

ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ  
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

# Ασαφή Συστήματα

*Εργασία #1 ~ FLC DC-Motor 4*

*Εργασία #2 ~ FLC Car Control A*

*Εργασία #3 ~ Regression*

*Εργασία #4 ~ Classification*

Δημήτρης Παππάς

AEM: 8391

e-mail: [dspappas@ece.auth.gr](mailto:dspappas@ece.auth.gr)



## Περιεχόμενα

1. Εργασία #1: FLC DC-Motor 4.....	4
1.1 Ελεγχόμενο Σύστημα .....	4
1.2 Σχεδίαση Ασαφούς Ελεγκτή.....	5
1.2.1 Συναρτήσεις Συμμετοχής .....	6
1.2.2 Επιλογή Τελεστών.....	9
1.2.3 Βάση Κανόνων .....	10
1.2.4 Simulink Μοντέλο .....	13
1.2.5 Ρύθμιση Κερδών .....	14
1.3 Υλοποίηση Σεναρίων.....	19
1.3.1 Σενάριο 1.....	19
1.3.2 Σενάριο 2.....	38
1.3.3 Σενάριο 3.....	44
2. Εργασία #2: FLC Car Control A.....	51
2.1 Συναρτήσεις Συμμετοχής.....	51
2.2 Βάση Κανόνων .....	57
2.2.1 Αρχική Βάση Κανόνων .....	57
2.2.2 Τελική Βάση Κανόνων.....	63
3. Εργασία #3: Regression.....	70
3.1 Part 1 - Εφαρμογή σε απλό dataset.....	70
3.1.1 TSK model 1.....	71
3.1.2 TSK model 2.....	74
3.1.3 TSK model 3.....	77
3.1.4 TSK model 4.....	80
3.1.5 Συμπεράσματα και Επιλογή μοντέλου .....	84
3.2 Part 2 - Εφαρμογή σε dataset με υψηλή διαστασιμότητα.....	85
3.2.1 Αποτελέσματα Σφαλμάτων του Grid Search .....	85
3.2.2 Διαγράμματα Grid Search.....	87
3.3.3 Βέλτιστο Μοντέλο.....	90
4. Εργασία #4: Classification .....	96
4.1 Part 1 - Εφαρμογή σε απλό dataset.....	96
4.1.1 TSK model 1.....	98
4.1.2 TSK model 2.....	101

4.1.3 TSK model 3.....	104
4.1.4 TSK model 4.....	107
4.1.5 Αξιολόγηση Μοντέλων και Πίνακες Σφαλμάτων.....	110
4.1.6 Συμπεράσματα και Βέλτιστο Μοντέλο .....	112
4.2 Part 2 - Εφαρμογή σε dataset με υψηλή διαστασιμότητα.....	113
4.2.1 Αποτελέσματα του Grid Search .....	114
4.2.2 Διαγράμματα του Grid Search .....	114
4.3.3 Βέλτιστο Μοντέλο.....	117

# 1. Εργασία #1: FLC DC-Motor 4

## 1.1 Ελεγχόμενο Σύστημα

$$\Omega(s) = \frac{18.69}{s + 12.064} V_a(s) - \frac{2.92(s + 440)}{s + 12.064} T_L(s)$$

Σχήμα 1: Συνάρτηση Μεταφοράς DC Κινητήρα

Για το παραπάνω σύστημα (Σχήμα 1) θα σχεδιαστεί ένας κλασικός ελεγκτής σύμφωνα με το παράδειγμα 9.1.1 των σημειώσεων του μαθήματος Υπολογιστική Νοημοσύνη και στη συνέχia ένας Ασαφής PI ελεγκτής.

Στόχος είναι να πιάσουμε τις παρακάτω προδιαγραφές:

- α) Απόρριψη Διαταραχών: Για κυκλική συχνότητα διαταραχής μικρότερη από 1 rad/sec να υπάρχει κέρδος διαταραχής το πολύ 20 dB. Επιπλέον, θέλουμε αν υπάρξει μεταβολή της TL, η οποία να μεταβληθεί παροδικά και μετά να επιστρέψει (με καλή προσέγγιση) στην τιμή που είχε πριν από τη διαταραχή.
- β) Το πολύ 5% υπερύψηση για βηματική είσοδο.
- γ) Μηδενικό σφάλμα θέσης
- δ) Χρόνος ανόδου το πολύ 160 msec.
- ε)  $V_a(t) \leq 200V$  για κάθε  $t > 0$ .

Η Συνάρτηση Μεταφοράς του DC-Motor είναι:

$$\Omega(s) = \frac{18.69}{s + 12.064} V_a(s) - \frac{2.92(s + 440)}{s + 12.064} T_L(s)$$

Όπου  $V_a$  είναι η είσοδος του κινητήρα και  $T_L$  είναι η διαταραχή από τη ροπή του φορτίου.

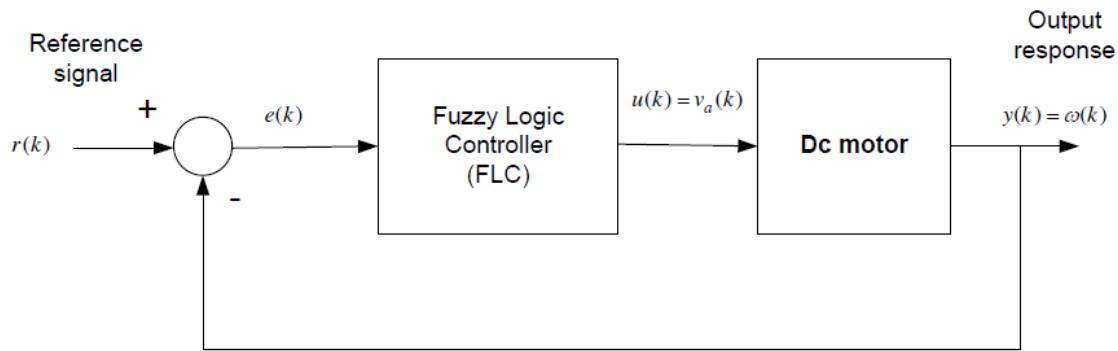
Γνωρίζουμε από το παράδειγμα 9.1.1 ότι τα αρχικά κέρδη ενός DC κινητήρα είναι:

$$K_p = 1.75$$

$$K_i = 8.75$$

Αρχικά, θεωρούμε ότι οι διαταραχές είναι μηδέν  $T_L = 0$ .

## 1.2 Σχεδίαση Ασαφούς Ελεγκτή



Σχήμα 2: Ασαφής Ελεγκτής

Επιλέγουμε FZ-PI ασαφή ελεγκτή, ο οποίος ικανοποιεί την προδιαγραφή για μηδενικό σφάλμα θέσης. Η υλοποίηση του ελεκτή θα γίνει μέσω Simulink, όπου χρησιμοποιείται συχνότητα δειγματοληψίας  $T=0.01$ . Ο Ασαφής ελεγκτής θα έχει τη μορφή του Σχήματος 2.

Αρχικά, κάνουμε κανονικοποίηση σφάλματος στο διάστημα  $[-1,1]$ .

Η μέγιστη τιμή της γωνιακής ταχύτητας  $\omega$  είναι  $150 \text{ rad/sec}$ .

Η μέγιστη τιμή της εισόδου στον DC κινητήρα είναι  $200V$ .

$$e_{max} = r_{max} - r_{min} = 150 - 0 = 150$$

$$e_{min} = r_{min} - r_{max} = 0 - 150 = -150$$

Άρα, το πεδίο ορισμού του σφάλματος είναι το  $[-150, 150]$  και ομοιώς το πεδίο ορισμού της μεταβολής του σφάλματος  $de$  είναι το  $[-50, 50]$ .

- Για το σφάλμα  $e$ :  $E = e/150$
- Για τη μεταβολή σφάλματος  $de$ :  $DE = de/50$
- Για την έξοδο του ελεγκτή  $du$ :  $DU = du/200$

Μετά την κανονικοποίηση το πεδίο ορισμού είναι το [-1,1].

Οι είσοδη του ελεγκτή είναι το σφάλμα ε και η μεταβολή του σφάλματος de, οι οποίες είναι κανονικοποιημένες στο διάστημα [-1,1].

Τα αρχικά κέρδη του Ασαφούς ελεκτή είναι:

$$a = K_p/K_i = 0.2$$

$$K_e = 1$$

$$K_d = a * K_e = 0.2$$

$$K = \frac{K_e}{F\{a*K_e\}} = \frac{1.75}{F\{0.2*1\}} = 8.75$$

### 1.2.1 Συναρτήσεις Συμμετοχής

Η είσοδος ε περιγράφεται από 9 λεκτικές τιμές, ενώ η de από 7 λεκτικές τιμές και η έξοδος από 9 λεκτικές τιμές, όπως φαίνεται παρακάτω.

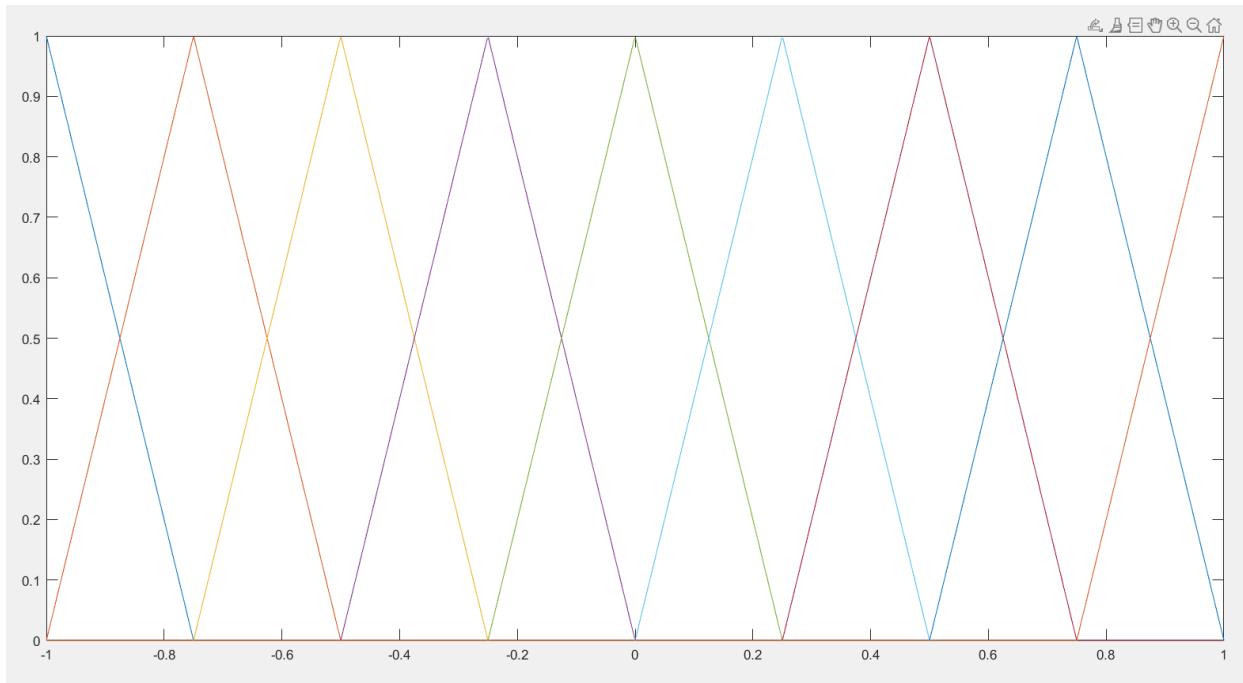
Οι Συναρτήσεις Συμμετοχής υλοποιούνται με τριγωνική mf.

```
% Συνάρτηση Συμμετοχής (Membership Function)
% Για την είσοδο e
fuzzy_pi = addmf(fuzzy_pi, 'input', 1, 'NV', 'trimf', [-1,-1,-0.75]); % trimf = Triangular membership function
fuzzy_pi = addmf(fuzzy_pi, 'input', 1, 'NL', 'trimf', [-1,-0.75,-0.5]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 1, 'NM', 'trimf', [-0.75,-0.5,-0.25]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 1, 'NS', 'trimf', [-0.5,-0.25,0]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 1, 'ZR', 'trimf', [-0.25,0,0.25]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 1, 'PS', 'trimf', [0,0.25,0.5]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 1, 'PM', 'trimf', [0.25,0.5,0.75]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 1, 'PL', 'trimf', [0.5,0.75,1]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 1, 'PV', 'trimf', [0.75,1,1]);

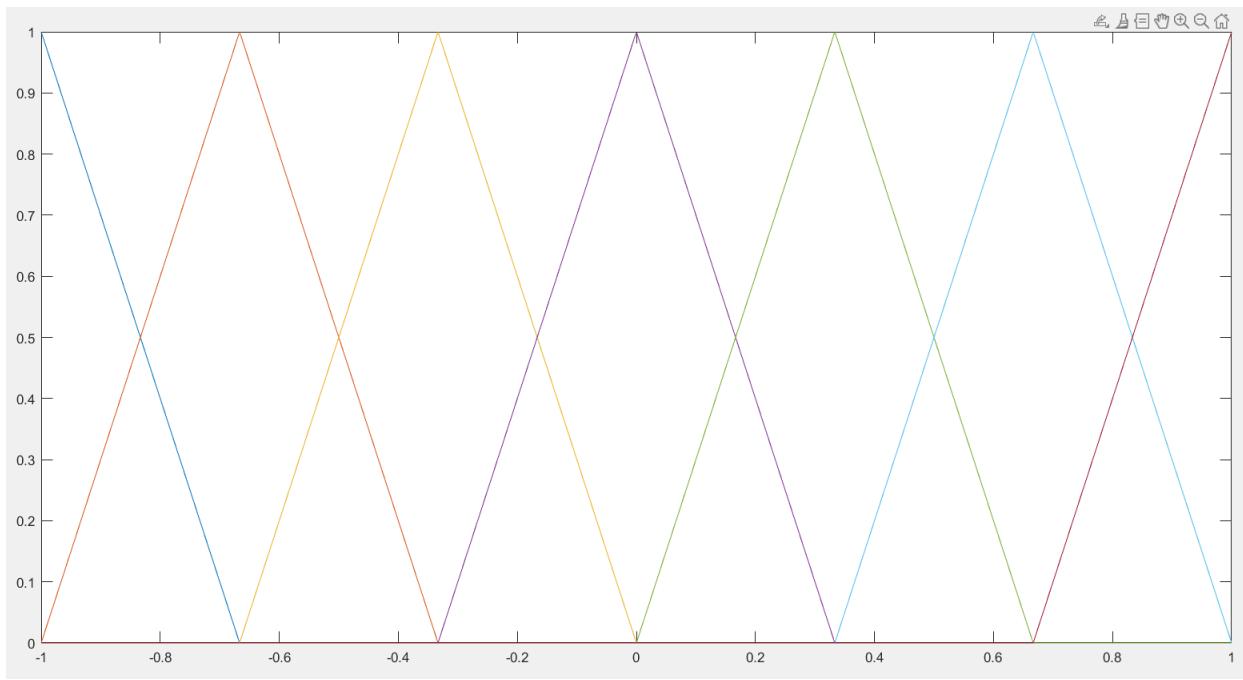
% Για την είσοδο de
fuzzy_pi = addmf(fuzzy_pi, 'input', 2, 'NL', 'trimf', [-1,-1,-2/3]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 2, 'NM', 'trimf', [-1,-2/3,-1/3]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 2, 'NS', 'trimf', [-2/3,-1/3,0]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 2, 'ZR', 'trimf', [-1/3,0,1/3]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 2, 'PS', 'trimf', [0,1/3,2/3]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 2, 'PM', 'trimf', [1/3,2/3,1]);
fuzzy_pi = addmf(fuzzy_pi, 'input', 2, 'PL', 'trimf', [2/3,1,1]);

% Για την έξοδο du
fuzzy_pi = addmf(fuzzy_pi, 'output', 1, 'NV', 'trimf', [-1,-1,-0.75]);
fuzzy_pi = addmf(fuzzy_pi, 'output', 1, 'NL', 'trimf', [-1,-0.75,-0.5]);
fuzzy_pi = addmf(fuzzy_pi, 'output', 1, 'NM', 'trimf', [-0.75,-0.5,-0.25]);
fuzzy_pi = addmf(fuzzy_pi, 'output', 1, 'NS', 'trimf', [-0.5,-0.25,0]);
fuzzy_pi = addmf(fuzzy_pi, 'output', 1, 'ZR', 'trimf', [-0.25,0,0.25]);
fuzzy_pi = addmf(fuzzy_pi, 'output', 1, 'PS', 'trimf', [0,0.25,0.5]);
fuzzy_pi = addmf(fuzzy_pi, 'output', 1, 'PM', 'trimf', [0.25,0.5,0.75]);
fuzzy_pi = addmf(fuzzy_pi, 'output', 1, 'PL', 'trimf', [0.5,0.75,1]);
fuzzy_pi = addmf(fuzzy_pi, 'output', 1, 'PV', 'trimf', [0.75,1,1]);
```

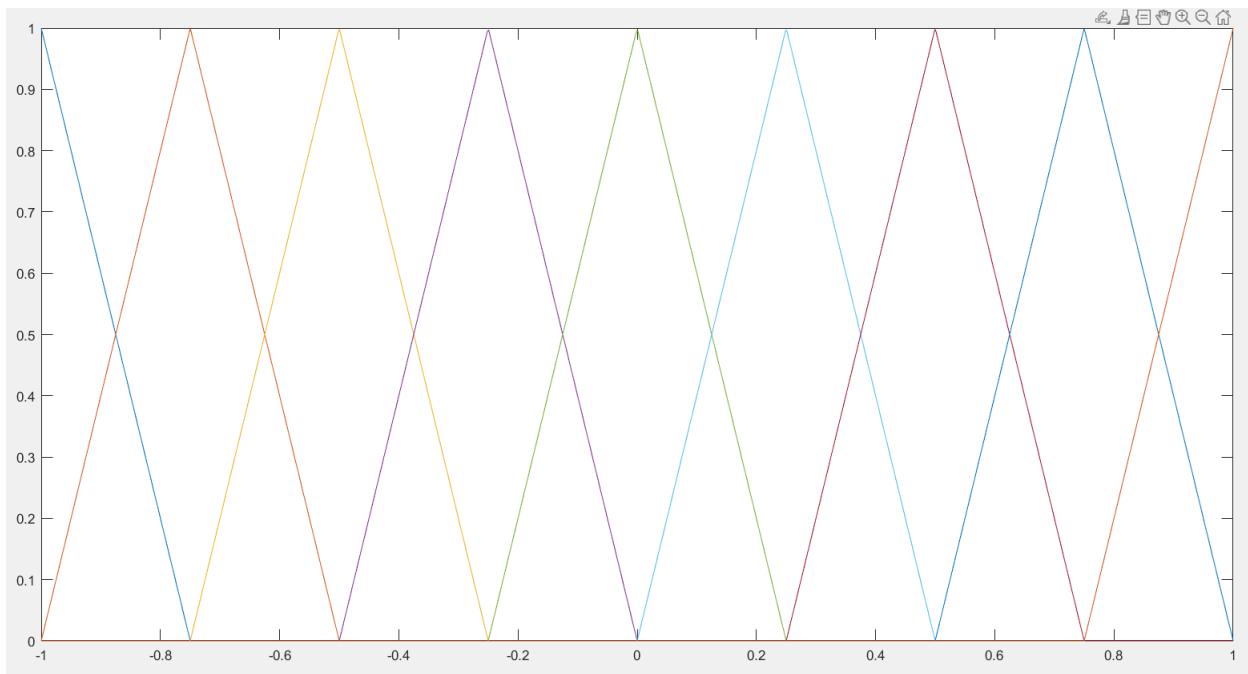
Σχήμα 3: Κώδικας Συναρτήσεων Μεταφοράς



Σχήμα 4: Συνάρτηση Συμμετοχής Input e

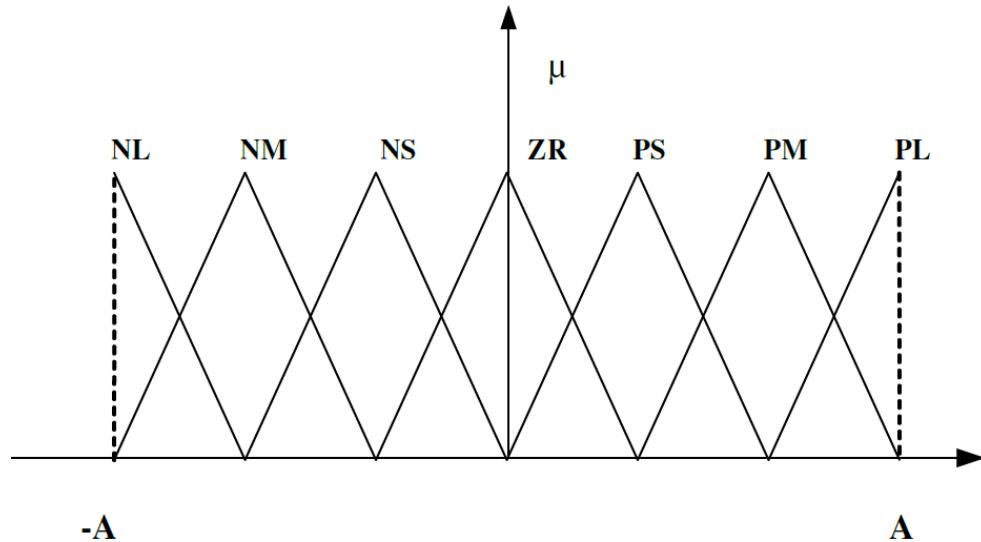


Σχήμα 5: Συνάρτηση Συμμετοχής Input de



Σχήμα 6: Συνάρτηση Συμμετοχής Output du

Οι λεκτικές τιμές ακολουθούν τους κανόνες της εκφώνησης.



NS: Neg. Small

NL: Neg. Large

ZR: Zero

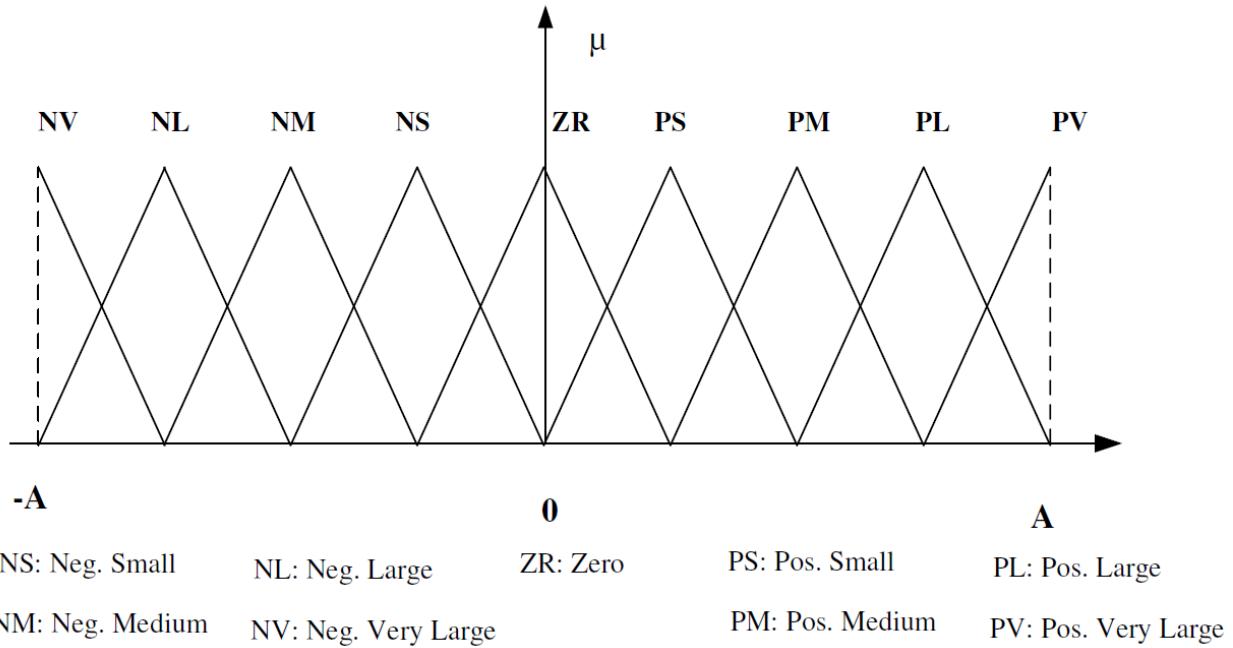
PS: Pos. Small

PL: Pos. Large

NM: Neg. Medium

PM: Pos. Medium

Σχήμα 7: 7 λεκτικές τιμές



Σχήμα 8: 9 λεκτικές τιμές

### 1.2.2 Επιλογή Τελεστών

Για την υλοποίηση του ασαφούς ελεγκτή επιλέγουμε παρακάτω τελεστές.

- Ασαφοποιητή Singleton
- Το συνδετικό AND υλοποιείται με τον τελεστή min.
- Το συνδετικό OR υλοποιείται με τον τελεστή max.
- Η συνάρτηση συμπερασμού υλοποιείται με τον κανόνα Mamdani.
- Το συνδετικό ALSO υλοποιείται με τον τελεστή max.
- Ο αποασαφοποιητής υλοποιείται με την τεχνική COA (center of area).

```
% Ασαφές Σύστημα
fuzzy_pi = newfis('fzpi', ... % name of FIS system
  'mamdani', ... % mamdani type
  'min', ... % το συνδετικό AND υλοποιείται με τελεστή min
  'max', ... % το συνδετικό OR υλοποιείται με τελεστή max
  'min', ... % το Implication υλοποιείται με τελεστή min
  'max', ... % το Aggregation υλοποιείται με τελεστή max
  'centroid'); % ο Defuzzifier υλοποιείται με τεχνική centroid (COA)
```

Το ασαφές σύστημα δημιουργείται στο script “dc\_motor.m”.

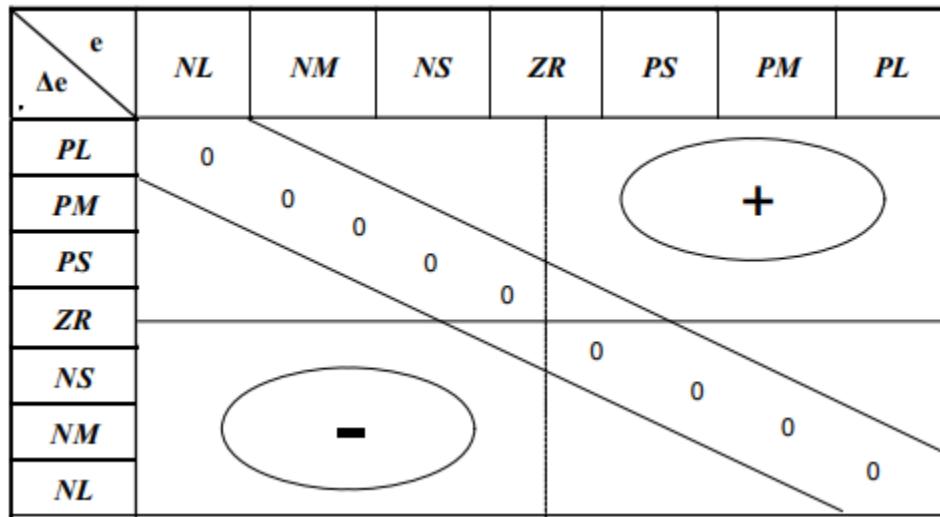
Τα στοιχεία του Ασαφούς συστήματος φαίνονται στο αρχείο “fuzzy\_pi.fis”.

### 1.2.3 Βάση Κανόνων

Οι κανόνες ακολουθούν τη λογική του Σχήματος 9 και του Σχήματος 10.

e\de	PL	PM	PS	ZR	NS	NM	NL
NV	NS	NM	NL	NV	NV	NV	NV
NL	ZR	NS	NM	NL	NV	NV	NV
NM	PS	ZR	NS	NM	NL	NV	NV
NS	PM	PS	ZR	NS	NM	NL	NV
ZR	PL	PM	PS	ZR	NS	NM	NL
PS	PV	PL	PM	PS	ZR	NS	NM
PM	PV	PV	PL	PM	PS	ZR	NS
PL	PV	PV	PV	PL	PM	PS	ZR
PV	PV	PV	PV	PV	PL	PM	PS

Σχήμα 9: Βάση Κανόνων



Σχήμα 10: Βάση Κανόνων part2

Η Βάση Κανόνων αποτελείται από 63 κανόνες:

- '1. If (e is NV) and (de is NL) then (du is NV) (1) '
- '2. If (e is NV) and (de is NM) then (du is NV) (1) '
- '3. If (e is NV) and (de is NS) then (du is NV) (1) '
- '4. If (e is NV) and (de is ZR) then (du is NV) (1) '
- '5. If (e is NV) and (de is PS) then (du is NL) (1) '

- '6. If (e is NV) and (de is PM) then (du is NM) (1) '
- '7. If (e is NV) and (de is PL) then (du is NS) (1) '
- '8. If (e is NL) and (de is NL) then (du is NV) (1) '
- '9. If (e is NL) and (de is NM) then (du is NV) (1) '
- '10. If (e is NL) and (de is NS) then (du is NV) (1)'
- '11. If (e is NL) and (de is ZR) then (du is NL) (1)'
- '12. If (e is NL) and (de is PS) then (du is NM) (1)'
- '13. If (e is NL) and (de is PM) then (du is NS) (1)'
- '14. If (e is NL) and (de is PL) then (du is ZR) (1)'
- '15. If (e is NM) and (de is NL) then (du is NV) (1)'
- '16. If (e is NM) and (de is NM) then (du is NV) (1)'
- '17. If (e is NM) and (de is NS) then (du is NL) (1)'
- '18. If (e is NM) and (de is ZR) then (du is NM) (1)'
- '19. If (e is NM) and (de is PS) then (du is NS) (1)'
- '20. If (e is NM) and (de is PM) then (du is ZR) (1)'
- '21. If (e is NM) and (de is PL) then (du is PS) (1)'
- '22. If (e is NS) and (de is NL) then (du is NV) (1)'
- '23. If (e is NS) and (de is NM) then (du is NL) (1)'
- '24. If (e is NS) and (de is NS) then (du is NM) (1)'
- '25. If (e is NS) and (de is ZR) then (du is NS) (1)'
- '26. If (e is NS) and (de is PS) then (du is ZR) (1)'
- '27. If (e is NS) and (de is PM) then (du is PS) (1)'
- '28. If (e is NS) and (de is PL) then (du is PM) (1)'
- '29. If (e is ZR) and (de is NL) then (du is NL) (1)'
- '30. If (e is ZR) and (de is NM) then (du is NM) (1)'
- '31. If (e is ZR) and (de is NS) then (du is NS) (1)'
- '32. If (e is ZR) and (de is ZR) then (du is ZR) (1)'

- '33. If (e is ZR) and (de is PS) then (du is PS) (1)'
- '34. If (e is ZR) and (de is PM) then (du is PM) (1)'
- '35. If (e is ZR) and (de is PL) then (du is PL) (1)'
- '36. If (e is PS) and (de is NL) then (du is NM) (1)'
- '37. If (e is PS) and (de is NM) then (du is NS) (1)'
- '38. If (e is PS) and (de is NS) then (du is ZR) (1)'
- '39. If (e is PS) and (de is ZR) then (du is PS) (1)'
- '40. If (e is PS) and (de is PS) then (du is PM) (1)'
- '41. If (e is PS) and (de is PM) then (du is PL) (1)'
- '42. If (e is PS) and (de is PL) then (du is PV) (1)'
- '43. If (e is PM) and (de is NL) then (du is NS) (1)'
- '44. If (e is PM) and (de is NM) then (du is ZR) (1)'
- '45. If (e is PM) and (de is NS) then (du is PS) (1)'
- '46. If (e is PM) and (de is ZR) then (du is PM) (1)'
- '47. If (e is PM) and (de is PS) then (du is PL) (1)'
- '48. If (e is PM) and (de is PM) then (du is PV) (1)'
- '49. If (e is PM) and (de is PL) then (du is PV) (1)'
- '50. If (e is PL) and (de is NL) then (du is ZR) (1)'
- '51. If (e is PL) and (de is NM) then (du is PS) (1)'
- '52. If (e is PL) and (de is NS) then (du is PM) (1)'
- '53. If (e is PL) and (de is ZR) then (du is PL) (1)'
- '54. If (e is PL) and (de is PS) then (du is PV) (1)'
- '55. If (e is PL) and (de is PM) then (du is PV) (1)'
- '56. If (e is PL) and (de is PL) then (du is PV) (1)'
- '57. If (e is PV) and (de is NL) then (du is PS) (1)'
- '58. If (e is PV) and (de is NM) then (du is PM) (1)'
- '59. If (e is PV) and (de is NS) then (du is PL) (1)'

'60. If (e is PV) and (de is ZR) then (du is PV) (1)'

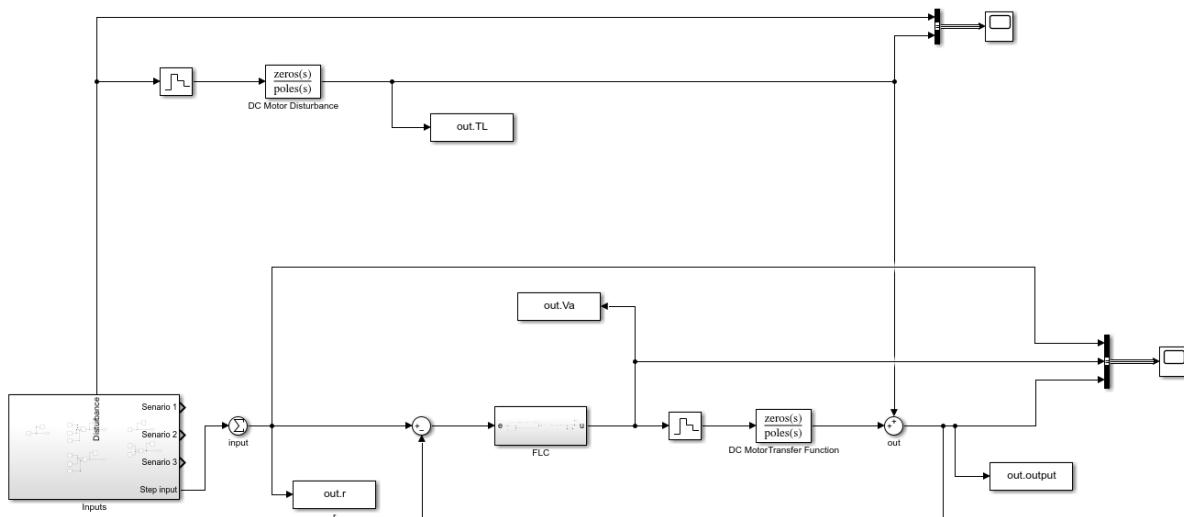
'61. If (e is PV) and (de is PS) then (du is PV) (1)'

'62. If (e is PV) and (de is PM) then (du is PV) (1)'

'63. If (e is PV) and (de is PL) then (du is PV) (1)'

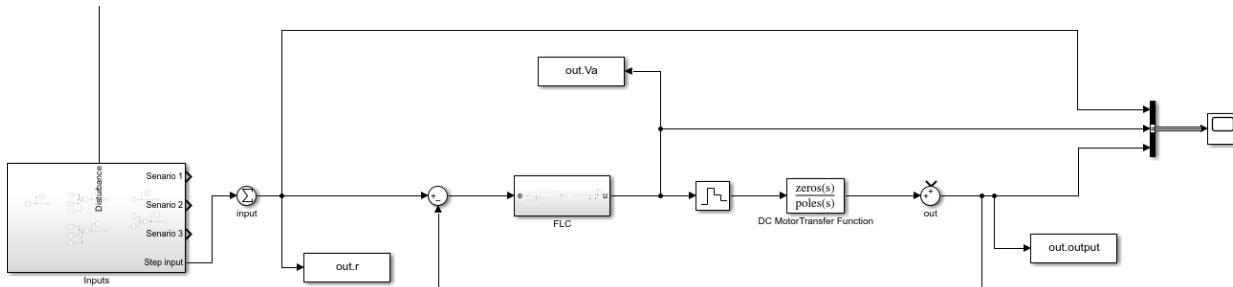
#### 1.2.4 Simulink Μοντέλο

Το τελικό μοντέλου του Ασαφή ελεγκτή φαίνεται στο Σχήμα 11.



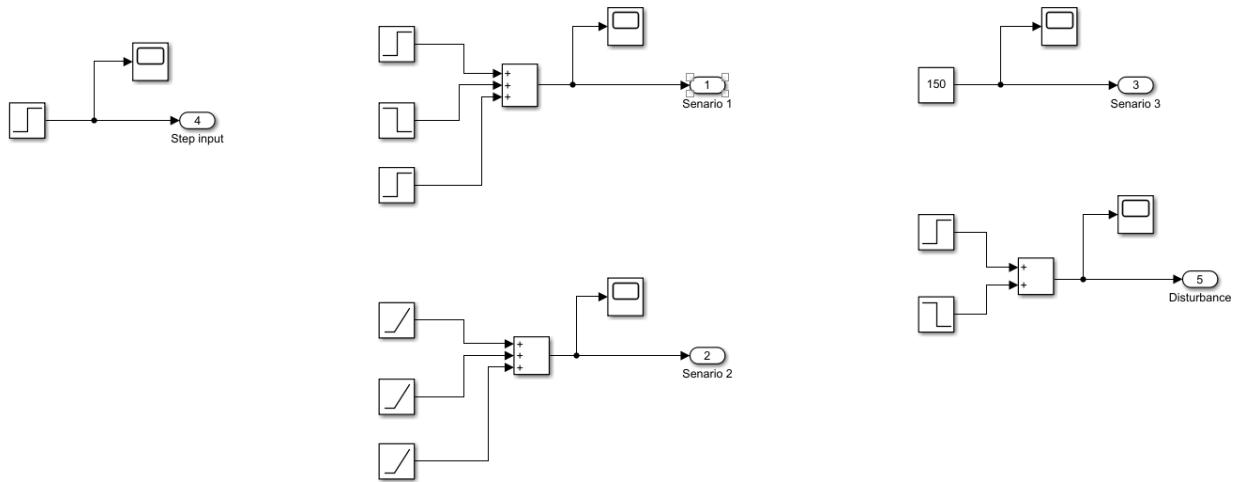
Σχήμα 11: Simulink Μοντέλο

Το μοντέλο χωρίς διαταραχές φαίνεται στο Σχήμα 12.



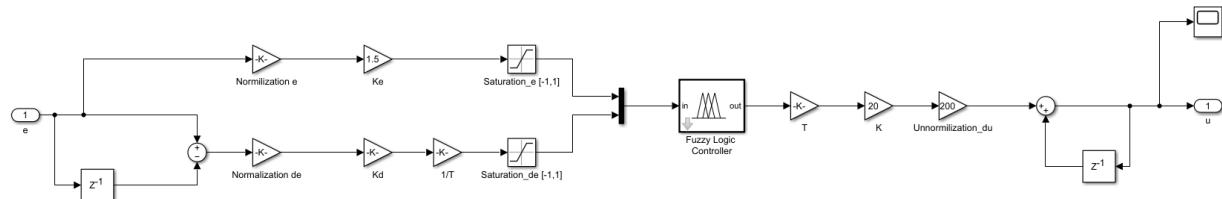
Σχήμα 12: Μοντέλο χωρίς διαταραχές

Οι είσοδοι στο μοντέλο περιέχονται στο Subsystem “Inputs”, όπως φαίνεται στο Σχήμα 13.



Σχήμα 13: Inputs

Ο Ασαφής Ελεγκτής FLC υλοποιείται στο Subsystem “FLC”, όπως φαίνεται στο Σχήμα 14.



Σχήμα 14: FLC FZ-PI

## 1.2.5 Ρύθμιση Κερδών

### 1.2.5.1 Αρχικά Κέρδη

Με βάση τα αρχικά κέρδη του κλασικού ελεγκτή έχουμε

$$a = 0.2$$

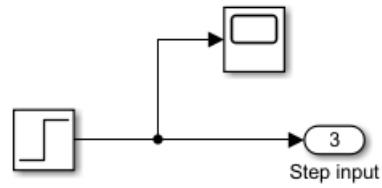
$$K_e = 1$$

$$K_d = 0.2$$

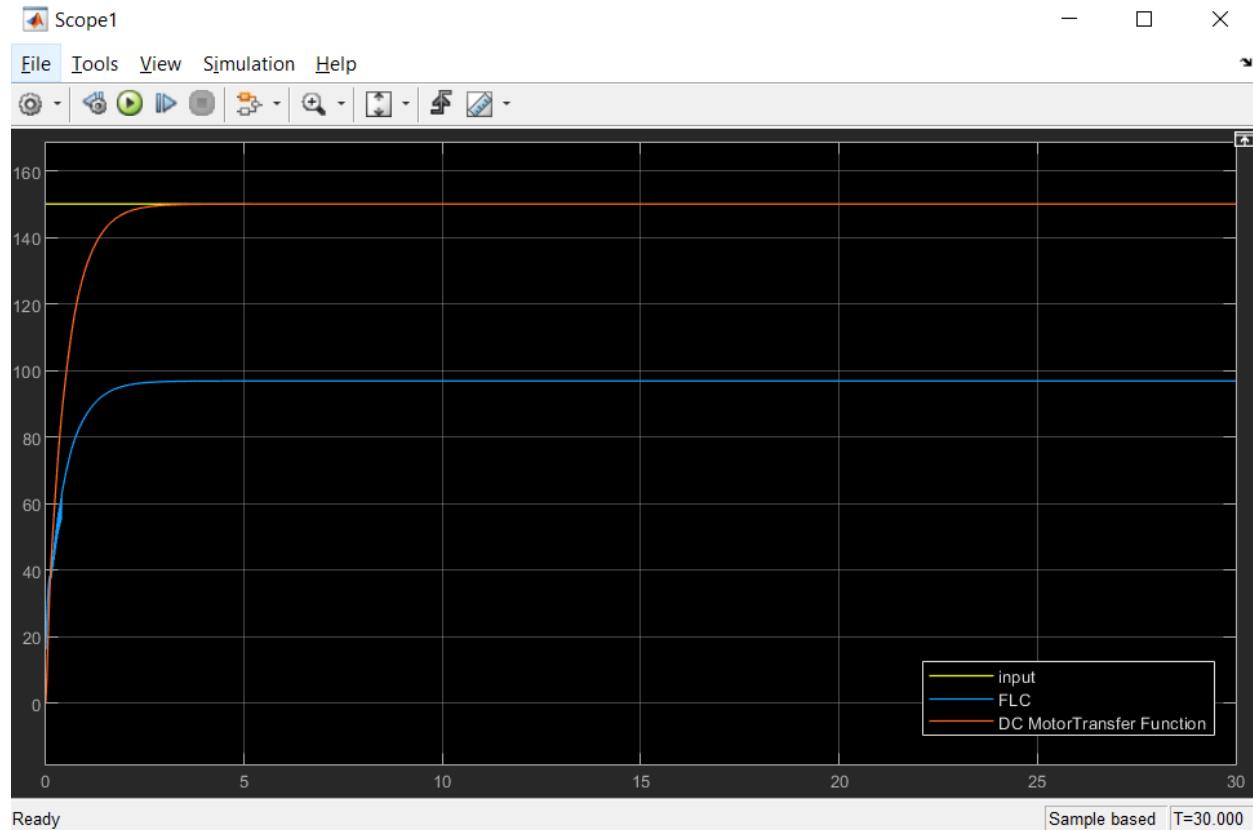
$$K = 8.75$$

Εισάγουμε μία βηματική είσοδο και προσομοιώνουμε τον ελεγκτήμε αυτά τα κέρδη, για να ελέγξουμε τις προδιαγραφές.

Το σήμα εισόδου δημιουργείται σύμφωνα με το Σχήμα 15.



Σχήμα 15: Βηματική είσοδος



Σχήμα 16: Απόκριση - Βηματική είσοδος – Αρχικά κέρδη

Παρατηρούμε ότι υπάρχει μεγάλος χρόνος ανόδου.

rise time = 1.071 > 0.16 sec

Δεν πιάνουμε την προδιαγραφή.

Bilevel Measurements	
Settings	
High	1.492e+02
Low	7.500e-01
Amplitude	1.485e+02
+ Edges	1
+ Rise Time	1.071 s
+ Slew Rate	110.884 (/s)
- Edges	0
- Fall Time	--
- Slew Rate	--
Transitions	
Overshoots / Undershoots	
+ Preshoot	0.505 %
+ Overshoot	0.502 %
+ Undershoot	1.953 %
+ Settling Time	--
- Preshoot	--
- Overshoot	--
- Undershoot	--
- Settling Time	--

Σχήμα 17: Βηματική είσοδος – Αρχικά κέρδη - Stats

#### 1.2.5.2 Τελικά Κέρδη

Κάνουμε tuning και καταλήγουμε στα κέρδη:

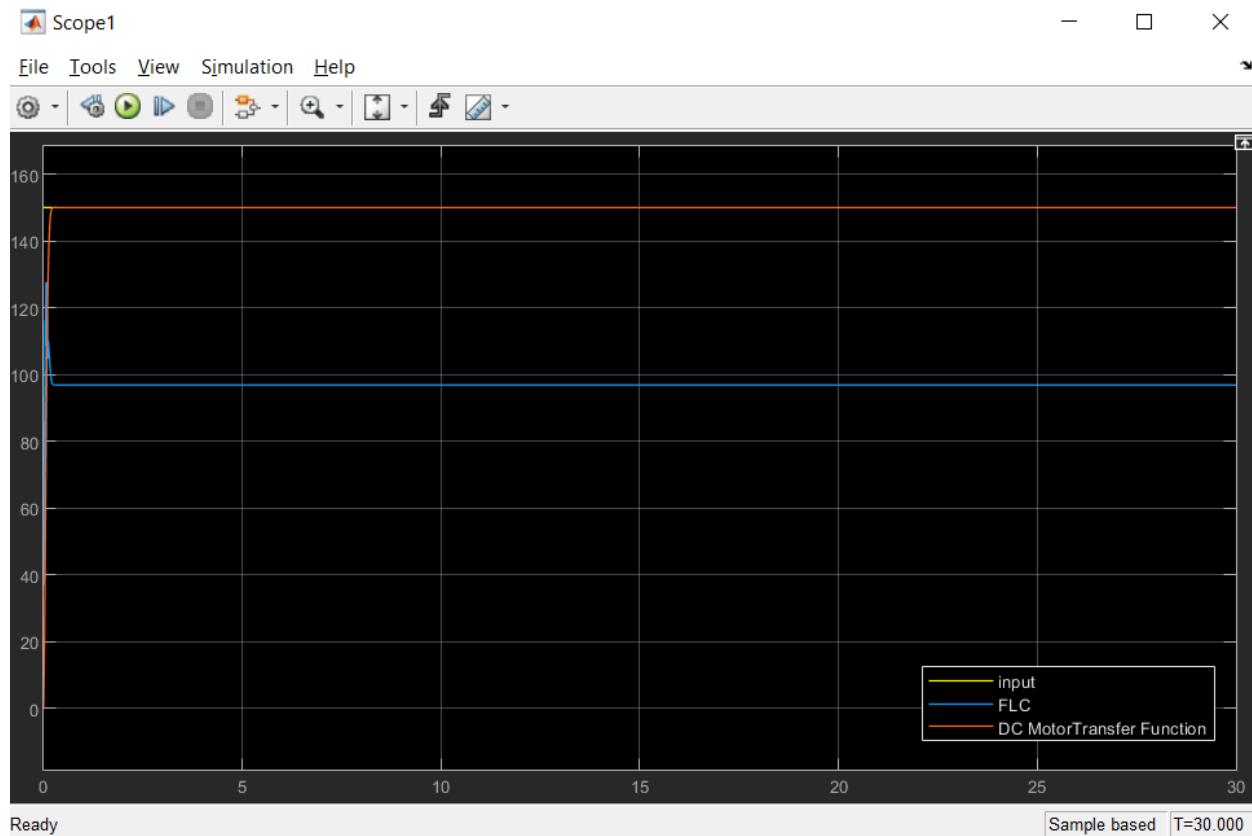
$$a = 0.02$$

$$K_e = 1.5$$

$$K = 20$$

Με τα νέα κέρδη έχουμε πιο γρήγορη απόκριση.

Προσομοιώνουμε το μοντέλο με τα νέα κέρδη.



Σχήμα 18: Απόκριση - Βηματική είσοδος – Τελικά κέρδη

Παρατηρούμε ότι ο χρόνος ανόδου έχει μειωθεί δραματικά.

$\text{rise time} = 0.104942 < 0.160 \text{ sec}$

$\text{overshoot} = 0.5\% < 5\%$

$V_a < 200V$

Πιάνουμε τις προδιαγραφές.

Bilevel Measurements	
► Settings	
▼ Transitions	
High	1.493e+02
Low	7.500e-01
Amplitude	1.485e+02
+ Edges	1
+ Rise Time	104.942 ms
+ Slew Rate	1.132 (/ms)
- Edges	0
- Fall Time	--
- Slew Rate	--
▼ Overshoots / Undershoots	
+ Preshoot	0.505 %
+ Overshoot	0.505 %
+ Undershoot	1.408 %
+ Settling Time	--
- Preshoot	--
- Overshoot	--
- Undershoot	--
- Settling Time	--

Σχήμα 19: Βηματική είσοδος – Τελικά κέρδη - Stats

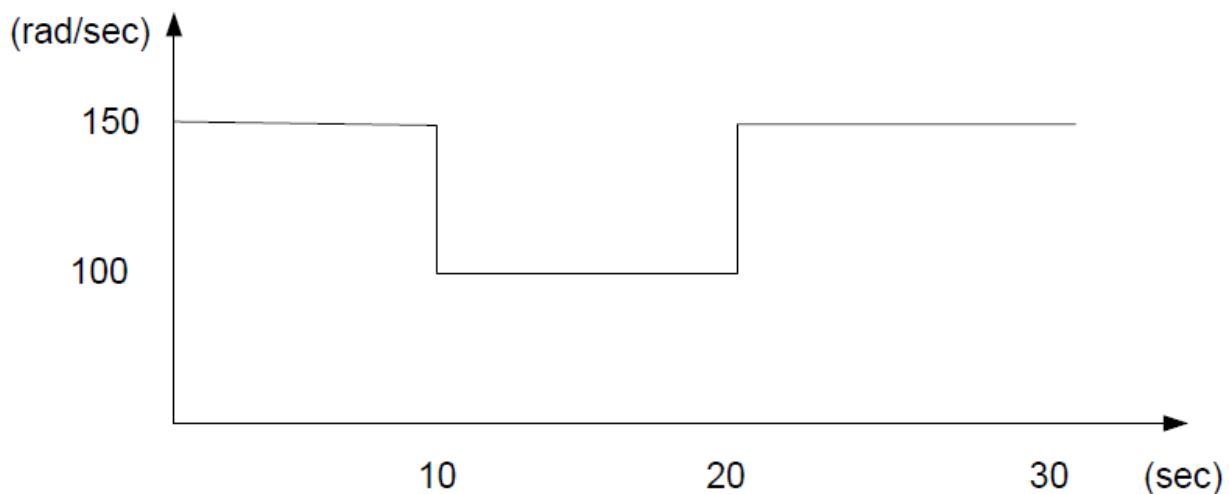
## 1.3 Υλοποίηση Σεναρίων

Παρακάτω θα δώσουμε διάφορα σήματα στην είσοδο του μοντέλου και θα βγάλουμε συμπεράσματα για το σήμα εξόδου και για τους κανόνες που διεγείρονται.

### 1.3.1 Σενάριο 1

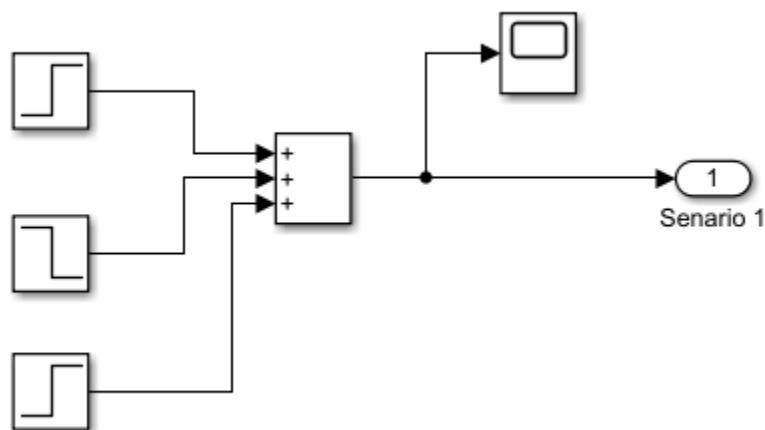
Το σύστημα υλοποιείται με βάση το μοντέλο του Σχήματος 12.

Δίνουμε ως είσοδο στο μοντέλο το σήμα του Σχήματος 20.



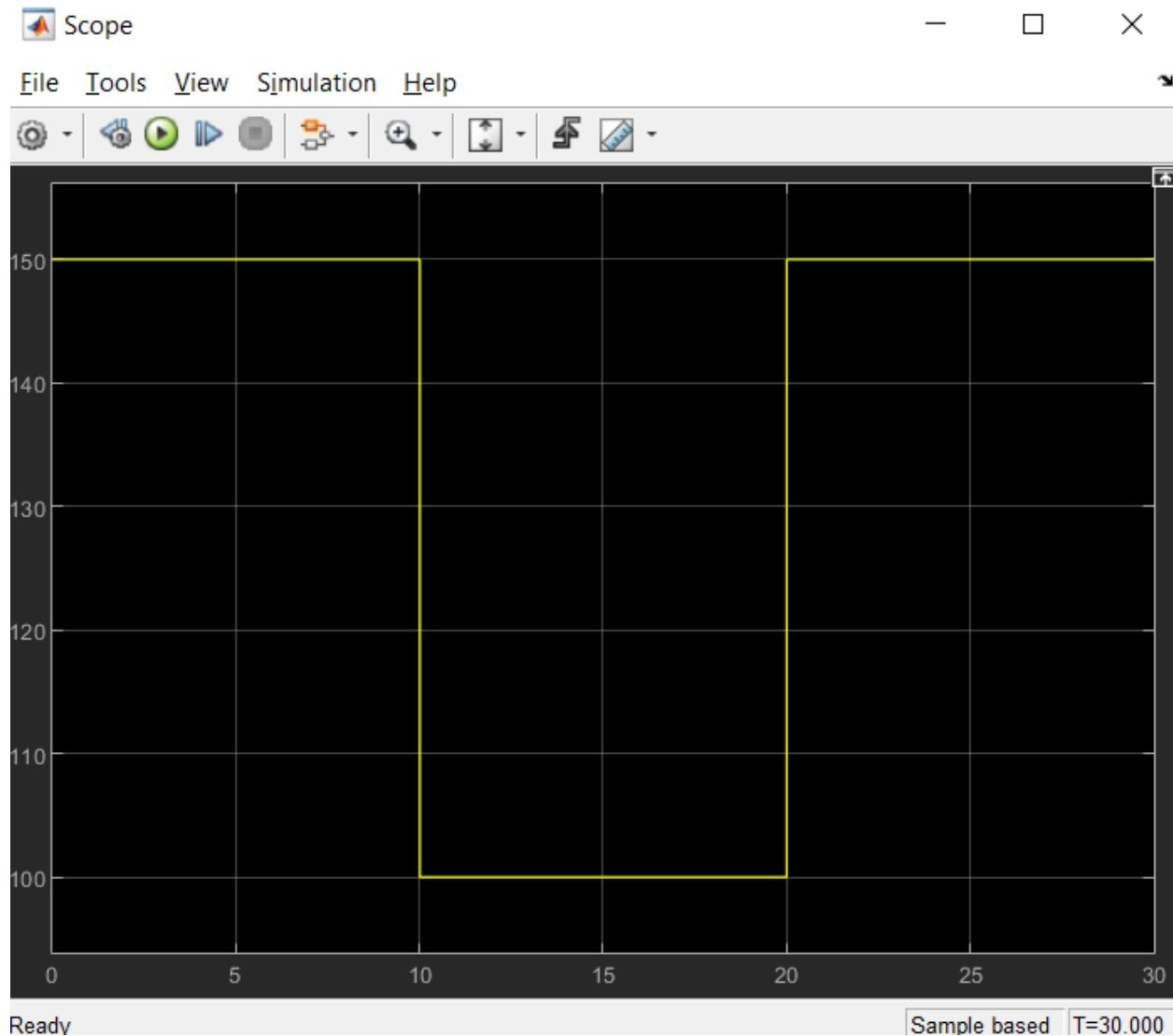
Σχήμα 20: Σενάριο 1 – Είσοδος

Κατασκευάζουμε το σήμα εισόδου στο Simulink.



