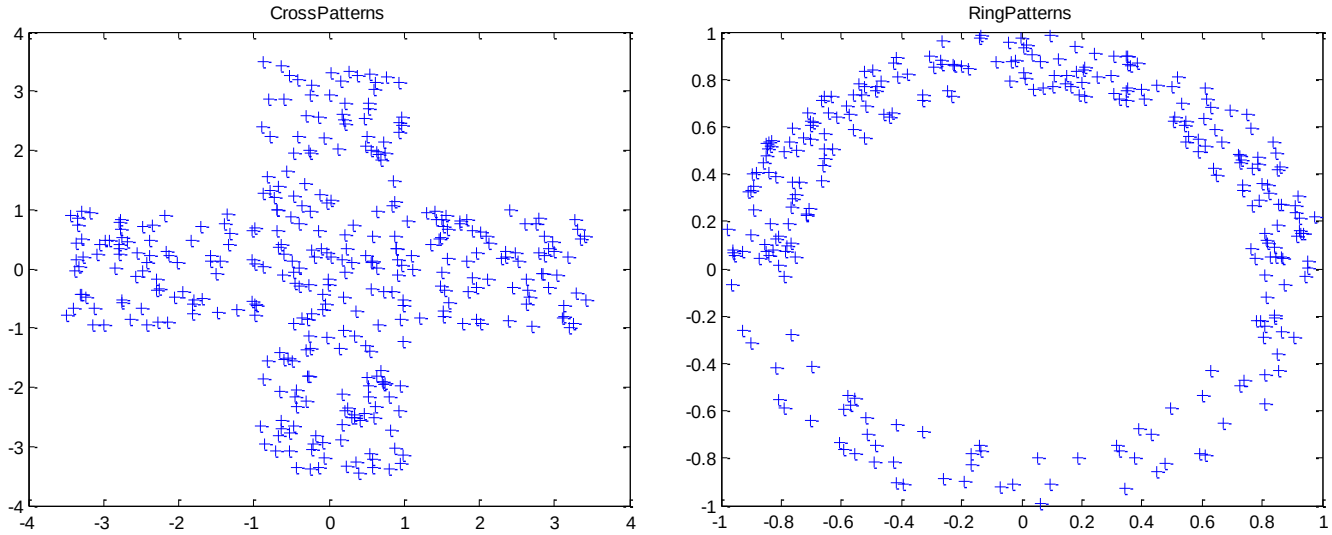
Μελέτη και ανάλυση SOM

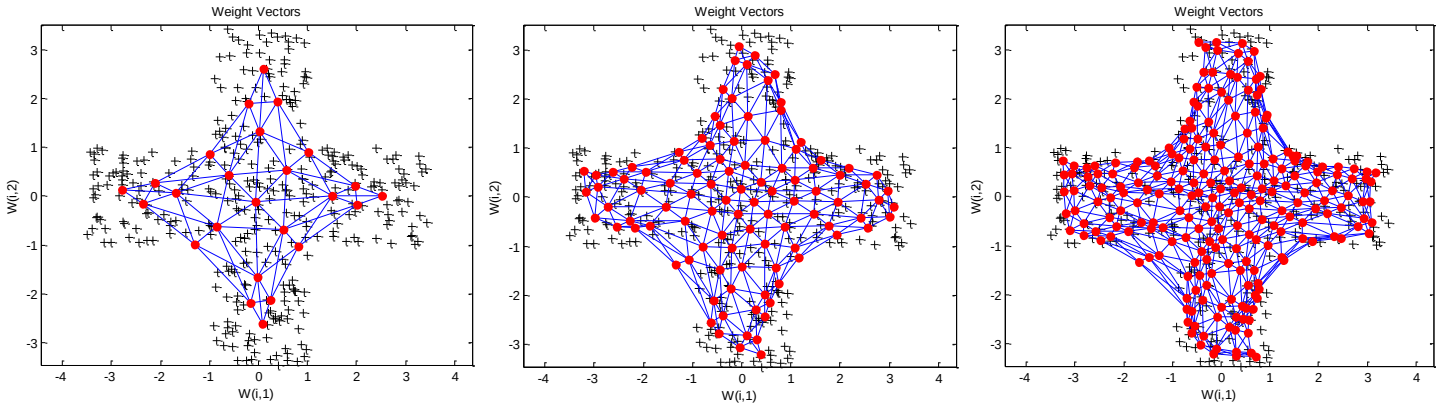
2A. Στο μέρος αυτό κατασκευάστηκαν διδιάστα πλέγματα και στη συνέχεια εκπαιδεύτηκαν χρησιμοποιώντας ώς σύνολα δεδομένων τα **CrossData.m** και **RingData.m** που φαίνονται στο σχήμα 1.



Σχήμα 1 : CrossData.m (αριστερά) και RingData.m (δεξιά)

Το γεγονός οτι χρησιμοποιήσαμε διδιάστατο πλέγμα μας επιτρέπει να οπτικοποιήσουμε και τα βάρη των νευρώνων. Συγκεκριμένα ο κάθε νευρώνας θα χαρακτηρίζεται απο δύο βάρη και μπορούμε να θεωρήσουμε αυτό το διατεταγμένο ζεύγος βαρών ώς συντεταγμένες του νευρώνα στο διδιάστατο χώρο.Φυσικά αυτό είναι απλά μια παραδοχή και δεν σημαίνει σε καμία περίπτωση οτι οι νευρώνες μετακινούνται απο το πλέγμα στο οποίο έχουν οριστεί.

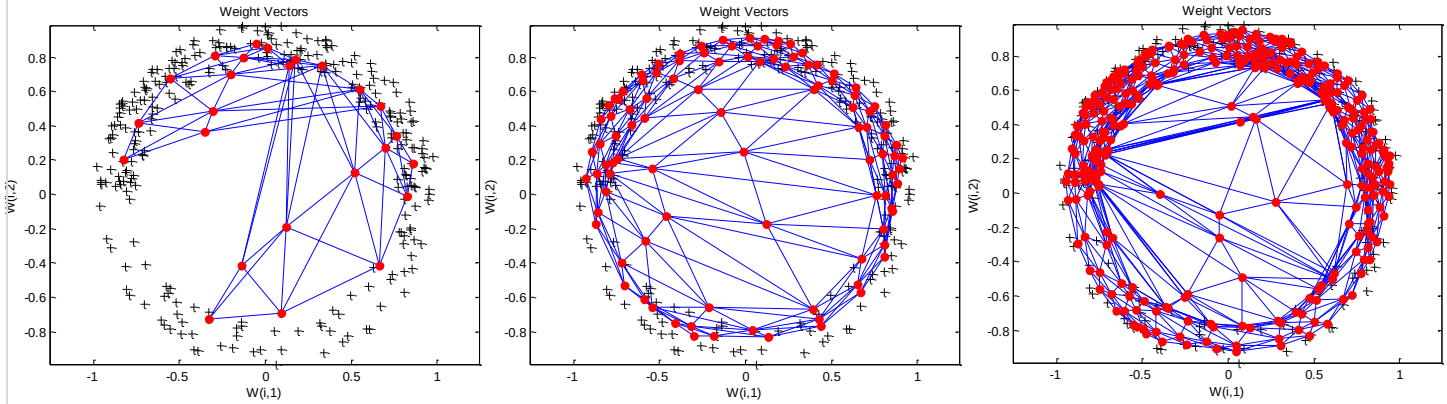
Αρχικά λοιπόν δημιουργήθηκαν οι ζητούμενες συναρτήσεις και στη συνέχεια για το πρώτο μέρος της εργασίας δημιουργήθηκε το m-file **askisi_2a.m** οπου στην αρχή του κώδικα δίνεται η δυνατότητα να επιλεχθούν οι βασικές παράμετροι για την εκπαίδευση του συστήματος, όπως το μέγεθος του grid και ο αριθμός των βημάτων orderSteps. Στο παρακάτω σχήμα παρουσιάζονται μερικά αποτελέσματα για διαφορετικές επιλογές των δύο αυτών παραμέτρων. Σημειώνουμε οτι στην περίπτωση που έχουμε μικρού μεγέθους grid δεν χρειάζονται πολλά βήματα για τη σύγκλιση των τιμών των νευρώνων.



Σχήμα 2 : αριστερά: grid 5x5 και 5 βήματα, κέντρο: grid 10x10 και 50 βήματα, δεξιά: grid 18x18 και 500 βήματα

Παρατηρούμε λοιπόν το αναμενόμενο, οτι δηλαδή οτι στην περίπτωση που έχουμε μεγαλύτερο πλέγμα τα βάρη των νευρώνων προσεγγίζουν καλύτερα την κατανομή των χαρακτηριστικών εισόδου. Φυσικά για να επιτευχθεί κάτι τέτοιο χρειάζονται και πολύ περισσότερα βήματα. Συγκεκριμένα για την περίπτωση που είχαμε πλέγμα 5x5 (λόγω της φύσης του προβλήματος χρησιμοποιήσαμε τετραγωνικά grid) χρειάστηκαν μόνο 5 βήματα για να καταλήξουμε σε μια σταθερή κατάσταση των βαρών των νευρώνων. Προφανώς στην περίπτωση που χρησιμοποιήθηκε πλέγμα 18x18 και συνολικά 500 βήματα εκπαίδευσης όπως φαίνεται στο σχήμα 2 τα βάρη των νευρώνων έχουν μια πολύ καλή αντιστοίχηση με τις συντεταγμένες των προτύπων. Κάτι τέτοιο επιτυγχάνεται πολύ νωρίτερα, περίπου απο τα 100 βήματα, αλλά αφήνουμε την εκπαίδευση του δικτύου να συνεχιστεί ωστε να παραχθεί μια πιο σταθερή κατάσταση.

Η ίδια διαδικασία επαναλήφθηκε με πρότυπα τα ringpatterns και τα αποτελέσματα για διάφορες τιμές φαίνονται στο παρακάτω σχήμα. Αναφέρουμε εδώ οτι στα συγκεκριμένα πρότυπα χρειάστηκαν περισσότερα βήματα για να καταλήξουμε σε κάποια σύγκλιση κατα την ανανέωση των βαρών.

