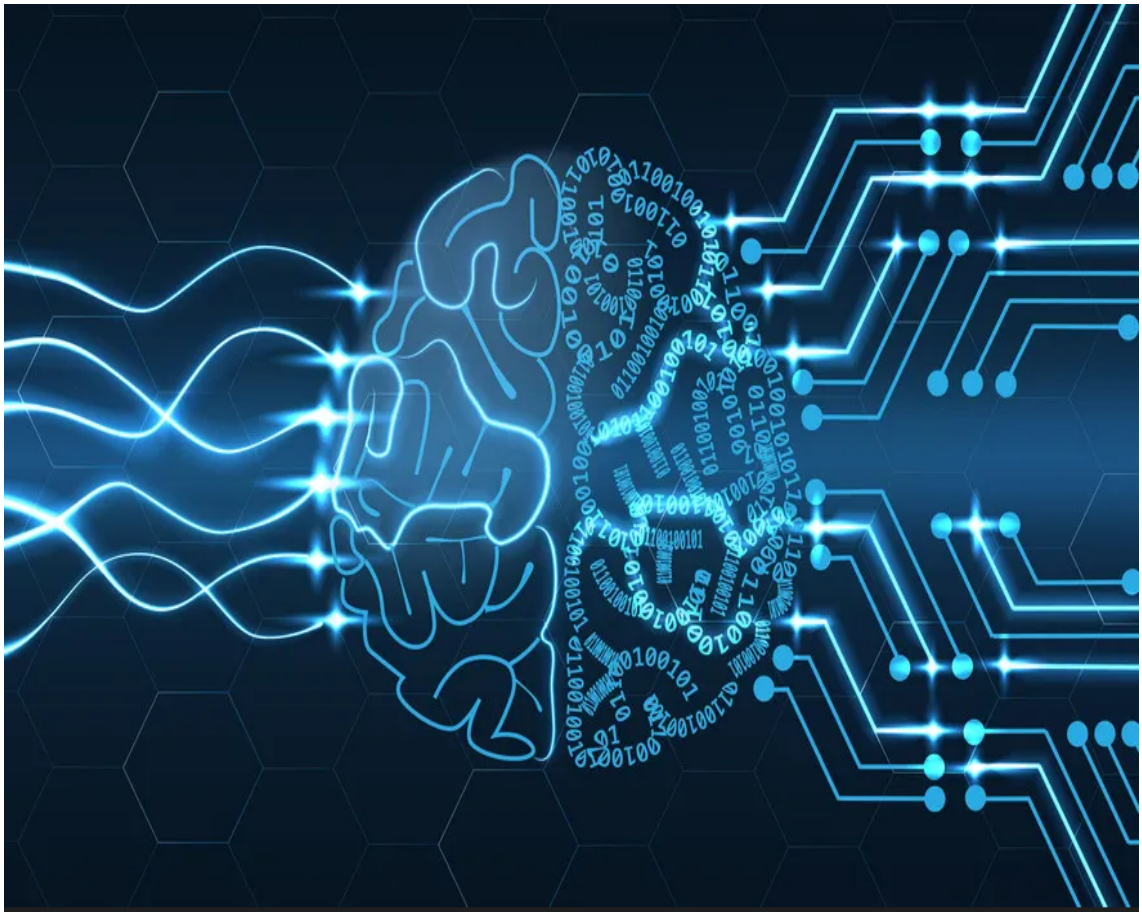
# Υπολογιστική Νοημοσύνη - Τρίτη Εργασία

Ονοματεπώνυμο: Πρωτοψάλτης Παναγιώτης
AEM: 9847
Email: pprotops@ece.auth.gr

March 10, 2024



# Εισαγωγή

Σε αυτήν την εργασία θέλουμε να διερευνήσουμε την ικανότητα των μοντέλων TSK στη μοντελοποίηση πολυμεταβλητών, μη γραμμικών συναρτήσεων. Συγκεκριμένα επιλέγονται δύο σύνολα δεδομένων από το U CI repository με σκοπό την εκτίμηση της μεταβλητής στόχου από τα διαθέσιμα δεδομένα, με χρήση ασαφών νευρωνικών μοντέλων. Το πρώτο σύνολο δεδομένων θα χρησιμοποιηθεί για μια απλή διερεύνηση της διαδικασίας εκπαίδευσης και αξιολόγησης μοντέλων αυτού του είδους, καθώς και για μια επίδειξη τρόπων ανάλυσης και ερμηνείας των αποτελεσμάτων. Το δεύτερο, πολυπλοκότερο σύνολο δεδομένων θα χρησιμοποιηθεί για μια πληρέστερη διαδικασία μοντελοποίησης, η οποία θα περιλαμβάνει μεταξύ άλλων προ-επεξεργαστικά βήματα όπως feature selection, καθώς και μεθόδους βελτιστοποίησης των μοντέλων μέσω του cross validation.

# Εφαρμογή σε απλό dataset

Επιλέγουμε από το UCI repository το Airfoil Self-Noice dataset, που περιέχει 1503 δείγματα και 6 χαρακτηριστικά.

Θα ακολουθήσουμε τα παρακάτω βήματα:

- Διαχωρισμός του dataset σε σύνολα εκπαίδευσης, επικύρωσης και ελέγχου (60-20-20)
- Η εκπαίδευση του μοντέλου έγινε σε 100 κύκλους εποχών.
- Εκπαίδευση TSK μοντέλων, με την υβριδική μέθοδο, με συναρτήσεις συμμετοχής bell-shaped, βαθμό επικάλυψης 0.5 και με διαφορετικές παραμέτρους (πλήθος συναρτήσεων συμμετοχής, μορφή εξόδου)
  - Model-1 - (2, Singleton)
  - Model-2 - (3, Singleton)
  - Model-3 - (2, Polynomial)
  - Model-4 - (3, Polynomial)
- Αξιολόγηση των μοντέλων με τους ακόλουθους δείκτες απόδοσης:
  - MSE (μέσο τετραγωνικό σφάλμα εξόδου μοντέλου- πραγματικής εξόδου)
  - Συντελεστής προσδιορισμού $R^2$
  - Δείκτες NMSE και NDEI

Παρακάτω θα δοθούν τα γραφήματα για κάθε μία από τις παραπάνω περιπτώσεις. Για κάθε περίπτωση έχουμε 4 γραφήματα.

- 1 γράφημα για το "Learning-Curve"
- 1 γράφημα για τις συναρτήσεις συμμετοχής πριν την εκπαίδευση
- 1 γράφημα για τις συναρτήσεις συμμετοχής μετά την εκπαίδευση
- 1 γράφημα για το προβλεπόμενο σφάλμα