Σχήμα 21: Σενάριο 1 – Είσοδος Simulink Μοντέλο

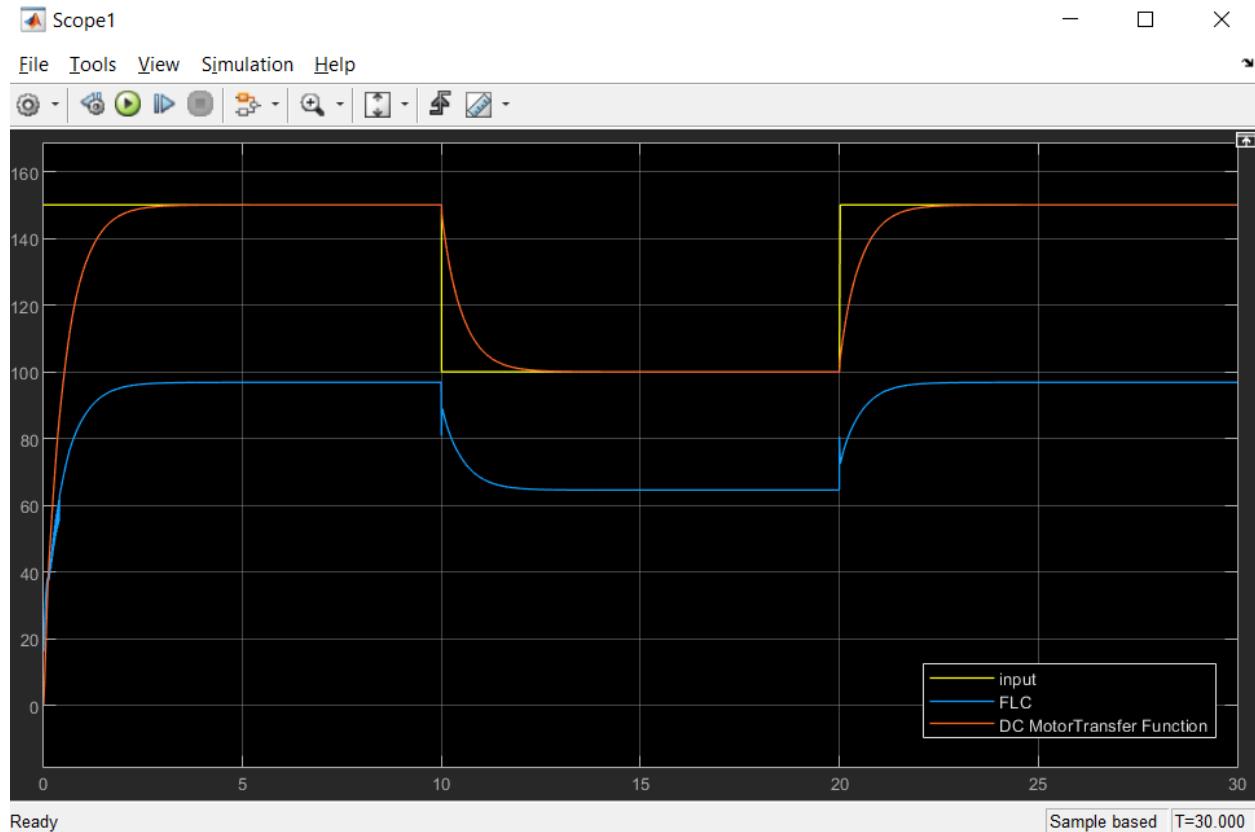
Πατατηρούμε ότι η είσοδος που δημιουργούμε είναι η επιθυμητή.



Σχήμα 22: Σενάριο 1 – Είσοδος Simulink

Θέλουμε η έξοδος να ακολουθεί πιστά το σήμα εισόδου.

Αρχικά, βλέπουμε την απόκριση με τα αρχικά κέρδη.

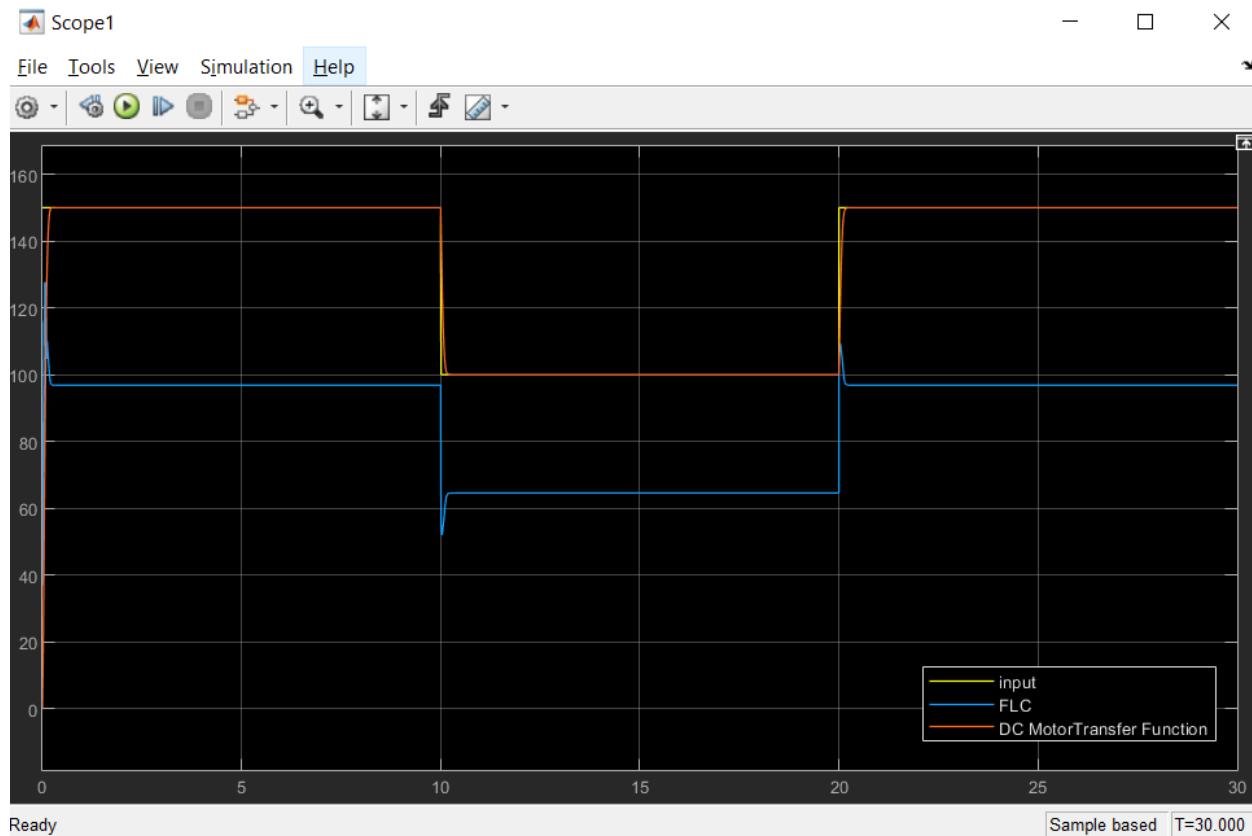


Σχήμα 23: Σενάριο 1 - Απόκριση - Αρχικά κέρδη

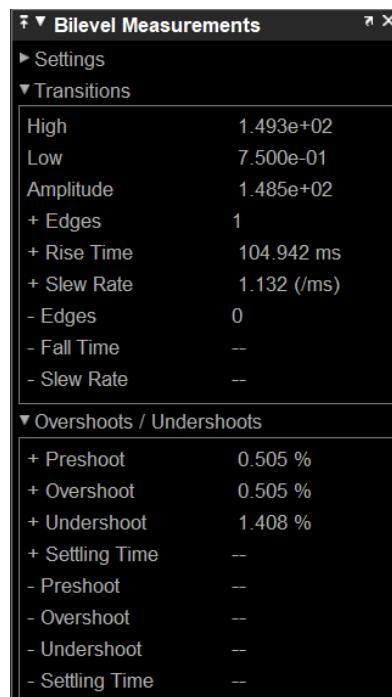
Η απόκριση του συστήματος δεν είναι καλή.

Όμως, μετά το tuning με τα τελικά κέρδη παρατηρούμε ότι η απόκριση είναι αρκετά καλύτερη.

Η έξοδος ακολουθεί τη είσοδο πλέον με ικανοποιητικό ρυθμό.



Σχήμα 24: Σενάριο 1 - Απόκριση - Τελικά κέρδη



Σχήμα 25: Σενάριο 1 – Τελικά κέρδη - Stats

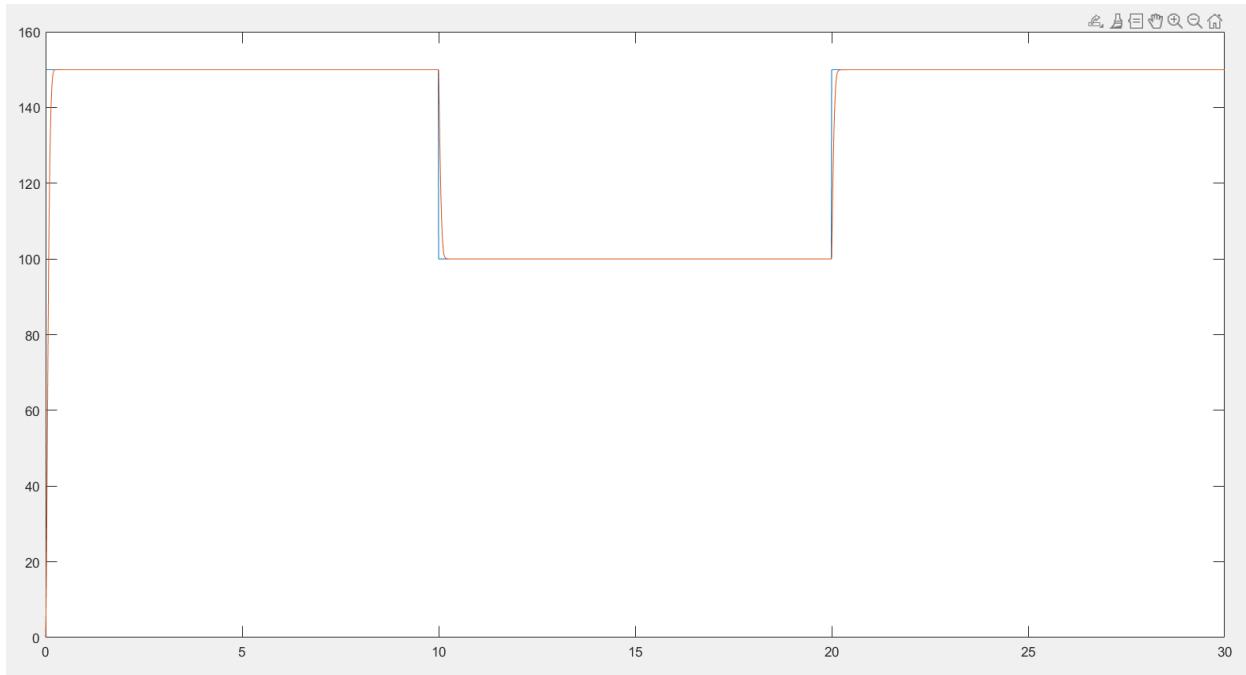
rise time = 0.104 < 0.160 sec

overshoot = 0.5% < 5%

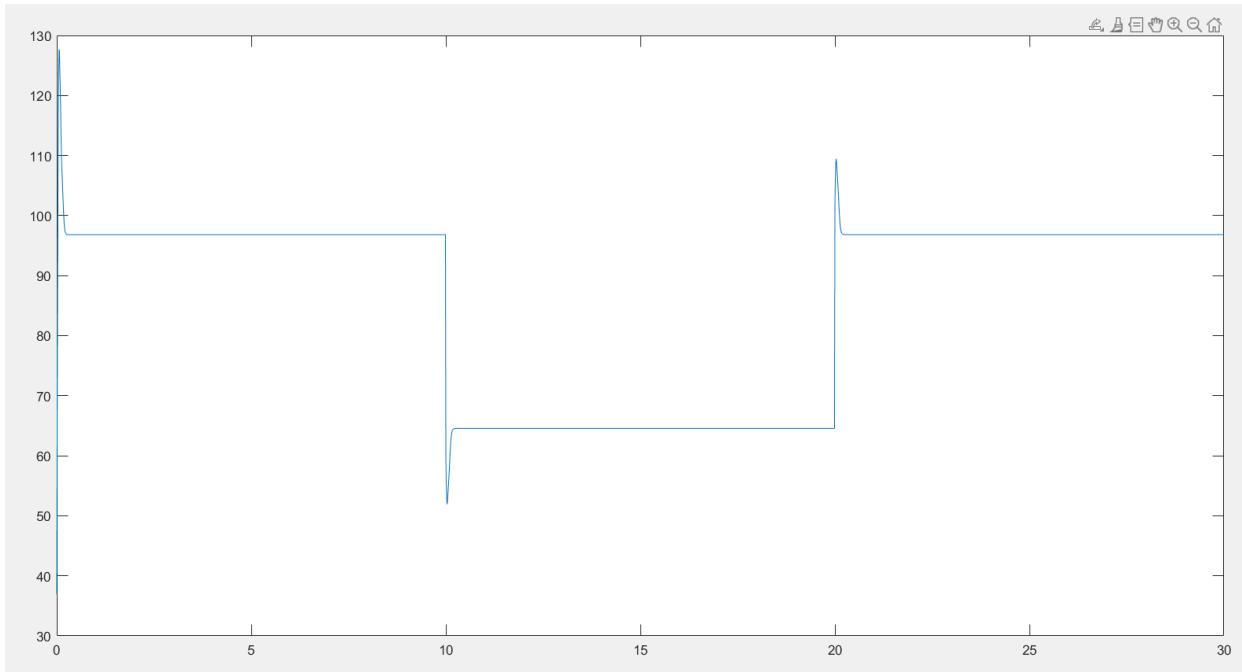
Πιάνουμε τις προδιαγραφές.

Η έξοδος ακολουθεί πιστά την είσοδο.

Και βάζοντας τις τιμές των σημάτων στο matlab με δειγματοληψία  $T=0.01$ , βλέπουμε ότι επαληθεύονται τα αποτελέσματα μας.



Σχήμα 26: Σενάριο 1 - Τελικα κέρδη - Απόκριση Συστήματος - Matlab



Σχήμα 27: Σενάριο 1 - Τελικα κέρδη - Έξοδος Ελεγκτή - Matlab

Παρατηρούμε πάλι ότι η ρύθμιση των κερδών επιτάχυνε σε πολύ μεγάλο βαθμό την απόκριση του συστήματος, ενώ ικανοποιούνται και οι υπόλοιπες προδιαγραφές.  
Η υπερύψωση είναι  $<5\%$ , ο χρόνος ανόδου είναι  $< 0.16 \text{ sec}$  και το σφάλμα μόνιμης κατάστασης είναι μηδεν, ενώ η έξοδος του ελεγκτή Να παραμένει κάτω από την τιμή 200V.

#### 1.3.1.1 Διέγερση Κανόνων

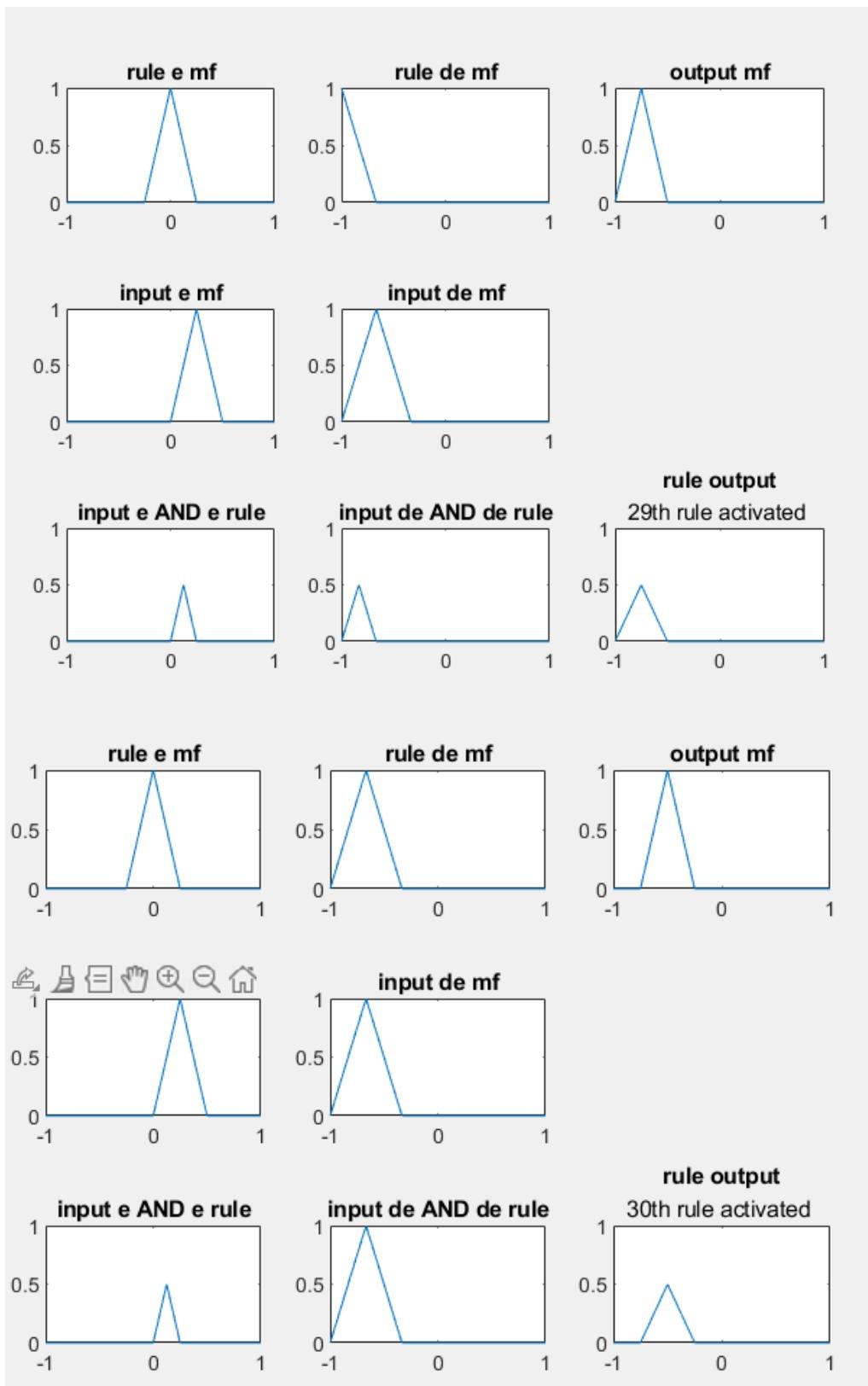
Θεωρούμε διέγερση e is PS και de is NM.

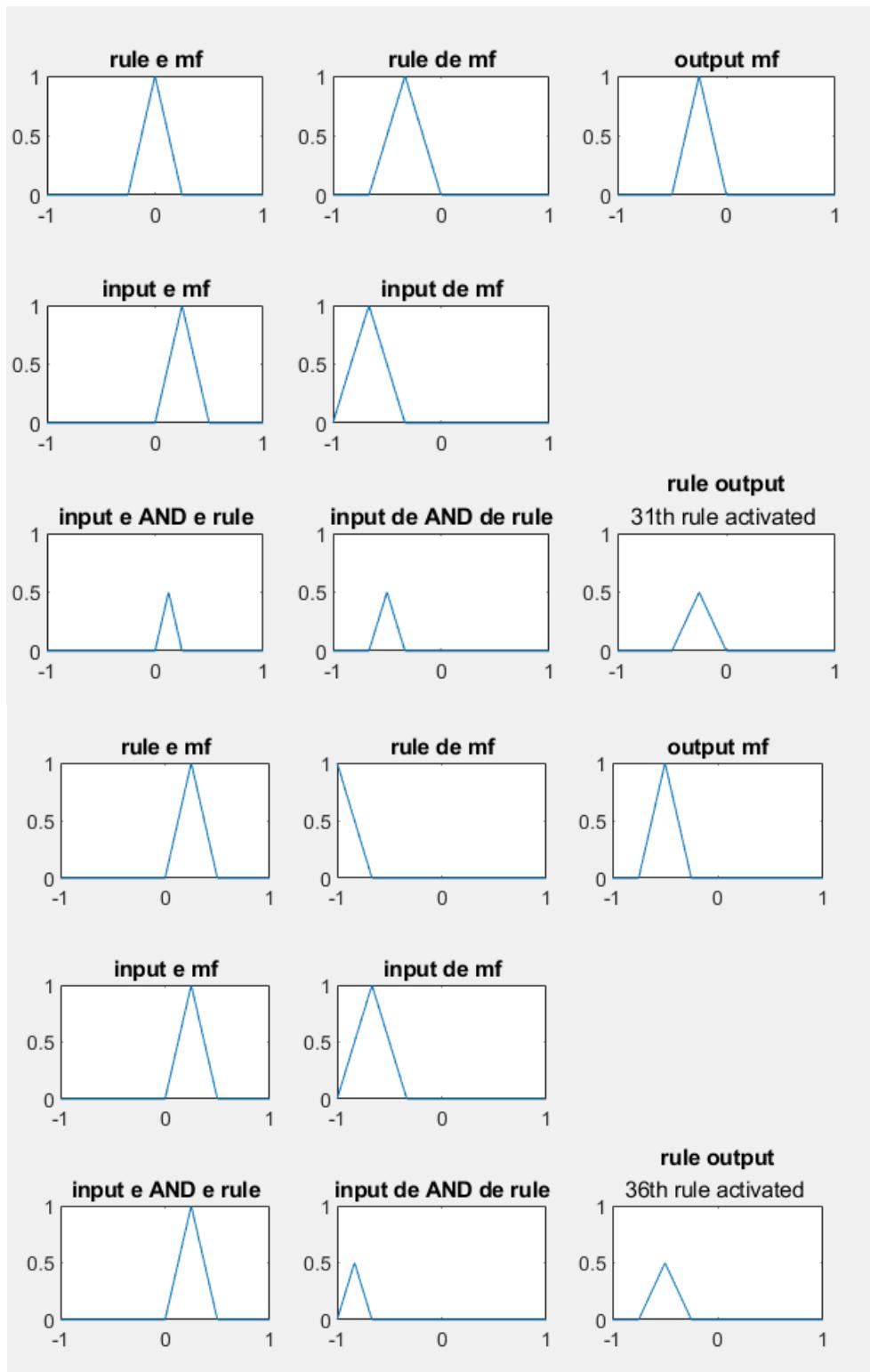
Θα δούμε ποιοί κανόνες διεγείρονται και θα τους αναπαραστήσουμε γραφικά.

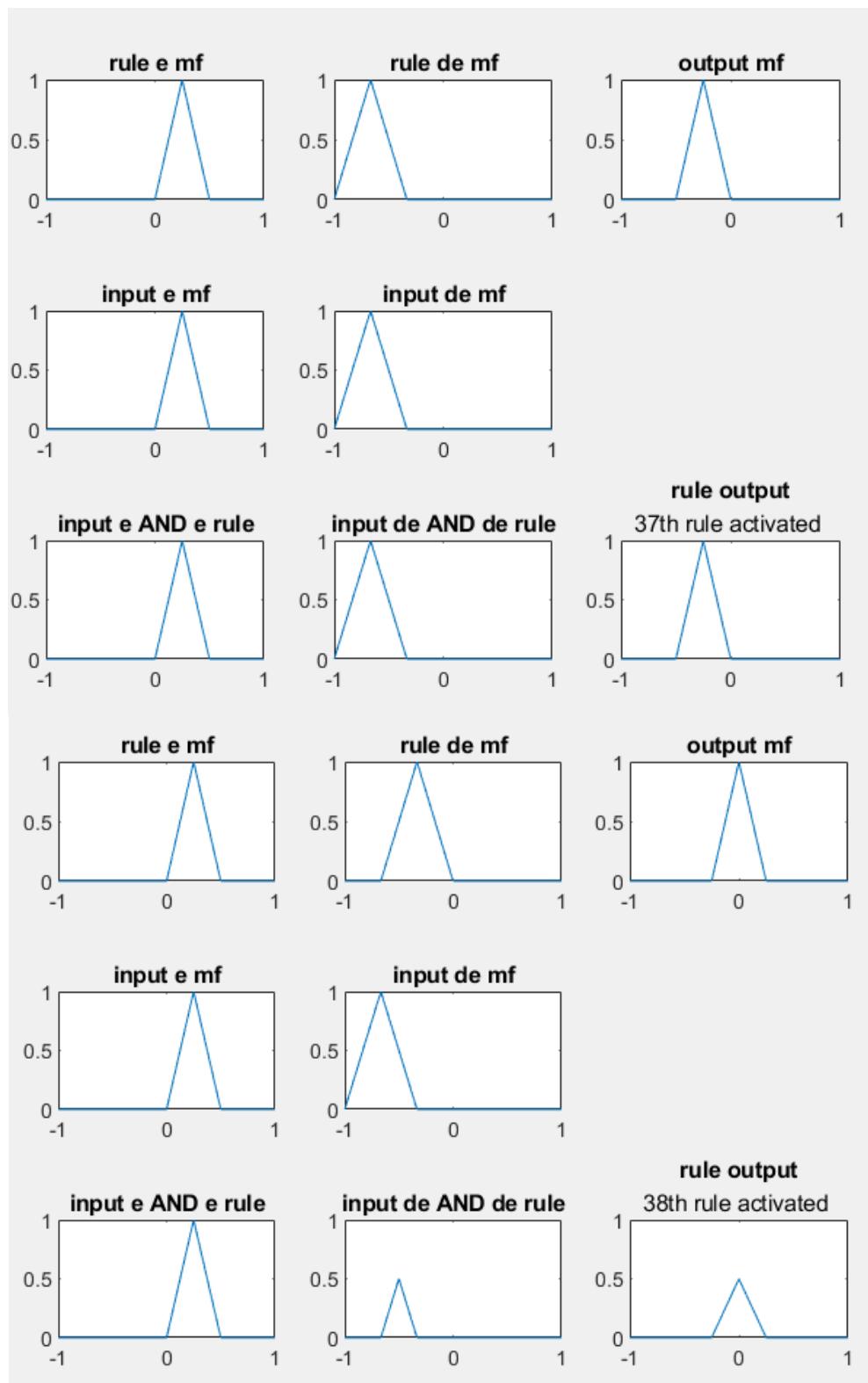
Θα χρησιμοποιήσουμε απο-ασαφοποιητή COA.

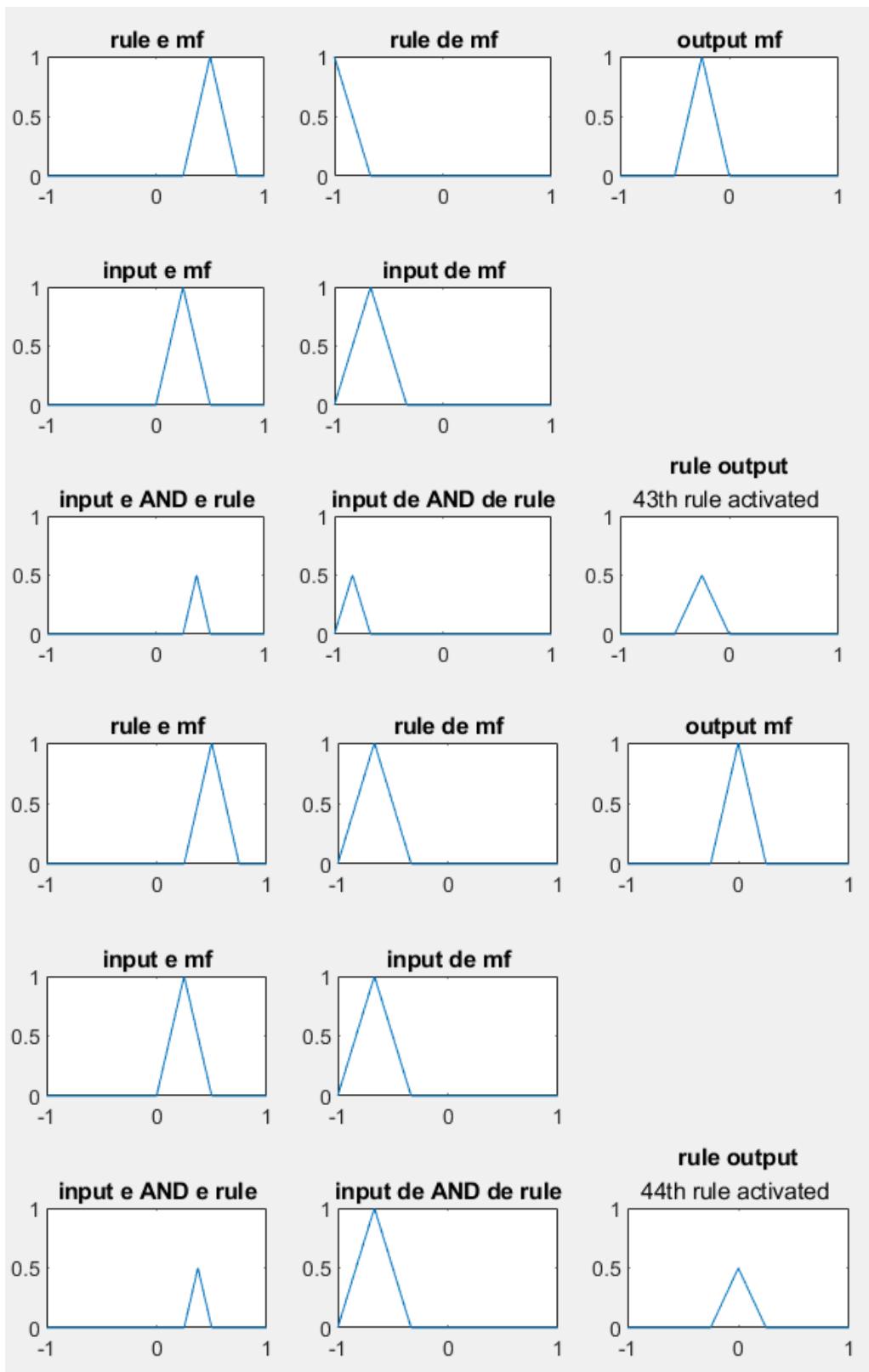
Οι κανόνες που διεγείρονται είναι οι: 29,30,31,36,37,38,43,44,45.

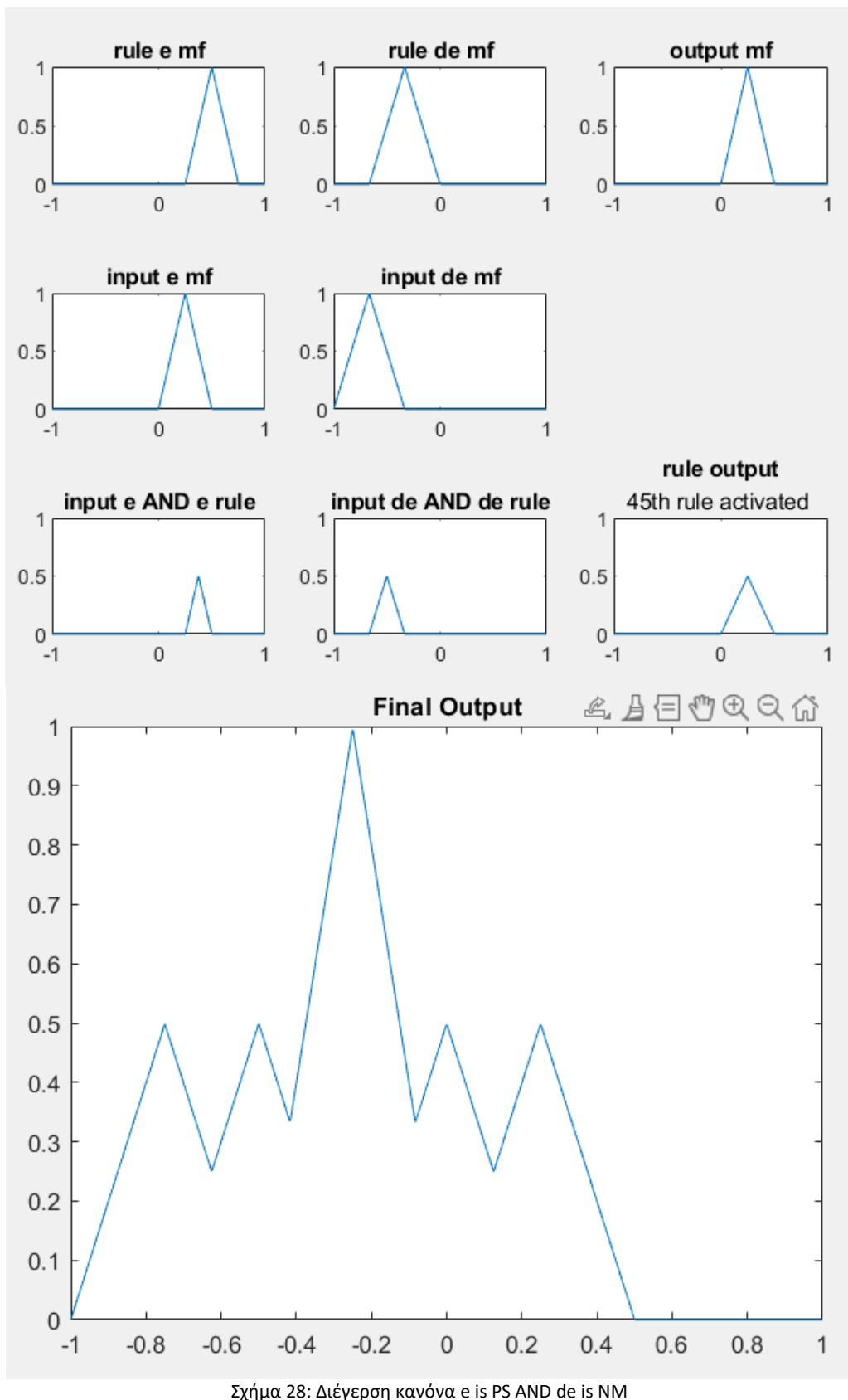
Με βάση το script “dc\_motor.m” πάιρνουμε τις εξής γραδικές παραστάσεις.







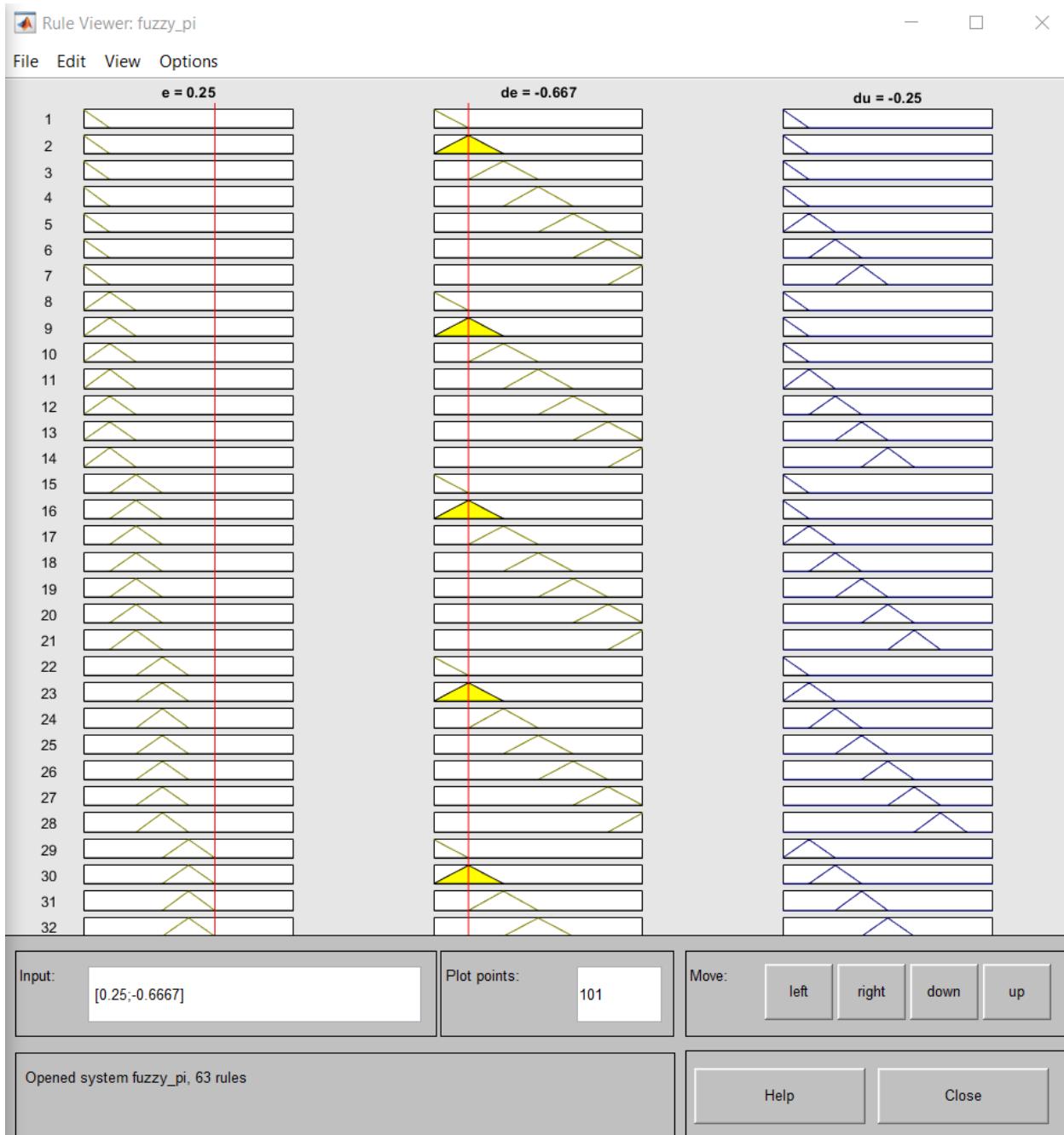


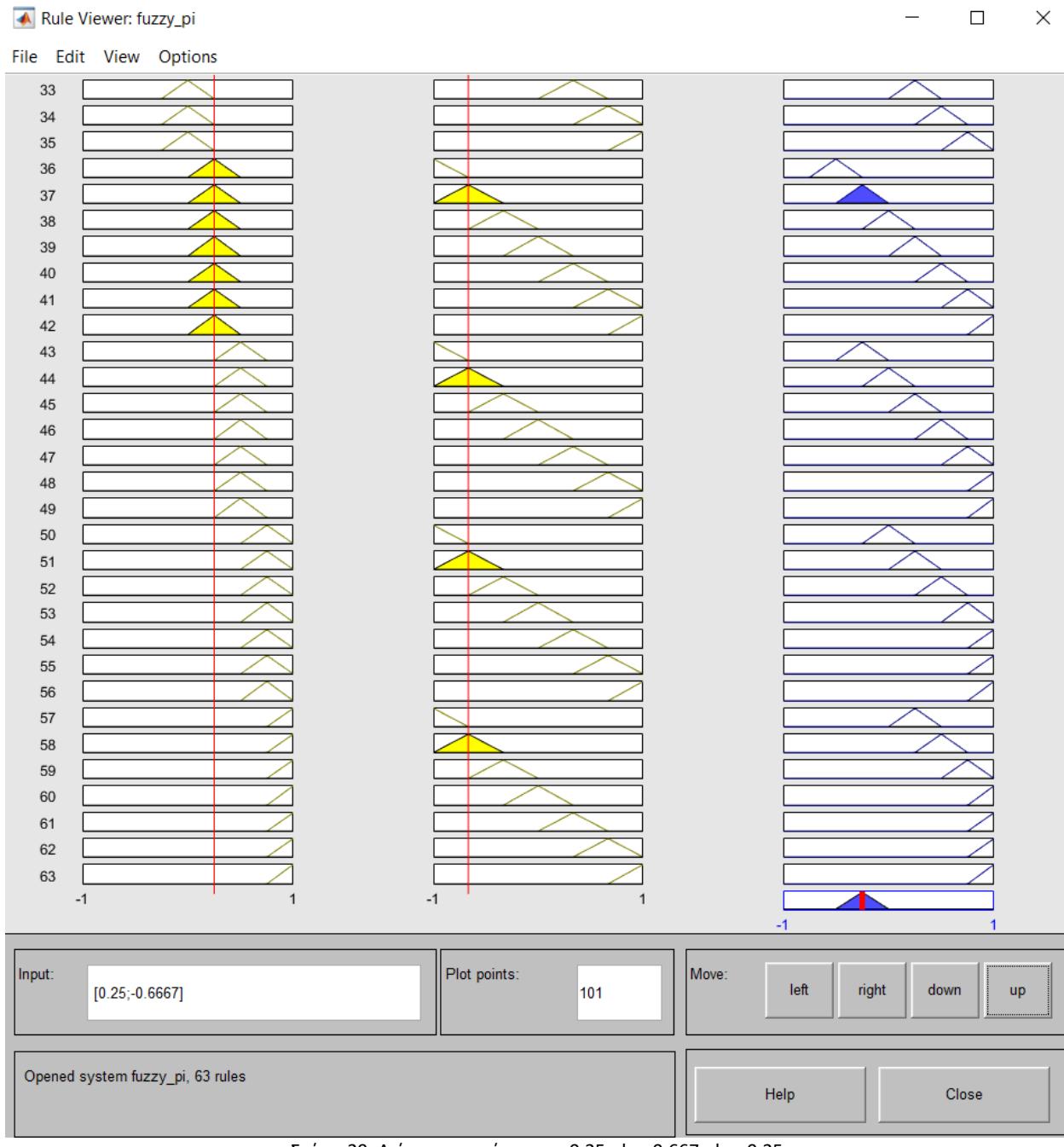


Η τελική έξοδος έχει peak στο -0.25 δηλαδή στο NS όπως περιμέναμε.

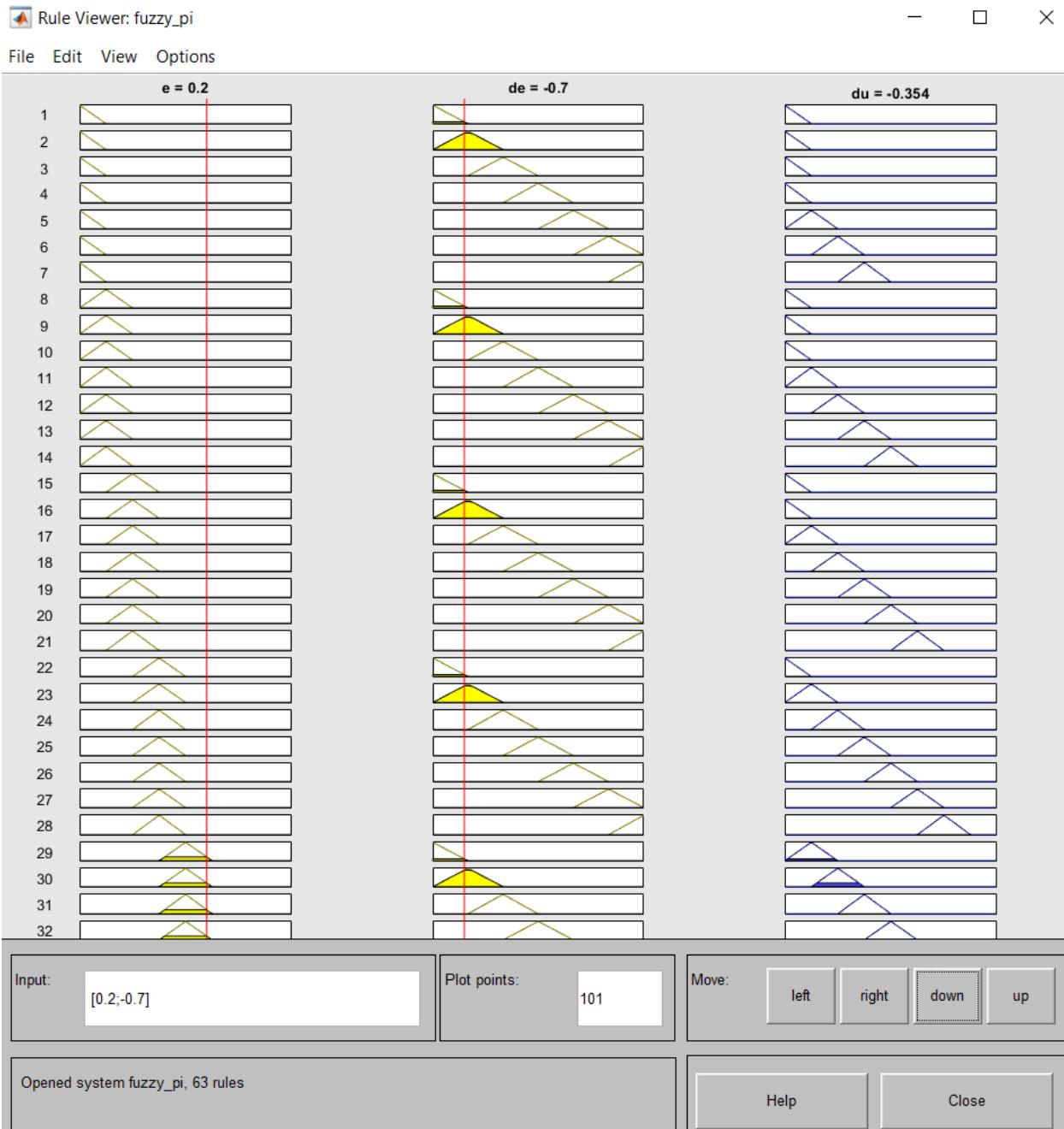
Επίσης, μπορούμε μέσω την εντολής “fuzzyLogicDesigner(‘fuzzy\_pi.fis’)\”, να δώσουμε διάφορες τιμές στα inputs e και de και να δούμε γραφικά ποιοί κανόνες διεγείρονται.

