

Υπολογιστική Νοημοσύνη  
Τμήμα Μηχανικών Η/Υ και Πληροφορικής  
Πανεπιστήμιο Πατρών  
Εργασία ΥΝ 2020  
Μέρος Α΄ Συνεργατικό Φιλτράρισμα  
με Χρήση Νευρωνικών Δικτύων  
για Συστάσεις Ταινιών

*Παναγιώτης Χριστόπουλος 1054409*



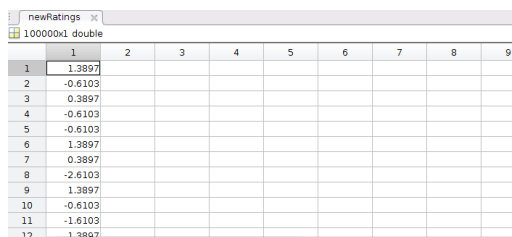
Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

A1

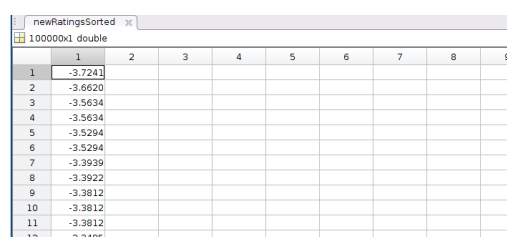
Προεπεξεργασία και  
Προετοιμασία δεδομένων

## α) Κεντράρισμα (centering)

Στο αρχείο u.data του dataset ml-100k βλέπουμε ότι οι αξιολογήσεις των χρηστών κυμαίνονται στο σύνολο [1,5]. Είναι φανερό ότι δεν επιτυγχάνεται αντικειμενικότητα, αφού κάθε χρήστης τείνει να βαθμολογεί τις προτιμήσεις του σε διάφορα υποσύνολα. Για αυτό το λόγο, αναπτύχθηκε το script centering.m το οποίο βρίσκεται στο φάκελο του παραδοτέου. Αφού ταξινομούνται τα δεδομένα σύμφωνα με το id του χρήστη που καταχώρησε το rating, εκτελείται η λούπα του προγράμματος. Για κάθε user βρίσκεται το άθροισμα των αξιολογήσεων και διαιρείται με τον αριθμό τους ώστε να βγει μία μέση τιμή για τον καθένα. Στη συνέχεια προκύπτουν τροποποιημένες αξιολογήσεις, αφαιρώντας από τις αρχικές τη μέση τιμή. Σύμφωνα με το νέο πίνακα newRatingsSorted, τα νέα ratings κυμαίνονται από -3.72 μέχρι 3.50. Κρίνεται σωστό να εφαρμοστεί η μέθοδος στα στοιχεία εκπαίδευσης για να επιτευχθεί ομοιογένεια μεταξύ των αξιολογήσεων κάθε χρήστη.



	1	2	3	4	5	6	7	8	9
1	1.3897								
2	-0.6103								
3	0.3897								
4	-0.6103								
5	-0.6103								
6	1.3897								
7	0.3897								
8	-2.6103								
9	1.3897								
10	-0.6103								
11	-1.6103								
12	1.3897								



	1	2	3	4	5	6	7	8	9
1	-3.7241								
2	-3.6620								
3	-3.5634								
4	-3.5634								
5	-3.5294								
6	-3.5294								
7	-3.3939								
8	-3.3922								
9	-3.3812								
10	-3.3812								
11	-3.3812								
12	-3.3486								

α) Οι αξιολογήσεις μετά το κεντράρισμα      β) Οι αξιολογήσεις σε αύξουσα σειρά

## β) Ελλιπείς τιμές:

Για να αντιμετωπιστεί το πρόβλημα των ελλιπών τιμών του δείγματος, χρησιμοποιώντας τα κεντραρισμένα δεδομένα του προηγούμενου ερωτήματος, δημιουργήθηκε ένας βρόχος που γεμίζει τους παρακάτω πίνακες:

- `dataTable(R)`: Πίνακας με ταξινομημένα στοιχεία ως εξής:  
Γραμμή: User Id Στήλη: Item id Περιεχόμενο: Rating  
Για παράδειγμα, αν ο χρήστης με id 12 βαθμολόγησε την ταινία με id 19 με βαθμό 3, το περιεχόμενο του κελιού της 12ης γραμμής και 19ης στήλης του νέου πίνακα έχει την τιμή 3.
- `weightTable(W)`: Ο `weightTable` είναι ένας πίνακας βαρών ο οποίος ως εξής: Αν υπάρχει βαθμολογία για μία ταινία τότε το αντίστοιχο κελί έχει τον αριθμό 1, αλλιώς έχει τον αριθμό 0.