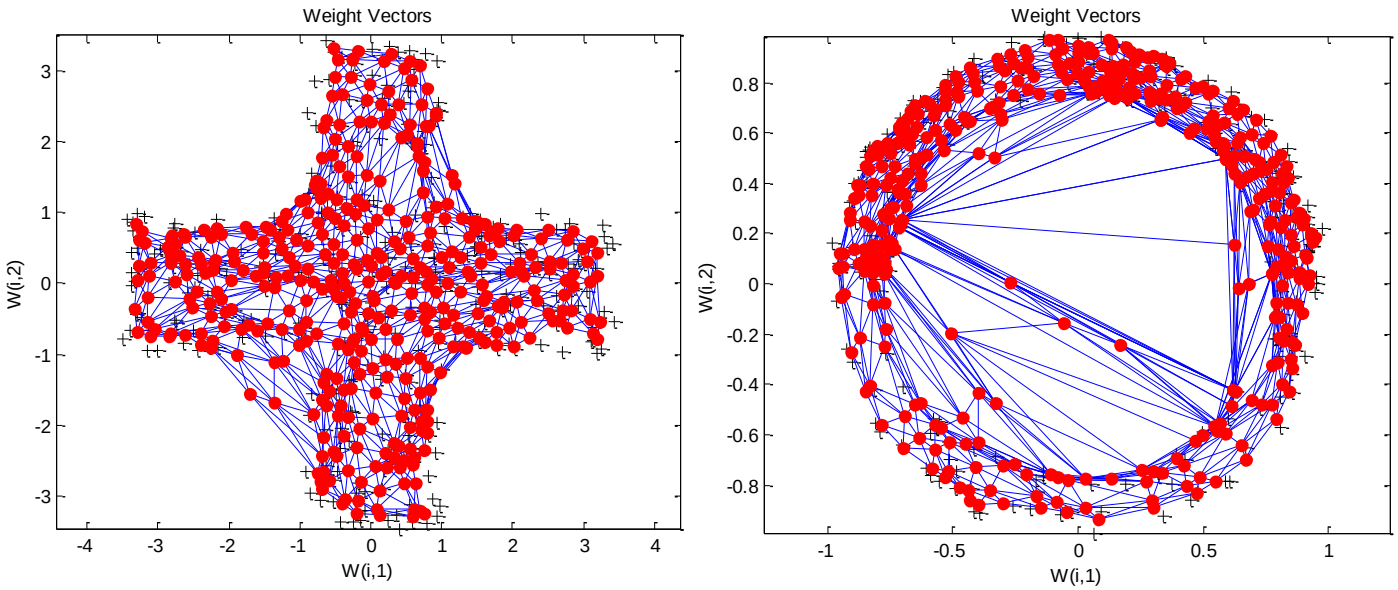


Σχήμα 3 : αριστερά: grid 5x5 και 5 βήματα, κέντρο: grid 10x10 και 50 βήματα, δεξιά: grid 18x18 και 500 βήματα

Φυσικά έγιναν δοκιμές και με διαφορετικές παραμέτρους orderLR, tuneLR, και tuneND και δίνεται η δυνατότητα αλλαγής αυτών στην αρχή του παραδωταίου κώδικα. Τα αποτελέσματα δεν είχαν μεγάλες αποκλίσεις απο τα ήδη παρουσιαζόμενα εκτός και αν αλλάξουμε κατα πολύ τις default τιμές τους. Ωστόσο παρατηρήθηκε οτι με οποιονδήποτε συνδιασμό παραμέτρων κι αν ελέγχθηκε δεν πάντα υπήρχαν νευρώνες που τα βάρη τους αντιστοιχίζονται σε ενδιάμεσα σημεία του δίσκου. Ουσιαστικά υπάρχουν νευρώνες που ¨αμφιταλαντέυονται¨ απο τη μία άκρη στην άλλη και άλλοι που μένουν ακίνητοι(τα βάρη τους). Βέβαια ικανοποιητικό είναι το οτι η κατανομή των νευρώνων είναι αντίστοιχη με την κατανομή των προτύπων καθώς όπως φαίνειται και απο τα σχήματα στις περιοχές που έχουμε περισσότερα πρότυπα έχουν αντιστοιχηθεί περισσότεροι νευρώνες, οπότε το som περιγράφει εκτός απο τη χωρική κατανομή και την κατανομή πυκνότητας των προτύπων.

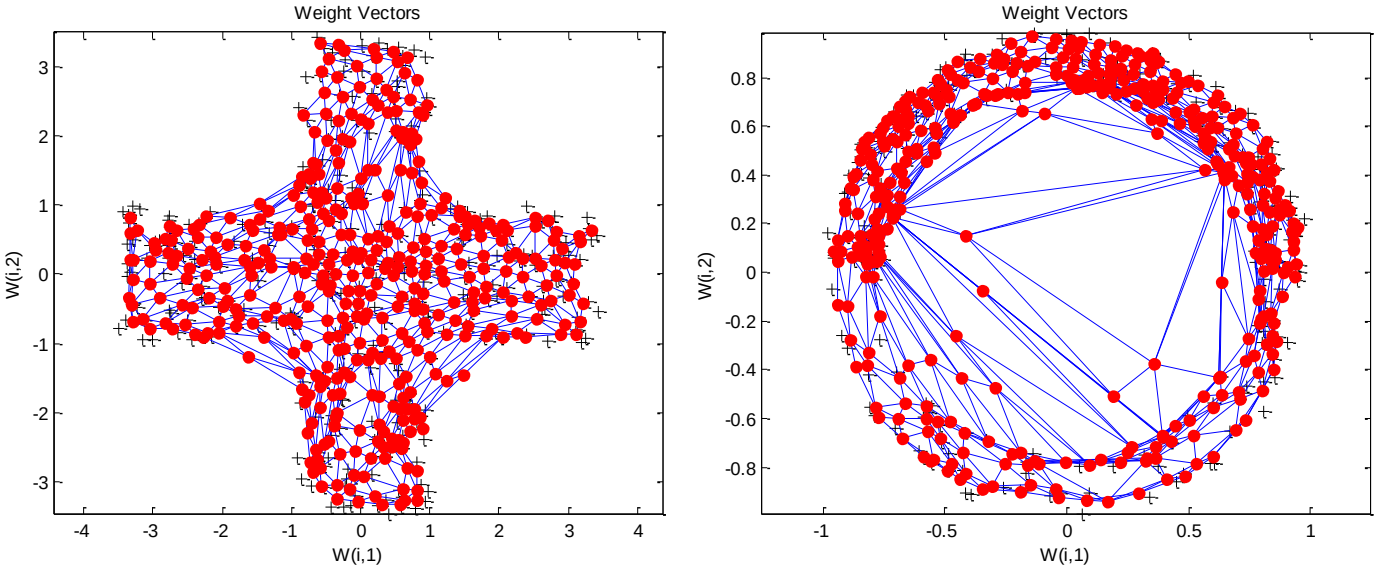
Τελικά μετά απο αρκετές προσομοιώσεις καταλήξαμε σε επιλογή grid με μέγεθος 20x20 και περίπου 500 βήματα orderSteps με τις υπόλοιπες παραμέτρους να έχουν τις default τιμές όπως αναφέρονται στην εκφώνηση της άσκησης για την περίπτωση που χρησιμοποιήθηκαν τα crossData ενώ grid μεγέθους 15x15 στην περίπτωση που χρησιμοποιήθηκαν ringData με τις υπόλοιπες τιμές ίδιες. Τα αποτελέσματα των προσομοιώσεων αυτών φαίνονται στο επόμενο σχήμα.

Να αναφέρουμε εδώ οτι για τις συναρτήσεις som που υλοποιήθηκαν υπάρχουν ακριβή σχόλια στον παραδωταίο κώδικα και γι'αυτό δεν αναλύεται η υλοποίησή τους επιπλέον στην παρούσα αναφορά.



Σχήμα 4 : αριστερά: crossData, δεξιά: ringData με 20x20 grid και 500 βήματα

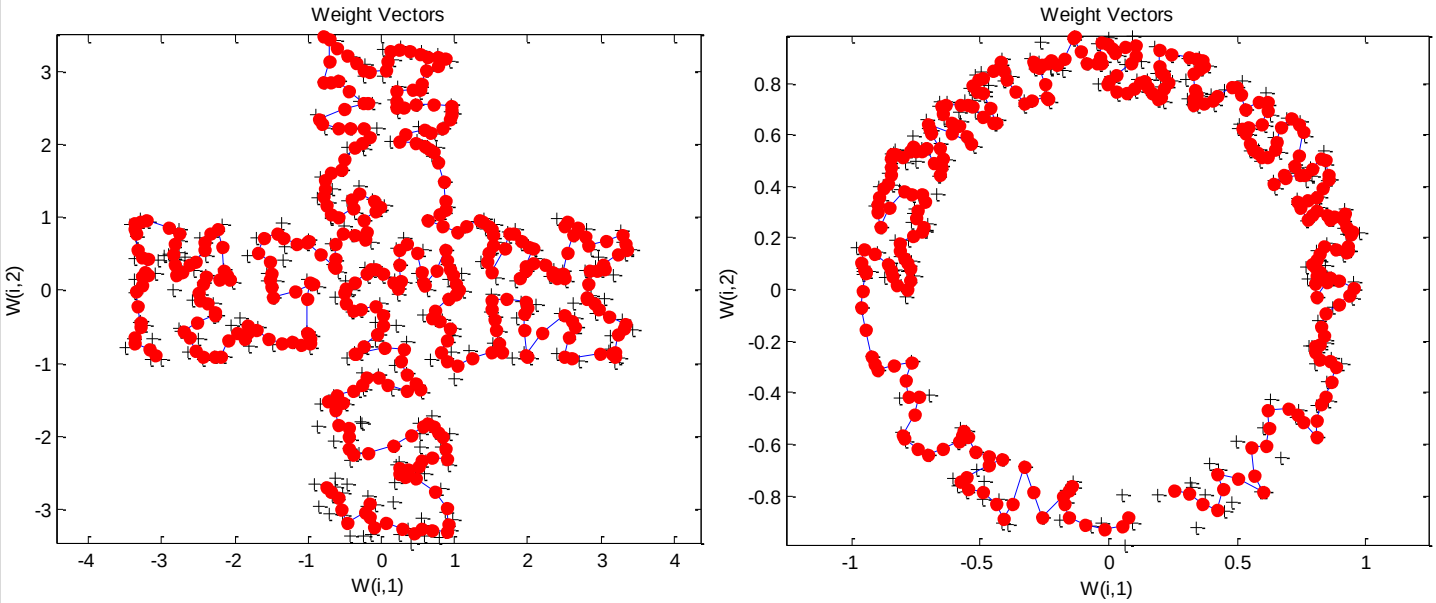
Όλα τα παραπάνω έγιναν για πλέγματα gridtop και μέτρο απόστασης boxdist. Στα παρακάτω σχήματα ακολουθούν προσομοιώσεις για τύπους πλέγματος με εξαγωνική τοπολογία.