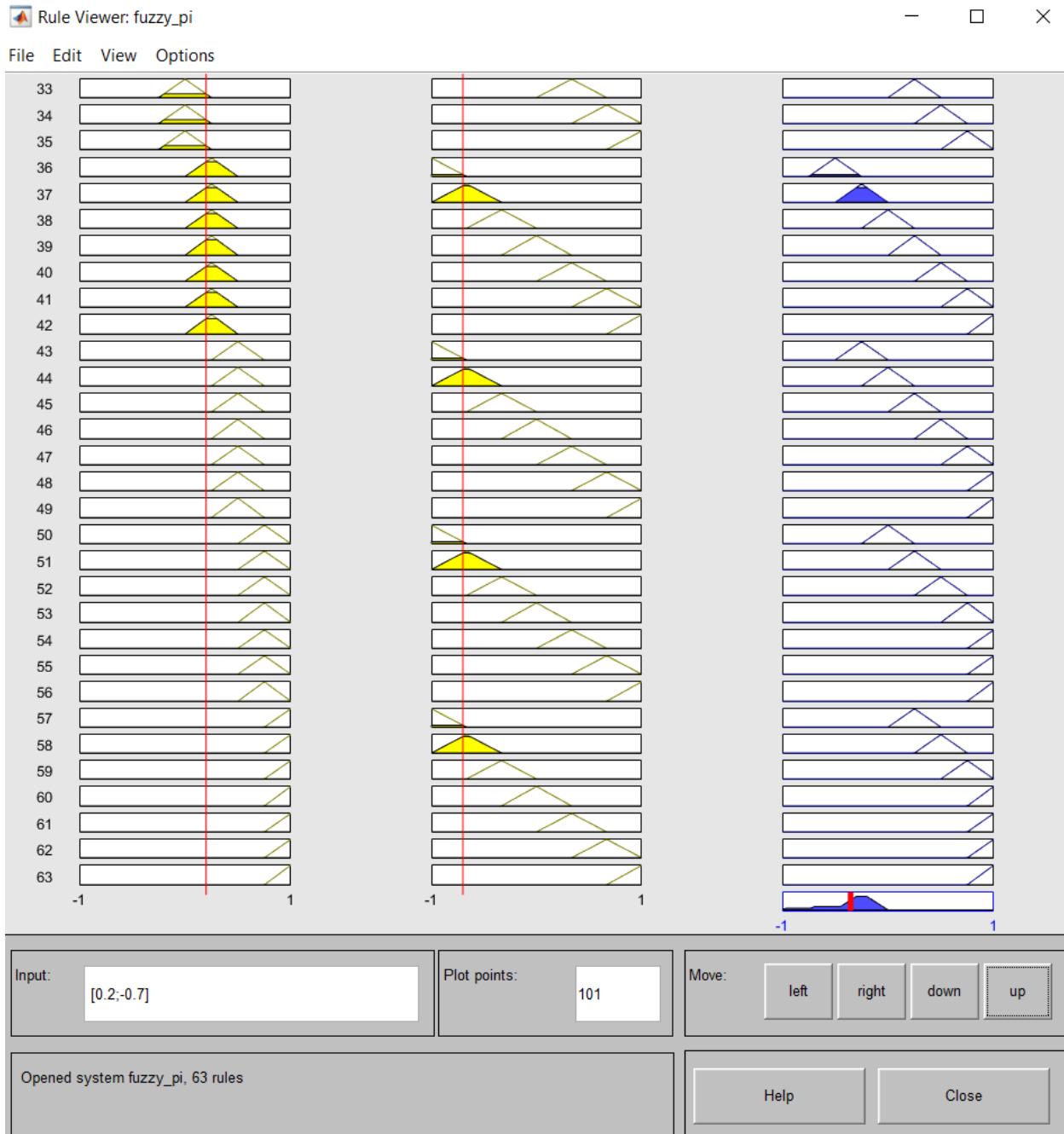
Για  $e=0.25$  και  $de=-0.667$  έχουμε  $du=-0.25$ :





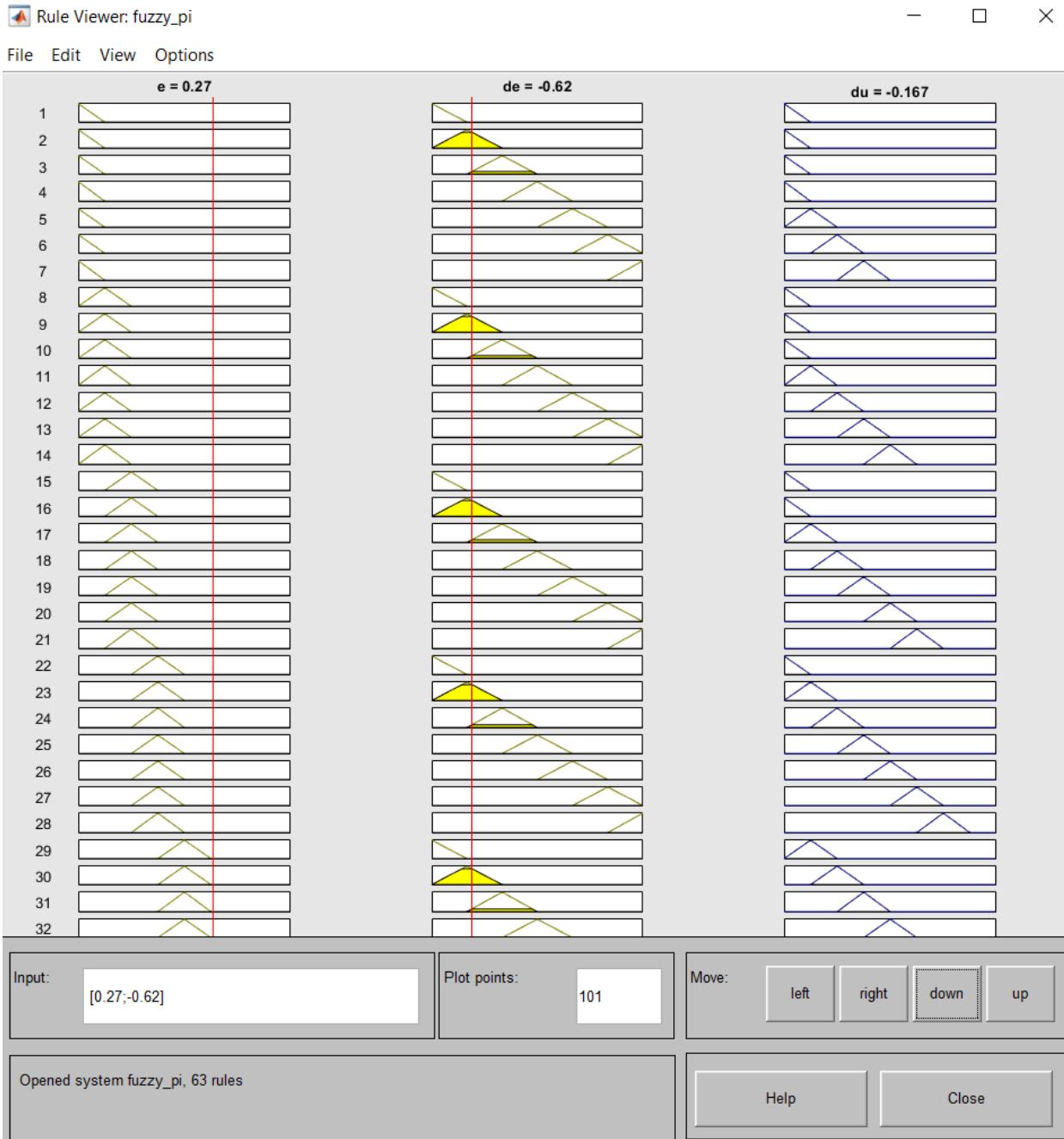
Για  $e=0.20$  και  $de=-0.70$  έχουμε  $du=-0.354$ :

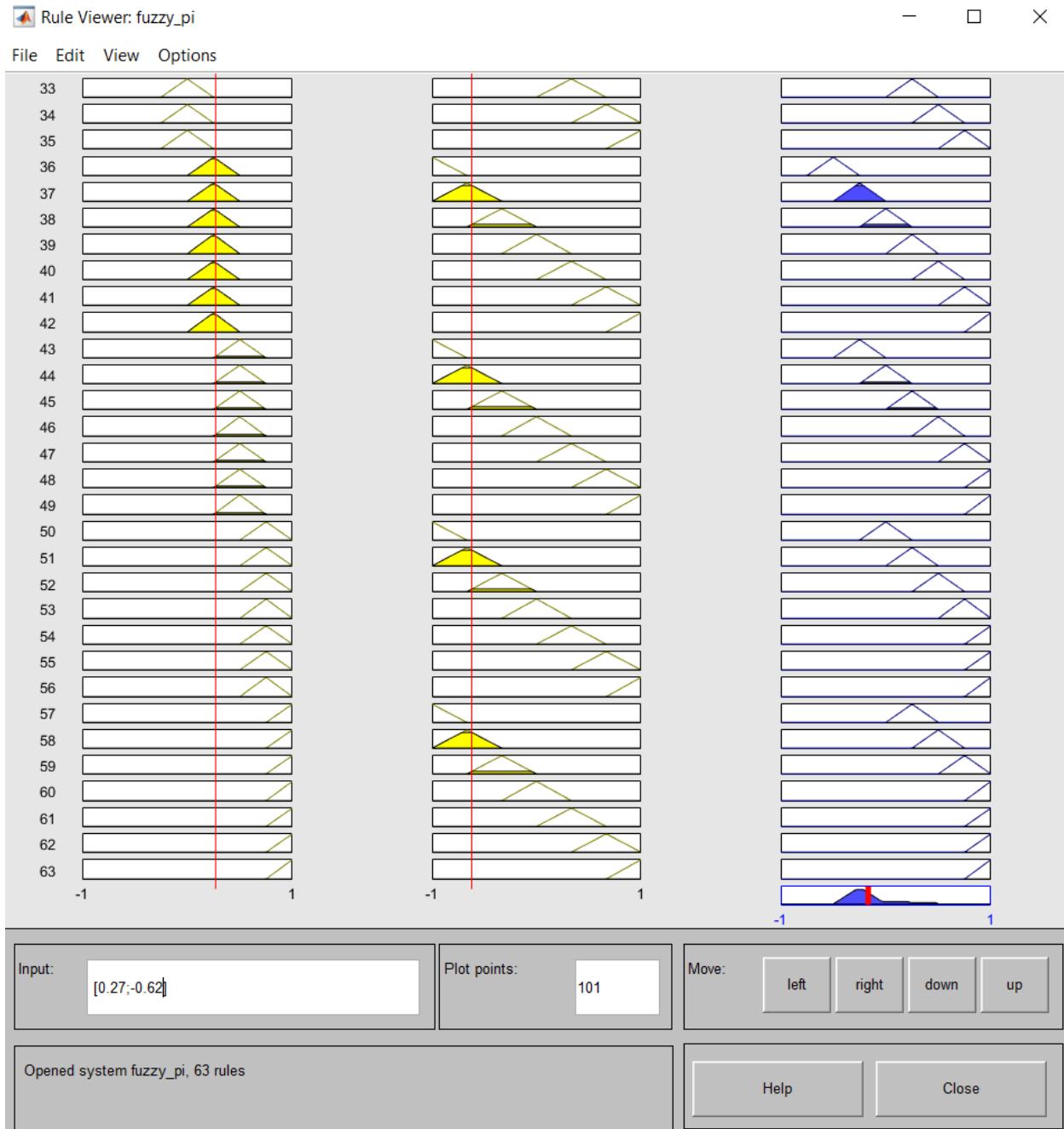




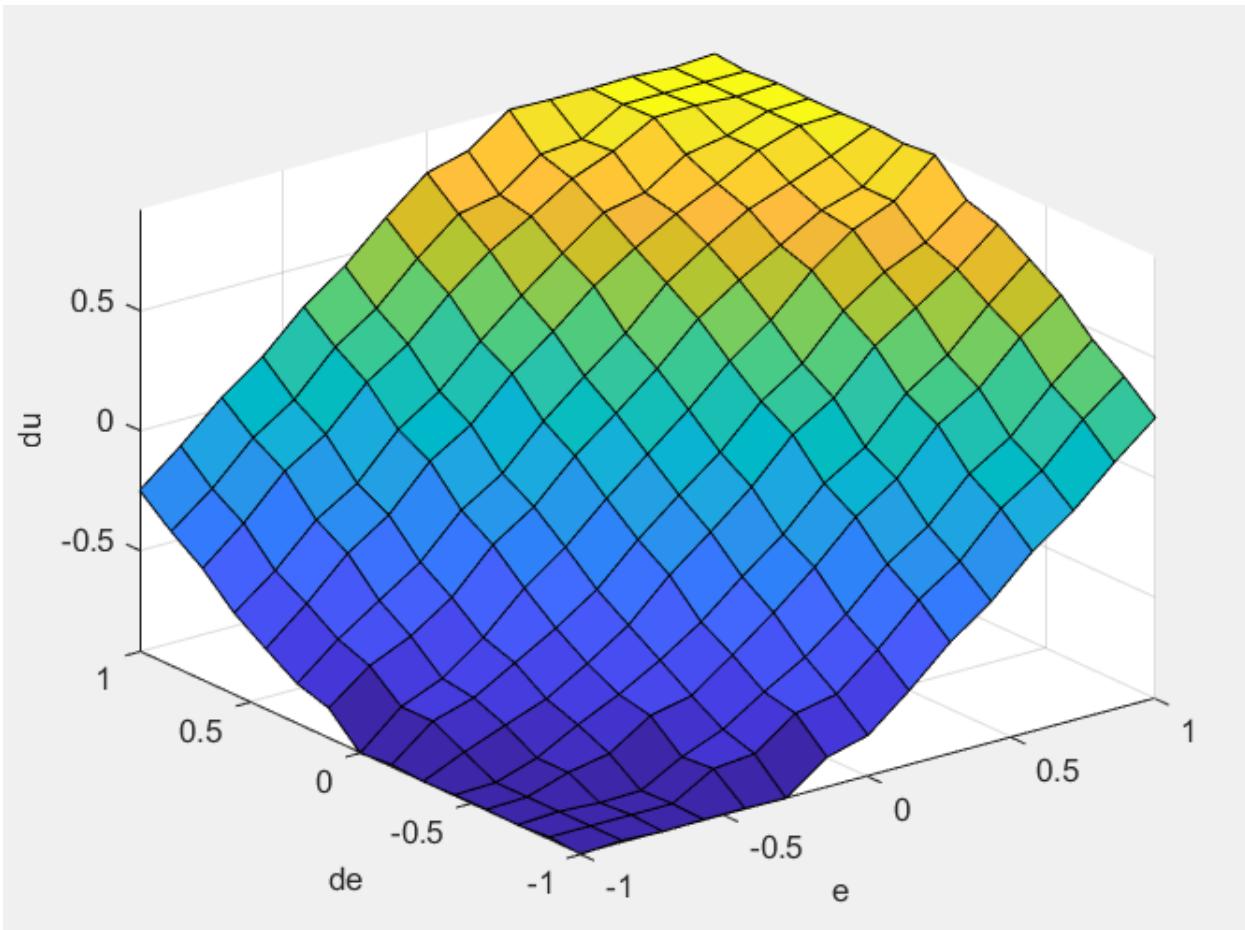
Σχήμα 30: Διέγερση κανόνων -  $e=0.20$ ,  $de=-70$ ,  $du=-0.354$

Για  $e=0.27$  και  $de=-0.62$  έχουμε  $du=-0.167$ :





Παρακάτω φαίνεται η τρισδιάστατη επιφάνεια εξόδου του ασαφούς ελεγκτή.



Σχήμα 32: gensurf

Η παραπάνω επιφάνεια είναι η τρισδιάστατη απεικόνιση της βάσης κανόνων.

Η τιμή αυξάνεται ή μειώνεται όταν απομακρυνόμαστε από την διαγώνιο.

Από την επιφάνεια της εξόδου βλέπουμε ότι στις περιπτώσεις όπου το σφάλμα είναι θετικό και η παράγωγος του θετική η έξοδος είναι θετική, ώστε να αυξηθεί η έξοδος του συστήματος και να πλησιάσει την επιθυμητή είσοδο αναφοράς.

Στην περίπτωση του αρνητικού σφάλματος και αρνητικής παραγώγου του σφάλματος η έξοδος είναι αρνητική, ώστε να μειωθεί η έξοδος του συστήματος και να πλησιάσει την είσοδο αναφοράς.

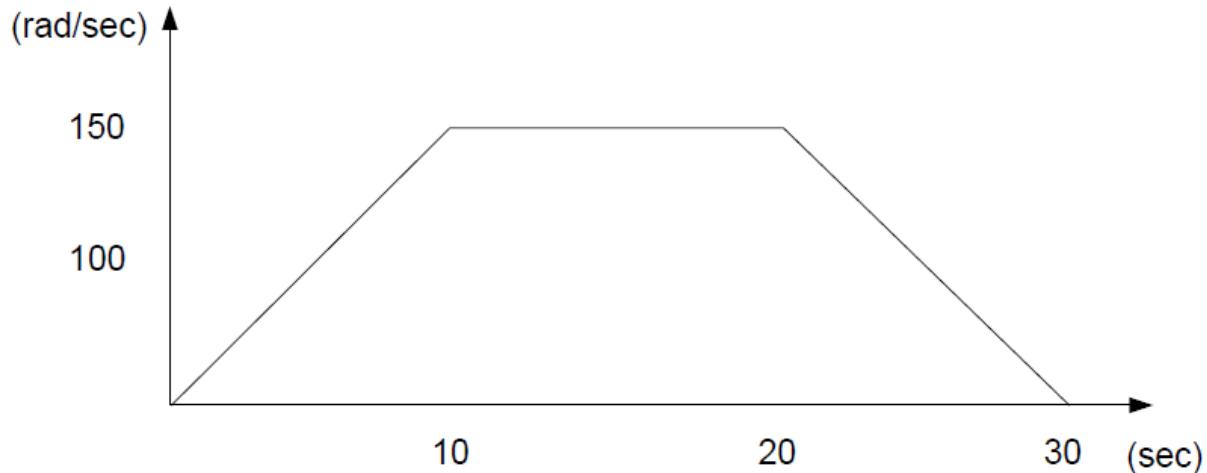
Για θετικό σφάλμα  $e$  και αρνητικό  $de$  η έξοδος κινείται κοντά στο 0, καθώς έχει σωστή συμπεριφορά το σύστημα, αφού το σφάλμα οδηγείται στο 0 και κατά συνέπεια θέλουμε να διατηρήσουμε αυτή την συμπεριφορά ή να την προσαρμόσουμε για να αποφύγουμε υπερυψώσεις. Το ίδιο ισχύει και στην περίπτωση του αρνητικού σφάλματος και της θετικής παραγώγου του, ωστόσο στην περίπτωση της προσαρμογής της εξόδου θέλουμε να αποφύγουμε τις μεγάλες βυθίσεις.

Όταν το σφάλμα είναι κοντά στο 0 και η παράγωγος κοντά στο 0, τότε θέλουμε η έξοδος να είναι μηδενική και να διατηρήσει αυτήν την κατάσταση.

### 1.3.2 Σενάριο 2

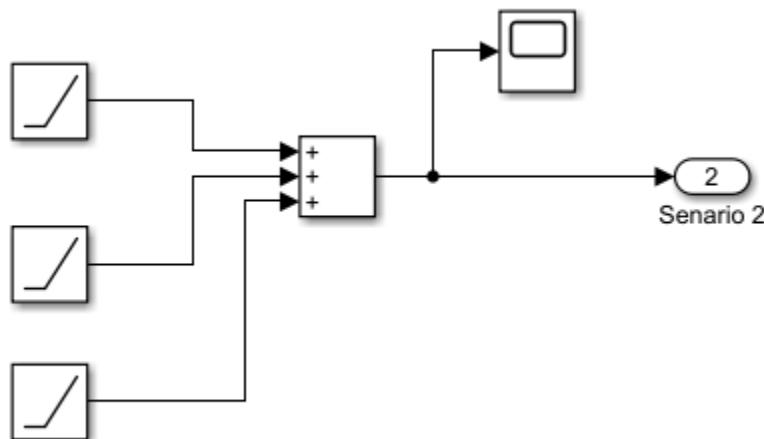
Το σύστημα υλοποιείται με βάση το μοντέλο του Σχήματος 12.

Δίνουμε ως είσοδο στο μοντέλο το σήμα του Σχήματος 33.



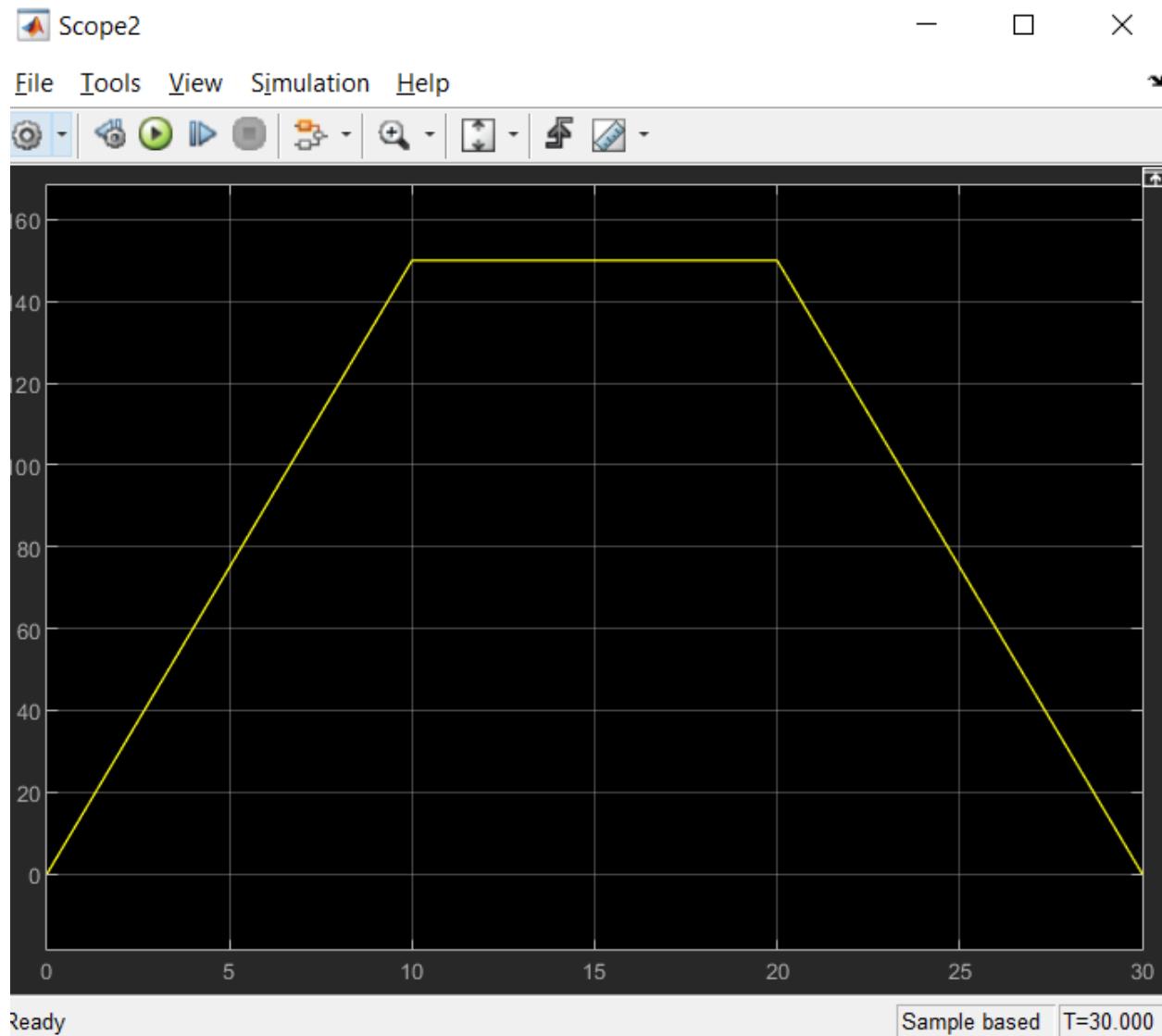
Σχήμα 33: Σενάριο 2 – Είσοδος

Κατασκευάζουμε το σήμα εισόδου στο Simulink.



Σχήμα 34: Σενάριο 2 – Είσοδος Simulink Μοντέλο

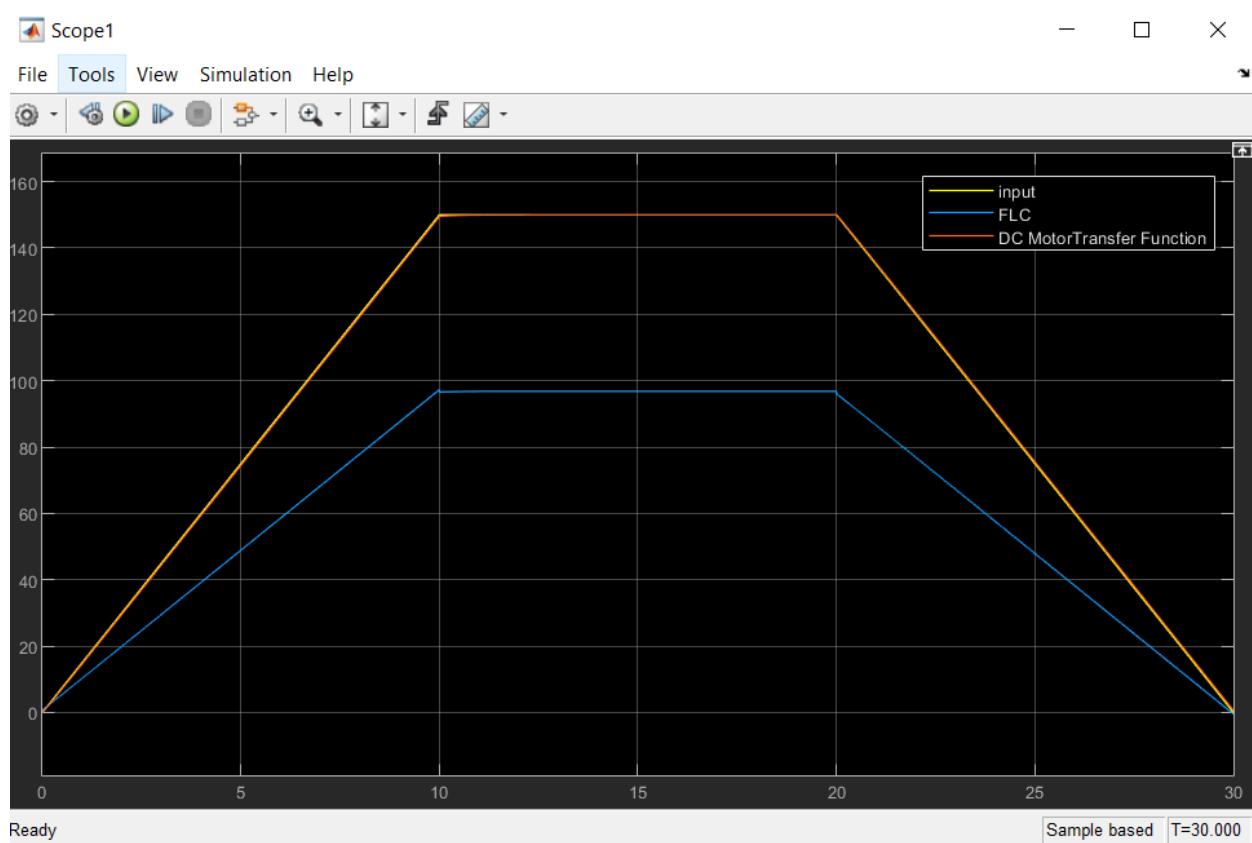
Πατατηρούμε ότι η είσοδος που δημιουργούμε είναι η επιθυμητή.



Σχήμα 35: Σενάριο 2 – Είσοδος Simulink

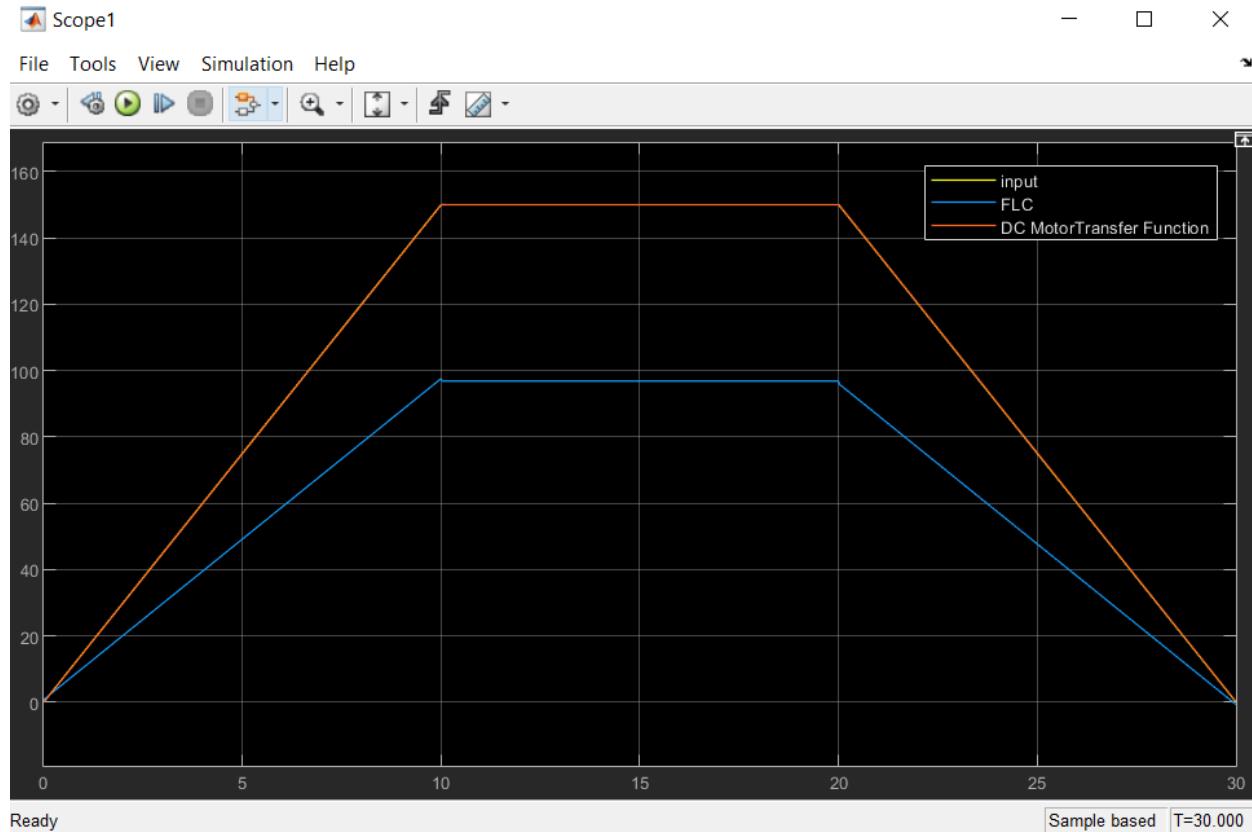
Θέλουμε η έξοδος να ακολουθεί πιστά το σήμα εισόδου.

Αρχικά, βλέπουμε την απόκριση με τα αρχικά κέρδη.



Σχήμα 36: Σενάριο 2 - Απόκριση - Αρχικά κέρδη

Και στη συνέχεια βλέπουμε ξανά την απόκριση με τα τελικά κέρδη.



Σχήμα 37: Σενάριο 2 - Απόκριση - Τελικά κέρδη

▼ Overshoots / Undershoots	
+ Preshoot	0.505 %
+ Overshoot	0.505 %
+ Undershoot	1.980 %
+ Settling Time	--
- Preshoot	--
- Overshoot	--
- Undershoot	--
- Settling Time	--

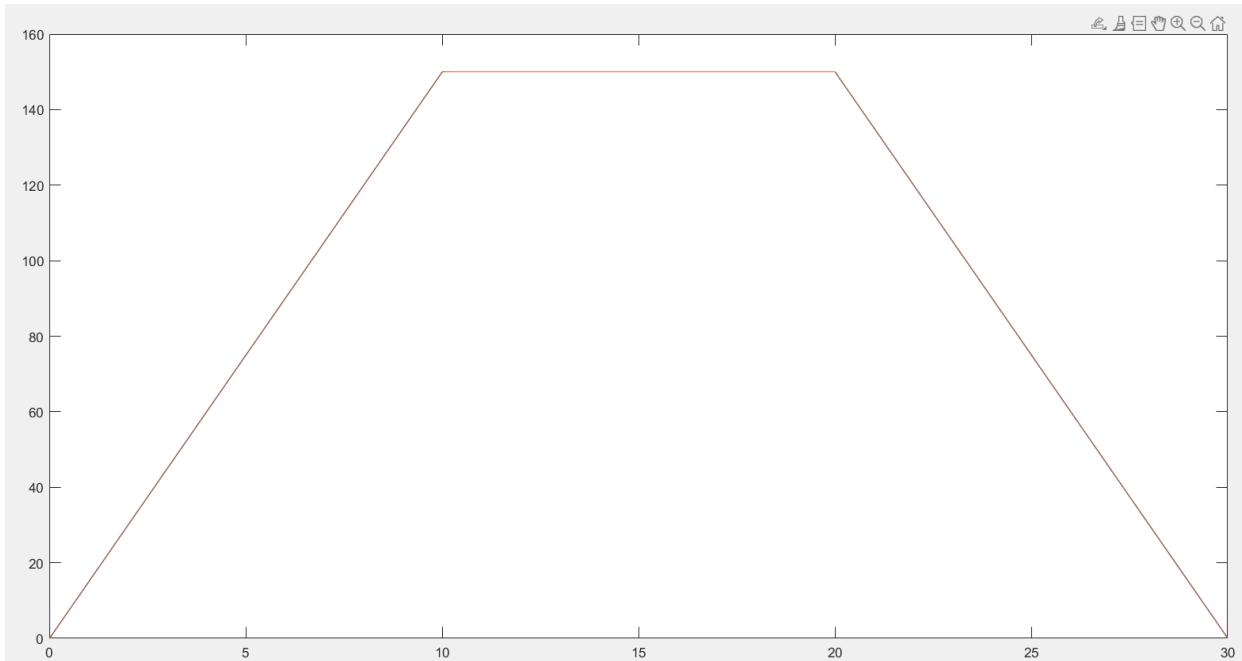
Σχήμα 38: Σενάριο 2 – Τελικά κέρδη - Stats

$$\text{overshoot} = 0.5\% < 5\%$$

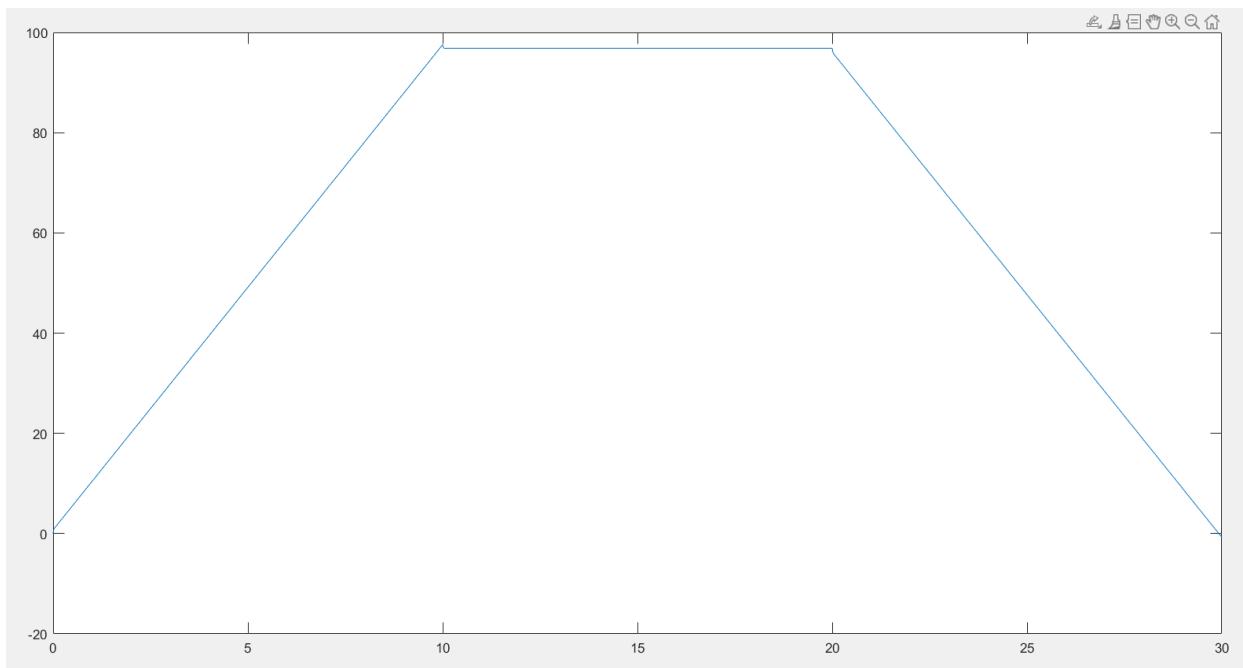
Πιάνουμε τις προδιαγραφές.

Η έξοδος ακολουθεί πιστά την είσοδο.

Και βάζοντας τις τιμές των σημάτων στο matlab με δειγματοληψία  $T=0.01$ , βλέπουμε ότι επαληθεύονται τα αποτελέσματα μας.

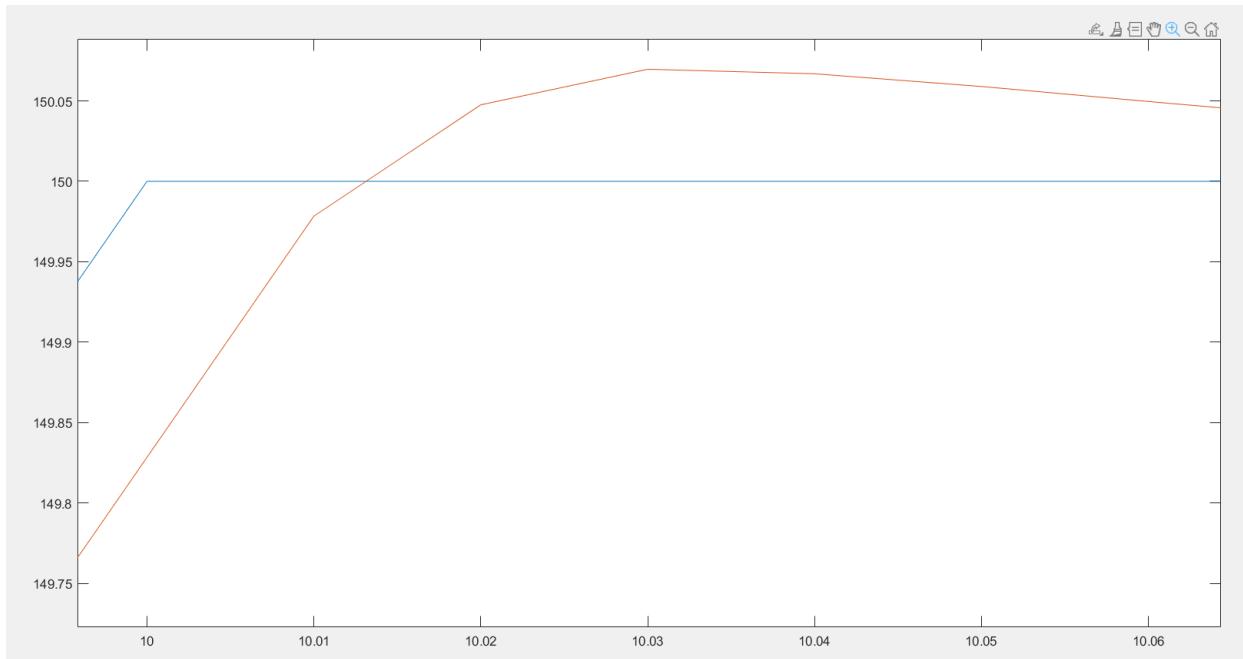


Σχήμα 39: Σενάριο 2 - Τελικά κέρδη – Απόκριση Συστήματος - Matlab



Σχήμα 40: Σενάριο 2 - Τελικα κέρδη - Έξοδος Ελεγκτή - Matlab

Κάνοντας zoom-in παρατηρούμε ότι το σφάλμα είναι της τάξης του 0.05 rad/sec.



Σχήμα 41: Σενάριο 2 - Τελικα κέρδη – Απόκριση Συστήματος- Zoom in - Matlab

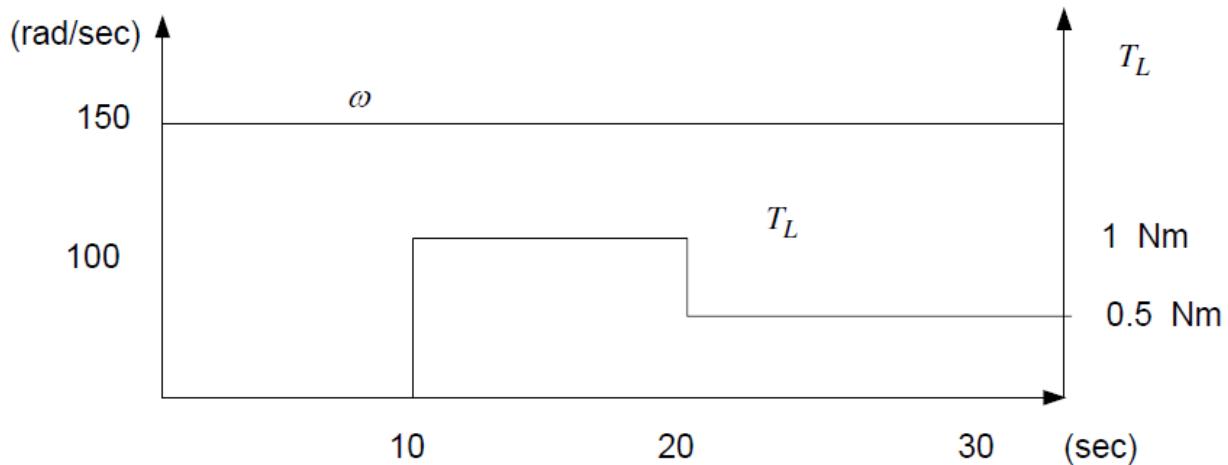
Παρατηρούμε ότι τα σφάλματα είναι της τάξης του 0.05 rad/sec στα χρονικά διαστήματα όπου η επιτάχυνση είναι μη μηδενική, ενώ όταν η γωνιακή ταχύτητα είναι σταθερή το σφάλμα είναι μηδέν.

### 1.3.3 Σενάριο 3

Το σύστημα υλοποιείται με βάση το μοντέλο του Σχήματος 11.

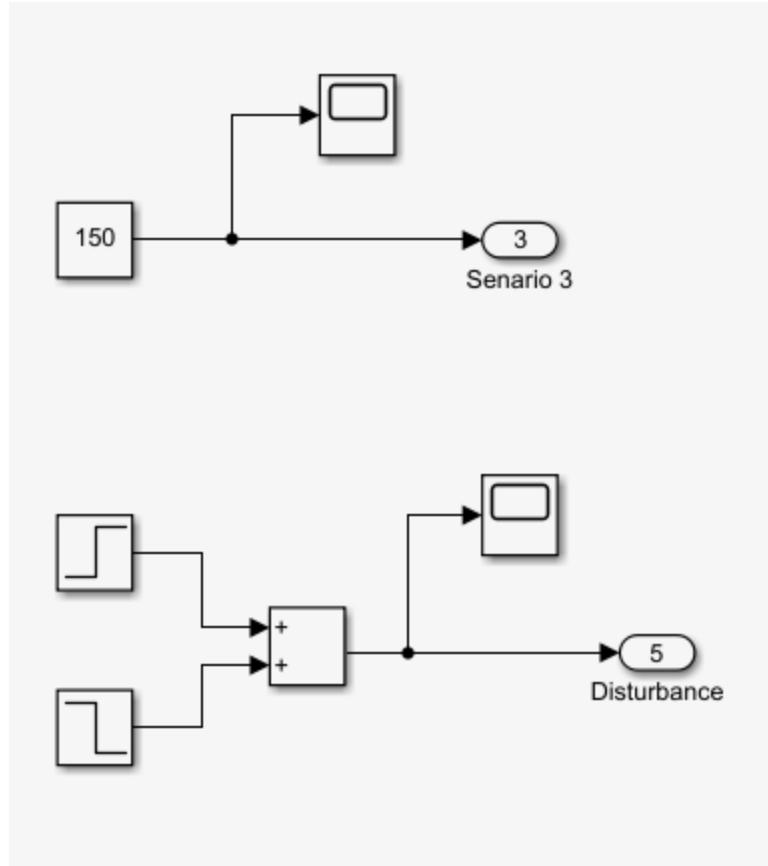
Σε αυτό το σενάριο θα εξετάσουμε την συμπεριφορά του συστήματος όταν υπάρχουν διαταραχές, με την μορφή ροπής. Υποθέτουμε ότι το σύστημα δουλεύει στις ονομαστικές στροφές, δηλαδή 150rad/sec

Η διαταραχή έχει τη μορφή του Σχήματος 42.



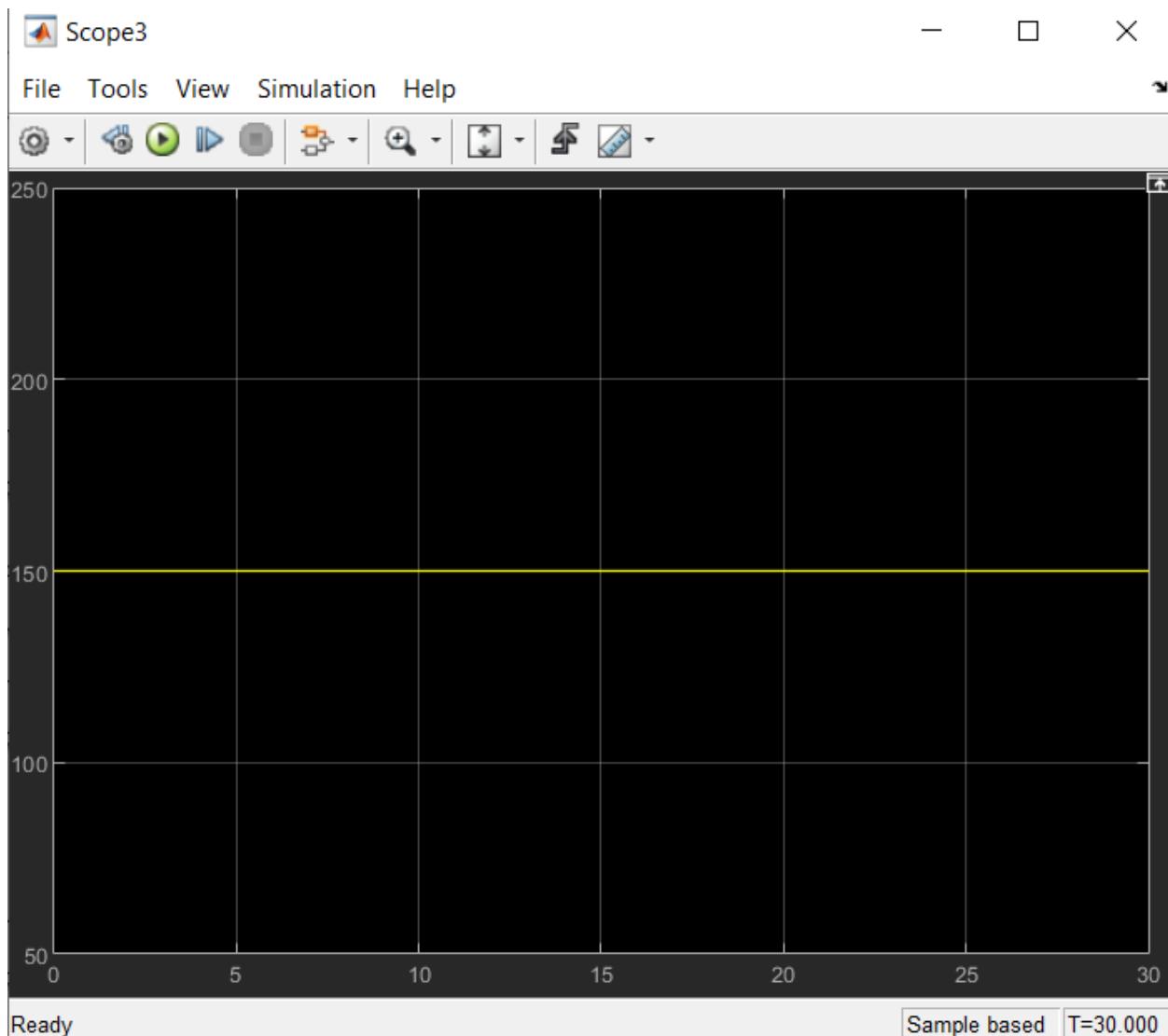
Σχήμα 42: Σενάριο 3 – Είσοδος

Κατασκευάζουμε το σήμα εισόδου και το σήμα διαταραχής στο Simulink.

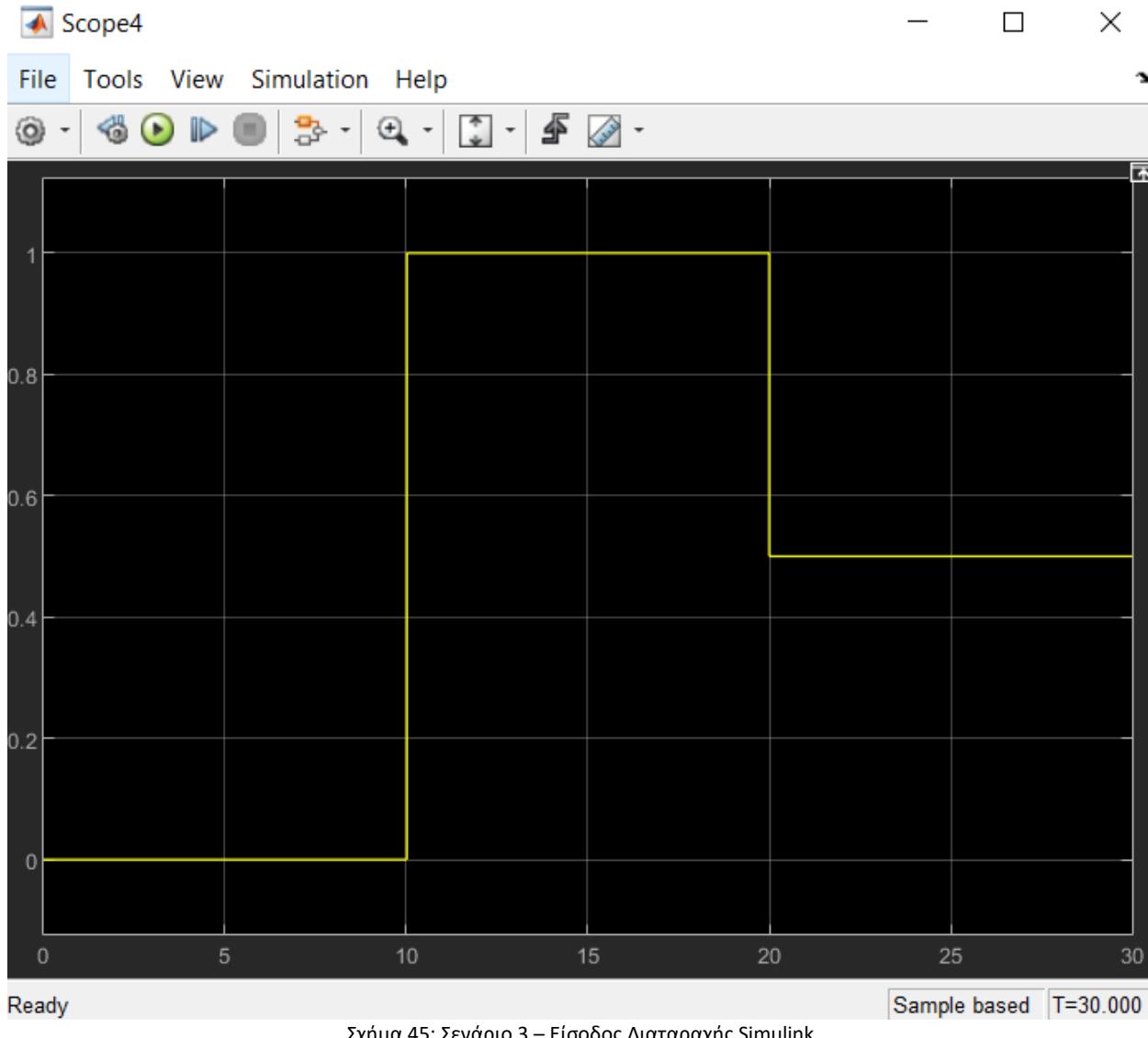


Σχήμα 43: Σενάριο 3 – Είσοδος Simulink Μοντέλο

Πατατηρούμε ότι η είσοδος που δημιουργούμε είναι η επιθυμητή.

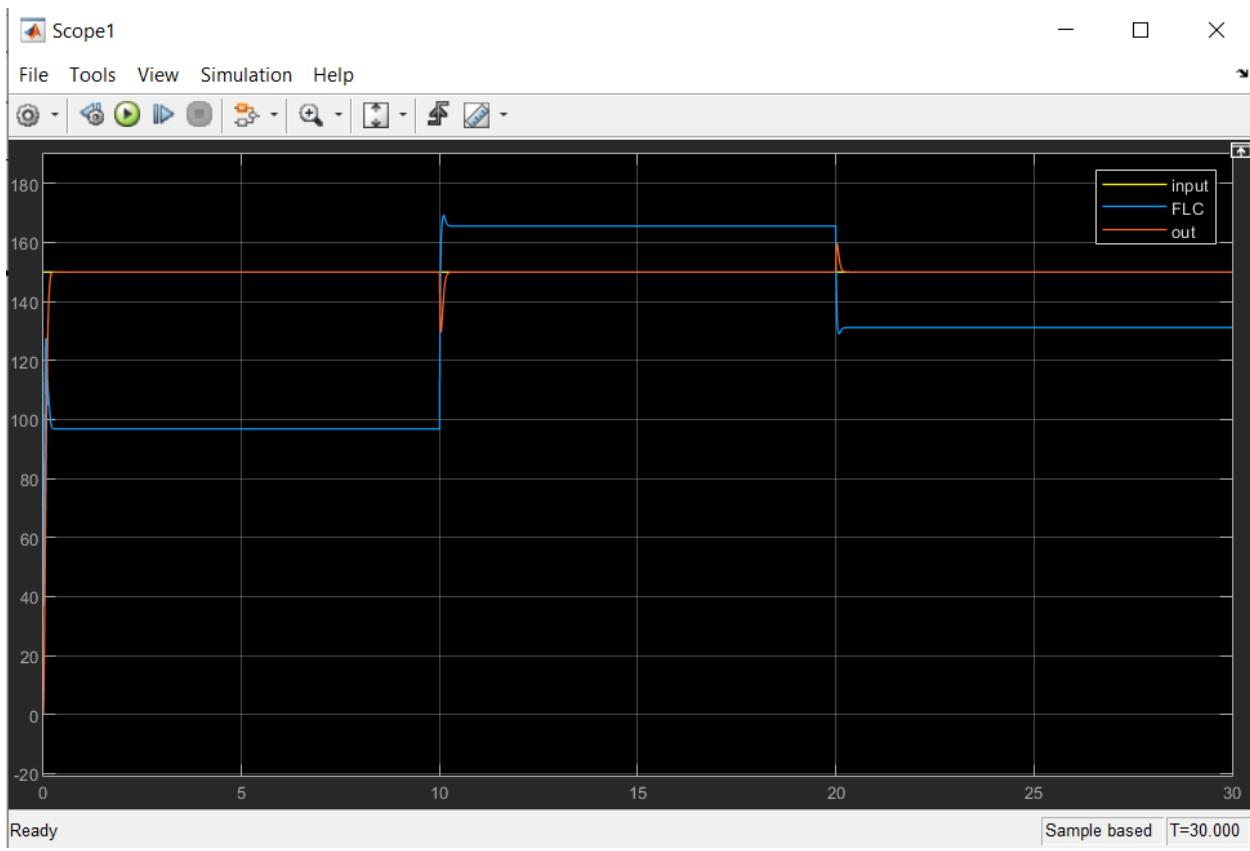


Σχήμα 44: Σενάριο 3 – Είσοδος Simulink

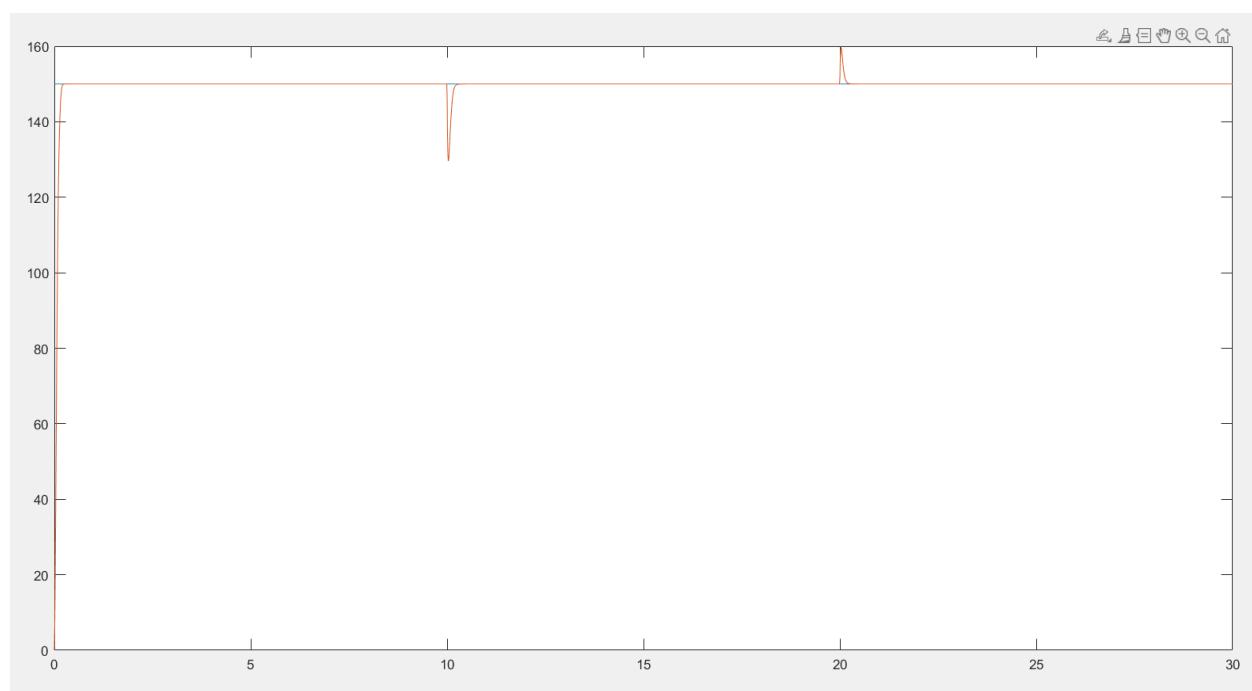
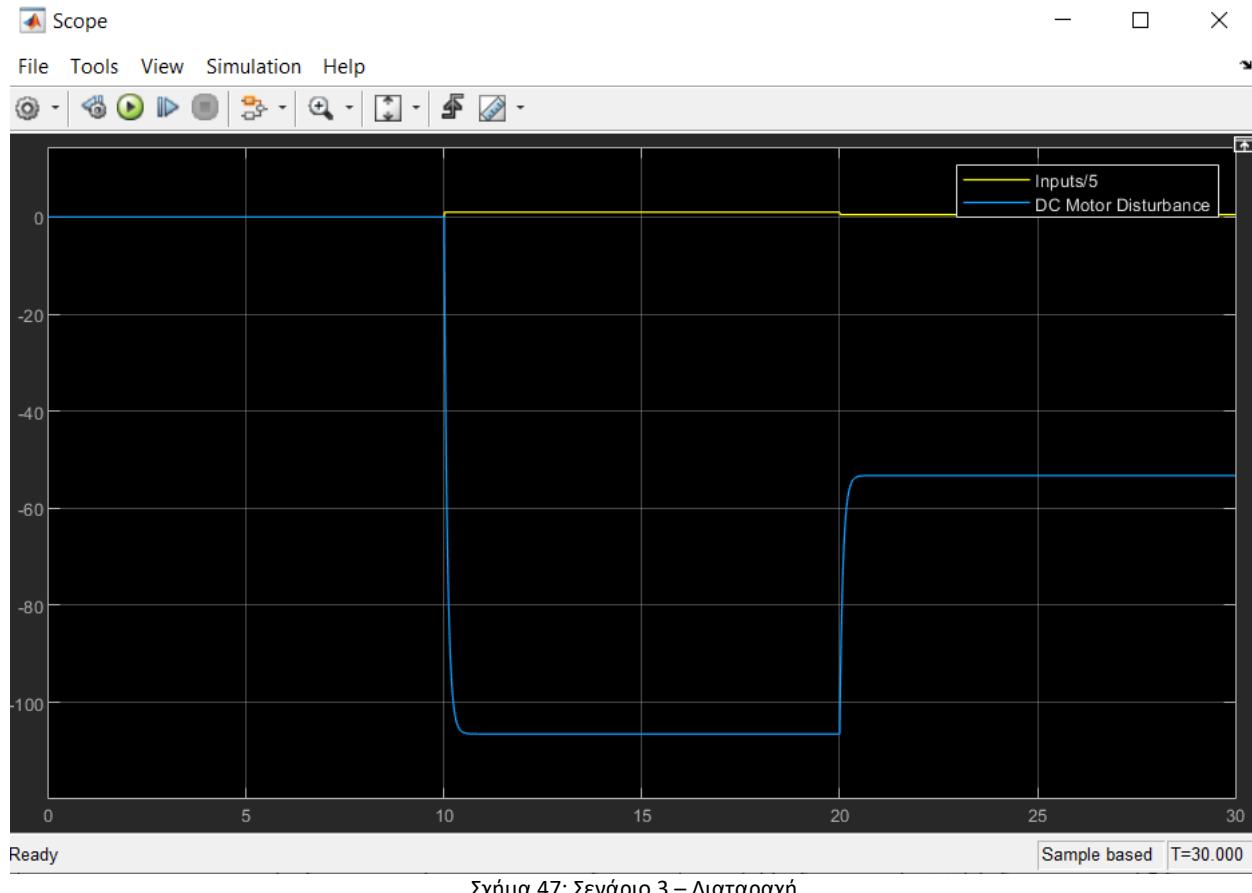


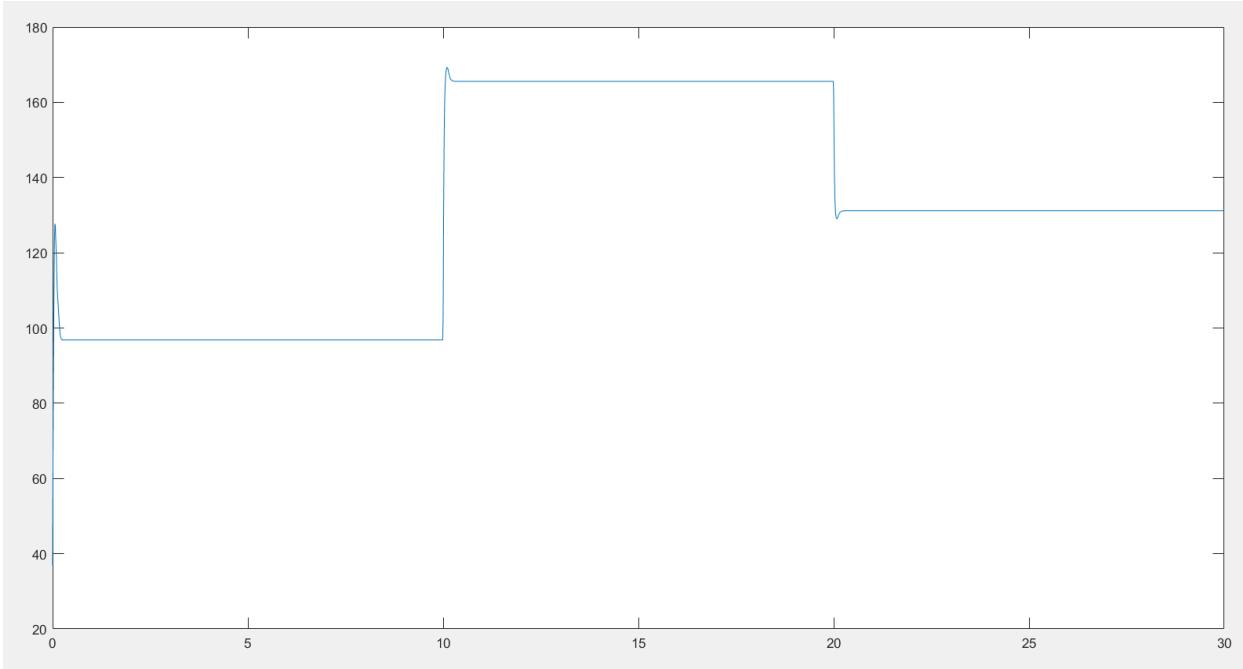
Θέλουμε η έξοδος να ακολουθεί πιστά το σήμα εισόδου, χωρίς να επηρεάζεται από τη διαταραχή.

