Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Νευρωνικά Δίκτυα Και Ευφυή Υπολογιστικά Συστήματα

Έτος: 2016 - 2017

2η Εργαστηριακή Άσκηση :

«Αυτό-οργανούμενοι Χάρτες (Self Organizing Maps) »

Συνεργάτες:

Βαρηά Χρυσούλα - ΑΜ: 03112105

Τσιβρά Κατερίνα - ΑΜ: 03108208

Τα scripts, τα οποία χρησιμοποιήθηκαν για την εκτέλεση των μετρήσεων και την εκπαίδευση του δικτύου περιέχονται στο zip.

Υλοποίηση SOM:

Ακολουθεί περιγραφή των συναρτήσεων, οι οποίες υλοποιήθηκαν/τροποποιήθηκαν για την εκπαίδευση του δικτύου SOM.

function somCreate(minMax, gridSize):

Η somCreate χρησιμοποιείται για τη δημιουργία του πλέγματος και την αρχικοποίηση των βαρών. Η έτοιμη συνάρτηση τροποποιήθηκε, με σκοπό να μπορεί ο χρήστης να επιλέξει τύπο πλέγματος και μέτρο απόστασης, χωρίς να είναι αναγκαίο να αλλάζει το script σε κάθε εκτέλεση.

function somTrainParameters(setOrderLR, setOrderSteps, setTuneLR):

Χρησιμοποιείται για την αρχικοποίηση των παραμέτρων εκπαίδευσης (ρυθμός μάθησης και αριθμός εποχών). Δεν έγινε καμία τροποποίηση στη δοσμένη συνάρτηση.

function [output] = somOutput(pattern):

Η somOutput λαμβάνει ένα pattern και επιστρέφει τον νευρώνα – νικητή, ο οποίος απέχει λιγότερο από αυτό. Συγκεκριμένα αρχικά υπολογίζεται η αρνητική Ευκλείδεια απόσταση όλων των νευρώνων του δικτύου από το pattern, με χρήση του πίνακα βαρών IW και της συνάρτησης negdist και έπειτα βρίσκεται ο νευρώνας με την καλύτερη απόσταση μέσω της compet.

function [a] = somActivation(pattern, neighborDist):

Η συγκεκριμένη συνάρτηση καθορίζει τη γειτονιά του νευρώνα - νικητή για το pattern εισόδου. Αρχικά προσδιορίζεται η θέση του νευρώνα – νικητή (winner_pos) με τη χρήση της προηγούμενης συνάρτησης. Έπειτα αρχικοποιείται ο πίνακας a[ ], ο οποίος περιέχει 1 στη θέση του νευρώνα νικητή, 0.5 στις θέσεις των νευρώνων, οι οποίοι είναι στο εύρος του νικητή **(distances(winner_pos, :) ≤ neighborDist)** και 0 στις υπόλοιπες θέσεις. Ο πίνακας a[ ] αντιπροσωπεύει τη γειτονιά του νευρώνα – νικητή.

function somUpdate(pattern, learningRate, neighborDist):

Η somUpdate, αρχικά για το pattern εισόδου, βρίσκει το νευρώνα - νικητή και τη γειτονιά του, καλώντας τη συνάρτηση somActivation. Έπειτα ενημερώνει κατάλληλα το βάρος κάθε νευρώνα με τη χρήση του (προηγούμενου) πίνακα a[ ]. Αυτή η διαδικασία γίνεται σε κάθε εποχή για όλα τα patterns.

function somTrain(patterns):

Χρησιμοποιείται για την εκπαίδευση του δικτύου SOM. Η εκπαίδευση χωρίζεται σε δύο στάδια (ordering και tuning), τα οποία υλοποιούνται με χρήση εμφωλευμένων

loops (για όλες τις εποχές και όλα τα patterns εισόδου). Το πρώτο στάδιο διαρκεί orderSteps εποχές και γίνεται ενημέρωση της γειτονιάς κάθε νευρώνα – νικητή. Το μέγεθος της γειτονιάς και ο ρυθμός μάθησης μειώνονται με εκθετικό ρυθμό. Συγκεκριμένα με χρήση της linspace δημιουργούνται δύο πίνακες με orderSteps θέσεις, στους οποίους αρχικοποιούνται εμμέσως οι τιμές των orderLR και neighborDist, οι οποίες χρησιμοποιούνται σε κάθε εποχή. Το δεύτερο στάδιο διαρκεί 2*orderSteps και η ενημέρωση των νευρώνων γίνεται για σταθερό μέγεθος γειτονιάς και ρυθμό μάθησης. Η ενημέρωση των βαρών του πίνακα IW γίνεται με χρήση της συνάρτησης somUpdate.

Μελέτη και Ανάλυση SOM:

2A) Τα scripts, τα οποία χρησιμοποιούνται για την μελέτη των επιδόσεων του SOM για διαφορετικά πλέγματα, μέτρα αποστάσεων και πλήθος νευρώνων είναι τα **runSom.m, sunSomAll.m και runSomNeurons.m.** Τα δύο πρώτα χρησιμοποιούνται για την μελέτη της επίδοσης του δικτύου για διαφορετικά είδη πλέγματος και μέτρα αποστάσεων. Η διαφορά έγκειται στο ότι στο script runSom.m ο χρήστης μπορεί να επιλέξει είδος πλέγματος και μέτρο απόστασης, ενώ στο δεύτερο script γίνεται εκπαίδευση του δικτύου για όλους τους πιθανούς συνδυασμούς. Το script runSomNeurons.m εκπαιδεύει το δίκτυο για διαφορετικό πλήθος νευρώνων (list_of_arch), χρησιμοποιώντας τη συνάρτηση runSom.

Όλες οι μετρήσεις έγιναν για orderLR = 0.9, tuneLR = 0.1 και orderSteps = 250.

Τα διαφορετικά είδη πλέγματος είναι:

- **gridtop:** Τετραγωνικό πλέγμα.
- **hextop και hexagonalTopology:** Εξαγωνικά πλέγματα.
- **randtop:** Τυχαίας διάταξης πλέγμα.

Οι συναρτήσεις απόστασης (για τον υπολογισμό των αποστάσεων μεταξύ των νευρώνων) είναι:

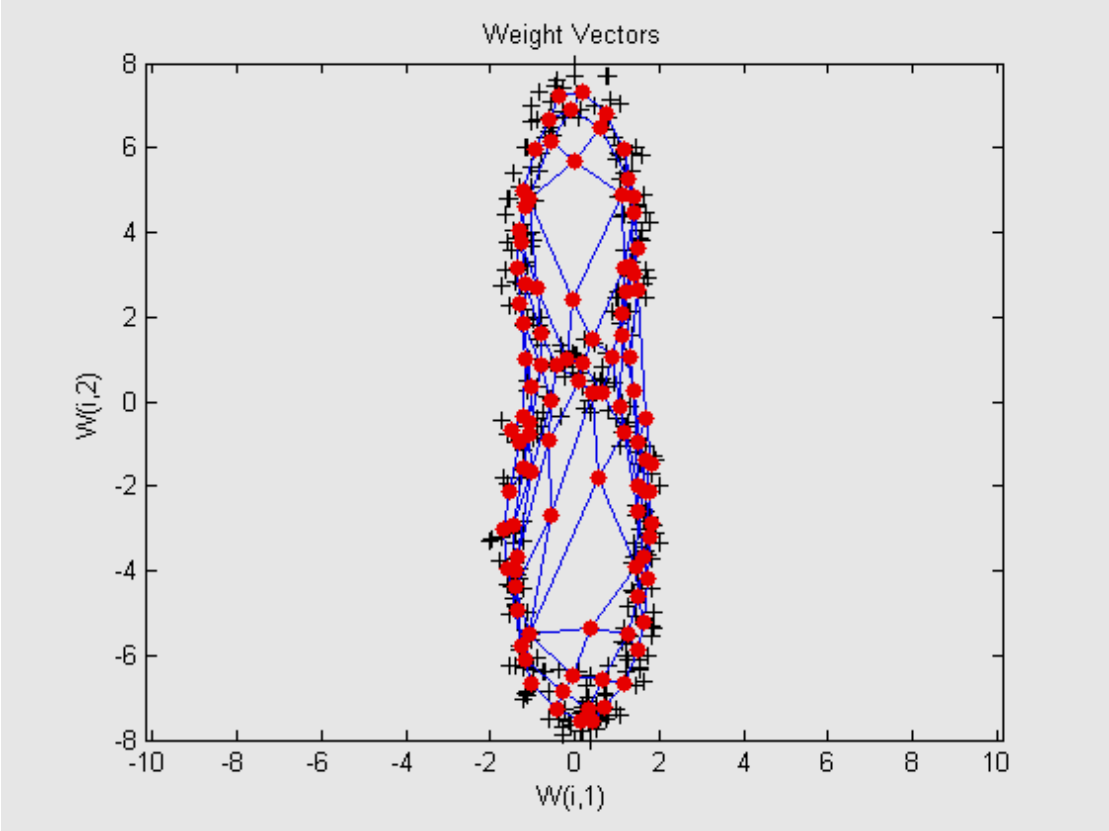
- **dist (Euclidean distance weight function)**
- **boxdist (Box distance function)**
- **mandist (Manhattan distance function)**
- **linkdist (Link distance function)**

Ακολουθούν διαγράμματα για διαφορετικά μέτρα απόστασης και τύπους πλεγμάτων. Οι μετρήσεις έγιναν για πλήθος νευρώνων [5 5], [8 8] και [10 10], ωστόσο δεν περιέχονται στο zip, λόγω περιορισμένου χώρου. Ενδεικτικά παρουσιάζονται τα διαγράμματα για μέγεθος πλέγματος [10 10]. Επειδή οι πιθανοί συνδυασμοί είναι 16, επιλέχθηκαν να παρουσιαστούν οι πιο αντιπροσωπευτικές μετρήσεις για κάθε παράμετρο.

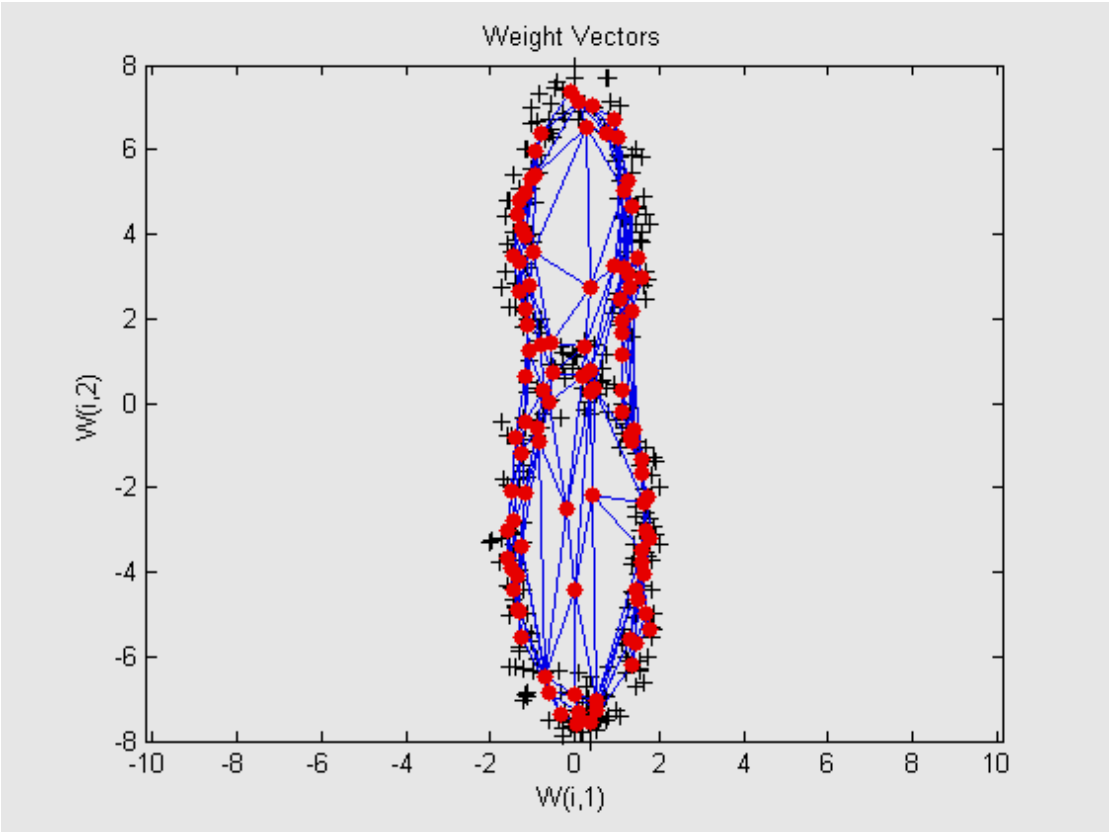
Σύγκριση τύπων πλέγματος:

Παρουσιάζονται τα διαγράμματα της εκπαίδευσης του δικτύου για τα EightData δεδομένα με χρήση της συνάρτησης απόστασης dist:

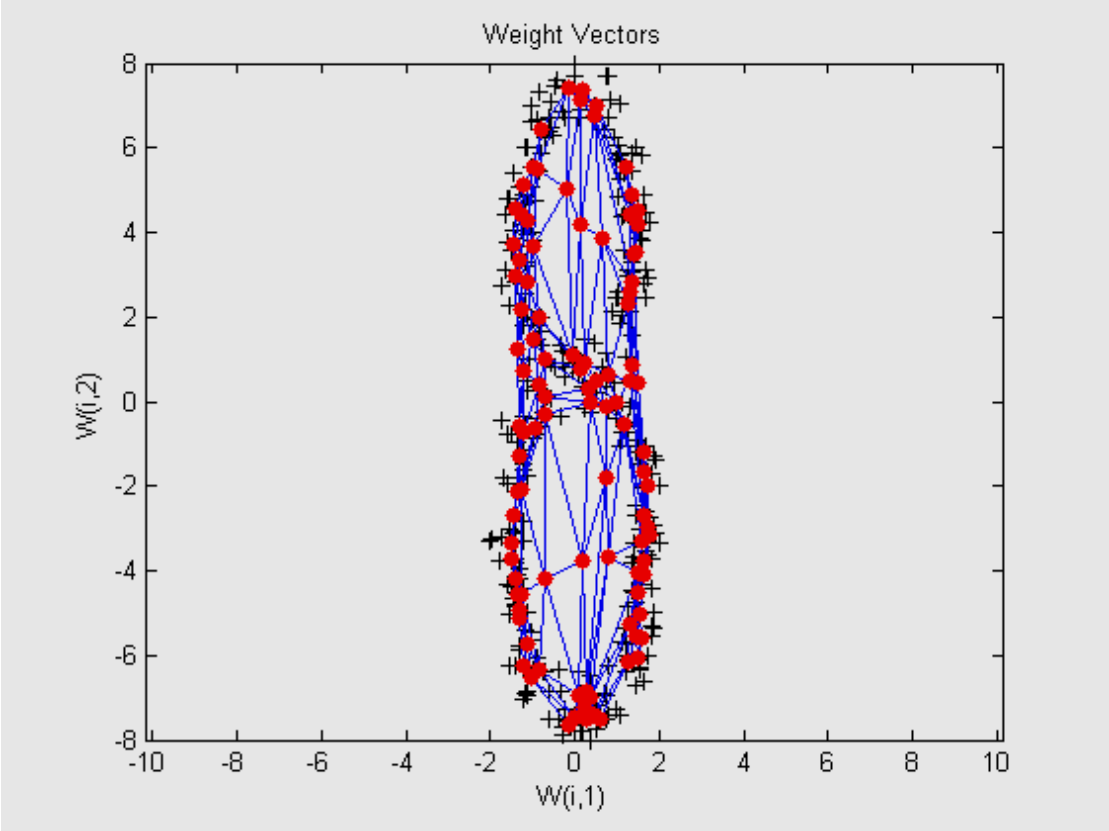
gridtop:



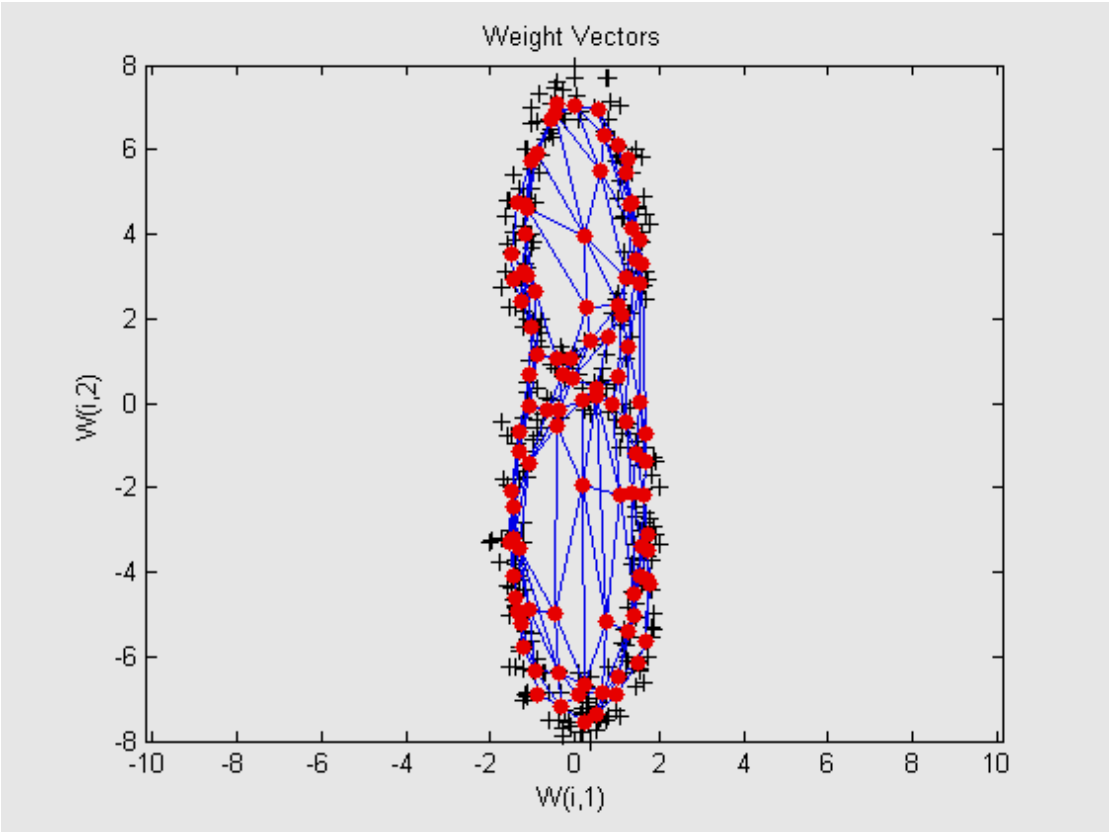
hextop:



hexagonalTopology:



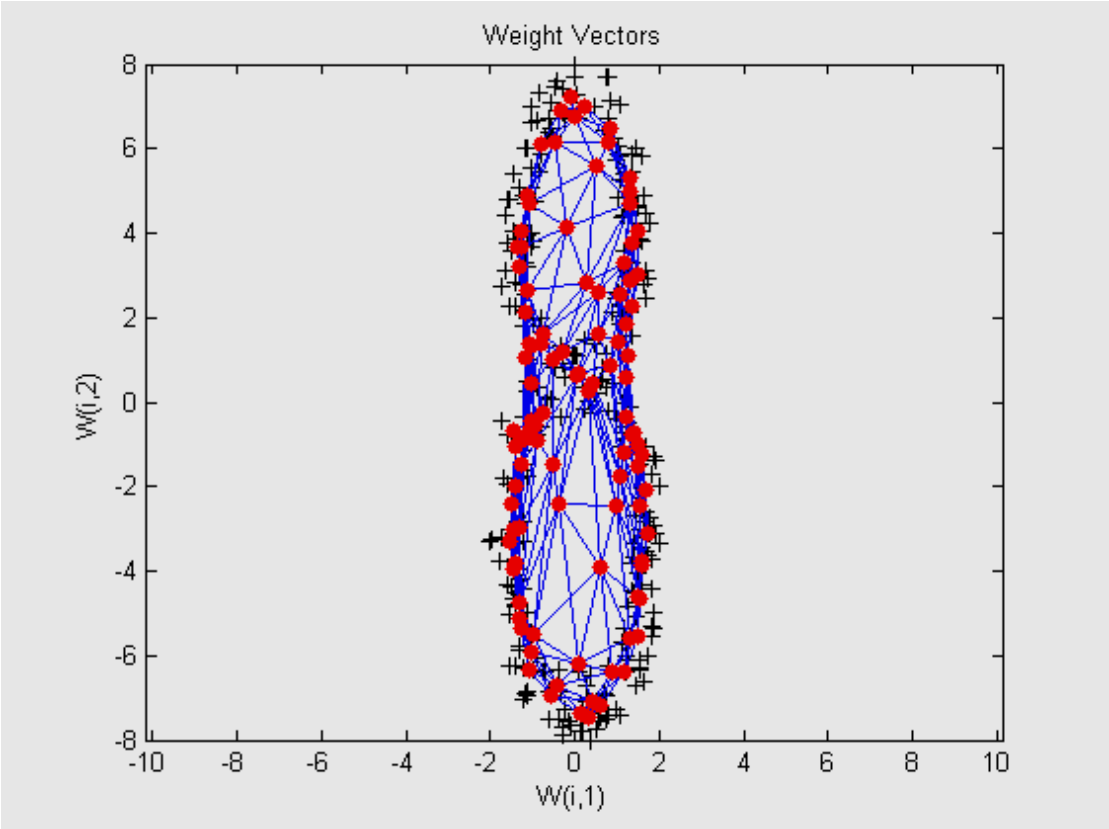
randtop:



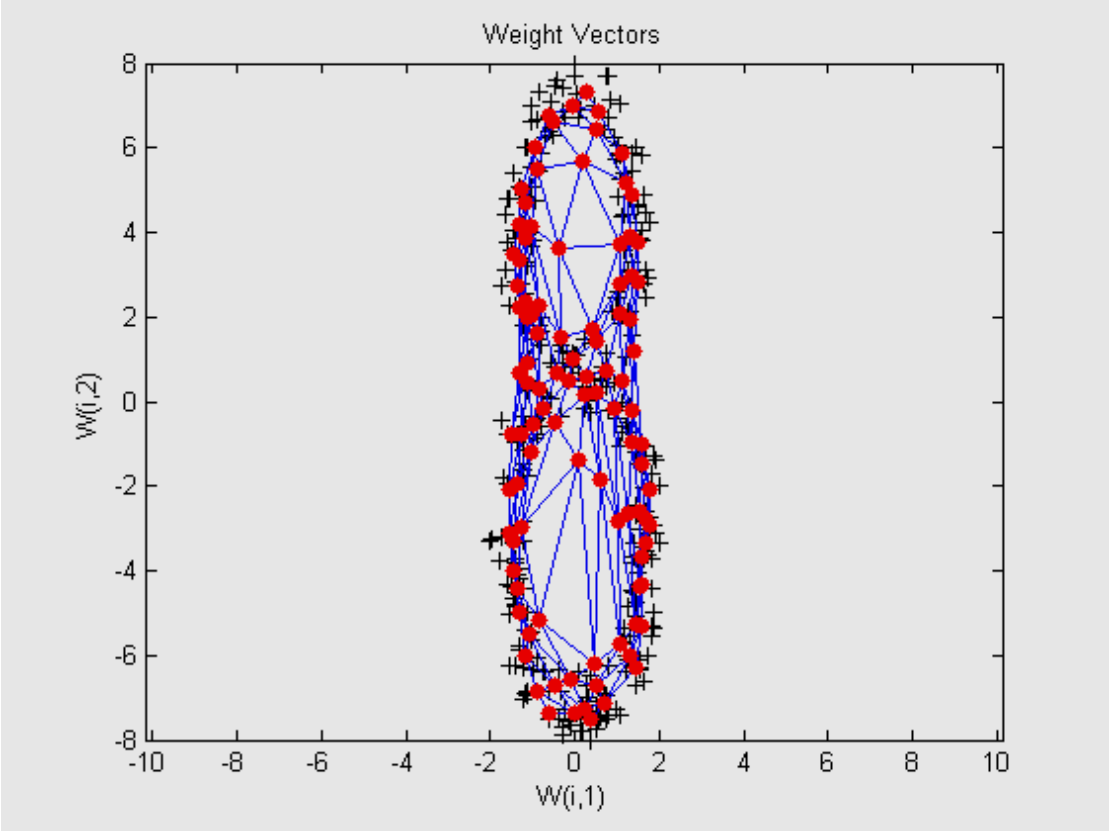
Παρατηρήσεις:

Σύμφωνα με τα παραπάνω διαγράμματα, η συνάρτηση hextop παρουσιάζει σχετικά καλύτερη επίδοση από τις υπόλοιπες συναρτήσεις, καθώς περισσότεροι νευρώνες έχουν διαταχθεί κοντά στις συντεταγμένες των σημείων εισόδου. Επίσης καλή επίδοση παρατηρείται για την διάταξη hexagonalTopology και σε ορισμένες περιπτώσεις για την διάταξη randtop. Η gridtop στα παραπάνω διαγράμματα παρουσιάζει καλή επίδοση, ωστόσο ανάλογα με το μέτρο της απόστασης μπορεί να μην έχει πάντα ικανοποιητικά αποτελέσματα. Ένα τέτοιο παράδειγμα παρατίθεται στη συνέχεια για μέτρο απόστασης boxdist:

gridtop:



hextop:

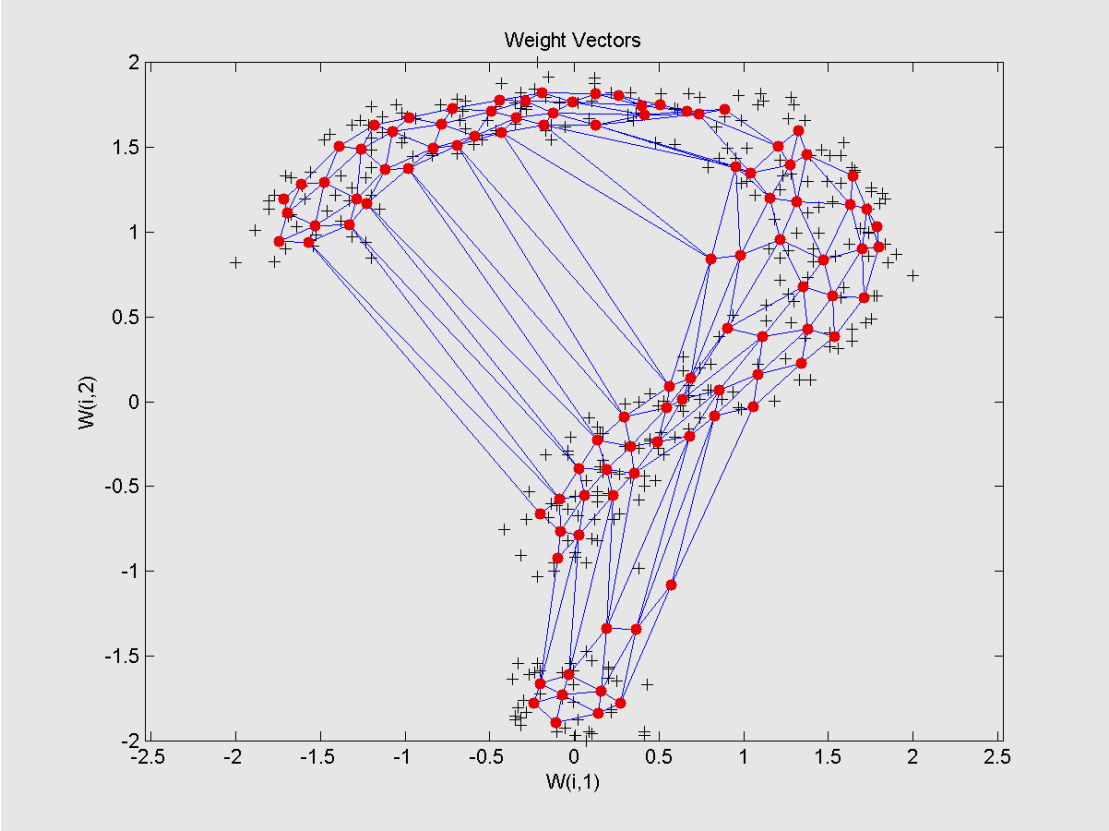


Αυτό είναι αναμενόμενο, καθώς στην εξαγωνική τοπολογία ένας νικητής-νευρώνας επηρεάζει μεγαλύτερο πλήθος γειτονικών νευρώνων (6 άμεσοι νευρώνες γείτονες), «οδηγώντας» τους σε περιοχές κοντά στα δεδομένα εισόδου, με αποτέλεσμα το δίκτυο να συγκλίνει ταχύτερα στη λύση (σε λιγότερες εποχές). Αντίθετα σε ένα τετραγωνικό πλέγμα, το πλήθος των γειτονικών νευρώνων είναι πιο περιορισμένο (4 άμεσοι νευρώνες γείτονες), επομένως οι νευρώνες απαιτούν περισσότερες εποχές για να διατάσσονται σε κατάλληλες θέσεις (οι οποίες δεν είναι πάντα κοντά στα δεδομένα εισόδου).