Με αυτό τον τρόπο, στη συνέχεια, θα μπορούμε να λάβουμε υπ'όψιν μόνο τις ταινίες τις οποίες ένας χρήστης έχει βαθμολογήσει. Θεωρείται επιβόλαιη η αυθαίρετη συμπλήρωση αξιολογήσεων, αφού ο μέσος χρήστης έχει βαθμολογήσει λιγότερες από τις μισές ταινίες, επομένως το σύστημά μας δε θα ήταν τόσο αξιόπιστο.

	1	2	3	4	5	6	7	8	9
1	1.3897	-0.6103	0.3897	-0.6103	-0.6103	1.3897	0.3897	-2.6103	1.3897
2	0.2903	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0
5	1.1257	0.1257	0	0	0	0	0	0	0
6	0.3649	0	0	0	0	0	-1.6351	0.3649	0.3649
7	0	0	0	1.0347	0	0	1.0347	1.0347	1.0347
8	0	0	0	0	0	0	-0.7966	0	0
9	0	0	0	0	0	0.7273	-0.2727	0	0
10	-0.2065	0	0	-0.2065	0	0	-0.2065	0	-0.2065
11	0	0	0	0	0	0	0.5359	1.5359	0

α) Ο πίνακας ταινία,χρήστης

	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	1	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0
5	1	1	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	1	1
7	0	0	0	1	0	0	0	1	1
8	0	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	1	1	0
10	1	0	0	1	0	0	0	1	0
11	0	0	0	0	0	0	0	1	1

β) Ο πίνακας βαρών

## γ) Κανονικοποίηση (rescaling):

Για την κανονικοποίηση του ΤΝΔ που παρουσιάζεται, επιλέχθηκε το πεδίο τιμών της σιγμοειδούς συναρτήσεως  $[0,1]$ . Για να επιτευχθεί αυτό, χρησιμοποιήθηκε η εντολή:

```
normalize(N, 'range');
```

Η υλοποίηση της κανονικοποίησης βρίσκεται στο συνημμένο normalization.m

	1	2	3	4	5	6	7	8	9
1	0.7071	0.4306	0.5688	0.4306	0.4306	0.7071	0.5688	0.1540	0.7071
2	0.5551	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0
5	0.6706	0.5323	0	0	0	0	0	0	0
6	0.5654	0	0	0	0	0	0.2889	0.5654	0.5654
7	0	0	0	0.6580	0	0	0.6580	0.6580	0.6580
8	0	0	0	0	0	0	0.4048	0	0
9	0	0	0	0	0	0.6155	0.4772	0	0
10	0.4864	0	0	0.4864	0	0	0.4864	0	0.4864
11	0	0	0	0	0	0	0.5890	0.7273	0

Ο ενημερωμένος πίνακας με τις κανονικοποιημένες αξιολογήσεις

## δ) Διασταυρούμενη Επικύρωση (cross-validation):

Για τη Διασταυρούμενη Επικύρωση του δικτύου διασπάστηκε το αρχείο των ratings σε 5 διαμερίσεις ίδιου μεγέθους ως εξής:

```
c = crossvalind('Kfold', rating, 5);
```

Με αυτό τον τρόπο, χρησιμοποιούνται στη συνέχεια οι 4 διαμερίσεις ως σύνολα εκπαίδευσης και η τελευταία διαμέριση ως σύνολο ελέγχου.