Σχήμα 5 : απεικονίσης μετά απο εκπαίδευση πλέγματος εξαγωνικής τοπολογίας

Παρατηρούμε οτι δεν υπάρχουν μεγάλες διαφορές χρησιμοποιώντας διαφορετική τοπολογία. Τουλάχιστον όχι όταν ο αριθμός των βημάτων είναι μικρός καθώς για μικρότερο αριθμό βημάτων οι διαφορές ήταν εμφανείς.

Ένα επόμενο βήμα που θα κάνουμε για να εκτιμήσουμε την απόδοση του som στην περίπτωση που χρησιμοποιήσουμε μονοδιάστατο πλέγμα για την περίπτωση των ringData προτύπων. Αναμένουμε πως οι νευρώνες εφόσον δεν θα περιορίζονται τόσο πολύ απο τις τιμές των γειτονικών τους θα μπορούν να ανανεώσουν τα βάρη τους καλύτερα και να μλην υπάρχει τόσο έντονα το φαινόμενο να υπάρχουν νευρώνες που αντιστοιχούν στο εσωτερικό του δακτυλίου. Τα αποτελέσματα για δύο διαφορετικού πλήθους νευρώνων grid εμφανίζονται στο επόμενο σχήμα.



Σχήμα 6 : απεικονίση μετά απο εκπαίδευση με πλέγμα 350x1 (αριστερά)των ringData και 100x1 και 280x1 (δεξια) των crossData

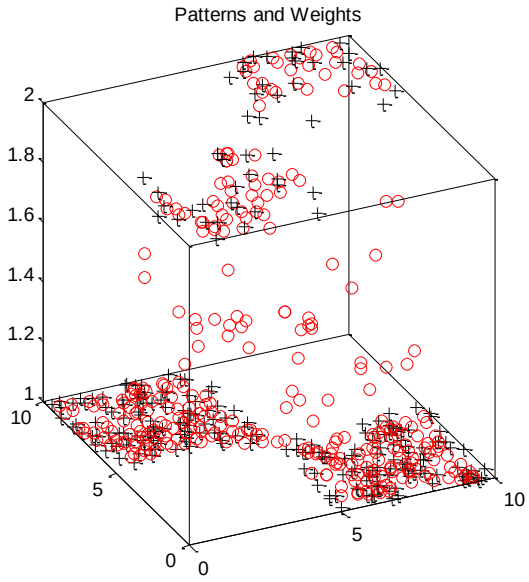
Παρατηρούμε λοιπόν απο το παραπάνω σχήμα οτι πραγματικά δεν υπάρχει το πρόβλημα αντιστοίχησης νευρώνων στον ενδιάμεσο χώρο του δακτυλίου ούτε και το πρόβλημα εμφάνησης νευρώνων στις εσωτερικές γωνίες του σταυρού. Επίσης η κατανομή των νευρώνων ακολουθεί την κατανομή των προτύπων. Μάλιστα σ'αυτήν την μπερίπτωση που χρησιμοποιούμε το ίδιο πλήθος νευρώνων με προτύπων βλέπουμε οτι σχεδόν ο κάθε νευρώνας αντιστοιχίζεται σε ένα πρότυπο.

Να αναφέρουμε εδώ οτι και στις δύο κατηγορίες προτύπων χρησιμοποιήθηκαν απο μικρό πλήθος έως μεγάλο πλήθος νευρώνων. Σε προβλήματα κατηγοριοποίοησης βέβαια η λογική είναι να κάνουμε μια κατάτμηση του χώρου και συνεπώς για να μειωθεί η πολυπλοκότητα του προβλήματος χτησιμοποιούμε αισθητά λιγότερους σε πλήθος νευρώνες απ'ότι πρότυπα. Στο συγκεκριμένο μέρος της άσκησης δεν έγινε κάτι τέτοιο καθώς ο σκοπός ήταν να μελετηθεί γενικότερα η συμπεριφορά του αυτο-οργανούμενο χάρτη κατα την εκπαίδευση επιβεβαιώνοντας την ικανότητά του να βρίσκει τη χωρική κατανομή αλλα και την κατανομή πυκνότητας των προτύπων στο χώρο.

Η εξαγωγή των βέλτιστων παραμέτρων συσχετίζεται άμεσα με την επιθυμία μας να μειώσουμε την πολυπλοκότητα του χώρου ή να περιγράψουμε την κατανομή των προτύπων με τον βέλτιστο δυνατό τρόπο καθώς επίσης σημαντικό λόγο έχει και ο χρόνος που έχουμε στη διάθεσή μας για την εκπαίδευση των χαρτών.

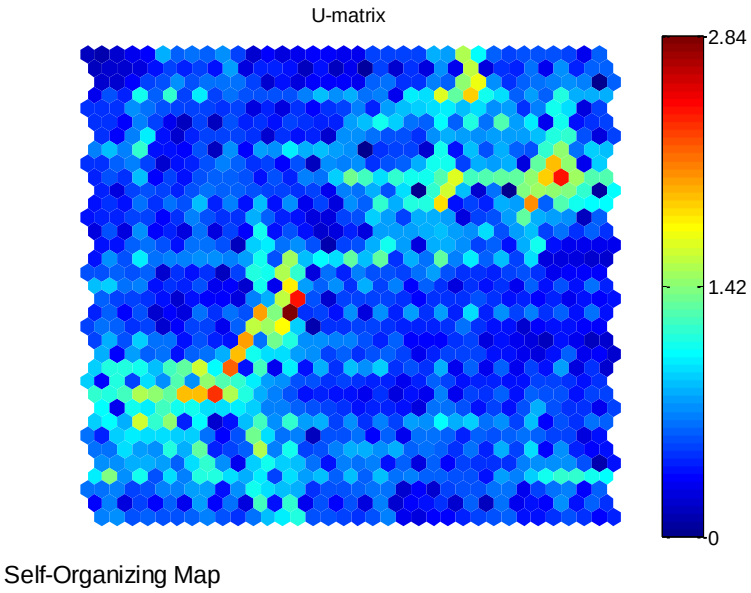
2B. Στο επόμενο μέρος αρχικά επαναλήφθηκε η παραπάνω διαδικασία με πρότυπα τα εξαγόμενα απο το groupData.m. Φυσικά πλέον τα διανύσματα των προτύπων έχουν τρείς διαστάσεις και η απεικόνιση όπως προηγουμένως δεν έχει κάποιο νόημα. Όμως μπορούμε εναλλακτικά να χρησιμοποιήσουμε τη συνάρτηση απεικόνισης **plot3** του MatLab για να κάνουμε μια μικρή αντιστοίχηση με το προηγούμενο μέρος πρίν γίνει χρήση του U-Matrix. Έτσι αντίστοιχα με πρίν τα βάρη των νευρώνων μετά την εκπαίδευση αναμένουμε να έχουν άμεση σχέση με τις συντεταγμένες των παραμέτρων στον χώρο. Φυσικά η τρίτη συντεταγμένη των προτύπων θα είναι πάντα 1 ή 2 , εφόσον θα έχει την τιμή της κλάσης στην οποία ανήκει το συγκεκριμένο πρότυπο. Στο παρακάτω σχήμα απεικονίζουμε τα πρότυπα στον χώρο και την αντιστοίχηση των βαρών των νευρώνων στον

χώρο μετά απο εκπαίδευση som πλέγματος gridtop 20x20. Για λόγους απλότητας δεν απεικονίζονται οι ενώσεις των νευρώνων.