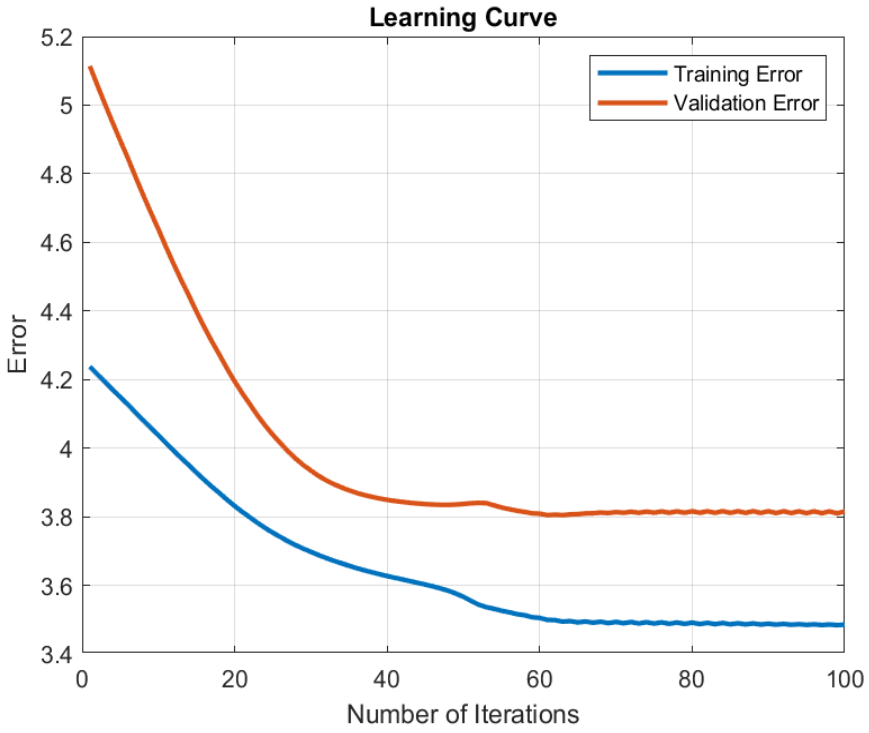



Figure 1: Καμπύλη εκμάθησής του 1ου μοντέλου

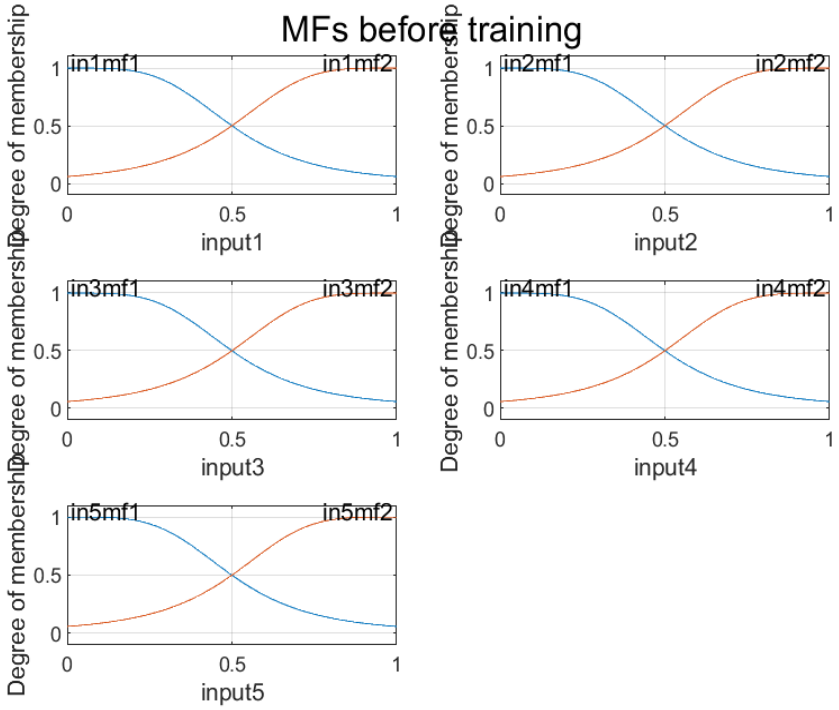



Figure 2: Συναρτήσεις συμμετοχής πριν την εκπαίδευση του 1ου μοντέλου

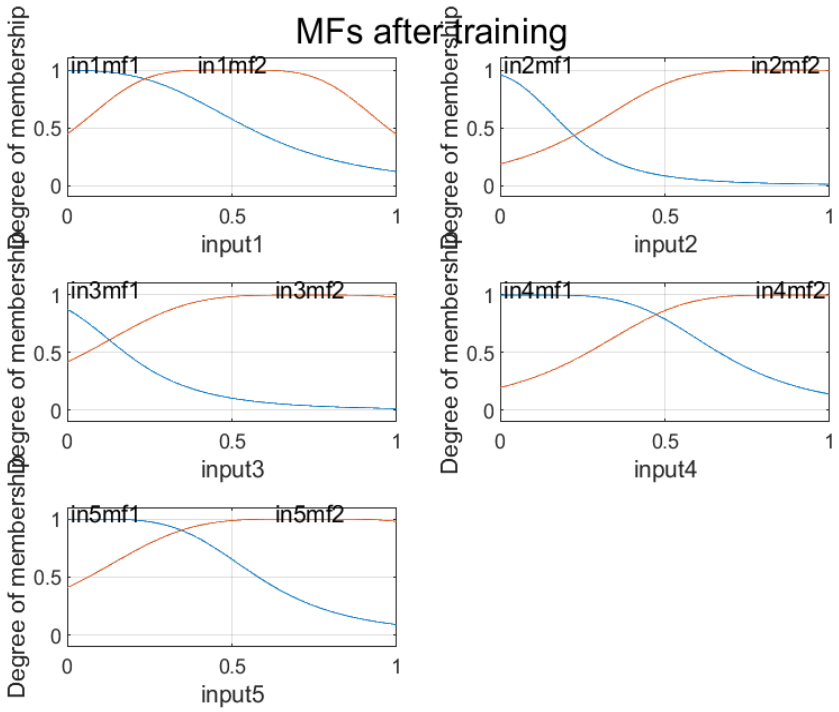



Figure 3: Συναρτήσεις συμμετοχής μετά την εκπαίδευση του 1ου μοντέλου

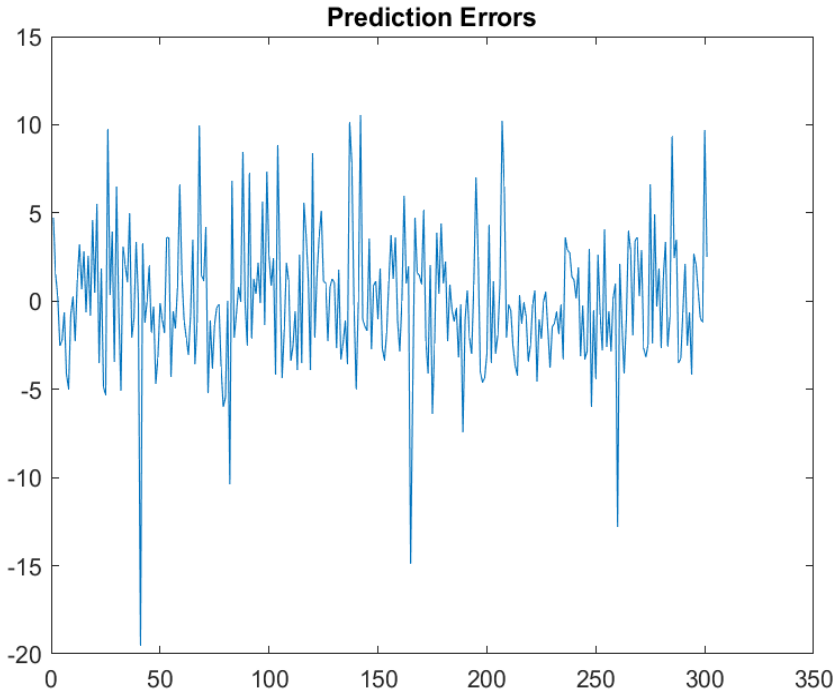



Figure 4: Σφάλμα πρόβλεψης 1ου μοντέλου

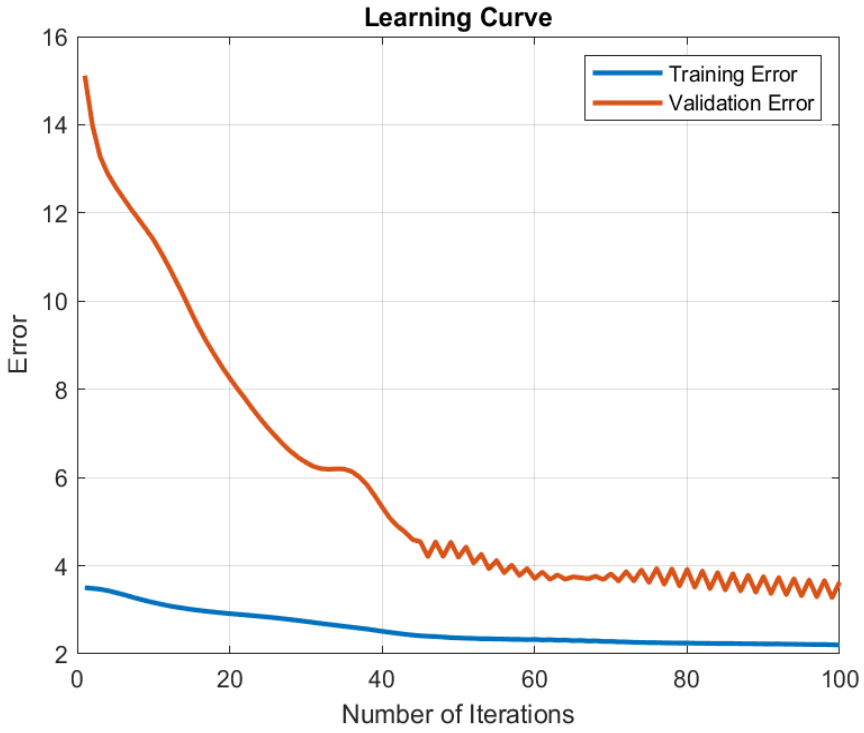



Figure 5: Καμπύλη εκμάθησής του 2ου μοντέλου

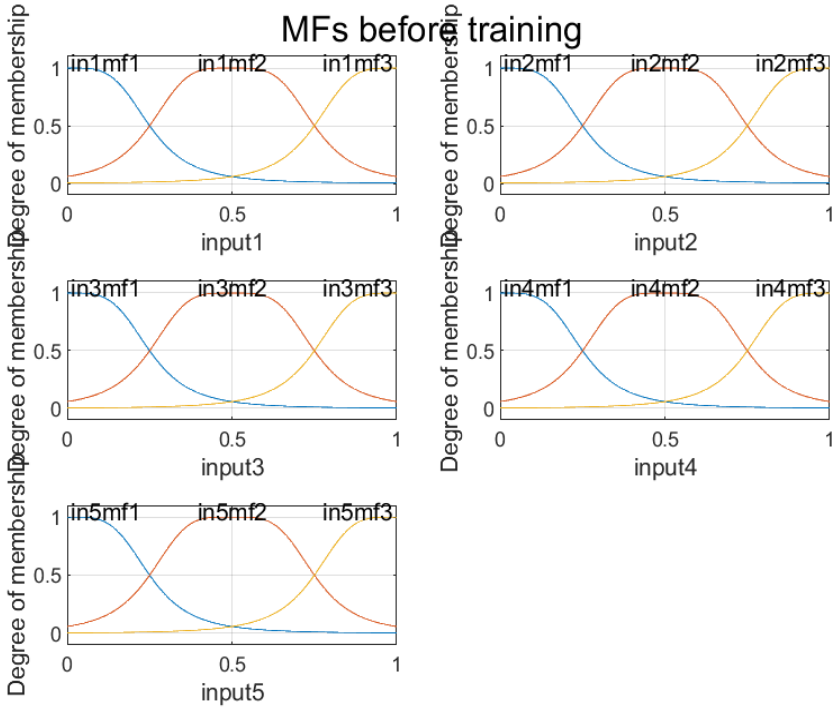



Figure 6: Συναρτήσεις συμμετοχής πριν την εκπαίδευση του 2ου μοντέλου

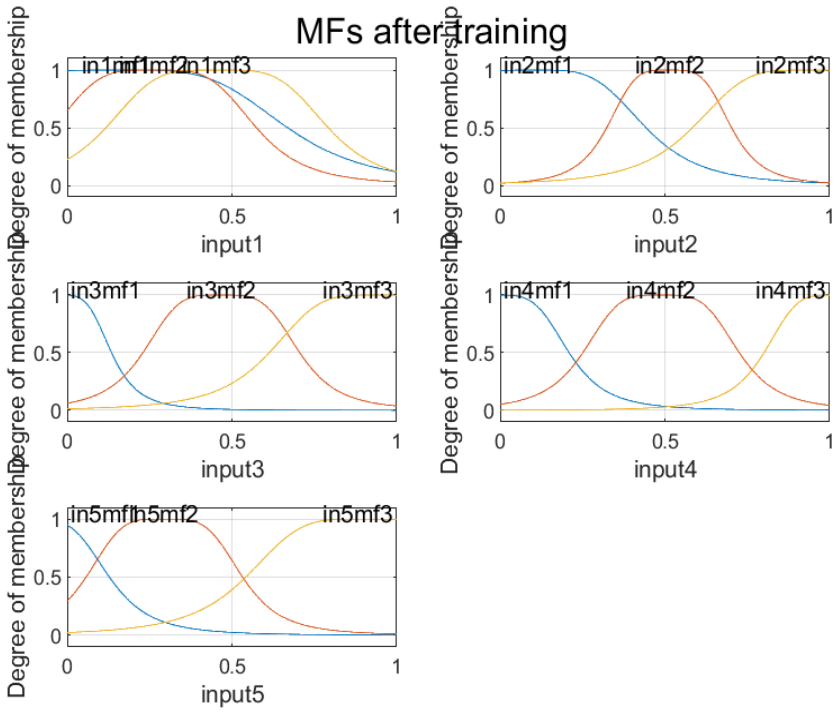



Figure 7: Συναρτήσεις συμμετοχής μετά την εκπαίδευση του 2ου μοντέλου

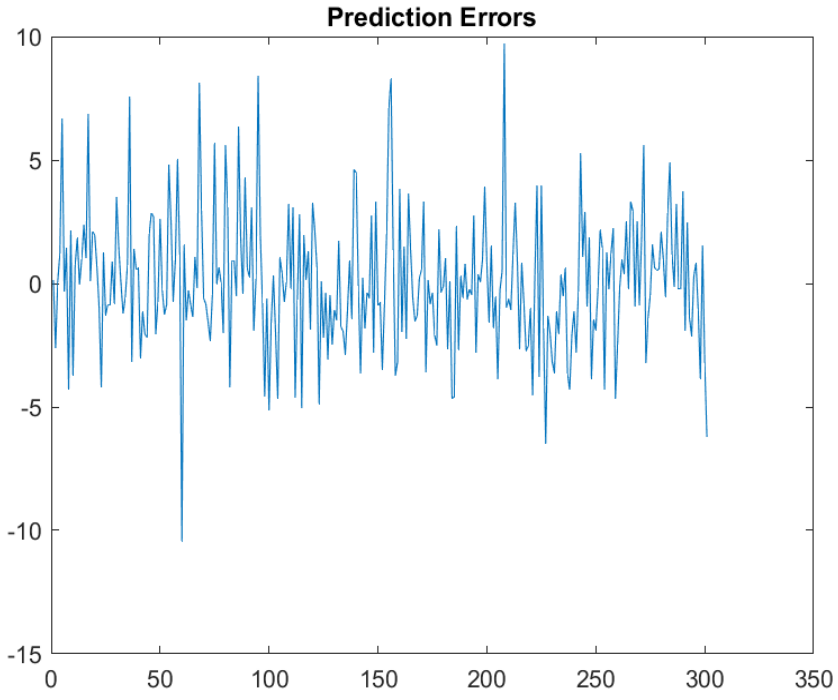



Figure 8: Σφάλμα πρόβλεψης 2ου μοντέλου

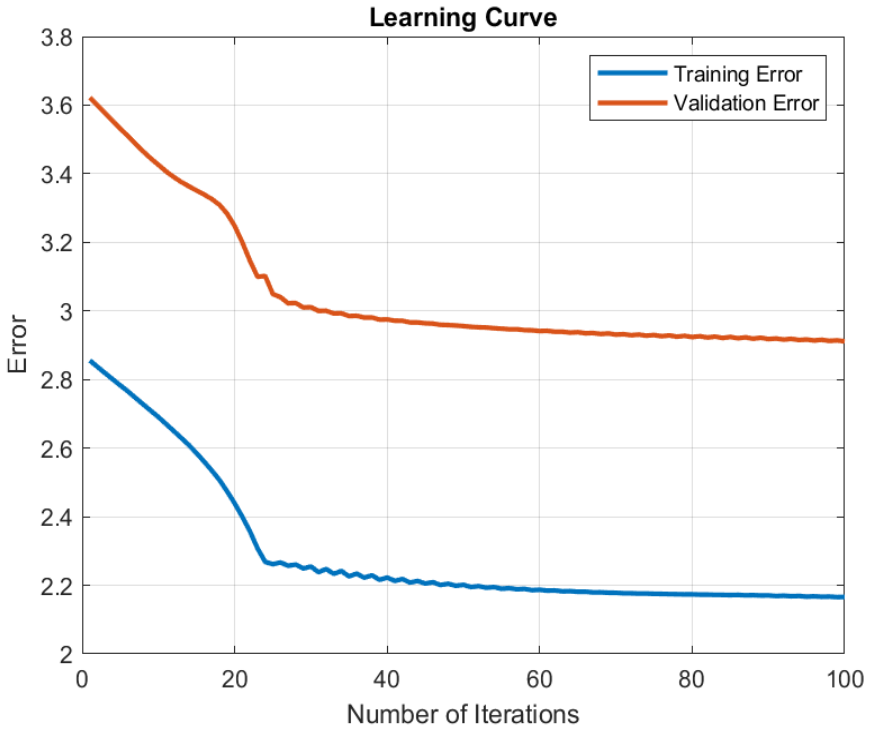



Figure 9: Καμπύλη εκμάθησής του 3ου μοντέλου

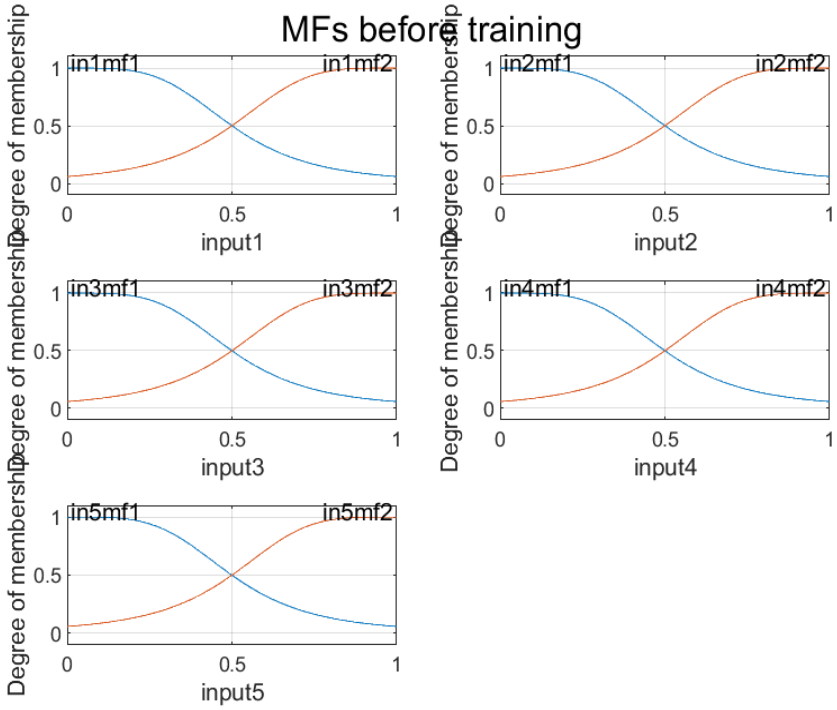



Figure 10: Συναρτήσεις συμμετοχής πριν την εκπαίδευση του 3ου μοντέλου

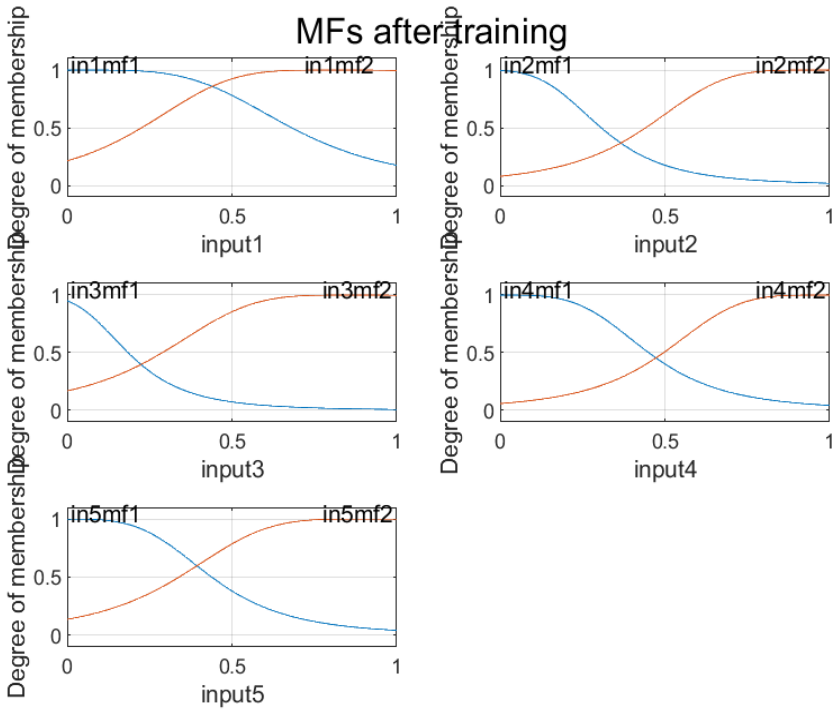



Figure 11: Συναρτήσεις συμμετοχής μετά την εκπαίδευση του 3ου μοντέλου

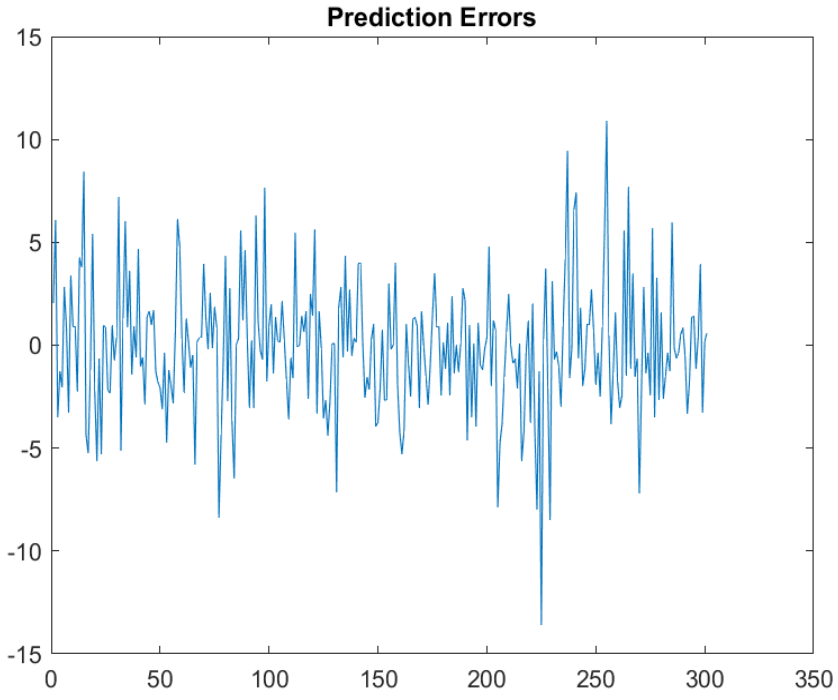



Figure 12: Σφάλμα πρόβλεψης 3ου μοντέλου