## A2

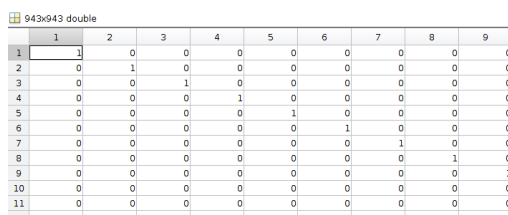
### Επιλογή αρχιτεκτονικής

## α) Σημασία Root Mean Squared Error (RMSE) και Mean Absolute Error (MAE) στο συγκριμένο πρόβλημα

Ως Μέσο Απόλυτο Σφάλμα (MAE) ορίζεται ο μέσος όρος των διαφορών μεταξύ πρόβλεψης και τιμής θεωρώντας όλα τα πιθανά λάθη ισοβαρή. Η Ρίζα του Μέσου Τετραγωνικού Σφάλματος (RMSE) είναι η τετραγωνική ρίζα του μέσου όρου των τετραγώνων των διαφορών μεταξύ μίας πρόβλεψης και της τιμής στην οποία αντιστοιχεί. Στο τρέχον πρόβλημα, θεωρείται καλύτερη η χρήση του RMSE, αφού αυτή η μετρική δίνει υψηλό βάρος στα μεγαλύτερα σφάλματα. Αυτή η ιδιότητα συνδυασμένη με την κανονικοποίηση των παραπάνω ερωτημάτων προσδίδει μεγαλύτερη ακρίβεια στο πείραμα.

## β) Πόσες εισόδους θα χρειαστείτε στο TNΔ, δεδομένου ότι μια είσοδος πρέπει να αναπαριστά έναν χρήστη;

Το TNΔ θα χρειαστεί ως είσοδο έναν πίνακα 943 γραμμών και 943 στηλών, συνεπώς θα έχει ως είσοδο συνολικά 889249 ψηφία. Με τη βοήθεια του one-hot encoding τοποθετούμε 1 στη θέση (userId,userId) του πίνακα και 0 σε όλες τις υπόλοιπες. Η παραπάνω υλοποίηση βρίσκεται στο συνημμένο script oneHotEncoding.m



	1	2	3	4	5	6	7	8	9
1	1	0	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0	0
4	0	0	0	1	0	0	0	0	0
5	0	0	0	0	1	0	0	0	0
6	0	0	0	0	0	1	0	0	0
7	0	0	0	0	0	0	1	0	0
8	0	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	0	0	1
10	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0

One-hot Encoding για τον πίνακα εισόδου

γ) Πόσους νευρώνες θα χρειαστείτε στο επίπεδο εξόδου, δεδομένου ότι η έξοδος πρέπει να αναπαριστά τις αξιολογήσεις του χρήστη για όλες τις ταινίες;

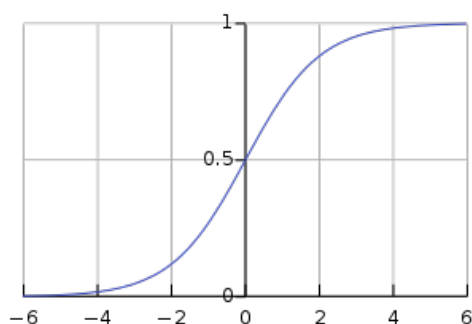
Στο επίπεδο εξόδου θα χρειαστεί ένας πίνακας 943 γραμμών και 1682 στηλών. Ο πίνακας αυτός θα περιέχει τις αξιολογήσεις όλων των χρηστών για όλες τις ταινίες. Στο script του ερωτήματος της κανονικοποίησης παραπάνω, δημιουργείται ο πίνακας rNormalized που χρησιμοποιείται ως target data στο δίκτυο

	1	2	3	4	5	6	7	8	9
1	0.7071	0.4306	0.5688	0.4306	0.4306	0.7071	0.5688	0.1540	0.7071
2	0.5551	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0
5	0.6706	0.5323	0	0	0	0	0	0	0
6	0.5654	0	0	0	0	0	0.2889	0.5654	0.5654
7	0	0	0	0.6580	0	0	0.6580	0.6580	0.6580
8	0	0	0	0	0	0	0.4848	0	0
9	0	0	0	0	0	0.6155	0.4772	0	0
10	0.4864	0	0	0.4864	0	0	0.4864	0	0.4864
11	0	0	0	0	0	0	0	0.5890	0.7273
12	0	0	0	0	0	0	0	0	0

Ο πίνακας rNormalized

δ) Να επιλέξετε κατάλληλη συνάρτηση ενεργοποίησης για τους κρυφούς κόμβους και να τεκμηριώσετε την επιλογή σας.

