

ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ ΚΑΙ ΕΥΦΥΗ ΥΠΟΛΟΓΙΣΤΙΚΑ ΣΥΣΤΗΜΑΤΑ

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 2: Αυτο-οργανούμενοι χάρτες

Σε αυτή την εργαστηριακή άσκηση εξετάζεται μια κλάση νευρωνικών δικτύων μη επιβλεπόμενης μάθησης, οι αυτο-οργανούμενοι χάρτες (Self Organizing Maps – SOMs) του Kohonen. Στην προσπάθεια διερεύνησης των δυνατοτήτων του αυτο-οργανούμενου χάρτη (SOM), κατ' αρχάς υλοποιείται ένα σχετικά απλό δίκτυο, που επιτρέπει την ενδοσκόπηση στην αρχιτεκτονική, τους αλγορίθμους και τις λειτουργίες που συνιστούν το σύνολο ενός αυτο-οργανούμενου χάρτη. Εν συνεχεία, μέσω καθοδηγούμενων πειραμάτων διερευνώνται και μελετώνται οι δυνατότητες του SOM ενώ στο τελευταίο μέρος της άσκησης γίνεται και εφαρμογή σε πραγματικά δεδομένα (ομαδοποίηση εγγράφων).

1. Υλοποίηση ενός SOM

Σε πρώτη φάση, η εργαστηριακή άσκηση επικεντρώνεται κυρίως στην υλοποίηση ενός - απλού μεν αρκετά γενικού δε- αυτο-οργανούμενου χάρτη. Εφαρμόζοντας τις ακόλουθες διεξοδικές προδιαγραφές μπορεί κανείς να υλοποιήσει ένα SOM με μονοδιάστατο ή δισδιάστατο πλέγμα νευρώνων. Επιπρόσθετα, παρέχεται η δυνατότητα χρήσης διαφόρων τύπων πλέγματος (π.χ. κανονικό ή εξαγωνικό), καθώς και διαφόρων τύπων τοπολογικής γειτονιάς (για παράδειγμα Ευκλείδεια ή Manhattan). Επιπλέον, κάνοντας χρήση (μιας αρχικής διατύπωσης) του κανόνα μάθησης του Kohonen δύναται να κατασκευαστεί ο αλγόριθμος μη επιβλεπόμενης εκπαίδευσης του SOM. Αξίζει να σημειωθεί ότι ο προκύπτων αλγόριθμος -αν και περιορισμένης πολυπλοκότητας- περιλαμβάνει τόσο το στάδιο *διάταξης* (*ordering*) όσο και το στάδιο *ρύθμισης* (*tuning*), κάθε ένα εκ των οποίων υποδιαιρείται στις τρεις φάσεις *ανταγωνισμού*, *συνεργασίας* και *ανταμοιβής*.

Ως ενδεικτικά παραδείγματα σας δίνονται ολοκληρωμένες οι δύο πρώτες συναρτήσεις (**somCreate.m** και **somTrainParameters.m**). Κατά αντιστοιχία με αυτές, σας ζητείται να υλοποιήσετε τις υπόλοιπες.

D διαστάσεις προτύπων, αριθμός χαρακτηριστικών εισόδου
N συνολικό πλήθος των νευρώνων

function somCreate(minMax, gridSize)

% συνάρτηση που κατασκευάζει ένα SOM

minMax πίνακας Dx2 που στην πρώτη στήλη του περιέχει τις ελάχιστες τιμές όλων των χαρακτηριστικών εισόδου, ενώ στην δεύτερη στήλη του τις μέγιστες τιμές.

gridSize πίνακας γραμμή του οποίου το πρώτο στοιχείο περιέχει το πλήθος των νευρώνων ανά γραμμή και το δεύτερο είναι το πλήθος των νευρώνων ανά στήλη.

Η εν λόγω συνάρτηση πρέπει να κατασκευάζει τις ακόλουθες **global** μεταβλητές ώστε να μπορούν να προσπελαστούν από τις υπόλοιπες συναρτήσεις που καθορίζονται στην συνέχεια.

neuronsPerRow το πλήθος των νευρώνων ανά γραμμή (οριζόντια διεύθυνση) που περιέχονται στο πλέγμα του αυτο-οργανούμενου χάρτη.

neuronsPerColumn το πλήθος των νευρώνων ανά στήλη (κατακόρυφη διεύθυνση) που περιέχονται στο πλέγμα του αυτο-οργανούμενου χάρτη.