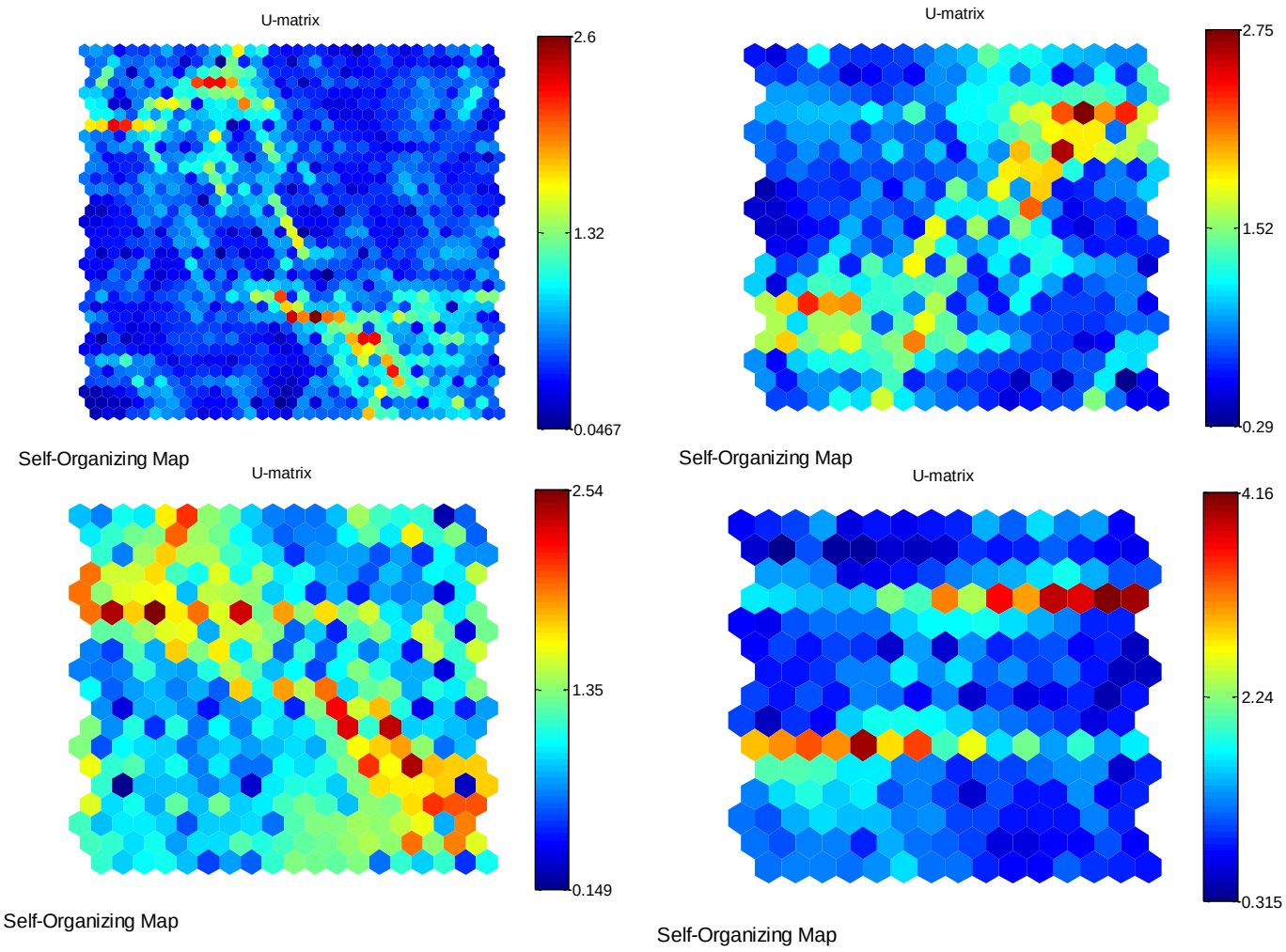
Σχήμα 7 : Απεικόνιση των προτύπων και των βαρών των νευρώνων στον χώρο (18x18 grid και 50 βήματα)

Απο το παραπάνω σχήμα παρατηρούμε οτι πράγματι και σε αυτή την περίπτωση τα βάρη των νευρώνων έχουν τιμές αντίστοιχες των "συντεταγμένων" των προτύπων. Μερικόι νευρώνες και πάλι όπως φαίνεται δεν έχουν αντιστοιχηθεί σε κάποια κλάση και αναμένεται στον U-Matrix να φαίνεται με κάποιο τρόπο οτι οι αποστάσεις των βαρών τους απο των γειτονικών τους είναι σχετικά μεγάλες (αν και εδω δεν φαίνεται ποιοί είναι οι γείτονες κάθε νευρώνα). Στο παρακάτω σχήμα φαίνεται και ο U-Matrix της ίδιας προσομοίωσης.



Σχήμα 8 : U-Matrix πλέγματος gridtop 18x18 και 50 βήματα

Βλέπουμε λοιπόν και στον U-matrix οτι κάποιοι νευρώνες απέχουν πολυ απο τους γειτονικούς τους (τα βάρη τους) και άλλοι είναι πολύ κοντά. Αυτό επιβεβαιώνεται και απο το σχήμα 7 και είναι το αναμενόμενο αποτέλεσμα. Στη συνέχεια παρουσιάζουμε και άλλες προσομοιώσεις με διαφορετικό μέγεθος grid, διαφορετική τοπολογία grid, διαφορετική τοπολογία γειτονιάς.



Σχήμα 9 : a:18x18-hexagonal-boxdist, b:10x10-hexagonal-boxdist, c:10x10-hexagonal-mandist, d:8x8-hexagonal-dist

Παρατηρώντας απο σχήμα 7 τη δομή στο χώρο των προτύπων μπορούμε να δούμε οτι απο τα παραπάνω αποτελέσματα έχουμε καλύτερες επιδόσεις όταν χρησιμοποιούμε απόσταση boxdist και gridtop πλέγμα. Ωστόσο δεν βρέθηκε κάποια συστηματική διαδικασία αξιολόγησης των αποτελεσμάτων παρα μόνο οπτικά οπου η παρατήρηση όλων των δυνατών συνδυασμών παραμέτρων με προσομοίωση ήταν εξαιρετικά χρονοβόρος διαδικασία. Απο μεμονωμένα πειράματα επιλέχθηκε τελικά πλέγμα 10x10 hexagonaltopology και αποσταση boxdist. Ο U-matrix αυτού του μοντέλου φαίνεται στο σχήμα 9.

Επόμενο βήμα είναι να απαντήσουμε στο ερώτημα αν υπάρχει συσχέτιση μεταξύ του πλήθους των προτύπων κάθε ομάδας και του συνόλου των νευρώνων που ανατίθεται στην εν λόγω ομάδα. Ο έλεγχος για μια τέτοια σχέση θα μπορούσε να γίνει με διάφορους τρόπους. Εδώ το σκεπτικό που ακολουθούμε έιναι οτι ο λόγος μεταξύ των κλάσεων προς των λόγο μεταξύ των νευρώνων που ανατίθενται σε κάθε κλάση θα πρέπει να είναι μονάδα. Άν λοιπόν ονομάσουμε ώς group1 το πλήθος των προτύπων που ανήκουν στην κλάση 1 και group2 τον αριθμό των προτύπων που ανήκουν στην κλάση 2 τότε μπορούμε να εξάγουμε το πηλίκο group1/group2. Να αναφέρουμε εδώ οτι μπορούμε να βρούμε τους αριθμούς ελέγχοντας την τελευταία γραμμή του πίνακα PATTERNS ο οποίος για τα GroupData είναι 3x250. Συγκεκριμένα η εντολή `group1 = sum(PATTERNS(3 , :)==1)` θα μας δώσει το πλήθος των προτύπων της κλάσης 1. Στη συνέχεια αν θεωρήσουμε ως weights1 το πλήθος των νευρώνων οι οποίοι ανατίθενται στην κλάση 1 και weights2 στην κλάση 2 μπορούμε να έχουμε το πηλίκο weights1/weights2. Συγκεκριμένα η εντολή `weights1 = sum(IW(:,3)<1.3)` θα μας δώσει το πλήθος των νευρώνων που ανατίθενται στην κλάση 1 θεωρώντας οτι μια τέτοια ανάθεση ΄γινεται

όταν η τρίτη συνιστώσα του βάρους του έχει τιμή μικρότερη απο 1.3. Έτσι μπορούμε πλέον να έχουμε το συνολικό πηλίκο (weights1/weights2)/(group1/group2) το οποίο στην περίπτωση που ο κανόνας ισ΄χυει θα πρέπει να λαμβάνει την τιμή "1". Άν η τιμή του πηλίκου αυτού είναι μεγαλύτερη απο τη μονάδα τότε σημαίνει οτι οι νευρώνες δείχνουν μια επιλεκτικότητα στην 1η κλάση ενώ αν η τιμή είναι μικρότερη της μονάδας έχουν επιλεκτικότητα στη δεύτερη. Να αναφέρουμε εδώ οτι μικρές αποκλίσεις θα υπάρχουν αναπόφευκτα. Όμως βασικό πρόβλημα είναι το γεγονός οτι ένω η επιλεκτικότητα ώς προς την 1η κλάση δεν έχει άνω όριο (μπορεί ακόμα και να απειριστεί στην περίπτωση που κανένας νευρώνας δεν αντιστοιχηθεί στην 2η κλάση) δεν συμβαίνει το ίδιο για την δεύτερη κλάση οπου η τιμή του πηλίκου είναι φραγμένη απο το μηδέν (γενικά θεωρούμε οτι στα πρότυπα υπάρχουν και οι δύο κλάσεις).

Ένας τρόπος να απο φύγουμε απ'αυτή τη μή γραμμική σχέση που δυσκολέυει την παρατήρηση και εξαγωγή συμπερασμάτων είναι να χρησιμοποιήσουμε τη λογαριθμική συνάρτηση. Συγκεκριμένα η τιμή bias=log2((weights1/weights2)/(group1/group2)) θα είναι μηδενική όταν δεν υπάρχει επιλεκτικότητα, θετική όταν η επιλεκτικότητα αφορά την πρώτη κλάση και αρνητική όταν αφορά τη δεύτερη κλάση. Πλέον υπάρχει μια σχετική αναλογία γύρω απο το μηδέν οπότε μπορούμε πιο εύκολα να αποφανθούμε για το μέγεθος της επιλεκτικότητας. Επιπλέον θα λαμβάνει την τιμή -Inf όταν όλοι οι νευρώνες έχουν αντιστοιχηθεί στη δεύτερη κλάση και +Inf όταν έχουν αντιστοιχηθεί στην πρώτη. Θα θεωρούμε οτι όταν η τιμή αυτή δεν ξεπερνάει κατα απόλυτο τιμή το 0.25 (καθώς log2(1.2) = 0.26) οι αναλογίες διατηρούνται ενώ όταν φτάσει να έχει απόλυτο τιμή μονάδα τότε η επιλεκτικότητα είναι διπλάσια ώς προς κάποια κλάση (εφόσον log2(2)=1 και log2(1/2)=-1).

Στην περίπτωση που χρησιμοποιήσαμε τις παραμέτρους που αναφέρθηκαν παραπάνω λάβαμε την τιμή bias=-0.1186 που δείχνει μια σχετικά ασήμαντη επιλεκτικότητα στη δεύτερη κλάση και συγκεκριμένα 2^0.1186=1.08 φορές. Στη συνέχεια θα επαναλάβουμε τη διαδικασία για όλους τους τύπους πλέγματος και τις διαφορετικές τοπολογικές γειτονιές. Για να γίνουν οι προσομοιώσεις αυτές μειώνουμε τα βήματα στα 250 τα οποία είναι σχετικά ικανοποιητικός αριθμός. Στον παρακάτω πίνακα φαίνονται τα αποτελέσματα για πλέγμα 10x10.