Για την εκπαίδευση του δικτύου χρησιμοποιείται η σιγμοειδής συνάρτηση ενεργοποίησης για τους κρυφούς κόμβους. Τα δεδομένα μας έχουν ήδη κανονικοποιηθεί στο πεδίο τιμών της λογιστικής συνάρτησης, για αυτό το λόγο χρησιμοποιείται και για τους κρυφούς κόμβους.



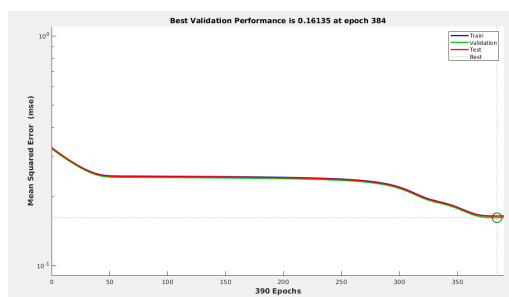
Η σιγμοειδής συνάρτηση

ε) Ποια συνάρτηση ενεργοποίησης θα χρησιμοποιήσετε για το επίπεδο εξόδου;

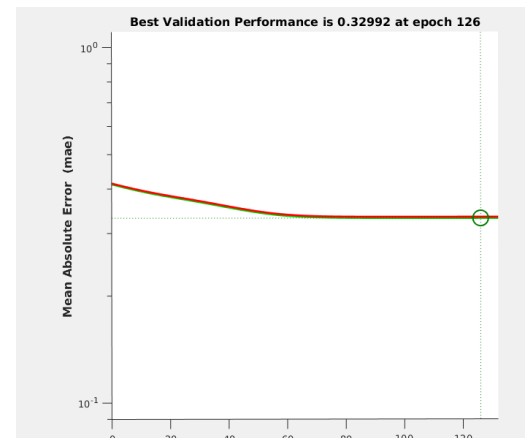
Για το επίπεδο εξόδου θα χρησιμοποιηθεί η γραμμική συνάρτηση. Επιλέχθηκε για την εξής ιδιότητα: Στους νευρώνες με διακριτή τιμή η είσοδος αναπαράγεται γραμμικά στην έξοδο. Στους νευρώνες με σιγμοειδή συνάρτηση (Κρυφούς) η έξοδος μπορεί να είναι οποιοσδήποτε πραγματικός αριθμός από το 0 έως το 1.

στ) Πειραματιστείτε με 3 διαφορετικές τιμές για τον αριθμό των νευρώνων του κρυφού επιπέδου και συμπληρώστε τον παρακάτω πίνακα

Για την εκπαίδευση του δικτύου νευρώνων χρησιμοποιήθηκε το neural net fitting της MATLAB. Αρχικά, χωρίζεται ο πίνακας εισόδου ανά index, έτσι ώστε να εξετάζονται διαφορετικά σημεία του ως training data και test data σε κάθε εκτέλεση της λούπας. Μετριέται το συνολικό performance του νευρωνικού δικτύου και στις 5 προσπελάσεις και διαιρείται δια 5 για την εύρεση του μέσου όρου λάθους (είτε RMSE είτε MAE).