N το συνολικό πλήθος των νευρώνων.

IW οι παράμετροι/βάρη του SOM είναι ένας πίνακας μεγέθους $N \times D$, κάθε στοιχείο της γραμμής του οποίου αρχικοποιείται σε μία τυχαία τιμή μεταξύ της μέγιστης και ελάχιστης τιμής του αντίστοιχου χαρακτηριστικού εισόδου.

Χρήσιμες Συναρτήσεις: **size**
rand

distances πίνακας $N \times N$ που περιέχει την απόσταση κάθε νευρώνα από όλους του υπόλοιπους νευρώνες, η κύρια διαγώνιος του αποτελείται από μηδενικά, ενώ ο πίνακας είναι συμμετρικός.

Χρήσιμες Συναρτήσεις: **gridtop**, **hextop**, **randtop**, **hexagonaltopology**
(τύπος πλέγματος)
boxdist, **dist**, **linkdist**, **mandist** (μέτρο απόστασης)

function somTrainParameters(setOrderLR,setOrderSteps,setTuneLR)

% συνάρτηση που αρχικοποιεί ή καθορίζει συγκεκριμένες παραμέτρους

% που σχετίζονται με την διαδικασία εκπαίδευσης ενός SOM

setOrderLR η τιμή που τίθεται για τον αρχικό ρυθμό μάθησης κατά το ordering στάδιο της εκπαίδευσης.

setOrderSteps η τιμή που τίθεται για το πλήθος των εποχών κατά το ordering στάδιο της διαδικασίας μη επιβλεπόμενης εκπαίδευσης.

setTuneLR η τιμή που τίθεται για τον ρυθμό μάθησης κατά το tuning στάδιο της εκπαίδευσης.

Η εν λόγω συνάρτηση πρέπει να κατασκευάζει τις ακόλουθες **global** μεταβλητές ώστε να μπορούν να προσπελαστούν από τις υπόλοιπες συναρτήσεις που καθορίζονται στην συνέχεια.

maxNeighborDist μέγιστη απόσταση που εμφανίζεται μεταξύ δύο οποιωνδήποτε νευρώνων.

Χρήσιμες Συναρτήσεις: **max**
ceil

tuneND η απόσταση νευρώνων που χρησιμοποιείται κατά το tuning στάδιο της εκπαίδευσης, συνήθως τιμή είναι το 1.

orderLR ο αρχικός ρυθμός μάθησης κατά το ordering στάδιο της εκπαίδευσης, ενδεικτική τιμή είναι το 0.9.

orderSteps το πλήθος των εποχών κατά το ordering στάδιο της διαδικασίας μη επιβλεπόμενης εκπαίδευσης, ενδεικτική τιμή είναι το 1000.

tuneLR ο ρυθμός μάθησης κατά το tuning στάδιο της εκπαίδευσης, ενδεικτική τιμή είναι το 0.01.

function [output] = somOutput(pattern)

% υπολογισμός της εξόδου ενός SOM

pattern πίνακας Dx1 που περιέχει διατεταγμένα τα χαρακτηριστικά ενός προτύπου εισόδου, δηλαδή ουσιαστικά είναι ένα διάνυσμα στήλη διάστασης D.

output πίνακας Nx1 που περιέχει το σύνολο των εξόδων του SOM, οι τιμές των εξόδων ενός SOM είναι όλες 0 εκτός της τιμής του κόμβου νικητή που είναι 1.

Η εν λόγω συνάρτηση σε πρώτη φάση υπολογίζει την αρνητική Ευκλείδεια απόσταση μεταξύ του προτύπου εισόδου και του διανύσματος παραμέτρων κάθε νευρώνα του SOM, ενώ σε δεύτερη φάση βρίσκει τον νευρώνα με τη μεγαλύτερη τιμή και θέτει την έξοδο του στο 1.

Χρήσιμες Συναρτήσεις: **negdist**
compet

function [a] = somActivation(pattern,neighborDist)

% υπολογισμός της ενεργοποίησης ενός SOM

pattern πίνακας Dx1 που περιέχει διατεταγμένα τα χαρακτηριστικά ενός προτύπου εισόδου, δηλαδή ουσιαστικά είναι ένα διάνυσμα στήλη διάστασης D.

neighborDist η απόσταση (στο πλέγμα του SOM) εντός της οποίας εάν βρίσκεται ένας νευρώνας θεωρείται γειτονικός του νευρώνα νικητή.

a πίνακας Nx1 που συνίσταται από τις τιμές των ενεργοποιήσεων όλων των νευρώνων ενός SOM.