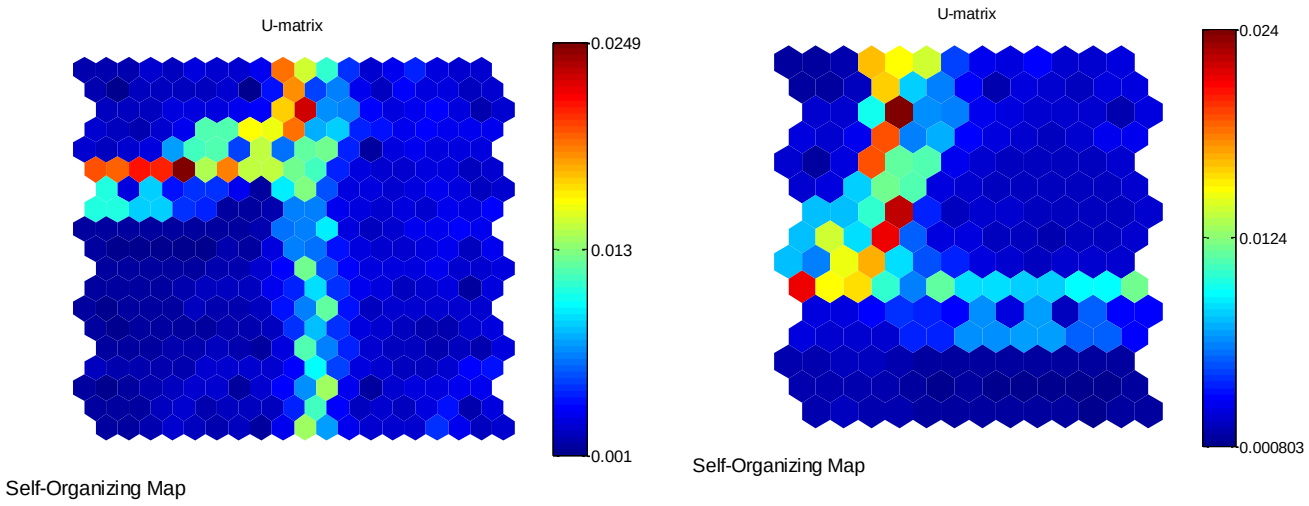
	gridtop	hextop	hexagonaltopology
boxdist	0.1497	-0.2134	-0.2134
dist	-0.1722	-0.2426	-0.1722
linkdist	-0.2426	-0.0982	-0.0982
mandist	-0.2134	-0.4150	-0.3720

Παρατηρούμε λοιπόν οτι εκτός απο δύο περιπτώσεις όλες τις υπόλοιπες φορές δεν υπάρχει μεγάλη επιλεκτικότητα ως προς μία κλάση. Γενικότερα υπάρχει μια μικρή μεροληψία προς τη δεύτερη κλάση προτύπων αλλα αυτή δεν ξεπερνάει τα όρια που θέσαμε. Έτσι επαληθέυεται το γεγονός οτι η υπάρχει άμεση συσχέτιση μεταξύ της πυκνότητας των προτύπων και και την πυκνότητα των νευρώνων που πεεριγράφουν την τοπολογία και τις ιδιότητες αυτής της ομάδας.

Όλα τα παραπάνω υλοποιούνται μέσω του m-file **askisi_2B.m**

3Α. Στο πρώτο μέρος απλώς εφαρμόστηκε η συνάρτηση **tdidf.m** ωστε να εξαχθεί ο πίνακας των βαρών ο οποίος στη συνέχεια θα αποτελέσει τα πρότυπα πάνω στα οποία θα εκαπαιδευτεί ο αυτο-οργανούμενος χάρτης. Βέβαια θέλει προσοχή καθώς ο εξαγώμενος πίνακας θα έχει τα έγγραφα ώς στήλες του πίνακα και θα χρειαστεί να πάρουμε τον ανάστροφο πίνακα για εφαρμογή στην somtrain.

3Β. Στη συνέχεια έγινε εκπαίδευση πάνω σε αυτά τα πρότυπα χρησιμοποιώντας διδιάστατο πλέγμα 10x10 και 50 βήματα λόγω της εξαιρετικά χρονοβόρου εκπαίδευσης αλλα και πλέγματος 7x8 με 250 βήματα. Τα αποτελέσματα των U-Matrix φαίνονται στο παρακάτω σχήμα. Επίσης επιλέχθηκε τύπος πλέγματος hexagonalTopology και τύπος απόστασης boxdist.



Σχήμα 10 : U-matrix 10x10 πλέγματος (αριστερά) και 7x8 πλέγματος (δεξιά)

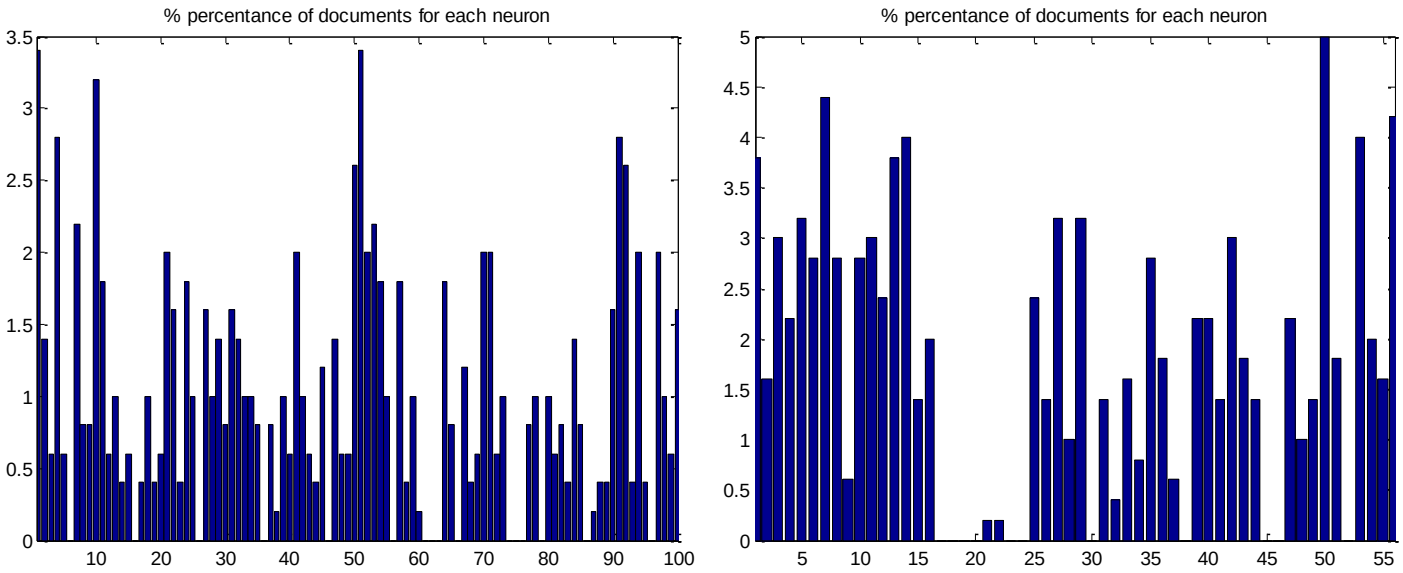
Απο τα παραπάνω σχήματα διαπιστώνουμε οτι θα μπορούσαμε να διαχωρίσουμε τα έγγραφα σε τρείς διαφορετικές ομάδες. Μάλιστα λαμβάνοντας υπόψιν το συμπέρασμα που εξήχθηκε στο δεύτερο μέρος της άσκησης μπορούμε να πούμε οτι η μία ομάδα έχει περισσότερα έγγραφα, η επόμενη λίγο λιγότερα και η τρίτη εμφανώς λιγότερα. Παρατηρώντας τω U-matrix του 7x8 πλέγματος οπου τα αποτελέσματα είναι πιο αξιόπιστα λόγω του πλήθους των επαναλήψεων μπορούμε να παρατηρήσουμε οι πάνω δεξιά νευρώνες έχουν αντιστοιχηθεί στην πρώτη ομάδα, οι πάνω αριστερά στη δεύτερη και οι κατώτεροι νευρώνες του πλέγματος στην τρίτη ομάδα εγγράφων.

3Γ.

(i) Για να μπορέσουμε να βρούμε το ποσσοστό των εγγράφων που αντιστοιχεί σε κάθε νευρώνα θα έχουμε έναν πίνακα winners στον οποίο θα προσθέτουμε τον πίνακα εξόδου της συνάρτησης somOutput για κάθε πρότυπο. Έτσι για κάθε πρότυπο θα έχουμε έναν πίνακα στήλη με μονάδα στον νευρώνα νικητή και μηδενικά αλλού. Στο τέλος αθροίζοντας όλους αυτούς θα βρούμε πόσσες φορές , δηλαδή για πόσα πρότυπα, ο κάθε νευρώνας ήταν νικητής. Τελικά διαιρώντας τον πίνακα winners με το πλήθος των εγγράφων και πολλαπλασιάζοντας με 100 θα έχουμε σε κάθε στοιχείο του πίνακα το επι τοις εκατό ποσσοστό των εγγράφων που αντιστοιχεί σε κάθε νευρώνα.

```
winners = zeros(n*m,1);
for i=1:size(PATTERNS,2)
    wins = somOutput(PATTERNS(:,i));
    winners = winners + wins;
end
wp = (winners/size(PATTERNS,2))*100;
```

