Υπολογιστική Νοημοσύνη
Εργαστηριακές Ασκήσεις ακ. έτους 2025-26

Η πρώτη άσκηση έχει βαρύτητα 20% και η δεύτερη 10%.

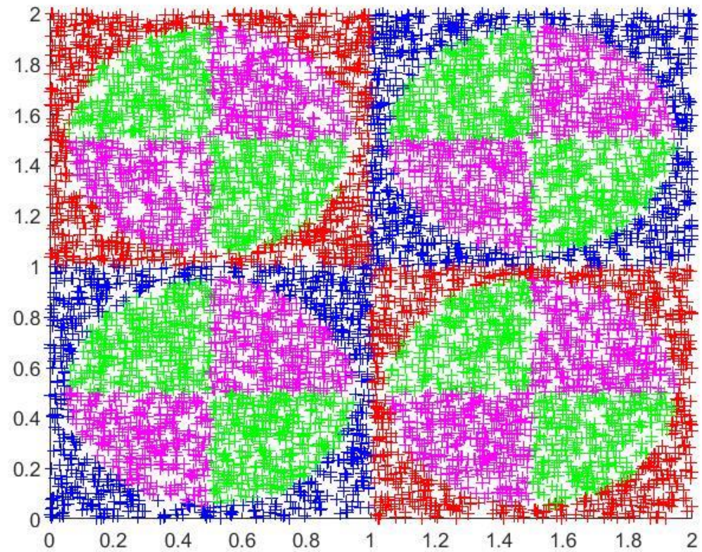
- Ομάδες με 2 ή 3 μέλη.
- **Δήλωση ομάδων**: **e-mail στον διδάσκοντα με (ΑΜ, ονοματεπώνυμο) μελών της ομάδας μέχρι 5 Δεκεμβρίου 2025** (αυστηρή προθεσμία). Η δήλωση συνεπάγεται υποχρέωση υποβολής των εργασιών.
- **Προθεσμία υποβολής των εργασιών: Τρίτη 23 Δεκεμβρίου 2025.**
- **Εξέταση εργασιών (διά ζώσης στο εργαστήριο): 12-13 Ιανουαρίου 2026.**
- Γλώσσα προγραμματισμού: **C ή Java (όχι Python).**
- Η υποβολή θα γίνει με χρήση της εντολής **turnin** ως εξής:
  turnin assignment@mye035 <your_filename>
- Δεκτές για εξέταση γίνονται μόνο ασκήσεις που είναι ολοκληρωμένες, δηλ. **τα προγράμματα θα πρέπει άμεσα να μεταγλωττίζονται και να εκτελούνται στους υπολογιστές του Τμήματος** (π.χ. opti3060ws03). Να αναφέρετε και την εντολή μεταγλώττισης για την παραγωγή του εκτελέσιμου από τον πηγαίο κώδικα.
- Θα δημιουργήσετε δύο καταλόγους, έναν για κάθε άσκηση. Κάθε κατάλογος θα περιλαμβάνει τον πηγαίο και τον εκτελέσιμο κώδικα της άσκησης, καθώς και ένα report (.pdf) με αποτελέσματα και συμπεράσματα σχετικά με την άσκηση. Υποβολές χωρίς report δεν βαθμολογούνται.
- Μην υποβάλετε συμπιεσμένα αρχεία .rar
- Θα πρέπει να υπάρχει πληροφορία για τα ονόματα και τα ΑΜ των μελών της ομάδας.

Εκφώνηση Ασκήσεων

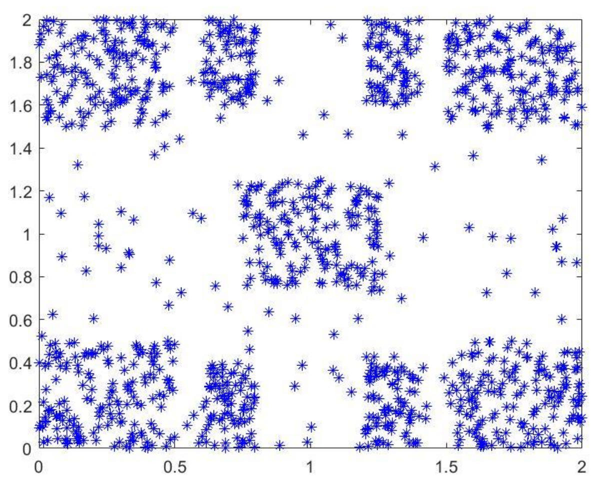
Να κατασκευάσετε σύνολα δεδομένων ΣΔΤ και ΣΔΟ για τα ακόλουθα προβλήματα:

ΣΔΤ) Πρόβλημα ταξινόμησης τεσσάρων κατηγοριών: Θα δημιουργήσετε **τυχαία** 8000 παραδείγματα (σημεία (x_1,x_2) στο επίπεδο) μέσα στο τετράγωνο [0,2]x[0,2] (4000 για το σύνολο εκπαίδευσης και 4000 για το σύνολο ελέγχου). Στη συνέχεια θα κατατάξετε κάθε παράδειγμα (x_1,x_2) (από τα 8000 παραδείγματα) σε μια κατηγορία από τέσσερις κατηγορίες ως εξής:

1) εάν $(x_1 - 0.5)^2 + (x_2 - 0.5)^2 < 0.2$, και $x_1 > 0.5$ και $x_2 > 0.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C1,
2) εάν $(x_1 - 0.5)^2 + (x_2 - 0.5)^2 < 0.2$, και $x_1 < 0.5$ και $x_2 > 0.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C2,
3) εάν $(x_1 - 0.5)^2 + (x_2 - 0.5)^2 < 0.2$, και $x_1 > 0.5$ και $x_2 < 0.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C2,
4) εάν $(x_1 - 0.5)^2 + (x_2 - 0.5)^2 < 0.2$, και $x_1 < 0.5$ και $x_2 < 0.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C1,
5) εάν $(x_1 - 1.5)^2 + (x_2 - 0.5)^2 < 0.2$, και $x_1 > 1.5$ και $x_2 > 0.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C1,
6) εάν $(x_1 - 1.5)^2 + (x_2 - 0.5)^2 < 0.2$, και $x_1 < 1.5$ και $x_2 > 0.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C2,
7) εάν $(x_1 - 1.5)^2 + (x_2 - 0.5)^2 < 0.2$, και $x_1 > 1.5$ και $x_2 < 0.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C2,
8) εάν $(x_1 - 1.5)^2 + (x_2 - 0.5)^2 < 0.2$, και $x_1 < 1.5$ και $x_2 < 0.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C1,
9) εάν $(x_1 - 0.5)^2 + (x_2 - 1.5)^2 < 0.2$, και $x_1 > 0.5$ και $x_2 > 1.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C1,
10) εάν $(x_1 - 0.5)^2 + (x_2 - 1.5)^2 < 0.2$, και $x_1 < 0.5$ και $x_2 > 1.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C2,
11) εάν $(x_1 - 0.5)^2 + (x_2 - 1.5)^2 < 0.2$, και $x_1 > 0.5$ και $x_2 < 1.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C2,
12) εάν $(x_1 - 0.5)^2 + (x_2 - 1.5)^2 < 0.2$, και $x_1 < 0.5$ και $x_2 < 1.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C1,
13) εάν $(x_1 - 1.5)^2 + (x_2 - 1.5)^2 < 0.2$, και $x_1 > 1.5$ και $x_2 > 1.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C1,
14) εάν $(x_1 - 1.5)^2 + (x_2 - 1.5)^2 < 0.2$, και $x_1 < 1.5$ και $x_2 > 1.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C2,
15) εάν $(x_1 - 1.5)^2 + (x_2 - 1.5)^2 < 0.2$, και $x_1 > 1.5$ και $x_2 < 1.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C2,
16) εάν $(x_1 - 1.5)^2 + (x_2 - 1.5)^2 < 0.2$, και $x_1 < 1.5$ και $x_2 < 1.5$ τότε το (x_1,x_2) κατατάσσεται στην κατηγορία C1,
17) εάν δεν ισχύει κάποια από τις συνθήκες 1 έως 16 και $(x_1-1)(x_2-1) > 0$, το (x_1,x_2) κατατάσσεται στην κατηγορία C3,
18) εάν δεν ισχύει κάποια από τις συνθήκες 1 έως 16 και $(x_1-1)(x_2-1) < 0$, το (x_1,x_2) κατατάσσεται στην κατηγορία C4.