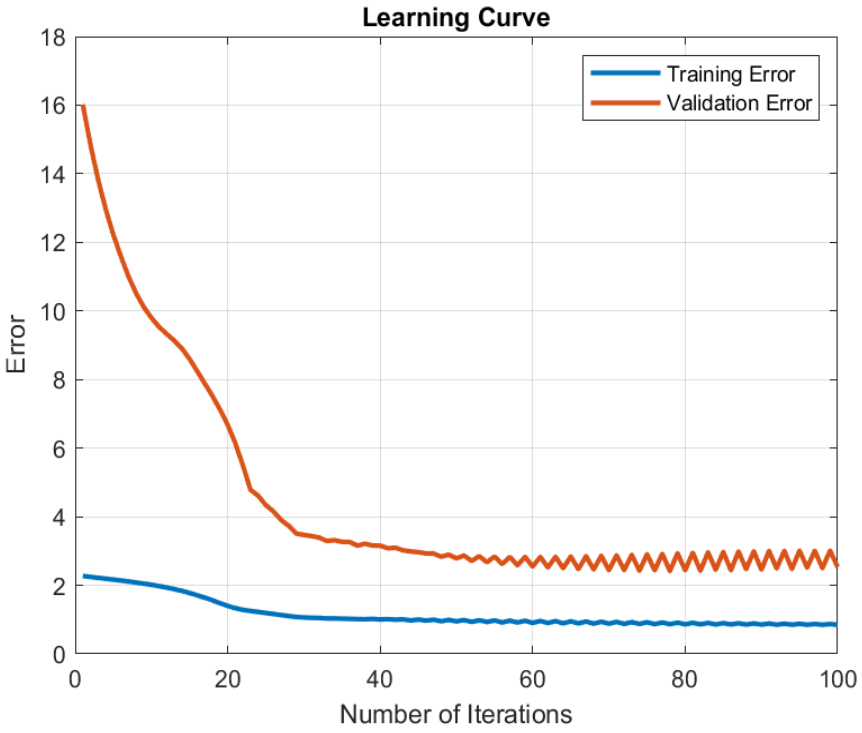



Figure 13: Καμπύλη εκμάθησής του 4ου μοντέλου

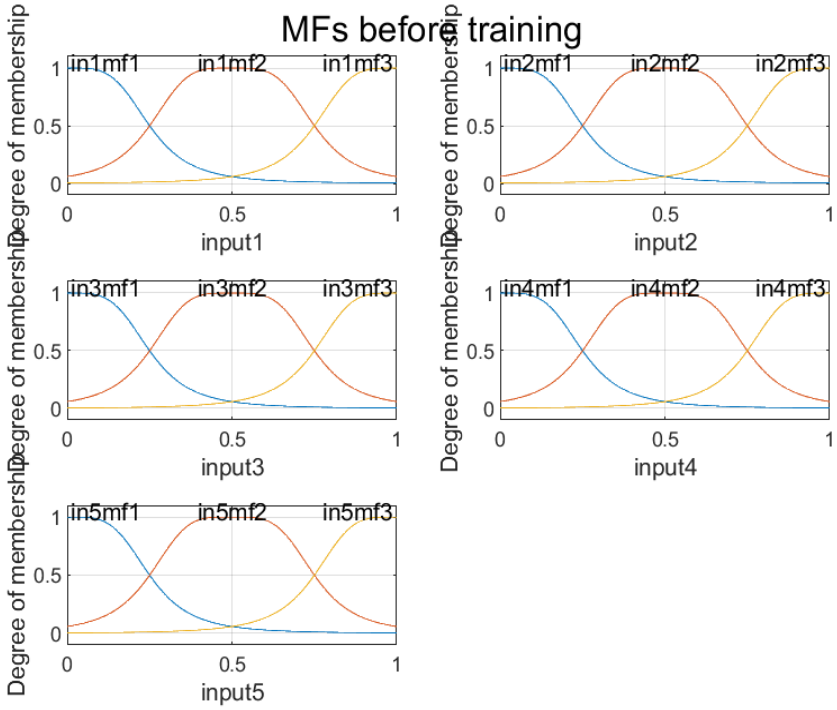



Figure 14: Συναρτήσεις συμμετοχής πριν την εκπαίδευση του 4ου μοντέλου

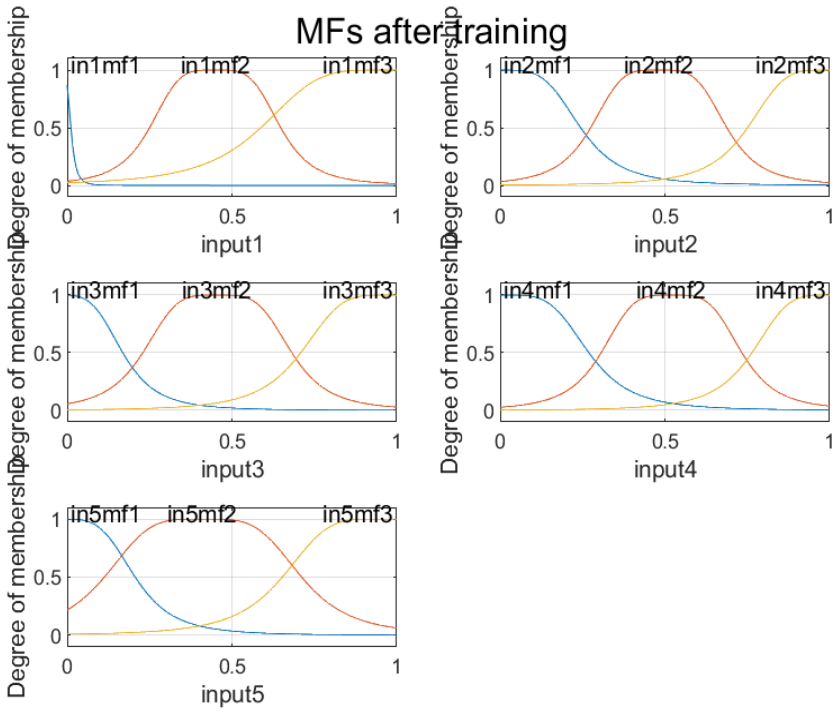



Figure 15: Συναρτήσεις συμμετοχής μετά την εκπαίδευση του 4ου μοντέλου

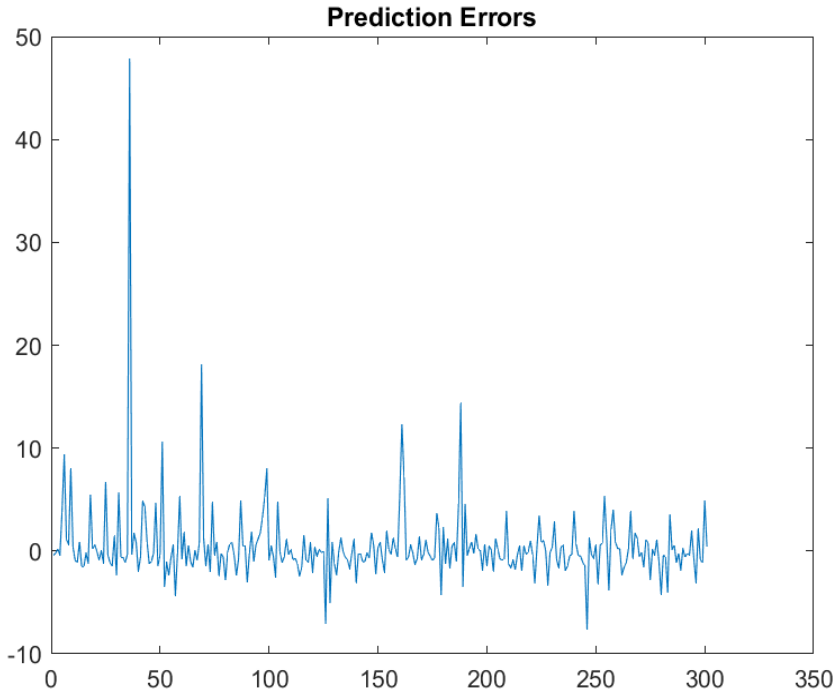



Figure 16: Σφάλμα πρόβλεψης 4ου μοντέλου

## Γενικές Παρατηρήσεις

Όπως παρατηρούμε τα validation error είναι πιο μεγάλα από τα training που είναι λογικό καθώς τα training εκπαιδεύονται σε περισσότερα δεδομένα. Όσο προχωράνε οι επαναλήψεις, το error και των δύο γραφημάτων μειώνονται και προσεγγίζουν καλύτερες τιμές. Παρατηρούμε ότι το 4ο μοντέλο πετυχαίνει το μικρότερο σφάλμα, μετά ακολουθεί το 2ο, μετά το 3ο και τέλος το 1ο. Αυτό είναι λογικό αν παρατηρήσουμε τις διαφορές των μοντέλων, ως προς τις συναρτήσεις συμμετοχής και την μορφή της εξόδου. **Παρατηρούμε δηλαδή ότι για πολυωνυμική έξοδο έχουμε καλύτερη προσέγγισή σε σχέση με την singleton, το ίδιο ακριβώς και με την αύξηση του πλήθους των συναρτήσεων συμμετοχής, καθώς και με την αύξηση των εποχών. Η αύξηση αυτή ωστόσο έχει επιρροή στον χρόνο, το 1ο μοντέλο κάνει 5 δευτερόλεπτα, το 2ο κάνει 2 λεπτά, το 3ο κάνει 30 δευτερόλεπτα και τέλος το 4ο κάνει 50 λεπτά.**