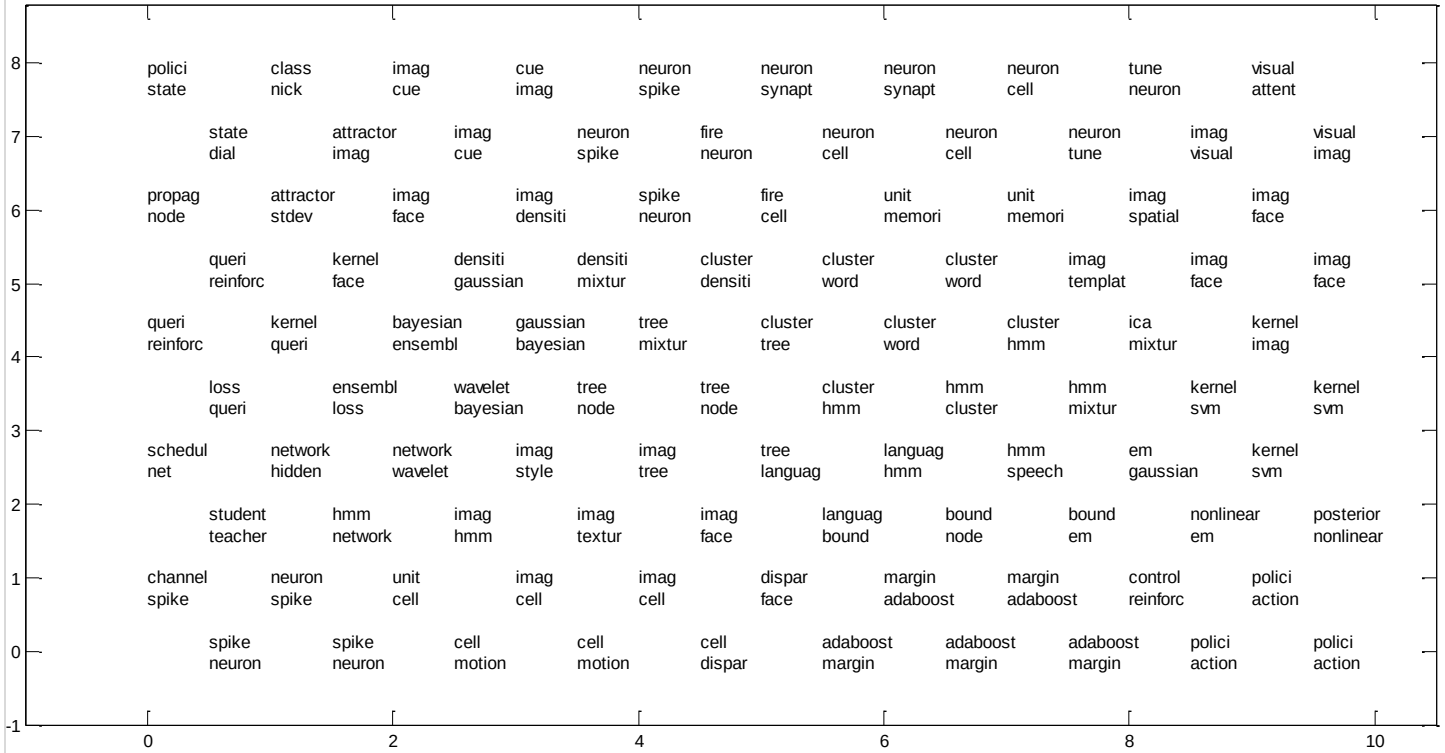
Το σποτέλεσμα όπως φαίνεται και στο ενδεικτικό τμήμα κώδικα βρίσκεται πλέον στον πίνακα wp όπότε με χρήση της συνάρτησης bar μπορούμε να το απεικονίσουμε.



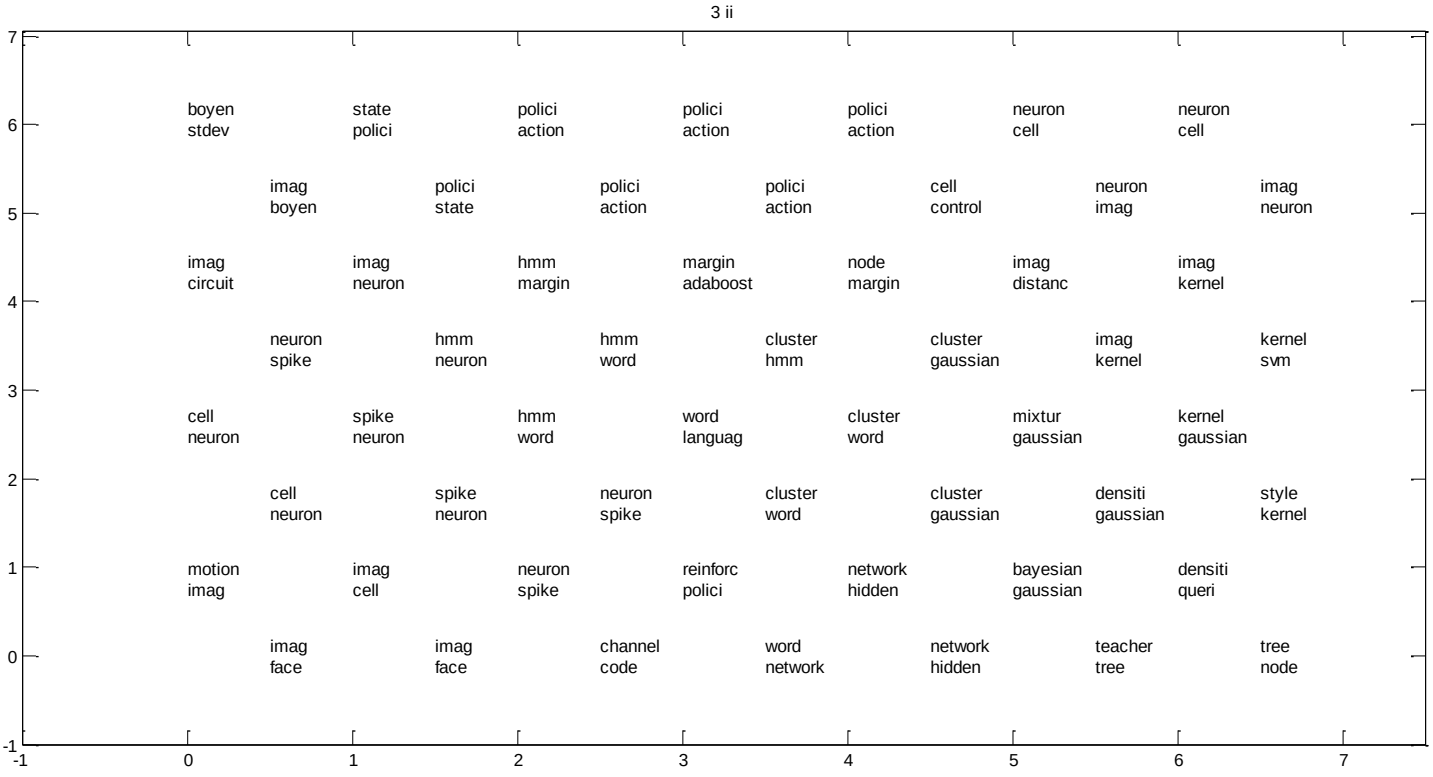
Σχήμα 11 : ποσσοστό των εγγράφων που αντιστοιχεί σε κάθε νευρώνα για 10x10 grid και 50 εποχές (αριστερά) και 7x8 grid και 250 εποχές (δεξιά)

Παρατηρούμε λοιπόν οτι υπάρχουν νευρώνες στους οποίους αντιστοιχίζεται αρκετά υψηλό ποσσοστό των εγγράφων, δηλαδή ήταν νικητές για μεγάλο πλήθος εγγράφων, νευρώνες που έχουν ένα μέσο ποσσοστό και νευρώνες που έχουν ένα πολύ μικρό έως και μηδενικό ποσσοστό. Αυτή η τελευταία κατηγορία νευρώνων είναι ουσιαστικά αυτοί που αντιστοιχίζονται στα όρια αλλαγής των τριών κατηγοριών των εγγράφων, δηλάδή αυτοί οι οποίοι στονU-Matrix αντιστοιχούν σε πίο κόκκινες ο περιοχές και δεν αντιστοιχούν γενικότερα σε καμία κατηγορία εγγράφων.

(ii). Στη συνέχεια για να απεικονίσουμε τους 2 όρους για κάθε νευρώνα που έχουν το μεγαλύτερο βάρος θα πρέπει για κάθε νευρώνα να βρούμε δυο δείκτες, ο ένας θα αντιστοιχεί στο μεγιστο και ο δεύτερος στο δεύτερο μέγιστο στοιχείο του πίνακα. Υπάρχουν διάφοροι τρόποι να κάνουμε αυτόν τον υπολογισμό. Ο ένας είναι να χρησιμοποιήσουμε τις εντολές findpeaks και find. Μπορούμε επίσης να χρησιμοποιήσουμε μόνο την εντολή find για να βρούμε τον δείκτη στον οποίο βρίσκεται το μέγιστο στοιχείο του διανύσματος του κάθε νευρώνα, στη συνέχεια να θέσουμε την τιμή αυτή -inf και να ξανακάνουμε το ίδιο. Αναλυτικά η διδικασία αυτή φαίνεται στον κώδικα της άσκησης. Έπειτα χρησιμοποιήθηκε η συνάρτηση text για να απεικονίσουμε τα αποτελέσματα, οπου για κάθε όρο απεικονίζονται τα δύο στοιχεία που αντιστοιχούν στους δείκτες που εξήχθησαν ώς μέγιστα απο την παραπάνω διαδικασία. Τα αποτελέσματα φάινονται στα σχήματα 12 και 13 για λόγους ευκολίας στην ανάγνωση. Αυτό που παρατηρείται απο αυτά τα δύο σχήματα είναι το γεγονός οτι οι γειτονικοί νευρώνες εμφανίζουν κοινούς όρους (τουλάχιστον ο ένας εκ των δύο είναι κοινός) κάτι που είναι αρκετά λογικό και επίσης υπάρχουν περισσότερες ομοιότητες ανα ομάδα απο οτι για νευρώνες διαφορετικών ομάδων.