~~ΣΔ~~Ο) Πρόβλημα ομαδοποίησης εννέα ομάδων (1200 παραδείγματα): δημιουργούμε **τυχαία** σημεία (x_1,x_2) στο επίπεδο ως εξής: 1) 150 σημεία στο τετράγωνο [0.75,1.25]x[0.75,1.25], 2) 150 σημεία στο τετράγωνο [0, 0.5]x[0,0.5], 3) 150 σημεία στο τετράγωνο [0,0.5] x [1.5,2], 4) 150 σημεία στο τετράγωνο [1.5,2]x[0,0.5], 5) 150 σημεία στο τετράγωνο [1.5,2] x [1.5,2], 6) 75 σημεία στο τετράγωνο [0.6,0.8]x[0,0.4], 7) 75 σημεία στο τετράγωνο [0.6,0.8] x [1.6,2], 8) 75 σημεία στο τετράγωνο [1.2,1.4]x[0,0.4], 9) 75 σημεία στο τετράγωνο [1.2,1.4] x [1.6,2], 10) 150 σημεία στο τετράγωνο [0,2]x[0,2].



Ασκηση 1: Να κατασκευάσετε **πρόγραμμα ταξινόμησης** (**ΠΤ**) βασισμένο στο **πολυεπίπεδο perceptron** (MLP) με **τρία κρυμμένα επίπεδα.**

Οι συναρτήσεις ενεργοποίησης ορίζονται ως εξής: i) στα κρυμμένα επίπεδα: λογιστική, υπερβολική εφαπτομένη (tanh(u)) ή relu και ii) για το επίπεδο εξόδου θα ορίσετε εσείς τη συνάρτηση ενεργοποίησης που απαιτείται για το συγκεκριμένο πρόβλημα.

Το πρόγραμμα θα πρέπει να αποτελείται από τις ακόλουθες μονάδες:

1) Με χρήση της εντολής define, καθορισμός αριθμού εισόδων (d), αριθμού κατηγοριών (K), αριθμού νευρώνων στο πρώτο κρυμμένο επίπεδο (H1), αριθμού νευρώνων στο δεύτερο κρυμμένο επίπεδο (H2),

αριθμού νευρώνων στο τρίτο κρυμμένο (H3) και είδος συνάρτησης ενεργοποίησης για τα κρυμμένα επίπεδα.
2) Φόρτωμα των συνόλων εκπαίδευσης και ελέγχου (από αντίστοιχα αρχεία) και κωδικοποίηση των κατηγοριών (ορισμός των επιθυμητών εξόδων για κάθε κατηγορία).
3) Καθορισμός της αρχιτεκτονικής του δικτύου MLP. Ορισμός των απαιτούμενων πινάκων και άλλων δομών ως καθολικών μεταβλητών. Καθορισμός του ρυθμού μάθησης και του κατωφλίου τερματισμού. Τυχαία αρχικοποίηση των βαρών/πολώσεων στο διάστημα (-1,1).
4) Υλοποίηση της συνάρτησης forward-pass (float *x, int d, float *y, int K) η οποία υπολογίζει το διάνυσμα εξόδου y (διάστασης K) του MLP δοθέντος του διανύσματος εισόδου x (διάστασης d).
5) Υλοποίηση της συνάρτησης backprop(float *x, int d, float *t, int K) η οποία λαμβάνει τα διανύσματα x διάστασης d (είσοδος) και t διάστασης K (επιθυμητή έξοδος) και υπολογίζει τις παραγώγους του σφάλματος ως προς οποιαδήποτε παράμετρο (βάρος ή πόλωση) του δικτύου ενημερώνοντας τους αντίστοιχους πίνακες.
6) Χρησιμοποιώντας τα παραπάνω να υλοποιήσετε τον **αλγόριθμο εκπαίδευσης gradient descent και ενημέρωση των βαρών ανά ομάδες των L παραδειγμάτων (mini-batches) θεωρώντας τα N παραδείγματα του συνόλου εκπαίδευσης** (όπου το L διαιρέτης του N και ορίζεται στην αρχή του προγράμματος). Σημειώστε ότι εάν L=1 έχουμε σειριακή ενημέρωση, ενώ εάν L=N έχουμε ομαδική ενημέρωση.
Στο τέλος κάθε εποχής θα πρέπει υποχρεωτικά να υπολογίζετε και να τυπώνετε την τιμή του συνολικού σφάλματος εκπαίδευσης. Τερματίζουμε όταν η διαφορά της τιμής του σφάλματος εκπαίδευσης μεταξύ δύο εποχών γίνει μικρότερη από κάποιο κατώφλι, αφού όμως ο αλγόριθμος έχει τρέξει για τουλάχιστον 800 εποχές.
7) Αφού τερματιστεί η εκπαίδευση του δικτύου να γίνεται υπολογισμός και εκτύπωση της **ικανότητας γενίκευσης** του δικτύου που προκύπτει, υπολογίζοντας το **ποσοστό σωστών αποφάσεων στο σύνολο ελέγχου**.