Η παραπάνω συνάρτηση υπολογίζει τις ενεργοποιήσεις όλων των νευρώνων του SOM ως εξής: Ο νευρώνας νικητής λαμβάνει τιμή ενεργοποίησης 1, οι νευρώνες που θεωρούνται γειτονικοί σε αυτόν (δηλαδή αυτοί που βρίσκονται σε απόσταση \leq neighborDist) λαμβάνουν τιμές μικρότερες του 1 (συνήθης τιμή είναι η 0.5) και όλοι οι υπόλοιποι νευρώνες έχουν τιμή ενεργοποίησης 0.

Χρήσιμες Συναρτήσεις: **somOutput**
find

function somUpdate(pattern,learningRate,neighborDist)

% ενημέρωση/ανανέωση του συνόλου των παραμέτρων ενός SOM

pattern πίνακας Dx1 που περιέχει διατεταγμένα τα χαρακτηριστικά ενός προτύπου εισόδου, δηλαδή ουσιαστικά είναι ένα διάνυσμα στήλη διάστασης D.

learningRate ο ρυθμός μάθησης κατά το παρόν βήμα της τροποποίησης των βαρών του SOM.

neighborDist η απόσταση (στο πλέγμα του SOM) εντός της οποίας εάν βρίσκεται ένας νευρώνας θεωρείται γειτονικός του νευρώνα νικητή.

Αυτή η συνάρτηση τροποποιεί/ενημερώνει τα βάρη (ή αλλιώς τις παραμέτρους) ενός SOM βάσει του αλγορίθμου μάθησης του Kohonen. Επιγραμματικά ο κανόνας μάθησης του Kohonen είναι ο εξής:

$$\Delta w_i = \eta a_i (x - w_i)$$

όπου ο δείκτης i υποδεικνύει κάθε νευρώνα του SOM, w_i το διάνυσμα παραμέτρων του νευρώνα i , x το πρότυπο εισόδου, a_i η ενεργοποίηση του νευρώνα i και η ο ρυθμός μάθησης.

Χρήσιμες Συναρτήσεις: **somActivation**

P συνολικό πλήθος των προτύπων εκπαίδευσης

function somTrain(patterns)

% συνάρτηση που εκπαιδεύει ένα SOM

patterns πίνακας DxP που περιέχει το σύνολο των προτύπων εκπαίδευσης για το SOM (training set).

Η εν λόγω συνάρτηση εκπαιδεύει ένα SOM παρουσιάζοντας κάθε πρότυπο, προσδιορίζοντας τον νευρώνα νικητή (ανταγωνισμός), υπολογίζοντας τις ενεργοποιήσεις (συνεργασία) και τροποποιώντας/ανανεώνοντας τις παραμέτρους του SOM βάσει του αλγορίθμου μάθησης του Kohonen (ανταμοιβή). Αυτή η διαδικασία γίνεται ακολουθιακά και ξεχωριστά για τα πρότυπα κάθε εποχής εκπαίδευσης. Γενικότερα, η εκπαίδευση περιλαμβάνει αμφότερα τα στάδια ordering και tuning, κάθε ένα εκ των οποίων διαρκεί έναν ορισμένο αριθμό εποχών.

Το στάδιο ordering διαρκεί orderSteps εποχές. Η απόσταση εντός της οποίας ο νευρώνας νικητής και κάθε άλλος νευρώνας θεωρούνται γειτονικοί, ξεκινάει ως η μέγιστη απόσταση μεταξύ δυο οποιωνδήποτε νευρώνων (μεταβλητή maxNeighborDist) και μειώνεται (εκθετικά) μέχρι την τιμή tuneND. Αντίστοιχα, η αρχική τιμή του ρυθμού μάθησης είναι orderLR και μειώνεται ομοίως (εκθετικά) μέχρι την τιμή tuneLR.