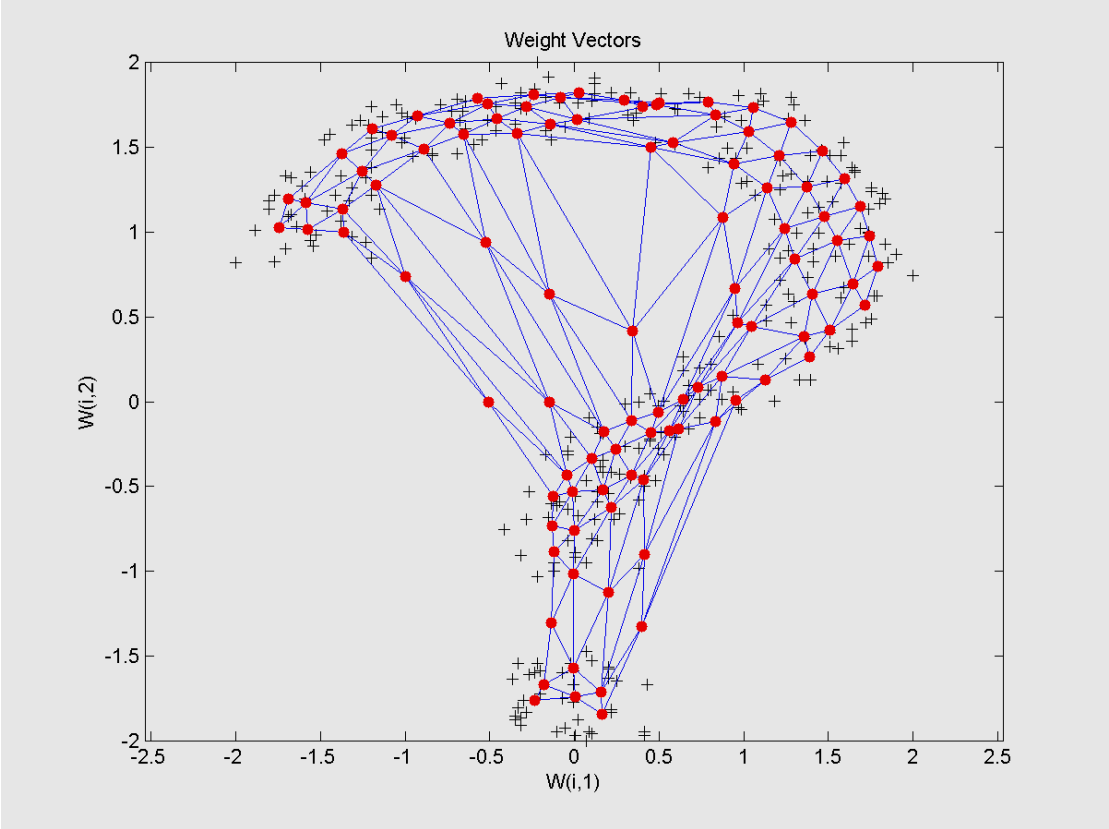
Σύγκριση συναρτήσεων απόστασης:

Παρουσιάζονται τα διαγράμματα της εκπαίδευσης του δικτύου για τα QuestionData δεδομένα με χρήση εξαγωνικού τύπου πλέγματος (hextop):

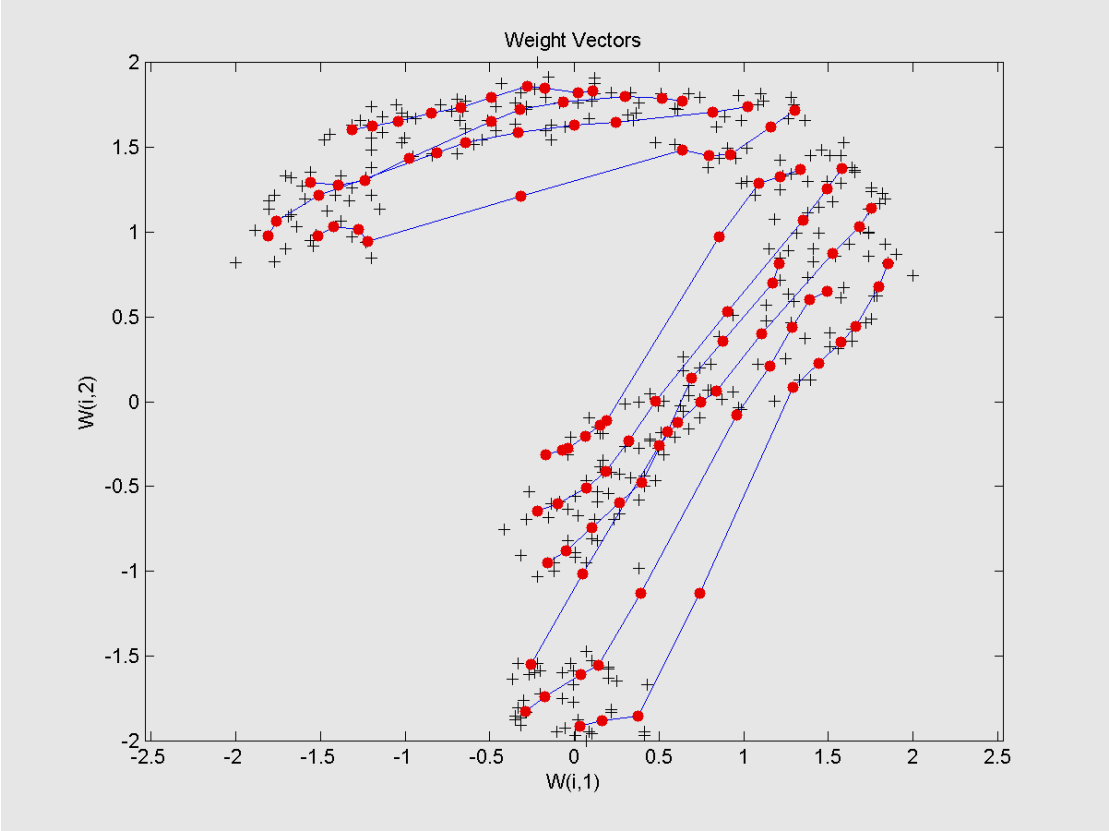
dist:



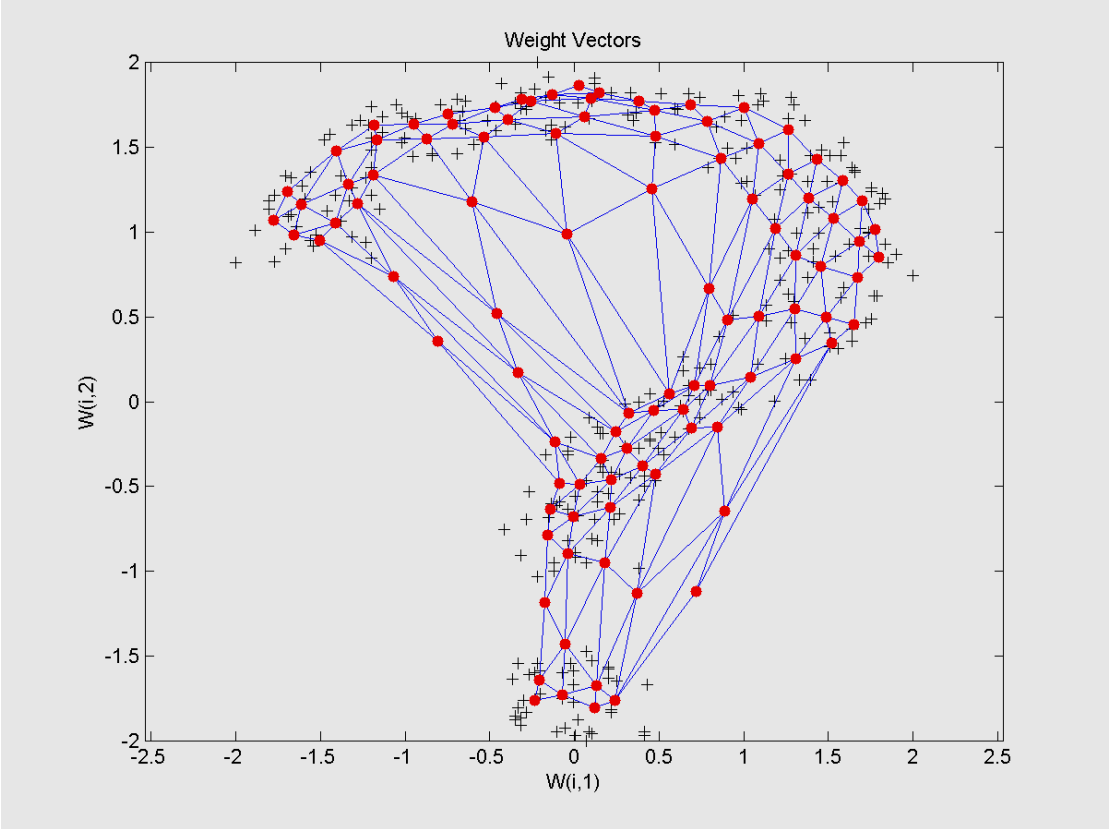
boxdist:



mandist:



linkdist:



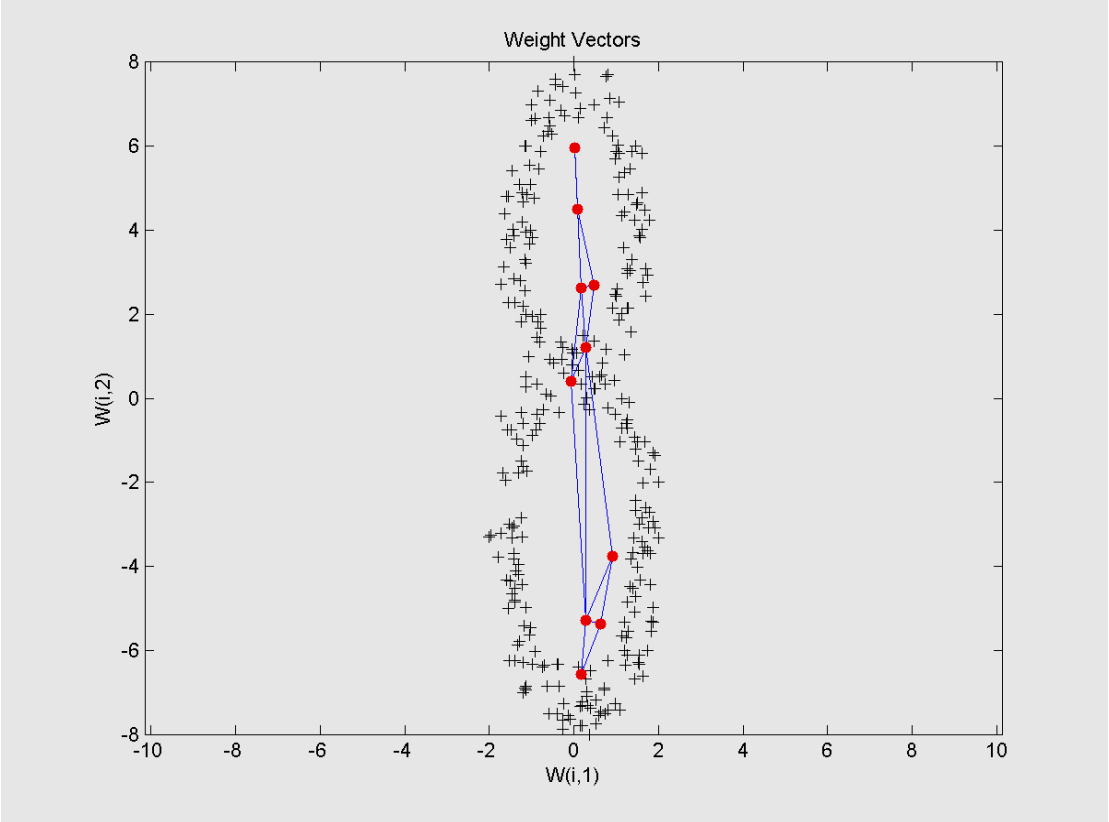
Παρατηρήσεις:

Σύμφωνα με τα παραπάνω διαγράμματα, καλύτερη εκπαίδευση παρουσιάζει το δίκτυο όταν χρησιμοποιούνται ως μέτρα αποστάσεων η Ευκλείδεια απόσταση (dist) και η απόσταση Manhattan (mandist). Αντίθετα οι linkdist και boxdist παρουσιάζουν σχετικά μικρότερη απόδοση, το οποίο μπορεί να οφείλεται στο ότι η απόσταση από έναν αρχικό νευρώνα προς έναν τελικό υπολογίζεται με βάση το πλήθος των ακμών του μονοπατιού που τους συνδέει (δηλαδή το πλήθος των βημάτων, τα οποία απαιτούνται για να φτάσουμε από τον αρχικό νευρώνα στον τελικό, μέσω του πλέγματος). Επειδή το μέτρο της απόστασης στις δύο προηγούμενες περιπτώσεις συνδέεται άμεσα με την τοπολογία του πλέγματος, η επίδοση τους μπορεί να μην είναι σταθερή (ανάλογα με το είδος του πλέγματος, η εκπαίδευση του δικτύου μπορεί να είναι αρκετά καλή ή το αντίθετο).

