να κρίνουμε αν είναι ικανοποιητικό το αποτέλεσμα, για αυτό και στην συνέχεια θα μελετήσουμε τα διαγράμματα του training error να δούμε ακριβώς την συμπεριφορά του αλγορίθμου αλλά και τα διαγράμματα του ρυθμού εκμάθησης καθώς είναι πολύ σημαντικό να γνωρίζουμε αν φτάνει το maximum του και μειώνεται και γενικότερα την συμπεριφορά του επίσης.

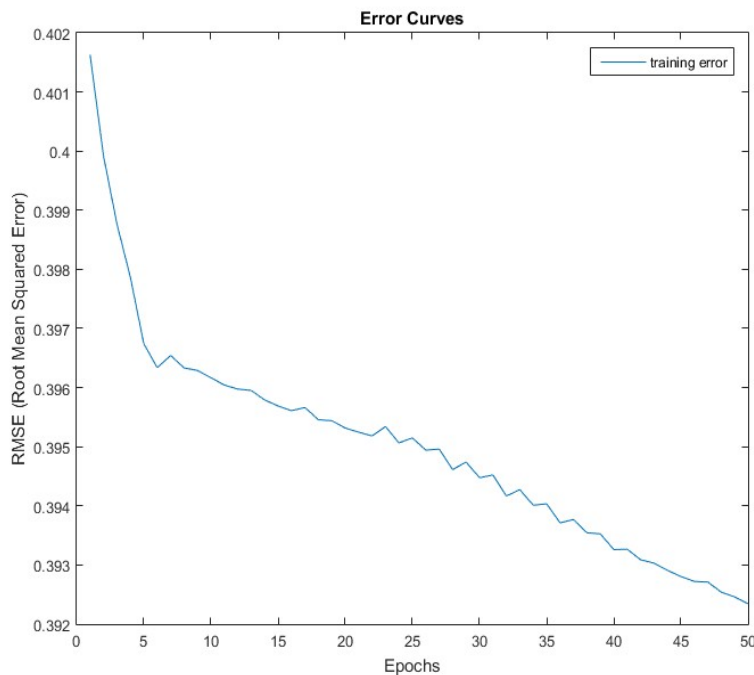
Διαγράμματα training error
2 inputs



2 inputs no feedback



2 feedback inputs

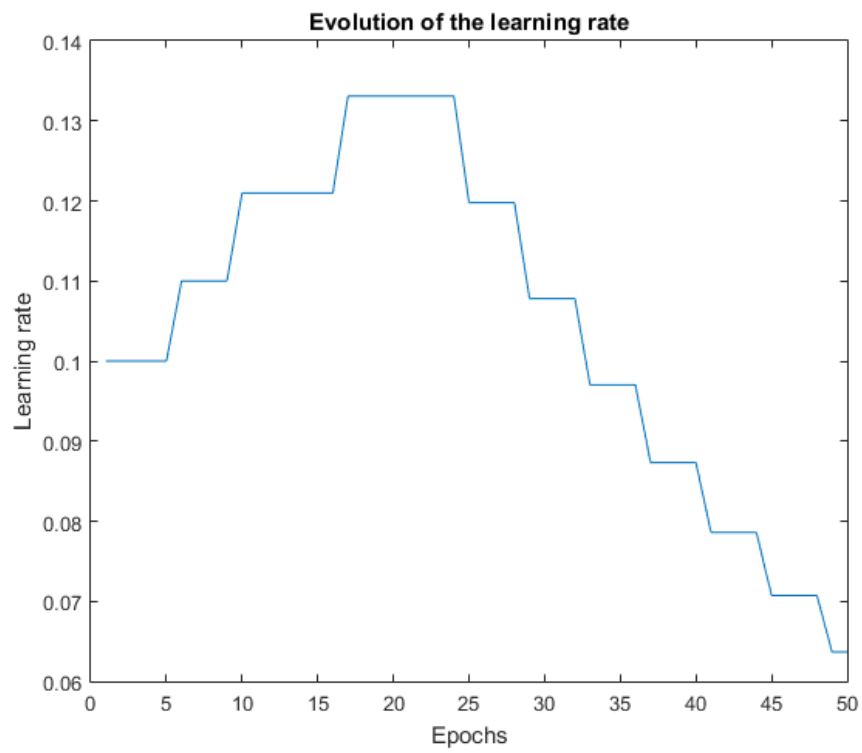


Παρατηρούμε στα διαγράμματα του σφάλματος, ότι έχουμε διαφοροποιήσεις για τις 3 περιπτώσεις των 2 inputs.

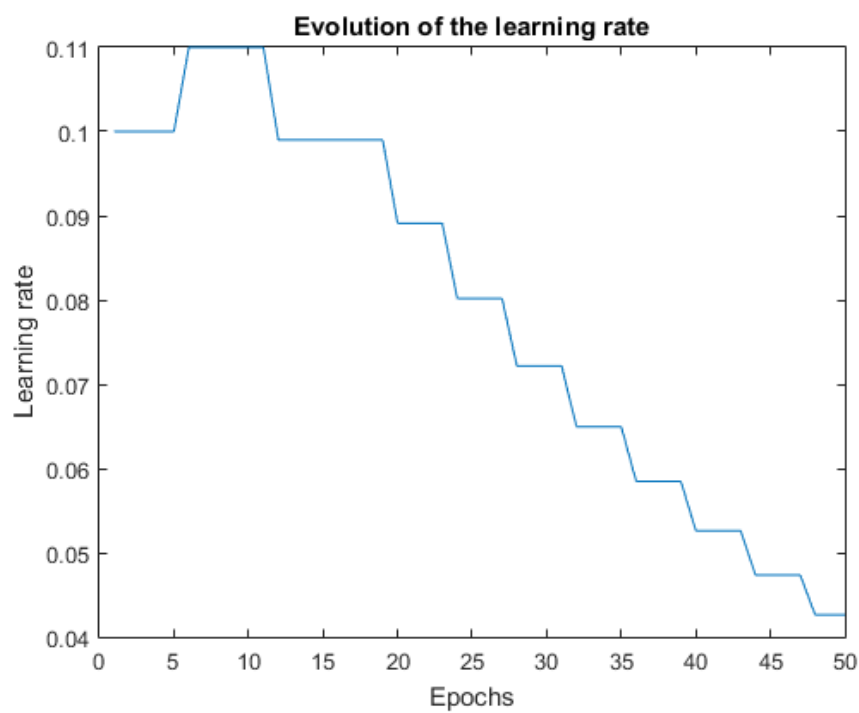
- Για την περίπτωση των **2 inputs**, βλέπουμε ότι η μείωση του error με το πέρασμα των επαναλήψεων για τα 2 ασαφή σύνολα ανά είσοδο είναι πιο ομαλή από ότι για τα 3 ασαφή σύνολα ανά είσοδο. Στην τελευταία περίπτωση, έχουμε πιο πολλές αυξομειώσεις κατά την εκπαίδευση, αλλά καταλήγει σε σημαντική μείωση, δεν παραμένει δηλαδή σταθερό και αυξομειώνεται.
- Για την περίπτωση των **2 inputs χωρίς feedback**, βλέπουμε ότι στα 3 ασαφή σύνολα ανά είσοδο αρκετές αυξομειώσεις και το αξιοσημείωτο εδώ είναι ότι, περίπου από τα μισά και μετά, το σφάλμα δεν έχει σημαντική μείωση όπως είχε στο πρώτο μισό της εκπαίδευσης. Ενώ αντίθετα, στα 2 ασαφή σύνολα ανά είσοδο έχουμε περισσότερη μείωση για το μεγαλύτερο μέρος της εκπαίδευσης. Όμως τελικώς, το τελικό σφάλμα είναι μικρότερο στα 3 ασαφή σύνολα ανά είσοδο, άρα ένας στόχος για την συνέχεια θα είναι η εξομάλυνση αυτών των αυξομειώσεων έτσι ώστε να πετύχουμε μεγαλύτερο training error
- Για την περίπτωση των **2 inputs με feedback**, έχουμε πολύ καλύτερη συμπεριφορά του αλγορίθμου. Καθώς συγκριτικά με τα 2 ασαφή σύνολα ανά είσοδο, τα 3 ασαφή σύνολα είσοδο έχουν λιγότερες αυξομειώσεις του σφάλματος με το πέρασμα της εκπαίδευσης. Στην πρώτη περίπτωση, έχουμε κορυφές προς από τα μισά της εκπαίδευσης και μετά, με το σφάλμα να μην μειώνεται σημαντικά. Ενώ στην περίπτωση μας τώρα, το σφάλμα μειώνεται σε όλα τα στάδια του αλγορίθμου εκπαίδευσης.

Η συχνότητα εμφάνισης και το είδος των αυξομειώσεων, οφείλεται σαφώς σε όλες τις μεταβλητές μας αλλά και στα δεδομένα. Με περισσότερα δεδομένα προς εκπαίδευση, θα μπορούσαν να είχαν μειωθεί οι αυξομειώσεις που παρατηρήσαμε.

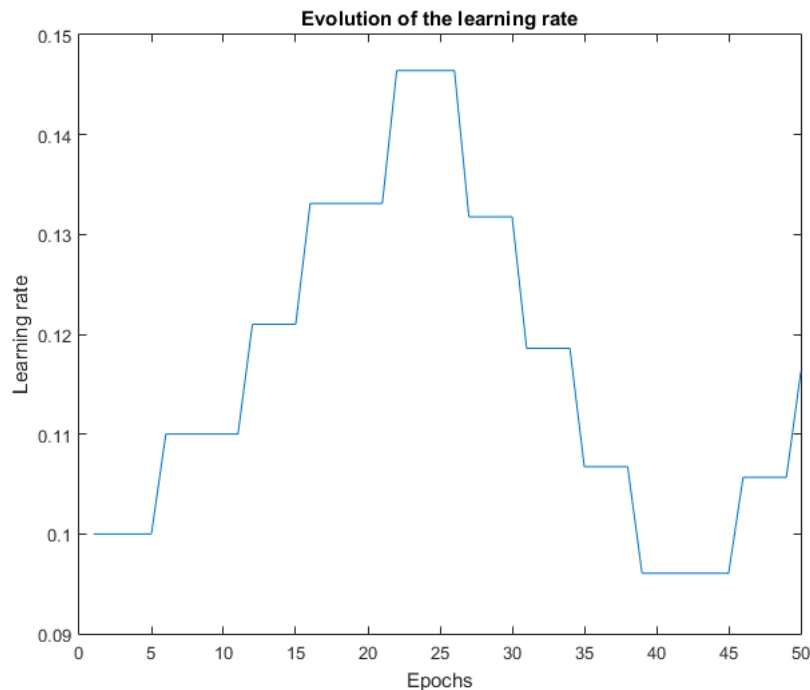
Διαγράμματα learning rate
2 inputs



2 inputs no feedback



2 feedback inputs



Όπως αναφέρθηκε προηγουμένως, ο ρυθμός εκμάθησης (learning rate) του αλγορίθμου εκπαίδευσης, πρέπει με το πέρασμα των επαναλήψεων να αυξάνεται, στη συνέχεια να φτάνει ένα maximum και έπειτα να μειώνεται για το υπόλοιπο της εκπαίδευσης.

Με γνώμονα αυτό, μπορούμε να εξάγουμε τα παρακάτω συμπεράσματα:

- Για την περίπτωση των **2 inputs**, βλέπουμε ότι ο ρυθμός εκμάθησης έχει την συμπεριφορά που ζητάμε, καθώς αυξάνεται αρχικά και στη συνέχεια μειώνεται. Θα μπορούσε να θεωρηθεί μια αποδεκτή τιμή για 3 ασαφή σύνολα ανά είσοδο.
- Για την περίπτωση των **2 inputs χωρίς feedback**, βλέπουμε ότι ναι μεν αρχικά αυξάνεται και μετά μειώνεται, αλλά ξεκινάει πολύ οριακά την αύξηση του. Είναι μια αποδεκτή συμπεριφορά, αλλά ενδεχομένως να μπορούμε να πετύχουμε καλύτερα αποτελέσματα στη συνέχεια.
- Για την περίπτωση των **2 inputs με feedback**, παρατηρούμε ότι υπάρχει η ζητούμενη συμπεριφορά όμως στο τέλος υπάρχει ξανά αύξηση χωρίς μείωση. Συνολικά, είναι μια συμπεριφορά που δεν ζητάμε, οπότε θα χρειαστεί σίγουρα να επέμβουν και άλλες μεταβλητές σε αυτή τη περίπτωση για να την αποδεχθούμε.

Παρατηρούμε ότι με την ανάλυση των αποτελεσμάτων, μπορούμε να σχηματίσουμε άποψη για το αν αρκεί η αλλαγή των εισόδων ανά ασαφή σύνολα για κάθε είσοδο, και τι κρύβεται στον αλγόριθμο εκπαίδευσης ANFIS.

Δοκιμή με 5 ασαφή σύνολα ανά είσοδο

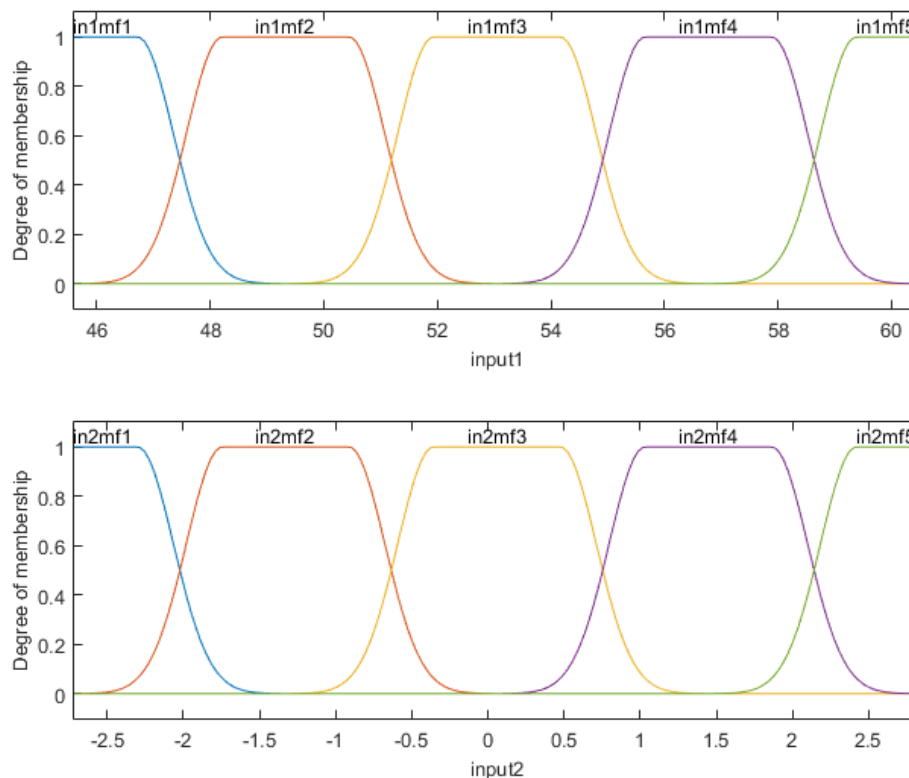
Προηγουμένως ελέγχθηκαν και αναλύθηκαν οι περιπτώσεις για 2 και 3 ασαφή σύνολα ανά είσοδο και τώρα θα ελέγξουμε για 5 ασαφή σύνολα ανά είσοδο για τη δομή του ασαφούς νευρωνικού δικτύου.

Η πρώτη παρατήρηση είναι η σημαντική μείωση του training error συγκριτικά με τις περιπτώσεις για 2 και 3 ασαφή σύνολα ανά είσοδο.

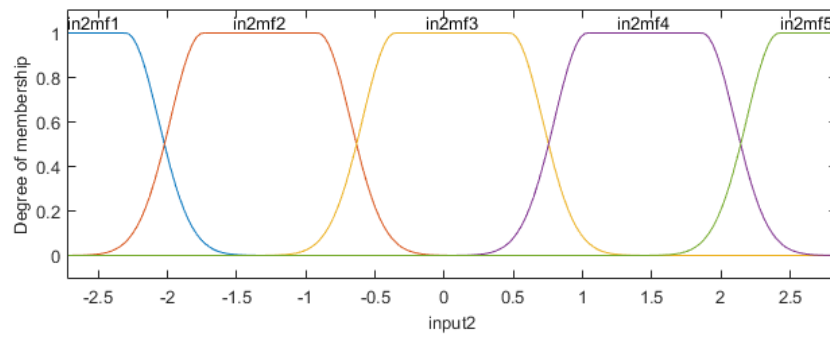
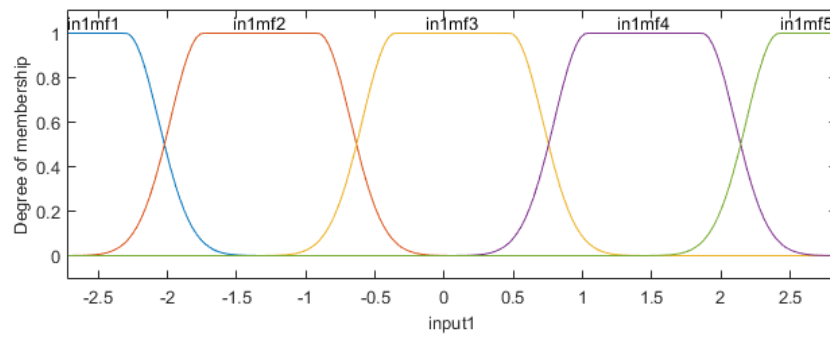
Πιο συγκεκριμένα:

- Για 2 inputs έχουμε training error από 0.5897 έως 0.5650
- Για 2 inputs χωρίς feedback έχουμε training error από 2.1637 έως 2.0392
- Για 2 inputs με feedback έχουμε training error από 0.3828 έως 0.3578

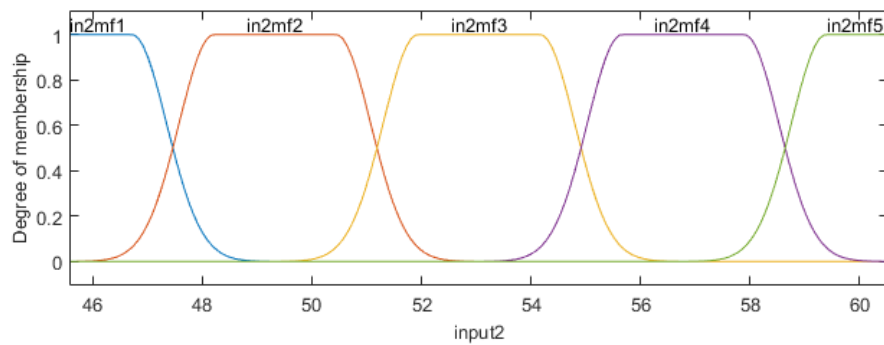
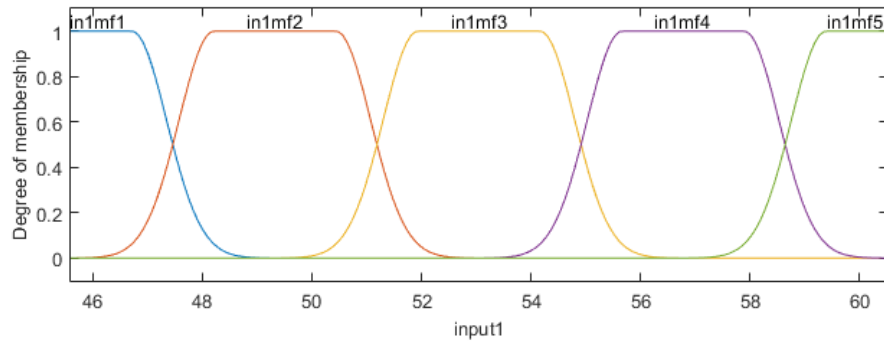
Διαγράμματα συναρτήσεων συμμετοχής **2 inputs**



2 inputs no feedback



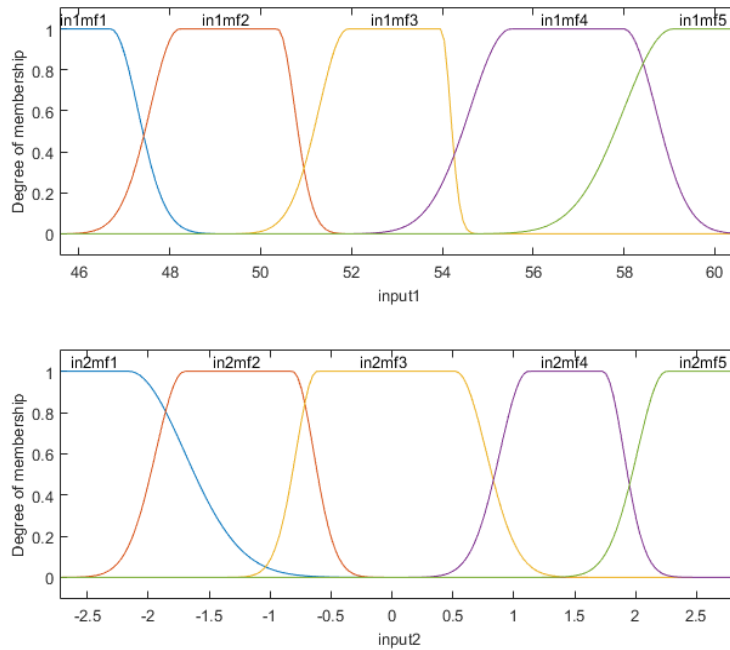
2 feedback inputs



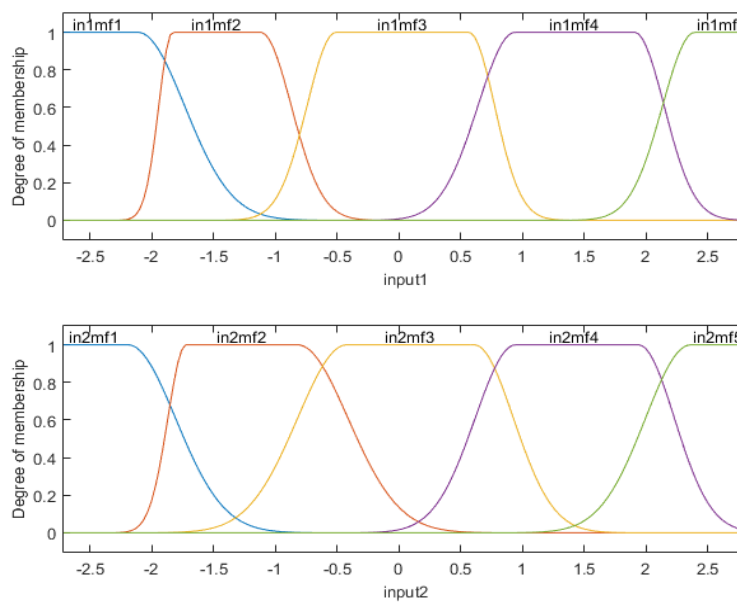
Με τη χρήση 5 ασαφή συνόλων για κάθε είσοδο, παρατηρούμε ότι γίνεται ακόμη μεγαλύτερος διαχωρισμός των δεδομένων προς εκπαίδευση. Προηγουμένως, είχαμε συμπεράνει ότι ενδέχεται όσο περισσότερα ασαφή σύνολα έχουμε τόσο καλύτερα αποτελέσματα θα πάρουμε. Αρχίζουμε να το βλέπουμε αυτό, καθώς έχουμε μια αρχική εικόνα ότι θα ελεγχθούν περισσότερες περιπτώσεις για τα δεδομένα μας. Όμως, έχουμε ακόμη αρκετά πράγματα να ελέγξουμε.