## Ειδικές Παρατηρήσεις

### Μοντέλο 1

Στο figure 1 βλέπουμε ότι το σφάλμα ξεκινάει από το 5.1 για το validation και από το 4.2 για το training και καταλήγει στο 3.8 και 3.5 αντίστοιχα, το οποίο είναι μια ικανοποιητική προσέγγιση για τον ελάχιστο χρόνο που έκανε για να τρέξει το μοντέλο. Στις καμπύλες βλέπουμε ότι δεν υπάρχουν ταλαντώσεις και επίσης βλέπουμε ότι έχουμε καλή εκπαίδευση του μοντέλου. Στο figure 2 και 3 βλέπουμε τις συναρτήσεις συμμετοχής και στο figure 4 βλέπουμε το σφάλμα, το οποίο έχει κάποια "spikes" αλλά κατά τα άλλα είναι ικανοποιητικό για τον χρόνο που κάνει να τρέξει, το σφάλμα είναι κυρίως στο διάστημα [-5, 10]. Οι μετρικές του μοντέλου δίνονται παρακάτω.

| R2 | RMSE | NDEI | NMSE |
|---|---|---|---|
| 0.703208 | 3.777705 | 0.543880 | 0.295806 |

### Μοντέλο 2

Στο figure 5 βλέπουμε ότι το σφάλμα ξεκινάει από το 15 για το validation και από το 3.5 για το training και καταλήγει στο 3.8 και 2.2 αντίστοιχα, το οποίο είναι επίσης πολύ καλή προσέγγιση για τον χρόνο που έκανε για να τρέξει το μοντέλο. Στις καμπύλες βλέπουμε ότι δεν υπάρχουν μεγάλες ταλαντώσεις και επίσης βλέπουμε