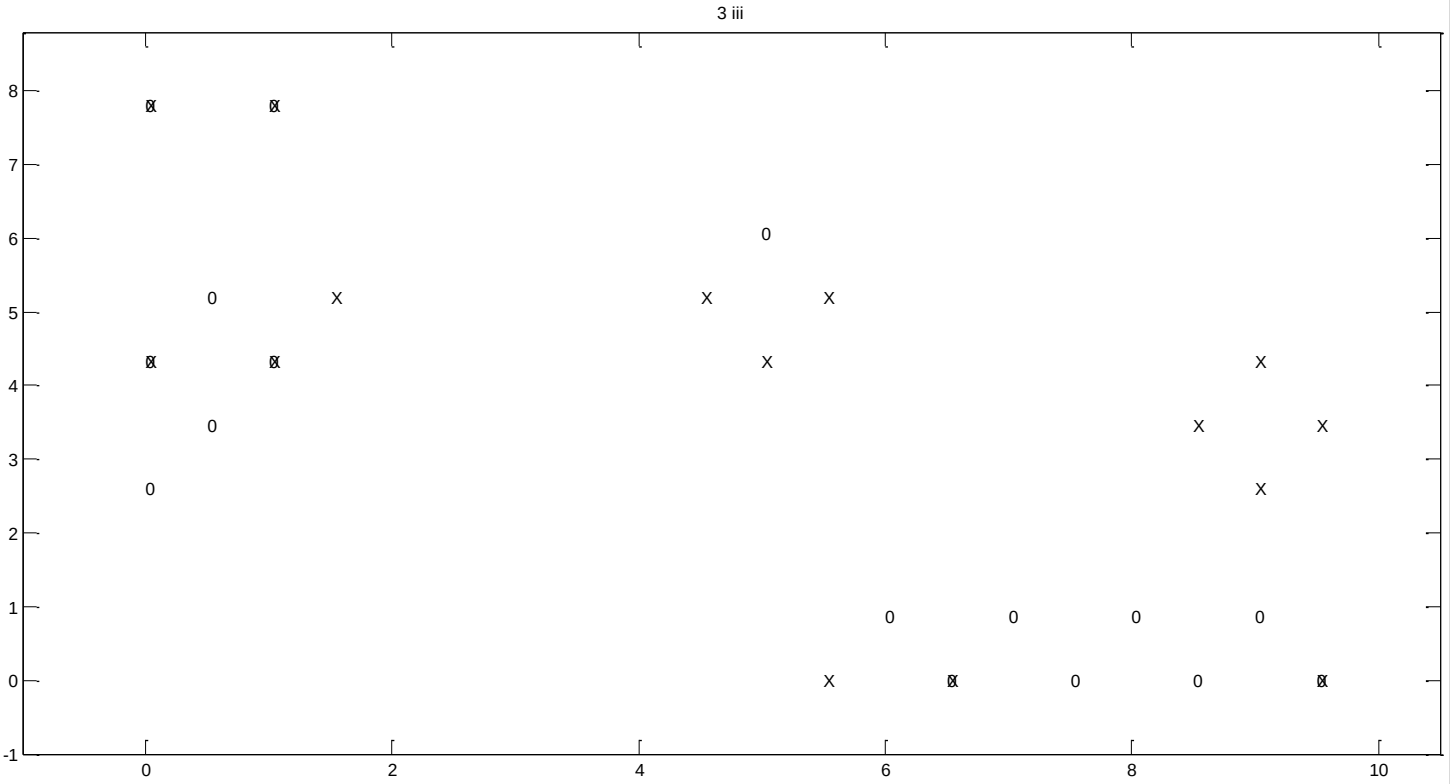
Σχήμα 12 : απεικόνιση 2 σημαντικότερων όρων για κάθε νευρώνα (grid 10x10)



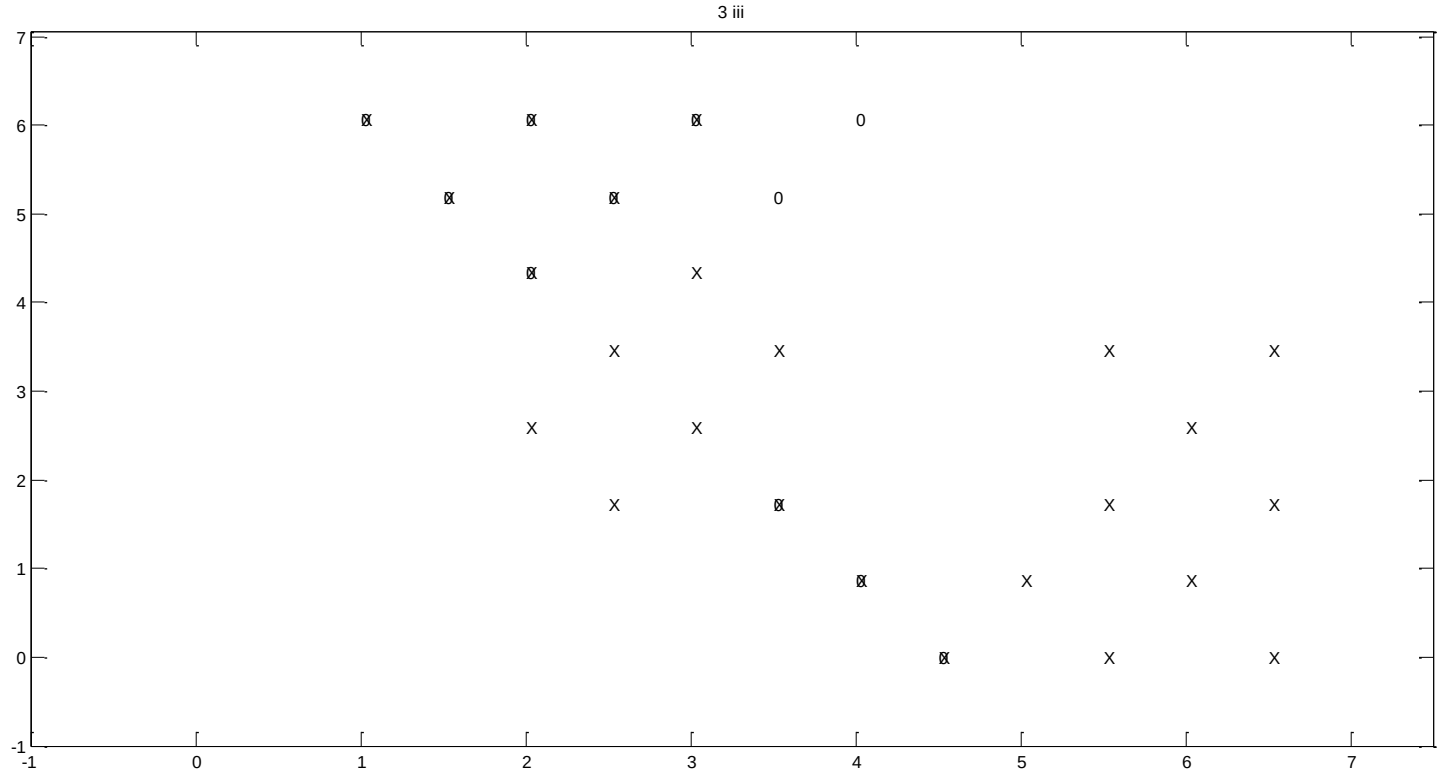
Σχήμα 13 : απεικόνιση 2 σημαντικότερων όρων για κάθε νευρώνα (grid 7x8)

(iii). Σ'αυτό το μέρος αφού υπολογίσουμε τους δείκτες γραμμών για τις οποίες ο πίνακας terms λαμβάνει τις μεταβλητές χαρακτήρων 'machin' και 'learn' ,οι οποίες βρέθηκε οτι αντιστοιχούν στον 135ο και τον 2ο όρο αντίστοιχα, υπολογίζουμε και τη μέγιστη τιμή βάρους κάθε αντίστοιχης στήλης του πίνακα IW (αφού οι όροι αντιστοιχίζονται στις στήλες του IW) , βρίσκοντας ουσιαστικά τα μέγιστα βάρη που αντιστοιχούν σ΄αυτούς τους δύο όρους. Έπειτα κρατάμε μόνο τις γραμμές του IW εκείνες για τις οποίες το βάρος της 135ης στήλης (για τον όρο 'machin') είναι μεγαλύτερο απο το μισό του μέγιστου που υπολογίστηκε και αντίστοιχα κρατάμε και τις γραμμές για τις οποίες η 2η

στήλη έχει τιμή μεγαλύτερη απο το μισό του μεγίστου της 2ης στήλης όλου του πίνακα IW. Προφανώς κάποιες γραμμές μπορεί να είναι κοινές. Έπειτα απεικονίζουμε αυτούς τους νευρώνες στη θέση του πλέγματος. Τα αποτελέσματα φαίνονται στα παρακάτω σχήματα, οπου οι όροι που αντιστοιχούν στο "machin" φαίνονται με '0' και αυτοί που αντιστοιχούν στο "learn" φάινονται με 'X'. Εφόσον δεν είναι σαφές απο την ερώτηση να πούμε οτι οι όροι για τους οποίους ισχύουν και τα δύο (πιθανότατα έγγραφα που αφορούν το πεδίο machine learning) παρουσιάζουν και τα δύο σύμβολα.