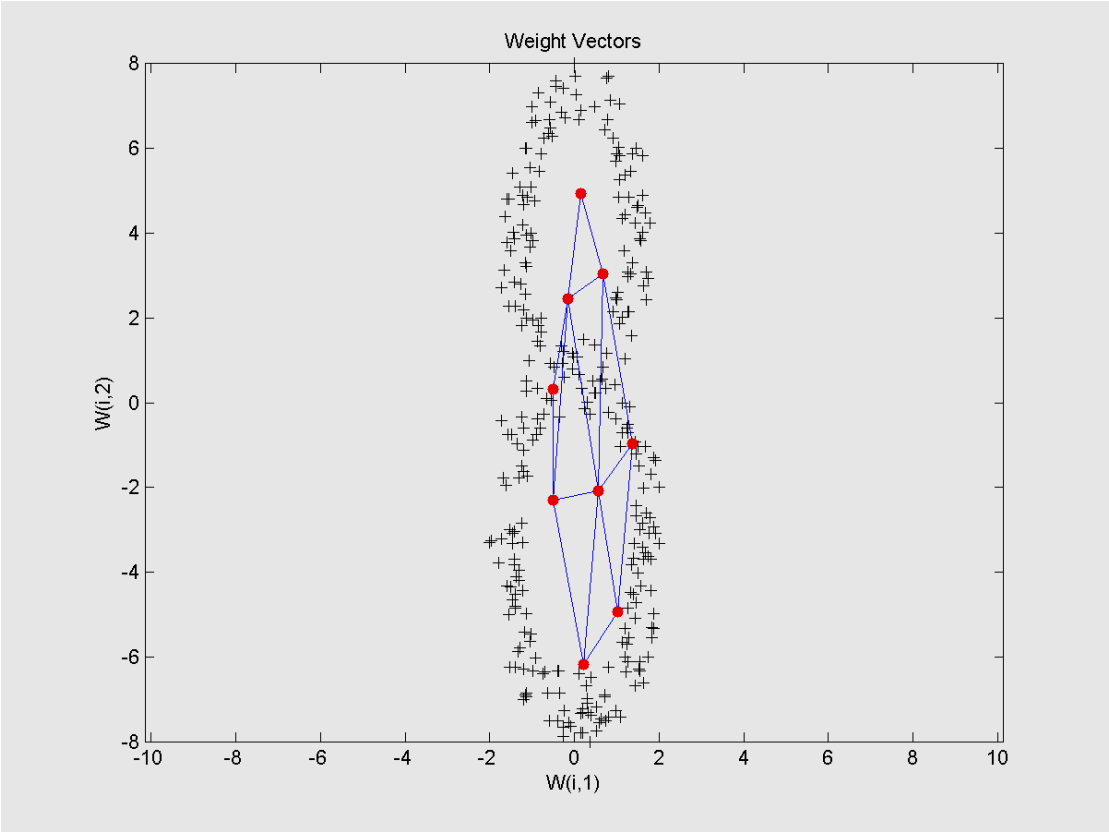
Επισημαίνεται ότι καλύτερες επιδόσεις παρατηρήθηκαν για τα ζεύγη παραμέτρων **(dist – hextop), (dist – hexagonalTopology) και (mandist – hextop).** Για το διαφορετικό πλήθος νευρώνων, έγιναν μετρήσεις για όλες τις παραπάνω διατάξεις (οι οποίες περιέχονται σε αντίστοιχους φακέλους στο zip). Ενδεικτικά παρουσιάζονται τα αποτελέσματα των πειραματισμών για τον συνδυασμό (dist – hextop) και δεδομένα εισόδου EightData.

Σύγκριση διαφορετικού πλήθους νευρώνων:

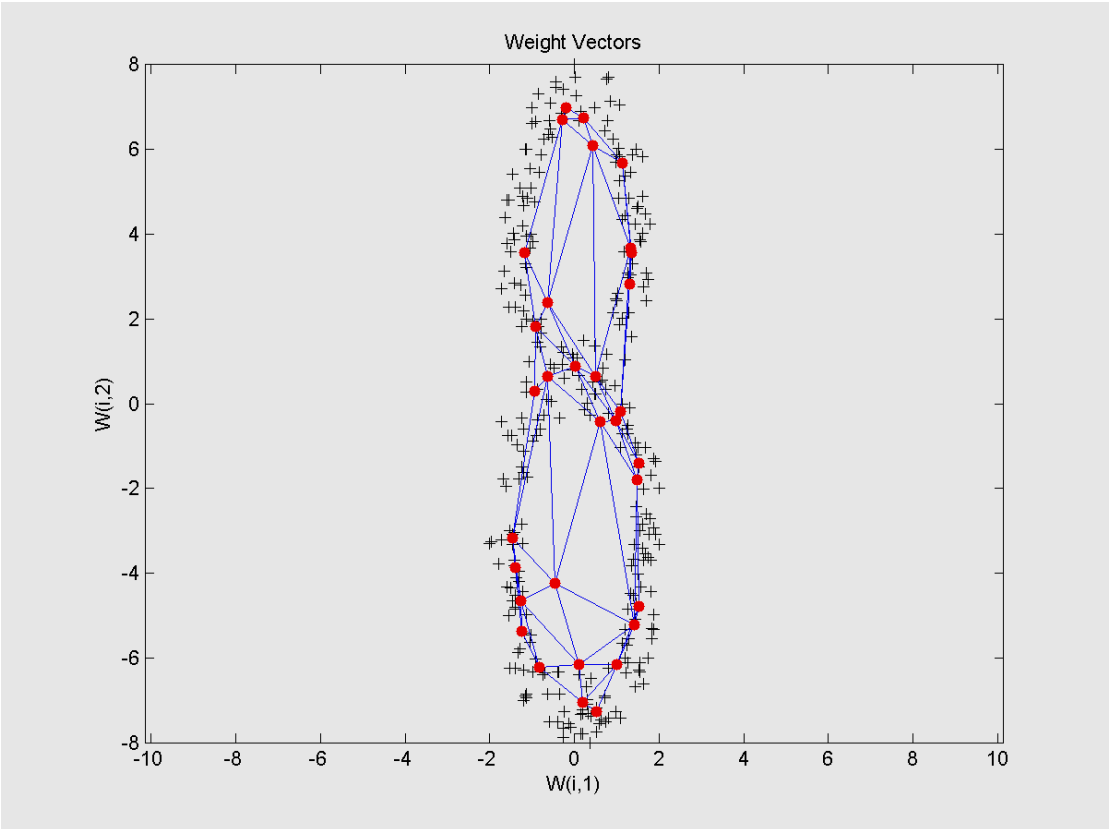
Grid [2 5]:



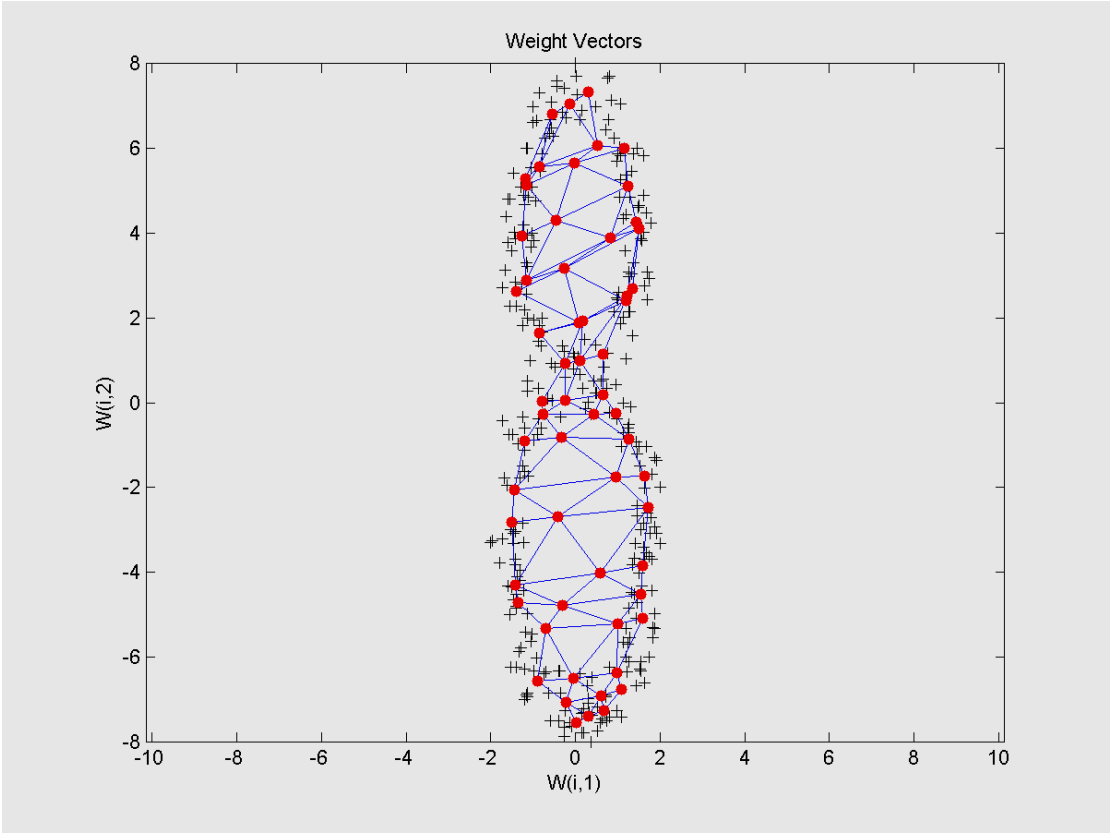
Grid [3 3]:



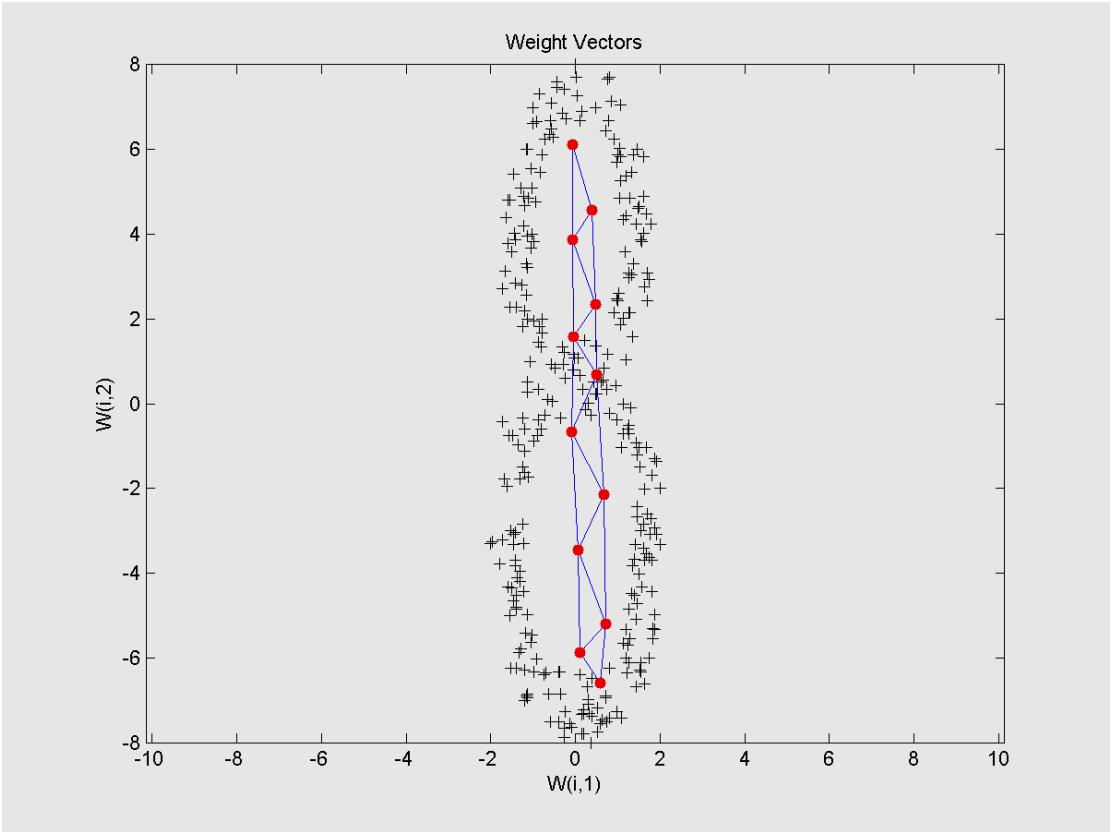
Grid [4 8]:



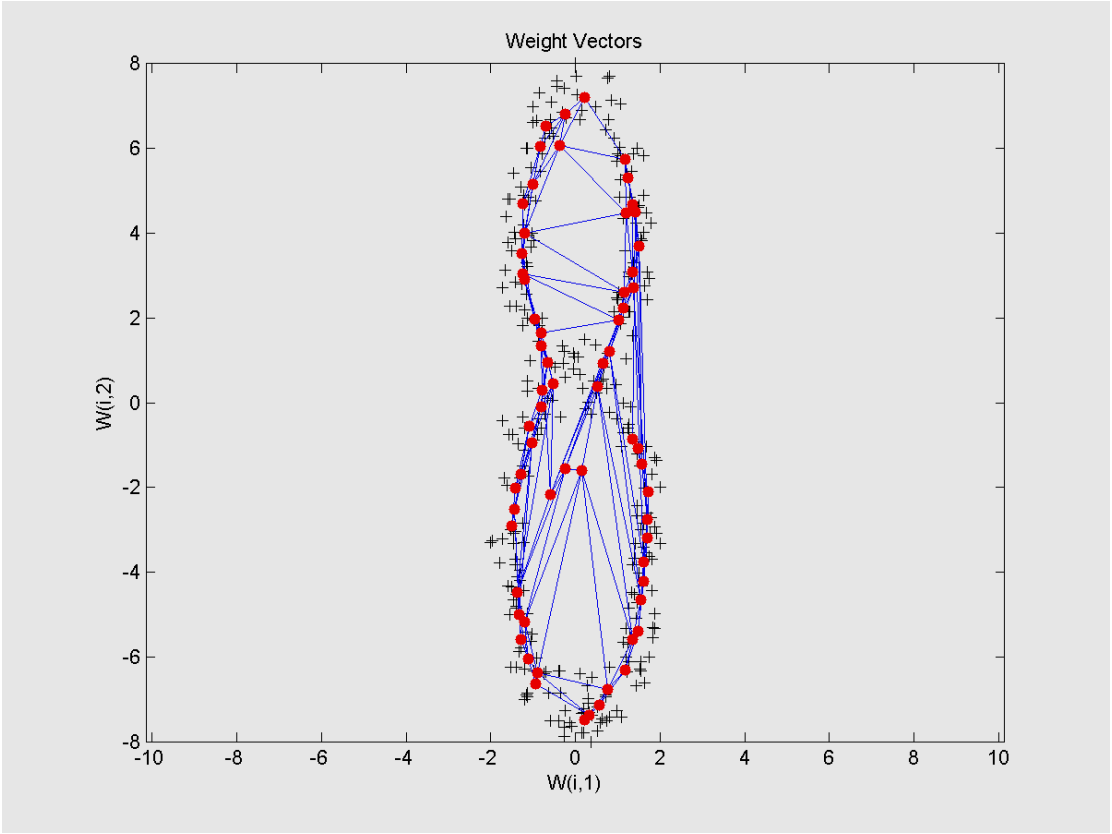
Grid [3 20]:



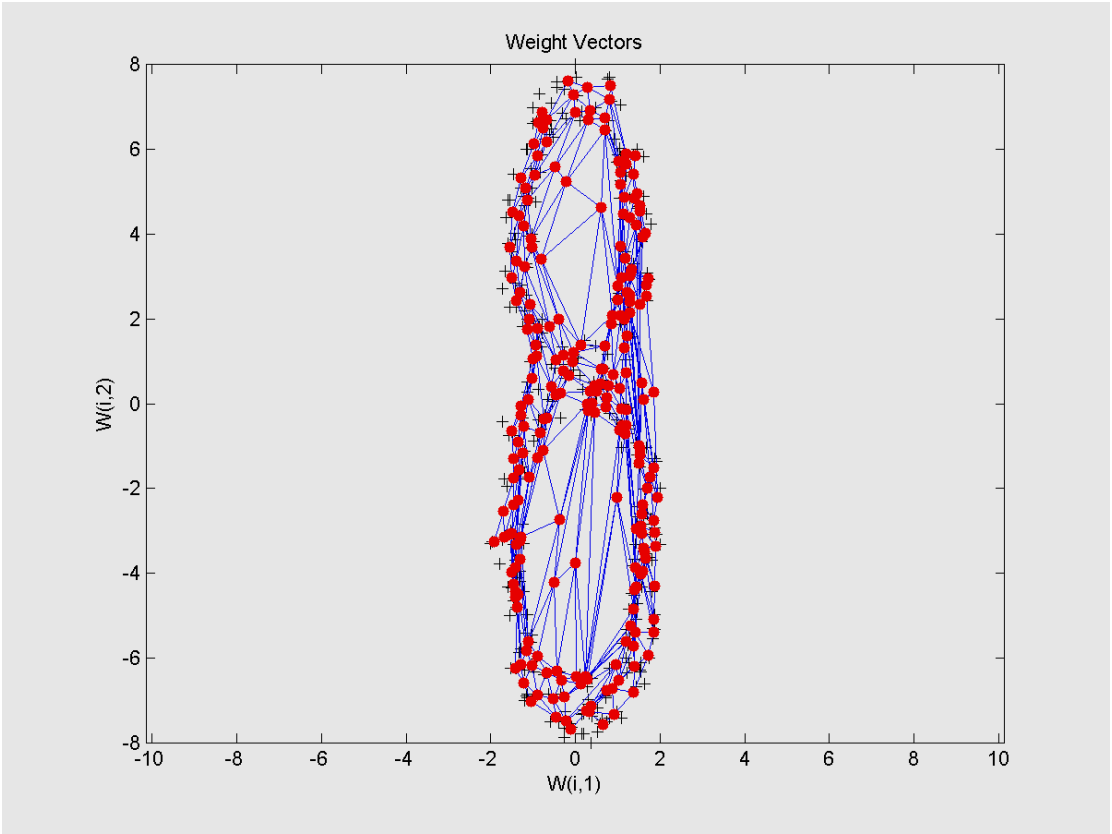
Grid [6 2]:



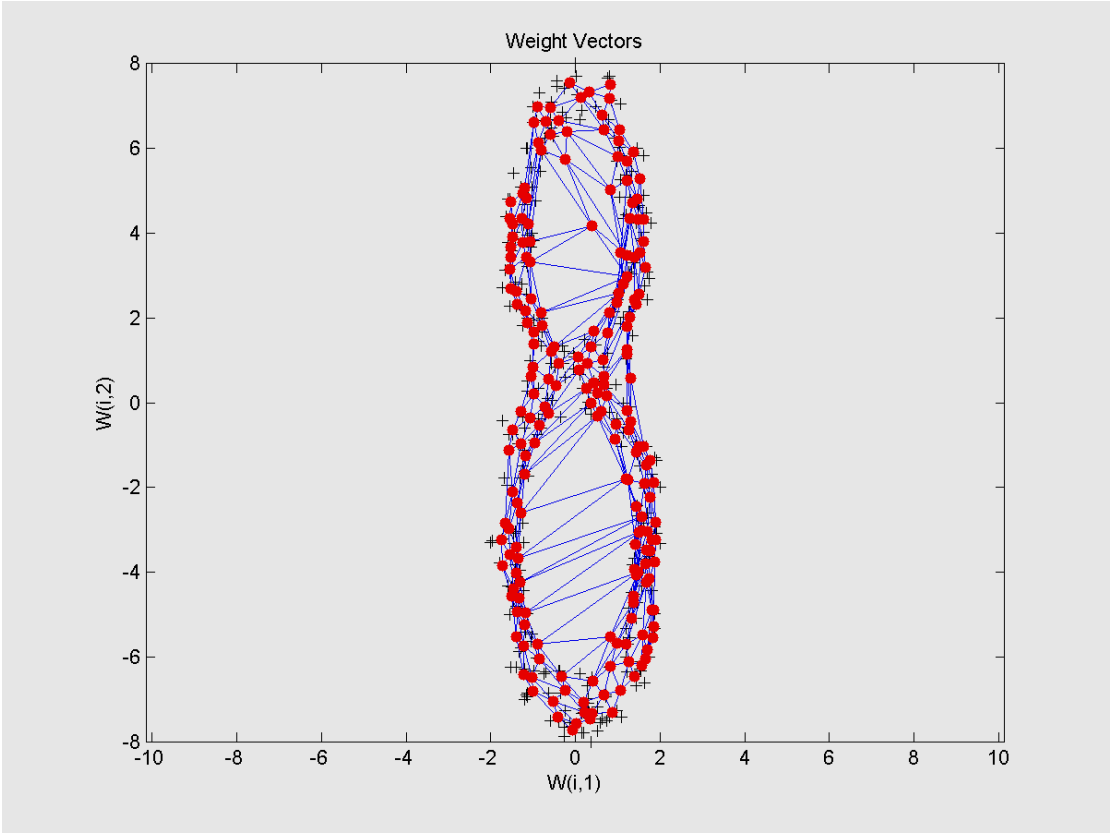
Grid [8 8]:



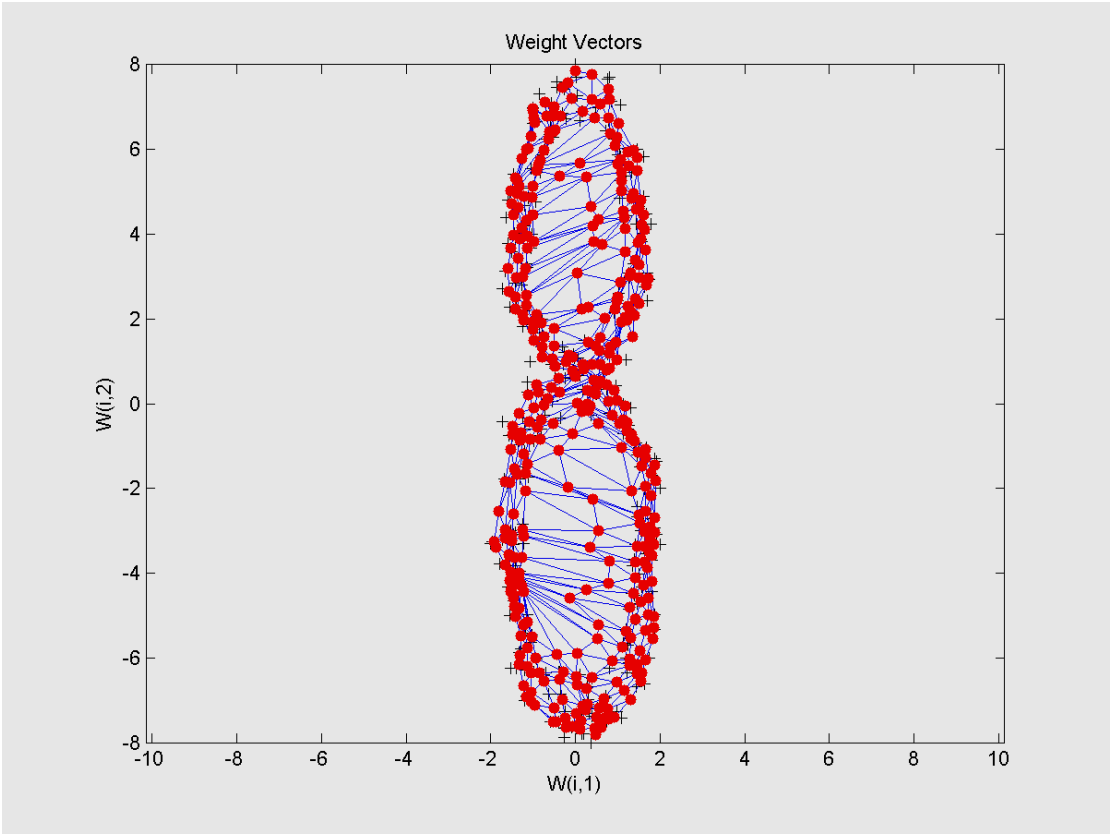
Grid [15 15]:



Grid [20 10]:



Grid [40 9]:



Παρατηρήσεις:

Σύμφωνα με τα παραπάνω διαγράμματα, για τη σωστή εκπαίδευση του δικτύου υπάρχει άμεση σύνδεση του πλήθους των νευρώνων με τα δεδομένα εισόδου. Μικρό πλήθος νευρώνων (Grid [3 3], Grid [2 5]) ή υπερβολικά μεγάλο πλήθος (Grid [40 9]) (σε σχέση με το μέγεθος των δεδομένων εισόδου) δεν εξασφαλίζει απαραίτητα ότι η εκπαίδευση του δικτύου είναι καλή. Συγκεκριμένα για μικρό πλήθος νευρώνων, υπάρχει περιορισμένη δυνατότητα για τον καθορισμό των θέσεων τους (η οποία εξαρτάται επίσης από την αντίστοιχη τοπολογία), με αποτέλεσμα να χάνεται πληροφορία ως προς τη διάταξη των δεδομένων εισόδου. Αντίθετα αν οι νευρώνες είναι πολλοί σε σχέση με το πλήθος των δεδομένων, δεν εξασφαλίζεται ότι η εκπαίδευση του δικτύου είναι καλύτερη από την εκπαίδευση με μικρότερο πλήθος νευρώνων.

Επίσης η διάταξη των νευρώνων στο πλέγμα (αριθμός νευρώνων ανά γραμμή και στήλη) επηρεάζει αρκετά την εκπαίδευση του δικτύου. Αυτό φαίνεται από τα διαγράμματα των Grid [8 8] και Grid [3 20], στα οποία ενώ ο συνολικός αριθμός νευρώνων είναι σχεδόν ίδιος (64 και 60 αντίστοιχα), υπάρχει μεγάλη διαφορά στην απόδοση του δικτύου. Η παραπάνω παρατήρηση οφείλεται στο διαφορετικό αριθμό γειτόνων, τους οποίους έχουν οι νευρώνες σε κάθε Grid.

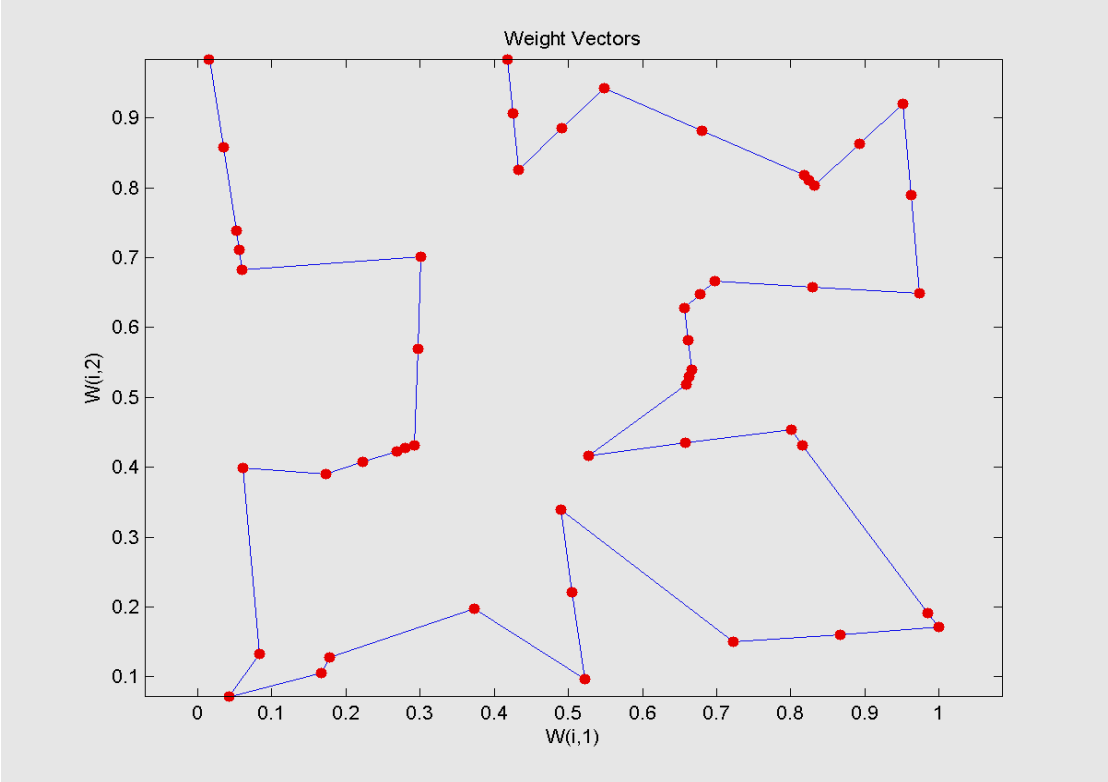
Αντίστοιχες είναι οι παρατηρήσεις για δεδομένα εισόδου QuestionData.

2Β) Για την επίλυση του προβλήματος TSP χρησιμοποιούνται επιπλέον τα scripts runSomTSP.m και somCreateTSP.m. Το πρώτο script χρησιμοποιείται για την ολοκληρωμένη εκπαίδευση του δικτύου (με χρήση των προηγούμενων συναρτήσεων), ενώ το δεύτερο script αποτελεί μια τροποποιημένη έκδοση της συνάρτησης somCreate, στην οποία το μέτρο της απόστασης υπολογίζεται είτε με χρήση της συνάρτησης ring_distances, είτε μέσω της linkdist. Ο χρήστης έχει δυνατότητα να ορίσει ποια συνάρτηση χρησιμοποιείται, μέσω των ορισμάτων εισόδου.