Διαγράμματα τελικών συναρτήσεων συμμετοχής

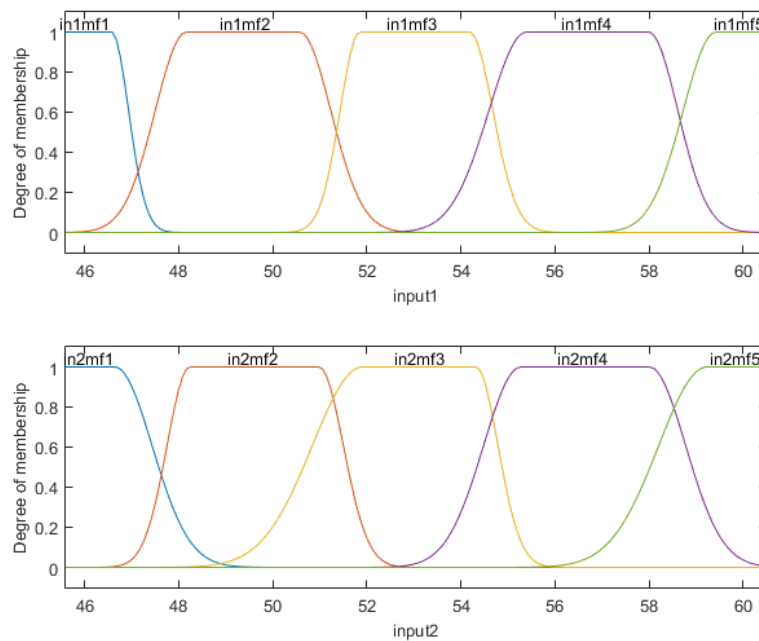
2 inputs



2 inputs no feedback



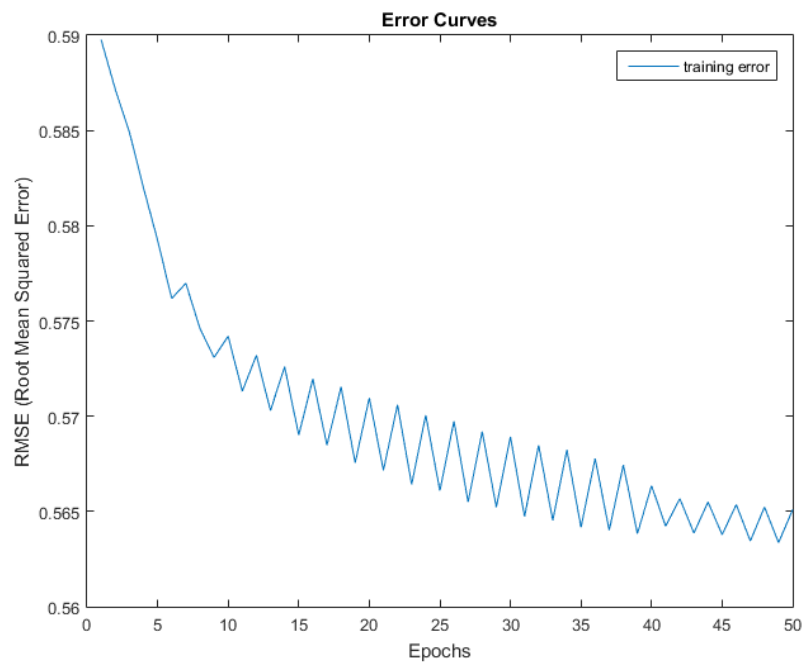
2 feedback inputs



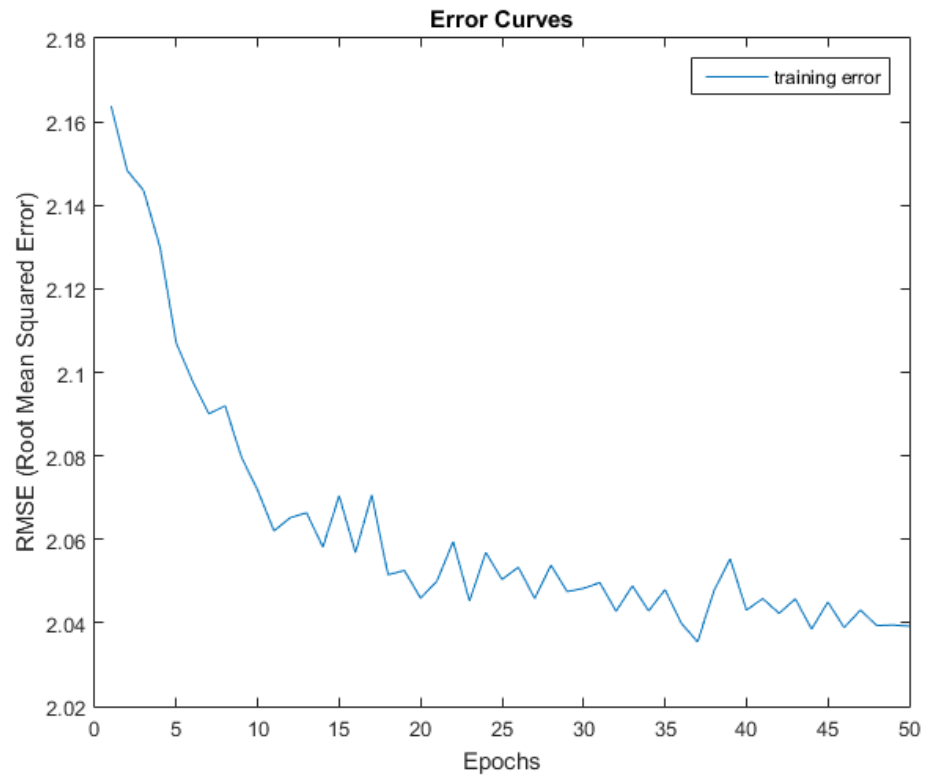
Στα διαγράμματα των τελικών συναρτήσεων συμμετοχής, παρατηρούμε ότι ο αλγόριθμος εκπαίδευσης ANFIS, έχει εκπαιδέψει τα δεδομένα πιο ομοιόμορφα πλέον. Αυτό οφείλεται εμφανώς, στα 5 ασαφή σύνολα ανά είσοδο που δόθηκαν. Το κάθε ασαφές σύνολο έχει μικρότερο χώρο ευθύνης πλέον, άρα και οι καμπύλες δεν έχουν τόσες αποκλίσεις μεταξύ των εισόδων.

Διαγράμματα training error

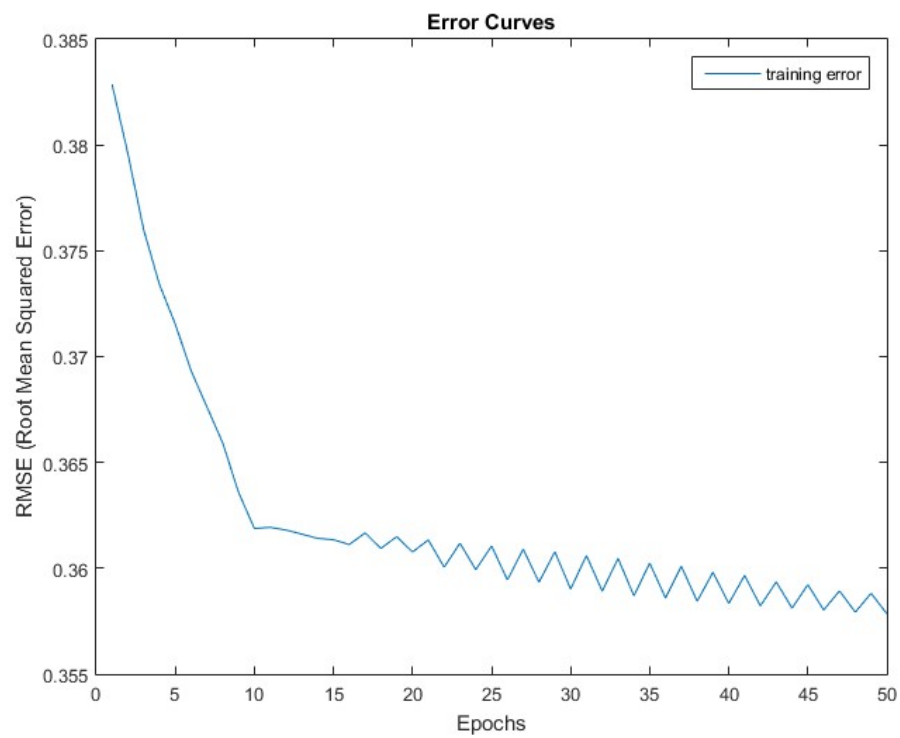
2 inputs



2 inputs no feedback



2 feedback inputs

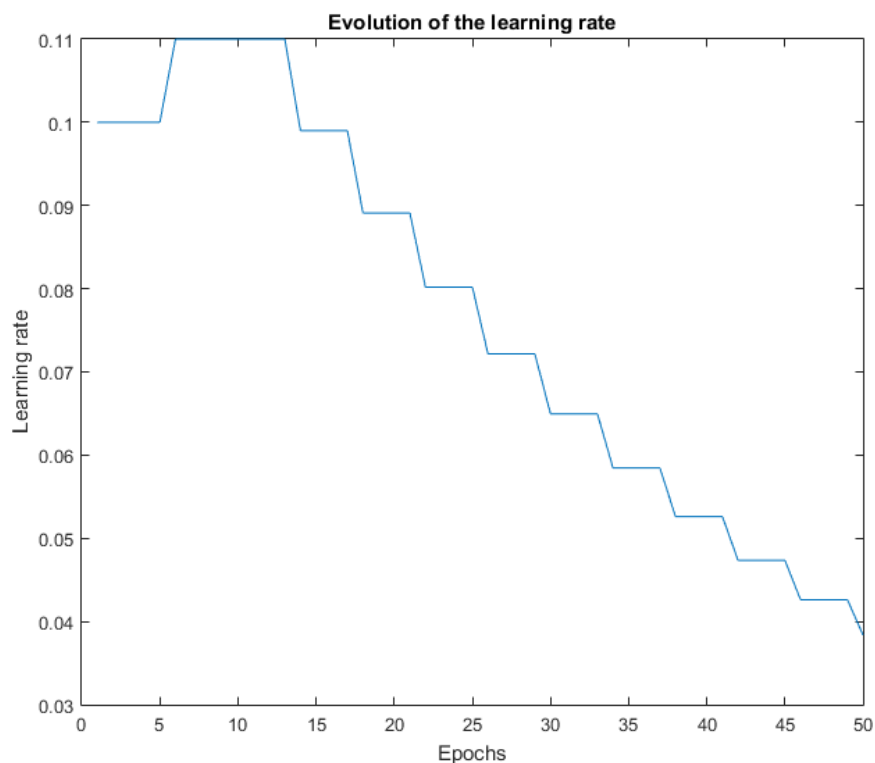


Βλέπουμε στα διαγράμματα του training error, ότι πάλι έχουμε διαφοροποιήσεις μεταξύ των περιπτώσεων για 2 inputs

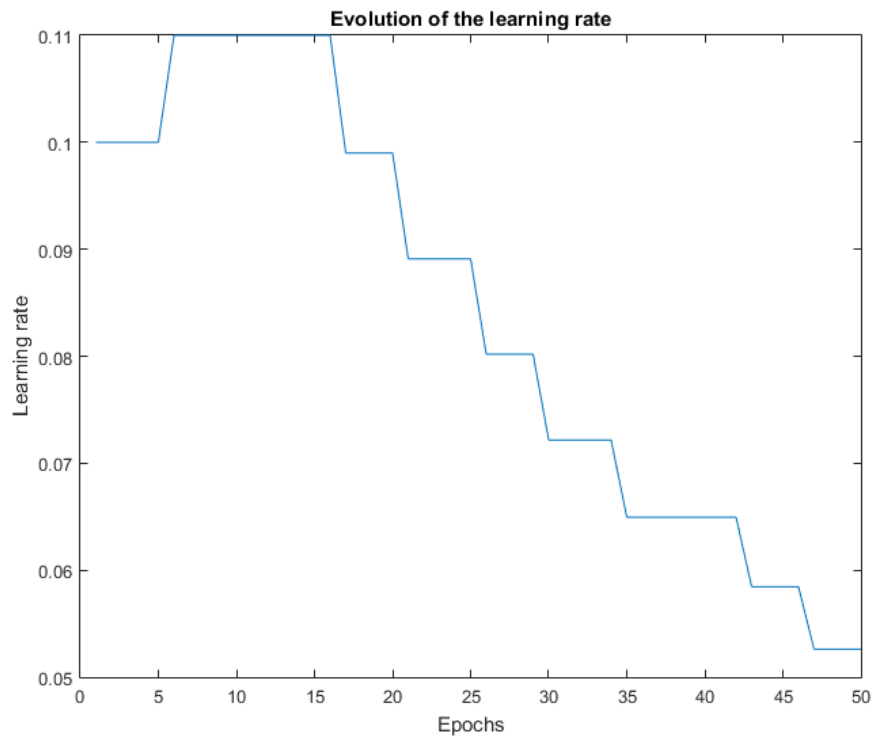
- Για την περίπτωση των **2 inputs**, βλέπουμε ότι αρχικά μειώνεται πολύ το σφάλμα, αλλά στη συνέχεια παρουσιάζει πάρα πολλές αυξομειώσεις, περισσότερο από το 70% της εκπαιδύσεως περιέχει αυξομειώσεις. Αυτό είναι ένα αρχικό δείγμα ότι θα χρειαστούμε σίγουρα διαφορετικές τιμές στις υπόλοιπες παραμέτρους για να αποδεχθούμε τα 5 ασαφή σύνολα ανά είσοδο σε αυτή τη περίπτωση.
- Για την περίπτωση των **2 inputs χωρίς feedback**, βλέπουμε, ίσως για πρώτη φορά, αυξομειώσεις οι οποίες δεν ακολουθούν το ίδιο μοτίβο. Είναι εμφανές ότι επιδέχεται βελτίωση η συμπεριφορά του αλγορίθμου εκπαίδευσης.
- Για την περίπτωση των **2 inputs με feedback**, η συμπεριφορά του training error κατά τη διάρκεια της εκπαίδευσης, είναι σαφώς πιο αποδεκτή από τις προηγούμενες αλλά επίσης και αυτή η περίπτωση, στο μεγαλύτερο μέρος της εκπαίδευσης έχει χαμηλό ρυθμό μείωσης του σφάλματος και πολλές αυξομειώσεις.

Συνολική παρατήρηση είναι ότι και στις τρεις περιπτώσεις, ο ρυθμός μείωσης του σφάλματος μειώνεται αρκετά στο μεγαλύτερο σημείο της διαδικασίας της εκπαίδευσης. Αυτό μας δείχνει ότι ενδεχομένως, να μη μας αρκούν οι τωρινές τιμές των υπόλοιπων μεταβλητών για 5 ασαφή σύνολα ανά είσοδο. Ο ευρύτερος στόχος, είναι να έχουμε κάποιες συνεχείς τιμές προς το τέλος της εκπαίδευσης και με λιγότερες αυξομειώσεις.

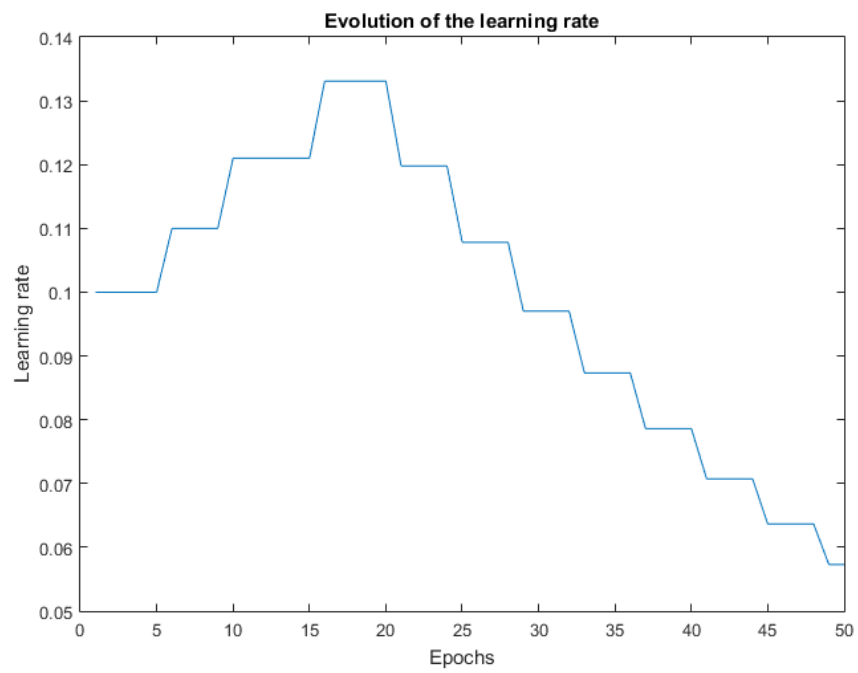
Διαγράμματα learning rate **2 inputs**



2 inputs no feedback



2 feedback inputs



Μια πρώτη παρατήρηση από τα διαγράμματα του ρυθμού εκμάθησης του αλγορίθμου εκπαίδευσης, είναι ότι και οι τρεις περιπτώσεις για 5 ασαφή σύνολα ανά είσοδο, έχουν αποδεκτή συμπεριφορά. Και για τις τρεις περιπτώσεις, αυξάνεται ο ρυθμός εκμάθησης, στη συνέχεια φτάνει σε ένα μέγιστο και στη συνέχεια μειώνεται μέχρι το τέλος της εκπαίδευσης που είναι και το ζητούμενο.

Είναι μια πρώτη ένδειξη, ότι τα 5 ασαφή σύνολα ανά είσοδο, είναι μια σωστή επιλογή για την εκπαίδευση των δεδομένων μας. Αλλά όπως είδαμε προηγουμένως, χρειάζεται να τροποποιηθούν και οι υπόλοιπες μεταβλητές για να βελτιστοποιήσουμε τα αποτελέσματα μας.

Είδος Συνάρτησης Συμμετοχής

Σε όλες τις παραπάνω εκτελέσεις και δοκιμές, χρησιμοποιήθηκε σαν είδος συνάρτησης συμμετοχής η **gaus2mf**, η οποία είναι μια Γκαουσιανή ασαφής συνάρτηση συμμετοχής δύο συνδυασμένων Γκαουσιανών μελών.

Παρακάτω θα παρουσιαστούν δοκιμές με συνάρτηση συμμετοχής την **trimf**, η οποία είναι τριγωνική συνάρτηση συμμετοχής.

Οι μεταβλητές έχουν τις default τιμές τους.

- epoch_number=50
- initial_learning_rate = 0.1
- outType = linear

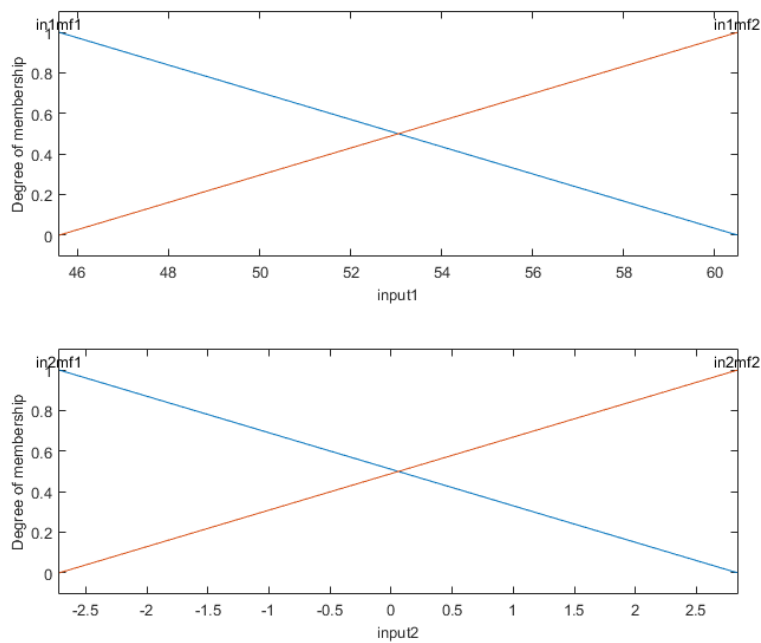
Και για ασαφή σύνολα ανά είσοδο, θα παρουσιαστούν αποτελέσματα για 2,3 και 5 (fuzzySets).

Αρχικά, η τριγωνική συνάρτηση συμμετοχής, ορίζει ένα κατώτατο όριο **a**, ένα ανώτατο όριο **b** και μια τιμή **m**. Όπου ισχύει **a < m < b**.

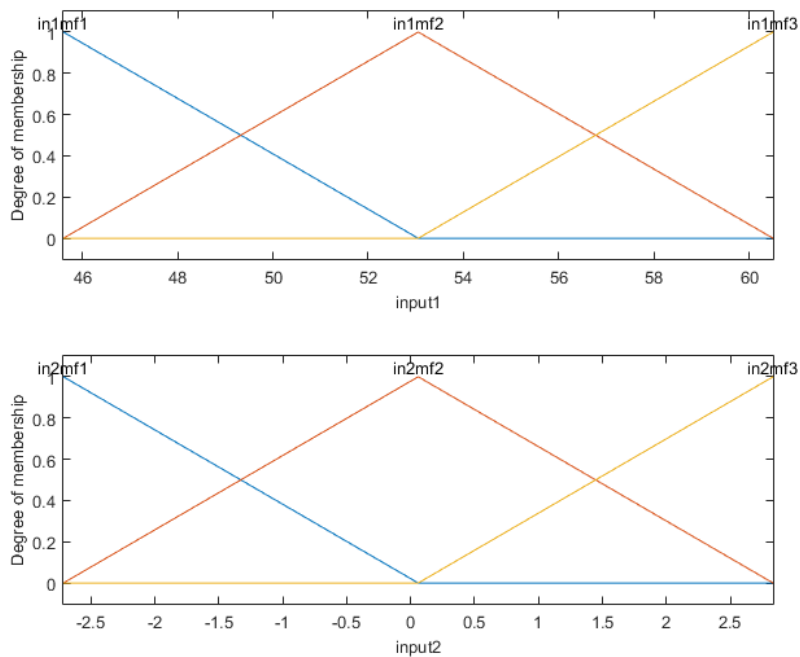
$$\mu_A(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{m-a}, & a < x \leq m \\ \frac{b-x}{b-m}, & m < x < b \\ 0, & x \geq b \end{cases}$$

Το παραπάνω, είναι εμφανές αν δούμε τα διαγράμματα των συναρτήσεων συμμετοχής.