Μετά το tuning και έχοντας εφαρμόσει τα τελικά κέρδη η απόκριση του συστήματος και η έξοδος του ελεγκτή και της συνάρτησης μεταφοράς των διαταραχών φαίνονται παρακάτω.



Σχήμα 46: Σενάριο 3 – Απόκριση - Τελικά κέρδη





Σχήμα 49: Σενάριο 3 - Τελικα κέρδη - Έξοδος Ελεγκτή - Matlab

Παρατηρούμε ότι παρόλο που το σήμα κάνει spike, λόγω της διαταραχής, επανέρχεται γρήγορα στην κανονική κατάσταση.

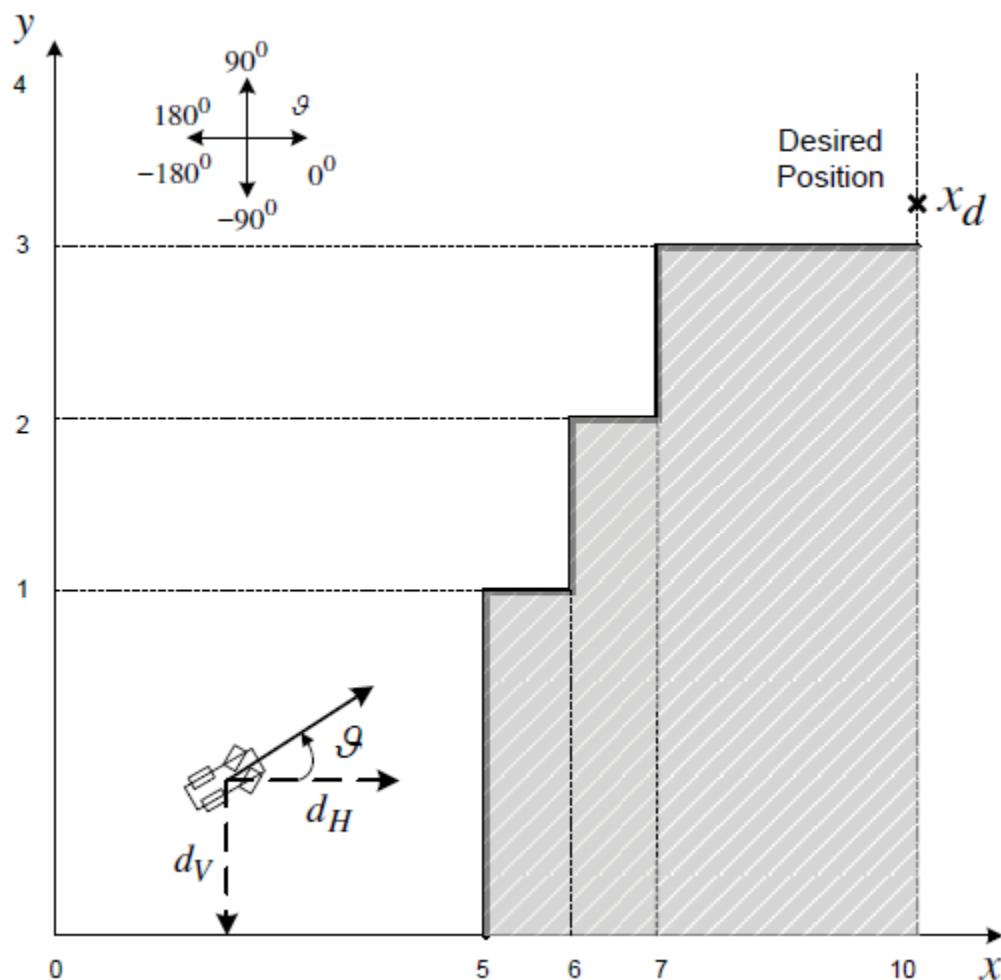
Όπως βλέπουμε από τα παραπάνω γραφήματα η έξοδος ακολουθεί την είσοδο αναφοράς, ενώ όταν εμφανίζεται η διαταραχή παρουσιάζει μία μικρή βύθιση κι έπειτα επανέρχεται στην κανονική τιμή λειτουργίας. Όταν η τιμή της ροπής της διαταραχής μειώνεται το σύστημα εμφανίζει μία μικρή υπερύψωση και στην συνέχεια επανέρχεται κι ακολουθεί την ονομαστική τιμή, δηλαδή το σήμα αναφοράς. Σε όλη την διάρκεια της προσομοίωσης η τάση εξόδου του ελεγκτή παραμένει μικρότερη από 200V και οι προδιαγραφές ικανοποιούνται.

## 2. Εργασία #2: FLC Car Control

A

### 2.1 Συναρτήσεις Συμμετοχής

Σκοπός του FLC ελεγκτή είναι να οδηγήσει το όχημα στην επιθυμητή θέση ( $x_d, y_d$ ) = (10, 3.2), χωρίς να πέσει πάνω στα εμπόδια. Η ταχύτητα το οχήματος έχει σταθερό μέτρο και είναι ίση με  $v=0.05$  m/sec.



Σχήμα 50: Εμπόδια και επιθυμητή θέση

Ο FLC δέχεται τρείς εισόδους, την κάθετη απόσταση από το εμπόδιο dV, της οριζόντια απόσταση από το εμπόδιο dH και την γωνία θ, δηλαδή την κατεύθυνση του οχήματος. Η έξοδος του ελεγκτή δίνει την μεταβολή dθ της γωνίας με την οποία θα πρέπει να στρήψει το όχημα. Τα μεγέθη dV και dH είναι κανονικοποιημένα στο διάστημα [0,1], η γωνία θ στο διάστημα [-180,180] και η μεταβολή της γωνίας στο διάστημα [-130,130].

Οι είσοδοι dV, dH αποτελούνται από 3 λεκτικές τιμές {S,M,L}={Small,Medium,Large}, ενώ οι τελευταία είσοδος θ και η έξοδος dθ αποτελούνται από 3 λεκτικές τιμές {N,Z,P}={Negative,Zero,Positive}.

Ο ελεγκτής κατασκευάζεται στο script “CarControl.m”.

```
%% Membership Functions
% Για την είσοδο dV
car_control = addmf(car_control, 'input', 1, 'S', 'trimf', [0,0,0.5]);
car_control = addmf(car_control, 'input', 1, 'M', 'trimf', [0,0.5,1]);
car_control = addmf(car_control, 'input', 1, 'L', 'trimf', [0.5,1,1]);

% Για την είσοδο dH
car_control = addmf(car_control, 'input', 2, 'S', 'trimf', [0,0,0.5]);
car_control = addmf(car_control, 'input', 2, 'M', 'trimf', [0,0.5,1]);
car_control = addmf(car_control, 'input', 2, 'L', 'trimf', [0.5,1,1]);

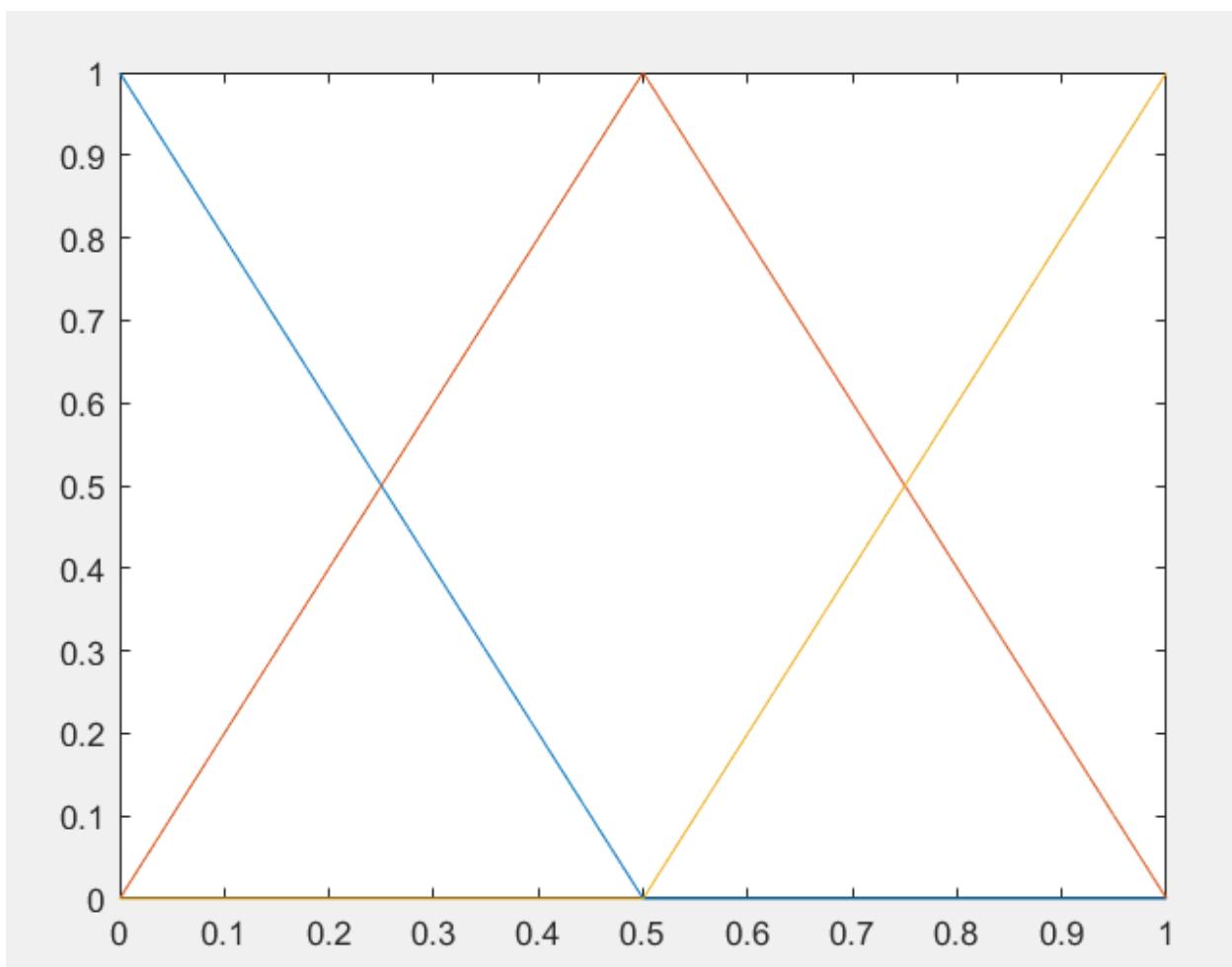
% Για την είσοδο theta
car_control = addmf(car_control, 'input', 3, 'N', 'trimf', [-180,-180,0]);
car_control = addmf(car_control, 'input', 3, 'Z', 'trimf', [-180,0,180]);
car_control = addmf(car_control, 'input', 3, 'P', 'trimf', [0,180,180]);

% Για την έξοδο dtheta - Αχρικές τιμές
car_control = addmf(car_control, 'output', 1, 'N', 'trimf', [-130,-130,0]);
car_control = addmf(car_control, 'output', 1, 'Z', 'trimf', [-130,0,130]);
car_control = addmf(car_control, 'output', 1, 'P', 'trimf', [0,130,130]);
```

Σχήμα 51: Κώδικας Συνάρτησης Συμμετοχής εισόδων dV,dH,θ και εξόδου dθ

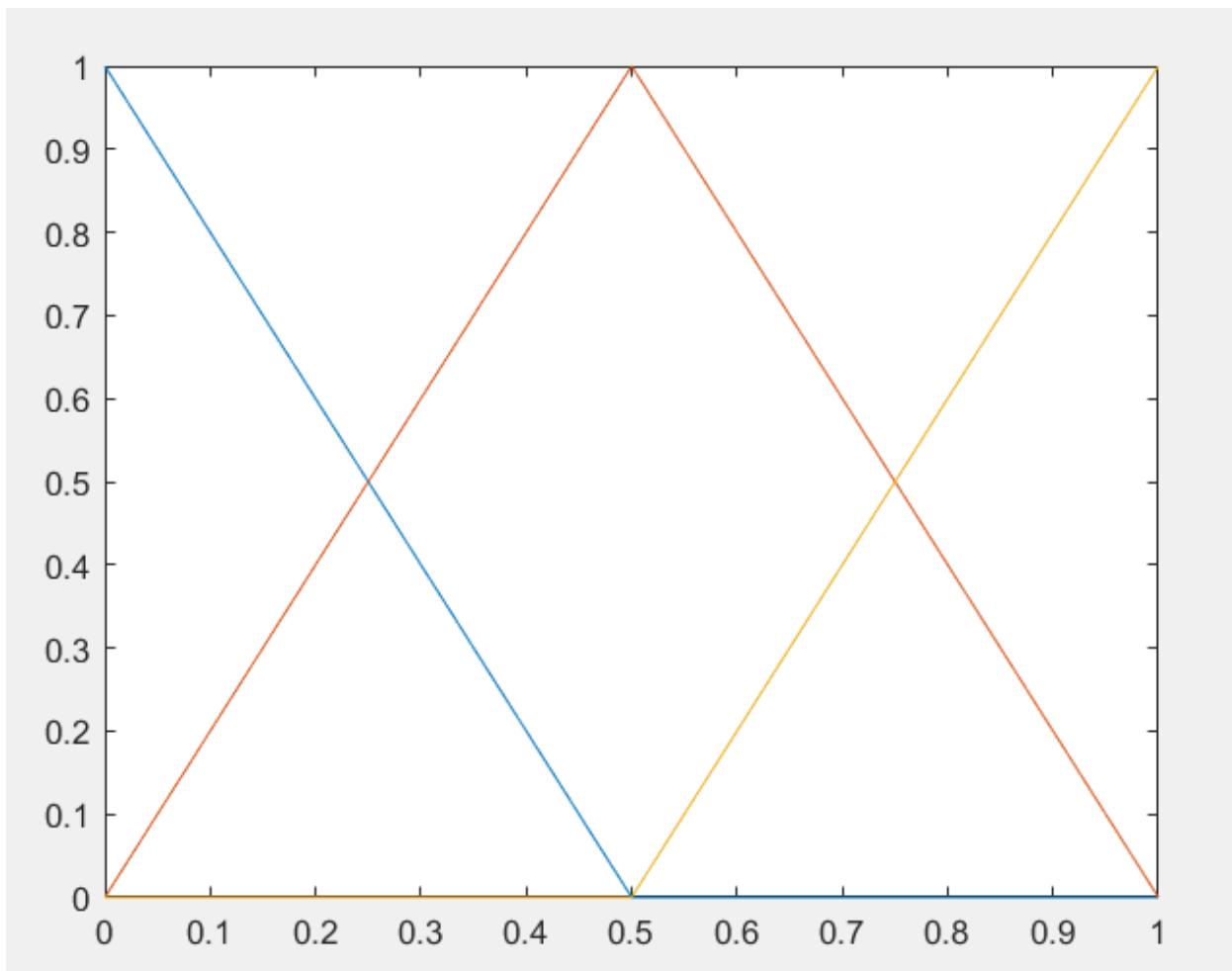
Οι λεκτικές τιμές των εισόδων και της εξόδου φαίνονται στα Σχήματα 52,53,54,55.

$\sim \text{plot}(xOut\_dV, yOut\_dV)$



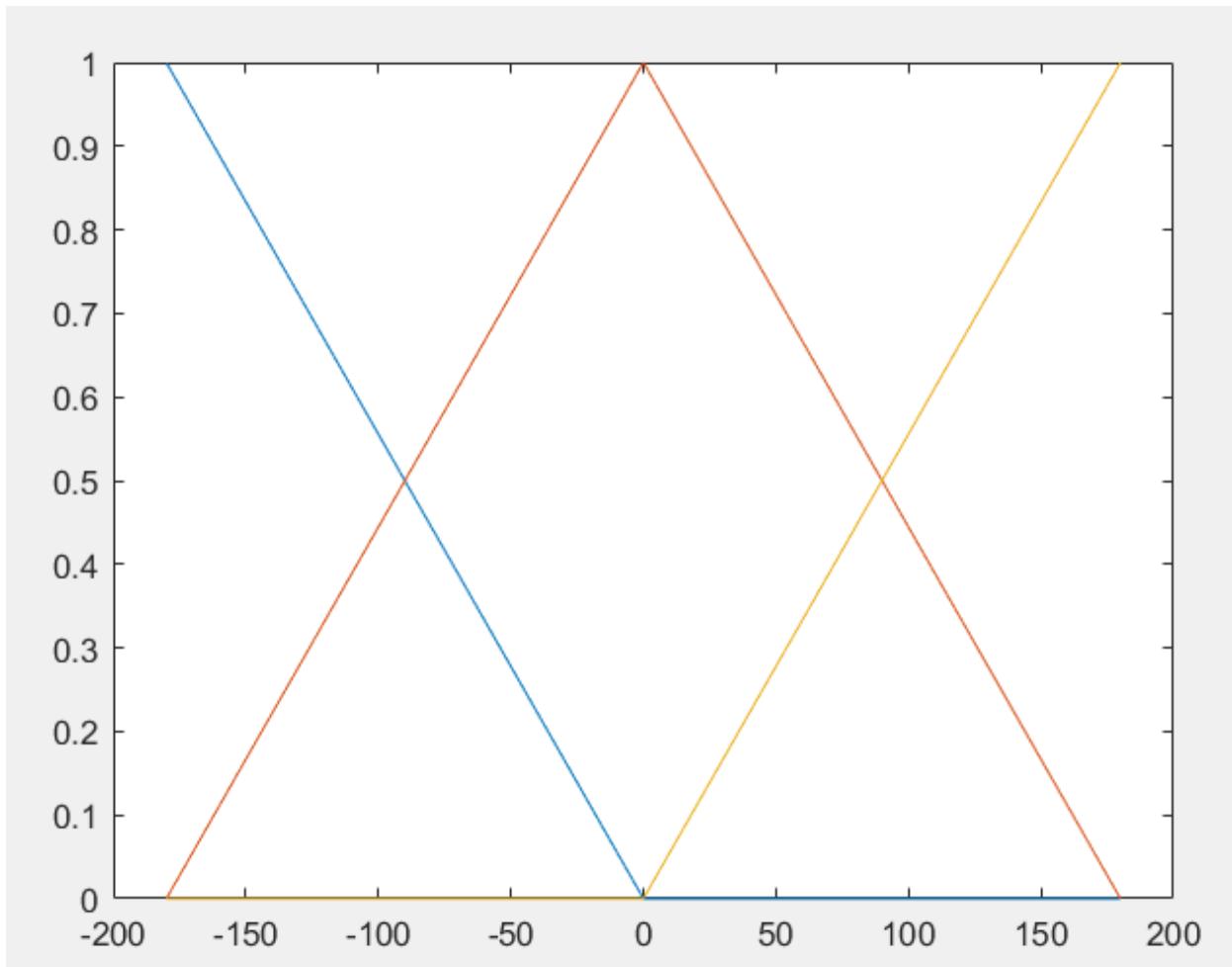
Σχήμα 52: Λεκτικές τιμές Συνάρτησης Συμμετοχής dV

`~ plot(xOut_dH,yOut_dH)`



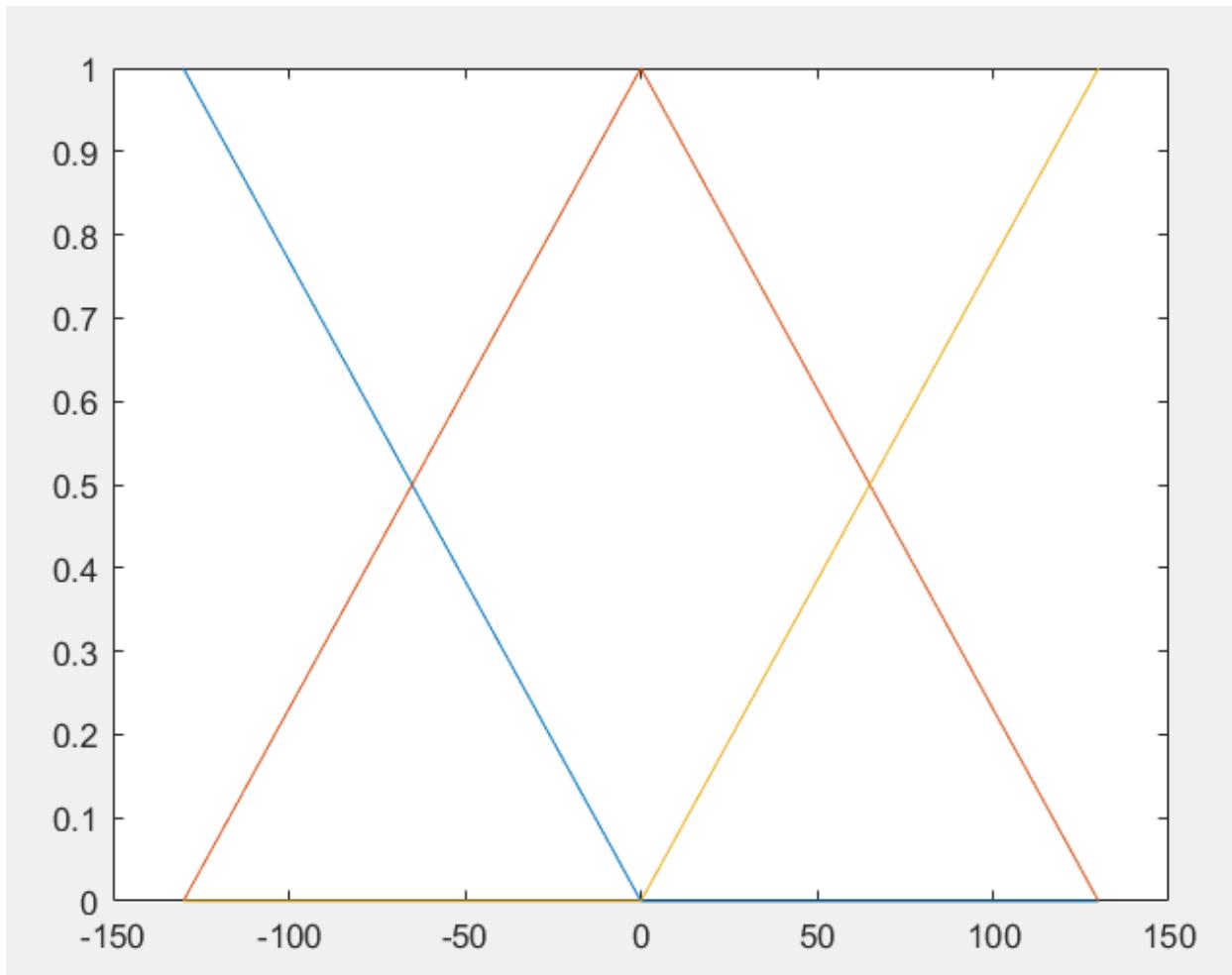
Σχήμα 53: Λεκτικές τιμές Συνάρτησης Συμμετοχής dH

`~ plot(xOut_theta,yOut_theta)`



Σχήμα 54: Λεκτικές τιμές Συνάρτησης Συμμετοχής  $\theta$

`~ plot(xOut_dtheta,yOut_dtheta)`



Σχήμα 55: Λεκτικές τιμές Συνάρτησης Συμμετοχής  $d\theta$

## 2.2 Βάση Κανόνων

### 2.2.1 Αρχική Βάση Κανόνων

Θα κατασκευάσουμε μία αρχική βάση κανόνων, με τη λογική ότι το όχημα θα ανεβαίνει πάνω↑, δηλαδή θέλω  $\theta_{new} \approx 90^\circ$ , όταν πλησιάζει το κάθετο εμπόδιο και θα στρήβει δεξιά→, δηλαδή θέλω  $\theta_{new} \approx 0^\circ$ , όταν πλησιάζει οριζόντιο εμπόδιο.

Δίνουμε προτεραιότητα στην κίνηση προς τα πάνω↑, έναντι της δεξιάς στροφής→.

- Όταν η κάθετη απόσταση είναι μικρή  $dV=S$  και η οριζόντια απόσταση είναι μικρή ή μεσαία  $dH=\{S,M\}$ , θέλω το όχημα να κατευθευθεί προς τα πάνω↑, ενώ όταν είναι μεγάλη  $dH=L$  προς τα δεξιά→.
- Όταν η κάθετη απόσταση είναι μεσαία  $dV=M$  και η οριζόντια απόσταση είναι μικρή ή μεσαία  $dH=\{S,M\}$ , θέλω το όχημα να κατευθευθεί προς τα πάνω↑, ενώ όταν είναι μεγάλη προς τα δεξιά→.
- Όταν η κάθετη απόσταση είναι μεγάλη  $dV=L$  και η οριζόντια απόσταση είναι μικρή ή μεσαία  $dH=\{S,M\}$ , θέλω το όχημα να κατευθευθεί προς τα πάνω↑, ενώ όταν είναι μεγάλη προς τα δεξιά→.

Η γωνία προσαρμόζεται με βάση το που θέλω να κατευθυνθεί το όχημα.

- ❖  $\theta_{new} = \theta + d\theta$   
π.χ. αν θέλω  $\theta_{new}=P$  και έχω  $\theta=N$ , θα πρέπει ο FLC να δώσει έξοδο  $d\theta=P$

Με βάση τους παραπάνω κανόνες η βάση είναι η εξής:

Για  $dV=S$

$dH \setminus \theta$	N	Z	P
S	P	P	Z
M	P	P	Z
L	P	Z	N

Για  $dV=M$

$dH \setminus \theta$	N	Z	P
S	P	P	Z
M	P	P	Z
L	P	Z	N

Για  $dV=L$

$dH \setminus \theta$	N	Z	P
S	P	P	Z
M	P	P	Z
L	P	Z	N

Και οι κανόνες που προκύπτου είναι οι 27 παρακάτω:

- '1. If ( $dV$  is S) and ( $dH$  is S) and ( $\theta$  is N) then ( $d\theta$  is P) (1) '
- '2. If ( $dV$  is S) and ( $dH$  is S) and ( $\theta$  is Z) then ( $d\theta$  is P) (1) '
- '3. If ( $dV$  is S) and ( $dH$  is S) and ( $\theta$  is P) then ( $d\theta$  is Z) (1) '
- '4. If ( $dV$  is S) and ( $dH$  is M) and ( $\theta$  is N) then ( $d\theta$  is P) (1) '
- '5. If ( $dV$  is S) and ( $dH$  is M) and ( $\theta$  is Z) then ( $d\theta$  is P) (1) '
- '6. If ( $dV$  is S) and ( $dH$  is M) and ( $\theta$  is P) then ( $d\theta$  is Z) (1) '
- '7. If ( $dV$  is S) and ( $dH$  is L) and ( $\theta$  is N) then ( $d\theta$  is P) (1) '
- '8. If ( $dV$  is S) and ( $dH$  is L) and ( $\theta$  is Z) then ( $d\theta$  is Z) (1) '
- '9. If ( $dV$  is S) and ( $dH$  is L) and ( $\theta$  is P) then ( $d\theta$  is N) (1) '
- '10. If ( $dV$  is M) and ( $dH$  is S) and ( $\theta$  is N) then ( $d\theta$  is P) (1)'
- '11. If ( $dV$  is M) and ( $dH$  is S) and ( $\theta$  is Z) then ( $d\theta$  is P) (1)'
- '12. If ( $dV$  is M) and ( $dH$  is S) and ( $\theta$  is P) then ( $d\theta$  is Z) (1)'
- '13. If ( $dV$  is M) and ( $dH$  is M) and ( $\theta$  is N) then ( $d\theta$  is P) (1)'
- '14. If ( $dV$  is M) and ( $dH$  is M) and ( $\theta$  is Z) then ( $d\theta$  is P) (1)'
- '15. If ( $dV$  is M) and ( $dH$  is M) and ( $\theta$  is P) then ( $d\theta$  is Z) (1)'
- '16. If ( $dV$  is M) and ( $dH$  is L) and ( $\theta$  is N) then ( $d\theta$  is P) (1)'
- '17. If ( $dV$  is M) and ( $dH$  is L) and ( $\theta$  is Z) then ( $d\theta$  is Z) (1)'
- '18. If ( $dV$  is M) and ( $dH$  is L) and ( $\theta$  is P) then ( $d\theta$  is N) (1)'
- '19. If ( $dV$  is L) and ( $dH$  is S) and ( $\theta$  is N) then ( $d\theta$  is P) (1)'
- '20. If ( $dV$  is L) and ( $dH$  is S) and ( $\theta$  is Z) then ( $d\theta$  is P) (1)'
- '21. If ( $dV$  is L) and ( $dH$  is S) and ( $\theta$  is P) then ( $d\theta$  is Z) (1)'

'22. If (dV is L) and (dH is M) and (theta is N) then (dtheta is P) (1)'

'23. If (dV is L) and (dH is M) and (theta is Z) then (dtheta is P) (1)'

'24. If (dV is L) and (dH is M) and (theta is P) then (dtheta is Z) (1)'

'25. If (dV is L) and (dH is L) and (theta is N) then (dtheta is P) (1)'

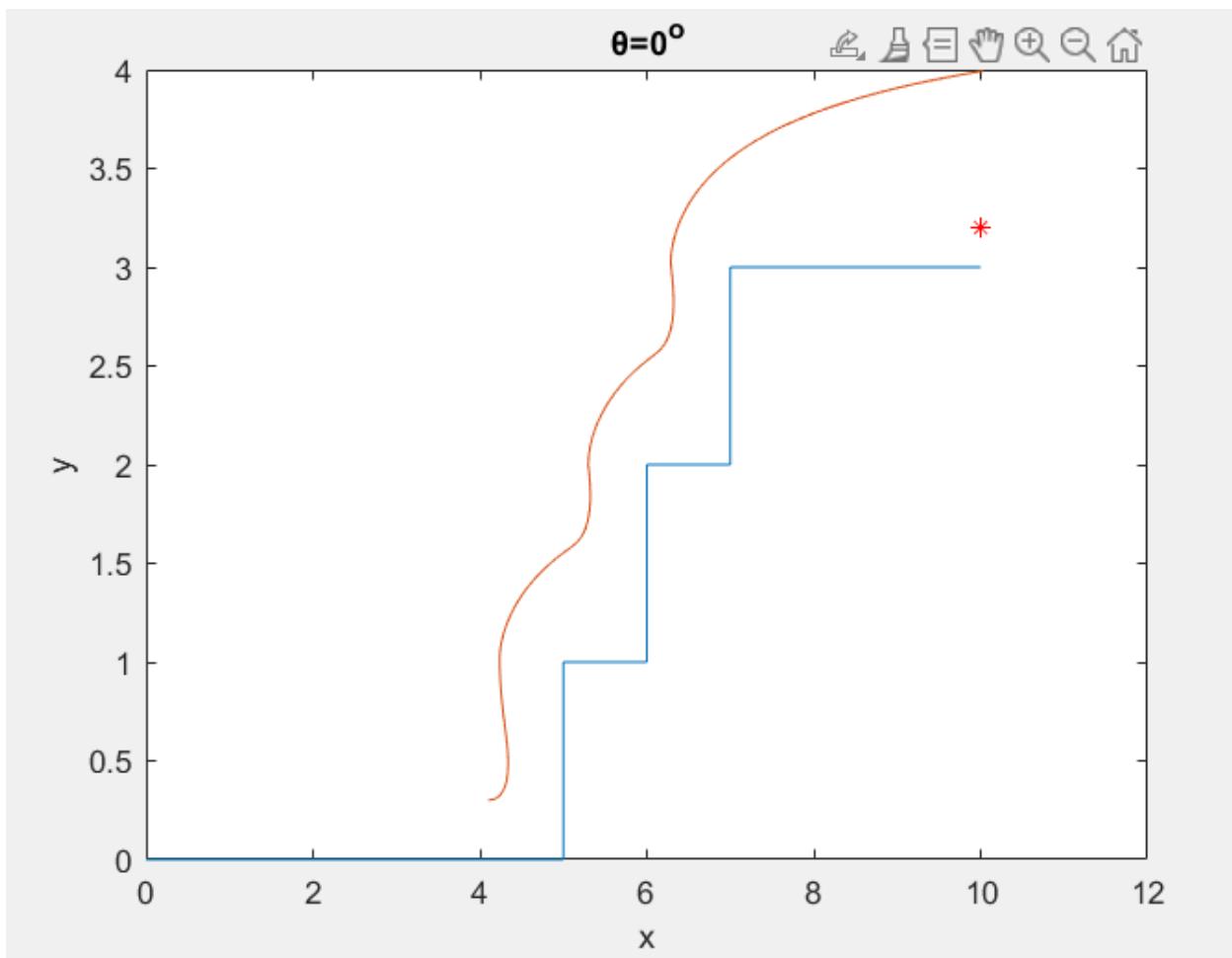
'26. If (dV is L) and (dH is L) and (theta is Z) then (dtheta is Z) (1)'

'27. If (dV is L) and (dH is L) and (theta is P) then (dtheta is N) (1)'

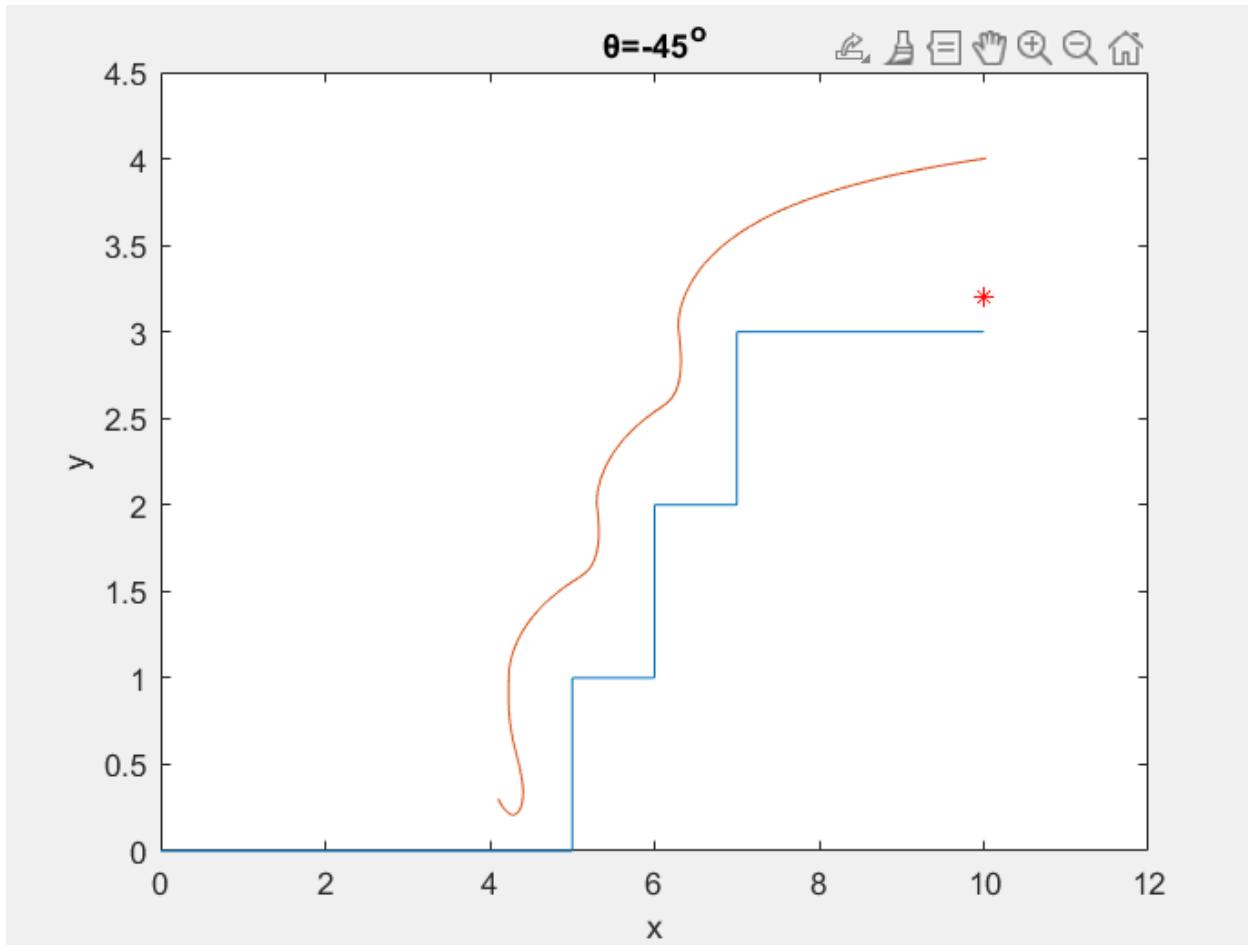
Στη συνέχεια βλέπουμε την τροχιά του οχήματος, η οποία υπολογίζεται με τη συνάρτηση “car\_trajectory.m”.

Τροφοδοτούμε το σύστημα με τις αρχικές τιμές:

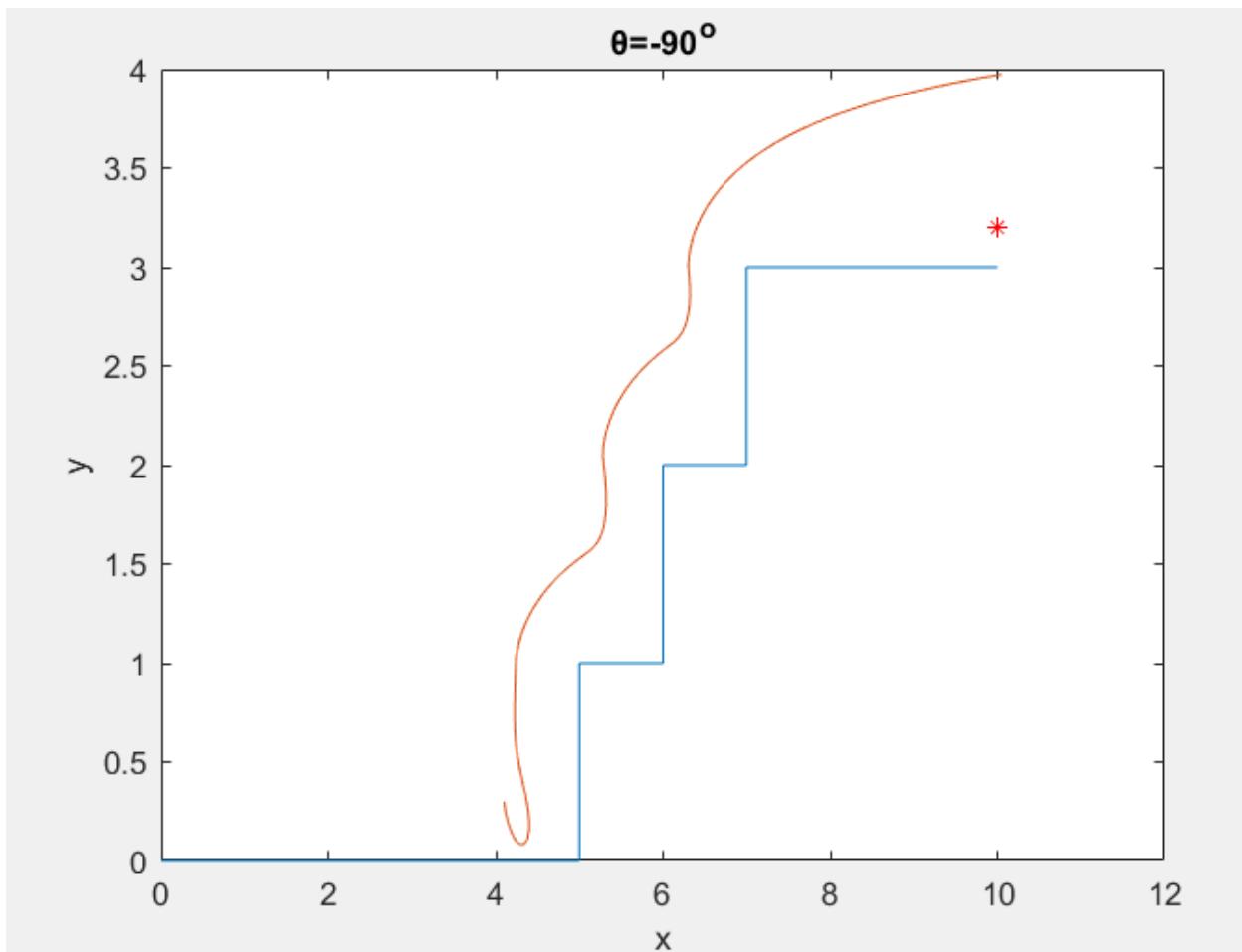
- ❖ Αρχική θέση:  $(x\_init, y\_init) = (4.1, 0.3)$
- ❖ Αρχικές γωνίες:  $\theta_{init} = \{0^\circ, -45^\circ, -90^\circ\}$



Σχήμα 56: Τροχιά οχήματος  $\theta_{init}=0^\circ$  - Αρχική Βάση Κανόνων



Σχήμα 57: Τροχιά οχήματος  $\theta_{\text{init}} = -45^\circ$  - Αρχική Βάση Κανόνων



Σχήμα 58: Τροχιά οχήματος  $\theta_{\text{init}} = -90^\circ$  - Αρχική Βάση Κανόνων

Παρατηρούμε ότι το όχημα ακολουθεί σωστά την πορεία και αποφεύγει τα εμπόδια και στον οριζόντιο και στον κατακόρυφο άξονα, όμως η ακρίβεια και η απόκριση του δεν είναι ικανοποιητική.

## 2.2.2 Τελική Βάση Κανόνων

Προκειμένου να πετύχουμε καλύτερη προσέγγιση στην επιθυμητή θέση θα κάνουμε δύο σημαντικές αλλαγές στο FLC σύστημα.

1. Νέα Βάση Κανόνων με προτεραιότητα στην κίνηση προς τα δεξιά →, έναντι της κίνησης προς τα πάνω ↑.
2. Προσαρμογές στις τιμές της Συνάρτησης Συμμετοχής της εξόδου dθ.

Δίνουμε προτεραιότητα στη δεξιά στροφή →, έναντι της κίνηση προς τα πάνω ↑.

Η νέα βάση κανόνων είναι η παρακάτω και σημειώνουμε με **κόκκινο χρώμα** τις αλλαγές σε σχέση με την αρχική βάση.

Με την παρακάτω βάση, οδηγούμε το όχημα πιο κοντά στα κάθετα εμπόδια και έχει μεγαλύτερη ακρίβεια.

- Όταν η κάθετη απόσταση είναι μικρή  $dV=S$  και η οριζόντια απόσταση είναι μεσαία  $dH=M$ , θέλω η τροχιά του οχήματος να είναι προς τα δεξιά →.
- Όταν η κάθετη απόσταση είναι μεσαία  $dV=M$  και η οριζόντια απόσταση είναι μεσαία  $dH=M$ , θέλω η τροχιά του οχήματος να είναι προς τα δεξιά →.
- Όταν η κάθετη απόσταση είναι μεγάλη  $dV=L$ , θέλω η τροχιά του οχήματος να είναι προς τα δεξιά →, ανεξάρτητα από την τιμή της οριζόντιας απόστασης.

Για  $dV=S$

<b><math>dH\backslash \theta</math></b>	<b>N</b>	<b>Z</b>	<b>P</b>
<b>S</b>	P	P	Z
<b>M</b>	P	<b>Z</b>	<b>N</b>
<b>L</b>	P	Z	N

Για  $dV=M$

<b><math>dH\backslash \theta</math></b>	<b>N</b>	<b>Z</b>	<b>P</b>
<b>S</b>	P	P	Z
<b>M</b>	P	<b>Z</b>	<b>N</b>
<b>L</b>	P	Z	N

Για  $dV=L$

<b><math>dH\backslash \theta</math></b>	<b>N</b>	<b>Z</b>	<b>P</b>
<b>S</b>	P	<b>Z</b>	<b>N</b>
<b>M</b>	P	<b>Z</b>	<b>N</b>
<b>L</b>	P	Z	N

Οι τελικοί κανόνες που προκύπτου είναι οι 27 παρακάτω:

- '1. If (dV is S) and (dH is S) and (theta is N) then (dtheta is P) (1) '
- '2. If (dV is S) and (dH is S) and (theta is Z) then (dtheta is P) (1) '
- '3. If (dV is S) and (dH is S) and (theta is P) then (dtheta is Z) (1) '
- '4. If (dV is S) and (dH is M) and (theta is N) then (dtheta is P) (1) '
- '5. If (dV is S) and (dH is M) and (theta is Z) then (dtheta is Z) (1) '
- '6. If (dV is S) and (dH is M) and (theta is P) then (dtheta is N) (1) '
- '7. If (dV is S) and (dH is L) and (theta is N) then (dtheta is P) (1) '
- '8. If (dV is S) and (dH is L) and (theta is Z) then (dtheta is Z) (1) '
- '9. If (dV is S) and (dH is L) and (theta is P) then (dtheta is N) (1) '
- '10. If (dV is M) and (dH is S) and (theta is N) then (dtheta is P) (1)'
- '11. If (dV is M) and (dH is S) and (theta is Z) then (dtheta is P) (1)'
- '12. If (dV is M) and (dH is S) and (theta is P) then (dtheta is Z) (1)'
- '13. If (dV is M) and (dH is M) and (theta is N) then (dtheta is P) (1)'
- '14. If (dV is M) and (dH is M) and (theta is Z) then (dtheta is Z) (1)'
- '15. If (dV is M) and (dH is M) and (theta is P) then (dtheta is N) (1)'
- '16. If (dV is M) and (dH is L) and (theta is N) then (dtheta is P) (1)'
- '17. If (dV is M) and (dH is L) and (theta is Z) then (dtheta is Z) (1)'
- '18. If (dV is M) and (dH is L) and (theta is P) then (dtheta is N) (1)'
- '19. If (dV is L) and (dH is S) and (theta is N) then (dtheta is P) (1)'
- '20. If (dV is L) and (dH is S) and (theta is Z) then (dtheta is Z) (1)'
- '21. If (dV is L) and (dH is S) and (theta is P) then (dtheta is N) (1)'
- '22. If (dV is L) and (dH is M) and (theta is N) then (dtheta is P) (1)'
- '23. If (dV is L) and (dH is M) and (theta is Z) then (dtheta is Z) (1)'
- '24. If (dV is L) and (dH is M) and (theta is P) then (dtheta is N) (1)'
- '25. If (dV is L) and (dH is L) and (theta is N) then (dtheta is P) (1)'
- '26. If (dV is L) and (dH is L) and (theta is Z) then (dtheta is Z) (1)'

'27. If (dV is L) and (dH is L) and (theta is P) then (dtheta is N) (1)'

Αφού βελτιώσαμε την Βάση Κανόνων, στην συνέχεια θα κάνουμε αλλαγές στην Συνάρτηση Συμμετοχής της εξόδου dθ.

- ❖ Για τις ακραίες τιμές 'N' και 'P' χρησιμοποιώ τραπεζοειδή Συνάρτηση Συμμετοχής
- ❖ Για την μεσαία τιμή 'Z' μικραίνουμε το πεδίο ορισμού στο [-90,90], ώστε να πετυχαίνουμε μεγαλύτερη ακρίβεια κάθε φορά που μεταβάλουμε τη γωνία.

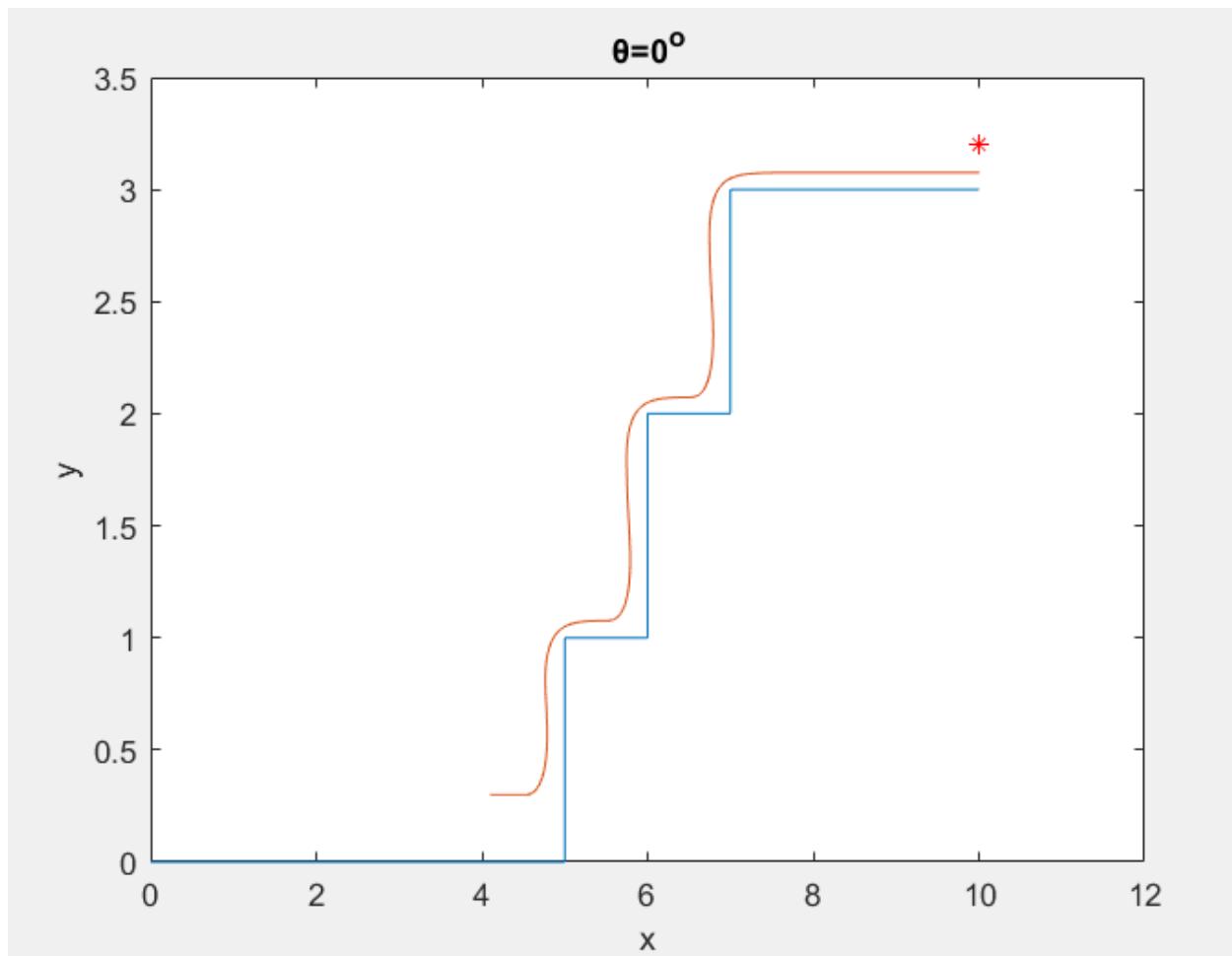
```
% Για την έξοδο dtheta - Τελικές τιμές
car_control = addmf(car_control, 'output', 1, 'N', 'trapmf', [-130,-130,-100,0]);
car_control = addmf(car_control, 'output', 1, 'Z', 'trimf', [-90,0,90]);
car_control = addmf(car_control, 'output', 1, 'P', 'trapmf', [0,100,130,130]);
```

Σχήμα 59: Κώδικας Τελικής Συνάρτησης Συμμετοχής εξόδου dθ

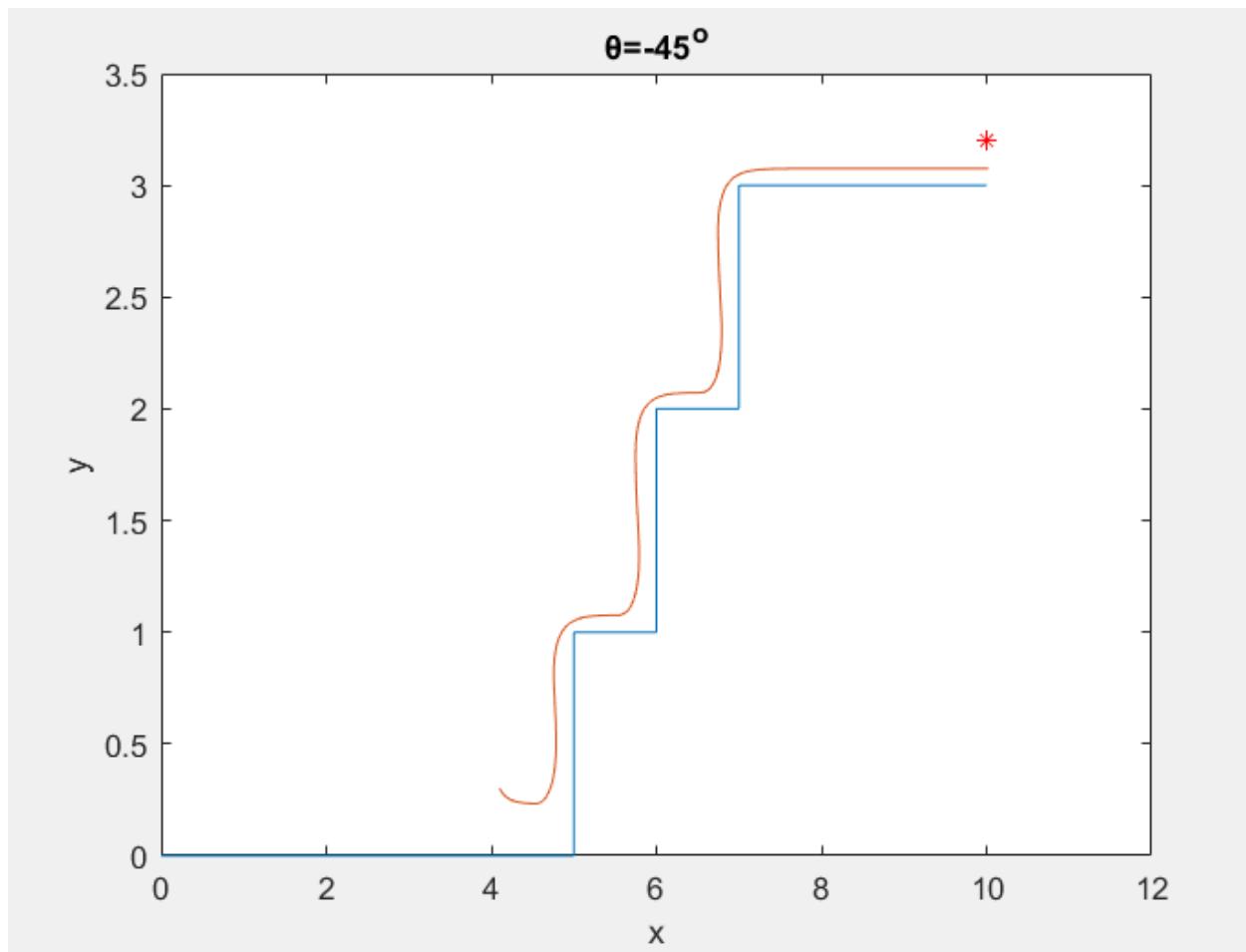
Μετά τις αλλαγές θα δοκιμάσουμε να ελέγξουμε την τροχιά του οχήματος και να συγκρίνουμε την καμπύλη με τις αρχικές.