Το στάδιο tuning διαρκεί σαφώς περισσότερες εποχές από ό,τι το στάδιο ordering (συνήθως κατά ένα συντελεστή από 2 έως 5). Η απόσταση εντός της οποίας ο νευρώνας νικητής και κάθε άλλος νευρώνας θεωρούνται γειτονικοί είναι σταθερή και ίση με tuneND. Ο ρυθμός μάθησης είτε διατηρείται αμετάβλητος και ίσος με tuneLR, είτε με αρχική τιμή την tuneLR μειώνεται με ιδιαίτερα αργό τρόπο.

Χρήσιμες Συναρτήσεις:

size
linspace
somUpdate

2. Μελέτη και ανάλυση SOM

2A. Μετά την υλοποίηση του SOM, θα επιχειρήσουμε τη μελέτη και την ανάλυση των ιδιοτήτων, των δυνατοτήτων και των επιδόσεων του αυτο-οργανούμενου χάρτη. Η χρησιμοποιούμενη μεθοδολογία, όπως θα φανεί και στη συνέχεια, είναι η πειραματική διερεύνηση/αντιμετώπιση χαρακτηριστικών προβλημάτων με στόχο να διαφωτιστούν σημεία που χρήζουν κάποιας εξέτασης και σχολιασμού. Κατά κύριο λόγο χρησιμοποιούνται προβλήματα χαμηλών διαστάσεων, έτσι ώστε να παρέχεται η δυνατότητα (μέσω της οδού της οπτικοποίησης) για εξαγωγή συμπερασμάτων και για βαθύτερη κατανόηση των επιτελούμενων λειτουργιών.

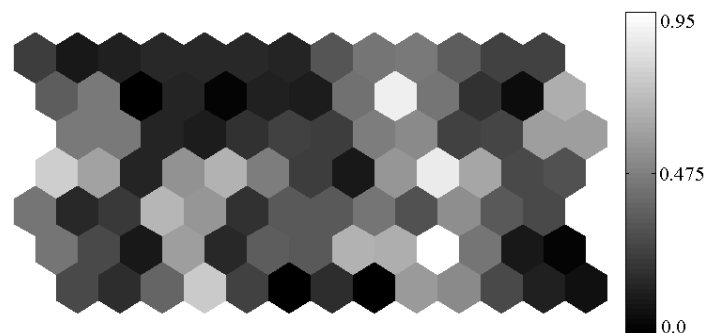
Μέσω των m-files **CrossData.m** και **RingData.m** σας παρέχονται δύο διαφορετικά δισδιάστατα σύνολα δεδομένων. Γενικότερα, κάνοντας χρήση της συνάρτησης **plot2DSomData.m** (ή κάποιας άλλης που θα κατασκευάσετε εσείς) είναι δυνατόν να οπτικοποιήσετε σύνολα δεδομένων και χάρτες (σχετιζόμενους με τα υπό εξέταση σύνολα δεδομένων). Για να ελέγξετε και να παρατηρήσετε την λειτουργία του SOM που υλοποιήσατε, γράψτε και εκτελέστε scripts που κατασκευάζουν μονοδιάστατα ή δισδιάστατα πλέγματα και εκπαιδεύονται βάσει των παραπάνω συνόλων προτύπων. Μπορείτε να πειραματιστείτε με παραμέτρους του SOM και παράλληλα είναι δυνατό να απεικονίζετε την διαδικασία της εκπαίδευσης. Είναι ενδιαφέρον να παρατηρήσετε ότι το SOM επιτυγχάνει (μετά το πέρας της διαδικασίας της μη επιβλεπόμενης μάθησης) να

ανακαλύψει καθώς και να περιγράψει τόσο τη χωρική κατανομή όσο και την κατανομή πυκνότητας των προτύπων του εκάστοτε συνόλου δεδομένων.

2B. Μία από τις βασικές λειτουργίες που καλείται να επιτελέσει το SOM (ως ένα κλασικό δίκτυο μη επιβλεπόμενης μάθησης) είναι η **ομαδοποίηση προτύπων (clustering)** ή αλλιώς η **μη επιβλεπόμενη ταξινόμηση (unsupervised classification)**. Στόχος του clustering είναι η αντιστοίχιση προτύπων προερχόμενων από σύνολα δεδομένων σε ομάδες (συστάδες) σύμφωνα με ορισμένα κριτήρια ομοιότητας. Ο διαχωρισμός των αρχικών συνόλων σε ομάδες γίνεται στην βάση της μεγιστοποίησης της ομοιότητας μεταξύ των προτύπων της ίδιας ομάδας και στην ελαχιστοποίησης της ομοιότητας μεταξύ προτύπων που ανήκουν σε διαφορετικές ομάδες.