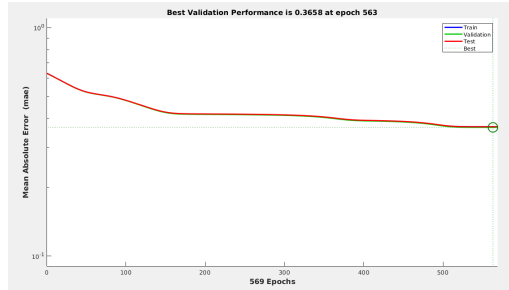


α) Ρίζα του Μέσου Τετραγωνικού Σφάλματος για 10 νευρώνες στο κρυφό επίπεδο

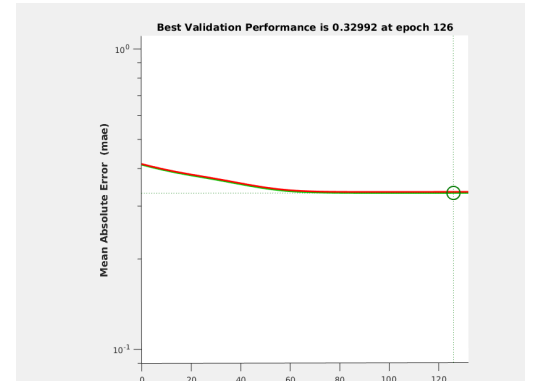


β) Μέσο Απόλυτο Σφάλμα για 10 νευρώνες

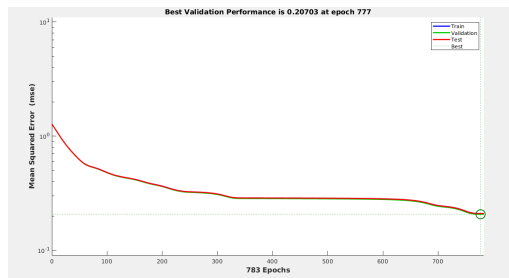




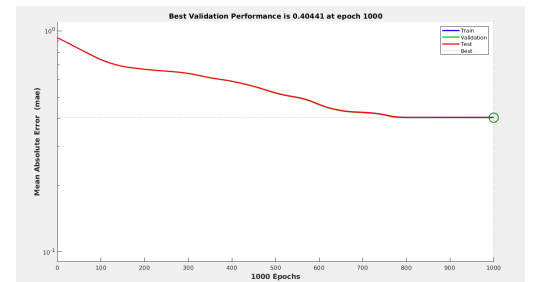
α) Ρίζα του Μέσου Τετραγωνικού Σφάλματος για 20 νευρώνες στο κρυφό επίπεδο



β) Μέσο Απόλυτο Σφάλμα για 20 νευρώνες



α) Ρίζα του Μέσου Τετραγωνικού Σφάλματος για 50 νευρώνες στο κρυφό επίπεδο



β) Μέσο Απόλυτο Σφάλμα για 50 νευρώνες

Νευρώνες στο Κρυφό Επίπεδο	RMSE	MAE
H = 10	0.4465	0.3645
H = 20	0.4785	0.3904
H = 50	0.7147	0.4237

Το script του νευρωνικού δικτύου βρίσκεται στο φάκελο των συνημμένων script

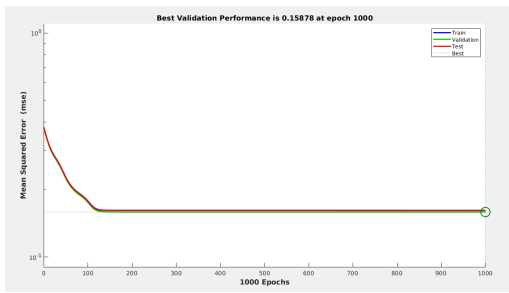
Τα screenshot των αποτελεσμάτων των σφαλμάτων βρίσκονται στο φάκελο Outputs.

**A3**

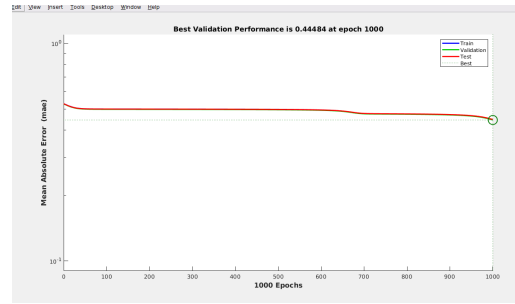
Μεταβολές στο ρυθμό  
εκπαίδευσης και σταθεράς  
ορμής

Για την παρατήρηση των μεταβολών στο ρυθμό εκπαίδευσης, έγινε αλλαγή της Train Function σε trainingda, ώστε να μπορεί να ληφθεί υπ'όψιν το κριτήριο της ορμής. Το νευρωνικό δίκτυο που αναπτύχθηκε για το ερώτημα αυτό περιέχει 10 κρυφούς νευρώνες, διότι όπως παρατηρήθηκε στο προηγούμενο ερώτημα αποδίδει καλύτερα από τις άλλες επιλογές. Θεωρητικά, ο συντελεστής  $m$  πρέπει να είναι πάντα μικρότερος από 1 για να αποφεύγεται η υπερεκπαίδευση.