Τροφοδοτούμε το σύστημα με τις ίδιες αρχικές τιμές:

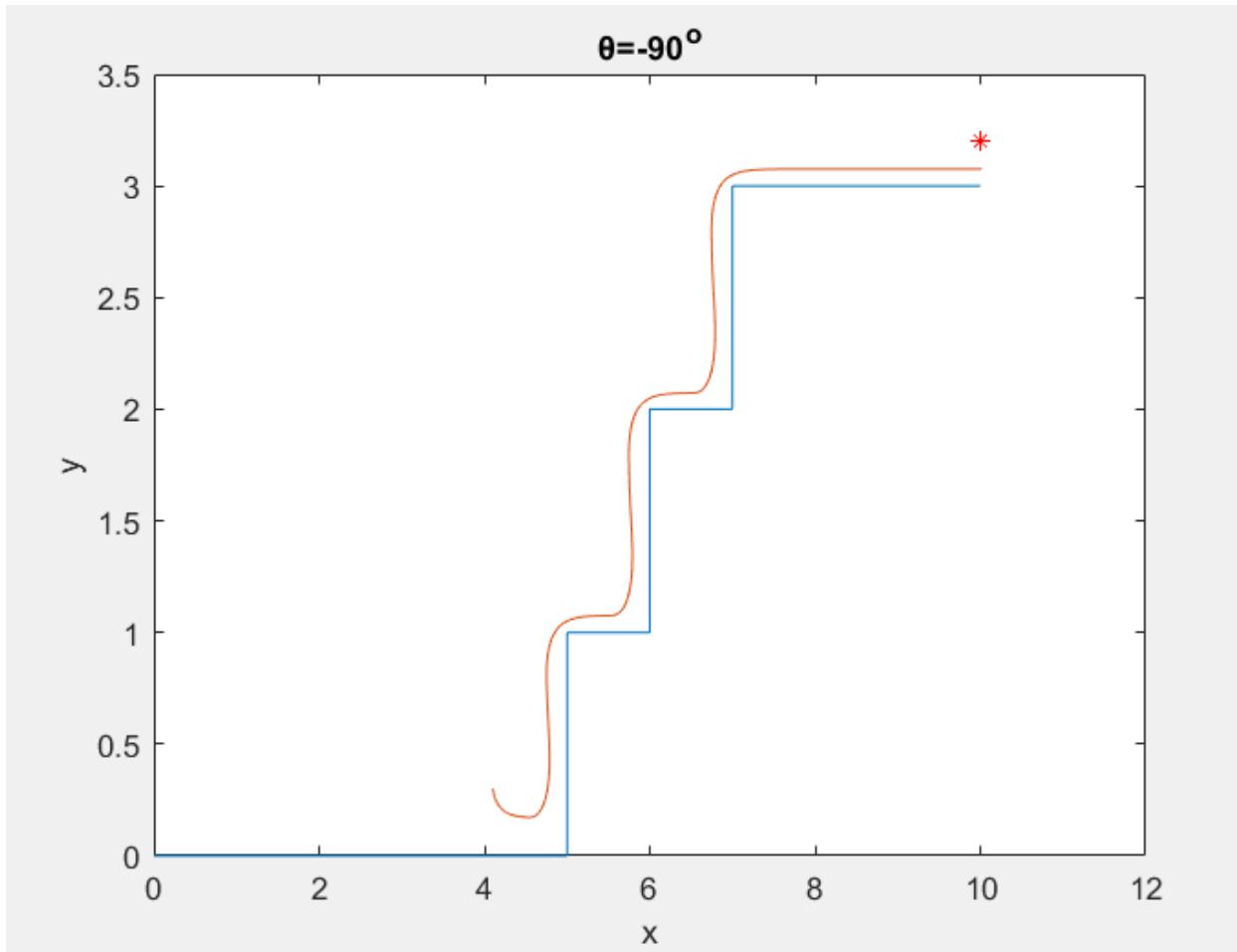
- ❖ Αρχική Θέση: (x\_init,y\_init)=(4.1,0.3)
- ❖ Αρχικές γωνίες: θinit={0°,-45°,-90°}



Σχήμα 60: Τροχιά οχήματος  $\theta_{\text{init}}=0^\circ$  - Τελική Βάση Κανόνων

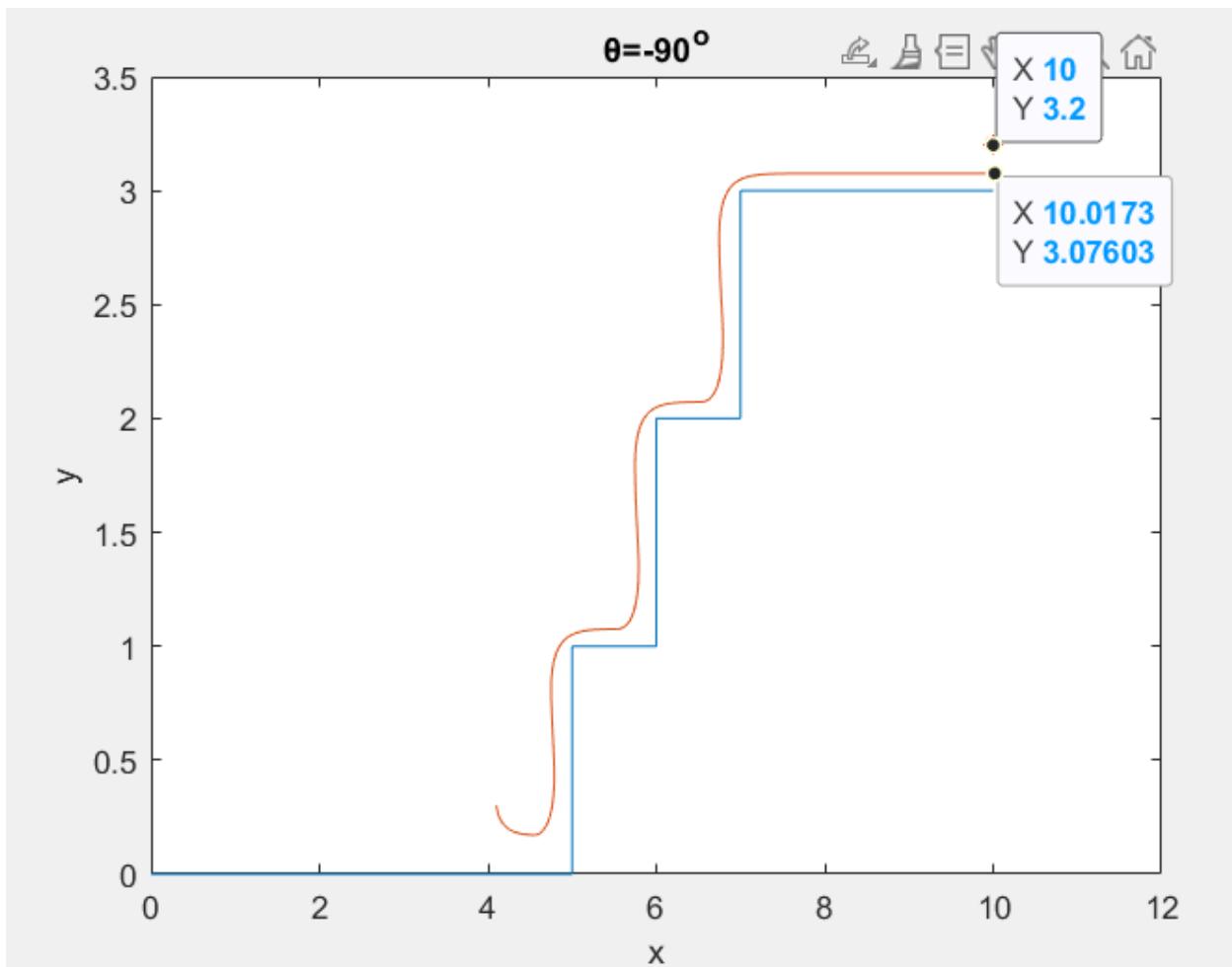


Σχήμα 61: Τροχιά οχήματος  $\theta_{init}=-45^\circ$  - Τελική Βάση Κανόνων



Σχήμα 62: Τροχιά οχήματος  $\theta_{\text{init}}=-90^\circ$  - Τελική Βάση Κανόνων

Παρατηρούμε ότι η τροχιά είναι αρκετά πιο ακριβής. Ακολουθεί πιστά τα εμπόδια, χωρίς να έρχεται σε επαφή μαζί τους. Ακόμη και στις  $-90^\circ$  το όχημα προσαρμόζεται γρήγορα στην επιθυμητή τροχιά. Το όχημα φτάνει στην θέση  $(10.017, 3.076)$ , δηλαδή πολύ κοντά στην επιθυμητή θέση  $(10, 3.2)$ . Πετυχαίνει ακριβώς την τετμημένη  $x=10$ , το οποίο είναι και το ζητούμενο της εργασίας και πλησιάζει αρκετά την τεταγμένη  $y=3.2$ , για το οποίο θέλουμε απλά να είμαστε κοντά.



Σχήμα 63: Τελική θέση vs Επιθυμητή θέση

Το FLC πετυχαίνει ικανοποιητική τροχιά και τελική θέση.

# 3. Εργασία #3: Regression

## 3.1 Part 1 - Εφαρμογή σε απλό dataset

Σε αυτή την εργασία μελετάμε την ικανότητα των TSK μοντέλων (Takagi Sugeno Kang) στην μοντελοποίηση μοντέλων με πολλές μεταβλητές.

Δημιουργούμε 4 μοντέλα:

Πλήθος συναρτήσεων συμμετοχής	Μορφή εξόδου
TSK_model_1	2
TSK_model_2	3
TSK_model_3	2
TSK_model_4	3

Σχήμα 64: TSK μοντέλα

Το dataset που θα χρησιμοποιήσουμε είναι το “Airfoil Self-Noise” από το UCI repository, το οποίο περιέχει 1503 δεδομένα και αποτελείται από 5 features και 1 output.

Για κάθε ένα από τα μοντέλα υπολογίζουμε τα σφάλματα RMSE, NMSE, NDEI, R2, με τη συνάρτηση “calculate.m”, ώστε να αποφασίσουμε ποιό είναι το βέλτιστο.

Χωρίζουμε το dataset σε τρία υποσύνολα, 60% training data, 20% validation data, 20% checking data.

Για την εκπαίδευση του μοντέλου χρησιμοπούμε τη συνάρτηση anfis και επιλέγουμε EpochNumber = 200.

Επίσης, δημιουργούμε διαγράμματα για τις τελικές εκβάσεις των μοντέλων και των Συναστήσεων Συμμετοχής, για την καμπύλη εκμάθησης και για το σφάλμα πρόβλεψης.

Οι είσοδοι των μοντέλων είναι:

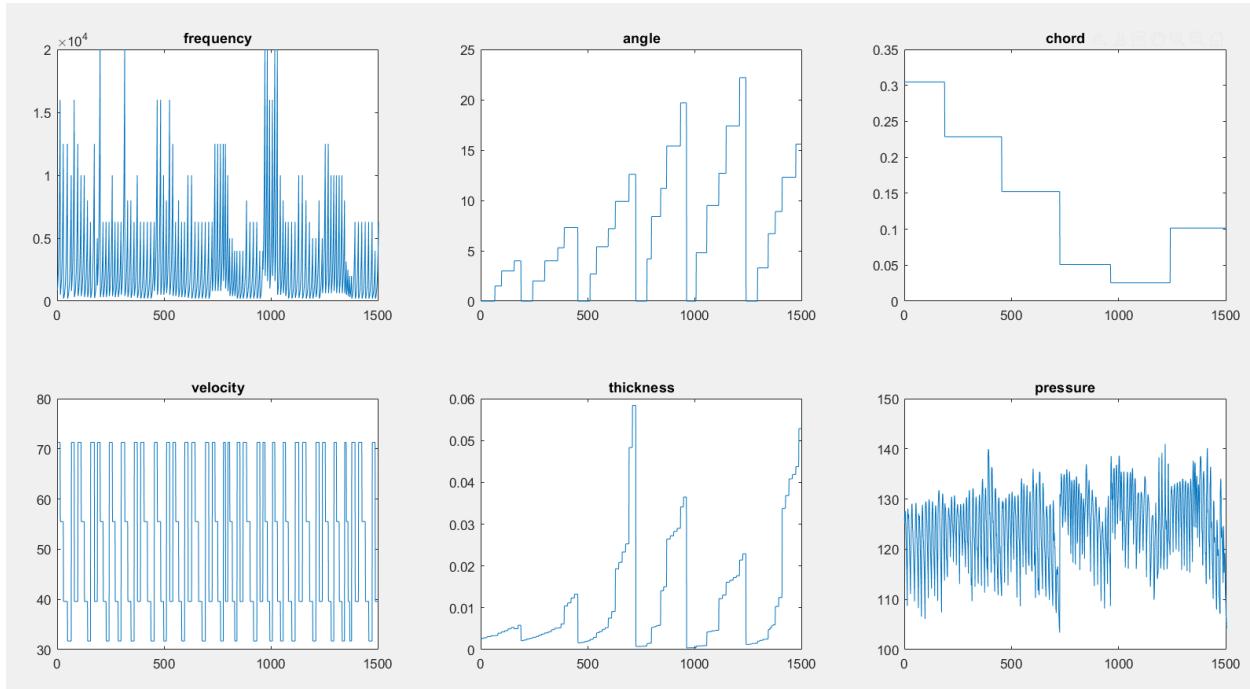
- frequency (Hz)
- angle of attack (degrees)
- chord length (m)
- free-stream velocity (m/s)

- suction side displacement thickness (m)

ενώ η έξοδος είναι :

- scaled sound pressure level (dB)

Παρακάτω φαίνονται οι τιμές των εισόδων και της εξόδου.



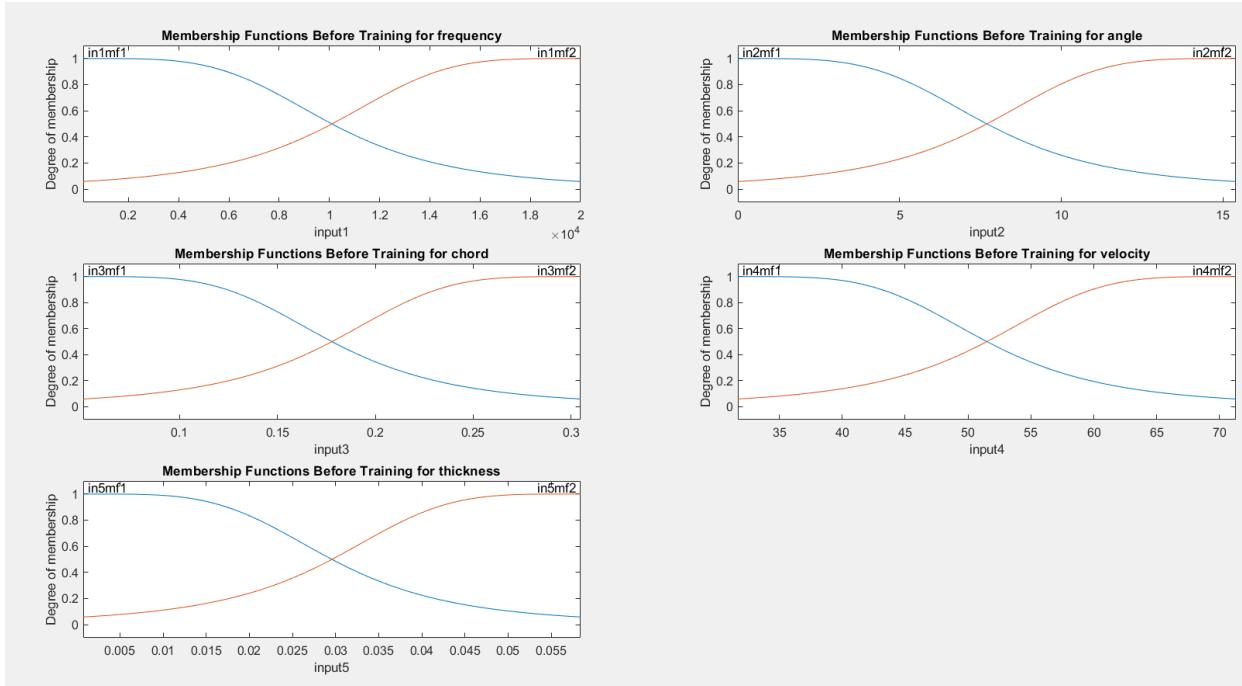
Σχήμα 65: Input & Output Values

### 3.1.1 TSK model 1

Το μοντέλο 1 υλοποιείται με το script “Regression\_TSK\_model\_1.m”.

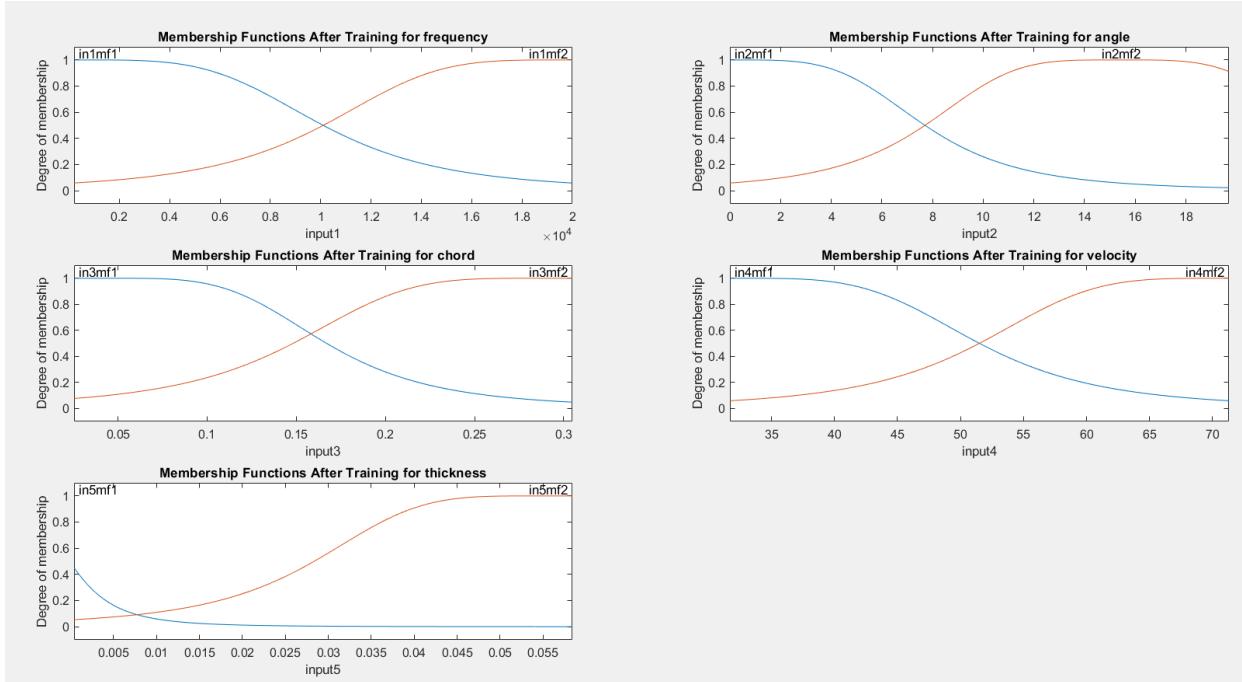
Χρησιμοποιεί μέθοδο ομαδοποίησης grid partition, 2 συναρτήσεις συμμετοχής “gbellmf” και “Singleton” output (constant).

## Συναρτήσεις Συμμετοχής Πριν την Εκπαίδευση



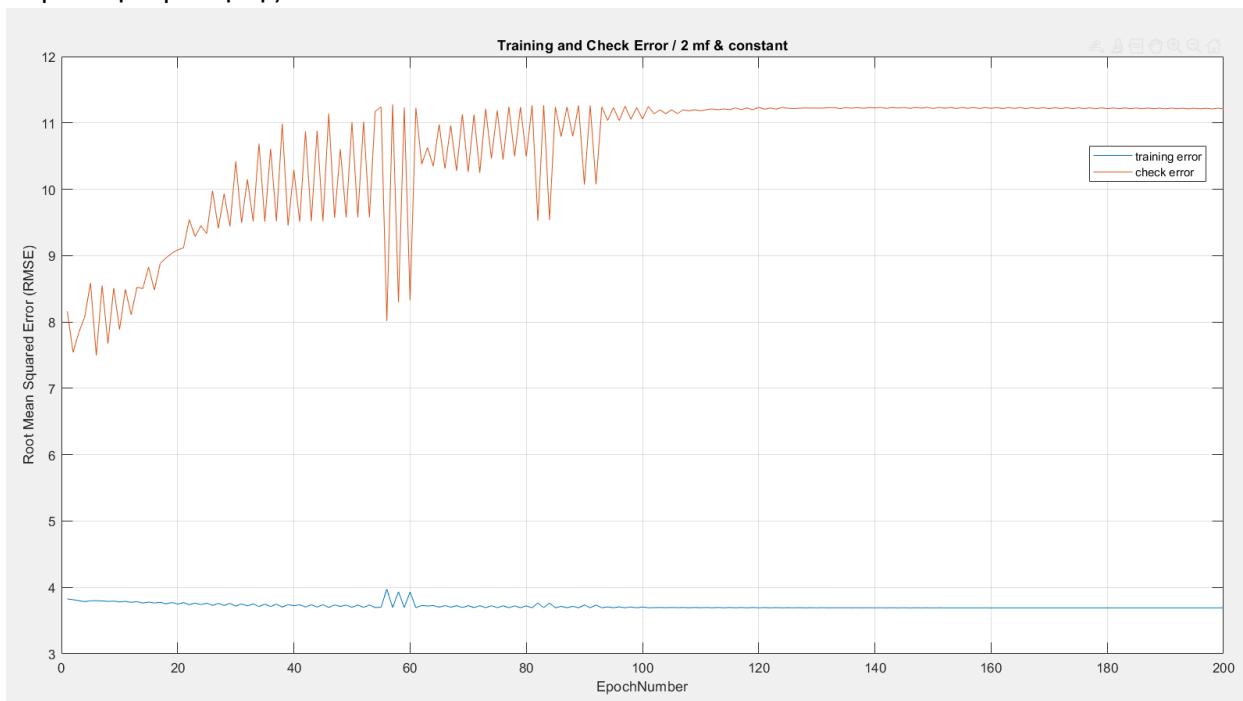
Σχήμα 66: mf Before Training

## Συναρτήσεις Συμμετοχής Μετά την Εκπαίδευση



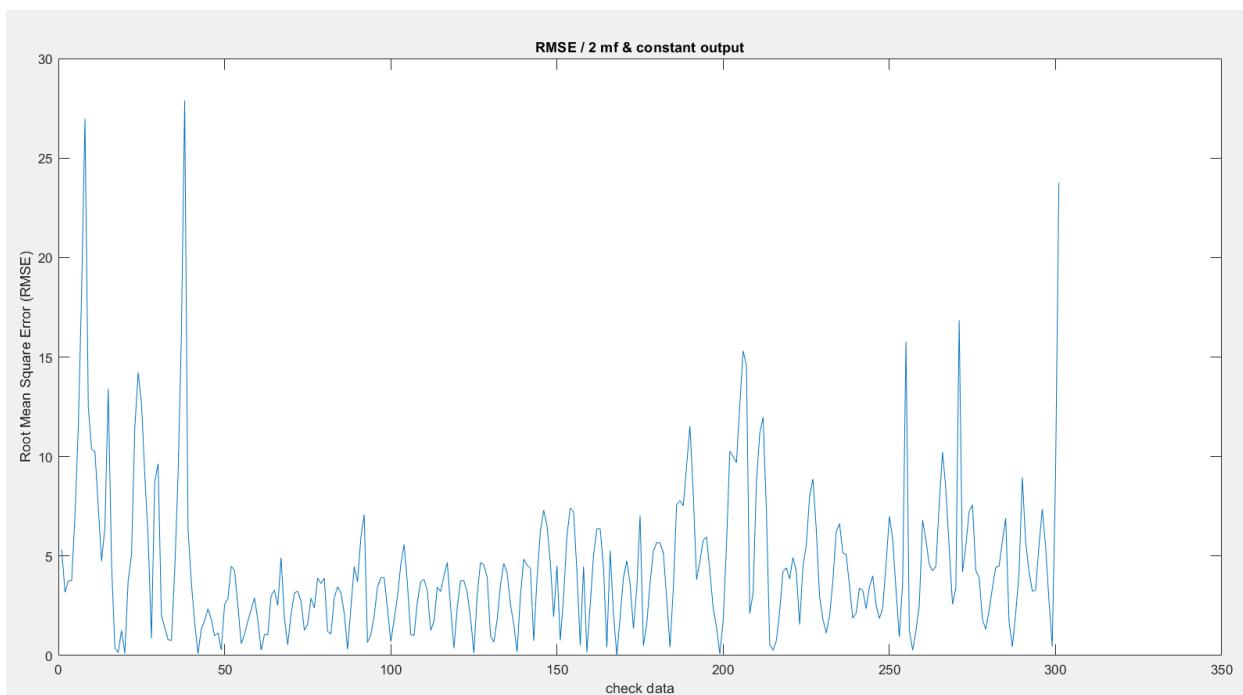
Σχήμα 67: mf After Training

## Καμπύλη Εκμάθησης



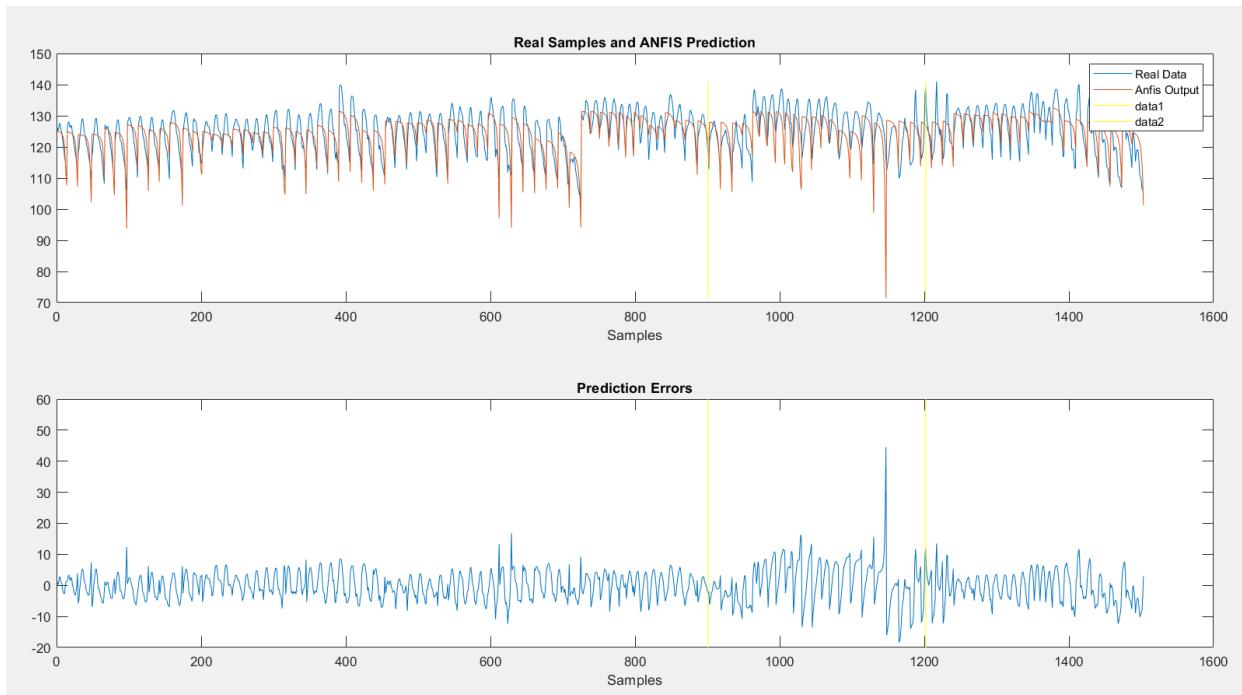
Σχήμα 68: Learning Curve

## RMSE



Σχήμα 69: RMSE

## Πραγματικές Τιμές VS Τιμές Πρόβλεψης & Σφάλμα Πρόβλεψης



Σχήμα 70: Real VS Prediction Values & Prediction Error

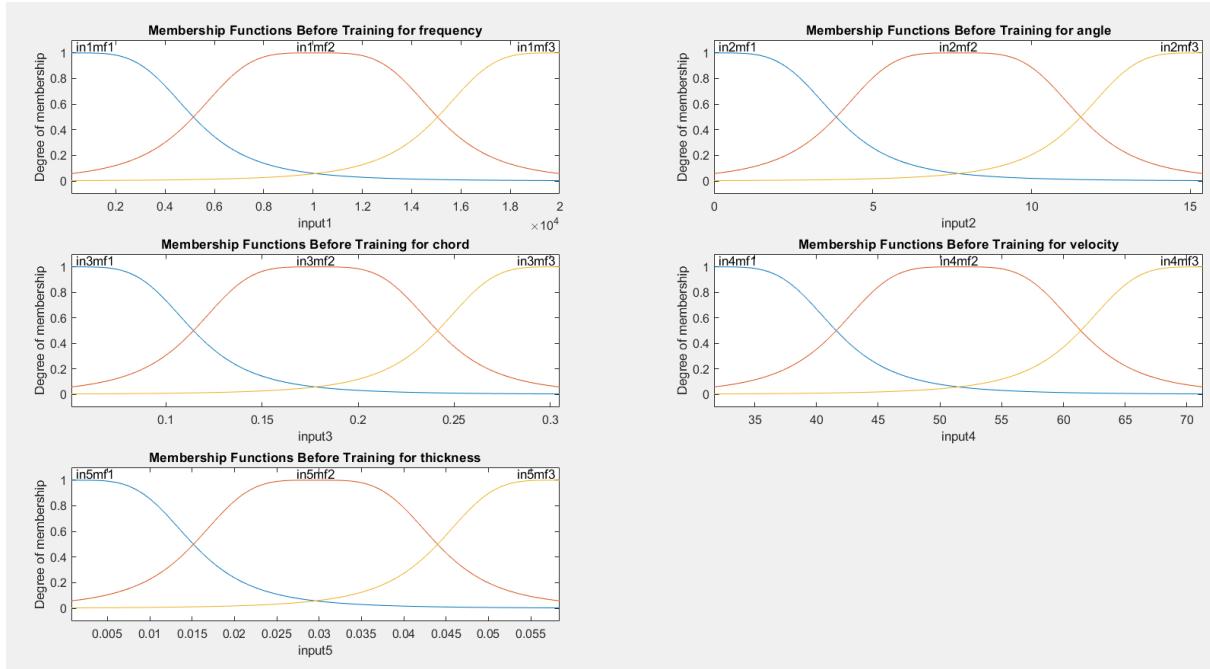
Error	RMSE	NMSE	NDEI	R2
TSK model 1	5.9544	0.6277	0.7923	0.3723

### 3.1.2 TSK model 2

Το μοντέλο 2 υλοποιείται με το script “Regression\_TSK\_model\_2.m”.

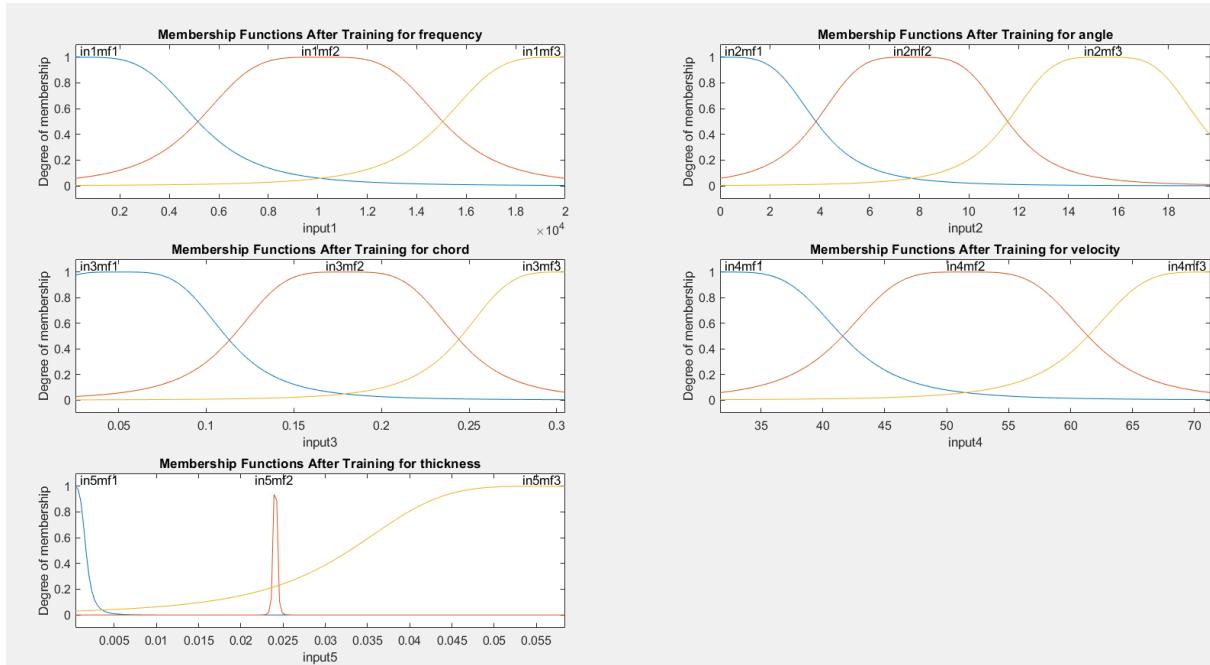
Χρησιμοποιεί μέθοδο ομαδοποίησης grid partition, 3 συναρτήσεις συμμετοχής “gbellmf” και “Singleton” output (constant).

## Συναρτήσεις Συμμετοχής Πριν την Εκπαίδευση



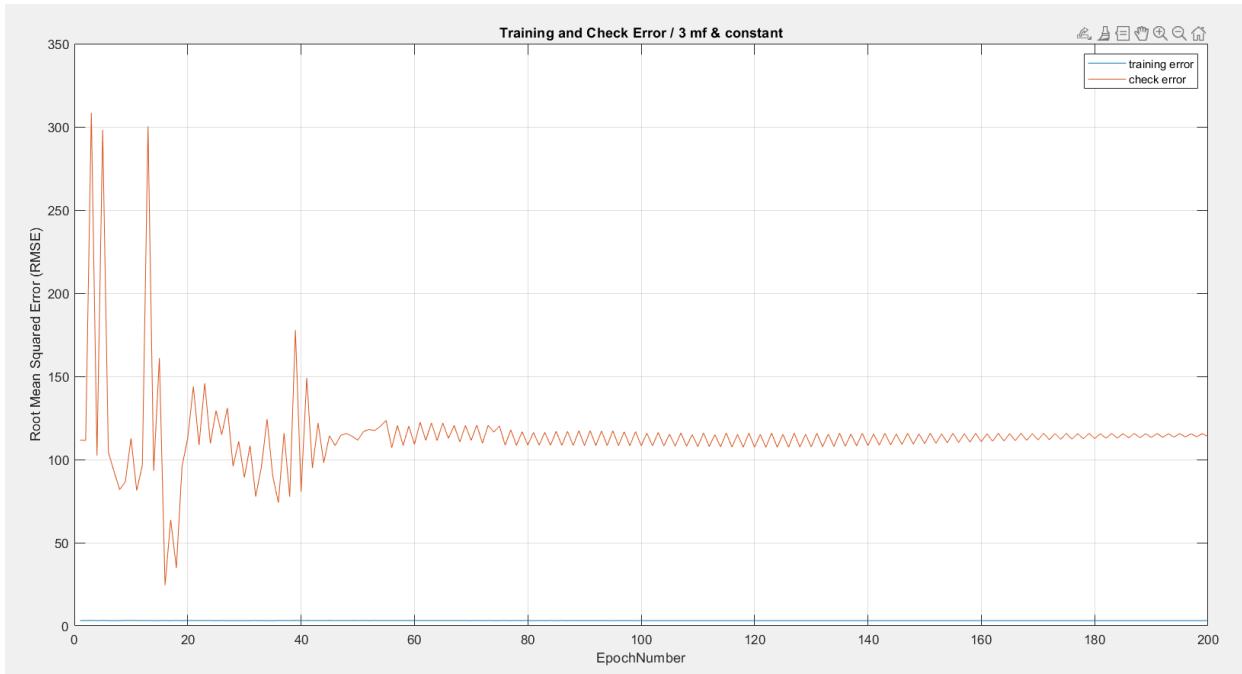
Σχήμα 71: mf Before Training

## Συναρτήσεις Συμμετοχής Μετά την Εκπαίδευση



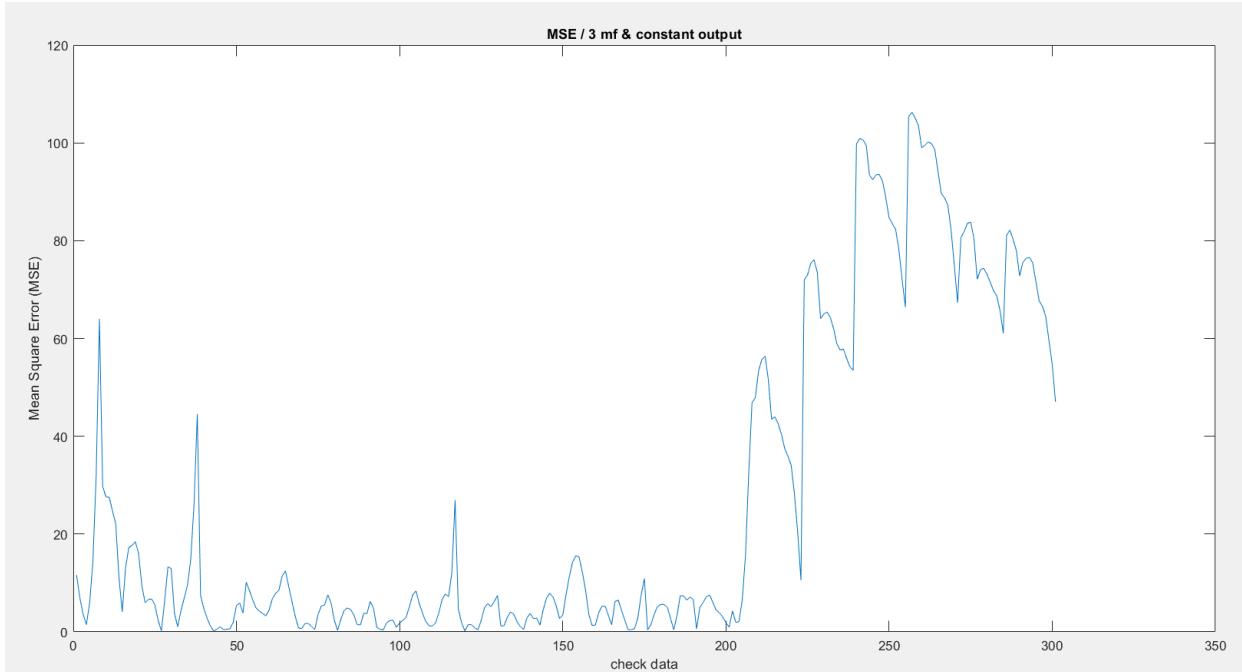
Σχήμα 72: mf After Training

## Καμπύλη Εκμάθησης



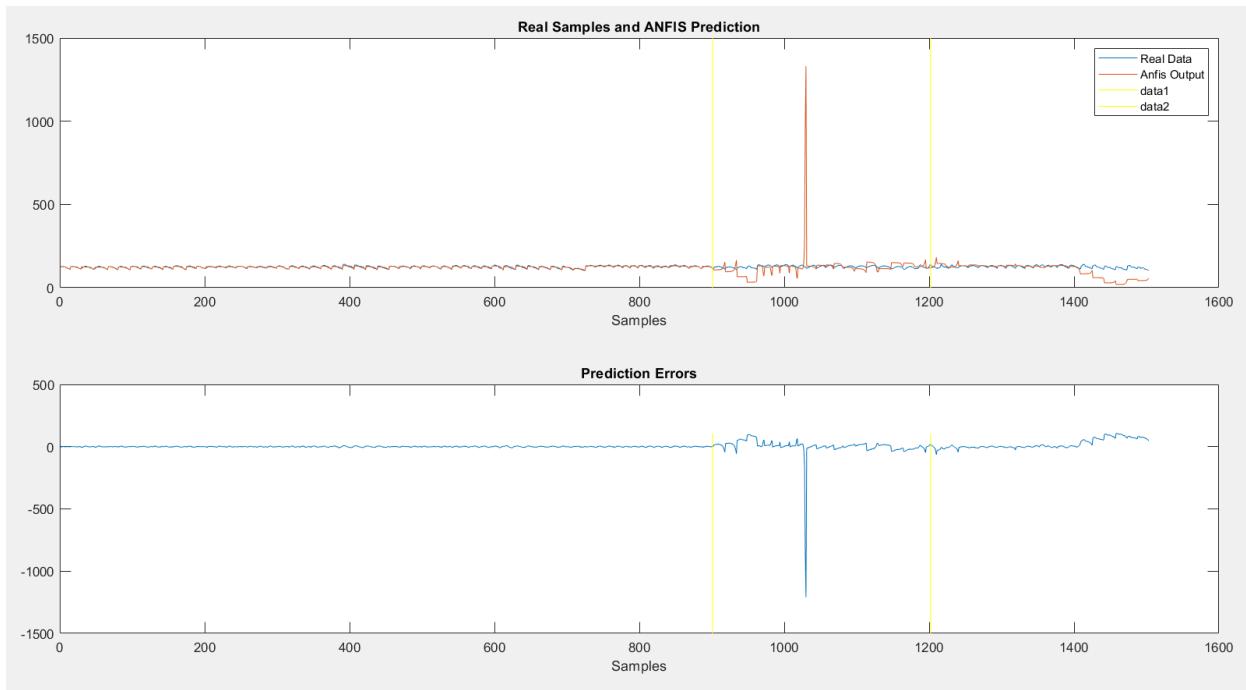
Σχήμα 73: Learning Curve

## RMSE



Σχήμα 74: RMSE

## Πραγματικές Τιμές VS Τιμές Πρόβλεψης & Σφάλμα Πρόβλεψης



Σχήμα 75: Real VS Prediction Values & Prediction Error

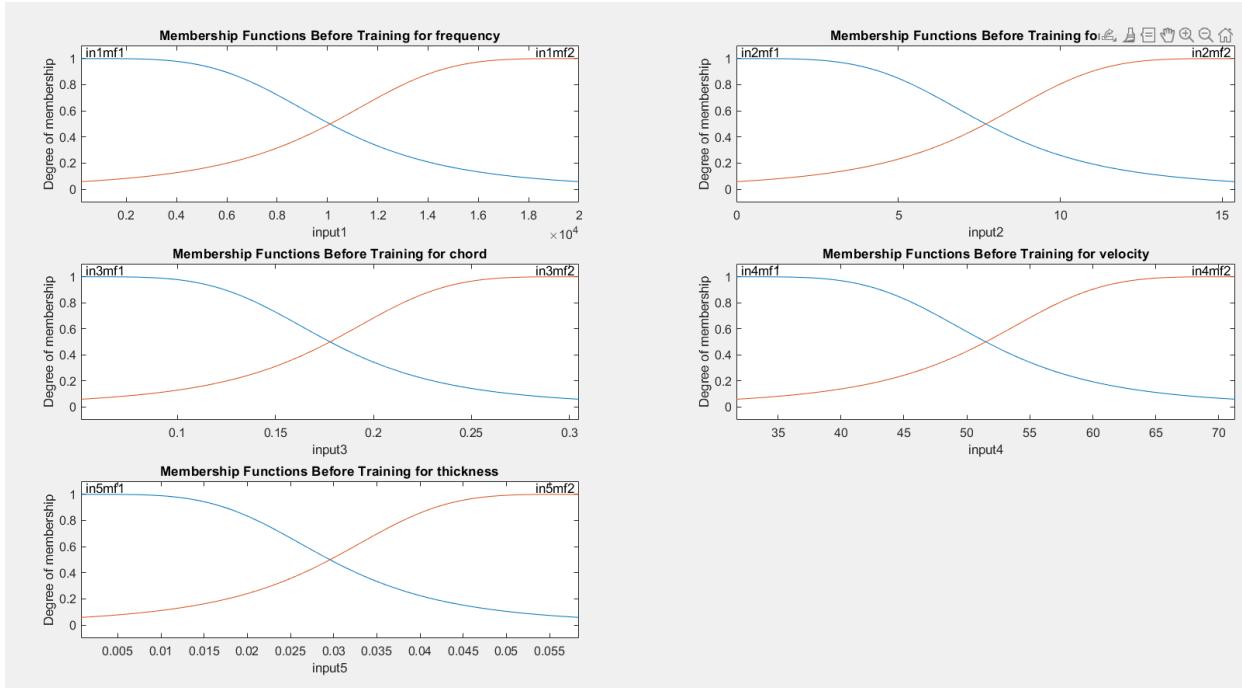
Error	RMSE	NMSE	NDEI	R2
TSK model 2	42.6229	32.1637	5.6713	-31.1637

### 3.1.3 TSK model 3

Το μοντέλο 3 υλοποιείται με το script “Regression\_TSK\_model\_3.m”.

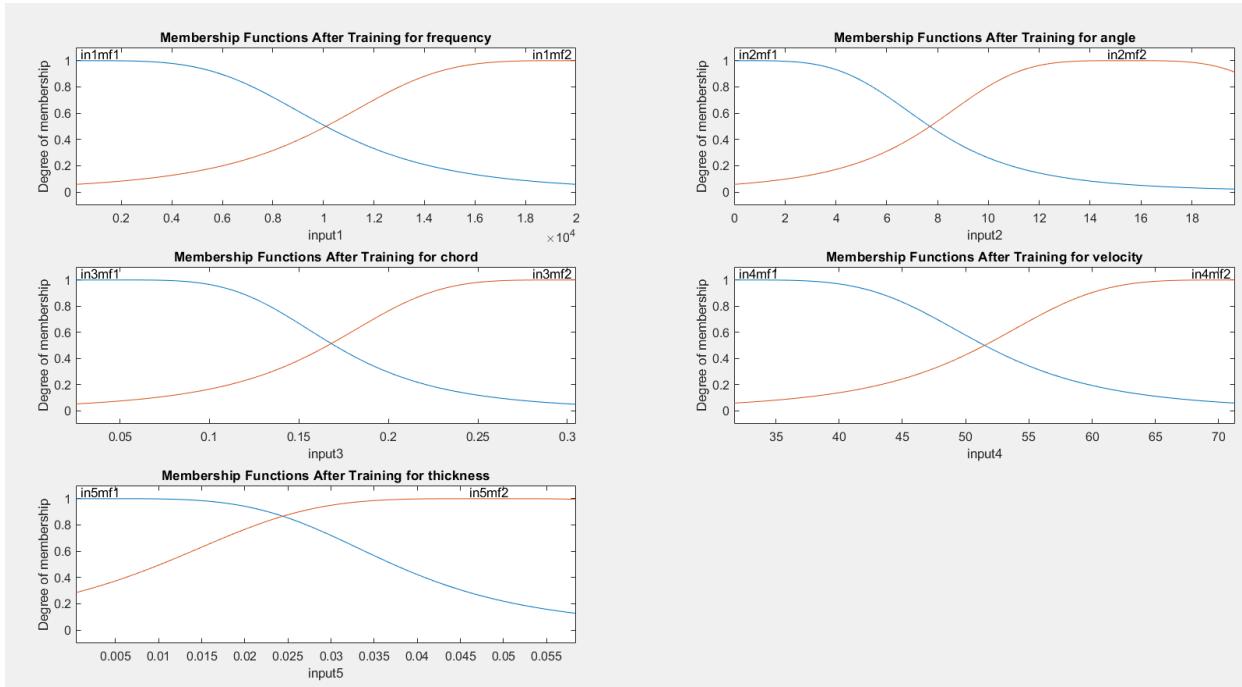
Χρησιμοποιεί μέθοδο ομαδοποίησης grid partition, 2 συναρτήσεις συμμετοχής “gbellmf” και “Polynomial” output (linear).