ότι έχουμε καλή εκπαίδευση του μοντέλου. Στο figure 6 και 7 βλέπουμε τις συναρτήσεις συμμετοχής και στο figure 8 βλέπουμε το σφάλμα, το οποίο έχει 3 αρκετά μεγάλα "spikes", το οποίο δεν είναι ανησυχητικό, καθώς το μοντέλο απλά σε εκείνες τις τιμές δεν προέβλεψε καλά τη σωστή τιμή, αλλά κατά τα άλλα είναι ικανοποιητικό για τον χρόνο που κάνει να τρέξει, το σφάλμα είναι κυρίως στο διάστημα [-4, 4]. Οι μετρικές του μοντέλου δίνονται παρακάτω.

| R2 | RMSE | NDEI | NMSE |
|---|---|---|---|
| 0.821727 | 2.711864 | 0.421521 | 0.177680 |

### Μοντέλο 3

Στο figure 9 βλέπουμε ότι το σφάλμα ξεκινάει από το 3.6 για το validation και από το 2.8 για το training και καταλήγει στο 3 και 2.2 αντίστοιχα, το οποίο είναι η καλύτερη προσέγγιση προς το παρόν. Στις καμπύλες βλέπουμε ότι δεν υπάρχουν ταλαντώσεις και επίσης βλέπουμε ότι έχουμε καλή εκπαίδευση του μοντέλου, καθώς επίσης και η ομαλή κάθοδος του σφάλματος είναι κι αυτό κάτι που πρέπει να σημειωθεί. Στο figure 10 και 11 βλέπουμε τις συναρτήσεις συμμετοχής και στο figure 12 βλέπουμε το σφάλμα, το οποίο έχει κάποια "spikes" αλλά κατά τα άλλα είναι ικανοποιητικό, το σφάλμα είναι κυρίως στο διάστημα [-3, 3].Οι μετρικές του μοντέλου δίνονται παρακάτω.