Το SOM, εκτός των δυνατοτήτων που προσφέρει για ομαδοποίηση προτύπων, είναι ένα αποδοτικό εργαλείο για **οπτικοποίηση δεδομένων μεγάλων διαστάσεων**. Στη βασική του μορφή, ουσιαστικά μετατρέπει τις μη γραμμικές στατιστικές συσχετίσεις προτύπων μεγάλης διάστασης σε απλές γεωμετρικές συσχετίσεις σημείων ενός επιπέδου μικρής διάστασης (το οποίο συνήθως είναι ένα δισδιάστατο πλέγμα νευρώνων/κόμβων). Κατά συνέπεια, καθώς το SOM συμπιέζει την πληροφορία διατηρώντας στις απεικονίσεις τις σημαντικότερες τοπολογικές και μετρικές συσχετίσεις των δεδομένων, μπορεί να υποστηριχθεί ότι παράγει ένα είδος **αφαίρεσης**. Αυτά τα δύο χαρακτηριστικά, η οπτικοποίηση και η αφαίρεση, σε συνδυασμό με την ομαδοποίηση προτύπων, είναι δυνατό να χρησιμοποιηθούν σε μία σειρά πολύπλοκων και σύνθετων εργασιών.

Μία μεθοδολογία που ενοποιεί τις τρεις αυτές ιδιότητες/δυνατότητες του SOM είναι η γραφική απεικόνιση που καλείται **πίνακας ενοποιημένων αποστάσεων (unified distance matrix ή αλλιώς U-matrix)**. Βάσει αυτής της μεθοδολογίας υπολογίζονται οι αποστάσεις μεταξύ των διανυσμάτων παραμέτρων (βαρών) των γειτονικών νευρώνων και στην συνέχεια αναπαριστώνται ως αποχρώσεις επί ενός επιπέδου. Παράλληλα, η απόχρωση καθενός νευρώνα καθορίζεται από τη μέση τιμή των αποστάσεων του νευρώνα αυτού από όλους τους γειτονικούς σε αυτόν νευρώνες.



Σχήμα: Ο U-matrix ενός SOM με εξαγωνικό πλέγμα διάστασης 7x4.

Μέσω του m-file **GroupData.m** σας δίδεται ένα δισδιάστατο σύνολο δεδομένων, με πρότυπα τα οποία ανήκουν σε δυο ομάδες. Το τρίτο χαρακτηριστικό του διανύσματος-στήλη κάθε δεδομένου περιγράφει την ομάδα στην οποία ανήκει το εν λόγω πρότυπο (δηλαδή παρέχει την πληροφορία εάν είναι ομάδας 1 ή 2).

Αφού έχετε καταλήξει προσεγγιστικά σε ένα σύνολο παραμέτρων, σας ζητείται αφενός να απεικονίσετε το παραπάνω σύνολο δεδομένων με χρήση του U-matrix και αφετέρου να διερευνήσετε και να **αναλύσετε το κατά πόσο υπάρχει συσχέτιση μεταξύ του πλήθους των προτύπων κάθε ομάδας και του πλήθους των νευρώνων που ανατίθενται στην**

εν λόγω ομάδα. Η προαναφερθείσα διαδικασία ενδείκνυται να γίνει για όλους τους τύπους πλέγματος και για όλες τις διαφορετικές τοπολογικές γειτονιές που διατίθενται.

Θεωρείται ότι ένας αυτοοργανούμενος χάρτης, μετά την εκπαίδευση, αντικατοπτρίζει την τοπολογία και την κατανομή των προτύπων εισόδου. Ουσιαστικά, αυτό που ζητείται είναι η πειραματική επαλήθευση ή η απόρριψη της πρότασης που διατυπώνεται για συσχέτιση μεταξύ της πυκνότητας μιας ομάδας προτύπων και της πυκνότητας των νευρώνων που περιγράφουν την τοπολογία και τις ιδιότητες αυτής της ομάδας.

Υπόδειξη: Αφού εκπαιδευτεί το SOM για τα πρότυπα εκπαίδευσης ελέγξτε τις τιμές των διανυσμάτων βαρών/παραμέτρων των νευρώνων που αφορούν στην τελευταία στήλη (που περιέχει την πληροφορία της ομάδας του προτύπου).