## Συναρτήσεις Συμμετοχής Πριν την Εκπαίδευση



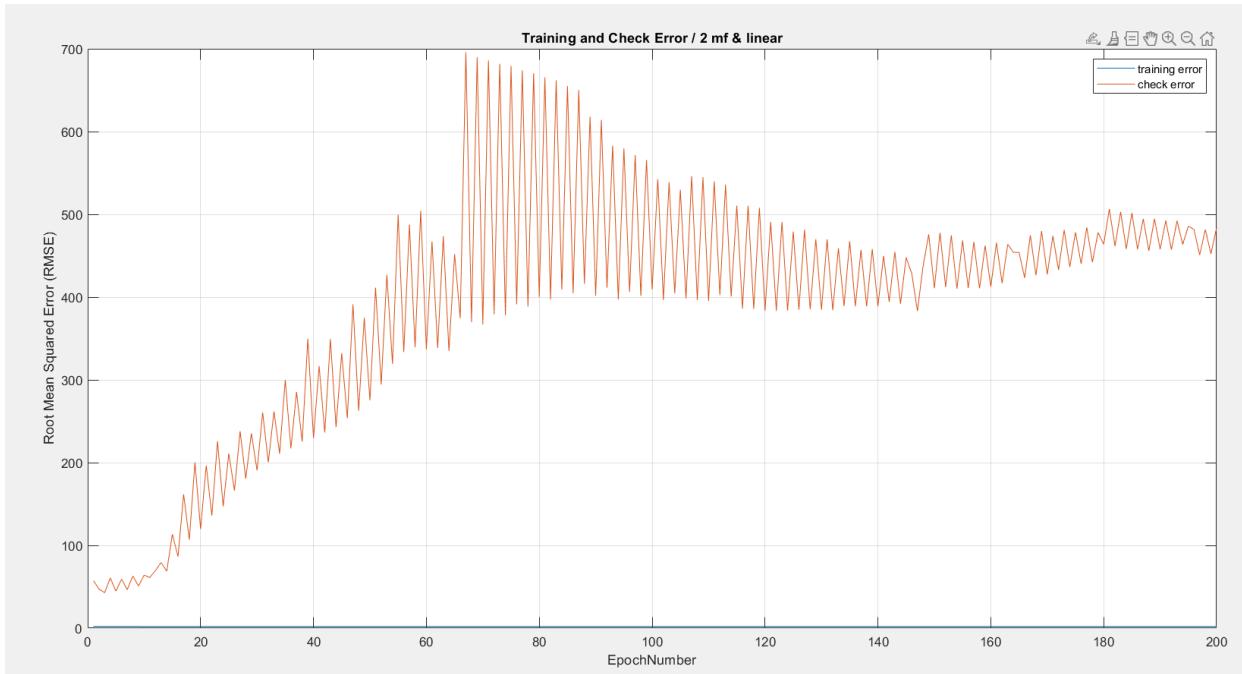
Σχήμα 76: mf Before Training

## Συναρτήσεις Συμμετοχής Μετά την Εκπαίδευση



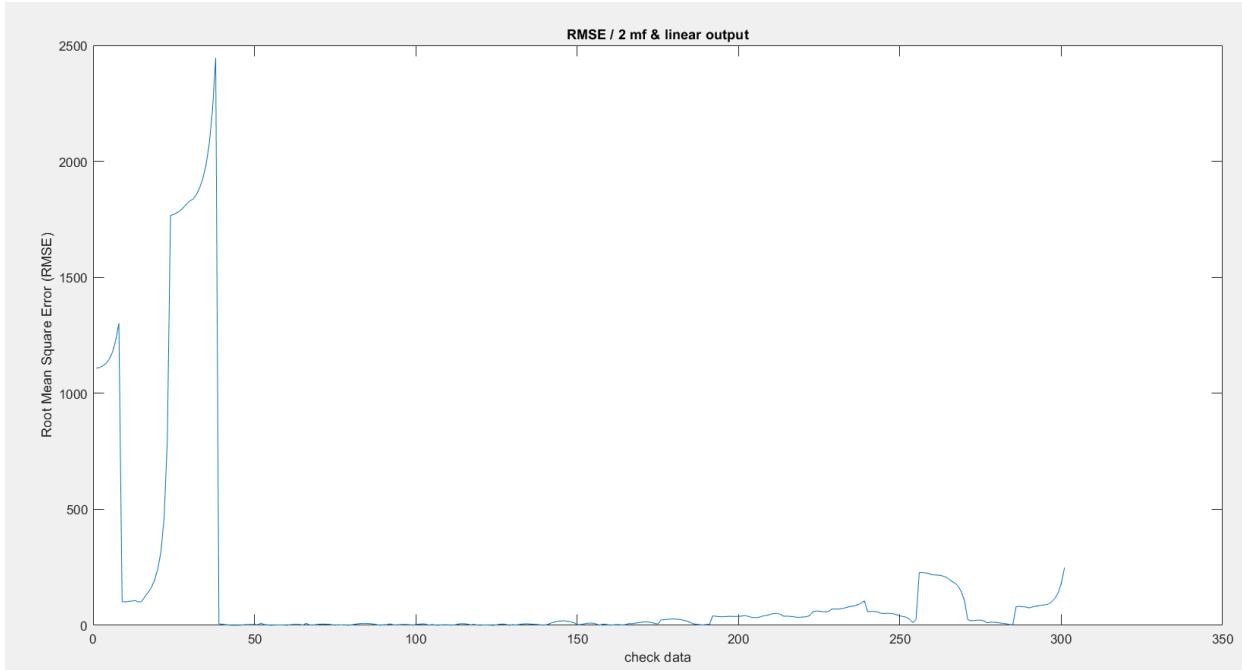
Σχήμα 77: mf After Training

## Καμπύλη Εκμάθησης



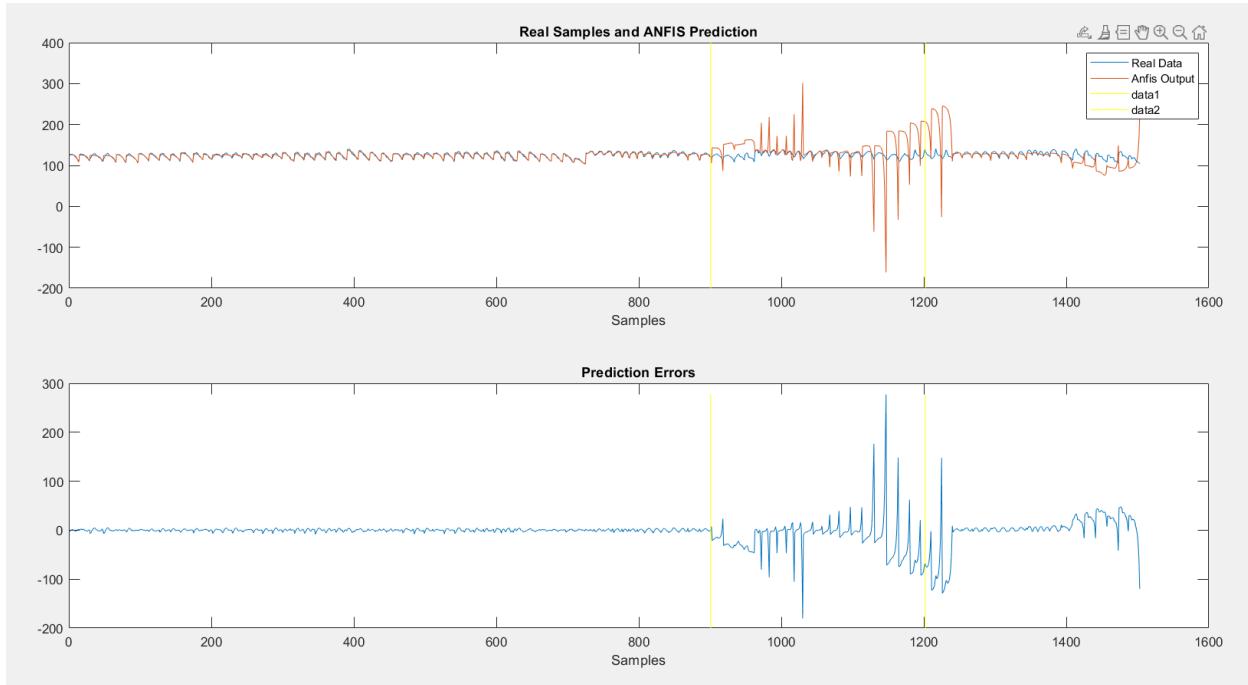
Σχήμα 78: Learning Curve

## RMSE



Σχήμα 79: RMSE

## Πραγματικές Τιμές VS Τιμές Πρόβλεψης & Σφάλμα Πρόβλεψης



Σχήμα 80: Real VS Prediction Values & Prediction Error

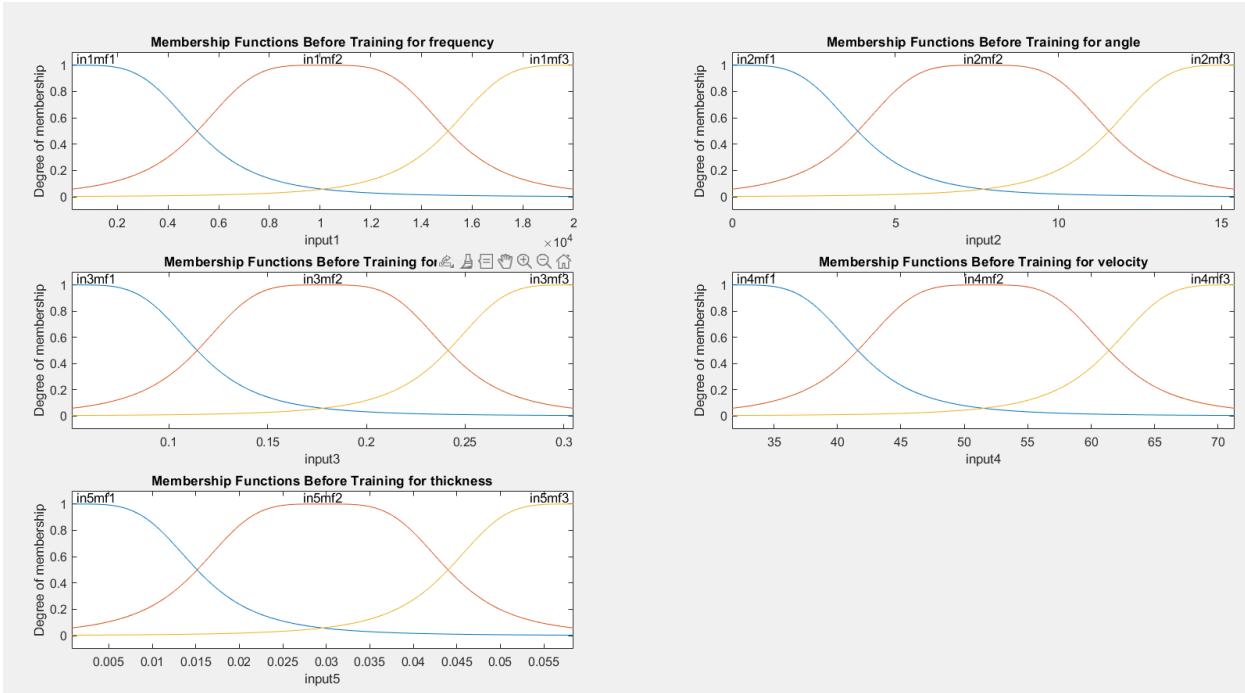
Error	RMSE	NMSE	NDEI	R2
TSK model 3	478.1159	4047.1	63.6170	-4046.1

### 3.1.4 TSK model 4

Το μοντέλο 4 υλοποιείται με το script “Regression\_TSK\_model\_3.m”.

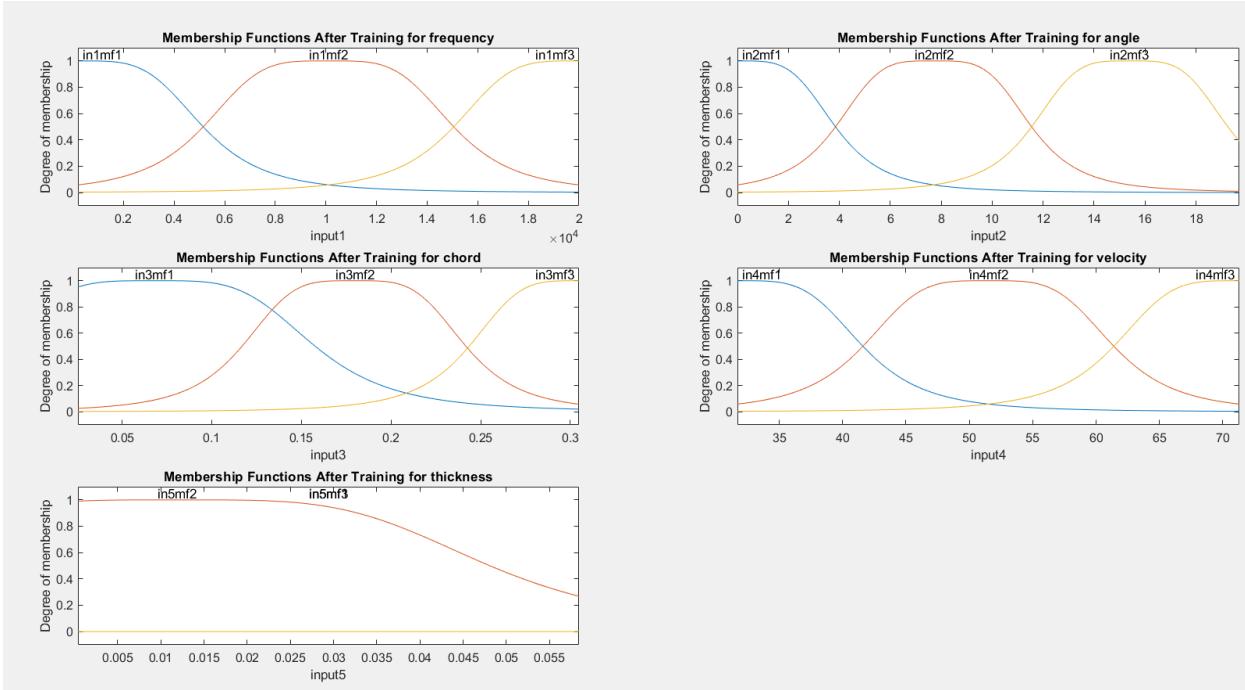
Χρησιμοποιεί μέθοδο ομαδοποίησης grid partition, 3 συναρτήσεις συμμετοχής “gbellmf” και “Polynomial” output (linear).

## Συναρτήσεις Συμμετοχής Πριν την Εκπαίδευση



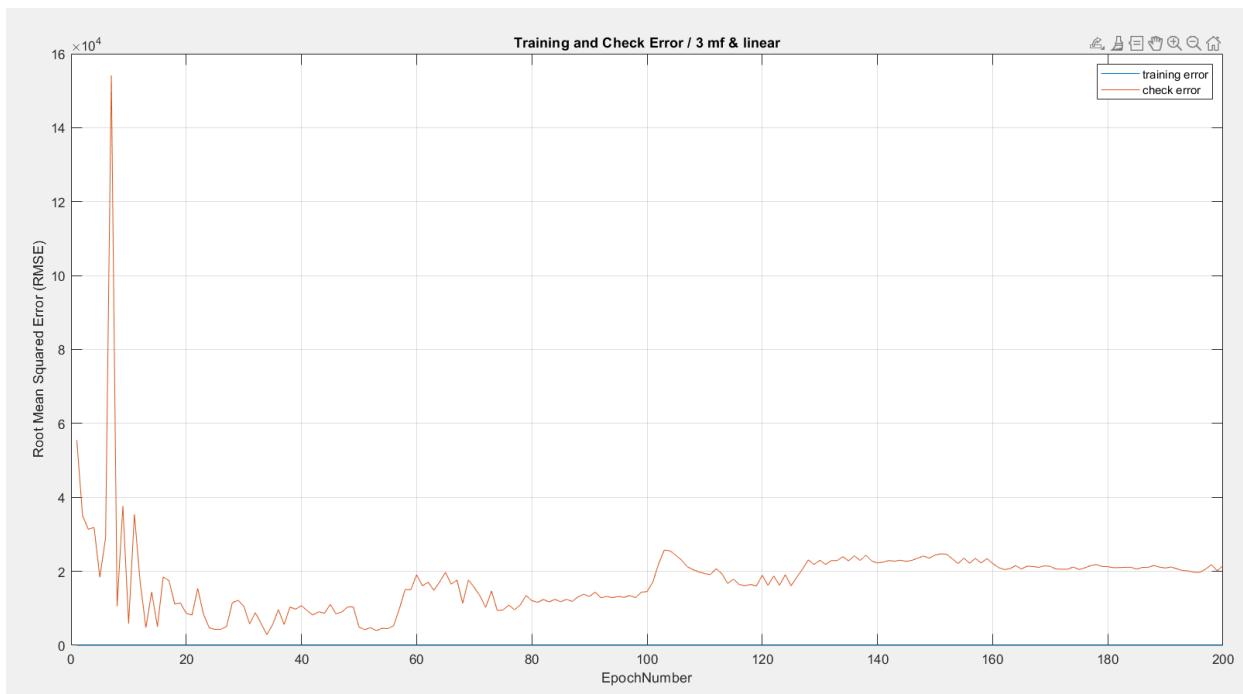
Σχήμα 81: mf Before Training

## Συναρτήσεις Συμμετοχής Μετά την Εκπαίδευση



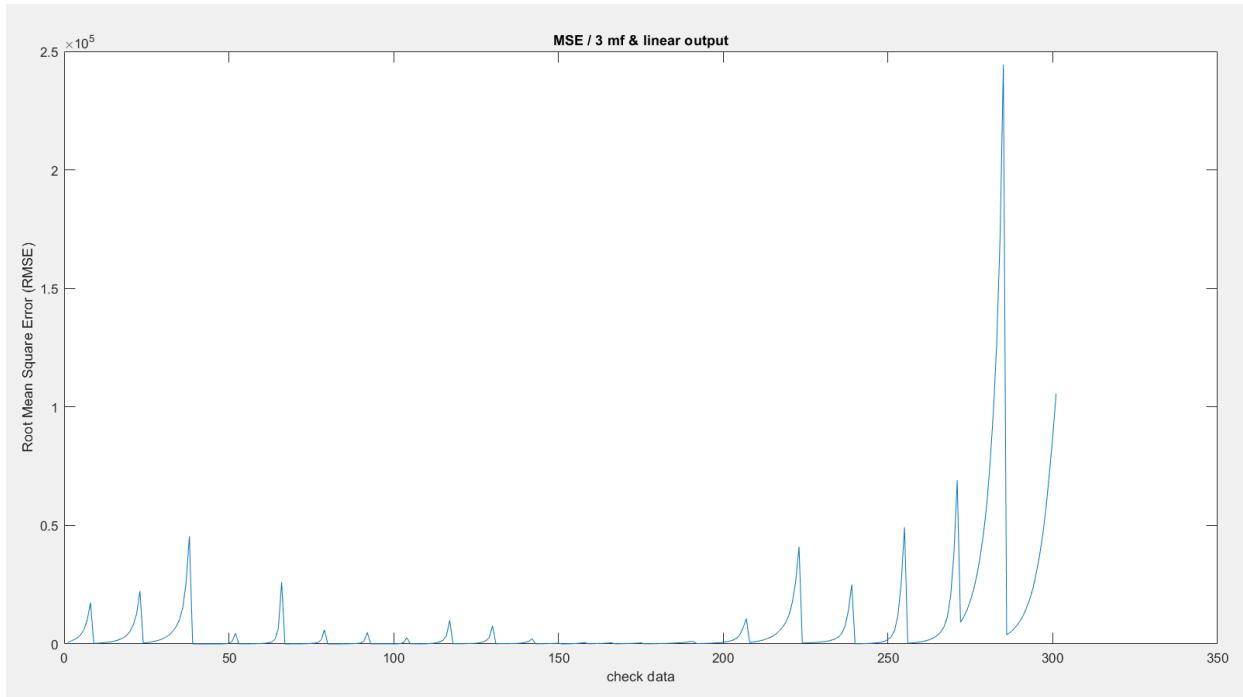
Σχήμα 82: mf After Training

## Καμπύλη Εκμάθησης



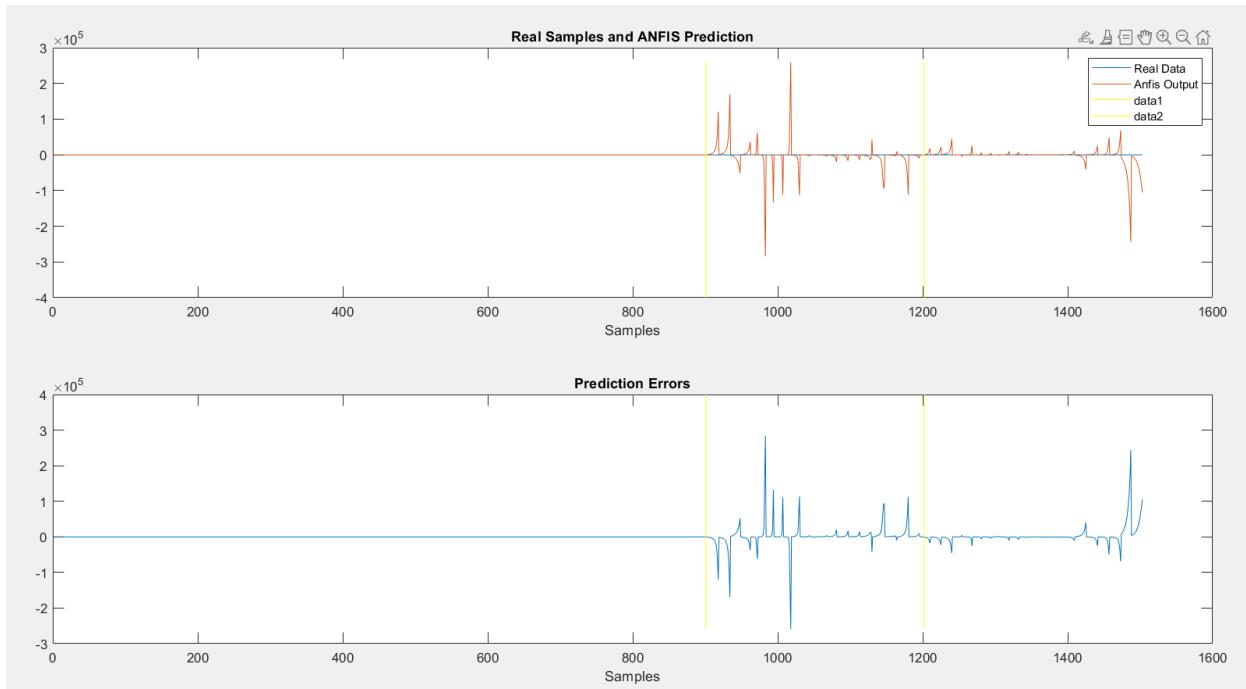
Σχήμα 83: Learning Curve

## RMSE



Σχήμα 84: RMSE

## Πραγματικές Τιμές VS Τιμές Πρόβλεψης & Σφάλμα Πρόβλεψης



Σχήμα 85: Real VS Prediction Values & Prediction Error

Error	RMSE	NMSE	NDEI	R2
TSK model 4	24685	10787850.9	3284.5	-10787849.9

### 3.1.5 Συμπεράσματα και Επιλογή μοντέλου Σφάλματα και των τεσσάρων μοντέλων.

Μοντέλο\Σφάλμα	RMSE	NMSE	NDEI	R2
TSK model 1	5.9544	0.6277	0.7923	0.3723
TSK model 2	42.6229	32.1637	5.6713	-31.1637
TSK model 3	478.1159	4047.1	63.6170	-4046.1
TSK model 4	24685	10787850.9	3284.5	-10787849.9

Το καλύτερο μοντέλο είναι το μοντέλο 1.

Το χειρότερο μοντέλο είναι το μοντέλο 4. Αν και το σφάλμα είναι ακραία μεγάλο, οπότε υπάρχει η πιθανότητα λάθους στον υπολογισμό.

Τα μοντέλα με Polynomial έξοδο παρουσιάζουμε μεμονωμένα μεγάλες τιμές Prediction Error, πράγμα που χαλάει την ακρίβεια του μοντέλου και μεγαλώνει το μέσο όρο του σφάλματος.

Τα μοντέλα φαίνεται από το Learning Curve να κάνουν υπερεκπαίδευση, αλλά στη συνέχεια σταθεροποιείται η καμπύλη. Τα μοντέλα με 3 mf είναι πιο επιφρεπή σε αυτή τη συμπεριφορά.

## 3.2 Part 2 - Εφαρμογή σε dataset με υψηλή διαστασιμότητα

Στο δεύτερο μέρος της εργασίας θα χρησιμοποιήσουμε το dataset “Superconductivity” από το UCI repository, το οποίο είναι dataset με υψηλό βαθμό διαστασιμότητας. Περιέχει 21263 δεδομένα και αποτελείται από 81 features και 1 output.

Είναι απαραίτητο να επιλέξουμε ένα μικρότερο σύνολο features με τα οποία θα εργαστούμε, διότι τα 81 είναι απαγορευτικά μεγάλο νούμερο, λογω της έκρηξης των κανόνων If-Then.

Το μοντέλο υλοποιείται στο script “Regression\_part2\_v2.m”.

Σύμφωνα με το script δοκιμάζουμε διαφορετικές τιμές για τον αριθμό των σημαντικότερων features που θα κρατήσουμε για να τροφοδοτήσουμε το μοντέλο, καθώς και διάφορες τιμές για την ακτίνα Ra των clusters. Χρησιμοποιούμε την τεχνική Grid Search και διαχωρίζουμε τα δεδομένα με την μέθοδο CV Partition k-fold με k=5.

Η μέθοδος ομαδοποίησης για την δημιουργία των κανόνων είναι η “Subtractive Clustering”, που υλοποιείται με την εντολή genfis2.

Κάθε μοντέλο εκπαιδεύεται με την μέθοδο anfis και χρησιμοποιούμε 100 EpochNumber.

Η επιλογή των σημαντικότερων features έγινε με την εντολή Relieff.

### 3.2.1 Αποτελέσματα Σφαλμάτων του Grid Search

- Πίνακας **RMSE** (Root Mean Square Error)

Features\Ra	0.2	0.4	0.6	0.8	1
5	16.4775	17.6595	18.7122	20.3321	20.4394
8	16.9144	16.8142	17.5359	20.1406	20.2999
11	36.3985	15.6254	16.6085	17.9002	18.1107
14	20.4471	15.2011	15.8688	17.2750	17.8301

- Πίνακας **NMSE** (Normalized Mean Square Error)

<b>Features\Ra</b>	<b>0.2</b>	<b>0.4</b>	<b>0.6</b>	<b>0.8</b>	<b>1</b>
<b>5</b>	0.2314	0.2658	0.2985	0.3523	0.3560
<b>8</b>	0.2459	0.2411	0.2621	0.3457	0.3512
<b>11</b>	2.6399	0.2081	0.2351	0.2732	0.2796
<b>14</b>	0.4280	0.1969	0.2146	0.2544	0.2710

- Πίνακας **NDEI** (Root of NMSE)

<b>Features\Ra</b>	<b>0.2</b>	<b>0.4</b>	<b>0.6</b>	<b>0.8</b>	<b>1</b>
<b>5</b>	0.4810	0.5155	0.5462	0.5935	0.5967
<b>8</b>	0.4938	0.4908	0.5119	0.5879	0.5926
<b>11</b>	1.0618	0.4561	0.4848	0.5225	0.5287
<b>14</b>	0.5970	0.4437	0.4632	0.5043	0.5205

- Πίνακας **R2** (Coefficient of Determination)

<b>Features\Ra</b>	<b>0.2</b>	<b>0.4</b>	<b>0.6</b>	<b>0.8</b>	<b>1</b>
<b>5</b>	0.7685	0.7341	0.7014	0.6476	0.6439
<b>8</b>	0.7540	0.7588	0.7378	0.6542	0.6487
<b>11</b>	-1.6399	0.7918	0.7648	0.7267	0.7203
<b>14</b>	0.5719	0.8030	0.7853	0.7455	0.7289

Σύμφωνα με τα παραπάνω αποτελέσματα πετυχαίνουμε το ελάχιστο σφάλμα

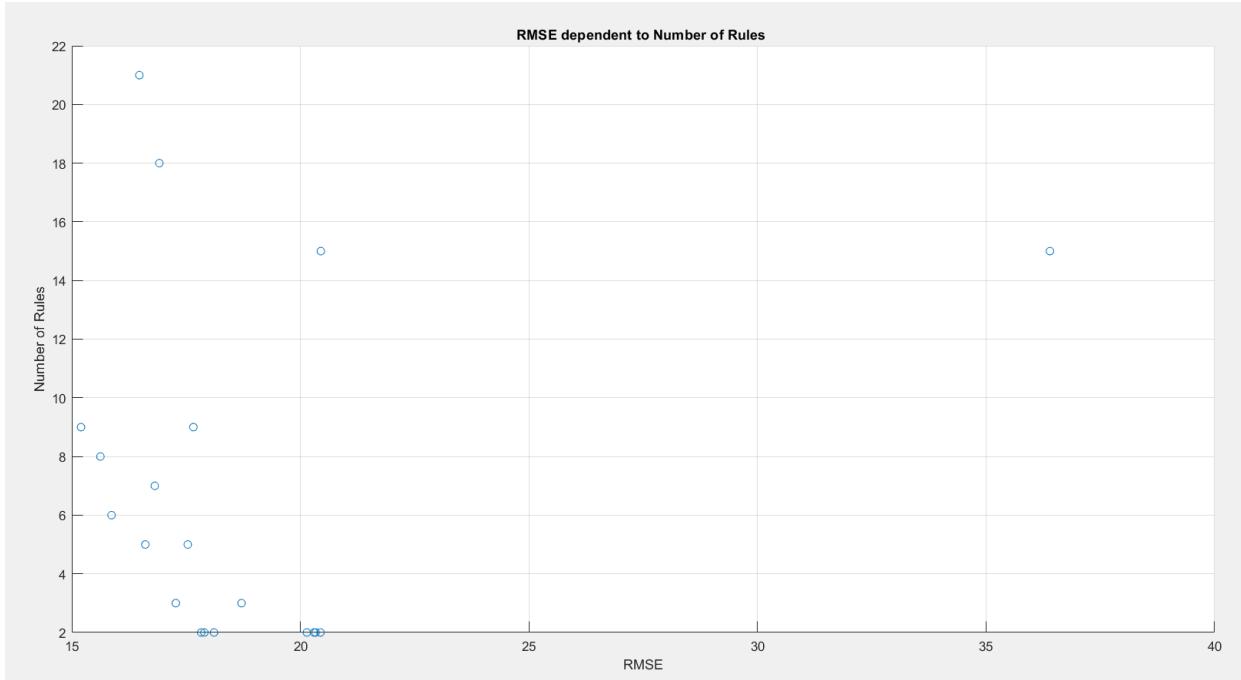
**RMSE = 15.2011** με παραμέτρους:

- features = 14
- rules = 9
- radius = 0.4

### 3.2.2 Διαγράμματα Grid Search

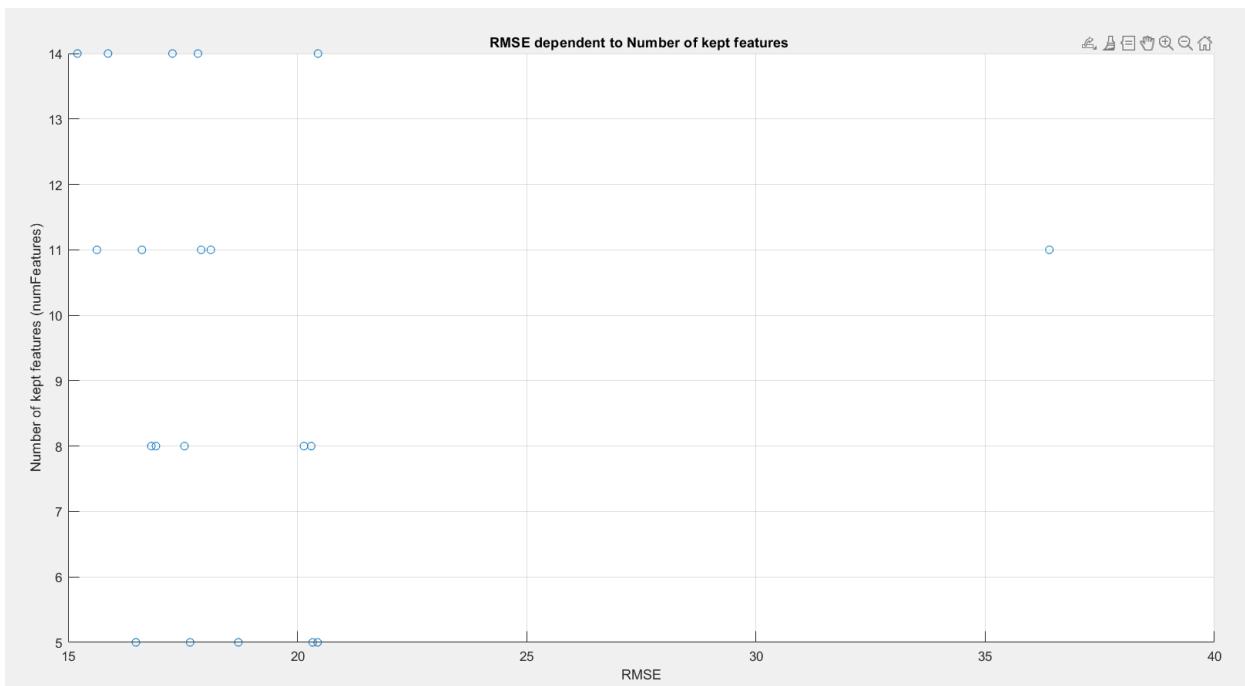
Παρακάτω φαίνονται τα διαγράμματα scatters για το σφάλμα RMSE συναρτήσει του αριθμού των σημαντικότερων features, του αριθμού των κανόνων και της ακτίνας Ra.

#### RMSE – Number of Rules



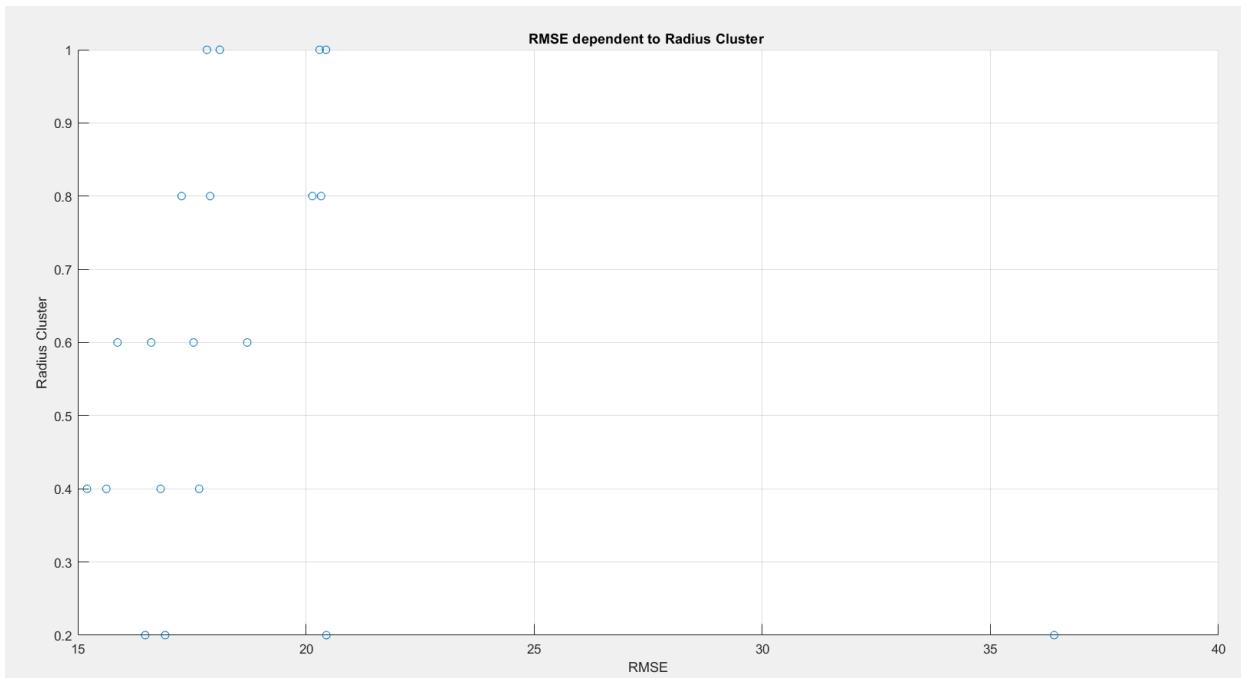
Σχήμα 86: RMSE συναρτήσει του Αριθμού των Κανόνων

## RMSE – Number of Features



Σχήμα 87: RMSE συναρτήσει του Αριθμού των Features

## RMSE – Radius Ra



Σχήμα 88: RMSE συναρτήσει της Ακτίνας Ra

Συμπεραίνουμε ότι το σφάλμα RMSE μεγαλώνει, όσο αυξάνεται η ακτίνα Ra, θεωρόντας σταθερό τον αριθμό των features. Με μικρή ακτίνα το μοντέλο εκπαιδεύεται με καλύτερη ακρίβεια και κάνει καλύτερες προβλέψεις. Αυτό όμως ισχύει μέχρι ένα σημείο, διότι με πολύ μικρή ακτίνα το μοντέλο κάνει overfitting και δεν μπορεί να εφαρμώσει αυτά που έμαθε και να τα γενικεύσει στο checking dataset. Γι αυτό το λόγο η βέλτιστη ακτίνα είναι η 0.4 και όχι η 0.2, όπως μας επιβεβαιώνει και το script “Regression\_part2\_v2.m”.

Επίσης, παρατηρούμε ότι το σφάλμα RMSE μειώνεται όσο αυξάνουμε τον αριθμό των features που χρησιμοποιούμε. Αυτό είναι λογικό, διότι τροφοδοτούμε το μοντέλο με περισσότερη πληροφορία. Όμως, δεν μπορούμε να το αυξήσουμε παραπάνω, γιατί ο χρόνος εκπαίδευσης αυξάνει εκθετικά και το μοντέλο θα γίνει μη λειτουργικό.

### 3.3.3 Βέλτιστο Μοντέλο

Το βέλτιστο μοντέλο επιλέχθηκε από το script “Regression\_part2\_v2.m” και υλοποιείται στο script “Optimal\_TSK\_model.m”.

Αποτελείται από 8 κανόνες, 14 features και 0.4 ακτίνα.

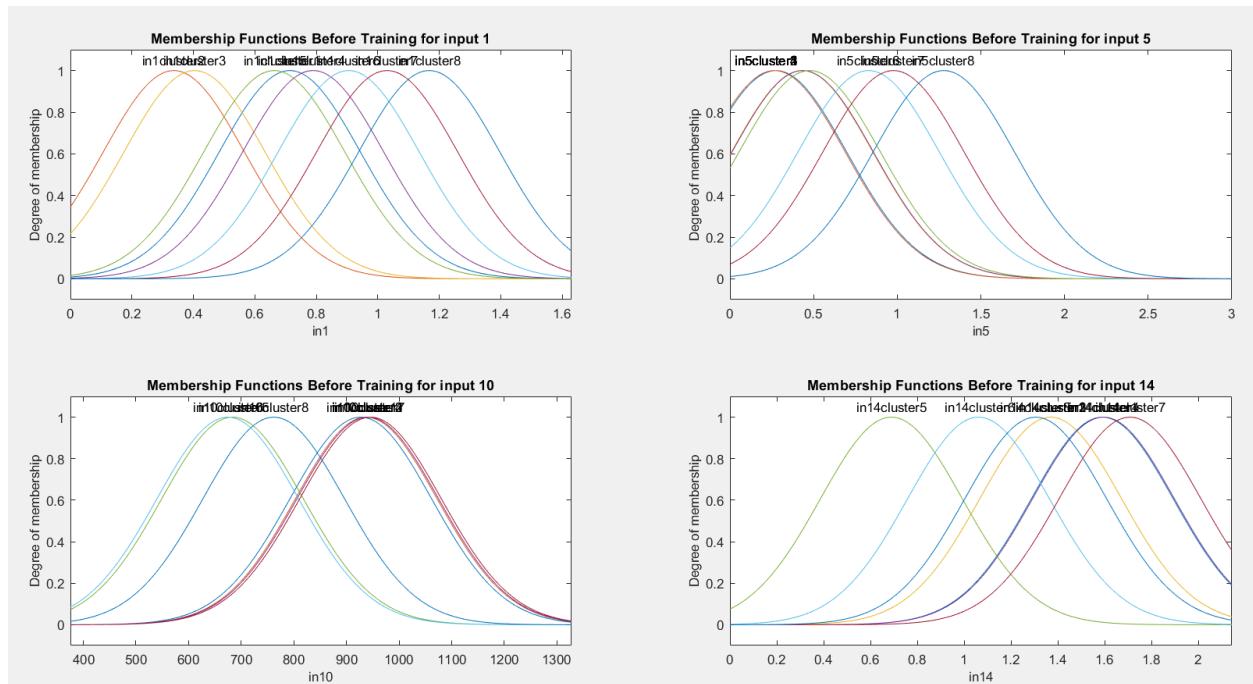
- Πίνακας Σφαλμάτων Βέλτιστου Μοντέλου:

Error	RMSE	NMSE	NDEI	R2
Optimal Model	15.1346	0.1952	0.4418	0.8047

Παρατηρούμε ότι πετυχαίνουμε πολύ ικανοποιητικά αποτελέσματα.

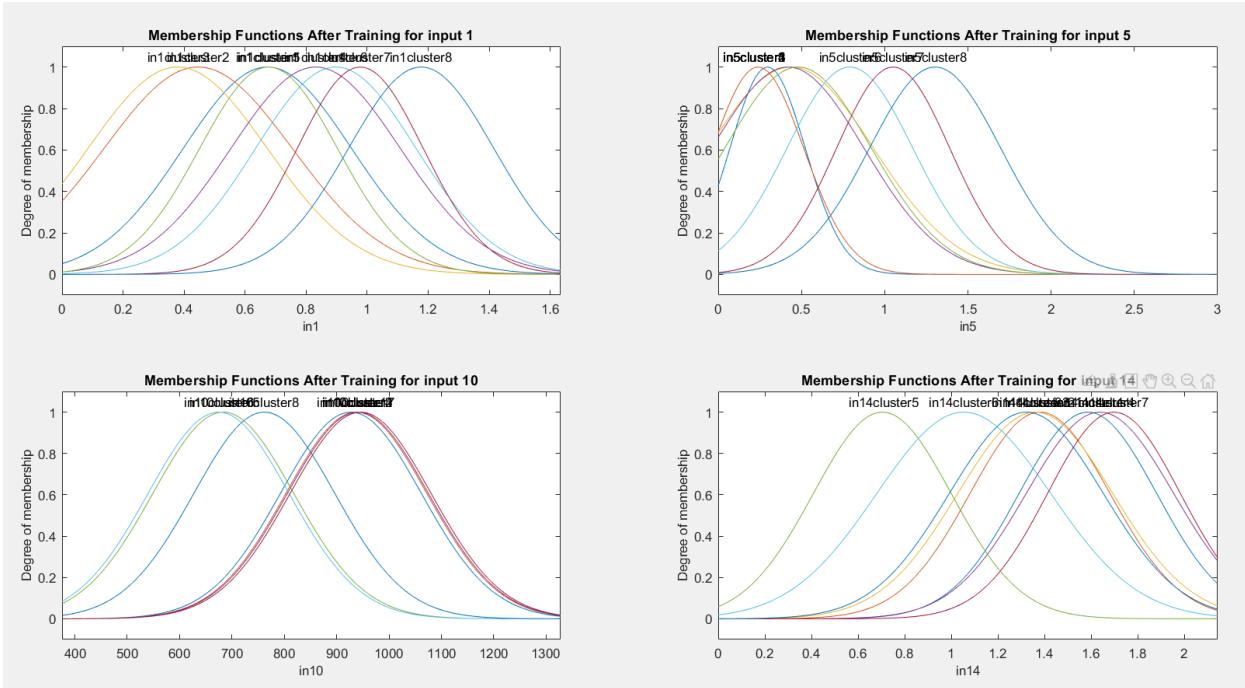
- Διαγράμματα Βέλτιστου Μοντέλου:

Συναρτήσεις Συμμετοχής πριν την εκπαίδευση



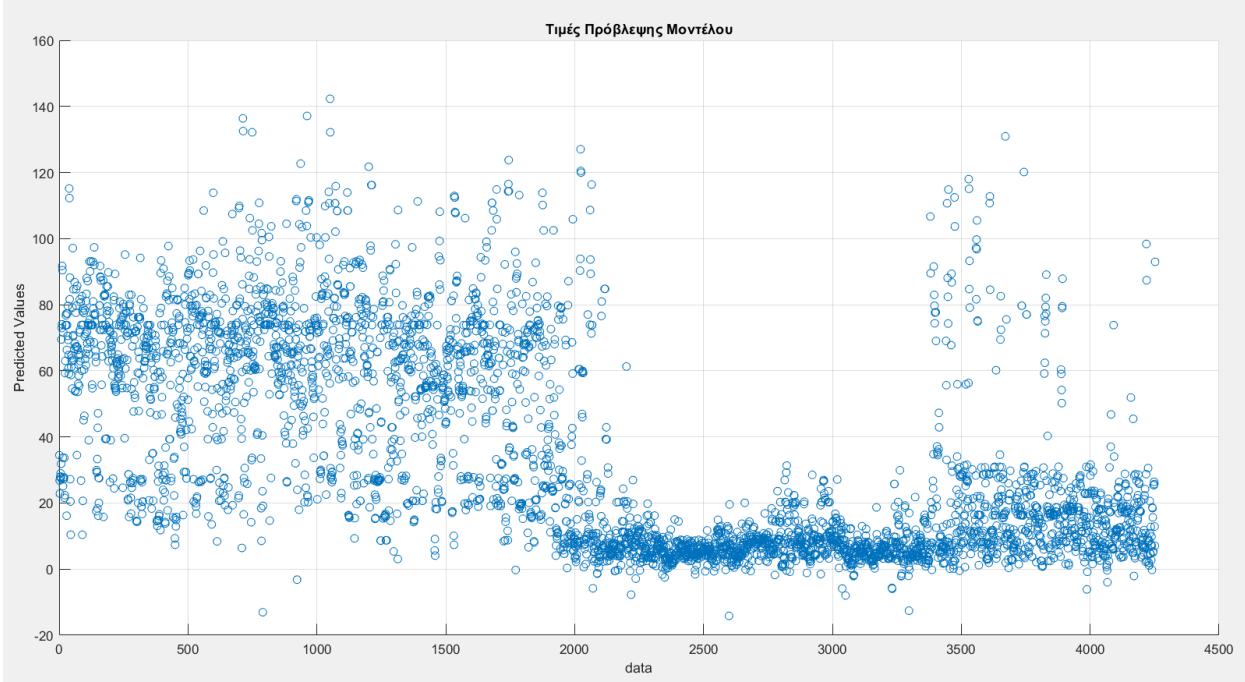
Σχήμα 89: mf Before Training

## Συναρτήσεις Συμμετοχής μετά την εκπαίδευση



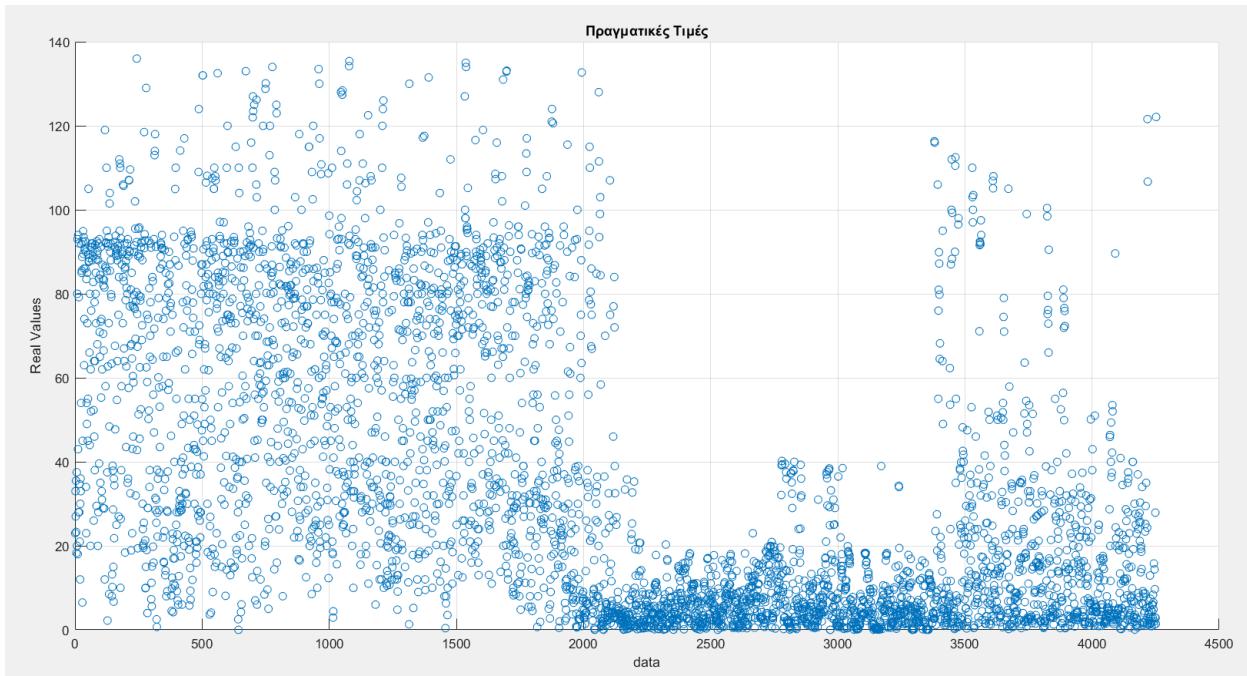
Σχήμα 90: mf After Training

## Τιμές Πρόβλεψης



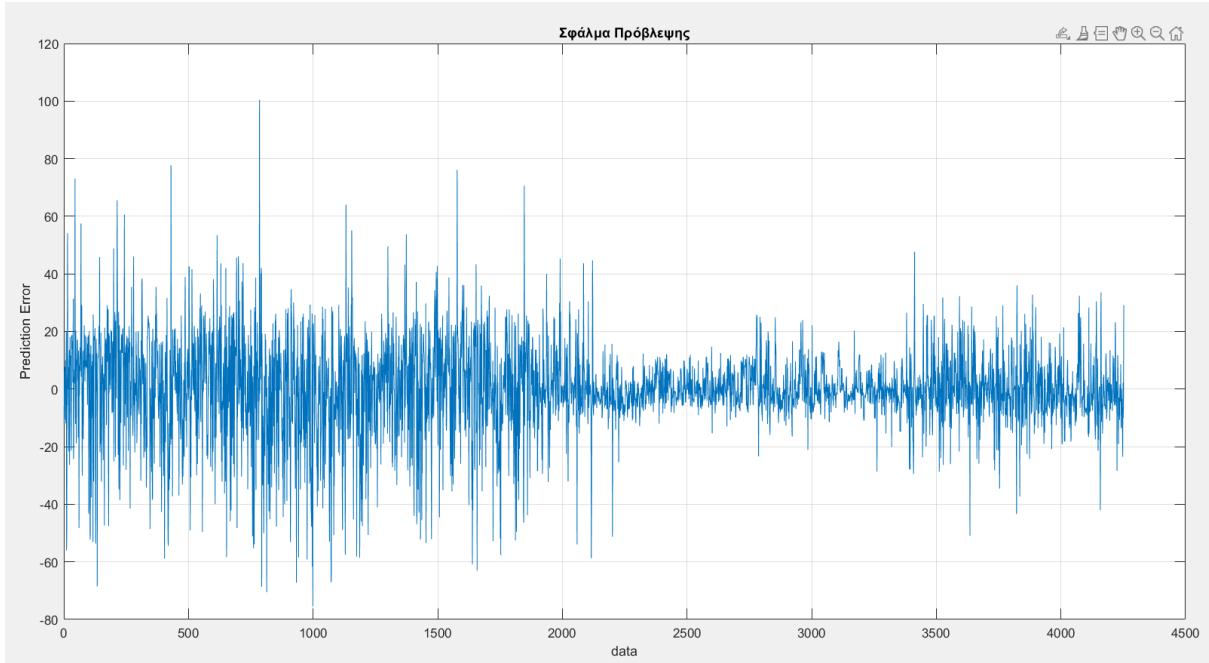
Σχήμα 91: Prediction Values

## Πραγματικές Τιμές



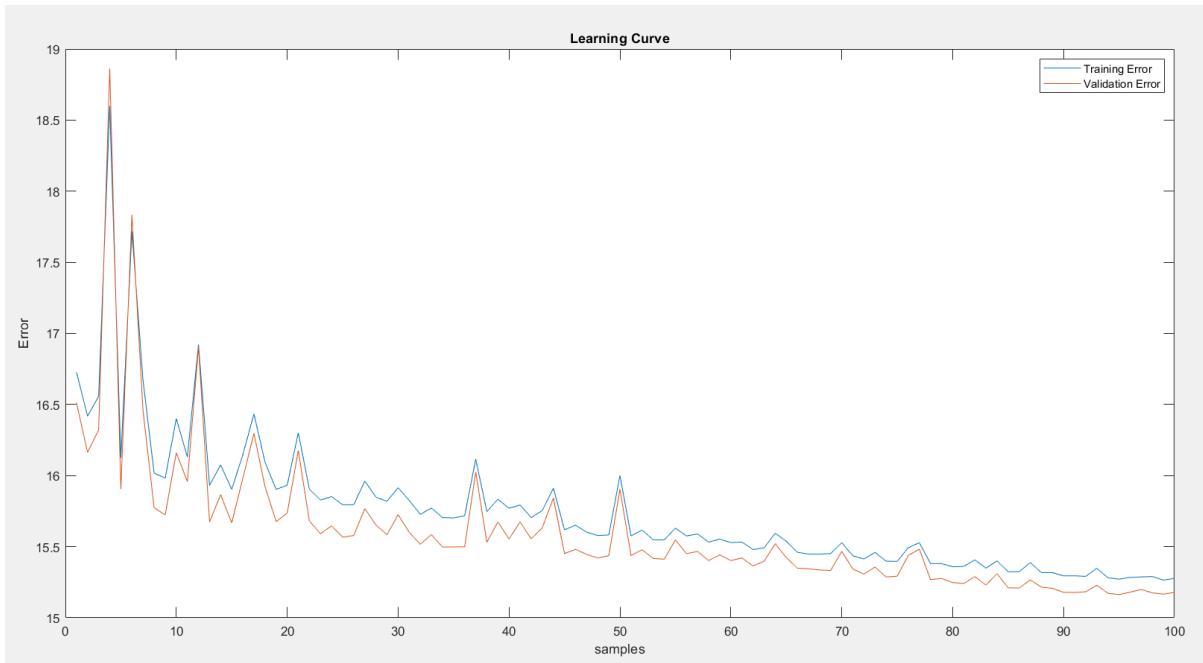
Σχήμα 92: Real Values

## Σφάλμα Πρόβλεψης



Σχήμα 93: Prediction Error

## Καμπύλη Εκμάθησης



Σχήμα 94: Learning Curve

- Οι κανόνες εκπαίδευσης είναι 8 και φαίνονται από την εντολή trainedFis.Rules:

```

1 "in1==in1cluster1 & in2==in2cluster1 & in3==in3cluster1 & in4==in4cluster1 &
in5==in5cluster1 & in6==in6cluster1 & in7==in7cluster1 & in8==in8cluster1 & in9==in9cluster1
& in10==in10cluster1 & in11==in11cluster1 & in12==in12cluster1 & in13==in13cluster1 &
in14==in14cluster1 => out1=out1cluster1 (1)"

2 "in1==in1cluster2 & in2==in2cluster2 & in3==in3cluster2 & in4==in4cluster2 &
in5==in5cluster2 & in6==in6cluster2 & in7==in7cluster2 & in8==in8cluster2 & in9==in9cluster2
& in10==in10cluster2 & in11==in11cluster2 & in12==in12cluster2 & in13==in13cluster2 &
in14==in14cluster2 => out1=out1cluster2 (1)"

3 "in1==in1cluster3 & in2==in2cluster3 & in3==in3cluster3 & in4==in4cluster3 &
in5==in5cluster3 & in6==in6cluster3 & in7==in7cluster3 & in8==in8cluster3 & in9==in9cluster3
& in10==in10cluster3 & in11==in11cluster3 & in12==in12cluster3 & in13==in13cluster3 &
in14==in14cluster3 => out1=out1cluster3 (1)"

4 "in1==in1cluster4 & in2==in2cluster4 & in3==in3cluster4 & in4==in4cluster4 &
in5==in5cluster4 & in6==in6cluster4 & in7==in7cluster4 & in8==in8cluster4 & in9==in9cluster4
& in10==in10cluster4 & in11==in11cluster4 & in12==in12cluster4 & in13==in13cluster4 &
in14==in14cluster4 => out1=out1cluster4 (1)"

5 "in1==in1cluster5 & in2==in2cluster5 & in3==in3cluster5 & in4==in4cluster5 &
in5==in5cluster5 & in6==in6cluster5 & in7==in7cluster5 & in8==in8cluster5 & in9==in9cluster5
& in10==in10cluster5 & in11==in11cluster5 & in12==in12cluster5 & in13==in13cluster5 &
in14==in14cluster5 => out1=out1cluster5 (1)"

6 "in1==in1cluster6 & in2==in2cluster6 & in3==in3cluster6 & in4==in4cluster6 &
in5==in5cluster6 & in6==in6cluster6 & in7==in7cluster6 & in8==in8cluster6 & in9==in9cluster6
& in10==in10cluster6 & in11==in11cluster6 & in12==in12cluster6 & in13==in13cluster6 &
in14==in14cluster6 => out1=out1cluster6 (1)"

7 "in1==in1cluster7 & in2==in2cluster7 & in3==in3cluster7 & in4==in4cluster7 &
in5==in5cluster7 & in6==in6cluster7 & in7==in7cluster7 & in8==in8cluster7 & in9==in9cluster7
& in10==in10cluster7 & in11==in11cluster7 & in12==in12cluster7 & in13==in13cluster7 &
in14==in14cluster7 => out1=out1cluster7 (1)"

8 "in1==in1cluster8 & in2==in2cluster8 & in3==in3cluster8 & in4==in4cluster8 &
in5==in5cluster8 & in6==in6cluster8 & in7==in7cluster8 & in8==in8cluster8 & in9==in9cluster8
& in10==in10cluster8 & in11==in11cluster8 & in12==in12cluster8 & in13==in13cluster8 &
in14==in14cluster8 => out1=out1cluster8 (1)"

```

- Συμπεράσματα

Από τα διαγράμματα παρατηρούμε ότι το μοντέλο προβλέπει σε ικανοποιητικό βαθμό τις πραγματικές τιμές.