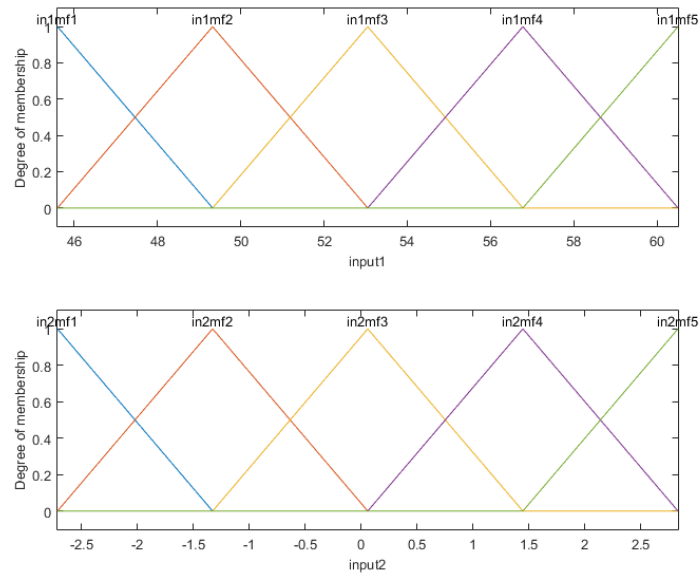
2 inputs με fuzzySets = 2



2 inputs με fuzzySets = 3



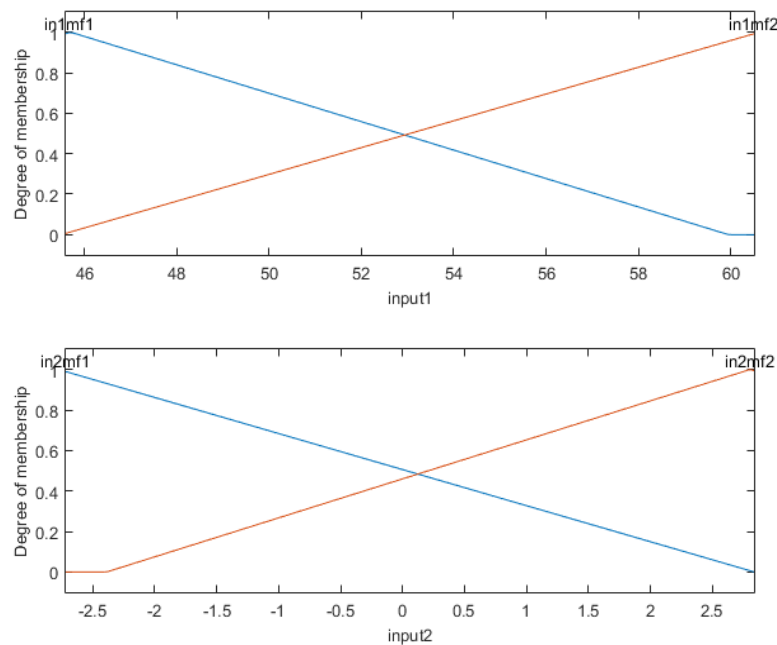
2 inputs με fuzzySets = 5



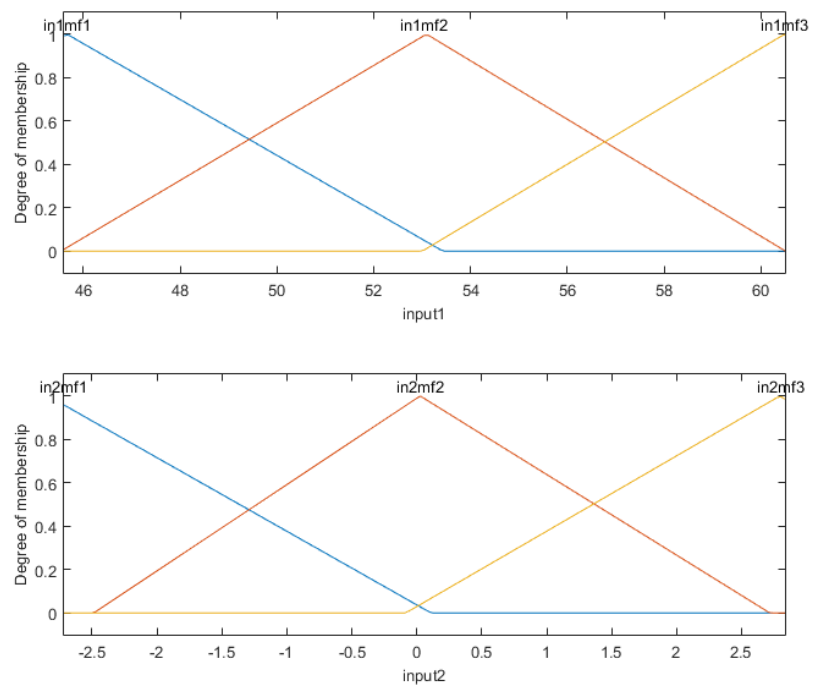
Στη περίπτωση της τριγωνικής συνάρτησης συμμετοχής, η δομή του ασαφούς νευρωνικού δικτύου είναι αρκετά διαφορετική από την γκαουσιανή συνάρτηση συμμετοχής. Τα ασαφή σύνολα, πλέον καθορίζουν το interface σύμφωνα με τις 3 τιμές που ορίστηκαν. Κάθε σύνολο έχει τις δικές του τιμές, και ορίζει τη δομή με βάση αυτές.

Θα μελετηθούν τα διαγράμματα των τελικών συναρτήσεων συμμετοχής

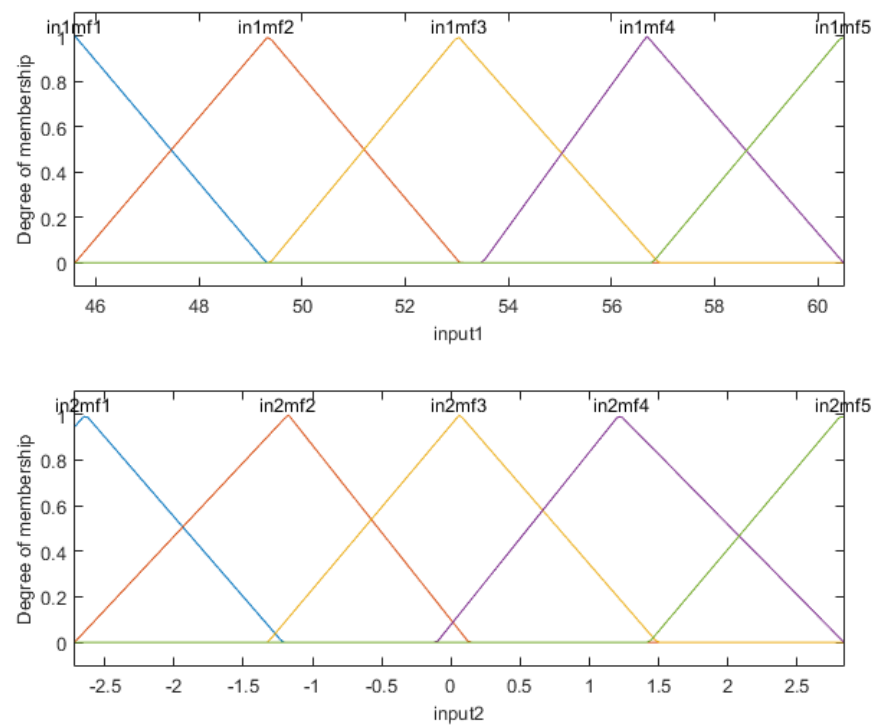
2 inputs με fuzzySets = 2



2 inputs $\mu \varepsilon$ fuzzySets = 3



2 inputs $\mu \varepsilon$ fuzzySets = 5



Με την χρήση της τριγωνικής συνάρτησης συμμετοχής, ο αλγόριθμος εκπαίδευσης ANFIS, παράγει με την σειρά του τα παραπάνω αποτελέσματα. Για κάθε ασαφή σύνολο ανά είσοδο, έχει δημιουργήσει τις αντίστοιχες μεταβλητές για τις περιπτώσεις που καθόρισε.

Μετά από μελέτη των παραπάνω, βλέπουμε ότι στις περισσότερες περιπτώσεις, η τριγωνική συνάρτηση συμμετοχής έχει μεγαλύτερο training error από την γκαουσιανή συνάρτηση συμμετοχής.

Παρουσιάζονται τα τελικά training error για τις 3 περιπτώσεις των 2 inputs για ασαφή σύνολα 2,3 και 5:

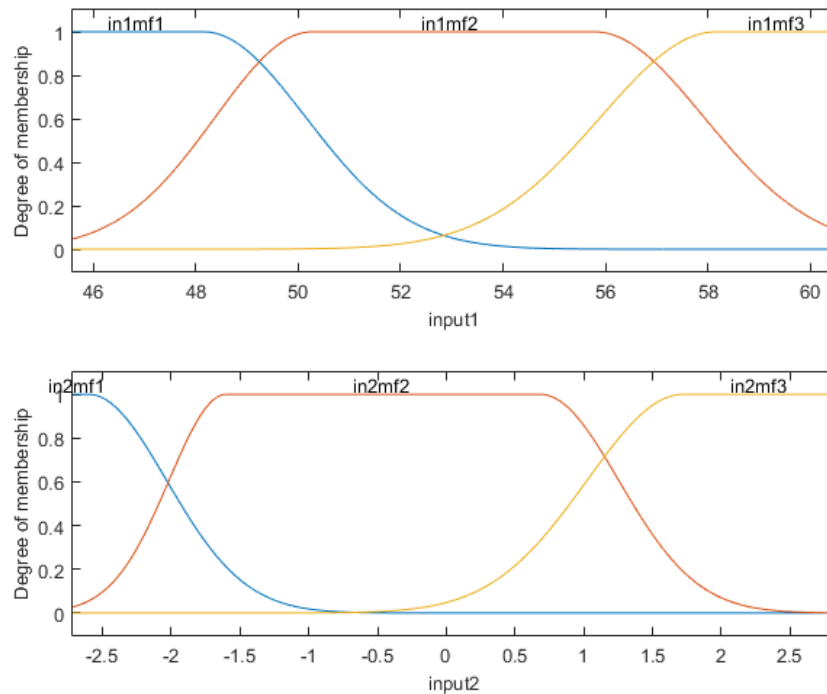
- **2 inputs**
 - fuzzySets = 2
 - gaus2mf: training error = 0.64023
 - trimf: training error = 0.65720
 - fuzzySets=3
 - gaus2mf: training error = 0.62026
 - trimf: training error = 0.63094
 - fuzzySets=5
 - gaus2mf: training error = **0.56509**
 - trimf: training error = **0.55964 (Μικρότερο σφάλμα από gaus2mf)**
- **2 inputs no feedback**
 - fuzzySets = 2
 - gaus2mf: training error = 2.20573
 - trimf: training error = 2.22638
 - fuzzySets=3
 - gaus2mf: training error = **2.16117**
 - trimf: training error = **2.15631 (Μικρότερο σφάλμα από gaus2mf)**
 - fuzzySets=5
 - gaus2mf: training error = 2.03920
 - trimf: training error = 2.05400
- **2 inputs feedback**
 - fuzzySets = 2
 - gaus2mf: training error = 0.392812
 - trimf: training error = 0.40967
 - fuzzySets=3
 - gaus2mf: training error = **0.39234**
 - trimf: training error = **0.39050 (Μικρότερο σφάλμα από gaus2mf)**
 - fuzzySets=5
 - gaus2mf: training error = 0.35784
 - trimf: training error = 0.37616

Παρατηρούμε, ότι υπάρχουν περιπτώσεις που η τριγωνική συνάρτηση συμμετοχής έχει μικρότερο σφάλμα από την γκαουσιανή συνάρτηση συμμετοχής, αλλά τις περισσότερες φορές είναι μεγαλύτερο το σφάλμα της. Αυτό καθορίζεται από το είδος των δεδομένων, στην δική μας περίπτωση φαίνεται να ταιριάζει περισσότερο η γκαουσιανή συνάρτηση συμμετοχής. Μετά και από περαιτέρω δοκιμές, θα συνεχίσουμε να χρησιμοποιούμε την **gauss2mf**

Είδος Εξόδου Του Κανόνα

Μέχρις στιγμής, σε όλους τους πειραματισμούς, είχαμε θέσει σαν είδος εξόδου του κανόνα την γραμμική (linear). Στη συνέχεια, θα παρουσιαστούν δοκιμές για το είδος εξόδου του κανόνα ως σταθερή (constant).

Για να καταλάβουμε τις διαφορές μεταξύ των δύο, αρκεί μονάχα να παρουσιάσουμε τα αποτελέσματα του 2 inputs με 3 ασαφή σύνολα ανά είσοδο του αλγορίθμου εκπαίδευσης ANFIS.



Το πρώτο που παρατηρούμε στο σταθερό είδος εξόδου του κανόνα, τα ασαφή σύνολα του αλγορίθμου εκπαίδευσης, θα έχουν σταθερή συμπεριφορά. Δηλαδή, οι καμπύλες θα είναι ταυτόσημες, σε σχέση με την γραμμική που το μήκος και πλάτος των καμπυλών διαφέρει.

Άλλη μια σημαντική διαφοροποίηση, είναι στον υπολογισμό του σφάλματος. Η αρχική τιμή του σφάλματος είναι μεγαλύτερη από την αρχική τιμή του σφάλματος την γραμμική έξοδο του κανόνα.

Παράδειγμα δοκιμής, 2 input με fuzzySets = 2:

- linear: training error = από 0.65255 έως 0.64678
- constant: training error = από 1.1046 έως 0.8774

Μετά από εξαντλητικούς ελέγχους, δεν βρέθηκε συνδυασμός μεταβλητών που να δίνει σφάλμα μικρότερο από της γραμμικής εξόδου.

Αλγόριθμος Εκπαίδευσης ANFIS (για τις τρεις περιπτώσεις των 2 inputs)

Προηγουμένως, μελετήθηκε η επίδραση των της δομής του νευρωνικού δικτύου στον αλγόριθμο εκπαίδευσης για τις τρεις περιπτώσεις των **2 inputs**. Τώρα, θα μελετηθούν οι μεταβλητές του αλγορίθμου εκπαίδευσης που είναι:

- Αριθμός επαναλήψεων εκπαίδευσης (epoch_number)
- Αρχικός ρυθμός εκμάθησης (initial_learning_rate)

Αριθμός επαναλήψεων εκπαίδευσης (epoch_number) για τις τρεις περιπτώσεις των 2 inputs

Είναι ο μέγιστος αριθμός επαναλήψεων που θα εκπαιδευτεί το νευρωνικό δίκτυο. Η διαδικασία της εκπαίδευσης από τον αλγόριθμο εκπαίδευσης ANFIS, σταματάει μόλις φτάσει σε αυτόν τον αριθμό τον οποίο έχουμε ορίσει εμείς.

Κατά κανόνα, όσες περισσότερες επαναλήψεις εκπαίδευσης έχει ο αλγόριθμος, τόσο καλύτερα αποτελέσματα θα έχουμε.

Όμως θα πρέπει να έχουμε υπόψιν μας τα παρακάτω:

- Η πολύ μεγάλη αύξηση των αριθμών επαναλήψεων, οδηγεί στην ανάγκη για περισσότερους πόρους από το υπολογιστικό σύστημα που εκτελείται ο αλγόριθμος. Άρα, προϋπόθεση το να κριθεί αριθμός κριθεί κατάλληλος, θα πρέπει να μην βαραίνει το σύστημα.
- Την αποφυγή του overfitting. Μπορεί με ένας μεγάλος αριθμός επαναλήψεων να μας δώσει ένα σημαντικά πιο μικρό training error, αλλά υπάρχει περίπτωση το δίκτυο να έχει απομνημονεύσει τις τιμές με τις οποίες εκπαιδεύτηκε και όταν εισαχθούν πιο γενικευμένες τιμές από αυτές, το training error να είναι αρκετά υψηλό.

Σε όλες τις προηγούμενες δοκιμές, παρουσιάστηκαν αποτελέσματα με αριθμό επαναλήψεων εκπαίδευσης ίσων με 50.

Στη συνέχεια θα παρουσιαστούν κάποια αποτελέσματα για τις παρακάτω τιμές (για χάριν οικονομίας και πρακτικότητα δεν θα παρουσιαστούν όλες οι περιπτώσεις):

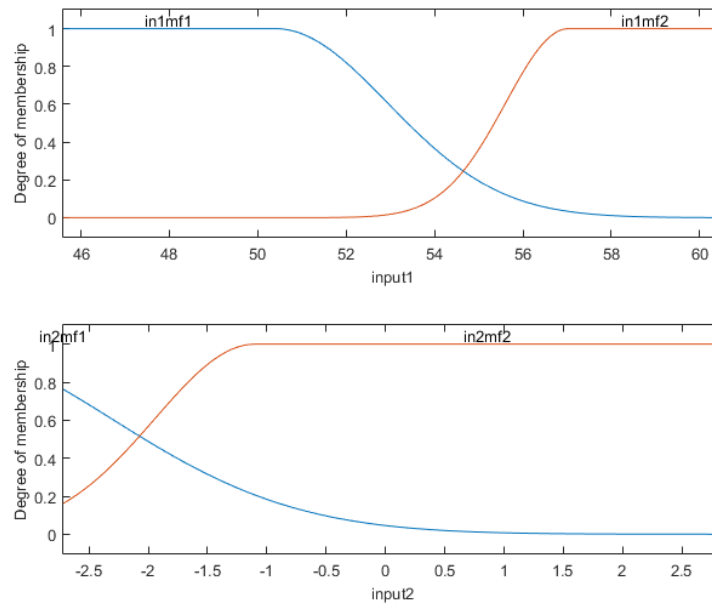
- 50
- 500
- 1500

Θα παρουσιαστούν όλα τα διαγράμματα που φαίνεται να επηρεάζει ο αριθμός επαναλήψεων για την περίπτωση των **2 inputs**.

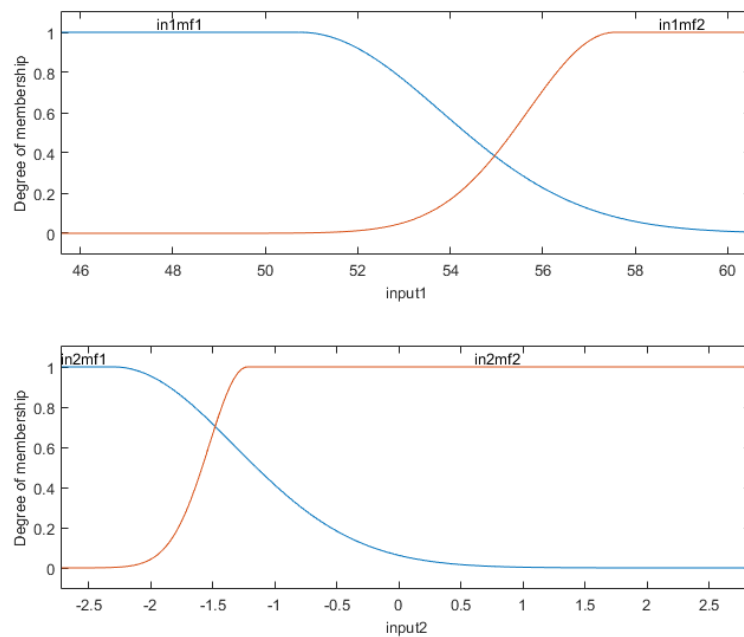
Διαγράμματα Τελικών Συναρτήσεων Συμμετοχής

2 inputs

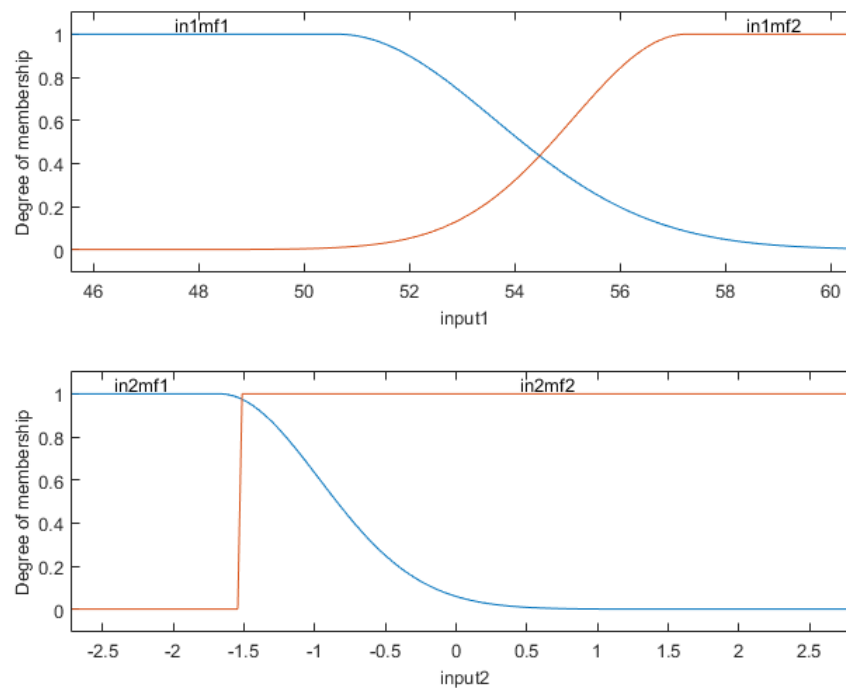
Αρχικά, βλέπουμε τα αποτελέσματα του αλγορίθμου εκπαίδευσης ANFIS με 50 επαναλήψεις.



500 επαναλήψεις



Και τέλος 1500 επαναλήψεις



Παρατηρούμε ότι, οι τελικές συναρτήσεις συμμετοχής έχουν διαφοροποιήσει τα δεδομένα. Άρα ο αριθμός των επαναλήψεων επηρεάζει τον αλγόριθμο εκπαίδευσης ANFIS.

Με την εκτέλεση τους, παρατηρήθηκε ότι το training error μειώθηκε μόνο από την αλλαγή των αριθμών επαναλήψεων εκπαίδευσης:

Για 50 επαναλήψεις:

training error από 0.65255 έως 0.640237

Για 500 επαναλήψεις

training error από 0.65255 έως 0.638144

Για 1500 επαναλήψεις

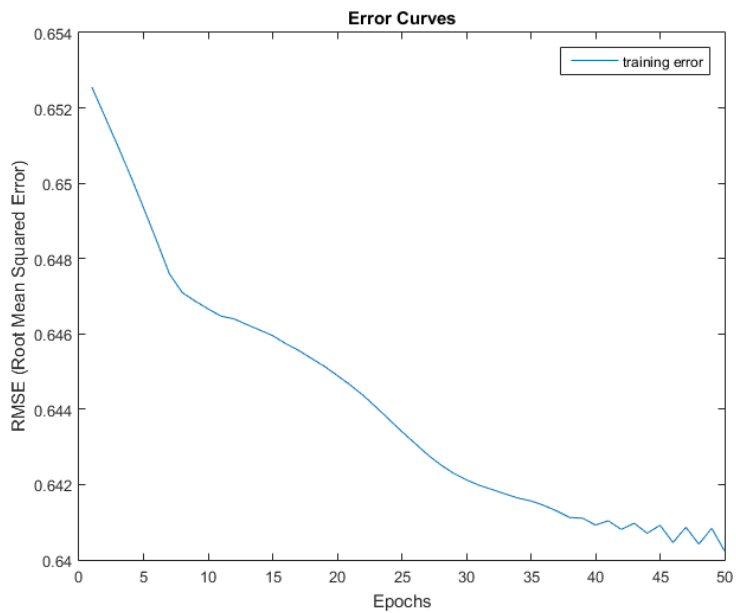
training error από 0.65255 έως 0.634872

Αν και πράγματι, έχουμε καλύτερο σφάλμα στις περισσότερες επαναλήψεις, ο αριθμός αυτός είναι αρκετά μεγάλος και φάνηκε να βάρυνε το υπολογιστικό μας σύστημα. Επίσης, η μείωση του σφάλματος από 500 επαναλήψεις στις 1500 επαναλήψεις, είναι πολύ μικρή για την υπολογιστική δύναμη που χρειάστηκε. Άρα, ένα πρώτο δείγμα ίσως να είναι ότι οι χιλιάδες επαναλήψεις για τα δεδομένα αυτά να είναι υπερβολικές ζημιογόνες.