| R2 | RMSE | NDEI | NMSE |
|---|---|---|---|
| 0.781977 | 3.161286 | 0.466153 | 0.217299 |

### Μοντέλο 4

Στο figure 13 βλέπουμε ότι το σφάλμα ξεκινάει από το 16 για το validation και από το 2.5 για το training και καταλήγει στο 3 και 1 αντίστοιχα, το οποίο είναι η καλύτερη προσέγγιση. Στις καμπύλες βλέπουμε ότι δεν υπάρχουν ταλαντώσεις και επίσης βλέπουμε ότι έχουμε καλή εκπαίδευση του μοντέλου, καθώς επίσης και η απότομη κάθοδος του σφάλματος για το validation ενώ ξεκίνησε από πολύ μεγάλη τιμή σε σχέση με το training. Στο figure 14 και 15 βλέπουμε τις συναρτήσεις συμμετοχής και στο figure 16 βλέπουμε το σφάλμα, το οποίο έχει κάποια "spikes" αλλά κατά τα άλλα είναι ικανοποιητικό, το σφάλμα είναι κυρίως στο διάστημα [-1, 5]. Οι μετρικές του μοντέλου δίνονται παρακάτω.

| R2 | RMSE | NDEI | NMSE |
|---|---|---|---|
| 0.691035 | 3.884325 | 0.554922 | 0.307938 |

# Εφαρμογή σε dataset με υψηλή διαστασιμότητα

Στη δεύτερη φάση της εργασίας θα ασχοληθούμε με το πρόβλημα της "έκρηξης" του πλήθους των IF-THEN κανόνων, το οποίο συμβαίνει γιατί με την αύξηση των εισόδων υπάρχει εκθετική αύξηση του αριθμού αυτών των κανόνων, γεγονός που καθιστά δύσκολη την μοντελοποίηση ακόμη για μεσαία dataset.

Το dataset που επιλέχθηκε είναι το Superconductivity dataset από το UCI Repository, το οποίο περιλαμβάνει 21253 δείγματα, καθένα από τα οποία περιγράφεται από 81 μεταβλητές/χαρακτηριστικά. Είναι εύκολο να δούμε ότι σε αυτό το dataset θα χρειαζόμασταν $2^81$ κανόνες, το οποίο είναι απαγορευτικό, για αυτό τον λόγο θα χρησιμοποιηθεί η λογική της **επιλογής χαρακτηριστικών** και της **διαμέρισης διασκορπισμού**. Λόγω αυτής της επιλογής καλούμαστε να επιλέξουμε τις παραμέτρους που εισάγει το πρόβλημα, δηλαδή τον αριθμό των χαρακτηριστικών και τον αριθμό των ομάδων που θα δημιουργηθούν.

Σε αυτή την εργασία χρησιμοποιώντας το "grid-search" θα βρούμε τις βέλτιστες τιμές για τις δύο παραπάνω παραμέτρους. Επιλέχθηκαν οι τιμές 0.2, 0.4, 0.7, 1 για την ακτίνα και οι τιμές 5, 10, 15, 20 για τον αριθμό των χαρακτηριστικών. Παρακάτω θα δοθούν κάποια διαγράμματα για τις τιμές του σφάλματος για κάθε συνδυασμό αυτών των επιλογών (δηλαδή 0.2-5, 0.2-10, 0.2-15, 0.2-20, 0.4-5, 0.4-10 κτλ). Αρχικά για να βρούμε τα επικρατέστερα 5,10,15 ή 20 χαρακτηριστικά του dataset, χρησιμοποιήθηκε η συνάρτηση relieff του matlab, που μας επιστρέφει την σειρά από τα επικρατέστερα αυτά χαρακτηριστικά. Έπειτα για κάθε ένα μοντέλο έγινε το "cross-validation", με το οποίο καταφέραμε να βρούμε το σφάλμα και τον αριθμό των κανόνων, τα οποία μας βοήθησαν στην καλύτερη επιλογή συνδυασμού που θα δούμε παρακάτω.

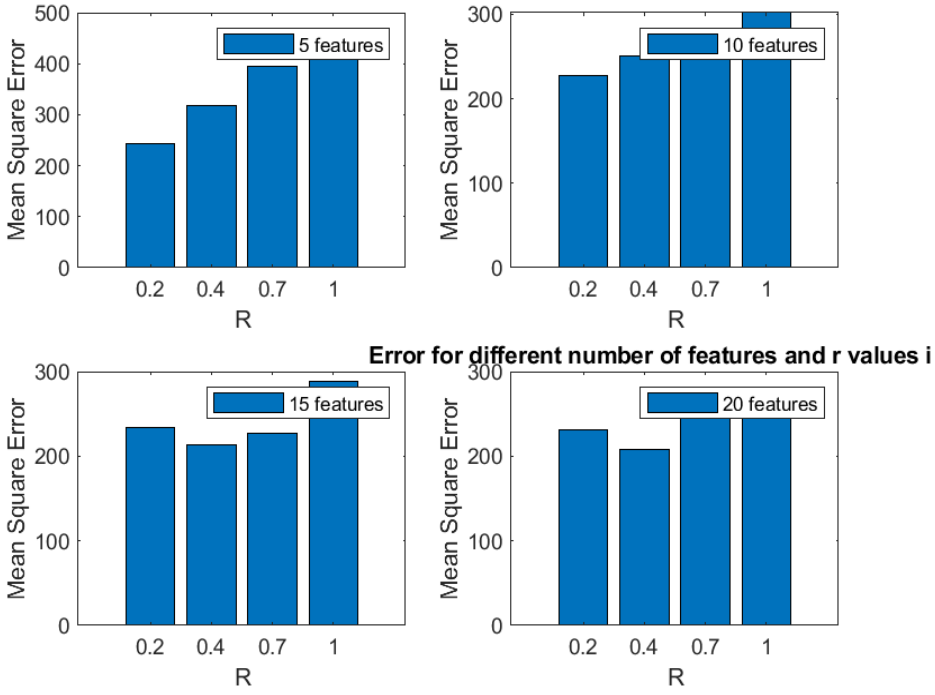



Figure 17: Σφάλμα πρόβλεψης κάθε συνδυασμού σε 2D

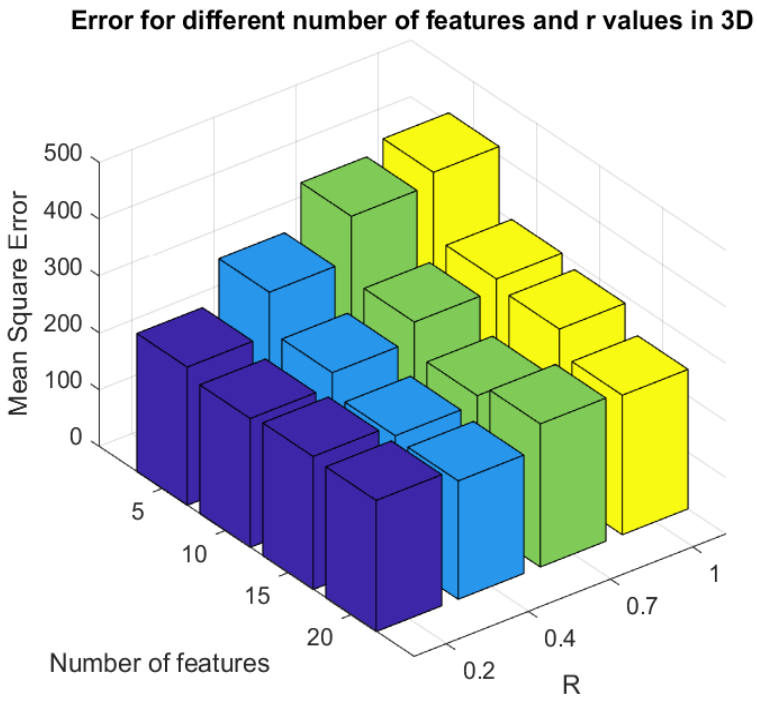



Figure 18: Σφάλμα πρόβλεψης κάθε συνδυασμού σε 3D

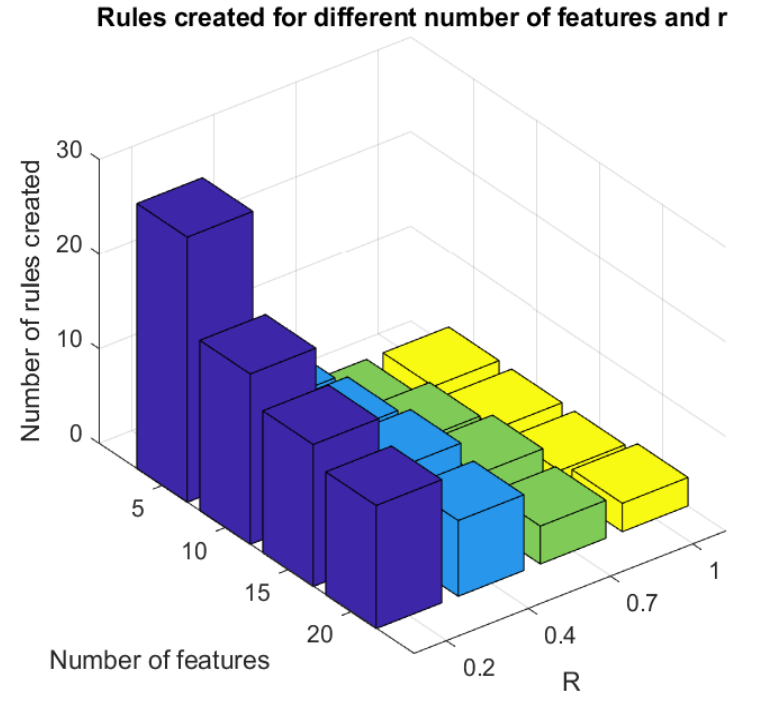



Figure 19: Ο αριθμός των κανόνων σε 3D

## Παρατηρήσεις

Ο κώδικας για το παραπάνω κομμάτι έτρεξε σε **50 ώρες**, λόγω της μεγάλης επιλογής των χαρακτηριστικών. Στα παραπάνω διαγράμματα παρατηρούμε ότι ο συνδυασμός τιμών που έχει το μικρότερο σφάλμα