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

Στα προηγούμενα μέρη της άσκησης (1 και 2), κατανοήσατε τον τρόπο λειτουργίας του αυτο-οργανούμενου χάρτη, το εύρος των παραμέτρων που μπορούν να χρησιμοποιηθούν και μερικές από τις ιδιότητές του. Στο τρίτο (και τελευταίο μέρος της άσκησης) καλείστε να εφαρμόσετε τον αυτο-οργανούμενο χάρτη που κατασκευάσατε, σε ένα πραγματικό πρόβλημα ομαδοποίησης εγγράφων.

Το πρόβλημα στοιχειοθετείται ως εξής :

Διαθέτουμε 500 έγγραφα που αποτελούν δημοσιεύσεις στο συνέδριο NIPS (Neural Information Processing Systems). Θέλουμε να διακρίνουμε τις ομάδες στις οποίες μπορούν να χωριστούν αυτά τα έγγραφα και πιο συγκεκριμένα ποιοι όροι (λέξεις) είναι πιο σημαντικοί για κάθε τέτοια ομάδα. Για την αναπαράσταση των εγγράφων χρησιμοποιείται το μοντέλο “Bag-of-Words”, στο οποίο αγνοείται η σειρά των λέξεων και μας ενδιαφέρει μόνο η συχνότητα εμφάνισής τους. Αφαιρώντας τις πολύ συχνές λέξεις (and, to, the κτλ) καταλήγουμε σε ένα σύνολο 8296 διαφορετικών λέξεων που απαντώνται στα παραπάνω 500 έγγραφα. Για κάθε έγγραφο υπολογίζουμε ένα διάνυσμα διάστασης 8296, κάθε συνιστώσα του οποίου αντιστοιχεί σε έναν διαφορετικό όρο και η τιμή της δείχνει τον αριθμό εμφανίσεων του συγκεκριμένου όρου στο έγγραφο. Για παράδειγμα :

	service	photography	utility	fun	software	art	imported	list	apple	resource
doc1	2	0	1	0	1	0	2	5	0	4
doc2	1	2	0	1	4	1	1	7	1	1
doc3	0	4	2	0	0	0	0	3	1	1

Το έγγραφο doc1 περιέχει 5 φορές τον όρο list, 2 φορές τον όρο service, καμία φορά τον όρο photography κ.ο.κ. Αντίστοιχα για τα έγγραφα doc2 και doc3.

Για την καλύτερη ανάθεση βαρών σε κάθε όρο, χρησιμοποιούμε το σχήμα tf-idf (term-frequency / inverse-document-frequency), στο οποίο οι συχνότητες των λέξεων (όπως φαίνονται π.χ. στον παραπάνω πίνακα) κανονικοποιούνται με βάση το άθροισμα των τιμών (π.χ. στο doc1 θα είναι η τιμή 15 και στο doc3 θα είναι η τιμή11). Έτσι έχουμε για κάθε όρο i σε κάθε έγγραφο j την τιμή tf:

$$tf(i, j) = \frac{f(i, j)}{\sum_i f(i, j)}$$

Αντίστοιχα, κανονικοποιούμε και με βάση τη συχνότητα εμφάνισης ενός όρου σε ολόκληρη τη συλλογή εγγράφων, με στόχο να δώσουμε μικρά βάρη σε όρους που εμφανίζονται πολλές φορές σε πολλά έγγραφα (όπως για παράδειγμα ο όρος *list* στα παραπάνω έγγραφα). Έτσι έχουμε για κάθε όρο i την εξής τιμή idf :

$$idf(i) = \log\left(\frac{N}{df_i}\right)$$

Όπου : N είναι ο συνολικός αριθμός εγγράφων στη συλλογή μας,
 df_i είναι ο αριθμός των εγγράφων που περιέχουν τον όρο i ,
 η λογαριθμική συνάρτηση χρησιμοποιείται για την αναγωγή της τιμής σε παρόμοια μεγέθη με την τιμή του tf .