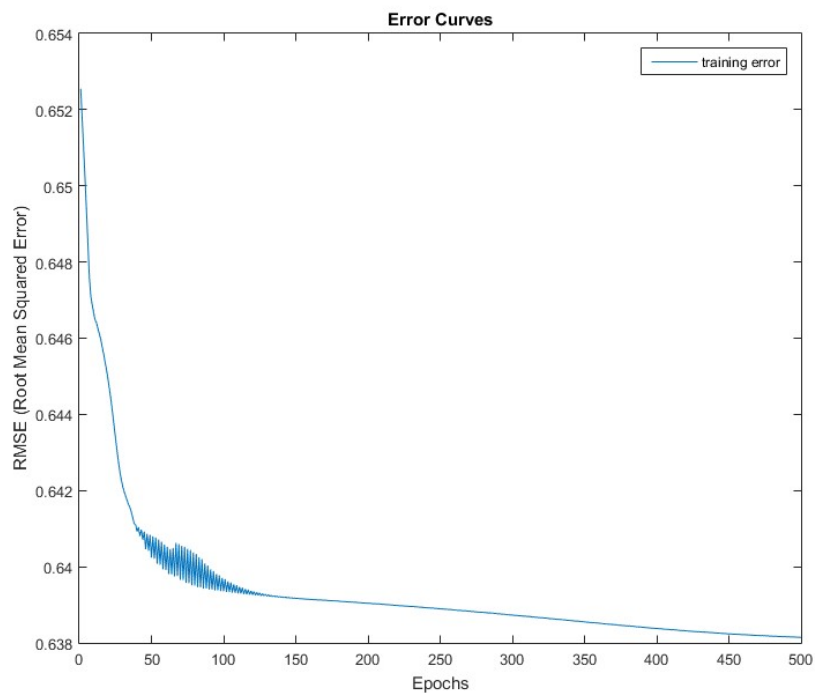
Διαγράμματα Σφάλματος

2 inputs

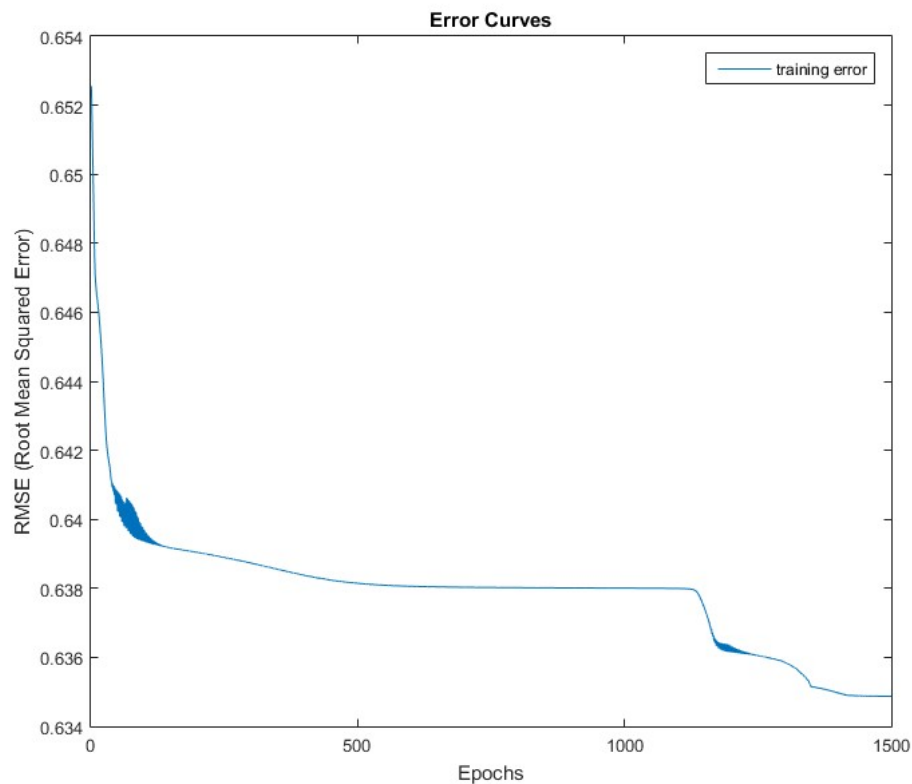
Για 50 επαναλήψεις βλέπουμε



500 επαναλήψεις



1500 επαναλήψεις



Στόχος στο σφάλμα, όπως έχει αναφερθεί, είναι να μειώνεται συνεχώς και στο τέλος να έχουμε κάποιες πιο σταθερές τιμές.

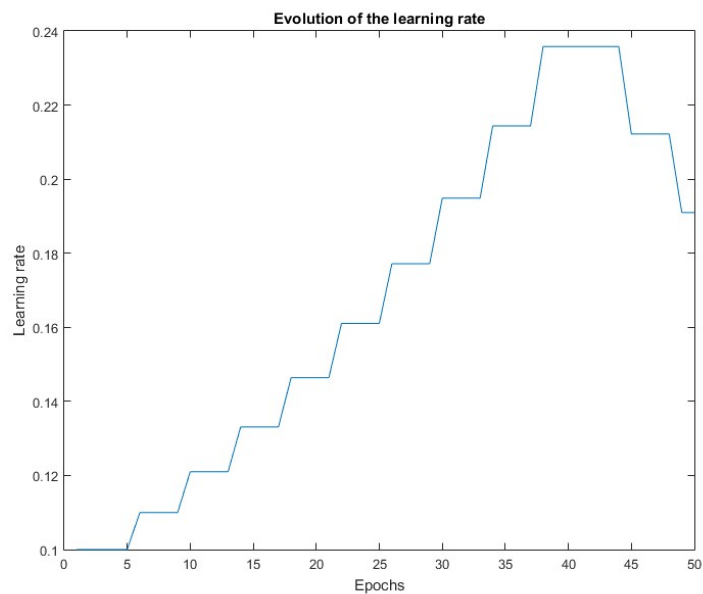
Μπορεί να βελτιώνεται το training error με το πέρασμα των επαναλήψεων, όμως βλέπουμε αρκετές ανωμαλίες στην συμπεριφορά του. Στις 500 και 1500 επαναλήψεις, από τις 50 έως τις 100 επαναλήψεις, βλέπουμε ότι υπάρχουν αυξομειώσεις. Αυτό είναι κάτι που δεν το θέλουμε και στόχος μας είναι η εξομάλυνση του.

Η εξομάλυνση θα έρθει με τον συνδυασμό των μεταβλητών, που θα δούμε αργότερα όταν θα παρουσιαστούν οι καλές περιπτώσεις που βρέθηκαν μετά από δοκιμές.

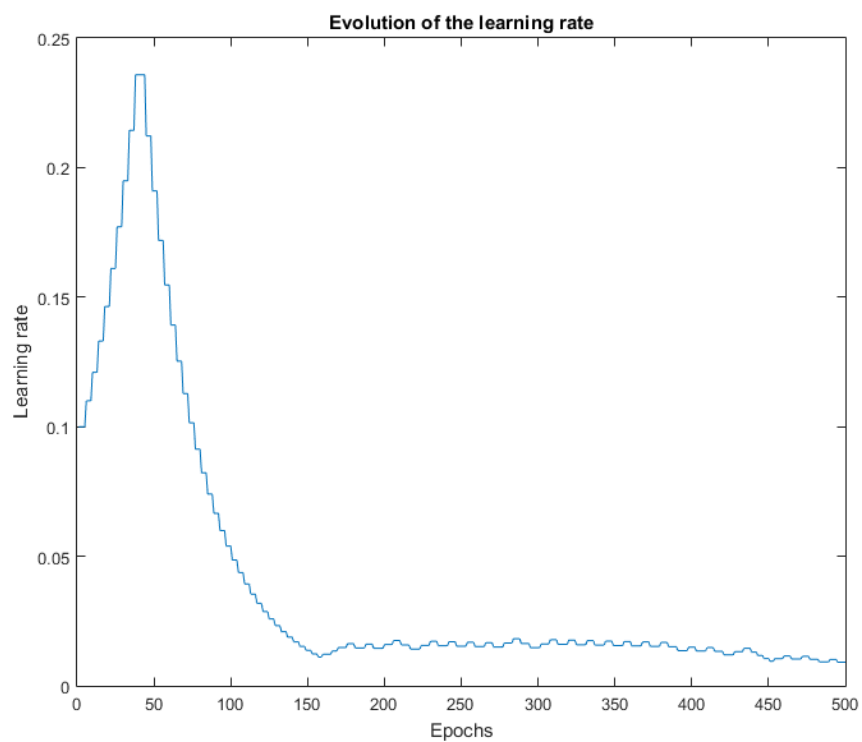
Διαγράμματα Ρυθμού Εκμάθησης

2 inputs

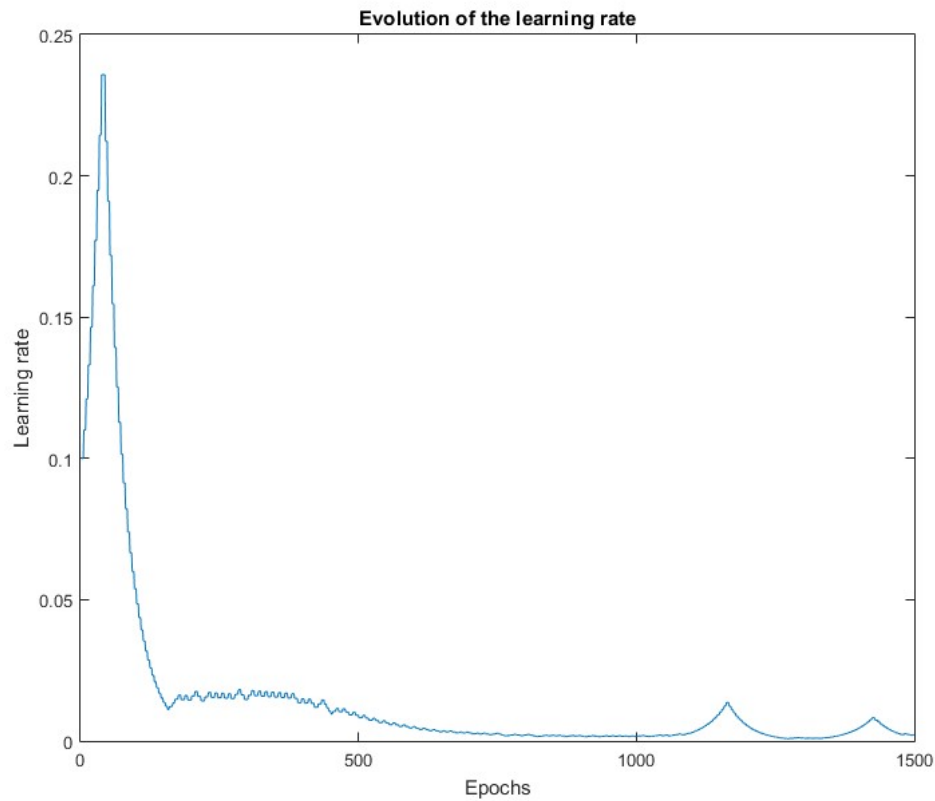
50 επαναλήψεις



500 επαναλήψεις



1500 επαναλήψεις



Έχει αναφερθεί, ότι η εξέλιξη του ρυθμού εκμάθησης του αλγορίθμου εκπαίδευσης ANFIS, πρέπει αρχικά να αυξάνεται και αφού φτάσει στο maximum, στη συνέχεια να μειωθεί για όλη τη διάρκεια της υπόλοιπης εκπαίδευσης.

Όπως βλέπουμε στα διαγράμματα, πράγματι η αύξηση των αριθμών επανάληψης εκπαίδευσης, οδηγεί συνεχώς στην ζητούμενη συμπεριφορά.

Άρα μπορούμε να συμπεράνουμε ότι, με όσο περισσότερες επαναλήψεις εκπαιδευτεί το δίκτυο, τόσο καλύτερο ρυθμό εκμάθησης θα έχει. Αντίστοιχα, με λιγότερες επαναλήψεις, υπάρχει το ρίσκο το να μη φτάσει ποτέ το μέγιστο του ο αλγόριθμος.

Να σημειωθεί, ότι ο κανόνας που λέει ότι πρέπει να μειώνεται ο ρυθμός εκμάθησης με το πέρασμα της εκπαίδευσης δεν ικανοποιείται εντελώς, καθώς έχουμε σημαντικές αυξομειώσεις.

Καταλήγουμε στο συμπέρασμα, ότι ο αριθμός επαναλήψεων εκπαίδευσης στον αλγόριθμο εκπαίδευσης ANFIS, έχει καθοριστικό ρόλο όπως βλέπουμε από τα διαγράμματα.

Αρχικός ρυθμός εκμάθησης (initial_learning_rate) για τις τρεις περιπτώσεις των 2 inputs

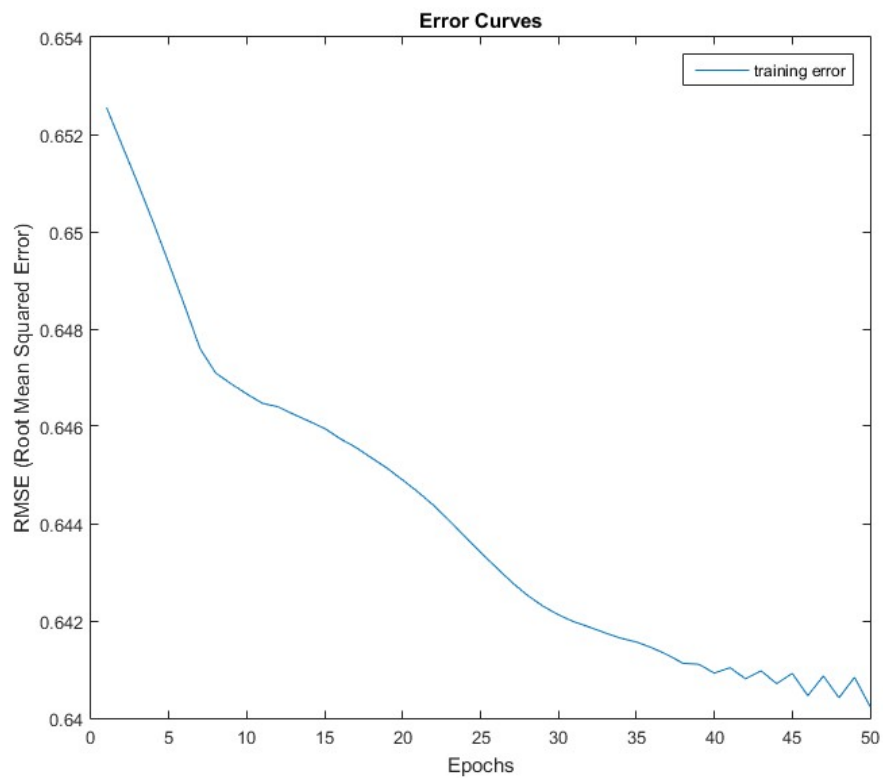
Στις προηγούμενες εκτελέσεις, ο αρχικός ρυθμός εκμάθησης ήταν ίσος με 0.1.

Τώρα θα γίνουν δοκιμές για τις παρακάτω τιμές:

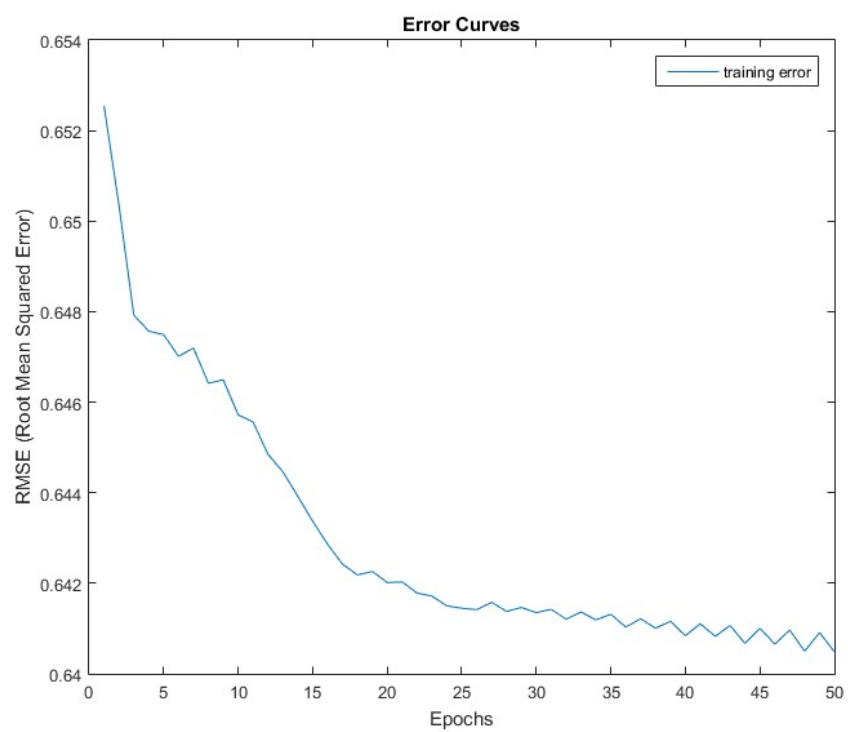
- 0.1
- 0.3
- 0.01
- 0.5

Διαγράμματα Σφάλματος

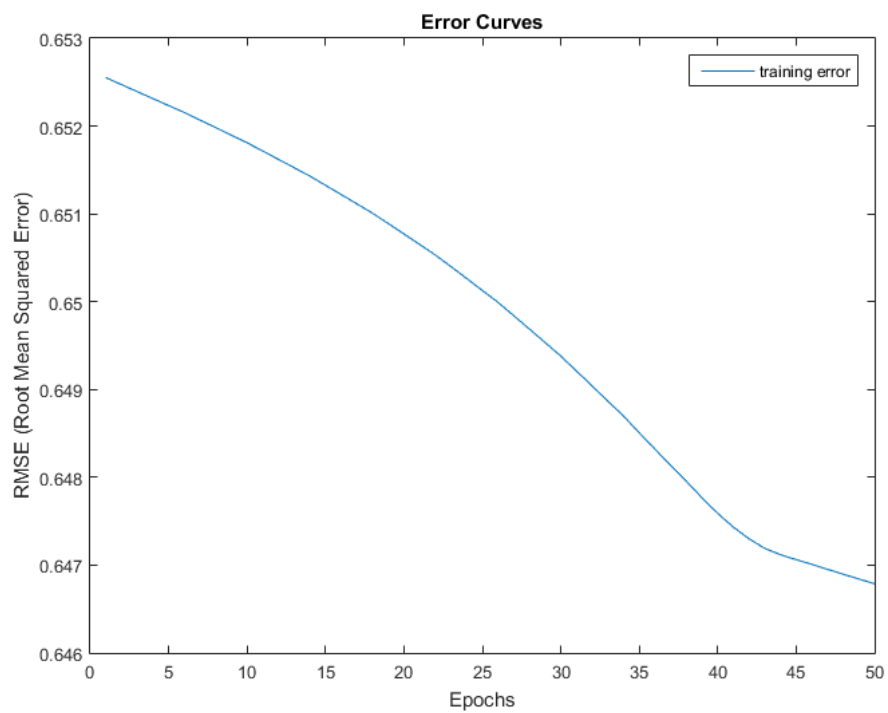
Για 0.1



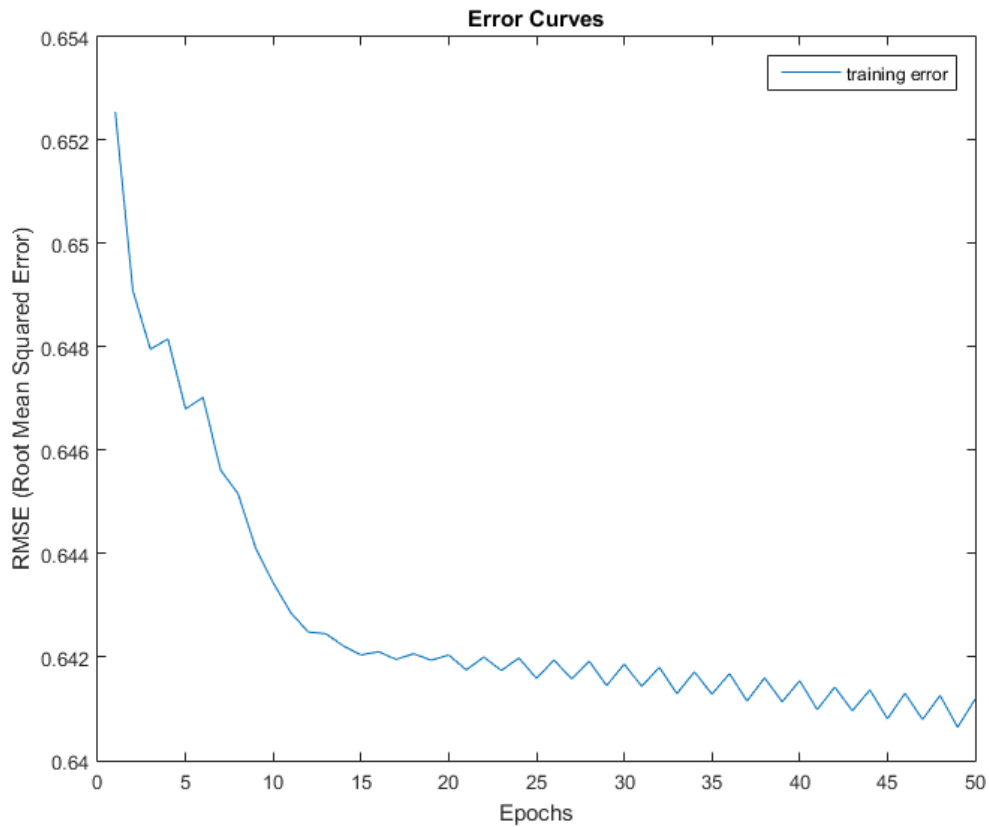
Για 0.3



Για 0.01



Για 0.5



Παρατηρούμε ότι με τις διάφορες τιμές του αρχικού ρυθμού εκμάθησης, πετυγχένεται ανά σημεία η εξομάλυνση που ζητάμε και η μείωση των αυξομειώσεων.

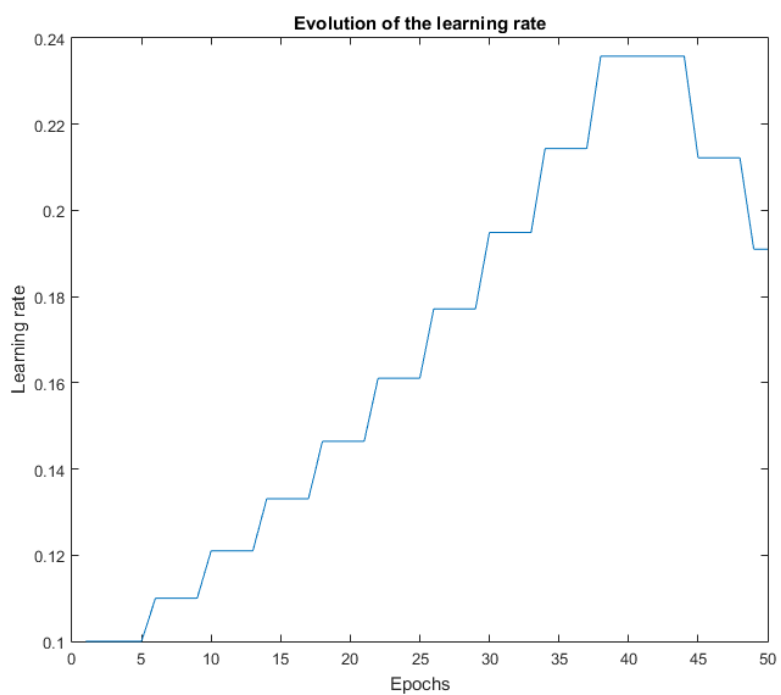
Κυρίως για 0.01, δεν έχουμε σε καμία περίπτωση αυξομείωση, κάτι που είναι επιθυμητό.

Όμως, δεν αρκεί μόνο αυτό για να κρίνει αν μια τιμή είναι αυτή ζητάμε.