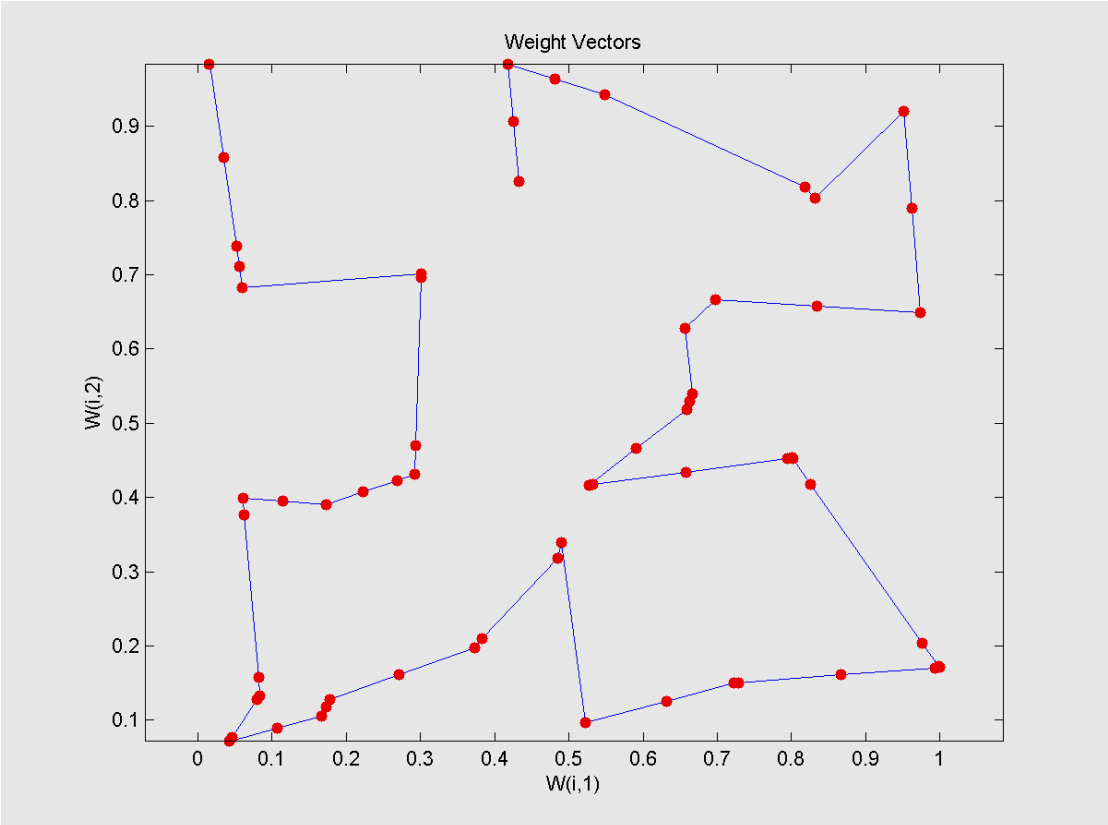
Οι μετρήσεις των πειραμάτων έγιναν για εξαγωνικό μονοδιάστατο πλέγμα και για διαφορετικά πλήθη νευρώνων (μεγαλύτερα ωστόσο από το πλήθος των πόλεων). Ενδεικτικά παρουσιάζονται παρακάτω ορισμένα από τα διαγράμματα, τα οποία προέκυψαν μετά την εκπαίδευση του δικτύου:

Με χρήση της συνάρτησης linkdist:

Grid [80 1]:

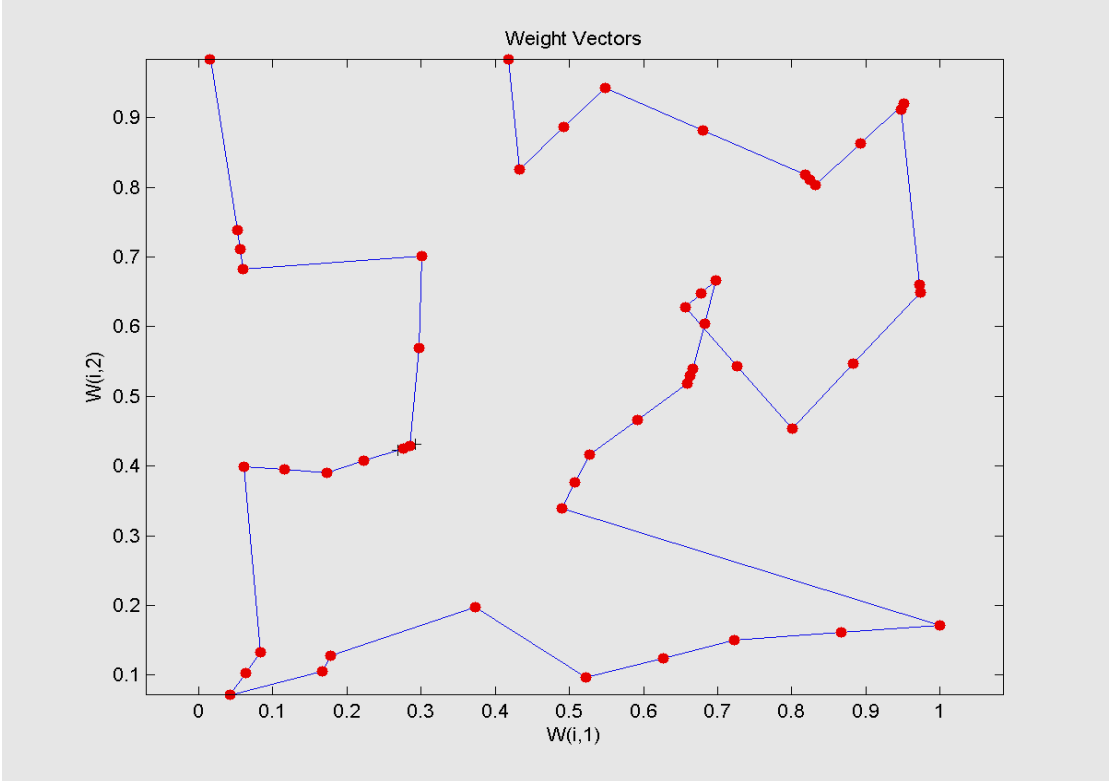


Grid [1 110]:

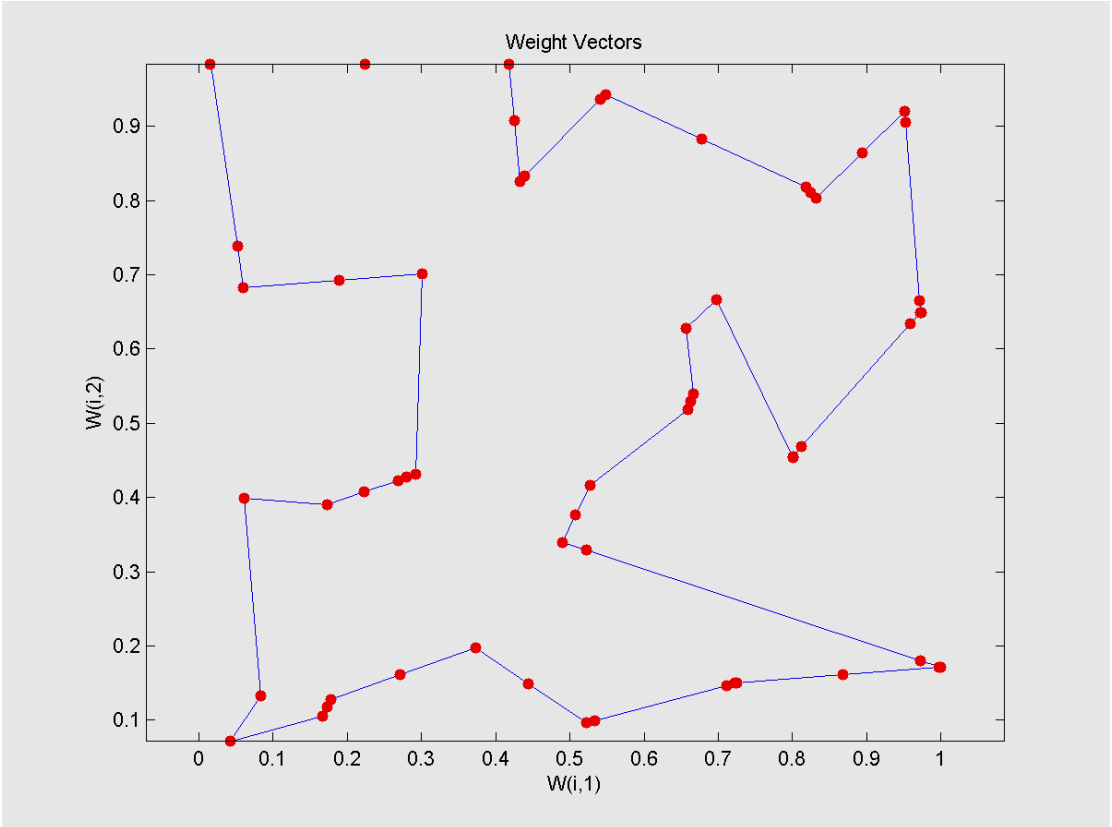


Με χρήση της συνάρτησης ring_distances:

Grid [80 1]:



Grid [1 110]:



Παρατηρήσεις:

Αρχικά επιβεβαιώνεται ότι οι νευρώνες, μετά την εκπαίδευση του δικτύου, έχουν βάρη ίδια με τις συντεταγμένες των πόλεων και γειτονικοί κόμβοι του δακτυλίου έχουν γειτονικά βάρη. Ωστόσο υπάρχουν ορισμένες διαφοροποιήσεις στην εκπαίδευση με χρήση των δύο συναρτήσεων.

Στην εκπαίδευση του δικτύου με χρήση της linkdist, ο πρώτος και ο τελευταίος νευρώνας δε συνδέονται, σε αντίθεση με την περίπτωση της συνάρτησης ring_distances.

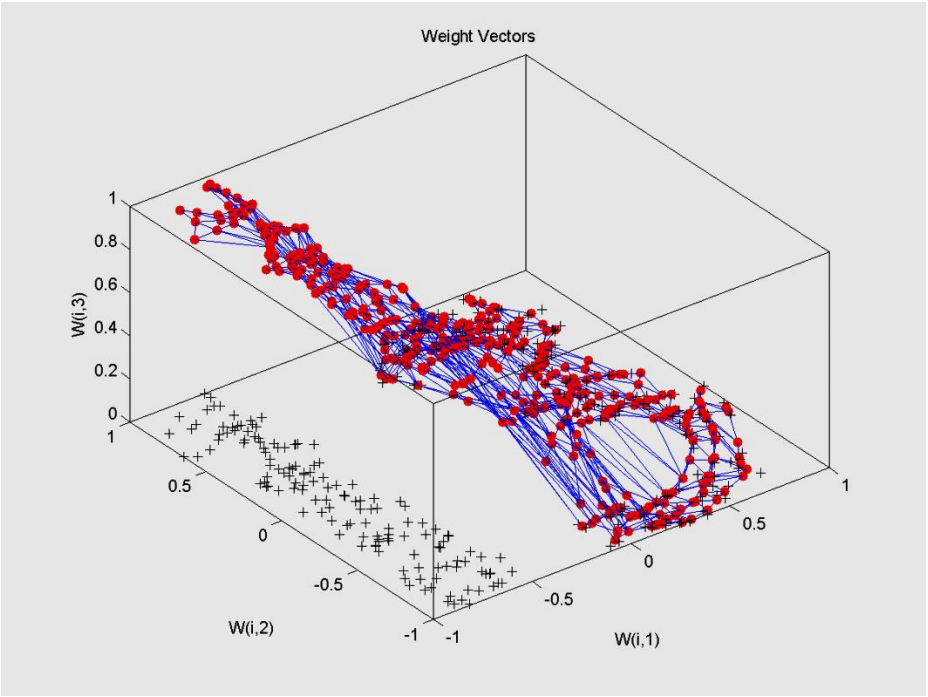
Επίσης με βάση τους πίνακες αποστάσεων, επιβεβαιώνεται ότι και στις δύο περιπτώσεις κάθε νευρώνας απέχει από τους γειτονικούς τους 1 ακμή και η απόσταση είναι ίση με το μήκος του μονοπατιού, το οποίο ενώνει τον νευρώνα πηγή με τον νευρώνα προορισμό.

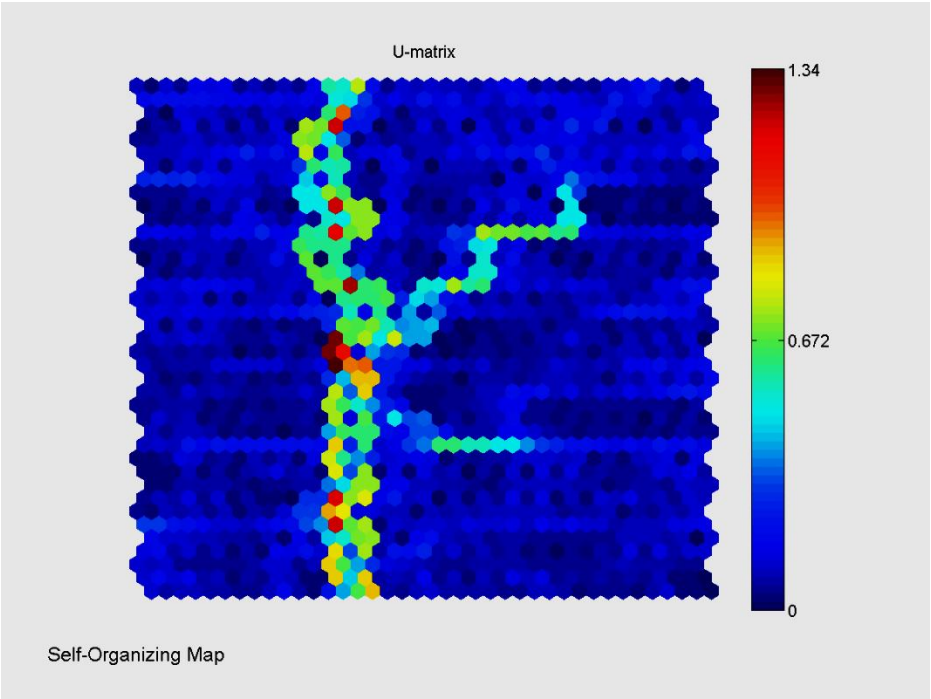
Τέλος παρατηρείται ότι ενώ το δίκτυο καταφέρνει να εκπαιδευτεί για την επίλυση του προβλήματος TSP, η διάταξη των νευρώνων μπορεί να είναι σχετικά διαφοροποιημένη κάθε φορά. Στα παραπάνω διαγράμματα της linkdist υπάρχει κάποια διαφοροποίηση στον αρχικό και τελικό νευρώνα - προορισμό. Αυτό μπορεί να οφείλεται στην τυχαία αρχικοποίηση των βαρών του δικτύου, με αποτέλεσμα οι νευρώνες να προσπίπτουν σε συντεταγμένες διαφορετικών πόλεων κάθε φορά, μετά την εκπαίδευση του δικτύου. Παρόμοιες παρατηρήσεις σημειώνονται και για τη συνάρτηση ring_distances, ωστόσο σε αυτή την περίπτωση δεν μπορεί να γίνει διάκριση αρχικού και τελικού νευρώνα, καθώς αυτοί ενώνονται.