Αν χρησιμοποιούσαμε Grid Partition αντί για Subtractive Clustering και υποθέσουμε ότι για κάθε είσοδο είχαμε 3 Ασαφή Σύνολα, αυτό σημαίνει ότι το μοντέλο θα χρειαζόταν  $3^{14}$  κανόνες, έναντι των 8 που χρησιμοποιούμε σε αυτή την εργασία.

# 4. Εργασία #4: Classification

## 4.1 Part 1 - Εφαρμογή σε απλό dataset

Σε αυτή την εργασία μελετάμε την ικανότητα των TSK μοντέλων (Takagi Sugeno Kang) στην ταξινόμηση και ομαδοποίηση δεδομένων σε classes, καθώς και στην αξιολόγηση των μοντέλων με διάφορες μετρικές.

Δημιουργούμε 4 μοντέλα:

Model	Ακτίνα των Clusters	Class
TSK model 1	0.1	Dependent
TSK model 2	0.9	Dependent
TSK model 3	0.1	Independent
TSK model 4	0.9	Independent

Το dataset που θα χρησιμοποιήσουμε είναι το “Haberman’s Survival ” από το UCI repository, το οποίο περιέχει 306 δείγματα και αποτελείται από 3 features και 1 output.

Τα μοντέλα υλοποιούνται στο matlab script “classification\_Haberman.m”.

Χωρίζουμε το dataset σε τρία υποσύνολα, 60% training data, 20% validation data, 20% checking data.

Εκπαιδεύουμε τέσσερα TSK μοντέλα.

- Στα 1 και 2 ο διαχωρισμός του χώρου του input dataset γίνεται με subtractive clustering στα δεδομένα του συνόλου εκπαίδευσης, που ανήκουν στην εκάστοτε κλάση ξεχωριστά (**Class Dependent**).
- Στα 3 και 4 ο διαχωρισμός του χώρου του input dataset γίνεται με subtractive clustering σε όλα τα δεδομένα του συνολικού training dataset (**Class Independent**) και η εκπαίδευση υλοποιείται με genfis2 και anfis training.

Για κάθε Dependent και Independent μοντέλο, έχουμε δύο μοντέλα που υλοποιούνται με διαφορετικό αριθμό κανόνων. Η τιμή του αριθμού των κανόνων καθορίζεται από την τιμή της ακτίνας των clusters. Επιλέγουμε δύο ακραίες τιμές [0.1 0.9], ώστε ο αριθμός των κανόνων να αλλάζει σε μεγάλο βαθμό.

Για την εκπαίδευση του μοντέλου χρησιμοπούμε τη συνάρτηση anfis και επιλέγουμε EpochNumber = 100.

Επίσης, δημιουργούμε διαγράμματα για τις τελικές εκβάσεις των μοντέλων και των Συναστήσεων Συμμετοχής, για την καμπύλη εκμάθησης και για το σφάλμα πρόβλεψης.

Οι είσοδοι των μοντέλων είναι:

- age (numerical)
- year – 1900 (numerical)
- nodes (numerical)

ενώ η έξοδος είναι :

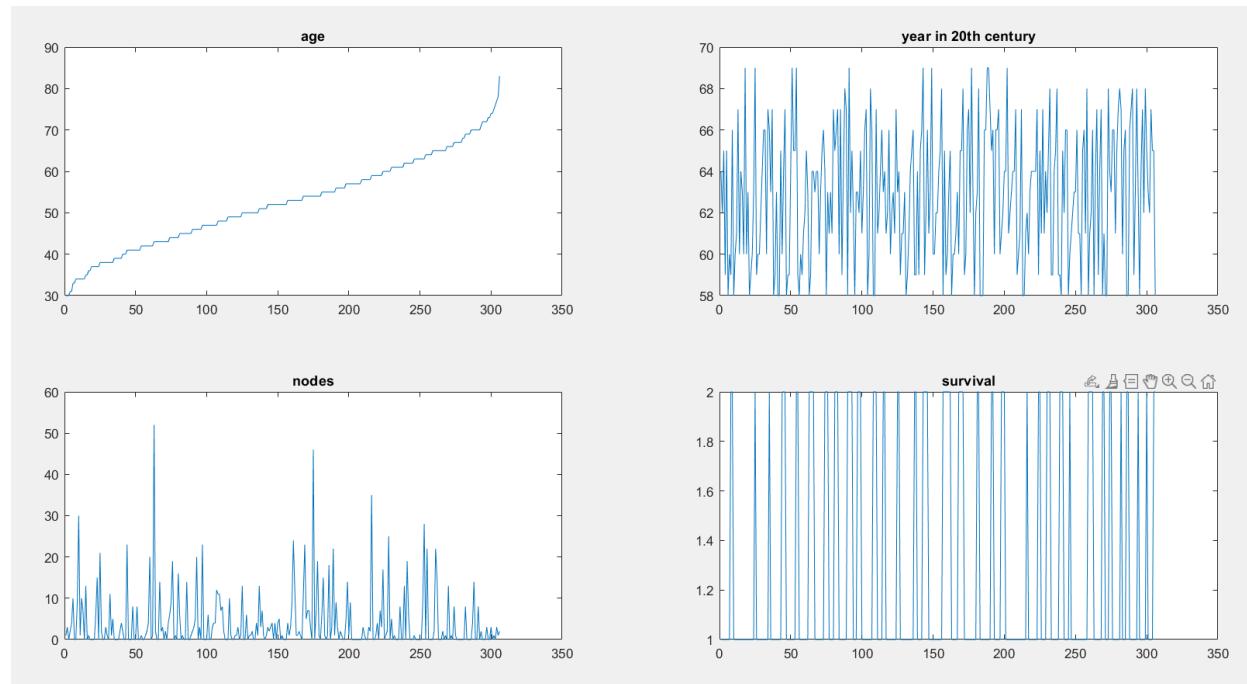
- survival (class attribute)

Η έξοδος αποτελείται από δύο κλάσεις {1,2}.

1 = the patient survived 5 years or longer

2 = the patient died within 5 year

Παρακάτω φαίνονται οι τιμές των εισόδων και της εξόδου.

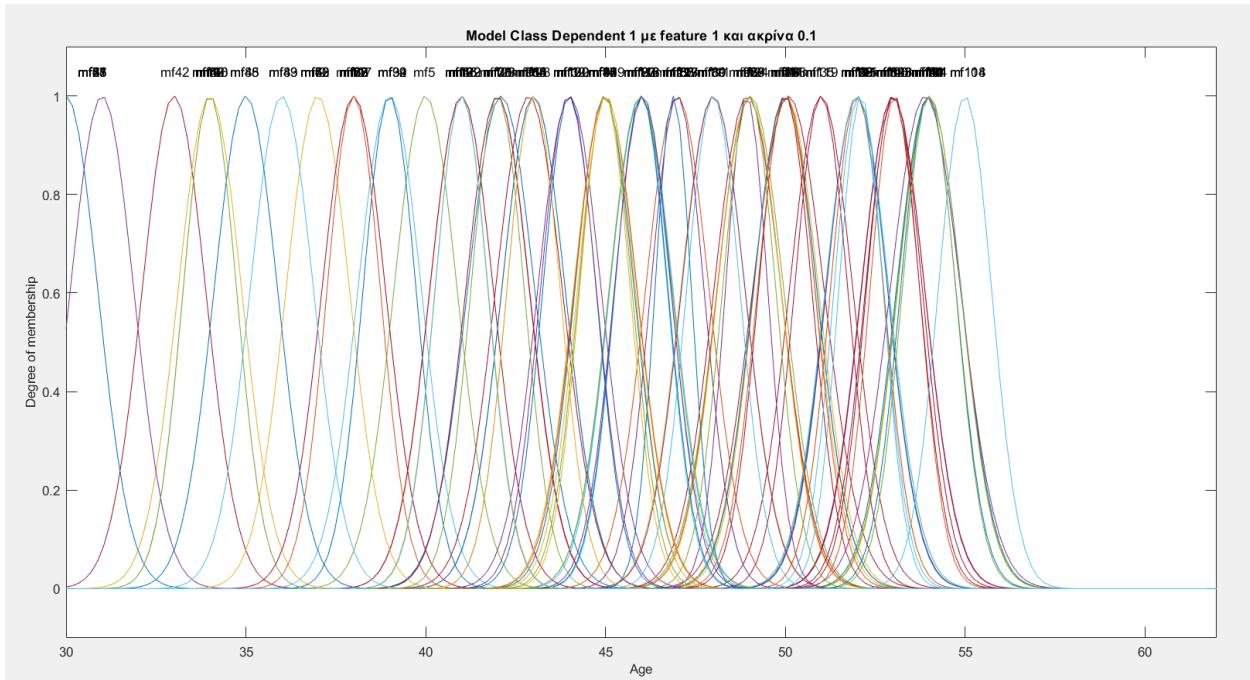


Σχήμα 95: Input & Output Values

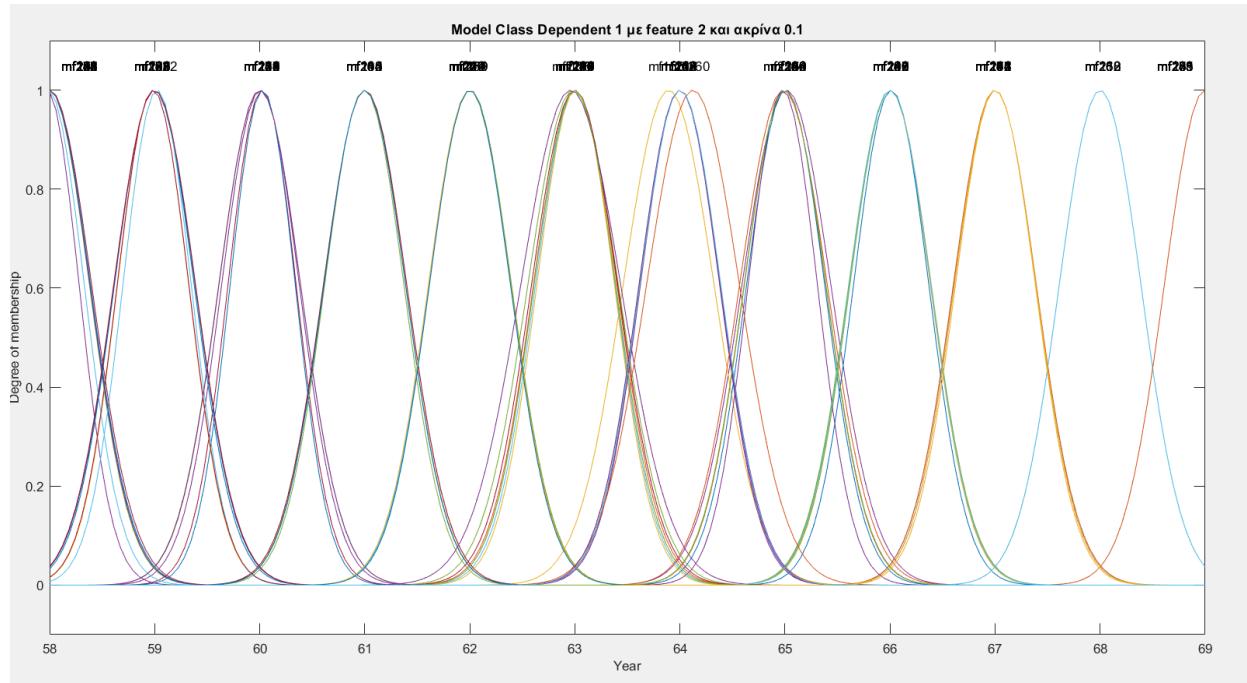
### 4.1.1 TSK model 1

TSK μοντέλο με ακτίνα 0.1 και class dependent.

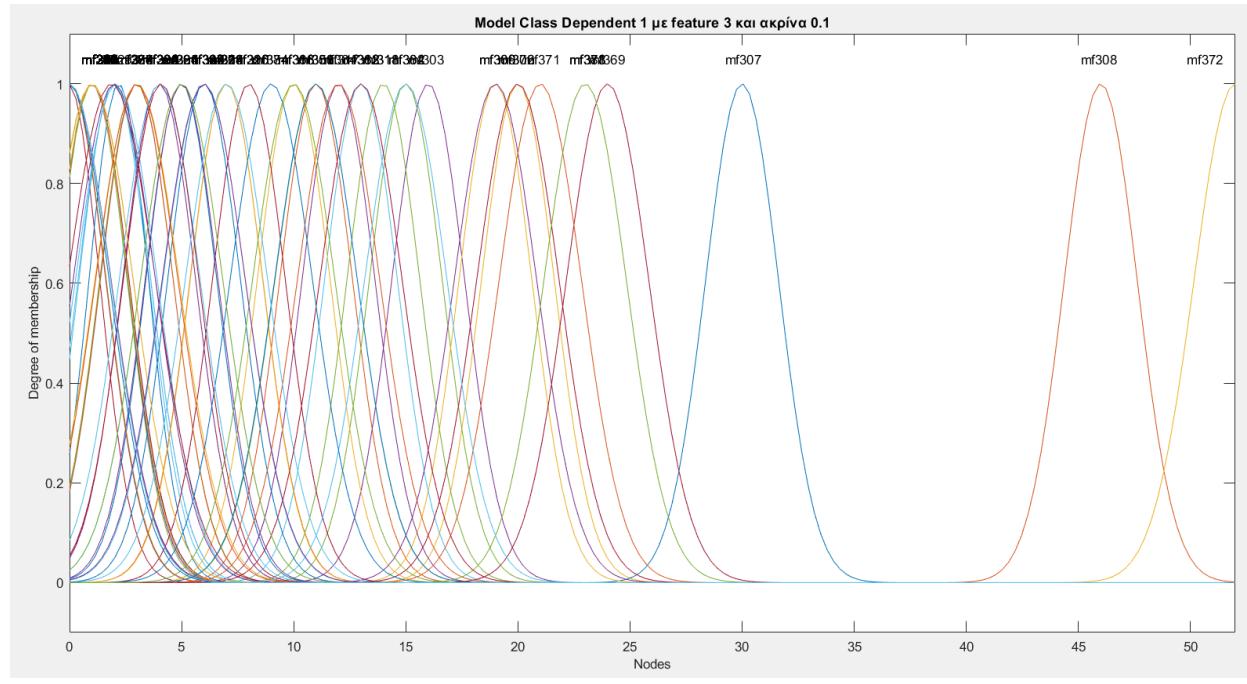
- Συναρτήσεις Συμμετοχής Μετά την Εκπαίδευση



Σχήμα 96: mf After Training – Age – class dependent - radius = 0.1

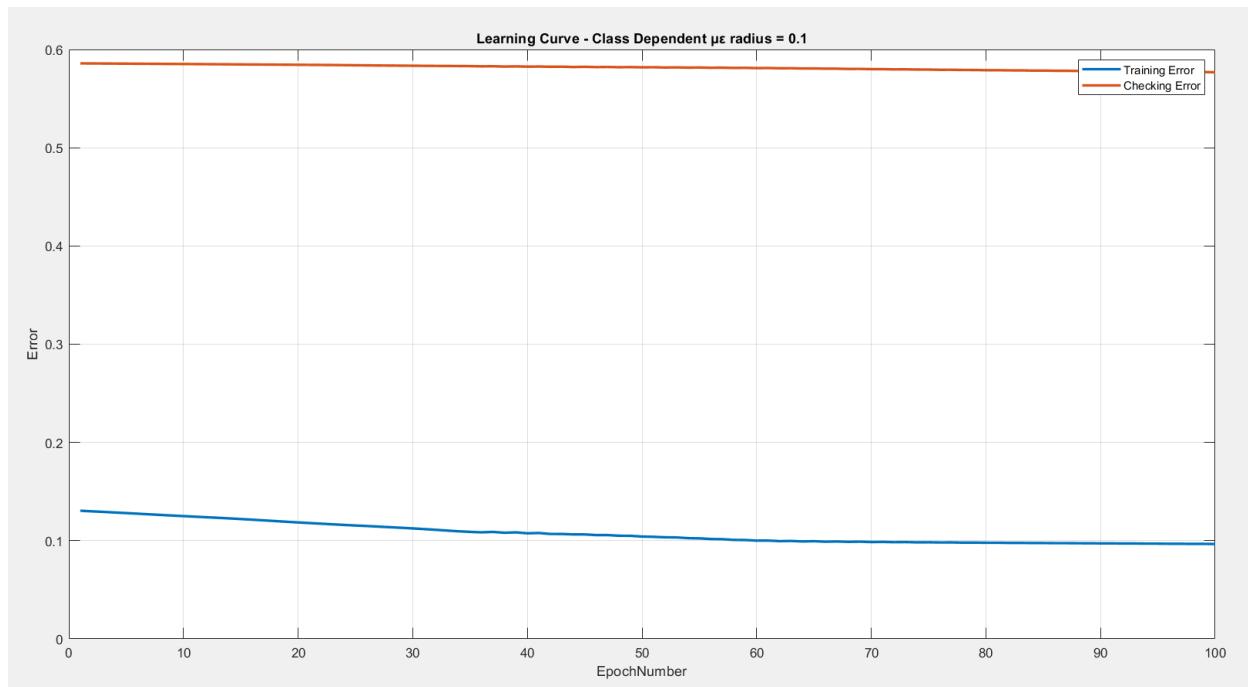


Σχήμα 97: mf After Training – Year – class dependent - radius = 0.1



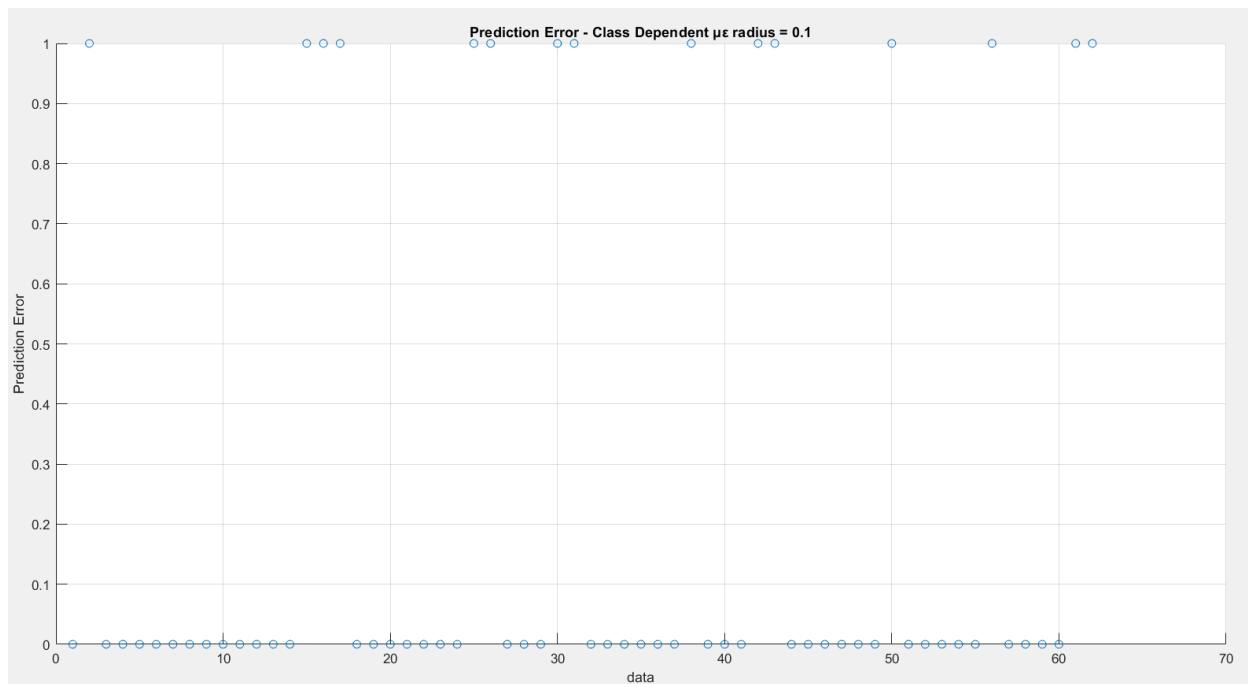
Σχήμα 98: mf After Training – Nodes – class dependent - radius = 0.1

- Learning Curve



Σχήμα 99: Learning Curve - Class Dependent - radius = 0.1

- Prediction Error



Σχήμα 100: Prediction Error - Class Dependent - radius = 0.1

{ 100 }

## Αξιολόγηση Μοντέλου 1

Μετρικές	Μοντέλο 1
OA	0.7581
PA 1	1.0000
PA 2	0.0625
UA 1	0.7541
UA 2	1.0000
K	0.0900
Rules	132

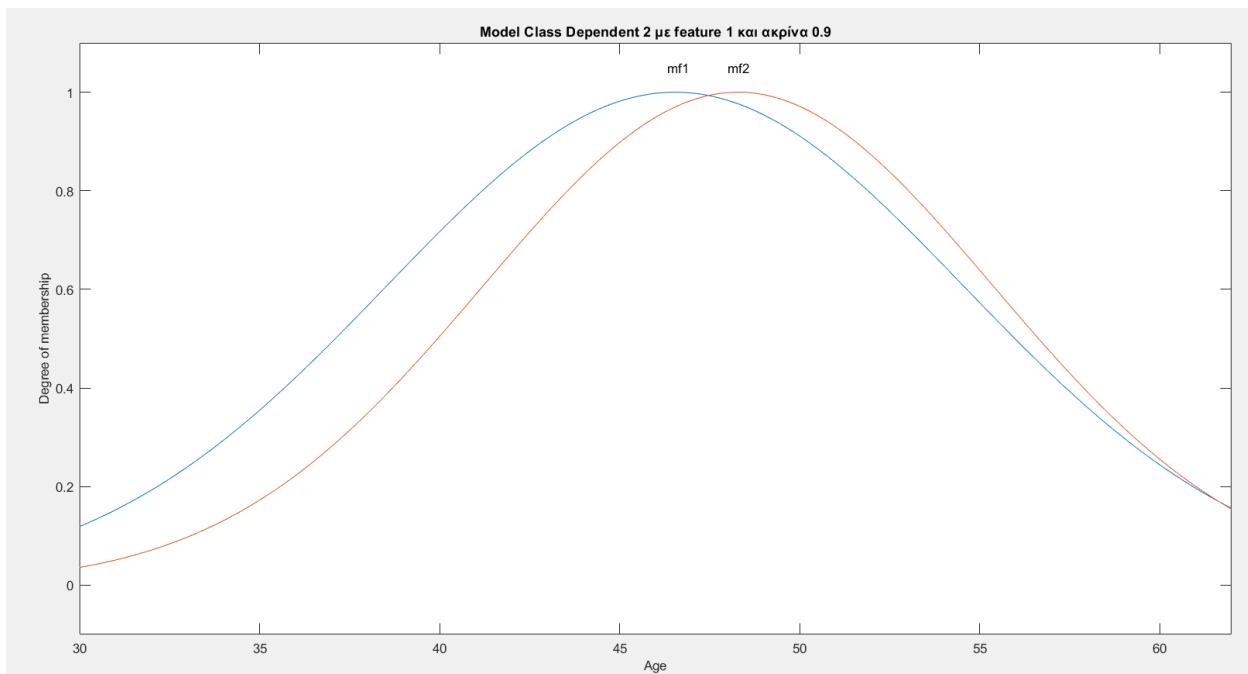
## Error Matrix Μοντέλου 1

46	0
15	1

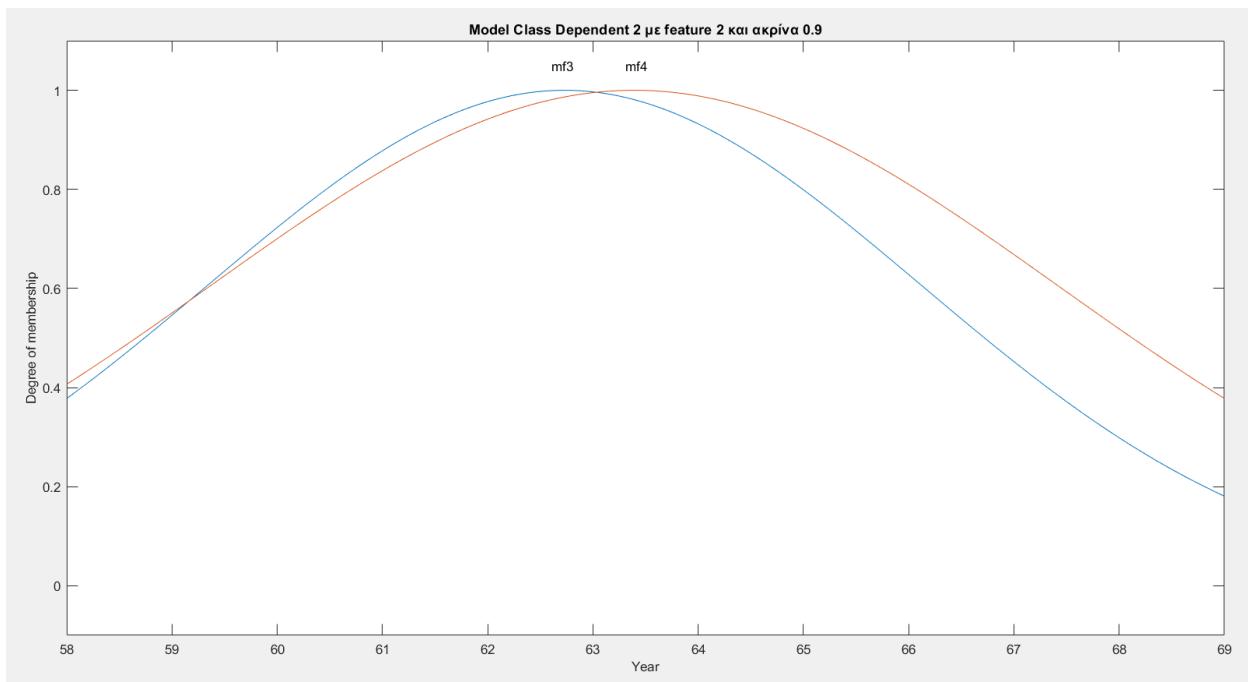
### 4.1.2 TSK model 2

TSK μοντέλο με ακτίνα 0.9 και class dependent.

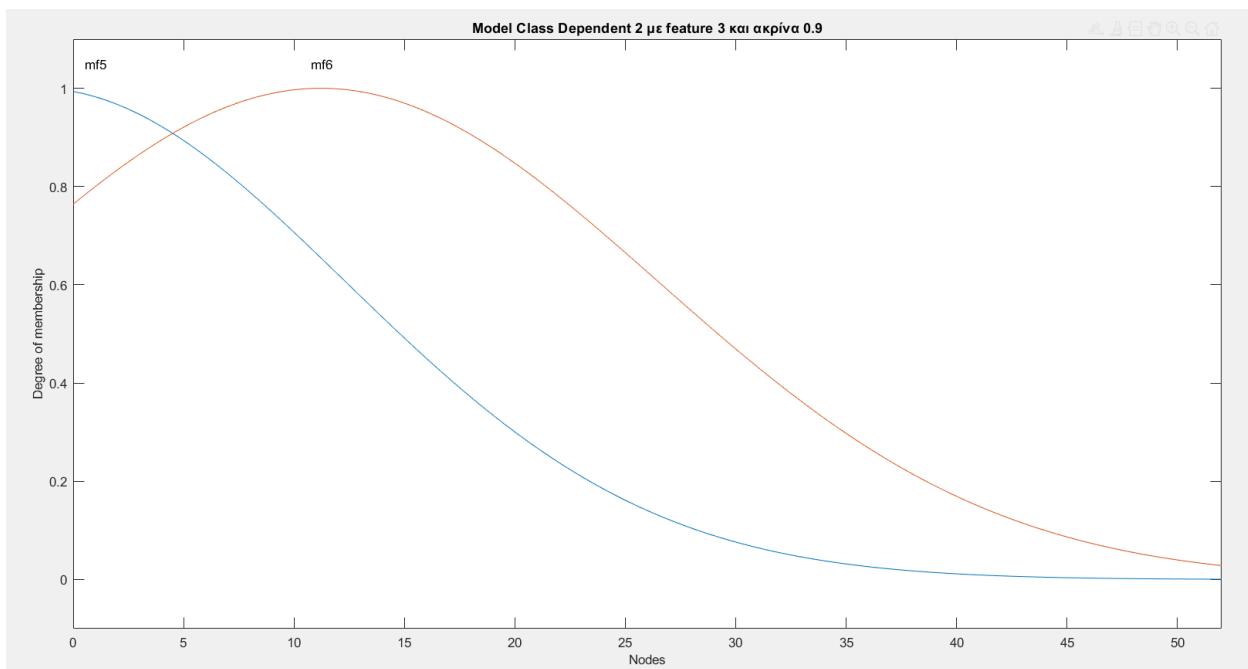
- Συναρτήσεις Συμμετοχής Μετά την Εκπαίδευση



Σχήμα 101: mf After Training – Age – class dependent - radius = 0.9

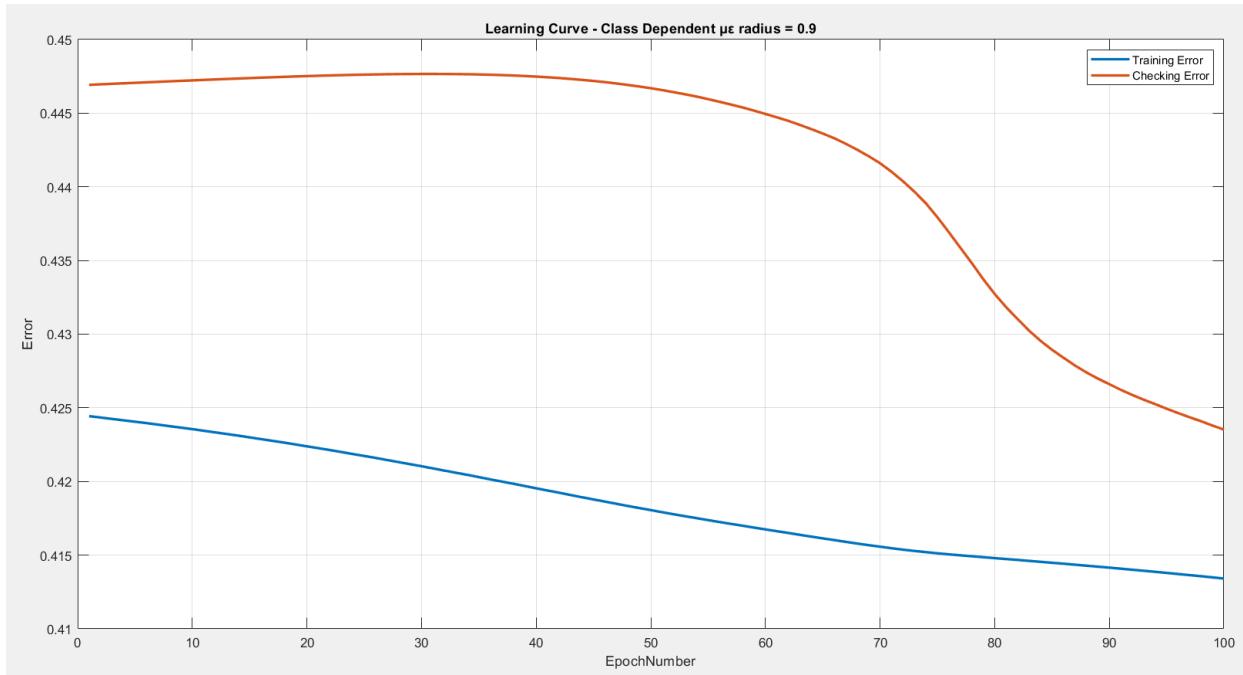


Σχήμα 102: mf After Training – Year – class dependent - radius = 0.9



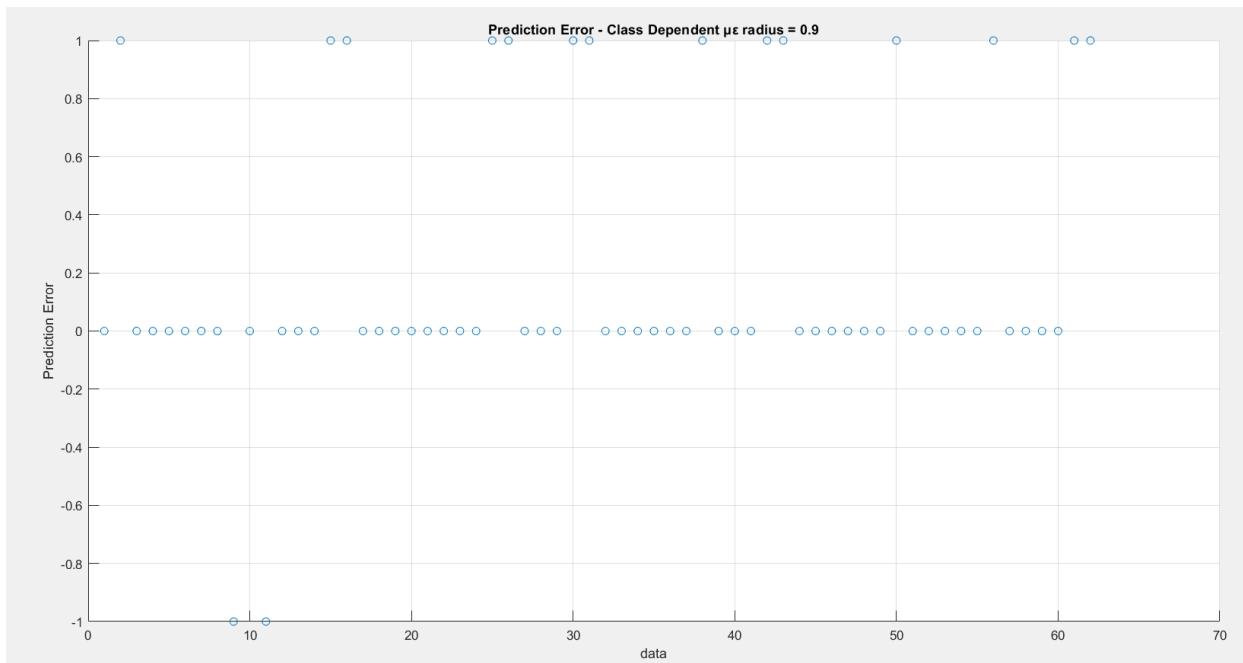
Σχήμα 103: mf After Training –Nodes – class dependent - radius = 0.9

- Learning Curve



Σχήμα 104: Learning Curve - Class Dependent - radius = 0.9

- Prediction Error



Σχήμα 105: Prediction Error - Class Dependent - radius = 0.9

## Αξιολόγηση Μοντέλου 2

Μετρικές	Μοντέλο 2
OA	0.7419
PA 1	0.9565
PA 2	0.1250
UA 1	0.7586
UA 2	0.5000
K	0.1079
Rules	2

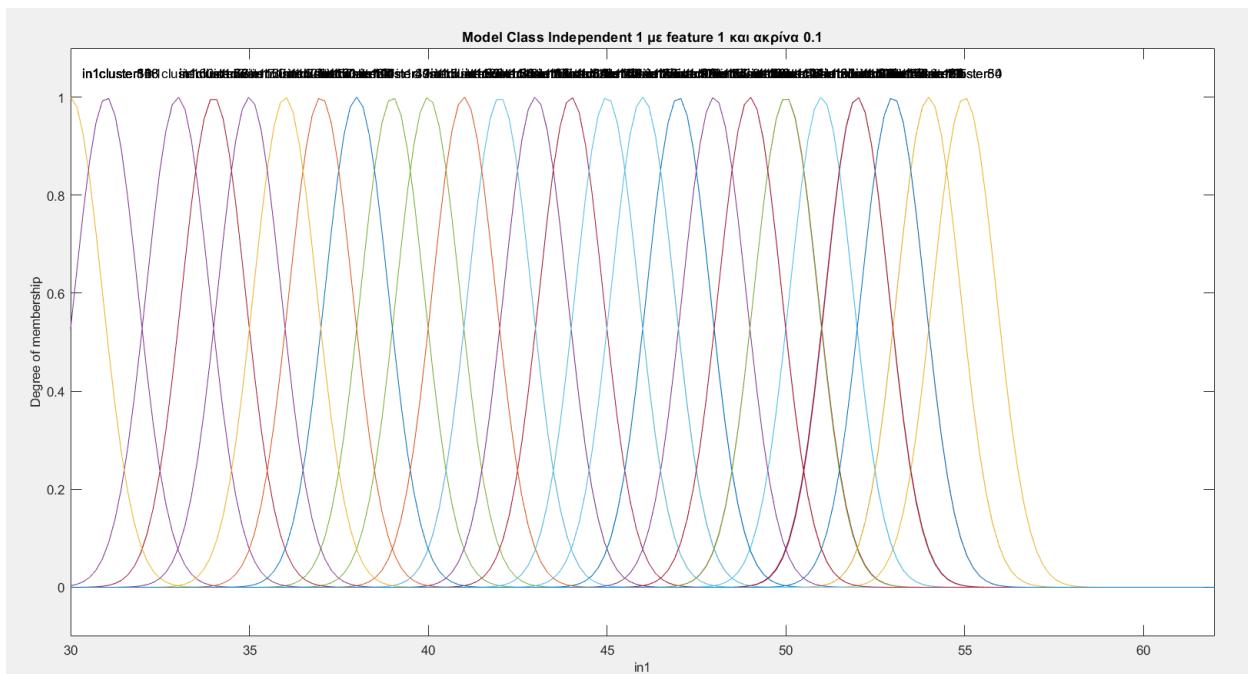
## Error Matrix Μοντέλου 2

44	2
14	2

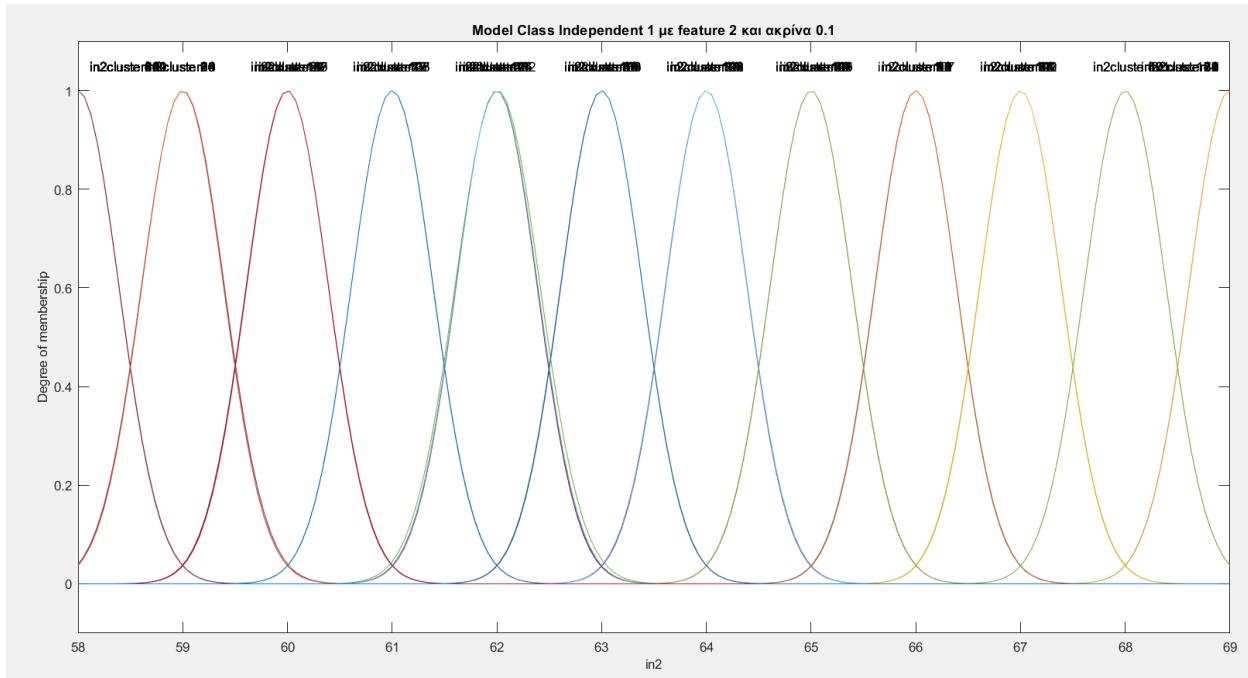
### 4.1.3 TSK model 3

TSK μοντέλο με ακτίνα 0.1 και class independent.

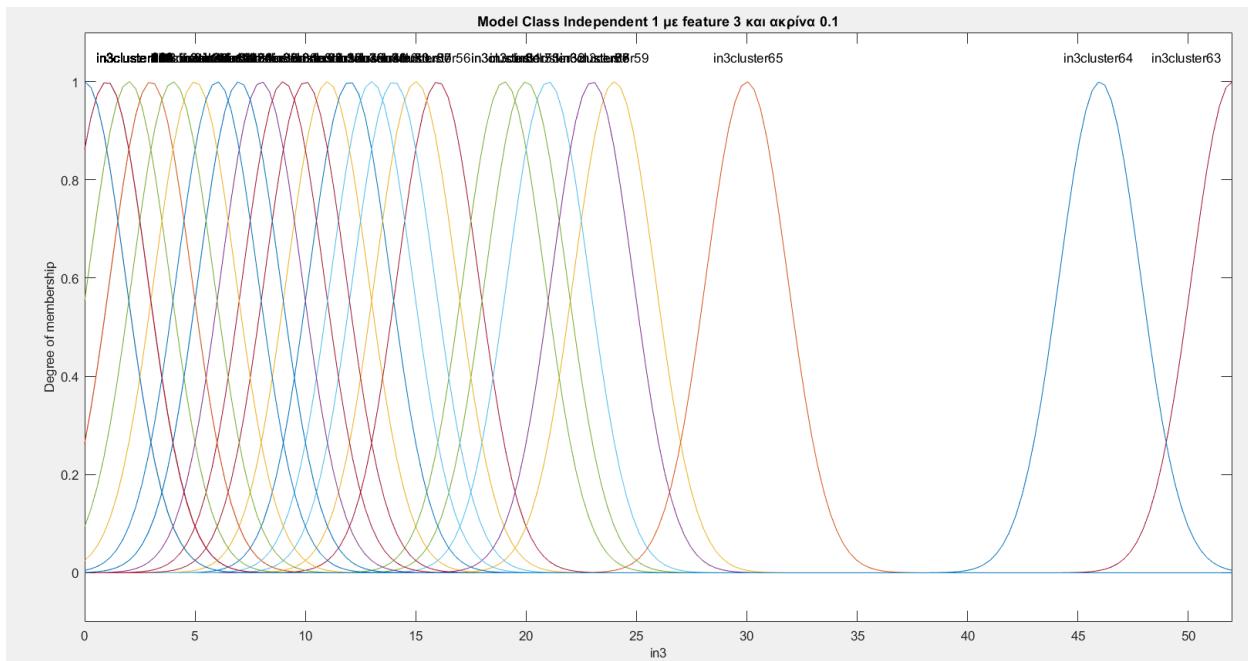
- Συναρτήσεις Συμμετοχής Μετά την Εκπαίδευση



Σχήμα 106: mf After Training – Age – class independent - radius = 0.1

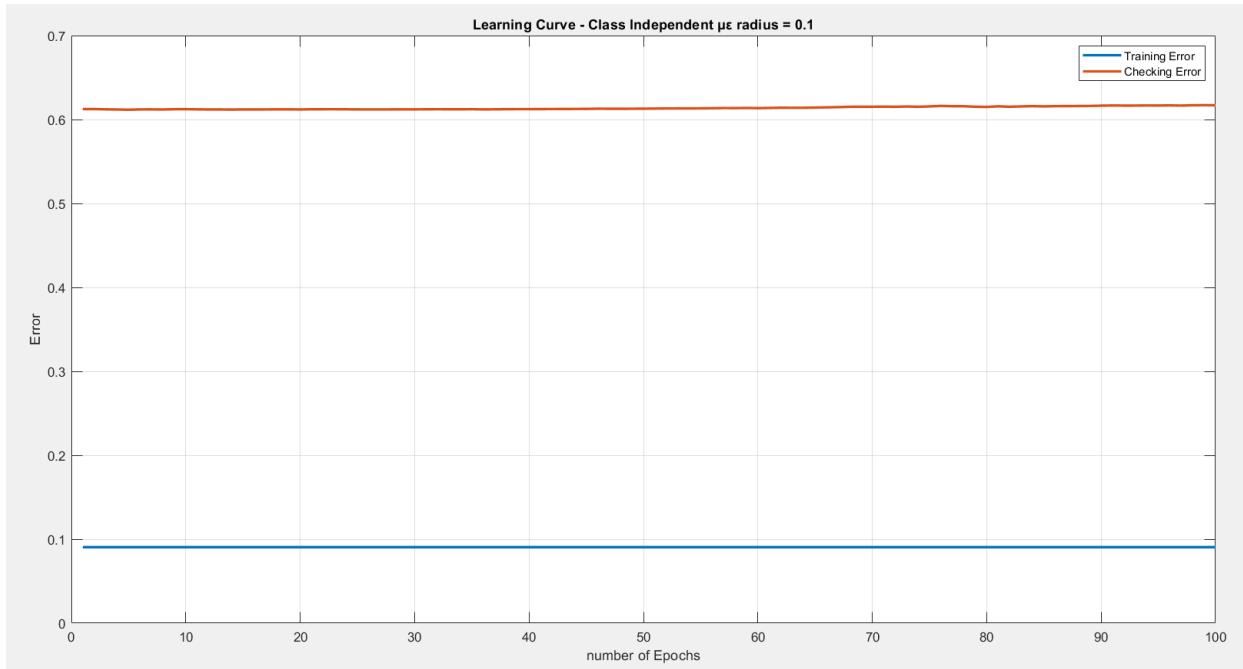


Σχήμα 107: mf After Training – Year – class independent - radius = 0.1



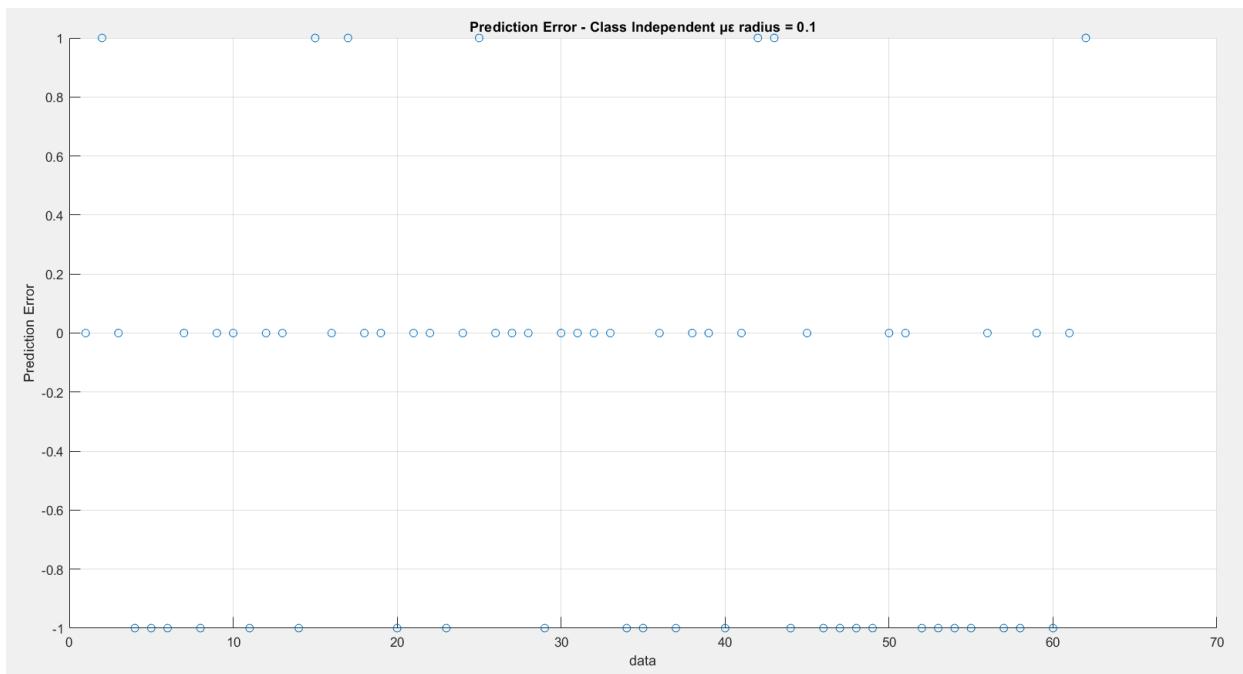
Σχήμα 108: mf After Training – Nodes – class independent - radius = 0.1

- Learning Curve



Σχήμα 109: Learning Curve - Class Independent - radius = 0.1

- Prediction Error



Σχήμα 110: Prediction Error - Class Independent - radius = 0.1

### Αξιολόγηση Μοντέλου 3

Μετρικές	Μοντέλο 3
OA	0.4839
PA 1	0.4565
PA 2	0.5625
UA 1	0.7500
UA 2	0.2647
K	0.0139
Rules	134

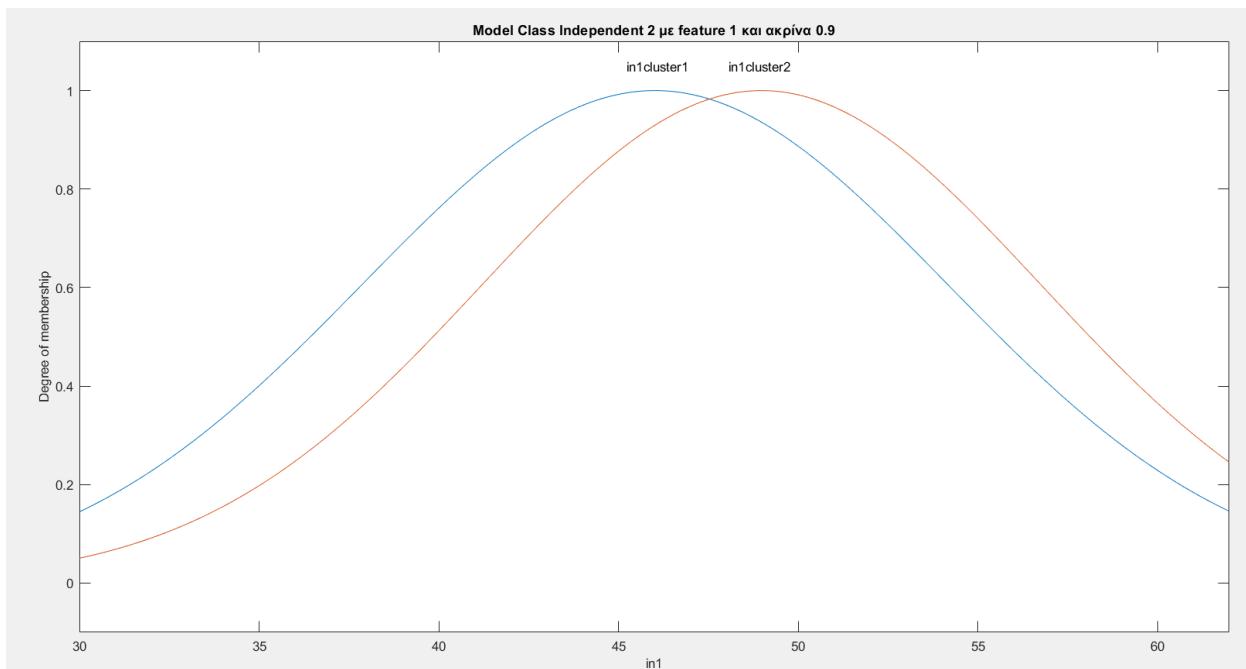
### Error Matrix Μοντέλου 3

21	25
7	9

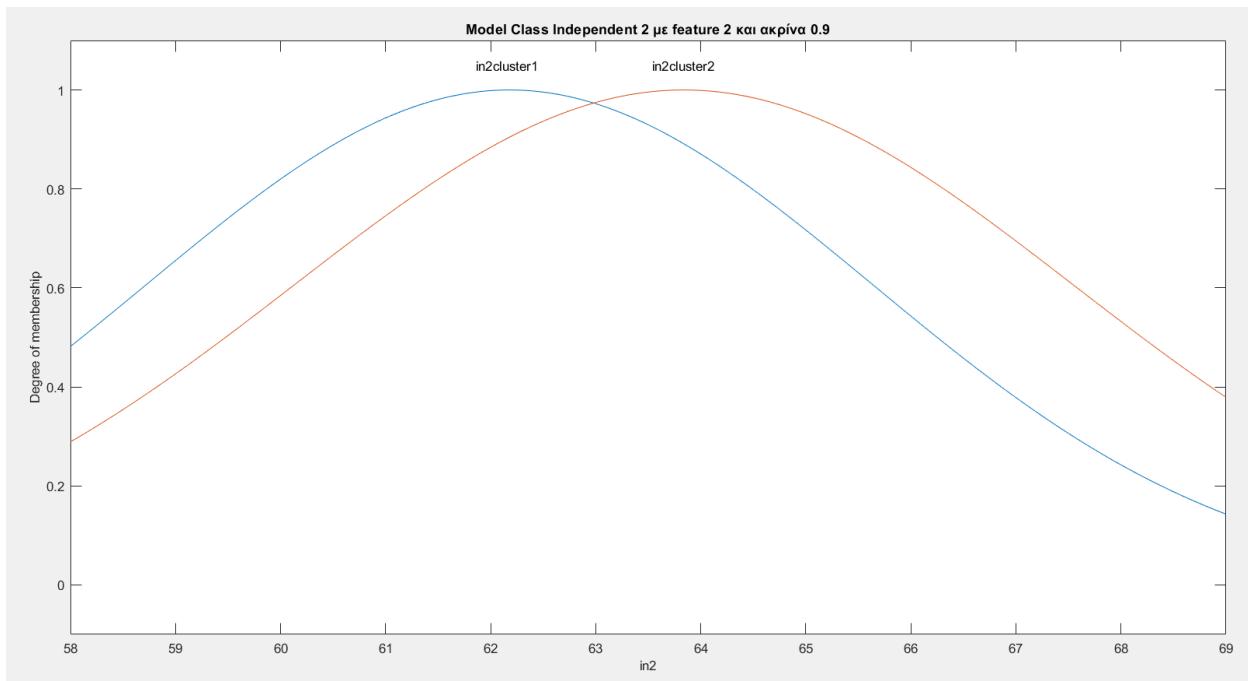
#### 4.1.4 TSK model 4

TSK μοντέλο με ακτίνα 0.9 και class independent.