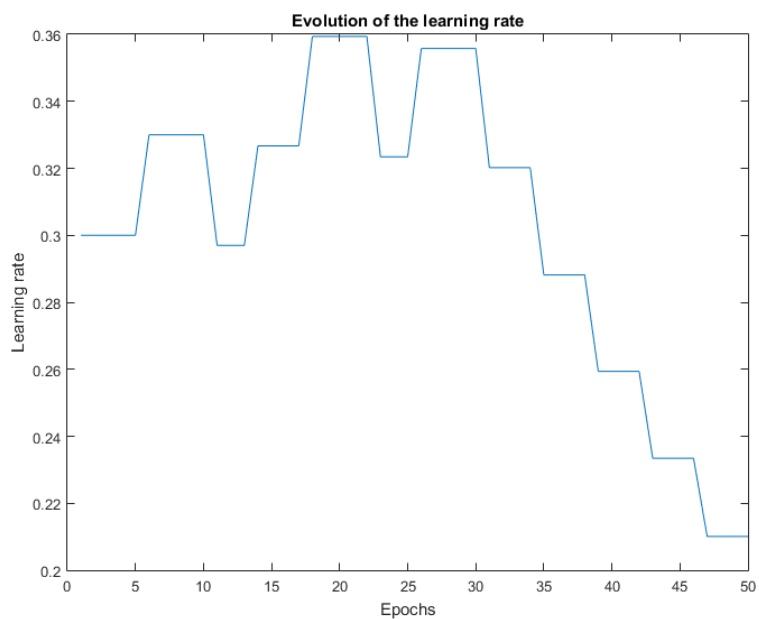
Πάντως, αρχίζουμε και βλέπουμε ότι σε συνδυασμό με τα προηγούμενα συμπεράσματα, έχουμε εικόνα για το πως ακριβώς επηρεάζουν τα δεδομένα οι μεταβλητές. Όποτε αργότερα, σε συνδυαστικές δοκιμές θα προσπαθήσουμε να έχουμε

Διαγράμματα Ρυθμού Εκμάθησης

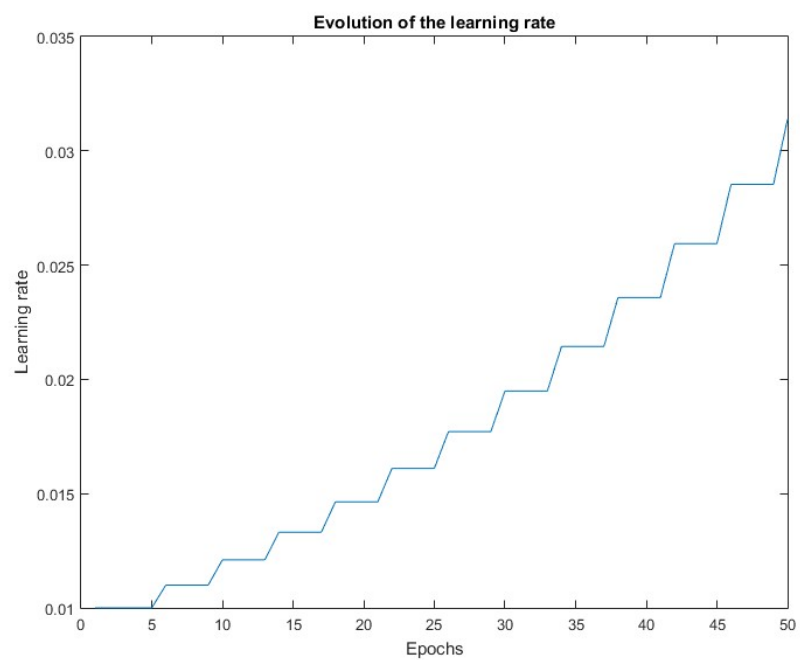
Για 0.1



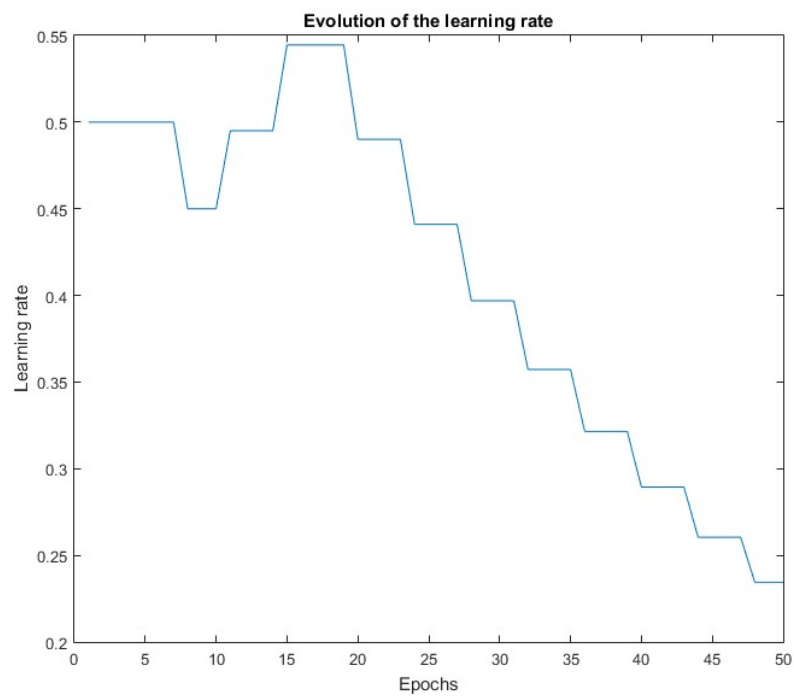
Για 0.3



$\Gamma \alpha 0.01$



$\Gamma \alpha 0.05$



Παρατηρούμε διαφοροποιήσεις στις περιπτώσεις μεταξύ τους.

- Στο 0.1 έχουμε μια αποδεκτή συμπεριφορά, αλλά με πιθανή αύξηση του αριθμού επανάληψης εκπαίδευσης, να είχαμε προτιμότερη συμπεριφορά.
- Στο 0.3 υπάρχουν σίγουρα περισσότερα βελτίωσης, καθώς αυτό το αποτέλεσμα δεν μας ικανοποιεί. Φτάνει ένα μέγιστο, αλλά τελικώς δεν ήταν μέγιστο και έχει συνολικά 3 αυξομειώσεις. Εμείς χρειαζόμαστε μία και μετά μείωση.
- Στο 0.01 έχουμε μόνο κορύφωση και δεν φτάνει ποτέ το μέγιστο φαινομενικά. Πιθανή αύξηση του αριθμού επανάληψης να μας δώσει καλύτερα αποτελέσματα.
- Στο 0.5 έχουμε οριακά μη αποδεκτή συμπεριφορά επίσης, καθώς αυξάνεται δύο φορές και στη συνέχεια μειώνεται. Εδώ ενδεχομένως, να μη χρειαζόμαστε αύξηση του αριθμού επανάληψης, οπότε θα πρέπει να πειραματιστούμε πιο πολύ.

Σύνολο Δοκιμών για τις τρεις περιπτώσεις των 2 inputs

Μετά από εξαντλητικούς ελέγχους και δοκιμές, θα παρουσιαστούν παρακάτω τα πιθανά σύνολα δοκιμών για τις τρεις περιπτώσεις των 2 inputs, που δίνουν ικανοποιητικά αποτελέσματα.

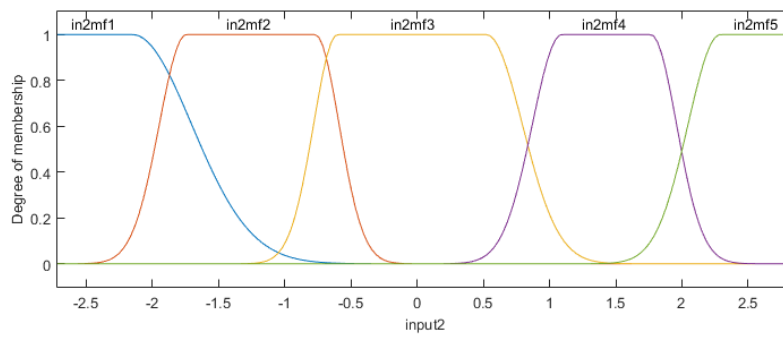
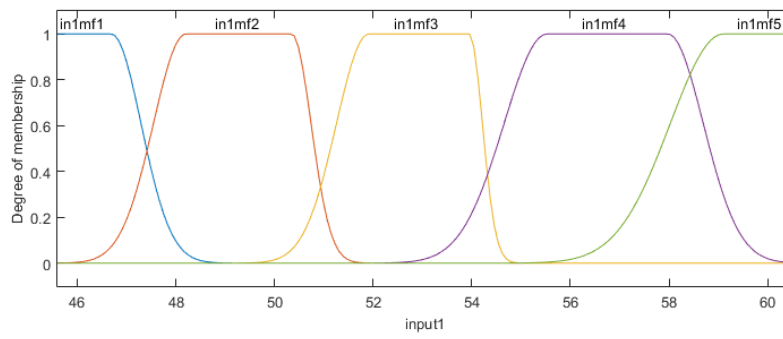
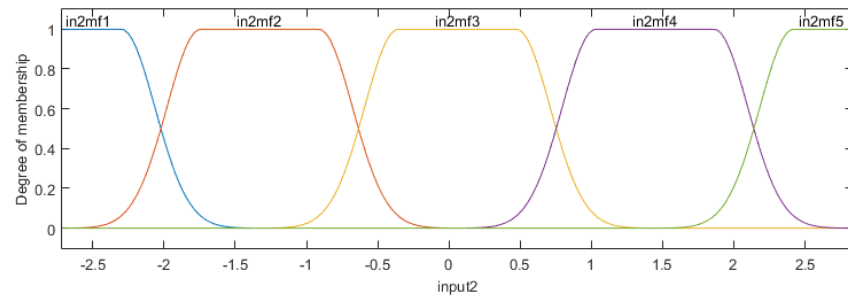
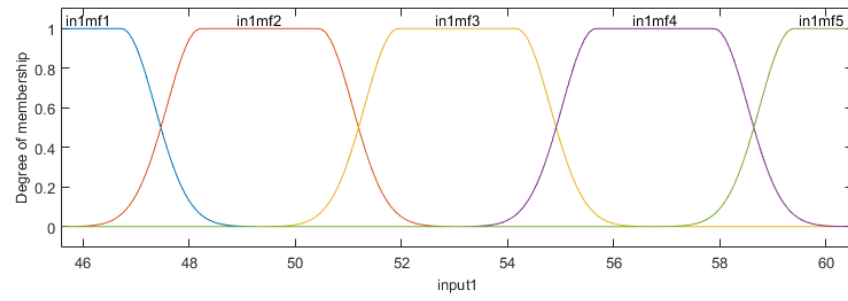
2 inputs

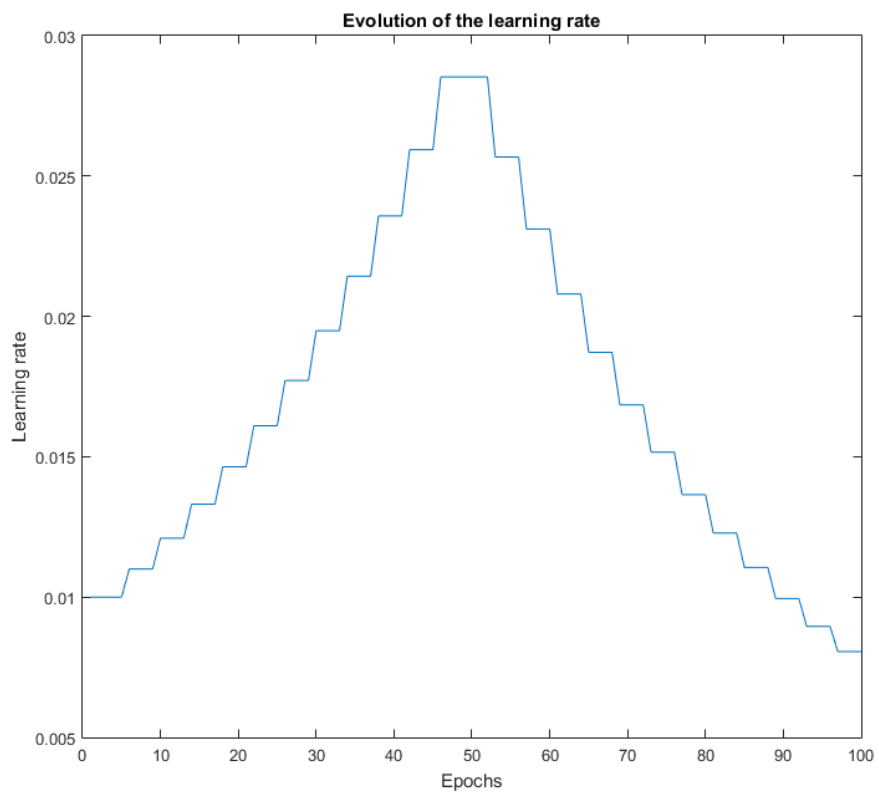
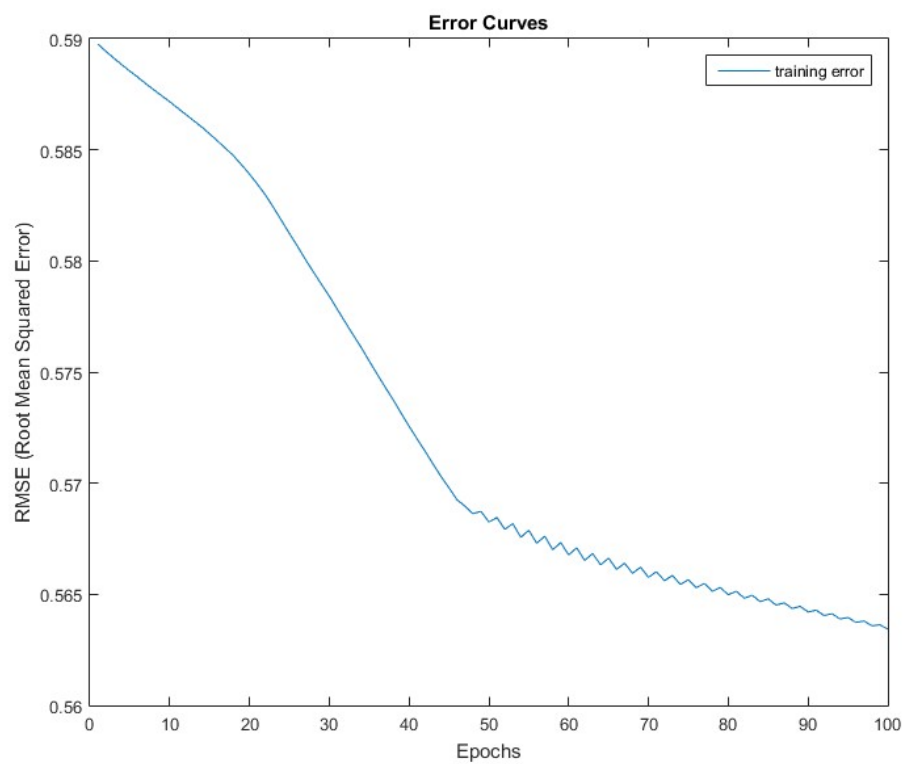
Το σύνολο δοκιμών που κρίθηκε ως καλύτερο για την περίπτωση των 2 inputs είναι:

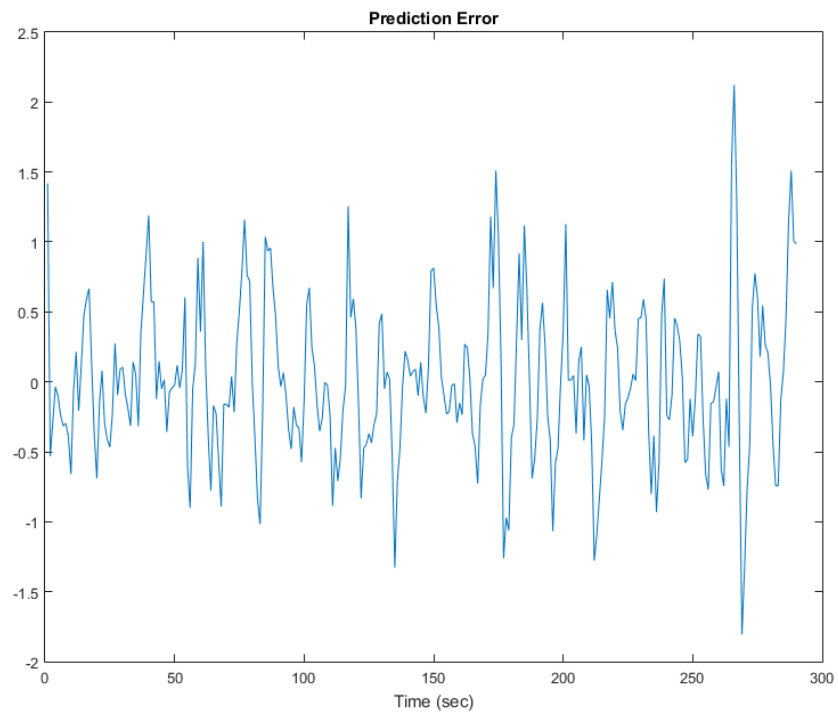
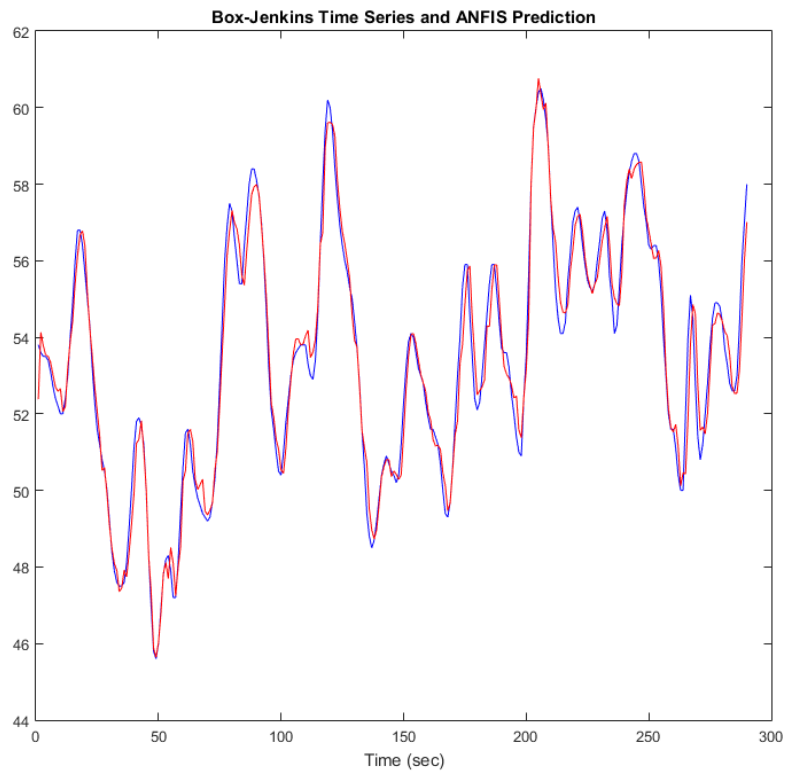
- fuzzySets = 5
- epoch_number = 100
- initial_learning_rate = 0.01
- mfType = gauss2mf
- outType = linear

Με training error από 0.58975 έως 0.56342

Παρακάτω παρουσιάζονται τα γραφήματα







Το παραπάνω σύνολο δοκιμών, κρίθηκε ως αποδεκτό καθώς:

- Στο διάγραμμα σφάλματος, φαίνεται ότι οι αυξομειώσεις είναι μικρές σε εύρος και στο μισό κομμάτι της εκπαίδευσης περίπου έχουμε εξομάλυνση
- Στο διάγραμμα του ρυθμού εκπαίδευσης βλέπουμε ότι υπάρχει η επιθυμητή συμπεριφορά. Αρχικά αυξάνεται, στη συνέχεια πιάνει ένα μέγιστο και μειώνεται μέχρι το τέλος της εκπαίδευσης.
- Η χρονική διάρκεια της εκπαίδευσης ήταν σε αποδεκτά πλαίσια. Το υπολογιστικό σύστημα δεν ταλαιπωρήθηκε ιδιαίτερα. Ενώ αντίθετα, με δεδομένα που πίεζαν το σύστημα, δεν είχαμε αποδεκτή συμπεριφορά.

2 inputs no feedback

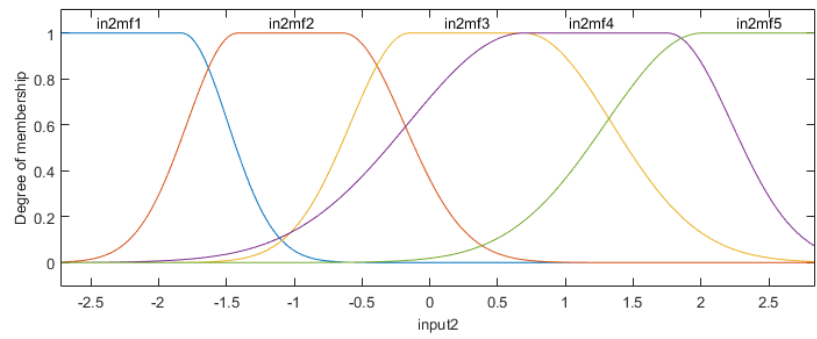
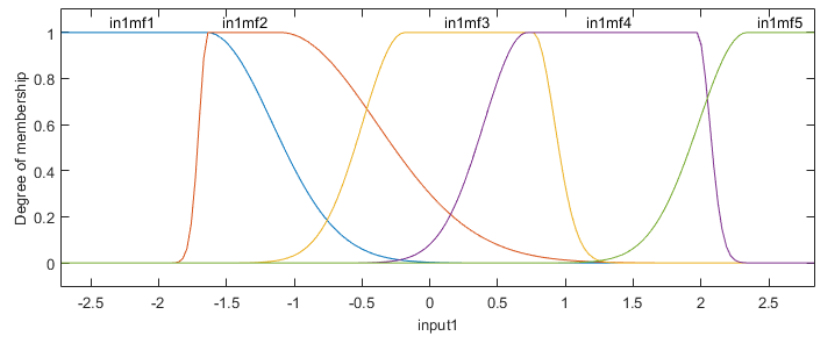
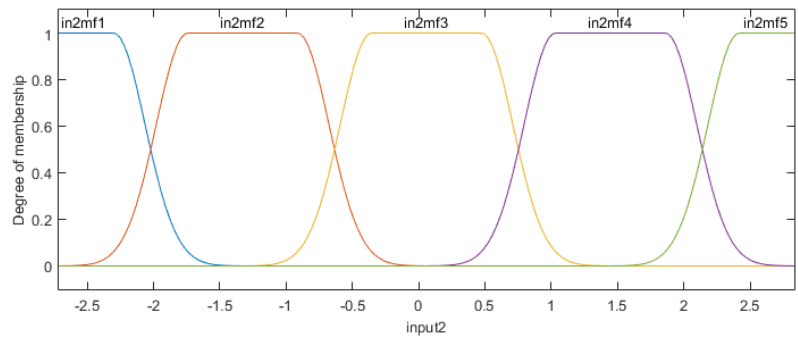
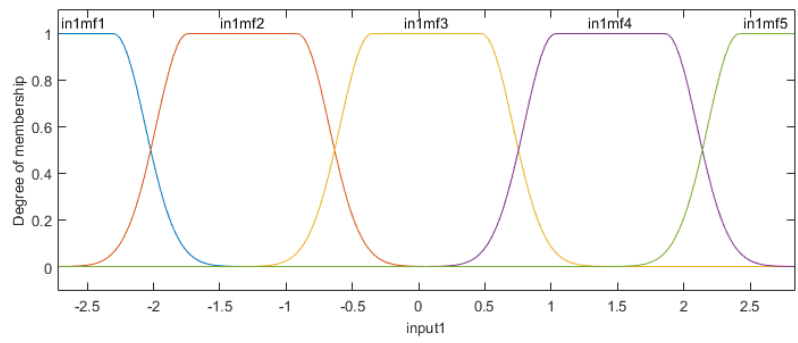
Το σύνολο δοκιμών που κρίθηκε ως καλύτερο για την περίπτωση των 2 inputs no feedback είναι:

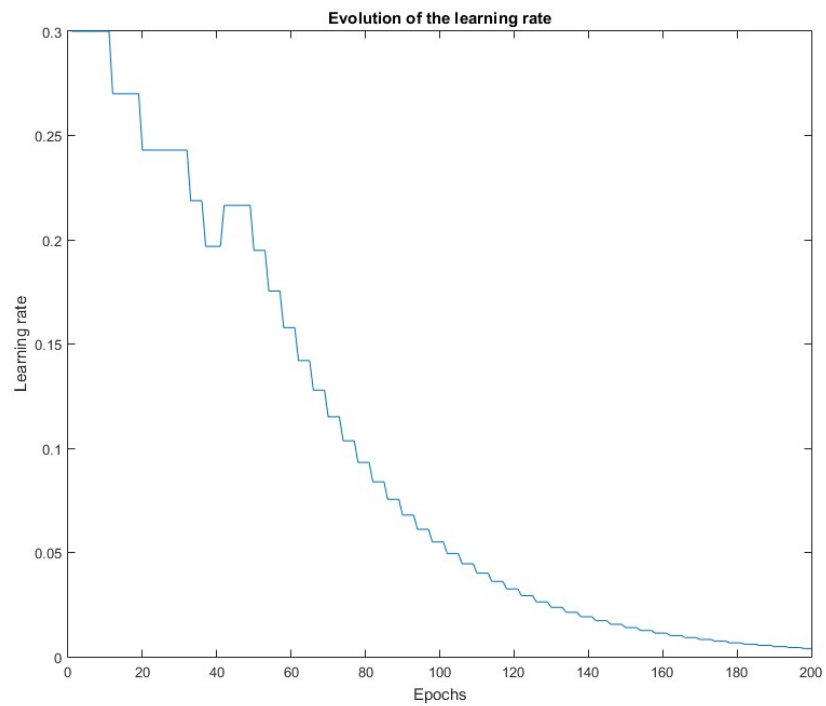
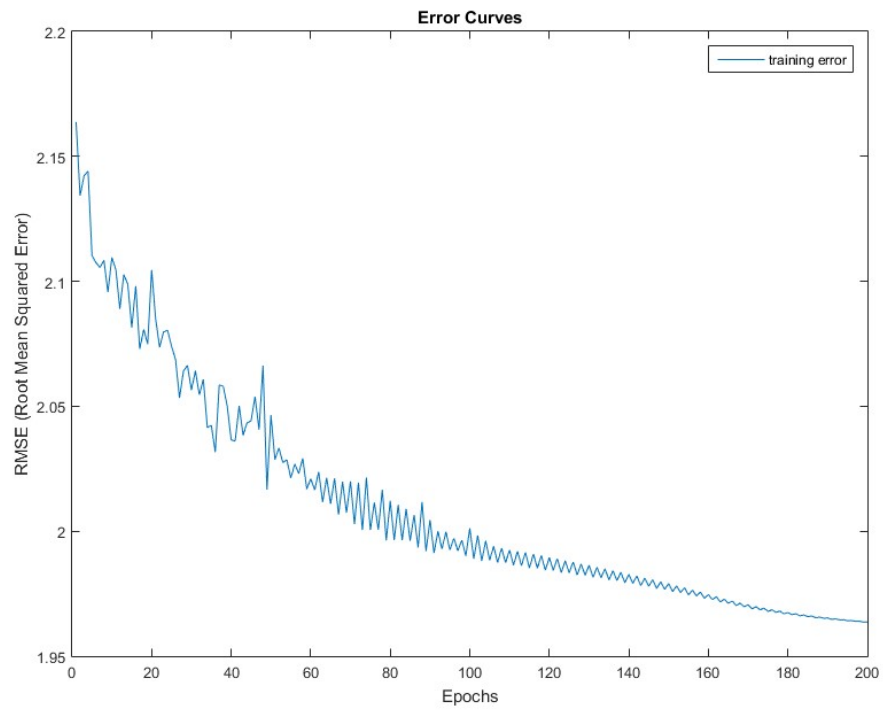
- fuzzySets = 5
- epoch_number = 200
- initial_learning_rate = 0.3
- mfType = gauss2mf
- outType = linear

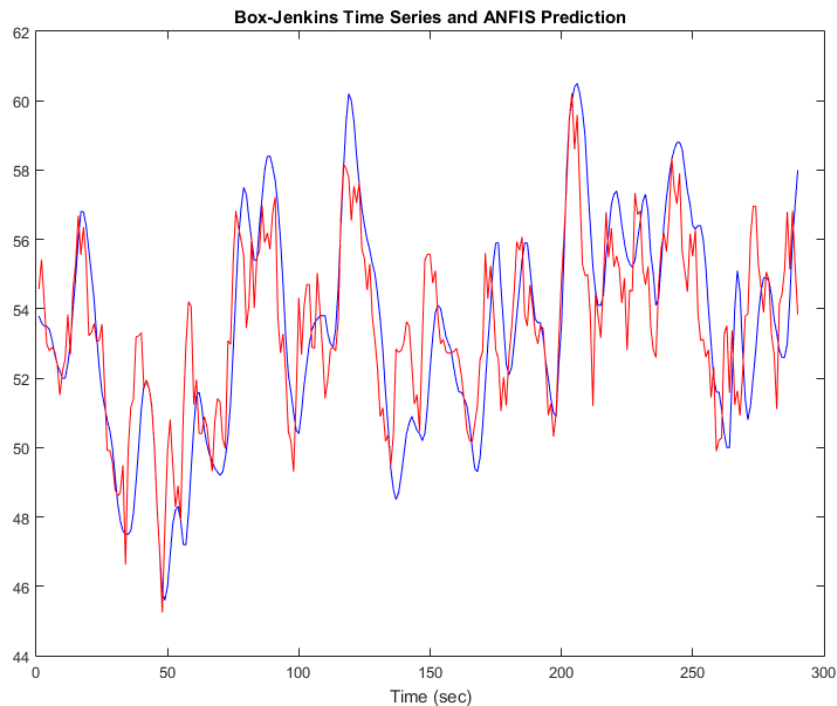
Με training error 2.16377 από έως 1.96364

Σε αυτή τη περίπτωση του 2 inputs no feedback, υπήρξαν μεγάλες δυσκολίες στην εύρεση ενός αποδεκτού συνόλου. Το σφάλμα σε αυτή τη περίπτωση είναι πολύ μεγάλο, και καμία δοκιμή δεν μπορούσε να το φτάσει σε ανεκτικά επίπεδα. Ακόμα και με πολλές επαναλήψεις, το σφάλμα ήταν πάντα λίγο πιο κάτω από το 2. Το οποίο γενικά στα νευρωνικά δίκτυα, είναι πάρα πολύ υψηλό. Άρα με την σειρά μας, αποδεχτήκαμε ένα σύνολο δοκιμών που δεν είναι δανικό.

Παρακάτω παρουσιάζονται τα γραφήματα







Όπως παρατηρούμε από τα γραφήματα, δεν έχουμε τα ιδανικά αποτελέσματα που θα θέλαμε.

- Το διάγραμμα του σφάλματος έχει πάρα πολλές αυξομειώσεις και ασταθής συμπεριφορά, αλλά μετά από ελέγχους που έγιναν, ήταν το μόνο αποδεκτό.
- Το διάγραμμα του ρυθμού εκμάθησης, ενδέχεται να ξεκινάει από το μέγιστο και μετά να μειώνεται. Αλλά θα έπρεπε να αυξάνεται.
- Συγκριτικά με τα πραγματικά αποτελέσματα, βλέπουμε ότι είναι αρκετά εκτός. Άρα καταλαβαίνουμε ότι αυτά τα δεδομένα χωρίς feedback, δεν μας κάνουν.

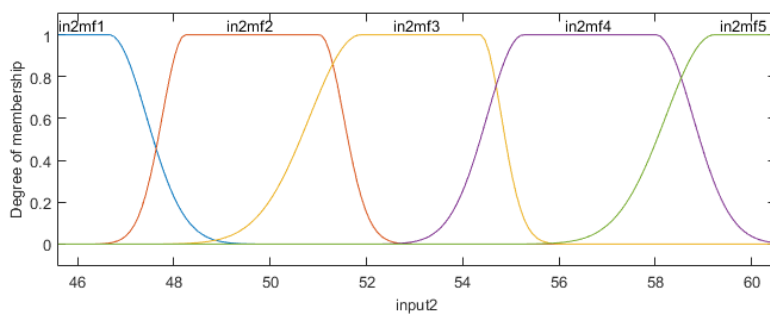
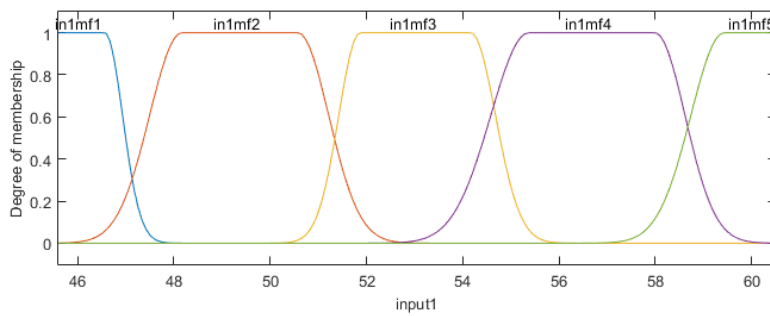
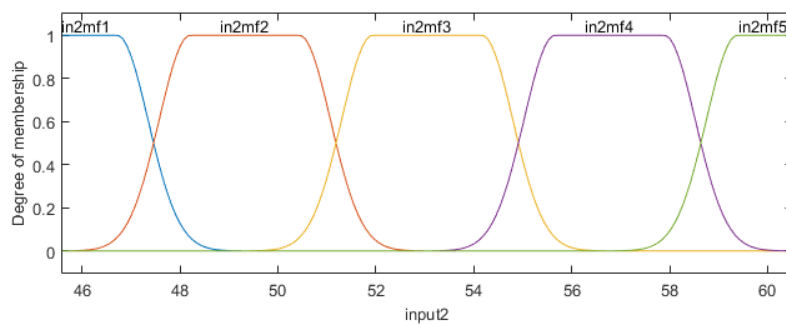
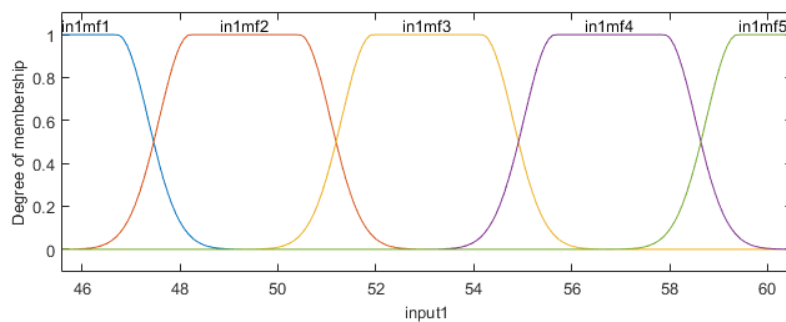
2 inputs feedback

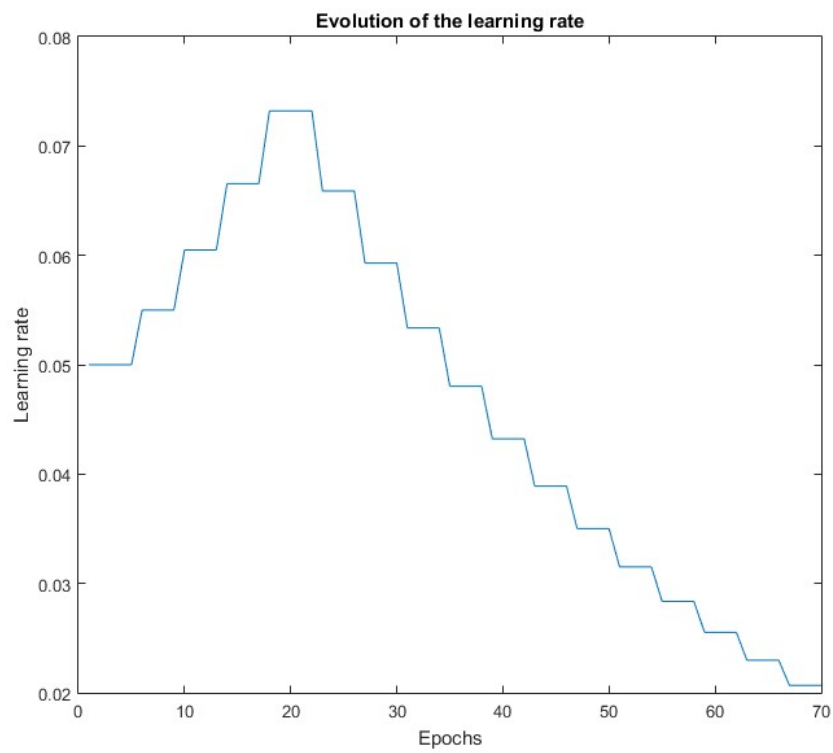
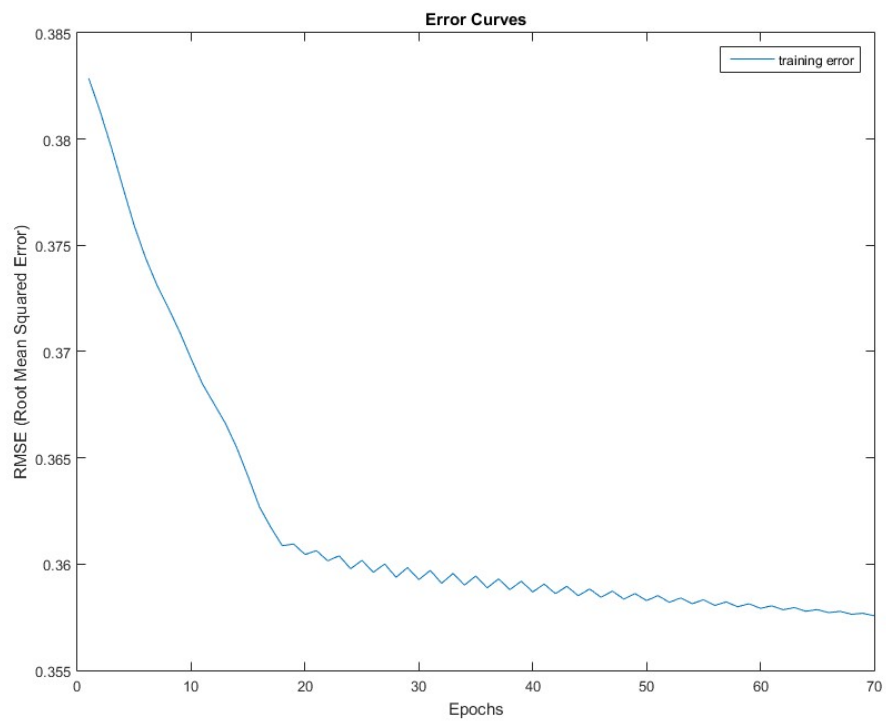
Το σύνολο δοκιμών που κρίθηκε ως καλύτερο για την περίπτωση των 2 inputs feedback είναι:

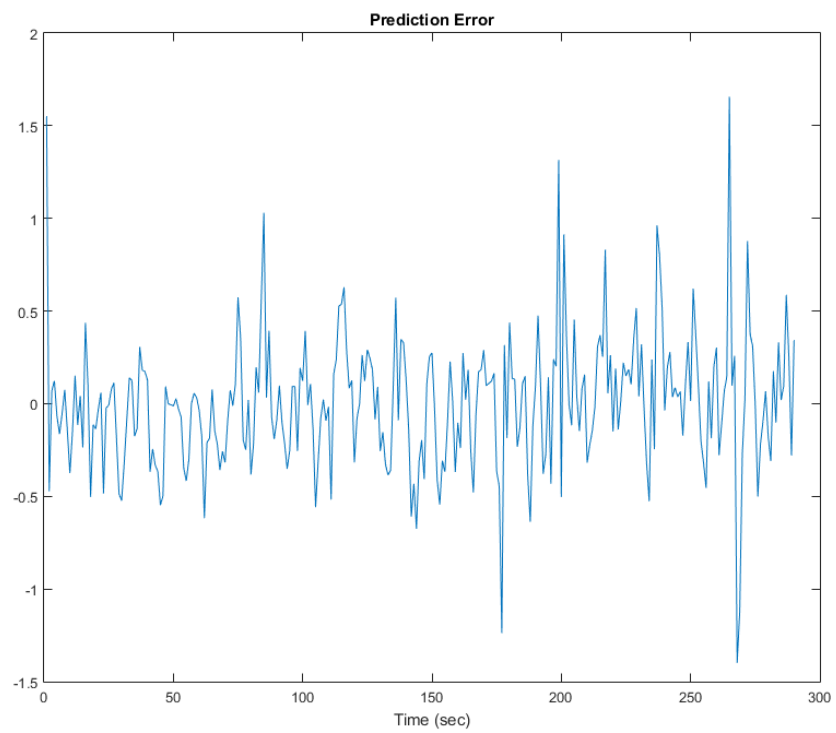
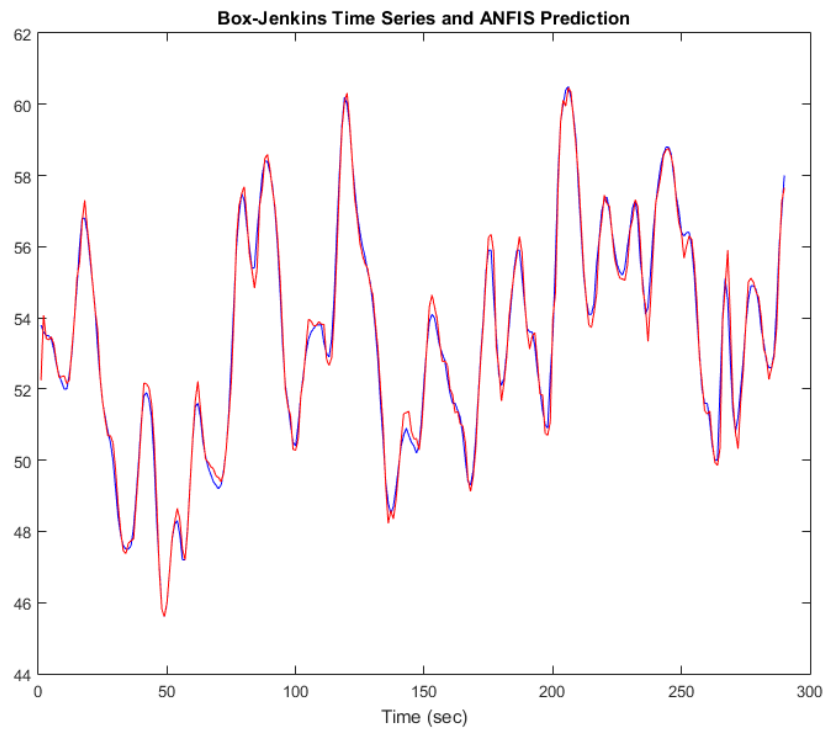
- fuzzySets = 5
- epoch_number = 70
- initial_learning_rate = 0.05
- mfType = gauss2mf
- outType = linear

Με training error 0.382866 από έως 0.357571

Παρακάτω παρουσιάζονται τα γραφήματα







Όπως βλέπουμε, η περίπτωση με 2 inputs feedback είναι η καλύτερη από θέμα αποτελεσμάτων και σφάλματος.

- Το σφάλμα είναι αρκετά μικρό, από τις δικές έφτασε μέχρι και 0.34
- Στο διάγραμμα σφάλματος βλέπουμε μια καλή συμπεριφορά με λίγες αυξομειώσεις και στο τέλος να υπάρχει η ζητούμενα συνέχεια των τιμών.
- Στο διάγραμμα του ρυθμού εκμάθησης βλέπουμε ότι υπάρχει η ζητούμενη συμπεριφορά που μπορούμε να την δεχθούμε σαν ικανοποιητική.

Μια γενική παρατήρηση για την κατηγορία των 2 inputs είναι, ότι φαίνεται να βγάζει καλύτερα αποτελέσματα με 5 ασαφή σύνολα ανά είσοδο. Άρα αυτό σημαίνει ότι ενδεχομένως τα 2 inputs να μην μας είναι αρκετά και μελλοντικά με τα 3 και τα 6 inputs, να δούμε καλύτερα αποτελέσματα.

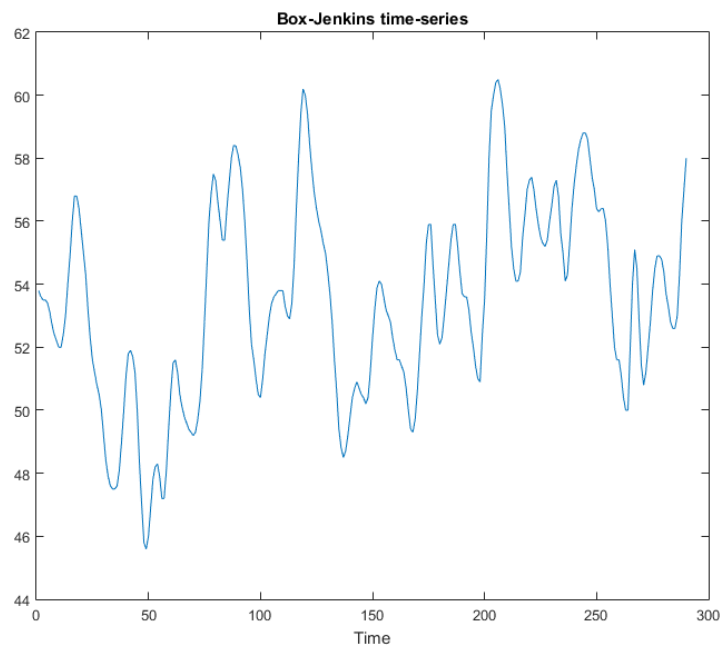
3 Inputs

Σε αυτήν την ενότητα, θα μελετήσουμε την περίπτωση των 3 inputs.

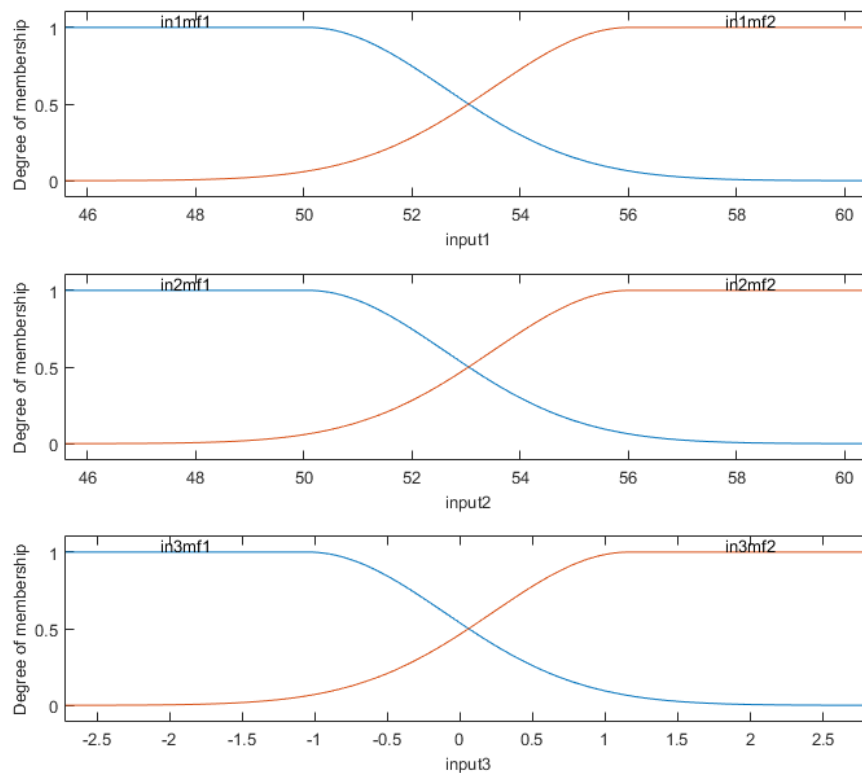
Οι αρχικές μας δοκιμές θα γίνουν με τις παρακάτω παραμέτρους

- fuzzySets = 2
- epoch_number = 30
- initial_learning_rate = 0.1
- mfType = gauss2mf
- outType = linear

Box-Jenkins για τις πραγματικές τιμές μας



Γράφημα αρχικών συναρτήσεων συμμετοχής



Βλέπουμε τώρα, ότι έχουμε 3 εισόδους για τον αλγόριθμο μας συγκριτικά με τις 2 εισόδους που εξετάσαμε προηγουμένως.