είναι αυτός του **0.4 ακτίνα και 20 αριθμός χαρακτηριστικών**. Ο συνδυασμός αυτός ήταν κοντά στους συνδυασμούς 0.4-15 και 0.7-15. Παρόλα αυτά κρίνουμε πως θα ήταν καλύτερο να επιλεγεί αυτός παρόλο που όπως βλέπουμε στο figure 19, ο αριθμός κανόνων του 0.4-20 είναι μεγαλύτερος του 0.7-15 και συνεπώς θα έτρεχε πιο γρήγορα, ωστόσο εμείς θέλουμε να φτιάξουμε ένα μοντέλο με το μικρότερο δυνατό σφάλμα και έτσι επιλέχθηκε ο συνδυασμός 0.4-20.

## Τελικό TSK μοντέλο με βέλτιστες τιμές παραμέτρων

Με τον συνδυασμό που επιλέχθηκε, δηλαδή 0.4-20, θα κάνουμε ένα μοντέλο παρόμοιας λογικής με το πρώτο μέρος της εργασίας. Θα χωρίσουμε το dataset σε 60-20-20 για training, validation και testing αντίστοιχα και θα το εκπαιδεύσουμε για 150 κύκλους επαναλήψεων. Παρακάτω δίνονται τα διαγράμματα για την εκπαίδευση αυτού του μοντέλου.