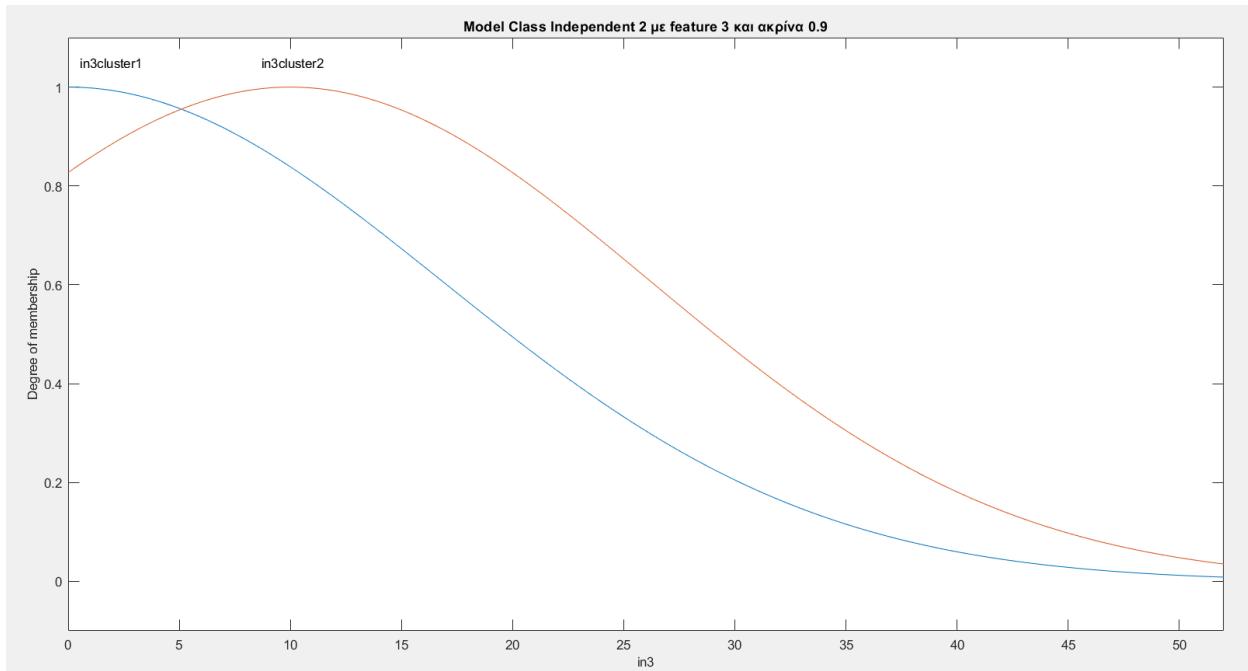
- Συναρτήσεις Συμμετοχής Μετά την Εκπαίδευση



Σχήμα 111: mf After Training – Age – class independent - radius = 0.9

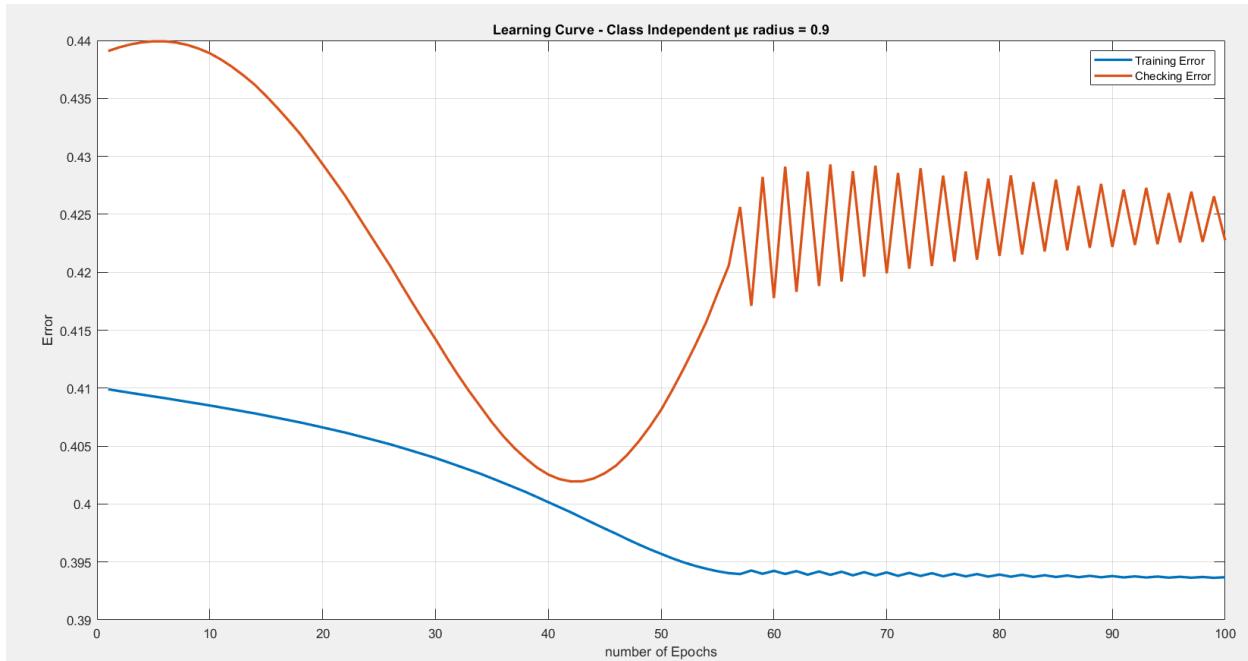


Σχήμα 112: mf After Training – Year – class independent - radius = 0.9



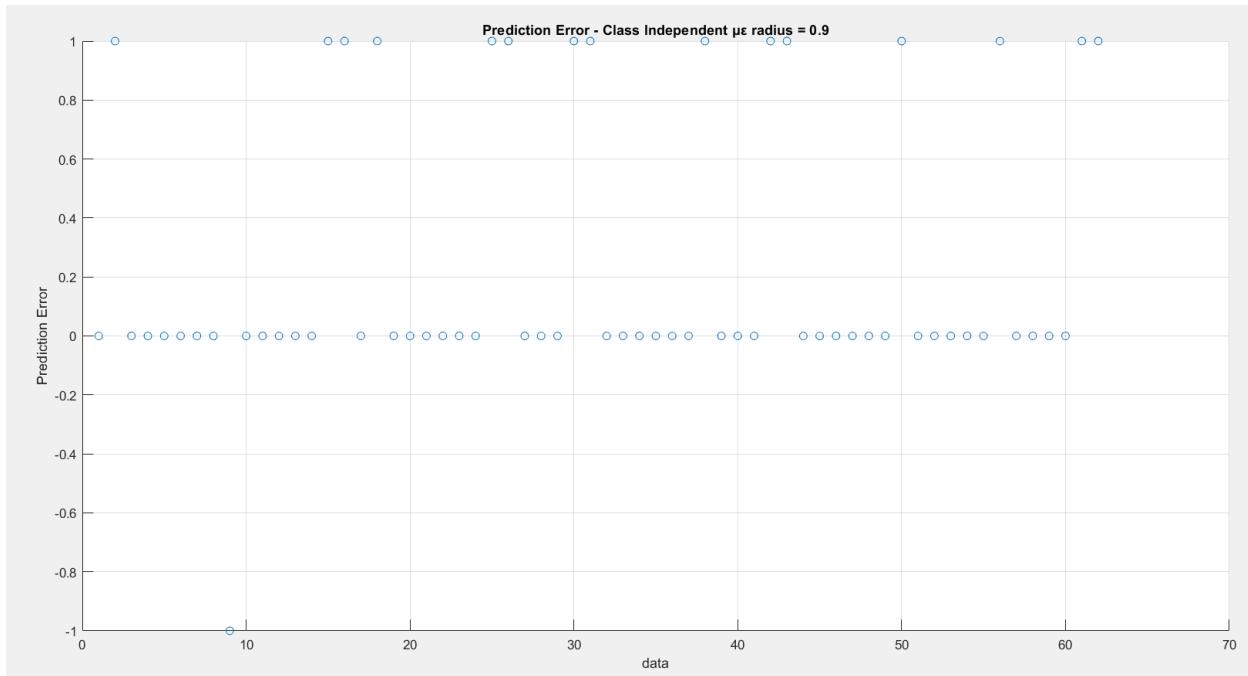
Σχήμα 113: mf After Training – Nodes – class independent - radius = 0.9

- Learning Curve



Σχήμα 114: Learning Curve - Class Independent - radius = 0.9

- Prediction Error



Σχήμα 115: Prediction Error - Class Independent - radius = 0.9

#### Αξιολόγηση Μοντέλου 4

Μετρικές	Μοντέλο 4
OA	0.7419
PA 1	0.9783
PA 2	0.0625
UA 1	0.7500
UA 2	0.5000
K	0.0570
Rules	2

#### Error Matrix Μοντέλου 4

45	1
15	1

#### 4.1.5 Αξιολόγηση Μοντέλων και Πίνακες Σφαλμάτων

- Πίνακας Μετρικών Αξιολόγησης

Μοντέλα\Μετρικές	OA	PA 1	PA 2	UA 1	UA 2	K	Rules
Μοντέλο 1	0.7581	1.0000	0.0625	0.7541	1.0000	0.0900	132
Μοντέλο 2	0.7419	0.9565	0.1250	0.7586	0.5000	0.1079	2
Μοντέλο 3	0.4839	0.4565	0.5625	0.7500	0.2647	0.0139	134
Μοντέλο 4	0.7419	0.9783	0.0625	0.7500	0.5000	0.0570	2

- Οι Μετρικές Αξιολόγησης υπολογίζονται από τους παρακάτω τύπους:

$$OA = (TP + TN) / (TP + FP + TN + FN)$$

$$PA1 = TP / (TP + FN)$$

$$PA2 = TN / (TN + FP)$$

$$UA1 = TP / (TP + FP)$$

$$UA2 = TN / (TN + FN)$$

$$K = (N * (TP + TN) - ((TP + FP) * (TP + FN) + (FN + TN) * (FP + TN))) / (N^2 - ((TP + FP) * (TP + FN) + (FN + TN) * (FP + TN)))$$

- Πίνακες Σφαλμάτων

<b>Μοντέλο 1</b>	Predicted Class 1	Predicted Class 2
True Class 1	46	0
True Class 2	15	1

<b>Μοντέλο 2</b>	Predicted Class 1	Predicted Class 2
True Class 1	44	2
True Class 2	14	2

<b>Μοντέλο 3</b>	Predicted Class 1	Predicted Class 2
True Class 1	21	25
True Class 2	7	9

<b>Μοντέλο 4</b>	Predicted Class 1	Predicted Class 2
True Class 1	45	1
True Class 2	15	1

- Οι πίνακες Σφαλμάτων έχουν την εξής δομή:

<b>Μοντέλο</b>	Predicted Class 1	Predicted Class 2
True Class 1	TP	FN
True Class 2	FP	TN

- Ο πίνακας Σφαλμάτων υπολογίζεται με την εντολή “confusionmat(TrueData, PredictedData)”, που επιστρέφει τον confusion matrix.

#### 4.1.6 Συμπεράσματα και Βέλτιστο Μοντέλο

Από τους παραπάνω πίνακες παρατηρούμε ότι τα καλύτερα μοντέλα είναι το 1 και το 2, δηλαδή τα Class Dependent, γιατί έχουν το μικρότερο Overall Accuracy, με τιμές 0.7581 και 0.7419 αντίστοιχα και πετυχαίνουν καλύτερες μετρικές.

Επίσης, έχουν πολύ καλό Producer's Accuracy με τιμές 1 και 0.9565 αντίστοιχα, δηλαδή ομαδοποιούν σωστά τα δείγματα της κλάσης 1.

Και τα δύο μοντέλα έχουν ικανοποιητικές τιμές User's Accuracy, με το μοντέλο 1 να έχει καλύτερο UA 2.

Όμως, το μοντέλο 2 έχει καλύτερη Εκτίμηση Πραγματικής Στατιστικής Παραμέτρου K με τιμή 0.1079, με μικρή διαφορά από το μοντέλο 1 που έχει K με τιμή 0.0900.

Βλέπουμε ότι τα μοντέλα με Class Dependent Subtractive Clustering, που έχουν ίδια ακτίνα για το cluster, πετυχαίνουν καλύτερα αποτελέσματα ομαδοποίησης σε σχέση με τα Class Independent μοντέλα. Δηλαλή, το μοντέλο 1 είναι καλύτερο από το μοντέλο 3 και το μοντέλο 2 είναι καλύτερο από το μοντέλο 4.

Παρατηρούμε ότι τα μοντέλα 1 και 3 με μικρή ακτίνα cluster 0.1 έχουν πολύ μεγάλο αριθμό κανόνων. Αντίθετα, τα μοντέλα 2 και 4 με μεγάλη ακτίνα 0.9 έχουν μόνο δύο κανόνες για κάθε Συνάρτηση Συμμετοχής.

Όμως, είναι φανερό ότι ο αριθμός των κανόνων (άρα και το μέγεθος της ακτίνας των clusters) δεν επηρεάζει την απόδοση του μοντέλου. Αυτό που επηρεάζει σημαντικά την απόδοση είναι η μέθοδος με την οποία έγινε ο διαμερισμός του χώρου των δεδομένων.

Γνωρίζουμε όμως ότι όσο πιο μικρή είναι η ακρίνα των clusters, τόσο περισσότερους κανόνες παράγει το μοντέλο. Αυτό συνεπάγεται ότι το μοντέλο γίνεται όλο και πιο περίπλοκο, με αποτέλεσμα να κινδυνεύει να πάθει overtraining, επομένως είναι πιο δύσκολο να γενικεύσει όσα έμαθε και σε διαφορετικά datasets. Ένας τρόπος αντιμετώπισης αυτού του προβλήματος είναι να αφαιρέσουμε τους κανόνες που κάνουν overlap, ώστε το μοντέλο να έχει λίγους και σημαντικούς κανόνες.

Με βάση όλα τα παραπάνω επιλέγω ως βέλτιστο μοντέλο, το **Μοντέλο 2** με Class Dependent και ακτίνα 0.9 (2 κανόνες).

## 4.2 Part 2 - Εφαρμογή σε dataset με υψηλή διαστασιμότητα

Σε αυτή την εργασία ερευνούμε τις βέλτιστες παραμέτρους (αριθμό σημαντικότερων features, αριθμό κανόνων, ακτίνα των clusters) σε dataset υψηλής διαστασιμότητας. Το dataset Που μελετάμε είναι το “Epileptic Seizure Recognition” από το UCI repository. Περιλαμβάνει 11500 δείγματα, τα οποία περιγράφονται από 178 features και 1 output.

Η έξοδος αποτελείται από πέντε κλάσεις {1,2,3,4,5}.

1 = Recording of seizure activity

2 = They recorder the EEG from the area where the tumor was located

3 = Yes, they identify where the region of the tumor was in the brain and recording the EEG activity from the healthy brain area

4 = Eyes closed, means when they were recording the EEG signal the patient had their eyes closed

5 = Eyes open, means when they were recording the EEG signal of the brain the patient had their eyes open

Είναι απαραίτητο να επιλέξουμε ένα μικρότερο σύνολο features με τα οποία θα εργαστούμε, διότι τα 178 είναι απαγορευτικά μεγάλο νούμερο, λογω της έκρηξης των κανόνων If-Then.

Αυτό επιτυγχάνεται με την εντολή Relieff.

Το μοντέλο υλοποιείται στο script “classification\_Epileptic.m”.

Όπως και στην εργασία #3, δοκιμάζουμε διαφορετικές τιμές για τον αριθμό των σημαντικότερων features [5, 7, 9, 11] που θα κρατήσουμε για να τροφοδοτήσουμε το μοντέλο, καθώς και διάφορες τιμές για την ακτίνα radius [0.2, 0.4, 0.6, 0.8, 1] των clusters.

Όσο αυξάνουμε τον αριθμό των features τόσο πιο αργό γίνεται το μοντέλο και σε πολύ θψηλές τιμές μη λειτουργικό.

Χρησιμοποιούμε την τεχνική Grid Search και διαχωρίζουμε τα δεδομένα με την μέθοδο CV Partition k-fold με k=5.

Ο διαχωρισμός του χώρου του input dataset γίνεται με subtractive clustering στα δεδομένα του συνόλου εκπαίδευσης, που ανήκουν στην εκάστοτε κλάση ξεχωριστά (**Class Dependent**).

Τα μοντέλα δημιουργούνται με new fis κατασκευασμένο εξ αρχής, με Membership Functions που ορίζουμε εμείς στο script.

Κάθε μοντέλο εκπαιδεύεται με την μέθοδο anfis και χρησιμοποιούμε 100 EpochNumber.

Για την αξιολόγηση των μοντέλων χρησιμοπούμε την μετρική Overall Accuracy.

#### 4.2.1 Αποτελέσματα του Grid Search

- Πίνακας Overall Accuracy **OA**

<b>Features\radius</b>	<b>0.2</b>	<b>0.4</b>	<b>0.6</b>	<b>0.8</b>	<b>1</b>
<b>5</b>	0.3629	0.3392	0.3235	0.3142	0.3250
<b>7</b>	0.3810	0.3604	0.3535	0.3435	0.3563
<b>9</b>	0.3939	0.3684	0.3857	0.3786	0.3544
<b>11</b>	0.4017	0.3730	0.3800	0.3901	0.3673

- Πίνακας Αριθμού Κανόνων **Rules**

<b>Features\radius</b>	<b>0.2</b>	<b>0.4</b>	<b>0.6</b>	<b>0.8</b>	<b>1</b>
<b>5</b>	31	10	5.6	5	5
<b>7</b>	42	10.8	6	5.2	5
<b>9</b>	53.2	12.2	7	5.2	5
<b>11</b>	122.2	13.8	7.2	6	5

Παρατηρούμε ότι ο αριθμός των κανόνων αυξάνεται όταν ο αριθμός των features αυξάνεται καιόταν η ακτίνα των clusters μειώνεται.

Σύμφωνα με τα παραπάνω αποτελέσματα πετυχαίνουμε το βέλτιστο Overall Accuracy

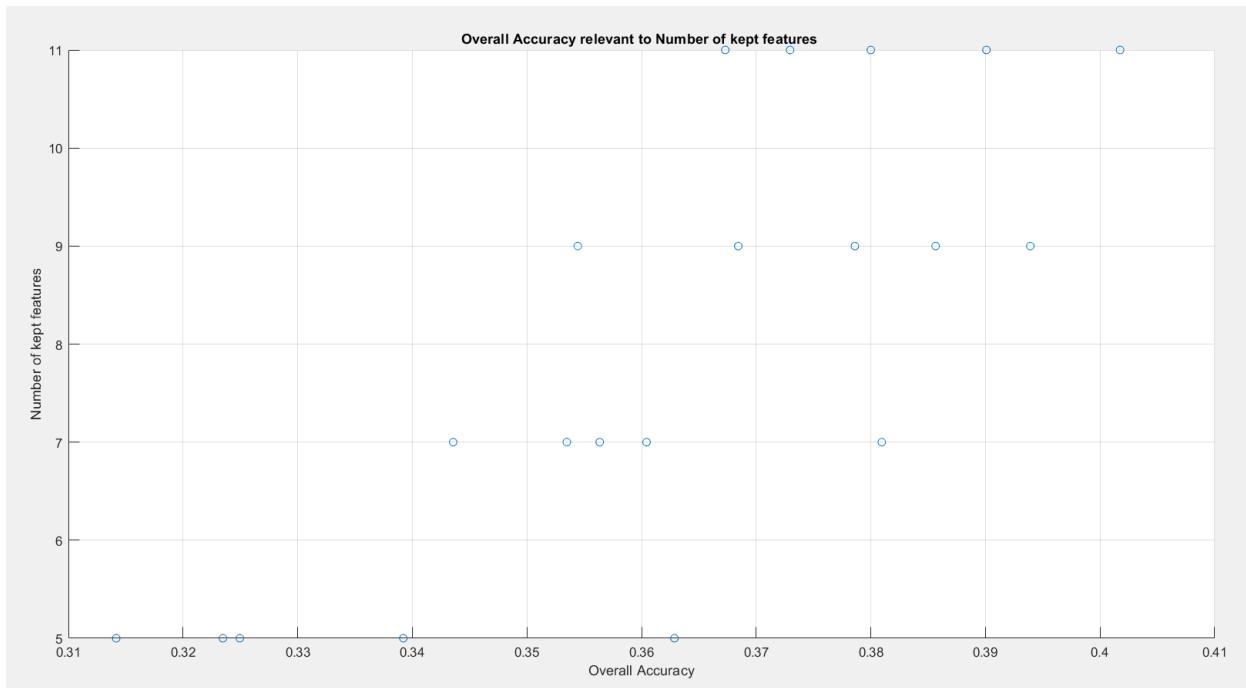
**OA = 0.4017** με παραμέτρους:

- features = 11
- rules = 122.2
- radius = 0.2

#### 4.2.2 Διαγράμματα του Grid Search

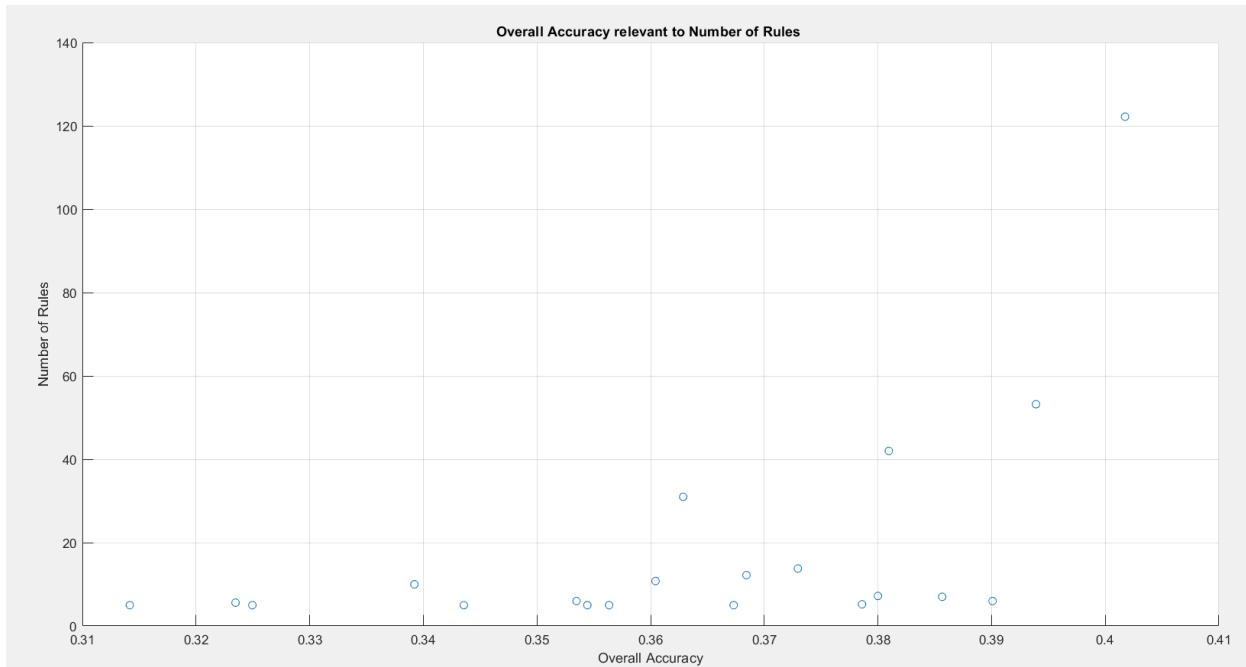
Παρακάτω φαίνονται τα διαγράμματα scatters για το Overall Accuracy συναρτήσει του αριθμού των σημαντικότερων features, του αριθμού των κανόνων και της ακτίνας των clusters.

- Overall Accuracy συναρτήσει του αριθμού των features



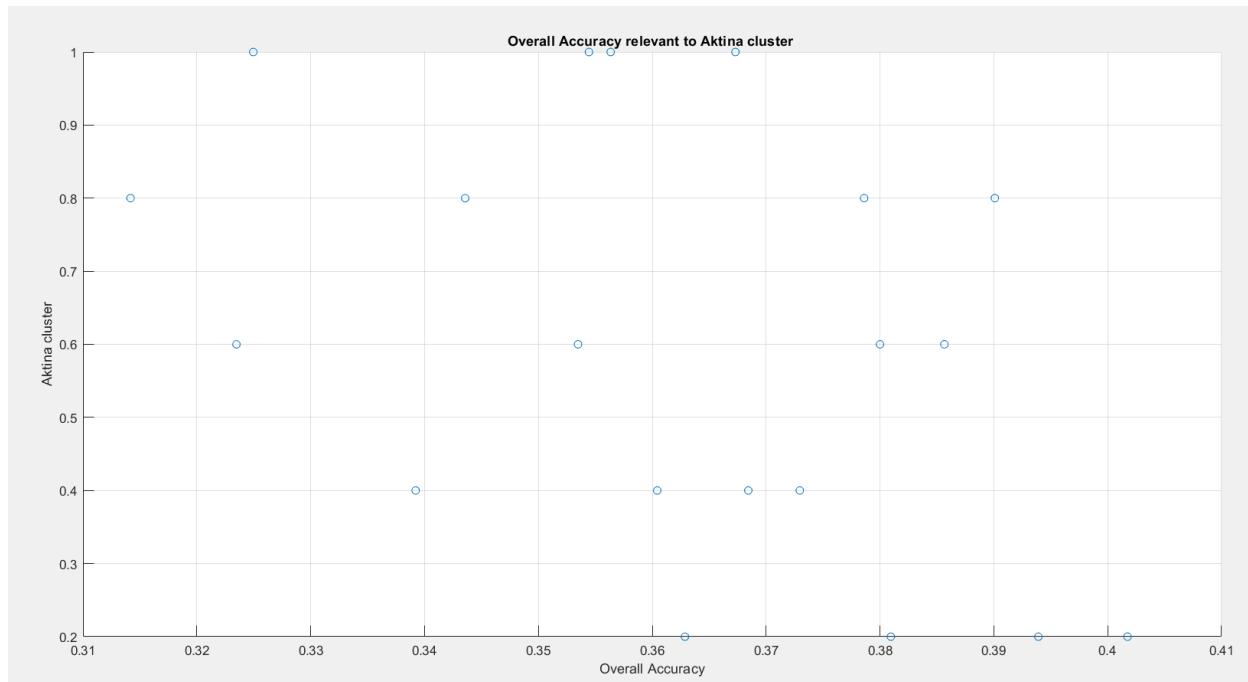
Σχήμα 116: OA - features

- Overall Accuracy συναρτήσει του αριθμού των κανόνων



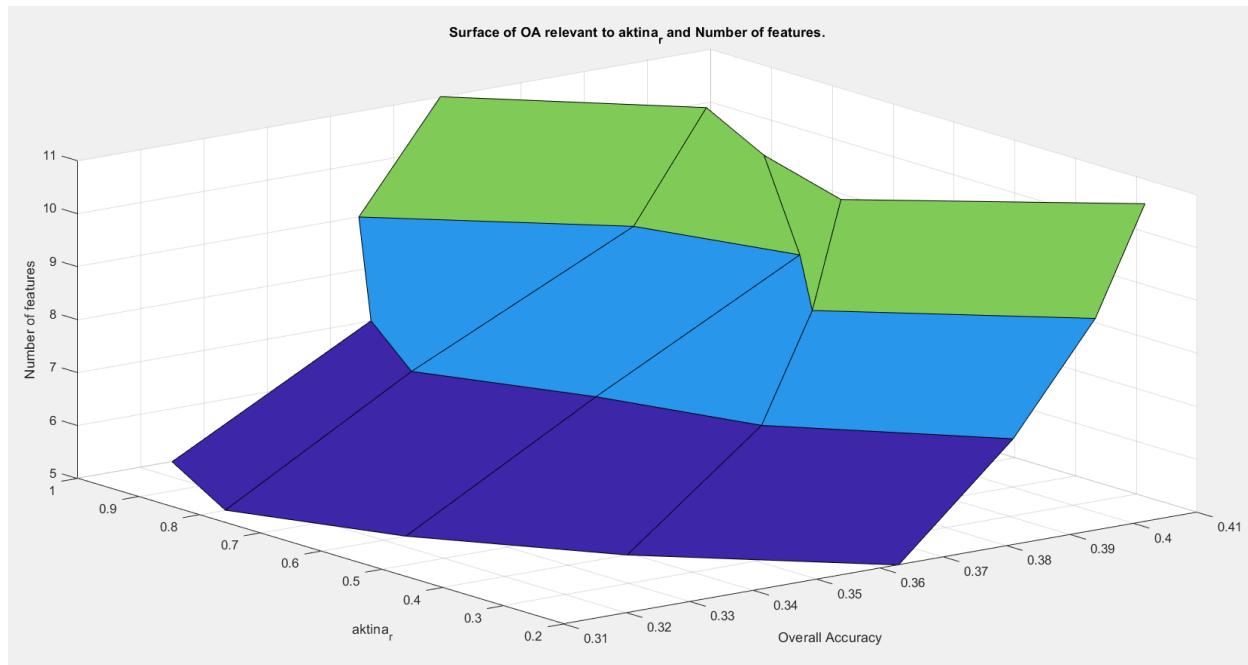
Σχήμα 117: OA - rules

- Overall Accuracy συναρτήσει της ακτίνας των clusters



Σχήμα 118: OA - radius

- Overall Accuracy συναρτήσει των features & radius



Σχήμα 119: OA – features & radius

Από τα παραπάνω διαγράμματα συμπεραίνουμε ότι το Overall Accuracy αυξάνεται όταν ο αριθμός των σημαντικότερων features αυξάνεται και όταν η ακτίνα των clusters μειώνεται.

Όμως, αυτό δεν είναι κανόνας, γιατί μεγάλος αριθμός feature και μικρή ακτίνα μπορεί να οδηγήσει σε overtraining, δηλαδή το μοντέλο να μάθει τέλεια το συγκεκριμένο dataset, αλλά να μην μπορεί να γενικεύσει όσα έμαθε.

Παρατηρούμε ότι παρόλο που έχουμε το καλύτερο OA για μεγάλο αριθμό features = 11, οι τιμές είναι αρκετά κοντά και για μικρό αριθμό features = 5, δεν υπάρχουν δηλαδή ακραίες τιμές.

#### 4.3.3 Βέλτιστο Μοντέλο

Το βέλτιστο μοντέλο επιλέχθηκε από το script “classification\_Epileptic.m” και υλοποιείται στο script “classification\_Optimal\_TSK\_model.m”.

Αποτελείται από 11 features και 0.2 ακτίνα.

- Πίνακας Μέσου Όρου Σφαλμάτων Error Matrix Βέλτιστου Μοντέλου (Confusion Matrix)

333.8	72.4	34.8	17.2	0.8
23.2	47.2	256.2	129.2	4.2
3	29	283.8	138.8	5.4
2.8	43.4	173	222	18.8
0.2	7.6	170	255.2	27

- Μέσος Όρος Μετρικών Βέλτιστου Μοντέλου

Mean Μετρικών	Βέλτιστο Μοντέλο
OA	0.3977
PA1	0.7278
PA2	0.1026
PA3	0.6170
PA4	0.4826
PA5	0.0587
UA1	0.9199
UA2	0.2351
UA3	0.3092
UA4	0.2911
UA5	0.4711
K	0.2472
Rules	92

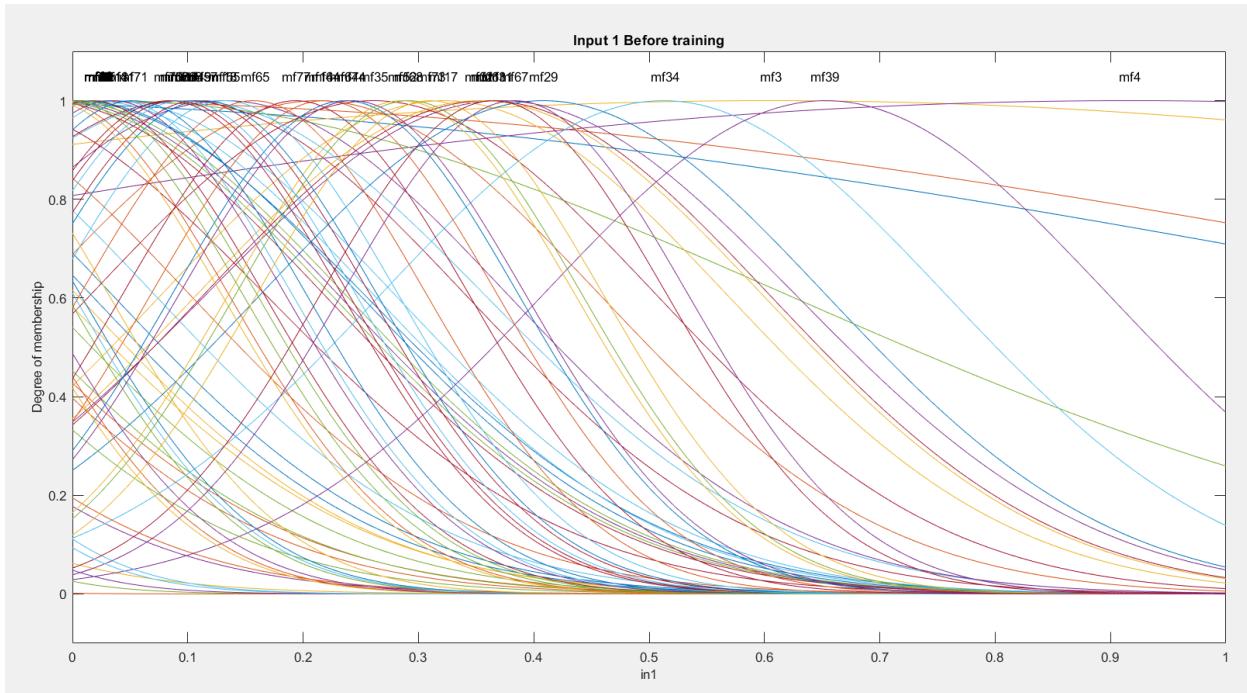
Παρατηρούμε ότι πετυχαίνουμε πολύ ικανοποιητικά αποτελέσματα.

- Ενδεικτικά κάποιες προβλέψεις του μοντέλου και οι πραγματικές του τιμές

Real Vale	Prediction
4	3
1	1
5	4
2	4
3	3
2	1
1	1
2	2
1	4
4	4
5	3
1	3
3	3

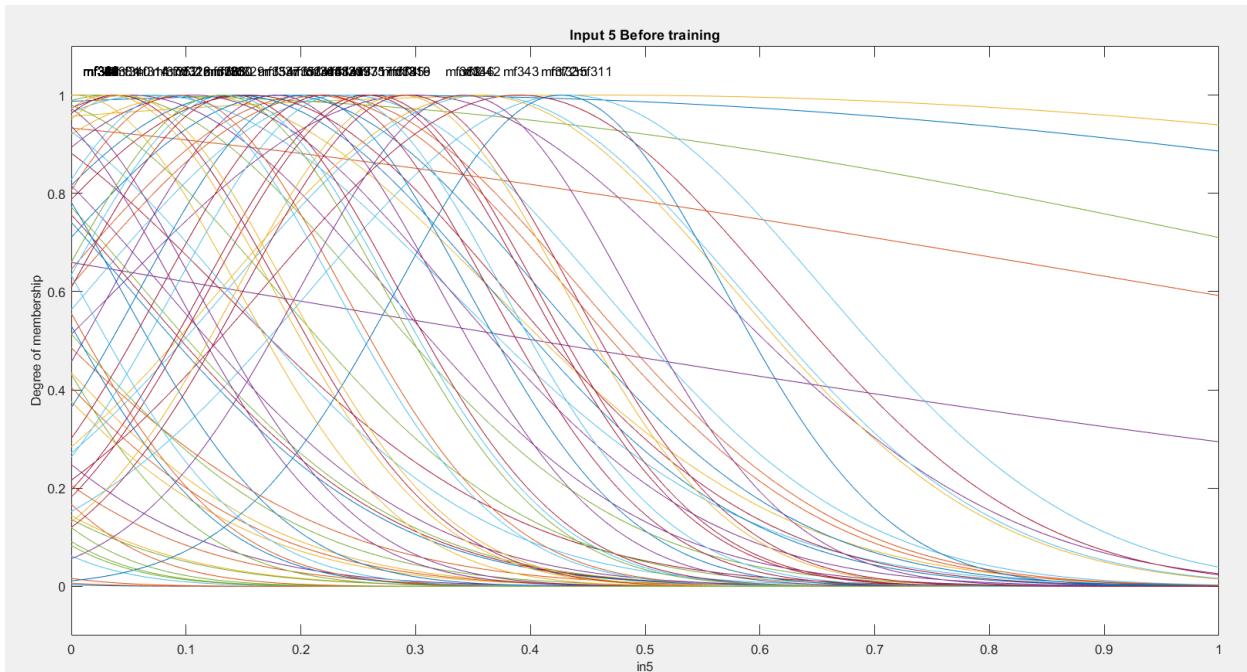
- Διαγράμματα Βέλτιστου Μοντέλου:

## Συναρτήσεις Συμμετοχής Πριν την εκπαίδευση – Input 1



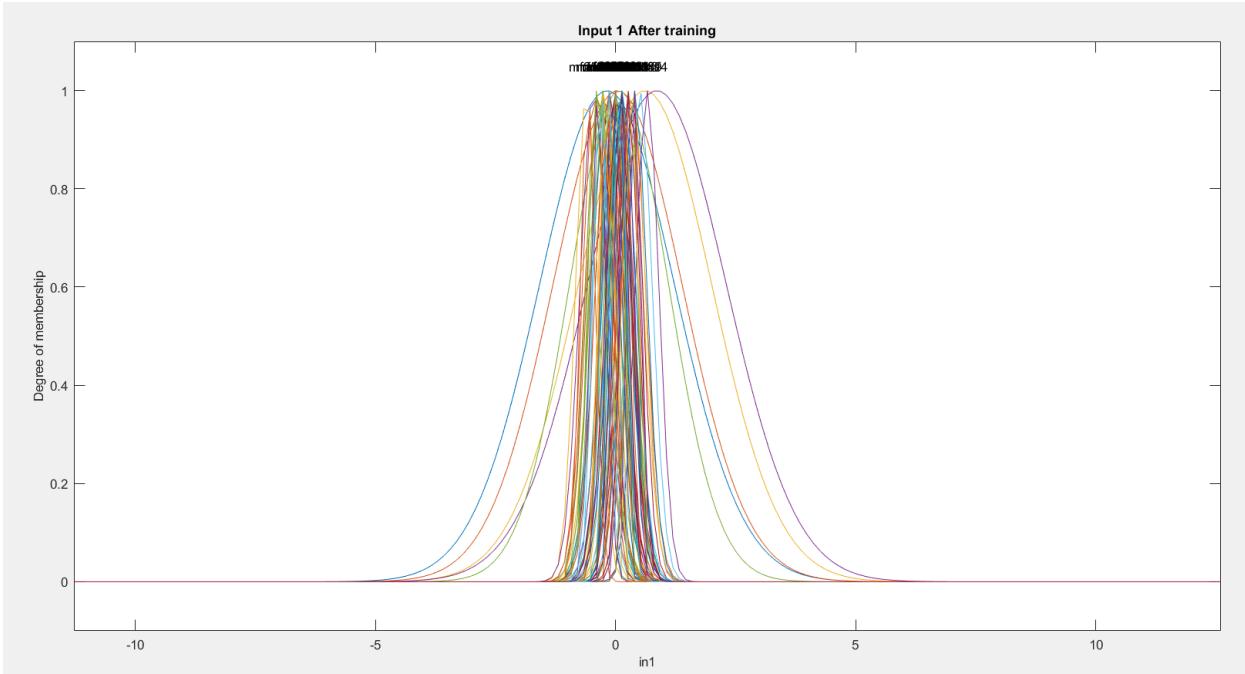
Σχήμα 120: mf Before Training – input 1

## Συναρτήσεις Συμμετοχής Πριν την εκπαίδευση – Input 5



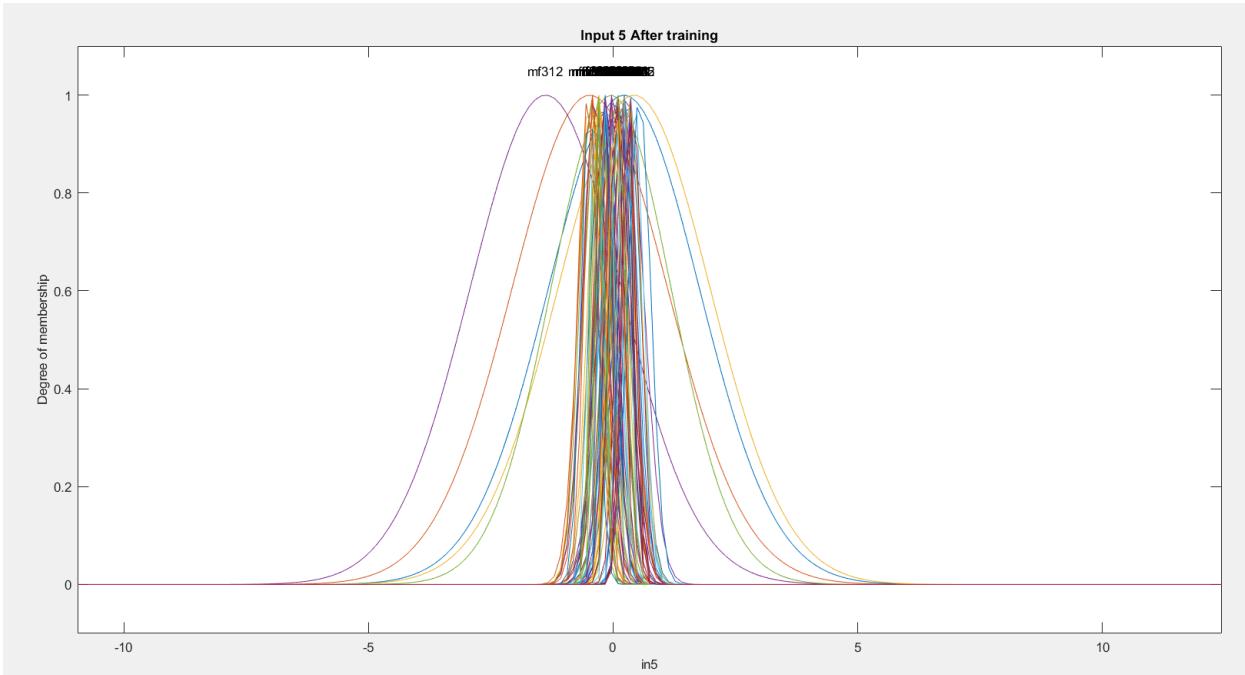
Σχήμα 121: mf Before Training – Input 5

### Συναρτήσεις Συμμετοχής Μετά την εκπαίδευση – Input 1



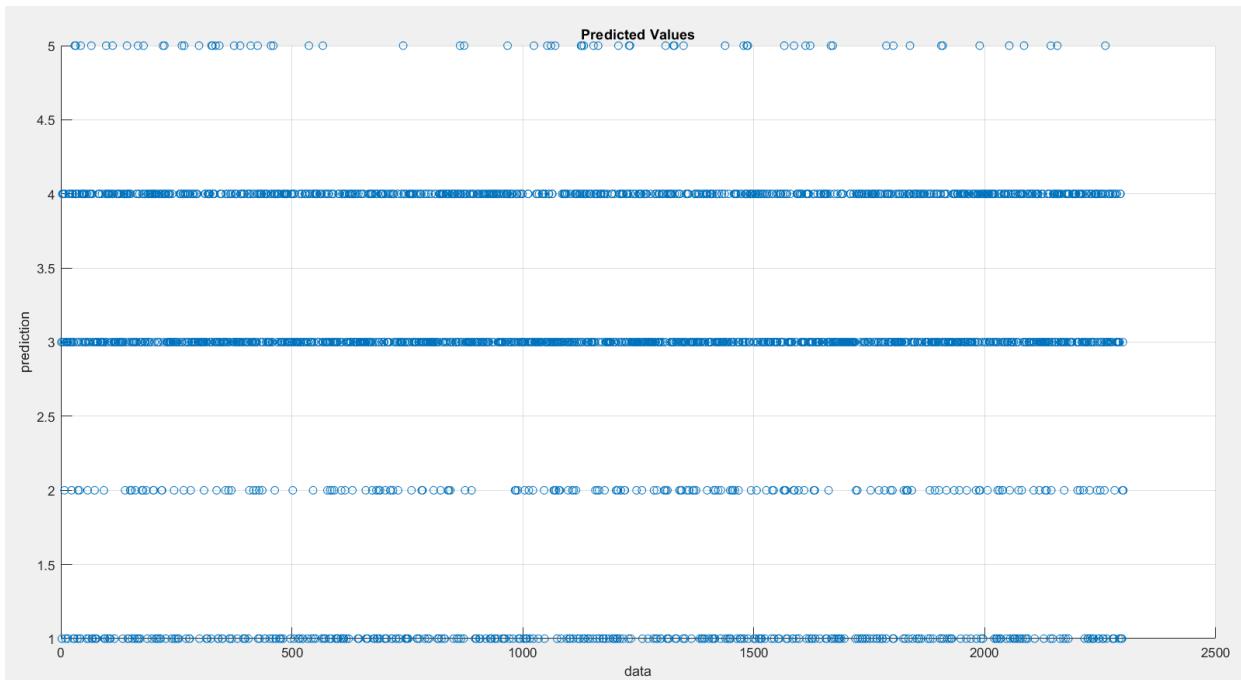
Σχήμα 122: mf After Training – input 1

### Συναρτήσεις Συμμετοχής Μετά την εκπαίδευση – Input 5



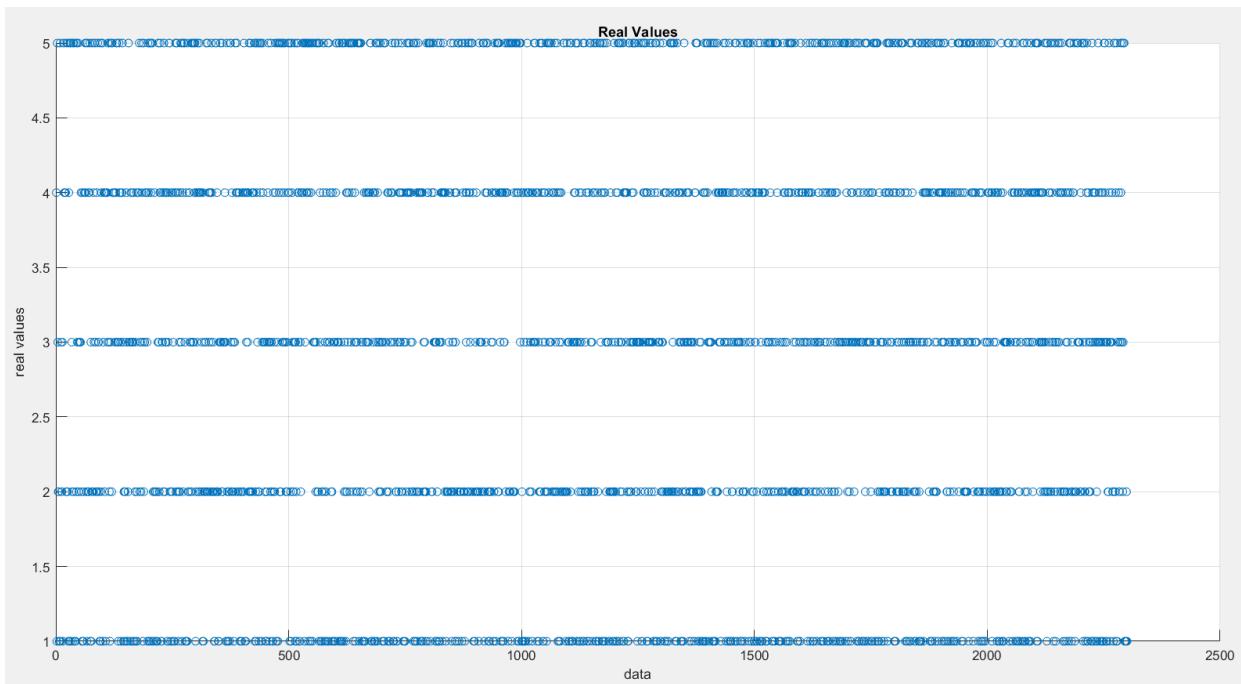
Σχήμα 123: mf After Training – input 5

## Τιμές Πρόβλεψης



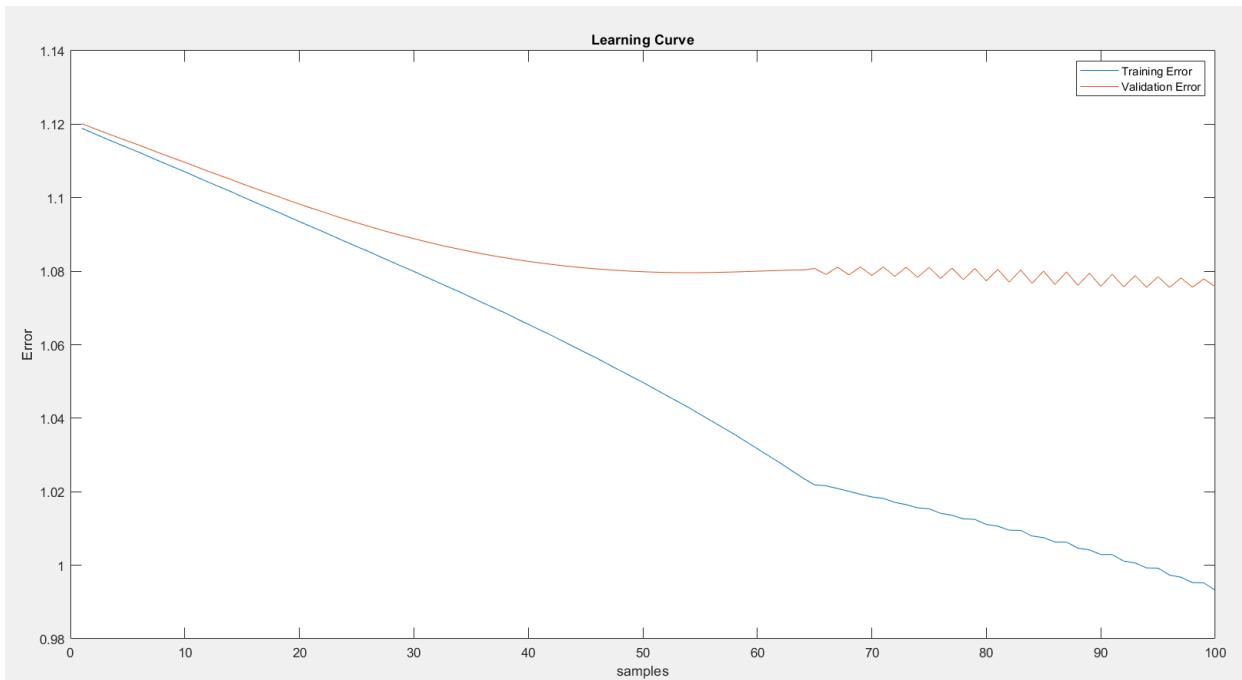
Σχήμα 124: Prediction Values

## Πραγματικές Τιμές



Σχήμα 125: Real Values

## Καμπύλη Εκμάθησης



Σχήμα 126: Learning Curve

- Συμπεράσματα

Αφού τρέξαμε το script “classification\_Optimal\_TSK\_model.m” με 11 features και 0.2 radius ως το βέλτιστο μοντέλο, εξάγαμε τους παραπάνω πίνακες και διαγράμματα.

Το μοντέλο χρησιμοποιεί κατά μέσο όρο 92 κανόνες με την μέθοδο Subtractive Clustering, σε αντίθεση με την Grid Partition, η οποία θα χρειαζόταν  $3^{11}$  κανόνες, αν υποθέσουμε ότι θα είχαμε 3 ασαφή σύνολα για κάθε είσοδο. Αυτό θα καθιστούσε το μοντέλο μη λειτουργικό.

Παρατηρούμε ότι πολλές Συναρτήσεις Συμμετοχής επικαλύπτονται, πράγμα που σημαίνει ότι αρκετά ασαφή σύνολα είναι αχρείαστα και μπορούν να αφαιρεθούν.

Από τους Πίνακες του Accuracy και του Error Matrix συμπεραίνουμε τα εξής:

- Τα δείγματα που ανήκουν στην κλάση 1, έχουν πιθανότητα να ταξινομηθούν σε αυτή την κλάση με πιθανότητα PA1 = 72.78%. Αντίστοιχα, αυτά που ανήκουν στην κλάση 2 ταξινομούνται στην σωστή κλάση με πιθανότητα PA2 = 10.26%, στην κλάση 3 με PA3 = 61.70%, στην κλάση 4 με PA4 = 48.26% και στην κλάση 5 με PA5 = 5.87%.
- Αν επιλέξουμε με τυχαίο τρόπο ένα δείγμα από το σύνολο του dataset, το οποίο ταξινομήθηκε στην κλάση 1 από το βέλτιστο μοντέλο, η πιθανότητα να ανήκει όντως στην κλάση 1 είναι UA1 = 91.99%. Αντίστοιχα, ένα δείγμα που ταξινομήθηκε στην κλάση 2, ανήκει όντως σε αυτή την κλάση με πιθανότητα UA2 = 23.51%, στην κλάση 3 με UA3 = 30.92%, στην κλάση 4 με UA4 = 29.11% και στην κλάση 5 με UA5 = 47.11%.

Με βάση τα παραπάνω συμπεράσματα και το Producer's Accuracy, το μοντέλο προβλέπει με ικανοποιητική ακρίβεια τις κλάσεις 1 και 3, με μέτρια ακρίβεια την κλάση 4, και με κακή ακρίβεια τις κλάσεις 2 και 5.

Με βάση το User's Accuracy το μοντέλο προβλέπει σε ικανοποιητικό βαθμό την κλάση 1 και σε μέτριο την κλάση 5.

Η σημαντικότερη κλάση είναι η 1, γιατί είναι αυτή που περιλαμβάνει τα θετικά κρούσματα σε επιληπτική κρίση, ενώ στις υπόλοιπες τέσσερεις κλάσεις τα δείγματα είναι αρνητικά. Η ακρίβεια της κλάσης 1 είναι πολύ υψηλή με πιθανότητες PA1 = 72.78% και UA1 = 91.99%.

Το μοντέλο προβλέπει με μεγάλη ακρίβεια τα θετικά δείγματα (κλάση 1), επομένως λειτουργεί ικανοποιητικά.