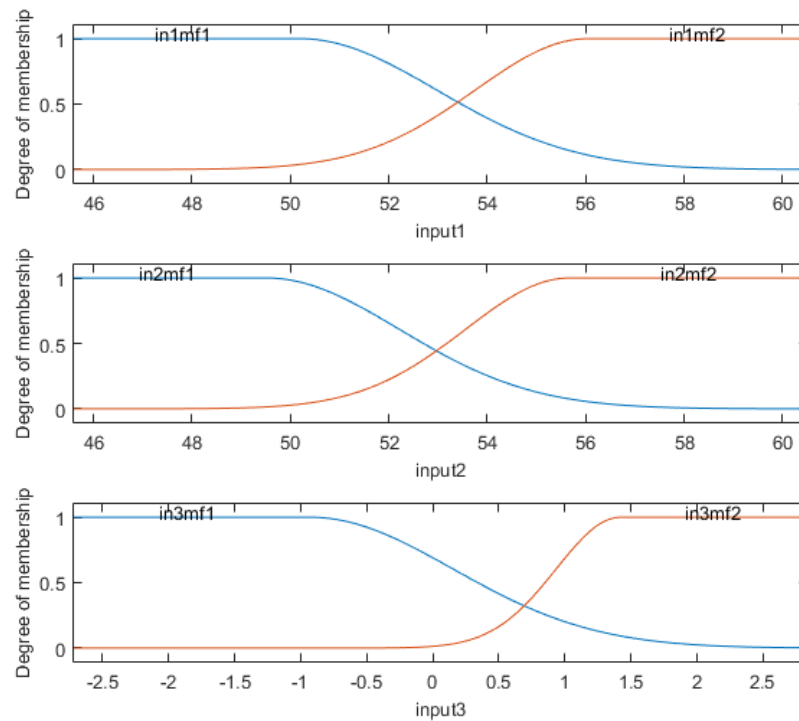
Κατά κανόνα, όσο περισσότερες εισόδους έχουμε, τόσο καλύτερα αποτελέσματα θα πάρουμε. Και σε συνδυασμό με το πλήθος των ασαφή συνόλων που θα εξετάσουμε παρακάτω, μπορούμε να μειώσουμε το σφάλμα αρκετά.

Όμως, όπως είναι λογικό, το υπολογιστικό μας σύστημα θα χρειαστεί περισσότερους πόρους καθώς ο αλγόριθμος εκπαίδευσης ANFIS, θα χρειαστεί περισσότερο χρόνο και υπολογιστική δύναμη.

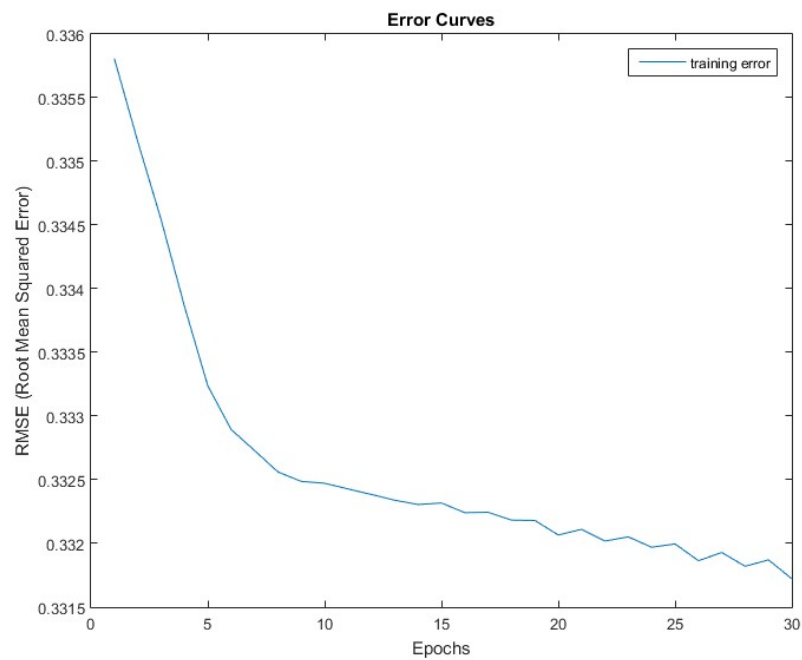
Αν μάλιστα σκεφτούμε ότι μπορούμε να αυξήσουμε τον αριθμό των ασαφή συνόλων ανά είσοδο, τον αριθμό επαναλήψεων εκπαίδευσης και τον αρχικό ρυθμό εκμάθησης, κάποιοι πολύ μεγάλοι αριθμοί θα καταστήσουν την εκπαίδευση πάρα πολύ αργή.

Για αυτό και προτάθηκε να χρησιμοποιηθούν 2 έως 3 ασαφή σύνολα ανά είσοδο, στην περίπτωση των 3 inputs.

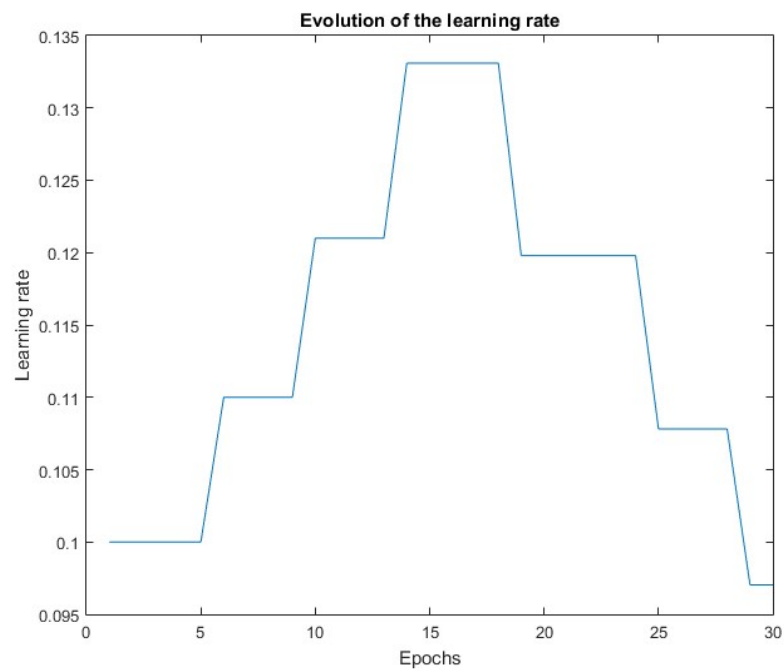
Γράφημα τελικών συναρτήσεων συμμετοχής



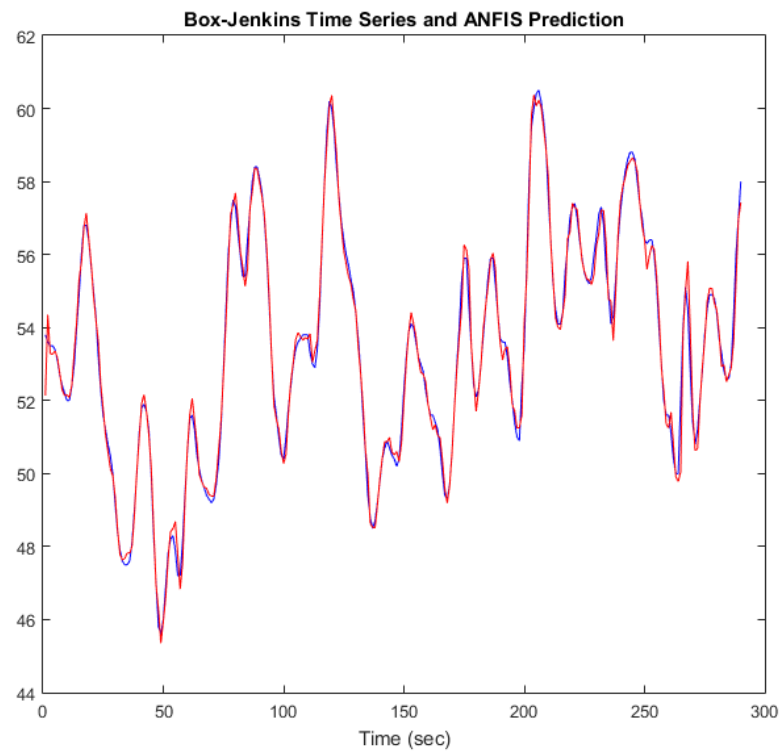
Διάγραμμα Σφάλματος



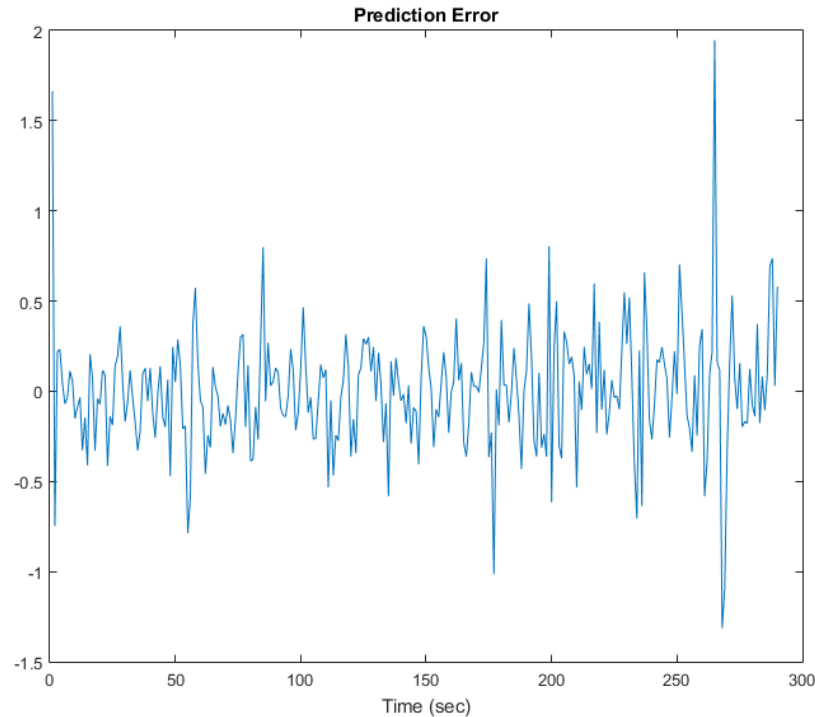
Διάγραμμα ρυθμού εκμάθησης



Διάγραμμα σύγκρισης των δεδομένων με τα πραγματικά



Γράφημα σφάλματος για κάθε δείγμα του συνόλου εκμάθησης



Αρχική μας παρατήρηση είναι, ότι συγκριτικά με τις τρεις περιπτώσεις των 2 inputs, το training error φτάνει σε επίπεδα του ~ 0.331 , που είναι το καλύτερο που έχουμε πετύχει μέχρι στιγμής χωρίς να έχουμε ούτε πολλές επαναλήψεις ούτε παραπάνω από 2 ασαφή σύνολα ανά είσοδο.

Από το διάγραμμα του σφάλματος βλέπουμε ότι έχουμε μια ομαλή καμπύλη με μικρές σε εύρος αυξομειώσεις. Είναι λογικό καθώς το σφάλμα δεν είχε μεγάλη μείωση κατά την διάρκεια της εκπαίδευσης από τον αλγόριθμο. Όμως, είναι μια συμπεριφορά που την ζητάμε.

Στο διάγραμμα ρυθμού εκμάθησης, βλέπουμε ότι έχουμε ακριβώς την συμπεριφορά που ζητάμε.

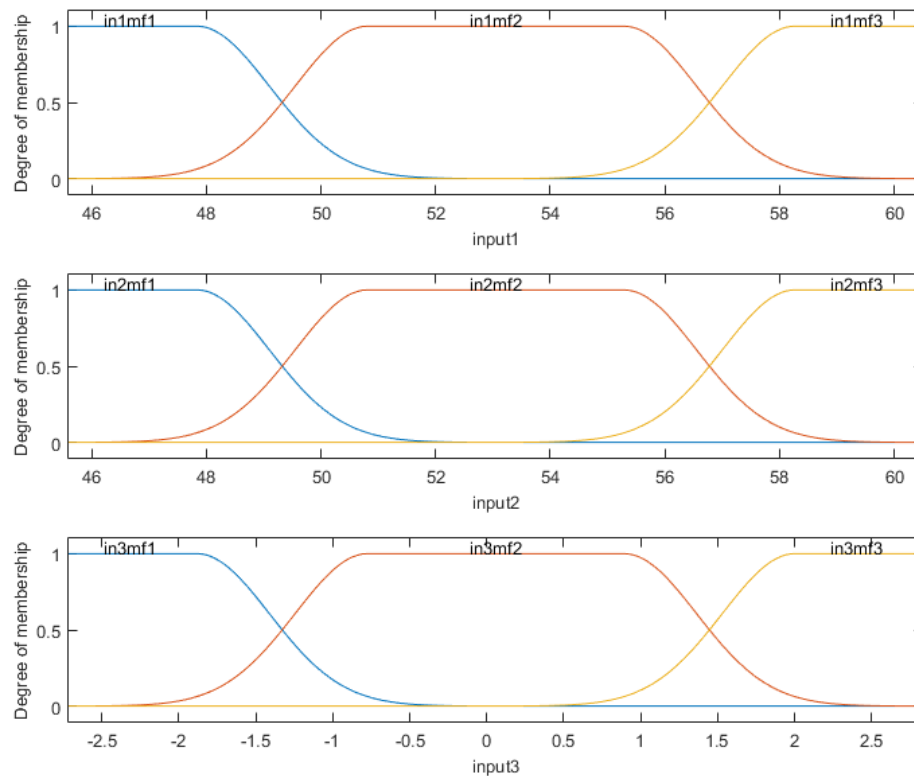
Αύξηση -> Μέγιστο -> Μείωση

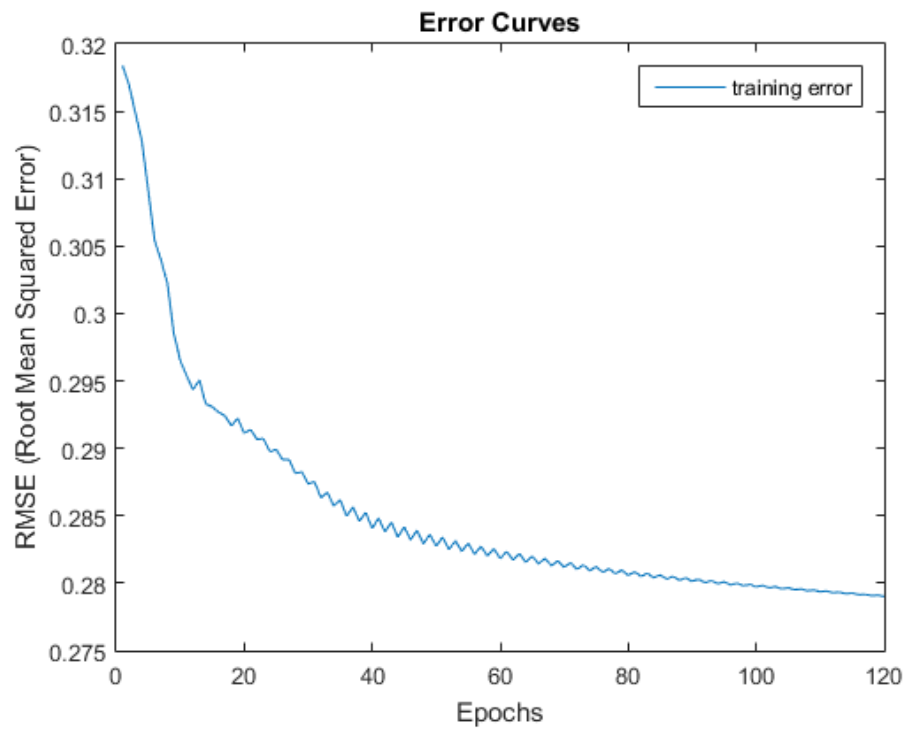
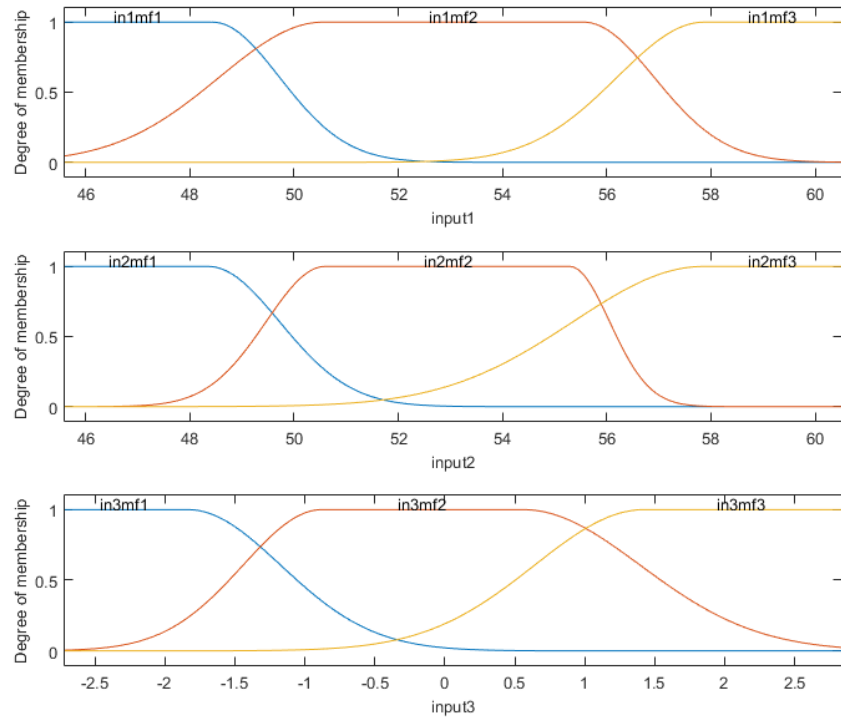
Δεν αυξάνεται ποτέ ξανά και θα μπορούσαμε να την αποδεχθούμε.

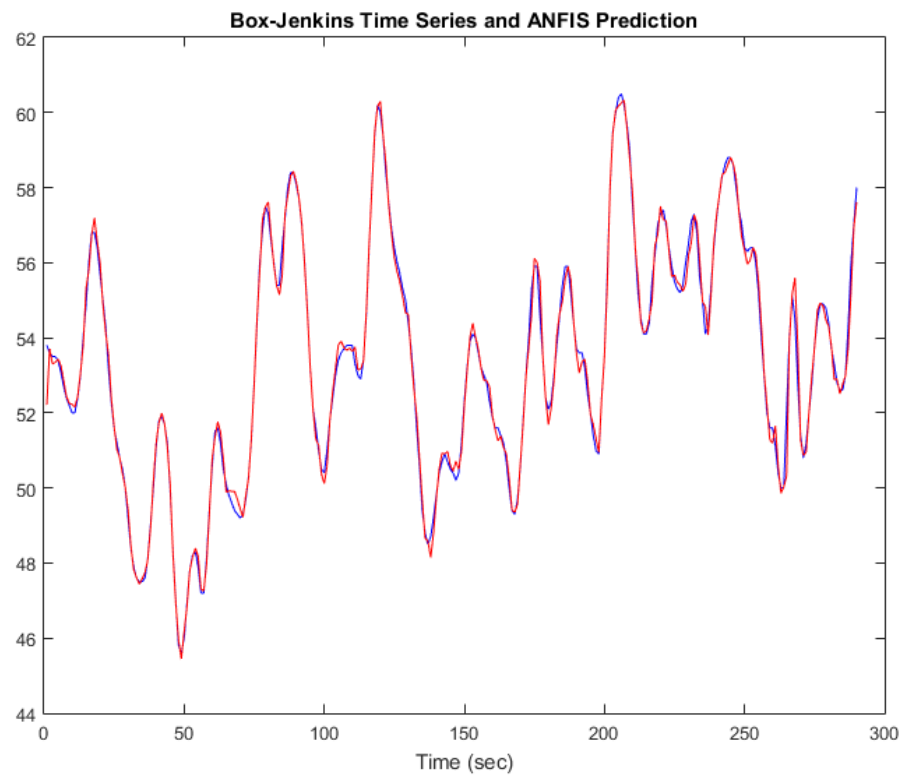
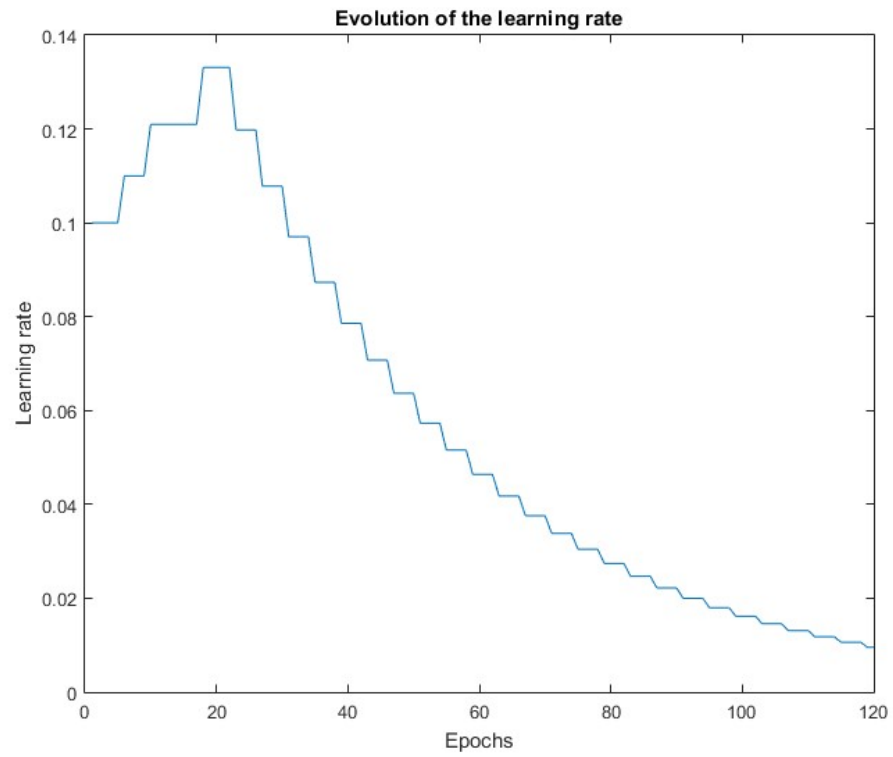
Σύνολο Δοκιμών

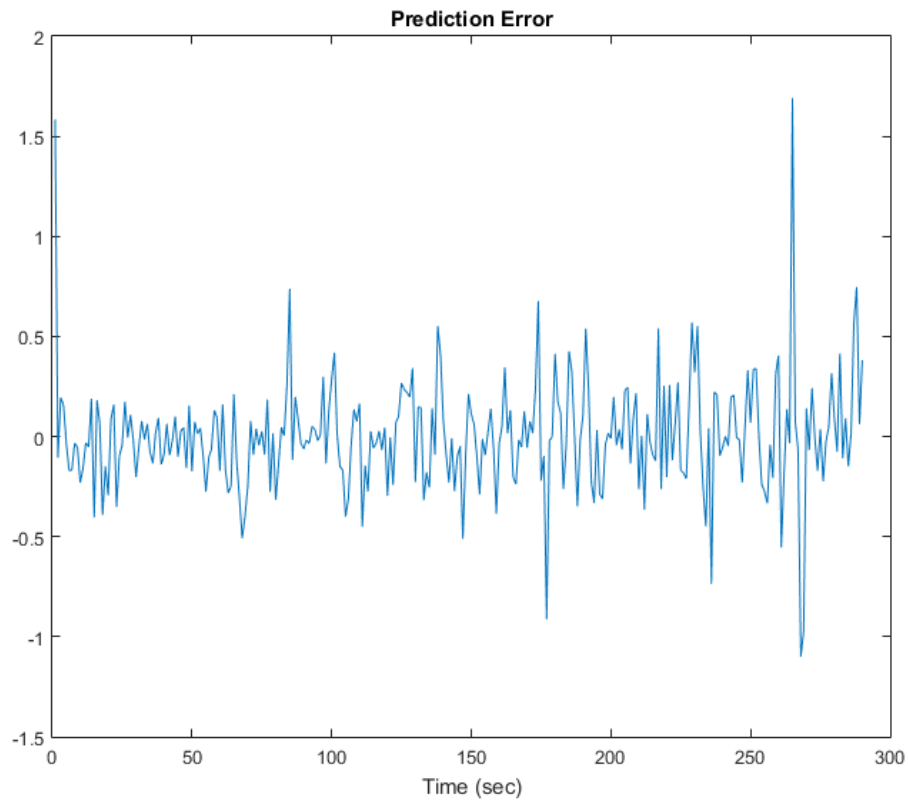
Μετά από ελέγχους και δοκιμές, και με τον περιορισμό ότι μπορούμε μέχρι 3 ασαφή σύνολα ανά είσοδο για τα 3 inputs, έχουμε το παρακάτω σύνολο δοκιμών:

- fuzzySets = 3
- epoch_number = 120
- initial_learning_rate = 0.1
- mfType = gauss2mf
- outType = linear









Παρατηρούμε ότι το training error μας έφτασε στην τιμή του 0.27900. Ένα σφάλμα το οποίο δεν πλησιάσαμε σε καμία άλλη περίπτωση.