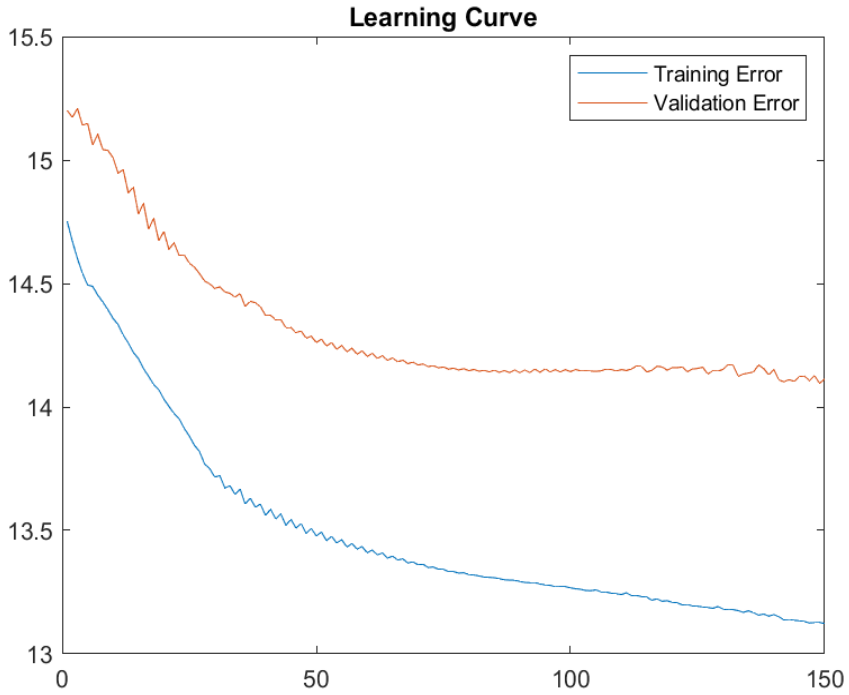



Figure 20: Καμπύλη εκμάθησής του τελικού μοντέλου

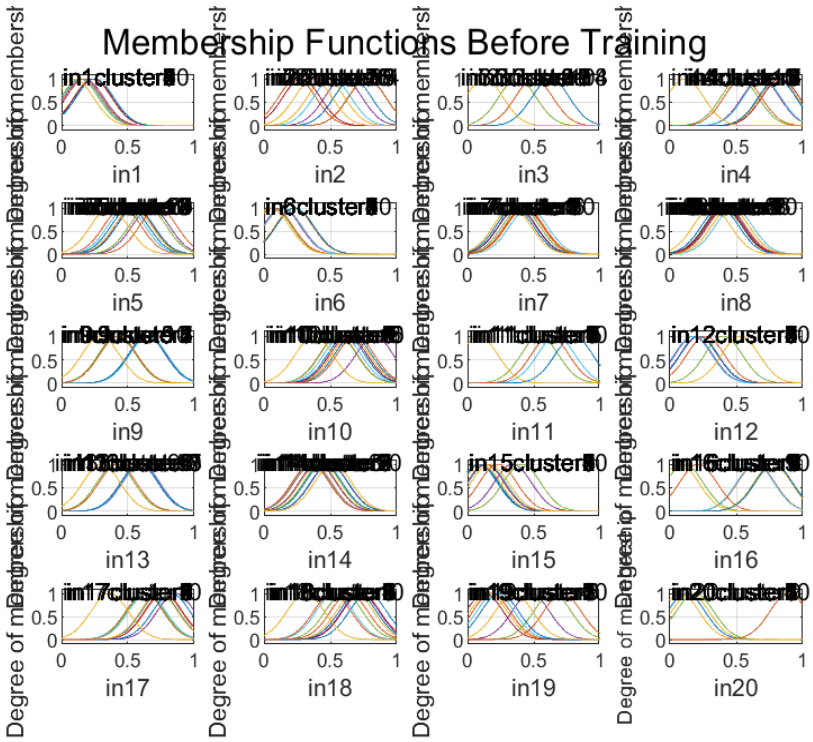



Figure 21: Συναρτήσεις συμμετοχής πριν την εκπαίδευση του τελικού μοντέλου

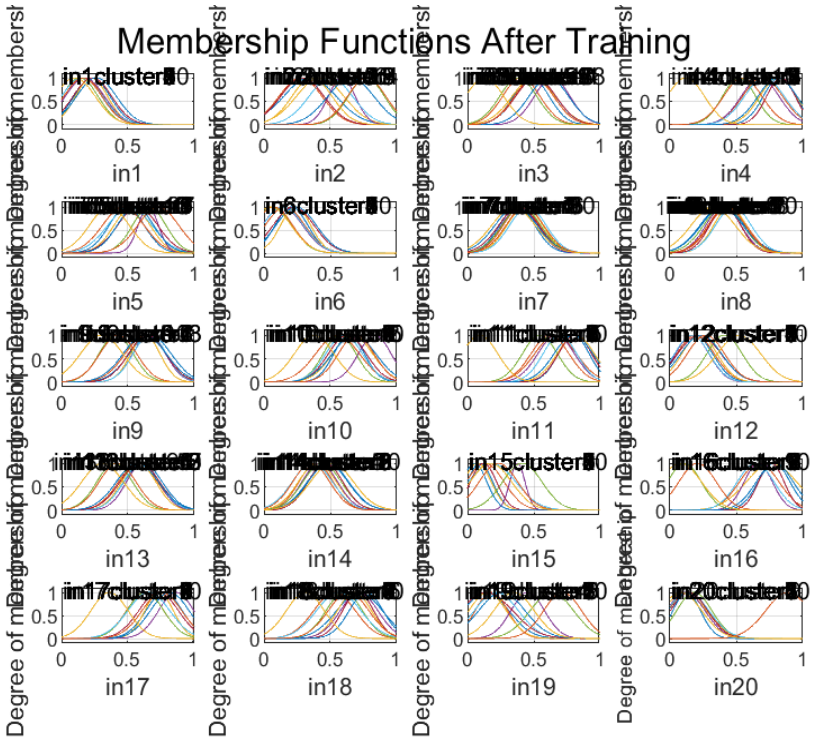



Figure 22: Συναρτήσεις συμμετοχής μετά την εκπαίδευση του τελικού μοντέλου

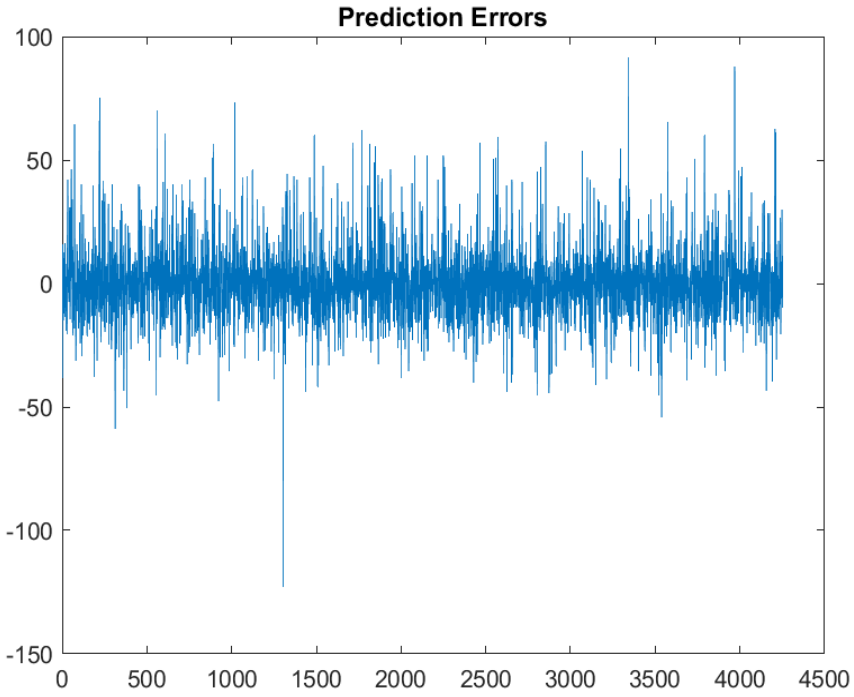



Figure 23: Σφάλμα πρόβλεψης τελικού μοντέλου

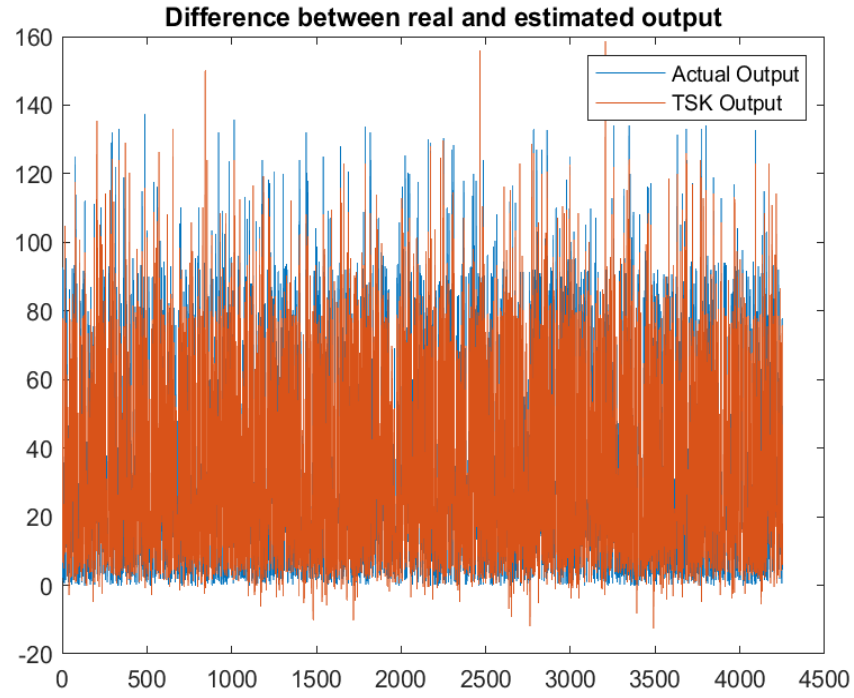



Figure 24: Διαφορά πραγματικής και εκτιμώμενης εξόδου

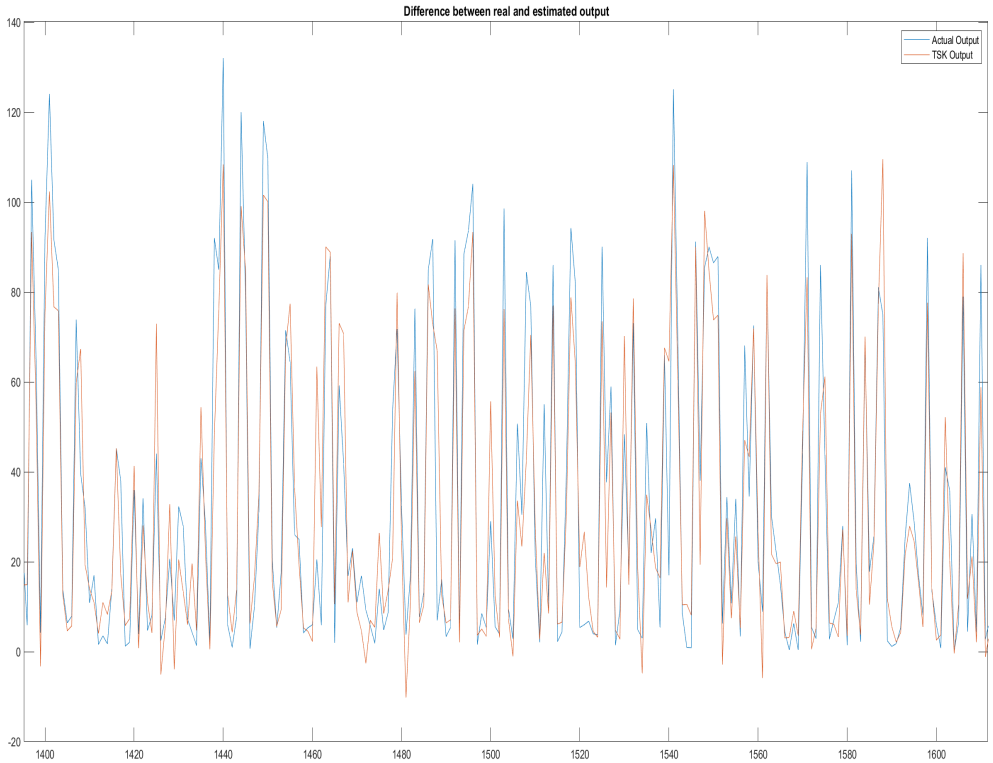



Figure 25: Διαφορά πραγματικής και εκτιμώμενης εξόδου εστιασμένη

## Παρατηρήσεις

Ο πίνακας για τις τιμές των μετρικών του μοντέλου είναι ο παρακάτω.

| R2 | RMSE | NDEI | NMSE |
|---|---|---|---|
| 0.836962 | 13.835028 | 0.403732 | 0.162999 |

Παρατηρούμε, ότι το μοντέλο για τις βέλτιστες τιμές των παραμέτρων που επιλέχθηκαν είναι πολύ ακριβές. Ο χρόνος για να τρέξει ο το μοντέλο ήταν περίπου 15 λεπτά. Στο figure 20 βλέπουμε ότι το validation error ξεκινάει από το 15.3 και το training από το 14.7 και καταλήγουν στο 14.3 και 13.3 αντίστοιχα, χωρίς την ύπαρξη ταλαντώσεων ή υπερεκπαίδευσης. Στο figure 21 και 22 βλέπουμε τις συναρτήσεις συμμετοχής πριν και μετά την εκπαίδευση του μοντέλου και στο figure 23 βλέπουμε το σφάλμα το οποίο βρίσκεται στο διάστημα [-30, 30]. Στα figure 24 και 25 παρατηρούμε την διαφορά της πραγματικής και της εκτιμώμενης εξόδου σε ολόκληρο το εύρος της, αλλά και σε μία εστιασμένη περίπτωσή της. Στο εστιασμένο κομμάτι που επιλέχθηκε μπορούμε να δούμε πιο καθαρά την διαφορά των δύο αυτών εξόδων και παρατηρούμε πόσο καλά προσεγγίζει το μοντέλο την πραγματική τιμή.

## Συμπεράσματα

Εν τέλει καταφέραμε να μειώσουμε τον σημαντικά τον χρόνο που κάνει για να τρέξει ένα μοντέλο για το αντίστοιχο dataset, κάνοντας το να τρέχει σε 15 λεπτά από έναν απαγορευτικό χρόνο και μάλιστα να έχει καλή απόδοση. Αυτό έγινε με την μείωση των κανόνων με την επιλογή χαρακτηριστικών και της ακτίνας. Αν είχαμε επιλέξει grid-partitioning με δύο ή τρία ασαφή σύνολα ανά είσοδο, θα είχαμε πολύ χειρότερη απόδοση για πολύ μεγαλύτερο χρόνο από αυτό που καταφέραμε σε αυτή την ενότητα. Επομένως το δεύτερο κεφάλαιο είναι πολύ πιο αποτελεσματικό για αυτό το dataset από το 1ο.