Χρησιμοποιώντας το πρόγραμμα ΠΤ να μελετήσετε το πρόβλημα ταξινόμησης ΣΔΤ.

- Να εξετάσετε και να **καταγράψετε σε ένα πίνακα** πώς μεταβάλλεται η γενικευτική ικανότητα του δικτύου (ποσοστό επιτυχίας στο σύνολο ελέγχου) θεωρώντας:
- α) Διάφορους συνδυασμούς τιμών για τα H1, H2, H3.
- β) Συνάρτηση ενεργοποίησης υπερβολική εφαπτομένη στο πρώτο και στο δεύτερο κρυμμένο επίπεδο και υπερβολική εφαπτομένη ή λογιστική ή relu στο τρίτο κρυμμένο επίπεδο.
- γ) B= N/10, N/20, N/100, N/200.
- Για το δίκτυο με την καλύτερη γενικευτική ικανότητα που θα βρείτε, να τυπώσετε τα παραδείγματα του συνόλου ελέγχου χρησιμοποιώντας διαφορετικό στυλ (πχ + και -) ανάλογα με το αν το παράδειγμα ταξινομείται από το δίκτυο στη σωστή κατηγορία ή όχι.
- Μπορείτε να κάνετε κάποιο σχόλιο σχετικά με την θέση των σφαλμάτων;

Ασκηση 2: Να κατασκευάσετε **πρόγραμμα ομαδοποίησης (ΠΟ) με M ομάδες** (το M θα ορίζεται με την εντολή #define) βασισμένο στον **αλγόριθμο k-means**. Το πρόγραμμα θα φορτώνει το αρχείο με τα παραδείγματα, θα εκτελεί τον αλγόριθμο k-means με M κέντρα και στο τέλος θα αποθηκεύει τις συντεταγμένες των κέντρων των ομάδων. Η αρχική θέση κάθε κέντρου να γίνεται επιλέγοντας τυχαία κάποιο από τα παραδείγματα. Επίσης θα πρέπει στο τέλος να υπολογίζεται **και να τυπώνεται το σφάλμα ομαδοποίησης ως εξής: για κάθε παράδειγμα x_i υπολογίζουμε την Ευκλείδεια απόσταση $||x_i-\mu_k||^2$ από το κέντρο μ_k της ομάδας στην οποία ανήκει και αθροίζουμε τις αποστάσεις για όλα τα παραδείγματα x_i.**

Να εκτελέσετε το πρόγραμμα ομαδοποίησης (ΠΟ) στο σύνολο δεδομένων (ΣΔΟ) για M=3,5,7,9,11,13 ομάδες. Για **κάθε τιμή του M** να κάνετε τα εξής:

α) Να εκτελέσετε 50 τρεξίματα του προγράμματος με διαφορετικά (τυχαία επιλεγμένα αρχικά κέντρα), να τυπώνετε το σφάλμα ομαδοποίησης καθεμίας από τις 50 λύσεις, και να αποθηκεύσετε (τα κέντρα και την τμή σφάλματος) για τη **λύση με το μικρότερο σφάλμα ομαδοποίησης**, καθώς και την τιμή του σφάλματος.
β) Στη συνέχεια να εμφανίσετε (plot) στο ίδιο σχήμα τόσο τα παραδείγματα (π.χ. με '+') όσο και τις θέσεις των κέντρων που βρήκατε (π.χ. με '*').
Βάσει των αποτελεσμάτων να φτιάξετε ένα διάγραμμα που να δείχνει πώς μεταβάλλεται το σφάλμα ομαδοποίησης με τον αριθμό των ομάδων. Μπορεί να χρησιμοποιηθεί το σφάλμα ομαδοποίησης για να εκτιμήσουμε τον πραγματικό αριθμό ομάδων; (στο σύνολο ΣΔΟ ο πραγματικός αριθμός των ομάδων είναι 9).