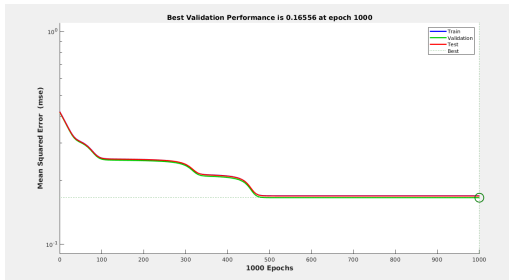
$\eta$	$m$	RMSE	MAE
0.001	0.2	0.1828	0.3886
0.001	0.6	0.1885	0.3743
0.05	0.6	0.321	0.3847
0.1	0.6	0.4528	0.3737



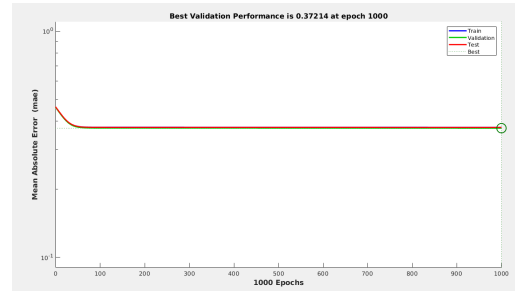
α) MSE  $\eta=0.001$   $m=0.2$



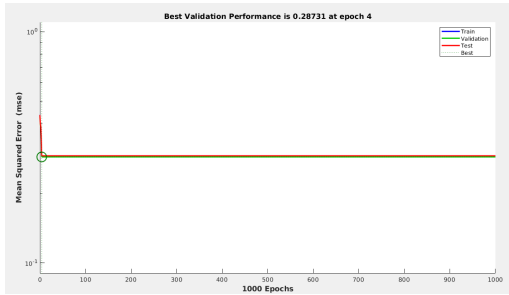
β) MAE  $\eta=0.001$   $m=0.2$



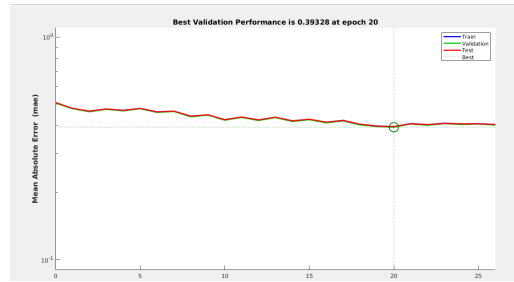
α) MSE  $\eta=0.001$   $m=0.6$



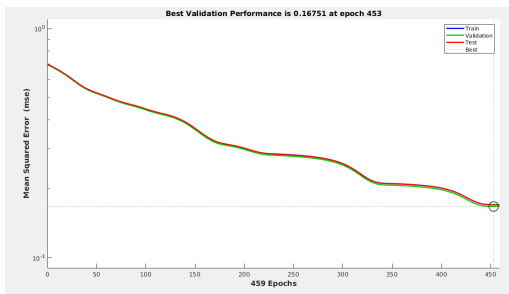
β) MAE  $\eta=0.001$   $m=0.6$



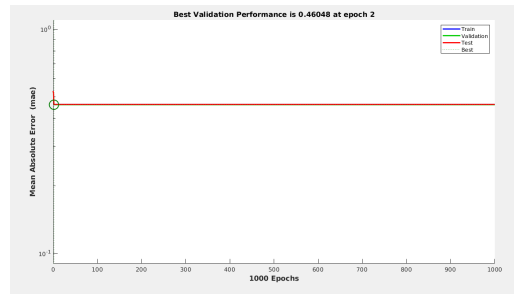
$\alpha)$  MSE  $\eta=0.05$   $m=0.6$



$\beta)$   $\eta=0.05$   $m=0.6$



$\alpha)$  MSE  $\eta=0.1$   $m=0.6$



$\beta)$  MAE  $\eta=0.05$   $m=0.6$

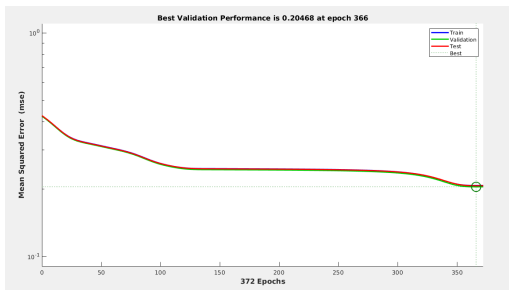
Το script του ερωτήματος είναι το neuralMomentum.m