2Γ) Για τη μελέτη της οπτικοποίησης του δικτύου χρησιμοποιήθηκε το script runSomGroup.m. Στο συγκεκριμένο script περιλαμβάνεται επίσης η διαδικασία διαχωρισμού των προτύπων και των νευρώνων σε ομάδες. Όλες οι μετρήσεις περιέχονται σε αντίστοιχο φάκελο. Ενδεικτικά παρουσιάζονται 3 διαγράμματα για διαφορετικό πλήθος νευρώνων και παραμέτρους. Οι τοπολογίες, οι οποίες επιλέχθηκαν για την εκτέλεση των μετρήσεων παρουσίασαν τις καλύτερες επιδόσεις στα προηγούμενα ερωτήματα.

Grid [20 20] – dist – hextop:

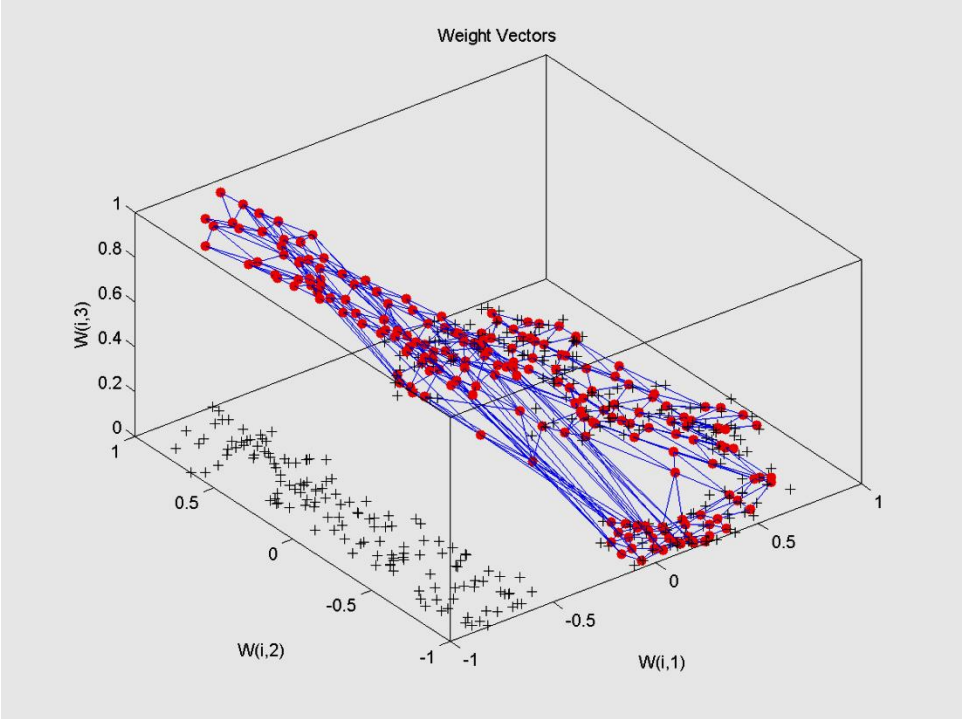
Αριθμός νευρώνων στην Ομάδα 0: 246
Αριθμός νευρώνων στην Ομάδα 1: 135
Αριθμός νευρώνων στα σύνορα: 19

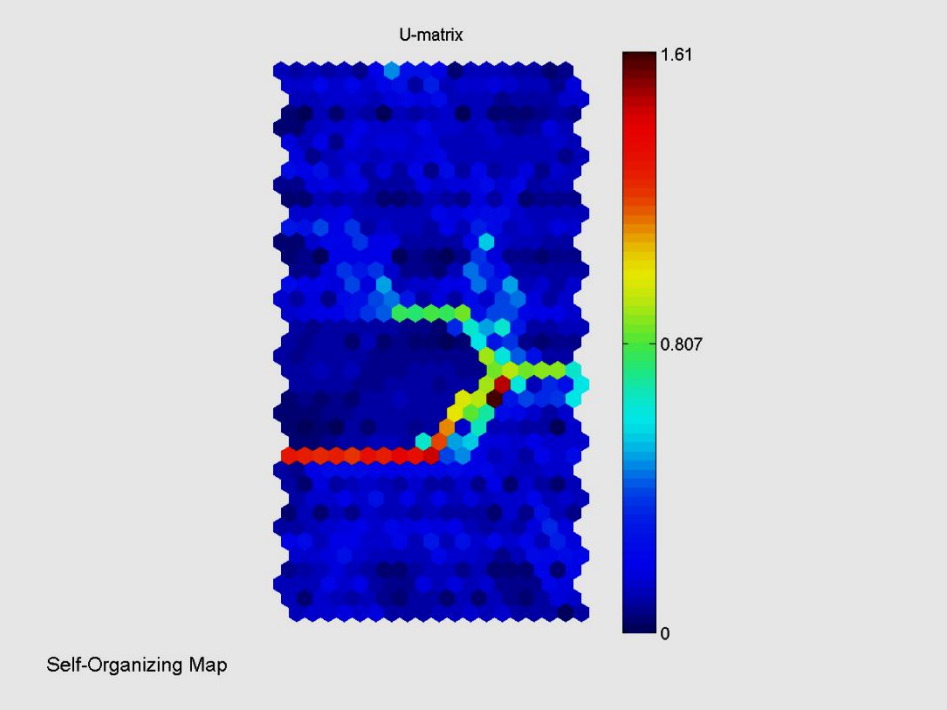




Grid [10 20] – dist – hexagonalTopology:

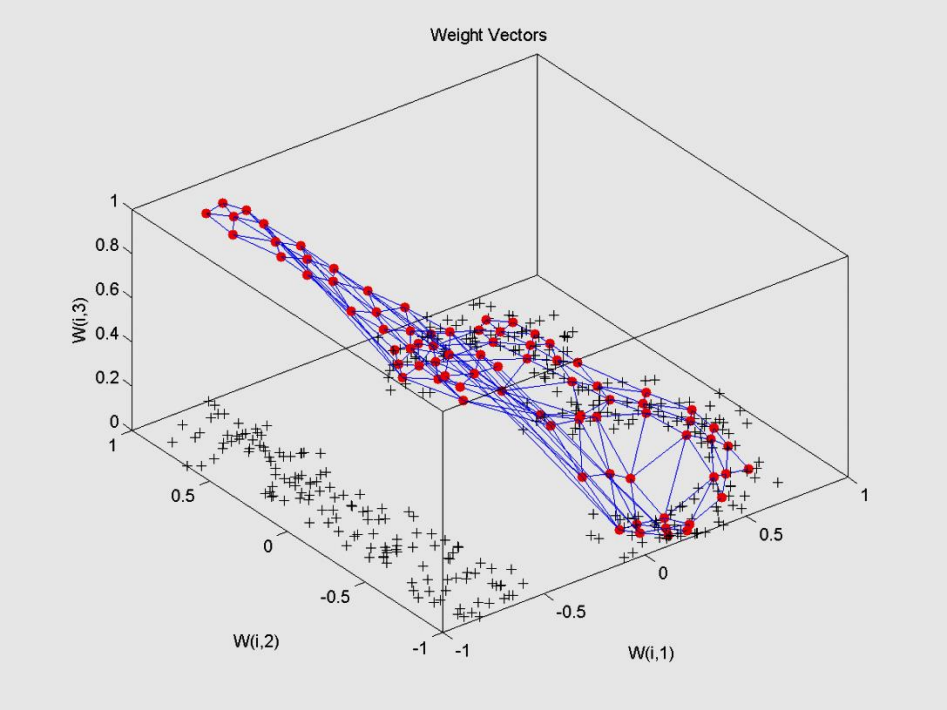
Αριθμός νευρώνων στην Ομάδα 0: 128
Αριθμός νευρώνων στην Ομάδα 1: 66
Αριθμός νευρώνων στα σύνορα: 6

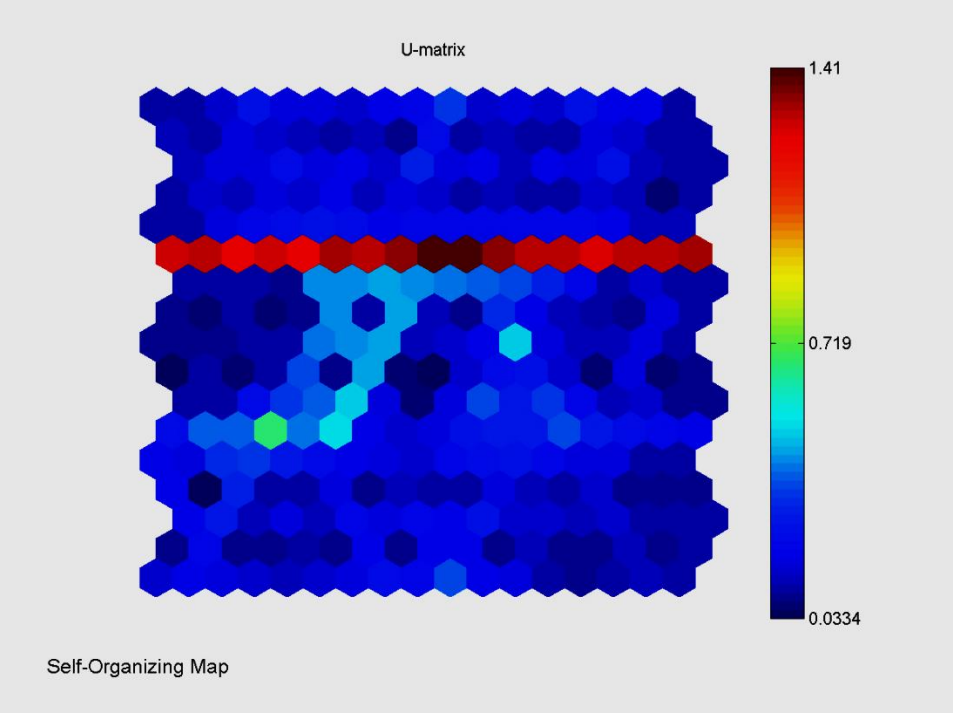




Grid [9 9] – dist – hextop:

Αριθμός νευρώνων στην Ομάδα 0: 54
Αριθμός νευρώνων στην Ομάδα 1: 27
Αριθμός νευρώνων στα σύνορα: 0





Παρατηρήσεις:

Για όλες τις μετρήσεις, οι οποίες έγιναν επαληθεύτηκε ότι στην **Ομάδα 0 ανήκουν 206** πρότυπα, ενώ στην **Ομάδα 1 144**. Ο υπολογισμός των προτύπων έγινε με απαρίθμηση των μηδενικών και των άσσων στο τρίτο πεδίο του πίνακα GroupPatterns. Γενικά το μέγεθος των ομάδων, όπως αυτές παρουσιάζονται στα διαγράμματα Umatrix, φαίνεται να είναι ανάλογο του πλήθους των προτύπων που ανήκουν σε καθεμία. Αυτό είναι αναμενόμενο, καθώς περισσότεροι νευρώνες συγκεντρώνονται στις περιοχές που βρίσκονται περισσότερα πρότυπα, «παρασύροντας» μαζί τους γειτονικούς νευρώνες. Επειδή τα δεδομένα είναι χωρισμένα σε δύο ομάδες, οι αντίστοιχοι νευρώνες επίσης διαχωρίζονται σε δύο ομάδες και το αποτέλεσμα του διαχωρισμού παρουσιάζεται στο Umatrix.

Για καθένα από τα παραπάνω διαγράμματα, αναγράφονται οι νευρώνες, οι οποίοι ανήκουν σε κάθε ομάδα. Ο υπολογισμός τους έγινε με απαρίθμηση των μηδενικών και άσσων από το τρίτο πεδίο του πίνακα βαρών IW. Οι νευρώνες, οι οποίοι παρουσιάζονται ως σύνορα, είναι εκείνοι των οποίων η συγκεκριμένη τιμή δεν ήταν ούτε 0 ούτε 1. Σε όλες τις παραπάνω περιπτώσεις επιβεβαιώνεται η αναλογία του πλήθους των προτύπων και των νευρώνων σε κάθε ομάδα. **Συγκεκριμένα οι νευρώνες, οι οποίοι κατατάσσονται στην Ομάδα 0 είναι σχεδόν διπλάσιοι από αυτούς της Ομάδας 1, ενώ το πλήθος των νευρώνων στα σύνορα είναι πολύ μικρότερο (γεγονός το οποίο φανερώνει την ικανοποιητική εκπαίδευση του δικτύου).** Αναφέρεται ότι οι νευρώνες – σύνορα αντιστοιχούν στις ερυθρές και γενικότερα ανοιχτόχρωμες αποχρώσεις στους πίνακες Umatrix, ενώ με μπλε χρώμα παρουσιάζονται οι νευρώνες στα δύο σύνολα.

Επισημαίνεται ότι παρουσιάστηκαν 3 διαγράμματα με διαφορετικό πλήθος νευρώνων, για να επιβεβαιωθεί η σωστή εκπαίδευση του δικτύου. Στο πρώτο διάγραμμα το SOM αποτελείται από νευρώνες μεγαλύτερου πλήθους από τον αριθμό των προτύπων. Στα άλλα δύο διαγράμματα το μέγεθος του δικτύου είναι αρκετά μικρότερο. Και στις τρείς περιπτώσεις γίνεται διαχωρισμός των νευρώνων κατ' αναλογία με το μέγεθος των ομάδων. Ωστόσο στις δύο τελευταίες διατάξεις επιβεβαιώνεται η ικανότητα του SOM να μειώνει τις διαστάσεις του προβλήματος, συμπιέζοντας την αντίστοιχη πληροφορία. Αυτό προκύπτει καθώς διατηρείται ο διαχωρισμός των νευρώνων στις δύο ομάδες, ενώ το μέγεθος του SOM μειώνεται.

Στα παραπάνω διαγράμματα παρατηρείται επίσης διαφοροποίηση στα σύνορα των ομάδων. Αυτό μπορεί να οφείλεται στην τυχαία αρχικοποίηση των βαρών των νευρώνων, η οποία γίνεται κατά τη δημιουργία του δικτύου SOM. Διαφορετικές αρχικοποιήσεις των βαρών οδηγούν σε διαφορετική εκπαίδευση του δικτύου, επομένως οι νευρώνες νικητές μπορεί να διαφοροποιούνται και να αλλάζει η διάταξη τους μέσα στο χώρο.

Τέλος αναφέρεται ότι από τις παραπάνω παρατηρήσεις, επιβεβαιώνονται οι ιδιότητες του SOM (κατηγοριοποίηση προτύπων, οπτικοποίηση δεδομένων μεγάλων διαστάσεων και αφαίρεση διαστάσεων προβλήματος).