Οι λόγοι που κρίθηκε αυτό το σύνολο δοκιμών ως το ιδανικό είναι:

- Το υπολογιστικό σύστημα δεν δυσκολεύτηκε να εκτελέσει τον αλγόριθμο εκπαίδευσης ANFIS
- Το διάγραμμα του σφάλματος έχει τη συμπεριφορά που ζητάμε, με μικρές αυξομειώσεις και εξομάλυνση και σταθερότητα στο τέλος της εκπαίδευσης
- Το διάγραμμα του ρυθμού εκμάθησης έχει και αυτό την συμπεριφορά που θέλουμε.
- Οι τιμές είναι σχετικά μικρές και πιθανόν να έχουμε μειώσει προβλήματα overfitting.

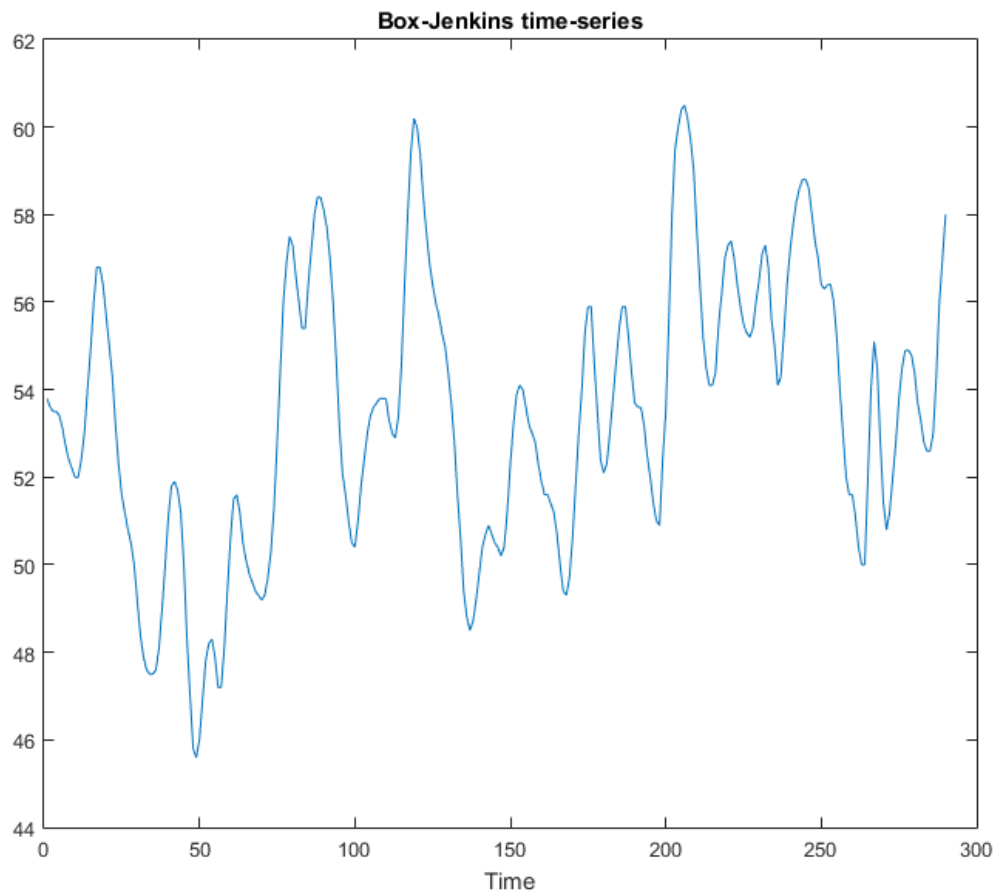
6 Inputs

Σε αυτήν την ενότητα, θα μελετήσουμε την περίπτωση των 3 inputs.

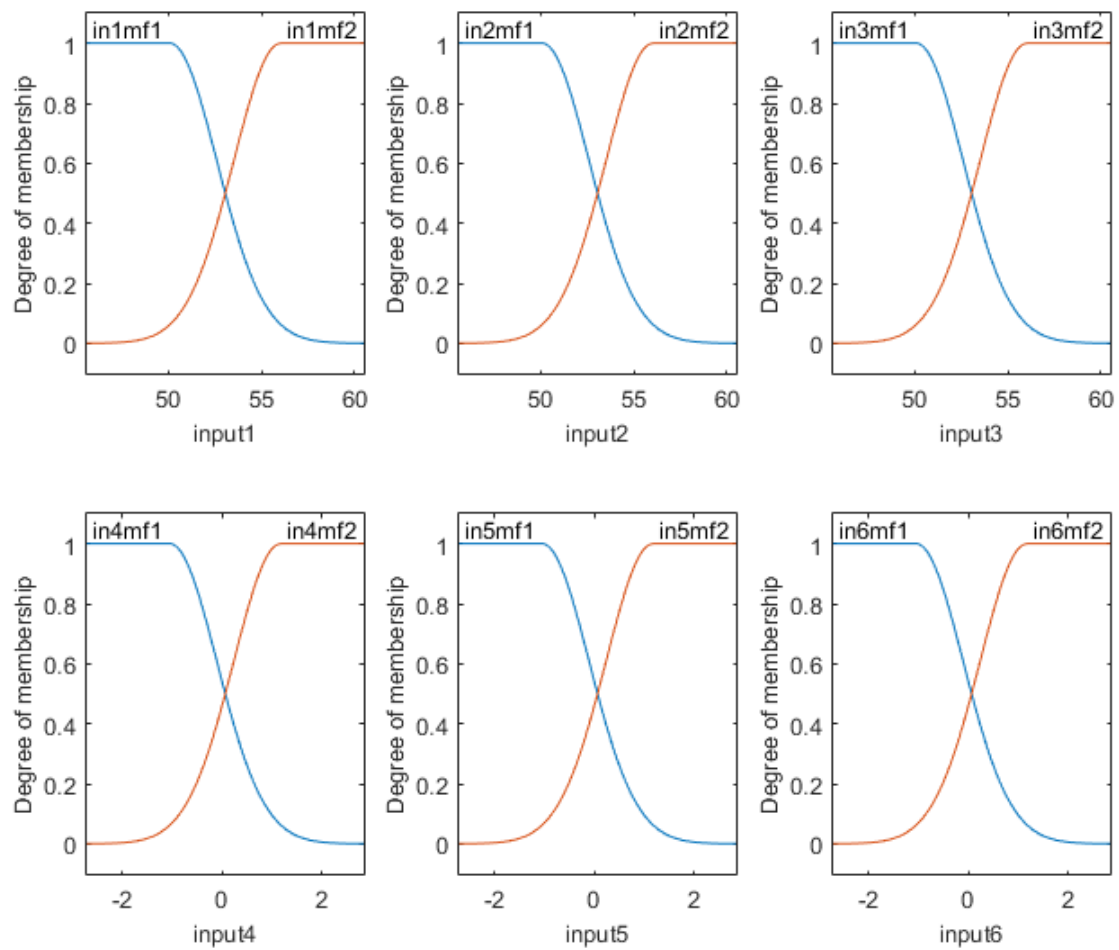
Οι αρχικές μας δοκιμές θα γίνουν με τις παρακάτω παραμέτρους

- fuzzySets = 2
- epoch_number = 30
- initial_learning_rate = 0.1
- mfType = gauss2mf
- outType = linear

Box-Jenkins για τις πραγματικές τιμές μας

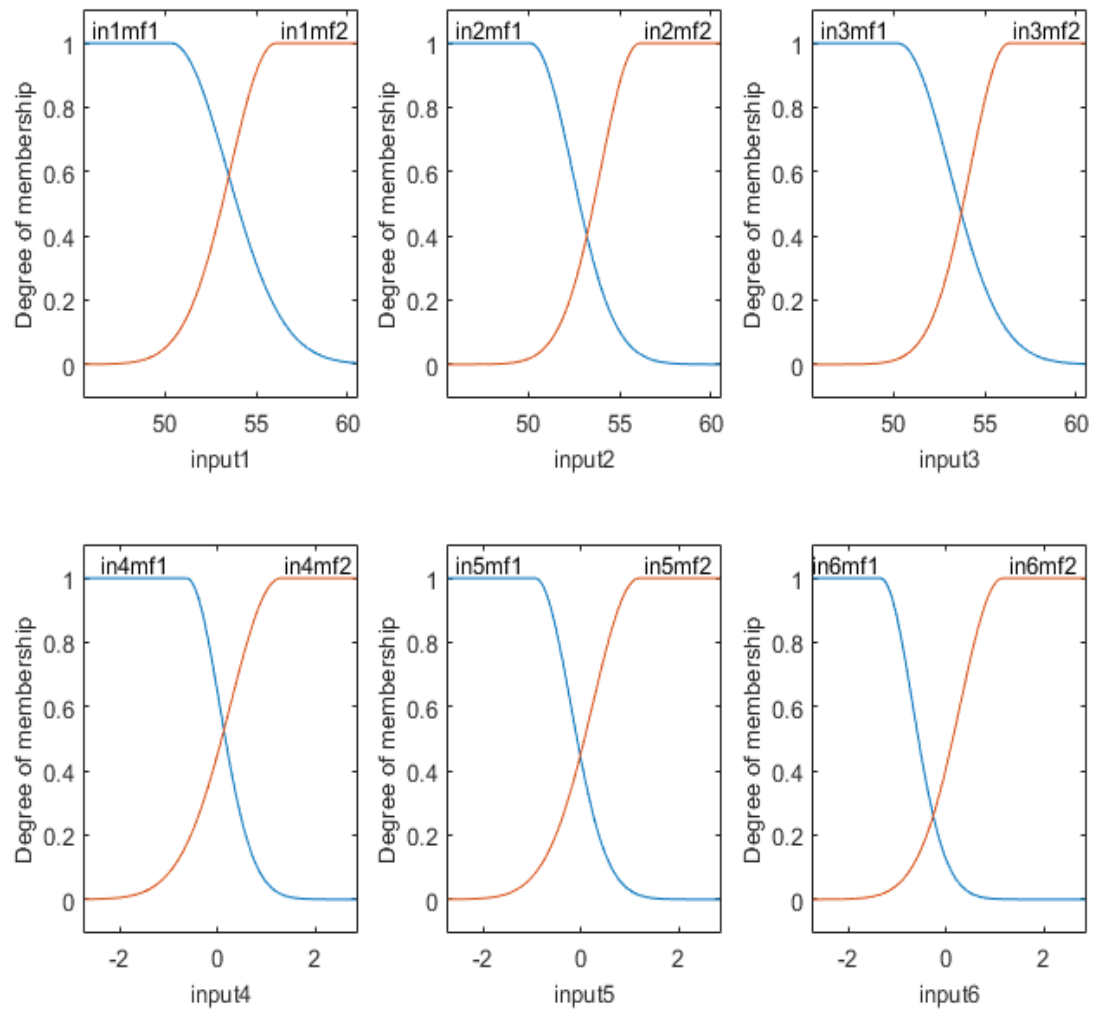


Γράφημα αρχικών συναρτήσεων συμμετοχής

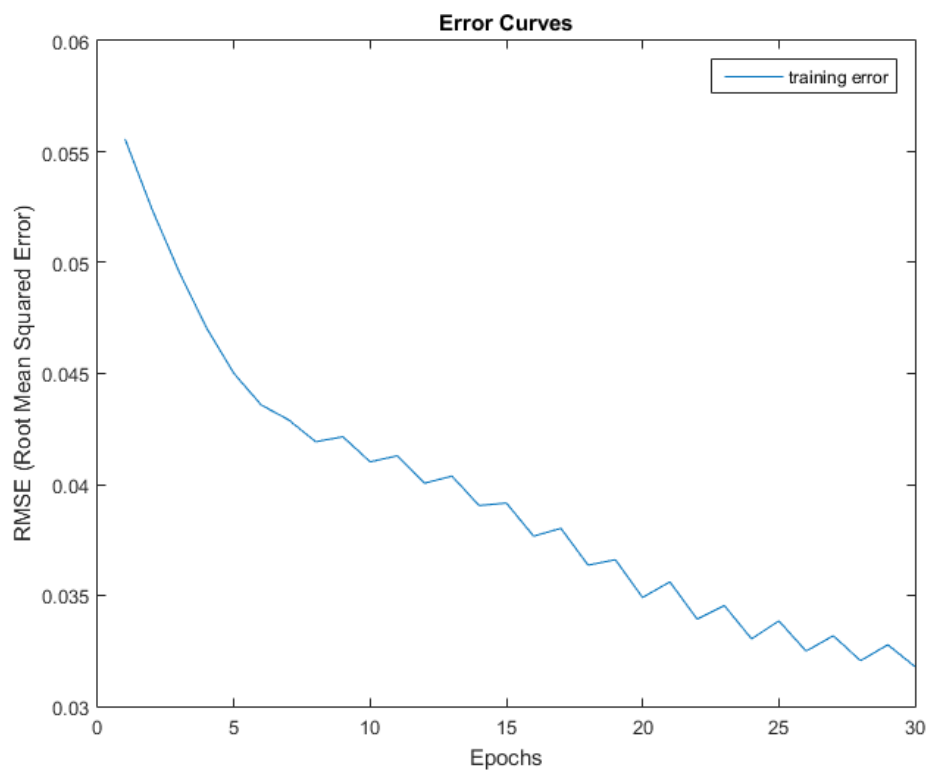


Βλέπουμε τώρα, ότι έχουμε 6 εισόδους για τον αλγόριθμο μας συγκριτικά με τις 2 και 3 εισόδους που εξετάσαμε προηγουμένως.

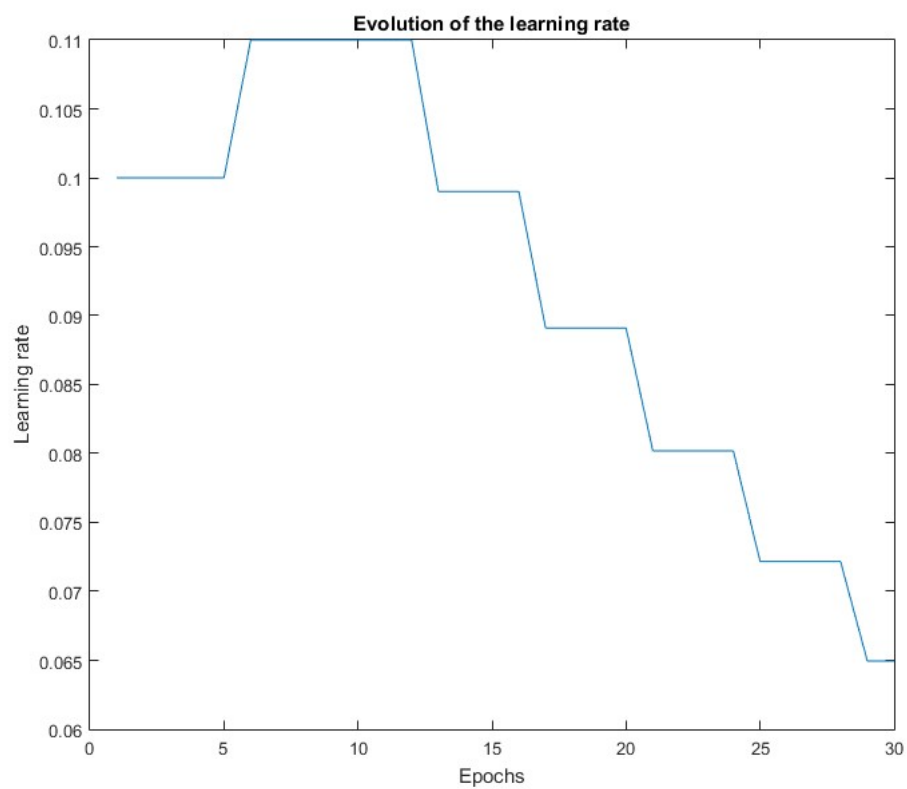
Γράφημα τελικών συναρτήσεων συμμετοχής



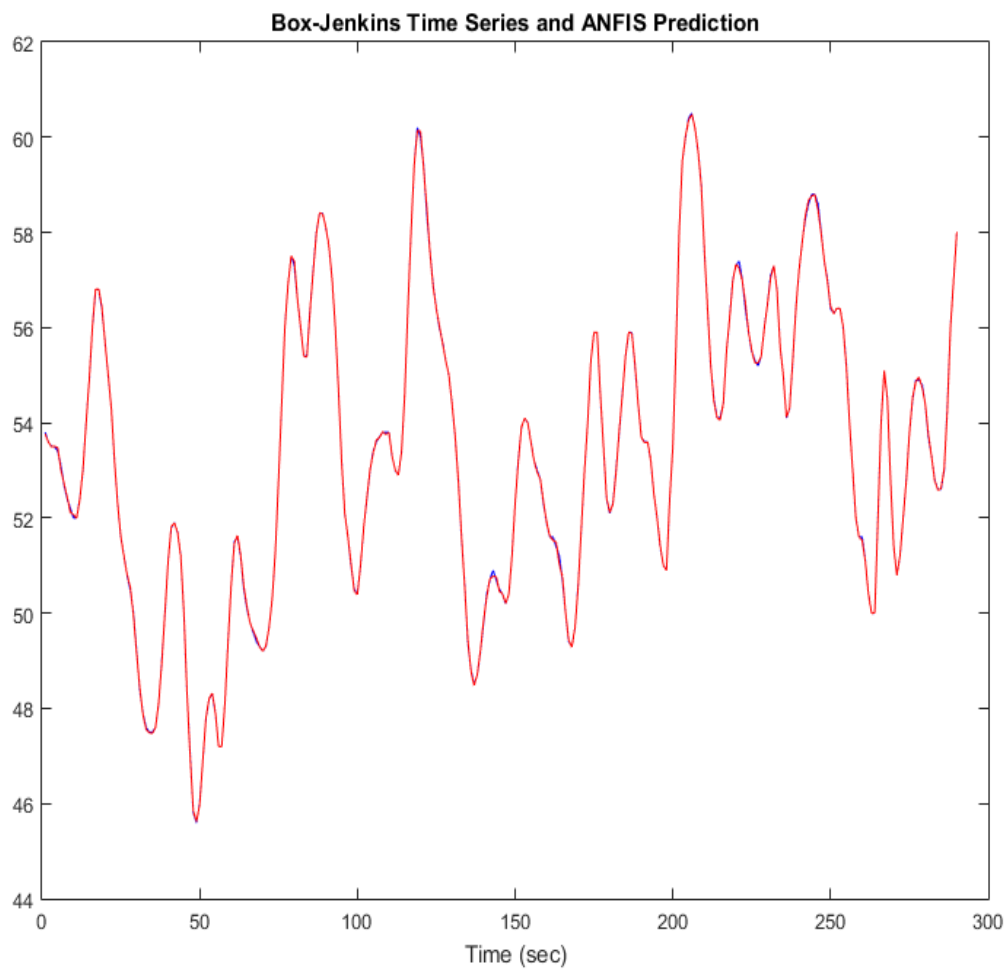
Διάγραμμα Σφάλματος



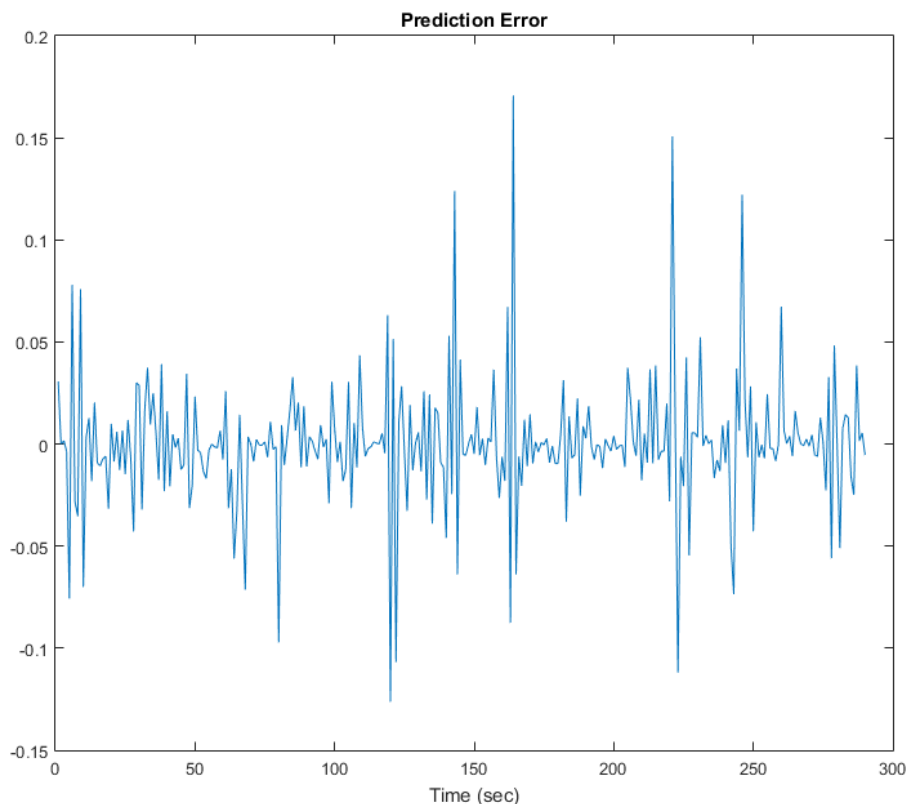
Διάγραμμα ρυθμού εκμάθησης



Διάγραμμα σύγκρισης των δεδομένων με τα πραγματικά



Γράφημα σφάλματος για κάθε δείγμα του συνόλου εκμάθησης



Πλέον είναι ξεκάθαρο, ότι όσες περισσότερες εισόδους έχει ο αλγόριθμος εκπαίδευσης ANFIS, τόσο καλύτερα αποτελέσματα θα παράγει.

Για πρώτη φορά, βλέπουμε ένα training error στην τιμή του 0.031 με μόνο* 2 ασαφή σύνολα ανά είσοδο και 30 επαναλήψεις εκπαίδευσης.

Μετά από εξαντλητικούς ελέγχους, το παρόν σύνολο δοκιμών φαίνεται να είναι κοντά σε αυτό που θέλουμε και με την προϋπόθεση ότι δεν μπορούμε να χρησιμοποιήσουμε παραπάνω ασαφή σύνολα ανά είσοδο, μας περιορίζει στις δοκιμές μας. Βέβαια το training error είναι πολύ μικρό, οπότε μπορούμε να το αποδεχθούμε.

*Μόνο εννοούμε συγκριτικά με τις υπόλοιπες περιπτώσεις (2 και 3 inputs) που χρησιμοποιήσαμε περισσότερα ασαφή σύνολα ανά είσοδο. Στην περίπτωση των 6 inputs, δεν μπορούμε να εκπαιδεύσουμε το νευρωνικό δίκτυο με παραπάνω από 2 ασαφή σύνολα ανά είσοδο, καθώς ο αλγόριθμος εκπαίδευσης απαιτεί πάρα πολλούς πόρους από το υπολογιστικό σύστημα και είναι αδύνατη η εκτέλεση του.