A4

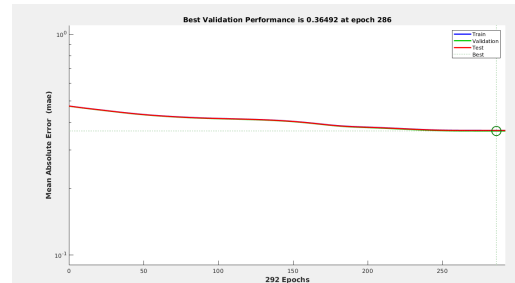
Μεταβολές στο ρυθμό  
εκπαίδευσης και σταθεράς  
ορμής

Για την ομαλοποίηση του νευρωνικού δικτύου χρησιμοποιήθηκε η L1 κανονικοποίηση. Επιλέχθηκε η L1(Lasso Regression) λόγω της ιδιότητάς της να συρρικνώνει τα ανεπιθύμητα στοιχεία σε σχεδόν μηδενικά βάρη.

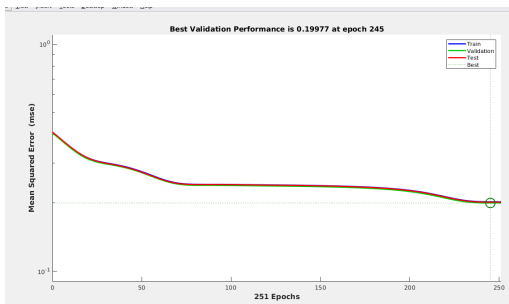
συντελεστής φθοράς	RMSE	MAE
0.1	0.1778	0.3714
0.5	0.2026	0.3491
0.9	0.2080	0.3522



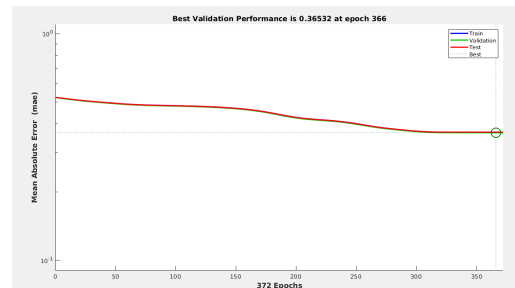
α) MSE  $r=0.1$



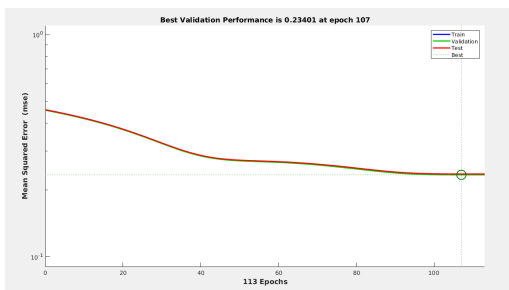
β) MAE  $r=0.1$



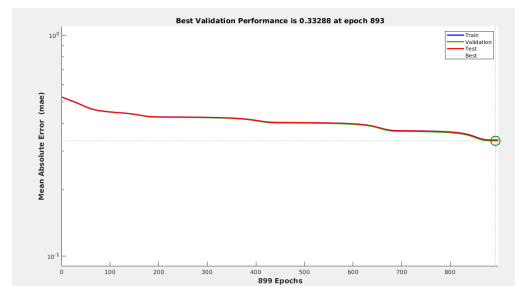
α) MSE  $r=0.5$



β) MAE  $r=0.5$



α) MSE  $r=0.9$



β) MAE  $r=0.9$

Το script του ερωτηματος είναι το neuralL1Regularization.m