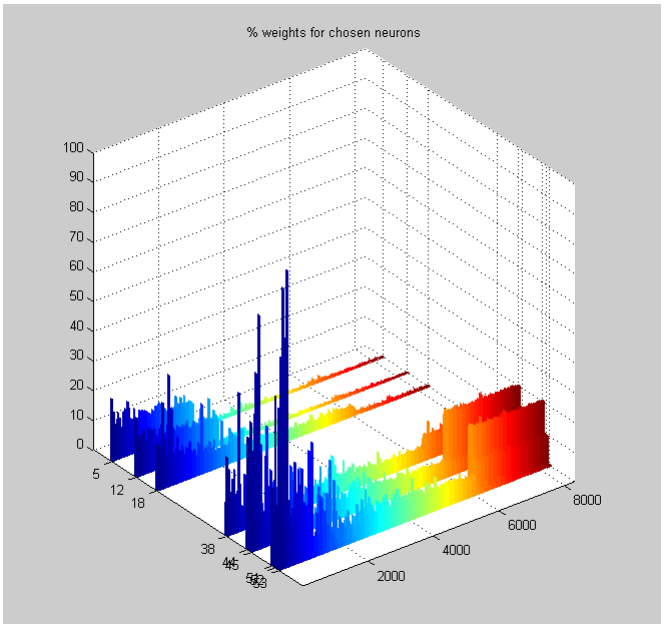


Σχήμα 14 : απεικόνιση με "0" των νευρώνων των οποίων τα βάρη των όρων machin έιναι πάνω απο το 50% του μέγιστου βάρους και με "X" για τον όρο learn (10x10 grid)



Σχήμα 15 : απεικόνιση με "0" των νευρώνων των οποίων τα βάρη των όρων machin έιναι πάνω απο το 50% του μέγιστου βάρους και με "X" για τον όρο learn (7x8 grid)

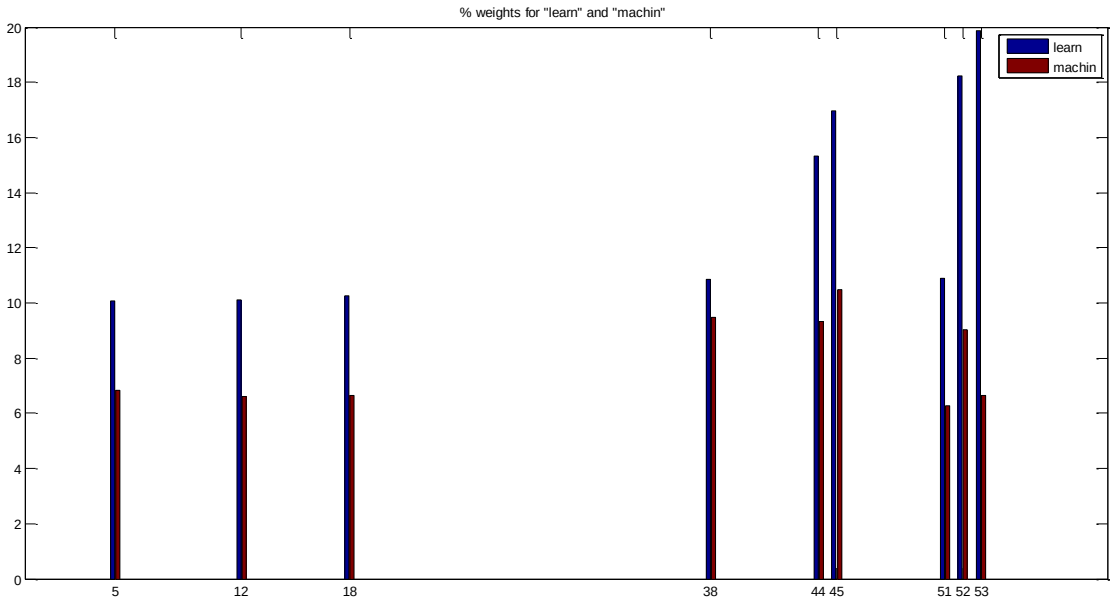
(iv). Γι'αυτό το ερώτημα θεωρούμε απο προηγουμένως οτι οι ζητούμενοι νευρώνες είναι αυτοί που παρουσιάζουν βάρη μεγαλύτερα του 50% της μέγιστης τιμής και των δύο όρων 'machin' , 'learn'. Στο παρακάτω σχήμα φαίνονται τα επι τοις εκατό ποσοστά των βαρών κάθε όρου των επιλεγμένων αυτών νευρώνων ώς προς τη μέγιστη τιμή βάρους όλων των νευρώνων του χάρτη.



Σχήμα 16 : απεικόνιση των % τιμών όλων των βαρών των επιλεγμένων νευρώνων (7x8 grid)

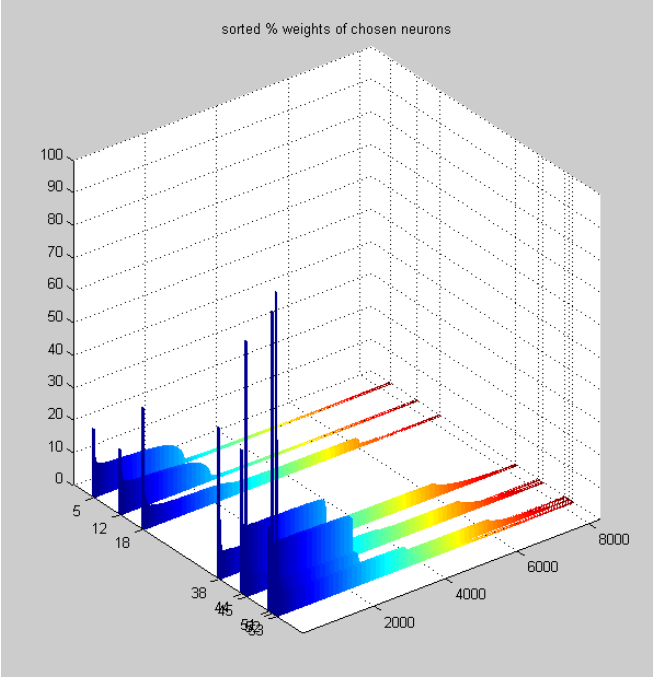
Απο το παραπάνω σχήμα παρατηρούμε όλα τα ποσοστιαία βάρη των επιλεγμένων νευρώνων, οι οποίοι είναι 9 το πλήθος όπως φαίνεται και απο το σχήμα 15. Μια παρατήρηση που μπορούμε να κάνουμε απο το παραπάνω σχήμα είναι οτι στους τελευταίους νευρώνες τα βάρη των όρων απο 6000 και μετά είναι αρκετά υψηλά ενώ στους πρώτους νευρώνες είναι σχεδόν μηδενικά. Αυτό μας κάνει να υποθέτουμε οτι οι νευρώνες αυτοί αντιστοιχίζονται σε διαφορετική κατηγορία κάτι που επιβεβαιώνεται και απο τον U-matrix του σχήματος 10 (δεξιά). Επίσης φαίνεται και το φαινόμενο οτι οι γειτονικοί νευρώνες έχουν παρόμοιες τιμές βαρών.

Στη συνέχεια θα απεικονίσουμε μόνο τις ενδιαφερόμενες ποσοστιαίες τιμές βαρών δηλαδή τη 2η και την 135η στήλη των επιλεγμένων νευρώνων.



Σχήμα 17 : απεικόνιση των % τιμών των ενδιαφερόμενων βαρών των επιλεγμένων νευρώνων (7x8 grid)

(v). Στο τέλος καλούμαστε να τα ταξινομήσουμε τα βάρη των επιλεγμένων νευρώνων απο το μεγαλύτερο προς το μικρότερο κρατώντας ουσιαστικά τις αλλαγές αυτές των θέσεων και εφαρμόζοντάς τις και στον πίνακα terms. Στην ουσία παρατηρώντας το σχήμα 16 τότε για κάθε έναν απο τους 9 νευρώνες θα ταξινομήσουμε τα βάρη τους (προφανώς δεν θα πειράξουμε τον ίδιο τον πίνακα βαρών ούτε τον terms αλλα αντίγραφα αυτών) με χρήση της συνάρτησης sort ηοποία μας δίνει τη δυνατότητα να γνωρίζουμε και τους δείκτες που αντιστοιχούν στη σειρά της ταξινόμησης. Στο σχήμα 18 φαίνονται τα ταξινομημένα βάρη.



Σχήμα 18 : απεικόνιση των % τιμών όλων των βαρών των επιλεγμένων νευρώνων (7x8 grid) σε φθίνουσα σειρά

Έπειτα το μόνο που χρειάζεται είναι να εφαρμόσουμε τον ίδιο μετασχηματισμό στον πίνακα όρων terms. Αυτό είναι πλέον εύκολο εφόσον έχουμε κρατήσει τους δείκτες κατα την ταξινόμηση αυτή. Αναλυτικότερα η διαδικασία φαίνεται εντός του παραδωταίου κώδικα οπου τα αποτελέσματα αποθηκεύονται σε ένα Nx1 cell με όνομα sorted_terms οπου N ο αριθμός των επιλεγμένων νευρώνων και κάθε στοιχείο είναι ένα 8296x1 cell που έχει ταξινομημένους τους όρους του αντίστοιχου νευρώνα. Εφόσον είναι αδύνατο να απεικονίσουμε όλους τους όρους αυτούς , θα απεικονίσουμε παρακάτω τους πρώτους 15 όρους για κάθε έναν απο τους 9 νευρώνες του παραπάνω πειράματος.

1	2	3	4	5	6	7	8	9
'network'	'network'	'cluster'	'hmm'	'polici'	'polici'	'state'	'polici'	'polici'
'hidden'	'hidden'	'word'	'margin'	'state'	'action'	'polici'	'action'	'action'
'train'	'chess'	'languag'	'adaboost'	'action'	'state'	'boyen'	'state'	'state'
'ensembl'	'game'	'document'	'state'	'face'	'mdp'	'belief'	'reinforc'	'reinforc'
'unit'	'unit'	'reinforc'	'boost'	'imag'	'reinforc'	'elf'	'mdp'	'mdp'
'net'	'train'	'polici'	'speech'	'rotat'	'reward'	'brows'	'reward'	'reward'
'pca'	'hmm'	'user'	'polici'	'brows'	'margin'	'undiscount'	'pomdp'	'control'
'layer'	'mse'	'state'	'bound'	'undiscount'	'adaboost'	'kanazawa'	'agent'	'agent'
'teacher'	'seri'	'net'	'rotat'	'rota'	'agent'	'stdev'	'environ'	'rl'
'mlp'	'reinforc'	'skill'	'train'	'mcreynold'	'boost'	'init'	'plan'	'pomdp'
'bayesian'	'skill'	'network'	'tree'	'elf'	'bound'	'patent'	'learn'	'environ'
'output'	'expert'	'analog'	'path'	'boyen'	'learn'	'dial'	'execut'	'learn'
'rbf'	'mlp'	'segment'	'theorem'	'lnt'	'algorithm'	'jectori'	'rl'	'plan'
'student'	'interpol'	'anneal'	'face'	'subroutin'	'theorem'	'hank'	'instruct'	'actor'
'hmm'	'net'	'unit'	'classifi'	'nick'	'environ'	'muri'	'markov'	'sutton'

Σχήμα 19 : απεικόνιση των 15 σημαντικότερων όρων των επιλεγμένων νευρώνων