Στο παραπάνω έγγραφο θα είχαμε τους ακόλουθους υπολογισμούς :

	service	photography	utility	fun	software	art	imported	list	apple	resource
$tf(doc1)$	0.133	0.000	0.067	0.000	0.067	0.000	0.133	0.333	0.000	0.267
$tf(doc2)$	0.053	0.105	0.000	0.053	0.211	0.053	0.053	0.368	0.053	0.053
$tf(doc3)$	0.000	0.364	0.182	0.000	0.000	0.000	0.000	0.273	0.091	0.091
idf	0.176	0.176	0.176	0.477	0.176	0.477	0.176	0.000	0.176	0.000

Έτσι τελικά, το βάρος κάθε όρου i στο έγγραφο j δίνεται από τη σχέση:

$$w(i, j) = tf(i, j) \cdot idf(i)$$

Για τα παραπάνω έγγραφα θα έχουμε τις εξής τιμές :

	service	photography	utility	fun	software	art	imported	list	apple	resource
$w(doc1)$	0.023	0.000	0.012	0.000	0.012	0.000	0.023	0.000	0.000	0.000
$w(doc2)$	0.009	0.019	0.000	0.025	0.037	0.025	0.009	0.000	0.009	0.000
$w(doc3)$	0.000	0.064	0.032	0.000	0.000	0.000	0.000	0.000	0.016	0.000

Μέσω του αρχείου NIPS500.mat σας δίνονται :

- Ο πίνακας tf (διάστασης 500x8296) που περιέχει τις συχνότητες εμφάνισης των 8296 λέξεων στα 500 έγγραφα (ο πίνακας δίνεται σε αραιή (sparse) μορφή),
- Ο πίνακας $terms$ που περιέχει τα ονόματα των 8296 όρων,
- Ο πίνακας $titles$ που περιέχει τους τίτλους των 500 άρθρων.

Ζητούνται :

3A. Εφαρμόζοντας τη συνάρτηση $tfidf.m$ υπολογίστε τον τελικό πίνακα βαρών ($tf-idf$) για τα 500 έγγραφα του NIPS500.

3B. Χρησιμοποιώντας τις συναρτήσεις του πρώτου μέρους της άσκησης και κατάλληλες τιμές παραμέτρων για εξαγωγικό πλέγμα, εκπαιδεύστε έναν αυτο-οργανούμενο χάρτη για τα 500 έγγραφα που σας δίνονται. Αποτυπώστε με χρήση της συνάρτησης **somShow.m** τον U-matrix. Μπορείτε να εκτιμήσετε τις πιθανές ομάδες που θα μπορούσαν να χωριστούν τα έγγραφα;

3Γ. Μετά την κατάλληλη εκπαίδευση του SOM απαντήστε στις παρακάτω ερωτήσεις :

- i. Για κάθε νευρώνα να υπολογιστεί το ποσοστό των εγγράφων που ανήκουν στον νευρώνα.
- ii. Για κάθε νευρώνα να βρεθούν οι 2 όροι (με χρήση του πίνακα terms) που έχουν το μεγαλύτερο βάρος στο νευρώνα και να αποτυπωθούν το ονόματά τους στο πλέγμα (στη θέση του αντίστοιχου νευρώνα).
*Υπόδειξη: Χρήση του πίνακα **positions** και των συναρτήσεων **scatter**, **text** της MATLAB (ή άλλων εφόσον εσείς το επιθυμείτε).*
- iii. Να βρεθούν οι νευρώνες του χάρτη για τους οποίους οι όροι “machin” και “learn” έχουν τιμή μεγαλύτερη από το 50% της μέγιστης τιμής.
- iv. Να υπολογιστεί η μέση τιμή του βάρους κάθε όρου για τους νευρώνες του ερωτήματος (iii) ως ποσοστό της μέγιστης τιμής για όλους τους νευρώνες του χάρτη.
- v. Να ιεραρχηθούν οι όροι για την παραπάνω ομάδα νευρώνων από τον σημαντικότερο έως τον πιο ασήμαντο βάσει της μέσης τιμής που υπολογίστηκε στο ερώτημα (iv).

4. Επικοινωνία

Η εκφώνηση της άσκησης και τα απαραίτητα συνοδευτικά αρχεία βρίσκονται στο site του μαθήματος <http://mycourses.ntua.gr/courses/ECE1078/>.

Για οποιαδήποτε απορία σχετικά με τη 2^η εργαστηριακή άσκηση μπορείτε να απευθύνεστε στο Εργαστήριο Ευφυών Συστημάτων (Αίθουσα 1.1.26, Παλαιά Κτήρια Ηλεκτρολόγων) ή με email στο courses@islab.ntua.gr.