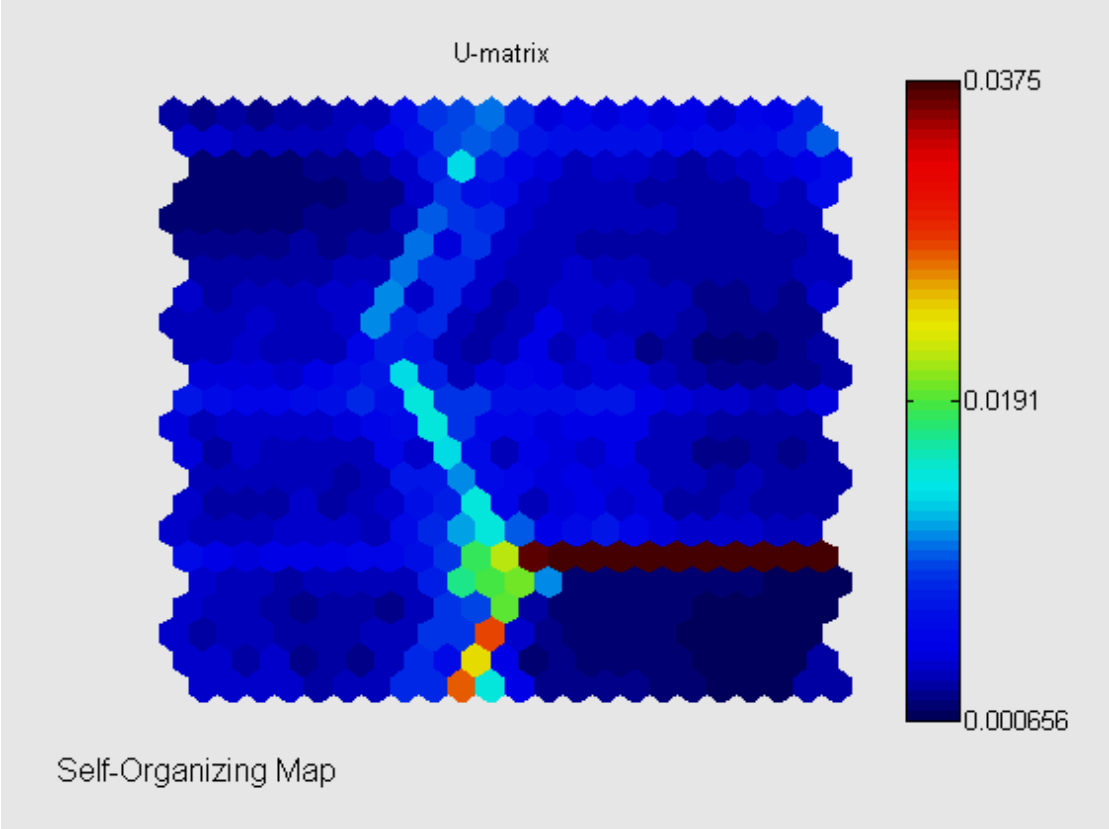
Εφαρμογή αυτό-οργανούμενων χαρτών για την ομαδοποίηση και οπτικοποίηση συλλογής εγγράφων (document clustering and visualization)

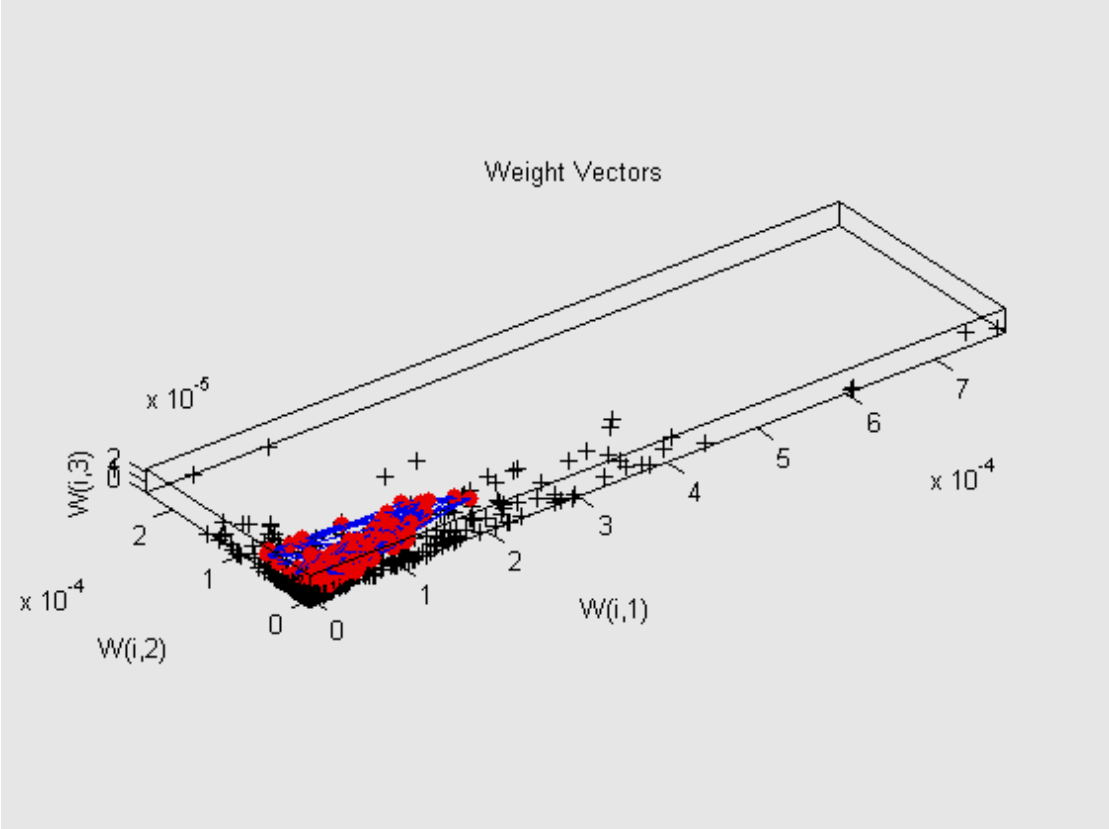
Για την εκπαίδευση του δικτύου SOM χρησιμοποιείται το script runSomDoc.m, ενώ για την επεξεργασία των ερωτημάτων της εκφώνησης το script DocProcess.m.

4A) Ο υπολογισμός του τελικού πίνακα γίνεται με χρήση της συνάρτησης tfidf1.m και περιέχεται στο script runSomDoc.m. Ο τελικός ζητούμενος πίνακας είναι ο new_P.

4B) Για την απάντηση του συγκεκριμένου ερωτήματος, έγινε εκπαίδευση του δικτύου αρκετές φορές για διαφορετικές παραμέτρους. Τα αποτελέσματα ορισμένων εκπαιδεύσεων είναι αποθηκευμένα σε .mat αρχεία και περιέχονται στο φάκελο zip. Τα αντίστοιχα _process.mat, τα οποία περιέχουν τις απαντήσεις των ερωτημάτων 4Γ, μπορούν να προκύψουν από την εκτέλεση του script DocProcess.m. Ενδεικτικά η εκπαίδευση έγινε για συνδυασμό παραμέτρων **(Grid[10 10] – dist – hextop – 600epochs), (Grid[12 12] – mandist – hexagonalTopology – 600 epochs) και (Grid[12 12] – dist – hexagonalTopology – 800epochs).** Επιλέγεται εξαγωνικό πλέγμα και κατάλληλο πλήθος εποχών, όπως αναφέρεται στην εκφώνηση, ενώ το πλήθος των νευρώνων διαμορφώνεται με βάση το αντίστοιχο πλήθος των εγγράφων. Για μέτρο απόστασης επιλέγεται η Ευκλείδεια απόσταση (dist), καθώς στα προηγούμενα ερωτήματα παρουσίασε αρκετά καλή επίδοση (ομοίως και για την απόσταση mandist).

Στην επόμενη σελίδα παρουσιάζονται τα διαγράμματα για εκπαίδευση δικτύου με παραμέτρους (Grid[12 12] – dist – hexagonalTopology – 800epochs).


U-matrix
0.0375
0.0191
0.000656
Self-Organizing Map


Weight Vectors
x 10^-5
W(i,3)
2
0
x 10^-4
W(i,2)
2
1
0
0
1
2
3
4
5
6
7
x 10^-4
W(i,1)

Με βάση τον πίνακα Umatrix οι νευρώνες του δικτύου φαίνεται να χωρίζονται σε τρεις ομάδες. Επειδή υπάρχει αναλογία στο χώρο διάταξης μεταξύ του πλήθους των νευρώνων και των προτύπων, συμπεραίνουμε ότι τα έγγραφα μπορούν να ομαδοποιηθούν σε τρία σύνολα.

4Γ) Για την απάντηση των επόμενων ερωτημάτων χρησιμοποιείται το δίκτυο SOM, το οποίο εκπαιδεύτηκε για παραμέτρους (Grid[12 12] – dist – hexagonalTopology – 800epochs) – doc3_process.mat (λόγω του όγκου δεδομένων δεν περιέχεται στο zip αλλά μπορεί να προκύψει από την εκτέλεση του προαναφερόμενου script). Ακολουθεί η μεθοδολογία υλοποίησης του κάθε ερωτήματος, ενώ τα αποτελέσματα των μετρήσεων παρουσιάζονται αναλυτικά στο .mat:

i) Για τον υπολογισμό των εγγράφων, τα οποία ανήκουν σε κάθε νευρώνα, γίνεται χρήση της συνάρτησης somOutput. Συγκεκριμένα για κάθε pattern βρίσκεται ο νευρώνας νικητής (winner_pos) και ενημερώνεται ένας πίνακας Nx1 αυξάνοντας την αντίστοιχη θέση (number_of_docs [winner_pos]) κατά 1. Ο ζητούμενος πίνακας number_of_docs αποθηκεύει για κάθε νευρώνα, πόσα έγγραφα του αντιστοιχούν (δηλαδή πόσα έγγραφα τον «επέλεξαν» ως νευρώνα νικητή).

ii) Για την εύρεση των εγγράφων με την μικρότερη απόσταση από κάθε νευρώνα, γίνεται αντίστοιχη διαδικασία με αυτή του προηγούμενου ερωτήματος. Συγκεκριμένα αρχικά υπολογίζεται ο πίνακας x (διαστάσεων N x P) των αρνητικών Ευκλείδειων αποστάσεων και ο ανάστροφος του x2 (διαστάσεων P x N). Έπειτα με χρήση της συνάρτησης compet υπολογίζεται για κάθε νευρώνα (στήλη του πίνακα x2) ποιο έγγραφο (γραμμή του πίνακα x2) έχει την μικρότερη προς αυτόν απόσταση. Έχοντας υπολογίσει τον παραπάνω πίνακα μικρότερων αποστάσεων (output_docs2) βρίσκεται η αντιστοιχία των εγγράφων (τίτλων) για κάθε νευρώνα με χρήση του πίνακα titles.

Το ζητούμενο αποθηκεύεται στον πίνακα min_dist_title διαστάσεων N x 1, ενώ παρατηρείται ότι γειτονικοί νευρώνες μπορεί να απέχουν από το ίδιο έγγραφο την μικρότερη απόσταση. Αυτό είναι αναμενόμενο, καθώς με την εκπαίδευση του δικτύου κάθε νευρώνας νικητής παρασύρει τους γειτονικούς κόμβους επηρεάζοντας αναλόγως τα βάρη τους.

iii) Η εύρεση των 3 όρων με το μεγαλύτερο βάρος για κάθε νευρώνα επιτυγχάνεται με αναζήτηση στον πίνακα βαρών του δικτύου IW. Αρχικά γίνεται φθίνουσα ταξινόμηση του πίνακα κατά γραμμές με χρήση της συνάρτησης sort και αποθηκεύονται οι δείκτες της μετάθεσης (της ταξινόμησης) των όρων σε έναν πίνακα Indexes. Έπειτα με χρήση των πινάκων Indexes και terms βρίσκονται οι 3 μεγαλύτερου βάρους όροι για κάθε νευρώνα, οι οποίοι αποθηκεύονται στον πίνακα max_weight_terms.

iv) Για το συγκεκριμένο ερώτημα παρατηρείται ότι οι όροι «network» και «function» βρίσκονται στις θέσεις [1,1] και [7,1] του πίνακα terms αντίστοιχα. Επομένως αρκεί να ελέγξουμε για κάθε νευρώνα i αν ικανοποιείται η σχέση :

$$\boldsymbol{IW(i,1) > 0.3 * max(IW(:)) \ \& \ IW(i,7) > 0.3 * max(IW(:))}$$

(δηλαδή αν κάθε νευρώνας, για τους δύο όρους, έχει βάρος μεγαλύτερο του 30% της μέγιστης τιμής του πίνακα IW). Οι ζητούμενοι νευρώνες αποθηκεύονται στον πίνακα nodes, ωστόσο παρατηρείται ότι για τη συγκεκριμένη εκπαίδευση του δικτύου δεν υπάρχουν νευρώνες που να ικανοποιούν την παραπάνω συνθήκη. Επίσης παρατηρείται ότι για τον πίνακα Patterns του NIPS500.mat, μετά την επεξεργασία των βαρών με τη χρήση της tfidf1, η μέγιστη τιμή του πίνακα είναι περίπου 0.02, ενώ οι μέση τιμή των όρων «network» και «function» είναι της τάξης 10^{-5} και 10^{-6}. Ο πίνακας IW των βαρών μετά την εκπαίδευση του δικτύου παρουσιάζει τιμές όμοιες με τις παραπάνω, επομένως δεν υπάρχουν νευρώνες που να ικανοποιούν τη συνθήκη. Επισημαίνεται ότι αυτή η παρατήρηση δεν αποκλείει το γεγονός να έχει γίνει κάποιο σφάλμα στην εκπαίδευση του δικτύου.

v) Όπως αναφέρεται παραπάνω δεν υπάρχουν νευρώνες, οι οποίοι ικανοποιούν τη συνθήκη του ερωτήματος (iv), επομένως δεν έχει βρεθεί κάποια απάντηση για το ζητούμενο ερώτημα. Ωστόσο στο script DocProcess.m υλοποιείται η μέθοδος υπολογισμού του συγκεκριμένου ερωτήματος. Θεωρητικά έχοντας ένα σύνολο νευρώνων αποθηκευμένων στον πίνακα nodes (λύση του προηγούμενου ερωτήματος), η μέση τιμή του κάθε όρου υπολογίζεται με χρήση της συνάρτησης mean ανά στήλες. Μετά τον υπολογισμό της, αρκεί να διαιρέσουμε με τη μέγιστη τιμή του πίνακα IW και να πολλαπλασιάσουμε με 100 για να βρεθεί το ζητούμενο ποσοστό.